# Simultaneous Registration and Modelling for Multi-dimensional Functional Data

Pengcheng Zeng

Thesis submitted for the degree of
Doctor of Philosophy


*School of Mathematics, Statistics & Physics*
*Newcastle University*
*Newcastle upon Tyne*
*United Kingdom*

April 2018

**Acknowledgements**

First of all, I would like to thank my supervisor Dr. Jian Qing Shi, who assisted me a lot in the past three and a half years. His great guidance for this new project gave me loads of confidence and markedly sped up the progress. Because of his consistent patience and long-term support, I had no worries about the errors or failures while doing the research. I have learnt a lot from him, including not just specific knowledge and problem-solving skills, but also how to become an eligible researcher in the future.

Secondly, I want to thank Dr. Won-Seok Kim for providing data and advice on medical background. He offered us enough video clips for data tracking and gave us a lot of suggestions on the part of data segmentation. Without his help, it could be difficult to finish the data acquisition.

Thirdly, I really appreciate that my school fully funded me during my research period. Without this funding, it would be very hard for me to complete my PhD programme successfully. I would also thank all the members of staff and colleagues in the school for their help in every aspect of my life. They provided a positive academic environment and very colorful social life zone for my research work.

Lastly, I would give many thanks to my parents and relatives in China and my friends in Newcastle. Their encouragement, support and company always gave me enough strength and confidence to face any challenges coming from both research work and daily life. This also played a key role in the completion of this research programme.

**Abstract**

Functional data analysis (FDA) has many applications in almost every branch of science, such as engineering, medicine and biology. It aims to cope with the analysis of data in the form of images, curves and shapes. In this thesis, we study the 2D trajectories of hyoid bone movement from X-ray image. Those curves are seen as the observations of multi-dimensional functional data. We firstly develop an all-in-one platform for the data acquisition and preprocessing. However, analyzing the data arises a lot of challenges. In this thesis, we provide solutions to solve some of those challenging problems.

We propose one new registration method for handling those raw 2D curves. It basically integrates Generalized Procrusts analysis and self-modelling registration method (*GPSM*). However, the application reveals that the classification followed by registration does not work well. Therefore, we propose two-stage functional models for joint curve registration and classification (*JCRC*). In the first stage, we use a functional logistic regression model where the aligned curves are estimated from the second stage. The latter uses a nonlinear warping function while modelling the 2D curves, i.e. resolving the misaligned problem and modelling problem simultaneously. This two-stage model takes into account both the scalar variables and the multi-dimensional functional data. For the functional data clustering, we propose mixtures of Gaussian process functional regression with time warping and logistic allocation model, allowing the use of both types of variables and also allowing simultaneous registration and clustering (*SRC*). A two-level model is introduced. For the data collected from subjects in different groups, a Gaussian process functional regression model is used as the first level model; an allocation model depending on scalar variables is used as the second level model providing further information over the groups. Those three methods, i.e., *GPSM*, *JCRC* and *SRC* are all examined on both simulated data and real data.

*Keywords*: Functional data analysis, Registration, Curve classification, Curve clustering, Gaussian process functional regression model, Allocation model.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Aim of the research

Dysphagia is defined as a subjective sensation of difficulty or abnormality of swallowing. Oropharyngeal dysphagia is characterized by difficulty initiating a swallow, which is caused by various diseases such as stroke, Parkinson's disease, neuromuscular diseases, head and neck cancer (AbdelJalil et al., 2015). The prevalence of dysphagia is expected to increase taking into consideration an aging population and the increase of the incidence of diseases related with dysphagia (Feiginl et al., 2003; Dorsey et al., 2007). Videofluoroscopic swallow study (VFSS) is considered to be a gold standard tool in the assessment of patients with dysphagia.

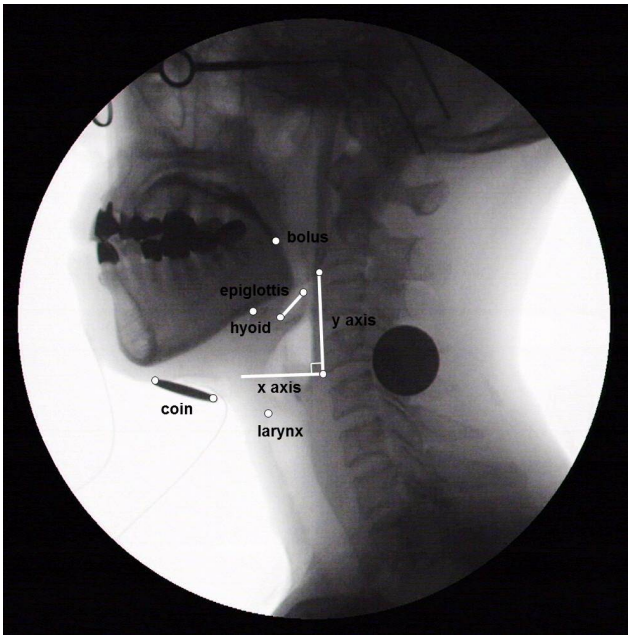



Figure 1.1: Example of one frame from a video clip.

With the image data inside those video clips from VFSS, this research programme aims to model patients' recovery level by analyzing the trajectories of several bones during swallowing, like the movement of hyoid bone and larynx shown in Figure 1.1, and patients' other related information, such as age, gender and smoking status. In this thesis, we focus on the trajectories of hyoid bone, which are considered as the observations of multi-dimensional functional data. Those 'related information' are the observations of scalar variables.

This thesis consists of three parts. The first part develops a platform to obtain the motion data of hyoid bone from the X-ray video clips. We design desirable and efficient algorithms to automatically or semi-automatically track the bone's movement during swallowing. The second part is concerned with preprocessing techniques, such as smoothing, calibration, segmentation and registration for the raw trajectories before modeling. The third part is about the classification and clustering for those 2D curves. We propose some new approaches in the modeling part, that are capable of registering and modelling the multi-dimensional functional data at the same time and allowing the use of both scalar and functional variables.

## 1.2 Background of VFSS and data tracking

Most of the research of VFSS in the clinical setting is qualitative or semi-quantitative and depends on subjective decision by an interpreter. Some clinicians or researchers are using temporal parameters (e.g. oral transit time, pharyngeal transit time) or kinematic parameters from motion analysis to classify the dysphagia, to predict the prognosis or to assess the treatment effect (Pai et al., 2008; Nam et al., 2013; Seo et al., 2011; Molfenter and Steele, 2014). The hyoid bone is the most commonly selected in kinematic analysis. Both displacement and velocity of the hyoid bone excursion are associated with swallowing function and dysphagia. The maximum excursion and peak velocity of the hyoid bone motion are associated with bolus bolume (Nagy et al., 2014) and changed with aging (Kang et al., 2010). Hyoid bone anterior displacement is reduced in patients with myopathy and irradiated nasopharyngeal carcinoma (Pai et al., 2008; Wang et al., 2010). Laryngeal elevation velocity was an independent predictor of aspiration in patients with acute ischemic stroke (Zhang et al., 2016). Therefore, the parameters from hyoid bone motion analysis provide some meaningful solutions in research or clinical practices. However, the classical manual tracking method is labor intensive and impractical in real clinical practice (Steele et al., 2011; Ludlow et al., 2007).

To overcome this limitation, researchers have tried to develop software to track the hyoid bone and to get the trajectory automatically. Kellen et al. (2010) have reported their computer-assisted assessment of hyoid bone motion and found a high correlation between

automatic tracking and manual tracking. This software can reduce the burdens for VFSS motion analysis and make further quantitative analysis practically possible. However, one of the limitations of the existing software is the lack of ability to track the masked points. Most of them are unable to do semi-automatic smoothing and segmentation in this stage. We developed algorithms to resolve these limitations. The trajectories obtained from the tracking are basically the observations of multi-dimensional functional data, so that the functional data analysis can be utilized to address the related issues.

## 1.3 Review of functional data analysis

The research area of functional data analysis (FDA) dates back to Grenander (1950) and Rao (1958) and the term was first used by Ramsay (1982). Nowadays, it has many applications in almost every branch of science, like engineering, medicine, biology and geology. Essentially, it aims at coping with the analysis of data in the form of images, curves and shapes. The most important characteristic of functional data is the intrinsically infinite dimensionality. This, on one hand, provides rich information and gives much chances for research work; on the other hand, brings challenges for theory and computation (Wang et al., 2015).

The typical first generation functional data are composed of independent real-valued functions $\{x_i(t), i = 1, \ldots, N\}$ defined on a interval $I = [0, L]$ on the real line. Gasser et al. (1984), Rice and Silverman (1991) and Gasser and Kneip (1995) have termed those data as curve data, which can also be regarded as the realizations of a one-dimensional stochastic process like Hilbert space. We usually model functional data with parametric approaches like the mixed effects nonlinear models (Raket et al., 2016), but the huge information hidden in the infinite dimensional data, the demand of a large degree of flexibility, as well as the natural ordering in the curve datum make loads of non- and semi-parametric approaches possible (Gervini and Gasser, 2004).

Furthermore, some challenges arise while extending those functional data from one-dimension to multi-dimensions, particularly the spatial and temporal registration problems (Gower, 1975a; Gervini and Gasser, 2004; Srivastava et al., 2011a). A more challenging problem is to do registration and modeling (classification and clustering) for multi-dimensional functional data at the same time. In this section, we will briefly illustrate the background of registration, focusing on the time warping, and the functional classification and clustering.

### 1.3.1 Time warping of functional data

Functional data always comes along with challenges, like observation noise, infinite-dimensionality of function spaces as aforementioned. Among these problems, the lateral displacement

termed as *phase variation* in curves, as opposed to *amplitude variation* in curve height, has drawn much attention. The former can always increase the data variance, distort principal components and make the underlying data structures unclear, so it is necessary to remove the phase variation from the amplitude in a desirable fashion.

To do so, we need to articulate the concept of a time-warping function, which is a mapping from one time scale to another. If we denote the system time or internal time scale as $t$, which is the underlying time process shared by all the observations, then the functional relationship $g^{-1}(t)$ represents the clock time or individual-specific time scale, varying one from another. We call $g^{-1}$ the *time warping function*. In statistics, we are always seeking methods to estimate $g^{-1}$.

### Time Warping function

There are many different types of warping functions to illustrate various phase variation. In most cases, the choice of warping function relies on the particular application context. It includes (a) uniform shift: shifting the time axis by a constant $a \in R$, i.e. $g^{-1}(t) = a+t$; (b) uniform scaling: rescaling the time warping by a constant $b \in R_+$ , i.e. $g^{-1}(t) = bt$; (c) linear transform: combining uniform shift and uniform scaling leads to linear transformation: $g^{-1}(t) = a + bt$; (d) diffeomorphisms: including domain warpings given by a set of diffeomorphisms of the domain to itself. If the domain is defined to be a full real line, the set of linear transformations is just a special case of the set of diffeomorphines. The warpings are practically restricted to compact intervals (Marron et al., 2015).

Generally, we define a warping function as the diffeomorphism: $g^{-1}(t) : [0, L] \to [0, L]$, which satisfies the following basic conditions:

1. Strict monotonicity: $g^{-1}(t_1) < g^{-1}(t_2)$ for $t_1 < t_2$ where $t_i \in [0, L]$,
2. Boundary conditions: $g^{-1}(0) = 0$ and $g^{-1}(L) = L$,
3. Continuity: $\forall \epsilon > 0$, $\exists \delta > 0$, as $|t_1 - t_2| < \delta$, $|g^{-1}(t_1) - g^{-1}(t_2)| < \epsilon$.

### Strategies of registration

The main purpose of registration is to remove phase variation from amplitude variation via estimating the warping function $g^{-1}$. We do this for the sake of reducing the variance of functional data and improving the statistical inference. Denote the functional data as

$$\boldsymbol{x}(t) : R \to R^A,$$

where $A$ is the dimension of $\boldsymbol{x}$ and $t$ represents the time scale for $A = 1$. $t$ can also be seen as the unit of the arc length along the curve as $A > 1$. The strategies for data registration, generally, can be divided into two categories as follows:

(a) While registering two curves $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, in other words, doing the pairwise alignment of functions, the mostly often used strategy is to find a good metric for $g^{-1}$: $\mu[g^{-1}|\boldsymbol{x}_1, \boldsymbol{x}_2]$. These metrics are, but not restricted to, a variety of loss functions like $L^2$ distance and similarity index defined by Sangalli et al. (2009). They might not only just focus on function $\boldsymbol{x}$ itself, but also care about its first derivative or second derivative, features like landmarks or even the related equivalence classes (Srivastava et al., 2011a,b). Then optimize this objective function

$$\tilde{g}^{-1} = \arg\max_{g^{-1}\in G} \mu[g^{-1}|\boldsymbol{x}_1, \boldsymbol{x}_2], \qquad \text{or} \qquad \tilde{g}^{-1} = \arg\min_{g^{-1}\in G} \mu[g^{-1}|\boldsymbol{x}_1, \boldsymbol{x}_2], \quad (1.1)$$

where $G$ is the group of warping function $g^{-1}$, having different structures for specific application context. The dynamic programming algorithm is widely used to obtain the approximate global optimal solution.

Sometimes, the registration for multiple curves seems difficult in analyzing data unless there already exist a template. In most cases the technique for multiple registration is just the extension of the binary case, constructing a template by an iterative way and aligning each of the remaining curves to it (Ramsay and Li, 1998; Kneip et al., 2000). The others, however, try to use new methods, deriving models tailored to the entire function data (Gervini and Gasser, 2004; Tang and Müller, 1998).

(b) Another strategy is to model $g^{-1}$ directly or indirectly first and then estimate $g^{-1}$ via maximum likelihood estimation (Raket et al., 2014) or Bayesian inference (Cheng et al., 2016; Wu and Hitchcock, 2016; Earls and Hooker, 2017)

$$\tilde{g}^{-1} = f(M[g^{-1}|\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]), \quad (1.2)$$

where $M$ denotes the model for $g^{-1}$ and $f$ represents any function, such as the mean of the posterior distribution or the maximum likelihood for the model, etc.

Most of these methods are confined to registering the functional data in the case of $A = 1$, i.e. one dimensional situation. Only a few approaches, like these by Sangalli et al. (2009), Srivastava et al. (2011a) and Cheng et al. (2016) can be applied to the multi-dimensional case. Their methods, however, are mostly used as one kind of preprocessing technique before statistical analysis. It is of interest for us to find a new way to do registration for multi-dimensional functional data, whether before or during modelling.

### 1.3.2 Functional regression

Functional regression has been widely studied and it generally has two kinds: (1) functional responses with either scalar or functional covariates or both and (2) scalar responses with both scalar and functional covariates (Ramsay and Silverman, 2005). For the former, Ramsay and Dalzell (1991) proposed the functional linear model (FLM), while the idea originates from Grenander (1950) who derived it as the regression of one Gaussian process on another. For the latter, the topic has been extensively explored, like Müller (2005, 2011) and Morris (2015). We will focus on the second kind in this thesis.

The functional linear model with scalar response $y \in R$ and functional covariate $x(t) \in R$ can be expressed as

$$y = b_0 + \int_I x(t)\beta(t)dt + \epsilon, \tag{1.3}$$

where $b_0$ and $\beta(t)$ are the regression coefficient and functional coefficient respectively, $\epsilon$ is a zero mean random error, $t \in I$; see e.g. Cardot et al. (1999, 2003), Hall and Horowitz (2007) and Hilgert et al. (2013). Usually, we use the same functional basis, like B-spline basis, to expand both the functional covariate $x(t)$ and the coefficient function $\beta(t)$. For instance, while expanding $x(t)$ and $\beta(t)$ in orthonormal basis $\{\phi_j, j \geq 1\}$ into $x(t) = \sum_{j=1}^{\infty} B_j\phi_j(t)$ and $\beta(t) = \sum_{j=1}^{\infty} \beta_j\phi_j$ respectively, model (1.3) is equivalent to the traditional linear model with the form

$$y = b_0 + \sum_{j=1}^{\infty} \beta_j B_j + \epsilon,$$

where the summation on the $\beta_j B_j$ can be approximated by a finite sum, which is truncated at the first $J$ terms. The functional linear model (1.3) can be extended to multiple functional covariates $\{x_a(t), a = 1, \ldots, A\}$, with a vector of scalar covariates $\{v_j, j = 1, \ldots, p\}$ by

$$y = \sum_{j=1}^{p} v_j\alpha_j + \sum_{a=1}^{A} \int_{I_a} x_a(t)\beta_a(t)dt + \epsilon. \tag{1.4}$$

The inference of model (1.4) is slightly different from model (1.3) because of the presence of the unknown parameters $\{\alpha_j, j = 1, \ldots, p\}$. Hu et al. (2004) proposed one combined least squares method to estimate $\alpha_j$ and $\beta_j$.

Adding a nonlinear link function $f$ to the functional linear model (1.3) produces a generalized functional linear model

$$y = f\big(b_0 + \int_I x(t)\beta(t)dt\big) + \epsilon. \tag{1.5}$$

Model (1.5) can be within the exponential family or a quasi-likelihood framework and a

suitable variance function. It has been investigated as $f$ is known (James, 2002; Cardot et al., 2003; Cardot and Sarda, 2005; Wang et al., 2010; Dou et al., 2012)) and unknown (Müller and Stadtmüller, 2005; Chen and Müller, 2011; Goldsmith et al., 2011). We refer to Wang et al. (2015) for a comprehensive review on this subject.

### 1.3.3 Classification of functional data

Functional data classification is aimed to assign a group membership to a new data object with a classifier or a discriminant. Most approaches, such as generalized functional linear regression models and functional multi-class logit models, are based on functional regression models featuring class labels as responses and the observed functional data and other scalar covariates as predictors. Those methods usually apply a dimension reduction technique using a truncated expansion in the data-adaptive eigenbasis or a pre-specified function basis.

Generalized functional linear models (James, 2002; Müller and Stadtmüller, 2005; Müller, 2005; Goldsmith et al., 2011), including the functional logistic regression model, are the most popular methods for regression-based functional classification. For a random sample with two groups $\{(y_i, \boldsymbol{x}_i); i = 1, \ldots, n\}$, where $y_i \in \{0, 1\}$ represents a class label and $\boldsymbol{x}_i$'s are functional observations, a classification model for a functional observation $\boldsymbol{x}^*$ based on functional logistic regression is

$$\begin{aligned} \pi &= p(y^* = 1|\boldsymbol{x}^*), \\ \text{logit}(\pi) &= b_0 + \int_I \boldsymbol{x}^*(t)\beta(t)dt, \end{aligned} \tag{1.6}$$

where $b_0$ is an intercept term and $\beta(t)$ the coefficient function of the predictor $\boldsymbol{x}^*$. The model-based Bayes classification rule chooses the class label $y^*$ with the maximal posterior probability among $\{p(y^* = k|\boldsymbol{x}^*); k = 0, 1\}$. This model can be easily extended to the case with $K(K > 2)$ groups. Several variants of the functional logistic regression model have been studied (Wang et al., 2007; Zhu et al., 2010; Rincon and Ruiz-Medina, 2012).

### 1.3.4 Clustering of functional data

Functional clustering is an active research area in FDA and has received great attention in the last decade. It is different from functional classification due to the unknown class labels while grouping those objects. The aim of clustering is to group a set of data such that data within groups (clusters) are more similar than across groups with respect to a metric. It is often used as a preliminary step for data exploration by identifying particular patterns to provide the user with convenient interpretation. Generally, it is a difficult task due to the lack of distances or estimation from noise data and a definition for the

probability of a functional variable. The most popular approaches of functional clustering can mainly be divided into two categories: model-based approaches and non-parametric approaches.

The model-based functional clustering technique is also called distribution-based clustering. One approach is to model principal components (Delaigle and Hall, 2010; Bouveyron and Jacques, 2011) or basis expansion coefficients (James and Sugar, 2003; Sam et al., 2011)) with mixture Gaussian distributions. Another method is to model those curves directly by mixture Gaussian process (Shi et al., 2005; Shi and Wang, 2008). Clusters can be defined as objects (principal components, coefficients or curves) belonging most likely to the same distribution. The related theoretical foundation is solid while suffering from over-fitting. The EM algorithm is one of the most popular methods to implement estimation, though may converge to a local optimum.

The non-parametric clustering mainly includes connectivity-based clustering, also known as hierarchical clustering (Ferraty and Vieu, 2006) and centroid-based clustering, also known as $k$-means clustering (Tarpey and Kinateder, 2003; Tokushige et al., 2007; Ieva et al., 2013). The hierarchical clustering is based on the idea of curves being more related to nearby curves than to curves further away. The related algorithms connect curves to form "clusters" based on their distance. A cluster can be largely described by the maximum distance required to connect parts of the cluster. $k$-means clustering is to find the $k$ cluster centers and then assign the curves to the nearest cluster center to minimize the squared distances from the cluster. We can also use the classical clustering tool for finite dimensional data after reducing dimension. Specifically, after approximating the curves into a finite basis of functions (Abraham et al., 2003), we can summarize the curves by their coefficients in a basis of functions or by their first principle component scores and then perform clustering. For instance, Abraham et al. (2003) and Peng and Müller (2008) perform the $k$-mean algorithm on B-spline coefficients and on a given number of principle component scores, respectively. We refer to Jacques and Cristian (2014) for a comprehensive review on functional clustering.

The common limitations of functional classification and clustering by those methods aforementioned are (1) few of them are able to do registration while modelling the multi-dimensional functional data; (2) most of them ignore the use of scalar variables, which often provide useful information. This thesis will focus on solving these problems.

## 1.4 Structure of the thesis

The thesis is organized as follows. In Chapter 2, we develop a framework for data acquisition from X-ray image. The background on subjects and experimental design is firstly described and then followed by the methodology of semi-automatic tracking for the hyoid

bone. We also introduce procedures of semi-automatic smoothing, calibration and automatic segmentation for the raw data. The validation results show that the semi-automatic tracking has high agreement with manual tracking.

Chapter 3 proposes a methodology to implement the registration for the multi-dimensional functional data. It firstly reviews the background of *GPA* (Gower, 1975a) and self-modeling registration (Gervini and Gasser, 2004) and then discusses the integration of those two methods (*GPSM*), as well as the corresponding algorithm. Numerical analyses are given afterwards.

In Chapter 4, we propose one methodology for joint curve registration and classification with mixed scalar and functional variables (*JCRC*). It consists of two-stage functional models with the first stage using a functional logistic regression model where the aligned curves are estimated from the second stage model. The later uses the functional mixed effect model for simultaneous registration and curve modelling. This methodology takes advantage of both scalar variables and functional variables. Procedures of model inference and implementation, as well as the asymptotic properties of interested parameters are introduced. We also implement an iterative algorithm for predicting the outcomes and present numerical analyses to investigate the performance of the proposed method.

Furthermore, Chapter 5 proposes the simultaneous curve registration and clustering (*SRC*) for multi-dimensional functional data. Two-level models are introduced, including the mixtures of Gaussian process functional regression with time warping as the first level model and the logistic allocation model using the scalar variable as the second level model. This methodology allows for simultaneous registration and modelling, and allows for the use of both functional variables and scalar variables. It is implemented using an EM algorithm. A comprehensive simulation study and real data analyses are followed in the end.

Finally, we conclude in Chapter 6 with comments on future work.

# Chapter 2

# Semi-automatic Tracking, Smoothing and Segmentation of Hyoid Bone Motion from Videofluoroscopic Swallowing Study

## 2.1 Introduction

Motion analysis of hyoid bone via videofluoroscopic study has been used in clinical research, but the classical manual tracking method is generally labor intensive and time consuming. Although some automatic tracking methods have been developed, masked points could not be tracked. The smoothing and segmentation, which are necessary for functional motion analysis prior to registration, were not provided by the previous software either. In this chapter, we try to develop a software to track the hyoid bone motion semi-automatically. It works even in the situation where the hyoid bone is masked by the mandible and it has been validated in dysphagia patients with stroke. In addition, we added the function of automatic smoothing and segmentation, which is necessary for further quantitative motion analysis and can reduce the time needed for manual working. The development of the automatic or semi-automatic process from hyoid bone tracking and smoothing to segmentation enables the motion analysis of VFSS to have a potential wide use in clinical practice and research. This work has been published (Kim et al., 2017).

The structure of this chapter is as follows. Firstly, Section 2.2 briefly introduces the background on subjects and experimental design. The methodology of semi-automatic tracking for the hyoid bone, the way of semi-automatic smoothing and calibration, as well

as the semi-automatic segmentation for the curves are given in Section 2.3. Section 2.4 discusses the validation and rough statistical analysis for hyoid bone motion. A summary of this chapter follows in Section 2.5.

## 2.2 Subjects and experimental design

VFSS data and medical information for stroke patients were retrospectively reviewed from the database of VFSS movie files and medical records in Seoul National University Bundang Hospital. A total of 30 patients' data (mean age: $62.0 \pm 11.4$ yrs, 23 men and 7 women) were used to develop software (tracking, smoothing and calibration, and segmentation) and 20 circles from 17 patients trajectories (10 circles from 8 unmasked trajectories and 10 circles from 9 masked trajectories) were used to validate the semi-automatic tracking method. One circle for each subjects trajectory of hyoid bone was usually detected and extracted while two circles were obtained in subjects number 5, 11 and 16. Each circle was used to validate the semi-automatic tracking method.

VFSS was tested in subjects with dysphagia after stroke with foods in various forms, including fluid, thickened fluid, a semi-blended food, and boiled rice, which was the modified protocol (Logemann, 1993). Each food was provided by spoon. The lateral projection of the VFSS taken during the 2-ml thin-fluid swallowing was used for software development and validation. VFSS were recorded at 30 frames per second.

One researcher performed the manual tracking and automatic tracking of hyoid bone from VFSS clips. When one type of tracking was performed, the tester did not consult the result of another type of tracking in each patient. After all tracking were completed, the validation were performed without modifying tracking results. The research protocol was approved by the Seoul National University Bundang Hospital institutional review board and was conducted in accordance with the regulatory standards of Good Clinical Practice and the Declaration of Helsinki (World Medical Association Declaration of Helsinki: Ethical Principles for Medical Research Involving Human Subjects, 2000).

## 2.3 All-in-one platform for the motion analysis of hyoid bone

### 2.3.1 Overview

Inspired by the method by Kellen et al. (2010), in our study we specify a target point on the hyoid bone on one frame and then track the target automatically for the whole video sequence. The ROI (region of interest) window size by default should be large enough so that it is not so sensitive to the smaller movements of the hyoid bone. Each frame of the

sequence is then processed to track the ROI centered at the target point across frames. The tracking of the ROI which is partly masked by other objects such as the mandible in some frames has been considered in our methodology. Furthermore, we also consider the situation where the tracking process might collapse due to the existence of unidentifiable and invisible ROI in some frames. An automatic monitoring and indication mechanism has been added, enabling us to re-specify the target point and reset the window size of ROI and then resume the tracking process. In order to correct for the subject's head motion during process, a new coordinate system is defined via the anterior-inferior border of the second and fourth cervical vertebrae across the entire procedure. Semi-automatic smoothing via cubic smoothing is added for those target points in the hyoid bone and in the cervical vertebra for the sake of reducing tracking errors. Our platform also emphasizes the segmentation. After selecting one desired circle from the data, the automatic segmentation will be carried out. By analyzing the first and second derivatives, a definition of splitting score is introduced and used for an automatic segmentation. This provides necessary and useful information for clinical assessment and further statistical analysis such as functional classification. The code for data tracking and for data preprocessing like semi-automatic smoothing, calibration, validation and segmentation is based on MATALAB (R2014a) and RStudio Version 0.99.484 - ©2009-2015 RStudio, Inc.

### 2.3.2 Procedures of tracking

To define the template ROI, the user uses the mouse to identify any target point on the hyoid bone and then a square centered at it with default side lengths can be created automatically. The target point, together with this square, called ROI or template, are tracked automatically frame by frame by utilizing the information from horizontal and vertical edge images calculated using Sobel edge operators (Sobel, 1990). The key point is to minimize the sum of the squared difference between the local edge characteristics in the templates and that in the images which have been rotated within and shifted over a suitable neighborhood. The best match for the template in the next frame can then be found. Hence the tracking process can be iterated by updating the positions of both ROI and the target point. For the partly masked frames, another two points along the edge of the mandible should be identified by mouse. Using the methodology described in the previous section, the target point can still be tracked automatically.

When it comes to extreme situations, for example the ROI or the target point on the hyoid bone being covered by other objects like the lower part of the mandible, the tracking method (Kellen et al., 2010) no longer works because the local edge characteristics of ROI are heavily interrupted. So far there has been little research on addressing this problem.

Technically, the masked ROI refers to the ROI totally or partly overlapped with other objects such as the mandible during the tracking process. This often happens over just a

few frames of the whole video sequences. In this case, it is hard to locate the target point even by human eyes. The idea is to cut off the masked part in the current ROI. Besides the target point, it is necessary to track another two points along the edge of the lower part of the mandible simultaneously via the same tracking method (Kellen et al., 2010). The locations of these two points should be flexible and the distance between them wide enough to guarantee the line segment connected by them approximates the lower edge of the mandible properly. We then check whether the current ROI crosses this line. The masked part above the line will be cut off if it crosses; otherwise, it will be treated as a normal case. Consequently, a new ROI$'$ on which the normal procedure of tracking is based can be obtained (Figure 2.1). Another key point is the requirement of averaging the template matching error over the number of pixels within the new ROI$'$.

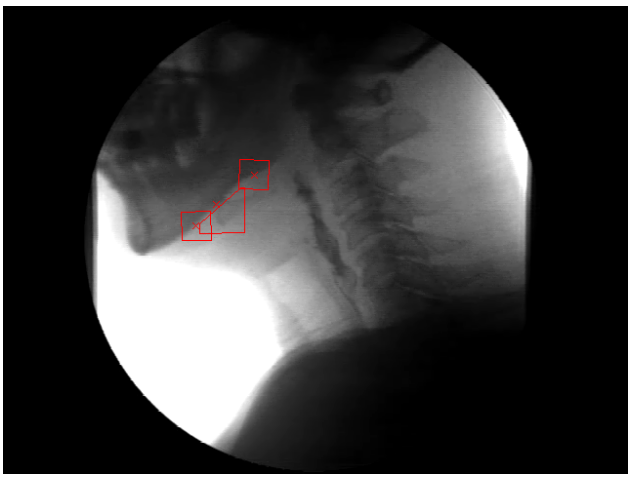


Figure 2.1: Example of tracking the partly masked ROI. The target point pinpointed by the middle red cross is covered by the mandible in this case. This ROI (the middle square) is cut off by the line segment linked by another two red crosses (their corresponding ROI$'$s are the upper and lower squares) along the lower edge of the mandible.

Assuming $i_0$ is the reference frame number, in which the template ROI($i_0$) is produced and $i$ represents the current frame number, the template in the current frame is then supposed to move with a rotation angle $\tilde{\theta}$ and translation $(\tilde{x}_1, \tilde{x}_2)^T$ [1]. After this transformation, the removal of the hidden part of the template (Figure 2.1) is required. As a result, we obtain a new ROI$'(i+1)$ in the $(i+1)$th frame, which is no longer a square. All the $x_1$'s and the corresponding $x_2$'s of pixels within ROI$'(i+1)$ are then saved in the vectors $\text{ROI}'_{x_1}(i+1)$ and $\text{ROI}'_{x_2}(i+1)$ in order respectively. The ROI($i_0$) needs to be changed to ROI$'(i_0)$ in the same way to make their local edge characteristics $E_{x_1}$ and $E_{x_2}$

[1] We use $x_1$ and $x_2$ to represent $x$-coordinate and $y$-coordinate of the point throughout the thesis.

comparable (Kellen et al., 2010). The template matching error $\Delta(\tilde{x}_1, \tilde{x}_2; \tilde{\theta})$ is given by

$$\Delta(\tilde{x}_1, \tilde{x}_2; \tilde{\theta}) = \frac{1}{N(i+1)} \sum_{\substack{x_1 \in \mathrm{ROI}'_{x_1}(i_0); x_2 \in \mathrm{ROI}'_{x_2}(i_0); \\ x'_1 \in \mathrm{ROI}'_{x_1}(i+1); x_2 \in \mathrm{ROI}'_{x_2}(i+1)}} (E_{x_1}(x_1, x_2, i_0) - E_{x_1}(x'_1, x'_2, i+1))^2 + (E_{x_2}(x_1, x_2, i_0) - E_{x_2}(x'_1, x'_2, i+1))^2, \tag{2.1}$$

where $N(i+1)$ stands for the total number of pixels within $\mathrm{ROI}'(i+1)$. Equation (2.1) is minimized by a global optimization method GA with a constrained search space (Michalewicz and Hartley, 1996). In our implementation, the search constraints are: $-2.5\pi/180 \leq \tilde{\theta} \leq 2.5\pi/180$, and $-5 \leq \tilde{x}_1, \tilde{x}_2 \leq 5$. The tracking for the next position of the template and target point in this special case can then successfully progress. However, the tracking process may become unstable if the ROI is totally masked by the mandible. In this rare case, we may estimate the underlying point by the points tracked in nearby frames or to track it manually.

In some rare extreme situations, such as the hyoid bone moving suddenly or too fast, the target point is hardly recognizable (Figure 2.2). The optimal search is not applicable

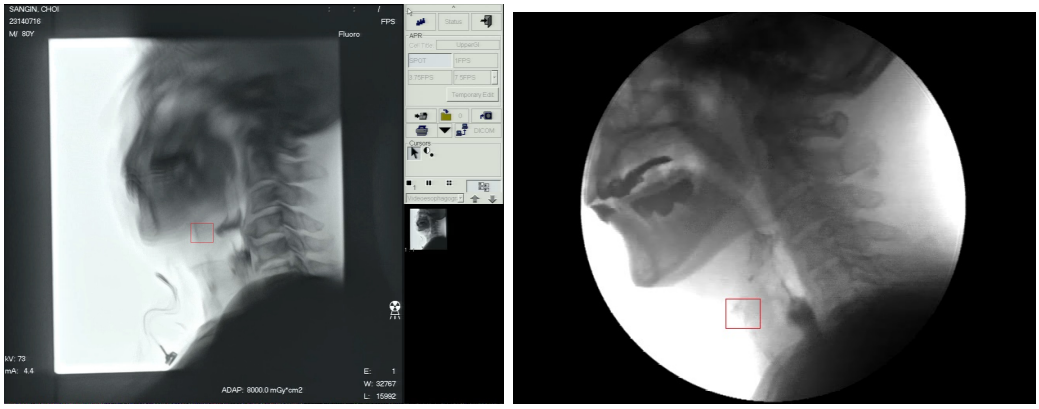


Figure 2.2: Example of unrecognizable hyoid bone located in the square in red. The left plot: the hyoid bone moves too fast, resulting in the almost equal gray scale value of the area around it. The right frame: the strong reflective light makes the hyoid bone invisible.

to those circumstances; therefore, a sensitive monitoring mechanic should be used to avoid possible wrong tracking. Kellen et al. (2010) used a prediction model to initialize the new point position for the sake of improving tracking accuracy. We used a similar idea but for monitoring purpose in our package. The prediction model and displacement error for the next position of the target point are given by

$$\tilde{H}(i+1) \approx H(i) + \dot{H}(i)\Delta t + \frac{\ddot{H}(i)}{2}(\Delta t)^2 + \frac{\dddot{H}(i)}{6}(\Delta t)^3,$$

and

$$\Delta H = |\tilde{H}(i+1) - H(i+1)|.$$

Here, $\Delta t$ is the inter frame period, $i$ is the current frame's sequence number, $H(i)$ and $H(i+1)$ are hyoid bone's current and next position (or coordinates), $\dot{H}(i)$, $\ddot{H}(i)$ and $\dddot{H}(i)$ are respectively the first, second and third derivative of $H(i)$. Given the previously acquired $H(i)$, the $\Delta H$ (absolute difference between $\tilde{H}(i+1)$ and $H(i+1)$) should be less than a prespecified threshold (8 pixel units in our implementation). Otherwise, the tracking will be regarded as a failure. The software will then automatically review the possible wrong-tracking frames backward and forward to identify the accurate sequence number of the first failure frame. After confirming it, we should then manually adjust the window size of ROI($i$), and then go back to the tracking procedure by re-specifying the same target point in the $i$th frame by mouse click and continue the tracking process.

### 2.3.3 Smoothing and calibration

The tracking described in the previous subsections is based on the coordinate system where the origin is located at the bottom-left corner of the image (image-based coordinate system). The problem is that there might be sudden body or head motion which would blur the movement of hyoid bone during the swallowing process. To remove as much of this kind of error as possible, a new so-called patient-centric coordinate system is required (Kellen et al., 2010; Potratz et al., 1992). Practically, it seems to be much more convenient and efficient to define a new coordinate system based on two special points. As described by Kim et al. (2015), the $y$-axis of the patient-centric coordinate system is defined as a straight line connecting the anterior-interior border of the fourth cervical vertebra ($C4(x_1^{c_4}, x_2^{c_4})$, origin) to that of the second cervical vertebra ($C2(x_1^{c_2}, x_2^{c_2})$). The $x$-axis is defined as a line perpendicular to the $y$-axis crossing the origin, $C4$, as seen in Figure 2.3B. The points $C4$ and $C2$ can be tracked at the same time using the methods illustrated in preceding subsections over the entire video sequences (Figure 2.3A). To reduce the errors caused by tracking, smoothing is carried out for both the target point in the hyoid bone and the two tracking points in the cervical vertebra using a cubic smoothing spline. The degree of smoothing is controlled by the smoothing parameter, which can be adjusted by the operator to avoid over-fitting (Figure 2.3D). Then all the data is normalized by the vertical distance from $C4$ to $C2$. Given those two points' coordinates, the target point $H(x_1, x_2)$ in the image-based coordinate system can be transformed to $H'(x_1', x_2')$ in the patient-centric coordinate system by a simple rotation and translation (Figure 2.3C).

The formula is given by

$$(x_1', x_2')^\intercal = \frac{1}{|C2 - C4|} \begin{bmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{bmatrix} (x_1 - x_2^{c_4}, x_2 - x_2^{c_4})^\intercal,$$

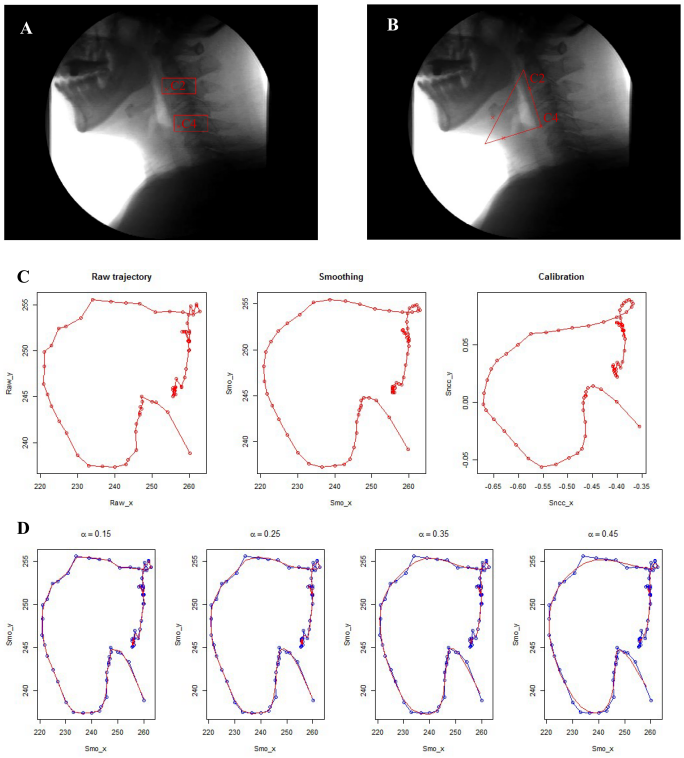



Figure 2.3: Example of smoothing and calibration. A. Manually specified two points, C2 and C4, indicated as two red crosses, in the anterior-interior border of the second and the fourth cervical vertebra at the beginning of tracking. B. The patient-centric coordinate system with the origin C4, where the $y$ axis is defined as the line crossing C4 and C2 upward and the $x$ axis is defined as the line perpendicular to the $y$ axis leftward. C. The rugged trajectory in the left panel is raw data based on the image-centric coordinate system (in pixel) while the smoothing one in the middle and the calibrated one in the right panel is based on patient-centric coordinate system (in CU). D. Semi-automatic smoothing by adjusting the spline parameter, which ranges from 0.15 to 0.45. Blue curves represent the raw trajectory while the red ones are smoothing curves.

where

$$\theta = \frac{\pi}{2} + \arctan\Big(\frac{x_1^{c_4} - x_1^{c_2}}{x_2^{c_4} - x_2^{c_2}}\Big), \qquad |C2 - C4| = \sqrt{(x_1^{c_4} - x_2^{c_2})^2 + (x_2^{c_4} - x_2^{c_2})^2}.$$

The effect of smoothing on diminishing tracking errors is demonstrated in the lower panels of Figure 2.3. The calibration procedure aims to reduce the errors caused by head motion and make the data collected from different subjects comparable. Our later data analysis will be based on the trajectory after both smoothing and calibration, of which the coordinate is based on cervical units (CU) (One cervical unit is defined as the distance (in pixel) between $C2$ and $C4$, i.e.$|C2 - C4|$).

### 2.3.4 Segmentation

Dividing one circle of the trajectory into certain phases is necessary for the assessment of the hyoid bone movement and is useful for further statistical analysis. However, little research has been carried out in this area, particularly on automatic segmentation. In terms of the concepts on phases, Kaneko (1992) performs a quantitative study manually dividing the movement into 5 phases: 1st elevation phase, 2nd elevation phase, static phase and 1st and 2nd descending phase. Yabunaka et al. (2011) have also done similar work on sonographic assessment of hyoid bone movement during swallowing by segmenting the movement into 4 phases: Elevation, Anterior, Remain and Return. We developed a semi-automatic segmentation method. After manually selecting one complete desired circle from the entire raw trajectory, an automatic segmentation is conducted.

For simplicity, we use points $(x_1(t), x_2(t))$ to represent $x$ and $y$ coordinates after calibration and smoothing in the $t$-th frame sequence. Two ends of the manually identified complete circle are denoted by $t_A$ and $t_B$. The technique is to acquire one desired time interval including one peak in $x_2(t)$ and one valley in $x_1(t)$ at the same time. These two points can be easily chosen by human eyes. For instance, the left panel of Figure 2.4A shows that it is easy to identify the peaks and valleys in these two marginal curves. The end points of $t_A$ and $t_B$ are chosen such that the interval $(t_A, t_B)$ contains both one peak in the upper curve $(x_2(t))$ and one valley in the lower curve $(x_1(t))$. Furthermore, the distance between $(x_1(t_A), x_2(t_A))$ and $(x_1(t_B), x_2(t_B))$ should be as small as possible (see the left panel of Figure 2.4B). The ideal situation is that the distance is equal to zero, i.e., the starting point $A$ and ending point $B$ overlap.

Once the desired circle is obtained, automatically dividing the movement into different phases over $(t_A, t_B)$ is workable via analyzing the corresponding velocity amplitude

$$v(t) = \sqrt{(\frac{dx_1(t)}{dt})^2 + (\frac{dx_2(t)}{dt})^2}, \quad t \in (t_A, t_B).$$

Specifically, we can find the splitting points $t$ from the equations: **(A)** $\frac{dv(t)}{dt} = 0$; **(B)** $\frac{d^2v(t)}{dt^2} \geq 0$. Conditions **(A)** and **(B)** guarantee that all the local minimal points in the velocity amplitude curve are found (see the blue curve in the left panel of Figure 2.4C). Those local minima are interesting splitting points, which can be directly used in segmentation in most cases. The corresponding segmentation result is shown in the right panel of Figure 2.4C. In this case the three splitting points, as well as the start point and end point, are used to split the trajectory into four phases: elevation phase, anterior movement phase, descending phase and returning phase.

It is not uncommon that more than three candidate splitting points can be obtained only based on the conditions **(A)** and **(B)** (Figure 2.5C), particularly for patients with

stroke. We propose to find the three best splitting points based on a new measurement, namely, the **Splitting Score**. Assume there exist $m-2$ candidate splitting points:$t_2, t_3, \ldots, t_{m-1}$(except the starting point $t_1$ and end point $t_m$). For the $i$th splitting point $t_i (i \in [2, m-1])$, we use the following measures.

- Forward Splitting Score

$$\mathrm{FSS}(t_i) = \max(v(t)) - v(t_i), \qquad t \in [t_{i-1}, t_i].$$

- Backward Splitting Score

$$\mathrm{BSS}(t_i) = \max(v(t)) - v(t_i), \qquad t \in [t_i, t_{i+1}].$$

- Splitting Score

$$\mathrm{SS}(t_i) = \mathrm{FSS}(t_i) + \mathrm{BSS}(t_i).$$

In fact, the $SS$ value can be regarded as the turning intensity for the candidate points, the larger, the better. Those $t_i'$s with the top 3 Splitting Scores are chosen as the desired splitting points. Figure 2.5C shows that there are 6 candidate splitting points satisfying the conditions of (A) and (B). It is easy to identify the three points, indicated by the labels 2, 5, 6, with the top 3 SS values. The result is shown in Figure 2.5D.

## 2.4 Validation and statistical analyses

We tracked 20 circles from 17 subjects using our semi-automatic tracking methodology. Next, each swallow was also tracked manually by a trained observer, who was instructed to track one recognizable and fixed target point on the hyoid bone across all frames by clicking the mouse. For the same swallow, we compare the two different trajectories tracked by automatic computer-assisted method and manually by human being. Similar to the previous study by Kellen et al. (2010), we used Pearson correlation coefficients and relative errors defined as

$$\frac{|\mathrm{ROM}_{\text{automatic tracking}} - \mathrm{ROM}_{\text{manual tracking}}|}{\mathrm{ROM}_{\text{manual tracking}}} \times 100\%$$

to measure the degree of agreement between the both, where ROM stands for range of motion. The range-of-motion measurement is calculated by finding the largest displacement between any two points on the hyoid bone trajectory. Apart from the raw trajectories (raw data without smoothing and calibration), five more comparisons were considered in our validation: RawNC (Raw data transforming to a new coordinate system without being scaled), RawNCC (Raw data with both coordinate system alternation and scaling), Smo

(Raw data with smoothing), SmoNC (RawNC data with smoothing), SmoNCC (RawNCC data with smoothing). Continuous variables are presented as mean ± 1 SD. Parameters generated were compared between aspiration and non-aspiration groups using an independent t-test. Furthermore, in order to justify the current validation methodology, the Intraclass correlation and Pearson correlation are utilized to measure the inter-rater reliability between those two observers. We calculate those two measurements for both $x$ coordinates and $y$ coordinates of three points' locations in each frame by two raters' manual tracking. As mentioned before, those points are respectively located at the bottom left of the hyoid bone, anterior-interior border of the second cervical vertebra and that of the fourth cervical vertebra.

### 2.4.1 Results

Figure 2.6 and Figure 2.7 show both computer defined and manual defined trajectories corresponding to six cases for two typical data sets, one from the unmasked group and the other from the masked group. Overall, two trajectories in each case match pretty well (see Pearson correlation coefficients and relative errors between manual tracking and automatic tracking in Table 2.3). The slight difference may be caused by different target points on the hyoid bone identified by computer and the trained observer.

| Cases | ROM in $x$-axis | | ROM in $y$-axis | | ROM in 2D | |
|---|---|---|---|---|---|---|
| | A | M | A | M | A | M |
| Unmasked group (10 circles from 8 trajectories) | | | | | | |
| Raw | 55.22 ±15.06 | 57.57±13.87 | 53.96 ± 26.69 | 55.77 ± 26.49 | 70.72 ± 23.31 | 72.67±23.45 |
| RawNC | 54.98±16.21 | 56.87±14.58 | 55.06 ± 27.24 | 56.39 ± 25.27 | 71.50 ± 23.65 | 72.54±22.45 |
| RawNCC | 0.27±0.10 | 0.29±0.11 | 0.27 ± 0.18 | 0.29 ± 0.18 | 0.36 ± 0.17 | 0.38±0.18 |
| Smo | 55.01±15.01 | 55.00±14.63 | 53.16 ± 26.53 | 53.12 ± 26.55 | 70.53 ± 22.80 | 69.88±23.62 |
| SmoNC | 54.49±16.19 | 53.58±14.41 | 53.95 ± 26.96 | 53.67 ± 25.97 | 71.35 ± 23.33 | 69.90±23.12 |
| SmoNCC | 0.27±0.10 | 0.28±0.11 | 0.27 ± 0.17 | 0.28 ± 0.18 | 0.36 ± 0.17 | 0.37±0.18 |
| Masked group (10 circles from 9 trajectories) | | | | | | |
| Raw | 56.23±22.04 | 58.44±22.34 | 41.80 ± 16.06 | 43.01 ± 15.89 | 63.80 ± 22.52 | 65.89±23.32 |
| RawNC | 56.00±18.39 | 57.30±16.89 | 41.63 ± 15.46 | 43.48 ± 15.32 | 64.01 ± 21.30 | 64.24±18.38 |
| RawNCC | 0.31±0.13 | 0.33 ±0.14 | 0.22 ± 0.10 | 0.23 ± 0.11 | 0.36 ± 0.14 | 0.36±0.14 |
| Smo | 55.50±22.41 | 56.11±22.00 | 40.66 ± 16.43 | 41.03 ± 15.51 | 62.68 ± 22.46 | 63.53±22.73 |
| SmoNC | 55.14±18.49 | 55.37±17.23 | 40.35 ± 15.89 | 41.67 ± 15.03 | 63.12 ± 21.44 | 62.56±18.48 |
| SmoNCC | 0.31±0.13 | 0.32 ±0.14 | 0.21 ± 0.10 | 0.22 ± 0.10 | 0.35 ± 0.15 | 0.35±0.14 |

Table 2.1: ROM comparison between two methods. ROM - range of motion, A - automatic tracking, M - manual tracking, 2D - 2 dimensions, Raw - raw data without smoothing and calibration, RawNC - raw data transformed to a new coordinate system yet without being scaled, RawNCC - raw data with complete calibration, Smo - raw data with smoothing, SmoNC - RawNC data with smoothing, SmoNCC - RawNCC data with smoothing. Values are mean ± SD.

Table 2.1 shows the range of motion between manual tracking and automatic tracking in terms of the $x$-axis, $y$-axis and 2D direction. Table 2.2 shows Pearson correlation

| Cases | Pearson r | | Relative errors(%) | | |
|---|---|---|---|---|---|
| | $x$-axis | $y$-axis | $x$-axis | $y$-axis | 2D |
| Unmasked group (10 circles from 8 trajectories) | | | | | |
| Raw | 0.982 | 0.982 | $6.1 \pm 4.7$ | $6.5 \pm 4.7$ | $5.6 \pm 4.1$ |
| RawNC | 0.977 | 0.954 | $8.7 \pm 5.0$ | $6.5 \pm 4.5$ | $5.9 \pm 3.9$ |
| RawNCC | 0.977 | 0.958 | $9.2 \pm 6.6$ | $8.6 \pm 6.0$ | $6.9 \pm 4.8$ |
| Smo | 0.991 | 0.990 | $4.8 \pm 3.8$ | $4.6 \pm 3.1$ | $4.6 \pm 3.4$ |
| SmoNC | 0.984 | 0.967 | $6.5 \pm 4.6$ | $4.8 \pm 4.6$ | $4.8 \pm 3.9$ |
| SmoNCC | 0.984 | 0.971 | $7.6 \pm 5.4$ | $5.4 \pm 4.6$ | $5.4 \pm 3.3$ |
| Masked group (10 circles from 9 trajectories) | | | | | |
| Raw | 0.975 | 0.965 | $6.1 \pm 5.1$ | $5.0 \pm 4.3$ | $5.3 \pm 5.8$ |
| RawNC | 0.972 | 0.946 | $7.3 \pm 6.5$ | $8.5 \pm 7.8$ | $6.4 \pm 5.4$ |
| RawNCC | 0.969 | 0.942 | $8.6 \pm 7.6$ | $7.1 \pm 6.5$ | $6.0 \pm 6.2$ |
| Smo | 0.982 | 0.978 | $4.3 \pm 4.0$ | $3.3 \pm 4.1$ | $4.1 \pm 4.7$ |
| SmoNC | 0.979 | 0.961 | $5.0 \pm 5.6$ | $6.9 \pm 7.1$ | $6.2 \pm 4.3$ |
| SmoNCC | 0.975 | 0.957 | $6.8 \pm 6.9$ | $5.8 \pm 6.0$ | $5.5 \pm 4.9$ |

Table 2.2: Pearson correlation coefficients between two methods and relative errors (%) from two methods. Values are Pearson correlation coefficients or mean ±1 SD. P-values for all Pearson correlation coefficient were less than 0.0001.

| Tracking results | Methods | Estimate | 95 % CI | P value |
|---|---|---|---|---|
| $x$-coordinates | Pearson's r | 0.999 | (0.998, 0.999) | $< 0.0001$ |
| | ICC | 0.998 | (0.998, 0.999) | $< 0.0001$ |
| $y$-coordinates | Pearson's r | 0.998 | (0.998, 0.998) | $< 0.0001$ |
| | ICC | 0.996 | (0.995, 0.996) | $< 0.0001$ |

Table 2.3: Average Pearson correlation coefficients and Intraclass correlation coefficients (ICC) between two independent observers for measuring the inter-rater reliability.

coefficients and relative errors in terms of ROM from two methods. We can see that all of coefficients are in the interval between 0.942 and 0.991 ($p < 0.0001$) and the relative errors in terms of the $x$-axis, $y$-axis and 2D range of hyoid bone excursion ranges from 3.3 % to 9.2 %. Overall, the case of proper smoothing usually outperforms better.

Table 2.3 shows the average of Intraclass correlation and Pearson correlation coefficients for both $x$ coordinates and $y$ coordinates range from 0.995 to 0.999 ($p$-value $<$ 0.0001). It indicates a high consistency of quantitative measurements made by those two independent trained observers, which provides a justification of the methodological errors in our study.

According to our automatic segmentation method, most of the hyoid bone motion (14 out of 30 subjects) can be typically segmented automatically into four phases. All of the subjects fall into four groups in terms of segmentation results. There is an error for automatic segmentation for six subjects, but the typical four phases can be segmented

manually. There are four subjects with only 3 phases capable of being segmented (e.g. no returning phase). In six subjects, the trajectories are abnormal and could not be segmented into typical phases. Figure 2.8 shows four typical examples from each group.

## 2.5 Chapter Summary

To sum up, the main contributions of the present work include,

(a) the development of a new algorithm based on the existing method by Kellen et al. (2010) to track the masked part of the hyoid bone and a dynamic monitoring mechanic to fix the wrong-tracking problems in time,

(b) the development of semi-automatic smoothing and calibration for reducing tracking errors,

(c) the development of a new method of automatic segmentation of hyoid bone motion, which could provide the researchers in the field of dysphagia a convenient, useful, and all-in-one platform for the analysis of hyoid bone motion.

Once we have obtained these functional data, the next task is data preprocessing like registration. The deformations or displacements, termed phase variation, always arise in these curves. This can be shown through the different locations of splitting points while doing automatic segmentation. The presence of phase variability often increases data variance and alters underlying data structures (Marron et al., 2015). The splitting points can also regarded as the landmarks from the perspective of functional registration. It seems the standard landmark registration method, or the ones related to landmarks might be employed. Thus, we will study the 2D curve registration for the movement of hyoid bone, which is the purpose of the next chapter.

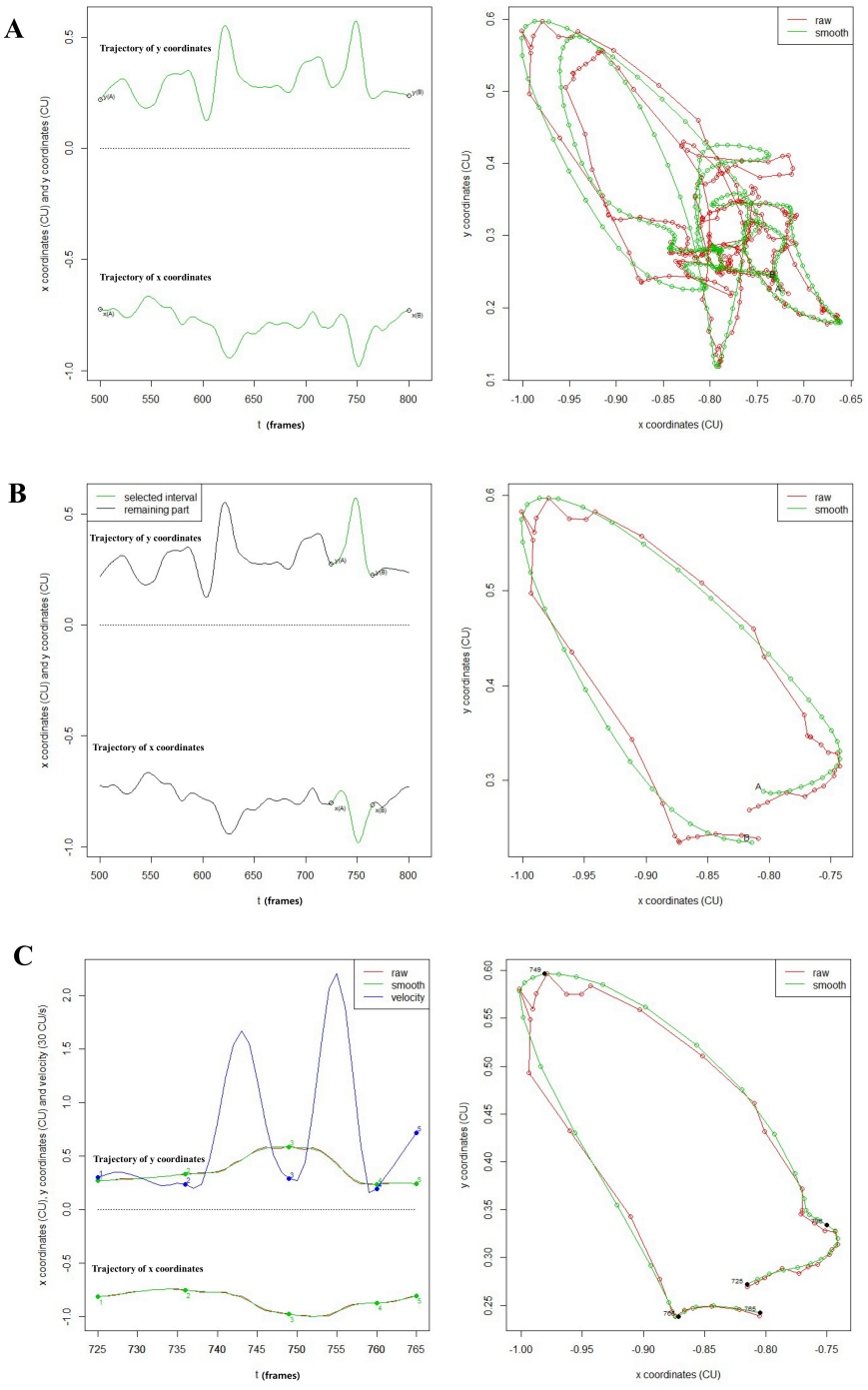



Figure 2.4: Example of automatic segmentation for one non-aspiration case. A. The curves of the $x$ coordinates and $y$ coordinates of all the data (the left panel) and the entire 2D trajectory (the right panel): the dots connected with a line in red show the raw trajectory just after being calibrated while the ones in green represent smoothing data after calibration. B. Extracted one circle based on the two cutting points A and B (the left panel) from the entire trajectory and the corresponding 2D trajectory (the right panel). C. Automatically segmenting the trajectory into four phases. The upper curve in green in the left panel stands for the smoothing $y$ coordinates and the lower one for the smoothing $x$ coordinates, together with the curves in red representing raw data. The curve in blue represents the velocity amplitude $v(t)$, where $t$ represents the video frame sequence and the numbers in different colors stand for splitting points order. The right panel shows the segmentation results in 2D trajectory.

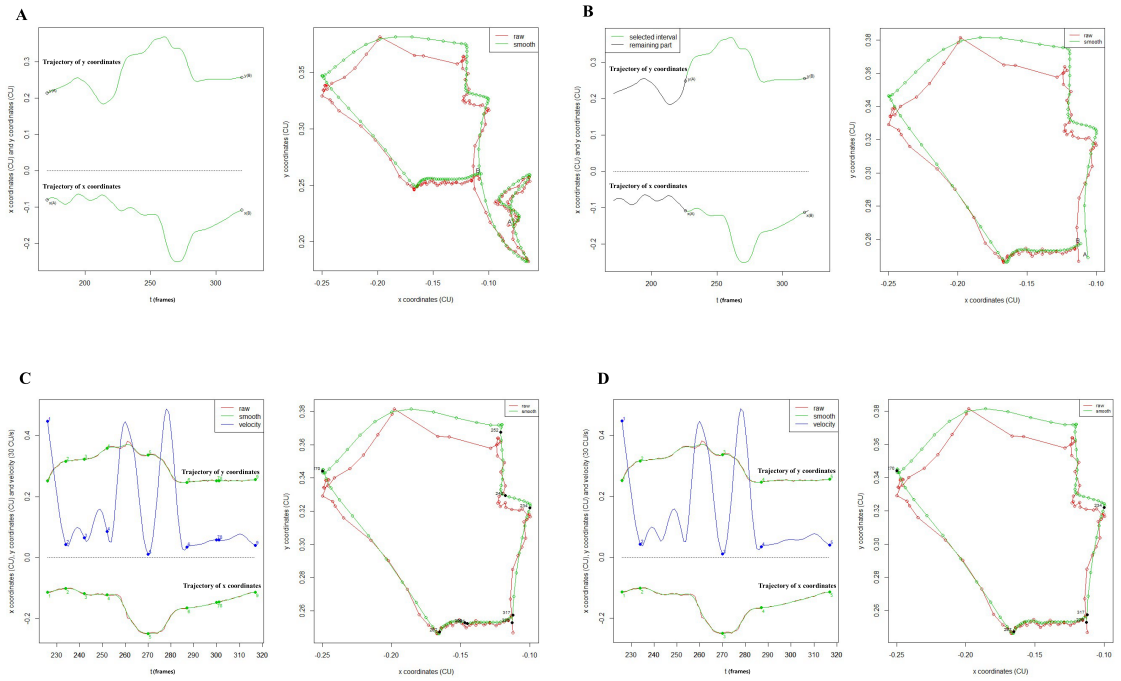



Figure 2.5: Examples of automatic segmentation. A. All the data and the entire trajectory. B. Extracting one circle from the entire trajectory. C. Automatically splitting the trajectory via choosing points satisfying the condition A and B. The numbers 2, 3, 4, 5, 6, 7 and 8 on the curves in the left panel are the candidate splitting points corresponding to the points in black on the 2D trajectory in the right panel. D. Further automatically segmenting the trajectory into four phases via selecting three splitting points from C. The selected points 2, 3 and 4 in the left panel of D are equivalent to the points 2, 5 and 6 in C.

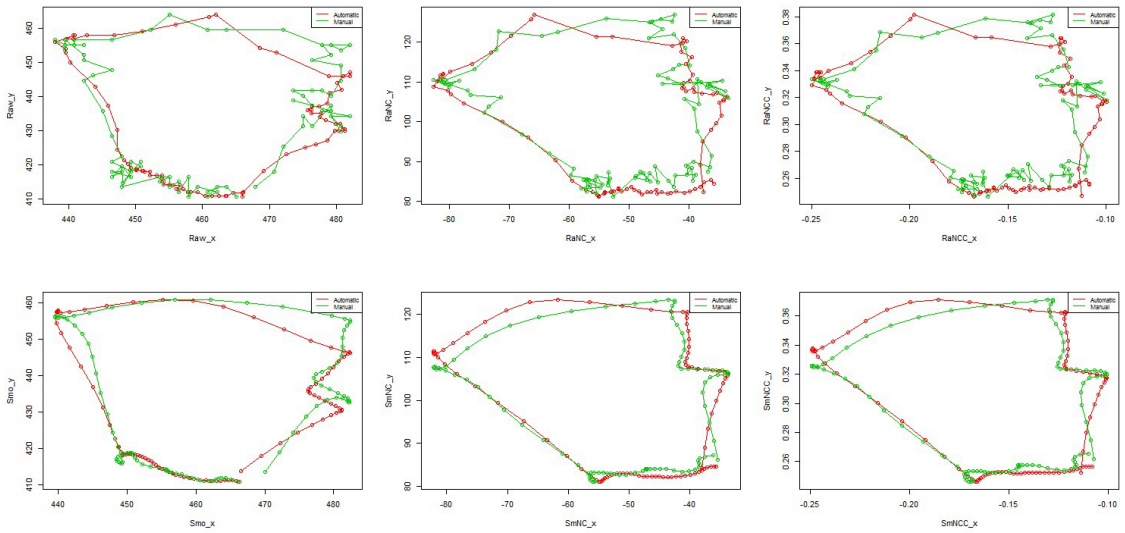



Figure 2.6: An example from unmasked group. The hyoid bone trajectories in red are based on semi-automatic tracking methodology while those in green are by manual method.

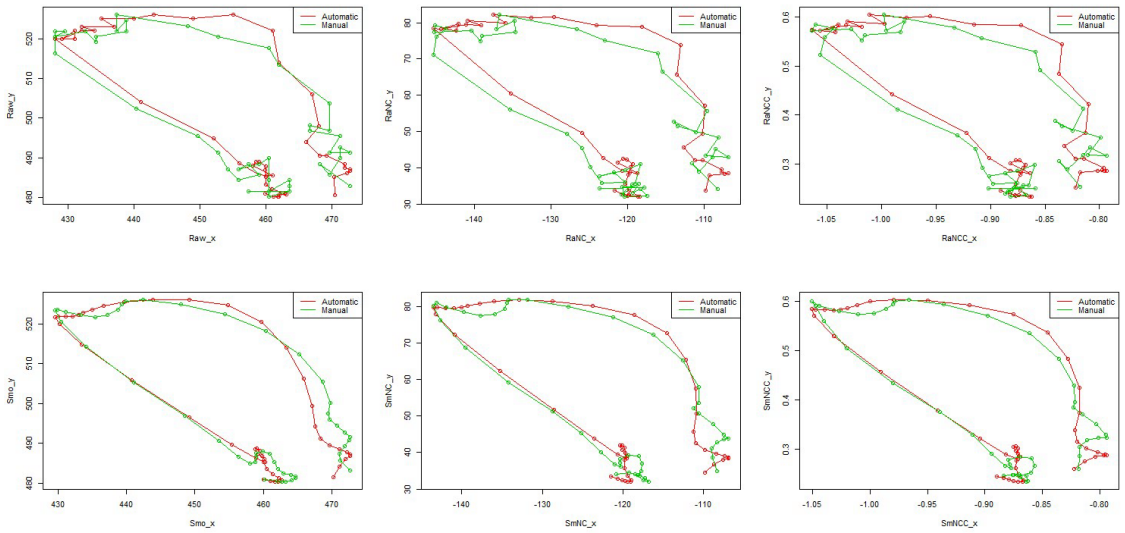



Figure 2.7: An example from masked group. The hyoid bone trajectories in red are based on semi-automatic tracking methodology while those in green are by manual method.

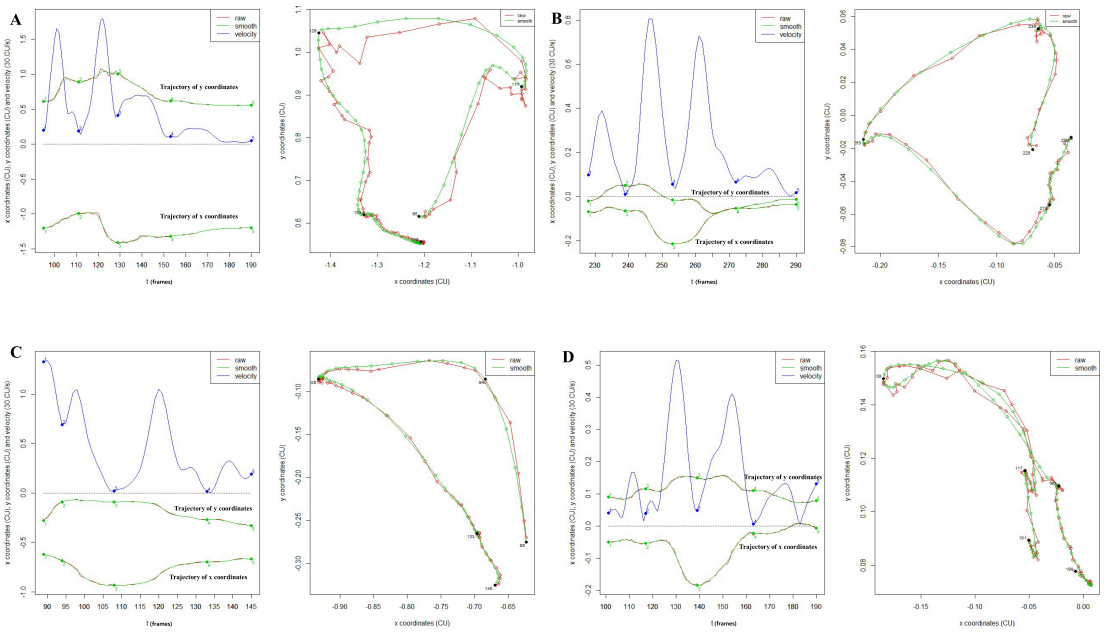



Figure 2.8: Examples of segmentation. A. Successful automatic segmentation to four phases. B. Failed automatic segmentation but successful manual segmentation to four phases. C. Manual segmentation to 3 phases (no returning phase). D. Failed segmentation due to abnormal trajectory.

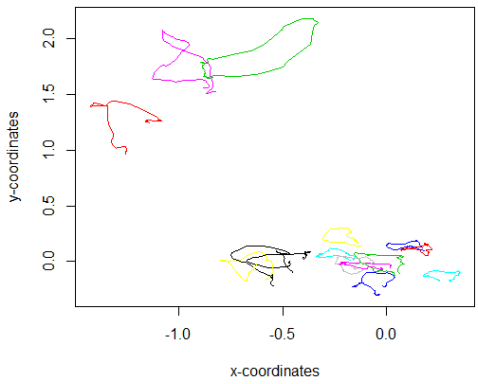
# Chapter 3

# Registration for the Multi-dimensional Functional Data

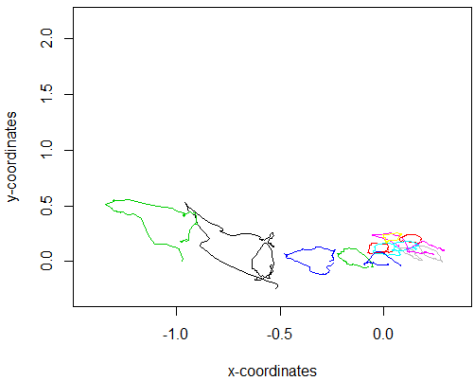
## 3.1 Introduction

Figure 3.1 displays two batches of motion data of hyoid bone from normal people and patients with stroke. They are acquired using the methodology described in Chapter 2. First of all, observing the first two subgraphs (a) and (b) gives us some insight on the spatial registration problems. Obviously, the issues concerning rotation, scaling and shift for those 2D curves need to be dealt with simultaneously. Generalized Procrustes analysis (*GPA*) proposed by Gower (1975a) is a straightforward method for those issues. Secondly, subgraphs (c)-(f) show that there exist temporal registration issues, namely, time warping, which is generally paid much attention in the registration of one dimensional functional data. We have mentioned in Chapter 2 that those splitting points or landmarks are sort of ambiguous and quite hard to identify fully automatically. That means it is not a good choice to directly apply the standard landmark registration method to the data. In this chapter, we attempt to develop a new framework by mixing two methods to address those problems at the same time.

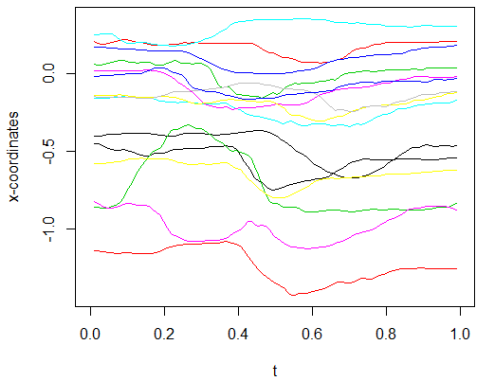
The structure of this chapter is as follows. Section 3.2 reviews the background of Generalized Procrustes analysis and self-modelling registration. A new methodology (*GPSM*) is proposed and the related algorithm is discussed in Section 3.3. Section 3.4 carries out numerical analysis, including the simulation study and real data analysis. Chapter summary is in Section 3.5.



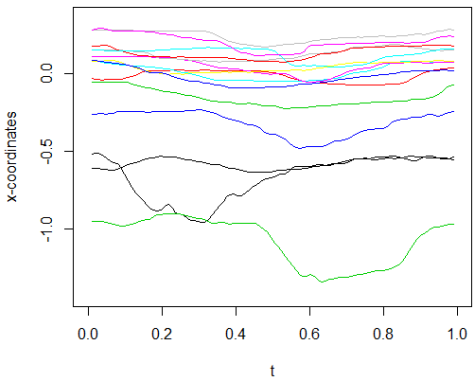

(a) 2D curves from 15 normal people



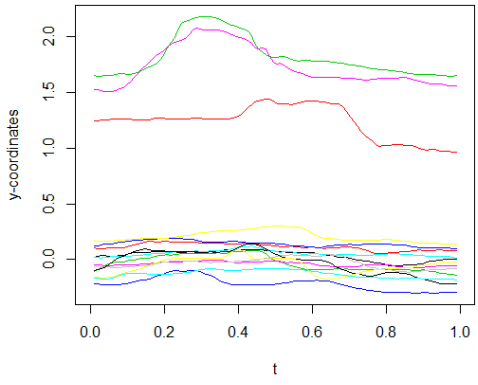

(b) 2D curves from 15 patients



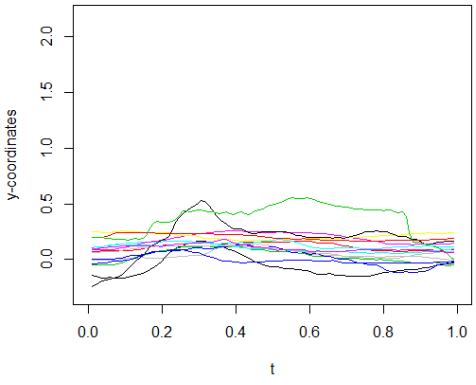

(c) $\boldsymbol{x}_1(t)$ from normal people




(d) $\boldsymbol{x}_1(t)$ from patients




(e) $\boldsymbol{x}_2(t)$ from normal people




(f) $\boldsymbol{x}_2(t)$ from patients

Figure 3.1: 30 samples of the movement of hyoid bone from normal people and patients with stroke. $\boldsymbol{x}_1(t)$ and $\boldsymbol{x}_2(t)$ represent the $x$-coordinates and $y$-coordinates of those 2D curves, respectively.

## 3.2 *GPA* and self-modelling registration

### 3.2.1 Generalized Procrustes analysis

Procrustes analysis is used to analyze the distribution of a set of shapes. In order to compare them, the objects must be optimally *superimposed*, which is carried out by optimally translating, uniformly scaling and rotating the objects. This means both the size of the objects and the placement in space are adjusted. To get a similar size and placement, we minimize a measure of shape difference called the *Procrustes distance* between those objects. Ordinary or classical Procrustes analysis is exploited when a shape is compared to another, or a set of shapes are compared to one arbitrarily selected reference shape.

The shape of an object can be thought of as a member of an equivalent class, which is formed by removing the translational, uniformly scaling and rotational components. For simplicity, we consider objects consisting of $m$ points in 2 dimensions

$$\{(x_{11}, x_{21}), (x_{12}, x_{22}), \ldots, (x_{1m}, x_{2m})\}.$$

The mean of those points is given by

$$(\bar{x}_1, \bar{x}_2), \text{ where } \bar{x}_a = \frac{\sum_{j=1}^{m} x_{aj}}{m}, a = 1, 2,$$

and the scale of the shape is

$$s = \sqrt{\frac{\sum_{j=1}^{m} (x_{1j} - \bar{x}_1)^2 + (x_{2j} - \bar{x}_2)^2}{m}},$$

which is also called root mean square distance, a statistical measure of the object's scale. Procedures of how to remove those components are briefly described as follows:

- Translation. Translate the points

  $$(x_{1j}, x_{2j}) \to (x_{1j} - \bar{x}_1, x_{2j} - \bar{x}_2), \qquad j = 1, \ldots, m,$$

  such that their mean is translated to the origin.

- Uniform scaling. All the points are divided by the object's initial scale

  $$((x_{1j} - \bar{x}_1)/s, (x_{2j} - \bar{x}_2)/s),$$

  so that the scale becomes 1. Note that there are other methods to define the scale in the literature.

- Rotation. Since a standard reference orientation is always unavailable, removing the

rotational component is more complicated. We consider two objects made up from the same number of points

$$\{(x_{11}, x_{21}), (x_{12}, x_{22}), \ldots, (x_{1m}, x_{2m})\} \text{ and } \{(z_{11}, z_{21}), (z_{12}, z_{22}), \ldots, (z_{1m}, z_{2m})\}$$

with scale and translation removed. Fix one of those objects as a reference orientation and rotate the other around the origin until the angle of rotation $\theta$ is found by minimizing the sum of squared distances. A rotation by angle $\theta$ gives

$$(v_{1j}, v_{2j}) = (\cos\theta z_{1j} - \sin\theta z_{2j}, \sin\theta z_{1j} + \cos\theta z_{2j}), \;\; j = 1, 2, \ldots, m,$$

where $(v_{1j}, v_{2j})$ are the coordinates of rotated points. Thus, the optimal angle is

$$\begin{aligned} \hat{\theta} &= \underset{\theta}{\operatorname{argmin}} \sum_{j=1}^{m} (v_{1j} - x_{1j})^2 + (v_{2j} - x_{2j})^2 \\ &= \tan^{-1}\left( \frac{\sum_{j=1}^{m} x_{2j} z_{1j} - x_{1j} z_{2j}}{\sum_{j=1}^{m} x_{1j} z_{1j} + x_{2j} z_{2j}} \right). \end{aligned}$$

If the object is $A$-dimensional, the optimum rotation is represented by an $A \times A$ rotation matrix and the singular value decomposition can be used to find the optimal value.

After *superimposing* the two objects by removing the translational, scaling and rotational components, the difference between the shape of two objects can be assessed by

$$d = \sum_{j=1}^{m} \sqrt{(v_{1j} - x_{1j})^2 + (v_{2j} - x_{2j})^2}.$$

We also call this measure as *Procrustes distance*.

The classical Procrustes analysis aims at superimposing a set of objects to an arbitrarily selected shape while Generalized Procrustes analysis (*GPA*), proposed by Gower (1975a), is mainly for optimally superimposing them. It compares a group of shapes to an optimally determined mean shape. The procedure is outlined as follows:

1. Initialise the reference shape by arbitrarily choosing it among all of the available instances.

2. Superimpose all instances to the current reference shape.

3. Compute the mean shape of the current group of superimposed shapes, as well as the Procrustes distance between the mean and reference shape.

4. Stop the procedure if the Procrustes distance is below a threshold, otherwise set the reference to the mean shape and continue to step 2.

In the real data set, we regard 2D curves as the objects and will apply *GPA* to them to deal with the translational, scaling and rotational components in the preprocessing stage. However, *GPA* is unable to address the warping issues, i.e. the existence of different time scale for each curve. Thus, it is necessary to develop some methods to handle the warping.

### 3.2.2 Self-modelling registration

Self-modelling registration method (*SM*) proposed by Gervini and Gasser (2004) aims to resolve warping problems by introducing a semi-parametric model for one dimensional functional data. They assume the warping function $g^{-1}(t)$ to be linear combinations of $p$ common components, which are estimated combining data across individuals, thereby avoiding over-fitting. They assume that sample curves $\{x_i(t), i = 1, \ldots, N\}$ follow the model

$$x_i(t) = d_i \tau\{g_i(t)\} + \epsilon_i(t), \quad i = 1, \ldots, N, \tag{3.1}$$

where $\{g_i\}$ are monotone increasing functions, $\tau$ is the structural mean and $\epsilon_i$ are random errors. The functions $\{g_i\}$ are seen as one kind of the inverse of warping functions. Assume $d_i \neq 0$, $E(d) = 1$, $E(g^{-1}(t)) = t$ and $E(\epsilon) = 0$. This model is a working model and allows rather limited type of amplitude variability, but it performs well in a real data set.

The warping functions proposed corresponding to the above model is

$$g_i^{-1}(t) = t + \sum_{j=1}^{p} w_{ij}\phi_j(t), \quad i = 1, \ldots, N, \tag{3.2}$$

where $\phi_j(t) = \boldsymbol{e}_j^{\intercal}\boldsymbol{\xi}(t)$, where $\boldsymbol{\xi}(t) = (\xi_1(t), \xi_2(t), \ldots, \xi_q(t))^{\intercal}$ is a vector of B-spline basis functions and the score vectors $\boldsymbol{w}_i = (w_{i1}, \ldots, w_{ip})^{\intercal}$ satisfy $E(\boldsymbol{w}) = \boldsymbol{0}$. These $\boldsymbol{\phi}-$functions are all localized non-negative bell-shaped functions, each of which accounting for time variability at different segments of $T$. For identifiability, the spline coefficients must satisfy three restrictions:

(A) $e_{jk} \geq 0$ for $k = 1, \ldots, q$ and $||\boldsymbol{e}_j|| = 1$ for $j = 1, \ldots, p$.

(B) The coefficient matrix $\boldsymbol{E} = (e_{jk}) \in R^{p \times q}$ has the block structure $1 \leq K_1 < K_2 < \cdots < K_{p+1} \leq q + 1$ such that $e_{jk} > 0$ for $K_j \leq k < K_{j+1}$ and $e_{jk} = 0$ for $k < K_j$ and $k \geq K_{j+1}$.

(C) $e_{j1} = e_{jq} = 0$ for all $j$, i.e. $K_1 = 2$ and $K_{p+1} = q$ such that $K_2 \geq 3$ and $K_p \leq q - 1$.

Condition (B) ensures that the components have connected and localized supports. Restriction (C) guarantees that $g_i^{-1}(a) = a$ and $g_i^{-1}(b) = b$ when the time interval is $[a, b]$. The proof on the identifiability of model (3.1) and model (3.2) are provided by Gervini and Gasser (2004).

Each $\phi_j$ can actually be regarded as a component associated with a *hidden landmark*, which motivates the model (3.2) to some extent. Take the case of two landmarks per 1D curve, $l_{i1}$ and $l_{i2}$, for example. Let $l_{01} = \bar{l}_{\cdot 1}$ and $l_{02} = \bar{l}_{\cdot 2}$ be the average landmarks. From the registered curves $\tilde{x}_i(t) = x_i(g^{-1}(t))$, we know that $\tilde{x}_i(l_{0j}) = x_i(l_{ij}), j = 1, 2$, for all $i$. Thus, the warping functions must satisfy $g_i^{-1}(a) = a, g_i^{-1}(l_{01}) = l_{i1}, g_i^{-1}(l_{02}) = l_{i2}$ and $g_i^{-1}(b) = b$. Using the simplest interpolation method, i.e. piecewise linear functions, we get

$$g_i^{-1}(t) = \begin{cases} t + (l_{i1} - l_{01})\frac{t-a}{l_{01}-a} & t \in [a, l_{01}], \\ t + (l_{i1} - l_{01})\frac{l_{02}-t}{l_{02}-l_{01}} + (l_{i2} - l_{02})\frac{t-l_{01}}{l_{02}-l_{01}} & t \in [l_{01}, l_{02}], \\ t + (l_{i2} - l_{02})\frac{b-t}{b-l_{02}} & t \in [l_{02}, b]. \end{cases}$$

Let $w_{ij} = l_{ij} - l_{0j}$ and

$$\phi_1(t) = \begin{cases} \frac{t-a}{l_{01}-a} & t \in [a, l_{01}], \\ \frac{l_{02}-t}{l_{02}-l_{01}} & t \in [l_{01}, l_{02}], \\ 0 & t \in [l_{02}, b]. \end{cases}$$

$$\phi_2(t) = \begin{cases} 0 & t \in [a, l_{01}], \\ \frac{t-l_{01}}{l_{02}-l_{01}} & t \in [l_{01}, l_{02}], \\ \frac{b-t}{b-l_{02}} & t \in [l_{02}, b]. \end{cases}$$

We can write $g_i^{-1}(t) = t + \sum_{j=1}^{2} w_{ij}\phi_j(t)$. In other words, those triangles with peaks at $l_{01}$ and $l_{02}$ can be expressed as combinations of linear $B$-splines with knots $\{a, l_{01}, l_{02}, b\}$. Therefore, each component in the model (3.2) is associated with an underlying landmarks. The self-registration method is used to estimate the associated components instead of the individual landmarks.

To estimate the parameters in models (3.1) and (3.2), we minimize the average integrated squared error given by

$$\begin{aligned} AISE_N &= \frac{1}{N}\sum_{i=1}^{N}\int_a^b ||x_i(t) - d_i\tau\{g_i(t)\}||^2 dt \\ &= \frac{1}{N}\sum_{i=1}^{N}\int_a^b ||x_i(g_i^{-1}(t)) - d_i\tau(t)||^2 (g_i^{-1})'(t) dt. \end{aligned} \tag{3.3}$$

Thus, we have the estimator of the structural mean

$$\hat{\tau}(t) = \frac{\sum_{i=1}^{N} \hat{d}_i(\hat{g}_i^{-1})'(t)x_i(\hat{g}_i^{-1})(t)}{\sum_{i=1}^{N} \hat{d}_i^2(\hat{g}_i^{-1})'(t)}. \tag{3.4}$$

However, there are no explicit estimating equations for $\hat{E}$ or the $\hat{g}_i^{-1}$s. Two-stage algorithms for minimizing equation (3.3) have been implemented by Gervini and Gasser

(2004):

- Stage 1: initialization.

  (a) Select desirable block delimiters $\boldsymbol{K} = (K_1, \ldots, K_{p+1})$ for the coefficient matrix $\hat{E}$ and re-parameterize

  $$(e_{j,K_j}, \ldots, e_{j,K_{j+1}-1}) = (1, \exp(\boldsymbol{u}_j))/\{1 + ||\exp(\boldsymbol{u}_j)||^2\}^{1/2},$$

  where $\boldsymbol{u}_j \in R^{K_{j+1}-K_j}$ are unconstrained vectors.

  (b) Set $\hat{\boldsymbol{u}}_j = 0$, $\hat{\boldsymbol{w}}_i = 0$, $\hat{d}_i = 1$ and $\hat{\tau}(t) = \bar{x}(t)$.

- Stage 2: iterations.

  (a) Update $g_i^{-1}(t)$. Update $\boldsymbol{u}_j$ by using a Newton-Raphson step and recentre the current $\hat{\boldsymbol{w}}_i$s so that $\bar{\hat{\boldsymbol{w}}} = 0$. Then, update $\hat{\boldsymbol{w}}_i$ by using a Newton-Raphson step.

  (b) Update $\hat{\tau}(t)$ and $\hat{d}_i$. Update $\hat{\tau}$ by using equation (3.4), compute $x_i(g_i^{-1}(t))$ by linear interpolation.

  (c) Update $\hat{d}_i$ and update the objective function (3.3). Exit if there is no significant improvement; otherwise go back to (a).

On one hand, the advantage of this semi-parametric model for warping functions of one-dimensional random curves over landmark registration is that there is no need to identify individual landmarks. Also, it sufficiently makes use of data and avoids over-fitting to a large degree by using the common structure of the warping functions. On the other hand, this model lies in one strong assumption on the existence of *hidden landmarks*. For our real data in Figure 3.1, there are some recognizable landmarks hiding in each 2D curve, such as the *turning points* or *splitting points* aforementioned in Section 2.3.4 of Chapter 2. Thus, we will exploit this model in our registration methodology for the multi-dimensional functional data.

## 3.3 The methodology and algorithm

We attempt to integrate the *GPA* and self-modelling registration in order to deal with the multi-dimensional functional data. For 2D curves $\{\boldsymbol{x}_i(t), t \in [0, L]; i = 1, \ldots, N\}$, where $t$ is usually transformed to the unit of arc-length in the case of more than one dimensions, assume $T_i$ is the transformation in terms of translation, rotation and scaling for curve $i$. We can regard $T_i$ as the parametric matrix to be estimated. The procedures of *GPA* for $\{\boldsymbol{x}_i, i = 1, \ldots, N\}$ can be outlined as:

(a) Initialise $\boldsymbol{\mu}^{(0)}(t)$, the reference curve, as $\boldsymbol{x}_0(t)$, which is arbitrarily chosen from $\{\boldsymbol{x}_i(t), t \in [0, L]; i = 1, \ldots, N\}$.

(b) For the $(i_0 + 1)$th iteration, superimpose all the curves to $\boldsymbol{\mu}^{(i_0)}(t)$ by

$$T_i^{(i_0+1)} = \underset{T}{\operatorname{argmin}} \int_0^L ||T_i(\boldsymbol{x}_i)(t) - \boldsymbol{\mu}^{(i_0)}(t)||^2 dt, \qquad i = 1, 2, \ldots, N. \tag{3.5}$$

(c) Compute the Procrustes distance

$$D^{(i_0+1)} = \int_0^L ||\frac{1}{N} \sum_{i=1}^N T_i^{(i_0+1)}(\boldsymbol{x}_i)(t) - \boldsymbol{\mu}^{(i_0)}(t)||^2 dt.$$

(d) If $|D^{(i_0+1)} - D^{(i_0)}| < \delta$, where $\delta$ is predetermined as 0.01, the iteration ends; otherwise set $\boldsymbol{\mu}^{(i_0+1)}(t) = \frac{1}{N} \sum_{i=1}^N T_i^{(i_0+1)}(\boldsymbol{x}_i)(t)$ and continue to step 2.

Suppose $k$ iterations are required to reach below the threshold, then $\hat{T}_i = T_i^{(k)} \cdot T_i^{(k-1)} \cdots T_i^{(1)}, i = 1, 2, \ldots, N$.

As for the warping issue, we try to extend the self-modeling registration aforementioned from one-dimension to multi-dimensions. Assume there are $A$ dimensions, the sample curves in $a$-th dimension are assumed to follow the model

$$x_{ai}(t) = d_{ai}\tau_a\{g_i(t)\} + \epsilon_{ai}(t), \quad t \in L_i \subset R, \quad i = 1, \ldots, N, \quad a = 1, \ldots, A. \tag{3.6}$$

The warping functions and the related restrictions are the same as the one-dimensional case mentioned in Section 3.2. The average integrated squared error is given by

$$E_N = \frac{1}{N} \sum_{i=1}^N \int_0^L ||\boldsymbol{x}_i(t) - \boldsymbol{d}_i\boldsymbol{\tau}\{g_i(t)\}||^2 dt. \tag{3.7}$$

The techniques of estimating the parameters in the model (3.6) are similar to the case of one-dimensional curves. After getting the estimated warping functions, the curves are required to be updated to a new time scale $\hat{g}^{-1}(t)$ and iteratively apply the *GPA* to them. We call the combination of *GPA* and self-modelling registration as *GPSM* methodology. The outline of the algorithm for *GPSM* is as follows (assuming *GPA* is used first and followed by *SM*):

1. Initialise $\boldsymbol{\mu}^{(0)}(t)$, the reference curve, as the cross-sectional mean of functional data, i.e.

$$\boldsymbol{\mu}^{(0)}(t) = \frac{1}{N} \sum_{i=1}^N \boldsymbol{x}_i(t).$$

2. For the $(i_0+1)$th iteration, superimpose all the curves to $\boldsymbol{\mu}^{(i_0)}(t)$ by

$$T_i^{(i_0+1)} = \underset{T}{\operatorname{argmin}} \int_0^L ||T_i(\boldsymbol{x}_i)(t) - \boldsymbol{\mu}^{(i_0)}(t)||^2 dt, \qquad i = 1, 2, \ldots, N.$$

3. Compute the Procrustes distance

$$D^{(i_0+1)} = \int_0^L ||\frac{1}{N}\sum_{i=1}^N T_i^{(i_0+1)}(\boldsymbol{x}_i)(t) - \boldsymbol{\mu}^{(i_0)}(t)||^2 dt.$$

4. If $|D^{(i_0+1)} - D^{(i_0)}| < \delta$, where $\delta$ is predetermined as 0.01, continue to step 5; otherwise, set

$$\boldsymbol{\mu}^{(i_0+1)}(t) = \frac{1}{N}\sum_{i=1}^N T_i^{(i_0+1)}(\boldsymbol{x}_i)(t),$$

and continue to step 2.

5. Suppose $k$ iterations are required to reach below the threshold, then the $\hat{T}_i = T_i^{(k)} \cdot T_i^{(k-1)} \cdots T_i^{(1)}, i = 1, 2, \ldots, N$. Calculate the synchronization coefficient $sync1$ defined by James (2007), see the details in Section 3.4.2.

6. Compute the warping function by $SM$ method, let $\hat{\boldsymbol{x}}_i(t) = \hat{T}_i(\boldsymbol{x}_i)(t)$,

$$g_i^{-1}(t) = \operatorname{argmin} \frac{1}{N}\sum_{i=1}^N \int_0^L [\hat{\boldsymbol{x}}_i(t) - \boldsymbol{d}_i\boldsymbol{\tau}\{g_i(t)\}]^2 dt, \qquad i = 1, 2, \ldots, N. \tag{3.8}$$

7. Calculate the $sync2$. If $sync2 < sync1$, update $\boldsymbol{x}_i(t) = \hat{\boldsymbol{x}}_i(g_i^{-1}(t))$ and $\boldsymbol{\mu}^{(i_0+k)}(t) = \boldsymbol{\tau}(t)$, where $\boldsymbol{\tau}(t) \approx \frac{1}{N}\sum_{i=1}^N \boldsymbol{x}_i(t)$, and continue to step 2. Otherwise, stop.

## 3.4 Numerical analyses

### 3.4.1 Data generation

We now compare the methodology ($GPSM$) with the Generalized Procrustes analysis ($GPA$) (Gower, 1975a), the extension of self-modelling registration ($SM$) (Gervini and Gasser, 2004) and square-root velocity method ($SRV$) (Srivastava et al., 2011a) in terms of both alignment and estimation. To generate data, we first generate one 2D curve $\mu(t) = \{(x_1(t), x_2(t)), t \in [0, L]\}$ selected randomly from hyoid bone motions data as the true structural mean. We use equidistant points $t_j = (j-1)/(m-1)$, where $j = 1, \ldots, m$, as input grid. Then we generate data as follows:

(a) Warping. Two natural landmarks, $lm1$ and $lm2$, are identified manually. Normally

the peak (maximum) and valley (minimum) of the structural mean along the $x$-axis or $y$-axis can be regarded as landmarks. Then two random landmarks are generated for each 2D curve. $lm_{ik} = lm_k + \delta_{ik}, k = 1, 2$ with $\delta_{ik} = \min\{\max(U_{ik}, -V), V\}$, $U_{ik}$ independent $N(0, \sigma_w^2)$ random variables, and $V = \frac{1}{3}\min\{lm_1, lm_2 - lm_1, L - lm_2\}$ (the truncation makes sure that $0 < lm_{i1} < lm_{i2} < L$). The inverse warping functions $g_i(t)$ are piecewise linear with $g_i(0) = 0, g_i(lm_{i1}) = lm_1, g_i(lm_{i2}) = lm_2$ and $g_i(L) = L$.

(b) Rotation. Each 2D curve is rotated around the center $\left(\frac{1}{m}\sum_{j=1}^m x_1(t_j), \frac{1}{m}\sum_{j=1}^m x_2(t_j)\right)$ with the angle $\theta_i = -\theta_0 + 2\theta_0 * W_i$ and $W_i$ independent $U(0, 1)$.

(c) Scaling. Each 2D curve is scaled with the same scaling factor $a_i$, which are independent and identically distributed $N(1, \sigma_a^2)$ random variables with $\sigma_a = 0.1$.

(d) Translating and adding random errors. The newly generated 2D curve $(x_1'(t_j), x_2'(t_j))$ will end up adding random translation and errors in the following way: $\big(x_1'(t_j) + \mathrm{M1}_i + \mathrm{N1}_{ij}, x_2'(t_j) + \mathrm{M2}_i + \mathrm{N2}_{ij}\big)$, where $\mathrm{M1}_i, \mathrm{M2}_i$ are independent $N(0, 0.1^2)$ random variables and $\mathrm{N1}_i, \mathrm{N2}_i$ independent $N(0, 0.01^2)$ random variables.

(e) In each dataset, sample size for the 2D curves $N = 30$ and grid size $m = 100$ are used, with $r = 50$ replications for each combination. Three scenarios are examined for each method. These are the following: (1) Scenario A: $\sigma_w = 0.1$ and $\theta_0 = \pi/4$; (2) Scenario B: $\sigma_w = 0.5$ and $\theta_0 = \pi/6$; (3) Scenario C: $\sigma_w = 1$ and $\theta_0 = \pi/8$. ($\sigma_w$ and $\theta_0$ control the warping and rotating intensity, respectively)

Three different mean curves are selected, which results in generating three types of data sets in each scenario. Figure 3.2 shows typical examples of one realization from Dataset 3 with three scenarios. The sub-figures (b)-(d) contains problems of rotation, translation, scaling and warping.

### 3.4.2 Measurements

To measure the estimation error for the structural mean, we use the root average squared error (James, 2007)

$$rase(\hat{\boldsymbol{\mu}}) = \sqrt{\frac{\sum_{j=1}^m ||\hat{\boldsymbol{\mu}}(t_j) - \boldsymbol{\mu}(t_j)||^2}{m}},$$

where $m$ is the number of observation points, $\boldsymbol{\mu}(t)$ and $\hat{\boldsymbol{\mu}}(t)$ are the cross-sectional mean of raw curves and of registered curves for each replication in each dataset.

Two other criteria are used to evaluate the registration comparison. We denote by $\boldsymbol{x}_i$ and $\hat{\boldsymbol{x}}_i$, $(i = 1, \ldots, N)$ the raw and the registered 2D curves, respectively. The synchro-

nization coefficient defined by James (2007)

$$sync = \frac{1}{N}\sum_{i=1}^{N}\frac{\int ||\hat{\boldsymbol{x}}_i(t) - \frac{1}{N-1}\sum_{j\neq i}\hat{\boldsymbol{x}}_j(t)||^2 dt}{\int ||\boldsymbol{x}_i(t) - \frac{1}{N-1}\sum_{j\neq i}\boldsymbol{x}_j(t)||^2 dt},$$

measures the overall cross-sectional variance of the registered curves relative the original curves. The smaller the value of *sync*, the better the registration is. The inverse of pairwise correlation between curves is defined as

$$ipc = \frac{\sum_{i\neq j}\text{corr2}(\boldsymbol{x}_i(t), \boldsymbol{x}_j(t))}{\sum_{i\neq j}\text{corr2}(\hat{\boldsymbol{x}}_i(t), \hat{\boldsymbol{x}}_j(t))},$$

where $\text{corr2}(A, B)$ computes the correlation coefficient using

$$R = \frac{\sum_m\sum_N(A_{mN} - \bar{A})(B_{mN} - \bar{B})}{\sqrt{(\sum_m\sum_N(A_{mN} - \bar{A})^2)((\sum_m\sum_N(B_{mN} - \bar{B})^2)}}.$$

Smaller values of *ipc* indicate better registration.

### 3.4.3 Results

All of the measures are averaged over $r = 50$ Monte Carlo simulations. Quantitatively, *GPSM* outperforms *GPA*, *SM* and the *SRV* method, in any of the three scenarios, as we can see from the Table 3.1. It is clear that the performance of *GPA* becomes worse as the $\sigma_w$ increases and $\theta_0$ decreases while *SM* becomes better, particularly in terms of *rase*. This is because *GPA* is specialized at the rotation issue while *SM* works mainly for warping. Overall, *GPSM* considers both rotation and warping issues, leading to much better performance than *GPA* and *SM*. Figure 3.3 to 3.5 display three examples of registration results by four methods in Dataset 3, corresponding to Scenario A to C. All of these demonstrate better registration by *GPSM*. More examples are given by Figures A.1 to A.8 in Appendix A.

### 3.4.4 Real data analysis

Our application to the real data, shown in Figure 3.1, is to do registration by the four methods *GPSM*, *GPA*, *SM* and *SRV*. Table 3.2 shows the *GPSM* is better than the other three methods in terms of registration. Figures 3.6 and 3.7 also justify its superiority.

After doing registration, we can do classification. Here, we use a simple rule similar to the $k$-means algorithm. The group means are firstly obtained via any of the 2D registration methods. Assume the group mean of training data is $\boldsymbol{\mu}_k(t)$ and $\boldsymbol{\mu}_k^+(t)$ is the mean after the addition of the test curve $\boldsymbol{x}^*$. Each test curve is then assigned to the trained group

| | | Dataset 1 | | | Dataset 2 | | | Dataset 3 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | *rase* | *sync* | *ipc* | *rase* | *sync* | *ipc* | *rase* | *sync* | *ipc* |
| *GPSM* | $\sigma_w = 0.1, \theta_0 = \pi/4$ | **0.07** | **0.02** | **0.80** | **0.08** | **0.02** | **0.80** | **0.06** | **0.03** | **0.80** |
| | $\sigma_w = 0.5, \theta_0 = \pi/6$ | **0.07** | **0.19** | **0.89** | **0.07** | **0.18** | **0.88** | **0.07** | **0.16** | **0.87** |
| | $\sigma_w = 1, \theta_0 = \pi/8$ | **0.09** | **0.27** | **0.90** | **0.08** | **0.19** | **0.89** | **0.07** | **0.18** | **0.89** |
| *GPA* | $\sigma_w = 0.1, \theta_0 = \pi/4$ | 0.07 | 0.03 | 0.80 | 0.08 | 0.03 | 0.80 | 0.06 | 0.05 | 0.81 |
| | $\sigma_w = 0.5, \theta_0 = \pi/6$ | 0.09 | 0.29 | 0.90 | 0.08 | 0.28 | 0.89 | 0.08 | 0.29 | 0.89 |
| | $\sigma_w = 1, \theta_0 = \pi/8$ | 0.12 | 0.45 | 0.92 | 0.10 | 0.40 | 0.92 | 0.09 | 0.38 | 0.91 |
| *SM* | $\sigma_w = 0.1, \theta_0 = \pi/4$ | 0.16 | 0.61 | 0.87 | 0.13 | 0.55 | 0.86 | 0.15 | 0.56 | 0.86 |
| | $\sigma_w = 0.5, \theta_0 = \pi/6$ | 0.10 | 0.60 | 0.92 | 0.10 | 0.57 | 0.91 | 0.10 | 0.51 | 0.91 |
| | $\sigma_w = 1, \theta_0 = \pi/8$ | 0.10 | 0.61 | 0.93 | 0.10 | 0.54 | 0.92 | 0.09 | 0.48 | 0.91 |
| *SRV* | $\sigma_w = 0.1, \theta_0 = \pi/4$ | 0.16 | 0.15 | 0.81 | 0.40 | 0.18 | 0.81 | 0.12 | 0.14 | 0.81 |
| | $\sigma_w = 0.5, \theta_0 = \pi/6$ | 0.13 | 0.41 | 0.90 | 0.44 | 0.29 | 0.88 | 0.12 | 0.27 | 0.88 |
| | $\sigma_w = 1, \theta_0 = \pi/8$ | 0.13 | 0.60 | 0.93 | 0.48 | 0.37 | 0.90 | 0.12 | 0.34 | 0.90 |

Table 3.1: The average results of estimation and registration by four methods. The **bold** numbers indicate the best results.

.

| | Normal | | Abnormal | |
|---|---|---|---|---|
| | *sync* | *ipc* | *sync* | *ipc* |
| *GPSM* | **0.46** | **0.68** | **0.25** | **0.68** |
| *GPA* | 0.67 | 0.76 | 0.41 | 0.73 |
| *SM* | 0.76 | 0.84 | 0.55 | 0.78 |
| *SRV* | 0.89 | 0.76 | 0.64 | 0.75 |

Table 3.2: 2D registration results based on 15 curves of hyoid bone motion in normal and abnormal group respectively.

which is closest in terms of Procrustes distance between means. In other words, $\boldsymbol{x}^*$ is classified as belong to $k^*$-th group if $d(k) = \int_0^L |\boldsymbol{\mu}_k(t) - \boldsymbol{\mu}_k^+(t)|dt$ takes its minimum value at $k = k^*$ for $k = 1, \ldots, K$. For our real dataset, $K$ is set as 2.

We evaluate the classification performance by 5-fold cross validation. Apart from the classification accuracy (CA), we use another two criteria for evaluating the classification results. The first one is the Rand index (RI) (Rand, 1971a), having a value between 0 and 1, with 0 indicating two data clusterings disagree on any pair while 1 indicating a perfect match. And the second one is called adjusted Rand index (ARI) (Hubert and Arabie, 1985a), a modified version of Rand index (ARI). A larger value of RI or ARI indicates a higher agreement of the method and the truth. The average classification results for those methods are shown in Table 3.3. It shows that all of the methods fail though *GPSM* seems slightly better than others. This motivates us to come up with some other methods to carry out prediction for this real dataset.

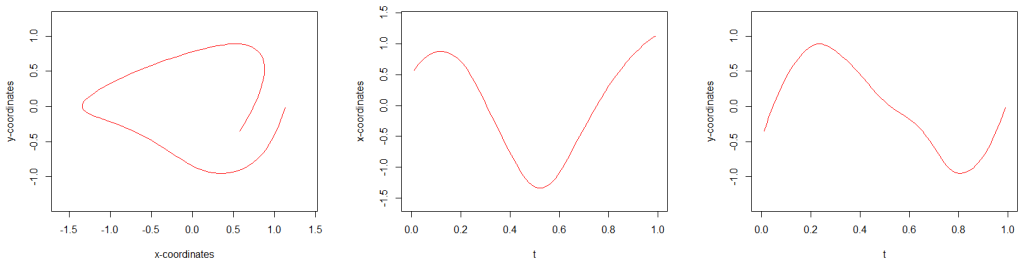
| | CA | RI | ARI |
|---|---|---|---|
| *GPSM* | **0.57** | 0.48 | 0.06 |
| *GPA* | 0.37 | 0.53 | 0.09 |
| *SM* | 0.37 | 0.53 | 0.13 |
| *SRV* | 0.50 | 0.43 | -0.11 |

Table 3.3: Average classification results of three measurements by four methods.

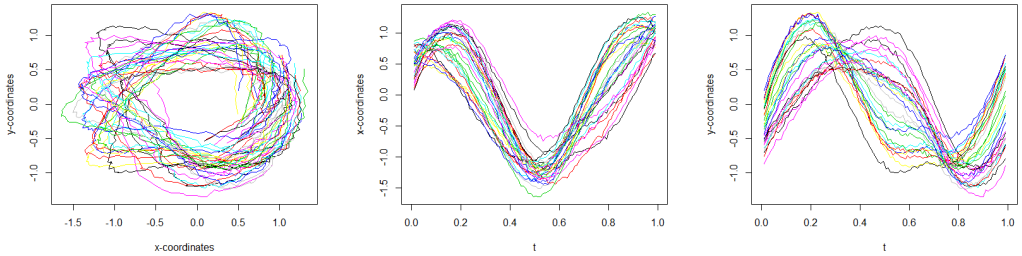
## 3.5 Chapter Summary

In this chapter we propose a new registration method (*GPSM*) for multi-dimensional functional data. It integrates the Generalized Procrustes analysis (*GPA*) and self-modelling registration (*SM*) for the sake of addressing both spatial and temperal registration issues, namely, rotation, shift, scaling and time warping. It outperforms the other methods as we see from the numerical results.

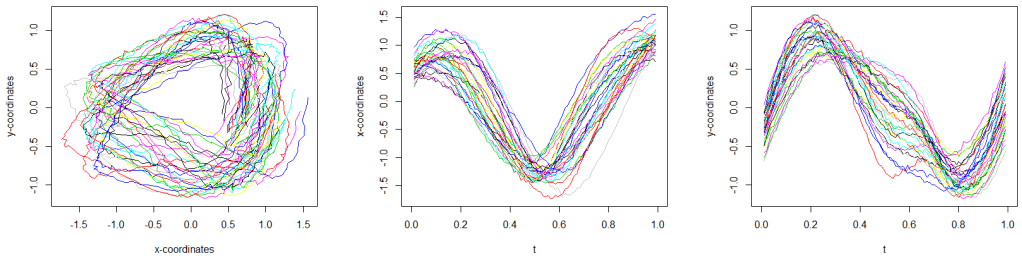
However, this framework generally belongs to a kind of standard preprocessing method for functional data analysis. It is usually conducted prior to modelling or classification. These two separate steps are sometimes inconvenient and time consuming. Furthermore, it does not work well when it comes to the prediction in some cases, for example, in our real dataset, as shown in Table 3.3. Thus, we need to explore another methodology which is capable of simultaneously carrying out registration and modelling, as well as considering some other scalar variables. This is the aim of next two chapters.



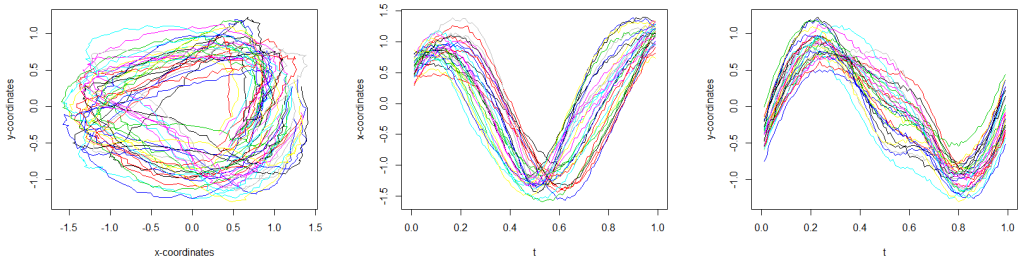

(a) 2D reference curve of Dataset 3




(b) Scenario A: $\sigma_w = 0.1$ and $\theta_0 = \pi/4$

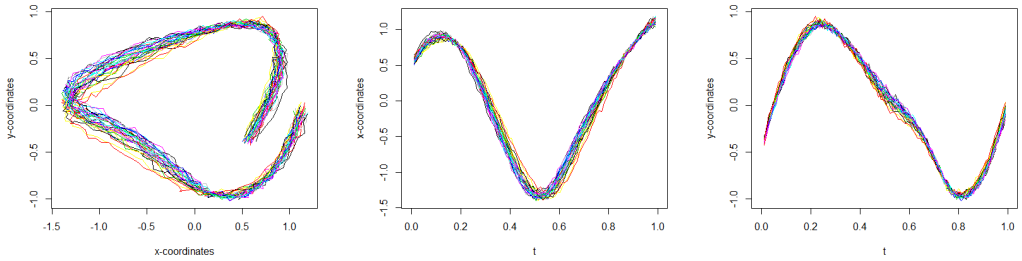



(c) Scenario B: $\sigma_w = 0.5$ and $\theta_0 = \pi/6$

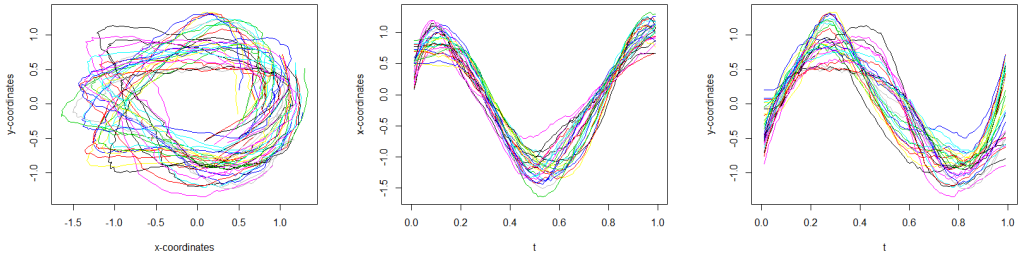



(d) Scenario C: $\sigma_w = 1$ and $\theta_0 = \pi/8$

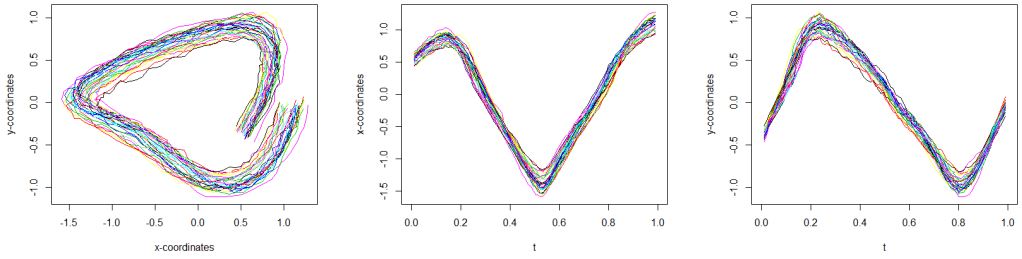
Figure 3.2: Three examples of data in Dataset 3 corresponding to three scenarios.



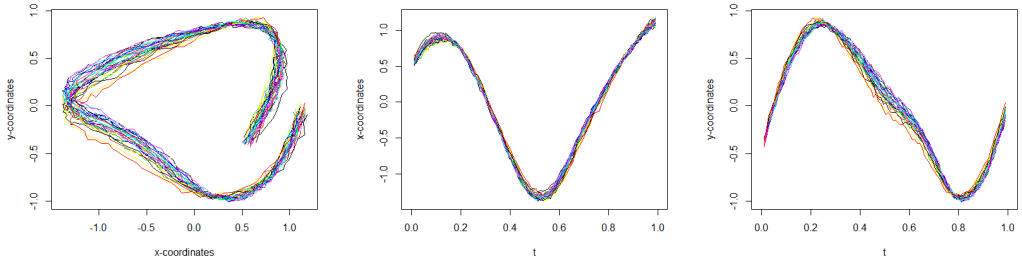

(a) Registration by *GPA*




(b) Registration by *SM*

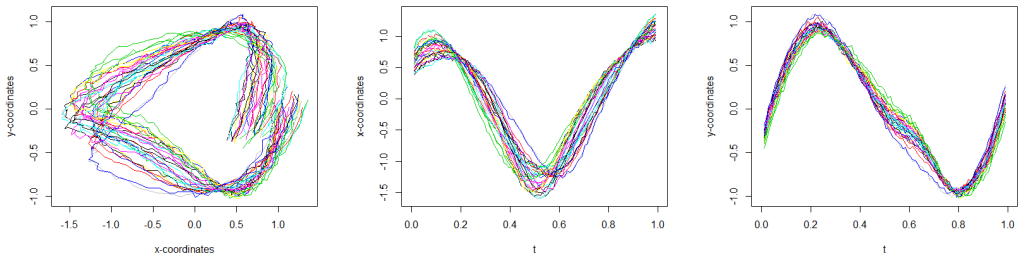



(c) Registration by *SRV*

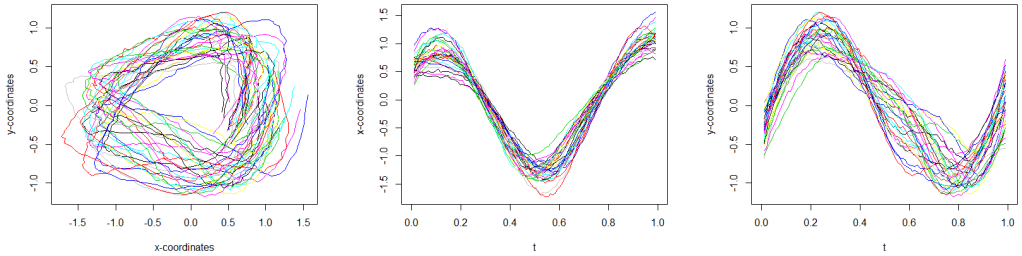



(d) Registration by *GPSM*

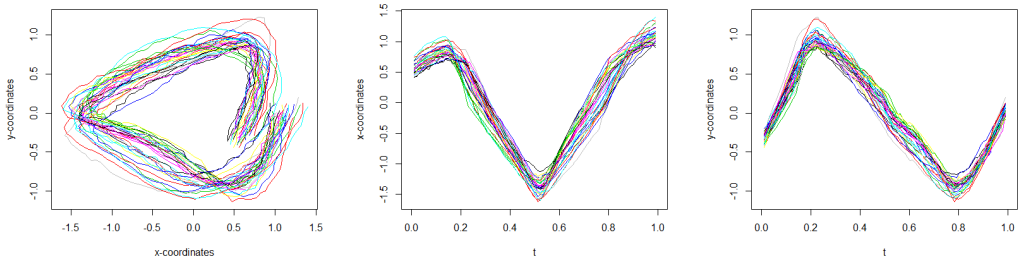
Figure 3.3: An example of registration results in Dataset 3 by four methods for Scenario A with $\sigma_w = 0.1$ and $\theta_0 = \pi/8$.



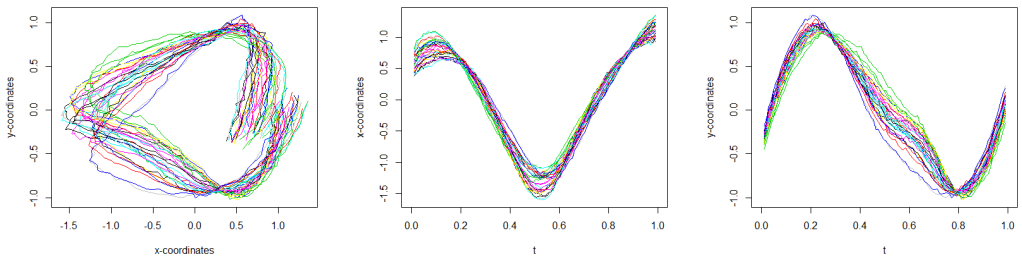

(a) Registration by *GPA*




(b) Registration by *SM*

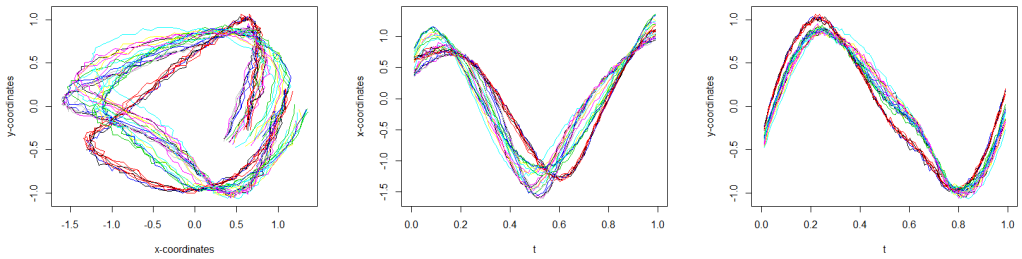



(c) Registration by *SRV*

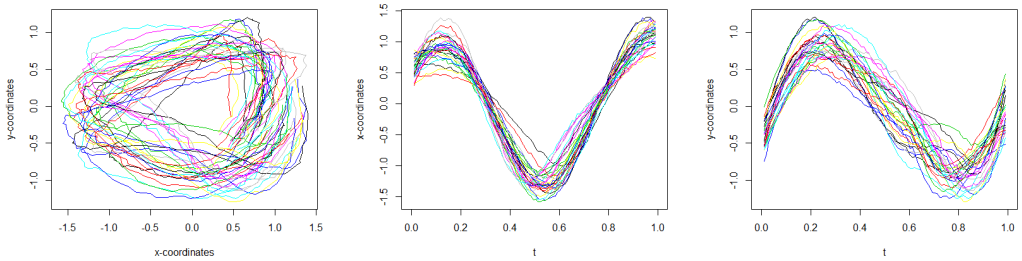



(d) Registration by *GPSM*

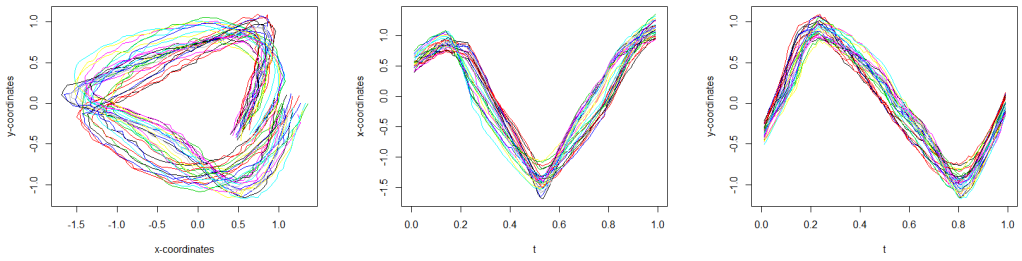
Figure 3.4: An example of registration results in Dataset 3 by four methods for Scenario B with $\sigma_w = 0.5$ and $\theta_0 = \pi/6$.



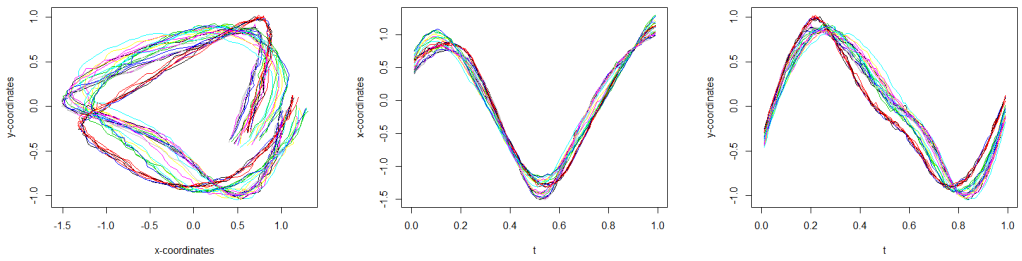

(a) Registration by *GPA*




(b) Registration by *SM*

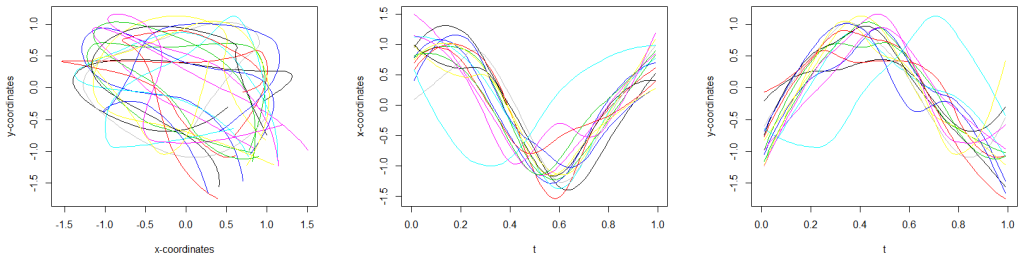



(c) Registration by *SRV*

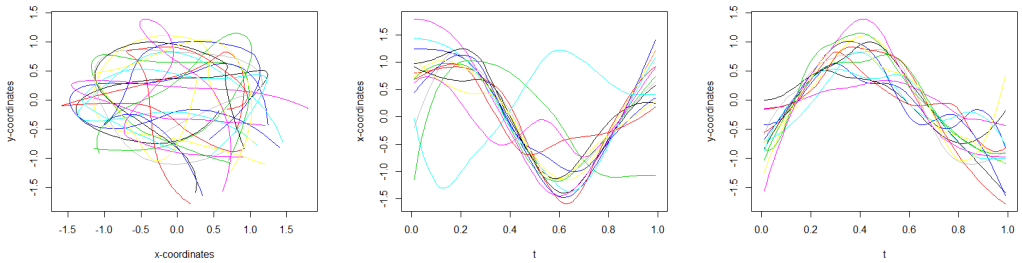



(d) Registration by *GPSM*

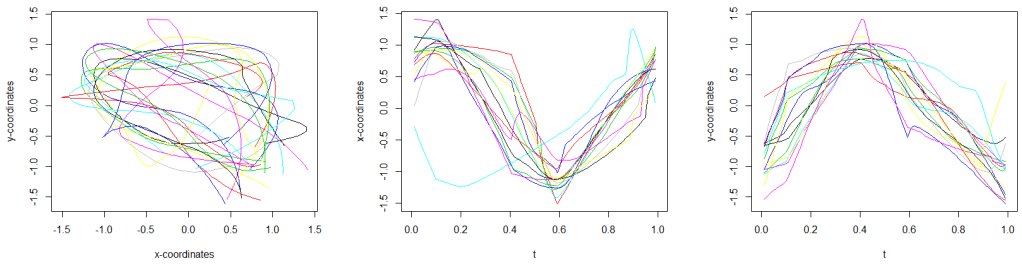
Figure 3.5: An example of registration results in Dataset 3 by four methods for Scenario C with with $\sigma_w = 1$ and $\theta_0 = \pi/8$.



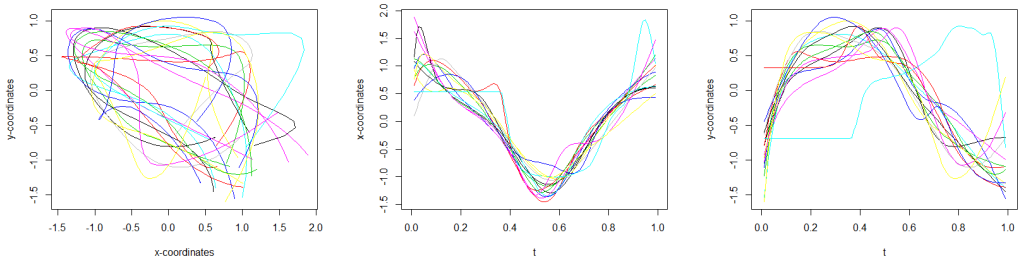

(a) Registration by *GPA*




(b) Registration by *SM*

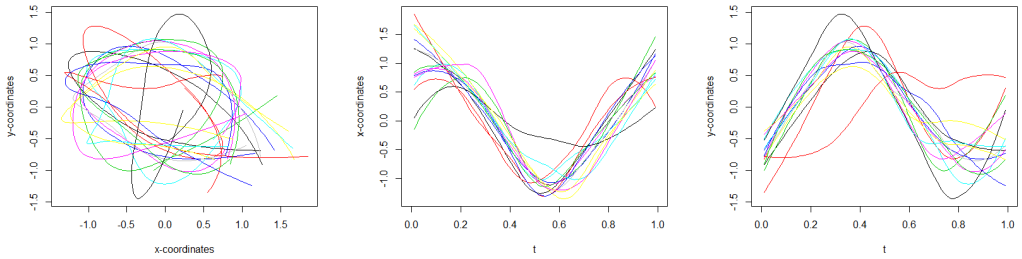



(c) Registration by *SRV*

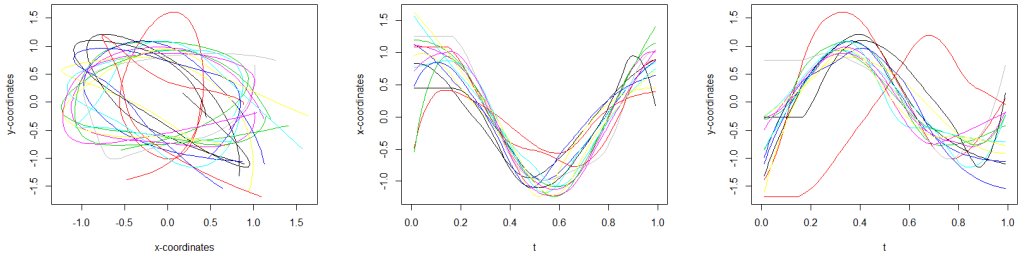



(d) Registration by *GPSM*

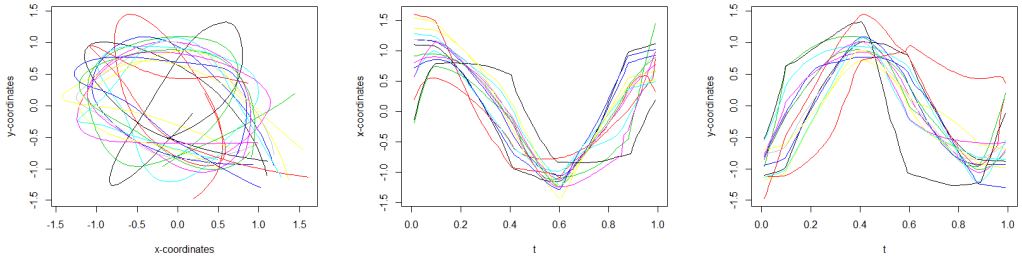
Figure 3.6: Registration of curves from 15 normal people by four methods.



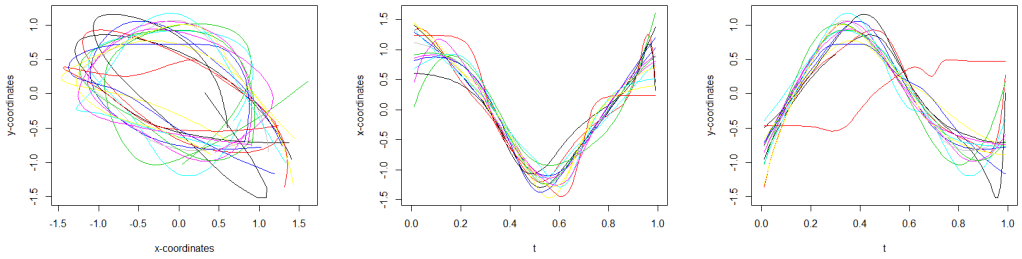

(a) Registration by *GPA*




(b) Registration by *SM*




(c) Registration by *SRV*




(d) Registration by *GPSM*

Figure 3.7: Registration of curves from 15 abnormal people by four methods.

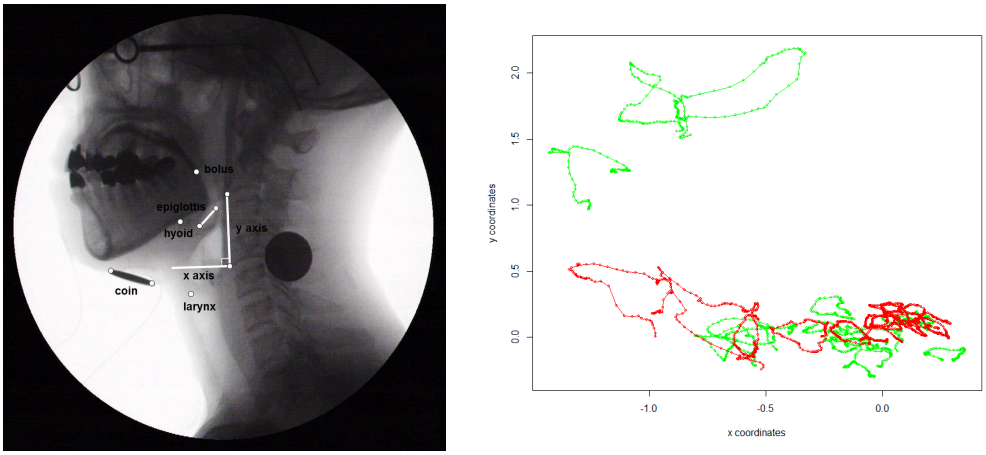
# Chapter 4

# Joint Curve Registration and Classification with Mixed Scalar and Functional Variables

## 4.1 Introduction

Data classification will be conducted after the data acquisition and data registration studied in Chapter 2 and Chapter 3, respectively. While doing the classification for the data (see Figure 4.1(a)), the misaligned problems, for example, the vertical variation and horizontal variation, as seen from Figure 4.1(b), should be addressed. Several works e.g. by Sangalli et al. (2009), Srivastava et al. (2011a) and Cheng et al. (2016) can be applied to carry out the registration for 2D curves. But, their curve alignment is generally performed as a preprocessing technique and the classification on the basis of curves is conducted afterwards. This way is not efficient, since on one hand, a subject belonging to the groups "normal" or "patient" is closely related to how it unfolds its progression pace. This leads to the necessity of simultaneous registration and modeling. On the other hand, those methods rely only on functional variables, i.e., the curves. This does not always work well, particularly for the data having heterogeneity which depends on both functional and scalar variables. In the X-ray video data, the variations depend on time warping as well as scalar variables such as average speed of hyoid bone and the initial level of disease. Therefore, simultaneous curve registration and classification by considering all those factors mentioned is a better way for modeling functional data. The purpose of this chapter is to resolve the problems aforementioned by proposing two-stage functional models for joint curve registration and classification.

The chapter is organized as follows. Section 4.2 proposes the joint curve registration and classification method, provides the related inference, implementation and asymptotic




(a) (b)

Figure 4.1: The motion data of hyoid bone. (**a**) One X-ray image showing the location of hyoid bone which will move forward and backward to form one 2D curve during swallowing, as shown in (**b**). (**b**) 30 trajectories of hyoid bone motion from 15 normal people (curves in green) and 15 patients with stroke (curves in red).

properties of estimators, and ends up with procedures for prediction. We present numerical examples with simulated data and real data to evaluate the proposed model in Section 4.3. Finally, a short summary and discussion are included in Section 4.4.

## 4.2 The joint registration and classification models

Suppose there are $N$ subjects coming from two groups. Let $y$ be the binary variable, where we label the subject by 1 if it is normal, or by 0 if it is abnormal. The number of normal and abnormal subjects are assumed to be $N_1$ and $N_0$, respectively, where $N_1 + N_0 = N$. Let $\boldsymbol{x}_1(t), \boldsymbol{x}_2(t), \ldots, \boldsymbol{x}_N(t)$ be 2D continuous curves, where $\boldsymbol{x}_i(t) = (x_{1i}(t), x_{2i}(t))$ and $x_{1i}(t)$, $x_{2i}(t)$ are the corresponding $x$-coordinates and $y$-coordinates of $\boldsymbol{x}_i(t)$. Let $\boldsymbol{x}_{ki}(t) \stackrel{\Delta}{=} \boldsymbol{x}_i(t)|_{y_i=k}$ be the $i$-th 2D curve in the $k$-th group, where $\boldsymbol{x}_{ki}(t) = (x_{1ki}(t), x_{2ki}(t)) and x_{aki}(t) \stackrel{\Delta}{=} x_{ai}(t)|_{y_i=k}, a = 1, 2; k = 0, 1$. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N$ be the observed scalar variables. For example, in our study, they can represent subjects' gender, age, smoking status and some features or summary statistics originated from those 2D curves, such as the average motion speed and average acceleration amplitude. Suppose there are $m_{ki}$ time points on which the $i$-th curve in $k$-th group are observed. The data set is

$$D = \{(y_{ki}, x_{1ki}(t_{ij}), x_{2ki}(t_{ij}), \boldsymbol{v}_{ki}); i = 1, \ldots, N_k; j = 1, \ldots, m_{ki}; k = 0, 1\},$$

where $y_{ki} \stackrel{\Delta}{=} y_i|_{y_i=k}, \boldsymbol{v}_{ki} \stackrel{\Delta}{=} \boldsymbol{v}_i|_{y_i=k}$. We can also denote $D$ by $\{(y_i, \boldsymbol{x}_i(t_{ij}), \boldsymbol{v}_i); i = 1, \ldots, N\}$, where $t_{ij}$ stand for the observed time points for the $i$-th curve.

### 4.2.1 The models

Most existing methods carry out the classification for the curves with registration problems depending only on the information from themselves. In some cases, this might not be enough. We may need to use information from other variables, either functional or scalar. One example is given in the previous chapter; see the discussion around Table 3.3. Thus, we will use both functional and scalar variables for classification. Given the data $\{(\boldsymbol{x}_i(t), \boldsymbol{v}_i)\}$, we start the first stage models with the assumption

$$y_i|\boldsymbol{v}_i, \boldsymbol{x}_i(t) \sim \text{Bernoulli}(1, \pi_i), \quad \pi_i = P(y_i = 1|\boldsymbol{v}_i, \boldsymbol{x}_i(t)).$$

Then use the following functional logistic regression model

$$\text{logit}(\pi_i|\boldsymbol{v}_i, \boldsymbol{x}_i(t)) = b_0 + \boldsymbol{v}_i^\intercal \boldsymbol{b_1} + \int \boldsymbol{x}_i(g_i^{-1}(t))\boldsymbol{\beta}(t)dt, \quad i = 1, \ldots, N, \tag{4.1}$$

where $b_0$ and $\boldsymbol{b_1}$ are scalar coefficients and $\boldsymbol{\beta}(t) = (\beta_1(t), \beta_2(t))^\intercal$ are coefficient functions. $\boldsymbol{x}_i(g_i^{-1}(t)) = \big(x_{1i}(g_i^{-1}(t)), x_{2i}(g_i^{-1}(t))\big)$, are the 2D curves after registration and $g_i^{-1}(t)$ is the warping function for the $i$-th curve.

In the second stage, we will model these curves. The preprocessing procedure Generalized Procrustes analysis (*GPA*) (Gower, 1975b) will be used at the beginning to address part of registration problems except warping. Little work has been done on model-based registration for multi-dimensional curves. Borrowing the ideas from Raket et al. (2016), we model the the continuous curve $x_{aki}(t)$ by

$$x_{aki}(t) = (\tau_{ak} \circ g_{ki})(t) + r_{aki}(t) + \epsilon_{ai}(t), \quad i = 1, \ldots, N_k, \tag{4.2}$$

where $a = 1$ or 2 represents the $x$- or $y$-coordinates of $x_i(t)$; the item $(\tau_{ak} \circ g_{ki})$ denotes functional composition: $(\tau \circ g)(t) = \tau(g(t))$, where $\tau_{ak}(\cdot)$ is a fixed but unknown nonlinear mean curve. We set $\tau_{ak}(t) = \xi_a(t) + \phi_{ak}(t)$, where $\xi_a$ is the underlying profile shared across two groups and $\phi_{ak}$ is the group-specific variation centered around $\xi_a$. Both can be approximated by a set of basis functions, the details are given in the next section. The variation among different subjects is modeled by a non-linear functional random-effects: $r_{aki}(t)$, by a Gaussian process with zero mean and a parametric covariance function $\boldsymbol{S}$. The error item $\epsilon_{ai}(t)$ is assumed to be Gaussian white noise with variance $\sigma^2$.

According to the discussion in Section 4.1, the variation among different subjects needs to be considered. Thus, we allow warping function depending on $k$ ($k = y_i$) and $i$. We further assume

$$g_{ki}(t) = t + w_k(t) + w_{ki}(t),$$

where $w_k(t)$ is the fixed part, representing consistent timing across all subjects in group

$k$, and $w_{ki}(t)$ is the random part, representing the random variation of timing of subject $i$ in the group $k$. Instead of making assumptions for curves $w_k(t)$ and $w_{ki}(t)$, we first discretize them by a set of fixed parameters, for instance, by $\boldsymbol{w}_k = (w_k(t_1), \ldots, w_k(t_{n_w}))$ and $\boldsymbol{w}_{ki} = (w_{ki}(t_1), \ldots, w_{ki}(t_{n_w}))$ respectively (Zeng et al., 2017). We then model $\boldsymbol{w}_{ki}$ by a Gaussian distribution with zero mean and a parametric covariance function $\boldsymbol{H}$. We can also define the warping function as linear functions, some of which have been examined by others (Liu and Yang, 2009; Sangalli et al., 2010).

We call the models defined in (4.1) and (4.2) as joint curve registration and classification ($JCRC$) models.

### 4.2.2 Estimation

Firstly, we estimate the function $g(t)$ involved in model (4.1) in the second stage. The discrete form of model (4.2) for the $i$-th curve data $\boldsymbol{x}_{aki} = \big(x_{aki}(t_{i1}), \ldots, x_{aki}(t_{im_{ki}})\big)^\intercal$ can be expressed as follows

$$\boldsymbol{x}_{aki} = \boldsymbol{\tau}_{ak}(g_{ki}) + \boldsymbol{r}_{aki} + \boldsymbol{\epsilon}_i, \qquad \text{for } a = 1, 2;\ \ k = 0, 1;\ \ i = 1, \ldots, N_k, \tag{4.3}$$

where $\boldsymbol{\tau}_{ak}(g_{ki}) = \Big(\tau_{ak}\big(g_{ki}(t_{i1})\big), \ldots, \tau_{ak}\big(g_{ki}(t_{im_{ki}})\big)\Big)^\intercal$, and $\boldsymbol{r}_{aki}$ and $\boldsymbol{\epsilon}_i$ are both $m_{ki}$ column vector. We respectively set $\boldsymbol{S}$ as the Matern covariance function with parameters $\boldsymbol{\rho}_s$ and $\boldsymbol{H}$ as the unstructured covariance function or Brownian covariance function with parameter $\boldsymbol{\rho}_h$ (Raket, 2016). This can be estimated by the data; the details are provided in the next subsection. Other covariance functions can also be used (Shi and Choi, 2011). Let $\boldsymbol{S}_{aki}$ and $\boldsymbol{H}_{ki}$ be the covariance matrix of $\boldsymbol{r}_{aki}$ and $\boldsymbol{w}_{ki}$ respectively, which can be calculated by the corresponding covariance function. We model $\tau_{ak}(t)$ using $q$ basis functions $\{\psi_1(t), \ldots, \psi_q(t)\}$ with weights $\boldsymbol{c}_a = (c_{a1}, \ldots, c_{aq})^\intercal$ for $\xi_a(t)$, with weights $\boldsymbol{d}_{ak} = (d_{ak1}, \ldots, d_{akq})^\intercal$ for $\phi_{ak}(t)$ . Thus, $\boldsymbol{\tau}_{ak}(g_{ki}) = \boldsymbol{\Psi}_{ki}(\boldsymbol{c}_a + \boldsymbol{d}_{ak})$, where $\boldsymbol{\Psi}_{ki} = [\boldsymbol{\Psi}_{ki1}, \ldots, \boldsymbol{\Psi}_{kiq}]_{m_{ki} \times q}, \boldsymbol{\Psi}_{kil} = (\boldsymbol{\psi}_l(g_{ki}(t_{i1})), \ldots, \boldsymbol{\psi}_l(g_{ki}(t_{im_{ki}})))^\intercal, l = 1, \ldots, q$. Here, we use a smooth non-linear deformation of the curves for $g_{ki}(t)$, which is produced by an increasing spline (Raket, 2016).

All the unknown parameters in the model (4.3) to be estimated are

$$\boldsymbol{\theta}_{\boldsymbol{x}} \stackrel{\Delta}{=} \{\boldsymbol{c}_a, \boldsymbol{d}_{ak}, \boldsymbol{w}_k, \boldsymbol{w}_{ki}, \boldsymbol{\rho}_h, \boldsymbol{\rho}_s, \sigma, a = 1, 2; k = 0, 1; i = 1, \ldots, N_k\}.$$

Borrowing the ideas from Raket et al. (2016), those parameters can be estimated iteratively through three conditional models, leading to the estimator of $g(t)$ determined by $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$ only. The details are given in Section 4.2.3.

Secondly, referring back to model (4.1) in the first stage, we have

$$\text{logit}(\pi_i|\boldsymbol{v}_i, \boldsymbol{x}_i(t)) = b_0 + \boldsymbol{v}_i^\intercal \boldsymbol{b_1} + \int \hat{\boldsymbol{x}}_i(t)\boldsymbol{\beta}(t)dt, \quad i = 1, \ldots, N, \tag{4.4}$$

where $\hat{\boldsymbol{x}}_i(t) = \boldsymbol{x}_i(\hat{g}_i^{-1}(t))$. Using the fast fitting methods for generalized functional linear models proposed by Goldsmith et al. (2011), the $x_{ai}(\hat{g}_i^{-1}(t))$ are estimated as

$$\begin{aligned}\hat{x}_{ai}(\hat{g}_i^{-1}(t)) &= \sum_{j=1}^{K_x} p_{aij}\phi_{aj}(t) \\ &= \boldsymbol{p_{ai}}^\intercal \boldsymbol{\phi}_a(t), \quad a = 1, 2.\end{aligned}$$

Here, $p_{aij} = \int \hat{x}_{ai}(\hat{g}_i^{-1}(t))\phi_{aj}(t)dt$, $\boldsymbol{\phi}_a(t) = \big(\phi_{a1}(t), \ldots, \phi_{aK_x}(t)\big)^\intercal$ is the collection of the first $K_x$ eigenfunctions of the smoothed covariance matrix $\sum_{\hat{\boldsymbol{x}}_a}(t_1, t_2) = \text{cov}[\hat{x}_{ai}(t_1), \hat{x}_{ai}(t_2)]$ (Ramsay and Silverman, 2005). Using the truncated power series spline basis, the coefficient function $\boldsymbol{\beta}_a(t)$ can be approximated as

$$\begin{aligned}\boldsymbol{\beta}_a(t) &= e_{a1} + e_{a2}t + \sum_{j=3}^{K_e} e_{aj}(t - \kappa_j)_+ \\ &= \sum_{j=1}^{K_e} e_{aj}\xi_{aj}(t) \\ &= \boldsymbol{\xi}_a^\intercal(t)\boldsymbol{e}_a, \quad a = 1, 2,\end{aligned}$$

where $K_e$ is the number of truncated power series spline basis, $\{\kappa_j\}_{j=3}^{K_e}$ are knots and $t_+ = \begin{cases} t, & t > 0 \\ 0, & t \leq 0 \end{cases}$. We further assume that $\{e_{aj}\}_{j=3}^{K_e} \sim N_{K_e-2}(0, \sigma_e^2\boldsymbol{I})$ for inducing smoothing. Other choices of basis functions can also be used with corresponding changes to penalty. Thus,

$$\begin{aligned}\int \hat{x}_{ai}(\hat{g}_i^{-1}(t))\boldsymbol{\beta}_a(t)dt &= \int \boldsymbol{p_{ai}}^\intercal \boldsymbol{\phi}_a(t)\boldsymbol{\xi}_a^\intercal(t)\boldsymbol{e}_a dt \\ &= \boldsymbol{p_{ai}}^\intercal \boldsymbol{J}_{a\phi\xi}\boldsymbol{e}_a, \quad a = 1, 2,\end{aligned}$$

where $\boldsymbol{J}_{a\phi\xi}$ is a $K_x \times K_e$ dimensional matrix $[J_{a\phi\xi}]_{K_x \times K_e}$ with the $(i, j)$th entry equal to $\int \phi_{ai}(t)\xi_{aj}(t)dt$. Denote $\boldsymbol{u}_i$ by the fixed effects:

$$\big(1, \boldsymbol{v}_i^\intercal, \boldsymbol{p_{1i}}^\intercal[J_{1\phi\xi}]_{\cdot 1}, \boldsymbol{p_{1i}}^\intercal[J_{1\phi\xi}]_{\cdot 2}, \boldsymbol{p_{2i}}^\intercal[J_{2\phi\xi}]_{\cdot 1}, \boldsymbol{p_{2i}}^\intercal[J_{2\phi\xi}]_{\cdot 2}\big)^\intercal,$$

where $[J_{a\phi\xi}]_{\cdot j}$ is the $j$-th column vector of the matrix $[J_{a\phi\xi}]_{K_x \times K_e}$, $\boldsymbol{\eta}$ by a vector of

parameters:

$$(b_0, \boldsymbol{b}_1, e_{11}, e_{12}, e_{21}, e_{22})^\intercal,$$

$\boldsymbol{z}_i$ by a design vector:

$$\left(\boldsymbol{p_{ai}}^\intercal [J_{a\phi\xi}]_{\cdot j}\right)^\intercal_{1\le a\le 2; 3\le j\le K_e},$$

and $\boldsymbol{\alpha}$ by the random effects $\{e_{1j}, e_{2j}\}_{j=3}^{K_e}$. Model (4.4) can then be reformulated as

$$\begin{aligned} \text{logit}(\pi_i|\boldsymbol{v}_i, \boldsymbol{x}_i(t)) &= b_0 + \boldsymbol{v}_i^\intercal \boldsymbol{b_1} + \sum_{a=1}^{2}\sum_{j=1}^{2} \boldsymbol{p_{ai}}^\intercal [J_{a\phi\xi}]_{\cdot j} e_{aj} + \sum_{a=1}^{2}\sum_{j=3}^{K_e} \boldsymbol{p_{ai}}^\intercal [J_{a\phi\xi}]_{\cdot j} e_{aj}, \\ &= \boldsymbol{u}_i^\intercal \boldsymbol{\eta} + \boldsymbol{z}_i^\intercal \boldsymbol{\alpha}, \quad i = 1, \ldots, N, \\ \{e_{aj}\}_{j=3}^{K_e} &\sim N_{K_e-2}(0, \sigma_e^2 \boldsymbol{I}), \quad a = 1, 2. \end{aligned} \tag{4.5}$$

In the classical definition of generalized linear mixed models, the responses $\{y_i\}_{1\le i\le N}$ are conditionally independent given the vector $\boldsymbol{\alpha}$. Let $\boldsymbol{\theta} \triangleq \{b_0, \boldsymbol{b}_1, e_{11}, e_{12}, e_{21}, e_{22}, \sigma_e\}$ denote all the parameters involved in model (4.4) to be estimated in this stage. Once we have chosen the basis functions for $\boldsymbol{\beta}(t)$, model (4.5) only depends on the choice of $K_x$ and $K_e$. According to Ruppert (2002) we select $K_e$ large enough to avoid under-smoothing and select $K_x \ge K_e$ to meet the identifiability constraint. A pair of desirable $K_x$ and $K_e$ are selected by the Cross-Validation method. This model can be fit robustly using standard mixed effects software (Ruppert, 2002; McCulloch et al., 2008).

### 4.2.3 Implementation

Model (4.3) has lots of parameters and also has effects that interact, which makes simultaneous likelihood estimation intractable. We borrow the scheme proposed by Raket et al. (2016) in which fixed effects $\tau_{ak}$, warping parameters $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$ and variance parameters $\sigma^2$, $\boldsymbol{\rho}_s$ and $\boldsymbol{\rho}_h$ are estimated iteratively on three different levels of modelling.

- Nonlinear model – estimating warping parameters $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$. At this level, we fix all the other parameters and simultaneously perform conditional likelihood estimation of group-specific warping effects $\boldsymbol{w}_k$ and predict the random subject-specific warping effects $\boldsymbol{w}_{ki}$.

- Fixed warp model – estimating the fixed effects $\tau_{ak}$. At this level, we fix the group-specific warping effects $\boldsymbol{w}_k$ at the conditional maximum likelihood estimate, and the random subject-specific warping effects $\boldsymbol{w}_{ki}$ at the predicted values. The resulting model turns out to be an approximate linear mixed-effects model with Gaussian random effects $r_{ki}$ and $\epsilon_i$. This allows direct maximum-likelihood estimation of the remaining fixed effects $\tau_{ak}$.

- Linearized model – estimating the variance parameters $\sigma^2$, $\boldsymbol{\rho}_s$ and $\boldsymbol{\rho}_h$. At this level, the first-order Taylor approximation of model (4.3) in the random warp $\boldsymbol{w}_{ki}$ is considered. We carry out this linearization around the estimate of $\boldsymbol{w}_k$ plus the given prediction of $\boldsymbol{w}_{ki}$ from the nonlinear model. The resulting model is also a linear mixed-effects model and we can explicitly compute the likelihood. All the variance parameters $\sigma^2$, $\boldsymbol{\rho}_s$ and $\boldsymbol{\rho}_h$ are estimated using maximum-likelihood estimation.

These three different levels of modelling leads to the estimator of warping function $g^{-1}(t)$. Going back to GLMM model (4.5), we can then estimate those parameters in $\boldsymbol{\theta}$.

First of all, let $\boldsymbol{x}_{ak} = (\boldsymbol{x}_{ak1}^\intercal, \ldots, \boldsymbol{x}_{akN_k}^\intercal)^\intercal \in \mathbb{R}^{m_k}$, where $m_k = \sum_{i=1}^{N_k} m_{ki}$, and $\boldsymbol{x}_a = (\boldsymbol{x}_{a0}^\intercal, \boldsymbol{x}_{a1}^\intercal)^\intercal \in \mathbb{R}^m$, where $m = \sum_{k=0}^{1} m_k$. Let $\sigma^2\boldsymbol{S}_{ak}$, $\sigma^2\boldsymbol{S}_a$ be the covariance matrices of $\boldsymbol{r}_{ak} = (\boldsymbol{r}_{aki})_i$ and $\boldsymbol{r}_a = (\boldsymbol{r}_{ak})_k$ respectively. In order to simplify the likelihood computations, all the random effects are scaled by a noise standard deviation $\sigma$. The norm induced by a full-rank covariance matrix $\boldsymbol{B}$ is denoted by $||\boldsymbol{A}||^2_{\boldsymbol{B}} = \boldsymbol{A}^T\boldsymbol{B}^{-1}\boldsymbol{A}$.

**(i) Estimate the fixed effects $\tau_{ak}$**

Given $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$, we have $\boldsymbol{x}_{aki} \sim N_{m_{ki}}(\boldsymbol{\Psi}_{ki}(\boldsymbol{c}_a + \boldsymbol{d}_{ak}), \boldsymbol{I}_{m_{ki}} + \boldsymbol{S}_{aki}), a = 1, 2$ and $i = 1, \ldots, N_k$, where $\boldsymbol{I}_n$ denotes the $n \times n$ identity matrix. The negative log likelihood for the weights $\boldsymbol{c}_a$ is proportional to

$$l(\boldsymbol{c}_a) = \sum_{k=0}^{1}\sum_{i=1}^{N_k} ||\boldsymbol{x}_{aki} - \boldsymbol{\Psi}_{ki}\boldsymbol{c}_a||^2_{\boldsymbol{I}_{m_{ki}}+\boldsymbol{S}_{aki}}.$$

The estimator of $\boldsymbol{c}_a$ is given by

$$\hat{\boldsymbol{c}}_a = (\boldsymbol{\Psi}^\intercal(\boldsymbol{I}_m + \boldsymbol{S}_a)^{-1}\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^\intercal(\boldsymbol{I}_m + \boldsymbol{S}_a)^{-1}\boldsymbol{x}_a,$$

where $\boldsymbol{\Psi} = [\boldsymbol{\Psi}_{01}^\intercal, \ldots, \boldsymbol{\Psi}_{0N_0}^\intercal, \boldsymbol{\Psi}_{11}^\intercal, \ldots, \boldsymbol{\Psi}_{1N_1}^\intercal]^\intercal \in \boldsymbol{R}^{m\times q}$. The negative log likelihood for the weights $\boldsymbol{d}_{ak}$ (its square magnitude is penalized by a weighting factor $\eta$) is proportional to

$$l(\boldsymbol{d}_{ak}) = \sum_{i=1}^{N_k} ||\boldsymbol{x}_{aki} - \boldsymbol{\Psi}_i(\hat{\boldsymbol{c}}_a + \boldsymbol{d}_{ak})||^2_{\boldsymbol{I}_{m_{ki}}+\boldsymbol{S}_{aki}} + \eta\boldsymbol{d}_{ak}^\intercal\boldsymbol{d}_{ak},$$

This gives the estimator

$$\hat{\boldsymbol{d}}_{ak} = (\boldsymbol{\Psi}_k^\intercal(\boldsymbol{I}_{m_k} + \boldsymbol{S}_{ak})^{-1}\boldsymbol{\Psi}_k + \eta\boldsymbol{I}_W)^{-1}\boldsymbol{\Psi}_k^\intercal(\boldsymbol{I}_{m_k} + \boldsymbol{S}_{ak})^{-1}(\boldsymbol{x}_{ak} - \boldsymbol{\Psi}_k\hat{\boldsymbol{c}}_a),$$

where $\boldsymbol{\Psi}_k = [\boldsymbol{\Psi}_{k1}^\intercal, \ldots, \boldsymbol{\Psi}_{kN_k}^\intercal]^\intercal \in \boldsymbol{R}^{m_k\times q}$.

**(ii) Estimate warping parameters $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$**

Given $\hat{\boldsymbol{c}}_a$ and $\hat{\boldsymbol{d}}_{ak}$, we have the joint probability density function of $(\boldsymbol{x}_{aki}, \boldsymbol{w}_{ki})$ given by

$$f(\boldsymbol{x}_{aki}, \boldsymbol{w}_{ki}) = f(\boldsymbol{x}_{aki}|\boldsymbol{w}_{ki}) * f(\boldsymbol{w}_{ki}) \sim N_{m_{ki}}(\boldsymbol{\Psi}_{ki}(\hat{\boldsymbol{c}}_a + \hat{\boldsymbol{d}}_{ak}), \boldsymbol{I}_{m_{ki}} + \boldsymbol{S}_{aki}) * N_{n_w}(0, \boldsymbol{H}_{ki}).$$

We can simultaneously estimate the fixed warping effects $\boldsymbol{w}_k$ and predict the random warping effects $\boldsymbol{w}_{ki}$ from the joint conditional negative log posterior. It is proportional to

$$l(\boldsymbol{w}_k, \boldsymbol{w}_{ki}) = \sum_{a=1}^{2}\sum_{i=1}^{N_k} ||\boldsymbol{x}_{aki} - \boldsymbol{\Psi}_{ki}(\hat{\boldsymbol{c}}_a + \hat{\boldsymbol{d}}_{ak})||^2_{\boldsymbol{I}_{m_{ki}}+\boldsymbol{S}_{aki}} + 2\sum_{i=1}^{N_k} ||\boldsymbol{w}_{ki}||^2_{\boldsymbol{H}_{ki}}, \tag{4.6}$$

where $\boldsymbol{\Psi}_{ki}$ is determined by $m_{ki}$ discrete values of the inverse of warping function $g_{ki}(t)$ which is totally characterized by $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$ as aforementioned. By minimizing $l(\boldsymbol{w}_k, \boldsymbol{w}_{ki})$ we can obtain the estimation of $\boldsymbol{w}_k$ and the prediction of $\boldsymbol{w}_{ki}$.

**(iii) Estimate the variance parameters $\sigma^2$, $\boldsymbol{\rho}_s$ and $\boldsymbol{\rho}_h$**

By using the first-order Taylor approximation of model (4.3) in the the random warping parameters $\boldsymbol{w}_{ki}$ around a given prediction $\boldsymbol{w}^0_{ki}$ ($\boldsymbol{w}^0_{ki}$ is specified by the estimate of $\boldsymbol{w}_{ki}$ from **(ii)** in the current iteration), we can write this model as a vectorized linear mixed-effects model

$$\boldsymbol{x}_a \approx \boldsymbol{G}_a + \boldsymbol{B}_a(\boldsymbol{W} - \boldsymbol{W}^0) + \boldsymbol{r}_a + \boldsymbol{\epsilon}, \qquad a = 1, 2, \tag{4.7}$$

where $\boldsymbol{x}_a = \{\boldsymbol{x}_{ai}, i = 1, \ldots, N\}$, with effects given by

$$\begin{aligned}
\boldsymbol{G}_a &= \left\{\boldsymbol{\Psi}_{ki}|_{g_{ki}=g^0_{ki}}(\boldsymbol{c}_a + \boldsymbol{d}_{ak})\right\}_{kij} \in \mathbb{R}^m,\\
\boldsymbol{B}_a &= \mathrm{diag}(\boldsymbol{B}_{aki})_{ki} \in \mathbb{R}^{m\times Nn_w},\\
\boldsymbol{B}_{aki} &= \left\{\partial_{\mathbf{g}_{ki}}\Big(\tau_{ak}\big(g_{ki}(t_j)\big)\Big)\Big|_{g_{ki}=g^0_{ki}}\Big(\nabla \boldsymbol{w}_{ki}\big(g_{ki}(t_j)\big)\Big)^{\intercal}\Big|_{\boldsymbol{w_{ki}}=\boldsymbol{w_{ki}}^0}\right\}_j \in \mathbb{R}^{m_{ki}\times n_w},\\
\boldsymbol{W} &= (\boldsymbol{w}_{ki})_{ki} \sim N_{Nn_w}(0, \sigma^2\boldsymbol{I}_N \otimes \boldsymbol{H}_{n_w\times n_w}), \quad \boldsymbol{W}^0 = (\boldsymbol{w}^0_{ki})_{ki} \in \mathbb{R}^{Nn_w},\\
\boldsymbol{r}_a &\sim N_m(0, \sigma^2\boldsymbol{S}_a), \quad \boldsymbol{S}_a = \mathrm{diag}(\boldsymbol{S}_{aki})_{ki} \in \mathbb{R}^{m\times m},\\
\boldsymbol{\epsilon} &\sim N_m(0, \sigma^2\boldsymbol{I}_m).
\end{aligned}$$

Here, $g^0_{ki}(t) = t + w_k(t) + w^0_{ki}(t)$, $\mathrm{diag}(\boldsymbol{B}_{aki})_{ki}$ is the block diagonal matrix with the $\boldsymbol{B}_{aki}$ matrices along its diagonal, and $\mathrm{diag}(\boldsymbol{S}_{aki})_i$ is the block diagonal matrix with the $\boldsymbol{S}_{aki}$ matrices along its diagonal. The derivation of the linearized model (4.7) is similar to the proof of C.2 in Appendix C. The negative profile log likelihood function for the model (4.7) is proportional to

$$l(\sigma^2, \boldsymbol{\rho}_s, \boldsymbol{\rho}_h) = \sum_{a=1}^{2} \sigma^2||\boldsymbol{x}_a - \boldsymbol{G}_a + \boldsymbol{B}_a\boldsymbol{W}^0||^2_{\boldsymbol{V}_a} + \sum_{a=1}^{2} \log\det\boldsymbol{V}_a + 2m\log\sigma^2,$$

where $\boldsymbol{V}_a = \boldsymbol{S}_a + \boldsymbol{B}_a(\boldsymbol{I}_n \otimes \boldsymbol{H}_{n_w \times n_w})\boldsymbol{B}_a^\intercal + \boldsymbol{I}_m$.

To speed up convergence, we usually repeat the above three steps for several times within each iteration. Given the estimators $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$, we can obtain the estimator of $g(t)$.

**(iv) Fit the generalized functional linear model**

Going back to GLMM model (4.5), we can then estimate the parameters $\boldsymbol{\theta}$ using maximum likelihood and restricted maximum likelihood techniques. Furthermore, variance estimators and confidence intervals, like Figure 4.5 in simulation study 1, can be obtained following standard methods and software (Ruppert et al., 2003; Wood, 2006). The details are given in Goldsmith et al. (2011). The *nlme* package (Jose et al., 2017) is used for fitting the generalized linear mixed effects model in our simulation studies.

### 4.2.4 Asymptotic properties of estimation of $\boldsymbol{\theta}$

Following the implementation, we will explore the asymptotic properties of the estimation of the parameters $\boldsymbol{\theta}$ involved in the generalized linear mixed model (4.5) using the methods of Jiang and Zhang (2001). They show that the first-step estimator $\tilde{\boldsymbol{\theta}}$ of the vector $\boldsymbol{\theta}$ of parameters acquired by solving a system of estimating equation is consistent. Additionally, a second-step estimator $\hat{\boldsymbol{\theta}}$, obtained by solving a system of optimal estimating equations whose coefficients are estimated by $\tilde{\boldsymbol{\theta}}$, maintains the asymptotic optimality. Their methods can be directly applied to the *JCRC* models in terms of asymptotic properties, which to some extent also justify our methodology.

For simplicity, we assume $\boldsymbol{b}_1$ in model (4.5) is univariate. Then the fixed effect $\boldsymbol{\mu}_i = \{\mu_{ij}\}_{j=1}^{6}$ and the design vector $\boldsymbol{z}_i = \{z_{ij}\}_{j=1}^{2(K_e-2)}$. Let the number of time points be the same for all subjects. The base statistics defined by Jiang and Zhang (2001) corresponding to the model (4.5) are

$$
\begin{aligned}
L_j &= \sum_{i=1}^{N} u_{ij} y_i, \quad 1 \le j \le 6, \\
L_{6+j} &= \big(\sum_{i=1}^{N} z_{ij} y_i\big)^2 - \sum_{i=1}^{N} (z_{ij} y_i)^2, \quad 1 \le j \le 2(K_e - 2),
\end{aligned}
$$

The base statistics $\boldsymbol{L}$ is a vector with dimension $(2K_e + 2)$. Let $\boldsymbol{D}$ be an arbitrary matrix with dimension $7 \times (2K_e + 2)$. Then the first-step estimator is obtained by solving the equation

$$
\boldsymbol{D}\boldsymbol{L} = \boldsymbol{D}\boldsymbol{\mu}(\boldsymbol{\theta}) \tag{4.8}
$$

where $\mu(\boldsymbol{\theta}) = E(\boldsymbol{L})$.

Let $\boldsymbol{\Theta}$ be the parameter space. Since $\boldsymbol{D}$, $\boldsymbol{L}$ and $\mu(\boldsymbol{\theta})$ may depend on $N$, we shall

use the notation $\boldsymbol{D}_N$, $\boldsymbol{L}_N$ and $\mu_N(\boldsymbol{\theta})$. The solution to (4.8) does not change if $\boldsymbol{D}_N$ is replaced by $\boldsymbol{C}_N^{-1}\boldsymbol{D}_N$, where $\boldsymbol{C}_N = \text{diag}(c_{N,1}, \ldots, c_{N,7})$ and $c_{N,j}$ $(1 \leq j \leq 7)$ is a sequence of positive constants. We write

$$\boldsymbol{A}_N = \boldsymbol{C}_N^{-1}\boldsymbol{D}_N\boldsymbol{L}_N,$$
$$\boldsymbol{A}_N(\boldsymbol{\theta}) = \boldsymbol{C}_N^{-1}\boldsymbol{D}_N\boldsymbol{\mu}_N(\boldsymbol{\theta}).$$

Then, the first-step estimator $\tilde{\boldsymbol{\theta}} = \tilde{\boldsymbol{\theta}}_N$ is the solution to the equation

$$\boldsymbol{A}_N(\boldsymbol{\theta}) = \boldsymbol{A}_N. \tag{4.9}$$

Assume $\boldsymbol{\theta}_0$ is the vector of true parameters and define $d(x, A) = \inf_{y \in A} |x - y|$. Let $V_N$ be the covariance matrix of $\boldsymbol{L}_N$. Write $U_N = \partial\boldsymbol{\mu}_N/\partial\boldsymbol{\theta}^\intercal|_{\boldsymbol{\theta}_0}$. Let $H_{N,j}(\boldsymbol{\theta}) = \partial^2\boldsymbol{\mu}_{N,j}/\partial\boldsymbol{\theta}^2$, where $\boldsymbol{\mu}_{N,j}$ is the $j$th component of $\boldsymbol{\mu}_N(\boldsymbol{\theta})$, and $H_{N,j,\epsilon} = \sup_{|\boldsymbol{\theta}-\boldsymbol{\theta}_0| \leq \epsilon} ||H_{N,j}(\boldsymbol{\theta})||$, for $1 \leq j \leq 2K_e+2$. Let $d_{N,ij}$ be the $(i,j)$ element of $\boldsymbol{D}_N$ and use $\lambda_{\min}$ as the smallest eigenvalue. We have the following theorems on the existence, consistency and asymptotic normality of the first- and second-step estimators.

**Theorem 4.1.** (Existence and Consistency) *Suppose that, as $N \to \infty$,*

$$\boldsymbol{A}_N - \boldsymbol{A}_N(\boldsymbol{\theta}_0) \to 0$$

*in probability, and*

$$\liminf d\{\boldsymbol{A}_N(\boldsymbol{\theta}_0), \boldsymbol{A}_N^c(\boldsymbol{\Theta})\} > 0.$$

*Then, with probability tending to one, the solution to (4.9) exists and is in $\boldsymbol{\Theta}$. If, in addition, there is a sequence $\boldsymbol{\Theta}_N \subset \boldsymbol{\Theta}$ such that*

$$\liminf \inf_{\boldsymbol{\theta} \notin \boldsymbol{\Theta}_N} |\boldsymbol{A}_N(\boldsymbol{\theta}) - \boldsymbol{A}_N(\boldsymbol{\theta}_0)| > 0,$$
$$\liminf \inf_{\boldsymbol{\theta} \in \boldsymbol{\Theta}_N, \boldsymbol{\theta} \neq \boldsymbol{\theta}_0} \frac{|\boldsymbol{A}_N(\boldsymbol{\theta}) - \boldsymbol{A}_N(\boldsymbol{\theta}_0)|}{|\boldsymbol{\theta} - \boldsymbol{\theta}_0|} > 0.$$

*Then, any solution $\tilde{\boldsymbol{\theta}}_N$ to (4.9) is consistent.*

**Theorem 4.2.** (Asymptotic Normality) *Suppose that*

*(i) the components of $\mu_N(\boldsymbol{\theta})$ are twice continuously differentiable;*

*(ii) $\tilde{\boldsymbol{\theta}}_N$ satisfies (4.9) with probability tending to one and is consistent;*

*(iii) there exists* $\epsilon > 0$ *such that*

$$\frac{|\tilde{\boldsymbol{\theta}}_N - \boldsymbol{\theta}_0|}{(\lambda_{N,1}\lambda_{N,2})^{\frac{1}{2}}} \max_{1\le i\le 7} c_{N,i}^{-1}\Big(\sum_{j=1}^{2K_e+2} |d_{N,ij}| H_{N,j,\epsilon}\Big) \to 0$$

*in probability, where*

$$\lambda_{N,1} = \lambda_{\min}\big(\boldsymbol{C}_N^{-1}\boldsymbol{D}_N V_N \boldsymbol{D}_N^{\intercal}\boldsymbol{C}_N^{-1}\big),$$
$$\lambda_{N,2} = \lambda_{\min}\big(U_N^{\intercal} D_N^{\intercal}(\boldsymbol{D}_N V_N \boldsymbol{D}_N^{\intercal})^{-1}\boldsymbol{D}_N U_N\big);$$

*(iv)* $[\boldsymbol{C}_N^{-1}\boldsymbol{D}_N V_N \boldsymbol{D}_N^{\intercal}\boldsymbol{C}_N^{-1}]^{-\frac{1}{2}}[\boldsymbol{A}_N - \boldsymbol{A}_N(\boldsymbol{\theta}_0)] \to N(0, \boldsymbol{I}_7)$ *in distribution.*

*Then,* $\tilde{\boldsymbol{\theta}}$ *is asymptotically normal with mean* $\boldsymbol{\theta}_0$ *and asymptotic covariance matrix*

$$(\boldsymbol{D}_N U_N)^{-1}\boldsymbol{D}_N V_N \boldsymbol{D}_N^{\intercal}(\boldsymbol{D}_N U_N)^{-\intercal}.$$

By replacing the conditions of Theorems 4.1 and 4.2 by corresponding conditions with a 'probability statement', we can get the sufficient conditions for existence, consistency and asymptotic normality of the second-step estimators. The details of the proofs are available in Jiang and Zhang (2001).

### 4.2.5 Prediction

It is of interest to predict $y^*$ at a new test data point $(\boldsymbol{x}^*(t), \boldsymbol{v}^*)$. We develop an iteration method to predict $y^*$ through both model (4.1) and model (4.2).

1. Initialise $y^*$ by fitting the model (4.1) without using functional variables, i.e.

$$\text{logit}(\pi_i|\boldsymbol{v}_i) = b_0 + \boldsymbol{v}_i^{\intercal}\boldsymbol{b_1}, \quad i = 1, \ldots, N.$$

We initially predict $\pi^* = \frac{1}{1+\exp\{\hat{b}_0 + \boldsymbol{v}^{*\intercal}\hat{\boldsymbol{b}}_1\}}$, where $\{\hat{b}_0, \hat{\boldsymbol{b}}_1\}$ are the estimators of $\{b_0, \boldsymbol{b}_1\}$. If $\pi^* \ge 0.5$, we set $y^{*(0)} = 1$; otherwise, $y^{*(0)} = 0$.

2. Calculate $\boldsymbol{x}^*(g^{-1}(t))$ given $y^{*(i_0)}$, where $i_0$ indicates the $i_0$-th iteration. Given the observed 2D curve $\boldsymbol{x}^*(t) = (x_1^*(t), x_2^*(t))^{\intercal}$ and the estimators $\hat{\boldsymbol{\theta}}_1$, the estimate of the subject-specific warping part $\boldsymbol{w}_{k*}$ can be obtained by minimizing the joint conditional negative log likelihood

$$l(\hat{\boldsymbol{w}}_k, \boldsymbol{w}_{k*}) = \sum_{a=1}^{2} ||\boldsymbol{x}_a^* - \boldsymbol{\Psi}_{k*}(\hat{\boldsymbol{c}}_a + \hat{\boldsymbol{d}}_{ak})||^2_{\boldsymbol{I}_{m_{k*}} + \hat{\boldsymbol{S}}_{ak*}} + 2||\boldsymbol{w}_{k*}||^2_{\hat{\boldsymbol{H}}_{ak*}}, \qquad k = y^{*(i_0)},$$

where $g_{k*}(t) = t + \hat{w}_k(t) + w_{k*}(t)$ and $\boldsymbol{\Psi}_{k*}$ is determined by $m_{ki}$ discrete values of the inverse of warping function $g_{k*}(t)$. Then $\boldsymbol{x}^*(g^{-1}(t))$ can be predicted as $\boldsymbol{x}^*(\hat{g}_{k*}^{-1}(t))$ where $\hat{g}_{k*}(t) = t + \hat{w}_k(t) + \hat{w}_{k*}(t)$.

3. Update $y^*$ as $y^{*(i_0+1)}$ from the logistic functional model (4.1) given the data $(\boldsymbol{x}^*(\hat{g}_{k*}^{-1}(t)), \boldsymbol{v}^*)$, where $k = y^{*(i_0)}$.

4. Repeat step 2 and 3 until the value of $y^*$ remains unchanged.

Regarding to how the initial value of $y^*$ influences the final results, two typical scenarios might be considered. The first scenario is that if $\pi^*$ is close to 1 or 0, say, 0.9 or 0.1, the initial value of $y^*$ always equals the final result 1 or 0. The reason is that the data is able to provide sufficient information and the above procedure will result in the convergence to the correct result, like Scenario A in Table 4.2 and Table 4.3 in the real data analysis. However, if $\pi^*$ is close to 0.5, say 0.45 or 0.55, it is very likely that the initial value of $y^*$ is different from final result. This is since the data does not give enough information, see the discussion of Scenario B in Table 4.2.

## 4.3 Numerical analyses

### 4.3.1 Simulation study 1

In this simulation study, we explore the performance of the proposed method *JCRC* in terms of estimating $b_0, \boldsymbol{b}_1, \boldsymbol{\beta}(t)$ in model (4.1). Scalar binary outcomes $y_i$ are generated based on the following models

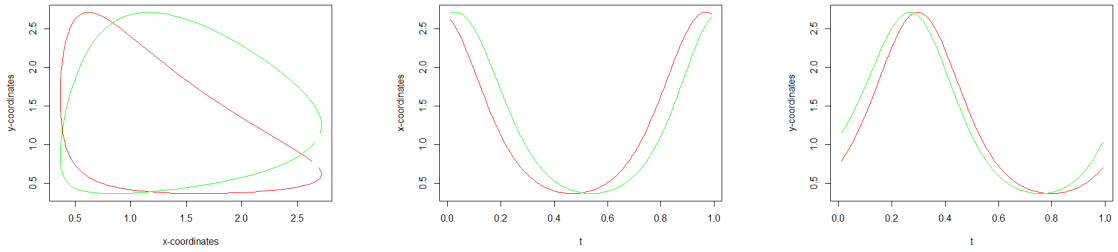
$$\begin{aligned} \omega_i &= b_0 + v_{i1} b_1 + \frac{1}{m} \sum_{j=1}^{m} \big( x_{1i}(g^{-1}(t_j)) \beta_1(t_j) + x_{2i}(g^{-1}(t_j)) \beta_2(t_j) \big), \\ \pi_i &= \frac{1}{1 + \exp(-\omega_i)}, \\ y_i &\sim \text{Bernoulli}(1, \pi_i), \qquad i = 1, \ldots, N, \end{aligned} \tag{4.10}$$

where $v_{i1}$ is scalar variable and $x_1(g^{-1}(t))$, $x_2(g^{-1}(t))$ are functional variables.

**Data generation**

1. We will generate the outcomes $y$'s through four steps.

   (a) Generate the underlying true curves, i.e. the curves based on the internal time scale $g^{-1}(t)$. We assume the curves in two groups share the following two different true means, similar to two patterns of the real curves of hyoid bone's




(a) 2D curves (b) $\boldsymbol{\mu}_1(t)$ (c) $\boldsymbol{\mu}_2(t)$

Figure 4.2: True mean curves. Curves in green indicate the first group ($y = 0$) while those in red represent the second group ($y = 1$).

movement

$$\begin{aligned}\boldsymbol{\mu}_1(t) &= \big(\mu_{11}(t), \mu_{21}(t)\big) = \big(\exp(\cos(2\pi t + 0.2)), \exp(\sin(2\pi t - 0.3))\big),\\ \boldsymbol{\mu}_2(t) &= \big(\mu_{12}(t), \mu_{22}(t)\big) = \big(\exp(\cos(2\pi t^{1.05} - 0.15)), \exp(\sin(2\pi t^{1.1} + 0.1))\big).\end{aligned} \tag{4.11}$$

Figure 4.2 shows the underlying true curves. We use the equidistant points $t_j = \frac{j+1}{102}, j = 1, \ldots, 100$ as the input grid, i.e. $m_{ki} = 100$. The underlying true mean curve for the $i$-th curve is generated by

$$\boldsymbol{\tau}_{aki} = \boldsymbol{\mu}_k + \boldsymbol{r}^0_{aki}, \qquad k = 0, 1; i = 1, \ldots, N/2,$$

where $\boldsymbol{r}^0_{aki} = \boldsymbol{T}_0^\intercal \cdot \boldsymbol{\Gamma}_{i0}$, $\boldsymbol{T}_0^\intercal \boldsymbol{T}_0 = \boldsymbol{M}_0$. The matrix $\boldsymbol{M}_0$ is created by Matern covariance function with $\boldsymbol{\rho}_r = (100, 0.3, 3)$, where the three elements represent the scale, range and smoothness, respectively (Raket, 2016). The vector $\boldsymbol{\Gamma}_{i0}$ consists of 100 independent normal random variables $N(0, \sigma_r^2)$. We can regard $\boldsymbol{\tau}_{aki}$ as the true curves $\boldsymbol{x}_{aki}(g^{-1})$.

(b) Generate the scalar variables $\boldsymbol{v}$'s. By sampling from the uniform distribution we generate scalar variables $v$

$$v_{ki} \sim \begin{cases} \mathrm{U}(1, 2), & i = 1, \ldots, N/2, \;\; k = 0, \\ \mathrm{U}(0.5, 1.5), & i = 1, \ldots, N/2, \;\; k = 1. \end{cases}$$

(c) Set coefficients $b_0 = -1$, $b_1 = 0.04$, and the coefficient functions $\beta_1(t) = 20t^2 - 14t + 1.2$, $\beta_2(t) = 30\sin(2\pi t - 1.8)$.

(d) Generate $y$'s from model (4.4). We have $N_k$ batches of data in each group, where $k = 0$ and 1, $N_0 + N_1 = N$.

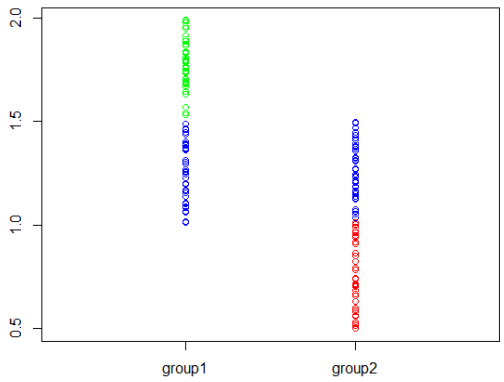



Figure 4.3: An example of the observations of scalar variable with $N = 180$. The 'blue' ones stand for those in the range of overlapping.

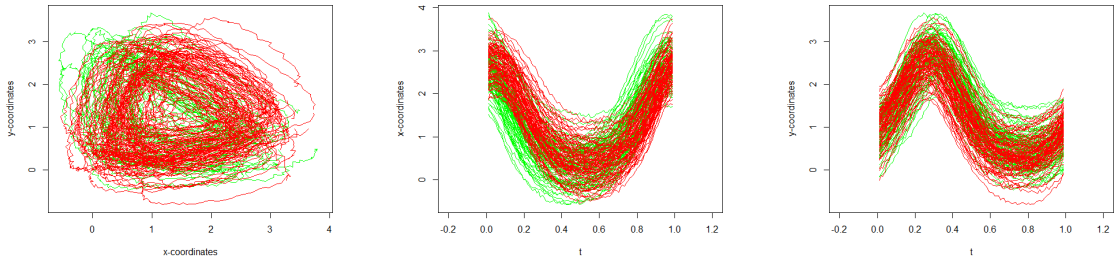
2. Generate the original 2D curves $\boldsymbol{x}(t)$ by adding the warping function and errors as follows.

   (a) Model the true mean curves $\boldsymbol{\mu}_1(t)$, $\boldsymbol{\mu}_2(t)$ and $\boldsymbol{r}^0_{aki}$ using B-spline basis function with 8 knots, resulting in the coefficients $\boldsymbol{c}_a$, $\boldsymbol{d}_{ak}$ and $\boldsymbol{d}_{aki}$ where $\sum_l \boldsymbol{d}_{al} = 0, a = 1, 2; k = 0, 1$. So, $\boldsymbol{\tau}_{aki} = \boldsymbol{\Psi}_{ki}(\boldsymbol{c}_a + \boldsymbol{d}_{ak} + \boldsymbol{d}_{aki})$.

   (b) Introduce time warping. For simplicity, we set $g_{ki}(t) = t + w_{ki}(t)$ and use hyman spline (monotone cubic spine using Hyman filtering) based on the anchor knots $t_w = (0, 0.33, 0.67, 1)$ $(n_w = 4)$. Set $\boldsymbol{w}_{ki} \sim N_4(0, \boldsymbol{T}_k^\intercal \boldsymbol{\Gamma}_i)$, where $\boldsymbol{T}_k^\intercal \boldsymbol{T}_k = \boldsymbol{O}_k$, $\boldsymbol{O}_1 = \begin{bmatrix} 10 & 4 \\ 4 & 8 \end{bmatrix}$ and $\boldsymbol{O}_2 = \begin{bmatrix} 10 & 8 \\ 8 & 15 \end{bmatrix}$, and $\boldsymbol{\Gamma}_i = (\boldsymbol{\Gamma}_{i1}, \boldsymbol{\Gamma}_{i2})^\intercal$ with $\boldsymbol{\Gamma}_{i1}, \boldsymbol{\Gamma}_{i2}$ being independent random variables $N(0, \sigma_w^2)$ for $i = 1, \ldots, N_k$. Thus, $\boldsymbol{\tau}_{aki}(g_{ki}) = \boldsymbol{\Psi}^*_{ki}(\boldsymbol{c}_a + \boldsymbol{d}_{ak} + \boldsymbol{d}_{aki})$.

   (c) Set $\boldsymbol{\epsilon} \sim N(0, \sigma^2 \mathbb{I})$ and generate $\boldsymbol{x}_{aki}(t)$ based on the model (4.2).

Figure 4.3 and Figure 4.4 show one example of the observations of the scalar variable and the functional variable, respectively, with $N = 180$ and $4\sigma_w = \sigma_r = \sigma = 0.02$. More examples can be found at Section B.1 in Appendix B.




(a) 2D curves (b) $\boldsymbol{x}_1(t)$ (c) $\boldsymbol{x}_2(t)$

Figure 4.4: An example of 2D raw curves from the scenario: $4\sigma_w = \sigma_r = \sigma = 0.02$ and $N = 180$ . Curves in green indicate the first group, i.e. $y = 0$, while those in red represent the second group, i.e. $y = 1$.

**Results**

Under the same constraints $4\sigma_w = \sigma_r = \sigma = 0.02$, we will study the performance of estimation by calculating the average bias (AB) for the coefficient $b$

$$\text{AB}(\hat{b}) = \frac{1}{100}\sum_{j=1}^{100}|\hat{b}_j - b|,$$

and the average root mean squared error (ARMSE) for the coefficient functions $\boldsymbol{\beta}$

$$\text{ARMSE}(\hat{\boldsymbol{\beta}}) = \frac{1}{100}\sum_{j=1}^{100}\sqrt{\frac{1}{100}\sum_{i=1}^{100}(\hat{\beta}_j(t_i) - \beta(t_i))^2}.$$

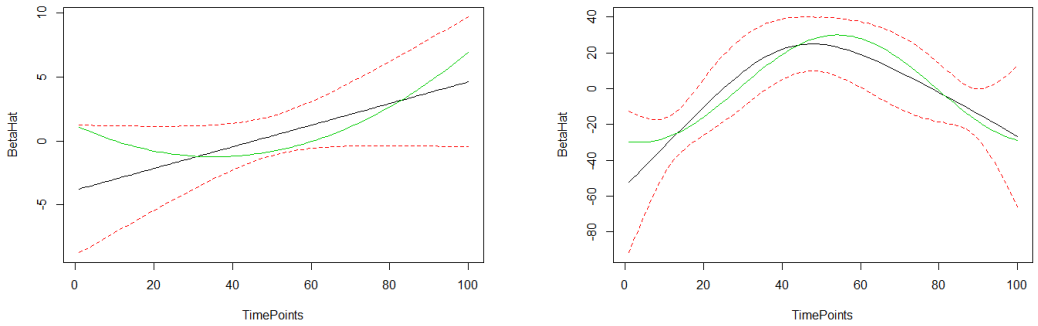
This is done over 100 replications for different sample sizes $N = 60, 90, 120, 180$.

The performances of the estimators in model (4.10) as $N$ increases is demonstrated in Table 4.1. It shows that the ARMSE of the estimators decreases while the sample size increases.

| | AB | | ARMSE | |
|---|---|---|---|---|
| | $\hat{b}_0$ | $\hat{b}_1$ | $\hat{\boldsymbol{\beta}}_1$ | $\hat{\boldsymbol{\beta}}_2$ |
| $N$=60, $K_x = 18$, $K_e = 12$ | 2.36 | 1.88 | 12.38 | 19.64 |
| $N$=90, $K_x = 30$, $K_e = 30$ | 2.27 | 1.45 | 11.50 | 16.56 |
| $N$=120, $K_x = 35$, $K_e = 35$ | 1.40 | 1.01 | 8.27 | 13.54 |
| $N$=180, $K_x = 35$, $K_e = 35$ | 1.09 | 0.92 | 7.88 | 11.91 |

Table 4.1: The average bias and average root mean squared error for the estimators as the number of subjects increases.

Also, we show one example of the 95% confidence intervals for the coefficients functions




(a) 95% confidence interval for $\hat{\beta}_1(t)$ (b) 95% confidence interval for $\hat{\beta}_2(t)$

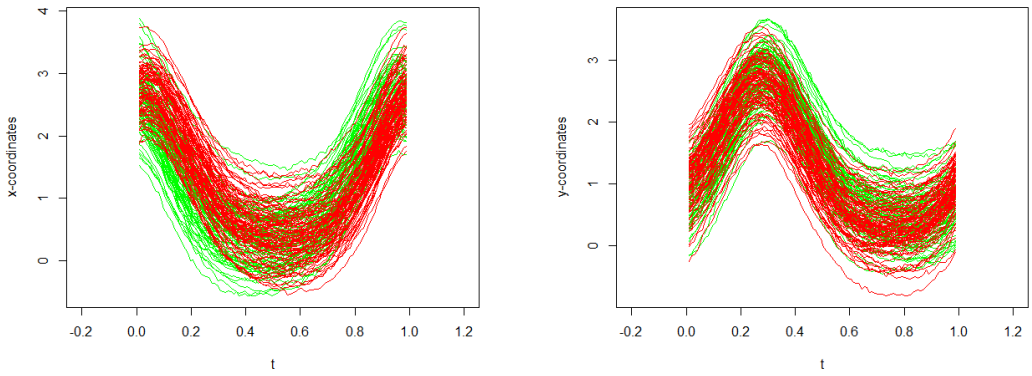
Figure 4.5: An example of confidence intervals for $\hat{\boldsymbol{\beta}}(t)$ from the scenario: $N = 180, K_x = 35, K_e = 35$. The lines in green are the true $\boldsymbol{\beta}$, the lines in black stand for the estimators $\hat{\boldsymbol{\beta}}$, and the dotted red lines represent the boundaries of 95% confidence intervals. 'TimePoints' are equal to $100t_j$.

$\boldsymbol{\beta}(t)$ and the distribution of $\hat{\pi}$ from the fourth scenario $N = 180, K_x = 35, K_e = 35$ in Figure 4.5 and Figure 4.7, respectively. Additionally, the result of registration for the example in Figure 4.4 is demonstrated in Figure 4.6. Extra numerical results can be seen at Section B.2 and B.3 in Appendix B.

### 4.3.2 Simulation study 2

The performance of the proposed *JCRC* models in terms of prediction will be evaluated in this simulation study. Meanwhile, we will compare it with the model defined in (4.1) without scalar variables (denoted by *JCRC-f*), and the simple logistic linear regression model without functional variables (denoted by *LLR*). We also compare them with the curve classification based on the square-root velocity representation for analyzing shapes of curves (Srivastava et al., 2011a) (denoted by *SRV*) and the integration of Generalized Procrustes analysis (Gower, 1975b) and self-modeling method (Gervini and Gasser, 2004) (denoted by *GPSM*). The principal of curve classification and the details of procedures for *GPSM* method has been discussed in Chapter 3.

We still consider 2D curves coming from two groups. For each group, the corresponding observations of functional variables $\boldsymbol{x}(t)$ and scalar variables $v$ will be generated. There are $N_k$ batches of data in each group, where $k = 0, 1$. We will first evaluate and compare five methods based on the simulated data $D = \{(y_{ki}, x_{1ki}(t_{ij}), x_{2ki}(t_{ij}), \boldsymbol{v}_{ki}); i = 1, \ldots, N_k; j = 1, \ldots, m_{ki}; k = 0, 1\}$ in two scenarios; see Scenario A and Scenario B in Table 4.2. Additionally, one of the data settings in the previous simulation study will also be used here as the third scenario (Scenario C).




(a) Aligned $\boldsymbol{x}_1(t)$ (b) Aligned $\boldsymbol{x}_2(t)$

Figure 4.6: The curves after registration by *JCRC*, corresponding to the raw curves in Figure 4.4.

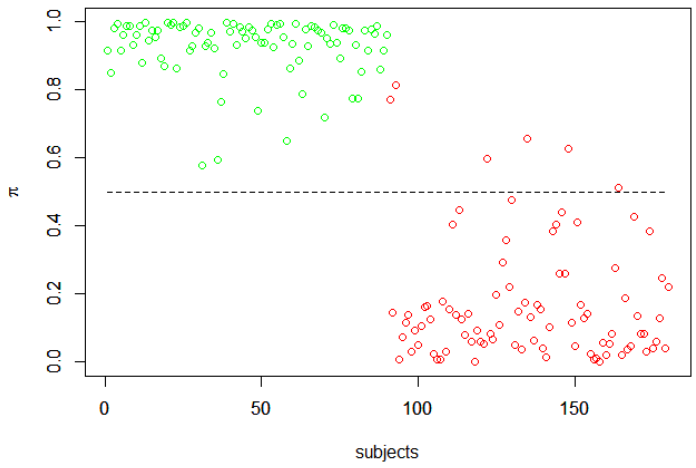
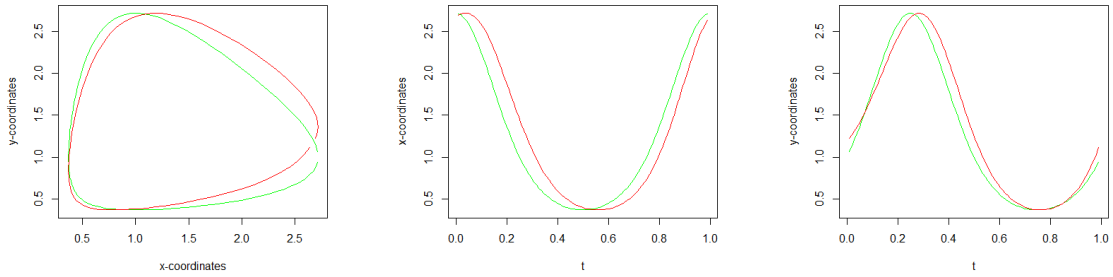



Figure 4.7: An example of the distribution of $\hat{\pi}$ from the scenario: $N = 180, K_x = 35, K_e = 35$. Circles in green indicate the first group, i.e. $y = 0$ while those in red represent the second group $y = 1$. The dotted line in black in the middle represents $\pi = 0.5$.




(a) 2D curves (b) $\boldsymbol{\mu}_1(t)$ (c) $\boldsymbol{\mu}_2(t)$

Figure 4.8: True mean curves for $\delta_1 = 0.18$. Curves in green indicate the first group ($y = 0$) while those in red represent the second group ($y = 1$).

**Data generation**

1. Generate the underlying true curves, i.e. the curves based on the internal time scale $g^{-1}(t)$. We first assume that the curves in two groups share the following two slightly different true means, which is similar to two patterns of those real curves of hyoid bone's movement

$$\begin{aligned}\boldsymbol{\mu}_1(t) &= \big(\mu_{11}(t), \mu_{21}(t)\big) = \big(\exp\{\cos(2\pi t)\}, \exp\{\sin(2\pi t)\}\big),\\ \boldsymbol{\mu}_2(t) &= \big(\mu_{12}(t), \mu_{22}(t)\big) = \big(\exp\{\cos(2\pi t^{1.1} - \delta_1)\}, \exp\{\sin(2\pi t^{1.2} + \delta_1)\}\big).\end{aligned} \tag{4.12}$$

   The degree of overlapping between two groups relies on the value of $\delta_1$. The smaller the value of $b_1$, the higher the degree of overlapping, and the harder it is to classify those curves. We use the equidistant points $t_j = \frac{j+1}{102}, j = 1, \ldots, 100$ as the input grid, i.e. $m_{ki} = 100$. The underlying true curves are generated by

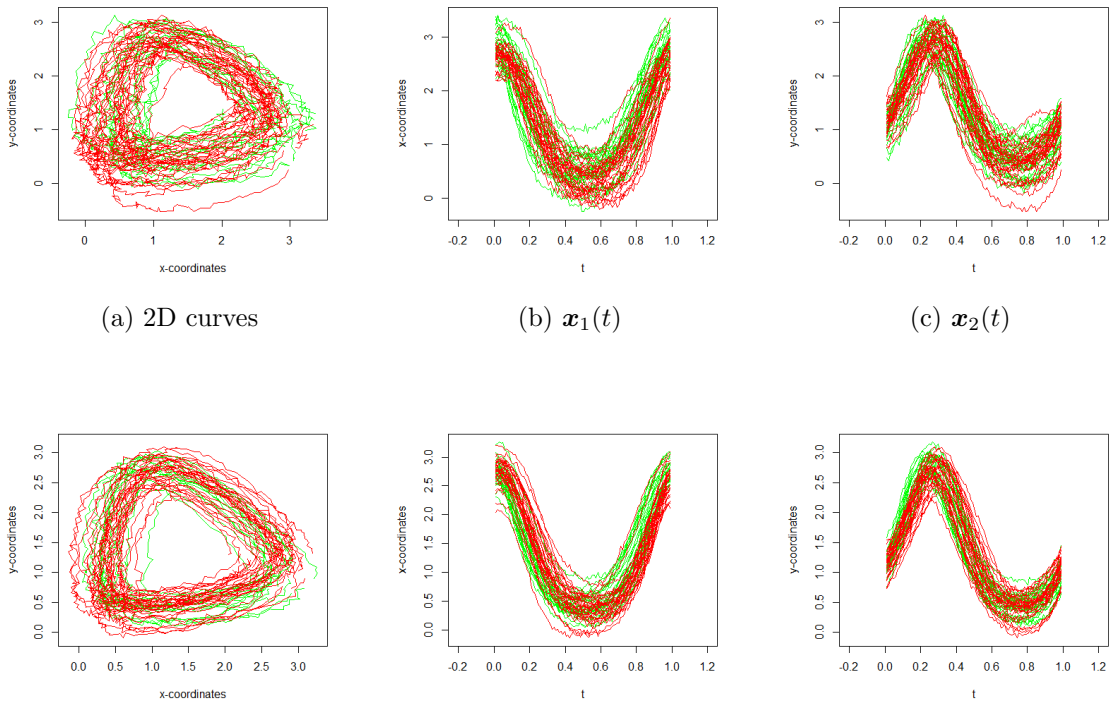
$$\boldsymbol{x}_{ak}(g^{-1}(t)) = \mu_{ak}(t), \qquad a = 1, 2; \;\; k = 0, 1.$$

   Figure 4.8 shows the shape of the true mean curves for $\delta_1 = 0.18$.

2. Generate the original 2D curves $\boldsymbol{x}(t)$ by adding the warping function, amplitude variation and errors as follows.

   (a) Model the true curves $\boldsymbol{x}_{ak}(g^{-1}(t))$ using B-spline basis function with 8 knots and obtain the coefficients $\boldsymbol{c}_a$ and $\boldsymbol{d}_{ak}$ where $\sum_k \boldsymbol{d}_{ak} = 0, a = 1, 2; k = 0, 1$.

   (b) Introduce time warping. For simplicity, we set $g_{ki}(t) = t + w_{ki}(t)$ and use hyman spline (monotone cubic spine using Hyman filtering) based on the anchor knots $t_w = (0, 0.33, 0.67, 1)$ ($n_w = 4$). Set $\boldsymbol{w}_{ki} \sim N_4(0, \boldsymbol{T}_k^\intercal \boldsymbol{\Gamma}_i)$, where $\boldsymbol{T}_k^\intercal \boldsymbol{T}_k = \boldsymbol{O}_k$,




(a) 2D curves (b) $\boldsymbol{x}_1(t)$ (c) $\boldsymbol{x}_2(t)$

(d) 2D curves (e) $\boldsymbol{x}_1(t)$ (f) $\boldsymbol{x}_2(t)$

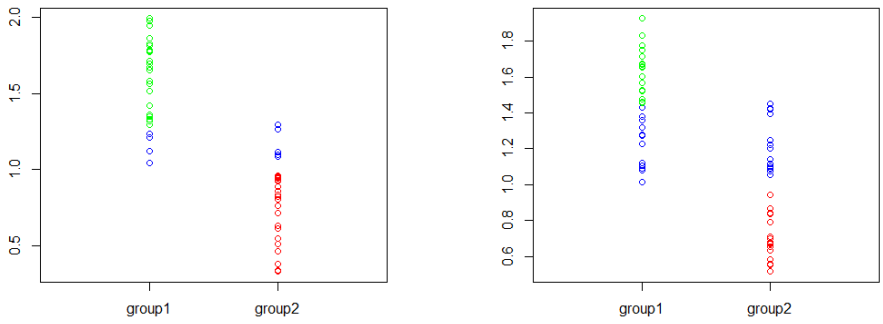
Figure 4.9: (a)-(c): an example of raw curves for Scenario A with $\delta_1 = 0.18$, $4\sigma_w = \sigma_r = \sigma = 0.03$; (d)-(f): an example of raw curves for Scenario B with $\delta_1 = 0.15$, $4\sigma_w = \sigma_r = \sigma = 0.02$. Curves in green indicate the first group ($y = 0$) while those in red represent the second group ($y = 1$).

$\boldsymbol{O}_1 = \begin{bmatrix} 10 & 4 \\ 4 & 8 \end{bmatrix}$ and $\boldsymbol{O}_2 = \begin{bmatrix} 10 & 8 \\ 8 & 15 \end{bmatrix}$, and $\boldsymbol{\Gamma}_i = (\boldsymbol{\Gamma}_{i1}, \boldsymbol{\Gamma}_{i2})^\intercal$ with $\boldsymbol{\Gamma}_{i1}, \boldsymbol{\Gamma}_{i2}$ being independent random variables $N(0, \sigma_w^2)$ for $i = 1, \ldots, N_k$.

(c) Set the amplitude variation $\boldsymbol{r}_{aki} = \boldsymbol{T}_0^\intercal \cdot \boldsymbol{\Gamma}_{i0}$, where $\boldsymbol{T}_0^\intercal \boldsymbol{T}_0 = \boldsymbol{O}_0$, $a = 1, 2$, $k = 0, 1$ and $i = 1, \ldots, N_k$. The matrix $\boldsymbol{O}_0$ is created by Matern covariance function with $\boldsymbol{\rho}_r = (100, 0.3, 3)$, where the three elements represent the scale, range and smoothness, respectively (Raket, 2016), and $\boldsymbol{\Gamma}_{j0}$ is a vector of 100 independent normal random variables $N(0, \sigma_r^2)$. Set $\boldsymbol{\epsilon} \sim N(0, \sigma^2 \mathbb{I})$.

(d) Generate $\boldsymbol{x}(t)$ based on the model (4.3). Figure 4.9 shows two examples of raw data.

3. Generate $\boldsymbol{v}$'s. We next generate those scalar variables $v$ by sampling from uniform




(a) $\delta_2 = 0.7$ (b) $\delta_2 = 0.5$

Figure 4.10: Observations of scalar variable in two groups. The 'blue' ones stand for those in the range of overlapping.

distribution as follows:

$$V_i \sim \begin{cases} \mathrm{U}(1,2), & i = 1, 2, \ldots, N_0, \\ \mathrm{U}(1-\delta_2, 2-\delta_2), & i = N_0+1, \ldots, N_0+N_1. \end{cases}$$

Note that as $\delta_2$ becomes larger, the degree of overlapping becomes smaller. Hence it becomes easier to carry out classification using scalar variables. Figure 4.10 shows two examples of $\boldsymbol{v}$'s with $\delta_2 = 0.7$ and 0.5.

**Results**

To investigate how the overlapping of the observations of both scalar variables and curves affects the performance of fitting and prediction, we study two scenarios: (A) $\delta_1 = 0.18$, $\delta_2 = 0.7$, $4\sigma_w = \sigma_r = \sigma = 0.03$ (Figure 4.9 (a)-(c) and Figure 4.10 (a)) ; (B)$\delta_1 = 0.15$, $\delta_2 = 0.5$, $4\sigma_w = \sigma_r = \sigma = 0.02$ (Figure 4.9 (d)-(f) and Figure 4.10 (b)). Both scenarios are studied under the same constraints $N_0 = N_1 = 60$, half for training data set and half for test data set. There are 100 replications for each scenario. Also, we study the scenario: (C) using the data setting in the previous simulation study with constraints $4\sigma_w = \sigma_r = \sigma = 0.02$ and $N = 120$ (half for training and half for test). We use three criteria to measure the performance of clustering. These are classification accuracy (CA), the Rand index (RI) (Rand, 1971b) and adjusted Rand index (ARI) (Hubert and Arabie, 1985b) are used to measure the performance of classification (or prediction).

We set $K_x = 18$ and $K_e = 10$ for Scenario A and Scenario B and $K_x = 18, K_e = 12$ for Scenario C by 5-fold Cross-Validation method. Table 4.2 summarizes the comparison of

average classification results by CA, RI and ARI. Firstly, we see the values of CA, RI and ARI for method *JCRC* are much higher than the other four in all scenarios. It shows our proposed method *JCRC* outperforms the rest in terms of classification. The reason is that the combination of functional variables and scalar variables is more helpful in classifying the subjects than using only the functional variables or scalar variables. Specifically, the CA values by *LLR* only based on scalar variable are 0.85, 0.75 and 0.75, are much less than 0.95, 0.94 and 0.91 by *JCRC* in three scenarios, respectively. The results of curve classification, depending on functional variables only, are all less than 0.8 except those in Scenario C (less than 0.86), by method *JCRC-f*, *GPSM* and *SRV*. Secondly, on one hand, as the overlapping of $v$'s increases from Scenario A to Scenario B (or Scenario C), contributing to harder differentiation of which group each subject belongs to, the method *LLR* performs worse with the value of CA decreasing from 0.85 to 0.75. On the other hand, because of the decrease of noise determined by $\sigma_w$, $\sigma_r$ and $\sigma$, *JCRC-f*, *GPSM* and *SRV* perform better from Scenario A to Scenario C. In whatever case, *JCRC* has the most stable and best results. Figure 4.11 shows the registration of raw curves from Figure 4.9 by method *JCRC* in Scenario A and Scenario B.

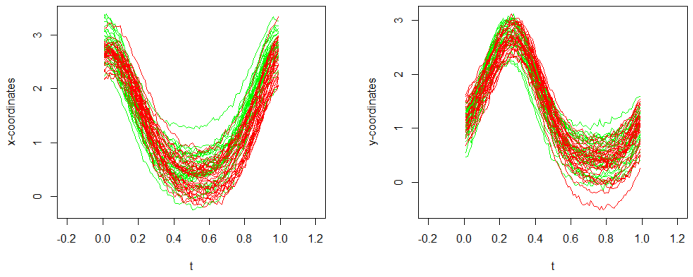
Other combinations with varying sample sizes, distinct overlapping determined by $\delta_1$ for functional variables and $\delta_2$ by scalar variables and different $\sigma_w$, $\sigma_r$ and $\sigma$ have also been examined. The results presented here are typical.

| | Scenario A | | | Scenario B | | | Scenario C | | |
|---|---|---|---|---|---|---|---|---|---|
| | CA | RI | ARI | CA | RI | ARI | CA | RI | ARI |
| *JCRC* | **0.95** | **0.90** | **0.81** | **0.94** | **0.89** | **0.79** | **0.91** | **0.86** | **0.71** |
| *LLR* | 0.85 | 0.78 | 0.49 | 0.75 | 0.63 | 0.25 | 0.75 | 0.63 | 0.25 |
| *JCRC-f* | 0.70 | 0.58 | 0.16 | 0.78 | 0.66 | 0.32 | 0.86 | 0.77 | 0.54 |
| *GPSM* | 0.73 | 0.61 | 0.22 | 0.76 | 0.64 | 0.27 | 0.86 | 0.77 | 0.54 |
| *SRV* | 0.56 | 0.52 | 0.03 | 0.58 | 0.52 | 0.04 | 0.63 | 0.55 | 0.11 |

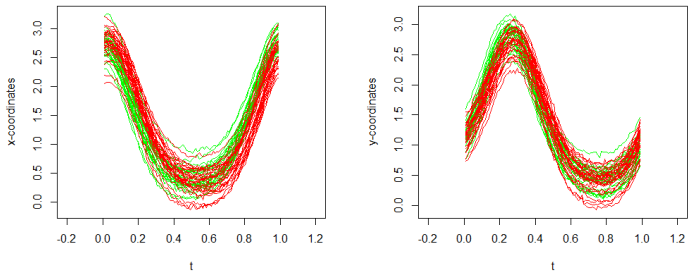
Table 4.2: Comparison of average classification results among five methods.

### 4.3.3 Real data analysis

The application to real data is to carry out the classification for normal people and patients with stroke by modelling the trajectories of their hyoid bone movement and the other scalar variables. The data set contains two groups, one for normal people and the other for patients. Figure 4.1(a) and 4.1(b) show one frame from a X-ray video clip and raw curves with 15 people in each group, respectively. The scalar variables we choose are motion time (duration), average velocity and acceleration amplitude of those curves. By using the package of *GPA*, we do the preprocessing for those 2D raw curves first. The procedures include multi-dimensional shift, scaling and rotation, as mentioned in Section




(a) Aligned curves from Scenario A




(b) Aligned curves from Scenario B

Figure 4.11: The aligned curves for both scenarios corresponding to raw data from Figure 4.9. Curves in green indicate the first group ($y = 0$) while those in red represent the second group ($y = 1$).

3.2.1 of Chapter 3. In the part of modelling those curves, we assume the warping function be a smooth nonlinear deformation produced by an increasing spline and the random vector $\boldsymbol{w}_{ki}$ be a Brownian bridge observed at discrete anchor points. B-spline basis functions are utilized for modeling the mean curves. The covariance function for the amplitude variance is the Matern covariance function. For our proposed method *JCRC*, we evaluate the classification performance by 5-fold cross-validation. It means 12 samples are trained and the remaining 3 are tested for each group. The results are shown in Table 4.3. We see that the method *JCRC* outperforms the other methods.

| Methods | CA | RI | ARI |
|---|---|---|---|
| *JCRC* | **0.76** | **0.69** | **0.39** |
| *LLR* | 0.63 | 0.45 | 0 |
| *JCRC-f* | 0.50 | 0.43 | 0 |
| *GPSM* | 0.57 | 0.48 | 0.06 |
| *SRV* | 0.50 | 0.43 | -0.11 |

Table 4.3: Average classification results of three measurements for five methods. The results by *GPSM* and *SRV* are from the real data analysis of Chapter 3.

## 4.4 Chapter Summary

We have proposed two-stage models for joint curve registration and classification (*JCRC*), with the first stage fitting the logistic functional linear regression model and the second stage modelling the multi-dimensional curves with the misaligned problems. The prediction of misaligned curves acquired in the first stage will be used in the second stage. The estimation and implementation of two-stage models are provided. We also developed an iterative algorithm to predict the outcomes. Numerical results show the superiority of our proposed model. The main contributions include:

(a) simultaneously carrying out registration and modeling for multi-dimensional functional data,

(b) the use of both functional and scalar covariates while conducting classification.

The methodology discussed in this chapter is just for the purpose of classification, i.e. supervised learning in the computing community. How about the task of clustering for multi-dimensional functional data? In the area of unsupervised learning, it is much more difficult than classification due to the lack of response value $y$. We will study the problem of simultaneous registration and clustering in the next chapter.

# Chapter 5

# Simultaneous Registration and Clustering for Multi-dimensional Functional Data

## 5.1 Introduction

As mentioned in the last chapter, in our study of the motion analysis of hyoid bone, there exist obvious misaligned problems for those curves in both vertical and horizontal variation (see Figure 4.1 in Chapter 4). Usually, curve registration is implemented as a preprocessing technique and the clustering is conducted afterwards. It is not efficient, since a subject belonging to which cluster is closely related to how it unfolds its progression pace. Another challenging problem for this study is that the heterogeneity of regression relationships among different groups. It consists in both the subjects' scalar covariates and the potential time warping for curves corresponding to the subjects. These scalar covariates include, but not restricted to, the initial level of disease, gender, age and the characteristics of those trajectories themselves, like motion time, average speed and range of motion. Therefore, simultaneous curve registration and clustering by considering all those factors seems to be a better way for modeling the functional data. There are some research work on handling the similar problems. For instance, Wu and Hitchcock (2016) proposed a Bayesian method for simultaneous registration and clustering for functional data. They used a discrete approximation generated from the family of Dirichlet distributions to allow warping functions of great flexibility. Liu and Yang (2009) developed a framework that allows for simultaneously aligning and clustering k-centers functional data. But their model did not use any subject specific information (scalar variables) and assumed the heterogeneity among groups just depends on the curves themselves; similar idea is also used in $k$-means alignment for curve clustering by Sangalli et al. (2010). On

the other hand, Shi and Wang (2008) proposed a hierarchical mixture of Gaussian process (GP) functional regression models with an allocation model to do curve prediction and clustering. They used the functional covariates to reconstruct the response curve and the personal scalar variables, such as height and gender, to deal with heterogeneity of the regression relationships among different groups. However, their method did not consider the misaligned problem. In addition, most related models are limited to one dimensional curves. To address the above problems, we try to construct one hierarchical mixture of models for the sake of simultaneous curve registration and clustering.

This chapter is organized as follows. Section 5.2.1 defines simultaneous registration and clustering (*SRC*) models via two-level models. We discuss the estimation and the details of implementation in Section 5.2.2 and Section 5.2.3 respectively. The problem of model selection and the related methods are discussed in Section 5.2.4. Section 5.3 presents a number of examples with simulated data and real data. A short summary and discussion are given in Section 5.4.

## 5.2 The simultaneous registration and clustering method

Suppose there are $N$ subjects coming from $K$ different groups, $\boldsymbol{x}_1(t), \boldsymbol{x}_2(t), \ldots, \boldsymbol{x}_N(t)$ being the observations of 2D continuous curves, where $\boldsymbol{x}_i(t) = (x_{1i}(t), x_{2i}(t))^\intercal$, $x_{1i}(t)$ and $x_{2i}(t)$ are the corresponding $x$-coordinates and $y$-coordinates of $\boldsymbol{x}_i(t)$. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N$ be the observed scalar variables. Suppose there are $m_i$ time points on which the $i$-th curve is measured. The data set is

$$D = \{(\boldsymbol{x}_i(t_{ij}), \boldsymbol{v}_i); i = 1, \ldots, N; j = 1, \ldots, m_i\}.$$

We introduce a latent indicator variable $\boldsymbol{z}_i = (z_{1i}, \ldots, z_{Ki})^\intercal$ for the $i$-th subject where $z_{ik}$ takes value 1 if they are in the $k$-th group and 0 otherwise.

### 5.2.1 The model

In our study of 2D curves, we will use the preprocessing procedure Generalized Procrustes Analysis (GPA) (Gower, 1975b) to address part of registration problems in advance except warping. Conventionally, most methods tried to complete all the registration problems including warping before clustering. This is not the best way since different warping functions may need to be used in different clusters, yet we have no such information before clustering, and heterogeneity among different subjects should also be considered. Thus, a hierarchical structure defined by two levels of models is proposed.

We start with the first level model for the continuous curve as follows

$$x_{ai}(t)|_{z_{ki}=1} = (\tau_{ak} \circ g_{ki})(t) + r_{aki}(t) + \epsilon_{ai}(t), \quad i = 1, \ldots, N, \tag{5.1}$$

where $a = 1$ or 2 represents $x$- or $y$-coordinates of $\boldsymbol{x}_i(t)$. The item $(\tau_{ak} \circ g_{ki})$ denotes functional composition: $(\tau \circ g)(t) = \tau(g(t))$, where $g_{ki}(t)$ is the inverse of a warping function. $\tau_{ak}(\cdot)$ is a fixed but unknown nonlinear mean curve, which can be approximated by a set of basis functions, the details will be given in the next subsection. The variation among different subjects is modeled by a non-linear functional random-effects, $r_{aki}(t)$, by a Gaussian process with zero-mean and a parametric covariance function $\boldsymbol{S}$ (Shi et al., 2012). The error item $\epsilon_{ai}(t)$ is assumed to be Gaussian white noise with variance $\sigma^2$. Following the previous discussion, we need to use different warping function in different cluster, and we also need to consider the variation among different subjects, and thus, we allow warping function depending on $k$ and $i$. Using the same assumption of the inverse of warping function $g_{ki}(t)$ in the previous chapter, we assume

$$g_{ki}(t) = t + w_k(t) + w_{ki}(t),$$

where $w_k(t)$ is the fixed part and $w_{ki}(t)$ is the random part in terms of different subjects. We then discretize them by a set of fixed parameters, for example, by $\boldsymbol{w}_k = (w_k(t_1), \ldots, w_k(t_{n_w}))$ and $\boldsymbol{w}_{ki} = (w_{ki}(t_1), \ldots, w_{ki}(t_{n_w}))$ respectively. $\boldsymbol{w}_{ki}$ are modelled by a Gaussian distribution with zero mean and a parametric covariance function $\boldsymbol{H}$.

We define a logistic allocation model in the second level model for the latent indicator variable in the form

$$p(z_{ki} = 1) = \pi_{ki} = \frac{\exp\{\boldsymbol{v}_i^\intercal \boldsymbol{\beta}_k\}}{1 + \sum_{j=1}^{K-1} \exp\{\boldsymbol{v}_i^\intercal \boldsymbol{\beta}_j\}}, \quad i = 1, \ldots, N; \quad k = 1, \ldots, K-1, \tag{5.2}$$

with $p(z_{Ki} = 1) = \pi_{Ki} = 1 - \sum_{l=1}^{K-1} \pi_{li}$, where $\{\boldsymbol{\beta}_k, k = 1, \ldots, K-1\}$ are unknown parameters to be estimated. We can also replace model (5.2) by other models, e.g. Potts model (Green and Richardson, 2000). The information of scalar variables is integrated with functional variables via the two-level models (5.1) and (5.2). The reason of using both types of variables is that the variation between subjects does not usually depend on the curve data only, summary statistics or some subject-specific variables do provide useful information, like the scenario in Figure 5.7 and Figure 5.8 in the simulated example and Table 5.2 and Figure 5.12 in the real data analysis. The introduction of the latent indicator variable is very useful in the implementation; see the details below.

We call the models defined in (5.1) and (5.2) as simultaneous registration and clustering ($SRC$) models.

### 5.2.2 Estimation

The discrete form of model (5.1) for the $i$th curve data $\boldsymbol{x}_{ai} = \big(x_{ai}(t_{i1}), \ldots, x_{ai}(t_{im_i})\big)^\intercal$ can be expressed as follows

$$\boldsymbol{x}_{ai}|_{z_{ki}=1} = \boldsymbol{\tau}_{ak}(g_{ki}) + \boldsymbol{r}_{aki} + \boldsymbol{\epsilon}_i, \quad \text{for } a = 1, 2; \;\; k = 1, \ldots, K, \tag{5.3}$$

where $\boldsymbol{\tau}_{ak}(g_{ki}) = \Big(\tau_{ak}\big(g_{ki}(t_{i1})\big), \ldots, \tau_{ak}\big(g_{ki}(t_{im_i})\big)\Big)^\intercal$. Similar to the last chapter, we still set $\boldsymbol{H}$ as Brownian covariance function or unstructured covariance function with parameter $\boldsymbol{\rho}_h$ (Raket, 2016) and let $\boldsymbol{H}_{ki}$ be the covariance matrix of $\boldsymbol{w}_{ki}$. We then model $\tau_{ak}(t)$ using $q$ basis functions $\{\psi_1(t), \ldots, \psi_q(t)\}$ with weights $\boldsymbol{d}_{ak} = (d_{ak1}, \ldots, d_{akq})^\intercal$. Thus, $\boldsymbol{\tau}_{ak}(g_{ki}) = \boldsymbol{\Psi}_{ki}\boldsymbol{d}_{ak}$ where $\boldsymbol{\Psi}_{ki} = [\boldsymbol{\Psi}_{ki1}, \ldots, \boldsymbol{\Psi}_{kiq}]_{m_i \times q}, \boldsymbol{\Psi}_{kil} = (\psi_l(g_{ki}(t_{i1})), \ldots, \psi_l(g_{ki}(t_{im_i})))^\intercal, l = 1, \ldots, q$. We still use a smooth non-linear deformation produced by a cubic Hermite spline (Raket, 2016) for the curves. $\boldsymbol{r}_{aki}$ and $\boldsymbol{\epsilon}_i$ are both $m_i$-dimensional column vector. We set $\boldsymbol{S}$ as the Matern covariance function with parameters $\boldsymbol{\rho}_s$ and let $\boldsymbol{S}_{aki}$ be the covariance matrix of $\boldsymbol{r}_{aki}$. $\boldsymbol{\rho}_h$ and $\boldsymbol{\rho}_s$ can be estimated by the data, and $\boldsymbol{H}_{ki}$ and $\boldsymbol{S}_{aki}$ can be calculated by the corresponding covariance function; see the details in the next subsection.

The unknown parameters from the $k$-th component for the $a$-coordinates ($x$- or $y$-coordinates) of the $i$-th curve are denoted by $\boldsymbol{\theta}_{aki} \stackrel{\Delta}{=} \{\boldsymbol{d}_{ak}, \boldsymbol{w}_k, \boldsymbol{w}_{ki}, \boldsymbol{\rho}_s, \boldsymbol{\rho}_h, \sigma\}$. Let $\boldsymbol{\theta}_{ki}$ be the vector of $\{\boldsymbol{\theta}_{aki}, a = 1, 2\}$. We can similarly define $\boldsymbol{\theta}_i = \{\boldsymbol{\theta}_{ki}, k = 1, \ldots, K\}$, $\boldsymbol{\theta} = \{\boldsymbol{\theta}_i, i = 1, \ldots, N\}$ and $\boldsymbol{\beta} = \{\boldsymbol{\beta}_k, k = 1, \ldots, K-1\}$. The Gaussian mixture distribution for the $i$-th curve data can be written in the form

$$p(\boldsymbol{x}_i|\boldsymbol{\theta}_i, \boldsymbol{\beta}) = \sum_{k=1}^{K} \pi_{ki} p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki}), \quad i = 1, \ldots, N,$$

where $p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki}) = p(\boldsymbol{x}_{1i}|\boldsymbol{\theta}_{1ki})p(\boldsymbol{x}_{2i}|\boldsymbol{\theta}_{2ki})$. We assume $\boldsymbol{x}_{1i}$ and $\boldsymbol{x}_{2i}$ are conditional independent given those parameters. The log-likelihood of $(\boldsymbol{\theta}, \boldsymbol{\beta})$ is therefore

$$l(\boldsymbol{\theta}, \boldsymbol{\beta}) = \sum_{i=1}^{N} \log\Big\{ \sum_{k=1}^{K} \pi_{ki} p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki}) \Big\}.$$

It is quite tricky to conduct the estimation due to the large number of unknown parameters. EM algorithm will be adopted in this paper. We have defined the latent indicator variable $\boldsymbol{z}_i$, which is treated as missing. The joint likelihood function of $\boldsymbol{x}$ and $\boldsymbol{z}$, where $\boldsymbol{z} = \{\boldsymbol{z}_i; i = 1, \ldots, N\}$, takes the form

$$p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}, \boldsymbol{\beta}) = p(\boldsymbol{z}|\boldsymbol{\beta})p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\theta}) = \prod_{i=1}^{N} \prod_{k=1}^{K} \pi_{ki}^{z_{ki}} p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})^{z_{ki}}.$$

Taking the logarithm, we have the log-likelihood for complete data $(\boldsymbol{x}, \boldsymbol{z})$

$$l_c(\boldsymbol{\theta}, \boldsymbol{\beta}) = \log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}, \boldsymbol{\beta}) = \sum_{i=1}^{N}\sum_{k=1}^{K} z_{ki}\Big(\log\pi_{ki} + \log p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})\Big). \tag{5.4}$$

The expected value of the complete log-likelihood with respect to $\boldsymbol{z}$ is given by

$$\begin{aligned} E_{\boldsymbol{z}}\{l_c(\boldsymbol{\theta}, \boldsymbol{\beta})\} &= \sum_{k=1}^{K}\sum_{i=1}^{N} E(z_{ki}|\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{\beta})\Big(\log\pi_{ki} + \log p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})\Big) \\ &= \sum_{k=1}^{K}\sum_{i=1}^{N} M_{ki}\Big(\log\pi_{ki} + \log p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})\Big), \end{aligned} \tag{5.5}$$

where

$$M_{ki} \triangleq E(z_{ki}|\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{\beta}) = \frac{\pi_{ki} p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})}{\sum_{j=1}^{K}\pi_{ji} p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ji})}, \quad i = 1, \ldots, N; k = 1, \ldots, K.$$

The derivation of $M_{ki}$ is given by C.1 in Appendix C. The procedure of EM algorithm includes

1. Initialise $\boldsymbol{\theta}^{(i_0)}$ and $\boldsymbol{\beta}^{(i_0)}$ and evaluate the $M_{ki}$ (E-step)

$$M_{ki} = \frac{\pi_{ki}^{(i_0)} p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki}^{(i_0)})}{\sum_{j=1}^{K}\pi_{ji}^{(i_0)} p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ji}^{(i_0)})}.$$

2. Fix $M_{ki}$ and maximize $Q(\boldsymbol{\theta}, \boldsymbol{\beta})$ with respect to $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$

$$Q(\boldsymbol{\theta}, \boldsymbol{\beta}) \triangleq \sum_{k=1}^{K}\sum_{i=1}^{N} M_{ki}\Big(\log\pi_{ki} + \log p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})\Big),$$

   leading to $\boldsymbol{\theta}^{(i_0+1)}$ and $\boldsymbol{\beta}^{(i_0+1)}$ (M-step).

The technical details are given in the next subsection.

### 5.2.3 Implementation

In E-step, we first initialize the weights $M_{ki}$. In practice, we choose $M_{ki}^{(0)} \sim U(0, 1)$ for the purpose of simplicity. Each $M_{ki}$ is then divided by their summation $\sum_{k=1}^{K} M_{ki}$ and we set $\pi_{ki}^{(0)} = M_{ki}^{(0)}$. In M-step, there are no analytic solutions to the maximization of $Q(\boldsymbol{\theta}, \boldsymbol{\beta})$ with respect to $\boldsymbol{\theta}$, so that we use the following algorithms. Maximizing $Q(\boldsymbol{\theta}, \boldsymbol{\beta})$

with respect to $\boldsymbol{\theta}$ given the current weights $M_{ik}$ is equivalent to maximizing

$$\sum_{k=1}^{K}\sum_{i=1}^{N} M_{ki}\Big(\sum_{a=1}^{2}\big(\log p(\boldsymbol{x}_{ai}|\boldsymbol{\theta}_{aki})\big)\Big).$$

All the parameters within $\boldsymbol{\theta}$ are estimated iteratively through three conditional models. The ideas are given in Section 4.2.3 of Chapter 4. In order to simplify the likelihood computations, all the random effects are scaled by a noise standard deviation $\sigma$.

**(i) Estimate the fixed effects $\boldsymbol{\tau}_{ak}$**

Given $\boldsymbol{\theta}^{(i_0)}$, we have $\boldsymbol{x}_{ai}|_{z_{ki}=1} \sim N_{m_i}(\boldsymbol{\Psi}_{ki}\boldsymbol{d}_{ak}, \boldsymbol{I}_{m_i} + \boldsymbol{S}_{aki}), a = 1, 2$ and $i = 1, \ldots, N$. The negative log likelihood for the weights $\boldsymbol{d}_{ak}$ (its square magnitude is penalized by a weighting factor $\eta$) is proportional to

$$l(\boldsymbol{d}_{ak}) = \sum_{i=1}^{N} M_{ki}||\boldsymbol{x}_{ai} - \boldsymbol{\Psi}_{ki}\boldsymbol{d}_{ak}||^2_{\boldsymbol{I}_{m_i}+\boldsymbol{S}_{aki}} + \eta\boldsymbol{d}_{ak}^{\intercal}\boldsymbol{d}_{ak}, \quad a = 1, 2; \quad k = 1, \ldots, K.$$

This gives the estimator

$$\hat{\boldsymbol{d}}_{ak} = (\boldsymbol{\Psi}_k^{\intercal}(\frac{\boldsymbol{I}_m + \boldsymbol{S}_{ak}}{\boldsymbol{M}_k})^{-1}\boldsymbol{\Psi}_k + \eta\boldsymbol{I}_q)^{-1}\boldsymbol{\Psi}_k^{\intercal}(\frac{\boldsymbol{I}_m + \boldsymbol{S}_{ak}}{\boldsymbol{M}_k})^{-1}\boldsymbol{x}_a, \quad a = 1, 2; \quad k = 1, \ldots, K,$$

where $\boldsymbol{\Psi}_k = [\boldsymbol{\Psi}_{k1}^{\intercal}, \ldots, \boldsymbol{\Psi}_{kN}^{\intercal}]^{\intercal} \in \boldsymbol{R}^{m\times q}$, $m = \sum_{i=1}^{N} m_i$, $\boldsymbol{x}_a = (\boldsymbol{x}_{a1}^{\intercal}, \ldots, \boldsymbol{x}_{aN}^{\intercal})^{\intercal}$ and

$$\frac{\boldsymbol{I}_m + \boldsymbol{S}_{ak}}{\boldsymbol{M}_k} \triangleq \begin{bmatrix} (\boldsymbol{I}_{m_1} + \boldsymbol{S}_{ak1})/M_{k1} & & \\ & \ddots & \\ & & (\boldsymbol{I}_{m_N} + \boldsymbol{S}_{akN})/M_{kN} \end{bmatrix} \in \mathbb{R}^{m\times m}. \tag{5.6}$$

**(ii) Estimate warping parameters $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$**

Given $\boldsymbol{\theta}^{(i_0)}$ and $\hat{\boldsymbol{d}}_{ak}$, we have the joint probability density function of $(\boldsymbol{x}_{ai}, \boldsymbol{w}_{ki})$ given by

$$p(\boldsymbol{x}_{ai}, \boldsymbol{w}_{ki}) = p(\boldsymbol{x}_{ai}|\boldsymbol{w}_{ki}) * p(\boldsymbol{w}_{ki}) \sim N_{m_i}(\boldsymbol{\Psi}_{ki}\hat{\boldsymbol{d}}_{ak}, \boldsymbol{I}_{m_i} + \boldsymbol{S}_{aki}) * N_{n_w}(0, \boldsymbol{H}_{ki}).$$

We can simultaneously estimate the fixed warping effects $\boldsymbol{w}_k$ and predict the random warping effects $\boldsymbol{w}_{ki}$ from the joint conditional negative log posterior. It is proportional to

$$l(\boldsymbol{w}_k, \boldsymbol{w}_{ki}) = \sum_{a=1}^{2}\sum_{i=1}^{N} M_{ki}||\boldsymbol{x}_{ai} - \boldsymbol{\Psi}_{ki}\hat{\boldsymbol{d}}_{ak}||^2_{\boldsymbol{I}_{m_i}+\boldsymbol{S}_{aki}} + 2\sum_{i=1}^{N} M_{ki}||\boldsymbol{w}_{ki}||^2_{\boldsymbol{H}_{ki}}, \quad k = 1, \ldots, K, \tag{5.7}$$

where $\boldsymbol{\Psi}_{ki}$ is determined by $m_i$ discrete values of the inverse of warping function $g_{ki}(t)$ which is totally characterized by $\boldsymbol{w}_k$ and $\boldsymbol{w}_{ki}$ as aforementioned. By minimizing $l(\boldsymbol{w}_k, \boldsymbol{w}_{ki})$

we can obtain the estimation of $\boldsymbol{w}_k$ and the prediction of $\boldsymbol{w}_{ki}$.

**(iii) Estimate the variance parameters $\sigma^2, \boldsymbol{\rho}_s$ and $\boldsymbol{\rho}_h$**

By using the first-order Taylor approximation of model (5.3) in the the random warping parameters $\boldsymbol{w}_{ki}$ around a given prediction $\boldsymbol{w}_{ki}^0$ ($\boldsymbol{w}_{ki}^0$ is specified by the estimate of $\boldsymbol{w}_{ki}$ from **(ii)** in the current iteration), we can write this model as a vectorized linear mixed-effects model

$$\boldsymbol{x}_a|_{z_{ki}=1} \approx \boldsymbol{G}_{ak} + \boldsymbol{B}_{ak}(\boldsymbol{W}_k - \boldsymbol{W}_k^0) + \boldsymbol{r}_{ak} + \boldsymbol{\epsilon}, \quad a = 1, 2; \quad k = 1, \ldots, K, \tag{5.8}$$

where $\boldsymbol{x}_a = \{\boldsymbol{x}_{ai}, i = 1, \ldots, N\}$, the effects are given by

$$\begin{aligned}
\boldsymbol{G}_{ak} &= \left\{\boldsymbol{\Psi}_{ki}|_{g_{ki}=g_{ki}^0}\boldsymbol{d}_{ak}\right\}_{ij} \in \mathbb{R}^m,\\
\boldsymbol{B}_{ak} &= \mathrm{diag}(\boldsymbol{B}_{aki})_i \in \mathbb{R}^{m\times Nn_w},\\
\boldsymbol{B}_{aki} &= \left\{\partial_{g_{ki}}\Big(\tau_{ak}\big(g_{ki}(t_j)\big)\Big)\Big|_{g_{ki}=g_{ki}^0}\Big(\nabla \boldsymbol{w}_{ki}\big(g_{ki}(t_j)\big)\Big)^{\intercal}\Big|_{\boldsymbol{w}_{ki}=\boldsymbol{w}_{ki}^0}\right\}_j \in \mathbb{R}^{m_i\times n_w},\\
\boldsymbol{W}_k &= (\boldsymbol{w}_{ki})_i \sim N_{Nn_w}(0, \sigma^2\boldsymbol{I}_N \otimes \boldsymbol{H}_{n_w\times n_w}), \quad \boldsymbol{W}_k^0 = (\boldsymbol{w}_{ki}^0)_i \in \mathbb{R}^{Nn_w},\\
\boldsymbol{r}_{ak} &\sim N_m(0, \sigma^2\boldsymbol{S}_{ak}), \quad \boldsymbol{S}_{ak} = \mathrm{diag}(\boldsymbol{S}_{aki})_i \in \mathbb{R}^{m\times m},\\
\boldsymbol{\epsilon} &\sim N_m(0, \sigma^2\boldsymbol{I}_m),
\end{aligned}$$

where $g_{ki}^0(t) = t + w_k(t) + w_{ki}^0(t)$. $\mathrm{diag}(\boldsymbol{B}_{aki})_{ki}$ is the block diagonal matrix with the $\boldsymbol{B}_{aki}$ matrices along its diagonal, and $\mathrm{diag}(\boldsymbol{S}_{aki})_i$ is the block diagonal matrix with the $\boldsymbol{S}_{aki}$ matrices along its diagonal. The derivation of the linearized model (5.8) is given by C.2 in Appendix C. The negative profile log likelihood function for the model (5.8) is proportional to

$$l(\sigma^2, \boldsymbol{\rho}_s, \boldsymbol{\rho}_h) = \sum_{k=1}^{K}\Big\{\sum_{a=1}^{2}\sigma^2||\boldsymbol{x}_a - \boldsymbol{G}_{ak} + \boldsymbol{B}_{ak}\boldsymbol{W}_k^0||^2_{\boldsymbol{V}_{ak}} + \sum_{a=1}^{2}\log\,\det\boldsymbol{V}_{ak}\Big\} + 2mK\log\sigma^2,$$

where $\boldsymbol{V}_{ak} = \dfrac{\Big(\boldsymbol{S}_{ak}+\boldsymbol{B}_{ak}\big(\boldsymbol{I}_N\otimes\boldsymbol{H}_{n_w\times n_w}\big)\boldsymbol{B}_{ak}^{\intercal}+\boldsymbol{I}_m\Big)}{\boldsymbol{M}_k} \in \mathbb{R}^{m\times m}$, with the definition similar to $\dfrac{\boldsymbol{I}_m+\boldsymbol{S}_{ak}}{\boldsymbol{M}_k}$ in (5.6).

To speed up convergence, we usually repeat the above three steps several times within each iteration.

**(iv) Update $M_{ki}$ and estimating $\boldsymbol{\beta}$**

Fix $\boldsymbol{\theta}^{(i_0+1)}$ and update

$$M_{ki} = \frac{\pi_{ki}^{(i_0)}p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki}^{(i_0+1)})}{\sum_{j=1}^{K}\pi_{ji}^{(i_0)}p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ji}^{(i_0+1)})},$$

where

$$\pi_{ki}^{(i_0)} = \frac{\exp\{\boldsymbol{v}_i^\intercal \boldsymbol{\beta}_k^{(i_0)}\}}{1+\sum_{j=1}^{K-1}\exp\{\boldsymbol{v}_i^\intercal \boldsymbol{\beta}_j^{(i_0)}\}}, \quad k=1,\ldots,K-1,$$

and $\pi_{Ki} = 1-\sum_{j=1}^{K-1}\pi_{ji}^{(i_0)}$. Then maximize $Q(\boldsymbol{\theta},\boldsymbol{\beta})$ with respect to $\boldsymbol{\beta}$, which is equivalent to maximize

$$l(\boldsymbol{\beta}) \triangleq \sum_{i=1}^{N}\left\{\sum_{k=1}^{K-1} M_{ki}\left\{\boldsymbol{v}_i^\intercal\boldsymbol{\beta}_k - \log\big[1+\sum_{j=1}^{K-1}\exp\{\boldsymbol{v}_i^\intercal\boldsymbol{\beta}_j\}\big]\right\} - M_{Ki}\log\big[1+\sum_{j=1}^{K-1}\exp\{\boldsymbol{v}_i^\intercal\boldsymbol{\beta}_j\}\big]\right\}.$$

This is very similar to the log-likelihood for a multinomial logit model ($M_{ki}$'s are corresponding to the observations) and can be maximized by iteratively re-weighted least square algorithm.

### 5.2.4 Model selection, clustering and related methods

There are two questions on the model selection for our proposed simultaneous registration and clustering method for multi-dimensional functional data: one is how to determine the number of knots for the splines and another is how many clusters. For the former, since our data is rather dense and insensitive, it works well using a relatively small number of equally-spaced knots. For the choice of the number of clusters, $K$, since the number of parameters, $p_l$, in model (5.1) is relative to the number of subjects, $N$, a second-order bias correction version of AIC called $\text{AIC}_c$ (Sugiura and Nariaki, 1978; Kenneth and David, 2004) is utilized:

$$\text{AIC}_c = -2l(\hat{\boldsymbol{\Theta}}) + 2p_l + \frac{2p_l(p_l+1)}{N-p_l-1},$$

where $l(\hat{\boldsymbol{\Theta}})$ is the maximized log-likelihood function, $\hat{\boldsymbol{\Theta}} = \{\hat{\boldsymbol{\theta}},\hat{\boldsymbol{\beta}}\}$ in this paper.

In inference, we first choose $K$ clusters by $\text{AIC}_c$. Then fit the data using the method discussed in the previous subsections and denote the estimates of the parameters by $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\beta}}$. Under the framework of *SRC* method, the fixed-effect part of the $i$th individual curve is calculated by

$$\hat{\boldsymbol{x}}_{ai}(t) = \sum_{k=1}^{K}\hat{\pi}_{ki}\big[\hat{\tau}_{ak}(\hat{g}_{ki}(t))\big], \quad a=1,2; \quad i=1,\ldots,N, \tag{5.9}$$

where $\hat{g}_{ki}(t) = t+\hat{\boldsymbol{w}}_k(t)+\hat{\boldsymbol{w}}_{ki}(t)$ and $\hat{\pi}_{ki} = \frac{\exp\{\boldsymbol{v}_i\hat{\boldsymbol{\beta}}_k\}}{1+\sum_{j=1}^{K-1}\exp\{\boldsymbol{v}_i\hat{\boldsymbol{\beta}}_j\}}$.

For any individual data $D^* = \{(\boldsymbol{x}_{1*},\boldsymbol{x}_{2*}),\boldsymbol{v}^*\}$ in $D$, the posterior distribution of the

cluster membership $\boldsymbol{z}^* = (z_1^*, \ldots, z_K^*)^\intercal$ is given by

$$p(z_k^* = 1|D^*) = \frac{\pi_k^* p(\boldsymbol{x}_{1*}|\hat{\boldsymbol{\theta}}_{1k}) p(\boldsymbol{x}_{2*}|\hat{\boldsymbol{\theta}}_{2k})}{\sum_{j=1}^{K_0} \pi_j^* p(\boldsymbol{x}_{1*}|\hat{\boldsymbol{\theta}}_{1j}) p(\boldsymbol{x}_{2*}|\hat{\boldsymbol{\theta}}_{2j})},$$

where

$$\pi_k^* = \frac{\exp\{\boldsymbol{v}^{*\intercal}\hat{\boldsymbol{\beta}}_k\}}{1 + \sum_{j=1}^{K-1} \exp\{\boldsymbol{v}^{*\intercal}\hat{\boldsymbol{\beta}}_j\}}.$$

As a result, the best cluster membership for $D^*$ can be determined by

$$k^* = \text{argmax}_{k=1,\ldots,K}\{p(z_k^* = 1|D^*)\}.$$

The average mean curve for each group can be calculated from $\{\hat{\boldsymbol{x}}_{ai}(t)|_{z_{ki}=1}\}$ for $k = 1, \ldots, K$.

**Related methods**

Functional $k$-means method is a popular approach for clustering curves (Chiou and Li, 2007), which is an extension of $k$-means cluster (MacQueen, 1967; Lloyd, 1982) for scalar variables. The idea can be extended to do clustering and registration simultaneously. Using the similar notation around (5.3), we can define the following objective function

$$F = \sum_{i=1}^{N} \sum_{k=1}^{K} z_{ki} d\big(\boldsymbol{x}_i, \boldsymbol{\tau}_k(g_{ki})\big), \tag{5.10}$$

where $d$ represents one kind of distance between each curve to its assigned mean curve $\boldsymbol{\tau}_k(g_{ki})$ and

$$z_{ki} = \begin{cases} 1, & \text{if } k = \text{argmin}_j \, d\big(\boldsymbol{x}_i, \boldsymbol{\tau}_j(g_{ji})\big), \\ 0, & \text{otherwise.} \end{cases}$$

In order to find the values $\{z_{ki}\}$ and the $\{\boldsymbol{\tau}_k(g_{ki})\}$ to minimize $F$, we can perform an iterative procedure in which each iteration involves two steps of the optimization with respect to $\{z_{ki}\}$ and $\{\boldsymbol{\tau}_k(g_{ki})\}$ respectively. This approach is denoted by *k-means-f* and its iterative procedures are as follows (use $d = ||\cdot||^2$ for the purpose of simplicity and convenience.):

1. Choose the initial values for the $z_{ki}$. We can use any clustering method for scalar variables $\{\boldsymbol{\beta}_i\}$ corresponding to the functional variables $\{\boldsymbol{x}_i\}$.

2. Fix $z_{ki}$ and minimize $F$ with respect to the $\boldsymbol{\tau}_{ak}(g_{ki})$. In this phase, minimizing $F$ is

equivalent to maximizing

$$\sum_{k=1}^{K}\sum_{i=1}^{N} z_{ki}\Big(\sum_{a=1}^{2}\big(\log p(\boldsymbol{x}_{ai}|\boldsymbol{\theta}_{aki})\big)\Big)$$

with the assumption that the covariance matrix of $\boldsymbol{x}_{ai}$ are the same over all the subjects. The detailed estimation of $\{\boldsymbol{\theta}_{aki}\}$ have been mentioned before and the $\boldsymbol{\tau}_{ak}(g_{ki})$ can be obtained straightforwardly.

3. Fix $\boldsymbol{\tau}_{ak}(g_{ki})$ and minimize $F$ with respect to $z_{ki}$. Since the term $F$ in (5.10) involving different $i$ are independent, we can optimize $F$ for each $i$ separately by choosing $z_{ki}$ as follows
$$z_{ki} = \begin{cases} 1, & \text{if } k = \text{argmin}_j \,||\big(\boldsymbol{x}_i - \boldsymbol{\tau}_j(g_{ji})\big)||^2; \\ 0, & \text{otherwise.} \end{cases}$$

4. Repeat Step 2 and Step 3 until convergence.

A special case of the *SRC* model defined in Section 5.2.1 is that the allocation model in (5.2) doesn't depend on any scalar variables (denoted by *SRC-f*, i.e. use the function variable only). This special case is very similar to the above *k-means-f* approach. Actually the *k-means-f* algorithm is a special case of EM algorithm for *SRC-f*. Using similar notation around (5.3) and assuming $\boldsymbol{x}_{ai}|_{z_{ki}=1} \sim N_{m_i}(\boldsymbol{\tau}_{ak}(g_{ki}), \delta\boldsymbol{I}), a = 1, 2; i = 1, \ldots, N$, where $\delta$ is shared by all the clusters, we have the density function of $\boldsymbol{x}_{ai}$ with the form

$$p(\boldsymbol{x}_{ai}|\boldsymbol{\theta}_{aki}) = (2\pi\delta)^{-\frac{m_i}{2}} \exp\big\{-\frac{1}{2\delta}||\boldsymbol{x}_{ai} - \boldsymbol{\tau}_{ak}(g_{ki})||^2\big\}.$$

Let $p(z_{ki} = 1) = \pi_k, k = 1, \ldots, K$ with $\sum_{k=1}^{K}\pi_k = 1$ be the allocation model. Using the EM algorithm for the Gaussian mixtures described in Section 5.2.2, we have

$$M_{ki} = \frac{\pi_k \prod_{a=1}^{2} \exp\big\{-||\boldsymbol{x}_{ai} - \boldsymbol{\tau}_{ak}(g_{ki})||^2/2\delta\big\}}{\sum_{j=1}^{K}\pi_j \prod_{a=1}^{2} \exp\big\{-||\boldsymbol{x}_{ai} - \boldsymbol{\tau}_{aj}(g_{ji})||^2/2\delta\big\}}.$$

Clearly, $M_{ki} \to z_{ki}$, when $\delta \to 0$. Thus $E_{\boldsymbol{z}}[\log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta}, \boldsymbol{\beta})] \approx -\frac{1}{2\delta}F + \text{costant}$, when $\delta$ is small. It means the optimization problem is the same as the *k-means-f* algorithm given by (5.10) (using $d = ||\cdot||^2$).

## 5.3 Numerical analyses

We shall evaluate the performance and properties of the proposed *SRC* model in this section. We will compare it with functional $k$-means clustering (*k-means-f*) with simultaneous registration as discussed in Section 5.2.4, the *SRC* without using an allocation

model (*SRC-f*) and scalar $k$-means clustering (*k-means-s*). The *k-means-s* is a general k-means clustering method using scalar variables only. We will conduct analysis on both simulated and real data.

### 5.3.1 Simulation study

In this simulation study, we consider 2D curves coming from two groups. For each group, the corresponding observations of functional variables $\boldsymbol{x}(t)$ and scalar variables $v$ will be generated. There are $N_k$ batches of data in each group, where $k = 1, 2$. We will evaluate and compare four methods based on the simulated data $D = \{(\boldsymbol{x}_i(t_{ij}), \boldsymbol{v}_i); i = 1, \ldots, N; j = 1, \ldots, m_i\}$ in different scenarios where $N = N_1 + N_2$.

**Data generation**

1. Generate the underlying true curves, i.e. the curves based on the internal time scale $g^{-1}(t)$. We first assume those curves in two groups share the following two slightly different true means, similar to two patterns of the real curves of hyoid bone's movement

$$\begin{aligned} \boldsymbol{\mu}_1(t) &= \big(\mu_{11}(t), \mu_{21}(t)\big) = \big(\exp\{\cos(2\pi t)\}, \exp\{\sin(2\pi t)\}\big), \\ \boldsymbol{\mu}_2(t) &= \big(\mu_{12}(t), \mu_{22}(t)\big) = \big(\exp\{\cos(2\pi t^{1.05} - b_1)\}, \exp\{\sin(2\pi t^{1.1} + b_1)\}\big). \end{aligned} \tag{5.11}$$

   The degree of overlapping between two groups relies on the value of $b_1$. The smaller the value of $b_1$, the higher the degree of overlapping, and more difficult to cluster those curves. We use the equidistant points $t_j = \frac{j+1}{102}, j = 1, \ldots, 100$ as the input grid, i.e. $m_i = 100$. The underlying true curves are generated as

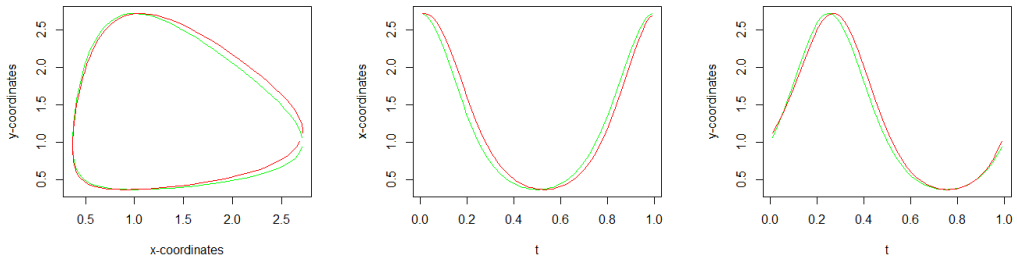
$$\boldsymbol{x}_{ak}(g^{-1}(t)) = \mu_{ak}(t), \quad a = 1, 2; \;\; k = 1, 2.$$

   Figure 5.1 shows the shape of the true mean curves for different values of $b_1$.

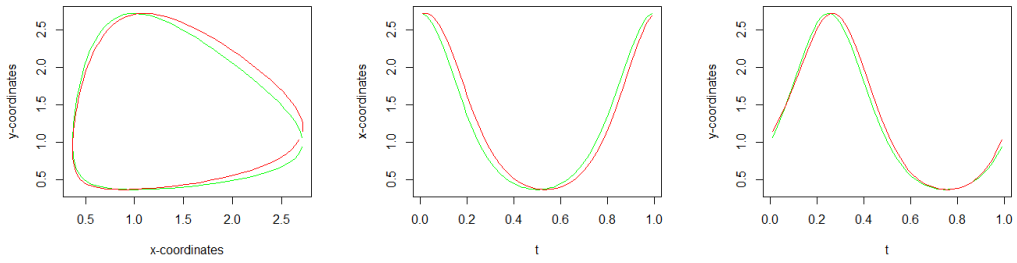
2. Generate the original 2D curves $\boldsymbol{x}(t)$ by adding the warping function, amplitude variation and errors as follows.

   (a) Model the true curves $\boldsymbol{x}_{ak}(g^{-1}(t))$ using B-spline basis function with 8 knots and obtain the coefficients $\boldsymbol{d}_{ak}$.

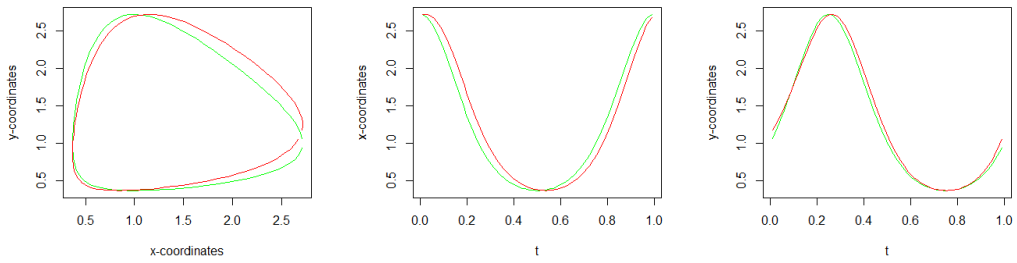
   (b) For simplicity, we set $g_{ki}(t) = t + w_{ki}(t)$ and use hyman spline (monotone cubic spine using Hyman filtering) based on the anchor knots $t_w = (0, 0.33, 0.67, 1)$ $(n_w = 4)$. Set $\boldsymbol{w}_{ki} \sim N_4(0, \boldsymbol{T}_k^\intercal \boldsymbol{\Gamma}_i)$, where $\boldsymbol{T}_k^\intercal \boldsymbol{T}_k = \boldsymbol{O}_k$, $\boldsymbol{O}_1 = \begin{bmatrix} 10 & 4 \\ 4 & 8 \end{bmatrix}$ and




(a) $b_1 = 0.08$




(b) $b_1 = 0.10$




(c) $b_1 = 0.12$

Figure 5.1: True mean curves $\boldsymbol{\mu}_1(t)$ (lines in green) and $\boldsymbol{\mu}_2(t)$ (lines in red) of group 1 and group 2 with $b_1 = 0.08, 0.10, 0.12$.

$\boldsymbol{O}_2 = \begin{bmatrix} 10 & 8 \\ 8 & 15 \end{bmatrix}$, and $\boldsymbol{\Gamma}_i = (\boldsymbol{\Gamma}_{i1}, \boldsymbol{\Gamma}_{i2})^\intercal$ with $\boldsymbol{\Gamma}_{i1}, \boldsymbol{\Gamma}_{i2}$ being independent random variables $N(0, \sigma_w^2)$, where $i = 1, \ldots, N_k$.

(c) Set the amplitude variation $\boldsymbol{r}_{aki} = \boldsymbol{T}_0^\intercal \cdot \boldsymbol{\Gamma}_{i0}$, where $\boldsymbol{T}_0^\intercal \boldsymbol{T}_0 = \boldsymbol{O}_0$, $a = 1, 2$, $k = 1, 2$, $i = 1, \ldots, N_k$. The matrix $\boldsymbol{O}_0$ is created by the Matern covariance function with $\boldsymbol{\rho}_r = (100, 0.3, 3)$, where the three elements represent the scale, range and smoothness, respectively (Raket, 2016), and $\boldsymbol{\Gamma}_{j0}$ is a vector of 100 independent normal random variables $N(0, \sigma_r^2)$. Set $\boldsymbol{\epsilon} \sim N(0, \sigma^2 \mathbb{I})$.

(d) Generate $\boldsymbol{x}(t)$ based on the model (5.3).

3. Generate $\boldsymbol{v}$'s. We generate the scalar variables $v$ by sampling from uniform distribution as follows:

$$V_i \sim \begin{cases} \mathrm{U}(1, 2), & i = 1, 2, \ldots, N_1, \\ \mathrm{U}(1 - b_2, 2 - b_2), & i = N_1 + 1, \ldots, N_1 + N_2. \end{cases}$$

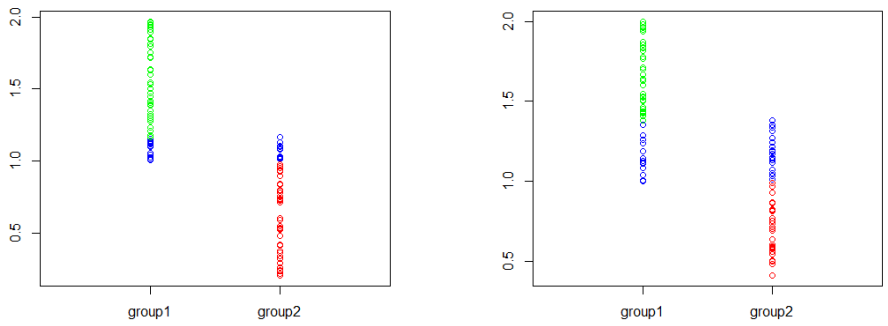
Note that the larger the value of $d_2$, the lower the degree of overlapping and easier to carry out clustering using scalar variables.

**Results**

In order to investigate how the overlapping of the observations of both scalar variables and functional variables affect the performance of clustering, we study four scenarios with the constraints $4\sigma_w^2 = \sigma_r^2 = \sigma^2 = 0.01^2$ and $N_1 = N_2 = 30$. There are 100 replications for each scenario. We use two criteria to measure the performance of clustering. These are Rand index (RI) (Rand, 1971a) and adjusted Rand index (ARI) (Hubert and Arabie, 1985a), mentioned in Section 3.4.4 of Chapter 3 for assessing the performance of each method.

Four methods are applied to the simulated data $D$ in Scenario 1 with $b_1 = 0.12$, $b_2 = 0.8$, Scenario 2 with $b_1 = 0.10$, $b_2 = 0.8$, Scenario 3 with $b_1 = 0.08$, $b_2 = 0.8$ and Scenario 4 with $b_1 = 0.08$, $b_2 = 0.6$. Figure 5.2 and Figure 5.3 show the raw data depending on different $b_2$ and $b_1$ respectively. First of all, we apply the $\mathrm{AIC}_c$ to choose the number of clusters. The results from Figure 5.4 show that $\mathrm{AIC}_c$ score reaches its minimum at 2 clusters. Table 5.1 summarizes the comparisons by average ARI and RI. Overall, both measures suggest that the proposed *SRC* outperform the other three methods in all scenarios because of the use of both functional and scalar data.

From Table 5.1, we note that all the four methods perform best in Scenario 1 compared with the other scenarios. In this scenario, both $b_1$ and $b_2$ take the largest values, indicating that the overlapping of the functional data (Figure 5.3 (c)) and scalar data (Figure 5.2 (a)) are the smallest and both greatly contribute to distinguishing those two clusters. The




(a) $b_2 = 0.8$ (b) $b_2 = 0.6$

Figure 5.2: Observations of scalar variable in two cases. The 'blue' ones stand for those in the range of overlapping.

other three methods, *SRC-f*, *k-means-f* and *k-means-s* based on either functional data or scalar data, also have good performance but not as good as *SRC*.

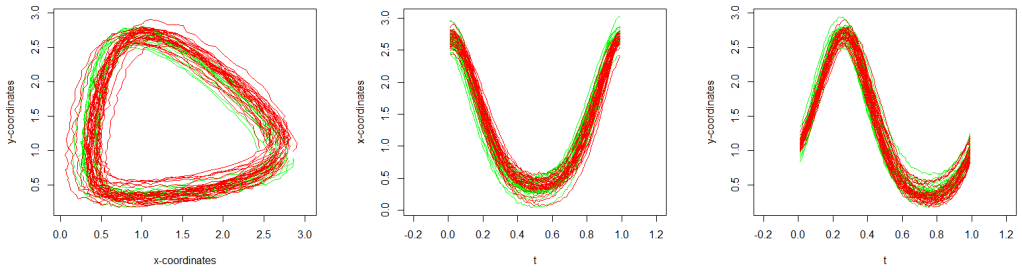
The first three scenarios share the same value of $b_2$, indicating that the degree of overlapping in two clusters for scalar data does not change (Figure 5.2 (a)). The performance of *k-means-s* remains the same. The overlapping in two clusters for functional data, however, gets smaller and smaller as the value of $b_1$ increases from Scenario 1 to Scenario 3. It leads to a sharp decline for the performance of *SRC-f* and *k-means–f*, both of which depend on functional data only, as opposed to a mild decrease of the performance of *SRC*, which is based on both scalar data and functional data.

The scenario 4 has the smallest $b_1$ (Figure 5.3 (a)) and $b_2$ (Figure 5.2 (b)) and it is quite difficult to carry out clustering just based on functional data or scalar data only. Consequently, the values of ARI for *SRC-f*, *k-means-f* and *k-means-s* are very small. But *SRC* still performs well and are much better than the others.

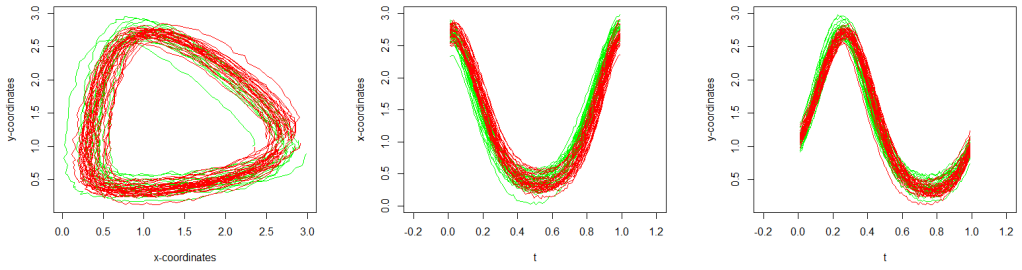
Other combinations with varying overlapping determined by $b_1$ and $b_2$ and with different sample sizes have also been examined. The results presented here are very typical.

| | Scenario 1 | | Scenario 2 | | Scenario 3 | | Scenario 4 | |
|---|---|---|---|---|---|---|---|---|
| | RI | ARI | RI | ARI | RI | ARI | RI | ARI |
| *SRC* | **0.98** | **0.96** | **0.95** | **0.90** | **0.91** | **0.82** | **0.80** | **0.61** |
| *SRC-f* | 0.90 | 0.80 | 0.75 | 0.49 | 0.62 | 0.25 | 0.62 | 0.25 |
| *k-means-f* | 0.91 | 0.83 | 0.76 | 0.53 | 0.64 | 0.29 | 0.64 | 0.29 |
| *k-means-s* | 0.81 | 0.62 | 0.81 | 0.62 | 0.81 | 0.62 | 0.69 | 0.37 |

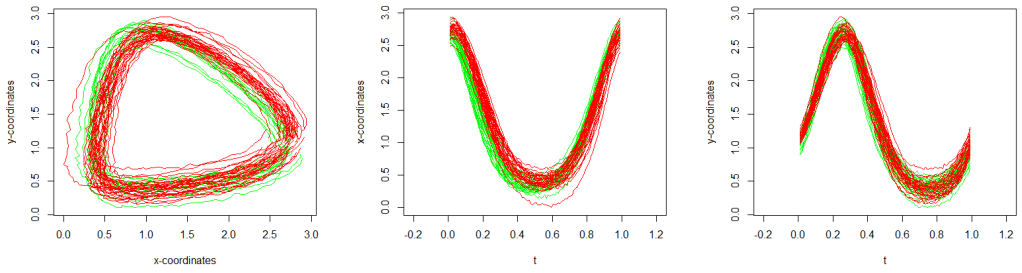
Table 5.1: Comparison of average clustering results among four methods.




(a) $b_1 = 0.08$

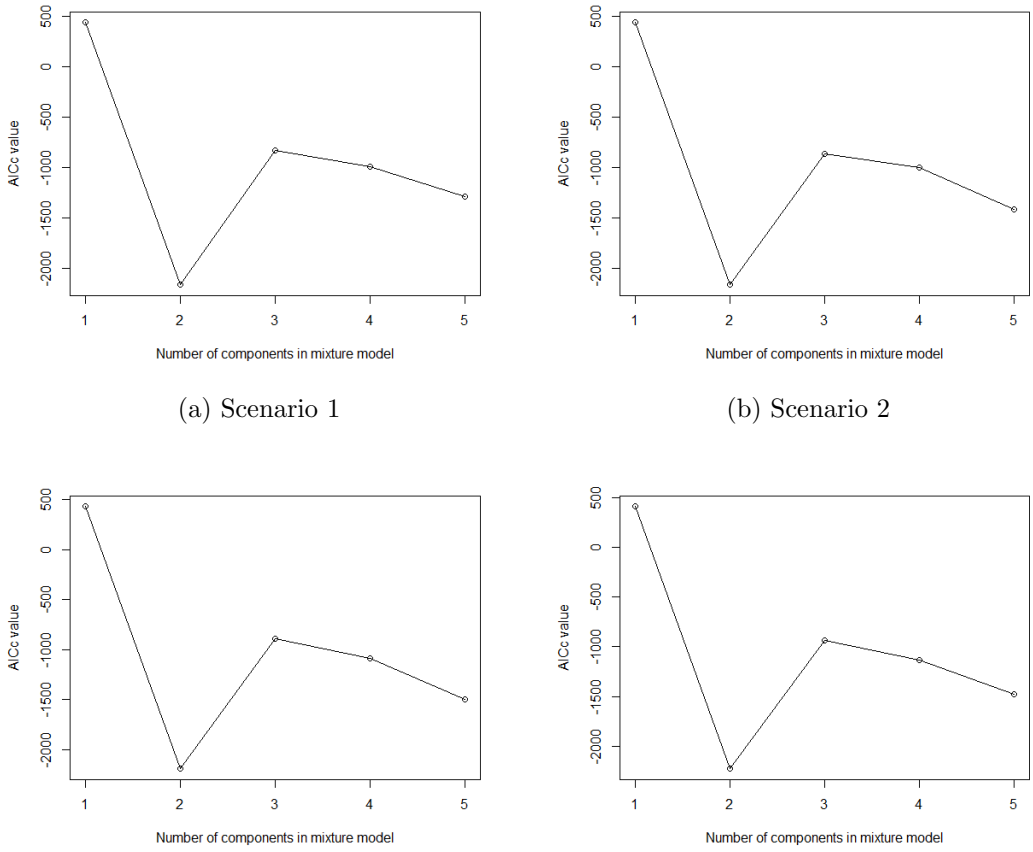



(b) $b_1 = 0.10$




(c) $b_1 = 0.12$

Figure 5.3: The raw 2D curves in one simulation run in three cases.




(a) Scenario 1

(b) Scenario 2

(c) Scenario 3

(d) Scenario 4

Figure 5.4: The value of AICc calculated from one replication in each scenario for the method *SRC*.

**Recovery of curves and cluster patterns**

To understand the underlying process better, it is necessary to use the optimal alignment to estimate the entire curve, so we estimate the aligned individual curves and reconstruct the cluster pattern using equation (5.9).

Figure 5.5 displays one simulation run of $N = 100$, with $N_1 = N_2 = 50$ curves in each group. The top panel presents the original raw curves in the two clusters in two colors in two dimensions ($x$-axis and $y$-axis) for a new scenario with $4\sigma_w^2 = \sigma_r^2 = \sigma^2 = 0.02^2, b_1 = 0.15, b_2 = 0.8$. The other panels respectively show the individual aligned curves resulting from *SRC*, *SRC-f* and *k-means-f*, with the value of RI (1, 0.63, 0.79) and the value of ARI (1, 0.26, 0.54) respectively. The *SRC* properly differentiates the two clusters (red and green) after curve alignment and performs better in recovering the cluster patterns.

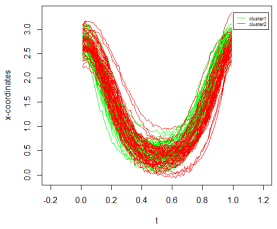
Figure 5.6 summarizes the result of clustering patterns. It shows the SRC method recovered the true pattens very well. As a measure of estimation error, we use the root average squared error (Gervini and Gasser, 2004), see the details in Section 3.4.2 in Chapter 3. The values of *rase* are 2.9, 4.6 and 5.4 corresponding to three models *SRC*, *SRC-f* and *k-means-f*.

**A simulation example in an extreme scenario**

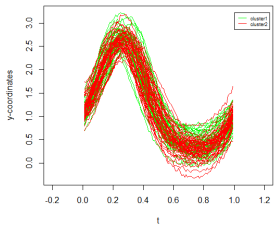
It is not uncommon that sometimes the functional variables provide little information so that it fails to implement the clustering just based on those curves. However, the addition of scalar variables can make the clustering possible. We simulate a run of $N = 100$ (sample size), with $4\sigma_w^2 = \sigma_r^2 = \sigma^2 = 0.02^2, b_1 = 0.05, b_2 = 0.8$, and $N_1 = N_2 = 50$ curves in each group. Figure 5.7 displays the individual aligned curves resulting from three methods, from which no discernible clusters are visible. The RI and ARI for *SRC*, *SRC-f* and *k-means-f* are, however, markedly different with the values of $(0.82, 0.50, 0.50)$ and $(0.64, 0, 0)$ respectively. Figure 5.8 summarizes the mean functions of two clusters by three methods. Their values of *rase* are 1.1, 18.3 and 4.3 respectively. Those results show that the use of SRC leads to meaningful findings but the other twos are equivalent to random guess. This extreme scenario provides further evidence of the good performance of the proposed *SRC*.

### 5.3.2 Real data analysis

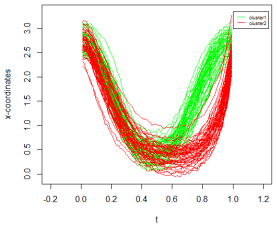
The application to a real data is to cluster the normal people and the patients with stroke by studying their hyoid bone motion as well as the other scalar variables. Two groups, one for normal people and the other for patients, are included. Figure 4.1(a) in Chapter 4 shows one frame from a X-ray video clip. The raw data before being preprocessed are shown in Figure 4.1(b). Most of the assumptions are the same as the example of real



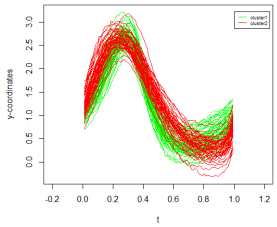

(a) Original $\boldsymbol{x}_1(t)$



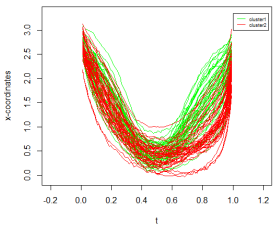

(b) Original $\boldsymbol{x}_2(t)$



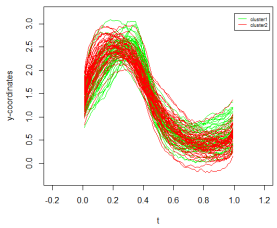

(c) *SRC*, aligned $\boldsymbol{x}_1(t)$



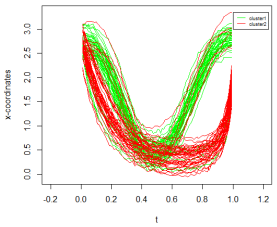

(d) *SRC*, aligned $\boldsymbol{x}_2(t)$



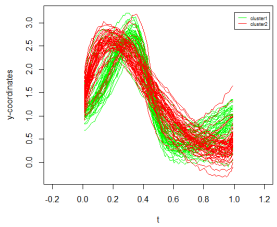

(e) *SRC-f*, aligned $\boldsymbol{x}_1(t)$




(f) *SRC-f*, aligned $\boldsymbol{x}_2(t)$

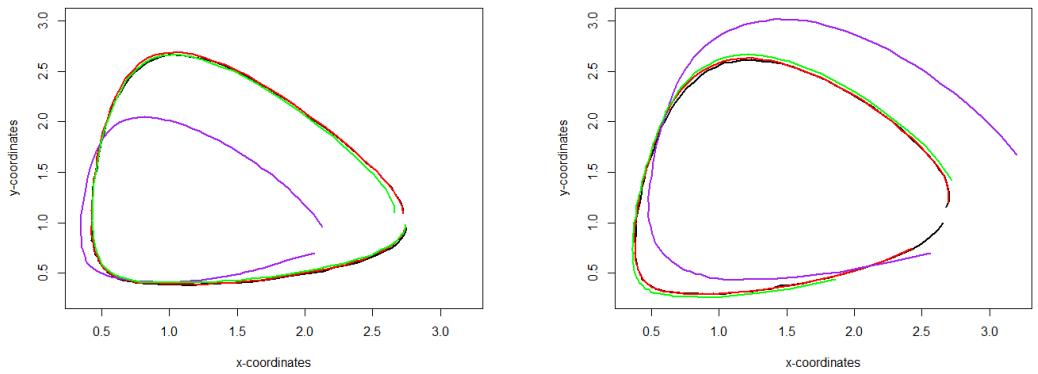



(g) *k-means-f*, aligned $\boldsymbol{x}_1(t)$




(h) *k-means-f*, aligned $\boldsymbol{x}_2(t)$

Figure 5.5: (a) and (b) are simulated 2D curves of two groups (green and red). (c)-(h) are aligned individual cruves by *SRC*, *SRC-f* and *k-means-f*.

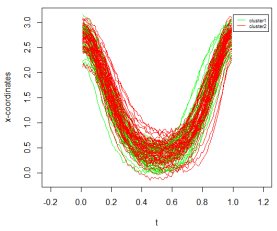



(a) Cluster 1 (b) Cluster 2

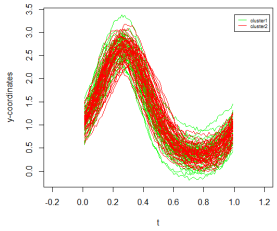
Figure 5.6: Mean functions for 2D curves in each cluster. *Black lines* are true mean curves. *Red lines*, *purple lines* and *green lines* stand for mean curves calculated from the results from *SRC*, *SRC-f* and *k-means-f* respectively.

data analysis in Chapter 4 except for the choice of scalar variable. The scalar variable we choose in this example is the size of Pyriform Sinus Residue (see its position in Figure 5.9). Regarding to those 2D curves, we firstly carry out the preprocessing procedures like multi-dimensional shift, scaling and rotation using the package of *GPA*. We then use B-spline basis functions for modeling the mean curves. The covariance function for the amplitude variance is assumed to be a Matern covariance function. We assume the warping function be a smooth nonlinear deformation produced by an increasing spline and the random vector $\boldsymbol{w}_{ki}$ be a Brownian bridge observed at discrete anchor points.

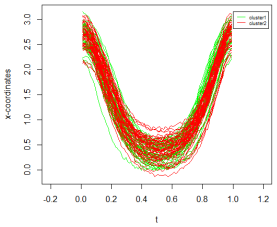
We examine the performance of four methods *SRC*, *SRC-f*, *k-means-f* and *k-means-s* aforementioned. The values of AICc are shown in Figure 5.10. It shows that the two-component mixture model has the smallest value. Table 5.2 shows the values of RI and ARI by comparing clustering results by the four methods with the clinic outcomes. We can see that the *SRC* method outperforms the other three. As a matter of fact, both *SRC-f* and *k-means-f* with value of RI equivalent to 0.5 fail in this real data example. It is similar to the extreme example in Figure 5.7 and Figure 5.8. More numerical results are provided in Figure 5.11 and Figure 5.12.



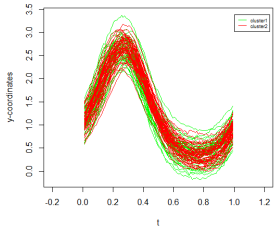

(a) Original $\boldsymbol{x}_1(t)$



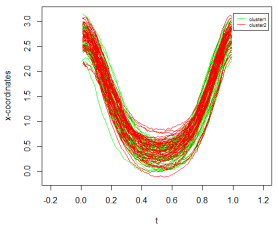

(b) Original $\boldsymbol{x}_2(t)$



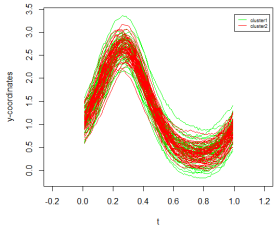

(c) *SRC*, aligned $\boldsymbol{x}_1(t)$



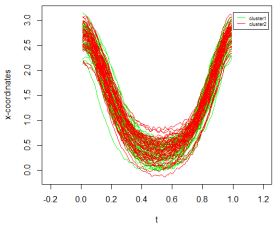

(d) *SRC*, aligned $\boldsymbol{x}_2(t)$



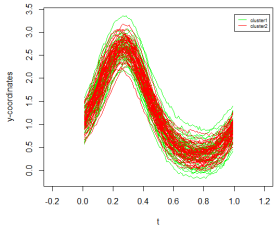

(e) *SRC-f*, aligned $\boldsymbol{x}_1(t)$




(f) *SRC-f*, aligned $\boldsymbol{x}_2(t)$

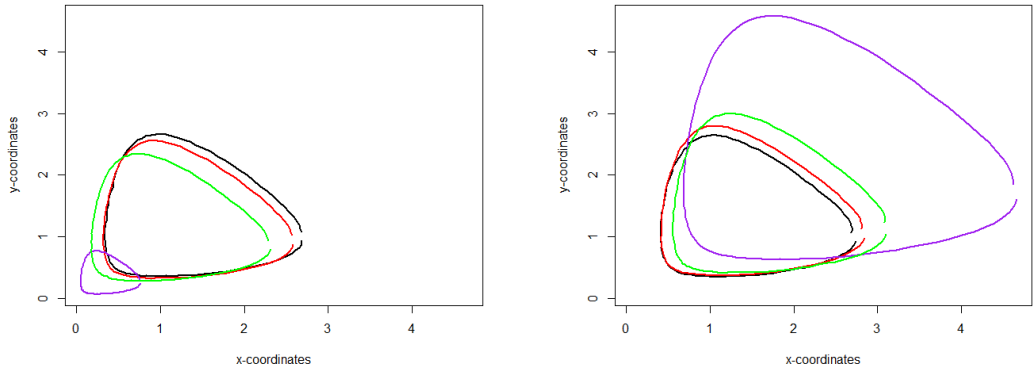



(g) *k-means-f*, aligned $\boldsymbol{x}_1(t)$




(h) *k-means-f*, aligned $\boldsymbol{x}_2(t)$

Figure 5.7: (a) and (b) are simulated 2D curves of two groups (green and red). (c)-(h) are aligned individual curves by *SRC*, *SRC-f* and *k-means-f*.




(a) Cluster 1 (b) Cluster 2

Figure 5.8: Mean functions for 2D curves from two clusters. The *black lines* are true mean functions. The *red lines*, *purple lines* and *green lines* are respectively corresponding to results obtained from the model *SRC*, *SRC-f* and *k-means-f*.

| Model | RI | ARI |
|---|---|---|
| *SRC* | **0.71** | **0.42** |
| *SRC-f* | 0.50 | 0.02 |
| *k-means-f* | 0.50 | 0.02 |
| *k-means-s* | 0.67 | 0.33 |

Table 5.2: Results of clustering by four methods for the real data

## 5.4 Chapter Summary

We have proposed a methodology for simultaneous registration and clustering, *SRC*, for multi-dimensional functional data which considers both the curves and scalar variables. This model captures the heterogeneity from the potential time warping for curves and scalar variables corresponding to each subject while carrying out the clustering in the meantime. It can be implemented with EM algorithm. Numerical examples show that it outperforms three other related methods, *SRC-f*, *k-means-f* and *k-means-s*. The results in Section 5.3.1 show that in most cases the inclusion of scalar variables can improve the performance of clustering in functional data analysis. The main contributions include:

(a) simultaneously carrying out registration and modeling for multi-dimensional functional data allowing variation among subjects,

(b) the use of both functional and scalar covariates while conducting clustering.

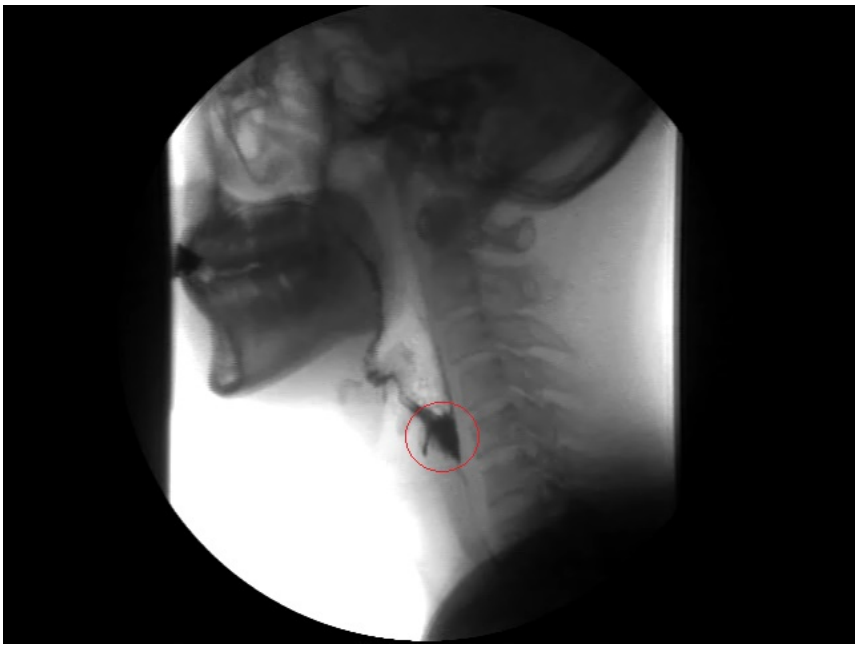


Figure 5.9: Highlight of Pyriform Sinus Residue, covered by the red circle

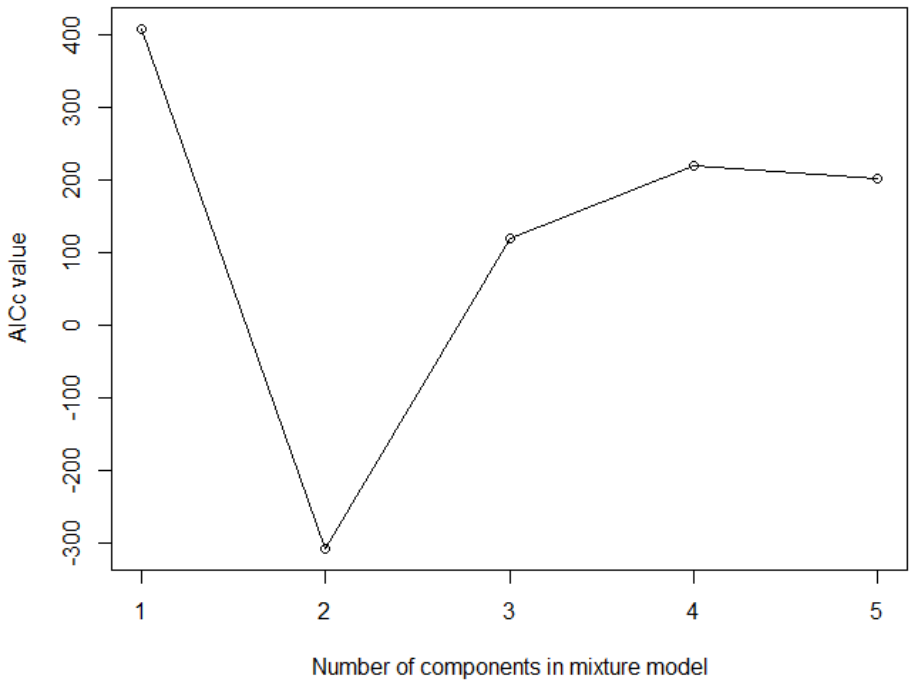
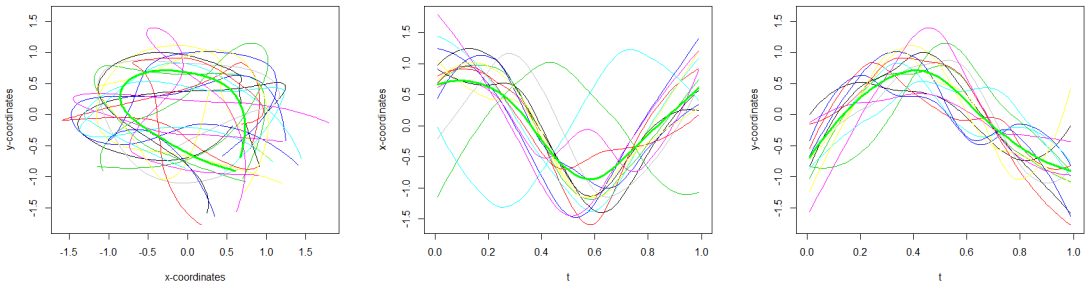
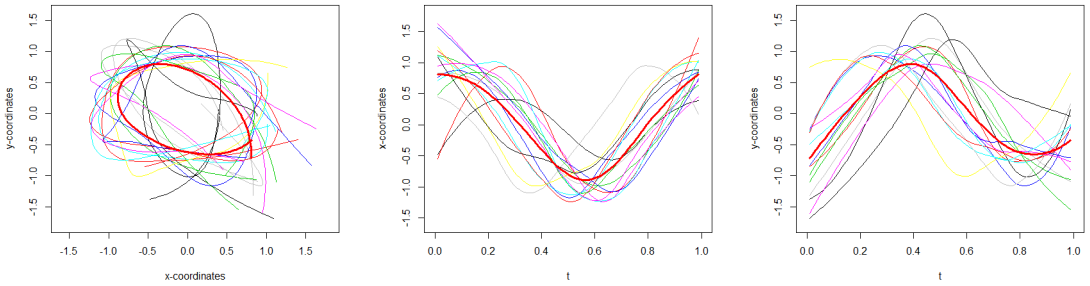



Figure 5.10: The values of AICc for $SRC$

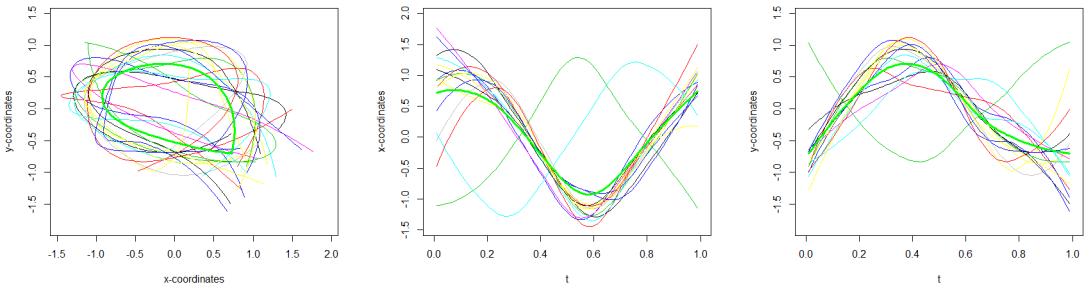



(a) Curves from 15 normal people

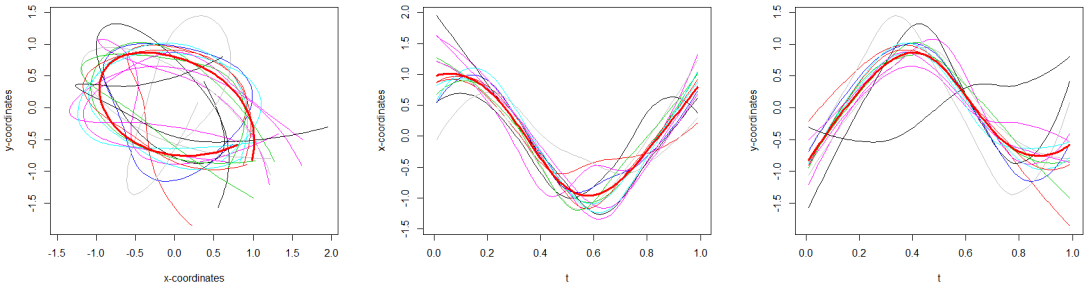



(b) Curves from 15 abnormal people

Figure 5.11: Curves of hyoid bone motion for two true groups, where the bold curves in green (upper panel) and in red (lower panel) are the average mean curve for each group




(a) Cluster 1 with 16 people




(b) Cluster 2 with 14 people

Figure 5.12: Curves of hyoid bone motion for two groups clustered by *SRC*, where the bold curves in green (upper panel) and in red (lower panel) are the average mean curve for each group

# Chapter 6

# Conclusion and Future Work

In this thesis, we conduct the acquisition, registration, classification and clustering for multi-dimensional functional data. Chapter 2 discusses how to acquire the movement data from the video clips. We develop an all-in-one platform to do semi-automatic tracking, data preprocessing like smoothing and segmentation, of hyoid bone motion from videofluoroscopic swallowing study. Once the observations of 2D functional data are obtained, we propose one new methodology (*GPSM*) for registration in Chapter 3. It combines the advantages of both Generalized Procrustes analysis (Gower, 1975a) and self-modelling registration (Gervini and Gasser, 2004). Good performance of registration is demonstrated in both simulation study and real data analysis. However, the classical classification after the registration seems not satisfactory. Thus, in Chapter 4, we propose the method of joint curve registration and classification (*JCRC*) with mixed scalar and functional data. Two-stage functional models are developed. The functional logistic regression model is utilized in the first stage, where the estimation of registered curves are obtained from the second stage. The latter aims to do the registration and modelling for the curves by a nonlinear mixed-effect model. Furthermore, we extend the problem from classification to clustering in Chapter 5. We propose the simultaneous registration and clustering (*SRC*) models via two-level models. They include mixtures of Gaussian process functional regression and logistic allocation model, allowing simultaneous registration and modeling for curves and the use of both scalar and function variables. Both *JCRC* in Chapter 4 and *SRC* in Chapter 5 consider two types of data, leading to much better results for classification and clustering on simulated data and real data.

To estimate the $\boldsymbol{x}(\hat{g}^{-1}(t))$ and the coefficient function $\boldsymbol{\beta}(t)$ in Section 4.2.2 of Chapter 4, we can alternatively use the same functional basis, like B-spline basis, to expand both $\boldsymbol{x}(\hat{g}^{-1}(t))$ and $\boldsymbol{\beta}(t)$ and follow with the truncation. We use the fast fitting method. It works well while $\boldsymbol{x}(\hat{g}^{-1}(t))$ is poorly observed and able to estimate arbitrary smooth coefficient functions (Goldsmith et al., 2011). It is also of interest to study the convergence of the

iterative algorithm developed in Section 4.2.5. Practically, we get the final prediction while the value of $y^*$ does not change and in general, five iterations are enough for those simulated examples. However, in theory, it still needs more work to prove the convergence of this algorithm in future.

Model selection for the *SRC* in Chapter 5 is an interesting but difficult issue for a mixture model, especially for the models with complex forms. Kenneth and David (2004) suggested AICc should be used unless $\frac{N}{p_l} > 40$ for the model with the large value of $p_l$. In our model, the number of parameters $p_l$ is quite close to the number of subject $N$. Thus, we use AICc. It works well for the examples discussed in that chapter. It is worth further study under a general functional data analysis framework.

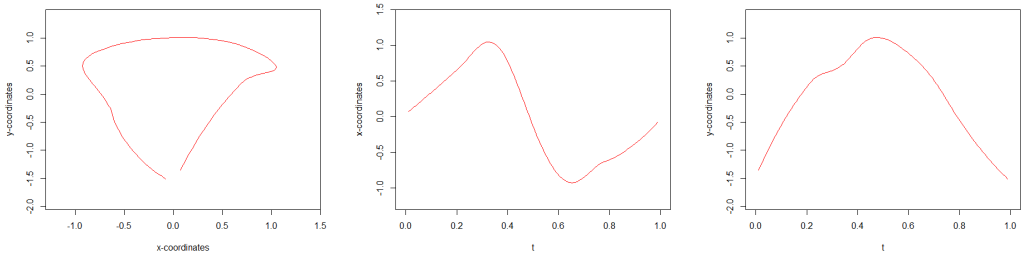
Generally, the registration for multi-dimensional functional data is much more complicated than one dimensional case. In both Chapter 4 and Chapter 5, we use the preprocessing package GPA (Gower, 1975b) and a further registration via a simple warping function. The latter is one of the key parts in our models. This approach performs very well in the numerical examples presented in both chapters. Further research is required to improve the iterative implementation for the complete registration, similar to the shape geodesic algorithm by the metric-based method proposed by Srivastava et al. (2011a). The inverse of warping function $g$ in model (5.1) can also be replaced with various types of other functions depending on types of data. For instance, we can define the warping function as simple as a horizontal shift, i.e. $g_{ki}(t) = t + b_{ki}$ or a linear stretch of the curves, i.e. $g_{ki}(t) = (1 + b_{ki})t + c_{ki}$, where $b_{ki}$ and $c_{ki}$ are both one dimensional unknown parameter. Those linear warping functions have been examined by others (Liu and Yang, 2009; Sangalli et al., 2010). The success of resolving registration problem often depends on the flexibility of choosing warping function.

The results we obtained for the real data are encouraging although it is still in early stage. Research for this topic is ongoing. More features extracted from video clips along with other variables, both functional and scalar, are under investigation. Different types of models for data fitting, clustering/classification and prediction are being developed.

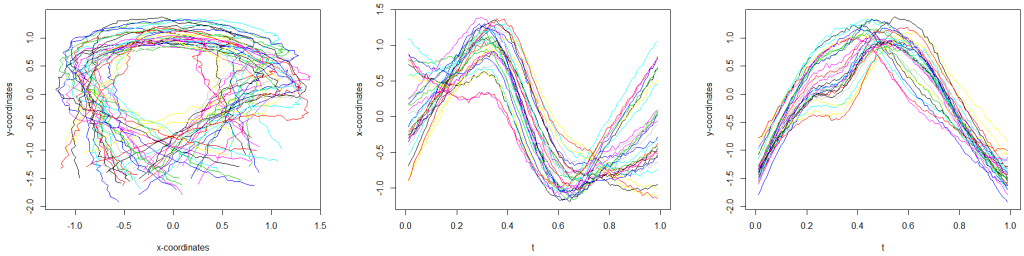
# Appendix A

# Extra Numerical Results of Registration for Multi-dimensional Functional Data

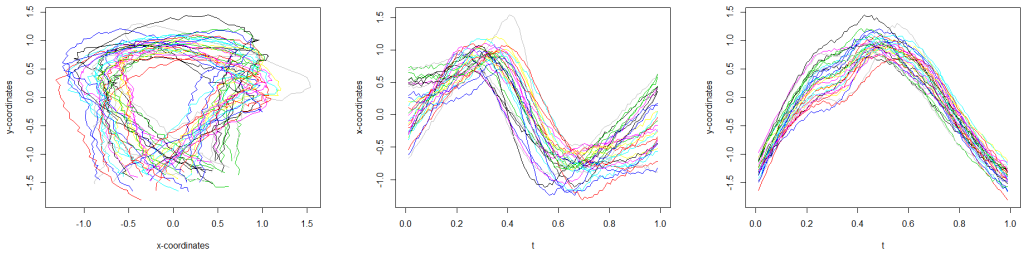
Figures A.1 to A.8 provides the extra numerical examples of Dataset 1 and Dataset 2 in Section 3.4.3.



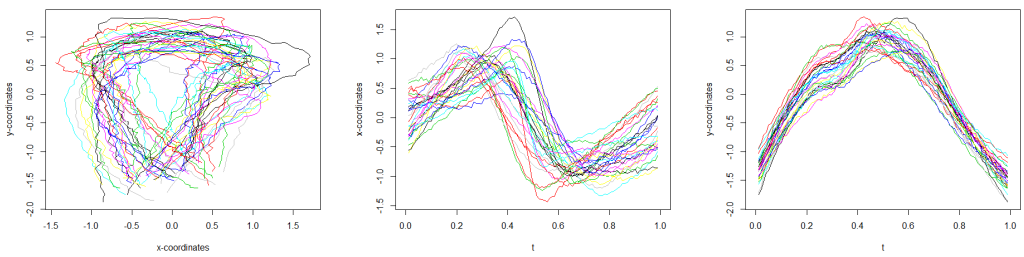

(a) 2D reference curve of Dataset 1




(b) Scenario A: $\sigma_w = 0.1$ and $\theta_0 = \pi/4$

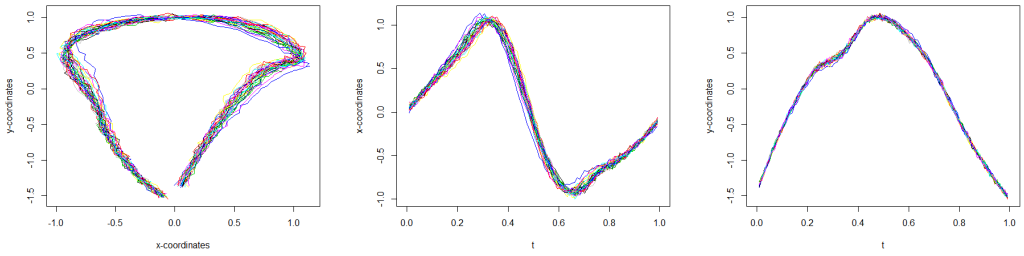



(c) Scenario B: $\sigma_w = 0.5$ and $\theta_0 = \pi/6$

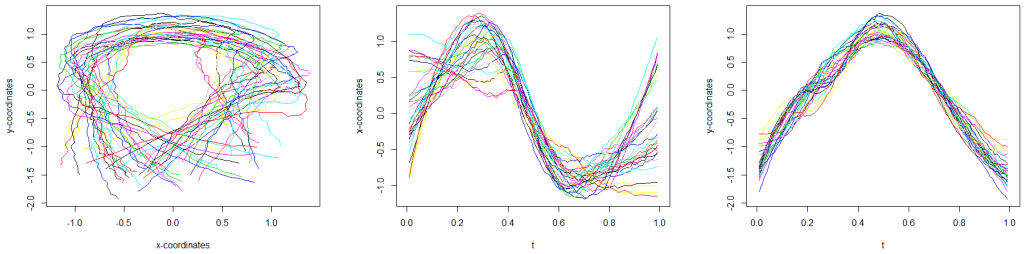



(d) Scenario C: $\sigma_w = 1$ and $\theta_0 = \pi/8$

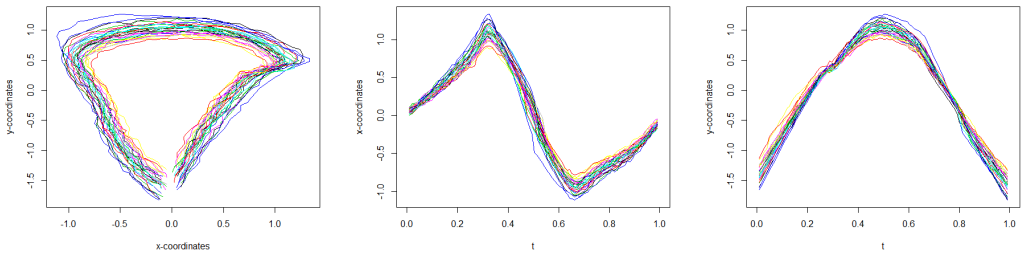
Figure A.1: Three examples of data in Dataset 1 corresponding to three scenarios.



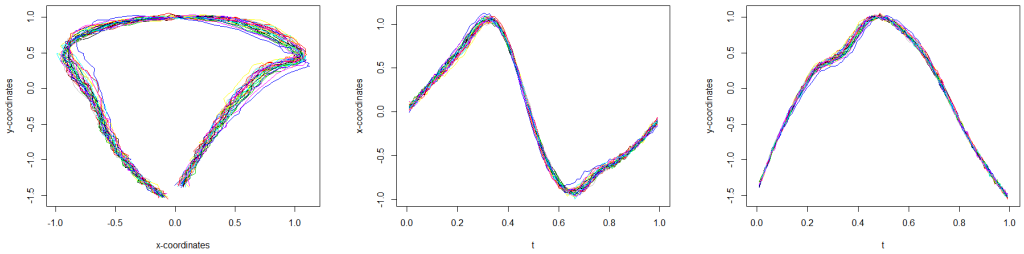

(a) Registration by *GPA*




(b) Registration by *SM*

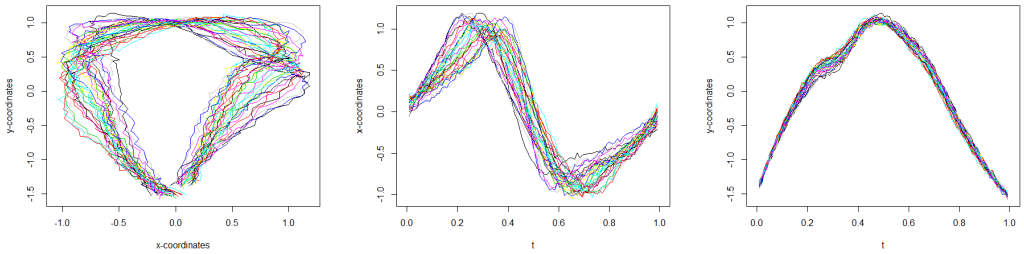



(c) Registration by *SRV*

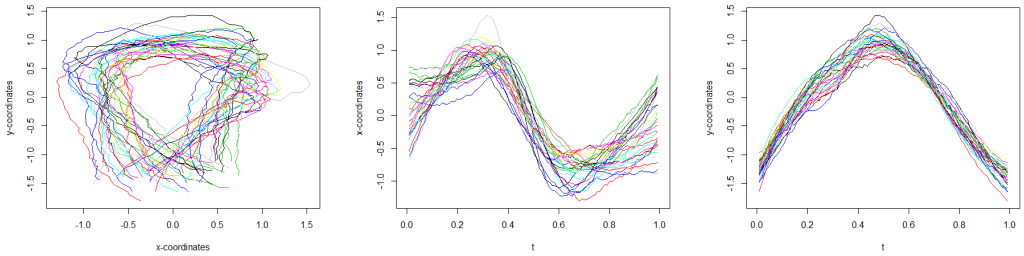



(d) Registration by *GPSM*

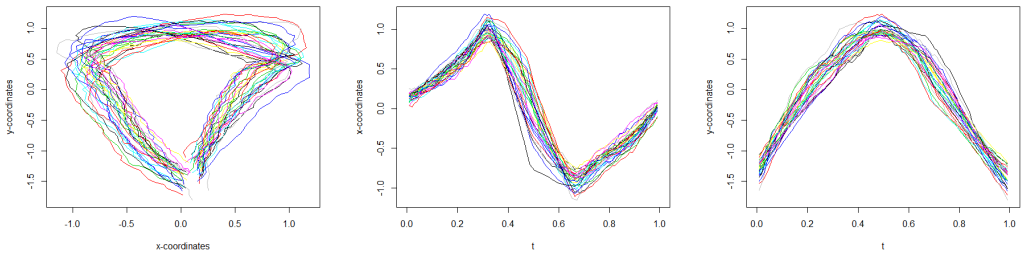
Figure A.2: An example of registration results in Dataset 1 by four methods for Scenario A with $\sigma_w = 0.1$ and $\theta_0 = \pi/8$.



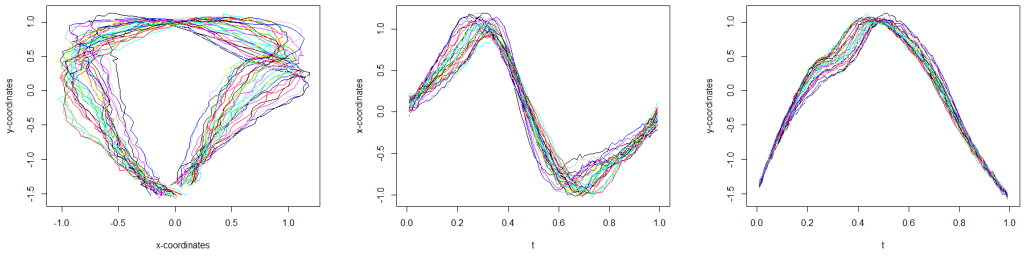

(a) Registration by *GPA*




(b) Registration by *SM*

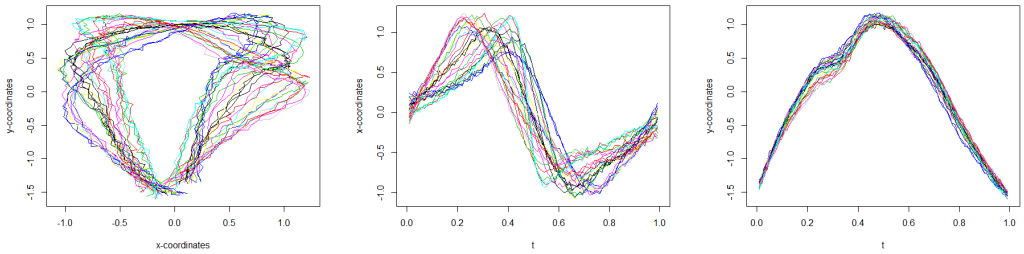



(c) Registration by *SRV*

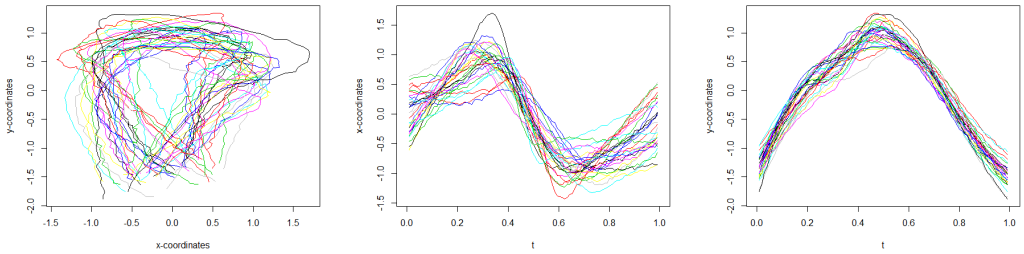



(d) Registration by *GPSM*

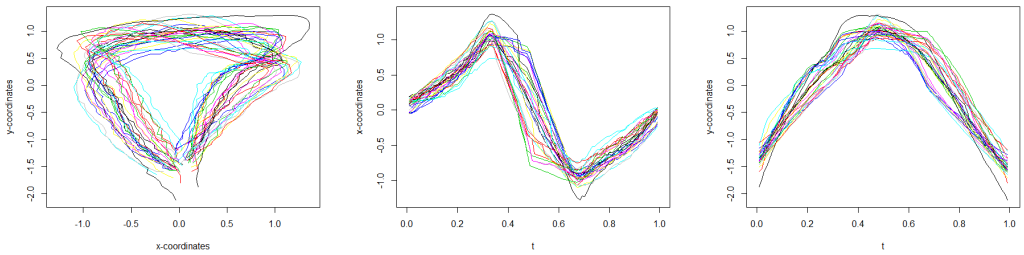
Figure A.3: An example of registration results in Dataset 1 by four methods for Scenario B with $\sigma_w = 0.5$ and $\theta_0 = \pi/6$.



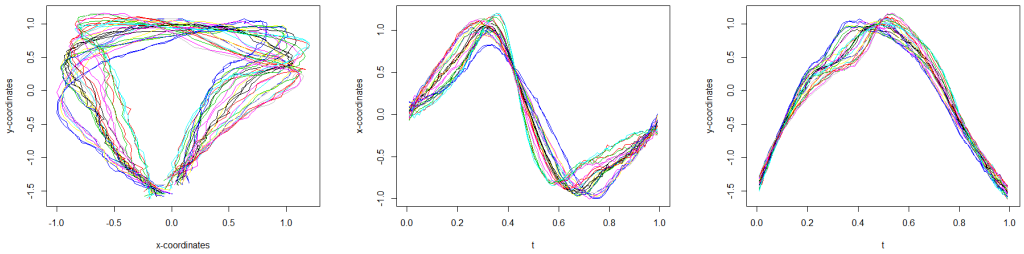

(a) Registration by *GPA*




(b) Registration by *SM*

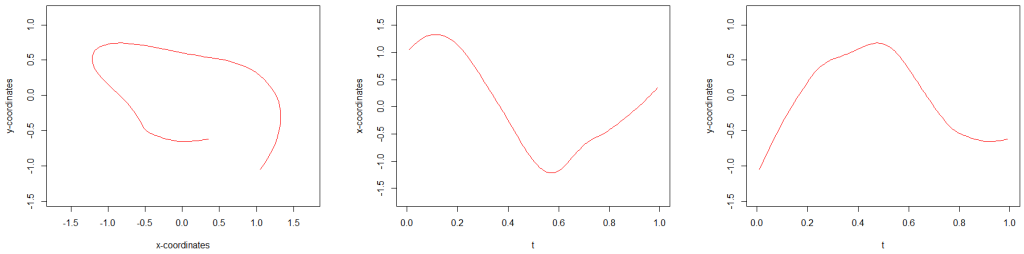



(c) Registration by *SRV*

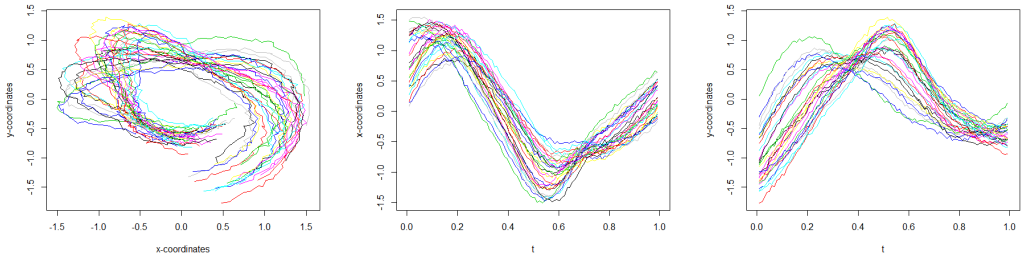



(d) Registration by *GPSM*

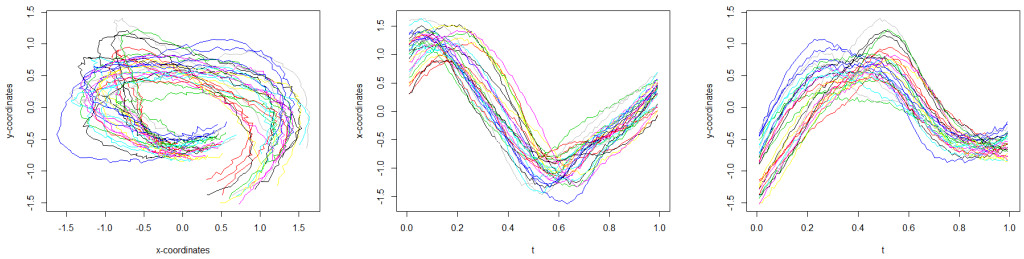
Figure A.4: An example of registration results in Dataset 1 by four methods for Scenario C with $\sigma_w = 1$ and $\theta_0 = \pi/8$.



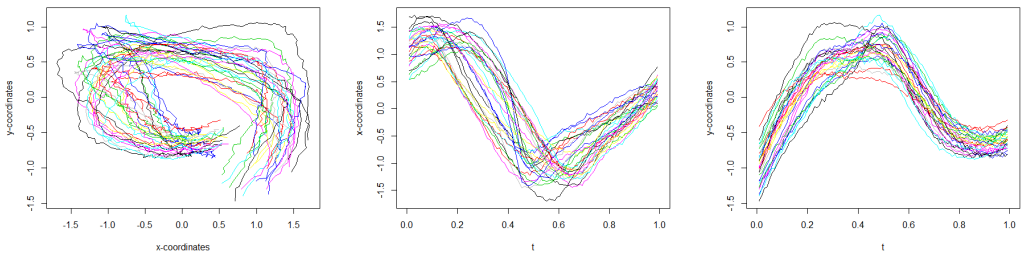

(a) 2D reference curve of Dataset 2




(b) Scenario A: $\sigma_w = 0.1$ and $\theta_0 = \pi/4$

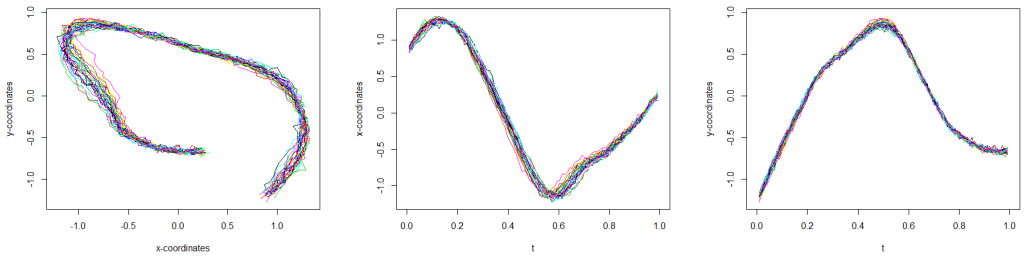



(c) Scenario B: $\sigma_w = 0.5$ and $\theta_0 = \pi/6$

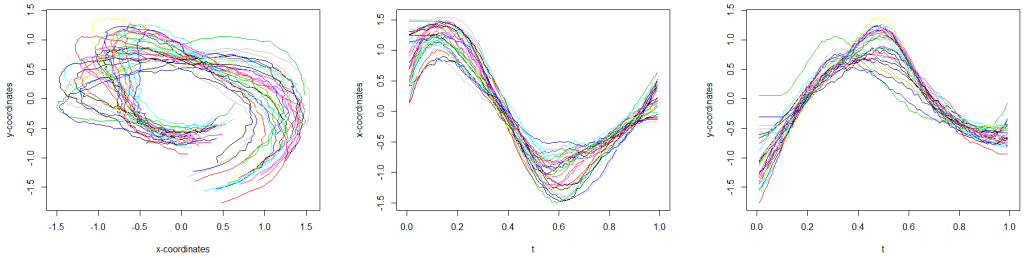



(d) Scenario C: $\sigma_w = 1$ and $\theta_0 = \pi/8$

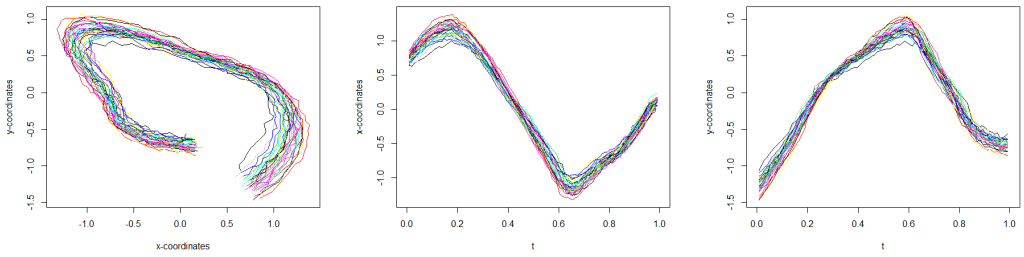
Figure A.5: Three examples of data in Dataset 2 corresponding to three scenarios.



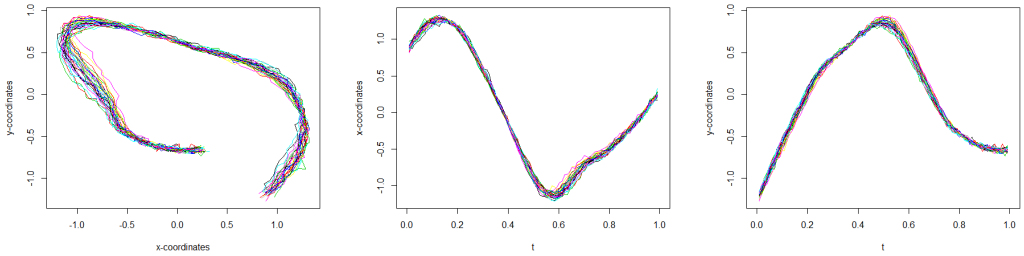

(a) Registration by *GPA*




(b) Registration by *SM*

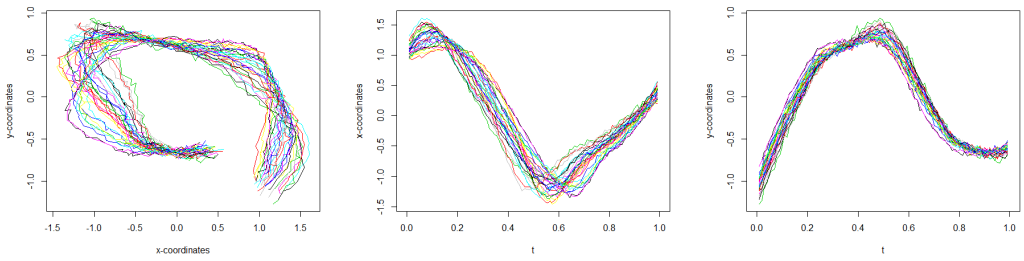



(c) Registration by *SRV*

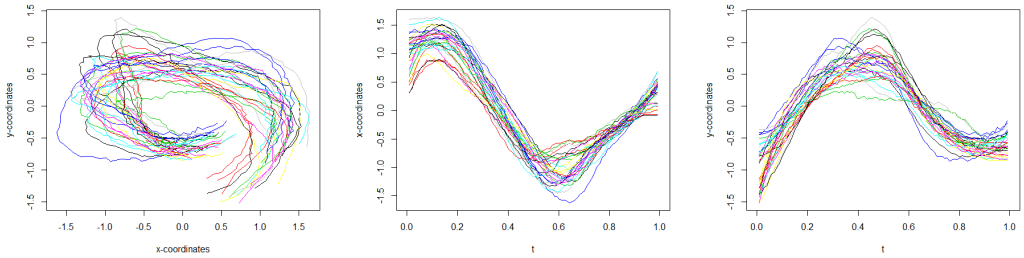



(d) Registration by *GPSM*

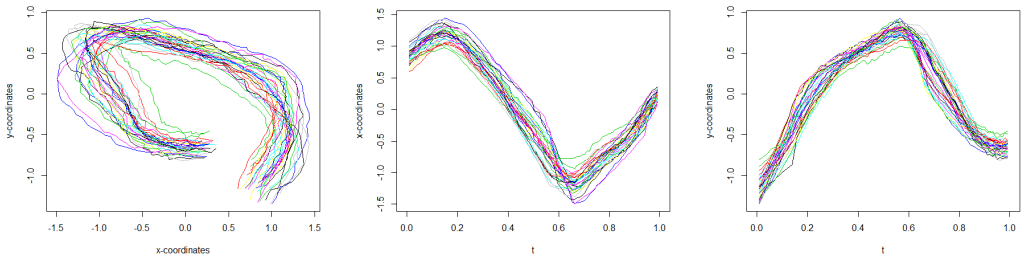
Figure A.6: An example of registration results in Dataset 2 by four methods for Scenario A with $\sigma_w = 0.1$ and $\theta_0 = \pi/8$.



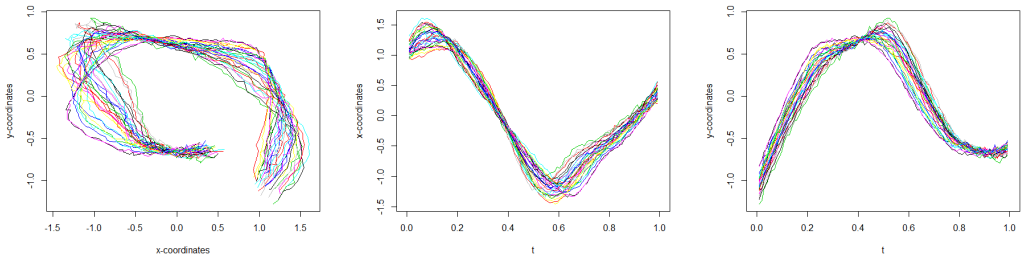

(a) Registration by *GPA*




(b) Registration by *SM*

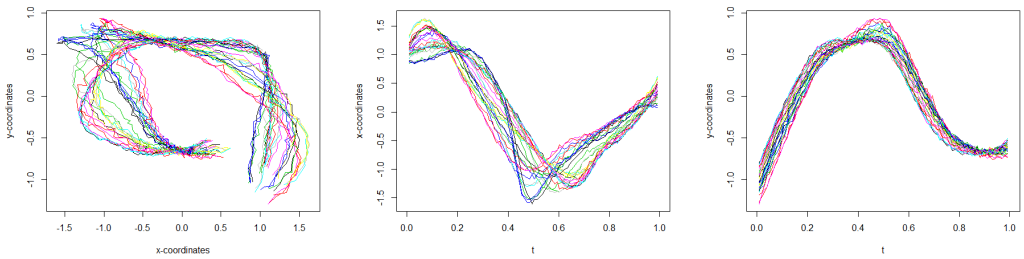



(c) Registration by *SRV*

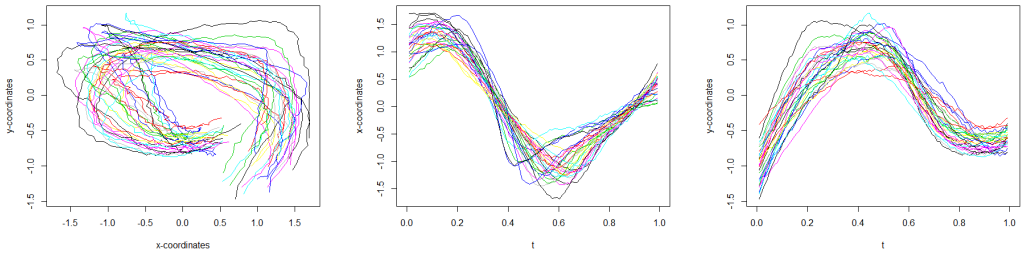



(d) Registration by *GPSM*

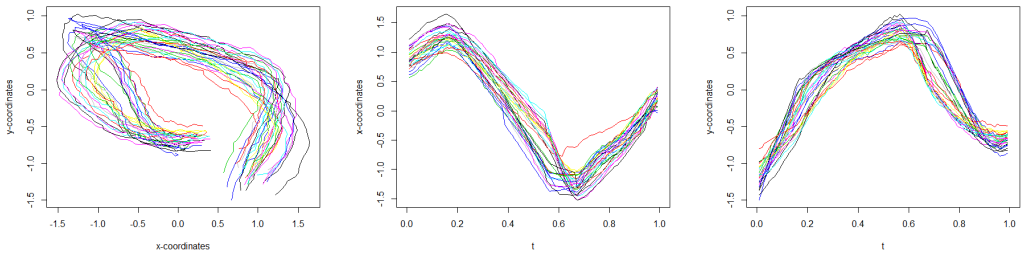
Figure A.7: An example of registration results in Dataset 2 by four methods for Scenario B with $\sigma_w = 0.5$ and $\theta_0 = \pi/6$.



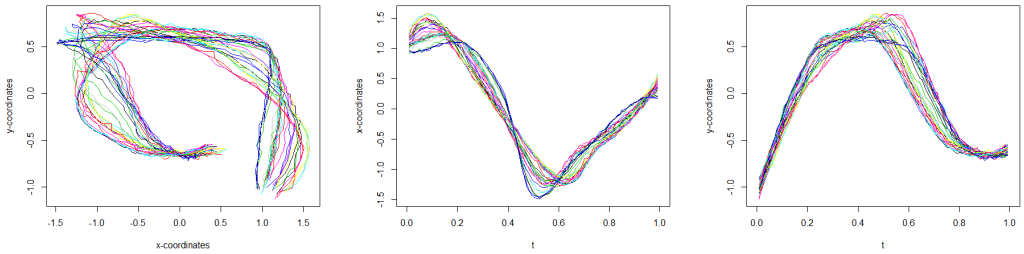

(a) Registration by *GPA*




(b) Registration by *SM*




(c) Registration by *SRV*




(d) Registration by *GPSM*

Figure A.8: An example of registration results in Dataset 2 by four methods for Scenario C with $\sigma_w = 1$ and $\theta_0 = \pi/8$.

# Appendix B

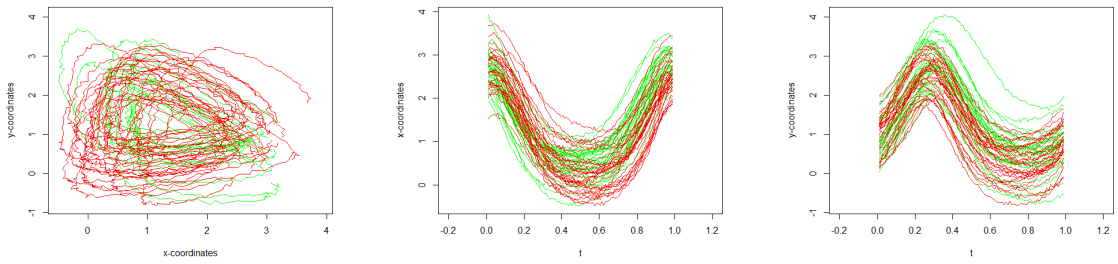
# Extra Numerical Results by *JCRC* method

## B.1 More examples of raw data

Figures B.1 to B.4 provide extra numerical results for the simulated examples discussed in Chapter 4.

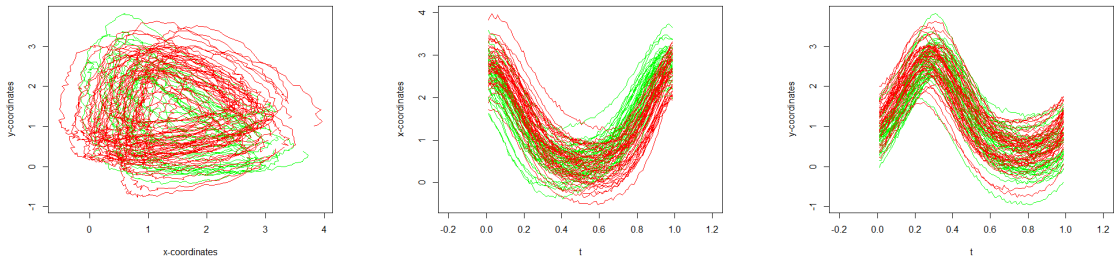
## B.2 More examples of aligned curves

Figures B.5 to B.7 show the results after registration, corresponding to the raw data from Figures B.1 to B.3.



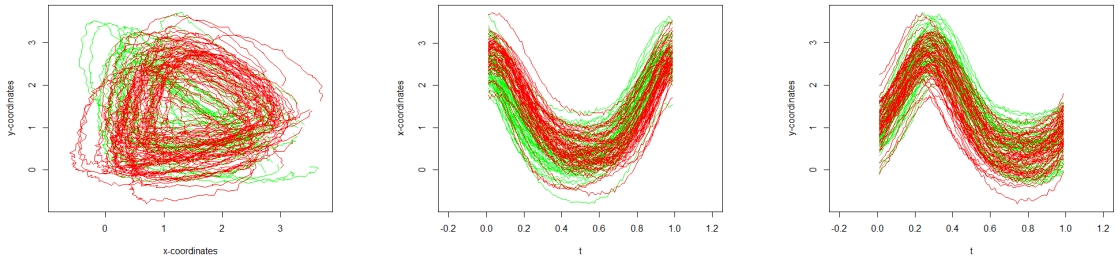

(a) 2D curves (b) $\boldsymbol{x}_1(t)$ (c) $\boldsymbol{x}_2(t)$

Figure B.1: An example of 2D raw curves from the scenario: $4\sigma_w = \sigma_r = \sigma = 0.02$ and $N = 60$ . Curves in green indicate the first group ($y = 0$), while these in red represent the second group ($y = 1$).



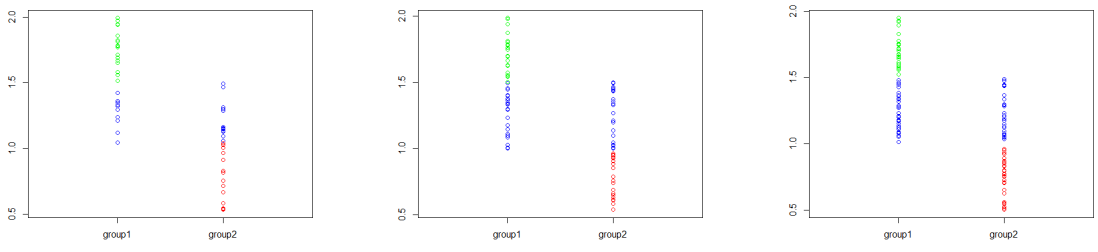

(a) 2D curves (b) $\boldsymbol{x}_1(t)$ (c) $\boldsymbol{x}_2(t)$

Figure B.2: An example of 2D raw curves from the scenario: $4\sigma_w = \sigma_r = \sigma = 0.02$ and $N = 90$.



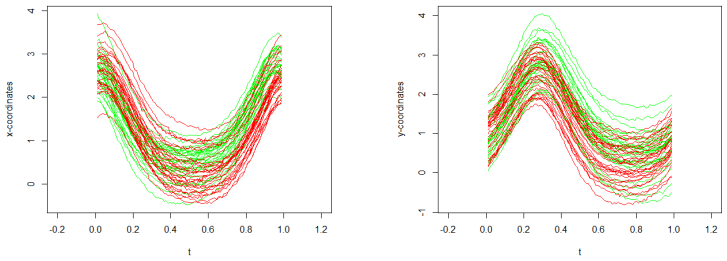

(a) 2D curves (b) $\boldsymbol{x}_1(t)$ (c) $\boldsymbol{x}_2(t)$

Figure B.3: An example of 2D raw curves from the scenario: $4\sigma_w = \sigma_r = \sigma = 0.02$ and $N = 120$.



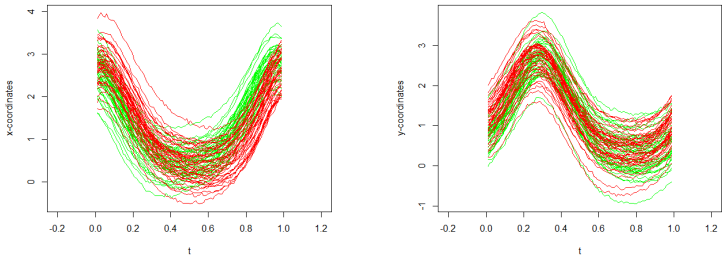

(a) (b) (c)

Figure B.4: Three examples of observations of scalar variable with $N = 60, 90, 120$.



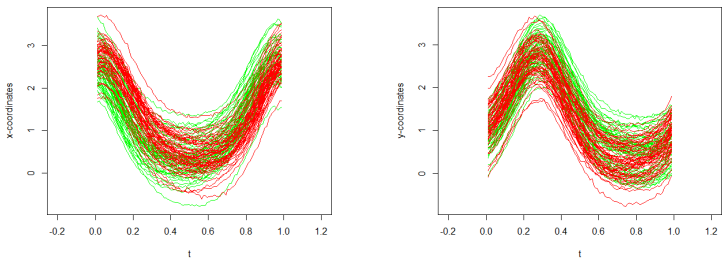

(a) $\boldsymbol{x}_1(t)$ (b) $\boldsymbol{x}_2(t)$

Figure B.5: The curves after registration by *JCRC*, corresponding to the raw curves in Figure B.1.




(a) $\boldsymbol{x}_1(t)$ (b) $\boldsymbol{x}_2(t)$

Figure B.6: The curves after registration by *JCRC*, corresponding to the raw curves in Figure B.2.

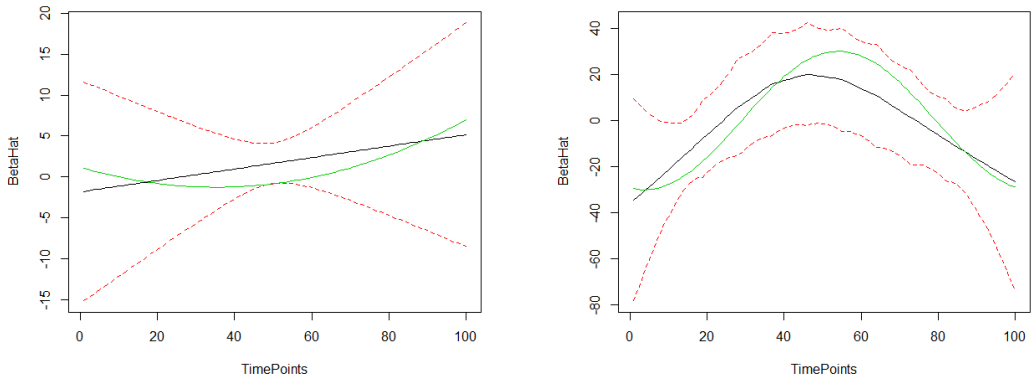



(a) $\boldsymbol{x}_1(t)$ (b) $\boldsymbol{x}_2(t)$

Figure B.7: The curves after registration by *JCRC*, corresponding to the raw curves in Figure B.3.

## B.3 More examples of inference and prediction

Figures B.8 to B.13 demonstrate the confidence intervals of $\hat{\boldsymbol{\beta}}(t)$ and the distribution of $\hat{\pi}$ under three different scenarios, corresponding to Table 4.1.




(a) 95% confidence interval for $\hat{\beta}_1(t)$ (b) 95% confidence interval for $\hat{\beta}_2(t)$

Figure B.8: An example of confidence intervals for $\hat{\boldsymbol{\beta}}(t)$ from the scenario: $N = 60, K_x = 18, K_e = 10$. The lines in green are the true $\boldsymbol{\beta}$, the lines in black stand for the estimators $\hat{\boldsymbol{\beta}}$, and the dotted red lines represent the boundaries of 95% confidence intervals. 'TimePoints' are equal to $100t_j$.

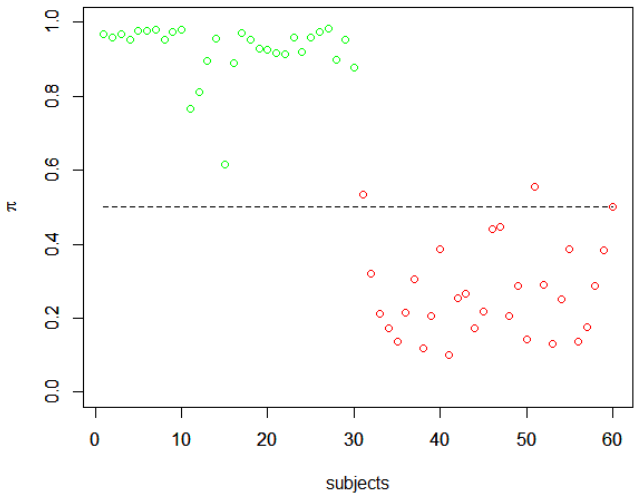
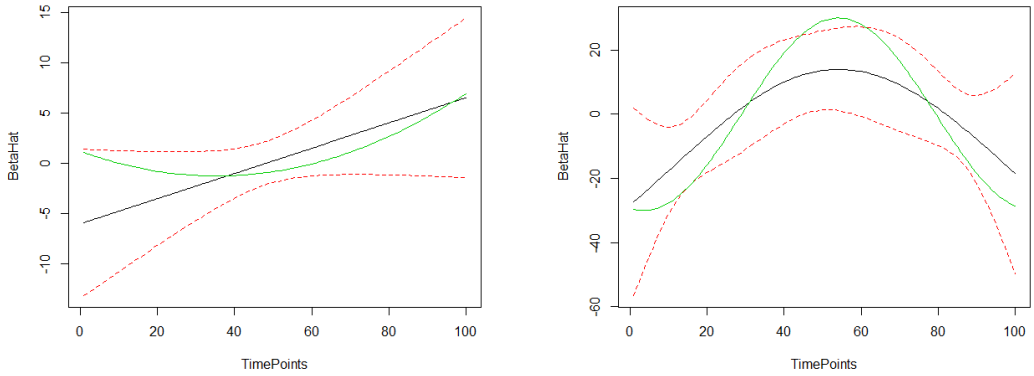



Figure B.9: An example of the distribution of $\hat{\pi}$ from the scenario: $N = 60, K_x = 18, K_e = 10$. Circles in green indicate the first group ($y = 0$) while those in red represent the second group ($y = 1$). The dotted line in black in the middle represent $\pi = 0.5$.




(a) 95% confidence interval for $\hat{\beta}_1(t)$ (b) 95% confidence interval for $\hat{\beta}_2(t)$

Figure B.10: An example of confidence intervals for $\hat{\boldsymbol{\beta}}(t)$ from the scenario: $N = 90, K_x = 30, K_e = 30$. The lines in green are the true $\boldsymbol{\beta}$, the lines in black indicate the estimators $\hat{\boldsymbol{\beta}}$, and the dotted red lines represent the boundaries of 95% confidence intervals. 'TimePoints' are equal to $100t_j$.

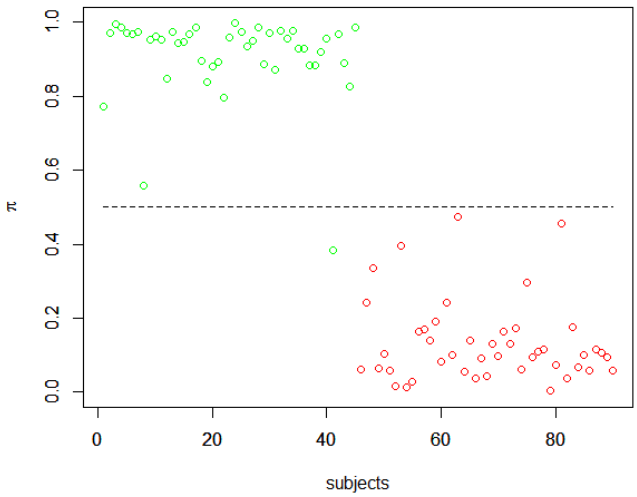
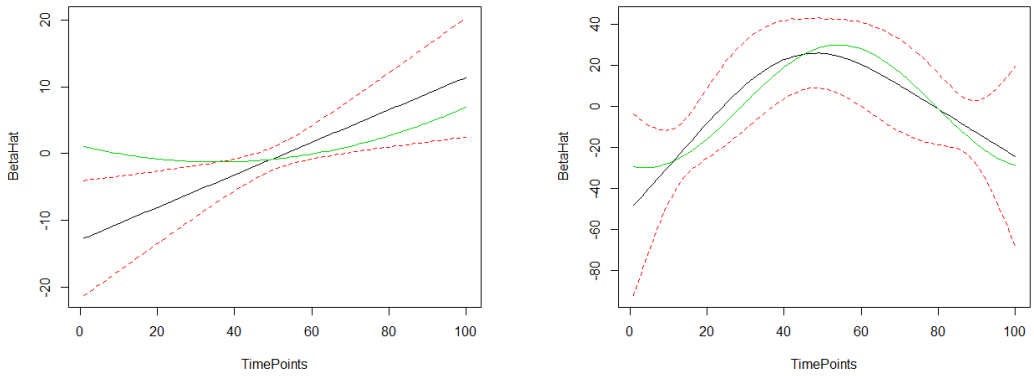



Figure B.11: An example of the distribution of $\hat{\pi}$ from the scenario: $N = 90, K_x = 30, K_e = 30$. Circles in green indicate the first group ($y = 0$) while those in red represent the second group ($y = 1$). The dotted line in black in the middle represents $\pi = 0.5$.




(a) 95% confidence interval for $\hat{\beta}_1(t)$ (b) 95% confidence interval for $\hat{\beta}_2(t)$

Figure B.12: An example of confidence intervals for $\hat{\boldsymbol{\beta}}(t)$ from the scenario: $N = 120, K_x = 35, K_e = 35$. The lines in green are the true $\boldsymbol{\beta}$, the lines in black indicate the estimators $\hat{\boldsymbol{\beta}}$, and the dotted red lines represent the boundaries of 95% confidence intervals. 'TimePoints' are equal to $100t_j$.

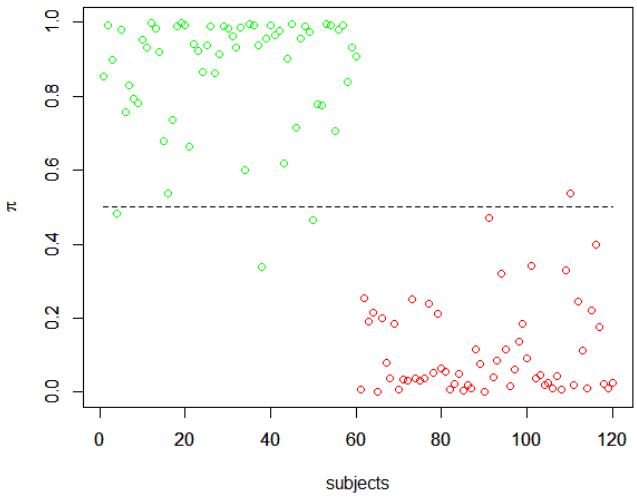



Figure B.13: An example of the distribution of $\hat{\pi}$ from the scenario: $N = 120, K_x = 35, K_e = 35$. Circles in green indicate the first group ($y = 0$) while those in red represent the second group ($y = 1$). The dotted line in black in the middle represents $\pi = 0.5$.

# Appendix C

# Derivation of $M_{ik}$ and the linearized model

## C.1 Derivation of $M_{ik}$

Using Bayes' theorem, the posterior distribution with respect to $\mathbf{z}$ has the form

$$p(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{\beta}) \propto \prod_{i=1}^{N}\prod_{k=1}^{K} \pi_{ki}^{z_{ki}} p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})^{z_{ki}}.$$

By factorizing it over $i$, it is clear that the $\{\boldsymbol{z}_i, i = 1,\ldots,N\}$ are independent under the posterior distribution. Hence,

$$\begin{aligned} E(z_{ki}|\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{\beta}) &= E(z_{ki}|\boldsymbol{x}_i,\boldsymbol{\theta},\boldsymbol{\beta}) \\ &= p(z_{ki}=1|\boldsymbol{x}_i,\boldsymbol{\theta},\boldsymbol{\beta}) \\ &= \frac{p(z_{ki}=1|\boldsymbol{\beta})p(\boldsymbol{x}_i|z_{ki}=1,\boldsymbol{\theta})}{p(\boldsymbol{x}_i|\boldsymbol{\theta},\boldsymbol{\beta})} \\ &= \frac{\pi_{ki}p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})}{p(\boldsymbol{x}_i|\boldsymbol{\theta},\boldsymbol{\beta})}. \end{aligned} \tag{C.1}$$

We know that

$$\begin{aligned} p(\boldsymbol{x}_i|\boldsymbol{\theta},\boldsymbol{\beta}) &= \sum_{\boldsymbol{z}_i} p(\boldsymbol{z}_i|\boldsymbol{\beta})p(\boldsymbol{x}_i|\boldsymbol{z}_i,\boldsymbol{\theta},\boldsymbol{\beta}) \\ &= \sum_{z_{ki}} \prod_{k=1}^{K} \left(p(z_{ki}=1|\boldsymbol{\beta})p(\boldsymbol{x}_i|z_{ki}=1,\boldsymbol{\theta})\right)^{z_{ki}} \\ &= \sum_{j=1}^{K} \pi_{ji}p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ji}). \end{aligned} \tag{C.2}$$

Thus,

$$E(z_{ki}|\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{\beta}) = \frac{\pi_{ki}p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ki})}{\sum_{j=1}^{K}\pi_{ji}p(\boldsymbol{x}_i|\boldsymbol{\theta}_{ji})}.$$

## C.2 Derivation of the linearized model

At the linearized level, we apply the first-order Taylor approximation of model (5.3) in Section 5.2.2 in the random warp $\boldsymbol{w}_{ki}$. We reconsider $g_{ki}(t)$ as a function with respect to $\boldsymbol{w}_k + \boldsymbol{w}_{ki}$. Thus, the linearization can be carried out around the estimate of $\boldsymbol{w}_k$ plus $\boldsymbol{w}_{ki}^0$ obtained from the previous step. This results in a linear mixed-effects model as follows:

$$\boldsymbol{x}_{ai}|_{z_{ki=1}} \approx \boldsymbol{x}_{ai}|_{z_{ki=1},\boldsymbol{w}_{ki}=\boldsymbol{w}_{ki}^0} + \nabla_{\boldsymbol{w}_{ki}}(\boldsymbol{x}_{ai}|_{z_{ki=1}})|_{\boldsymbol{w}_{ki}=\boldsymbol{w}_{ki}^0}(\boldsymbol{w}_{ki} - \boldsymbol{w}_{ki}^0)$$

where

$$\begin{aligned}\boldsymbol{x}_{ai}|_{z_{ki=1},\boldsymbol{w}_{ki}=\boldsymbol{w}_{ki}^0} &= \boldsymbol{\tau_{ak}}(g_{ki})|_{\boldsymbol{w}_{ki}=\boldsymbol{w}_{ki}^0} + \boldsymbol{r}_{aki} + \boldsymbol{\epsilon} \\ &= \boldsymbol{\Psi}_{ki}|_{g_{ki}=g_{ki}^0}\boldsymbol{d}_{ak} + \boldsymbol{r}_{aki} + \boldsymbol{\epsilon},\end{aligned}$$

and according to the chain rule,

$$\begin{aligned}\nabla_{\boldsymbol{w}_{ki}}(\boldsymbol{x}_{ai}|_{z_{ki=1}})|_{\boldsymbol{w}_{ki}=\boldsymbol{w}_{ki}^0} &= \left\{\frac{\partial \boldsymbol{x}_{ai}|_{z_{ki=1},t=t_j}}{\partial g_{ki}}\Big(\nabla_{\boldsymbol{w}_{ki}}\big(g_{ki}(t_j)\big)\Big)^{\mathsf{T}}\Big|_{\boldsymbol{w}_{ki}=\boldsymbol{w}_{ki}^0}\right\}_j \\ &= \left\{\frac{\partial\Big(\tau_{ak}\big(g_{ki}(t_j)\big)\Big)}{\partial g_{ki}}\Big|_{g_{ki}=g_{ki}^0}\Big(\nabla_{\boldsymbol{w}_{ki}}\big(g_{ki}(t_j)\big)\Big)^{\mathsf{T}}\Big|_{\boldsymbol{w}_{ki}=\boldsymbol{w}_{ki}^0}\right\}_j \in \mathbb{R}^{m_i\times n_w}.\end{aligned}$$

In practical, we use finite difference for calculating the derivative of $g_{ki}(t)$ with respect to $\boldsymbol{w}_{ki}$.

# Bibliography


AbdelJalil, A. A., Katzka, D. A., and Castell, D. O. (2015). Approach to the patient with dysphagia. *Am J Med*, 1138:e17–23.

Abraham, C., Cornillon, P. A., Matzner-Lober, E., and Molinari, N. (2003). Unsupervised curve clustering using b-splines. *Scandinavian Journal of Statistics. Theory and Applications*, 30(3):581–589.

Bouveyron, C. and Jacques, J. (2011). Model-based clustering of time series in group-specific functional subspaces. *Advances in Data Analysis and Classification*, 5(4):281–300.

Cardot, H., Ferraty, F., and Sarda, P. (1999). Functional linear model. *Statistics and Probability Letters*, 45:11–22.

Cardot, H., Ferraty, F., and Sarda, P. (2003). Spline estimators for the functional linear model. *Statistica Sinica*, 13:571–591.

Cardot, H. and Sarda, P. (2005). Estimation in generalized linear model for functional data via penalized likelihood. *Journal of Multivariate Analysis*, 92:24–41.

Chen, D. and Müller, H. G. (2011). Single and multiple index functional regression models with nonparametric link. *Annals of Statistics*, 39:1720–1747.

Cheng, W., Dryden, L. L., and Huang, X. Z. (2016). Bayesian registration of functions and curves. *Bayesian Analysis*, 11(2):447–475.

Chiou, J. M. and Li, P. L. (2007). Functional clustering and identifying substructures of longitudinal data. *J. R. Statist. Soc. B*, 69(4):679–699.

Couette, M. F. A. (1890). Études sur le frottement des liquides. *Annales de Chimie et de Physique*, 21:433–510.

Delaigle, A. and Hall, P. (2010). Defining probability density for a distribution of random functions. *The Annals of Statistics*, 38:1171–1193.

Dorsey, E. R., Constantinescu, R., Thompson, J. P., Biglan, K. M., Holloway, R. G., and Kieburtz, K. (2007). Projected number of people with parkinson disease in the most populous nations, 2005 through 2030. *Neurology*, 68(5):384–6.

Dou, W. W., Pollard, D., and Zhou, H. H. (2012). Estimation in functional regression for general exponential families. *Annals of Statistics*, 40:2421–2451.

Earls, C. and Hooker, G. (2017). Variational bayes for functional data registration, smoothing, and prediction. *Bayesian Analysis*, 12:557–582.

Feiginl, V. L., Lawes, C. M., Bennett, D. A., and Anderson, C. S. (2003). Stroke epidemiology: a review of population-based studies of incidence, prevalence, and case-fatality in the late 20th century. *Lancet Neurol*, 2(1):43–53.

Ferraty, F. and Vieu, P. (2006). *Nonparametric functional data analysis*. Springer Series in Statistics, Springer, New York.

Gasser, J., Müller, H. G., Köhler, W., Molinari, L., and Prader, A. (1984). Nonparametric regression analysis of growth curves. *Annals of Statistics*, 12:210–229.

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

Gervini, D. and Gasser, T. (2004). Self-modelling warping functions. *J.R.Statist.Soc.B*, 66(4):959–971.

Goldsmith, J., Bobb, J., Crainiceanu, C. M., Caffo, B., and Reich, D. (2011). Penalized functional regression. *J Comput Graph Stat.*, 20(4):830–851.

Gower, J. C. (1975a). Generalized procrustes analysis. *Psychometrika*, 40:33–51.

Gower, J. C. (1975b). Generalized procrustes analysis. *Psychometrika*, 40(1):33–51.

Green, P. J. and Richardson, S. (2000). Spatially corrrelated allocation models for count data.

Grenander, U. (1950). Stochastic processes and statistical inference. *Arkiv for Maternatik*, 1:195–277.

Hall, P. and Horowitz, J. L. (2007). Methodology and convergence rates for functional linear regression. *Annals of Statistics*, 35:70–91.

Hilgert, N., Mas, A., and Verzelen, N. (2013). Minimax adaptive tests for the functional linear model. *Annals of Statistics*, 41:838–869.

Hu, Z., Wang, N., and Carroll, R. J. (2004). Profile-kernel versus backfitting in the partially linear models for longitudinal/clustered data. *Biometrika*, 91:251–262.

Hubert, L. and Arabie, P. (1985a). Comparing partitions. *J. Classif.*, 2:193–218.

Hubert, L. and Arabie, P. (1985b). Comparing partitions. *J. Classif.*, 2:193–218.

Ieva, F., Paganoni, A. M., Pigoli, D., and Vitelli, V. (2013). Multivariate functional clustering for the analysis of ecg curves morphology. *Journal of the Royal Statistical Society. Series C*, 62(3):401–418.

Jacques, J. and Cristian, P. (2014). Functional data clustering: a survey. *Advances in Data Analysis and Classification*, 8(3):1–24.

James, G. (2007). Curve alignments by moments. *Annals of Applied Statistics*, 1(2):480–501.

James, G. M. (2002). Generalized linear models with functional predictors. *Journal of The Royal Statistical Society Series B*, 64:411–432.

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

Jiang, J. and Zhang, W. (2001). Robust estimation in generalized linear mixed models. *Biometrika.*, 88(3):753–765.

Jose, P., Douglas, B., Saikat, D., and Deepayan, S. (2017). nlme: Linear and nonlinear mixed effects models. https://cran.r-project.org/web/packages/nlme/.

Kaneko, I. (1992). A cinefluorographic study of hyoid bone movement during deglutition. *Nihon Jibiinkoka Gakkai kaiho*, 95(7):974–87.

Kang, B. S., Oh, B. M., Kim, I. S., Chung, S. G., Kim, S. J., and Han, T. R. (2010). Influence of aging on movement of the hyoid bone and epiglottis during normal swallowing: a motion analysis. *Gerontology*, 56(5):474–82.

Kellen, P. M., Becker, D. L., Reinhardt, J. M., V, D. J., and Daele (2010). Computer-assisted assessment of hyoid bone motion from videofluoroscopic swallow studies. *Dysphagia*, 25(4):298–306.

Kenneth, P. B. and David, R. A. (2004). Understanding aic and bic in model selection. *Sociological Methods Research*, 33:261–304.

Kim, W.-S., Zeng, P., Shi, J. Q., Lee, Y., and Paik, N.-J. (2017). Semi-atomatic tracking, smoothing and segmentation of hyoid bone motion from videofluoroscopic swallowing study. *PLoS ONE 12(11): e0188684. https://doi.org/10.1371/journal.pone.0188684.*

Kim, Y. H., Oh, B. M., Jung, I. Y., Lee, J. C., and Han, T. R. (2015). Spatiotemporal characteristics of swallowing in parkinson's disease. *Laryngoscope*, 125(2):389–95.

Kneip, A., Li, X., MacGibbon, K. B., and Ramsay, J. O. (2000). Curve registration by local regression. *The Canadian Journal of Statistics*, 28(1):19–29.

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

Lloyd, S. P. (1982). Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137.

Logemann, J. (1993). Manual for the videofluoroscopic study of swallowing. Austin, TX: Pro-Ed.

Ludlow, C. L., Humbert, I., Saxon, K., Poletto, C., Sonies, B., and Crujido, L. (2007). Effects of surface electrical stimulation both at rest and during swallowing in chronic pharyngeal dysphagia. *Dysphagia*, 22(1):1–10.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press.*, pages 281–297.

Marron, J., Ramsay, J. O., Sangalli, L. M., and Srivastava, A. (2015). Functional data analysis of amplitude and phase variation. *Statistical Science*, 30(4):468–484.

McCulloch, C., Searle, S., and Neuhaus, J. (2008). *Generalized, Linear, and Mixed Models.* Wiley.

Michalewicz, Z. and Hartley, S. J. (1996). Genetic algorithms+data structures = evolution programs. *Mathematical Intelligencer*, 18(3):71.

Molfenter, S. M. and Steele, C. M. (2014). Kinematic and temporal factors associated with penetration-aspiration in swallowing liquids. *Dysphagia*, 29(2):269–76.

Morris, J. S. (2015). Functional regression. *Annual Review of Statistics and Its Application 2: Online.*

Müller, H. G. (2005). Functional modelling and classification of longitudinal data. *Scandinavian Journal of Statistics*, 32:223–240.

Müller, H. G. (2011). International encyclopedia of statistical scienceed, ed. m lovric. Springer, Heidelberg, 554-555.

Müller, H. G. and Stadtmüller, U. (2005). Generalized functional linear models. *Annals of Statistics*, 33:774–805.

Nagy, S., Molfenter, S. M., Peladeau-Pigeon, M., Stokely, S., and Steele, C. M. (2014). The effect of bolus volume on hyoid kinematics in healthy swallowing. *Biomed Res Int*.

Nam, H. S., Beom, J., Oh, B. M., and Han, T. R. (2013). Kinematic effects of hyolaryngeal electrical stimulation therapy on hyoid excursion and laryngeal elevation. *Dysphagia*, 28(4):548–56.

Pai, N. J., Kim, S. J., Lee, H. J., Jeon, J. Y., Lim, J. Y., and Han, T. R. (2008). Movement of the hyoid bone and the epiglottis during swallowing in patients with dysphagia from different etiologies. *J Electromyogr Kinesiol*, 18(2):329–35.

Peng, J. and Müller, H. G. (2008). Distance-based clustering of sparsely observed stochastic processes, with application to online auctions. *The Annals of Applied Statistics*, 2(3):1056–1077.

Potratz, J. R., Dengel, G., and Robbins, J. (1992). A comparison of swallowing in three subjects using an interactive image processing system. Computer-Based Medical Systems, 1992 Proceedings, Fifth Annual IEEE Symposium.

Raket, L. L. (2016). pavpop version 0.10. http://github.com/larslau/pavpop/.

Raket, L. L., Grimme, B., Schoner, G., Igel, C., and Markussen, B. (2016). Separating timing, movement conditions and individual differences in the analysis of human movement. *PLoS Comput Biol*, 12(9).

Raket, L. L., Stefan, S., and Bo, M. (2014). A nonlinear mixed-effects model for simultaneous smoothing and registration of functional data. *Pattern Recognition Letters*, 38:1–7.

Ramsay, J. O. (1982). When the data are functions. *Psychometrika*, 47:379–396.

Ramsay, J. O. and Dalzell, C. (1991). Some tools for functional data analysis. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 539–572.

Ramsay, J. O. and Li, X. (1998). Curve registration. *J.R.Statist.Soc.B*, 60:351–363.

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

Rand, W. M. (1971a). Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc.*, 66:846–850.

Rand, W. M. (1971b). Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc.*, 66:846–850.

Rao, C. R. (1958). Some statistical methods for comparison of growth curves. *Biometrics*, 14:1–17.

Rice, J. and Silverman, B. (1991). Estimating the mean and covariance structure nonparametrically when the data are curves. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53:210–229.

Rincon, M. and Ruiz-Medina, M. D. (2012). Wavelet-rkhs-based functional statistical classification. *Advances in Data Analysis and Classification*, 6:201–217.

Ruppert, D. (2002). Selecting the number of knots for penalized splines. *Journal of Computational and Graphical Statistics.*, 11:735–757.

Ruppert, D., Wand, M., and Carroll, R. (2003). Semiparametric regression. *Cambridge University Press.*, 66.

Sam, A., Chamroukhi, F., Govaert, G., and Aknin, P. (2011). Model-based clustering and segmentation of time series with changes in regime. *Advances in Data Analysis and Classification*, 5(4):301–322.

Sangalli, L. M., Secchi, P., Vantini, S., and Veneziani, A. (2009). A case study in exploratory functional data analysis: geometrical features of the internal carotid artery. *J.Amer.Statist.Assoc.*, 104:37–48.

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

Seo, H. G., Oh, B. M., and Han, T. R. (2011). Longitudinal changes of the swallowing process in subacute stroke patients with aspiration. *Dysphagia*, 26(1):41–8.

Shi, J. Q. and Choi, T. (2011). *Gaussian Process Regression Analysis for Functional Data.* CRC Press Taylor & Francis Group.

Shi, J. Q., Murray-Smith, R., and Titterington, D. M. (2005). Hierarchical gaussian process mixtures for regression. *Statistics and Computing*, 15:31–41.

Shi, J. Q. and Wang, B. (2008). Curve prediction and clustering with mixtures of gaussian process functional regression models. *Stat Comput*, 18(3):267–283.

Shi, J. Q., Wang, B., Will, E. J., and West, R. M. (2012). Mixed-effect gaussian process functional regression models with application to dose-response curve prediction. *Statist. Med.*, 31:3165–3177.

Sobel, I. (1990). An isotropic $3 \times 3$ image gradient operator. Machine vision for three-dimensional scenes (H. Freeman editor). Academic Press, Boston.

Srivastava, A., Klassen, E., Joshi, S. H., and Jermyn, I. H. (2011a). Shape analysis of elastic curves in euclidean spaces. *IEEE Trans. Pattern.Anal.Mach.Intell*, 33(7):1415–1428.

Srivastava, A., Wu, W., Kurtek, S., Klassen, E., and Marron, J. S. (2011b). Registration of functional data using the fisher-rao metric. *Preprint. Available at arXiv:1103.3817v2[math.ST].*

Steele, C. M., Bailey, G. L., Chau, T., Molfenter, S. M., Oshalla, M., and Waito, A. A. (2011). The relationship between hyoid and laryngeal displacement and swallowing impairment. *Clinotolaryngol*, 36(1):30–6.

Sugiura and Nariaki (1978). Further analysis of the data by akaike's information criterion and the finite corrections. *Communications in Statistics, Theory and Methods*, A7:13–26.

Tang, R. and Müller, H. G. (1998). Pairwise curve synchronization for functional data. *Biometricka*, 95(4):875–889.

Tarpey, T. and Kinateder, K. J. (2003). Clustering functional data. *Journal of Classification*, 20(1):93–114.

Tokushige, S., Yadohisa, H., and Inada, K. (2007). Crisp and fuzzy k-means clustering algorithms for multivariate functional data. *Computational Statistics*, 22:1–16.

Wang, J. L., Chiou, J. M., and Müller, H. G. (2015). Review of functional data analysis. *Annu. Rev. Statist.*, pages 1–41.

Wang, T. G., Chang, Y. C., Chen, W. S., Lin, P. H., and Hsiao, T. Y. (2010). Reduction in hyoid bone forward movement in irradiated nasopharyngeal carcinoma patients with dysphagia. *Arch Phys Med Rehabil*, 91(6):926–31.

Wang, X. H., Ray, S., and Mallick, B. K. (2007). Bayesian curve classification using wavelets. *Journal of the American Statistical Association*, 102:962–973.

Wood, S. (2006). Generalized additive models: An introduction with r. Chapman & Hall.

Wu, Z. and Hitchcock, D. B. (2016). A bayesian method for simultaneous registration and clustering of functional observations. *Computational Statistics and Data Analysis*, (101):121–136.

Yabunaka, K., Sanada, H., Sanada, S., Konishi, H., Hashimoto, T., and Yatake, H. (2011). Sonographic assessment of hyoid bone movement during swallowing: a study of normal adults with advancing age. *Radiol phys and technol*, 4(1):73–7.

Zeng, P., Shi, J. Q., and Kim, W.-S. (2017). Simultaneous registration and classification for multi-dimensional functional data. *preprint. Available at: arXiv:1711.04761.*

Zhang, J., Zhou, Y., Wei, N., Yang, B., Wang, A., Zhou, H., Zhao, X., Wang, Y., Liu, L., and Ouyoung, M. (2016). Laryngeal elevation velocity and aspiration in acute ischemic stroke patients. *PloS one*, 11(9).

Zhu, H. X., Vannucci, M., and Cox, D. D. (2010). A bayesian hierarchical model for classification with selection of functional predictors. *Biometrics*, 66:463–473.