A Bayesian Method for Simultaneous Registration and Clustering of Functional Observations

Zizhen Wu[a], David B. Hitchcock[a]

[a]*Department of Statistics, University of South Carolina, Columbia, SC*

Abstract

We develop a Bayesian method that simultaneously registers and clusters functional data of interest. Unlike other existing methods, which often assume a simple translation in the time domain, our method uses a discrete approximation generated from the family of Dirichlet distributions to allow warping functions of great flexibility. Under this Bayesian framework, a MCMC algorithm is proposed for posterior sampling. We demonstrate this method via simulation studies and applications to growth curve data and cell cycle regulated yeast genes.

Keywords: functional data, time warping, curve registration

1. Introduction

An important example of exploratory data analysis, cluster analysis involves grouping observations that share similar characteristics. In many clustering methods, similarities or dissimilarities between pairs of observations are measured by some relevant distance metric. Methods based on dissimilarity measures include hierarchical clustering and the K-medoids method (Everitt et al., 2011). Another major clustering technique, model-based clustering, requires statistical assumptions about the observations. A popular model-based method (Fraley and Raftery, 2002) assumes a multivariate normal distribution for the measurements and assigns objects to clusters by comparing the posterior group probabilities given the observations.

If the observations to be clustered are functional data, i.e., repeated measures over time or some other domain, one may consider fitting a function to each observational unit using some basis function expansion (Ramsay and Silverman, 2005). Throughout this paper, we use the B-spline basis (De Boor, 2001). One advantage of functional data analysis over traditional multivariate analysis is the ability to examine higher-order derivatives of fitted functions. For example, the first-order derivative of a fitted monotone smoothing function (Ramsay and Silverman, 2005) measuring children's height over a given period represents the estimated growth velocity, and the second-order derivative is the estimated growth acceleration, etc.

Recently, several methods have been developed for clustering functional data. Luan and Li (2003) use mixed-effect models for time-course gene expression and cluster the curves by calculating their posterior cluster probabilities via the EM algorithm. This mixed-effect model is a special case of the model proposed by James and Sugar (2003).

A more challenging yet often-encountered problem is clustering the observations in the presence of time distortions, also known as phase variation. The time distortion is usually

modeled by a warping function $h(\cdot)$ (Ramsay and Silverman, 2005), which is a non-decreasing continuous function defined on the time domain $\mathcal{T}$ satisfying the endpoint conditions $h(a) = a$, and $h(b) = b$, where a and b are two endpoints of the time domain. Figure 1 shows eight warping functions, while the bold dashed line is the 45° reference line representing an identity warp. The cluster structure is blurred by the effect of the time distortions, which should be eliminated for the purpose of clustering. However, it is not feasible to estimate the warping functions without knowing the cluster memberships.

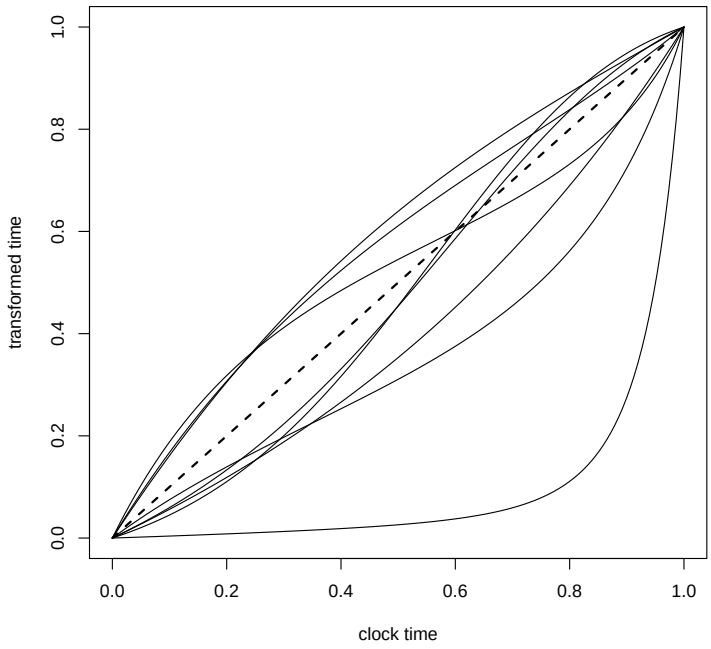

Figure 1: Examples of warping functions

Liu and Yang (2009) propose the SACK model, which is capable of clustering functional data when a simple time translation is presented. They translate the shift in the time domain into variation in the measurement space by a first-order Taylor expansion on the B-spline basis functions. The conditional cluster probabilities are calculated via the EM algorithm. Also assuming a simple time translation, Sangalli et al. (2010) propose an iterative method based on a dissimilarity measure called k-mean alignment, which iterates among a template identification step, alignment and cluster step, and a normalization step until convergence.

To handle more realistic scenarios under arbitrary time warpings, Tang and Müller (2009) propose a method based on pairwise warping. However, this method assumes the mean curves in different clusters are well separated vertically to some degree, a condition potentially too strong for some applications. Zhang and Telesca (2014) propose a hierarchical model for joint curve clustering and registration. They use a reproducing kernel representation of phase variability for registration.

In this paper, we develop a Bayesian method for simultaneous clustering and registration

of functional data when both arbitrary time distortions and vertical shifts are possible. We model the curves by B-spline basis functions and we approximate the time warping functions by the cumulative sum of realizations from a Dirichlet distribution (following Cheng et al., 2013). The posterior cluster memberships are based on a multinomial distribution. Details of our Markov chain Monte Carlo algorithm are introduced in section 3. Our method of choosing the number of clusters based on a log likelihood is discussed in section 4. Various simulation studies indicate that our method is capable of estimating the time warpings and curve clusters. We apply our method to the well-known Berkeley growth data and compare our result with that of the SACK model by Liu and Yang (2009) and the kCFC model of Chiou and Li (2007). We also apply our method to the cell-cycle data collected by Alter et al. (2000).

2. Model Assumption

In a functional dataset, we assume that there are N objects, on which we take K measurements over time. Given a certain number of repeated measurements, we may model the response trajectory as a function of time using some basis (such as splines) in the context of functional data analysis.

We assume that each observation is composed of a signal function and random error terms, that is,

$$\mathbf{Y} = af(\mathbf{t}) + \boldsymbol{\epsilon},$$

where $a \in \mathbb{R}^+$ is a stretching/shrinking factor (Zhang and Telesca, 2014), $f(\mathbf{t})$ is the set of underlying responses at the vector of time points $\mathbf{t}$, and $\boldsymbol{\epsilon}$ is an i.i.d. $N(0, \sigma^2)$ error vector.

When our observed data must be aligned, we model the effect of the warping function associated with $\mathbf{Y}$ as $\mathbf{Y} = f[h(\mathbf{t})] + \boldsymbol{\epsilon}$, where h is the underlying warping function, and therefore,

$$\mathbf{Y}|\boldsymbol{\beta}, \gamma, \sigma^2, a \sim \text{MVN}(af[h(\mathbf{t})], \sigma^2\mathbf{I}).$$

For the purpose of clustering, we introduce notation for different groups. For a fixed number of clusters C, we use the vector $\boldsymbol{z}_i = (z_{i1}, z_{i2}, \ldots, z_{iC})$ to denote the cluster membership for the i-th observation. Note that only one element of $\boldsymbol{z}_i$ equals 1 and the rest all equal 0. Throughout this paper, we will use B-splines with q basis functions to model the signal curve. It follows that for a K-dimensional observation $\mathbf{Y}$, we have $f(\mathbf{t}) \approx \boldsymbol{\phi}(\mathbf{t})\boldsymbol{\beta}$, where $\boldsymbol{\phi}$ is a $K \times q$ matrix of coefficients of the B-spline basis evaluated at each time point. To be more specific,

$$\boldsymbol{\phi}(\mathbf{t}) = \begin{pmatrix} \phi_1(t_1) & \phi_2(t_1) & \ldots & \phi_q(t_1) \\ \phi_1(t_2) & \phi_2(t_2) & \ldots & \phi_q(t_2) \\ \ldots & \ldots & \ldots & \ldots \\ \phi_1(t_K) & \phi_2(t_K) & \ldots & \phi_q(t_K) \end{pmatrix},$$

where $\phi_i(\cdot)$ denotes the i-th B-spline basis function, and $\boldsymbol{\beta}$ is a vector of B-spline coefficients. We use the same basis functions and assume the same variance σ^2 across all groups. Let $\boldsymbol{\beta}_i$ denote the spline coefficients for the i-th group, $i = 1, 2, \ldots, C$. The discretized mean curve for the i-th cluster is represented as $\boldsymbol{\mu}_i \approx \boldsymbol{\phi}[\gamma(\mathbf{t})]\boldsymbol{\beta}_i$, where $\gamma(\cdot)$ is the discrete approximation of the corresponding warping function h, which will be discussed in the next section.

3. Likelihood and Bayesian Analysis

3.1. Prior Distributions on Parameters

To estimate the warping function h_i for the i-th observation, a discrete approximation generated by a Dirichlet distribution is utilized (Cheng et al., 2015). Without loss of generality, let us assume that the time domain $\mathcal{T} = [0, 1]$. Any general time domain $[T_1, T_2]$ may be converted into $[0, 1]$ by the transformation $g(t) = (t - T_1)/(T_2 - T_1)$. Let $\gamma_{i1}, \gamma_{i2}, \ldots, \gamma_{iM} \sim Dir(\boldsymbol{\alpha})$, where $\boldsymbol{\alpha}$ is a M-vector of positive parameters.

For the Dirichlet distribution, we have $\sum_j \gamma_{ij} = 1$, which suggests that the linear interpolation of the cumulative sum over γ_{ij} can serve as a discrete approximation of the continuous warping h_i. The parameter M controls the smoothness of the approximation. A large M results in a smoother approximation, but more computational burden.

The hyperparameter $\boldsymbol{\alpha}$ can be chosen to affect the "concentration" of the warping functions relative to the 45° reference line, which corresponds to no warping. Small values in $\boldsymbol{\alpha}$ allow more variability in each step of the approximation, and vice versa. Figure 2 shows two sets of discrete warping functions, each with 20 jumps, generated from $Dir(0.8, 0.8, \ldots, 0.8)$, and $Dir(5, 5, \ldots, 5)$, respectively.

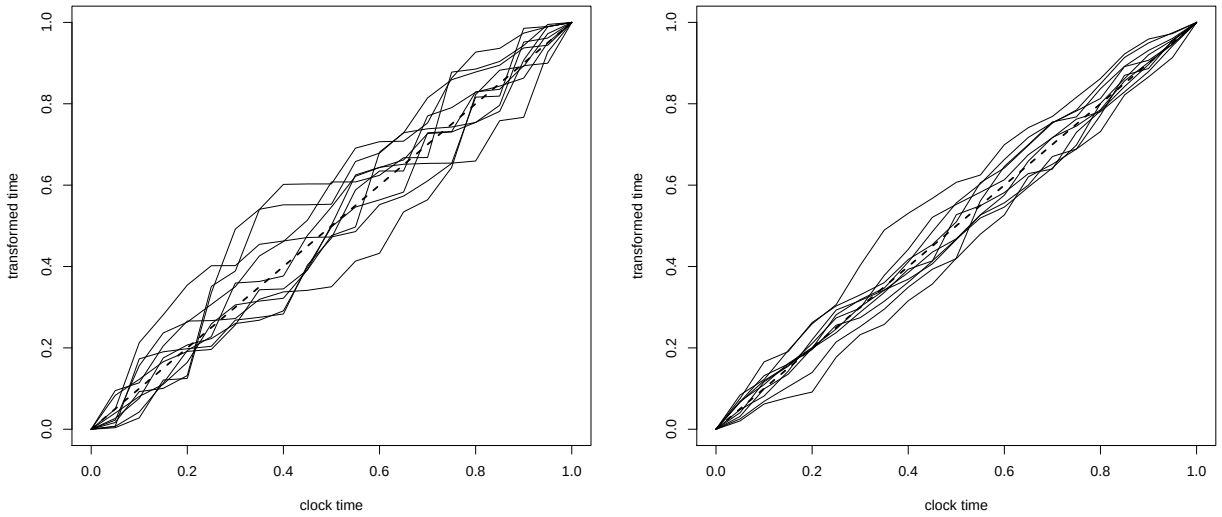

Figure 2: Left: Warping functions from $Dir(0.8, 0.8, \ldots, 0.8)$. Right: Warping functions from $Dir(5, 5, \ldots, 5)$.

If observation i is assigned to cluster j, then the cluster membership indicator $\boldsymbol{z}_i$ is a vector of size C containing a 1 in the j-th position and 0 elsewhere. We model $\boldsymbol{z}_i$ with a multinomial distribution, i.e., $\boldsymbol{z}_i \sim Multi\ (1, (p_1, \ldots, p_C))$, where $p_1, \ldots, p_C$ are the membership probabilities satisfying $\sum_j p_j = 1$. We choose a conjugate Dirichlet prior for those probabilities; i.e., $p_1, \ldots, p_C \sim Dir(\boldsymbol{\eta})$, where $\boldsymbol{\eta}$ is a vector of hyperparameters.

For the i-th cluster, we assume that $\boldsymbol{\beta}_i \sim \text{MVN}(\boldsymbol{\beta}_{0i}, \Gamma)$. It will be seen later that the full conditional distribution of $\boldsymbol{\beta}_i$ is still multivariate normal. We model the precision parameter $\tau = 1/\sigma^2$ with a (conjugate) gamma prior, i.e., $\tau \sim Gamma(\kappa, \theta)$.

For functional observations, one possible source of amplitude variation is composed of vertical shifts among observations in the same cluster. The left panel in Figure 3 shows a set of simulated observations from the same cluster with phase variations; the right panel shows the same observations with additional vertical shifts following *Unif* $(-0.5, 0.5)$. The bold curve is the true signal function generating the observations. Our prior model assumes

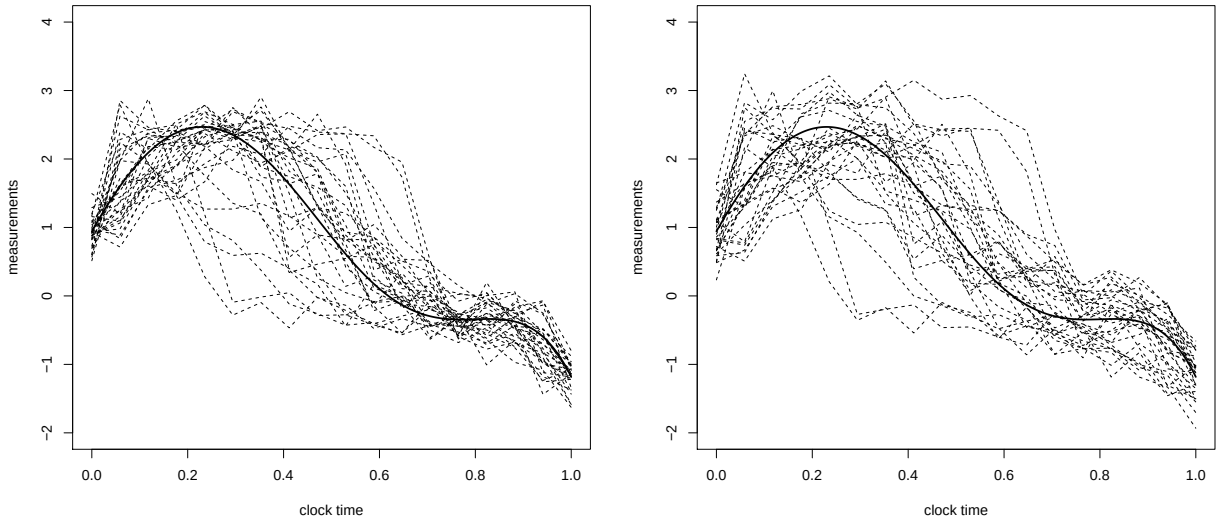

Figure 3: Left: Simulated data with phase variations. Right: Simulated data with additional vertical shifts generated from *Unif* $(-0.5, 0.5)$.

the vertical shift S_i for the i-th observation is *Unif* $(-\phi, \phi)$ for some positive ϕ. On the stretching/shrinking factors a_i, we place independent $N(1, \sigma_a^2)$ priors, $i = 1, 2, \ldots, N$.

3.2. Likelihood and Posterior of the Model

Under the preceding model assumptions, for a vector of measurements taken on the same functional observation, we have

$$\mathbf{Y} = a\boldsymbol{\phi}[\gamma(\boldsymbol{t})]\boldsymbol{\beta} + \mathbf{S} + \boldsymbol{\epsilon},$$

where $\mathbf{S} = S \otimes \mathbf{1}$ ($\otimes$ is the Kronecker product) is a vector of size K containing the same vertical shifts. Hence, the distribution of the i-th observation $\boldsymbol{y}_i$ belonging to a specific cluster in the presence of phase variation is given by

$$\mathbf{Y}_i|\boldsymbol{\beta}, \gamma_i, \boldsymbol{z}_i, \tau, s \sim \text{MVN}\left(a_i\boldsymbol{\phi}[\gamma_i(\mathbf{t})]\prod_{c=1}^{C}\boldsymbol{\beta}_c^{z_{ic}} + \mathbf{s}_i, \tau^{-1}\mathbf{I}\right).$$

With the above prior distributions on the parameters, the joint distribution of the data and parameters is

$$\mathcal{L}(\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_C, \gamma_1, \ldots, \gamma_N, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_N, p_1, \ldots, p_C, \tau, s_1, \ldots, s_N, a_1, \ldots, a_N, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_N)$$

$$
\begin{aligned}
&= \prod_{i=1}^{N} \mathcal{P}(\boldsymbol{y}_i|\boldsymbol{\beta}, \boldsymbol{z}_i, \gamma_i, p_1, \ldots, p_C, \tau, s_1, \ldots, s_N, a_1, \ldots, a_N) \prod_{c=1}^{C} \mathcal{P}(\boldsymbol{\beta}_c|\boldsymbol{\beta}_0^c, \Gamma) \prod_{i=1}^{N} \mathcal{P}(\gamma_i|\boldsymbol{\alpha}) \\
&\quad \prod_{i=1}^{N} \mathcal{P}(\boldsymbol{z}_i|p_1, \ldots, p_C) \mathcal{P}(p_1, \ldots, p_C|\boldsymbol{\eta}) \mathcal{P}(\tau|\kappa, \theta) \prod_{i=1}^{N} \mathcal{P}(s_i|\phi) \prod_{i=1}^{N} \mathcal{P}(a_i|\sigma_a^2) \\
&\propto \prod_{i=1}^{N} \tau^{K/2} \exp\left\{ -\frac{1}{2}\tau \left[\boldsymbol{y}_i - a_i \boldsymbol{\phi}[\gamma_i(\mathbf{t})] \prod_{c=1}^{C} \boldsymbol{\beta}_c^{z_{ic}} - \mathbf{s}_i \right]' \left[\boldsymbol{y}_i - a_i \boldsymbol{\phi}[\gamma_i(\mathbf{t})] \prod_{c=1}^{C} \boldsymbol{\beta}_c^{z_{ic}} - \mathbf{s}_i \right] \right\} \\
&\quad \prod_{c=1}^{C} \exp\left\{ -\frac{1}{2}(\boldsymbol{\beta}_c - \boldsymbol{\beta}_0^c)' \Gamma^{-1} (\boldsymbol{\beta}_c - \boldsymbol{\beta}_0^c) \right\} \prod_{i=1}^{N} \prod_{m=1}^{M} \gamma_{im}^{\alpha_m - 1} \prod_{i=1}^{N} \prod_{c=1}^{C} p_c^{z_{ic}} \prod_{c=1}^{C} p_c^{\eta_c - 1} \\
&\quad \tau^{\kappa+1} \exp\{-\tau\theta\} \prod_{i=1}^{N} \mathbf{1}_{\{-\phi < s_i < \phi\}} \prod_{c=1}^{N} \exp\left\{ -\frac{1}{2}(a_i - 1)^2 \right\} \\
&\propto \tau^{KN/2} \exp\left\{ -\frac{1}{2}\tau \sum_{i=1}^{N} \left\| \boldsymbol{y}_i - a_i \boldsymbol{\phi}[\gamma_i(\mathbf{t})] \prod_{c=1}^{C} \boldsymbol{\beta}_c^{z_{ic}} - \mathbf{s}_i \right\|^2 \right\} \prod_{c=1}^{C} \exp\left\{ -\frac{1}{2}(\boldsymbol{\beta}_c - \boldsymbol{\beta}_0^c)' \Gamma^{-1} (\boldsymbol{\beta}_c - \boldsymbol{\beta}_0^c) \right\} \\
&\quad \prod_{i=1}^{N} \prod_{m=1}^{M} \gamma_{im}^{\alpha_m - 1} \prod_{c=1}^{C} p_c^{\sum_{i=1}^{N} z_{ic} + \eta_c - 1} \tau^{\kappa-1} \exp\{-\tau\theta\} \prod_{i=1}^{N} \mathbf{1}_{\{-\phi < s_i < \phi\}} \exp\left\{ -\frac{1}{2} \sum_{i=1}^{N} (a_i - 1)^2 \right\}.
\end{aligned}
$$

This joint distribution will be used to obtain the relevant full conditional distributions for the MCMC algorithm. The details of the sampling algorithm are given in Appendix A.

From experimentation using various simulated data, we note two concerns: (1) The posterior cluster memberships converge quickly usually after several hundred iterations, and barely change afterwards; (2) the "converged" cluster memberships depend on the initial values of the Markov chain. These phenomena are partially due to the fact that the misclassified observations affect the posterior sampling of coefficients $\boldsymbol{\beta}$, and cluster memberships are in turn influenced by those coefficients in the next iteration.

We need to "force" the individual curves to accept new group membership from time to time to avoid the vicious circle described above. We adjust the sampling algorithm in the following way: In the burn-in stage, every I iterations, p% of the curves in each group switch clusters at random (for practical purposes, we recommend $I = 10$ to 100, $p = 3$ to 15). We make these switches only in the burn-in stage, and thus we use an ordinary MCMC algorithm afterward with the initial values obtained from the burn-in stage. This switch reduces the influence of initial values. Should the switch result in a poorer clustering, we note based on experimentation that the chain can adjust itself and is likely to recover individual classifications of the previous partitions that were correct.

Our proposed method can cluster observations under nonlinear time distortion and vertical shifting and does not require or estimate any template for the purpose of registration.

4. Choosing the Number of Clusters

Determining the number of clusters is a common problem in cluster analysis. A wide variety of solutions have been proposed. The "elbow criterion" examines the percentage of variation explained as a function of the number of clusters, with the number of clusters chosen where when the plot levels off. The variance ratio criterion (Caliński and Harabasz, 1974) chooses the number of clusters which maximizes the ratio of the between-cluster and the within-cluster sum-of-squares. For model-based clustering methods, information criteria such as AIC (Akaike, 1974) and BIC (Schwarz et al., 1978)

are frequently employed as a measure of clustering quality. These information-theoretic approaches are based essentially on the log-likelihood and penalize the number of parameters in the model.

For our method, it is simple to calculate the log-likelihood for a given cluster number C^* at each iteration. Recall that

$$\mathbf{Y}_i|\boldsymbol{\beta}, \gamma_i, \boldsymbol{z}_i, \tau, s_i, a_i \sim \text{MVN}\left(a_i\boldsymbol{\phi}[\gamma_i(\mathbf{t})]\prod_{c=1}^{C}\boldsymbol{\beta}_c^{z_{ic}} + \mathbf{s}_i, \tau^{-1}\mathbf{I}\right).$$

The likelihood is given by

$$\begin{aligned}
& \mathcal{L}(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N|\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{C^*}, \gamma_1, \ldots, \gamma_N, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_N, s_1, \ldots, s_N, a_1, \ldots, a_N) \\
= & \prod_{i=1}^{C^*}\prod_{j=1}^{n_i}(2\pi)^{K/2}|\tau^{-1}\mathbf{I}|^{-1/2}\exp\left\{-1/2\tau||\boldsymbol{y}_j - a_j\boldsymbol{\phi}[\gamma_j(\mathbf{t})]\prod_{c=1}^{C}\boldsymbol{\beta}_c^{z_{jc}} - \mathbf{s}_j)||^2\right\} \\
= & (2\pi)^{-K/2}\prod_{i=1}^{C^*}\prod_{j=1}^{n_i}\tau^{K/2}\exp\left\{-1/2\tau||\boldsymbol{y}_j - a_j\boldsymbol{\phi}[\gamma_j(\mathbf{t})]\prod_{c=1}^{C}\boldsymbol{\beta}_c^{z_{jc}} - \mathbf{s}_j)||^2\right\}.
\end{aligned}$$

The log-likelihood follows as

$$\begin{aligned}
& \log\mathcal{L}(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N|\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{C^*}, \gamma_1, \ldots, \gamma_N, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_N, s_1, \ldots, s_N, a_1, \ldots, a_N) \\
= & \text{ constant} + \frac{KN}{2}\tau - \frac{1}{2}\tau\sum_{i=1}^{C^*}\sum_{j=1}^{n_i}||\boldsymbol{y}_j - a_j\boldsymbol{\phi}[\gamma_j(\mathbf{t})]\prod_{c=1}^{C}\boldsymbol{\beta}_c^{z_{jc}} - \mathbf{s}_j)||^2.
\end{aligned}$$

To be conservative, we start our algorithm with an excessive initial number of clusters (at least 1/4 of the total number of observations) and allow the number of non-empty clusters to decrease across iterations. Such a decrease occurs when at iteration t, based on the last sampled parameter values, no objects are assigned to some cluster in the Metropolis-Hastings cluster membership step.

We apply the following procedure to select the number of clusters during the initial (burn-in) stage of the algorithm, in conjunction with the cluster membership-switching procedure described at the end of Section 4. After this initial stage, we fix the number of clusters and proceed with ordinary MCMC, using only the Gibbs step to assign cluster membership to each observation.

When the total number of non-empty clusters decreases from C^* to C^*-1, we calculate the average log-likelihood for the most recent block of iterations with C^* clusters (denoted by avg $\log\mathcal{L}_{C^*}$) and compare it to the average log-likelihood for the most recent block of iterations with C^*+1 clusters (denoted by avg $\log\mathcal{L}_{C^*+1}$). If avg $\log\mathcal{L}_{C^*} >$ avg $\log\mathcal{L}_{C^*+1}$, we accept the decrease. Otherwise, we reset the cluster membership to the first iteration in the most recent block of iterations where the number of clusters is C^*+1. The pseudo code is given in Appendix B.

If the number of clusters remains constant for a long period of time, it either achieves the optimal number of clusters in terms of average log-likelihood, or the algorithm is trapped at the current number of clusters. Let M_{C^*} be number of consecutive iterations that the Markov chain stays at the current number of clusters. If M_{C^*} is larger than some predetermined threshold, we compare the average log-likelihood of the current block of iterations where $C = C^*$ to the average log-likelihood of the most recent block of iterations with $C = C^*+1$. If the average log-likelihood is smaller for the current C^*, we reset $C = C^*+1$, and reset the cluster membership to the first iteration in the most recent block of iterations where the number of clusters is C^*+1. The pseudo code is given in Appendix B.

5. Simulation Study

To illustrate our algorithm's ability to estimate warping functions and cluster structure, we generate a simulated dataset and apply our method to it.

On the domain $\mathcal{T} = [0, 1]$, we choose 6 B-spline basis functions of order 5 using an equally-spaced knots sequence. We specify 5 clusters, and thus generate 5 sets of B-spline coefficients of size 6 distributed as $\text{MVN}(\mathbf{0}, 2 \times \mathbf{I})$, which are shown in Table 1. We assign 10, 12, 11, 10, and 13 observations (56 total observations) to each cluster, respectively, and generate 56 warping functions with 20 steps distributed as $Dir(\boldsymbol{\alpha} = (1, \ldots, 1))$. We assume that 30 equally spaced measurements on $\mathcal{T}$ are taken from each curve. The simulated warping functions are applied to the clock time and the underlying process times are obtained for each observation. For the i-th observation, we evaluate the B-spline function at its corresponding process times. A set of stretching/shrinking factors of size 56 is generated as independent $N(1, 0.05^2)$ and multiplied to the mean values of the corresponding observations. Finally, we add white noise with $\sigma^2 = 0.01$ to each observation at each time point. A vertical shift generated from $Unif\,(-1, 1)$ is added to each observation. A plot of the simulated dataset is shown in the top left panel of Figure 4.

	β_1	β_2	β_3	β_4	β_5	β_6
coef 1	2.38	0.46	-2.18	0.39	-2.56	1.81
coef 2	0.38	-2.89	-0.65	3.24	-1.38	4.08
coef 3	0.94	2.50	4.06	-2.79	0.59	-1.18
coef 4	-0.48	1.35	-4.88	2.48	-0.65	0.31
coef 5	-1.12	-0.00	0.83	2.62	4.48	0.70

Table 1: 5 sets of coefficients for the B-spline basis functions

To analyze the simulated data, we use a B-spline representation with 9 basis functions of order 6 with equally spaced knots. Our simulation experimentation indicates the clustering results are insensitive to the choice of spline basis having reasonable number and order, which is also noted by Liu and Yang (2009) and James and Sugar (2003). The means $\boldsymbol{\beta}_0$ of the B-spline are taken to be $\mathbf{0}$, and we assume those coefficients are independent with variance $\mathbf{1}$, i.e., $\boldsymbol{\beta}|\boldsymbol{\beta}_0, \Gamma \sim N(\mathbf{0}, \mathbf{I})$. Based on Appendix A, the posterior samples for those coefficients are dominated by the data unless we have very strong prior knowledge. For the hyperparameters, we choose $\kappa = 100, \theta = 1$ for the precision, $\phi = 1$ for the vertical shifts, and $\alpha = 1$ for the warping functions. Following our algorithm for choosing the number of clusters, we start with $C = 30$ clusters having equal prior cluster probabilities.

We perform 20000 iterations, with the first 10000 discarded as burn-in. There are 5859 iterations in all whose number of non-empty clusters is 5, indicating that $C = 5$ is the most appropriate choice for this simulated dataset. To find a good set of starting values, we run another chain with $C = 5$ for 20000 iterations, with the first 10000 discarded as burn-in. We switch 15% of the observations in each cluster every 20 iterations. Finally, a regular MCMC is performed using the initial values obtained from the last step. The correct classification rate (cRate) (Liu and Yang, 2009), defined as the maximum proportion of agreements between estimated and true cluster memberships (among all labeling permutations), is a measure of clustering quality. The cRate of our simulation study is 100%. We compare the result from joint registration and clustering to other existing methods using these simulated data. Of methods involving only clustering, the K-means method (Hothorn and Everitt, 2014), Ward's hierarchical agglomerative method (Hothorn and Everitt, 2014), and a model-based clustering method (Fraley and Raftery, 2002) produce a cRate

of 83.93%, 85.71%, and 89.28%, respectively. To compare our result to a stepwise registration and clustering approach, we apply the registration method of Ramsay and Silverman (Ramsay and Silverman, 2005) implemented with the `register.fd` function in the `fda` package in `R` (Ramsay et al., 2013) to smooth and register the curves. Applied to the resulting registered curves, the cRate of the above three methods are 62.50%,75%, and 82.14%, respectively.

The lower left panel of Figure 4 displays the true signal curves (gray) and our posterior estimated signal curves (black). We use means of the posterior samples having 5 clusters as the point estimates of the B-spline coefficients. The estimated signal curves basically capture the characteristics of the true signal curve. The lower right panel of Figure 4 shows the estimated warping functions.

To test the convergence of the chain, we use the Heidelberg-Welch stationarity test (Heidelberger and Welch, 1981). One advantage of this method is that it does not require multiple chains with different initial values, since our chain starts with the initial values determined by a preliminary run. For our simulation study, the sample for τ passes the test; 85% of the spline coefficient samples pass the test; 85% of the stretching/shrinking factor samples pass the test; 95% of the vertical shift samples pass the test; 93% of the warping function jumps pass the test. Overall, the vast majority of the posterior samples are considered to be drawn from their stationary distributions.

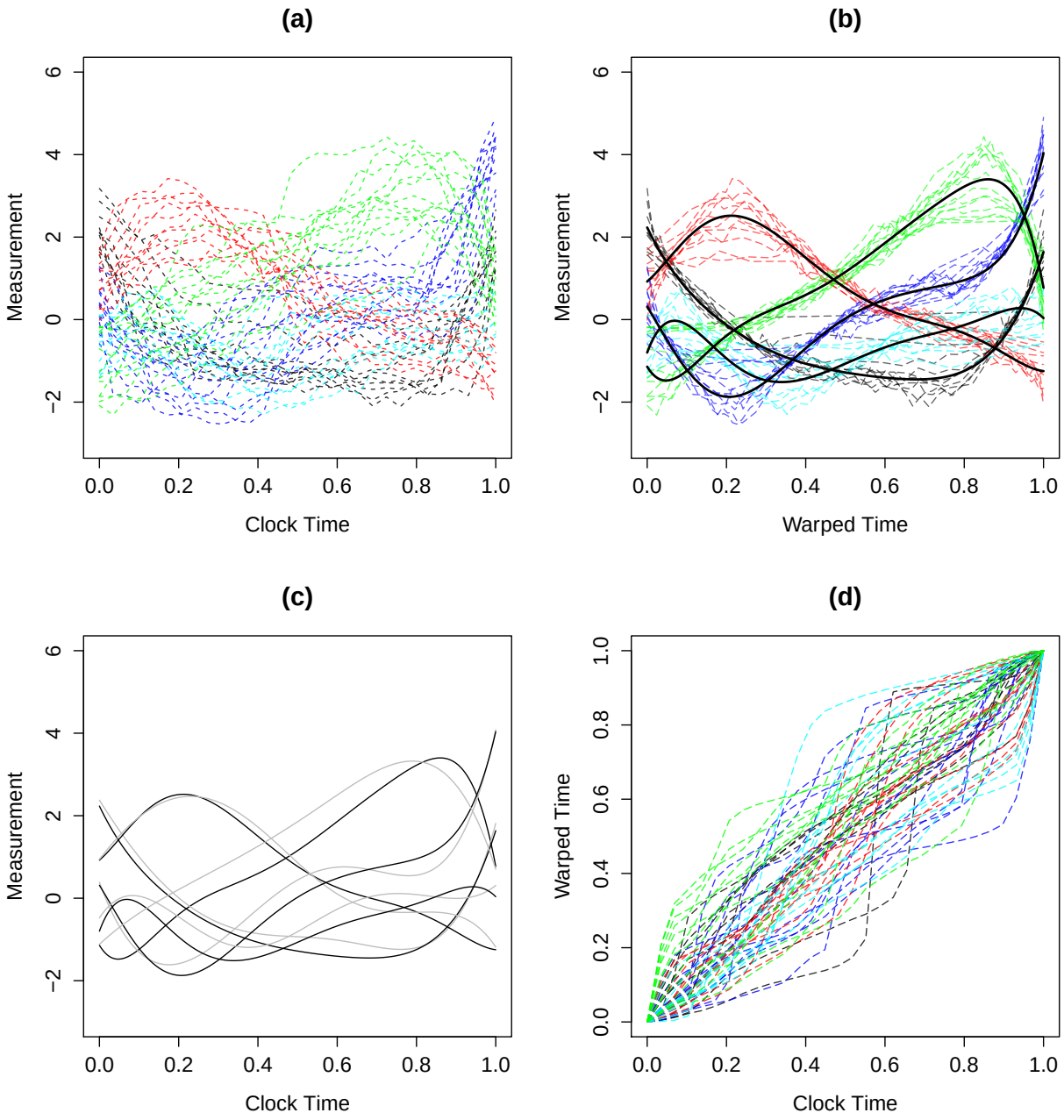

Figure 4: (a) A set of 56 simulated observations with 5 clusters. (b) Simulated data with phase variation removed, with superimposed posterior estimated mean curves (solid black). (c) True mean curves (gray) and estimated mean curves (black). (d) Estimated warping functions for all 5 clusters.

The goals of our study are estimating cluster membership and the warping functions associated with each observation. For a given observation, each step of the discrete warping function is estimated via the mean posterior jump at that step. The phase variation can be removed by applying the estimated warping function to the clock time for each observation. For our simulated dataset, the curves with phase variation removed are shown in the top right panel of Figure 4, from which we see a clear cluster structure.

The user-chosen value of M determines the degree of discretization of the warping function. Our philosophy is to achieve a balance between a reasonable approximation and affordable computational time. As a guide for the choice of M, we proposed the criterion

$$\psi^{M,\alpha} = \sum_{i=1}^{N} \int_0^1 |\gamma_i^{M,\alpha}(t) - t| dt$$

to measure the concentration of the warping functions (as a function of the dimension M and concentration parameter α) around the 45° reference line. If we change M, we need to adjust α simultaneously so that the variabilities among the warping functions remain roughly the same across different choices of M and α. We may obtain a positive real K by specifying a base Dirichlet distribution with $M = M_0$ and $\alpha = \alpha_0$, and then letting $K = \alpha M$.

To inform the choice of M, we run 5 preliminary chains with 5000 iterations on our simulated data. We hold all parameters and hyperparameters constant except M and α, which we vary. We choose $M = 20, \alpha = 1$ as the base distribution and thus $K = 20$. We examined the cases of $M = 5, 10, 20, 30, 40$, and 50. Figure 5 shows a scatter plot of ψ against M. A value of M around the "elbow" of this plot should be sufficiently large to represent well the true nature of the distribution of warpings. We see that values of $M \geq 10$ are acceptable, since the elbow of Figure 5 is at $M = 10$. We still prefer using $M = 20$ due to more precise approximation and a still reasonable computing time. Note that the classification rate (cRate) for $M = 5$ is only 68%, while all other cases have cRate around 95% even for such a preliminary run.

We conduct a sensitivity analysis by examining the specifications of several hyperparameters. We investigate the effect of various choices of α, ϕ, and σ_a^2. We vary the hyperparameters one at a time, separately multiplying each by 10, then by 0.1. The original values for α, ϕ, and σ_a^2 are 1, 1, and 0.05^2, respectively.

Table 2 shows the cRate for different altered choices of hyperparameters. The alteration of α only results in 1 and 3 incorrect curves, respectively. Using a large shift parameter $\phi = 10$ misclassifies 3 curves, while the small shift misclassifies 5 curves. This makes sense since the conditional posterior distribution of ϕ is a truncated normal bounded at $-\phi$ and ϕ. The small choice of σ_a^2 results in a much better cRate.

Based on our simulation study, our method seems to be insensitive to the specification of α. One exception is for data like the Berkeley accelerations that we present in Section 6, for which all the curves are similar and the phase variation contributes significantly to the cluster structure. In such a case, α must be chosen with caution. We would recommend choosing ϕ fairly large rather than small, since a small ϕ may be too restrictive to sample a proper shift. Finally, we would recommend choosing σ_a^2 relatively small when uncertain.

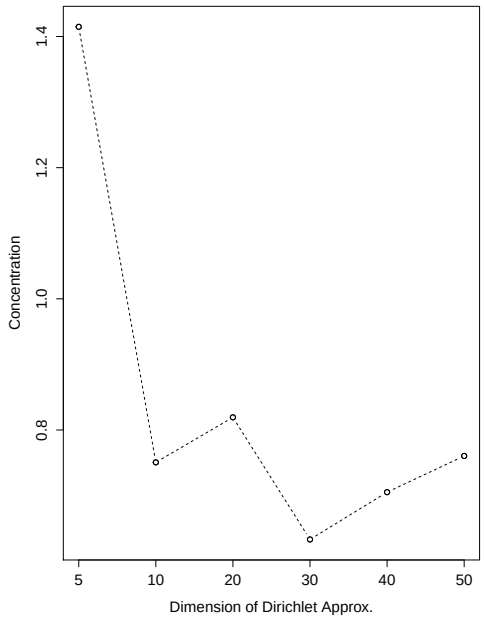

Figure 5: ψ values for different choices of M.

Parameter	Value	cRate
α_1	0.1	94.64%
α_2	10	98.21%
ϕ_1	0.1	91.07%
ϕ_2	10	94.64%
$\sigma^2_{a_1}$	0.025	87.50%
$\sigma^2_{a_2}$	0.00025	100%

Table 2: Sensitivity analysis for simulated data

We perform another simulation study based on the previous setup but with only 10 evenly spaced measurement points. The cRate is 100%, which suggests our method performs well for sparsely sampled data.

6. Real Data Analysis

6.1. Berkeley Growth Curves

The Berkeley growth data (Tuddenham and Snyder, 1953) measured 54 girls and 39 boys at 31 time points from age 1 to age 18. In the literature, this dataset often serves as a benchmark to test clustering accuracy. A monotone smoothing spline (Ramsay and Silverman, 2005) can be applied to the original height data. If we evaluate the corresponding second order derivatives at these 31 measurement time points, there exists obvious phase variation as shown in Figure 6. The left panel shows the acceleration data; the right panel shows the acceleration values without first 5 timepoints excluded due to the bias of the function estimation near the boundary (Cheng et al., 1997). Based on Figure 6, we assume that there are small vertical shifts with $\phi = 1.2$ and the

variation among observations is caused by both phase variation and random error $\boldsymbol{\epsilon}$. We choose $\kappa = 50$ and $\theta = 10$ to accommodate possible amplitude variation, and we choose $\alpha = 4$ for the Dirichlet approximation. We model the signal functions with 8 B-spline basis functions of order 6 defined on a equally spaced knot sequence. The prior means of the spline coefficients are generated as $N(1, 4)$, and the spline coefficients are assumed independent with variance 1. We switch 10 percent of the observations from each cluster every 10 iterations in the burn-in stage. The number of clusters is fixed at 2 throughout the entire MCMC. The prior cluster probabilities are both 0.5 for males and females. We perform 20000 iterations, with the first 10000 discarded as burn-in.

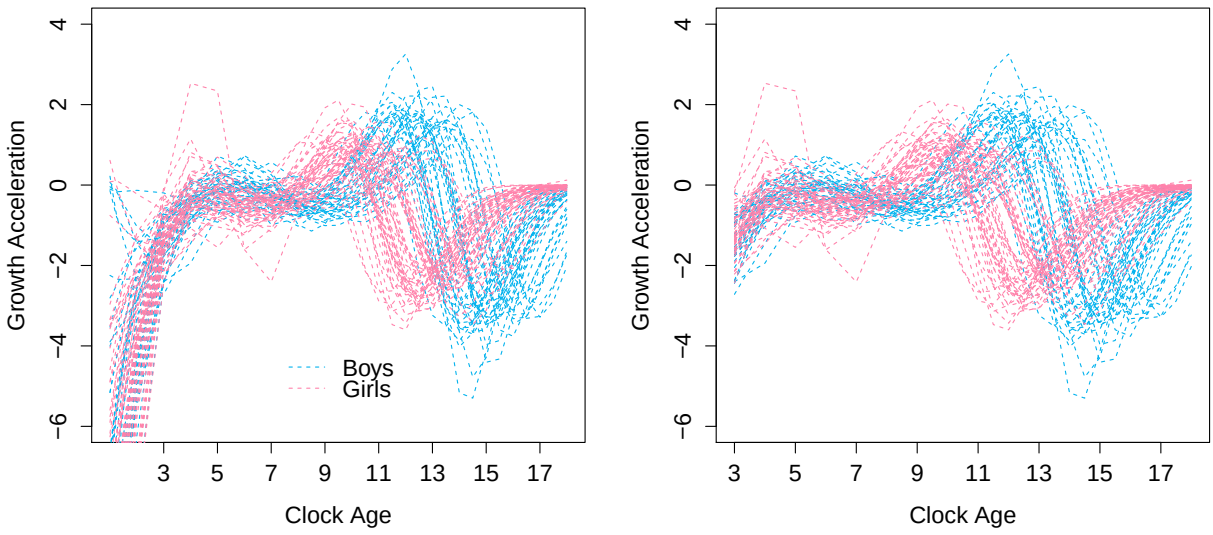

Figure 6: Left: original growth acceleration; Right: growth acceleration without first 5 measures.

The clustering results are shown in Table 3; only 5 females are misclassified to the opposite gender, yielding overall cRate 94.6%. The clustering results are plotted in the second row of Figure 7; the bold solid curves represent those boys who are misclassified as girls, and the bold dashed curves represent the misclassified girls. The right panel shows the curves after registration. For comparison, we apply Ward's hierarchical clustering on the unregistered data (Hothorn and Everitt, 2014), which produces a cRate of 75.26% with 23 girls misclassified as boys. A model-based method (Fraley and Raftery, 2002) produces a 73.08% cRate with 23 girls misclassified as boys. After registration, the Ward's method and the model-based method yields a 63.44% and 68.82% cRate, respectively.

	True cluster	
	Male	Female
Cluster I	37	3
Cluster II	2	51

Table 3: Clustering results for Berkeley acceleration curves.

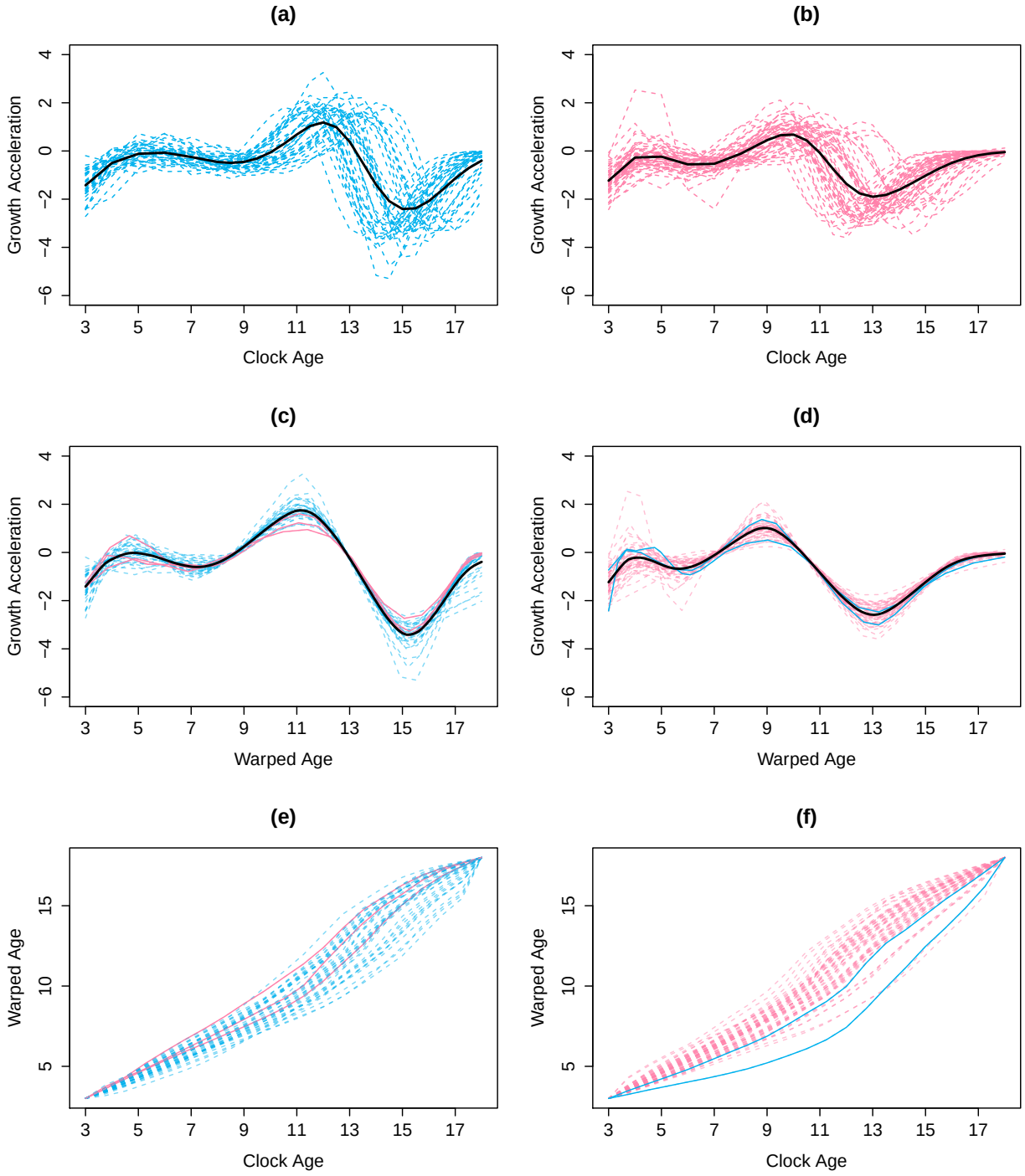

Figure 7: (a)-(b) Unregistered growth acceleration data for 39 boys (blue dashed) and 54 girls (pink dashed) with cross-sectional mean superimposed. (c) Registered cluster 1 with 37 boys (blue dashed) and 3 girls (pink solid). (d) Registered cluster 2 with 51 girls (pink dashed) and 3 boys (blue dashed). (e)-(f) Estimated warping functions for cluster 1 and cluster 2, respectively.

We also apply the proposed method to the original height data and velocity data. For the original height curves, we set $\alpha = 100$ and $\sigma_a^2 = 10^{-3}$, since there is no strong evidence of time distortion and the vertical shifts constitute the majority of the variation. We put a strong precision-related hyperparameter with $\kappa = 5 \times 10^4$ and $\theta = 1$ due to the highly precise height measurements. Our corresponding cRate is 91.4%, while the SACK model (Liu and Yang, 2009) reports a 86% accuracy rate, and KCFC (Chiou and Li, 2007) reports a 93.35% accuracy rate. For the velocity curves, we apply our method with $\alpha = 10, \kappa = 50, \theta = 10, \phi = 5$ and $\sigma_a^2 = 0.1^2$. The cRate produced is 84.9%, while Zhang and Telesca (2014) reported a cRate of 83%.

6.2. Elutriation-Synchronized Cell Cycle

The elutriation dataset, collected by Alter et al. (2000), measures ratios of gene expression levels in log-scale 18 times, at 7-minute intervals. We apply our proposed Bayes method to a subset of 78 gene expressions. According to Spellman et al. (1998), this dataset is classified into five cell-cycle subgroups: M/G_1, G_1, S, S/G_2 and G_1/M. Among these 78 gene expressions, genes 1 to 13, genes 14 to 52, genes 53 to 60, gene 61 to 67, and gene 68 to 78 are classified into these five respective phases. Note that these different cycle phases are based on biologists' beliefs, and therefore are not absolutely true cluster structure. The trajectories of the dataset are shown in the left panel of Figure 8.

First, we apply our method with five clusters to examine whether the clustering results agree with the underlying biological process. Table 4 shows that 39 out of 78 genes are classified in their corresponding cycle phases, highlighted by bold numbers. The gene expressions adjacent to each other should behave similarly due to adjacent-phase correlation. Therefore, we also highlight in italics cells adjacent to the diagonal elements. Note that 67 out of 78 gene expression profiles are clustered on the tridiagonal positions.

Cluster	M/G_1	G_1	S	S/G_2	G_1/M
I (9)	**5**	*1*	0	2	*1*
II (26)	*2*	**19**	*1*	3	1
III (24)	1	*15*	**7**	*1*	0
IV (5)	0	4	0	0	*1*
V(14)	*5*	0	0	*1*	**8**
total	13	39	8	7	11

Table 4: Clustering results for cell cycle when $C = 5$

We next allow the algorithm to choose the number of clusters, initially using 20 clusters. The mode of the number of non-empty clusters is 4, indicating 4 clusters. The clustered gene expression profiles are shown in Figure 8 (right panel). The aligned curves show a clear cluster structure, and all curves in the same cluster display roughly the same pattern.

Cluster	M/G_1	G_1	S	S/G_2	G_1/M
I (9)	5	1	0	2	1
II (38)	0	27	8	2	1
III (13)	1	11	0	1	0
IV (18)	7	0	0	2	9
total	13	39	8	7	11

Table 5: Clustering results for cell cycle when $C = 4$

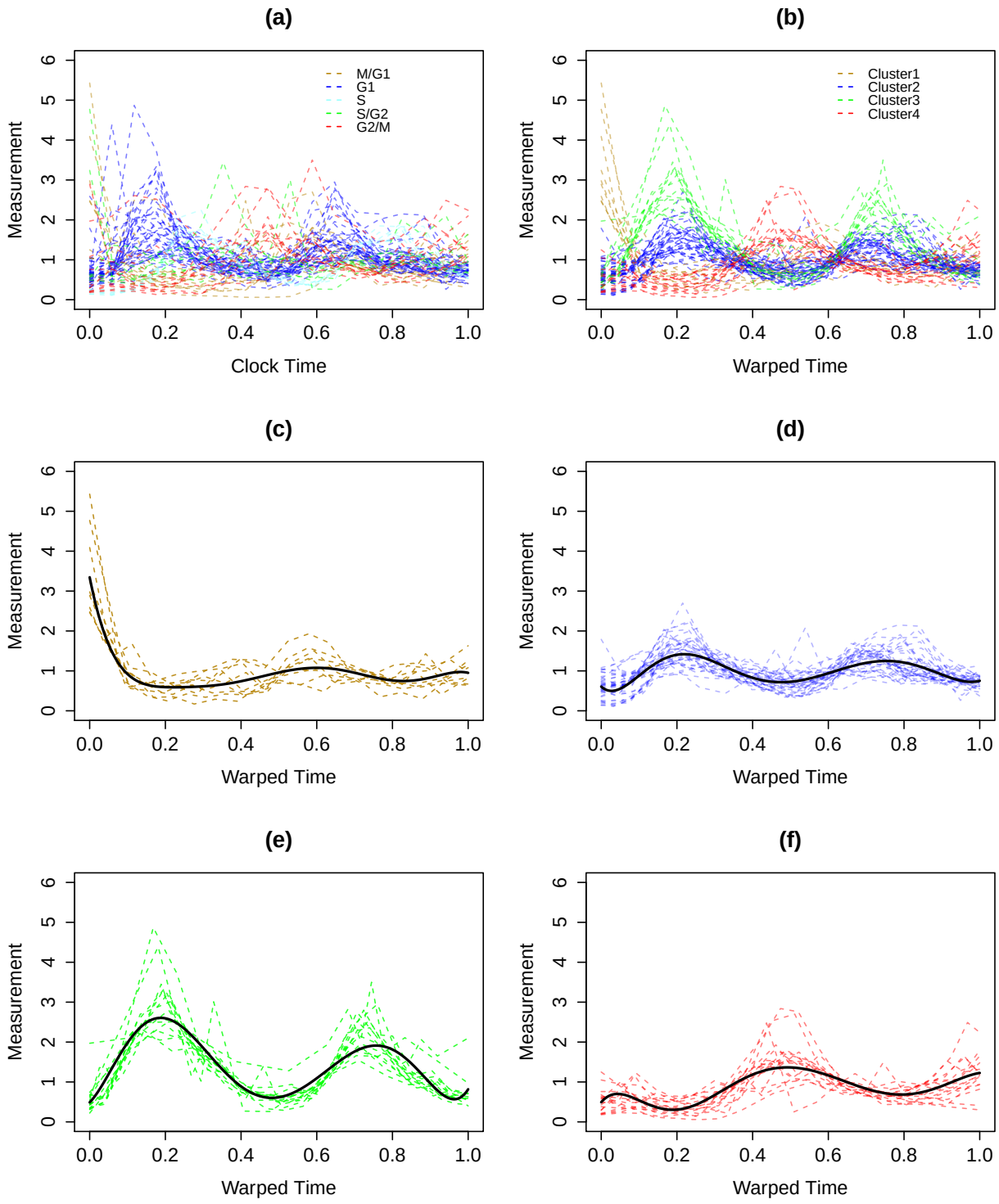

Figure 8: (a) Raw gene expression with cluster structure determined by the biologists. (b) Registered curves with 4 clusters. (c)-(f) Registered four clusters with their estimated mean curves superimposed.

7. Discussion

We have developed a Bayesian clustering method for functional observations that works especially well for data having phase variations. If one believes the phase variations are important characteristics in distinguishing different clusters or that there is no phase variation, one may specify large values of $\boldsymbol{\alpha}$ to discourage the warping functions from departing from a 45° straight line. In this case, our method approximates a Bayesian clustering of functional data without registration.

We demonstrate our algorithm's ability to capture cluster structure and estimate warping functions through simulation studies and real data analyses. Based on our simulation, we observe that one should pick hyperparameters $\boldsymbol{\alpha}$ carefully when phase variations contribute significantly to the clustering structure. We recommend large ϕ and small σ_a^2 when uncertain.

By using the Dirichlet warping approach, our method allows fairly arbitrary warping functions and places no assumptions on the vertical separation among clusters. Thus, the scope of application of our method may exceed that of existing methods, which make more restrictive assumptions. Our simultaneous registration and clustering approach simplifies the analysis procedure and should benefit researchers who cluster functional data.

Appendix A. Sampling Algorithm

Due to the complexity of the proposed model, an analytical posterior derivation is intractable, so our inference is based on MCMC sampling of the posterior distribution. At iteration t, the MCMC algorithm is as follows:

- **Gibbs Sampling for Cluster Membership $\boldsymbol{z}_i$**
  The full conditional distribution of $\boldsymbol{z}_i$ is

$$\mathcal{P}(\boldsymbol{z}_i|\text{rest}) \propto \exp\left\{-\frac{1}{2}\tau^{[t-1]}\left\|\boldsymbol{y}_i - a_i^{[t-1]}\boldsymbol{\phi}[\gamma_i^{[t-1]}(\mathbf{t})]\prod_{c=1}^{C}(\boldsymbol{\beta}_c^{[t-1]})^{z_{ic}} - \mathbf{s}_i^{[t-1]}\right\|^2\right\}\prod_{c=1}^{C}(p_c^{[t-1]})^{z_{ic}}.$$

  The cluster membership indicator vector is discrete and follows a multinomial distribution. The probability of belonging to the j-th cluster is proportional to

$$\exp\left\{-\frac{1}{2}\tau^{[t-1]}\left\|\boldsymbol{y}_i - a_i^{[t-1]}\boldsymbol{\phi}[\gamma_i^{[t-1]}(\mathbf{t})]\ \boldsymbol{\beta}_j^{[t-1]} - \mathbf{s}_i^{[t-1]}\right\|^2\right\}p_j^{[t-1]}.$$

  Let us denote the above quantity by q_j. We have

$$\boldsymbol{z}_i|\text{rest} \sim multi\left(\frac{q_1}{\sum_j q_j}, \ldots, \frac{q_C}{\sum_j q_j}\right).$$

- **Gibbs Sampling for Cluster Probabilities $p_1, \ldots, p_C$**
  After updating the cluster membership, the full conditional distribution of the probabilities is

$$\begin{aligned}\mathcal{P}(p_1, \ldots, p_C|\text{rest}) &\propto \prod_{i=1}^{N}\prod_{c=1}^{C} p_c^{z_{ic}^{[t]}}\prod_{c=1}^{C} p_c^{\eta_c-1} \\ &\propto \prod_{c=1}^{C} p_c^{\sum_{i=1}^{N} z_{ic}^{[t]}+\eta_c-1}.\end{aligned}$$

It follows that

$$p_1, \ldots, p_C | \text{rest} \sim Dir\left(\sum_{i=1}^{N} z_{i1}^{[t]} + \eta_1, \ldots, \sum_{i=1}^{N} z_{iC}^{[t]} + \eta_C\right).$$

- **Metropolis-Hastings Algorithm for Sampling Warping γ_i**
  We update $\gamma_{i1}, \ldots, \gamma_{iM-1}$. The two endpoints satisfy the conditions $\gamma_{i0} = 0$, and $\gamma_{iM} = 1 - \sum_{j=1}^{M-1} \gamma_{ij}$, because of the constraints of the warping function, and hence are not involved in the updating procedure. After updating the $\boldsymbol{z}_i$, we propose a value of γ_{ij}^* from a truncated normal with mean $\gamma_{ij}^{[t-1]}$ and variance σ_γ^2 on $[0, \gamma_{iM} + \gamma_{ij}]$ to guarantee a positive γ_{ij}^* and γ_{iM}^*. We accept the proposed value with probability

$$\lambda = \min\left\{1, \frac{\exp\left\{-\frac{1}{2}\tau^{[t-1]}\left\|\boldsymbol{y}_i - a_i^{[t-1]}\boldsymbol{\phi}[\gamma_i^{*(j)}(\mathbf{t})]\prod_{c=1}^{C}\boldsymbol{\beta}_c^{[t-1]z_{ic}^{[t]}} - \mathbf{s}_i^{[t-1]}\right\|^2\right\}(\gamma_{ij}^*)^{\alpha_j-1}(\gamma_{iM}^*)^{\alpha_M-1}\left[\Phi\left(\frac{r_{ij}^{[t]}-\gamma_{ij}^*}{\sigma_\gamma}\right) - \Phi\left(\frac{-\gamma_{ij}^*}{\sigma_\gamma}\right)\right]}{\exp\left\{-\frac{1}{2}\tau^{[t-1]}\left\|\boldsymbol{y}_i - a_i^{[t-1]}\boldsymbol{\phi}[\gamma_i^{(j-1)}(\mathbf{t})]\prod_{c=1}^{C}\boldsymbol{\beta}_c^{[t-1]z_{ic}^{[t]}} - \mathbf{s}_i^{[t-1]}\right\|^2\right\}(\gamma_{ij}^{[t-1]})^{\alpha_j-1}(\gamma_{iM}^{[t-1]})^{\alpha_M-1}\left[\Phi\left(\frac{r_{ij}^{[t]}-\gamma_{ij}^{[t-1]}}{\sigma_\gamma}\right) - \Phi\left(\frac{-\gamma_{ij}^{[t-1]}}{\sigma_\gamma}\right)\right]}\right\},$$

  where $\gamma_i^{(j)}$ is the warping function with the jump updated through the j-th element, and Φ is the standard normal CDF.

- **Gibbs Sampling for Spline Coefficients $\boldsymbol{\beta}_k$**
  After updating the γ_i's and $\boldsymbol{z}_i$'s, we use a superscript as the updated membership indicator. For example, $\boldsymbol{y}_i^{(k)}$ signifies that we classify observation $\boldsymbol{y}_i$ into group k. Furthermore, let $n_k^{[t]}$ denote the size of group k at the current iteration. The full conditional of $\boldsymbol{\beta}_k$ is given by

$$\begin{aligned}
&\mathcal{P}(\boldsymbol{\beta}_k|\text{rest}) \\
&\propto \exp\left\{-\frac{1}{2}\tau^{[t-1]}\sum_{l=1}^{n_k^{[t]}}\left\|\boldsymbol{y}_l^{(k)} - a_i^{[t-1]}\boldsymbol{\phi}[\gamma_l^{[t]}(\mathbf{t})]\boldsymbol{\beta}_k - \mathbf{s}_l^{[t-1]}\right\|^2\right\}\exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_k - \boldsymbol{\beta}_{0k})'\Gamma^{-1}(\boldsymbol{\beta}_k - \boldsymbol{\beta}_{0k})\right\} \\
&\propto \exp\left\{-\frac{1}{2}\boldsymbol{\beta}_k'\underbrace{\left(\tau^{[t-1]}\sum_{l=1}^{n_k^{[t]}}\left[\left(a_l^{[t-1]}\right)^2\boldsymbol{\phi}'[\gamma_l^{[t]}(\mathbf{t})]\boldsymbol{\phi}[\gamma_l^{[t]}(\mathbf{t})]\right] + \Gamma^{-1}\right)}_{\text{call it } \mathbf{A}}\boldsymbol{\beta}_k - \right. \\
&\qquad\qquad \left. \boldsymbol{\beta}_k'\underbrace{\left(\tau^{[t-1]}\sum_{l=1}^{n_k^{[t]}}a_l^{[t-1]}\boldsymbol{\phi}'[\gamma_l^{[t]}(\mathbf{t})](\boldsymbol{y}_l^{(k)} - \mathbf{s}_l^{[t-1]}) + \Gamma^{-1}\boldsymbol{\beta}_{0k}\right)}_{\text{call it } \mathbf{C}}\right\} \\
&\propto \exp\left\{-\frac{1}{2}(\boldsymbol{\beta}_k - \mathbf{A}^{-1}\mathbf{C})'\mathbf{A}(\boldsymbol{\beta}_k - \mathbf{A}^{-1}\mathbf{C})\right\}.
\end{aligned}$$

  Therefore,

$$\boldsymbol{\beta}_k|\text{rest} \sim \text{MVN}(\mathbf{A}^{-1}\mathbf{C}, \mathbf{A}^{-1}).$$

- **Gibbs Sampling for Precision τ**

After updating the γ_i's, $\boldsymbol{z}_i$, and $\boldsymbol{\beta}_k$'s, the full conditional distribution of τ is given by

$$\begin{aligned}\mathcal{P}(\tau|\text{rest}) &\propto \tau^{KN/2}\exp\left\{-\frac{1}{2}\tau\sum_{i=1}^{N}\left\|\boldsymbol{y}_i-\prod_{c=1}^{C}\left[a_i^{[t-1]}\boldsymbol{\phi}(\gamma_i^{[t]}(\mathbf{t}))\boldsymbol{\beta}_c^{[t]}\right]^{z_{ic}^{[t]}}-\mathbf{s}_i^{[t-1]}\right\|^2\right\}\tau^{\kappa-1}\exp\{-\tau\theta\}\\ &\propto \tau^{KN/2+\kappa-1}\exp\left\{-\tau\left(\frac{1}{2}\sum_{i=1}^{N}\left\|\boldsymbol{y}_i-\prod_{c=1}^{C}\left[a_i^{[t-1]}\boldsymbol{\phi}(\gamma_i^{[t]}(\mathbf{t}))\boldsymbol{\beta}_c^{[t]}\right]^{z_{ic}^{[t]}}-\mathbf{s}_i^{[t-1]}\right\|^2+\theta\right)\right\}.\end{aligned}$$

It follows that

$$\tau|\text{rest}\sim Gamma\left(KN/2+\kappa,\frac{1}{2}\sum_{i=1}^{N}\left\|\boldsymbol{y}_i-\prod_{c=1}^{C}\left[a_i^{[t-1]}\boldsymbol{\phi}(\gamma_i^{[t]}(\mathbf{t}))\boldsymbol{\beta}_c^{[t]}\right]^{z_{ic}^{[t]}}-\mathbf{s}_i^{[t-1]}\right\|^2+\theta\right).$$

- **Gibbs Sampling for Vertical Shift** S_i
  After updating the γ_i's, $\boldsymbol{z}_i$, $\boldsymbol{\beta}_k$'s, and τ, the full conditional distribution of S_i is given by

$$\begin{aligned}&\mathcal{P}(s_i|\text{rest})\\ &\propto \exp\left\{-\frac{1}{2}\tau^{[t]}\left\|\boldsymbol{y}_i-\prod_{c=1}^{C}\left[a_i^{[t-1]}\boldsymbol{\phi}(\gamma_i^{[t]}(\mathbf{t}))\boldsymbol{\beta}_c^{[t]}\right]^{z_{ic}^{[t]}}-\mathbf{s}_i\right\|^2\right\}\mathbf{1}_{\{-\phi<s_i<\phi\}}.\end{aligned}$$

  To simplify the notation, let us define d_l as the l-th element of the vector $\boldsymbol{y}_i-\prod_{c=1}^{C}\left[a_i^{[t-1]}\boldsymbol{\phi}(\gamma_i^{[t]}(\mathbf{t}))\boldsymbol{\beta}_c^{[t]}\right]^{z_{ic}^{[t]}}$. The posterior then is

$$\begin{aligned}\mathcal{P}(s_i|\text{rest}) &\propto \exp\left\{-\frac{1}{2}\tau^{[t]}\sum_{l=1}^{K}(s_i-d_l)^2\right\}\mathbf{1}_{\{-\phi<s_i<\phi\}}\\ &\propto \exp\left\{-\frac{1}{2}\tau^{[t]}\sum_{l=1}^{K}(s_i^2-d_l s_i)^2\right\}\mathbf{1}_{\{-\phi<s_i<\phi\}}\\ &\propto \exp\left\{-\frac{1}{2}\tau^{[t]}K(s_i-\sum_{l=1}^{K}d_l/K)^2\right\}\mathbf{1}_{\{-\phi<s_i<\phi\}}\end{aligned}$$

  The normal kernel indicates that the posterior distribution of the vertical shift S_i is a truncated normal with mean $\sum_{l=1}^{K}d_l/K$, and variance $1/(\tau^{[t]}K)$, i.e.,

$$S_i|\text{rest}\sim N\left(\frac{\sum_{l=1}^{K}d_l}{K},\frac{1}{\tau^{[t]}K}\right)\mathbf{1}_{\{-\phi<s_i<\phi\}}.$$

  Note that for a point estimate of these shifts, we simply require $\sum_i s_i=0$ to ensure identifiability.

- **Gibbs Sampling for Stretching/Shrinking Factor** a_i
  After updating the γ_i's, $\boldsymbol{z}_i$, $\boldsymbol{\beta}_k$'s, τ, and s_i's, the full conditional distribution of a_i is given by

$$\mathcal{P}(a_i|\text{ rest})$$

$$\propto \quad \exp\left\{-\frac{1}{2}\tau^{[t]}\left\|\boldsymbol{y}_i - \prod_{c=1}^{C}\left[a_i\boldsymbol{\phi}(\gamma_i^{[t]}(\mathbf{t}))\boldsymbol{\beta}_c^{[t]}\right]^{z_{ic}^{[t]}} - \mathbf{s}_i^{[t]}\right\|^2\right\}\exp\left\{-\frac{1}{2\sigma_a^2}(a_i-1)^2\right\}.$$

For economy of notation, let us denote the l-th element of $\prod_{c=1}^{C}[\boldsymbol{\phi}(\gamma_i^{[t]}(\mathbf{t}))\boldsymbol{\beta}_c^{[t]}]^{z_{ic}^{[t]}}$ and $\boldsymbol{y}_i$ by $\mu_{il}^{[t]}$ and y_{il}, respectively. The posterior becomes

$$\begin{aligned}
&\mathcal{P}(a_i|\text{ rest})\\
\propto\quad &\exp\left\{-\frac{1}{2}\tau^{[t]}\left[\sum_{l=1}^{K}a_i^2(\mu_{il}^{[t]})^2 - \sum_{l=1}^{K}2a_i\mu_{il}^{[t]}(y_{il}-s_i^{[t]})\right]\right\}\exp\left\{-\frac{1}{2\sigma_a^2}a_i^2 + \frac{1}{\sigma_a^2}a_i\right\}\\
\propto\quad &\exp\left\{-\frac{1}{2}\left[\frac{1}{\sigma_a^2}+\tau^{[t]}\sum_{l=1}^{K}(\mu_{il}^{[t]})^2\right]a_i^2 + \left[\tau^{[t]}\sum_{l=1}^{K}\mu_{il}^{[t]}(y_{il}-s_i^{[t]}) + \frac{1}{\sigma_a^2}\right]a_i\right\}.
\end{aligned}$$

By completing the square, we have

$$a_i|\text{ rest} \sim N\left(\frac{\tau^{[t]}\sum_{l=1}^{K}\mu_{il}^{[t]}(y_{il}-s_i^{[t]}) + 1/\sigma_a^2}{1/\sigma_a^2+\tau^{[t]}\sum_{l=1}^{K}(\mu_{il}^{[t]})^2}, \frac{1}{1/\sigma_a^2+\tau^{[t]}\sum_{l=1}^{K}(\mu_{il}^{[t]})^2}\right).$$

Appendix B. Choosing the Number of Clusters C

The following algorithm determines whether we accept the decrease of the number of clusters by 1. Let the number of non-empty clusters at iteration t be denoted by $C^{[t]}$.

At iteration t, $C^{[t-1]} = C^*$ and $C^{[t]} = C^* - 1$;
if *avg* $\log\mathcal{L}_{C^*} >$ *avg* $\log\mathcal{L}_{C^*+1}$ **then**
  accept $C^{[t]} = C^* - 1$;
else
  reset $C^{[t]} = C^* + 1$;
  reset the cluster membership to the first iteration in the most recent block of iterations where the number of clusters is $C^* + 1$;
end

Algorithm 1: Accept or reject the change of the number of clusters.

The following algorithm determines whether we increase the number of clusters by 1 if the Markov chain stays at the same number of non-zero clusters for a long period.

At iteration t, $C^{[t]} = C^*$ and $M_{C^*} > M_0$;
if *avg* $\log\mathcal{L}_{C^*} >$ *avg* $\log\mathcal{L}_{C^*+1}$ **then**
  keep $C^{[t]} = C^*$;
else
  reset $C^{[t]} = C^* + 1$;
  reset the cluster membership to the first iteration in the most recent block of iterations where the number of clusters is $C^* + 1$;
end

Algorithm 2: Accept or reject the change the number of clusters when $M_{C^*} > M_0$.

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Alter, O., Brown, P. O., and Botstein, D. (2000). Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences*, 97(18):10101–10106.

Caliński, T. and Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics: Theory and Methods*, 3(1):1–27.

Cheng, M.-Y., Fan, J., Marron, J. S., et al. (1997). On automatic boundary corrections. *The Annals of Statistics*, 25(4):1691–1708.

Cheng, W., Dryden, I. L., and Huang, X. (2015). Bayesian registration of functions and curves. *In press: Bayesian Analysis doi: 10.1214/15-BA957*.

Chiou, J.-M. and Li, P.-L. (2007). Functional clustering and identifying substructures of longitudinal data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):679–699.

De Boor, C. (2001). *A Practical Guide to Splines*. New York: Springer-Verlag.

Everitt, B., Landau, S., Leese, M., and Stahl, D. (2011). *Cluster Analysis*. Chichester, UK: Wiley.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631.

Heidelberger, P. and Welch, P. D. (1981). A spectral method for confidence interval generation and run length control in simulations. *Communications of the ACM*, 24(4):233–245.

Hothorn, T. and Everitt, B. S. (2014). *A handbook of statistical analyses using R*. CRC press.

James, G. M. and Sugar, C. A. (2003). Clustering for sparsely sampled functional data. *Journal of the American Statistical Association*, 98(462):397–408.

Liu, X. and Yang, M. C. (2009). Simultaneous curve registration and clustering for functional data. *Computational Statistics and Data Analysis*, 53(4):1361–1376.

Luan, Y. and Li, H. (2003). Clustering of time-course gene expression data using a mixed-effects model with B-splines. *Bioinformatics*, 19(4):474–482.

Ramsay, J., Wickham, H., and Ramsay, M. J. (2013). Package fda.

Ramsay, J. O. and Silverman, B. W. (2005). *Functional Data Analysis*. Springer: New York.

Sangalli, L. M., Secchi, P., Vantini, S., and Vitelli, V. (2010). K-mean alignment for curve clustering. *Computational Statistics and Data Analysis*, 54(5):1219–1233.

Schwarz, G. et al. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M. B., Brown, P. O., Botstein, D., and Futcher, B. (1998). Comprehensive identification of cell cycle–regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell*, 9(12):3273–3297.

Tang, R. and Müller, H.-G. (2009). Time-synchronized clustering of gene expression trajectories. *Biostatistics*, 10(1):32–45.

Tuddenham, R. D. and Snyder, M. M. (1953). Physical growth of California boys and girls from birth to eighteen years. *University of California Publications in Child Development*, 1(2):183–364.

Zhang, Y. and Telesca, D. (2014). Joint clustering and registration of functional data. *arXiv:1403.7134*.