

Functional Generalized Structured Component Analysis

Hye Won Suk

Doctor of Philosophy

Department of Psychology

McGill University
Montreal, Quebec, Canada
December 2012

A thesis is submitted to McGill University in partial fulfillment of
the requirements of the degree of Doctor of Philosophy

Copyright © 2012 Hye Won Suk
All rights reserved.

ACKNOWLEDGMENTS

Foremost, I wish to express my gratitude to my supervisor, Dr. Heungsun Hwang, for his invaluable guidance and immeasurable support for my graduate study. With his immense knowledge, enthusiasm, and patience, he helped me grow as a researcher in quantitative psychology. He was always willing to provide insightful comments, practical assistance, and a number of great research ideas and opportunities. I could not have imagined having a better advisor and mentor for my graduate study. I am also greatly indebted to Dr. Yoshio Takane for his help and guidance as my advisory committee. His persistent hard working, broad research interests without barriers, and incredible patience to students, showed me a great example of what a scholar should be. His advice and encouragement helped me overcome difficult moments of my graduate study and go forward with confidence. I am deeply grateful to Dr. Hsiu-Ting Yu, who also guided me as my advisory committee with supportive encouragement, constructive comments, and interesting questions. I would like to thank Dr. Jooseop Lim for kindly providing his data. I am also thankful to Lylia Khennache for the French translation of the abstract. Additionally, I am grateful to all the members of the quantitative psychology program at McGill University, Kwanghee Jung, Lixing Zhou, Tianyu Tan, Jungkyu Park, and Ji Yeh Choi, who shared countless memories as collaborators and good friends. I am also thankful to my best friends,

Hyeyeon Gong, Myunghee Kim, Yongjin Kim, Saebyul Lee, Sooyeon Lee, Jin Yu and for their enormous emotional support. Special thanks to Jae-Myoung Yu for his unconditional support and faith. Lastly and most importantly, I wish to thank my family. This work would not have been possible without unwavering support and love of my parents and sister. To all these people, I dedicate my dissertation.

ABSTRACT

The invention of sophisticated measurement tools, such as motion capture devices, handheld computers, Bluetooth devices, eye-trackers, and brain scanners, has facilitated the collection of functional data that can be considered to arise from an underlying smooth function varying over a continuum such as time and space. Functional data analysis (FDA) is an emerging branch of statistics, which develops and applies statistical methods for the analysis of such types of data. Various FDA methods have been proposed by extending traditional multivariate statistical methods to accommodate functional data. Nonetheless, there has been little attempt to develop functional extensions of structural equation modeling (SEM), in spite of the remarkable popularity of SEM in various disciplines due to its flexibility of modeling complex relationships among observed and latent variables. This thesis thus aims to propose a general framework for functional SEM, called functional generalized structured component analysis (functional GSCA), to examine a variety of hypothesized relationships among observed and latent variables, while permitting observed variables to be functional rather than multivariate. The thesis begins by describing GSCA and penalized least squares smoothing as the two basic building blocks of the proposed method. Subsequently, it provides the technical details of the proposed method. The model for functional GSCA is provided and a penalized least

squares criterion is developed for parameter estimation, which is minimized by an alternating penalized least squares algorithm. The thesis also demonstrates the usefulness of the proposed method by analyzing synthetic and real data sets. It concludes with discussions on limitations and possible extensions of the proposed method.

ABRÉGÉ

L'invention d'outils de mesures sophistiqués tel que les appareils de capture de mouvements, les ordinateurs portables, les appareils Bluetooth, l'oculométrie et les scanners cérébraux ont facilité la compilation de données fonctionnelles qui peuvent être considérées comme provenant d'une fonction variant sur un continuum tel que l'espace et le temps. L'analyse de données fonctionnelles (ADF) est une discipline émergente des statistiques, qui développe et applique les méthodes statistiques pour l'analyse de ce type de données. Plusieurs méthodes d'ADFs ont été proposées en prolongeant les méthodes traditionnelles de statistiques multi variées pour s'adapter aux données multifonctionnelles. Toutefois, peu de tentatives ont été effectuées dans le développement des extensions fonctionnelles des modèles d'équations structurelles (MES), malgré la popularité significative des MESs en plusieurs disciplines grâce à sa souplesse de la modélisation des relations complexes entre les variables observées et latentes. Cette thèse a donc pour objectif de proposer un cadre général pour les MESs fonctionnels, appelé l'analyse en composantes structurée généralisée fonctionnelle (ASSG fonctionnelle), qui combine l'analyse en composantes structurée généralisée fonctionnelle avec les moindres carrés pénalisés lissés par la fonction spline dans un cadre unifié. La méthode proposée peut être utilisée pour analyser une variété de relations hypothétiques entre des variables observées et latentes, tout en

permettant aux variables observées d'être fonctionnelles plutôt que scalaires.

La thèse commence en décrivant l'ACSG et les moindres carrés pénalisés lissés par la fonction spline tel que les deux parties constituant de la méthode proposée. Le modèle pour l'ACSG fonctionnelle est apporté et le critère des moindres carrés pénalisés sont développés par une estimation paramétrique, qui est minimisée par un algorithme alternatif de moindres carrés pénalisés. La thèse démontre également l'utilité de la méthode proposée par l'analyse de base de données réelles et synthétiques. En conclusion, sont présentées les discussions, limites et possibles extensions de la méthode proposée.

TABLE OF CONTENTS

ACKNOWLEDGMENTS.....	i
ABSTRACT.....	iii
ABRÉGÉ	v
Chapter 1 Introduction	1
Chapter 2 Generalized Structured Component Analysis	8
2.1 The Model	8
2.2 Parameter Estimation	11
Chapter 3 Penalized Least Squares Smoothing.....	16
3.1 Smoothing Methods	16
3.2 Penalized Least Squares Estimation for Smoothing.....	20
Chapter 4 Functional Generalized Structured Component Analysis	24
4.1 The Model	24
4.2 Parameter Estimation	29
4.3 Constrained Models.....	35
4.4 Other Computational Considerations	37
4.4.1 Smoothing Parameters.....	37
4.4.2 Confidence Intervals of Estimates.....	40
4.4.3 Scaling Raw Data and Latent Scores	42
Chapter 5 A Simulation Study	46
5.1 Data Generation Procedure	46
5.2 Results	51
5.3 Summary and Discussion	55

Chapter 6	Empirical Examples.....	67
6.1	The Movie Data.....	67
6.2	The Gait Data	73
Chapter 7	Summary and Discussion	80
References	87

LIST OF TABLES

Table 5-1: The congruence coefficients of the loading functions at $\tau^2 = 0.2$ averaged over 100 replications.....	59
Table 5-2: The congruence coefficients of the loading functions at $\tau^2 = 0.6$ averaged over 100 replications.....	60
Table 5-3: The congruence coefficients of the loading functions at $\tau^2 = 0.9$ averaged over 100 replications.....	61
Table 5-4: The congruence coefficients of the latent variables at $\tau^2 = 0.2$ averaged over 100 replications.....	62
Table 5-5: The congruence coefficients of the latent variables at $\tau^2 = 0.6$ averaged over 100 replications.....	63
Table 5-6: The congruence coefficients of the latent variables at $\tau^2 = 0.9$ averaged over 100 replications.....	64
Table 5-7: The mean squared errors of the path coefficient b_1	65
Table 5-8: The mean squared errors of the path coefficient b_2	66
Table 6-1: The estimates of the path coefficients and their 95% bootstrap confidence intervals of the movie data obtained from functional GSCA	72
Table 6-2: The estimated loadings and weights of the physical size of body and the severity of Parkinson's disease on their indicator variables	77
Table 6-3: The estimates of the path coefficients and their 95% bootstrap confidence intervals of the gait data obtained from functional GSCA	78

LIST OF FIGURES

Figure 2-1: A hypothetical example with three sets of observed variables....	11
Figure 5-1: The structural model used for generating data in the simulation study	47
Figure 5-2: The three loading functions used for generating data in the simulation study	49
Figure 6-1: Weekly advertising spending on (a) network televisions, (b) newspapers, and (c) national spot radios over 15 weeks and (d) box office revenue over 10 weeks of the 152 movies	68
Figure 6-2: The structural model used for analyzing the movie data.....	69
Figure 6-3: The estimated loading functions of the movie data for the advertising spending on the four media obtained under $\lambda = 10$ and $\rho = 10^{-8}$	70
Figure 6-4: The estimated weight functions of the movie data for the advertising spending on the four media obtained under $\lambda = 10$ and $\rho = 10^{-8}$	73
Figure 6-5: The total force under the left foot measured for 8 seconds from 23 Parkinson's disease patients	75
Figure 6-6: The structural model used for analyzing the gait data.....	76
Figure 6-7: The estimated (a) loading function and (b) weight function of the total force under the left foot and their 95% point-wise confidence intervals obtained under $\lambda = 10^5$ and $\rho = 10^2$	78
Figure 7-1: Five synthetic curves that vary in (a) both amplitude and phase, (b) amplitude only, and (c) phase only.....	84

CHAPTER 1

Introduction

The invention of sophisticated measurement tools, such as motion capture devices, handheld computers, Bluetooth devices, eye-trackers, and brain scanners, has facilitated the collection of so-called functional data. Functional data refer to data that are considered to arise from an underlying smooth function varying over a continuum (Ramsay & Silverman, 2005, Chapter 1). The continuum is often time, as will be assumed throughout this thesis, but it can also be spatial location, wavelength, probability, etc.

Researchers in many disciplines have collected a variety of functional data including motion capture data (e.g., Ormoneit, Black, Hastie, & Kjellström, 2005), motor control data (e.g., Mattar & Ostry, 2010), music perception data (e.g., Vines, Krumhansl, Wanderley, & Levitin, 2006), neuroimaging data (e.g., Viviani, Grön, & Spitzer, 2005), pupil diameter data (e.g., Jackson & Sirois, 2009), and speech data (e.g., Aston, Chiou, & Evans, 2010).

Statistical methods for analyzing functional data have been developed under the general name of functional data analysis (FDA), coined by Ramsay and Dalzell (1991). The basic idea of FDA is to treat responses measured at multiple occasions as a single entity, or a smooth function, rather than a collection of unconnected scores. Being smooth means that the adjacent

values of a function are necessarily connected and unlikely to be too different from each other (Ramsay & Silverman, 2005, Chapter 3). Any smooth function can be approximated up to an arbitrary accuracy by a linear combination of a sufficient number of basis functions such as B-splines and Fourier series, which is called a basis function expansion. Basis function expansions enable to represent intrinsically infinite-dimensional functions onto a finite-dimensional vector space and to formulate analyses of functional data within the framework of matrix algebra. Therefore basis function expansions can be readily integrated with multivariate analysis methods established upon matrix algebra.

A variety of FDA methods have been developed by combining traditional multivariate analysis methods with basis function expansions for the analysis of functional data. A few examples of such FDA methods include functional principal components analysis (e.g., Besse & Ramsay, 1986; Cardot, 2000; Girard, 2000; Hall & Hosseini-Nasab, 2006; James, Hastie, & Sugar, 2000; Ocaña, Aguilera, & Valderrama, 1999; Ramsay & Dalzell, 1991; Silverman, 1995; Yao & Lee, 2006; Yao, Müller, & Wang, 2005), functional canonical correlation analysis (e.g., He, Müller, & Wang, 2003; Hwang, Jung, Takane, & Woodward, 2012; Leurgans, Moyeed, & Silverman, 1993), and functional linear models (e.g., Cardot, Ferraty, & Sarda, 2003; Fan & Zhang, 2000; Fan & Zhang, 1999; Gu, 2002; Hastie, Tibshirani, & Friedman., 2001; Tan, Shiyko, Li, Li, & Dierker, 2012; Wu,

Fan, & Müller, 2010; Yao et al., 2005). Refer to Ramsay and Silverman (2005) for a comprehensive discussion of a wide range of FDA methods.

Nonetheless, there has been little attempt to develop functional extensions of structural equation modeling (SEM) despite that SEM is currently the most dominant multivariate statistical methodology in various disciplines including behavioral and social sciences (Hershberger, 2003). SEM has gained its popularity mainly due to its tremendous flexibility that allows researchers to frame and answer complicated research questions (e.g., Kaplan, 2009, Chapter 1). Since SEM is designed for a simultaneous analysis of multiple equations that relate observed and/or latent variables, it provides a general framework for modeling complex relationships among observed and latent variables.

Until now, researchers have analyzed functional data within the framework of SEM by transforming functional data to be suitable for SEM, rather than developing functional extensions of SEM. In this approach, a functional response is often summarized as a few features, which are used for further analyses in place of the functional response. For example, in a pain study (Donaldson et al., 2003), several functional responses were measured such as event related evoked potential (ERP), pupil dilation, skin conductance, heart rate, and respiration, while unpleasant electrical stimuli were delivered to subjects' fingertips. In this study, ERP signal was summarized as its amplitudes and latencies of principal negative and positive

evoked potentials, pupil dilation and skin conductance as their peak amplitudes, and respiration and heart rate as their variances, so as to be used in a series of SEM analyses that examined how the physiological responses varied depending on the intensity of the stimuli. In a research of posttraumatic stress disorder (Taft et al., 2007), the effect of PTSD symptoms on aggression was assumed to be mediated by physiological reactivity such as heart rate and skin conductance, which were functional responses measured throughout task periods. In several SEM analyses to examine such hypothesized relationship among the variables, the overall means of heart rate and skin conductance were used instead of their functional responses. Although this approach is easy to implement, the summary measures of a functional response, such as peak amplitude, latency, mean, and variance, may not necessarily capture the functional response itself as well as its relationship with other variables sufficiently well. Moreover, this way of summarizing functional responses totally ignores temporal variations in data.

Recently, a functional extension of SEM, called linear functional structural equation model (lfSEM; Lindquist, in press), was proposed to assess a mediation effect when the mediating variable is a continuous function rather than a scalar. More specifically, lfSEM assumes that a relationship between two scalar variables is mediated by a functional variable. Unlike the previous approach that summarizes a functional response as a few features, this model allows researchers to take into account the entire

functional response when examining its mediation effect. However, lfSEM is limited in the following two aspects. First, lfSEM can analyze only three observed variables of certain types related in a specific way, i.e., a scalar predictor, a scalar outcome, and a functional mediator. Therefore, lfSEM may not be of use to analyze other types of data containing a larger number of variables that could be associated in a more complicated way. Second, lfSEM does not involve latent variables. A latent variable can be considered as a single summary measure of a functional response just like the features such as peak amplitude, latency, mean, or variance. However, a latent variable is distinct from such a feature that is arbitrarily chosen by researchers with a hope that this feature happens to capture important variation in a functional response. Latent variables are estimated to maximally explain the hypothesized relationships among themselves as well as the variation in the corresponding functional responses. By introducing such latent variables, we can fully exploit the flexibility of SEM without losing important variations in functional data.

This thesis thus aims to propose a more general framework for functional SEM, called functional generalized structured component analysis (functional GSCA hereafter), that can examine a variety of directional relationships among multiple functional responses via latent variables. Functional GSCA integrates generalized structured component analysis (GSCA; Hwang, Desarbo, & Takane, 2007; Hwang & Takane, 2004) with

basis function expansions to deal with functional data. GSCA is a component-based approach to SEM that aims to examine various directional relationships among latent variables as well as the relationships between latent and observed variables, in which a latent variable is defined as a component, or a weighted composite, of a vector of observed variables. GSCA estimates three sets of parameters: weight vectors, loading vectors, and path coefficients. A weight vector is to define a latent variable, or a component; a loading vector indicates the effect of a latent variable on the corresponding response vector; path coefficients reflect directional relationships among latent variables. Functional GSCA estimates the same three sets of parameters. However, functional GSCA permits an observed variable to be functional rather than scalar. A functional response defined over a period of time can be regarded as an infinite-dimensional vector, in which each dimension corresponds to each time point. Therefore, functional GSCA estimates an infinite-dimensional weight vector, i.e., a weight function, to define a latent variable of a functional response. Similarly, a loading vector in GSCA is replaced by a loading function in functional GSCA when a response is functional. A weight function indicates which time point of a functional response is crucial for investigating hypothesized relationships with other latent variables as well as for capturing the between-subject variation in the functional response. A loading function represents

how the effect of a latent variable varies over time to generate a functional response, which captures the temporal variation in the functional response.

The remaining chapters of this thesis are organized as follows.

Chapters 2 and 3 provide brief descriptions of the two building blocks of functional GSCA. Chapter 2 reviews the original GSCA, and Chapter 3 introduces penalized least squares smoothing methods. Chapter 4 discusses the technical details of functional GSCA. In this chapter, the functional GSCA model is provided and a penalized least squares criterion is developed for parameter estimation, which is minimized by an alternating penalized least squares algorithm. In addition, the relationship between functional GSCA and the original GSCA is expounded, which is followed by some discussions on various computational issues. In Chapter 5, the results of a simulation study are given with focusing on the accuracy of parameter recovery of functional GSCA. Chapter 6 illustrates the empirical usefulness of functional GSCA by analyzing two real data sets. The final chapter summarizes this thesis and discusses some limitations as well as possible extensions of functional GSCA.

CHAPTER 2

Generalized Structured Component Analysis

As stated earlier, the original GSCA was proposed as a component-based SEM approach that aims to examine a variety of directional relationships among latent variables as well as the relationships between latent and observed variables, in which latent variables are defined as weighted composites of observed variables. Functional GSCA builds upon and extends the original GSCA to deal with functional data. In this chapter, the original GSCA model and its algorithm for parameter estimation are briefly reviewed in order to facilitate the understanding of functional GSCA.

2.1 The Model

GSCA involves two submodels: measurement and structural models. The measurement model specifies hypothesized relationships between latent and observed variables. On the other hand, the structural model reflects hypothesized directional relationships among latent variables. Let \mathbf{Z}_k denote an N by P_k matrix of observed responses of N subjects on P_k variables in the k th set ($k = 1, \dots, K$). Each set of observed variables is assumed to be generated and well summarized by a latent variable. Then the measurement model can be written as:

$$\mathbf{Z}_k = \gamma_k \mathbf{c}_k' + \boldsymbol{\varepsilon}_k, \quad (2.1)$$

where γ_k is an N by 1 vector of the scores on the k th latent variable of N subjects, \mathbf{c}_k is a P_k by 1 vector of loadings relating the k th latent variable to the corresponding observed variables, and $\boldsymbol{\varepsilon}_k$ is an N by 1 vector of measurement errors or residuals for the k th set of observed variables. The structural model is defined by:

$$\mathbf{\Gamma} = \mathbf{\Gamma B} + \mathbf{E}, \quad (2.2)$$

where $\mathbf{\Gamma}$ is an N by K matrix of the scores of the K latent variables, i.e., $\mathbf{\Gamma} = [\gamma_1, \dots, \gamma_K]$, \mathbf{B} is a K by K matrix of path coefficients reflecting directional relationships among the latent variables, and \mathbf{E} is an N by K matrix of structural errors or residuals.

The GSCA model has three sets of parameters to estimate: latent variable scores (γ_k), loadings (\mathbf{c}_k), and path coefficients (\mathbf{B}). In order to estimate the scores of latent variables, GSCA assumes that latent variables are defined as components, or weighted composites of observed variables, which can be written as:

$$\gamma_k = \mathbf{Z}_k \mathbf{w}_k, \quad (2.3)$$

where \mathbf{w}_k is a P_k by 1 vector of weights to define the k th latent variable.

Therefore, estimating the scores of latent variable γ_k reduces to estimating the corresponding weight vector \mathbf{w}_k .

To illustrate the GSCA model including (2.1), (2.2) and (2.3), a hypothetical example is displayed in Figure 2-1, in which squares represent observed variables, circles indicate latent variables, ε 's are measurement errors, and e is a structural error. In Figure 2-1, there are three sets of observed variables ($K = 3$), where the first set involves two observed variables (Z_1 and Z_2), the second set three observed variables (Z_3 , Z_4 , and Z_5) and the third set four observed variables (Z_6 , Z_7 , Z_8 , and Z_9). In other words, \mathbf{Z}_1 is a matrix of order N by 2, \mathbf{Z}_2 is of N by 3, and \mathbf{Z}_3 is of N by 4. The corresponding three weight vectors are given as follows:

$$\mathbf{w}_1 = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}, \mathbf{w}_2 = \begin{bmatrix} w_3 \\ w_4 \\ w_5 \end{bmatrix}, \mathbf{w}_3 = \begin{bmatrix} w_6 \\ w_7 \\ w_8 \\ w_9 \end{bmatrix}, \quad (2.4)$$

and similarly, the loading vectors are given as follows:

$$\mathbf{c}_1 = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}, \mathbf{c}_2 = \begin{bmatrix} c_3 \\ c_4 \\ c_5 \end{bmatrix}, \mathbf{c}_3 = \begin{bmatrix} c_6 \\ c_7 \\ c_8 \\ c_9 \end{bmatrix}. \quad (2.5)$$

In Figure 2-1, the latent variable γ_3 is predicted by the two other latent variables γ_1 and γ_2 , which is reflected by the following matrix of path coefficients:

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & b_1 \\ 0 & 1 & b_2 \\ 0 & 0 & 0 \end{bmatrix}, \quad (2.6)$$

where the exogenous latent variables, γ_1 and γ_2 , have path coefficients of unity heading to themselves. Note that without such unity path coefficients, the error matrix \mathbf{E} in (2.2) will contain structural errors as well as exogenous latent variable scores.

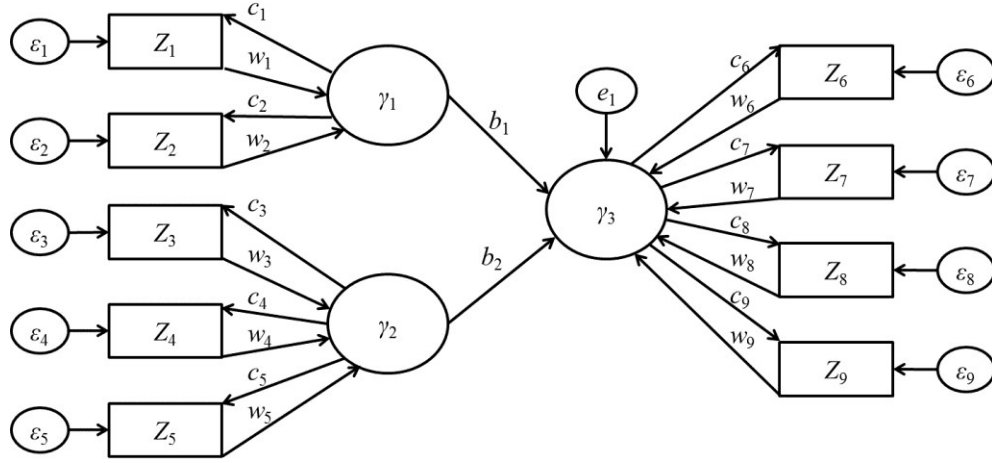


Figure 2-1: A hypothetical example with three sets of observed variables.

2.2 Parameter Estimation

GSCA assumes that the data matrix \mathbf{Z}_k is columnwise normalized to have zero mean and unit norm for each variable. The parameters, \mathbf{w}_k , \mathbf{c}_k , and \mathbf{B} , are estimated by minimizing the following objective function subject to the constraint that the norm of each latent variable should equal to unity:

$$\phi = \sum_{k=1}^K SS(\boldsymbol{\epsilon}_k) + SS(\mathbf{E}), \quad (2.7)$$

where $SS(\mathbf{E}) = tr(\mathbf{E}'\mathbf{E})$ indicates the sum of squares of all elements in the matrix \mathbf{E} . The first term of the objective function represents the sum of

squared errors in the measurement model and the second term the sum of squared errors in the structural model. In other words, GSCA estimates the parameters so as to minimize the sum of squared errors in both measurement and structural models.

To estimate \mathbf{w}_k , \mathbf{c}_k , and \mathbf{B} that minimize the objective function, an alternating least squares algorithm (de Leeuw, Young, & Takane, 1976; Hwang, Desarbo, et al., 2007; Hwang & Takane, 2004) was developed, in which the estimates are randomly initialized and the following three steps iterate until convergence. In each of the three steps, one set of parameters is updated to minimize the objective function with other sets of parameters being fixed.

STEP 1. Update \mathbf{w}_k for fixed \mathbf{c}_k and \mathbf{B} . Based on (2.1) and (2.3), the first term of the objective function can be rewritten as:

$$\begin{aligned}\phi_1 &= \sum_{k=1}^K SS(\boldsymbol{\varepsilon}_k) \\ &= \sum_{k=1}^K SS(\mathbf{Z}_k - \mathbf{Z}_k \mathbf{w}_k \mathbf{c}_k') \\ &= \sum_{k=1}^K SS(\text{vec}(\mathbf{Z}_k) - (\mathbf{c}_k \otimes \mathbf{Z}_k) \mathbf{w}_k),\end{aligned}\tag{2.8}$$

where $\text{vec}(\mathbf{Z}_k)$ is a super-vector obtained by stacking the columns of \mathbf{Z}_k in order and \otimes indicates the Kronecker product. Similarly, by using (2.3), the second term of the objective function can be rewritten as:

$$\begin{aligned}
\phi_2 &= SS(\mathbf{E}) \\
&= SS(\mathbf{\Gamma} - \mathbf{\Gamma}\mathbf{B}) \\
&= SS(\mathbf{\Gamma}(\mathbf{I}_K - \mathbf{B})) \\
&= SS\left((\mathbf{\Gamma}_{(-k)} + \gamma_k \mathbf{u}_k')(\mathbf{I}_K - \mathbf{B})\right) \\
&= SS\left(\mathbf{\Gamma}_{(-k)}(\mathbf{I}_K - \mathbf{B}) + \gamma_k \mathbf{u}_k'(\mathbf{I}_K - \mathbf{B})\right) \\
&= SS\left(\mathbf{\Gamma}_{(-k)}(\mathbf{I}_K - \mathbf{B}) + \mathbf{Z}_k \mathbf{w}_k \mathbf{u}_k'(\mathbf{I}_K - \mathbf{B})\right) \\
&= SS\left(\text{vec}\left(\mathbf{\Gamma}_{(-k)}(\mathbf{I}_K - \mathbf{B})\right) + \left((\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \otimes \mathbf{Z}_k\right) \mathbf{w}_k\right),
\end{aligned} \tag{2.9}$$

where $\mathbf{\Gamma} = \mathbf{\Gamma}_{(-k)} + \gamma_k \mathbf{u}_k'$, $\mathbf{\Gamma}_{(-k)}$ indicate $\mathbf{\Gamma}$ whose k th column is replaced by $\mathbf{0}$, \mathbf{u}_k is the unit vector of order K whose k th element is unity but all the other elements are zeroes, and \mathbf{I}_K is the identity matrix of order K . The updated \mathbf{w}_k is obtained by taking partial derivative of ϕ with respect to \mathbf{w}_k and setting it to zero as given by:

$$\begin{aligned}
\frac{\partial \phi}{\partial \mathbf{w}_k} &= \frac{\partial \phi_1}{\partial \mathbf{w}_k} + \frac{\partial \phi_2}{\partial \mathbf{w}_k} \\
&= -2(\mathbf{c}_k \otimes \mathbf{Z}_k)' \text{vec}(\mathbf{Z}_k) + 2(\mathbf{c}_k \otimes \mathbf{Z}_k)' (\mathbf{c}_k \otimes \mathbf{Z}_k) \mathbf{w}_k \\
&\quad + 2\left((\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \otimes \mathbf{Z}_k\right)' \text{vec}\left(\mathbf{\Gamma}_{(-k)}(\mathbf{I}_K - \mathbf{B})\right) \\
&\quad + 2\left((\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \otimes \mathbf{Z}_k\right)' \left((\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \otimes \mathbf{Z}_k\right) \mathbf{w}_k = \mathbf{0}.
\end{aligned} \tag{2.10}$$

Solving (2.10) yields the updated \mathbf{w}_k given by:

$$\begin{aligned}
\hat{\mathbf{w}}_k &= \left(\mathbf{c}_k' \mathbf{c}_k \mathbf{Z}_k' \mathbf{Z}_k + \mathbf{u}_k' (\mathbf{I}_K - \mathbf{B})' (\mathbf{I}_K - \mathbf{B}) \mathbf{u}_k \mathbf{Z}_k' \mathbf{Z}_k \right)^{-1} \\
&\quad \left(\mathbf{Z}_k' \mathbf{Z}_k \mathbf{c}_k - \mathbf{Z}_k' \mathbf{\Gamma}_{(-k)} (\mathbf{I}_K - \mathbf{B}) (\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \right),
\end{aligned} \tag{2.11}$$

which is derived based on the following properties of the Kronecker product:

$$(\mathbf{c}_k \otimes \mathbf{Z}_k)' = \mathbf{c}_k' \otimes \mathbf{Z}_k', \quad (2.12)$$

$$(\mathbf{c}_k' \otimes \mathbf{Z}_k')(\mathbf{c}_k \otimes \mathbf{Z}_k) = \mathbf{c}_k' \mathbf{c}_k \otimes \mathbf{Z}_k' \mathbf{Z}_k = \mathbf{c}_k' \mathbf{c}_k \mathbf{Z}_k' \mathbf{Z}_k. \quad (2.13)$$

After $\hat{\mathbf{w}}_k$ is obtained, $\hat{\mathbf{w}}_k$ is normalized so that the norm of the latent variable is unity, i.e., $\boldsymbol{\gamma}_k' \boldsymbol{\gamma}_k = \mathbf{w}_k' \mathbf{Z}_k' \mathbf{Z}_k \mathbf{w}_k = 1$.

STEP 2. Update \mathbf{c}_k for fixed \mathbf{w}_k and \mathbf{B} . Note that the loading vector \mathbf{c}_k is involved in the first term of the objective function only. The first term of the objective function can be rewritten as:

$$\begin{aligned} \phi_1 &= \sum_{k=1}^K SS(\boldsymbol{\varepsilon}_k) \\ &= \sum_{k=1}^K SS(\mathbf{Z}_k - \boldsymbol{\gamma}_k \mathbf{c}_k') \end{aligned} \quad (2.14)$$

Solving $\partial \phi_1 / \partial \mathbf{c}_k = \mathbf{0}$ yields the updated \mathbf{c}_k given by:

$$\hat{\mathbf{c}}_k = \mathbf{Z}_k' \boldsymbol{\gamma}_k (\boldsymbol{\gamma}_k' \boldsymbol{\gamma}_k)^{-1}. \quad (2.15)$$

STEP 3. Update \mathbf{B} for fixed \mathbf{w}_k and \mathbf{c}_k . When updating \mathbf{B} , it should be considered that there are many fixed elements in \mathbf{B} as illustrated in (2.6), which are not be to updated. Also note that the path coefficient matrix \mathbf{B} is involved in the second term of the objective function only. The second term of the objective function can be rewritten as:

$$\begin{aligned} \phi_2 &= SS(\boldsymbol{\Gamma} - \boldsymbol{\Gamma} \mathbf{B}) \\ &= SS(\text{vec}(\boldsymbol{\Gamma}) - (\mathbf{I}_K \otimes \boldsymbol{\Gamma}) \text{vec}(\mathbf{B})) \\ &= SS(\text{vec}(\boldsymbol{\Gamma}) - \boldsymbol{\Phi} \mathbf{b}), \end{aligned} \quad (2.16)$$

where \mathbf{b} is a vector containing only the free parameters of $\text{vec}(\mathbf{B})$ and Φ is a matrix containing the columns of $\mathbf{I}_K \otimes \Gamma$ corresponding to the free parameters of $\text{vec}(\mathbf{B})$. Solving $\partial\phi_2 / \partial\mathbf{b} = \mathbf{0}$ yields the updated \mathbf{b} given by:

$$\hat{\mathbf{b}} = (\Phi' \Phi)^{-1} \Phi' \text{vec}(\Gamma), \quad (2.17)$$

and the updated \mathbf{B} can be obtained by putting the elements of $\hat{\mathbf{b}}$ in the appropriate locations in \mathbf{B} . In fact, the parameters \mathbf{c}_k and \mathbf{B} can be updated simultaneously (see Hwang, Desarbo, et al., 2007, Appendix). However, for the purpose of clearly showing the relationship between the original GSCA with functional GSCA, the two steps updating \mathbf{c}_k and \mathbf{B} are presented separately.

The overall fit of a hypothesized model can be measured by the FIT (Hwang & Takane, 2004). The FIT indicates the total variation of all endogenous variables explained by the specified model, which is given by:

$$FIT = 1 - \frac{\phi}{SS(\Psi)}, \quad (2.18)$$

where ϕ is the objective function value as in (2.7) indicating the sum of squared errors of the specified model and Ψ is a matrix consisting of all endogenous observed and latent variables. The FIT ranges from 0 to 1. A larger FIT value indicates that more variation in the endogenous variables is explained by the specified model.

CHAPTER 3

Penalized Least Squares Smoothing

Functional GSCA is a functional extension of the original GSCA that can deal with functional data and estimate weight and loading functions. In order to work with functions, functional GSCA builds upon basis function expansions, in which functions are represented and estimated without any assumptions on their shapes except smoothness. This chapter introduces smoothing methods that are required to represent and estimate smooth functions, mainly focusing on the penalized least squares smoothing.

3.1 Smoothing Methods

In functional GSCA, data are assumed to be functional. By functional, it means that responses measured over multiple occasions are assumed to arise from an underlying smooth function (Ramsay & Silverman, 2005, Chapter 3). A smooth function has the property that a pair of adjacent values of the function tends to be similar and unlikely to be too different from each other. Technically speaking, a smooth function indicates a function that possesses one or more derivatives. Raw observed scores, however, often do not look smooth because they are perturbed by noise or measurement error.

Therefore observed scores, usually given as a pair (t_j, z_j) , can be modeled by:

$$z_j = \xi(t_j) + \delta_j, \quad (3.1)$$

where z_j indicates a score measured at the j th measurement occasion ($j = 1, \dots, J$), $\xi(t_j)$ is an underlying smooth function generating z_j evaluated at time t_j , and δ_j is noise or measurement error at occasion j . The underlying smooth function $\xi(t_j)$ will be called *data function* hereafter.

In most real applications, data functions fluctuate in somewhat complicated ways so that it may not be satisfactory to assume a parametric shape of change for representing data functions, such as linear, quadratic, or cubic. In this situation, a more flexible method for turning raw discrete data into smooth functions needs to be considered, which is called *smoothing*. There are two different approaches for smoothing: kernel smoothing and basis function expansions.

The key idea of the kernel smoothing is that the value of a data function at time t must be similar and linked to the observed scores measured at time points near t . Therefore the value of a data function at time t can be estimated by a weighted sum of observed scores measured at time points around t , which is given by:

$$\xi(t) = \sum_{j=1}^J s_j(t) z_j, \quad (3.2)$$

where the weight $s_j(t)$ becomes greater as t approaches to t_j . There have been proposed a number of different ways to define appropriate weight values $s_j(t)$ for kernel smoothing (see Hastie et al., 2001, Chapter 6, for various types of kernel smoothing). Although this approach makes intuitive sense and is easy to implement, this approach will not be used in functional GSCA. Instead, the basis function expansion will be adopted for functional GSCA because it is more useful for representing not only data functions but also unknown parameter functions that are of the main interest of functional GSCA.

The basic idea of basis function expansions is that any function in a function space can be represented as a linear combination of basis functions that span the function space. This is just as any vector in a vector space can be represented as a linear combination of basis vectors that span the vector space. In other words, any data function can be approximated to some arbitrary degree of precision by taking a linear combination of a sufficiently large number L of basis functions, which can be written as:

$$\xi(t) = \sum_{l=1}^L x_l \theta_l(t) = \mathbf{x}'\boldsymbol{\theta}(t) \quad (3.3)$$

where x_l indicates the weight, or coefficient, of the l th basis function, $\theta_l(t)$ is the value of the l th basis function evaluated at time t , $\mathbf{x} = [x_1, \dots, x_L]'$, and $\boldsymbol{\theta}(t) = [\theta_1(t), \dots, \theta_L(t)]'$. There have been proposed a number of different kinds of basis functions to approximate different types of functions. Among

them, Fourier and spline basis functions are the two most widely used basis functions (Ramsay & Silverman, 2005, Chapter 3). Fourier basis functions are useful for representing a stable periodic function that does not show strong local fluctuations. Spline basis functions are appropriate for representing a non-periodic function that might fluctuate locally. Functional GSCA uses spline basis function to represent data functions as well as unknown parameter functions due to its flexibility.

Spline functions are piecewise polynomial functions (for a comprehensive discussion on spline functions, see de Boor, 2001; Gu, 2002; Wahba, 1990). To define a spline function, we need to divide the interval over which a function is approximated into the predetermined number M of subintervals. The values that separate these M subintervals are called breakpoints, or knots. Over each subinterval, a spline function is defined as a polynomial of order m . The order of a polynomial indicates the number of parameters required to define the polynomial. For example, the order is two for a linear function, three for a quadratic function, four for cubic function, etc. A spline function must join at each breakpoint and its derivatives up to order $m - 2$ must also match up at each breakpoint so as to be smooth. There are a number of different spline functions that satisfy these constraints such as P-spline, M-spline, and B-spline (for various kinds of spline functions, see de Boor, 2001; Ramsay & Silverman, 2005, Chapter 3; Schumaker, 2007). Functional GSCA adopts B-spline functions due to its flexibility, popularity,

and computational efficiency. How to determine the location of breakpoints and the order of polynomials will be discussed at the end of Section 3.2. In the meantime, they will be assumed to be already determined.

3.2 Penalized Least Squares Estimation for Smoothing

Once the type and number of basis functions have been determined, the remaining problem is to estimate the coefficients of the basis functions, or \mathbf{x} in (3.3). The coefficients of the basis functions can be estimated by minimizing the sum of squared errors in approximation, which is given by:

$$\begin{aligned} \text{SSE} &= \sum_{j=1}^J \left(z_j - \xi(t_j) \right)^2 \\ &= \sum_{j=1}^J \left(z_j - \mathbf{x}'\boldsymbol{\theta}(t_j) \right)^2 \\ &= (\mathbf{z} - \boldsymbol{\Theta}\mathbf{x})' (\mathbf{z} - \boldsymbol{\Theta}\mathbf{x}), \end{aligned} \tag{3.4}$$

where \mathbf{z} is a J by 1 vector of observed scores, i.e., $\mathbf{z} = [z_1, \dots, z_J]'$ and $\boldsymbol{\Theta}$ is a J by L matrix of basis function values, i.e., $\boldsymbol{\Theta} = [\boldsymbol{\theta}(t_1), \dots, \boldsymbol{\theta}(t_J)]'$. Solving $\partial \text{SSE} / \partial \mathbf{x} = \mathbf{0}$ yields the estimated vector of the coefficients as follows:

$$\hat{\mathbf{x}} = (\boldsymbol{\Theta}'\boldsymbol{\Theta})^{-1} \boldsymbol{\Theta}'\mathbf{z}. \tag{3.5}$$

When the number of basis functions equals to the number of measurement occasions ($L = J$), the raw data are interpolated, i.e., the estimated coefficients of basis functions will yield $z_j = \mathbf{x}'\boldsymbol{\theta}(t_j)$ for all j . In this case, the observed scores are perfectly fitted by the estimated data

function, which is not always desirable because there is a risk of overfitting. When observed scores are overfitted by the estimated data function, noise or measurement error that we want to ignore is also fitted and hence the estimated data function becomes rough or highly fluctuating due to the error.

One way to reduce the risk of overfitting is to control the degree of roughness of the estimated data function by introducing a penalty term to the objective function (3.4) as follows:

$$\text{PENSSE} = \sum_{j=1}^J \left(z_j - \xi(t_j) \right)^2 + \lambda \int_T \left(D^2 \xi(t) \right)^2 dt, \quad (3.6)$$

where $D^2 \xi(t)dt$ indicates the second derivative of the data function $\xi(t)$,

$\int_T \left(D^2 \xi(t) \right)^2 dt$ is the squared second derivative of the data function integrated over the interval T over which the data function is defined, and λ is a non-negative penalty parameter called *smoothing parameter*. The integrated squared second derivative is used as a penalty term for the following reason. A straight line, which has no curvature or roughness, will have a zero second derivative. Therefore the squared second derivative of a function at time t indicates its curvature or roughness at time t . Consequently the squared second derivative of a function integrated over T indicates its overall curvature or roughness over T .

The smoothing parameter λ controls the importance of the penalty term in estimating the data function. When $\lambda = 0$, we obtain the least squares estimate (3.5) that minimizes the sum of squared errors in approximation. As

λ increases, the penalty term becomes more and more important and the resultant estimated data function will become smoother. As λ approaches to infinity, the resultant estimated data function will become linear. In other words, the degree of roughness of a data function is determined by the size of λ . An optimal value of λ can be determined by using the generalized cross-validation method (Craven & Wahba, 1979; Ramsay & Silverman, 2005, Chapter 5).

The basis function coefficients that minimize (3.6) can be obtained as follows. The PENSSE can be rewritten as:

$$\begin{aligned}
 \text{PENSSE} &= \sum_{j=1}^J \left(z_j - \mathbf{x}'\boldsymbol{\theta}(t_j) \right)^2 + \lambda \int_T \left(D^2 \mathbf{x}'\boldsymbol{\theta}(t) \right)^2 dt, \\
 &= (\mathbf{z} - \boldsymbol{\Theta}\mathbf{x})' (\mathbf{z} - \boldsymbol{\Theta}\mathbf{x}) + \lambda \int_T \left(D^2 \mathbf{x}'\boldsymbol{\theta}(t) D^2 \boldsymbol{\theta}(t) \mathbf{x} \right) dt \\
 &= (\mathbf{z} - \boldsymbol{\Theta}\mathbf{x})' (\mathbf{z} - \boldsymbol{\Theta}\mathbf{x}) + \lambda \mathbf{x}' \left(\int_T D^2 \boldsymbol{\theta}(t) D^2 \boldsymbol{\theta}(t) dt \right) \mathbf{x} \\
 &= (\mathbf{z} - \boldsymbol{\Theta}\mathbf{x})' (\mathbf{z} - \boldsymbol{\Theta}\mathbf{x}) + \lambda \mathbf{x}' \mathbf{R} \mathbf{x},
 \end{aligned} \tag{3.7}$$

where $\mathbf{R} = \int_T D^2 \boldsymbol{\theta}(t) D^2 \boldsymbol{\theta}(t) dt$, which can be numerically obtained (Ramsay & Silverman, 2005, Chapter 5). Solving $\partial \text{PENSSE} / \partial \mathbf{x} = \mathbf{0}$ yields the estimated vector of the coefficients given by:

$$\hat{\mathbf{x}} = (\boldsymbol{\Theta}'\boldsymbol{\Theta} + \lambda \mathbf{R})^{-1} \boldsymbol{\Theta}'\mathbf{z}. \tag{3.8}$$

After estimating the basis function coefficients, a data function can be obtained by using (3.3). This idea of penalized least squares smoothing will be used repeatedly in estimating parameter functions of functional GSCA later on.

Now that the penalized least squares smoothing has been introduced, we can discuss how to determine the location of the breakpoints of spline basis functions and the order of polynomials. When the objective function PENSSE is minimized over an entire set of functions without any assumptions, it has been proven that the curve $\xi(t)$ that minimizes the PENSSE is a cubic spline with breakpoints at every data point t_j (de Boor, 2001; Ramsay & Silverman, 2005, Chapter 3; Wahba, 1990). According to this theorem, we can put breakpoints at every observed time point, t_j , and use spline functions of order four, which is called *cubic spline smoothing*. Once the location of breakpoints and the order of spline functions are determined, the number of spline basis functions L automatically follows, i.e., $L = M + m - 1$ (see Ramsay & Silverman, 2005, Chapter 3).

From now on, it will be assumed that data functions are already obtained from raw data by using the cubic spline smoothing with B-spline basis functions and functional GSCA model will be developed upon these data functions.

Chapter 4

Functional Generalized Structured Component Analysis

In this chapter, functional GSCA is developed by combining the two building blocks, the original GSCA and the penalized least squares smoothing. First, the functional GSCA model is formulated, which is followed by the parameter estimation algorithm. The relationship between functional GSCA and the original GSCA is demonstrated. Various computational issues are also discussed.

4.1 The Model

Let's assume that K data functions are obtained by measuring subjects on K variables continuously over time, upon which various relationships among latent variables are to be examined. As the original GSCA, functional GSCA consists of two submodels: measurement and structural models. The measurement model specifies hypothesized relationships between data functions and latent variables, which can be written as:

$$\xi_{ik}(t_k) = \gamma_{ik}\alpha_k(t_k) + \varepsilon_{ik}(t_k), \quad (3.9)$$

where $\xi_{ik}(t_k)$ is a data function of subject i ($i = 1, \dots, N$) on variable k ($k = 1, \dots, K$) evaluated at time t_k defined over an interval T_k , γ_{ik} is a score

of the k th latent variable of subject i , $\alpha_{ik}(t_k)$ is a loading function evaluated at time t_k , and $\varepsilon_{ik}(t_k)$ is a function of measurement error or residual. In this measurement model, the data function $\xi_{ik}(t_k)$ is assumed to be generated by an underlying latent variable γ_{ik} , which reflects the amplitude of the k th data function of subject i . The effect, or loading, of the latent score γ_{ik} on the data function $\xi_{ik}(t_k)$ is assumed to change over time, which is represented by the loading function $\alpha_k(t_k)$. Note that $\alpha_k(t_k)$ does not have subscript i , which indicates that the loading function is assumed to be common to all subjects. In other words, functional GSCA assumes that data functions on variable k share a common shape of change across all subjects, which is represented by $\alpha_k(t)$, and the data functions vary across subjects only in terms of amplitude of change, represented by γ_{ik} . Therefore, the k th loading function represents the temporal variation in the functional data measured on the k th variable and the latent variable γ_{ik} captures the between-subjects variability in terms of amplitude. The residual function, $\varepsilon_{ik}(t_k)$, represents the portion of the data function that cannot be explained by the hypothesized measurement model.

The structural model of functional GSCA specifies hypothesized directional relationships among latent variables, or amplitude scores. The structural model of functional GSCA is defined by:

$$\mathbf{\Gamma} = \mathbf{\Gamma B} + \mathbf{E}, \quad (3.10)$$

which is identical to the structural model of the original GSCA in (2.2).

Therefore functional GSCA has three sets of parameters to estimate: the scores of latent variables (Γ), loading functions ($\alpha_k(t_k)$), and path coefficients (\mathbf{B}). In order to estimate the scores of latent variables, functional GSCA assumes that a latent variable is defined as a weighted integration of a data function, i.e.,

$$\gamma_{ik} = \int_{T_k} \xi_{ik}(t_k) \omega_k(t_k) dt_k, \quad (3.11)$$

where $\omega_k(t_k)$ indicates a weight function to define the k th latent variable.

This assumption is comparable to the assumption of the original GSCA that a latent variable is defined as a weighted sum of observed variables as in (2.3).

A data function can be thought of as an infinite dimensional vector of observed variables, in which each dimension corresponds to each time point. Consequently, a weight vector to define a latent variable in the original GSCA becomes an infinite dimensional weight vector, i.e., a weight function in functional GSCA. The value of a weight function at time t_k indicates the effect of a data function on the corresponding latent variable at t_k . In other words, by examining the value of a weight function over time, we can see which time point is crucial for defining a latent variable, which will be used for examining the relationships with other latent variables as well as for capturing the variability in the corresponding data function.

In order to estimate weight functions and loading functions, we need to represent these functions by using basis function expansions as introduced in Section 3.1:

$$\omega_k(t_k) = \mathbf{y}_k' \boldsymbol{\theta}(t_k) = \boldsymbol{\theta}(t_k)' \mathbf{y}_k, \quad (3.12)$$

$$\alpha_k(t_k) = \mathbf{a}_k' \boldsymbol{\theta}(t_k) = \boldsymbol{\theta}(t_k)' \mathbf{a}_k, \quad (3.13)$$

where \mathbf{y}_k is an L_k by 1 vector of basis function coefficients for the k th weight function, L_k is the number of basis functions for the k th data function determined by the cubic spline smoothing as introduced in Section 3.2, \mathbf{a}_k is an L_k by 1 vector of basis function coefficients for the k th loading function, and $\boldsymbol{\theta}(t_k)$ is an L_k by 1 vector of basis function values evaluated at time t_k . Note that both \mathbf{y}_k and \mathbf{a}_k are of the same order L_k , which indicates that the number of basis functions for both weight and loading functions is assumed to be identical to that for data functions.

By using (3.12) and (3.3), we can rewrite latent variable in (3.11) as follows:

$$\begin{aligned} \gamma_{ik} &= \int_{T_k} \xi_{ik}(t_k) \omega_k(t_k) dt_k \\ &= \int_{T_k} \mathbf{x}_{ik}' \boldsymbol{\theta}(t_k) \boldsymbol{\theta}(t_k)' \mathbf{y}_k dt_k \\ &= \mathbf{x}_{ik}' \left(\int_{T_k} \boldsymbol{\theta}(t_k) \boldsymbol{\theta}(t_k)' dt_k \right) \mathbf{y}_k \\ &= \mathbf{x}_{ik}' \mathbf{Q}_k \mathbf{y}_k, \end{aligned} \quad (3.14)$$

where \mathbf{x}_{ik} is an L_k by 1 vector of the coefficients of basis functions for the data function of subject i on variable k and $\mathbf{Q}_k = \int_{T_k} \boldsymbol{\theta}(t_k) \boldsymbol{\theta}(t_k)' dt_k$, which can be obtained numerically (see Ramsay & Silverman, 2005, Chapter 5). Putting the scores of the k th latent variable of N subjects into one vector yields the following compact representation:

$$\boldsymbol{\gamma}_k = \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k, \quad (3.15)$$

where $\boldsymbol{\gamma}_k$ is an N by 1 vector of the scores of the k th latent variable, i.e., $\boldsymbol{\gamma}_k = [\gamma_{1k}, \dots, \gamma_{Nk}]'$, and \mathbf{X}_k is an N by L_k matrix of basis function coefficients of the k th data functions, i.e., $\mathbf{X}_k = [\mathbf{x}_{1k}, \dots, \mathbf{x}_{Nk}]'$. Likewise, by using (3.3), (3.13), and (3.14), the measurement model (3.9) can be rewritten as:

$$\mathbf{x}_{ik}' \boldsymbol{\theta}(t_k) = \mathbf{x}_{ik}' \mathbf{Q}_k \mathbf{y}_k \mathbf{a}_k' \boldsymbol{\theta}(t_k) + \varepsilon_{ik}(t_k), \quad (3.16)$$

and the measurement model for all N subjects can be compactly represented as the following:

$$\mathbf{X}_k \boldsymbol{\theta}(t_k) = \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{a}_k' \boldsymbol{\theta}(t_k) + \boldsymbol{\varepsilon}_k(t_k), \quad (3.17)$$

where $\boldsymbol{\varepsilon}_k(t_k)$ is an N by 1 vector of measurement errors on variable k at time t_k , i.e., $\boldsymbol{\varepsilon}_k(t_k) = [\varepsilon_{1k}(t_k), \dots, \varepsilon_{Nk}(t_k)]'$.

In sum, the two submodels, (3.17) and (3.10), define the functional GSCA model and the three sets of parameters, \mathbf{y}_k , \mathbf{a}_k , and \mathbf{B} , are to be estimated.

4.2 Parameter Estimation

Functional GSCA estimates the parameters, \mathbf{y}_k , \mathbf{a}_k , and \mathbf{B} , by minimizing the following objective function:

$$f = \sum_{k=1}^K \int_{T_k} SS(\boldsymbol{\varepsilon}_k(t_k)) dt_k + SS(\mathbf{E}) + \sum_{k=1}^K \lambda_k \mathbf{y}_k' \mathbf{R}_k \mathbf{y}_k + \sum_{k=1}^K \rho_k \mathbf{a}_k' \mathbf{R}_k \mathbf{a}_k, \quad (3.18)$$

subject to the constraint that the norm of each latent variable should equal to the number of subjects. The first term of the objective function indicates the integrated squared errors in the measurement model (3.17) summed over all N subjects and all K variables. The second term of the objective function indicates the sum of squared errors in the structural model (3.10). The third and the last terms are the penalty terms that control the degree of roughness of weight and loading functions, respectively. As mentioned in Section 3.2, the overall curvature or roughness of a function can be measured by its integrated squared second derivative. Therefore the overall curvature of the k th weight function $\omega_k(t_k)$ can be measured by:

$$\int_{T_k} \left(D^2 \omega_k(t_k) \right)^2 dt_k, \quad (3.19)$$

which can be rewritten by using (3.12) as the following:

$$\begin{aligned} \int_{T_k} \left(D^2 \omega_k(t_k) \right)^2 dt_k &= \int_{T_k} \left(D^2 \mathbf{y}_k' \boldsymbol{\theta}(t_k) D^2 \boldsymbol{\theta}(t_k)' \mathbf{y}_k \right) dt_k \\ &= \mathbf{y}_k' \left(\int_{T_k} D^2 \boldsymbol{\theta}(t_k) D^2 \boldsymbol{\theta}(t_k)' dt_k \right) \mathbf{y}_k \\ &= \mathbf{y}_k' \mathbf{R}_k \mathbf{y}_k, \end{aligned} \quad (3.20)$$

where $\mathbf{R}_k = \int_{T_k} D^2 \boldsymbol{\theta}(t_k) D^2 \boldsymbol{\theta}(t_k)' dt_k$ as in (3.7). Similarly, the overall curvature of the k th loading function $\alpha_k(t_k)$ can be measured by:

$$\int_{T_k} \left(D^2 \alpha_k(t_k) \right)^2 dt_k = \mathbf{a}_k' \mathbf{R}_k \mathbf{a}_k. \quad (3.21)$$

The nonnegative smoothing parameters, λ_k and ρ_k , determine the importance of the corresponding penalty terms. When $\lambda_k = \rho_k = 0$ for all k , minimizing the objective function is equivalent to minimizing the errors in the measurement and structural models. As discussed in Section 3.2, minimizing the errors, in other words, maximizing the fit to given data, takes the risk of overfitting, which may yield highly fluctuating estimated functions. By using greater values of the smoothing parameters, the risk of overfitting can be reduced and smoother weight and loading functions can be obtained. The optimal values of the smoothing parameters can be determined by a cross-validation method, which will be further discussed in Section 4.4.1.

In order to estimate \mathbf{y}_k , \mathbf{a}_k , and \mathbf{B} that minimize the objective function (3.18), an alternating penalized least squares algorithm is developed. This algorithm starts with random initial values of \mathbf{y}_k , \mathbf{a}_k , and \mathbf{B} and iterates the following three steps until convergence. In each of the three steps, only one set of parameters is updated to minimize the objective function while the other two sets of parameters are being fixed.

STEP 1. Update \mathbf{y}_k for fixed \mathbf{a}_k and \mathbf{B} . By using (3.17), the first term of the objective function in (3.18) can be rewritten as follows:

$$\begin{aligned}
f_1 &= \sum_{k=1}^K \int_{T_k} SS(\boldsymbol{\varepsilon}_k(t_k)) dt_k \\
&= \sum_{k=1}^K \int_{T_k} SS(\mathbf{X}_k \boldsymbol{\theta}(t_k) - \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{a}'_k \boldsymbol{\theta}(t_k)) dt_k \\
&= \sum_{k=1}^K tr \left[\int_{T_k} (\mathbf{X}_k \boldsymbol{\theta}(t_k) - \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{a}'_k \boldsymbol{\theta}(t_k)) (\mathbf{X}_k \boldsymbol{\theta}(t_k) - \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{a}'_k \boldsymbol{\theta}(t_k))' dt_k \right] \\
&= \sum_{k=1}^K \left\{ tr \left[\mathbf{X}_k \left(\int_{T_k} \boldsymbol{\theta}(t_k) \boldsymbol{\theta}(t_k)' dt_k \right) \mathbf{X}_k' \right] \right. \\
&\quad \left. - 2tr \left[\mathbf{X}_k \left(\int_{T_k} \boldsymbol{\theta}(t_k) \boldsymbol{\theta}(t_k)' dt_k \right) \mathbf{a}_k \mathbf{y}_k' \mathbf{Q}_k \mathbf{X}_k' \right] \right. \\
&\quad \left. + tr \left[\mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{a}'_k \left(\int_{T_k} \boldsymbol{\theta}(t_k) \boldsymbol{\theta}(t_k)' dt_k \right) \mathbf{a}_k \mathbf{y}_k' \mathbf{Q}_k \mathbf{X}_k' \right] \right\}. \\
&= \sum_{k=1}^K \left\{ tr [\mathbf{X}_k \mathbf{Q}_k \mathbf{X}_k'] - 2tr [\mathbf{X}_k \mathbf{Q}_k \mathbf{a}_k \mathbf{y}_k' \mathbf{Q}_k \mathbf{X}_k'] \right. \\
&\quad \left. + tr [\mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{a}'_k \mathbf{Q}_k \mathbf{a}_k \mathbf{y}_k' \mathbf{Q}_k \mathbf{X}_k'] \right\}
\end{aligned} \tag{3.22}$$

Similarly, by using (3.15) and $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}_{(-k)} + \boldsymbol{\gamma}_k \mathbf{u}_k'$ as in (2.9), the second term of the objective function in (3.18) can be rewritten as follows:

$$\begin{aligned}
f_2 &= SS(\mathbf{E}) \\
&= SS(\mathbf{\Gamma} - \mathbf{\Gamma B}) \\
&= SS(\mathbf{\Gamma}(\mathbf{I}_K - \mathbf{B})) \\
&= SS((\mathbf{\Gamma}_{(-k)} + \mathbf{\gamma}_k \mathbf{u}'_k)(\mathbf{I}_K - \mathbf{B})) \\
&= SS((\mathbf{\Gamma}_{(-k)} + \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{u}'_k)(\mathbf{I}_K - \mathbf{B})) \\
&= tr \left[(\mathbf{\Gamma}_{(-k)} + \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{u}'_k)(\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' (\mathbf{\Gamma}_{(-k)} + \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{u}'_k)' \right] \\
&= tr \left[\mathbf{\Gamma}_{(-k)} (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{\Gamma}_{(-k)}' \right] \\
&\quad + 2tr \left[\mathbf{\Gamma}_{(-k)} (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \mathbf{y}_k' \mathbf{Q}_k \mathbf{X}_k \right] \\
&\quad + tr \left[\mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{u}'_k (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \mathbf{y}_k' \mathbf{Q}_k \mathbf{X}_k \right]
\end{aligned} \tag{3.23}$$

The updated \mathbf{y}_k is obtained by taking partial derivative of f with respect to

\mathbf{y}_k and setting it to zero as follows:

$$\begin{aligned}
\frac{\partial f}{\partial \mathbf{y}_k} &= \frac{\partial f_1}{\partial \mathbf{y}_k} + \frac{\partial f_2}{\partial \mathbf{y}_k} + \frac{\partial \lambda_k \mathbf{y}_k' \mathbf{R}_k \mathbf{y}_k}{\partial \mathbf{y}_k} \\
&= -2\mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \mathbf{a}_k + 2\mathbf{a}_k' \mathbf{Q}_k \mathbf{a}_k \mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \\
&\quad + 2\mathbf{Q}_k \mathbf{X}_k' \mathbf{\Gamma}_{(-k)} (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \\
&\quad + 2\mathbf{u}_k' (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \\
&\quad + 2\lambda_k \mathbf{R}_k \mathbf{y}_k \\
&= \mathbf{0},
\end{aligned} \tag{3.24}$$

which yields

$$\begin{aligned}
\hat{\mathbf{y}}_k &= \left(\mathbf{a}_k' \mathbf{Q}_k \mathbf{a}_k \mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k + \mathbf{u}_k' (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k + \lambda_k \mathbf{R}_k \right)^{-1} \\
&\quad \left(\mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \mathbf{a}_k - \mathbf{Q}_k \mathbf{X}_k' \mathbf{\Gamma}_{(-k)} (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \right).
\end{aligned} \tag{3.25}$$

After $\hat{\mathbf{y}}_k$ is obtained, $\hat{\mathbf{y}}_k$ is normalized so that the squared norm of the latent variable is equal to the number of subjects, i.e., $\hat{\mathbf{y}}_k' \mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \hat{\mathbf{y}}_k = N$. Scaling of latent variables as well as raw data will be further discussed in Section 4.4.3.

STEP 2. Update \mathbf{a}_k for fixed \mathbf{y}_k and \mathbf{B} . The updated \mathbf{a}_k is obtained by taking partial derivative of f with respect to \mathbf{a}_k and setting it to zero as follows:

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{a}_k} &= \frac{\partial f_1}{\partial \mathbf{a}_k} + \frac{\partial \rho_k \mathbf{a}_k' \mathbf{R}_k \mathbf{a}_k}{\partial \mathbf{a}_k} \\ &= -2\mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k + 2\mathbf{y}_k' \mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{Q}_k \mathbf{a}_k + 2\rho_k \mathbf{R}_k \mathbf{a}_k \\ &= \mathbf{0}, \end{aligned} \quad (3.26)$$

where f_1 is as given in (3.22). Solving (3.26) yields the update \mathbf{a}_k as the following:

$$\hat{\mathbf{a}}_k = (\mathbf{y}_k' \mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k \mathbf{Q}_k + \rho_k \mathbf{R}_k)^{-1} (\mathbf{Q}_k \mathbf{X}_k' \mathbf{X}_k \mathbf{Q}_k \mathbf{y}_k). \quad (3.27)$$

STEP 3. Update \mathbf{B} for fixed \mathbf{y}_k and \mathbf{a}_k . As in the original GSCA, the second term of the objective function can be written as follows:

$$\begin{aligned} f_2 &= SS(\mathbf{\Gamma} - \mathbf{\Gamma B}) \\ &= SS(\text{vec}(\mathbf{\Gamma}) - (\mathbf{I}_K \otimes \mathbf{\Gamma}) \text{vec}(\mathbf{B})) \\ &= SS(\text{vec}(\mathbf{\Gamma}) - \mathbf{\Phi b}), \end{aligned} \quad (3.28)$$

where \mathbf{b} is a vector containing only free parameters of $\text{vec}(\mathbf{B})$ and $\mathbf{\Phi}$ is a matrix containing the columns of $\mathbf{I}_K \otimes \mathbf{\Gamma}$ corresponding to the free

parameters of $\text{vec}(\mathbf{B})$ as already defined in (2.16). Solving $\partial f_2 / \partial \mathbf{b} = \mathbf{0}$

yields the updated \mathbf{b} given by:

$$\hat{\mathbf{b}} = (\Phi' \Phi)^{-1} \Phi' \text{vec}(\Gamma), \quad (3.29)$$

which is identical to the updated \mathbf{b} in the original GSCA as in (2.17). The

updated \mathbf{B} can be obtained by putting the elements of $\hat{\mathbf{b}}$ in the appropriate locations in \mathbf{B} .

The objective function (3.18) is bounded above zero and each step of the algorithm decreases the value of the objective function. Therefore this algorithm will converge to a solution. However, this does not guarantee that the solution is the global minimum. In order to increase the change of convergence to the global minimum, the algorithm repeats a number of times, say 20, with different initial values each time and the solution associated with the smallest objective function value is determined as the final one.

The goodness-of-fit of a hypothesized model can be measured by the FIT index as in the original GSCA, which is given by:

$$\text{FIT} = 1 - \frac{f_1 + f_2}{\sum_{k=1}^K \text{tr}(\mathbf{X}'_k \mathbf{Q} \mathbf{X}_k) + NK_E}. \quad (3.30)$$

As in the original GSCA, the FIT represents the amount of variation in all endogenous variables that can be explained by the specified model. The first term in the denominator in (3.30) indicates the total variation in data functions, which can be shown by:

$$\begin{aligned}
\sum_{k=1}^K \sum_{i=1}^N \int_{T_k} \xi_{ik}(t_k)^2 dt_k &= \sum_{k=1}^K \sum_{i=1}^N \int_{T_k} (\mathbf{x}'_{ik} \boldsymbol{\theta}(t_k))^2 dt_k \\
&= \sum_{k=1}^K \int_{T_k} \text{tr}(\mathbf{X}_k \boldsymbol{\theta}(t_k) \boldsymbol{\theta}(t_k)' \mathbf{X}_k') dt_k \\
&= \sum_{k=1}^K \text{tr}(\mathbf{X}_k \mathbf{Q}_k \mathbf{X}_k').
\end{aligned} \tag{3.31}$$

When the total variation is calculated, the mean of $\xi_{ik}(t)$ does not have to be subtracted because $\xi_{ik}(t)$ is already centered to have zero mean at each time point t_k , which will be further discussed in Section 4.4.3. The second term in the denominator represents the total variation in endogenous latent scores. Since each latent variable is centered to have zero mean and normalized to have its squared norm equal to N , the sum of the squared norms of all endogenous latent variables is NK_E , where K_E indicates the number of endogenous latent variables. In the numerator in (3.30), f_1 and f_2 indicate the amount of errors in the measurement and structural model, respectively. Therefore, the FIT ranges from 0 to 1 and a larger value indicates a better fit. One can use the FIT to compare different models and choose one associated with the highest FIT value as the best model.

4.3 Constrained Models

In real applications, we may encounter situations in which some observed variables are functional but others are not. For example, one might be interested in examining the relationship between subjects' brain signals

and their cognitive ability. In this case, brain signals can be regarded as functional data. Cognitive ability, however, may be measured by a cognitive rating scale consisting of a certain number of items, which cannot be considered functional.

Functional GSCA can easily accommodate this situation as follows. Functional GSCA defines latent variables as given by (3.15). If the k th set of observed variables is not functional but multivariate as in the original GSCA, the k th latent variable can be defined as in the original GSCA:

$$\boldsymbol{\gamma}_k = \mathbf{Z}_k \mathbf{w}_k, \quad (3.32)$$

where \mathbf{Z}_k is an N by P_k matrix of P_k observed variables in the k th set, and \mathbf{w}_k is a P_k by 1 vector of weights to define the k th latent variable. We can see that (3.15) reduces to (3.32) by constraining $\mathbf{X}_k = \mathbf{Z}_k$, $\mathbf{Q}_k = \mathbf{I}_{P_k}$, and $\mathbf{y}_k = \mathbf{w}_k$. Note that the corresponding penalty parameters, λ_k and ρ_k , should be set to zero. By setting, $\mathbf{X}_k = \mathbf{Z}_k$, $\mathbf{Q}_k = \mathbf{I}_{P_k}$, $\mathbf{y}_k = \mathbf{w}_k$, and $\rho_k = 0$, the updated $\hat{\mathbf{a}}_k$ given in (3.27) reduces to:

$$\begin{aligned} \hat{\mathbf{a}}_k &= (\mathbf{w}_k' \mathbf{Z}_k' \mathbf{Z}_k \mathbf{w}_k)^{-1} (\mathbf{Z}_k' \mathbf{Z}_k \mathbf{w}_k) \\ &= (\boldsymbol{\gamma}_k' \boldsymbol{\gamma}_k)^{-1} \mathbf{Z}_k' \boldsymbol{\gamma}_k \\ &= \mathbf{Z}_k' \boldsymbol{\gamma}_k (\boldsymbol{\gamma}_k' \boldsymbol{\gamma}_k)^{-1}, \end{aligned} \quad (3.33)$$

which is identical to $\hat{\mathbf{c}}_k$, the updated loading in the original GSCA as given in (2.15). Similarly, by setting $\mathbf{X}_k = \mathbf{Z}_k$, $\mathbf{Q}_k = \mathbf{I}_{P_k}$, $\mathbf{a}_k = \mathbf{c}_k$, and $\lambda_k = 0$, the updated $\hat{\mathbf{y}}_k$ in (3.25) reduces to

$$\hat{\mathbf{y}}_k = \left(\mathbf{c}'_k \mathbf{c}_k \mathbf{Z}'_k \mathbf{Z}_k + \mathbf{u}'_k (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \mathbf{Z}'_k \mathbf{Z}_k \right)^{-1} \left(\mathbf{Z}'_k \mathbf{Z}_k \mathbf{c}_k - \mathbf{Z}'_k \mathbf{\Gamma}_{(-k)} (\mathbf{I}_K - \mathbf{B})(\mathbf{I}_K - \mathbf{B})' \mathbf{u}_k \right), \quad (3.34)$$

which is identical to the updated weight $\hat{\mathbf{w}}_k$ in the original GSCA as given in (2.11). The matrix of path coefficients \mathbf{B} will be updated as described in the step 3 in Section 4.2, which is again identical to the updated path coefficients in the original GSCA.

In sum, functional GSCA can deal with both functional and multivariate data by constraining $\mathbf{X}_k = \mathbf{Z}_k$, $\mathbf{Q}_k = \mathbf{I}_{P_k}$, and $\lambda_k = \rho_k = 0$ for the k th set of observed variables when it is multivariate. Consequently, the original GSCA can be considered as the most constrained version of functional GSCA, in which $\mathbf{X}_k = \mathbf{Z}_k$, $\mathbf{Q}_k = \mathbf{I}_{P_k}$, and $\lambda_k = \rho_k = 0$ for all k .

4.4 Other Computational Considerations

4.4.1 Smoothing Parameters

This section discusses how to determine the values of the smoothing parameters λ_k and ρ_k that were assumed to be known in Sections 4.1 and 4.2. In order to reduce a computational burden, it will be assumed that $\lambda_1 = \dots = \lambda_K = \lambda$ and $\rho_1 = \dots = \rho_K = \rho$. Functional GSCA uses G -fold cross-validation for estimating λ and ρ . First, a manageable number of pairs of smoothing parameter values (λ, ρ) are set to be tested. Original data are

divided into G subgroups of approximately equal size. Among the G subgroups, a single subgroup is retained as the validation or test set and the remaining $G - 1$ subgroups are used as the training set. The parameters are estimated based on the training set under a given pair of smoothing parameters and the estimated parameters are used to calculate a predictor error in the test set. This process repeats G times with each of the G subgroups being used as the test. At the end, the prediction errors from the G test sets are summed to produce the overall prediction error. The pair of smoothing parameter values (λ, ρ) associated with the smallest overall prediction error is chosen as the final one.

The overall prediction error produced by a given pair of smoothing parameter values can be calculated as follows:

$$\begin{aligned} \text{PE}(\lambda, \rho) = & \sum_{g=1}^G \left\{ \sum_{k=1}^K \int_{T_k} SS \left(\xi_k^{(g)}(t_k) - \gamma_k^{(g)} \alpha_k^{(-g)}(t_k) \right) dt_k \right. \\ & \left. + SS \left(\Gamma^{(g)} - \Gamma^{(g)} \mathbf{B}^{(-g)} \right) \right\}, \end{aligned} \quad (3.35)$$

where $\xi_k^{(g)}(t_k)$ indicates an N_g by 1 vector of data function values of the N_g subjects belonging to the g th test set on variable k at time t_k , $\gamma_k^{(g)}$ in an N_g by 1 vector of the scores of the k th latent variable of the same N_g subjects, $\Gamma^{(g)} = [\gamma_1^{(g)}, \dots, \gamma_K^{(g)}]$, $\alpha_k^{(-g)}(t_k)$ is the loading function of the k th variable obtained from the g th training set under (λ, ρ) , and $\mathbf{B}^{(-g)}$ is the matrix of path coefficients obtained from the g th training set under (λ, ρ) . In (3.35),

the first term $\sum_{k=1}^K \int_{T_k} SS(\xi_k^{(g)}(t_k) - \gamma_k^{(g)} \alpha_k^{(-g)}(t_k)) dt_k$ indicates the sum of squared prediction errors in the measurement model integrated over t_k and the second term $SS(\Gamma^{(g)} - \Gamma^{(g)} \mathbf{B}^{(-g)})$ represents the sum of squared prediction errors in the structural model.

Based on (3.3), the vector of data function values, $\xi_k^{(g)}(t_k)$ can be rewritten as:

$$\xi_k^{(g)}(t_k) = \mathbf{X}_k^{(g)} \boldsymbol{\theta}_k(t_k), \quad (3.36)$$

where $\mathbf{X}_k^{(g)}$ indicates an N_g by L_k matrix of basis function coefficients of the N_g subjects belonging to the test set g for variable k . Likewise, based on (3.13) the loading function $\alpha_k^{(-g)}(t_k)$ can be rewritten as:

$$\alpha_k^{(-g)}(t_k) = \mathbf{a}_k^{(-g)'} \boldsymbol{\theta}_k(t_k), \quad (3.37)$$

where $\mathbf{a}_k^{(-g)}$ indicates the vector of basis function coefficients for the k th loading function obtained from the g th training set. Then the prediction error in the g th test set can be rewritten as follows:

$$\begin{aligned}
\text{PE}_g(\lambda, \rho) &= \sum_{k=1}^K \int_{T_k} SS\left(\xi_k^{(g)}(t_k) - \gamma_k^{(g)} \alpha_k^{(-g)}(t_k)\right) dt_k \\
&\quad + SS\left(\Gamma^{(g)} - \Gamma^{(g)} \mathbf{B}^{(-g)}\right) \\
&= \sum_{k=1}^K \int_{T_k} SS\left(\mathbf{X}_k^{(g)} \boldsymbol{\theta}_k(t_k) - \gamma_k^{(g)} \mathbf{a}_k^{(-g)'} \boldsymbol{\theta}_k(t_k)\right) dt_k \\
&\quad + SS\left(\Gamma^{(g)} - \Gamma^{(g)} \mathbf{B}^{(-g)}\right) \\
&= \sum_{k=1}^K \left\{ \text{tr}\left(\mathbf{X}_k^{(g)} \mathbf{Q}_k \mathbf{X}_k^{(g)'}\right) - 2\text{tr}\left(\mathbf{X}_k^{(g)} \mathbf{Q}_k \mathbf{a}_k^{(-g)} \gamma_k^{(g)'}\right) \right. \\
&\quad \left. + \text{tr}\left(\gamma_k^{(g)} \mathbf{a}_k^{(-g)'} \mathbf{Q}_k \mathbf{a}_k^{(-g)} \gamma_k^{(g)'}\right) \right\} \\
&\quad + SS\left(\Gamma^{(g)} - \Gamma^{(g)} \mathbf{B}^{(-g)}\right),
\end{aligned} \tag{3.38}$$

in which $\mathbf{\Gamma}^{(g)} = [\gamma_1^{(g)}, \dots, \gamma_K^{(g)}]$ and $\gamma_k^{(g)}$ is obtained by using (3.15) as follows:

$$\gamma_k^{(g)} = \mathbf{X}_k^{(g)} \mathbf{Q}_k \mathbf{y}_k^{(-g)}, \tag{3.39}$$

where $\mathbf{y}_k^{(-g)}$ is the vector of basis function coefficients for the k th weight function obtained from the g th training set. Once $\mathbf{y}_k^{(-g)}$, $\mathbf{a}_k^{(-g)}$, and $\mathbf{B}^{(-g)}$ are estimated, the prediction error in the g th test set can be easily calculated based on (3.38) and (3.39).

4.4.2 Confidence Intervals of Estimates

Researchers are often interested in not only the values of estimates themselves but also the stability of estimates. If an estimate is highly variable across different samples, the obtained value of the estimate from a given sample may not be meaningful for other samples. The confidence interval of

an estimate delivers such information about the variability of the estimate across different samples. To obtain the confidence intervals of the estimates of functional GSCA, we can use a bootstrap method (Efron, 1982; Hastie et al., 2001, Chapter 8), in which the sampling distribution of an estimate is approximated by the distribution of the estimate obtained from a number of resamples from the observed data. More specifically, if a data set consists of the scores obtained from N subjects, each resample is constructed by a random sampling with replacement of size N from the original N subjects and then an estimate is obtained from this resample. Repeating this resampling and estimating process a sufficient number of times will produce an empirical distribution of the estimate, from which the confidence interval of the estimate is obtained. For example, the 95% confidence interval of an estimate can be obtained by finding 97.5 and 2.5 percentile values from the empirical distribution.

Functional GSCA obtains the confidence intervals of path coefficients in exactly the same way as explained above. Calculating the confidence intervals of weight and loading functions, however, needs special considerations. Although functional GSCA estimates the vectors of basis function coefficients, \mathbf{y}_k and \mathbf{a}_k , for weight and loading functions, respectively, these vectors themselves may not be of the main interest of functional GSCA. It is the weight and loading functions obtained from these basis function coefficient vectors that are meaningful and interpretable.

Therefore calculating the confidence intervals of \mathbf{y}_k and \mathbf{a}_k may not be of use. Instead, it is more informative to calculate the confidence intervals of weight and loading functions.

Functional GSCA obtains the confidence interval of a function as follows. A weight function will be used as an example here but the same procedure will be used for a loading function as well. Once the basis function coefficient vector, \mathbf{y}_k , is obtained for each resample in the bootstrapping procedure, the weight function value evaluated at time t_k , or $\omega_k(t_k)$, can be obtained by using (3.12). After a sufficient number of resamples are obtained, in each of which the weight function is estimated, we can sort the values of the weight function at each time point t_k obtained over all resamples in an ascending order and find the 97.5 percentile and 2.5 percentiles values, which are the upper and lower limits of the 95% confidence interval at the time point t_k .

4.4.3 Scaling Raw Data and Latent Scores

Functional GSCA can obtain estimates that are unique up to scale only. The scale of latent variables is necessarily arbitrary because latent variables are not directly observable and the scales of weight functions, loading functions, and path coefficients depend on the scale of latent scores. This indeterminacy problem is not unique to functional GSCA but true for the

original GSCA as well. In order to resolve this problem, in the original GSCA, observed raw data are centered to have zero mean, which leads the mean of each latent variable to be zero. In addition, both observed and latent variables are normalized to have unit norm so that they become comparable in size.

In functional GSCA, observed raw data (or equivalently data functions) are centered to have zero mean at each time point, which leads the mean of each latent variable to be zero just as in the original GSCA. However, normalizing observed raw data to have unit norm at each time point as in the original GSCA will yield a disastrous result. Figure 4-1(a) shows a set of four synthetic data functions. Each function was generated by using the measurement model (3.9), in which the loading function was the gamma function with the shape parameter of 10 and the size parameter of 5 and the scores of the latent variable were randomly generated from a uniform distribution over the interval (0, 1). Figure 4-1(b) shows the four data functions centered to have zero mean at each time point. Centering changes the scores of the latent variable but the shape of the loading function and the relative locations of the latent variable scores are preserved. Figure 4-1(c) shows the data functions centered to have zero mean plus normalized to have unit norm at each time point. We can see that this type of normalization fails to retain the temporal variation, or the shape of the loading function, and only preserves the relative locations of the latent variable scores. In other words,

normalizing raw data to have unit norm at each time point will result in throwing away the temporal variation in the data. Therefore functional GSCA does not normalize raw data in this way. Instead, functional GSCA normalizes raw data to satisfy the following:

$$SS(\mathbf{Z}_k^*) = NJ_k, \quad (3.40)$$

where \mathbf{Z}_k^* indicates the N by J_k matrix of raw data measured on the k th variable over J_k occasions for N subjects and centered to have zero mean at each occasion. In other words, the data matrix is normalized matrixwise, not columnwise, so that the squared norm of the matrix equals to NJ_k . As we can see in Figure 4-1(d), this type of normalization not only preserves the relative locations of the latent variable scores but also retains the temporal variation, or the information on the shape of the loading function. In addition, functional GSCA normalizes each latent variable to make its squared norm equal to the sample size N , which makes observed data and latent scores comparable in size.

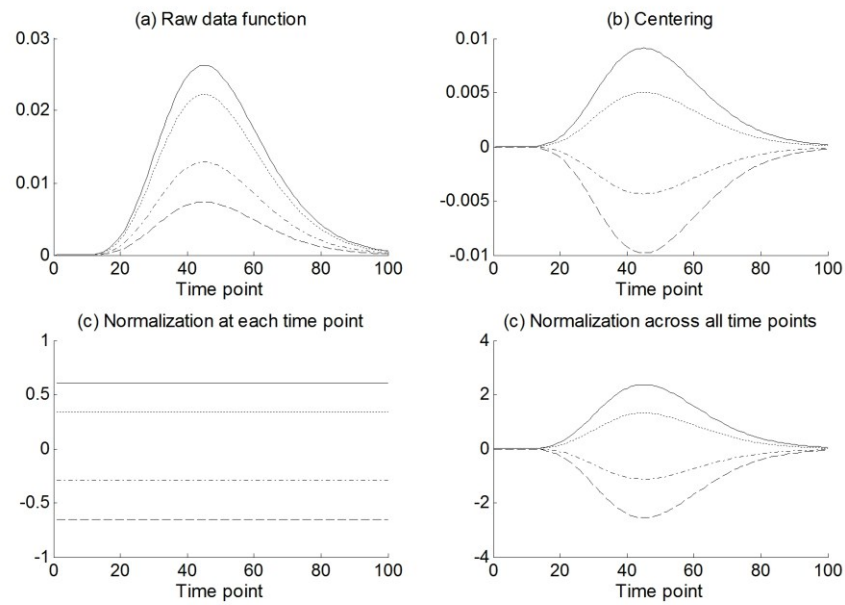


Figure 4-1. The results of different scaling methods on four data functions, each of which is represented by a line with different style: (a) raw data functions, (b) centering only, (c) centering plus columnwise normalization, and (d) centering plus matrixwise normalization.

Chapter 5

A Simulation Study

In this chapter, a Monte Carlo simulation study is designed and conducted in order to investigate the performance of the estimates of functional GSCA under a variety of conditions. In Monte Carlo simulation studies, true models and true parameter values are initially specified, which depend on the conditions that are manipulated so as to affect the accuracy of the estimates. A sufficient number of data sets are generated under each condition. The estimates are obtained for each data set under each condition and then are systematically examined for their accuracy of parameter recovery as a function of the manipulated factors.

5.1 Data Generation Procedure

In the Monte Carlo study for functional GSCA, three latent variables, γ_1 , γ_2 , and γ_3 , were involved, each of which was generated as follows. First, the two latent variables, γ_1 and γ_2 , were assumed to be exogenous variables and the remaining one, γ_3 , was assumed to be an endogenous variable, as shown in Figure 5-1. The two exogenous latent variables, γ_1 and γ_2 , were generated from the bivariate normal distribution $N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)$,

in which each latent variable was generated from the standard normal distribution and the two latent variables were assumed to be uncorrelated, i.e., $\text{cov}(\gamma_1, \gamma_2) = 0$.

The endogenous latent variable, γ_3 , was generated from the structural model as shown in Figure 5, which can be written as:

$$\gamma_3 = b_1\gamma_1 + b_2\gamma_2 + \mathbf{e}, \quad (5.1)$$

where the values of the path coefficients b_1 and b_2 were manipulated to vary at three different levels depending on the variance of the structural error \mathbf{e} .

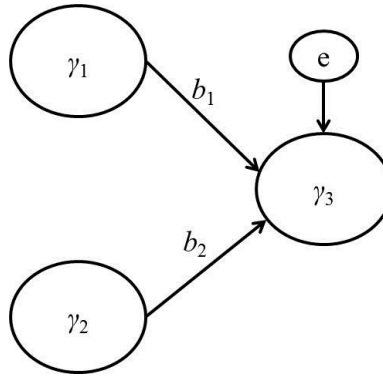


Figure 5-1: The structural model used for generating data in the simulation study.

Assuming that the error \mathbf{e} in the structural model is not correlated with the two exogenous latent variables γ_1 and γ_2 , i.e.,

$\text{cov}(\gamma_1, \mathbf{e}) = \text{cov}(\gamma_2, \mathbf{e}) = 0$, the variance of the endogenous latent variable γ_3 can be calculated as:

$$\begin{aligned}
\text{var}(\gamma_3) &= b_1^2 \text{var}(\gamma_1) + b_2^2 \text{var}(\gamma_2) + \text{var}(\mathbf{e}) \\
&= b_1^2 + b_2^2 + \tau^2 \\
&= 1,
\end{aligned} \tag{5.2}$$

where $\tau^2 = \text{var}(\mathbf{e})$ and the variance of γ_3 is set to unity as for the two exogenous latent variables. The error variance τ^2 in the structural model was varied at three different levels: 0.2, 0.6, and 0.9. For $\tau^2 = 0.2$, $b_1 = 0.8$ and $b_2 = 0.4$ were used to yield $b_1^2 + b_2^2 = 0.8$ and hence $b_1^2 + b_2^2 + \tau^2 = 1$. Likewise, for $\tau^2 = 0.6$, $b_1 = 0.6$ and $b_2 = 0.2$ were used and for $\tau^2 = 0.9$, $b_1 = 0.3$ and $b_2 = 0.1$ were used. After γ_1 and γ_2 were generated and b_1 , b_2 , and τ^2 were determined depending on the condition under which data were generated, γ_3 was generated by using (5.1), in which \mathbf{e} was generated from a normal distribution with mean of zero and variance of τ^2 . More precisely, $[\gamma_1, \gamma_2, \mathbf{e}]$ was generated from the following multivariate normal distribution:

$$N \left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \tau^2 \end{bmatrix} \right). \tag{5.3}$$

Once the latent variables were generated, the corresponding data functions were generated. It was assumed that the responses of all subjects on all variables were measured at the same time points t_j ($j = 1, \dots, J$), in which $t_1 = 0$ and $t_J = 1$. The data functions were generated from the following measurement model:

$$\mathbf{z}_{jk} = \boldsymbol{\gamma}_k \alpha_k(t_j) + \boldsymbol{\varepsilon}_{jk}, \quad k = 1, 2, 3. \quad (5.4)$$

where \mathbf{z}_{jk} is an N by 1 vector of measured responses of N subject on k th variable at time t_j , $\boldsymbol{\gamma}_k$ is an N by 1 vector of the latent variable scores generated as described above, and the loading functions $\alpha_k(t)$ were defined as:

$$\alpha_1(t) = \frac{1}{2} \cos 2\pi(t - \frac{1}{4}) + \frac{1}{2}, \quad (5.5)$$

$$\alpha_2(t) = \cos \pi(t - \frac{1}{2}), \quad (5.6)$$

$$\alpha_3(t) = \cos \frac{\pi}{2} t, \quad (5.7)$$

over $t \in [0, 1]$, which are displayed in Figure 5-2.

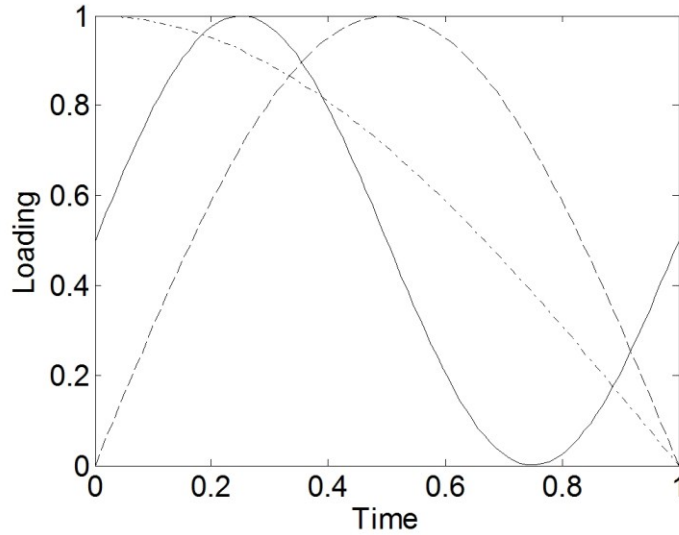


Figure 5-2: The three loading functions used for generating data in the simulation study: the solid line indicates $\alpha_1(t)$, the dashed line $\alpha_2(t)$, and the dash-dotted line $\alpha_3(t)$.

The three loading functions were manipulated to have three different degrees of roughness, i.e., the frequency of 1, $\frac{1}{2}$, and $\frac{1}{4}$, for $\alpha_1(t)$, $\alpha_2(t)$, and $\alpha_3(t)$, respectively. The higher the frequency of a function is the rougher it is. Finally, an N by 1 vector of measurement errors or residuals, ϵ_k , was generated from a normal distribution with mean 0 and variance σ^2 . The error variance σ^2 in the measurement model was assumed to be equal across all variables and all time points, and was varied at three different levels: 0.5, 1, and 2. The data functions were generated by multiplying the already generated latent variables γ_k by the corresponding loading functions $\alpha_k(t)$ and then adding the measurement errors ϵ_k .

Two other factors were manipulated for generating data: the number of subjects (N) and the number of time points (J). The number of subjects was varied at four levels: 25, 50, 100, and 200. The number of time points, which were assumed to be equally spaced, was varied at four levels: 10, 25, 50, and 100. In sum, there were four factors manipulated in this simulation study: the amount of errors in the structural model ($\tau^2 = 0.2, 0.6$, and 0.9), the amount of errors in the measurement model ($\sigma^2 = 0.5, 1$, and 2), the number of subjects ($N = 25, 50, 100$, and 200), and the number of time points ($J = 10, 25, 50$, and 100), which yielded $3 \times 3 \times 4 \times 4 = 144$ conditions. For each

condition, 100 replications were generated, i.e., a total of 14400 data sets were analyzed.

Each data set under each condition was analyzed by functional GSCA with assuming the structural model given in (5.1). Before functional GSCA was applied, each data set was smoothed by the cubic spline smoothing with B-spline basis functions as described in Section 3.2. Under each condition, the optimal values of the smoothing parameters, λ and ρ , were determined by five-fold cross-validation. Each smoothing parameter was varied at six levels, 10^0 , 10^1 , 10^2 , 10^3 , 10^4 , and 10^5 , which yielded $6 \times 6 = 36$ pairs of smoothing parameter values to be tested. This cross-validation was performed only for the first sample out of 100 replications in order to reduce computation time.

5.2 Results

For the investigation of the accuracy of parameter recovery of the loading functions and latent variables, the congruence coefficients (Tucker, 1951) between true parameters and their estimates were examined. The congruence coefficient of a vector of true parameters $\boldsymbol{\eta}$ and a vector of their estimates $\hat{\boldsymbol{\eta}}$ is calculated by:

$$CC(\boldsymbol{\eta}, \hat{\boldsymbol{\eta}}) = \frac{\boldsymbol{\eta}'\hat{\boldsymbol{\eta}}}{\sqrt{\boldsymbol{\eta}'\boldsymbol{\eta}}\sqrt{\hat{\boldsymbol{\eta}}'\hat{\boldsymbol{\eta}}}}, \quad (5.8)$$

which is equivalent to the cosine of the angle between the two vectors $\boldsymbol{\eta}$ and $\hat{\boldsymbol{\eta}}$. As the angle between the two vectors approaches 0 degree, i.e., the two vectors collapse each other, the cosine of the angle approaches 1. As the angle between the two vector approaches 90 degree, i.e., the two vectors are orthogonal, the cosine of the angle approaches 0. As the angle between the two vectors approaches 180 degree, i.e., the two vectors lie in the opposite direction, the cosine of the angle approaches -1. Therefore the congruence coefficient ranges between -1 and 1 and measures the agreement or similarity in terms of the direction of two vectors regardless of their size. A larger value of the congruence coefficient indicates a better agreement of the two vectors. Conventionally a value greater than 0.9 of the congruence coefficient is regarded as an acceptable degree of similarity or agreement (Mulaik, 1971). In order to calculate the congruence coefficient between two functions, i.e., a true loading function and its estimated loading function, each function is evaluated at 100 equally spaced time points and the vector of 100 evaluated values was used instead of the function itself.

In order to examine the accuracy of parameter recovery of the path coefficients, the mean squared errors of the estimates were calculated as given by:

$$\text{MSE}(\hat{\beta}) = E[(\hat{\beta} - \beta)^2], \quad (5.9)$$

where β is a true path coefficient and $\hat{\beta}$ is an estimate of the true path coefficient and the expectation $E[\cdot]$ is taken over 100 replications. The mean squared error of an estimate indicates the average squared distance between the estimate and its true parameter. A smaller value of the mean squared error indicates a smaller distance between an estimate and its true parameter, i.e., a better estimate.

Tables 5-1, 5-2, and 5-3 present the average congruence coefficients of the estimates of the three loading functions over 100 replications at $\tau^2 = 0.2, 0.6$, and 0.9 , respectively. It seems that the amount of errors in the structural model τ^2 had no systematic effect on the accuracy of parameter recovery of the loading functions, which is the reason why the results are presented separately at different levels of τ^2 . It can be observed that the average congruence coefficients were greater than 0.9 for all conditions except for a few cases where the number of subjects was relatively small ($N = 25$ or 50), the number of time points was very small ($J = 10$), the amount of errors in the measurement model was relatively high ($\sigma^2 = 1$ or 2), and the loading function was relatively rough ($\alpha_1(t)$ or $\alpha_2(t)$). Overall, the average congruence coefficients tended to increase as the amount of errors in measurement model decreased, the roughness of loading functions decreased, the number of time points increased, and the number of subjects increased. However, it can be observed that increasing the number of subjects did not

necessarily yield better estimates when the loading functions were relatively rough and the number of time points was relatively small.

The average congruence coefficients of the estimates of the three latent variables over 100 replications at $\tau^2 = 0.2, 0.6$, and 0.9 are shown in Tables 5-4, 5-5, and 5-6, respectively. Again, the amount of errors in the structural model had no systematic effect on the accuracy of parameter recovery of the latent variables as for the case of the loading functions. The average congruence coefficients were greater than 0.9 for all conditions except for the cases where the number of time points was relatively small ($J = 10$ or 25) and the amount of errors in measurement model was relatively high ($\sigma^2 = 1$ or 2). Overall, the average congruence coefficients tended to increase as the amount of errors in the measurement model decreased, the number of time points increased, and the number of subjects increased. Also there was a tendency that the average congruence coefficients were higher for the latent variables whose loading functions were smoother.

Tables 5-7 and 5-8 display the mean squared errors of the estimates of the path coefficients, b_1 and b_2 , respectively. Although there is no clear-cut standard for the mean squared errors to be acceptable, the mean squared errors of the estimated path coefficients were smaller than 0.1 except for a few cases where the number of time points was very small ($J = 10$), the number of subjects was very small ($N = 25$), and the amount of errors in the measurement model was very large ($\sigma^2 = 2$). In most cases, the mean

squared errors were smaller than 0.05, which indicates that functional GSCA yielded reasonably good estimates of the path coefficients. In ideal conditions where the amount of errors in the measurement model was small ($\sigma^2 = 0.5$), the number of subjects was large ($N = 100$, or 200), and the number of time points was relatively large ($T = 25$, 50 , or 100), the mean squared errors were very small, i.e., less than 0.01, which indicates that functional GSCA worked fairly well in such conditions.

Overall, the mean squared errors tended to decrease as the number of subjects increased. In addition, when the amount of errors in the measurement model was small ($\sigma^2 = 0.5$), the mean squared errors tended to increase as the amount of errors in the structural model increased. However, as the amount of errors in the measurement model became larger and the number of time points got smaller, this tendency became less salient. Moreover, when the amount of errors in the measurement model was large ($\sigma^2 = 2$), less number of time points tended to yield more elevated mean squared errors. However, as the amount of errors in the measurement model got smaller, this tendency became less strong.

5.3 Summary and Discussion

In order to examine the accuracy of parameter recovery of functional GSCA, a simulation study was performed in which four factors were manipulated: the amount of errors in the measurement model, the amount of

errors in the structural model, the number of subjects, and the number of time points.

The simulation study showed that functional GSCA yielded reasonably good estimates of the path coefficients in most conditions (the mean squared errors were smaller than 0.05). In addition, functional GSCA produced fairly good estimates of the loading functions and latent variables (congruence coefficients ≥ 0.9) even for a very small number of subjects ($N = 25$) when the amount of errors in the measurement model was small ($\sigma^2 = 0.5$) or the number of time points was relatively large ($J \geq 50$). In addition, functional GSCA produced reasonably good estimates of the loading functions and latent variables for a very small number of time points ($J = 10$) when the sample size was relatively large ($N \geq 100$), the amount of errors in the measurement model was relatively small ($\sigma^2 \leq 1$), and the loading function was fairly smooth ($\alpha_3(t)$).

From the results of the simulation study, we can make the following conclusions. It is beneficial to increase the number of subjects and the number of time points for obtaining better estimates of loading functions and latent variables. The quality of the estimates of loading functions and latent variables also depends on the amount of measurement errors, but not much on the amount of structural errors. As for path coefficients, increasing the number of subjects yields better estimates. Increasing the number of time points, however, is beneficial only when the amount of measurement errors is

high. In other words, in situations where the amount of measurement errors is small, we can still obtain reasonably good estimates based on a small number of time points. However, if the amount of measurement errors is high, a larger number of time points are required to obtain good estimates. The accuracy of the estimates of path coefficients also depends on the amount of structural errors. When the amount of structural errors is small, i.e., the relationships among latent variables are strong, the estimates of path coefficients tend to be more accurate. However, if the amount of measurement errors is high and the number of time points is relatively small, the accuracy of the estimates of path coefficients tends to deteriorate regardless of the amount of structural errors.

Although the simulation study showed that functional GSCA worked as the way it was supposed to, there are some possible limitations. In the simulation study, all the generated data sets were analyzed under the true structural model. However, in most real applications, researchers may not know the true structural model and hence use incorrect or misspecified one. A systematic investigation of the effect of misspecification in the structural model is in order. Another limitation of the simulation study is that the measurement errors were assumed to be normally and identically distributed across all variables and all time points. This assumption may not be the case in real applications and a further investigation of the effect of various structures of measurement errors may be of use. In addition, this simulation

study concerned only three variables. It would be informative to investigate the performance of functional GSCA in situations where a larger number of variables are involved.

Table 5-1: The congruence coefficients of the loading functions at $\tau^2 = 0.2$ averaged over 100 replications (standard deviations in parentheses).

N	J	$\alpha_1(t)$			$\alpha_2(t)$			$\alpha_3(t)$		
		$\sigma^2 = 0.5$	1	2	0.5	1	2	0.5	1	2
25	10	0.9234	0.9198	0.8727	0.9691	0.9571	0.8944	0.9944	0.9910	0.9829
		(.0168)	(.0146)	(.2432)	(.0150)	(.0260)	(.0293)	(.0033)	(.0087)	(.0287)
	25	0.9732	0.9448	0.9349	0.9927	0.9721	0.9605	0.9975	0.9965	0.9942
		(.0160)	(.0190)	(.0164)	(.0124)	(.0189)	(.0168)	(.0019)	(.0032)	(.0083)
	50	0.9872	0.9874	0.9752	0.9984	0.9975	0.9940	0.9989	0.9979	0.9956
		(.0038)	(.0052)	(.0091)	(.0013)	(.0021)	(.0053)	(.0010)	(.0021)	(.0041)
	100	0.9970	0.9972	0.9941	0.9992	0.9982	0.9971	0.9993	0.9981	0.9967
		(.0016)	(.0022)	(.0035)	(.0006)	(.0013)	(.0023)	(.0006)	(.0017)	(.0033)
50	10	0.9273	0.9179	0.9085	0.9800	0.9647	0.9157	0.9957	0.9941	0.9917
		(.0174)	(.0118)	(.0196)	(.0061)	(.0257)	(.0202)	(.0016)	(.0031)	(.0062)
	25	0.9794	0.9543	0.9345	0.9968	0.9750	0.9606	0.9982	0.9975	0.9960
		(.0082)	(.0209)	(.0055)	(.0063)	(.0166)	(.0113)	(.0007)	(.0015)	(.0041)
	50	0.9899	0.9876	0.9816	0.9990	0.9984	0.9971	0.9995	0.9990	0.9979
		(.0025)	(.0034)	(.0052)	(.0009)	(.0013)	(.0024)	(.0005)	(.0008)	(.0016)
	100	0.9992	0.9982	0.9971	0.9995	0.9992	0.9986	0.9995	0.9992	0.9984
		(.0005)	(.0011)	(.0019)	(.0004)	(.0006)	(.0010)	(.0003)	(.0006)	(.0013)
100	10	0.9249	0.9191	0.9166	0.9886	0.9533	0.9309	0.9960	0.9952	0.9936
		(.0125)	(.0081)	(.0112)	(.0033)	(.0102)	(.0305)	(.0010)	(.0013)	(.0039)
	25	0.9815	0.9498	0.9382	0.9983	0.9763	0.9626	0.9985	0.9980	0.9978
		(.0048)	(.0184)	(.0057)	(.0009)	(.0163)	(.0044)	(.0005)	(.0009)	(.0013)
	50	0.9900	0.9886	0.9853	0.9992	0.9989	0.9981	0.9997	0.9995	0.9990
		(.0016)	(.0023)	(.0039)	(.0005)	(.0008)	(.0015)	(.0002)	(.0004)	(.0008)
	100	0.9996	0.9990	0.9983	0.9998	0.9996	0.9992	0.9997	0.9996	0.9992
		(.0002)	(.0005)	(.0010)	(.0001)	(.0003)	(.0005)	(.0002)	(.0003)	(.0006)
200	10	0.9232	0.9202	0.9176	0.9886	0.9842	0.9326	0.9963	0.9959	0.9946
		(.0087)	(.0060)	(.0087)	(.0022)	(.0033)	(.0291)	(.0006)	(.0008)	(.0021)
	25	0.9803	0.9436	0.9386	0.9984	0.9749	0.9637	0.9985	0.9982	0.9980
		(.0017)	(.0136)	(.0024)	(.0006)	(.0168)	(.0035)	(.0003)	(.0006)	(.0008)
	50	0.9901	0.9891	0.9880	0.9994	0.9992	0.9989	0.9998	0.9997	0.9995
		(.0011)	(.0016)	(.0019)	(.0003)	(.0004)	(.0009)	(.0001)	(.0002)	(.0004)
	100	0.9996	0.9995	0.9992	0.9999	0.9998	0.9995	0.9999	0.9998	0.9996
		(.0002)	(.0003)	(.0006)	(.0001)	(.0002)	(.0003)	(.0001)	(.0001)	(.0003)

Table 5-2: The congruence coefficients of the loading functions at $\tau^2 = 0.6$ averaged over 100 replications (standard deviations in parentheses).

N	J	$\alpha_1(t)$			$\alpha_2(t)$			$\alpha_3(t)$		
		$\sigma^2 = 0.5$	1	2	0.5	1	2	0.5	1	2
25	10	0.9307 (.0280)	0.8962 (.1754)	0.8102 (.3993)	0.9826 (.0121)	0.9515 (.0299)	0.9043 (.1808)	0.9936 (.0042)	0.9906 (.0101)	0.9844 (.0171)
		0.9765 (.0142)	0.9440 (.0179)	0.9183 (.1786)	0.9933 (.0120)	0.9715 (.0195)	0.9607 (.0142)	0.9976 (.0016)	0.9962 (.0035)	0.9936 (.0058)
	25	0.9889 (.0034)	0.9852 (.0059)	0.9758 (.0090)	0.9984 (.0013)	0.9972 (.0021)	0.9938 (.0061)	0.9990 (.0008)	0.9978 (.0020)	0.9962 (.0036)
		0.9986 (.0009)	0.9959 (.0027)	0.9937 (.0044)	0.9990 (.0007)	0.9984 (.0014)	0.9972 (.0024)	0.9991 (.0007)	0.9987 (.0011)	0.9973 (.0022)
	50	0.9285 (.0213)	0.9200 (.0148)	0.8772 (.2351)	0.9856 (.0107)	0.9788 (.0186)	0.9343 (.0362)	0.9958 (.0016)	0.9939 (.0033)	0.9918 (.0075)
		0.9716 (.0100)	0.9535 (.0210)	0.9354 (.0105)	0.9970 (.0052)	0.9770 (.0172)	0.9604 (.0104)	0.9980 (.0008)	0.9977 (.0016)	0.9959 (.0034)
	100	0.9891 (.0023)	0.9882 (.0033)	0.9852 (.0058)	0.9990 (.0007)	0.9983 (.0016)	0.9973 (.0022)	0.9994 (.0004)	0.9990 (.0009)	0.9979 (.0021)
		0.9986 (.0007)	0.9981 (.0011)	0.9971 (.0018)	0.9996 (.0003)	0.9992 (.0006)	0.9985 (.0010)	0.9997 (.0002)	0.9992 (.0007)	0.9984 (.0010)
100	10	0.9235 (.0096)	0.9209 (.0070)	0.9118 (.0173)	0.9874 (.0030)	0.9830 (.0100)	0.9369 (.0324)	0.9960 (.0011)	0.9953 (.0016)	0.9934 (.0041)
		0.9781 (.0023)	0.9487 (.0178)	0.9375 (.0026)	0.9977 (.0037)	0.9746 (.0160)	0.9613 (.0053)	0.9983 (.0005)	0.9981 (.0009)	0.9972 (.0018)
	25	0.9897 (.0015)	0.9887 (.0022)	0.9843 (.0040)	0.9992 (.0004)	0.9989 (.0007)	0.9982 (.0014)	0.9998 (.0002)	0.9995 (.0004)	0.9990 (.0008)
		0.9996 (.0002)	0.9990 (.0005)	0.9985 (.0008)	0.9998 (.0002)	0.9996 (.0003)	0.9992 (.0006)	0.9997 (.0001)	0.9996 (.0003)	0.9992 (.0005)
	50	0.9233 (.0076)	0.9197 (.0063)	0.9172 (.0095)	0.9866 (.0023)	0.9824 (.0099)	0.9350 (.0299)	0.9963 (.0006)	0.9957 (.0010)	0.9948 (.0022)
		0.9804 (.0017)	0.9475 (.0165)	0.9380 (.0021)	0.9984 (.0008)	0.9761 (.0159)	0.9625 (.0036)	0.9985 (.0004)	0.9984 (.0005)	0.9980 (.0010)
	100	0.9899 (.0011)	0.9896 (.0014)	0.9866 (.0024)	0.9994 (.0003)	0.9992 (.0004)	0.9988 (.0007)	0.9998 (.0001)	0.9997 (.0002)	0.9995 (.0003)
		0.9996 (.0002)	0.9994 (.0003)	0.9991 (.0005)	0.9999 (.0001)	0.9998 (.0001)	0.9996 (.0002)	0.9999 (.0001)	0.9998 (.0001)	0.9996 (.0003)
200	10	0.9233 (.0076)	0.9197 (.0063)	0.9172 (.0095)	0.9866 (.0023)	0.9824 (.0099)	0.9350 (.0299)	0.9963 (.0006)	0.9957 (.0010)	0.9948 (.0022)
		0.9804 (.0017)	0.9475 (.0165)	0.9380 (.0021)	0.9984 (.0008)	0.9761 (.0159)	0.9625 (.0036)	0.9985 (.0004)	0.9984 (.0005)	0.9980 (.0010)
	25	0.9899 (.0011)	0.9896 (.0014)	0.9866 (.0024)	0.9994 (.0003)	0.9992 (.0004)	0.9988 (.0007)	0.9998 (.0001)	0.9997 (.0002)	0.9995 (.0003)
		0.9996 (.0002)	0.9994 (.0003)	0.9991 (.0005)	0.9999 (.0001)	0.9998 (.0001)	0.9996 (.0002)	0.9999 (.0001)	0.9998 (.0001)	0.9996 (.0003)
	50	0.9233 (.0076)	0.9197 (.0063)	0.9172 (.0095)	0.9866 (.0023)	0.9824 (.0099)	0.9350 (.0299)	0.9963 (.0006)	0.9957 (.0010)	0.9948 (.0022)
		0.9804 (.0017)	0.9475 (.0165)	0.9380 (.0021)	0.9984 (.0008)	0.9761 (.0159)	0.9625 (.0036)	0.9985 (.0004)	0.9984 (.0005)	0.9980 (.0010)
	100	0.9899 (.0011)	0.9896 (.0014)	0.9866 (.0024)	0.9994 (.0003)	0.9992 (.0004)	0.9988 (.0007)	0.9998 (.0001)	0.9997 (.0002)	0.9995 (.0003)
		0.9996 (.0002)	0.9994 (.0003)	0.9991 (.0005)	0.9999 (.0001)	0.9998 (.0001)	0.9996 (.0002)	0.9999 (.0001)	0.9998 (.0001)	0.9996 (.0003)

Table 5-3: The congruence coefficients of the loading functions at $\tau^2 = 0.9$ averaged over 100 replications (standard deviations in parentheses).

		$\alpha_1(t)$				$\alpha_2(t)$			$\alpha_3(t)$	
N	J	$\sigma^2 = 0.5$	1	2	0.5	1	2	0.5	1	2
25	10	0.9336	0.9161	0.8444	0.9847	0.9590	0.8933	0.9943	0.9914	0.9833
		(.0300)	(.0163)	(.3076)	(.0136)	(.0272)	(.0314)	(.0034)	(.0073)	(.0363)
	25	0.9735	0.9459	0.9337	0.9950	0.9755	0.9593	0.9976	0.9961	0.9935
		(.0165)	(.0189)	(.0172)	(.0087)	(.0191)	(.0150)	(.0016)	(.0032)	(.0062)
	50	0.9878	0.9788	0.9823	0.9986	0.9964	0.9950	0.9990	0.9982	0.9956
		(.0035)	(.0067)	(.0088)	(.0012)	(.0029)	(.0043)	(.0009)	(.0020)	(.0047)
	100	0.9986	0.9962	0.9935	0.9991	0.9984	0.9971	0.9991	0.9984	0.9976
		(.0009)	(.0025)	(.0045)	(.0006)	(.0012)	(.0026)	(.0006)	(.0017)	(.0019)
50	10	0.9264	0.9172	0.9122	0.9791	0.9714	0.9393	0.9956	0.9938	0.9901
		(.0180)	(.0134)	(.0218)	(.0054)	(.0202)	(.0360)	(.0013)	(.0043)	(.0091)
	25	0.9768	0.9522	0.9365	0.9971	0.9763	0.9608	0.9982	0.9974	0.9966
		(.0124)	(.0204)	(.0049)	(.0060)	(.0170)	(.0088)	(.0009)	(.0014)	(.0026)
	50	0.9893	0.9876	0.9810	0.9991	0.9984	0.9969	0.9994	0.9988	0.9981
		(.0022)	(.0034)	(.0056)	(.0008)	(.0014)	(.0025)	(.0005)	(.0012)	(.0016)
	100	0.9993	0.9981	0.9969	0.9996	0.9992	0.9984	0.9995	0.9993	0.9982
		(.0005)	(.0011)	(.0021)	(.0003)	(.0007)	(.0012)	(.0003)	(.0005)	(.0016)
100	10	0.9243	0.9192	0.9154	0.9874	0.9493	0.9439	0.9961	0.9950	0.9940
		(.0158)	(.0072)	(.0163)	(.0028)	(.0147)	(.0347)	(.0010)	(.0017)	(.0035)
	25	0.9814	0.9502	0.9381	0.9983	0.9790	0.9620	0.9984	0.9981	0.9975
		(.0052)	(.0184)	(.0025)	(.0011)	(.0168)	(.0071)	(.0005)	(.0008)	(.0015)
	50	0.9897	0.9866	0.9851	0.9993	0.9990	0.9983	0.9997	0.9995	0.9991
		(.0016)	(.0024)	(.0038)	(.0005)	(.0006)	(.0013)	(.0002)	(.0004)	(.0008)
	100	0.9993	0.9990	0.9985	0.9998	0.9996	0.9992	0.9998	0.9996	0.9992
		(.0004)	(.0006)	(.0009)	(.0002)	(.0002)	(.0005)	(.0001)	(.0002)	(.0006)
200	10	0.9215	0.9205	0.9151	0.9721	0.9676	0.9252	0.9962	0.9957	0.9942
		(.0050)	(.0057)	(.0102)	(.0025)	(.0102)	(.0258)	(.0006)	(.0010)	(.0028)
	25	0.9824	0.9450	0.9385	0.9986	0.9728	0.9630	0.9985	0.9983	0.9981
		(.0016)	(.0139)	(.0024)	(.0006)	(.0167)	(.0035)	(.0004)	(.0005)	(.0009)
	50	0.9901	0.9894	0.9870	0.9994	0.9992	0.9990	0.9998	0.9998	0.9995
		(.0012)	(.0015)	(.0023)	(.0003)	(.0005)	(.0007)	(.0002)	(.0002)	(.0004)
	100	0.9996	0.9995	0.9992	0.9999	0.9998	0.9996	0.9999	0.9998	0.9996
		(.0002)	(.0003)	(.0005)	(.0001)	(.0001)	(.0003)	(.0001)	(.0001)	(.0004)

Table 5-4: The congruence coefficients of the latent variables at $\tau^2 = 0.2$ averaged over 100 replications (standard deviations in parentheses).

N	J	$\sigma^2 = 0.5$	γ_1			γ_2			γ_3		
			1	2	0.5	1	2	0.5	1	2	
25	10	0.9087	0.8540	0.7472	0.9222	0.8715	0.7608	0.9352	0.8954	0.8146	
		(.0458)	(.0526)	(.0938)	(.0385)	(.0536)	(.1084)	(.0340)	(.0423)	(.0781)	
	25	0.9445	0.9194	0.8658	0.9499	0.9394	0.8965	0.9513	0.9452	0.9070	
		(.0382)	(.0423)	(.0552)	(.0348)	(.0322)	(.0497)	(.0387)	(.0281)	(.0388)	
	50	0.9699	0.9509	0.9281	0.9675	0.9529	0.9334	0.9713	0.9609	0.9413	
		(.0238)	(.0297)	(.0394)	(.0267)	(.0378)	(.0439)	(.0259)	(.0262)	(.0359)	
	100	0.9717	0.9641	0.9549	0.9702	0.9675	0.9633	0.9731	0.9661	0.9576	
		(.0344)	(.0356)	(.0286)	(.0324)	(.0319)	(.0320)	(.0283)	(.0380)	(.0327)	
50	10	0.9199	0.8584	0.7621	0.9367	0.8758	0.7983	0.9397	0.9078	0.8295	
		(.0234)	(.0364)	(.0529)	(.0205)	(.0378)	(.0609)	(.0182)	(.0258)	(.0443)	
	25	0.9638	0.9356	0.8810	0.9685	0.9442	0.9034	0.9704	0.9566	0.9191	
		(.0159)	(.0181)	(.0324)	(.0151)	(.0185)	(.0305)	(.0157)	(.0168)	(.0230)	
	50	0.9770	0.9593	0.9359	0.9791	0.9703	0.9507	0.9805	0.9696	0.9523	
		(.0137)	(.0191)	(.0222)	(.0154)	(.0142)	(.0201)	(.0121)	(.0171)	(.0167)	
	100	0.9858	0.9781	0.9638	0.9858	0.9797	0.9686	0.9882	0.9820	0.9706	
		(.0112)	(.0128)	(.0172)	(.0137)	(.0113)	(.0181)	(.0103)	(.0118)	(.0147)	
100	10	0.9175	0.8548	0.7704	0.9405	0.8843	0.7859	0.9471	0.9037	0.8376	
		(.0187)	(.0292)	(.0410)	(.0125)	(.0229)	(.0440)	(.0122)	(.0198)	(.0300)	
	25	0.9660	0.9381	0.8901	0.9739	0.9512	0.9110	0.9749	0.9571	0.9241	
		(.0089)	(.0147)	(.0212)	(.0099)	(.0121)	(.0167)	(.0077)	(.0105)	(.0156)	
	50	0.9827	0.9674	0.9428	0.9855	0.9756	0.9568	0.9861	0.9751	0.9581	
		(.0062)	(.0093)	(.0114)	(.0068)	(.0077)	(.0113)	(.0063)	(.0076)	(.0103)	
	100	0.9873	0.9818	0.9698	0.9906	0.9856	0.9744	0.9901	0.9848	0.9759	
		(.0089)	(.0068)	(.0078)	(.0057)	(.0075)	(.0088)	(.0093)	(.0069)	(.0072)	
200	10	0.9230	0.8589	0.7659	0.9428	0.8959	0.7982	0.9507	0.9086	0.8403	
		(.0115)	(.0189)	(.0293)	(.0081)	(.0145)	(.0334)	(.0087)	(.0131)	(.0211)	
	25	0.9698	0.9418	0.8909	0.9767	0.9550	0.9140	0.9783	0.9588	0.9251	
		(.0043)	(.0095)	(.0153)	(.0058)	(.0079)	(.0108)	(.0045)	(.0073)	(.0117)	
	50	0.9840	0.9710	0.9450	0.9874	0.9781	0.9586	0.9880	0.9784	0.9600	
		(.0040)	(.0049)	(.0080)	(.0039)	(.0041)	(.0062)	(.0039)	(.0036)	(.0065)	
	100	0.9911	0.9841	0.9726	0.9925	0.9875	0.9779	0.9928	0.9876	0.9783	
		(.0032)	(.0043)	(.0044)	(.0033)	(.0047)	(.0044)	(.0030)	(.0040)	(.0042)	

Table 5-5: The congruence coefficients of the latent variables at $\tau^2 = 0.6$ averaged over 100 replications (standard deviations in parentheses).

		Υ_1				Υ_2			Υ_3	
N	J	$\sigma^2 = 0.5$	1	2	0.5	1	2	0.5	1	2
25	10	0.9015	0.8341	0.7474	0.9234	0.8634	0.7706	0.9303	0.8883	0.8199
		(.0420)	(.0660)	(.0891)	(.0423)	(.0697)	(.0851)	(.0399)	(.0453)	(.0722)
	25	0.9452	0.9165	0.8676	0.9632	0.9302	0.8845	0.9613	0.9302	0.9059
		(.0339)	(.0420)	(.0539)	(.0234)	(.0361)	(.0522)	(.0265)	(.0434)	(.0389)
	50	0.9675	0.9486	0.9273	0.9734	0.9620	0.9327	0.9755	0.9623	0.9397
		(.0241)	(.0346)	(.0394)	(.0201)	(.0252)	(.0406)	(.0191)	(.0274)	(.0356)
	100	0.9730	0.9649	0.9520	0.9725	0.9699	0.9566	0.9731	0.9699	0.9654
		(.0237)	(.0271)	(.0301)	(.0305)	(.0350)	(.0325)	(.0270)	(.0253)	(.0227)
50	10	0.9186	0.8490	0.7594	0.9361	0.8866	0.7945	0.9423	0.9029	0.8349
		(.0221)	(.0451)	(.0679)	(.0197)	(.0280)	(.0553)	(.0186)	(.0278)	(.0449)
	25	0.9614	0.9321	0.8795	0.9661	0.9441	0.9059	0.9684	0.9535	0.9197
		(.0149)	(.0215)	(.0313)	(.0186)	(.0228)	(.0239)	(.0155)	(.0184)	(.0269)
	50	0.9735	0.9626	0.9357	0.9774	0.9675	0.9517	0.9808	0.9711	0.9506
		(.0160)	(.0158)	(.0184)	(.0184)	(.0181)	(.0219)	(.0144)	(.0126)	(.0201)
	100	0.9839	0.9775	0.9651	0.9861	0.9774	0.9719	0.9862	0.9805	0.9719
		(.0160)	(.0138)	(.0137)	(.0098)	(.0148)	(.0143)	(.0118)	(.0121)	(.0138)
100	10	0.9188	0.8604	0.7632	0.9408	0.8925	0.7997	0.9435	0.9061	0.8402
		(.0169)	(.0241)	(.0436)	(.0123)	(.0225)	(.0395)	(.0129)	(.0182)	(.0300)
	25	0.9663	0.9381	0.8908	0.9752	0.9508	0.9094	0.9749	0.9575	0.9209
		(.0097)	(.0140)	(.0199)	(.0086)	(.0117)	(.0187)	(.0069)	(.0100)	(.0184)
	50	0.9803	0.9680	0.9405	0.9856	0.9743	0.9559	0.9851	0.9744	0.9584
		(.0087)	(.0093)	(.0146)	(.0065)	(.0072)	(.0099)	(.0062)	(.0080)	(.0092)
	100	0.9887	0.9806	0.9703	0.9901	0.9828	0.9760	0.9896	0.9844	0.9766
		(.0059)	(.0086)	(.0094)	(.0075)	(.0092)	(.0068)	(.0068)	(.0097)	(.0070)
200	10	0.9212	0.8606	0.7665	0.9426	0.8963	0.7970	0.9505	0.9075	0.8444
		(.0105)	(.0196)	(.0308)	(.0077)	(.0150)	(.0311)	(.0076)	(.0139)	(.0212)
	25	0.9691	0.9403	0.8914	0.9770	0.9548	0.9120	0.9775	0.9599	0.9253
		(.0046)	(.0083)	(.0155)	(.0052)	(.0088)	(.0122)	(.0044)	(.0062)	(.0100)
	50	0.9833	0.9702	0.9454	0.9867	0.9778	0.9589	0.9884	0.9781	0.9598
		(.0045)	(.0052)	(.0078)	(.0046)	(.0042)	(.0062)	(.0027)	(.0048)	(.0061)
	100	0.9912	0.9840	0.9719	0.9930	0.9874	0.9776	0.9926	0.9870	0.9784
		(.0026)	(.0049)	(.0049)	(.0028)	(.0039)	(.0041)	(.0030)	(.0044)	(.0050)

Table 5-6: The congruence coefficients of the latent variables at $\tau^2 = 0.9$ averaged over 100 replications (standard deviations in parentheses).

		Υ_1				Υ_2			Υ_3	
N	J	$\sigma^2 = 0.5$	1	2	0.5	1	2	0.5	1	2
25	10	0.8994	0.8385	0.7417	0.9189	0.8684	0.7767	0.9283	0.8832	0.8125
		(.0472)	(.0605)	(.0913)	(.0382)	(.0462)	(.0788)	(.0321)	(.0534)	(.0746)
	25	0.9535	0.9234	0.8716	0.9576	0.9371	0.9017	0.9588	0.9423	0.9112
		(.0321)	(.0413)	(.0452)	(.0285)	(.0380)	(.0424)	(.0323)	(.0363)	(.0419)
	50	0.9582	0.9545	0.9258	0.9676	0.9628	0.9335	0.9719	0.9658	0.9374
		(.0367)	(.0256)	(.0451)	(.0288)	(.0250)	(.0366)	(.0323)	(.0203)	(.0318)
	100	0.9756	0.9639	0.9496	0.9735	0.9698	0.9549	0.9744	0.9679	0.9578
		(.0240)	(.0341)	(.0301)	(.0267)	(.0275)	(.0360)	(.0284)	(.0274)	(.0296)
50	10	0.9132	0.8463	0.7594	0.9321	0.8799	0.7984	0.9444	0.9021	0.8238
		(.0289)	(.0404)	(.0676)	(.0227)	(.0344)	(.0599)	(.0211)	(.0303)	(.0434)
	25	0.9603	0.9326	0.8817	0.9696	0.9444	0.9074	0.9764	0.9516	0.9203
		(.0168)	(.0220)	(.0326)	(.0177)	(.0191)	(.0322)	(.0090)	(.0170)	(.0248)
	50	0.9730	0.9625	0.9397	0.9779	0.9680	0.9532	0.9772	0.9707	0.9558
		(.0163)	(.0155)	(.0175)	(.0184)	(.0191)	(.0146)	(.0167)	(.0131)	(.0148)
	100	0.9816	0.9789	0.9655	0.9845	0.9791	0.9668	0.9862	0.9783	0.9679
		(.0148)	(.0128)	(.0168)	(.0179)	(.0165)	(.0169)	(.0118)	(.0157)	(.0163)
100	10	0.9182	0.8587	0.7676	0.9399	0.8784	0.8025	0.9479	0.9053	0.8418
		(.0171)	(.0247)	(.0439)	(.0133)	(.0233)	(.0446)	(.0122)	(.0184)	(.0261)
	25	0.9677	0.9369	0.8911	0.9761	0.9529	0.9076	0.9756	0.9574	0.9236
		(.0094)	(.0139)	(.0226)	(.0069)	(.0130)	(.0181)	(.0071)	(.0104)	(.0149)
	50	0.9814	0.9644	0.9429	0.9849	0.9749	0.9554	0.9845	0.9761	0.9573
		(.0067)	(.0115)	(.0119)	(.0068)	(.0082)	(.0104)	(.0093)	(.0071)	(.0117)
	100	0.9860	0.9822	0.9688	0.9910	0.9857	0.9750	0.9897	0.9841	0.9749
		(.0081)	(.0078)	(.0089)	(.0063)	(.0070)	(.0090)	(.0076)	(.0077)	(.0093)
200	10	0.9216	0.8602	0.7613	0.9408	0.8920	0.7963	0.9499	0.9082	0.8362
		(.0121)	(.0197)	(.0318)	(.0085)	(.0144)	(.0315)	(.0077)	(.0127)	(.0202)
	25	0.9687	0.9405	0.8915	0.9769	0.9541	0.9116	0.9784	0.9591	0.9258
		(.0057)	(.0080)	(.0124)	(.0050)	(.0074)	(.0130)	(.0039)	(.0060)	(.0103)
	50	0.9842	0.9703	0.9443	0.9871	0.9769	0.9595	0.9878	0.9780	0.9586
		(.0036)	(.0045)	(.0075)	(.0040)	(.0042)	(.0071)	(.0037)	(.0050)	(.0066)
	100	0.9901	0.9841	0.9720	0.9922	0.9875	0.9773	0.9929	0.9873	0.9785
		(.0062)	(.0042)	(.0043)	(.0036)	(.0051)	(.0048)	(.0033)	(.0042)	(.0049)

Table 5-7: The mean squared errors of the path coefficient b_1 (standard deviations in parentheses).

N	J	$\sigma^2 = 0.5$			1			2		
		$\tau^2 = 0.2$	0.6	0.9	0.2	0.6	0.9	0.2	0.6	0.9
25	10	0.0182 (.0358)	0.0281 (.0456)	0.0491 (.0607)	0.0407 (.0504)	0.0632 (.1329)	0.0515 (.0839)	0.1263 (.2096)	0.1278 (.2818)	0.0667 (.1005)
	25	0.0089 (.0121)	0.0214 (.0410)	0.0427 (.0662)	0.0181 (.0282)	0.0362 (.0595)	0.0373 (.0561)	0.0379 (.0464)	0.0536 (.1227)	0.0493 (.0697)
	50	0.0100 (.0141)	0.0183 (.0238)	0.0299 (.0394)	0.0107 (.0165)	0.0261 (.0453)	0.0321 (.0367)	0.0122 (.0191)	0.0213 (.0306)	0.0394 (.0522)
	100	0.0083 (.0141)	0.0276 (.0433)	0.0396 (.0511)	0.0099 (.0148)	0.0163 (.0290)	0.0425 (.0679)	0.0098 (.0161)	0.0179 (.0277)	0.0430 (.0614)
50	10	0.0141 (.0206)	0.0164 (.0252)	0.0210 (.0306)	0.0319 (.0313)	0.0216 (.0298)	0.0259 (.0333)	0.0846 (.0516)	0.0707 (.1327)	0.0332 (.0473)
	25	0.0049 (.0063)	0.0112 (.0153)	0.0167 (.0212)	0.0074 (.0108)	0.0123 (.0212)	0.0205 (.0245)	0.0233 (.0229)	0.0229 (.0344)	0.0183 (.0197)
	50	0.0039 (.0066)	0.0091 (.0164)	0.0186 (.0226)	0.0052 (.0079)	0.0157 (.0313)	0.0145 (.0216)	0.0100 (.0114)	0.0112 (.0161)	0.0181 (.0286)
	100	0.0035 (.0048)	0.0084 (.0122)	0.0139 (.0241)	0.0045 (.0059)	0.0088 (.0117)	0.0183 (.0250)	0.0053 (.0067)	0.0077 (.0099)	0.0166 (.0227)
100	10	0.0126 (.0114)	0.0107 (.0140)	0.0113 (.0168)	0.0339 (.0225)	0.0238 (.0262)	0.0137 (.0220)	0.0798 (.0384)	0.0545 (.0465)	0.0158 (.0190)
	25	0.0036 (.0051)	0.0061 (.0102)	0.0096 (.0149)	0.0078 (.0081)	0.0088 (.0134)	0.0099 (.0125)	0.0194 (.0134)	0.0139 (.0167)	0.0143 (.0187)
	50	0.0019 (.0026)	0.0040 (.0056)	0.0085 (.0128)	0.0047 (.0068)	0.0050 (.0062)	0.0075 (.0114)	0.0070 (.0076)	0.0098 (.0138)	0.0077 (.0106)
	100	0.0023 (.0039)	0.0037 (.0052)	0.0083 (.0111)	0.0028 (.0038)	0.0071 (.0096)	0.0120 (.0133)	0.0029 (.0040)	0.0040 (.0057)	0.0096 (.0121)
200	10	0.0096 (.0067)	0.0085 (.0093)	0.0051 (.0074)	0.0311 (.0150)	0.0215 (.0155)	0.0072 (.0099)	0.0868 (.0317)	0.0465 (.0235)	0.0155 (.0163)
	25	0.0022 (.0027)	0.0040 (.0060)	0.0041 (.0054)	0.0071 (.0060)	0.0053 (.0069)	0.0047 (.0106)	0.0205 (.0107)	0.0125 (.0098)	0.0069 (.0081)
	50	0.0014 (.0019)	0.0022 (.0034)	0.0046 (.0069)	0.0025 (.0033)	0.0038 (.0050)	0.0036 (.0054)	0.0055 (.0054)	0.0041 (.0054)	0.0054 (.0071)
	100	0.0011 (.0016)	0.0018 (.0025)	0.0038 (.0053)	0.0012 (.0016)	0.0024 (.0039)	0.0039 (.0059)	0.0019 (.0020)	0.0034 (.0052)	0.0043 (.0071)

Table 5-8: The mean squared errors of the path coefficient b_2 (standard deviations in parentheses).

N	J	$\sigma^2 = 0.5$			1			2		
		$\tau^2 = 0.2$	0.6	0.9	0.2	0.6	0.9	0.2	0.6	0.9
25	10	0.0195 (.0255)	0.0327 (.0394)	0.0407 (.0570)	0.0221 (.0304)	0.0336 (.0427)	0.0439 (.0604)	0.0597 (.0901)	0.0499 (.0747)	0.0570 (.0781)
	25	0.0172 (.0227)	0.0218 (.0312)	0.0434 (.0546)	0.0217 (.0293)	0.0337 (.0484)	0.0315 (.0391)	0.0257 (.0413)	0.0365 (.0646)	0.0542 (.0911)
	50	0.0140 (.0186)	0.0291 (.0445)	0.0294 (.0317)	0.0184 (.0365)	0.0274 (.0373)	0.0531 (.0686)	0.0190 (.0231)	0.0308 (.0422)	0.0366 (.0466)
	100	0.0177 (.0288)	0.0252 (.0347)	0.0372 (.0501)	0.0123 (.0155)	0.0241 (.0330)	0.0397 (.0555)	0.0142 (.0176)	0.0264 (.0349)	0.0482 (.0685)
50	10	0.0110 (.0145)	0.0174 (.0237)	0.0203 (.0283)	0.0167 (.0201)	0.0131 (.0188)	0.0213 (.0300)	0.0299 (.0389)	0.0218 (.0400)	0.0236 (.0290)
	25	0.0056 (.0073)	0.0131 (.0214)	0.0199 (.0357)	0.0090 (.0107)	0.0121 (.0194)	0.0208 (.0287)	0.0118 (.0155)	0.0144 (.0181)	0.0223 (.0250)
	50	0.0067 (.0107)	0.0126 (.0168)	0.0211 (.0277)	0.0070 (.0101)	0.0122 (.0169)	0.0232 (.0351)	0.0063 (.0083)	0.0146 (.0212)	0.0183 (.0211)
	100	0.0066 (.0085)	0.0116 (.0165)	0.0230 (.0305)	0.0060 (.0073)	0.0123 (.0164)	0.0157 (.0211)	0.0080 (.0150)	0.0141 (.0169)	0.0140 (.0170)
100	10	0.0050 (.0075)	0.0067 (.0092)	0.0094 (.0120)	0.0120 (.0144)	0.0087 (.0149)	0.0095 (.0132)	0.0227 (.0247)	0.0175 (.0247)	0.0124 (.0193)
	25	0.0039 (.0064)	0.0068 (.0091)	0.0100 (.0141)	0.0052 (.0068)	0.0066 (.0093)	0.0087 (.0115)	0.0076 (.0086)	0.0101 (.0122)	0.0104 (.0137)
	50	0.0019 (.0028)	0.0062 (.0081)	0.0087 (.0098)	0.0031 (.0049)	0.0054 (.0067)	0.0080 (.0120)	0.0046 (.0060)	0.0083 (.0121)	0.0080 (.0113)
	100	0.0031 (.0045)	0.0057 (.0082)	0.0107 (.0140)	0.0037 (.0052)	0.0059 (.0078)	0.0092 (.0116)	0.0026 (.0035)	0.0065 (.0092)	0.0084 (.0117)
200	10	0.0035 (.0036)	0.0045 (.0063)	0.0048 (.0068)	0.0076 (.0075)	0.0055 (.0075)	0.0059 (.0085)	0.0173 (.0152)	0.0077 (.0088)	0.0060 (.0082)
	25	0.0017 (.0024)	0.0030 (.0041)	0.0049 (.0078)	0.0028 (.0046)	0.0034 (.0048)	0.0054 (.0071)	0.0057 (.0068)	0.0048 (.0069)	0.0054 (.0080)
	50	0.0015 (.0022)	0.0027 (.0032)	0.0044 (.0065)	0.0019 (.0025)	0.0039 (.0050)	0.0043 (.0075)	0.0028 (.0029)	0.0029 (.0038)	0.0045 (.0057)
	100	0.0019 (.0027)	0.0029 (.0040)	0.0055 (.0080)	0.0017 (.0026)	0.0036 (.0043)	0.0038 (.0044)	0.0017 (.0024)	0.0029 (.0042)	0.0040 (.0057)

Chapter 6

Empirical Examples

In this chapter, the usefulness of functional GSCA is demonstrated by applying functional GSCA to two real data sets. In the first example, the relationships among three functional responses are examined. In the second example, the relationships among a functional response and two multivariate responses are investigated.

6.1 The Movie Data

This example pertains to a movie data set that was obtained by combining two sets of data: the movie box office revenue data and the movie advertising spending data. The movie box office revenue data were collected from a publicly available movie box office database (www.the-numbers.com) and the movie advertising spending data were provided by a commercial advertising consulting company. In this data set, 152 movies released in the US from 2006 to 2007 were measured on their weekly box office revenue for 10 consecutive weeks from the release and weekly advertising spending on three different media, network televisions, newspapers, and national spot radios, for 15 consecutive weeks, more specifically, 5 weeks before the

release and 10 weeks from the release. Figure 6-1 displays the values of the four functional responses of the 152 movies varying over the weeks.

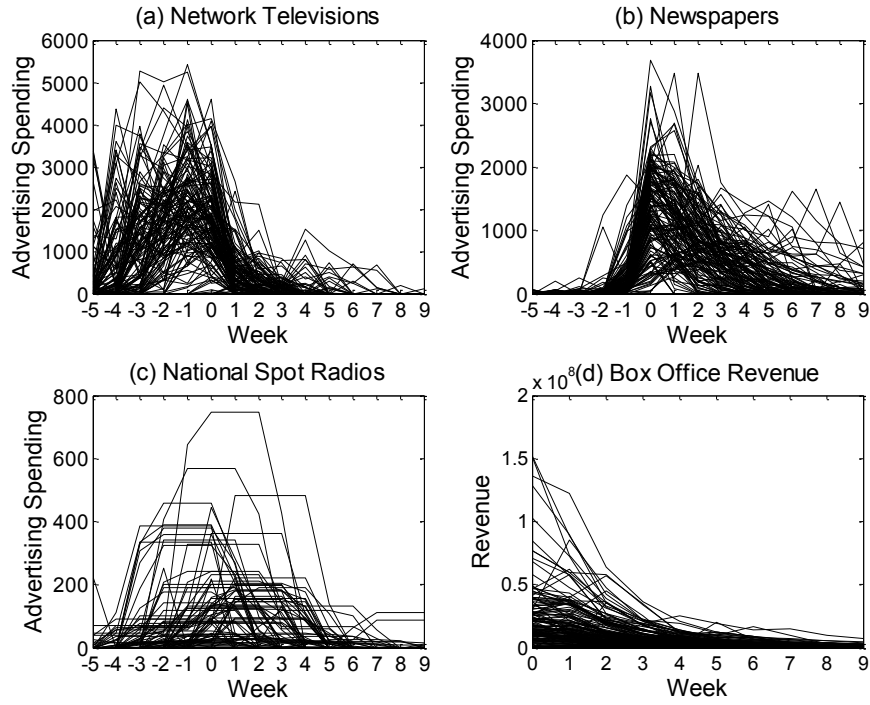


Figure 6-1: Weekly advertising spending on (a) network televisions, (b) newspapers, and (c) national spot radios over 15 weeks and (d) box office revenue over 10 weeks of the 152 movies. Each line represents each movie. Week 0 indicates the week when each movie was released.

The aim of collecting this data set was to examine how the advertising spending on different media affected the movie box office revenue. Therefore the structural model shown in Figure 6-2 was hypothesized for the analysis of the data set. The optimal values of the smoothing parameters λ and ρ were determined by five-fold cross-validation, in which 13 different values for

each smoothing parameter, $[10^{-10}, 10^{-9}, \dots, 10^1, 10^2]$ were tested. The resultant prediction errors across all possible combinations of the values of λ and ρ reached their minimum at $\lambda = 10$ and $\rho = 10^{-8}$. These optimal smoothing parameter values were used in the subsequent analysis as well as in bootstrapping.

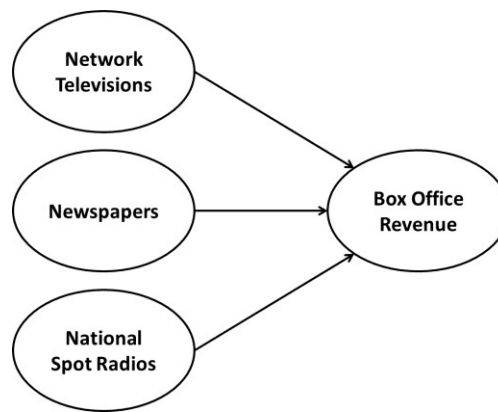


Figure 6-2: The structural model used for analyzing the movie data.

The FIT of this hypothesized model was 0.7523, which indicates that this model captured 75.23% of the variation in data functions and endogenous latent variables. Figure 6-3 displays the estimated loading functions, which show overall patterns of change of the four functional responses over time. It can be observed that the advertising spending on network televisions started earlier than five weeks before the release of the movies, increased at the beginning until it peaked at around two weeks before the release, and tended to drain away later on until it touched zero at around three weeks after the release to maintain. On the other hand, the advertising spending on

newspapers started later at around three weeks before the release, increased until one week after the release, and then decreased gradually. The 95% point-wise confidence interval of this loading function shows that there was a substantial variation on the peak time of the advertising spending on newspapers, which spread from the release week to two weeks after the release.

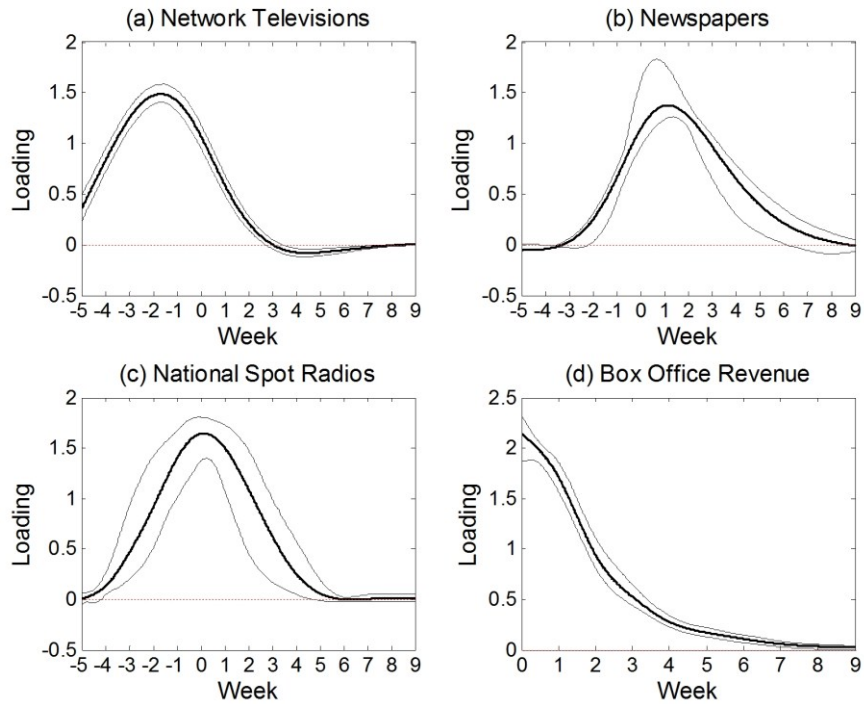


Figure 6-3: The estimated loading functions of the movie data for the advertising spending on the four media (thick solid lines) and their 95% point-wise confidence intervals (thin solid lines) obtained under $\lambda = 10$ and $\rho = 10^{-8}$.

The advertising spending on national spot radios showed a similar pattern of change over time as on newspapers. However, the advertising spending on national spot radios manifested a much larger variation as indicated by its wider 95% point-wise confidence interval. The box office revenue reached its maximum at the week of release and then decreased exponentially.

Table 6-1 presents the estimated path coefficients and their 95% confidence intervals. The estimated path coefficient of the movie box office revenue on the advertising spending on network televisions was 0.33. In other words, as the amount of the advertising spending on network televisions, which changed over time in the fashion as shown in Figure 6-3(a), increased by 1, the movie box office revenue tended to increase by 0.33. Its 95% confidence interval indicates that this effect was statistically significant. Similarly, the advertising spending on newspapers had a significant positive effect on the movie box office revenue with the estimated path coefficient of 0.37. The effect of the advertising spending on national spot radios was also significant and positive as indicated by its path coefficient value of 0.14.

Figure 6-4 displays the estimated weight functions to obtain the latent variable scores. Weight functions indicate which time periods are crucial for defining latent variable in such a way that the latent variables can capture the relationships among themselves and explain the data functions

simultaneously as much as possible. It can be observed that the weight functions looked similar to the loading functions.

Table 6-1: The estimates of the path coefficients and their 95% bootstrap confidence intervals of the movie data obtained from functional GSCA.

Path		Estimate	95% Confidence Interval	
From	To		Lower Limit	Upper Limit
Network Televisions	Box Office Revenue	0.3282	0.1087	0.5208
Newspapers	Box Office Revenue	0.3717	0.2042	0.5328
National Spot Radios	Box Office Revenue	0.1442	0.0071	0.2945

In Figure 6-4, we can see that the movies that had more advertising spending on network televisions for the first eight weeks (from five weeks before the release until three weeks after the release) tended to have higher scores of the corresponding latent variable. On newspapers, the advertising spending from three weeks before the release to seven weeks after the release was crucial to explain the data functions and to examine the effect of this variable on the movie box office revenue. On national spot radios, as the advertising spending from four weeks before the release to four weeks after the release increased, the corresponding latent variable scores tended to be higher. For the box office revenue, the weeks closer to the release time were more important to explain the data functions as well as the relationships with other variables.

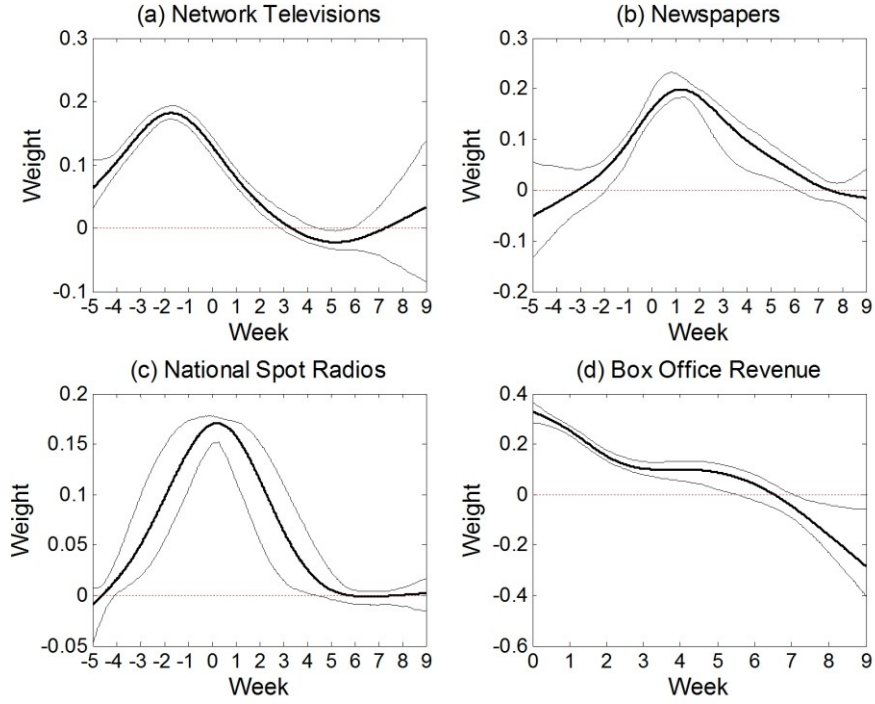


Figure 6-4: The estimated weight functions of the movie data for the advertising spending on the four media (thick solid lines) and their 95% point-wise confidence intervals (thin solid lines) obtained under $\lambda = 10$ and $\rho = 10^{-8}$.

6.2 The Gait Data

The second example concerns the gait data set (Yogev et al., 2005), which is available on the PhysioNet website (<http://www.physionet.org/physiobank/database/gaitpdb/>). In this example, a subset of the data set was analyzed, in which 23 patients diagnosed as having idiopathic Parkinson's disease were measured on the total force under the left foot (in Newtons) for eight seconds while they walked at their usual pace.

The total force under each foot was measured at 10 Hz per second, which yielded 80 time points. Figure 6-5 displays the total force under the left foot changing over eight seconds for 23 patients. Since patients started to measure their total force under each foot at different phase of walking, the data functions were aligned to be in the same phase by using the continuous registration method (Ramsay, Hooker, & Graves, 2009, Chapter 8; Ramsay & Silverman, 2005, Chapter 7). Besides the total force under the left foot, the physical size of body was measured by two observed variables, height (in centimeters) and weight (in kilograms). In addition, the severity of Parkinson's disease was measured by two scales – the Hoehn and Yahr staging scale (HY; Hoehn & Yahr, 1967) and unified Parkinson's disease rating scale (UPDRS; Fahn, Elton, & members of the UPDRS Development Committee, 1987). Higher values of these scales indicate more severe cognitive impairment.

This data set is an example of typical functional data in which variables are measured at a large number of time points for a small number of subjects. In this example, the total force under the left foot was measured over 80 time points, the number of which was much greater than the number of subjects, 23. In addition, this data set contains a functional variable, the total force under the foot, as well as two multivariate variables, the physical size of body and the severity of Parkinson's disease.

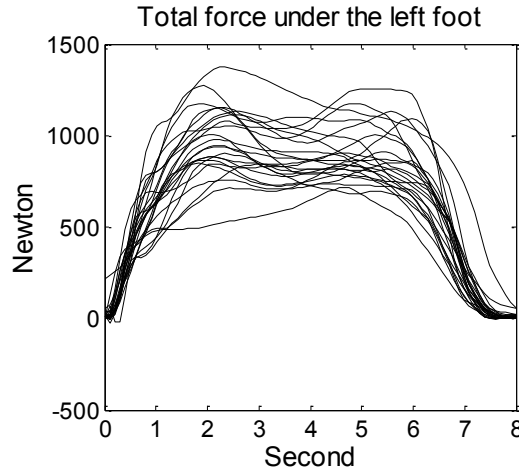


Figure 6-5: The total force under the left foot measured for 8 seconds from 23 Parkinson's disease patients.

In order to examine the effect of the severity of Parkinson's disease on the total force under the left foot with controlling for the effect of the physical size of body that presumably has an influence on the total force under the left foot, the structural model shown in Figure 6-6 was specified, in which both exogenous latent variables have multivariate indicators and the endogenous latent variable has a functional indicator. The optimal values of the smoothing parameters λ and ρ were determined by five-fold cross-validation, in which each of the smoothing parameters was varied at 10 different values, $[10^0, 10^1, \dots, 10^8, 10^9]$. The resultant optimal smoothing parameter values associated with the minimum prediction error were $\lambda = 10^5$ and $\rho = 10^2$, which were used in the further analysis and bootstrapping.

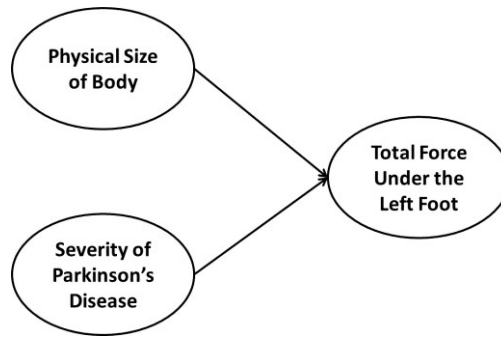


Figure 6-6: The structural model used for analyzing the gait data.

The FIT value of this hypothesized model was 0.7441, i.e., this model explained 74.41% of the variation in the observed variables/functions and endogenous latent variables. Table 6-2 presents the estimated loadings and weights of the two exogenous latent variables on their indicator variables. It can be observed that the loadings and weights on height, weight, and UPDRS were positive and significant. However, the loading and weight on HY were not significant. The Hoehn and Yahr staging scale has the values 1, 1.5, 2, 2.5, 3, 4, and 5 that indicate different stages of progress in the symptoms of Parkinson's disease. In this data set, the 23 patients were belonging to the stages 2, 2.5, or 3 only. Therefore the variation in this indicator variable, HY, was relatively small and hence, HY had no significant contribution to explaining the effect of the severity of Parkinson's disease on the total force under the left foot.

The estimated loading and weight functions of the total force under the left foot under $\lambda = 10^5$ and $\rho = 10^2$ are depicted in Figure 6-7. By

examining the estimated loading function, it can be observed that the patients' total force under the left foot increased sharply for the first two seconds, decreased slowly for the next two and a half seconds, increased a bit for the next two seconds, and then vanished out. The estimated weight function manifested a similar shape of change over time, i.e., the time points associated with higher loading function values were more crucial for defining the latent variable in such a way that it can explain its data function as well as its relationship with other latent variables as much as possible.

Table 6-2: The estimated loadings and weights of the physical size of body and the severity of Parkinson's disease on their indicator variables.

Latent Variable	Indicator Variable	Loading	95% Confidence Interval		Weight	95% Confidence Interval	
			Lower Limit	Upper Limit		Lower Limit	Upper Limit
Physical Size of Body	Height	0.8991	0.8188	0.9586	0.4545	0.3217	0.5838
	Weight	0.9478	0.8827	0.9833	0.6239	0.5040	0.7309
Severity of Parkinson's Disease	HY	0.7870	-0.0392	0.9333	0.5731	-0.0045	0.6424
	UPDRS	0.8397	0.7878	0.9796	0.6565	0.5034	0.9800

The estimated path coefficients and their 95% confidence intervals are given in Table 6-3. As expected, the physical size of body had a positive and significant effect on the total force under the left foot. The bigger a patient was in his physical size, the more force was exerted under the left foot. On the other hand, the severity of Parkinson's disease had a negative and significant effect on the total force under the left foot. This indicates that as a

patient progressed into a more severe stage of Parkinson's disease, the total force exerted under the left foot became weaker when controlling for the physical size of body.

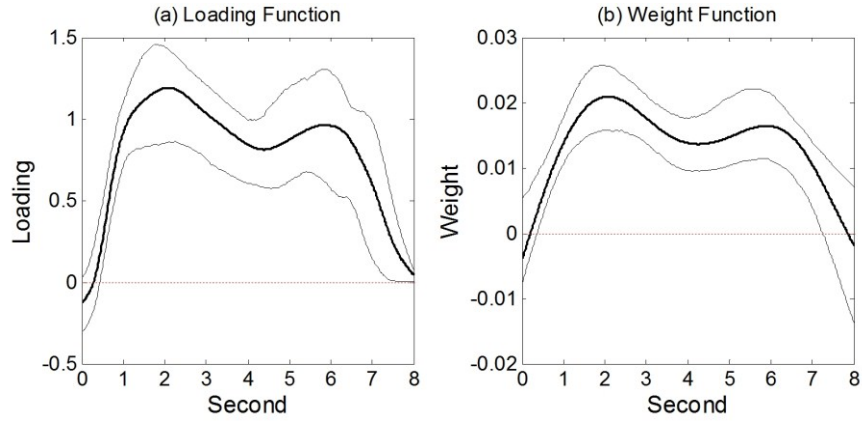


Figure 6-7: The estimated (a) loading function and (b) weight function of the total force under the left foot (thick solid lines) and their 95% point-wise confidence intervals (thin solid lines) obtained under $\lambda = 10^5$ and $\rho = 10^2$.

Table 6-3: The estimates of the path coefficients and their 95% bootstrap confidence intervals of the gait data obtained from functional GSCA.

Path		Estimate	95% Confidence Interval	
From	To		Lower Limit	Upper Limit
Physical Size of Body	Total Force Under The Left Foot	0.6264	0.3733	0.8369
Severity of Parkinson's Disease	Total Force Under The Left Foot	-0.4475	-0.6783	-0.1449

In this chapter, the applicability and usefulness of functional GSCA was demonstrated by two empirical examples. The first example was chosen to illustrate that functional GSCA can analyze the relationships among multiple functional responses. For the movie box office revenue and advertising spending data, functional GSCA revealed the advertising spending on the three media had positive effects on the movie box office revenue while uncovering the temporal variations, or the shapes of change of the advertising spending on the three different media, as well as the movie box office revenue over weeks.

The second example was included to demonstrate that functional GSCA is of use for the analysis of data that involve both functional and multivariate responses. For the gait data, the total force under the foot was a functional response whereas other variables were multivariate. Functional GSCA revealed that the total force under the left foot of a patient tended to decrease, as his Parkinson's disease progressed into a more severe stage, when controlling for the effect of his physical size of body. At the same time, functional GSCA uncovered the M-shaped temporal variation of the total force under the left foot.

Chapter 7

Summary and Discussion

In this thesis, functional GSCA was proposed for the analysis of functional data by integrating the original GSCA with penalized least squares spline smoothing into a unified framework. Unlike the previous approaches for analyzing functional data in the framework of SEM, which ignores the temporal variation in functional data, or allows only specific relationships among observed variables to be examined, functional GSCA enables to analyze a variety of directional relationships among multiple functional responses via latent variables without losing the temporal variation in data.

The usefulness of functional GSCA was demonstrated by using both synthetic and real data sets. The Monte Carlo study discussed in Chapter 6 demonstrated that functional GSCA worked reasonably well under various conditions. The two empirical examples given in Chapter 7 illustrated that functional GSCA could examine directional relationships among multiple latent variables while revealing the overall shape of change of each functional response. The two examples also showed that functional GSCA could accommodate the situations where all responses were functional as well as the cases where functional and multivariate responses were mixed.

Functional GSCA can be regarded as a functional analogue to latent trajectory models, also known as latent growth curve models and hierarchical linear models (e.g., Bollen & Curran, 2006; Browne & Du Toit, 1991; Duncan, Duncan, & Strycker, 2006; McArdle, 1986; Meredith & Tisak, 1990; Rao, 1958; Raudenbush, 2001; Raudenbush & Bryk, 2002), for the analysis of longitudinal data. Latent trajectory models assume that individual trajectories on a variable have a common shape of change, such as linear and quadratic, but different values of the parameters, usually intercepts and slopes. Likewise, functional GSCA assumes that individual functions on a variable have a common shape (a loading function) but different values of amplitude (latent variable scores). In latent trajectory models, directional relationships among parameters (usually intercepts and slopes) are to be examined. Similarly, functional GSCA investigates various directional relationships among latent variables. However, functional GSCA is distinguished from latent trajectory models in the following respects. Unlike latent trajectory models that assume a parametric shape of change on a variable, such as linear and quadratic, functional GSCA does not make any assumptions on the shape of the functional responses except that they are smooth. In addition, latent trajectory models estimate the distributions (usually means and covariances) of latent variables such as intercepts and slopes whereas functional GSCA estimates the scores of latent variables themselves. Moreover, unlike latent trajectory models that mainly concern

longitudinal data repeatedly measured over time, functional GSCA can deal with functional data measured over any continuum not to mention time.

Although functional GSCA is of use for the analysis of functional data, it may produce absurd results unless the characteristics of data are carefully considered. Even if functional responses of multiple subjects on a variable share a common shape of change, they can still exhibit two types of variation: amplitude variation and phase variation (Ramsay et al., 2009, Chapter 8; Ramsay & Silverman, 2005, Chapter 7). Figure 7-1 displays these two types of variation in a set of synthetic curves. When researchers collect functional data, they usually look like Figure 7-1(a), in which functions vary in both amplitude and phase. If the curves are aligned to have zero value at time = 0, i.e., the variation in phase is eliminated, the curves vary only in their amplitude as shown in Figure 7-1(b). On the other hand, if the curves are scaled to have the same amplitude, i.e., the variation in amplitude is eliminated, the curves vary only in their phase as shown in Figure 7-1(c). Functional GSCA focuses on the amplitude variation ignoring the phase variation. In other words, functional GSCA assumes that data functions are already aligned or registered to be in the same phase as in Figure 7-1(b). Therefore data functions should be preprocessed with a proper registration method especially when they manifest a considerable amount of variation in phase. Refer to Ramsay and Silverman (2005, Chapter 7) for a comprehensive discussion of curve registration.

In addition, functional GSCA may not work when data functions exhibit different shapes of change across subjects. This happens when a sample consists of heterogeneous groups of subjects involving different shapes of trajectories. For example, Wiesner and Windle (2004) studied adolescent delinquency trajectories and revealed six different trajectory groups: rare offenders, moderate late peakers, high late peakers, decreasers, moderate-level chronics, and high-level chronics. These six trajectories cannot be modeled by a single representative curve, or a loading function, as in functional GSCA. One might think that registering data could resolve this problem to some extent. However, in some cases, researchers are reluctant to register data because the phase variation in data reflects an important characteristic of data that should not be ignored. A promising way of extending functional GSCA to uncover such cluster-level heterogeneity is to combine functional GSCA with a clustering method. Hwang, Desarbo, et al. (2007) already showed that the original GSCA can be nicely combined with fuzzy clustering to deal with heterogeneous groups of subjects. Functional GSCA can be readily extended to fuzzy clusterwise functional GSCA in a similar way. Another promising approach is a multilevel extension of functional GSCA, which can be used when one is interested in examining differences in trajectories as well as in path coefficients across already existing groups, such as gender, geographical regions, treatment conditions, etc. The original GSCA has been extended to multilevel GSCA (Hwang,

Takane, & Malhotra, 2007), in which loadings and path coefficients are allowed to vary across different groups. Similarly, functional GSCA can be generalized to incorporate such multilevel structures.

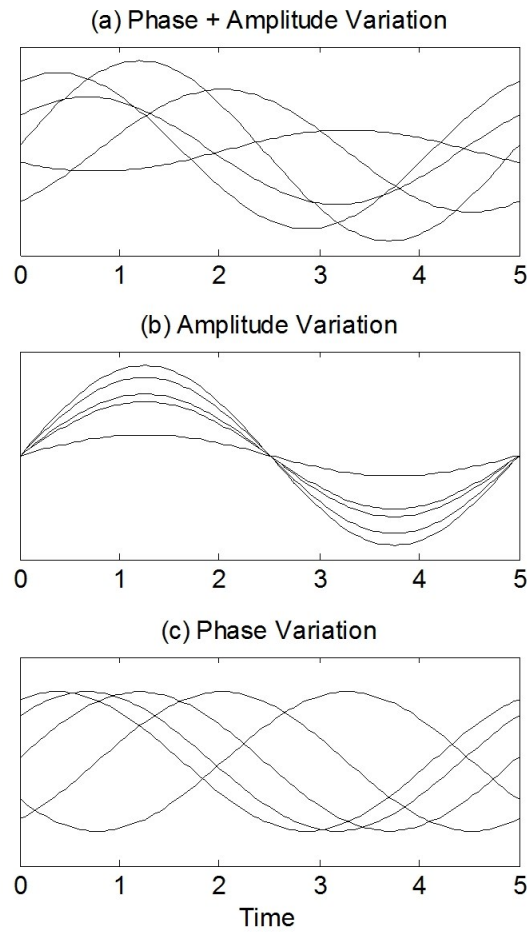


Figure 7-1: Five synthetic curves that vary in (a) both amplitude and phase, (b) amplitude only, and (c) phase only.

A more fundamental limitation of functional GSCA is that it equalizes the amount of variation in each latent variable. As mentioned in Sections 4.2

and 4.4.3, functional GSCA normalizes each latent variable in such a way that the squared norm of each latent variable equals to the number of subjects. Considering that each latent variable is centered to have zero mean, this way of normalization amounts to constraining the variance of each latent variable to unity. As a result, the difference in the amount of variation across different latent variables is ignored, which can be of the main interest of a study. For example, researchers may want to examine whether physical development in adolescents manifests greater variability than cognitive development. Functional GSCA is not equipped to address such research questions. In order to investigate the difference in the amount of variation across different latent variables, it is necessary to scale latent variables in a different way so that they do not lose such information. Devising such a scaling method for functional GSCA is not as trivial as it may seem to be.

Another research question that functional GSCA cannot address is how the relationships among variables evolve over time. Sometimes researchers are interested in examining change in associations among variables rather than the shape of change on a variable. For example, Li, Root, and Shiffman (2006) revealed that the effect of negative mood on urge to smoke changed over various stages of the smoking-cessation process. Functional GSCA is not appropriate at all for examining such time-varying relationships since it assumes that the relationships among latent variables, or path coefficients, are not time-varying. In order to investigate time-varying

path coefficients, one may need to devise a different model. Consider the situation in which each latent variable is measured by several indicator variables over multiple occasions and the relationship among the latent variables is assumed to vary over time. In such situations, one might consider extending the original GSCA in the following way. First, a latent variable is defined as a weighted composite of multiple observed indicator variables as in the original GSCA. Second, the loadings and weights are not allowed to vary across subjects and occasions to ensure that the same constructs are measured for all subjects and all occasions. This assumption is comparable to that in simultaneous component analysis (Kiers & ten Berge, 1994; Timmerman & Kiers, 2003) that decomposes multivariate longitudinal data from each subject into sets of time-varying component scores and a constant loading matrix, in which the loading matrix is assumed to be common to all subjects and occasions. Third, the path coefficients are allowed to vary over time. As in time-varying effect models developed in the context of linear regression (e.g., Hastie & Tibshirani, 1993; Ramsay & Silverman, 2005; Tan et al., 2012), time-varying path coefficients in GSCA can be modeled by representing path coefficient functions with basis function expansions. Although this approach seems feasible, a more rigorous research is necessary to make it work.

References

- Aston, J. A. D., Chiou, J.-M., & Evans, J., P. (2010). Linguistic pitch analysis using functional principal component mixed effect models. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 59(2), 297-317.
- Besse, P., & Ramsay, J. O. (1986). Principal components analysis of sampled functions. *Psychometrika*, 51(2), 285-311.
- Bollen, K. A., & Curran, P., J. (2006). *Latent curve models: a structural equation perspective*. Hoboken, NJ: Wiley-Interscience.
- Browne, M. W., & Du Toit, S. H. C. (1991). Models for learning data. In L. M. Collins & J. L. Horn (Eds.), *Best methods for the analysis of change: Recent advances, unanswered questions, future directions* (pp. 47-68). Washington, DC: American Psychological Association.
- Cardot, H. (2000). Nonparametric estimation of smoothed principal components analysis of sampled noisy functions. *Nonparametric Statistics*, 12, 503-538.
- Cardot, H., Ferraty, F., & Sarda, P. (2003). Spline estimators for the functional linear model. *Statistica Sinica*, 13, 571-591.
- Craven, P., & Wahba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, 31, 377-403.
- de Boor, C. (2001). *A practical guide to splines*. New York: Springer.

- de Leeuw, J., Young, F. W., & Takane, Y. (1976). Additive structure in qualitative data: An alternating least squares method with optimal scaling features. *Psychometrika*, 41(4), 471-503.
- Donaldson, G. W., Chapman, C. R., Nakamura, Y., Bradshaw, D. H., Jacobson, R. C., & Chapman, C. N. (2003). Pain and the defense response: Structural equation modeling reveals a coordinated psychophysiological response to increasing painful stimulation. *Pain*, 102, 97-108.
- Duncan, T. E., Duncan, S. C., & Strycker, L. A. (2006). an introduction to latent variable growth curve modeling: Concepts, issues, and applications. Mahwah, NJ: Erlbaum.
- Efron, B. (1982). *The Jackknife, the bootstrap and other resampling plans*. Philadelphia: SIAM.
- Fahn, S., Elton, R. L., & members of the UPDRS Development Committee. (1987). Unified Parkinson's disease rating scale. In S. Fahn, D. Marsden, D. Calne & M. Goldstein (Eds.), *Recent developments in Parkinson's disease*. Florham Park, NJ: MacMillan Healthcare Information.
- Fan, J., & Zhang, J. (2000). Two-step estimation of functional linear models with applications to longitudinal data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 62(2), 303-322.

- Fan, J., & Zhang, W. (1999). Statistical estimation in varying coefficient models. *The Annals of Statistics*, 27(5), 1491-1518.
- Girard, S. (2000). A nonlinear PCA based on manifold approximation. *Computational Statistics*, 15(2), 145-167.
- Gu, C. (2002). *Smoothing spline ANOVA models*. New York: Springer.
- Hall, P., & Hosseini-Nasab, M. (2006). On properties of functional principal components analysis. *Journal of the Royal Statistical Society. Series B (Methodological)*, 68(1), 109-126.
- Hastie, T. J., & Tibshirani, R. (1993). Varying-coefficients models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 55(4), 757-796.
- Hastie, T. J., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning : Data mining, inference, and prediction*. New York: Springer.
- He, G., Müller, H.-G., & Wang, J.-L. (2003). Functional canonical analysis for square integrable stochastic processes. *Journal of Multivariate Analysis*, 85, 54-77.
- Hershberger, S. L. (2003). The growth of structural equation modeling: 1994-2001. *Structural Equation Modeling*, 10(1), 35-46.
- Hoehn, M. M., & Yahr, M. D. (1967). Parkinsonism: Onset, progression and mortality. *Neurology*, 17, 427-442.

- Hwang, H., Desarbo, W. S., & Takane, Y. (2007). Fuzzy clusterwise generalized structured component analysis. *Psychometrika*, 72(2), 181-198.
- Hwang, H., Jung, K., Takane, Y., & Woodward, T. S. (2012). Functional multiple-set canonical correlation analysis. *Psychometrika*, 77(1), 48-64.
- Hwang, H., & Takane, Y. (2004). Generalized structured component analysis. *Psychometrika*, 69(1), 81-99.
- Hwang, H., Takane, Y., & Malhotra, N. (2007). Multilevel generalized structured component analysis. *Behaviormetrika*, 34(2), 95-109.
- Jackson, I., & Sirois, S. (2009). Infant cognition: Going full factorial with with pupil dilation. *Developmental Science*, 12(4), 670-679.
- James, G. M., Hastie, T. J., & Sugar, C. A. (2000). Principal component models for sparse functional data. *Biometrika*, 87(3), 587-602.
- Kaplan, D. (2009). *Structural Equation Modeling: Foundations and Extensions*. Thousand Oaks, CA: Sage.
- Kiers, H. A. L., & ten Berge, J. M. F. (1994). Hierarchical relations between methods for simultaneous component analysis and a technique for rotation to a simple simultaneous structure. *British Journal of Mathematical and Statistical Psychology*, 47(1), 109-126.

- Leurgans, S. E., Moyeed, R. A., & Silverman, B. W. (1993). Canonical correlation analysis when the data are curves. *Journal of the Royal Statistical Society. Series B (Methodological)*, 55(3), 725-740.
- Li, R., Root, T. L., & Shiffman, S. (2006). A local linear estimation procedure for functional multilevel modeling. In T. A. Walls & J. L. Schafer (Eds.), *Models for intensive longitudinal data* (pp. 63-83). New York: Oxford University Press.
- Lindquist, M. A. (in press). Functional Causal Mediation Analysis with an Application to Brain Connectivity. *Journal of the American Statistical Association*.
- Mattar, A. A. G., & Ostry, D. J. (2010). Generalization of dynamics learning across changes in movement amplitude. *Journal of Neurophysiology*, 104, 426-438.
- McArdle, J. J. (1986). Latent growth within behavior genetic models. *Behavior Genetics*, 16, 163-200.
- Meredith, W., & Tisak, J. (1990). Latent curve analysis. *Psychometrika*, 55(1), 107-122.
- Mulaik, S. A. (1971). *The foundations of factor analysis*. New York: McGraw-Hill.
- Ocaña, F. A., Aguilera, A. M., & Valderrama, M. J. (1999). Functional principal components analysis by choice of norm. *Journal of Multivariate Analysis*, 71, 262-276.

- Ormoneit, D., Black, M. J., Hastie, T., & Kjellström, H. (2005). Representing cyclic human motion using functional analysis. *Image and Vision Computing*, 23, 1264-1276.
- Ramsay, J. O., & Dalzell, C. J. (1991). Some tools for functional data analysis. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53(3), 539-572.
- Ramsay, J. O., Hooker, G., & Graves, S. (2009). *Functional data analysis with R and MATLAB*. New York: Springer.
- Ramsay, J. O., & Silverman, B. W. (2005). *Functional data analysis*. New York: Springer.
- Rao, C. R. (1958). Some statistical methods for the comparison of growth curves. *Biometrics*, 110, 49-58.
- Raudenbush, S. W. (2001). Comparing personal trajectories and drawing causal inferences from longitudinal data. *Annual Review of Psychology*, 52, 501-525.
- Raudenbush, S. W., & Bryk, A. S. (2002). *Hierarchical Linear Models: Applications and Data Analysis Methods* (Second ed.). Thousand Oaks, CA: Sage.
- Schumaker, L. L. (2007). *Spline functions: Basic theory*. Cambridge: Cambridge University Press.

- Silverman, B. W. (1995). Incorporating Parametric Effects into Functional Principal Components Analysis. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(4), 673-689.
- Taft, C. T., Kaloupek, D. G., Schumm, J. A., Marshall, A. D., Panuzio, J., King, D. W., & Keane, T. M. (2007). Posttraumatic Stress Disorder Syndrom, physiological reactivity, alcohol problems, and aggression among military veterans. *Journal of Abnormal Psychology*, 116(3), 498-507.
- Tan, X., Shiyko, M. P., Li, R., Li, Y., & Dierker, L. (2012). A time-varying effect model for intensive longitudinal data. *Psychological Methods*, 17, 61-77.
- Timmerman, M. E., & Kiers, H. A. L. (2003). Four simultaneous component models for the analysis of multivariate time series from more than one subject to model intraindividual and interindividual differences. *Psychometrika*, 68(1), 105-121.
- Tucker, L. R. (1951). *A method for synthesis of factor analysis studies*. Washington D.C.: Dept. of the Army.
- Vines, B. W., Krumhansl, C. L., Wanderley, M. M., & Levitin, D. J. (2006). Cross-modal interactions in the perception of musical performance. *Cognition*, 101, 80-113.
- Viviani, R., Grön, G., & Spitzer, M. (2005). Functional principal component analysis of fMRI data. *Human Brain Mapping*, 24, 109-129.

- Wahba, G. (1990). *Spline models for observational data*. Philadelphia: Society for Industrial and Applied Mathematics.
- Wiesner, M., & Windle, M. (2004). Assessing covariates of adolescent delinquency trajectories: A latent growth mixture modeling approach. *Journal of Youth and Adolescence*, 33(5), 431-422.
- Wu, Y., Fan, J., & Müller, H.-G. (2010). Varying-coefficient functional linear regression. *Bernoulli*, 16(3), 730-758.
- Yao, F., & Lee, T. C. M. (2006). Penalized spline models for functional principal component analysis. *Journal of the Royal Statistical Society. Series B (Methodological)*, 68(1), 3-25.
- Yao, F., Müller, H.-G., & Wang, J.-L. (2005). Functional data analysis for sparse longitudinal data. *Journal of the American Statistical Association*, 100(470), 577-590.
- Yogev, G., Giladi, N., Peretz, C., Springer, S., Simon, E. S., & Hausdorff, J. M. (2005). Dual tasking, gait rhythmicity, and Parkinson's disease: Which aspects of gait are attention demanding? *European Journal of Neuroscience*, 22, 1248–1256.