8-2017

Generalized Partial Least Squares Approach for Nominal Multinomial Logit Regression Models with a Functional Covariate

Amani Mohammed H Albaqshi

UNIVERSITY OF NORTHERN COLORADO

Greeley, Colorado

The Graduate School

GENERALIZED PARTIAL LEAST SQUARES APPROACH FOR NOMINAL MULTINOMIAL LOGIT REGRESSION MODELS WITH A FUNCTIONAL COVARIATE

A Dissertation Submitted in Partial Fulfillment
of the Requirements for the Degree of
Doctor of Philosophy

Amani Mohammed H Albaqshi

College of Education and Behavioral Sciences
Department of Applied Statistics and Research Methods

August 2017

This dissertation by: Amani Mohammed H Albaqshi

Entitled: *Generalized Partial Least Squares Approach for Nominal Multinomial Logit Regression Models with A Functional Covariate*

has been approved as meeting the requirement for the Degree of Doctor of Philosophy in College of Education and Behavioral Sciences in Department of Applied Statistics and Research Methods

Accepted by the Doctoral Committee

Khalil Shafie, Ph.D., Research Advisor

Jay Schaffer, Ph.D., Committee Member

Trent Lalonde, Ph.D., Committee Member

Heng-Yu Ku, Ph.D., Faculty Representative

Date of Dissertation Defense ______________________________

Accepted by the Graduate School

Linda L. Black, Ed.D.
Associate Provost and Dean
Graduate School and International Admissions

ABSTRACT

Albaqshi, Amani Mohammed H. *Generalized Partial Least Squares Approach for Nominal Multinomial Logit Regression Models with A Functional Covariate.* Published Doctor of Philosophy dissertation, University of Northern Colorado, 2017.

Functional Data Analysis (FDA) has attracted substantial attention for the last two decades. Within FDA, classifying curves into two or more categories is consistently of interest to scientists, but multi-class prediction within FDA is challenged in that most classification tools have been limited to binary response applications. The functional logistic regression (FLR) model was developed to forecast a binary response variable in the functional case. In this study, a functional nominal multinomial logit regression (F-NM-LR) model was developed that shifts the FLR model into a multiple logit model. However, the model generates inaccurate parameter function estimates due to multicollinearity in the design matrix. A generalized partial least squares (GPLS) approach with cubic B-spline basis expansions was developed to address the multicollinearity and high dimensionality problems that preclude accurate estimates and curve discrimination with the F-NM-LR model. The GPLS method extends partial least squares (PLS) and improves upon current methodology by introducing a component selection criterion that reconstructs the parameter function with fewer predictors. The GPLS regression estimates are derived via Iteratively ReWeighted Partial Least Squares

(IRWPLS), defining a set of uncorrelated latent variables to use as predictors for the F-GPLS-NM-LR model. This methodology was compared to the classic alternative estimation method of principal component regression (PCR) in a simulation study. The performance of the proposed methodology was tested via simulations and applications on a spectrometric dataset. The results indicate that the GPLS method performs well in multi-class prediction with respect to the F-NM-LR model. The main difference between the two approaches was that PCR usually requires more components than GPLS to achieve similar accuracy of parameter function estimates of the F-GPLS-NM-LR model. The results of this research imply that the GPLS method is preferable to the F-NM-LR model, and it is a useful contribution to FDA techniques. This method may be particularly appropriate for practical situations where accurate prediction of a response variable with fewer components is a priority.

ACKNOWLEDGEMENTS

First and foremost, Alhamdulillah (All praises to Allah), I would like to thank Almighty Allah for His blessings and guidance, for providing the inspiration to attempt this dissertation, and instilling in me the strength to see it through.

I would also like to express my sincere appreciation to my research advisor, Dr. Khalil Shafie, without whose guidance, support, and ideas this dissertation would never have become a reality. I am grateful for the stimulating conversations, patience, and guidance provided along the way. Additionally, I wish to express appreciation to my committee members. I am thankful to Dr. Jay Schaffer for his additional insight on the aspect of this work and for the feedback that has strengthened my research. I am also grateful to Dr. Trent Lalonde, who encouraged me to complete my PhD. His comments and discussions provided additional avenues to explore in my research, and he was always there to listen and give me valuable advice. It has been a great honor and pleasure to study with all of you. I am also thankful to Dr. Ku for providing helpful feedback and for kindly serving as a committee member. I wish to extend a special acknowledgment to Dr. Susan Hutchinson and Keyleigh Gurney for their encouragement and support throughout my time at the University of Northern Colorado.

To my friends at ASRM, many thanks for being there and always supporting me. I am thankful for the support I received from Dr. Jamil, and I would also like to express my gratitude to my best friends ever in Greeley, CO. I am lucky to have your friendship.

To my Husband Dr. Naif, my beloved daughters Aljazi and Lujain, and my twins Maria and Lamar, I am immensely grateful for your manifest love and support in pursuing my dreams, your outer energy, your encouragement, and your belief in me. I thank all of you for filling my life with joy and happiness and providing me with the inner strength to persevere in this endeavor. I love you all, Always & Forever.

I will forever be thankful to my dearest family, whose love and support helped me through the difficulties of this journey even though we were not always together. I would like to express my appreciation towards my aunties Zhara, Alia, Wajeha, and my Mom Fawzia for devoting their energy and love to promoting my happiness especially aunty Zhara. I can barely find the words to express all the wisdom, love, and support you've given me. I dedicate this dissertation to you, and for that I am eternally grateful.

To my F.C and all my cousins, I sincerely thank you for your love and support through my entire life and for being as real sisters and brothers. Lots of love and thanks to all my siblings for their support. Thanks to all my friends who wished me good luck and encouraged me to do it. I dedicate this dissertation to my community and my family honor, and I am proud that I was able to become the first Ph.D. in the Albaqshi' family.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

A Generalized Partial Least Squares (GPLS) method was applied to a Functional Nominal Multinomial Logit Regression (F-NM-LR) model as the latest development in the estimation techniques that are used to improve the estimation of functional parameter. The performance of the proposed methodology will be studied and compared to the classic alternative estimation method of Functional Principal Component (FPC) regression.

Due to advances in modern technology, Functional Data Analysis (FDA) has been a prominent subject in statistical literature for the last two decades. Various researchers have developed approaches to FDA across scientific disciplines having different objectives, e.g., science, engineering, biology, medicine, chemistry, geology, and sports (Aguilera, Aguilera-Morillo, and Preda, 2016; Ferraty and Vieu, 2006; Ramsay and Silverman, 2002). Regardless of variety in applications, the purpose of FDA is to provide information about curves, surfaces, or anything else that varies over a continuum. To avoid confusion, FDA could be described as a shape-to-numbers converter; in FDA, a single functional curve, i.e., a single observed function, is called a replication.

Often, FDA deals with high-dimensional space; this type of data is measured repeatedly over time or at several discrete points, and it comes to us through a process naturally described as a functional form. For example, this is the case of the evolution of a magnitude value for temperature as a function of time. The goal of FDA is to express repeated measurement data for each individual as a smooth function and then to draw information from the collection of these functions. Also, FDA emerges as a natural generalization of multivariate data analysis techniques to the case where the data are curves that are generated. Its purpose is to address the problems of high-dimensionality and correlations between observations at nearby time points that occur with functional data. A procedure for this type of analysis was developed by Ramsay and Silverman (2005).

The most recently developed applications in FDA are regression models. The main objective of these statistical techniques is to model and predict one or more response variables in terms of a set of related functional predictor variables. The primary applications of FDA that have been developed include functional principal component (FPC) regression (Aguilera and Escabias, 2000), functional partial least squares (FPLS) regression (Preda and Saporta, 2005), and functional logistic regression (FLR) James (2002). The interested reader can review Ramsey and Silverman (2002, 2005) for the statistical methodology introduced and interesting real data applications such as growth curves in medicine, financial series derived from stock market movements, and rainfall and temperature curves in the environmental field.

Although substantial advances have been made in FDA, there are still challenges to the accurate estimation of parameter functions in this type of analysis, ranging from noisy, discrete observations to representing infinite-dimensional objects numerically and measuring variation in infinite dimensions. One of the most important of these challenges to functional linear regression models is that the infinite dimension of the predictor space in real data causes a problem with estimating the parameter function.

There are two commonly used processes for constructing a functional dataset (an infinite object) from a finite observed sample: basis expansion and smoothing methods. In a functional regression model it is common to represent the functional data in terms of basis functions (B-splines, wavelets, Fourier, etc.) and assume both the sample curves and the parameter function belong to a finite space generated by this basis. This way, the functional linear regression model is converted into a multiple regression model in terms of sample curve basis coefficients.

Specific techniques for basis expansion are best chosen to represent the characteristics of the functional curve. For instance, Fourier basis are a good choice for periodic data; B-splines are an efficient, flexible, generic choice for non-periodic data. Some other common basis functions are wavelet and monomial. On the other hand, good basis systems approximate any sufficiently smooth data. Basically, smoothing methods eliminate small "wiggles" in the data while retaining the right shape. The smoothness of the process that generates functional data differentiates it from classic multivariate data and the smoothing process ensures the information in the derivatives of the functions is reasonably accurate (Ramsay and Silverman, 1997).

Despite the advantages of multiple linear models, some problems arise related to the estimations produced by this type of model. One issue is that the least squares estimation of the parameters function of model is usually affected by multicollinearity because of high correlations between the columns of the design matrix. Additionally, the number of basis functions of the sample curves can in many cases be higher than the sample size due to the large number of variables available from discrete observations. These problems are usually solved by regressing the response variable on an optimum set of orthogonal covariates such as principal component regression or partial least squares regression.

Currently, partial least squares (PLS) regression and principal component regression (PCR) are the most popular estimation methods used to reduce the high dimensionality and multicollinearity frequently encountered in functional regression analysis (Escabias, Aguilera, and Aguilera-Morillo, 2014). Several researchers have presented FPCR as a method to obtain an estimate of the functional parameters of a model (Escabias et al., 2014; Preda, Saporta, and Lévéder, 2007). The main criticism of FPCR is that it is computed without taking into account the response variable. Hence, FPLS regression is an attractive alternative to FPCR because it takes into account the correlation between the predictor and response when selecting regression components (A. Aguilera, Aguilera-Morillo, and Preda, 2016; Delaigle and Hall, 2012a; Preda and Saporta, 2005).

Partial least squares regression is a technique that reduces the predictors to a smaller set of uncorrelated components and performs least squares regression on these components, instead of on the original data. The PLS approach usually leads to stable and highly predictive models. The latent variables are mutually independent (orthogonal) linear combinations of original descriptors. Thus, the PLS model contains the smallest necessary number of factors. This method first became popular in Chemometrics (i.e., computational chemistry) and in sensory evaluation (Martens and Stark, 1991). However, PLS regression is also becoming a tool of choice in the social sciences as a multivariate technique for non-experimental and experimental data alike (McIntosh, Bookstein, Haxby, and Grady, 1996).

In terms of procedure, the PLS regression method iteratively produces a sequence of orthogonal functions as does functional principal components but it offers maximum predictive performance. Particularly, PLS is used to obtain sensible estimates of the functional parameters of the model and the selection of components used in these methodologies appears to be a major task in real data analysis. With this method, the comparison is performed in terms of the integrated mean squared error (IMSE) of the parameter function of the fitted model. The capacity of standard methods for establishing criteria to select PLS components (leave-one-out cross-validation) in terms producing accurate estimates of functional parameters is reviewed in Chapter III.

Three common categories are used for classifying functional regression models based on the role played by the functional curves in each model: scalar responses and functional predictors (scalar-on-function regression), functional responses and scalar predictors (function-on-scalar regression), and functional responses and functional predictors (function-on-function regression) (Ramsay and Silverman, 2005). Domains in which scalar-on-function regression has been applied include chemometrics, cardiology, brain science, climate science, and many others. This dissertation focused on nonlinear approaches to scalar-on- function regression.

Beyond functional linear regression models, two nonlinear functional regression models that are promising for certain applications include the functional logistic regression model and the functional multinomial logit regression model. The functional logistic regression model has received broader attention in the literature in recent years as it has been applied with different objectives for predicting a binary response from a functional predictor (Escabias, Aguilera, and Valderrama, 2004). However, this model approximates the sample curves and the parameter function on a finite dimension space generated by a basis, so it has the same multicollinearity problem as linear models. Therefore, the parameter function estimation is very inaccurate so its its interpretation in terms of odds ratios may be erroneous. Again, PLS regression and PCR are the most popular methods used in functional regression to solve this problem.

Classification of samples into two or more categories is almost always of interest to scientists. Thus, the natural generalization of the functional logit model in the case of a categorical response variable with a finite set of categories greater than two is the functional multinomial response model. A functional multinomial logit regression model based on FPC regression was introduced by Escabias et al. (2014). Different types of logit transformations can be applied in the analysis depending of the type of response being considered (nominal or ordinal).

Classic PLS univariate regression for continuous responses results from the iterated use of ordinary least squares (OLS). Marx (1996) proposed an extension of classical PLS in the context of generalized linear regression as a dimension reduction tool. Partial least squares generalized linear regression follows the rationale of PLS, but criterion optimization at each step is based on maximum likelihood. The acronym PLS, is retained to refer to the general methodology used to relate a response variable to a set of predictor variables. The approach proposed for PLS generalized linear regression is easy to implement as it generalizes readily to any linear model at the level of the predictor variables (Ding and Gentleman, 2005). This dissertation examined generalized PLS regression for a categorical response variable associated to a functional curve with respect to a logit transformation of a nominal response variable.

Purpose of the Study

The purpose of this research was to (a) extend the ordinary partial least squares method within the context of the functional generalized linear model (James, 2002) based on iteratively reweighted partial least squares to obtain an accurate estimation of the functional parameters and (b) to classify sample curves in the categories of the response variable of the functional nominal multinomial logit regression (F-NM-LR) model. This adaptation is designated as the Generalized Partial Least Squares method (GPLS). This method has two key advantages: (a) it is expressed only in terms of functional predictors that are explicitly computable, and (b) it demonstrates consistency.

In this study, the GPLS regression method is used as an alternative to functional principal component (FPC) regression as presented by Escabias et al. (2014). This research applies the proposed GPLS method directly to the F-NM-LR model, achieving an improvement in functional parameter estimates for supervised multicategory classification problems that deals with dependence, produces lower classification error rates, and have high predictive ability.

Hence, for the present purposes, the author focuses on the commonly used nominal logit response model with functional predictors. The objective of the proposed methods and results presented in this work was to help practitioners use real-world functional data to make valid interpretations and decisions. A simulation and a real dataset were used in this study to investigate the usefulness of the proposed method.

Statement of the Problem

When traditional functional multinomial logit regression model are used to predict categorical logit response variables associated with observed curves, two critical problems arise. First, the classical FPC method calculates PCs without taking into account the response variable. Second, the original PLS algorithm was designed for a continuous outcome with constant variance that has a linear relationship with the predictor. The GPLS method is a strategy that integrates these concepts into functional generalized linear models.

Significance of the Study

This study generalizes the functional PLS logistic regression model for a binary response variable to the case of a functional GPLS nominal multinomial logit regression model. This process is intended to provide more accurate estimation of functional parameters and improved sample curve classification. To investigate the effectiveness of the proposed method, the results generated by the proposed model were compared to those of classical FPC regression method under a baseline logit regression model as presented by Escabias et al. (2014).

Research Questions

The following questions were be addressed in this study:

Q1 How is a generalized partial least squares regression method developed for parameter function estimation in the functional nominal multinomial logit regression model?

Q2 How does the functional generalized partial least squares nominal multinomial logit regression model behave in terms of goodness-of-fit measures, such as the correct classification rate and the integrated mean squared error, with changes of the functional predictor dependent on arbitrary values assigned to a and b?

Q3 How does the functional generalized partial least squares nominal multinomial logit regression model behave in terms of goodness-of-fit measures, such as the correct classification rate and the integrated mean squared error, based on changes in the number of the nominal response categories?

Q4 How does the precision of the generalized partial least squares regression method compare to the principal component regression method proposed by Escabias et al. (2014) under the functional nominal multinomial logit regression model?

Q5 How to develop an R code to fit a functional generalized partial least squares nominal multinomial logit regression model to real data?

The effectiveness of the proposed methodology was evaluated using a simulation study conducted with changes in the number of categories of a nominal response variable. In addition, a real-world dataset (spectrometric data) was used to demonstrate the performance of the GPLS estimation techniques in terms of curve classification, model precision, and the accuracy of parameter estimates.

Saeys, De Ketelaere, and Darius (2008) suggested the potential use of functional data analysis for spectroscopy and chemometric data. The spectrometric data consist of spectrometry curves (absorbance measured in terms of wavelength) generated for different substances, such as food. For this study, the the functional generalized partial least squares nominal multinomial logit regression model was used to classify near infrared (NIR) spectra of corn samples according to the spectrometer that generates them so was is a retrospective study of spectral curves. The data are publicly available and can be downloaded from http://www.eigenvector.com/data/Corn/index.html.

Rationale for the Study

Limited studies have focused on applying PLS procedures for functional logistic models to classifying sample curves into the categories of a response variable. The power of using the PLS method was it uses a set of uncorrelated latent variables (as the PCs) and it takes into account the relationship between the response and the predictor variables in the regression model. The PLS method has been used to estimate the discriminant coefficient functions for linear discriminant analysis (LDA) when predictors are functional curves; however, it is built based on the similarities between LDA and multiple linear regression (Preda et al., 2007). A. Aguilera et al. (2016) proposed two penalized versions of functional PLS regression with roughness penalties on the weight functions to improve the classification ability of functional logistic models such that they produce suitable estimates.

The PLS approach has also been used for classification purposes by Escabias et al. (2007) for modeling a binary response with a functional predictor because of the equivalence of linear discriminant analysis and linear regression. Although the results for classification look good, their approach may not be ideal because the original PLS algorithm was designed for a continuous outcome with constant variance that had a linear relationship with the predictor.

In contrast to partial least squares, the FPC regression method is the classic tool and base solution for addressing the multicollinearity and high dimensionality problems that occur with functional data (Escabias et al., 2014). However, the FPC method has been subject to many criticisms based on the fact that PCs are calculated without taking into account the response variable. As mentioned earlier, Cardot, Faivre, and Goulard (2003) considered a functional logistic model for predicting land use with the temporal evolution of coarse resolution remote sensing data. These authors proposed a quadrature method to approximate the linear predictor of the model from discrete data and a functional PCA to reduce the dimensions of the problem.

However, Escabias et al. (2007) indicated it was more informative to consider a PLS regression method as an alternative to an FPC regression method. They have demonstrated that a functional PLS logistic regression model provides better estimations of the parameter function than did the FPC method with a greater reduction in the number of components needed and similar predictions. More broadly, Mahesh, Jayas, Paliwal, and White (2015) have shown that PLS regression models demonstrated better prediction performance than the PCR models for predicting protein contents and hardness of wheat.

In conclusion, to the author's knowledge, no work has been done to apply the proposed GPLS method directly to the F-NM-LR model. In addition, the GPLS method would be flexible enough to be applied to a broad family of statistical problems. Therefore, this dissertation offers substantial progress toward improving estimates generated from functional data.

Delimitations of the Study

The current study faces a few limitations. First, the author investigated the procedure with only a nominal response variable; thus, there was no guarantee the results of this dissertation would be valid for an ordinal response variable. Second, the simulation results were limited to one-dimensional curves for one functional predictor. Additionally, the proposed GPLS method was limited when only predictor variables were functional; thus, there was no guarantee the results of this dissertation would be valid when both response and predictor variables are functional. Finally, this study applied regularization techniques, (e.g., reduction methods) to improve the precision of regression models; which might not reflect the diversity that occurs in shrinkage methods.

Definition of Terms

Definition I.0.1. A sample $X = (x_1, \ldots, x_n)$ is called functional data when the *ith* observation is a real function $\{X_i(t) : t \in T\,, i = 1, \ldots, n\}$, and hence, each $X_i(t)$ is a point in some function space $\mathcal{H}$.

Definition I.0.2. An inner product on the real function space $\mathcal{H}$ is a function $\langle \cdot, \cdot \rangle$ defined on $\mathcal{H} \times \mathcal{H}$ with values in $\mathbb{R}$ and satisfying the properties

1 $\langle ax + y, z\rangle = a\langle x, z\rangle + \langle y, z\rangle$,

2 $\langle x, y\rangle = \langle y, x\rangle$,

3 $\langle x, x\rangle \geq 0$ and $\langle x, x\rangle = 0$, iff $x = 0$

for all $x, y, z \in \mathcal{H}; a, b \in \mathbb{R}$.

Definition I.0.3. Basis representation represented by supposing $L^2(T)$ the space of all squared integrable function defined on T. The inner product defined on L^2 is

$$\langle f, g\rangle = \int_T f(t)g(t)dt, \quad \forall f, g \in L^2(T).$$

A system of basis function $\phi_p(t)$ is called orthonormal if

$$\|\phi_p\|^2 = \int_T \mid \phi_p(t)|^2 dt = 1$$

, and $\boldsymbol{\psi}_{vu} =< \phi_v, \phi_u >$ being the $(p \times p)$ matrix of inner products of the basis function. Where $\boldsymbol{\psi}_{vu}$=1 for $u = v$, and 0 otherwise.

Abbreviations

For the sake of readability, the following is a list of the main abbreviations used in this dissertation:

FDA	Functional Data Analysis
FPC	Functional Principal Components
PCR	Principal Components Regression
FPCR	Functional Principal Components Regression
PCR	Principal Components Regression
PLS	Partial Least Square
FPLS	Functional Partial Least Square
FPLSR	Functional Partial Least Square Regression
GPLS	Generalized Partial Least Square
FLR	Functional Logistic Regression
F-PLS-LR	Functional-PLS-Logistic Regression
FMLR	Functional Multinomial Logit Regression
F-NM-LR	Functional-Nominal Multinomial-Logit Regression
F-GPLS-NM-LR	Functional-GPLS-Nominal Multinomial-Logit Regression
CCR	Correct Classification Rate

ROC	Receiver Operating Characteristic
IMSE	Integrated Mean Squared Error
CVMSE	Cross-Validation for Mean Squared Error

Dissertation Structure

The remainder of this dissertation is organized as follows. Chapter II provides relevant background about functional data analysis and a literature review showing the development of estimation techniques for FDA. In addition, the necessary knowledge for basis expansion and how it is applied in functional data settings is given. Chapter III describes the proposed methodology for a GPLS regression base solution for a functional multinomial logit regression (FMLR) model and the development of this model. Chapter IV contains a thorough analysis results of both a simulation study and a real dataset application to illustrate the proposed GPLS method. Chapter V presents a brief discussion of the results, conclusions, and potential motives for additional work with F-GPLS-NM-LR model as well as future research directions. Finally, supplementary material including figures, tables, and details on R code are available in the appendices.

CHAPTER II

REVIEW OF LITERATURE

This chapter contains some necessary definitions and basic tools for functional data analysis; basis expansion, the B-spline smoothing method, and registration are presented. In addition, the most well-known estimation methods in functional data analysis, FPC and FPLS regression, and their algorithms are shown in order to give the appropriate foundation for the model and the estimation procedure developed in this dissertation. The chapter is organized as follows: A brief overview of Functional Data Analysis (FDA) and the fundamentals of the basis function approach with common types of basis and smoothing methods are followed by the primary idea of curve registration. The functional linear regression model with a scalar response and a review of the current literature on functional logistic regression used to model a binary response variable are then presented. Additionally, a review of the functional PCA and PLS based solutions for the multicollinearity problem that arises with functional logistic regressions is provided. Next, a functional multinomial logit model for nominal responses and the motivation for the use of this framework throughout this work are demonstrated. Finally, the prospective methodology for parameter estimation of the functional multinomial logit model using FPCA, FPLS, and G-PLS are discussed.

A Brief Overview of Functional Data Analysis

Functional data analysis (FDA) is a relatively new field of research in statistics that became popular at the end of the nineties. Historically, this term is rooted in work by Ramsay (1982) and Ramsay and Dalzell (1991), and two additional monographs addressing the subject were written by Ramsay and Silverman (1997, 2002). These works provide an accessible overview of the foundations of FDA and its applications, and they were followed by Ramsay and Silverman (2005), which provides advanced references for work in this field. This resource inspired substantial interest in developing statistical models for FDA. Ramsay and Silverman (2005) present many ideas and techniques for collecting functional data samples, mainly curves and other functional observations measured over a continuous parameter such as time.

To establish the general nature of functional data, Figure 1 shows examples of functional data sets from three different disciplines. Furthermore, to simplify the idea of FDA, a general way of thinking is to view each replication as a single observation, where the basic unit of information is the entire observed function rather than a string of numbers. For example, stock and option prices in finance are often treated as functions of time because the data observed are not the univariate or multivariate observations of classical statistics; they are functions that are attributable to an underlying infinite dimensional process. Several functional data sets have been studied in Ferraty and Vieu (2006), and they present methods that are considered useful for analyzing discretely observed data and generalizations of classical multivariate techniques to the FDA field. Aguilera et al. (2010) developed the formulation and estimation of functional PLS regression from basis expansion of sample curves, and a number of other books have subsequently appeared. In short, the history of this area is much older and dates back to Grenander (1950) and Rao (1958).

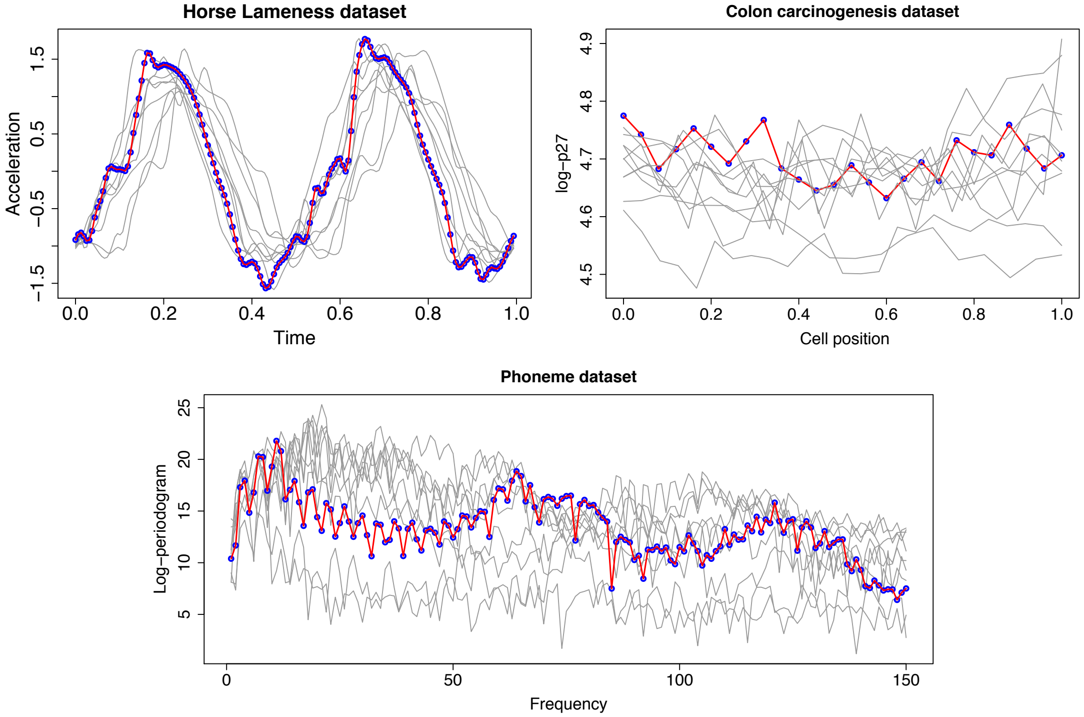


Figure 1. Example of functional data from different disciplines.

Currently, data that are being recorded continuously during a time interval or intermittently at several discrete time points are called functional data. FDA has become one of the most motivating and popular statistical topics due to its influence on crucial societal issues, which is a failure of standard multivariate statistics (Ferraty, 2011). This subfield is still in rapid development and becoming more common in scientific studies, for example, data collected by weather stations, human growth curve data, marketing applications, and handwriting data. From this point of view, FDA can be challenging when the data objects are curves that need to be measured repeatedly over time. A current interest is in how well FDA methods extend to the case of functional regression and serve as alternatives to the classical multivariate methods of classification such as regression methods for a binary response (logistic regression model) or multi-category response (multi-logit regression models) as observed in Hervás, Silva, Gutiérrez, and Serrano (2008).

These approaches allow modeling variables from a set of predictors and interpreting the relationship between the categorical response and the functional predictor via the parameters of the model. There are numerous examples of methodologies developed that extend functional logistic regression techniques and functional multinomial regression models to use for functional PCA and PLS to reduce the dimensions (Escabias, Aguilera, and Valderrama, 2004; A. M. Aguilera, Escabias, and Valderrama, 2008; Aguilera-Morillo, Aguilera, Escabias, and Valderrama, 2013; Ratcliffe, Heller, and Leader, 2002).

A commonly arising quandary for the researcher, however, is deciding which datasets should and should not be treated as functional data. Generally speaking, one could say that there are some prerequisites for considering data to be appropriate for functional data analysis. For instance, the data must realistically arise from an underlying smooth process, and there must be enough data to extract the essential feature of the underlying process; additionally, there must be repetitions in order to study the variations of interest. Ultimately, in functional data there is no need for equally spaced or perfect measurement. Ramsay and

Silverman (2002), Ferraty and Vieu (2006), and Horváth and Kokoszka (2012) provide excellent summaries of methods and case studies for handling functional data from different perspectives, and statistical inference related to some FDA methods was recently studied by Horváth and Kokoszka (2012).

Definition of Functional Data Analysis

Functional Data Analysis (FDA) is " a branch of statistics that analyses data providing information about curves, surfaces ,or anything else varying over a continuum" (Ramsay and Silverman, 2005, p. 9). The continuum is usually time, but it can be other indices such as wavelength, probability, spatial position, frequency, weight, and so on. The aforementioned definition has been discussed in many different studies with varied purposes (Escabias, Aguilera, and Valderrama, 2004; Kayano, Dozono, and Konishi, 2010; Müller and Stadtmüller, 2005).

Essentially, FDA is a generalization of multivariate data analysis techniques to the case where the data are curves. This type of analysis draws information from collecting multiple functions as a smooth curve that occurs over some domain (e.g., time, spatial location, wavelength, or probability). Escabias et al. (2012) present a review of the FDA methods usually used in biometrics and biostatistics, and they discuss some interesting applications. Fundamentally, the statistical tool based on the analysis of functional data can be viewed as the realization of a one-dimensional stochastic process, often assumed to be in a Hilbert space, such as $L^2(T)$ (Gasser et al., 1984; Gasser and Kneip, 1995; Rice and Silverman, 1991). Regardless of the stochastic nature of functional data, the usual assumption in functional data analysis, and in this work, is that the curves belong to the squared integrable functions space $L^2(T)$.

Summary Statistics of Functional Data

Functional means and variances. The classical summary statistics for univariate data familiar in the introductory statistics classes apply equally to functional data. The mean function with values

$$\bar{x}(t) = N^{-1} \sum_{i=1}^{N} x_i(t) \tag{2.1}$$

is the average of the continuous point-wise across replications. Similarly the variance function `var` has values

$$\mathtt{var}_X(t) = (N-1)^{-1} \sum_{i=1}^{N} [x_i(t) - \bar{x}(t)]^2, \tag{2.2}$$

and the standard deviation function is the square root of the variance function.

Covariance and correlation functions. The covariance function summarizes the dependence of records across different argument values, and it is computed for all t_1 and t_2 by

$$\mathtt{cov}_X(t_1, t_2) = (N-1)^{-1} \sum_{i=1}^{N} \{x_i(t_1) - \bar{x}(t_1)\}\{x_i(t_2) - \bar{x}(t_2)\} \tag{2.3}$$

The associated correlation function.

$$\mathtt{corr}_X(t_1, t_2) = \frac{\mathtt{cov}_X(t_1, t_2)}{\sqrt{\mathtt{var}_X(t_1)\mathtt{var}_X(t_2)}}. \tag{2.4}$$

Correlation function is mainly intended for situations where the two functions are reflecting the same variable, such as angle in the gait data. However, one also can want to correlate two variables of an entirely different character, such as temperature and precipitation in the weather data (Ramsay et al., 1997).

Steps of Functional Data Processing

In reality, the random function of FDA cannot be tracked continuously; instead, a sample of the random function can take a number of measurements at discrete times or separate spatial locations for one object. Generally, these sample curves are independent of each other. Usually, a single observed function is a called a replication; and in turn, functional data is a random sample of replications. Typically, the sampling designs do not have strict requirements. Figure 2 shows the change in temperature over the course of a full year, taken from 35 weather stations across Canada. Each data point, marked with an "x", represents the mean temperature recorded by a weather station for the entire month, collected over 30 years. The colors correspond to the geographic climates of the stations.

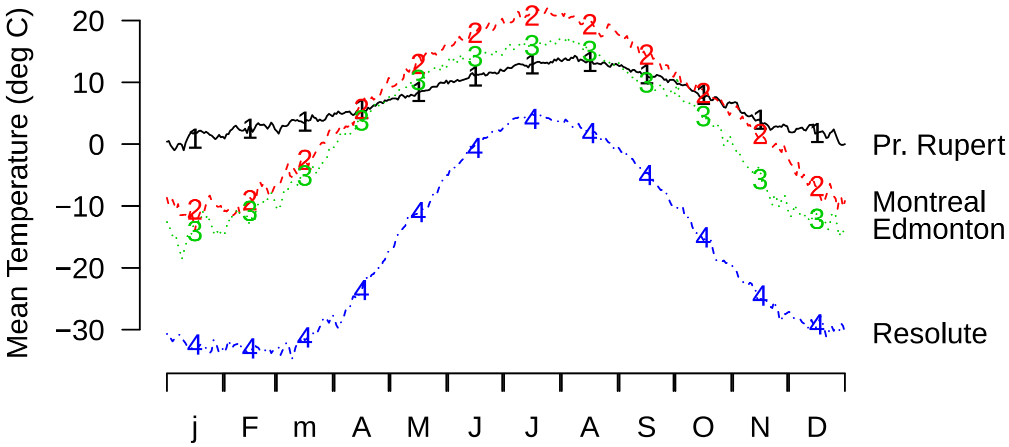


Figure 2. Raw data for mean monthly temperatures in 35 Canadian weather stations. The colors and the numbers correspond to the geographic climates of the stations.

However, analyzing the variation within and between functions requires the extensive use of derivatives. In addition, the typical underlying function x is smooth, and consequently the concepts of roughness and smoothness of functions play a crucial role, warranting review.

In summary, the process for FDA is as follows:

1. Collect, clean, and organize the raw data.

2. Convert the data to functional form.

   - Select Basis Set.

   - Select Smoothing Operator.

3. Explore the data through plots and summary statistics.

4. Register the data, so that important features occur at the same aligned argument values.

5. Carry out exploratory analysis, such as functional principal components analysis, or functional partial least squares regression.

6. Construct models, if appropriate.

7. Evaluate model performance.

In terms of programming for FDA, well-developed computational code for Matlab, Splus, and R are available from the FDA website (Ramsay, 2011). The book by Ramsay, Hooker, and Graves (2009) illustrates strong connections between functional data analysis techniques and their applied applications using R and Matlab. Moreover, the FDA package in R provides an extensive range of smoothing and modeling tools, along with a number of classical functional data sets (Ramsay, Hooker, and Graves, 2009).

Advantages of Functional Data Analysis

The advantages of FDA reflect its objectives as described by Ramsay and Silverman (2005). These objectives are more or less the same as for other branches of statistics, and they include the following: (a) representing and transforming the data in appropriate ways for further analysis; (b) investigating the main sources of variation and patterns among data with high dimensionality; (c) explaining variations in the response variable by using the information about the covariance of variables; (d) displaying the data to highlight various characteristics; and (e) comparing two or more sets of data with respect to certain types of variation, where the two sets of data can contain different sets of replicates of the same functions or different functions for a common set of replicates.

In reality, the observed data often are collected with random noise called error, which are often assumed to be independent across and within subjects. Thus, the strength of FDA is that it can adjust for error easily because for each subject, one observes repeated measurements. However, functional data that are assumed to be observed continuously without errors are the easiest type to handle (J.-L. Wang, Chiou, and Mueller, 2015). Due to these practical advantages, many applied statisticians work with functional data analysis, for example, in the analysis of growth curves (Rao, 1958), demographic forecasting (Hyndman and Booth, 2008; Hyndman and Shang, 2009; Hyndman and Ullah, 2007), electronic commerce research, marketing science (S. Wang, Jank, and Shmueli, 2008), and many more.

Basis Expansion of Functional Data

Basis functions and smoothing methods are the two most common approaches to constructing a functional dataset. Basis functions are a known set of mathematically independent functions that describe a curve or any other data distributed over a continuum. Basis functions provide all appropriate computations necessary for storing information, fitting the data quickly with minimal programming, and providing flexibility, paired with the computational power required to fit hundreds of thousands of data points. Additionally, the basis approach is not too technical; it uses more advanced methods, such as the calculation of variations and functional analysis. As such, there are no practical limitations as to what functional data require.

The main feature of a functional variable is the infinite dimension of the space to which the observations belong. However, due to the infinite dimensionality of the functional variable, direct estimation of a functional regression model (in which some of the response and/or predictor variables are functional), particularly a linear one, is generally impossible. A customary approach to the estimation of functional regression models is to employ a roughness penalty, which performs well in the case of noisy or unequally spaced observations of the curves. Another common solution is the generalization of the linear regression approach to represent the functional data in terms of basis functions (B-splines, wavelets, trigonometrics), and provide approximations for the basis coefficients via the use of a large number of discrete observations, which may be characterized by irregularity and sparseness. Thus, the functional model is converted into a multiple model in terms of sample curve basis coefficients.

However, because sample curves are usually observed in a finite set of sampling points that could be unequally spaced and different among the sample units, typically, the first step in FDA is to reconstruct the true functional form (an infinite object) of each sample curve from a finite set of discrete observations by assuming an expansion of each sample curve in terms of a basis of functions and fitting the basis coefficients using smoothing or interpolation. The most common method assumes that the sample curves belong to a finite dimensional space generated by a basis of functions. This way, the estimation of a functional regression model is reduced to an equivalent multivariate regression model with high correlation between the predictor variables.

Basis Coefficients for Functional Data

In practice, interpolation (for data observed without error) or least squares approximation (for noisy data) can be used to compute the basis coefficients for FDA. For example, A. M. Aguilera, Gutiérrez, and Valderrama (1996) first considered natural cubic spline interpolation to calculate estimates for functional PCA. Also, Escabias, Aguilera, and Valderrama (2007, 2014, 2015) introduced quasi-natural cubic spline interpolation to estimate the risk of drought from time evolution of temperatures. However, A. M. Aguilera et al. (2008) used least squares approximation with both B-splines and trigonometric functions to interpret the relationship between time evolution of stress and flares in Systemic Lupus

- If the sample curves are observed without error

  $$X_{ij} = X_i(t_{ij}), \qquad j = 1, \ldots, m_i,$$

  an interpolation procedure can be used.

- On the other hand, if the sample curves are observed with error

  $$X_{ij} = X_i(t_{ij}) + \epsilon_{ij}, \qquad j = 1, \ldots, m_i,$$

  least squares smoothing is used after choosing a suitable basis.

In practice, most functional data are contaminated with random noise (measurement error). These errors are sometimes insignificant, (e.g., recording the height of children over time), but in other cases, noise is a critical issue, (e.g., accounting for the influence of head movements when taking functional magnetic resonance images (fMRIs).

Basis Representation

In practice, various classes of basis can be used depending on the characteristics of the curves and the observations. The assumptions of the underlying stochastic process $\{X(t), t \in T\}$ are necessary in order to develop the theory of the basis function approach, the first and second order moments of $X(t)$ are assumed to exist and to be finite within a Hilbert space of measurable functions. A curve can be represented by a basis when you assume that the data belong to this space. A basis is a set of known functions $\{\Phi_p(t)\}_{p \in N}$ such that any function could be arbitrarily approximated by a linear combination of a sufficiently large number of these functions (Ramsay and Silverman, 2005). Fitting the sample curve $x_i(t)$ can permit the following basis expansions:

$$x_i(t) = \boldsymbol{a}_i'\Phi(t), \qquad \beta(t) = \boldsymbol{b}'\Phi(t),$$

where $\boldsymbol{a}_i = (a_{i1}, \ldots, a_{ip})'$ is the vector of sample curve basis coefficients and $\boldsymbol{b} = (b_1, \ldots, b_p)'$ is the vector of the parameter function basis coefficients.

Note that choosing the number of basis functions is important and critical for all subsequent computations. Small numbers of basis functions mean minimum flexibility (under-fitting), and larger numbers of basis functions result in overfitting (Febrero-Bande and Oviedo de la Fuente, 2012; Ramsay and Silverman, 2002). However, the choice of the basis should be based on the objective of the analysis and on the data. Figure 3 shows a noisy functional observation approximated by B-splines in order to graphically show the phenomenon of over or under-smoothing. The grey curve represents the actual function, the circles are the observation points, and the blue and red lines represent two B-spline approximations, where $K = 5$ and $K = 40$, respectively. The smaller K under-fits the underlying model, while the larger k replicates the noise, and not the original function (over-fitting).

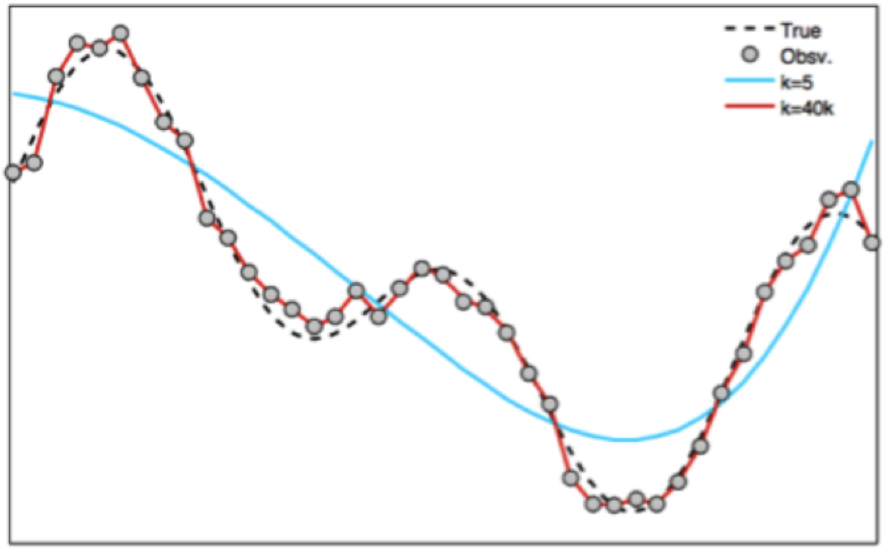


Figure 3. Approximation of a functional data via B-splines. Grey: true data. Blue: under-fitted ($K = 5$). Red: over-fitted approximation ($K = 40$).

Common Types of Basis Systems

This section provides a short summary of common types of basis used with functional data. Choosing the ideal basis and its dimension for approximating the functional form of a set of sample curves is very important, and it must be done according to the characteristics of the data. The most common examples of useful basis systems according to the main features of the sample curves are the following: (a) Fourier basis for the case of periodic data; (b) B-spline basis for non-periodic smooth data with continuous derivatives up to a certain order, which provides better local behavior (De Boor, 2001); and (c) wavelet basis for data with strong local behavior where derivatives are not required (Ramsay and Silverman, 2005). Additionally, there are several different types of basis functions that can be created using the function basis package in R and Matlab (constant, monomial, polynomial, B-splines, power, exponential, and Fourier). The most common basis functions according to the main features of the data are reviewed below:

Polynomial basis. The oldest system consists of the powers of t, that is, $1, t, t^2, \ldots, t^k$. This basis was important in the days of hand calculation. Actually, polynomials are easy to compute but also inflexible, and they seem to be appropriate only for simple data structures without many local features.

Fourier basis. In the early nineteenth century the Fourier basis technique was been developed. This basis is known as a good choice for modeling periodic functions. This basis is easy to use to fit periodic or near periodic data with a fixed and known frequency, such as weather data, some types of economic data, and so on. In this basis, which containing 1 and a series of pairs of *sines* and *cosines* that is, $1, sin(\omega t), cos(\omega t), sin(2\omega t), cos(2\omega t)$, $sin(3\omega t)$, $cos(3\omega t), \ldots$ the constant ω plays an important role (Ramsay, 2006). However, Fourier series are not great at capturing sharp changes.

Wavelets basis. These types of basis are terrific at capturing sharp changes. Commonly, the wavelet basis is used for nonlinear techniques for approximating functions, particularly those whose domains are defined on a bounded interval (Morris and Carroll, 2006). However, wavelets basis shares the computational advantages of potentially orthogonal basis systems similar to the Fourier series and splines basis.

Spline basis. These are related to polynomial systems, and they require some further explanation. Currently, a spline basis function is mostly used for non-periodic data. This is a popular application for many reasons. Spline basis systems provide fast coefficient computation when thousands of equations are required.

Spline basis functions are excellent systems that indicate special structures for the equations, which is a distinct advantage of a spline basis. This fast computation of individual basis functions increases the ability to create appropriate and smooth approximations of the underlying data. This basis combines the efficiency of polynomials (which are included in it) with a greater flexibility for fitting highly curvy data or sharp curvatures at specific locations. The technique basically relies on dividing the time interval and making a polynomial approximation in each subinterval while taking care of the breakpoints (De Boor, De Boor, Mathématicien, De Boor, and De Boor, 1978). Expansions using the spline basis system are used in this dissertation. Examples of B-spline and Fourier basis are illustrated in Figure 4.

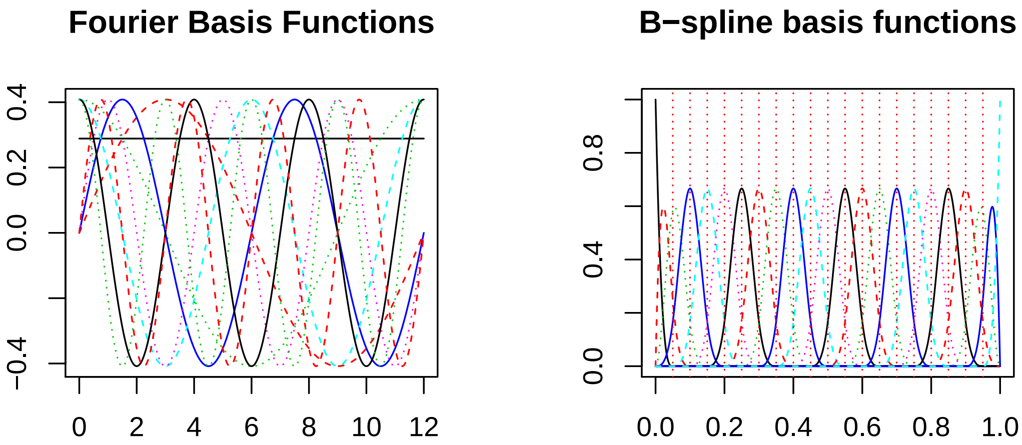


Figure 4. Basis systems: The Fourier basis function (left panel), and the B-spline basis function (right panel).

The Characteristics of B-Spline Functions

Before considering how to construct a spline basis system, an understanding of the essential characteristics of splines is necessary. In computer science, the term spline refers to a piecewise polynomial curve. In mathematics, a spline is a piecewise polynomial function of degree m formed by joining polynomials together at fixed points called knots ξ_l (Wood and Jennings, 1979). Knots are given by dividing the interval extending from lower limit t_L to upper limit t_U , to approximate a curve into $L + 1$ sub-intervals separated by L interior boundaries ξ_l (knots, or sometimes breakpoints) for the spline.

For example, consider the simplest case in which a single breakpoint divides interval $[t_L : t_U]$ into two subintervals. Then, the spline function within each interval is a polynomial of specified degree (the highest power defining the polynomial) or order (the number of coefficients defining the polynomial, which is one more than its degree), a spline function being of order m or degree $m - 1$ over each sub-interval. The two polynomials are required to join smoothly at the interior breakpoint ξ_1. This means, in the most common case, that the derivatives match up to the order one less than the degree. Knots are often spaced equally, but two important rules should be observed in placing knots: Place more knots where you know there is strong curvature and fewer where the function changes slowly. However, one must be sure that there is at least one data point in any interval. Therefore, the order m (order = degree + 1) of the polynomial segments and the location of the knots define the spline basis system. Because splines are constructed from polynomials, computing their derivative at any point between two knots is simple; at a knot, it is required that the derivatives up to order (m - 2) also join. That is, the derivative of order (m - 2) of a spline function is usually continuous.

Once the knots are given, B-splines can be evaluated recursively for any degree of the polynomial by using a numerically stable algorithm (De Boor, 2001). For example, when $q = 3$ the basis functions are called cubic B-splines ($q =$ the order of the polynomials plus the number of interior breakpoints). They are used to fit regular sample curves with first and second continuous derivatives. Typically, cubic splines are used to design objects because they are reasonably flexible, and they can be computed and stored efficiently (Hall, Poskitt, and Presnell, 2001).

On the other hand, the choice of knots is an important problem when working with B-splines. If too many knots are selected, over-fitting the data will result. On the other hand, too few knots results in under-fitting. Some automatic numerical schemes for optimizing the number and the position of the knots have been proposed to solve this problem (J. H. Friedman and Silverman, 1989). However, in each region or station there must be at least one observed value t_i within each subinterval, because practically speaking, it is a waste to have breakpoints without data. Moreover, if one suspects that a sharp feature exists in a particular region and there are only one or two data values in its area, there will be little hope of adequately describing the data, and the fitted curve may as well be smooth. Typically, the hope is for gains in statistical power and computational efficiency with a small number of basis functions, reflecting the old philosophers saying, " the simpler, the better."

There are two strategies for determining exactly where to position breakpoints. The first, and most commonly used strategy, is to make them equally spaced, but the requirement of having at least one observation in every subinterval needs to be considered. The second strategy has the advantage of ensuring that there is a practical amount of data associated with each subinterval. This is called the quantile placement, which places a breakpoint at every fixed number of observed values of t.

Quasi Natural Cubic Spline System

A quasi-natural cubic spline approximation method consists of two main steps: First, the degree and knot vector are determined, and then the B-spline coefficients of the approximation are computed from given data according to a formula. For some methods, like spline interpolation and least squares approximation, this formula corresponds to the solution of a linear system of equations. Typical global methods are cubic spline interpolation, which are popular local method. Quasi-interpolants allow us to establish important properties of B-splines. The true functional forms of the curves have been reconstructed via quasi-natural cubic spline interpolation, as used by Escabias et al. (2014). The advantage of quasi-natural cubic spline interpolation is its flexibility and simplicity. As mentioned above, B-spline functions have good local behavior, and this is the reason for their frequent use in practice.

However, many authors have stated that in the case where the interpolation design matrix is obtained from the observation of correlated data, the model will have collinearity between predictor variables Hosmer and Lemeshow (1989). Consequently, using basis expansion usually results in the functional model turning to multiple models that have no unique solution. Escabias et al. (2005), proposed to use quasi-natural cubic spline interpolation. Quasi-natural cubic spline interpolation consists of cubic spline interpolation, but it uses uniformly generated values (around and next to zero) as boundary conditions. That is, the interpolation matrix is uniformly generated in the interval $[0, 1]$.

Beyond the basis approximations, there exist a great variety of smoothing methods (with or without a basis representation) to remove noise or just make the discretized data continuous.

Smoothing Methods for Functional Data

The smoothing process in generating functional data is an important step that distinguishes this type of data from classical multivariate observations (Ramsay and Silverman, 1997). Ideally, smoothness is the required assumption for functional data analysis, making it flexible enough for modeling. The basic idea of smoothing methods for functional data is to retain the right shape of the underlying random function by making the variations caused by measurements error and other factors more even.

Basic smoothing techniques are nonparametric, and researchers can select from many tools (Ramsay, 2006). Usually, the choice of a smoothing technique is crucial, and it is subject to both mathematical considerations and computational limitations. However, in principle, there is no universal rule that would provide an optimal choice.

Furthermore, for scientific models, derivatives play an important role; they are needed to construct models for data based on differential equations. The order of the derivative depends on the problem at hand. In other words, assuming that the underlying process is smooth, one can deduce that the adjacent observations should be linked together. Therefore, smoothness ensures that the information in the derivatives of functions can be used in a reasonable way. Figure 5 presents an example of smoothing data with different numbers of basis functions. Additionally, the first and second derivatives can reflect the energy exchange within a system, as described by Ramsay and Silverman (2002). Consequently, if the smoothness property did not apply to a functional linear model, there would be nothing much to be gained by treating the data as functional rather than just multivariate.

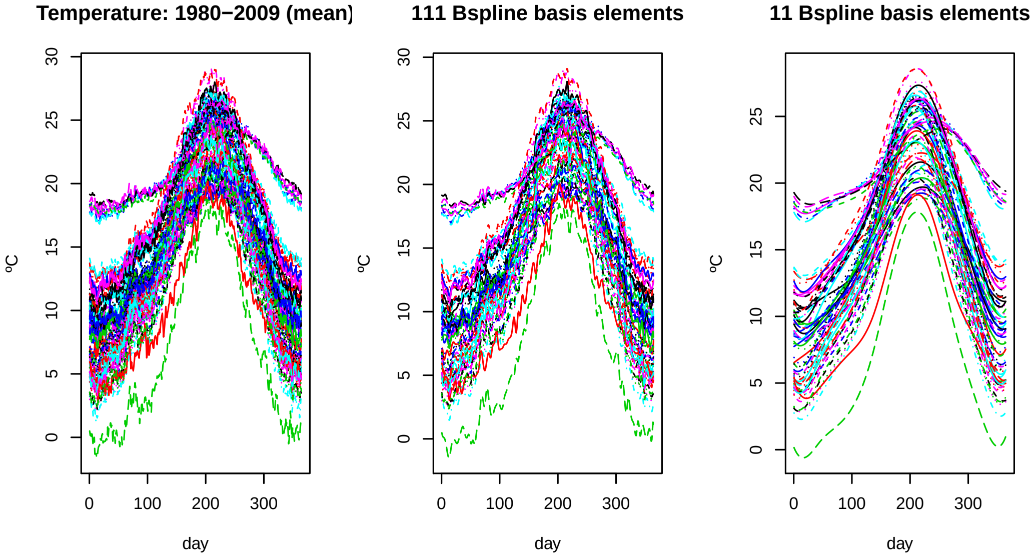


Figure 5. Example of smoothing functional data by changing the number of basis functions.

It is far from clear whether smoothing is a good practice when classifying functional data. Carroll, Delaigle, and Hall (2013) showed that under-smoothing is appropriate in practical cases such as the functional logistic model because the smoothing parameters that achieve good and even optimal performance in prediction and hypotheses testing fail in this context.

Crambes, Kneip, and Sarda (2009) carried out a comparative study of regression splines and smoothing splines. Basically, there are three different approaches for approximating smooth sample curves: regression splines (non-penalized least squares approximation), smoothing splines (continuous roughness penalty based on the integrated squared d-order derivative of each sample curve), and P-splines (discrete roughness penalty based on d-order differences between coefficients of adjacent B-splines). The purpose of all smoothing spline methods for sample curves is to improve the statistical estimates. Moreover, different basis systems (B-splines, wavelets, trigonometric ....) can be used depending on the smoothness and performance of sample functions. Cross-validation and generalized cross-validation are adapted to select a common smoothing parameter for all sample curves with the roughness penalty approaches.

Choosing the Smoothing Parameter

Garthwaite (1994) distinguish between two philosophical approaches to the question of choosing the smoothing parameter. The first approach is to choose the smoothing parameter subjectively. Varying features of the data that arise on different scales can be explored, and the one parameter value which “looks best” might be chosen. The second approach is to use an automatic method of choice such as cross-validation. Graven and Wahba (1979) provide a fundamental reference regarding the use of cross-validation to guide the choice of a smoothing parameter. In the functional case, the same two approaches to selecting a smoothing technique apply:

- Subjective choice
- Automatic method - driven choice:

    1. Cross-validation 2. Generalized cross-validation

Curve Registration

Curve registration or curve alignment is a procedure for transforming the time argument such that the curves are more aligned, as illustrated in Figure 6. In a functional dataset, there are two sources of variation: phase variation and amplitude variation. From this point of view, amplitude variation is referred to as variation in the magnitude or size of functional data, which measures the differences in the y-axis; variation in the time scale is often referred to as phase variation, which measures the differences in the x-axis. Thus, curve registration is an additional assumption if both amplitude variation (magnitude) and phase variation (time) are present in a functional dataset. The procedure for removing phase variation has been investigated under different names in different disciplines, namely, curve registration, curve alignment, and time warping in statistics, biology, and engineering.

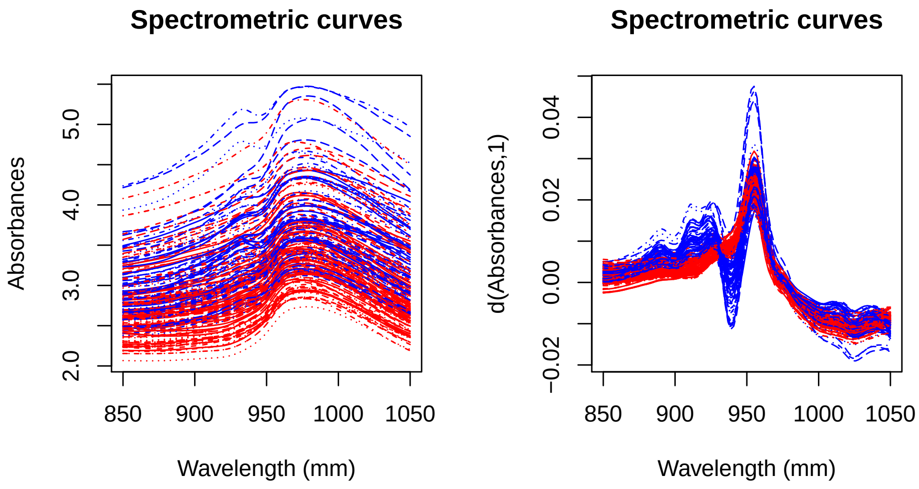


Figure 6. Example of comparing functional data with a) no curve registration of the data, and b) aligned data.

However, separating amplitude variation from phase variation is still a challenging problem for FDA. Thus, FDA techniques are designed to handle either phase or amplitude variation depending upon the particular problem at hand. For example, there may be cases in which amplitude variation is of primary interest and phase variation of secondary or little interest, such as in the case of a number of spectral datasets. In other circumstances, the opposite may be the case. Moreover, there may be times when both phase and amplitude variation are equally important. Curve registration and shift registration address these variations.

Curve registration transforms the functional curves by transforming their time or location arguments rather than transforming the measurements themselves. Shift registration removes amplitude effects that can be accounted for by vertical shifts by using a linear transformation including centering and rescaling the curves (Nason and Silverman, 1995). Then, the standard least square defined as the global sum of squared vertical differences between the shifted curves and the sample mean curve over all sample curves' can be minimized iteratively to obtain the estimates of shift parameters simultaneously.

Unlike the simple linear transformation of the argument in shift registration, landmark registration aligns the curves by lining up their most representative landmark. In order to identify the argument value for each landmark selected, landmark registration transforms the argument nonlinearly using a warping function. The landmark registration method removes phase variation by transforming the domain for each curve so that points specifying the locations of shape features are aligned across curves.

So far, the discussion has focused on exploratory data analysis: basis function systems, smoothing for FDA, and curve registration. The next section includes a discussion of the generalization of linear models that examine predictive relationships: functional linear regression models, functional logistic regression models, functional PC regression, functional PLS regression, and the functional multinomial logit regression model.

Functional Linear Regression Model with Scalar Response

The linear regression model was the first to be considered within the framework of functional data analysis (FDA). The functional linear regression model was intended for dealing with continuous scalar response variables when one of the predictor variables has a functional nature. Functional regression models with scalar response have been studied extensively (Cardot, Ferraty, and Sarda, 1999; James, 2002; Müller and Stadtmüller, 2005; Ramsay and Dalzell, 1991). The first theoretical contributions to the study of functional linear regression models for scalar response and functional predictors were made by Cardot et al. (1999). Meanwhile, Müller and Stadtmüller (2005), and Aguilera et al. (2015) considered the case where both predictor and response are functional. Therefore, the functional linear regression model is split into three types: (1) functional predictor regression (scalar on function), (2) functional response regression (function on scalar) and (3) function on function regression (function on function). This dissertation addresses functional regression models where the response variable is scalar and there is at least one functional covariate.

Functional Linear Model

This model assumes that the relationship between the scaler random response Y and the functional predictor $X(t)$ has a linear structure.

Let Y be a scalar random variable (scalar response) and $X(t)$ be a functional predictor $\{X(t) : t \in T\}$ whose sample curve belongs to the space $L^2(T)$ of the square integrable function (Ramsay and Silverman, 2005). With the purpose of predicting Y from $X(t)$, the functional linear model is then expressed as

$$y_i = \beta_0 + \langle x_i, \beta \rangle_{L^2[0,T]} = \beta_0 + \int_0^T x_i(t)\beta(t)dt + \epsilon_i \ , \tag{2.5}$$

where $y_i = (y_1, \ldots, y_n)'$ be a random sample of Y, β_0 is a constant, $\beta(t)$ is a parameter function, and $\epsilon_i = 1, \ldots, n$ are centered independent random errors of the model (2.5).

In practice, it is impossible to directly estimate the parameters of the functional linear model because of the infinite dimension of the predictor space. In addition, in practice there are only discrete observations of each sample curve at a finite set of knots, which could be unequally spaced and different for sample individuals.

One of the most common solutions to solve this problem is to assume that both the sample curves and the parameter function belong to a finite space generated by a basis of functions. An appropriate basis must be selected according to the main characteristics of the observed sample curves.

Basis of functions. Lets consider that both the sample curve and the parameter function belong to the same finite space generated by the basis. Then, let $\Phi(t) = (\phi_1(t), \ldots, \phi_k(t))'$ is a basis functions that span the space where $x_i(t)$ and $\beta(t)$ belong.

Thus, the $x_i(t)$ and $\beta(t)$ can be expressed in terms of the basic expansion as:

$$x_i(t) = \boldsymbol{a}_i'\Phi(t), \qquad \beta(t) = \boldsymbol{b}'\Phi(t),$$

where $\boldsymbol{a}_i = (a_{i1}, \ldots, a_{nk})'$ is the vector of basis coefficients of the i^{th} sample curve, and $\boldsymbol{b} = (b_1, \ldots, b_k)'$ is the vector of of basis coefficients of the parameter function.

This way, the functional linear model (2.5) is equivalent to the multiple linear model, and it is expressed in matrix form as

$$Y = \mathbf{1}\beta_0 + \boldsymbol{A}\boldsymbol{\Psi}\boldsymbol{b}, \tag{2.6}$$

with $\boldsymbol{A}$ is the design matrix which has as rows the parameter function basis coefficients of the i^{th} sample curve, and $\boldsymbol{\Psi} = (\psi_{jk}) =< \phi_j(t), \phi_k(t) >$ is the $p \times p$ matrix that has as entries the usual inner products between basis functions.

It is known that the use of the least squares criterion to estimate this model yields an ill posed problem because of the Wiener-Hopf equation, which does not have a unique solution (Saporta, 1981). There are additional problems related to the estimation of this multiple linear model. First, the number of basis functions needed to get accurate estimations of the sample curves could be higher than the sample size so that a dimension reduction procedure is necessary. Second, least squares estimation of the parameters of model (2.5) of the parameters of the model is usually affected by multicollinearity because of the high correlation between the columns of its design matrix.

In order to reduce the infinite dimension of the functional predictor and to solve the multicollinearity problem associated with the estimation of the functional linear model, most approaches regress the response variable on a set of orthogonal covariates, such as a reduced number of functional principal components (FPC) (A. M. Aguilera et al., 2010; Valderrama, Ocaña, Aguilera, and Ocaña-Peinado, 2010; Preda et al., 2007; Reiss and Ogden, 2007; Escabias et al., 2007) or functional partial least squares (FPLS) components (Escabias et al., 2012; Cardot and Sarda, 2005; Preda et al., 2007; Reiss and Ogden, 2007). These components can then be used as predictor variables to provide an accurate estimation of the functional parameter.

Both methods produce linear combinations of the original predictor variables to construct new predictor variables (or components), but they construct the components in different ways (Escabias et al., 2007). In the next section the generalization of principal components regression and partial least squares regression to the case of a functional predictor is presented.

Functional Principal Components and Partial Least Squares Based Solutions

All of the representations of functional linear regression in terms of basis functions consider the high multicollinearity problem. Unfortunately, the effects of multicollinearity make the estimation of the parameters of the models inaccurate with high estimated variance. In the presence of multicollinearity (when the smallest eigenvalue of the predictors is close to zero), the sample covariance matrix can be nearly singular; so statistical inferences drawn from the singular covariance matrix linear could be incorrect. For example, the estimates of ordinary least squares (OLS) for regression coefficients are likely to be too large in absolute values and possibly of the wrong sign (Wichern and Churchill, 1978). Consequently, the overall MSE tends to be large when multicollinearity is present.

As in the multivariate case, the problems of multicollinearity and infinite dimension impede estimations of the FLR model. So similarly, the estimation may be inaccurate due to a strong correlation between the components of the design matrix, as well as the possibility that basic functions used to approximate sample curves can outnumber observations. Because principal components and partial least squares components methods use predictors as uncorrelated sets of variables, these methods can generally solve this problem in the functional case.

Functional Principal Component Based Solution

Before reviewing Functional Principal Component (FPC), it is more meaningful to review multivariate PCA, which is used as a dimension reduction tool for multivariate data, because principal component regression (PCR) is a technique that is based on principal component analysis.

Classical PCA is the most useful tool for reducing dimension while preserving the maximum amount of information from the original variables (Pearson, 1901). Typically, the variables selected as regressors are the principal components with higher variances. Selected regressors are based on eigenvectors that correspond to higher eigenvalues in the sample variance-covariance matrix of predictor variables (James, 2002). Therefore, the goal of PCA is to find the sequence of orthogonal components that most efficiently explains the variance of the observations. PCA is covered in almost all textbooks on multivariate data analysis, in particular, Jolliffe (2002), and Witten, Tibshirani, and Hastie (2009)

The PCR method includes four steps: (a) Conduct an initial PCA on the observed data matrix for the predictor variables to identify principal components; (b) (in most cases) choose a subset of those components, based on appropriate criteria, for further use; (c) use ordinary least squares regression to calculate a vector of regression coefficient estimates to regress the observed vector of outcomes on the selected principal components as covariates; and (d) use the selected PCA loadings to generate a final PCR estimator (equal in dimension to the total number of covariates) to transform this vector back to the scale of the actual covariates for estimating the regression coefficients that characterize the original model (Jolliffe, 2002).

Functional principle components (FPC). The PCA method was the first multivariate data analysis method to be extended to functional data, so the process is called functional principal component analysis (FPCA). This process has become the main tool used in FDA (Dauxois, Pousse, and Romain, 1982; Jolliffe, 2002). It is one of the most popular techniques used in functional linear models, and it has been considered by many authors (Cardot, Faivre, and Goulard, 2003; Ferraty, Van Keilegom, and Vieu, 2012; Hall and Vial, 2006). From this point of view, Escabias et al. (2004) have proposed functional regression solutions based on FPCA methods to avoid multicollinearity by taking covariates as a reduced set of PCs of the design matrix of the multiple logit models equivalent to the original FLR model. The technique contributed by Dauxois, Pousse, and Romain (1982) for

FPCA is now the basis of many developments in functional data analysis. For example, a two-step functional regression approach has been applied to forecast curves of pollen concentration from temperature curves (Valderrama et al., 2010). Ramsay and Silverman (2005) provide an introductory exposition on FPCA and a thorough review of methods for deriving principal component functions.

FPCA has been central to the development of the functional linear model, and this method generates a parsimonious model with orthogonal regressors and uncorrelated regression coefficients. FPCA is carried out in a similar fashion to PCA, except that it is necessary to renormalize the eigenvectors and interpolate them with a suitable smoother (Ramsay and Silverman, 2005). FPCA finds the set of orthogonal principal component functions to explore major sources of variation in a sample of curves. The differences in notation between PCA and FPCA are summarized in Table 1.

Table 1
The differences in notation between PCA and FPCA

	PCA	FPCA
Variables	$\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_p]$	$X(t) = [x_1(t), \ldots, x_n(t)]$
Data	Matrix $\mathbf{X} \in \mathbb{R}^{p \times p}$	Curves $\in L^2[T]$
Dimension	$p < \infty$	∞
Inner Product	$< \mathbf{X}, \mathbf{Y} >= \sum_{\mathbf{k=1}}^{\mathbf{p}} \mathbf{X_k Y_k}$	$< X, Y >= \int_T X(t)Y(t)dx$
Eigen Structure	Vector $\xi_k \in \mathbb{R}^p, \mathbf{V}\xi_k = \lambda_k \xi_k$, for $1 \leq k < \min(n, p)$	Function $\xi_k(x) \in L^2[T]$, $\int_{x_1}^{x_p} T\xi_k(x)dx = \lambda_k \xi_k(x)$, for $1 \leq k < n$
Components	Random variables in $\mathbb{R}^p$	Random function in $L^2[T]$

In FPCA, selecting the optimal number of functional principle components achieves more stable parameter estimate. Yao, Müller, and Wang (2005) proposed a way to use a functional version of the Akaike's information criterion to select the optimal number of components, and Hall and Vial (2006) proposed a bootstrap method to determine the optimal number of components. Also, Escabias et al. (2014) introduced scree plots, or the portion of total variance explained by each principal component.

In FPCA, the principle components are obtained as uncorrelated generalized linear combinations with maximum variance.

In general, the j^{th} principal component is given by

$$\xi_{ij} = \int_T x_i(t) f_j(t) dt, \quad i = 1, \ldots, n,$$

where the weight function or loading f_j is obtained by maximizing the variance

$$Max_f Var\left[\int_T x_i(t) f_j(t) dt\right]$$

$$w.r.t\left\{\|f\|^2 = 1 \; and \; \int f_l(t) f_j(t) dt = 0, \quad \forall l = 1, \ldots, j-1\right\}.$$

The weight functions are obtained as the eigenfunctions of the covariance operator C defined by

$$Cf(s) = \int c(t,s) f(t) dt, \quad s \in T.$$

also, in terms of the sample covariance function

$$c(t,s) = \frac{1}{n-1} \sum_{i=1}^{n} x_i(t) x_i(s).$$

That is, $Cf_i = \lambda_i f_i$. The principal components ξ_i are uncorrelated and their variances are given by the eigenvalues $Var[\xi_i] = \lambda_i$.

When the sample curves admit a basis expansion, functional PCA is equivalent to multivariate PCA of matrix $\mathbf{A\Psi^{1/2}}$ so that the weight functions w_j are computed by diagonalizing the covariance matrix of $\mathbf{A\Psi^{1/2}}$ (Ocaña, Aguilera, and Escabias, 2007).

Then, the sample curves are expressed in terms of FPC as:

$$x_i(t) = \sum_{j=1}^{n-1} \xi_{ij} f_j(t).$$

By truncating this representation in terms of the q principal components, can obtain an approximation of the sample curves whose explained variance is given by $\sum_{i=1}^{q} \lambda_i$, where q first principal components.

Different methods exist to select the optimum number of FPC to use and the order in which they must be included in the model that take into account both explained variability and the ability to predict the response while providing the best estimation of the functional parameters. However, Escabias et al. (2014) proved that in the case of FPC, the more efficient way for including PCs in the model consists of including principal components using a forward stepwise method based on a conditional likelihood ratio test, which takes into account the PCs relationships with the response variable rather than the natural order of explained variability to obtain an accurate estimated parameter function.

Remark. Functional PCA of the sample curves with respect to the usual inner product in $L^2(T)$ is equivalent to finite multivariate PCA of the matrix $\boldsymbol{A\Psi}^{1/2}$ with respect to the usual metric in $\mathbb{R}^p$ (Aguilera et. al., 2010). However, the FPCR method has been criticized because the PCs are calculated without taking into account the relation between the response and the predictor variables; and thus, their choice for regression has drawbacks. As alternative solution to this problem, PLS regression has recently been generalized to the case of functional data (Escabias et al., 2007).

Functional Partial Least Squares Based Solution

An alternative to FPC regression is Functional Partial Least Squares (FPLS) regression. The FPLS regression model was introduced by Preda and Saporta (2005) to solve the problems of high dimensionality and multicollinearity associated with the scalar-on-function linear model. PLS regression is a recent technique that generalizes and combines the features of principal component analysis and multiple regression.

The PLS regression components are obtained by replacing the least squares criterion with that of maximizing the covariance between linear spans of X(t) and response variables Y, respectively, as a solution to the Tucker's criterion (Tucker, 1938). PLS can handle both univariate and multivariate responses, and the process is computationally fast. The success of PLS as a standard tool has led to its application in scientific fields, for example, analyzing chemical data (Wold, 1975). Before reviewing FPLS, it is helpful to revisit the ordinary PLS method that is used for dimension reduction for multivariate data.

Partial least squares (PLS). Basically, the PLS method operates by forming linear combinations of the predictors using the response and then regressing the response on these latent variables. The PLS method can predict response variables, perform regression, and reconstruct the original dataset matrix simultaneously. PLS in its original form is for a continuous response variable, and the process is usually presented as an algorithm.

PLS algorithm. Let $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2 \ldots, \mathbf{x}_p)$ be the $n \times p$ matrix of predictors and $\mathbf{y}$ be the $(n \times 1)$ response vector. $\mathbf{X}$ can often be written as a bilinear form (Geladi and Kowalski, 1986):

$$\begin{aligned} \mathbf{X} &= \mathbf{TP}' + \mathbf{E}_k \\ &= \mathbf{t}_1\mathbf{p}_1' + \mathbf{t}_2\mathbf{p}_2' + \cdots + \mathbf{t}_K\mathbf{p}_K' + \mathbf{E}_K \end{aligned}$$

where $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_K]$ is the $n \times k$ matrix of *latent variables* or *scores*, and $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_K]$ is the $p \times k$ matrix of *loadings*. Also, $\mathbf{E}_K$ is the $n \times p$ residual matrix. Moreover, it is usually assumed that the $\mathbf{X}$ matrix is standardized so that each column has mean 0 and standard deviation 1 (although the latter is not necessary). Moreover, further assume that

$$\begin{aligned} \mathbf{y} &= \mathbf{X}\boldsymbol{\beta} \\ &= \mathbf{TQ} + \mathbf{F}_K \\ &= \mathbf{t}_1 q_1 + \mathbf{t}_2 q_2 + \cdots + \mathbf{t}_K q_K + \mathbf{f}_K, \end{aligned}$$

where $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_k]$ is matrix of *latent variables* and $\mathbf{F}_K$ is the residual matrix. Thus, $\mathbf{X}$ and $\mathbf{y}$ are linked via the latent variables $\mathbf{T}$.

Usually, the criterion for constructing components in PLS is to sequentially maximize the covariance between the response vector $\mathbf{y}$ and the predictor matrix. The PLS components are orthogonal. If is chosen to be the rank of $\mathbf{X}$ (i.e. minimum of the row rank and column rank of $\mathbf{X}$), and $\mathbf{X}$ is of full rank, then the PLS estimates of $\boldsymbol{\beta}$ are identical to ordinary least squares (OLS) estimates.

Functional PLS. The FPLS regression is a good alternative method to FPC regression, which visualizes the relation between object and functional predictor variables by maximizing the variance of X(t) without taking into account the response variable. There are some interesting studies that compare FPLS regression and FPC regression (A. M. Aguilera et al., 2010; Delaigle and Hall, 2012b; Reiss and Ogden, 2007). The main conclusion drawn from this research was that FPLS regression is superior to FPC regression because FPLS regression requires fewer components to capture the same interactions. FPLS also provides a much more accurate estimation of the parameter function than FPC regression because FPLS does take the response variable into account, often leading to a more parsimonious model. However, the prediction ability of both approaches has been found to be similar.

The main objective of FPLS method is to build a set of orthogonal components from a data sample and use them as predictor variables in a least squares fit. The orthogonal components consist of linear regressions of the functional predictor that are calculated by maximizing their squared covariance with the response (Tucker's criterion). A stepwise process determines the FPLS components using an iterative procedure, which is based on the residuals of each regression of response and predictor variables on the component that was calculated in the previous step. Delaigle and Hall (2012b) completed a detailed study about the theoretical properties and explicit formulation of functional PLS.

Functional PLS algorithm. The FPLS algorithm was introduced by A. Aguilera et al. (2016), and A. M. Aguilera et al. (2010) to solve the problems of high dimensionality and multicollinearity associated with the function linear model. The PLS components associated with the functional regression of a real random response Y in terms of a functional predictor, $X = X(t),\ t \in [0, T]$ are obtained as solutions of Tucker's criterion extended to functional data as

$$\max_{w \in L_2([0,T]), \|w\|_{L_2([0,T])}=1} Cov^2\bigg(\int_0^T X(t)w(t)dt, Y\bigg). \tag{2.7}$$

Let $\mathcal{C}_{YX}$ be the cross-covariance operator of X and Y and $\mathcal{C}_{XY}$ be its adjoint defined by

$$\begin{aligned} &\mathcal{C}_{YX} : L_2(T) \to \mathbb{R} \\ &\qquad f \mapsto x = \int_T Cov(X(t), Y)f(t)dt \\ &\mathcal{C}_{XY} : \mathbb{R} \to L_2(T) \\ &\qquad x \mapsto f(t) = \ x \cdot Cov(X(t), Y), \quad \forall \in [0, T]. \end{aligned}$$

The optimization problem to equation (2.7) can be rewritten as

$$\max_{w \in L_2(T)} \frac{\langle \mathcal{U}w, w\rangle}{\langle w, w\rangle}, \quad \text{where } \mathcal{U} = \mathcal{C}_{XY} \circ \mathcal{C}_{YX}$$

Therefore, the solution to equation (2.7) is the eigenfunction of the operator $\mathcal{U}$ associated to its largest eigenvalue λ_1,

$$\mathcal{U}w_1 = \lambda_{1(max)}w_1$$

and the first PLS component is defined as

$$t_1 = \int_T X(t)w_1(t)dt.$$

The PLS algorithm is an alternative procedure.

Let $X_0 = X$ and $Y_0 = Y$. For any positive integer h, let X_h and Y_h be the residuals of the linear regressions of X_{h-1} and Y_{h-1} respectively, with as predictor the h^{th} PLS component t_h, i.e.

$$X_h(t) = X_{h-1}(t) - p_h(t)t_h, \quad t \in T.$$

$$Y_h = Y_{h-1} - c_h t_h,$$

where $p_h(t) = (E(X_{h-1}(t)t_h)/E(t_h^2))$ and $c_h = (E(Y_{h-1}t_h)/E(t_h^2))$.

Then, at step h, the h^{th} PLS component is defined as the random variable maximizing the Tucker criterion (2.7) using the residuals X_{h-1} and Y_{h-1}

$$t_h = \int_T X_{h-1}(t)w_h(t)dt, \tag{2.8}$$

where $w_h(t)$ is the solution of

$$w_h = \arg \max_{w, \|w\|^2=1} Cov^2\left(\int_T X_{h-1}(t)w(t)dt, Y_{h-1} \right)$$

given by the largest eigenvalue of $\mathcal{U}_{h-1} = \mathcal{U} = \mathcal{C}_{XY}^{h-1} \circ \mathcal{C}_{YX}^{h-1}$. That is,

$$\mathcal{U}_{h-1}(w_h) = \lambda_h w_h,$$

with $\mathcal{C}_{XY}^{h-1}$ and $\mathcal{C}_{YX}^{h-1}$ being the cross-covariance operators of $X_{h-1}(t)$ and Y_{h-1}, respectively. The PLS linear approximation of Y at h^{th} iteration is then given by

$$\hat{Y}^h = c_1 t_1 + c_2 t_2 \ldots c_h t_h.$$

Notice that the expression of the PLS components defined by equation (2.8) can be rewritten as elements of the linear space spanned by $\{X(t) : t \in T\}$, i.e.

$$t_i = \int_T v_i(t)X(t)dt$$

with $v_i \in span\{w_1, \ldots, w_i\}$, $i = 1, \ldots, h$. Thus, the PLS linear approximation at step h becomes

$$\hat{Y}^h = c_1 \int_T v_1(t)X(t)dt + \cdots + c_h \int_T v_h(t)X(t)dt = \int_T \hat{\beta}^{PLS,h}(t)X(t)dt,$$

where $\hat{\beta}^{PLS,h}$ is the estimation of the parameter function $\hat{\beta}(t)$ in equation (2.5) provided by the functional PLS approach with h components. As in the case of functional PCR functional PLS is equivalent to multivariate PLS of Y on matrix $\boldsymbol{A\Psi}^{1/2}$ with respect to the usual metric in $\mathbb{R}^p$ (A. M. Aguilera et al., 2010), when the sample curves are expressed terms of basis functions. It has been reduced to regression of Y on a set of PLS component of $\boldsymbol{A\Psi}^{1/2}$.

Functional linear model component estimation in terms of original variables. In both FPCR and FPLS processes, the dimension reduction problem is reduced to the regression of Y on a set of PCR or PLS components of $\boldsymbol{A\Psi}^{1/2}$, so both approaches use the computational algorithm with the following steps:

- Computation and selection by cross-validation of a set of m components

$$\boldsymbol{Z}_{n\times m} = (\boldsymbol{A\Psi}^{1/2})_{n\times K}\boldsymbol{V}_{K\times m},$$

  where $\boldsymbol{Z}$ is the matrix whose columns are the first m FPCR or FPLS components, and $\boldsymbol{V}$ is the matrix comprising the columns of the first h eigenvectors associated with thet^{th} FPCR or FPLS components of each considered method.

- The estimated functional linear model of Y in terms of the first m FPCR or FPLS components is given by

$$\hat{Y}^m = \boldsymbol{1}\hat{\beta_0}^m + \boldsymbol{Z}\hat{\boldsymbol{\gamma}}^m$$

where $\hat{\boldsymbol{\gamma}}$ is the vector of the regression coefficients of Y on $\boldsymbol{Z}$.

- The vector of basis coefficients of the estimated parameter function

$$\hat{\beta}^m(t) = \Sigma_{k=1}^{K} \hat{\boldsymbol{b}}_k^m \phi_k(t)$$

with $\hat{\boldsymbol{b}}^m = \boldsymbol{Z}\hat{\boldsymbol{\gamma}}^m$.

The limitation of these approaches is that they are intended for use with continuous variables, so they have limited usefulness for the analysis of binary or categorical variables.

Functional Logistic Regression Model

The logistic regression model has been used for applications in many areas, including clinical studies, epidemiology, social sciences, marketing, and engineering. The root of binary logistic regression is a generalized linear model that uses a binomial distribution and a logit link function (Derr, 2013). Thus, binary logistic regression was developed to handle the case in which the response is binary, meaning that it can take only two values (the "event" and the "non-event"). In this case, the conditions for linear regression are not met because the responses are binomial and not normally distributed. Ideally, instead of modeling the response itself, the logistic regression model is constructed for modeling the log of odds that an event occurs or does not occur as the alternative. Thus, the odds of success are defined as the ratio of the probability of success over the probability of failure.

The functional logistic regression model with the logit link is a particular case of the functional generalized linear model proposed by James (2002). This section is a review of the development of the logistic regression model for predicting a binary response variable in the functional case, known as the Functional Logistic Regression (FLR) model (Escabias et al., 2004). The FLR appears in many applications throughout the literature that have been developed in recent years with different objectives. Escabias et al. (2005) used FLR to establish the relationship between the risk of drought and time evolution of temperatures. A. M. Aguilera et al. (2008) estimated the probability of lupus flares from the time evolution of stress level measurements. With the same objective, Müller and Stadtmüller (2005) have expanded asymptotic inference for a class of functional generalized linear model based on approximating the predictor process with a truncated Karhunen-Loéve expansion.

The main contribution of the FLR model is its usefulness for predicting and modeling the relationship between a binary response variable Y, or equivalently, the probability of occurrence of an event, in terms of a functional covariate $X(t)$ (i.e., a continuous variable that has been measured repeatedly over time). In the FLR model, the probability, π_i, of the occurrence of an event, $y_i = 1$, rather than the event $y_i = 0$, conditional on a a random sample $x_i(t)$ of functional covariates $X(t)$ is expressed as:

$$\pi_i = \frac{\exp\{\alpha + \langle x_i, \beta \rangle\}}{1 + \exp\{\alpha + \langle x_i, \beta \rangle\}} = \frac{\exp\{\alpha + \int x_i(t)\beta(t)\mathrm{d}t\}}{1 + \exp\{\alpha + \int x_i(t)\beta(t)\mathrm{d}t\}} \tag{2.9}$$

with α being a real parameter and $\beta(t)$ the parameter function of the model that belongs to the space $L^2(T)$. Equation (2.9) can be expressed in terms of the logit transformation, $l_i = \log[\pi_i/(1-\pi_i)]$, $\forall\ i = 1, \ldots, n$, as

$$l_i = \alpha + \int_T x_i(t)\beta(t)\mathrm{d}t, \tag{2.10}$$

$$\boldsymbol{l} = \mathbf{1}\alpha + \boldsymbol{A\Psi b}, \tag{2.11}$$

with $\boldsymbol{l} = (l_1, \ldots, l_n)'$ is vector of logit transformations, $\mathbf{1} = (1, \ldots, 1)'$. Also, $\boldsymbol{A\Psi}$ is the design matrix has as rows the vector of basis coefficients of the i^{th} sample curve, and $\boldsymbol{\Psi} = (\psi_{jk})$ is the $p \times p$ matrix that has as entries the usual inner products between basis functions. Finally, $\boldsymbol{b}$ is the vector of the parameter function basis coefficients. So the FLR model can be seen as a particular case of the functional generalized linear model introduced in James (2002). The conditional distribution of Y given $X(t)$ is a Bernoulli distribution that belongs to the exponential family.

The interpretation of the parameter function is drawn from formula (2.10). That is, " the integral of the parameter function multiplied by a constant K, can be interpreted as the multiplicative change in the odds of response $Y = 1$ obtained when a functional observation is incremented constantly in K units along T " (Escabias et al., 2005, p. 4892). However, the FLR model faces two issues faced by other linear models. First, the functional observations occur across a finite set of discrete time points. Secondly, it is impossible to estimate the infinite (non-numerable) parameter function with a finite number of observations n. The usual solution for these pointed questions in functional logistic regression as a functional linear model is to express both the sample curve $X(t)$ and the parameter function $\beta(t)$ in terms of basis functions that belong to the same finite space.

In order to avoid the multicollinearity and dimensionality problem in FLR, different algorithms for computing the PLS components for the FLR model have been developed, such as the Nonlinear Iterative Partial Least Squares (NIPALS) algorithm (Geladi and Kowalski, 1986); Statistically Inspired Modification of PLS (SIMPLS), which has fewer intuitive constraints than NIPALS (De Jong, 1993); and powered PLS (Indahl, 2005). Garthwaite (1994) compared PLS with different methods for solving multicollinearity in linear regression including PLS, PCR, ridge regression, variable subset selection, ordinary least squares, forward variable selection, and a Stein shrinkage method. This author concluded that PLS is suited to models with numerous variables and large error variances. Escabias et al. (2004) proposed several FPC methods for selecting the PCs to be included in a logit model. Preda and Saporta (2005) proposed the application of PLS discriminant analysis to the FLR model; later on, A. M. Aguilera et al. (2010) used basis expansions for estimating functional PLS regression.

Escabias et al. (2007) applied the PLS algorithm for generalized linear regression proposed by Bastien, Vinzi, and Tenenhaus (2005) for the particular case of the logit model to functional logistic regression. In addition, Garthwaite (1994) provides an interpretation of the PLS components such that they can be viewed as weighted averages of predictor variables, where each predictor holds the residual information in an explanatory variable that is not contained in earlier components.

PLS for logistic response. Traditionally PLS was designed for use with continuous response variables. When the response variable is not continuous, the ordinary PLS method does not apply directly. The modification to classical PLS regression for use with non-continuous response variables is PLS logistic regression, which has been studied in detail by Bastien et al. (2005). Classical PLS univariate regression results from an iterated use of ordinary least squares, and PLS generalized linear regression retains the rationale of PLS while the criterion optimized at each step is based on maximum likelihood.

The technique for PLS was originally developed in high-dimensional and collinear multivariate settings, and it is especially popular in chemometrics (Wold, 1975). Also, C.-Y. Wang et al. (1999) proposed a probability-based multivariate algorithm combining partial least squares and logistic regression to identify the development stages of oral cancer by analyzing the auto fluorescence spectra of oral tissues. These authors used leave-one-out cross-validation to determine the number of PLS components and to evaluate the performance of the algorithm.

PLS for logistic regression is a two-stage process. The original PLS procedure was first used for dimension reduction where the response variable was either 0 or 1, and then Logistic Discrimination (LD) was applied to the chosen PLS components for classification (C.-Y. Wang et al., 1999). Meanwhile, Frank and Friedman (1993) compared LD with quadratic discriminant analysis. These researchers applied their method to various datasets involving human tumor samples and analyzed the stability of the classification results. Nguyen and Rocke (2002) have completed two-stage PLS regression on microarray gene expression data. Although these authors obtained positive results for two-group classification, their approach was designed for a continuous outcome variable with constant variance, so it may not be ideal.

Functional Partial Least Squares for Logistic Response Variables

For the functional logit model with binary response variables, a PLS estimation approach was recently introduced and compared with functional principal component logistic regression (Escabias et al., 2007). The PLS approach is employed in a variety of other functional data problems as well. For example, Ferraty and Vieu (2006) used it to define a semimetric for nonparametric functional predictors or classifiers, and Delaigle and Hall (2012) used it for functional data classification. However, the literature of PLS is very diverse. Numerous studies have generally indicated that FPLS is better suited for logistic problems and has more accurate estimation than FPCR because the influence of the response is considered in extracting FPLS components (Aguilera at el., 2015; Escabias et al., 2007). Also, PLS has been viewed as having advantages over PCA for regression and logistic problems in both multivariate and functional data analysis (Aguilera et al., 2010). Reiss and Ogden (2007) have studied PLS for functional logistic regression via estimating the conditional distribution of Y/X. The main problem with FPCA for regression for classification is that it ignores the relationship between the predictor and response, which has been emphasized throughout this work.

The functional PLS logit estimation in terms of the original predictors. The FLR model fits the response variable y_i on the retained PLS components. Also, $\boldsymbol{\Gamma}$ is the matrix of logit PLS components of the design matrix $\boldsymbol{A\Psi}$, and $\boldsymbol{\Gamma} = \boldsymbol{A\Psi V}$, with $\boldsymbol{V}$ being the matrix whose columns are the vector of coefficients of the logit PLS components in terms of the original predictors. Then, the multiple model, (2.11), can express the logit model in terms of the PLS components as

$$\widehat{\boldsymbol{L}} = \mathbf{1}\widehat{\alpha} + \boldsymbol{\Gamma}\widehat{\boldsymbol{\gamma}}, \tag{2.12}$$

where $\widehat{\boldsymbol{\gamma}} = (\widehat{\gamma}_1, \ldots, \widehat{\gamma}_s)'$ are the maximum likelihood estimators of the vector of the coefficients of the logit model in terms of the logit PLS components. Finally, the estimation of the parameter function is obtained as: $\widehat{\beta}(t) = \widehat{\boldsymbol{b}}'\Phi(t)$, in terms of its basis coefficients estimated from the gamma parameters $\widehat{\boldsymbol{b}} = \boldsymbol{V}\widehat{\boldsymbol{\gamma}}'$.

Logistic Models for Multinomial Responses

This section is focused on generalizing a binary logistic regression model to allow modeling a response variable with more than two categories. When a categorical dependent variable has more than two possible responses (is not binary), a multinomial distribution and different link functions address the nature of the responses with different linear predictors to model the probabilities. J. Friedman, Hastie, and Tibshirani (2010) considered classifying data into three or more groups using the multinomial logistic regression model.

For this type of categorization, Park and Hastie (2007) used a generalized linear model (GLM) to do multinomial logistic regression where the random component is the multinomial distribution. The systematic components are explanatory variables, which can be continuous, discrete, or both. The goal of using a multinomial response is to model ordered behavior while considering whether covariates have common slopes across response functions Derr (2013). Also, multinomial logistic regression can use individual characteristics as predictor variables. "The multinomial model can be written in the same form as the conditional logit model " (Agresti, 2007, p. 316-317).

On the other hand, different types of logit transformations can be used depending of the type of response (nominal or ordinal). When analyzing a multinomial response, it is important to note whether the response is ordinal categories (e.g., satisfaction ratings with 1=very poor ... 5 = very pleased), or nominal consisting of unordered categories (e.g., race with 1 = White, 2 = African American, 3 = Hispanic) Some types of models are appropriate only for ordinal responses, such as the cumulative logits model, the adjacent categories model, or the continuation ratios model. Other models may be used regardless of whether the response is ordinal or nominal such as the baseline logit model, and the conditional logit model. However, the estimation methods for odds ratios for a multinomial response have a similar manner to the logistic models.

Nominal response variable. If a response variable takes values that have no order, such as voting (Democratic, Green, Independent, Republican), then it is nominal. Typically, nominal logit models have one response vector $\boldsymbol{y}$ with S components and $\pi_{i1}, ..., \pi_{iS}$ are the probabilities for a randomly chosen individual to fall into categories $1, ..., S$, respectively. The n independent observations falling into the different categories have a multinomial distribution. One or more explanatory (predictor) variables may be quantitative, qualitative, or both. Furthermore, the set of explanatory variables $X = (x_1, ..., x_k)$ can be discrete, continuous, or a combination of both. However, any of the categories can be chosen to be the baseline (reference) and the choice can be the first, last, or the most common category; otherwise a software program can do it automatically (an arbitrary process). Then, the process pairs the probability of being a member of a group in another response category to the probability of membership in the baseline category for the purpose of computing odds ratio.

The nominal logit regression process also simultaneously models all relationships between probabilities for the pairs of categories. This is done by modeling the odds of falling within one category instead of another (baseline). So, multi-category logit models for nominal response variables define all of the $S-1$ log odds for all pairs of categories, given a particular choice of $S-1$, the rest are redundant. The baseline-category logits for each nominal response paired with a baseline category can be written as

$$l_{is} = \alpha_{is} + {x_i}'\beta_{is}, \quad s = 1, 2, \ldots, S-1.$$

Typically, the maximum likelihood method is used for estimating the parameters. The β_{is} can be interpreted as the increase in log-odds of falling into category s versus category S resulting from a one-unit increase in the k^{th} covariate, holding the other covariates constant. Since any of the categories can be chosen to be the baseline, the model is fitted equally well, achieving the same likelihood and producing the same fitted values. Only the values and interpretation of the coefficients is changed.

The multinomial model is a type of GLM, so the overall goodness-of-fit statistics and their interpretations and limitations still apply. Wald tests are used to test whether a predictor variable is related to the response variable or if the parameters for two or more categories of the response variable are the same (Agresti, 2007). From this point of view, the Newton-Raphson method yields the ML parameter estimates in case of nominal logit since the log likelihood is concave. A Fisher scoring algorithm is used for iterative calculation of ML estimates in the case of cumulative logits (Agresti, 2007, 2010).

Functional Multinomial Logit Models for Nominal Responses

This section is an introduction to the general scheme of the functional logistic regression (FLR) model for predicting a binary response. The purpose of FLR is modeling the relationship between a functional predictor whose sample information is given by a set of curves that vary over time and a binary response variable. In this line of research, the Functional Multinomial Logit Regression (FMLR) model is used where $\boldsymbol{y}_i$ is the response vectors with S categories, takes a finite set of categories greater than two and the predictor is functional. This section is a presentation of FMLR models used for curves classification into more than two groups where the type of response could be nominal or ordinal. FMLR models allow modeling variables from a set of predictors and the interpreting of the relationship between the categorical response and the functional predictor via the parameters of the model.

Recently, the FMLR model has been applied in research from many different disciplines. Ferraty and Vieu (2003) used nonparametric FDA methodologies for curve classification of spectrometric food data. From the classical point of view of functional regression methods, Preda, Saporta, and Lévéder (2007) used FDA based on PLS to classify the quality of cookies based on the resistance of the dough. Aguilera-Morillo et al. (2013), presented a review of different calibration and classification methods for functional data in the context of chemometric applications. Following previous research, Aguilera et al. (2014) proposed a functional PCA and baseline category logit model in order to predict the relationship between predictors and a multi-category response variable.

Typically, this model is a particular case of a generalized linear model in which the link functions can take different types of logit transformations, such as the baseline logit for nominal responses, or cumulative, adjacent-categories and continuation-ratio logits for the ordinal response variables (Agresti and Kateri, 2011).

Functional multinomial response model. The multinomial response model is a particular case of a functional generalized linear model with

$$g_s(\boldsymbol{\mu}_i) = \alpha_s + \int_T x_i(t)\beta_s(t)dt, \tag{2.13}$$

where $g_s(\boldsymbol{\mu}_i)$ is the link functions for different types of logit transformations, and α_s and $\beta_s(t)$ are a set of parameters to be estimated (Agresti and Kateri, 2011). Also, the link function components $g_s(\boldsymbol{\mu}_i)$ can be defined in different ways. Thus, the FMLR model for nominal response pair each response with a baseline category, and they can be written as

$$l_{is} = \log\left(\frac{\pi_{is}}{\pi_{iS}}\right), \quad s = 1, \ldots, S-1, \quad i = 1, \ldots, n,$$

then, the above equation that expresses the multinomial logit models directly in terms of response probabilities is

$$\pi_{is} = \frac{\exp\{\alpha_s + \int_T x_i(t)\beta_s(t)dt\}}{\Sigma_s \exp\{\alpha_s + \int_T x_i(t)\beta_s(t)dt\}}$$

with $\alpha_S = 0, \beta_S(t) = 0$.

The estimation of the parameter of the functional logit model presents the same problems that occur with the functional linear model. In the case of basis expansion of the sample curves and the parameter function, the FMLR model is equivalent to a multiple logit regression model given by

$$l_{is} = \alpha_s + \int_T x_i(t)\beta_s(t)dt \tag{2.14}$$

Estimation for the functional multinomial logit models. The inherent problems in parameter estimation in functional data analysis (infinite dimension of the predictor space, discrete time observations, and individual differences) are solved by reducing dimension via performing an orthonormal basis expansion of the functional predictor (Ramsay and Silverman, 2005). Then, a first estimation of the parameter functions can be obtained by considering that both the predictor curves and functional parameters belong to a finite space generated by a basis function. However, it is mandatory to obtain the sample curve basis coefficients from the sample information.

Typically, in the case of FMLR, the parameters are estimated by least squares or maximum likelihood (ML). Müller and Stadtmüller (2005) have studied the dimension reduction approach in the theoretical framework of functional generalized linear models where asymptotic tests and simultaneous confidence bands for the parameter function have been obtained. On the other hand, Marx and Eilers (1999) studied B-spline expansion of the functional parameter as an alternative estimation procedure for the functional parameter (but not the predictor curves) that maximizes the penalized log-likelihood for a functional binomial response model. Cardot and Sarda (2005) examined the procedure in the general context of functional generalized linear models. A. M. Aguilera et al. (2008) and Escabias et al. (2007) have considered natural and quasi-natural cubic spline interpolation to generate estimates for functional PCA. This procedure is accomplished by using the slope parameters of the model, that in the case of the functional multinomial logit model are a set of functions. With this objective in mind, an accurate and interpretable estimation of these functional parameters is very important in functional data analysis.

The FPCA and FPLS methods have been constructed again in multinomial logit regression in order to provide an accurate estimation of the parameter functions. The FPCA and FPLS regression methods are used to estimate the functional parameters in terms of basis expansions of the sample curves (Escabias et al., 2004, 2007; Preda et al., 2005). The next section includes an overview of estimation methods based on FPC and FPLS regression and a discussion of their advantages and disadvantages in terms of the FMLR model.

Functional Principal Component Estimation of Functional Multinomial Logit Regression Model

FPC method is a dimension reduction technique that explains the dependence structure of a functional data set in terms of uncorrelated variables. Typically, the estimation method for FLR based on the FPCA of sample curves has been generalized to the case of a multi-category response. Escabias et al. (2014) proposed different FPCA approaches for solving the multicollinearity problem in the FMLR model by using a set of functional principal components of the functional predictor as covariates of the multiple multinomial regression model, and Massy (1965) followed the idea of PCR in proposing the use of a reduced set of FPCA of the sample curves of the baseline category response model as regressors. The two different FPCAs of the functional predictor considered make efforts to improve the estimation of the functional parameters in the sense of smoothness.

Escabias et al. (2014) introduced two different methods for selecting the number of components, which take into account both their explained variability and their ability to predict the response while providing the best estimation of the functional parameters. These authors follow the methodology considered in A. M. Aguilera et al. (2006) for multiple binary logit models, which was extended to the case of the functional multinomial logistic model. The first method consists of including principal components in the order given by their explained variability (variability order). In the second method, PCs are included by a forward stepwise method based on conditional types of criteria that can be fixed based on the optimum number of principal components: the response prediction-type criterion and the functional parameters-type criterion. Then, a likelihood ratio test takes into account their relationship with the response variable (stepwise method). To obtain an estimate of the functional slope of the model, the standard functional principal component regression estimation method regresses the response on the principal component scores linked with the largest eigenvalues of the functional predictor covariance operator.

In relation to the optimum number of principal components to retain, two methods have been compared. The first is the classical method used in principal component regression based on minimization. The leave-one-out cross-validation mean squared error of prediction (CVMSE) and the leave-one-out cross-validation correct classification rate (CVCCR) are defined as the rate of agreements between the observed category for an individual and the predicted category (that associated with the highest predicted probability) based on the model. These values are estimated without taking the individual into account in the process (A. M. Aguilera et al., 2008).

Different measures have been used to evaluate the accuracy of estimated functional parameters and the fit of the different considered models. Goodness of fit has been measured in terms of the Probabilities Mean Square Error (PMSE). Additionally, an accuracy measure of the estimation of the functional parameters can be determined by the mean of the Integrated Mean Square Error (IMSE) of the parameter functions and the estimated variance of the estimated parameter functions given by the mean of the sum of variances of intercepts and parameter function basis coefficients (Aguilera et al., 2014). In general, FPCA is good solution for representing data in a reduced-dimension space, but when a scalar variable is observed, a Functional Partial Least Squares approach allows more direct use of this information.

Generalized Partial Least Squares Estimation Method for the Functional Multinomial Logit Regression Model

As an alternative to FPC mothed, Preda and Saporta (2005) introduced a new functional regression method based on the PLS logit model that consists of adapting the classical PLSR algorithm introduced by (Wold, 1975) as an alternative to PCR for solving the multicollinearity problem and reducing the number of predictor variables in linear regression. In particular, A. Aguilera et al. (2016) introduced the penalized versions of functional PLS regression.

The expansion of PLS to sparsely observed functional data is of considerable interest not only as applied to classification but also to the more general context of regression problems, as shown in Escabias et al. (2007). Preda and Saporta (2005) studied LDA-PLS, and Delaigle and Hall (2012a) tested centroid-PLS. Furthermore, Preda et al. (2007) used this expanded approach for linear discriminant analysis purposes, where it was applied to the estimation of the functional linear model and used with the FPLS method. Aguilera-Morillo and Aguilera (2015) conducted an extensive simulation study comparing both methodologies with their multivariate versions with both equally spaced and irregularly spaced sampling points.

In conclusion, the objective of FPLS regression is to regress $\boldsymbol{Y}$ matrix on a set of uncorrelated random variables (FPLS components), considered as the linear spans coefficients of $\boldsymbol{\Gamma}$, using a transformation of the parameters of the simple logit fittings of the response $\boldsymbol{Y}$ on each single explicative variable as covariates, which takes into account the correlation between the response $\boldsymbol{Y}$ and the functional predictor $X(t)$.

Iteratively reweighted partial least squares (IRWPLS) for multinomial response. Since Wold (1975) introduction of PLS regression, there have been numerous studies that have sought to (a) improve PLS algorithms, (b) adapt PLS linear regression to the logistic regression model, and (c) extend PLS logistic regression to include functional covariates. Of particular interest to this work is the work of Escabias et al. (2007), in which they propose a functional PLS logit regression model to forecast a binary response variable from a functional predictor.

McCullagh and Nelder (1989) demonstrated that the maximum likelihood estimation of the parameters of generalized linear models via the Fisher scoring method can be rephrased as iteratively reweighted least squares. Also, in their work the dependent variable is a linearized form of the link function that is applied to the response variable. The parameter estimates are calculated via iterative updates of the adjusted dependent variable and weights until a convergence criterion is satisfied. Marx (1996) proposed incorporating PLS into the framework of generalized linear models using an iteratively reweighted PLS algorithm. In this approach, the weighted PLS steps are embedded in the iterative steps, and the process treats and updates the adjusted dependent variable z as the response as opposed to working with the original outcome. These nested loops are iterated until convergence is reached. In high-dimensional problems, separation often occurs, which Marx did not directly address.

The IRWPLS procedure has been extended to a multi-group classification scenario. By treating the classes as nominal without special ordering, the process is a generalization of logit models for binary responses (Fahrmeir and Tutz, 2001). However, Firth's procedure can be extended to the multinomial case, denoted by (MIRWPLSF) incorporated to address the nonconvergence problem frequently encountered in logistic regression. The dependent variable is a linearized form of the link function applied to the response variable,

$$\boldsymbol{z}_i = \boldsymbol{\eta}_i + \frac{\partial \boldsymbol{\eta}_i}{\partial \boldsymbol{p}_i}(\boldsymbol{y}_i - \boldsymbol{p}_i), \tag{2.15}$$

where $\boldsymbol{z}_i$ is the adjusted dependent vector as the response rather than working with the original outcome, $\boldsymbol{\eta}$ is the link function applied to the response variable, $\boldsymbol{p}_i$ is the covariate vector corresponding to the i^{th} logit , and $\boldsymbol{y}_i$ is the response vector for the i^{th} sample. As such, MIRWPLSF provides a more stable model than MIRWPLS. However, the literature of PLS is very diverse a large number of algorithmic variants of the process exist. The first approach for PLS was proposed by Ding and Gentleman (2005), and it can be seen as an adaptation of Marx (1996) IRPLS method that solves the problem of separation. In GLMs, the Newton-Raphson algorithm is usually used to maximize the log likelihood, which results in the Iteratively Re-weighted Least Squares (IRLS) method. Marx (1996) and Fort and Lambert-Lacroix (2005) adapted PLS for classification by using PLS to solve the weighted least squares problem arising within the IRLS method. Ding and Gentleman (2005) reported that G-PLS regression achieved lower classification error rates than two- stage PLSR.

The G-PLS procedure proposed by Ding and Gentleman (2005) carries out multi-group classification from a generalization of the PLS method to multi-categorical response variables. The procedure is based on the multinomial logit model and denoted as M-IRWPLS-F. The IRWPLS-F and M-IRWPLS-F are reported to achieve better classification performance. Usually, G-PLS demonstrates several advantages over other approaches:

1. It performs variable selection automatically.

2. t can be applied to diverse tasks, including classification, survival analysis, or modeling transcription factors activities.

3. It is statistically efficient.

4. It is computationally very fast, so it is practical for application to large data sets.

The Algorithm for Generalized Partial Least Square

The dimension reduction concept of PLS has been used with generalized linear models resulting in a new algorithm named G-PLS (Bastien et al., 2005). The basic concept of G-PLS is to apply generalized linear regression of the response on retained PLS components. Then, the original explanatory variables are expressed in terms of the latent components. In the class of G-PLS models, some examples are logistic PLS and ordinal PLS (Chen, Phan, and Reutens, 2010). The process includes the following steps:

1. Computation of the m G-PLS components. This step is described in details in next section.
2. Generalized linear regression of $\boldsymbol{y}_i$ on the m retained PLS components.
3. Expression of PLS-GLR in terms of the original explanatory variables.

Computation of the PLS components. Let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_p]$ be the matrix of numerical or categorical explanatory variables, and assumed to be centered p explanatory variables $\mathbf{x}_j$'s. The objective is to search for m PLS orthogonal components $\mathbf{t}_h$'s defined as linear combinations of $\mathbf{x}_j$ obtained as follows:

Computation of the first PLS component $\mathbf{t}_1$.

Step 1: Compute the regression coefficient a_{1j} of $\mathbf{x}_j$ in the generalized linear regression of $\mathbf{y}$ on $\mathbf{x}_j$ for each variable $\mathbf{x}_j$, $j = 1$ to p,

Step 2: Normalize the column vector $\boldsymbol{a}_1$ made by a_{1j}'s: $\mathbf{w}_1 = \boldsymbol{a}_1/||\boldsymbol{a}_1||$,

Step 3: Compute the component $\mathbf{t}_1 = \mathbf{X}\mathbf{w}_1/\mathbf{w}_1'\mathbf{w}_1$.

Computation of the second PLS component $\mathbf{t}_2$.

Step 1: Compute the regression coefficient a_{2j} of $\mathbf{x}_j$ in the generalized linear regression of $\mathbf{y}$ on $\mathbf{t}_j$ and $\mathbf{x}_j$ for each variable $\mathbf{x}_j$, $j = 1$ to p,

Step 2: Normalize the column vector $\boldsymbol{a}_2$ made by a_{2j}'s: $\mathbf{w}_2 = \boldsymbol{a}_2/||\boldsymbol{a}_2||$,

Step 3: Compute the residual matrix of $\mathbf{X}_1$ of the linear regression of $\mathbf{X}$ on $\mathbf{t}_1$,

Step 4: Compute the component $\mathbf{t}_2 = \mathbf{X}_1\mathbf{w}_2/\mathbf{w}_2'\mathbf{w}_2$,

Step 5: Express the component $\mathbf{t}_2$ in terms of $\mathbf{X}$: $\mathbf{t}_2 = \mathbf{X}\mathbf{w}_2^*$, where $\mathbf{w}_2^*$ is the $\text{Cov}(\mathbf{XY}, \mathbf{w})$.

Computation of the h^{th} ***PLS component*** $\mathbf{t}_h$. In the previous steps, the PLS components $\mathbf{t}_1, \ldots, \mathbf{t}_{h-1}$ have been yielded. The component $\mathbf{t}_h$ is obtained by iterating the search for the second component.

Step 1: Compute the regression coefficient a_{hj} of $\mathbf{x}_j$ in the generalized linear regression of $\mathbf{y}$ on $\mathbf{t}_1, \ldots, \mathbf{t}_{h-1}$ and $\mathbf{x}_j$ for each variable $\mathbf{x}_j$, $j = 1$ to p.

Step 2: Normalize the column vector $\boldsymbol{a}_h$ made by a_{hj}'s: $\mathbf{w}_h = \boldsymbol{a}_h/||\boldsymbol{a}_h||$,

Step 3: Compute the residual matrix $\mathbf{X}_{h-1}$ of the linear regression of $\mathbf{X}$ on $\mathbf{t}_1, \ldots, \mathbf{t}_{h-1}$,

Step 4: Compute the component $\mathbf{t}_h = \mathbf{X}_{h-1}\mathbf{w}_h/\mathbf{w}_h'\mathbf{w}_h$,

Step 5: Express the component $\mathbf{t}_h$ in terms of $\mathbf{X}$: $\mathbf{t}_h = \mathbf{X}\mathbf{w}_h^*$.

On the other hand, an important problem in FMLR model is to select the optimum number of G-PLS components to be retained.

The Generalized partial least square component selection. In relation to the optimum number of G-PLS components to retain, the methods have been compared. Escabias et al. (2007) introduced different methods for selecting the number of components to be retained such as cross-validation procedure. These authors follow the methodology considered in A. M. Aguilera et al. (2006) for multiple binary logit models, which was extended to the case of the functional multinomial logistic model. Additional, the area under the Receiver Operating Characteristic (ROC) curve which is defined as the rate of correct response classifications (Menard, 2000) and (Mittlböck and Schemper, 1996). The ROC curve plots the proportion of data points that are correctly classified (the true positive rate) against the false positive rate to illustrate the different possible cut-off points for a diagnostic test. It shows the trade-off between sensitivity and specificity. The closer the ROC curve comes to the diagonal of the ROC space, the less accurate the test. The proposed method generates a graphic that shows the trade-off between the rate at which the model can correctly predict categories and the rate of incorrectly predicting the response classifications. Ultimately, the performance of the model using the ROC (also called AUROC) curve is as follows:

$$\text{AUROC} = 0.5 \text{ useless}$$

$$0.7 \leq \text{AUROC} < 0.8 \text{ acceptable}$$

$$0.8 \leq \text{AUROC} < 0.9 \text{ excellent}$$

$$\text{AUROC} \geq 0.9 \text{ outstanding}$$

So, the range from 0.50 to 1.00, and values above 0.80 indicate that the model does a good job in discriminating between the two categories that comprise the target variable. In conclusion, by consider the G-PLS, the primary goal of this dissertation is to develop a modified functional G-PLS logistic regression algorithm that can be used to predict a classification response with the FMLR model.

The remainder of this chapter introduces the focus of this dissertation. The proposed method applies the classical PLS estimation method to the case of a functional logistic regression model to improve estimation of the functional parameters of a FMLR model with nominal response resulting in a new model named Functional Nominal Multinomial Logit Regression (F-NM-LR). It is hypothesized that the F-MN-LR model based on the generalized PLS estimation method is more parsimonious than the alternative functional PCA and base-line logit models proposed by Escabias et al. (2014). This dissertation an effort to improve on principal components-based procedures, which do not take into account the relationship between response and predictor variables. Chapter III provides an explanation of how the proposed model provides various methodological contributions in statistics. An investigation of the performance of the generalized PLS method, via a simulation study and the analysis of a real dataset, is presented in Chapter IV. Curve classification functions are presented, and the discussion was emphasize the differences existing between F-MN-LR models using PCR and GPLS methods of parameter estimation. Finally, Chapter V contains the conclusion and how these methodological contributions improve actual estimation methods and production functions for a wide scope of applied scientific fields.

CHAPTER III

METHODOLOGY

This chapter illustrates the methodology used to develop the functional generalized partial least squares nominal multinomial logit regression model.

Introduction

The primary goal of this dissertation is to develop an alternative to the classical Functional Principal Components (FPC) regression method for the problem of predicting a categorical logit response variable associated to an observed curve. The proposed approach uses the Generalized Partial Least Squares (GPLS) regression method, while avoiding extreme local fluctuation in estimation of the functional parameters of the Functional Nominal Multinomial Logit Regression (F-NM-LR) model. Particularly, the author is interested in comparing the performance of the GPLS method under the F-NM-LR model to the one proposed by Escabias et al. (2014) based on FPC regression. To this end, the F-GPLS-NM-LR model is used to predict a multinomial logit response associated with a functional predictor. To be more meaningful, the F-GPLS-NM-LR model is viewed as a particular case of a functional generalized linear model. An important contribution of the GPLS method is that it easily generalizes to any functional logit regression model that is linear at the level of the functional predictor variable. The strength of using the GPLS method is that of producing more accurate estimations of the functional parameters and providing the precise probability of each sample curve's falling into a specific class, with the high predictive ability of the F-GPLS-NM-LR model.

Specifically, the GPLS and FPC regression methods of interest construct new predictor variables as linear combinations of the original predictor variable, but they construct their components differently. The FPC regression method creates components without considering the response variable at all to explain the observed variability in the predictor variables. On the other hand, the GPLS regression method takes the response variable into account, so this process often leads to models that are able to achieve fit for the categorical response variable with fewer components. The comparative performance of the proposed methodology is compared using both a simulation study and a real world dataset.

The following research questions are investigated:

Q1 How is a generalized partial least squares regression method developed for parameter function estimation in the functional nominal multinomial logit regression model?

Q2 How does the functional generalized partial least squares nominal multinomial logit regression model behave in terms of goodness-of-fit measures, such as the correct classification rate and the integrated mean squared error, with changes of the functional predictor dependent on arbitrary values assigned to a and b?

Q3 How does the functional generalized partial least squares nominal multinomial logit regression model behave in terms of goodness-of-fit measures, such as the correct classification rate and the integrated mean squared error, based on changes in the number of the nominal response categories?

Q4 How does the precision of the generalized partial least squares regression method compare to the principal component regression method proposed by Escabias et al. (2014) under the functional nominal multinomial logit regression model?

Q5 How to develop an R code to fit a functional generalized partial least squares nominal multinomial logit regression model to real data?

The rest of this chapter is organized as follows. First a brief summary of the theoretical framework of the F-NM-LR model within the context of a functional generalized linear model is presented, followed by an introduction to basis expansion approaches for the F-NM-LR model. Next, the PLS regression method and the GPLS-based estimation approaches within the context of the F-NM-LR model are discussed. Afterward, the algorithm for computing the GPLS for the F-NM-LR model and an overview of the Functional PCR-based solution is presented, followed by the F-NM-LR component estimation in terms of original variables. An overview of component selection with different estimation methods (PCR and GPLS) and goodness of fit measures is then presented. Finally, the scheme for a simulation study and the real dataset example that is used to compare the two methods is described.

Functional Nominal Multinomial Logit Regression Model

This section presents the development of an F-NM-LR model that can be seen as a particular case of a functional generalized linear model (James, 2002) with a finite set of categories for the response variable that is greater than two. Logit transformations are used depending on the nominal response variable. The purpose of the proposed model is to predict the class membership (multinomial response variable) that is associated with an observed curve (functional data).

Model formulation. In order to formulate the F-NM-LR model that is considered in this work, it is assumed that $X(t)$ a functional predictor and $\boldsymbol{Y}$ a matrix of response variable are defined on the same probability space. The sample curves $x_i(t)$ belong to the space $L^2(T)$ of squared integrable functions on T called Hilbert space, defined as

$$L^2(t) = \left\{ f : T \rightarrow \mathbb{R} : \int_T f^2(t)dt < \infty \right\}$$

where the usual inner product is defined as

$$\langle f, g \rangle_u = \int_T f(t)g(t)dt, \quad \forall f, g \in L^2(T).$$

Let $\{x_i(t) : t \in T, i = 1, \ldots, n\}$ be a random sample of observations (sample curve) of a functional predictor $\{X(t) : t \in T\}$ where T is some interval on the real line and each curve can be observed at a different time point t.

Let $\{(y_{i1}, \ldots, y_{iS})' : \; i = 1, \ldots, n\}$ be a set of n sampled from a categorical response vectors $\boldsymbol{y}_i$ associated with the S categories, defined for each $s = 1, 2, \ldots, S$ by

$$y_{is} = \begin{cases} 1 & \text{if category } s \text{ is observed for } X(t) = x_i(t) \\ 0 & \text{other case} \end{cases}$$

so that each observation is generated by a multinomial distribution,

$$\boldsymbol{y}_i \sim M(1; \pi_{i1}, \ldots, \pi_{iS}),$$

with

$$\pi_{is} = P[y_{is} = 1 | X(t) = x_i(t)] \ \text{ and } \ \sum_{s=1}^{S} \pi_{is} = 1 \ \forall i = 1, \ldots, n,$$

it means π_{is} is the conditional probability that subject i will have a response $y_{is} = 1$, given the functional covariate $x_i(t)$. Also $\pi_{is} \in [0, 1]$, $\ \forall \ i = 1, \ldots, n$.

Observe that y_{iS} is redundant, as in any multinomial logistic regression model (Agresti and Kateri, 2011), so it is denoted by $\boldsymbol{y}_i = (y_{i1}, \ldots, y_{i,S-1})'$ the response vector for subject i, with mean vector $\boldsymbol{\mu}_i = E[\boldsymbol{y}_i] = (\pi_{i1}, \ldots, \pi_{i,S-1})'$.

The π_{is} only takes values on the interval [0, 1], but linear functions are unbounded. To deal with this problem, the logit transformation of π_{is} as the response is applied. The multinomial logit regression model for nominal responses can be extended to the functional case as a Functional Generalized Linear Regression (FGLR) model, as introduced in James (2002), whose link function is given by the nominal multinomial logit transformations l_{is} that pair each response with a baseline category (usually the last one),

$$l_{is} = \log\left(\frac{\pi_{is}}{\pi_{iS}}\right). \tag{3.1}$$

Then equation (3.1) can be expressed alternatively in terms of the logit transformation as

$$l_{is} = \alpha_s + \int_T x_i(t)\beta_s(t)dt, \quad s = 1, 2, \ldots, S-1, \tag{3.2}$$

with l_{is} is logit transformations that pairs each response with a baseline category, α_s is a real parameter, $x_i(t)$ are random samples of observations of a functional variable $X(t)$, and $\beta_s(t)$ is the parameter function of the model that belongs to the space $L^2(T)$, which need to be estimated.

Then, the probabilities of the nominal multinomial response are modeled in terms of the functional predictor and the parameters as

$$\pi_{is} = \frac{\exp\{\alpha_s + \int_T x_i(t)\beta_s(t)dt\}}{\Sigma_s \exp\{\alpha_s + \int_T x_i(t)\beta_s(t)dt\}}, \tag{3.3}$$

for $s = 1, 2, \ldots, S$, $i = 1, 2, \ldots, n$, with $\alpha_S = 0$, $\beta_S(t) = 0$.

Essentially, equation (3.2) leads to an interpretation of the relationship between the nominal response and the functional predictor. For example, “ the exponential of the integral $\int_{t_0}^{t_0+h} \beta_s(t)g(t)dt$ of the parameter function is the multiplicative change in the odds of response $(y_s = 1)$ against response $(y_S = 1)$ provided by a change of the curve $x_i(t)$ according to a function $g(t)$ in the interval $[t_0, t_0 + h]$ ” (Escabias et al., 2014, p. 299), obtained when a functional observation is incremented constantly in K units along T. The F-NM-LR model is a multi-equation model, similar to multinomial logistic regression. So $(S-1)$ logit equations, with $(S-1)$ log odds of each model are required.

As in the case of the functional linear and logistic regression model, the use of the least squares criterion for estimating the model in equation (3.2) has to take into account different aspects:

- First, the functional predictor $x_i(t)$ cannot be observed continuously in time but at a set of discrete time points that could be different for each sampled individual (Aguilera-Morillo and Aguilera, 2015; Delaigle and Hall, 2012).
- Second, due to the infinite dimension of the predictor space, which in general is not invertible (i.e. does not propose a unique solution), estimating the parameter function $\beta_s(t)$ is not possible (that is infinite non-numerable) with a finite number of observations (Escabias et al., 2007).

Basis expansion methods are the most recognized solution within functional regression methods to overcoming these two problems simultaneously. In this study, the approach is to reduce dimensions by performing an orthonormal basis expansion of the functional predictor and the parameter. This dimension reduction approach has been studied by Müller and Stadtmüller (2005) in the theoretical framework of functional generalized linear models and has been presented in Chapter II. The next section discusses a basis expansion approach in terms of the F-NM-LR model.

Basis Expansion of Functional Nominal Multinomial Logit Regression (F-NM-LR) Model

In practice, besides the impossibility of direct estimation of the functional parameter, normally the functional predictor can only be observed in a set of discrete time points called knots $\{t_{ik} : k = 1, ..., m_i\}$ for each sample curve $x_i(t)$.

Because of this factor, sample curves can have different time points for each sampled individual. Thus, the first step in FDA is often reconstructing the functional form of data from discrete observations using basis functions. Basis functions allow the representation of complicated functions in a relatively simple form. However, as mentioned in Chapter II, there are many possible choices for basis functions depending on the character of the observed curves.

In general, an appropriate basis function should be chosen according to the main features of the sample curves. This work considers only the B-Spline basis system because it is widely used with functional regression models. B-splines are piecewise polynomials of degree m, and requiring continuity at the knots ensures smoothness of the function. This work uses fourth degree polynomials (i.e. cubic B-Spline) with the knots unequally spaced within the interval of observation.

Regardless of the basis function system that is chosen, this method was simultaneously solve the two-pointed problem in the F-NM-LR model, by considering that both the predictor function $x_i(t)$ and parameter function $\beta_s(t)$ belong to the same finite space spanned by an orthonormal basis of functions. However, in spite of having considered the same type of basis for sampled curves and functional parameters, they could differ. If the functional predictor and the parameter function are represented as a linear combination of basis functions, the basic concept is as follows:

1. Let $\Phi(t) = (\phi_1(t), \ldots, \phi_p(t))'$ be a basis functions that generate the space where $x(t)$ belongs. Then

(a) $x_i(t) = \boldsymbol{a}_i'\Phi(t)$, where $\boldsymbol{a}_i = (a_{i1}, \ldots, a_{ip})'$ are the vectors of basis coefficients of the sample curves.

(b) $\beta_s(t) = \boldsymbol{b}_s'\Phi(t)$, where $\boldsymbol{b}_s = (b_{s1}, \ldots, b_{sp})'$ are the vectors of basis coefficients of the parameter function.

This way the F-NM-LR model in equation (3.2) becomes a multiple linear model for the response variable in terms of a transformation of the functional predictor and parameter using basis coefficients given by:

2. $l_{is} = \alpha_s + \int_T \; \boldsymbol{a}_i'\Phi(t)\boldsymbol{b}_s'\Phi(t)dt = \alpha_s + \boldsymbol{a}_i'\boldsymbol{\Psi}\boldsymbol{b}_s, \;\; s = 1, \ldots, S-1, \, i = 1, \ldots, n$,

   with $\boldsymbol{\Psi} = \boldsymbol{\psi}_{vu} =< \phi_v(t), \phi_u(t) >$ is the $p \times p$ matrix of inner products between basis functions.

3. In the matrix form, Escabias et al. (2014) showed that under these conditions, the F-NM-LR model as given is equivalent to the following multivariable logistic regression model

$$\boldsymbol{l}_s = \boldsymbol{\alpha} \otimes \mathbf{1} + \boldsymbol{A}\boldsymbol{\Psi}\boldsymbol{B}. \tag{3.4}$$

Also, the F-NM-LR model can be expressed as

$$\boldsymbol{L} = \boldsymbol{\alpha} \otimes \mathbf{1} + \boldsymbol{H}\boldsymbol{B}, \; s = 1, \ldots, S-1, \tag{3.5}$$

where

(a) $\boldsymbol{L} = (\boldsymbol{l}_1, \ldots, \boldsymbol{l}_{S-1})'$, the matrix of $n \times (S-1)$ logit transformations.

(b) $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{S-1})$, is a $(S-1) \times 1$ vector of the intercept parameter.

(c) $\boldsymbol{H} = \boldsymbol{A\Psi}$ is an $n \times p$ design matrix whose product of the matrix of sample curve basis coefficients and the matrix of the inner products.

(d) $\boldsymbol{A}$ is a $(n \times p)$ matrix of the sample curve basis coefficients.

(e) $\boldsymbol{\Psi}$ is a $p \times p$ matrix of the inner products of between basis functions.

(f) $\boldsymbol{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{S-1})'$, is a $(S-1) \times p$ matrix of the parameter function basis coefficients for the multivariable logit model (3.5).

Then, the estimation of equation (3.5) is provided by maximizing the multinomial log likelihood under equation (3.3). However, the estimation procedure of the parameter function basis coefficients of equation (3.5) has the following issue that must be dealt with:

1. The sample curves basis coefficients are estimated from discrete-time observations $(t_{i1}, ..., t_{im})$ by using an appropriate numerical method. An interpolation (data observed without error) or least squares approximation (noisy data) can be used to compute these basis coefficients in practice. For example, Escabias et al. (2005) proposed quasi-natural cubic B-spline interpolation for reconstructing annual temperature curves from monthly values, and it has been introduced to estimate the risk of drought from the time evolution of temperatures (Escabias et al., 2005). Natural cubic spline

interpolation was first considered to estimate functional PCR by A. M. Aguilera et al. (1996).

On the other hand, if the functional predictor is observed with errors then least squares smoothing is used. For example, least squares approximation with both B-splines and trigonometric functions has been used for interpreting the relationship between time evolution of stress and flares in Systemic Lupus Erythematous patients (Aguilera et al., 2008). Also, least squares smoothing on cubic B-splines is used in the application developed to approximate dough resistance curves during the kneading process and spectrometry curves of fine chopped meat pieces (Aguilera et al., 2010).

In this study, the true functional forms of the curves are reconstructed via natural cubic spline interpolation under the F-NM-LR model adaption from Escabias et al. (2005; 2007; 2014) as,

$$x_i = x_i(t_{ik}) \quad k = 1, ..., m_i.$$

2. The estimation of the basis coefficients of the functional parameter under equation (3.6) is provided by maximizing the multinomial log likelihood under equation (3.3). The most widely used method that yields the ML parameter estimates is a Newton-Raphson because of the concavity of the log-likelihood equation. However, when logistic regression is performed, the likelihood estimation of the parameter function of this model inaccurate, as proved by

Ramsey and Silverman (1997). Inaccurate estimation of the parameter is due to the following issues:

(a) The first issue that must be dealt with is multicollinearity of the covariates. As established in the previous sections, basis expansion estimation of logit models usually provides good predictions of the response and consequently a fair classification rule, but Aguilera et al. (2006) demonstrated that it results in inaccurate parameter estimation with high variability due to a strong correlation between the columns of the design matrix (multicollinearity). These inaccuracies make it difficult to interpret the true relationship between variables, so the intent of this research is to strike a balance between estimation of the functional parameters of the model and accurate prediction of the response.

(b) The second issue is high dimensionality. The number of basis functions used in the approximation of the sample curves could be higher than the number of observations.

To solve the multicollinearity and dimensionality issues and to obtain an accurate estimation of the parameter function, a well-known solution in functional data analysis is to use a reduced set of uncorrelated components instead of the columns of the $\boldsymbol{H}$ design matrix as regressors. In the particular case of binary and multinomial response variables, different approaches based on the Functional Partial Least Squares (FPLS) or Functional Principal Component (FPC) regression that agree with orthonormal basis functions are usually used (Escabias, Aguilera, and

Valderrama, 2007; Escabias et al., (2014). This work was consider the PLS approach and compare its efficiency to the FPC regression method of obtaining accurate estimations for both the functional parameter and the categorical nominal response.

The next section is an introduction to a PLS method that agrees with orthonormal basis functions as one of the most efficient solutions to the inverse problem in the framework of functional data, which has been investigated in several studies (Aguilera et al., 2010; Aguilera-Morillo and Aguilera, 2015; Delaigle and Hall, 2012; Escabias et al., 2007; Preda et al., 2005).

Partial Least Squares Regression Method

The PLS method is used in regression to find the direction in the response that explains the maximum variance of the direction in the predictive space. This technique was originally developed as a multivariate linear regression method to deal with a large number of predictors, small sample size, and high collinearity among predictors (Wold, 1975). The PLS components are defined as uncorrelated linear spans of the latent variables of the regression model that maximize the covariance between the predictors in the columns of design matrix and the response variables as a solution to Tucker's criterion (Tucker, 1938). These ideas have been described in Chapter II. That is why PLSR is called a supervised method in contrast to the FPC method, which visualizes the relation between object and function predictor variables by maximizing the variance of $X(t)$ without taking into account the response variable for the construction of the new components.

There have been numerous studies that have looked to improve PLS algorithms. Bastien et al. (2005) adapted PLS linear regression to generalized linear models, and Preda et al. (2005) extended PLS logistic regression to include functional covariates and used it for classification purposes in Preda et al. (2007) with the name PLS classification of functional data. Escabias et al. (2007) proposed a functional PLS logit regression model to forecast a binary response variable from a functional predictor, and Aguilera et al. (2015) adapted PLS logistic regression to include functional covariates. However, the literature regarding PLS is very diverse, and numerous studies have generally believed that PLS is better suited for classification problems and produces more accurate estimation than PCR (Aguilera et al., 2015; Escabias et al., 2007).

On the other hand, the PLS method was originally designed for continuous response variables to estimate the slope parameter in multivariate parametric models. The next section includes a brief summary of the main objective of this study, which is the proposal of the generalized PLS estimation procedure as a basic estimation approach for the F-NM-LR model that produces accurate estimation of functional parameters and the precise probability of each sample curve's falling into a specific class. Without loss of generality and in order to clarify the theoretical aspects, curves are considered to be centered.

The Generalized Partial Least Squares Estimation Approach

The purpose of this research is to develop an alternative to the traditional PCR method by using a Generalized PLS (GPLS) method for estimating the parameter functions in the F-NM-LR model. The proposed method is based on work from Ding and Gentleman (2005), and it consists of adapting the classic

Iteratively Reweighted Partial Least Squares (IRWPLS) regression algorithm (Wold, 1975) to the M-IRWPLS-F, also denoted as the Generalized-PLS regression model for a classification setting.

The PCR method for predicting a categorical logit response associated with an observed curve is limited by two issues. This method fails to account for the response variable, and the original algorithm was designed for a continuous response variable with a linear relationship with the predictor. The proposed method is a strategy that integrates these concepts into functional generalized linear models by extending PLS, a popular dimension reduction tool, to the context of the F-NM-LR model, based on M-IRWPLS-F. Usually the criterion for constructing components in PLS is to sequentially maximize the covariance between the response and the predictor variable.

While the PLS approach applied to a nonlinear response may not be ideal, Marx (1996) presented a development of generalized linear models to accommodate regression of nonlinear responses on a set of covariates by considering the extension of PLS from the linear model to the G-PLS for classification settings. The present approach adapts this line of G-PLS to the case of a functional logit model and incorporates the G-PLS method into an F-NM-LR model as a natural generalization of the PLS regression method.

On the other hand, the estimation of this multi-category logit model is carried out by maximizing the multinomial log likelihood under model (3.5). Thus, the log likelihood is usually maximized using the Newton-Raphson algorithm, which in turn results in the iteratively re-weighted least squares (IRLS) method. Ding and

Gentleman (2004) studied this method specifically for classification problems and developed the multinomial regression version denoted as MIRWPLS. Ding and Gentleman (2004) also applied the Firth bias reduction procedure (Firth, 1992) denoted as MIRWPLS-F in order to avoid the common non-convergence and infinite parameter estimates problems of logistic regression in large number of predictor, small sample size problems.

The M-IRWPLS-F procedure carries out multi-group classification from a generalization of the method to predict multi-categorical response variables, which is based on the multinomial logit model. The M-IRWPLS-F method is reported to achieve better classification performance with a lower classification error rate. For a more detailed description of classification using generalized partial least squares, refer to Ding and Gentleman (2005). However, the proposed approach is simpler; its implementation requires minimal and easy programming, and it easily generalizes to all models that are linear at the level of explanatory variables.

To the best of the author's knowledge, the M-IRWPLS-F method, also known as (G-PLS), has never been applied directly to the classic of a functional logistic regression model. Therefore, this study adapts the classical G-PLS algorithm, resulting in a new algorithm for the G-PLS method, the generalized PLS method, denoted as (GPLS) for the F-NM-LR model. The greater accuracy of GPLS under the F-NM-LR model improves the lines of functional PLS classification approaches as multiple stages and in general performs comparably to other methods.

In this study, classes are always nominal with no special ordering, so it is easy to use any other model deemed appropriate, such as an adjacent logit model.

Therefore, this model generalizes well, and it also accommodates complex multi-group relationships that make classification procedures based on a functional generalized linear models framework so appealing.

The maximum likelihood estimation of the parameters for generalized linear models via Fisher's scoring method that was presented by McCullagh and Neider (1989) is adapted for the proposed approach. Also, by following Ding and Gentleman (2005), the PLS components are functions of the fitted values. Estimates of these components are obtained iteratively. Details of an algorithm to compute the functional GPLS components for the F-NM-LR model are given in the next section.

The Functional Generalized Partial Least Squares Nominal Multinomial Logit Regression (F-GPLS-NM-LR) Model

As an alternative to FPC-NM-LR model, the goal of this study is to propose a new functional regression model based on the GPLS method that combines a modified version of the functional PLS logistic regression algorithm presented by Escabias et al. (2007) and the classical PLS regression algorithm (Wold et al., 1983) to create the F-NM-LR model. The modification of the functional GPLS nominal multinomial logit regression (F-GPLS-NM-LR) model proposed in this work consists of applying this GPLS logit algorithm to the nominal multinomial response vector and the design matrix $\boldsymbol{H}$, where the nominal multinomial response vector $\boldsymbol{y}_i$ is changed by the corresponding logit model; meanwhile, the rest of the linear fits are kept.

As is the case for all PLS algorithms, the goal of GPLS method is to find a set of uncorrelated latent vectors that are linear spans of the functional predictor

$X(t)$ that maximize the covariance between explicative variables and the nominal multinomial response variable, respectively, and use them as predictors in the F-NM-LR model. This is done iteratively. The F-GPLS-NM-LR model proposed in this study is based on a multiple GPLS logit regression model that has $\boldsymbol{H}$ as a design matrix.

The algorithm for computing the F-GPLS-NM-LR model consists of the three following steps: (1) Computation of a set l of GPLS components, (2) Logit regression fitting of the nominal multinomial response matrix $\boldsymbol{Y}$ on the retained GPLS components and (3) Formulation of the F-GPLS-NM-LR model in terms of the original predictor variables.

1. **Computation of a set of GPLS components.** Let $\boldsymbol{Y}$ be a multinomial response matrix and $\boldsymbol{H}_j,\ j = 1, \ldots, p$ the columns of the design matrix of the F-NM-LR model. Then, the algorithm that computes the GPLS components is summarized as follows:

   Step 1: Extraction of first GPLS component which is denoted by, $\boldsymbol{T}_1$, obtained as follows:

   - Logit regress $\boldsymbol{Y}$ on each column of the design matrix $\boldsymbol{H}_j$. Let the estimated parameters be denoted by $\widehat{\boldsymbol{\Delta}}_1 = (\widehat{\boldsymbol{\delta}}_{11}, \ldots, \widehat{\boldsymbol{\delta}}_{1p})'$ and $\boldsymbol{V}_1 = (\boldsymbol{v}_{11}, \ldots, \boldsymbol{v}_{1p})'$ its normalized form where $\boldsymbol{v}_{1j} = \widehat{\boldsymbol{\delta}}_{1j}/\|\widehat{\boldsymbol{\delta}_{1j}}\|$. Thus, the results of interest are the regression coefficients associated with $\boldsymbol{H}_j$, denoted by $\boldsymbol{V}_1$.

- If the coefficient v_{1jk} is not significant, as determined by the classic Wald statistical test, then set v_{1jk} equal to zero. That means to delete those components of $\boldsymbol{V}_1$ that verify $|\widehat{\delta}_{1jk}/SE(\widehat{\delta}_{1jk})| \leq z_{\alpha/2}$, with $SE(\widehat{\delta}_{1jk})$ being the estimated standard deviation of $\widehat{\delta}_{1jk}$ and $z_{\alpha/2}$ being a fixed critical value of the standard normal distribution. This ensures that only those covariates that predict the response are used to build the GPLS component.

- Thus, the first GPLS component is defined as $\boldsymbol{T}_1 = \boldsymbol{V}_1\boldsymbol{H}$.

Step l: Given $\boldsymbol{T}_1, \ldots, \boldsymbol{T}_{l-1}$ the first $l-1$ GPLS components, the l^{th} one is obtained as follows:

- Logit regress $\boldsymbol{Y}$ on $\boldsymbol{T}_1, \ldots, \boldsymbol{T}_{l-1}$ and each $\boldsymbol{H}_j$ $(j = 1, \ldots, p)$. Let $\widehat{\boldsymbol{\Delta}}_l = (\widehat{\boldsymbol{\delta}}_{l1}, \ldots, \widehat{\boldsymbol{\delta}}_{lp})'$ and $\boldsymbol{V}_l = (\boldsymbol{v}_{l1}, \ldots, \boldsymbol{v}_{lp})'$ its normalized form where $\boldsymbol{v}_{lj} = \widehat{\boldsymbol{\delta}}_{lj}/\|\widehat{\boldsymbol{\delta}_{lj}}\|$. Then, set equal to zero those coefficients of v_{ljk} that are not significant, $|\widehat{\delta}_{ljk}/SE(\widehat{\delta}_{ljk})| \leq z_{\alpha/2}$, with $SE(\widehat{\delta}_{ljk})$ being the estimated standard deviation of $\widehat{\delta}_{ljk}$.

- Then, computation of the l^{th} GPLS component as the first principal component th of the data matrix $\boldsymbol{T}_h$

- After that, in order to find a GPLS component that is orthogonal to all previous GPLS components, linearly regress each $\boldsymbol{H}_j$, identified in step l, on $\boldsymbol{T}_1, \ldots, \boldsymbol{T}_{l-1}$, and the primary result of interest is denoted by $\boldsymbol{R}^{(l-1)}$= $(\boldsymbol{r}_1^{(l-1)}, \ldots, \boldsymbol{r}_p^{(l-1)})$ which is their $n \times p$ residual matrix.

- The l^{th} GPLS component is defined by $\boldsymbol{T}_l = \boldsymbol{V}_l\boldsymbol{R}^{(l-1)}$.

Note: The algorithm stops calculating GPLS components when none of the v_{ljk} are considered significantly different from zero. The other usual method for selecting the number m of GPLS components to be retained is cross-validation on the predictive power of the model. Tenenhaus (2002) presents a detailed study.

2. **Regressing the response variable on GPLS components.** Logit regression fitting of multinomial response matrix $\boldsymbol{Y}$ on the retained GPLS components. The first step is to express all GPLS components in terms of the original covariates (columns of the $\boldsymbol{A\Psi}$ matrix) instead of the corresponding residual vectors. Let $\boldsymbol{\Gamma}$ be the matrix of GPLS components of the design matrix $\boldsymbol{A\Psi}$, then $\boldsymbol{\Gamma} = \boldsymbol{A\Psi U}$, with $\boldsymbol{U}$ being the matrix whose columns are the vector of coefficients of the GPLS components in terms of the original predictors.

3. **Formulation of the GPLS logit regression model in terms of the original predictors.** The third step is the formulation of the GPLS logit regression model in terms of the original predictors, which is discussed in detail in the next section.

The Generalized Partial Least Squares Estimation Method for the Functional Nominal Multinomial Logit Regression Model

Following the principles of functional PLS logistic regression (Escabias et al., 2007), the proposed method uses a reduced set of GPLS of the sample curves as regressors for the F-NM-LR model. Then the F-NM-LR model (3.5) can be equivalently expressed in terms of all GPLS as

$$\widehat{\boldsymbol{L}} = \widehat{\boldsymbol{\alpha}} \otimes \mathbf{1} + \boldsymbol{\Gamma}\widehat{\boldsymbol{\gamma}}, \tag{3.6}$$

where $\widehat{\boldsymbol{\gamma}} = (\widehat{\boldsymbol{\gamma}}_1, \ldots, \widehat{\boldsymbol{\gamma}}_{S-1})'$, is a $(S-1) \times p$ matrix of the maximum likelihood estimators of the coefficients of the logit model. So the Maximum Likelihood (ML) estimation of the basis coefficients of the functional parameters can obtained through the estimation of the parameter of equation (3.6) by

$$\widehat{\boldsymbol{B}} = \boldsymbol{U}\widehat{\boldsymbol{\gamma}}$$

However, the ML estimation of the functional parameter provided by using all GPLS components as predictor variables is very rough and inaccurate, which has been shown by Escabias et al. (2014). Consequently, the proposed method adapts Escabias et al. (2014) method by using as predictors an optimum set of m GPLS components contained in the columns of the following matrix:

$$\boldsymbol{\Gamma}^{(m)} = (\boldsymbol{A\Psi})\boldsymbol{U}^{(m)}.$$

where m is chosen by cross validation. Then, the F-GPLS-NM-LR model is given by

$$\widehat{\boldsymbol{L}}^{(m)} = \widehat{\boldsymbol{\alpha}}^{(m)} \otimes \mathbf{1} + \boldsymbol{\Gamma}^{(m)}\widehat{\boldsymbol{\gamma}}^{(m)}, \tag{3.7}$$

finally the ML estimation of the functional parameters in terms of the m GPLS components is given by

$$\widehat{\beta}_s^{(m)}(t) = \Phi'(t)\ \widehat{\boldsymbol{B}}^{(m)},$$

with $\widehat{\boldsymbol{B}}^{(m)} = \boldsymbol{U}^{(m)}\widehat{\boldsymbol{\gamma}}^{(m)}$.

The next section includes a brief summary of the functional PCA method to provide an understanding of how this process compares to GPLS approach.

Overview of Functional Principal Components Regression Based Solution

The basic idea behind the PCR-based solution of the functional PCA and base-line logit models is to calculate the principal components and then use a reduced set of components as predictors in a linear regression model, which is fitted using the typical least squares procedure. The FPC method generates uncorrelated linear combinations of the functional predictor with maximum variance (Escabias et al., 2014). The theoretical results are shown in Ocaña, Aguilera, and Escabias (2007). A more detailed review of the FPC method as a classic solution to dimensionality and multicollinearity problems in FDA is presented in Chapter II.

Methods of selecting the optimum number of FPCs to use and the order in which they must be included in the model are discussed next. These methods take into account both explained variability and the ability to predict the response while providing the best estimate of the functional parameters. However, Escabias et al. (2014) demonstrated that, in the case of FPC, the most efficient way to select PCs in the model is to choose principal components using either way; a forward stepwise method based on a conditional likelihood ratio test, which takes into account the relationships of components with the response variable or the natural order of explained variability in order to maximize accuracy.

Remark. Aguilera et. al. (2010) demonstrated that Multivariate PCA of $\boldsymbol{A\Psi}$ matrix with respect to the usual metric in $\mathbb{R}^p$ is equivalent to FPCA of the transformed sample curves basis coefficients with respect to the usual metric in L^2.

The Generalized Partial Least Squares Component Selection

Two methods for selecting the optimal number of GPLS to be retained are compared below, the response prediction type and the functional parameter type, using Escabias et al. (2014) methodology for FPCA and base-line logit models.

Criterion 1. The Correct Classification Rate (CCR) is defined as the rate of agreement between the observed and predicted response category. The predicted category of an individual is the one associated with the highest predicted probability.

Criterion 2. The second criterion consists of selecting the number of components that provides the most accurate estimation of the functional parameters. Thus, minimizing the Integrated Mean Square Error (IMSE) of the parameter functions can be used to evaluate the accuracy of F-GPLS-NM-LR model, which can be computed only for simulations where $\beta_s(t)$ is known. The IMSE of the parameter functions is defined as

$$IMSE(m) = \frac{1}{S-1}\sum_{s=1}^{S-1}\frac{1}{T}\int_T \left(\beta_s(t) - \widehat{\beta}_s^{(m)}(t)\right)^2 dt. \tag{3.8}$$

Criterion 3. Minimizing the leave-one-out Cross-Validation Mean Squared Error (CVMSE) criteria was also used in this study to find an accurate estimate of the functional parameters in real dataset example, where it is impossible to calculate the IMSE. Therefore, taking into account the response, the number of components that optimizes the F-GPLS-NM-LR model by minimizing the CVMSE of prediction, is defined as

$$CVMSE(m) = \frac{1}{S}\frac{1}{n}\sum_{s=1}^{S}\sum_{i=1}^{n}\left(y_{is} - \widehat{\pi}_{(-i)s}^{(m)}\right)^2, \tag{3.9}$$

where $\widehat{\pi}_{(-i)s}$ is the probability of the s category provided by the model with m GPLS components, predicted for the i^{th} individual by using the model fitted after removing the i^{th} individual from the data.

In conclusion, the different types of criteria that can help us to detect the optimum number of GPLS components, the response-prediction-type and the functional parameters type, are used for this study. In the case of the response-prediction-type criterion, the CCR was considered. On the other hand, for the functional parameter type criterion, the optimum number of components is that which minimizes the IMSE or CVMSE was considered because it provides the best estimation of the functional parameters.

For this study, simulation and real datasets were developed to allow the proposed GPLS method to be used to estimate parameter functions and discriminate a set of curves for the F-GPLS-NM-LR model more precisely.

Simulation Study Techniques

To test the effectiveness of the GPLS method, a simulation study was conducted following the scheme proposed in Escabias et al. (2014) with some modification. The following explains the simulation procedure:

(1) Generate the functional predictor as $X(t) = Z(t) + at + bE$, where $a = \frac{1}{2}$, $\frac{1}{4}$, and 1, and $b = 1, 5$, and 10, are arbitrary values that have been chosen by the researcher. $Z(t)$ is a centered Gaussian process with covariance function $C(s,t) = 0.5^{80|t-s|}$. E is a Bernoulli random variable with probability (0.1), and the considered functional variable is in the domain of T = [0, 12], which is adopted from Escabias et al. (2014).

(2) Meanwhile, it is impossible to record the functional form of the curves of a functional variable with only discrete observations at different points of the domain interval. As such, to obtain a sample of curves for this study, an $n = 80$ sample of 15-dimensional curves were simulated using the defined process at a set of 9 unequally spaced knots $\{0, 1.1, 2.5, 3.7, 5.1, 7.3, 8.5, 9.6, 12\}$ on the interval [0, 12]. Also, natural cubic spline interpolation is used to reconstruct the true functional forms of the curves.

(3) The number of categories (e.g., $2, 3$, and 4) of the nominal response variable are changed during the process, as displayed in Table 2, and the response probabilities are simulated in terms of cubic B-spline expansion of sample curves and parameter functions using the F-NM-LR model. The observations of the response are randomly simulated from a multinomial distribution with the simulated probabilities as parameters.

(4) From Escabias et al. (2014), the intercept parameter is $(\alpha_1, \alpha_2, \alpha_3) =$ (0.30, 0.19, 0.20), and (5) the functional parameters are represented by natural cubic spline interpolation of the sinusoidal functions

$$\beta_1(t) = \cos(t - \frac{\pi}{4}),$$

$$\beta_2(t) = \sin(t - \frac{\pi}{4}),$$

$$\beta_3(t) = \cos(t - \frac{\pi}{4}) - \sin(t - \frac{\pi}{4}).$$

(6) After data simulation, the nominal response functional logit model (3.5) is fitted. A total of 9 schemes were simulated using the process described in Table 2.

(7) Next, the problem of multicollinearity is solved using the proposed method (GPLS), and then this method was compared in terms of its ability to improve the estimation of the functional parameters. Two criteria are used to select optimal models (GPLS components). Moreover, the number of GPLS components are chosen by minimizing the IMSE as the form of the estimated functional parameters and the accuracy measure.

Another important aspect of the simulation is the opportunity to compare the classification ability of the F-GPLS-NM-LR model with the alternative method, the FPC-NM-LR model proposed by Escabias et al. (2014). In order to draw conclusions about the relative performance of the estimation approaches, 500 simulation replications are run.

Table 2
Schemes of the Parameters

Sample	Function Predictor	Functional	Number of
Curve		Parameters	Categories Response
80	$X(t) = Z(t) + a_i t + b_i E, \quad i = 1, 2, 3,$		
	$a_1 = \frac{1}{4} \quad b_1 = 0.5,$	$\beta_1(t) = \cos\left(t - \frac{\pi}{4}\right)$	2
	$a_2 = \frac{1}{2} \quad b_2 = 5,$	$\beta_1(t) = \cos\left(t - \frac{\pi}{4}\right)$	3
	$a_3 = 2 \quad b_3 = 10,$	$\beta_2(t) = \sin\left(t - \frac{\pi}{4}\right)$	
		$\beta_1(t) = \cos\left(t - \frac{\pi}{4}\right)$	4
		$\beta_2(t) = \sin\left(t - \frac{\pi}{4}\right)$	
		$\beta_3(t) = \cos\left(t - \frac{\pi}{4}\right) - \sin\left(t - \frac{\pi}{4}\right)$	
The Intercept Parameter	$(\alpha_1, \alpha_2, \alpha_3) = (0.30, 0.19, 0.20).$		
Knots	$t = \{0, 1.1, 2.5, 3.7, 5.1, 7.3, 8.5, 9.6, 12\}.$		
	$\pi = 3.14159$ is a mathematical constant.		
The Response Probabilities	$\pi_{is} = \frac{\exp\left\{\alpha_s + \int_T x_i(t)\beta_s(t)dt\right\}}{\sum_{s=1}^{S} \exp\left\{\alpha_s + \int_T x_i(t)\beta_s(t)dt\right\}},$		
	$s = 1, \cdots, S, \ i = 1, \cdots, n,$		
	with $\alpha_S = 0, \ \beta_S(t)=0.$		

In addition to the simulation, a real world dataset, spectrometric data consisting of curves of spectrometry (absorbance measured in terms of wavelength) of corn, is used to demonstrate the estimation technique in terms of reducing the number of components required to get a parsimonious model. In spectroscopy the most common problem is calibration that consists of estimating a scalar response variable from the spectrum. Despite the functional nature of spectral data, this problem is usually analyzed with multivariate statistical methods such as PCR and PLS regression that consider the spectrum as a vector associated with its measures at a finite number of wavelengths. Taking into account that the absorbance at two nearby wavelengths is highly correlated, the proposed process could provide better estimations via considering the spectrum as a curve instead of a vector. The potential use of functional data analysis in spectroscopy and chemometric data was stated by Saeys, De Ketelaere, and Dairus (2008).

Chemometricians are also interested in the classification of chemometric data (curves of spectrum) according to a characteristic of interest of the substance that generated the curve (Ferraty and Vieu, 2003). In this study an F-GPLS-NM-LR model is used to classify near infrared reflectance (NIR) spectra of corn samples according to the spectrometer that generates them. This is a retrospective study of curves of the spectrum. The NIR spectra of 80 corn samples were measured by three different instruments at Cargill Inc. (m5, mp5, and mp6 spectrometers). The wavelength domain was [1100, 2498] nm, measured at 2 nm intervals (700 observations).

The F-GPLS-NM-LR model was developed using R-3.3.2. All resulting output is assembled and presented in both tables and figures, which are presented and discussed in Chapter IV.

CHAPTER IV

RESULTS

This chapter describes the results of both a simulation study and a real dataset application to illustrate the proposed GPLS methodology developed in chapter III. The proposed method is built on basis expansions of the sample curves of the functional predictor and parameters. A classification problem with a functional predictor was studied to demonstrate the GPLS approach to estimation of the functional parameter and curve classification for the F-NM-LR model. This description is followed by a comparison in terms of performance of the GPLS method to the classic PCR method for parameter estimation and classification in the F-NM-LR model, providing an illustration of how the F-GPLS-NM-LR model behaves under different circumstances. Goodness of fit and accuracy measures were calculated to test the classification ability of the two methods considered to determine whether one or the other model provides better performance in terms of CCR. The IMSE was used to select the optimum number of components (GPLS or PCRs) that provided the most accurate estimation of the functional parameters. To facilitate comparisons of the two estimation methods (F-GPLS-NM-LR and FPC-NM-LR), it is important to clarify that all results compare the means and standard deviations of the number of covariates, IMSE, and CCR of the different optimum logit models after 500 simulation replications.

The simulation study was carried out in R version (3.3.2); most of the program was written by the author. To evaluate and compare the aforementioned approaches accurately, this study followed the simulation scheme developed by Escabias et al. (2014), providing a comparable replication of the data analyses used in previous research. The simulation results are reported, and they are presented in tables and figures relative to each research question.

The remainder of this chapter is divided into the following sections. The first section describes the steps in the simulation study and how each estimation method was implemented with respect to the F-NM-LR model. The second section, presents the simulation study results for research Question 1. The third section contains the simulation study results comparing the behavior of the proposed model under different conditions to address the second and third research questions. The fourth section presents simulation study results comparing the performance of the F-GPLS-NM-LR and the FPC-NM-LR model proposed in Escabias et al. (2014) in terms of their performance in estimation and prediction (research Question 4). The fifth section presents results of the performance of the F-GPLS-NM-LR model on a real data set (research Question 5). Lastly, the simulation and the real dataset results are summarized.

Simulation Study Algorithm

The algorithm for the simulation study consisted of generating the functional covariates, the response probabilities, the intercept parameters, the parameter functions, then, the observations of the response. The steps for the algorithm are as follows:

Simulation algorithm. The F-NM-LR model considered in this work was defined as

$$l_{is} = \alpha_s + \int_T x_i(t)\beta_s(t)dt, \quad s = 1, 2, \ldots, S-1. \tag{4.1}$$

Step 1: Generate a sample of curves, sized $n = 80$, of a second order stochastic process of a functional predictor $\{X(t) : t \in T\}$ whose sample curve is

$$X(t) = Z(t) + at + bE,$$

where $Z(t)$ is a centered Gaussian process that has covariance function $C(s,t) = 0.580|t - s|$, and E is a Bernoulli random variable with probability (0.1). More specifically, for the first step, a set of unequally spaced knots $\{0, 1.1, 2.5, 3.7, 5.1, 7.3, 8.5, 9.6, 12\}$ on the interval $T = [0, 12]$ was obtained. The values a and b are set at $a = .25$ and $b = 5$ for the development of the model and for the comparison of the model with the PCR method (Questions 1 and 4). These values are manipulated to explore research Questions 2 and 3. This method of generating the functional covariates was chosen based on Escabias et al. (2014) research to allow direct comparisons to an existing estimation method, the FPC-NM-LR model, later on.

Step 2: A natural cubic spline interpolation was used to reconstruct the true functional forms of each sample of curves.

Step 3: This simulation study involved a nominal random variable with four categories as the response variable to explore research Questions 1 and 4. Also, different category response levels were considered to classify a set of curves into four, three, or two groups defined by a a nominal random variable to explore research Questions 2 and 3.

Step 4: The response probabilities given in model (3.3) were simulated in terms of cubic B-spline expansion of the sample curves and parameter functions corresponding to 15 basis functions over the interval $[0, 12]$.

Step 5: The intercept parameters were $(\alpha_1, \alpha_2, \alpha_3) = (0.30, 0.19, 0.20)$, and the functional parameters were also represented by natural cubic spline interpolation at these nodes $\{0, 1.1, 2.5, 3.7, 5.1, 7.3, 8.5, 9.6, 12\}$ of the sinusoidal functions, following the methodology of Escabias et al. (2014).

$$\beta_1(t) = \cos\left(t - \frac{\pi}{4}\right)$$

$$\beta_2(t) = \sin\left(t - \frac{\pi}{4}\right)$$

$$\beta_3(t) = \cos\left(t - \frac{\pi}{4}\right) - \sin\left(t - \frac{\pi}{4}\right).$$

Step 6: The observations of the response were randomly simulated from a multinomial distribution with the simulated probabilities as parameters. The probabilities of the nominal multinomial response are modeled in terms of the functional predictor and the parameters as

$$\pi_{is} = \frac{\exp\{\alpha_s + \int_T x_i(t)\beta_s(t)dt\}}{\Sigma_s \exp\{\alpha_s + \int_T x_i(t)\beta_s(t)dt\}}, \tag{4.2}$$

for $s = 1, 2, \ldots, S$, $i = 1, 2, \ldots, n$, with $\alpha_S = 0$, $\beta_S(t) = 0$.

Step 7: Fit the F-NM-LR model

$$\boldsymbol{L} = \boldsymbol{\alpha} \otimes \mathbf{1} + \boldsymbol{HB}, \qquad \mathrm{s} = 1, \ldots, \mathrm{S} - 1. \tag{4.3}$$

Step 8: In order to improve the estimation of the functional parameters of the F-NM-LR model, the GPLS and PCR methods were fitted using:

- The algorithm for computing the FPC-NM-LR model described in Chapter II.
- The algorithm for computing the F-GPLS-NM-LR model described in Chapter III.

However, the estimated parameter functions are not comparable because they correspond to different regression models from a theoretical point of view. The two methods are compared in terms of plots and the selected criteria (number of covariates, IMSE, and CCR).

Step 9: In order to draw conclusions about the performance of the GPLS-based estimation approach, the simulation of a nominal response variable and following computation of GPLS to fit the F-NM-LR model was repeated 500 times.

Step 10: For each repetition goodness of fit and accuracy measures (number of covariates, and CCR) were recorded to test the performance of each model. Also, leave-one-out cross-validation was completed to select the number of GPLS components that provided the most accurate estimates of the functional parameters by minimizing the IMSE.

Step 11: To compare the degree of dimension reduction produced by the two models' selection criteria, box plots for the distribution of the selected number of PLS and PCR components were drawn.

After that, steps 1 through 7 generate simulated data to fit the F-NM-LR model (4.3) with no modifications; steps 8 through 11 are those that are used for further generation of results based on GPLS and PCR methods. Table 3 includes results indicating the precision of the F-NM-LR model. In terms of goodness-of-fit statistics, the correct classification rate (CCR) suggested good prediction ability for this model (CCR=87.5), but the estimated functional parameter plots and the accuracy measure (IMSE=3.94E+4) showed poor estimations of the functional parameters.

Table 3
Goodness of fit and accuracy measures of the F-NM-LR model

Model	Covariates	IMSE	CCR
F-NM-LR	15	3.94E+4	87.5

Figure 7 shows the estimates of the parameter functions, and Figure 8 illustrates the distribution of the correlations between the columns of the design matrix $\boldsymbol{H}$. The inaccuracy of these estimates makes it very difficult to interpret the relationship between the functional predictor and the nominal multinomial response variable. Figures 7 and 8 clearly illustrate the issues encountered with the F-NM-LR model in FDA. The model fails to provide a good fit for any of the simulated parameter functions, and the box plot highlights the multicollinearity problem. The GPLS approach demonstrated in next section is intended to improve the estimates of the parameter functions of the F-NM-LR model by avoiding the multicollinearity problem.

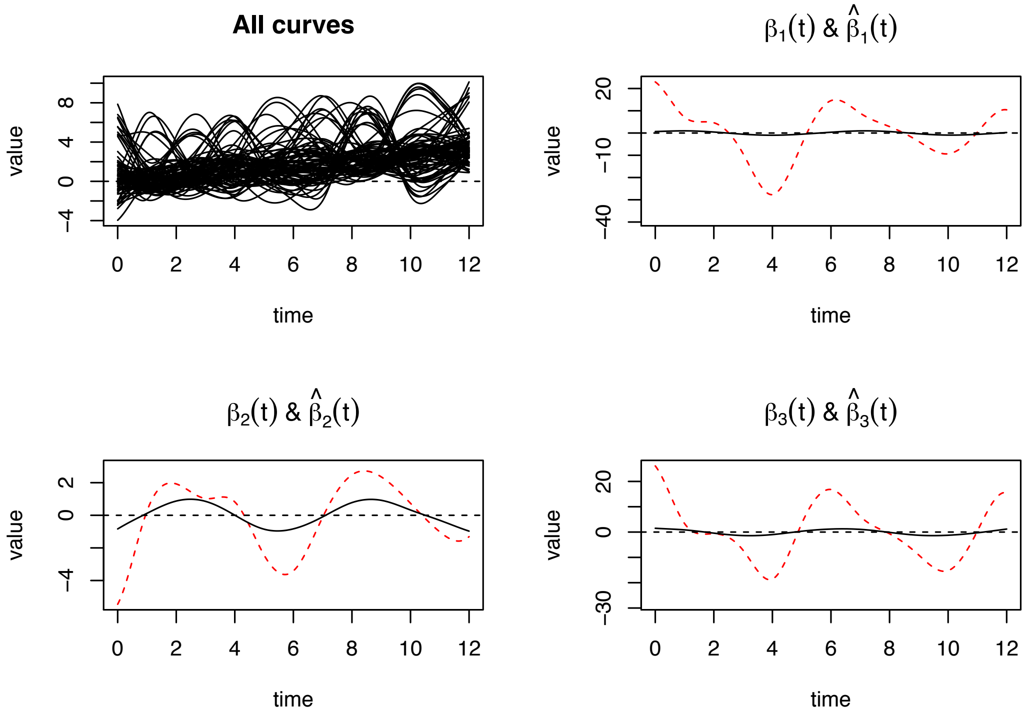


Figure 7. Simulated curves of the functional predictor variable (top left), $y = 0$ (black dashed line), simulated parameter function (black solid line), and its estimation in terms of cubic B-spline smoothing without using GPLS (red dashed line) for the F-NM-LR model.

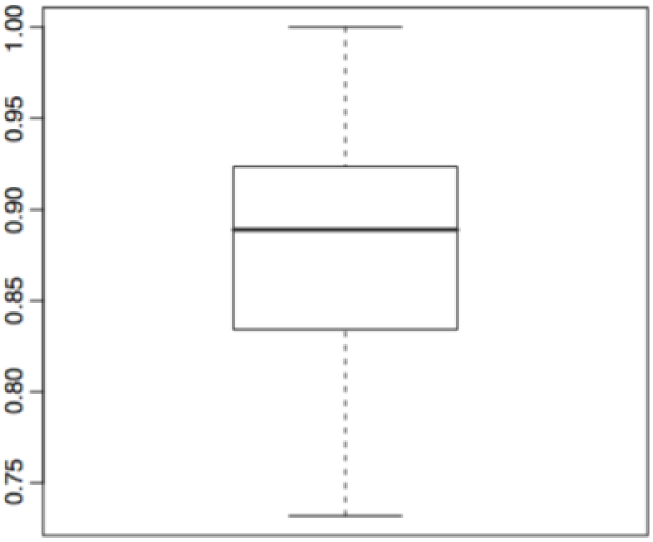


Figure 8. Box plot of the distribution of correlations between columns of the design matrix $\boldsymbol{H}$.

The Development of Generalized Partial Least Square Method for the Functional Nominal Multinomial Logit Regression Model

This section is focused on addressing Question 1 by demonstrating the development of the GPLS method using a functional predictor to classify a four-category response variable:

Q1 How is a generalized partial least squares regression method developed for parameter function estimation in the functional nominal multinomial logit regression model?

In order to solve the problem of multicollinearity in F-NM-LR models, the process followed the same steps 1 through 7 from the above section, but in step 8, the proposed GPLS algorithm was applied to fit the F-NM-LR model in simulated data. As was demonstrated in Chapter III, the algorithm for obtaining GPLS estimates as covariate functions for computing the F-GPLS-NM-LR model consists of the following three steps:

1. Computation of a set the l^{th} GPLS orthogonal components called $\boldsymbol{T}_l$.

2. Logit regression fitting of multinomial response matrix $\boldsymbol{Y}$ on the l^{th} retained GPLS components, the l^{th} GPLS component must capture the discriminant information not available in the $l-1$ previous ones.

3. Computation of the l^{th} GPLS component as the first principal component $\boldsymbol{T}_l$ of the design matrix.

4. Formulation of the F-GPLS-NM-LR model in terms of the original predictor variables, where $\boldsymbol{\Gamma} = \boldsymbol{HU}$, with $\boldsymbol{U}$ being the matrix whose columns are the vector of coefficients of the GPLS components in terms of the original predictors.

5. Then, the F-NM-LR model (4.3) is expressed in terms of the GPLS components as

$$\widehat{\boldsymbol{L}} = \widehat{\boldsymbol{\alpha}} \otimes \mathbf{1} + \boldsymbol{\Gamma}\widehat{\boldsymbol{\gamma}}, \tag{4.4}$$

The ML estimation of the functional parameter in the current study was obtained by using an optimum set of m GPLS contained in the columns of the design matrix as predictors. This process yields the F-GPLS-NM-LR model as follows:

$$\widehat{\boldsymbol{L}}^{(m)} = \widehat{\boldsymbol{\alpha}}^{(m)} \otimes \mathbf{1} + \boldsymbol{\Gamma}^{(m)}\widehat{\boldsymbol{\gamma}}^{(m)}, \tag{4.5}$$

where is m chosen by cross validation.

Finally, model (4.5) provides ML estimation of the functional parameter given by:

$$\hat{\beta}_s^{(m)}(t) = \Phi'(t)\hat{\boldsymbol{B}}^{(m)}$$

with $\hat{\boldsymbol{B}}^{(m)} = \boldsymbol{U}^{(m)}\hat{\boldsymbol{\gamma}}^{(m)}$.

In order to draw conclusions about the performance of the GPLS method, this simulation was repeated 500 times. In each repetition, the optimum number of components for the F-GPLS-NM-LR model was indicated by the lowest IMSE and the highest CCR after fitting the model. Finally, the means and standard deviations of the goodness-of-fit and accuracy measures previously defined for the optimum models were obtained. The simulation results for the F-GPLS-NM-LR model are presented in Table 4, and Figure 9 shows box plots for the distributions of IMSE and CCR for the model after data simulation.

Table 4
Sample means and standard deviations of the distributions of number of GPLS components, IMSE, and CCR of the optimum F-GPLS-NM-LR models

Measures	F-GPLS-NM-LR	
	Mean	SD
Covariates	2.26	0.90
IMSE	0.26	0.08
CCR	75.81	5.86

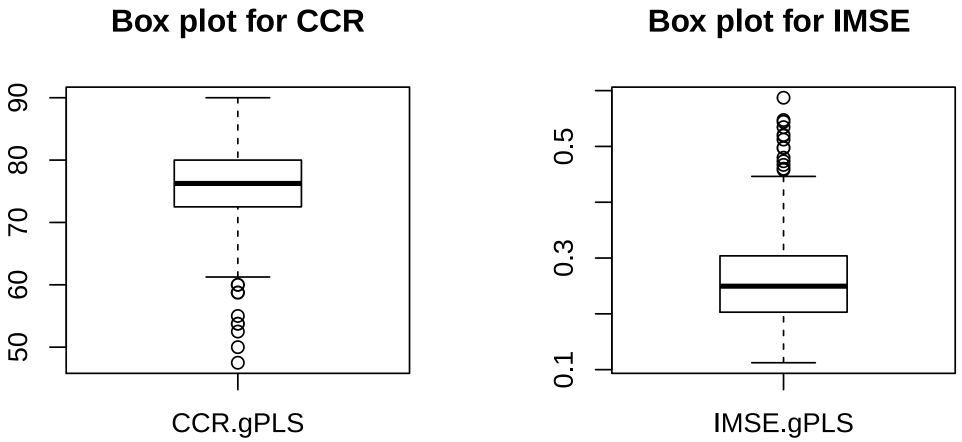


Figure 9. Box plots for the distributions of IMSE and CCR for the optimum F-GPLS-NM-LR model using the GPLS method.

The results shown in Table 4 indicate that the mean number of components needed for the best possible estimation of the parameter function is 2.26, selected by minimizing IMSE. Therefore, the mean accuracy of the estimated parameter functions is IMSE = 0.26. The mean value calculated for the CCR was 75.81, indicating an acceptable degree of accuracy. These results show that the GPLS approach produced a good fit for the model, and all of these measures showed that the estimations improved after using the proposed GPLS method. These results support the ability of the developed GPLS method to provide accurate parameter function estimates and good prediction ability.

Overall, the F-GPLS-NM-LR model performed well in terms of the selected criteria (number of covariates, IMSE, and CCR). To further illustrate the improvement in estimation provided by the GPLS method, the goodness of fit and accuracy measures previously established were used to compare the results and to evaluate each model (F-GPLS-NM-LR verses F-NM-LR) as can be seen in Table 5 and Figure 10.

Table 5
Goodness of fit and accuracy measures of the F-GPLS-NM-LR and F-NM-LR models

Measures	F-GPLS-NM-LR	F-NM-LR
Covariates	2.26	15
CCR	75.81	87.5
IMSE	0.26	3.94E+4

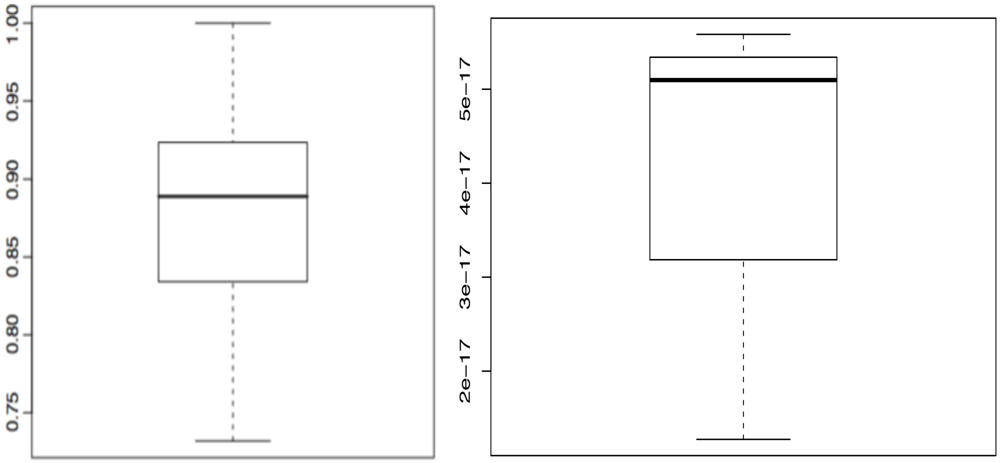


Figure 10. Box plot of the distribution of correlations between columns of the design matrix $\boldsymbol{H}$ verses $\boldsymbol{\Gamma}$.

To compare the models in more detail, box plots for the distribution of correlations between columns of the design matrix $\boldsymbol{\Gamma}$ are displayed in Figure 11. This figure highlights that the GPLS approach is a good choice for use with the F-NM-LR model as a solution for the multicollinearity problem. These results support the ability of the developed GPLS method to provide accurate parameter function estimates as can see in Figure 12. This figure illustrates the improvement in parameter function estimates obtained with the GPLS method with respect to the F-NM-LR model. To determine if the proposed method could be used as general rule, simulation results for this estimation method under varying circumstances are presented in the following section in response to Questions 2 and 3.

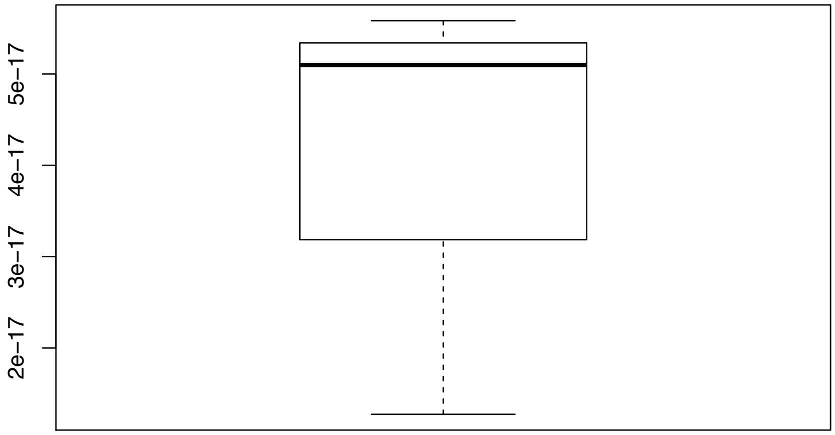


Figure 11. Box plot of the distribution of correlations between columns of the design matrix $\mathbf{\Gamma}$.

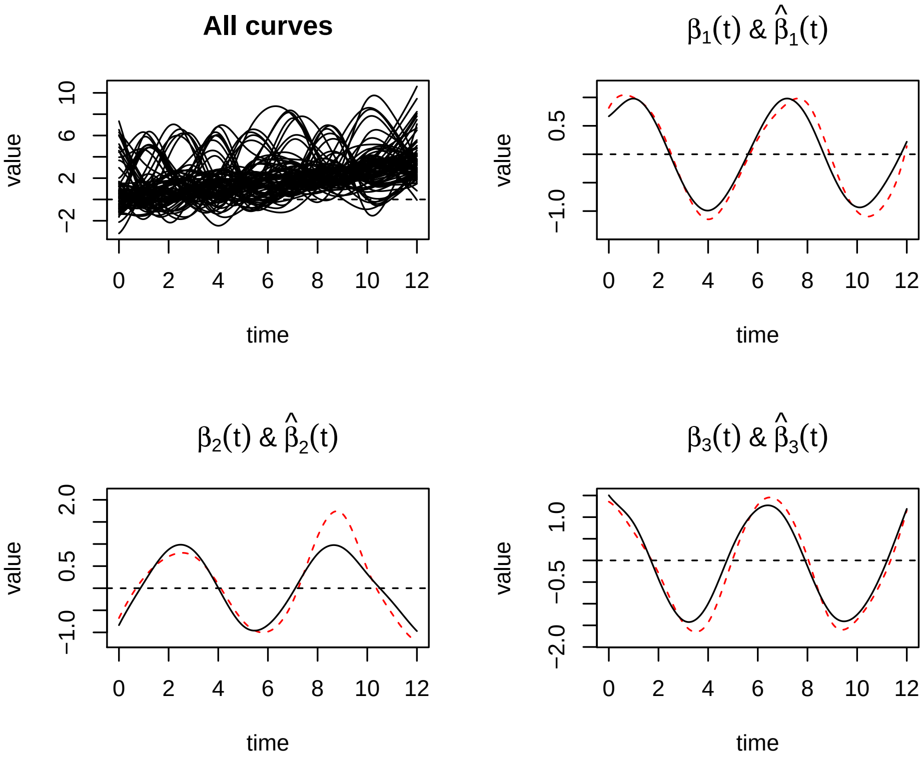


Figure 12. Simulated curves of the functional predictor variable (top left), $y = 0$ (black dashed line), simulated parameter function (black solid line), and its estimation in terms of cubic B-spline expansion and GPLS (red dashed line) for the F-NM-LR model.

Behavior of the Functional Generalized Partial Least Square Nominal Multinomial Logit Regression Model in Three Cases

To investigate the performance of the F-GPLS-NM-LR model more extensively, simulation studies were used to examine the behavior of the model under varying circumstances. These simulations provided important information about the consistency of the advantages provided by the model. This section focuses on the following two research questions:

Q2 How does the functional generalized partial least squares nominal multinomial logit regression model behave in terms of goodness-of-fit measures, such as the correct classification rate and the integrated mean squared error, with changes of the functional predictor dependent on arbitrary values assigned to a and b?

Q3 How does the functional generalized partial least squares nominal multinomial logit regression model behave in terms of goodness-of-fit measures, such as the correct classification rate and the integrated mean squared error, based on changes in the number of the nominal response categories?

To address Q2 and Q3, the behavior of the F-GPLS-NM-LR model was tested and results compared in a simulation study where different category response levels were considered to classify a set of curves into four, three, or two groups defined by a nominal response, with different values of a and b used to generate the functional predictor. To create three different case scenarios (Case 1, Case 2, and Case 3), the researcher arbitrarily chose the specific values for $a = 0.25, 0.5, 1.0$ and $b = 5, 1, 10$ as three cases. In order to draw conclusions about the behavior of the proposed model and the argument for using the GPLS approach as a general rule, the simulation was repeated 500 times for each case.

The criteria previously described in step 10 were applied in each repetition to optimize the F-GPLS-NM-LR model (i.e., the minimum number of covariates according to the lowest IMSE after fitting the model). Finally, the means and standard deviations over the 500 repetitions were obtained for the goodness-of-fit and accuracy measures previously defined for each model. Table 6 shows the results that illustrate the behavior of each model corresponding to the number of response categories 4, 3, or 2, respectively.

Table 6

Sample means and standard deviations for the distributions of the three criteria considered for the optimum F-GPLS-NM-LR models, for three cases and varied levels of response categories

Sample size	Number of Categories	Functional Parameter	Criterion Measures	Case 1 (a= 0.25, b = 5)		Case 2 (a= 0.5, b = 1)		Case 3 (a= 1.0, b = 10)	
				Mean	SD	Mean	SD	Mean	SD
80	4	β_1, α_1	Covariates	2.26	0.90	1.75	0.81	1.79	0.81
		β_2, α_2	IMSE	0.26	0.08	0.98	3.60	0.54	0.75
		β_3, α_3	CCR	75.81	5.86	78.54	5.59	83.45	5.19
80	3	β_1, α_1	Covariates	2.13	0.82	2.43	1.05	1.67	0.91
		β_2, α_2	IMSE	0.39	0.12	0.32	0.21	0.38	0.05
			CCR	72.68	7.65	78.70	5.51	85.34	3.87
80	2	β_1, α_1	Covariates	1.22	0.64	1.36	0.80	1.61	0.67
			IMSE	0.04	0.01	0.08	0.06	0.08	0.01
			CCR	88.06	2.74	91.39	2.45	93.86	2.10

The results shown in Table 6 indicate that in **Case 1**, $a = 0.25$, $b = 5$, at the different levels of response (4, 3, and 2 categories), respectively, the outcomes are very similar, and the GPLS method provides a great fit for these models. The mean number of components needed for the best possible estimation of the parameter function was 2.26, 2.13, and 1.22, respectively, selected by minimizing IMSE among the three models. The degree of dimension reduction and the mean accuracy of the estimated parameter functions (IMSEs) produced in the models with 4 and 3 response categories are similar, but the binary model needed fewer components and had the lowest IMSE, as would be expected. The results for CCR are also close for the models with 4 and 3 response categories and higher with binary model, and all models show an acceptable degree of accuracy.

Overall, the values generated in Case 1 corresponding to 4 response categories compared relatively well to the other two models on all criteria. Box plots for the distributions of the IMSE and CCR for the model with 3 response categories are displayed in Figure 13. It can be concluded from the results in Case 1, where $a = 0.25$, and $b = 5$, that the GPLS approach is a good option for predicting the categorical response and estimating the parameter function at all three response category levels. These results indicate that the observed strong performance of the GPLS method in the first simulation with 4, 3, and 2 response categories is likely to be the rule, meaning it is not particular to a specific case or number of categories in the response variable.

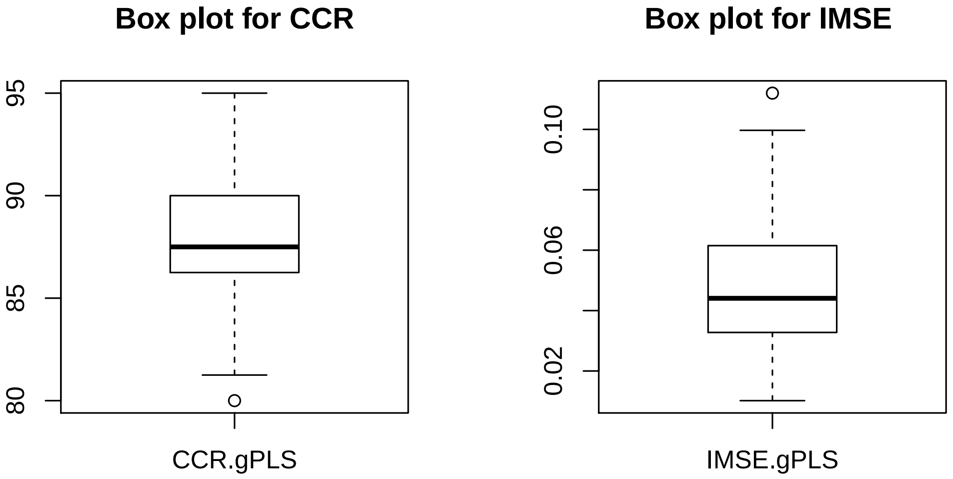


Figure 13. Box plots for the distributions of IMSE and CCR for the optimum F-GPLS-NM-LR model with 2 response categories in Case 1.

The optimum estimations of functional parameters in terms of different numbers of nominal response categories for the three cases after using the GPLS method are displayed in Figure 14. This figure shows that the parameter function estimates obtained using the F-GPLS-NM-LR model in the three different cases are very similar. All models provided accurate parameter function estimates. There are no major differences related to the number of response categories in Case 1 relative to their forecasting ability or in the accuracy of functional parameter estimates. The proposed GPLS method with respect to all models for Case 1 behaves as well as other classification methods in the F-NM-LR model. However, the objective of the proposed GPLS method is to provide an improvement over other methods for the estimation of the functional parameter associated with the F-NM-LR model for categorical responses.

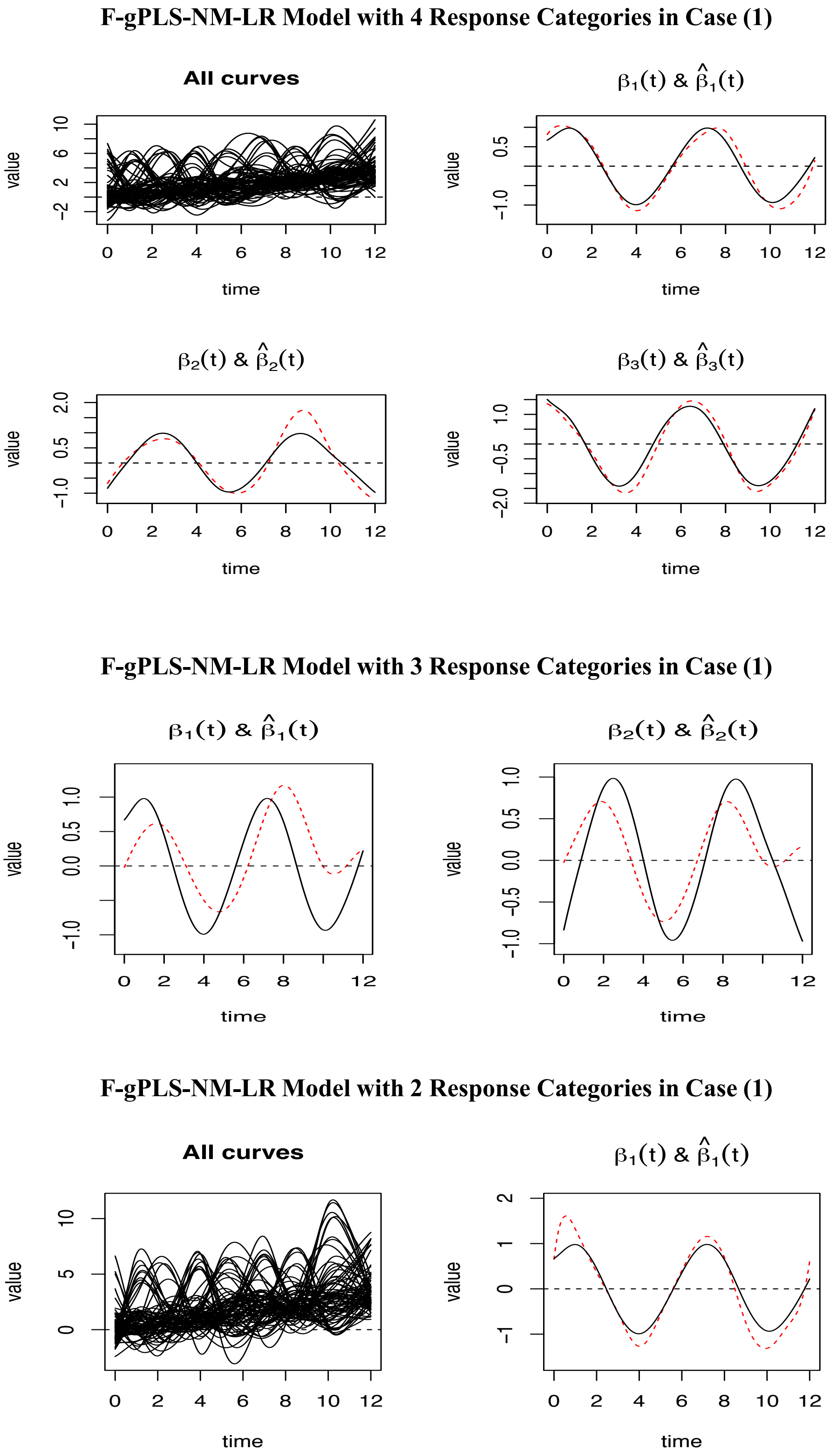


Figure 14. Simulated curves of the functional predictor variable (solid line) and its estimation for the optimum F-GPLS-NM-LR models (broken line) with 4, 3, and 2 response categories in Case 1.

As can be seen in Table 6, the results for **Case 2**, $a = 0.5$, $b = 1.0$, indicate that the F-GPLS-NM-LR models again produced good fit. The mean number of components needed for the best parameter function estimation was $1.75, 2.43$, and 1.36 for the three models, respectively. Thus, the degree of dimension reduction produced by the three models in Case 2 was close to that in Case 1. Also, the mean of the accuracy of the estimated parameter functions for all three models shows that they have no major differences in prediction ability, IMSEs were 0.98, 0.32, and 0.08, respectively. However, in the model with 4 response categories, the standard deviation is notably larger than in the models with 3 and 2 response categories. This result follows logically in that if values are set at $a = 0.5$ and $b = 1$, it is very likely that data for the response categories 2, 3, and 4 are generated while category 1 is missed, changing the operation to a three-category classification problem. The solution in this case was to delete the simulation repetitions that resulted in the wrong number of categories, so after 500 repetitions, only 440 of them were useful. Overall, similar to Case 1, the results for Case 2 in terms of CCR (78.54, 78.70, and 91.39) are close for the models with 4 and 3 response categories and higher with the binary model. Box plots for the distributions of IMSE and CCR in Case 2 for the model with 4 response categories are displayed in Figure 15. Overall, the simulation results for Case 2 corresponding to 3 categories compared relatively well to the binary model on all criteria.

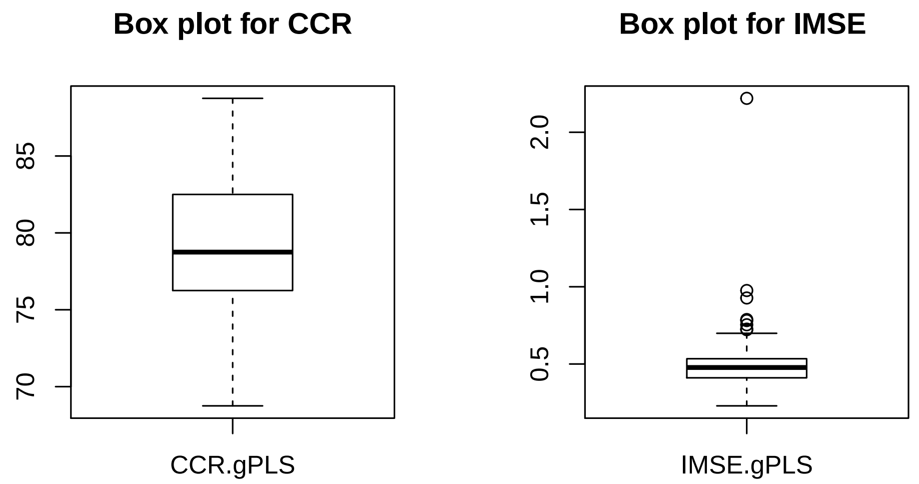


Figure 15. Box plots for the distributions of IMSE and CCR for the optimum F-GPLS-NM-LR model with 4 response categories in Case 2.

Table 6 shows that in **Case 3**, $a = 1.0$, $b = 10$, for all three response category levels, the F-GPLS-NM-LR model achieves a particularly good fit. The mean number of components needed for the best possible parameter function estimation was 1.79, 1.67, and 1.61, respectively. Thus, the degree of dimension reduction produced in the three models is larger than in Cases 1 and 2. The mean of the accuracy of the estimated parameter functions, IMSE = 0.54, 0.38, and 0.08, respectively, shows good prediction ability in all models. However, IMSE is larger than in Case 2, but lower than in Case 1 with 4 response categories. CCR values (83.45, 85.34, and 93.86) are higher than those obtained in Cases 1 and 2. And again, these values are close for the models with 4 and 3 response categories and higher with the binary model. Case 3 box plots for the distributions of the IMSE and CCR for the model with 4 response categories are displayed in Figure 16.

Overall, the simulation results for Case 3 corresponding to 4 response categories also compared relatively well to the other two models on all criteria.

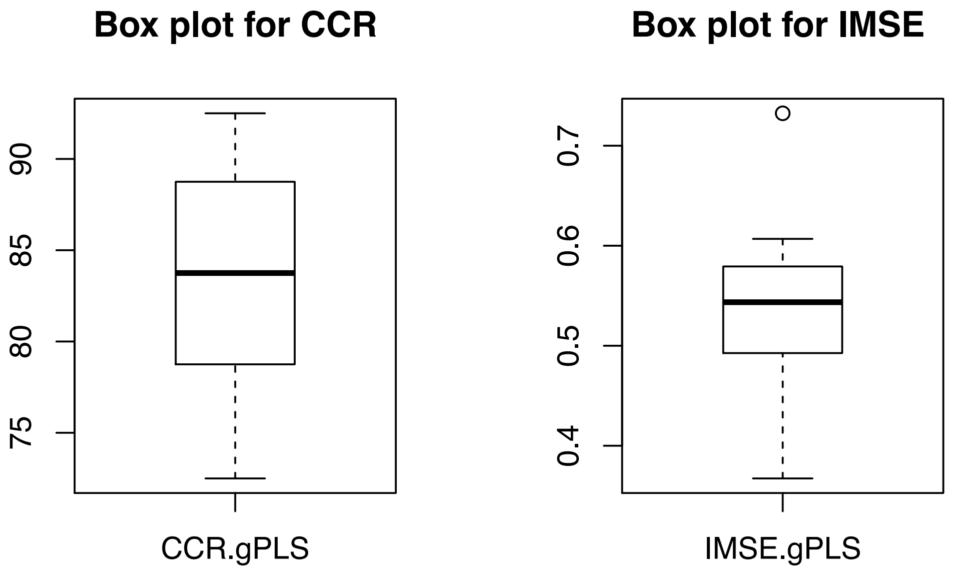


Figure 16. Box plots for the distributions of IMSE and CCR for the optimum F-GPLS-NM-LR model with 4 response categories in Case 3.

Based these results, Figure 17 shows the parameter function estimates for the three cases with 4 response categories. This figure shows that the optimum estimations of the functional parameters after using the GPLS method are very similar for Cases 1 and 3. The shape of Case 2 is roughly the same as that of Case 3. The parameter estimation is the most stable across multinomial models for Case 1. It was expected that the behavior of the functional predictors in the F-GPLS-NM-LR model with 4 response categories would be similar to the behavior with 3 and 2 categories. This was indeed the case for functional parameters estimation with respect to shape.

Overall, with respect to research questions 2 and 3, the GPLS method produced similar results under several different conditions. The results shown in Table 6 and all of the figures indicate that the GPLS method provides a clear improvement in the parameter function estimates in the model with 4 response categories in all three cases. However, there is no major difference between Cases 2 and 3 in terms of accurate estimation. On the other hand, in Case 2 the improvement may be superior to Case 1 and Case 3 in that there was a larger reduction in the number of components needed to obtain the best possible estimate in the model with 4 response categories. Ultimately, the behavior of the F-GPLS-NM-LR model in Case 1 is the best overall.

In fact, Case 1 could be a good general option for predicting responses and estimating parameter functions with the GPLS approach. In terms of how the model behaves at different levels of response categories, there are no major differences among the three cases relative to improving the estimation of functional parameters of the F-NM-LR model; researchers can choose the one that is more precise to apply. Furthermore, all models showed that the GPLS approach is a good choice for use with the F-NM-LR model for predicting responses categories associated with functional predictor. The models showed consistent performance at all 3 levels of response categories with the three different case scenarios (Case 1, Case 2, and Case 3). To confirm these results, the performance of the GPLS method was compared to the most popular method in FDA, the PCR method for the F-NM-LR model proposed by Escabias et al. (2014).

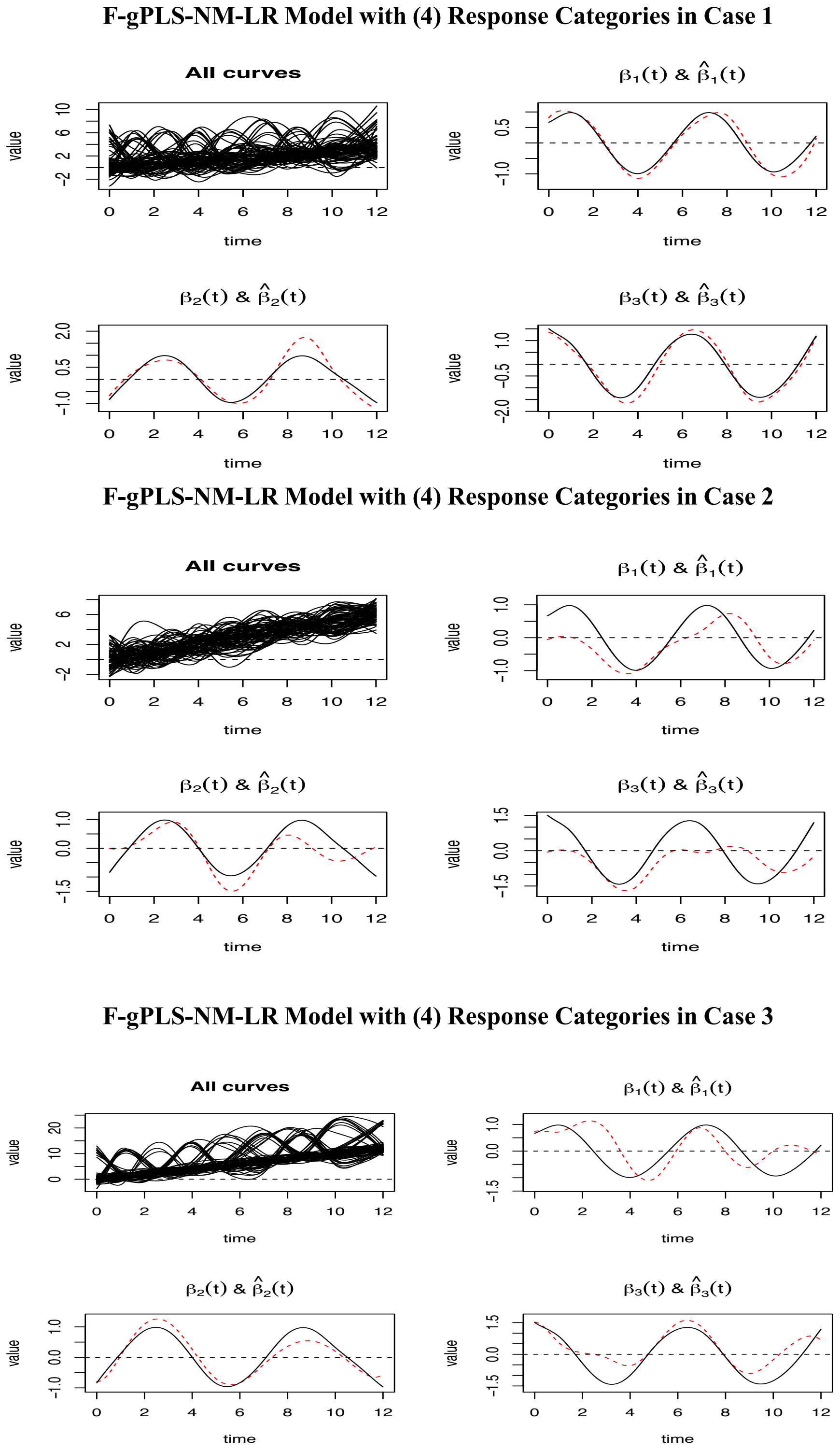


Figure 17. Simulated curves of the functional predictor variable (top left) and $y = 0$ (black dashed line). True parameter function (black solid line) and its estimation for the optimum F-GPLS-NM-LR models (red dashed line) for Case 1, Case 2, and Case 3 with 4 response categories.

Comparison of Generalized Partial Least Square and Principal Components Regression Methods in Terms of Precision

To confirm the above results, the precision of the GPLS method was compared to the PCR method for the F-NM-LR model proposed by Escabias et al. (2014), addressing research question 4 and providing a direct comparison of the FPC-NM-LR and F-GPLS-NM-LR models

> Q4 How does the precision of the generalized partial least squares regression method compare to the principal component regression method proposed by Escabias et al. (2014) under the functional nominal multinomial logit regression model?

After developing the F-GPLS-NM-LR model, the precision of the model was evaluated by comparing its performance to that of the alternative PCR method proposed Escabias et al. (2014). At step 8 of the algorithm, the PCR method fitted to the F-NM-LR model is denoted as the FPC-NM-LR model. The FPC-NM-LR model consists of a principal component analysis. More specifically, Escabias et al. (2014) proved that the best possible estimation of the parameter function (the one with the lowest IMSE) was achieved after including PCs in the model one by one according to the order of explained variance. The same process of obtaining the most accurate parameter function estimation in FPC-NM-LR model was used in this study.

To further facilitate the comparison, the goodness of fit and accuracy measures previously established (optimum number of covariates for GPLS and PCRs, minimum IMSE, and highest CCR) were used to compare the results and to evaluate each model (F-GPLS-NM-LR vs. FPC-NM-LR) across repetitions. The results, after 500 repetitions, are summarized in Table 7. This study is, to the best of the author's knowledge, the first study on the F-GPLS-NM-LR model, so it is not possible to compare the estimates of the coefficients of the functional parameter using the different approaches, because they correspond to different regression models from a theoretical point of view. The two methods are compared in terms of plots and the selected criteria (number of covariates, IMSE, and CCR).

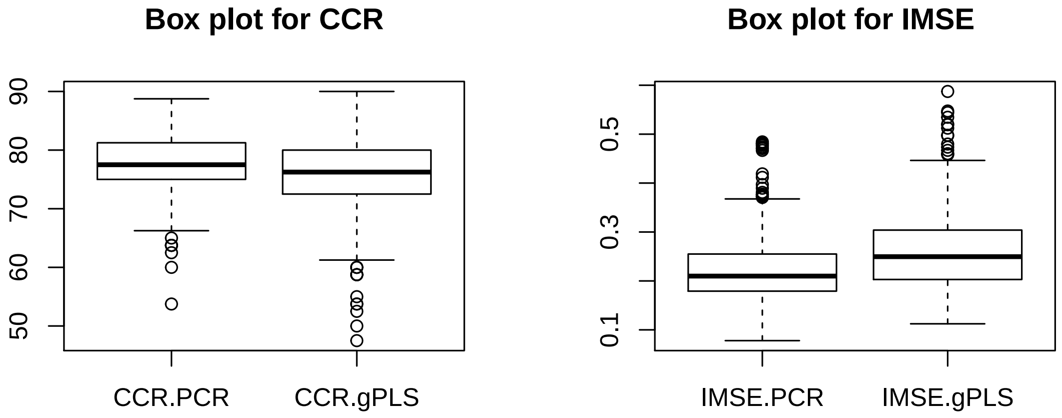


Figure 18. Box plots for the distributions of CCR and IMSE for the optimum F-GPLS-NM-LR and FPC-NM-LR models.

Table 7
Means and standard deviations of goodness of fit and accuracy measures of the F-GPLS-NM-LR and FPC-NM-LR optimum models

Measures	F-GPLS-NM-LR		FPC-NM-LR	
	Mean	SD	Mean	SD
Covariates	2.26	0.90	5.29	0.94
CCR	75.81	5.86	77.64	4.80
IMSE	0.26	0.08	0.23	0.07

To compare the models in more detail, Figure 18 shows box plots for the distributions of the mean IMSE and CCR of the optimal models generated by using GPLS and PCA estimation methods, respectively. The optimum estimates of the functional parameters for the F-GPLS-NM-LR and FPC-NM-LR models for this simulation study are displayed in Figures 18, 19, and 20. According to these figures, the parameter function estimates obtained by the F-GPLS-NM-LR and FPC-NM-LR models are very similar. Both methods provide accurate parameter function estimates with respect to the F-NM-LR model. Furthermore, Table 7 shows that the F-GPLS-NM-LR model may be superior to FPC-NM-LR model in that it provides greater dimension reduction (fewer necessary components for optimal estimates). Based on these results, it is reasonable to conclude that the proposed GPLS method with respect to the F-NM-LR model is, in general, comparable with the PCR method for classification in F-NM-LR model.

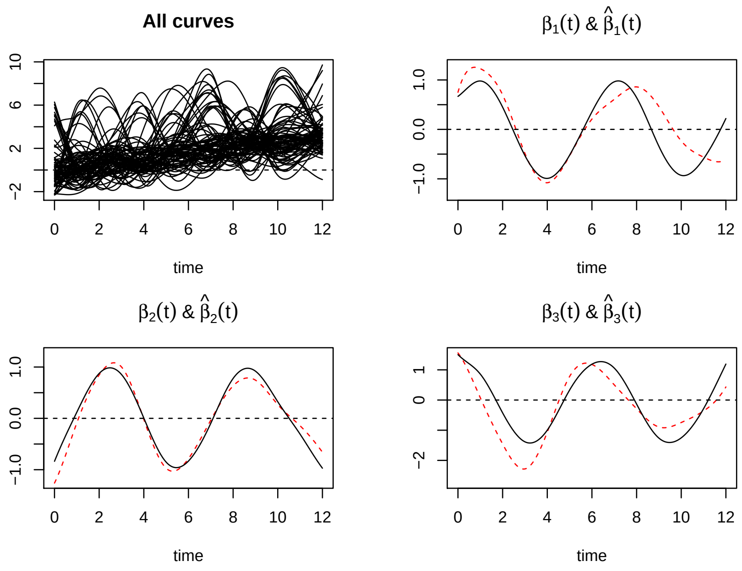


Figure 19. FPC-NM-LR model: Simulated curves of the functional predictor (top left) and $y = 0$ (black dashed line). Simulated functional parameters (black solid line) and their estimates in terms of 8 principal components, included by variability order in the F-NM-LR model (red dashed line).

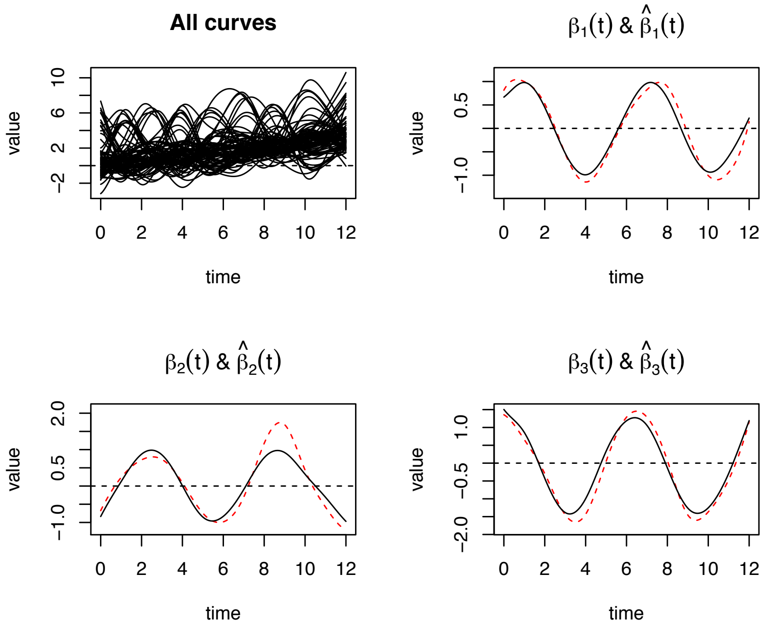


Figure 20. F-GPLS-NM-LR model: Simulated curves of the functional predictor variable (top left) and $y = 0$ (black dashed line). Simulated functional parameters (black solid line) and their estimates in terms of 3 GPLS components (red dashed line).

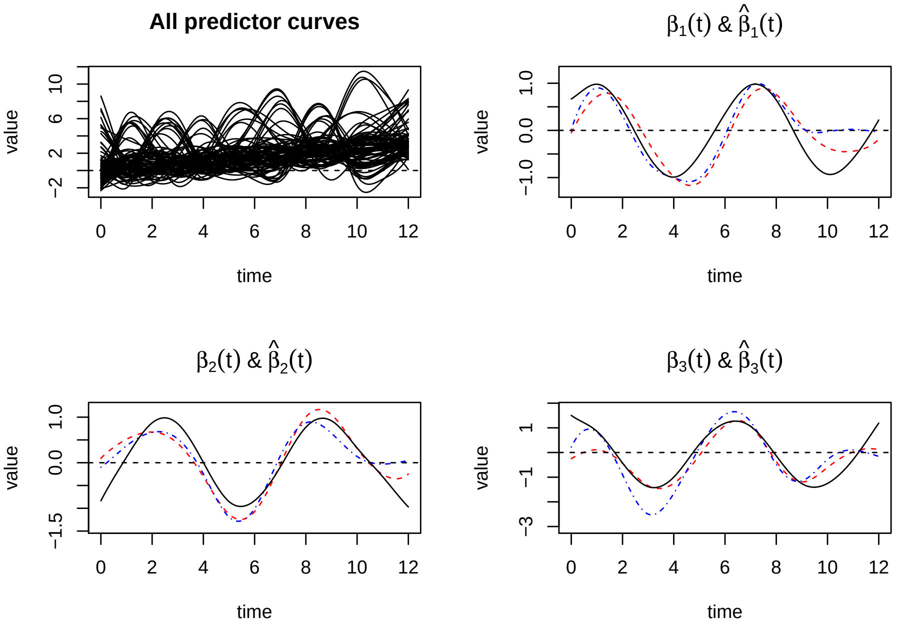


Figure 21. Simulated parameter (black solid line), FPC-NM-LR estimation in terms of 8 PCR (blue dashed line) and F-GPLS-NM-LR estimation in terms of 3 GPLS (red dashed line).

Table 7 summarizes simulation results for the three defined criteria obtained with optimal F-GPLS-NM-LR and FPC-NM-LR models with four response categories. Note that the mean number of GPLS covariates is (2.26), and this figure corresponds to the lowest IMSE (0.26); furthermore, GPLS covariates incorporate information on both the response category and functional predictor for the optimum F-GPLS-NM-LR model. In contrast, the mean number of components with the lowest IMSE (0.23) in the FPC-NM-LR model was 5.29, encompassing the ones that cumulate more than 90% of the total variability. These results show that this model required more components to achieve similar prediction accuracies to the F-GPLS-NM-LR model. The mean CCRs calculated using GPLS (75.81) and PCR (77.64) methods are similar, so both models show an acceptable degree of accuracy.

This result indicates that the main practical difference between the GPLS and PCR methods is a larger reduction of dimension on the part of the F-GPLS-NM-LR model in that fewer components are required to obtain the best possible estimation for the F-NM-LR model.

Table 7 also shows that there is no major difference in the mean IMSE and its standard deviations between the two models (mean IMSE = 0.26 with mean standard deviation (0.08) for the F-GPLS-NM-LR and mean IMSE = 0.23 with mean standard deviation (0.07) for the FPC-NM-LR model). Although there is a notable difference in the mean number of covariates for the F-GPLS-NM-LR, and FPC-NM-LR models (2.26 and 5.29, with standard deviations of 0.90 and 0.94, respectively) the mean of the CCR is not significantly different between the F-GPLS-NM-LR and FPC-NM-LR models (75.81 and 77.46 with standard deviations 5.86 and 4.80, respectively), which indicates good prediction ability on the part of both models.

In summary, the mean number of retained covariates, selected by minimizing IMSE, with the F-GPLS-NM-LR model is notably lower than with the FPC-NM-LR model. The F-GPLS-NM-LR model provides good accuracy in functional parameter estimation, with IMSE that is not significantly larger than that of the FPC -NM-LR model. However, there was no major difference in prediction accuracies (CCR) between the two models. The PCR method provides a slightly better classification rate, as shown in Table 7. It is also clear that both GPLS and PCR methods are useful for avoiding the multicollinearity problem in F-NM-LR models while retaining good prediction ability. Table 7, and Figures

17,18, and 19 demonstrate that the parameter function estimates obtained by the F-GPLS-NM-LR and FPC-NM-LR models are very similar; furthermore, these results provide empirical evidence that both methods improve the estimation of the functional parameters of the F-NM-LR model.

Overall, these results can reaffirm that the behavior of the F-GPLS-NM-LR model observed in the first simulation is consistent, so it is likely the rule instead of limited to a particular case. The conclusion regarding Q4 is that there is no major difference between the GPLS and PCA methods in their overall ability to improve the estimation of functional parameters of the F-NM-LR model, researchers can choose the one that is easiest to apply. However, there is a major difference in that the GPLS method achieves good precision and prediction with more dimension reduction (fewer covariates) than the PCA method. Given the results obtained in simulations, it is appropriate to test the F-GPLS-NM-LR model as applied to real data.

Real Data Application

The last phase of this research involved developing the R code needed to use the GPLS method for a classification problem with actual functional data to address research question 5

> Q5 How to develop an R code to fit a functional generalized partial least squares nominal multinomial logit regression model to real data?

The purpose of question 5 is to illustrate the use of the proposed F-GPLS-NM-LR model in a real application by using the developed GPLS method to classify near infrared reflectance (NIR) spectra of corn samples according to the spectrometer that generates them. Considering the functional nature of spectral data and taking into account that the absorbance at two nearby wavelengths is highly correlated, the F-GPLS-NM-LR model could provide better estimates by considering the spectrum as a curve instead of a vector. The potential use of functional data analysis in spectroscopy data is described in detail by Saeys et al. (2008).

For the purpose of comparability, the procedure used by Escabias et al. (2014) was precisely replicated. Three different instruments at Cargill Inc. (m5, mp5, and mp6 spectrometers) measured the (NIR) spectra of 80 corn samples. Measurements were taken at 2-nm intervals within a wavelength domain of [1100, 2498] nm, and data included 700 discrete observations for each sample curve. The NIR spectra of the 80 corn samples are shown in Figure 22 and Figure 23.

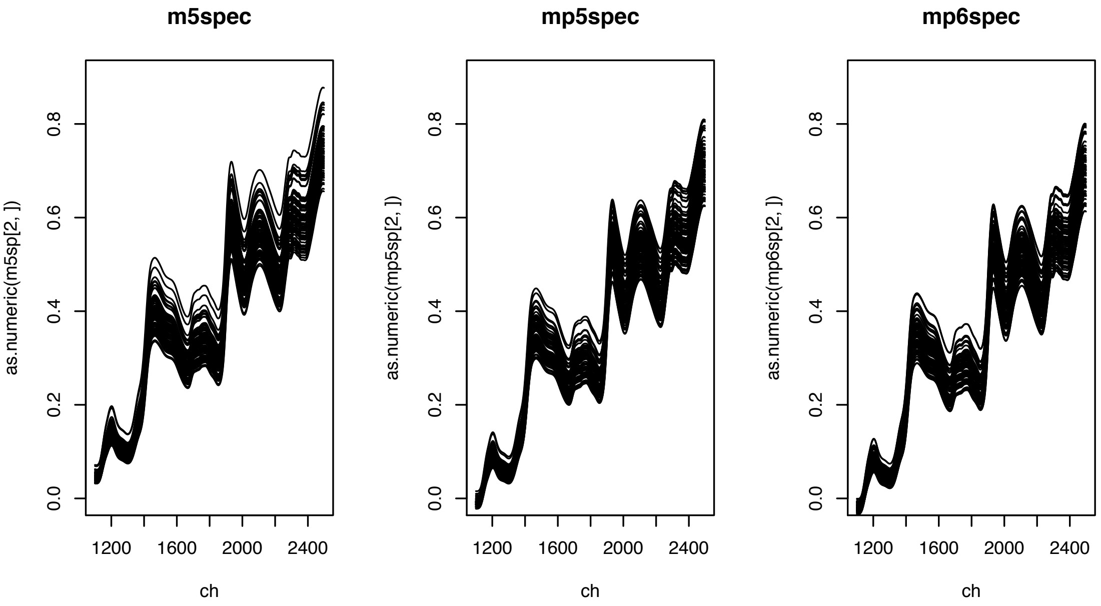


Figure 22. Curves of corn spectrum measured with the 3 different spectrometers.

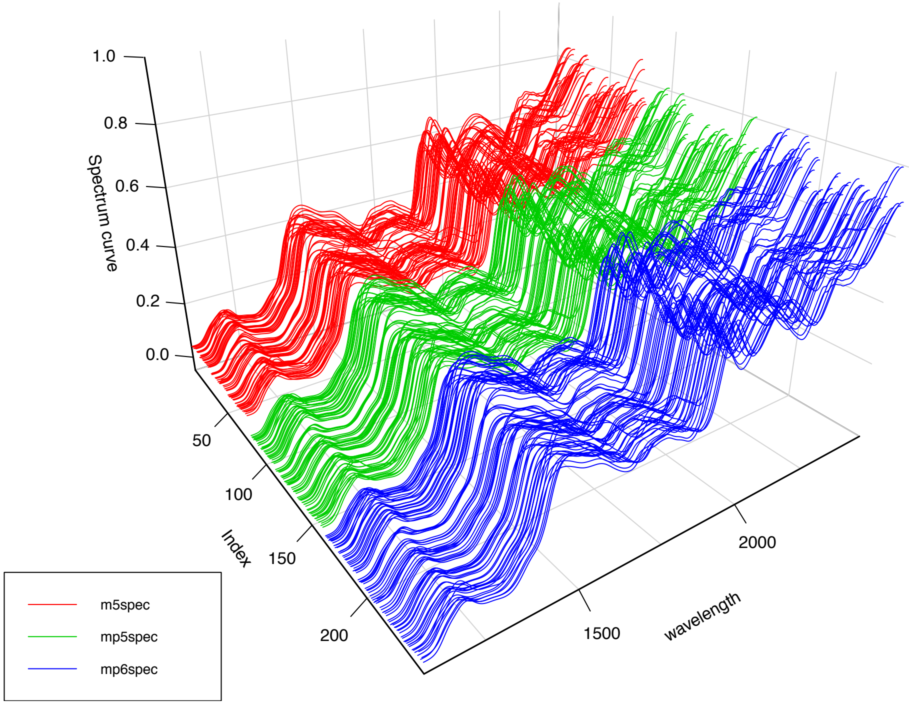


Figure 23. The NIR 3D spectra of corn spectrum measured with the 3 different spectrometers.

The original data set (80 corn samples) was split into a set of 64 samples for training purposes and a set of 16 samples for testing purposes. Taking into account that spectrometry (absorbance measured in terms of wavelength) generates a smooth curve but the observed one is measured with error, the true functional form of each sample curve was reconstructed via least squares approximation on the basis of cubic B-spline functions with 30 equally spaced knots on the wavelength range as shown in Figure 24. Then, an F-GPLS-NM-LR model was used to estimate the categorical response representing the measuring instrument used based on the NIR spectrum. The parameter estimates for this model were generated by regressing the categorical response on a set of functional GPLS components.

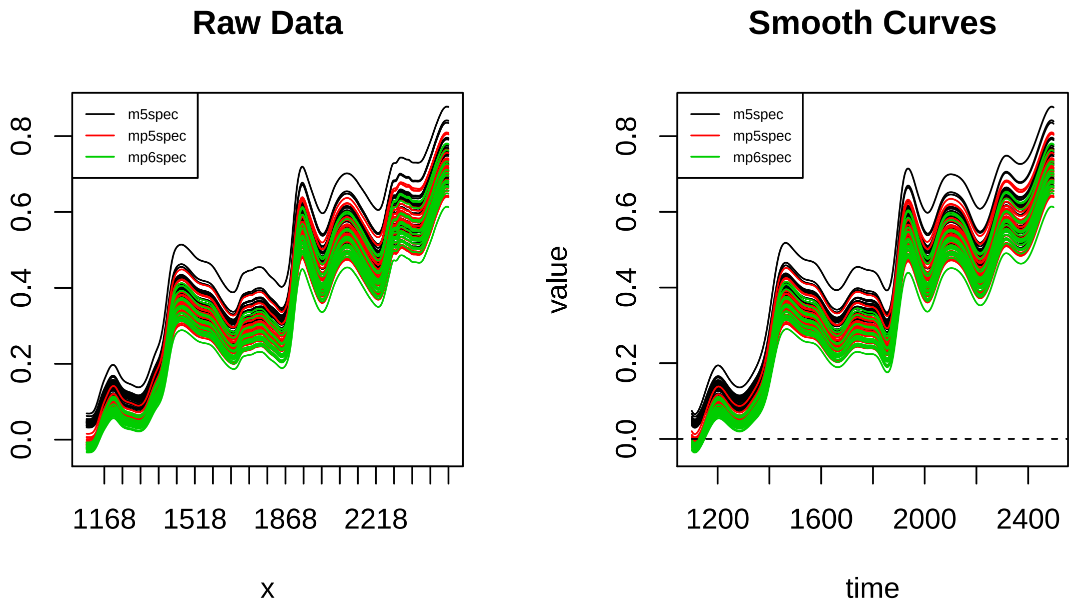


Figure 24. Observed curves of the NIR spectrum of a corn sample measured with the 3 different spectrometers (left) and its least squares approximation on the basis of cubic B-splines with 30 equally spaced knots (right).

Taking into account that in real applications the parameter function is unknown and a criterion based on minimizing the IMSE measures was used to find an accurate estimate of the functional parameters where it is impossible to calculate, this study concluded that the CVMSE criterion for model selection is a good option for predicting the response category and estimating the parameter function.

Table 8 shows goodness-of-fit and accuracy measures for the F-GPLS-NM-LR model. In almost all cases the model produces correct classification rates that are 98.43%. These results indicate that the best possible response prediction (spectrometer classification) is obtained by the model with 2 GPLS components selected by minimizing CVMSE (0.029). These models are highly accurate, providing CCR = 100% in the test set and CCR = 98.43% in the training set.

Table 8
Goodness of fit and accuracy measures of the F-GPLS-NM-LR model

Method	Covariates	CVMSE	CCR
GPLS	2	0.029	98.43

In terms of interpretability of the functional parameters, the smoothest estimations were achieved in all cases by the model with the first two GPLS components as shown in Table 8. Also, this model provides an interpretation of the odds of selecting one spectrometer over another. The functional parameter β_1 is associated with the odds of the mp5spec spectrometer's generating a specific NIR spectral curve against those of the m5spec spectrometer's (baseline) generating the curve, and β_2 is associated with odds of the mp6spec spectrometer's generating the NIR spectral curve against baseline (the m5spec).

Figure 25 shows that in low wavelengths (under 1400nm) β_1 is always greater than β_2 with negative values, and the opposite occurs in high wavelengths (over 1400nm). Furthermore, the form of the estimated functional parameters is very similar to the observed NIR spectra of the sample, which could facilitate interpreting the functional parameters. So the form of the functional parameters may suggest that the increment of NIR spectra in high wavelengths increases the odds that the spectrum is generated by mp6spec or mp5spec instead of m5spec, with the odds being higher for mp6spec than mp5spec. The opposite occurs in low wavelengths. High NIR spectra in high wavelengths are associated with higher probability with mp6spec and mp5spec spectrometers, respectively. However, in low wavelengths, the highest values of the spectrum are associated with a higher probability that the reference spectrometer is the m5spec.

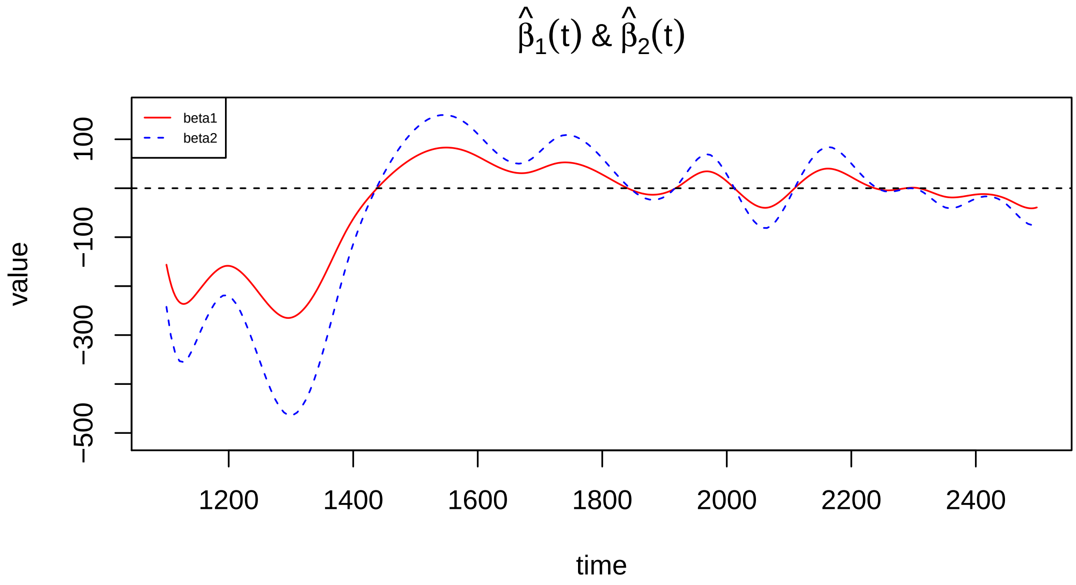


Figure 25. Estimation of the parameter functions given by the F-GPLS-NM-LR model in terms of the two GPLS components. The function $\hat{\beta}_1(t)$ is associated with the odds of spectrometer mp5spec against m5spec and $\hat{\beta}_2(t)$ is associated with the odds of mp6spec against m5spec.

In summary, in spectroscopy the measured spectra are typically plotted as a function of the wavelength. From a physical point of view it could be more informative to describe the spectrum as a function rather than as a set of points, thereby taking into account the physical background of the spectrum, and a motivating example for this functional approach is given. It is shown that for spectral data the use of cubic B-splines proves to be an appealing basis to accurately describe the data. By applying the GPLS estimation method on the data sets the predictive ability of the GPLS method for functional data analysis was found to be comparable to that of PCR method, maintaining strong performance while reducing dimensionality.

Summary and Implications for the Generalized Partial Least Square Method

In general, the performance of the proposed F-GPLS-NM-LR model was assessed in terms of several different criteria: (a) as a method to avoid the multicollinearity problem in FDA, (b) as a way to generate more accurate parameter function estimates, and (c) as a method for accurately discriminating a set of curves generated by a functional variable. Thus, based on a simulation study and the application of the method to real datasets, the results of the proposed method can be summarized as the following:

- Results from a simulation study implementation indicated that the developed F-GPLS-NM-LR model gives the most stable estimates of the parameter function.

- The developed F-GPLS-NM-LR model behaved well in all cases based on the three criteria considered for the optimum models, the IMSEs are similar for the three cases considered, with a slightly better mean CCR in Case 3.

- The F-GPLS-NM-LR provides accuracy in estimations of the parameter function similar to the alternative FPC-NM-LR model but with a larger dimension reduction. Ultimately, there is no major difference between the two methods in overall predictive ability; researchers can choose the one that is easiest to apply. However, if the objective is to predict the response, the GPLS method is a good choice because it produces similar prediction accuracy with fewer retained components.

- Goodness-of-fit measures show that the F-GPLS-NM-LR model performed well in both the simulation study and the real application, and the estimates of the parameter function are reliable. In summary, the GPLS method seems to be preferable to the F-NM-LR model, and it is a useful contribution to methods for FDA.

CHAPTER V

CONCLUSIONS

The Generalized Partial Least Squares (GPLS) approach with cubic B-spline basis expansions of the functional predictor and parameters was developed in this dissertation as a method for avoiding the multicollinearity and high dimensionality problems that preclude accurate parameter function estimates in the case of the nominal multinomial logit regression model with a functional covariate. The developed GPLS algorithm improves upon more recent methods in that the process introduces a new criterion for selecting the optimum set of GPLS components as covariate functions in the F-NM-LR model that reconstructs the parameter function with a smaller number of predictor variables. Marx (1996) introduced PLS regression using the iterated weighted least-squares algorithm for likelihood maximization for estimating the parameters in generalized linear regression. The GPLS regression estimates are derived based on Iteratively ReWeighted Partial Least Squares (IRWPLS), which defines a set of uncorrelated latent variables (as the PCs), taking into account the relationship between the nominal multinomial response and the functional predictors, respectively, and using them as predictors in the F-GPLS-NM-LR model. The F-GPLS-NM-LR model improves the accuracy of parameter function estimates and discriminates a set of curves of the model more precisely than the original F-NM-LR model. The performance of the proposed methodology was tested in a simulation study and in applications using spectrometric data sets. The results regarding the accuracy of F-GPLS-NM-LR and FPC-NM-LR models' selection criteria were tabulated and discussed.

The initial simulation developed in this dissertation followed the scheme proposed in Escabias et al. (2014), so the ML functional parameter estimates and IMSE associated to the F-GPLS-NM-LR model were provided by using all PLS components as predictor variables, yielding extremely inaccurate results. Furthermore, the solution was not unique, and interpretation in terms of odds ratios using this technique may be erroneous. In such cases, it is common to apply regularization techniques (i.e., reduction methods) to improve the precision of the model. This study extended PLS for parameter function estimation problems as introduced by Escabias et al. (2007). In order to improve the accuracy of the parameter function estimation, the IMSE and CCR were the criteria applied for selecting the optimum number of GPLS components to be introduced as predictors in the F-GPLS-NM-LR model. To validate the results obtained from the simulation process, the means and standard deviations of the goodness-of-fit and accuracy measures associated with the optimum model were computed across 500 simulation replications.

The objective of the proposed methodology and all results presented in this dissertation is to help practitioners use real-world functional data to make valid interpretations and decisions. The contribution of the proposed GPLS method is that it is a strategy that integrates PLS concepts into functional generalized linear models. A review of the literature on the functional logistic regression models indicated that no work has been done that applies the GPLS method directly to nominal multinomial data within the FDA field. In addition, the GPLS method is flexible enough to be applied to a broad family of statistical problems. The performance of the proposed methodology was studied and compared to a competing classification procedure (PCR) in a simulation study. The PCR method is a well-known tool and base solution for dimension reduction and multicollinearity problems in functional data (Escabias et al., 2014). However, this study addressed the limitations of the PCR method and the need to explore alternative estimation techniques, such as the GPLS approach.

The GPLS approach developed in this dissertation has two major advantages over the PCR method that has been used extensively in the literature on the functional logistic regression model. First, the GPLS method uses a set of uncorrelated latent variables (as the PCs), taking into account the relationship between the response and the functional predictors variables in the regression model; whereas, the traditional PCR method is subject to many criticisms based on the fact that PCs are calculated without taking into account the response variable at all. Second, classic PLS regression was designed for a continuous outcome variable with constant variance that has a linear relationship with the predictor. In this framework, the proposed GPLS method is a strategy that integrates these concepts into functional generalized linear models. This study was used to determine whether or not the F-GPLS-NM-LR model performs well enough to be considered as a base solution for issues in FDA.

The first research question addressed the process of developing a generalized partial least squares regression method for the F-NM-LR model. The GPLS approaches that agreed with orthonormal basis systems were combined with the cubic B-splines basis system. The GPLS regression method linking the nominal multinomial response variable to functional covariate showed that the model fit well based in the three criteria considered for the optimum models. This method is demonstrated in detail in chapter III, and in chapter IV results are presented where models were constructed and tested on 500 simulated datasets.

The results obtained from the simulation showed that the GPLS procedures obtained an evident dimension reduction, as is necessary for an accurate estimation of the parameter function of the F-NM-LR model, and the method showed strong curve discrimination ability. The reduction was obtained by minimizing the IMSE and including the smallest number of PLS components (in the F-GPLS-NM-LR model) necessary to generate the best possible parameter function estimates, with an acceptable degree of prediction accuracy, based on CCR.

The simulation study was also used to answer the second and third research questions regarding the behavior of the proposed method under varying circumstances. Comparisons of goodness-of-fit and accuracy measures were used to test the behavior of the F-GPLS-NM-LR model with three cases, using different functional predictors dependent on arbitrary values assigned to a and b, which were tested with four, three, and two nominal response categories. The estimation method results varied depending upon the number of nominal response categories and the arbitrary values that generated the functional predictor. The means and standard deviations of goodness-of-fit and accuracy measures were calculated for the different optimum logit models over 500 simulation replications for each model. All three cases tested generated acceptable results relative to predictive ability, and the CCRs of all three models (4, 3, and 2 response categories) within each case are almost the same. Furthermore, the optimum number of PLS components needed for each model for optimal estimations were slightly different, but once the IMSE was optimized for each model, the mean number of components was similar. Also, it is important to mention that the models with 3 categories of nominal response yielded a slightly lower CCR compared to the models with 4 and 2 response categories, but there was no significant difference in the CCR or IMSE for the models with 4 and 2 categories. Also, the Case 2 scenario resulted in a large SD for IMSE in the model with 4 response categories because the values assigned to a and b (0.5 and 1.0, respectively) created the possibility that response category 1 could be missing from the generated data. This problem is solved by eliminating the simulation repetitions that have only 3 response categories from the analysis.

Results from the simulation study also addressed the fourth research question about comparing the precision of the proposed methodology to the classic alternative PCR method for the F-NM-LR model. Based on the 500 simulation replications, the results indicated that the F-GPLS-NM-LR model provided parameter function estimates with accuracy similar to the ones generated by the alternative FPC-NM-LR model. However, with respect to the criterion for selecting the optimum number of GPLS and PCs with lower IMSE, the F-GPLS-NM-LR model showed a larger dimension reduction. On the other hand, the strength of the

F-GPLS-NM-LR model for curve discrimination (CCR) and the IMSE were not notably different from those obtained with the FPC-NM-LR model. However, in agreement with most studies that explore PCA and traditional PLS approaches (A. M. Aguilera et al., 2010; Aguilera-Morillo and Aguilera, 2015; Delaigle and Hall, 2012a; Escabias et al., 2007; Preda and Saporta, 2005), the results show that GPLS almost always requires fewer components than the PCR method. Furthermore, this study shows that parameter function estimates with GPLS are more accurate than with PCR under the F-NM-LR model. The results also highlight the necessity for functional data analysis to accurately estimate parameter functions.

In addition, this study compared solutions to the problems of high dimensionality and multicollinearity in the classification of curves based on PCR and GPLS methods. The results indicated that there is no difference between GPLS and PCR methods relative to improving the estimation of the functional parameters of the F-NM-LR model, the one that is easiest to apply is most likely suitable. However, if the main objective is to predict the response, the GPLS method is a good choice because it produces similar prediction accuracy with fewer components needed for dimension reduction.

For the application considered in this dissertation, the GPLS method was applied to spectrometric data. The spectrometric data analysis is a retrospective study of curves of spectrum wherein the proposed F-GPLS-NM-LR model was used to classify Near Infrared Reflectance (NIR) spectra of corn samples according to the spectrometer that generates them. The objective of spectroscopy data is to explain a nominal multinomial response from a functional variable (the spectrum) whose observations are functions of wavelengths rather than vectors. The proposed approach reduced the functional GPLS to the multivariate GPLS of the response on a transformation of the matrix of sample curve basis coefficients. Again, goodness-of-fit and accuracy measures, such as CVMSE, and CCR procedures were considered in selecting the number of GPLS components.

The results of the real data study indicate that the proposed method is an appropriate one for the application that was tested. Therefore, it is concluded that the GPLS approach provides the necessary dimension reduction for generating accurate estimations of the functional parameter. In the case of the F-GPLS-NM-LR model, the number of components that minimizes the CVMSE provides the most accurate estimation of the parameter function.

In summary, the purpose of F-GPLS-NM-LR model is to classify a set of curves into groups and, most importantly, to interpret the relationship between the nominal multinomial response and the functional covariate in terms of the functional parameter. Taking into account the functional form of data, the research presented here has proposed a new estimation procedure for the F-NM-LR model based on the GPLS approach with cubic B-spline basis expansions of the functional predictor and parameters. Generally, the results from the simulated examples developed in this study and a real data application allow the conclusion that the GPLS approach showed an evident dimension reduction, and it is important to highlight that accurate estimates of the parameter function are useful for designing efficient selection methods for explanatory variables in addition to facilitating interpretations and predictions. Overall, as expected, the GPLS approach presented in this dissertation produced results similar to the PCR estimation method in terms of their predictive ability and their capacity to provide accurate estimates of the functional parameter, but PCR retains more components than GPLS.

Recommendations for Future Research

The research line on GPLS methodologies with a functional covariate is not closed. The author intends to continue with development efforts in several areas: First, the estimation of ordinal response variables with the F-GPLS-NM-LR model requires development. Also, a comparative study between penalized functional PLS and GPLS methods is warranted. Furthermore, using the GPLS method when both response and predictor variables are functional is a challenge that needs to be addressed. Additionally, it is common to apply regularization techniques, (e.g., reduction methods or shrinkage methods) to improve the precision of regression models. Therefore, future research using different algorithm modification methods with the F-GPLS-NM-LR model is of interest, such as using least absolute shrinkage and selection operator (LASSO) or power of prediction measures, such as McFadden's R^2, as criteria to extract the GPLS components that explicitly consider the individual predictors, reducing the number of components needed in the final model. It is important to note that the data sets analyzed in this dissertation were considered relatively smooth, and they did not contain outliers. Because this study algorithm can be considered a form of variable selection, it is also of interest in future research to see if outlying observations would be filtered out in the GPLS component extraction process.

Closing Remarks

It was expected that the GPLS method would be useful in the development of accurate parameter estimation for FDA. The proposed method proved to be useful in this study; however, the results indicated that there is no major difference between the GPLS and PCR methods with respect to improving the estimation of the functional parameters of the F-NM-LR model; the one that is easiest to apply is appropriate. This issue is important when the key objective is an accurate estimate of the functional parameter because the approach that is used provides a measure to identify the optimum number of components when there are non-simulated examples in which the real functional parameter is unknown. If the goal is to predict the response, the GPLS method is a good alternative to PCR because it produces similar prediction accuracy with fewer retained components. Finally, in practice, when a strong degree of multicollinearity shows up, stepwise multiple regressions are commonly used. On the contrary, the GPLS method allows the retention of all variables with a strong explanatory power in the MLR model, and it accounts for the relationship between the nominal multinomial response and the predictor variables.

REFERENCES

Agresti, A. (2007). Logistic regression. *An Introduction to Categorical Data Analysis, Second Edition*, 99–136.

Agresti, A. (2010). *Analysis of ordinal categorical data* (Vol. 656). John Wiley and Sons.

Agresti, A., and Kateri, M. (2011). *Categorical data analysis.* Springer.

Aguilera, A., Aguilera-Morillo, M., and Preda, C. (2016). Penalized versions of functional pls regression. *Chemometrics and Intelligent Laboratory Systems*, *154*, 80–92.

Aguilera, A. M., Escabias, M., Preda, C., and Saporta, G. (2010). Using basis expansions for estimating functional pls regression: applications with chemometric data. *Chemometrics and Intelligent Laboratory Systems*, *104*(2), 289–305.

Aguilera, A. M., Escabias, M., and Valderrama, M. J. (2006). Using principal components for estimating logistic regression with high-dimensional multicollinear data. *Computational Statistics and Data Analysis*, *50*(8), 1905–1924.

Aguilera, A. M., Escabias, M., and Valderrama, M. J. (2008). Discussion of different logistic models with functional data. application to systemic lupus erythematosus. *Computational Statistics and Data Analysis*, *53*(1), 151–163.

Aguilera, A. M., Gutiérrez, R., and Valderrama, M. J. (1996). Approximation of estimators in the pca of a stochastic process using b-splines. *Communications in Statistics-Simulation and Computation*, *25*(3), 671–690.

Aguilera-Morillo, M. C., and Aguilera, A. M. (2015). P-spline estimation of functional classification methods for improving the quality in the food industry. *Communications in Statistics-Simulation and Computation*, *44*(10), 2513–2534.

Aguilera-Morillo, M. C., Aguilera, A. M., Escabias, M., and Valderrama, M. J. (2013). Penalized spline approaches for functional logit regression. *Test*, *22*(2), 251–277.

Bastien, P., Vinzi, V. E., and Tenenhaus, M. (2005). Pls generalised linear regression. *Computational Statistics and data analysis*, *48*(1), 17–46.

Cardot, H., Faivre, R., and Goulard, M. (2003). Functional approaches for predicting land use with the temporal evolution of coarse resolution remote sensing data. *Journal of Applied Statistics*, *30*(10), 1185–1199.

Cardot, H., Ferraty, F., and Sarda, P. (1999). Functional linear model. *Statistics and Probability Letters*, *45*(1), 11–22.

Cardot, H., and Sarda, P. (2005). Estimation in generalized linear models for functional data via penalized likelihood. *Journal of Multivariate Analysis*, *92*(1), 24–41.

Carroll, R. J., Delaigle, A., and Hall, P. (2013). Unexpected properties of bandwidth choice when smoothing discrete data for constructing a functional data classifier. *Annals of statistics*, *41*(6), 2739.

Chen, J., Phan, T. G., and Reutens, D. C. (2010). Ridge penalized logistical and ordinal partial least squares regression for predicting stroke deficit from infarct topography. *Journal of Biomedical Science and Engineering*, *3*(06), 568.

Crambes, C., Kneip, A., and Sarda, P. (2009). Smoothing splines estimators for functional linear regression. *The Annals of Statistics*, 35–72.

Dauxois, J., Pousse, A., and Romain, Y. (1982). Asymptotic theory for the principal component analysis of a vector random function: some applications to statistical inference. *Journal of multivariate analysis*, *12*(1), 136–154.

De Boor, C. (2001). Calculation of the smoothing spline with weighted roughness measure. *Mathematical Models and Methods in Applied Sciences*, *11*(01), 33–41.

De Boor, C., De Boor, C., Mathématicien, E.-U., De Boor, C., and De Boor, C. (1978). *A practical guide to splines* (Vol. 27). Springer-Verlag New York.

De Jong, S. (1993). Simpls: an alternative approach to partial least squares regression. *Chemometrics and intelligent laboratory systems*, *18*(3), 251–263.

Delaigle, A., and Hall, P. (2012a). Achieving near perfect classification for functional data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *74*(2), 267–286.

Delaigle, A., and Hall, P. (2012b). Methodology and theory for partial least squares applied to functional data. *The Annals of Statistics*, *40*(1), 322–352.

Derr, B. (2013). Ordinal response modeling with the logistic procedure. In *Sas global forum* (pp. 1–20).

Ding, B., and Gentleman, R. (2004). *Testing gene associations using co-citation* (Tech. Rep.). Technical report, The Bioconductor Project 2004, 379–383.

Ding, B., and Gentleman, R. (2005). Classification using generalized partial least squares. *Journal of Computational and Graphical Statistics*, *14*(2), 280–298.

Escabias, M., Aguilera, A. M., and Aguilera-Morillo, M. C. (2014). Functional pca and base-line logit models. *Journal of classification*, *31*(3), 296–324.

Escabias, M., Aguilera, A. M., and Valderrama, M. J. (2004). Principal component estimation of functional logistic regression: discussion of two different approaches. *Journal of Nonparametric Statistics*, *16*(3-4), 365–384.

Escabias, M., Aguilera, A. M., and Valderrama, M. J. (2005). Modeling environmental data by functional principal component logistic regression. *Environmetrics*, *16*(1), 95–107.

Escabias, M., Aguilera, A. M., and Valderrama, M. J. (2007). Functional pls logit regression model. *Computational Statistics and Data Analysis*, *51*(10), 4891–4902.

Escabias, M., Valderrama, M., and Aguilera, M. (2012). Functional data analysis in biometrics and biostatistics. *J Biom Biostat*, *3*.

Fahrmeir, L., and Tutz, G. (2001). Models for multicategorical responses: Multivariate extensions of generalized linear models. In *Multivariate statistical modelling based on generalized linear models* (pp. 69–137). Springer.

Febrero-Bande, M., and Oviedo de la Fuente, M. (2012). Statistical computing in functional data analysis: the r package fda. usc. *Journal of Statistical Software*, *51*(4), 1–28.

Ferraty, F. (2011). *Recent advances in functional data analysis and related topics.* Springer Science and Business Media.

Ferraty, F., Van Keilegom, I., and Vieu, P. (2012). Regression when both response and predictor are functions. *Journal of Multivariate Analysis*, *109*, 10–28.

Ferraty, F., and Vieu, P. (2003). Curves discrimination: a nonparametric functional approach. *Computational Statistics and Data Analysis*, *44*(1), 161–173.

Ferraty, F., and Vieu, P. (2006). *Nonparametric functional data analysis: theory and practice.* Springer Science and Business Media.

Fort, G., and Lambert-Lacroix, S. (2005). Classification using partial least squares with penalized logistic regression. *Bioinformatics*, *21*(7), 1104–1111.

Frank, L. E., and Friedman, J. H. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, *35*(2), 109–135.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, *33*(1), 1.

Friedman, J. H., and Silverman, B. W. (1989). Flexible parsimonious smoothing and additive modeling. *Technometrics*, *31*(1), 3–21.

Garthwaite, P. H. (1994). An interpretation of partial least squares. *Journal of the American Statistical Association*, *89*(425), 122–127.

Gasser, T., and Kneip, A. (1995). Searching for structure in curve samples. *Journal of the American Statistical Association*, *90*(432), 1179–1188.

Gasser, T., Köhler, W., Müller, H., Kneip, A., Largo, R., Molinari, L., and Prader, A. (1984). Velocity and acceleration of height growth using kernel estimation. *Annals of human biology*, *11*(5), 397–411.

Geladi, P., and Kowalski, B. R. (1986). Partial least-squares regression: a tutorial. *Analytica chimica acta*, *185*, 1–17.

Graven, P., and Wahba, G. (1979). Smoothing noisy data with spline functions. *Numerische Mathematik*, *31*, 377–403.

Grenander, U. (1950). Stochastic processes and statistical inference. *Arkiv för matematik*, *1*(3), 195–277.

Hall, P., Poskitt, D. S., and Presnell, B. (2001). A functional data?analytic approach to signal discrimination. *Technometrics*, *43*(1), 1–9.

Hall, P., and Vial, C. (2006). Assessing the finite dimensionality of functional data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *68*(4), 689–705.

Hervás, C., Silva, M., Gutiérrez, P. A., and Serrano, A. (2008). Multilogistic regression by evolutionary neural network as a classification tool to discriminate highly overlapping signals: Qualitative investigation of volatile organic compounds in polluted waters by using headspace-mass spectrometric analysis. *Chemometrics and Intelligent Laboratory Systems*, *92*(2), 179–185.

Horváth, L., and Kokoszka, P. (2012). *Inference for functional data with applications* (Vol. 200). Springer Science and Business Media.

Hosmer, D. W., and Lemeshow, S. (1989). Applied regression analysis. *New York, John Willey.*

Hyndman, R. J., and Booth, H. (2008). Stochastic population forecasts using functional data models for mortality, fertility and migration. *International Journal of Forecasting*, *24*(3), 323–342.

Hyndman, R. J., and Shang, H. L. (2009). Forecasting functional time series. *Journal of the Korean Statistical Society*, *38*(3), 199–211.

Hyndman, R. J., and Ullah, M. S. (2007). Robust forecasting of mortality and fertility rates: a functional data approach. *Computational Statistics and Data Analysis*, *51*(10), 4942–4956.

Indahl, U. (2005). A twist to partial least squares regression. *Journal of Chemometrics*, *19*(1), 32–44.

James, G. M. (2002). Generalized linear models with functional predictors. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *64*(3), 411–432.

Jolliffe, I. (2002). *Principal component analysis.* Wiley Online Library.

Kayano, M., Dozono, K., and Konishi, S. (2010). Functional cluster analysis via orthonormalized gaussian basis expansions and its application. *Journal of Classification*, *27*(2), 211–230.

Mahesh, S., Jayas, D., Paliwal, J., and White, N. (2015). Comparison of partial least squares regression (plsr) and principal components regression (pcr) methods for protein and hardness predictions using the near-infrared (nir) hyperspectral images of bulk samples of canadian wheat. *Food and bioprocess technology*, *8*(1), 31–40.

Martens, H., and Stark, E. (1991). Extended multiplicative signal correction and spectral interference subtraction: new preprocessing methods for near infrared spectroscopy. *Journal of pharmaceutical and biomedical analysis*, *9*(8), 625–635.

Marx, B. D. (1996). Iteratively reweighted partial least squares estimation for generalized linear regression. *Technometrics*, *38*(4), 374–381.

Marx, B. D., and Eilers, P. H. (1999). Generalized linear regression on sampled signals and curves: a p-spline approach. *Technometrics*, *41*(1), 1–13.

Massy, W. F. (1965). Principal components regression in exploratory statistical research. *Journal of the American Statistical Association*, *60*(309), 234–256.

McCullagh, P., and Nelder, J. A. (1989). *Generalized linear models, no. 37 in monograph on statistics and applied probability.* Chapman and Hall,.

McIntosh, A., Bookstein, F. L., Haxby, J. V., and Grady, C. (1996). Spatial pattern analysis of functional brain images using partial least squares. *Neuroimage*, *3*(3), 143–157.

Menard, S. (2000). Coefficients of determination for multiple logistic regression analysis. *The American Statistician*, *54*(1), 17–24.

Mittlböck, M., and Schemper, M. (1996). Explained variation for logistic regression. *Statistics in medicine*, *15*(19), 1987–1997.

Morris, J. S., and Carroll, R. J. (2006). Wavelet-based functional mixed models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *68*(2), 179–199.

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

Nason, G. P., and Silverman, B. W. (1995). The stationary wavelet transform and some statistical applications. In *Wavelets and statistics* (pp. 281–299). Springer.

Nguyen, D. V., and Rocke, D. M. (2002). Multi-class cancer classification via partial least squares with gene expression profiles. *Bioinformatics*, *18*(9), 1216–1226.

Ocaña, F. A., Aguilera, A. M., and Escabias, M. (2007). Computational considerations in functional principal component analysis. *Computational Statistics*, *22*(3), 449–465.

Park, M. Y., and Hastie, T. (2007). L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *69*(4), 659–677.

Pearson, K. (1901). Principal components analysis. *The London, Edinburgh and Dublin Philosophical Magazine and Journal*, *6*(2), 566.

Preda, C., and Saporta, G. (2005). Clusterwise pls regression on a stochastic process. *Computational Statistics and Data Analysis*, *49*(1), 99–108.

Preda, C., Saporta, G., and Lévéder, C. (2007). Pls classification of functional data. *Computational Statistics*, *22*(2), 223–235.

Ramsay, J. O. (1982). When the data are functions. *Psychometrika*, *47*(4), 379–396.

Ramsay, J. O. (2006). *Functional data analysis.* Wiley Online Library.

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

Ramsay, J. O., Heckman, N., and Silverman, B. (1997). Spline smoothing with model-based penalties. *Behavior Research Methods*, *29*(1), 99–106.

Ramsay, J. O., Hooker, G., and Graves, S. (2009). *Functional data analysis with R and MATLAB.* Springer Science and Business Media.

Ramsay, J. O., and Silverman, B. (1997). Functional linear models for scalar responses. In *Functional data analysis* (pp. 157–177). Springer.

Ramsay, J. O., and Silverman, B. (2005). Functional data analysis, 2nd edn springer. *New York*.

Ramsay, J. O., and Silverman, B. W. (2002). *Applied functional data analysis: methods and case studies* (Vol. 77). Citeseer.

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

Ratcliffe, S. J., Heller, G. Z., and Leader, L. R. (2002). Functional data analysis with application to periodically stimulated foetal heart rate data. i: Functional regression. *Statistics in Medicine*, *21*(8), 1103–1114.

Reiss, P. T., and Ogden, R. T. (2007). Functional principal component regression and functional partial least squares. *Journal of the American Statistical Association*, *102*(479), 984–996.

Rice, J. A., and Silverman, B. W. (1991). Estimating the mean and covariance structure nonparametrically when the data are curves. *Journal of the Royal Statistical Society. Series B (Methodological)*, 233–243.

Saeys, W., De Ketelaere, B., and Darius, P. (2008). Potential applications of functional data analysis in chemometrics. *Journal of chemometrics*, *22*(5), 335–344.

Saporta, G. (1981). *Méthodes exploratoires d'analyse de données temporelles* (Unpublished doctoral dissertation). Université Pierre et Marie Curie-Paris VI.

Tucker, R. S. (1938). The reasons for price rigidity. *The American Economic Review*, 41–54.

Valderrama, M. J., Ocaña, F. A., Aguilera, A. M., and Ocaña-Peinado, F. M. (2010). Forecasting pollen concentration by a two-step functional model. *Biometrics*, *66*(2), 578–585.

Wang, C.-Y., Chen, C.-T., Chiang, C.-P., Young, S.-T., Chow, S.-N., and Chiang, H. K. (1999). A probability-based multivariate statistical algorithm for autofluorescence spectroscopic identification of oral carcinogenesis. *Photochemistry and photobiology*, *69*(4), 471–477.

Wang, J.-L., Chiou, J.-M., and Mueller, H.-G. (2015). Review of functional data analysis. *arXiv preprint arXiv:1507.05135*.

Wang, S., Jank, W., and Shmueli, G. (2008). Explaining and forecasting online auction prices and their dynamics using functional data analysis. *Journal of Business and Economic Statistics*, *26*(2), 144–160.

Wichern, D. W., and Churchill, G. A. (1978). A comparison of ridge estimators. *Technometrics*, *20*(3), 301–311.

Witten, D. M., Tibshirani, R., and Hastie, T. (2009). A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, kxp008.

Wold, H. (1975). Soft modeling by latent variables: the nonlinear iterative partial least squares approach. *Perspectives in probability and statistics, papers in honour of MS Bartlett*, 520–540.

Wood, G. A., and Jennings, L. S. (1979). On the use of spline functions for data smoothing. *Journal of biomechanics*, *12*(6), 477–479.

Yao, F., Müller, H.-G., and Wang, J.-L. (2005). Functional data analysis for sparse longitudinal data. *Journal of the American Statistical Association*, *100*(470), 577–590.

APPENDIXA

EXTRA TABLES AND FIGURES

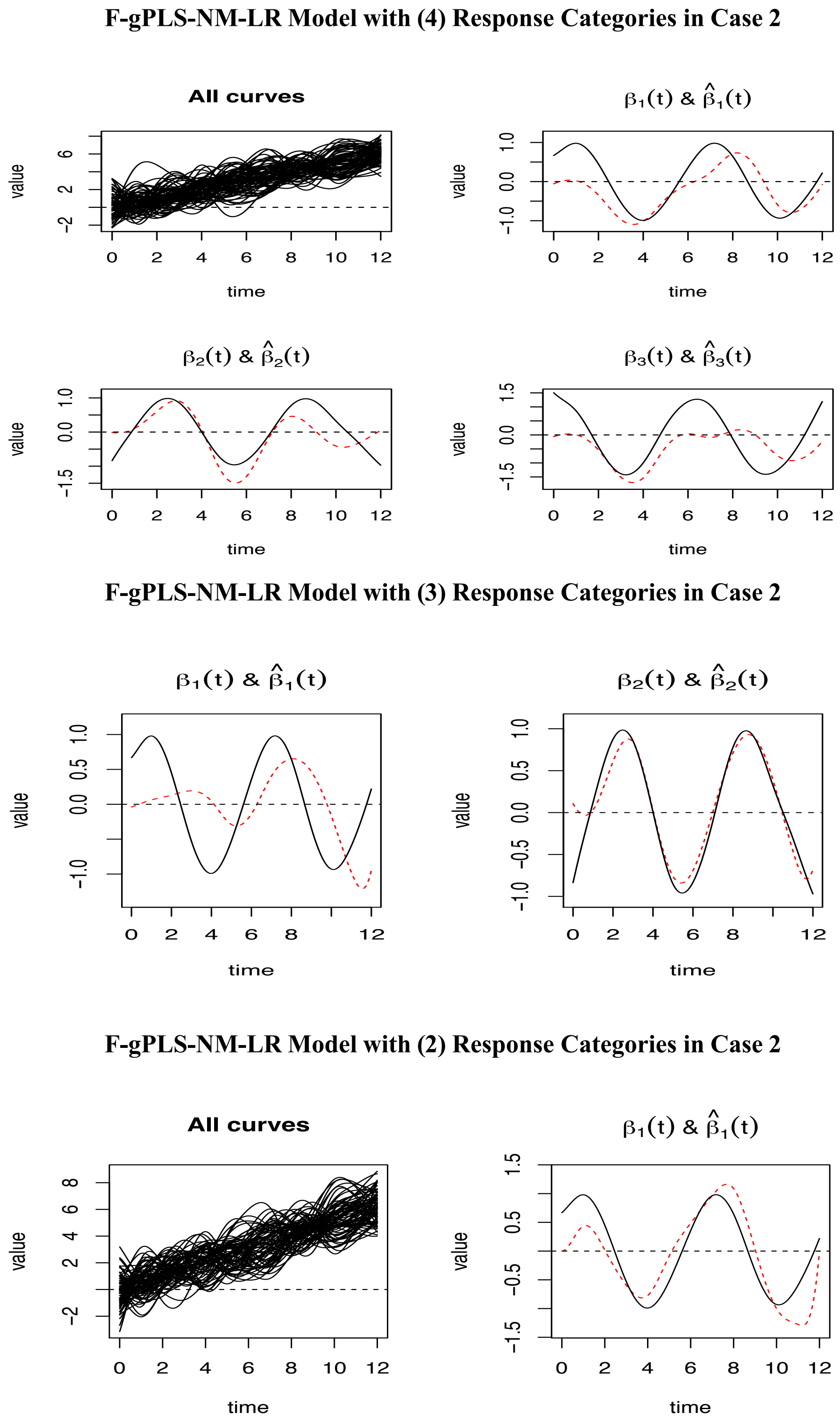
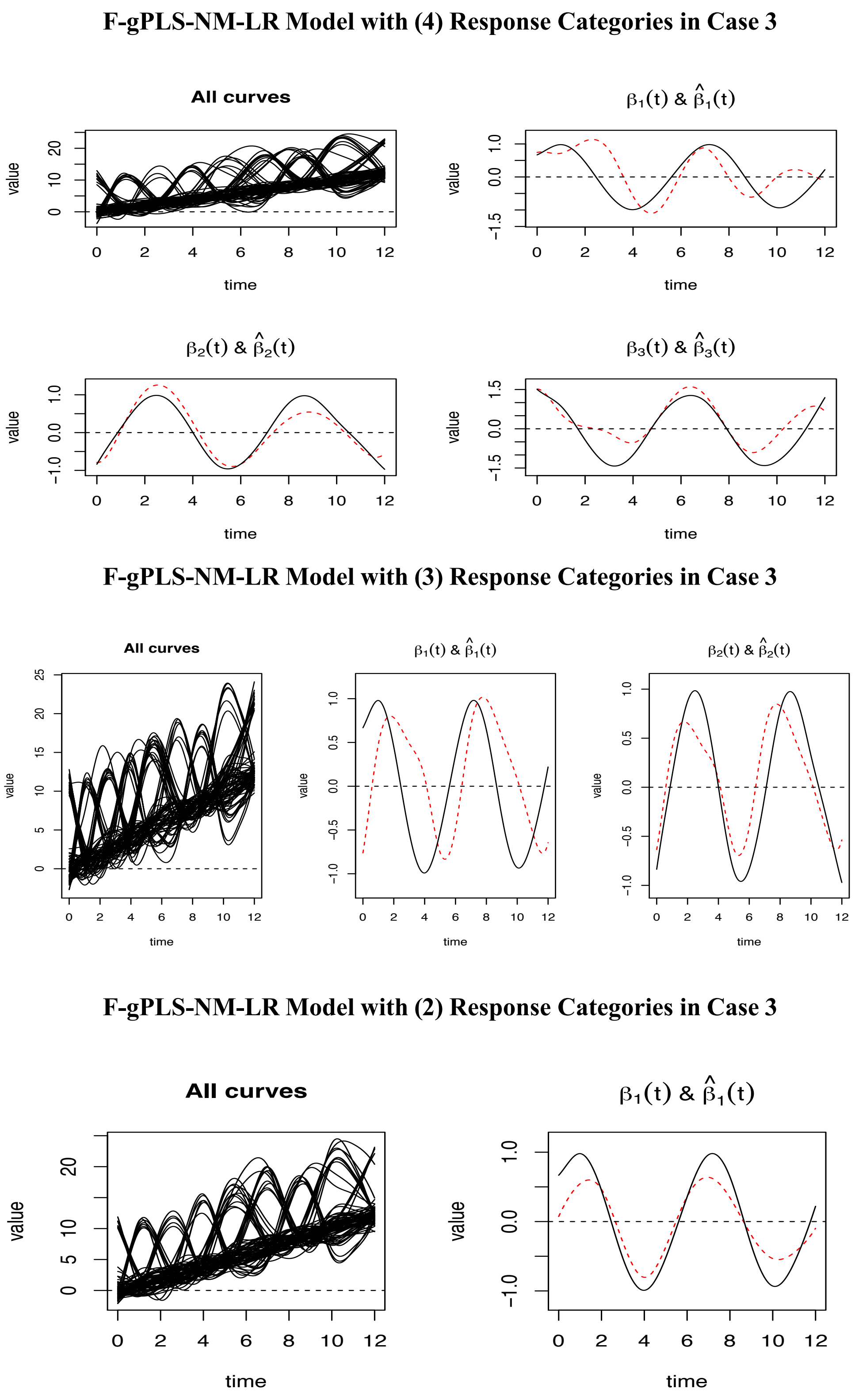


Figure 26. Simulated parameter function (solid line) and its estimation for the optimum F-GPLS-NM-LR model (broken line) with 4, 3, and 2 response categories in Case 2.



F-gPLS-NM-LR Model with (4) Response Categories in Case 3

F-gPLS-NM-LR Model with (3) Response Categories in Case 3

F-gPLS-NM-LR Model with (2) Response Categories in Case 3

Figure 27. Simulated parameter function (solid line) and its estimation for the optimum F-GPLS-NM-LR model (broken line) with 4, 3, and 2 response categories in Case 3.

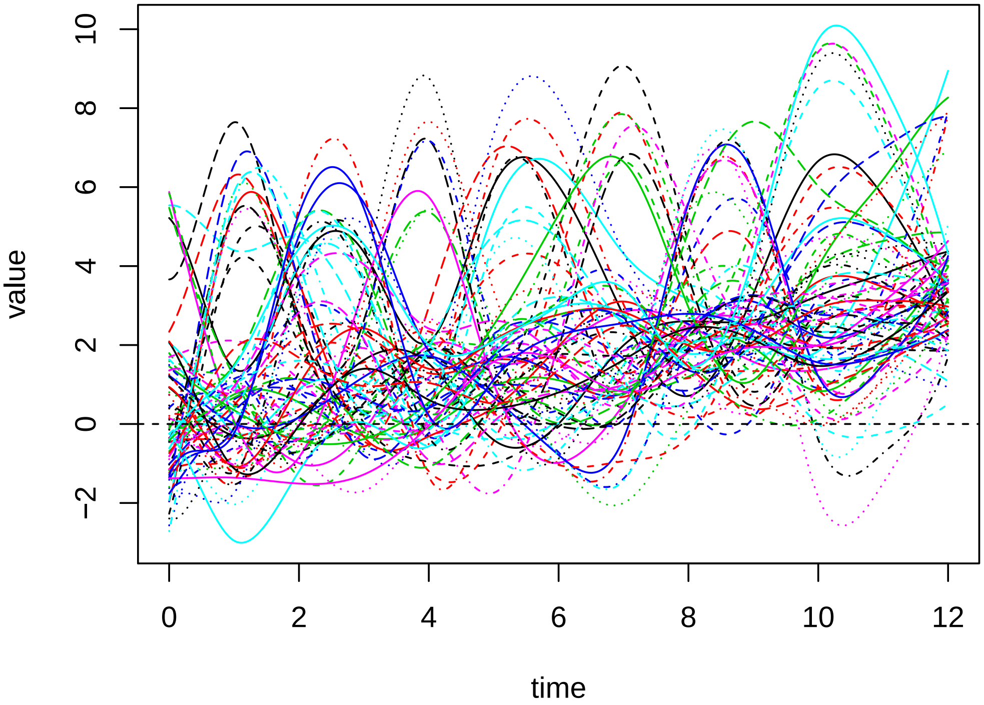


Figure 28. Simulated curves of the functional predictor variable before and their basis expansion estimation without using GPLS for the F-NM-LR model.

Table 9

Goodness of fit and accuracy measures example of the FPC-NM-LR model

N	m	IMSE	CCR
1	8	0.08	82.50
2	6	0.20	71.25
3	5	0.26	82.50
4	5	0.24	83.75
5	7	0.19	77.50
6	5	0.15	80.00
7	5	0.23	85.00
8	5	0.18	73.75
9	6	0.18	77.50
10	6	0.10	77.50
11	5	0.12	77.50

Table 10

Goodness of fit and accuracy measures example of the F-GPLS-NM-LR model.

N	m	IMSE	CCR
1	2	0.05	90.00
2	3	0.02	86.25
3	2	0.08	92.50
4	2	0.02	88.75
5	3	0.02	88.75
6	3	0.03	87.50
7	1	0.01	90.00
8	2	0.08	92.00
9	1	0.09	87.50
10	1	0.07	86.25
11	1	0.08	88.75

APPENDIXB

R CODE FOR SIMULATION STUDY AND REAL DATA

```
####################################################
##                       R code of:               ##
##   Utilities for gPLS and PCR of F-NM-LR model  ##
##                                                ##
####################################################

####################################################
# Packages #
############
library(mvtnorm)
library(splines)
library(fda)
require(fields)
library(nnet)
library(MASS)
######################################################

######################################################
# To generate X(t) and Z(t) #
#############################
t=c(0,1.1,2.5,3.7,5.1,7.3,8.5,9.6,12)
lt=length(t)
n=80
Sigma=matrix(0,lt,lt)
for(i in 1:lt){
  for (j in 1:lt){
    Sigma[i,j] = 0.5^(n*abs(t[i]-t[j]))
  }
}
Z=rmvnorm(n,mean=rep(0,lt),Sigma)
#############
a=0.25  # can change this to 0.5, 0.25 or 1
b=5     # can change this to 1.0, 5 or 10
############
X=matrix(0,n,lt)
for(i in 1:n){
  X[i,]=Z[i,]+a*t+b*rbinom(lt,1,0.1)
}
###########################################################

###########################################################
# Reconstruct the true functional curves
#  via Natural cubic spline interpolation
###########################################################
nks=101
Xf=matrix(,n,nks)
for(i in 1:n){
  sp=spline(t,X[i,],n=nks,method="natural")
  Xf[i,]=sp$y
}
matplot(t(Xf),type="l",col=1,lty=1)
############################################################
```

```
############################################################
# Generate X(t) with bsplines in fda() #
########################################
nba=15                 # number of basis
splinebasis= create.bspline.basis(c(0,12), nbasis=nba)
tn=(1:nks/nks)*12
Xt=Data2fd(tn, t(Xf), splinebasis)
############################################################

############################################################
# generate beta curves #
########################
# This shows the true beta curves!
# It can change this depend on number of response categories
########################
b1=cos(t-pi/4)
b1c=spline(t,b1,n=nks,method="natural")
b1tn=b1c$y
b1tt=Data2fd(tn, b1tn, splinebasis)    # it’s a "fd"

b2=sin(t-pi/4)
b2c=spline(t,b2,n=nks,method="natural")
b2tn=b2c$y
b2tt=Data2fd(tn, b2tn, splinebasis)    # it’s a "fd"

b3=cos(t-pi/4)-sin(t-pi/4)
b3c=spline(t,b3,n=nks,method="natural")
```

```
b3tn=b3c$y

b3tt=Data2fd(tn, b3tn, splinebasis)    # it’s a "fd"

###### The estimated beta curves######

bt<-Xt[1:3]

bt$coefs=cbind(b1tt$coef,b2tt$coef,b3tt$coef)

b1t=bt[1]

b2t=bt[2]

b3t=bt[3]


par(mfrow=c(2,2))

plot(Xt,col=1,lty=1)


plot(t,b1,col=1,lty=1,main =expression(beta[1](t)))

abline(h=0,lty=2)

lines(b1c)

lines(b1t,col=2)


plot(t,b2,col=1,lty=1, main = expression(beta[2](t)))

abline(h=0,lty=2)

lines(b2c)

lines(b2t,col=2)


plot(t,b3,col=1,lty=1, main = expression(beta[3](t)))

abline(h=0,lty=2)

lines(b3c)

lines(b3t,col=2)

###############################################################
```

```
###############################################################
# Set up intercept
#it can change this depend on number of response categories
####################
alpha=c(0.30,0.19,0.20,0)
S=length(alpha)
################################################################

################################################################
# Create a function "f.PI" to compete PI with alpha, bt and Xt #
################################################################
f.PI<-function(a0,bt0,Xt0){
  n1=length(Xt0$fdnames$reps)
  n2=length(a0)
  PI=matrix(,n1,n2)
  for(i in 1:n1){
    expv=rep(0,n2)
    for(j in 1:(n2-1)){
      expv[j]=exp(a0[j]+inprod(Xt0[i],bt0[j]))
    }
    expv[n2]=1
    PI[i,]=expv/sum(expv)
  }
  return(PI)
}
###############################################################
```

```
#####################################################################
# Create a function "f.resp" to generate a vector response using PI #
#####################################################################
f.resp<-function(PI0){
  d1=nrow(PI0)
  d2=ncol(PI0)
  res=rep(0,d1)
  for(i in 1:d1){
    y0=rmultinom(1,1,PI0[i,])
    res[i]<-which(y0==1)
  }
  res<-as.factor(res)

### Choose the last "category" as the baseline###
    resp0<-relevel(res, ref=levels(res)[d2])
  return(resp0)
}
###############################################################

###############################################################
# Generate response using PI #
##############################
PI=f.PI(alpha,bt,Xt)
resp<-f.resp(PI)
################################################################
```

```
################################################################
# Get design matrix H #
#######################
A=t(Xt$coefs)         # A is a n*p(nba) coefficient matrix
Psi=inprod(splinebasis,splinebasis)
H=A%*%Psi             # H is a n*p design matrix
# With resp, H (A and Psi), I will be able to estimate
# and betas. H is the design matrix
###############################################################


###############################################################
# Create a function "f.bcurves" for estimated beta curves     #
###############################################################
f.bcurves<-function(bh){
  bb=Xt[1:3]
  bb$coefs=bh
  par(mfrow=c(2,2))
  plot(Xt,col=1,lty=1,main="All curves")
  plot(bb[1],col=2,lty=2,
       main=expression(beta[1](t) ~'&'~ hat(beta)[1](t)),
       ylim=range(c(range(b1t$coefs),range(bb[1]$coefs))))
  lines(b1t,col=1,lty=1)

  plot(bb[2],col=2,lty=2,
       main=expression(beta[2](t) ~'&'~ hat(beta)[2](t)),
       ylim=range(c(range(b2t$coefs),range(bb[2]$coefs))))
  lines(b2t,col=1,lty=1)
  plot(bb[3],col=2,lty=2,
       main=expression(beta[3](t) ~’&’~ hat(beta)[3](t)),
       ylim=range(c(range(b3t$coefs),range(bb[3]$coefs))))
  lines(b3t,col=1,lty=1)
}
#############################################################

##############################################################
# Create a function "f.IMSE" to calculate IMSE #
################################################
f.IMSE<-function(bh){
  bb=Xt[1:3]
  bb$coefs=bh
  bb1=bb[1]
  bb2=bb[2]
  bb3=bb[3]
  dd1=b1t-bb1
  dd2=b2t-bb2
  dd3=b3t-bb3
  imse=as.numeric((inprod(dd1,dd1)/12+inprod(dd2,dd2)/12
  +inprod(dd3,dd3)/12)/3)
  return(imse)
}
###############################################################
```

```
##################################################
# Estimation of beta parameters using multinom() #
##################################################
##################################################
test=multinom(resp ~ H)
betam=summary(test)$coefficients
beta.hat=t(betam[,-1])
f.bcurves(beta.hat)
pred=predict(test)
CCR=sum(pred==resp)/length(pred)*100
IMSE=f.IMSE(beta.hat)
cat("CCR is:", CCR, "\n");cat("IMSE is:", IMSE)
###############################################################

##############################################################
# Create a function f.PCA to get beta.hat
# using PCA for a given m
# Input "resp" and "H", returns "beta.hat", "CCR" and "IMSE"
##############################################################
f.PCA<-function(resp0,H0,m0){
  co=cov(H0)
  eg=eigen(co)
  vc=cumsum(eg$values)/sum(eg$values)
  varprop=vc[m0]
  V=eg$vectors                 # p*p
  Vm=V[,1:m0]                  # p*m0
  Gamm=H0%*%Vm                 # n*m0
  test0=multinom(resp0 ~ Gamm)
  betam=summary(test0)$coefficients
  gammas.hat=t(betam[,-1])     # m0*3
  beta.hat=Vm%*%gammas.hat     # p*3
  pred0=predict(test0)
  CCR=sum(pred0==resp0)/length(pred0)*100
  IMSE=f.IMSE(beta.hat)                      # Compute IMSE
  list(beta.hat=beta.hat,CCR=CCR,IMSE=IMSE,cumvarprop=varprop)
}
##################################################################
#################################################################
# Use "f.PCA" to choose optimal "m" with the minimum IMSE #
#################################################################
imse.pca=rep(0,nba)
for(ii in 1:nba){
  fm.pca=f.PCA(resp,H,ii)
  imse.pca[ii]=fm.pca$IMSE
 }
round(cbind(1:nba,imse.pca),3)
mpca=which(imse.pca==min(imse.pca))
fpca=f.PCA(resp,H,mpca)
cat("The results for PCA method:")
cat("m is", mpca,"\n");cat("CCR is:", fpca$CCR, "\n")
;cat("IMSE is:", fpca$IMSE)
beta.hat.pca=fpca$beta.hat
f.bcurves(beta.hat)
###############################################################
```

```
############################################################
# Create a function "f.gPLS" to
# get beta.hat using PLS ( + PCA) for a given m.
# Input "resp" & "H",
# returns PLS components "T", "beta.hat", "CCR" and "IMSE"
###########################################################
 f.PLS<-function(resp0,H0,m0){
 stop.early=0
 s=length(levels(resp0))-1
 n=nrow(H0)
 p=ncol(H0)
 # Step1
 delta=matrix(,nrow=s,ncol=p)
 sedelta=matrix(,nrow=s,ncol=p)
 for(j in 1:p){
  mfit=multinom(resp0 ~ H0[,j])
  delta[,j]=summary(mfit)$coefficients[,2]
  sedelta[,j]=summary(mfit)$standard.errors[,2]
 }
 V0=matrix(,nrow=s,ncol=p)
 for(i in 1:s){
 V0[i,]=delta[i,]/sqrt(sum(delta[i,]^2))
 }
 Z=delta/sedelta
 ind1=abs(Z)>1.96
 V1=V0*ind1        # V1 is a s*p matrix
```

```
T1m=H0%*%t(V1)   # T1m is a n*s matrix
d0=apply(abs(T1m), 2, sum)!=0
T1m=T1m[,d0]
v1=ginv(t(H0)%*%H0)%*%t(H0)%*%T1m   # T=Hv
#round(sum(abs(H0%*%v1-T1m)),6)
co=cov(T1m)
eg=eigen(co)
Vpc=eg$vectors
Vpcm=cbind(Vpc[,1])
T1=T1m%*%Vpcm
w1=v1%*%Vpcm
#round(sum(abs(H0%*%w1-T1)),6)
if(m0==1){
# Get the results when m0=1
 Th=T1
 plsfit=multinom(resp0 ~ Th)
 gamm=summary(plsfit)$coefficients[,2]
 V=w1
 beta.hat=V%*%t(gamm)
 pred=predict(plsfit)
 CCR=sum(pred==resp0)/length(pred)*100
 IMSE=f.IMSE(beta.hat)
  list(Th=Th,beta.hat=beta.hat,CCR=CCR,IMSE=IMSE,
        stop.early=stop.early)
 }
 else{
 # Get the results when m0!=1
```

```
W=matrix(,nrow=ncol(H0),m0)
TT=matrix(,nrow(H0),m0)
W[,1]=w1
 TT[,1]=T1
 for(k in 2:m0){
 Tp=cbind(TT[,1:(k-1)])
 delta=matrix(,s,p)
 sedelta=matrix(,s,p)
 for(j in 1:p){
 mfit=multinom(resp0 ~ Tp + H0[,j])
 delta[,j]=summary(mfit)$coefficients[,ncol(Tp)+2]
 sedelta[,j]=summary(mfit)$standard.errors[,ncol(Tp)+2]
  }
 if(sum(is.nan(sedelta))+sum(is.nan(delta))>0){
 cat("The maxium m is", k-1)
 stop.early=1
 TT=TT[,1:(k-1)]
 W=W[,1:(k-1)]
 break
 }
Vp=matrix(,s,p)
for(i in 1:s){
Vp[i,]=delta[i,]/sqrt(sum(delta[i,]^2))
}
Z=delta/sedelta
ind2m=abs(Z)>1.96
V2m=Vp*ind2m
```

```
if(sum(abs(V2m))==0){
cat("The maxium m is", k-1)
stop.early=1
TT=TT[,1:(k-1)]
 W=W[,1:(k-1)]
 break
 }
 #else{
 R=matrix(,n,p)
 for(j in 1:p){
 R[,j]=resid(lm(H0[,j] ~ Tp))
 }
 T2m=R%*%t(V2m)
 d0=apply(abs(T2m), 2, sum)!=0
 T2m=cbind(T2m[,d0])
v2=ginv(t(H0)%*%H0)%*%t(H0)%*%T2m   # T=Hv
if(ncol(T2m)==1){
TT[,k]=T2m
 W[,k]=v2
 }
else{
co=cov(T2m)
eg=eigen(co)
Vpc=eg$vectors
Vpcm=cbind(Vpc[,1])
TT[,k]=T2m%*%Vpcm
W[,k]=v2%*%Vpcm
```

```
   }
 #}
   }
  Th=TT
  plsfit=multinom(resp0 ~ Th)
  gamm=summary(plsfit)$coefficients[,2:(ncol(Th)+1)]
   V=W
   beta.hat=V%*%t(gamm)
   pred=predict(plsfit)
   CCR=sum(pred==resp0)/length(pred)*100
   IMSE=f.IMSE(beta.hat)
   list(Th=Th,beta.hat=beta.hat,CCR=CCR,IMSE=IMSE,
           stop.early=stop.early)
  }
}
################################################################

############################################################
# Use "f.gPLS" to choose optimal "m" with the minimum IMSE #
############################################################
imse.pls=rep(100000,nba)
for(ii in 1:nba){
  fm.pls=f.PLS(resp,H,ii)
  imse.pls[ii]=fm.pls$IMSE
  if(fm.pls$stop.early==1){
    ms=ii-1
    break
  }
}
round(cbind(1:nba,imse.pls),3)
mpls=min(which(imse.pls==min(imse.pls)))
fpls=f.PLS(resp,H,mpls)
cat("The results for PLS method:")
cat("m is", mpls,"\n");cat("CCR is:", fpls$CCR, "\n");
              cat("IMSE is:", fpls$IMSE)
beta.hat.pls=fpls$beta.hat
f.bcurves(beta.hat)
##############################################################

##############################################################
# Finally, compare the results for both PCR and gPLS methods:
##############################################################
cat("The results for PCA method:")
cat("m is", mpca,"\n");cat("CCR is:", fpca$CCR, "\n");
cat("IMSE is:", fpca$IMSE)
cat("The results for PLS method:")
cat("m is", mpls,"\n");cat("CCR is:", fpls$CCR, "\n");
cat("IMSE is:", fpls$IMSE)
################################
bb.pca = Xt[1:3]
bb.pca$coefs = beta.hat.pca
bb.pls = Xt[1:3]
bb.pls$coefs = beta.hat.pls
par(mfrow=c(2,2))
```

```
plot(Xt,col=1,lty=1,main="All predictor curves")
plot(bb.pca[1],col=2,lty=2,
      main=expression(beta[1](t) ~'&'~ hat(beta)[1](t)),
     ylim=range(c(range(b1t$coefs),range(bb.pca[1]$coefs),
     range(bb.pls[1]$coefs))))
lines(bb.pls[1], col = 4, lty=4)
lines(b1t,col=1,lty=1)
plot(bb.pca[2],col=2,lty=2,
      main=expression(beta[2](t) ~'&'~ hat(beta)[2](t)),
     ylim=range(c(range(b2t$coefs),range(bb.pca[2]$coefs),
     range(bb.pls[2]$coefs))))
lines(bb.pls[2], col = 4, lty=4)
lines(b2t,col=1,lty=1)
plot(bb.pca[3],col=2,lty=2,
     main=expression(beta[3](t) ~'&'~ hat(beta)[3](t)),
     ylim=range(c(range(b3t$coefs),range(bb.pca[3]$coefs),
     range(bb.pls[3]$coefs))))
lines(bb.pls[3], col = 4, lty=4)
lines(b3t,col=1,lty=1)
legend("bottomleft",cex=0.4,l
       egend=c("True curves", "PCA", "gPLS"),
       col=c(1, 2, 4), lty = 1:3)
############################
# Box plot of the distribution of
# correlations between columns of
# the design matrix H
############################
```

```
cH=cor(Th)
index=lower.tri(cH, diag = FALSE)
upt=cH[index]
boxplot(upt)
####################################################################

####################################################################
##############
# Simulation #
##############
#####################################################################
set.seed(12345)
REP=500       # of simulations
pcat=matrix(,REP,3)
colnames(pcat)=c("m","CCR","IMSE")
rownames(pcat)=1:REP
plst=matrix(,REP,3)
colnames(plst)=c("m","CCR","IMSE")
rownames(plst)=1:REP

t1=Sys.time()
for(h in 1:REP){
  PI=f.PI(alpha,bt,Xt)
  resp<-f.resp(PI)
  A=t(Xt$coefs)  # A is a n*p(nba) coefficient matrix
  Psi=inprod(splinebasis,splinebasis)
  H=A%*%Psi      # H is a n*p design matrix
```

```
#######################
  # Use "f.PCA" to choose optimal "m" with the minimum IMSE #
#######################
  imse.pca=rep(0,nba)
  for(ii in 1:nba){
    fm.pca=f.PCA(resp,H,ii)
    imse.pca[ii]=fm.pca$IMSE
  }
  mpca=which(imse.pca==min(imse.pca))
  fpca=f.PCA(resp,H,mpca)
  pcat[h,]=c(mpca,fpca$CCR,fpca$IMSE)

  # Use "f.gPLS" to choose optimal "m" with the minimum IMSE #
  imse.pls=rep(100000,nba)
  for(ii in 1:nba){
    fm.pls=f.PLS(resp,H,ii)
    imse.pls[ii]=fm.pls$IMSE
    if(fm.pls$stop.early==1){
      ms=ii-1
      break
    }
  }
  mpls=min(which(imse.pls==min(imse.pls)))
  fpls=f.PLS(resp,H,mpls)
  plst[h,]=c(mpls,fpls$CCR,fpls$IMSE)

  print(h)
```

```
}
t2=Sys.time()
t2-t1          # tell us the time span
##########################################################
#sink("Q1,2,3,4.txt")
pcat
plst

pca.ccr=pcat[,2]
pca.imse=pcat[,3]

pls.ccr=plst[,2]
pls.imse=plst[,3]

par(mfrow=c(1,2))
boxplot(pca.ccr,pls.ccr,xaxt="n",main=("Box plot for CCR"))
axis(1, at=1:2,labels=c("CCR.PCR","CCR.gPLS"))
boxplot(pca.imse,pls.imse,xaxt="n",main=("Box plot for IMSE "))
axis(1, at=1:2,labels=c("IMSE.PCR","IMSE.gPLS"))

par(mfrow=c(1,2))
boxplot(pls.ccr,xaxt="n",main=("Box plot for CCR"))
axis(1, at=1:1,labels=c("CCR.gPLS"))
boxplot(pls.imse,xaxt="n",main=("Box plot for IMSE "))
axis(1, at=1:1,labels=c("IMSE.gPLS"))

mean(pcat[,1]);sd(pcat[,1])
mean(pcat[,2]);sd(pcat[,2])
mean(pcat[,3]);sd(pcat[,3])

mean(plst[,1]);sd(plst[,1])
mean(plst[,2]);sd(plst[,2])
mean(plst[,3]);sd(plst[,3])

t2-t1

summary(pcat)
summary(plst)

#sink()
################################################################
################################################################
```

```
###############################################################
###############################################################
##                R code of Real dataset:                    ##
#     Description:This data set consists of 80 samples        #
#     of corn measured on 3 different NIR spectrometers.      #
#     The wavelength range is 1100-2498nm                     #
#              at 2 nm intervals (700 channels).              #
###############################################################
###############################################################
##### reading m5spec, mp5spec, mp6spec data ####
setwd("/Users/amani/Desktop/myRealData")

par(mfrow=c(1,3))
m5sp=read.csv("m5spec.csv", header=F)
ch=as.numeric(m5sp[1,])
plot(ch,as.numeric(m5sp[2,]),
type="l", main="m5spec", ylim=c(0,0.9))

for(i in 3:81){
  lines(ch,as.numeric(m5sp[i,]))}

mp5sp=read.csv("mp5spec.csv", header=F)
ch=as.numeric(mp5sp[1,])
plot(ch,as.numeric(mp5sp[2,]),
type="l", main="mp5spec", ylim=c(0,0.9))

for(i in 3:81){
  lines(ch,as.numeric(mp5sp[i,]))}

mp6sp=read.csv("mp6spec.csv", header=F)
ch=as.numeric(mp6sp[1,])
plot(ch,as.numeric(mp6sp[2,]),
type="l", main="mp6spec", ylim=c(0,0.9))

for(i in 3:81){
  lines(ch,as.numeric(mp6sp[i,]))}

###########################
# draw 3D curves
###########################
x <- c(1, nrow(m5sp)*3)
y <- range(ch)
ygrid <- ch
z <- matrix(-0.05+0*rnorm(length(y)*length(x)), nrow = length(x))
dim(z)

op <- par(bg = "white")
par(oma=c(0,0,0,0))
persp(x, y, z, phi = 25, theta = 55, expand = 0.7, col = "white",
      ltheta = 120,
      ticktype = "detailed",
      #ticktype = "simple",
      #box = FALSE,
      #border = FALSE,
      xlab = "Index", ylab = "wavelength", zlab = "Spectrum curve",
      zlim = c(-0.05, 1)
) -> res
round(res, 3)

lines (trans3d(x = 243, z = 1, y = range(ch), pmat = res),
       col = "white", lty = 1, lwd = 2)
lines (trans3d(x = c(1, 243), z = 1, y = range(ch)[1], pmat = res),
       col = "white", lty = 1, lwd = 2)
lines (trans3d(x = 243, z = c(-0.05, 1), y = range(ch)[1], pmat = res),
       col = "white", lty = 1, lwd = 2)
for(i in (1:4)*50){
  lines (trans3d(x = i, z = -0.05, y = range(ch), pmat = res),
         col = "lightgrey", lty = 1)
}
for(i in c(1250, 1500, 1750, 2000, 2250)){
  lines (trans3d(x = c(1, 243), z = -0.05, y = i, pmat = res),
         col = "lightgrey", lty = 1)
}
for(i in (1:4)*50){
  lines (trans3d(x = i, z = c(-0.05, 1), y = range(ch)[2], pmat = res),
         col = "lightgrey", lty = 1)
}
for(i in 1:5/5){
  lines (trans3d(x = c(1, 243), z = i, y = range(ch)[2],
                 pmat = res), col = "lightgrey", lty = 1)
}
```

```
for(i in 1:5/5){
  lines (trans3d(x = 1, z = i, y = range(ch), pmat = res),
         col = "lightgrey", lty = 1)
}
for(i in c(1250, 1500, 1750, 2000, 2250)){
  lines (trans3d(x = 1, z = c(-0.05, 1), y = i,
                 pmat = res), col = "lightgrey", lty = 1)
}
lines (trans3d(x = 1, z = -0.05, y = range(ch), pmat = res),
       col = "lightgrey", lty = 1)
lines (trans3d(x = c(1, 243), z = -0.05, y = range(ch)[2], pmat = res),
       col = "lightgrey", lty = 1)
lines (trans3d(x = 1, z = c(-0.05, 1), y = range(ch)[2], pmat = res),
       col = "lightgrey", lty = 1)
lines (trans3d(x = c(1, 243), z = 1, y = range(ch)[2], pmat = res),
       col = "white", lty = 1, lwd = 2)
lines (trans3d(x = 1, z = 1, y = range(ch), pmat = res),
       col = "white", lty = 1, lwd = 2)
lines (trans3d(x = 243, z = c(-0.05, 1), y = range(ch)[2], pmat = res),
       col = "white", lty = 1, lwd = 2)
lines (trans3d(x = 243, z = -0.05, y = range(ch), pmat = res),
       col = 1, lty = 1, lwd = 1.5)
lines (trans3d(x = c(1, 243), z = -0.05, y = range(ch)[1], pmat = res),
       col = 1, lty = 1, lwd = 1.5)
lines (trans3d(x = 1, z = c(-0.05, 1), y = range(ch)[1], pmat = res),
       col = 1, lty = 1, lwd = 1.5)
```

```
for(i in 1:nrow(m5sp)){
  lines (trans3d(x = i, y = ygrid, z = as.numeric(m5sp[i,]), pmat = res),
         col = 2, lwd = 0.95)
}
for(i in 1:nrow(mp5sp)){
  lines (trans3d(x = nrow(m5sp) + i, y = ygrid, z = as.numeric(mp5sp[i,]),
                 pmat = res), col = 3, lwd = 0.95)
}
for(i in 1:nrow(mp6sp)){
  lines (trans3d(x = nrow(m5sp) + nrow(mp5sp) + i, y = ygrid,
                 z = as.numeric(mp6sp[i,]), pmat = res), lwd = 0.95, col = 4)
}
legend("bottomleft", legend = c("m5spec","mp5spec","mp6spec"), col = 2:4,
       lty = 1, cex = 0.8)

#############################
###  240 predictor curves ###
#############################
mp=rbind(m5sp[2:81,],mp5sp[2:81,],mp6sp[2:81,])
colnames(mp)=round(m5sp[1,])
rownames(mp)=1:240

mp1=m5sp[2:81,]
colnames(mp1)=round(m5sp[1,])
rownames(mp1)=1:80
mp2=mp5sp[2:81,]
colnames(mp2)=round(mp5sp[1,])
```

```
rownames(mp2)=1:80
mp3=mp6sp[2:81,]
colnames(mp3)=round(mp6sp[1,])
rownames(mp3)=1:80

mp1[1:6,1:6]
dim(mp1)

set.seed(20170409)
ind=sample(1:80)
ind1=ind[1:30]       # first instrument has 30 curves
ind2=ind[31:60]      # second instrument has 30 curves
ind3=ind[61:80]      # third instrument has 20 curves

xm=rbind(mp1[ind1,],mp2[ind2,],mp3[ind3,])
dim(xm)
y=c(rep(1,30),rep(2,30),rep(3,20))

par(mfrow=c(1,1))
matplot(t(xm),type="l")

# now xm is a 80 by 700 matrix of predictor curves
# y is the response
##########################################################
# X(t) with bsplines in fda() #
######################################
rangeval=c(1100,2498)
```

```
nks=30
nba=nks+2    # number of basis
splinebasis= create.bspline.basis(rangeval, nbasis=nba)
tn=(0:699/700)*(2500-1100)+1100
Xt=Data2fd(tn, t(xm), splinebasis)
#######################################
par(mfrow=c(1,2))
labl=(1:20/20)*700
xseq=seq(1100,2498,2)[labl]
matplot(t(xm),xaxt="n",type="l",main=c("Raw Data"),col=y,lty=1)
axis(1, at=labl,labels=xseq)
legend("topleft",cex=0.5,legend=c("m5spec","mp5spec","mp6spec"),
col=1:3, lty=1)
plot(Xt, main="Smooth Curves",col=y,lty=1)
legend("topleft",cex=0.5,legend=c("m5spec","mp5spec","mp6spec"),
col=1:3, lty=1)
##########################################################
# Get design matrix H #
#######################
A=t(Xt$coefs)  # A is a n*p(nba) coefficient matrix
dim(A)
Psi=inprod(splinebasis,splinebasis)
H=A%*%Psi
dim(H)
resp=as.factor(y)
resp
##########################################################
```

```
###########################################
# Split the data into train and test sets #
###########################################
# we need to split xm and y into two sets!

xtrain = rbind(mp1[ind1[1:24],], mp2[ind2[1:24],],
 mp3[ind3[1:16],])

xtest = rbind(mp1[ind1[25:30],], mp2[ind2[25:30],],
 mp3[ind3[17:20],])

ytrain = c(rep(1, 24), rep(2, 24), rep(3, 16))
ytest = c(rep(1, 6), rep(2, 6), rep(3, 4))

#xm = rbind(xtrain, xtest)
#y = c(ytrain, ytest)
#xtrain=xtest
#ytrain=ytest
#################
# training data #
#################
Xttrain=Data2fd(tn, t(xtrain), splinebasis)

par(mfrow=c(1,2))
labl=(1:20/20)*700
xseq=seq(1100,2498,2)[labl]
```

```
matplot(t(xtrain),xaxt="n",type="l",
main=c("Raw Data"),col=ytrain,lty=1)

axis(1, at=labl,labels=xseq)
legend("topleft",cex=0.5,legend=c("m5spec","mp5spec","mp6spec"),
col=1:3, lty=1)

plot(Xttrain, main="Smooth Curves",col=ytrain,lty=1)
legend("topleft",cex=0.5,legend=c("m5spec","mp5spec","mp6spec"),
 col=1:3, lty=1)

A=t(Xttrain$coefs)  # A is a n*p(nba) coefficient matrix
dim(A)
Psi=inprod(splinebasis,splinebasis)
H=A%*%Psi
dim(H)
resp=as.factor(ytrain)
length(resp)
resp
################
# testing data #
################

 Xttest=Data2fd(tn, t(xtest), splinebasis)
 A.test=t(Xttest$coefs)
 dim(A.test)
 Psi=inprod(splinebasis,splinebasis)
 H.test=A.test%*%Psi
 resp.test=as.factor(ytest)
 length(resp.test)
 resp.test

 par(mfrow=c(1,2))
 labl=(1:20/20)*700
 xseq=seq(1100,2498,2)[labl]

 matplot(t(xtest),xaxt="n",type="l",
      main=c("Raw data"),col=ytest,lty=1)

 axis(1, at=labl,labels=xseq)
 legend("topleft",cex=0.5,legend=c("m5spec","mp5spec","mp6spec"),
     col=1:3, lty=1)

 plot(Xttest, main="Smooth Curves",col=ytest,lty=1)
 legend("topleft",cex=0.5,legend=c("m5spec","mp5spec","mp6spec"),
       col=1:3, lty=1)
###########################################################
# Create a function "f.bcurves" for estimated beta curves
###########################################################
 f.bcurves<-function(bh){
  bb=Xt[1:2]
  bb$coefs=bh
  par(mfrow=c(1,1))
  plot(bb[1],col=2,lty=1,
  main=expression(hat(beta)[1](t) ~'&'~ hat(beta)[2](t))
          , ylim=range(c(bb[1]$coefs,bb[2]$coefs)))
  lines(bb[2],col=4,lty=2)
  legend("topleft",cex=0.5,legend=c("beta1","beta2"),
                   col=c(2,4), lty=1:2)

  #par(mfrow=c(1,2))
  #plot(bb[1],col=2,lty=1,main="beta1hat",
  ylim=range(c(bb[1]$coefs,bb[2]$coefs)))
   #plot(bb[2],col=4,lty=1,main="beta2hat",
   ylim=range(c(bb[1]$coefs,bb[2]$coefs)))
 }
#######################################################

#####################
# Compute pi.is.hat #
#####################
 fpihat <- function(H1, a.hat, b.hat){

  S <- length(a.hat) + 1
  nr <- nrow(H1)
  L <- matrix(NA, nrow = nr, ncol = S)
  L[,1]=rep(0, nr)
  for(s in 2:S){
    L[,s] <- rep(a.hat[s-1], nr) + H1%*%b.hat[,s-1]
  }
  pi.is.hat <- matrix(NA, nrow = nr, ncol = S)
  for(i in 1:nr){

    expl = exp(L[i,])
    if(max(expl) == Inf){
      expl0 = expl
      expl0[which(expl == Inf)] <- 1
      expl0[which(expl != Inf)] <- 0
      expl = expl0
    }
    pi.is.hat[i,] <- expl/sum(expl)
  }
  pred.y=rep(0, nr)
  for(i in 1:nr){
    pred.y[i] <- which(pi.is.hat[i,] == max(pi.is.hat[i,]))[1]
  }
  list(pih = pi.is.hat, pry = pred.y)
 }
 #fff=fpihat(H, alpha.hat, beta.hat)
###########################################################

###########################################################
# Create a function "f.PLS" to get beta.hat
#using gPLS (+ PCA) for a given m.
# Input "resp" & "H",
# returns PLS components "T", "beta.hat", "CCR" and "IMSE"
###########################################################
 f.PLS<-function(resp0,H0,m0){
```

```
stop.early=0
s=length(levels(resp0))-1
n=nrow(H0)
p=ncol(H0)

# Step1
delta=matrix(,nrow=s,ncol=p)
sedelta=matrix(,nrow=s,ncol=p)
for(j in 1:p){
  mfit=multinom(resp0 ~ H0[,j])
  delta[,j]=summary(mfit)$coefficients[,2]
  sedelta[,j]=summary(mfit)$standard.errors[,2]
}
V0=matrix(,nrow=s,ncol=p)
for(i in 1:s){
  V0[i,]=delta[i,]/sqrt(sum(delta[i,]^2))
}
Z=delta/sedelta
ind1=abs(Z)>1.96
V1=V0*ind1        # V1 is a s*p matrix
T1m=H0%*%t(V1)
d0=apply(abs(T1m), 2, sum)!=0
T1m=T1m[,d0]
v1=ginv(t(H0)%*%H0)%*%t(H0)%*%T1m   # T=Hv
#round(sum(abs(H0%*%v1-T1m)),6)
```

```
co=cov(T1m)
eg=eigen(co)
Vpc=eg$vectors
Vpcm=cbind(Vpc[,1])
T1=T1m%*%Vpcm
w1=v1%*%Vpcm
#round(sum(abs(H0%*%w1-T1)),6)

if(m0==1){

  # Get the results when m0=1
  Th=T1
  plsfit=multinom(resp0 ~ Th)
  alph=unname(summary(plsfit)$coefficients[,1])
  gamm=summary(plsfit)$coefficients[,2]
  V=w1
  beta.hat=V%*%t(gamm)
  pred=predict(plsfit)
  CCR=sum(pred==resp0)/length(pred)*100
  list(Th=Th,beta.hat=beta.hat,alpha.hat=alph,
               CCR=CCR,stop.early=stop.early)
}

else{
  # Get the results when m0!=1
  W=matrix(,nrow=ncol(H0),m0)
  TT=matrix(,nrow(H0),m0)
```

```
W[,1]=w1

TT[,1]=T1


for(k in 2:m0){


  Tp=cbind(TT[,1:(k-1)])

  delta=matrix(,s,p)

  sedelta=matrix(,s,p)

  for(j in 1:p){

    mfit=multinom(resp0 ~ Tp + H0[,j])

    delta[,j]=summary(mfit)$coefficients[,ncol(Tp)+2]

    sedelta[,j]=summary(mfit)$standard.errors[,ncol(Tp)+2]

  }


  if(sum(is.nan(sedelta))+sum(is.nan(delta))>0){

    cat("The maxium m is", k-1)

    stop.early=1

    TT=TT[,1:(k-1)]

    W=W[,1:(k-1)]

    break

  }


  Vp=matrix(,s,p)

  for(i in 1:s){

    Vp[i,]=delta[i,]/sqrt(sum(delta[i,]^2))

  }

  Z=delta/sedelta
```

```
ind2m=abs(Z)>1.96

V2m=Vp*ind2m


if(sum(abs(V2m))==0){

  cat("The maxium m is", k-1)

  stop.early=1

  TT=TT[,1:(k-1)]

  W=W[,1:(k-1)]

  break

}


#else{

R=matrix(,n,p)

for(j in 1:p){

  R[,j]=resid(lm(H0[,j] ~ Tp-1))

}

T2m=R%*%t(V2m)

d0=apply(abs(T2m), 2, sum)!=0

T2m=cbind(T2m[,d0])


v2=ginv(t(H0)%*%H0)%*%t(H0)%*%T2m   # T=Hv

round(sum(abs(H0%*%v2-T2m)),6)


if(ncol(T2m)==1){

  TT[,k]=T2m

  W[,k]=v2

}
```

```
      else{
        co=cov(T2m)
        eg=eigen(co)
        Vpc=eg$vectors
        Vpcm=cbind(Vpc[,1])
        TT[,k]=T2m%*%Vpcm
        W[,k]=v2%*%Vpcm
        #round(sum(abs(H0%*%W[,k]-TT[,k])),6)
      }
      #}
    }

    Th=TT
    plsfit=multinom(resp0 ~ Th)
    alph=unname(summary(plsfit)$coefficients[,1])
    gamm=summary(plsfit)$coefficients[,2:(ncol(Th)+1)]
    V=W
    beta.hat=V%*%t(gamm)
    pred=predict(plsfit)
    CCR=sum(pred==resp0)/length(pred)*100
    list(Th=Th,beta.hat=beta.hat,alpha.hat=alph,
            CCR=CCR,stop.early=stop.early)
  }
}
#######################################################
```

```
#########################
# CVMSE & CVCCR for PLS #
#########################
f.pls.CV <- function(H0, resp0, m0){

  nr = nrow(H0)
  S = length(levels(resp0))
  y0 = as.numeric(resp0)
  picv = matrix(0, nr, S)
  ypcv = rep(NA, nr)
  for(j in 1:nr){

    H_j = H0[-j,]
    Hj = matrix(H0[j,], nrow = 1)
    resp_j = resp0[-j]

    fpls = f.PLS(resp_j, H_j, m0)
    a.j = fpls$alpha.hat
    b.j = fpls$beta.hat
    fpi = fpihat(Hj, a.j, b.j)
    picv[j,] = fpi$pih
    ypcv[j] = fpi$pry

    print(j)
  }
  # get CVMSE
  Y0 = matrix(0, nr, S)
  for(k in 1:S){
    Y0[y0 == as.numeric(levels(resp0))[k], k] <- 1
  }
  CVMSE <- sum((Y0 - picv)^2)/(nr*S)

  # get CVCCR
  CVCCR <- sum(ypcv == y0)/nr*100

  list(predicted.pi = picv, predicted.y = ypcv,
        CVMSE = CVMSE, CVCCR = CVCCR)
}
###################################
# Using CVMSE & CVCCR to choose m #
###################################
 cvmse = NULL
 cvccr = NULL
 npls = 5
 for(i in 1:npls){
  fplscv = f.pls.CV(H, resp, i)
  cvmse[i] = fplscv$CVMSE
  cvccr[i] = fplscv$CVCCR
 }
 pls_CV = data.frame(cvmse,cvccr)
 rownames(pls_CV) = 1:npls
 pls_CV
 par(mfrow=c(1,2))
 plot(pls_CV$cvmse, type = "b", xlab = "no. of pls",
    ylab = "CVMSE")

 plot(pls_CV$cvccr, type = "b", xlab = "no. of pls",
    ylab = "CVCCR")
 par(mfrow=c(1,1))
###################
# In_sample error #
###################
 mpls=2
 fpls=f.PLS(resp,H,mpls)
 cat("The results for PLS method:")
     cat("m is", mpls,"\n");cat("CCR is:", fpls$CCR, "\n" );
     cat("CVMSE is:",fplscv$CVMSE, "\n")
 beta.hat=fpls$beta.hat
 alpha.hat=fpls$alpha.hat
 f.bcurves(beta.hat)
####################
# Out_sample error #
####################
 fpi = fpihat(H.test, alpha.hat, beta.hat)
 picv = fpi$pih
 ypcv = fpi$pry
 round(picv, 3)
 ypcv
```

```
#########
# get PMSE
##########
 nr = nrow(H.test);nr
 S = length(levels(resp.test));S
 Y0 = matrix(0, nr, S)
 y0 = as.numeric(resp.test)
 for(k in 1:S){
  Y0[y0 == as.numeric(levels(resp.test))[k], k] <- 1
 }
 PMSE <- sum((Y0 - picv)^2)/(nr*S)
##########
# get CCR
##########
 CCR <- sum(ypcv == y0)/nr *100
 cat("The results for PLS method:")
  cat("m is", mpca,"\n");cat("CCR is:", CCR, "\n");
  cat("PMSE is:", PMSE, "\n")
##############################################################
```