2018

Elastic Functional Regression Model

Kyungmin Ahn
ELASTIC FUNCTIONAL REGRESSION MODEL

By

KYUNGMIN AHN

A Dissertation submitted to the
Department of Statistics
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

2018

Copyright © 2018 Kyungmin Ahn. All Rights Reserved.
Kyungmin Ahn defended this dissertation on April 17, 2018.
The members of the supervisory committee were:

Anuj Srivastava
Professor Directing Thesis

Eric Klassen
University Representative

Wei Wu
Committee Member

Fred Huffer
Committee Member

The Graduate School has verified and approved the above-named committee members, and certifies that the dissertation has been approved in accordance with university requirements.
I dedicate this work to my family.
ACKNOWLEDGMENTS

First and foremost, I thank God for His grace for the accomplishment of receiving a Ph.D. degree in Statistics at Florida State University.

I would like to acknowledge my sincere thanks to my major advisor Dr. Anuj Srivastava who made this dissertation possible. I could not have completed this study so successfully without his continuous support, advice, and encouragement. I truly understand why some students call their faculty advisor as a father in academic, and I thank him for his motivation, which guided me to finish this dissertation in a perfect manner.

I would also like to extend my gratitude to my doctoral committee members, Dr. Wei Wu, Dr. Eric Klassen, and Dr. Fred Huffer. I am very thankful for their incredible help and sincere feedback throughout my research.

Dr. Wu helped me a lot especially during my second year of Ph.D. when my advisor Dr. Srivastava went on sabbatical. He has kindly welcomed me to join his group meeting and enlightened me on finishing my research. Dr. Klassen advised me on mathematical approach on my research. Without his insight, it would not be possible to conduct this research thoroughly. Dr. Huffer always reminds me of his smile. I will never forget his kind comments and suggestions for encouraging me to learn more from him.

I would like to thank my wonderful and fantastic friends: David Geb, Duncan Leaf, Shuguang Zhang, Kyubin Hwang, Seungkwan Lim, Myonghoon Park, and all of my friends at FSU.

I would also like to take this opportunity to thank my dearest parents for their constant prayers and love. Truly, I believe that they are the greatest and words cannot express the feelings I have for my parents for their persistent unconditional support. Also I give my sincere thanks to my dear brother. He has always been there for me to listen when I needed an ear. During the course of my study, he has kept me emotionally stable and entertained.

Last but not the least, I would like to thank my newly wedded wife Kacy for supporting me spiritually throughout this journey and without her support I could not have accomplished this degree. I love you.
# TABLE OF CONTENTS

List of Tables ........................................ vii  
List of Figures ....................................... viii  
Abstract ............................................. x  

## 1 Introduction 1  
1.1 Functional Data Analysis.......................... 1  
1.2 Functional Regression Model....................... 2  
1.3 Challenges of Functional Data Analysis.......... 3  
1.4 Solution of the Problems......................... 5  
1.5 Overview of Dissertation......................... 6  

## 2 Literature Review of Functional Data Analysis 8  
2.1 Functional Data Analysis......................... 8  
2.1.1 Smoothing Functional Data..................... 8  
2.1.2 Summary Statistics on Functional Data Analysis 11  
2.1.3 Functional Principal Component Analysis.... 13  
2.2 Phase - Amplitude Separation.................... 16  
2.2.1 Current Registration Idea........................ 16  
2.2.2 Phase Amplitude Separation Using Extended Fisher-Rao Metric 19  
2.3 Functional Regression Model..................... 22  
2.3.1 Functional Predictor Regression (Scalar-on-Function) 22  
2.3.2 Functional Response Regression (Function-on-Scalar) 22  
2.3.3 Function-on-Function Regression.............. 23  

## 3 Literature Review of Scalar-on-Function Regression Models 24  
3.1 Linear Scalar-on-Function Regression............. 24  
3.1.1 Functional Linear Model with basis representation 24  
3.1.2 Functional Linear Model with Functional PC Basis 26  
3.2 Nonlinear Scalar-on-Function Regression......... 28  
3.2.1 Single-Index Model............................. 28  
3.2.2 Multiple-Index Model........................... 29  
3.3 Generalization and Extensions.................... 30  
3.3.1 Including Scalar Covariates................... 30  
3.3.2 Multiple Functional Predictors................ 30  
3.3.3 Generalized Functional Linear Regression (GLM) 31  
3.4 Nonparametric Scalar-on-Function Regression.... 31  

## 4 Elastic Functional Regression Model 34  
4.1 Proposed Elastic Framework....................... 35  
4.2 Parameter Estimation............................. 36  
4.2.1 Estimation of $\beta$ Keeping $h$ Fixed.......... 36  

v
4.2.2  Estimation of a Link Function $h$ Keeping $\beta$ Fixed ........................................... 38
4.3  Prediction of Elastic Regression Model ................................................................. 39

5  Simulation Study  40
5.1  Initial Setup .............................................................................................................. 40
5.2  Using the Same Basis with the Same Number of Elements ................................. 43
  5.2.1  Estimation ........................................................................................................... 43
  5.2.2  Prediction ............................................................................................................ 45
5.3  Using the Same Basis with the Different Number of Elements ....................... 47
  5.3.1  Estimation ........................................................................................................... 47
  5.3.2  Prediction ............................................................................................................ 49
5.4  Using the Different Basis ......................................................................................... 50
  5.4.1  Estimation ........................................................................................................... 50
  5.4.2  Prediction ............................................................................................................ 51
5.5  Comparison with Nonparametric Regression Models ........................................ 53
5.6  Additional Work ...................................................................................................... 54

6  Application to Real Data  56
6.1  Gait in Parkinson’s Disease ...................................................................................... 56
  6.1.1  Estimation ........................................................................................................... 57
  6.1.2  Prediction ............................................................................................................ 59
6.2  Metabonomic 1H-NMR (Onion) ............................................................................ 60
  6.2.1  Estimation ........................................................................................................... 61
  6.2.2  Prediction ............................................................................................................ 63
6.3  Historical Stock ......................................................................................................... 63
  6.3.1  Estimation ........................................................................................................... 64
  6.3.2  Prediction ............................................................................................................ 66

7  Summary and Future Work  68
7.1  Summary .................................................................................................................. 68
7.2  Directions of Future Works ..................................................................................... 68

References ..................................................................................................................... 70
Biographical Sketch ....................................................................................................... 74
LIST OF TABLES

5.1 MSPE of each method using the same basis with the same number of elements . . . . 46
5.2 MSPE of each method using the same number of basis with the different number of
elements ............................................................... 50
5.3 MSPE of each method using the different basis ................................................. 52
5.4 MSPE of Nonparametric Regression Models on Simulated Data ....................... 54
5.5 The average and the standard deviation (in parentheses) of the five MSPEs for three
model-based methods on simulated test data. .............................................. 54
5.6 The average and the standard deviation (in parentheses) of the five MSPEs for non-
parametric regression model under the two distances. ................................ 54
6.1 MSPE of each model on Gait in Parkinson’s Disease Data ............................. 59
6.2 MSPE of each model on Onion data .......................................................... 63
6.3 MSPE of each model on Historical Stock data .............................................. 67
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Example of Functional Data</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Example of functional data with and without phase variability.</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>Example of Representing Function</td>
<td>10</td>
</tr>
<tr>
<td>2.2</td>
<td>Example of smoothing of functional data by changing the amount of smoothing penalty.</td>
<td>11</td>
</tr>
<tr>
<td>2.3</td>
<td>Sample Mean of Canadian Weather</td>
<td>12</td>
</tr>
<tr>
<td>2.4</td>
<td>Covariance of Canadian Weather</td>
<td>13</td>
</tr>
<tr>
<td>2.5</td>
<td>Correlation of Canadian Weather</td>
<td>13</td>
</tr>
<tr>
<td>2.6</td>
<td>Example of FPCA on <em>Toy Data</em></td>
<td>14</td>
</tr>
<tr>
<td>2.7</td>
<td>First three principal directions</td>
<td>15</td>
</tr>
<tr>
<td>2.8</td>
<td>Example of Pairwise Registration</td>
<td>17</td>
</tr>
<tr>
<td>2.9</td>
<td>Example of Multiple Registration</td>
<td>18</td>
</tr>
<tr>
<td>2.10</td>
<td>Example of Pinching Effect on Pairwise Registration [38]</td>
<td>19</td>
</tr>
<tr>
<td>2.11</td>
<td>Example of Phase-Amplitude Separation</td>
<td>21</td>
</tr>
<tr>
<td>3.1</td>
<td>Example of Tecator Data</td>
<td>26</td>
</tr>
<tr>
<td>3.2</td>
<td>Example of Single-Index Model</td>
<td>29</td>
</tr>
<tr>
<td>3.3</td>
<td>Nonlinear Relationship between ( y ) and ( \hat{y} ) on the training set</td>
<td>30</td>
</tr>
<tr>
<td>3.4</td>
<td>Example of Nonparametric Regression</td>
<td>32</td>
</tr>
<tr>
<td>4.1</td>
<td>The left panel shows a number of predictor functions ( {f_i} ), the right panel shows the alignment of these functions to the regression coefficient</td>
<td>38</td>
</tr>
<tr>
<td>4.2</td>
<td>The profile of ( H ) over a 2D domain.</td>
<td>38</td>
</tr>
<tr>
<td>5.1</td>
<td>Simulated Data</td>
<td>41</td>
</tr>
<tr>
<td>5.2</td>
<td>Training Set of Simulated Data</td>
<td>42</td>
</tr>
<tr>
<td>5.3</td>
<td>Test Set of Simulated Data</td>
<td>42</td>
</tr>
<tr>
<td>5.4</td>
<td>Warped functions and warping functions on the training set</td>
<td>43</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>5.5</td>
<td>Warped functions and warping functions on the test set</td>
<td></td>
</tr>
<tr>
<td>5.6</td>
<td>Evolution of Cost Function $H$ for each link function $h$</td>
<td></td>
</tr>
<tr>
<td>5.7</td>
<td>Prediction Plots of Regression Models</td>
<td></td>
</tr>
<tr>
<td>5.8</td>
<td>Warped functions and warping functions on the training set</td>
<td></td>
</tr>
<tr>
<td>5.9</td>
<td>Warped functions and warping functions on the test set</td>
<td></td>
</tr>
<tr>
<td>5.10</td>
<td>Evolution of Cost Function $H$ for each link function $h$</td>
<td></td>
</tr>
<tr>
<td>5.11</td>
<td>Prediction Plots of Regression Models</td>
<td></td>
</tr>
<tr>
<td>5.12</td>
<td>Warped functions and warping functions on the training set</td>
<td></td>
</tr>
<tr>
<td>5.13</td>
<td>Warped functions and warping functions on the test set</td>
<td></td>
</tr>
<tr>
<td>5.14</td>
<td>Evolution of Cost Function $H$ for each link function $h$</td>
<td></td>
</tr>
<tr>
<td>5.15</td>
<td>Prediction Plots of Regression Models</td>
<td></td>
</tr>
<tr>
<td>6.1</td>
<td>Gait in Parkinson’s Disease Data</td>
<td></td>
</tr>
<tr>
<td>6.2</td>
<td>${f_i}$ from Gait in Parkinson’s Disease Data</td>
<td></td>
</tr>
<tr>
<td>6.3</td>
<td>Training Set of Gait in Parkinson’s Disease Data</td>
<td></td>
</tr>
<tr>
<td>6.4</td>
<td>Test Set of Gait in Parkinson’s Disease Data</td>
<td></td>
</tr>
<tr>
<td>6.5</td>
<td>Prediction Plots of Regression Models on Gait in Parkinson’s Disease Data</td>
<td></td>
</tr>
<tr>
<td>6.6</td>
<td>Metabonomic 1H-NMR data (Onion)</td>
<td></td>
</tr>
<tr>
<td>6.7</td>
<td>${f_i}$ from Onion Data</td>
<td></td>
</tr>
<tr>
<td>6.8</td>
<td>Training Set of Onion Data</td>
<td></td>
</tr>
<tr>
<td>6.9</td>
<td>Test Set of Onion Data</td>
<td></td>
</tr>
<tr>
<td>6.10</td>
<td>Prediction Plots of Regression Models on Onion Data</td>
<td></td>
</tr>
<tr>
<td>6.11</td>
<td>Historical Stock Data</td>
<td></td>
</tr>
<tr>
<td>6.12</td>
<td>Training Set of Historical Stock Data</td>
<td></td>
</tr>
<tr>
<td>6.13</td>
<td>Test Set of Historical Stock Data</td>
<td></td>
</tr>
<tr>
<td>6.14</td>
<td>Prediction Plots of Regression Models on Historical Stock Data</td>
<td></td>
</tr>
</tbody>
</table>
ABSTRACT

Functional variables serve important roles as predictors in a variety of pattern recognition and vision applications. Focusing on a specific subproblem, termed scalar-on-function regression, most current approaches adopt the standard $\mathbb{L}_2^2$ inner product to form a link between functional predictors and scalar responses. These methods may perform poorly when predictor functions contain nuisance phase variability, i.e., predictors are temporally misaligned due to noise. While a simple solution could be to pre-align predictors as a pre-processing step, before applying a regression model, this alignment is seldom optimal from the perspective of regression.

In this dissertation, we propose a new approach, termed elastic functional regression, where alignment is included in the regression model itself, and is performed in conjunction with the estimation of other model parameters. This model is based on a norm-preserving warping of predictors, not the standard time warping of functions, and provides better prediction in situations where the shape or the amplitude of the predictor is more useful than its phase. We demonstrate the effectiveness of this framework using simulated and real data.
CHAPTER 1
INTRODUCTION

1.1 Functional Data Analysis

Functional data analysis (FDA) [31] is a branch of statistics that deals with functional variables, i.e. variables that are real- or vector-valued functions on a fixed domain. FDA has gained prominence in recent years because of the centrality of functional data in many applications. In fact, many problems involving so-called big data often consists of time-varying measurements, leads to functional data. Such data can be found in all branches of science, including vision, pattern recognition, biology, medical imaging, bioinformatics, social science, and so on. Research in Functional Data Analysis deals with:

- Summary Statistics
- Classification and Clustering
- Regression
- Modeling
- Dimension Reduction
- Registration
- Functional Experiment Design
- Nonparametric Modeling
- Bayesian Method

Figure 1.1 shows the example of functional data. The first panel shows 200 stock prices from Historical Stock data and the second panel shows the temperature of 30 Canadian areas from Canadian Weather data.
One of the fast growing subtopics in FDA is the problem of regression involving functional variables, either as predictors or responses or both. Depending whether the predictor or the response or both are functional variables, Morris [28] categorizes functional regression problems into three types: (1) functional predictor regression (scalar-on-function), (2) functional response regression (function-on-scalar) and (3) function-on-function regression. In this dissertation, we are interested in the problem of scalar-on-function regression [7, 8, 32], where the predictor \( \{ f \} \) is real-valued function over a fixed interval, element of a pre-specified functional space \( \mathcal{F} \), and the response variable \( \{ y \} \) is a scalar. A simple baseline model used frequently in the literature is called the functional linear model (FLM) given by:

\[
y_i = \alpha + \left( \int_0^T f_i(t) \beta(t) \, dt \right) + \epsilon_i, \quad i = 1, \ldots, n.
\]

Here \( \beta \in \mathcal{F} \) plays the role of regression coefficient, \( f_i s \) are observations of the predictor, \( y_i s \) are observations of the response, and \( \epsilon_i s \) are the measurement errors. Various advancements beyond this simple model have been proposed but the inner-product based mapping from \( \mathcal{F} \) to \( \mathbb{R} \) remains central to this framework.
1.3 Challenges of Functional Data Analysis

While the use of functional data has grown in recent years, there has also been a growing awareness of a problem/issue that is specific to functional data. Functional data most often comes with a phase variability, i.e., a lack of registration between peaks and valleys across functions. This situation arises, for example, when using bio-signals for diagnosing medical conditions, and where the measurements of signals across subjects lack synchronizations. Different subjects have different temporal evolutions of their bio-signals, introducing an intrinsic phase variability in the data. In mathematical terms, the predictor data is not \( \{f_i\} \) but rather \( \{(f_i \circ \gamma_i)\} \), where \( \gamma_i \)'s are unknown time warping functions that misalign predictor functions. (There are other forms of phase variability also, such \((f \circ \gamma_i)\dot{\gamma}_i\), \((f \circ \gamma_i)\sqrt{\dot{\gamma}_i}\), and so on, depending on if the time warping affects the amplitudes or not. The nature and expression of time warping is ultimately dependent on the application.) In some cases these misalignments or phase variability are simply linear or affine shifts, while in other cases the misalignments are nonlinear.

In analysis and modeling of functional data, it is often advantageous to separate the phase and amplitude components in the functional data. The separation results in alignment of peaks and valleys across functions using nonlinear time warpings. These warpings correspond to the phase components and the aligned functions correspond to the shape or amplitude components. One may envision requiring regression models where these components – phase and shape – are both treated as predictors. However, there are some other situations where only one of them, most notably, the shape of the function, that may be of interest in predicting a response variable. This situation arises, for instance, in cases where the response depends primarily on the number of modes, or the relative heights of the modes, of the predictor functions. The locations of these modes and anti-modes do not play a role and are considered nuisance variables. Motivated by this problem, we develop a regression model where only the shape (or amplitude) of a function is considered as a predictor and its phase is removed from the consideration.

To illustrate these concepts, consider the two examples shown in Figure 1.2. On the left we see the Tecator data that has been used commonly in functional regression papers. The predictor functions here was well registered and one can use them directly in a statistical model without any consideration of phase or phase separation. The right side shows a different situation of Medfly data [9], where the records of number of eggs laid by Mediterranean Fruit Fly in each of 25 days
of the total of 30 flies are displayed in the left panel. While each curve has a similar number of peaks and valleys, these features are not well aligned across subjects. This data contains a large phase variability and their consideration becomes important in any statistical analysis. One can remove this misalignment using a registration step, as shown in the right panel. One can imagine a certain response variable that depends primarily on the shapes of these curves and not on the location variability, and consequently shape-based regression becomes important.

**Remark 1.** We remark that this is a special situation and one should not always expect the shapes of predictor functions to be predominant in prediction. Phase components may also carry important information about the responses and one can not always ignore them. However, in some cases, as illustrated through examples presented later in this paper, shape can be the primary predictor and one wants model that can accomplish this prediction.

![Tecator Benchmark Data](image1.png) ![Medfly Data](image2.png)

**Figure 1.2:** Example of functional data with and without phase variability.

One approach to handling phase variability is to separate the phase and amplitude components in the predictor functions, as described in [25, 38] and others. The separation results in alignment of peaks and valleys across functions using nonlinear time warpings. These warpings correspond to the phase components and the aligned functions correspond to the shape or amplitude components. One may envision regression models where both these components—phase and shape—are useful as predictors in the model. However, there are some other situations where only one of them, most notably the shape of the function, that may be of interest in predicting a response variable. This situation arises, for instance, in cases where the response depends primarily on the number of modes of the predictor functions. The locations of these modes and anti-modes are less important.
and, therefore, phase components are considered nuisance variables. Motivated by such situations, we develop a regression model where only the shape (or amplitude) of a function is considered as a predictor and its phase is removed from the consideration.

Motivated by the necessity of being invariant to phase component of predictors, one may be tempted to use a model of the type:

\[ y_i = \alpha + \sup_{\gamma_i} \left( \int_0^T f_i(\gamma_i(t)) \beta(t) \, dt \right) + \epsilon_i, \quad i = 1, \ldots, n. \]  
(1.3.1)

However, this has a major shortcoming in that the alignment is based on standard warping under the standard \( L^2 \) metric. As described in several places, see e.g. Marron [25], Ramsay-Silverman [33], and Srivastava-Klassen [38], the alignment of functions under the \( L^2 \) leads to a degeneracy termed the pinching effect. Some authors avoid or minimize pinching by restricting the set of warpings to a much smaller set, in a pre-determined manner. This restriction is unnatural as it is mostly impossible to pre-determine the optimal set of warpings needed to align future data. Another possibility is to assume the presence of additional information in form of landmarks, points that need to be registered across functions, to help avoid pinching and misalignment. However, the availability of landmarks is rare in real data.

### 1.4 Solution of the Problems

In this dissertation, we present a novel solution that is motivated by the use of the Fisher-Rao metric in functional data alignment [38, 39]. In fact, this elastic functional data analysis (EFDA) framework suggests several ideas, only one of which is pursued in this paper. EFDA is based on replacing the use of \( L^2 \) inner product and the \( L^2 \) distance between functions by the Fisher-Rao Riemannian metric and the Fisher-Rao distance between these functions. The latter provides better mathematical and numerical properties, and indeed leads to a superior registration between functions. The challenge in using the original Fisher-Rao metric comes from its complicated expression, but that is overcome using the square root velocity function (SRVF), as described in Srivastava et al. [39]. One works with the SRVFs \( q_i \)s instead of the predictors \( f_i \)s and the Fisher-Rao metric becomes the standard \( L^2 \) metric. In this framework, the time warpings of \( q_i \)s, given by \((q_i \circ \gamma_i)\sqrt{\gamma_i}\), are norm preserving. That is, \( \|q_i\| = \|(q_i \circ \gamma_i)\sqrt{\gamma_i}\| \) for \( q_i \in L^2 \) and all warpings \( \gamma_i \), and thus pinching is no longer possible. This, in turn, suggests two ways of fixing the problem in Eqn. 1.3.1:
1. Use Fisher-Rao Metric and SRVF Representation: One can compute SRVFs of the given predictors, and then simply replace the term \( \sup_{\gamma_i} \langle f_i \circ \gamma_i, \beta \rangle \) in Eqn. 1.3.1 by the Fisher-Rao based inner product: \( \sup_{\gamma_i} \langle (q_i \circ \gamma_i) \sqrt{\dot{\gamma}_i}, \beta \rangle \). Since any warping of \( q_i \) in this way preserves its norm, the pinching problem is avoided.

2. Use a Norm-Preserve Warping and \( L^2 \) Metric: The other option is to change the basic definition of the warping itself, from the classical composition \( (f_i \circ \gamma_i) \) to the norm-preserving action \( ((f_i \circ \gamma_i) \sqrt{\dot{\gamma}_i}) \). In the new definition, a warping changes both the location and the height of a function value. This suggests using \( \sup_{\gamma_i} \langle (f_i \circ \gamma_i) \sqrt{\dot{\gamma}_i}, \beta \rangle \) in Eqn. 1.3.1. That is, we assume that \( f_i \)s are already in SRVF space and use them as such. This process may be useful when the data is noisy and a further SRVF transformation enhances this noise due to the presence of a derivative. By treating \( f_i \)s as SRVFs, one obtains the nice properties of this framework and avoids enhancing the noise. On the other hand, this warping is different from the typical warping \( f \circ \gamma_i \) used in the alignment literature.

Each of these models avoid the pinching effect, and have their own pros and cons. Ultimately, the choice of a model depends on the nature of the data and the goals of the application. The response variable in both these models is invariant to the action of the time warping group on the predictor functions.

### 1.5 Overview of Dissertation

The rest of this dissertation is organized as follows. We briefly summarize the concepts and methods of Functional Data Analysis in Chapter 2. We first introduce how the functions can be constructed from the data by using basis and kernel smoothing method. Then we summarize the functional version of summary statistics and principal component analysis. We also summarize the alignment/registration of functions and address the problem called a pinching effect on \( L^2 \) space. Then we propose the solutions of avoiding pinching effect using extended Fisher-Rao metric. Chapter 2 introduces three types of functional regression models which can be categorized as scalar-on-function, function-on-scalar, and function-on-function. In Chapter 3, we focus on scalar-on-function regression model where the predictor is function and response variable is scalar or vector. We summarize linear, non-linear, extended version of linear, and non-parametric scalar-on-function regression models in this chapter. Then we address the problems that the past techniques on scalar-on-function regression models ignore the phase variation or have a pinching effect problem. In Chapter 4, we propose an Elastic Functional Regression Model which can handle the
issue/problem of previous regression models. We demonstrate the effectiveness of our proposed model by using simulated data in Chapter 6. In Chapter 7, we apply the elastic functional regression model to real data such as gait-signals and historical stock data to demonstrate the efficiency and prediction accuracy compared to the other traditional regression models. Finally, Chapter 8 provides concluding remarks and future research directions.
CHAPTER 2

LITERATURE REVIEW OF FUNCTIONAL DATA ANALYSIS

In this chapter, we summarize the state of the art in functional data analysis. We introduce the general construction of the functions by using basis and kernel method, sample mean and sample variance of the functions, sample covariance and sample correlation of the functions, functional principal component analysis known as FPCA, and registration/alignment of the functions. We will also describe three types of functional regression models depend on whether the predictor or the response or both are functional variables in functional data analysis.

2.1 Functional Data Analysis

The most general space for functional data analysis is a *Hilbert space*. Hilbert space can be defined as a complete normed space (Banach space) where norm derives from an inner product and inner product can be defined as $||f|| = \langle f, f \rangle^{1/2}$. The most well-known example of this space is $L^2$ space for real square-integrable functions defined on $[a, b]$ with $\langle f, g \rangle = \int_a^b fg$. Then we can define a function $f$ as a real-valued function over a fixed interval $[0, T]$. The functions are absolutely continuous on $[0, T]$ and let $\mathcal{F}$ denotes the set of all such functions.

2.1.1 Smoothing Functional Data

Let’s assume that $g(t)$ is the observed function through the model: $g(t_i) = f(t_i) + \epsilon(t_i)$ for $i = 1, \ldots, m$ where $\epsilon(t)$ is the residuals. Then we can reconstruct the original function $f(t)$ from observed function $g(t)$ by using a linear smoother,

$$\hat{f}(t) = \sum_{i=1}^m s_{ij}g(t_i) \quad \text{or} \quad f = Sg \quad (2.1.1)$$

where $s_{ij}$ is the weight that the point $t_j$ gives to the point $t_i$. $f, g \in \mathbb{R}^{m \times 1}, S \in \mathbb{R}^{m \times m}$ are in matrix forms. There are two ways for representing functions: 1) using basis function and 2) using kernel smoothing method.
Representing Functions by Basis Functions.

We can represent a function by a set of basis functions, $\theta_k$, $k = 1, \ldots, K$ when the data are assumed to belong to $L^2$ space. An example of types of basis can be Fourier basis or B-spline basis functions. Then we can represent a function $f$ by a linear expansion

$$f(t) = \sum_{k=1}^{\infty} p_k \theta_k(t) \text{ or } f = \Theta p$$

(2.1.2)

where $f$ is a vector containing the values of the function $f(t_i)$, $\Theta \in \mathbb{R}^{m \times K}$ is a matrix whose columns are the basis functions $\theta_k$, and $p \in \mathbb{R}^{K \times 1}$ is the vector of length $K$ of the coefficient $p_k$. We can estimate the basis coefficients, $p_k$, by minimizing the least-squares criterion,

$$p^* = \arg \min_p (f - \Theta p)^T (f - \Theta p)$$

(2.1.3)

The closed-form solution to this problem is

$$p^* = (\Theta^T \Theta)^{-1} \Theta^T f$$

(2.1.4)

Then the smoothing matrix can be expressed as: $S = \Theta(\Theta^T \Theta)^{-1} \Theta^T$ where $\Theta \in \mathbb{R}^{m \times K}$.

In here, we could truncate $\infty$ into finite case in terms of $K$ known basis functions $\theta_k$ and coefficients $p_k$. By setting the dimension of the basis expansion $K$, we can handle the degree to which the functions are smoothed. For example, if we choose $K$ large enough then we can fit the data well, but might fit any additive noises and if we choose $K$ too small then we remove structure from the data that might be important to the analysis. In general, there is no general acceptable basis in functional data analysis so we choose basis depends on the type of the data. For example, we choose Fourier basis if the data are periodic and B-spline basis if the data are aperiodic.

Figure 2.1 shows the example of representing function by using basis functions. In this figure, a function fits the data well if we choose $K = 20$ which is large enough and function is smoothed but does not fit the data well when $K$ is becoming smaller.

Roughness Penalty. We can add a roughness penalty to the least-square fit if the smoothing penalization is required. In this case, the projection (smoothing) matrix $S$ is:

$$S = \Theta(\Theta^T \Theta + \lambda R)^{-1} \Theta^T$$

(2.1.5)
where $R$ is the penalization matrix, $\Theta \in \mathbb{R}^{m \times K}$ is the basis matrix, and $\lambda$ is the smoothing parameter. When $\lambda$ is large enough, the roughness increasingly penalized, $f(t)$ becomes smooth and when $\lambda$ gets small, the penalty reduces, $f(t)$ fits the data better. Generally, roughness is the square of the second derivative, $[D^2 f(t)]^2$. Hence, a natural measure of function’s roughness is the integrated squared second derivative as follows.

$$R = \int [D^2 f(t)]^2 dt$$

The example of smoothing of functional data by changing the amount of smoothing penalty is shown in Figure 2.2. We can fit the data well when we choose $\lambda$ small ($\lambda = 0$) and the function is smoothed when we choose $\lambda$ large enough ($\lambda = 50$).

**Kernel Smoothing.**

The nonparametric method also can be used to represent functional data. In particular, we use kernel smoothing method using *Nadaraya-Watson* \[30\] estimator.

$$S = (s_{ij}) = \frac{K((t_i - t_j)/v)}{\sum_{k=1}^{n} K((t_i - t_k)/v)}$$

(2.1.6)

where $K(\cdot)$ is the kernel function and $v$ is the smoothing parameter.
2.1.2 Summary Statistics on Functional Data Analysis

Mean and Variance of the functions.

Similar to classical summary statistics, we can apply this idea into functional data in a cross-sectional manner. Consider a set of given functions $\mathcal{F} = \{f_i, i = 1, 2, \ldots, n\}$, with each $f_i \in L^2$, then we can define the sample mean function, $\bar{f}(t)$ as:

$$\bar{f}(t) = \mu(t) = \frac{1}{n} \sum_{i=1}^{n} f_i(t)$$

Figure 2.3 shows the example of functional sample mean of Canadian Weather data. There are total 35 curves in this data. Black dashed lines are the original curves and red solid line is the sample mean of the original curves.

Then we can define the variance of the functions as,

$$\text{var}(f(t)) = \sigma^2(t) = \frac{1}{n-1} \sum_{i=1}^{n} \left(f_i(t) - \mu(t)\right)^2$$

and the standard deviation is the point-wise square root of the variance function.
σ(t) = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (f_i(t) - \mu(t))^2}

Covariance of Functions.

Define covariance function \( \sigma(s, t) : [0, 1] \times [0, 1] \rightarrow \mathbb{R} \) according to:

\[
\text{cov}(s, t) = \sigma(s, t) = \frac{1}{n-1} \sum_{i=1}^{n} (f_i(s) - \mu(s))(f_i(t) - \mu(t))
\]

\( \sigma(s, t) \) is by definition symmetric and positive semidefinite.

Figure 2.4 shows the sample covariance of the functions of Canadian weather data.

Correlation of Functions.

Correlation of functions can be defined as:

\[
\rho(s, t) = \frac{\sigma(s, t)}{\sigma(s)\sigma(t)}
\]

Figure 2.5 shows the correlation of the functions of Canadian weather data.
2.1.3 Functional Principal Component Analysis

The motivation for Functional Principal Component Analysis (FPCA) [36, 37, 43] is that the directions of high variance will contain more information than direction of low variance. Let
Let $f_1, \cdots, f_n$ be a given set of functions, the optimization problem for FPCA can be written as

$$
\min_{\hat{w}_i} E \left\| f - \hat{f} \right\|^2 
$$

(2.1.7)

where $\hat{f} = \bar{f} + \sum_{i=1}^{n} \beta_i w_i(t)$ is the FPCA approximation of $f$ with corresponding mean $\bar{f}$, $\beta_i = \int (f - \bar{f}) w_i(t) \, dt$, and basis functions $\{w_i(t)\}$.

We then use the sample covariance function $\text{cov}(s,t)$ to form a sample covariance matrix $S$. Taking the SVD, $S = U \Sigma V^T$ we can calculate the directions of principle variability in the given functions using the first $p \leq n$ columns of $U$. Moreover, we can calculate the observed principal coefficients as $\langle f_i, U_j \rangle$.

![Image](image.png)

(a) Toy Data  
(b) $\mu(t)$  
(c) $\lambda_i$

Figure 2.6: Example of FPCA on Toy Data
Figure 2.6 shows the example of the FPCA. In here, the first panel shows the functions of Toy Data, the second panel shows the sample mean $\tilde{f}(t)$ of the functions, and the third panel presents the singular values, $\lambda_i$. Figure 2.7 shows the first three principal direction, $U1, U2,$ and $U3$ from singular value decomposition of $\hat{\sigma}(s, t)$, respectively.

![Figure 2.6: Functions of Toy Data](image1)

![Figure 2.7: First three principal directions](image2)

Figure 2.7: First three principal directions
2.2 Phase - Amplitude Separation

An important problem of the functional data analysis is the alignment of the functional data using time warping functions. In this section, we will describe the alignment of the functional data on $L^2$ space and address the problem when aligning functions on $L^2$ which is known as a pinching effect. To overcome this problem, we extended this idea to Fisher-Rao Metric space.

2.2.1 Current Registration Idea

Let’s assume that $F \subset L^2$. Then we can align the functions in two different cases: when there are 1) two functions (pairwise alignment) and 2) more than two functions (multiple alignment). In this technique, we use group of warping functions $\Gamma_I$ for the registration problems [39, 41]. We define $\gamma : [0, 1] \rightarrow [0, 1]$ that satisfies the following properties:

1. $\gamma(0) = 0, \gamma(1) = 1$
2. $\gamma$ is absolutely continuous.
3. For all $t_1, t_2 \in [0, 1]$, if $t_1 < t_2$ then $\gamma(t_1) \leq \gamma(t_2)$

Such a $\gamma$ is called a boundary-preserving diffeomorphism of $[0, 1]$ and let $\Gamma_I$ denote the set of all such functions.

1. Pairwise Alignment Problem: Given any two functions $f_1$ and $f_2$ in $F$, we define their pairwise alignment to be the problem of finding a warping function $\gamma$ such that a certain term $E[f_1, f_2 \circ \gamma]$ is minimized.

   $$\gamma^* = \arg \min_{\gamma \in \Gamma_I} E[f_1, f_2 \circ \gamma]$$ (2.2.1)

   Then, for any $t \in [0, 1]$, the value $f_1(t)$ is said to be registered to $f_2(\gamma^*(t))$.

   Figure 2.8 shows the example of pairwise registration. In Figure 2.8a, two functions, $f_1$ (solid blue curve) and $f_2$ (dashed red curve) are shown. By finding optimal warping function using Eqn. 2.2.1, function $f_2$ is warped in Figure 2.8b. The optimal time warping function is shown in Figure 2.8c.

2. Multiple Alignment: Given a set of functions $f_i$ where $i = 1, \ldots, n$, we define their multiple registration to be the problem of finding a set of warping functions $\{\gamma_i| i = 1, 2, \ldots, n\}$ such that, for any $t \in [0, 1]$, the values $f_i(\gamma_i(t))$ are said to be registered.
Figure 2.8: Example of Pairwise Registration

Figure 2.9 shows the example of multiple registration on toy data. Similar to the example of pairwise registration, original data, warped data, and corresponding warping functions are shown in Figure 2.9a, 2.9b, and 2.9c, respectively.

One of the drawbacks of this registration process on $L^2$ is that the action of $\Gamma_I$ on $F$ under the $L^2$ metric is not by isometries since $\dot{\gamma}(\gamma^{-1}(s)) \neq 1$, we can see generally

$$||f_1 \circ \gamma - f_2 \circ \gamma|| \neq ||f_1 - f_2||$$

Despite identical warping and preservation of matches between the two functions, the $L^2$ norm of their difference changes. This relates the problem in using $L^2$ to perform matching is called the pinching effect.
Pinching Effect. The basic idea of pinching effect is that in matching of two functions, say \( f_1 \) and \( f_2 \) using \( \inf_{\gamma \in \Gamma_I} ||f_1 - f_2 \circ \gamma|| \), one can squeeze or pinch a large part of \( f_2 \) and make this cost function arbitrarily close to zero.

Figure 2.10 shows the example of pinching effect on pairwise alignment. The top row shows a degenerate analytical solution to match function under \( L^2 \) norm for functions. The bottom row shows a similar problem in a numerical procedure for the same matching.

A common approach of handling this pinching effect problem is to impose an additional regularization term in the optimization that constraints the roughness of \( \gamma \).

\[
\inf_{\gamma \in \Gamma_I} \left( ||f_1 - f_2 \circ \gamma|| + \lambda R(\gamma) \right), \tag{2.2.2}
\]
where $R(\gamma)$ is the regularization term, $R(\gamma) = \int [D^2\gamma(t)]^2 \, dt$ and $\lambda > 0$ is a constance parameter. This approach solved the pinching effect problem at some point, but this solution has several problems including the violation of inverse symmetry. To solve this problem, we introduce the Square Root Slope Function (SRSF) using extended Fisher-Rao Metric space which is a more comprehensive solution in registration problem avoiding a pinching effect.

### 2.2.2 Phase Amplitude Separation Using Extended Fisher-Rao Metric

A. Srivastava et al. [39] described that functions can be separate into two components, phase and amplitude. This can be implemented using SRSFs and karcher mean by the “Complete Alignment Algorithm”. J. D. Tucker et al. [41] expanded this idea to demonstrate the functional alignment by using Square-Root Slope Functions (SRSFs) that defined as,

$$ q(t) = \text{sign}(\dot{f}(t)) \sqrt{|\dot{f}(t)|} $$

SRSF is in $\mathbb{R}$ with a group action given as $\langle q, \gamma \rangle \equiv (q \circ \gamma) \sqrt{\gamma}$. It is easy to show that for two SRSF’s $q_1$ and $q_2$, $||q_1 - q_2|| = ||(q_1, \gamma) - (q_2, \gamma)||$. Hence, the action of the group $\Gamma$ is isometric on the space of SRSFs. From the paper, we could align the functions and the functions can be separated as

![Figure 2.10: Example of Pinching Effect on Pairwise Registration [38]](image)
phase and amplitude based on karcher mean of the warping functions. Karcher mean (also known as Frechet mean, intrinsic mean or the centroid) is defined as follows.

\[ \mu_f = \underset{f \in \mathcal{F}}{\text{arg min}} \sum_{i} d(f, f_i)^2 \]

\[ \mu_q = \underset{q \in \mathbb{L}^2}{\text{arg min}} \sum_{i} \left( \inf_{\gamma_i \in \Gamma} \| q - (q_i, \gamma_i) \|^2 \right) \]

Note that two equations above are actually same but the first one (\( \mu_f \)) is in the function space \( \mathcal{F} \) and other (\( \mu_q \)) is in SRSF space \( \mathbb{L}^2 \). We could compute the mean of warping functions by following algorithm:

**Algorithm 2.1 Karcher Mean of Warping Functions**

1: Let \( \psi_i = \sqrt{\gamma_i} \) be the SRSFs for the given warping functions. Initialize \( \mu_\psi \) to be one of the \( \psi_i \)'s or simply \( w/\|w\| \), where \( w = \frac{1}{n} \sum_{i=1}^{n} \psi_i \).

2: \textbf{while} \( \| \bar{v} \| \) is small \textbf{do}
3: \quad \textbf{for} \( i = 1:n \) \textbf{do}
4: \quad \quad Compute the shooting vector \( v_i = \frac{\theta_i}{\sin(\theta_i)} (\psi_i - \cos(\theta_i) \mu_\psi), \theta_i = \cos^{-1} (\langle \mu_\psi, \psi \rangle) \). By definition, each of these \( v_i \in T_{\mu_\psi}(\mathbb{S}_\infty) \).
5: \quad \textbf{end for}
6: \quad Compute the average \( \bar{v} = \frac{1}{n} \sum_{i=1}^{n} v_i \in T_{\mu_\psi}(\mathbb{S}_\infty) \).
7: \quad Update \( \mu_\psi \mapsto \cos(\epsilon \| \bar{v} \|) \mu_\psi + \sin(\epsilon \| \bar{v} \|) \frac{\bar{v}}{\| \bar{v} \|} \), for a small step size \( \epsilon > 0 \).
8: \textbf{end while}
9: Compute the mean warping function using \( \tilde{\gamma}(t) = \int_0^t \mu_\psi(s)^2 ds \).

The Algorithm 2.1 is used for separating phase and amplitude method which we call Phase-Amplitude Separation method in [41]:

From Algorithm 2.2, we get the set of aligned of SRSFs, \( \{ \tilde{q}_i \} \), the set of aligned original functions, \( \{ \tilde{f}_i \} \), and the set of optimal warping functions, \( \{ \gamma^*_i \} \). In here, \( \{ \tilde{q}_i \} \) is considered as the amplitude of \( \{ q_i \} \), \( \{ \tilde{f}_i \} \) is amplitude of \( \{ f_i \} \), and \( \{ \gamma^*_i \} \) is phase of the functions, respectively. In this way, we could focus on one of the components, phase or amplitude.

Figure 2.11 shows the example of Phase-Amplitude Separation method using Medfly data. The first panel shows the number of egg counts \( \{ f_i \} \), the second panel shows the aligned functions \( \{ \tilde{f}_i \} \), and the optimal warping functions \( \{ \gamma^*_i \} \) for Algorithm 2.2 are shown in the third panel of Figure 2.11.
Algorithm 2.2 Phase-Amplitude Separation

1: Compute SRSFs \( q_1, q_2, \ldots, q_n \) of the given \( f_1, f_2, \ldots, f_n \) and select \( \mu = q_i \), where \( i = \arg \min_{1 \leq i \leq n} ||q_i - \frac{1}{n} \sum_{j=1}^{n} q_j|| \).

2: For each \( q_i \) find the \( \gamma^*_i \) such that \( \gamma^*_i = \arg \min_{\gamma \in \Gamma} (||\mu - (q_i \circ \gamma)||) \). The solution to this optimization comes from the dynamic programming algorithm.

3: Compute the aligned SRSFs using \( \tilde{q}_i \mapsto (q_i \circ \gamma^*_i) \sqrt{\gamma^*_i} \).

4: If the increment \( \frac{1}{n} \sum_{i=1}^{n} \tilde{q}_i - \mu \) is small, then stop. Else, update the mean using \( \mu \mapsto \frac{1}{n} \sum_{i=1}^{n} \tilde{q}_i \) and return to step 2.

5: The function \( \mu \) represents a whole equivalence class of solutions and now we select the preferred element \( \mu_q \) of that orbit:

1. Compute the mean \( \gamma_\mu \) of all \( \{\gamma^*_i\} \) (using Algorithm 2.1). Then compute \( \mu_q = (\mu \circ \gamma_\mu^{-1}) \sqrt{\gamma_\mu^{-1}} \).

2. Update \( \gamma^*_i \mapsto \gamma^*_i \circ \gamma_\mu^{-1} \). Then compute the aligned SRSFs using \( \tilde{q}_i \mapsto (q_i \circ \gamma^*_i) \sqrt{\gamma^*_i} \) and aligned functions using \( \tilde{f}_i \mapsto f_i \circ \gamma^*_i \)

---

Original \( \{f_i\} \) | Amplitude \( \{\tilde{f}_i\} \) | Phase \( \{\gamma^*_i\} \)
---

Figure 2.11: Example of Phase-Amplitude Separation
2.3 Functional Regression Model

One of the fast growing subtopics in FDA is the problem of involving functional variables, either as predictors or responses or both. Morris [28] categorizes regression problems involving functional data into three types: (1) Scalar-on-Function (functional covariate, scalar response), (2) Function-on-Scalar (scalar covariate, functional response) and (3) Function-on-Function regression (functional covariate, functional response) (See below).

\[
\begin{align*}
  \mathbf{X} & \rightarrow \mathbf{Y} \\
  \mathbb{R}^d & \rightarrow \mathbb{R} \quad \text{Multiple Regression, GLM} \\
  \mathbb{R}^{d_1} & \rightarrow \mathbb{R}^{d_2} \quad \text{Multivariate Regression} \\
  (1) \mathbb{L}^2 & \rightarrow \mathbb{R} \quad \text{Scalar-on-Function Regression} \\
  (2) \mathbb{R}^d & \rightarrow \mathbb{L}^2 \quad \text{Function-on-Scalar Regression} \\
  (3) \mathbb{L}^2 & \rightarrow \mathbb{L}^2 \quad \text{Function-on-Function Regression}
\end{align*}
\]

We summarize these models in this section.

2.3.1 Functional Predictor Regression (Scalar-on-Function)

One of the most general scalar-on-function regression models is functional linear regression model when there is a linear relationship between predictors \{f_i\} and response variables \{y_i\}. Functional Linear Model (FLM) is first introduced by Ramsay & Dalzell [32] and first written in its commonly encountered form by Hastie & Mallows [21]:

\[
y_i = \alpha + \int f_i(t)\beta(t) \, dt + \epsilon_i,
\]

where \(y_i, i = 1, \ldots, n\) is a continuous response, \(f_i(t)\) a functional predictor, \(\beta(t)\) functional coefficients, \(\alpha\) intercept, and \(\epsilon_i \sim N(0, \sigma^2)\) residual errors. In here, \(y_i, \alpha, \) and \(\epsilon\) are all scalars.

2.3.2 Functional Response Regression (Function-on-Scalar)

Functional response regression involves the regression of functional responses on a set of scalar predictors [32]. Given a sample of functional responses \(y_i(t), i = 1, \ldots, n\) and scalar predictors \(f_{ij}, j = 1, \ldots, p\), a general linear functional response regression model is given by
\[ y_i(t) = \beta_0(t) + \sum_{j=1}^{p} f_{ij} \beta_j(t) + \epsilon_i(t), \]

where functional coefficient \( \beta_j(t) \) represents the partial effect of predictor \( f_j \) on the response at position \( t \).

### 2.3.3 Function-on-Function Regression

Ramsay & Dalzell [32] introduced the functional linear model:

\[ y_i(t) = \alpha(t) + \int f_i(s) \beta(s,t) \ ds + \epsilon_i(t). \]

where \( \alpha(t) \) is a functional intercept and \( \beta(s,t) \) is unconstrained surface coefficient.
CHAPTER 3
LITERATURE REVIEW OF SCALAR-ON-FUNCTION REGRESSION MODELS

In this chapter, we focus on summarizing several types of scalar-on-function regression model. Reiss et al. [35] categorized the scalar-on-function regression model into four types: 1) linear scalar-on-function regression, 2) nonlinear scalar-on-function regression, 3) generalized and extended model of linear and nonlinear scalar-on-function regression, and 4) nonparametric scalar-on-function regression. Febrero-Bande and Oviedo de la Fuente [13] summarized the scalar-on-function regression models with tecator benchmark examples using a statistical software $R$. We recapitulate these summaries in this chapter. Moreover, we summarize the estimation of the linear scalar-on-function model and address the problems of the functional linear regression model.

3.1 Linear Scalar-on-Function Regression

3.1.1 Functional Linear Model with basis representation

To recap, Functional Linear Model (FLM) can be expressed as follow:

$$ y_i = \alpha + \int f_i(t)\beta(t) \, dt + \epsilon_i, $$

where $y_i, i = 1, \ldots, n$ is a continuous response, $f_i(t)$ a functional predictor, $\beta(t)$ functional coefficients which can be expressed as $\beta(t) = \sum_{j=1}^{J} c_j b_j(t)$ for a fixed basis $B = \{b_j, j = 1, 2, \cdots \}$ with coefficients $c_j$ respect to the basis, $\alpha$ intercept, and $\epsilon_i \sim N(0, \sigma^2)$ residual errors. In here, $y_i, \alpha, \text{ and } \epsilon$ are all scalars.

Estimation of the parameters that we use is Ordinary Least Squares (OLS) method. OLS is the method to estimate the parameters of linear regression model when both predictors and responses are scalars. Based on this idea, Ramsay [32] proposed the same procedure as the multivariate linear regression case but the predictors are changed to functions from scalars. We decide the coefficients of the expansion $c_j$ by minimizing the least squares criterion based on functional simple linear regression from equation 3.1.1.
\[ \text{SSE}(y_i|\beta) = \sum_{i}^{n} \left[ y_i - \alpha - \int_{0}^{T} f_i(t)\beta(t) \, dt \right]^2 \]

In here, one problem is raised since for any finite \( n \), \( \beta(t) \) is a full function that there are infinitely many solutions for \( \beta(t) \) without imposing any restrictions on this. It is an element of an infinite-dimensional space while its specification for any \( n \) finite dimensional. Ramsay and Silverman [33] proposed two solutions to manage this problem: (1) represent \( \beta(t) \) using \( p \) basis functions in which \( p \) is kept large to allow desired variations of \( \beta(t) \), and (2) add a smoothing penalty term to the cost function (SSE) which chooses a smooth solution by finding an optimal balance between SSE and the smoothing penalty.

\[ \text{PEN} \text{SSE}_\lambda = \sum_{i=1}^{n} \left[ y_i - \alpha - \int f_i(t)\beta(t) \, dt \right]^2 + \lambda R(\beta(t)) \]  
(3.1.2)

Eqn. 3.1.2 is the estimation problem of functional linear regression model adding regularization \( R(\beta(t)) = \int [D^2\beta(t)]^2 \, dt \) where \( \lambda \) is a smoothing parameter.

One of the common ways to choose smoothing parameters is using leave-one-out Cross-Validation (CV) score function:

\[ CV(\lambda) = \frac{1}{n} \sum \left( \frac{y_i - \hat{y}_i}{1 - S_{ii}} \right)^2 \]

where \( S_{ii} \) is the \( i \)th diagonal element of hat matrix \( S \), \( f = Sy \). \( \lambda \) is chosen as the minimizer of \( CV(\lambda) \).

We implemented the example of functional linear regression model. In Figure 3.1, the first panel (a) shows the functional predictors of tecator dataset and the second panel (b) shows the response variables, fat contents, and the last panel (c) shows the result of predictions of the functional linear model. We use this dataset to predict the fat contents from the absorbance as a functional covariate, \( f_i(t) \). We set the training set for 165 curves to fit the model. The last 50 records will be used to check the predictions. By fitting the data to FLM in 3.1.1, we would get Sum of Squared Error, \( \text{SSE} = 395.8719 \) on the test set by choosing the optimal smoothing parameter.
3.1.2 Functional Linear Model with Functional PC Basis

In section 2.1.4, we summarized the concept of Functional Principal Component Analysis. Based on this idea, Cardot et al. [8] proposed the functional predictor model using Functional Principal Component (FPC) basis. The key idea of this model is that we use data-driven basis which is the eigenbasis associated with the covariance function $\text{cov}(s,t)$. That is, we use the orthonormal set of FPC $\phi_1(t), \phi_2(t), \ldots$ such that for each $j$, $\int \text{cov}(s,t)\phi_j(s)\,ds = \lambda_j \phi_j(t)$ for eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$. Then each of the functional covariates can be expressed by its truncated Karhunen - Loéve expansion,
\[ f_i(t) = \sum_{l=1}^{L} \xi_{il} \phi_l(t) \]

and the coefficient regression \( \beta(t) = \sum_{l'=1}^{L} \phi_{l'}(t) \) using the same basis. Then the Eqn. 3.1.1 becomes

\[ \sum_{l=1}^{L} \sum_{l'=1}^{L} \xi_{il} \beta_{l'}(t) \int \phi_l(t) \phi_{l'}(t) \, dt = \sum_{l=1}^{L} \xi_{il} \beta_l \]

since \( \int \phi_l(t)^2 \, dt = 1 \) and \( \int \phi_l(t) \phi_{l'}(t) \, dt = 0 \) \((l < l')\). Hence, the regression model can be expressed as

\[ y_i = \alpha + \sum_{l=1}^{L} \xi_{il} \beta_l + \epsilon_i \quad (3.1.3) \]

**Asymptotic Theory of the estimation, \( \beta(t) \) [8].** Let’s assume the following condition:

\[ \hat{\lambda}_1 > \hat{\lambda}_2 > \cdots > \hat{\lambda}_k > 0 \quad \text{a.s..} \quad (3.1.4) \]

which ensures almost surely that \( K = U \sum V^T \) is regular and its eigenvectors are identifiable. Then define

\[ a_k = \begin{cases} 
\frac{2\sqrt{2}}{\lambda_1 - \lambda_2} & \text{if } k = 1 \\
\frac{2\sqrt{2}}{\min(\lambda_{k-1} - \lambda_k, \lambda_k - \lambda_{k+1})} & \text{if } k \neq 1 
\end{cases} \]

**Theorem 1.** Suppose that 3.1.4 and the following hypotheses are satisfied:

1. \( \lambda_1 > \lambda_2 > \cdots > 0 \),
2. \( E||f||^4 < +\infty \)
3. \( \lim_{n \to +\infty} n \lambda_{k_n}^4 = +\infty \)
4. \( \lim_{n \to +\infty} \frac{n \lambda_{k_n}^2}{(\sum_{k=1}^{k_n} a_k)^2} = +\infty \)

Then,

\[ ||\hat{\beta}_{(1:k_n)} - \beta|| \to 0 \quad \text{as } n \to +\infty \quad \text{in probability} \]
Theorem 2. Suppose that 3.1.4 and 1 in Theorem 1 and the following hypotheses are satisfied:

1. \[ ||f|| \leq c_1 \text{ a.s.}, \]

2. \[ ||\epsilon|| \leq c_2 \text{ a.s.}, \]

3. \[ \lim_{n \to +\infty} \frac{n\lambda_k^4}{\log n} = +\infty \]

4. \[ \lim_{n \to +\infty} \frac{n\lambda_k^2}{\left(\sum_{k=1}^{n} a_k\right)^2 \log n} = +\infty \]

Then,

\[ ||\beta_{(1:k_n)} - \beta|| \to 0 \text{ as } n \to +\infty \text{ a.s.} \]

3.2 Nonlinear Scalar-on-Function Regression

3.2.1 Single-Index Model

A Single-Index Model (SIM) is the extended functional linear model by allowing the function \( h(\cdot) \) to be any smooth function defined on the real line [3, 11, 35, 40],

\[
y_i = h\left( \int f_i(t) \beta(t) \, dt \right) + \epsilon_i, \quad i = 1, \ldots, n
\]  

(3.2.1)

where \( h(\cdot) \) is the link function. Both coefficient function \( \beta \) and the link function \( h \) are estimated in an iterative way. Estimation of \( h \) can be accomplished by the estimation method in [12]. This process is iterated until convergence. This single-index model looks same as generalized functional linear model but in this case, we assume not only \( \beta \) but also a link function \( h \) are unknown.

Figure 3.2 shows the example of SIM on functional data. The first panel shows the original data and the second panel shows the corresponding response variables. The third panel displays the actual responses versus predicted responses from: functional linear model and single-index model. By applying SIM, the predictions performed very well compared to FLM.

Figure 3.3 shows the link function \( h \) on the scatter plot of \( y_0 \) versus \( \hat{y} \) on the training set. In this figure, \( x \)-axis shows the estimated responses \( \hat{y} \) based on the estimated coefficient function, \( \hat{\beta}(t) \) on the training set. \( y \)-axis shows the true responses \( y_0 \) on the training set. A quadratic polynomial function fits two variables (\( \hat{y} \) vs. \( y_0 \)) well and this gives better performance on predictions as shown in the third panel of the Figure 3.2.
3.2.2 Multiple-Index Model

James and Silverman [33], Chen et al. [10] and Ferraty et al. [15] developed a model by allowing multiple linear functionals of the predictors via the Multiple-Index Model (MIM), it can be expressed as the extension of Single-Index Model in 3.2.1:

\[
y_i = \sum_{j=1}^{J} h_j \left( \int f_i(t) \beta_j(t) \, dt \right) + \epsilon_i
\]  

(3.2.2)

which extends projection pursuit regression to the functional predictor case.
3.3 Generalization and Extensions

3.3.1 Including Scalar Covariates

Aneiros-Perez and Vieu [5, 6] and Aneiros et al. [4] proposed semi-functional partial linear models of the form by adding scalar covariates in the FLM:

\[ y_i = z_i^T \alpha + m(f_i) + \epsilon_i \]  \hspace{1cm} (3.3.1)

which includes linear effects of scalar covariates \( z_i \) and effects of functional predictors \( f_i \). \( z_i \) can be estimated by using Weighted Least Squares (WLS) and \( f_i \) can be estimated nonparametrically using Nadaraya-Watson (NW) [30] weights.

3.3.2 Multiple Functional Predictors

The multiple functional regression model considers the situation in which the \( i \)-th observation includes multiple functional predictors, \( f_{i,1}, \ldots, f_{i,R} \), with different domains, \( I_1, \ldots, I_R \).

\[ y_i = \alpha + \sum_{r=1}^{R} \int_{I_r} f_{ir}(t) \beta_r \, dt + \epsilon_i. \]  \hspace{1cm} (3.3.2)

This model is just the extended version of FLM to the multiple functional regression model. Zhu et al. [45], Gertheiss et al. [18] and Lian [24] have been proposed Penalized or fully Bayesian approaches to selecting among candidate functional predictors.
3.3.3 Generalized Functional Linear Regression (GLM)

Many papers proposed the method that have been generalized to allow responses with exponential family distributions, including both linear (Marx & Eilers [26]; James [22]; Müller & Stadtmüller [29]; Reiss & Ogden [34]; Goldsmith et al. [20]; Aguilera-Morillo et al. [1] and nonlinear (James & Silverman [23]; McLean et al. [27]) models. For a single functional predictor and no scalar covariates, the functional generalized linear model can be expressed as:

\[ y_i = g^{-1}(\eta) + \epsilon_i \]  (3.3.3)

where \( y_i \) follows an exponential family with mean \( E(y) = \mu \); monotone link function \( g(\cdot) \); linear predictors \( \eta = \alpha + \int f_i(t)\beta_i(t) \, dt \) for \( i = 1, \ldots, n \); \( \epsilon_i \sim N(0, \sigma^2) \).

3.4 Nonparametric Scalar-on-Function Regression

Nonparametric regression model (NP) is studied by Ferraty and Vieu [14] which can be expressed as follows:

\[ y_i = r(f_i(t)) + \epsilon_i \]  (3.4.1)

where the unknown smooth function \( r \) is estimated by the functional Nadaraya-Watson (NW) estimator [30].

\[ \hat{r}(f) = \frac{\sum_{i=1}^{n} y_i K(v^{-1}d(f, f_i))}{\sum_{i=1}^{n} K(v^{-1}d(f, f_i))} \]

where \( K(\cdot) \) is a kernel function, \( v \) is a smoothing parameter, and \( d(\cdot, \cdot) \) is a metric or semi-metric. In here, general kernel function is a Gaussian kernel function in \( \mathbb{L}^2 \) space. Gaussian kernel function can be expressed as \( K(\mu) = \frac{1}{\sqrt{2\pi}^2} e^{-\frac{1}{2} \mu^2} \). \( \mathbb{L}^2 \) norm distance is defined as \( d(f, f_i) = ||f - f_i|| \) and \( ||f|| = \sqrt{\langle f, f \rangle} = \sqrt{\int f(t) f(t) \, dt} \).

For the estimation and the prediction problems, we use functional kernel estimator with automatic bandwidth selection [14] which we compute the quantity:

\[ G(f) = \frac{\sum_{i=1}^{n} y_i K(d(f_i, f)/v^*)}{\sum_{i=1}^{n} K(d(f_i, f)/v^*)}, \]  (3.4.2)
where \((f_i, y_i)\) for \(i = 1, 2, \ldots, n\) are the observed pairs and \(v^*\) is the optimal bandwidth by a cross-validation procedure:

\[
v^* = \arg \min_v \sum_{i=1}^{n} (y_i - G_{(-i)}(f_i))^2,
\]

with

\[
G_{(-i)}(f) = \frac{\sum_{j=1, j \neq i}^{n} y_j K(d(f_j, f))/v}{\sum_{j=1, j \neq i}^{n} K(d(f_j, f))/v}.
\]

Figure 3.4: Example of Nonparametric Regression

(a) \(\{f_i\}\)  
(b) \(y_i\)  
(c) \(y_i\) (black) vs. \(\hat{y}_i\) (blue)
Figure 3.4 shows the first derivative of the spectrometric curves from *tecator* data. The first panel shows the spectrometric curves and the second panel shows the responses, fat contents. The third panel of the figure shows the prediction using kernel regression model (blue) versus actual responses (black). The SSE using the nonparametric functional regression model on *tecator* data is 191.7893.
CHAPTER 4

ELASTIC FUNCTIONAL REGRESSION MODEL

In this chapter, we present a novel solution that is motivated by the use of the Fisher-Rao metric in functional data alignment [38, 39]. In fact, this elastic functional data analysis (EFDA) framework suggests several ideas, only one of which is pursued in this paper. EFDA is based on replacing the use of $L^2$ inner product and the $L^2$ distance between functions by the Fisher-Rao Riemannian metric and the Fisher-Rao distance between these functions. The latter provides better mathematical and numerical properties, and indeed leads to a superior registration between functions. The challenge in using the original Fisher-Rao metric comes from its complicated expression, but that is overcome using the square root velocity function (SRVF), as described in Srivastava et al. [39]. One works with the SRVFs $q_i$s instead of the predictors $f_i$s and the Fisher-Rao metric becomes the standard $L^2$ metric. In this framework, the time warplings of $q_i$s, given by $(q_i \circ \gamma_i)\sqrt{\dot{\gamma}_i}$, are norm preserving. That is, $\|q_i\| = \|(q_i \circ \gamma_i)\sqrt{\dot{\gamma}_i}\|$ for $q_i \in L^2$ and all warpings $\gamma_i$, and thus pinching is no longer possible. This, in turn, suggests two ways of fixing the problem in Eqn. 1.3.1:

1. **Use Fisher-Rao Metric and SRVF Representation:** One can compute SRVFs of the given predictors, and then simply replace the term $\sup_{\gamma_i} \langle f_i \circ \gamma_i, \beta \rangle$ in Eqn. 1.3.1 by the Fisher-Rao based inner product: $\sup_{\gamma_i} \langle (q_i \circ \gamma_i)\sqrt{\dot{\gamma}_i}, \beta \rangle$. Since any warping of $q_i$ in this way preserves its norm, the pinching problem is avoided.

2. **Use a Norm-Preserve Warping and $L^2$ Metric:** The other option is to change the basic definition of the warping itself, from the classical composition $(f_i \circ \gamma_i)$ to the norm-preserving action $((f_i \circ \gamma_i)\sqrt{\dot{\gamma}_i})$. In the new definition, a warping changes both the location and the height of a function value. This suggests using $\sup_{\gamma_i} \langle (f_i \circ \gamma_i)\sqrt{\dot{\gamma}_i}, \beta \rangle$ in Eqn. 1.3.1. That is, we assume that $f_i$s are already in SRVF space and use them as such. This process may be useful when the data is noisy and a further SRVF transformation enhances this noise due to the presence of a derivative. By treating $f_i$s as SRVFs, one obtains the nice properties of this framework and avoids enhancing the noise. On the other hand, this warping is different from the typical warping $f \circ \gamma_i$ used in the alignment literature.

Each of these models avoid the pinching effect, and have their own pros and cons. Ultimately, the choice of a model depends on the nature of the data and the goals of the application. The response
variable in both these models is invariant to the action of the time warping group on the predictor functions.

4.1 Proposed Elastic Framework

In this section, we layout an elastic functional regression model for scalar-on-function problem with the property that the response variable is invariant to the phase component of the predictor [2]. This framework is based on ideas used previously for alignment of functional data, or phase-amplitude separation, using the Fisher-Rao metric and the SRVF representation of functions. We start by briefly introducing those concepts and refer the reader to [39] for details.

As mentioned earlier, the use of $L^2$ inner-product or $L^2$ norm for alignment of functions leads to a well-known problem called the pinching effect. While some papers avoid this problem using a combination of external penalties and search space reductions, a superior solution comes from using an elastic Riemannian metric with appropriate invariance properties. This metric, called the Fisher-Rao metric, avoids the pinching effect without any external constraint and results in better alignment results. Let $f$ be a real-valued function on the interval $[0,1]$ (with appropriate smoothness) and let $\mathcal{F}$ denote the set of all such functions. For the purpose of alignment, one represents it using a square-root velocity function (SRVF) defined as $q(t) = \dot{f}(t)/\sqrt{|\dot{f}(t)|}$ or $q(t) = \text{sign}(\dot{f}(t))\sqrt{|\dot{f}(t)|}$. These two expressions are algebraically equivalent. One of the advantages of using SRVF is that under the transformation $f \mapsto q$, a complicated Fisher-Rao Riemannian metric and the Fisher-Rao distance into much simpler expressions. That is:

$$\langle\langle f_1, f_2 \rangle\rangle_{FR} = \langle q_1, q_2 \rangle, \quad \text{and} \quad d_{FR}(f_1, f_2) = \|q_1 - q_2\|.$$

If we warp a function $f$ by a time warping $\gamma$, i.e., map $f \mapsto (f \circ \gamma)$, then its SRVF changes by $q \mapsto (q \circ \gamma)\sqrt{\gamma}$. The latter is often denoted by $(q, \gamma)$. The invariance property of the Fisher-Rao metric implies that for any $q_1, q_2 \in L^2$ and $\gamma \in \Gamma$, we have: $\|(q_1, \gamma) - (q_2, \gamma)\| = \|q_1 - q_2\|$. In other words, the action of $\Gamma$ on $L^2$ is by isometries. A special case of this equation is that $\|(q, \gamma)\| = \|q\|$ for all $q$ and $\gamma$. Thus, this action preserves the $L^2$ norm of the SRVF and, therefore, avoids any pinching effect.

This framework motivates several solutions for avoiding the pinching problem associated with the inner-product term in Eqn. 1.3.1. While one can work with the SRVFs of the given predictor
functions, they are prone to noise in the original data due to the involvement of a time derivative in the definition of SRVF. In case the original data is noisy, this noise gets enhanced by the derivative. As a workaround to this problem, we treat the given predictor functions to be in the SRVF space already. That is, we assume the action of warping $\gamma_i$ on an $f_i$ is given by $(f_i \circ \gamma_i)\sqrt{\dot{\gamma}_i}$ and not $f_i \circ \gamma_i$. With this action, we have that $\|(f_i, \gamma_i)\| = \|(f_i \circ \gamma_i)\sqrt{\dot{\gamma}_i}\| = \|f_i\|$. 

Based on this argument, the inner-product term in Eqn. 1.3.1 can be replace by the term: $\sup_{\gamma_i} \langle \beta, (f_i, \gamma_i) \rangle$. This is a scalar quantity and represents a modified linear relationship between the predictor and the response. One can impose a more sophisticated single-index model on top of this construction as follows. Such single-index models have been used commonly in conjunction with the Functional Linear Model (FLM), see e.g. [3, 11, 35, 40]. Let $h : \mathbb{R} \rightarrow \mathbb{R}$ be any smooth function defined on the real line, and define the model:

$$y_i = h\left(\sup_{\gamma_i} \langle \beta, (f_i, \gamma_i) \rangle\right) + \epsilon_i, i = 1, \ldots, n$$

(4.1.1)

The inclusion of $h$ allows the model to capture nonlinear relationships between the predictor and the response variables. This single-index model (SIM) is generally the same as a generalized functional linear model (GFLM), but in SIM the link function $h$ is unknown.

### 4.2 Parameter Estimation

Next we consider the problem of estimating model parameters under the model given in Eqn. 4.1.1. The list of parameters, include the link function, $h$, and the coefficient of regression $\beta$. We take an iterative approach given in [12], where one updates the estimates of $h$ or $\beta$ while keeping the other fixed. Thus, we first focus on the techniques for updating 1) the estimation of $\beta$ and 2) the estimation of single-index model $h$ separately, and then we propose an iterative process for joint-estimation.

#### 4.2.1 Estimation of $\beta$ Keeping $h$ Fixed

Given a set of observations $\{(f_i, y_i)\}$, the goal here is to solve for the coefficient of regression $\beta$, while keeping $h$ fixed, using maximum-likelihood estimation. In order to reduce the search space to a finite-dimensional set, we will assume that $\beta \in \{\sum_{j=1}^{J} c_j b_j | c_j \in \mathbb{R}\}$ for a fixed basis set
\[ B = \{ b_j, j = 1, 2, \ldots \} \text{ of } L^2([0,1], \mathbb{R}). \] The estimation problem is now given by:

\[
\hat{c} = \arg \min_{c \in \mathbb{R}^J} H(c), \text{ where } H : \mathbb{R}^d \to \mathbb{R},
\]

\[
H(c) = \left( \sum_{i=1}^J (y_i - h(\sup_{\gamma_i} \langle \sum_{j=1}^J c_j b_j, (f_i, \gamma_i) \rangle))^2 \right). 
\]

This optimization can be summarized into an algorithm as follows.

**Algorithm 4.1 Estimation of \( \beta \) keeping \( h \) fixed**

1: Initialization Step. Choose an initial \( c \in \mathbb{R}^J \) and compute \( \beta(t) = \sum_{j=1}^J c_j b_j(t) \).
2: Find \( \{ \gamma_i^* \} \) using Dynamic Programming, \( \gamma_i^* = \arg \min_{\gamma_i} \| \beta - (f_i, \gamma_i) \|_2 \), for each \( i = 1, \ldots, n \).
3: Compute the aligned functions \( \tilde{f}_i \leftarrow (f_i \circ \gamma_i^*) \sqrt{\gamma_i^*} \).
4: Use an optimization code (such as `fminunc` in `matlab`) to find \( \hat{c} \) that minimize the cost function \( H \).
5: Update \( \beta(t) \mapsto \sum_{j=1}^J \hat{c}_j b_j(t) \). If the \( |H(\hat{c})| \) is small, then stop. Else return to step 2.

**Profile of Objective Function**: We study Algorithm 4.1 using a simple experiment. In the experiment, we are interested in the feasibility of estimation of regression coefficients \( c \in \mathbb{R}^J \) from a given data set. Here we use the first \( J/2 \) harmonics of Fourier basis (both sines and cosines) to form a fixed basis \( B \). For simulating the predictor functions \( \{ f_i, i = 1, 2, \ldots, n \} \), we use the same basis function. Using \( J = 2 \), we set \( c_0 = [1 1] \) and form the coefficient function \( [\beta_0] = F(c_0) \).

Finally, we form the training data using the model in Equation 4.1.1, we also add observation noise (i.e. \( \epsilon_i \sim N(0, 0.05^2) \)). Given the training data, we study the nature of the objective function \( H(c) \) as a function around the point \( c_0 \).

To summarize, we have this initial setting as follows:

- \( J = 2k \) and \( k = 1 \)
- \( SS = \sqrt{2\sin(2\pi kt)} \)
- \( CS = \sqrt{2\cos(2\pi kt)} \)
- \( c_0 = [c(1) \ c(2)] = [1 \ 1] \)
- \( \beta_0(t) = c(1)SS + c(2)CS \)
- \( \epsilon_i \sim N(0, 0.05^2) \)
4.2.2 Estimation of a Link Function $h$ Keeping $\beta$ Fixed

Next we consider the problem of estimating the link function $h$ given the data and the estimated $\beta$. The reason for introducing this single-index model is to capture nonlinear relationship between

$\bullet f_i(t) = \text{rand}(i) \times \beta_0(t)$

Figure 4.1 and Figure 4.2 show the illustration of the estimation in Algorithm 4.1. Figure 4.1 shows 10 predictor functions and warped functions used in this experiment and Figure 4.2 shows the profile of $H$ as a function of $C$ around $c_0$. As the mesh plot shows, there is a unique local minimum at $c_0$. 

Figure 4.1: The left panel shows a number of predictor functions $\{f_i\}$, the right panel shows the alignment of these functions to the regression coefficient

Figure 4.2: The profile of $H$ over a 2D domain.
the predicted responses and observed responses. While there are many nonparametric estimators for handling $h$, we keep the model simple by restricting to lower-order polynomials. Hence, this link function can either be linear, quadratic, cubic, and so on: $h(x) = ax + b$, $h(x) = ax^2 + bx + c$, and $h(x) = ax^3 + bx^2 + cx + d$, etc. In our experiment, we use the first three polynomial functions for $h$.

In terms of estimating $h$, we use the current estimate $\hat{\beta}$ to predict the responses according to: $\hat{y}_{i}^{(train)} = \sup_{\hat{\gamma}_i} \langle \hat{\beta}, (\hat{f}_{i}^{(train)}, \hat{\gamma}_i) \rangle$. Then, we fit a polynomial function $h$ between the predicted responses $\hat{y}_{i}^{(train)}$ and the observed responses $y_{i}^{(train)}$ using the least squares error criterion.

The full parameter estimation procedure is as presented in Algorithm 4.2.

Algorithm 4.2 Elastic Scalar-on-Function Regression Model
1: Initialize $h$ as the identity function ($h(x) = x$).
2: Given $h$, use Algorithm 4.1 to estimate $\hat{\beta}$.
3: For a given $\hat{\beta}$, fit the single-index model using the least squares criterion and update $h$.
4: If the stopping criterion is met, then stop. Else, return to step 2.

4.3 Prediction of Elastic Regression Model

One of the main goals of a regression model is to predict values of the response variable for the future predictor observations. We describe that prediction process in the elastic functional regression model. This process involves aligning the predictors to the coefficient $\hat{\beta}$ using dynamic programming algorithm. For a given $f^{(test)}$, the predicted value of $y$ is:

$$\hat{y} = \hat{h}\left(\sup_{\gamma_i} \left( \sum_{j=1}^{J} \hat{c}_j b_j, (f^{(test)}, \gamma_i) \right) \right).$$

(4.3.1)

We will use this process to evaluate the prediction performance of our proposed model, and other current models, using both simulated data in Chapter 5 and real data in Chapter 6.
CHAPTER 5
SIMULATION STUDY

In this chapter, we compare our elastic functional regression model with other scalar-on-function regression models which are Functional Linear Model (FLM), Pre-Aligned Functional Linear Model (PAFLM), Nonparametric regression model (NP) with Gaussian kernel function; with two different semi-metrics: \( L^2 \) distance and elastic distance. FLM is the model which assumes that all functions are well-aligned that there are no phase variations between the functions. PAFLM is the model when we pre-align the functions by using any statistical tools, then we do the same estimation as FLM. Nonparametric regression is the model that does not require any predetermined form but driven from the observed data.

5.1 Initial Setup

Similar to the profile of objective functions in section 4.2.1, we simulate \( n = 20 \) observations using the model stated in Equation 4.1.1. For the predictors, we use a truncated Fourier basis and random coefficients to form the functions \( f_i \). Given these functions, we perturb them using random time warpings \( \{\gamma_i\} \) to obtain the predictors \( \{(f_i, \gamma_i)\} \). We also simulate the coefficient function \( \beta \) using the same Fourier basis but with a fixed coefficient vector \( c_0 = [1, 1, ..., 1] \). We plug these quantities in the model and add independent observation noise, \( \epsilon \sim N(0, 0.01^2) \), to obtain the responses \( \{y_i\} \). These simulated quantities are shown in Figure 5.1. In the following evaluations, we randomly divide this set in half training and half testing. In summary, the actual model of the regression model that we constructed is:

\[
y_i = \left( \sup_{\gamma_i} \langle \beta_0, (f_i, \gamma_i) \rangle \right)^2 + \epsilon_i, \ i = 1, \ldots, 20
\]

where \( f_i \) is the functional predictor using Fourier basis and \( \beta_0 \) is the functional coefficient using same Fourier basis, \( \gamma_i \) is a warping function, and measurement error, \( \epsilon_i \sim N(0, 0.01^2) \).
Figure 5.1: Simulated Data

The generated functions are shown in Figure 5.1a and observed predictor functions are shown in Figure 5.1b. The response variables are shown in Figure 5.1c. Red curves/dots correspond to the variables on the training set and blue curves/dots correspond the variables on the test set in Figure 5.1.

Figure 5.2 and Figure 5.3 show how the functions are aligned with respect to the actual/true regression coefficient $\beta_0$ to construct $y_i$ using Equation 4.1.1 on the training set and the test set, respectively. The original functions $\{f_i\}$ (black dashed lines), aligned functions with respect to $\beta_0$, $\{\sup_{\gamma_i} \langle \beta_0, (f_i, \gamma_i) \rangle\}$ (red/blue solid lines), actual regression coefficient $\beta_0$ (green solid line) are shown in Figure 5.2a and Figure 5.3a. The corresponding time warping functions $\{\gamma_i\}$ are shown in Figure 5.2b and Figure 5.3b.
For the estimation and the prediction of the elastic functional regression model, we focus on joint-estimation of the coefficient of regression $\beta$ and a link function $h$. We use three different basis functions to fit the model: 1) the same basis with the same number of elements as constructing variables in initial step (Fourier basis with two elements), 2) the same basis with the different number of basis (Fourier basis with 4 elements), and 3) the different basis (B-spline basis with 4 elements). We estimate the parameters using Algorithm 4.2. The reason for using three different basis functions for the estimation problem is to evaluate the estimation accuracy of our model no matter what basis function we use. We also use three different link functions: linear, quadratic,
and cubic polynomial functions for the estimation and prediction of the model.

5.2 Using the Same Basis with the Same Number of Elements

In this section, we use the same basis and the same number of elements (Fourier basis with two elements) but the different initial coefficients as a coefficient of regression $\beta$.

5.2.1 Estimation

![Figure 5.4: Warped functions and warping functions on the training set](image)

(a) $\hat{\beta}$ vs. $\beta_0$ vs. warped $\{f_i\}$

(b) $\{\gamma_i^*\}$

Figure 5.4: Warped functions and warping functions on the training set

![Figure 5.5: Warped functions and warping functions on the test set](image)

(a) $\hat{\beta}$ vs. $\beta_0$ vs. warped $\{f_i\}$

(b) $\{\gamma_i^*\}$

Figure 5.5: Warped functions and warping functions on the test set
Figure 5.4 and Figure 5.5 show the analysis of estimation problem using the same basis with the same number of elements, Fourier basis with $J = 2$ on the training set and the test set, respectively. The estimated coefficient of regression $\hat{\beta}$ (solid red line) versus actual regression coefficient $\beta_0$ (solid green line) versus warped functions $\{(f_i, \gamma_i)\}$ (dashed black lines) on the training set and the test set are shown in Figure 5.4a and Figure 5.5a, respectively. The functions are warped which are similar to Figure 5.2a on the training set and Figure 5.3a on the test set. The corresponding warping functions on the training set and the test set are shown in Figure 5.4b and Figure 5.5b, respectively. These warping functions also can be compared with Figure 5.2b and Figure 5.3b and they are similar to the warpings in Figure 5.2b and Figure 5.3b on the training set and the test set, respectively.

Figure 5.6: Evolution of Cost Function $H$ for each link function $h$
The evolution of the cost function $H$ where 

$$H(c) = \left( \sum_{i=1}^{J}(y_i - h(\sup \gamma_i \left\langle \sum_{j=1}^{J} c_j b_j, f_i, \gamma_i \right\rangle))^2 \right)$$

for Algorithm 4.2 for each link function: linear ($h(x) = ax + b$), quadratic ($h(x) = ax^2 + bx + c$), and cubic ($h(x) = ax^3 + bx^2 + cx + d$) polynomial functions are shown in Figure 5.6a, Figure 5.6b, and Figure 5.6c, respectively. As the results, the evolution for the Algorithm 4.2 using the same basis with the same number of elements or Fourier basis with two elements converged in less than 25 iterations. These plots show that the cost function $H$ goes down in all cases and the optimization algorithm provides at least local solutions.

5.2.2 Prediction

We predict the response variables on the test set using Equation 4.3.1. For comparison, we also implemented Functional Linear Model (FLM) using Equation 3.1.1 and Pre-Aligned Functional Linear Model (PAFLM). PAFLM is the model which pre-aligns the training data and the test data using one of several existing alignment algorithms and then performs standard FLM. For example, the registration can be implemented by using Square-Root Velocity Functions (SRVFs) and template function or karcher mean from the “Complete Alignment Algorithm” [39]. This alignment is naturally suboptimal from the perspective of regression, since the response variable is not used in phase separation.

Figure 5.7 shows the prediction results from our elastic functional regression model, FLM, and PAFLM. We use three different link functions on elastic functional regression model: linear, quadratic, and cubic polynomial function and they are shown in Figure 5.7a, Figure 5.7b, and Figure 5.7c, respectively. Each figure presents the true responses ($True$) versus predictions from FLM ($FLM$), PAFLM ($PAFLM$), and elastic functional regression model using three different link function ($Linear$ or $Quadratic$ or $Cubic$).

To evaluate the performance of our model, we compute Mean Squared Prediction Error (MSPE) to compare the prediction accuracy of the elastic functional regression model with two traditional scalar-on-function regression models. MSPE can be defined as:

$$MSPE = \frac{1}{m} \sum_{i=1}^{m} \left( y_i - \hat{y}_i \right)^2$$

where $m$ is the number of observed functions on the test set, $y_i$ is the true response variable on the test set, and $\hat{y}_i$ is the prediction of the functional regression model.
Prediction of elastic functional regression model was performed as was described in Equation 4.3.1.

Table 5.1: MSPE of each method using the same basis with the same number of elements

<table>
<thead>
<tr>
<th></th>
<th>FLM</th>
<th>PAFLM</th>
<th>$h$: Linear</th>
<th>$h$: Quadratic</th>
<th>$h$: Cubic</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSPE</td>
<td>4.8866</td>
<td>21.948</td>
<td>0.1693</td>
<td>0.1603</td>
<td>0.1307</td>
</tr>
</tbody>
</table>

Table 5.1 shows that predictions of elastic functional regression model performed well, especially using a cubic polynomial function as a link function $h$ compared to the other traditional regression models.
5.3 Using the Same Basis with the Different Number of Elements

In Section 5.2, we use the same basis with the different number of elements to see how the different number of elements with same basis function (Fourier basis with four elements) would perform in estimation and prediction using Algorithm 4.2 and Equation 4.3.1.

5.3.1 Estimation

![Figure 5.8: Warped functions and warping functions on the training set](image1)

(a) \( \hat{\beta} \) vs. \( \beta_0 \) vs. warped \( \{f_i\} \)  

(b) \( \{\gamma^*_i\} \)

Figure 5.8: Warped functions and warping functions on the training set

![Figure 5.9: Warped functions and warping functions on the test set](image2)

(a) \( \hat{\beta} \) vs. \( \beta_0 \) vs. warped \( \{f_i\} \)  

(b) \( \{\gamma^*_i\} \)

Figure 5.9: Warped functions and warping functions on the test set
Figure 5.8 and Figure 5.9 show the analysis of estimation problem using the same basis with the different number of elements, Fourier basis with 4 elements or Fourier with $J = 4$ on the training set and the test set, respectively. The estimated coefficient of regression $\hat{\beta}$ (solid red/blue line) versus actual coefficient of regression $\beta_0$ (solid green line) versus warped functions $\{(f_i, \gamma_i)\}$ (dashed black lines) on the training set and the test set are shown in Figure 5.8a and Figure 5.9a, respectively. The original functions are aligned with respect to $\hat{\beta}$ which are similar to Figure 5.2a on the training set and Figure 5.3a on the test set. The corresponding warping functions on the training set and the test set are shown in Figure 5.8b and Figure 5.9b, respectively. These warping functions also can be compared to the warpings in Figure 5.2b and Figure 5.3b.

![Figure 5.8a](image1.png) ![Figure 5.9a](image2.png)

(a) $h$: Linear  
(b) $h$: Quadratic  
(c) $h$: Cubic

Figure 5.10: Evolution of Cost Function $H$ for each link function $h$

The evolution of the cost function $H$ for Algorithm 4.2 for each link function: linear, quadratic,
and cubic polynomial functions are shown in Figure 5.10a, Figure 5.10b, and Figure 5.10c, respectively. As the results, the cost functions are decreasing and they converged to at some local points. This process takes less than 20 iterations.

5.3.2 Prediction

Similar to the previous section, we predict the response variables on the test set using Equation 4.3.1.

![Prediction Plots of Regression Models](image)

Figure 5.11: Prediction Plots of Regression Models

Figure 5.11 shows the prediction results from elastic functional regression model, FLM, and PAFLM. Predictions from elastic functional regression model with different link functions: linear, quadratic, and cubic polynomial functions are shown in Figure 5.11a, Figure 5.11b, and Figure
5.11c, respectively. Similar to Section 5.2.2, we compute MSPE to compare the prediction accuracy of the elastic functional regression model versus the other two traditional scalar-on-function regression models.

Table 5.2: MSPE of each method using the same number of basis with the different number of elements

<table>
<thead>
<tr>
<th></th>
<th>FLM</th>
<th>PAFLM</th>
<th>$h$: Linear</th>
<th>$h$: Quadratic</th>
<th>$h$: Cubic</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSPE</td>
<td>10.782</td>
<td>113.91</td>
<td>0.9266</td>
<td><strong>0.1201</strong></td>
<td>0.3961</td>
</tr>
</tbody>
</table>

Table 5.2 shows that predictions from elastic functional regression model are more accurate, especially using a quadratic polynomial function for a link function $h$ compared to the other two traditional models.

5.4 Using the Different Basis

In this section, we use the different basis function (B-spline basis with 4 elements) using Algorithm 4.2 in order to see the effectiveness of estimation and prediction by using a totally different basis function on simulated data.

5.4.1 Estimation

![Figure 5.12](image)

(a) $\hat{\beta}$ vs. $\beta_0$ vs. warped $\{f_i\}$

(b) $\{\gamma_i^*\}$

Figure 5.12: Warped functions and warping functions on the training set
Figure 5.13: Warped functions and warping functions on the test set

Figure 5.12 and Figure 5.13 show the analysis of estimation problem using the different basis, B-spline basis with 4 elements, on the training set and the test set, respectively. The estimated coefficient of regression \( \hat{\beta} \) (red/blue solid line) versus actual coefficient of regression \( \beta_0 \) (solid green line) versus warped functions \( \{(f_i, \gamma_i)\} \) (dashed black lines) on the training set and the test set are shown in Figure 5.12a and Figure 5.13a, respectively. The original functional predictors are aligned with respect to \( \hat{\beta} \). The corresponding warping functions on the training set and the test set are shown in Figure 5.12b and Figure 5.13b, respectively. These warping functions also can be compared to the actual warpings in Figure 5.2b and Figure 5.3b.

The evolution of the cost function \( H \) using Algorithm 4.2 for each link function: linear, quadratic, and cubic polynomial functions are shown in Figure 5.14a, Figure 5.14b, and Figure 5.14c, respectively. These plots show that the cost function \( H \) goes down in all cases and the optimization algorithm provides at least local solutions.

5.4.2 Prediction

Lastly, we predict the response variables on the test set using Equation 4.3.1.

Figure 5.15 shows the prediction results from our elastic functional regression model, FLM, and PAFLM. Similar to the above sections, we compute Mean Squared Prediction Error (MSPE) to compare the prediction accuracy of the elastic functional regression model with two traditional scalar-on-function regression models. Prediction was performed as was described in Equation 4.3.1.
Figure 5.14: Evolution of Cost Function $H$ for each link function $h$

Table 5.3: MSPE of each method using the different basis

<table>
<thead>
<tr>
<th>Method</th>
<th>FLM</th>
<th>PAFLM</th>
<th>$h$: Linear</th>
<th>$h$: Quadratic</th>
<th>$h$: Cubic</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSPE</td>
<td>2.7218</td>
<td>25.3082</td>
<td>1.6123</td>
<td><strong>0.4901</strong></td>
<td>5.6898</td>
</tr>
</tbody>
</table>

Table 5.3 shows that predictions from elastic functional regression model performed well, especially using a quadratic polynomial function for a link function $h$ compared to the other traditional functional regression models.
5.5 Comparison with Nonparametric Regression Models

We implement Nonparametric regression model in Equation 3.4.1 with automatic bandwidth selection method in Equation 3.4.2. We use two different distance function as follows:

1. Gaussian kernel function in $L^2$ space ($NP-L^2$): Gaussian kernel function can be expressed as $K(\mu) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \mu^2}$. $L^2$ norm distance is defined as $d(f, f_i) = ||f - f_i||$ and $||f|| = \sqrt{\langle f, f \rangle} = \sqrt{\int f(t)f(t)\,dt}$.

2. Gaussian kernel function in SRVF space ($NP$-elastic): $d_s(f, f_i) = \lambda d_a + (1 - \lambda) d_p$, where $\lambda$ is a proportion parameter, $d_a = \arg \min_{\gamma \in \Gamma} ||f - (f_i, \gamma)||$ and $d_p = ||\sqrt{\dot{\gamma}} - \sqrt{\dot{\gamma_i}}||$.
Table 5.4 shows the results of nonparametric regression models. By comparing with MSPEs in Table 5.1, Table 5.2, and Table 5.3, our elastic functional regression model outperformed in prediction accuracy.

### 5.6 Additional Work

To evaluate the performance of the models more precisely and deeply, we randomly iterate this process 5 times to ensure how elastic functional regression performs compared to other 4 traditional regression models. Then we compute the average and the standard deviation of Mean Squared Prediction Errors (MSPEs) from 5 different sets and use these quantities to compare different models. In addition to FLM and PAFLM mentioned above, we also implement and compute MSPEs of two nonparametric regression models –NP using the $\ell^2$ norm and $\ell^2$ using elastic distance.

Table 5.5: The average and the standard deviation (in parentheses) of the five MSPEs for three model-based methods on simulated test data.

<table>
<thead>
<tr>
<th>Basis</th>
<th>Fourier2</th>
<th>Fourier4</th>
<th>Bspline4</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLM</td>
<td>21.70 (20.79)</td>
<td>21.15 (19.16)</td>
<td>27.98 (33.47)</td>
</tr>
<tr>
<td>PAFLM</td>
<td>22.66(11.68)</td>
<td>147.12 (141.10)</td>
<td>29.89 (37.74)</td>
</tr>
<tr>
<td>$h$: Linear</td>
<td>1.99 (0.90)</td>
<td>2.16 (1.04)</td>
<td>3.76 (3.17)</td>
</tr>
<tr>
<td>$h$: Quadratic</td>
<td>0.87 (0.96)</td>
<td>0.96 (1.14)</td>
<td>5.47 (4.38)</td>
</tr>
<tr>
<td>$h$: Cubic</td>
<td>4.93 (4.32)</td>
<td>4.14 (4.06)</td>
<td>11.56 (7.88)</td>
</tr>
</tbody>
</table>

Table 5.6: The average and the standard deviation (in parentheses) of the five MSPEs for nonparametric regression model under the two distances.

<table>
<thead>
<tr>
<th>Model</th>
<th>NP-$\ell^2$</th>
<th>NP-elastic</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSPE</td>
<td>10.39 (9.12)</td>
<td>17.73 (14.81)</td>
</tr>
</tbody>
</table>
The numerical results for the average of the five MSPEs and corresponding standard deviation on simulated data are shown in Table 5.5 and Table 5.6. As these results show, the proposed elastic functional regression model is able to provide a better prediction performance than the competing models despite using very simple models. In addition, the predictions from PAFLM are less accurate since this method is aligning the functional predictors without considering response variables, \( \{y_i\} \). Also, the nonparametric regression model cannot perform well since this model captures all its information about data. This can be a problem since it captures all errors.
CHAPTER 6
APPLICATION TO REAL DATA

We applied our elastic functional regression model on real data to demonstrate the effectiveness of this framework. Our goal of this paper is to demonstrate how our model performs well when the phase variation exists and is considered as a nuisance variable on real data and compare with this model to the other traditional models.

6.1 Gait in Parkinson’s Disease

We use Gait in Parkinson’s disease data from the Physionet [19]. The database contains the Vertical Ground Reaction Force (VGRF) records of subjects as they walked at their usual, self-selected pace for approximately 2 minutes on level ground. 8 sensors were underneath of each foot that measure force (in Newtons) as a function of time. The outputs of each of these 16 sensors (left: 8 and right: 8) has been digitized and recorded at 100 samples of per second. Based on demographic information, each patient has own Timed Up And Go (TUAG) test which is a simple test used to assess a person’s mobility and requires both static and dynamic balance (second panel). In statistical analysis, we consider VGRF records as the predictor curves and TUAG as scalar responses.

There are three different group of patients in Gait in Parkinson’s disease data. We focus on two convention of the group named as “Ga” and “Si” [16, 17, 44] in the dataset to restrict the same condition of the demographic information of the data. So we have total 59 functions or curves for the analysis.

Figure 6.1 display the plot of one of the vertical ground reaction force (VGRF) of 59 patients (first panel) and TUAG values in the first panel and the second panel, respectively. From the original Gait in Parkinson’s disease data, we exported the first 1-100 time points from 12119 time points for simplicity and fast computation. Then we randomly selected 39 curves to make as the training set and rest of 20 curves as the test set.
VGRF of Gait in Parkinson’s Disease

Figure 6.1: Gait in Parkinson’s Disease Data

Figure 6.2: \( \{f_i\} \) from Gait in Parkinson’s Disease Data

Figure 6.2 shows the exported curves from the original Gait in Parkinson’s disease dataset. We use these curves as a functional covariate \( f_i \).

6.1.1 Estimation

In this section, we show how the observed functions are warped to the estimated regression coefficient \( \hat{\beta} \) on the real data.
Figure 6.3 and Figure 6.4 show the aligned functional predictors obtained by warping during the estimation and the prediction stages of the method. The original functions are drawn in black dashed curves and the warped functions are drawn using the red/blue solid curves. Figure 6.3a shows the curves for the training data and Figure 6.4a shows the curves for the test data. The corresponding optimal time warping functions, $\{\gamma^*_i\}$ on the training set and the test set are shown in Figure 6.3b and Figure 6.4b, respectively. Since the predictor functions look more aligned after the algorithm than before, we can assume that the data contains phase variations that are detrimental to the prediction performance. By handling these phase variations, we can expect higher prediction
accuracy as shown next.

### 6.1.2 Prediction

We plot the prediction results of functional predictor models in Figure 6.5. Red dots present the true responses on the test set (*True*), green marks are the predicted responses from FLM (*FLM*), blue marks are the predictions from PAFLM (*PAFLM*), cyan dots are the predicted responses from the elastic functional regression model using a single-index model, *h*(·): linear (*Linear*), quadratic (*Quadratic*), and cubic (*cubic*) polynomial functions. Magenta dots are the results of nonparametric regression model with Gaussian kernel function in *L*^2^ space (*NP – *L*^2^), yellow dots are the predictions from nonparametric regression model with Gaussian kernel function in SRVF space (*NP-elastic*).

In here, we can see that our predicted responses are closer to the true responses compared to the predictions from any other methods. To investigate our results numerically of the real data, we compute MSPE to compare our elastic functional regression model and the other four traditional functional regression models. The numerical results are shown in the table below.

<table>
<thead>
<tr>
<th>Model</th>
<th>MSPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLM</td>
<td>3.1127</td>
</tr>
<tr>
<td>PAFLM</td>
<td>3.5957</td>
</tr>
<tr>
<td><em>h</em> : <strong>Linear</strong></td>
<td><strong>2.9547</strong></td>
</tr>
<tr>
<td><em>h</em> : Quadratic</td>
<td>2.9694</td>
</tr>
<tr>
<td><em>h</em> : Cubic</td>
<td>3.2817</td>
</tr>
<tr>
<td><em>NP</em>-L^2</td>
<td>3.1495</td>
</tr>
<tr>
<td><em>NP</em>-elastic</td>
<td>3.0021</td>
</tr>
</tbody>
</table>

In Table 6.1, it shows that prediction accuracy of elastic functional regression model using a linear polynomial function performed well compared to any other traditional regression models.
6.2 Metabonomic 1H-NMR (Onion)

Metabonomic 1H-NMR (Nuclear Magnetic Resonance) (or Onion) data [42] originates from 1H NMR analysts of urine from thirty-two rats, fed a diet containing an onion by-product. The aim is to evaluate the in vivo metabolome following intake of onion by-products. The data set contains 31 NMR spectra in the region between 0.7 and 0.3 ppm of 31 rats and some reference chemical values.

Since we have 31 rats informations (observations), we randomly selected 21 curves to make as training set and rest of 10 curves as test set. Figure 6.6 displays the plots of NMR spectra of 31 rats (first panel) and the chemical values which are considered as response variable (second panel). Similar to the gait in Parkinson’s disease data, we exported the first 300 time points from 29001 time points for fast computation on statistical analysis.
Figure 6.6: Metabonomic 1H-NMR data (Onion)

Figure 6.7: \( \{ f_i \} \) from Onion Data

Figure 6.7 shows the exported curves from the Onion NMR dataset. We use these curves as functional covariate \( \{ f_i \} \).

### 6.2.1 Estimation

In this section, we show how the observed functions are warped to the regression coefficient on onion data.
Figure 6.8: Training Set of Onion Data

Figure 6.9: Test Set of Onion Data

Figure 6.8 show the warped \( \{ f_i \} \) on the training set. The first panel shows the original functions (black dashed curves) vs. warped functions (red solid curves) on the training set and the second panel shows corresponding optimal time warping functions, \( \{ \gamma^*_i \} \) by using a dynamic programming. Figure 6.9 show the warped \( \{ f_i \} \) on the test set. The first panel shows original functions (black dashed curves) vs. warped functions (blue solid curves) on the test set and the second panel shows corresponding optimal time warping functions, \( \{ \gamma^*_i \} \). In Figure 6.8 and Figure 6.9, the original functions are aligned under the composition with the optimal time warping functions. In this process, it can be assumed that the phase error or noise were reduced to fit the functional
regression model well. By handing these phase variations, we can expect higher prediction accuracy as shown in later section.

6.2.2 Prediction

Similar to the previous real data example, we plot the prediction results of functional predictor models in Figure 6.10. Red dots present the true responses on the test set (True), green marks are the predicted responses from FLM (FLM), blue marks are the predictions from PAFLM (PAFLM), cyan dots are the predicted responses from the elastic functional regression model using a single-index model, $h(\cdot)$: linear (Linear), quadratic (Quadratic), and cubic (cubic) polynomial functions. Magenta dots are the results of nonparametric regression model with Gaussian kernel function in $L^2$ space (NP – $L^2$), yellow dots are the predictions from nonparametric regression model with Gaussian kernel function in SRVF space (NP-elastic).

Table 6.2: MSPE of each model on Onion data

<table>
<thead>
<tr>
<th>Model</th>
<th>MSPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLM</td>
<td>21.0620</td>
</tr>
<tr>
<td>PAFLM</td>
<td>22.8256</td>
</tr>
<tr>
<td>$h$ : Linear</td>
<td>15.7624</td>
</tr>
<tr>
<td>$h$ : Quadratic</td>
<td>18.2172</td>
</tr>
<tr>
<td>$h$ : Cubic</td>
<td>13.3216</td>
</tr>
<tr>
<td>NP-$L^2$</td>
<td>18.2690</td>
</tr>
<tr>
<td>NP-elastic</td>
<td>21.4728</td>
</tr>
</tbody>
</table>

In Table 6.2, it shows that elastic functional regression model using a cubic polynomial function performed the predictions well compared to any other traditional regression models.

6.3 Historical Stock

QuantQuote has large amount of free historical stock data that is freely available for download from their website. There are total of 200 companies and each company has total 3,926 stock entries during the interval 1/2/1998 to 8/9/2013. For each company’s stock, we exported stock prices from 7/8/2011 to 11/28/2011 to form functional predictors. So there are 100 time points over the selected interval for describing the predictor functions. Then, we compute the average
value of stock prices over a future interval, namely 11/29/2011 to 8/9/2013, to form the scalar response variable.

The Figure 6.11 shows the example of the historical stock data. The functional predictors, 200 stock prices are shown in the first panel of the figure with proportion of total time and scalar response variables. The average of future stock prices are shown in the second panel of Figure 6.11. We use first 140 curves to fit the model and 40 curves will be used to check the prediction.

### 6.3.1 Estimation

Figure 6.12 and Figure 6.13 show the aligned functional predictors obtained by warping during the estimation and the prediction stages of the method. The original functions are drawn in black.
dashed curves and the warped functions are drawn using the red/blue solid curves. Figure 6.12a shows the curves for the training data and Figure 6.13a shows the curves for the test data. The corresponding optimal time warping functions, $\{\gamma_i^*\}$ on the training set and the test set are shown in Figure 6.12b and Figure 6.13b, respectively. Since the predictor functions look more aligned after the algorithm than before, we can assume that the data contains phase variations that are detrimental to the prediction performance. By handling these phase variations, we can expect
higher prediction accuracy as shown next.

6.3.2 Prediction

We plot the prediction results of functional predictor models in Figure 6.14. Red dots present the true responses on the test set (True), green marks are the predicted responses from FLM (FLM), blue marks are the predictions from PAFLM (PAFLM), cyan dots are the predicted responses from the elastic functional regression model using a single-index model, $h(\cdot)$: linear (Linear), quadratic (Quadratic), and cubic (cubic) polynomial functions. Magenta dots are the results of non-parametric regression model with Gaussian kernel function in $L^2$ space ($NP - L^2$), yellow dots are the predictions from non-parametric regression model with Gaussian kernel function in SRVF space ($NP$-elastic).

To investigate our results numerically of the real data, we compute MSPE to compare our elastic regression model and the other four traditional models. The numerical results are shown in the table below.

Table 6.3 presents MSPE of each model. It shows that the predictions of elastic functional regression model outperformed most compare to the other predictions of the functional regression models.

Predictions from the kernel regression model performed less accurately. This might be due to the observed functions having all different heights (relatively) and different starting points. Functional
predictors in each training data and test data have different shapes (different heights and starting points) so nonparametric method cannot handle this problem.
CHAPTER 7
SUMMARY AND FUTURE WORK

7.1 Summary

The statistical functional regression model with phase variability is a well-known challenging problem. We proposed a new elastic approach for handling predictor phase in functional regression models which is based on a norm-preserving warping of the predictors and handling the nuisance phase variability by optimizing the $L^2$ inner product over the warping group in the model. We compare MSPE of the model with several existing methods to demonstrate the effectiveness of this technique using simulated data and real data. An important condition of the elastic functional regression model is that the shape of predictor functions assumed to be the predominant in prediction. Therefore, if phase components also carry important information about the responses then the elastic functional regression model cannot perform well compared to the traditional models.

7.2 Directions of Future Works

As the main contribution of this work is functional data analysis using an elastic framework, we focus on that direction for future work. We provide a short list of open problems that still remain to be addressed:

1. Extend this elastic idea to Functional Logistic Regression or Generalized Functional Linear Regression

Most of data in medical science, biology, and other natural science are binary data which contains only two outcomes, \{0, 1\}. In this sense, functional logistic regression is generally used on binary data to identify the characteristic and find the pattern of the data. Functional logistic regression also can be used to classify any subjects such as genes or neuroimages. However, most of articles do not consider time variations between the subjects and this might lead to wrong outcomes. For example, identification of genes that are involved with cancer is an important study. Some methods and models ignore time or time variation which leads to misclassification of temporal gene expression data. By allowing time variable and performing
functional data analysis, we can increase the accuracy of classification of the gene expression
data. Moreover, if there is a time variation between the replicates of the samples, we could
account for this by approaching elastically to increase the accuracy of the classification. Hence,
we can extend our elastic functional regression model to elastic functional logistic regression
model to have better performances of predictions and classifications on these binary datasets.
Moreover, we can develop the elastic generalized functional linear regression model which can
be the generalized regression model for any datasets.

2. Prove the Asymptotic Convergence Theory

We have added time warping functions, \{\gamma_i\}, in elastic functional regression model and we
use dynamic programming to find the optimal time warping functions. We need to prove
the theory of asymptotic convergence of elastic functional regression model. That is, by
increasing the number of curves, we need to verify that the estimated functional coefficients,
\hat{\beta}(t) becomes true \beta(t). So we could demonstrate the performance of the model by using not
only the computational results but also theoretical results.
REFERENCES


Kyungmin (Mike) Ahn received his B.S. in Mathematics from Yonsei University, Seoul, South Korea and M.S. in Applied Statistics from Purdue University, Indiana, United States. In the Fall of 2013, he began his Ph.D. studies in the department of Statistics at Florida State University under the advisement of Dr. Anuj Srivastava. During this time, his research has focused on functional data analysis, shape analysis, and bioinformatics. He defended his Ph.D. dissertation in April, 2018.