Published in final edited form as:

Stat Modelling. 2017 February ; 17(1-2): 59–85. doi:10.1177/1471082X16681875.

Comparison and Contrast of Two General Functional Regression Modeling Frameworks

Jeffrey S. Morris

The University of Texas, MD Anderson Cancer Center, Unit 1411, PO Box 301402, Houston, TX 77230-1402

Abstract

In this article, Greven and Scheipl describe an impressively general framework for performing functional regression that builds upon the generalized additive modeling framework. Over the past number of years, my collaborators and I have also been developing a general framework for functional regression, functional mixed models, which shares many similarities with this framework, but has many differences as well. In this discussion, I compare and contrast these two frameworks, to hopefully illuminate characteristics of each, highlighting their respective strengths and weaknesses, and providing recommendations regarding the settings in which each approach might be preferable.

Keywords

Bayesian modeling; Functional data analysis; Functional regression; Linear Mixed Models

1 Introduction

In recent decades, functional data of various forms and characteristics have been increasingly encountered in many areas of research, and the statistical field of Functional Data Analysis (FDA, Ramsay and Silverman, 2007) has emerged to develop methods to analyze such data. Much of the work done in FDA has involved functional regression, which includes functional response regression (function-on-scalar), functional predictor regression (scalar-on-function), and function-on-function regression. In this article, Greven and Scheipl describe an impressively general framework for performing functional regression that encompasses all three of these areas, and can be described as containing many of the existing methods as special cases. The formulation and building up of this general framework has been a topic in this research group's work over the past few years, introducing much of this structure for Gaussian functions largely represented by splines and fit using generalized additive model (GAM) software in Scheipl, Staicu and Greven (2015), incorporating functional principal components (fPC) to flexibly model sparse, irregularly sampled outcomes in Cederbaum, et al. (2015), extending to generalized outcomes in Scheipl, Gertheiss, and Greven (2016), and introducing a new boosting-based fitting procedure that allows extension to robust functional regression and confers other benefits in Brockhaus, et al. (2015). Other specific work has been done developing details for additive scalar-on function models (McLean et al. 2014) and function-on-function regression

(Ivanescu, et al. 2015; Scheipl and Greven 2016; Brockhaus, et al. 2016), and undoubtedly this productive group will continue to further develop this framework in the coming years.

I congratulate the authors for this well-written paper and impressive body of work. The framework they present here is very general, encompassing many of the existing functional regression models in the literature, especially those based on splines and functional principal components. Recent work has extended its applicability beyond mean regression of Gaussian functions to generalized functional regression with different types of functional responses (e.g. exponential families) and performing robust (median) regression and other quantile regression. They introduce an elegant notation sufficiently broad so as to accommodate an extremely broad array of model components, including scalar or functional predictors with constant or functional coefficients, with the form of each predictor being linear (parametric) or smooth nonparametric. Their framework represents the coefficients with tensor products of basis coefficients in the covariates and functional domains, and with accompanying penalty terms that allow separable penalization both in the covariate and functional domains. They have incorporated this framework into freely available general R packages that contain many useful functions for model building, diagnostics, and summaries. This work is already making a strong impact on the field and promises to continue to do so.

Over the past number of years, my collaborators and I have also been developing a general framework for functional regression that shares many similarities with this framework but has many differences as well. This work started with Morris, et al. (2003), the general framework was first introduced in Morris and Carroll (2006), and it has been developed in various ways through a series of subsequent papers (most published, some currently under review). Morris, et al. (2006) introduced methods to deal with missing data, Morris, et al. (2008) applied the approach to high dimensional mass spectrometry data and introduced methods for flagging functional regions while controlling Bayesian false discovery rate (FDR). Morris, et al. (2011) extended to functions on higher dimensional domains, e.g. images, discussed use of basis functions other than wavelets, applied the method to functional data on a grid of $>1/2$ million, and introduced joint thresholding approaches to find a parsimonious subset of basis functions that yield near-lossless representations for all observed functions. Zhu, Brown and Morris (2011) extended the framework to perform robust functional regression, and Zhu, Brown and Morris (2012) demonstrated how to perform functional discriminant analysis within this framework. Morris (2013) illustrated how the framework can be used to analyze proteomics data of various types, and demonstrated that this functional regression approach found discoveries missed by the more commonly used peak detection-based analysis workflows. Martinez, et al. (2013) adapted the framework to analyze nonstationary acoustic signals by modeling the spectrogram as a function in this framework, and introduced registration approaches to register acoustical signals of differing length without distortion in the frequency domain. Lee and Morris (2015) applied the framework to whole-genome methylation data, and demonstrated that modeling these markers as functions yields greater power for discoveries than modeling markers independently. Meyer, et al. (2015) demonstrated how to perform function-on-function regression with this framework, and introduced Sim-BaS scores, statistics that can be used to flag local regions of functions as significant while controlling the expected

experiment wise error rate (EER). Zhang, et al. (2016a) extended this framework to model functions observed on a spatial lattice, introducing functional conditional autoregressive processes, and Zhu, et al. (2016c) presented a general functional regression model for spatially or temporally correlated functions and introduced predictive likelihood-based model selection techniques. Lee, et al. (2016) demonstrated how to perform semiparametric functional regression with terms nonparametric in both x and t using this framework, and how it can accommodate longitudinally correlated functional data. Zhu, et al. (2016a) developed unified methods for functional regression and discrimination for sonar-terrain data, and Zhu, et al. (2016b) developed methods for multivariate functional response regression for fluorescence spectroscopy data. Zhang, et al. (2016b) introduced functional graphical models that can be used to model multivariate functional data.

All of this work can be described as part of a single unified framework, but given that it has been presented in separate pieces over a number of publications, its broad scope and generality may not be clear to many researchers. We refer to this general framework as Functional Mixed Models (FMM). Although it could be fit using frequentist or Bayesian approaches, for inferential reasons we have built the framework using a unified Bayesian approach. I believe it would be valuable to compare and contrast these two frameworks, to hopefully illuminate characteristics of both frameworks and perhaps stimulate future research.

In this discussion, I first summarize in my own words some of salient features of the framework and approach broadly laid out by Greven and Scheipl in this paper, highlight its major strengths and mention what I view as some of its potential limitations and shortcomings, particularly in the setting of complex, high dimensional functional data. Given that the scope of the Functional Linear Array Model (FLAM, Brockhaus et al. 2015) appears to most broadly cover the framework presented in this paper, for convenience I will refer to Greven and Scheipl's framework as FLAM throughout this discussion, even if this is somewhat of an oversimplification. Next, I summarize our FMM framework in its generality, hoping to give the readers an understanding of its scope, strengths, and limitations. Finally, I will compare and contrast various aspects of these two frameworks, highlighting the strengths and weaknesses of each, and providing recommendations regarding the settings in which each approach might be preferable.

2 Summary of Functional Linear Array Model Framework

2.1 Description of Framework

Core regression model—The core regression model underlying the framework laid out in this article (model 2.1) is a generalized additive model (GAM) in which a generic response Y is regressed on a set of predictors X through a series of additive terms $h_j(x)$, $j = 1, \dots, J$, with a transformation function ξ defining the transformation over which the regression is defined:

$$\xi(Y|X=x)=h(x)=\sum_{j=1}^J h_j(x). \quad (1)$$

This general definition introduces a notation that is able to include a vast array of different types of regression models, handling many types of responses and predictors, and many different approaches to regression. The additive terms allow predictors that are scalar or functions, continuous or discrete, and parametric (linear) or smooth nonparametric in form, and given the observation that x could involve multiple predictors, could include interactions of any type. Defining the transformation function as an expectation yields classical mean regression models for continuous responses, but other choices can be used to handle other types of responses, e.g. link functions for exponential family responses, different types of objective functions, e.g. median chosen for classical robust regression or check functions for quantile regression, and other types of regression, e.g. scale or shape regression.

Note that there is nothing explicitly functional about this core regression model – it is essentially a GAM. However, as the authors show, this framework can be used for functional regression by representing any functional responses, predictors, or coefficients with a basis function expansion that provides a finite dimensional representation of the infinite dimensional functions. Their approach is to incorporate the basis functions into the regression design matrices, which allows great convenience and forward compatibility by enabling use of existing software for fitting scalar regression models, but also has some drawbacks, especially for data sets involving larger and/or more complex functional data sets, as I will later point out. A vast majority of their work has utilized spline bases of various types, with some recent work utilizing eigenfunctions from functional principal component (fPC) decompositions for representing random effects, but in principle could be used with any basis function.

Basis Representations—In general, they use tensor products of basis functions to represent the additive terms. For example, in (3.2) they present a general basis representation:

$$h_j(x, t) = (\mathbf{b}_{xj}(x)' \otimes \mathbf{b}_{Yj}(t)') \boldsymbol{\theta}_j. \quad (2)$$

This allows for terms that vary over x and t , such as functionally varying coefficients for functional response regression models. For additive terms varying only in x such as non-functionally-varying coefficients for scalar predictors, $\mathbf{b}_{Yj}(t)$ can be made constant in t , for terms varying only in t such as intercepts for functional responses, $\mathbf{b}_{xj}(x)$ can be made constant in x . The non-constant basis functions can be parametric, e.g. linear, or can be the basis for nonparametric smooth function like a spline to allow flexible nonparametric regression in both x and t . This tensor representation is computationally convenient and very flexible, allowing many of the types of terms one would want to include in functional response, functional predictor, and function-on-function regression models. However, as

mentioned in Section 3.2, the separability assumptions underlying the kronecker product poses some limitations, so in some settings such as historical functional or concurrent functional terms they propose a more general row tensor product, e.g. $(\mathbf{b}_X(x, t)') \odot \mathbf{b}_Y(t')$.

Penalization—To induce smoothness/regularity, they introduce quadratic penalties for each additive term, defined in (3.3) as Kronecker penalties that are additive in x and t :

$$P_j = \lambda_{xj} P_{xj} \otimes I_{K_{Yj}} + \lambda_{Yj} I_{K_{xj}} \otimes P_{Yj}. \quad (3)$$

The scalars λ_{xj} and λ_{Yj} are marginal smoothing parameters corresponding to the basis functions $\mathbf{b}_X(x)$ and $\mathbf{b}_Y(t)$, respectively, and the matrices P_{xj} and P_{Yj} are $K_{xj} \times K_{xj}$ and $K_{Yj} \times K_{Yj}$ corresponding penalty matrices, where K_{Yj} and K_{xj} are the number of basis functions in the basis expansions $\mathbf{b}_X(x)$ and $\mathbf{b}_Y(t)$, respectively. As pointed out by the authors, this quadratic penalty is mathematically equivalent to assuming a Gaussian process prior on the additive term with semiparametric covariance kernel given by (3.4):

$$\text{cov}(h_j(x, t), h_j(x', t')) = (\mathbf{b}_{xj}(x) \otimes \mathbf{b}_{Yj}(t))' P_j^{-1} (\mathbf{b}_{xj}(x') \otimes \mathbf{b}_{Yj}(t')). \quad (4)$$

The authors specifically illustrate how to utilize this penalty notation to fit P-splines and fPC bases. P-splines can be used to induce smoothness in the predictor x and/or functional index t by specifying difference penalties for $P_{\bullet}$ and interpreting the $\lambda_{\bullet}$ as smoothing parameters. When using fPC bases for $\mathbf{b}_Y(t)$, they specify the diagonals of P_{Yj} to be the estimated eigenvalues and set $\lambda_{Yj} \equiv 1$, which obviates the need to estimate a smoothing parameter.

Model Fitting and Inference—They fit their models either using existing generalized additive models software (Wood, 2016) in the R package *refund* (Huang et al. 2016) or through a component-wise gradient boosting (Buhlmann and Hothorn, 2007) in the R package *FDboost* (Brockhaus, 2016).

Their overall modeling approach is to column-stack the entire set of observed functional data into one long vector and then analyze this vector using existing regression software designed for scalar data. Given that the formulation of their general framework is a GAM, they are able to utilize existing GAM software package *mgcv* (Wood, 2016). Specifically, they use the function *gam* within this package, which uses penalized likelihood to fit the GAMs, and yields likelihood-based inference providing penalized likelihood estimates of all model parameters, plus asymptotic standard errors and pointwise confidence intervals. The package includes various methods for estimating smoothing parameters λ , including GCV, AIC, or REML.

Their modeling framework can also be viewed as a series of iterative penalized optimization problems with L2 penalties. Component-wise gradient boosting (Buhlmann and Hothorn, 2007) is an iterative computational approach for fitting penalized optimization problems that is fast and also provides a type of variable selection through early stopping and/or stability selection. Its various objective functions can be used to yield mean (squared error), median

(absolute error), or quantile (check function) regression. The authors mention that the smoothing parameters λ are chosen to be fixed values such that the degrees of freedom are the same for each model term $h_j(x, t)$. This procedure provides only point estimates, so inference must be obtained using resampling-based approaches.

2.2 Strengths of FLAM

Vast Scope of Modeling Framework—The modeling framework described in this paper is impressively general, largely built upon the GAM framework that has been extensively developed in recent years. Their framework encompasses the various types of functional regression while allowing many different types of responses, predictors, coefficient types, objective functions, and sampling grids.

The framework can be utilized for functional response regression, functional predictor regression, or function-on-function regression. It allows continuous Gaussian and exponential family responses that can be either scalar or functional-valued random variables. It allows any number of additive predictor terms, and each predictor term can involve scalar covariates with scalar or functional coefficients or functional covariates with scalar or functional coefficients, that when paired with functional responses can accommodate unconstrained, historical, and concurrent function-on-function models. These additive terms can be used to represent population-level or *fixed effect functions* as well as subject- or cluster-specific terms, or *random effect functions*. These coefficients can be parametric (e.g. linear) in x and t , or can be allowed to be nonparametric and smooth in x and/or t to yield semiparametric models. Classic mean regression can be done, with appropriate link functions for non-Gaussian responses, or when the *FDBOost* fitting approach is used, alternative objective functions can be used for the regression to yield robust (median) or quantile regression. The model can be fit to functional data with sparse, irregular grids that vary across observed sampling units, or if a common grid is observed for each sampling unit, then computational approaches that save time and memory allocation are available.

Elegant Notation for Basis Functions and Penalization—The authors introduce an elegant notation for basis expansions and penalization for the additive terms in the model that can accomplish spline smoothing in both x and t in the additive terms as well as incorporate fPC decompositions to flexibly accommodate random effect structure. The specific choice of basis functions and/or penalty is allowed to vary across the different additive terms, allowing flexibility in the representation and smoothing of the various terms in the model.

Natural Connection to Many Existing Methods—The modeling framework and basis function/ penalization strategy employed are presented in sufficient generality to encompass many of the functional regression methods in existing literature, especially those based on spline and fPC representations. Thus, their general notation can help unify thinking about these types of models, which promises to enhance researchers' comprehension of existing methods and stimulate the development of new ones.

Use of Existing Software—The strategy of building on existing software has substantial benefits. Of course, it is convenient to not have develop their own general software from scratch, which allows for rapid development and dissemination of general software for their framework. The existing GAM software is impressively general, well tested and vetted, and has powerful capabilities in terms of model checking and summaries, which are advantages automatically conferred to the authors' functional regression framework. As illustrated by their recent incorporation of boosting approaches to speed model fitting, their approach is naturally forward compatible with other advancements made for scalar regression models.

2.3 Potential Limitations of FLAM

While impressively general, some of the features of this framework that allow for convenient calculations and use of existing software, including the representation of the functional data as a single long vector, the separable kronecker structure on the basis functions and penalties, and the L2-based penalization strategy, also lead to some drawbacks and limitations, especially in terms of types of functional data that can be adequately modeled by this framework. In this section, I describe some of these drawbacks.

Dependence on L2 Penalization May Limit Flexibility—As highlighted in the review article by Morris (2015), a hallmark of existing functional regression methods is representation with basis functions and regularization through penalization. The form of penalization can be L2-based, which is natural for spline smoothing, or sparsity or L1-based, which is natural for wavelets, PC regression, and other multiresolution basis functions. The reliance on L2 penalties in the present framework fits naturally with splines, but does not make as much sense for other basis functions for which sparsity may be the preferable mode of regularization. Splines do an excellent job when modeling simple, smooth functions, and are also especially useful for modeling smoothness in sparsely observed functional data as in longitudinal studies. However, other types of functional data are complex with irregularities such as change points, flat regions, or spikes that are not well-modeled by splines, but require more spatially adaptive basis functions.

Also, the authors mention that when using the component-wise gradient boosting, they choose the smoothing parameters to be fixed values to make the degrees of freedom the same for each modeling term. It is not clear to me why this constraint is needed, but this seems like it would significantly limit the ability of the modeling framework to flexibly penalize each additive term.

Scalability to Enormous Dimensional Functional Data—Given a set of functional data $Y_i(t); i = 1, \dots, N$ observed for values of $t \in \{t_{11}, \dots, t_{JT_j}\}$, the approach presented in this paper models the stacked vector of length $\sum_{i=1}^N T_i$, which is of length N^*T if each function is observed on a common grid of size T . Given J additive predictors, this vector is regressed on a design matrix of dimension $K = \sum_{j=1}^J K_j$, where $K_j = K_{Xj} \times K_{Yj}$. For high dimensional functional data observed on extremely large grids (e.g. in some settings T can be 10,000 to millions or more), the fitting of this single large regression model may become computationally untenable in terms of speed and memory management. One aspect of this

strategy that limits scalability is that the order of calculations appears to be quadratic in K_j , which can be problematic since many large functional data sets are also complex and tend to require a relatively high number of basis functions K_{Y_j} to represent the functional structure in t . As mentioned by the authors in Section 5.3, for a limited subset of models they can fit their design matrix in a separable fashion, which reduces memory demands and would bring computational demands down from $O(K_j^2)$ to $O(K_{xj}^2) + O(K_{Yj}^2)$, which helps but may still be prohibitive for complex data that require large K_{Yj} .

Handling of Within-Function Correlations—One of the key hallmarks of FDA is intrafunctional correlation, i.e. that measurements at nearby functional domain locations should be correlated with each other. This motivates the standard strategy of regularizing and smoothing functional regression coefficients, and is also why functional response models should always (in my strongly held opinion) accommodate this correlation in the curve-to-curve deviations and account for it in estimation and inference. While the penalized basis function modeling handles regularization, it is only through specification of an additive term dedicated to smooth curve-to-curve deviations for each observed function that this modeling framework can handle intrafunctional correlation in the curve-to-curve deviations.

The residual error terms are never formally defined in the modeling framework, but in examples they are always independent across t . This means the residual errors can only be plausibly interpreted as measurement errors, not curve-to-curve residual deviations, since they do not accommodate the correlations across t that characterize functional data. Thus, in order to induce intrafunctional correlation, one must define an additive term h_j including N random effect functions for the curve-to-curve deviations, $B_{\lambda}(t)$, as described in Section 3.1.2. Given a choice of K_{Y_j} basis functions, this additive term will contain $N \times K_{Y_j}$ coefficients, which can add a great deal of computational burden to the fitting procedure whenever N or K_{Y_j} are large, limiting its scalability to large data sets.

The choice of basis functions and penalty induces a specific covariance structure, with $\text{cov}(B_i(t), B_i(t')) = \lambda_{Y_j}^{-1} \mathbf{b}_{Y_j}(t)' P_{Y_j}^- \mathbf{b}_{Y_j}(t')$. It is important to consider the form of this induced covariance structure, since it determines the assumed characteristics of the curve-to-curve deviations. If splines are used, then the structure of this intrafunctional covariance matrix is fully determined by the basis and penalty choice up to a single scalar constant λ_{Y_j} , which in my experience does not provide enough flexibility to capture the form of the curve-to-curve deviations for many data sets. fPCs provide a much greater level of flexibility here, so are preferred in settings where the sample sizes are sufficiently large to well-estimate the fPCs.

It would be instructive for the authors to provide a more detailed discussion of the intrafunctional covariance structures induced by their modeling framework.

Handling of Between-Function Correlations—In many studies, the observed functions are not independent, but instead sampled in such a way to induce some form of interfunctional correlation. For example, the functions themselves may be sampled longitudinally, spatially, or within a nested, multilevel design, or one may observe

multivariate functional data for which a set of q different functions are observed for each experimental unit. It is important to take such correlation into account in order to obtain efficient estimation and accurate inferential summaries. The framework presented here appears to have relatively limited provisions for accounting for such correlations.

The paper does not mention any option to incorporate parametric covariance structures such as those used in time series (e.g. AR(p) models) and spatial (e.g. Matern) models. Although not discussed in this article, it seems that one could use the penalty matrix P_{Y_j} to induce spatial or temporal correlation across the different curves, but would have to be done by completely prespecifying the entire penalty matrix, including fixed values for any covariance parameters. A grid search could be done across different values of the covariance parameters, but this would add a great deal of computational intensity and would restrict consideration to a rather coarse grid of potential values for covariance parameters. Alternatively, within the *mgcv* package one could fit the model using the full mixed-model based *gamm* function, which might contain these covariance structures, rather than the penalized likelihood-based *gam* function featured in this article. However, as mentioned in the documentation for *mgcv*, *gamm* is not as fast or stable as *gam* for models with large number of random effect functions, so it is not clear whether it would be a viable option for regression for many spatially or longitudinally correlated functional data sets that tend to be rather large.

Handling of Random Effect Functions—In linear mixed models, random effects can be used as a convenient mechanism to induce correlation between observations.

Analogously, in functional mixed effect models, random effect functions can be used to induce correlation between functions. As mentioned in Section 3.1.2, additive terms containing random effect functions can be defined within this modeling framework. For example, if we have multiple subsampled functions for each subject, an additive term could be defined that includes a random effect function for each subject. If we have multiple functions for each subject observed longitudinally over time, an additive term can be specified containing random slope functions for each subject. If these so-called random effect functions are treated as *random effects* in the truest sense of the word, then these additive terms would account for the interfunctional correlation. However, it is not clear whether the implementation of this framework described in this paper treats these functions as truly random effects, or whether they are effectively just smooth additive terms, not substantially different from other fixed effect functions in the model. I will elaborate on this subtle point.

In classic linear mixed models (LMM), random effects are characterized by two properties: (1) They represent randomly sampled observations from some population, (2) there is not typically direct interest in the random effect units themselves, but rather in performing inference to the population from which they were sampled. In order to make inference on the population from which the units were drawn, when fitting LMM one effectively fits a marginalized version of the model with the random effects integrated out, $Y \sim \mathcal{N}(XB, \Sigma)$, with covariance Σ depending on the random effects and residual errors. The fixed effect coefficients B are effectively estimated by a generalized least squares type formula, e.g. $\hat{B} = (X' \Sigma^{-1} X)^{-1} X' \Sigma^{-1} Y$, that automatically re-weights observations according to the

interobservational covariance structure, that is also taken into account for inference given that $SE(\hat{B}) = (X' \Sigma^{-1} X)^{-1}$. In this way, LMMs automatically account for interobservational correlation in estimation and inference. This inference accounts for all sources of variability in the data, e.g. in the case of iid subject-level random effects, the inferential summaries in the model account for both between-subject and within-subject variability in the data. If subject-specific random effects were in fact modeled as fixed effects (i.e. not marginalized out but conditioned upon), then the subsequent inference would not account for the subject-to-subject variability, the only source of stochastic variability in the model would be the residual errors, and the model would not effectively make inference on the population from which the subjects were drawn.

For the framework introduced in this paper, if the more computationally intensive *gam* procedure is used to fit the model, then it seems that in this way the random effects would be treated as truly random in the model fitting since the computational engine of this procedure is LMMs. With the penalized likelihood-based *gamm* or stepwise boosting (*FDBOost*) procedures, however, it is not clear to me whether this is the case. It is possible that the more computationally efficient modeling approach chosen in these functions sacrifices this inferential interpretation of random effects, does not account for their corresponding sources of variability or interfunctional correlation, and thus may sacrifice efficiency in estimation and inference when strong interfunctional correlation is present. It would be interesting to see a more detailed discussion of this important but tricky topic from the authors.

Inferential Limitations—When fit with *mgcv*, the model fitting provides asymptotic standard errors and pointwise confidence intervals for any terms in the model. There is no mention, however, of functional hypothesis tests or joint bands, which are important inferential quantities in functional regression models, nor whether inference can be obtained for other quantities computed from the modeled parameters, including functions or transformations of model parameters or for aggregated values across prespecified regions of t . When fit with *FDBOost*, inference can only be done by resampling methods. This is straightforward for some models, for other models, especially for multi-level functional data or longitudinally/spatially correlated functional data, it is not immediately obvious how to perform the bootstrap. Also, it is unclear to me under each of these model fitting procedures whether the inference accounts for all sources of variability in the data (see discussion on random effects above) or whether the only effective stochastic variability comes from the residual errors in the model. It is also not clear whether the inference integrates over uncertainty in the choice/estimation of the smoothing and covariance parameters.

Suitability for Complex, High-Dimensional Functional Data—For many people, when they think of functional data, they think of relatively smooth functions sampled on coarse or moderately sampled 1D Euclidean domains. The present framework is ideally suited for these data. However, the scope of functional data is much broader than this. Other functional data involve functions on 2D, 3D or even higher dimensional domains, that are possibly non-Euclidean, and sometimes sampled on extremely high dimensional, fine grids, and sometimes being complex functions with local features like change points, flat regions,

or spikes. One common example of this is fMRI data, for which 3D brain volumes sampled on a grid of ≈ 1 million voxels are sampled repeatedly over time for a number of subjects. The present framework is not as well suited to many of these more complex, high-dimensional functional data because of the computational and modeling limitations mentioned above. This is the setting that has motivated the development of the general functional mixed model (FMM) framework that I will now overview.

3 Summary of Functional Mixed Model Framework

As mentioned above, my collaborators and I have developed our own general modeling framework for functional response regression, conflating linear mixed models, basis functions modeling, and regularization through shrinkage priors to build a general functional mixed model framework and corresponding fully Bayesian modeling approach. This framework has some commonalities with the present modeling framework, and some key differences as well. Since the general framework has been developed over a series of papers, its scope and generality may not be immediately apparent, so I will first overview the framework in its entirety, and then discuss its strengths and weaknesses relative to the FLAM framework laid out by Greven and Scheipl.

3.1 Description of Framework

Core Functional Mixed Model—This work started with a JASA-ACS paper (Morris, et al. 2003) that developed a hierarchical functional model to model colon carcinogenesis data consisting of functions nested within a 3-level hierarchy. The modeling approach involved specifying the hierarchical functional model, which is a special case of a functional mixed model, representing the functions with a wavelet basis expansion, and regularizing the functional coefficients in the model using spike-slab priors and fitting the unified Bayesian model. In Morris and Carroll (2006), we generalized this model to construct a functional mixed model framework, which can contain arbitrary numbers and types of predictors and multiple levels of random effect functions. Suppose we have a sample of functions $Y_i(t)$, $i = 1, \dots, N$ observed on a common fine grid of size T on a domain $\mathcal{T}$, which is potentially multi-dimensional and/or non-Euclidean. The FMM is a functional response regression model given by

$$Y_i(t) = \sum_{a=1}^A X_{ia} B_a(t) + \sum_{h=1}^H \sum_{m=1}^{M_h} Z_{ihm} U_{hm}(t) + E_i(t). \quad (5)$$

As in linear mixed models for scalar data, the fixed and random effect predictors can be discrete or continuous, and can involve individual covariates or interactions of multiple covariates. The corresponding functional regression coefficients are defined on the same domain $\mathcal{T}$, although as described below the framework can also allow the coefficients to be constant in t as well.

Distributional and Covariance Assumptions—For simplicity we first describe the Gaussian FMM with conditionally independent random effect and residual error functions,

for which the random effect functions $U_{hm}(\mathbf{t})$ are iid mean zero Gaussian Processes with covariance $Q_h(\mathbf{t}, \mathbf{t})$ and the residual error functions $E_f(\mathbf{t})$ are iid mean zero Gaussian Processes with covariance $S(\mathbf{t}, \mathbf{t})$. As described below, this framework has been generalized to allow parametrically specified correlation structures across the N functional residuals to accommodate correlated functional data, including AR(p), Matern (Zhu, et al. 2016c), and conditional autoregressive (CAR) (Zhang, et al. 2016a) processes between functions, or to handle multivariate functional data via functional graphical models (Zhang, et al. 2016b). These processes could be assumed to be stationary, with covariance parameters common across $\mathbf{t}$, or nonstationary, allowing the covariance parameters to themselves vary across $\mathbf{t}$. As shown by Zhu, et al. (2011), we are also able to utilize non-Gaussian likelihoods such as Laplace or t distributions on the random effect and residual error functions to obtain robust functional regression that automatically downweights global (entire function) or local (part of a function) outliers.

Note that our model does not separate out independent measurement errors, but absorbs them into the correlated functional residuals $E_f(\mathbf{t})$. This makes no difference for estimation or inference of fixed effect functions $B_d(\mathbf{t})$, the primary goal in the FMM, and significantly speeds calculations by eliminating an additional level containing N random effect functions. If, for some reason, one wished to include iid residual errors, they could define an additional random effect level for correlated curve to curve deviations and constrain $E_f(\mathbf{t})$ to be iid across i and $\mathbf{t}$.

Function-on-Function Regression and Semiparametric Functional Mixed Models—Model (5) from Morris and Carroll (2006) assumes scalar predictors and functional coefficients that, while nonparametric in $\mathbf{t}$, are linear in predictors x_a . Meyer, et al. (2015) showed how this framework could allow functional predictors $X_{ia}(s)$ for $s \in \mathcal{S}$ by introducing function-on-function regression terms

$$\int_{s \in \mathcal{S}} X_{ia}(s) B(s, \mathbf{t}) ds. \quad (6)$$

Lee, et al. (2016) showed how to fit semiparametric FMMs that include smooth functional terms $f(X_{ia}, \mathbf{t})$ that are nonparametric in both x_a and $\mathbf{t}$. These terms can be directly accommodated in model (5) by specifying a fixed effect predictor that is linear in x_a and a single random effect level whose design matrix $\mathbf{Z}_h$ involves Demmler-Reinsch orthonormal spline basis functions (Demmler and Reinsch 1975) evaluated on X_{ia} . In a similar fashion, the framework can accommodate interactions of a linear covariate X_{iaJ} and nonparametric covariate X_{iaN} through the term $X_{iaJ} f_a(X_{iaN}, \mathbf{t})$, which for X_{iaJ} that are discrete effects represented by dummy variables allows separate nonparametric fits of $(X_{iaN}, \mathbf{t})$ for each discrete level, and for X_{iaJ} that are continuous allows the corresponding slope to vary nonparametrically in $(X_{iaN}, \mathbf{t})$. With these capabilities, the FMM has the same broad set of potential predictor structures as the FLAM.

Basis Transform Modeling Approach—A *basis transform modeling approach* is used to fit model (5). This first involves representing the observed functions with a basis expansion with a set of basis functions $\phi_k(t)$, $k = 1, \dots, K$:

$$Y_i(t) = \sum_{k=1}^K Y_{ik}^* \phi_k(t) \quad (7)$$

This framework is intended to be used with *lossless* transforms, such that

$Y_i(t) \equiv \sum_k Y_{ik}^* \phi_k(t)$ for all observed t , so that the basis coefficients $\{Y_{ik}^*; k = 1, \dots, K\}$ contain all information within the observed functional data $\{Y_i(t); t = t_1, \dots, t_T\}$, or at least *near lossless* such that

$$\|Y_i(t) - \sum_{k=1}^K Y_{ik}^* \phi_k(t)\| < \varepsilon \quad \forall i=1, \dots, N \quad (8)$$

for some small value ε and some measure $\|\bullet\|$. This condition assures that the basis space chosen is sufficiently rich such that for all practical purposes it can recapitulate the observed functional data. If this condition is met, one should be able to plot each raw function beside its full basis projection and see no discernible difference. Any basis functions can be used, including commonly used choices splines, wavelets, Fourier bases, PCs as well as any other that one conceives. If the space $\mathcal{T}$ is multi-dimensional or in some non-Euclidean manifold, a set of basis functions appropriate for that space can be chosen, e.g. spherical harmonics or spherical wavelets for functional data on spherical domains.

Rather than putting the basis functions in a design matrix and fitting the model using scalar regression methods, our approach is to project the observed functions into the basis space to obtain the basis coefficients $\mathbf{Y}^*$, a $N \times K$ matrix for which element (i, k) contains the empirical basis coefficient k for function i , fit a basis-space version of the FMM to these coefficients, and then project results back to the data space model (5) for estimation and inference. With basis representation written in matrix form $\mathbf{Y} = \mathbf{Y}^* \Phi$ with Φ a $K \times T$ matrix of basis functions evaluated on the observational grid, the coefficients can be computed by $\mathbf{Y}^* = \mathbf{Y} \Phi^-$ with $\Phi^- = \Phi' (\Phi \Phi')^{-1}$ as long as $\text{rank}(\Phi) = K$, or for certain basis functions special fast algorithms are available for computing these coefficients. For example, Fourier coefficients can be computed via the Fast Fourier Transform (FFT, $O(T \log T)$), wavelet coefficients computed via the Discrete Wavelet Transform (DWT, $O(T)$), and PC scores can be computed using fast methods, as well (e.g. Zipunnikov et al. 2011). While penalized least squares with basis functions contained within a design matrix is the standard computational approach used for penalized spline fitting, this basis transform modeling approach is commonly used for many other bases, including wavelets, Fourier, and fPC.

Basis Space Model—The hallmark of our approach is that rather than fitting model (5), directly, we fit the basis-space version of this model:

$$Y_{ik}^* = \sum_{a=1}^A X_{ia} B_{ak}^* + \sum_{h=1}^H \sum_{m=1}^{M_h} Z_{ihm} U_{hmk}^* + E_{ik}^*, \quad (9)$$

where B_{ak}^* , U_{hmk}^* , and E_{ik}^* are the basis coefficients for the functional fixed effects $B_a(t) = \sum_k B_{ak}^* \phi_k(t)$, functional random effects $U_{hm}(t) = \sum_k U_{hmk}^* \phi_k(t)$, and functional residuals $E_i(t) = \sum_k E_{ik}^* \phi_k(t)$, respectively. For functional predictors (6) (Meyer, et al. 2015), we specify basis expansions $E_{ia}(s) = \sum_l X_{ial}^* \psi_l(s)$, and the corresponding term in the basis-space model is $\sum_l X_{ial}^* B_{alk}^*$.

Under the Gaussian FMM with conditionally independent random effect and residual error functions, we assume $U_{hmk}^* \sim N(0, q_{hk})$ and $E_{ik}^* \sim N(0, s_k)$ with q_{hk} and s_k scalar variance components. For Robust FMMs (Zhu, et al. 2011), the Gaussian assumptions are replaced by Laplace or t distributions, represented using scale mixtures of Gaussians. For correlated FMMs (Zhang, et al. 2016a, Zhu, et al. 2016c), we assume that

$\mathbf{E}_k^* = (E_{1k}^* \cdots E_{Nk}^*)' \sim MVN(0, s_k \mathbf{R}_k)$ for some correlation matrix $\mathbf{R}_k(\boldsymbol{\rho}_k)$ characterizing the interfunctional correlation structure through some parametric form (e.g. AR(p), Matern, CAR) indexed by correlation parameters $\boldsymbol{\rho}_k$. For multivariate functional data, $\mathbf{R}_k = \mathbf{P}_k^{-1}$, where $\mathbf{P}_k$ is a basis coefficient-specific precision matrix that can be fit nonparametrically using principles of Gaussian graphical models (Zhang, et al. 2016b), or alternatively principal components can be defined across the functions for each basis k (Zhu, et al. 2016b). When the data consists of repeated spatially/temporally correlated or multivariate functions for each subject, $\mathbf{R}_k$ is block diagonal. Allowing the correlation parameters $\boldsymbol{\rho}_k$ to vary across basis coefficients k engenders nonstationarity whereby the interfunctional correlation is allowed to vary over t while setting $\boldsymbol{\rho}_k \equiv \boldsymbol{\rho} \forall k$ leads to stationarity.

Shrinkage Priors for Regularization of Fixed Effects—Regularization of the fixed effect functions $B_{ik}(t)$ is accomplished through specification of shrinkage priors for the corresponding basis coefficients:

$$B_{ak}^* \sim g(\boldsymbol{\nu}_{aj}) \quad (10)$$

for some mean zero distribution $g(\bullet)$ with corresponding regularization parameters $\boldsymbol{\nu}_{aj}$ indexed by $j = 1, \dots, J$ that define a partitioning of the basis coefficients $k = 1, \dots, K$ into *regularization sets*, which are subsets of basis coefficients sharing the same regularization parameters. Common choices for $g(\bullet)$ include Gaussians, spike-slabs (George and McCulloch, 1993), or other sparsity priors such as Normal-Gamma (Griffin and Brown, 2010), Laplace (Park and Casella, 2008), Horseshoe (Carvalho, et al. 2010), Generalized Double Pareto (Armagan, Dunson, and Lee, 2013), or Dirichlet-Laplace (Bhattacharya, et al. 2015). The regularization parameters $\boldsymbol{\nu}_{aj}$ can be given hyperpriors or estimated by empirical Bayes. These priors can be adapted to also incorporate other structure, e.g. to smooth in both

spatial and functional domains for a series of fixed effect functions $B_a(\mathbf{t})$ corresponding to a series of spatial locations $a = 1, \dots, A$ (Zhu, et al. 2016c).

These prior distributions induce penalties on the basis coefficients, with Gaussian leading to L2 penalization, Laplace leading to L1 penalization, and other sparsity priors leading to other forms of sparsity-inducing penalization. In terms of shrinkage, Gaussian priors lead to linear shrinkage whereby larger coefficients are shrunk more towards zero, while sparsity priors lead to nonlinear shrinkage for which small coefficients are shrunk more strongly towards zero than larger coefficients. As emphasized in Morris (2015), penalization of basis coefficients is a fundamental tool for inducing regularization for functional regression methods, with L2 penalization of splines leading to smoothing, and L1 or sparsity penalization of wavelet coefficients or PCs leading to a form of adaptive regularization that can adapt locally to features of the functions.

Note that if one wanted to constrain a coefficient $B_a(\mathbf{t})$ to be non-functional, i.e. $B_a(\mathbf{t}) \equiv B_a \forall \mathbf{t}$, they could do so by choosing a basis transform whose first basis function is a constant $\phi_1(\mathbf{t}) = 1 \forall \mathbf{t} \in \mathcal{T}$, and setting $B_{ak} = 0 \forall k = 2, \dots, K$.

Model Fitting Approach—Given that our modeling approach involves fitting a series of penalized mixed models, in principle the FMM could be fit using penalized likelihood or GAM software just as for the FLAM, and inference would have to be done using a bootstrap.

Instead, we fit the model using a fully Bayesian approach, using a Markov Chain Monte Carlo (MCMC) to fit the basis space model (9) for each k , and then transforming back to the data space, e.g. using $B_a(\mathbf{t}) = \sum_k B_{ak}^* \phi_k(\mathbf{t})$ or a fast inverse transform (FFT, IDWT), to yield posterior samples for each parameter in the data-space FMM (5). We work with a marginalized model with all random effects $\mathbf{U}$ integrated out, which speeds convergence of the chain. Vague proper priors (with automatic choices) are used for all variance components and correlation parameters. Each update involves a standard Gibbs step or Metropolis-Hastings with automatically computed proposal variances.

Bayesian Inference—Given the posterior samples, one can construct any desired posterior probabilities and pointwise or joint credible bands, which can be used to construct global or local (in $\mathbf{t}$) inferential summaries. Joint bands are constructed using the approach described in Ruppert, Wand, and Carroll (2003). As introduced in Meyer, et al. (2015), we invert joint bands to construct *Simultaneous Band Scores* (SimBaS) $P_a(\mathbf{t})$, interpreted to be the minimum α such that the $100(1-\alpha)\%$ joint credible band of $B_a(\mathbf{t})$ excludes 0. The measure $GBPV_a = \min_{\mathbf{t}} \{P_a(\mathbf{t})\}$ can be used as a global Bayesian test of whether $B_a(\mathbf{t}) \equiv 0 \forall \mathbf{t}$, and $P_a(\mathbf{t})$ can be considered a pointwise measure that identifies regions of $\mathbf{t}$ for which $B_a(\mathbf{t})$ differs from zero while adjusting for multiple testing based on expected experiment wise error rate (EER). Alternatively, given a minimum practical effect size δ , one could compute $\Pr\{|B_a(\mathbf{t})| \geq \delta | \mathbf{Y}\}$, which can be interpreted as a local false discovery rate (FDR, Morris, et al. 2008) if we consider a discovery to be a location for which the true effect is in fact greater than δ in magnitude.

These inferential summaries can be constructed in the data or basis space for any functional of model parameters, including contrasts (e.g. $B_1(t) - B_2(t)$), nonlinear transformations (e.g. $\exp\{B_d(t)\}$), derivatives (e.g. $f(x_d, t)/x_d$), or integrals (e.g. $\int_{t \in \mathcal{T}_0} B_d(t) dt$ for some $\mathcal{T}_0 \subset \mathcal{T}$).

Functional Discriminant Analysis—Although modeling the functions $Y(t)$ as responses, one could perform classification or prediction of a scalar X with this framework by *functional discriminant analysis* (Zhu, et al. 2012). After a fully Bayesian fitting the FMM to $\{Y(t)|X\}$, it is straightforward to compute predictive probabilities $\Pr\{X=x|Y(t)\}$ through Bayes' Rule, with the option to induce an additional level of regularity using a shrunken nearest centroid approach.

Model Selection—One rigorous but intensive way to perform model selection in this Bayesian framework is through CV-based posterior predictive probabilities for each model. This involves splitting the data into training and validation subsets, fitting an MCMC for each proposed model to the training data, and then calculating the predictive likelihood for the validation data (Zhu, et al. 2016c). To select fixed and random covariates and their functional forms, Lee, et al. (2016) propose a faster alternative that involves a weighted Bayesian Information Criterion (BIC) comparison across models based on preliminary basis-space fits of the proposed models using *lme* in R.

Software—A standalone executable to fit the FMM is available for Windows and Linux (https://biostatistics.mdanderson.org/softwaredownload/SingleSoftware.aspx?Software_Id=70). This version has options for wavelets, PCs, wavelet-regularized PCs, or user-specified basis functions, allows spike-and-slab or Gaussian shrinkage priors, and fits the Gaussian FMM with conditionally independent random effects and predictors. This software can utilize cluster resources to parallelize the calculations, and produces a broad array of specified inferential summaries and writes posterior samples out to compressed binary files. The mixed model engine underlying the software utilizes the same computational strategies to efficiently compute linear mixed models in PROC MIXED in SAS. We have alternative Matlab code for the Gaussian or Robust FMM with either conditionally independent random effects or accommodating interfunctional correlation through functional AR(p), Matern, or CAR processes, Matlab code for performing functional discriminant analysis, and R code to construct the X and Z matrices for fitting semiparametric terms $f(x, t)$ and to perform variable selection for which fixed/random effect terms to contain in the model.

Thus, we have general software for this framework, although not all of it is yet unified into a single software package. We are in the process of developing an R interface *wrapfmm* for this framework (Rausch, et al. 2013) that uses *lmer*-based model statements to specify the model and produces an array of commonly-used summary plots, and are working towards inclusion of the entire general framework within this single package.

3.2 Strengths of FMM

Vast Scope of Modeling Framework—The FMM framework is focused on functional response regression, but is extremely flexible and general. The model incorporates the same broad range of predictors as FLAM, with any number of scalar or functional predictors with scalar or functional coefficients, and these coefficients are nonparametric in t , can be parametric (e.g. linear) or smooth nonparametric in x , and could be modeled as either fixed or random effect functions. Function-on-function terms can be unconstrained or concurrent, and historical function-on-function terms can be accommodated if specially constructed wavelet packet bases are used (work in preparation). Gaussian likelihoods can be used to perform mean regression, Laplace likelihoods for median regression, or other scale mixtures (e.g. t) used for other types of robust regression.

General Basis Functions and Regularization Approaches—While initially developed using wavelets (Morris and Carroll 2006), subsequent papers (starting with Morris, et al. 2011) describe how the same basis transform modeling approach can be used with any chosen basis functions. The flexible set of available shrinkage priors can induce either sparsity (e.g. L1 penalization) or linear shrinkage (L2 penalization) so can be used with many different basis functions.

The use of a general basis transform approach enables this method to be applied to many types of functional data objects, not just smooth functions on 1D Euclidean domains. Higher order basis functions can be used for 2D or 3D data, basis functions defined on various other non-Euclidean domains can be used to represent functional objects on other domains, e.g. for functional data on a sphere or shape spaces.

Scalability to Large Functional Data Sets—The computational strategy enables scalability to extremely large functional data sets in terms of the number of observations per function T and the number of functions N . Once the basis coefficients $\mathbf{Y}^*$ are calculated, the MCMC calculations are linear in K , which is at most T and sometimes much smaller whenever a sparse but near-lossless basis can be found. When utilizing a Gaussian model with conditionally independent random effect functions and residual error functions, the calculations done at each MCMC iteration are free of N since the MCMC is done on summaries that have already collapsed over N (Herrick and Morris 2006) and random effects are not needed to capture curve-to-curve deviations with intrafunctional correlation. In many cases the basis coefficients can be calculated rather quickly (wavelets $O(NT)$, Fourier coefficients $O(NT \log T)$, PC scores $O(\min(NT^2, N^2 T))$), and even when not, the basis transform only needs to be computed once up front for each of N observed functions and the inverse transform only needs to be computed for A fixed effect functions at each MCMC iteration. Cluster and GPU resources can be used since the method is parallelizable across chains of MCMC samples, and frequently also across basis coefficients k or at least regularization groups j . For memory management, online calculation of posterior summaries can be done so not all posterior samples need be loaded into memory at one time. This has allowed us to apply this fully Bayesian method to very large data sets, e.g. T on the order of 1 million (Morris, et al. 2011), and N on the order of 100,000 (Zhu, et al. 2016c). By not splitting out independent errors, this approach is able to capture intrafunctional correlations

in the curve-to-curve deviations without specifying an extra level with N random effect functions, which significantly speeds calculations since mixed models (with “true” random effects) are cubic in the number of random effects.

Nonstationarities Provide Flexibility to Model Complex Functional Data—The basis transform modeling approach underlying the FMM accommodates incorporation of many useful types of flexible nonstationary structures into the model without much computational difficulty. By indexing covariance parameters by basis function k , it allows the variance of the curve-to-curve deviations, the degree of intrafunctional correlation, the strength of interfunctional correlation from random effect functions or parametric structures (e.g. AR(p), Matern, CAR), the edge strength in graphical models, and the scale and/or shape parameters characterizing the heaviness of tails in Robust FMM to all vary across functional locations t . These types of nonstationarities are crucial for many modern functional data, especially complex high-dimensional functional data, since the assumption of these quantities being constant over t would be naive and unrealistic in many of these settings.

Handling of Within-Function Correlations—The basis transform modeling approach enables the FMM to automatically account for interfunctional correlation even though the model for each basis k is fit separately. In the Gaussian model with conditionally independent random effect functions, the induced intrafunctional covariance of the residual error functions is determined by

$$\text{cov}\{E_i(t_1), E_i(t_2)\} = \sum_k s_k \phi_k(t_1) \phi_k(t_2), \quad (11)$$

and across random effect functions at level h is $\text{cov}\{U_{hm}(t_1), U_{hm}(t_2)\} = \sum_k q_{hk} \phi_k(t_1) \phi_k(t_2)$. Note that this induces within-function covariance according to the chosen basis functions even when using orthogonal basis functions, as a result of the heteroscedasticity of the variance components across basis functions k .

Handling of Between-Function Correlations—This framework can accommodate between-function correlation induced by the experimental design either through random effect functions or parametric structures. Integrating out the random effect functions, we have $\text{cov}(Y_{ik}^*, Y_{i'k}^*) = \sum_h q_{hk} \sum_m Z_{ihm} Z_{i'hm} + s_k R_k(i, i') = \Omega_k(i, i')$ in the basis space, and $\text{cov}\{Y_i(t_1), Y_{i'}(t_2)\} = \sum_k \Omega_k(i, i') \phi_k(t_1) \phi_k(t_2)$ back in the data space. Our LMM-based model fitting procedure effectively integrates over the random effects, so inference is performed in a way that applies to the population from which the random effect units were sampled. Since the fixed effect functions are updated based on this marginal model, the estimation and inference of the fixed effect functions automatically accounts for these correlation structures.

Extensive Inferential Outputs Provided by Unified Bayesian Model—As described above, this fully Bayesian unified modeling approach yields a vast array of Bayesian inference, including posterior probabilities, pointwise and joint bands, global tests,

and local tests (over t) that can flag regions of t for which $B_d(t)$ is significant while accounting for multiple testing by EER or FDR criteria. This inference integrates over all sources of variability in the data, including estimation of all nuisance parameters, and can be performed on any functional of the parameters in the model.

Modularity of Unified Bayesian Framework—The modularity of the Bayesian framework allows other model components to be easily integrated into this modeling framework. For example, if there is missingness in the observed functions $Y_i(t)$ or measurement error in predictors X_{ia} , then the observed data (projected in the basis space) or predictors can be modeled as parameters and updated from their complete conditional distribution during the MCMC (see Morris, et al. 2006).

Modularity of Basis Transform Modeling Approach—The basis space model (9) is essentially a scalar mixed model that is fit to each basis coefficient, so in principle any scalar regression software could be incorporated within this framework by applying it to each basis coefficient.

3.3 Potential Limitations of FMM

Some of the features of this framework that allow for scalable computing, flexible modeling of nonstationary features, and vast Bayesian inferential summaries, including the transformation into a single basis space and independent modeling across basis coefficients, also lead to some drawbacks and limitations. In this section, I describe some of these drawbacks.

Restriction to Lossless or Near-Lossless Basis Representations—As stated above, this modeling framework is intended for lossless and near-lossless basis representations. If one applies this approach using a “lossy” basis transform for which (8) does not apply, the results would be potentially biased and will fail to account for this lack-of-fit as a source of variability.

Restrictions to Functions Observed on a Common, Fine Grid—The FMM presented above presumes all functions are observed on the same grid, while for some functional data sets, they are not. This may appear to be a severe restriction for a functional regression method. It is for sparsely sampled functional data, but not so much for the complex, high-dimensional functional data setting that motivated this framework, for which the functions are typically sampled on a very fine grid. In that case, the functions are often sampled on a common grid and, even if not, if the grid is sufficiently dense, interpolation could be used to map onto a common grid without substantively changing the data. However, this cannot be done for functions sampled on a sparse grid that varies across subjects, which tends to be the case for simple, smooth functions, the setting that motivated the genesis of FDA.

Focus on Functional Response Regression—The FMM framework we presented is a functional response regression framework that regresses a functional response on scalar or functional predictors, and we did not include functional predictor regression (i.e. scalar

regressed on function) as a part of this framework. We have previously built a fully Bayesian functional predictor regression framework (Malloy, et al. 2010) utilizing wavelets and spike-slab priors that can also be used with other basis functions and regularization priors to build a similarly general framework, but we have not presented this here. We find that functional predictor regression methods can be problematic at times and difficult to interpret, especially for complex, high-dimensional functions, and if we are interested in modeling $(X|Y(t))$, we can always fit $(Y(t)|X)$ using an FMM and then use functional discriminant analysis to estimate $f(X|Y(t))$ as described above.

Gaussian Functional Responses—The framework presented assumes Gaussian (or at least absolutely continuous) responses, and unlike the FLAM does not utilize link functions to generalize to other exponential family responses. We have actually extended the framework in this fashion and used a probit link function to model genomic SNP data as discrete functional data (Meyer, et al. 2016), but this method requires application of the basis transform and inverse transform for each observed function at each MCMC iteration, so sacrifices many of the computational advantages of the FMM we have presented here. Thus, we don't strictly consider this to be part of the same framework. One strategy that can be used within the FMM framework that expands the class of potential functional responses that can be modeled is classical regression transformation approaches utilizing variance stabilizing transformations (e.g. log, square root) or Box-Cox transformations to transform $Y_X(t)$ before modeling (Carroll and Ruppert 1988), with the understanding that our approach can yield inference for any transformations of the model parameters.

Limitations on the Structure of Within Function Correlation Structures—Although the FMM accommodates intrafunctional correlation, its precise form is not unrestricted because of the basis transform modeling approach and corresponding independence assumption across basis coefficients. For example, the residual error functions, as shown above in (11), the form of $S(t, t)$ is determined by the K variance components s_k and basis functions $\phi_k(t)$, so care must be taken to choose basis functions that capture the features of the functional data being modeled. Also, the basis transform modeling approach requires a **single basis representation** for all levels of fixed effects, random effects, and residual error functions, unlike FLAM for which separate basis representations can be used for each additive term if desired.

General Software Package Not Complete—Probably the biggest practical limitation to the impact of this FMM is the lack of availability of a general R package to fit any variants of the general modeling framework. As described above, we have general software for fitting these models that we have written from scratch, but currently there are two different versions of the code: (1) a C++ executable implementing the Gaussian FMM with conditionally independent random effect and residual error functions, and (2) Matlab code that includes Gaussian or Robust models for conditionally independent or correlated functional data. We need to get these integrated into a common package, and develop an accompanying R package that contains functions for model selection, specification of the model through an *lmer*-based model statement (the current software requires specification of X and Z matrices that can be challenging for inexperienced users), and post-hoc inference,

diagnostics, and plotting. We are in the process of developing this package, and hope to have it available in the near future.

4 Comparison and Contrast of FLAM and FMM

Scope of Modeling Framework—For functional response regression, both methods have the same broad array of potential predictor terms, including functional or scalar covariates with coefficients that are constant or functional, and either parametric (linear) or smooth in form, fixed or random. FLAM can handle general exponential link functions, while FMM is designed for continuous functional responses. Both FLAM and FMM can perform median (robust) regression, but FLAM can be used for generalized quantile regression, while the current version of the FMM cannot. FMM can obtain inference while modeling any transformation of $Y(t)$, such as log, square root, or Box-Cox transformations, since inferential summaries can be computed for any transformation of model parameters, and other likelihoods can be used with FMM, including any scale mixtures of Gaussians (e.g. t), but not with FLAM. Only FLAM includes scalar-on-function regression as part of its package.

Basis Function Modeling and Regularization—Both methods represent the functional data through basis functions and regularize via some type of penalization. FLAM utilizes L2 penalization, which may limit the types of basis functions that can be effectively used, e.g. splines and fPCs, while the general shrinkage priors available for FMM yield much richer class of regularization properties (including sparsity) that make it broadly applicable across a wide range of potential basis functions. Its accommodation of a broader class of basis functions enables the FMM to immediately be applied to functional data defined on higher dimensional (e.g. 2D or 3D) and potentially non-Euclidean domains, making it well equipped to model more complex functional objects. The form of the penalty for the FLAM involves just a scalar smoothing parameter and fixed penalty matrix, while the parameterization of the FMM can allow a broader class of regularization parameters, which can potentially yield more flexibility in modeling within-function correlations and also enables nonstationary structures, e.g. allowing the smoothing parameter of nonparametric term $f(x, t)$ regulating smoothness of x to vary over k , which subsequently allows it to vary over t (Lee, et al. 2016).

Modeling Approach—Both frameworks are based on linear mixed models and include the basis functions for predictors x in the design matrix, but differ in how they handle the basis functions for functional index t . FLAM also includes these basis functions in the design matrix using a tensor product structure, while the FMM uses a basis transform modeling approach, projecting the data into the t basis space and modeling columns separately. This approach has computational advantages, with calculations $< O(T)$ and parallelizable, and not requiring all data to be loaded into memory at once, but has the disadvantage of requiring the same basis functions in t for all additive terms. For FLAM, different basis functions can be used for each additive term, e.g. using multi-level PC decompositions (Di, et al. 2009), but calculations do not scale as well with T or N , so can experience challenges for enormous functional data sets. FLAM's design matrix centric

approach also enables it to easily handle sparsely sampled functional data for which the observational grid can be different across subjects, while the FMM's basis transform modeling approach requires common grids and for practical purposes, reasonably dense grids, for its use.

Handling of Between-Function Correlation—Both frameworks allow random effect functional predictors, but it is not clear whether the random effect functions are treated the same way by FMM and FLAM. In the FMM, the random effect functions are effectively integrated out of the model, meaning that they induce between-function correlation that is taken into account in estimation and inference for the fixed effect functions, and inference can be made on the population from which the random effect units were sampled. It is not clear whether the various computational approaches available for fitting the FLAM (*gam* and *gamm* functions, and the component wise boosting procedure) have these properties or whether they instead treat the random effect functions as simply shrunk additive terms that are conditioned upon. Also, the FMM allows a broad class of between-function correlation structures including various spatial and temporal processes as well as graphical models, and these correlation structures are allowed to be nonstationary, with correlation strength varying over t . The current version of the FLAM appears to be much more limited in its ability to account for this types of between-function structure.

Model Fitting and Inference—The FLAM is fit either using existing GAM software or component wise boosting procedures, both of which are penalized-likelihood frequentist approaches. The GAM software yields asymptotic standard errors and confidence intervals, and bootstrap procedures must be conducted to get inferential summaries for the boosting approach. The FMM is fit using a unified Bayesian modeling approach, which yields posterior samples that can be used to perform various global and local Bayesian inferences on any transformation or function of model parameters. Pointwise summaries can be computed that can identify regions of t with significant differences while adjusting for multiple testing based on EER/FDR criteria. Furthermore, this inference integrates over the uncertainty of any parameters in the model, including covariance parameters and regularization parameters, and the unified Bayesian approach can easily accommodate other issues such as missing data or measurement error when they arise.

Software—Both frameworks have general software, but currently the FLAM software is much further along in terms of being a usable, general package. There are R functions implementing both the GAM-based and boosting modeling approaches with intuitive model statements and functions for diagnostics and presentation of results. The Gaussian version of the FMM with conditionally independent random effect functions and residual error functions has general standalone software that incorporates several built-in basis functions and the option for user-defined basis functions, but the design matrices X and Z_h must be explicitly specified, and input files are in Matlab, which is not accessible to everyone. There is general Matlab code for fitting the robust and correlated FMMs, but this is not integrated into the standalone executable at this point. We have begun the process of developing a single R package for the FMM, including all of these options, plus the ability to specify the model using *lmer*-based model statements and producing various plots, summaries, and

diagnostics, but it is not complete yet. When complete, we hope this will enhance the ability of other researchers to use the FMM to analyze their data.

5 Conclusions

Both frameworks are very general and promise to make a strong impact in many areas of science. Both have similar generality in terms of functional response regression, but their differences make each suited to a different type of functional data.

The design matrix-centric, L2 penalized-based modeling approach used by FLAM makes it especially well-suited to simple, smooth functional data, especially when observed on a sparse sampling grid that may vary across subjects, which are data for which splines and fPCs are especially effective. The FMM is not designed for use with such sparsely sampled functional data, and we recommend the use of the FLAM for such data, and also in cases where functional quantile regression or scalar on functional regression is desired.

The FMM was motivated by and designed for use with complex functional data observed on fine and potentially very high dimensional grids. The basis transform modeling approach underlying the FMM makes it very well suited to this context, yielding computational algorithms that yield rigorous inference yet scale up to these enormous data sizes. The framework allows flexibility in allowing the types of modeling structures useful for these complex and sometimes irregular data, including more generality in terms of basis functions and regularization strategies, and various types of nonstationarities and between-function correlation structures that are commonly encountered in these data types.

I am sure that both frameworks will continue to be developed, and provide researchers with a useful set of tools for extracting the rich information contained in their functional data.

Acknowledgments

I wish to thank my colleagues and trainees who have collaborated with me in developing the FMM framework, including Hongxiao Zhu, Wonyul Lee, Lin Zhang, Mark Meyer, Betty Malloy, Hoin Yang, Philip Brown, Veera Baladandayuthapani, Raymond Carroll, Brent Coull, Marina Vannucci, Keith Baggerly, and Kevin Coombes. Also, I thank Richard Herrick for developing the C++ interface, and Philip Rausch for the R package being developed, and I thank my many biomedical collaborators whose data has motivated the development of the FMM. This work was supported by NIH grants CA-107304 and CA-016672, and NSF grant 1550088.

References

- Armagan A, Dunson DB, Lee J. Generalized double pareto shrinkage. *Statistica Sinica*. 2013; 23(1): 119–143. [PubMed: 24478567]
- Bhattacharya A, Pati D, Pillai NS, Dunson DB. Dirichlet-Laplace priors for optimal shrinkage. *J Am Statist Assoc*. 2015; 110(512):1479–1490.
- Brockhaus, S. FDboost: Boosting Functional Regression Models. 2016. URL <http://cran.r-project.org/web/packages/FDboost>. R package version 0.1–1
- Brockhaus S, Melcher M, Seisch, Greven S. Boosting flexible functional regression models with a high number of functional historical effects. *Statistics and Computing*. 2016; doi: 10.1007/s11222-016-9662-1
- Brockhaus S, Scheipl F, Hothorn T, Greven S. The functional linear array model. *Statistical Modeling*. 2015; 15(3):279–300.

- Buhlmann P, Hothorn T. Boosting algorithms: regularization, prediction, and model fitting. *Statistical Science*. 2007; 22(4):477–505.
- Carroll, RJ., Ruppert, D. Transformation and Weighting in Regression. New York: Chapman and Hall; 1988.
- Carvahlo CM, Polson NG, Scott JG. The horseshoe estimator for sparse signals. *Biometrika*. 2010; 97(2):465–480.
- Cederbaum J, Pouplier M, Hoole P, Greven S. Functional linear mixed models for irregularly or sparsely sampled data. *Statistical Modeling*. 2015; 16(1):67–88.
- Demmler A, Reinsch C. Oscillation matrices with spline smoothing. *Numerische Mathematik*. 1975; 24:375–382.
- Di CZ, Crainiceanu CM, Caffo BM, Punjabi NM. Multilevel functional principal component analysis. *Annals of Applied Statistics*. 2009; 3:458–488. [PubMed: 20221415]
- George EI, McCulloch RE. Variable selection via Gibbs sampling. *J Am Statist Ass*. 1993; 88(423): 881–889.
- Griffin JE, Brown PJ. Inference with normal-gamma prior distributions in regression problems. *Bayesian Analysis*. 2010; 5:171–188.
- Herrick, RC., Morris, JS. Wavelet-based functional mixed model analysis: Computational considerations. Joint Statistical Meetings 2006 Proceedings, ASA Section on Statistical Computing; 2006.
- Huang, L., Scheipl, F., Goldsmith, J., Gellar, J., Harezlak, J., McLean, MW., Swihart, B., Xiao, L., Crainiceanu, C., Reiss, P. refund: Regression with Functional Data. 2016. URL <https://CRAN.R-project.org/package=refund>
- Ivanescu AE, Staicu AM, Scheipl F, Greven S. Penalized function-on-function regression. *Computational Statistics*. 2015; 30(2):539–568.
- Lee W, Baladandayuthapani V, Fazio M, Downs C, Morris JS. Semiparametric functional mixed models for longitudinally correlated functional data with application to glaucoma data. 2016 Under review.
- Lee W, Morris JS. Identification of differentially expressed methylated loci using wavelet-based functional mixed models. *Bioinformatics*. 2015 Nov 11; 32(5):664–672. [PubMed: 26559505]
- Malloy EJ, Morris JS, Adar SD, Suh HH, Gold DR, Coull BA. Wavelet-based functional linear mixed models: an application to measurement error connected distributed lag models. *Biostatistics*. 2010; 11(3):432–452. [PubMed: 20156988]
- Martinez JG, Bohn KM, Carroll RJ, Morris JS. A study of Mexican free-tailed bat syllables: Bayesian functional mixed modeling of nonstationary acoustic time series. *J Am Statist Ass*. 2013; 108(502):514–526.
- McLean MW, Hooker G, Staicu AM, Scheipl F, Ruppert D. Functional generalized additive models. *Journal of Computational and Graphical Statistics*. 2014; 23(1):249–269. [PubMed: 24729671]
- Meyer M, Coull BA, Versace F, Cinciripini P, Morris JS. Bayesian function-on-function regression for multi-level functional data. *Biom*. 2015; 71(3):563–574.
- Meyer M, Morris JS, Morrow JD, Hersh CP, Lange C, Coull BA. Ordinal probit wavelet-based functional models for eQTL analysis. 2016 Under review.
- Morris JS. Statistical methods for proteomic biomarker discovery using feature extraction or functional data analysis approaches. *Statistics and its Interface*. 2013; 5(3):117–136.
- Morris JS. Functional regression. *Annual Review of Statistics and Its Application*. 2015; 2:321–359.
- Morris JS, Arroyo C, Coull BA, Ryan LM, Herrick RC, Gortmaker SL. Using wavelet-based functional mixed models to characterize population heterogeneity in accelerometer profiles: A case study. *J Am Statist Ass*. 2006; 101(476):1352–1364.
- Morris JS, Baladandayuthapani V, Herrick RC, Sanna PP, Gutstein H. Automated analysis of quantitative image data using isomorphic functional mixed models, with application to proteomics data. *Ann Appl Statist*. 2011; 5:894–923.
- Morris JS, Brown PJ, Herrick RC, Baggerly KA, Coombes KR. Bayesian analysis of mass spectrometry data using wavelet-based functional mixed models. *Biom*. 2008; 12:479–489.

- Morris JS, Carroll RJ. Wavelet-based functional mixed models. *J R Statist Soc B*. 2006; 68(2):179–199.
- Morris JS, Vannucci M, Brown PJ, Carroll RJ. Wavelet-based nonparametric modeling of hierarchical functions in colon carcinogenesis. *J Am Statist Ass*. 2003; 98:573–583.
- Park T, Casella G. The Bayesian Lasso. *J Am Statist Ass*. 2008; 103(482):681–686.
- Ramsay, JO., Silverman, BW. *Functional Data Analysis*. 2. New York: Springer-Verlag; 2005.
- Rausch P, Morris JS, Sommer W, Krifka M. When you are thrown a curve: Two R packages for swerving with wavelet-based functional mixed models. *Linguistic Evidence Conference*. 2013
- Ruppert, D., Wand, MP., Carroll, RJ. *Semiparametric Regression*. New York: Cambridge University Press; 2003.
- Scheipl F, Gertheiss J, Greven S. Generalized functional additive mixed models. *Electronic Journal of Statistics*. 2016; 10:1455–1492.
- Scheipl F, Greven S. Identifiability in penalized function-on-function regression models. *Electronic Journal of Statistics*. 2016; 10:495–526.
- Scheipl F, Staicu AM, Greven S. Functional additive mixed models. *JCGS*. 2015; 24:477–501.
- Wood, SN. mgcv: Mixed GAM computational vehicle with GCV/AIC/REML smoothness estimation. 2016. URL <http://CRAN.R-project.org/package=mgcv>. R package version 1.8–12
- Zhang L, Baladandayuthapani V, Zhu H, Baggerly KA, Majewski TA, Czerniak BA, Morris JS. Functional CAR models for large spatially correlated functional datasets. *J Am Statist Ass*. 2014; 114(514):772–786.
- Zhang L, Baladandayuthapani V, Versace F, Cinciripini P, Morris JS. Bayesian functional graphical modeling. 2016 Under review.
- Zhu H, Brown PJ, Morris JS. Robust, adaptive functional regression in functional mixed model framework. *J Am Statist Ass*. 2011; 106(495):1167–1179.
- Zhu H, Brown PJ, Morris JS. Robust classification of functional and quantitative image data using functional mixed models. *Biometrics*. 2012; 68:1260–1268. [PubMed: 22670567]
- Zhu H, Caspers P, Morris JS, Wu X, Muller R. A unified analysis of sonar terrain data using Bayesian functional mixed models. *Technometrics*. 2016 to appear.
- Zhu H, Morris JS, Wei F, Cox DD. Multivariate functional response regression, with application to fluorescence spectroscopy in a cervical pre-cancer study. *Computational Statistics and Data Analysis*. 2016 to appear.
- Zhu H, Versace F, Cinciripini P, Morris JS. Robust functional mixed models for spatially correlated functional regression, with application to event-related potentials for nicotine-addicted individuals. 2016 Under review.
- Zipunnikinov V, Caffo B, Yousem D, Davitzikos C, Schwartz BS, Crainiceanu C. Multilevel functional principal component analysis for high-dimensional data. *Journal of Computational and Graphical Statistics*. 2011; 20:852–873. [PubMed: 25960627]