ON GENERALIZED ADDITIVE MODELS FOR REGRESSION WITH FUNCTIONAL DATA

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Mathew W. McLean

August 2013

ON GENERALIZED ADDITIVE MODELS FOR REGRESSION WITH FUNCTIONAL DATA

Mathew W. McLean, Ph.D.

Cornell University 2013

The focus of this dissertation is the introduction of the functional generalized additive model (FGAM), a novel regression model for association studies between a scalar response and a functional predictor. The FGAM extends the commonly used functional linear model (FLM), offering greater flexibility while still being simple to interpret and easy to estimate. The link-transformed mean response is modelled as the integral with respect to t of $F\{X(t), t\}$ where $F(\cdot, \cdot)$ is an unknown, bivariate regression function and $X(t)$ is a functional covariate. Compare this with the FLM which has $F\{X(t), t\} = \beta(t)X(t)$, where $\beta(t)$ is an unknown coefficient function. Rather than having an additive model in some projection of the data, the model incorporates the functional predictor directly and thus can be viewed as the natural functional extension of generalized additive models.

The first part of the dissertation shows how to estimate $F(\cdot, \cdot)$ using tensor-product B-splines with roughness penalties. Fast, stable methods are used to fit the FGAM and I discuss how approximate confidence bands can be constructed for the true regression surface. Additional functional predictors can be included with little added difficulty. The performance of the estimation procedure and the confidence bands is evaluated using simulated data and I compare FGAM's predictive performance with other competing scalar-on-function regression alternatives, including the popular functional linear model. I illustrate the usefulness of the approach through an application to brain tractography, where $X(t)$ is a signal

from diffusion tensor imaging at position t, along a tract in the brain. In one example, the response is disease-status (case or control) and in a second example, it is the score on a cognitive test. `R` code for performing estimation, plotting, and prediction for the FGAM is explained and is available in the package `refund` on CRAN.

Frequently in practise, only incomplete, noisy versions of the functions one wishes to analyze are observed. The estimation procedure used in the first part of the thesis requires that the functional predictors be noiselessly observed on a regular grid. In the second part of the dissertation, I restrict attention to the identity link-Gaussian error case and develop a Bayesian version of FGAM. This approach allows for the functional covariates to be sparsely observed and measured with error. I consider both Monte Carlo and variational Bayes methods for jointly fitting the FGAM with sparsely observed covariates and recovering the true functional predictors. Due to the complicated form of the model posterior distribution and full conditional distributions, standard Monte Carlo and variational Bayes algorithms cannot be used. As such, the work should be of independent interest to applied Bayesian statisticians. The numerical studies demonstrate the benefits of the proposed algorithms over a two-step approach of first recovering the complete trajectories using standard techniques and then fitting a functional regression model. In a real data analysis, the methods are applied to forecasting closing price for items being auctioned on the online auction website eBay.

Finally, in the third part of the thesis I propose and compare several different procedures for testing when a scalar on function regression relationship is truly nonlinear. By using an alternative parametrization for the FGAM as a mixed model, it is shown how the functional linear model can be represented as a simple mixed model nested within the FGAM. Using this representation, I then consider

two types of tests, those based on restricted likelihood ratio tests for zero variance components in mixed models and those involving Bayes factors where we use generalizations of g-priors as priors for the random effects coefficients. The methods are general and can also be applied to testing for interactions in a multivariate additive model or for testing for no effect in the functional linear model. The performance of the proposed tests is assessed on simulated data and in an application to measuring diesel truck emissions, where strong evidence of nonlinearities in the relationship between the functional predictor and the response are found.

BIOGRAPHICAL SKETCH

In west Philadelphia born and raised, on the playground was where I spent most of my days. Chilling out, maxing, relaxing all cool and all, shooting some b-ball outside of the school. When a couple of guys, who were up to no good, started making trouble in my neighbourhood. I got in one little fight and my mom got scared and said "You're moving with your auntie and uncle in Bel-Air".

Mathew William McLean has only spent one day in Philadelphia and never been to Bel-Air, but he did watch every episode of The Fresh Prince of Bel-Air while growing up in Winnipeg, Manitoba, Canada. He loved the long, freezing winters and short, mosquito-filled summers so much that he did not dare leave until his early twenties. It was at this time that the School of Operations Research and Information Engineering at Cornell University came calling, plucking him from his sheltered life of perpetual education in Winnipeg and transporting him to Ithaca, New York to continue the pursuit of his dream of never leaving school.

After five fun-filled years in Ithaca, Mr. McLean will be departing with a PhD and the promise of more schooling as a post-doctoral researcher at Texas A&M University in College Station, Texas.

To The Chapter House,

thanks for all the pints.

To my friends,

thanks for always being there to enjoy one with me.

ACKNOWLEDGEMENTS

I first wish to thank all the wonderful friends I have met during my time in graduate school who have made living in Ithaca such a pleasure these past five years. I was extremely lucky to have such an incredible group of people start the PhD program at the same time as me: Rolf, Martin, Brad, Shanshan, Zach, Dima, Tia, Gabriel, Yi, and Eric somehow managed to make being away from home and swamped with homework and exams amazingly fun our first year here. Some of the other unforgettable people ahead or behind me in the PhD program were Jim, Jake, Sunny, Collin, Baldur, Dennis, and Joyjit.

I must single out The Chapter House for being the site for so many of my most fun and memorable times at Cornell, and also John, Corey, Matt, and Mel for friendly service on every one of my many, many visits. Thanks to all my friends who made it such a fun place, including those mentioned above and also Raj, Dave Zeber, Inder, Matti, Jon, Dave Huland, Fran, Matthias, Michael, and Herb.

A massive thanks must go to my advisors David Ruppert and Giles Hooker. They were both such a pleasure to learn from and work with. I want to thank David Matteson and Sid Resnick for many fun, honest (some would say cynical), and interesting discussions and for being on my Special Committee. I learned a great deal from David and Shane Henderson on an early project I worked on at Cornell before starting my thesis research. I must single out Martin and Joyjit (again) for countless extremely helpful discussions regarding my research and half-baked ideas. I thank the Department of Statistical Science at Cornell, especially Martin Wells, for always treating me as if I was a member of their department.

Thanks to Fabian Scheipl for contributing ideas and helping with coding for the third chapter of the dissertation. Sonja Greven and Ana-Maria Staicu provided useful feedback and helped improve the quality of parts of this work. Thanks to

Ciprian Crainiceanu and Daniel Reich for providing the Diffusion Tensor Imaging data, Wolfgang Jank for providing the Ebay auction data, and Oliver Gao from Civil and Environmental Engineering for the truck emissions data. I thank the Natural Sciences and Engineering Research Council of Canada for all their support. The amount of funding they have contributed to my education is staggering to think about, and I am very grateful for it.

Thanks go to all the staff members of ORIE for their frequent help and for being such a pleasure to deal with on a daily basis. To name just a few, Dennis, Kathy, Celene, Eric, Tara, Monica, Lisa, Mark, and Jake.

I must single out Ann for the unforgettable time we spent together at Cornell. A huge thanks to my Uncle Dave for being like a second father to me after my dad passed away. I thank my brothers Rob and Ian and Lorraine, Emilie, and Rylie for, among many other things, frequent phone calls, texts, and Skype chats that ensured I never missed home too much while away in Ithaca. Lastly, I thank my parents, Bob and Shirley, for their unconditional love and support, and for working so astoundingly hard every single day of their lives; something which I will forever strive and fail to live up to.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Firstly, wow! You are actually reading my thesis. Secondly, forgive me for my Canadian English; a bastardized version of proper, British English. It is much less bastardized than American English, but it's still bastardized. Let's get started.

1.1 A Brief Overview of Functional Data Analysis

The need for functional data analysis (FDA) tools has arisen as data sets have continued to balloon in size with advances in technology. In several fields, sampling can be done on such a fine grid that it makes sense to view each sample as being observed on a continuum and coming from a smooth function. The continuum is often, but not always, time; and the functions are often, but need not be, univariate. FDA methods have been successfully applied in a wide array of fields such as chemometrics, econometrics, and biomechanics. In this dissertation, we will demonstrate applications to brain imaging, online auctions, and automobile exhaust emissions.

First introduced in the seminal paper by Ramsay and Dalzell[89], FDA is by now a fairly mature, but still rapidly developing field. There currently are many applied and theoretical monographs available; including Ferraty[27], Ferraty and Romain[28], Ferraty and Vieu[29], Horváth and Kokoszka[44], Ramsay and Silverman[90], Ramsay et al.[92], Shi and Choi[106], Zhang[136], and the standard introductory reference Ramsay and Silverman[91]. There have been several special journal editions on FDA and in `R` there are at least three software packages with a suite

of FDA methods available: `fda` (Ramsay et al.[93]), `refund` (Crainiceanu et al.[16]), and `fda.usc` (Febrero-Bande and Oviedo de la Fuente[25]).

Paramount to any FDA, is that the underlying functions are smooth, i.e. that one or more of the functions' derivatives exist. Smoothness of the functions is the key property that makes functional data methods advantageous over treating the data as discrete and using tools from multivariate statistics. Many of the methods of multivariate statistics have FDA counterparts. One of the most popular, which we will need in Chapter 3, is the extension of principal components analysis to functional data, called FPCA. Typically, FPCA is one of the first methods considered in an FDA in order to understand the underlying modes of variation present in the data. This is done by analyzing the eigenvalues and eigenfunctions of the functions' covariance surface.

Another useful preliminary tool for FDA is registration, which enables the sampled functions to be compared more easily. This is achieved by aligning the observed curves to remove the effect of any uninformative horizontal (phase) shifts from function to function or aligning based on some shared characteristic, such as minima or maxima or points where the functions cross zero. One of the main uses of FDA is for the study of derivatives, differential equations, and dynamical systems. The FDA tool that will be the focus of this dissertation is that of using the sampled functions in a regression model in order to understand the relationship between the functions and some other variable(s) of interest. Methods are available for when either one or both of the response and predictor in the model are functions. We will concentrate on the case of predicting a scalar response when the predictors are functions.

We will introduce this topic in more detail after a short detour to multivariate

data to discuss the key modelling tool used throughout the thesis, penalized splines.

1.2 Semiparametric Regression and Penalized Splines

Parametric models such as the multiple linear regression model, typically make very strong assumptions about the underlying data generation mechanism, assuming it depends only on a small number of parameters. Nonparametric models, on the other hand, make little to no assumptions about the underlying data generation and depend on an infinite number of parameters. As such, nonparametric models can be useful because they allow for capturing additional, more complicated structure that parametric models cannot. Practically, we must index a nonparametric model by some large, but finite set of parameters. Semiparametric models are an attempt to provide the best of both worlds, consisting of models that have both parametric and nonparametric components.

Additive models are one of the most popular nonparametric tools for describing how a response variable depends on one or more covariates. Standard, early references are Buja et al.[8] and Hastie and Tibshirani[42]. Additive models allow the relationship between the response and a covariate to be modelled by an unspecified smooth function, but traditionally make the strong assumption that the covariates do not interact to avoid unacceptably large variance in estimation. In general, additive models offer increased flexibility and potentially lower estimation bias than linear models while having less variance in estimation and being less susceptible to the curse of dimensionality than models that make no additivity assumptions. The goal of this dissertation is to develop a model that provides greater flexibility than the linear regression model for functional data (introduced shortly), while

still being simple to estimate and interpret.

The unspecified regression functions mentioned above are represented using a linear combination of basis functions. B-splines will be our basis functions of choice throughout the dissertation because of their popularity and computational efficiency. A key idea is that we can take tensor products of marginal bases to represent functions of higher order in a simple manner.

Central to any nonparametric method is a tuning parameter (usually called a smoothing parameter) and penalty which control the complexity and smoothness of the estimated regression functions. The tuning parameter must be estimated from the data and adequate choice of tuning parameter is essential for the success of the method. Not smoothing enough results in overfitting, and estimates that have low bias but high variance that will provide poor predictions for new data. Smoothing too much results in models that fail to explain key features of the data.

By penalized splines, we mean that the regression function is represented using low rank spline bases subject to a quadratic roughness penalty. Great introductions to penalized splines can be found in Ruppert et al.[99] and Wood[126]. In this work, we will frequently make use of the P-splines of Eilers and Marx[20], which we will describe in detail later. The roughness penalty is often, but not always, the squared second derivative of the function. For P-splines, the penalties are differences (of a prespecified order) of adjacent B-splines. Once the type of basis and penalty are specified, the user must also specify/estimate the number of basis functions used to represent the function, the location of the knots (usually taken to just be equally spaced along the domain of the function), the order of the spline, the order of the penalty, and finally the value of the smoothing parameter that multiplies the penalty. As mentioned, the smoothing parameter is the key component of the

model controlling function shape (assuming one uses an adequate number of spline functions).

One concept extremely important for this dissertation, is that penalized spline models may be represented as mixed models, which allows for parameters to be estimated using techniques for those models. We will make use of a different mixed model representation in each of the three main chapters of this thesis. In Chapter 2, we only mention in passing that an alternative estimation procedure using mixed models is available, but in Chapters 3 and 4, the representations are fundamental to the methods used and we discuss them in detail. In Chapter 3, the mixed model representation used gives rise to a proper prior and results in a proper full conditional for the regression coefficients in a Bayesian version of our model. In Chapter 4, the third mixed model representation we consider allows us to explicitly show how the canonical model for functional regression is nested within our model providing a means for hypothesis tests regarding which model better fits the data.

1.3 The Scalar on Function Regression Landscape

This dissertation studies regression with a functional predictor and a scalar response. Suppose one observes data $\{(X_i(t), Y_i) : \ t \in \mathcal{T}\}$ for $i = 1, \ldots, N$, where X_i is a real-valued, continuous, square-integrable, random curve on the compact interval $\mathcal{T}$ and Y_i is a scalar. It is usually assumed that the predictor, $X(\cdot)$, is observed at a dense grid of points. The problem addressed here is estimation of $E(Y_i|X_i)$, which is assumed independent of i. The most commonly used regression model in functional data analysis is the functional linear model (Ramsay and

Dalzell[89]), henceforth the FLM, given by

$$E(Y_i|X_i) = \theta_0 + \int_{\mathcal{T}} \beta(t)X_i(t)\,dt, \tag{1.1}$$

where $\beta(\cdot)$ is the functional coefficient with $\beta(t)$ describing the effect on the response of the functional predictor at time t. We can see that for any fixed t, the effect of $X(t)$ on Y is linear. This model has been the subject of far too many papers to list. Ramsay and Silverman[91] provides a nice introduction and uses penalized splines. Extensions to generalized responses are available (e.g., James[46], Müller and Stadtmüller[75]).

Given how often a linear model is not complex enough to model the true regression relationship for multivariate data, it would seem that there would also be functional data sets for which the FLM is not a flexible enough model, and there have been occasional attempts to propose nonlinear models for functional regression. One model that has seen a fair amount of attention is the fully nonparametric kernel estimator of Ferraty and Vieu[29]. This model is more of a black box, sometimes useful for predictions, but not for providing any insights into how exactly the functional predictor affects the response. Several authors have considered additive models that use linear functionals of the predictor curves as covariates, e.g. $E(Y_i|X_i) = \beta_0 + f\{\langle\beta(t)X_i(t)\rangle\} = \beta_0 + f\{\int \beta(t)X_i(t)dt\}$, for unknown β_0, $f(\cdot)$, and $\beta(t)$. Two such examples are Müller and Yao[76] and James and Silverman[47]. The former approach regresses on a finite number of functional principal components scores and the latter approach searches for linear functionals using projection pursuit. Both models rely strongly on the linear directions they estimate; for ease of interpretation, we desire a model that incorporates the functional predictors directly. A model that is additive in the principal component scores is not additive in $X(t)$ itself, and vice versa. We have the same complaints about Ait-Saïdi et al.[1], Chen et al.[12], Febrero-Bande et al.[26]. A less general model is the functional quadratic

regression model of Yao and Müller[132], which adds in the following term to the FLM: $\int_{\mathcal{T}} \int_{\mathcal{T}} \beta(s,t)X(s)X(t)dsdt$. Two noteworthy, though not direct competitors for the model we consider are Guillas and Lai[40], which examines the case when X is a bivariate function so that $E(Y_i|X_i) = \beta_0 + \int \int \beta(s,t)X(s,t)dsdt$; and Li et al.[60], which allows for interaction between a scalar and functional covariate though a single index.

1.4 Contributions of This Dissertation

The model that we introduce and that will be the focus of the thesis is

$$g\{E(Y_i|X_i)\} = \theta_0 + \int_{\mathcal{T}} F\{X_i(t), t\}\, dt, \tag{1.2}$$

where θ_0 is the intercept, g is a called a link function, and F is an unspecified smooth function to be estimated. We call model (1.2) the functional generalized additive model (henceforth abbreviate as FGAM). As a special case, when $g(x) = x$ and $F(x,t) = \beta(t)X_i(t)$, we obtain the FLM (1.1).

Our model allows for greater flexibility in representing the response-predictor relationship, as it does not make the strong assumption of linearity between the functional predictor and the functional parameter. To overcome the curse of dimensionality, we will perform smoothing in both the x and t components of $F(\cdot,\cdot)$. It will be shown that our model is the natural extension of generalized additive models (GAMs) to functional data.

The first core chapter of the dissertation shows how to estimate $F(\cdot,\cdot)$ using penalized splines. We review tensor-product P-splines and show how they can be used to estimate FGAM using very fast and stable methods, and also discuss

the implementation of FGAM in the popular statistics programming language R. Formulas for approximate confidence bands for the true regression surface are given and we discuss how additional functional predictors can be incorporated in the model. We compare FGAM's predictive performance with several other competing scalar-on-function regression models, including the FLM on simulated and real data sets and evaluate the coverage properties of the proposed confidence bands. We apply FGAM to a study in diffusion tensor imaging, where $X(t)$ is a signal from the one-dimensional image at position t, along a tract in the brain.

In order to extend FGAM to the common case where the functional predictors are sparsely observed and measured with error, we consider both Monte Carlo and variational Bayes (VB) methods for fitting the FGAM with sparsely observed covariates and recovering the true functional predictors simultaneously. Variational Bayes (VB) refers to a specific variational approximation used for Bayesian inference that relies on the assumption that a posterior density of interest factors into a product form over certain groups of model parameters. Though they are commonly used in computer science, the application of variational approximations in statistics is relatively new; Ormerod and Wand [80] provides an overview. When the amount of posterior dependence is small, it has been demonstrated in a number of applications that there can be little loss of accuracy and very large improvements in computation time over MCMC methods. Applications of VB to regression problems with missing data can be found in Faes et al. [22] and Goldsmith et al. [34], the latter of which considered the FLM.

Due to nonconjugacies in our model specification, we cannot use a vanilla Gibbs sampler or easily derive a simple VB algorithm. As such, the algorithms we develop are new and should be of independent interest. Our numerical experiments

show the difficulties that can occur if one applies standard tools for recovering sparse curves and then attempts to run a functional regression as if the curves were fully observed, and demonstrates the superiority of our algorithms. In a real data analysis, the methods are applied to forecasting closing price for items being auctioned on the online auction website eBay.

Finally, in the third part of the thesis we explore hypothesis tests for formally testing an FGAM fit for linearity. Through an alternative parametrization, we nest the FLM in the FGAM in a simple way that allows us to recast our testing problem as one of testing for zero variance components in a mixed model. We consider both restricted likelihood ratio tests and tests involving Bayes factors and g-priors. The use of these types of tests for checking for interactions in nonparametric models with bivariate functions has also not been considered before. The methods can also be used for testing for no effect in the functional linear model. The performance of the proposed tests is assessed on simulated data and in an application to measuring diesel truck emissions, where strong evidence of nonlinear effects in the data are found.

CHAPTER 2

DENSE COVARIATES

2.1 Preliminaries

2.1.1 Some Intuition for FGAM

To build intuition for the model, we start off immediately with an example. The application we consider in this chapter is diffusion tensor imaging (DTI), which we analyze in detail in Section 2.4. The dataset contains closely spaced evaluations of measures of neural functioning on multiple tracts in the brain for patients with multiple sclerosis and healthy controls. We will use these measurements as regressors and predict multiple health outcomes to gain a better understanding of how the disease is related to DTI signals. Our model is able to quantify the effect that the functional predictor has on the response at each position along the tract, something that a model such as the functional additive model of Müller and Yao[76] is unable to do, since it uses principal component scores and hence loses information about tract location. Another potential application of FGAM is to study how a risk factor trajectory such as body mass index or systolic blood pressure is related to a health outcome such as developing hypertension (e.g., see the study in Li et al.[59]). Our FGAM can locate times of life when the risk factor has its greatest effect; this is not possible if principal component scores are used in a GAM.

To see how our model can aid in uncovering the underlying structure of a functional regression problem consider Figure 2.1, which shows an estimated surface,

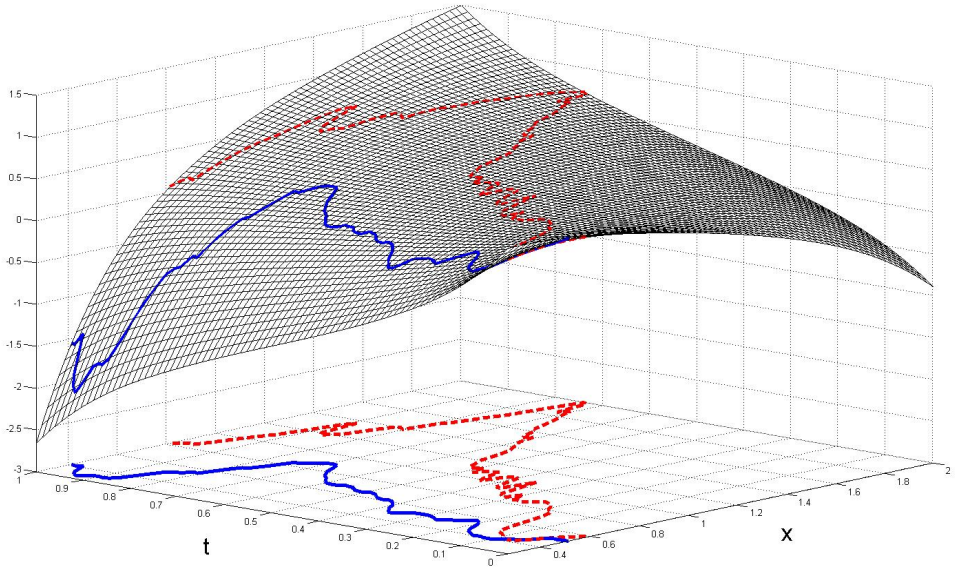

Figure 2.1: Estimated surface $\hat{F}(x,t)$ and two predictor curves for the DTI dataset. The solid curve belongs to a control and the dashed curve belongs to an MS patient.

$\widehat{F}(\cdot,\cdot)$, for one of the functional predictors in the DTI dataset when the response is disease status (= 1 if the subject has the disease). Overlaid on the surface are the observed functional predictor values for two subjects. The estimated surface is non-linear in x, so an FLM based on the predictors may be inadequate for this problem. We see that for the most part, the solid curve, belonging to a control subject, takes smaller values on the surface than the dashed curve, which belongs to a MS patient, does; thus, the subject with MS will have a higher fitted value and is more likely to be classified as having the disease. It will be shown for this dataset that the added generality of our approach leads to improved predictive accuracy over the FLM.

Additional insight can be gained by considering multivariate regression using the raw, discrete data. The FLM can be thought of as multiple linear regression with an infinite number of predictors, as we now explain. Let $t_{ij} = t_j$

for $1, \ldots, J$ denote the observation times for the curves $X_i(\cdot)$; then the usual multiple linear regression model $E(Y_i|X_i(t_1), \ldots, X_i(t_J)) = \beta_0 + \sum_{i=1}^{J} \beta_j X_i(t_j) = \beta_0 + J^{-1} \sum_{i=1}^{J} \beta_j' X_i(t_j)$ can be viewed as a Riemann sum approximation that converges to (1.1) as $J \to \infty$.

Now consider an additive model of the form $E\{Y_i|X_i(t_1), \ldots, X_i(t_J)\} = \theta_0 + \sum_{j=1}^{J} f_j\{X_i(t_j)\}$, where the f_j's are unspecified smooth functions. The basic idea is to rewrite the model as $E\{Y_i|X_i(t_1), \ldots, X_i(t_J)\} = \theta_0 + J^{-1} \sum_{j=1}^{J} F\{X_i(t_j), t_j\}$, and then let $J \to \infty$ and add a link function. The model obtained is our model (1.2). Hence, we believe are model to be the natural extension of additive models to functional data.

2.1.2 Parametrization of the Regression Surface

In this section, we introduce our representation for $F(\cdot, \cdot)$. It is assumed that $\mathcal{T} = [0, 1]$ and that $X(\cdot)$ takes values in a bounded interval which, without loss of generality, can be taken as $[0, 1]$. The latter assumption is guaranteed by the proposed transformation of the functional predictors discussed in Section 2.1.4.

We will model $F(\cdot, \cdot)$ using tensor products of B-splines. Splines are commonly used for estimation of functional linear models. For example, smoothing splines are used by Crambes et al.[18] and Yuan and Cai[133] and penalized splines are considered by Cardot et al.[10] and Goldsmith et al.[32]. These papers impose smoothness using a penalty on the integrated, squared second derivative of the coefficient function. Instead, we use the popular P-splines of Eilers and Marx[20], for reasons we explain shortly. P-splines use low rank B-splines bases with equally-spaced knots and a simple difference penalty on adjacent coefficients to control

smoothness.

Note that there will be some differences from standard fitting of tensor product P-splines. Namely, our design matrix is obtained from integrating products of B-splines evaluated at functional covariates. P-splines offer many computational advantages. Fast and flexible software is available for estimating our model in the `R` package `refund` (Crainiceanu et al.[16]) which makes use of smoothing parameter selection algorithms available in `mgcv` (Wood[127]). Additional scalar or functional predictors can be incorporated in a simple way and will not require backfitting. Both types of predictors can be included in either a linear or an additive fashion. Though we use P-splines, our estimation procedure can incorporate other bases and penalties for some or all of the covariates. It will be shown that the fitted values for the FGAM are linear in the tensor product B-spline coefficients so we actually have a penalized generalized linear model (GLM). We use

A bivariate spline model is used for $F(\cdot, \cdot)$ so that

$$F(x,t) = \sum_{j=1}^{K_x} \sum_{k=1}^{K_t} \theta_{j,k} B_j^X(x) B_k^T(t) \tag{2.1}$$

where $\{B_j^X(x) : j = 1, \ldots, K_x\}$ and $\{B_k^T(x) : k = 1, \ldots, K_t\}$ are spline bases on $[0, 1]$. We will use B-spline bases. It follows from combining (1.2) and (2.1), that we obtain the GLM

$$g\{E(Y_i|X_i)\} = \theta_0 + \int_{\mathcal{T}} F\{X_i(t), t\}dt = \theta_0 + \sum_{j=1}^{K_x} \sum_{k=1}^{K_t} \theta_{j,k} Z_{j,k}(i), \tag{2.2}$$

where $Z_{j,k}(i) = \int_{\mathcal{T}} B_j^X\{X_i(t)\}B_k^T(t)dt$. Each $Z_{j,k}(i)$ can be approximated by, say, Simpson's rule. Associated with each marginal basis are parameters, d_x and d_t, for the x and t bases, respectively, which specify the degree of differencing for the penalties for each axis. The penalties will be discussed in detail in Section 2.2.

2.1.3 Identifiability Constraints

Notice that for $F^*(x,t) = F(x,t) + g(t)$, where $\int_{\mathcal{T}} g(t)\,dt = 0$ we have $\int_{\mathcal{T}} F^*(x,t)\,dt = \int_{\mathcal{T}} F(x,t)\,dt$ and thus we must impose constraints to ensure identifiability. If no constraints were used, the function $g(t)$ would be chosen to maximize the penalized log-likelihood given in Section 3.2 and $g(t)$ would be regularized by the difference penalties we use. The penalties alone are not enough to ensure identifiability, however. One possibility is to simply use a ridge penalty as in Marx and Eilers[67]. For our difference penalties, functions of t in the null space of the penalty are polynomials of degree $d_t - 1$. Therefore $d_t - 1$ constraints are necessary for identifiability.

The constraint explicitly used by the fitting procedure is $\sum_{i=1}^{N} \int_{\mathcal{T}} F(X_i(t), t)\,dt = 0$. Any additional constraint necessary to ensure identifiability are determined by checking for numerical rank deficiency during fitting. The details are explained in the next section.

For fixed smoothing parameters, different identifiability constraints yield the same predictions and the same estimated $\hat{F}(\cdot,\cdot)$ up to a constant, though different estimates for the variance of the estimated surface (and therefore different confidence bands) will be obtained. The GCV score is also invariant to the constraints used. It is possible to switch to an alternative set of constraints after fitting our model using a pivoted QR decomposition along the lines of Wood et al.[129].

2.1.4 Transformation of the Predictors

Depending on the number of B-splines used for each axis, there could be a particular tensor product of B-splines that has no observed data on its support. This would lead to $Z_{j,k}(i) = 0$ for all i for some j, k pair, resulting in the design matrix containing a column of zeros. One remedy for this is to transform $X(t)$ by $G_t(x) := P\{X(t) < x\}$ for each value of t. Our model becomes

$$g\{E(Y_i|X_i)\} = \theta_0 + \int_{\mathcal{T}} F[G_t\{X_i(t)\}, t]dt = \theta_0 + \sum_{j=1}^{K_x}\sum_{k=1}^{K_t}\theta_{j,k}\int_{\mathcal{T}} B_j^G[G_t\{X_i(t)\}]B_k^T(t)dt, \tag{2.3}$$

where $B^G(\cdot)$ is a new B-spline basis with support on $[0, 1]$. Loosely, the data are being "stretched out" to fill the entire space that the grid of B-splines will cover. For any t on the grid where observations are taken, the transformed points will lie uniformly between $[0, 1]$. Though the estimation procedure is the same in both cases, clearly, $F(\cdot, \cdot)$ in (2.3) will have a different estimate from $F(\cdot, \cdot)$ in (1.2). We estimate $G_t(\cdot)$ using the empirical cdf $\widehat{G}_t(x) = n^{-1}\sum_{i=1}^{n} I\{X_i(t) < x\}$, where $I\{A\} = 1$ if condition A is true and $I\{A\} = 0$ otherwise. Once the $Z_{j,k}(i)$'s have been estimated, the fitting procedure is analogous to the case when the cdf transformation is not used. Another advantage of using this approach is that it does not require any assumptions about the range of the predictors. Besides the computational advantages, this transformation retains the benefit of ease of interpretation. In fact, $F(p, t)$ is the effect of $X(t)$ being at its pth quantile.

Another potentially useful transformation we do not pursue in this paper is $\hat{H}_t(x) = n^{-1}\sum_{i=1}^{n}\Phi\left[\frac{x - X_i(t)}{h_t}\right]$, where $\Phi(\cdot)$ denotes the standard normal cdf and h_t is a user chosen bandwith that can depend on t. The advantage of this transformation over the empirical cdf transformation is that future observations falling below [above] the minimum [maximum] value of the training data at a particular t are

not all assigned the value zero [one].

Due to the penalization used later when fitting the FGAM, parameter estimates can still be obtained when the design matrix has a column of zeros. However, we expect our transformation will improve both the numerical and statistical stability of our estimates. Note also that if there exists any pointwise transformation, $H_t(\cdot)$, such that $g\{E(Y_i|X_i)\} = \int_{\mathcal{T}} \beta(t) H_t\{X_i(t)\}\, dt$, then the FGAM will still hold; and similarly, for any model of the form (2.3) for a general transformation $G_t(\cdot)$. The FLM will hold only if $X_i(t)$ is transformed by H_t, but H_t is generally not known. Thus, the FGAM is invariant to transformations of the predictor, unlike the FLM.

2.2 Estimation

In this section, we present the estimation procedure for $F(\cdot, \cdot)$. First, we review P-spline type penalties and discuss penalized GLMs and the selection of smoothing parameters. We then describe the estimated surface and discuss construction of pointwise confidence bands for these estimates. We conclude the section by showing how to include additional functional and non-functional predictors in the model.

2.2.1 Roughness Penalties

Smoothing can be achieved by using row and column penalties as in Marx and Eilers[67]. The row penalty is $\lambda_1 \sum_{j=d_x+1}^{K_x} (\Delta_j^{d_x} \theta_{j,k})^2$, where $\Delta_j^{d_x} \theta_{j,k}$ is the d_xth difference of the sequence $\theta_{j-d_x,k}, \ldots, \theta_{j,k}$ (k held fixed). The column penalty is $\lambda_2 \sum_{k=d_t+1}^{K_t} (\Delta_k^{d_t} \theta_{j,k})^2$, where $\Delta_k^{d_t} \theta_{j,k}$ is the d_tth difference of the sequence $\theta_{j,k-d_t}, \ldots, \theta_{j,k}$ (j held fixed). Selection of the smoothing parameters λ_1 and λ_2 is

discussed in the next section.

Proceeding similarly to Marx and Eilers[68], we first place the $Z_{j,k}(i)$'s in a matrix as follows. Let $\mathbf{Z}_i = \text{vec}\{\mathbb{Z}(i)\}$ be the $_{K_xK_t}$-vector obtained by stacking the columns of $\mathbb{Z}(i) = [Z_{j,k}(i)]_{j=1,\ldots,K_x}^{k=1,\ldots,K_t}$, and let $\mathbb{Z} = [\mathbf{Z}_1\ \mathbf{Z}_2 \cdots \mathbf{Z}_N]^T$. The penalty matrix is given by

$$\mathbb{P} = \lambda_1 \mathbb{P}_1^T \mathbb{P}_1 + \lambda_2 \mathbb{P}_2^T \mathbb{P}_2, \tag{2.4}$$

with $\mathbb{P}_1 = \mathbb{D}_x \otimes \mathbb{I}_{K_t}$, $\mathbb{P}_2 = \mathbb{I}_{K_x} \otimes \mathbb{D}_t$ where $\mathbb{I}_p$ is the $p \times p$ identity matrix, $\otimes$ is the Kronecker product, and $\mathbb{D}_x$ and $\mathbb{D}_t$ are matrix representations of the row and column difference penalties with dimension $(K_x - d_x) \times K_x$ and $(K_t - d_t) \times K_t$, respectively. The parameter, d, denotes the prespecified degree of differencing. Note that additional penalties such as an overall ridge penalty could also be incorporated.

To incorporate the intercept, a leading column of ones must be added to $\mathbb{Z}$ and a leading column of zeros must be added to $\mathbb{P}_1$ and $\mathbb{P}_2$. Throughout the rest of the paper, this has been done unless otherwise indicated. When we do not wish to consider the intercept, $\mathbb{M}_{[-i,-j]}$ will denote the matrix $\mathbb{M}$ with its ith row and jth column removed and $\mathbf{v}_{[-i]}$ will denote the vector $\mathbf{v}$ excluding its ith entry.

2.2.2 Penalized GLMs and Smoothing Parameter Selection

Let the response vector, $\mathbf{Y}$, be from an exponential family with density having the form $f_Y(\mathbf{y}; \boldsymbol{\zeta}, \phi) = \prod_{i=1}^N \exp\left[\{y_i\zeta_i - b(\zeta_i)\}/a(\phi) + c(y_i, \phi)\right]$, where $\boldsymbol{\zeta}$ is the canonical parameter vector with components satisfying $\zeta_i = (b')^{-1}(\mu_i)$ and ϕ is the dispersion parameter. Parameterizing $E(\mathbf{Y}|\mathbb{X})$ as a standard GLM with known link function, $g(\cdot)$, let $\boldsymbol{\eta} := \mathbb{Z}\boldsymbol{\theta}$ and $\boldsymbol{\mu} := E(\mathbf{Y}|\mathbb{X})$, so that $\boldsymbol{\eta} = g(\boldsymbol{\mu})$. It is

easily seen that the constraint, $\sum_{i=1}^{N} \int_{\mathcal{T}} F\{X_i(t), t\}\, dt = 0$, is enforced by requiring $\mathbf{1}^T \mathbb{Z}_{[,-1]} \boldsymbol{\theta} = 0$. Formally, this is done by obtaining the QR decomposition of $(\mathbf{1}^T \mathbb{Z}_{[,-1]})^T = [\mathbf{Q}_1\ \mathbb{Q}_2] \begin{bmatrix} r_1 \\ \mathbf{0}_{K_x K_t - 1} \end{bmatrix}$, where $\mathbb{Q}_2$ has dimension $K_x K_t \times (K_x K_t - 1)$. The constrained optimization problem is then replaced by an unconstrained optimization (outlined below) over $\boldsymbol{\theta}_q$, where $\boldsymbol{\theta}_q$ is such that $\boldsymbol{\theta} = \mathbb{Q}_2 \boldsymbol{\theta}_q$. For notational simplicity, for any matrix $\mathbb{M}$, define $\widetilde{\mathbb{M}} = \mathbb{M}\mathbb{Q}_2$.

The penalized log-likelihood to be maximized is

$$l(\boldsymbol{\theta}_q; \lambda_1, \lambda_2) = \sum_{i=1}^{N} \log\{f_Y(y_i; \zeta_i, \phi)\} - \lambda_1 ||\widetilde{\mathbb{P}}_1 \boldsymbol{\theta}_q||^2 - \lambda_2 ||\widetilde{\mathbb{P}}_2 \boldsymbol{\theta}_q||^2. \tag{2.5}$$

The coefficients are estimated using penalized iteratively re-weighted least squares (P-IRLS). Specifically, at the $(m+1)th$ iteration we take

$$\widehat{\boldsymbol{\theta}}_{q,m+1} = \left(\widetilde{\mathbb{Z}}^T \widehat{\mathbb{W}}_m \widetilde{\mathbb{Z}} + \lambda_1 \widetilde{\mathbb{P}}_1^T \widetilde{\mathbb{P}}_1 + \lambda_2 \widetilde{\mathbb{P}}_2^T \widetilde{\mathbb{P}}_2\right)^{-1} \widetilde{\mathbb{Z}}^T \widehat{\mathbb{W}}_m \widehat{\mathbf{u}}_m, \tag{2.6}$$

where $\widehat{\mathbf{u}}_m$ is the current estimate of the adjusted dependent variable vector, $\mathbf{u}$, and $\widehat{\mathbb{W}}_m$ is the current estimate of the diagonal weight matrix, $\mathbb{W}$. The components of $\mathbf{u}$ are given by $u_i = \eta_i + (y_i - \mu_i) g'(\mu_i)$. The ith diagonal element of $\mathbb{W}$ is $w_{ii} = 1/\{V(\mu_i)[g'(\mu_i)]^2\}$, with $V(\mu_i) = b''(\zeta_i)$. To initialize the algorithm, use $\boldsymbol{\mu}_0 = \mathbf{Y}$ and $\boldsymbol{\eta}_0 = g(\mathbf{Y})$, adjusting y_i if necessary to avoid $\eta_i = \infty$.

To efficiently construct (2.6) and to detect rank deficiency, the following procedure is used. First, use the QR-decomposition to form $\mathbb{W}^{1/2}\widetilde{\mathbb{Z}} = \mathbb{Q}\mathbb{R}$ where $\mathbb{Q}$ is orthogonal, $\mathbb{R}$ is upper triangular, and $\mathbb{W}^{1/2} = \text{diag}(w_{11}^{1/2}, \ldots, w_{NN}^{1/2})$. Next, use the Choleski decomposition to obtain $\mathbb{Q}_2^T \mathbb{P} \mathbb{Q}_2 = \mathbb{L}^T \mathbb{L}$. Pivoting should be used here because $\mathbb{P}$ is positive semi-definite instead of positive definite. Now, from a singular value decomposition form $[\mathbb{R}^T\ \mathbb{L}^T]^T = \mathbb{U}\mathbb{D}\mathbb{V}^T$, where $\mathbb{U}$ and $\mathbb{V}$ are orthogonal and $\mathbb{D}$ is a diagonal matrix containing the singular values. At this point, we ensure identifiability by removing the columns and rows of $\mathbb{D}$ and the columns

of $\mathbb{U}$ and $\mathbb{V}$ corresponding to singular values that are less than the square root of the machine precision times the largest singular value (Wood[126], p. 183). It then follows that (2.6) can be obtained from $\widehat{\boldsymbol{\theta}}_{q,m+1} = \mathbb{V}\mathbb{D}^{-1}\mathbb{U}_1^T\mathbb{Q}^T\mathbb{W}^{1/2}\widehat{\mathbf{u}}_m$, where $\mathbb{U}_1$ is the sub-matrix of $\mathbb{U}$ satisfying $\mathbb{R} = \mathbb{U}_1\mathbb{D}\mathbb{V}^T$. At the final iteration, say M, our solution for $\boldsymbol{\theta}$ is given by $\widehat{\boldsymbol{\theta}} = \mathbb{Q}_2\widehat{\boldsymbol{\theta}}_{q,M}$ and it can be shown that this satisfies $\mathbf{1}^T\mathbb{Z}_{[,-1]}\widehat{\boldsymbol{\theta}} = 0$ as required (Wood[126], Sec. 1.8.1).

Generalized cross validation (GCV) can be used to choose the smoothing parameters; see Wood[124], Sec 4.5.4 for justification of its use for non-identity link GAMs. The GCV score for λ_1 and λ_2 is given by

$$\text{GCV}(\lambda_1, \lambda_2) = \frac{nD(\mathbf{Y}; \widehat{\boldsymbol{\mu}} : \lambda_1, \lambda_2)}{\{n - \gamma\text{tr}(\mathbb{H})\}^2}, \tag{2.7}$$

where $\mathbb{H}$ is known as the influence matrix and is related to the fitted values by $\widehat{\boldsymbol{\mu}} := g^{-1}(\mathbb{Z}\widehat{\boldsymbol{\theta}}_M) = g^{-1}(\mathbb{H}\mathbf{u}_M)$ and $D(\mathbf{Y}; \widehat{\boldsymbol{\mu}} : \lambda_1, \lambda_2)$ denotes the model deviance. The model deviance is defined to be twice the difference between the log-likelihoods of the saturated model, which has one parameter for each observation, and the given model. Formulas for the deviance for some common GLMs are given in McCullagh and Nelder[69], Sec. 2.3; for example, for an identity link GLM, $D(\mathbf{Y}; \widehat{\boldsymbol{\mu}} : \lambda_1, \lambda_2) = ||\mathbf{Y} - \mathbb{H}\mathbf{Y}||^2$. The constant $\gamma \geq 1$ is usually chosen to take values between 1.2 and 1.4 to combat the tendency of GCV to undersmooth. For additional safeguards against undersmoothing, lower bounds could also be placed on the smoothing parameters.

A choice must be made on the order in which the P-IRLS and the smoothing parameter selection iterations are performed. For what is termed outer iteration, for each pair of smoothing parameters considered, a GAM is estimated using P-IRLS until convergence. The other possibility, known as performance iteration, is to optimize the smoothing parameters at each iteration of the P-IRLS algorithm. The

latter approach can be faster than outer iteration; however, it is more susceptible to convergence problems in the presence of multicollinearity (Wood[126], Ch. 4).

Our model can conveniently be fit in `R` using the `mgcv` package (Wood[124],[128]). The details of how this is done are discussed in Section 2.2.6. We use outer iteration and Newton's method for minimizing the GCV score, the package defaults. Using this package also allows for many possible extensions (e.g. mixed effects terms, formal model selection, alternative estimation procedures, etc.) beyond the scope of the current paper. Our code is implemented in the R package `refund`.

2.2.3 Estimated Surface

For a given $\widehat{\boldsymbol{\theta}}$, we can evaluate the estimated surface at any grid of points in its domain. Let $\mathbf{X}$ be an arbitrary column vector of length n_1 taking values in the range of $X(\cdot)$ and $\mathbf{T}$ be the observation times or any vector of length n_2 taking values in $[0, 1]$. We let $\widehat{\mathbf{F}}$ denote the estimated surface evaluated on the mesh defined by $\mathbf{X}$ and $\mathbf{T}$. To obtain $\widehat{\mathbf{F}}$, let $\mathbb{B}_x$ be the $n_1 n_2 \times K_x$ matrix of x-axis B-splines evaluated at $\mathbf{X} \otimes \mathbf{1}_{n_2}$, i.e., $\mathbb{B}_x = \left[\mathbf{B}_1^X(\mathbf{X} \otimes \mathbf{1}_{n_2}) \cdots \mathbf{B}_{K_x}^X(\mathbf{X} \otimes \mathbf{1}_{n_2})\right]$, where $\mathbf{1}_n$ denotes a column vector of length n. Similarly, define $\mathbb{B}_t$ as the $n_1 n_2 \times K_t$ matrix of B-splines evaluated at $\mathbf{1}_{n_1} \otimes \mathbf{T}$. Next, define the $n_1 n_2 \times K_x K_t$ matrix

$$\mathbb{B} = (\mathbb{B}_x \otimes \mathbf{1}_{K_t}^T) \odot (\mathbf{1}_{K_x}^T \otimes \mathbb{B}_x), \tag{2.8}$$

where $\odot$ denotes element-wise matrix multiplication. The estimated surface is then given by $\widehat{\mathbf{F}} = \mathbb{B}\widehat{\boldsymbol{\theta}}_{[-1]}$.

2.2.4 Standard-Error Bands

For a response from any exponential family distribution, one simple way to construct approximate, pointwise confidence bands for $\widehat{F}(x,t)$ conditional on the estimated smoothing parameters is to use a sandwich estimator in the same manner as Hastie and Tibshirani[42], Section 6.8.2 and Marx and Eilers[67]. However, we found through our simulation studies that these intervals do not have adequate coverage for our model, a result also noticed for univariate GAMs in Wood[125]. This is because these intervals assume $\widehat{\boldsymbol{\theta}}$ is unbiased, which will not be the case when $\boldsymbol{\theta} \neq \mathbf{0}$, due to the penalization involved in the estimation.

To overcome the bias in the parameter estimation, we use the Bayesian approach of Wahba[117]. Using the improper prior $\pi(\boldsymbol{\theta}) \propto \exp\left(-\boldsymbol{\theta}^T \mathbb{P} \boldsymbol{\theta}/2\right)$, it can be shown that

$$\boldsymbol{\theta} | \mathbb{Z}^T \mathbb{W} \mathbf{u}, \lambda_1, \lambda_2 \sim N\left([\mathbb{Z}^T \mathbb{W} \mathbb{Z} + \mathbb{P}]^{-1} \mathbb{Z}^T \mathbb{W} \mathbf{u}, [\mathbb{Z}^T \mathbb{W} \mathbb{Z} + \mathbb{P}]^{-1} \phi \right),$$

see e.g. Wood[126], Sec. 4.8. To estimate $\mathbb{W}$, we use the estimated weight matrix at the final P-IRLS iteration, $\widehat{\mathbb{W}}_M$. If it is necessary to estimate the dispersion parameter, ϕ, we use $\widehat{\phi} = \sum_{i=1}^{n} V(\hat{\mu}_i)^{-1}(y_i - \hat{\mu}_i)^2/[N - \mathrm{tr}(\mathbb{H})]$. Letting $\mathbb{V}_{\widehat{\boldsymbol{\theta}}} = (\mathbb{Z}^T \widehat{\mathbb{W}}_M \mathbb{Z} + \mathbb{P})^{-1} \widehat{\phi}$ and recalling that the estimated surface is given by $\widehat{\mathbf{F}} = \mathbb{B}\widehat{\boldsymbol{\theta}}_{[-1]}$, where $\mathbb{B}$ is defined in (2.8), the variance of $\widehat{\mathbf{F}}$ is given by $\mathrm{var}\left[\widehat{\mathbf{F}}\right] = \mathbb{B}\mathbb{V}_{\widehat{\boldsymbol{\theta}}_{[-1,-1]}}\mathbb{B}^T$. Taking $\widehat{\mathbf{F}} \pm 2\left\{\mathrm{diag}\left(\mathrm{var}\left[\widehat{\mathbf{F}}\right]\right)\right\}^{1/2}$ gives approximate 95% empirical Bayesian confidence bands for $\mathbf{F}$.

These Bayesian intervals have a nice frequentist property "across the function": in repeated random experiments with the same F, the observed coverage probabilities averaged over the observation points will tend to be close to the nominal coverage level. This property was borne out in several papers including Wahba[117]

and Nychka[79] for the case of smoothing splines and Wood[125] for thin-plate regression splines. Theoretical explanations for the property for generalized additive models were recently provided in Marra and Wood[65]. It will be examined for the FGAM through a simulation study in Section 2.3.2.

Depending on the application, a particular linear combination of the elements of $\widehat{\mathbf{F}}$ may be of interest. If we let $\mathbf{c}$ be a vector of the same length as $\widehat{\mathbf{F}}$, then we can also construct confidence bands of the form $\mathbf{c}^T\widehat{\mathbf{F}} \pm 2\left\{\mathbf{c}^T\left(\text{var}\left[\widehat{\mathbf{F}}\right]\right)\mathbf{c}\right\}^{1/2}$. For example, this could be used to determine approximately whether two observed curves have significantly different effects on the response at a particular value of t. Under a null hypothesis of $H_0 : \boldsymbol{\theta} = \mathbf{0}$, $\widehat{\boldsymbol{\theta}}$ is unbiased and we can use the sandwich estimator for the variance, $\mathbb{V}_f = \mathbb{V}_{\widehat{\boldsymbol{\theta}}}\mathbb{Z}^T\widehat{\mathbb{W}}_M\mathbb{Z}\mathbb{V}_{\widehat{\boldsymbol{\theta}}}/\hat{\phi}$, to conduct approximate hypothesis tests for subsets of $\boldsymbol{\theta}$. For example, we can construct surfaces of approximate t-statistics by scaling the estimated surface values by the reciprocal of their standard error (the diagonal elements of $\mathbb{V}_f$).

For any pointwise transformation, $H_t(\cdot)$, of the predictor used (including $H_t(x) = x$), it is of interest to test whether $\partial^2/\partial h^2 F(h,t) = 0$ for all h and t, since this implies $F\{H_t(x), t\} = \beta(t)H_t(x)$ for some function $\beta(\cdot)$. Since derivatives of B-splines are simple to compute, an estimate of the second derivative of the surface and the Bayesian confidence intervals for the second derivative are easily obtained by replacing $\mathbb{B}_x$ in (2.8) with evaluations of the second derivatives of the x-axis B-splines evaluated at the same points used for $\mathbb{B}_x$. While we cannot use our confidence bands for global inferences of this type, they do provide a rough heuristic for the desired test. We consider more formal tests of this hypothesis in Chapter 4. Marra and Wood[65] provides some evidence that coverage can be improved slightly by including the intercept when calculating the proposed intervals

(which slightly changes the interpretation of the intervals as well). In our numerical studies, which we will discuss in detail shortly, we found that for FGAM the Bayesian confidence bands that did not include the intercept provided adequate coverage.

2.2.5 Multiple Predictors

Because of the modularity of penalized splines (Ruppert et al.[99]), including multiple functional predictors as well as scalar predictors in the model is straightforward. Each additional functional predictor requires that two more smoothing parameters be selected. We will outline the procedure for the case of two functional covariates [say $X_1(\cdot)$, $X_2(\cdot)$] and one scalar covariate (say W). The model is $g\{E(Y_i|X_{i,1}, X_{i,2}, W_i)\} = \theta_0 + \int_{\mathcal{T}_1} F_1\{X_{i,1}(t), t\}dt + \int_{\mathcal{T}_2} F_2\{X_{i,2}(t), t\}dt + F_3(W_i)$, and both $X_1(\cdot)$ and $X_2(\cdot)$ can be transformed by their empirical cdfs. Further extensions are similar. As before, we use B-spline bases for both axes for both functional predictors and now also for W. One must also choose degrees of differencing to be used for each penalty. Let $\mathbb{Z}^{(1)}$ and $\mathbb{Z}^{(2)}$ denoted the matrices of integrated tensor product B-splines for X_1 and X_2, respectively. Similarly, define $\mathbb{P}^{(1)}$ and $\mathbb{P}^{(2)}$ [see (2.4)]. Let $\mathbb{B}^{(W)}$ be the matrix of W B-splines evaluated at the observed values of W and let $\boldsymbol{\theta}^{(W)}$ be the corresponding vector of B-spline coefficients for W. The penalty matrix for the smooth of W is given by $\mathbb{P}^{(W)} = \lambda_w \mathbb{D}_w^T \mathbb{D}_w$, where $\mathbb{D}_w$ is the differencing matrix for W and λ_w is its smoothing parameter. For identifiability, add the constraint $\mathbf{1}^T \mathbb{B}^{(W)} \boldsymbol{\theta}^{(W)} = 0$ (the usual constraint for each functional component in a standard additive model). We place the same constraint on both functional predictors as in the previous section. Thus, we have three total

constraints. Construct

$$\mathbb{Z} = \begin{bmatrix} \mathbf{1} & \mathbb{B}^{(W)} & \mathbb{Z}^{(1)} & \mathbb{Z}^{(2)} \end{bmatrix}, \quad \mathbb{P} = \text{diag}(0, \mathbb{P}^{(W)}, \mathbb{P}^{(1)}, \mathbb{P}^{(2)}), \text{ and } \boldsymbol{\theta} = (1, \boldsymbol{\theta}^{(W)}, \boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)})^T.$$

To accommodate a linear effect of the covariate W, replace $\mathbb{B}^{(W)}$ in $\mathbb{Z}$ with the observed values of W and replace $\mathbb{P}^{(W)}$ with zero in the above formula for $\mathbb{P}$.

Note that it is also possible to have a linear effect for some functional predictors and additive effects for others; e.g. a model of the form $g\{E(Y_i|X_i)\} = \theta_0 + f(W_i) + \int_{\mathcal{T}_1} \beta(t)X_{1i}(t)dt + \int_{\mathcal{T}_2} F\{X_{2i}(t), t\}dt$. Using the roughness penalty approach for estimating FLMs mentioned in Section 2.3.1, this can be implemented by making straightforward changes to $\mathbb{Z}^{(1)}$ and $\mathbb{P}^{(1)}$ (see Ramsay and Silverman[91], Ch. 15 for details).

2.2.6 Fitting FGAM in `R`

Let $\mathbb{X}$ denote an $N \times J$ matrix of the observed measurements of the functional predictor, where N is the number of sampled curves and J is the number of measurements for each curve. Let $\mathbb{T}$ denote the $N \times J$ matrix of observation times for the predictor curves. Let $\mathbb{L}$ denote the $N \times J$ matrix of quadrature weights to use in our numerical integration of the surface $F(x, t)$ and let $\mathbf{y}$ be the N-vector of observed response values. The simplest FGAM (using all the function defaults) and without an intercept is specified in `refund` (Crainiceanu et al.[16]) by

```
fgam(y~af(X,xind=T)-1).
```

The interface is meant to conveniently extend the functions `lm` and `glm` in base `R`. As in those functions, the `-1` is included so that no intercept is fit; this is done here

to simplify the notation. The function `af` in the model formula is used to specify that the predictor X be fit in the FGAM form (1.2). See the documentation of either function for how to specify the spline bases used, how smoothing parameters are estimated, what penalties are used, etc. A functional predictor can also be fit as an FLM by using the function `lf` in the formula specification. Also available are functions `vis.fgam` for visualizing FGAM fits and `predict.fgam` for predictions using an FGAM fit returned by a called to `fgam`.

The `fgam` function acts as a convenience wrapper for the `gam, gamm`, or `bam` functions in package `mgcv` (Wood[126]). To understand what is being done by that package, an equivalent call (with slightly different defaults) to fit the above FGAM in `mgcv` is

```
gam(y~te(X,T,by=L)-1),
```

where `te` specifies a tensor product smooth. The variables in the `by` argument to `te` are treated as the covariates in a varying coefficient model. To make this association more explicit, a generalized varying coefficient model has the form (e.g., see Wood[126], p. 169).

$$g(\mu_i) = \theta_0 + f_1(x_{i1})x_{i2} + f_2(x_{i3}, x_{i4})x_{i5} + f_3(x_{i6})x_{i7} + \ldots$$

As a special case, consider

$$g(\mu_i) = \theta_0 + f(x_{i1}, x_{i2})x_{i3} + f(x_{i1}, x_{i2})x_{i4} + \ldots + f(x_{i1}, x_{i2})x_{i,J+2},$$

so each covariate, $x_{i3},\ x_{i4}, \ldots, x_{i,J+2}$ has the same bivariate varying coefficient. Now suppose we have $x_{i1} \equiv X_i(t_j),\ x_{i2} \equiv t_{ij} \equiv t_j\ , x_{ip} \equiv l_{ij} \equiv l_j; p = j + 2; j = 1, \ldots, J$ where the l_j's are quadrature weights. Note that `mgcv` treats both variables

t and l as if they depend on $i = 1, \ldots, N$ though they do not for the FGAM. We now arrive at

$$g(\mu_i) = \theta_0 + \sum_{j=1}^{J} f(x_{i1}, x_{i2}) x_{i,j+2} = \theta_0 + \sum_{j=1}^{J} f\{X_i(t_j), t_j\} l_j \approx \theta_0 + \int_{\mathcal{T}} f\{X_i(t), t\}\, dt,$$

so `mgcv` is fitting the model

$$E(Y_i|X_i) = \sum_{j=1}^{J} F(x_{ij}, t_{ij}) l_{ij} = \sum_{j=1}^{J} \sum_{k=1}^{K_x} \sum_{m=1}^{K_t} \theta_{km} B_k^X(x_{ij}) B_m^T(t_{ij}) l_{ij},$$

where as in the paper, $B_k^X(\cdot)$ denotes the kth B-spline for the x-axis and K_x is the dimension of the basis for X (with equivalent definitions for the t-axis).

The matrix $\mathbb{B}_T$ which consists of $J \times K_x$ blocks of size $N \times K_t$ each is formed in `mgcv`. The (i, j) entry in the (m, n) block of $\mathbb{B}_T$ is given by $B_n^X(x_{im}) B_j^T(t_{im})$. The design matrix used for the smooth is then the $NJ \times K_x K_t$ matrix

$$\mathbb{D} = \text{diag}[\text{vec}(\mathbb{L})]\mathbb{B}_T$$

The package enforces one constraint at this point because the row sums of the `by` variable matrix are constant ($\mathbb{L}\mathbf{1} = \mathbf{0}$). The constraint is $\mathbf{1}^T \mathbb{D} \boldsymbol{\theta} = \mathbf{0}$. How to implement this constraint during fitting and every other detail of the estimation procedure used by `mgcv` has already been discussed in this chapter.

The default smoothing method for a tensor product smooth in mgcv is cubic regression splines, so the `bs` argument to `te` must be specified as `'ps'` for P-splines to be used. The `m` argument to mgcv specifies both the order of the spline and the order of the penalty. For P-splines, `m` can be specified as a list with length equal to the number of marginal bases. The argument `k` is a vector specifying the dimension of each marginal basis.

As an example, say we have an N-vector of responses $\mathbf{y}$, the $N \times J$ matrix of observed functional predictors $\mathbb{X}$ with observation times occurring at equally

spaced points in $[0, 1]$, and that wish to use the midpoint rule aka rectangle method for approximating the integral. If we wish to use 10 cubic basis functions for the x-axis, 15 4th order basis functions for the t-axis, a second order difference penalty for the x-axis and a third order difference penalty for the t-axis, then the code to fit the FGAM with intercept is as follows

```
T=matrix( seq(0,1,l=J) ,N,J)
L=matrix(1/J,N,J)
fit=gam( y~te(X,T,by='L',bs='ps',k=c(10,15),m=list(c(2,2),c(4,3))) )
```

Note that in the documentation for P-spline smooths in mgcv (see `?p.spline`), it is noted that a smooth term of the form

```
s(x,bs="ps",m=c(2,3))
```

"specifies a 2nd order P-spline basis (cubic spline), with a third order difference penalty..." Though it is not standard for a cubic spline to be called 2nd order, this does seem to be what is implemented within `mgcv` and so we follow along with this specification.

Additional functional predictors are added by including additional `te` terms. Responses from other exponential family distributions are handled in the exact same way as the `glm` function in R.

2.3 Simulation Experiment

In this section, we perform simulations to assess the empirical performance of our FGAM. We first assess the ability of our FGAM to predict out-of-sample data

in the Gaussian response case and compare its performance with several other functional regression models. Next, we examine the coverage properties of the empirical Bayesian confidence bands proposed in Section 2.2.4.

To generate the data, we created 1000 replicate data sets each consisting of N curves sampled at 200 equally-spaced points in $[0,1]$ as follows: Let $X_i(t) = \sum_{j=1}^{J} \gamma_j [Z_{1ij}\phi_{1j}(t) + Z_{2ij}\phi_{2j}(t)]$ where $Z_{hij} \sim N(0,1)$, $\phi_{1j}(t) = \sqrt{2}\cos(\pi jt)$, $\phi_{1j}(t) = \sqrt{2}\sin(\pi jt)$, and $\gamma_j = \frac{2}{j}$; $h = 1,2$; $i = 1,\ldots,N$; $j = 1,\ldots,J$. We consider two values for J, $J = 5$ and $J = 500$, the former resulting in much smoother predictor trajectories. We examine two cases for the true surface, $F(x,t)$, one where the FLM holds, $F(X(t),t) = \beta(t)X(t)$ and the other where it does not. For the linear true model, $F(x,t) = xt$. For the nonlinear true model, we use $F(x,t) = -.5 + \exp\left[-(\frac{x}{5})^2 - (\frac{t-.5}{.3})^2\right]$, which looks like a hill or bivariate normal density.

The error variance changes with each sample so that the empirical signal to noise ratio (SNR) defined by SNR$=\frac{s^2_{\hat{y}}}{\sigma^2}$, where $s^2_{\hat{y}} = \frac{1}{N-1}\sum_{i=1}^{N}\left[\int_{\mathcal{T}} F(X_i(t),t)\,dt - N^{-1}\sum_{i=1}^{N}\int_{\mathcal{T}} F(X_i(t),t)\,dt\right]^2$ remains constant. We consider the values SNR$= 1,2,4,8$ in our simulations.

2.3.1 Out-of-Sample Predictive Performance

We fit FGAM and compare its out-of-sample predictive accuracy with three other popular functional regression models, the FLM, the kernel estimator of Ferraty and Vieu[29], and the functional additive model (FAM) of Müller and Yao[76]. The coding used in our analyses was done in `R` (R Core Team[88]). The `fda` package (Ramsay et al.[93]) implements the standard tools of functional data analysis in `R`.

As an initial step in fitting our model, the FLMs and the FAM, we use this package to smooth the data using B-spline basis functions and a roughness penalty with smoothing parameter chosen by GCV.

There are two main approaches for estimating the coefficient function $\beta(\cdot)$ for a FLM. The first uses smoothing or penalized splines and the second uses a functional principal component analysis (fPCA). We refer to these as FLM1 and FLM2, respectively. These models can be fit in `R` using the fda package, more specifically, the functions `fRegress` for FLM1 and `pca.fd` for FLM2. See Ramsay et al.[92], Chapter 9 for computational details. For FLM1, we choose the smoothing parameter by minimizing GCV. For FLM2, we conduct a functional principal component analysis with a constant, light amount of smoothing and retain enough components for each simulation scenario to explain 90% of the total variability of the functional predictor. Once the scores are estimated, the final step to estimating FLM2 is fitting an unpenalized linear model in the scores.

To fit the FAM, we use the same number of principal component scores and the same estimation procedure as for FLM2. The difference comes in the next step, where a generalized additive model is fit using the scores as predictors. To estimate the GAM, we use the default settings in the mgcv package and 11 basis functions for each additive term.

The final model we fit is described in detail in Ferraty and Vieu[29], Ch. 5. The response is predicted by the nonlinear operator $r(X) := E(Y|X)$. This operator is estimated by a functional extension of the Nadaraya-Watson kernel estimator:

$$\widehat{r}(X) = \frac{\sum_{i=1}^{N} Y_i \, K\left\{h^{-1} d(X, X_i)\right\}}{\sum_{i=1}^{N} K\left\{h^{-1} d(X, X_i)\right\}}, \tag{2.9}$$

where K is an asymmetrical kernel with bandwidth h and d is a semimetric. Continuity or Lipschitz continuity of the regression operator in the semimetric

is assumed. We used the quadratic kernel, $K(u) = \frac{3}{4}(1-u^2)1_{[-1,1]}(u)$, and the semimetric $d(X_i, X_{i'}) = [\int_{\mathcal{T}}\{X_i(t) - X_{i'}(t)\}^2 dt]^{1/2}$. Code for fitting this model with automatic bandwidth selection can be obtained from: `http://www.math.univ-toulouse.fr/staph/npfda`. Note the differences in the assumptions and complexities of these three models: the simplest model assumes the response is linear in the functional predictor, the FGAM lessens the restrictions to additivity in the functional predictor, FAM restricts to additivity in a linear projection of the functional predictor, and the kernel estimator makes no restrictions on the form of the regression function other than continuity.

Each training set contained 67 curves and 33 curves were used for the test set. The performance of the models was measured by the out-of-sample RMSE$=\left[33^{-1}\sum_{i\in\{\text{test set}\}}(y_i - \widehat{y}_i)^2\right]^{1/2}$. We report results for both the FGAM fit to the original data and the FGAM fit after X has been transformed using the empirical cdf transformation given in (2.3). In both cases, six cubic B-splines were used for the x-axis and seven cubic B-splines were used for the t-axis with second degree difference penalties for both axes. The tuning parameter, γ, for the GCV criterion (2.7) was taken to be 1.0 in all cases. The mgcv package requires that the number of coefficients to estimate be less than the sample size, so we must have the product of the dimensions of the bases be less than the sample size minus one (for the intercept). The results of the simulations are summarized in Figure 2.2.

The figure reports the median RMSE's across the 1000 simulations for each scenario and model. We see that the FGAM loses little to the FLM in terms of predictive accuracy when the FLM is the true model and provides substantial improvements in the case when the FLM is not the true model. In fact, all the models perform quite similarly in the linear true model case with the exception

of the Ferraty and Vieu model (2.9) which performs considerably worse. In the nonlinear true model case, we see that fitting an FGAM to the transformed data performs slightly better than fitting an FGAM to the original curves and that in general the FGAM offers significant advantages over all the other models. As expected, the differences in performance between the different models become more pronounced as the fixed empirical signal to noise ratio increases.

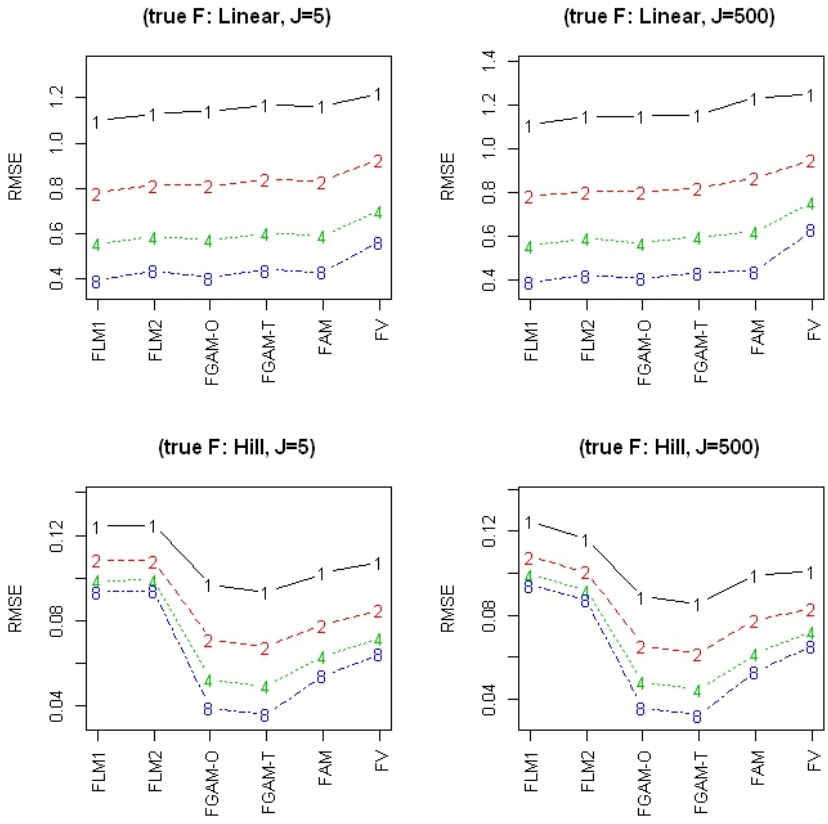

Figure 2.2: Median RMSE across 1000 simulations for six different functional regression models, four different empirical signal to noise ratios and rough (J=500) and smooth (J=5) predictor functions. a) Linear true model, b) Nonlinear ("Hill") true surface.

	N=100		N=500	
True Surface	SNR=2	SNR=4	SNR=2	SNR=4
Linear	0.9746	0.9684	0.9704	0.9702
Nonlinear	0.9597	0.9665	0.9613	0.9592

Table 2.1: Mean ACP across 500 simulations for nominal coverage probability 0.95.

2.3.2 Bayesian Confidence Band Performance

We now assess the average coverage probabilities (ACP) of the confidence bands from Section 2.2.4. The observed ACP for the ith simulation is given by

$$ACP = \frac{1}{625}\sum_{j=1}^{25}\sum_{k=1}^{25} I\{F(x_j^{(i)}, t_k^{(i)}) \in C_{.95}(x_j^{(i)}, t_k^{(i)})\},$$

where $\{(x_j^{(i)}, t_k^{(i)});\ j,k = 1,\ldots,25\}$ are a subset of the $N \times 200$ observed $(X(t), t)$ values for the ith simulation and $C_{.95}(x_j^{(i)}, t_k^{(i)})\}$ is the entry of $\widehat{\mathbf{F}}^{(i)} \pm 2\{\text{diag}\left(\text{var}\left[\widehat{\mathbf{F}}^{(i)}\right]\right)\}^{1/2}$ corresponding to $(x_j^{(i)}, t_k^{(i)})$. We consider two values for the sample size, $N = 100$ (combining the training and test sets from the previous section) and $N = 500$, the same true surfaces from the previous section, and two values for the empirical signal to noise ratio, two and four. For both the x and t axes, we use nine basis functions, cubic B-splines, and a second order difference penalty. We report results for the FGAM fit without an intercept to the untransformed predictor curves with $J = 500$. The results for $J = 5$ were nearly identical.

To reduce the number of times that the confidence bands are evaluated at points outside the region jointly defined by the observed $(X_i(t_j), t_j)$ values, only grid points that are inside the convex hull defined by the observed values for each simulation are used in the calculation of mean ACP. A final modification is necessary to account for the identifiability constraint imposed on the FGAM. To do this, we fit the FGAM (including the constraint) with negligible amounts of

smoothing to the true $E(Y_i|X_i)$ values (without noise) and take the fitted values to be the true responses. The mean ACP across the 500 simulations is displayed in Table 2.1 for each simulation scenario.

We see from the table that the coverage is fairly close to the nominal level of 0.95, though there is a slight problem with over-coverage in all the scenarios. Further analysis shows that the average estimated Bayesian standard errors for the surface are larger than the Monte Carlo standard deviation of the estimated surface, which is causing in the over-coverage. This is a biproduct of the Bayesian intervals trying to correct for the smoothing bias inherent in nonparametric regression. Recall that these intervals do not account for uncertainty in the estimation of λ_1 and λ_2. If more precise confidence bands are required, alternatives such as bootstrapping could be employed; see Wood[125], Sec. 4. Another possibility is a fully Bayesian analysis. These results indicate that it is safe to use the Bayesian confidence bands to make inferences about the true surface $F(x,t)$. We additionally ran a subset of these simulation scenarios while computing the confidence bands using the sandwich estimator of the variance of the estimated surface(results not included) and found there could be substantial under-coverage in the nonlinear true model case as a result of bias due to smoothing.

2.4 Application to Diffusion Tensor Imaging Dataset

We now assess the performance of our model on a DTI tractography study. DTI is a technique for measuring the diffusion of water in tissue. Water diffuses differently in different types of tissue, and measuring these differences allows for detailed images to be obtained. Our dataset comes from a study comparing certain white

matter tracts of multiple sclerosis (MS) patients with control subjects. MS is a central nervous system disorder that leads to lesions in the white matter of the brain which disrupts the ability of cells in the brain to communicate with each other. This dataset was previously analyzed in Goldsmith et al.[32] and Greven et al.[38].

The result of the DTI tractography, is a 3×3 symmetric, positive definite matrix (equivalently, a three dimensional ellipsoid) that describes diffusion at each desired location in the tract. We consider three functions of the estimated eigenvalues from these matrices: fractional anisotropy, parallel diffusivity, and perpendicular diffusivity. Fractional anisotropy measures the degree to which the diffusion is different in directions parallel and perpendicular to the tract, with zero indicating an isotrophic diffusion. More precisely, if the eigenvalues of the ellipsoid are given by $\lambda_1, \lambda_2, \lambda_3$, fractional anisotropy is equal to $\Big[3\{(\lambda_1 - \bar{\lambda})^2 + (\lambda_2 - \bar{\lambda})^2 + (\lambda_3 - \bar{\lambda})^2\}/\{2(\lambda_1^2 + \lambda_2^2 + \lambda_3^2)\}\Big]^{1/2}$, where $\bar{\lambda} = (\lambda_1 + \lambda_2 + \lambda_3)/3$. Parallel (or axial or longitudinal) diffusivity is the largest eigenvalue of the ellipsoid. Perpendicular diffusivity is an average of the two smaller eigenvalues. See Mori[73] for an overview of DTI.

Standard magnetic resonance imagining is used for diagnosing MS, but it is believed that the extra information provided by the tract profiles produced from DTI can be used to understand the disease process better. As an example of the types of effects we could investigate with our model, it has been found (Reich et al.[94]) that parallel diffusivity is increased along the corticospinal tracts of people with MS. We would hope to see this effect if we were using parallel diffusivity measurements along that tract to predict MS status. We consider the corpus callosum tract in our analysis because it is related to cognition.

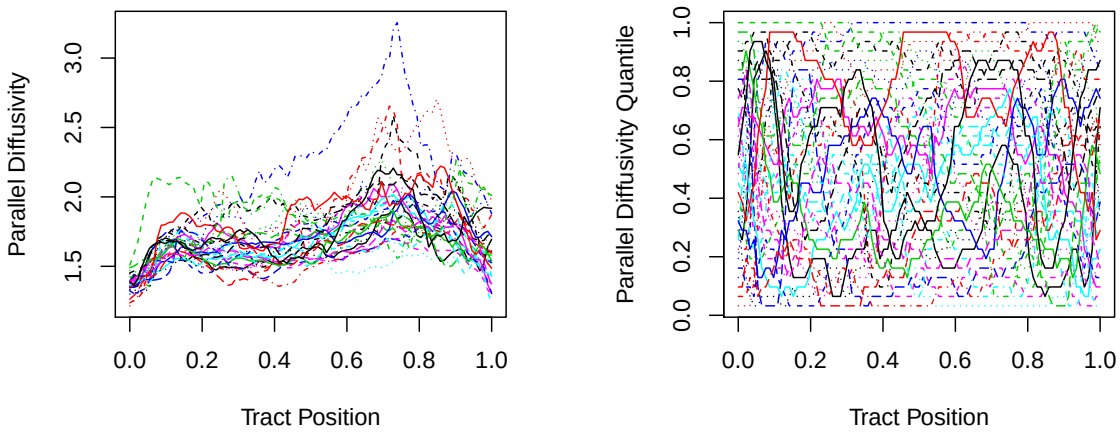

Figure 2.3: a) Observed parallel diffusivity along the corpus callosum tract for a sample of MS patients. b) Parallel diffusivity along the corpus callosum tract transformed by its empirical cdf for the same patients.

As an illustration of the FGAM, we fit our model using each of the three diffusion measures separately and compare the results with the same models introduced in the previous section. We also compare using the original curves as the predictor (1.2) with using the empirical cdf of the curves (2.3). Figure 2.3 contains plots of the parallel diffusivity measurements along the corpus callosum tract and the corresponding empirical cdf-transformed values for each subject in the training set.

Throughout the analysis, when fitting the FGAM, we use cubic B-splines with second-order difference penalties, six B-splines for the $x\,(p)$-axis, and seven B-splines for the t-axis. We found our results to be insensitive to these choices, and for brevity we do not include results for other values considered. Throughout this section, γ in (2.7) is taken to equal 1.4. To evaluate the performance of the models, we examine their leave-one-curve-out prediction error. We repeatedly fit each model using all the samples except one and then use the fit to predict the left-out sample. This process is repeated until every sample has been left-

out once. Our performance measure is the root mean squared error, defined as RMSE=$\left[N^{-1}\sum_{i=1}^{N}(y_i - \widehat{y}_{(i)})^2\right]^{1/2}$, where $\widehat{y}_{(i)}$ is the predicted value of the ith response value when this sample is left out of the estimation.

2.4.1 Predicting PASAT Score

The first variable we predict is the result of a Paced Auditory Serial Addition Test (PASAT), a cognitive measure taking integer values between 0 and 60. The subject is given numbers at three second intervals and asked to add the current number to the previous one. The final score is the total number of correct answers out of 60. MS patients often perform significantly worse than controls on this test. Since the corpus callosum is known to play a role in cognitive function, we might expect to see that the functional measurements along this tract have a significant impact in forecasting PASAT score. The PASAT was only administered to subjects with MS. One subject with peculiar tract profiles was removed for simplicity and to avoid dealing with missing values.

The estimated surface $\widehat{F}(p,t)$ [see (2.1)] is shown in Figure 2.4(a) for transformed parallel diffusivity. Figure 2.4 b) shows a contour plot of the observed pseudo-t statistics discussed in Section 2.2.4. We can see from this figure that parallel diffusivity for tract positions around $0.4 - 0.6$ appears to be influential on the predicted response; subjects in the middle quantiles for this measurement at these positions are more likely to score higher on the PASAT, while the opposite is true for subjects in the upper quantiles at this location.

Figure 2.5 shows an example of a slice of the estimated surface when the untransformed curves are used for a fixed x value (left) and for a fixed position along

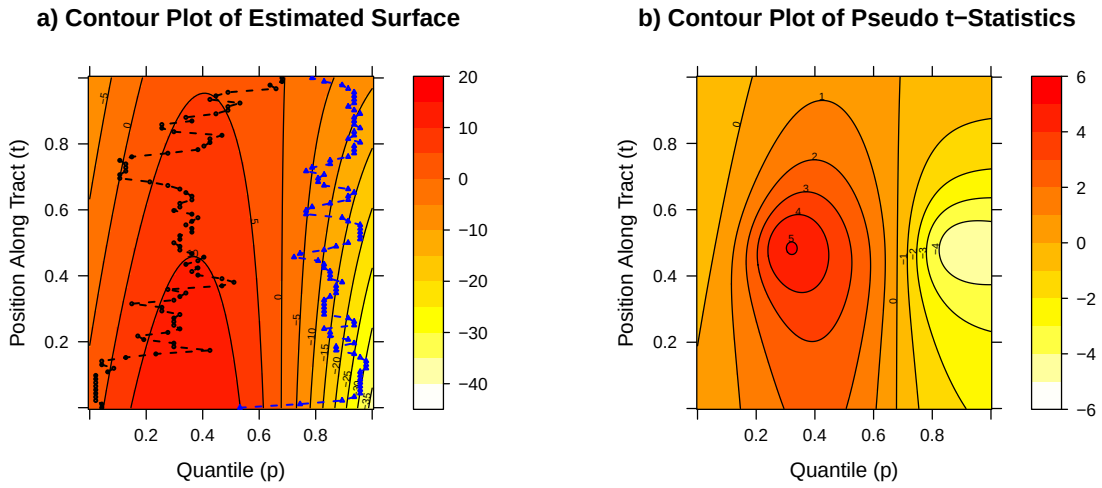

Figure 2.4: a) Contour plot of the estimated surface, $\widehat{F}(p,t)$ [see (2.1)], for transformed parallel diffusivity along the corpus callosum tract. Also included are the transformed parallel diffusivity measurements for two subjects. b) Contour plot of pseudo t-statistics (estimated surface value divided by its standard error). The response is PASAT score.

the tract, t, (right). Parallel diffusivity along the corpus callosum is used as the predictor in these plots which also include twice standard error bands based on the sandwich estimator described earlier. Figure 2.5 also shows the same slices for the estimated second derivative of the surface with respect to t. This can give us a rough idea of whether the linear model is sufficient. In practice, we look at these plots for a representative sample of values with both the predictor value fixed and with the position fixed. We see that the second derivative is significantly non-zero in some regions, which suggests inadequacy of using an FLM in the untransformed predictors.

Table 2.2 reports out-of-sample RMSE from separately using each of the three different diffusivity measurements along the corpus callosum tract as predictors in the five models under consideration. Here, using FGAM with the empirical cdf transformation (FGAM-T) led to improved forecasting accuracy compared to using the raw measurements as predictors (FGAM-O). In fact, FGAM-T (2.3) has lower

Measurement	FGAM-O	FGAM-T	FLM1	FLM2	FV	FAM
Perp. Diffusivity	12.22	10.46	10.98	11.27	11.16	11.71
Frac. Anisotropy	12.55	11.60	11.87	11.91	12.11	12.70
Para. Diffusivity	11.94	12.09	12.32	12.24	11.97	11.86

Table 2.2: Leave-one-curve-out RMSEs for the three different functional predictors of PASAT score using the following models: FGAM using the original curves (FGAM-O), FGAM using the empirical cdf transformation [FGAM-T, (2.3)], FLM1, FLM2, FV (2.9), and FAM.

out-of-sample RMSE than both FLMs for all the functional predictors considered, indicating that a linear model may be too restrictive in this application. Our FGAM-T compares favourably with the functional kernel regression model (2.9) and the FAM, showing better performance when either perpendicular diffusivity or fractional anisotropy are used as predictors. Though the kernel estimator provided slightly improved predictions in the parallel diffusivity case, the complex nature of its fit makes visualization difficult, so it is less useful than the FGAM for helping us understand the relationship between the DTI measurements and the PASAT scores.

2.4.2 Predicting MS status: Logistic Link

We now consider classifying the disease status of subjects. Since the PASAT was only given to the subjects with MS, our sample size is now 88 and includes controls. We include results using the untransformed curves only. The results using the quantile transformation were similar. We again use the leave-one-curve-out procedure described earlier. Fitting the FGAM resulted in the estimated surface displayed in Figure 2.1 when perpendicular diffusivity is used as the predictor. The observed perpendicular diffusivity for two subjects is overlaid on the plot; recall

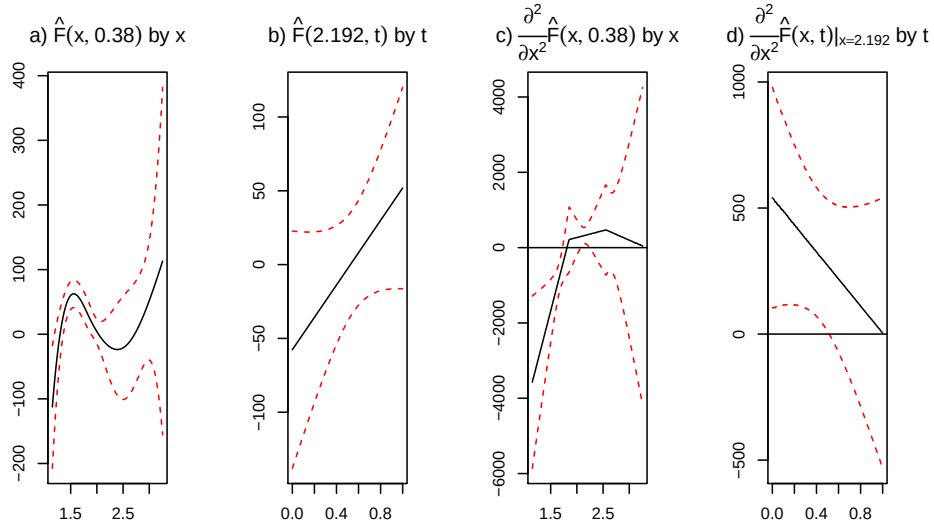

Figure 2.5: A sample of slices of the estimated surface [plots a) and b)] and estimated second derivative surface [c) and d)] for fixed tract positions [a) and c)] and fixed untransformed actual predictor [b) and d)] along with the corresponding Bayesian confidence bands for parallel diffusivity with PASAT score as the response variable.

the interpretation given in the introduction. It appears that the predictor values at the end of the tract corresponding to $t = 1$ have a strong influence in predicting disease status. Subjects in the lower range for perpendicular diffusivity at this end of the tract seem to be less likely to be classified as having MS, whereas subjects in the upper range at this position are more likely to have MS. Models were also fit using fractional anisotropy and parallel diffusivity as predictors. A fourth model was considered that included a nonparametric component for the subject's age in addition to using perpendicular diffusivity. Figure 2.6 contains a plot of the ROC curves for these fitted models. The model using fractional anisotropy performs almost universally worse than the other three models. None of the other three models considered perform universally better than the others. Including age as a covariate in the model with perpendicular diffusivity did not improve performance.

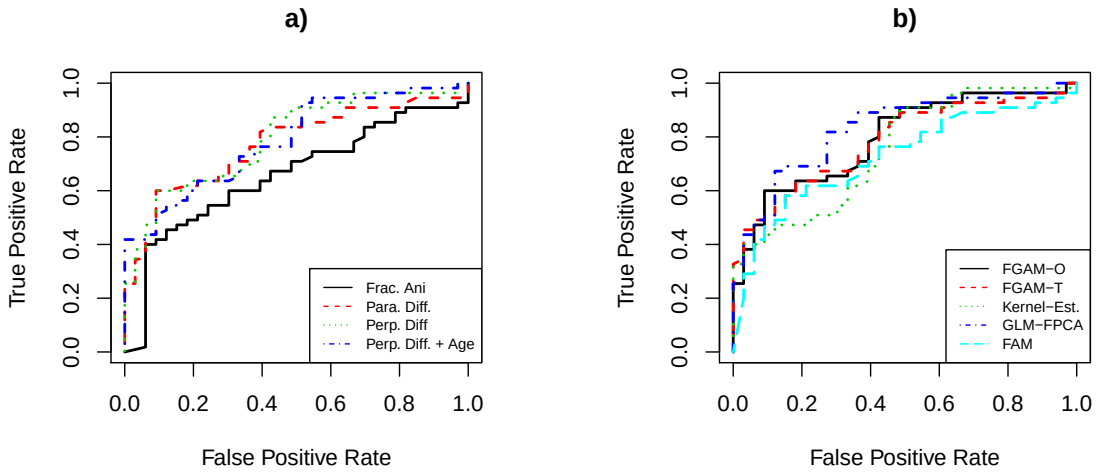

Figure 2.6: a) Leave-one-curve-out ROC curves for different FGAMs fit each using a different functional predictor and an FGAM including perpendicular diffusivity and a functional component for age. The response is MS status. b) Leave-one-curve-out ROC curves for both FGAM fits, and three other functional regression models when perpendicular diffusivity is the functional predictor.

We also compared the FGAM fits to three other generalized functional regression models. The first is the Ferraty and Vieu estimator (2.9) from the previous section. The use of this estimator for classification is discussed in detail in Ferraty and Vieu[29], Ch. 8. The second alternative model considered is a GLM in the functional principal component scores (GLM-FPCA) and the third model is a GAM in the functional principal component scores (FAM). The leave-one-curve-out ROC curves are displayed in the right plot of Figure 2.6 when perpendicular diffusivity is the covariate. There is little difference in performance between the models used.

CHAPTER 3

SPARSE COVARIATES

Often in FDA, the measurements one has for each curve can be subject to considerable error. In addition, many of the curves can be missing a large number of measurements. One might first try naively interpolating between points or applying a smoother individually to each curve to recover the underlying functions, followed by fitting the FGAM as if the functions were completely observed. However, as we will demonstrate, this approach quickly becomes inadequate as the amount of missingness in the data increases. A better approach, is to pool information across the functions and jointly estimate the functions while simultaneously fitting the FGAM. Developing algorithms for accomplishing this will be the topic of the chapter. Note that for the remainder of the dissertation, attention is restricted to the identity link-Gaussian error case.

Our goals for this chapter are three-fold: 1) accurate recovery of the sparsely observed trajectories, 2) accurate recovery of the surface, $F(x,t)$, and 3) accurate prediction of the response, Y. The missing parts of the trajectories must be imputed during the estimation procedure. Three possibilities for doing this are an expectation-maximization (EM) algorithm, Markov Chain Monte Carlo (MCMC), or a variational approximation. The advantage of MCMC over an EM algorithm approach is that uncertainty about the imputed curves is automatically taken into account during the estimation. Due to the computational overhead associated with MCMC, we also present a variational Bayes algorithm that can be used for fast approximate inference and to initialize an MCMC sampler.

Variational Bayes (VB) refers to a specific variational approximation used for Bayesian inference that relies on the assumption that a posterior density of interest

factors into a product form over certain groups of model parameters. Though they are commonly used in computer science, the application of variational approximations in statistics is relatively new; Ormerod and Wand[80] provides an overview. When the amount of posterior dependence is small, there is little loss of accuracy and often very large improvements in computation time over MCMC methods. Applications of VB to regression problems with missing data can be found in Faes et al.[22] and Goldsmith et al.[34], the latter of which considered the FLM.

The success of the approximation hinges on the amount of between-group dependence among the parameters in the posterior distribution. The cost of the computational efficiency gains from the approximations made in VB is the loss of guaranteed convergence to the correct distribution provided by MCMC. Factorization assumptions are often reasonable for certain groups of parameters in functional data models (Goldsmith et al.[34]). We agree with those authors that VB should not be considered a replacement for fully Bayesian inference. Instead we consider it as complementary to MCMC: a useful tool for approximate answers in large data situations when MCMC becomes intractable. One natural way to use the two as complements is to use VB estimates as starting values for an MCMC algorithm in the hopes of achieving faster convergence to, and better exploration of, the posterior distribution of interest. In our experience, the choice of starting values is critical for high-dimensional problems such as functional regression.

It is common to estimate the complete functional trajectories from the sparse data by performing a functional principal components analysis (FPCA); for example, the principal components analysis through conditional expectation (PACE) method of Yao et al.[131]. Whereas a typical functional data analysis smooths the measurements for each subject separately, the advantage of PACE is that it pools

data across subjects at each time point to estimate an entire covariance surface. This "borrowing of strength" across subjects is a main reason for the method's success. Although it is not considered in Yao et al.[131], one might think it reasonable to use a two-stage approach of first using PACE to recover the function predictors and then in a second step fitting an FLM using standard techniques or an FGAM using the procedure in Chapter 2. The main advantage of our Bayesian algorithms over a two-stage approach is that they allow us to directly account for uncertainty in the estimates from the FPCA. Our numerical results demonstrate the inadequacy of a conventional two-stage estimation procedure and we believe that our algorithms also gain from using information in the response when estimating the functional trajectories.

An important step in the PACE procedure is estimating the covariance surface of the functions using local polynomial modeling. Although PACE often performs well in a variety of situations, in our simulation studies we observe similar results to Peng and Paul[83], who found that PACE can have problems in more challenging settings with higher sparsity and a true covariance function that has more than three non-zero eigenvalues. In a number of the simulations in Peng and Paul[83], and in our own experiments, the covariance surface estimated by PACE is not positive definite and the estimated measurement error variance is negative. We will demonstrate that our Bayesian algorithms do not suffer from this problem. Our methods can also be used to effectively recover a greater number of principal components. Several currently available techniques only consider recovery of two non-zero principal components in simulation studies and attempt to estimate three components in real data studies (e.g., Yao and Lee[130], Yao et al.[131]).

When conjugate priors are used and closed-form expressions exist for all full conditional distributions in a model, the optimal densities for approximating the posterior using VB have closed-form expressions as well. It is not possible to obtain closed-form updates for all the parameters in the FGAM due to the nonconjugate full conditional distribution for the principal component scores, as they appear in the likelihood as arguments to the B-spline basis functions used to parameterize the regression surface. Therefore, Metropolis-Hasting steps are needed for our MCMC algorithm. For our VB algorithm, we alternatively overcome the nonconjugacy using a Laplace approximation. An additional complication is the necessity of an anisotropic roughness penalty for $F(x,t)$, owing to the possibly differing amounts of smoothness in x and t, which makes the two smoothing parameters difficult to separate. Using our VB approach, we are typically able to obtain a speed-up of at least an order of magnitude over generating 10,000 samples from our MCMC sampler, with minimal sacrifice in accuracy. Our approaches perform quite well at both out-of-sample prediction and recovering the true surface whether the true model is linear or nonlinear.

The remainder of the chapter proceeds as follows: Section 3.1 briefly reviews functional principal component analysis, Section 3.2 discusses our parametrization for the unknown surface, $F(x,t)$, Section 3.3 discusses our MCMC algorithm for fitting FGAM, Section 3.4 reviews variational Bayes and provides a VB algorithm for fitting FGAM, Section 3.5 discusses results of simulation experiments, and in Section 3.6 we apply our algorithms to forecasting closing prices for seven day auctions on the auction website eBay.

3.1 Recovering Sparsely Observed Functional Data

In this section we give a brief overview of the literature on estimating trajectories from sparsely observed functional data; one of our goals mentioned in the previous section and a key step in building our regression model. Most methods involve various techniques for estimating eigenfunctions and eigenvalues from an FPCA. A common approach for this is to use mixed model representations for penalized or smoothing splines; see James et al.[48] and the references therein. Another frequently used approach uses local polynomial modeling; see e.g., Yao et al.[131]. Bayesian approaches to functional data analysis include the wavelet-based mixed model method of Morris and Carroll[74] and the Dirichlet process based approach of Rodríguez et al.[97]. Though some papers in the Bayesian literature, including the ones cited above, appear to be able to deal with irregularly sampled functional data, it is unclear how these methods perform in the high-sparsity situations we wish to consider here, and we are not aware of any of these papers analyzing how their methods perform under varying degrees of sparsity/missingness.

The usual model for the unknown functions is to assume n_i noisy measurements have been taken of $X_i(t)$: $\widetilde{\mathbf{x}}_i = \{\widetilde{x}_i(t_{i,1}), \ldots, \widetilde{x}_i(t_{i,n_i})\}^T$ with $\widetilde{x}_i(t_{ij}) = X_i(t_{ij}) + e_{ij}$; $e_{ij} \overset{\text{i.i.d.}}{\sim} N(0, \sigma_x^2)$; $i = 1, \ldots, N$; $j = 1, \ldots, n_i$. We define the mean and covariance functions $\mu_x(t) := E\{X(t)\}$ and $G(s,t) := \text{Cov}\{X(s), X(t)\}$. If $X \in \mathcal{L}^2$, then by Mercer's theorem $G(s,t)$ admits an expansion $G(s,t) = \sum_{m=1}^{\infty} \nu_m \phi_m(s)\phi_m(t)$ with (orthonormal) eigenfunctions $\phi_m(\cdot)$ and associated eigenvalues ν_m, and the curves have a Karhunen-Loève representation $X_i(t) = \mu_x(t) + \sum_{m=1}^{\infty} \phi_m(t)\xi_{im}$; $\xi_{im} \overset{\text{ind.}}{\sim} (0, \nu_m)$, where the ξ's are known as principal component (PC) scores. If $X(t)$ is assumed to be a Gaussian process, then the principal component scores are Gaussian random variables.

For all FPCA methods, it is necessary to choose an integer, M, at which to truncate the basis expansion for the unknown functions (i.e. assume $\xi_k = 0$ for all $k > M$). This is typically done by including enough scores to explain a prespecified percentage (e.g. 99%) of the total observed variation in the data, and that is the approach we take in our analysis of the auction data in Section 3.6.

To initialize both our MCMC and VB algorithms, we take a similar (though not identical) approach to Yao et al.[131] and perform an FPCA as follows

1. Obtain an estimate $\hat{\mu}(t)$ of $\mu(t)$ via semiparametric regression of the pooled data $\tilde{\mathbf{x}} = (\tilde{\mathbf{x}}_1^T, \ldots, \tilde{\mathbf{x}}_N^T)^T$ on $\tilde{\mathbf{t}} = (\mathbf{t}_1^T, \ldots, \mathbf{t}_N^T)^T$; $\mathbf{t}_i = (t_{i,1}, \ldots, t_{i,n_i})^T$ using penalized splines.
2. Estimate $G(s,t)$ by fitting a cubic tensor product spline with third-derivative penalties (to shrink to a quadratic surface) to the "raw" covariances with the diagonal removed: $\{\widetilde{x}_i(t_{il}) - \hat{\mu}_x(t_{il})\}\{\widetilde{x}_i(t_{is}) - \hat{\mu}_x(t_{is})\},\ l \neq s$.
3. σ_x^2 is estimated as the average of the middle two thirds of the diagonal of the raw covariance matrix minus the diagonal of the smoothed covariance surface. This is as in Yao et al.[131] and is done to avoid boundary effects.
4. $\hat{\boldsymbol{\nu}} = (\hat{\nu}_1, \ldots, \hat{\nu}_M)^T$, and $\hat{\phi}_1(t), \ldots, \hat{\phi}_M(t)$ are obtained as the eigenvalues and eigenvectors, respectively, from an eigendecomposition of the estimated covariance matrix.
5. The principal component scores are the best linear unbiased prediction (BLUP) estimates: $\hat{\boldsymbol{\xi}}_i = \text{diag}(\hat{\boldsymbol{\nu}})\{\hat{\boldsymbol{\Phi}}(\mathbf{t}_i)\text{diag}(\hat{\boldsymbol{\nu}})\hat{\boldsymbol{\Phi}}(\mathbf{t}_i)^T + \hat{\sigma}_x^2\mathbb{I}_{N_i}\}^{-1}\hat{\boldsymbol{\Phi}}(\mathbf{t}_i)^T\{\widetilde{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_x(\mathbf{t}_i)\}$, $\hat{\boldsymbol{\xi}}_i = (\hat{\xi}_{i1}, \ldots, \hat{\xi}_{iM})^T$, and $\hat{\boldsymbol{\Phi}}(\mathbf{t}_i) = [\hat{\boldsymbol{\phi}}_1(\mathbf{t}_i) : \cdots : \hat{\boldsymbol{\phi}}_M(\mathbf{t}_i)]$, where $\hat{\boldsymbol{\phi}}_j(\mathbf{t}_i)$ and $\hat{\boldsymbol{\mu}}_x(\mathbf{t}_i)$ denotes the vector of evaluations of the jth estimated eigenfunction and estimated mean function, respectively, at the timepoints $\mathbf{t}_i,\ i = 1, \ldots, N$.

The parameters M, $\mu(t)$, $\nu_1, \ldots, \nu_M$, and $\phi_1(t), \ldots, \phi_M(t)$ are fixed at these initial estimates for our MCMC and VB algorithms. This is as done in Goldsmith et al.[34], though they do update $\nu_1, \ldots, \nu_M$. For ease of notation, we suppress the "hat"/circumflex for these parameters when developing our algorithms in later sections. The principal component scores as well as the measurement error variance are updated by both algorithms, and we will demonstrate that our methods can be used to accurately estimate more principal components beyond the first two. This procedure is also used in our numerical experiments when, for comparison, we also estimate FGAM using the two-step approach mentioned in the introduction.

3.2 A Mixed Model Formulation of FGAM

We next discuss our representation for the bivariate surface $F(\cdot, \cdot)$ and show how to formulate (1.2) as a mixed model. The mixed model formulation of penalized splines is now well-known and widely-used, see e.g., Ruppert et al.[99] for a review. The FGAM looks superficially like a bivariate smoothing problem, but it is more challenging since we do not observe $F(x, t)$ (with error) for pairs (x, t) but instead we observe only the integral of $F\{X(t), t\}$ with respect to t. Nonetheless, some ideas from bivariate smoothing are applicable to the FGAM. As in McLean et al.[70], we start with a bivariate spline model for $F(\cdot, \cdot)$ based on P-splines (Eilers and Marx[20], Marx and Eilers[68]). We take a more general approach than the Bayesian P-splines of Lang and Brezger[54], which performed isotropic smoothing via a first-order Gaussian random walk prior for the bivariate components in their additive model.

We must specify a grid of time points $\mathbf{t} = (t_1, \ldots, t_T)^T$ for approximating the

integral in (1.2) and we define $\mathbf{x}_i = \{x_i(t_1), \ldots, x_i(t_T)\}^T$ to be the ith estimated trajectory evaluated at $\mathbf{t}$. Our parameterization for the surface $F(x,t)$ follows.

$$E(Y_i|X_i) = \eta_{0i} + \int F\{x_i(t), t\}\, dt \approx \eta_{0i} + \sum_{j=1}^{K_x}\sum_{k=1}^{K_t} \int B_j^{\mathcal{X}}\{x_i(t)\} B_k^{\mathcal{T}}(t)\theta_{j,k}\, dt$$
$$\approx \eta_{0i} + \sum_{j=1}^{K_x}\sum_{k=1}^{K_t} \mathbf{L}^T(\mathbf{B}_{j,i}^{\mathcal{X}} \odot \mathbf{B}_k^{\mathcal{T}})\theta_{j,k} = \eta_{0i} + \sum_{j=1}^{K_x}\sum_{k=1}^{K_t} Z_{j,k,i}\theta_{j,k},$$

where $B^{\mathcal{X}}(\cdot)$ and $B^{\mathcal{T}}(\cdot)$ represent spline basis functions over the domains of $X(t)$ and t, with $\mathbf{B}_{j,i}^{\mathcal{X}} = [B_j^{\mathcal{X}}\{x_i(t_1)\}, \ldots, B_j^{\mathcal{X}}\{x_i(t_T)\}]^T$ and $\mathbf{B}_k^{\mathcal{T}} = \{B_k^{\mathcal{T}}(t_1), \ldots, B_k^{\mathcal{T}}(t_T)\}^T$ denoting vectors of these basis functions evaluated at the time points $\mathbf{t}$. $\mathbf{L} = (L_1, \ldots, L_T)^T$ is a vector of quadrature weights for the numerical integration. For ease of notation, we will write $\boldsymbol{\mu}_x(\mathbf{t})$ and $\boldsymbol{\Phi}(\mathbf{t})$ as $\boldsymbol{\mu}_x$ and $\boldsymbol{\Phi}$, respectively, and only specify the grid of evaluation points if it differs from $\mathbf{t}$. We also define the $T \times K_x K_t$ matrix

$$\mathbb{B}_{\xi_i} = \left[\{B_1^X(\boldsymbol{\mu}_x + \boldsymbol{\Phi}\boldsymbol{\xi}_i) \cdots B_{K_x}^X(\boldsymbol{\mu}_x + \boldsymbol{\Phi}\boldsymbol{\xi}_i)\} \otimes \mathbf{1}_{K_t}^T\right] \odot \left[\mathbf{1}_{K_x}^T \otimes \{B_1^T(\mathbf{t}) \cdots B_{K_t}^T(\mathbf{t})\}\right], \tag{3.1}$$

$i = 1, \ldots, N$. This matrix is always multiplied on the left by the vector of quadrature weights, $\mathbf{L}$, so we define $\mathbf{b}_{\boldsymbol{\xi}_i}^T \equiv \mathbf{L}^T\mathbb{B}_{\boldsymbol{\xi}_i}$. Note that $\mathbf{b}_{\boldsymbol{\xi}_i}^T = (Z_{1,1,i}, \ldots, Z_{K_x,K_t,i})^T$ is the ith row of the matrix $\mathbb{Z}$ from McLean et al.[70].

Owing to $X(t)$ and t having differing scales, it is not appropriate to assume a priori that the amount of smoothing for $F(x,t)$ should be the same in both arguments. Though we may scale x and t to lie in the unit square, this would still not result in a scale-invariant tensor product smooth (Wood et al.[129]). The necessitated anisotropic roughness penalty associated with the spline coefficients, $\boldsymbol{\theta} = (\theta_{11}, \ldots, \theta_{1,K_t}, \theta_{2,1} \ldots, \theta_{K_x,K_t})^T$, requires considerable more care than the univariate smoothing necessary for the Bayesian FLM in Goldsmith et al.[33], the isotropic penalty used in Müller et al.[77], or the penalized structured additive regression literature (e.g., Fahrmeir et al.[23]).

Wahba[117] first made the connection between spline smoothing and Bayesian modeling, showing that the usual (frequentist) estimator for a cubic smoothing spline was equivalent to placing a particular improper Gaussian prior on the spline coefficients. The penalization used in McLean et al.[70] is equivalent to imposing the following prior on the spline coefficients

$$p(\boldsymbol{\theta}|\lambda_x, \lambda_t) \propto \exp\left(-\frac{1}{2}\boldsymbol{\theta}^T \mathbb{P}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t)\boldsymbol{\theta}\right),$$

with $\mathbb{P}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t) = \lambda_x \mathbb{P}_x + \lambda_t \mathbb{P}_t$, with $\mathbb{P}_x = \mathbb{D}_x^T \mathbb{D}_x \otimes \mathbb{I}_{K_t}$, $\mathbb{P}_t = \mathbb{I}_{K_x} \otimes \mathbb{D}_t^T \mathbb{D}_t$. $\mathbb{I}_p$ is the identity matrix of dimension p, $\mathbb{D}_t$ and $\mathbb{D}_x$ are difference operator matrices of the prespecified degrees, d_x and d_t, respectively. This penalty structure leads to a partially improper Gaussian prior since $\mathbb{P}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t)$ is rank deficient: $\mathbb{D}_x^T \mathbb{D}_x$ has rank $K_x - d_x$, $\mathbb{D}_t^T \mathbb{D}_t$ has rank $K_t - d_t$, so that $\mathbb{P}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t)$ has rank $K_x K_t - d_x d_t$ (Horn and Johnson[43], Sec. 4.4). To avoid numerical instability associated with inversion of numerically rank-deficient matrices when sampling from the full conditional of $\boldsymbol{\theta}$ and the appearance of the zero determinant of $\mathbb{P}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t)$ in the full conditionals of λ_x and λ_t, we aim for a simpler representation of the function by employing the mixed model representation of tensor product splines used in Currie et al.[19], Sec. 6. The idea is to simultaneously diagonalize the marginal penalties for x and t. This results in a diagonal penalty structure which is efficient for computations and easy to interpret.

More precisely, we split the function $F(x, t)$ into an unpenalized part parameterizing functions from the nullspace of the penalty (i.e., associated with a diffuse Gaussian prior on the coefficients) and a penalized part (associated with a non-diffuse Gaussian prior on the coefficients). We begin by rewriting the vector of function evaluations for subject i as $F(\mathbf{x}_i, \mathbf{t}) = \sum_j^{K_x} \sum_k^{K_t} (\mathbf{B}_{j,i}^{\mathcal{X}} \odot \mathbf{B}_k^{\mathcal{T}}) \theta_{j,k} = \mathbb{B}_{\xi_i} \boldsymbol{\theta}$.

We take the spectral decompositions of the marginal penalties, i.e.,

$$\mathbb{D}_x^T\mathbb{D}_x = \mathbb{V}_x\mathbb{S}_x\mathbb{V}_x^T,\ \mathbb{D}_t^T\mathbb{D}_t = \mathbb{V}_t\mathbb{S}_t\mathbb{V}_t^T,$$

where both $\mathbb{V}_x$ and $\mathbb{V}_t$ are orthogonal matrices and $\mathbb{S}_x$ and $\mathbb{S}_t$ are diagonal. We define $\widetilde{\mathbb{V}}_x$ and $\widetilde{\mathbb{V}}_t$ to be the matrices of eigenvectors associated with zero eigenvalues, which have dimension $K_x \times d_x$ and $K_t \times d_t$, respectively. The basis functions for the unpenalized part of the tensor product spline can then be defined as $\mathbb{B}_{i,0} = \mathbb{B}_{\xi_i}(\widetilde{\mathbb{V}}_t \otimes \widetilde{\mathbb{V}}_x)$,

For the basis for the penalized part of the tensor product spline, $\mathbb{B}_{i,p}$, we first define $\mathbb{S}_{t,x} = (\mathbb{I}_{K_t}\otimes\mathbb{S}_x)+(\mathbb{S}_t\otimes\mathbb{I}_{K_x})$, a matrix that has all combinations of sums of the eigenvalues on the diagonal, and form $\widetilde{\mathbb{S}}_{t,x}$, which is $\mathbb{S}_{t,x}$ without the zero entries on the diagonal corresponding to $\mathbb{B}_{i,0}$. This can be written as $\widetilde{\mathbb{S}}_{t,x} = \mathbb{U}^T\mathbb{S}_{t,x}\mathbb{U}$, where $\mathbb{U}$ is a $K_xK_t \times (K_xK_t - d_xd_t)$ orthogonal matrix constructed by removing d_xd_t columns from $\mathbb{I}_{K_xK_t}$. We thus have

$$\mathbb{B}_{i,p} = \mathbb{B}_{\xi_i}(\mathbb{V}_t \otimes \mathbb{V}_x)\mathbb{U}\widetilde{\mathbb{S}}_{t,x}^{-1/2},$$

so that

$$\mathbb{B}_{\xi_i}\boldsymbol{\theta} = \mathbb{B}_{i,0}\boldsymbol{\beta} + \mathbb{B}_{i,p}\boldsymbol{\delta},$$

or, for clearer exposition,

$$\mathbb{B}_{\xi_i}\boldsymbol{\theta} = (\mathbb{B}_{\xi_i}\mathbb{T})(\mathbb{T}^{-1}\boldsymbol{\theta}) \text{ with } \mathbb{T} = [\mathbb{T}_0 : \mathbb{T}_p] = \left[\ (\widetilde{\mathbb{V}}_t \otimes \widetilde{\mathbb{V}}_x) : (\mathbb{V}_t \otimes \mathbb{V}_x)\mathbb{U}\widetilde{\mathbb{S}}_{t,x}^{-1/2}\ \right],$$

and $\mathbb{T}^{-1} = \left[\ (\widetilde{\mathbb{V}}_t \otimes \widetilde{\mathbb{V}}_x) : (\mathbb{V}_t \otimes \mathbb{V}_x)\mathbb{U}\widetilde{\mathbb{S}}_{t,x}^{1/2}\ \right]^T$.

The penalty matrix $\mathbb{P}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t)$ of the reparameterized coefficient vector $(\boldsymbol{\beta}^T, \boldsymbol{\delta}^T)^T = \mathbb{T}^{-1}\boldsymbol{\theta}$ becomes $\widetilde{\mathbb{P}}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t) = \mathbb{T}^T\mathbb{P}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t)\mathbb{T}$. Since $\mathbb{P}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t)\mathbb{T}_0 = 0$, only the lower right $(K_xK_t - d_xd_t) \times (K_xK_t - d_xd_t)$-quadrant of $\widetilde{\mathbb{P}}_{\boldsymbol{\theta}}(\lambda_x, \lambda_t)$ is of

interest. Denoting this submatrix by $\widetilde{\mathbb{P}}_{\boldsymbol{\delta}}(\lambda_x, \lambda_t)$, our penalty is now given by the diagonal matrix

$$\widetilde{\mathbb{P}}_{\boldsymbol{\delta}}(\lambda_x, \lambda_t) = \lambda_t \boldsymbol{\Psi}_t + \lambda_x \boldsymbol{\Psi}_x; \quad \boldsymbol{\Psi}_t = \widetilde{\mathbb{S}}_{t,x}^{-1/2} \mathbb{U}^T (\mathbb{S}_t \otimes \mathbb{I}_{K_x}) \mathbb{U} \widetilde{\mathbb{S}}_{t,x}^{-1/2}; \quad \boldsymbol{\Psi}_x = \mathbb{I}_{K_x K_t - d_x d_t} - \boldsymbol{\Psi}_t;$$

see Currie et al.[19].

Recalling that $\mathbf{b}_{\boldsymbol{\xi}_i}^T = \mathbf{L}^T \mathbb{B}_{\xi_i}$ with $\mathbb{B}_{\xi_i}$ given by (3.1), we can now write

$$\int F(X_i(t), t)\, dt \approx \mathbf{b}_{\boldsymbol{\xi}_i}^T \boldsymbol{\theta} = \mathbf{L}^T \mathbb{B}_{\xi_i} \mathbb{T} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\delta} \end{pmatrix} = \mathbf{L}^T \mathbb{B}_{\xi_i} \mathbb{T}_0 \boldsymbol{\beta} + \mathbf{L}^T \mathbb{B}_{\xi_i} \mathbb{T}_p \boldsymbol{\delta} = \mathbf{L}^T \mathbb{B}_{i,0} \boldsymbol{\beta} + \mathbf{L}^T \mathbb{B}_{i,p} \boldsymbol{\delta}.$$

We use diffuse inverse gamma (IG) priors for the variance components and our full model is given by

$$\begin{aligned}
Y_i &\sim N(\eta_{0i} + \mathbf{L}^T \mathbb{B}_{i,0} \boldsymbol{\beta} + \mathbf{L}^T \mathbb{B}_{i,p} \boldsymbol{\delta}, \sigma^2); \quad \sigma^2 \sim \mathrm{IG}(a_e, b_e); \\
\tilde{\mathbf{x}}_i(\mathbf{t}_i) &\sim N(\mu_x(\mathbf{t}_i) + \boldsymbol{\Phi}(\mathbf{t}_i) \boldsymbol{\xi}_i, \sigma_x^2 \mathbb{I}_{n_i}); \quad \sigma_x^2 \sim \mathrm{IG}(a_x, b_x); \\
\xi_{im} &\sim N(0, \nu_m); \quad m = 1, \ldots, M; \\
\boldsymbol{\delta} &\sim N\left(0, [\lambda_t \boldsymbol{\Psi}_t + \lambda_x \boldsymbol{\Psi}_x]^{-1}\right); \quad \lambda_x, \lambda_t \sim \mathrm{Gamma}(a_l, b_l); \\
\boldsymbol{\beta} &\sim N(0, \sigma_\beta^2 \mathbb{I}_{d_x d_t}); \quad \eta_{0i} \sim N(0, \sigma_\eta^2); \quad i = 1, \ldots, N
\end{aligned} \tag{3.2}$$

3.3 An MCMC algorithm for fitting FGAM

We now describe an MCMC algorithm for fitting FGAM. We will use a Metropolis-within-Gibbs sampler. The conjugate priors used for the spline coefficients and the variance components (excluding the smoothing parameters) in our hierarchical model allow for closed-form expressions for those parameters' full conditional distributions. Since their derivations are quite standard, we omit the details until Appendix A and focus in this section on the more complicated updates for the smoothing parameters and principal component scores.

To understand what is being updated and in what order, we start by providing pseudocode outlining the updates made by our MCMC algorithm to sample the posterior of model (3.2). Details of how the updates are done will be provided subsequently. This pseudocode also applies to our variational Bayes algorithm developed in the next section; the change being that instead of parameters being updated by randomly drawing from posterior distributions, they are deterministic updates of hyperparameters and moments of optimal densities. The pseudocode is given in Algorithm 1.

Algorithm 1 Pseudocode for fitting FGAM given by (3.2)

```
Obtain initial estimates, x, for the trajectories using the method from Section 3.1.
Specify penalties and bases for F(x,t). Obtain decomposition from Section 3.2.
Initialize other parameters.
repeat
    for i = 1 → N do
        Update principal component scores, ξ_i.
        Update x_i.
        Update B_{i,p}.
    end for
    for i = 1 → N do
        Update terms involving scalar covariates, η_{0i}.
    end for
    Update unpenalized spline coefficients, β.
    Update penalized spline coefficients, δ.
    Update smoothing parameters, λ_x, λ_t.
    Update measurement error variance, σ_x^2.
    Update response error variance, σ^2.
until Maximum number of iterations reached OR [for VB] convergence criteria met.
```

The updates for λ_x and λ_t require special attention because of the non-

conjugality of their full conditional distributions. To see this, we have

$$\begin{aligned} p(\lambda_x|\text{rest}) = p(\lambda_x|\lambda_t, \boldsymbol{\delta}) &\propto p(\boldsymbol{\delta}|\lambda_x, \lambda_t)p(\lambda_x) \propto |\lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t|^{1/2} (\lambda_x)^{a_l+1} \\ &\times \exp\{-(b_l + \frac{1}{2}\boldsymbol{\delta}^T \boldsymbol{\Psi}_x \boldsymbol{\delta})\lambda_x\} \propto |\lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t|^{1/2} \\ &\times \Gamma(\text{shape} = a_l + 2, \text{scale} = \{b_l + \frac{1}{2}\boldsymbol{\delta}^T \boldsymbol{\Psi}_x \boldsymbol{\delta}\}^{-1}) \equiv f_{\lambda_x}(\lambda_x), \end{aligned} \tag{3.3}$$

where "rest" is used to denote all parameters and data in the model besides λ_x. The derivation is analogous for λ_t. We do not obtain a closed-form expression for these full conditionals because of the determinant in (3.3). We overcome this difficulty by using slice sampling (Neal[78]). Slice sampling is a method for efficiently sampling from nonstandard distributions such as (3.3) by alternatingly sampling from the vertical region under $f_{\lambda_x}(x)$ and then sampling from the horizonal region under the density at the location of the vertical sample. Neal[78], Sec. 8 demonstrated that slice sampling can be more efficient than Metropolis methods for fitting Bayesian hierarchical models.

In our implementation, given an initial value, λ_0, and defining $g(x) := \log[f_{\lambda_x}(x)]$, we obtain a draw λ_1 from $p(\lambda_x|\text{rest})$ as follows

1. Draw $u \sim \text{Unif}\{0, g(\lambda_0)\}$ which defines a "slice" $S := \{x : u < g(x)\}$
2. Obtain an interval $[L, R]$ such that $S \subset [L, R]$ by starting with $[L_0, R_0] = [0, 2]$ and expanding the interval until $[L, R]$ contains S
3. Draw $\lambda_1 \sim \text{Unif}(L, R)$. If $\lambda_1 \notin S$, shrink $[L, R]$ and draw λ_1 again until $\lambda_1 \in S$,

and analogously for λ_t. For further details including proof of convergence to the proper posterior, see Neal[78]; his Fig. 1 is especially recommended for building intuition.

The second difficulty in developing our MCMC algorithm occurs when updating the principal component scores. This stems from the likelihood being a nonlinear function of the scores (they appear as arguments to B-spline basis functions). We have

$$
\begin{aligned}
p(\boldsymbol{\xi}_i|\text{rest}) &\propto p(y_i|\eta_{0i}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}_i, \sigma^2) p(\tilde{\mathbf{x}}_i|\boldsymbol{\xi}_i, \sigma_x^2) p(\boldsymbol{\xi}_i) \\
&\propto \exp\left\{ -\frac{1}{2\sigma^2} \left[y_{\eta_0,i} - \sum_j^{K_x} \sum_k^{K_t} \mathbf{L}^T \left\{ \mathbf{B}_j^{\mathcal{X}} \left(\boldsymbol{\mu}_x + \boldsymbol{\Phi} \boldsymbol{\xi}_i \right) \odot \mathbf{B}_k^{\mathcal{T}}(\mathbf{t}) \right\} \theta_{j,k} \right]^2 \right\} \cdot \\
&\quad \cdot \exp\left[-\frac{\{\widetilde{\mathbf{x}}_{\mu,i} - \boldsymbol{\Phi}(\mathbf{t}_i)\boldsymbol{\xi}_i\}^T \{\widetilde{\mathbf{x}}_{\mu,i} - \boldsymbol{\Phi}(\mathbf{t}_i)\boldsymbol{\xi}_i\}}{2\sigma_x^2} \right] \cdot \exp\left\{ -\frac{\boldsymbol{\xi}_i^T \operatorname{diag}(\boldsymbol{\nu}^{-1}) \boldsymbol{\xi}_i}{2} \right\}
\end{aligned}
$$

where $\widetilde{\mathbf{x}}_{\mu,i} = \tilde{\mathbf{x}}_i - \mu_x(\mathbf{t}_i)$ and $y_{\eta_0,i} = y_i - \eta_{0i}$, so that

$$
\begin{aligned}
p(\boldsymbol{\xi}_i|\text{rest}) &= \exp\left\{ -\frac{1}{2\sigma^2} \left[y_{\eta_0,i} - \sum_j^{K_x} \sum_k^{K_t} \theta_{j,k} \sum_t^{T} L_t B_j^{\mathcal{X}} \left\{ \mu_x(t_t) + \boldsymbol{\Phi}(t_t)^T \boldsymbol{\xi}_i \right\} B_k^{\mathcal{T}}\{t_t\} \right]^2 \right\} \cdot \\
&\quad \cdot N\left[\mathbf{m}_{\xi,i} = \mathbb{S}_{\boldsymbol{\xi},i} \boldsymbol{\Phi}(\mathbf{t}_i)^T \widetilde{\mathbf{x}}_{\mu,i},\ \mathbb{S}_{\boldsymbol{\xi},i} = \left\{ \boldsymbol{\Phi}(\mathbf{t}_i)^T \boldsymbol{\Phi}(\mathbf{t}_i)/\sigma_x^2 + \operatorname{diag}(\boldsymbol{\nu}^{-1}) \right\}^{-1} \right]
\end{aligned}
$$

We update each $\boldsymbol{\xi}_i,\ i = 1, \ldots, n$ based on its full conditional, with a proposal density for new values, $\boldsymbol{\xi}_i^\star$, based only on the trajectories and a Metropolis-Hastings (M-H) acceptance correction to account for the intractable part of the full conditional involving the likelihood of $\mathbf{y}$.

Specifically, the proposal distribution is

$$
q_1(\boldsymbol{\xi}_i, \boldsymbol{\xi}_i^\star) = N\left[\mathbf{m}_{\xi,i} = \mathbb{S}_{\xi,i} \boldsymbol{\Phi}(\mathbf{t}_i)^T \widetilde{\mathbf{x}}_{\mu,i},\ \mathbb{S}_{\xi,i} = \left\{ \boldsymbol{\Phi}(\mathbf{t}_i)^T \boldsymbol{\Phi}(\mathbf{t}_i)/\sigma_x^2 + \operatorname{diag}(\boldsymbol{\nu}^{-1}) \right\}^{-1} \right],
$$

so that $q_1(\boldsymbol{\xi}_i, \boldsymbol{\xi}_i^\star) = q_1(\boldsymbol{\xi}_i^\star)$ independent of the current state. The acceptance probability $\alpha(\boldsymbol{\xi}_i, \boldsymbol{\xi}_i^\star)$ is then the minimum of one and the following

$$
\frac{q_1(\boldsymbol{\xi}_i^\star, \boldsymbol{\xi}_i) p(\boldsymbol{\xi}_i^\star|\cdot)}{q_1(\boldsymbol{\xi}_i, \boldsymbol{\xi}_i^\star) p(\boldsymbol{\xi}_i|\cdot)} = \frac{\exp\left\{ -\frac{1}{2\sigma^2} \left[y_{\eta_0,i} - \sum_{j=1}^{K_x} \sum_{k=1}^{K_t} \mathbf{L}^T \left\{ \mathbf{B}_j^{\mathcal{X}} \left(\boldsymbol{\mu}_x + \boldsymbol{\Phi} \boldsymbol{\xi}_i^\star \right) \odot \mathbf{B}_k^{\mathcal{T}}(\mathbf{t}) \right\} \theta_{j,k} \right]^2 \right\}}{\exp\left\{ -\frac{1}{2\sigma^2} \left[y_{\eta_0,i} - \sum_{j=1}^{K_x} \sum_{k=1}^{K_t} \mathbf{L}^T \left\{ \mathbf{B}_j^{\mathcal{X}} \left(\boldsymbol{\mu}_x + \boldsymbol{\Phi} \boldsymbol{\xi}_i \right) \odot \mathbf{B}_k^{\mathcal{T}}(\mathbf{t}) \right\} \theta_{j,k} \right]^2 \right\}}
$$

because the ratio of proposal distributions cancels with the ratio of the tractable parts of the full conditionals.

As we will see in our numerical studies, the implausible trajectories that occasionally result from an FPCA occur much less frequently in our MCMC approach. This is because the proposals of extreme PC scores are likely to be rejected by our M-H step since they seem even more implausible when considered along with the response and current estimates of the regression coefficients in the acceptance probability.

The formula for the full model posterior can be found in Appendix A.

3.4 A Variational Bayes Approach

In this section we develop a variational Bayes algorithm for fitting the FGAM. We begin with a quick review of variational approximations.

3.4.1 Review of Variational Bayes

Our notation in this section closely follows that of Goldsmith et al.[34]. For an arbitrary density, $q(\theta)$, we define $\mu_{q(\theta)} \equiv E_q(\theta) = \int \theta_0 q_\theta(\theta_0)\, d\theta_0$ and $\sigma^2_{q(\theta)} \equiv \mathrm{Var}_q(\theta) = \int \{\theta_0 - E_q(\theta)\}^2 q_\theta(\theta_0)\, d\theta_0$ for scalar parameters, and analogously define $\mu_{q(\boldsymbol{\theta})}$ and $\Sigma_{q(\boldsymbol{\theta})}$ for vector parameters. We will give a brief overview of the main ideas of VB, and refer the reader to Bishop[7], Chapter 10 or Jaakkola and Jordan[45] for further details. Given observed data $\mathbf{y}$ and a collection of parameters $\boldsymbol{\theta}$, the goal of variational Bayes is to find a simplified density $q(\boldsymbol{\theta})$ that approximates the de-

sired posterior $p(\boldsymbol{\theta}|\mathbf{y})$ as closely as possible according to Kullback-Leibler (KL) divergence. The derivation of a variational Bayes algorithm relies on the result from Kullback and Leibler[53] that for an arbitrary density, $q(\boldsymbol{\theta})$, the marginal likelihood, $p(\mathbf{y})$, satisfies $p(\mathbf{y}) \geq \underline{p}(\mathbf{y}; q) := \exp\left[\int q(\boldsymbol{\theta}) \log\left\{p(\mathbf{y}; \boldsymbol{\theta})/q(\boldsymbol{\theta})\right\} d\boldsymbol{\theta}\right]$, with equality if and only if $q(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})$.

While other simplifications, for example that the density of interest, $q(\boldsymbol{\theta})$, is parametric, are sometimes used for variational approximations, variational Bayes uses the assumption that a posterior density can be factorized as $q(\boldsymbol{\theta}) = \prod_{p=1}^{P} q_p(\boldsymbol{\theta}_p)$ for some partition $\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_P\}$ of $\boldsymbol{\theta}$. Assuming this factorization for q and using the above result on KL divergence, it is easy to show (see e.g., Ormerod and Wand[80]) that $\underline{p}(\mathbf{y}; q)$ is maximized when q_p is chosen to be

$$q_p^*(\boldsymbol{\theta}_p) \propto \exp\left[E_{-\boldsymbol{\theta}_p}\{\log p(\mathbf{y}, \boldsymbol{\theta})\}\right] \propto \exp\left[E_{-\boldsymbol{\theta}_p}\{\log p(\boldsymbol{\theta}_p|\text{rest})\}\right]; \quad p = 1, \ldots, P; \tag{3.4}$$

where $E_{-\boldsymbol{\theta}_p}[\cdot]$ denotes expectation w.r.t. all model parameters excluding $\boldsymbol{\theta}_p$. We thus have a deterministic algorithm where one full iteration updates each component $\boldsymbol{\theta}_p$ sequentially using $q_p^*(\boldsymbol{\theta}_p)$. The algorithm terminates when the change in $\underline{p}(\mathbf{y}; q)$ becomes sufficiently small. Notice that the density, $p(\boldsymbol{\theta}_p|\text{rest})$, in (3.4) is precisely the full conditional from Gibbs sampling, and the optimal density is tractable when the full conditional is conjugate.

Helpful tools for deriving VB algorithms are directed acyclic graphs (DAGs) and Markov blankets. A Markov blanket is the set of all child, parent, and co-parent nodes of a particular node in a DAG. Examples can be found in Bishop[7], Ch. 8. Calculating the densities in (3.4) is made much simpler because of the result that $p(\boldsymbol{\theta}_p|\text{rest}) = p(\boldsymbol{\theta}_p|\text{Markov blanket of } \boldsymbol{\theta}_p)$.

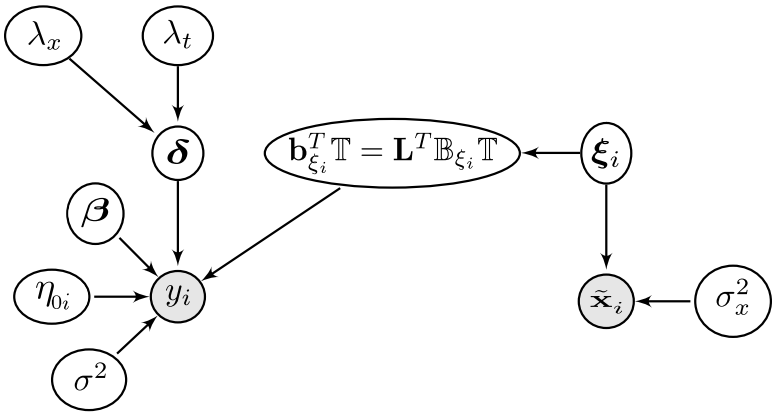

Figure 3.1: Directed Acyclic Graph for FGAM. Shaded vertices denote known quantities. The parameters $\{\nu_m\}$, $\{\phi_m\}$, M, and μ_x are omitted since they are not updated by the VB algorithm.

3.4.2 Fitting FGAM Using Variational Bayes

Our VB algorithm for fitting FGAM follows the same general steps used by our MCMC approach and given in Algorithm 1. As with MCMC, updates for the spline coefficients and variance components (smoothing parameters excluded) follow from standard calculations, so we leave them to A.2. The non-standard updates of the principal component scores and smoothing parameters are discussed below.

Using $\boldsymbol{\Theta}$ to denote all unknown parameters in our model (3.2), we assume the posterior $p(\boldsymbol{\Theta}|\mathbf{y}, \tilde{\mathbf{x}})$ admits the factorization $p(\boldsymbol{\Theta}|\mathbf{y}, \tilde{\mathbf{x}}) = q(\boldsymbol{\beta})q(\boldsymbol{\delta})q(\lambda_x)q(\lambda_t)q(\sigma^2)q(\sigma_x)\prod_{i=1}^N q(\boldsymbol{\xi}_i)q(\eta_{0i})$. The DAG for FGAM is shown in Figure 3.1.

For the optimal density for λ_x, we have from (3.4)

$$
\begin{aligned}
q^*(\lambda_x) &\propto \exp\left[E_{-\lambda_x}\{\log p(\lambda_x|\text{rest})\}\right] \\
&= \exp\left[E_{-\lambda_x}\left\{\frac{1}{2}\log|\lambda_x\boldsymbol{\Psi}_x + \lambda_t\boldsymbol{\Psi}_t| - \frac{1}{2}\boldsymbol{\delta}^T(\lambda_x\boldsymbol{\Psi}_x)\boldsymbol{\delta} + (a_l+1)\log(\lambda_x) - b_l\lambda_x\right\}\right] \\
&\approx \exp\left[\frac{1}{2}\log\left|\lambda_x\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right| - \frac{\lambda_x}{2}\left\{\text{tr}(\boldsymbol{\Psi}_x\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}) + \mu_{q(\boldsymbol{\delta})}{}^T\boldsymbol{\Psi}_x\mu_{q(\boldsymbol{\delta})}\right\}\right. \\
&\qquad \left. + (a_l+1)\log(\lambda_x) - b_l\lambda_x\right] = \left|\lambda_x\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right|^{1/2} \\
&\times \exp\left[-b_l\lambda_x - \frac{\lambda_x}{2}\left\{\text{tr}(\boldsymbol{\Psi}_x\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}) + \mu_{q(\boldsymbol{\delta})}{}^T\boldsymbol{\Psi}_x\mu_{q(\boldsymbol{\delta})}\right\}\right]\lambda_x^{a_l+1} \equiv \tilde{q}_{\lambda_x}(\lambda_x), \qquad (3.5)
\end{aligned}
$$

where the approximation comes from plugging in $\mu_{q(\lambda_t)}$ for λ_t to avoid taking an expectation of the determinant term over λ_t. Notice $c_{q(\lambda_x)} \equiv \int_0^\infty \tilde{q}_{\lambda_x}(x)\,dx$ has the form $c_{q(\lambda_x)} = \int_0^\infty x^{a_l+1}e^{-x}f(x)\,dx$ which can be approximated by generalized Gauss-Laguerre quadrature. This type of quadrature is implemented in `R` in the package `statmod` (Smyth et al.[109]), and we use it to determine a grid of G points, $\mathbf{g}$, and quadrature weights, $\mathbf{L}_g$. Our approximations are then given by $c_{q(\lambda_x)} \approx \mathbf{L}_g^T\widetilde{q}_{\lambda_x}(\mathbf{g})$ and $\mu_{q(\lambda_x)} \approx \{\mathbf{L}_g^T\widetilde{q}_{\lambda_x}(\mathbf{g})\}^{-1}\mathbf{L}_g^T\left\{\mathbf{g}\odot\widetilde{q}_{\lambda_x}(\mathbf{g})\right\}$.

Due to the exponential term in (3.5), moderate to large values of λ_x result in $\tilde{q}_{\lambda_x}(\lambda_x)$ being evaluated to be zero, unless care is taken during the computation to avoid underflow. One strategy for avoiding loss of precision is as follows. Define $\ell_{\lambda_x}(x) = \log\tilde{q}_{\lambda_x}(x)$ and $m_{\lambda_x} = \max_{\mathbf{g}}\ell_{\lambda_x}(\mathbf{g})$, then $c_{q(\lambda_x)} \approx \exp(m_{\lambda_x})\mathbf{L}_g^T\exp\{\ell_{\lambda_x}(\mathbf{g}) - m_{\lambda_x}\}$. The term $\exp(m_{\lambda_x})$ is in both the numerator and the denominator of $\mu_{q(\lambda_x)}$ and thus drops out in that calculation. Taking the logarithm of the determinant in $\tilde{q}_{\lambda_x}(\lambda_x)$ is not a problem because $\boldsymbol{\Psi}_x$ and $\boldsymbol{\Psi}_t$ are diagonal.

For updating the principal component scores in our VB algorithm, recall the

form of the full conditional

$$
\begin{aligned}
p(\boldsymbol{\xi}_i|\text{rest}) &\propto p(y_i|\eta_{0i},\boldsymbol{\beta},\boldsymbol{\delta},\boldsymbol{\xi}_i,\sigma^2)p(\tilde{\mathbf{x}}_i|\boldsymbol{\xi}_i,\sigma_x^2)p(\boldsymbol{\xi}_i) \\
&\propto \exp\left\{-\frac{1}{2\sigma^2}(y_i-\eta_{0i}-\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\theta})^2\right\}\exp\left\{-\frac{1}{2\sigma_x^2}||\tilde{\mathbf{x}}_i-\boldsymbol{\mu}_x(\mathbf{t}_i)-\boldsymbol{\Phi}(\mathbf{t}_i)\boldsymbol{\xi}_i||_2^2\right\} \\
&\times \exp\left\{-\frac{1}{2}\boldsymbol{\xi}_i^T\,\text{diag}(\boldsymbol{\nu}^{-1})\boldsymbol{\xi}_i\right\},
\end{aligned}
$$

where as before $\mathbf{b}_{\boldsymbol{\xi}_i}^T=\mathbf{L}^T\mathbb{B}_{\xi_i}$ with $\mathbb{B}_{\xi_i}$ given by (3.1). We have,

$$
\begin{aligned}
\text{E}_{-\boldsymbol{\xi}_i}\left\{-\frac{1}{2\sigma^2}(y_i-\eta_{0i}-\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\theta})^2\right\} &= -\frac{\mu_{q(1/\sigma^2)}}{2}\,\text{E}_{-\boldsymbol{\xi}_i}\left[\{y_i-\mu_{q(\eta_{0i})}-\text{E}_{-\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\theta})\}^2\right] \\
&-\frac{1}{2}\mu_{q(1/\sigma^2)}\sigma^2_{q(\eta_{0i})}-\frac{\mu_{q(1/\sigma^2)}}{2}\,\text{E}_{-\boldsymbol{\xi}_i}\left\{(\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\theta}-\text{E}_{-\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\theta}))^2\right\} \\
&= -\frac{\mu_{q(1/\sigma^2)}}{2}\left[(y_i-\mu_{q(\eta_{0i})}-\mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})})^2+\sigma^2_{q(\eta_{0i})}\right. \\
&\quad\left.+\,\text{E}_{-\boldsymbol{\xi}_i}\left\{(\boldsymbol{\theta}-\mu_{q(\boldsymbol{\theta})})^T\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\theta}-\mu_{q(\boldsymbol{\theta})})\right\}\right] \\
&= -\frac{\mu_{q(1/\sigma^2)}}{2}\left\{(y_i-\mu_{q(\eta_{0i})}-\mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})})^2+\sigma^2_{q(\eta_{0i})}+\text{tr}(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})})\right\}
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
q^*(\boldsymbol{\xi}_i) &\propto \exp\left[-\frac{\mu_{q(1/\sigma^2)}}{2}\left\{(y_i-\mu_{q(\eta_{0i})}-\mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})})^2+\sigma^2_{q(\eta_{0i})}+\text{tr}(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})})\right\}\right. \\
&\quad\left.-\frac{\mu_{q(1/\sigma_x^2)}}{2}||\tilde{\mathbf{x}}_i-\boldsymbol{\mu}_x(\mathbf{t}_i)-\boldsymbol{\Phi}(\mathbf{t}_i)\boldsymbol{\xi}_i||_2^2-\frac{1}{2}\boldsymbol{\xi}_i^T\,\text{diag}(\boldsymbol{\nu}^{-1})\boldsymbol{\xi}_i\right] \\
&\propto \exp\left[\mu_{q(1/\sigma^2)}\{y_i-\mu_{q(\eta_{0i})}\}\mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})}-\frac{\mu_{q(1/\sigma^2)}}{2}\left\{(\mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})})^2+\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i}\right\}\right. \\
&\quad+\mu_{q(1/\sigma_x^2)}\{\tilde{\mathbf{x}}_i-\boldsymbol{\mu}_x(\mathbf{t}_i)\}^T\boldsymbol{\Phi}(\mathbf{t}_i)\boldsymbol{\xi}_i \\
&\quad\left.-\,\frac{1}{2}\boldsymbol{\xi}_i^T\left\{\mu_{q(1/\sigma_x^2)}\boldsymbol{\Phi}^T(\mathbf{t}_i)\boldsymbol{\Phi}(\mathbf{t}_i)+\text{diag}(\boldsymbol{\nu}^{-1})\right\}\boldsymbol{\xi}_i\right] \equiv q(\boldsymbol{\xi}_i).
\end{aligned}
$$

Since this does not have the form of a standard, known density, we will employ a Laplace approximation. The use of Laplace approximations for variational inference with nonconjugate models was also explored in Wang and Blei[119]. This is given by

$$
q^*(\boldsymbol{\xi}_i)=N(\boldsymbol{\xi}_{i,0},\boldsymbol{\Lambda}_i^{-1}) \quad \text{where} \quad \boldsymbol{\Lambda}_i=-\mathcal{D}_{\boldsymbol{\xi}_i^T}\mathcal{D}_{\boldsymbol{\xi}_i}\log q(\boldsymbol{\xi}_i)\Big|_{\boldsymbol{\xi}_i=\boldsymbol{\xi}_{i,0}}, \tag{3.6}
$$

with $\mathcal{D}_{\mathbf{a}}[\cdot]$ denoting differentiation w.r.t. the vector $\mathbf{a}$ and $\boldsymbol{\xi}_{i,0}$ denoting the mode of $q^*(\boldsymbol{\xi}_i)$, which is found by a numerical optimization routine. The formula for $\boldsymbol{\Lambda}_i$ is given in A.2. We expect the Laplace approximation to perform well in high sparsity settings because the Gaussian prior becomes the dominant part of the posterior in these situations.

To construct our algorithm, we also require the expectation of $\mathbf{b}_{\boldsymbol{\xi}_i}$ and the expectation of its outer product with respect to $\boldsymbol{\xi}_i$. To do this we use second-order Taylor expansions about $\boldsymbol{\xi}_{i,0}$. These derivations are also left to Appendix A.2. Our log-likelihood lower bound, which is used for monitoring convergence of our algorithm, is derived in Appendix A.3 and the full variational Bayes algorithm is given in Appendix A.4 as Algorithm 2.

3.5 Simulation Study

We now conduct a simulation study to compare the efficacy of our proposed approaches. We fit each model to 100 simulated data sets. The true functional covariates are given by $X(t) = \sum_{j=1}^{4} \xi_j \phi_j(t)$, with $\xi_j \sim N(0, 8j^{-2})$ and $\{\phi_1(t), \ldots, \phi_4(t)\} = \{\sin(\pi t/|\mathcal{T}|), \cos(\pi t/|\mathcal{T}|), \sin(2\pi t/|\mathcal{T}|),$
$\cos(2\pi t/|\mathcal{T}|)\}$, with $|\mathcal{T}|$ denoting the measure of the interval $\mathcal{T}$. To examine how our model performs with both sparse and dense but irregularly observed data, we generate observed covariates by randomly selecting $J_i = 10$ or $J_i = 40$ points for each subject from a grid of 50 equally-space points used to generate the true response. We consider three different levels of the measurement error variance, $\sigma_x^2 = 0,\ 1,$ and 4. The response error variance is taken to be $\sigma^2 = 1$. We examine two different possibilities for the regression surface $F(x, t)$. First, a case where the

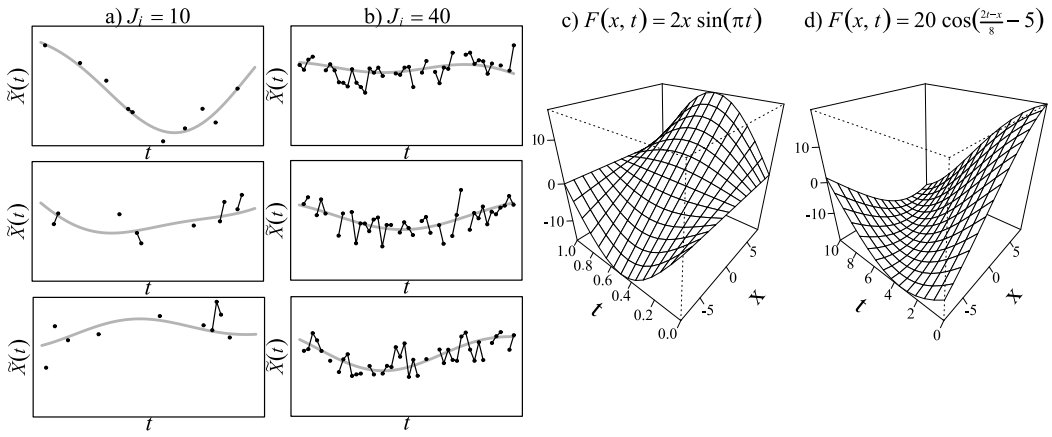

Figure 3.2: Plots a) and b) show three observed functional predictors for varying levels of sparsity when $\sigma_x = 1$. The true trajectories are also plotted in grey. Plot c) shows the surface $F(x,t) = 2x\sin(\pi t)$ and plot d) the surface $F(x,t) = 20\cos\left(-\frac{x}{8} + \frac{t}{4} - 5\right)$.

FLM is the true model, $F(x,t) = 2x\sin(\pi t)$, with $\mathcal{T} = [0,1]$; and next, a case where the FLM does not hold, $F(x,t) = 20\cos\left(-\frac{x}{8} + \frac{t}{4} - 5\right)$, with $\mathcal{T} = [0,10]$. A sampling of some generated curves including measurement error for both levels of sparsity as well as plots of both true surfaces can be found in Figure 3.2.

For our comparison we consider seven different methods for fitting FLMs and FGAMs: 1) a baseline/oracle FGAM fit by the Chapter 2 approach when the fully observed curves without measurement error are known (trueX), 2) FGAM fit as in Chapter 2 with fixed trajectories estimated using the procedure outlined in Section 3.1 (PACE), 3) FGAM fit using variational Bayes on the sparse, noisy curves (VB), 4) FGAM fit using MCMC and the sparse, noisy curves (MCMC), 5) as in 4) except initial values are supplied by the VB fit (VB-MCMC), 6) FLM fit using penalized splines with trajectories obtained from the Section 3.1 procedure (FLM-PACE), and 7) FLM fit to the fully observed curves without measurement error (FLMtrueX). Each method used cubic B-splines and second-order difference penalties. The Chapter 2 implementation of FGAM is again fit using the code available in the package `refund` (Crainiceanu et al.[16]) in `R` (R Core Team[88]).

Smoothing parameters are chosen by generalized cross validation (GCV) using the package `mgcv` (Wood[128]), which is also used to estimate the FLMs. MCMC runs one chain for 10,000 iterations after a burn-in of 1000, whereas VB-MCMC uses only 1000 iterations after a burn-in of 500. Each method uses and, if applicable, estimates exactly the true number of non-zero components $M = 4$. For each simulated data set, we use two thirds of the 100 observations to fit the models and the other one third for prediction.

We first compare how well PACE, VB, MCMC, and VB-MCMC do at estimating the functional covariates. The median over simulations of the in-sample root mean integrated square error, RMISE-X^2 $= \frac{1}{N}\sum_{i=1}^{67}\int_{\mathcal{T}}\left\{X_i(t) - \hat{X}_i(t)\right\}^2 dt$, for each scenario and method is reported in Figure 3.3 a). We see that the PACE method does not perform well in the sparse data scenarios ($J_i = 10$). One reason for this is that it does not account for the variability from imputing the principal component scores. An additional reason is difficulties in estimating a covariance matrix for the functional predictors. The estimate is often singular or near-singular and this causes numerical problems when attempting to estimate all four non-zero principal component scores using the method presented in Section 3.1. Our Bayesian algorithms do not suffer from this problem even when starting from poorly conditioned initial estimates from our PACE implementation. We see that VB performs quite well at recovering the trajectories, even in the $J_i = 10$ scenarios. MCMC performs slightly worse than VB here. Further investigation showed that MCMC on average slightly overestimated σ_x^2 which made it less accurate for in-sample recovery, but that this added variance made for more accurate prediction of trajectories out-of-sample.

Now turning to estimation of the true surface $F(x, t)$, we report the median

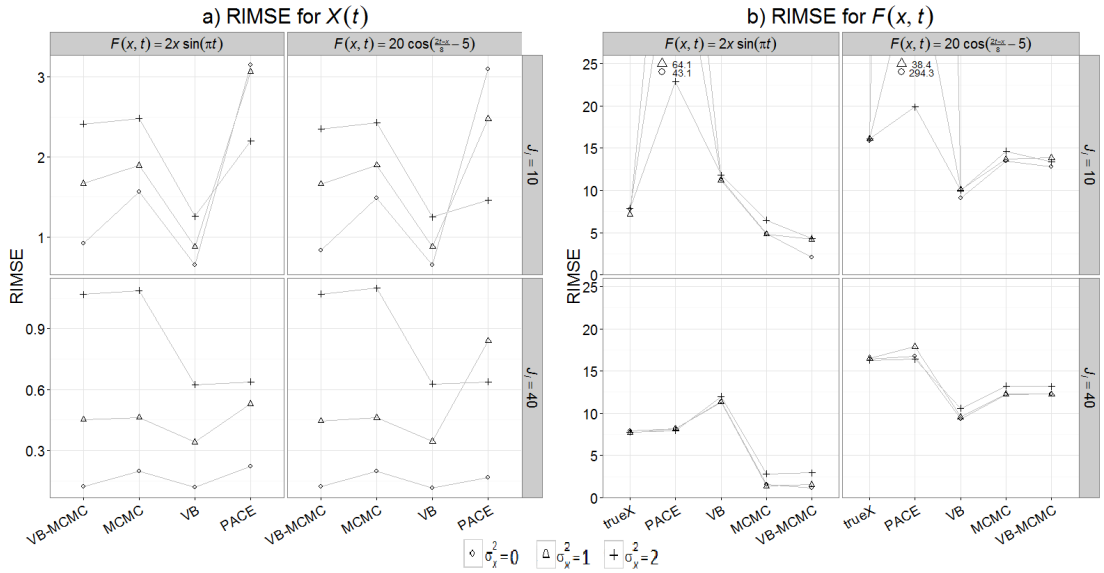

Figure 3.3: a) Median RMISE over 100 simulations for two levels of sparsity and different values for the measurement error variance for recovering in-sample trajectories, $X(t)$. b) Median RISE for predicting the true surface, $F(x,t)$. b) Includes trueX which is not relevant for a). Values that do not fall within the y-axis limits are individually labeled.

root integrated square error, RISE-F^2 $= \int_{\mathcal{X}} \int_{\mathcal{T}} \left\{F(x,t) - \hat{F}(x,t)\right\}^2 dt\, dx$, in Figure 3.3 b). We evaluate the RISE only at (x,t) values that are inside the convex hull defined by the observed trajectories for that sample to avoid regions of the plane where there are no data. We again observe performance from the PACE method to be poor in the sparse settings. Interestingly, the MCMC and MCMC-VB approaches have lower ISE than the trueX method. We suspect this is due to the MCMC algorithm on average choosing larger smoothing parameters which are closer to the optimal values for smoothing the surface than those chosen by GCV for the trueX fits. Due to the additional smoothing performed by the integration in (1.2), the optimal amount of smoothing for estimating the response and for estimating the surface are different (Cai and Hall[9]). Also noteworthy is the substantial difference between VB and MCMC depending on the true regression surface. This again seems to be due to differences in how the smoothing parameters are chosen.

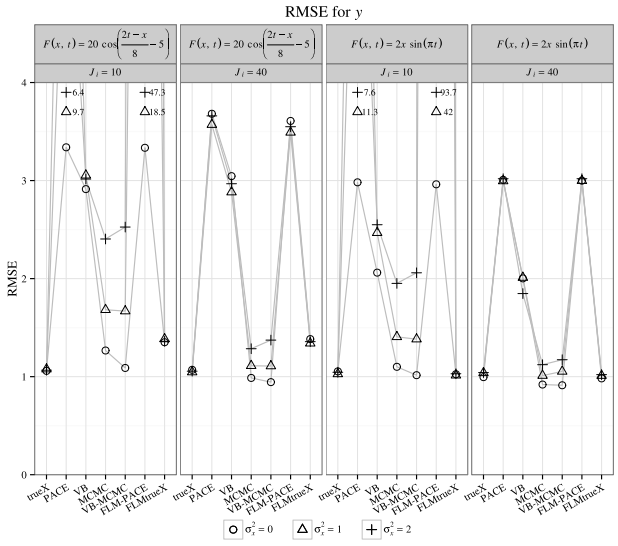

Figure 3.4: Median RMSE over 100 simulations for out-of-sample predictions of the response, Y, for two levels of sparsity and different values for the measurement error variance. Values that do not fall within the y-axis limits are individually labeled.

Finally, results for root mean square error (RMSE) for predicting the out-of-sample response, RMSE-$Y^2 = \frac{1}{33}\sum_{i=68}^{100}(Y_i - \hat{Y}_i)^2$, can be found in Figure 3.4. We see that the performance of MCMC matches and even sometimes outperforms the oracle trueX method that knows the entire trajectories. Overall, we recommend the combination of VB for initial estimates followed by MCMC as it appears to be best or close to best in nearly all scenarios. The total elapsed time for estimating FGAM on one data set averaged over all simulations and scenarios was 43.3 seconds for VB, 732.0 seconds for MCMC, and 153.5 seconds for VB-MCMC.

3.6 Analysis of Auction Data

In this section we fit our proposed models to auction data from the online auction website eBay and attempt to forecast closing auction price. The data set contains the time and amount of every bid for 155 seven-day auctions of Palm M515 Personal

Digital Assistants (PDA) that took place between March and May, 2003. Each auction is "standardized" to start at time 0. This data was previously analyzed using functional data methods in a series of work by W. Jank, G. Shmueli and coauthors (e.g., Jank and Shmueli[49], Wang et al.[121]). The PACE methodology introduced in Section 3.1 was used to analyze this data set in Liu and Müller[63]. Typically, each auction consists of three clearly discernible parts: an initial period with some bidding, a middle period with very few bids, and a final period of rapid bidding as the auction finishes (Wang et al.[121]). This sparsity and irregularity in the observed bid data means that the usual methods of function data analysis are not appropriate.

Our raw data is actually the maximum amount the bidder is willing to pay for the item, often called the willing-to-pay (WTP) value. To recover the current item price from the WTP values, we must use the table available at `http://pages.ebay.com/help/buy/bid-increments.html`. When a new WTP value is entered that is more than any previous WTP value, the new price is determined by incrementing the current price in an amount given by this table. A new bidder must enter an amount at least as large as this new price plus the increment given by the table. We assume there is an underlying smooth price process that we attempt to recover with our proposed approaches.

We use the logarithm of the ratio of successive prices during the first six days of the auction to predict the logarithm of the closing price on the final day. Hourly prices are used so that we are trying to recover $6 \times 24 = 144$ prices for each auction. When an auction has multiple bids in the same hour, we take the average of the prices corresponding to those bids as the observed price for that hour. As in Liu and Müller[63], we set any negative values for the log-price ratio equal to zero,

which can occur because initial log-price at time 0 is taken to be zero. To show the usefulness of our MCMC and VB methods, we fit the FGAM and FLM using the trajectory of observed log-price ratios, $\log\{\tilde{x}_i(t_{i,j})/\tilde{x}_i(t_{i,j-1})\}$, for the first six days in order to predict the logarithm of the final selling price at the end of the seventh day. We emphasize that no information on the prices from the final day of the auction are included in the functional predictor so that we have a true measure of forecasting accuracy.

We randomly partition the data into training and test sets with two thirds of the samples used for training and one third for testing. We compute the root mean square error (RMSE) for predicting the logarithm of the closing price for the test data set after fitting each model to the training data. This is repeated for 25 different splits into test and training sets. For comparison, we also considered the simple two-step approach of using PACE to recover the functional predictors and then using these estimates to fit FLMs and FGAMs in `refund` as in the fully-observed predictor case from McLean et al.[70]. For the FGAM methods, ten basis functions were used for both axes.

The surface estimated by our MCMC algorithm fit to the entire data set is displayed in Figure 3.5 b) along with the observed and estimated log-price ratios for five randomly chosen auctions. Figure 3.5 a) plots all estimated trajectories and additionally histograms showing the frequencies of observations for both $X(t)$ and t; notice from the histogram on the right part of the plot that the majority of the data is grouped at very low log-price ratios. In b) we see that large values of the log-price ratio in the early hours of the auction result in a lower predicted value for the closing price and that smaller ratios later towards the end of the sixth day of the auction result in higher predicted closing price. Nonlinearities in

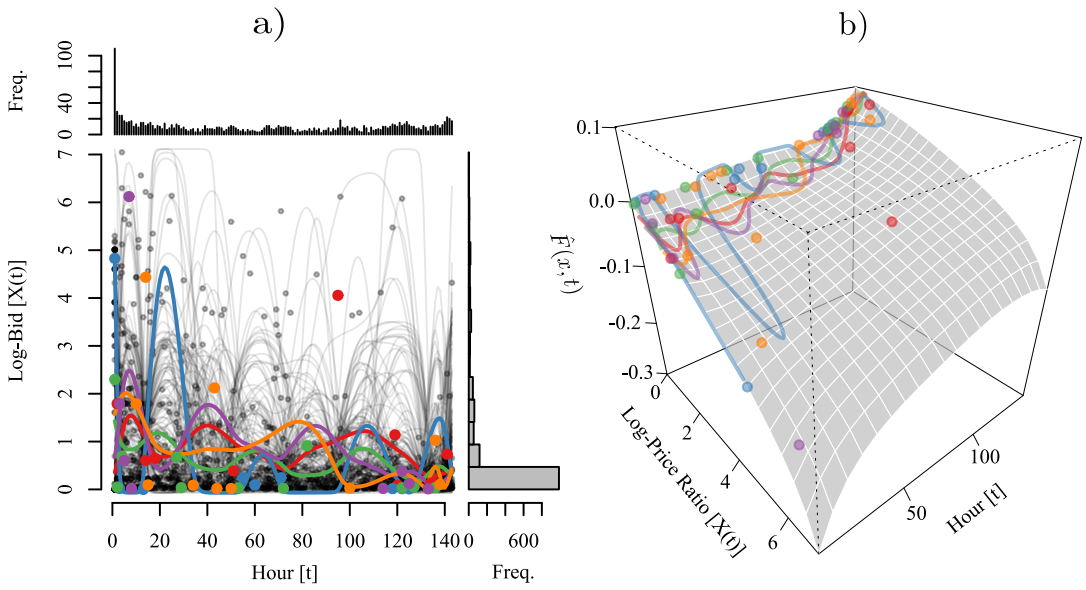

Figure 3.5: a) All estimated trajectories from use of our MCMC algorithm on the auction data with points representing observed data. Five trajectories are highlighted and also plotted in b). Also included are two histograms showing which covariate values occur with the highest frequency (on the right) and the frequency of bids for each hour of the auction (on top). b) Shows the estimated surface $\hat{F}(x,t)$ from fitting FGAM to the auction data using MCMC. The overlayed points and curves are the same as a).

the log-price component of the estimated surface suggest that an FLM may not be flexible enough for this data set. There appears to be some undersmoothing of the functional predictors in Figure 3.5 a). Cai and Hall[9] showed that for optimal prediction in the FLM, the coefficient function should be undersmoothed because of the additional smoothing performed by the integral in the regression function. We conjecture that some degree of undersmoothing of the functional predictors is desirable for our forecasting problem when estimating (3.2) for similar reasons.

The median out-of-sample RMSE over 25 partitions of the data is reported in Table 3.1 along with standard deviations. We can see that our Bayesian approach for fitting FGAM offers the best performance in this case, with both FGAM-MCMC and FGAM-VB offering much improved performance over the methods that only use PACE followed by estimation of FGAM in `refund`. Both methods

FLM-PACE	FGAM-PACE	FGAM-MCMC	FGAM-VB
0.5917(1.3093)	4.913(0.4322)	0.0914(0.0052)	0.0905(0.0037)

Table 3.1: Median RMSE (with standard deviation in parentheses) for out of sample predictions of log-final selling price for 25 random splits of the auction data

that simply used PACE and then assumed fully observed data had very poor performance for some of the splits when the imputed trajectories were especially bad.

CHAPTER 4

TESTS FOR LINEARITY

Our main goal in this chapter will be to test $H_0: \; E(Y_i|X_i) = \theta_0 + \int_{\mathcal{T}} \beta(t)X_i(t)\,dt$ (FLM) vs. $H_1: \; E(Y_i|X_i) = \theta_0 + \int_{\mathcal{T}} F(X_i(t), t)\,dt$ (FGAM). As was noted in Chapter 2, $\frac{\partial^2}{\partial x^2}F(x,t) \equiv 0$ implies H_0, $F(x,t) = \beta(t)x$. Using a second-order difference penalty, this corresponds to an infinite amount of smoothing in the x-direction. It will be shown how this corresponds to having a zero variance component in a linear mixed model. The key idea is to reparameterize the model, partitioning into a parametric (unpenalized) term and a smooth, nonparametric term[s] subject to a slightly different penalty than was considered in Chapter 3. In this chapter, variance components in mixed models will explicitly take the role of smoothing parameters in the standard nonparametric model. The idea of using mixed models in this way was popularized by the monograph of Ruppert et al.[99] and has been the subject of much research since its publication. In this chapter, we focus on the parameterization owing to Wood et al.[129], which has a number of useful properties, as will be demonstrated shortly. We will explore likelihood ratio tests (LRTs), restricted likelihood ratio tests (RLRTs), and Bayes factor approaches for our testing problem.

We begin by reviewing restricted maximum likelihood estimation in Section 4.1, followed by a review of LRTs and RLRTs for zero variance components in linear mixed models in Section 4.2, and wrap up our review of background material with a discussion of Bayes factors in Section 4.3. We introduce the Wood et al.[129] construction in Section 4.4. In Section 4.5 we demonstrate how the construction can be used to test for linearity of FGAM as well as no effect of the functional covariate. In Section 4.6 by extending the work of Maruyama and George[66] and

Rouder et al.[98], we show how generalized g-priors for the penalized coefficients (random effects) can be used to obtain simple expressions for Bayes factors for our hypotheses of interest. Section 4.7 provides a simulation study of our proposed approaches, and Section 4.8 concludes with an application of our methods to some motor vehicle emissions data.

4.1 Restricted Maximum Likelihood Estimation

Consider a linear mixed model of the form

$$\mathbf{Y} = \mathbb{X}\boldsymbol{\beta} + \sum_{j=1}^{J} \mathbb{Z}_j \mathbf{b}_j + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N(0, \sigma^2 \mathbb{I}_N), \quad \mathbf{b}_j \sim N(0, \sigma_j^2 \mathbb{I}_{q_j}), \tag{4.1}$$

where $\dim(\boldsymbol{\beta}) = q_0$ and $\dim(\mathbf{b}_j) = q_j$, and it is assumed that $\mathbf{b}_j \perp\!\!\!\perp \mathbf{b}_k;\ j \neq k$ and $\boldsymbol{\epsilon} \perp\!\!\!\perp \mathbf{b}_j;\ j = 1, \ldots, J$. Typically, $\boldsymbol{\beta}$ are called the fixed effects and the $\mathbf{b}_j$'s are known as random effects. Though the random effects often have a more general covariance structure, we do not need to consider such a case in this dissertation. We have $E(\mathbf{Y}) = \mathbb{X}\boldsymbol{\beta}$ and $\text{cov}(\mathbf{Y}) = \boldsymbol{\Sigma}_J := \sigma^2 \mathbb{I}_N + \sum_{j=1}^{J} \sigma_j^2 \mathbb{Z}_j \mathbb{Z}_j^T$. Frequently, one is interested in testing that a variance component, say σ_j^2 is equal to zero, which implies the random effect, $\mathbf{b}_j$, has no effect on predicting the response and can be removed from the model. It is easily seen that the log-likelihood for this model is

$$\ell(\boldsymbol{\beta}, \sigma^2, \sigma_1^2, \ldots, \sigma_J^2) = -\frac{N}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}_J| - \frac{(\mathbf{Y} - \mathbb{X}\boldsymbol{\beta})^T \boldsymbol{\Sigma}_J^{-1} (\mathbf{Y} - \mathbb{X}\boldsymbol{\beta})}{2\sigma^2}.$$

Maximum likelihood estimates for the error variance and other components of variance in this model are known to be biased because they do not account for lost degrees of freedom from estimating $\boldsymbol{\beta}$. For this reason, a popular alternative to maximum likelihood is restricted maximum likelihood (Patterson and Thompson[81]). The restricted likelihood can be obtained by integrating the usual likeli-

hood w.r.t. $\boldsymbol{\beta}$. The logarithm of the restricted likelihood is given by

$$\ell_R(\sigma^2, \sigma_1^2, \ldots, \sigma_J^2) = \ell(\boldsymbol{\beta}, \sigma^2, \sigma_1^2, \ldots, \sigma_J^2) - \frac{1}{2}|\mathbb{X}^T \boldsymbol{\Sigma}_J^{-1} \mathbb{X}|.$$

Restricted maximum likelihood (REML) and maximum likelihood are the main alternatives to methods that minimize prediction error, such as GCV or Akaike's information criteria (AIC) and its siblings, for estimating penalized spline models. There is some evidence to suggest that it should be preferred to GCV because it avoids GCVs occasional tendencies to badly undersmooth and offers slightly better RMSE performance in practise (Reiss and Ogden[95]).

When the variance components and error variance are known, the best linear unbiased estimates (BLUPs) for $\boldsymbol{\beta}$ and the random effects $\mathbf{b}_j$, can be shown to be given by the minimizer of the penalized least squares objective function

$$\frac{1}{2\sigma^2}||\mathbf{Y} - \mathbb{X}\boldsymbol{\beta} - \mathbb{Z}\mathbf{b}||^2 + \mathbf{b}^T \mathbb{V}_J^{-1} \mathbf{b},$$

where $\mathbb{Z} = [\mathbb{Z}_1 : \mathbb{Z}_2 : \cdots : \mathbb{Z}_J]$, $\mathbb{V}_J = \text{diag}(\sigma_1^2 \mathbf{1}_{q_1}, \cdots, \sigma_J^2 \mathbf{1}_{q_J})$, and $\mathbf{b} = (\mathbf{b}_1^T, \cdots, \mathbf{b}_J^T)^T$. The solution is given by

$$(\hat{\boldsymbol{\beta}}^T, \hat{\mathbf{b}}^T)^T = (\mathbb{B}^T \boldsymbol{\Sigma}_J^{-1} \mathbb{B} + \mathbb{D})^{-1} \mathbb{B}^T \mathbf{Y},$$

where $\mathbb{B} = [\mathbb{X} : \mathbb{Z}]$ and $\mathbb{D} = \text{bdiag}(\mathbf{0}_{q_0}, \sigma^2 \mathbb{V}_J^{-1})$. Note the similarities with the estimates from Chapter 2 if we define $\lambda_j = \sigma^2/\sigma_j^2$. The variance components must be estimated using some numerical optimization routine such as the EM algorithm or Newton's method (see, e.g., Pinheiro and Bates[84]). Estimation in `R` is handled by the package `lme` and `lme4` (Bates et al.[4]). Penalized spline models can be fit via REML using the package `mgcv` (Wood[128]).

4.2 LRTs and RLRTs for Linear Mixed Models

Restricting attention to the one variance component case ($J = 1$), consider testing the hypothesis $H_0 : \sigma_1 = 0$ vs. $H_1 : \sigma_1 > 0$, using a likelihood ratio test

$$\text{LRT} = 2 \sup_{H_1} \ell(\boldsymbol{\beta}, \sigma^2, \sigma_1^2) - 2 \sup_{H_0} \ell(\boldsymbol{\beta}, \sigma^2, \sigma_1^2),$$

or a restricted likelihood ratio test

$$\text{RLRT} = 2 \sup_{H_1} \ell_R(\boldsymbol{\beta}, \sigma^2, \sigma_1^2) - 2 \sup_{H_0} \ell_R(\boldsymbol{\beta}, \sigma^2, \sigma_1^2).$$

Wilk's theorem/approximation states that under some mild regularity conditions, a LRT statistic converges in distribution to a chi-squared random variable under H_0. However, because σ_1 is on the boundary of the parameter space under the null hypothesis, the conditions for Wilk's theorem do not hold. Self and Liang[104] were able to show for some simple mixed models with one variance component, that the LRT above actually converges to a mixture distribution $0.5\chi_0^2 + 0.5\chi_1^2$, where χ_d^2 denotes a chi-squared random variable with d degrees of freedom. Further developments for longitudinal mixed models were made by Stram and Lee[112]. However, both those papers require that the response variables, y_i, be separable into independent clusters.

This requirement is not satisfied by model (4.1). Crainiceanu and Ruppert[17] found that the mixture-χ^2 approximation is too conservative to be used for penalized spline models. Crainiceanu and Ruppert[17] were able to derive the exact finite-sample distributions for both the RLRT and LRT for models with one variance component. Since we will only consider RLRTs in this chapter, we provide the distribution of RLRT only:

$$RLRT = \sup_{\lambda} \left[(N - q_0 - 1) \log\{1 + U_n(\lambda)\} - \sum_{k=1}^{q_1} \log(1 + \lambda\mu_k) \right]; \qquad (4.2)$$

where $\lambda = \sigma_1^2/\sigma^2$, with $U(\lambda) = N(\lambda)/D(\lambda)$ for

$$N(\lambda) = \sum_{k=1}^{q_1} \frac{\lambda\mu_k}{1+\lambda\mu_k} w_k^2, \qquad D(\lambda) = \sum_{k=1}^{q_1} \log(1+\lambda\mu_k) + \sum_{k=q_1+1}^{N-q_0} w_k^2;$$

with $w_k \overset{\text{i.i.d.}}{\sim} N(0,1);\ k = 1, \ldots, N-q_0-1$ and μ_k being the eigenvalues of the matrix $\mathbb{Z}^T(\mathbb{I}_N - \mathbb{X}(\mathbb{X}^T\mathbb{X})\mathbb{X}^T)\mathbb{Z}$. While this may look awful, it is actually quite easy to simulate from. The eigendecompostion of the $q_1 \times q_1$ matrix need only be computed once, and then all that is required to obtain a draw from the RLRT distribution is simulation of q_1 χ_1^2 random variables plus one $\chi^2_{N-q_0-q_1-1}$ random variable.

While the theory is fully developed for the one variance component case, extensions to tests for models with multiple variance components (which we will be needing for FGAM) have proven much harder and this is still an open problem. An approach that has proven to work well empirically is that of Greven et al.[37], which used ideas from pseudo-likelihood estimation and relied on the assumption that the restricted likelihood ratio tests (RLRT) for their variance components of interest could be accurately approximated by an RLRT that assumes the nuisance random effects are known. Another possible approach has been proposed by Wang and Chen[123], which developed F-tests for penalized spline models estimated in the mixed model framework. We choose to work with the approach of Greven et al.[37] because it has been shown through extensive simulation studies by Scheipl et al.[102] to work well and because the method is available in an `R` package by the same authors. The simulations in Wang and Chen[123] also confirmed the effectiveness of the Greven et al.[37] approach, with their F-tests only offering minor improvements in the case where a nuisance variance component is very close to zero. As will be seen shortly, this would seem to be an unlikely occurrence for the test we will develop for FGAM because the nuisance effect will correspond to the

effect of the FLM. We refer to the Greven et al.[37] method as the pseudo-RLRT.

A further complication not addressed by the above papers is how to test for multiple variance components being simultaneously zero under the null hypothesis. This is the situation that we will be faced with for FGAM and the subsequent sections will explore some ideas for how to deal with this problem. One additional important fact to be aware of is that it is a requirement for RLRTs (but not LRTs) that the fixed effects have the same structure under both the null and alternative hypotheses. The tests we consider in this chapter always have the same fixed effects structure under both hypotheses.

4.3 Review of Bayes Factors

Bayes factors were introduced in Jeffreys[50] and are the most popular approach for hypothesis testing and model selection in Bayesian statistics. They provide a measure of the evidence in the observed data in favour of one hypothesis/model over another. Given two models $\mathcal{M}_0$ and $\mathcal{M}_1$, the Bayes factor, B_{10} is the ratio of the posterior odds of $\mathcal{M}_1$ to the prior odds, where the odds for a given probability are defined by odds=probability/(1 − probability). Denoting the observed data by $\mathcal{D}$, the posterior probability of model $\mathcal{M}_1$ is $p(\mathcal{M}_1|\mathcal{D}) = \frac{p(\mathcal{D}|\mathcal{M}_1)p(\mathcal{M}_1)}{p(\mathcal{D})}$, and similarly for $\mathcal{M}_0$. Using this result the Bayes factor is given by

$$B_{10} := \frac{p(\mathcal{M}_1|\mathcal{D})}{1-p(\mathcal{M}_1|\mathcal{D})}\frac{1-p(\mathcal{M}_1)}{p(\mathcal{M}_1)} = \frac{p(\mathcal{M}_1|\mathcal{D})}{p(\mathcal{M}_0|\mathcal{D})}\frac{p(\mathcal{M}_0)}{p(\mathcal{M}_1)} = \frac{p(\mathcal{D}|\mathcal{M}_1)}{p(\mathcal{D}|\mathcal{M}_0)}.$$

Computation of $p(\mathcal{D}|\mathcal{M})$ involves integrating over all model parameters. For a model parameterized by the vector $\boldsymbol{\theta}$, we have $p(\mathcal{D}|\mathcal{M}) = \int p(\mathcal{D}|\boldsymbol{\theta}, \mathcal{M})\pi(\boldsymbol{\theta}|\mathcal{M})\, d\boldsymbol{\theta}$. These integrals are typically intractable and one must resort to approximations for their computation. Though many different Bayes factor approximations have been

B_{10}	Evidence against $\mathcal{M}_0$
1 to 3	Not worth more than a bare mention
3 to 20	Postive
20 to 150	Strong
>150	Very strong

Table 4.1: Rules from Kass and Raftery[51] for interpreting Bayes factors.

proposed, the most common are Laplace approximations and MCMC techniques such as bridge sampling (Meng and Wong[71]). This is normally what is done in the Bayesian linear mixed model and generalized linear mixed model setting; see Pauler et al.[82], Sinharay and Stern[108], or Saville and Herring[101] for details. An advantage of Bayes factors over likelihood ratio testing is that there is no requirement for the models to be nested (though we will only consider nested models in this dissertation) and unlike p-values, Bayes factors can be used to provide evidence in favour of the null hypothesis. There is also evidence that Bayes factors naturally protect against overfitting (Kass and Raftery[51]). Whenever Bayes factors are used, it is important to check for sensitivity of the Bayes factor to the prior specification used. Table 4.1 reproduces a table in Kass and Raftery[51] that provides guidelines for interpreting Bayes factor values.

4.4 More Mixed Models for Penalized Splines

In this section we discuss how to represent penalized splines smooths as linear mixed models with the goal of expressing the FGAM (1.2) in the form (4.1). Before going through the reparameterization that will receive most of the attention in this chapter, we introduce a simple first approach using a more common mixed model representation.

4.4.1 A Simple First Approach

The following representation which can be found in Section 6.2.2 of Wood[126]. Using notation from Chapter 2, our penalized likelihood is

$$(\mathbf{y} - \mathbb{Z}\boldsymbol{\theta})^T(\mathbf{y} - \mathbb{Z}\boldsymbol{\theta}) + \lambda_x \boldsymbol{\theta}^T \mathbb{P}_x^T \mathbb{P}_x \boldsymbol{\theta} + \lambda_t \boldsymbol{\theta}^T \mathbb{P}_t^T \mathbb{P}_t \boldsymbol{\theta}$$

where $\mathbb{P}_x = \mathbb{D}_x \otimes \mathbb{I}_{K_t}$ and $\mathbb{P}_t = \mathbb{I}_{K_x} \otimes \mathbb{D}_t$ with $\mathbb{D}_x$ and $\mathbb{D}_t$ being second order differencing matrices on adjacent B-spline coefficients for the x-axis and t-axis, respectively. First, we obtain the eigendecomposition of the penalty, $\mathbb{P}_x^T\mathbb{P}_x + \mathbb{P}_t^T\mathbb{P}_t = \mathbb{U}\boldsymbol{\Lambda}\mathbb{U}^T$, where $\mathbb{U}$ is an orthogonal matrix of eigenvectors and $\boldsymbol{\Lambda}$ is the diagonal matrix of eigenvalues. Write $\mathbb{U} = [\mathbb{U}_r : \mathbb{U}_f]$, where $\mathbb{U}_r$ corresponds to the non-zero eigenvalues and $\mathbb{U}_f$ to the four zero-eigenvalues corresponding to the null space of the penalty. Now re-parameterize to $(\mathbf{b}_r^T, \boldsymbol{\beta}^T)^T = \mathbb{U}^T\boldsymbol{\theta}$, and form $\mathbb{X}_f = \mathbb{Z}\mathbb{U}_f$, $\mathbb{X}_r = \mathbb{Z}\mathbb{U}_r$ and $\widetilde{\mathbb{P}} = \mathbb{U}_r^T(\lambda_x\mathbb{P}_x^T\mathbb{P}_x + \lambda_t\mathbb{P}_t^T\mathbb{P}_t)\mathbb{U}_r$. We now have a linear mixed model

$$\mathbf{y} = \mathbb{X}_f\boldsymbol{\beta} + \mathbb{X}_r\mathbf{b} + \boldsymbol{\epsilon}, \qquad \mathbf{b} \sim N(0, \widetilde{\mathbb{P}}^{-1}), \qquad \boldsymbol{\epsilon} \sim N(0, \sigma_e^2\mathbb{I}).$$

This mixed model can be fit in `R` using the package `mgcv` (Wood[128]); however, it is not in a very helpful form for our testing problem because the covariance of the vector of random effects depends on two smoothing parameters. Thus, more work is required as we cannot use this formulation for our desired test.

Recall that the null hypothesis is equivalent to $H_0 : \lambda_x = \infty$ with λ_t as a nuisance parameter. Arguing along the same lines as Greven et al.[37], a reasonable approach would be to fix the variance component for λ_t at its REML estimate, form pseudo residuals and conduct our test in their framework discussed in Section 4.2. The approach is outlined below. Form the augmented response vector $\widetilde{\mathbf{y}} = (\mathbf{y}^T, \mathbf{0}^T)^T$ and the augmented design matrix $\widetilde{\mathbb{Z}} = [\mathbb{Z}^T : \lambda_t^{1/2}\mathbb{P}_t^T]^T$, with λ_t

assumed known. Recalling (2.5), our penalized pseudo-likelihood is

$$-\frac{1}{2}(\widetilde{\mathbf{y}}-\widetilde{\mathbb{Z}}\boldsymbol{\theta})^T(\widetilde{\mathbf{y}}-\widetilde{\mathbb{Z}}\boldsymbol{\theta})+\frac{\lambda_x}{2}\boldsymbol{\theta}^T\mathbb{P}_x^T\mathbb{P}_x\boldsymbol{\theta}.$$

First, obtain the eigendecomposition of the penalty for x, $\mathbb{P}_x^T\mathbb{P}_x=\mathbb{U}\boldsymbol{\Lambda}\mathbb{U}^T$, with $\boldsymbol{\Lambda}$ containing four zeros on the diagonal corresponding to the null space of the tensor product penalty with second-order marginal penalties. Re-parameterize so that the new coefficient vector is $(\mathbf{b}_r^T,\boldsymbol{\beta}^T)^T=\mathbb{U}^T\boldsymbol{\theta}$ and form $\mathbb{X}_f=\widetilde{\mathbb{Z}}\mathbb{U}_f$ and $\mathbb{X}_r=\widetilde{\mathbb{Z}}\mathbb{U}_r$. We now have the linear mixed model

$$\widetilde{\mathbf{y}}=\mathbb{X}_f\boldsymbol{\beta}+\mathbb{X}_r\mathbf{b}_r+\widetilde{\boldsymbol{\epsilon}},\qquad \mathbf{b}_r\sim N(0,\lambda_x^{-1}\boldsymbol{\Lambda}_*^{-1}),\qquad \widetilde{\boldsymbol{\epsilon}}\sim N(0,\sigma_e^2\mathbb{I}),$$

where $\boldsymbol{\Lambda}_*$ is the diagonal matrix of non-zero eigenvalues of the penalty for x. Re-parameterizing again to $\mathbf{b}=\boldsymbol{\Lambda}_*^{1/2}\mathbf{b}_r$ and $\widetilde{\mathbb{Z}}=\mathbb{X}_r\boldsymbol{\Lambda}_*^{1/2}$, and we now get a one variance component linear mixed model

$$\widetilde{\mathbf{y}}=\mathbb{X}_f\boldsymbol{\beta}+\widetilde{\mathbb{Z}}\mathbf{b}+\widetilde{\boldsymbol{\epsilon}},\qquad \mathbf{b}\sim N(0,\lambda_x^{-1}\mathbb{I}),\qquad \widetilde{\boldsymbol{\epsilon}}\sim N(0,\sigma_e^2\mathbb{I}).$$

Note that while the data augmentation idea we have presented is a standard trick to efficiently compute the solution of a penalized least squares problem by recasting it as an unpenalized least squares problem, it has not before been suggested to simplify tests of variance components/smoothing parameters in penalized spline models. Therefore, this approach is of interest in general for penalized spline models and not simply for our FGAM testing problem.

4.4.2 (Low Rank) Penalized Spline ANOVA Models

In this section to allow for maximum generality, we work with a nonparametric model of the form

$$Y_i=\beta_0+\sum_{j=1}^{p}L_{ij}\{f_j(x_{ij})\}+\sum\sum_{k<l}L_{ikl}\{f_{kl}(x_{ik},x_{il})\}+\cdots+\epsilon_i,$$

where $\epsilon_i \sim N(0, \sigma^2)$; $i = 1, \ldots, N$ and the L's are known linear functionals. In the usual additive model, $L_{ij}\{f_j(x_{ij})\} = f_j(x_{ij})$. Using this notation, for the FLM and FGAM we have

$$Y_i = \beta_0 + L_{i1}\{f_1(t)\} + \epsilon_i = \beta_0 + \int f_1(t) X_{i1}(t)\, dt + \epsilon_i$$

and

$$Y_i = \beta_0 + L_{i11}[f_{11}\{X_{i1}(t), t\}] + \epsilon_i = \beta_0 + \int f_{11}\{X_i(t), t\}\, dt + \epsilon_i,$$

respectively, where f_1 is the coefficient function $\beta(t)$ and f_{11} is F from previous chapters. Of course the above integrals will need to be approximated as in the previous chapters. The model could also contain both fixed effects and random effects of scalar covariates, but we omit them here for simplicity.

We now review the tensor product spline basis construction of Wood et al.[129] which parallels smoothing spline ANOVA (for e.g., Gu[39], Wang[122]); the main difference being the use of low-rank spline bases. Each marginal smooth term, f_j, is represented using K_j B-splines and is associated with a quadratic penalty matrix $\mathbb{P}_j$ of order d_j and a smoothing parameter g_j which controls the smoothness of f_j. In matrix notation we have $\mathbf{f}_j := \mathbb{B}_j \theta_j$, where $\mathbb{B}_j$ is a $N \times K_j$ matrix of B-spline basis function evaluations and θ_j is the K_j-vector of unknown spline coefficients. The most common choices in practise are cubic B-splines and a second-order penalty. In this case, $\mathbb{P}_j$ is rank $K_j - 2$ and contains integrated products of second-derivatives of the B-splines

$$\{P_j\}_{m,n} = \int_{\mathcal{X}_j} B_m^{\prime\prime(j)}(x) B_n^{\prime\prime(j)}(x), \quad m, n = 1, \ldots, K_j,$$

where $\mathcal{X}_j$ is the range of the spline basis. The full model will be projected onto a tensor sum of orthogonal subspaces. Each component in the new construction is either unpenalized or has its own unique penalty that is interpretable in terms of

the original model component functions. This is convenient because it will lead to a mixed model representation where each random effect has a diagonal covariance matrix independent of the other effects. Despite this simple penalty structure, we will not simply shrink all coefficients as in a simple ridge regression, but instead have multiple penalties interpretable in terms of function shape.

This penalty structure greatly simplifies computations in both frequentist and Bayesian settings and is a feature not shared by other tensor product constructions, such as Belitz and Lang[6], Lee and Durbán[56], Lee et al.[57], or Wood[126]. Typically in a Bayesian setup, an improper prior is used for the spline coefficients of each smooth term $\pi(\boldsymbol{\theta}_j) \propto \exp(-\lambda_j \boldsymbol{\theta}_j^T \mathbb{P}_j \boldsymbol{\theta}_j)$. Improper priors can be problematic when trying to define Bayes factors. Additionally, tensor-product smooth terms involving two or more covariates would have priors involving multiple smoothing parameters.

For our construction, we begin as in Wood et al.[129] by performing an eigendecomposition of each marginal penalty $\mathbb{P}_j = \mathbb{U}_j \mathbb{D}_j \mathbb{U}_j^T;\ j = 1, \ldots, J$, where $\mathbb{U}_j$ is orthogonal and $\mathbb{D}_j$ is diagonal with d_j zeros on the diagonal, $\mathbb{D}_j = \text{diag}(d_{j1}, d_{j2}, \ldots, d_{jq_j}, 0, \ldots, 0);\ d_{j1} \geq \cdots \geq d_{jq_j} > 0$, where $q_j = K_j - d_j$. Letting $\mathbb{U}_{n,j}$ be the columns of $\mathbb{U}_j$ corresponding to the zero eigenvalues and $\mathbb{U}_{p,j}$ be the remaining columns, we define $(\mathbf{b}_j^T, \boldsymbol{\beta}_j^T)^T := \mathbb{U}_j^T \boldsymbol{\theta}_j$, $\mathbb{X}_{n,j} := \mathbb{B}_j \mathbb{U}_{n,j}$, $\mathbb{X}_{p,j} := \mathbb{B}_j \mathbb{U}_{p,j}$, and $\mathbb{D}_{+,j}$ to be the largest submatrix of $\mathbb{D}_j$ with no zeros on the diagonal. For the one covariate case, $y_i = f(x_i) + e_i$, we arrive at the mixed model

$$\mathbf{y} = \mathbb{X}_n \boldsymbol{\beta} + \mathbb{X}_p \mathbf{b} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N_N(\mathbf{0}, \sigma^2 \mathbb{I}_N), \quad \mathbf{b} \sim N_{q_j}(\mathbf{0}, g \mathbb{D}_+^{-1}), \tag{4.3}$$

with $g := \frac{1}{\lambda}$. By forming $\tilde{\mathbb{X}}_p := \mathbb{B} \mathbb{U}_p \mathbb{D}_+^{-1/2}$ and $\tilde{\mathbf{b}} := \mathbb{D}_+^{-1/2} \mathbb{U}_p^T \boldsymbol{\theta}$, (4.3) can be further simplified to have random effect covariance matrix $g \mathbb{I}_{q_j}$.

To construct tensor product interactions we use the box product, also known

as the row-wise Kronecker product, which is defined for two matrices $\mathbb{A}_1$ and $\mathbb{A}_2$ of dimension $n \times m_1$ and $n \times m_2$, respectively as $\mathbb{A}_1 \square \mathbb{A}_2 := \mathbb{A}_1 \otimes \mathbf{1}_{m_1}^T \odot \mathbf{1}_{m_2}^T \otimes \mathbb{A}_2$, where $\otimes$ represents the Kronecker product and $\odot$ denotes element-wise matrix multiplication. Now given two marginal smooths decomposed as described in the preceding paragraph, we can form the model matrix, $\mathbb{M}$, for the tensor product smooth in one of two ways

1. By forming $\mathbb{M} = [\mathbb{X}_{n,1} \square \mathbb{X}_{n,2} : \mathbb{X}_{p,1} \square \mathbb{X}_{n,2} : \mathbb{X}_{n,1} \square \mathbb{X}_{p,2} : \mathbb{X}_{p,1} \square \mathbb{X}_{p,2}]$, where the first box product contains the basis for the null space of the smooth and the three remaining terms are the bases for three i.i.d. normal random effects.
2. As in 1), but instead forming the box products involving $\mathbb{X}_{n,1}$ and $\mathbb{X}_{n,2}$ using each individual column of those matrices separately. Letting $\mathbf{x}_{n,j}^{(k)}$ denote the kth column of $\mathbb{X}_{n,j}$, when $d_1 = d_2 = 2$ this becomes $\mathbb{M} = [\mathbb{X}_{n,1} \square \mathbb{X}_{n,2} : \mathbf{x}_{n,1}^{(1)} \square \mathbb{X}_{p,2} : \mathbf{x}_{n,1}^{(2)} \square \mathbb{X}_{p,2} : \mathbb{X}_{p,1} \square \mathbf{x}_{n,2}^{(1)} : \mathbb{X}_{p,1} \square \mathbf{x}_{n,2}^{(2)} : \mathbb{X}_{p,1} \square \mathbb{X}_{p,2}]$.

Construction 2) increases the number of variance components/smoothing parameters from three to five, but has the advantage of being fully scaling invariant. This means that inferences about the smooth terms are not affected by arbitrarily rescaling the covariates. See Wood et al.[129] for in depth discussion of this fact. We called the term $\mathbb{X}_{p,1} \square \mathbb{X}_{p,2}$ a range space-range space interaction and the terms $\mathbb{X}_{p,1} \square \mathbb{X}_{n,2}$ and $\mathbb{X}_{n,1} \square \mathbb{X}_{p,2}$ null space-range space interactions.

A third covariate can be added by constructing $\mathbb{X}_{n,3}$ and $\mathbb{X}_{p,3}$ as described above for the new covariate and then forming box products of $\mathbb{X}_{n,3}$ (or the columns of $\mathbb{X}_{n,3}$ for construction 2.) with all the terms in $\mathbb{M}$ and also all box products of $\mathbb{X}_{p,3}$ with all the terms in $\mathbb{M}$ above. Additional covariates can be added in an analogous manner. Higher order interactions can be ignored by dropping terms

involving higher numbers of $\mathbb{X}_p$ matrices. For example, the third-order interaction in a three covariate smooth can be ignored by dropping $\mathbb{X}_{p,1}\square\mathbb{X}_{p,2}\square\mathbb{X}_{p,3}$ from $\mathbb{M}$. For interpretability, it is important to ensure that the constant function is always part of the null space basis. If a null space that explicitly includes the constant function is not obvious, further reparameterization of the null space to achieve this is possible (Wood et al.[129]). After this reparameterization, if we let $\mathbb{X}$ denote the one component of $\mathbb{M}$ that involves only box products of the null space terms, $\mathbb{X}_{n,j}$'s, and use $\mathbb{Z}_j$'s to denote all other components (i.e. ones involving at least one $\mathbb{X}_{p,j}$), than we arrive at a mixed model in the standard form (4.1). The specific case of FGAM will be demonstrated in the next section to make this more clear.

4.5 A Test for Linearity of FGAM

We now discuss how to implement the construction of the previous section for FGAM. First reviewing notation, $\mathbb{B}_x$ will denote the $NJ \times K_x$ matrix of the x-axis B-splines evaluated at vec($\mathbf{X}$) where $\mathbf{X}$ is the $N \times J$ matrix of observed functional predictor values (N curves measured J times each). As should become more apparent shortly, we must use $d_x \geq 2$ here in order to have the FLM nested in FGAM. Choices of $d_x > 2$ are both uncommon in practise and result in addition variance components needing to be estimated if the fully invariant construction is used. Choosing $d_t = 1$ is possible, and perhaps worth exploring, but was not done in this chapter in favour the more common choice $d_x = d_t = 2$.

Obtain the eigendecomposition of the marginal penalty for the x-axis, $\mathbb{D}_x^T\mathbb{D}_x = \mathbb{U}_x\mathbf{\Lambda}_x\mathbb{U}_x^T = [\mathbb{U}_{x,r} : \mathbb{U}_{x,f}]\mathbf{\Lambda}_x[\mathbb{U}_{x,r} : \mathbb{U}_{x,f}]^T$, with $\mathbb{U}_{x,f}$ containing the two eigenvectors corresponding to the zero eigenvalues and $\mathbb{U}_{x,r}$ containing the others. Form $\mathbf{\Lambda}_{x,+}$,

the matrix $\mathbf{\Lambda}_x$ with the zero entries on the diagonal replaced with ones and then form $\mathbb{B}_x^* = \mathbb{B}_x \mathbb{U}_x \mathbf{\Lambda}_{x,+}^{-1}$. The first $K_x - 2$ columns of $\mathbb{B}_x^*$, say $\mathbb{Z}_x$, form a basis for the random effects of the marginal smooth (i.e. a basis for the range space of the marginal penalty) and the last two columns, say $\mathbb{X}_x$ form a basis for the fixed effects of the smooth. The marginal penalty matrix will become the identity matrix of appropriate dimension except with its last two diagonal entries equal to zero; denote this as $\mathbb{I}_-$. For the $N \times J$ matrix $\mathbf{T}$ of observations times, form $\mathbb{B}_t$, the matrix of t-axis B-spline evaluations, and obtain $\mathbf{\Lambda}_{t,+}$, $\mathbb{B}_t^*$, $\mathbb{X}_t$ and $\mathbb{Z}_t$ in the same way as was done for the marginal smooth for x. Our design matrix for the tensor product smooth results from taking box products of all the marginal bases

$$\mathcal{M} = [\mathbb{X}_x \square \mathbb{X}_t : \mathbb{X}_t \square \mathbb{Z}_t : \mathbb{Z}_x \square \mathbb{X}_t : \mathbb{Z}_x \square \mathbb{Z}_t] \,.$$

The term $\mathbb{X}_x \square \mathbb{X}_t$ corresponds to the unpenalized, fixed effects part of the smooth, and the three other terms are the random effects with each component having a separate ridge penalty.

Let $\mathbf{x} = \text{vec}(\mathbf{X})$ and $\mathbf{t} = \text{vec}(\mathbf{T})$; we can re-parameterize the null space bases as $\mathbb{X}_x = [\mathbf{1} : \mathbf{x}]$, $\mathbb{X}_t = [\mathbf{1} : \mathbf{t}]$, and $\mathbb{X}_x \square \mathbb{X}_t = [\mathbf{1} : \mathbf{x} : \mathbf{t} : \mathbf{x} \odot \mathbf{t}]$. The function $F(x,t)$ is decomposed into an unpenalized, parametric part and three orthogonal, nonparametric parts each subject to a different penalty

$$\underbrace{\text{term}}_{\text{penalty}} : \underbrace{F(x,t)}_{\lambda_t \int (\frac{\partial^2}{\partial t^2} F)^2 + \lambda_x \int (\frac{\partial^2}{\partial x^2} F)^2} = \underbrace{\beta_0 + \beta_1 x + \beta_2 t + \beta_3 x \cdot t}_{\text{unpenalized}} + \underbrace{f_1(t) + x f_2(t)}_{\lambda_1 [\int (\frac{\partial^2}{\partial t^2} f_1)^2 + (\frac{\partial^2}{\partial t^2} f_2)^2]}$$
$$+ \underbrace{g_1(x) + t g_2(x)}_{\lambda_2 [\int (\frac{\partial^2}{\partial x^2} g_1)^2 + (\frac{\partial^2}{\partial x^2} g_2)^2]} + \underbrace{h(x,t)}_{\lambda_3 \int (\frac{\partial^4}{\partial x^2 \partial t^2} h)^2} \tag{4.4}$$

The fully scaling-invariant construction for the basis which results in five

Term	Basis for functions of the form	Penalty
$\mathbb{X}_t \square \mathbb{Z}_t$	$f_1(t) + x f_2(t)$	$\int (\frac{\partial^2}{\partial t^2} f_1)^2 + (\frac{\partial^2}{\partial t^2} f_2)^2$
$\mathbb{Z}_x \square \mathbb{X}_t$	$g_1(x) + t g_2(x)$	$\int (\frac{\partial^2}{\partial x^2} g_1)^2 + (\frac{\partial^2}{\partial x^2} g_2)^2$
$\mathbb{Z}_x \square \mathbb{Z}_t$	$h(x,t)$ excluding previous blocks' bases	$\int (\frac{\partial^4}{\partial x^2 \partial t^2} h)^2$

Table 4.2: Description of penalized components of the tensor production construction (4.5).

smoothing parameters instead of three is

$$\mathcal{M} = [\mathbb{X}_x \square \mathbb{X}_t : \mathbb{Z}_x : \mathbb{Z}_t : \mathbf{x} \square \mathbb{Z}_t : \mathbb{Z}_x \square \mathbf{t} : \mathbb{Z}_x \square \mathbb{Z}_t],$$

with $F(x,t)$ decomposed as

$$\underbrace{\text{term}}_{\text{penalty}} : \underbrace{F(x,t)}_{\lambda_t \int (\frac{\partial^2}{\partial t^2} F)^2 + \lambda_x \int (\frac{\partial^2}{\partial x^2} F)^2} = \underbrace{\beta_0 + \beta_1 x + \beta_2 t + \beta_3 x \cdot t}_{\text{unpenalized}} + \underbrace{f_1(t)}_{\lambda_1 \int (\frac{\partial^2}{\partial t^2} f_1)^2} + \underbrace{x f_2(t)}_{\lambda_2 \int (\frac{\partial^2}{\partial t^2} f_2)^2}$$
$$+ \underbrace{g_1(x)}_{\lambda_3 \int (\frac{\partial^2}{\partial x^2} g_1)^2} + \underbrace{t g_2(x)}_{\lambda_4 \int (\frac{\partial^2}{\partial x^2} g_2)^2} + \underbrace{h(x,t)}_{\lambda_5 \int (\frac{\partial^4}{\partial x^2 \partial t^2} h)^2}$$

Both of the above tensor product constructions are available in the `mgcv` package using the `t2` smooth class. The interpretation of each penalized component is summarized in Table 4.2. Remembering that we must integrate w.r.t. t, it is clear that $\mathbf{t}$ must be dropped from the null space basis and that $\mathbb{Z}_t$ which corresponds to functions of the form $f_1(t)$, must be dropped from $\mathcal{M}$ as well. Define $\mathbb{X} = \mathbb{X}_x \square \mathbb{X}_t$, $\mathbb{Z}_1 = \mathbf{x} \square \mathbb{Z}_t$, $\mathbb{Z}_2 = \mathbb{Z}_x \square \mathbb{X}_t$, $\mathbb{Z}_3 = \mathbb{Z}_x \square \mathbb{Z}_t$ and the $N \times NJ$ matrix $\mathbb{L} = J^{-1}(\mathbb{I}_N \otimes \mathbf{1}_j^T)$ containing the quadrature weights for the integration. We can write our model as

$$\mathbf{y} = \theta_0 + \int_{\mathcal{T}} F(X_i(t), t)\, dt \approx \mathbb{L}\mathbb{X}\boldsymbol{\beta} + \sum_{j=1}^{3} \mathbb{L}\mathbb{Z}_j \mathbf{b}_j + \boldsymbol{\epsilon}; \tag{4.5}$$

$$\mathbf{b}_j \sim N(\mathbf{0}, \sigma_j^2 \mathbb{I}_{q_j}),\ j = 1, 2, 3;$$

$$\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma_e^2 \mathbb{I}_N).$$

Notice that this has the same form as Equation (1.1) in Greven et al.[37] and Equation (5) in Wang and Chen[123]. We have $q_0 = 3,\ q_1 = K_t - 2,\ q_2 = 2(K_x - 2),$ and $q_3 = (K_t - 2)(K_x - 2)$. Referring to Table 4.2, we can see that variance component σ_1^2 corresponds to the random effect for the FLM and that testing H_0: *FLM is the true model* vs. H_0: *FGAM is the true model* is equivalent to testing $H_0 :\ \sigma_2^2 = \sigma_3^2 = 0$ vs. $H_0 :\ \sigma_2^2 > 0$ and/or $\sigma_3^2 > 0$ with one nuisance variance component, σ_1^2. As mentioned previously, testing two variance components being zero simultaneously has received little attention in the literature in the mixed model literature.

To overcome this difficulty we consider several different approaches. The first approach is to do two tests each with one variance component being zero under the null hypothesis, one nuisance variance component, and one variance component fixed at zero under both hypotheses; and then to apply a Bonferroni correction to account for the multiple testing. In the first test, σ_3 is set to zero under both the null and alternative hypotheses and we test σ_2 for equality to zero. In the second test, σ_2 is set to zero under both hypotheses and we test σ_3. The Bonferroni correction is the simplest and most conservative commonly applied correction for multiple testing. The idea is very simple, if an α level test is desired and one has m tests, then one simply conducts each test at level α/m. This approach is guaranteed to have a family-wise error rate (probability of one or more type I errors) of at most α, but ignores any dependence between the tests and hence can be conservative. Another possibility would be to assume a priori that $\sigma_2^2 = \sigma_3^2$. Referring to Table 4.2, this assumption is difficult to interpret based on the functional forms and different interpretation of the penalties corresponding to σ_2 and σ_3. However, it does place our testing problem in the simpler setting of testing one variance component with one nuisance component.

Inspired by the Greven et al.[37] idea of forming pseudo-residuals based on an initial REML estimation, an additional approach we have considered is to first obtain REML estimates, say $\hat{\sigma}_2$ and $\hat{\sigma}_3$, of σ_2 and σ_3, respectively; and then form $\hat{\gamma} := \hat{\sigma_2}^2/\hat{\sigma_3}^2$. If we assume that $\sigma_2^2 = \gamma\sigma_3^2$ and replace γ by its estimate, then our test of FLM vs. FGAM is reduced to testing $\sigma_3 = 0$. However this, technique did not offer any performance improvements over the already discussed approaches and hence we do not consider it any further in the text.

4.5.1 A Test For No Effect In the Functional Linear Model

Before assessing whether an FLM or FGAM provides a better fit to the data, one will want to determine whether the functional predictor has any effect on the response at all. This is quite simple to test in our framework. By simply dropping the random effects $\mathbf{b}_2$ and $\mathbf{b}_3$, we can test for no effect by considering $H_0 : \ \beta_2 = \beta_3 = 0, \ \sigma_1 = 0$ versus $H_1 : \beta_2 \neq 0$ or $\beta_3 \neq 0$ or $\sigma_1 > 0$ (FLM is true). The exact distribution of the LRT statistic for this test is known due to Crainiceanu and Ruppert[17]. Note that a restricted likelihood ratio test is inappropriate here because the fixed effects are different under the two hypotheses. One can also use either a LRT or RLRT to test $H_0 : \ \sigma_1 = 0$ vs. $H_1 : \ \sigma_1 > 0$ which is a test that the effect of $X(t)$ is linear in t; $Y_i = \beta_0 + \mathbf{L}^T\{\mathbf{x}_i \odot (\beta_1 + \beta_2\mathbf{t})\}$. If one instead uses a first order penalty for x and t, then a test for no effect is equivalent to testing $\sigma_1 = 0$. This proposal is similar to one recently considered in Swihart et al.[113] for the penalized functional regression model of Goldsmith et al.[33]. Those authors first perform an FPCA to estimate the predictor trajectories (as was done in Chapter 3) and then estimate the coefficient function in the FLM using penalized splines with a first-order difference penalty and different mixed model representation than the

one considered here. It is also possible to test for a quadratic effect of the form $\int \zeta(t)X^2(t)dt$ if one uses a third order penalty for the marginal basis for x.

4.6 Bayes Factors For P-Spline ANOVA Models

In this section we develop Bayes factor approaches for testing for nonparametric effects in semiparametric regression models (including testing for linearity in FGAM). For a prior for the random effects, we use generalizations of the g-prior (Zellner[134]) which is widely used in the Bayesian variable selection literature due to its computational convenience.

We first discuss the choice of prior for the variance components in Section 4.6.1. The two approaches we consider make use of different prior specifications for the variance components as well as different priors for the random effects coefficients. In Section 4.6.2, we extend the work of Maruyama and George[66] for linear models to test for zero variance components in our penalized spline ANOVA model. In Section 4.6.3, we consider a simpler approach that extends the classic work for linear models of Zellner and Siow[135].

4.6.1 Choice of Priors For the Variance Components

In this section we briefly summarize a fraction of the rotund body of literature on the usage of shrinkage priors for Bayesian model selection problems, which ties in to the choice of prior distribution for scale parameters (variance components) in hierarchical models.

The g-prior is currently one of the most popular choices for model selection in Bayesian linear models. For a linear model $y|\alpha, \boldsymbol{\beta}, \sigma^2 \sim N(\alpha\mathbf{1} + \mathbb{X}\boldsymbol{\beta}, \sigma^2\mathbb{I})$, the g-prior is $\boldsymbol{\beta} \sim N(0, g\sigma^2(\mathbb{X}^T\mathbb{X})^{-1})$ (Zellner[134]) with g denoting a variance component instead of a precision parameter. The choice of g has been found to be crucial to the success of the fitting procedure. There is a rapidly growing body of literature considering the use of this prior and various generalizations for variable selection in high-dimensional Bayesian linear models. A small sampling of papers includes Liang et al.[62], Saville and Herring[101], Polson and Scott[85], Maruyama and George[66], Bayarri et al.[5], Celeux et al.[11], and Polson and Scott[87]. Note that the vanilla g-prior will not shrink any coefficients completely to zero and therefore cannot completely remove terms from the model. To achieve this requires a prior for the coefficients that includes a point mass at zero, a so called spike-and-slab prior. Coupling a point mass at zero with the various g-priors mentioned above is considered in Ley and Steel[58]. Those authors' extensive simulation studies demonstrated the effectiveness of the Maruyama and George[66] prior on g for performing model selection via Bayesian Model Averaging.

Nearly all work in the literature considers only using one g-prior per model. This is usually not desirable for nonparametric regression if we think of each g as a smoothing parameter controlling the complexity of the regression functional corresponding to the random effect. Preferable would be a separate g-prior for each regression functional so that each functional can have differing amounts of smoothing. One exception is the work of Rouder et al.[98], which is quite similar to the second approach we consider. Those authors allow for a separate g-prior to be placed on each random effect in a classical ANOVA model. They make use of the Zellner and Siow[135] g-prior arguing that it is more appropriate for categorical covariates than the hyper-g/n prior advocated by Liang et al.[62] to achieve

consistency of the Bayes factor. Two papers have recently appeared that examine the use of these priors for additive models. One is Sabanés Bové et al.[100], which uses a very different approach from the one we will take and does not examine tensor product smooths. In that paper, the authors use a different mixed model representation from the one considered here, integrate out the random effects, and place a generalized g-prior on the parametric part of each additive component in the model. The other is Shang and Peng[105] which considers the ultra-high dimensional setting where the number of covariates grows exponentially with the sample size. Both approaches use a spike-and-slab prior similar to the ones considered in Ley and Steel[58] to select covariates in the additive model.

Testing variance components using Bayes factors has previously been considered in Pauler et al.[82]. Chen and Dunson[13] consider the problem of random effect selection in linear mixed models, but do not make use of g-priors or consider Bayes factors. They too reparametrize the random effects, but using a Cholesky factorization instead of an SVD of the covariance matrix and then place a zero-inflated half-normal distribution on the resulting variance components.

Both methods we use can be expressed in the form

$$\begin{aligned}
\mathbf{y}|\boldsymbol{\beta}, \mathbf{b}_1, \ldots, \mathbf{b}_j, \sigma^2 &\sim N_N(\mathbf{X}\boldsymbol{\beta} + \sum_{j=1}^{J} \mathbb{Z}_j \mathbf{b}_j, \sigma^2 \mathbb{I}_N) \qquad (4.6)\\
p(\boldsymbol{\beta}) &= \mathbf{1}_{(-\infty,\infty)}(\boldsymbol{\beta})\\
\mathbf{b}_j|g_j, \sigma^2 &\sim N_{q_j}\{\mathbf{0}, \sigma^2 \boldsymbol{\Psi}_j(g_j)\}, \quad j = 1, \ldots, J\\
p(\sigma^2) &= \sigma^{-2} 1_{(0,\infty)}(\sigma^2)\\
p(g_j) &:= \pi(g_j), \quad j = 1, \ldots, J;
\end{aligned}$$

where $\pi(g_j)$ denotes the prior for g_j, which will be detailed shortly. The two approaches differ based on the form of the $\boldsymbol{\Psi}$ they use and on the form of the prior

used for each element of $\mathbf{g} = (g_1, \ldots, g_J)^T$. Notice that the priors on $\boldsymbol{\beta}$ and σ^2 are both improper because we are not interested in hypotheses about either component for our current problem. Typically improper priors give rise to Bayes factors that are not well-defined. By first considering the proper priors $p(\boldsymbol{\beta}; h_\beta) = \frac{1}{2h_\beta}\mathbf{1}_{(-h_\beta, h_\beta)}$ and $p(\sigma^2; h_\sigma) = \frac{\sigma^{-2}}{2\log h_\sigma}$; $h_\sigma^{-1} < \sigma^2 < h_\sigma$ and then letting $h_\beta, h_\sigma \to \infty$ when computing the required marginal densities, we obtain well defined Bayes factors (see; Maruyama and George[66]). This approach is reasonable because σ^2 and $\boldsymbol{\beta}$ appear in every model we would want to test in this framework. One reason for not specifying a proper prior on the fixed effects $\boldsymbol{\beta}$ is because for the moment we are only interested in tests not involving the fixed effects. If one did want to consider a test involving the fixed effects, say to test for no effect in the FLM as outlined in Section 4.5.1, one could try placing an additional g-prior on the fixed effects.

While the Rouder et al.[98] approach is simpler, it requires approximating two integrals of dimension J_i each, where J_i is the number of non-zero variance components in the model under hypothesis H_i, $i = 0, 1$. For linear models, the setup of Maruyama and George[66] allows for closed-form expressions for the Bayes factors to be obtained due to its use of a data-driven generalization of the g-prior. Their approach can accommodate the case of more parameters than data ($p > N$), but that case will not be considered in this dissertation.

4.6.2 An Approach Using Type IV Beta Priors

The prior used for g by Maruyama and George[66] is known as a Pearson Type VI or beta-prime distribution

$$p(g_j) = \frac{g_j^b(1+g_j)^{-(a+b+2)}}{B(a+1,b+1)};\ g_j > 0;\ a,b > -1;\ j = 1,\ldots,J \tag{4.7}$$

with a and b being hyperparameters to be specified and $B(\cdot,\cdot)$ denoting the Beta function. This prior has received considerable attention in the literature, with several different suggestions for the hyperparameters a and b. Polson and Scott[86] argues that this prior should replace the inverse-Gamma distribution as the default choice for variance components in hierarchical models. Smaller values of a result in heavier tails for the distribution of random effects; whereas smaller values of b make the distribution more concentrated near the origin (Polson and Scott[86]). The choice of b used by Maruyama and George[66] turns out to be very convenient for computational reasons, but somewhat unusual because it depends on the size of the model (for us the dimension of the corresponding random effect vector). The issues with the use of inverse-Gamma priors for variance components are well documented (e.g., Gelman[31]).

Temporarily ignoring the dependence on j, in the standard g-prior framework, the matrix $\mathbf{\Psi}$ would be $g(\mathbb{Z}^T\mathbb{Z})^{-1}$, or as recommended by Liang et al.[62], the scaled version $g(\mathbb{Z}^T\mathbb{Z}/N)^{-1}$. For linear models, the usual g-prior has the undesirable effect of putting stronger prior information on components that have been estimated with higher precision. It is for this reason that Maruyama and George[66] proposed using the diagonal matrix $\mathbf{\Psi}$ with diagonal entries $\psi_i = \frac{1}{d_i^2}[\nu_i(1+g)-1]$, where d_i^2, $d_1^2 > \cdots > d_q^2$, are the eigenvalues from an eigendecomposition of $\mathbb{Z}^T\mathbb{Z}$ and $\nu_i = d_i^2/d_q^2$, and we denote the eigenvectors by $\mathbf{u}_i$. Maruyama and George[66]

define

$$R^2 = \sum_{i=1}^{q} \frac{(\mathbf{u}_i^T \mathbf{r})^2}{\mathbf{r}^T \mathbf{r}} \text{ and } Q^2 = \sum_{i=1}^{q} (1 - \nu_i^{-1}) \frac{(\mathbf{u}_i^T \mathbf{r})^2}{\mathbf{r}^T \mathbf{r}}, \tag{4.8}$$

where $\mathbf{r} = \mathbf{y} - \bar{y}\mathbf{1}$. For those authors, who considered only multivariate linear models, R^2 was the usual coefficient of determination. In our case $\mathbf{r} = \{\mathbb{I} - \mathbb{X}(\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\}\mathbf{y}$, and R^2, Q^2, $\mathbf{u}_i$, and ν_i, are all dependent on their corresponding random effect, $\mathbf{b}_j$, and effect basis $\mathbb{Z}_j$, $j = 1, \ldots, J$. Thus, all those terms gain an index, j. Detailed calculations are provided in Appendix B.1.

In Appendix B.1 it is also shown that our marginal density is

$$M_{\mathcal{M}}(\mathbf{y}) = k \prod_{j=1}^{J} \left(1 - \sum_{j=1}^{J} Q_j^2\right)^{(N-q_0)/2} F_A\left(\alpha, b_1 + 1, \ldots, b_J + 1, \alpha, \ldots, \alpha; \upsilon_1, \ldots, \upsilon_J\right), \tag{4.9}$$

where $\upsilon_j := \frac{R_j^2 - Q_j^2}{1 - \sum_{j=1}^{J} Q_j^2}$, $\alpha := \frac{N - q_0}{2}$, and

$$k = \Gamma[(N - q_0)/2] \cdot |\mathbb{X}^T\mathbb{X}|^{-1/2} (\pi^{1/2} \mathbf{r}^T \mathbf{r})^{-N+q_0} \prod_{j=1}^{J} \frac{B(b_j + 1, a + q_j/2 + 1)}{B(a + 1, b_j + 1)} \prod_{l=1}^{q_j} \nu_{jl}^{-1/2}.$$

F_A denotes one of Lauricella's hypergeometric series[55], which is defined as follows

$$F_A(a, b_1, \ldots, b_n, c_1, \ldots, c_n; x_1, \ldots, x_n) = \sum_{i_1, \ldots, i_n = 0}^{\infty} \frac{(a)_{i_1 + \cdots + i_n} (b_1)_{i_1} \cdots (b_n)_{i_n}}{(c_1)_{i_1} \cdots (c_n)_{i_n} i_1! \cdots i_n!} x_1^{i_1} \cdots x_n^{i_n},$$

where $(a)_n$ denotes the rising factorial $(a)_n = \Gamma(a + n)/\Gamma(a)$. In one variable, all Lauricella's series simplify to Gauss' hypergeometric function and in two variables, F_A is equivalent to Appell's function F_2.

The series F_A converges for $\sum_{j=1}^{n} |x_j| < 1$. We know $\upsilon_j > 0$ ($x_j > 0$) because $R_j^2 - Q_j^2 > 0$. Considering that R^2 for the full model is bounded by one and that we have partitioned the model into orthogonal components and each R_j^2 is a semi-partial R^2 for one of the orthogonal components, it seems reasonable that the sum

of the R_j^2's should be bounded by one. This has been the case in the small number of numerical experiments we have done.

From e.g., Srivastava and Karlsson[110], Page 285, Equation (35); F_A can be represented by a one-dimensional Laplace type integral as follows

$$F_A(a, b_1, \ldots, b_n, c_1, \ldots, c_n; x_1, \ldots, x_n) = \frac{1}{\Gamma(a)} \int_0^\infty e^{-t} t^{a-1} \prod_{i=1}^{n} {}_1F_1(b_j, c_j; x_j t) dt, \tag{4.10}$$

where ${}_1F_1(a, b; x) := \sum_{i=1}^{n} \frac{(a)_i x^i}{(b)_i i!}$ is Kummer's confluent hypergeometric function which can be efficiently approximated. Lauricella's function F_A appears occasionally in literature on performance analysis of wireless communications systems (e.g., Annamalai et al.[2]). Several other applications in statistics and physics can be found in Exton[21], Ch. 7 and 8. An efficient method for computing (4.10) appears in Shi and Karasawa[107]. The approach is based on semi-infinite Gauss-Hermite quadrature with weights given by Steen et al.[111]. This approach was shown to offer less approximation error than the standard approach for approximating integrals on the positive real line, Gauss-Laguerre quadrature.

For the two variance components case, we have (Srivastava and Karlsson[110], p. 305, Eq. 107)

$$F_A(\alpha, b_1 + 1, b_2 + 1, \alpha, \alpha; \upsilon_1, \upsilon_2) = \frac{(1 - \upsilon_1)^{-b_1 - 1}}{(1 - \upsilon_2)^{b_2 + 1}} \times F\left(b_1 + 1, b_2 + 1, \alpha; \frac{\upsilon_1 \upsilon_2}{(1 - \upsilon_1)(1 - \upsilon_2)}\right).$$

Thus, a Bayes factor where each model has either one or two variance components reduces to computing a ratio of Gauss' hypergeometric functions. A method for accurately computing these ratios using continued fraction expansions is given in Wand and Ormerod[118].

As an example, the Bayes factor for testing $H_0 : \sigma_J = 0$ vs. $H_1 : \sigma_J > 0$ is

$$BF_{10} = \frac{B(b_J + 1, a + q_J + 1)}{B(a + 1, b_J + 1)}(1 - Q_J)^{(N-q_0)/2} \times \frac{F(\alpha, b_1 + 1, \ldots, b_J + 1, \alpha, \ldots, \alpha, \upsilon_1, \ldots, \upsilon_J)}{F(\alpha, b_1 + 1, \ldots, b_{J-1} + 1, \alpha, \ldots, \alpha, \upsilon_1, \ldots, \upsilon_{J-1})} \prod_{l=1}^{q_J} \nu_{Jl}^{-1/2}$$

Other tests have a similar form.

In Appendix B.2, we derive a slightly different form for the marginal density for FGAM by integrating w.r.t. each g_j individually. The result is a univariate integral involving only one hypergeometric function

$$\begin{aligned} M_{FGAM}(\mathbf{y}) &= \frac{k_2 B(a + q_2/2 + 1, b_2 + 1)}{B(a + 1, b_3 + 1)} \int_0^1 u_3^{b_3}(1 - u_3)^{a+q_3/2} c_2^{a+q_1/2+1} \\ &\times (Q_1^2 - R_1^2 + c_2)^{-b_1-1} \left(1 - \frac{R_2^2 - Q_2^2}{c_2 + Q_1^2 - R_1^2}\right)^{a+q_1/2+1} \\ &\times F\left[b_2 + 1, a + q_1/2 + 1, \frac{N - q_0}{2}; \frac{(R_2^2 - Q_2^2)(Q_1^2 - R_1^2)}{c_2(c_2 + Q_1^2 - R_1^2 + Q_2^2 - R_2^2)}\right] du_3 \\ &= \frac{k_2 B(a + q_2/2 + 1, b_2 + 1)}{B(a + 1, b_3 + 1)} \int_0^1 u_3^{b_3}(1 - u_3)^{a+q_3/2} c_2^{a+q_1/2+1} \\ &\times (Q_1^2 - R_1^2 + c_2)^{-(N-q_0)/2} \left(c_2 + Q_1^2 + Q_2^2 - R_1^2 - R_2^2\right)^{a+\frac{q_1}{2}+1} \\ &\times F\left[b_2 + 1, a + \frac{q_1}{2} + 1, \frac{N - q_0}{2}; \frac{(R_2^2 - Q_2^2)(Q_1^2 - R_1^2)}{c_2(c_2 + Q_1^2 - R_1^2 + Q_2^2 - R_2^2)}\right] du_3, \end{aligned}$$

where $k_2 = k_1 B^{-1}(a + 1, b_1 + 1)B^{-1}(a + 1, b_2 + 1)B(b_1 + 1, a + q_1/2 + 1)$ with $k_1 = \Gamma[(N - q_0)/2] \cdot |\mathbb{X}^T\mathbb{X}|^{-1/2}(\pi^{1/2}\mathbf{r}^T\mathbf{r})^{-N+q_0} \prod_{j=1}^{J} \prod_{l=1}^{q_j} \nu_{jl}^{-1/2}$. This integral can be approximated by, for example, Simpson's rule. However, hypergeometric functions can be difficult to approximate for argument values near one (Liang et al.[62]). Gauss's hypergeometric function and Kummer's confluent hypergeometric function can both be computed in `R` using the package `gsl` (Hankin[41]).

Under the FLM, the marginal density is given by

$$M_{FLM}(\mathbf{y}) = \frac{k_1 B(b_1 + 1, a + q_1/2 + 1)}{B(a + 1, b_1 + 1)}(1/3 - Q_1^2)^{(N-q_0)/2} \left(1 + \frac{Q_1^2 - R_1^2}{1/3 - Q_1^2}\right)^{-b_1-1}.$$

It follows that the Bayes factor for testing $H_0:$ FLM vs. $H_1:$ FGAM is given by

$$\begin{aligned} B_{10} = \frac{M_{FGAM}(\mathbf{y})}{M_{FLM}(\mathbf{y})} &= \frac{B(a+q_2/2+1, b_2+1)\prod_{j=2}^{3}\prod_{l=1}^{q_j}\nu_{jl}^{-1/2}}{B(a+1,b_2+1)B(a+1,b_3+1)} \\ &\times (1/3-Q_1^2)^{-(N-q_0)/2}\left(1+\frac{Q_1^2-R_1^2}{1/3-Q_1^2}\right)^{b_1+1}\int_0^1 u_3^{b_3}(1-u_3)^{a+q_3/2}c_2^{a+q_1/2+1} \\ &\times (Q_1^2-R_1^2+c_2)^{-(N-q_0)/2}\left(c_2+Q_1^2+Q_2^2-R_1^2-R_2^2\right)^{a+\frac{q_1}{2}+1} \\ &\times F\left[b_2+1, a+\frac{q_1}{2}+1, \frac{N-q_0}{2}; \frac{(R_2^2-Q_2^2)(Q_1^2-R_1^2)}{c_2(c_2+Q_1^2-R_1^2+Q_2^2-R_2^2)}\right]du_3, \end{aligned} \tag{4.11}$$

where $c_2 = u_3(Q_3^2-R_3^2)+1-\sum_{j=1}^{3}Q_j^3$.

4.6.3 An Approach Using Inverse-Gamma Priors

In this section, we do not use a data-dependent specification for the prior covariance for the random effects and simply use $\mathbf{b}_j \sim N(0, g_j\sigma^2\mathbf{I}_{q_j})$. An advantage of this is that this is our original prior/penalty from Section 4.4.2 and thus we have not changed the interpretation of our penalties in terms of function shape from that section. The downside of this is that we will have to numerically integrate over each g_j to obtain marginal densities. In their influential paper on Bayesian linear model selection, Zellner and Siow[135] proposed the use a multivariate Cauchy density for the marginal distribution of the coefficients being tested in a multiple linear regression. The use of the Cauchy marginal implies that the g parameter has an inverse-gamma distribution (Liang et al.[62]). In an attempt to extend their approach to our setting, we choose $\pi(g_j) = \text{Inv-Gamma}(1/2, 1/2) = \text{Inv-}\chi^2(1)$. The derivation of the marginal densities follows along the same lines as the derivation in Appendix B.1 for the proposal of the previous section, up to the integration w.r.t. $\mathbf{g}$. For this reason, we omit most of the details.

To give the form of the marginal density for a model with J random effects,

we first define the diagonal matrix $\mathbb{G}_J = \text{diag}(g_1\mathbf{1}_{q_1} \ldots, g_J\mathbf{1}_{q_J})$ and the covariance matrix $\mathbb{V}_J = \mathbb{Z}_{\{J\}}^T\mathbb{Z}_{\{J\}} + \mathbb{G}_J^{-1}$. The marginal density is then given by

$$M_J(\mathbf{y}) = \int_{\mathbb{R}_+^J} \frac{\Gamma\{(N-q_0)/2\}}{\pi^{(N-q_0)/2}} \frac{(\mathbf{r}^T\mathbf{r} - \mathbf{r}^T\mathbb{Z}_{\{J\}}\mathbb{V}_J^{-1}\mathbb{Z}_{\{J\}}^T\mathbf{r})^{-(N-q_0)/2}}{|\mathbb{X}^T\mathbb{X}|^{1/2}|\mathbb{G}_J|^{1/2}|\mathbb{V}_J|^{1/2}}\pi(g_1)\cdots\phi(g_J)d\mathbf{g}, \tag{4.12}$$

where $\mathbb{R}_+^J$ denotes the upper half-space of J-dimensional Euclidean space, $\mathbf{r} = \mathbf{y} - \mathbb{X}(\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\mathbf{y}$, and $\mathbb{Z}_{\{J\}} = [\mathbb{Z}_1 : \mathbb{Z}_2 : \cdots : \mathbb{Z}_J]$. Now for a null model with $J-1$ variance components, our Bayes factor is $B_{10} = M_J(\mathbf{y})/M_{J-1}(\mathbf{y})$. This is similar to the Bayes factor used in Rouder et al.[98], but those authors only consider classical ANOVA models and thus have categorical $\mathbb{Z}$ matrices.

4.7 Testing FLM Versus FGAM: Simulation Study

In this section we study the performance of our proposed tests for linearity of FGAM on simulated data for two different setups. First, in Section 4.7.1, we generate the response variable using a convex combination of an FLM and an FGAM in the functional predictor. This is done to assess the size of the departure from linearity that our tests can detect in a way that is interpretable in terms of the original models. In Section 4.7.2, we assess empirical type I error rates and power for our tests by generating the response from the mixed model in Section 4.5 for several different values of the variance components and compare performance with tests that know the value of the nuisance parameters.

A summary of the testing procedures we consider in this section can be found in Table 4.3 along with a reference to their introduction point in the dissertation. Not all testing procedures were considered in both simulation sections; this is also indicated in the table. We consider one method not previously introduced

Abbr. Name	Section	Description
Bonferroni	4.7.1, 4.7.2	Two separate pseudo-RLRTs for each non-FLM variance component for SSANOVA-like parameterization from Section 4.5 with Bonferroni correction
Augment	4.7.1	RLRT using usual tensor product parameterization and augmented data; see Section 4.4.1
BF	4.7.1	Bayes factor approach with inverse-gamma priors on variance components; see Section 4.6.3
GPGMFB	4.7.1	Cramér-von Mises test for goodness of fit of FLM proposed in García-Portugués et al.[30]
EqualVC	4.7.1, 4.7.2	pseudo-RLRT using SSANOVA-like parameterization from Section 4.5, but assuming $\sigma_2 = \sigma_3$
DCOR	4.7.2	Using distance correlation t-test for independence from Székely and Rizzo[114]
KnownSig1	4.7.2	"quasi-Oracle" test that knows the true value of the random effects corresponding to FLM component in (4.5)

Table 4.3: Description of all methods considered for testing for linearity in the simulation studies.

in the thesis that has seen little attention for functional data, but seems potentially promising: the distance correlation of Székely et al.[115]. Distance correlation appears well-suited for functional regression because it can measure independence between two stochastic processes in differing, arbitrary dimensions and is equal to zero if and only if the two processes are independent. Recently the same authors have developed a test especially suited for high-dimensional random vectors (Székely and Rizzo[114]). If the FLM is a good fit to the data, then the residuals and the functional predictor should be independent. Thus, for comparison we consider computing the distance correlation between the functional predictor and the residuals from an FLM fit to the data, and then using the proposed test of Székely and Rizzo[114]. The test can be conducted in `R` using the package `energy` (Rizzo and Szekely[96]). The use of distance correlation for model selection in SSANOVA models for multivariate data has been explored by Kong et al.[52].

We also consider the recently proposed method of García-Portugués et al.[30]. This is the only work besides ours we our aware of that focuses on a null hypothesis of the FLM being true. They use a Cramér-von Mises statistic and use the bootstrap to approximate the null distribution of the statistic. The method relies on an assumption that the coefficient function can be expanded in a finite number of basis functions without penalization. Their method is implemented in the `R` package `fda.usc` (Febrero-Bande and Oviedo de la Fuente[25]). Due to singularities in the model matrix when estimating an FLM using their package, we were unable to get their method to work for more than four basis functions for the functional coefficient for any of our simulations. This seems to be due to the lack of regularization in the method. We therefore only report results for the four basis function case for this method. We label this method GPGMFB.

RLRTs, for the methods based on them, are computed in `R` (R Core Team[88]) using the package `RLRsim` (Scheipl et al.[102]). The package requires fitted model objects for the model under both hypotheses, as well as a fit to the data with nuisance variance components equal to zero. These fits are obtained using the package `lme4` (Bates et al.[4]). For method Augment, the REML estimation procedure used to estimate FGAM and the FLM is due to Wood[128] and available in the package `mgcv` (Wood[126]). The code used to estimate Bayes factors is an extension of code available in the package `BayesFactor` (Morey and Rouder[72]). At the time of this writing, that package can only be used for a model with one variance component.

4.7.1 True Model as Convex Combination of FLM and FGAM

We consider two sample sizes, $N = 100, 500$; three significance levels, $\alpha = 0.1, 0.05, 0.01$; and three values for the number of basis functions for each axis, $K_x = K_t = 5, 8, 10$. Data will be generated in a similar manner to Section 3.5. We fit each model to 500 simulated data sets. The true functional covariates are given by $X(t) = \sum_{j=1}^{4} \xi_j \phi_j(t)$, with $\xi_j \sim N(0, 8j^{-2})$ and $\{\phi_1(t), \ldots, \phi_4(t)\} = \{\sin(\pi t), \cos(\pi t), \sin(2\pi t), \cos(2\pi t)\}$. Each functional predictor was observed at 30 equally-space points. To generate the response, we take a convex combination of a bivariate function linear in x and one nonlinear in x, $F_1(x,t) = 2x\sin(\pi t)$, and $F_2(x,t) = 10\cos\left(-\frac{x}{8} + \frac{t}{4} - 5\right)$, with $t = [0,1]$. The response is given by

$$Y_i = \int_0^1 \lambda F_1\{X_i(t), t\} + (1-\lambda)F_2\{X_i(t), t\}dt + \epsilon_i,$$

with $\epsilon_i \sim N(0,1)$ and $0 \leq \lambda \leq 1$. The constants in F_1 and F_2 were chosen so that each surface contributed roughly equally to the signal for each generated data set (prior to multiplication by λ). Both true surfaces along with some generated functional predictors are shown in Figure 3.2. Note that the figure additionally displays sparse, noisy measurements of the functional predictor which applied for the simulation study of the previous chapter, but not the current one.

The results for this simulation study are summarized in Figure 4.1 where for each of our proposed tests we plot the proportion of the 500 simulations where the null hypothesis is rejected by $(1-\lambda)$. We use $1-\lambda$ so that zero on the x-axis corresponds to the null model (FLM) being true. Each panel contains the results for the three different values for the number of basis functions, with different colours representing each. Different column panels correspond to different

sample sizes and different panel rows correspond to different significance levels for the hypothesis test. For the Bayes factor approach, we used Table 4.1 as our guideline for assessing significance. To facilitate comparisons with the frequentist methods and as an affront to Bayesians everywhere, we mapped positive evidence as corresponding to a p-value < 0.1, strong evidence to a p-value < 0.05, and very strong evidence to a p-value < 0.01.

Since the values in the plot when $\lambda = 1$ can be difficult to distinguish across methods, Figure 4.1 also contains a table which gives the proportion of rejections when $\lambda = 1$ (i.e. the observed type I error or false discovery rate), which we denote $\hat{\alpha}$. The reported values in the table are the $\hat{\alpha}$'s averaged over the two sample sizes and three values for the number of basis functions. We see that the augmented data approach of Section 4.4.1 turns out to not give reasonable results, having a far too high number of false discoveries. The EqualVC method is able to achieve a type I error rate fairly close to the nominal level and also has the highest power of any of the methods not including the sorry Augment method. Method Bonferroni with is seen to be conservative, as expected, though not nearly as conservative as the Bayes factor method. GPGMFB has lower power and observed type I error rate further from the nominal level than method EqualVC, but is less conservative than Bonferroni and BF. Our choice for the mapping of Bayes factors to p-values seems to be quite poor.

There appears to be little effect on any of the methods as the number of basis functions change, with the exception being the BF method for the small basis function, small size setting. Note that the choice of $K_x = K_t = 5$ results in a random effect vector which is only of dimension three. This is quite small and is difficult for `lme4` to estimate. We note that `lme4` would frequently (correctly)

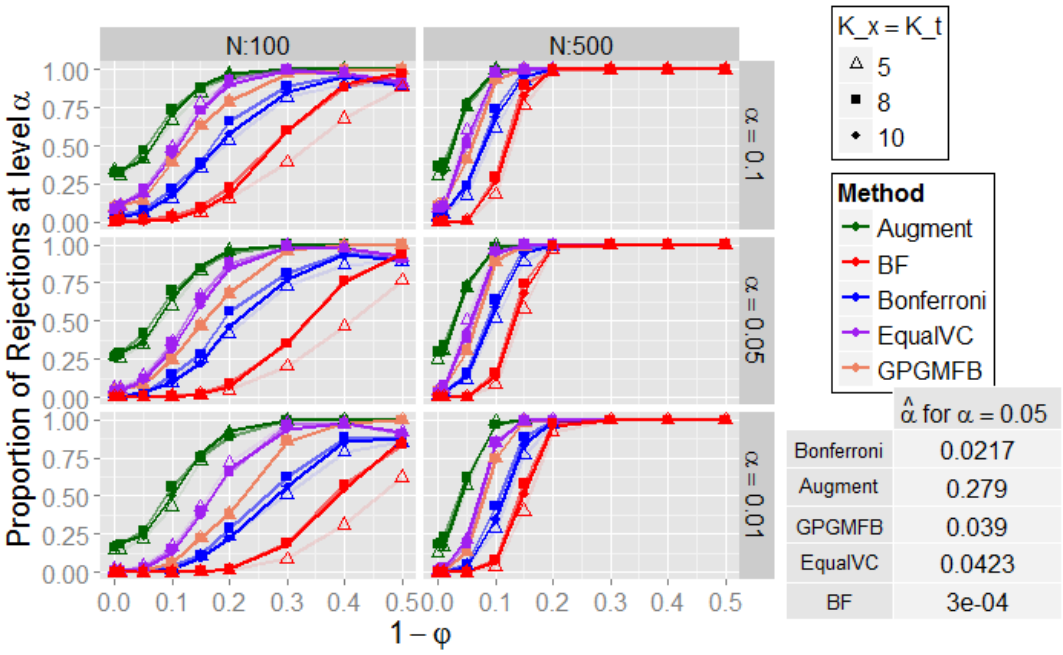

Figure 4.1: Proportion of rejected null hypotheses for various test levels (α), number of basis functions (K_x, K_t), sample sizes (N), and λ values for each testing method under consideration. See the text for more description.

estimate variance components to be zero in the $\lambda = 1$ case (when the FLM is true). When this happens, `RLRsim` cannot conduct an RLRT. When this occurred for method EqualVC, the results in Figure 4.1 include these cases as failures to reject the null hypothesis. Similarly, when *both* fitted models under the two alternatives had estimated zero variance components for method Bonferroni, which involved conducting two tests, these cases were counted as failures to reject. More problematic are cases where `lme4` incorrectly estimates the variance component to be zero when fitting the null model or estimates σ_2 or σ_3 to be zero when fitting the alternative model and $\lambda < 1$. This happened infrequently, with the slight exception being values of λ close to 0.5 for method Bonferroni and EqualVC. This is the cause of the slight downward bend at the right end of the power curves for those methods plotted in Figure 4.1. This could be the result of identifiability issues as the signals corresponding to the FLM and FGAM become roughly equal.

4.7.2 True Model as SSANOVA-like Mixed Model

We now change how the response is generated so that it comes from the mixed model (4.5). The functional predictors are generated in the same manner as the previous section. For a given simulated sample of N curves, the response is formed as follows. First, the Section 4.5 parameterization is used to form the bases $\mathbb{X},\ \mathbb{Z}_1,\ \mathbb{Z}_2,$ and $\mathbb{Z}_3$. Next, the random effect vector for the nuisance variance component corresponding to the FLM term (see Table 4.2) in the construction is drawn as $\mathbf{b}_1 \sim N(\mathbf{0}, \mathbb{I}_{q_1})$ and the two random effects vectors corresponding to non-FLM terms are drawn as $\mathbf{b}_j \sim N(\mathbf{0}, \sigma_j^2 \mathbb{I}_{q_j});\ j = 2, 3$. The response is than given by

$$\mathbf{Y} = \mathbb{X}\boldsymbol{\beta} + \sum_{j=1}^{3} \mathbb{Z}\mathbf{b}_j + \boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \mathbb{I}_N)$ and $\boldsymbol{\beta} = (1, 0.01, 0.01)^T$.

Referring to Table 4.3, note that the testing procedures we are comparing have changed for this section. While methods Bonferroni and EqualVC remain, we do not consider the method Augment which was not able to satisfactorily detect the FLM in the previous section, and we also do not use method BF. Instead, to better assess the performance of method Bonferroni, we consider a "quasi-oracle" test that knows the true value of the nuisance random effects vector for each simulation. In more detail, the pseudo-residuals used as inputs to the pseudo-RLRTs for this method are $\mathbf{Y} - \mathbb{Z}_1\mathbf{b}_1$, for the true $\mathbf{b}_1$ instead of its prediction using REML. The method still tests σ_2 and σ_3 separately and uses a Bonferroni correction, but with no nuisance variance component and only one variance component to test at a time, we are in exactly the framework of Crainiceanu and Ruppert [17], where the distribution of the test statistic is known and easily simulated from. We label this method "KnownSig1". The other procedure considered is the distance correlation

approach described earlier, which we label "DCOR".

Note also that changing the number of basis functions with this data generation scheme changes the dimension of the random effects in the true model. We can perhaps gain some insight into the performance of pseudo-RLRTs for penalized spline mixed models as the dimension of the random effects grow. The values of σ_j^2 $j = 2, 3$; considered are $\sigma_j^2 = (0, 0.04, 0.1, 0.25, 0.5, 0.75)$ for $N = 100$ and $\sigma_j^2 = (0, .004, .04, .14, .2, .3)$ when $N = 500$. Of particular interest will be how much method EqualVC suffers for assuming that $\sigma_2 = \sigma_3$ as that assumption becomes further and further from the truth.

As in the previous section, we generate 500 data sets for each simulation setting and report the proportion of times each method rejects the null hypothesis that the FLM is the true model. The empirical power of the proposed tests for significance level $\alpha = 0.05$ is plotted in Figure 4.2. Each panel corresponds to a different combination of sample size and number of basis functions and contains one point for each test for each combination of $(x = \sigma_2^2, y = \sigma_3^2)$. Larger point sizes correspond to a larger proportion of rejected null hypotheses and different colours differentiate the different methods. The points have been jittered slightly to distinguish the four testing procedures at each $(x = \sigma_2^2, y = \sigma_3^2)$ grid point.

The method EqualVC appears to be the most powerful of the four tests for this study. We see similar levels of disparity between EqualVC and Bonferroni as in the simulation study of the previous section. The DCOR method is seen to be least powerful here. It is promising that method EqualVC for the most part outperforms the method that knows the nuisance random effect. We can also see that Bonferroni is competitive with KnownSig1 for moderate to large values of σ_1^2

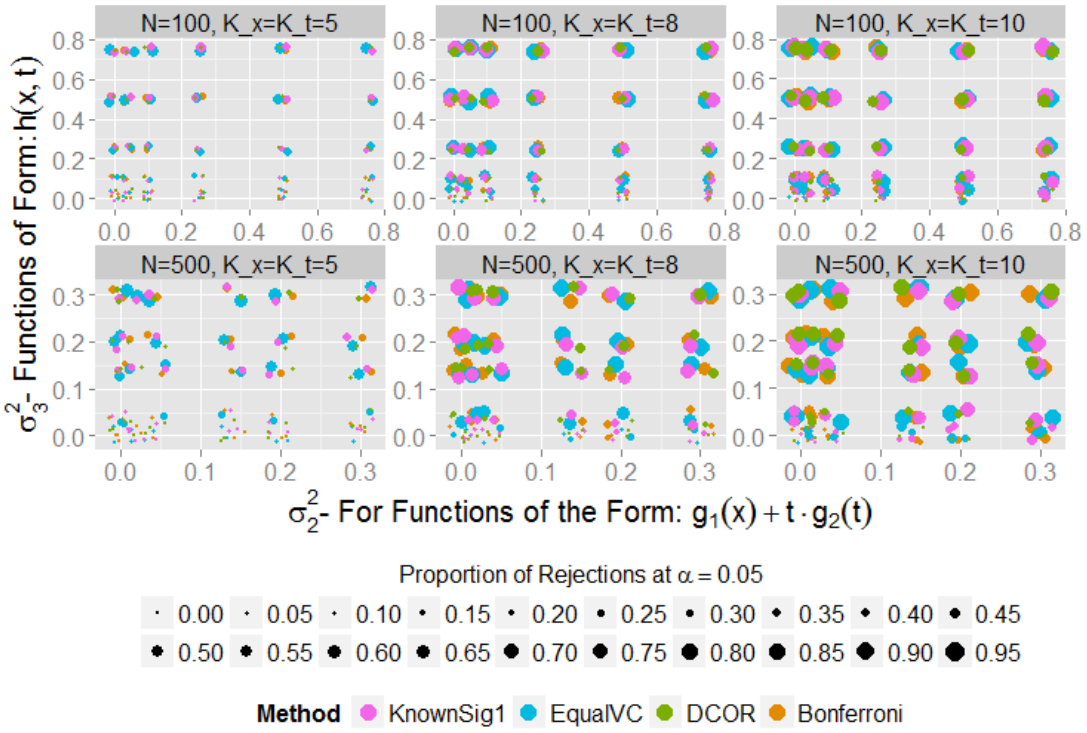

Figure 4.2: Proportion of rejected null hypotheses at $\alpha = 0.05$ over 500 simulations for a grid of several values of σ_2^2 and σ_3^2. The points have been jittered in order to distinguish differences between the methods. See the text for more description.

and σ_2^2.

It is clear that nonzero σ_2 and σ_3 become easier to detect for all four methods as the number of basis functions increases. Comparing the top left and bottom right quadrants of any panel, we see it is much easier to detect σ_3 being non-zero when σ_2 is small or zero than vice versa. Note that the relation between the dimension of $\mathbf{b}_2$ and $\mathbf{b}_3$ in this setup with $K_x = K_t$ and second-order penalties is $q_3 = q_2^2$. We also see that method EqualVC does not seem to lose its advantage in power over the other methods when one variance component is non-zero while the other is zero. It must be noted however, that pseudo-RLRTs for EqualVC could not be performed for a large proportion of the simulations when at least one of the variance components was zero or very near zero. This was due to zero

variance component estimates being returned by `lme4` and was especially true for the small σ_2 cases when the number of basis functions for each axis was five. As was mentioned, the use of $K_x = K_t = 5$ results in a random effect which is very difficult for `lme4` to reliably estimate. The problem was greatly reduced when the number of basis functions went up to ten.

While the zero estimates are desirable when both σ_2 and σ_3 are zero, it is important to be able to detect the cases when only one of the two are zero. If one wants to double-check a zero variance component estimate returned by `lme4` when using EqualVC, method Bonferroni is ideally suited for this because it tests each component separately. When `lme4` estimates say, σ_2 to be zero, the test for σ_3 being zero is still conducted so that the null hypothesis that the FLM is true can still be rejected if there turns out to be significant evidence that σ_3 is nonzero with σ_2 set to zero. The proportion of simulations where both RLRTs could not be conducted for method Bonferroni was similar to the previous section. Another possibility is to use a parametric bootstrap as suggested by Pinheiro and Bates[84].

The type I error rates varied little as either the sample size or number of basis functions changed. For this reason, we simply report the observed values averaged over the different settings for those parameters. For $\alpha = 0.05$, the empirical type I error rate for Bonferroni was 0.021, 0.046 for EqualVC, 0.000 for DCOR, and 0.023 for KnownSig1. Given how close its rate is to the nominal level and its strong power performance compared to the other methods in both simulation sections, we recommend using the EqualVC method which assumes a priori $\sigma_2 = \sigma_3$ and then conducts a single pseudo-RLRT using the Greven et al.[37] approach. We recommend the Bonferroni method or a Bayes factor approach be used as additional checks for nonlinearity for the occasional cases where `lme4` estimates zero variance

components that prevent EqualVC from conducting the necessary pseudo-RLRT.

4.8 Analysis of Emissions Data

In this section we apply our proposed procedures to study the quantities of various pollutants in truck exhaust emissions. The data come from chassis dynamometer emissions readings from the Coordinating Research Council E55/59 emissions inventory program (Clark et al.[14]). The goal of the study was to assess particulate matter emissions in heavy-duty trucks in California. Vehicles were tested in a lab setting designed to mimic everyday driving conditions. Particulate matter was captured using 70 mm filters on the dilute exhaust. For each sample a truck was subjected to one of four driving cycles; for example, cruising at highway speeds or stop-and-go city driving. For our application, we wish to predict the logarithm of particulate matter at the end of 40 seconds of driving using the speed and/or acceleration trajectories over the 40 seconds. For simplicity, we have down-sampled the data to avoid temporal dependence between response samples. Figure 4.3 plots both the original velocity data and estimated accelerations for all trucks in the data grouped according to driving cycle.

In the subsection that follows, we analyze the fit of the FLM to these data using our proposed tests for linearity. After that, we compare FLM and FGAM out-of-sample predictive performance for this data set and also compare predictive performance of the SSANOVA-like parameterization discussed in Section 4.5 with the parameterization used in Chapter 2.

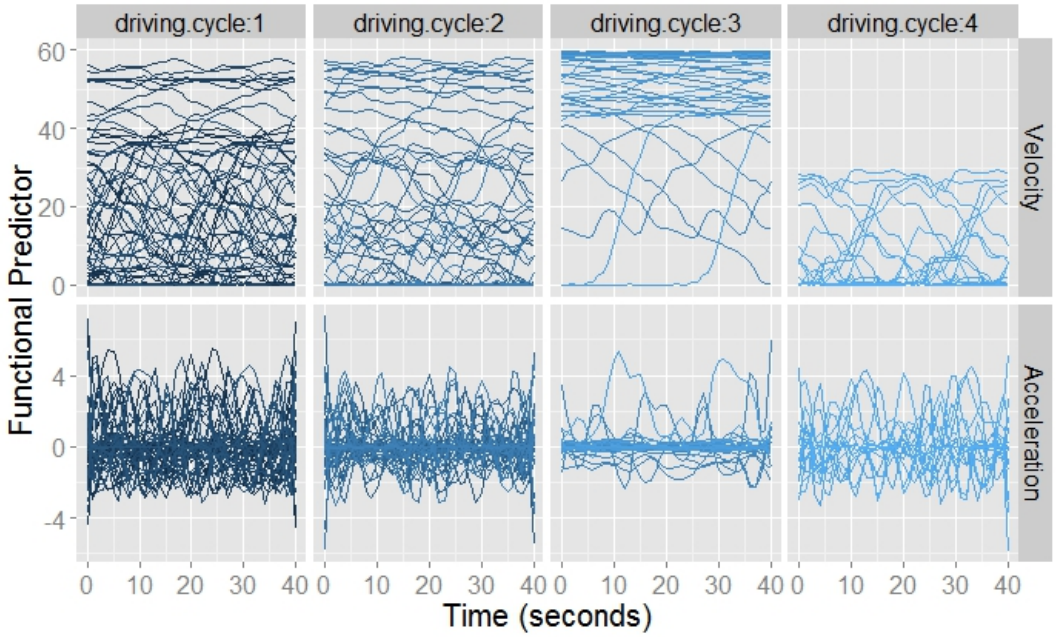

Figure 4.3: Speed and acceleration trajectories over forty seconds for each truck in the emissions study grouped according to velocity pattern (driving conditions).

4.8.1 FLM Fit Assessment

Leaving the finer details of our fitting procedures to the next section, we now discuss some diagnostics for assessing the fit of both the FLM and the FGAM including use of our proposed testing procedures. We consider predicting particulate matter using vehicle acceleration as the functional predictor and also include a categorical covariate for the driving cycle. Some residual plots for an FLM fit to the entire data set using tuning parameters that had been chosen to optimize performance for the next section are given in Figure 4.4. The top row of plots shows the residuals grouped according to the driving cycle covariate and the bottom shows the residuals plotted against the predicted value and also a normal Q-Q plot of the residuals. We can see a very strong correlation between the residuals and the response and also that the variance of the residuals is not constant across the driving cycle factor. The Q-Q plot indicates non-normality of the residuals.

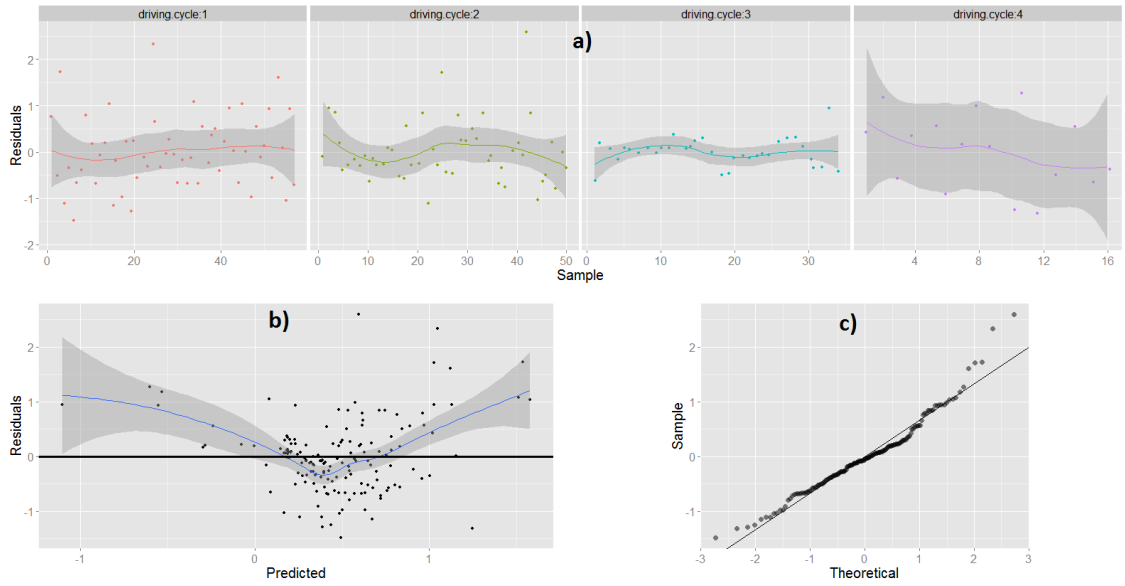

Figure 4.4: Residual plots for an FLM fit with truck acceleration as the functional predictor: a) plots the residuals grouped by driving cycle, b) plots residuals vs. predicted value, and c) is the normal Q-Q plot.

The p-value for the Shapiro-Wilk test for normality of the residuals was $< 10^{-6}$. This suggests that FLM is a poor fit for this data. To assess this more formally, we consider the proposed tests of Section 4.5 and Section 4.6.3. Using the procedure that conducts one RLRT after assuming $\sigma_2 = \sigma_3$ in the SSANOVA-like formulation of FGAM (see (4.2)), we obtain a p-value ≈ 0. This very strongly suggests the FLM is not adequate here. We also obtain a p-value that is zero to machine precision using the method involving two separate RLRTs for each non-FLM variance component with a Bonferroni correction. The Bayes factor using the Zellner-Siow prior for the variance components was approximately 2023, indicating extremely strong evidence the FGAM is to be preferred (recall Table 4.1). The results remain overwhelming regardless of the number of basis functions used. As a final check, we can assess the independence of the residuals using the distance correlation t-test (Székely and Rizzo[114]). We obtain a p-value that again is zero to machine precision.

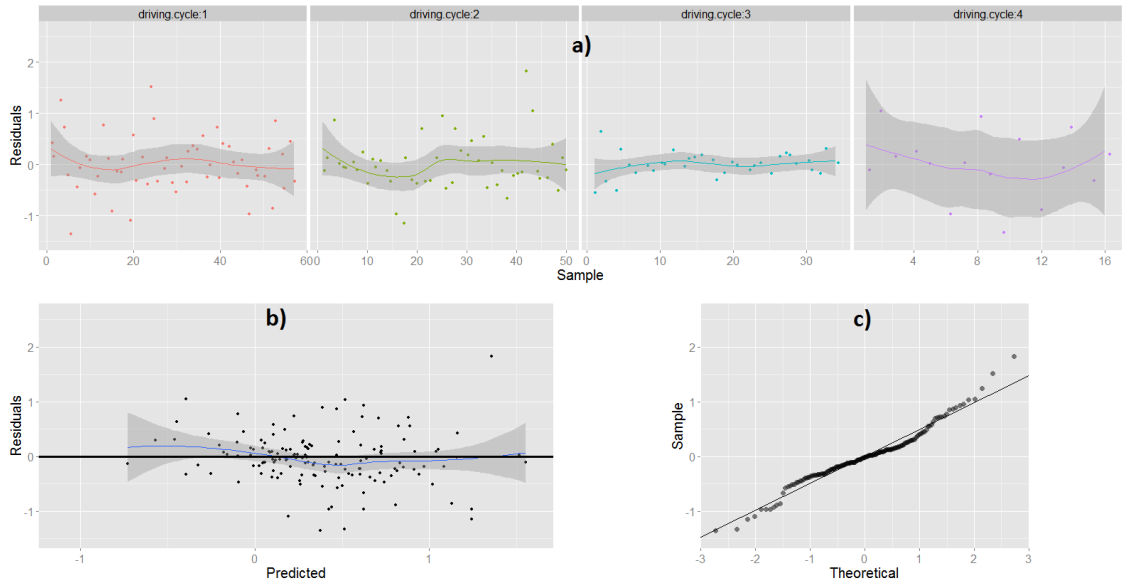

Figure 4.5: a) Residuals vs. sample grouped by driving cycle for FGAM fit with acceleration trajectories for predictors, and b) Residuals vs. response.

Finally, the residual plots for an FGAM fit to the data using the basis construction from this chapter can be seen in Figure 4.5. We can see that the magnitude of the residuals has gone down and that all three plots seem to be less in violation of the model assumptions than the FLM fit. The variance of the residuals also appears to be more constant across driving cycles. The p-value for the distance correlation t-test increased to 0.15, failing to find evidence that the residuals and functional predictor are not independent.

Figure 4.6 shows contours of the estimated surface obtained by using all 157 samples and the acceleration curves as predictors. The surface was estimated using `lme4` (Bates et al.[4]). Also plotted are the individual components of the basis construction of Section 4.4.2; the unpenalized component, along with the three penalized components (see (4.2)). The marginal bases for the x and t axes were both of dimension eight. Interestingly, the variance component for the FLM

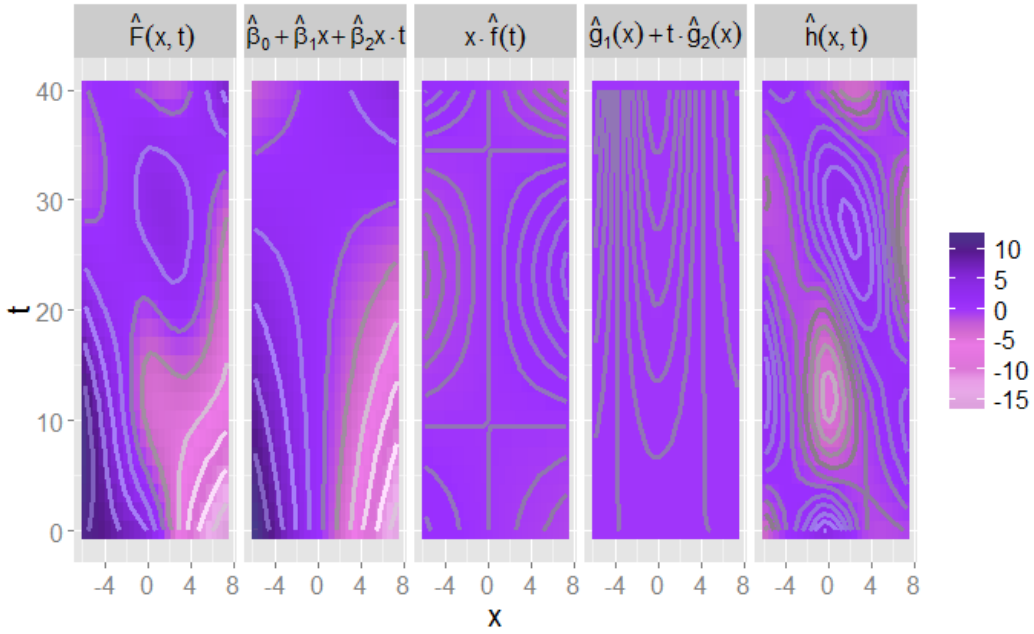

Figure 4.6: Contours of estimated surface, $\hat{F}(x,t)$, and components of FGAMM fit from using acceleration trajectories as predictors. The second panel is the parametric component of the fit, third is the component parametric in x and nonparametric in t, fourth vice versa, and finally $\hat{f}(x,t)$ is nonparametric in both and subject to a fourth order penalty.

portion of the fit was estimated to be very close to zero in this case.

The FLM fairs only marginally better if the truck speeds are used as the functional predictor. We omit the diagnostic measures, but the out-of-sample prediction performance using either covariate or both is examined in the next section.

4.8.2 Out-of-Sample Prediction of Particulate Matter

As further confirmation that the FGAM provides a better fit to this data than the FLM, we considered out-of-sample prediction of the log-particulate matter. We also compare the two different basis constructions for the tensor product surface in our model: the Chapter 2 construction (FGAM) and the construction from this

chapter (FGAMM). We also fit the fully nonparametric kernel estimator of Ferraty and Vieu[29] from 2.9. We considered fitting one functional predictor models using the truck velocities and the accelerations, as well as models including both functional covariates at once. Since multiple functional predictors or scalar covariates do not seem to be implemented for the model in Ferraty and Vieu[29], we could not consider the model with both functional predictors for that method. For this reason, also we did not include a categorical predictor for the driving cycle for any of the models. Including the categorical predictor for the FLM and FGAM methods had no effect on the results.

For FLM, FGAM, and FGAMM, smoothing parameters were chosen using REML. The nonparametric kernel estimator was fit using code from the authors, which includes automatic bandwidth selection and can be obtained from: `http://www.math.univ-toulouse.fr/staph/npfda`. Several different basis dimensions were considered for both the FLM and FGAMs. Results varied little as the number of basis functions changed for each of the methods, so for compactness we only report the values that produced the lowest root-mean-square error (RMSE) averaged over the different predictors for each method. For the FLM this was ten basis functions for the functional coefficient, for FGAM this was six basis functions for both axes, and for FGAMM this was ten basis functions for each axis for the one functional predictor models and seven basis functions for the two predictor model. Both the FGAM and the FLM can be fit in `R` using the `refund` package (Crainiceanu et al.[16]). The underlying estimation is handled by the `R` recommended package `mgcv` (Wood[128]). For FGAMM, the variance components are estimated by the package `lme4` (Bates et al.[4]). The data was randomly divided so that 105 samples ($\approx$ two thirds of the data) were using for training the models and the remaining samples were used for testing. Boxplots of the RMSEs for predicting

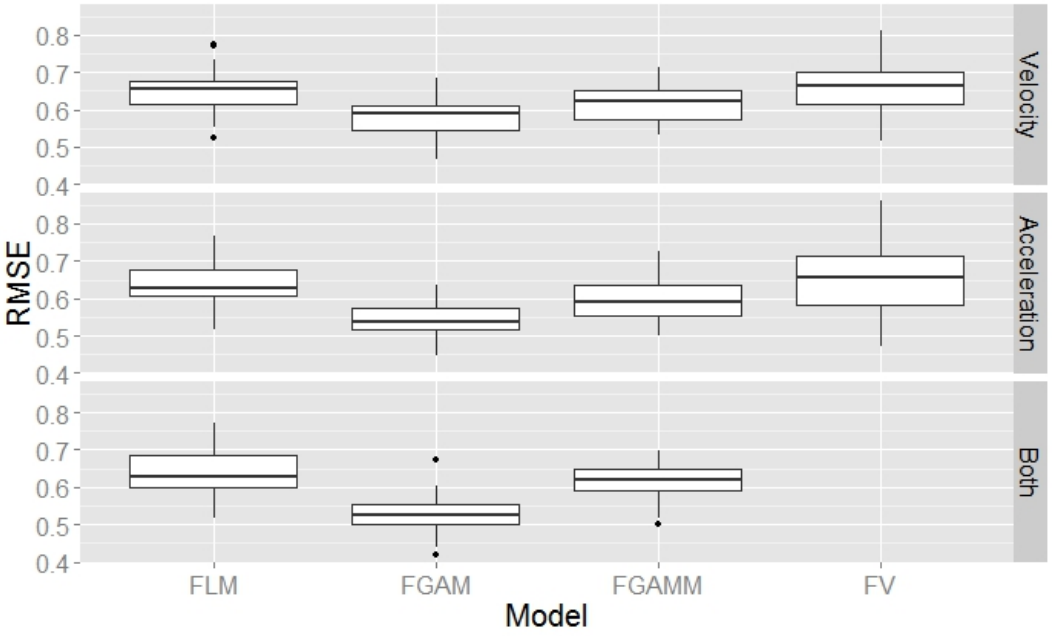

Figure 4.7: Boxplots for prediction error for FLM, FGAM, and FGAMM using truck velocity, acceleration, or both at once as predictors.

the test set samples over 25 different random partitions into training-test sets is displayed in Figure 4.7.

We see that both FGAM and FGAMM had lower mean RMSE for the velocity, acceleration, and two functional predictor models. While FGAMM performed similarly to FGAM when truck velocity was the functional predictor, the FGAM basis construction provided better out-of-sample predictions for the other two models. Both parameterizations for the FGAM gave superior prediction results to the Ferraty and Vieu model as well. The lowest mean RMSE was achieved by the FGAM that included both functional predictors and used the more standard tensor product construction. A method for continuously predicting emissions over time for this data using a functional response model is considered by Asencio et al.[3]. A non-functional data approach on an expanded data set from the original study authors can be found in Clark et al.[15].

CHAPTER 5

DISCUSSION

5.1 Conclusions

Continued advances in technology and data collecting can be expected to increasingly often produce samples which are natural to analyze as functional data. Methods for functional data analysis will surely receive more and more attention in a variety of scientific fields as these types of data sets proliferate. This dissertation makes an important contribution to the field by introducing a new model for regression when one wishes to use sampled functions as inputs to predict a scalar response variable. The functional linear model has been extended to an additive, nonparametric structure which allows for more complicated relationships to be modelled while still being highly interpretable. Our approach can handle responses from any exponential family distribution as well as multiple functional or scalar predictors. We have demonstrated the effectiveness of the model in a number of applications; modelling health outcomes using brain scans from diffusion tensor imaging, predicting closing price of online auctions, and predicting truck exhaust emissions using travel speeds over short trips in various driving situations.

In Chapter 2, we introduced and confirmed the efficacy of estimation and inference procedures for FGAM that relied on penalized splines. We showed that the FGAM can provide nearly identical prediction accuracy to the FLM when the FLM is the true model, and offered substantial improvements when the FLM was not the true model. We also showed that our proposed confidence bands can achieve average coverage probabilities close to the nominal confidence level. For the analysis of the DTI dataset, FGAM performed favourably when compared with some

standard functional regression models.

Applications where the functional data have significant missingness and measurement error are very common. It is also the most often encountered form for data sets in longitudinal data analysis. The work of Chapter 3 greatly extends the applicability of FGAM, allowing it to still be fit to data with high amounts of sparsity. We proposed two algorithms for fitting the FGAM after first expressing it as a linear mixed model, we then took a Bayesian hierarchical modelling approach and fit the model using a Metropolis-within-Gibbs sampler. Our MCMC algorithm was able to provide useful inferences in difficult situations where initial estimates provided by standard FPCA methods were quite poor due to rank deficiency in the estimated covariance matrix. Our algorithms also worked well when the data had several significant modes of variation present.

As functional data sets grow in size, it is important to have algorithms that can quickly obtain approximate solutions for estimating functional regression models. We developed a VB algorithm in a difficult setting involving multiple nonconjugate full conditionals, which provided a substantial reduction in computation time over MCMC while maintaining accuracy. Due to the shortened computation time of our VB algorithm, computationally intensive methods for inference become feasible. For example, one could bootstrap our VB fits to obtain improved estimates of the standard errors necessary for constructing confidence bands for the true surface $F(x, t)$, as well as bands for the true trajectories $X(t)$. We also demonstrated the benefits of initializing our MCMC algorithm at the final estimates returned from our VB method to achieve faster convergence of the Markov chain. In addition, we found that a two-step approach of first using standard functional data methods to recover the function predictors and then fitting a functional regression model was

inadequate due to singularities or near singularities in the estimated covariance operator for the curves.

The last chapter of this dissertation is a first step to answering a very important question, which to this point has few answers in the literature: when is a scalar on function regression problem not well-modelled by the functional linear model? How often is the relationship between the response and functional predictor truly nonlinear? Using an alternative mixed model representation for FGAM, we are able to develop several simple tests for assessing linearity of an FGAM fit to functional data. Through two simulation studies we were able to find an approach that gave type I error rates quite close to the nominal level and also had high power. In an application to measuring the amount of pollutants emitted by heavy-duty trucks in various driving conditions, we presented strong evidence that particulate matter could not be adequately predicted from truck speed or acceleration using a functional linear model, whereas the (nonlinear) FGAM provided a much better fit to the data.

5.2 Open Questions and Future Work

Many interesting extensions of FGAM are possible. One that is obvious is extending FGAM to function on function regression, using a model of the form $Y_i(s) = \beta_0 + \int_{\mathcal{T}} F(X_i(t), t, s)dt + \epsilon_i(s)$. The implementation of this model can be done fairly simply using the penalized spline framework we have presented in the dissertation, requiring a third marginal basis for use in a trivariate tensor product and the estimation of a third smoothing parameter. One must make sure that there is enough resolution in the data to accurately estimate such a high dimensional

function and be careful that all model parameters are identifiable. This model is considered briefly in Scheipl et al.[103].

Another interesting application calling for a trivariate F, but with a scalar response, would be to account for interactions between a scalar covariate and a functional predictor; e.g. $Y_i = \beta_0 + \int_{\mathcal{T}} F(X_i(t), t, z_i)dt + \epsilon_i$ for a scalar covariate z_i. Using FGAM would offer increased flexibility for modelling a more complex interaction than the single index structure used in Li et al.[60]. The DTI data considered in Chapter 2 also had a more complex structure than what was considered in the thesis. The complete data set consisted of multiple brain images being taken for each subject, with several months or years between scans. One could attempt extensions to FGAM along the lines of Goldsmith et al.[35] to account for the longitudinal aspect of the data.

An application that is sure to receive more attention in the functional regression literature is that of simultaneously performing smoothing and selection in models with a large number of functional covariates. This problem is of significant interest for time-course microarray data found in genomics (e.g., Wang et al.[120]). It is also considered for the FLM by Lian[61] and for a functional extension of projection pursuit regression in Fan and James[24]. To fit FGAM with a very large numbers of predictors, one could simply assume that each surface has the same amount of smoothness. Alternatively, one could use a hierarchical prior on the smoothing parameters to shrink them to a common value. For more moderate number of predictors, performing variable selection in addition to smoothing is possible for penalized spline models estimated in `mgcv` using Marra and Wood[64]. For fitting to data with increasing numbers of functional predictors and incorporating varying amounts of smoothness for each $F(\cdot, \cdot)$, faster computational methods are neces-

sary. The increased availability of tools for parallel computing, easy porting of `C` code into `R`, and computing on graphics processing units are key developments which will help achieve this.

These computational developments will also be essential for improving the variational Bayes algorithm of Chapter 3 so that it can quickly fit larger data sets and estimate higher numbers of principal components. An important extension of both the VB and MCMC algorithms will be to allow for the handling of generalized responses. It will also be important to investigate coverage for the confidence bands provided by our variational Bayes algorithm and compare with credible intervals from MCMC. Typically, confidence bands derived from VB procedures suffer from undercoverage. Bootstrapping the estimates from our variational Bayes algorithm may be a promising way around this issue (Goldsmith et al.[34]).

There is still much work to be done on our linearity testing problem from Chapter 4. The approach using Type IV Beta priors requires extreme care when computing the required hypergeometric functions, which can be quite difficult to estimate for particular values of the inputs and parameters. More investigation is needed to better understand the quantities R_j^2 and Q_j^2 to prove that the Lauricella hypergeomtric functions always converge and in order to prove model selection consistency (that the Bayes factor goes to zero in probability under H_0). The design assumed for the functional predictors and amount of multicollinearity will no doubt play a large role. As with any time one uses Bayes factors, it will be important to assess the sensitivity of the Bayes factors to changes in the prior hyperparameters as well.

For the restricted likelihood ratio tests, it may be worthwhile to check how the proposed methods perform when σ_1^2, the nuisance variance component correspond-

ing to the FLM term, is near zero. As was mentioned, it has been found that the pseudo-RLRT's performance can suffer in this situation.

APPENDIX A

DERIVATIONS FOR THE VARIATIONAL BAYES ALGORITHM

This appendix details the derivation of the variational Bayes algorithm in Chapter 3. After providing the forms for the full conditionals in Section A.1, we than derive the optimal densities for each parameter in Section A.2. Next, we provide a approximate lower bound for the log-likelihood in Section A.3. The full algorithm is then provided in Section A.4.

A.1 Derivation of Full Conditional Distributions

In this section we derive the full conditional distributions for the variance components and spline coefficients in (3.2) and also give the full posterior distribution.

Variance parameters

We begin by defining the $N \times d_x d_t$ matrix $\mathbb{Z}_0$ whose ith row is given by $\mathbf{Z}_{0,i}^T = \mathbf{L}^T \mathbb{B}_{\xi_i} \mathbb{T}_0 = \mathbf{b}_{\boldsymbol{\xi}_i}^T \mathbb{T}_0$ and the $N \times (K_x K_t - d_x d_t)$ matrix $\mathbb{Z}_p$ with ith row given by $\mathbf{Z}_{p,i}^T = \mathbf{b}_{\boldsymbol{\xi}_i}^T \mathbb{T}_p$. We also define $\mathbf{y}_{\eta_0} = (y_1 - \eta_{0,1}, \ldots, y_N - \eta_{0,N})^T$, $\boldsymbol{\eta}_1 = \mathbb{Z}_0 \boldsymbol{\beta} + \mathbb{Z}_p \boldsymbol{\delta}$ with ith component $\eta_{1,i} = \mathbf{b}_{\boldsymbol{\xi}_i}^T \{\mathbb{T}(\boldsymbol{\beta}^T, \boldsymbol{\delta}^T)^T\} = \mathbf{b}_{\boldsymbol{\xi}_i}^T \mathbb{T}_0 \boldsymbol{\beta} + \mathbf{b}_{\boldsymbol{\xi}_i}^T \mathbb{T}_p \boldsymbol{\delta}$,, and
$\boldsymbol{\Xi} = [\boldsymbol{\xi}_1 : \cdots : \boldsymbol{\xi}_N]^T$, we have

$$
\begin{aligned}
p(\sigma^2|\cdot) &\propto p(\mathbf{y}|\eta_{0,1}, \ldots, \eta_{0,N}, \boldsymbol{\beta}, \boldsymbol{\delta}, \sigma^2, \boldsymbol{\Xi}) p(\sigma^2) \\
&\propto (\sigma^2)^{-N/2 - a_s - 1} \exp\left\{ - \frac{b_s + \frac{1}{2} \left(\mathbf{y}_{\eta_0} - \boldsymbol{\eta}_1\right)^T \left(\mathbf{y}_{\eta_0} - \boldsymbol{\eta}_1\right)}{\sigma^2} \right\} \\
\text{so that } \sigma^2|\cdot &\sim \text{IG}\left(a = N/2 + a_s, b = b_s + \frac{1}{2} \{\mathbf{y}_{\eta_0} - \boldsymbol{\eta}_1\}^T \{\mathbf{y}_{\eta_0} - \boldsymbol{\eta}_1\} \right).
\end{aligned}
$$

Similarly,

$$\sigma_x^2|\cdot \sim \text{IG}\left(a = \sum_i^N n_i/2 + a_x, b = b_x + \frac{1}{2}\sum_i^N \sum_j^{n_i} \left\{\widetilde{x}_{ij} - \mu_x(t_{ij}) - \sum_m^M \phi_m(t_{ij})\xi_{im}\right\}^2\right).$$

Spline coefficients $\beta, \boldsymbol{\delta}$

$$p(\boldsymbol{\beta}, \boldsymbol{\delta}|\cdot) \propto p(\boldsymbol{\beta})p(\boldsymbol{\delta}|\lambda_x, \lambda_t)p(\lambda_x)p(\lambda_t) \propto \exp\left\{-\frac{1}{2}\boldsymbol{\delta}^T(\lambda_x\boldsymbol{\Psi}_x + \lambda_t\boldsymbol{\Psi}_t)\boldsymbol{\delta}\right\}$$
$$\times \exp\left\{-\frac{(\mathbf{y} - \boldsymbol{\eta}_0 - \mathbb{Z}_0\boldsymbol{\beta} - \mathbb{Z}_p\boldsymbol{\delta})^T(\mathbf{y} - \boldsymbol{\eta}_0 - \mathbb{Z}_0\boldsymbol{\beta} - \mathbb{Z}_p\boldsymbol{\delta})}{2\sigma^2}\right\}.$$

In other words,

$\boldsymbol{\delta}|\cdot \sim N(\mathbf{m}_b, \mathbb{S}_b)$ where

$\mathbb{S}_b = (\mathbb{Z}_p^T\mathbb{Z}_p/\sigma^2 + \lambda_x\boldsymbol{\Psi}_x + \lambda_t\boldsymbol{\Psi}_t)^{-1}$, and $\mathbf{m}_b = \mathbb{S}_b\mathbb{Z}_p^T\left(\mathbf{y}_{\eta_0} - \mathbb{Z}_0\boldsymbol{\beta}\right)/\sigma^2$;

$\boldsymbol{\beta}|\cdot \sim N(\mathbf{m}_\beta, \mathbb{S}_\beta)$ where

$\mathbb{S}_\beta = (\mathbb{Z}_0^T\mathbb{Z}_0/\sigma^2)^{-1}$, and $\mathbf{m}_\beta = \mathbb{S}_\beta\mathbb{Z}_0^T\left(\mathbf{y}_{\eta_0} - \mathbb{Z}_p\boldsymbol{\delta}\right)/\sigma^2$.

The full posterior distribution is given by

$$
\begin{aligned}
&p(\boldsymbol{\beta}, \boldsymbol{\delta}, \sigma^2, \lambda_x, \lambda_t, \boldsymbol{\Xi}, \sigma_x^2 | \mathbf{y}, \tilde{\mathbf{x}}, \eta_{0,1}, \ldots, \eta_{0,N}, \boldsymbol{\mu}_x, \boldsymbol{\Phi}, \boldsymbol{\nu}) \propto \\
&\quad \propto (\sigma^2)^{-N/2} \\
&\quad \times \exp\left[-\frac{1}{2\sigma^2} \sum_i^N \left\{ y_{\eta_0,i} - \sum_{j=1}^{K_x} \sum_{k=1}^{K_t} \mathbf{L}^T \left[\mathbf{B}_j^{\mathcal{X}} \left(\boldsymbol{\mu}_x + \boldsymbol{\Phi} \boldsymbol{\xi}_i \right) \cdot \mathbf{B}_k^{\mathcal{T}}(\mathbf{t}) \right] \left[\mathbb{T} (\boldsymbol{\beta}^T, \boldsymbol{\delta}^T)^T \right]_{j,k} \right\}^2 \right] \\
&\quad \times (\sigma_x^2)^{-\sum_i^N n_i/2} \exp\left[-\frac{1}{2\sigma_x^2} \sum_i^N \| \tilde{\mathbf{x}}_i - \mu_x(\mathbf{t}_i) - \boldsymbol{\Phi}(\mathbf{t}_i) \boldsymbol{\xi}_i \|_2^2 \right] \\
&\quad \times |\lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t|^{1/2} \exp\left(-\frac{1}{2} \boldsymbol{\delta}^T (\lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t) \boldsymbol{\delta} \right) \times \\
&\quad \times \exp\left(-\frac{1}{2} \sum_i^N \boldsymbol{\xi}_i^T \operatorname{diag}(\boldsymbol{\nu}^{-1}) \boldsymbol{\xi}_i \right) \cdot (\sigma^2)^{-a_s - 1} \exp(-b_s/\sigma^2) \cdot (\sigma_x^2)^{-a_x - 1} \exp(-b_x/\sigma_x^2) \\
&\quad \times \cdot (\lambda_x)^{a_l + 1} \exp(-b_l \lambda_x) (\lambda_t)^{a_l + 1} \exp(-b_l \lambda_t),
\end{aligned}
$$

where $[\mathbb{A}]_{j,k}$ denotes the entry in the jth row and kth column of the matrix $\mathbb{A}$.

A.2 Derivation of Optimal Proposal Densities

In this section we derive the optimal densities, q^*, for parameters that were given conjugate priors and give detailed calculations for our Laplace approximation to the optimal density for the principal component scores. We use the notation and full conditionals from Appendix A and often make use of the results that for $\mathbf{x} \sim (\boldsymbol{\mu}, \boldsymbol{\Sigma})$, $\mathrm{E}[\mathbf{x}^T \mathbf{S} \mathbf{x}] = \operatorname{tr}(\mathbf{S}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^T \mathbf{S} \boldsymbol{\mu}$ and $\mathrm{E}[\mathbf{x}\mathbf{x}^T] = \mathrm{E}[\mathbf{x}]\,\mathrm{E}[\mathbf{x}]^T + \mathrm{Var}[\mathbf{x}]$.

We first discuss the updates for the offset terms, η_{0i}, $i = 1, \ldots, N$. For simplicity, we assume that they can be expressed as $\eta_{0i} = \mathbf{u}_i^T \boldsymbol{\eta}_0$ or $(\eta_{01}, \ldots, \eta_{0N})^T = \mathbb{U} \boldsymbol{\eta}_0$, where $\mathbb{U}$ is an $N \times p_0$ matrix with rows $\mathbf{u}_i^T$ containing, for e.g., scalar covariate observations for parametric terms, basis function evaluations for nonparametric

terms, or a leading column of ones for an intercept. Further generalizations are straightforward. The coefficient vector $\boldsymbol{\eta}_0$ has prior density $p(\boldsymbol{\eta}_0) = N(\mathbf{0}, \sigma^2_{\boldsymbol{\eta}_0}\mathbb{I}_{p_0})$, with $\sigma^2_{\boldsymbol{\eta}_0}$ large and fixed. The full conditional is given by

$$\begin{aligned} p(\boldsymbol{\eta}_0|\text{rest}) &\propto p(\mathbf{y}|\boldsymbol{\eta}_0, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\Xi}, \sigma^2)p(\boldsymbol{\eta}_0) \\ &\propto \exp\left[-\frac{(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1)^T(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1)}{2\sigma^2} - \frac{1}{\sigma^2_{\boldsymbol{\eta}_0}}\boldsymbol{\eta}_0^T\mathbb{I}_{p_0}\boldsymbol{\eta}_0\right] \\ &\propto \exp\left\{-\frac{1}{2}\left[\boldsymbol{\eta}_0^T\left(\frac{1}{\sigma^2}\mathbb{U}^T\mathbb{U} + \frac{1}{\sigma^2_{\boldsymbol{\eta}_0}}\mathbb{I}_{p_0}\right)\boldsymbol{\eta}_0 - 2\left((\mathbf{y} - \boldsymbol{\eta}_1)^T\mathbb{U}/\sigma^2\right)\boldsymbol{\eta}_0\right]\right\}. \end{aligned}$$

Thus,

$$\begin{aligned} q^*(\boldsymbol{\eta}_0) &\propto \exp\left\{-\frac{1}{2}\,\mathrm{E}_{-\boldsymbol{\eta}_0}\left[\boldsymbol{\eta}_0^T\left(\frac{1}{\sigma^2}\mathbb{U}^T\mathbb{U} + \frac{1}{\sigma^2_{\boldsymbol{\eta}_0}}\mathbb{I}_{p_0}\right)\boldsymbol{\eta}_0 - 2\left((\mathbf{y} - \boldsymbol{\eta}_1)^T\mathbb{U}/\sigma^2\right)\boldsymbol{\eta}_0\right]\right\} \\ &\propto \exp\left\{-\frac{1}{2}\left[\boldsymbol{\eta}_0^T\left(\mu_{q(1/\sigma^2)}\mathbb{U}^T\mathbb{U} + \frac{1}{\sigma^2_{\boldsymbol{\eta}_0}}\mathbb{I}_{p_0}\right)\boldsymbol{\eta}_0 - 2\left((\mathbf{y} - \mu_{q(\boldsymbol{\eta}_1)})^T\mathbb{U}\mu_{q(1/\sigma^2)}\right)\boldsymbol{\eta}_0\right]\right\}, \end{aligned}$$

where $\mu_{q(\boldsymbol{\eta}_1)} = \mu_{q(\mathbf{b}_{\boldsymbol{\xi}})}\mathbb{T}(\mu_{q(\boldsymbol{\beta})}{}^T, \mu_{q(\boldsymbol{\delta})}{}^T)^T$. Denote the rows of the $N \times K_xK_t$ matrix, $\mu_{q(\mathbf{b}_{\boldsymbol{\xi}})}$, by $\mu^T_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}$. By completing the square, we see $q^*(\boldsymbol{\eta}_0) = N(\mu_{q(\boldsymbol{\eta}_0)}, \boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)})$ where $\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)} = \left(\mu_{q(1/\sigma^2)}\mathbb{U}^T\mathbb{U} + \frac{1}{\sigma^2_{\boldsymbol{\eta}_0}}\mathbb{I}_{p_0}\right)^{-1}$ and $\mu_{q(\boldsymbol{\eta}_0)} = \boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)}\mathbb{U}^T(\mathbf{y} - \mu_{q(\boldsymbol{\eta}_1)})\mu_{q(1/\sigma^2)}$.

Next, for $\boldsymbol{\beta}$

$$\begin{aligned} p(\boldsymbol{\beta}|\text{rest}) &\propto p(\mathbf{y}|\boldsymbol{\eta}_0, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\Xi}, \sigma^2)p(\boldsymbol{\beta}) \\ &\propto \exp\left[-\frac{(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1)^T(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1)}{2\sigma^2} - \frac{1}{\sigma^2_{\boldsymbol{\beta}}}\boldsymbol{\beta}^T\mathbb{I}_{d_xd_t}\boldsymbol{\beta}\right] \\ &\propto \exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}^T\left(\frac{1}{\sigma^2}\mathbb{Z}_0^T\mathbb{Z}_0 + \frac{1}{\sigma^2_{\boldsymbol{\beta}}}\mathbb{I}_{d_xd_t}\right)\boldsymbol{\beta} - 2\left((\mathbf{y} - \mathbb{U}\eta_0 - \mathbb{Z}_p\boldsymbol{\delta})^T\mathbb{Z}_0/\sigma^2\right)\boldsymbol{\beta}\right]\right\} \end{aligned}$$

Thus,

$$\begin{aligned} q^*(\boldsymbol{\beta}) &\propto \exp\left\{-\frac{1}{2}\left[\boldsymbol{\beta}^T\left(\mu_{q(1/\sigma^2)}\,\mathrm{E}_{-\boldsymbol{\beta}}[\mathbb{Z}_0^T\mathbb{Z}_0] + \frac{1}{\sigma^2_{\boldsymbol{\beta}}}\mathbb{I}_{d_xd_t}\right)\boldsymbol{\beta}\right]\right\}\cdot \\ &\quad\times \exp\left\{-\frac{\mu_{q(1/\sigma^2)}}{2}\left[(\mathbf{y} - \mathbb{U}\mu_{\mathbf{q}(\boldsymbol{\eta}_0)})^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}})}\mathbb{T}_0 - \mu_{q(\boldsymbol{\delta})}{}^T\,\mathrm{E}_{-\boldsymbol{\beta}}(\mathbb{Z}_p^T\mathbb{Z}_0)\right]\boldsymbol{\beta}\right\}, \end{aligned}$$

where

$$\mathrm{E}(\mathbb{Z}_j^T\mathbb{Z}_k) = \mathrm{E}\left[\sum_{i=1}^{N}(\mathbb{T}_j^T\mathbf{b}_{\boldsymbol{\xi}_i})(\mathbf{b}_{\boldsymbol{\xi}_i}^T\mathbb{T}_k)\right] = \mathbb{T}_j^T\left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T)\right]\mathbb{T}_k, \qquad j,k=0,p.$$

Thus, $q^*(\boldsymbol{\beta}) = N(\mu_{q(\boldsymbol{\beta})}, \boldsymbol{\Sigma}_{q(\boldsymbol{\beta})})$ with

$$\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})} = \left\{\mathbb{T}_0^T\left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}\left(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\right)\right]\mathbb{T}_0\mu_{q(1/\sigma^2)} + \frac{1}{\sigma_\beta^2}\mathbb{I}_{d_x d_t}\right\}^{-1}$$

$$\mu_{q(\boldsymbol{\beta})} = \boldsymbol{\Sigma}_{q(\boldsymbol{\beta})}\mathbb{T}_0^T\left\{\mu_{q(\mathbf{b}_{\boldsymbol{\xi}})}^T(\mathbf{y} - \mathbb{U}\mu_{\mathbf{q}(\boldsymbol{\eta}_0)}) - \left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}\left(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\right)\right]\mathbb{T}_p\mu_{q(\boldsymbol{\delta})}\right\}\mu_{q(1/\sigma^2)}.$$

The derivation for $\boldsymbol{\delta}$ is analogous and given by $q^*(\boldsymbol{\delta}) = N(\mu_{q(\boldsymbol{\delta})}, \boldsymbol{\Sigma}_{q(\boldsymbol{\delta})})$ with

$$\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})} = \left\{\mathbb{T}_p^T\left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}\left(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\right)\right]\mathbb{T}_p\mu_{q(1/\sigma^2)} + \mu_{q(\lambda_x)}\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right\}^{-1}$$

$$\mu_{q(\boldsymbol{\delta})} = \boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}\mathbb{T}_p^T\left\{\mu_{q(\mathbf{b}_{\boldsymbol{\xi}})}^T(\mathbf{y} - \mathbb{U}\mu_{\mathbf{q}(\boldsymbol{\eta}_0)}) - \left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}\left(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\right)\right]\mathbb{T}_0\mu_{q(\boldsymbol{\beta})}\right\}\mu_{q(1/\sigma^2)}.$$

For σ_x^2, we have,

$$\sigma_x^2|\cdot \sim \mathrm{IG}\left(\sum_{i=1}^{N} n_i/2 + a_x,\ b_x + \frac{1}{2}\sum_{i=1}^{N}||\tilde{\mathbf{x}}_i - \mu_x(\mathbf{t}_i) - \Phi(\mathbf{t}_i)\boldsymbol{\xi}_i||^2\right)$$

so that

$$q^*(\sigma_x^2) \propto \exp\left\{-\left(a_x + \sum_{i=1}^{N} n_i/2 - 1\right)\log(\sigma_x^2) - \frac{1}{\sigma_x^2}\left[b_x + \frac{1}{2}\,\mathrm{E}_{-\sigma_x^2}\left(\sum_{i=1}^{N}||\tilde{\mathbf{x}}_i - \mu_x(\mathbf{t}_i) - \Phi(\mathbf{t}_i)\boldsymbol{\xi}_i||_2^2\right)\right]\right\}.$$

Therefore, $q^*(\sigma_x^2) = \mathrm{IG}(a_x + \sum_{i=1}^{N} n_i/2,\ B_{q(\sigma_x^2)})$, where

$$B_{q(\sigma_x^2)} = b_x + \frac{1}{2}\sum_{i=1}^{N}\left[||\tilde{\mathbf{x}}_i - \mu_x(\mathbf{t}_i) - \Phi(\mathbf{t}_i)\mu_{q(\boldsymbol{\xi}_i)}||_2^2 + \mathrm{tr}\left(\Phi(\mathbf{t}_i)^T\Phi(\mathbf{t}_i)\boldsymbol{\Sigma}_{q(\boldsymbol{\xi}_i)}\right)\right]$$

Note that for $\theta = \mathrm{IG}(A, B),\ \mu_\theta(1/\theta) = A/B$.

Similarly,

$$p(\sigma^2|\cdot) \propto (\sigma^2)^{-N/2-a_s-1}\exp\left(-\frac{b_s + \frac{1}{2}\left(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1\right)^T\left(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1\right)}{\sigma^2}\right)$$

$$\text{so that } \sigma^2|\cdot \sim \mathrm{IG}\left(a = N/2 + a_s,\ b = b_s + \frac{1}{2}\left(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1\right)^T\left(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1\right)\right)$$

Thus,

$$q^*(\sigma^2) \propto \exp\left\{-(a_s + N/2 - 1)\log(\sigma^2)\right\} \times \left\{-\frac{1}{\sigma^2}\left[b_s + \frac{1}{2}\,\mathrm{E}_{-\sigma^2}\left(||\left(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1\right)||_2^2\right)\right]\right\}.$$

$$\mathrm{E}_{-\sigma^2}\left[||\left(\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1\right)||_2^2\right] = \mathrm{E}_{-\sigma^2}\left[||(\mathbf{y} - \mathbb{U}\mu_{q(\boldsymbol{\eta}_0)} - \mu_{q(\boldsymbol{\eta}_1)})||_2^2\right] + \mathrm{E}_{-\sigma^2}\left[||\mathbb{U}\boldsymbol{\eta}_0 - \mathbb{U}\mu_{q(\boldsymbol{\eta}_0)}||_2^2\right] + \mathrm{E}_{-\sigma^2}\left[||\boldsymbol{\eta}_1 - \mu_{q(\boldsymbol{\eta}_1)}||_2^2\right].$$

Now $\mathrm{E}_{-\sigma^2}\left[||\mathbb{U}\boldsymbol{\eta}_0 - \mathbb{U}\mu_{q(\boldsymbol{\eta}_0)}||_2^2\right] = \mathrm{tr}\left(\mathbb{U}^T\mathbb{U}\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)}\right)$ and for the third term on the RHS we have

$$\mathrm{E}_{-\sigma^2}||\boldsymbol{\eta}_1 - \mu_{q(\boldsymbol{\eta}_1)}||_2^2] = \mathrm{E}_{-\sigma^2}\left[\sum_{i=1}^{N}(\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\theta} - \mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}{}^T\mu_{q(\boldsymbol{\theta})})^2\right] = \mathrm{E}_{-\sigma^2}\left[\sum_{i=1}^{N}\boldsymbol{\theta}^T\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\theta}\right]$$
$$- \mu_{q(\boldsymbol{\theta})}{}^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}{}^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}\mu_{q(\boldsymbol{\theta})} = \mathrm{E}_{-\boldsymbol{\xi}_i}\left[\mathrm{tr}\left(\sum_{i=1}^{N}\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\right)\right.$$
$$\left. + \mu_{q(\boldsymbol{\theta})}{}^T\sum_{i=1}^{N}\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})}\right] - \mu_{q(\boldsymbol{\theta})}{}^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}{}^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}\mu_{q(\boldsymbol{\theta})} = \mathrm{tr}\left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}\left(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\right)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\right]$$
$$+ \mu_{q(\boldsymbol{\theta})}{}^T\left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}\left(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\right)\right]\mu_{q(\boldsymbol{\theta})} - \mu_{q(\boldsymbol{\theta})}{}^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}{}^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}\mu_{q(\boldsymbol{\theta})},$$

where, as before, $\boldsymbol{\theta} = \mathbb{T}(\boldsymbol{\beta}^T, \boldsymbol{\delta}^T)^T$.

Therefore, we have, $q^*(\sigma^2) = \mathrm{IG}(a_s + N/2,\ B_{q(\sigma^2)})$, where $B_{q(\sigma^2)}$ is given by

$$b_s + \frac{1}{2}||(\mathbf{y} - \mathbb{U}\mu_{q(\boldsymbol{\eta}_0)} - \mu_{q(\boldsymbol{\eta}_1)})||_2^2 + \frac{1}{2}\,\mathrm{tr}\left(\mathbb{U}^T\mathbb{U}\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)}\right) + \frac{1}{2}\,\mathrm{tr}\left\{\left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}\left(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\right)\right]\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\right\}$$
$$+ \frac{1}{2}\mu_{q(\boldsymbol{\theta})}{}^T\left[\sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\xi}}\left(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\right)\right]\mu_{q(\boldsymbol{\theta})} - \frac{1}{2}\mu_{q(\boldsymbol{\theta})}{}^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}{}^T\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}\mu_{q(\boldsymbol{\theta})}.$$

Laplace Approx. for Optimal Density for PC Scores

First, defining some notation, the derivatives of the matrix valued function $\mathbf{M} : \mathbb{R}^p \to \mathbb{R}^{m\times n}$ with respect to v_i, $\mathbf{v}^T = (v_1, \ldots, v_p)$ and $\mathbf{v}$ are

$$\mathcal{D}_{v_i}\mathbf{M}(\mathbf{v}) \equiv \begin{bmatrix} \frac{\partial m_{11}}{v_i} & \ldots & \frac{\partial m_{1n}}{v_i} \\ \vdots & \ddots & \vdots \\ \frac{\partial m_{m1}}{v_i} & \ldots & \frac{\partial m_{mn}}{v_i} \end{bmatrix},$$

$$\mathcal{D}_{\mathbf{v}^T}\mathbf{M} \equiv \left[\mathcal{D}_{v_1}\mathbf{M}|\cdots|\mathcal{D}_{v_p}\mathbf{M}\right], \qquad \mathcal{D}_{\mathbf{v}}\mathbf{M} \equiv \begin{bmatrix} \mathcal{D}_{v_1}\mathbf{M} \\ \vdots \\ \mathcal{D}_{v_p}\mathbf{M} \end{bmatrix},$$

respectively. Also define $\mathcal{D}_{\mathbf{v}^2}\mathbf{M} \equiv \mathcal{D}_{\mathbf{v}}(\mathcal{D}_{\mathbf{v}}\mathbf{M})$. We first differentiate the components of $\log q(\boldsymbol{\xi}_i)$ with respect to $\boldsymbol{\xi}_i$

$$\begin{aligned}\mathcal{D}_{\mathbf{b}_{\boldsymbol{\xi}_i}} \operatorname{tr}(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}) &= \mathcal{D}_{\mathbf{b}_{\boldsymbol{\xi}_i}} \operatorname{tr}(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}) = \mathcal{D}_{\mathbf{b}_{\boldsymbol{\xi}_i}} \operatorname{tr}(\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i}) \\ &= \mathcal{D}_{\mathbf{b}_{\boldsymbol{\xi}_i}}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i} = 2\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i}\end{aligned}$$

We then have (See, Vetter [116]),

$$\mathcal{D}_{\boldsymbol{\xi}} \operatorname{tr}(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}) = \mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mathcal{D}_{\mathbf{b}_{\boldsymbol{\xi}_i}} \operatorname{tr}(\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}) = 2\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i}.$$

$$\mathcal{D}_{\boldsymbol{\xi}_i}(y_i - \mathbf{u}_i^T\mu_{q(\boldsymbol{\eta}_0)} - \mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})})^2 = -2\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}(y_i - \mathbf{u}_i^T\mu_{q(\boldsymbol{\eta}_0)} - \mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})})$$

$$\mathcal{D}_{\boldsymbol{\xi}_i}\left\{\boldsymbol{\xi}_i^T[\mu_{q(1/\sigma_x^2)}\boldsymbol{\Phi}(\mathbf{t}_i)^T\boldsymbol{\Phi}(\mathbf{t}_i) + \operatorname{diag}(\boldsymbol{\nu}^{-1})]\boldsymbol{\xi}_i\right\} = 2[\mu_{q(1/\sigma_x^2)}\boldsymbol{\Phi}(\mathbf{t}_i)^T\boldsymbol{\Phi}(\mathbf{t}_i)+\operatorname{diag}(\boldsymbol{\nu}^{-1})]\boldsymbol{\xi}_i$$

We arrive at

$$\begin{aligned}\mathcal{D}_{\boldsymbol{\xi}_i}\log q(\boldsymbol{\xi}_i) &= \mu_{q(1/\sigma^2)}\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}(y_i - \mathbf{u}_i^T\mu_{q(\boldsymbol{\eta}_0)} - \mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})}) \\ &\quad + \mu_{q(1/\sigma_x^2)}(\tilde{\mathbf{x}}_i - \boldsymbol{\mu}_x(\mathbf{t}_i))^T\boldsymbol{\Phi}(\mathbf{t}_i) - [\mu_{q(1/\sigma_x^2)}\boldsymbol{\Phi}(\mathbf{t}_i)^T\boldsymbol{\Phi}(\mathbf{t}_i) + \operatorname{diag}(\boldsymbol{\nu}^{-1})]\boldsymbol{\xi}_i \\ &\quad - \mu_{q(1/\sigma^2)}\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i}\end{aligned}$$

Now to compute $\mathcal{D}_{\boldsymbol{\xi}_i^2} \log q(\boldsymbol{\xi}_i)$:

$$\begin{aligned}
&\mathcal{D}_{\boldsymbol{\xi}_i}\left[\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}(y_i - \mathbf{u}_i^T\mu_{q(\boldsymbol{\eta}_0)} - \mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})})\right] \\
&\quad = \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}(y_i - \mathbf{u}_i^T\mu_{q(\boldsymbol{\eta}_0)} - \mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})}) - [\mathbb{I}_M \otimes \mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}]\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})} \\
&\quad = \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}(y_i - \mathbf{u}_i^T\mu_{q(\boldsymbol{\eta}_0)} - \mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})}) - \mathrm{vec}\left\{\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}[\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}]^T\right\}
\end{aligned}$$

$$\mathcal{D}_{\boldsymbol{\xi}_i}\left\{[\mu_{q(1/\sigma_x^2)}\boldsymbol{\Phi}(\mathbf{t}_i)^T\boldsymbol{\Phi}(\mathbf{t}_i) + \mathrm{diag}(\boldsymbol{\nu}^{-1})]\boldsymbol{\xi}_i\right\} = \mathrm{vec}[\mu_{q(1/\sigma_x^2)}\boldsymbol{\Phi}(\mathbf{t}_i)^T\boldsymbol{\Phi}(\mathbf{t}_i) + \mathrm{diag}(\boldsymbol{\nu}^{-1})]$$

$$\begin{aligned}
\mathcal{D}_{\boldsymbol{\xi}_i}\left[\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i}\right] &= \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i} + [\mathbb{I}_M \otimes \mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)](\mathbb{I}_M \otimes \boldsymbol{\Sigma}_{q(\boldsymbol{\theta})})\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}) \\
&= \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i} + [\mathbb{I}_M \otimes \mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}]\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}) \\
&= \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i} + [\mathbb{I}_M \otimes \mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}]\mathrm{vec}\left[\mathcal{D}_{\boldsymbol{\xi}_i}^T(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\right] \\
&= \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i} + \mathrm{vec}\left[\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathcal{D}_{\boldsymbol{\xi}_i}^T(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\right],
\end{aligned}$$

where $\otimes$ denotes the Kronecker product and the last equality follows from, e.g., Vetter[116], Eq. (9). Thus, we have

$$\begin{aligned}
\mathcal{D}_{\boldsymbol{\xi}_i^2} \log q(\boldsymbol{\xi}_i) &= \mu_{q(1/\sigma^2)}\mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}(y_i - \mathbf{u}_i^T\mu_{q(\boldsymbol{\eta}_0)} - \mathbf{b}_{\boldsymbol{\xi}_i}^T\mu_{q(\boldsymbol{\theta})}) \\
&\quad - \mu_{q(1/\sigma^2)}\mathrm{vec}\left\{\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}[\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\mu_{q(\boldsymbol{\theta})}]^T\right\} \\
&\quad - \mathrm{vec}[\mu_{q(1/\sigma_x^2)}\boldsymbol{\Phi}(\mathbf{t}_i)^T\boldsymbol{\Phi}(\mathbf{t}_i) + \mathrm{diag}(\boldsymbol{\nu}^{-1})] \\
&\quad - \mu_{q(1/\sigma^2)}\left\{\mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathbf{b}_{\boldsymbol{\xi}_i} + \mathrm{vec}\left[\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\boldsymbol{\Sigma}_{q(\boldsymbol{\theta})}\mathcal{D}_{\boldsymbol{\xi}_i}^T(\mathbf{b}_{\boldsymbol{\xi}_i}^T)\right]\right\} \qquad \text{(A.1)}
\end{aligned}$$

Next to derive expressions for $\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)$ and $\mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T)$. Let $\mathbf{c}(\boldsymbol{\xi}_i) = \boldsymbol{\mu}_x + \boldsymbol{\Phi}\boldsymbol{\xi}_i$ and let $\mathbb{B}'_{\boldsymbol{\xi}_i}$ be the $T \times K_x K_t$ matrix of derivatives of the tensor product B-splines evaluated at $\mathbf{c}(\boldsymbol{\xi}_i)$ with jth row denoted by $(\mathbf{B}')_{j,i}^T$. Similarly, define $\mathbb{B}''_{\boldsymbol{\xi}_i}$, then

$$\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T) = \mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{c}^T)\mathcal{D}_{\mathbf{c}}\mathbf{b}_{\boldsymbol{\xi}_i}^T = \mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{c}^T)\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{L}^T\mathbf{B}_{\xi_i}) = \boldsymbol{\Phi}^T\mathbb{B}'_{\boldsymbol{\xi}_i} \odot (\mathbf{L} \otimes \mathbf{1}_{K_x K_t}^T)$$

and

$$\begin{aligned}
\mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T) &= [\mathbb{I}_M \otimes \boldsymbol{\Phi}^T]\mathcal{D}_{\boldsymbol{\xi}_i}[\mathbb{B}'_{\boldsymbol{\xi}_i} \odot (\mathbf{L} \otimes \mathbf{1}_{K_xK_t}^T)] \\
&= (\mathbb{I}_M \otimes \boldsymbol{\Phi}^T)(\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{c}^T) \otimes \mathbb{I}_T)\mathcal{D}_{\mathbf{c}}[\mathbb{B}'_{\boldsymbol{\xi}_i} \odot (\mathbf{L} \otimes \mathbf{1}_{K_xK_t}^T)] \\
&= (\mathbb{I}_M \otimes \boldsymbol{\Phi}^T)(\boldsymbol{\Phi}^T \otimes \mathbb{I}_T)\begin{pmatrix} \ell_1 \cdot (\mathbf{B}'')_{1,i}^T \\ \mathbf{0}_{T\times K_xK_t} \\ \ell_2 \cdot (\mathbf{B}'')_{2,i}^T \\ \vdots \\ \ell_T \cdot (\mathbf{B}'')_{T,i}^T \end{pmatrix} = (\boldsymbol{\Phi}^T \otimes \boldsymbol{\Phi}^T)\begin{pmatrix} \ell_1 \cdot (\mathbf{B}'')_{1,i}^T \\ \mathbf{0}_{T\times K_xK_t} \\ \ell_2 \cdot (\mathbf{B}'')_{2,i}^T \\ \vdots \\ \ell_T \cdot (\mathbf{B}'')_{T,i}^T \end{pmatrix},
\end{aligned}$$

where $\mathbf{0}_{m\times n}$ denotes a $m \times n$ matrix with every entry equal to 0. Thus, we arrive at our Laplace approximation 3.6.

Next, we compute the expectations with respect to $\boldsymbol{\xi}_i$ involving $\mathbf{b}_{\boldsymbol{\xi}_i}$. We use a second order matrix Taylor expansion about $\boldsymbol{\xi}_{i,0}$. Let $\tilde{\boldsymbol{\xi}}_i = \boldsymbol{\xi}_i - \boldsymbol{\xi}_{i,0}$, we have

$$\mathbf{b}_{\boldsymbol{\xi}_i} \approx \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}) + \mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})]\tilde{\boldsymbol{\xi}}_i + \frac{1}{2}\mathcal{D}_{\boldsymbol{\xi}_i^{T^2}}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})](\tilde{\boldsymbol{\xi}}_i \otimes \tilde{\boldsymbol{\xi}}_i)$$

where $\mathcal{D}^2_{\xi_i^{T^2}}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})] \equiv \mathcal{D}_{\xi_i^T}\{\mathcal{D}_{\xi_i^T}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})]\}$ with dimension $K_xK_t \times M^2$ (see, **?**)]vetter1973matrix. Therefore, we have

$$\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})} \approx \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}) + \frac{1}{2}\mathcal{D}_{\boldsymbol{\xi}_i^{T^2}}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})]\mathrm{vec}(\boldsymbol{\Lambda}) = \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}) + \frac{1}{2}\left\{\mathcal{D}_{\boldsymbol{\xi}_i^2}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})]\right\}^T \mathrm{vec}(\boldsymbol{\Lambda})$$

and

$$\begin{aligned}
\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T \approx &\left\{\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}) + \mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})]\tilde{\boldsymbol{\xi}}_i + \frac{1}{2}\mathcal{D}_{\boldsymbol{\xi}_i^{T^2}}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})](\tilde{\boldsymbol{\xi}}_i \otimes \tilde{\boldsymbol{\xi}}_i)\right\} \\
&\times \left\{\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}) + \mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})]\tilde{\boldsymbol{\xi}}_i + \frac{1}{2}\mathcal{D}_{\boldsymbol{\xi}_i^{T^2}}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})](\tilde{\boldsymbol{\xi}}_i \otimes \tilde{\boldsymbol{\xi}}_i)\right\}^T \\
= &\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0}) + \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})\tilde{\boldsymbol{\xi}}_i^T\mathcal{D}_{\boldsymbol{\xi}_i}^T[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})] + \mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})]\tilde{\boldsymbol{\xi}}_i\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0}) \\
&+ \frac{1}{2}\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})(\tilde{\boldsymbol{\xi}}_i^T \otimes \tilde{\boldsymbol{\xi}}_i^T)\mathcal{D}_{\boldsymbol{\xi}_i^2}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})] + \frac{1}{2}\mathcal{D}_{\boldsymbol{\xi}_i^{T^2}}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})](\tilde{\boldsymbol{\xi}}_i \otimes \tilde{\boldsymbol{\xi}}_i)\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0}) \\
&+ \mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})]\tilde{\boldsymbol{\xi}}_i\tilde{\boldsymbol{\xi}}_i^T\mathcal{D}_{\boldsymbol{\xi}_i^T}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})] + o(||\tilde{\boldsymbol{\xi}}_i||^2)
\end{aligned}$$

so that

$$\begin{aligned}
\mathrm{E}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}\mathbf{b}_{\boldsymbol{\xi}_i}^T] &\approx \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0}) + \frac{1}{2}\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})\mathrm{vec}(\boldsymbol{\Lambda})^T\mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})) \\
&\quad + \frac{1}{2}\mathcal{D}_{\boldsymbol{\xi}_i^{T^2}}(\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}))\mathrm{vec}(\boldsymbol{\Lambda})\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0}) + \mathcal{D}_{\boldsymbol{\xi}_i^T}[\mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})]\boldsymbol{\Lambda}\mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})] \\
&= \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0}) + \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0})\mathrm{vec}(\boldsymbol{\Lambda})^T\mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})) \\
&\quad + \left\{\mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})]\right\}^T\boldsymbol{\Lambda}\mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})]
\end{aligned}$$

A.3 Derivation of Log-Likelihood Lower Bound

For any density, q^*, a lower bound on our log-likelihood can be derived using Kullbeck-Leibler divergence and is given by $\log[p(\mathbf{y},\tilde{\mathbf{x}};\boldsymbol{\Theta})] \geq \log[\underline{p}(\mathbf{y},\tilde{\mathbf{x}};q)] := \int q^*(\boldsymbol{\Theta})\log\left(\frac{p(\mathbf{y},\tilde{\mathbf{x}},\boldsymbol{\Theta})}{q^*(\boldsymbol{\Theta})}\right)d\boldsymbol{\Theta} =$
$\mathrm{E}_{q^*}\{\log[p(\mathbf{y},\tilde{\mathbf{x}},\boldsymbol{\Theta})] - \log[q^*(\boldsymbol{\Theta})]\}$ (Ormerod and Wand[80]).

$$\begin{aligned}
\log[\underline{p}(\mathbf{y},\tilde{\mathbf{x}};q)] &= \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\mathbf{y}|\boldsymbol{\eta}_0,\boldsymbol{\beta},\boldsymbol{\delta},\boldsymbol{\Xi},\sigma^2)]\} + \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\tilde{\mathbf{x}}|\boldsymbol{\Xi},\sigma_x^2)]\} \\
&\quad + \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\boldsymbol{\eta}_0)] - \log[q^*(\boldsymbol{\eta}_0)]\} + \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\boldsymbol{\beta})] - \log[q^*(\boldsymbol{\beta})]\} \\
&\quad + \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\boldsymbol{\delta})] - \log[q^*(\boldsymbol{\delta})]\} + \sum_{i=1}^{N}\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\boldsymbol{\xi}_i)] - \log[q^*(\boldsymbol{\xi}_i)]\} \\
&\quad + \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\lambda_x)] - \log[q^*(\lambda_x)]\} + \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\lambda_t)] - \log[q^*(\lambda_t)]\} \\
&\quad + \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\sigma^2)] - \log[q^*(\sigma^2)]\} + \mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\sigma_x^2)] - \log[q^*(\sigma_x^2)]\} \qquad \text{(A.2)}
\end{aligned}$$

The first term in (A.2) is

$$\begin{aligned}
\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\mathbf{y}|\boldsymbol{\eta}_0,\boldsymbol{\beta},\boldsymbol{\delta},\boldsymbol{\Xi},\sigma^2)]\} &= \mathrm{E}_{\boldsymbol{\Theta}}\left[-\frac{N}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}||\mathbf{y} - \mathbb{U}\boldsymbol{\eta}_0 - \boldsymbol{\eta}_1||_2^2\right] + C \\
&= -\frac{N}{2}\mathrm{E}_{\boldsymbol{\Theta}}[\log(\sigma^2)] - \mu_{q(1/\sigma^2)}(B_{q(\sigma^2)} - b_s) + C,
\end{aligned}$$

where C is used from here on to represent any constant that will not affect the log-likelihood as the parameter estimates are updated. The second term in (A.2)

is

$$\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\tilde{\mathbf{x}}|\boldsymbol{\Xi},\sigma_x^2)]\} = \mathrm{E}_{\boldsymbol{\Theta}}\left[-\frac{\sum_{i=1}^N n_i}{2}\log(\sigma_x^2) - \frac{1}{2\sigma_x^2}\sum_{i=1}^N ||\tilde{\mathbf{x}}_i - \mu_x(\mathbf{t}_i) - \Phi(\mathbf{t}_i)\boldsymbol{\xi}_i||_2^2\right]$$
$$+ C = -\frac{\sum_{i=1}^N n_i}{2}\mathrm{E}_{\boldsymbol{\Theta}}[\log(\sigma_x^2)] - \mu_{q(1/\sigma_x^2)}(B_{q(\sigma_x^2)} - b_x) + C.$$

The third term (recalling that $\sigma_{\boldsymbol{\eta}_0}^2$ is fixed) is

$$\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\boldsymbol{\eta}_0)] - \log[q^*(\boldsymbol{\eta}_0)]\} = \mathrm{E}_{\boldsymbol{\Theta}}\left[-\frac{1}{2\sigma_{\boldsymbol{\eta}_0}^2}\boldsymbol{\eta}_0^T\boldsymbol{\eta}_0\right.$$
$$\left. + \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)}|) + \frac{1}{2}(\boldsymbol{\eta}_0 - \mu_{q(\boldsymbol{\eta}_0)})^T{\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)}}^{-1}(\boldsymbol{\eta}_0 - \mu_{q(\boldsymbol{\eta}_0)})\right] + C$$
$$= -\frac{1}{2\sigma_{\boldsymbol{\eta}_0}^2}\left[{\mu_{q(\boldsymbol{\eta}_0)}}^T\mu_{q(\boldsymbol{\eta}_0)} + \mathrm{tr}(\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)})\right] + \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)}|) + C$$

The fourth term (recalling that $\sigma_{\boldsymbol{\beta}}^2$ is fixed) is

$$\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\boldsymbol{\beta})] - \log[q^*(\boldsymbol{\beta})]\} = \mathrm{E}_{\boldsymbol{\Theta}}\left[-\frac{1}{2\sigma_{\boldsymbol{\beta}}^2}\boldsymbol{\beta}^T\boldsymbol{\beta} + \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})}|)\right.$$
$$\left. + \frac{1}{2}(\boldsymbol{\beta} - \mu_{q(\boldsymbol{\beta})})^T{\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})}}^{-1}(\boldsymbol{\beta} - \mu_{q(\boldsymbol{\beta})})\right] + C$$
$$= -\frac{1}{2\sigma_{\boldsymbol{\beta}}^2}\left[{\mu_{q(\boldsymbol{\beta})}}^T\mu_{q(\boldsymbol{\beta})} + \mathrm{tr}(\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})})\right] + \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})}|) + C$$

The fifth term is

$$\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\boldsymbol{\delta})] - \log[q^*(\boldsymbol{\delta})]\} = \mathrm{E}_{\boldsymbol{\Theta}}\left[\frac{1}{2}\log|\lambda_x\boldsymbol{\Psi}_x + \lambda_t\boldsymbol{\Psi}_t| - \frac{1}{2}\boldsymbol{\delta}^T(\lambda_x\boldsymbol{\Psi}_x + \lambda_t\boldsymbol{\Psi}_t)\boldsymbol{\delta}\right.$$
$$\left. + \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}|) + \frac{1}{2}(\boldsymbol{\delta} - \mu_{q(\boldsymbol{\delta})})^T{\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}}^{-1}(\boldsymbol{\delta} - \mu_{q(\boldsymbol{\delta})})\right] + C$$
$$\leq \frac{1}{2}\log\left|\mu_{q(\lambda_x)}\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right| - \frac{1}{2}{\mu_{q(\boldsymbol{\delta})}}^T(\mu_{q(\lambda_x)}\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t)\mu_{q(\boldsymbol{\delta})}$$
$$- \frac{1}{2}\mu_{q(\lambda_x)}\,\mathrm{tr}\left(\boldsymbol{\Psi}_x\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}\right) - \frac{1}{2}\mu_{q(\lambda_t)}\,\mathrm{tr}\left(\boldsymbol{\Psi}_t\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}\right) + \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}|) + C$$

Where the inequality follows from Jensen's inequality and the log-concavity of the determinant over the class of positive definite matrices. This inequality is not in the direction we want. If we use the approximation $\mathrm{E}_{\boldsymbol{\Theta}}\log|\lambda_x\boldsymbol{\Psi}_x + \lambda_t\boldsymbol{\Psi}_t| \approx$

$\log\left|\mu_{q(\lambda_x)}\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right|$, we appear to lose our guarantee of increasing the lower bound on the log-likelihood at each iteration.

In the sixth term we have

$$\begin{aligned}
\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\boldsymbol{\xi}_i)] - \log[q^*(\boldsymbol{\xi}_i)]\} &= \mathrm{E}_{\boldsymbol{\Theta}}\left[-\frac{1}{2}\boldsymbol{\xi}_i^T \operatorname{diag}(\boldsymbol{\nu}^{-1})\boldsymbol{\xi}_i + \frac{M}{2}\log(|\boldsymbol{\Lambda}_i|)\right. \\
&\left. + \frac{1}{2}(\boldsymbol{\xi}_i - \boldsymbol{\xi}_{i,0})^T\boldsymbol{\Lambda}_i^{-1}(\boldsymbol{\xi}_i - \boldsymbol{\xi}_{i,0})\right] + C \\
&= -\frac{1}{2}\left\{\boldsymbol{\xi}_{i,0}^T \operatorname{diag}(\boldsymbol{\nu}^{-1})\boldsymbol{\xi}_{i,0} + \operatorname{tr}[\operatorname{diag}(\boldsymbol{\nu}^{-1})\boldsymbol{\Lambda}_i]\right\} + \frac{M}{2}\log(|\boldsymbol{\Lambda}_i|) + C,\ i = 1, \ldots, N
\end{aligned}$$

For the seventh term

$$\begin{aligned}
\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\lambda_x)] - \log[q^*(\lambda_x)]\} &= \mathrm{E}_{\boldsymbol{\Theta}}\left\{(a_l + 1)\log(\lambda_x) - b_l\lambda_x\right. \\
&- \frac{1}{2}\log\left|\lambda_x\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right| - \log(c_{q(\lambda_x)}) \\
&\left. + \frac{1}{2}\left(\operatorname{tr}(\boldsymbol{\Psi}_x\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}) + {\mu_{q(\boldsymbol{\delta})}}^T\boldsymbol{\Psi}_x\mu_{q(\boldsymbol{\delta})}\right)\lambda_x - (a_l + 1)\log(\lambda_x) + b_l\lambda_x\right\} + C \\
&\approx (a_l + 1)\mathrm{E}_{\boldsymbol{\Theta}}[\log(\lambda_x)] - \frac{1}{2}\log\left|\mu_{q(\lambda_x)}\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right| - \log(c_{q(\lambda_x)}) \\
&\quad + \frac{1}{2}\left(\operatorname{tr}(\boldsymbol{\Psi}_x\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}) + {\mu_{q(\boldsymbol{\delta})}}^T\boldsymbol{\Psi}_x\mu_{q(\boldsymbol{\delta})}\right)\mu_{q(\lambda_x)} + C
\end{aligned}$$

For the eighth term

$$\begin{aligned}
\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\lambda_t)] - \log[q^*(\lambda_t)]\} &\approx (a_l + 1)\mathrm{E}_{\boldsymbol{\Theta}}[\log(\lambda_t)] - \frac{1}{2}\log\left|\mu_{q(\lambda_x)}\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right| \\
&\quad - \log(c_{q(\lambda_t)}) + \frac{1}{2}\left(\operatorname{tr}(\boldsymbol{\Psi}_t\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}) + {\mu_{q(\boldsymbol{\delta})}}^T\boldsymbol{\Psi}_t\mu_{q(\boldsymbol{\delta})}\right)\mu_{q(\lambda_t)} + C
\end{aligned}$$

For the ninth term

$$\begin{aligned}
\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\sigma^2)] - \log[q^*(\sigma^2)]\} &= \mathrm{E}_{\boldsymbol{\Theta}}\left\{-(a_s + 1)\log(\sigma^2) - \frac{b_s}{\sigma^2}\right. \\
&\quad \left. - (a_s + N/2)\log(B_{q(\sigma^2)}) + (a_s + N/2 + 1)\log(\sigma^2) + \frac{B_{q(\sigma^2)}}{\sigma^2}\right\} + C \\
&= \frac{N}{2}\mathrm{E}_{\boldsymbol{\Theta}}[\log(\sigma^2)] - (a_s + N/2)\log(B_{q(\sigma^2)}) + \mu_{q(1/\sigma^2)}(B_{q(\sigma^2)} - b_s) + C
\end{aligned}$$

The tenth term is

$$\begin{aligned}
\mathrm{E}_{\boldsymbol{\Theta}}\{\log[p(\sigma_x^2)] - \log[q^*(\sigma_x^2)]\} &= \mathrm{E}_{\boldsymbol{\Theta}}\left\{-(a_x+1)\log(\sigma_x^2) - \frac{b_x}{\sigma_x^2}\right. \\
&- (a_x + \sum_{i=1}^{N} n_i/2)\log(B_{q(\sigma_x^2)}) + (a_x + \sum_{i=1}^{N} n_i/2 + 1)\log(\sigma_x^2) + \left.\frac{B_{q(\sigma_x^2)}}{\sigma_x^2}\right\} + C \\
&= \frac{\sum_{i=1}^{N} n_i}{2}\mathrm{E}_{\boldsymbol{\Theta}}[\log(\sigma_x^2)] - (a_x + \sum_{i=1}^{N} n_i/2)\log(B_{q(\sigma_x^2)}) + \mu_{q(1/\sigma_x^2)}(B_{q(\sigma_x^2)} - b_x) + C
\end{aligned}$$

Combining all ten terms, several components cancel and we are left with

$$\begin{aligned}
\log[\underline{p}(\mathbf{y}, \tilde{\mathbf{x}}; q)] \approx &-\frac{1}{2\sigma_{\boldsymbol{\eta}_0}^2}\left[{\mu_{q(\boldsymbol{\eta}_0)}}^T \mu_{q(\boldsymbol{\eta}_0)} + \mathrm{tr}(\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)})\right] + \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)}|) \\
&- \frac{1}{2\sigma_{\boldsymbol{\beta}}^2}\left[{\mu_{q(\boldsymbol{\beta})}}^T \mu_{q(\boldsymbol{\beta})} + \mathrm{tr}(\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})})\right] + \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\beta})}|) \\
&- \frac{1}{2}\sum_{i=1}^{N}\left\{\boldsymbol{\xi}_{i,0}^T \,\mathrm{diag}(\boldsymbol{\nu}^{-1})\boldsymbol{\xi}_{i,0} + \mathrm{tr}[\mathrm{diag}(\boldsymbol{\nu}^{-1})\boldsymbol{\Lambda}_i] - M\log(|\boldsymbol{\Lambda}_i|)\right\} \\
&+ (a_l+1)\mathrm{E}_{\boldsymbol{\Theta}}[\log(\lambda_x)] - (a_s + N/2)\log(B_{q(\sigma^2)}) - (a_x + \sum_{i=1}^{N} n_i/2)\log(B_{q(\sigma_x^2)}) \\
&+ \frac{1}{2}\log(|\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}|) + (a_l+1)\mathrm{E}_{\boldsymbol{\Theta}}[\log(\lambda_t)] \\
&- \frac{1}{2}\log\left|\mu_{q(\lambda_x)}\boldsymbol{\Psi}_x + \mu_{q(\lambda_t)}\boldsymbol{\Psi}_t\right| - \log(c_{q(\lambda_t)}/c_{q(\lambda_x)})
\end{aligned} \tag{A.3}$$

A.4 Complete Variational Bayes Algorithm

Below is the full VB algorithm. Note that it is spread over two pages.

Algorithm 2 Steps for estimating parameters from optimal densities, $q^*(\boldsymbol{\theta})$, for FGAM

Initialize $B_{q(\sigma^2)}, B_{q(\sigma_x^2)}, \mu_{q(\lambda_x)}, \mu_{q(\lambda_t)} > 0$, $\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)} = \mathbb{I}_{p_0}$, $\boldsymbol{\Sigma}_{q(\beta)} = \mathbb{I}_{d_x d_t}$, $\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})} = \mathbb{I}_{K_x K_t - d_x d_t}$, $\mu_{q(\boldsymbol{\eta}_0)} = \mathbf{0}$, $\mu_{q(\beta)} = \mathbf{0}$, $\mu_{q(\boldsymbol{\delta})} = \mathbf{0}$.

Choose grid of G points, $\mathbf{g}$, and obtain Gauss-Laguerre quadrature weights, $\mathbf{L}_g$, for numerical integration of optimal densities for λ_x, λ_t.

Compute $\boldsymbol{\nu}$, $\boldsymbol{\mu}_x$, $\boldsymbol{\Phi}$, $\boldsymbol{\mu}_x(\mathbf{t}_i)$, $\boldsymbol{\Phi}(\mathbf{t}_i)$, $i = 1, \ldots, N$, from an initial functional principal components analysis.

repeat

  for $i = 1 \rightarrow N$ **do**

    $\boldsymbol{\xi}_{i,0} \leftarrow$ mode of $\log q(\boldsymbol{\xi}_i)$

$$= \mu_{q(1/\sigma^2)} \left[(y_i - \mathbf{u}_i^T \mu_{q(\boldsymbol{\eta}_0)}) \mathbf{b}_{\boldsymbol{\xi}_i}^T \mu_{q(\boldsymbol{\theta})} - \tfrac{1}{2} (\mathbf{b}_{\boldsymbol{\xi}_i}^T \mu_{q(\boldsymbol{\theta})})^2 + \tfrac{1}{2} \mathbf{b}_{\boldsymbol{\xi}_i}^T \boldsymbol{\Sigma}_{q(\boldsymbol{\theta})} \mathbf{b}_{\boldsymbol{\xi}_i} \right]$$
$$+ (\tilde{\mathbf{x}}_i - \boldsymbol{\mu}_x(\mathbf{t}_i))^T \boldsymbol{\Phi}(\mathbf{t}_i) \boldsymbol{\xi}_i - \tfrac{1}{2} \boldsymbol{\xi}_i^T \left[\mu_{q(1/\sigma_x^2)} \boldsymbol{\Phi}^T(\mathbf{t}_i) \boldsymbol{\Phi}(\mathbf{t}_i) + \operatorname{diag}(\boldsymbol{\nu}^{-1}) \right] \boldsymbol{\xi}_i$$

    $\mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})] \leftarrow \boldsymbol{\Phi}^T \mathbf{B}'_{\boldsymbol{\xi}_{i,0}} \odot (\mathbf{L} \otimes \mathbf{1}_{K_x K_t}^T)$

    $\mathcal{D}_{\boldsymbol{\xi}_i^2}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})] \leftarrow (\boldsymbol{\Phi}^T \otimes \boldsymbol{\Phi}^T)$

$$\times \left[\ell_1 \cdot (\mathbf{B}''_{\boldsymbol{\xi}_{i,0}})_{1,i}, \mathbf{0}_{K_x K_t \times T}, \ell_2 \cdot (\mathbf{B}''_{\boldsymbol{\xi}_{i,0}})_{2,i}, \ldots, \ell_T \cdot (\mathbf{B}''_{\boldsymbol{\xi}_{i,0}})_{T,i} \right]^T$$

    $\operatorname{vec}(\boldsymbol{\Lambda}_i^{-1}) \leftarrow \Big[\mu_{q(1/\sigma^2)} \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T) \mu_{q(\boldsymbol{\theta})} (y_i - \mathbf{u}_i^T \mu_{q(\boldsymbol{\eta}_0)} - \mathbf{b}_{\boldsymbol{\xi}_i}^T \mu_{q(\boldsymbol{\theta})})$

$$+ \mu_{q(1/\sigma^2)} \operatorname{vec} \left\{ \mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T) \mu_{q(\boldsymbol{\theta})} [\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T) \mu_{q(\boldsymbol{\theta})}]^T \right\}$$
$$+ \operatorname{vec}[\mu_{q(1/\sigma_x^2)} \boldsymbol{\Phi}(\mathbf{t}_i)^T \boldsymbol{\Phi}(\mathbf{t}_i) + \operatorname{diag}(\boldsymbol{\nu}^{-1})]$$
$$+ \mu_{q(1/\sigma^2)} \left\{ \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T) \boldsymbol{\Sigma}_{q(\boldsymbol{\theta})} \mathbf{b}_{\boldsymbol{\xi}_i} + \operatorname{vec} \left[\mathcal{D}_{\boldsymbol{\xi}_i}(\mathbf{b}_{\boldsymbol{\xi}_i}^T) \boldsymbol{\Sigma}_{q(\boldsymbol{\theta})} \mathcal{D}_{\boldsymbol{\xi}_i}^T(\mathbf{b}_{\boldsymbol{\xi}_i}^T) \right] \right\} \Big]_{\boldsymbol{\xi}_i = \boldsymbol{\xi}_{i,0}}$$

    $\mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})} \leftarrow \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}) + \frac{1}{2} \left\{ \mathcal{D}_{\boldsymbol{\xi}_i^2}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})] \right\}^T \operatorname{vec}(\boldsymbol{\Lambda}_i)$

    $\mathrm{E}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i} \mathbf{b}_{\boldsymbol{\xi}_i}^T] \leftarrow \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}) \mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0}) + \mathbf{b}_{\boldsymbol{\xi}_i}(\boldsymbol{\xi}_{i,0}) \operatorname{vec}(\boldsymbol{\Lambda}_i)^T \mathcal{D}_{\boldsymbol{\xi}_i^2}(\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0}))$

$$+ \left\{ \mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})] \right\}^T \boldsymbol{\Lambda}_i \mathcal{D}_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}^T(\boldsymbol{\xi}_{i,0})]$$

  end for

  $\boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)} \leftarrow \left\{ \mu_{q(1/\sigma^2)} \mathbb{U}^T \mathbb{U} + \frac{1}{\sigma_{\eta_0}^2} \mathbb{I}_{p_0} \right\}^{-1}$

  $\mu_{q(\boldsymbol{\eta}_0)} \leftarrow \boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)} \mathbb{U}^T \left(\mathbf{y} - \mu_{q(\boldsymbol{\eta}_1)} \right) \mu_{q(1/\sigma^2)}$

$\boldsymbol{\Sigma}_{q(\beta)} \leftarrow \left\{ \mathbb{T}_0^T \left[\sum_{i=1}^N \mathrm{E}_{\boldsymbol{\xi}} \left(\mathbf{b}_{\boldsymbol{\xi}_i} \mathbf{b}_{\boldsymbol{\xi}_i}^T \right) \right] \mathbb{T}_0 \mu_{q(1/\sigma^2)} + \frac{1}{\sigma_\beta^2} \mathbb{I}_{d_x d_t} \right\}^{-1}$

$\mu_{q(\beta)} \leftarrow \boldsymbol{\Sigma}_{q(\beta)} \mathbb{T}_0^T \left\{ \mu_{q(\mathbf{b}_{\boldsymbol{\xi}})}^T (\mathbf{y} - \mathbb{U} \mu_{\mathbf{q}(\boldsymbol{\eta}_\mathbf{0})}) - \left[\sum_{i=1}^N \mathrm{E}_{\boldsymbol{\xi}} \left(\mathbf{b}_{\boldsymbol{\xi}_i} \mathbf{b}_{\boldsymbol{\xi}_i}^T \right) \right] \mathbb{T}_p \mu_{q(\boldsymbol{\delta})} \right\} \mu_{q(1/\sigma^2)}$

$\boldsymbol{\Sigma}_{q(\boldsymbol{\delta})} \leftarrow \left\{ \mathbb{T}_p^T \left[\sum_{i=1}^N \mathrm{E}_{\boldsymbol{\xi}} \left(\mathbf{b}_{\boldsymbol{\xi}_i} \mathbf{b}_{\boldsymbol{\xi}_i}^T \right) \right] \mathbb{T}_p \mu_{q(1/\sigma^2)} + \mu_{q(\lambda_x)} \boldsymbol{\Psi}_x + \mu_{q(\lambda_t)} \boldsymbol{\Psi}_t \right\}^{-1}$

$\mu_{q(\boldsymbol{\delta})} \leftarrow \boldsymbol{\Sigma}_{q(\boldsymbol{\delta})} \mathbb{T}_p^T \left\{ \mu_{q(\mathbf{b}_{\boldsymbol{\xi}})}^T (\mathbf{y} - \mathbb{U} \mu_{\mathbf{q}(\boldsymbol{\eta}_\mathbf{0})}) - \left[\sum_{i=1}^N \mathrm{E}_{\boldsymbol{\xi}} \left(\mathbf{b}_{\boldsymbol{\xi}_i} \mathbf{b}_{\boldsymbol{\xi}_i}^T \right) \right] \mathbb{T}_0 \mu_{q(\beta)} \right\} \mu_{q(1/\sigma^2)}$

for $i = 1 \rightarrow G$ **do**

  $\ell_{\lambda_x}(g_i) \leftarrow \frac{1}{2} \log \left| g_i \boldsymbol{\Psi}_x + \mu_{q(\lambda_t)} \boldsymbol{\Psi}_t \right| - g_i \left\{ b_l + \frac{1}{2} \left[\mathrm{tr}(\boldsymbol{\Psi}_x \boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}) + \mu_{q(\boldsymbol{\delta})}{}^T \boldsymbol{\Psi}_x \mu_{q(\boldsymbol{\delta})} \right] \right\}$

end for

$\mu_{q(\lambda_x)} \leftarrow [\mathbf{L}_g^T \ell_{\lambda_x}(\mathbf{g})]^{-1} \mathbf{L}_g^T \{ \mathbf{g} \odot \exp[\ell_{\lambda_x}(\mathbf{g}) - \max_{\mathbf{g}} \ell_{\lambda_x}(\mathbf{g})] \}$

for $i = 1 \rightarrow G$ **do**

  $\ell_{\lambda_t}(g_i) \leftarrow \frac{1}{2} \log \left| \mu_{q(\lambda_x)} \boldsymbol{\Psi}_x + g_i \boldsymbol{\Psi}_t \right| - g_i \left\{ b_l + \frac{1}{2} \left[\mathrm{tr}(\boldsymbol{\Psi}_t \boldsymbol{\Sigma}_{q(\boldsymbol{\delta})}) + \mu_{q(\boldsymbol{\delta})}{}^T \boldsymbol{\Psi}_t \mu_{q(\boldsymbol{\delta})} \right] \right\}$

end for

$\mu_{q(\lambda_t)} \leftarrow [\mathbf{L}_g^T \ell_{\lambda_t}(\mathbf{g})]^{-1} \mathbf{L}_g^T \{ \mathbf{g} \odot \exp[\ell_{\lambda_t}(\mathbf{g}) - \max_{\mathbf{g}} \ell_{\lambda_t}(\mathbf{g})] \}$

$B_{q(\sigma_x^2)} \leftarrow b_x + \frac{1}{2} \sum_{i=1}^N \left[||\tilde{\mathbf{x}}_i - \mu_x(\mathbf{t}_i) - \Phi(\mathbf{t}_i) \boldsymbol{\xi}_{i,0}||_2^2 + \mathrm{tr} \left(\Phi(\mathbf{t}_i)^T \Phi(\mathbf{t}_i) \boldsymbol{\Lambda}_i \right) \right]$

$\mu_{q(1/\sigma_x^2)} \leftarrow (a_x + \sum_{i=1}^N n_i/2) / B_{q(\sigma_x^2)}$

$B_{q(\sigma^2)} \leftarrow b_s + \frac{1}{2} ||(\mathbf{y} - \mathbb{U} \mu_{q(\boldsymbol{\eta}_0)} - \mu_{q(\boldsymbol{\eta}_1)})||_2^2$
$+ \frac{1}{2} \mathrm{tr} \left(\mathbb{U}^T \mathbb{U} \boldsymbol{\Sigma}_{q(\boldsymbol{\eta}_0)} \right) + \frac{1}{2} \mathrm{tr} \left\{ \left[\sum_{i=1}^N \mathrm{E}_{\boldsymbol{\xi}} \left(\mathbf{b}_{\boldsymbol{\xi}_i} \mathbf{b}_{\boldsymbol{\xi}_i}^T \right) \right] \boldsymbol{\Sigma}_{q(\boldsymbol{\theta})} \right\}$
$+ \frac{1}{2} \mu_{q(\boldsymbol{\theta})}{}^T \left[\sum_{i=1}^N \mathrm{E}_{\boldsymbol{\xi}} \left(\mathbf{b}_{\boldsymbol{\xi}_i} \mathbf{b}_{\boldsymbol{\xi}_i}^T \right) \right] \mu_{q(\boldsymbol{\theta})} - \frac{1}{2} \mu_{q(\boldsymbol{\theta})}{}^T \mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})}{}^T \mu_{q(\mathbf{b}_{\boldsymbol{\xi}_i})} \mu_{q(\boldsymbol{\theta})}$

$\mu_{q(1/\sigma^2)} \leftarrow (a_x + N/2) / B_{q(\sigma^2)}$

until Change in $\underline{p}(\mathbf{y}, \tilde{\mathbf{x}}; q)$ is negligible *OR* maximum number of iterations reached

APPENDIX B

DERIVATION OF BAYES FACTORS FOR TESTING LINEARITY

In this appendix we provide derivations for the Bayes factors given in Section 4.6.2. We use the general linear mixed model (4.6), which we denote $\mathcal{M}$. In Section B.1 we derive a formula for a model with an unspecified number, J, of variance components and in Section B.2 we provide an alternate expression for the $J = 3$ case that applies to FGAM.

B.1 For Arbitrary Number of Variance Components

To compute the desired Bayes factor, we first need $\int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{b}, \sigma^2)p(\boldsymbol{\beta})d\boldsymbol{\beta}$. Defining $\hat{\boldsymbol{\beta}} := (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\mathbf{y}$, we have

$$||\mathbf{y} - \mathbb{X}\boldsymbol{\beta} - \mathbb{Z}\mathbf{b}||^2 = ||\mathbf{y} - \mathbb{X}\hat{\boldsymbol{\beta}} - \mathbb{Z}\mathbf{b}||^2 + ||\mathbb{X}\hat{\boldsymbol{\beta}} - \mathbb{X}\boldsymbol{\beta}||^2 = ||\mathbf{y} - \mathbb{X}\hat{\boldsymbol{\beta}} - \mathbb{Z}\mathbf{b}||^2 + (\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})^T(\mathbb{X}^T\mathbb{X})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}),$$

so that

$$\int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{b}, \sigma^2)p(\boldsymbol{\beta})d\boldsymbol{\beta} = \frac{|\mathbb{X}^T\mathbb{X}|^{-1/2}}{(2\pi\sigma^2)^{(n-q_0)/2}} \int_{\mathbb{R}^{q_0}} \phi_{q_0}(\boldsymbol{\beta}; \hat{\boldsymbol{\beta}}, \sigma^2(\mathbb{X}^T\mathbb{X})^{-1})d\boldsymbol{\beta} = \frac{|\mathbb{X}^T\mathbb{X}|^{-1/2}}{(2\pi\sigma^2)^{(n-q_0)/2}},$$

where $\phi_p(\cdot\,; \boldsymbol{\mu}, \Sigma)$ is the pdf of a $N_p(\boldsymbol{\mu}, \Sigma)$ distribution.

Now for each j consider the eigendecomposition of $\mathbb{Z}_j^T\mathbb{Z}_j = \mathbb{W}_j^T\bar{\mathbb{D}}^2\mathbb{W}_j$ with $\mathbb{W}_j$ orthogonal and $\bar{\mathbb{D}}_j = \text{diag}(\bar{d}_{j1}, \ldots, \bar{d}_{jq_j})$, and define $\bar{\mathbb{U}}_j := \mathbb{Z}_j\mathbb{W}_j\bar{\mathbb{D}}_j^{-1}$ so that $\mathbb{Z}_j = \bar{\mathbb{U}}_j\bar{\mathbb{D}}_j\mathbb{W}_j^T$. We also define $\bar{\mathbb{D}} := \text{blkdiag}(\bar{\mathbb{D}}_1, \ldots, \bar{\mathbb{D}}_j)$, $\bar{\mathbb{U}} := [\bar{\mathbb{U}}_1 : \cdots : \bar{\mathbb{U}}_j]$, and $\mathbb{W} := [\mathbb{W}_1 : \cdots : \mathbb{W}_j]$, so that $\mathbb{Z} = [\bar{\mathbb{U}}_1\bar{\mathbb{D}}_1\mathbb{W}_1^T : \cdots : \bar{\mathbb{U}}_j\bar{\mathbb{D}}_j\mathbb{W}_j^T] = \bar{\mathbb{U}}\bar{\mathbb{D}}\mathbb{W}^T$ and $\mathbb{Z}^T\mathbb{Z} = \mathbb{W}^T\bar{\mathbb{D}}^2\mathbb{W}$. When integrating with respect to $\mathbf{b}$, we make the orthogonal

transformation $\mathbf{b}_* = \mathbb{W}^T\mathbf{b}$ and thus have

$$\int_{\mathbb{R}^q}\int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta},\mathbf{b},\sigma^2)p(\boldsymbol{\beta})p(\mathbf{b}|\mathbf{g},\sigma^2)d\boldsymbol{\beta}d\mathbf{b} = \frac{|\mathbb{X}^T\mathbb{X}|^{-1/2}|\boldsymbol{\Psi}|^{-1/2}}{(2\pi\sigma^2)^{(n-q_0)/2}(2\pi\sigma^2)^{q/2}}$$
$$\times \int_{\mathbb{R}^q} \exp\left[-\frac{1}{2\sigma^2}(||y-\mathbb{X}\hat{\boldsymbol{\beta}}-\bar{\mathbb{U}}\bar{\mathbb{D}}\mathbf{b}_*||^2 + \mathbf{b}_*^T\boldsymbol{\Psi}^{-1}\mathbf{b}_*)\right]d\mathbf{b}_*,$$

where $\boldsymbol{\Psi}$ is the diagonal matrix $\boldsymbol{\Psi} = \text{blkdiag}(\boldsymbol{\Psi}_1,\ldots,\boldsymbol{\Psi}_j)$. If we define $\mathbf{r} = \mathbf{y}-\mathbb{X}\hat{\boldsymbol{\beta}}$, then by completing the square in the exponential function, we obtain

$$||\mathbf{r}-\bar{\mathbb{U}}\bar{\mathbb{D}}\mathbf{b}_*||^2 + \mathbf{b}_*^T\boldsymbol{\Psi}^{-1}\mathbf{b}_* = [\mathbf{b}_* - (\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1})^{-1}\bar{\mathbb{D}}\bar{\mathbb{U}}^T\mathbf{r}]^T(\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1})$$
$$\times [\mathbf{b}_* - (\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1})^{-1}\bar{\mathbb{D}}\bar{\mathbb{U}}^T\mathbf{r}] - \mathbf{r}^T\bar{\mathbb{U}}\bar{\mathbb{D}}(\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1})^{-1}\bar{\mathbb{D}}\bar{\mathbb{U}}^T\mathbf{r} + \mathbf{r}^T\mathbf{r}.$$

Our integral becomes

$$\begin{aligned}\int_{\mathbb{R}^q}\int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta},\mathbf{b},\sigma^2)p(\boldsymbol{\beta})p(\mathbf{b}|\mathbf{g},\sigma^2)d\boldsymbol{\beta}d\mathbf{b} &= \frac{|\mathbb{X}^T\mathbb{X}|^{-1/2}|\boldsymbol{\Psi}|^{-1/2}|\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1}|^{-1/2}}{(2\pi\sigma^2)^{(n-q_0)/2}(2\pi\sigma^2)^{q_0/2}}\\
&\times \exp\left[-\frac{1}{2\sigma^2}(\mathbf{r}^T\bar{\mathbb{U}}\bar{\mathbb{D}}(\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1})^{-1}\bar{\mathbb{D}}\bar{\mathbb{U}}^T\mathbf{r}-\mathbf{r}^T\mathbf{r})\right]\\
&\times \int_{\mathbb{R}^q}\phi_q(\mathbf{b}_*;(\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1})^{-1}\bar{\mathbb{D}}\bar{\mathbb{U}}^T\mathbf{r},\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1})d\mathbf{b}_*\\
&= \frac{|\mathbb{X}^T\mathbb{X}|^{-1/2}|\boldsymbol{\Psi}|^{-1/2}|\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1}|^{-1/2}}{(2\pi\sigma^2)^{(n-q_0)/2}}\\
&\times \exp\left[-\frac{1}{2\sigma^2}(\mathbf{r}^T\bar{\mathbb{U}}\bar{\mathbb{D}}(\bar{\mathbb{D}}^2+\boldsymbol{\Psi}^{-1})^{-1}\bar{\mathbb{D}}\bar{\mathbb{U}}^T\mathbf{r}-\mathbf{r}^T\mathbf{r})\right]. \qquad \text{(B.1)}\end{aligned}$$

To proceed further, we must discuss the form of the diagonal matrices $\boldsymbol{\Psi}_j$. We follow Maruyama and George[66] and define $\boldsymbol{\Psi}_j = \text{diag}\{\psi_{j1}(g_j,\nu_{j1}),\ldots,\psi_{jq_j}(g_j,\nu_{jq_j})\}$ where $\psi_{ji}(g_j,\nu_{ji}) = \bar{d}_{ji}^{-2}[(1+g_j)\nu_{ji}-1]$, with $\nu_{ji}\geq 1$ for all $1\leq i\leq q_j$ and $1\leq j\leq J$ so that $\psi_{ji}(g_j,\nu_{ji})>0$. Maruyama and George[66] use $\nu_{ji}^{1/2} = \bar{d}_{ji}/\bar{d}_{q_j}$.

Letting $\bar{\mathbf{u}}_{ji}$ denoted the ith column of $\bar{\mathbb{U}}_j$, the term inside the exponential

function in (B.1) can be simplified as follows

$$
\begin{aligned}
-\mathbf{r}^T\bar{\mathbb{U}}\bar{\mathbb{D}}(\bar{\mathbb{D}}^2+\mathbf{\Psi}^{-1})^{-1}\bar{\mathbb{D}}\bar{\mathbb{U}}^T\mathbf{r}+\mathbf{r}^T\mathbf{r} &= -\mathbf{r}^T\left(\sum_{j=1}^{J}\sum_{i=1}^{q_j}\bar{\mathbf{u}}_{ji}\bar{\mathbf{u}}_{ji}^T\frac{\bar{d}_{ji}^2}{\bar{d}_{ji}^2+\psi_{ji}^{-1}}\right)\mathbf{r}+\mathbf{r}^T\mathbf{r}\\
&= -\mathbf{r}^T\left(\sum_{j=1}^{J}\sum_{i=1}^{q_j}\bar{\mathbf{u}}_{ji}\bar{\mathbf{u}}_{ji}^T\frac{(1+g_j)\nu_{ji}-1}{(1+g_j)\nu_{ji}}\right)\mathbf{r}+\mathbf{r}^T\mathbf{r}\\
&= -\sum_{j=1}^{J}\frac{1}{1+g_j}\sum_{i=1}^{q_j}(\bar{\mathbf{u}}_{ji}^T\mathbf{r})^2\frac{(1+g_j)\nu_{ji}-1}{\nu_{ji}}+\mathbf{r}^T\mathbf{r}\\
&= -\sum_{j=1}^{J}\frac{g_j}{1+g_j}\sum_{i=1}^{q_j}(\bar{\mathbf{u}}_{ji}^T\mathbf{r})^2+-\sum_{j=1}^{J}\frac{1}{1+g_j}\sum_{i=1}^{q_j}(\bar{\mathbf{u}}_{ji}^T\mathbf{r})^2(1-v_{ji}^{-1})+\mathbf{r}^T\mathbf{r}\\
&= \sum_{j=1}^{J}\frac{g_j}{1+g_j}\left[J^{-1}\mathbf{r}^T\mathbf{r}-\sum_{i=1}^{q_j}(\bar{\mathbf{u}}_{ji}^T\mathbf{r})^2\right]+\sum_{j=1}^{J}\frac{1}{1+g_j}\left[J^{-1}\mathbf{r}^T\mathbf{r}-\sum_{i=1}^{q_j}(\bar{\mathbf{u}}_{ji}^T\mathbf{r})^2(1-v_{ji}^{-1})\right]\\
&= \sum_{j=1}^{J}\frac{||\mathbf{r}||^2}{1+g_j}\left[g_j(J^{-1}-R_j^2)+J^{-1}-Q_j^2\right],
\end{aligned}
$$

where

$$
R_j^2=\sum_{i=1}^{q_j}\frac{(\bar{\mathbf{u}}_{ji}^T\mathbf{r})^2}{\mathbf{r}^T\mathbf{r}}\text{ and }Q_j^2=\sum_{i=1}^{q_j}(1-\nu_{ji}^{-1})\frac{(\bar{\mathbf{u}}_{ji}^T\mathbf{r})^2}{\mathbf{r}^T\mathbf{r}}. \tag{B.2}
$$

For Maruyama and George[66], where there are no random effects and $\mathbf{r}$ is just the centred response, their R^2 is the usual coefficient of multiple determination. Using their choices for the $\boldsymbol{\nu}_j$'s, we get

$$
\begin{aligned}
|\mathbf{\Psi}|^{-1/2}|\bar{\mathbb{D}}^2+\mathbf{\Psi}^{-1}|^{-1/2} &= \left(\prod_{j=1}^{J}\prod_{i=1}^{q_j}\frac{\nu_{ji}+\nu_{ji}g_j-1}{\bar{d}_{ji}^2}\right)^{-1/2}\left(\prod_{j=1}^{J}\prod_{i=1}^{q_j}\frac{\bar{d}_{ji}^2\nu_{ji}(1+g_j)}{\nu_{ji}+\nu_{ji}g_j-1}\right)^{-1/2}\\
&= \prod_{j=1}^{J}\left[\frac{(1+g_j)^{q_j}}{\prod_{i=1}^{q_j}\nu_{ji}^{1/2}}\right],
\end{aligned}
$$

so that (B.1) becomes

$$
\begin{aligned}
&\int_{\mathbb{R}^q}\int_{\mathbb{R}^{q_0}}p(\mathbf{y}|\boldsymbol{\beta},\mathbf{b},\sigma^2)p(\boldsymbol{\beta})p(\mathbf{b}|\mathbf{g},\sigma^2)d\boldsymbol{\beta}d\mathbf{b}\\
&\qquad= \frac{|\mathbb{X}^T\mathbb{X}|^{-1/2}}{(2\pi\sigma^2)^{(n-q_0)/2}}\prod_{j=1}^{J}\left[\frac{(1+g_j)^{-q_j/2}}{\prod_{i=1}^{q_j}\nu_{ji}^{1/2}}\right]\\
&\qquad\quad\times\exp\left\{-\frac{1}{2\sigma^2}\sum_{j=1}^{J}\frac{||\mathbf{r}||^2}{1+g_j}\left[g_j(J^{-1}-R_j^2)+J^{-1}-Q_j^2\right]\right\}
\end{aligned} \tag{B.3}
$$

Now for the integration with respect to σ^2

$$\int_0^\infty \int_{\mathbb{R}^q} \int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{b}, \sigma^2) p(\boldsymbol{\beta}) p(\mathbf{b}|\mathbf{g}, \sigma^2) p(\sigma^2) d\boldsymbol{\beta} d\mathbf{b} d(\sigma^2) = \int_0^\infty \frac{|\mathbb{X}^T\mathbb{X}|^{-1/2}}{(2\pi\sigma^2)^{(n-q_0)/2}}$$
$$\times \prod_{j=1}^J \left[\frac{(1+g_j)^{-q_j/2}}{\prod_{i=1}^{q_j} \nu_{ji}^{1/2}} \right] \exp\left\{ -\frac{1}{2\sigma^2} \sum_{j=1}^J \frac{||\mathbf{r}||^2}{1+g_j} \left[g_j(J^{-1} - R_j^2) + J^{-1} - Q_j^2 \right] \right\} \frac{1}{\sigma^2} d(\sigma^2)$$
$$= \frac{\Gamma[(N-q_0)/2] \cdot |\mathbb{X}^T\mathbb{X}|^{-1/2}}{\pi^{(N-q_0)/2} (\mathbf{r}^T\mathbf{r})^{(N-q_0)}} \prod_{j=1}^J \left[\frac{(1+g_j)^{-q_j/2}}{\prod_{i=1}^{q_j} \nu_{ji}^{1/2}} \right]$$
$$\times \left\{ \sum_{j=1}^J \frac{\left[g_j(J^{-1} - R_j^2) + J^{-1} - Q_j^2 \right]}{1+g_j} \right\}^{-(N-q_0)/2}$$
$$= k_1 \prod_{j=1}^J (1+g_j)^{-q_j/2} \left\{ \sum_{j=1}^J \frac{\left[g_j(J^{-1} - R_j^2) + J^{-1} - Q_j^2 \right]}{1+g_j} \right\}^{-(N-q_0)/2},$$

where $\Gamma[\cdot]$ denotes the Gamma function and

$k_1 = \Gamma[(N-q_0)/2] \cdot |\mathbb{X}^T\mathbb{X}|^{-1/2} (\pi^{1/2} \mathbf{r}^T\mathbf{r})^{-N+q_0} \prod_{j=1}^J \prod_{l=1}^{q_j} \nu_{jl}^{-1/2}$.

Finally, we are left with the integral w.r.t. $\mathbf{g}$. We have

$$M_{\mathcal{M}}(\mathbf{y}) = \int_{\mathbb{R}_+^J} \int_0^\infty \int_{\mathbb{R}^q} \int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{b}, \sigma^2) p(\boldsymbol{\beta}) p(\mathbf{b}|\mathbf{g}, \sigma^2) p(\sigma^2) p(\mathbf{g}) d\boldsymbol{\beta} d\mathbf{b} d(\sigma^2) d\mathbf{g}$$
$$k_1 \int_{\mathbb{R}_+^J} \prod_{j=1}^J p(g_j)(1+g_j)^{-q_j/2} \left\{ \sum_{j=1}^J \frac{\left[g_j(J^{-1} - R_j^2) + J^{-1} - Q_j^2 \right]}{1+g_j} \right\}^{-(N-q_0)/2} d\mathbf{g}$$
$$\frac{k_1}{\prod_{j=1}^J B(a+1, b_j+1)} \int_{\mathbb{R}_+^J} \prod_{j=1}^J \frac{g_j^{b_j}}{(1+g_j)^{a+b_j+2}} (1+g_j)^{-q_j/2}$$
$$\times \left\{ \sum_{j=1}^J \frac{\left[g_j(J^{-1} - R_j^2) + J^{-1} - Q_j^2 \right]}{1+g_j} \right\}^{-(N-q_0)/2} d\mathbf{g}$$

To proceed further we make the substitution $s_j = g_j/(1+g_j)$ so that $g_j = s_j/(1-s_j)$, $ds_j = (1+g_j)^{-2} dg_j$, and $(1+g_j)^{-1} = (1-s_j)^{-1}$. The s_j's are commonly called

shrinkage factors in the Bayesian model selection literature. Our integral becomes

$$\begin{aligned}
M_{\mathcal{M}}(\mathbf{y}) &= \frac{k_1}{\prod_{j=1}^{J} B(a+1, b_j+1)} \int_{[0,1]^J} \prod_{j=1}^{J} \left(\frac{s_j}{1-s_j}\right)^{b_j} (1-s_j)^{a+b_j+2+q_j} \\
&\times \left[\sum_{j=1}^{J} \left\{ s_j(J^{-1}-R_j^2) + (1-s_j)(J^{-1}-Q_j^2) \right\}\right]^{(N-q_0)/2} (1-s_j)^{-2} d\mathbf{s} \\
&= \frac{k_1}{\prod_{j=1}^{J} B(a+1, b_j+1)} \int_{[0,1]^J} \prod_{j=1}^{J} s_j^{b_j} (1-s_j)^{a+q_j} \\
&\times \left[\sum_{j=1}^{J} \left\{ s_j(J^{-1}-R_j^2) + (1-s_j)(J^{-1}-Q_j^2) \right\}\right]^{(N-q_0)/2} d\mathbf{s}
\end{aligned}$$

For the summation inside the integral we have

$$\begin{aligned}
\sum_{j=1}^{J} \left\{ s_j(J^{-1}-R_j^2) + (1-s_j)(J^{-1}-Q_j^2) \right\} &= \sum_{j=1}^{J} \left\{ s_j(Q_j^2-R_j^2) + J^{-1} - Q_j^2 \right\} \\
&= \left(1 - \sum_{j=1}^{J} Q_j^2\right) \left(1 - \sum_{j=1}^{J} s_j \frac{R_j^2 - Q_j^2}{1-\sum_{j=1}^{J} Q_j^2}\right)
\end{aligned}$$

Defining $\upsilon_j := \frac{R_j^2-Q_j^2}{1-\sum_{j=1}^{J} Q_j^2}$ and $\alpha := \frac{N-q_0}{2}$ our marginal density is

$$\begin{aligned}
M_{\mathcal{M}}(\mathbf{y}) &= \frac{k_1}{\prod_{j=1}^{J} B(a+1, b_j+1)} (1 - \sum_{j=1}^{J} Q_j^2)^{(N-q_0)/2} \int_{[0,1]^J} \prod_{j=1}^{J} s_j^{b_j} (1-s_j)^{a+q_j} \\
&\times \left(1 - \sum_{j=1}^{J} \upsilon_j s_j\right)^{(N-q_0)/2} d\mathbf{s} = \frac{k_1}{\prod_{j=1}^{J} B(a+1, b_j+1)} \left(1 - \sum_{j=1}^{J} Q_j^2\right)^{(N-q_0)/2} \\
&\times \prod_{j=1}^{J} \frac{\Gamma(b_j+1)\Gamma(a+q_j/2+1)}{\Gamma(\alpha)} F_A\left(\alpha, b_1+1, \ldots, b_J+1, \alpha, \ldots, \alpha; \upsilon_1, \ldots, \upsilon_J\right) \\
&= k_1 \prod_{j=1}^{J} \frac{B(b_j+1, a+q_j/2+1)}{B(a+1, b_j+1)} \left(1 - \sum_{j=1}^{J} Q_j^2\right)^{(N-q_0)/2} \\
&\quad \times F_A\left(\alpha, b_1+1, \ldots, b_J+1, \alpha, \ldots, \alpha; \upsilon_1, \ldots, \upsilon_J\right).
\end{aligned}$$

The integral representation we have used for the series can be found in Lauricella[55], Eq. (10). The simplification in the numerator terms ($c_1 = \cdots = c_n = \alpha$) came from the choice of $b_j = (N - q_0 - q_j - 2a - 4)/2$.

B.2 Alternative Expression For FGAM

In this appendix, we use the results from Appendix B.1 up to the step where we must integrate w.r.t. $\mathbf{g}$, but consider a different approach for that final integration. We will only consider the $J = 3$ case necessary for FGAM and integrate w.r.t. each component of $\mathbf{g}$ separately. Recall, the form of the prior for g_j,
$p(g_j) = g_j^{b_j}(1+g_j)^{-(a+b_j+2)}B^{-1}(a+1, b_j+1)$. It will be convenient to consider the choice $b_j = (N - q_0 - q_j - 2a - 4)/2$ when computing the Bayes factor.

First considering the integration over g_1, we have

$$\begin{aligned}
&\int_0^\infty \int_0^\infty \int_{\mathbb{R}^q} \int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{b}, \sigma^2) p(\boldsymbol{\beta}) p(\mathbf{b}|\mathbf{g}, \sigma^2) p(g_1) d\boldsymbol{\beta} d\mathbf{b} d(\sigma^2) dg_1 \\
&= k_1 B(a+1, b_1+1) \prod_{j=2}^{3} (1+g_j)^{-q_j/2} \\
&\times \int_0^\infty g_1^a (1+g_1)^{-(a+b_1+2+q_j/2)} \left\{ \sum_{j=1}^{J} \frac{\left[g_j(J^{-1} - R_j^2) + J^{-1} - Q_j^2 \right]}{1+g_j} \right\}^{-(N-q_0)/2} dg_1
\end{aligned}$$

We transform to the shrinkage factors $u_j = g_j/(1+g_j)$, so that our integral becomes

$$\begin{aligned}
&\int_0^\infty \int_0^\infty \int_{\mathbb{R}^q} \int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{b}, \sigma^2) p(\boldsymbol{\beta}) p(\mathbf{b}|\mathbf{g}, \sigma^2) p(g_1) d\boldsymbol{\beta} d\mathbf{b} d(\sigma^2) dg_1 \\
&= k_1 B^{-1}(a+1, b_1+1) \prod_{j=2}^{3} (1-u_j)^{q_j/2} \int_0^1 \left(\frac{u_1}{1-u_1} \right)^{b_1} (1-u_1)^{a+b_1+2+q_1/2} \\
&\times \left\{ \sum_{j=1}^{3} \left[u_j(3^{-1} - R_j^2) + (1-u_j)(3^{-1} - Q_j^2) \right] \right\}^{-(N-q_0)/2} (1-u_1)^{-2} du_1 \\
&= \frac{(1-u_2)^{q_2/2}(1-u_3)^{q_3/2}}{\{k_1^{-1} B(a+1, b_1+1)\}} \int_0^1 u_1^{b_1} (1-u_1)^{a+q_1/2} \left\{ u_1(Q_1^2 - R_1^2) + c_1 \right\}^{-(N-q_0)/2} du_1 \\
&= \frac{(1-u_2)^{q_2/2}(1-u_3)^{q_3/2}}{\{k_1^{-1} B(a+1, b_1+1)\}} c_1^{-(N-q_0)/2} \int_0^1 u_1^{(N-q_0-q_1-2a-4)/2} (1-u_1)^{a+q_1/2} \\
&\times \left\{ u_1(Q_1^2 - R_1^2)/c_1 + 1 \right\}^{-(N-q_0)/2} du_1 = k_1 B^{-1}(a+1, b_1+1) \\
&\times \prod_{j=2}^{3} (1-u_j)^{q_j/2} c_1^{-(N-q_0)/2} \left[1 + (Q_1^2 - R_1^2)/c_1 \right]^{-b_1-1} B[b_1+1, a+q_1/2+1],
\end{aligned}$$

where $c_1 = 3^{-1} - Q_1^2 + \sum_{j=2}^{3}[u_j(Q_j^2 - R_j^2) + 3^{-1} - Q_j^2]$ and the final equality follows from Gradshteyn and Ryzhik[36], p. 287, Eq. 3.197#4.

For g_2 we have

$$\begin{aligned}
&\int_0^\infty \int_0^\infty \int_0^\infty \int_{\mathbb{R}^q} \int_{\mathbb{R}^{q_0}} p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{b}, \sigma^2) p(\boldsymbol{\beta}) p(\mathbf{b}|\mathbf{g}, \sigma^2) p(g_1) p(g_2) d\boldsymbol{\beta} d\mathbf{b} d(\sigma^2) dg_1 dg_2 \\
&\quad = k_2 (1-u_3)^{q_3/2} \int_0^1 \left(\frac{u_2}{1-u_2}\right)^{b_2} (1-u_2)^{a+b_2+2+q_2/2} c_1^{-(N-q_0)/2+b_1+1} \\
&\qquad \times \left\{c_1 + Q_1^2 - R_1^2\right\}^{-b_1-1} (1-u_2)^{-2} du_2 \\
&\quad = k_2 (1-u_3)^{q_3/2} \int_0^1 u_2^{b_2} (1-u_2)^{a+q_2/2} \left[u_2(Q_2^2 - R_2^2) + c_2\right]^{-(a+q_1/2+1)} \\
&\qquad \times \left[u_2(Q_2^2 - R_2^2) + Q_1^2 - R_1^2 + c_2\right]^{-b_1-1} du_2 \\
&\quad = k_2 (1-u_3)^{q_3/2} c_2^{-(a+q_1/2+1)} (Q_1^2 - R_1^2 + c_2)^{-b_1-1} \int_0^1 u_2^{b_2} (1-u_2)^{a+q_2/2} \\
&\quad \times \left[u_2(Q_2^2 - R_2^2)/c_2 + 1\right]^{-(a+q_1/2+1)} \left[u_2(Q_2^2 - R_2^2)/(Q_1^2 - R_1^2 + c_2) + 1\right]^{-b_1-1} du_2 \\
&\quad = k_2 (1-u_3)^{q_3/2} c_2^{-(a+q_1/2+1)} (Q_1^2 - R_1^2 + c_2)^{-b_1-1} B(a + q_2/2 + 1, b_2 + 1) \\
&\qquad \times F_1(b_2 + 1, a + q_1/2 + 1, b_1 + 1, (N-q_0)/2; r, s),
\end{aligned}$$

where $k_2 = k_1 B^{-1}(a+1, b_1+1) B^{-1}(a+1, b_2+1) B(b_1+1, a+q_1/2+1)$; $c_2 = u_3(Q_3^2 - R_3^2) + 1 - \sum_{j=1}^{3} Q_j^2$; $r = (R_2^2 - Q_2^2)/c_2$; $s = (R_2^2 - Q_2^2)/(c_2 + Q_1^2 - R_1^2)$; $F_1(a, b, c, d; \cdot, \cdot)$ is one of Appell's bivariate hypergeometric functions

$$F_1(a, b, c, d; x, y) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \frac{(a)_{m+n} (b)_m (c)_n}{(d)_{m+n} m! n!} x^m y^n; \quad (a)_n = \frac{\Gamma(a+n)}{\Gamma(a)};$$

and the final equality follows from Gradshteyn and Ryzhik[36], p. 287, Eq. 3.211.

Now because $a + q_1/2 + 1 + b_1 + 1 = (N - q_0)/2$, we have

$$\begin{aligned}
&F_1\left[b_2 + 1, a + \frac{q_1}{2} + 1, b_1 + 1, \frac{N-q_0}{2}; r, s\right] = (1-s)^{a+\frac{q_1}{2}+1} \\
&\quad \times F\left[b_2 + 1, a + \frac{q_1}{2} + 1, \frac{N-q_0}{2}; \frac{r-s}{1-s}\right] = \left(1 - \frac{R_2^2 - Q_2^2}{c_2 + Q_1^2 - R_1^2}\right)^{a+q_1/2+1} \\
&\quad F\left[b_2 + 1, a + q_1/2 + 1, \frac{N-q_0}{2}; \frac{(R_2^2 - Q_2^2)(Q_1^2 - R_1^2)}{c_2(c_2 + Q_1^2 - R_1^2 + Q_2^2 - R_2^2)}\right],
\end{aligned}$$

by [*ibid*, p. 1054, Eq. 9.182#1] where $F(a,b,c;\cdot)$ is Gauss' hypergeometric function $F(a,b,c;x) = \frac{(a)_n (b)_n}{(c)_n n!} x^n$.

Finally integrating out u_3 (g_3), we arrive at the stated marginal density for FGAM

$$
\begin{aligned}
M_{FGAM}(\mathbf{y}) &= \frac{k_2 B(a+q_2/2+1, b_2+1)}{B(a+1, b_3+1)} \int_0^1 u_3^{b_3} (1-u_3)^{a+q_3/2} c_2^{a+q_1/2+1} \\
&\times (Q_1^2 - R_1^2 + c_2)^{-b_1-1} \left(1 - \frac{R_2^2 - Q_2^2}{c_2 + Q_1^2 - R_1^2}\right)^{a+q_1/2+1} \\
&\times F\left[b_2+1, a+q_1/2+1, \frac{N-q_0}{2}; \frac{(R_2^2-Q_2^2)(Q_1^2-R_1^2)}{c_2(c_2+Q_1^2-R_1^2+Q_2^2-R_2^2)}\right] du_3 \\
&= \frac{k_2 B(a+q_2/2+1, b_2+1)}{B(a+1, b_3+1)} \int_0^1 u_3^{b_3} (1-u_3)^{a+q_3/2} c_2^{a+q_1/2+1} \\
&\times (Q_1^2 - R_1^2 + c_2)^{-(N-q_0)/2} \left(c_2 + Q_1^2 + Q_2^2 - R_1^2 - R_2^2\right)^{a+\frac{q_1}{2}+1} \\
&\times F\left[b_2+1, a+\frac{q_1}{2}+1, \frac{N-q_0}{2}; \frac{(R_2^2-Q_2^2)(Q_1^2-R_1^2)}{c_2(c_2+Q_1^2-R_1^2+Q_2^2-R_2^2)}\right] du_3.
\end{aligned}
$$

BIBLIOGRAPHY

[1] A. Ait-Saïdi, F. Ferraty, R. Kassa, and P. Vieu. Cross-validated estimations in the single-functional index model. *Statistics*, 42(6):475–494, 2008.

[2] A Annamalai, C Tellambura, and Vijay K Bhargava. Equal-gain diversity receiver performance in wireless channels. *Communications, IEEE Transactions on*, 48(10):1732–1745, 2000.

[3] M. Asencio, G. Hooker, and H. O. Gao. Functional convolution models. Available at `http://www.bscb.cornell.edu/~hooker/EmissionsPaper.pdf`, last accessed July 7, 2013, 2012.

[4] D. Bates, M. Maechler, and B. Bolker. *lme4: Linear mixed-effects models using S4 classes, R package version 0.999999-2*, 2013. Available at `http://CRAN.R-project.org/package=lme4`.

[5] M. J. Bayarri, J. O. Berger, A. Forte, and G. García-Donato. Criteria for Bayesian model choice with application to variable selection. *The Annals of Statistics*, 40(3):1550–1577, 2012.

[6] C. Belitz and S. Lang. Simultaneous selection of variables and smoothing parameters in structured additive regression models. *Computational Statistics & Data Analysis*, 53(1):61–81, 2008.

[7] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[8] A. Buja, T. Hastie, and R. Tibshirani. Linear smoothers and additive models. *The Annals of Statistics*, pages 453–510, 1989.

[9] T. T. Cai and P. Hall. Prediction in functional linear regression. *The Annals of Statistics*, 34(5):2159–2179, 2006.

[10] H. Cardot, F. Ferraty, and P. Sarda. Spline estimators for the functional linear model. *Statistica Sinica*, 13(3):571–592, 2003. ISSN 1017-0405.

[11] G. Celeux, M. El Anbari, J. M. Marin, and C. P. Robert. Regularization in regression: comparing Bayesian and frequentist methods in a poorly informative situation. *Bayesian Analysis*, 7(2):477–502, 2012.

[12] D. Chen, P. Hall, and H. G. Müller. Single and multiple index functional regression models with nonparametric link. *The Annals of Statistics*, 39(3): 1720–1747, 2011.

[13] Z. Chen and D. B. Dunson. Random effects selection in linear mixed models. *Biometrics*, 59(4):762–769, 2003.

[14] N. N. Clark, M. Gautam, W. S. Wayne, G. W. Lyons, and G. J. Thompson. Heavy-duty vehicle chassis dynamometer testing for emissions inventory, air quality modeling, source apportionment, and air toxics emissions inventory. Technical Report CRC Rep. No. E55/59, Coordinating Research Coubncil, Inc. (CRC), 2007. Available at `http://www.crcao.com/reports/recentstudies2007/E-55-59/E-55_59_Final_Report_23AUG2007.pdf`.

[15] Nigel N Clark, Kuntal A Vora, Lijuan Wang, Mridul Gautam, W Scott Wayne, and Gregory J Thompson. Expressing cycles and their emissions on the basis of properties and results from other cycles. *Environmental science & technology*, 44(15):5986–5992, 2010.

[16] C. Crainiceanu, P. T. Reiss, J. Goldsmith, L. Huang, L. Huo, and F. Scheipl. *refund: Regression with Functional Data, R package version 0.1-7*, 2013. `http://CRAN.R-project.org/package=refund`.

[17] C. M. Crainiceanu and D. Ruppert. 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, 2004.

[18] C. Crambes, A. Kneip, and P. Sarda. Smoothing splines estimators for functional linear regression. *Annals of Statistics*, 37(1):35–72, 2009.

[19] I. D. Currie, M. Durban, and P. H. C. Eilers. Generalized linear array models with applications to multidimensional smoothing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(2):259–280, 2006.

[20] P. H. C. Eilers and B. D. Marx. Flexible smoothing with B-splines and penalties. *Statistical Science*, 11(2):89–121, 1996.

[21] Harold Exton. *Multiple hypergeometric functions and applications.* Ellis Horwood Chichester, 1976.

[22] C. Faes, J. T. Ormerod, and M. P. Wand. Variational Bayesian inference for parametric and nonparametric regression with missing data. *Journal of the American Statistical Association*, 106(495):959–971, 2011.

[23] L. Fahrmeir, T. Kneib, and S. Lang. Penalized structured additive regression for space-time data: a Bayesian perspective. *Statistica Sinica*, 14(3):731–762, 2004.

[24] Y. Fan and G. James. Functional additive regression. Available at `http://www-bcf.usc.edu/~gareth/research/FAR.pdf`, 2012.

[25] M. Febrero-Bande and M. Oviedo de la Fuente. Statistical computing in functional data analysis: The R package fda.usc. *Journal of Statistical Software*, 51(4):1–28, 2012. `http://www.jstatsoft.org/v51/i04/`.

[26] M. Febrero-Bande, P. Galeano, and W. González-Manteiga. Generalized additive models for functional data. *TEST*, 22(2):278–292, 2013.

[27] F. Ferraty, editor. *Recent advances in functional data analysis and related topics*, 2011. Springer.

[28] F. Ferraty and Y. Romain, editors. *The Oxford handbook of functional data analysis*, 2011. Oxford University Press.

[29] F. Ferraty and P. Vieu. *Nonparametric Functional Data Analysis: Theory and Practice*. Springer Verlag, 2006.

[30] E García-Portugués, W. González-Manteiga, and M. Febrero-Bande. A goodness-of-fit test for the functional linear model with scalar response. *Journal of Computational and Graphical Statistics*, page to appear, 2013.

[31] A. Gelman. Prior distributions for variance parameters in hierarchical models (comment on article by Browne and Draper). *Bayesian analysis*, 1(3):515–534, 2006.

[32] J. Goldsmith, J. Bobb, C. M. Crainiceanu, B. Caffo, and D. Reich. Penalized functional regression. *Journal of computational and graphical statistics*, 20 (4):830, 2011.

[33] J. Goldsmith, J. Bobb, C. M. Crainiceanu, B. Caffo, and D. Reich. Penalized functional regression. *Journal of Computational and Graphical Statistics*, 20 (4):830–851, 2011.

[34] J. Goldsmith, M. P. Wand, and C. Crainiceanu. Functional regression via variational Bayes. *Electronic Journal of Statistics*, 5:572–602, 2011.

[35] J. Goldsmith, C. M. Crainiceanu, B. Caffo, and D. Reich. Longitudinal penalized functional regression for cognitive outcomes on neuronal tract measurements. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61(3):453–469, 2012.

[36] I. S. Gradshteyn and I. M. Ryzhik. *Table of integrals, series and products*, volume 1. Academic Press, New York, fifth edition, 1994.

[37] S. Greven, C. M. Crainiceanu, H. Küchenhoff, and A. Peters. Restricted likelihood ratio testing for zero variance components in linear mixed models. *Journal of Computational and Graphical Statistics*, 17(4):870–891, 2008.

[38] S. Greven, C. M. Crainiceanu, B. Caffo, and D. Reich. Longitudinal functional principal component analysis. *Electronic Journal of Statistics*, 4:1022–1054, 2010. ISSN 1935-7524.

[39] C. Gu. *Smoothing spline ANOVA models*. Springer, 2002.

[40] S. Guillas and M. J. Lai. Bivariate splines for spatial functional regression models. *Journal of Nonparametric Statistics*, 22(4):477–497, 2010.

[41] R. K. S. Hankin. Special functions in R: introducing the gsl package. *R News*, 6(4), 2006.

[42] T. Hastie and R. Tibshirani. *Generalized Additive Models*. Chapman & Hall/CRC, 1990. ISBN 0412343908.

[43] R. A. Horn and C. R. Johnson. *Topics in matrix analysis*. Cambridge University Press, 1994.

[44] L. Horváth and P. Kokoszka. *Inference for functional data with applications*. Springer, 2012.

[45] T. S. Jaakkola and M. I. Jordan. Bayesian parameter estimation via variational methods. *Statistics and Computing*, 10(1):25–37, 2000.

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

[47] G. M. James and B. W. Silverman. Functional adaptive model estimation. *Journal of the American Statistical Association*, 100(470):565–577, 2005.

[48] G. M. James, T. J. Hastie, and C. A. Sugar. Principal component models for sparse functional data. *Biometrika*, 87(3):587–602, 2000.

[49] W. Jank and G. Shmueli. Functional data analysis in electronic commerce research. *Statistical Science*, 21(2):155–166, 2006.

[50] H. Jeffreys. *Theory of probability*. Clarendon Press Oxford, first edition, 1939.

[51] R. E. Kass and A. E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795, 1995.

[52] J. Kong, B. E. K. Klein, R. Klein, K. E. Lee, and G. Wahba. Using distance correlation and ss-anova to assess associations of familial relationships, lifestyle factors, diseases, and mortality. *Proceedings of the National Academy of Sciences*, 109(50):20352–20357, 2012.

[53] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[54] S. Lang and A. Brezger. Bayesian P-splines. *Journal of computational and graphical statistics*, 13(1):183–212, 2004.

[55] G. Lauricella. Sulle funzioni ipergeometriche a piu variabili. *Rendiconti del Circolo Matematico di Palermo*, 7:111–158, 1893.

[56] D. J. Lee and M. Durbán. P-spline ANOVA-type interaction models for spatio-temporal smoothing. *Statistical Modelling*, 11(1):49–69, 2011.

[57] D. J. Lee, M. Durbán, and P. Eilers. Efficient two-dimensional smoothing with p-spline anova mixed models and nested bases. *Computational Statistics & Data Analysis*, 61:22–37, 2013.

[58] E. Ley and M. F. J. Steel. Mixtures of g-priors for Bayesian model averaging with economic applications. *Journal of Econometrics*, 2012.

[59] E. Li, N. Wang, and N. Y. Wang. Joint models for a primary endpoint and multiple longitudinal covariate processes. *Biometrics*, 63(4):1068–1078, 2007. ISSN 1541-0420.

[60] Y. Li, N. Wang, and R. J. Carroll. Generalized functional linear models with semiparametric single-index interactions. *Journal of the American Statistical Association*, 105(490), 2010.

[61] H. Lian. Shrinkage estimation and selection for multiple functional regression. *Statistica Sinica*, 23(1):51–74, 2013.

[62] F. Liang, R. Paulo, G. Molina, M. A. Clyde, and J. O. Berger. Mixtures of g priors for Bayesian variable selection. *Journal of the American Statistical Association*, 103(481):410–423, 2008.

[63] B. Liu and H. G. Müller. Functional data analysis for sparse auction data. *Statistical Methods in eCommerce Research*, pages 269–290, 2008.

[64] G. Marra and S. N. Wood. Practical variable selection for generalized additive models. *Computational Statistics & Data Analysis*, 55(7):2372–2387, 2011.

[65] G. Marra and S. N. Wood. Coverage properties of confidence intervals for generalized additive model components. *Scandinavian Journal of Statistics*, 39(1):53–74, 2012.

[66] Y. Maruyama and E. I. George. Fully Bayes factors with a generalized g-prior. *The Annals of Statistics*, 39(5):2740–2765, 2011.

[67] B. D. Marx and P. H. C. Eilers. Direct generalized additive modeling with penalized likelihood. *Computational Statistics & Data Analysis*, 28(2):193–209, 1998.

[68] B. D. Marx and P. H. C. Eilers. Multidimensional penalized signal regression. *Technometrics*, 47(1):13–22, 2005.

[69] P. McCullagh and J. A. Nelder. *Generalized Linear Models*. Chapman & Hall/CRC, second edition, 1989.

[70] M. McLean, G. Hooker, A. M. Staicu, F. Scheipl, and D. Ruppert. Functional generalized additive models. *Journal of Computational and Graphical Statistics*, to appear. Available at `http://amstat.tandfonline.com/doi/full/10.1080/10618600.2012.729985`.

[71] X. L. Meng and W.H. Wong. Simulating ratios of normalizing constants via a simple identity: a theoretical exploration. *Statistica Sinica*, 6:831–860, 1996.

[72] R. D. Morey and J. N. Rouder. *BayesFactor: Computation of Bayes factors for common designs, R package version 0.9.4*, 2013. Available at `http://CRAN.R-project.org/package=BayesFactor`.

[73] S. Mori. *Introduction to Diffusion Tensor Imaging*. Elsevier Science, 2007. ISBN 0444528288.

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

[75] H. G. Müller and U. Stadtmüller. Generalized functional linear models. *Annals of Statistics*, 33(2):774–805, 2005.

[76] H. G. Müller and F. Yao. Functional additive models. *Journal of the American Statistical Association*, 103(484):1534–1544, 2008. ISSN 0162-1459.

[77] H. G. Müller, Y. Wu, and F. Yao. Continuously additive models for nonlinear functional regression. *Biometrika*, to appear. doi: 10.1093/biomet/ast004. Available at `http://biomet.oxfordjournals.org/content/early/2013/03/26/biomet.ast004.full.pdf+html`.

[78] R. M. Neal. Slice sampling. *Annals of statistics*, pages 705–741, 2003.

[79] D. Nychka. Bayesian confidence intervals for smoothing splines. *Journal of the American Statistical Association*, 83(404):1134–1143, 1988.

[80] J. T. Ormerod and M. P. Wand. Explaining variational approximations. *The American Statistician*, 64(2):140–153, 2010.

[81] H. D. Patterson and R. Thompson. Recovery of inter-block information when block sizes are unequal. *Biometrika*, 58(3):545–554, 1971.

[82] D. K. Pauler, J. C. Wakefield, and R. E. Kass. Bayes factors and approximations for variance component models. *Journal of the American Statistical Association*, 94(448):1242–1253, 1999.

[83] J. Peng and D. Paul. 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, 2009.

[84] J. C. Pinheiro and D. M. Bates. *Linear mixed-effects models: basic concepts and examples*. Springer, 2000.

[85] N. G. Polson and J. G. Scott. Shrink globally, act locally: sparse Bayesian regularization and prediction. *Bayesian Statistics*, 9:76, 2010.

[86] N. G. Polson and J. G. Scott. On the half-Cauchy prior for a global scale parameter. *Bayesian Analysis*, 7(4):887–902, 2012.

[87] N. G. Polson and J. G. Scott. Local shrinkage rules, lévy processes and regularized regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2012.

[88] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2012. Available at `http://www.R-project.org/`.

[89] J. O. Ramsay and C. J. Dalzell. Some tools for functional data analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 53(3):539–572, 1991. ISSN 0035–9246.

[90] J. O. Ramsay and B. W. Silverman. *Applied functional data analysis: methods and case studies*. Springer, 2002.

[91] J. O. Ramsay and B. W. Silverman. *Functional Data Analysis*. Springer, second edition, 2005.

[92] J. O. Ramsay, G. Hooker, and S. Graves. *Functional Data Analysis with R and MATLAB*. Springer, 2009.

[93] J. O. Ramsay, H. Wickham, S. Graves, and G. Hooker. *fda: Functional Data Analysis, R package version 2.3.6*, 2013. Available at `http://CRAN.R-project.org/package=fda`.

[94] D. S. Reich, S. A. Smith, K. M. Zackowski, E. M. Gordon-Lipkin, C. K. Jones, J. A. D. Farrell, S. Mori, P. van Zijl, and P. A. Calabresi. Multiparametric magnetic resonance imaging analysis of the corticospinal tract in multiple sclerosis. *Neuroimage*, 38(2):271–279, 2007. ISSN 1053–8119.

[95] P. T. Reiss and R. T. Ogden. Smoothing parameter selection for a class of semiparametric linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(2):505–523, 2009.

[96] Maria L. Rizzo and Gabor J. Szekely. *energy: E-statistics (energy statistics), R package version 1.6.0*, 2013. Available at `http://CRAN.R-project.org/package=energy`.

[97] A. Rodríguez, D. B. Dunson, and A. E. Gelfand. Bayesian nonparametric functional data analysis through density estimation. *Biometrika*, 96(1):149–162, 2009.

[98] J. N. Rouder, R. D. Morey, P. L. Speckman, and J. M. Province. Default bayes factors for ANOVA designs. *Journal of Mathematrical Psychology*, 56 (5):356–374, 2012.

[99] D. Ruppert, M. P. Wand, and R. J. Carroll. *Semiparametric regression*. Cambridge University Press, 2003.

[100] D. Sabanés Bové, L. Held, and G. Kauermann. Hyper-g priors for generalised additive model selection with penalised splines. In D. Conesa, A. Forte, A. Lopez-Quilez, and F. Munoz, editors, *Proceedings of the 26th International Workshop on Statistical Modelling*, pages 538–543. Valencia, 2011. ISBN 978–84–694–5129–8.

[101] B. R. Saville and A. H. Herring. Testing random effects in the linear mixed model using approximate Bayes factors. *Biometrics*, 65(2):369–376, 2008.

[102] F. Scheipl, S. Greven, and H. Küchenhoff. Size and power of tests for a zero random effect variance or polynomial regression in additive and linear mixed models. *Computational Statistics & Data Analysis*, 52(7):3283–3299, 2008.

[103] F. Scheipl, A. M. Staicu, and S. Greven. Additive mixed models for correlated functional data. Available at `http://arxiv.org/abs/1207.5947`, 2012.

[104] S. G. Self and K. Y. Liang. Asymptotic properties of maximum likelihood estimators and likelihood ratio tests under nonstandard conditions. *Journal of the American Statistical Association*, 82(398):605–610, 1987.

[105] Z. Shang and L. Peng. Bayesian functional selection in ultrahigh-dimensional nonparametric additive models. Available at `http://arxiv.org/abs/1307.0056`, 2013.

[106] J. Q. Shi and T. Choi. *Gaussian process regression analysis for functional data*. CRC Press, 2011.

[107] Q. Shi and Y. Karasawa. Some applications of lauricella hypergeometric function f_a in performance analysis of wireless communications. *IEEE Communications Letters*, 16(5):581–584, 2012.

[108] S. Sinharay and H. S. Stern. An empirical comparison of methods for computing Bayes factors in generalized linear mixed models. *Journal of Computational and Graphical Statistics*, 14(2):415–435, 2005.

[109] G. Smyth, Y. Hu, P. Dunn, and B. Phipson. *statmod: Statistical Modeling, R package version 1.4.14*, 2011. `http://CRAN.R-project.org/package=statmod`.

[110] H. M. Srivastava and P. W. Karlsson. *Multiple Gaussian hypergeometric series*. E. Horwood, 1985.

[111] N. M. Steen, G. D. Byrne, and E. M. Gelbard. Gaussian quadratures for the integrals $\int_0^\infty \exp(-x^2)f(x)dx$ and $\int_0^b \exp(-x^2)f(x)dx$. *Mathematics of Computation*, 23(107):661–671, 1969.

[112] D. O. Stram and J. W. Lee. Variance components testing in the longitudinal mixed effects model. *Biometrics*, 50(4):1171–1177, 1994.

[113] B. J. Swihart, J. Goldsmith, and C. M. Crainiceanu. Restricted likelihood ratio tests for functional effects in the functional linear model. Technical Report 247, Johns Hopkins University, Dept. of Biostatistics, 2013. Available at `http://biostats.bepress.com/jhubiostat/paper247`.

[114] G. J. Székely and M. L. Rizzo. The distance correlation t-test of independence in high dimension. *Journal of Multivariate Analysis*, 117:193–213, 2013.

[115] Gábor J Székely, Maria L Rizzo, and Nail K Bakirov. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794, 2007.

[116] W. J. Vetter. Matrix calculus operations and Taylor expansions. *SIAM review*, 15(2):352–369, 1973.

[117] G. Wahba. Bayesian "confidence intervals" for the cross-validated smoothing spline. *Journal of the Royal Statistical Society: Series B (Methodological)*, pages 133–150, 1983.

[118] M. P. Wand and J. T. Ormerod. Continued fraction enhancement of Bayesian computing. *Stat*, 1(1):31–41, 2012.

[119] C. Wang and D. Blei. Variational inference in nonconjugate models. *Journal of Machine Learning Research*, 14:899–925, 2013.

[120] L. Wang, G. Chen, and H. Li. Group scad regression analysis for microarray time course gene expression data. *Bioinformatics*, 23(12):1486–1494, 2007.

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

[122] Y. Wang. *Smoothing Splines: Methods and Applications*. CRC Press, 2011.

[123] Y. Wang and H. Chen. On testing an unspecified function through a linear mixed effects model with multiple variance components. *Biometrics*, 68(4): 1113–1125, 2012. doi: 10.1111/j.1541-0420.2012.01790.x.

[124] S. N. Wood. Stable and efficient multiple smoothing parameter estimation for generalized additive models. *Journal of the American Statistical Association*, 99(467):673–686, 2004.

[125] S. N. Wood. On confidence intervals for generalized additive models based on penalized regression splines. *Australian & New Zealand Journal of Statistics*, 48(4):445–464, 2006.

[126] S. N. Wood. *Generalized Additive Models: An Introduction with R.* CRC Press, 2006.

[127] S. N. Wood. Low-rank scale-invariant tensor product smooths for generalized additive mixed models. *Biometrics*, 62(4):1025–1036, 2006.

[128] S. N. Wood. 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, 2011.

[129] S. N. Wood, F. Scheipl, and J. J. Faraway. Straightforward intermediate rank tensor product smoothing in mixed models. *Statistics and Computing*, 23(3):341–360, 2013. Available at `http://opus.bath.ac.uk/28333/`.

[130] F. Yao and T. Lee. Penalized spline models for functional principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):3–25, 2005.

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

[132] Fang Yao and Hans-Georg Müller. Functional quadratic regression. *Biometrika*, 97(1):49–64, 2010.

[133] M. Yuan and T. T. Cai. A reproducing kernel Hilbert space approach to functional linear regression. *The Annals of Statistics*, 38(6):3412–3444, 2010.

[134] A. Zellner. On assessing prior distributions and bayesian regression analysis with g-prior distributions. In P. K. Goel and A. Zellner, editors, *Bayesian*

inference and decision techniques: Essays in Honor of Bruno De Finetti, pages 233–243. North Holland, 1986.

[135] A. Zellner and A. Siow. Posterior odds ratios for selected regression hypotheses (with discussion). In J. M. Bernardo, M. H. DeGroot, D. V. Lindley, and A. F. M. Smith, editors, *Bayesian Statistics*, pages 585–603. University Press, 1980.

[136] J. T. Zhang. *Analysis of Variance For Functional Data*. CRC Press, 2013.