

Jona Cilia Rachel Cederbaum

Functional Linear Mixed Models for Complex Correlation Structures and General Sampling Grids

Dissertation an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig-Maximilians-Universität München

Eingereicht am 06. Juni 2017

Jona Cilia Rachel Cederbaum

Functional Linear Mixed Models for Complex Correlation Structures and General Sampling Grids

Dissertation an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig-Maximilians-Universität München

Eingereicht am 06. Juni 2017

Erste Berichterstatterin: Prof. Dr. Sonja Greven
Zweiter Berichterstatter: Prof. Dr. Thomas Kneib

Tag der Disputation: 10. Oktober 2017

Zusammenfassung

Der technologische Fortschritt ermöglicht es heutigen Wissenschaftlern verschiedener Bereiche, zunehmend Daten zu erheben, die aus funktionalen Beobachtungen anstelle von einzelnen Datenpunkten bestehen. Intensive Forschung im Bereich der funktionalen Datenanalyse zielte in den letzten Jahren darauf ab, Methoden zu entwickeln, um das gesamte Potenzial dieser Art von Daten zu erschließen. Viele der vorgeschlagenen Methoden basieren auf der Annahme unabhängiger funktionaler Beobachtungen. In der Praxis kann dies eine große Einschränkung darstellen, da die funktionalen Beobachtungen häufig korreliert sind, beispielsweise aufgrund wiederholter Beobachtungen pro Subjekt oder Gruppierung in den Daten.

Der Schwerpunkt dieser Dissertation liegt auf der Analyse von funktionalen Daten mit komplexen Korrelationsstrukturen. Funktionale lineare gemischte Modelle, das funktionale Pendant zu skalaren linearen gemischten Modellen, werden verwendet um korrelierte funktionale Daten zu analysieren. Hierbei werden die zufälligen Effekte der skalaren linearen gemischten Modelle durch Funktionen ersetzt, die über den Träger der beobachteten Daten variieren.

Zusätzlich zu der Annahme unabhängiger funktionaler Beobachtungen setzen die meisten bestehenden Methoden voraus, dass die funktionalen Beobachtungen an einer typischerweise großen Anzahl an Beobachtungspunkten vorliegen, die über alle Kurven gleich sind. Diese starke Anforderung wird in Anwendungen oft nicht erfüllt, da funktionale Beobachtungen häufig an kurvenspezifischen, möglicherweise wenigen, irregulären Gitterpunkten vorliegen. Mit dem Ziel, diese Einschränkung zu überwinden, liegt ein besonderer methodologischer Schwerpunkt dieser Dissertation auf der Erweiterung von funktionalen linearen gemischten Modellen und ihrer Schätzung auf Daten, die auf ungleichen Gittern oder sogar spärlich beobachtet werden.

Diese Dissertation entwickelt ein neues Modellierungsframework, das sowohl komplexe Korrelationsstrukturen zwischen funktionalen Beobachtungen als auch Beobachtungen auf generellen Gittern behandelt. Bisherige Ansätze erlauben entweder weniger allgemeine Korrelationsstrukturen oder lassen keine generellen Gitter und Spärlichkeit der Beobachtungen zu.

Dem funktionalen Charakter der Daten wird Rechnung getragen, indem die Modellterme des additiven Prädiktors in geeigneten Basen dargestellt werden. Zur Darstellung der funktionalen zufälligen Effekte werden Basen aus funktionalen Hauptkomponenten gewählt. Diese können als natürliche funktionale Erweiterungen multivariater Hauptkomponenten angesehen werden und repräsentieren somit die Hauptrichtungen der Variation in den Daten. Durch Verwendung der bedeutendsten Richtungen wird eine, für funktionale Daten besonders entscheidende, Dimensionsreduktion erreicht. In Analogie zum multivariaten Fall entsprechen die funktionalen Hauptkomponenten der funktionalen zufälligen Effekte den Eigenfunktionen der entsprechenden Kovarianzoperatoren. Die Schätzung der Kovarianzen latenter Prozesse ist ein nicht triviales Problem und stellt daher ein zentrales Element dieser Dissertation dar.

In den ersten beiden Teilen dieser Arbeit werden zwei neue Momentenschätzer für die Kovarianz latenter Prozesse vorgeschlagen. Sie unterscheiden sich in der Allgemeinheit der angenommenen Korrelationsstrukturen und der unterstützten Beobachtungsgitter. Beide Kovarianzschätzmethoden

beinhalten bivariate Glättung von einer oder mehreren Kovarianzen. Im dritten Teil wird ein schneller, symmetrischer bivariater Glättungsansatz vorgeschlagen, der besonders geeignet ist um glatte Kovarianzen zu schätzen, indem er sich deren Symmetrieeigenschaften zunutze macht. Seine Anwendung reduziert sowohl Rechenzeit als auch Speicherbedarf erheblich.

Das vorgeschlagene Modellierungsframework wird in umfangreichen Simulationsstudien evaluiert. Die Relevanz der vorgeschlagenen Methoden wird durch Analysen von Daten aus der Sprachproduktionsforschung und aus medizinischen Studien hervorgehoben. Um die praktische Verwendung der Methoden zu ermöglichen, werden open-source Implementationen in den beiden R Paketen `denseFLMM` und `sparseFLMM` zur Verfügung gestellt.

Abstract

Technological advances allow today's scientists in various fields to collect an increasing amount of data consisting of functional observations rather than single data points. Intense research in statistical methodology for functional data during the last years has aimed at developing methods that exploit the whole potential of this type of data. Many of the proposed approaches assume that the functional observations are independent. This may be very restrictive in practice, where correlation is frequently induced by, e.g., repeated observations per subject or grouping in the data.

The main focus of this thesis is on the analysis of functional data with complex correlation structures. Functional linear mixed models that represent functional counterparts to scalar linear mixed models are applied to analyze correlated functional data. The random effects of scalar linear mixed models are replaced by functions that vary over the same domain as the observed data.

In addition to assuming independent functional observations, most existing methods are restricted to functional observation that are available at a typically large number of observation points that are the same across all curves. This strong requirement is often not met in applications, where functional observations are frequently evaluated at curve-specific—possibly few—irregularly spaced points. To overcome this restriction, special methodological emphasis of this thesis is placed on the extension of functional linear mixed models and their estimation to data that are observed on unequal grids or even sparsely.

This thesis develops a new estimation framework that addresses both complex correlation structures between functional observations as well as observations on general sampling grids. Previous work is either less general in the assumed correlation structure or does not allow for general grids and sparseness. The functional nature of the data is accounted for by expanding all model terms in the additive predictor in suitable bases. For the functional random effects, bases of functional principal components are chosen. These can be seen as natural functional extensions of multivariate principal components and thus represent the dominant modes of variation in the data. Using only the most important directions provides the dimension reduction critically important for functional data analysis.

In analogy to the multivariate case, the functional principal components of the functional random effects correspond to the eigenfunctions of their respective covariance operators. How to estimate covariances of latent processes is non-trivial and thus constitutes an essential element in this thesis.

In the first two parts of this work, two novel method of moments estimators for covariances of latent processes are proposed. They differ in the generality of the assumed correlation structures and the supported sampling grids. Both covariance estimation methods involve bivariate smoothing of one or multiple covariances. In the third part, a fast symmetric bivariate smoothing approach is proposed that is particularly suited to estimate smooth covariances by taking advantage of their symmetry. Its application considerably reduces computation time and memory requirements.

The proposed modeling framework is evaluated in extensive simulation studies. The relevance of the proposed methods is highlighted in applications to data from speech production research as well as from medical studies. To allow the practical application of the methods, open-source implementations are provided in the two R add-on packages `denseFLMM` and `sparseFLMM`.

Acknowledgments

First of all I would like to thank my supervisor Sonja Greven for giving me this opportunity. I am thankful for her constant support and encouragement, for jointly overcoming one or the other challenge, for interesting impulses and perspectives, for helping me to work on a more concise writing style, and for everything else that I could learn from her.

My special thanks also go to Thomas Kneib for offering to be my second reviewer. I would like to thank him for his personal advice, for good conversations, and for his positive attitude.

Particular thanks go to my project partners and co-others Marianne Pouplier and Phil Hoole for the good collaboration and interesting insights into phonetics. Thanks to Marianne for her effort to immerse into the depths of statistical research and code development and for her help to make our work accessible to phoneticians. Thanks also for giving me the opportunity to give one of my first scientific talks in a very pleasant atmosphere.

As part of the Emmy Noether Research Group “Statistical Methods for Longitudinal Functional Data” (Emmy Noether grant GR 3793/1-1), the major part of my work was financially supported by the German Research Foundation (DFG), which I gratefully acknowledge.

I am grateful for having been a member of the LMU-Mentoring program and I would like to thank the program and my mentor Francesca Biagini for financial and ideational support.

I wish to thank Ciprian Crainiceanu for his invitation to the Johns Hopkins University. Many thanks to all the people I have met there for a great and interesting time in Baltimore and overwhelming hospitality.

I thank all my colleagues in the working group and all members of the department for a friendly atmosphere, for shared lunch and coffee breaks, for interesting discussions on statistical and non-statistical issues, and for quick amusing cool-downs in the ‘Schwabinger Bach’. Special thanks go to the working group for their tolerance of high server workloads, for—sometimes adventurous—joint excursions, and for a very fond farewell. Thanks to Clara Happ for her mathematical explanations and for proofreading parts of this thesis. I would also like to thank Thomas Augustin and Brigitte Maxa for always finding a solution for everything.

A thousand thanks go to my “roommates” Sarah Brockhaus and Fabian Scheipl for accompanying me through all ups and downs. It was a pleasure to work, laugh, and swear together. Thanks for all the interesting conversations, for listening, for sharing your thoughts (and chocolate), and for cheering me up when necessary. Many thanks also for proofreading parts of this thesis. Fabian I would especially like to thank for hosting us in his former individual office, for all his patience and support, for sharing his knowledge, for repeatedly helping me to fight the ‘Cederbaumschen Technikfluch’, and for all the funny gifs and his humor. Sarah I wish to thank for her positive attitude, for the fresh breeze she brought, for all the good questions she asked, for her calm, and for the encouragement and support to implement an R package.

Finally, I would like to give my warmest thanks to my family and friends for their unremitting support, for their understanding, for open ears, and for sometimes just not asking the wrong questions.

My deepest gratitude goes to Kai—for everything.

Contents

1	Introduction	1
1.1	Introduction to functional data and their analysis	1
1.2	Functional regression	6
1.3	Functional principal component analysis	9
1.4	Scope of this work	11
1.5	Contributing manuscripts	15
1.6	Software	16
2	Extending Linear Mixed Models and Principal Component Analysis to Functional Data	17
2.1	Linear mixed models	17
2.1.1	Scalar linear mixed models	18
2.1.2	From scalar to functional linear mixed models	22
2.2	Principal component analysis	30
2.2.1	Multivariate principal component analysis	31
2.2.2	From multivariate to functional principal component analysis	34
3	Functional Linear Mixed Models for Equal Sampling Grids	43
3.1	Introduction	44
3.2	The functional linear mixed model	46
3.3	Estimation	49
3.3.1	Estimation of the mean structure	49
3.3.2	Estimation of the covariance structure	50
3.3.3	Estimation of the eigenfunctions and eigenvalues	51
3.3.4	Prediction of the random basis weights	52
3.3.5	Extension to missing values	52
3.4	Application to tissue spectroscopy	53
3.4.1	Background and scientific questions	53
3.4.2	Application of the functional linear mixed model to the spectroscopy data	55
3.4.3	Classification of the tissue types	58

3.5	Discussion and outlook	60
4	Functional Linear Mixed Models for Unequal and Sparse Sampling Grids	63
4.1	Introduction	64
4.2	Functional linear mixed models	67
4.2.1	The general model	67
4.2.2	Special case: the FLMM for a crossed design	68
4.2.3	Irregularly and sparsely sampled functional data	68
4.3	Estimation	68
4.3.1	Step 1: Estimation of the mean function	69
4.3.2	Step 2: Estimation of the auto-covariances	70
4.3.3	Step 3: Eigen decompositions of estimated auto-covariances	71
4.3.4	Step 4: Prediction of the basis weights	72
4.4	Application to the speech production research data	74
4.4.1	Background and scientific questions	74
4.4.2	A model for the speech production research data	76
4.4.3	Application results	77
4.5	Simulations	78
4.5.1	Simulation designs	78
4.5.2	Simulation results	79
4.6	Discussion and outlook	82
5	Fast Symmetric Additive Covariance Smoothing	85
5.1	Introduction	86
5.2	Fast symmetric covariance smoothing	89
5.2.1	Model with independent curves	90
5.2.2	Estimation in the independent case	90
5.3	Fast symmetric additive covariance smoothing	93
5.3.1	General functional linear mixed model	94
5.3.2	Estimation in the general functional linear mixed model	95
5.3.3	Functional linear mixed model with crossed random intercepts	97
5.3.4	Covariance of the products of the centered functional responses	98
5.4	Application in functional principal component analysis	99
5.5	Implementation	101
5.6	Applications	102
5.6.1	CD4 cell count data	102
5.6.2	Speech production research data	102
5.7	Simulations	106
5.7.1	Simulation designs	106
5.7.2	Simulation results	107

5.7.3	Computational efficiency	110
5.7.4	Summary	110
5.8	Discussion and outlook	111
6	Concluding Summary and Outlook	113
6.1	Concluding summary	113
6.2	Outlook	116
	Appendices	123
A	Notation	125
B	Appendix of Chapter 3	129
B.1	Derivations	129
B.1.1	Representation of the covariance matrices and computational effort	129
B.1.2	Representation of the random basis weights and computational effort	131
B.2	Details on the application	131
C	Appendix of Chapter 4	133
C.1	Derivations	134
C.1.1	Empirical best linear unbiased predictor (EBLUP)	134
C.1.2	Matrices in the prediction of the basis weights as EBLUPs	134
C.1.3	Marginal bases for FPC-FAMM	136
C.2	Supplementary details on the estimation and implementation	137
C.2.1	Implementation of the auto-covariance estimation	137
C.2.2	Rescaling of the eigenvectors and eigenvalues	138
C.2.3	Truncation of the FPCs	138
C.2.4	Implementation of FPC-FAMM	138
C.2.5	Fixing the smoothing parameter in FPC-FAMM	139
C.2.6	Iterative estimation	139
C.3	Supplementary application details and results	139
C.3.1	Pre-processing	139
C.3.2	Supplementary application results	139
C.4	Supplementary simulation details and results	142
C.4.1	Measures of goodness of fit	142
C.4.2	Generation details for the sparse scenario	142
C.4.3	Results for simulations with centered and decorrelated basis weights	143
C.4.4	Results for simulations with original basis weights	153

D Appendix of Chapter 5	163
D.1 Derivations	163
D.1.1 Derivation for the covariance of the products of the centered functional responses in the general FLMM	163
D.1.2 Simplification of the covariance of the products of the centered functional responses for crossed fRIs	164
D.2 Supplementary details on the estimation and implementation	165
D.2.1 (Additive) varying coefficient model using tensor product B-splines	165
D.2.2 Form of the constraint matrix	167
D.3 Supplementary application details and results	169
D.3.1 CD4 cell count data	169
D.3.2 Speech production research data	174
D.4 Supplementary simulation details and results	176
D.4.1 Generation details	176
D.4.2 Measures of goodness of fit	177
D.4.3 Results for the scenario with independent curves	178
D.4.4 Results for the scenario with crossed fRIs	189
E Details on the Implementations	191
E.1 R add-on package <code>denseFLMM</code>	191
E.1.1 Manual of R function <code>denseFLMM</code>	191
E.1.2 R code for the application to the spectroscopy data	198
E.2 R add-on package <code>sparseFLMM</code>	201
E.2.1 Manual of R function <code>sparseFLMM</code>	201
E.2.2 Manual of R function <code>make_summation_matrix</code>	211
E.2.3 Manual of R function <code>smooth.construct.symm.smooth.spec</code>	212
E.2.4 Manual of R function <code>Predict.matrix.symm.smooth</code>	213
List of Figures	219
List of Tables	222
References	223

Chapter 1

Introduction

1.1 Introduction to functional data and their analysis

Steady technological progress in the last decades has made it increasingly affordable to collect and store a growing amount of functional data. These data have a functional nature in the sense that they can—at least theoretically—be observed in arbitrarily fine resolution. The desire to exploit the whole potential of these data in combination with the resultant challenges has put forth a new branch of statistics called functional data analysis (FDA; see, e.g., Ramsay and Silverman, 2005). As in ‘classical’ statistics, FDA is concerned with data description, exploration, and inference, where in this case the data are functions living in a suitable infinite-dimensional function space. Most commonly, these functions are real-valued one-dimensional curves, frequently observed over time. Yet, functional data can also be collected on higher-dimensional domains yielding more complex objects such as surfaces, images, or shapes. Although for ease of understanding, functional data are often introduced as functions varying over time, much more general domains such as space, wavelength, or combinations of those are possible. Examples of functional data are numerous and come from diverse fields. They include acoustic recordings in speech sciences, spectroscopy data in chemistry or medicine, climate and neuroimaging data, and data from wearable devices in health care, among many others (e.g., Pouplier et al., 2014; Reiss and Ogden, 2007; Besse et al., 2000; Zipunnikov et al., 2011; Goldsmith et al., 2015).

As, in practice, the data at hand consist of—potentially high-dimensional—vectors of discrete observations rather than continuous functions, a legitimate question is what differentiates functional data from multivariate data, justifying the need to develop new statistical theory and tools. The key difference is that functional observations can be seen as structured objects with a natural ordering in their dimensions rather than a collection of single data points. It is typically assumed that an underlying continuous stochastic process is giving rise to the data, often equipped with some kind of additional smoothness assumption reflecting the similarity of adjacent values. Smoothness can, for example, mean that the first and second derivatives exist (Ramsay and Silverman, 2005, Chapter 3). FDA accounts for the continuous nature by treating an entire function as one statistical object

and thus the unit of observation is a function. A functional data set thus consists of a sample of functions that can be regarded as realizations from a stochastic process. This allows to combine both information within and between functional observations, sometimes referred to as regularity and replication (Ramsay and Silverman, 2005, Chapter 22).

History of functional data analysis

Today's concept of functional data analysis can be dated back at least to the work of Ramsay (1982) and has gained increasing popularity since. Access to a growing amount of interesting functional data sets as well as increasing ability to store, transfer, and manipulate high-dimensional data has led to a great interest in the field. The close connection of FDA to the analysis of stochastic processes, longitudinal data and time series analysis, non-parametric modeling and functional analysis along with the broad range of applications has motivated scientists from different backgrounds to contribute to the field. This has created a versatile framework consisting of theory, methodology, and practical applications, including suitable statistical software.

The often-cited book by Ramsay and Silverman (2005), already published in its first edition in 1997, can be considered a milestone in FDA. It made the field accessible to a wide range of scientists by providing an overview with practical orientation and numerous illustrative examples. More theoretical aspects of FDA are covered in the books by Ferraty and Vieu (2006) and Horváth and Kokoszka (2012). The former concentrates on non-parametric functional data analysis and in particular deals with semi-metrics for functions on one-dimensional domains. The latter covers applicable inferential methods for functional data including statistical hypothesis tests. In contrast to Ramsay and Silverman (2005), who assume independent and identically distributed (i.i.d.) observations, both additionally cover the case of dependent functional observations, which is also a main issue in this thesis. The recent book by Hsing and Eubank (2015) elaborates mathematical concepts relevant for the (further) theoretical development of FDA. In addition to these monographs, a number of collections of different topics in FDA reflect the ongoing interest and the thematic diversity of the field (e.g., Ferraty, 2011; Ferraty and Romain, 2011; Bongiorno et al., 2014). Several special issues in various statistical journals highlight different aspects of FDA, such as the special issue on the connection to longitudinal data analysis from 2004 in *Statistica Sinica* (volume 14(3)). For some recent review articles on the development of FDA, see Cuevas (2014), Goia and Vieu (2016), and Wang et al. (2016). A systematic review on applications of FDA is given in Ullah and Finch (2013). Some software for FDA is listed in, e.g., Febrero-Bande and Oviedo de la Fuente (2012), Morris (2015), Wang et al. (2016), and Greven and Scheipl (2017), including add-on packages for R (R Core Team, 2016), MATLAB (MATLAB, 2013), and WINBugs (Lunn et al., 2000).

Assumptions

As in every statistical field, assumptions have to be made in order to provide a framework to work in. In FDA this includes the choice of a suitable function space, depending on the aspects of interest. For curves—which are in the focus of this thesis—a common choice is the $L^2(\mathcal{T})$ -space of square integrable functions defined on a bounded interval \mathcal{T} of the real numbers. The requirement that the functions

are square integrable ensures finite first and second moments, which are at the heart of many statistical procedures and key elements in this thesis. Moreover, the $L^2(\mathcal{T})$ -space with the inner product $\langle f, g \rangle = \int_{\mathcal{T}} f(t)g(t) dt$ is a separable Hilbert space (see, e.g., Horváth and Kokoszka, 2012). The existence of an inner product and its induced norm are necessary for definitions of proximity, magnitude, and for a concept of orthogonality. A *separable* Hilbert space allows the approximation of any element by a finite linear combination of a certain set of orthonormal functions, which is important for numerical treatment. Although the L^2 -space with the induced norm is suitable for many methods in FDA, other choices exist and may be preferred for certain objectives (see, e.g., Ferraty and Vieu, 2006, Chapter 1). In this thesis, all random functions are assumed to live in the L^2 -space.

In most practical applications, the trajectories are observed with additional random measurement error, which is typically assumed to be uncorrelated within and across the functional observations. A common assumption is that a random (homoscedastic) white noise measurement error with finite variance contaminates the underlying process, leading to random fluctuations around the smooth trajectories. As a consequence, denoising is often necessary in order to avoid over-fitting and to recover the underlying true process. This usually involves some kind of smoothing (e.g., using splines).

Further assumptions concern the dependency structure of the functions in the sample. A large number of approaches are based on the assumption of i.i.d. observations. As for scalar or multivariate data, however, numerous examples exist in which dependence of the observations is induced, e.g., by study design or spatial sampling. Therefore, an increasing number of approaches aim to account for correlation structures of different generality and type (see, e.g., Kokoszka, 2012).

Another important issue is the choice of the sampling grids, i.e., the discrete points on which the functions are observed. It affects the choice of a suitable estimation procedure as well as the asymptotic theory (see, e.g., Zhang and Wang, 2016). Different aspects need to be considered as the sampling grids may be assumed equal for the whole sample or observation-specific, and the grids can consist of points of different number and regularity. The majority of approaches exclusively applies to data sampled on a common grid, which facilitates notation and often speeds up computation as information can be compressed. Although some approaches allow a certain amount of missing values, this is still too restrictive for many practical applications such as medical observational studies, where measurements are frequently recorded at patient- or observation-specific time points. It is commonly distinguished between so-called ‘dense’ and ‘sparse’ sampling grids, describing the magnitude of the number of measurement points relative to the sample size. Often, functional data are termed as dense when the number of measurement points for all observations is larger than some order of the sample size. A rigorous definition is, however, still lacking (Zhang and Wang, 2016). Extensions of the concept of dense and sparse grids exist, in which grids may belong to neither category or may even be ‘ultradense’ (Wang et al., 2016). Sparsely sampled data need careful handling as some methods such as smoothing of single functions or numerical integration become very difficult or

impossible. A central assumption in the sparse setting is that in total (across all observations) the measurement points are well distributed over the whole domain. One can take advantage of this by pooling the data in order to borrow strength across different observations (e.g., Yao et al., 2005). When the functions are observed at sampling grids of different lengths, functions with a larger number of points contribute more to the estimation. Depending on the objectives of the analysis, this may be intended or not. Different propositions have been made to control the contribution; see Zhang and Wang (2016) for a study on different weighting schemes in the context of local smoothers for mean and covariance functions. To avoid confusion, it should be noted that at least three different meanings of ‘sparsity’ exist in FDA. The first relates to the sparsity of the sampling grid as described above and is the meaning referred to in this thesis. A second conception of sparsity is concerned with the sparsity of the sampled functional data in their infinite-dimensional space, which is closely related to the concept of the ‘curse of dimensionality’ (Bellman, 1961); a more detailed explanation and discussion is given in Chapter 3 of Ferraty and Vieu (2006). A third meaning of sparsity exists in the context of model selection for functional data. It generalizes the concept of parameter sparsity in parametric models to the functional framework; for a thorough discussion, see Wang and Kai (2015).

Aims and extensions

The aims in FDA are the same as in any other statistical area. They include, among others, an appropriate representation of the data and their graphical visualization, the study of similarities and differences between statistical objects, the analysis of variability, and the recognition of (ir)regularities in data. Moreover, one is often interested in relating a variable to one or a combination of multiple other variables of interest used for explanation or prediction; all with the idea in mind to simplify for better understanding and usually with a focus on interpretability and parsimony. Furthermore, the quantification of all kinds of uncertainty plays a major role.

Dealing with functional data brings new opportunities but also poses additional challenges. While on the one hand, new sources of information can be exploited by assuming ordered and related adjacent values, on the other hand, new requirements have to be met when working with complex objects in infinite-dimensional spaces. The resulting challenges are of both a theoretical and a practical nature.

Theoretical concepts such as definitions of the mean and variance of a random variable and notions of quantiles or outliers need to be rethought or adapted, if at all possible (e.g., Cuevas, 2014). In analogy to scalar or multivariate random variables, for which the mean is a scalar or rather a finite-dimensional vector, the mean of a random function is a function living in the same function space. Of major importance for this thesis is the extension of matrices to compact linear operators, that can be seen as their infinite-dimensional analogues. Thus, scalar variances and multivariate variance-covariance matrices are extended to (auto-)covariance operators. For one-dimensional functions they can be represented as surfaces, with the variance function located on the diagonal. Different notions of so-called ‘functional depth’ have been introduced that allow to order a sample of functions with respect to their centrality. This is crucial for definitions of, e.g., quantiles and outliers (see, e.g., Cuevas et al., 2007; López-Pintado and Romo, 2009). For rigorous definitions and a comprehensive

overview on extensions from a mathematical perspective, see, e.g., Horváth and Kokoszka (2012) and Hsing and Eubank (2015).

Besides the theoretical concepts that need to be extended, also the practical methods in FDA need to meet certain requirements. First, methods for functional data should make use of the assumed smoothness of the underlying process, which is not accounted for by standard methods for scalar or multivariate data. Second, the methods need to combine information both within and across functional observations. Third, as functional data are intrinsically infinite-dimensional, dimension reduction is an essential ingredient in FDA. Another reason for the importance of dimension reduction is that, in practice, functional observations often involve a high number of points per curve compared to the number of observed curves. From a standard multivariate point of view, this means that the number of variables exceeds the number of observations and thus corresponds to a high-dimensional data problem. Fourth, a frequently encountered challenge specific to the analysis of random functions is that in addition to the amplitude variation, which is commonly of interest, also phase variation is present, which might be of interest but can also be a source of confounding. A typical example is the analysis of growth curves, where the timing of important features such as rapid growth during puberty varies from subject to subject. To avoid confounding the two sources of variation, registration or alignment of functions is an important issue in FDA (Ramsay and Silverman, 2005).

All this raises the need to extend existing methods in order to match the specific characteristics of functional data and to exploit their whole potential. Functional analogues of various scalar and multivariate practical methods have been proposed in the literature and are constantly extended and combined. These include regression modeling, classification and clustering approaches, resampling methods, and dimension reduction tools; see Morris (2015) and Greven and Scheipl (2017) for a review on regression modeling (see also Section 1.2), Baffo et al. (2011) for a review on functional classification, Jacques and Preda (2014a) on functional clustering, McMurry and Politis (2011) on resampling methods for functional data, and Hall (2011) and Shang (2014) on functional principal component analysis (FPCA), which is a key tool for dimension reduction in FDA (see Section 1.3). For a recent general overview on extensions to functional data, see Cuevas (2014) and Wang et al. (2016). In addition to such extensions of well-studied approaches from scalar or multivariate to functional data, different methods for registration and alignment have been proposed; see Ramsay (2011) for an overview on curve registration. Moreover, derivatives of curves are often of interest as they have practical interpretations such as velocity and acceleration and new methods have been developed that make use of this additional available information; see, e.g., Ramsay and Silverman (2005) for an introduction.

A key element in the analysis of functional data is smoothing. It plays a prominent role for a number of reasons, which are in fact closely linked. First, it may be used for data representation. As the data come to us as discrete measurements, the observed curves may be smoothed to account for their functional nature. Commonly, a representation in basis functions is used, allowing for high flexibility while representing the functions in a finite-dimensional framework (see, e.g., Ramsay and Silverman, 2005). Thus, smoothing functional data is also closely related to dimension reduction. Second, smoothing serves as an imputation method. Once the underlying process of functional

observations is reconstructed, one can obtain values at any desired point on the domain, which is of particular importance for sparsely sampled data. Third, smoothing can be seen as a tool for regularization—a crucial issue for complex high-dimensional problems.

The next two sections (Section 1.2 and Section 1.3) introduce two of the mentioned methods for functional data, functional regression and FPCA, in more detail, as the main focus of this thesis is on a specific class of functional regression models and their estimation using dimension reduction based on FPCA.

1.2 Functional regression

Regression analysis plays a central role in statistics as in many empirical problems, the interest lies in the type and extent of the influence of one or more explanatory variables on a response variable. It is one of the most frequently used methods in many fields of application. The extension of regression analysis to the functional framework has received much attention in the literature. A wide range of regression models has been proposed, which can be applied depending on the objective and data situation. Moreover, for each type of regression model, numerous approaches to estimation exist. For a comprehensive summary on different model types and estimation approaches, see Morris (2015) and Greven and Scheipl (2017). In the following, a brief overview is given in order to put the regression models considered in this thesis into context; a more detailed description is provided in Chapter 2.

Functional regression models allow to capture the relationship among different kinds of data. On the whole, one distinguishes between models in which the response and/or (some of) the explanatory variables are functions. Correspondingly, the three functional model types are commonly termed as ‘function-on-scalar’, ‘scalar-on-function’, and ‘function-on-function’ regression (Reiss et al., 2010). Further classifications of functional regression models can be made in analogy to scalar data. The models differ in the generality of the assumed type of influence of the explanatory variables, that can, e.g., be linear or smooth and may include interactions; see, e.g., McLean et al. (2014) for smooth effects of functional covariates and Fuchs et al. (2015) for interactions of functional covariates. Different distributional assumptions of the response variables are made, including distributions from the exponential family as well as many more (e.g., James, 2002; Müller and Stadtmüller, 2005; McLean et al., 2014; Scheipl et al., 2016a). As for scalar data, most models assume independent responses, but extensions of different degree of generality exist and are also in the focus of this thesis (e.g., Brumback and Rice, 1998; Morris et al., 2003; Di et al., 2009; Greven et al., 2010; Scheipl et al., 2015). Most commonly, the focus is on the conditional mean of the response variable. However, a number of approaches beyond mean regression have been developed in the last years (e.g., Chen and Müller, 2012; Staicu et al., 2012; Brockhaus et al., 2015). Finally, models from both the frequentist and the Bayesian perspective have been proposed.

Function-on-scalar regression for independent data

This thesis deals with function-on-scalar regression models for a sample of n one-dimensional real-valued functions $Y_i \in L^2(\mathcal{T})$, $i = 1, \dots, n$, observed at possibly curve-specific measurement points $t_{ij} \in \mathcal{T}$, $j = 1, \dots, D_i$. The simplest model with functional response is given by the linear model

$$\begin{aligned} Y_i(t_{ij}) &= X_i(t_{ij}) + \varepsilon_i(t_{ij}) \\ &= \mu(t_{ij}) + E_i(t_{ij}) + \varepsilon_i(t_{ij}), \end{aligned} \tag{1.1}$$

in which the functional observations are assumed to be independent, noisy realizations from the underlying smooth random functions $X_i(t_{ij})$ with mean $\mu(t_{ij})$. More precisely, the underlying functions $X_i(t_{ij}) = \mu(t_{ij}) + E_i(t_{ij})$ are assumed to be i. i. d. copies of an L^2 -stochastic process $\{X(t) : t \in \mathcal{T}\}$. Note that whenever one assumes that the observed data are noisy realizations of a latent process, one directly finds oneself in a basic regression framework with a model as the ‘signal plus noise’ model (1.1) (see, e.g., Ramsay and Silverman, 2005, Chapter 3). The reconstruction of the underlying curves allows to model their whole course and to impute values at any point $t \in \mathcal{T}$. The relationship to explanatory variables besides the function argument can be captured by extending Model (1.1) to allow for different kinds of effects of scalar covariates which can be included by extending the mean function. For linear effects of only factor covariates, the model can be seen as a model for functional analysis of variance (FANOVA; see Zhang, 2013, for an overview).

Function-on-scalar regression models of this type are frequently applied in the analysis of longitudinal data, which consist of repeated measurements of subjects over time. Whereas classical models in longitudinal data analysis (LDA; see, e.g., Diggle et al., 2002) are strictly parametric, the use of FDA allows to relax the assumptions both for fixed and random effects. The latent process $E_i(t_{ij})$ in Model (1.1) extends the notion of a random effect for subject i by capturing the within-subject correlation along t . For a comprehensive discussion on the connection between LDA and FDA, see Rice (2004). Typically, longitudinal data are irregularly spaced and the number of measurements per subject are often small. As many approaches in FDA are restricted to data sampled on a common, fine grid, a number of extensions have been proposed that are specifically designed for irregularly or sparsely sampled longitudinal data (e.g., Staniswalis and Lee, 1998; James et al., 2000; Yao et al., 2005; Peng and Paul, 2009; Chen and Wang, 2011; Xiao et al., 2017).

Function-on-scalar regression for dependent data

Model (1.1) accounts for the within-curve correlation of the functional responses but different curves are assumed to be independent. The assumption of independent curves is very restrictive and unrealistic for many applications. Dependence may be introduced, for example, when multiple curves per subjects are measured, e.g., over time (longitudinal functional data), or when subjects are nested within groups. Moreover, experimental setups with crossed designs are frequently encountered, e.g., in the speech sciences, where curves are measured for subject-item combinations.

Scalar correlated data are commonly analyzed using linear mixed models (LMMs; see, e.g., Pinheiro and Bates, 2000). A natural extension to correlated functional data are functional linear mixed

models (FLMMs; see, e.g., Morris, 2015, for a discussion and further references). The term ‘functional mixed effects model’ was first introduced by Guo (2002). Their model, however, accounts only for within-curve correlation and is thus suited for a non-parametric analysis of longitudinal data rather than of correlated functional data. In contrast, in this thesis the term FLMM is used for models that are able to capture both within- and between-curve correlation. As in the scalar case, FLMMs contain both fixed and random effects, characterizing the population average and observation unit-specific deviations, respectively. The key difference is that the random effects in the FLMM are random functions varying over \mathcal{T} , which accounts for the functional nature of the response variables. In contrast to usual scalar LMMs, in which random intercepts and slopes only allow linear deviations from the population average, functional random effects are much more flexible. For curves observed on a common grid, an FLMM can be thought of as an LMM at each observation point $t \in \mathcal{T}$, where additionally smoothness is assured along the domain. An FLMM with one functional random intercept $B_i(t_{ijk})$ for each subject i , $i = 1, \dots, n$, is given by

$$Y_{ij}(t_{ijk}) = \mu(t_{ijk}) + B_i(t_{ijk}) + E_{ij}(t_{ijk}) + \varepsilon_{ij}(t_{ijk}), \quad j = 1, \dots, J_i, \quad k = 1, \dots, D_{ij}, \quad (1.2)$$

where $Y_{ij}(t_{ijk})$ denotes the value of the response curve j of subject i at the curve-specific point t_{ijk} . The response $Y_{ij}(t_{ijk})$ is additively decomposed into a global mean function $\mu(t_{ijk})$, the functional random intercept for subject i , $B_i(t_{ijk})$, and a random curve-specific smooth deviation in the form of a smooth residual curve $E_{ij}(t_{ijk})$. The smooth residual curve can in fact be seen as a curve-specific functional random intercept. Additional random measurement error $\varepsilon_{ij}(t_{ijk})$ captures random uncorrelated variation within each curve. Note that in case of equal grids, the notation can be simplified. Various extensions of Model (1.2) (mainly for equal grids) have been proposed in the literature. The models mainly differ in the generality of the assumed correlation structure in the data and in the type of influence of the covariates on the response. For a review on FLMMs, see Liu and Guo (2012), Morris (2015), and Greven and Scheipl (2017); see also the introductions of Chapters 3, 4, and 5. Greven and Scheipl (2017) also show how FLMMs can be embedded in a framework for general functional regression models.

In this thesis, a very general class of FLMMs is considered that includes Model (1.1) and Model (1.2) as special cases. The FLMMs apply to data with a broad range of correlation structures and accommodate functional random intercepts and slopes. Different linear and smooth effects of scalar covariates and their interactions can be included. More details on the general class of FLMMs are provided in Chapter 2 and in the subsequent chapters.

Estimation

A number of different estimation approaches for function-on-scalar regression models and for FLMMs in particular have been proposed. Although the model formulation is essentially the same for data that are observed on a common, dense grid and data that are irregularly sampled with possibly few points per curve, the sampling grid plays an important role in the estimation. In particular, the

sampling grid becomes crucial for the choice of an appropriate dimension reduction and smoothing technique. A comprehensive overview on different approaches to represent and estimate models with functional responses is presented in Greven and Scheipl (2017).

A common approach is to pre-smooth each vector of observations, often using a basis function expansion. The data are then treated as continuous functions for further analysis (see, e.g., Besse and Ramsay, 1986; Ramsay and Dalzell, 1991; Ramsay and Silverman, 2005). This may, however, become very difficult or impossible for sparsely observed data. Moreover, pre-smoothing the data usually implies that any variability discarded by the smoothing method, in particular the measurement error, is not accounted for in the following estimation steps.

An alternative approach, which is pursued in this thesis, models the raw data directly and accounts for the functional nature in the additive predictor by expanding the model terms in suitable basis functions. This second approach is not only more suitable for sparsely sampled data and allows to account for the measurement error in subsequent estimation steps, it is also more easily extendable to generalized responses. Besides, this approach has the advantage that the model boils down to a model for scalar data as the finite evaluations of the responses are modeled rather than the continuous curves. This allows to take advantage of well-established methods and flexible algorithms for scalar data (see, e.g., Scheipl et al., 2015; Greven and Scheipl, 2017). In particular, there is a close connection to varying coefficient models (Hastie and Tibshirani, 1993). The type of basis in which the model terms are expanded can be chosen according to the data. Commonly, splines, wavelets, and Fourier bases are used to model smooth, spiky, and periodic data, respectively. These are usually combined with regularization penalties that control the bias-variance trade-off. Another basis choice for smooth data is an expansion in a small number of functional principal components (FPCs). The number of FPCs then serves as discrete regularization parameter. The expansion in FPC basis functions has several advantages and has become a popular choice in FDA. As the model terms are expanded in FPC bases in this thesis, a short introduction to FPCA is given in the following.

1.3 Functional principal component analysis

Functional principal component analysis is a key tool in FDA. It can be seen as the natural functional extension of multivariate principal component analysis (PCA; see Jolliffe, 2002), as described in more detail in Chapter 2. In analogy to multivariate PCA, FPCA uses an orthogonal transformation of the data to a (typically) lower-dimensional feature space while retaining as much as possible of the variation in the data. Thus, FPCA is suitable for dimension reduction when a large amount of variability in the data can be described by a small number of dominant modes of variation, which here again are functions. These modes are termed functional principal components. FPCA finds its application as explanatory tool for data representation and visualization and is frequently used prior to the application of other statistical methods, such as classification or clustering. Moreover, the expansion of model terms in FPC bases as an alternative to spline bases is particularly attractive, which is mainly due to three favorable properties. First, the basis functions are estimated from the data and not arbitrarily chosen. Second, the FPC expansion allows for an explicit variance

decomposition of the variability in the data. Third, an FPC basis gives the best linear approximation in a space with finite dimension N , in the sense that no other N -dimensional basis explains more of the variation with respect to the L^2 -norm (see, e.g., Wang et al., 2016). This yields a particularly parsimonious basis which is advantageous for interpretation, prediction, and computation.

The foundation for FPCA from a mathematical perspective was laid by Loève (1946) and Karhunen (1947), who provide an optimal representation of a continuous stochastic process as linear combination of orthogonal functions, which is known as the Karhunen-Loève (KL) or FPC(A) expansion. The KL expansion is frequently used in many different disciplines such as signal detection or atmospheric science; for an overview, see Chapter 12.3 of Jolliffe (2002). In the statistical context, it was first applied to study inference for random functions by Grenander (1950), followed by an application to growth curves by Rao (1958). Important theoretic results for FPCA were obtained by Dauxois et al. (1982), Bosq (2000), Yao et al. (2005), and Hall and Hosseini-Nasab (2006), among many others. Up to now, it has been an active field of research with many contributions of both theoretical and practical nature. Extensions include different robust versions of FPCA (e.g., Locantore et al., 1999; Gervini, 2008) and multivariate FPCA for studying the simultaneous variation of multiple random functions based on a multivariate extension of the KL expansion; see Ramsay and Silverman (2005) for an introduction and Happ and Greven (2017) for an extension to data observed on different (dimensional) domains. For a survey of FPCA, see Shang (2014).

In the multivariate case, PCA results in an eigen analysis of the variance-covariance matrix of the data. The eigenvectors specify the main directions of variation and the corresponding eigenvalues quantify the amount of variability explained by each direction. Likewise, the functional analogue boils down to an eigen analysis of the covariance operator, based on the spectral theorem for compact symmetric, bounded linear operators in separable Hilbert spaces (see, e.g., Horváth and Kokoszka, 2012). The finite-dimensional eigenvectors in the multivariate case are replaced by infinite-dimensional eigenfunctions. Again, the amount of explained variability by the FPCs is quantified by the corresponding real-valued, non-negative eigenvalues, which are sorted in descending order.

As the covariance operator is at the heart of FPCA, practical applications require an appropriate estimate of its covariance kernel, also termed covariance function. A number of different approaches for the estimation of covariance functions that apply to different data situations have been proposed in the literature (e.g., Staniswalis and Lee, 1998; Yao et al., 2003, 2005; Di et al., 2009; Greven et al., 2010; Kauermann and Wegener, 2011; Shou et al., 2015; Xiao et al., 2017). They differ in the generality of the assumed correlation structure in the data (if any) as well as in the generality of the sampling grid. Moreover, some approaches explicitly take advantage of the specific properties of covariance functions, e.g., their symmetry (Xiao et al., 2017).

Commonly, covariance functions are assumed to be smooth. Thus, smoothing becomes neces-

sary when the data are either observed with error and/or not observed on a common, dense grid. At least four approaches that differ in when and how to apply smoothing in FPCA exist. The first pre-smooths the observed curves before applying FPCA (see, e.g., Besse and Ramsay, 1986; Ramsay and Dalzell, 1991). This has several disadvantages for the same reasons explained in the regression context above (see Section 1.2). The second approach adds a roughness penalty term to obtain smooth FPCs (see Silverman, 1996; Ramsay and Silverman, 2005; Huang et al., 2008). The third directly estimates smooth FPCs under an orthonormality constraint (see James et al., 2000; Peng and Paul, 2009; Goldsmith et al., 2015). The fourth uses bivariate smoothing of the empirical covariance function, for which different smoothing techniques can be applied, including local smoother (e.g., Yao et al., 2003, 2005) or penalized splines (e.g., Di et al., 2009; Greven et al., 2010; Di et al., 2014). This last approach to FPC estimation is studied in this thesis.

The FPC scores are given by the projection of the data in the direction of the FPCs. Based on the KL expansion, they can also be interpreted as the (random) weights of the contribution of the FPCs to each curve. They give insight into the individual structure of each curve and can be used for further analysis. In order to draw the parallel to basis expansions in general, the FPC scores are termed FPC weights or simply basis weights throughout this thesis.

Traditionally, the random FPC weights are predicted using numerical integration (see Ramsay and Silverman, 2005). This, however, only works (well) for uncorrelated functional data observed without error on dense grids. In more general data settings, numerical integration may be inaccurate and biased (see Yao et al., 2005). As a remedy, one can use that FPC expansions of random functions yield a clear separation into a random part (the individual basis weights) and into a functional deterministic part (the common FPCs); see, e.g., Panaretos and Tavakoli (2013). For the fourth approach to FPC estimation, this separation allows to predict the FPC weights as random effects in a mixed model framework for scalar data; see, e.g., Di et al. (2009), Greven et al. (2010), and for a special case Yao et al. (2005).

In this thesis, the mixed model framework is used to obtain predictions of the random basis weights to represent functional random effects in FLMMs with a very general correlation structure.

1.4 Scope of this work

This thesis proposes a computationally efficient estimation framework for a very general class of functional linear mixed models. The estimation is based on dimension reduction using functional principal component analysis combined with mixed model methodology. The proposed modeling framework extends most existing approaches mainly in two ways. First, it allows to analyze functional data with a broad range of complex correlation structures. And second, it applies to data observed on very general sampling grids. The simultaneous extension to data with a complex correlation structure sampled on curve-specific grids with possibly few observation points has so far been scarcely addressed. By combining these two aspects, this thesis aims to provide a flexible and widely applicable framework for the analysis of functional data.

The considered models may include different kinds of functional fixed effects as well as functional random effects, which are all expanded in basis functions. This accounts for the functional nature of the data and linearizes the estimation problem. Most central to this thesis is the prediction of the functional random effects that capture the between- and within-function correlation. The functional random effects are expanded in bases of eigenfunctions of their respective covariance operators which have to be estimated beforehand.

The estimation of the covariances is thus a crucial step in the analysis. It is, however, challenging to estimate covariances of latent processes for correlated functions with complex correlation structures, in particular when the data are observed on unequal grids or even sparsely. In this thesis, two methods of moments estimators are proposed to obtain smooth covariance estimates of latent processes in very general models.

The first estimator applies to data sampled on a common grid. It makes use of the grid structure to obtain point-wise raw covariance estimates and to increase computational efficiency. The raw estimates can then be smoothed separately using any bivariate smoother (Chapter 3).

The second estimator applies to general sampling grids, where point-wise estimates may not be feasible. The proposed method of moments approach is represented as an additive bivariate varying coefficient model for the sample covariance of the centered data. It is particularly well-suited for irregularly and sparsely sampled data as strength is borrowed across all curves (Chapter 4).

Both covariance estimation approaches involve bivariate smoothing of possibly multiple (in the second approach even additive) covariances. Even for sparsely sampled data, this quickly becomes a computationally challenging task as the number of elements that enter the estimation increase quadratically with the number of grid points. To overcome this computational bottleneck, a fast symmetric smoothing approach is proposed in this thesis (Chapter 5) that is particularly suited to estimate smooth additive covariances as required in Chapter 4. It takes advantage of the symmetry of covariances, leading to a considerably faster estimation and requiring less memory. The symmetric smoother can be applied in very general bivariate symmetric smoothing problems, including the two covariance estimation approaches proposed in Chapter 3 and Chapter 4 of this thesis.

Once the smooth estimated covariances are available, they are evaluated on an equidistant grid, which reduces the eigen problem for the covariance operators to the corresponding matrix eigen decompositions. Replacing the functional random effects by the truncated FPC expansions then allows to approximate the functional linear mixed models by scalar linear mixed models with random effects corresponding to the random FPC weights, which can be predicted as empirical best linear unbiased predictors (Di et al., 2009; Greven et al., 2010).

In summary, Chapter 3 considers very general correlation structures while assuming a common sampling grid. In contrast, Chapter 4 allows for general sampling grids with a focus on less general correlation structures. Finally, Chapter 5 draws the connection by extending the approach in Chapter 4 to very general correlation structures. It thus covers both very general correlation structures as well as general sampling grids while reducing computation time. The smoother can also be used to smooth the raw covariance estimates in Chapter 3. Despite the close connection,

each chapter is self-contained apart from some cross-references and can be read separately. In more detail, the thesis is organized as follows:

Chapter 2 gives an introduction to scalar linear mixed models and multivariate principal component analysis, followed by an outline of the extension of both methods to the functional framework. In this context, the general class of functional linear mixed models is introduced, which comprises all models considered in this thesis.

Chapter 3 considers the general class of functional linear mixed models and proposes a computationally efficient estimation approach for data sampled on a common grid that explicitly takes advantage of the grid structure. The approach is applied to study tissue spectroscopy data that have a hierarchical structure inducing correlation. The correlation is accounted for by hierarchical functional random intercepts that are allowed to have tissue type specific covariance operators. The aim of the application is to train a classification algorithm that uses reflectance spectra to reliably classify the tissue type during surgery. To achieve this, the proposed modeling framework is combined with the functional classification approach of Zhu et al. (2012). A comparison with other classification methods shows that class designations are improved using the proposed FPC-based estimation approach which accounts for all different sources of variation in the data.

Chapter 4 proposes an approach to model correlated functional data that may be sampled on curve-specific grids with possibly few observation points. General correlation structures are allowed in the model. The focus in the presentation of the estimation approach is, however, on a model with crossed functional random intercepts, which is relevant for the motivating application to data from a speech production study. A new combination of the FPC estimation with the framework of functional additive mixed models introduced by Scheipl et al. (2015) is proposed, which allows for approximate statistical inference conditional on the FPCA. Extensive simulation studies are performed to compare FPC bases and spline bases, which can also be considered in the framework of Scheipl et al. (2015). They lead to the conclusion that FPC bases have a clear advantage both in terms of computation times and estimation quality.

Chapter 5 proposes a fast symmetric bivariate smoothing approach that is widely applicable, in particular to speed up the FPC-based estimation of functional linear mixed models with very general correlation structures. The approach can handle possibly noisy data observed on general grids. It extends the covariance estimation in Chapter 4 by accounting for the symmetry of the covariances and by allowing for more general correlation structures. Covariance estimation for longitudinal data as well as for functional data with general correlation structures is discussed. Many existing covariance estimation approaches, including those proposed in Chapters 3 and 4, involve a quadratic loss function which implies working assumptions, such as working independence, that do not hold. For the estimation approach in Chapter 4, this chapter develops a remedy in form of an iterative estimation algorithm that may find application in simple data settings with low

computational burden. The practical relevance of the fast symmetric smoother is demonstrated in applications to longitudinal data from a medical study and to the speech production data with a crossed correlation structure, which are also considered in Chapter 4.

In addition to the development of the theoretical estimation framework, fully documented open-source software is provided as part of this thesis in order to make the proposed methods readily accessible to users. An implementation of the estimation approach for densely sampled functional data proposed in Chapter 3 is made available in the R add-on package `denseFLMM` (R Core Team, 2016; Greven and Cederbaum, 2017). Implementations of the approaches described in Chapters 4 and 5 are provided in the R add-on package `sparseFLMM` (Cederbaum, 2016). A description and examples for the usage of both R packages can be found in Appendix E.

This thesis closes with a concluding summary of the contained chapters and an outlook to future research in **Chapter 6**.

Underlying assumptions

Regarding the different perspectives and the numerous assumptions that can be made in connection with functional data as discussed in Section 1.1 and Section 1.2, it seems essential to set the scene. Throughout this thesis, the focus is on a frequentist view on mean regression models with functional responses which are one-dimensional functions living in the L^2 -space defined over a bounded interval $\mathcal{T} \subset \mathbb{R}$ with the inner product as defined in Section 1.1. It is implicitly assumed that the inherent smoothness is a sensible choice for the observed data and that the degree of smoothness is roughly the same over the whole domain. Moreover, the random number and location of measurement points for curve-specific sampling grids and potential missings for equal grids are assumed to be non-informative. All considered models have an additive (and in most parts linear) predictor and contain additive random measurement error. Note that the linearity is with respect to the covariates and not to the function argument. The additive predictor consists of functional fixed effects as well as functional random effects. The latter are assumed to be independent copies of Gaussian processes in most parts of this thesis. The Gaussian assumption can be relaxed at some points, while it is assumed throughout that the mean and the covariance can be modeled separately. Their covariances are assumed to be smooth. The measurement error is assumed to only capture random uncorrelated variation. Hence, conditional on the additive predictor, the observed response values are uncorrelated both within and between functions. In most parts, the error variance is assumed to be homoscedastic.

Language conventions

As in every other field, technical terms are used in FDA that often allow an efficient, precise manner of expression. However, they can sometimes be misleading, such as for instance the term ‘vector’, which is often used in FDA to differentiate from functions, although the considered functions are obviously also vectors in their respective vector space. To avoid confusion, it should be noted that in

this thesis the term is used to describe finite-dimensional vectors. Moreover, as is frequently the case in FDA, the covariance operator and its kernel are not always clearly distinguished. For clarification, Chapter 2 briefly introduces the two notions.

Also within the same area of research, some terms have different meanings. An important example for this thesis is in FPCA, where the main modes of variation are sometimes also termed ‘weights’ or ‘weight functions’ (see, e.g., Ramsay and Silverman, 2005) as opposed to this thesis, in which the term weights refers to the projections.

Following common mixed model terminology, in this thesis, the term ‘prediction’ is used rather than ‘estimation’ when referring to a random parameter or function.

1.5 Contributing manuscripts

Parts of this thesis have already been published in peer reviewed journals, in conference proceedings, or in manuals accompanying the R add-on packages `denseFLMM` and `sparseFLMM`. The remaining parts are based on submitted or uncompleted manuscripts. All manuscripts were written in cooperation with my supervisor Sonja Greven and with co-authors from statistics and other fields. Below, all relevant manuscripts are listed chapter by chapter. Information on the individual contributions of all authors is given at the beginning of each chapter.

Chapter 2 on the extension of linear mixed models and principal component analysis to functional data was specifically prepared for this thesis but references at various points to the contents of Cederbaum et al. (2016), Cederbaum et al. (2018), and Greven et al. (2016).

Chapter 3 on general functional linear mixed models for data sampled on equal sampling grids is based on the working paper in preparation

Greven, S., Cederbaum, J., and Shou, H. (2016): Principal component-based functional linear mixed models. *Working paper*.

Chapter 4 on functional linear mixed models for data sampled on unequal and sparse sampling grids is based on

Cederbaum, J., Pouplier, M., Hoole, P., and Greven, S. (2016): Functional linear mixed models for irregularly or sparsely sampled data. *Statistical Modelling*, 16(1):67–88.

Preliminary work on Chapter 4 can be found in the conference proceedings of IWSM 2014–29th International Workshop on Statistical Modelling (Cederbaum et al., 2014) and in the conference proceedings of ISSP 2014–10th International Seminar on Speech Production (Pouplier et al., 2014).

Chapter 5 on fast symmetric bivariate smoothing is based on

Cederbaum, J., Scheipl, F., and Greven, S. (2018): Fast symmetric additive covariance smoothing. *Computational Statistics & Data Analysis*, 120:25–41.

Chapter 6, which provides an overall summary and outlook, was specifically prepared for this thesis.

In a cooperation with project partners from phonetics, the approaches proposed in Chapters 4 and 5 were applied to different data types with the aim to answer current scientific questions in phonetic research and to introduce the method to the phonetic society. The joint work can be found in

Poupplier, M., Cederbaum, J., Hoole, P., Marin, S., and Greven, S. (2017): Mixed modeling for irregularly sampled and correlated functional data: Speech science applications. *Journal of the Acoustical Society of America*, 142(2):935–946.

The contributing manuscripts are cited at the beginning of each chapter. For better readability of this thesis, they are not repeatedly referenced within each chapter despite the textual matches.

1.6 Software

All analyses in the context of this thesis were carried out in the R system of statistical computing (R Core Team, 2016) on two different platforms (x86_64-pc-linux-gnu (64-bit) and x86_64-w64-mingw32/x64 (64-bit)).

Comprehensive implementations of all developed methods are made available. The implementations for Chapter 3 are provided in the R add-on package `denseFLMM` (Greven and Cederbaum, 2017). The functions for Chapters 4 and 5 are provided in the R add-on package `sparseFLMM` (Cederbaum, 2016). All software employed in this thesis is open-source and therefore free to be used by anyone.

Estimation of the mean and covariance functions was performed using the R add-on package `mcgv` (Wood, 2006, 2011), for which a new smoothing class is developed in Chapter 5. For the combination of the proposed FPC estimations with the framework of functional additive mixed models (Scheipl et al., 2015), the R add-on package `refundDevel` (Huang et al., 2016b) was used, which was also employed for comparison with the competing spline-based approach in Chapter 4. For comparison with a competing symmetric smoothing approach in Chapter 5, the R add-on package `face` was used. To read the speech production data that were available in MATLAB format (MATLAB, 2013), the one-directional interface provided in the R add-on package `R.matlab` (Bengtsson, 2016) was employed. For the graphical visualizations, the R add-on packages `ggplot2` (Wickham, 2009), `nlme` (Pinheiro et al., 2016), and `lattice` (Sarkar, 2008) were used. Additional information on the software used, including R and R package versions, is given at the beginning of the corresponding chapters.

Chapter 2

Extending Linear Mixed Models and Principal Component Analysis to Functional Data

This chapter provides an introduction to the two main topics this thesis is based on, linear mixed models and principal component analysis. In the first part of this chapter (Section 2.1), linear mixed models for scalar correlated data are briefly introduced, followed by an outline of the extension to the functional framework. A very general class of functional linear mixed models is presented, in which all models discussed in Chapter 3 to Chapter 5 can be embedded. In the second part of this chapter (Section 2.2), multivariate principal component analysis is introduced and the transition to its functional counterpart is described.

2.1 Linear mixed models

This section gives a motivation of linear mixed models and briefly summarizes different aspects and views. The general scalar linear mixed model is introduced and model assumptions are briefly explained. Given the importance of scalar linear mixed models in this thesis, this section provides an outline of the estimation and prediction of the model parameters. The content of the first part of this section (Section 2.1.1) is in a large part based on Chapter 7 in Fahrmeir et al. (2013) and Chapter 2 in Greven (2007). In the second part (Section 2.1.2), the extension of scalar linear mixed models to their functional counterpart is described. A general functional linear mixed model is introduced that is described in more detail in Chapter 3. It is outlined how all models considered in this thesis can be embedded in this general framework. The model assumptions and identifiability issues are discussed and the main estimation steps, which the approaches proposed in Chapter 3 to Chapter 5 have in common, are briefly summarized.

2.1.1 Scalar linear mixed models

Linear mixed models (LMMs; see, e.g., Pinheiro and Bates, 2000) are a powerful tool for the analysis of data with multiple sources of variation, such as longitudinal or clustered data, or data with a spatial structure. Unlike most regression models, which are based on the assumption that the data are independent and identically distributed (i. i. d.), LMMs are a flexible approach to quantify and account for the correlation induced by different sources of variation, which is important for valid statistical inference. The name ‘mixed models’ derives from the fact that besides the usual fixed effects (in frequentist models), also random effects are included. In fact, linear mixed models are sometimes also called ‘random effects models’ (e.g., Laird and Ware, 1982). There are different motivations to use random effects, leading to different perspectives of LMMs (for a good overview, see Fahrmeir et al., 2013, Chapter 7). One idea is to view the random effects as surrogate effects for omitted or insufficiently measured covariates that induce correlation. The random effects account for the unobserved heterogeneity and capture the induced correlation by allowing deviations from the population mean. From a different perspective, the random effects can also be regarded as additional error terms. The fact that the effects are assumed to be random, i.e., to follow a (usually Gaussian) distribution, is commonly considered to reflect that the grouping levels (e.g., individuals or clusters) are a random sample from a larger population. The distributional assumption can, however, also be seen as a regularization which stabilizes the estimation. This regularization property allows to take advantage of the inferential methods for LMMs when choosing the smoothing parameter in penalized spline regression with quadratic penalties (for more details, see, e.g., Brumback et al., 1999; Ruppert et al., 2003; Fahrmeir et al., 2013).

The role of LMMs in this thesis is threefold. First, LMMs are the foundation of functional LMMs, which are at the heart of this thesis and explained in more detail below (see Section 2.1.2). Second, this thesis makes use of the fact that penalized splines can be represented as predictors in LMMs, allowing to take advantage of LMM methodology for the smoothing parameter estimation (see, e.g., Ruppert et al., 2003, Chapter 4). Third, the functional LMMs are approximated by LMMs in order to predict the functional principal component weights as the random effects in the LMMs (see also Di et al., 2009; Greven et al., 2010; Di et al., 2014). Due to the importance of LMMs for this thesis, not only a general model formulation will be introduced, but the model estimation will also be outlined in the following.

General linear mixed model

In its general form, a linear mixed model can be defined as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \boldsymbol{\varepsilon}, \quad (2.1)$$

where $\mathbf{y} = (y_1, \dots, y_n)^\top$ is a vector of n observable random response variables, \mathbf{X} and \mathbf{Z} are known $n \times p$ and $n \times q$ design matrices, corresponding to the $p \times 1$ and $q \times 1$ vectors $\boldsymbol{\beta}$ and \mathbf{u} of fixed and random effects, respectively. The structure for the random effects can be quite general, including

hierarchical, crossed, and potentially correlated random effects. The additive vector $\boldsymbol{\varepsilon}$ contains unobservable random errors.

Model assumptions

It is assumed that \mathbf{u} and $\boldsymbol{\varepsilon}$ are independent and both have zero mean as the population mean is represented by the fixed effects. Instead of acting on the mean, the random effects induce a correlation structure. A common assumption, which is useful for inference, is that \mathbf{u} and $\boldsymbol{\varepsilon}$ follow the multivariate Gaussian distribution

$$\begin{pmatrix} \mathbf{u} \\ \boldsymbol{\varepsilon} \end{pmatrix} \sim \mathcal{N} \left[\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \begin{pmatrix} \mathbf{K} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma} \end{pmatrix} \right], \quad (2.2)$$

with positive semi-definite variance-covariance matrices \mathbf{K} and $\boldsymbol{\Sigma}$, respectively. The Gaussian assumption is, however, not necessary for all inferential conclusions. Strictly speaking, the assumptions in (2.2) need to be complemented by a covariate exogeneity assumption as in the linear model, i.e., $\mathbb{E}(\boldsymbol{\varepsilon}|\mathbf{X}\mathbf{u}) = \mathbf{0}$, and by the ‘random effects assumption’ $\mathbb{E}(\mathbf{u}|\mathbf{X}) = \mathbf{0}$, which ensures that $\mathbb{E}(\mathbf{y}|\mathbf{X}) = \mathbf{X}\boldsymbol{\beta}$.

The randomness of \mathbf{u} results in two different formulations of Model (2.1); a marginal formulation and a formulation conditional on the random effects. With the assumptions in (2.2), the marginal formulation yields the following response distribution

$$\mathbf{y} \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \mathbf{V}),$$

with $\mathbf{V} = \mathbf{Z}\mathbf{K}\mathbf{Z}^\top + \boldsymbol{\Sigma}$. From a conditional or ‘hierarchical’ perspective and with the assumptions in (2.2), the conditional response distribution is given by

$$\mathbf{y}|\mathbf{u} \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u}, \boldsymbol{\Sigma}).$$

Although the marginal model formulation is implied by the conditional formulation, this does not apply in reverse and thus the two formulations are not equivalent. In contrast to the marginal view on LMMs, the conditional view additionally allows for individual predictions. The interpretation of the fixed effects, however, is the same for both formulations. Note that this is generally not the case for generalized linear mixed models, i.e., mixed models for more general response distributions; for extensions to generalized linear mixed models, see, e.g., Molenberghs and Verbeke (2005) and McCulloch et al. (2008).

Estimation and prediction

Several approaches to estimating the fixed effects and to predicting the random effects result in the same solutions. Let $\boldsymbol{\vartheta}$ denote the vector of all variance parameters in the matrices \mathbf{K} and $\boldsymbol{\Sigma}$ and assume that it is known. One way to simultaneously estimate the fixed effects and to predict the

random effects, is to maximize the joint log-likelihood of \mathbf{y} and \mathbf{u} with respect to $\boldsymbol{\beta}$ and \mathbf{u} , which is given by (up to an additive constant) (Henderson, 1950)

$$\ell(\boldsymbol{\beta}, \mathbf{u}) = -\frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\mathbf{u})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\mathbf{u}) - \frac{1}{2}\mathbf{u}^\top \mathbf{K}^{-1}\mathbf{u}. \quad (2.3)$$

Maximizing the joint likelihood (2.3) is equivalent to minimizing the penalized least squares criterion

$$\text{ls}_{\text{pen}}(\boldsymbol{\beta}, \mathbf{u}) = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\mathbf{u})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\mathbf{u}) + \mathbf{u}^\top \mathbf{K}^{-1}\mathbf{u}, \quad (2.4)$$

which is given by the sum of the weighted least squares criterion $(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\mathbf{u})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\mathbf{u})$ and the penalty term $\mathbf{u}^\top \mathbf{K}^{-1}\mathbf{u}$. The penalty term accounts for the fact that the random effects \mathbf{u} follow a distribution and shrinks the random effects towards their mean, i.e., towards zero compared to when they were estimated as fixed effects. This shrinkage effect becomes stronger when \mathbf{K} approaches the zero matrix, with $\hat{\mathbf{u}} = \mathbf{0}$ in the extreme. Conversely, when \mathbf{K}^{-1} approaches the zero matrix, the random effects are estimated as fixed effects.

The optimization of (2.3) or rather (2.4) results in the following weighted least squares or Aitken's estimator (Aitken, 1936) for the fixed effects

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top \mathbf{V}^{-1} \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{V}^{-1} \mathbf{y}, \quad (2.5)$$

which is the best linear unbiased estimator (BLUE) in the sense of the generalized Gauss-Markov theorem and the best unbiased estimator under the Gaussian assumption (see Zyskind and Martin, 1969; Harville, 1976). The resulting best linear unbiased predictor (BLUP) for the random effects is given by

$$\hat{\mathbf{u}} = \mathbf{K} \mathbf{Z}^\top \mathbf{V}^{-1} \left(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right), \quad (2.6)$$

which, under the Gaussian assumption is also the best unbiased predictor (BUP; Harville, 1976). Note that the B(L)UP $\hat{\mathbf{u}}$ is,

- linear with respect to \mathbf{y}
- unbiased referring to $\mathbb{E}(\hat{\mathbf{u}}) = \mathbb{E}(\mathbf{u}) = \mathbf{0}$
- the best in the sense that it minimizes the mean squared error $\mathbb{E}\left[(\tilde{\mathbf{u}} - \mathbf{u})^\top (\tilde{\mathbf{u}} - \mathbf{u})\right]$ among all (linear) unbiased predictors $\tilde{\mathbf{u}}$ for \mathbf{u} .

Commonly, the stacked vector $\left(\hat{\boldsymbol{\beta}}^\top, \hat{\mathbf{u}}^\top\right)^\top$ is also referred to as BLUP.

As in practice the variance parameters $\boldsymbol{\vartheta}$ and thus \mathbf{K} , $\boldsymbol{\Sigma}$, and \mathbf{V} are unknown, they also have to be estimated. This is commonly done using maximum likelihood (ML) or restricted maximum likelihood (REML) estimation. Both ML and REML estimation are based on the marginal model formulation. To emphasize the dependence on the (unknown) $\boldsymbol{\vartheta}$, the covariance matrices, the estimators for the fixed effects, and the predictors for the random effects are in the following denoted

as $\mathbf{V}(\boldsymbol{\vartheta})$ and so forth. Let $\det[\mathbf{V}(\boldsymbol{\vartheta})]$ denote the determinant of $\mathbf{V}(\boldsymbol{\vartheta})$. The log-likelihood derived from the marginal model formulation is then given by (up to an additive constant)

$$\ell(\boldsymbol{\beta}, \boldsymbol{\vartheta}) = -\frac{1}{2} \left[\log \{ \det [\mathbf{V}(\boldsymbol{\vartheta})] \} + (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{V}(\boldsymbol{\vartheta})^{-1} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \right]. \quad (2.7)$$

Under the Gaussian assumption, the ML estimate of $\boldsymbol{\vartheta}$ is obtained by maximizing the profile log-likelihood $\ell_P(\boldsymbol{\vartheta})$ with respect to $\boldsymbol{\vartheta}$, which is given by substituting the ML estimate of the fixed effects $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}(\boldsymbol{\vartheta})$ as defined in (2.5) in the log-likelihood (2.7) from the marginal model formulation. The ML estimator for $\boldsymbol{\vartheta}$ can be shown to be biased downwards as it does not account for the loss in the degrees of freedom induced by substituting the fixed effects $\boldsymbol{\beta}$ by its estimate $\hat{\boldsymbol{\beta}}(\boldsymbol{\vartheta})$ in its estimation (e.g., Patterson and Thompson, 1971). This is analogous to the bias of the ML estimator for the error variance in the linear model.

As a remedy, Patterson and Thompson (1971) propose a ‘modified maximum likelihood method’ that was later termed ‘restricted’ maximum likelihood approach (Harville, 1977). Instead of maximizing the likelihood of all data, they divide the data into two statistically independent parts and obtain separate likelihoods for the estimation of the fixed effects and of the variance parameters. The likelihood of all data is given as the product of the two likelihoods. For the estimation of the variance parameters, they derive the restricted log-likelihood $\ell_R(\boldsymbol{\vartheta})$ as log-likelihood of a set of $n - p$ linearly independent error contrasts $\mathbf{A}\mathbf{y}$ of which the distribution is independent of the fixed effects. Harville (1974) later shows that any contrast matrix \mathbf{A} fulfilling $\mathbf{A} \neq \mathbf{0}$ and $\mathbb{E}(\mathbf{A}\mathbf{y}) = \mathbf{0}$ can be used as the resulting likelihoods are proportional. The restricted log-likelihood can be written as

$$\ell_R(\boldsymbol{\vartheta}) = \ell_P(\boldsymbol{\vartheta}) - \frac{1}{2} \log \left\{ \det \left[\mathbf{X}^\top \mathbf{V}(\boldsymbol{\vartheta})^{-1} \mathbf{X} \right] \right\}.$$

The log-likelihood $\ell_R(\boldsymbol{\vartheta})$ can also be motivated from an empirical Bayesian perspective. Using the likelihood of error contrasts yields an estimator for $\boldsymbol{\vartheta}$ which is equivalent to the mode of the posterior when using a flat prior for the fixed effects $\boldsymbol{\beta}$ and all the data (Harville, 1974). This allows the alternative derivation of the REML estimator as maximum of the marginal log-likelihood for the variance parameters $\boldsymbol{\vartheta}$, which is obtained by integrating out the fixed effects from the likelihood $L(\boldsymbol{\beta}, \boldsymbol{\vartheta})$, i.e.,

$$\ell_R(\boldsymbol{\vartheta}) = \log \left[\int L(\boldsymbol{\beta}, \boldsymbol{\vartheta}) \, d\boldsymbol{\beta} \right].$$

In linear models for uncorrelated data, the REML estimator for the error variance removes the bias of the ML estimator. This generally does not apply to the REML estimator for the variance parameters $\boldsymbol{\vartheta}$ in linear mixed models, which reduces the bias compared to the ML estimator but generally does not fully remove it (see, e.g., Fahrmeir et al., 2013, Chapter 7).

As in general, for both ML and REML estimation, no closed-form expression of the estimator $\hat{\boldsymbol{\vartheta}}$ exists, the maximization of the log-likelihoods $\ell_P(\boldsymbol{\vartheta})$ and $\ell_R(\boldsymbol{\vartheta})$ is done numerically through iterative algorithms, such as Newton-Raphson or Fisher scoring.

Once the variance parameters $\boldsymbol{\vartheta}$ are estimated either using ML or REML, they are plugged-in to obtain covariance estimates $\widehat{\mathbf{K}}(\widehat{\boldsymbol{\vartheta}})$, $\widehat{\boldsymbol{\Sigma}}(\widehat{\boldsymbol{\vartheta}})$, and hence $\widehat{\mathbf{V}}(\widehat{\boldsymbol{\vartheta}})$. Replacing \mathbf{V} in Equations (2.5) and (2.6) by its estimate finally yields the estimators for the fixed effects $\widehat{\boldsymbol{\beta}}(\widehat{\boldsymbol{\vartheta}})$ and predictors for the random effects $\widehat{\mathbf{u}}(\widehat{\boldsymbol{\vartheta}})$. To emphasize that variability is induced by the estimation of the variance parameters, $\widehat{\boldsymbol{\beta}}(\widehat{\boldsymbol{\vartheta}})$ and $\widehat{\mathbf{u}}(\widehat{\boldsymbol{\vartheta}})$ are commonly referred to as empirical BLUE and empirical BLUP, respectively.

For more details on the inference in LMMs, including the construction of confidence bands, testing on fixed and random effects, and model selection, it is referred to, e.g., Stram and Lee (1994), Crainiceanu and Ruppert (2004), Greven (2007), Greven et al. (2008), and Greven and Kneib (2010).

2.1.2 From scalar to functional linear mixed models

Functional linear mixed models (FLMMs; for an overview, see, e.g., Morris, 2015) can be seen as functional counterparts to scalar linear mixed models. Correspondingly, FLMMs are used to analyze functional data with multiple sources of variation. As in the scalar case, correlation can be induced by, e.g., repeated observations per subject (longitudinal functional data), grouping in the data, or spatial sampling. These sources of correlation are additional to the correlation of adjacent values on the domain of the functions.

As explained in Chapter 1, Section 1.2, it is commonly distinguished between three types of functional regression models, namely function-on-scalar, scalar-on-function, and function-on-function regression. The term FLMM commonly refers to function-on-scalar regression models that include both fixed and random effects. In some models, additional effects of functional covariates are included, yielding a function-on-function regression model (see, e.g., Scheipl et al., 2015), but they are not in the main focus of this thesis. In FLMMs, the random effects are functions varying over the same domain as the functional responses. To emphasize their functional nature, they are commonly referred to as ‘functional random effects’ (fREs). fREs allow very flexible deviations from the mean function that are represented as latent smooth functions. In analogy to LMMs, the fREs can be thought of as surrogate effects that account for unobserved heterogeneity in the data. The assumption in LMMs that the surrogate effects are random can be translated to the assumption that the functional surrogate effects are random (or stochastic) processes. In the following, a very general FLMM is introduced, which is proposed in Chapter 3. An explanation of how all other models considered in this thesis fall into this model class is given at the end of this section.

General functional linear mixed model

A very general FLMM for one-dimensional, square integrable functional responses on a bounded interval $\mathcal{T} \in \mathbb{R}$ can be defined as

$$Y_i(t_{ij}) = \mu(t_{ij}, \mathbf{x}_i) + \mathbf{z}_i^\top \mathbf{U}(t_{ij}) + \varepsilon_i(t_{ij}), \quad i = 1, \dots, n, \quad t_{ij} \in \mathcal{T}, \quad (2.8)$$

where $Y_i(t_{ij})$ denotes the response curve i at observation point t_{ij} . $\mu(t_{ij}, \mathbf{x}_i)$ is a smooth mean function depending on a vector of p known scalar covariates \mathbf{x}_i that replaces the population mean $\mathbf{X}\boldsymbol{\beta}$ in the

scalar LMM (2.1). The mean function can be written as $\mu(t_{ij}, \mathbf{x}_i) = f_0(t_{ij}) + \sum_{k=1}^r f_k(t_{ij}, \tilde{\mathbf{x}}_{ik})$, where $f_0(t_{ij})$ is a functional intercept and $\tilde{\mathbf{x}}_{ik}$ is a subvector of the vector of covariates \mathbf{x}_i . The subvector $\tilde{\mathbf{x}}_{ik}$ may consist of a single scalar covariate but can also include several covariates allowing for interaction effects. The deviations from the population mean in Model (2.1), $\mathbf{Z}\mathbf{u}$, are replaced by $\mathbf{z}_i^\top \mathbf{U}(t_{ij})$, where \mathbf{z}_i is a vector of q known scalar covariates and $\mathbf{U}(t_{ij})$ is a q -dimensional vector of functional random effects. Similar to the scalar LMM (2.1), a general structure for the fREs is allowed, including hierarchical and crossed fREs, and the fREs may also be correlated. Commonly, the fREs include a curve-specific deviation in form of a smooth error curve that captures deviations from the mean function which are correlated along \mathcal{T} . Often, the smooth error is written separately from the vector of functional random effects $\mathbf{U}(t_{ij})$ (compare Chapters 4, 5), but Model (2.8) is more general allowing for group-specific smooth error terms that can be included in $\mathbf{U}(t_{ij})$ (for more details, see Chapter 3). The additional random error $\varepsilon_i(t_{ij})$ accounts for uncorrelated variation within each curve i .

Possible structures for the mean function $\mu(t_{ij}, \mathbf{x}_i)$ and examples for $\mathbf{z}_i^\top \mathbf{U}(t_{ij})$ yielding FLMMs with, e.g., hierarchical and crossed functional random effects will be given at the end of this section. Note that when all functions are observed on a common grid, the notation can be simplified by omitting the indices of t_{ij} and it is possible to formulate a discretized version of Model (2.8) using matrix notation similar to Model (2.1) for all curves, $i = 1, \dots, n$, rather than on curve-level (see Chapter 3).

Model assumptions

In analogy to the scalar random effects, $\mathbf{U}(\cdot)$ is assumed to be a vector-valued square integrable random process on \mathcal{T} with zero mean. The common Gaussian assumption for scalar random effects is replaced by a Gaussian process assumption, which is, however, not necessary for all inferential conclusions. The random error $\varepsilon_i(\cdot)$ is commonly assumed to be i.i.d. white noise measurement error with constant variance σ^2 . As in the scalar case, it is assumed that the fREs $\mathbf{U}(t)$ and the measurement error $\varepsilon_i(t')$ are uncorrelated for all i and for all $t, t' \in \mathcal{T}$. Again, the population mean is represented by the fixed effects, which are part of the mean function $\mu(t, \mathbf{x}_i)$, and the fREs induce a correlation structure. The vector-valued random process $\mathbf{U}(\cdot)$ has a matrix-valued covariance, in the following denoted by $\mathbf{K}^U(t, t') = \text{Cov}[\mathbf{U}(t), \mathbf{U}(t')]$, with $t, t' \in \mathcal{T}$. Given the additive predictor that commonly includes a smooth error curve capturing correlation along \mathcal{T} , the observed response values are assumed to be independent within and across functions.

To give a clearer idea of the vector-valued fRE and its covariance, a more detailed description is provided in the following. Assume that the correlation in the FLMM (2.8) is induced by G different grouping factors with L^{U_g} levels, $g = 1, \dots, G$. To give an example for two grouping factors ($G = 2$), consider four students in two schools. Then, $L^{U_1} = 2$ (number of schools) and $L^{U_2} = 4$ (number of students across schools). Further, denote the number of random effects associated with grouping factor g by ρ^{U_g} , $g = 1, \dots, G$. In the above example, setting $\rho^{U_1} = 1$ models a functional random intercept (fRI) for each school and $\rho^{U_2} = 2$ models an fRI and a functional random slope (fRS) for each student. Note that the fREs for the same level of a grouping factor are assumed to be correlated. Uncorrelated functional random effects would be assumed by assigning different grouping factors, in

the example by letting $G = 3$ with $L^{U_1} = 2$, $L^{U_2} = L^{U_3} = 4$ and $\rho^{U_1} = \rho^{U_2} = \rho^{U_3} = 1$ for an uncorrelated fRI and fRS for each student.

The vector of fREs $\mathbf{U}(t_{ij})$ can be divided into G independent blocks, one for each grouping factor, i.e., $\mathbf{U}(t_{ij}) = \left[\mathbf{U}_1(t_{ij})^\top, \dots, \mathbf{U}_G(t_{ij})^\top \right]^\top$. Each block $\mathbf{U}_g(t_{ij})$ again consists of L^{U_g} independent copies (in the sense of random processes) $\mathbf{U}_{gl}(t_{ij})$, $l = 1, \dots, L^{U_g}$, corresponding to the levels of the respective grouping factor. Finally, the independent copies are vector-valued random processes consisting of ρ^{U_g} fREs $U_{gls}(t_{ij})$, $s = 1, \dots, \rho^{U_g}$. The total number of fREs is given by $q = \sum_{g=1}^G L^{U_g} \rho^{U_g}$.

In summary, it is assumed that blocks of fREs of different grouping factors $\mathbf{U}_g(\cdot)$, $\mathbf{U}_h(\cdot)$, $g \neq h$, and blocks of fREs of different levels of the same grouping factor $\mathbf{U}_{gl}(\cdot)$, $\mathbf{U}_{gk}(\cdot)$, $l \neq k$, are independent. Blocks of different fREs of the same level of a grouping factor $U_{gls}(\cdot)$, $U_{gls'}(\cdot)$, $s \neq s'$, are assumed to be correlated. Thus, in the above example of students within schools, the fREs of schools are independent of the fREs of students. Moreover, the fREs of different schools are independent and the fREs of different students are independent. The fREs of the same schools are assumed to be correlated as are the fREs of the same students.

Consequently, the covariance of the vector-valued random process $\mathbf{U}(\cdot)$ is a $q \times q$ diagonal block matrix of the form

$$\mathbf{K}^U(t, t') = \text{diag} \left[\underbrace{\mathbf{K}^{U_1}(t, t'), \dots, \mathbf{K}^{U_1}(t, t')}_{L^{U_1} \text{ times}}, \dots, \underbrace{\mathbf{K}^{U_G}(t, t'), \dots, \mathbf{K}^{U_G}(t, t')}_{L^{U_G} \text{ times}} \right],$$

where $\mathbf{K}^{U_g}(t, t') = \text{Cov} [\mathbf{U}_{gl}(t), \mathbf{U}_{gl}(t')]$ is the matrix-valued covariance of $\mathbf{U}_{gl}(\cdot)$, which is equal for all levels, $l = 1, \dots, L^{U_g}$, as the $\mathbf{U}_{gl}(\cdot)$ are copies of the same random process. Note that fREs with group-specific covariances as in Chapter 3 can be included in the framework by considering a separate grouping factor for each group rather than different levels of a single grouping factor. That is, if the schools in the above example are, e.g., in different countries and one wants to model an fRI for each school with a country-specific covariance, a separate grouping factor for each country is specified rather than one single grouping factor for all schools.

$\mathbf{K}^{U_g}(t, t') = \left[K_{ss'}^{U_g}(t, t') \right]_{s, s'=1, \dots, \rho^{U_g}}$ is a $\rho^{U_g} \times \rho^{U_g}$ matrix with auto-covariances $K_{ss}^{U_g}(t, t') = \text{Cov} [U_{gls}(t), U_{gls}(t')]$, $s = 1, \dots, \rho^{U_g}$, on the diagonal and cross-covariances $K_{ss'}^{U_g}(t, t') = \text{Cov} [U_{gls}(t), U_{gls'}(t')]$, $s \neq s'$, as off-diagonal elements. Whereas an auto-covariance gives the covariance of the function with itself at pairs of points (t, t') , the cross-covariance gives the covariance of a function at point t with another function at point t' . Returning to the example above, the cross-covariance $\text{Cov} [U_{2l1}(t), U_{2l2}(t')]$ gives the covariance of the fRI at point t with the fRS at point t' for each student. Note that in general, $\mathbf{K}^{U_g}(t, t')$ is not symmetric as the cross-covariances $K_{ss'}^{U_g}(t, t')$ and $K_{s's}^{U_g}(t', t)$ are generally not equal, but it clearly holds that $K_{ss'}^{U_g}(t, t') = K_{s's}^{U_g}(t', t)$ and thus $\mathbf{K}^{U_g}(t, t') = \mathbf{K}^{U_g}(t', t)^\top$. All auto- and cross-covariances are assumed to be smooth in t and t' .

As in the scalar case, the FLMM (2.8) can be viewed from a marginal as well as from a conditional perspective, which are again not equivalent but yield the same interpretation for the fixed effects,

i.e., for $\mu(t_{ij}, \mathbf{x}_i)$. The estimation of the model parameters is usually based on the marginal model formulation.

Estimation and prediction

As briefly discussed in Chapter 1, different approaches to estimation in function-on-scalar regression models and in particular in FLMMs have been proposed. In this thesis, the observed response values are modeled directly while the functional nature of the data is accounted for in the additive predictor. The smoothness assumption along t is accounted for by expanding all terms in the predictor in basis functions. This has several advantages (see Chapter 1, Section 1.2) and in particular reduces the functional models to models for scalar data. For the FLMM (2.8), this means that it can be reduced to its scalar counterpart (2.1) and inference can be based on scalar LMMs as described in Section 2.1.1.

In this thesis, the fREs are expanded in functional principal component (FPC) bases; for a recent overview on FPC-based approaches, see, e.g., Wang et al. (2016). The FPC-based estimation approaches considered in this thesis have the following four main steps in common:

- Step 1** The mean function including the fixed covariate effects $\mu(t_{ij}, \mathbf{x}_i)$ is estimated under a working independence assumption. This is done by representing the unknown functions in $\mu(t_{ij}, \mathbf{x}_i)$ using penalized splines. The response curves are subsequently centered by subtracting the estimated smooth mean curves.
- Step 2** Based on the centered responses, the covariances for the G grouping factors, $\mathbf{K}^{U_g}(t, t')$, $g = 1, \dots, G$, and the error variance σ^2 are estimated using either a method of moments approach and (if necessary) subsequent smoothing (Chapter 3) or a smooth method of moments approach (Chapters 4, 5). The choice depends on the sampling grid. The fast bivariate smoothing approach proposed in this thesis (Chapter 5) can be applied to both covariance estimation approaches in order to reduce computation time. After smoothing, the estimated covariances are evaluated on an equidistant grid.
- Step 3** Eigen decompositions of the estimated covariance matrices yielding estimated eigenfunctions and eigenvalues are used to represent the fREs in truncated bases of eigenfunctions of their covariances.
- Step 4** Replacing the fREs in the FLMM (2.8) by these truncated bases allows to approximate the FLMM by the scalar LMM (2.1) with random effects corresponding to the random basis weights (see also Di et al., 2009; Greven et al., 2010; Di et al., 2014). Thus, the FLMM reduces to an LMM for scalar data. The basis weights can then be predicted as EBLUPs (see Section 2.1.1) by plugging in the estimated eigenfunctions, eigenvalues and the estimated error variance.

Alternatively, for moderate sample sizes and when the interest is in statistical inference, e.g., confidence bands for covariate effects, the FPC estimation is combined with the functional additive mixed model approach (FAMM) of Scheipl et al. (2015), in which the random basis

weights are predicted together with a re-estimation of the mean function in a mixed model framework. Also in this alternative approach, the FLMM (2.2) reduces to a mixed model for scalar data.

More details on each estimation step for the different models and estimation approaches are given in Chapter 3 to Chapter 5. Further explanation and details on FPCA and multivariate FPCA (required for correlated fREs) are provided below in Section 2.2.2.

Fixed effects structures

Possible structures for the mean function $\mu(t_{ij}, \mathbf{x}_i)$ in Model (2.8), include linear and smooth effects of scalar covariates x_i that can be constant or varying over \mathcal{T} . Moreover, interactions between scalar covariates are possible. A detailed overview on possible effects of scalar covariates is given in Table 2.1. Note that in the current implementations of the proposed modeling framework in R (R Core Team, 2016), not all of these effects are allowed yet; see Appendix E and the manuals of the R add-on packages `denseFLMM` (Greven and Cederbaum, 2017) and `sparseFLMM` (Cederbaum, 2016) for more details.

Table 2.1: Overview of possible fixed effects of scalar covariates that can be included in the mean function $\mu(t_{ij}, \mathbf{x}_i)$ in the FLMM (2.8). This overview table is based on Scheipl et al. (2015).

$\tilde{\mathbf{x}}_{ik}$	constant over \mathcal{T}	varying over \mathcal{T} (functional (f.) effects)
scalar covariate x_i	linear effect: $x_i\beta$ smooth effect: $f(x_i)$	f. linear effect: $x_i f(t_{ij})$ f. smooth effect: $f(t_{ij}, x_i)$
vector of scalar covariates $\tilde{\mathbf{x}}_{ik}$	linear interaction effect: $x_{i1}x_{i2}\beta$ varying coefficient: $x_{i1}f(x_{i2})$ smooth effect: $f(\tilde{\mathbf{x}}_{ik})$	f. linear interaction effect: $x_{i1}x_{i2}f(t_{ij})$ f. varying coefficient: $x_{i1}f(t_{ij}, x_{i2})$ f. smooth effect $f(t_{ij}, \tilde{\mathbf{x}}_{ik})$

Strictly speaking, when non-linear effects (non-linear in the covariates) are included in the model, the term ‘functional *additive* mixed model’ rather than ‘functional *linear* mixed model’ is more appropriate. As non-linear effects are, however, not in the main focus, the term FLMM is used throughout this thesis.

For all effects that contain the functional intercept $f_0(t_{ij})$ as special case, centering constraints need to be imposed in order to avoid that different parametrizations lead to the same fit. This is in analogy to additive models for scalar data with an additive predictor of the form $\beta_0 + \sum_k f_k(x_k)$, where identifiability is commonly ensured by imposing sum-to-zero constraints of the form $\sum_{i=1}^n f_k(x_{ik}) = 0$ for each function (Wood, 2006, Chapter 4). Similarly, sum-to-zero constraints are necessary to ensure identifiability in Model (2.8), which need, however, hold for each $t_{ij} \in \mathcal{T}$, i.e., the mean effect of each covariate should be zero in each point t_{ij} . Such constraints need to be imposed for the fREs and for non-linear effects $f(t_{ij}, x_i)$ and $f(t_{ij}, \tilde{\mathbf{x}}_{ik})$. This specific centering constraint has the advantage that

all effects that vary over the index of the response can be interpreted as deviations from $f_0(t_{ij})$ (see Scheipl et al., 2015).

The combination of the modeling approach proposed in this thesis with the FAMM framework allows to additionally include effects of functional covariates, turning the model into a function-on-function regression model. The effects of functional covariates can again be linear or smooth and all kinds of interactions are possible; for an overview, see Scheipl et al. (2015), where also identifiability issues that arise when dealing with function-on-function regression are discussed; see also Scheipl et al. (2016b) for a more detailed and extensive discussion on identifiability in function-on-function regression.

Special cases

All models considered in this thesis are special cases of the general FLMM (2.8). In the following, it is shown how $\mathbf{z}_i^\top \mathbf{U}(t_{ij})$ is specified to obtain a model with a) only a curves-specific fRE, b) hierarchical fREs, and c) crossed fREs. Motivations and examples for each of these special cases are given in Chapters 3 to 5, respectively.

First of all, a closer inspection of the structure of the q -dimensional covariate vector \mathbf{z}_i is necessary. In accordance with the vector-valued fRE $\mathbf{U}(t_{ij})$, the covariate vector \mathbf{z}_i can be divided into G blocks, $\mathbf{z}_i^\top = \left(\mathbf{z}_i^{U_1^\top}, \dots, \mathbf{z}_i^{U_G^\top} \right)$. Each block can then again be written as $\mathbf{z}_i^{U_g^\top} = \left(z_{i1}^{U_g^\top}, \dots, z_{iL^{U_g}}^{U_g^\top} \right)$, with $\mathbf{z}_i^{U_g^\top} = \left(z_{i1}^{U_g}, \dots, z_{iL^{U_g}}^{U_g} \right)$, $l = 1, \dots, L^{U_g}$, $g = 1, \dots, G$. The scalars $z_{ils}^{U_g}$ take the value of the respective covariate, here denoted by $\omega_{is}^{U_g}$ times an indicator $\delta_{\ell_g(i)l}$ which specify whether observation i belongs to level l of grouping factor g .

- a) *Model with a curve-specific fRE.* The simplest model considered in this thesis (Chapter 5) is given by

$$Y_i(t_{ij}) = \mu(t_{ij}, \mathbf{x}_i) + E_i(t_{ij}) + \varepsilon_i(t_{ij}), \quad i = 1, \dots, n, \quad t_{ij} \in \mathcal{T}, \quad (2.9)$$

where the functional response of curve i at point t_{ij} is decomposed as in Model (2.8) with $\mathbf{z}_i^\top \mathbf{U}(t_{ij})$ reducing to $E_i(t_{ij})$. $E_i(t_{ij})$ is a curve-specific fRE for curve i and can also be seen as a smooth error that captures correlation along \mathcal{T} . As mentioned above, such a smooth error curve is commonly contained in the vector of fREs $\mathbf{U}(t_{ij})$ in order to meet the assumption of conditionally independent response values within and across functions. Model (2.9) is not an FLMM as it solely accounts for the within-function correlation but different functions are assumed to be independent. The model can rather be seen as an extension of a scalar LMM for longitudinal data that relaxes the parametric assumptions and allows for smooth effects along t .

Nonetheless, it is possible to represent Model (2.9) as a special case of the general model (2.8) by letting $G = 1$, $L^{U_1} = n$ and specifying \mathbf{z}_i as a block of indicators for each curve, i.e.,

$\mathbf{z}_i = (z_{i11}^{U_1}, \dots, z_{in1}^{U_1})^\top = (\delta_{\ell_1(i)1}, \dots, \delta_{\ell_1(i)n})^\top$. $\mathbf{U}(t_{ij})$ correspondingly consists of a block of curve-specific functional random effects, i.e.,

$$\mathbf{U}(t_{ij}) = [U_{111}(t_{ij}), \dots, U_{1n1}(t_{ij})]^\top = [E_1(t_{ij}), \dots, E_n(t_{ij})]^\top.$$

For simplicity, consider in the following again the case of two grouping factors ($G = 2$) with an fRI for the first grouping factor and correlated fRI and fRS (in variable ω) for the second grouping factor, i.e., $\rho^{U_1} = 1$, $\rho^{U_2} = 2$. Note that commonly also a smooth error as in Model (2.9) is included, which is omitted here for ease of notation. Let n_i denote the number of observations (curves) for each level of the second grouping factor (e.g., for each student in the above example), $i = 1, \dots, L^{U_2}$. Further let ω_i denote the value of variable ω for observation i . Then, $\mathbf{z}_i^\top \mathbf{U}(t_{ij})$, $i = 1, \dots, n$, is given by

$$\mathbf{z}_i^\top \mathbf{U}(t_{ij}) = \left(\underbrace{\delta_{\ell_1(i)1}, \dots, \delta_{\ell_1(i)L^{U_1}}}_{\text{fRI, } g=1}, \underbrace{\delta_{\ell_2(i)1}, \delta_{\ell_2(i)1}\omega_i, \dots, \delta_{\ell_2(i)L^{U_2}}, \delta_{\ell_2(i)L^{U_2}}\omega_i}_{\text{fRI and fRS, } g=2} \right) \begin{pmatrix} U_{111}(t_{ij}) \\ \vdots \\ U_{1L^{U_1}1}(t_{ij}) \\ U_{211}(t_{ij}) \\ U_{212}(t_{ij}) \\ \vdots \\ U_{2L^{U_2}1}(t_{ij}) \\ U_{2L^{U_2}2}(t_{ij}) \end{pmatrix} \left. \begin{array}{l} \left. \vphantom{\begin{matrix} U_{111}(t_{ij}) \\ \vdots \\ U_{1L^{U_1}1}(t_{ij}) \end{matrix}} \right\} g=1 \\ \left. \vphantom{\begin{matrix} U_{211}(t_{ij}) \\ U_{212}(t_{ij}) \\ \vdots \\ U_{2L^{U_2}1}(t_{ij}) \\ U_{2L^{U_2}2}(t_{ij}) \end{matrix}} \right\} g=2 \end{array} \right\} .$$

Hierarchical in contrast to crossed fREs are used when the levels of the second grouping factor are different for each level of the first grouping factor. The above example of students in schools is a typical example for a hierarchical fREs structure as the students are not the same for the different schools. In contrast, crossed fREs are suitable, for example, in the speech production data analyzed in this thesis (see Chapter 4 and Chapter 5), where acoustic signals of nine different speakers are recorded while reading the same sixteen target words. As each speaker reads the same target words, crossed fREs for speakers and target words are used to model speaker- and target word-specific deviations from the mean.

The hierarchical and the crossed fREs thus differ in the form of \mathbf{z}_i . Assume for simplicity that there are two and four levels of the two grouping factors ($L^{U_1} = 2$, $L^{U_2} = 4$). Then, in total $q = \sum_{g=1}^G \rho^{U_g} L^{U_g} = 10$ functional random effects are specified.

2.2.1 Multivariate principal component analysis

Principal component analysis (PCA; Pearson, 1901; Hotelling, 1933) is a key tool for exploration and dimension reduction in high-dimensional multivariate data sets. It can be used to structure and display the data and is frequently applied preliminary to, or in combination with, other statistical methods such as cluster analysis, factor analysis, or regression (see, e.g., Jolliffe, 2002, Chapters 7 to 9). The underlying idea of PCA is to replace a high number of possibly correlated variables by a typically smaller number of pairwise uncorrelated variables, the principal component scores (or weights), while retaining as much as possible of the variance present in the data. From a mathematical perspective, PCA corresponds to an orthogonal transformation of the data to a new vector space in which the basis is formed by the principal components (PCs). The PCs are ordered such that the first PC accounts for most variability and each succeeding PC accounts for less. A lossy compression of the data can thus be obtained by only using the first few PCs for the data representation. In order to prevent a serious information loss, the number of PCs has to be chosen carefully. In the following, the derivation of the PCs is outlined and important properties of PCA are discussed, which is partly based on Jolliffe (2002), Ramsay and Silverman (2005), and Bishop (2006).

Derivation of PCA

Let $\mathbf{x} \in \mathbb{R}^p$ denote a random vector with mean $\boldsymbol{\mu}$ and variance-covariance matrix $\boldsymbol{\Sigma}$. In the following, consider n independent samples of \mathbf{x} summarized in the $n \times p$ data matrix \mathbf{X} with rows $\mathbf{x}_i^\top \in \mathbb{R}^{1 \times p}$, $i = 1, \dots, n$. Assume that the columns of \mathbf{X} are centered and denote the $p \times p$ sample variance-covariance matrix of \mathbf{X} by \mathbf{S} . Let $\langle \mathbf{x}, \mathbf{y} \rangle$ denote the canonical scalar product of vectors \mathbf{x} and \mathbf{y} in the Euclidean space and let $\|\mathbf{x}\|$ denote the induced norm.

The aim is then to find normalized, orthogonal vectors $\boldsymbol{\phi}_k \in \mathbb{R}^p$, $k = 1, \dots, p$, that map each row of \mathbf{X} to a new vector while retaining as much as possible of the variation in the data.

The linear projections can be found following a stepwise procedure:

1. Find a linear projection of the data $\xi_{i1} = \langle \boldsymbol{\phi}_1, \mathbf{x}_i \rangle = \boldsymbol{\phi}_1^\top \mathbf{x}_i$ with maximal sample variance subject to the normalization constraint $\|\boldsymbol{\phi}_1\|^2 = 1$. The first PC is defined by

$$\boldsymbol{\phi}_1 = \arg \max_{\|\boldsymbol{\phi}\|^2=1} \frac{1}{n-1} \sum_{i=1}^n \left(\boldsymbol{\phi}^\top \mathbf{x}_i \right)^2 = \arg \max_{\|\boldsymbol{\phi}\|^2=1} \frac{1}{n-1} \boldsymbol{\phi}^\top \mathbf{X}^\top \mathbf{X} \boldsymbol{\phi},$$

which can be expressed in terms of the sample variance-covariance matrix \mathbf{S} as

$$\boldsymbol{\phi}_1 = \arg \max_{\|\boldsymbol{\phi}\|^2=1} \boldsymbol{\phi}^\top \mathbf{S} \boldsymbol{\phi} = \arg \max_{\|\boldsymbol{\phi}\|^2=1} \langle \mathbf{S} \boldsymbol{\phi}, \boldsymbol{\phi} \rangle.$$

Applying the Lagrangian method for constrained (quadratic) optimization (see, e.g., Fletcher, 2000, Chapter 9) shows that this is equivalent to finding the solution of the eigenvalue-eigenvector problem

$$\mathbf{S}\phi = \nu\phi,$$

where ν denotes an eigenvalue of \mathbf{S} and ϕ is the corresponding eigenvector. Let in the following denote the ordered eigenvalues of \mathbf{S} by $\nu_1 \geq \nu_2 \geq \dots \geq 0$. As the sample variance of the linear projects equals $\phi^\top \mathbf{S}\phi = \nu\phi^\top \phi = \nu$, the first PC, which is defined to explain most of the variability, is given by the eigenvector corresponding to the largest eigenvalue ν_1 . The number of non-zero eigenvalues of \mathbf{S} is given by the rank r of matrix \mathbf{X} .

2. The subsequent PCs ϕ_2, \dots, ϕ_p are found analogously with the additional constraint(s) that each PC is orthogonal to the previously obtained components. This guarantees that the new component explains variability not already explained by the previously obtained components and thus avoids the use of redundant information. Thus, the k th PC ϕ_k is given by the direction that maximizes the sample variance of the projected data subject to the normalization constraint $\|\phi\|^2 = 1$ and additional constraint(s) $\langle \phi, \phi_m \rangle = 0$, $m < k \leq p$. With the same justification as for the first PC, the k th component, $k \leq p$, is given by the eigenvector corresponding to ν_k , the k th largest eigenvalue of \mathbf{S} .

In other words, PCA of a data matrix corresponds to an eigen analysis of its sample variance-covariance matrix. The new coordinate system, to which the data are mapped, is given by the eigenvectors. The corresponding eigenvalues quantify the variance explained. The PCs, which give the dominant modes of variation, are unique up to sign.

A computationally efficient way to obtain the eigenvectors and eigenvalues of \mathbf{S} is to use a singular value decomposition (SVD) of the data matrix \mathbf{X} , which is given by

$$\mathbf{X} = \mathbf{A}\mathbf{G}\mathbf{\Phi}^\top, \tag{2.10}$$

where \mathbf{A} is a matrix with orthonormal columns of dimension $n \times r$, with r the rank of matrix \mathbf{X} . The columns of \mathbf{A} contain the left singular vectors to non-zero singular values. The $r \times r$ diagonal matrix \mathbf{G} has diagonal entries corresponding to the non-zero singular values s_k , which are sorted in descending order. The columns of the $p \times r$ matrix $\mathbf{\Phi}$ consist of the right singular vectors which are equivalent to the eigenvectors of \mathbf{S} corresponding to non-zero eigenvalues. The eigenvalues of \mathbf{S} can be obtained as $\nu_k = (n-1)^{-1}s_k^2$ (for a proof, see, e.g., Jolliffe, 2002). Note that this connection only holds for a centered data matrix \mathbf{X} . The columns of \mathbf{A} contain scaled PC weights and $\mathbf{A}\mathbf{G}$ corresponds to the matrix of PC weights. Estimating the PCs using the SVD of the data matrix has the advantage that the covariance matrix does not have to be built. Furthermore, the SVD of \mathbf{X} compared to the eigen analysis of \mathbf{S} allows a different view on PCA as linear approximation of the data.

Dimension reduction

So far, the projection maps the data from the original space of p possibly correlated variables to a new space of p uncorrelated variables, which—from a geometrical view—simply is a rotation (see, e.g., Bishop, 2006, Chapter 12). Dimension reduction can be achieved by mapping the data onto a lower-dimensional subspace of dimension $N < p$, which means that only the first N PCs are used and thus only the most relevant information in terms of explained variance is kept. The truncation of the PCs can be seen as regularization with a discrete regularization parameter, aiming to reveal the structure in the data and to filter out the noise. To achieve a considerable dimension reduction, N is chosen as small as possible while preventing a serious information loss. It is thus necessary to specify what an acceptable loss means.

The choice of N thus plays an important role in PCA. Various rules for choosing the subset of PCs have become standard, most of which are rather ad hoc. A common criterion that is relevant in the scope of this thesis is to choose N by selecting the percentage of total variation that is desired to be explained, in the following denoted by L , e.g., $L = 90\%$. The PCs are then chosen in decreasing order until the chosen percentage is exceeded. As the variance explained by the k th PC is given by the corresponding eigenvalue, the criterion can be written as

$$\text{Choose } \phi_k \text{ corresponding to } \nu_k \text{ in decreasing order, until } \sum_{k=1}^N \nu_k \bigg/ \sum_{k=1}^p \nu_k \geq L. \quad (2.11)$$

For an extensive overview on rules for choosing a subset of PCs, it is referred to Jolliffe (2002), Chapter 6.

Some properties of PCA

PCA has several interesting properties and motivations (see, e.g., Bishop, 2006). In fact, the PCs are not only optimal directions with respect to the maximum retained variance, the expansion in PC basis functions in Equation (2.10) also minimizes the mean squared approximation error for a given dimension N of the subspace to which the data are mapped. That is, using only the first $N < r$ PCs in the SVD expansion provides the optimal reduced basis representation of the original data with respect to the mean squared error (see, e.g., Jolliffe, 2002, Chapter 3).

PCA as derived above is based on the eigen decomposition of the sample variance-covariance matrix \mathbf{S} . Therefore, PCA is sensitive to the units of measurements used for the data which makes comparisons difficult and can lead to domination of variables with large variances. As a remedy, an eigen decomposition of the sample correlation matrix for standardized data can be used instead. When the data are measured in the same units, however, using the covariance matrix has the advantage that the data do not need to be standardized. Furthermore, PCA based on the covariance rather than the correlation matrix facilitates statistical inference based on the sample PCs (for a discussion, see Jolliffe, 2002, Chapter 2).

Connection to other approaches

PCA is closely connected to a number of statistical methods, including canonical correlation analysis (CCA; Hotelling, 1935), independent component analysis (ICA; see, e.g., Hastie et al., 2009), and principal component regression (PCR; Hotelling, 1957; Kendall, 1957). CCA extends the idea of PCA to two or more random variables and aims at finding a pair of linear subspaces that have a high cross-correlation. ICA can be seen as a generalization of PCA by additionally requiring that the new variables are statistically independent. In contrast to PCA, which is based on the first and second order moments, ICA employs higher moments which may be better suited for non-Gaussian data (see, e.g., Jolliffe, 2002, Chapter 14). To overcome the problem of multicollinearity in linear regression models, PCR uses a PCA of the design matrix and replaces the original matrix by the matrix consisting of the PC weights. PCR is closely related to partial least squares (PLS; see, e.g., Hastie et al., 2009) and to other shrinkage regression approaches such as Ridge Regression (Hoerl and Kennard, 1970). In contrast to PCR, PLS uses a projection that has a high variance and also has a high correlation with the response. Ridge regression utilizes an SVD of the data matrix and shrinks the directions corresponding to the smallest singular values most, using a continuous rather than a discrete regularization parameter. For further details and a discussion on the connection to other statistical methods, see, e.g., Jolliffe (2002) and Hastie et al. (2009).

2.2.2 From multivariate to functional principal component analysis

Functional principal component analysis (FPCA; see, e.g., Ramsay and Silverman, 2005, Chapter 8) can be considered as the functional analogue to multivariate PCA. Like in the multivariate case, FPCA uses an orthogonal transformation of the data to a lower-dimensional vector space while maximizing the retained variability in the data. As dimension reduction is of particular importance when the data consist of (at least theoretically) infinite-dimensional functions, FPCA was the first method to be considered in functional data analysis and is a key element since (Ramsay and Silverman, 2005, Chapter 8). FPCA is frequently applied for data representation and exploration and finds its use in conjunction with numerous other statistical methods, such as functional regression. The basic ideas and main properties of PCA carry over to the functional counterpart, which—in analogy to the multivariate case—corresponds to an eigen analysis of the covariance. The differences between PCA and FPCA come from the fact that in the functional case the data live in an infinite-dimensional space requiring an adaptation of both theory and application. The covariance matrix in multivariate PCA is replaced by a covariance operator, which turns the multivariate eigen problem into an eigen problem for a special class of linear operators. As a consequence, the main directions of variation, i.e., the principal components, are functions that vary over the same domain as the functional observations. In the following it is assumed that the functions live in the $L^2(\mathcal{T})$ -space of square integrable functions defined on a bounded interval $\mathcal{T} \subset \mathbb{R}$.

Transition to the functional case

The following outlines how the idea of PCA carries over to the functional framework. This

involves the extension of the definitions of a multivariate random variable, its mean vector and variance-covariance matrix, as well as the scalar product of vectors to their functional counterparts. The following is in parts based on Jolliffe (2002), Ramsay and Silverman (2005), and Horváth and Kokoszka (2012).

In the functional case, the random variable $\mathbf{x} \in \mathbb{R}^p$ is replaced by a square integrable random process $X(t) \in L^2(\mathcal{T})$ with mean function $\mu(t) = \mathbb{E}[X(t)]$ and auto-covariance function $K(t, t') = \text{Cov}[X(t), X(t')]$, $t, t' \in \mathcal{T}$. Thus, the mean in this context is a curve and the covariance is a surface. Consider, as in the multivariate context, n independent samples of $X(t)$, denoted by $x_1(t), \dots, x_n(t)$, where a common sampling grid is assumed to keep notation simple. Thus, the discrete index j of the elements x_{ij} in the multivariate case is replaced by the continuous function argument t , yielding $x_i(t)$, $i = 1, \dots, n$. In analogy to multivariate PCA, FPCA requires that the data are centered, which in the functional case means that the sample mean function, given by

$$\bar{x}(t) = \frac{1}{n} \sum_{i=1}^n x_i(t),$$

is zero for all $t \in \mathcal{T}$. Similarly, the sample auto-covariance function that summarizes the dependence of different function values can be defined as

$$k(t, t') = \frac{1}{n-1} \sum_{i=1}^n [x_i(t) - \bar{x}(t)] [x_i(t') - \bar{x}(t')]. \quad (2.12)$$

The sample variance function thus results from the sample auto-covariance function (2.12) by setting $t' = t$. The $p \times p$ sample variance-covariance matrix \mathbf{S} , on which the PCA is based in the multivariate case, is replaced by a sample auto-covariance operator, which can be seen as its infinite-dimensional counterpart. A brief introduction to covariance operators will be given below. Finally, the scalar product for vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$, $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^\top \mathbf{y} = \sum_{k=1}^p x_k y_k$, is replaced by the canonical inner product for functions $f, g \in L^2(\mathcal{T})$ given by

$$\langle f, g \rangle = \int_{\mathcal{T}} f(t)g(t) dt, \text{ which induces the } L^2\text{-norm } \|f\| = \langle f, f \rangle^{\frac{1}{2}} = \left[\int_{\mathcal{T}} f^2(t) dt \right]^{\frac{1}{2}}. \quad (2.13)$$

Thus, summation over the p variables in the scalar product is replaced by integration over the function domain \mathcal{T} , which accounts for the functional nature of the data.

The covariance operator

The covariance operator generalizes the variance-covariance matrix and is thus at the heart of FPCA. Assume for simplicity that $X(t)$ is centered, i.e., $\mu(t) \equiv 0$. The (auto-)covariance operator of $X(t)$ can then be defined as the mapping $K : L^2(\mathcal{T}) \rightarrow L^2(\mathcal{T})$ of the form

$$[Ky](t) = \int_{\mathcal{T}} K(t, t')y(t') dt', \quad y \in L^2(\mathcal{T}), \quad (2.14)$$

where $K(t, t') \in L^2(\mathcal{T} \times \mathcal{T})$ is the covariance function $K(t, t') = \mathbb{E}[X(t)X(t')]$. To illustrate the parallel of the covariance operator (2.14) to its multivariate counterpart, the variance-covariance matrix Σ can be written as linear mapping $\Sigma : \mathbb{R}^p \rightarrow \mathbb{R}^p$. Then, applying Σ to a vector $\mathbf{y} \in \mathbb{R}^p$ yields

$$[\Sigma \mathbf{y}]_t = \sum_{t'=1}^p \Sigma_{tt'} y_{t'}, \quad \mathbf{y} \in \mathbb{R}^p,$$

where $\Sigma_{tt'}$ denotes the (t, t') entry of Σ . Thus, the covariance operator (2.14) is defined in analogy to the covariance matrix, where summation is replaced by integration. The sample covariance operator \tilde{K} is then given by replacing the auto-covariance function $K(t, t')$ by the sample auto-covariance function $k(t, t')$.

The covariance operator (2.14) has several interesting properties that allow the direct extension of PCA to functional data. First of all, the covariance operator falls into the class of integral operators and the covariance function $K(t, t')$ is commonly referred to as the covariance kernel or kernel function. It can be shown that K is a Hilbert-Schmidt operator and thus continuous and compact (see, e.g., Horváth and Kokoszka, 2012, Chapter 2). Compact linear operators are closely related to operators with a finite-dimensional range, which facilitates the transition from PCA to FPCA as many matrix properties carry over. Moreover, compact linear operators can be approximated by a sequence of finite-dimensional operators (see, e.g., Atkinson and Han, 2009, Chapter 2), allowing for numerical treatment. It can be easily shown that K inherits the symmetry of the covariance function $K(t, t')$ and that it is positive semi-definite, i.e., $\langle Kx, x \rangle \geq 0$, $x \in L^2(\mathcal{T})$. As symmetric positive-definite Hilbert-Schmidt operator, K has an eigen decomposition of the form

$$Kx = \sum_{k=1}^{\infty} \nu_k \langle x, \phi_k \rangle \phi_k, \quad x \in L^2(\mathcal{T}),$$

with orthonormal eigenfunctions ϕ_k and real-valued eigenvalues ν_k that can be ordered as $\nu_1 \geq \nu_2 \geq \dots \geq 0$ as they are non-negative and have zero as their only possible point of accumulation (see, e.g., Naylor and Sell, 2000, Chapter 6). The eigen problem for covariance operator K can be written as

$$K\phi_k = \nu_k \phi_k \text{ and thus } \int_{\mathcal{T}} K(t, t') \phi_k(t') \, dt' = \nu_k \phi_k(t), \quad k \in \mathbb{N}. \quad (2.15)$$

Furthermore, Mercer's theorem (Mercer, 1909) for continuous kernel functions provides a series representation of the covariance function, given by

$$K(t, t') = \sum_{k=1}^{\infty} \nu_k \phi_k(t) \phi_k(t'), \quad t, t' \in \mathcal{T}, \quad (2.16)$$

and states that the series of (ν_k, ϕ_k) converges uniformly. For a detailed introduction to linear operators from a functional data analysis perspective, see, e.g., Horváth and Kokoszka (2012). Proofs can be found in, e.g., Bosq (2000).

Derivation of FPCA

As in the multivariate case, different motivations and derivations of FPCA exist. One is to search for an orthogonal projection of the original data that maximizes the retained variance (see, e.g., Shang, 2014). When the data are functions, the PC vectors ϕ are replaced by PC functions $\phi(t)$, which are called functional principal components.

Assume that $x_1(t), \dots, x_n(t)$ are centered. One aims to find normalized, orthogonal functions $\phi_k(t) \in L^2(\mathcal{T})$, $k \in \mathbb{N}$, that project each $x_i(t)$ to a new vector space while retaining as much as possible of the variation present in the data. The procedure to find the new basis can be carried out analogously to the multivariate case:

1. Find a linear projection of the data $\xi_{i1} = \langle \phi_1, x_i \rangle = \int_{\mathcal{T}} \phi_1(t) x_i(t) dt$ with maximal sample variance subject to the normalization constraint $\|\phi_1\|^2 = \int_{\mathcal{T}} \phi_1^2(t) dt = 1$. The first FPC is then defined by

$$\phi_1(t) = \arg \max_{\|\phi\|^2=1} \frac{1}{n-1} \sum_{i=1}^n \langle \phi, x_i \rangle^2 = \arg \max_{\|\phi\|^2=1} \frac{1}{n-1} \sum_{i=1}^n \left(\int_{\mathcal{T}} \phi(t) x_i(t) dt \right)^2,$$

which can—in analogy to multivariate PCA—be expressed in terms of the sample covariance operator (see, e.g., Tran, 2008, Chapter 1)

$$\phi_1(t) = \arg \max_{\|\phi\|^2=1} \langle \tilde{K} \phi, \phi \rangle. \quad (2.17)$$

Due to the nice properties of the covariance operator (see above), the maximization problem (2.17) can be solved in analogy to the multivariate case by considering the eigen problem of the sample covariance operator \tilde{K} . The first FPC is given by the eigenfunction corresponding to the largest eigenvalue (see, e.g., Ramsay and Silverman, 2005, Chapter 8).

2. The k th FPC can be defined analogously to the maximization problem (2.17), subject to the additional constraint $\langle \phi, \phi_m \rangle = 0$ for $m < k$. The k th FPC is thus given by the eigenfunction corresponding to the k th largest eigenvalue.

In summary, FPCA can be derived in analogy to multivariate PCA and results in an eigen analysis of the sample covariance operator. The new basis is given by the eigenfunctions and the corresponding eigenvalues quantify the amount of explained variance. The new variables, also called FPC scores or FPC weights are defined as $\xi_{ik} = \langle \phi_k, x_i \rangle$. As in the multivariate case, the FPCs are only unique up to sign. Note that for simplicity, the same notation is used here for the eigenfunctions, eigenvalues, and FPC weights no matter if they correspond to the covariance operator or to the sample covariance operator.

Another way to look at FPCA arises from the Karhunen-Loève (KL) expansion (Loève, 1946; Karhunen, 1947) of a random process. Based on Mercer's theorem (2.16), the KL expansion provides a series representation, which breaks up the random process into fixed orthogonal components $\phi_k(t)$ and random uncorrelated basis weights ξ_k . For a zero mean random process $X(t)$, the KL expansion is given by

$$X(t) = \sum_{k=1}^{\infty} \xi_k \phi_k(t), \quad (2.18)$$

where $\xi_k = \langle \phi_k, x \rangle$, $k \in \mathbb{N}$, are uncorrelated real zero mean random variables with variance ν_k . The series (ν_k, ϕ_k) is defined as in Mercer's theorem (2.16). That is, the KL expansion represents the random process in a basis of eigenfunctions of its covariance operator and the variance of each basis weight corresponds to the respective eigenvalue. The series in (2.18) converges uniformly with respect to the L^2 -norm in (2.13) (see, e.g., Bosq, 2000, Chapter 1). Under a Gaussian process assumption, the basis weights ξ_k are independent (Hall and Hosseini-Nasab, 2006). For a given functional data set, the KL expansion allows to interpret the FPC weights ξ_{ik} as individual weights of the contribution of the k th FPC to curve i .

Dimension reduction

In particular in the functional case, dimension reduction plays an important role. With the same justification as in the multivariate context, the dimension can be reduced by using only the first N FPCs to represent the data. The original data are then approximated by truncated KL expansions. Note that in practice, the data consist of vectors of discrete observations rather than continuous functions. That is, the maximal number of FPCs that can be obtained in practice is limited by either the number of observations or by the number of grid points on which the covariance is evaluated. This may depend on both, the original sampling grid and potential pre-processing steps.

As in multivariate PCA, various rules for choosing the optimal number of FPCs have been proposed in the literature; for an overview, see, e.g., Di et al. (2009), Greven et al. (2010), and Shang (2014). In this thesis, the subset of FPCs is chosen based on the fraction of variability explained. The definition of the criterion is a direct extension of the definition in (2.11) to the infinite-dimensional case, i.e., the summation limit in the denominator in (2.11), p , is replaced by infinity and in practice by the maximal number of eigenvalues. The truncation level controls the amount of regularization and can thus be seen as discrete regularization parameter. This is in analogy to other basis approaches, such as spline bases, where the number of basis functions controls the bias-variance trade-off. Commonly, the FPCs decline in smoothness as dominant variation of a smooth random process tends to be smooth (e.g., Peng and Paul, 2009). The rate of decay of the eigenvalues thus gives insight into the smoothness of the covariance function (Rasmussen and Williams, 2006). The number of FPCs that is needed to avoid a severe loss of information is an indicator for the complexity of the data (Ramsay and Silverman, 2005, Chapter 8).

Some properties of FPCA

Similar to the multivariate case, it can be shown that the truncated KL expansion yields the best approximation of the data with respect to the L^2 -norm for a given truncation level (see, e.g., Ramsay and Silverman, 2005, Chapter 8). In other words, the approximation of a random process $X(t)$, denoted as $\widehat{X}(t)$, in a basis of eigenfunctions of its covariance operator as in Equation (2.18) yields a smaller approximation error

$$\int_{\mathcal{T}} \left[\widehat{X}(t) - X(t) \right]^2 dt$$

than in any other basis with the same number of basis functions. FPC bases therefore yield a parsimonious representation of the data, which makes them a popular choice in dimension reduction problems (see Chapter 4 for a simulation comparison of FPC bases with spline bases).

Besides the aspect of reducing high-dimensional data, the FPCs commonly allow for interesting interpretations. The FPC weights provide insights into the individual structure of the curves and can be used in further analysis, such as classification or clustering. For sparsely sampled data, expansions in FPC bases can be used for interpolation (see, e.g., Wang et al., 2016).

The role of FPCA in this thesis

In this thesis, FPC bases are used to represent the fREs in FLMMs. As the data are available on a discrete sampling grid and are observed with error, smoothing is applied in the estimation of the mean function as well as in the estimation of the covariance function of each fRE. The challenge of the covariance estimation lies in the fact that the fREs are latent underlying processes that cannot be observed. Two new approaches for estimating the smooth covariances in FLMMs are proposed for equal sampling grids and for unequal or sparse sampling grids in Chapters 3 and 4, respectively. Once the smooth covariance estimates are available, they can be evaluated on a pre-specified dense grid yielding estimated covariance matrices. Eigen decompositions of these matrices yield estimated eigenvectors and eigenvalues. In this way, FPCA can be transferred to multivariate PCA for given covariance estimates. Rescaling is necessary to ensure that the estimated eigenvectors, which serve as estimates for the eigenfunctions of the covariance operators, are orthonormal with respect to the inner product in L^2 . The eigenvalues are then adjusted accordingly (for details, see Appendix C).

Consider for simplicity Model (2.9) with one curve-specific fRI. The truncated KL expansion of $E_i(t_{ij})$ is given by

$$E_i(t_{ij}) \approx \sum_{k=1}^N \xi_{ik} \phi_k(t_{ij}),$$

where ξ_{ik} and $\phi_k(t_{ij})$ are the FPC weights and FPCs of the fRI, respectively. The FPCs are obtained as described above as the rescaled eigenvectors of the discretized estimated covariance. Recall that the FPC weights are defined as $\xi_{ik} = \langle \phi_k, E_i \rangle = \int_{\mathcal{T}} \phi_k(t) E_i(t) dt$. Traditionally, numerical approximation of the integral is used to predict the FPC weights in simpler situations (e.g., Ramsay and Silverman,

2005). This is, however, not possible here as $E_i(t_{ij})$ is a latent, unknown process. Moreover, when the data are sparsely sampled, numerical integration would not work (well). Instead, this thesis makes use of the fact that replacing the fRE by its truncated KL expansion allows to approximate Model (2.9) for discrete observation points by a scalar LMM of the form

$$Y_i(t_{ij}) \approx \mu(t_{ij}) + \sum_{k=1}^N \xi_{ik} \phi_k(t_{ij}) + \varepsilon_i(t_{ij}),$$

in which the random effects correspond to the FPC weights, which can be predicted as EB(L)UPs (see, e.g., Di et al., 2009; Greven et al., 2010). This way of predicting the FPC weights is extended to the general FLMM (2.8), in which multiple fREs are approximated by their truncated KL expansions (Chapter 3). In such models for functional data with multiple sources of variation, FPC bases have the additional advantage that they allow for an explicit decomposition of the variability.

Multivariate FPCA

Above, FPCA has been described for univariate functional data, that is, for an univariate underlying random process. For the general FLMM (2.8), in which the fREs for the same level of a grouping factor are assumed to be correlated, an extension of FPCA to multivariate FPCA dealing with vector-valued random processes is necessary, which will be briefly outlined in the following. For more details on multivariate FPCA, it is referred to Ramsay and Silverman (2005), Jacques and Preda (2014b), Chiou et al. (2014), as well as Happ and Greven (2017).

Consider the vector-valued random process $\mathbf{U}_g(t_{ij})$, which is defined as in Section 2.1.2. In order to study the joint variation of the ρ^{U_g} functional random effects, not only the auto-covariance functions $\text{Cov}[U_{gls}(t), U_{gls}(t')]$, $t, t' \in \mathcal{T}$, $s = 1 \dots, \rho^{U_g}$, but also the pairwise cross-covariance functions $\text{Cov}[U_{gls}(t), U_{gls'}(t')]$, $s \neq s'$, are considered in the FPCA. The cross-covariance operator is defined in analogy to the auto-covariance operator (2.14) as (e.g., Hsing and Eubank, 2015)

$$\left[K_{ss'}^{U_g} y \right] (t) = \int_{\mathcal{T}} K_{ss'}^{U_g}(t, t') y(t') \, dt', \quad s \neq s', \quad y \in L^2(\mathcal{T}),$$

with cross-covariance function $K_{ss'}^{U_g}(t, t') = \text{Cov}[U_{gls}(t), U_{gls'}(t')]$, which is the same for all levels $l = 1, \dots, L^{U_g}$.

When all ρ^{U_g} components in the vector-valued random process are measured on the same one-dimensional interval and have similar variation, a suitable inner product between the functions is given by the additive inner product (see, e.g., Ramsay and Silverman, 2005, Chapter 8), which is defined as

$$\left\langle \left(f_1, \dots, f_{\rho^{U_g}} \right), \left(g_1, \dots, g_{\rho^{U_g}} \right) \right\rangle = \sum_{s=1}^{\rho^{U_g}} \int_{\mathcal{T}} f_s(t) g_s(t) \, dt \quad f_s, g_s \in L^2(\mathcal{T}). \quad (2.19)$$

A multivariate extension of Mercer's theorem for the complete covariance operator with covariance kernel $\mathbf{K}^{U_g}(t, t')$ is then given by (Balakrishnan, 1960; Kelly and Root, 1960)

$$\mathbf{K}^{U_g}(t, t') = \sum_{k=1}^{\infty} \nu_k^{U_g} \phi_k^{U_g}(t) \left[\phi_k^{U_g}(t') \right]^\top, \quad (2.20)$$

with eigenvalues $\nu_k^{U_g}$ and vector-valued eigenfunctions $\phi_k^{U_g}(t) = \left[\phi_{ks}^{U_g}(t) \right]_{s=1, \dots, \rho^{U_g}}$, $k \geq 1$. Based on the additive inner product (2.19) and the vector-valued extension of Mercer's theorem (2.20), univariate FPCA can be extended to multivariate FPCA by concatenating the functions and applying univariate FPCA to the composite function (see, e.g., Ramsay and Silverman, 2005, Chapter 8). An extension of the KL expansion to vector-valued random processes is then given by (Balakrishnan, 1960; Kelly and Root, 1960)

$$\mathbf{U}_{gl}(t_{ij}) = \sum_{k=1}^{\infty} \xi_{lk}^{U_g} \phi_k^{U_g}(t_{ij}), \quad l = 1, \dots, L^{U_g}, \quad (2.21)$$

with uncorrelated zero mean random basis weights $\xi_{lk}^{U_g}$ with variance $\nu_k^{U_g}$, $k \geq 1$. In analogy to univariate FPCA, the multivariate FPCs $\phi_k^{U_g}(t)$, $k \geq 1$, represent the main modes of variation and the corresponding eigenvalues represent the amount of variability explained. The multivariate extension of the KL expansion (2.21) allows to interpret the random basis weights $\xi_{lk}^{U_g}$ as the individual weights of the contribution of the k th multivariate FPC. The FPC weights are uncorrelated by construction. For vector-valued random processes, multivariate FPCA has at least two advantages over applying separate univariate FPCA to each random function in the vector. First, multivariate FPCA yields a more parsimonious representation of the data as each level of the grouping factor g has a single basis weight per vector-valued FPC $\phi_k^{U_g}(t_{ij})$. Second, the FPC weights in multivariate FPCA explicitly model the correlation between the functional random effects contained in $\mathbf{U}_g(t_{ij})$ in contrast to univariate FPCA, where the correlation is captured only implicitly yielding correlated FPC weights (Happ and Greven, 2017).

Normalized extensions that apply to multivariate FPCA with components of different nature, e.g., different degrees of variability or units of measurement, have been proposed by Jacques and Preda (2014b) and Chiou et al. (2014). Both approaches assume that the components are measured on the same one-dimensional interval. Happ and Greven (2017) further extend multivariate FPCA to components measured on different domains that may also differ in dimension. They derive a KL expansion and establish the link to univariate KL expansions facilitating the calculation of multivariate FPCs and basis weights based on their univariate counterparts.

Chapter 3

Functional Linear Mixed Models for Equal Sampling Grids

Contributing manuscript

This chapter is based on the following working paper in preparation:

Greven, S., Cederbaum, J., and Shou, H. (2016): Principal component-based functional linear mixed models. *Working paper*.

This is joint work with Sonja Greven (Department of Statistics, LMU Munich, Germany) and Haochang Shou (Department of Biostatistics and Epidemiology, University of Pennsylvania, Philadelphia, United States). Sonja Greven had the idea of a general class of functional linear mixed models and derived the modeling framework. She implemented the main part of the method in R (R Core Team, 2016) and investigated the computational effort. Jona Cederbaum extended the implementation to allow for group-specific covariances of functional random effects, which is necessary for the application to the tissue spectroscopy data that were kindly provided by Florian Stelzle (Department of Oral and Maxillofacial Surgery, Erlangen University Hospital, Germany) and Werner Adler (Department of Medical Informatics, Biometry and Epidemiology, Friedrich-Alexander University Erlangen-Nuremberg, Germany). Jona Cederbaum also made corrections in the implementation. She conducted all data analyses, including the application of the functional linear mixed models to tissue classification based on the idea of Sonja Greven to combine the approach with the work of Zhu et al. (2012). She tried out alternative classification ideas based on functional linear mixed models not included in the manuscript. She implemented different versions of the classification and compared the classification rates with those of several competing approaches, some of which are shown in the manuscript and in this chapter. Haochang Shou conducted simulation studies, which are not shown as part of this thesis for reasons of incompleteness. She also pointed out a mistake in the implementation of group-specific covariances. Sonja Greven conducted the literature research on functional linear mixed models and wrote the main part of the manuscript. Haochang Shou contributed to the

manuscript by writing the simulation section, which was proofread by and discussed in close collaboration with the other co-authors. Jona Cederbaum investigated the literature on classification, wrote the application section, and added the discussion of this chapter which is lacking in the manuscript. She also contributed in proofreading the manuscript. Jona Cederbaum made the implementation of the approach available to users in the R add-on package `denseFLMM` (Greven and Cederbaum, 2017).

This chapter is a modified version of the working paper in preparation Greven et al. (2016). Sections 3.1 to 3.4 are very similar to the corresponding sections of the manuscript. The main modifications concern wording and illustrations. In addition, some references were added in this chapter and small corrections were made in the formula for the choice of the truncation lags and in the application section. Moreover, the notation was adapted to the notation of this thesis and references to the recently released R add-on package `denseFLMM` implementing the approach are added. The simulation section of the manuscript was omitted in this chapter. The discussion of this chapter was prepared for this thesis.

Software

The analyses within this chapter were carried out using R version 3.2.2 (2015-12-10) (R Core Team, 2016) on the platform `x86_64-pc-linux-gnu` (64-bit). The add-on package `Matrix` (Bates and Mächler, 2017, version 1.2-8) was used for data preparation, for the construction of the design matrices in case of group-specific covariances, and for matrix operations during covariance estimation. For smoothing the covariance functions, the add-on package `mgcv` (Wood, 2006, 2011, version 1.8-17) was applied, which depends on the add-on package `nlme` (Pinheiro et al., 2016, attached version 3.1-124). For parallelization of the nested cross-validation, the add-on packages `foreach` (Analytics and Weston, 2015b, version 1.4.3) and `doMC` (Analytics and Weston, 2015a, version 1.3.4) were used, the latter of which depends on the add-on package `iterators` (Analytics and Weston, 2015c, attached version 1.0.8). For comparison with penalized discriminant analysis, the add-on packages `mda` (Hastie et al., 2016, version 0.4-9) and `class` (Venables and Ripley, 2002, version 7.3-14) were additionally employed.

3.1 Introduction

Scientific studies now commonly collect functional or imaging data subject to an additional correlation structure due, for example, to a longitudinal, crossed, nested or spatial study design. Our motivating data set comes from a tissue spectroscopy study, where multiple reflectance spectra are collected each at several spots on a number of pigs for four different tissue types. Interest lies in using these spectra for classification into tissue groups, while also trying to understand the different sources of variability in the data. As each observation consists of a spectrum, the data are functional. The study design induces a hierarchical structure and correlation in these functional data, which is in part also tissue-specific.

Correlated scalar data are commonly analyzed using mixed models (see, e.g., Pinheiro and Bates, 2000). For functional data (e.g., Ramsay and Silverman, 2005; Ferraty and Vieu, 2006), functional

versions of linear mixed models have been proposed under varying degrees of generality and using different bases to expand functions in the model; see Liu and Guo (2012), Morris (2015), and Greven and Scheipl (2017) for recent reviews. Approaches typically either use splines (Guo, 2002, 2004; Bigelow and Dunson, 2007; Baladandayuthapani et al., 2008; Scarpa and Dunson, 2009; Chen, 2012; Scheipl et al., 2015), wavelets (Morris et al., 2003; Morris and Carroll, 2006; Abramovich and Angelini, 2006; Antoniadis and Sapatinas, 2007; Morris et al., 2011; Zhu et al., 2011) or functional principal components (Di et al., 2009; Greven et al., 2010; Aston et al., 2010; Zhou et al., 2010; Staicu et al., 2010; Di et al., 2014; Shou et al., 2015, and Chapters 4 and 5 of this thesis) to expand the functional fixed and random effects.

A large part of previous work has been limited to particular special cases of functional linear mixed models (FLMMs) often motivated by applications, such as the functional random intercept model (Abramovich and Angelini, 2006; Di et al., 2009; Krafty et al., 2011), functional intercept and slope model (Greven et al., 2010), a single level of random effects functions (Guo, 2002; Qin and Guo, 2006; Antoniadis and Sapatinas, 2007), a two or three-level hierarchy with or without spatial correlation on the last level (Brumback and Rice, 1998; Morris et al., 2003; Baladandayuthapani et al., 2008; Bigelow and Dunson, 2007; Li et al., 2007; Scarpa and Dunson, 2009; Zhou et al., 2010; Staicu et al., 2010), or nested or crossed random intercepts (Shou et al., 2015 and Chapter 4 of this thesis). An extensive literature is also available on models with smooth residual process only (e.g., Wu and Zhang, 2002; Yao et al., 2005; Chen and Wang, 2011).

Among the most general of previous works are the functional additive mixed models (FAMMs) of Scheipl et al. (2015), who however assume independence across functional random effects, with the P-spline-based approach incurring much higher computational cost than a functional principal component (FPC)-based approach in the case of many random effects levels (see Chapter 4). How to combine the FAMM approach and our FPC-based approach to best advantage is briefly outlined in Section 3.3.1 (see also Greven and Scheipl, 2017 and Chapter 4 of this thesis). Aston et al. (2010) assume the same FPC basis for all random processes in the model, which they estimate using a working independence assumption between curves. The assumption of common FPC bases for all latent processes may be a limitation in applications different from theirs and would be too restrictive in our application to the tissue spectroscopy data. Moreover, it is unclear how the correlation between curves in our application would affect the properties of the FPCs estimated from all the data. Morris and Carroll (2006) (with extensions to robust regression and images given in Zhu et al., 2011; Morris et al., 2011) consider a general FLMM and Bayesian estimation based on wavelet decompositions. Whereas wavelets are well-suited to spiky data and require equidistant grids with length a power of two, our approach can be seen as complementary in terms of assuming smooth underlying processes and allowing for non-equidistant grids of arbitrary length and potential missings.

We consider very general FLMMs which can, for example, accommodate nested and/or crossed designs, correlated functional random intercepts and/or slopes as well as group-specific covariances of functional random effects, which is relevant for our application in Section 3.4. We base estimation on process-specific FPCs, generalizing Di et al. (2009), Greven et al. (2010), Shou et al. (2015), and Chapters 4 and 5 of this thesis to more general models while pursuing a similar approach to estimating

the mean function and to predicting the basis weights. Our FPC-based approach, in contrast to spline or wavelet-based approaches, provides an explicit variance decomposition with interpretable main modes of variation. The approach increases computational efficiency by using a comparatively small number of FPC basis functions, using the fact that for a given number of basis functions, FPCs provide the best approximation to the data (see, e.g., Wang et al., 2016). How to estimate FPCs for the latent processes in very general FLMMs is non-trivial and has not been described before other than for certain special cases and for a general functional linear mixed model without group-specific covariances of random effects in Chapter 5 of this thesis. Extensions of the approach to images and to sparsely sampled functional observations seem feasible (compare Yao et al., 2005; Di et al., 2014; Zipunnikov et al., 2014, and Chapters 4 and 5 of this thesis, for simpler models). We will here focus on functions observed on a common one-dimensional grid, not necessarily equidistant, possibly with some missings, as this structure corresponds to our application and we make use of the grid structure to increase computational efficiency.

We provide fully documented open-source software implementing our estimation approach in the R add-on package `denseFLMM` (R Core Team, 2016; Greven and Cederbaum, 2017). A description and examples for the usage of the R package as well as code for the analysis of the tissue spectroscopy data can be found in Appendix E.

The remainder of the chapter is organized as follows. Sections 3.2 and 3.3 describe the considered FLMM and our FPC-based approach to estimation. We apply our method to the tissue spectroscopy data in Section 3.4. The chapter closes with a discussion in Section 3.5.

3.2 The functional linear mixed model

We consider functional linear mixed models of the form

$$Y_i(t) = \mu(t, \mathbf{x}_i) + \mathbf{z}_i^\top \mathbf{U}(t) + \varepsilon_i(t), \quad i = 1, \dots, n, \quad (3.1)$$

where the domain \mathcal{T} for t is a bounded interval in \mathbb{R} . Here, $\mu(t, \mathbf{x}_i)$ denotes a curve-specific smooth mean function dependent on scalar and/or functional covariates \mathbf{x}_i . Possible structures for $\mu(t, \mathbf{x}_i)$ are described in Section 3.3.1. \mathbf{z}_i is a covariate vector of length q and $\mathbf{U}(t)$ denotes a vector of functional random effects with possibly group-specific covariances as in our application in Section 3.4. $\varepsilon_i(t)$ represents white noise measurement error. Usually, the functional random effects will additionally include a smooth error term which is a functional random intercept with a special structure that captures deviations from the mean which are correlated along \mathcal{T} . In this case, the last block of \mathbf{z}_i corresponds to an indicator vector of indicators for each curve and the last block in $\mathbf{U}(t)$ consists of curve-specific functional random effects. In applications in which the curve-specific variability is high and depends on a grouping variable, it may be desirable to replace the smooth error term by a group-specific smooth error, i.e., with group-specific auto-covariance.

We assume $\varepsilon_i(t)$ to be i.i.d. mean zero random variables with variance σ^2 for all i and t . We further assume that $\mathbf{U}(t)$ is a vector-valued zero mean square integrable random process on \mathcal{T} and that $\mathbf{U}(t')$ is independent of $\varepsilon_i(t)$ for all i, t, t' .

The structure for the functional random effects can be quite general, including, e.g., hierarchical, crossed and/or correlated functional random effects, and mimics the possible structures for scalar random effects in popular linear mixed model implementations such as in the packages `nlme` (Pinheiro et al., 2016) or `lme4` (Bates et al., 2016) in R (R Core Team, 2016). For each $t \in \mathcal{T}$, Model (3.1) can in fact be seen as a scalar linear mixed model, while our covariance assumptions below ensure smoothness over the additional dimension t . In the following, we aim to stay close in notation to Bates et al. (2015) in order to exploit familiarity with the scalar case, where possible.

If we have G random effect terms in our model, i.e., G different grouping factors, we subdivide $\mathbf{U}(t) = [\mathbf{U}_1(t)^\top, \dots, \mathbf{U}_G(t)^\top]^\top$ into the G independent blocks of functional random effects, with $\mathbf{U}_g(t)$ and $\mathbf{U}_{g'}(t)$ independent for $g \neq g'$. Each block $\mathbf{U}_g(t)$ further contains L^{U_g} independent copies $\mathbf{U}_{gl}(t)$, $l = 1, \dots, L^{U_g}$, of a vector-valued stochastic process with ρ^{U_g} vector components $U_{gls}(t)$, $s = 1, \dots, \rho^{U_g}$, yielding $\mathbf{U}_g(t) = [U_{g11}(t), \dots, U_{g1\rho^{U_g}}(t), \dots, U_{gL^{U_g}1}(t), \dots, U_{gL^{U_g}\rho^{U_g}}(t)]$. The total number of functional random effects then amounts to $q = \sum_{g=1}^G L^{U_g} \rho^{U_g}$. The matrix-valued covariances $\mathbf{K}^{U_g}(t, t') = \text{Cov}[\mathbf{U}_{gl}(t), \mathbf{U}_{gl}(t')]$, $l = 1, \dots, L^{U_g}$, consist of blocks $\text{Cov}[U_{gls}(t), U_{gls'}(t')]$, $s, s' = 1, \dots, \rho^{U_g}$, which are assumed to be smooth in t and t' .

To give some intuition, consider the example of subjects nested within groups and a model including for each group a functional random intercept and for each subject a correlated functional random intercept and functional random slope. Then, we have $G = 2$ independent blocks of functional random effects, with the number of independent levels L^{U_1} and L^{U_2} equal to the number of groups and of subjects, respectively. For the corresponding vector-valued random processes, the lengths are $\rho^{U_1} = 1$ (only random intercept) and $\rho^{U_2} = 2$ (random intercept and slope), respectively. Uncorrelated functional random intercept and slope on the subject level would be assumed by letting $G = 3$, $L^{U_3} = L^{U_2}$, and $\rho^{U_2} = \rho^{U_3} = 1$.

For estimation of Model (3.1), dimension reduction will be important. We choose to base estimation on FPCs, which typically yield a small basis explaining most of the variation in the data, and thus allow for efficient computations as well as interpretable results. Given our assumptions and using an extension of Mercer's theorem (Mercer, 1909) to vector-valued random processes (Balakrishnan, 1960; Kelly and Root, 1960) and an extension of the Karhunen-Loève expansion (Loève, 1946; Karhunen, 1947) to vector-valued random processes (Balakrishnan, 1960; Kelly and Root, 1960), we can write the random processes as

$$\mathbf{U}_{gl}(t) = \sum_{k=1}^{\infty} \xi_{lk}^{U_g} \boldsymbol{\phi}_k^{U_g}(t) \quad (3.2)$$

for all g, l , where the vector-valued eigenfunctions $\boldsymbol{\phi}_k^{U_g}(\cdot) = [\phi_{k1}^{U_g}(\cdot), \dots, \phi_{k\rho^{U_g}}^{U_g}(\cdot)]^\top$ corresponding to the eigenvalues $\nu_1^{U_g} \geq \nu_2^{U_g} \geq \dots \geq 0$ in $\mathbf{K}^{U_g}(t, t') = \sum_{k=1}^{\infty} \nu_k^{U_g} \boldsymbol{\phi}_k^{U_g}(t) [\boldsymbol{\phi}_k^{U_g}(t')]^\top$ form an orthonormal basis for the direct sum of $\rho^{U_g} L^2(\mathcal{T})$ -spaces with respect to the additive inner product

$$\left\langle (f_1, \dots, f_{\rho^{U_g}}), (g_1, \dots, g_{\rho^{U_g}}) \right\rangle = \sum_{s=1}^{\rho^{U_g}} \int_{\mathcal{T}} f_s(t) g_s(t) dt \quad (3.3)$$

and $\xi_{lk}^{U_g}$ are uncorrelated zero mean random basis weights (also denoted as FPC scores) with variance $\nu_k^{U_g}$ for all g, l, k . In practice, the infinite sums in Equation (3.2) will be truncated at suitably chosen values N^{U_g} .

Now, assume that each curve is observed on the same D grid points $\mathcal{D} = \{t_1, \dots, t_D\}$. Note that a common grid for all observations facilitates notation and speeds up computation, but that our method can also handle missing values; we will discuss this case in Section 3.3.5. Then, the discretized version of Model (3.1) becomes

$$\mathbf{Y} = \boldsymbol{\mu} + \mathbf{Z}\mathbf{U} + \boldsymbol{\varepsilon}, \quad (3.4)$$

where $\mathbf{Y} = [Y_i(t_d)]_{i=1, \dots, n, d=1, \dots, D}$ is an $n \times D$ matrix, $\boldsymbol{\mu}$ and $\boldsymbol{\varepsilon}$ are analogously built $n \times D$ matrices, the $n \times q$ matrix \mathbf{Z} contains rows \mathbf{z}_i^\top , $i = 1, \dots, n$, and \mathbf{U} is $q \times D$ with columns $\mathbf{U}(t_d)$, $d = 1, \dots, D$.

Let \mathbf{Z} be divided as $\mathbf{Z} = [\mathbf{Z}^{U_1} | \dots | \mathbf{Z}^{U_G}]$ with $\mathbf{Z}^{U_g} = [\mathbf{Z}_{11}^{U_g} | \dots | \mathbf{Z}_{1\rho^{U_g}}^{U_g} | \dots | \mathbf{Z}_{L^{U_g}1}^{U_g} | \dots | \mathbf{Z}_{L^{U_g}\rho^{U_g}}^{U_g}]$, where $\mathbf{Z}_{ls}^{U_g}$ is a vector containing the covariate values for the s th functional random effect for grouping factor g for the l th level of that grouping factor, times an indicator for that level. Let \mathbf{U} be correspondingly divided as $\mathbf{U} = [\mathbf{U}_1^\top | \dots | \mathbf{U}_G^\top]^\top$ with $\mathbf{U}_g = [\mathbf{U}_{g1}^\top | \dots | \mathbf{U}_{gL^{U_g}}^\top]^\top$ and $\mathbf{U}_{gl} = [\mathbf{U}_{gl1}^\top | \dots | \mathbf{U}_{gl\rho^{U_g}}^\top]^\top$. Our above-described assumptions then mean that \mathbf{U}_g and $\mathbf{U}_{g'}$ are uncorrelated for $g \neq g'$ (different grouping factors) and \mathbf{U}_{gl} and $\mathbf{U}_{gl'}$ are uncorrelated for $l \neq l'$ (independent copies), but that \mathbf{U}_{gls} and $\mathbf{U}_{gls'}$ are correlated in general for $s \neq s'$ (e.g., correlated functional random intercept and slope).

For given truncation lags N^{U_g} and eigenfunctions $\{\phi_k^{U_g}(\cdot), k = 1, \dots, N^{U_g}\}$, $g = 1, \dots, G$, we can approximate Model (3.4) as

$$\begin{aligned} \mathbf{Y} &\approx \boldsymbol{\mu} + \sum_{g=1}^G \sum_{l=1}^{L^{U_g}} \sum_{s=1}^{\rho^{U_g}} \mathbf{Z}_{ls}^{U_g} \boldsymbol{\Xi}_l^{U_g} \boldsymbol{\Phi}_s^{U_g} + \boldsymbol{\varepsilon} \\ &= \boldsymbol{\mu} + \sum_{g=1}^G \sum_{s=1}^{\rho^{U_g}} \mathbf{Z}_{\cdot s}^{U_g} \boldsymbol{\Xi}^{U_g} \boldsymbol{\Phi}_s^{U_g} + \boldsymbol{\varepsilon}, \end{aligned} \quad (3.5)$$

where $\Xi^{U_g} = \left[\xi_{lk}^{U_g} \right]_{l=1, \dots, L^{U_g}, k=1, \dots, N^{U_g}}$ with rows $\Xi_l^{U_g}$, $\mathbf{Z}_{\cdot s}^{U_g}$ contains columns $\mathbf{Z}_{ls}^{U_g}$, $l = 1, \dots, L^{U_g}$, and $\Phi_s^{U_g} = \left[\phi_{ks}^{U_g}(t) \right]_{k=1, \dots, N^{U_g}, t \in \mathcal{D}}$, $r = 1, \dots, \rho^{U_g}$ and $g = 1, \dots, G$. Model (3.5) is a scalar linear mixed model in which the random effects correspond to the random basis weights $\xi_{lk}^{U_g}$, as perhaps more clearly seen using matrix vectorization (the vec operator) and the Kronecker product of matrices \otimes (e.g., Harville, 1997, Chapter 16),

$$\begin{aligned} \text{vec}(\mathbf{Y}) &\approx \text{vec}(\boldsymbol{\mu}) + \sum_{g=1}^G \left[\sum_{s=1}^{\rho^{U_g}} \left(\Phi_s^{U_g \top} \otimes \mathbf{Z}_{\cdot s}^{U_g} \right) \right] \text{vec}(\Xi^{U_g}) + \text{vec}(\boldsymbol{\varepsilon}) \\ &= \text{vec}(\boldsymbol{\mu}) + \mathbf{Z}_{\Phi} \boldsymbol{\xi} + \text{vec}(\boldsymbol{\varepsilon}), \end{aligned} \quad (3.6)$$

where $\mathbf{Z}_{\Phi} = \left[\sum_{s=1}^{\rho^{U_1}} \Phi_s^{U_1 \top} \otimes \mathbf{Z}_{\cdot s}^{U_1} \mid \dots \mid \sum_{s=1}^{\rho^{U_G}} \Phi_s^{U_G \top} \otimes \mathbf{Z}_{\cdot s}^{U_G} \right]$ and $\boldsymbol{\xi} = \left[\text{vec}(\Xi^{U_1})^\top \mid \dots \mid \text{vec}(\Xi^{U_G})^\top \right]^\top$.

3.3 Estimation

Estimation is conducted in four steps. First, the mean is estimated under a working independence assumption. Second, after demeaning the responses, the covariance structure is estimated. This is used to, third, obtain the estimated eigenfunctions and eigenvalues, which are then used to, fourth, predict the random basis weights.

3.3.1 Estimation of the mean structure

If the mean $\mu(t)$ does not depend on covariates or depends only on a discrete variable k , i.e., is a group-specific mean function $\mu_k(t)$ as it is in our application, we can estimate it by simply averaging curves $Y_i(t)$ point-wise or point-wise within each group k .

If the mean depends more generally on scalar and/or functional covariates, the FAMM framework of Scheipl et al. (2015) can be used for estimation under a working independence assumption conditional on the mean. This framework allows for linear or smooth effects of scalar covariates x that are constant or varying over the interval \mathcal{T} , i.e., effects of the form $x\beta$, $x\beta(t)$, $f(x)$ or $f(t, x)$. In addition, linear or smooth effects of functional covariates $x(s)$ can be included of the form $\int x(s)\beta(t, s) ds$ or $\int f(x(s), s, t) ds$. Interactions between these terms for scalar and/or functional covariates are also possible. FAMM uses a tensor product basis representation of all model terms with marginal bases for the covariate effects and over \mathcal{T} . Smoothness of coefficients is achieved by using spline bases with an appropriate smoothness penalty, with the smoothing parameters estimated by restricted maximum likelihood (REML; Patterson and Thompson (1971); cf. Ruppert et al. (2003), Section 4.9). For full details, see Scheipl et al. (2015). The approach is implemented in the `pfpr` function of the R add-on package `refund` (Huang et al., 2016a).

3.3.2 Estimation of the covariance structure

For estimation of the covariance structure, we use a method of moments-based approach. We rewrite Model (3.4) as

$$\mathbf{Y} - \boldsymbol{\mu} = \sum_{g=1}^G \sum_{s=1}^{\rho^{U_g}} \mathbf{Z}_{\cdot s}^{U_g} \mathbf{U}_{g \cdot s},$$

where $\mathbf{U}_{g \cdot s} = (\mathbf{U}_{g1s}, \dots, \mathbf{U}_{gL^{U_g}s})$ and for ease of notation we absorbed $\boldsymbol{\varepsilon}$ into the last random effect, assuming this to be a smooth error term E_i for each curve $i = 1, \dots, n$ (or group-specific smooth error).

Denote the covariance matrix for \mathbf{U}_{gls} on the grid \mathcal{D} by $\mathbf{K}_{ss'}^{U_g}(\mathcal{D}) = [\text{Cov}[U_{gls}(t), U_{gls'}(t')]]_{t, t' \in \mathcal{D}}$. Then, using $\mathbb{E}(\mathbf{U}_{g \cdot s'} \otimes \mathbf{U}_{g \cdot s}) = \text{vec}(\mathbf{I}_{L^{U_g}}) \text{vec}(\mathbf{K}_{ss'}^{U_g})^\top$ for all g, s, s' and rules for the Kronecker product of matrices (Harville, 1997, Chapter 16), we obtain

$$\mathbb{E}[(\mathbf{Y} - \boldsymbol{\mu}) \otimes (\mathbf{Y} - \boldsymbol{\mu})] = \sum_{g=1}^G \sum_{s=1}^{\rho^{U_g}} \sum_{s'=1}^{\rho^{U_g}} \text{vec}(\mathbf{Z}_{\cdot s}^{U_g} \mathbf{Z}_{\cdot s'}^{U_g \top}) \text{vec}(\mathbf{K}_{ss'}^{U_g})^\top =: \mathbf{X}_Z \boldsymbol{\beta}_K,$$

with

$$\begin{aligned} \mathbf{X}_Z &= \left[\text{vec}(\mathbf{Z}_{\cdot 1}^{U_1} \mathbf{Z}_{\cdot 1}^{U_1 \top}) \mid \dots \mid \text{vec}(\mathbf{Z}_{\cdot 1}^{U_1} \mathbf{Z}_{\cdot \rho^{U_1}}^{U_1 \top}) \mid \dots \mid \text{vec}(\mathbf{Z}_{\cdot \rho^{U_1}}^{U_1} \mathbf{Z}_{\cdot \rho^{U_1}}^{U_1 \top}) \mid \dots \mid \text{vec}(\mathbf{Z}_{\cdot \rho^{U_g}}^{U_g} \mathbf{Z}_{\cdot \rho^{U_g}}^{U_g \top}) \right], \\ \boldsymbol{\beta}_K &= \left[\text{vec}(\mathbf{K}_{11}^{U_1}) \mid \dots \mid \text{vec}(\mathbf{K}_{1\rho^{U_1}}^{U_1}) \mid \dots \mid \text{vec}(\mathbf{K}_{\rho^{U_1}\rho^{U_1}}^{U_1}) \mid \dots \mid \text{vec}(\mathbf{K}_{\rho^{U_g}\rho^{U_g}}^{U_g}) \right]^\top. \end{aligned}$$

We can then estimate the covariance matrices contained in $\boldsymbol{\beta}_K$ based on least squares

$$\widehat{\boldsymbol{\beta}}_K = (\mathbf{X}_Z^\top \mathbf{X}_Z)^{-1} \mathbf{X}_Z^\top [(\mathbf{Y} - \widehat{\boldsymbol{\mu}}) \otimes (\mathbf{Y} - \widehat{\boldsymbol{\mu}})]$$

after replacing $\boldsymbol{\mu}$ by the estimate $\widehat{\boldsymbol{\mu}}$ from Section 3.3.1. $\widehat{\boldsymbol{\beta}}_K$ is a $[\sum_{g=1}^G (\rho^{U_g})^2] \times D^2$ matrix, with the columns corresponding to the covariance estimates for each (t, t') combination, $t, t' \in \mathcal{D}$.

Using rules for Kronecker products and vec notation, we can write $\widehat{\boldsymbol{\beta}}_K$ in a representation that allows the computation of all covariance estimates with $\mathcal{O}(nD^2)$ computational effort (see Appendix B). This effort is due to the computation of the raw covariance of \mathbf{Y} . If D is large, computation of the $D \times D$ covariance matrices needs to be avoided and an approach similar to Zipunnikov et al. (2011, 2014), developed for special cases of our model, could likely be extended to our more general setting, but is beyond the scope of this chapter.

After obtaining the raw covariance estimates, we can smooth each $\mathbf{K}_{ss'}^{U_g}$ separately using a bivariate smoother in t and t' . Any bivariate smoother could be used for this step in principle. We use penalized splines, with the smoothing parameter estimated using REML in a mixed model framework (e.g., Ruppert et al., 2003). REML estimation has been found to be more stable than generalized

cross-validation for the estimation of smoothing parameters (Reiss and Ogden, 2009) in the sense of being less prone to multiple optima, and more robust under misspecification of the correlation structure than mean squared error minimizers such as the Akaike information criterion (Krivobokova and Kauermann, 2007).

For the covariance of the error terms, we can separate $\text{Cov}[E_i(t) + \varepsilon_i(t), E_i(t') + \varepsilon_i(t')]$ into $\text{Cov}[E_i(t), E_i(t')]$ and $\text{Cov}[\varepsilon_i(t), \varepsilon_i(t')] = \sigma^2 \delta_{tt'}$, where $\delta_{tt'} = 1$ if $t = t'$ and $\delta_{tt'} = 0$ otherwise. As $\text{Cov}[\varepsilon_i(t), \varepsilon_i(t')]$ induces an offset term σ^2 on the diagonal $t = t'$ of the bivariate surface $\text{Cov}[E_i(t) + \varepsilon_i(t), E_i(t') + \varepsilon_i(t')]$, we smooth this surface leaving out the diagonal to obtain an estimate of $\text{Cov}[E_i(t), E_i(t')]$, then estimate σ^2 from the average difference of the raw and smoothed diagonal values (Staniswalis and Lee, 1998; Yao et al., 2005). For group-specific smooth errors, smoothing without the diagonal of the error covariance can be done for each group separately.

Smooth covariance estimates are not guaranteed to be positive semi-definite. We impose positive semi-definiteness by trimming eigenfunction-eigenvalue pairs corresponding to negative eigenvalues. This method has been found to increase the L^2 -accuracy (Hall et al., 2008) and to work well in practice (e.g., Yao et al., 2003; Greven et al., 2010).

Our implementation of the covariance smoothing is based on fast and robust available routines in the `mgcv` package in R (Wood, 2006, 2011; R Core Team, 2016). If data sets are large, the `bam` function instead of the `gam` function in the `mgcv` package can be used, which increases speed and reduces memory need. Building on existing software allows us to benefit from established methods and flexible implementations and to take advantage of future extensions and improvements.

3.3.3 Estimation of the eigenfunctions and eigenvalues

Estimates of the eigenfunctions and eigenvalues are obtained using spectral decompositions of the covariances. We evaluate the estimated covariances on a fine equidistant grid $\tilde{\mathcal{D}} = \{t_1, \dots, t_{\tilde{D}}\}$ in \mathcal{T} . We obtain estimates $\hat{\phi}_k^{U_g} = [\hat{\phi}_{ks}^{U_g}(t)]_{s=1, \dots, \rho^{U_g}, t \in \tilde{\mathcal{D}}} \in \mathbb{R}^{\tilde{D}\rho^{U_g}}$ and $\hat{\nu}_k^{U_g}$, $k = 1, \dots, \tilde{D}\rho^{U_g}$, from a spectral decomposition of

$$\widehat{\mathbf{K}}^{U_g}(\tilde{\mathcal{D}}) = \begin{bmatrix} \widehat{\mathbf{K}}_{11}^{U_g}(\tilde{\mathcal{D}}) & \dots & \widehat{\mathbf{K}}_{1\rho^{U_g}}^{U_g}(\tilde{\mathcal{D}}) \\ \vdots & & \vdots \\ \widehat{\mathbf{K}}_{\rho^{U_g}1}^{U_g}(\tilde{\mathcal{D}}) & \dots & \widehat{\mathbf{K}}_{\rho^{U_g}\rho^{U_g}}^{U_g}(\tilde{\mathcal{D}}) \end{bmatrix} = \sum_{k=1}^{\tilde{D}\rho^{U_g}} \hat{\nu}_k^{U_g} \hat{\phi}_k^{U_g} \hat{\phi}_k^{U_g \top},$$

where $\widehat{\mathbf{K}}_{ss'}^{U_g}(\tilde{\mathcal{D}}) = [\widehat{\text{Cov}}[U_{gls}(t), U_{gls'}(t')]]_{t, t' \in \tilde{\mathcal{D}}}$ for all g, l, s, s' . To ensure orthonormality with respect to the additive inner product (3.3), the approximated eigenfunctions are rescaled (see, e.g., Chapter 4 of this thesis).

Suitable truncation lags N^{U_g} , $g = 1, \dots, G$, could be chosen using testing (Crainiceanu and Ruppert, 2004; Greven et al., 2008) or model selection (Vaida and Blanchard, 2005; Greven and Kneib, 2010) for random effects in the linear mixed model (3.6). We choose a simple approach based

on variance explained (cf. Greven et al., 2010), ordering the $\widehat{\nu}_k^{U_g}$, $g = 1, \dots, G$, by decreasing size and selecting components until

$$\left(\sum_{g=1}^G \sum_{k=1}^{N^{U_g}} \widehat{\nu}_k^{U_g} + \widehat{\sigma}^2 |\mathcal{T}| \right) / \left(\sum_{g=1}^G \sum_{k=1}^{\widetilde{D}\rho^{U_g}} \widehat{\nu}_k^{U_g} + \widehat{\sigma}^2 |\mathcal{T}| \right) \geq L$$

for a pre-specified L , e.g., $L = 0.95$.

3.3.4 Prediction of the random basis weights

After choosing the truncation lags, we base prediction of the basis weights on the linear mixed model (3.6), replacing the mean function, the error variance, the eigenfunctions and eigenvalues by their estimates. We obtain predictions for the basis weights collected in $\boldsymbol{\xi}$ based on best linear unbiased predictors (BLUPs)

$$\boldsymbol{\xi} = \left(\mathbf{Z}_{\Phi}^{\top} \mathbf{Z}_{\Phi} + \sigma^2 \mathbf{G}^{-1} \right)^{-1} \mathbf{Z}_{\Phi}^{\top} \text{vec}(\mathbf{Y} - \boldsymbol{\mu}).$$

Here, we use $\text{Cov}[\text{vec}(\boldsymbol{\varepsilon})] = \sigma^2 \mathbf{I}_{nD}$ and \mathbf{G} denotes $\text{Cov}[\text{vec}(\boldsymbol{\xi})]$, a diagonal matrix with blocks $\text{diag} \left[\left(\nu_1^{U_g}, \dots, \nu_{N^{U_g}}^{U_g} \right) \otimes \mathbf{I}_{L^{U_g}} \right]$, $g = 1, \dots, G$. Predictions $\widehat{\boldsymbol{\xi}}$ are obtained as empirical BLUPs (EBLUPs), replacing $\boldsymbol{\mu}$, σ^2 , $\nu_k^{U_g}$ and $\Phi_s^{U_g}$, for all g, k, s , by their estimates.

Again, we can make use of the Kronecker product structure of our matrices and derive a representation of $\widehat{\boldsymbol{\xi}}$ (see Appendix B) that has computational effort of $\mathcal{O} \left[nD + \left(\sum_{g=1}^G N^{U_g} L^{U_g} \right)^3 \right]$, or at most $\mathcal{O}(nD + n^3)$. The first component is linear in n and D , while the second component is independent of the functional part of our model and equal to the effort of computing $\sum_{g=1}^G N^{U_g} L^{U_g}$ BLUPs in a scalar linear mixed model.

3.3.5 Extension to missing values

Now consider the case where some of the observations $Y_i(t)$, $t \in \mathcal{D}$, are missing. Denote the subset of t with some missing values by $\mathcal{M} \subset \mathcal{D}$, and the corresponding values of i by $\mathcal{M}_t \subset \{1, \dots, n\}$.

For t and/or t' in \mathcal{M} , we have

$$\mathbb{E} \left\{ [\mathbf{Y}^{-i}(t) - \boldsymbol{\mu}^{-i}(t)] \otimes [\mathbf{Y}^{-i}(t') - \boldsymbol{\mu}^{-i}(t')] \right\} = \sum_{g=1}^G \sum_{s=1}^{\rho^{U_g}} \sum_{s'=1}^{\rho^{U_g}} \text{vec} \left[\mathbf{Z}_{.s}^{U_g^{-i}} \left(\mathbf{Z}_{.s'}^{U_g^{-i}} \right)^{\top} \right] \text{vec} \left[\mathbf{K}_{ss'}^{U_g}(t, t') \right]^{\top},$$

where $\mathbf{Y}^{-i}(t)$ denotes the column in \mathbf{Y} corresponding to t , after deleting the rows corresponding to $i \in \mathcal{M}_t$, $\boldsymbol{\mu}^{-i}(t)$ is constructed analogously, $\mathbf{Z}_{.s}^{U_g^{-i}}$ results from $\mathbf{Z}_{.s}^{U_g}$ after deletion of the corresponding rows, and $\mathbf{K}_{ss'}^{U_g}(t, t')$ denotes the column in $\mathbf{K}_{ss'}^{U_g}(\mathcal{D})$ corresponding to (t, t') . Estimation of $\mathbf{K}_{ss'}^{U_g}(t, t')$ can then proceed as before using reduced design matrices and responses.

This approach works well for a relatively small proportion of missing values. If the proportion of missings is large, computational effort increases, as the equality of design matrices $\text{vec} \left(\mathbf{Z}_{.s}^{U_g} \mathbf{Z}_{.s'}^{U_g \top} \right)$ across \mathcal{D} is lost and less calculations can be shared. Additionally, estimation may deteriorate if the number of available observations per t decreases. In such cases, it would be more suitable to develop approaches aimed at sparse data and to borrow strength across neighboring t (cf. Yao et al., 2005 for the case of i. i. d. functions and Chapters 4 and 5 of this thesis for the case of correlated functional data). This is, however, beyond the scope of this chapter, which uses the (near-)grid structure to increase computational efficiency.

3.4 Application to tissue spectroscopy

3.4.1 Background and scientific questions

Oral and maxillofacial surgery has to deal with the complex anatomy in the head and neck region. Therefore, lasers have become the preferred surgical technique as they provide several advantages over traditional scalpels. They allow the surgeons to cut biological tissue with high precision and minimal trauma (see, e.g., Engelhardt et al., 2014).

One drawback however is that during surgery the surgeons do not receive sufficient information about the type of tissue being ablated at the bottom of the laser cut. Thus, the usage of laser scalpels is accompanied by a high risk of iatrogenic damage due to a lack of haptic feedback. This is very dangerous especially as the head and neck region include mayor sensory and motor nerves which might be unintentionally damaged during surgery (see, e.g., Stelzle et al., 2011).

One idea to overcome this inherent risk of nerve damage in laser surgery is to use statistical methods to train an algorithm that classifies the tissue type during surgery. Stelzle et al. (2011) propose an algorithm that uses diffuse reflectance spectra measured with a backscattering probe. The light applied is absorbed or scattered, depending on the optical properties of each tissue type. Diffuse reflectance spectra provide a simple approach for intra-operative tissue differentiation. The surgeon should either automatically get a warning before delicate tissue is damaged or the laser cut should be automatically stopped in time.

The spectroscopy data analyzed in this chapter were provided by Stelzle et al. (2011). They investigated diffuse reflectance spectra to reliably differentiate between the four tissue types cortical bone, nerves, salivary glands, and cancellous bone. Special emphasis was placed on the identification of nerve tissue in order to reduce the risk of iatrogenic nerve damage in laser surgery. The data consist of 8640 diffuse reflectance spectra evaluated at 1150 wavelengths of 350-650 nm range (0.26 nm wavelength resolution). The tissue samples were taken from 12 bisected ex vivo domestic pig heads. For each of the 12 pigs, 6 different spots were chosen per tissue with a distance of 0.5 cm from each other. Per spot, 30 diffuse reflectance spectra were acquired. In total, 2160 spectra were recorded per tissue type. The diffuse reflectance spectra for the four tissue types are shown in Figure 3.1.

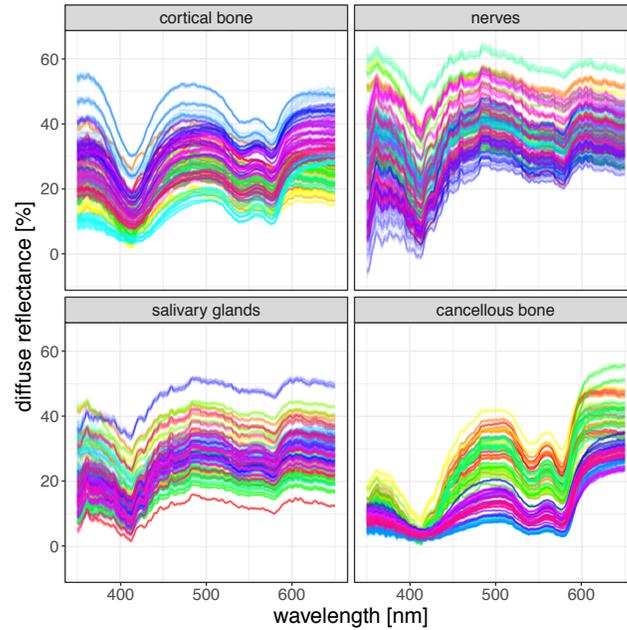


Figure 3.1: Non-standardized diffuse reflectance spectra for cortical bone (upper left), nerves (upper right), salivary glands (bottom left), and cancellous bone (bottom right) colored by pig.

We can see in Figure 3.1 that the spectra of the four tissue types have a similar course. Nevertheless, they differ in various aspects. Differences can be found in the degree of smoothness and the intra-tissue variability is different for the four tissue types. For salivary glands (bottom left) for instance, we mainly observe a vertical shift of the spectra whereas for cancellous bone (bottom right), variation changes along wavelengths as well. Nerves spectra show the greatest variability leading to difficult nerve detection. The spectra of nerves and salivary glands are very similar, which is why they are difficult to reliably differentiate (see also Stelzle et al., 2011).

There are multiple sources of variability in the spectroscopy data. First, there is variability between the different tissue types. Second, variation is also induced by measurements on different pigs. Third, there is variability between the measurements at different spots within each pig. Note that in our data, the spots differ between pigs. Fourth, the repeated observations at each spot induce variability. And fifth, there may be additional measurement error.

In order to illustrate the variability between the tissue types, we depict the point-wise mean curves per tissue as well as the overall mean curve in the left plot of Figure 3.2 and the point-wise variance curves per tissue as well as the overall variance curve in the right plot of Figure 3.2. As can already be expected from Figure 3.1, the form of the tissue-specific means is similar for all tissue types and the mean curves mainly differ in their absolute values. The tissue-specific variances however are very different.

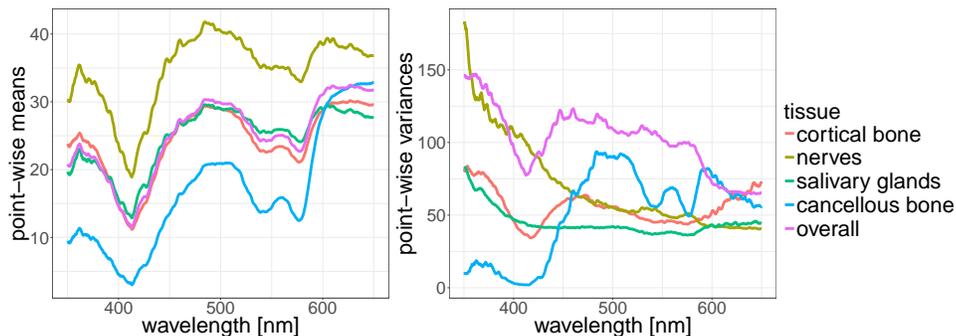


Figure 3.2: Left: Point-wise mean curves per tissue and overall mean curve. Right: Point-wise variance curves per tissue and overall variance curve.

For tissue classification, the important source of variation in the data is the tissue type (shown in Figure 3.2). Classification can, however, be improved by further breaking down the variability into the variability induced by pigs, spots, and repetitions. We propose a method that allows us to decompose the variability in our data and to take advantage of the information on all sources of variability.

3.4.2 Application of the functional linear mixed model to the spectroscopy data

Stelzle et al. (2011) conduct a principal component analysis to reduce the dimensionality of the data followed by a linear discriminant analysis based on (manually) selected principal components for classification. All data are pooled and the principal components are computed from the overall covariance of the spectra.

We propose to decompose the variability in the spectroscopy data by applying a FPC-based FLMM. We obtain the main modes of variation for each source of variability separately. This allows us to subsequently perform the classification described in Section 3.4.3.

The diffuse reflectance spectra are nearly equidistantly measured. For simplicity, we linearly interpolate the spectra and obtain a fully equidistant grid of length $D = 1150$, which is also used as evaluation grid \tilde{D} for the covariances.

We then perform a nested leave-one-pig-out cross-validation yielding 12 training and testing data sets which is described in detail in Appendix B. We fit the FLMM (3.7) introduced below to the training data to train our classification algorithm described in Section 3.4.3. The remaining data serve for testing. We repeat this for all 12 parts and thus obtain predictions of class membership for all of our 8640 reflectance spectra.

Before applying our model to the training data, we center each spectrum by its mean and divide it by its standard deviation to work out the main features of each tissue type. As the spectra are referenced, the absolute value should not play a role and standardization removes the dependence on the optical set-up (compare Fuchs et al., 2015).

The FLMM, which we apply to our training data, is a hierarchical model taking into account that observations are nested within spots, spots are nested within pigs and for each pig we have measurements for each of the four tissue types. As we only consider four tissue types, we include a fixed effect for tissue type in our model. The remaining hierarchy levels are accounted for by including functional random intercepts. The model we apply is of the following form

$$Y_{\tau pso}(t) = \mu_{\tau}(t) + B_{\tau p}(t) + C_{\tau ps}(t) + E_{\tau pso}(t) + \varepsilon_{\tau pso}(t), \quad (3.7)$$

with $\tau = 1, \dots, 4$ (tissue types), $p = 1, \dots, 12$ (pigs), $s = 1, \dots, 6$ (spots), $o = 1, \dots, 30$ (observations), $t \in \mathcal{T} = [350 \text{ nm}; 650 \text{ nm}]$. $Y_{\tau pso}(t)$ represents the reflectance spectrum of tissue type τ , pig p , at spot s , and observation o at wavelength t . $\mu_{\tau}(t)$ is the fixed effect for tissue type. $B_{\tau p}(t)$ is a tissue-specific functional random intercept for pigs and $C_{\tau ps}(t)$ represents a tissue-specific functional random intercept for spots. $E_{\tau pso}(t)$ and $\varepsilon_{\tau pso}(t)$ are a smooth error term and white noise measurement error, respectively.

We assume that $B_{\tau p}(t)$, $C_{\tau ps}(t)$, and $E_{\tau pso}(t)$ are mutually uncorrelated random processes with zero mean. We further assume that $B_{\tau p}(t)$ and $C_{\tau ps}(t)$ have tissue-specific covariances $K_{\tau}^B(t, t')$ and $K_{\tau}^C(t, t')$, $t, t' \in \mathcal{T}$, respectively. The covariance of the smooth error term is denoted by $K^E(t, t')$ and is not tissue-specific, as we found the observation-specific variability to be small and not to depend on tissue in preliminary analyses. Model (3.7) is a very general model as we allow that the covariances of the functional random intercepts are different for each tissue type. This assumption can be motivated by Figures 3.1 and 3.2 where we can see that the intra-tissue variation differs between tissue types. Note that we also compared with a model without tissue-specific covariances. As expected, we obtained much worse classification results.

In our model, the number of covariances to be estimated is nine, as we have $K_1^B(t, t'), \dots, K_4^B(t, t')$, $K_1^C(t, t'), \dots, K_4^C(t, t')$, and $K^E(t, t')$. In each training set, the numbers of independent copies per level equal $L^{U_1} = L^{U_2} = L^{U_3} = L^{U_4} = 11$, $L^{U_5} = L^{U_6} = L^{U_7} = L^{U_8} = 11 \cdot 6 = 66$, $L^{U_9} = 4 \cdot 11 \cdot 6 \cdot 30 = 7920$. Specifying one functional random effect for each level, $\rho^{U_g} = 1$, $g = 1, \dots, G = 9$, yields $q = \sum_{g=1}^G L^{U_g} \rho^{U_g} = 8228$ functional random effects in total. As the reflectance spectra are relatively smooth, we do not smooth the covariance estimates obtained by the method of moments approach to avoid smoothing out relevant features. Therefore, the error variance, $\text{Var}[\varepsilon_{\tau pso}(t)] = \sigma^2$, solely captures the left out variability due to the truncation of the infinite Karhunen-Loève expansions.

The fixed effect for tissue type $\mu_{\tau}(t)$ is estimated by a method of moments estimator which is equivalent to centering the spectra by tissue.

Our FPC-based approach allows for an explicit variance decomposition and gives us interpretable measures of where in the spectrum variability occurs between pigs, spots, and repetitions. Moreover, the FPC weights can be used for further analyses. Classification is only one possible application of our approach. Rather than a black box classification, the model can also explain where the relevant information in the spectra is contained. This may be helpful both for the medical experts working with these spectra as well as for statisticians when investigating ways to even further improve classification.

To illustrate this, we fit Model (3.7) again to all data (not divided into training and test data) after standardization of the spectra as described above. For each tissue type, the estimated first principal component of $B_{\tau p}(t)$, the tissue-specific functional random intercept for pigs, is depicted in Figure 3.3. For ease of interpretation, we show the effect of adding and of subtracting the estimated principal components multiplied by the square root of the respective eigenvalue to the tissue-specific mean.

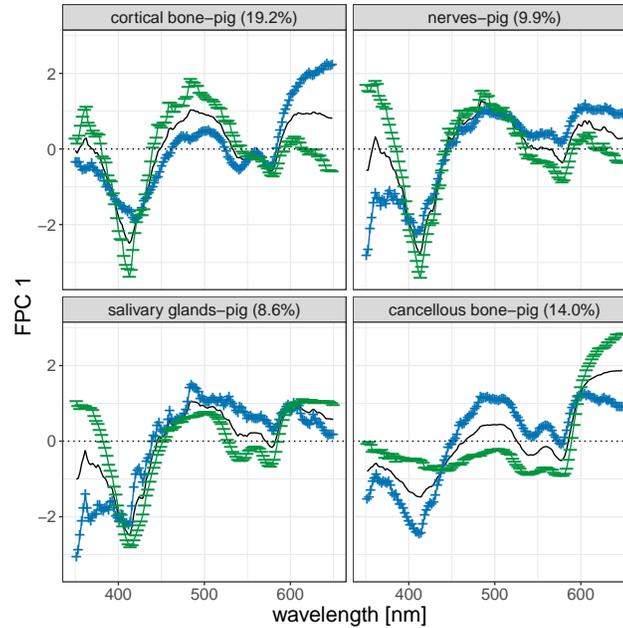


Figure 3.3: Tissue-specific means (solid black) plus (+) and minus (−) a suitable multiple of the first principal component of $B_{\tau p}(t)$ for cortical bone (upper left), nerves (upper right), salivary glands (bottom left), cancellous bone (bottom right). The respective amount of explained variability is given in brackets.

The amount of variability explained by the pig effect differs between tissue types. The first principal component of $B_{\tau p}(t)$ explains most variability in the data for cortical bone (19.2%) followed by cancellous bone (14%). Less variability is explained for nerves (9.9%) and salivary glands (8.6%). For smaller wavelengths (350-450 nm), the effect is quite similar for cortical bone, nerves, and salivary glands, where pigs with negative basis weights for this component have more extreme maxima and minima. Further similarities between the effect for nerves and that for salivary glands can be found around 530-570 nm. For cancellous bone, we see a vertical shift of the mean curve in the range of around 450-550 nm. For higher wavelengths (> 600 nm), the effect becomes larger for all four tissue types.

In order to obtain the full variance decomposition shown in Table 3.1, we fit a model similar to Model (3.7) to all data with the difference that the fixed effect $\mu_{\tau}(t)$ is replaced by a random intercept for tissue type. Again, the spectra are standardized beforehand and we set $L = 0.99999$ as this value is chosen most frequently in our nested cross-validation. The variance decomposition

highlights the importance of accounting for the different sources of variability as only 39.47% of the overall variability are explained by tissue type but 52.75% are induced by pigs.

variability source	tissue type	pig	spot	observation	error variance
variance explained [in%]	39.47	52.75	7.65	0.12	2.22×10^{-08}

Table 3.1: Variance decomposition for an FLMM with a functional random intercept for tissue type with pre-specified $L = 0.99999$. The model is fit to all reflectance spectra.

3.4.3 Classification of the tissue types

One application of our FPC-based FLMM is classification. In order to achieve intra-operative tissue differentiation for the spectroscopy data, we use the estimates obtained from fitting the FLMM (3.7) to the training data to compute estimates for the (tissue-specific) covariances reconstructed from the truncated FPCs and the corresponding eigenvalues. Denote the covariance matrices by $\mathbf{K}_\tau^B(\mathcal{D}) = [K_\tau^B(t, t')]_{t, t' \in \mathcal{D}}$, $\mathbf{K}_\tau^C(\mathcal{D}) = [K_\tau^C(t, t')]_{t, t' \in \mathcal{D}}$, and $\mathbf{K}^E(\mathcal{D}) = [K^E(t, t')]_{t, t' \in \mathcal{D}}$, respectively. The covariance matrix of each tissue type comprises the information of all sources of variability and is of the following form

$$\mathbf{K}_\tau := \mathbf{K}_\tau^B + \mathbf{K}_\tau^C + \mathbf{K}^E + \sigma^2 \mathbf{I}_D, \quad \tau = 1, \dots, 4,$$

where we omitted the argument \mathcal{D} for simplicity.

The algorithm we use for classification of the test data into one of the four tissue types adapts the work of Zhu et al. (2012) who introduce a robust approach to functional classification for a wavelet-based Bayesian FLMM. Using a wavelet transformation, their model is more suited for spiky functional data or functional data with local features whereas our FPC-based FLMM is well-suited when the data are rather smooth as is the case for the tissue spectroscopy data being considered. Moreover, the explicit variance decomposition obtained by our FPC-based approach can help to understand the different sources of variability in the data. In contrast to the approach of Zhu et al. (2012), our approach does not require that the data are sampled on an equally spaced grid.

Let \mathbf{t} denote the vector of observed wavelengths, $y^0(\mathbf{t})$ is a reflectance spectrum in the test set and c^0 its unknown tissue type. For each of the four tissue types, we can define the marginal log-likelihood under a Gaussian assumption as

$$\begin{aligned} \log \{ f [y^0(\mathbf{t}) | c^0 = \tau, \mu_\tau, \mathbf{K}_\tau] \} &= \text{const} - \frac{1}{2} \log [\det (\mathbf{K}_\tau)] \\ &- \frac{1}{2} [y^0(\mathbf{t}) - \mu_\tau(\mathbf{t})]^\top \mathbf{K}_\tau^{-1} [y^0(\mathbf{t}) - \mu_\tau(\mathbf{t})], \end{aligned} \quad (3.8)$$

for $\tau = 1, \dots, 4$, where const is a constant and $\det (\mathbf{K}_\tau)$ denotes the determinant of the tissue-specific covariance matrix \mathbf{K}_τ . Note that the straightforward computation of the determinant as the product of the eigenvalues of \mathbf{K}_τ may be numerically unstable in case of small eigenvalues. We therefore use

the properties of the logarithmic function and compute the sum of the logarithmized eigenvalues to directly obtain $\log[\det(\mathbf{K}_\tau)]$.

As the functional random effects for a future laser surgery are not available, we use the marginal log-likelihood where the functional random effects are integrated out. In other applications where new observations may be obtained from the same subjects, the conditional likelihood could be used to improve predictions.

After standardization (in analogy to the standardization of the training data), we evaluate the four marginal log-likelihoods for the test data. Each spectrum of the test data is then classified with respect to the highest log-likelihood, i.e., the log-likelihood for the test data given that the data originate from tissue type τ . We repeat this procedure for each test data set and obtain class designations for all of our 8640 reflectance spectra.

When multiple repetitions per spot are available during surgery, it is possible to combine them to possibly improve classification while accounting for their correlation. Then, the marginal log-likelihood (3.8) is replaced by a joint log-likelihood. We additionally perform such a classification by dividing each of the 30 replications per spot in the test data into 15 pairs of adjacent observations and using the joint log-likelihood of the pair during classification. The tissue-specific covariances then have the following form

$$[\text{Cov}[Y_{\tau ps o}(t), Y_{\tau ps o'}(t')]]_{t, t' \in \mathcal{T}} = \begin{cases} \mathbf{K}_\tau^B + \mathbf{K}_\tau^C + \mathbf{K}^E + \sigma^2 \mathbf{I}_D, & \text{if } o = o', \\ \mathbf{K}_\tau^B + \mathbf{K}_\tau^C, & \text{otherwise.} \end{cases} \quad (3.9)$$

The results for combined and non-combined classification are summarized in Table 3.2. For comparison, we show the classification results of Stelzle et al. (2011). In addition, Engelhardt et al. (2014) compare different classification algorithms for a similar data set and conclude that penalized discriminant analysis (PDA) works best. The authors, however, average spectra within each pig over all spots and observations per spot. As such averaged spectra will never be available for classification in practice, we try to apply their method as closely as possible without averaging and for this use a PDA on the original spectra. We select the smoothing parameter for the PDA via a nested cross-validation of the same structure as for the selection of L to avoid over-fitting for both methods.

In almost all cases, class designations are highly improved by applying our FLMM compared to the method of Stelzle et al. (2011). Especially for nerves, which is the tissue of main interest, only two (for combined six) of the 2160 reflectance spectra are misclassified as salivary glands compared to 620 misclassifications of nerves using the approach of Stelzle et al. (2011). Merely the detection of cancellous bone slightly diminishes as more misclassifications in favor of cortical bone are made. With the method of Stelzle et al. (2011), a total of 1870 (21.64%) spectra are misclassified whereas a total of 465 (5.38%) misclassifications occur with the proposed method and only 436 (5.05%) when two replications are combined for classification. The total misclassification rate also compares favorably to the results using PDA with a total of 481 (5.57%) misclassified spectra.

true \ predicted	cortical bone		nerves		salivary glands		cancellous bone	
	FLMM							
cortical bone	1988	(92.04%)	0	(0.00%)	3	(0.14%)	169	(7.82%)
nerves	0	(0.00%)	2158	(99.91%)	2	(0.09%)	0	(0.00%)
salivary glands	35	(1.62%)	180	(8.33%)	1945	(90.05%)	0	(0.00%)
cancellous bone	76	(3.52%)	0	(0.00%)	0	(0.00%)	2084	(96.48%)
FLMM combined								
cortical bone	1978	(91.57%)	0	(0.00%)	0	(0.00%)	182	(8.43%)
nerves	0	(0.00%)	2154	(99.72%)	6	(0.28%)	0	(0.00%)
salivary glands	16	(0.74%)	180	(8.33%)	1964	(90.93%)	0	(0.00%)
cancellous bone	52	(2.41%)	0	(0.00%)	0	(0.00%)	2108	(97.59%)
Stelzle								
cortical bone	1430	(66.20%)	204	(9.44%)	337	(15.60%)	189	(9.17%)
nerves	270	(12.50)	1540	(71.30%)	142	(6.57%)	208	(9.63%)
salivary glands	256	(11.85%)	234	(10.83%)	1670	(77.31%)	0	(0.00%)
cancellous bone	30	(1.39%)	0	(0.00%)	0	(0.00%)	2130	(98.61%)
PDA								
cortical bone	1960	(90.74%)	0	(0.00%)	2	(15.60%)	198	(8.75%)
nerves	2	(0.09%)	2158	(99.91%)	0	(0.00%)	0	(0.00%)
salivary glands	15	(0.69%)	180	(8.33%)	1965	(90.97%)	0	(0.00%)
cancellous bone	82	(3.80%)	0	(0.00%)	2	(0.09%)	2076	(96.11%)

Table 3.2: Classification results based on the FLMM (3.7) (block 1), with combined replications (block 2), based on Stelzle et al. (2011) (block 3), and based on PDA as in Engelhardt et al. (2014) (block 4). Shown are the numbers of classifications for each tissue type as well as the proportion of classifications per true class.

Overall, we conclude that accounting for the different sources of variation improves classification. In our application, measurement error as captured by the observation-specific variation is relatively small and classification thus does not improve dramatically when using more than one repetition. However, in other settings with higher measurement error being able to use more than one spectrum for classification is a nice feature that can help in further decreasing misclassification. When measurement error variability is tissue-specific, we expect that having repeated observations would be particularly useful in providing another source of information on tissue membership.

3.5 Discussion and outlook

In this chapter, we consider very general functional linear mixed models for correlated functional data. In particular, the models accommodate hierarchical and/or crossed functional random effects, correlated functional random effects such as correlated random intercepts and slopes, and group-specific covariances of functional random effects that allow for different intra-group variations between groups. The functional random effects are expanded in parsimonious functional principal component bases that are estimated from the data, also yielding an explicit variance decomposition. The estimation of the functional principal components had so far only been described for models with less general

correlation structures than the ones we consider here. We propose a computationally efficient method of moments approach for the estimation of the covariances of the functional random effects that makes use of rules for the Kronecker product of matrices and the vec operator. The functional principal component weights are predicted as empirical best linear unbiased predictors (EBLUPs) of the resulting linear mixed model. We apply our method to a study on tissue spectroscopy data that aims at improving intra-operative tissue differentiation. We show that our method can provide a better understanding of the different components contributing to the variability in the data and that accounting for the different sources of variation improves classification results. We make the proposed method accessible to users by providing freely available software in the R add-on package `denseFLMM` (Greven and Cederbaum, 2017).

This chapter opens up a number of interesting directions for future research. A first direction concerns the extension of our approach to estimate the covariances of irregularly or sparsely sampled functional data. In Section 3.3.5, we briefly discuss the extension of our method to a relatively small proportion of missing values. To precisely state how small the proportion needs to be depending on the complexity of the data would require further investigation. For curves that are sampled on curve-specific or even sparse grids, the current approach to first estimate the raw covariances which are then smoothed would not work (well) and borrowing strength across neighboring observation points would be necessary (see, e.g., Yao et al., 2005). Extensions to correlated functional data that are irregularly or sparsely sampled are discussed in the following two chapters. Besides the possibility to accommodate irregularly and sparsely sampled data, direct smoothing without previously estimating the raw covariances of the functional random effects as least squares estimates would prevent potential undesirable effects caused by this two-step procedure. Whether and to what extent smoothing the previously obtained covariance estimates affects the properties of the estimated covariances remains to be explored.

So far, our method focuses on point estimates for all model components. A further direction of future research thus concerns the consideration of uncertainty. One idea is to embed the estimated FPCs in the framework of functional additive mixed models (FAMMs; Scheipl et al., 2015) who pursue a mixed model approach to obtain valid inference (for more details on the combination with the FAMM approach, see, e.g., Greven and Scheipl, 2017 and Chapter 4 of this thesis). The implementation of the FAMM approach in the R add-on package `refund` (Huang et al., 2016a), however, does not support correlated functional random effects. Note that the inference resulting from the combination with the FAMM approach is conditional on the FPCA and the obtained confidence bands are point-wise. For simpler models accommodating only crossed functional random intercepts, however, simulations in Chapter 4 of this thesis show a good coverage of the confidence bands obtained from the combination with FAMM for covariate effects in the mean function. Bootstrap-based confidence bands represent a possible alternative. Future research could explore extensions of the approach of Goldsmith et al. (2013) who propose a non-parametric bootstrap that accounts for the uncertainty in the FPC decomposition. Their approach is, however, restricted to uncorrelated functional observations and it remains unclear how their non-parametric bootstrap could be extended to, e.g., crossed

functional random effects. Moreover, further work is needed to investigate whether it is extendable to the very general functional linear mixed models while keeping computation costs low.

Further potential developments include taking advantage of the symmetry of the covariances and ensuring their positive semi-definiteness during estimation. The first could be addressed by building on the fast symmetric additive bivariate smoother proposed in Chapter 5 of this thesis which also applies to irregularly and sparsely sampled data. Finally, the extension of our method to higher-dimensional grids and to high-dimensional functional data, e.g., by building on ideas from Zipunnikov et al. (2011, 2014) would widen the scope of potential applications.

Acknowledgements

The authors thank Florian Stelzle and Werner Adler for providing us with the tissue spectroscopy data and for explaining their estimation procedure in more detail. Thanks also to Alexander Engelhardt for sharing his implementations with us. Sonja Greven and Jona Cederbaum were supported by the Emmy Noether grant GR 3793/1-1 from the German Research Foundation.

Chapter 4

Functional Linear Mixed Models for Unequal and Sparse Sampling Grids

Contributing manuscript

This chapter is based on the following paper:

Cederbaum, J., Pouplier, M., Hoole, P., Greven, S. (2016): Functional linear mixed models for irregularly or sparsely sampled data. *Statistical Modelling*, 16(1):67–88.

This is joint work with Marianne Pouplier (Department of Phonetics and Speech Processing, LMU Munich, Germany), Phil Hoole (Department of Phonetics and Speech Processing, LMU Munich, Germany), and Sonja Greven (Department of Statistics, LMU Munich, Germany). Sonja Greven initiated the extension of existing estimation approaches for functional linear mixed models by considering irregularly or sparsely sampled functional data and supervised the project. She had the idea to combine the estimation approach with the general framework of functional additive mixed models (Scheipl et al., 2015). The concrete estimation approach with all its details was developed by Jona Cederbaum in close cooperation with Sonja Greven. Jona Cederbaum implemented the approach in R (R Core Team, 2016) and made the implementation available to users in the R add-on package `sparseFLMM` (Cederbaum, 2016). She planned and conducted all simulations and analyzed the speech production data, which were kindly provided by Marianne Pouplier and Phil Hoole, who also gave background information on the data and formulated research questions of interest. Sonja Greven and Marianne Pouplier initiated the cooperation between statistics and phonetics, which was later mainly continued by Jona Cederbaum and Marianne Pouplier. Marianne Pouplier and Phil Hoole contributed to choosing a suitable pre-processing of the speech production data and helped with the interpretation of the results of the data analysis. Jona Cederbaum conducted the literature research and wrote the manuscript assisted by input from Sonja Greven. Marianne Pouplier and Phil Hoole contributed to the application section. All authors were involved in proofreading the manuscript. Jona Cederbaum prepared the response to the revision including all necessary changes in the manuscript with the support of all other authors.

Except for minor changes, mainly concerning spelling, notation, and added references to the recently released R add-on package `sparseFLMM`, this chapter and the paper Cederbaum et al. (2016) match.

Software

The analyses within this chapter were carried out using R versions 3.1.0 (2014-04-10) to 3.1.2 (2014-10-31) (R Core Team, 2016) on the two platforms `x86_64-pc-linux-gnu` (64-bit) and `x86_64-w64-mingw32/x64` (64-bit). The add-on packages `orthopolynom` (Novomestky, 2013, version 1.0-5), `mvt-norm` (Genz et al., 2014, version 0.9-99992), and `expm` (Goulet et al., 2017, version 0.99-1.1) were used for the generation of the simulated data. The add-on package `data.table` (Dowle and Srinivasan, 2015, versions 1.9.2 and 1.9.4) was used for fast and memory efficient data manipulation. For matrix operations, the add-on packages `Matrix` (Bates and Mächler, 2017, versions 1.1-3 and 1.1-4) and `MASS` (Venables and Ripley, 2002, versions 7.3-33 and 7.3-34) were used. For smoothing the mean and covariance functions, the add-on package `mgcv` (Wood, 2006, 2011, versions 1.7-29 and 1.8-3) was applied, which depends on the add-on package `nlme` (Pinheiro et al., 2016, attached versions 3.1-117 and 3.1-118). For the combination with the approach of functional additive mixed models of Scheipl et al. (2015) and for comparisons with the spline-based alternative, the add-on package `refundDevel` (Huang et al., 2016b, version 0.3-3) was employed. For parallelization of the simulation runs, the add-on packages `foreach` (Analytics and Weston, 2015b, version 1.4.2) and `doMC` (Analytics and Weston, 2015a, version 1.3.3) were used, the latter of which depends on the add-on package `iterators` (Analytics and Weston, 2015c, attached version 1.0.7).

4.1 Introduction

Advancements in technology allow today's scientists to collect an increasing amount of data consisting of functional observations rather than single data points. Most methods in functional data analysis (FDA; see, e.g., Ramsay and Silverman, 2005) assume that observations are *a*) independent and/or *b*) observed at a typically large number of the same (equidistant) observation points across curves. Linguistic research is only one of numerous fields in which the data often do not meet these strong requirements. Our motivating data come from a speech production study (Pouplier et al., 2014; Pouplier and Hoole, 2016) on assimilation, the phenomenon that the articulation of two consonants becomes more alike when they appear subsequently in spoken language. The data consist of audio recordings of nine speakers repeating the same 16 target words, including the two consonants of interest, each five times. The recorded acoustic signals during the duration of the two consonants were summarized by the phoneticians in a functional index over time (shown in Figure 4.1) varying between +1 and -1. Positive (negative) index values indicate proximity of the acoustic signal to a reference signal for the first (second) consonant of the target word. Thus, without assimilation, curves show a clear transition from strongly positive to strongly negative values. Assimilation can result in an earlier onset of the second consonant and/or a weakening of the first consonant, i.e., smaller

positive index values. In the extreme, curves become quite flat and only negative index values remain, indicating that the first consonant is dominated completely, possibly even replaced by the second. Due to the repeated measurements for speakers and for target words, the data have a crossed design structure. All recordings were taken with the same sampling rate, but the speaking durations differ. As changes relative to the length of the time interval (i.e., the duration of the consonant combination) are of interest, the index curves were standardized to a $[0,1]$ time interval. This results in different numbers and locations of the observation points between the observed curves.

We propose a model and an estimation approach that extend existing methods by accounting for both *a)* correlation between functional data and for *b)* irregular spacing of—possibly very few—observation points per curve. The model is a functional analogue of the linear mixed model (LMM), which is frequently used to analyze scalar correlated data.

We use functional principal component analysis (FPCA; see, e.g., Ramsay and Silverman, 2005) to extract the dominant modes of different sources of variation in the data. The functional random effects are expanded in bases of eigenfunctions of their respective auto-covariances, which we estimate beforehand using a novel smooth method of moments approach represented as an additive, bivariate varying coefficient model. FPCA is a key tool in FDA as it yields a parsimonious representation of the data. It is attractive as the eigenfunction bases are estimated from the data and have optimal approximation properties for a fixed number of basis functions. It also allows for an explicit decomposition of the variability in the data.

We propose two ways of predicting the eigenfunction weights. We either compute them directly as empirical best linear unbiased predictors (EBLUPs) of the resulting LMM or we alternatively embed our previously estimated eigenfunctions and λ -values in the general framework of functional additive mixed models (FAMMs) introduced by Scheipl et al. (2015). The first approach is straightforward and computationally much more efficient; it does not require additional estimation steps as a plug-in estimate is used, and is thus almost a by-product of the eigenfunction estimation. The latter has the advantage that all model components are estimated/predicted in one framework, allowing for approximate statistical inference conditional on the FPCA.

There is previous work on dependent functional data as well as on functional data that is irregularly or sparsely observed, but with few exceptions noted below, existing work has not addressed both issues simultaneously.

First, methods for dependent functional data differ in their generality and in their restrictions on the sampling grid. Brumback and Rice (1998) consider a smoothing spline-based method for nested or crossed curves, which are modeled as fixed effect curves. They allow for missing observations in equal grids but do not consider any covariate effects. A Bayesian wavelet-based functional mixed model approach is introduced by Morris et al. (2003) and extended by Morris and Carroll (2006), Morris et al. (2006), and subsequent work by this group. While this approach is quite general in the possible functional random effects structure, and fixed and random effects are estimated within one framework allowing for full Bayesian inference, it assumes regular and equal grids with at most a small proportion of missings and a reasonable number of completely observed curves. Di et al. (2009), Greven et al. (2010), and Shou et al. (2015) consider functional linear mixed models with

a functional random intercept, with a functional random intercept and slope, and with nested and crossed functional random intercepts, respectively. While following a similar approach to estimation for these models, all three are restricted to data sampled on a fine grid and fixed effects are estimated under an independence assumption not allowing for the statistical inference we provide. Di et al. (2014) extend the random intercept model of Di et al. (2009) to sparse functional data; the correlation structure, however, remains less general than ours and the estimation approach cannot easily be generalized to more complex structures. Also motivated by an application from linguistics, Aston et al. (2010) perform an FPCA on all curves ignoring the correlation structure and then use the functional principal component (FPC) weights as the response variables in an LMM with random effects for speakers and words. Only linear effects of scalar covariates are considered, FPC bases are restricted to be the same for all latent processes, and it is assumed that the data are sampled on a common grid. Brockhaus et al. (2015) propose a unified class for functional regression models including group-specific functional effects, which are represented as linear array models and estimated using boosting. The array structure requires equal grids and boosting does not provide inference. Other approaches concentrate specifically on spatially correlated functional data on equal grids, as, e.g., Staicu et al. (2010). Baayen et al. (2017) propose spline-based estimation of models that accommodate factor smooths for random effect factors which do not assume equal grids. Rather than smooth errors, their models, however, contain auto-correlated errors with a pre-specified auto-correlation parameter assuming variance homogeneity along the function argument, which usually does not fit the requirements of functional data. Scheipl et al. (2015) develop a flexible class of functional response models, allowing for various functional random effects with flexible correlation structures. Both spline-based and FPC-based representations are considered, and densely as well as sparsely sampled data are allowed. In the case of the FPC-based representation, they assume that appropriate FPC estimates are available. Yet, the estimation of the auto-covariances is challenging for correlated functional data with complex correlation structures, especially when observed on unequal grids, and no estimation approach is currently available. We combine our newly proposed FPC estimation with this general framework to obtain estimates and approximate point-wise confidence bands for the mean and covariate effects. In addition to providing an interpretable variance decomposition, our FPC-based approach reduces computation time by orders of magnitude compared to the spline-based estimates from Scheipl et al. (2015) (compare Section 4.5), allowing the analysis of realistically sized data in practice. Estimation errors and confidence band coverage also compare favorably.

Second, a number of approaches allow for irregularly or sparsely sampled functional data but assume that curves are independent. Guo (2002, 2004) first introduce the term *functional mixed effects models* for their model. The model does not capture between-function correlation as only curve-level random effect functions are included, which are modeled using smoothing splines. The approach is not restricted to regularly sampled grid data. Chen and Wang (2011) propose a spline-based approach that is suitable for sparsely sampled data, but similar to Guo (2002, 2004) they only consider curve-level random effects. James et al. (2000), Yao et al. (2005), and Peng and Paul (2009) among others propose FPCA approaches for sparsely observed functional data with uncorrelated

curves. For functional data with independent curves, there is a direct relationship to the longitudinal data literature as well, too extensive to cover here.

For an extensive overview and further references for functional regression approaches, including functional response regression, see Morris (2015) and Greven and Scheipl (2017).

We provide fully documented open-source software implementing our approach in the R add-on package `sparseFLMM`, where we also make the speech production data set available (R Core Team, 2016; Cederbaum, 2016). A description and examples for the usage of the R package can be found in Appendix E.

The remainder of this chapter is organized as follows. Section 4.2 introduces the general functional linear mixed model and presents an important special case which is used to analyze the motivating linguistic data on assimilation. Section 4.3 develops our estimation framework. Our method is evaluated in an application to the assimilation data and in simulations in Sections 4.4 and 4.5, respectively. Section 4.6 closes with a discussion and outlook. Theoretical results, and supplementary material including estimation details as well as additional results for application and simulations are available in Appendix C.

4.2 Functional linear mixed models

4.2.1 The general model

The general functional linear mixed model (FLMM) is given by

$$Y_i(t) = \mu(t, \mathbf{x}_i) + \mathbf{z}_i^\top \mathbf{U}(t) + E_i(t) + \varepsilon_i(t), \quad i = 1, \dots, n, \quad (4.1)$$

where $Y_i(t)$ is the square integrable functional response at observation point t in \mathcal{T} , a bounded interval in \mathbb{R} , and n is the number of curves. $\mu(t, \mathbf{x}_i)$ is a fixed main effect surface dependent on a vector of known covariates \mathbf{x}_i of length p . To account for the functional nature of the $Y_i(t)$, the random effects of an LMM are replaced by a vector-valued random process $\mathbf{U}(t)$. \mathbf{z}_i is a known covariate vector of length q . $E_i(t)$ is a curve-specific deviation in form of a smooth residual curve. We assume that there is white noise measurement error denoted by $\varepsilon_i(t)$ with variance σ^2 that captures random uncorrelated variation within each curve. Note that if needed, the error variance may also vary across t , $\sigma^2(t)$. We further assume that $\mathbf{U}(t)$, $E_i(t)$, and $\varepsilon_i(t)$, $i = 1, \dots, n$, are zero mean, mutually uncorrelated random processes and that $\mathbf{U}(t)$ and $E_i(t)$ are square integrable, which assures model identification. Therefore, each of the q components of $\mathbf{U}(t)$ has an auto-covariance function $K^{U_j}(t, t')$, $j = 1, \dots, q$, and cross-covariance functions $K^{U_{j,k}}(t, t')$, $j, k = 1, \dots, q$, some of which might be zero for uncorrelated functional random effects. $E_i(t)$ has an auto-covariance function $K^E(t, t') = \text{Cov}[E_i(t), E_i(t')]$. In the following, mean, auto-covariances, and thus also the eigenfunctions are assumed to be smooth in t . For any given t , Model (4.1) with our assumptions corresponds to an LMM with general mean $\mu(\mathbf{x}_i)$.

$\mu(t, \mathbf{x}_i)$ is an additive function of t and \mathbf{x}_i . For example, it can be constant in t , $\mu(t, \mathbf{x}_i) = \mu(\mathbf{x}_i)$, or additive in t and \mathbf{x}_i , $\mu(t, \mathbf{x}_i) = \mu_1(t) + \mu_2(\mathbf{x}_i)$. Another special case is when all x_{i1}, \dots, x_{ip} in \mathbf{x}_i

act as index-varying coefficients, $\mu(t, \mathbf{x}_i) = f_0(t) + f_1(t)x_{i1} + \dots + f_p(t)x_{ip}$, with unknown smooth functions $f_0(\cdot), \dots, f_p(\cdot)$.

4.2.2 Special case: the FLMM for a crossed design

For our application in speech production research (Section 4.4), we use an FLMM with a crossed design structure to account for correlation between measurements of the same speaker and between measurements of the same target word.

$$Y_{ijh}(t) = \mu(t, \mathbf{x}_{ijh}) + B_i(t) + C_j(t) + E_{ijh}(t) + \varepsilon_{ijh}(t), \quad (4.2)$$

with $i = 1, \dots, I$ (number of speakers), $j = 1, \dots, J$ (number of target words), and $h = 1, \dots, H_{ij}$ (number of repetitions). Here, $Y_{ijh}(t)$ is the h th index curve for speaker i and target word j at time t . $B_i(t)$ and $C_j(t)$ are functional random intercepts (fRIs) for the speakers and target words, respectively. Curve-specific deviations are accommodated by the smooth residual term $E_{ijh}(t)$, which also captures interactions between speakers and words. Based on substantive considerations and the limited sample size, we decided to not include an interaction effect separately. $\varepsilon_{ijh}(t)$ is additional white noise measurement error with variance σ^2 . We denote the auto-covariance functions by $K^B(t, t') = \text{Cov}[B_i(t), B_i(t')]$, $K^C(t, t') = \text{Cov}[C_j(t), C_j(t')]$, and $K^E(t, t') = \text{Cov}[E_{ijh}(t), E_{ijh}(t')]$, $i = 1, \dots, I$, $j = 1, \dots, J$, $h = 1, \dots, H_{ij}$.

4.2.3 Irregularly and sparsely sampled functional data

Let us now assume that for our general model (4.1) we have observed n curves on observation points $\{t_{i1}, \dots, t_{iD_i}\} \in \mathcal{T}$, $i = 1, \dots, n$. The number and the location of the observation points are allowed to differ from curve to curve. In the extreme, only one point may be observed for a curve. Moreover, the observation points of a curve do not have to be equally spaced. We denote realizations of the functional response $Y_i(t)$ at point t_{ij} by $y_{it_{ij}}$, $j = 1, \dots, D_i$. Accordingly, we denote realizations of the response in Model (4.2) by y_{ijht} with $t \in \{t_{ijh1}, \dots, t_{ijhD_{ijh}}\}$.

4.3 Estimation

We base our estimation on FPCA, which provides the dimension reduction so important for functional data and allows an explicit decomposition of the variability. Compared to other basis approaches, e.g., using splines, FPCA has the advantage that the eigen bases are optimal in the sense of giving the best approximation for a given number of basis functions and thus typically small numbers of basis functions give good approximations. To pool information across observations, which is particularly important in the case of irregularly or sparsely sampled functional data, we use smoothing of the auto-covariances of $\mathbf{U}(t)$ and $E_i(t)$, cf. Yao et al. (2005) for non-correlated sparse functional data. Previous approaches for smoothing the auto-covariances are restricted to less complex correlation structures or data sampled on an equal, fine grid. We apply eigen decompositions of the auto-covariances based

on Mercer's theorem (Mercer, 1909). The eigenfunctions, also known as FPCs, describe the main modes of variation of processes $\mathbf{U}(t)$ and $E_i(t)$ and the eigenvalues quantify the amount of variability explained by the corresponding FPCs. The eigenfunction weights, or FPC weights, give insight into the individual structure of each grouping level and can be used in further analyses, e.g., classification. The four main steps of our estimation procedure are outlined in the following.

Step 1 We estimate the mean $\mu(t, \mathbf{x}_i)$ using penalized splines based on a working independence assumption.

Step 2 We use a smooth method of moments estimator based on the centered curves to estimate the auto-covariances of the functional random effects.

Step 3 We conduct an eigen decomposition of each estimated auto-covariance matrix evaluated on a pre-specified, fine grid. Using the Karhunen-Loève (KL) expansion (Loève, 1946; Karhunen, 1947), we represent the functional random effects in truncated bases of eigenfunctions.

Step 4 We propose two ways of predicting the random basis weights.

Step 1, Step 3, and the first option for Step 4 are analogous to the estimation proposed in Di et al. (2009), Greven et al. (2010), and Shou et al. (2015) for functional data sampled on an equal, fine grid and in Di et al. (2014) for a simpler model. Step 2 is new and leads to a new combination with the FAMM approach of Scheipl et al. (2015) in the second option for Step 4. For simplicity, we focus in the remainder of this section, where we describe the four steps in detail, on Model (4.2).

4.3.1 Step 1: Estimation of the mean function

We estimate the mean $\mu(t, \mathbf{x}_{ijh})$ based on the working independence assumption

$$Y_{ijh}(t) = \mu(t, \mathbf{x}_{ijh}) + \varepsilon_{ijh}(t), \quad (4.3)$$

with i. i. d. Gaussian random variables $\varepsilon_{ijh}(t)$. Model (4.3) is an additive model with additive mean $\mu(t, \mathbf{x}_{ijh}) = f_0(t) + \sum_{k=1}^P f_k(t)x_{ijhk}$. We represent the unknown, smooth functions $f_k(\cdot)$ using B-splines and control the trade-off between goodness of fit and smoothness by adding a difference penalty (so called P-splines; Eilers and Marx, 1996). Using the penalized splines approximation of Model (4.3) allows us to represent the model as a scalar LMM, which has the advantage that the smoothing parameter can be estimated as a variance component ratio using restricted maximum likelihood (REML; Patterson and Thompson, 1971; cf. Ruppert et al., 2003, Section 4.9). We center the data using the estimated mean $\hat{\mu}(t, \mathbf{x}_{ijh})$ and focus in the following on the centered functional responses $\tilde{Y}_{ijh}(t) := Y_{ijh}(t) - \mu(t, \mathbf{x}_{ijh})$ and denote their realizations by \tilde{y}_{ijht} . For more general mean models than varying coefficient models, see Wood et al. (2015).

4.3.2 Step 2: Estimation of the auto-covariances

We estimate the auto-covariances using a smooth method of moments estimator. Whereas for data sampled on an equal, fine grid, estimation can be done point-wise, this is not possible for irregularly or sparsely sampled data, which makes the estimation of the auto-covariances more challenging and requires a new approach. We exploit the fact that for centered data, the expectation of the products $\tilde{Y}_{ijh}(t)\tilde{Y}_{i'j'h'}(t')$ corresponds to the auto-covariance, which can be decomposed as follows

$$\begin{aligned} \mathbb{E} \left[\tilde{Y}_{ijh}(t)\tilde{Y}_{i'j'h'}(t') \right] &= \text{Cov} \left[\tilde{Y}_{ijh}(t), \tilde{Y}_{i'j'h'}(t') \right] \\ &= K^B(t, t')\delta_{ii'} + K^C(t, t')\delta_{jj'} + [K^E(t, t') + \sigma^2\delta_{tt'}] \delta_{ii'}\delta_{jj'}\delta_{hh'}, \end{aligned} \quad (4.4)$$

with $\delta_{xx'}$ equal to one if $x = x'$ and zero otherwise. We propose to see Model (4.4) as an additive, bivariate varying coefficient model, in which the auto-covariances are the unknown smooth bivariate functions to be estimated, while $\delta_{ii'}$, $\delta_{jj'}$, $\delta_{ii'}\delta_{jj'}\delta_{hh'}$, and $\delta_{ii'}\delta_{jj'}\delta_{hh'}\delta_{tt'}$ represent the covariates. Under a working assumption of independence and homoscedastic variance of the products, we can use each empirical product $\tilde{y}_{ijht}\tilde{y}_{i'j'h't'}$ for which at least $i = i'$ or $j = j'$ to obtain smooth estimates of $K^B(t, t')$, $K^C(t, t')$, and $K^E(t, t')$ and an estimate of the error variance σ^2 . The total number of products $\tilde{y}_{ijht}\tilde{y}_{i'j'h't'}$ used for the estimation of the auto-covariances is of order $\mathcal{O}[\mathfrak{D}^2(1/I + 1/J)]$, with \mathfrak{D} the total number of observation points.

We use bivariate tensor product P-splines (see, e.g., Wood, 2006, Section 4.1.8) for the estimation of the auto-covariances, where low rank marginal bases for each t, t' are combined in order to obtain smooth functions of the two covariates. Let \otimes denote the Kronecker product. Then, given the appropriate ordering of the parameter vector, the part of the design matrix corresponding to $K^X(t, t')$, $X \in \{B, C, E\}$, is given by the respective indicator matrix multiplied entry-wise by $(\mathbf{M}_t^X \otimes \mathbf{1}_{F^X}^\top) \cdot (\mathbf{1}_{F^X}^\top \otimes \mathbf{M}_{t'}^X)$, where \mathbf{M}_t^X and $\mathbf{M}_{t'}^X$ denote the corresponding marginal spline design matrices of rank F^X for covariate t and t' , and $\mathbf{1}_{F^X} = (1, \dots, 1)^\top$ of length F^X . A smoothness penalty is introduced in order to avoid over-fitting. To account for the natural symmetry of the auto-covariances, we choose an isotropic penalty with a penalty matrix of the form $\mathbf{S}^X = \mathbf{S}_t^X \otimes \mathbf{S}_{t'}^X$, where \mathbf{S}_t^X and $\mathbf{S}_{t'}^X$ represent the respective marginal penalty matrices for t and t' . For reasons of model complexity and computational feasibility, we use marginal B-spline bases combined with marginal difference penalties. In principle, other bases or smoothing techniques are possible, which also applies to the estimation of the mean in Step 1. We take advantage of the mixed model representation of Model (4.4) for the estimation of the tensor product basis coefficients and the smoothing parameter using REML. During the estimation, strength is borrowed across all curves. This can be extremely advantageous for sparse functional data when some curves only have very few measurements and smoothing of curves would be infeasible.

In practice, negative estimated values of σ^2 are set to zero for the final estimate. Symmetry of the auto-covariances is ensured through the model apart from numerical inaccuracies.

For the practical implementation of Step 1 and Step 2 in the R add-on package `sparseFLMM`, we build on existing software and use R function `bam`, implemented in the R add-on package `mgcv`

which is especially designed for large data sets (Wood, 2011). Avoiding the construction of the complete design matrix leads to a low memory footprint and the possibility of parallelization gives a considerable speed-up in computation time. For further details, see Wood et al. (2015).

4.3.3 Step 3: Eigen decompositions of estimated auto-covariances

Based on Mercer's theorem, the eigen decompositions of the auto-covariances are

$$K^X(t, t') = \sum_{k=1}^{\infty} \nu_k^X \phi_k^X(t) \phi_k^X(t'), \quad X \in \{B, C, E\},$$

where $\nu_1^X \geq \nu_2^X \geq \dots \geq 0$ are the respective eigenvalues, $k \in \mathbb{N}$. The corresponding eigenfunctions $\{\phi_k^X, k \in \mathbb{N}\}$, $X \in \{B, C, E\}$, form an orthonormal basis in the Hilbert space $L^2(\mathcal{T})$ with respect to the L^2 -inner product $\langle f, g \rangle = \int f(t)g(t) dt$. In practice, the smooth auto-covariances are evaluated on an equally spaced, dense grid $\{t_1, \dots, t_{\tilde{D}}\}$ of pre-specified length \tilde{D} . The resulting matrices are in the following denoted as $\widehat{\mathbf{K}}^X = [\widehat{K}^X(t_d, t_{d'})]_{d, d'=1, \dots, \tilde{D}}$, $X \in \{B, C, E\}$. We conduct an eigen decomposition of each estimated auto-covariance matrix yielding estimated eigenvectors and eigenvalues. Rescaling is necessary to ensure that the approximated eigenfunctions are orthonormal with respect to the L^2 -inner product. Negative estimated eigenvalues are trimmed to zero to guarantee positive semi-definiteness.

Truncation of the FPCs

While in theory there is an infinite number of eigenfunctions, dimension reduction achieved by the selection of the number of FPCs for each random process is necessary in practice. This truncation has a theoretical justification and can be seen as a form of penalization (see, e.g., Di et al., 2009; Peng and Paul, 2009). Among the multiple proposals in the literature (for an overview, see Greven et al., 2010), we base our choice on the proportion of variance explained. This allows us to quantify the contribution of the random processes to the variation in the observed data. It is based on the variance decomposition of the response

$$\int_{\mathcal{T}} \text{Var}[Y_{ijh}(t)] dt = \sum_{k=1}^{\infty} \nu_k^B + \sum_{k=1}^{\infty} \nu_k^C + \sum_{k=1}^{\infty} \nu_k^E + \sigma^2 |\mathcal{T}|.$$

The sums $\sum_{k=1}^{\infty} \nu_k^X$, $X \in \{B, C, E\}$, quantify the relative importance of each of the three random processes. We choose principal components of decreasing importance until a pre-specified level of explained variation is reached.

Approximation of the functional random processes

Based on the truncation, we use KL expansions to obtain parsimonious basis representations for the random processes

$$B_i(t) \approx \sum_{k=1}^{N^B} \xi_{ik}^B \phi_k^B(t), \quad C_j(t) \approx \sum_{k=1}^{N^C} \xi_{jk}^C \phi_k^C(t), \quad E_{ijh}(t) \approx \sum_{k=1}^{N^E} \xi_{ijhk}^E \phi_k^E(t).$$

Note that in the case of irregularly or sparsely sampled data, the observation points t also depend on i , j , and h , which we omit throughout this chapter for better readability. For the same reason, we do not emphasize that the truncation lags and eigenfunctions are estimated. By construction, the basis weights ξ_{ik}^B , ξ_{jk}^C , and ξ_{ijhk}^E are uncorrelated random variables with zero mean and variance ν_k^X , $k \in \mathbb{N}$, $X \in \{B, C, E\}$.

For prediction of the FPC weights, we first linearly interpolate the chosen eigenfunctions such that they are available on the original observation points. Due to the smoothness of all model components, this leads to a small error which could be further decreased, if desirable, by further increasing the number of grid points \tilde{D} .

See Section C.2 in Appendix C for further details, including the rescaling of the FPCs, and Section C.1 in Appendix C for the derivation of the variance decomposition.

4.3.4 Step 4: Prediction of the basis weights

The basis weights for a centered random process $X_i(t)$ are often represented as the inner product of $X_i(t)$ and the respective FPC. Estimation is more complicated for dependent functional data contaminated with additional measurement error as the weights belonging to the different basis expansions cannot be separated, and ignoring the measurement error leads to biased predictions. Moreover, numerical integration would not work (well) for irregularly or sparsely sampled data.

These considerations motivate our two proposals for the prediction of the basis weights which are both implemented in the R add-on package `sparseFLMM`. The first is straightforward and computationally very efficient. It is almost a by-product of the FPC estimation, taking only a few seconds for our large speech production data. It generalizes the conditional expectations introduced by Yao et al. (2005). The second involves higher computational costs but has the advantage that the mean is re-estimated in the same framework, allowing for approximate statistical inference, e.g., for the construction of point-wise confidence bands (CBs) conditional on the FPCA. Depending on the sample size of the data and the main question of interest, one or the other may be preferred. Further details for both, such as concrete matrix forms, can be found in Section C.1 in Appendix C.

Prediction of the basis weights as EBLUPs

Using the truncated KL expansions of the random processes, we can approximate Model (4.2) by

$$Y_{ijh}(t) \approx \mu(t, \mathbf{x}_{ijh}) + \sum_{k=1}^{N^B} \xi_{ik}^B \phi_k^B(t) + \sum_{k=1}^{N^C} \xi_{jk}^C \phi_k^C(t) + \sum_{k=1}^{N^E} \xi_{ijhk}^E \phi_k^E(t) + \varepsilon_{ijh}(t) \quad (4.5)$$

for the discrete observation points $t \in \{t_{ijh1}, \dots, t_{ijhD_{ijh}}\}$. The resulting model (4.5) is a scalar LMM in which the random effects correspond to the basis weights (Di et al., 2009). The basis weights are directly predicted as EBLUPs without fitting Model (4.5), plugging in the previously estimated components, as derived in the following. Note that under Gaussian assumption, the predictors are empirical best unbiased predictors (EBUPs). When the assumption is relaxed they remain best linear unbiased predictors (EBLUPs) (Harville, 1976).

Let $\tilde{\mathbf{Y}}$ denote the stacked centered response vector of length \mathfrak{D} . Let $L^X \in \{I, J, n\}$ and $N^X \in \{N^B, N^C, N^E\}$ denote the levels of the grouping factor and the truncation lag for process X , $X \in \{B, C, E\}$, respectively. We define $\boldsymbol{\xi} = (\boldsymbol{\xi}^{B^\top}, \boldsymbol{\xi}^{C^\top}, \boldsymbol{\xi}^{E^\top})^\top$, with $\boldsymbol{\xi}^X = (\boldsymbol{\xi}_1^{X^\top}, \dots, \boldsymbol{\xi}_{L^X}^{X^\top})^\top$ the stacked vector of the basis weights of length $L^X N^X$. Thus, $\boldsymbol{\xi}$ is a vector of length $\mathfrak{N} := IN^B + JN^C + nN^E$. $\hat{\boldsymbol{\Phi}}$ is the joint $\mathfrak{D} \times \mathfrak{N}$ design matrix of the form $\hat{\boldsymbol{\Phi}} = [\hat{\boldsymbol{\Phi}}^B | \hat{\boldsymbol{\Phi}}^C | \hat{\boldsymbol{\Phi}}^E]$, where $\hat{\boldsymbol{\Phi}}^B$, $\hat{\boldsymbol{\Phi}}^C$, and $\hat{\boldsymbol{\Phi}}^E$ are the respective design matrices containing the rescaled FPC estimates evaluated on the original observation points. $\hat{\mathbf{G}}$ denotes the estimated covariance matrix of $\boldsymbol{\xi}$. It is a diagonal matrix with elements corresponding to the estimated eigenvalues of the random processes.

The EBLUP for the basis weights in Model (4.2) in the usual form (see Section C.1 in Appendix C) requires the inversion of the estimated covariance matrix of $\tilde{\mathbf{Y}}$, which is of dimension $\mathfrak{D} \times \mathfrak{D}$. This can be computationally demanding for large numbers of observation points. Furthermore, when $\hat{\sigma}^2 \approx 0$, the covariance becomes singular. Transformations with the Woodbury formula yield the more favorable form

$$\hat{\boldsymbol{\xi}} = \left(\hat{\sigma}^2 \hat{\mathbf{G}}^{-1} + \hat{\boldsymbol{\Phi}}^\top \hat{\boldsymbol{\Phi}} \right)^{-1} \hat{\boldsymbol{\Phi}}^\top \tilde{\mathbf{Y}}, \quad (4.6)$$

for which the inversion is simplified to that of an $\mathfrak{N} \times \mathfrak{N}$ matrix which has full rank when either $\hat{\sigma}^2$ is positive or when $\hat{\boldsymbol{\Phi}}^\top \hat{\boldsymbol{\Phi}}$ has full rank. In practice, when neither of these requirements is met, the Moore-Penrose generalized inverse is used. Note that when $\hat{\sigma}^2 = 0$, the EBLUP simplifies to the least squares estimator.

This computationally efficient way of predicting $\boldsymbol{\xi}$ can be used when the focus is not on inference for covariate effects or when the data are large and the computational resources are limited. One drawback is, however, that the mean is estimated using a working independence assumption. This may not be statistically efficient and does not directly provide valid statistical inference. This motivates our second proposal.

Prediction of the basis weights using FAMMs

The second option uses the fact that under Gaussian assumption, Model (4.5) together with the distribution of the basis weights implied by the KL expansion falls into the general framework of a FAMM (Scheipl et al., 2015) using suitable marginal bases and penalties. We combine our FPC estimation with the FAMM idea and write Model (4.5) using estimated eigenfunctions and -values as

$$\mathbf{Y} = \sum_{k=0}^p \left(\Psi_c^k \otimes \mathbf{1}_{F^k}^\top \right) \cdot \Psi_t^k \boldsymbol{\theta}^k + \sum_{X \in \{B, C, E\}} \left(\Psi_g^X \otimes \mathbf{1}_{N^X}^\top \right) \cdot \left(\mathbf{1}_{L^X}^\top \otimes \Psi_t^X \right) \boldsymbol{\xi}^X + \boldsymbol{\varepsilon}, \quad (4.7)$$

with $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_{\mathcal{D}})$. \mathbf{Y} is the stacked uncentered response vector of length \mathcal{D} , and the mean is re-estimated with Ψ_c^k denoting an inflated vector of length \mathcal{D} of covariate values. Ψ_t^k of dimension $\mathcal{D} \times F^k$ comprises the evaluations of F^k spline basis functions on the \mathcal{D} time points t_{ijh} . $\boldsymbol{\theta}^k$ is a coefficient vector of length F^k . For the functional random effects, Ψ_g^X denotes an inflated $\mathcal{D} \times L^X$ matrix of grouping indicators. The $\mathcal{D} \times N^X$ matrix Ψ_t^X comprises the evaluations of the N^X respective estimated eigenfunctions on the original observation points. Adding penalties of the form $\boldsymbol{\xi}^{X^\top} (\mathbf{I}_{L^X} \otimes \mathbf{P}_t^X) \boldsymbol{\xi}^X$, with $\mathbf{P}_t^X = \text{diag}(\hat{\nu}_1^X, \dots, \hat{\nu}_{N^X}^X)^{-1}$, corresponds to the distributional assumption $\boldsymbol{\xi}_l^X \sim \mathcal{N}[0, \text{diag}(\hat{\nu}_1^X, \dots, \hat{\nu}_{N^X}^X)]$, $l = 1, \dots, L^X$, $X \in \{B, C, E\}$, implied by the KL expansions under Gaussianity. This set-up using linear combinations of the above tensor product bases with an appropriate penalty falls naturally into the framework of a FAMM and was in fact discussed in Scheipl et al. (2015) without, however, providing an approach to estimation of the eigenfunctions and -values needed in Ψ_t^X and \mathbf{P}_t^X . Model (4.7) is a scalar additive mixed model, which allows to take advantage of established methods for estimation and for statistical inference (for more details, see Scheipl et al., 2015). Re-estimation of the mean in one framework with the basis weights, particularly allows us to construct point-wise CBs for the mean and for covariate effects. Note that the inference is conditional on the estimated FPCA, i.e., it accounts neither for the uncertainty in the estimated eigenfunctions and -values nor for the truncation, which may lead to an underestimation of the variability. (Compare, however, the good coverage in our simulations in 4.5.2). In practice, we use function `pffr` that Scheipl et al. (2015) provide in the R add-on package `refundDevel` (Huang et al., 2016b). A constraint on the functional random effects assures that they are centered. In addition to the parsimonious basis of eigenfunctions, this approach has the advantage of not necessitating the estimation of any smoothing parameters for the random processes, as the variances of the random weights have already been estimated and the smoothing parameter can be set to one. These two features lead to a drastic decrease in computational cost compared to spline-based prediction of the random processes, as is shown in our simulations in Section 4.5.

The estimation quality can be further improved, if desirable, by applying the four estimation steps iteratively. Several possibilities are described in Section C.2 in Appendix C, where also further details on the estimation and implementation can be found.

4.4 Application to the speech production research data

4.4.1 Background and scientific questions

In linguistics, the term assimilation refers to the common phenomenon whereby a consonant becomes phonetically more like an adjacent, usually following consonant. Assimilation commonly occurs in

English phrases such as ‘Paris show’ in which the word-final /s/-sound is, in fluent speech, pronounced very similar to the following, word-initial /sh/-sound (Pouplier et al., 2011). Assimilation patterns are conditioned by a complex interaction of perceptual, articulatory and language-specific factors and are therefore a central research topic in the speech sciences. In order to investigate assimilation in German, Pouplier et al. (2014) obtained audio recordings of $I = 9$ speakers reading the same $J = 16$ target words, each five times. Due to recording errors, for some combinations only four repetitions are included in the data, i.e., $H_{ij} \in \{4, 5\}$. The authors concentrated on variation in assimilation patterns for the consonants /s/, /sh/ as a function of their order (/s#sh/versus /sh#s/, where # denotes a word boundary), syllable stress and vowel context. Target words consisted of bisyllabic noun-noun compounds. In half of the target words consonant /s/ is followed by word-initial /sh/, such as in the word ‘Callas-**S**chimmel’. The other half contains the sequence /sh#s/, e.g., ‘Gulasch-**S**ymbol’. In the following, we will refer to the syllables containing the consonants of interest as final and initial target syllables (and correspondingly to final and initial target consonants). The time interval in which the consonants of interest appear in the utterance was cut out manually from the audio recording for each repetition and the resulting time-varying acoustic signal was summarized in a functional index over time, varying between +1 and -1. Reference patterns for both consonants were used to construct the index such that it ranges for both orders from +1 for sounds close to the reference for the first consonant of the sequence to -1 for sounds close to the reference for the second consonant of the sequence (for more details, see Pouplier et al., 2011 and Section C.3 in Appendix C for data pre-processing). The resulting index curves are displayed in Figure 4.1.

A special focus lies on the asymmetry arising from the order of the consonants. We investigate under which conditions (order, syllable stress, vowel context) the two consonants assimilate and whether assimilation is symmetric with respect to the orders /s#sh/and /sh#s/. A common approach is to extract curve values at pre-defined points on the time axis (e.g., 25%, 50%, 75%) which are subsequently used in multivariate methods (e.g., Pouplier et al., 2011). Such analyses fail to capture the continuous dynamic change characteristic of speech signals. Applying our FDA-based method allows us to take into consideration the temporal dynamics and to account for the complex correlation structure in the data which arises from the repeated measurements of speakers and of target words. Moreover, we can quantify the effect of covariates and interactions and obtain a variance decomposition.

All utterances were recorded with the same sampling rate (32768 Hz) and then standardized to a [0,1] interval as the speaking rate, and hence the target consonant duration, differs across experiments. After standardization, measurements are unequally spaced for different curves. In some data settings, registration can be used to account for variation in time. For this application, however, registration cannot replace the standardization of the time interval as different transition speeds between the two consonants are part of the research question of interest and thus a change relative to the length of the time interval is of interest. Registration would remove a main source of information on the assimilation process and flat curves, arising from (near) complete assimilation, would render registration problematic.

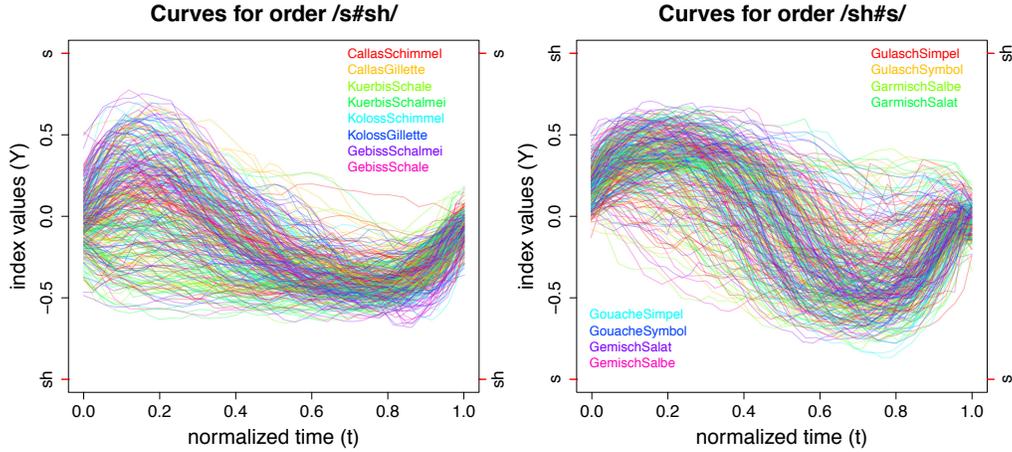


Figure 4.1: Index curves of the consonant assimilation data over time. Left [right]: Curves of order $/s\#sh/$ [$/sh\#s/$]. Positive values approaching $+1$ indicate a reference $/s/$ [$/sh/$] acoustic pattern, while negative values approaching -1 indicate a reference $/sh/$ [$/s/$] acoustic pattern.

4.4.2 A model for the speech production research data

In order to account for the repeated measurements of speakers and target words, we fit an FLMM with crossed fRIs, Model (4.2), to the consonant assimilation data. The number of measurements per curve D_{ijh} ranges from 22 to 57 with a median of 34. During estimation, we truncate the numbers of FPCs using a pre-specified proportion of explained variance of $L = 0.95$. The equidistant grid on which the auto-covariances are evaluated is of length $\tilde{D} = 100$. We use cubic B-splines with third order difference penalties for the estimation of the mean effects and as marginal basis functions for the estimation of the auto-covariances. We predict the FPC weights using both options. As confidence bands for the covariate and interaction effects are of interest here, the focus lies on the second approach using the FAMM framework.

Covariate effects

We consider four dummy-coded covariates: consonant order (order), stress of the final (stress1) and of the initial (stress2) target syllable, which can be strong or weak, and vowel context (vowel), which refers to the vowels immediately adjacent to the target consonants and is either of the form ia or ai, e.g., Callas-Schimmel. Moreover, we include the interactions of the consonant order with each of the other three covariates. All covariates enter the mean as varying coefficients,

$$\begin{aligned}
 \mu(t, \mathbf{x}_{ijh}) &= f_0(t) + f_1(t) \cdot \text{order}_j + f_2(t) \cdot \text{stress1}_j + f_3(t) \cdot \text{stress2}_j \\
 &+ f_4(t) \cdot \text{vowel}_j + f_5(t) \cdot \text{order}_j \cdot \text{stress1}_j + f_6(t) \cdot \text{order}_j \cdot \text{stress2}_j \\
 &+ f_7(t) \cdot \text{order}_j \cdot \text{vowel}_j.
 \end{aligned} \tag{4.8}$$

Thus, in total, eight covariates characterize the 16 target words.

4.4.3 Application results

Our estimation yields two and three FPCs for the fRI for speakers and for the smooth error, respectively. No FPC is chosen for the fRI for target words. It is likely that the eight covariate and interaction effects describe the target words sufficiently, as confirmed by obtaining one FPC for the fRI for target words in the model without covariate effects. Most variability (67.29%) is explained by the three chosen FPCs for the curve-specific deviation which also captures interactions between speakers and target words. The two chosen FPCs for speakers explain 20.45% of the estimated variability.

The left panel of Figure 4.2 shows the effect of covariate order (f_1), which has the largest effect on the index trajectories. Covariate order is dummy-coded with reference category /s#sh/. Thus, the mean curves of target words with order /sh#s/ are pulled towards the ideal reference /sh/ during the first consonant and differ slightly from the ideal /s/ during the second consonant compared to order /s#sh/. We conclude that there is an asymmetry of consonant assimilation with respect to the consonant order and that /s/ is more affected by the assimilation than /sh/. These results are consistent with results for English obtained by Pouplier et al. (2011).

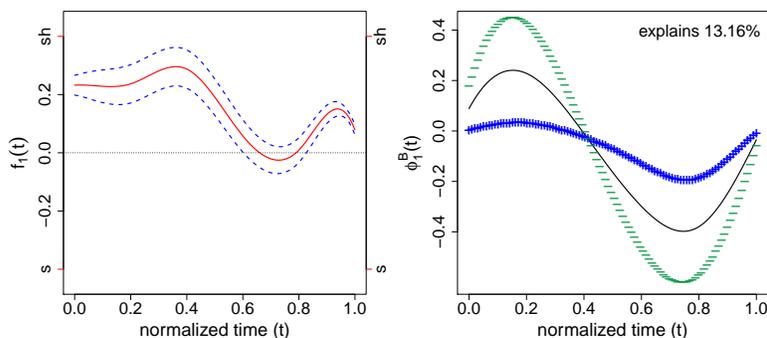


Figure 4.2: Left: Effect of covariate order (red solid line) with point-wise confidence bands (dashed lines). Right: Mean function (solid line) and the effect of adding (+) and subtracting (−) a suitable multiple ($2\sqrt{\widehat{v}_1^B}$) of the first FPC for speakers.

Moreover, we find that assimilation is stronger for target words with unstressed final syllables (f_2), especially for order /s#sh/(f_5). Changing the stress of the initial syllable only has an effect for order /sh#s/(f_6). This means that in both final and initial position, stress effects are evident during /s/ but not during /sh/. For both consonant orders, the vowel context mainly affects the transition between the two consonants (f_4 and f_7). The first consonant is closer to the ideal reference value in the ai compared to the ia condition, yet the second consonant is pulled away from its reference value. These results show that the /i/ vowel perturbs an adjacent consonant away from its ideal reference pattern.

In the right panel of Figure 4.2, we show the effect of adding (+) and subtracting (−) a suitable multiple of the first FPC for speakers to the overall mean (solid line) obtained by setting all covariates to 0.5. The interpretation is straightforward: speakers with a negative weight for the first FPC

distinguish better between the two consonants. The estimates for the basis weights can be used for further analysis. Further application results including plots for all mean effects can be found in Section C.3 in Appendix C.

The data analysis is fully reproducible as the speech production data set is included in the R add-on package `sparseFLMM`.

4.5 Simulations

4.5.1 Simulation designs

We conduct extensive simulation studies to investigate the performance of our method. The data generating processes can be divided into two main groups: 1) data that mimics the irregularly sampled consonant assimilation data and 2) sparsely sampled data with a higher number of observations per grouping level but fewer observations per curve. For all settings, we generate 200 data sets.

Application-based simulation scenarios

We consider two application-based scenarios, one with an fRI for speakers and covariate mean effects (fRI scenario) and another with crossed fRIs for speakers and for target words, respectively, but no covariate mean effects (crossed-fRIs scenario). We generate the data based on the estimates of Model (4.2) for our consonant assimilation data with $\mu(t, \mathbf{x}_{ijh})$ corresponding to (4.8) and to a simple smooth intercept $\mu(t)$, respectively. The data analysis yields two FPCs for the fRI for the speakers and three FPCs for the smooth error term. For the crossed-fRIs scenario, we additionally obtain one FPC for the fRI for the target words. The FPC weights and the measurement errors are independently drawn from normal distributions with zero mean and with the respective estimated variances. To assess the effect of model misspecification, we conduct additional simulations of the crossed-fRI scenario, using FPC weights drawn from a mixture of two normals, with equal probability from either $\mathcal{N}(\sqrt{\nu_k/2}, \nu_k/2)$ or $\mathcal{N}(-\sqrt{\nu_k/2}, \nu_k/2)$ as in Yao et al. (2005). We obtain very similar results to corresponding results for normal weights and the curves can be reconstructed equally well. More details on the data generation can be found in Sections 4.4.3, 4.5.2, and in Sections C.3 and C.4 in Appendix C.

Sparse simulation scenario

In order to investigate the estimation performance in the sparse case, we additionally generate data with crossed fRIs as in Model (4.2) consisting of observations that are sparsely sampled on $[0,1]$. The number of observation points per curve is drawn from the discrete uniform distribution $\mathcal{U}[3, 10]$. For $B_i(t)$ and $C_j(t)$, we choose $I = J = 40$ replications each with each combination observed $H_{ij} = 3$ times. We use two FPCs each to generate the underlying process. Eigenvalues are generated as $\nu_k^X = 2/k, k = 1, 2, X \in \{B, C, E\}$. We choose normalized Legendre Polynomials adapted to the interval $[0, 1]$ as FPCs for $B_i(t)$ and $C_j(t)$. For the smooth error $E_{ijh}(t)$, we choose a basis of sine and cosine functions. See Section C.4 in Appendix C for details. The FPC weights and the measurement

errors are independently drawn from the normal distributions $\mathcal{N}(0, \nu_k)$ and $\mathcal{N}(0, \sigma^2)$, respectively. No covariates are included in the mean function $\mu(t) = \sin(t) + t$. We set the error variance to $\sigma^2 = 0.05$.

For all scenarios, we center the FPC weights such that the weights of each grouping factor also empirically have zero mean. Moreover, we decorrelate the basis weights belonging to one grouping factor and assure that the empirical variance corresponds to the respective eigenvalue. This is done to obtain data that meets the requirements of our model. It allows us to separate the effect of unfavorably drawn weights and of the estimation performance. This adjustment gains importance for small sample sizes I , J , and n and also when the true eigenvalues are high. Note that in practice, we do not have centered and decorrelated FPC weights and thus estimates for small sample sizes will reflect the distribution in the sample rather than that in the population. To assess the impact of this procedure, we also compare our results to those of simulations using the original (non-centered and non-decorrelated) FPC weights, which can be found in Section C.4 in Appendix C.

We fix the number of FPCs in order to separate the effect of the truncation from the estimation quality. We use five marginal basis functions each for the estimation of the auto-covariances and eight basis functions for the estimation of the mean. We predict the FPC weights as EBLUPs for all scenarios and additionally compare with the computationally more expensive FAMM prediction (FPC-FAMM) for the fRI scenario with covariates.

We compare our FPC-based approach to a spline basis representation of the functional random effects (using eight basis functions) within the FAMM framework of Scheipl et al. (2015) (spline-FAMM). To the best of our knowledge, the work of Scheipl et al. (2015) is the only competitor to our approach as all other methods that meet the requirements of functional data are either restricted to equal, fine grids or do not allow for a crossed structure. Due to the high computational costs of Scheipl et al. (2015), we restrict our comparison to the fRI scenario, in which we can compare estimation quality and CBs coverage for covariate effects.

4.5.2 Simulation results

We focus our discussion on the FPC-based results for the application-based scenario with crossed fRIs and compare with the other settings and estimation approaches.

We use root relative mean squared errors (rrMSE) as measures of goodness of fit which are of the general form $\sqrt{(\text{true-estimated})^2/\text{true}^2}$. For the simulations of the fRI scenario with covariate effects, we additionally evaluate the average point-wise and the simultaneous coverage of the point-wise CBs. The complete results for all simulations as well as rrMSE definitions for scalars, vectors, and functions are given in Section C.4 in Appendix C.

Simulation results for the crossed-fRIs scenario

Figure 4.3 shows the true and estimated FPCs of the two fRIs as well as of the smooth error term. As expected, the FPCs are estimated better the more independent levels there are for the corresponding grouping factor which can enter the estimation of the auto-covariance. The FPCs of the smooth

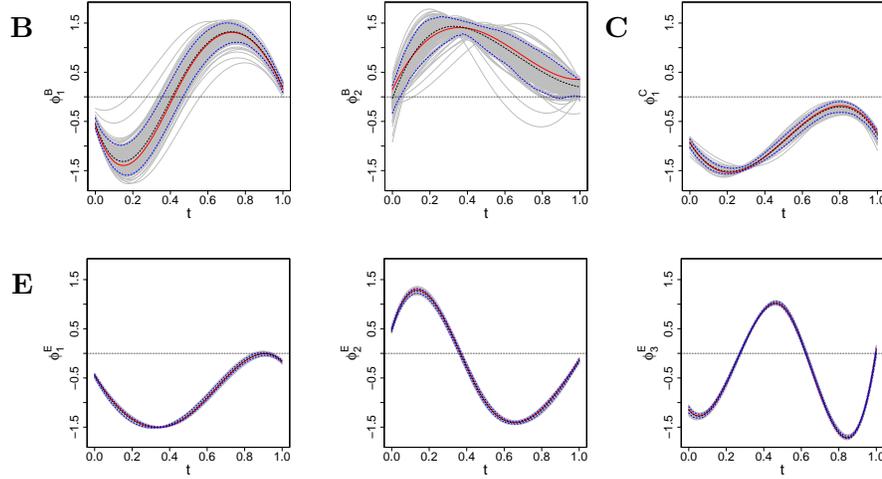


Figure 4.3: True and estimated FPCs of the crossed fRIs $B_i(t)$ and $C_j(t)$ (top row), as well as of the smooth error $E_{ijh}(t)$ (bottom row). Shown are the true functions (red solid line), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

error term (707 levels) are estimated best, followed by the FPC of the fRI for target words (16 levels). Most variability in the estimates is found for the FPCs of the fRI for speakers due to the small number of speakers ($I = 9$), but the main features of the curves are still recovered relatively well. We obtain similar results for the fRI scenario. The number and complexity of the FPCs also plays an important role for the estimation quality, as can be seen from the results for the sparse scenario, where the first FPC of $B_i(t)$ (40 levels) is estimated better than the first FPC of $E_{ijh}(t)$ (4800 levels). The latter has a more complex form, difficult to capture with five basis functions.

Table 4.1 lists the rrMSEs averaged over 200 simulation runs for all model components. It shows that the mean function is reconstructed very well, which is also the case in the sparse scenario. The covariate effects for the fRI scenario are discussed below.

Table 4.1: rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSE for $B_i(t)$, $C_j(t)$, and $E_{ijh}(t)$ and their covariance decompositions. Last row: Average rrMSEs for $Y_{ijh}(t)$, $\mu(t, \mathbf{x}_{ijh})$, and σ^2 .

X	L^X	K^X	ϕ_1^X	ϕ_2^X	ϕ_3^X	ν_1^X	ν_2^X	ν_3^X	ξ_1^X	ξ_2^X	ξ_3^X	X	μ	σ^2
B	9	0.26	0.15	0.18		0.15	0.34		0.18	0.35		0.22		
C	16	0.32	0.05			0.31			0.12			0.13		
E	707	0.06	0.02	0.03	0.02	0.04	0.08	0.03	0.17	0.19	0.26	0.19		
Y												0.10	0.02	0.09

The auto-covariances and their eigenvalues have similar low average rrMSEs for both application-based scenarios. For the sparse scenario, the eigenvalues are estimated even better with average

rrMSEs between 0.02 and 0.05. For the auto-covariances for the sparse scenario, we obtain average rrMSEs of 0.06 for each of the crossed fRIs and an average of 0.14 for the smooth error which is due to the complex eigenfunctions mentioned above. The error variance has similar low average rrMSEs for the two application-based scenarios. For the sparse scenario, the average rrMSE is higher, which is due to the estimation inaccuracies in the auto-covariance of the smooth error.

The prediction quality of the basis weights clearly depends on the estimation quality of the FPCs and of the eigenvalues, as well as of the error variance, as evident from Equation (4.6). Also important for the prediction of the basis weights is the number of curves with the given weight entering the prediction. Thus, the basis weights of $C_j(t)$ are better predicted than those of $E_{ijh}(t)$. As expected, basis weights of FPCs that explain more variability are predicted better. Similar results can be found for the fRI and for the sparse scenario.

For all scenarios, we obtain good results for the functional random effects as well as for the functional response. The rrMSEs for the functional response are lowest, which is due to the fact that even if the FPC bases are not perfectly estimated, they can still serve as a good empirical basis. Thus, the data can be reconstructed very well.

We found considerably more outliers of the relative errors for the sparse scenario than for the other two scenarios, which is most probably due to an unfavorable distribution of the few observation points across the curves in a few data sets.

Overall, we can conclude that all components are estimated well and especially for the functional response we obtain very small rrMSEs across all simulations.

Comparison of the different estimation results for the fRI scenario

We find that the functional random processes and the functional response are estimated equally well for the two options of the basis weights prediction. The functional response is again estimated very well with an average rrMSE of 0.09 for both EBLUP and FPC-FAMM estimation. The spline-FAMM results are considerably worse for the random processes (almost three (smooth error) and almost seven (fRI) times higher average rrMSEs), which results from the fact that the constraint $\sum_{l=1}^{L^X} X_l(t) \equiv 0$, $X \in \{B, E\}$, is not fulfilled and parts are shifted between terms. The functional response is recovered reasonably well, but has a more than 1.5 times higher average rrMSE than the EBLUP and FPC-FAMM estimates. Note that due to high computation times (see below), we only consider 100 simulation runs for the spline-FAMM simulation.

For the covariate effects, the FPC-FAMM estimation gives better results than the estimation under an independence assumption (between 1 and 1.28 times lower average rrMSEs) and considerably better results than the spline-FAMM estimation (between 2.8 and six times lower average rrMSEs). In spite of ignoring the variability of the estimated FPCA, the average point-wise coverage of the point-wise CBs is very good for most effects for FPC-FAMM (between 91.18% and 95.54%) and the simultaneous coverage is reasonable. Both are considerably better than for the spline-FAMM alternative (point-wise coverage between 35.12% and 41.67%). The coverage for the latter would most probably improve by increasing the number of spline basis functions which is, however, limited by the high computation time.

Computation times

Our simulations show that the FPC-based approach has clear advantages in terms of computational complexity, despite the computational cost of the auto-covariance estimation. We compare times for one simulation run of the fRI scenario for each estimation option obtained under the same conditions (without parallelization in function `bam` that would speed up the estimation). The study was run on a 64 Bit Linux platform with 660 Gb of RAM memory. The FPC-based approach with the basis weights predicted as EBLUPs took 1.6 hours and predicting the basis weights using FPC-FAMM took slightly more than six hours longer. The spline-FAMM took by far the longest with a duration of ten days which is due to the two extra smoothing parameters each for the fRI and the smooth error which have to be estimated. Moreover, using FPCs reduces the number of necessary basis functions. To assess the feasibility to apply our approach in practice on a desktop PC, we also ran our real data analysis on a 64 Bit Windows PC with 64 Gb of RAM. Without parallelization, the FPC-based estimation and EBLUP computation took two hours and the FPC-FAMM an additional 20 hours.

4.6 Discussion and outlook

We propose an FPC-based estimation approach for functional linear mixed models that is particularly suited to irregularly or sparsely sampled observations. To pool information, we smooth both the mean and auto-covariance functions. We propose and compare two options for the prediction of the FPC weights and obtain conditional point-wise confidence bands for the functional covariate effects. Our simulations show that our method reliably recovers the features of interest. The parsimonious representation of the functional random effects in bases of eigenfunctions outperforms the spline-based alternative of Scheipl et al. (2015) with which we compare, both in terms of error rates and coverage as well as in terms of computation time. To the best of our knowledge, there is no other competitor to our approach as all other methods that are suitable for functional data are either restricted to regular grid data or simpler correlation structures. In our application to speech production data, we show that our method allows conclusions to be drawn about the asymmetry of consonant assimilation to an extent which is not achievable using conventional methods with data reduction. To make the proposed estimation method readily accessible to users, we provide documented open-source software implementing our approach in the R add-on package `sparseFLMM` (Cederbaum, 2016).

Building on existing methods for our estimation approach allows us to take advantage of robust, flexible algorithms with a high functionality. The computational efficiency, however, could potentially be improved by exploiting the special structure of our model. An interesting direction is to improve the estimation of the auto-covariances in order to better account for their symmetry and positive semi-definiteness and for the fact that the products in Model (4.4) are not homoscedastic. Moreover, it would be interesting to compare the different options for iterative estimation in detail. How the first aspect—the symmetry of the auto-covariances—can be addressed is discussed in the next chapter.

The construction of point-wise and simultaneous confidence bands that account for the variability of the estimated FPC decomposition is beyond the scope of this work, but would be of interest. For uncorrelated functions, Goldsmith et al. (2013) propose bootstrap-based corrected confidence bands

for densely and sparsely sampled functional data. However, it remains an open question how to extend their non-parametric bootstrap to our correlated curves, and computational cost is another issue.

Acknowledgements

We thank Fabian Scheipl and Simon Wood for making available and further improving their extensive software at our requests, for technical support, and for fruitful discussions. Sonja Greven and Jona Cederbaum were funded by the Emmy Noether grant GR 3793/1-1 from the German Research Foundation. Marianne Pouplier was supported by the ERC under the EU's 7th Framework Programme (FP/2007-2013)/Grant Agreement n. 283349-SCSPL. We thank the referees, the associate editor, and the editor for their useful comments.

Chapter 5

Fast Symmetric Additive Covariance Smoothing

“ Do the best you can until you know better. Then when you know better, do better.

MAYA ANGELOU

Contributing manuscript

This chapter is based on the following paper:

Cederbaum, J., Scheipl, F., and Greven, S. (2018): Fast symmetric additive covariance smoothing. *Computational Statistics & Data Analysis*, 120:25–41.

This is joint work with Fabian Scheipl (Department of Statistics, LMU Munich, Germany) and Sonja Greven (Department of Statistics, LMU Munich, Germany). The idea to develop an efficient covariance smoothing approach in the framework of functional linear mixed models was first initiated by Sonja Greven and taken up after previous joint work (Cederbaum et al., 2016) confirmed the need for a fast covariance estimation approach. Fabian Scheipl came up with the idea to only estimate the upper triangle of the surface including the diagonal and Sonja Greven pointed out that smoothness across the diagonal needs to be ensured. Jona Cederbaum pursued different approaches before Fabian Scheipl finally proposed a way to incorporate the symmetry constraint into the modeling framework in a special case. Jona Cederbaum developed the theoretic framework and extended the idea for the special case to the general model under the supervision of Sonja Greven. In particular, Jona Cederbaum derived the concrete forms of the constraint matrices and the covariance of the products

of the centered functional responses used for the covariance estimation. She implemented the approach in the R add-on package `sparseFLMM` (Cederbaum, 2016) with the assistance of Fabian Scheipl, who advised on the implementation of the novel smoothing class. Jona Cederbaum planned and carried out all simulations and the data analyses. She conducted the literature research on covariance estimation approaches and wrote the manuscript assisted by input from the two other authors. All three authors contributed in proofreading the manuscript.

Except for minor changes, mainly concerning notation and wording, this chapter and the paper Cederbaum et al. (2018) match. For a better understanding, an additional example was added to the appendix of this chapter, which can be found in Section D.2 in Appendix D. The examples for the model specifications yielding functional linear mixed models with crossed and hierarchical functional random effects were moved from the appendix of the manuscript to Chapter 2.

Software

The analyses within this chapter were carried out using R version 3.2.3 (2015-12-10) (R Core Team, 2016) on the two platforms `x86_64-pc-linux-gnu` (64-bit) and `x86_64-w64-mingw32/x64` (64-bit). The add-on packages `orthopolynom` (Novomestky, 2013, version 1.0-5), `mvtnorm` (Genz et al., 2014, version 0.9-99992), and `expm` (Goulet et al., 2017, version 0.99-1.1) were used for the generation of the simulated data. The add-on package `data.table` (Dowle and Srinivasan, 2015, version 1.9.6) was used for fast and memory efficient data manipulation. For matrix operations, the add-on packages `Matrix` (Bates and Mächler, 2017, versions 1.2-3 and 1.2-4) and `MASS` (Venables and Ripley, 2002, versions 7.3-35 and 7.3-45) were used. For smoothing the mean and covariance functions, the add-on package `mgcv` (Wood, 2006, 2011, versions 1.8-11 and 1.8-12) was applied, for which a novel smoothing class for bivariate smooths estimated subject to a symmetry constraint is proposed in this chapter. The package `mgcv` depends on the add-on package `nlme` (Pinheiro et al., 2016, attached versions 3.1-124 and 3.1-126). For parallelization of the simulation runs, the add-on packages `foreach` (Analytics and Weston, 2015b, version 1.4.3) and `doMC` (Analytics and Weston, 2015a, version 1.3.4) were used, the latter of which depends on the add-on package `iterators` (Analytics and Weston, 2015c, attached version 1.0.8). For comparison with the symmetric smoothing approach of Xiao et al. (2017), the add-on package `face` (Xiao et al., 2016a, version 0.1-1) was employed. The CD4 cell count data analyzed in this chapter were taken from the add-on package `refund` (Huang et al., 2016a, version 0.1-14).

5.1 Introduction

Covariance functions play a central role in many areas of statistics. They summarize the dependency between stochastic observations and encode smoothness assumptions about (observed or latent) random processes. We propose a fast bivariate smoothing approach for symmetric surfaces which can estimate covariance functions in a wide range of data situations. Our approach can handle dependent processes based on an additive decomposition of the covariance function and is also applicable to processes that are observed on irregular or sparse grids.

In functional data analysis (FDA; see, e.g., Ramsay and Silverman, 2005), covariance functions are at the heart of functional principal component analysis (FPCA), a key tool for dimension reduction based on an eigen analysis of the covariance operator of a random process. FPCA is commonly used to estimate the model parameters in functional predictor and functional response regression models (for an overview, see Morris, 2015). Other examples that are based on covariance functions include functional discriminant analysis (James and Hastie, 2001) and functional canonical correlation analysis (Leurgans et al., 1993). In longitudinal data analysis (LDA), where measurements are frequently recorded at irregularly spaced time points, the correct specification of the covariance benefits the estimation efficiency of the fixed effects and improves the individual predictions (cf. Fan et al., 2007). The covariance is also a crucial ingredient in time series analysis, e.g., in risk models and portfolio allocation (cf. Tai, 2009). The interest commonly lies in a single time series in contrast to FDA (and LDA) where multiple curves are observed, e.g., over time. In principle, our symmetric smoothing approach is also applicable to time series which is, however, not the focus in this chapter.

Covariance functions are commonly assumed to be smooth. Thus, when the observed curves are not sufficiently smooth (i.e., observed with error) or not measured on a common dense grid, smoothing becomes necessary at some point during covariance estimation. Directly smoothing the observed curves (see, e.g., Besse and Ramsay, 1986), however, is very difficult or impossible for sparsely observed data which are frequently recorded both in FDA and LDA (Yao et al., 2005). Moreover, pre-smoothing the observed curves removes the measurement error, which is not accounted for in subsequent estimation steps. We pursue an alternative approach and apply bivariate smoothing to the sample covariance of the observed data points.

Most existing work on non-parametric covariance estimation is either restricted to independent functional (or longitudinal) observations and/or only applies to data sampled on a common grid. Furthermore, most bivariate smoothing approaches are not specifically designed for covariances. They do not exploit the symmetry of the estimated surface and thus use redundant information in the available data. To the best of our knowledge, previous approaches have never addressed these issues simultaneously. They can be divided according to three main criteria: 1) the generality of the assumed correlation structure in the data, 2) the generality of possible sampling grids, and 3) the estimation procedure including the selection of the degree of smoothing.

A number of approaches address covariance smoothing in LDA. They are restricted to independent curves but allow for general sampling grids. Smoothing is either accomplished by bivariate kernel smoothing (e.g., Staniswalis and Lee, 1998; Yao et al., 2003, 2005) or by bivariate (penalized) spline smoothing (e.g., Kauermann and Wegener, 2011). The degree of smoothing is either chosen by visual inspection (Staniswalis and Lee, 1998), different leave-one-curve-out cross-validation algorithms (e.g., Yao et al., 2003, 2005) or based on a mixed model representation (e.g., Kauermann and Wegener, 2011). These approaches do not account for the symmetry of the estimated surface. James et al. (2000) directly estimate the smooth eigenfunctions of the covariance function. They estimate a reduced rank mixed effects model via the EM algorithm and use B-spline basis functions to represent the eigenfunctions of the covariance operator. Peng and Paul (2009) estimate the same reduced rank model based on a more efficient Newton-Raphson procedure on the Stiefel manifold. The extension

of these reduced rank methods to complex correlation structures is not straightforward. Xiao et al. (2017) recently proposed a bivariate smoother designed for covariance smoothing which can be used for sparsely observed, independent functions. They use bivariate penalized B-splines and enforce a symmetry constraint on the spline coefficients which we take up in our extension to correlated curves. Estimation is done by a three-step procedure which accounts for the covariance of the sample covariance. Their leave-one-curve-out cross-validation procedure for selecting the smoothing parameter is not applicable for correlated functional data, however.

Other covariance smoothing approaches can be applied to correlated functions but are restricted to functions sampled on a common grid and considerably simpler correlation structures than ours. Di et al. (2009) and Greven et al. (2010) use bivariate penalized splines and select the smoothing parameter using restricted maximum likelihood (REML; Patterson and Thompson, 1971) estimation. Shou et al. (2015) apply a method of moments approach based on symmetric sums represented in a sandwich form. For smoothing, they propose to use an extension of the fast covariance estimation algorithm of Xiao et al. (2016b) to correlated functions. Di et al. (2014) extend the functional random intercept model of Di et al. (2009) to sparsely sampled functional data, but the correlation structure remains less general than ours and an extension is not straightforward. More general correlation structures are allowed in the approach in Chapter 4 of this thesis that is also suitable for sparsely and irregularly sampled functional data. The focus in Chapter 4 lies, however, on a model with crossed functional random effects and estimation is only discussed for this special case. Apart from considering less general correlation structures, all these approaches neither avoid the use of redundant information nor account for the symmetry of the smoothed surface.

We propose a fast symmetric bivariate smoothing approach that applies to data with a broad range of possible correlation structures, much broader than existing methods. Furthermore, our approach is well-suited for (possibly noisy) data sampled on a common, dense grid as well as for irregularly or sparsely sampled data. Strength is borrowed by pooling information across different curves, which is particularly important for curves observed on sparse, unequal grids. The smoothing approach we present is widely applicable: In this chapter, we demonstrate how it can be applied to longitudinal data as a special case of independent functional data as well as to correlated functional data with very general and complex correlation structures. For the latter, we extend our bivariate smoothing approach to smoothing additive covariance functions. To the best of our knowledge, all previous proposals in this field have been restricted to estimating much less general dependency structures.

We estimate the covariance functions using a smooth method of moments approach represented as a bivariate additive varying coefficient model. The estimation is based on bivariate penalized splines. We choose the smoothing parameters using REML, which allows the direct extension to additive bivariate smoothing of a superposition of multiple covariance functions. This allows our method to be used for a broad range of complex real-world data settings. It also frees us from having to pre-specify a discrete grid of candidate values for the smoothing parameters that is required for cross-validation-based approaches like in Xiao et al. (2017). Smoothing the sample covariance quickly becomes a high-dimensional problem as the number of elements in the sample covariance increases quadratically with the number of grid points. We take advantage of the symmetry of the sample covariance and only

estimate the upper triangle of the surface including the diagonal. The estimates are then reflected across the diagonal to obtain the entire estimated covariance, which is continuous but not necessarily smooth across the diagonal. To avoid boundary effects on the diagonal and to ensure identifiability of our models, we enforce smoothness across the diagonal by imposing a symmetry constraint on the spline coefficients, which for the simplest case of independent curves reduces to that of Xiao et al. (2017). We show how the symmetry constraint can be applied separately to additive covariances and can even be used for any bivariate symmetric smoothing problem beyond covariance functions. Our approach modifies the covariance smoothing approach proposed in Chapter 4 of this thesis and extends it to more general models. It reduces both the data entering the estimation and the number of spline coefficients that have to be estimated, which leads to considerably faster estimation requiring less memory.

We provide documented open-source software implementing our approach in the R add-on package `sparseFLMM` (R Core Team, 2016; Cederbaum, 2016). The implementation is based on a novel constructor function for the R add-on package `mgcv`, which provides a general framework for additive models allowing for a very flexible model specification (Wood, 2006, 2011). A description and examples for the usage of the R package `sparseFLMM` can be found in Appendix E.

We outline the application of our approach to FPCA and demonstrate its practical relevance by an application to sparse longitudinal observations of CD4 cell count trajectories and to densely but irregularly observed acoustic signals from a speech production study. This study requires crossed functional random effects due to repeated measurements for both speakers and target words and thus corresponds to a case of dependent functional data with an additive covariance structure. Both data analyses are fully reproducible as the speech production data are included in the R add-on package `sparseFLMM` and the CD4 cell count data are available in the R add-on package `refund` (Huang et al., 2016a).

This chapter is organized as follows: Section 5.2 first develops our fast symmetric covariance smoothing approach for a simple special case with only one smooth covariance function and additional measurement error. In Section 5.3, the smoother is extended to complex dependency structures involving the smoothing of multiple additive covariance functions. Section 5.4 outlines the application of our covariance smoother in FPCA. In Section 5.5, details on the implementation are given. In Section 5.6 and 5.7, we evaluate our approach in an application to speech production data and in simulations, respectively (a longitudinal application to CD4 cell counts is given in Appendix D). Section 5.8 closes with a discussion and outlook. Theoretical results and supplementary material are available in Appendix D.

5.2 Fast symmetric covariance smoothing

For simplicity, we first explain our covariance smoothing approach for a simple special case with only one smooth auto-covariance and additional measurement error. This will be extended to additive covariance smoothing in Section 5.3.

5.2.1 Model with independent curves

Consider the following model

$$Y_i(t_{ij}) = \mu(t_{ij}, \mathbf{x}_i) + E_i(t_{ij}) + \varepsilon_i(t_{ij}), \quad j = 1, \dots, D_i, \quad i = 1, \dots, n, \quad (5.1)$$

where $Y_i(t_{ij})$ is the value of response curve i at the j th observation point $t_{ij} \in \mathcal{T}$, a bounded interval in \mathbb{R} . $\mu(t_{ij}, \mathbf{x}_i)$ is a global mean function depending on a vector of known covariates \mathbf{x}_i . $E_i(t_{ij})$ is a smooth curve-specific deviation from the global mean and $\varepsilon_i(t_{ij})$ is additional independent and identically distributed white noise measurement error with constant variance σ^2 that accounts for random uncorrelated variation within curve i . The model can be seen as a function-on-scalar regression model (e.g., Faraway, 1997; Ramsay and Silverman, 2005; Reiss et al., 2010) where all n curves are assumed to be independent and it is a special case of the general functional linear mixed model (FLMM; for a discussion and further references, see Morris, 2015) in Section 5.3 with only curve-specific smooth residuals. Model (5.1) is often applied to longitudinal data with \mathcal{T} denoting a time interval. Note that all curves may either be observed on a common, fine grid or on curve-specific, possibly sparse, D_i evaluation points t_{ij} , $j = 1, \dots, D_i$, $i = 1, \dots, n$.

In the following, we assume that $E_i(\cdot)$ and $\varepsilon_i(\cdot)$, $i = 1, \dots, n$, are zero mean, mutually uncorrelated random processes and that $E_i(\cdot)$ is square integrable. We denote the auto-covariance function of $E_i(\cdot)$ by $K^E(t, t') = \text{Cov}[E_i(t), E_i(t')]$, $t, t' \in \mathcal{T}$. We further assume that the mean and the auto-covariance are smooth in t and in arguments t, t' , respectively.

5.2.2 Estimation in the independent case

We apply the following smooth method of moments approach to estimate the auto-covariance $K^E(t, t')$. It modifies the approach presented in Chapter 4 by accounting for the symmetry of covariances, which leads to a considerable reduction of computation times. In this simple case of independent functional responses, our smoother is closely related to that of Xiao et al. (2017), who approach the problem from a slightly different perspective. While their focus is mainly on the symmetry constraint for the spline coefficients and on the development of a fast smoothing parameter estimation, our aim is to avoid redundant information in symmetric smoothing which in this simple case leads us to the same symmetry constraint. Their approach, however, is not directly extendable to correlated data as will be discussed at the end of this section.

We focus in the following on the centered functional responses $\tilde{Y}_i(t_{ij}) := Y_i(t_{ij}) - \mu(t_{ij}, \mathbf{x}_i)$ with expectation zero and denote their realizations by $\tilde{y}_{it_{ij}}$. We exploit the fact that the expectation of the centered products $\tilde{Y}_i(t_{ij})\tilde{Y}_i(t_{ij'})$ corresponds to the covariance of the functional response which is given as

$$\begin{aligned} \mathbb{E} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_i(t_{ij'}) \right] &= \text{Cov} [Y_i(t_{ij}), Y_i(t_{ij'})] = K^E(t_{ij}, t_{ij'}) + \sigma^2 \delta_{jj'}, \\ & \quad j, j' = 1, \dots, D_i, \quad i = 1, \dots, n, \end{aligned} \quad (5.2)$$

with $\delta_{jj'}$ equal to one if $j = j'$ and zero otherwise. Equation (5.2) can be seen as a special case of a bivariate additive varying coefficient model for the empirical covariances $\tilde{y}_{it_{ij}}\tilde{y}_{it_{ij'}}$, in which the auto-covariance and the error variance are the unknown components. We estimate the smooth auto-covariance function $K^E(t, t')$ and the error variance σ^2 simultaneously using a bivariate spline representation for $K^E(t, t')$ under working assumptions of independence and homoscedasticity.

For this, let \mathbf{C} denote the $\mathcal{C} \times 1$ stacked vector of all centered products, with $\mathcal{C} = \sum_{i=1}^n D_i^2$. Then, Model (5.2) can be represented as

$$\mathbb{E}(\mathbf{C}) = [\mathbf{M}^E | \delta^\varepsilon] \left(\boldsymbol{\theta}^{E\top}, \sigma^2 \right)^\top =: \mathbf{M}\boldsymbol{\alpha}, \quad (5.3)$$

where \mathbf{M}^E denotes the $\mathcal{C} \times (F^E)^2$ bivariate spline design matrix, containing the evaluations of any bivariate spline basis with $(F^E)^2$ basis functions that are symmetric across the diagonal, i.e., across $t_{ij} = t_{ij'}$. We use bivariate tensor product B-splines, but other bases are possible. See Section 5.5 and Section D.2 in Appendix D for details. δ^ε is an indicator vector of length \mathcal{C} whose elements take values $\delta_{jj'}$. $\boldsymbol{\theta}^E$ is a spline coefficient vector of length $(F^E)^2$. To avoid over-fitting, we use an isotropic quadratic smoothness penalty of the form

$$\text{pen}(\lambda) = \lambda \boldsymbol{\theta}^{E\top} \mathbf{S}^E \boldsymbol{\theta}^E, \quad (5.4)$$

where λ denotes the smoothing parameter that controls the bias-variance tradeoff and \mathbf{S}^E is a suitable penalty matrix of dimension $(F^E)^2 \times (F^E)^2$, see Section 5.5 for a discussion.

The development above uses all \mathcal{C} centered products for the estimation of the covariance as is commonly done in published smooth method of moments approaches for covariance estimation (Yao et al., 2003; Yao et al., 2005; for correlated curves, e.g., Staniswalis and Lee, 1998; Di et al., 2009; Greven et al., 2010). As it is quadratic in the number of function evaluations, \mathcal{C} quickly becomes extremely large in practice and often poses significant computational challenges. Since the products $\tilde{y}_{it_{ij}}\tilde{y}_{it_{ij'}}$ and $\tilde{y}_{it_{ij'}}\tilde{y}_{it_{ij}}$ are identical, we, like Xiao et al. (2017), avoid the use of this redundant information and only estimate the upper triangle of the auto-covariance surface including the diagonal. Using the identical products only once also decreases the dependency among the products in \mathbf{C} and thus reduces the strongest violations of the implicit working assumption of independent observations in our additive model (5.3). A detailed discussion of working assumptions and possible strategies for handling violations is given in Section 5.3.4.

We assume in the following that \mathbf{C} is sorted such that it can be partitioned as $\mathbf{C} = (\mathbf{C}_{t < t'}^\top, \mathbf{C}_{t = t'}^\top, \mathbf{C}_{t > t'}^\top)^\top$, where $\mathbf{C}_{t < t'}$, $\mathbf{C}_{t = t'}$, $\mathbf{C}_{t > t'}$ comprise all products $\tilde{Y}_{it_{ij}}\tilde{Y}_{it_{ij'}}$, $i = 1, \dots, n$, with $t_{ij} < t_{ij'}$, $t_{ij} = t_{ij'}$, and $t_{ij} > t_{ij'}$, respectively. Then, with suitable sorting within the three partitions of \mathbf{C} , the symmetry of the products implies that $\mathbf{C}_{t < t'} = \mathbf{C}_{t > t'}$. In order to speed up estimation, we only use $\mathbf{C}^\Delta := (\mathbf{C}_{t < t'}^\top, \mathbf{C}_{t = t'}^\top)^\top$ for the estimation in the bivariate model (5.3). The total number of products thus amounts to $\mathcal{C}^\Delta := \sum_{i=1}^n D_i(D_i + 1)/2$. The design matrix of the bivariate additive model can accordingly be partitioned as $\mathbf{M} = (\mathbf{M}_{t < t'}^\top, \mathbf{M}_{t = t'}^\top, \mathbf{M}_{t > t'}^\top)^\top$. Let $\mathbf{M}^{E\Delta}$ and $\boldsymbol{\delta}^{E\Delta}$ denote the submatrix and subvector corresponding to \mathbf{C}^Δ . Then, the bivariate additive

model (5.3) reduces to $\mathbb{E}(\mathbf{C}^\Delta) = [\mathbf{M}^{E\Delta} | \boldsymbol{\delta}^{\varepsilon\Delta}] \left(\boldsymbol{\theta}^{E\top}, \sigma^2 \right)^\top$. Reflecting the upper triangle across the diagonal ensures that the obtained surface estimate is symmetric and continuous in both directions t and t' , but it does not guarantee smoothness across the diagonal without additional constraint.

Wiggly estimates in the area of the diagonal can occur as coefficients for basis functions near the diagonal have few observations available for estimation. This is because much of the support of the basis functions is on the other side of the diagonal, i.e., in the lower triangle that does not contain any data. An expected consequence is that the smooth auto-covariance surface and the error variance on the diagonal can be less well-separated, which we indeed observe in our application in Section 5.6 and in our simulations in Section 5.7.

We enforce smoothness across the diagonal in order to avoid such boundary effects. The assumption of smoothness across the diagonal guarantees identifiability of our model as the smooth auto-covariance $K^E(t_{ij}, t_{ij'})$ and the variance of the additional unsmooth error term $\varepsilon_i(t_{ij})$ can be separated. A symmetric surface implies a symmetric spline coefficient matrix

$$\boldsymbol{\Theta}^E = [\theta_{bb'}^E]_{b,b'=1,\dots,F^E} = \boldsymbol{\Theta}^{E\top},$$

where $\boldsymbol{\theta}^E = \left(\boldsymbol{\theta}_{b<b'}^E{}^\top, \boldsymbol{\theta}_{b=b'}^E{}^\top, \boldsymbol{\theta}_{b>b'}^E{}^\top \right)^\top$ contains first the entries of $\boldsymbol{\Theta}^E$ below the diagonal ($\theta_{bb'}^E$, $b < b'$), then the diagonal entries ($\theta_{bb'}^E$, $b = b'$) and lastly the entries above the diagonal ($\theta_{bb'}^E$, $b > b'$). We impose a symmetry constraint on $\boldsymbol{\Theta}^E$. Thus, our approach differs in two crucial points from the naive covariance estimation in (5.3) and from most previous covariance smoothing approaches. First, we reduce the number of products that enter the estimation and second, imposing the symmetry constraint almost halves the number of spline coefficients that have to be estimated. Both aspects greatly speed up the computation as we show in Sections 5.6 and 5.7.

With suitable sorting within the partitions $\boldsymbol{\theta}_{b<b'}^E$ and $\boldsymbol{\theta}_{b>b'}^E$, the above symmetry constraint on the coefficient matrix corresponds to the following symmetry constraint on the coefficient vector

$$\theta_{bb'}^E = \theta_{b'b}^E, \quad b, b' = 1, \dots, F^E \quad \Leftrightarrow \quad \boldsymbol{\theta}_{b<b'}^E = \boldsymbol{\theta}_{b>b'}^E, \quad (5.5)$$

which corresponds to the constraint used in Xiao et al. (2017). This allows us to consider the reduced coefficient vector $\boldsymbol{\theta}^{Er} = \left(\boldsymbol{\theta}_{b<b'}^E{}^\top, \boldsymbol{\theta}_{b=b'}^E{}^\top \right)^\top$ of length $F^E (F^E + 1) / 2$. For the implementation of the above symmetry constraint, we use that under the constraint we have

$$\mathbf{M}^{E\Delta} \boldsymbol{\theta}^E = (\mathbf{M}_{b<b'}^{E\Delta} + \mathbf{M}_{b>b'}^{E\Delta}) \boldsymbol{\theta}_{b<b'}^E + \mathbf{M}_{b=b'}^{E\Delta} \boldsymbol{\theta}_{b=b'}^E,$$

with $\mathbf{M}_{b<b'}^{E\Delta}$, $\mathbf{M}_{b=b'}^{E\Delta}$, and $\mathbf{M}_{b>b'}^{E\Delta}$ containing the respective columns of $\mathbf{M}^{E\Delta}$. Thus, the constraint is equivalent to adding up columns $\mathbf{M}_{b<b'}^{E\Delta}$ and $\mathbf{M}_{b>b'}^{E\Delta}$ of the design matrix. This can be achieved by right multiplication of $\mathbf{M}^{E\Delta}$ with the $(F^E)^2 \times F^E (F^E + 1) / 2$ constraint matrix

$$\mathbf{W}^E = \begin{bmatrix} \mathbf{I}_{\frac{F^E(F^E-1)}{2}} & \mathbf{0}_{\frac{F^E(F^E-1)}{2} \times F^E} \\ \mathbf{0}_{F^E \times \frac{F^E(F^E-1)}{2}} & \mathbf{I}_{F^E} \\ \mathbf{I}_{\frac{F^E(F^E-1)}{2}} & \mathbf{0}_{\frac{F^E(F^E-1)}{2} \times F^E} \end{bmatrix},$$

where \mathbf{I}_x is an identity matrix of dimension x and $\mathbf{0}_{x \times y}$ is a null matrix of dimension $x \times y$. We denote the reduced design matrix by $\mathbf{M}^{E\Delta r} := \mathbf{M}^{E\Delta} \mathbf{W}^E$. Under the symmetry constraint, $\mathbf{M}^{E\Delta r} \boldsymbol{\theta}^{Er} = \mathbf{M}^{E\Delta} \boldsymbol{\theta}^E$. The penalty matrix in (5.4) also needs to be adjusted to the reduced coefficient vector $\boldsymbol{\theta}^{Er}$ as $\mathbf{S}^{Er} := \mathbf{W}^{E\top} \mathbf{S}^E \mathbf{W}^E$, corresponding to $\boldsymbol{\theta}^{Er\top} \mathbf{S}^{Er} \boldsymbol{\theta}^{Er} = \boldsymbol{\theta}^{E\top} \mathbf{S}^E \boldsymbol{\theta}^E$.

Altogether, the bivariate additive model for the reduced response vector with symmetry constraint (5.5) is given by

$$\mathbb{E}(\mathbf{C}^\Delta) = [\mathbf{M}^{E\Delta r} | \boldsymbol{\delta}^{\varepsilon\Delta}] \left(\boldsymbol{\theta}^{Er\top}, \sigma^2 \right)^\top =: \mathbf{M}^{\Delta r} \boldsymbol{\alpha}^r. \quad (5.6)$$

Note that in Model (5.6), each product $\tilde{y}_{it_{ij}} \tilde{y}_{it_{ij}'}, t_{ij} \leq t_{ij}'$, enters the estimation with the same weight. This is not the case when all products $\tilde{y}_{it_{ij}} \tilde{y}_{it_{ij}'}, j, j' = 1, \dots, D_i$, are used as in Model (5.3), where all products appear twice except for those on the diagonal ($t_{ij} = t_{ij}'$). Our implementation in the R add-on package `sparseFLMM` allows to estimate Model (5.6) with the same weights as in Model (5.3) by putting a weight of 0.5 on the products on the diagonal. There is room for debate on whether it is desirable to down-weight the data on the diagonal compared to the rest. In any case, our simulations in Section 5.7 show that the difference is not very large.

In contrast to Xiao et al. (2017), who derive a leave-one-curve-out generalized cross-validation (GCV) algorithm to choose the smoothing parameter for independent curves, we choose the smoothing parameter as variance component ratio using REML. REML has been shown to be more stable than GCV and to result in lower mean squared errors (Reiss and Ogden, 2009; Wood, 2011). Moreover, Krivobokova and Kauermann (2007) have shown that REML-based smoothing parameter selection is less sensitive towards misspecifications of the correlation structure than prediction error methods. Even more importantly, using REML allows us to directly extend our symmetric smoothing approach to additive smoothing needed for functional data with complex dependency structures as will be shown in Section 5.3. In more general designs, where the responses cannot be decomposed into independent subvectors, it is not clear how to perform smoothing parameter selection based on GCV and optimizing multiple smoothing parameters would require a computationally costly multi-dimensional grid search.

Details on the implementation in the R add-on package `sparseFLMM` are given in Section 5.5 and in Section D.2 in Appendix D.

5.3 Fast symmetric additive covariance smoothing

Simultaneous REML estimation of multiple smoothing parameters allows the direct extension of our approach to more general models with complex correlation structures, for which we derive appropriate symmetry constraint matrices.

5.3.1 General functional linear mixed model

The general FLMM (see, e.g., Morris, 2015) can be seen as the functional analogue of the linear mixed model (LMM; see, e.g., Pinheiro and Bates, 2000), which is often applied to scalar correlated data. The random effects in the linear mixed model are replaced by functional random effects in order to account for the functional nature of the response. A functional random intercept (fRI) for a subject, for example, is a subject-specific deviation from the mean in the form of a function. The FLMM is given by

$$Y_i(t_{ij}) = \mu(t_{ij}, \mathbf{x}_i) + \mathbf{z}_i^\top \mathbf{U}(t_{ij}) + E_i(t_{ij}) + \varepsilon_i(t_{ij}), \quad j = 1, \dots, D_i, \quad i = 1, \dots, n, \quad (5.7)$$

where $Y_i(t_{ij})$ denotes the response of curve i at observation point t_{ij} , which can be additively decomposed as in Model (5.1). Model (5.7), however, additionally accounts for correlation between (groups of) curves by the vector-valued random process $\mathbf{U}(t_{ij})$ which is multiplied by \mathbf{z}_i , a known covariate vector of length q . Examples for $\mathbf{z}_i^\top \mathbf{U}(t_{ij})$ yielding FLMMs with, e.g., crossed and hierarchical functional random effects are given in Section 5.3.3 and in Section 2.1.2 of Chapter 2.

We assume that $\mathbf{U}(\cdot)$, $E_i(\cdot)$, and $\varepsilon_i(\cdot)$ are zero mean, mutually uncorrelated random processes and that $\mathbf{U}(\cdot)$ and $E_i(\cdot)$ are square integrable. As for Model (5.1), we denote the auto-covariance of $E_i(\cdot)$ by $K^E(t, t') = \text{Cov}[E_i(t), E_i(t')]$, $t, t' \in \mathcal{T}$. The $q \times q$ matrix-valued auto-covariance of $\mathbf{U}(\cdot)$ is denoted by $\mathbf{K}^U(t, t') = \text{Cov}[\mathbf{U}(t), \mathbf{U}(t')]$. The covariances are assumed to be smooth (for each component in the case of $\mathbf{U}(t)$).

Let G denote the number of grouping factors. Then, $\mathbf{U}(t_{ij})$ can be divided into G independent blocks $\mathbf{U}_g(t_{ij})$, $g = 1, \dots, G$, which again contain blocks of L^{U_g} independent copies $\mathbf{U}_{gl}(t_{ij})$, $l = 1, \dots, L^{U_g}$, where L^{U_g} is the number of levels of the g th grouping factor. $\mathbf{U}_{gl}(t_{ij}) = \left[U_{gl1}(t_{ij}), \dots, U_{gl\rho^{U_g}}(t_{ij}) \right]^\top$, in turn, is a vector-valued random process of ρ^{U_g} components for each level of this grouping factor, for example $\rho^{U_g} = 2$ if the g th grouping factor is associated with a fRI and a functional random slope. The total number of entries in $\mathbf{U}(t_{ij})$ is given by $q = \sum_{g=1}^G L^{U_g} \rho^{U_g}$. The $\rho^{U_g} \times \rho^{U_g}$ matrix-valued covariance of $\mathbf{U}_{gl}(\cdot)$, $\mathbf{K}^{U_g}(t, t') = \left[K_{ss'}^{U_g}(t, t') \right]_{s, s'=1, \dots, \rho^{U_g}} = \text{Cov}[\mathbf{U}_{gl}(t), \mathbf{U}_{gl}(t')]$, with $K_{ss'}^{U_g}(t, t') = K_{s's}^{U_g}(t', t)$, is the same for all levels, $l = 1, \dots, L^{U_g}$, of the g th grouping factor. We can thus write the block diagonal auto-covariance of $\mathbf{U}(\cdot)$ as

$$\mathbf{K}^U(t, t') = \text{diag} \left[\underbrace{\mathbf{K}^{U_1}(t, t'), \dots, \mathbf{K}^{U_1}(t, t')}_{L^{U_1} \text{ times}}, \dots, \underbrace{\mathbf{K}^{U_G}(t, t'), \dots, \mathbf{K}^{U_G}(t, t')}_{L^{U_G} \text{ times}} \right].$$

5.3.2 Estimation in the general functional linear mixed model

Our fast symmetric covariance smoothing approach can be extended to the general model (5.7) by generalizing it to a matrix of covariances as described above and applying it to each additive component separately.

In analogy to Model (5.1), we base the covariance smoothing on the following decomposition of the expectation of the products of the centered functional responses

$$\begin{aligned} \mathbb{E} \left[\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'}) \right] &= \text{Cov} [Y_i(t_{ij}), Y_{i'}(t_{i'j'})] \\ &= \mathbf{z}_i^\top \mathbf{K}^U(t_{ij}, t_{i'j'}) \mathbf{z}_{i'} + [K^E(t_{ij}, t_{i'j'}) + \sigma^2 \delta_{jj'}] \delta_{ii'}, \end{aligned} \quad (5.8)$$

where, in contrast to Model (5.1), products are now also computed across different curves i, i' .

Let $\ell_g(i)$ denote the level of grouping factor g for observation i . Then, similar to $\mathbf{U}(\cdot)$, the covariate vector \mathbf{z}_i can be divided into G blocks $\mathbf{z}_i^\top = (\mathbf{z}_i^{U_1^\top}, \dots, \mathbf{z}_i^{U_G^\top})$, where the blocks $\mathbf{z}_i^{U_g}$, $g = 1, \dots, G$, can again be written as $\mathbf{z}_i^{U_g^\top} = (\mathbf{z}_{i1}^{U_g^\top}, \dots, \mathbf{z}_{iL^{U_g}}^{U_g^\top})$ with $\mathbf{z}_{il}^{U_g^\top} = (z_{il1}^{U_g}, \dots, z_{il\rho^{U_g}}^{U_g})$, $l = 1, \dots, L^{U_g}$. The scalars $z_{ils}^{U_g}$ take the value of the respective covariate $\omega_{is}^{U_g}$ times an indicator $\delta_{\ell_g(i)l}$, specifying whether observation i belongs to level l of grouping factor g . Based on this partition, the expectation in Model (5.8) can be rewritten as

$$\begin{aligned} \mathbb{E} \left[\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'}) \right] &= \sum_{g=1}^G \sum_{l=1}^{L^{U_g}} \sum_{s=1}^{\rho^{U_g}} \sum_{s'=1}^{\rho^{U_g}} z_{ils}^{U_g} z_{i'l's'}^{U_g} K_{ss'}^{U_g}(t_{ij}, t_{i'j'}) \\ &+ [K^E(t_{ij}, t_{i'j'}) + \sigma^2 \delta_{jj'}] \delta_{ii'}. \end{aligned} \quad (5.9)$$

For example in the case of an FLMM with only one fRI ($G = 1, \rho^{U_1} = 1$), the products $z_{ils}^{U_g} z_{i'l's'}^{U_g}$ yield indicators for whether the two observations in the product $\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'})$ belong to the same level of grouping factor g .

We exploit the symmetry of covariances $\mathbf{K}^{U_g}(t_{ij}, t_{i'j'}) = \mathbf{K}^{U_g}(t_{i'j'}, t_{ij})^\top$, $g = 1, \dots, G$, and of $K^E(t_{ij}, t_{i'j'}) = K^E(t_{i'j'}, t_{ij})$ and only use the products $\tilde{Y}_{it_{ij}} \tilde{Y}_{i't_{i'j'}}$ with $t_{ij} \leq t_{i'j'}$ if $i \leq i'$ and $t_{ij} < t_{i'j'}$, otherwise, suitably sorted in the long vector \mathbf{C}^Δ . Let \cdot denote the Hadamard (point-wise) product. As in the case with independent curves, Model (5.9) can be represented as bivariate additive varying coefficient model, here of the form

$$\mathbb{E}(\mathbf{C}^\Delta) = [\mathbf{M}^{U_1^\Delta} | \dots | \mathbf{M}^{U_G^\Delta} | \mathbf{M}^{E^\Delta} | \delta^{\varepsilon^\Delta}] (\boldsymbol{\theta}^{U_1^\top}, \dots, \boldsymbol{\theta}^{U_G^\top}, \boldsymbol{\theta}^{E^\top}, \sigma^2)^\top, \quad (5.10)$$

where $\mathbf{M}^{U_g^\Delta}$, $g = 1, \dots, G$, contain the column-wise concatenated submatrices $\mathbf{M}_{ss'}^{U_g^\Delta}$, corresponding to the covariances $K_{ss'}^{U_g}(t, t')$, $s, s' = 1, \dots, \rho^{U_g}$. The submatrices $\mathbf{M}_{ss'}^{U_g^\Delta}$ are given by $\mathbf{Q}_{ss'}^{U_g^\Delta} \cdot \mathbf{B}_{ss'}^{U_g^\Delta}$, where $\mathbf{Q}_{ss'}^{U_g^\Delta}$ contain suitably sorted and repeated entries $\delta_{\ell_g(i)\ell_g(i')} \cdot \omega_{is}^{U_g} \omega_{i's'}^{U_g}$, where $\delta_{\ell_g(i)\ell_g(i')}$ take value one if the two curves i and i' are of the same level of grouping factor g and zero otherwise. $\mathbf{B}_{ss'}^{U_g^\Delta}$

denote the bivariate spline design matrices. $\mathbf{M}^{E\Delta}$ is analogously given by $\mathbf{Q}^{E\Delta} \cdot \mathbf{B}^{E\Delta}$, with bivariate spline design matrix $\mathbf{B}^{E\Delta}$ and $\mathbf{Q}^{E\Delta}$ a new indicator matrix, which reduces to an all-ones matrix in the model with independent curves, for which thus $\mathbf{M}^{E\Delta} = \mathbf{B}^{E\Delta}$. The concrete form of $\mathbf{Q}_{ss'}^{U_g\Delta}$ and $\mathbf{Q}^{E\Delta}$, an example for the case of one grouping factor with two levels, as well as the bivariate spline design matrices for tensor product B-splines are provided in Section D.2 in Appendix D.

We assume that for each $g = 1, \dots, G$, the coefficient vector $\boldsymbol{\theta}^{U_g}$ is sorted with the order corresponding to the columns of $\mathbf{M}^{U_g\Delta} = \left[\mathbf{M}_{11}^{U_g\Delta} \mid \dots \mid \mathbf{M}_{1\rho^{U_g}}^{U_g\Delta} \mid \dots \mid \mathbf{M}_{\rho^{U_g}1}^{U_g\Delta} \mid \dots \mid \mathbf{M}_{\rho^{U_g}\rho^{U_g}}^{U_g\Delta} \right]$. Moreover, with suitable sorting, each submatrix $\mathbf{M}_{ss'}^{U_g\Delta}$ can be partitioned as in the case of independent curves, $\mathbf{M}_{ss'}^{U_g\Delta} = \left[\mathbf{M}_{ss',b<b'}^{U_g\Delta} \mid \mathbf{M}_{ss',b=b'}^{U_g\Delta} \mid \mathbf{M}_{ss',b>b'}^{U_g\Delta} \right]$. Let $\boldsymbol{\Theta}_{ss'}^{U_g}$ denote the coefficient matrices, where the correspondingly sorted coefficient vectors $\boldsymbol{\theta}_{ss'}^{U_g} = \left(\boldsymbol{\theta}_{ss',b<b'}^{U_g\top}, \boldsymbol{\theta}_{ss',b=b'}^{U_g\top}, \boldsymbol{\theta}_{ss',b>b'}^{U_g\top} \right)^\top$ contain first the entries of the spline coefficient matrix $\boldsymbol{\Theta}_{ss'}^{U_g}$ below the diagonal, then the diagonal entries and lastly the entries above the diagonal. Assume further that within the three blocks of $\boldsymbol{\theta}_{ss'}^{U_g}$, the entries $\theta_{ss',bb'}$ are sorted correspondingly for all $s, s' = 1, \dots, \rho^{U_g}$.

As a modular component, the symmetry constraint

$$\boldsymbol{\Theta}_{ss'}^{U_g} = \boldsymbol{\Theta}_{s's}^{U_g\top}, \quad s, s' = 1, \dots, \rho^{U_g}, \quad (5.11)$$

can be applied to each, $g = 1, \dots, G$, due to the symmetry of covariances $\mathbf{K}^{U_g}(t, t')$, yielding the reduced coefficient vectors $\boldsymbol{\theta}_{ss'}^{U_g r}$, $s \leq s'$, and thus the reduced long coefficient vector $\boldsymbol{\theta}^{U_g r}$. As in the case of independent curves, the constraint (5.11) is equivalent to adding up the respective columns of the large design matrix $\mathbf{M}^{U_g\Delta}$. This can be achieved by right-multiplication of $\mathbf{M}^{U_g\Delta}$ with a suitable constraint matrix \mathbf{W}^{U_g} , yielding the reduced design matrix

$$\mathbf{M}^{U_g\Delta r} = \left[\mathbf{M}_{11}^{U_g\Delta r} \mid \dots \mid \mathbf{M}_{1\rho^{U_g}}^{U_g\Delta r} \mid \mathbf{M}_{22}^{U_g\Delta r} \mid \dots \mid \mathbf{M}_{\rho^{U_g}-1\rho^{U_g}-1}^{U_g\Delta r} \mid \mathbf{M}_{\rho^{U_g}-1\rho^{U_g}}^{U_g\Delta r} \mid \mathbf{M}_{\rho^{U_g}\rho^{U_g}}^{U_g\Delta r} \right].$$

Each $\mathbf{M}_{ss'}^{U_g\Delta r}$, $s \leq s'$, consists of column-wise concatenated matrices

$$\mathbf{M}_{ss'}^{U_g\Delta r} = \left[\mathbf{M}_{ss',b<b'}^{U_g\Delta} + \mathbf{M}_{s's,b>b'}^{U_g\Delta} \mid \mathbf{M}_{ss',b=b'}^{U_g\Delta} + \mathbf{M}_{s's,b=b'}^{U_g\Delta} \delta_{s<s'} \right], \quad s \leq s' = 1, \dots, \rho^{U_g}.$$

The constraint matrix \mathbf{W}^{U_g} consists of $(\rho^{U_g})^2 \times \left[(\rho^{U_g})^2 + 1 \right] / 2$ blocks, most of which are zero. The block rows of \mathbf{W}^{U_g} correspond to the constraint on the spline coefficients of the covariances $\mathbf{K}_{ss'}^{U_g}(t, t')$, $s, s' = 1, \dots, \rho^{U_g}$, sorted as in $\mathbf{M}^{U_g\Delta}$. The columns are sorted correspondingly to the reduced matrix $\mathbf{M}^{U_g\Delta r}$. For the auto-covariances ($s = s'$), the blocks are of the same form as the constraint matrix \mathbf{W}^E for independent curves. For the cross-covariances ($s < s'$), the blocks either correspond to diagonal block matrices or to anti-diagonal block matrices, depending on whether the respective rows correspond to $s < s'$ or $s > s'$, respectively. The specific form of \mathbf{W}^{U_g} and examples for $\rho^{U_g} = 2, 3$ are given in Section D.2 in Appendix D.

A quadratic smoothness penalty associated with each smooth term controls the bias-variance tradeoff. Each penalty matrix \mathbf{S}^{U_g} , consisting of blocks for each $K_{ss'}^{U_g}(t, t')$, is accordingly reduced by left- and right-multiplication with the constraint matrix \mathbf{W}^{U_g} . Smoothing the components in the upper triangle of $\mathbf{K}^{U_g}(t, t')$ separately allows to define different penalties for the auto-covariances and the cross-covariances, respectively. In particular, it is possible to apply anisotropic penalties for the cross-covariances.

Reflecting the estimated triangular covariance surfaces across the diagonal yields estimates for the whole covariance surfaces $\mathbf{K}^{U_g}(t, t')$ and $K^E(t, t')$, with smoothness assured also across the diagonal. Note that reducing computation time becomes even more important when multiple covariances are smoothed simultaneously rather than a single one as in the case of independent functional data in Section 5.2.

5.3.3 Functional linear mixed model with crossed random intercepts

Motivated by our application to the speech production data in Section 5.6.2, we now illustrate the specification of the FLMM for the special case of an FLMM with crossed fRIs accounting for the repeated measurements on two grouping factors (e.g., speakers and target words) in a crossed design. In this model, we have $G = 2$ grouping factors with $\rho^{U_1} = \rho^{U_2} = 1$ associated random effects for each grouping factor, i.e., one fRI each. L^{U_1} , L^{U_2} are the numbers of levels of the first (e.g., speakers) and second (e.g., target words) grouping factor, respectively. The covariate vector \mathbf{z}_i only consists of indicators taking value one or zero to code group membership for the two grouping factors. The explicit specification of the covariate vector for crossed and hierarchical functional random effects is given in Section D.2 in Appendix D. For better readability, we rename in the following the components of the vector-valued random process as $B := U_1$ and $C := U_2$. The total number of components in $\mathbf{U}(t_{ij})$ is $q = L^B + L^C$. Note that this model corresponds to Model (4.2) in Chapter 4.

For this model, Equation (5.9) can be simplified as

$$\begin{aligned} \mathbb{E} \left[\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'}) \right] &= K^B(t_{ij}, t_{i'j'}) \delta_{\ell_1(i)\ell_1(i')} \\ &+ K^C(t_{ij}, t_{i'j'}) \delta_{\ell_2(i)\ell_2(i')} \\ &+ [K^E(t_{ij}, t_{i'j'}) + \sigma^2 \delta_{jj'}] \delta_{ii'}, \end{aligned} \quad (5.12)$$

where $\delta_{\ell_1(i)\ell_1(i')}$ and $\delta_{\ell_2(i)\ell_2(i')}$ take value one when the two curves i and i' belong to the same level of the respective grouping factor and zero otherwise. As can be seen in Equation (5.12), the products for which neither $\delta_{\ell_1(i)\ell_1(i')}$ nor $\delta_{\ell_2(i)\ell_2(i')}$ equals one do not have to be considered due to expectation zero. Equation (5.10) then reduces to

$$\mathbb{E}(\mathbf{C}^\Delta) = [\mathbf{M}^{B\Delta} | \mathbf{M}^{C\Delta} | \mathbf{M}^{E\Delta} | \boldsymbol{\delta}^{\varepsilon\Delta}] \left(\boldsymbol{\theta}^B{}^\top, \boldsymbol{\theta}^C{}^\top, \boldsymbol{\theta}^E{}^\top, \sigma^2 \right)^\top \quad (5.13)$$

and the symmetry constraint can be applied to each $\boldsymbol{\theta}^B$, $\boldsymbol{\theta}^C$, and $\boldsymbol{\theta}^E$.

5.3.4 Covariance of the products of the centered functional responses

We estimate the auto-covariances as unknown, smooth functions in a bivariate additive varying coefficient model using a quadratic loss function. Since this is equivalent to a penalized likelihood criterion for Gaussian data, we implicitly assume independence of the products of the centered functional responses with homoscedastic Gaussian measurement error. As already mentioned in Section 5.2.2 and shortly discussed in Chapter 4, these are working assumptions which do not hold as the products frequently involve two points on the same curve or on correlated curves. Nevertheless, these implicit assumptions are made by many existing works (e.g., Yao et al., 2005; Di et al., 2009; Greven et al., 2010). We now briefly discuss how the covariance of the products can be accounted for.

For the model with independent curves (5.1), Xiao et al. (2017) derive an expression for the covariance of the products in terms of $K^E(t, t')$ and the error variance σ^2 under the assumption of Gaussian responses. They apply a three-step algorithm in which they first estimate $K^E(t, t')$ and σ^2 under the working assumptions. Second, they estimate the covariance of the products by plugging in the estimates for $K^E(t, t')$ and σ^2 . In the third step, they re-estimate $K^E(t, t')$ and σ^2 using the estimated covariance of the products as a working covariance.

We derive an expression for the covariance of the products for the general model (5.7) based on results from Isserlis (1918) on fourth moment rules for multivariate Gaussian random variables. The covariance of the products can be written as

$$\begin{aligned}
& \text{Cov} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_{i'}(t_{i'j'}), \tilde{Y}_m(t_{mo})\tilde{Y}_{m'}(t_{m'o'}) \right] \\
&= \left\{ \mathbf{z}_i^\top \mathbf{K}^U(t_{ij}, t_{mo}) \mathbf{z}_m + [K^E(t_{ij}, t_{mo}) + \sigma^2 \delta_{jo}] \delta_{im} \right\} \\
&\quad \cdot \left\{ \mathbf{z}_{i'}^\top \mathbf{K}^U(t_{i'j'}, t_{m'o'}) \mathbf{z}_{m'} + [K^E(t_{i'j'}, t_{m'o'}) + \sigma^2 \delta_{j'o'}] \delta_{i'm'} \right\} \\
&+ \left\{ \mathbf{z}_i^\top \mathbf{K}^U(t_{ij}, t_{m'o'}) \mathbf{z}_{m'} + [K^E(t_{ij}, t_{m'o'}) + \sigma^2 \delta_{j'o'}] \delta_{im'} \right\} \\
&\quad \cdot \left\{ \mathbf{z}_{i'}^\top \mathbf{K}^U(t_{i'j'}, t_{mo}) \mathbf{z}_m + [K^E(t_{i'j'}, t_{mo}) + \sigma^2 \delta_{jo}] \delta_{i'm} \right\}.
\end{aligned} \tag{5.14}$$

The derivation and simplifications for the model with crossed fRIs are given in Section D.1 in Appendix D. The covariance (5.14) is a function of the unknown covariances $\mathbf{K}^U(t, t')$, $K^E(t, t')$ and σ^2 , giving rise to the need for an iterative procedure as proposed in Xiao et al. (2017) for the simpler model (5.1). Our implementation with the R add-on package `mgcv` allows to directly include the variance of the products by a specification of the `weights` argument in function `bam`. Our simulations showed, however, that accounting for only the heterogeneous variance does not lead to a substantial improvement in estimation accuracy. The dependencies could also be accounted for by pre-multiplication with the inverse square root of the covariance of the products, for which its construction and inversion would become necessary. For our application to the speech production data, the covariance matrix of the products would be a $52,156,566 \times 52,156,566$ dense and unstructured matrix whose construction is not feasible with current technology ($\approx 17,500$ Terabytes storage space would be required) and we thus do not focus on this extension in the following. However, Equation (5.14) allows the inclusion of

the covariance of the products in less complex settings and for (much) smaller data sets and can then result in more efficient estimates. Despite the violations of working assumptions, we achieve good results with our approach in simulations (cp. Section 5.7). Note that a relevant improvement of the covariance estimation can only be obtained if the working covariance is reasonably well-estimated and thus more than one iteration of the algorithm might be necessary. Especially for multiple iterations, our fast smoothing algorithm considerably speeds up the estimation compared to smoothing the entire covariances.

5.4 Application in functional principal component analysis

An important application of our fast symmetric additive smoothing approach is functional principal component analysis. FPCA is a key tool for dimension reduction in FDA that extracts the dominant modes of variation in the data and provides an explicit variance decomposition. In the following, we briefly outline the four main steps of FPCA for the general FLMM (5.7) using our newly proposed covariance estimation approach. Applying FPCA to Model (5.7) yields parsimonious representations of each random process $\mathbf{U}_g(t)$, $g = 1, \dots, G$, and $E_i(t)$, in bases of eigenfunctions of the respective, previously estimated smooth auto-covariances. In addition, we briefly describe how our covariance smoothing approach can be combined with the general framework of functional additive mixed models (FAMM; Scheipl et al., 2015) allowing for approximate statistical inference for the mean conditional on the FPCA. For a more detailed description, see Chapter 4.

In the first step, the smooth mean function is commonly estimated based on a working independence assumption (see, e.g., Yao et al., 2005; Di et al., 2009; Greven et al., 2010, and Chapter 4 of this thesis). We use penalized splines implemented in the R add-on package `mgcv`, which can be used to estimate a large variety of covariate and interaction effects in the mean function. Note that the combination with the FAMM framework in the fourth step allows to re-estimate the mean function under a more suitable covariance assumption. For the subsequent steps, the curves are then centered by subtracting the estimated mean from the functional observations.

For the second step, we propose to simultaneously estimate the upper triangles of the covariances $\mathbf{K}^{U_g}(t, t')$, $g = 1, \dots, G$, and $K^E(t, t')$ and the error variance using our novel covariance smoothing approach. The triangular covariance surfaces are then reflected across the diagonal, yielding estimates for the complete covariance surfaces $\mathbf{K}^{U_g}(t, t') = \left[K_{ss'}^{U_g}(t, t') \right]_{s, s'=1, \dots, \rho^{U_g}}$, $g = 1, \dots, G$, and $K^E(t, t')$. Negative estimated values of σ^2 are set to zero.

In the third step, we use spectral decompositions of the estimated covariance surfaces based on Mercer's theorem (Mercer, 1909) and an extension of Mercer's theorem to vector-valued random processes (Balakrishnan, 1960; Kelly and Root, 1960)

$$\mathbf{K}^{U_g}(t, t') = \sum_{k=1}^{\infty} \nu_k^{U_g} \phi_k^{U_g}(t) \phi_k^{U_g}(t')^\top, \quad \mathbf{K}^E(t, t') = \sum_{k=1}^{\infty} \nu_k^E \phi_k^E(t) \phi_k^E(t'),$$

with eigenvalues $\nu_k^{U_g}$, ν_k^E and (vector-valued) eigenfunctions $\phi_k^{U_g}(t) = [\phi_{ks}^{U_g}(t)]_{s=1, \dots, \rho^{U_g}}$, $\phi_k^E(t)$, respectively. In practice, the covariance surfaces are evaluated on a dense grid $\tilde{\mathcal{D}} = \{t_1, \dots, t_{\tilde{D}}\} \in \mathcal{T}$ of pre-specified length \tilde{D} . We obtain estimated eigenvalues $\hat{\nu}_k^{U_g}$, $k = 1, \dots, \tilde{D}\rho^{U_g}$, and $\hat{\nu}_k^E$, $k = 1, \dots, \tilde{D}$, as well as orthonormal eigenfunctions evaluated on $\tilde{\mathcal{D}}$, $\hat{\phi}_k^{U_g} = [\hat{\phi}_{ks}^{U_g}(t)]_{s=1, \dots, \rho^{U_g}, t \in \tilde{\mathcal{D}}} \in \mathbb{R}^{\tilde{D}\rho^{U_g}}$ and $\hat{\phi}_k^E = [\hat{\phi}_k^E(t)]_{t \in \tilde{\mathcal{D}}} \in \mathbb{R}^{\tilde{D}}$, of each corresponding covariance operator. The multivariate eigenvectors $\hat{\phi}_k^{U_g}$ consist of blocks for the respective eigenfunction components. The eigenvectors $\hat{\phi}_k^{U_g}$, $\hat{\phi}_k^E$ (and accordingly the eigenvalues) are rescaled to ensure orthonormality with respect to the additive inner product $\langle (f_1, \dots, f_{\rho^{U_g}}), (g_1, \dots, g_{\rho^{U_g}}) \rangle = \sum_{s=1}^{\rho^{U_g}} \int_{\mathcal{T}} f_s(t)g_s(t) dt$, and with respect to the L^2 -inner product $\langle f, g \rangle = \int_{\mathcal{T}} f(t)g(t) dt$, respectively. For more details on multivariate FPCA, see, e.g., Ramsay and Silverman (2005). To guarantee positive semi-definiteness of the covariances, which is not ensured by our smoothing approach, negative eigenvalues can be set to zero, which has been shown to improve accuracy in terms of the L^2 -norm (Hall et al., 2008) and to work well in practice (e.g., Yao et al., 2003). Dimension reduction is achieved by truncating the number of eigenfunctions. We choose the truncation levels based on the proportion of variance explained (for an overview, see Greven et al., 2010) and denote them by N^{U_g} , $g = 1, \dots, G$, and N^E , respectively. The truncated Karhunen-Loève (KL) expansion (Loève, 1946; Karhunen, 1947) and an extension to vector-valued random processes (Balakrishnan, 1960; Kelly and Root, 1960) allows parsimonious representations of the random processes in truncated bases of the corresponding eigenfunctions

$$\mathbf{U}_{gl}(t) \approx \sum_{k=1}^{N^{U_g}} \xi_{lk}^{U_g} \phi_k^{U_g}(t), \quad E_i(t) \approx \sum_{k=1}^{N^E} \xi_{ik}^E \phi_k^E(t), \quad (5.15)$$

with uncorrelated zero mean random basis weights $\xi_{lk}^{U_g}$, $l = 1, \dots, L^{U_g}$, $k = 1, \dots, N^{U_g}$, and ξ_{ik}^E , $i = 1, \dots, n$, $k = 1, \dots, N^E$, with variance $\nu_k^{U_g}$ and ν_k^E , respectively.

In the fourth step, we predict the random basis weights, which give insight into the individual structure of each grouping level. Replacing the random processes in Model (5.7) by their truncated KL expansions in (5.15) allows to approximate the model by a scalar linear mixed model with random effects corresponding to the random basis weights $\xi_{lk}^{U_g}$, ξ_{ik}^E (Di et al., 2009). The basis weights can then be predicted as empirical best linear unbiased predictors by simply plugging in the estimated eigenfunctions, eigenvalues, and the estimated error variance (Di et al., 2009; Greven et al., 2010, and Chapter 4 of this thesis).

Alternatively, we can represent our model as a FAMM using our estimated eigenfunctions and -values in basis expansions of the random processes as proposed by Scheipl et al. (2015). Under the assumption of Gaussian responses, the random basis weights can be predicted together with a re-estimation of the mean function in a mixed model framework. This allows for more efficient mean estimation due to taking the covariance structure into account and for approximate statistical inference conditional on the FPCA, such as point-wise confidence bands for the mean and for covariate

effects. For more details on the combination with the FAMM approach and an extensive comparison of the two ways to predict the basis weights, see Chapter 4.

5.5 Implementation

We provide freely available software implemented in the R add-on package `sparseFLMM` (Cederbaum, 2016). Our implementation is based on the R add-on package `mgcv` (Wood, 2006, 2011), which allows to add user-defined spline bases and penalties. In this framework, we define a novel class for bivariate smooths estimated subject to our symmetry constraint (5.5) called 'symm' by providing a new constructor method function `smooth.construct.symm.smooth.spec` and a corresponding predictor method function for the estimation of smooth surfaces in additive models. The class can be applied to any bivariate smooth term in a `gam`-formula. It is not restricted to symmetric data although we here consider symmetric data in the form of products of the centered functional responses in the smoothing of the covariance. It can be applied to (possibly noisy) data sampled on a regular grid as well as to irregularly or sparsely sampled data. As a modular component, our constructor can be applied separately to the auto-covariances of independent functional random effects in an FLMM. The case of correlated functional random effects, such as correlated functional random intercepts and slopes, is currently not covered in the implementation. In our application, we show how the constructor can be applied to complex designs on the basis of the FLMM with crossed fRIs as in Equation (5.13). One main advantage of using standard software is that it allows for flexible extensions. The current implementation in the R add-on package `sparseFLMM` is based on tensor product B-splines with difference penalties (Eilers and Marx, 2003), but extensions to other bivariate bases that are symmetric across the diagonal and other penalties are possible.

The spline degree and the number of basis functions can be chosen. The user currently has the choice between two different quadratic penalties. One can either use the Kronecker sum penalty $\text{pen}(\lambda) = \lambda \boldsymbol{\theta}^\top [(\mathbf{S}_t \otimes \mathbf{I}_F) + (\mathbf{I}_F \otimes \mathbf{S}_{t'})] \boldsymbol{\theta}$ or alternatively a Kronecker product penalty of the form $\text{pen}(\lambda) = \lambda \boldsymbol{\theta}^\top (\mathbf{S}_t \otimes \mathbf{S}_{t'}) \boldsymbol{\theta} = \lambda \boldsymbol{\theta}^\top [(\mathbf{S}_t \otimes \mathbf{I}_F) \cdot (\mathbf{I}_F \otimes \mathbf{S}_{t'})] \boldsymbol{\theta}$, where $\mathbf{S}_t = \mathbf{S}_{t'}$ denote the marginal penalty matrices and $\boldsymbol{\theta}$ is the coefficient vector. Note that both penalties are isotropic, reflecting the symmetry of the surface. The main difference between the two penalty matrices is that the null space of the latter is larger, more likely leading to wigglier estimates. Other possible penalties could be added by the user.

To speed up estimation, function `bam` can compute the computationally expensive steps in parallel on multiple cores.

In addition to the implementation of our constructor, the R add-on package `sparseFLMM` includes an implementation for the FPCA based on our (additive) covariance smoothing approach for three special cases of the FLMM (Model (5.1), a model with an fRI and a smooth error curve, and the model with two crossed fRIs and a smooth error curve as in Section 5.3.3).

5.6 Applications

We demonstrate the practical relevance of our approach in two distinct applications. We consider a standard data set consisting of sparse longitudinal observations as well as functional data with a complex correlation structure and different grids between curves.

5.6.1 CD4 cell count data

In order to compare our approach to that of Xiao et al. (2017) for the common special case of longitudinal data, we apply FPCA to analyze the CD4 cell count trajectories in HIV positive individuals which are available in the R add-on package `refund` (Huang et al., 2016a). As our focus here is on the more complex case of correlated functional data, to which the approach of Xiao et al. (2017) does not apply, the application to the CD4 cell count data is given in Section D.3 in Appendix D.

5.6.2 Speech production research data

We apply FPCA based on our covariance smoother to acoustic signal data with a crossed correlation structure. We show the increase of the computational efficiency compared to the approach proposed in Chapter 4, for a case where that approach is applicable. To the best of our knowledge, the smooth method of moments approach presented in Chapter 4 is the only competitor for covariance smoothing of irregularly observed correlated curves with a crossed design structure. No alternative approach is available for general models (5.7), where our approach is the first available for additive covariance smoothing.

In phonetic research, the term consonant assimilation refers to the phenomenon that the articulation of two consonants becomes phonetically more alike when they appear subsequently in fluent speech. Consonant assimilation is accompanied by a complex interaction of language-specific, perceptual and articulatory factors which makes it an important topic in speech production research. The data we consider are part of a large study which was conducted by Pouplier and Hoole (2016) in order to investigate among others the assimilation of the consonants /s/ and /sh/ as a function of their order (/s#sh/ versus /sh#s/, where # denotes a word boundary), syllable stress and vowel context in the German language. The same data were previously described and analyzed in Chapter 4. In order to make each chapter self-contained, a short data description is provided in the following. Pouplier and Hoole (2016) recorded the audio signals for nine native speakers which repeated the same sixteen target words each five times. The target words consisted of (semantically nonsensical) bisyllabic noun-noun compound words with abutting consonants /s/ and /sh/ in either order, e.g., ‘Callas-Schimmel’ and ‘Gulasch-Simpel’, and with either stressed or unstressed syllables and varying vowel combinations. The time interval during the duration of the two consonants of interest was cut out manually by the phoneticians and standardized to a [0,1] interval in which the recorded acoustic signals were summarized in a functional index over time. The $n = 707$ index curves (shown in Figure 5.1) take values between +1 and -1, with positive [negative] values indicating proximity of the signal to a reference signal for the first [second] consonant of the target word, respectively. To illustrate the

effect of consonant assimilation, two acoustic signals are highlighted in Figure 5.1. The curve without or with little assimilation shows a clear transition from a strong positive to a strong negative value, whereas the curve with strong assimilation is quite flat and mostly takes negative values. The curves differ in the number and location of the observation points, ranging from 22–57 with a median of 34 points per curve. For a more detailed description of the data (including pre-processing steps), see Pouplier and Hoole (2016) and Chapter 4.

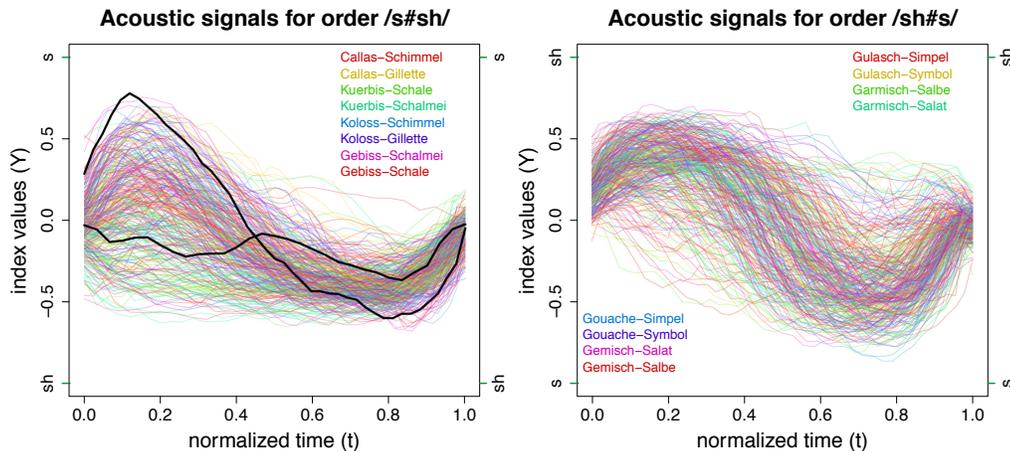


Figure 5.1: Acoustic signal curves of the speech production data over time (cf. Chapter 4). Left [right]: Signal curves of consonant order $/s\#sh/$ $[sh\#s]$ colored by target word. Example curves with no/little assimilation and with strong assimilation highlighted (black).

We fit an FLMM with crossed fRIs as described in Section 5.3.3 and previously considered in Chapter 4 to account for the repeated measurements of both speakers and target words. The mean function $\mu(t, \mathbf{x}_i)$ includes effects and interaction effects of the dummy-coded covariates consonant order, syllable stress, and vowel context, smoothly varying over time. As the focus here is on the smooth auto-covariances of the functional random effects, $K^B(t, t')$, $K^C(t, t')$, and $K^E(t, t')$, it is referred to Chapter 4 for more details on and interpretations of the covariate effects. For each auto-covariance, we use cubic marginal B-spline bases with marginal third order difference penalty matrices and use the Kronecker sum penalty for bivariate smoothing (cp. Section 5.5). We compare the results from our novel symmetric bivariate smoother (denoted by TRI-CONSTR and by TRI-CONSTR-W with weights of 0.5 for the products on the diagonal, cp. Section 5.2.2) with the results we obtain using the smoothing approach proposed in Chapter 4 (denoted by WHOLE). The latter does not exploit the symmetry of the estimated surface and is equivalent to TRI-CONSTR-W except for the estimation of the smoothing parameter and numerical differences. To highlight the need for a symmetry constraint when only the upper triangle is considered, we further compare with the results obtained by estimating the upper triangle without a symmetry constraint (denoted by TRI) which does not guarantee smoothness across the diagonal (cp. Section 5.2.2). The estimated auto-covariances are evaluated on a pre-specified grid of length $\tilde{D} = 100$. During the FPCA, we

choose the truncation levels based on a pre-specified proportion of explained variance of $L = 0.95$, yielding two [four] eigenfunctions for the auto-covariance of the fRI for speakers, $K^B(t, t')$, and three [twelve] eigenfunctions for the auto-covariance of the smooth error, $K^E(t, t')$, for our two approaches and WHOLE [for TRI]. For all four smoothing methods, no eigenfunction is chosen for the fRI for target words, which is due to the high number of covariates that describe the target words sufficiently (cf. Chapter 4. Figure 5.2 shows the estimated surfaces and contours of the auto-covariance of the fRI for speakers, reconstructed after truncation from the estimated eigenvalues and eigenfunctions, which are shown in the bottom of the figure (black). As a sensitivity analysis, we additionally depict jackknife estimates of the eigenfunctions with each of the nine speakers left out once (gray lines, bottom row). The variability over jackknife samples is similar for all compared approaches. However, while the same number of FPCs is chosen in all jackknife samples for TRI-CONSTR, TRI-CONSTR-W, and WHOLE, jackknife results for TRI are more sensitive. In two out of nine jackknife samples, TRI selects more FPCs than for the whole data set.

We can see from Figure 5.2 that the estimated covariance $\widehat{K}^B(t, t')$ based on our symmetric smoother (TRI-CONSTR-W) is very similar to the one obtained by using all products (WHOLE). Moreover, it shows that for this application, the weights for the diagonal products do not make a great difference. As expected, we observe wigglier estimates especially on the diagonal for TRI, for which the error variance is estimated to be zero. This is also reflected in the wiggleness and higher number of chosen eigenfunctions for TRI. Similar results can be found for $K^E(t, t')$. These are given in Section D.3 in Appendix D, where additional estimation details and results including the complete variance decompositions are provided. For all four approaches, the first eigenfunction in Figure 5.2 (solid line) corresponds to the discrimination of the speaker between the first and the second consonant and the second eigenfunction (dashed line) mainly leads to a vertical shift of the signal curves. Accounting for the symmetry of the covariances leads to a considerable reduction of computation times. TRI-CONSTR and TRI-CONSTR-W have the shortest computation times for smoothing the three auto-covariances, using five kernels in parallel, amounting to 24.51 and 25.82 minutes, respectively. Smoothing the auto-covariances using WHOLE takes more than twice as long (55.76 minutes) and using TRI still amounts to 32.95 minutes, which partly results from the fact that more spline coefficients have to be estimated. In addition, WHOLE and TRI require the estimation of two (instead of one) smoothing parameters for each auto-covariance using the Kronecker sum penalty implemented in the R add-on package `mgcv` of the form $\text{pen}(\lambda) = \lambda_t \boldsymbol{\theta}^\top (\mathbf{S}_t \otimes \mathbf{I}_F) \boldsymbol{\theta} + \lambda_{t'} \boldsymbol{\theta}^\top (\mathbf{I}_F \otimes \mathbf{S}_{t'}) \boldsymbol{\theta}$.

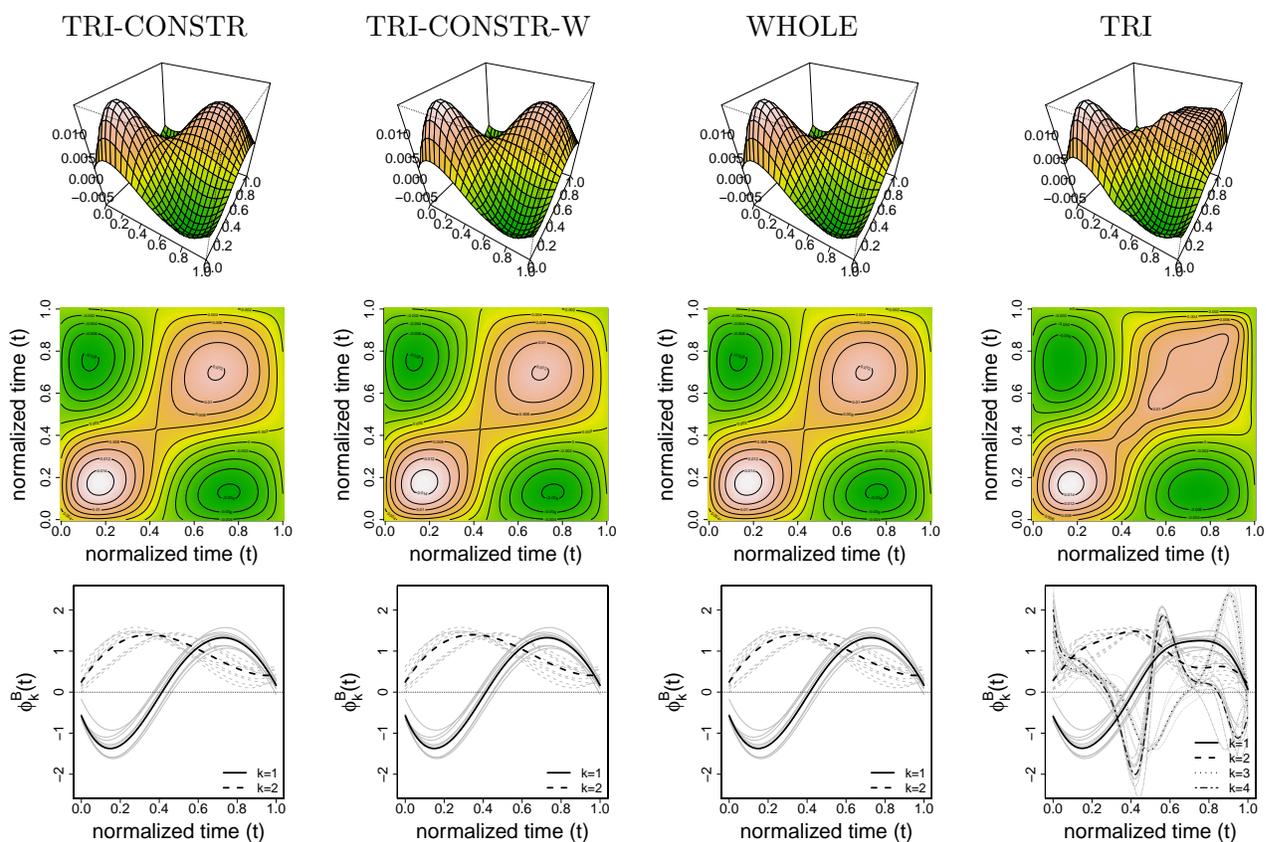


Figure 5.2: Results for the fRI for speakers using the four smoothing methods. Top row: Estimated covariance surfaces. Middle row: Contours of the estimated covariance surfaces. Bottom row: Estimated eigenfunctions $\hat{\phi}_k^B(t)$ based on the entire data set (black) and jackknife estimates with one of the speakers left out in turn (gray).

5.7 Simulations

5.7.1 Simulation designs

To investigate the performance of our covariance smoothing approach, we conduct an extensive simulation study based on two different data generating processes. For the first scenario (Scenario 1), we simulate data consisting of $n = 100$ independent curves from Model (5.1) with $\mu(t) = \sin(t) + t$. The data for the second scenario (Scenario 2) are generated from an FLMM with crossed fRIs as in Section 5.3.3, such that they mimic the irregularly observed speech production data. Note that no covariate mean effects are included such that we additionally obtain one eigenfunction for the fRI for target words and thus really have crossed fRIs. Additional generation details for this scenario are given in Section D.4 in Appendix D. We compare the performance of our fast (additive) symmetric covariance smoother (TRI-CONSTR, TRI-CONSTR-W with weights of 0.5 for the diagonal products) as basis for FPCA with the performance of the smoothing approach proposed in Chapter 4 (WHOLE). To evaluate the need of the symmetry constraint when only the upper triangle is estimated, we further compare with the results obtained without posing a symmetry constraint (TRI). For Scenario 1, we additionally provide a comparison with the smoothing approach of Xiao et al. (2017) (denoted by FACE), implemented in function `face.sparse` in the R add-on package `face` (Xiao et al., 2016a). FACE does, however, not apply to additive covariance smoothing needed for correlated curves. Note that Xiao et al. (2017) compare their symmetric covariance smoother to a number of other approaches that are all restricted to independent curves. They show that their approach is superior or comparable in terms of median integrated squared errors and inter quartile ranges (IQR) to the approach implemented in function `fpc.sc` (Di et al., 2009) in the R add-on package `refund` based on bivariate B-splines with a difference penalty and to a self-coded variant based on thin plate regression splines for covariance smoothing. Moreover, they demonstrate their supremacy over the geometric likelihood approach of Peng and Paul (2009) and the local polynomial approach proposed in Yao et al. (2003). We thus do not include these alternatives here.

Based on Scenario 1, we investigate the sensitivity of the estimates to varying model complexity in terms of the complexity of the underlying eigenfunctions, signal to noise levels as functions of error variances and eigenvalues, and degree of sparseness. We consider all possible combinations of

1. simple eigenfunctions: $\{\phi_1(t) = 1, \phi_2(t) = \sqrt{3}(2t - 1)\}$,
complex eigenfunctions $\{\phi_1(t) = \sin(2\pi t), \phi_2(t) = \cos(2\pi t)\}$
2. error variance: $\sigma^2 = 0.5, \sigma^2 = 0.05$
3. eigenvalues: $\{\nu_1 = 0.15, \nu_2 = 0.075\}, \{\nu_1 = 2, \nu_2 = \}$
4. number of observation points: drawn from uniform distributions $\mathcal{U}[40, 60]$.

For the simple eigenfunctions, we additionally consider a sparse setting, in which the number of observation points is drawn from the uniform distribution $\mathcal{U}[3, 10]$. For the complex eigenfunctions and eigenvalues $\nu_1 = 2, \nu_2 = 1$, we additionally reduce the value of the error variance to $\sigma^2 = 0.01$. For all settings, we generate 200 data sets. The random basis weights are centered and decorrelated such that the weights empirically have zero mean and a correlation of zero (see Chapter 4 for a discussion).

For the estimation in Scenario 1, we use ten cubic B-splines each for the estimation of the mean function and as marginal bases for the auto-covariances. For the estimation in Scenario 2, we use eight and five cubic B-splines for the estimation of the mean function and as marginal bases for the auto-covariances, respectively. We use Kronecker sum penalties (cp. Section 5.5) of marginal third order difference penalties for bivariate smoothing. Estimation of the smoothing parameter is based on REML, except for FACE, which uses leave-one-curve-out cross-validation. We use equidistant knots in function `face.sparse` instead of the default (quantile-based knots) which would require an adapted penalty that is not implemented. The arguments that determine the smoothing parameter search in function `face.sparse` are left at their defaults. As function `face.sparse` does not allow to specify a fixed truncation level, we choose the number of eigenfunctions based on a pre-specified proportion of explained variance of $L = 0.95$. Note that we use the proportion of explained variance in the functional observations, whereas Xiao et al. (2016a) use that in $E_i(t)$. In order to be able to differentiate the error incurred by the truncation of the covariance surface to a few leading FPCs from the pure covariance surface estimation error, we pre-specify the correct truncation lags for Scenario 2, for which no comparison with FACE is possible anyway.

5.7.2 Simulation results

We present and discuss the results of both simulation scenarios. For Scenario 1, we focus our presentation of the results on the setting with complex eigenfunctions, an error variance of $\sigma^2 = 0.05$ and eigenvalues of size $\nu_1 = 2$, $\nu_2 = 1$ (denoted as Setting 1). As a measure of goodness of fit, we use root relative mean squared errors (rrMSEs) of the form $\sqrt{(\text{true} - \text{estimated})^2 / \text{true}^2}$. The complete results for all settings and the specific forms of the rrMSE for all model components, are given in Section D.4 in Appendix D.

Figure 5.3 depicts boxplots of the rrMSEs for 200 simulation runs for Setting 1. For each model component, it shows the boxplots for the compared smoothing methods and, in addition, for a modified version of FACE (denoted by FACE-STEP-1), in which the covariance of the products is not accounted for and thus only the first step of the three-step procedure is performed. FACE-STEP-1 is added to evaluate the effect of accounting for the covariance of the products (cf. Section 5.14). It shows that all components, except for the error variance, are estimated very well for our approach (TRI-CONSTR, TRI-CONSTR-W). The weights for the diagonal products do not have a great influence. Moreover, our approach yields similar rrMSEs as WHOLE. We also obtain similar results for TRI for most components, except for the error variance for which TRI has a higher median rrMSE. FACE yields a more than 2.7 times higher median rrMSE for the auto-covariance $K^E(t, t')$ compared to our approach and consequently also worse results for the eigenvalues ν_k^E , $k = 1, 2$. It yields smaller median rrMSEs for the eigenfunctions ϕ_k^E , $k = 1, 2$, but the IQR is larger and more outliers occur than in all other approaches. The estimation of the error variance profits most from accounting for the covariance of the products, which is reflected in a much lower median rrMSE for FACE than for the other methods. This also leads to lower median rrMSEs for the random basis weights, ξ_1^E and ξ_2^E , which depend on the error variance and consequently also to lower median rrMSEs for the reconstructed processes

$E_i(t)$ and $Y_i(t)$. For all components, however, FACE yields a high number of outliers that range above the maximal rrMSEs obtained for the other methods. It is noticeable that the estimation of the auto-covariance is considerably better (in terms of rrMSE) for FACE when the covariance of the products is not accounted for. Moreover, we see that FACE-STEP-1 yields roughly similar results to our approach and WHOLE. For TRI-CONSTR, TRI-CONSTR-W and WHOLE, the truncation level is correctly estimated to be two in all 200 simulation runs. A higher number of eigenfunctions (three to four) is chosen for TRI in eight simulation runs of this setting, which is consistent with our results in the application to the speech production data. FACE and FACE-STEP-1 choose more than two (three to five) eigenfunctions in 190 and 198 simulation runs, respectively.

To sum up the results for the other ten settings of Scenario 1, we can say that over all settings and components TRI-CONSTR, TRI-CONSTR-W and WHOLE yield similar rrMSEs with a tendency to a supremacy of TRI-CONSTR, especially in the sparse settings. TRI yields similar to worse results compared to our method and WHOLE. Especially for the error variance it yields up to 82% higher median rrMSEs (in one of the sparse settings). The estimation quality of all methods differs between the dense and the sparse settings. Our approach yields relatively similar results within the dense settings and higher rrMSEs in the sparse settings. Moreover, TRI-CONSTR and TRI-CONSTR-W tend to perform better in the settings with complex eigenfunctions and favor smaller error variances (except for the estimation quality of the error variance itself). In contrast, FACE tends to perform better in the settings with simple eigenfunctions and favors larger error variances. Our approach and WHOLE always select the correct truncation level, except in the sparse settings, where for some simulation runs more eigenfunctions are selected. TRI tends to select more eigenfunctions. For all settings, FACE and FACE-STEP-1 have simulation runs in which more than two eigenfunctions are selected.

In contrast to FACE and FACE-STEP-1, our approach is directly extendable to smoothing multiple auto-covariances simultaneously, which is required in Scenario 2. Figure 5.4 depicts the rrMSEs for the auto-covariances, the error variance and curves $Y_i(t)$ for Scenario 2. It shows that the three auto-covariances are estimated equally well for the four compared smoothing methods. For the error variance, however, TRI performs worse with a 32.5% higher median rrMSE and a larger IQR, which also results in slightly higher rrMSEs for the reconstructed curves. Over all, our methods perform very well considering the small number of levels for the fRIs B (9 levels) and C (16 levels). For all model components, TRI-CONSTR and TRI-CONSTR-W yield similar rrMSEs.

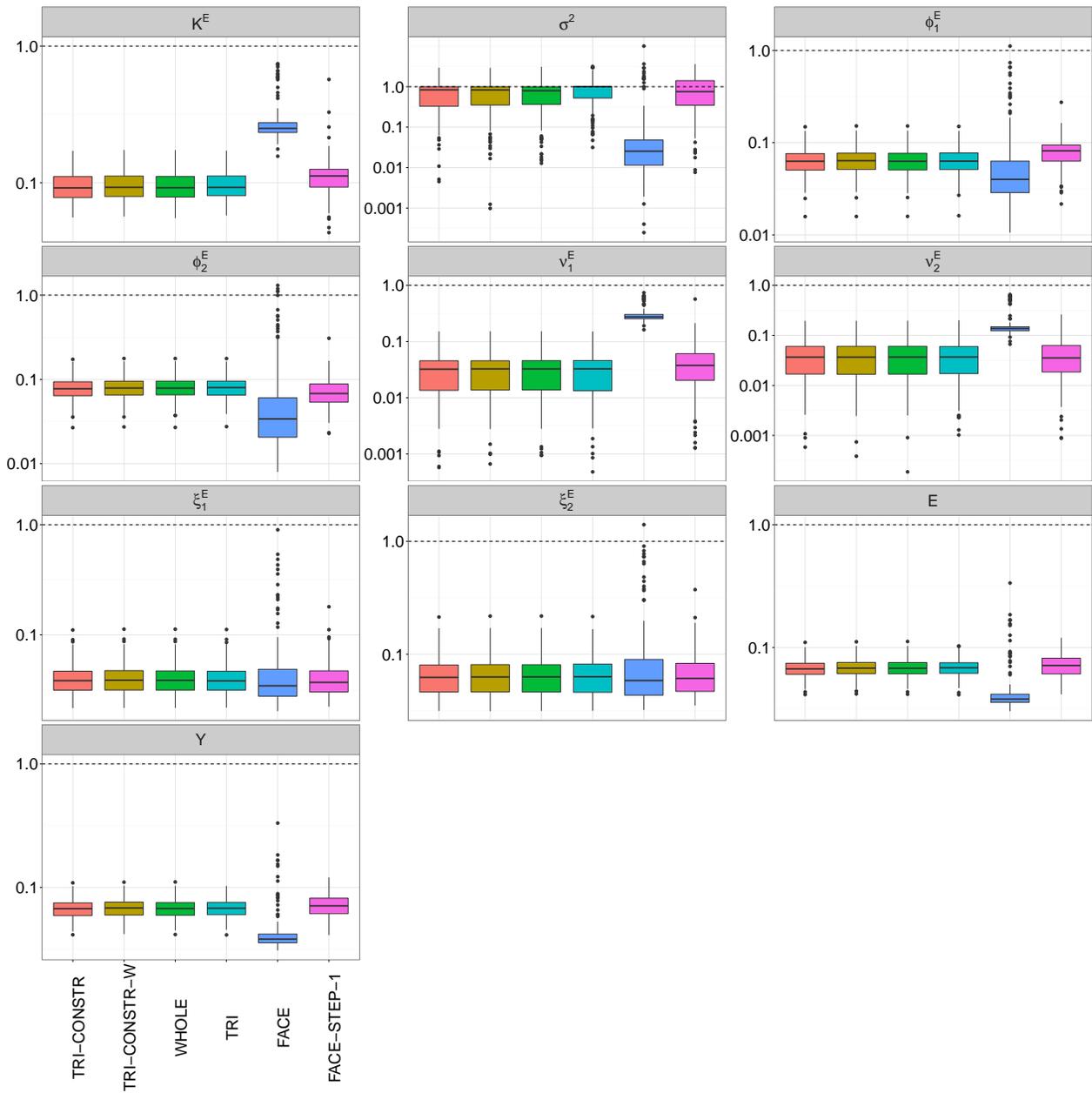


Figure 5.3: Boxplots of the rrMSEs (log10 scale at y-axis) for the scenario with independent curves (Scenario 1) for the smoothing methods being compared. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

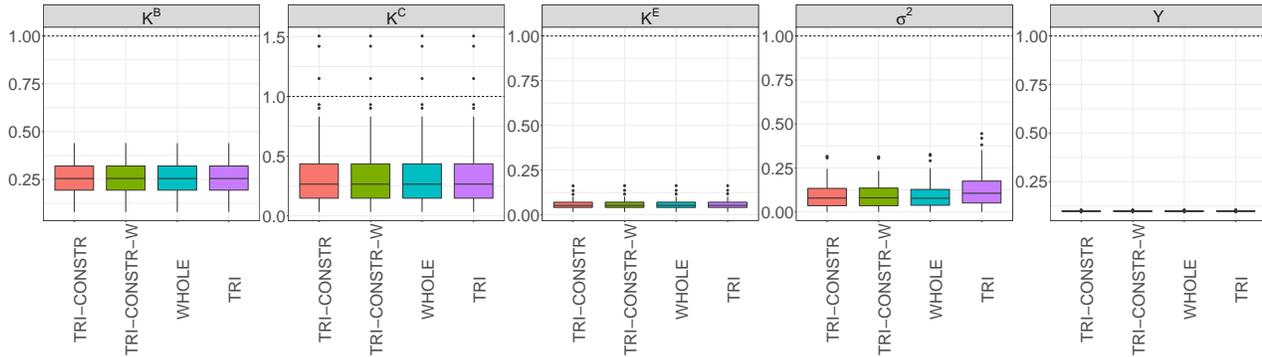


Figure 5.4: Boxplots of the rmMSEs for the scenario with crossed fRIs (Scenario 2) for the smoothing methods being compared. Shown are boxplots of the rmMSEs for the three auto-covariances $K^B(t, t')$, $K^C(t, t')$, and $K^E(t, t')$ as well as for the error variance σ^2 and the curves $Y_i(t)$.

5.7.3 Computational efficiency

Figure 5.5 shows computation times on a 64 Bit Linux platform with 660 Gb of RAM memory for the two settings discussed above. Our approach (with and without weights for the diagonal products) greatly speeds up the computation compared to WHOLE and also to TRI, especially in the scenario with crossed fRIs (right figure), where the computation times are longer and thus matter more. Note that in addition to accounting for the symmetry of covariances, we reduce the number of smoothing parameters to be estimated compared to WHOLE and TRI, for which two smoothing parameters are estimated for each auto-covariance using the Kronecker sum penalty implemented in the R add-on package `mgcv`. In the scenario with independent curves (left figure), FACE is considerably slower than the other methods. This applies to all except the sparse settings, in which the computation times are extremely short anyway (max. 15 sec.). Note that variation for FACE and FACE-STEP-1 is considerably smaller as the smoothing parameter is chosen based on a fixed grid.

5.7.4 Summary

Overall, one can conclude from our simulations that the proposed symmetric smoothing approach performs well both in the simple scenario as well as in the scenario with crossed fRIs. In the regarded 12 simulation settings, the novel symmetric smoother yields similar results with and without weights for the products on the diagonal. It shows that the results are also similar to those of the approach proposed in Chapter 4 that, however, requires considerably longer computation times. As expected, the proposed symmetric smoothing approach performs better than the estimation of the upper triangle without a symmetry constraint, most notably for the estimation of the error variance. For the scenario with independent curves, in which a comparison with the approach of Xiao et al. (2017) is possible, neither is clearly preferable for all estimated components in all settings. In addition to the computational advantages of the proposed symmetric smoother, it also has much fewer outliers

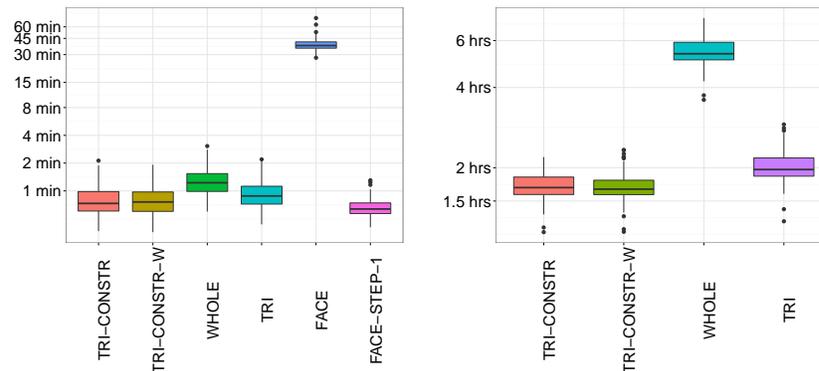


Figure 5.5: Computation times (log10 scale at y-axis) for 200 simulation runs for all compared methods. Left: Computation times for Setting 1 of the scenario with independent curves. Right: Computation times for the scenario with crossed fRIs.

than *FACE* and more reliably selects the correct number of FPCs. Most importantly, the proposed approach—in contrast to *FACE*—allows to smooth multiple auto-covariances simultaneously.

5.8 Discussion and outlook

We have introduced a fast bivariate smoothing approach for symmetric surfaces which applies to a broad range of data situations. We focus on its application to estimate covariance functions in longitudinal data as well as multiple additive covariance functions in functional data with very general correlation structures. Our smoother can handle (possibly noisy) data sampled on a common, dense grid as well as irregularly or sparsely observed data, which are frequently encountered in practice. It extends the smooth method of moments estimator proposed in Chapter 4 of this thesis to more general correlation structures and additionally takes advantage of the symmetry of the sample covariances, which leads to considerably faster estimation requiring less memory. The covariance functions are estimated as unknown, smooth functions in a bivariate additive varying coefficient model for the empirical covariances. A symmetry constraint additionally ensures smoothness of the estimated covariance surfaces across the diagonal and further reduces computational costs. We show how our smoother can be applied as basis for FPCA, a key tool for dimension reduction in FDA, and demonstrate its practical relevance in a longitudinal data application and in an application to complexly correlated functional speech production data. We provide fully documented open-source software implementing our approach in the R add-on package `sparseFLMM`, which builds on the established R add-on package `mgcv` allowing for flexible extensions (Cederbaum, 2016; Wood, 2006, 2011). Within this framework, we provide a novel constructor function for the estimation of smooth surfaces in additive models subject to the symmetry constraint. Our constructor can be applied as a modular component to general bivariate symmetric smoothing problems. Simulation experiments (in Section

5.7) show that the proposed method recovers the true functions very well and yields similar results as the estimation approach proposed in Chapter 4 while considerably speeding up the estimation and extending the range of possible model structures.

This work opens up a number of interesting directions for future research. A first direction concerns the working assumptions of independent and homoscedastic products of the centered functional responses that are implicitly made by using a quadratic loss function in the estimation of the covariances in a bivariate additive varying coefficient model (cp. Section 5.3.4). It would be interesting to investigate whether a suitable loss function for the products could be derived. Under the assumption of Gaussian responses, the products follow a product normal distribution, for whose probability density function Nadarajah and Pogány (2016) recently derived a closed-form expression based on the modified Bessel function. A second direction concerns the positive semi-definiteness of the covariance operator, which is not ensured in our approach and in most existing covariance smoothing approaches (e.g., Yao et al., 2003; Hall et al., 2008; Di et al., 2009; Greven et al., 2010). Although Hall et al. (2008) show that setting negative eigenvalues to zero improves the estimation quality and Yao et al. (2003) demonstrate that this works well in practice, it could be desirable to ensure positive semi-definiteness in the estimation. Wu and Pourahmadi (2003) estimate a positive semi-definite covariance based on an auto-regressive model with regression coefficients corresponding to the components of a modified Cholesky decomposition of the covariance. The approach of Peng and Paul (2009) also ensures positive semi-definiteness. It remains an open question, however, how these approaches could be extended to smoothing multiple additive covariances for functional data with complex correlation structures. Another possible path for future investigation would be the extension of our models to allow for more complex noise terms. This extension requires careful, extensive analyses in order to ensure identifiability. For much simpler models, Descary and Panaretos (2016) develop identifiability conditions for the separation of smooth and rough variation (corresponding to $E_i(t_{ij})$ and $\varepsilon_i(t_{ij})$ in our notation). Whether and how these conditions can be extended to the very general functional linear mixed models considered in this chapter remain an open question.

Acknowledgements

The authors thank Marianne Pouplier and Phil Hoole for supplying and explaining the speech production data. All authors were supported by the Emmy Noether grant GR 3793/1-1 from the German Research Foundation. We thank the referees and the editor for their useful comments.

Chapter 6

Concluding Summary and Outlook

“ The more I learn, the more
I realize how much I don’t know.

ALBERT EINSTEIN

In this closing chapter, an overall concluding summary is given that highlights the main developments of this thesis and their importance for concrete applications (Section 6.1) based upon which relevant starting points for future research directions are discussed (Section 6.2). A more specific discussion of the proposed modeling approaches including their strengths and limitations can be found at the end of the respective chapters.

6.1 Concluding summary

The increasing collection of data that have a functional nature has motivated the development of numerous functional counterparts to methods that were initially designed for scalar or multivariate data. Regarding these data as functional observations allows to account for the inherent correlation and to extract additional information that often remains ignored when standard tools are applied. As regression analysis plays a central role in many statistical applications, the extension of various types of regression models to the functional framework has received particular attention. In many situations, one is interested in regressing a functional response on one or multiple scalar predictor variables. A common assumption in this context is that the observations of the functional response are *a)* independent and/or *b)* observed on a common, fine grid. Both assumptions are often not met in practice, where one frequently encounters data that have complex correlation structures and/or are available at—possibly few—curve-specific measurement points. In particular the co-occurrence of both aspects increases the need for flexible modeling approaches that are computationally feasible and yield interpretable results.

In this thesis, a computationally efficient estimation framework was developed that allows to analyze functional data with potentially complex correlation structures. The data may be observed on general grids, including curve-specific grids with very few grid points. Linear mixed models (LMMs) for scalar correlated data were extended to meet the requirements and to exploit the whole potential of the functional nature of the data. In its simplest form, the modeling framework can be utilized to enhance the analysis of longitudinal data, which frequently arise from, e.g., medical applications such as clinical trials, but the framework goes far beyond. The considered general class of functional linear mixed models (FLMMs) also applies to functional data with correlation induced by nested and/or crossed study designs. It further accommodates correlated functional random effects and allows the inclusion of effects of multiple scalar covariates.

The thesis is divided into three main parts that complement each other. In the first part (Chapter 3), a modeling approach was proposed that allows for very general correlation structures but is specifically designed for curves observed on a common grid. A closely related approach was presented in the second part (Chapter 4). This approach is particularly designed for irregularly or sparsely sampled data with the focus on a less general correlation structure. The third part (Chapter 5) introduced a fast symmetric bivariate smoothing approach that allows to considerably decrease the computational costs of the two modeling approaches, but also finds its application in general symmetric bivariate smoothing problems. Moreover, the bivariate smoother can be applied to extend the approach in the second part to more general correlation structures.

Estimation in both modeling approaches is based on functional principal component analysis (FPCA), yielding the dimension reduction crucially important when dealing with functional data. The functional random effects in the models are approximated by truncated bases of eigenfunctions of their respective covariance operators, which have to be estimated beforehand. Despite the additional effort to obtain appropriate covariance estimates, expansions in functional principal component (FPC) bases have several advantages over alternative basis approaches. Not only are the basis functions estimated from the data, but they also yield a parsimonious basis as they can be shown to optimally approximate the underlying process for a given number of basis functions. This was confirmed in a comparison with a spline-based alternative in Chapter 4, which showed a clear advantage of the use of FPC bases, both in terms of computation times and estimation quality. Moreover, FPC bases allow to explicitly decompose the variance in the data, which provides additional insight in the underlying structure. Furthermore, the FPCs often allow for interesting interpretations and the individual basis weights can be used for further analyses.

A main contribution of this thesis is the development of such smooth covariance estimates of the functional random effects on which the FPCA is based. The challenge lies in the fact that the functional random effects are unobserved processes that underlie the correlated functional observations. Their estimation is further complicated when the data are observed on curve-specific grids. Whereas for data observed on a common grid the covariances can be estimated using

a computationally efficient point-wise estimation approach and subsequent smoothing (Chapter 3), this approach fails when the measurement points are not the same among curves. In particular, when only few measurements per curve are available, direct smoothing becomes even more important. The sampling grid thus plays a crucial role for the estimation methods of the covariances. Once the smooth covariance estimates are available, the FPCA can be carried out in the same way irrespective of the sampling grid. Therefore, the estimation approaches proposed in Chapter 3 and Chapter 4 mainly differ in the estimation of the covariances. A remedy to the computational burden of smoothing (multiple) covariance surfaces is the fast symmetric bivariate smoother proposed in Chapter 5, which applies to complex correlated data as well as general grids.

Apart from the development of novel covariance estimation and smoothing approaches, existing FPC-based FLMMs were extended in this thesis to more general correlation structures, involving possibly correlated functional random intercepts and functional random slopes (Chapter 3 and Chapter 5). For correlated functional random effects, multivariate FPCA was applied in order to account for their co-variation.

Furthermore, this thesis proposed two approaches to predict the random basis weights. The first approach (Chapter 3, Chapter 4, and Chapter 5) predicts the basis weights as random effects in the resulting scalar LMMs and extends existing approaches of Yao et al. (2005), Di et al. (2009), and Greven et al. (2010) to allow for more general correlation structures. The second approach (Chapter 4) embeds the considered class of FLMMs in the framework of functional additive mixed models (FAMMs) introduced by Scheipl et al. (2015) and benefits from the resulting statistical inference conditional on the FPCA. This new combination of the FPC estimation with the FAMM framework additionally allows to include functional covariates in the—to this point—function-on-scalar regression model, taking care of identifiability issues arising in this context. A detailed explanation on how the FLMMs fall into the FAMM framework was presented in Chapter 4 but the FAMM framework can also be combined with the modeling approach proposed in Chapter 3. Note, however, that the implementation of the FAMM approach does not support correlated functional random effects.

The practical use of the presented modeling framework and its benefits were highlighted in three distinct applications, each requiring a different amount of complexity in the random effects structure and applicability to different sampling grids.

In Chapter 3, a model with hierarchical functional random effects with group-specific covariance structures was estimated, providing new insights into the structure of tissue spectroscopy data (Stelzle et al., 2011) that are sampled on an equal, dense grid. The combination with the classification approach of Zhu et al. (2012) yields a classifier that performs equally well to or even outperforms the considered competitors.

In Chapter 4 and Chapter 5, acoustic signals from a speech production study (Pouplier and Hoole, 2016) were analyzed. The pre-processed signals are available on dense, but curve-specific grids. The considered model includes crossed functional random intercepts that account for repeated recordings

for subjects and items, respectively. It is found that the main modes of variation in this application have meaningful interpretations. For the project partners from phonetics, the main interest lies in the effects of several dummy-coded covariates and their interactions. Therefore, a great benefit comes from the combination of the FPC estimation with the FAMM framework, which in particular yields point-wise confidence bands for the covariate effects (conditional on the FPCA). During the development of the estimation approach, five additional data sets stemming from the same experiment were analyzed. As the data are very similar and also lead to similar conclusions, the presentation of these additional results is not considered within this thesis. Moreover, in current joint research with Marianne Pouplier, Phil Hoole (Department of Phonetics and Speech Processing, LMU Munich), and Sonja Greven (Department of Statistics, LMU Munich), additional data sets are being analyzed. The collaboration with the phoneticians shows that data of this kind with similar correlation structures are frequently encountered in phonetics research which are commonly analyzed with tools that do not sufficiently account for the functional character of the data and thus lose valuable information.

In a second application in Chapter 5, the proposed framework was used to analyze sparsely sampled longitudinal CD4 cell count data as biomarkers of progression of HIV infection (Kaslow et al., 1987). This application demonstrates that although the framework can handle complex correlation between curves, it also applies to independent functional observations and in particular to longitudinal data that are only available at few measurement points. Again, the FPCs have interesting interpretations such as the speed with which the disease progresses.

As part of this thesis, open source software implementing parts of the proposed estimation framework is provided in the R add-on packages `sparseFLMM` and `denseFLMM` (R Core Team, 2016; Cederbaum, 2016; Greven and Cederbaum, 2017). The former is being used by the project partners from phonetics. The estimation of the smooth mean and covariance functions builds on well-established, flexible algorithms implemented in the R add-on package `mgcv` (Wood, 2006, 2011). For the combination with the FAMM framework, the R add-on package `refund` (Huang et al., 2016a) is used, which again is based on the `mgcv` package. For the fast symmetric bivariate smoother in Chapter 5, a new smoothing class in the `mgcv` framework is implemented that can be applied to any bivariate smoothing problem. Building the implementation of the proposed modeling framework on established software avoids ‘reinventing the wheel’, allows for flexible extensions, and benefits from future improvements. For example, the new fitting approach for huge data sets proposed by Wood et al. (2017), implemented in the package `mgcv`, can directly be applied in the estimation framework. This allows to handle very large data sets and further reduces the computation times obtained in this thesis.

6.2 Outlook

This thesis aimed at providing a flexible framework to model correlated functional data observed on general grids. Referring to Einstein’s quote at the beginning of this chapter, obviously, not all aspects can be addressed within this scope. Worthwhile avenues for future research include extensions of the

considered class of FLMMs and further improvements in the estimation, in particular the estimation of the covariance surfaces.

Model extensions

Several extensions of the considered model class may be relevant for the analyses of complex data sets. A first extension would be to overcome the limitations in the predictor and to look at general effects of both scalar and functional covariates. The inclusion of linear as well as smooth non-parametric effects of both covariate types can be achieved by embedding the model class in the FAMM framework of Scheipl et al. (2015) as described in Chapter 4.

Moreover, it would be interesting to consider spatial correlation to allow an adequate analysis in a wider range of applications, including, for instance, climate and neuroimaging data. In the proposed modeling framework, the FPC weights are assumed to be uncorrelated for different levels of the grouping factors. This assumption could be relaxed in order to allow for spatial correlation. Both approaches to prediction of the basis weights could be adapted, the prediction as random effects in a scalar LMM as well as the prediction by embedding the model in the FAMM framework. The latter extension seems to be straightforward as long as the spatial dependence structure is known. For the former, it might be possible to build on existing FPC-based approaches. For example, Liu et al. (2016) propose an extensions to the PACE approach of Yao et al. (2005) to spatially correlated data. Their SPACE (spatial PACE) approach assumes a Matérn correlation for the FPC weights. SPACE applies to irregular and sparse grids. It does, however, only account for correlation between functions that is induced by the spatial sampling and does not apply to functional data with multiple sources of correlation. A hierarchical FPC-based modeling approach with spatial correlation on the deepest level is proposed by Staicu et al. (2010). Their fast modeling approach is, however, only applicable to data with a hierarchical structure. For an overview on models for functional data with spatial correlation, see Morris (2015).

Another direction for possible extensions concerns the response distribution. As in many empirical problems, binary indicators or counts rather than continuous Gaussian data are observed, it would be interesting to look at functional analogues of generalized (linear) mixed models for scalar data (see, e.g., McCulloch et al., 2008; Molenberghs and Verbeke, 2005). Correlated non-Gaussian responses need to be treated carefully, mainly for two reasons. First, in analogy to scalar generalized linear mixed models, the interpretation of the fixed effects in the marginal and the conditional model perspective do in general no longer fall together when the data are non-Gaussian. Second, even for uncorrelated functional data, Gertheiss et al. (2017) argue that the direct extension of FPCA to generalized responses incorrectly uses a marginal mean and covariance estimate for an inherently conditional model (conditional with respect to the FPC weights), leading to biased estimates of both functional fixed and random effects. Gertheiss et al. (2017) propose two FPC estimation techniques to allow for an exclusively conditional perspective. The techniques are applicable to irregularly and sparsely sampled curves but are so far restricted to uncorrelated functional responses

and the extension to FLMMs would require further work. Once such an extension of FPCA to correlated non-Gaussian data is available, the FPC estimates could be combined with recent work of Scheipl et al. (2016a) who extend the FAMM approach to more general response distributions. This would allow for statistical inference based on scalar generalized additive mixed models. In general, interpretation of the fixed effects in these models needs to be done conditional on the functional random effects. It would thus also be interesting to consider a functional analogue of marginal models for scalar correlated data which could be applied in cases where population-specific interpretations of the fixed effects are of interest.

A further current limitation of the presented model class is that it solely focuses on modeling the conditional mean. In order to capture more complex relationships and to enable robust regression, it could be worth considering functional extensions of scalar regression approaches to modeling general features of the conditional response distribution. Recent extensions to, in particular, functional responses are proposed by Brockhaus et al. (2015), Brockhaus et al. (2017), and further extended by Greven and Scheipl (2017). Whereas the first requires data sampled on a common grid, the latter two allow for a more general grid structure. The underlying idea of their approaches is to model the different properties of the conditional response distribution by minimizing a suitable loss function. For the case of functional responses, which is of interest in this thesis, point-wise loss computation with subsequent integration is used that does not fully account for the functional nature of the response and could be further adapted (for a brief discussion, see Brockhaus, 2016). Moreover, it remains unclear if and how the general correlation structures considered in this thesis could be included in their framework and how their resampling-based statistical inference could be adjusted accordingly.

A number of further directions appear interesting, such as modeling derivatives of, in particular, sparsely sampled correlated functional data or the extension of the model class to functional data with higher-dimensional domains.

Estimation improvements

Apart from extensions of the model class, different potential improvements and extensions of the estimation approaches could be considered. One avenue for future research could be to further improve the estimation of the covariances. A first direction would be to work out how positive semi-definiteness can be ensured in the estimation. Although it has been shown that setting negative eigenvalues to zero improves the estimation quality (Hall et al., 2008) and works well in practice (e.g., Yao et al., 2003), it would be preferable to directly account for the specific properties of covariances. Several very different approaches to obtain positive semi-definite estimates have been proposed in the literature (e.g., Wu and Pourahmadi, 2003; Peng and Paul, 2009). A closer investigation of whether and how these approaches can be extended to more complex correlation structures (and the former also to more general grids) could be a first step.

Another step in the direction of enhancing the covariance estimation would be the development of a fast symmetric bivariate smoothing approach that replaces the heuristic quadratic loss which

implies working assumptions, such as working independence, that do not hold. A possible starting point for deriving a suitable loss function is the distribution of the products of the centered functional responses which are smoothed using a bivariate smoother. Under the assumption of Gaussian functional responses, these products follow a product-normal distribution. A closed-form expression of the corresponding probability density function has recently been derived by Nadarajah and Pogány (2016). A different approach also assuming Gaussian responses is followed by Kauermann and Wegener (2011) who jointly estimate the mean and covariance function using penalized splines. Their approach applies to longitudinal data and it would be interesting to explore the possibility to extend it to correlated functional responses.

Furthermore, a more detailed analysis of the two-step covariance estimation approach proposed in Chapter 3 to further inspect the effect of smoothing the previously estimated covariances would be desirable.

Another interesting point is the further consideration of uncertainty. As discussed in detail in Chapter 4, embedding the considered models in the FAMM framework allows for valid approximate statistical inference (conditional on the FPCA), including the construction of point-wise confidence bands for mean and covariate effects of interest. Although the simulation studies performed in the scope of this thesis show good coverage of the confidence bands, it would be desirable to take more sources of uncertainty into account.

Sources of uncertainty, which are currently not accounted for, include the estimation of the smoothing parameters. For a very general model class which includes (scalar) additive mixed models, Wood et al. (2016) propose an approach that accounts for the estimation uncertainty of the smoothing parameters. The authors provide an implementation in the R add-on package `mgcv`. This development could be integrated into the FAMM framework. From a theoretical perspective, this extension appears relatively straightforward as in the FAMM framework the functional estimation problem is reduced to a scalar estimation problem which enables the use of statistical methods for scalar data. In addition, the practical implementation of the FAMM estimation is already based on the package `mgcv` facilitating the technical realization.

One advantage of FPC basis expansions is that the basis functions are estimated from the data. However, this estimation also induces uncertainty which should be accounted for. When the random basis weights are predicted as random effects in scalar LMMs, the usually known design matrix for the random effects is unknown and has to be estimated. Not only the estimated eigenfunctions, but also estimates for the eigenvalues, the error variance, and the truncation levels enter the prediction of the basis weights both when they are predicted as random effects in a scalar LMM and when they are predicted in the FAMM framework. If the estimated mean (under independence assumption) is previously subtracted from the data in order to obtain estimates for the covariances, another source of uncertainty comes into play. Thus, the resulting inference has to be considered conditional on the FPCA. For the case of uncorrelated functional responses, Goldsmith et al. (2013) propose a non-parametric bootstrap-based approach that accounts for the uncertainty in FPC decompositions. The approach applies to data sampled on general grids, including sparsely sampled data, for which

the added variability can be of particular importance (Goldsmith et al., 2013). The authors derive both corrected point-wise and simultaneous confidence bands (or rather prediction bands) for the predicted curves. It would be interesting to extend their approach to correlated functional data with potentially complex correlation. This is, however, not straightforward and requires further investigation for mainly two reasons. First, it is not clear how to extend the resampling method to complexly correlated functions, with, e.g., crossed functional random effects. Second, for more complex data, the computational costs may rapidly increase and require further careful consideration. An alternative way to account for the estimation uncertainty is to pursue a fully Bayesian approach (e.g., Goldsmith et al., 2015).

Furthermore, it would be preferable to obtain simultaneous rather than point-wise confidence bands while keeping the computational costs low. Wiesenfarth et al. (2012) derive simultaneous inference in additive models with penalized splines using a mixed model representation that does not require resampling methods. Extensions of their approach to include random effects might allow the construction of simultaneous confidence bands in the FAMM framework. A combination with approaches that account for the different sources of uncertainty would be particularly attractive.

Further potential starting points to complete the modeling framework of this thesis are given by relaxing the underlying assumptions (compare Chapter 1, Section 1.4). First, it is so far assumed that the meaning at some point $t \in \mathcal{T}$ is the same for all curves, which may not always be true. It would therefore be interesting to extend the modeling framework to jointly consider amplitude and phase variation (e.g., Hadjipantelis et al., 2015).

Second, throughout this thesis, it is assumed that either the number and location of the sampling grids or the missing values in case of equal grids are non-informative. This is very restrictive and might not always hold in practice. It would thus be interesting to combine the modeling framework with work on missing data (e.g., Little and Rubin, 2014)

Third, a rather implicit assumption made in this thesis is that one does not have to deal with outliers in the observed data. As FPCA is not robust to outliers because it is based on covariances, the detection and treatment of outliers might be important in order to enhance the applicability in a wider range of empirical problems (for an overview, see Wang et al., 2016).

Fourth, to relax the assumption of a constant degree smoothness over the whole domain, extensions using adaptive smoothing could be of interest (e.g., Krivobokova et al., 2012), in cases where the data provide enough information. Adaptive smoothers are available in the R add-on package `mgcv` (Wood, 2006, 2011) which underlies the implementation of the modeling framework of this thesis. As the terms are, however, not suitable as components of tensor product smooths and are computationally expensive, additional work would be required to make adaptive smooths directly applicable (Wood, 2011).

Many other possible directions for further research exist. These include, for example, work on hypothesis testing as well as variable and model selection for FLMMs, which have so far not received much attention in the literature (see Liu and Guo, 2012). In particular, the selection of

functional random effects is an important issue and would allow to ensure parsimony in the very flexible class of FLMMs considered in this thesis. As the problem boils down to the selection of random effects in scalar additive mixed models, the corrected conditional Akaike Information Criterion proposed by Greven and Kneib (2010) could be applied for model selection under a Gaussian assumption. In recent work, Wood et al. (2016) develop a more heuristic approach that—similar to the AIC of Greven and Kneib (2010)—accounts for the uncertainty of the variance parameters. Their approach is implemented in the R add-on package `mgcv` which allows its direct application.

Moreover, it would be interesting to further consider how many levels for each grouping factor are necessary for a meaningful analysis in FLMMs, which is of great importance in scientific fields in which the sample sizes are limited, e.g., due to high experimental costs.

Finally, further extension of the implementations would ease applicability of the proposed methods, in particular, extending the implementation for sparse and irregular grids provided in the R add-on package `sparseFLMM` (Cederbaum, 2016) to allow for more general correlation structures. Particularly convenient would be a common interface for the two estimation approaches in Chapters 3 and 4 that makes use of the fast symmetric smoothing approach (Chapter 5).

Overall, the analysis of correlated functional data is a broad field that entails numerous challenges, especially when the data are sampled on curve-specific, possibly sparse grids. I hope that this thesis makes a contribution to the field.

Appendices

Appendix A

Notation

Numbers

\mathbb{R}	real numbers
\mathbb{N}	integers

Matrix algebra

$\mathbf{A} \otimes \mathbf{B}$	Kronecker product of matrices \mathbf{A} and \mathbf{B}
$\mathbf{A} \cdot \mathbf{B}$	Hadamard (point-wise) product of matrices \mathbf{A} and \mathbf{B}
$\text{vec}(\mathbf{A})$	vec operator applied to matrix \mathbf{A}
$\text{diag}(a_1, \dots, a_n)$	$n \times n$ diagonal matrix with diagonal elements a_1, \dots, a_n
$\text{blockdiag}(\mathbf{A}_1, \dots, \mathbf{A}_n)$	block diagonal matrix with diagonal blocks $\mathbf{A}_1, \dots, \mathbf{A}_n$
$\mathbf{1}_F$	$F \times 1$ vector of ones
$\mathbf{0}_{x \times y}$	$x \times y$ matrix of zeros
\mathbf{I}_x	identity matrix of size x
$[a_{ij}]_{i=1, \dots, n, j=1, \dots, m}$	$n \times m$ matrix with elements a_{ij}
$[a_i(t)]_{i=1, \dots, n, t \in T}$	matrix with elements $a_i(t)$
$[\mathbf{A} \mathbf{B}]$	column-wise concatenated matrices \mathbf{A}, \mathbf{B}
\mathbf{A}^\top	transpose of matrix \mathbf{A}
$\det(\mathbf{A})$	determinant of matrix \mathbf{A}
$\text{tr}(\mathbf{A})$	trace of matrix \mathbf{A}
$\langle \cdot, \cdot \rangle$	Euclidean scalar product
$\ \cdot\ $	Euclidean norm

Functional analysis

$L^2[a, b]$	space of square integrable functions on $[a, b]$
$f \equiv g$	identical equality of functions f and g , i.e., $f(x) = g(x)$ for all x
$\langle \cdot, \cdot \rangle$	L^2 -inner product
$\ \cdot\ $	L^2 -norm

Geometry

$\sin(t)$	trigonometric function sine
$\cos(t)$	trigonometric function cosine

General symbols

$x := y$	assign y to x
\approx	approximately equal
const	constant
$ \mathcal{T} $	length of interval \mathcal{T}
δ_{xy}	Kronecker delta; equal to one if $x = y$ and equal to zero otherwise
$\mathcal{O}[f(n)]$	order of complexity of f in n
$\log(a)$	natural logarithm of a
$\log_{10}(a)$	common logarithm of a

Statistical symbols

\sim	distributed as
i. i. d.	independent and identically distributed
$X Y$	X conditioned on Y
$\mathbb{E}()$	mean
$\text{Var}()$	variance
$\text{Cov}()$	covariance
\bar{x}	empirical mean of \mathbf{x}
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	(multivariate) normal distribution with mean $\boldsymbol{\mu}$ and variance-covariance matrix $\boldsymbol{\Sigma}$
$\mathcal{U}[a, b]$	discrete uniform distribution on the interval $[a, b]$
$\hat{\mathbf{b}}$	estimate of \mathbf{b}
$L()$	Likelihood function
$\ell()$	log-likelihood function
$\text{ls}_{\text{pen}}()$	penalized least squares criterion

Special symbols

$/s/$	voiceless alveolar sibilant [s]
$/sh/$	voiceless palato-alveolar sibilant [ʃ]
$/s\#sh/$	phonetic sequence of first $/s/$ and then $/sh/$, with $\#$: word boundary
δ^ε	indicator vector whose elements take values $\delta_{jj'}$
$\delta_{X(i)X(m)}$	indicator which takes value one if curves i and m belong to the same level of grouping factor X and zero otherwise
\mathbf{X}^Δ	submatrix (subvector) of \mathbf{X} corresponding to the reduced products of the centered functional responses used for the covariance estimation in Chapter 5
$\mathbf{X}_{t<t'}, \mathbf{X}_{t=t'}, \mathbf{X}_{t>t'}$	submatrices (subvectors) corresponding to the products of the centered functional responses $\tilde{y}_{it_j} \tilde{y}_{it_{j'}}$ with $t_{ij} < t_{ij'}$, $t_{ij} = t_{ij'}$, and $t_{ij} > t_{ij'}$, respectively
\mathbf{X}^r	reduced matrix (vector) \mathbf{X} resulting from enforcing the symmetry constraint in Chapter 5
$\mathbf{X}_{b<b'}, \mathbf{X}_{b=b'}, \mathbf{X}_{b>b'}$	submatrices (subvectors) corresponding to the entries below, on, and above the diagonal, respectively, in a bivariate smoothing problem with a symmetric spline coefficient matrix
$\mathbf{X}_{a\cdot c}, \mathbf{X}_{\cdot c}$	matrices containing columns \mathbf{X}_{abc} and \mathbf{X}_{bc} , respectively, for all values of b
$\mathbf{Y}^{-i}(t)$	column in matrix \mathbf{Y} corresponding to t , after deleting the rows corresponding to i

Abbreviations

BLUE	best linear unbiased estimator
B(L)UP	best (linear) unbiased predictor
CCA	canonical correlation analysis
CB	confidence band
EB(L)UP	empirical best (linear) unbiased predictor
FACE	fast covariance estimation for sparse functional data
FACE-STEP-1	first iteration of algorithm FACE

FAMM	functional additive mixed model
FDA	functional data analysis
FLMM	functional linear mixed model
FPC	functional principal component
FPCA	functional principal component analysis
FPC-FAMM	functional additive mixed models with eigen bases
fRE	functional random effect
fRI	functional random intercept
fRS	functional random slope
GCV	generalized cross-validation
ICA	independent component analysis
IQR	inter quartile range
KL expansion	Karhunen-Loève expansion
LDA	longitudinal data analysis
LMM	linear mixed model
MACS	Multicenter AIDS Cohort Study
ML	maximum likelihood
PC	principal component
PCA	principal component analysis
PCR	principal component regression
PDA	penalized discriminant analysis
PLS	partial least squares
REML	restricted maximum likelihood
rMSE	root mean squared error
rrMSE	root relative mean squared error
SC	seroconversion
spline-FAMM	functional additive mixed models with spline bases
SVD	singular value decomposition
TRI	bivariate smoothing of the upper triangle without symmetry constraint
TRI-CONSTR	bivariate smoothing of the upper triangle with symmetry constraint
TRI-CONSTR-W	bivariate smoothing of the upper triangle with symmetry constraint and diagonal weights
WHOLE	bivariate smoothing of the whole surface without symmetry constraint

Appendix B

Appendix of Chapter 3

Appendix B is based on the appendix of the following working paper in preparation:

Greven, S., Cederbaum, J., and Shou, H. (2016): Principal component-based functional linear mixed models. *Working paper*.

This appendix is divided into two main parts. The first part shows the derivation of the representation of the covariance matrices and of the basis weights and elaborates on the computational effort. The second part provides supplementary details on the analysis of the tissue spectroscopy data analyzed in Chapter 3.

B.1 Derivations

B.1.1 Representation of the covariance matrices and computational effort

Let in the following for ease of notation denote

$$\mathbf{A} := \mathbf{Z}_{\cdot 1}^{U_1} \quad \mathbf{B} := \mathbf{Z}_{\cdot \rho^{U_1}}^{U_1}, \quad \text{and } \mathbf{C} := \mathbf{Z}_{\cdot \rho^{U_G}}^{U_G}.$$

Thus, \mathbf{X}_Z can be written as

$$\mathbf{X}_Z = \left[\text{vec}(\mathbf{A}\mathbf{A}^\top) \middle| \dots \middle| \text{vec}(\mathbf{A}\mathbf{B}^\top) \middle| \dots \middle| \text{vec}(\mathbf{B}\mathbf{B}^\top) \middle| \dots \middle| \text{vec}(\mathbf{C}\mathbf{C}^\top) \right].$$

Using rules for the vec operator and Kronecker products of matrices

$$\text{vec}(\mathbf{X})^\top \text{vec}(\mathbf{Y}) = \text{tr}(\mathbf{X}^\top \mathbf{Y})$$

for $m \times n$ matrices \mathbf{X} and \mathbf{Y} with $\text{tr}(\mathbf{X})$ the trace of matrix \mathbf{X} and

$$\text{vec}(\mathbf{X}\mathbf{Y}\mathbf{Z}) = (\mathbf{Z}^\top \otimes \mathbf{X}) \text{vec}(\mathbf{Y})$$

for $m \times n$, $n \times p$ and $p \times q$ matrices \mathbf{X} , \mathbf{Y} and \mathbf{Z} (Harville, 1997, Chapter 16), we obtain

$$\mathbf{X}_Z^\top \mathbf{X}_Z = \begin{bmatrix} \text{tr}(\mathbf{A}^\top \mathbf{A} \mathbf{A}^\top \mathbf{A}) & \cdots & \text{tr}(\mathbf{A}^\top \mathbf{B} \mathbf{A}^\top \mathbf{A}) & \cdots & \text{tr}(\mathbf{A}^\top \mathbf{B} \mathbf{B}^\top \mathbf{A}) & \cdots & \text{tr}(\mathbf{A}^\top \mathbf{C} \mathbf{C}^\top \mathbf{A}) \\ \text{tr}(\mathbf{B}^\top \mathbf{A} \mathbf{A}^\top \mathbf{A}) & \cdots & \text{tr}(\mathbf{B}^\top \mathbf{B} \mathbf{A}^\top \mathbf{A}) & \cdots & \text{tr}(\mathbf{B}^\top \mathbf{B} \mathbf{B}^\top \mathbf{A}) & \cdots & \text{tr}(\mathbf{B}^\top \mathbf{C} \mathbf{C}^\top \mathbf{A}) \\ \text{tr}(\mathbf{B}^\top \mathbf{A} \mathbf{A}^\top \mathbf{B}) & \cdots & \text{tr}(\mathbf{B}^\top \mathbf{B} \mathbf{A}^\top \mathbf{B}) & \cdots & \text{tr}(\mathbf{B}^\top \mathbf{B} \mathbf{B}^\top \mathbf{B}) & \cdots & \text{tr}(\mathbf{B}^\top \mathbf{C} \mathbf{C}^\top \mathbf{B}) \\ \text{tr}(\mathbf{C}^\top \mathbf{A} \mathbf{A}^\top \mathbf{C}) & \cdots & \text{tr}(\mathbf{C}^\top \mathbf{B} \mathbf{A}^\top \mathbf{C}) & \cdots & \text{tr}(\mathbf{C}^\top \mathbf{B} \mathbf{B}^\top \mathbf{C}) & \cdots & \text{tr}(\mathbf{C}^\top \mathbf{C} \mathbf{C}^\top \mathbf{C}) \end{bmatrix}$$

and

$$\mathbf{X}_Z^\top [(\mathbf{Y} - \boldsymbol{\mu}) \otimes (\mathbf{Y} - \boldsymbol{\mu})] = \begin{bmatrix} \text{vec} [(\mathbf{Y} - \boldsymbol{\mu})^\top \mathbf{A} \mathbf{A}^\top (\mathbf{Y} - \boldsymbol{\mu})]^\top \\ \vdots \\ \text{vec} [(\mathbf{Y} - \boldsymbol{\mu})^\top \mathbf{A} \mathbf{B}^\top (\mathbf{Y} - \boldsymbol{\mu})]^\top \\ \vdots \\ \text{vec} [(\mathbf{Y} - \boldsymbol{\mu})^\top \mathbf{B} \mathbf{B}^\top (\mathbf{Y} - \boldsymbol{\mu})]^\top \\ \vdots \\ \text{vec} [(\mathbf{Y} - \boldsymbol{\mu})^\top \mathbf{C} \mathbf{C}^\top (\mathbf{Y} - \boldsymbol{\mu})]^\top \end{bmatrix}.$$

The matrices $\mathbf{Z}_{.s}^{U_g}$ are of dimension $n \times L^{U_g}$ and of block diagonal forms. Computational effort for $\mathbf{X}_Z^\top [(\mathbf{Y} - \boldsymbol{\mu}) \otimes (\mathbf{Y} - \boldsymbol{\mu})]$ thus is $\mathcal{O}(nD + L^{U_g} D^2)$ for each entry and $\mathcal{O}(nD^2)$ for the last entry if $\mathbf{Z}_{.\rho^{U_G}}^{U_G} = \mathbf{I}_n$ is used for a smooth error term. As $L^{U_g} \leq n$ for all g , the effort is dominated by the last entry and is $\mathcal{O}(nD^2)$ in total.

For the computation of $\mathbf{X}_Z^\top \mathbf{X}_Z$, we use that $\text{tr}(\mathbf{X}^\top \mathbf{Y}) = \sum_i \sum_j x_{ij} y_{ij}$ for $\mathbf{X} = (x_{ij})_{i,j}$ and $\mathbf{Y} = (y_{ij})_{i,j}$ of the same dimensions. Computational effort thus includes computing all products $\mathbf{Z}_{.s}^{U_g \top} \mathbf{Z}_{.s'}^{U_h}$, $\mathcal{O}(L^{U_g} L^{U_h} n)$ each or $\mathcal{O}(q^2 n)$ total, the computation of the entry-wise matrix products and sums of all entries, $\mathcal{O}(L^{U_g} L^{U_h})$ each or $\mathcal{O}(\rho_{\max}^2 q^2)$ total with $\rho_{\max} = \max(\rho^{U_g}, g = 1, \dots, G)$.

The inversion of $\mathbf{X}_Z^\top \mathbf{X}_Z$ and the final multiplication of $\mathbf{X}_Z^\top \mathbf{X}_Z$ and $\mathbf{X}_Z^\top [(\mathbf{Y} - \boldsymbol{\mu}) \otimes (\mathbf{Y} - \boldsymbol{\mu})]$ are at most quadratic in D and thus the overall effort is dominated by the computation of the raw covariance of \mathbf{Y} , $(\mathbf{Y} - \boldsymbol{\mu})^\top \mathbf{C} \mathbf{C}^\top (\mathbf{Y} - \boldsymbol{\mu}) = (\mathbf{Y} - \boldsymbol{\mu})^\top (\mathbf{Y} - \boldsymbol{\mu})$, and is $\mathcal{O}(nD^2)$ total.

B.1.2 Representation of the random basis weights and computational effort

The BLUPs can be written as

$$\begin{aligned} \hat{\xi} &= \left(\mathbf{Z}_{\Phi}^{\top} \mathbf{Z}_{\Phi} + \sigma^2 \mathbf{G}^{-1} \right)^{-1} \mathbf{Z}_{\Phi}^{\top} \text{vec}(\mathbf{Y} - \boldsymbol{\mu}) \\ &= \left\{ \begin{array}{l} \left[\begin{array}{ccc} \sum_{s=1}^{\rho^{U_1}} \sum_{s'=1}^{\rho^{U_1}} \left(\Phi_s^{U_1} \Phi_{s'}^{U_1 \top} \right) \otimes \left(\mathbf{Z}_{\cdot s}^{U_1 \top} \mathbf{Z}_{\cdot s'}^{U_1} \right) & \dots & \sum_{s=1}^{\rho^{U_1}} \sum_{s'=1}^{\rho^{U_G}} \left(\Phi_s^{U_1} \Phi_{s'}^{U_G \top} \right) \otimes \left(\mathbf{Z}_{\cdot s}^{U_G \top} \mathbf{Z}_{\cdot s'}^{U_G} \right) \\ \vdots & & \vdots \\ \sum_{s=1}^{\rho^{U_G}} \sum_{s'=1}^{\rho^{U_1}} \left(\Phi_s^{U_G} \Phi_{s'}^{U_1 \top} \right) \otimes \left(\mathbf{Z}_{\cdot s}^{U_G \top} \mathbf{Z}_{\cdot s'}^{U_1} \right) & \dots & \sum_{s=1}^{\rho^{U_G}} \sum_{s'=1}^{\rho^{U_G}} \left(\Phi_s^{U_G} \Phi_{s'}^{U_G \top} \right) \otimes \left(\mathbf{Z}_{\cdot s}^{U_G \top} \mathbf{Z}_{\cdot s'}^{U_G} \right) \otimes \mathbf{I}_n \end{array} \right] \\ + \sigma^2 \text{blockdiag} \left[\text{diag} \left(\frac{1}{\nu_1^{U_g}}, \dots, \frac{1}{\nu_{N^{U_g}}^{U_g}} \right) \otimes \mathbf{I}_{L^{U_g}} \right] \end{array} \right\}^{-1} \begin{bmatrix} \text{vec} \left(\sum_{s=1}^{\rho^{U_1}} \mathbf{Z}_{\cdot s}^{U_1 \top} (\mathbf{Y} - \boldsymbol{\mu}) \Phi_s^{U_1 \top} \right) \\ \vdots \\ \text{vec} \left(\sum_{s=1}^{\rho^{U_G}} \mathbf{Z}_{\cdot s}^{U_G \top} (\mathbf{Y} - \boldsymbol{\mu}) \Phi_s^{U_G \top} \right) \end{bmatrix}. \end{aligned}$$

Entries in the vector can be computed with computational effort $\mathcal{O}(nD)$. Computation of the entries in the matrix have effort of smaller order, while inversion of the matrix and matrix-vector multiplication are of order $\mathcal{O} \left[\left(\sum_{g=1}^G N^{U_g} L^{U_g} \right)^3 \right]$ and $\mathcal{O} \left[\left(\sum_{g=1}^G N^{U_g} L^{U_g} \right)^2 \right]$, respectively. If the last functional random effect represents a smooth error term with $L^{U_G} = n$, this would be cubic in n^3 . The order can then be reduced to $\mathcal{O} \left[\left(\sum_{g=1}^{G-1} N^{U_g} L^{U_g} \right)^3 \right]$ using the Schur complement (see, e.g., Harville, 1974, Chapter 8) with respect to the bottom right entry of the matrix, which is less than $\mathcal{O}(n^3)$ for reasonably small values G and N^{U_g} . The overall computational effort thus is at most $\mathcal{O}(nD + n^3)$.

B.2 Details on the application

We perform the following nested leave-one-pig-out cross-validation for the classification of the tissue types in our application to the spectroscopy data:

Step 1 (outer) Split the spectroscopy data into 12 parts, each part consisting of all observations of one single pig. In each outer iteration, use 11 out of the 12 parts as training data and the remaining as test data.

Step 1 (inner) Split the training data again and leave out one additional pig. In each inner iteration, use 10 out of the 11 parts as training data for fitting the functional linear mixed model in (3.7) and the remaining data as test data. The following ten different pre-specified levels of variance explained are used for fitting:

$$L \in \{0.995, 0.999, 0.9995, 0.9999, 0.99995, 0.99999, 0.999995, 0.999999, 0.9999995, 0.9999999\}.$$

Step 2 (inner) Choose the optimal level of variance explained as the level that yields the best prediction on the test data.

Step 2 (outer) Estimate the functional linear mixed model in (3.7) using the corresponding optimal level of variance explained and predict the tissue types in the test data.

Appendix C

Appendix of Chapter 4

Appendix C is based on the appendix of the following paper:

Cederbaum, J., Pouplier, M., Hoole, P., Greven, S. (2016): Functional linear mixed models for irregularly or sparsely sampled data. *Statistical Modelling*, 16(1):67–88.

This appendix is divided into four main parts. The first part shows the derivation of the variance decomposition, defines the empirical best linear unbiased predictor in its usual form, and gives the concrete forms of the matrices used in the estimation of the mean function and in the prediction of the basis weights. The second part provides supplementary details on the estimation and implementation of our approach. The third part gives additional results for our application to the speech production data and a detailed description of the pre-processing. The last part gives the concrete forms of all measures of goodness of fit used in our simulations, provides more details on the data generation, and shows supplementary simulation results for all simulations including those with uncentered and non-decorrelated basis weights.

C.1 Derivations

Derivation of the variance decomposition of Model (4.2) in Chapter 4. Using iterated expectations allows to decompose the variance of the response as follows

$$\begin{aligned}
\int_{\mathcal{T}} \text{Var} [Y_{ijh}(t)] dt &= \int_{\mathcal{T}} \text{Var} [B_i(t)] dt + \int_{\mathcal{T}} \text{Var} [C_j(t)] dt + \int_{\mathcal{T}} \text{Var} [E_{ijh}(t)] dt \\
&+ \int_{\mathcal{T}} \varepsilon_{ijh}(t) dt \\
&= \sum_{k=1}^{\infty} \nu_k^B \underbrace{\int_{\mathcal{T}} \phi_k^B(t) \phi_k^B(t) dt}_{=1} + \sum_{k=1}^{\infty} \nu_k^C \underbrace{\int_{\mathcal{T}} \phi_k^C(t) \phi_k^C(t) dt}_{=1} \\
&+ \sum_{k=1}^{\infty} \nu_k^E \underbrace{\int_{\mathcal{T}} \phi_k^E(t) \phi_k^E(t) dt}_{=1} + \int_{\mathcal{T}} \sigma^2 dt \\
&= \sum_{k=1}^{\infty} \nu_k^B + \sum_{k=1}^{\infty} \nu_k^C + \sum_{k=1}^{\infty} \nu_k^E + \sigma^2 |\mathcal{T}|.
\end{aligned}$$

C.1.1 Empirical best linear unbiased predictor (EBLUP)

Consider Model (4.2) in Chapter 4. The EBLUP in the usual form is given by

$$\hat{\xi} = \hat{\mathbf{G}} \hat{\mathbf{\Phi}}^\top \widehat{\text{Cov}}(\tilde{\mathbf{Y}})^{-1} \tilde{\mathbf{Y}} = \hat{\mathbf{G}} \hat{\mathbf{\Phi}}^\top (\hat{\sigma}^2 \mathbf{I}_{\mathfrak{D}} + \hat{\mathbf{\Phi}} \hat{\mathbf{G}} \hat{\mathbf{\Phi}}^\top)^{-1} \tilde{\mathbf{Y}}, \quad (\text{C.1})$$

where $\hat{\mathbf{G}}$ denotes the estimate of the covariance matrix of the FPC weights and $\hat{\mathbf{\Phi}}$ is the joint design matrix of the form $\hat{\mathbf{\Phi}} = [\hat{\mathbf{\Phi}}^B | \hat{\mathbf{\Phi}}^C | \hat{\mathbf{\Phi}}^E]$, where $\hat{\mathbf{\Phi}}^X$, $X \in \{B, C, E\}$, are the respective design matrices containing the rescaled FPC estimates evaluated on the original observation points. $\tilde{\mathbf{Y}}$ denotes the stacked centered response vector of length \mathfrak{D} , the total number of observation points and $\hat{\sigma}^2$ is the estimated error variance.

C.1.2 Matrices in the prediction of the basis weights as EBLUPs

$\hat{\mathbf{\Phi}}^B$ is a block diagonal matrix of dimension $\mathfrak{D} \times IN^B$.

$$\hat{\mathbf{\Phi}}^B = \begin{bmatrix} \hat{\phi}_1^B(t_{1111}) & \cdots & \hat{\phi}_{NB}^B(t_{1111}) & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ \hat{\phi}_1^B(t_{1JH_{1J}T_{1JH_{1J}}}) & \cdots & \hat{\phi}_{NB}^B(t_{1JH_{1J}T_{1JH_{1J}}}) & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & \cdots 0 & & \hat{\phi}_1^B(t_{I111}) & \cdots & \hat{\phi}_{NB}^B(t_{I111}) \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & \cdots 0 & & \hat{\phi}_1^B(t_{IJH_{1J}T_{1JH_{1J}}}) & \cdots & \hat{\phi}_{NB}^B(t_{IJH_{1J}T_{1JH_{1J}}}) \end{bmatrix}$$

$\widehat{\Phi}^C$ consists of I block diagonal matrices—one for each speaker—and is of dimension $\mathfrak{D} \times JN^C$.

$$\widehat{\Phi}^C = \begin{bmatrix} \widehat{\phi}_1^C(t_{1111}) & \cdots & \widehat{\phi}_{N^C}^C(t_{1111}) \\ \vdots & & \vdots \\ \widehat{\phi}_1^C(t_{11H_{11}T_{11H_{11}}}) & \cdots & \widehat{\phi}_{N^C}^C(t_{11H_{11}T_{11H_{11}}}) \\ & \ddots & \\ & & \widehat{\phi}_1^C(t_{1J11}) & \cdots & \widehat{\phi}_{N^C}^C(t_{1J11}) \\ & & \vdots & & \vdots \\ & & \widehat{\phi}_1^C(t_{1JH_{1J}T_{1JH_{1J}}}) & \cdots & \widehat{\phi}_{N^C}^C(t_{1JH_{1J}T_{1JH_{1J}}}) \\ & & \vdots & & \vdots \\ \widehat{\phi}_1^C(t_{I111}) & \cdots & \widehat{\phi}_{N^C}^C(t_{I111}) \\ \vdots & & \vdots \\ \widehat{\phi}_1^C(t_{I1H_{I1}T_{I1H_{I1}}}) & \cdots & \widehat{\phi}_{N^C}^C(t_{I1H_{I1}T_{I1H_{I1}}}) \\ & \ddots & \\ & & \widehat{\phi}_1^C(t_{IJ11}) & \cdots & \widehat{\phi}_{N^C}^C(t_{IJ11}) \\ & & \vdots & & \vdots \\ & & \widehat{\phi}_1^C(t_{IJH_{IJ}T_{IJH_{IJ}}}) & \cdots & \widehat{\phi}_{N^C}^C(t_{IJH_{IJ}T_{IJH_{IJ}}}) \end{bmatrix}$$

Obviously, the role of speakers and target words is exchangeable. For present purposes, however, we assume that all vectors and matrices are first ordered by speakers and within each speaker ordered by target words. $\widehat{\Phi}^E$ is a block diagonal matrix of dimension $\mathfrak{D} \times nN^E$ with blocks

$$\begin{bmatrix} \widehat{\phi}_1^E(t_{1111}) & \cdots & \widehat{\phi}_{N^E}^E(t_{1111}) \\ \vdots & & \vdots \\ \widehat{\phi}_1^E(t_{111T_{111}}) & \cdots & \widehat{\phi}_{N^E}^E(t_{111T_{111}}) \end{bmatrix}, \dots, \begin{bmatrix} \widehat{\phi}_1^E(t_{IJH_{IJ}1}) & \cdots & \widehat{\phi}_{N^E}^E(t_{IJH_{IJ}1}) \\ \vdots & & \vdots \\ \widehat{\phi}_1^E(t_{IJH_{IJ}T_{IJH_{IJ}}}) & \cdots & \widehat{\phi}_{N^E}^E(t_{IJH_{IJ}T_{IJH_{IJ}}}) \end{bmatrix}.$$

Note that for irregularly spaced functional data, the FPCs evaluated on the original observation points are curve-specific. The estimated covariance matrix of the ξ , $\widehat{\mathbf{G}}$, is of the form

$$\widehat{\mathbf{G}} = \begin{bmatrix} \widehat{\text{Cov}}(\xi^B) & & \\ & \widehat{\text{Cov}}(\xi^C) & \\ & & \widehat{\text{Cov}}(\xi^E) \end{bmatrix},$$

where each covariance matrix $\widehat{\text{Cov}}(\xi^X)$, $X \in \{B, C, E\}$, is a diagonal matrix with elements $\widehat{\nu}_1^X, \dots, \widehat{\nu}_{N^X}^X, \dots, \widehat{\nu}_1^X, \dots, \widehat{\nu}_{N^X}^X$.

C.1.3 Marginal bases for FPC-FAMM

The mean is re-estimated in the FAMM with Ψ_c^k , $k = 1, \dots, p$, an inflated vector of length \mathcal{D} containing the values of covariate $\mathbf{x}_k = (x_{111k}, \dots, x_{IJH_IJk})^\top$. For each curve, the covariate values are replicated for each observation point yielding

$$\Psi_c^k = \begin{pmatrix} x_{111k} \\ \vdots \\ x_{111k} \\ \vdots \\ x_{IJH_IJk} \\ \vdots \\ x_{IJH_IJk} \end{pmatrix}.$$

For the functional intercept $f_0(t)$, $\Psi_c^0 = \mathbf{1}_{\mathcal{D}}$. Ψ_t^k , $k = 0, \dots, p$, is a $\mathcal{D} \times F^k$ matrix comprising the evaluations of the basis functions on the original observations points of the form

$$\Psi_t^k = \begin{bmatrix} \psi_1^k(t_{1111}) & \cdots & \psi_{F^k}^k(t_{1111}) \\ \vdots & & \vdots \\ \psi_1^k(t_{111T_{111}}) & \cdots & \psi_{F^k}^k(t_{111T_{111}}) \\ \vdots & & \vdots \\ \psi_1^k(t_{IJH_IJ1}) & \cdots & \psi_{F^k}^k(t_{IJH_IJ1}) \\ \vdots & & \vdots \\ \psi_1^k(t_{IJH_IJT_{IJH_IJ}}) & \cdots & \psi_{F^k}^k(t_{IJH_IJT_{IJH_IJ}}) \end{bmatrix}.$$

In the estimation of the FPC weights, Ψ_g^B is a $\mathcal{D} \times I$ incidence matrix of which the entries in the d th column are one wherever the row belongs to the d th speaker and zero otherwise. Ψ_g^C and Ψ_g^E are analogously $\mathcal{D} \times J$ and $\mathcal{D} \times n$ matrices with entries in the d th column equal to one wherever the row belongs to the d th target word or to the d th curve, respectively. In the following, the part of the matrices Ψ_g^X , $X \in \{B, C, E\}$, that belongs to the i th speaker (denoted by $\Psi_g^{X,i}$) is exemplarily shown, assuming that the data are ordered by speakers, within each speaker ordered by target words, within each target word ordered by repetition, then by curves, and finally by observation points.

C.2.2 Rescaling of the eigenvectors and eigenvalues

Denote the estimated eigenvectors by $\widehat{\phi}_k^X = \left[\widehat{\phi}_k^X(t_1), \dots, \widehat{\phi}_k^X(t_{\bar{D}}) \right]^\top$, and the estimated eigenvalues by $\widehat{\nu}_k^X$, $k \in \mathbb{N}$, $X \in \{B, C, E\}$. In order to ensure that the approximated functions are orthonormal with respect to the L^2 -scalar product $\langle f, g \rangle = \int f(t)g(t) dt$, we rescale the eigenvectors by

$$\widehat{\phi}_k^X = \frac{1}{\sqrt{a}} \widetilde{\phi}_k^X,$$

with a denoting the constant interval width of the equidistant grid $\{t_1, \dots, t_{\bar{D}}\}$. As a consequence, the eigenvalues to the rescaled eigenvectors need to be adjusted as $\widehat{\nu}_k^X = a\widetilde{\nu}_k^X$.

C.2.3 Truncation of the FPCs

Due to the additive structure in the variance decomposition, we can choose the truncation lags N^B , N^C , and N^E in the following way:

1. Specify the proportion of explained variance L , e.g., $L = 0.95$ as used in the application in Section 4.4 of Chapter 4.
2. Select the FPCs corresponding to their eigenvalues in decreasing order, until

$$\frac{\sum_{k=1}^{N^B} \nu_k^B + \sum_{k=1}^{N^C} \nu_k^C + \sum_{k=1}^{N^E} \nu_k^E + \sigma^2|\mathcal{T}|}{\sum_{k=1}^{\infty} \nu_k^B + \sum_{k=1}^{\infty} \nu_k^C + \sum_{k=1}^{\infty} \nu_k^E + \sigma^2|\mathcal{T}|} \geq L. \quad (\text{C.2})$$

Note that the three random processes are treated equally, i.e., in each step, the FPC with the highest corresponding eigenvalue is chosen regardless of the associated process. In practice, all infinite sums in Criterion (C.2) are approximated by the finite sums of all obtained eigenvalues. The eigenvalues and the error variance are replaced by their estimates.

C.2.4 Implementation of FPC-FAMM

In practice, we base the FPC-FAMM estimation on the R function `pffr` that Scheipl et al. (2015) provide in the R add-on package `refundDevel` (Huang et al., 2016b). Function `pffr` is a wrapper function for function `gam` and for related functions in the package `mgcv` (Wood, 2006, 2011) and therefore builds on existing flexible and robust software. We use the `pffr`-constructor `pcre` for FPC-based fRIs for the prediction of the random processes. A constraint on the functional random effects assures that they are centered. The resulting point-wise CBs are with a constraint correction (Marra and Wood, 2012).

C.2.5 Fixing the smoothing parameter in FPC-FAMM

Technically, in order that a fixed smoothing parameter λ_* (here $\lambda_* = 1$) is used in the prediction of the FPC-based fRIs, the smoothing parameter in function `pffr` has to be specified as

$$\lambda_{\text{fix}} = \lambda_* \mathbf{S.scale} \hat{\sigma}^2, \quad (\text{C.3})$$

where `S.scale` is a scaling factor used in the set-up of the design matrices in order to numerically stabilize the prediction. It is given out as part of the smooth term in the output of function `pffr` and can be obtained previously to the estimation by setting `fit = FALSE` in the call of function `pffr`. The estimate of the error variance $\hat{\sigma}^2$ can be taken from Step 2 of our estimation procedure. Note that Equation (C.3) makes clear that the point-wise confidence bands for the mean function are not only conditional on the estimated FPCs and the truncation level, but also on the estimated error variance.

C.2.6 Iterative estimation

If desired, the estimation accuracy can be improved by applying the estimation steps iteratively. At least two possibilities exist: One can either perform Steps 1 to 4 and then re-start with Step 1 with the adjusted observations $Y_{ijh}^*(t) := Y_{ijh}(t) - \hat{B}_i(t) - \hat{C}_j(t) - \hat{E}_{ijh}(t)$ until a pre-defined criterion is reached. Alternatively, one can replace the mean with that obtained in the FAMM framework in Step 4 and then restart with Step 2.

C.3 Supplementary application details and results

C.3.1 Pre-processing

The index calculation is based on the calculation of the power spectrum over a time window of approximately 20 ms, shifted in 5 ms steps over the time interval of the consonants. In order to be able to compare the index curves for the two consonant orders (`/s#sh/`, `/sh#s/`) directly, the index curves of `/sh#s/` were mirrored along the time axis (mapping `+1` to `-1` and vice versa) such that for both orders the index dynamic ranges from `+1` for the first consonant to `-1` for the second consonant rather than from `+1` for `/s/` to `-1` for `/sh/`. To achieve this, first smooth mean curves of available reference curves of orders `/sh#sh/` and `/s#s/` per speaker and for each combination of the covariates `vowel`, `stress1`, `stress2` were estimated using penalized splines and evaluated on the measurement points. The index curves of order `/sh#s/` were then mirrored at the speaker-condition specific mean values, averaging over the mean curves for `/sh#s/` and `/s#sh/`.

C.3.2 Supplementary application results

In the following, we show additional application results, including the eigenvalues, the variance decomposition, the effects of the covariates and interactions, the second FPC for speakers, and the

Table C.1: Estimated eigenvalues and estimated error variance for the model with covariates in the following order: Estimated eigenvalue corresponding to the first and second FPCs for speakers, estimated eigenvalue corresponding to the first, second, and third FPCs for the smooth error, estimated error variance.

$\hat{\nu}_1^B$	$\hat{\nu}_2^B$	$\hat{\nu}_1^E$	$\hat{\nu}_2^E$	$\hat{\nu}_3^E$	$\hat{\sigma}^2$
$5.84 \cdot 10^{-3}$	$3.23 \cdot 10^{-3}$	$19.53 \cdot 10^{-3}$	$7.59 \cdot 10^{-3}$	$2.73 \cdot 10^{-3}$	$3.94 \cdot 10^{-3}$

Table C.2: Estimated variance explained in percent of the estimated total variance given in the following order: Explained variance by the first and second FPCs for speakers, explained variance by the first, second, and third FPCs for the smooth error, estimated error variance.

$\hat{\nu}_1^B$	$\hat{\nu}_2^B$	$\hat{\nu}_1^E$	$\hat{\nu}_2^E$	$\hat{\nu}_3^E$	$\hat{\sigma}^2$
13.16%	7.29%	44.02%	17.11%	6.16%	8.88%

variance decomposition of the model without covariates on which we base our crossed-fRI simulation setting.

The estimated eigenvalues and the variance decomposition for the model with covariates included are given in Table C.1 and Table C.2, respectively. Figure C.1 shows the estimated effects and point-wise confidence bands of the covariates stress1 (0: Strong, 1: Weak), stress2 (0: Strong, 1: Weak), and vowel (0: ia, 1:ai) as well as of the interactions between covariate order and the other three covariates.

The second FPC for speakers is depicted in Figure C.2. The interpretation is as follows: The index curves of speakers with positive FPC weights for the second FPC are pulled towards the ideal reference value of the second consonant, whereas the index curves of speakers with negative FPC weights are pulled towards the ideal reference value of the first consonant.

Table C.3 gives the variance decomposition of the model with no covariate included.

Table C.3: Estimated variance explained in percent of the estimated total variance given in the following order: Explained variance by the first and second FPCs for speakers, explained variance by the first FPC for target words, estimated variance by the first, second, and third FPCs for the smooth error, estimated error variance.

$\hat{\nu}_1^B$	$\hat{\nu}_2^B$	$\hat{\nu}_1^C$	$\hat{\nu}_1^E$	$\hat{\nu}_2^E$	$\hat{\nu}_3^E$	$\hat{\sigma}^2$
$5.86 \cdot 10^{-3}$	$2.71 \cdot 10^{-3}$	$8.89 \cdot 10^{-3}$	$19.1 \cdot 10^{-3}$	$7.53 \cdot 10^{-3}$	$2.66 \cdot 10^{-3}$	$5.62 \cdot 10^{-3}$
10.83%	5.00%	16.44%	35.23%	13.93%	4.92%	10.39%

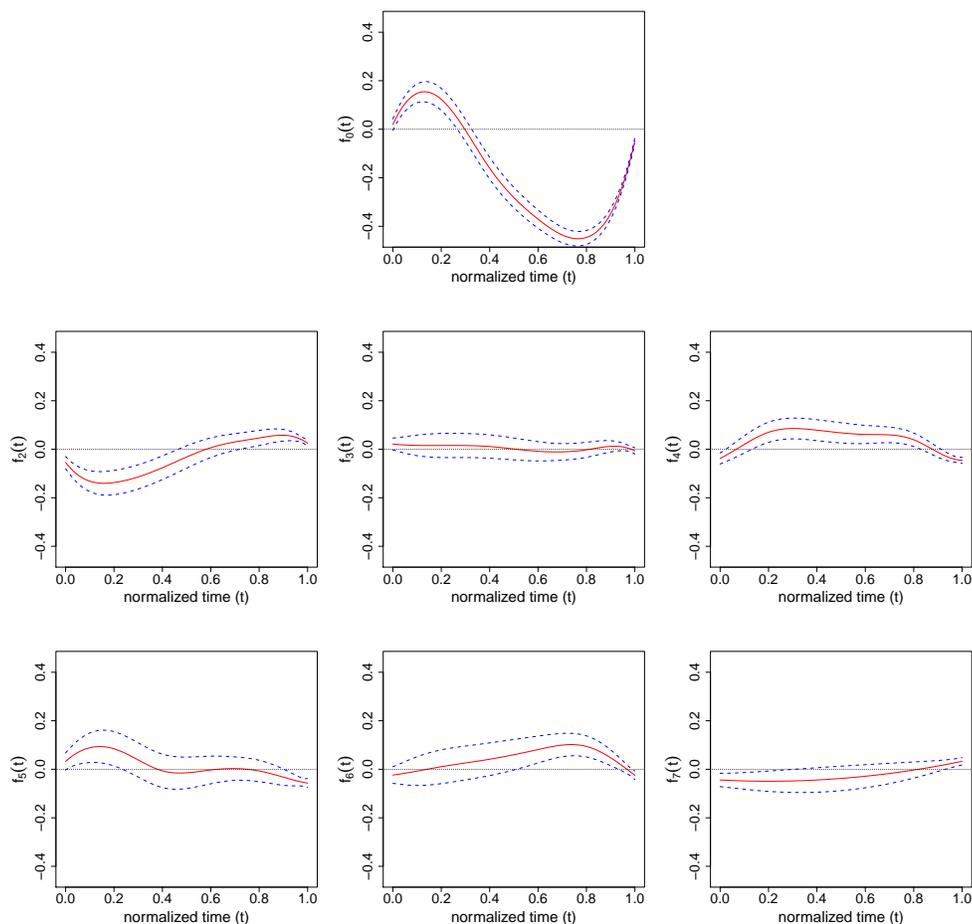


Figure C.1: Covariate mean effects (red solid lines) with conditional point-wise confidence bands (dashed lines). Upper: Reference mean $f_0(t)$. Middle (from left to right): Covariate effects of covariates stress1, stress2, vowel. Lower (from left to right): Interaction effects of order and stress1, order and stress2, order and vowel.

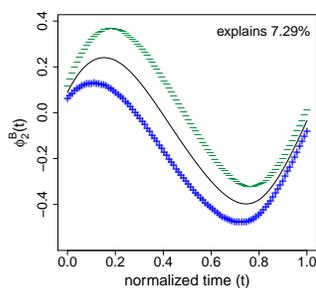


Figure C.2: Second FPC for speakers. Shown is the mean function (solid line) and the effect of adding (+) and subtracting (-) a suitable multiple ($2\sqrt{\hat{\nu}_2^B}$) of the first FPC for speakers.

C.4 Supplementary simulation details and results

C.4.1 Measures of goodness of fit

We use the root relative mean squared error

$$\text{rrMSE}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) = \sqrt{\frac{\frac{1}{L} \sum_{l=1}^L (\theta_l - \hat{\theta}_l)^2}{\frac{1}{L} \sum_{l=1}^L \theta_l^2}}, \quad (\text{C.4})$$

as a measure of goodness of fit for vector-valued estimates $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta} = (\theta_1, \dots, \theta_L)^\top$ (FPC weights $\boldsymbol{\xi}_k^X = (\xi_{1k}^X, \dots, \xi_{L^X k}^X)$, $k = 1, \dots, N^X$, $X \in \{B, C, E\}$), and for scalar estimates (eigenvalues ν_k^X , $k = 1, \dots, N^X$, $X \in \{B, C, E\}$, and the error variance σ^2) as a special case with $L = 1$. For the FPC weights, the rrMSE (C.4) is approximately

$$\sqrt{1/L^X \sum_{l=1}^{L^X} (\xi_{lk}^X - \hat{\xi}_{lk}^X)^2 / \nu_k^X}.$$

For all functions $\theta(t)$ (mean function without and with covariates $\mu(t)$, $f_0(t), \dots, f_7(t)$ and eigenfunctions $\phi_k^X(t)$, $k = 1, \dots, N^X$, $X \in \{B, C, E\}$), we approximate the integrals by sums yielding

$$\text{rrMSE} [\theta(\cdot), \hat{\theta}(\cdot)] = \sqrt{\frac{\frac{1}{\bar{D}} \sum_{d=1}^{\bar{D}} [\theta(t_d) - \hat{\theta}(t_d)]^2}{\frac{1}{\bar{D}} \sum_{d=1}^{\bar{D}} \theta(t_d)^2}}. \quad (\text{C.5})$$

Note that for the eigenfunctions, the denominator simplifies to approximately one as

$$\int \phi_k(t)^2 dt = 1.$$

As the eigenfunctions are only unique up to sign, we compare the rrMSE for the estimated eigenfunctions to that for the estimated eigenfunctions mirrored around the x-axis and choose the smaller rrMSE. For the fRIs and for the response function, we additionally average over the respective levels. As the fRIs are centered, the denominator simplifies to the average variance. The rrMSE for the bivariate functions (auto-covariances) are defined analogously (see Appendix D for a concrete definition of the rrMSE for covariances).

C.4.2 Generation details for the sparse scenario

We use each two FPCs to generate the underlying process. Eigenvalues are generated as $\nu_k^X = 2/k$, $k = 1, 2$, $X \in \{B, C, E\}$. We choose normalized Legendre Polynomials adapted to the interval $[0, 1]$

as FPCs for $B_i(t)$ and $C_j(t)$. For the smooth error $E_{ijh}(t)$, we choose a basis of sine and cosine functions. The first two elements of the orthonormal bases are:

$$\begin{aligned} \phi_1^B(t) &= 1 & \phi_1^C(t) &= \sqrt{3}(2t - 1) & \phi_1^E(t) &= \sqrt{2}\sin(2\pi t) \\ \phi_2^B(t) &= \sqrt{5}(6t^2 - 6t + 1) & \phi_2^C(t) &= \sqrt{7}(20t^3 - 30t^2 + 12t - 1) & \phi_2^E(t) &= \sqrt{2}\cos(2\pi t). \end{aligned}$$

Note that the different bases need not be mutually orthogonal.

C.4.3 Results for simulations with centered and decorrelated basis weights

Simulation results for the crossed-fRIs scenario

In the following, additional simulation results for the application-based crossed-fRIs scenario with centered and decorrelated basis weights are shown. Figure C.3 shows the true and estimated mean functions. In Figure C.4, the boxplots of the estimated eigenvalues for each $B_i(t)$, $C_j(t)$, and $E_{ijh}(t)$ are depicted. Figure C.5 shows the boxplot of the estimated error variances.

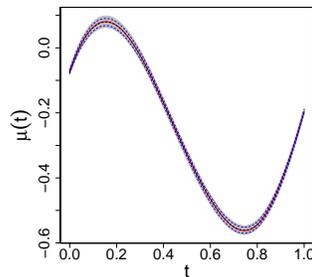


Figure C.3: True and estimated mean functions. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

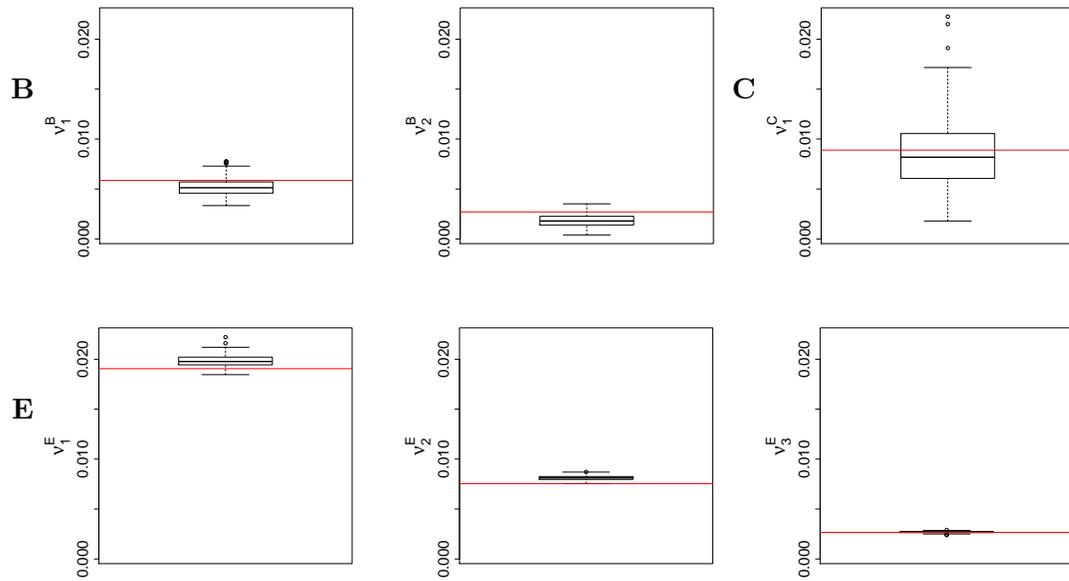


Figure C.4: Boxplots of the estimated eigenvalues of the auto-covariances of the crossed fRIs (top row), as well as the eigenvalues of the auto-covariance of the smooth error (bottom row) for all 200 simulations runs.

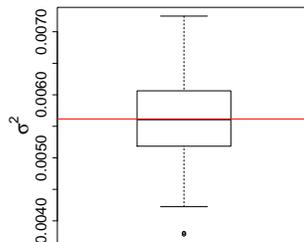


Figure C.5: Boxplot of the estimated error variances σ^2 for all 200 simulation runs.

Simulation results for the fRI scenario

In the following, we show additional simulation results for the application-based fRI scenario with centered and decorrelated basis weights. Figure C.6, Figure C.7, and Figure C.8 show the true and estimated covariate and interaction effects based on the independence assumption (Figure C.6), on FPC-FAMM estimation (Figure C.7), and on the spline-FAMM alternative (Figure C.8), respectively. The performance of the point-wise CBs obtained by FPC-FAMM and spline-FAMM is evaluated by looking at the point-wise coverage shown in Figures C.9 and C.10. Table C.4 gives the coverage averaged over all time points and simulation runs for FPC-FAMM and spline-FAMM estimation. In addition, the simultaneous coverage of the point-wise CBs in terms of percentage of completely covered curves is compared in Table C.5.

Figure C.11 depicts the true and estimated FPCs of the fRI and of the smooth error and Figure C.12 shows the boxplots of the estimated eigenvalues. The boxplot of the estimated error variances is shown in Figure C.13. Table C.6 lists the average rrMSEs for all model components except for the covariate effects and Table C.7 lists the rrMSEs for the covariate effects.

The average rrMSEs of the covariate and interaction effects lie between 0.06 ($f_0(t)$) for both estimation options and 1.43 or 1.34 ($f_3(t)$) for the estimation using the independence assumption or FPC-FAMM, respectively. Note that the high value for $f_3(t)$ is the result of the fact that the true covariate effect is very close to zero along the whole time interval, and to avoid dividing by values near zero it is more meaningful to look at the root mean squared error (rMSE) instead which is similar to the rMSEs of other covariates.

Table C.4: Point-wise coverage of the point-wise CBs for FPC-FAMM and for spline-FAMM. Shown is the coverage averaged over all time points for all covariate and interaction effects. For FPC-FAMM, the coverage refers to 200 simulation runs, whereas for splines-FAMM, 100 simulation runs are taken into account.

	$f_0(t)$	$f_1(t)$	$f_2(t)$	$f_3(t)$	$f_4(t)$	$f_5(t)$	$f_6(t)$	$f_7(t)$
$\mu(t, \mathbf{x}_{ijh})_{\text{FPC-FAMM}}$	95.19%	94.44%	95.28%	93.71%	95.54%	94.11%	91.18%	93.55%
$\mu(t, \mathbf{x}_{ijh})_{\text{spline-FAMM}}$	36.24%	36.35%	41.67%	37.13%	35.12%	38.87%	38.36%	35.94%

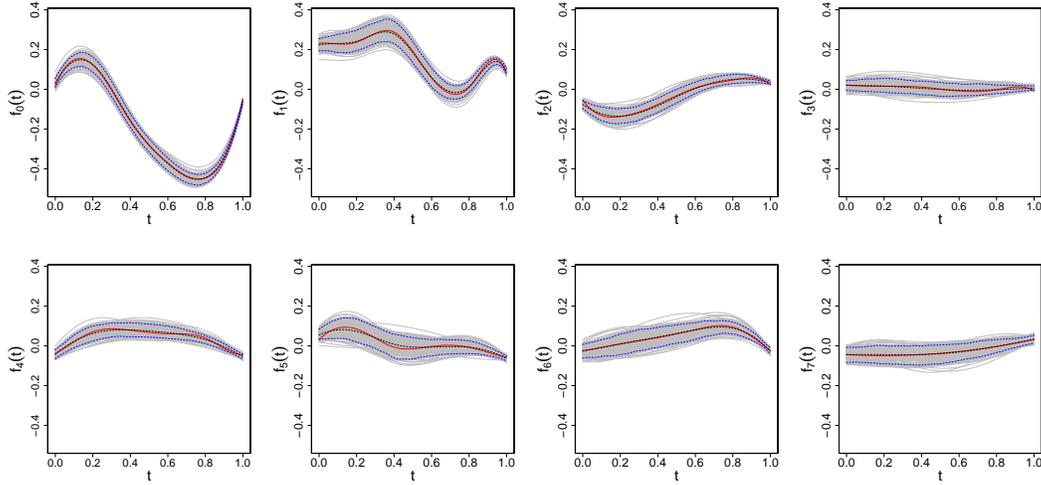


Figure C.6: True and estimated covariate and interaction effects estimated using the independence assumption. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

Table C.5: Simultaneous coverage of the point-wise CBs for FPC-FAMM and for spline-FAMM. Shown is the average proportion of completely covered curves for all covariate and interaction effects. For FPC-FAMM, the coverage refers to 200 simulation runs, whereas for splines-FAMM, 100 simulation runs are taken into account.

	$f_0(t)$	$f_1(t)$	$f_2(t)$	$f_3(t)$	$f_4(t)$	$f_5(t)$	$f_6(t)$	$f_7(t)$
$\mu(t, \mathbf{x}_{ijh})_{\text{FPC-FAMM}}$	43.50%	71.50%	70.50%	64.50%	76.00%	70.00%	66.00%	78.50%
$\mu(t, \mathbf{x}_{ijh})_{\text{spline-FAMM}}$	1.00%	5.00%	1.00%	1.00%	2.00%	2.00%	2.00%	5.00%

Table C.6: rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSEs for the fRI and the smooth error. Last row: Average rrMSEs for the functional response, the mean, and the error variance.

X	L^X	K^X	ϕ_1^X	ϕ_2^X	ϕ_3^X	ν_1^X	ν_2^X	ν_3^X	ξ_1^X	ξ_2^X	ξ_3^X	X	$X_{\text{FPC-FAMM}}$	$X_{\text{spline-FAMM}}$	σ^2
B	9	0.23	0.21	0.22		0.15	0.21		0.22	0.30		0.17	0.17	1.17	
E	707	0.03	0.02	0.02	0.02	0.02	0.03	0.03	0.15	0.19	0.23	0.17	0.17	0.50	
Y												0.09	0.09	0.15	0.10

Table C.7: Average rrMSEs for the estimated mean and covariate effects. Rows 1-2: rrMSEs for the estimation based on the independence assumption and for the estimation using FPC-FAMM averaged over 200 simulation runs. Last row: rrMSEs for the spline-based alternative averaged over 100 simulation runs.

	$f_0(t)$	$f_1(t)$	$f_2(t)$	$f_3(t)$	$f_4(t)$	$f_5(t)$	$f_6(t)$	$f_7(t)$
$\mu(t, \mathbf{x}_{ijh})$	0.06	0.13	0.22	1.43	0.30	0.58	0.39	0.60
$\mu(t, \mathbf{x}_{ijh})_{\text{FPC-FAMM}}$	0.06	0.12	0.20	1.34	0.25	0.53	0.39	0.47
$\mu(t, \mathbf{x}_{ijh})_{\text{spline-FAMM}}$	0.36	0.38	0.58	3.83	0.84	1.54	1.10	1.85

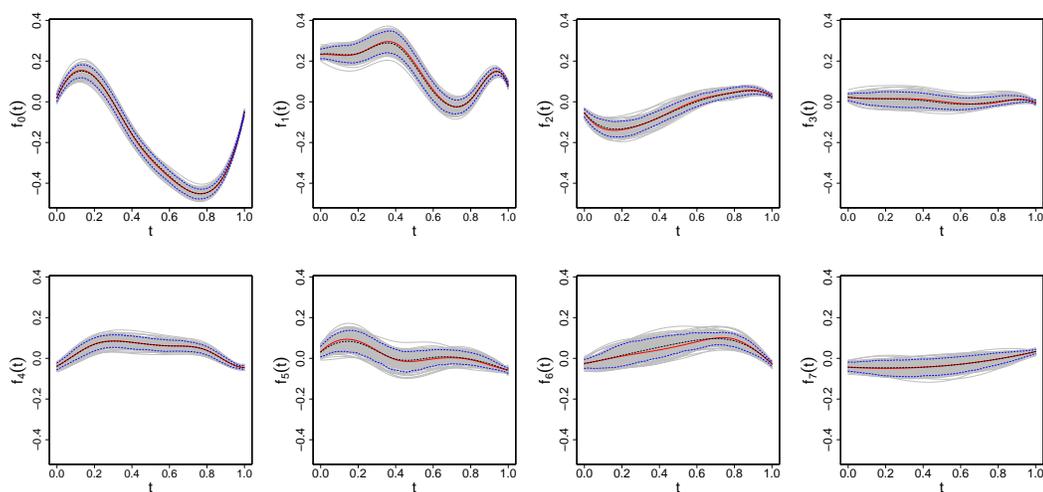


Figure C.7: True and estimated covariate and interaction effects estimated using FPC-FAMM. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

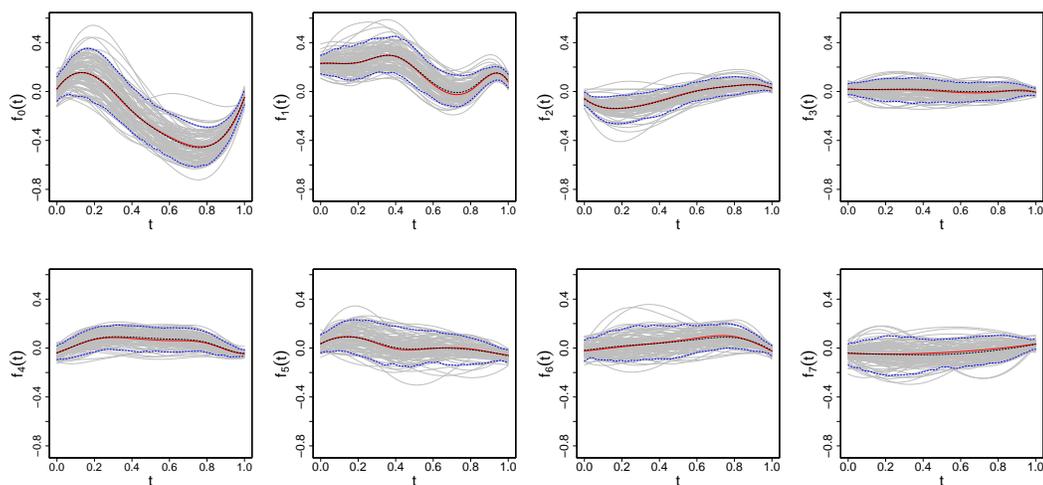


Figure C.8: True and estimated covariate and interaction effects estimated using spline-FAMM. Shown are the true function (red), the mean of the estimated functions over 100 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 100 simulation runs (grey).

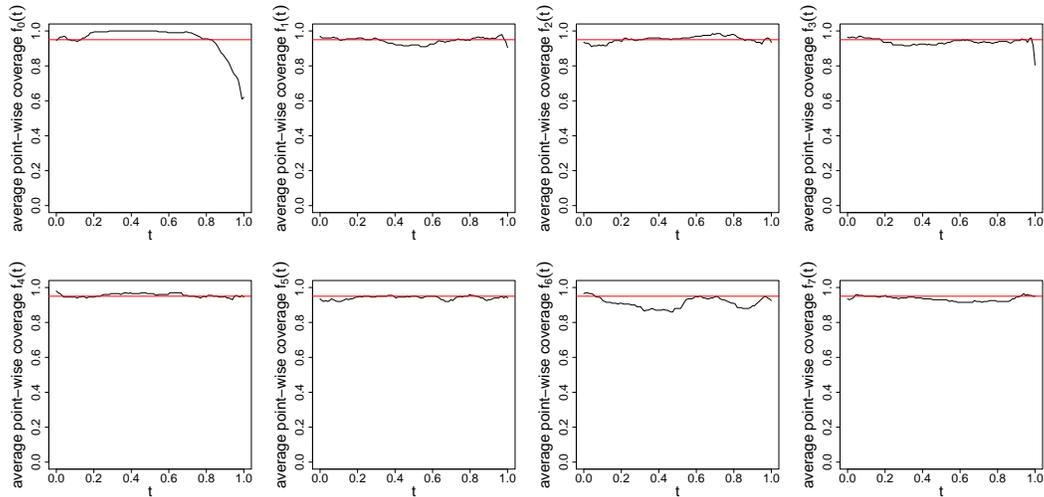


Figure C.9: Average point-wise coverage of the point-wise CBs obtained by FPC-FAMM for all covariate and interaction effects. For each effect, the point-wise coverage averaged over 200 simulation runs (black line) is shown. The red line indicates the nominal value of 0.95.

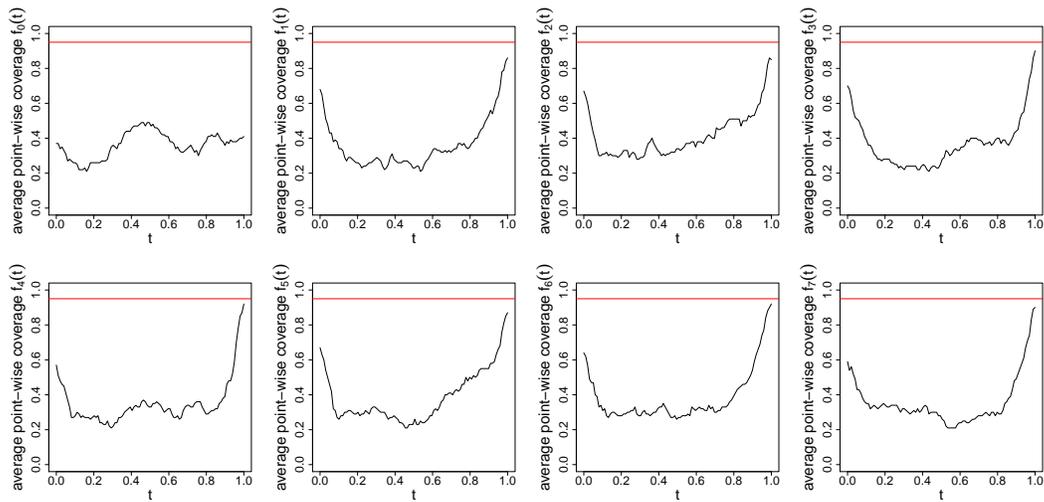


Figure C.10: Average point-wise coverage of the point-wise CBs obtained by spline-FAMM for all covariate and interaction effects. For each effect, the point-wise coverage averaged over 100 simulation runs (black line) is shown. The red line indicates the nominal value of 0.95.

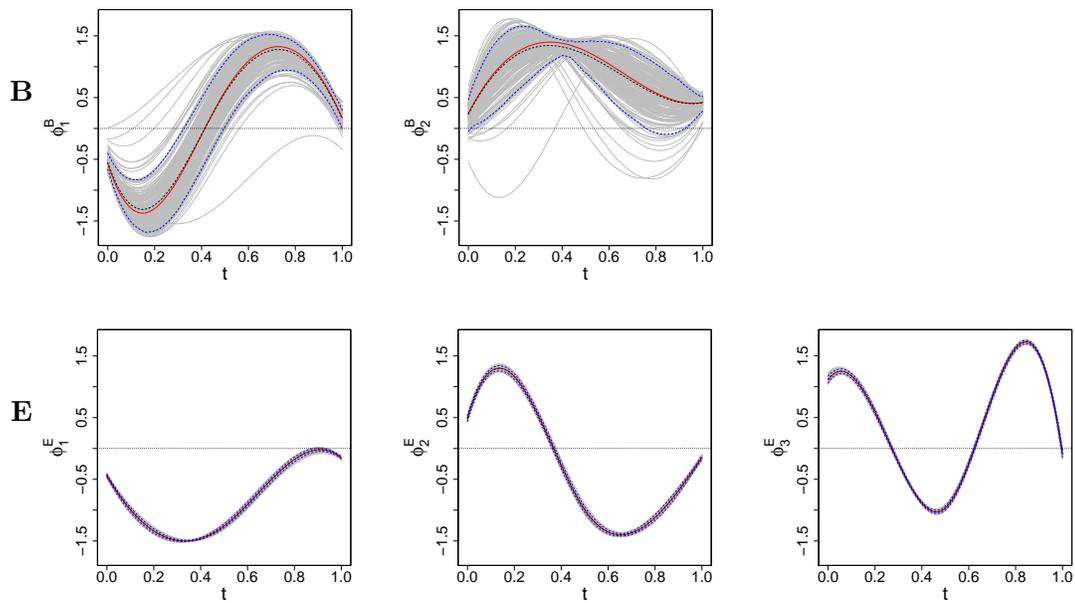


Figure C.11: True and estimated FPCs of the fRI (top row) and of the smooth error (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

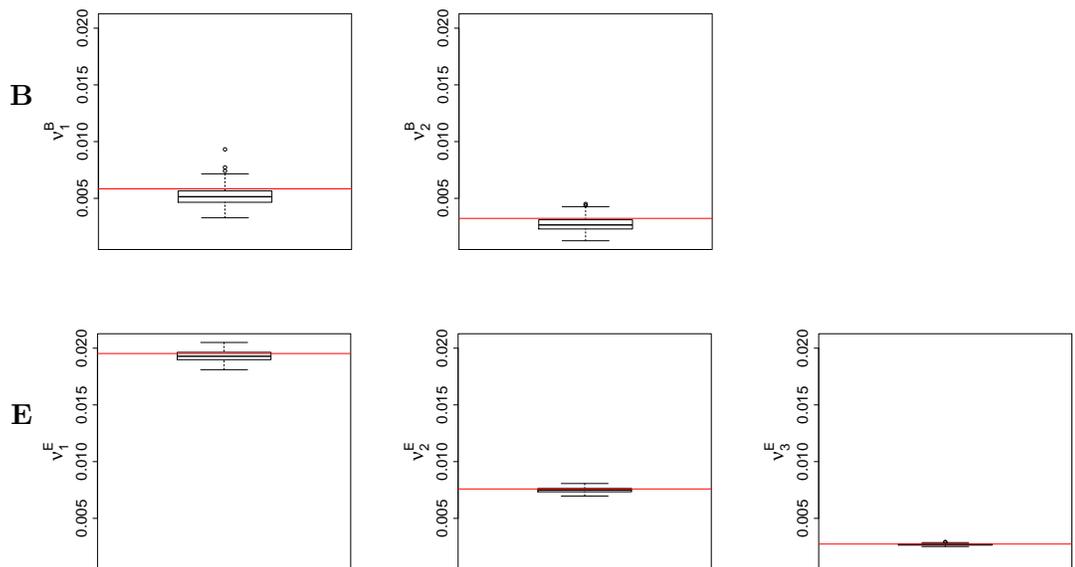


Figure C.12: Boxplots of the estimated eigenvalues of the auto-covariances of the fRI (top row), and of the smooth error (bottom row) for all 200 simulations runs.

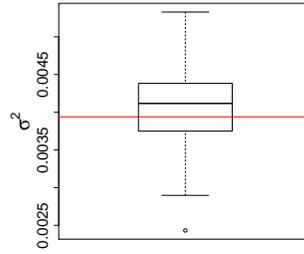


Figure C.13: Boxplot of the estimated error variances σ^2 for all 200 simulation runs.

Simulation results for the sparse scenario

In the following, we show additional results for the sparse scenario with centered and decorrelated basis weights. Figure C.14 shows the true and estimated mean functions, Figure C.15 shows the true and estimated FPCs, and Figure C.16 depicts the boxplots of the estimated eigenvalues for the two fRIs and for the smooth error. Figure C.17 shows the boxplot of the estimated error variances. In Table C.8, the average relative errors for all model components are given.

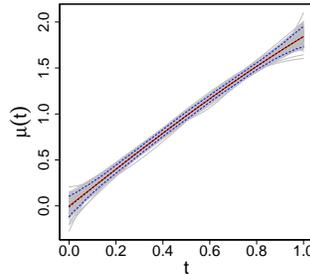


Figure C.14: True and estimated mean function $\mu(t, \mathbf{x}_{ijh})$. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

Table C.8: rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average relative errors for the functional random effects and their covariance decompositions. Last row: Average rrMSE of the functional response, the mean, and the error variance.

X	L^X	K^X	ϕ_1^X	ϕ_2^X	ν_1^X	ν_2^X	ξ_1^X	ξ_2^X	X	μ	σ^2
B	40	0.06	0.05	0.07	0.02	0.04	0.04	0.11	0.06		
C	40	0.06	0.07	0.11	0.03	0.05	0.23	0.25	0.21		
E	4800	0.14	0.11	0.07	0.02	0.05	0.30	0.19	0.29		
Y									0.09	0.03	1.81

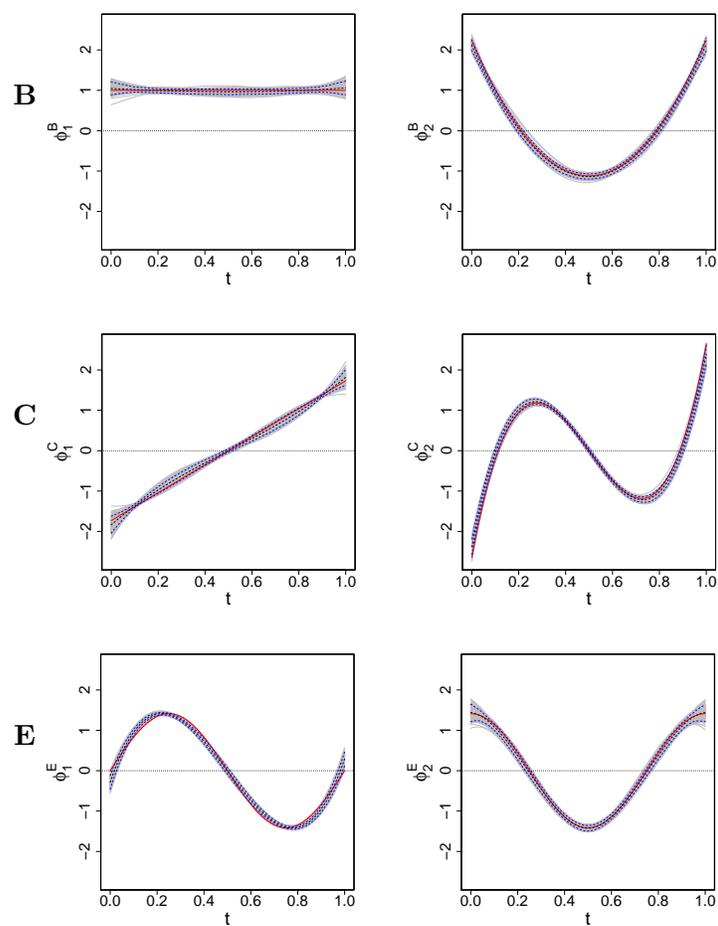


Figure C.15: True and estimated FPCs of the crossed fRIs $B_i(t)$ (top row) and $C_j(t)$ (middle row), as well as the FPCs of the smooth error $E_{ijh}(t)$ (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

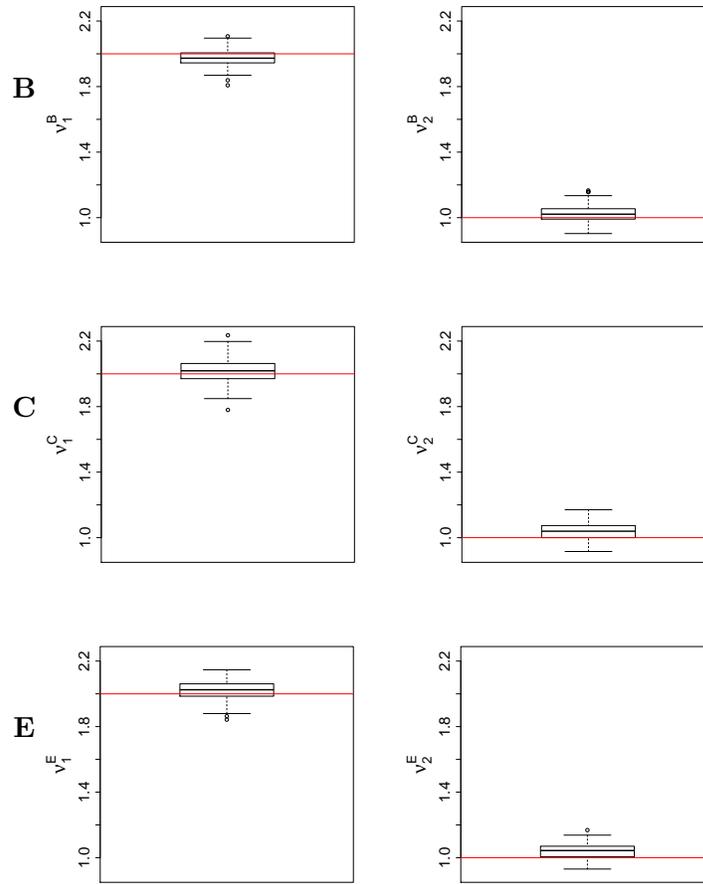


Figure C.16: Boxplots of the estimated eigenvalues of the auto-covariances of the crossed fRIs $B_i(t)$ (top row), $C_j(t)$ (middle row), as well as the eigenvalues of the auto-covariance of the smooth error $E_{ijh}(t)$ (bottom row) for all 200 simulation runs.

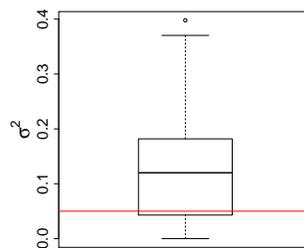


Figure C.17: Boxplot of the estimated error variances σ^2 for all 200 simulation runs.

C.4.4 Results for simulations with original basis weights

Previous to the presentation of the results for the simulation with the original (non-centered and non-decorrelated) basis weights, a short summary and a comparison with the results for the simulations with centered and decorrelated basis weights is given.

It shows that with the original basis weights, the estimated FPCs are often permuted within one grouping factor, e.g., the first and second FPC of the speakers are interchanged or are linear combinations of them, due to the correlation in the basis weights. As expected, this effect increases the smaller the corresponding number of independent levels, as the empirical FPC weights then have an empirical distribution far from the theoretical one. Moreover, a higher average rrMSEs for the eigenvalues is obtained when using the original basis weights. Correspondingly, worse results are obtained for the auto-covariances. For all three simulation settings, the rrMSEs for the basis weights tend to be higher with the original basis weights. Using the original basis weights results in higher rrMSEs for the functional random effects, especially for functional random effects with a small number of grouping levels. The estimation of the mean function for non-centered basis weights is much worse due to a shift of the mean by the respective FPC multiplied by the empirical mean of the basis weights. Correspondingly, also the coverage of the point-wise CBs decreases for most points. Yet, the functional response for the original basis weights is again estimated very well as shifts between mean and non-centered random effects cancel out. The covariate effects and the coverage of the CBs shows hardly any difference to the results for the simulations with centered and decorrelated basis weights. Also for the error variance no considerable change in the rrMSEs is found.

Simulation results for crossed-fRIs scenario

In the following, we show the results for the simulations of the application-based crossed-fRIs scenario with non-centered and non-decorrelated basis weights. Figure C.18 shows the true and estimated mean functions, Figure C.19 shows the true and estimated FPCs, and Figure C.20 depicts the boxplots of the estimated eigenvalues for the two fRIs and for the smooth error. Figure C.21 shows the boxplot of the estimated error variances. In Table C.9, the average rrMSEs for all model components are given.

Table C.9: rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSE for the functional random effects. Last row: Average rrMSE for the functional response, the mean, and the error variance.

X	L^X	K^X	ϕ_1^X	ϕ_2^X	ϕ_3^X	ν_1^X	ν_2^X	ν_3^X	ξ_1^X	ξ_2^X	ξ_3^X	X	μ	σ^2
B	9	0.55	0.37	0.46		0.38	0.53		0.41	0.70		0.38		
C	16	0.32	0.05			0.31			0.24			0.25		
E	707	0.09	0.04	0.05	0.04	0.06	0.09	0.05	0.19	0.21	0.27	0.20		
Y												0.10	0.10	0.10

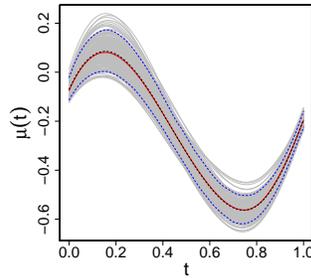


Figure C.18: True and estimated mean function $\mu(t, \mathbf{x}_{ijh})$. Shown is the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

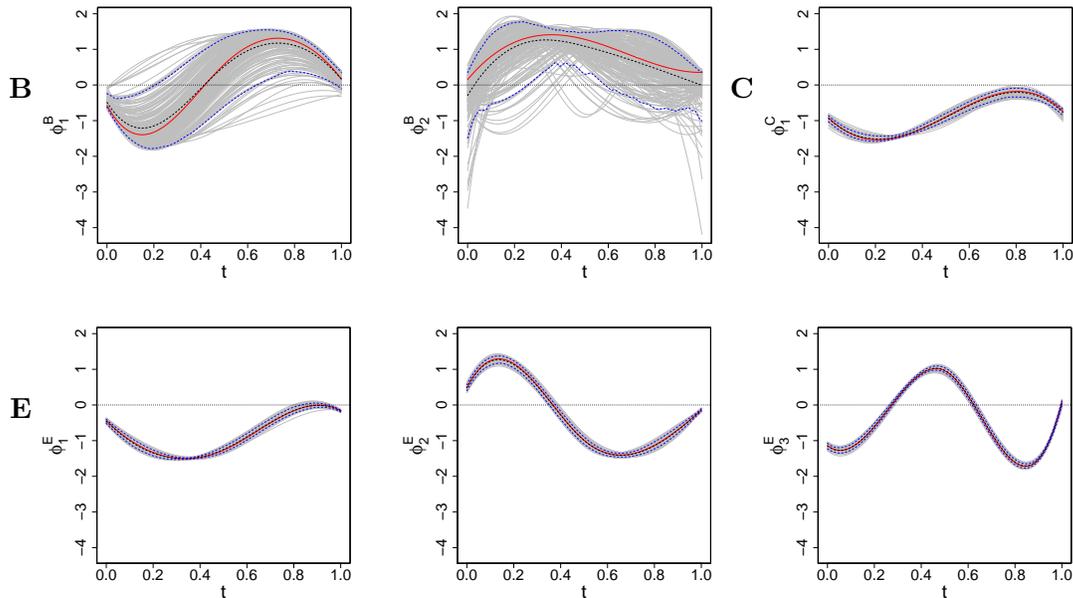


Figure C.19: True and estimated FPCs of the crossed fRIs $B_i(t)$ and $C_j(t)$ (top row), as well as the FPCs of the smooth error $E_{ijh}(t)$ (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

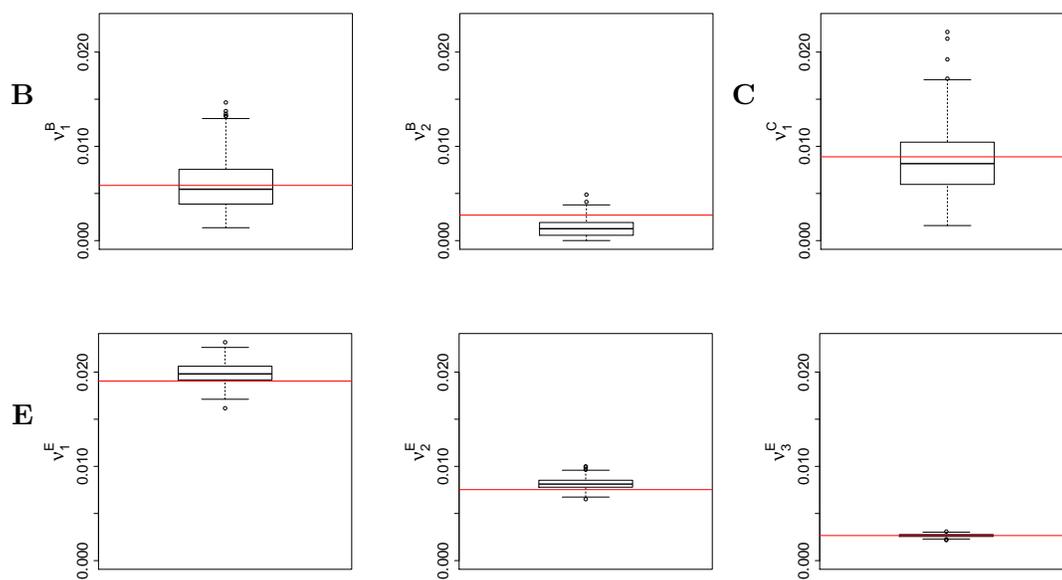


Figure C.20: Boxplots of the estimated eigenvalues of the auto-covariances of the crossed fRIs $B_i(t)$ and $C_j(t)$ (top row), as well as the eigenvalues of the auto-covariance of the smooth error $E_{ijh}(t)$ (bottom row) for all 200 simulation runs.

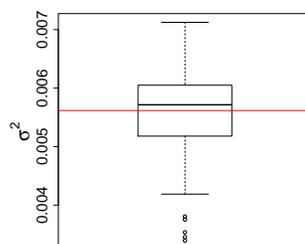


Figure C.21: Boxplot of the estimated error variances σ^2 for all 200 simulation runs.

Simulation results for the fRI scenario

In the following, we show additional simulation results for the simulations of the application-based fRI scenario with non-centered and non-decorrelated basis weights. Figure C.22 and Figure C.23 show the true and estimated covariate and interaction effects estimated based on the independence assumption, and on FPC-FAMM, respectively. The performance of the point-wise CBs is evaluated by looking at the point-wise coverage shown in Figure C.24. Table C.10 gives the coverage averaged over all time points and simulation runs for FPC-FAMM and spline-FAMM estimation. In addition, the simultaneous coverage of the point-wise CBs in terms of percentage of completely covered curves is given in Table C.11.

Figure C.25 depicts the true and estimated FPCs of the fRI and of the smooth error and Figure C.26 shows the boxplots of the estimated eigenvalues. The boxplot of the estimated error variances is shown in Figure C.27. Table C.12 lists the average rrMSEs for all model components except for the covariate effects and Table C.13 lists the rrMSEs for the covariate effects.

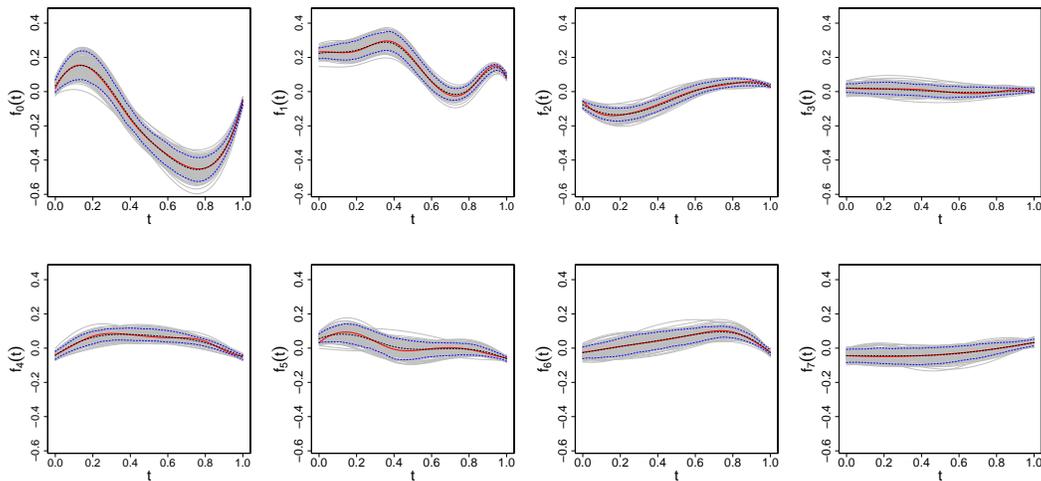


Figure C.22: True and estimated covariate and interaction effects estimated based on the independence assumption. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

Table C.10: Point-wise coverage of the point-wise CBs for FPC-FAMM. Shown is the coverage averaged over all time points for all covariate and interaction effects. The coverage refers to 200 simulation runs.

	$f_0(t)$	$f_1(t)$	$f_2(t)$	$f_3(t)$	$f_4(t)$	$f_5(t)$	$f_6(t)$	$f_7(t)$
$\mu(t, \mathbf{x}_{ijh})_{\text{FPC-FAMM}}$	70.48%	94.45%	94.94%	93.86%	95.23%	93.88%	91.50%	93.62%

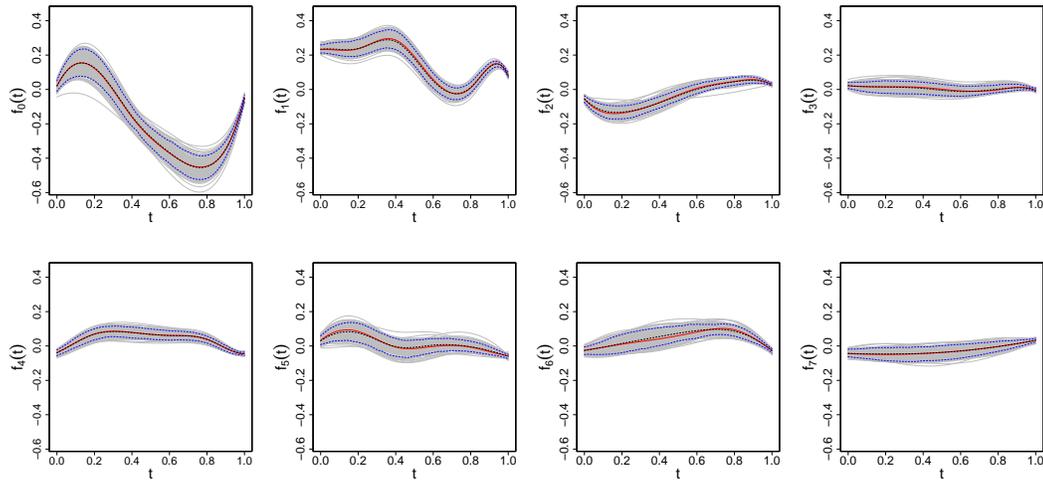


Figure C.23: True and estimated covariate and interaction effects using FPC-FAMM. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

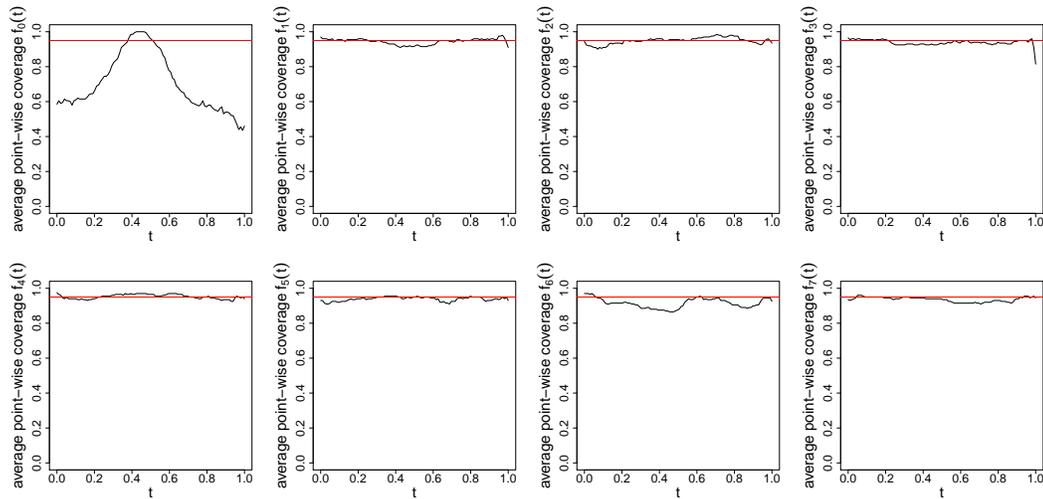


Figure C.24: Average point-wise coverage of the point-wise CBs obtained by FPC-FAMM for all covariate and interaction effects. For each effect, the point-wise coverage averaged over 200 simulation runs (black line) is shown. The red line indicates the nominal value of 0.95.

Table C.11: Simultaneous coverage of the point-wise CBs obtained by FPC-FAMM. Shown is the proportion of completely covered curves for all covariate and interaction effects. The coverage refers to 200 simulation runs.

	$f_0(t)$	$f_1(t)$	$f_2(t)$	$f_3(t)$	$f_4(t)$	$f_5(t)$	$f_6(t)$	$f_7(t)$
$\mu(t, \mathbf{x}_{ijh})_{\text{FPC-FAMM}}$	11.00%	71.00%	71.50%	65.50%	74.50%	69.00%	67.00%	78.00%

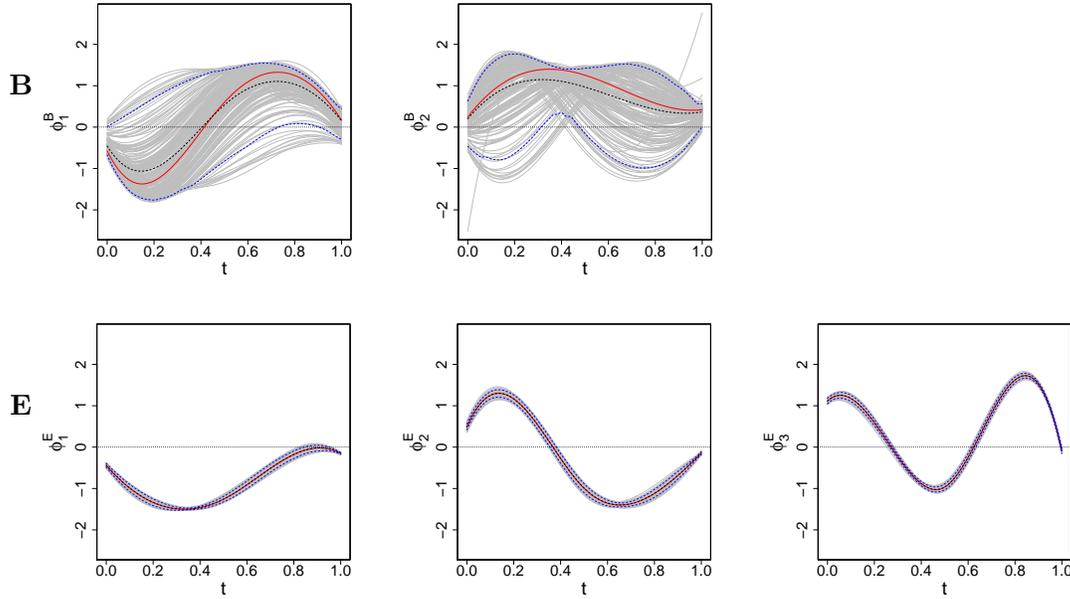


Figure C.25: True and estimated FPCs of the fRI $B_i(t)$ (top row) and of the smooth error (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

Table C.12: rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSEs for the fRI and the smooth error. Last row: Average rrMSEs for the functional response, the mean, and the error variance.

X	L^X	K^X	ϕ_1^X	ϕ_2^X	ϕ_3^X	ν_1^X	ν_2^X	ν_3^X	ξ_1^X	ξ_2^X	ξ_3^X	X	$X_{\text{FPC-FAMM}}$	σ^2
B	9	0.51	0.49	0.51		0.31	0.44		0.50	0.74		0.35	0.35	
E	707	0.07	0.04	0.05	0.04	0.05	0.05	0.05	0.16	0.20	0.25	0.18	0.17	
Y												0.09	0.09	0.11

Table C.13: Average rrMSEs for the estimated mean and covariate effects for the estimation using the independence assumption (first row) and using FPC-FAMM (second row) averaged over 200 simulation runs.

	$f_0(t)$	$f_1(t)$	$f_2(t)$	$f_3(t)$	$f_4(t)$	$f_5(t)$	$f_6(t)$	$f_7(t)$
$\mu(t\mathbf{x}_{ijh})$	0.13	0.13	0.22	1.43	0.30	0.58	0.39	0.59
$\mu(t\mathbf{x}_{ijh})_{\text{FPC-FAMM}}$	0.12	0.12	0.21	1.34	0.25	0.53	0.38	0.47

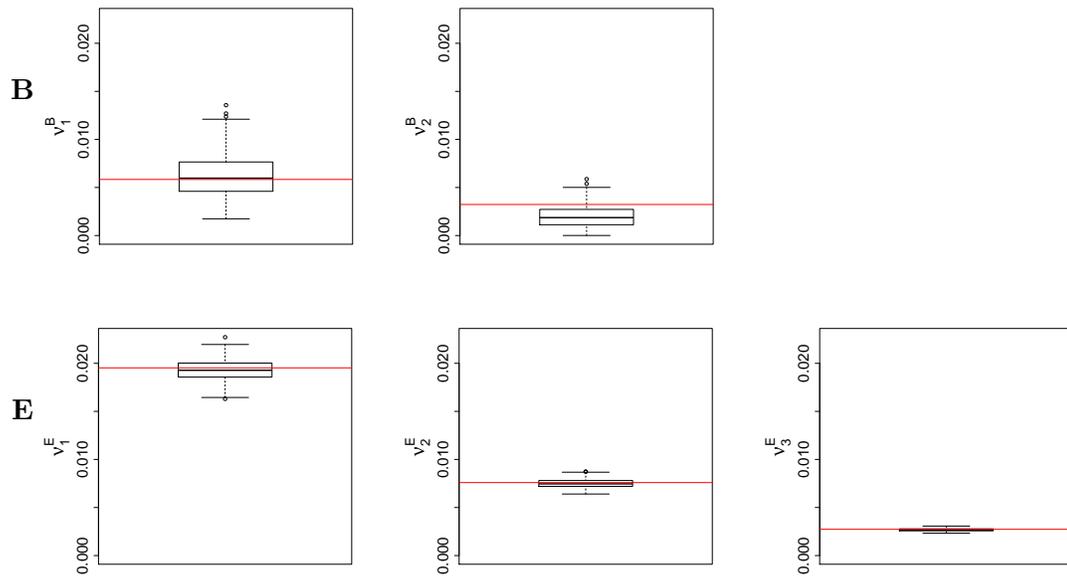


Figure C.26: Boxplots of the estimated eigenvalues of the auto-covariances of the fRI $B_i(t)$ (top row), and of the smooth error (bottom row) for all 200 simulation runs.

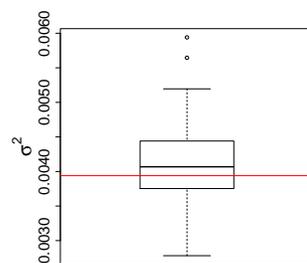


Figure C.27: Boxplot of the estimated error variances σ^2 for all 200 simulation runs.

Simulation results for the sparse scenario

In the following, additional results for the simulations of the sparse scenario with non-centered and non-decorrelated basis weights are shown. Figure C.28 shows the true and estimated mean functions, Figure C.29 shows the true and estimated FPCs, and Figure C.30 depicts the boxplots of the estimated eigenvalues for the two fRIs and for the smooth error. Figure C.31 shows the boxplot of the estimated error variances. In Table C.14, the average rrMSEs for all model components are given.

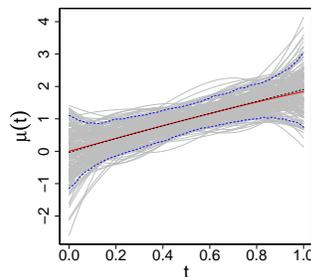


Figure C.28: True and estimated mean function $\mu(t, \mathbf{x}_{ijh})$. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

Table C.14: rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSEs for the random processes. Last row: Average rrMSEs for the functional response, the mean, and the error variance.

X	L^X	K^X	ϕ_1^X	ϕ_2^X	ν_1^X	ν_2^X	ξ_1^X	ξ_2^X	X	μ	σ^2
B	40	0.25	0.22	0.22	0.17	0.18	0.21	0.36	0.15		
C	40	0.24	0.24	0.26	0.17	0.20	0.36	0.48	0.28		
E	4800	0.14	0.11	0.07	0.03	0.05	0.30	0.20	0.30		
Y									0.09	0.32	1.76

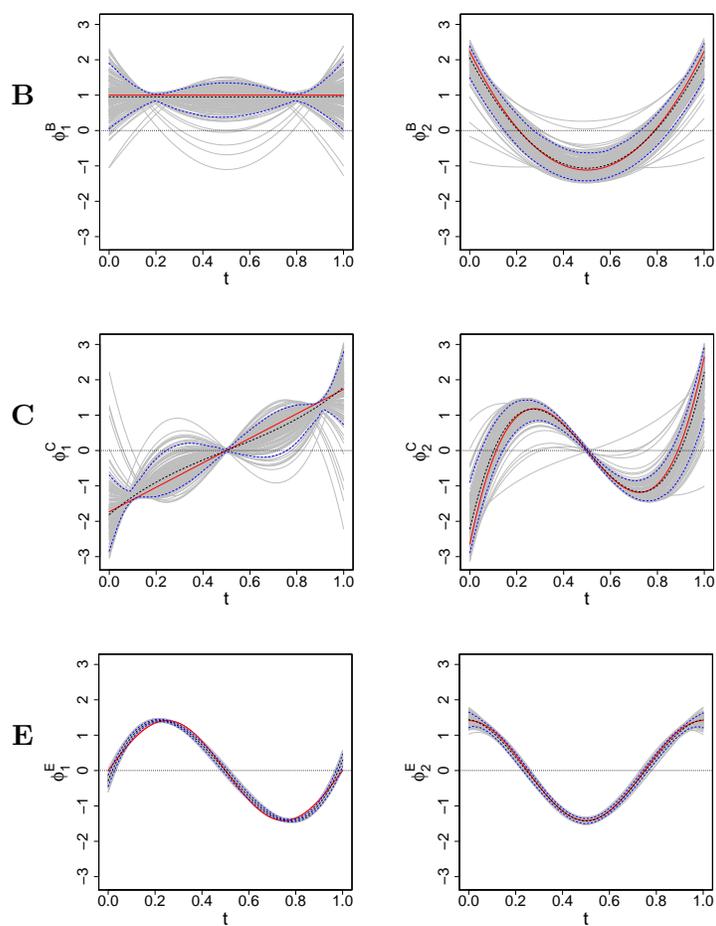


Figure C.29: True and estimated FPCs of the crossed fRIs $B_i(t)$ (top row) and $C_j(t)$ (middle row), as well as the FPCs of the smooth error $E_{ijh}(t)$ (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).

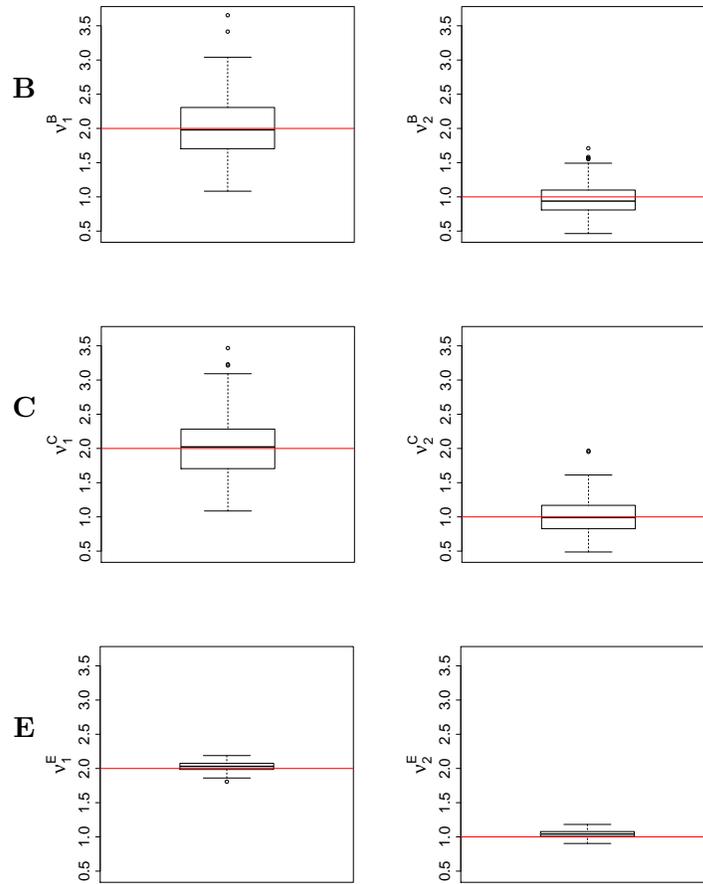


Figure C.30: Boxplots of the estimated eigenvalues of the auto-covariances of the crossed fRIs $B_i(t)$ (top row), $C_j(t)$ (middle row), as well as the eigenvalues of the auto-covariance of the smooth error $E_{ijh}(t)$ (bottom row) for all 200 simulation runs.

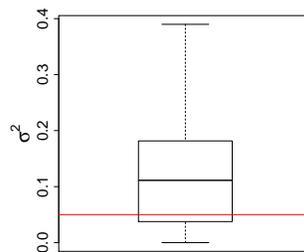


Figure C.31: Boxplot of the estimated error variances σ^2 for all 200 simulation runs.

Appendix D

Appendix of Chapter 5

Appendix D is based on the appendix of the following paper:

Cederbaum, J., Scheipl, F., and Greven, S. (2018): Fast symmetric additive covariance smoothing. *Computational Statistics & Data Analysis*, 120:25–41.

This appendix is divided into four main parts. The first part provides the derivation of the covariance of the products of the centered functional responses for the general FLMM and shows how it simplifies for the FLMM with crossed fRIs. The second part provides supplementary details on the estimation and the implementation in the R add-on package `sparseFLMM`. For both, the model with independent curves and the general FLMM, the concrete forms of the (additive) varying coefficient model using tensor product B-splines are shown. The third part shows the application of our approach to CD4 cell count data and gives additional results for the application to the speech production data. In the last part, the concrete forms of the measures of goodness of fit are provided and additional simulation results for both simulation scenarios are presented.

D.1 Derivations

D.1.1 Derivation for the covariance of the products of the centered functional responses in the general FLMM

Let $\mathbf{X} = (X_1, \dots, X_p)^\top$ be a p -dimensional Gaussian random variable with zero mean and covariance Σ . Then, given $p \geq 4$, we can express the fourth moment of \mathbf{X} based on Isserlis' theorem (Isserlis, 1918) as

$$\mathbb{E}(X_i X_j X_k X_l) = \Sigma_{ij} \Sigma_{kl} + \Sigma_{ik} \Sigma_{jl} + \Sigma_{il} \Sigma_{jk}, \quad (\text{D.1})$$

where Σ_{ij} is the covariance of X_i and X_j .

Consider the covariance of the products of the centered functional responses in the general FLMM

$$\begin{aligned}
& \text{Cov} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_{i'}(t_{i'j'}), \tilde{Y}_m(t_{mo})\tilde{Y}_{m'}(t_{m'o'}) \right] \tag{D.2} \\
&= \mathbb{E} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_{i'}(t_{i'j'})\tilde{Y}_m(t_{mo})\tilde{Y}_{m'}(t_{m'o'}) \right] - \underbrace{\mathbb{E} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_{i'}(t_{i'j'}) \right]}_{\text{Cov}[\tilde{Y}_i(t_{ij}), \tilde{Y}_{i'}(t_{i'j'})]} \underbrace{\mathbb{E} \left[\tilde{Y}_m(t_{mo})\tilde{Y}_{m'}(t_{m'o'}) \right]}_{\text{Cov}[\tilde{Y}_m(t_{mo}), \tilde{Y}_{m'}(t_{m'o'})]} \\
&\stackrel{(D.1)}{=} \text{Cov} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_{i'}(t_{i'j'}) \right] \text{Cov} \left[\tilde{Y}_m(t_{mo}), \tilde{Y}_{m'}(t_{m'o'}) \right] \\
&+ \text{Cov} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_m(t_{mo}) \right] \text{Cov} \left[\tilde{Y}_{i'}(t_{i'j'}), \tilde{Y}_{m'}(t_{m'o'}) \right] \\
&+ \text{Cov} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_{m'}(t_{m'o'}) \right] \text{Cov} \left[\tilde{Y}_{i'}(t_{i'j'}), \tilde{Y}_m(t_{mo}) \right] \\
&- \text{Cov} \left[\tilde{Y}_i(t_{ij}), \tilde{Y}_{i'}(t_{i'j'}) \right] \text{Cov} \left[\tilde{Y}_m(t_{mo}), \tilde{Y}_{m'}(t_{m'o'}) \right] \\
&= \text{Cov} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_m(t_{mo}) \right] \text{Cov} \left[\tilde{Y}_{i'}(t_{i'j'}), \tilde{Y}_{m'}(t_{m'o'}) \right] \\
&+ \text{Cov} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_{m'}(t_{m'o'}) \right] \text{Cov} \left[\tilde{Y}_{i'}(t_{i'j'}), \tilde{Y}_m(t_{mo}) \right] \\
&= \left\{ \mathbf{z}_i^\top \mathbf{K}^U(t_{ij}, t_{mo}) \mathbf{z}_m + [K^E(t_{ij}, t_{mo}) + \sigma^2 \delta_{jo}] \delta_{im} \right\} \\
&\cdot \left\{ \mathbf{z}_{i'}^\top \mathbf{K}^U(t_{i'j'}, t_{m'o'}) \mathbf{z}_{m'} + [K^E(t_{i'j'}, t_{m'o'}) + \sigma^2 \delta_{j'o'}] \delta_{i'm'} \right\} \\
&+ \left\{ \mathbf{z}_i^\top \mathbf{K}^U(t_{ij}, t_{m'o'}) \mathbf{z}_{m'} + [K^E(t_{ij}, t_{m'o'}) + \sigma^2 \delta_{j'o'}] \delta_{im'} \right\} \\
&\cdot \left\{ \mathbf{z}_{i'}^\top \mathbf{K}^U(t_{i'j'}, t_{mo}) \mathbf{z}_m + [K^E(t_{i'j'}, t_{mo}) + \sigma^2 \delta_{j'o}] \delta_{i'm} \right\}. \quad \square
\end{aligned}$$

D.1.2 Simplification of the covariance of the products of the centered functional responses for crossed fRIs

For the special case of an FLMM with two crossed fRIs as in Section 5.3 of Chapter 5, the covariance of the products of the centered functional responses in Equation (D.2) simplifies to

$$\begin{aligned}
& \text{Cov} \left[\tilde{Y}_i(t_{ij})\tilde{Y}_{i'}(t_{i'j'}), \tilde{Y}_m(t_{mo})\tilde{Y}_{m'}(t_{m'o'}) \right] \\
&= \left\{ K^B(t_{ij}, t_{mo})\delta_{\ell_1(i)\ell_1(m)} + K^C(t_{ij}, t_{mo})\delta_{\ell_2(i)\ell_2(m)} + [K^E(t_{ij}, t_{mo}) + \sigma^2 \delta_{jo}] \delta_{im} \right\} \\
&\cdot \left\{ K^B(t_{i'j'}, t_{m'o'})\delta_{\ell_1(i')\ell_1(m')} + K^C(t_{i'j'}, t_{m'o'})\delta_{\ell_2(i')\ell_2(m')} + [K^E(t_{i'j'}, t_{m'o'}) + \sigma^2 \delta_{j'o'}] \delta_{i'm'} \right\} \\
&+ \left\{ K^B(t_{ij}, t_{m'o'})\delta_{\ell_1(i)\ell_1(m')} + K^C(t_{ij}, t_{m'o'})\delta_{\ell_2(i)\ell_2(m')} + [K^E(t_{ij}, t_{m'o'}) + \sigma^2 \delta_{j'o'}] \delta_{im'} \right\} \\
&\cdot \left\{ K^B(t_{i'j'}, t_{mo})\delta_{\ell_1(i')\ell_1(m)} + K^C(t_{i'j'}, t_{mo})\delta_{\ell_2(i')\ell_2(m)} + [K^E(t_{i'j'}, t_{mo}) + \sigma^2 \delta_{j'o}] \delta_{i'm} \right\},
\end{aligned}$$

where $\delta_{\ell_1(i)\ell_1(m)}$ and $\delta_{\ell_2(i)\ell_2(m)}$ take value one when the two curves i and m belong to the same level of the respective grouping factor and zero otherwise.

D.2 Supplementary details on the estimation and implementation

D.2.1 (Additive) varying coefficient model using tensor product B-splines

Let \otimes and \cdot denote the Kronecker product and the Hadamard (point-wise) product, respectively.

Model with independent curves

Using tensor product B-splines yields the following form of Model (5.3)

$$\begin{aligned} \mathbb{E}[\mathbf{C}] &= \left[\left(\mathbf{B}_t^E \otimes \mathbf{1}_{F^E}^\top \right) \cdot \left(\mathbf{1}_{F^E}^\top \otimes \mathbf{B}_{t'}^E \right) \middle| \boldsymbol{\delta}^\varepsilon \right] \left(\boldsymbol{\theta}^{E^\top}, \sigma^2 \right)^\top \\ &= \left[\mathbf{M}^E \middle| \boldsymbol{\delta}^\varepsilon \right] \left(\boldsymbol{\theta}^{E^\top}, \sigma^2 \right)^\top = \mathbf{M}\boldsymbol{\alpha}, \end{aligned}$$

where \mathbf{B}_t^E , $\mathbf{B}_{t'}^E$ are the $\mathcal{C} \times F^E$ marginal spline design matrices that contain the evaluated spline basis functions for the directions t and t' , respectively. \mathbf{B}_t^E and $\mathbf{B}_{t'}^E$ contain identical but permuted rows. $\mathbf{1}_{F^E} = (1, \dots, 1)^\top$ is of length F^E , the number of marginal basis functions in each direction. The bivariate additive model for the reduced response vector is obtained by replacing the marginal spline design matrices by the reduced $\mathcal{C}^\Delta \times F^E$ matrices $\mathbf{B}_t^{E\Delta}$, $\mathbf{B}_{t'}^{E\Delta}$ and the index vector by the reduced index vector $\boldsymbol{\delta}^{\varepsilon\Delta}$. Note that in the model with independent curves, the bivariate spline design matrix $\mathbf{B}^{E\Delta}$ corresponds to the design matrix $\mathbf{M}^{E\Delta}$ as products are only computed on the same curves and thus the indicator matrix $\mathbf{Q}^{E\Delta}$ reduces to an all-ones matrix.

General FLMM

For each $g = 1, \dots, G$, and each $s, s' = 1, \dots, \rho^{U_g}$, let $\mathbf{B}_{ss',t}^{U_g\Delta}$ and $\mathbf{B}_{ss',t'}^{U_g\Delta}$ denote the marginal spline design matrices of dimensions $\mathcal{C}^\Delta \times F_{ss',t}^{U_g}$ and $\mathcal{C}^\Delta \times F_{ss',t'}^{U_g}$, respectively. Due to the symmetry of the covariances $\mathbf{K}^{U_g}(t, t')$, we assumed $F_{ss',t}^{U_g} = F_{s',s,t}^{U_g}$ and $F_{ss',t'}^{U_g} = F_{s',s,t'}^{U_g}$ respectively. Then, the bivariate spline design matrices $\mathbf{B}_{ss'}^{U_g\Delta}$ are given by

$$\mathbf{B}_{ss'}^{U_g\Delta} = \left(\mathbf{B}_{ss',t}^{U_g\Delta} \otimes \mathbf{1}_{F_{ss',t'}^{U_g}}^\top \right) \cdot \left(\mathbf{1}_{F_{ss',t}^{U_g}}^\top \otimes \mathbf{B}_{ss',t'}^{U_g\Delta} \right).$$

The submatrices $\mathbf{M}_{ss'}^{U_g\Delta}$ corresponding to the covariances $K_{ss'}^{U_g}(t, t')$ are given as $\mathbf{M}_{ss'}^{U_g\Delta} = \mathbf{Q}_{ss'}^{U_g\Delta} \cdot \mathbf{B}_{ss'}^{U_g\Delta}$, where $\mathbf{Q}_{ss'}^{U_g\Delta}$ are $\mathcal{C}^\Delta \times F_{ss',t}^{U_g} F_{ss',t'}^{U_g}$ matrices with entries $\delta_{\ell_g(i)\ell_g(i')} \cdot \omega_{is}^{U_g} \omega_{i's'}^{U_g}$. The indicators $\delta_{\ell_g(i)\ell_g(i')}$ take value one if the two curves i and i' are of the same level of grouping factor g and zero otherwise. An example for how these indicators result from Model (5.9) is given below. The columns of $\mathbf{Q}_{ss'}^{U_g\Delta}$ are all identical and contain the suitably sorted and repeated entries. Suitably sorted and repeated in this context means that the sorting corresponds to the sorting in \mathcal{C}^Δ and that the entries $\delta_{\ell_g(i)\ell_g(i')} \cdot \omega_{is}^{U_g} \omega_{i's'}^{U_g}$ are repeated for all considered combinations of observation points $t_{ij} \leq t_{i'j'}$ if $i \leq i'$ and $t_{ij} < t_{i'j'}$, otherwise.

In analogy, matrix $\mathbf{Q}^{E\Delta}$ is a $\mathcal{C}^\Delta \times (F^E)^2$ matrix with identical columns consisting of suitably sorted and repeated indicators, which take value one if the two points in the products of the centered

functional responses belong to the same curve and zero otherwise.

Example for the indicators $\delta_{\ell_g(i)\ell_g(i')}$

Consider for simplicity the case of one grouping factor ($G = 1$) for which two levels exist ($L^{U_1} = 2$). Further assume that for each level of this grouping factor, a functional random intercept and a functional random slope in variable ω are fitted ($\rho^{U_1} = 2$). Then, Model (5.9) can be reformed to

$$\begin{aligned} \mathbb{E} \left[\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'}) \right] &= \sum_{s=1}^2 \sum_{s'=1}^2 \left[z_{i1s}^{U_1} z_{i'1s'}^{U_1} K_{ss'}^{U_1}(t_{ij}, t_{i'j'}) + z_{i2s}^{U_1} z_{i'2s'}^{U_1} K_{ss'}^{U_1}(t_{ij}, t_{i'j'}) \right] \\ &+ \left[K^E(t_{ij}, t_{i'j'}) + \sigma^2 \delta_{jj'} \right] \delta_{ii'} \\ &= z_{i11}^{U_1} z_{i'11}^{U_1} K_{11}^{U_1}(t_{ij}, t_{i'j'}) + z_{i21}^{U_1} z_{i'21}^{U_1} K_{11}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ z_{i11}^{U_1} z_{i'12}^{U_1} K_{12}^{U_1}(t_{ij}, t_{i'j'}) + z_{i21}^{U_1} z_{i'22}^{U_1} K_{12}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ z_{i12}^{U_1} z_{i'11}^{U_1} K_{21}^{U_1}(t_{ij}, t_{i'j'}) + z_{i22}^{U_1} z_{i'21}^{U_1} K_{21}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ z_{i12}^{U_1} z_{i'12}^{U_1} K_{22}^{U_1}(t_{ij}, t_{i'j'}) + z_{i22}^{U_1} z_{i'22}^{U_1} K_{22}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ \left[K^E(t_{ij}, t_{i'j'}) + \sigma^2 \delta_{jj'} \right] \delta_{ii'}. \end{aligned}$$

The $z_{ils}^{U_1}$ are defined as in Chapter 5 as $z_{ils}^{U_1} = \omega_{is}^{U_1} \delta_{\ell_1(i)l}$, where $\delta_{\ell_1(i)l}$ is an indicator that takes value one if curve i belongs to level l of the grouping factor and zero otherwise. With $\omega_{i1}^{U_1} = \omega_{i'1}^{U_1} = 1$ (functional random intercepts), we can write

$$\begin{aligned} \mathbb{E} \left[\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'}) \right] &= \delta_{\ell_1(i)1} \delta_{\ell_1(i')1} K_{11}^{U_1}(t_{ij}, t_{i'j'}) + \delta_{\ell_1(i)2} \delta_{\ell_1(i')2} K_{11}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ \delta_{\ell_1(i)1} \omega_{i'2}^{U_1} \delta_{\ell_1(i')1} K_{12}^{U_1}(t_{ij}, t_{i'j'}) + \delta_{\ell_1(i)2} \omega_{i'2}^{U_1} \delta_{\ell_1(i')2} K_{12}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ \omega_{i2}^{U_1} \delta_{\ell_1(i)1} \delta_{\ell_1(i')1} K_{21}^{U_1}(t_{ij}, t_{i'j'}) + \omega_{i2}^{U_1} \delta_{\ell_1(i)2} \delta_{\ell_1(i')2} K_{21}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ \omega_{i2}^{U_1} \delta_{\ell_1(i)1} \omega_{i'2}^{U_1} \delta_{\ell_1(i')1} K_{22}^{U_1}(t_{ij}, t_{i'j'}) + \omega_{i2}^{U_1} \delta_{\ell_1(i)2} \omega_{i'2}^{U_1} \delta_{\ell_1(i')2} K_{22}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ \left[K^E(t_{ij}, t_{i'j'}) + \sigma^2 \delta_{jj'} \right] \delta_{ii'}. \end{aligned}$$

Thus, when

1. both curves in the product $\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'})$ belong to the first level of the grouping factor (or both to the second), Model (5.9) reduces to

$$\begin{aligned} &K_{11}^{U_1}(t_{ij}, t_{i'j'}) + \omega_{i'2}^{U_1} K_{12}^{U_1}(t_{ij}, t_{i'j'}) + \omega_{i2}^{U_1} K_{21}^{U_1}(t_{ij}, t_{i'j'}) + \omega_{i2}^{U_1} \omega_{i'2}^{U_1} K_{22}^{U_1}(t_{ij}, t_{i'j'}) \\ &+ \left[K^E(t_{ij}, t_{i'j'}) + \sigma^2 \delta_{jj'} \right] \delta_{ii'}. \end{aligned}$$

2. the first curve in the product $\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'})$ belongs to the first level of the grouping factor and the second curve in the product belongs to the second level or vice versa, Model (5.9) takes value zero for this product.

This can be generalized to more grouping factors, meaning that the product $\tilde{Y}_i(t_{ij}) \tilde{Y}_{i'}(t_{i'j'})$ only plays a role in the covariance estimation if the two curves in the product are of the same level for at

least one grouping factor. Thus, the entries of $\mathbf{Q}_{ss'}^{U_g \Delta}$ in Model (5.10) are given as $\delta_{\ell_g(i)\ell_g(i')}\omega_{is}^{U_g}\omega_{i's'}^{U_g}$. Note that these entries do not equal the products $z_{ils}^{U_g}z_{i'ls'}^{U_g} = \delta_{\ell_g(i)l}\delta_{\ell_g(i')l}\omega_{is}^{U_g}\omega_{i's'}^{U_g}$ in Model (5.9), in which the sum is taken over the levels of the grouping factors.

D.2.2 Form of the constraint matrix

For each $g = 1, \dots, G$, the constraint matrix \mathbf{W}^{U_g} is a block matrix consisting of $(\rho^{U_g})^2 \times [(\rho^{U_g})^2 + 1] / 2$ blocks, most of which are zero. The rows and columns of \mathbf{W}^{U_g} are sorted as in matrix $\mathbf{M}^{U_g \Delta}$ and in the reduced matrix $\mathbf{M}^{U_g \Delta r}$, respectively. The non-zero blocks can be divided into two groups: blocks corresponding to the auto-covariances $K_{ss}^{U_g}(t, t')$, $s = 1, \dots, \rho^{U_g}$, and blocks corresponding to the cross-covariances $K_{ss'}^{U_g}(t, t')$, with $s < s'$.

Let $(F_{ss}^{U_g})^2$ denote the number of spline basis functions used for smoothing the auto-covariance $K_{ss}^{U_g}(t, t')$. The blocks for the auto-covariances are of the same form as \mathbf{W}^E and given by the $(F_{ss}^{U_g})^2 \times F_{ss}^{U_g} (F_{ss}^{U_g} + 1) / 2$ matrices

$$\begin{bmatrix} \mathbf{I}_{\frac{F_{ss}^{U_g}(F_{ss}^{U_g}-1)}{2}} & \mathbf{0}_{\frac{F_{ss}^{U_g}(F_{ss}^{U_g}-1)}{2} \times F_{ss}^{U_g}} \\ \mathbf{0}_{F_{ss}^{U_g} \times \frac{F_{ss}^{U_g}(F_{ss}^{U_g}-1)}{2}} & \mathbf{I}_{F_{ss}^{U_g}} \\ \mathbf{I}_{\frac{F_{ss}^{U_g}(F_{ss}^{U_g}-1)}{2}} & \mathbf{0}_{\frac{F_{ss}^{U_g}(F_{ss}^{U_g}-1)}{2} \times F_{ss}^{U_g}} \end{bmatrix},$$

where \mathbf{I}_x is an identity matrix of dimension x and $\mathbf{0}_{x \times y}$ is a null matrix of dimension $x \times y$.

Consider for simplicity the case of bivariate tensor product spline bases, where $F_{ss',t}^{U_g}$ and $F_{ss',t'}^{U_g}$ denote the number of marginal spline basis functions for smoothing the cross-covariance $K_{ss'}^{U_g}(t, t')$, $s < s'$, in direction t and t' , respectively. Due to the symmetry, we have $F_{ss',t}^{U_g} = F_{s's,t'}^{U_g}$ and $F_{ss',t'}^{U_g} = F_{s't,t}^{U_g}$. Let $F_{ss',b=b'}^{U_g}$ denote the number of coefficients on the diagonal in $\Theta_{ss'}^{U_g}$, which corresponds to the minimum of $F_{ss',t}^{U_g}$ and $F_{ss',t'}^{U_g}$ and denote the number of coefficients below and above the diagonal as $F_{ss',b < b'}^{U_g} := \sum_{i=1}^{F_{ss',b=b'}^{U_g}} (F_{ss',t'}^{U_g} - i)$ and $F_{ss',b > b'}^{U_g} := \sum_{i=1}^{F_{ss',b=b'}^{U_g}} (F_{ss',t}^{U_g} - i)$, respectively. The blocks for the cross-covariances are then $F_{ss',t}^{U_g} F_{ss',t'}^{U_g} \times F_{ss',t}^{U_g} F_{ss',t'}^{U_g}$ diagonal block matrices of the form

$$\begin{bmatrix} \mathbf{I}_{F_{ss',b < b'}^{U_g}} & \mathbf{0}_{F_{ss',b < b'}^{U_g} \times F_{ss',b=b'}^{U_g}} & \mathbf{0}_{F_{ss',b < b'}^{U_g} \times F_{ss',b > b'}^{U_g}} \\ \mathbf{0}_{F_{ss',b=b'}^{U_g} \times F_{ss',b < b'}^{U_g}} & \mathbf{I}_{F_{ss',b=b'}^{U_g}} & \mathbf{0}_{F_{ss',b=b'}^{U_g} \times F_{ss',b > b'}^{U_g}} \\ \mathbf{0}_{F_{ss',b > b'}^{U_g} \times F_{ss',b < b'}^{U_g}} & \mathbf{0}_{F_{ss',b > b'}^{U_g} \times F_{ss',b=b'}^{U_g}} & \mathbf{I}_{F_{ss',b > b'}^{U_g}} \end{bmatrix},$$

when the respective rows correspond to $s < s'$ and $F_{ss',t}^{U_g} F_{ss',t'}^{U_g} \times F_{ss',t}^{U_g} F_{ss',t'}^{U_g}$ anti-diagonal block matrices of the form

$$\begin{bmatrix} \mathbf{0}_{F_{ss',b < b'}^{U_g} \times F_{ss',b < b'}^{U_g}} & \mathbf{0}_{F_{ss',b < b'}^{U_g} \times F_{ss',b = b'}^{U_g}} & \mathbf{I}_{F_{ss',b > b'}^{U_g}} \\ \mathbf{0}_{F_{ss',b = b'}^{U_g} \times F_{ss',b < b'}^{U_g}} & \mathbf{I}_{F_{ss',b = b'}^{U_g}} & \mathbf{0}_{F_{ss',b = b'}^{U_g} \times F_{ss',b > b'}^{U_g}} \\ \mathbf{I}_{F_{ss',b < b'}^{U_g}} & \mathbf{0}_{F_{ss',b < b'}^{U_g} \times F_{ss',b = b'}^{U_g}} & \mathbf{0}_{F_{ss',b < b'}^{U_g} \times F_{ss',b > b'}^{U_g}} \end{bmatrix},$$

when the respective rows correspond to $s > s'$.

Example with two random effects

Consider a grouping factor g with $\rho^{U_g} = 2$ components. Omitting the dimensions of the submatrices for better readability, the constraint matrix \mathbf{W}^{U_g} is given by

$$\mathbf{W}^{U_g} = \begin{array}{l} (1,1) \\ (1,2) \\ (2,1) \\ (2,2) \end{array} \left[\begin{array}{c|c|c} \mathbf{I} & \mathbf{0} & \\ \mathbf{0} & \mathbf{I} & \\ \mathbf{I} & \mathbf{0} & \\ \hline & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \hline & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \hline & & & \mathbf{I} & \mathbf{0} \\ & & & \mathbf{0} & \mathbf{I} \\ & & & \mathbf{I} & \mathbf{0} \end{array} \right],$$

yielding the reduced design matrix $\mathbf{M}^{U_g \Delta r}$

$$\left[\underbrace{\mathbf{M}_{11,b < b'}^{U_g \Delta} + \mathbf{M}_{11,b > b'}^{U_g \Delta}}_{s=s'=1} \mid \mathbf{M}_{11,b = b'}^{U_g \Delta} \mid \underbrace{\mathbf{M}_{12,b < b'}^{U_g \Delta} + \mathbf{M}_{21,b > b'}^{U_g \Delta} \mid \mathbf{M}_{12,b = b'}^{U_g \Delta} + \mathbf{M}_{21,b = b'}^{U_g \Delta}}_{s < s' (s=1, s'=2)} \mid \mathbf{M}_{12,b > b'}^{U_g \Delta} + \mathbf{M}_{21,b < b'}^{U_g \Delta} \mid \underbrace{\mathbf{M}_{22,b < b'}^{U_g \Delta} + \mathbf{M}_{22,b > b'}^{U_g \Delta}}_{s=s'=2} \mid \mathbf{M}_{22,b = b'}^{U_g \Delta} \right].$$

Example with three random effects

Analogously, for a grouping factor g with $\rho^{U_g} = 3$ components the constraint matrix \mathbf{W}^{U_g} is given by

$$\mathbf{W}^{U_g} = \begin{pmatrix} (1,1) & \begin{matrix} I & 0 \\ 0 & I \\ I & 0 \end{matrix} & & & & & \\ (1,2) & & \begin{matrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{matrix} & & & & \\ (1,3) & & & \begin{matrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{matrix} & & & \\ (2,1) & & \begin{matrix} 0 & 0 & I \\ 0 & I & 0 \\ I & 0 & 0 \end{matrix} & & & & \\ (2,2) & & & & \begin{matrix} I & 0 \\ 0 & I \\ I & 0 \end{matrix} & & & \\ (2,3) & & & & & \begin{matrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{matrix} & \\ (3,1) & & & \begin{matrix} 0 & 0 & I \\ 0 & I & 0 \\ I & 0 & 0 \end{matrix} & & & \\ (3,2) & & & & & \begin{matrix} 0 & 0 & I \\ 0 & I & 0 \\ I & 0 & 0 \end{matrix} & \\ (3,3) & & & & & & \begin{matrix} I & 0 \\ 0 & I \\ I & 0 \end{matrix} \end{pmatrix}$$

D.3 Supplementary application details and results

D.3.1 CD4 cell count data

In AIDS research, the CD4 cell counts as a function of time since seroconversion (SC)—the time at which HIV becomes detectable—often serve as a longitudinally measured biomarker which provides insight into the progression of the disease. As the virus destroys the CD4 cells, a decreasing number of CD4 cells indicates a progress of the disease. The considered data set is part of the Multicenter AIDS Cohort Study (MACS; Kaslow et al., 1987). It contains the CD4 cell count trajectories of 366 HIV infected subjects collected from month -18 to month 42 since SC. Measurements were taken at roughly semi-annual visits yielding a total of 1888 CD4 cell counts per milliliter of blood, with between 1 to 11 counts per subject and a median of 5. To reduce skewness, we base our analysis on the square root of the CD4 cell counts, which are depicted in Figure D.1, with some trajectories highlighted for better display and an estimated overall mean function. We can see that on average, the CD4 cell counts are decreasing over time.

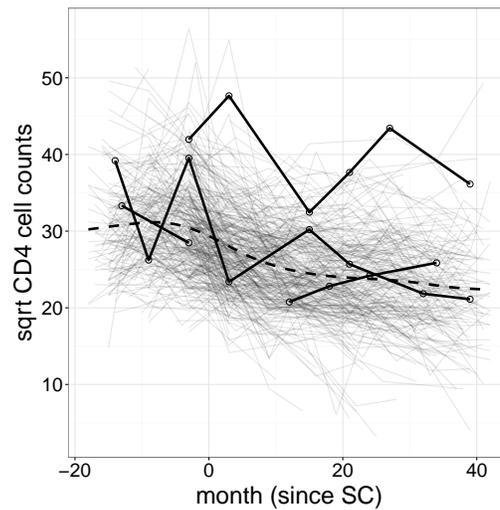


Figure D.1: Square root of observed CD4 cell count trajectories plotted against the months since SC. Shown are the trajectories of 366 HIV infected subjects. Some trajectories are highlighted for better display and an estimated smooth mean function (dashed) is shown.

The data are available in the R add-on package `refund` (Huang et al., 2016a) and are further described in Goldsmith et al. (2013). Similar data from this study were previously analyzed in, e.g., Diggle et al. (2002), Yao et al. (2005), and Peng and Paul (2009).

We fit Model (5.1) with only one fRI for each curve and an overall mean $\mu(t)$. In order to predict the continuous subject-specific trajectories with only few observations per subject available, we perform an FPCA based on our fast covariance smoothing approach (TRI-CONSTR and TRI-CONSTR-W with weights on the diagonal products). We demonstrate the similarity to the computationally less efficient approach proposed in Chapter 4 (WHOLE), in which all products of the centered functional responses enter the estimation. Moreover, we show that boundary effects occur on the diagonal when only the triangular surface is estimated without a symmetry constraint (TRI). We compare our results to those obtained from applying the covariance smoothing approach proposed by Xiao et al. (2017) (FACE) using R function `face.sparse` in the package `face` (Xiao et al., 2016a). Moreover, we compare with FACE-STEP-1, a modification of FACE, in which the covariance of the products of the centered functional responses is not accounted for and thus only the first step of the three-step procedure is performed.

We use 13 cubic B-spline basis functions for the estimation of the mean function and as marginal bases for the estimation of the auto-covariance surface using tensor products. To avoid over-fitting, we add a second order difference penalty. For our approach, we use the Kronecker sum penalty (cf. Chapter 5, Section 5.5). We use equidistant knots in function `face.sparse` instead of the default (quantile-based knots) which would require an adapted penalty that is not implemented. The equidistant grid, on which the mean and the auto-covariance are evaluated, is of length $\tilde{D} = 100$, with values between -18 and 42 . Note that so far function `face.sparse` assumes that the function argument takes

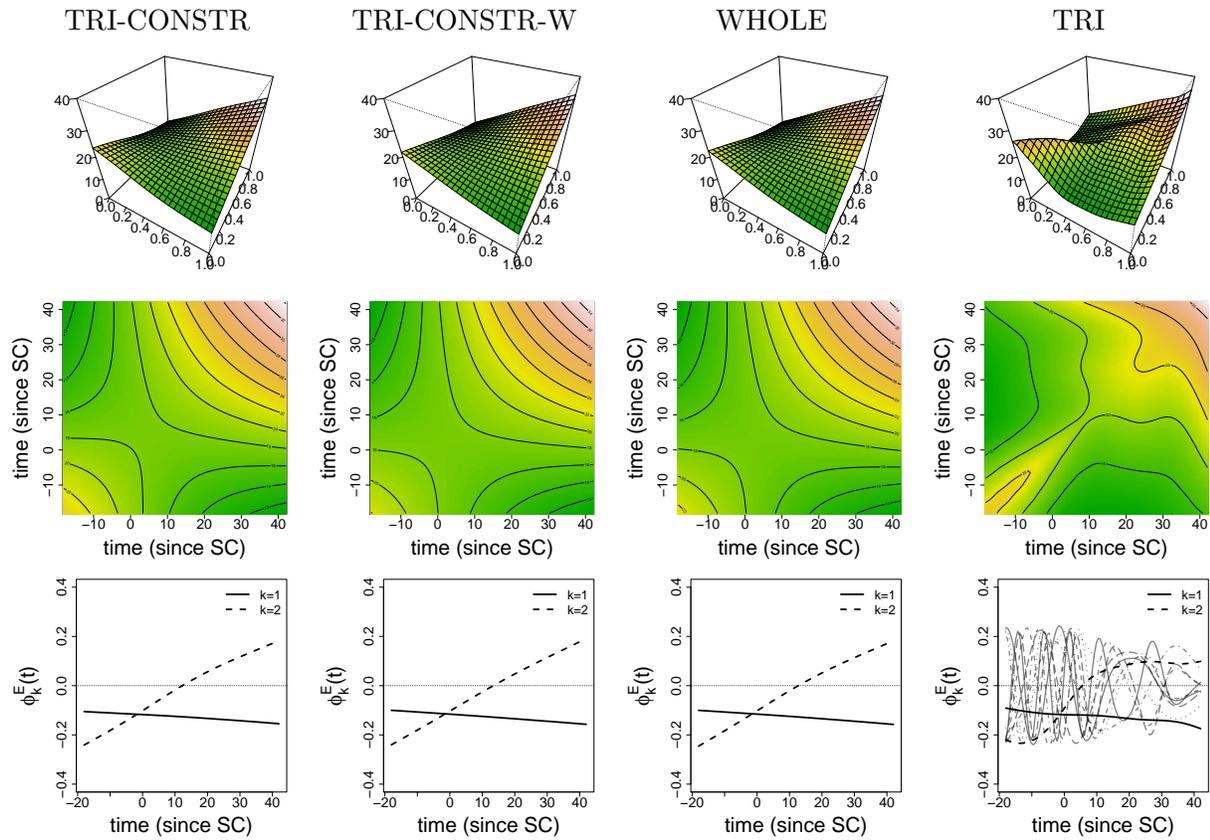


Figure D.2: Results for the curve-specific fRI for TRI-CONSTR, TRI-CONSTR-W, WHOLE, and TRI. Top row: Estimated covariance surfaces. Middle row: Contours of the estimated covariance surfaces. Bottom row: Estimated corresponding eigenfunctions.

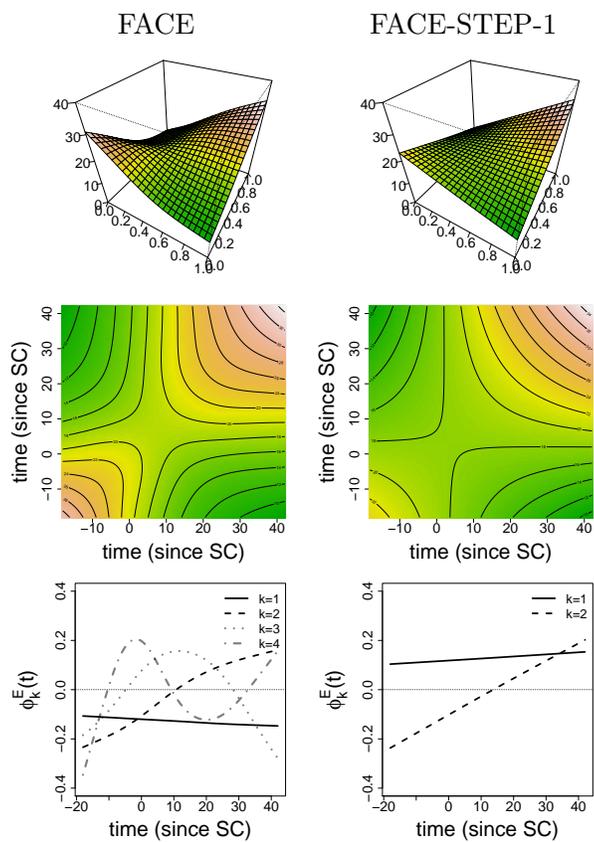


Figure D.3: Results for the curve-specific fRI for FACE and FACE-STEP-1. Top row: Estimated covariance surfaces. Middle row: Contours of the estimated covariance surfaces. Bottom row: Estimated corresponding eigenfunctions.

D.3.2 Speech production research data

In the following, we show additional results for our application to the speech production data (cf. Chapter 5, Section 5.6.2), including the estimated auto-covariances for the smooth error $E_i(t)$, the estimated eigenvalues for both random processes, and the estimated error variance.

Figure D.4 depicts the estimated surfaces and contours of the auto-covariance of the smooth error $E_i(t)$, reconstructed after truncation from the estimated eigenvalues and eigenfunctions which are shown in the bottom of the figure. As for the auto-covariance of the fRI for speakers, we can see from Figure D.4 that the two estimates based on our symmetric smoother (TRI-CONSTR, TRI-CONSTR-W) are very similar to each other and to the one obtained by using all products of the centered functional responses (WHOLE). Again, we obtain wigglier estimates for TRI—especially on the diagonal of the estimated surface. This corresponds to the fact that for TRI, the error variance is estimated to be zero. The first three eigenfunctions are very similar for all four compared methods. For TRI, however, nine more (high-frequency) eigenfunctions are chosen, yielding a wigglier surface estimate.

Table D.2 gives the complete variance decomposition for our model. The upper table shows the truncated estimated eigenvalues of \hat{K}^B for the four compared methods. The lower table shows the truncated estimated eigenvalues of the smooth error as well as the estimated error variance. For better display, all values are multiplied with 10^3 . It shows that the first two [three] estimated eigenvalues for $K^B(t, t')$ [$K^E(t, t')$] are very similar for all smoothing methods and that TRI-CONSTR-W and WHOLE are most similar. The estimated error variance is slightly higher for TRI-CONSTR than for the other approaches.

Table D.2: Truncated estimated eigenvalues of $K^B(t, t')$, $\hat{\nu}_k^B \cdot 10^3$, and of $K^E(t, t')$, $\hat{\nu}_k^E \cdot 10^3$, and estimated error variance $\hat{\sigma}^2 \cdot 10^3$ for all compared methods.

	$\hat{\nu}_1^B$	$\hat{\nu}_2^B$	$\hat{\nu}_3^B$	$\hat{\nu}_4^B$								
TRI-CONSTR	5.84	3.23										
TRI-CONSTR-W	5.83	3.23										
WHOLE	5.84	3.23										
TRI	5.85	3.27	0.42	0.23								

	$\hat{\nu}_1^E$	$\hat{\nu}_2^E$	$\hat{\nu}_3^E$	$\hat{\nu}_4^E$	$\hat{\nu}_5^E$	$\hat{\nu}_6^E$	$\hat{\nu}_7^E$	$\hat{\nu}_8^E$	$\hat{\nu}_9^E$	$\hat{\nu}_{10}^E$	$\hat{\nu}_{11}^E$	$\hat{\nu}_{12}^E$	$\hat{\sigma}^2$
TRI-CONSTR	19.54	7.57	2.73										4.21
TRI-CONSTR-W	19.53	7.59	2.71										4.02
WHOLE	19.53	7.59	2.73										3.95
TRI	19.68	7.78	2.96	1.39	0.94	0.64	0.50	0.36	0.30	0.23	0.20	0.16	0.00

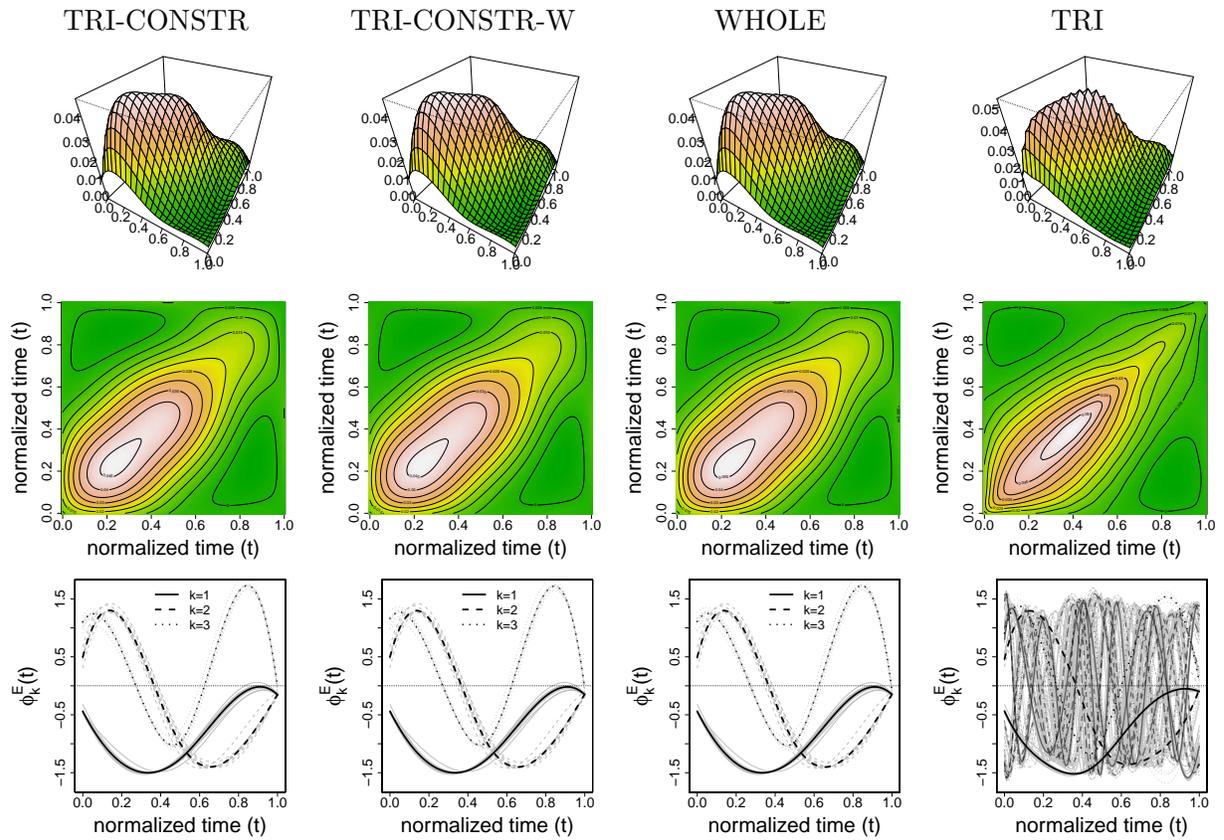


Figure D.4: Results for the smooth error curve $E_i(t)$ using the four smoothing methods. Top row: Estimated covariance surfaces. Middle row: Contours of the estimated covariance surfaces. Bottom row: Estimated eigenfunctions $\phi_k^E(t)$ based on the entire data set (black) and jackknife estimates with one of the speakers left out in turn (gray).

D.4 Supplementary simulation details and results

D.4.1 Generation details

For the scenario with crossed fRIs, we use two, one, and three eigenfunctions, estimated from the speech production data, for the generation of the auto-covariances of processes $B_i(t)$, $C_i(t)$, and $E_i(t)$, respectively. The resulting auto-covariance surfaces are shown in Figure D.5. The eigenfunctions used for data generation are shown in the bottom of the Figure. The corresponding eigenvalues used for data generation are shown in Table D.3, where also the error variance is given. The values are multiplied with 10^3 for better display. The underlying mean function is depicted in Figure D.6.

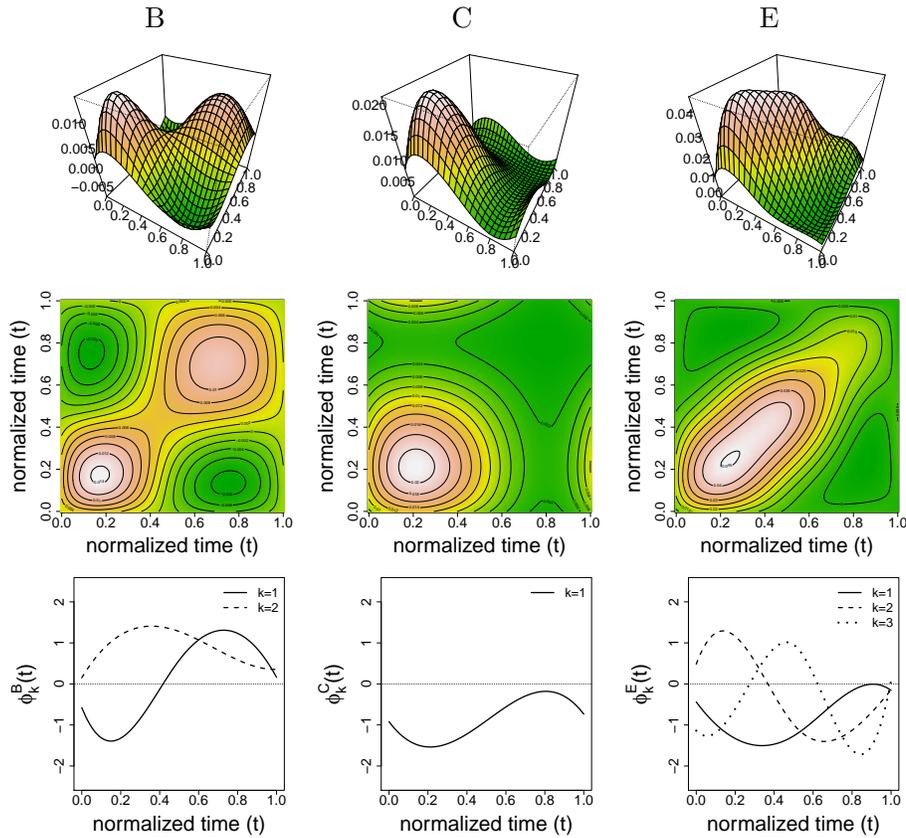


Figure D.5: Auto-covariances $K^B(t, t')$, $K^C(t, t')$, and $K^E(t, t')$ and their eigenfunctions used for the data generation for the scenario with crossed fRIs. Top row: Covariance surfaces. Middle row: Contours of the covariance surfaces. Bottom row: Corresponding eigenfunctions $\phi_k^B(t)$, $\phi_k^C(t)$, and $\phi_k^E(t)$.

Table D.3: Eigenvalues $\nu_k^X \cdot 10^3$, $X \in \{B, C, E\}$, and error variance $\sigma^2 \cdot 10^3$ used for data generation for the scenario with crossed fRIs.

ν_1^B	ν_2^B	ν_1^C	ν_1^E	ν_2^E	ν_3^E	σ^2
5.86	2.71	8.89	19.05	7.53	2.66	5.62

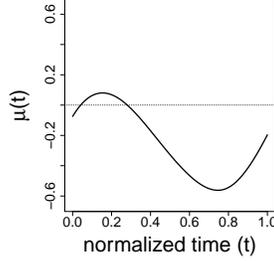


Figure D.6: Mean function used for the data generation for the scenario with crossed fRIs.

D.4.2 Measures of goodness of fit

We use root relative mean squared errors (rrMSEs) as measures of goodness of fit for all model components (cp. Appendix C.)

For vector-valued estimates $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta} = (\theta_1, \dots, \theta_L)^\top$, we define the rrMSE as

$$\text{rrMSE}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) = \sqrt{\frac{\frac{1}{L} \sum_{l=1}^L (\theta_l - \hat{\theta}_l)^2}{\frac{1}{L} \sum_{l=1}^L \theta_l^2}}. \quad (\text{D.3})$$

We use that for the random basis weights ξ_{ik}^X , (D.3) is approximately $\sqrt{1/L^X \sum_{l=1}^{L^X} (\xi_{ik}^X - \hat{\xi}_{ik}^X)^2} / \nu_k^X$, $X \in \{B, C, E\}$. The form of the rrMSE for scalar estimates results as special case of D.3 with $L = 1$.

For all functions $\theta(t)$, we approximate the integrals by sums and obtain

$$\text{rrMSE}[\theta(\cdot), \hat{\theta}(\cdot)] = \sqrt{\frac{\frac{1}{\tilde{D}} \sum_{d=1}^{\tilde{D}} [\theta(t_d) - \hat{\theta}(t_d)]^2}{\frac{1}{\tilde{D}} \sum_{d=1}^{\tilde{D}} \theta(t_d)^2}}. \quad (\text{D.4})$$

As the eigenfunctions are only unique up to sign, we also compute the rrMSEs of the estimated eigenfunctions mirrored around the x-axis and choose the smaller rrMSE. For the random processes, we additionally average over the respective levels. For centered processes, we use that the denominator simplifies to the average variance.

For bivariate functions, such as the auto-covariances, we define

$$\text{rrMSE} \left[\theta(\cdot, \cdot), \hat{\theta}(\cdot, \cdot) \right] = \sqrt{\frac{\frac{1}{\tilde{D}^2} \sum_{t_d, t_{d'}=1}^{\tilde{D}} \left[\theta(t_d, t_{d'}) - \hat{\theta}(t_d, t_{d'}) \right]^2}{\frac{1}{\tilde{D}^2} \sum_{t_d, t_{d'}=1}^{\tilde{D}} \theta(t_d, t_{d'})^2}}. \quad (\text{D.5})$$

D.4.3 Results for the scenario with independent curves

In the following, we show the complete results for the remaining ten settings (Setting 2–Setting 11) for the scenario with independent curves. Table D.4 lists the different settings we consider for this scenario. In Figure D.7 to Figure D.16, we depict boxplots of the rrMSEs based on the 200 simulation runs for all model components.

Table D.4: Specification of the eleven considered settings for the scenario with independent curves. The simple eigenfunctions are given as $\{\phi_1(t) = 1, \phi_2(t) = \sqrt{3}(2t - 1)\}$, the complex eigenfunctions are given as $\{\phi_1(t) = \sin(2\pi t), \phi_2(t) = \cos(2\pi t)\}$. The results for Setting 1 are shown in Section 5.7.2 of Chapter 5.

Setting	grid	eigenfunctions	eigenvalues	error variance
Setting 1	dense	complex	$\nu_1 = 2, \nu_2 = 1$	$\sigma^2 = 0.05$
Setting 2	dense	complex	$\nu_1 = 2, \nu_2 = 1,$	$\sigma^2 = 0.5$
Setting 3	dense	simple	$\nu_1 = 2, \nu_2 = 1$	$\sigma^2 = 0.05$
Setting 4	dense	simple	$\nu_1 = 2, \nu_2 = 1$	$\sigma^2 = 0.5$
Setting 5	sparse	simple	$\nu_1 = 2, \nu_2 = 1$	$\sigma^2 = 0.05$
Setting 6	sparse	simple	$\nu_1 = 2, \nu_2 = 1$	$\sigma^2 = 0.5$
Setting 7	dense	complex	$\nu_1 = 0.15, \nu_2 = 0.075$	$\sigma^2 = 0.05$
Setting 8	dense	complex	$\nu_1 = 0.15, \nu_2 = 0.075$	$\sigma^2 = 0.5$
Setting 9	dense	simple	$\nu_1 = 0.15, \nu_2 = 0.075$	$\sigma^2 = 0.05$
Setting 10	dense	simple	$\nu_1 = 0.15, \nu_2 = 0.075$	$\sigma^2 = 0.5$
Setting 11	dense	complex	$\nu_1 = 2, \nu_2 = 1$	$\sigma^2 = 0.01$

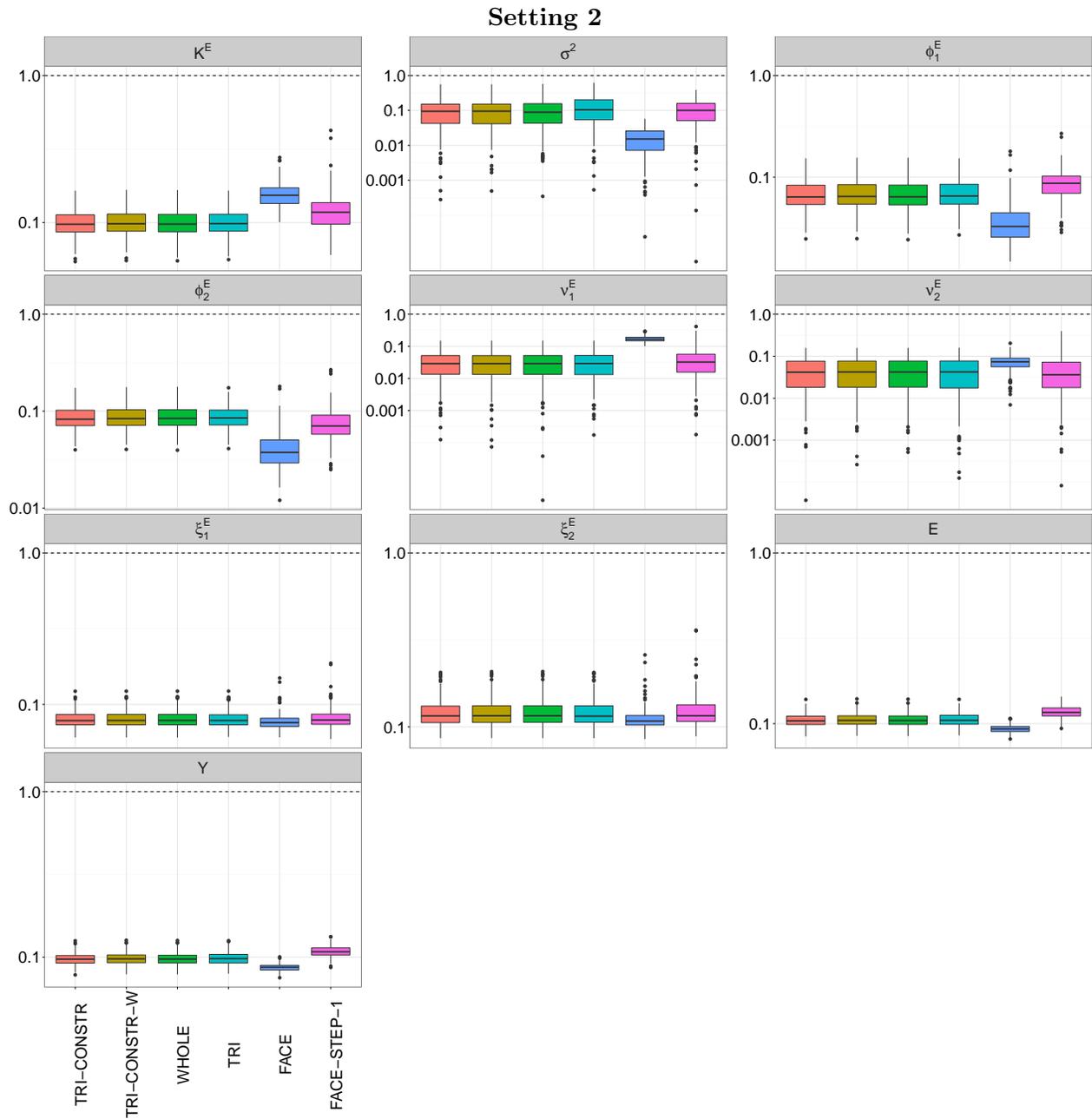


Figure D.7: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 2 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E , ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E , ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

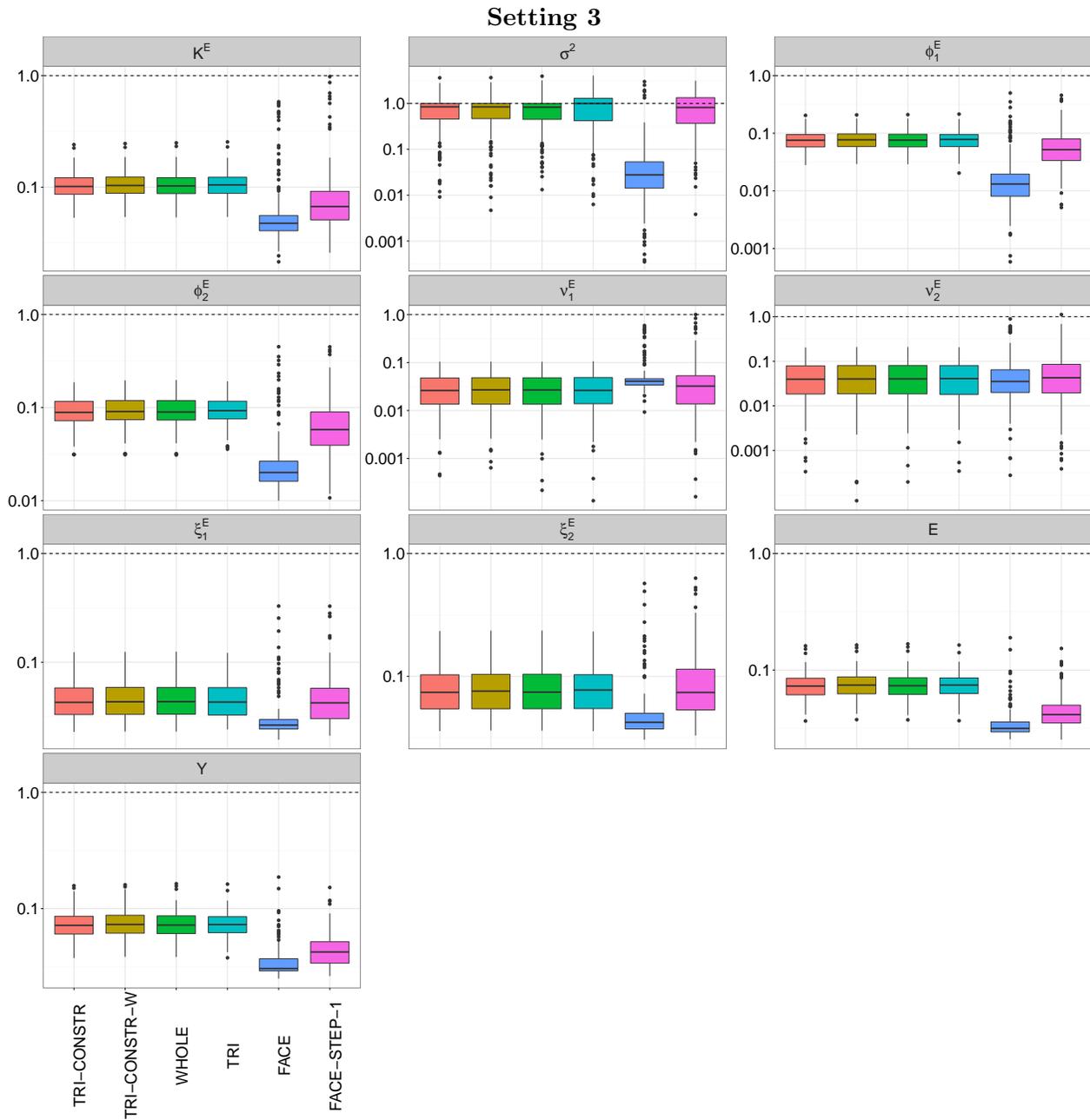


Figure D.8: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 3 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

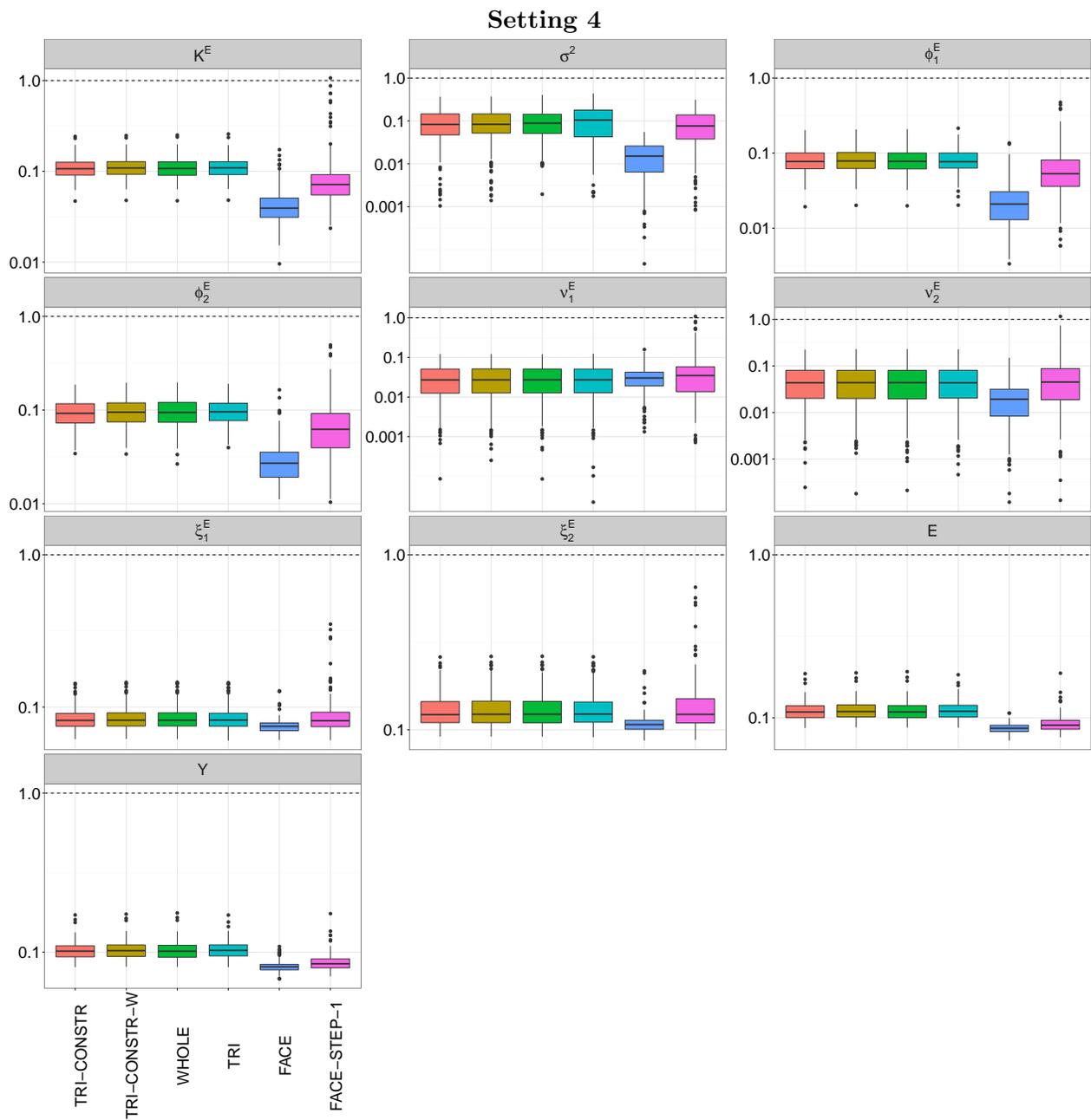


Figure D.9: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 4 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E , ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E , ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

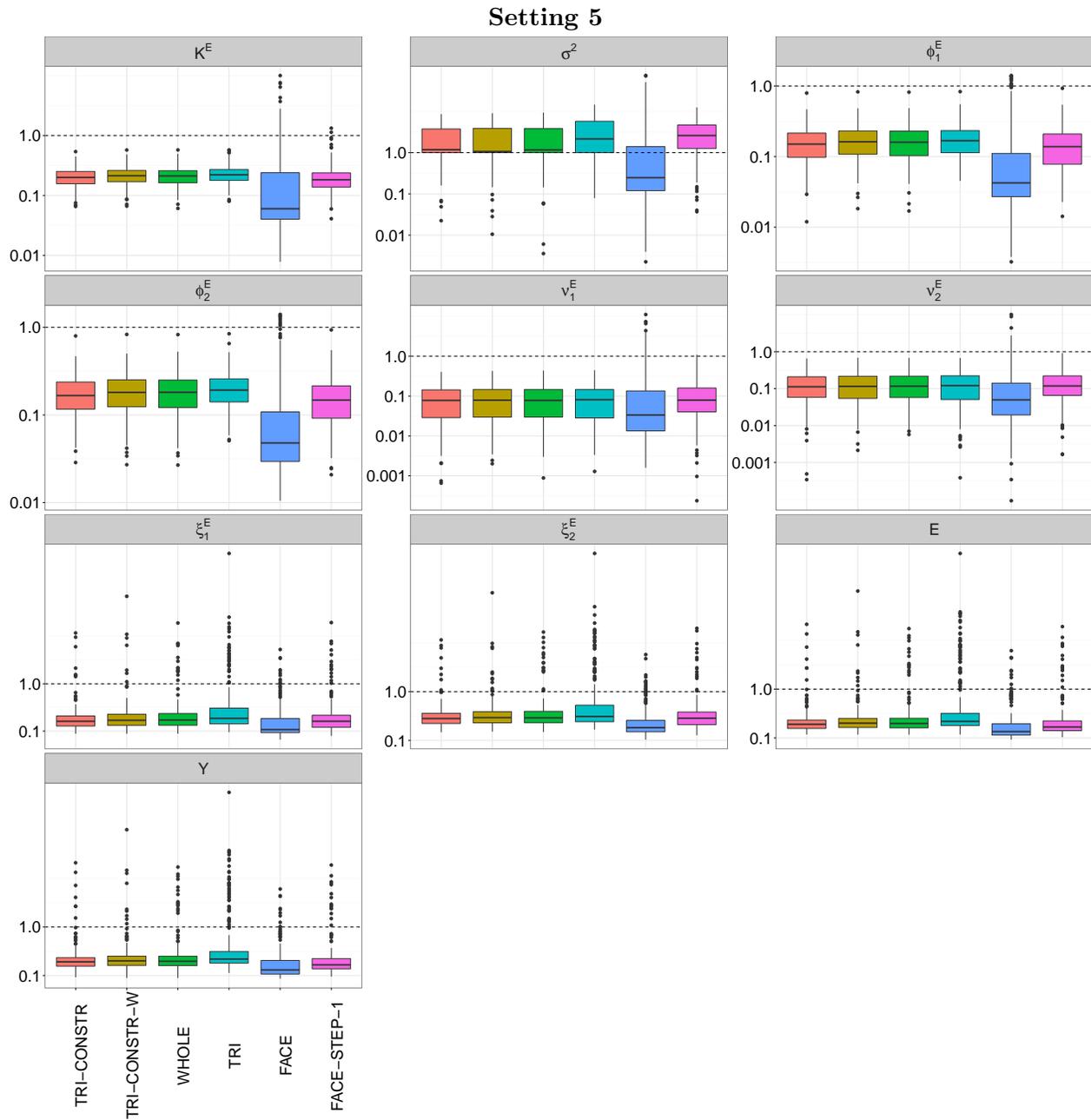


Figure D.10: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 5 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

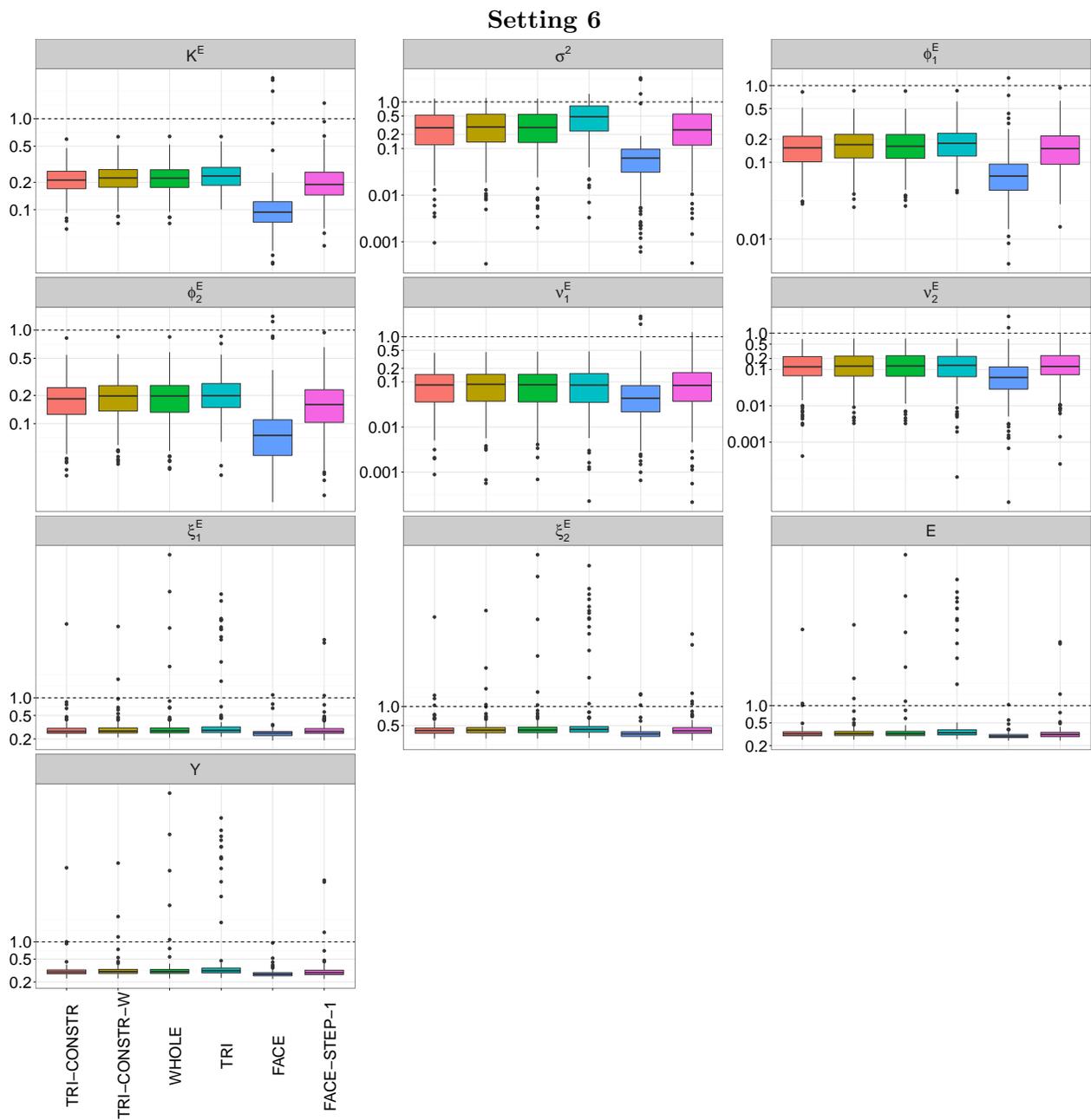


Figure D.11: Boxplots of the rrMSEs (log₁₀ scale at y-axis) for Setting 6 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

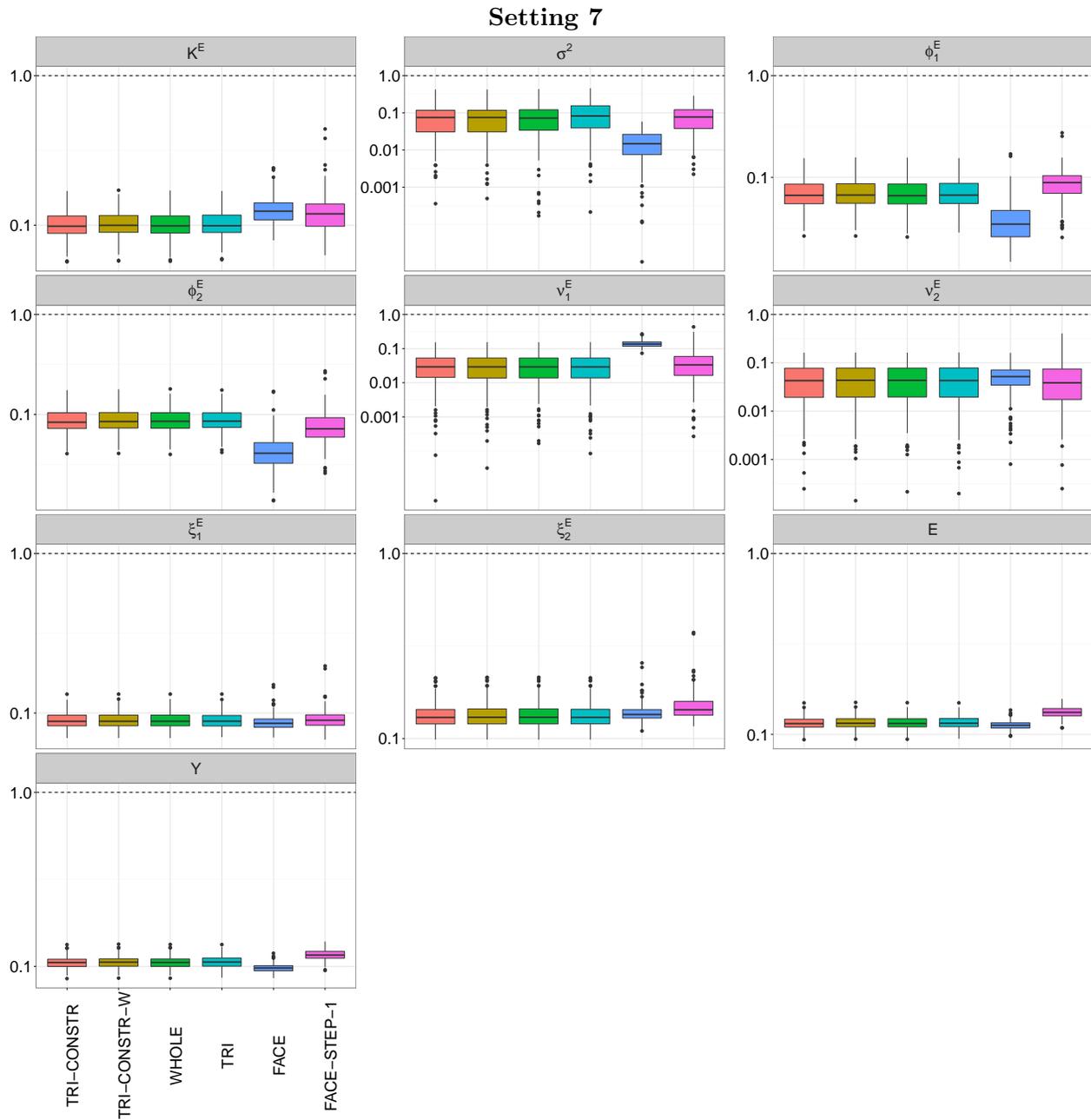


Figure D.12: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 7 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

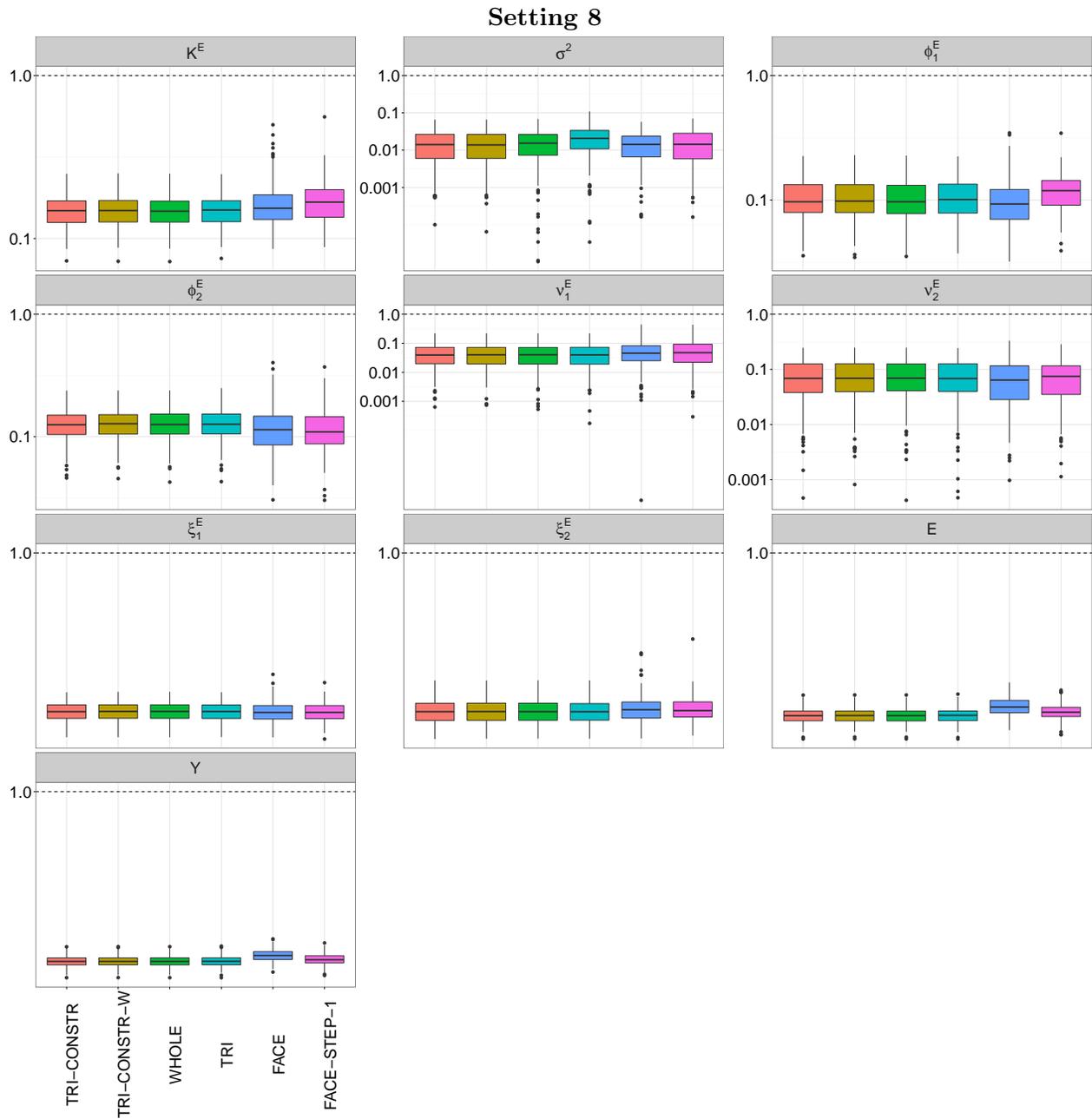


Figure D.13: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 8 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E , ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E , ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

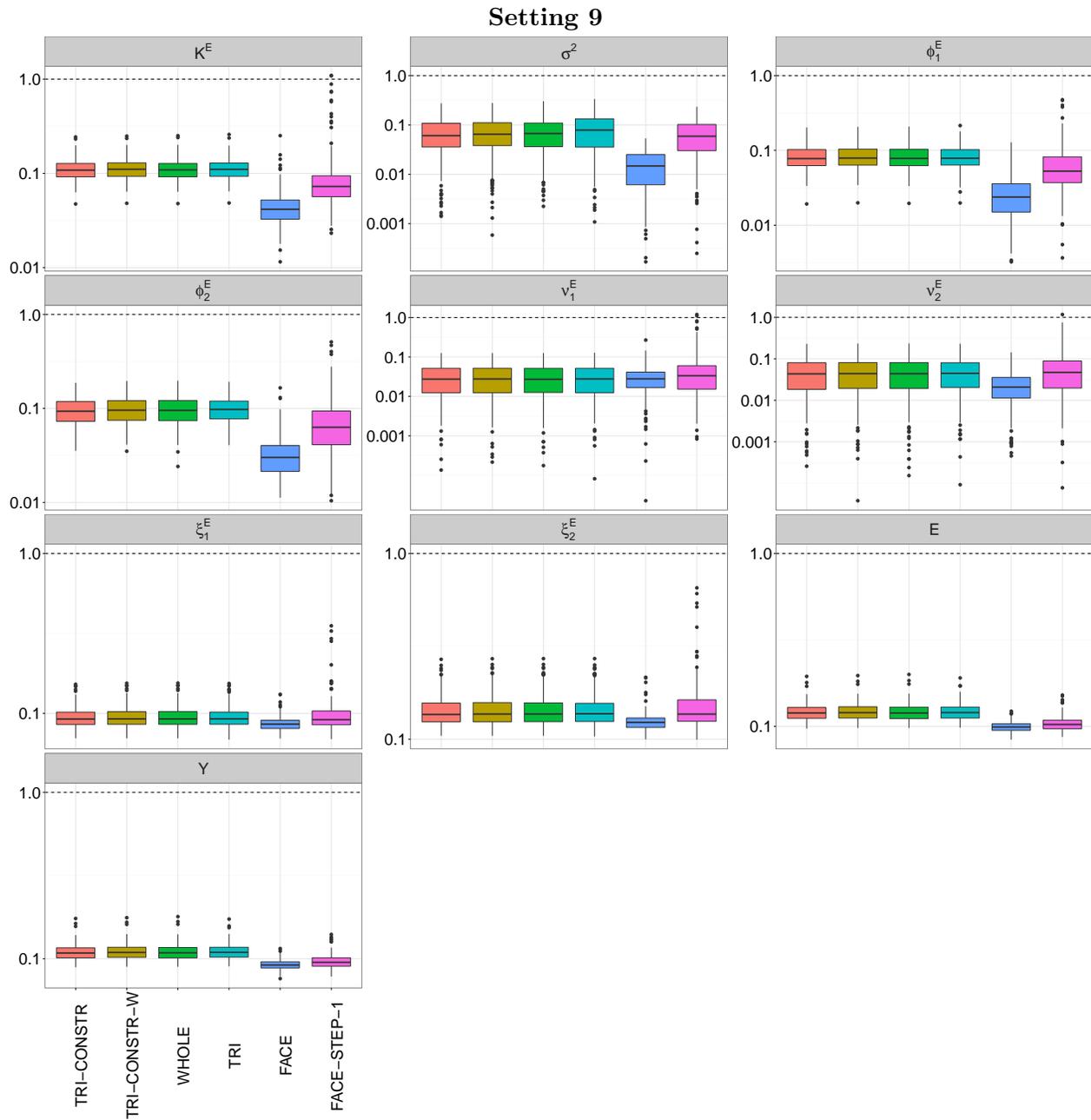


Figure D.14: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 9 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

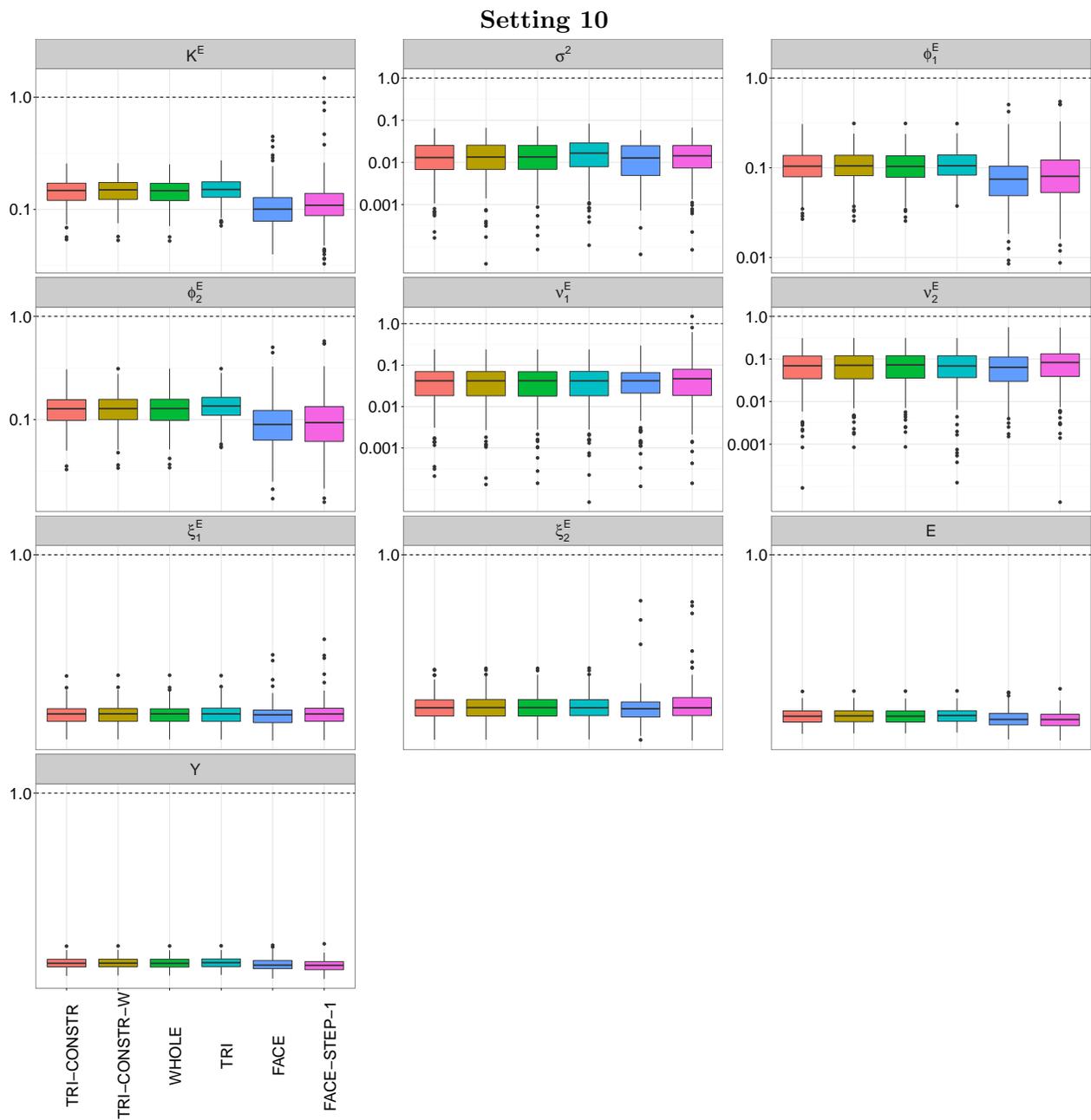


Figure D.15: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 10 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

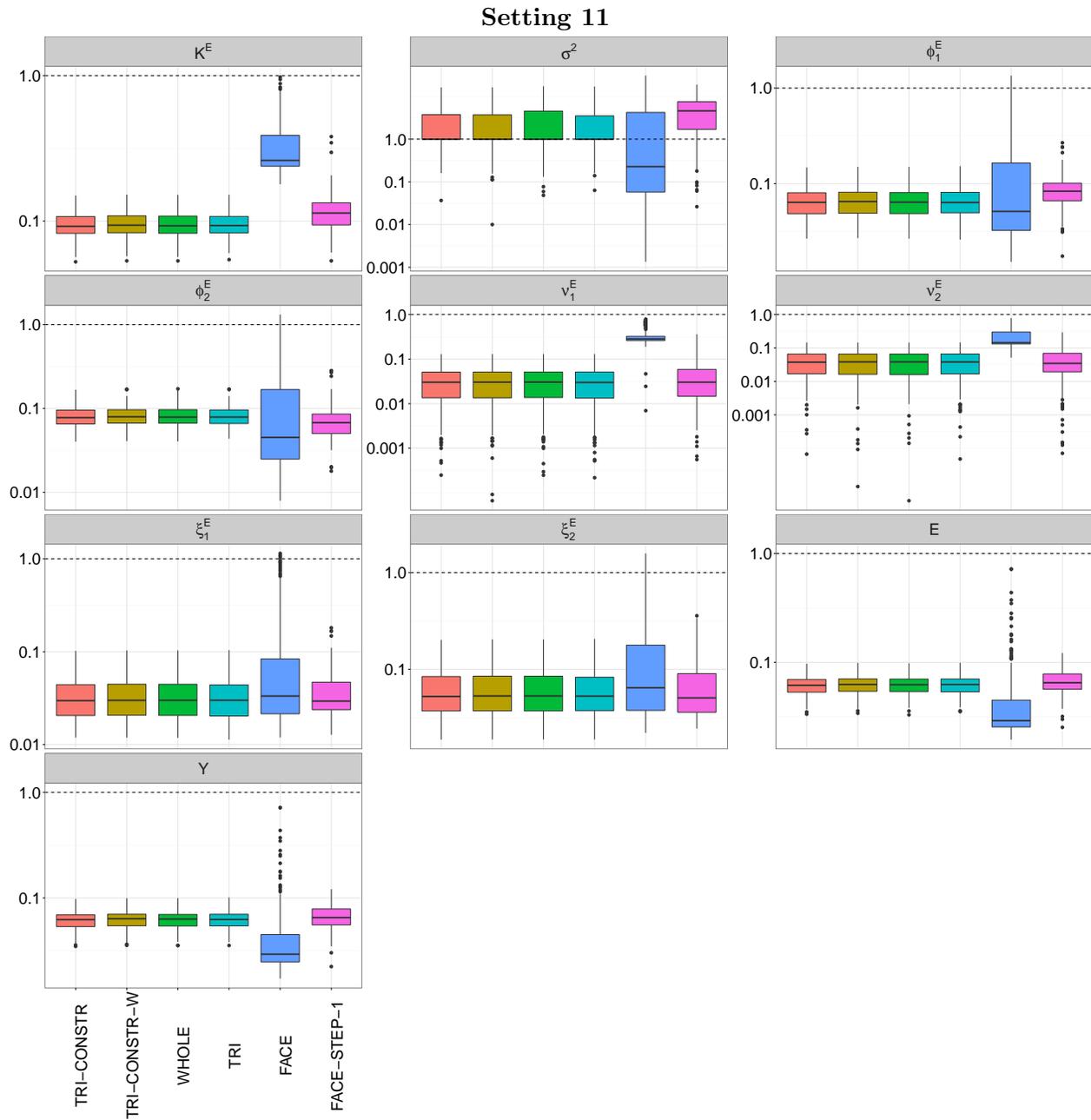


Figure D.16: Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 11 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$.

D.4.4 Results for the scenario with crossed fRIs

In the following, we show the remaining results for the scenario with crossed fRIs. Figure D.17, shows boxplots of the rrMSEs for the estimated eigenfunctions and eigenvalues, as well as for the random basis weights for the three random processes $B_i(t)$, $C_i(t)$, and $E_i(t)$. All boxplots are based on 200 simulation runs.

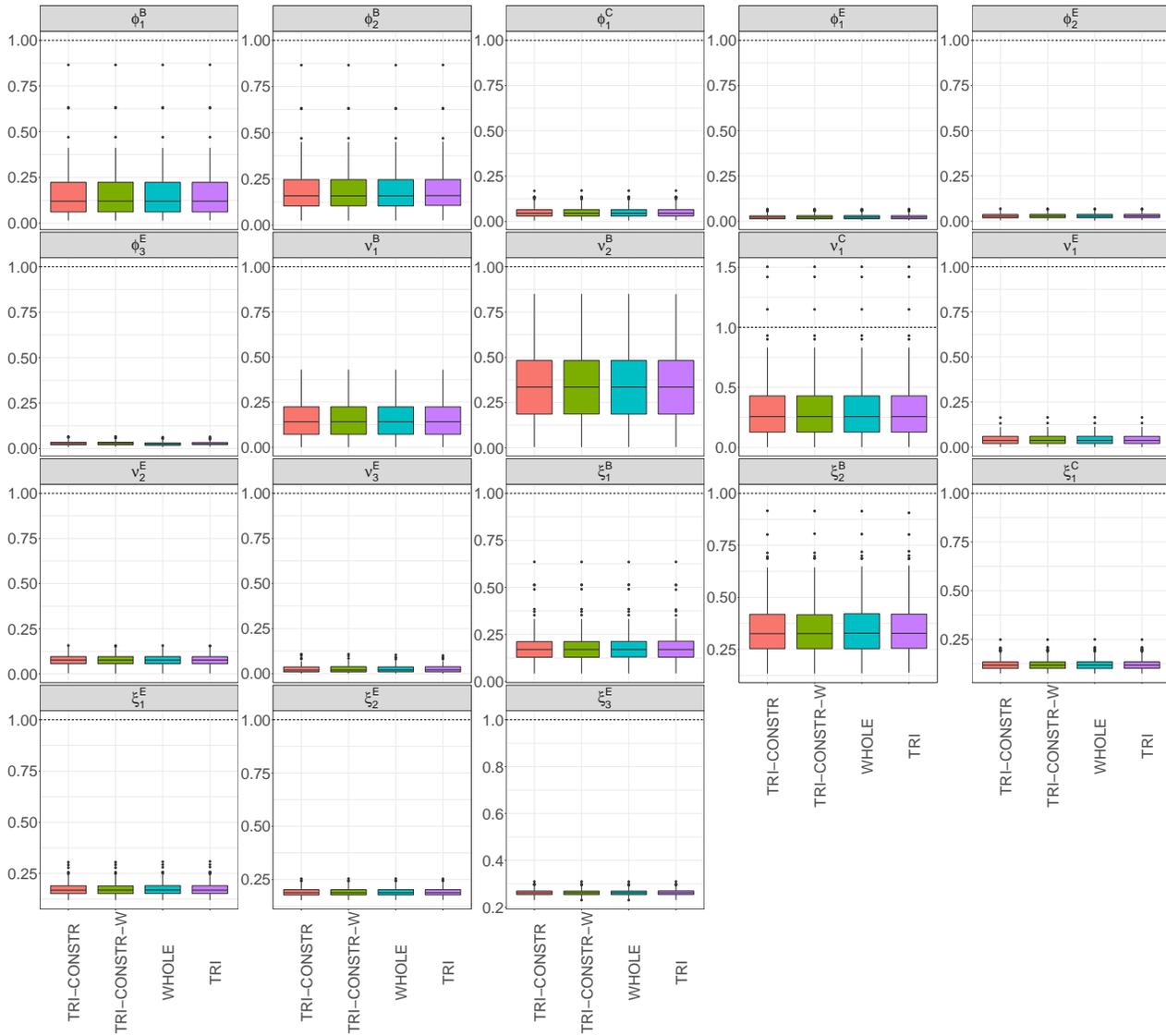


Figure D.17: Boxplots of the rrMSEs for the crossed fRIs setting. Shown are boxplots for all remaining model components, which are not shown in Section 5.7.2 of Chapter 5: The rrMSEs for the eigenfunctions and eigenvalues of the three auto-covariances $K^B(t, t')$, $K^C(t, t')$, and $K^E(t, t')$, as well as the rrMSEs for the corresponding random basis weights.

Appendix E

Details on the Implementations

This appendix is divided into two main parts. The first part provides further details on the implementation of the estimation approach for functional linear mixed models for equal sampling grids as proposed in Chapter B. The implementation is included in the R add-on package `denseFLMM` (R Core Team, 2016; Greven and Cederbaum, 2017). The second part gives additional details on the R add-on package `sparseFLMM` (Cederbaum, 2016) that implements the estimation approach for functional linear mixed models for unequal and sparse sampling grids (Chapter 4) as well as the fast additive covariance smoothing approach (Chapter 5).

E.1 R add-on package `denseFLMM`

This section provides a slightly amended version of the manual of function `denseFLMM` in the package of the same name, including example code. In addition, code for the analysis of the tissue spectroscopy data analyzed in Chapter 3 is shown.

E.1.1 Manual of R function `denseFLMM`

Functional linear mixed models for densely sampled data

Description

Estimation of functional linear mixed models (FLMMs) for functional data sampled on equal grids based on functional principal component analysis (FPCA). The implemented models are special cases of the general FLMM

$$Y_i(t_d) = \mu(t_d, \mathbf{x}_i) + \mathbf{z}_i^\top \mathbf{U}(t_d) + \varepsilon_i(t_d), \quad i = 1, \dots, n, \quad d = 1, \dots, D,$$

with $Y_i(t_d)$ the value of the response of curve i at observation point t_d , $\mu(t_d, \mathbf{x}_i)$ is a mean function, which may depend on covariates \mathbf{x}_i and needs to be estimated beforehand. \mathbf{z}_i is a covariate vector,

which is multiplied with the vector of functional random effects $\mathbf{U}(t_d)$. $\varepsilon_i(t_d)$ is independent and identically distributed white noise measurement error with homoscedastic, constant variance. For more details, see Chapter 3.

The code implements the general functional linear mixed model for n curves observed at D grid points. Grid points are assumed to be equidistant and so far no missings are assumed. The curves are assumed to be centered. The functional random effects for each grouping factor are assumed to be correlated (e.g., random intercept and slope curves). The code can handle group-specific functional random effects including group-specific smooth errors. Covariates are assumed to be standardized.

Note that for consistency with the code, G here denotes the number of grouping factors not used for the estimation of the error variance, i.e., all except the smooth error term(s). This is in contrast to the model formulation in Chapter 3, where G denotes the total number of grouping factors. The total number of grouping factors is here denoted by H .

Usage

```
denseFLMM(Y, gridpoints = 1:ncol(Y), Zlist = NA, G = NA, Lvec = NA,
  groups = matrix(1, nrow(Y), 1), Zvars, L = NA, NPC = NA,
  smooth = FALSE, bf = 10, smoothalg = "gamm")
```

Arguments

- Y** $n \times D$ matrix of n curves observed on D grid points. **Y** is assumed to be centered by its mean function.
- gridpoints** vector of grid points along curves, corresponding to columns of **Y**. Defaults to `matrix(1, nrow(Y), 1)`.
- Zlist** list of length H of ρ^{U_h} design matrices $\mathbf{Z}_{.s}^{U_h}$, $s = 1, \dots, \rho^{U_h}$, $h = 1, \dots, H$. Needed instead of **Zvars** and **groups** if group-specific functional random effects are present. Defaults to `NA`, then **Zvars** and **groups** needed.
- G** number of grouping factors not used for estimation of error variance. Needed if **Zlist** is used instead of **Zvars** and **groups**. Defaults to `NA`.
- Lvec** vector of length H containing the number of levels for each grouping factor. Needed if **Zlist** is used instead of **Zvars** and **groups**. Defaults to `NA`.
- groups** $n \times G$ matrix with G grouping factors for the rows of **Y**, where G denotes the number of grouping factors not used for the estimation of the error variance. Defaults to `groups = matrix(1, nrow(Y), 1)`. Set to `NA` when **Zlist** is used to specify group-specific functional random effects.

<code>Zvars</code>	list of length G with $n \times \rho^{U_g}$ matrices of random variables for grouping factor g , where G denotes the number of grouping factors not used for the estimation of the error variance. Random curves for each grouping factor are assumed to be correlated (e.g., random intercept and slope). Set to <code>NA</code> when <code>Zlist</code> is used to specify group-specific functional random effects.
<code>L</code>	pre-specified level of variance explained (between 0 and 1), determines number of functional principal components.
<code>NPC</code>	vector of length H with number of functional principal components to keep for each functional random effect. Overrides argument <code>L</code> if not <code>NA</code> . Defaults to <code>NA</code> .
<code>smooth</code>	<code>TRUE</code> to add smoothing of the covariance matrices, otherwise covariance matrices are estimated using least squares. Defaults to <code>FALSE</code> .
<code>bf</code>	number of marginal basis functions used for all smooths. Defaults to <code>bf = 10</code> .
<code>smoothalg</code>	smoothing algorithm used for covariance smoothing. Available options are <code>"gamm"</code> , <code>"gamGCV"</code> , <code>"gamREML"</code> , <code>"bamGCV"</code> , <code>"bamREML"</code> , and <code>"bamfREML"</code> . <code>"gamm"</code> uses restricted maximum likelihood (REML) estimation based on function <code>gamm</code> in R add-on package <code>mgcv</code> . <code>"gamGCV"</code> and <code>"gamREML"</code> use generalized cross-validation (GCV) and REML estimation based on function <code>gam</code> in package <code>mgcv</code> , respectively. <code>"bamGCV"</code> , <code>"bamREML"</code> , and <code>"bamfREML"</code> use GCV, REML, and a fast REML estimation based on function <code>bam</code> in package <code>mgcv</code> , respectively. Defaults to <code>"gamm"</code> .

Details

The model fit for centered curves $Y_i(\cdot)$ is

$$\mathbf{Y} - \boldsymbol{\mu} = \mathbf{Z}\mathbf{U} + \boldsymbol{\varepsilon},$$

with $\mathbf{Y} = [Y_i(t_d)]_{i=1, \dots, n, d=1, \dots, D}$, \mathbf{Z} consisting of H blocks \mathbf{Z}^{U_h} for H grouping factors, $\mathbf{Z} = [\mathbf{Z}^{U_1} | \dots | \mathbf{Z}^{U_H}]$, and each $\mathbf{Z}^{U_h} = [\mathbf{Z}_{11}^{U_h} | \dots | \mathbf{Z}_{1\rho^{U_h}}^{U_h} | \dots | \mathbf{Z}_{L^{U_h}1}^{U_h} | \dots | \mathbf{Z}_{L^{U_h}\rho^{U_h}}^{U_h}]$. \mathbf{U} is row-wise divided into blocks $\mathbf{U}_1, \dots, \mathbf{U}_H$, corresponding to \mathbf{Z} . Let further $\mathbf{Z}_{\cdot s}^{U_h}$ denote the $n \times L^{U_h}$ matrix containing columns $\mathbf{Z}_{l s}^{U_h}$, $l = 1, \dots, L^{U_h}$.

In case no group-specific functional random effects are specified, column l in $\mathbf{Z}_{\cdot s}^{U_g}$, $s = 1, \dots, \rho^{U_g}$, is assumed to be equal to the s th variable (column) in `Zvars[[g]]` times an indicator for the l th level of grouping factor g , $g = 1, \dots, G$.

The covariances are evaluated on an equidistant grid $\tilde{\mathcal{D}}$ of length \tilde{D} . The estimated eigenvectors and eigenvalues are rescaled to ensure that the approximated eigenfunctions are orthonormal with respect to the L^2 -inner product.

The estimation of the error variance takes place in two steps. In case of smoothing (`smooth = TRUE`), the error variance is first estimated as the average difference of the raw and the smoothed diagonal values as described in Chapter 3. In case no smoothing is applied, the estimated

error variance is zero. Subsequent to the eigen decomposition and selection of the eigenfunctions to keep for each grouping factor, the estimated error variance is recalculated in order to capture the left out variability due to the truncation of the infinite Karhunen-Loève expansions.

Value

The function returns a list containing the input arguments `Y`, `gridpoints`, `groups`, `Zvars`, `L`, `smooth`, `bf`, and `smoothalg`. It additionally contains:

- `Zlist` either the input argument `Zlist` or if set to `NA`, the computed list of design matrices $\mathbf{Z}_{.s}^{U_h}$, $s = 1, \dots, \rho^{U_h}$, $h = 1, \dots, H$.
- `G` either the input argument `G` or if set to `NA`, the computed number of grouping factors G not used for the estimation of the error variance.
- `Lvec` either the input argument `Lvec` or if set to `NA`, the computed vector of length H containing the number of levels for each grouping factor (including the smooth error(s)).
- `NPC` either the input argument `NPC` or if set to `NA`, the number of functional principal components kept for each functional random effect (including the smooth error(s)).
- `rhovec` vector of length H of number of random effects for each grouping factor (including the smooth error(s)).
- `phi` list of length H of $\tilde{D} \times N^{U_h}$ matrices containing the N^{U_h} functional principal components kept per grouping factor (including the smooth error(s)), $h = 1, \dots, H$, where \tilde{D} denotes the length of the equidistant evaluation grid.
- `sigma2` estimated measurement error variance σ^2 .
- `nu` list of length H of $N^{U_h} \times 1$ vectors of estimated eigenvalues $\nu_k^{U_h}$, $h = 1, \dots, H$.
- `xi` list of length H of $L^{U_h} \times N^{U_h}$ matrices containing the predicted random basis weights, $h = 1, \dots, H$. Within matrices, basis weights are ordered corresponding to the ordered levels of the grouping factors, e.g., a grouping factor with levels 4, 2, 3, 1 ($L^{U_h} = 4$) will result in rows in `xi[[g]]` corresponding to levels 1, 2, 3, 4.
- `totvar` total average variance of the curves.
- `exvar` level of variance explained by the selected functional principal components (+ error variance).

Author(s)

Sonja Greven, Jona Cederbaum

See Also

For the estimation of functional linear mixed models for irregularly or sparsely sampled data based on functional principal component analysis, see function `sparseFLMM` in the package `sparseFLMM`.

Examples

```
#####
# load libraries
#####
require(mvtnorm)
require(Matrix)
set.seed(123)

#####
# specify data generation
#####
nus <- list(c(0.5, 0.3), c(1, 0.4), c(2)) # eigenvalues
sigmasq <- 2.5e-05 # error variance
NPCs <- c(rep(2, 2), 1) # number of eigenfunctions
Lvec <- c(rep(2, 2), 480) # number of levels
H <- 3 # total number of functional random effects
G <- 2 # number of functional random effects not used for
# the estimation of the error variance
gridpoints <- seq(from = 0, to = 1, length = 100) # grid points
class_nr <- 2 # number of groups

# define eigenfunctions
#####
funB1 <- function(k,t){
  if(k == 1)
    out <- sqrt(2) * sin(2 * pi * t)
  if(k == 2)
    out <- sqrt(2) * cos(2 * pi * t)
  out
}

funB2 <- function(k,t){
  if(k == 1)
    out <- sqrt(7) * (20 * t^3 - 30 * t^2 + 12 * t - 1)
  if(k == 2)
    out <- sqrt(3) * (2 * t - 1)
  out
}
```

```

funE <- function(k,t){
  if(k == 1)
    out <- 1 + t - t
  if(k == 2)
    out <- sqrt(5) * (6 * t^2 - 6 * t + 1)
  out
}

#####
# generate data
#####
D <- length(gridpoints) # number of grid points
n <- Lvec[3] # number of curves in total
class <- rep(1:class_nr, each = n / class_nr)
group1 <- rep(rep(1:Lvec[1], each = n / (Lvec[1] * class_nr)), class_nr)
group2 <- 1:n
data <- data.frame(class = class, group1 = group1, group2 = group2)

# get eigenfunction evaluations
#####
phis <- list(sapply(1:NPCs[1], FUN = funB1, t = gridpoints),
  sapply(1:NPCs[2], FUN = funB2, t = gridpoints),
  sapply(1:NPCs[3], FUN = funE, t = gridpoints))

# draw basis weights
#####
xis <- vector("list", H)
for(i in 1:H){
  if(NPCs[i] > 0){
    xis[[i]] <- rmvnorm(Lvec[i], mean = rep(0, NPCs[i]),
      sigma = diag(NPCs[i]) * nus[[i]])
  }
}

# construct functional random effects
#####
B1 <- xis[[1]] %*% t(phis[[1]])
B2 <- xis[[2]] %*% t(phis[[2]])
E <- xis[[3]] %*% t(phis[[3]])

B1_mat <- B2_mat <- E_mat <- matrix(0, nrow = n, ncol = D)
B1_mat[group1 == 1 & class == 1, ] <-
  t(replicate(n = n / (Lvec[1] * class_nr), B1[1, ], simplify = "matrix"))
B1_mat[group1 == 2 & class == 1, ] <-
  t(replicate(n = n / (Lvec[1] * class_nr), B1[2, ], simplify = "matrix"))

```

```

B2_mat[group1 == 1 & class == 2, ] <-
  t(replicate(n = n / (Lvec[1] * class_nr), B2[1, ], simplify = "matrix"))
B2_mat[group1 == 2 & class == 2, ] <-
  t(replicate(n = n / (Lvec[1] * class_nr), B2[2, ], simplify = "matrix"))

E_mat <- E

# draw white noise measurement error
#####
eps <- matrix(rnorm(n * D, mean = 0, sd = sqrt(sigmasq)),
  nrow = n, ncol = D)

# construct curves
#####
Y <- B1_mat + B2_mat + E_mat + eps

#####
# construct Zlist
#####
Zlist <- list()
Zlist[[1]] <- Zlist[[2]] <- Zlist[[3]] <- list()

d1 <- data.frame(a = as.factor(data$group1[data$class == 1]))
Zlist[[1]][[1]] <- rBind(sparse.model.matrix(~ -1 + a, d1),
  matrix(0, nrow = (1 / class_nr * n), ncol = (Lvec[1])))

d2 <- data.frame(a = as.factor(data$group1[data$class == 2]))
Zlist[[2]][[1]] <- rBind(matrix(0, nrow = (1 / class_nr * n),
  ncol = (Lvec[2])), sparse.model.matrix(~ -1 + a, d2))

d3 <- data.frame(a = as.factor(data$group2))
Zlist[[3]][[1]] <- sparse.model.matrix(~ -1 + a, d3)

#####
# run estimation
#####
results <- denseFLMM(Y = Y, gridpoints = gridpoints, Zlist = Zlist,
  G = G, Lvec = Lvec, groups = NA, Zvars = NA, L = 0.99999,
  NPC = NA, smooth = FALSE)

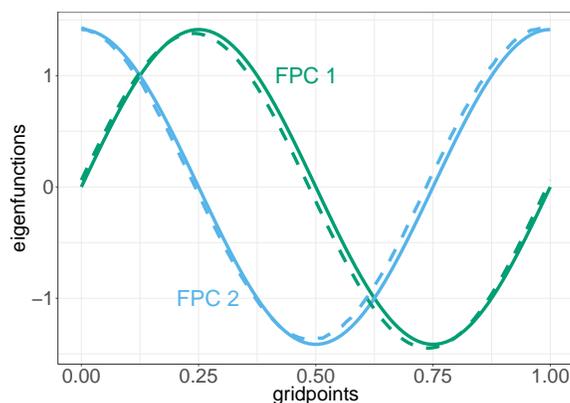
#####
# plot true versus estimated
# eigenfunctions of B1
#####

```

```
require(ggplot2)

eigenfun_df <- data.frame(phiB1_1 = phis[[1]][, 1], phiB1_2 = phis[[1]][, 2],
                        phiB1_1_hat = (-1) * results$phi[[1]][, 1],
                        phiB1_2_hat = (-1) * results$phi[[1]][, 2],
                        grid = gridpoints)

ggplot() + theme_bw() +
  geom_line(data = eigenfun_df, aes(gridpoints, phiB1_1), size = 2, color = "#009E73") +
  geom_line(data = eigenfun_df, aes(gridpoints, phiB1_2), size = 2, color = "#56B4E9") +
  geom_line(data = eigenfun_df, aes(gridpoints, phiB1_1_hat), size = 2, lty = 2,
            color = "#009E73") +
  geom_line(data = eigenfun_df, aes(gridpoints, phiB1_2_hat), size = 2, color = "#56B4E9",
            lty = 2) + ylab("eigenfunctions") + theme(axis.text = element_text(size = 25),
            axis.title = element_text(size = 25)) +
  annotate("text", x = 0.48, y = 1, label = "FPC 1", color = "#009E73", size = 10) +
  annotate("text", x = 0.27, y = -1, label = "FPC 2", color = "#56B4E9", size = 10)
```



E.1.2 R code for the application to the spectroscopy data

The following example R code using function `denseFLMM` demonstrates parts of the analysis of the tissue spectroscopy data considered in Chapter 3 after pre-processing. Note that the spectroscopy data cannot be made publicly available.

The pre-processing steps include interpolation of the curves on an equidistant grid and standardization of the curves as described in Chapter 3.

After pre-processing, the object `curves` is a matrix of dimension $n \times D = 8640 \times 1150$, `wavelengths` is a vector of length 1150 containing the wavelengths at which each curve is available. The vectors `Tissue`, `Pig`, and `Spot` of length 8640 specify for each observation to which tissue type, pig, and spot the observation belongs. Correspondingly, `Obs` is a vector of length 8640 with observation numbers from 1 to $n = 8640$.

The code shows the estimation of the model components of Model (3.7) in Chapter 3. For purpose of demonstration, the function is applied to the complete spectroscopy data rather than to the training data as in Chapter 3. The model includes a fixed effect for tissue type, tissue-specific functional random intercepts for pigs and spots, and a smooth error that is not tissue-specific.

Due to the tissue-specific functional random effects, the function argument `Zlist` rather than arguments `groups` and `Zvars` has to be specified. The construction of the `Zlist`-argument for this specific example is given below.

```
#####
# load libraries
#####
require(denseFLMM)
require(mgcv)
require(Matrix)

#####
# specify input parameters
#####
nplevs <- 12 # number of pigs
ntlevs <- 4  # number of tissue types
nslevs <- 6  # number of spots
nolevs <- 30 # number of observations per spot

# number of levels for each grouping factor
Lvec <- c(rep((nplevs), ntlevs), rep((nplevs) * nslevs, ntlevs),
  (ntlevs * (nplevs) * nslevs * nolevs))

# number of grouping factors not used for the estimation
# of the error variance, i.e., without smooth error
G <- length(Lvec) - 1

# construct Zlist
#####
Zlist <- list()
Zlist[[1]] <- Zlist[[2]] <- Zlist[[3]] <- Zlist[[4]] <- Zlist[[5]] <- Zlist[[6]] <-
  Zlist[[7]] <- Zlist[[8]] <- Zlist[[9]] <- list()

# functional random intercept for pigs for tissue 1
b1 <- data.frame(a = as.factor(Pig[Tissue == 1]))
Zlist[[1]][[1]] <- rBind(sparse.model.matrix(~ -1 + a, b1),
matrix(0, nrow = (3 / 4 * n), ncol = (nplevs)))

# functional random intercept for pigs for tissue 2
b2 <- data.frame(a = as.factor(Pig[Tissue == 2]))
Zlist[[2]][[1]] <- rBind(matrix(0, nrow=(1 / 4 * n),
```

```

ncol = (nplevs)), sparse.model.matrix(~ -1 + a, b2),
matrix(0, nrow = (2 / 4 * n), ncol = (nplevs)))

# functional random intercept for pigs for tissue 3
b3 <- data.frame(a = as.factor(Pig[Tissue == 3]))
Zlist[[3]][[1]] <- rBind(matrix(0, nrow = (2 / 4 * n),
  ncol = (nplevs)), sparse.model.matrix(~ -1 + a, b3),
  matrix(0, nrow = (1 / 4 * n), ncol = (nplevs)))

# functional random intercept for pigs for tissue 4
b4 <- data.frame(a = as.factor(Pig[Tissue == 4]))
Zlist[[4]][[1]] <- rBind(matrix(0, nrow = (3 / 4 * n),
  ncol = (nplevs)), sparse.model.matrix(~ -1 + a, b4))

# functional random intercept for spots for tissue 1
b5 <- data.frame(a = as.factor(Spot[Tissue == 1]))
Zlist[[5]][[1]] <- rBind(sparse.model.matrix(~ -1 + a, b5),
  matrix(0, nrow = (3 / 4 * n), ncol = ((nplevs) * nslevs)))

# functional random intercept for spots for tissue 2
b6 <- data.frame(a = as.factor(Spot[Tissue == 2]))
Zlist[[6]][[1]] <- rBind(matrix(0, nrow = (1 / 4 * n),
  ncol = (nplevs) * nslevs), sparse.model.matrix(~ -1 + a, b6),
  matrix(0, nrow = (2 / 4 * n), ncol = (nplevs) * nslevs))

# functional random intercept for spots for tissue 3
b7 <- data.frame(a = as.factor(Spot[Tissue == 3]))
Zlist[[7]][[1]] <- rBind(matrix(0, nrow = (2 / 4 * n),
  ncol = (nplevs) * nslevs), sparse.model.matrix(~ -1 + a, b7),
  matrix(0, nrow = (1 / 4 * n), ncol = (nplevs) * nslevs))

# functional random intercept for spots for tissue 4
b8 <- data.frame(a = as.factor(Spot[Tissue == 4]))
Zlist[[8]][[1]] <- rBind(matrix(0, nrow = (3 / 4 * n),
  ncol = (nplevs) * nslevs), sparse.model.matrix(~ -1 + a, b8))

# smooth error E
b9 <- data.frame(a = as.factor(Obs))
Zlist[[9]][[1]] <- sparse.model.matrix(~ -1 + a, b9)

#####
# run estimation
#####
FLMM_spectra <- denseFLMM(Y = curves, gridpoints = wavelengths, Zlist = Zlist, G = G,
  Lvec = Lvec, groups = NA, Zvars = NA, L = 0.99999, NPC = NA, smooth = FALSE)

```

E.2 R add-on package sparseFLMM

This section provides a slightly modified version of the manual of the main functions of the R add-on package sparseFLMM, including example code.

E.2.1 Manual of R function sparseFLMM

Functional linear mixed models for irregularly or sparsely sampled data

Description

Estimation of functional linear mixed models (FLMMs) for irregularly or sparsely sampled data based on functional principal component analysis (FPCA). The implemented models are special cases of the general FLMM

$$Y_i(t_{ij}) = \mu(t_{ij}, \mathbf{x}_i) + \mathbf{z}_i^T \mathbf{U}(t_{ij}) + \varepsilon_i(t_{ij}), \quad i = 1, \dots, n, \quad j = 1, \dots, D_i,$$

with $Y_i(t_{ij})$ the value of the response of curve i at observation point t_{ij} , $\mu(t_{ij}, \mathbf{x}_i)$ is a mean function, which may depend on covariates \mathbf{x}_i . \mathbf{z}_i is a covariate vector, which is multiplied with the vector of functional random effects $\mathbf{U}(t_{ij})$. $\varepsilon_i(t_{ij})$ is independent and identically distributed white noise measurement error with homoscedastic, constant variance. For more details, see Chapters 2 to 5.

The current implementation can be used to fit three special cases of the above general FLMM:

- a model for independent functional data (e.g., longitudinal data), for which $\mathbf{z}_i^T \mathbf{U}(t_{ij})$ only consists of a smooth curve-specific deviation (smooth error curve)
- a model for correlated functional data with one functional random intercept (fRI) for one grouping factor in addition to a smooth curve-specific error
- a model for correlated functional data with two crossed fRIs for two grouping factors in addition to a smooth curve-specific error.

Usage

```
sparseFLMM(curve_info, use_RI = FALSE, use_simple = FALSE, method = "fREML",
  use_bam = TRUE, bs = "ps", d_grid = 100, min_grid = 0, max_grid = 1,
  my_grid = NULL, bf_mean = 8, bf_covariates = 8, m_mean = c(2, 3),
  covariate = FALSE, num_covariates, covariate_form, interaction,
  which_interaction = matrix(NA), save_model_mean = FALSE, para_estim_mean = FALSE,
  para_estim_mean_nc = 0, bf_covs, m_covs, use_whole = FALSE, use_tri = FALSE,
  use_tri_constr = TRUE, use_tri_constr_weights = FALSE, np = TRUE, mp = TRUE,
  use_discrete_cov = FALSE, para_estim_cov = FALSE, para_estim_cov_nc = 0,
```

```
var_level = 0.95, N_B = NA, N_C = NA, N_E = NA, use_famm = FALSE,
use_bam_famm = TRUE, bs_int_famm = list(bs = "ps", k = 8, m = c(2, 3)),
bs_y_famm = list(bs = "ps", k = 8, m = c(2, 3)), save_model_famm = FALSE,
use_discrete_famm = FALSE, para_estim_famm = FALSE, para_estim_famm_nc = 0)
```

Arguments

<code>curve_info</code>	<p>data table in which each row represents a single observation point. <code>curve_info</code> needs to contain the following columns:</p> <p><code>y_vec</code> (numeric): the response values for each observation point.</p> <p><code>t</code> (numeric): the observations point locations, i.e., t_{ij}.</p> <p><code>n_long</code> (integer): unique identification number for each curve.</p> <p><code>subject_long</code> (integer): unique identification number for each level of the first grouping factor (e.g., speakers for the speech production data in the example below). In the case of independent functions <code>subject_long</code> should be set equal to <code>n_long</code>.</p> <p>For models with two crossed functional random intercepts, the data table additionally needs to have columns:</p> <p><code>word_long</code> (integer): unique identification number for each level of the second grouping factor (e.g., words for the speech production data in the example below).</p> <p><code>combi_long</code> (integer): number of the repetition of the combination of the corresponding level of the first and of the second grouping factor.</p> <p>For models with covariates as part of the mean function $\mu(t_{ij}, \mathbf{x}_i)$, the covariate values (numeric) need to be in separate columns with names: <code>covariate.1</code>, <code>covariate.2</code>, etc.</p>
<code>use_RI</code>	<p>TRUE to specify a model with one functional random intercept for the first grouping factor (<code>subject_long</code>) and a smooth random error curve. Defaults to FALSE, which specifies a model with crossed functional random intercepts for the first and second grouping factor and a smooth error curve.</p>
<code>use_simple</code>	<p>TRUE to specify a model with only a smooth random error function, argument <code>use_RI</code> should then also be set to TRUE. Defaults to FALSE.</p>
<code>method</code>	<p>estimation method for <code>gam</code> or <code>bam</code>, see package <code>mgcv</code> for more details. Defaults to "fREML".</p>
<code>use_bam</code>	<p>TRUE to use function <code>bam</code> in the package <code>mgcv</code> instead of function <code>gam</code> (syntax is the same, <code>bam</code> is faster for large data sets). Function <code>bam</code> is recommended and set as default.</p>

<code>bs</code>	spline basis function type for the estimation of the mean function and the auto-covariance, see functions <code>s</code> and <code>te</code> in the package <code>mgcv</code> for more details. Defaults to penalized B-splines, i.e., <code>bs = "ps"</code> . This choice is recommended as others have not been tested yet.
<code>d_grid</code>	pre-specified grid length for equidistant grid on which the mean, the auto-covariance surfaces, the eigenfunctions and the functional random effects are evaluated. NOTE: the length of the grid can be important for computation time (approx. quadratic influence). Defaults to <code>d_grid = 100</code> .
<code>min_grid</code>	minimum value of equidistant grid (should approx. correspond to minimum value of time interval). Defaults to <code>min_grid = 0</code> .
<code>max_grid</code>	maximum value of equidistant grid (should approx. correspond to maximum value of time interval). Defaults to <code>max_grid = 1</code> .
<code>my_grid</code>	optional evaluation grid, which can be specified and used instead of <code>d_grid</code> , <code>min_grid</code> , <code>max_grid</code> . NOTE: the grid should be equidistant.
<code>bf_mean</code>	basis dimension (number of basis functions) used for the functional intercept $f_0(t_{ij})$ in the mean estimation via function <code>bam</code> or <code>gam</code> in the package <code>mgcv</code> . Defaults to <code>bf_mean = 8</code> .
<code>bf_covariates</code>	basis dimension (number of basis functions) used for the functional effects of covariates in the mean estimation via <code>bam/gam</code> . Defaults to <code>bf_covariates = 8</code> . NOTE: in the current implementation, the same basis dimension for all covariates is used.
<code>m_mean</code>	order of the penalty for this term in <code>bam/gam</code> of mean estimation, for <code>bs = "ps"</code> spline and penalty order, defaults to <code>m_mean = c(2, 3)</code> , i.e., cubic B-splines with third order difference penalty, see function <code>s</code> in the package <code>mgcv</code> for details.
<code>covariate</code>	TRUE to estimate covariate effects (as part of the mean function).
<code>num_covariates</code>	number of covariates that are included in the model. NOTE: not number of effects in case interactions of covariates are specified.
<code>covariate_form</code>	vector with entries for each covariate that specify the form in which the respective covariate enters the mean function. Possible forms are "by" for varying coefficient ($f(t_{ij}) \cdot covariate$), which is possible for dummy coded covariates and metric covariates and "smooth" for smooth effect in the function argument and in the covariate ($f(t_{ij}, covariate)$), which is only possible for metric covariates! NOTE: metric covariates should be centered such that the global functional intercept $f_0(t_{ij})$ can be interpreted as global mean function and the effect can be interpreted as difference from the global mean.

<code>interaction</code>	TRUE to estimate interaction effects of covariates, which interactions, see argument <code>which_interaction</code> (below). Interactions are possible for dummy-coded covariates that act as varying coefficients.
<code>which_interaction</code>	symmetric matrix that specifies which interactions should be considered in case <code>covariate = TRUE</code> and <code>interaction = TRUE</code> . Entry <code>which_interaction[k, l]</code> specifies that the interaction between <code>covariate.k</code> and <code>covariate.l</code> is modeled (example below). NOTE: entries are redundant, <code>which_interaction[l, k]</code> should be set to the same as <code>which_interaction[k, l]</code> (symmetric). Defaults to <code>which_interaction = matrix(NA)</code> , which should be used when <code>interaction = FALSE</code> .
<code>save_model_mean</code>	TRUE to give out the <code>gam/bam</code> object (attention: can be large!), defaults to <code>FALSE</code> .
<code>para_estim_mean</code>	TRUE to parallelize mean estimation (only possible using <code>bam</code>), defaults to <code>FALSE</code> .
<code>para_estim_mean_nc</code>	number of cores for parallelization of mean estimation (only possible using <code>bam</code> , only active if <code>para_estim_mean = TRUE</code>). Defaults to 0.
<code>bf_covs</code>	vector of marginal basis dimensions (number of basis functions) used for covariance estimation via <code>bam/gam</code> for each functional random effect (including the smooth error curve). In the case of multiple grouping factors, the first entry corresponds to the first grouping factor, the second vector entry corresponds to the second grouping factor, and the third to the smooth error curve.
<code>m_covs</code>	list of marginal orders of the penalty for <code>bam/gam</code> for covariance estimation, for <code>bs = "ps"</code> marginal spline and penalty order. As only symmetric surfaces are considered: same for both directions. For crossed fRIs: list of three vectors, e.g., <code>m_covs = list(c(2, 3), c(2, 3), c(2, 3))</code> , where the first and second entry correspond to first and second grouping factor, respectively and the third entry corresponds to the smooth error. For one fRI: list of two vectors, e.g., <code>m_covs = list(c(2, 3), c(2, 3))</code> , where first entry corresponds to (first) grouping factor and second entry corresponds to smooth error. For independent curves: list of one vector, e.g., <code>m_covs = list(c(2,3))</code> corresponding to smooth error.
<code>use_whole</code>	TRUE to estimate the whole auto-covariance surfaces without symmetry constraint. Defaults to <code>FALSE</code> as is much slower than <code>use_tri_constr</code> and <code>use_tri_constr_weights</code> . For more details, see Chapter 5.

<code>use_tri</code>	TRUE to estimate only the upper triangle of the auto-covariance surfaces without symmetry constraint. Defaults to FALSE and it is not recommended. Note that <code>use_tri</code> should only be used for research purposes. For more details, see Chapter 5.
<code>use_tri_constr</code>	TRUE to estimate only the upper triangle of the auto-covariance surfaces with symmetry constraint using the smooth class <code>symms.mooth</code> . Defaults to TRUE. For more details, see Chapter 5.
<code>use_tri_constr_weights</code>	TRUE to estimate only the upper triangle of the auto-covariances with symmetry constraint, using the smooth class <code>symm.smooth</code> and weights of 0.5 on the diagonal to use the same weights as for estimating the whole auto-covariance surfaces using <code>use_whole = 1</code> . Defaults to FALSE. For more details, see Chapter 5.
<code>np</code>	TRUE to use ‘normal parameterization’ for a tensor product smooth, see function <code>te</code> in the package <code>mgcv</code> for more details. Defaults to TRUE.
<code>mp</code>	FALSE to use Kronecker product penalty instead of Kronecker sum penalty with only one smoothing parameter (for <code>use_whole = TRUE</code> and <code>use_tri = TRUE</code>), for details, see function <code>te</code> in the package <code>mgcv</code> . For <code>use_tri_constr = TRUE</code> and <code>use_tri_constr_weights = TRUE</code> , only one smoothing parameter is estimated anyway. Defaults to TRUE.
<code>use_discrete_cov</code>	TRUE to further speed up the auto-covariance computation by discretization of the covariate values for storage and efficiency reasons. It includes parallelization controlled by argument <code>para_estim_cov_nc</code> (below), see function <code>bam</code> in the package <code>mgcv</code> for more details. Defaults to FALSE.
<code>para_estim_cov</code>	TRUE to parallelize auto-covariance estimation (only possible using <code>bam</code>), defaults to FALSE.
<code>para_estim_cov_nc</code>	number of cores (if <code>use_discrete_cov = FALSE</code>) or number of threads (if <code>use_discrete_cov = TRUE</code>) for parallelization of auto-covariance estimation (only possible using <code>bam</code> , only active if <code>para_estim_cov = TRUE</code>). Defaults to 0.
<code>var_level</code>	pre-specified level of explained variance used for the choice of the number of the functional principal components (FPCs). Alternatively, a specific number of FPCs can be specified (see below). Defaults to <code>var_level = 0.95</code> .
<code>N_B</code>	number of components to keep for the fRI for the first grouping factor, overrides <code>var_level</code> if not NA.
<code>N_C</code>	number of components to keep for the fRI for the second grouping factor, overrides <code>var_level</code> if not NA.

<code>N_E</code>	number of components to keep for the smooth error, overrides <code>var_level</code> if not NA.
<code>use_famm</code>	TRUE to embed the model into the framework of functional additive mixed models using re-estimation of the mean function together with the prediction of the FPC weights (scores). This allows for point-wise confidence bands for the covariate effects. Defaults to FALSE.
<code>use_bam_famm</code>	TRUE to use function <code>bam</code> instead of function <code>gam</code> in the FAMM estimation (reduces computation time for large data sets), highly recommended. Defaults to TRUE.
<code>bs_int_famm</code>	specification of the estimation of the functional intercept $f_0(t_{ij})$ (as part of the mean function), see function <code>pffr</code> in the package <code>refund</code> (Huang et al., 2016a) for details. Defaults to <code>bs_int = list(bs = "ps", k = 8, m = c(2, 3))</code> , where <code>bs</code> : type of basis functions, <code>k</code> : number of basis functions, <code>m</code> : order of the spline and order of the penalty.
<code>bs_y_famm</code>	specification of the estimation of the covariates effects (as part of the mean function), see function <code>pffr</code> in the package <code>refund</code> for details. Defaults to <code>bs_y_famm = list(bs = "ps", k = 8, m = c(2, 3))</code> , where <code>bs</code> : type of basis functions, <code>k</code> : number of basis functions, <code>m</code> : order of the spline and order of the penalty.
<code>save_model_famm</code>	TRUE to give out the FAMM model object (attention: can be very large!). Defaults to FALSE.
<code>use_discrete_famm</code>	TRUE to further speed up the FPC-FAMM computation by discretization of the covariate values for storage and efficiency reasons. It includes parallelization controlled by argument <code>para_estim_famm_nc</code> (below), see function <code>bam</code> in the package <code>mgcv</code> for more details. Defaults to FALSE.
<code>para_estim_famm</code>	TRUE to parallelize the FAMM estimation. Defaults to FALSE.
<code>para_estim_famm_nc</code>	number of cores (if <code>use_discrete_famm = FALSE</code>) or number of threads (if <code>use_discrete_famm = TRUE</code>) for parallelization of FAMM estimation (only possible using <code>bam</code> , only active if <code>para_estim_famm = TRUE</code>). Defaults to 0.

Details

The code can handle irregularly and possibly sparsely sampled data. Of course, it can also be used to analyze regular grid data, but as it is especially designed for the irregular case and there may be a more efficient way to analyze regular grid data.

The mean function is of the form

$$\mu(t_{ij}, \mathbf{x}_i) = f_0(t_{ij}) + \sum_{k=1}^r f_k(t_{ij}, \tilde{\mathbf{x}}_{ik}),$$

where $f_0(t_{ij})$ is a functional intercept and $\tilde{\mathbf{x}}_{ik}$ is a subvector of the vector of covariates \mathbf{x}_{ik} . Currently implemented are effects of dummy-coded and metric covariates x_i which act as varying coefficients of the form $f_k(t_{ij}) \cdot x_i$ and smooth effects of metric covariates (smooth in t_{ij} and in the covariate x_i) of the form $f(t_{ij}, x_i)$. NOTE: metric covariates should be centered such that the global functional intercept can be interpreted as global mean function and the effect can be interpreted as difference from the global mean. Interaction effects of dummy-coded covariates acting as varying coefficients are possible.

The estimation consists of four main steps:

1. Estimation of the smooth mean function (including covariate effects) under independence assumption using splines.
2. Estimation of the smooth auto-covariances of the functional random effects. A fast bivariate symmetric smoother implemented in the smooth class `symm.smooth` can be used to speed up estimation (see below).
3. Eigen decomposition of the estimated auto-covariances, which are evaluated on a pre-specified equidistant grid. This yields estimated eigenvalues and eigenfunctions, which are rescaled to ensure orthonormality with respect to the L^2 -inner product.
4. Prediction of the functional principal component weights (scores) yielding predictions for the functional random effects.

The estimation of the mean function and auto-covariance functions is based on the package `mgcv` (Wood, 2006, 2011). The functional principal component weights (scores) are predicted as best (linear) unbiased predictors. In addition, this implementation allows to embed the model in the general framework of functional additive mixed models (FAMM; Scheipl et al., 2016a) based on the package `refund`, which allows for the construction of point-wise confidence bands for covariate effects (in the mean function) conditional on the FPCA. Note that the estimation as FAMM may be computationally expensive as the model is re-estimated in a mixed model framework.

The three special cases of the general FLMM (two crossed fRIs, one fRI, independent curves) are implemented as follows:

- In the special case with two crossed fRIs, three random processes named B , C , and E are considered, where B is the fRI for the first grouping factor (e.g., speakers in the speech production example below), C denotes the fRI for the second grouping factor (e.g., target words in the speech production example below) and E denotes the smooth error. For this special case, arguments `use_RI` and `use_simple` are both set to `FALSE`.

- In the special case with only one fRI, only B and E are considered and the number of levels for the second grouping factor is to zero. For this special case, argument `use_RI` is set to `TRUE` and argument `use_simple` is set to `FALSE`.
- The special case with independent curves is internally seen as a special case of the model with one fRI for the first grouping factor, with the number of levels for this grouping factor corresponding to the number of curves. Thus, for each level of the first grouping factor there is one curve. Therefore, for the special case of independent curves, the estimation returns an estimate for the auto-covariance of B (instead of E) and all corresponding results are indicated with ‘ B ’, although they correspond to the smooth error. For this special case, arguments `use_RI` and `use_simple` are both set to `TRUE`.

Value

The function returns a list of two elements: `time_all` and `results`. `time_all` contains the total `system.time()` for calling function `sparseFLMM`. `results` is a list, including:

`my_grid` pre-specified evaluation grid.

`mean_hat` including the components of the estimated mean function:

`mean_pred`: contains effects of dummy covariates or metric covariates with a linear effect (varying coefficients).

`mean_pred_smooth`: contains effects of metric covariates with a smooth effect.

`intercept`: is the estimated intercept, which is part of $f_0(t_{ij})$.

For each auto-covariance smoothing alternative X in `{use_whole, use_tri, use_tri_constr, use_tri_constr_weights}` the list `results` additionally contains:

`cov_hat_<X>` including:

`sigmasq`: the estimated error variance.

`sigmasq_int`: the integral of the estimated error variance over the domain.

`grid_mat_<B/C/E>`: the estimated auto-covariance(s) evaluated on the pre-specified grid.

`sp`: the smoothing parameter(s) for smoothing the auto-covariance(s).

`time_cov_estim`: the time for the smoothing the auto-covariance(s) only.

`time_cov_pred_grid`: the time for evaluating the estimated auto-covariance(s) on the pre-specified grid.

`time_cov_<X>`: the total time for the auto-covariance estimation.

`fpc_hat_<X>` including:

`phi_<B/C/E>_hat_grid`: the estimated rescaled eigenfunctions evaluated on the pre-specified grid.

`nu_<B/C/E>_hat`: the estimated rescaled eigenvalues.

`N_<B/C/E>`: the estimated truncation numbers, i.e., number of FPCs which are kept.

`total_var`: the estimated total variance.

`var_explained`: the estimated explained variance.

`xi_<B/C/E>_hat`: the predicted FPC weights (scores).

`time_fpc_<X>`: the total time for the eigen decompositions and prediction on the FPC weights (scores).

If `use_famm = TRUE`, the list `results` additionally contains:

`fpc_famm_hat_<X>` including:

`intercept`: the estimated intercept, which is part of $f_0(t_{ij})$.

`residuals`: the residuals of the FAMM estimation.

`xi_<B/C/E>_hat_famm`: the predicted basis weights.

`famm_predict_<B/C/E>`: the predicted functional processes evaluated on the pre-specified grid.

`famm_cb_mean`: the re-estimated functional intercept $f_0(t_{ij})$.

`famm_cb_covariate.1`, `famm_cb_covariate.2`, etc: possible re-estimated covariate effects.

`famm_cb_inter_1.2`: `famm_cb_inter_1.3`, etc: possible interaction effects.

`time_fpc_famm_<X>`: the total `system.time()` for the FAMM estimation.

The unique identification numbers for the levels of the grouping factors and curves are renumbered for convenience during estimation from 1 in ascending order. The original identification numbers are returned in the list `results`:

`n_orig` curve levels as they entered the estimation.

`subject_orig` levels of the first grouping factor as they entered the estimation.

`word_orig` levels of the second grouping factor (if existent) as they entered the estimation.

Author(s)

Jona Cederbaum

References

Cederbaum, J., Pouplier, M., Hoole, P., and Greven, S. (2016): Functional linear mixed models for irregularly or sparsely sampled data. *Statistical Modelling*, 16(1):67–88.

Cederbaum, J., Scheipl, F., and Greven, S. (2018): Fast symmetric additive covariance smoothing. *Computational Statistics & Data Analysis*, 120:25–41.

Scheipl, F., Staicu, A.-M., and Greven, S. (2015): Functional additive mixed models, *Journal of Computational and Graphical Statistics*, 24(2):477–501.

See Also

Note that function `sparseFLMM` calls function `bam` or `gam` in the package `mgcv` directly.

For functional linear mixed models with complex correlation structures for data sampled on equal grids based on functional principal component analysis, see function `denseFLMM` in the package `denseFLMM`.

Examples

```
# subset of speech production acoustic data (very small subset, no meaningful results
# can be expected and FMM estimation does not work for this subset example.
# For FMM estimation, see below.)
data("acoustic_subset")

acoustic_results <- sparseFLMM(curve_info = acoustic_subset, use_RI = FALSE,
  use_simple = FALSE, method = "FREML", use_bam = TRUE, bs = "ps", d_grid = 100,
  min_grid = 0, max_grid = 1, my_grid = NULL, bf_mean = 8, bf_covariates = 8,
  m_mean = c(2,3), covariate = TRUE, num_covariates = 4, covariate_form = rep("by", 4),
  interaction = TRUE, which_interaction = matrix(c(FALSE, TRUE, TRUE, TRUE, TRUE, FALSE,
  FALSE, FALSE, TRUE, FALSE, FALSE, FALSE, TRUE, FALSE, FALSE, FALSE, FALSE, FALSE,
  FALSE, FALSE, FALSE, FALSE, FALSE), byrow = TRUE, nrow = 4, ncol = 4),
  save_model_mean = FALSE, para_estim_mean = FALSE, para_estim_mean_nc = 0,
  bf_covs = c(5, 5, 5), m_covs = list(c(2, 3), c(2, 3), c(2, 3)), use_whole = FALSE,
  use_tri = FALSE, use_tri_constr = TRUE, use_tri_constr_weights = FALSE, np = TRUE,
  mp = TRUE, use_discrete_cov = FALSE, para_estim_cov = FALSE, para_estim_cov_nc = 5,
  var_level = 0.95, N_B = NA, N_C = NA, N_E = NA, use_famm = FALSE, use_bam_famm = TRUE,
  bs_int_famm = list(bs = "ps", k = 8, m = c(2, 3)), bs_y_famm = list(bs = "ps", k = 8,
  m = c(2, 3)), save_model_famm = FALSE, use_discrete_famm = FALSE, para_estim_famm = FALSE,
  para_estim_famm_nc = 0)
```

E.2.2 Manual of R function `make_summation_matrix`

Construct symmetry constraint matrix for bivariate symmetric smoothing

Description

This function can be used to construct a symmetry constraint matrix that imposes a symmetry constraint on spline coefficients in symmetric bivariate smoothing problems and is especially designed for constructing objects of the class `symm.smooth`, see function `smooth.construct.symm.smooth.spec`.

Usage

```
make_summation_matrix(F)
```

Arguments

`F` number of marginal basis functions.

Details

Imposing a symmetry constraint to the spline coefficients in order to obtain a reduced coefficient vector is equivalent to right multiplication of the bivariate design matrix with the symmetry constraint matrix obtained with function `make_summation_matrix`. The penalty matrix of the bivariate smooth needs to be adjusted to the reduced coefficient vector by left and right multiplication with the symmetry constraint matrix. This function is used in the constructor function `smooth.construct.symm.smooth.spec`.

Value

A symmetry constraint matrix of dimension $F^2 \times F(F + 1)/2$.

References

Cederbaum, J., Scheipl, F., and Greven, S. (2018): Fast symmetric additive covariance smoothing. *Computational Statistics & Data Analysis*, 120:25–41.

See Also

See functions `smooth.construct` and `smoothCon` in the package `mgcv` for details on constructors.

Examples

```
make_summation_matrix(F = 2)
```

```
      [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    1    0
[4,]    0    0    1
```

```
make_summation_matrix(F = 3)
```

```
      [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1    0    0    0    0    0
[2,]    0    1    0    0    0    0
[3,]    0    0    1    0    0    0
[4,]    0    1    0    0    0    0
[5,]    0    0    0    1    0    0
[6,]    0    0    0    0    1    0
[7,]    0    0    1    0    0    0
[8,]    0    0    0    0    1    0
[9,]    0    0    0    0    0    1
```

E.2.3 Manual of R function `smooth.construct.symm.smooth.spec`

Symmetric bivariate smooths constructor

Description

The `symm.smooth` class is a new smooth class that is appropriate for symmetric bivariate smooths, e.g., of covariance functions, using tensor-product smooths in a `gam` formula using the package `mgcv`. A symmetry constraint matrix is constructed (see function `make_summation_matrix`) to impose a symmetry constraint on the spline coefficients, which considerably reduces the number of coefficients that have to be estimated.

Usage

```
## S3 method for class 'symm.smooth.spec'
smooth.construct(object, data, knots)
```

Arguments

- `object` smooth specification object or a smooth object.
- `data` data frame, model frame or list containing the values of the (named) covariates at which the smooth term is to be evaluated.
- `knots` optional data frame supplying any knot locations to be supplied for basis construction.

Details

The underlying procedure is the following:

1. The marginal spline design matrices and the corresponding marginal penalties are built.
2. The tensor product of the marginal design matrices and the bivariate penalty matrix are built.
3. The constraint matrix is applied to the tensor product design matrix and to the penalty matrix.

Value

An object of class `symm.smooth`. See function `smooth.construct` in the package `mgcv` for the elements it will contain.

References

Cederbaum, J., Scheipl, F., and Greven, S. (2018): Fast symmetric additive covariance smoothing. *Computational Statistics & Data Analysis*, 120:25–41.

See Also

See functions `smooth.construct` and `smoothCon` in the package `mgcv` for details on constructors.

E.2.4 Manual of R function `Predict.matrix.symm.smooth`

Predict matrix method for symmetric bivariate smooths

Description

Predict matrix method for symmetric bivariate smooths.

Usage

```
## S3 method for class 'symm.smooth'  
Predict.matrix(object, data)
```

Arguments

- `object` `symm.smooth` object created by function `smooth.construct.symm.smooth.spec`, see function `smooth.construct` in the package `mgcv`.
- `data` see function `smooth.construct` in the package `mgcv`.

See Also

`Predict.matrix` and `smoothCon` in the package `mgcv` for details on constructors.

List of Figures

3.1	Non-standardized diffuse reflectance spectra for cortical bone (upper left), nerves (upper right), salivary glands (bottom left), and cancellous bone (bottom right) colored by pig.	54
3.2	Left: Point-wise mean curves per tissue and overall mean curve. Right: Point-wise variance curves per tissue and overall variance curve.	55
3.3	Tissue-specific means (solid black) plus (+) and minus (−) a suitable multiple of the first principal component of $B_{\tau p}(t)$ for cortical bone (upper left), nerves (upper right), salivary glands (bottom left), cancellous bone (bottom right). The respective amount of explained variability is given in brackets.	57
4.1	Index curves of the consonant assimilation data over time. Left [right]: Curves of order /s#sh/[sh#s/]. Positive values approaching +1 indicate a reference /s/ [/sh/] acoustic pattern, while negative values approaching −1 indicate a reference /sh/ [/s/] acoustic pattern.	76
4.2	Left: Effect of covariate order (red solid line) with point-wise confidence bands (dashed lines). Right: Mean function (solid line) and the effect of adding (+) and subtracting (−) a suitable multiple ($2\sqrt{\hat{\nu}_1^B}$) of the first FPC for speakers.	77
4.3	True and estimated FPCs of the crossed fRIs $B_i(t)$ and $C_j(t)$ (top row), as well as of the smooth error $E_{ijh}(t)$ (bottom row). Shown are the true functions (red solid line), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	80
5.1	Acoustic signal curves of the speech production data over time (cf. Chapter 4). Left [right]: Signal curves of consonant order /s#sh/ [sh#s] colored by target word. Example curves with no/little assimilation and with strong assimilation highlighted (black).	103
5.2	Results for the fRI for speakers using the four smoothing methods. Top row: Estimated covariance surfaces. Middle row: Contours of the estimated covariance surfaces. Bottom row: Estimated eigenfunctions $\hat{\phi}_k^B(t)$ based on the entire data set (black) and jackknife estimates with one of the speakers left out in turn (gray).	105
5.3	Boxplots of the rrMSEs (log10 scale at y-axis) for the scenario with independent curves (Scenario 1) for the smoothing methods being compared. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$	109

5.4	Boxplots of the rrMSEs for the scenario with crossed fRIs (Scenario 2) for the smoothing methods being compared. Shown are boxplots of the rrMSEs for the three auto-covariances $K^B(t, t')$, $K^C(t, t')$, and $K^E(t, t')$ as well as for the error variance σ^2 and the curves $Y_i(t)$	110
5.5	Computation times (log10 scale at y-axis) for 200 simulation runs for all compared methods. Left: Computation times for Setting 1 of the scenario with independent curves. Right: Computation times for the scenario with crossed fRIs.	111
C.1	Covariate mean effects (red solid lines) with conditional point-wise confidence bands (dashed lines). Upper: Reference mean $f_0(t)$. Middle (from left to right): Covariate effects of covariates stress1, stress2, vowel. Lower (from left to right): Interaction effects of order and stress1, order and stress2, order and vowel.	141
C.2	Second FPC for speakers. Shown is the mean function (solid line) and the effect of adding (+) and subtracting (-) a suitable multiple ($2\sqrt{\hat{v}_2^B}$) of the first FPC for speakers.	141
C.3	True and estimated mean functions. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	143
C.4	Boxplots of the estimated eigenvalues of the auto-covariances of the crossed fRIs (top row), as well as the eigenvalues of the auto-covariance of the smooth error (bottom row) for all 200 simulations runs.	144
C.5	Boxplot of the estimated error variances σ^2 for all 200 simulation runs.	145
C.6	True and estimated covariate and interaction effects estimated using the independence assumption. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	146
C.7	True and estimated covariate and interaction effects estimated using FPC-FAMM. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	147
C.8	True and estimated covariate and interaction effects estimated using spline-FAMM. Shown are the true function (red), the mean of the estimated functions over 100 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 100 simulation runs (grey).	147
C.9	Average point-wise coverage of the point-wise CBs obtained by FPC-FAMM for all covariate and interaction effects. For each effect, the point-wise coverage averaged over 200 simulation runs (black line) is shown. The red line indicates the nominal value of 0.95.	148
C.10	Average point-wise coverage of the point-wise CBs obtained by spline-FAMM for all covariate and interaction effects. For each effect, the point-wise coverage averaged over 100 simulation runs (black line) is shown. The red line indicates the nominal value of 0.95.	148
C.11	True and estimated FPCs of the fRI (top row) and of the smooth error (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	149
C.12	Boxplots of the estimated eigenvalues of the auto-covariances of the fRI (top row), and of the smooth error (bottom row) for all 200 simulations runs.	149

C.13	Boxplot of the estimated error variances σ^2 for all 200 simulation runs.	150
C.14	True and estimated mean function $\mu(t, \mathbf{x}_{ijh})$. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	150
C.15	True and estimated FPCs of the crossed fRIs $B_i(t)$ (top row) and $C_j(t)$ (middle row), as well as the FPCs of the smooth error $E_{ijh}(t)$ (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	151
C.16	Boxplots of the estimated eigenvalues of the auto-covariances of the crossed fRIs $B_i(t)$ (top row), $C_j(t)$ (middle row), as well as the eigenvalues of the auto-covariance of the smooth error $E_{ijh}(t)$ (bottom row) for all 200 simulations runs.	152
C.17	Boxplot of the estimated error variances σ^2 for all 200 simulation runs.	152
C.18	True and estimated mean function $\mu(t, \mathbf{x}_{ijh})$. Shown is the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	154
C.19	True and estimated FPCs of the crossed fRIs $B_i(t)$ and $C_j(t)$ (top row), as well as the FPCs of the smooth error $E_{ijh}(t)$ (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	154
C.20	Boxplots of the estimated eigenvalues of the auto-covariances of the crossed fRIs $B_i(t)$ and $C_j(t)$ (top row), as well as the eigenvalues of the auto-covariance of the smooth error $E_{ijh}(t)$ (bottom row) for all 200 simulations runs.	155
C.21	Boxplot of the estimated error variances σ^2 for all 200 simulation runs.	155
C.22	True and estimated covariate and interaction effects estimated based on the independence assumption. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	156
C.23	True and estimated covariate and interaction effects using FPC-FAMM. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	157
C.24	Average point-wise coverage of the point-wise CBs obtained by FPC-FAMM for all covariate and interaction effects. For each effect, the point-wise coverage averaged over 200 simulation runs (black line) is shown. The red line indicates the nominal value of 0.95.	157
C.25	True and estimated FPCs of the fRI $B_i(t)$ (top row) and of the smooth error (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	158
C.26	Boxplots of the estimated eigenvalues of the auto-covariances of the fRI $B_i(t)$ (top row), and of the smooth error (bottom row) for all 200 simulations runs.	159
C.27	Boxplot of the estimated error variances σ^2 for all 200 simulation runs.	159

C.28	True and estimated mean function $\mu(t, \mathbf{x}_{ijh})$. Shown are the true function (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	160
C.29	True and estimated FPCs of the crossed fRIs $B_i(t)$ (top row) and $C_j(t)$ (middle row), as well as the FPCs of the smooth error $E_{ijh}(t)$ (bottom row). Shown are the true functions (red), the mean of the estimated functions over 200 simulation runs (black dashed line), the point-wise 5th and 95th percentiles of the estimated functions (blue dashed lines), and the estimated functions of all 200 simulation runs (grey).	161
C.30	Boxplots of the estimated eigenvalues of the auto-covariances of the crossed fRIs $B_i(t)$ (top row), $C_j(t)$ (middle row), as well as the eigenvalues of the auto-covariance of the smooth error $E_{ijh}(t)$ (bottom row) for all 200 simulations runs.	162
C.31	Boxplot of the estimated error variances σ^2 for all 200 simulation runs.	162
D.1	Square root of observed CD4 cell count trajectories plotted against the months since SC. Shown are the trajectories of 366 HIV infected subjects. Some trajectories are highlighted for better display and an estimated smooth mean function (dashed) is shown.	170
D.2	Results for the curve-specific fRI for TRI-CONST, TRI-CONST-W, WHOLE, and TRI. Top row: Estimated covariance surfaces. Middle row: Contours of the estimated covariance surfaces. Bottom row: Estimated corresponding eigenfunctions.	172
D.3	Results for the curve-specific fRI for FACE and FACE-STEP-1. Top row: Estimated covariance surfaces. Middle row: Contours of the estimated covariance surfaces. Bottom row: Estimated corresponding eigenfunctions.	173
D.4	Results for the smooth error curve $E_i(t)$ using the four smoothing methods. Top row: Estimated covariance surfaces. Middle row: Contours of the estimated covariance surfaces. Bottom row: Estimated eigenfunctions $\phi_k^E(t)$ based on the entire data set (black) and jackknife estimates with one of the speakers left out in turn (gray).	175
D.5	Auto-covariances $K^B(t, t')$, $K^C(t, t')$, and $K^E(t, t')$ and their eigenfunctions used for the data generation for the scenario with crossed fRIs. Top row: Covariance surfaces. Middle row: Contours of the covariance surfaces. Bottom row: Corresponding eigenfunctions $\phi_k^B(t)$, $\phi_k^C(t)$, and $\phi_k^E(t)$	176
D.6	Mean function used for the data generation for the scenario with crossed fRIs.	177
D.7	Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 2 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E , ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E , ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$	179
D.8	Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 3 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E , ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E , ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$	180

D.9 Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 4 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$ 181

D.10 Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 5 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$ 182

D.11 Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 6 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$ 183

D.12 Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 7 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$ 184

D.13 Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 8 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$ 185

D.14 Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 9 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$ 186

D.15 Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 10 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$ 187

D.16 Boxplots of the rrMSEs (log10 scale at y-axis) for Setting 11 of the scenario with independent curves. Top row: rrMSEs for auto-covariance $K^E(t, t')$, error variance σ^2 , and the first eigenfunction $\phi_1^E(t)$. Second row: rrMSEs for the second eigenfunction $\phi_2^E(t)$ and eigenvalues ν_1^E, ν_2^E . Third row: rrMSEs for the random basis weights ξ_1^E, ξ_2^E and process $E_i(t)$. Bottom row: rrMSEs for curves $Y_i(t)$ 188

D.17 Boxplots of the rrMSEs for the crossed fRIs setting. Shown are boxplots for all remaining model components, which are not shown in Section 5.7.2 of Chapter 5: The rrMSEs for the eigenfunctions and eigenvalues of the three auto-covariances $K^B(t, t')$, $K^C(t, t')$, and $K^E(t, t')$, as well as the rrMSEs for the corresponding random basis weights. 190

List of Tables

2.1	Overview of possible fixed effects of scalar covariates that can be included in the mean function $\mu(t_{ij}, \mathbf{x}_i)$ in the FLMM (2.8). This overview table is based on Scheipl et al. (2015).	26
3.1	Variance decomposition for an FLMM with a functional random intercept for tissue type with pre-specified $L = 0.99999$. The model is fit to all reflectance spectra.	58
3.2	Classification results based on the FLMM (3.7) (block 1), with combined replications (block 2), based on Stelzle et al. (2011) (block 3), and based on PDA as in Engelhardt et al. (2014) (block 4). Shown are the numbers of classifications for each tissue type as well as the proportion of classifications per true class.	60
4.1	rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSE for $B_i(t)$, $C_j(t)$, and $E_{ijh}(t)$ and their covariance decompositions. Last row: Average rrMSEs for $Y_{ijh}(t)$, $\mu(t, \mathbf{x}_{ijh})$, and σ^2	80
C.1	Estimated eigenvalues and estimated error variance for the model with covariates in the following order: Estimated eigenvalue corresponding to the first and second FPCs for speakers, estimated eigenvalue corresponding to the first, second, and third FPCs for the smooth error, estimated error variance.	140
C.2	Estimated variance explained in percent of the estimated total variance given in the following order: Explained variance by the first and second FPCs for speakers, explained variance by the first, second, and third FPCs for the smooth error, estimated error variance.	140
C.3	Estimated variance explained in percent of the estimated total variance given in the following order: Explained variance by the first and second FPCs for speakers, explained variance by the first FPC for target words, estimated variance by the first, second, and third FPCs for the smooth error, estimated error variance.	140
C.4	Point-wise coverage of the point-wise CBs for FPC-FAMM and for spline-FAMM. Shown is the coverage averaged over all time points for all covariate and interaction effects. For FPC-FAMM, the coverage refers to 200 simulation runs, whereas for splines-FAMM, 100 simulation runs are taken into account.	145
C.5	Simultaneous coverage of the point-wise CBs for FPC-FAMM and for spline-FAMM. Shown is the average proportion of completely covered curves for all covariate and interaction effects. For FPC-FAMM, the coverage refers to 200 simulation runs, whereas for splines-FAMM, 100 simulation runs are taken into account.	146

C.6	rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSEs for the fRI and the smooth error. Last row: Average rrMSEs for the functional response, the mean, and the error variance.	146
C.7	Average rrMSEs for the estimated mean and covariate effects. Rows 1-2: rrMSEs for the estimation based on the independence assumption and for the estimation using FPC-FAMM averaged over 200 simulation runs. Last row: rrMSEs for the spline-based alternative averaged over 100 simulation runs.	146
C.8	rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average relative errors for the functional random effects and their covariance decompositions. Last row: Average rrMSE of the functional response, the mean, and the error variance.	150
C.9	rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSE for the functional random effects. Last row: Average rrMSE for the functional response, the mean, and the error variance.	153
C.10	Point-wise coverage of the point-wise CBs for FPC-FAMM. Shown is the coverage averaged over all time points for all covariate and interaction effects. The coverage refers to 200 simulation runs.	156
C.11	Simultaneous coverage of the point-wise CBs obtained by FPC-FAMM. Shown is the proportion of completely covered curves for all covariate and interaction effects. The coverage refers to 200 simulation runs.	157
C.12	rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSEs for the fRI and the smooth error. Last row: Average rrMSEs for the functional response, the mean, and the error variance.	158
C.13	Average rrMSEs for the estimated mean and covariate effects for the estimation using the independence assumption (first row) and using FPC-FAMM (second row) averaged over 200 simulation runs.	158
C.14	rrMSEs averaged over 200 simulation runs for all model components by random process. Rows 1-3: Number of grouping levels L^X and average rrMSEs for the random processes. Last row: Average rrMSEs for the functional response, the mean, and the error variance.	160
D.1	Truncated estimated eigenvalues and estimated error variance for all compared methods.	171
D.2	Truncated estimated eigenvalues of $K^B(t, t')$, $\hat{\nu}_k^B \cdot 10^3$, and of $K^E(t, t')$, $\hat{\nu}_k^E \cdot 10^3$, and estimated error variance $\hat{\sigma}^2 \cdot 10^3$ for all compared methods.	174
D.3	Eigenvalues $\nu_k^X \cdot 10^3$, $X \in \{B, C, E\}$, and error variance $\sigma^2 \cdot 10^3$ used for data generation for the scenario with crossed fRIs.	177
D.4	Specification of the eleven considered settings for the scenario with independent curves. The simple eigenfunctions are given as $\{\phi_1(t) = 1, \phi_2(t) = \sqrt{3}(2t - 1)\}$, the complex eigenfunctions are given as $\{\phi_1(t) = \sin(2\pi t), \phi_2(t) = \cos(2\pi t)\}$. The results for Setting 1 are shown in Section 5.7.2 of Chapter 5.	178

References

- Abramovich, F. and Angelini, C. (2006). Testing in mixed-effects fanova models. *Journal of statistical planning and inference*, 136(12):4326–4348.
- Aitken, A. C. (1936). On least squares and linear combination of observations. *Proceedings of the Royal Society of Edinburgh*, 55:42–48.
- Analytics, R. and Weston, S. (2015a). *doMC: Foreach Parallel Adaptor for 'parallel'*. R package version 1.3.4. Available at <https://CRAN.R-project.org/package=doMC>.
- Analytics, R. and Weston, S. (2015b). *foreach: Provides Foreach Looping Construct for R*. R package version 1.4.3. Available at <https://CRAN.R-project.org/package=foreach>.
- Analytics, R. and Weston, S. (2015c). *iterators: Provides Iterator Construct for R*. R package version 1.0.8. Available at <https://CRAN.R-project.org/package=iterators>.
- Antoniadis, A. and Sapatinas, T. (2007). Estimation and inference in functional mixed-effects models. *Computational statistics & data analysis*, 51(10):4793–4813.
- Aston, J. A. D., Chiou, J.-M., and Evans, J. P. (2010). Linguistic pitch analysis using functional principal component mixed effect models. *Journal of the Royal Statistical Society: Series C (Applied statistics)*, 59(2):297–317.
- Atkinson, K. and Han, W. (2009). *Theoretical Numerical Analysis: A Functional Analysis Framework*. Texts in Applied Mathematics. Springer, New York.
- Baayen, R. H., van Rij, J., de Cat, C., and Wood, S. N. (2017). Autocorrelated errors in experimental data in the language sciences: Some solutions offered by generalized additive mixed models. In Speelman, D., Heylen, K., and Geeraerts, D., editors, *Mixed Effects Regression Models in Linguistics*. Springer, Berlin Heidelberg. To appear.
- Baíllo, A., Cuevas, A., and Fraiman, R. (2011). Classification methods with functional data. In Ferraty, F. and Romain, Y., editors, *The Oxford Handbook of Functional Data Analysis*, pages 259–297. Oxford University Press, Oxford.

- Baladandayuthapani, V., Mallick, B. K., Young Hong, M., Lupton, J. R., Turner, N. D., and Carroll, R. J. (2008). Bayesian hierarchical spatially correlated functional data analysis with application to colon carcinogenesis. *Biometrics*, 64(1):64–73.
- Balakrishnan, A. (1960). Estimation and detection theory for multiple stochastic processes. *Journal of Mathematical Analysis and Applications*, 1(3–4):386–410.
- Bates, D. and Mächler, M. (2017). *Matrix: Sparse and Dense Matrix Classes and Methods*. R package version 1.2-8. Available at <https://CRAN.R-project.org/package=Matrix>.
- Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48.
- Bates, D., Mächler, M., Bolker, B., and Walker, S. (2016). *lme4: Linear mixed-effects models using Eigen and S4*. R package version 1.1-11. Available at <http://CRAN.R-project.org/package=lme4>.
- Bellman, R. E. (1961). *Adaptive Control Processes: A Guided Tour*. Princeton university press, Princeton, NJ.
- Bengtsson, H. (2016). *R.matlab: Read and Write MAT Files and Call MATLAB from Within R*. R package version 3.6.1. Available at <https://CRAN.R-project.org/package=R.matlab>.
- Besse, P. and Ramsay, J. O. (1986). Principal components analysis of sampled functions. *Psychometrika*, 51(2):285–311.
- Besse, P. C., Cardot, H., and Stephenson, D. B. (2000). Autoregressive forecasting of some functional climatic variations. *Scandinavian Journal of Statistics*, 27(4):673–687.
- Bigelow, J. L. and Dunson, D. B. (2007). Bayesian adaptive regression splines for hierarchical data. *Biometrics*, 63(3):724–732.
- Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, New York.
- Bongiorno, E. G., Salinelli, E., Goia, A., and Vieu, P. (2014). *Contributions in Infinite-dimensional Statistics and Related Topics*. Società Editrice Esculapio, Bologna.
- Bosq, D. (2000). *Linear Processes in Function Spaces: Theory and Applications*, volume 149 of *Lecture Notes in Statistics*. Springer, New York.
- Brockhaus, S. (2016). *Boosting Functional Regression Models*. Dissertation, Ludwig-Maximilians-Universität München.
- Brockhaus, S., Melcher, M., Leisch, F., and Greven, S. (2017). Boosting flexible functional regression models with a high number of functional historical effects. *Statistics and Computing*, 27(4):913–926.

- Brockhaus, S., Scheipl, F., Hothorn, T., and Greven, S. (2015). The functional linear array model. *Statistical Modelling*, 15(3):279–300.
- Brumback, B., Ruppert, D., and Wand, M. (1999). Comment on "Variable selection and function estimation in additive nonparametric regression using a data-based prior". *Journal of the American Statistical Association*, 94(447):794–797.
- Brumback, B. A. and Rice, J. A. (1998). Smoothing spline models for the analysis of nested and crossed samples of curves. *Journal of the American Statistical Association*, 93(443):961–976.
- Cederbaum, J. (2016). *sparseFLMM: Functional linear mixed models for irregularly or sparsely sampled data*. R package version 0.1.0. Available at <https://CRAN.R-project.org/package=sparseFLMM>.
- Cederbaum, J., Greven, S., Pouplier, M., and Hoole, P. (2014). Functional linear mixed model for irregularly spaced phonetics data. In Kneib, T., Sobotka, F., Fahrholz, J., and Irmer, H., editors, *Proceedings of the 29th International Workshop on Statistical Modelling*, pages 75–80.
- Cederbaum, J., Pouplier, M., Hoole, P., and Greven, S. (2016). Functional linear mixed models for irregularly or sparsely sampled data. *Statistical Modelling*, 16(1):67–88.
- Cederbaum, J., Scheipl, F., and Greven, S. (2018). Fast symmetric additive covariance smoothing. *Computational Statistics & Data Analysis*, 120:25–41. Advance online publication. DOI: <https://dx.doi.org/10.1016/j.csda.2017.11.002>.
- Chen, H. (2012). *Flexible Models and Methods for Longitudinal and Multilevel Functional Data*. PhD thesis, Graduate School of Arts and Sciences, Columbia University.
- Chen, H. and Wang, Y. (2011). A penalized spline approach to functional mixed effects model analysis. *Biometrics*, 67(3):861–870.
- Chen, K. and Müller, H.-G. (2012). Conditional quantile analysis when covariates are functions, with application to growth data. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 74(1):67–89.
- Chiou, J.-M., Chen, Y.-T., and Yang, Y.-F. (2014). Multivariate functional principal component analysis: A normalization approach. *Statistica Sinica*, 24(4):1571–1596.
- Crainiceanu, C. and Ruppert, D. (2004). Likelihood ratio tests in linear mixed models with one variance component. *Journal of the Royal Statistical Society, Series B (Statistical methodology)*, 66(1):165–185.
- Cuevas, A. (2014). A partial overview of the theory of statistics with functional data. *Journal of Statistical Planning and Inference*, 147:1–23.

- Cuevas, A., Febrero, M., and Fraiman, R. (2007). Robust estimation and classification for functional data via projection-based depth notions. *Computational Statistics*, 22(3):481–496.
- 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.
- Descary, M.-H. and Panaretos, V. M. (2016). Functional data analysis by matrix completion. arXiv preprint: arXiv:1609.00834v1. Accepted.
- Di, C.-Z., Crainiceanu, C. M., Caffo, B. S., and Punjabi, N. M. (2009). Multilevel functional principal component analysis. *The Annals of Applied Statistics*, 3(1):458–488.
- Di, C.-Z., Crainiceanu, C. M., and Jank, W. S. (2014). Multilevel sparse functional principal component analysis. *Stat*, 3(1):126–143.
- Diggle, P., Heagerty, P., Liang, K.-Y., and Zeger, S. (2002). *Analysis of Longitudinal Data*. Oxford University Press, Oxford.
- Dowle, M. and Srinivasan, A. (2015). *data.table: Extension of ‘data.frame’*. R package version 1.9.6. Available at <https://CRAN.R-project.org/package=data.table>.
- Eilers, P. H. and Marx, B. D. (2003). Multivariate calibration with temperature interaction using two-dimensional penalized signal regression. *Chemometrics and Intelligent Laboratory Systems*, 66(2):159–174.
- Eilers, P. H. C. and Marx, B. D. (1996). Flexible smoothing with B-splines and penalties. *Statistical Science*, 11(2):89–121.
- Engelhardt, A., Kanawade, R., Knipfer, C., Schmid, M., Stelzle, F., and Adler, W. (2014). Comparing classification methods for diffuse reflectance spectra to improve tissue specific laser surgery. *BMC Medical Research Methodology*, 14:1–15. Article number 91.
- Fahrmeir, L., Kneib, T., Lang, S., and Marx, B. (2013). *Regression: Models, Methods and Applications*. Springer, Berlin Heidelberg.
- Fan, J., Huang, T., and Li, R. (2007). Analysis of longitudinal data with semiparametric estimation of covariance function. *Journal of the American Statistical Association*, 102(478):632–641.
- Faraway, J. J. (1997). Regression analysis for a functional response. *Technometrics*, 39(3):254–261.
- 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., editor (2011). *Recent Advances in Functional Data Analysis and Related Topics*. Physica-Verlag, Heidelberg.

- Ferraty, F. and Romain, Y., editors (2011). *The Oxford Handbook of Functional Data Analysis*. Oxford University Press, Oxford.
- Ferraty, F. and Vieu, P. (2006). *Nonparametric Functional Data Analysis: Theory and Practice*. Springer Series in Statistics. Springer, New York.
- Fletcher, R. (2000). *Practical Methods of Optimization*. John Wiley & Sons, Chichester, UK, second edition.
- Fuchs, K., Scheipl, F., and Greven, S. (2015). Penalized scalar-on-functions regression with interaction term. *Computational Statistics & Data Analysis*, 81:38–51.
- Genz, A., Bretz, F., Miwa, T., Mi, X., Leisch, F., Scheipl, F., and Hothorn, T. (2014). *mvtnorm: Multivariate Normal and t Distributions*. R package version 0.9-99992. Available at <http://CRAN.R-project.org/package=mvtnorm>.
- Gertheiss, J., Goldsmith, J., and Staicu, A.-M. (2017). A note on modeling sparse exponential-family functional response curves. *Computational Statistics & Data Analysis*, 105:46–52.
- Gervini, D. (2008). Robust functional estimation using the median and spherical principal components. *Biometrika*, 95(3):587–600.
- Goia, A. and Vieu, P. (2016). An introduction to recent advances in high/infinite dimensional statistics. *Journal of Multivariate Analysis*, 146:1–6.
- Goldsmith, J., Greven, S., and Crainiceanu, C. M. (2013). Corrected confidence bands for functional data using principal components. *Biometrics*, 69(1):41–51.
- Goldsmith, J., Zipunnikov, V., and Schrack, J. (2015). Generalized multilevel function-on-scalar regression and principal component analysis. *Biometrics*, 71(2):344–353.
- Goulet, V., Dutang, C., Mächler, M., Firth, D., Shapira, M., and Stadelmann, M. (2017). *expm: Matrix Exponential, Log, 'etc'*. R package version 0.999-1. Available at <https://CRAN.R-project.org/package=expm>.
- Grenander, U. (1950). Stochastic processes and statistical inference. *Arkiv för Matematik*, 1(3):195–277.
- Greven, S. (2007). *Non-Standard Problems in Inference for Additive and Linear Mixed Models*. Dissertation, Ludwig-Maximilians-Universität München.
- Greven, S. and Cederbaum, J. (2017). *denseFLMM: Functional linear mixed models for densely sampled data*. R package version 0.1.0. Available at <https://CRAN.R-project.org/package=denseFLMM>.

- Greven, S., Cederbaum, J., and Shou, H. (2016). Principal component-based functional linear mixed models. Working paper.
- Greven, S., Crainiceanu, C., Küchenhoff, H., and Peters, A. (2008). Restricted likelihood ratio testing for zero variance components in linear mixed models. *Journal of Computational and Graphical Statistics*, 17(4):870–891.
- Greven, S., Crainiceanu, C. M., Caffo, B., and Reich, D. (2010). Longitudinal functional principal component analysis. *Electronic Journal of Statistics*, 4:1022–1054.
- Greven, S. and Kneib, T. (2010). On the behaviour of marginal and conditional AIC in linear mixed models. *Biometrika*, 97(4):773–789.
- Greven, S. and Scheipl, F. (2017). A general framework for functional regression modelling. *Statistical Modelling*, 17(1–2):1–35.
- Guo, W. (2002). Functional mixed effects models. *Biometrics*, 58(1):121–128.
- Guo, W. (2004). Functional data analysis in longitudinal settings using smoothing splines. *Statistical Methods in Medical Research*, 13(1):49–62.
- Hadjipantelis, P. Z., Aston, J. A., Müller, H.-G., and Evans, J. P. (2015). Unifying amplitude and phase analysis: A compositional data approach to functional multivariate mixed-effects modeling of Mandarin Chinese. *Journal of the American Statistical Association*, 110(510):545–559.
- Hall, P. (2011). Principal component analysis for functional data: Methodology, theory and discussion. In Ferraty, F. and Romain, Y., editors, *The Oxford Handbook of Functional Data Analysis*, pages 210–234. Oxford University Press, Oxford.
- Hall, P. and Hosseini-Nasab, M. (2006). On properties of functional principal components analysis. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 68(1):109–126.
- Hall, P., Müller, H., and Yao, F. (2008). Modelling sparse generalized longitudinal observations with latent Gaussian processes. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 70(4):703–723.
- Happ, C. and Greven, S. (2017). Multivariate functional principal component analysis for data observed on different (dimensional) domains. *Journal of the American Statistical Association*. Accepted, DOI: <https://dx.doi.org/10.1080/01621459.2016.1273115>.
- Harville, D. (1976). Extension of the Gauss-Markov theorem to include the estimation of random effects. *The Annals of Statistics*, 4(2):384–395.
- Harville, D. (1997). *Matrix Algebra from a Statistician's Perspective*. Springer, New York.

- Harville, D. A. (1974). Bayesian inference for variance components using only error contrasts. *Biometrika*, 61(2):383–385.
- Harville, D. A. (1977). Maximum likelihood approaches to variance component estimation and to related problems. *Journal of the American Statistical Association*, 72(358):320–338.
- Hastie, T. and Tibshirani, R. (1993). Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 55(4):757–796.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, New York, second edition.
- Hastie, T., Tibshirani, R., Leisch, F., Hornik, K., and Ripley, B. D. (2016). *mda: Mixture and Flexible Discriminant Analysis*. R package version 0.4-9. Available at <https://CRAN.R-project.org/package=mda>.
- Henderson, C. R. (1950). Estimation of genetic parameters (abstract). *Annals of Mathematical Statistics*, 21(2):309–310.
- Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.
- Horváth, L. and Kokoszka, P. (2012). *Inference for Functional Data with Applications*, volume 200 of *Springer Series in Statistics*. Springer, New York.
- Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441.
- Hotelling, H. (1935). The most predictable criterion. *Journal of Educational Psychology*, 26(2):139–142.
- Hotelling, H. (1957). The relations of the newer multivariate statistical methods to factor analysis. *British Journal of Statistical Psychology*, 10(2):69–79.
- Hsing, T. and Eubank, R. (2015). *Theoretical Foundations of Functional Data Analysis, with an Introduction to Linear Operators*. Wiley Series in Probability and Statistics. John Wiley & Sons, Sussex.
- Huang, J. Z., Shen, H., and Buja, A. (2008). Functional principal components analysis via penalized rank one approximation. *Electronic Journal of Statistics*, 2:678–695.
- Huang, L., Scheipl, F., Goldsmith, J., Gellar, J., Harezlak, J., McLean, M. W., Swihart, B., Xiao, L., Crainiceanu, C., and Reiss, P. (2016a). *refund: Regression with Functional Data*. R package version 0.1-14. Available at <https://CRAN.R-project.org/package=refund>.

- Huang, L., Scheipl, F., Goldsmith, J., Gellar, J., Harezlak, J., McLean, M. W., Swihart, B., Xiao, L., Crainiceanu, C., and Reiss, P. (2016b). *refundDevel: Regression with Functional Data (Developer Version)*.
- Isserlis, L. (1918). On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables. *Biometrika*, 12(1–2):134–139.
- Jacques, J. and Preda, C. (2014a). Functional data clustering: A survey. *Advances in Data Analysis and Classification*, 8(3):231–255.
- Jacques, J. and Preda, C. (2014b). Model-based clustering for multivariate functional data. *Computational Statistics & Data Analysis*, 71:92–106.
- James, G. M. (2002). Generalized linear models with functional predictors. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 64(3):411–432.
- James, G. M. and Hastie, T. J. (2001). Functional linear discriminant analysis for irregularly sampled curves. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 63(3):533–550.
- James, G. M., Hastie, T. J., and Sugar, C. A. (2000). Principal component models for sparse functional data. *Biometrika*, 87(3):587–602.
- Jolliffe, I. (2002). *Principal Component Analysis*. Springer, New York, second edition.
- Karhunen, K. (1947). Über lineare Methoden in der Wahrscheinlichkeitsrechnung. *Annales Academiae Scientiarum Fennicae. Series A.1, Mathematica-Physica*, 37:1–79.
- Kaslow, R. A., Ostrow, D. G., Detels, R., Phair, J. P., Polk, B. F., and Rinaldo, C. R. (1987). The multicenter AIDS cohort study: Rationale, organization, and selected characteristics of the participants. *American journal of epidemiology*, 126(2):310–318.
- Kauermann, G. and Wegener, M. (2011). Functional variance estimation using penalized splines with principal component analysis. *Statistics and Computing*, 21(2):159–171.
- Kelly, E. J. and Root, W. L. (1960). A representation of vector-valued random processes. *Journal of Mathematics and Physics*, 39(1–4):211–216.
- Kendall, M. G. (1957). *A Course in Multivariate Analysis*. Number 2 in Griffin’s Statistical Monographs and Courses. Charles Griffin & Co., London.
- Kokoszka, P. (2012). Dependent functional data. *ISRN Probability and Statistics*, 2012:1–30. Article ID 958254.
- Krafty, R. T., Hall, M., and Guo, W. (2011). Functional mixed effects spectral analysis. *Biometrika*, 98(3):583–598.

- Krivobokova, T., Crainiceanu, C. M., and Kauermann, G. (2012). Fast adaptive penalized splines. *Journal of Computational and Graphical Statistics*, 17(1):1–20.
- Krivobokova, T. and Kauermann, G. (2007). A note on penalized spline smoothing with correlated errors. *Journal of the American Statistical Association*, 102(480):1328–1337.
- Laird, N. M. and Ware, J. H. (1982). Random-effects models for longitudinal data. *Biometrics*, 38(4):963–974.
- Leurgans, S. E., Moyeed, R. A., and Silverman, B. W. (1993). Canonical correlation analysis when the data are curves. *Journal of the Royal Statistical Society. Series B (Statistical methodology)*, 55(3):725–740.
- Li, Y., Wang, N., Hong, M., Turner, N. D., Lupton, J. R., and Carroll, R. J. (2007). Nonparametric estimation of correlation functions in longitudinal and spatial data, with application to colon carcinogenesis experiments. *The Annals of Statistics*, 35(4):1608–1643.
- Little, R. J. and Rubin, D. B. (2014). *Statistical Analysis with Missing Data*. Wiley Series in Probability and Statistics. John Wiley & Sons, second edition.
- Liu, C., Ray, S., and Hooker, G. (2016). Functional principal component analysis of spatially correlated data. *Statistics and Computing*, 27(6):1639–1654.
- Liu, Z. and Guo, W. (2012). Functional mixed effects models. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(6):527–534.
- Locantore, N., Marron, J., Simpson, D., Tripoli, N., Zhang, J., Cohen, K., Boente, G., Fraiman, R., Brumback, B., Croux, C., et al. (1999). Robust principal component analysis for functional data. *Test*, 8(1):1–73.
- Loève, M. (1946). Fonctions aléatoires du second ordre. *La Revue Scientifique*, 84:195–206.
- López-Pintado, S. and Romo, J. (2009). On the concept of depth for functional data. *Journal of the American Statistical Association*, 104(486):718–734.
- Lunn, D., Thomas, A., Best, N., and Spiegelhalter, D. (2000). Winbugs – a Bayesian modelling framework: Concepts, structure, and extensibility. *Statistics and Computing*, 10(4):325–337.
- Marra, G. and Wood, S. N. (2012). Coverage properties of confidence intervals for generalized additive model components. *Scandinavian Journal of Statistics*, 39(1):53–74.
- MATLAB (2013). *Version 8.2.0.701 (R2013b)*. The MathWorks Inc., Natick, MA.
- McCulloch, C. E., Searle, S. R., and Neuhaus, J. M. (2008). *Generalized, Linear, and Mixed Models*. Wiley Series in Probability and Statistics. John Wiley & Sons, Hoboken, NJ, second edition.

- McLean, M. W., Hooker, G., Staicu, A.-M., Scheipl, F., and Ruppert, D. (2014). Functional generalized additive models. *Journal of Computational and Graphical Statistics*, 23(1):249–269.
- McMurry, T. and Politis, D. (2011). Resampling methods for functional data. In Ferraty, F. and Romain, Y., editors, *The Oxford Handbook of Functional Data Analysis*, pages 189–209. Oxford University Press, Oxford.
- Mercer, J. (1909). Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society of London. Series A (Statistics in society)*, 209(441–458):415–446.
- Molenberghs, G. and Verbeke, G. (2005). *Models for Discrete Longitudinal Data*. Springer Series in Statistics. Springer, New York.
- Morris, J. S. (2015). Functional regression. *Annual Review of Statistics and Its Application*, 2(1):321–359.
- Morris, J. S., Arroyo, C., Coull, B. A., Ryan, L. M., Herrick, R., and Gortmaker, S. L. (2006). Using wavelet-based functional mixed models to characterize population heterogeneity in accelerometer profiles: A case study. *Journal of the American Statistical Association*, 101(476):1352–1364.
- Morris, J. S., Baladandauthapani, V., Herrick, R., Sanna, P., and Gutstein, H. (2011). Automated analysis of quantitative image data using isomorphic functional mixed models, with application to proteomic data. *Annals of Applied Statistics*, 5(2A):894–923.
- 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.
- Morris, J. S., Vannucci, M., Brown, P. J., and Carroll, R. J. (2003). Wavelet-based nonparametric modeling of hierarchical functions in colon carcinogenesis. *Journal of the American Statistical Association*, 98(463):573–583.
- Müller, H.-G. and Stadtmüller, U. (2005). Generalized functional linear models. *Annals of Statistics*, 33(2):774–805.
- Nadarajah, S. and Pogány, T. K. (2016). On the distribution of the product of correlated normal random variables. *Comptes Rendus Mathématique*, 354(2):201–204.
- Naylor, A. W. and Sell, G. R. (2000). *Linear Operator Theory in Engineering and Science*. Applied Mathematical Sciences. Springer, New York.
- Novomestky, F. (2013). *orthopolynom: Collection of functions for orthogonal and orthonormal polynomials*. R package version 1.0-5. Available at <https://CRAN.R-project.org/package=orthopolynom>.

- Panaretos, V. M. and Tavakoli, S. (2013). Cramér–Karhunen–Loève representation and harmonic principal component analysis of functional time series. *Stochastic Processes and their Applications*, 123(7):2779–2807.
- Patterson, H. D. and Thompson, R. (1971). Recovery of inter-block information when block sizes are unequal. *Biometrika*, 58(3):545–554.
- Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.
- Peng, J. and Paul, D. (2009). A geometric approach to maximum likelihood estimation of the functional principal components from sparse longitudinal data. *Journal of Computational and Graphical Statistics*, 18(4):995–1015.
- Pinheiro, J., Bates, D., DebRoy, S., Sarkar, D., and R Core Team (2016). *nlme: Linear and Nonlinear Mixed Effects Models*. R package version 3.1-126. Available at <http://CRAN.R-project.org/package=nlme>.
- Pinheiro, J. C. and Bates, D. M. (2000). *Mixed-Effects Models in S and S-PLUS*. Statistics and Computing. Springer, New York.
- Poupplier, M., Cederbaum, J., Hoole, P., Marin, S., and Greven, S. (2017). Mixed modeling for irregularly sampled and correlated functional data: Speech science applications. *Journal of the Acoustical Society of America*, 142(2):935–946.
- Poupplier, M. and Hoole, P. (2016). Articulatory and acoustic characteristics of German fricative clusters. *Phonetica*, 73(1):52–78.
- Poupplier, M., Hoole, P., Cederbaum, J., Greven, S., and Pastätter, M. (2014). Perceptual and articulatory factors in German fricative assimilation. In Fuchs, S., Grice, M., Hermes, A., Lancia, L., and Mücke, D., editors, *Proceedings of the 10th International Seminar on Speech Production (ISSP)*, pages 332–335.
- Poupplier, M., Hoole, P., and Scobbie, J. M. (2011). Investigating the asymmetry of English sibilant assimilation: Acoustic and EPG data. *Laboratory Phonology*, 2(1):1–33.
- Qin, L. and Guo, W. (2006). Functional mixed-effects model for periodic data. *Biostatistics*, 7(2):225–234.
- R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Ramsay, J. O. (1982). When the data are functions. *Psychometrika*, 47(4):379–396.
- Ramsay, J. O. (2011). Curve registration. In Ferraty, F. and Romain, Y., editors, *The Oxford Handbook of Functional Data Analysis*, pages 235–258. Oxford University Press, Oxford.

- Ramsay, J. O. and Dalzell, C. (1991). Some tools for functional data analysis. *Journal of the Royal Statistical Society. Series B (Statistical methodology)*, 53(3):539–572.
- Ramsay, J. O. and Silverman, B. W. (2005). *Functional Data Analysis*. Springer Series in Statistics. Springer, New York, second edition.
- Rao, C. R. (1958). Some statistical methods for comparison of growth curves. *Biometrics*, 14(1):1–17.
- Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.
- Reiss, P. T., Huang, L., and Mennes, M. (2010). Fast function-on-scalar regression with penalized basis expansions. *The International Journal of Biostatistics*, 6(1):1–30.
- 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.
- Reiss, P. T. and Ogden, R. T. (2009). Smoothing parameter selection for a class of semiparametric linear models. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 71(2):505–523.
- Rice, J. A. (2004). Functional and longitudinal data analysis: Perspectives on smoothing. *Statistica Sinica*, 14(3):631–648.
- Ruppert, D., Wand, M. P., and Carroll, R. J. (2003). *Semiparametric Regression*. Cambridge University Press, Cambridge.
- Sarkar, D. (2008). *Lattice: Multivariate Data Visualization with R*. Springer, New York.
- Scarpa, B. and Dunson, D. B. (2009). Bayesian hierarchical functional data analysis via contaminated informative priors. *Biometrics*, 65(3):772–780.
- Scheipl, F., Gertheiss, J., and Greven, S. (2016a). Generalized functional additive mixed models. *Electronic Journal of Statistics*, 10(1):1455–1492.
- Scheipl, F., Greven, S., et al. (2016b). Identifiability in penalized function-on-function regression models. *Electronic Journal of Statistics*, 10(1):495–526.
- Scheipl, F., Staicu, A.-M., and Greven, S. (2015). Functional additive mixed models. *Journal of Computational and Graphical Statistics*, 24(2):477–501.
- Shang, H. L. (2014). A survey of functional principal component analysis. *ASTA Advances in Statistical Analysis*, 98(2):121–142.
- Shou, H., Zipunnikov, V., Crainiceanu, C. M., and Greven, S. (2015). Structured functional principal component analysis. *Biometrics*, 71(1):247–257.

- Silverman, B. W. (1996). Smoothed functional principal components analysis by choice of norm. *The Annals of Statistics*, 24(1):1–24.
- Staicu, A.-M., Crainiceanu, C. M., and Carroll, R. J. (2010). Fast methods for spatially correlated multilevel functional data. *Biostatistics*, 11(2):177–194.
- Staicu, A.-M., Crainiceanu, C. M., Reich, D. S., and Ruppert, D. (2012). Modeling functional data with spatially heterogeneous shape characteristics. *Biometrics*, 68(2):331–343.
- Staniswalis, J. and Lee, J. (1998). Nonparametric regression analysis of longitudinal data. *Journal of the American Statistical Association*, 93(444):1403–1404.
- Stelzle, F., Zam, A., Adler, W., Tangermann-Gerk, K., Douplik, A., Nkenke, E., and Schmidt, M. (2011). Optical nerve detection by diffuse reflectance spectroscopy for feedback controlled oral and maxillofacial laser surgery. *Journal of Translational Medicine*, 9:1–9. Article number 20.
- Stram, D. O. and Lee, J. W. (1994). Variance components testing in the longitudinal mixed effects model. *Biometrics*, 50(3):1171–1177.
- Tai, W. (2009). *Regularized Estimation of Covariance Matrices for Longitudinal Data Through Smoothing and Shrinkage*. PhD thesis, Columbia University.
- Tran, N. M. (2008). *An Introduction to Theoretical Properties of Functional Principal Component Analysis*. PhD thesis, Department of Mathematics and Statistics, The University of Melbourne.
- Ullah, S. and Finch, C. F. (2013). Applications of functional data analysis: A systematic review. *BMC Medical Research Methodology*, 13(43):1–12.
- Vaida, F. and Blanchard, S. (2005). Conditional Akaike Information for mixed-effects models. *Biometrika*, 92(2):351–370.
- Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition.
- Wang, H. and Kai, B. (2015). Functional sparsity: Global versus local. *Statistica Sinica*, 25(4):1337–1354.
- Wang, J.-L., Chiou, J.-M., and Müller, H.-G. (2016). Functional data analysis. *Annual Review of Statistics and Its Application*, 3(1):257–295.
- Wickham, H. (2009). *ggplot2: Elegant Graphics for Data Analysis*. Springer, New York.
- Wiesenfarth, M., Krivobokova, T., Klasen, S., and Sperlich, S. (2012). Direct simultaneous inference in additive models and its application to model undernutrition. *Journal of the American Statistical Association*, 107(500):1286–1296.

- Wood, S. N. (2006). *Generalized Additive Models: An Introduction with R*. Chapman & Hall/CRC, Boca Raton, FL.
- Wood, S. N. (2011). Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 73(1):3–36.
- Wood, S. N., Goude, Y., and Shaw, S. (2015). Generalized additive models for large data sets. *Journal of the Royal Statistical Society: Series C (Applied statistics)*, 64(1):139–155.
- Wood, S. N., Li, Z., Shaddick, G., and Augustin, N. H. (2017). Generalized additive models for gigadata: Modelling the UK black smoke network daily data. *Journal of the American Statistical Association*, 112(519):1199–1210.
- Wood, S. N., Pya, N., and Säfken, B. (2016). Smoothing parameter and model selection for general smooth models. *Journal of the American Statistical Association*, 111(516):1548–1575.
- Wu, H. and Zhang, J.-T. (2002). Local polynomial mixed-effects models for longitudinal data. *Journal of the American Statistical Association*, 97(459):883–897.
- Wu, W. B. and Pourahmadi, M. (2003). Nonparametric estimation of large covariance matrices of longitudinal data. *Biometrika*, 90(4):831–844.
- Xiao, L., Li, C., Checkley, W., and Crainiceanu, C. (2016a). *face: Fast Covariance Estimation for Sparse Functional Data*. R package version 0.1-1. Available at <https://CRAN.R-project.org/package=face>.
- Xiao, L., Li, C., Checkley, W., and Crainiceanu, C. M. (2017). Fast covariance estimation for sparse functional data. *Statistics and Computing*. Advance online publication. DOI: <https://dx.doi.org/10.1007/s11222-017-9744-8>.
- Xiao, L., Zupnik, V., Ruppert, D., and Crainiceanu, C. M. (2016b). Fast covariance estimation for high-dimensional functional data. *Statistics and Computing*, 26(1):409–421.
- Yao, F., Müller, H.-G., Clifford, A. J., Dueker, S. R., Follett, J., Lin, Y., Buchholz, B. A., and Vogel, J. S. (2003). Shrinkage estimation for functional principal component scores with application to the population kinetics of plasma folate. *Biometrics*, 59(3):676–685.
- 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.
- Zhang, J.-T. (2013). *Analysis of Variance for Functional Data*. Chapman & Hal/CRC, Boca Raton, FL.
- Zhang, X. and Wang, J.-L. (2016). From sparse to dense functional data and beyond. *The Annals of Statistics*, 44(5):2281–2321.

- Zhou, L., Huang, J. Z., Martinez, J. G., Maity, A., Baladandayuthapani, V., and Carroll, R. J. (2010). Reduced rank mixed effects models for spatially correlated hierarchical functional data. *Journal of the American Statistical Association*, 105(489):390–400.
- Zhu, H., Brown, P. J., and Morris, J. S. (2011). Robust, adaptive functional regression in functional mixed model framework. *Journal of the American Statistical Association*, 106(495):1167–1179.
- Zhu, H., Brown, P. J., and Morris, J. S. (2012). Robust classification of functional and quantitative image data using functional mixed models. *Biometrics*, 68(4):1260–1268.
- Zipunnikov, V., Caffo, B., Yousem, D. M., Davatzikos, C., Schwartz, B. S., and Crainiceanu, C. M. (2011). Multilevel functional principal component analysis for high-dimensional data. *Journal of Computational and Graphical Statistics*, 20(4):852–873.
- Zipunnikov, V., Greven, S., Shou, H., Caffo, B. S., Reich, D. S., Crainiceanu, C. M., et al. (2014). Longitudinal high-dimensional principal components analysis with application to diffusion tensor imaging of multiple sclerosis. *The Annals of Applied Statistics*, 8(4):2175–2202.
- Zyskind, G. and Martin, F. B. (1969). On best linear estimation and general Gauss-Markov theorem in linear models with arbitrary nonnegative covariance structure. *SIAM Journal on Applied Mathematics*, 17(6):1190–1202.

Eidesstattliche Versicherung

(Siehe Promotionsordnung vom 12. Juli 2011, §8 Abs. 2 Pkt. 5)

Hiermit erkläre ich an Eides statt, dass die Dissertation von mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

Berlin, den 06. Juni 2017

Jona Cederbaum

