Penalized Function-on-Function Regression

Andrada E. Ivanescu*, Ana-Maria Staicu†, Fabian Scheipl‡, Sonja Greven‡

April 8, 2013

Abstract

We propose a general framework for smooth regression of a functional response on one or multiple functional predictors. Using the mixed model representation of penalized regression expands the scope of function on function regression to many realistic scenarios. In particular, the approach can accommodate a densely or sparsely sampled functional response as well as multiple functional predictors that are observed: 1) on the same or different domains than the functional response; 2) on a dense or sparse grid; and 3) with or without noise. It also allows for seamless integration of continuous or categorical covariates and provides approximate confidence intervals as a by-product of the mixed model inference. The proposed methods are accompanied by easy to use and robust software implemented in the `pffr` function of the `R` package `refund`. Methodological developments are general, but were inspired by and applied to a Diffusion Tensor Imaging (DTI) brain tractography dataset.

Keywords: functional data analysis; functional regression model; mixed model; multiple functional predictors; penalized splines; tractography data.

1 Introduction

We study the relationship between a functional response and one or multiple functional predictors. While the approach is general, the presentation is focused on the case of two functional predictors. Let $Y_i(t_{ij})$ be the functional outcome for subject i measured at $t_{ij} \in \mathcal{T}$, an interval on the real line, $1 \leq i \leq n$, $1 \leq j \leq m_i$, where n is the number of curves or subjects

*Department of Biostatistics, East Carolina University, Greenville, NC 27834, USA.
†Department of Statistics, North Carolina State University, Raleigh, NC 27695, USA.
‡Department of Statistics, Ludwig-Maximilians-University Munich, Germany.

and m_i is the number of observations for curve i. We assume that the observed data for the i^{th} subject are $[\{Y_i(t_{ij})\}_j, \{X_{i1}(s_{ik})\}_k, \{X_{i2}(r_{iq})\}_q, \boldsymbol{W}_i]$, where $\{X_{i1}(s_{ik}) : 1 \leq k \leq K_i\}$ and $\{X_{i2}(r_{iq}) : 1 \leq q \leq Q_i\}$ are functional predictors and $\boldsymbol{W}_i$ is a $p \times 1$ vector of scalar covariates. Furthermore, it is assumed that $\{X_{i1}(s_{ik})\}$ and $\{X_{i2}(r_{iq})\}$ are square-integrable, finite sample realizations of some underlying stochastic processes $\{X_1(s) : s \in \mathcal{S}\}$ and $\{X_2(r) : r \in \mathcal{R}\}$ respectively, where $\mathcal{S}$ and $\mathcal{R}$ are intervals on the real line. We start with the following model for $Y_i(t_{ij})$:

$$Y_i(t_{ij}) = \boldsymbol{W}_i\boldsymbol{\gamma} + \beta_0(t_{ij}) + \int_{\mathcal{S}} \beta_1(t_{ij}, s)X_{i1}(s)\, ds + \int_{\mathcal{R}} \beta_2(t_{ij}, r)X_{i2}(r)\, dr + \epsilon_i(t_{ij}), \qquad (1)$$

where the mean function is modeled semiparametrically (Ruppert, Wand and Carroll 2003; Wood 2006) and consists of two components: a linear parametric function $\boldsymbol{W}_i\boldsymbol{\gamma}$ to account for the scalar covariates $\boldsymbol{W}_i$ and an overall nonparametric function $\beta_0(t)$. The effect of the functional predictors is captured by the component $\int_{\mathcal{S}} \beta_1(t_{ij}, s)X_{i1}(s)\, ds + \int_{\mathcal{R}} \beta_2(t_{ij}, r)X_{i2}(r)\, dr$. The regression parameter functions, $\beta_1(\cdot, \cdot)$ and $\beta_2(\cdot, \cdot)$, are assumed to be smooth and square-integrable over their domains. The errors ϵ_i are mean zero random processes and are uncorrelated with the functional predictors and scalar covariates. The model will be extended in several different ways in Section 3, but for now we keep the presentation simple.

Function on function regression models are well known; see, for example, Ramsay and Silverman 2005, ch 12. Most of the work in this area (e.g. Yao, Müller and Wang 2005b; He, Müller, Wang and Wang 2010; Wu, Fan and Müller 2010) has concentrated on functional regression with one functional predictor and principal component expansions for both the functional predictor and the corresponding functional coefficient. While these approaches are intuitive, they are also subject to two subtly dangerous problems. First, estimating the number of components used for the expansion of the functional predictors is known to be difficult but relatively unimportant if the focus is on predicting the outcome, see Goldsmith, Bobb, Crainiceanu, Caffo and Reich (2011). However, as noticed in Crainiceanu, Staicu and Di (2009), the shape and interpretation of the functional parameter can change

dramatically when one includes one or two additional principal components. This problem is exacerbated by the fact that eigenfunctions corresponding to smaller eigenvalues tend to be much wigglier in applications. Second, the smoothness of the functional parameter is induced by the choice of basis dimension of the functional predictor. This may lead to strong under-smoothing when the functional parameter is much smoother than the higher order principal components. We found this to be the norm rather than the exception in applications. Furthermore, principal component-based function-on-function regression methods are not currently extended to incorporate several functional predictors and multiple scalar covariates.

A second common approach assumes a concurrent relationship between predictor and response. This is a simpler model that is additionally restricted to the case of both variables being observed over the same domain (e.g. Ramsay and Silverman, 2005, ch 14; Jiang and Wang, 2011, consider a single index model for the concurrent model).

In this paper we provide a novel solution that addresses these problems using a philosophically and practically different approach to smoothing. Our approach is inspired by the penalized functional regression (PFR) in Goldsmith et al. (2011), developed for the simpler case of scalar on function regression. Smoothing of the functional coefficient is controlled by only one parameter, which is estimated using restricted maximum likelihood (REML) in an associated mixed model. We choose a similar approach for the more complex case of function on function regression involving multiple functional predictors.

The lack of methods for regression of functional outcomes on functional predictors in wider generality, such as functional predictors observed on the same or different domains than the functional response, and in various realistic scenarios, such as multiple functional predictors or functional responses that are observed on a dense or sparse grid, is a serious methodological and computational gap in the literature. We propose a functional linear regression framework able to accommodate these complications, by introducing penalized function-on-function regression (PFFR) and the associated software: the `pffr` function of the `R` package `refund` (Crainiceanu, Reiss, Goldsmith, Greven, Huang and Scheipl 2012).

The paper is organized as follows. Our PFFR method is introduced in Section 2 and then

extended to accommodate various realistic scenarios in Section 3. The performance of PFFR in a simulation experiment is presented in Section 4. Section 5 presents an application to a Diffusion Tensor Imaging (DTI) tractography dataset. Section 6 provides our discussion.

2 Mixed model representation of function on function regression

In this section we introduce a representation of model (1) that allows us to estimate the model as a mixed model using available software. To begin, we discuss first the case when the predictor curves, $X_{i1}(\cdot)$ and $X_{i2}(\cdot)$, are measured densely and without noise. For simplicity of presentation, we consider the case when $K_i = K$, $s_{ik} = s_k$, $Q_i = Q$ and $r_{iq} = r_q$ for every i, k and q. Also, we assume that the outcome curves are measured on the same common grid, that is $m_i = m$ and $t_{ij} = t_j$ for every i and j. Section 3 presents extensions to other realistic designs.

2.1 A penalized criterion for function on function regression

The functional intercept we expand as $\beta_0(t) \approx \sum_{l=1}^{\kappa_0} A_{0,l}(t)\beta_{0,l}$, where $A_{0,l}(\cdot)$ is a known univariate basis and $\beta_{0,l}$ are the corresponding coefficients. For the linear function-on-function term, we approximate $\int_{\mathcal{S}} \beta_1(t,s)X_{i1}(s)ds$ using Riemann sums on a fine grid. More precisely, $\int_{\mathcal{S}} \beta_1(t,s)X_{i1}(s)ds \approx \sum_{k=1}^{K} \Delta_k \beta_1(t,s_k)X_{i1}(s_k)$, where s_k, $k = 1,\ldots,K$, forms a grid of points in $\mathcal{S}$ and Δ_k are the lengths of the corresponding intervals. The next step is to expand $\beta_1(t,s) \approx \sum_{l=1}^{\kappa_1} a_{1,l}(t,s)\beta_{1,l}$, where $a_{1,l}(\cdot,\cdot)$ is a bivariate basis and $\beta_{1,l}$ are the corresponding coefficients. Thus, $\int_{\mathcal{S}} \beta_1(t,s)X_{i1}(s)ds$ can be approximated arbitrarily closely by

$$\sum_{l=1}^{\kappa_1} \left\{ \sum_{k=1}^{K} a_{1,l}(t,s_k)\{\Delta_k X_{i1}(s_k)\} \right\} \beta_{1,l} = \sum_{l=1}^{\kappa_1} \left\{ \sum_{k=1}^{K} a_{1,l}(t,s_k)\widetilde{X}_{i1}(s_k) \right\} \beta_{1,l},$$

where $\widetilde{X}_{i1}(s_k) = \Delta_k X_{i1}(s_k)$, by increasing the density of the grid $\{s_k : k = 1,\ldots,K\}$. Using

similar notation, $\int_{\mathcal{R}} \beta_2(t,r)X_{i2}(r)dr$ can be approximated arbitrarily closely by

$$\sum_{l=1}^{\kappa_2} \left\{ \sum_{q=1}^{Q} a_{2,l}(t,r_q)\widetilde{X}_{i2}(r_q) \right\} \beta_{2,l}.$$

Thus, we approximate model (1) by the additive model:

$$Y_i(t) = \boldsymbol{W}_i\boldsymbol{\gamma} + \sum_{l=1}^{\kappa_0} A_{0,l}(t)\beta_{0,l} + \sum_{l=1}^{\kappa_1} A_{1,l,i}(t)\beta_{1,l} + \sum_{l=1}^{\kappa_2} A_{2,l,i}(t)\beta_{2,l} + \varepsilon_i(t), \tag{2}$$

where $A_{1,l,i}(t) = \sum_{k=1}^{K} a_{1,l}(t,s_k)\widetilde{X}_{i1}(s_k)$ and $A_{2,l,i}(t) = \sum_{q=1}^{Q} a_{2,l}(t,r_q)\widetilde{X}_{i2}(r_q)$ are known because the predictor functions are observed without noise.

While the presentation was provided in full generality, the various choices involved are crucial when one develops practical software. Our philosophical approach to smoothing is to use rich bases that reasonably exceed the maximum complexity of the parameter functions to be estimated and then penalize the roughness of these functions. This translates into choosing a large number of basis functions, κ_0, κ_1, κ_2, and introducing the penalties $\lambda_0 P_0(\boldsymbol{\beta}_0)$, $\lambda_1 P_1(\boldsymbol{\beta}_1)$, and $\lambda_2 P_2(\boldsymbol{\beta}_2)$, where $\boldsymbol{\beta}_d$ is the vector of all parameters $\beta_{d,l}$ for $d = 0, 1, 2$. Thus, if we denote by $\mu_i(t;\boldsymbol{\gamma},\boldsymbol{\beta}_0,\boldsymbol{\beta}_1,\boldsymbol{\beta}_2)$ the mean of $Y_i(t)$, our penalized criterion to be minimized is

$$\sum_{i,j} ||Y_i(t_j) - \mu_i(t_j;\boldsymbol{\gamma},\boldsymbol{\beta}_0,\boldsymbol{\beta}_1,\boldsymbol{\beta}_2)||^2 + \lambda_0 P_0(\boldsymbol{\beta}_0) + \lambda_1 P_1(\boldsymbol{\beta}_1) + \lambda_2 P_2(\boldsymbol{\beta}_2). \tag{3}$$

A least squares criterion as in the first summand of (3) seems like a natural criterion for the model fit of continuous functional responses; see Ramsay and Silverman 2005, equation (16.2). For the criterion defined in (3) we choose to employ a fitting procedure with a fast and well established smoothness selection approach.

Penalties, $P_0(\boldsymbol{\beta}_0)$, $P_1(\boldsymbol{\beta}_1)$, and $P_2(\boldsymbol{\beta}_2)$ are of known functional form with the amount of shrinkage being controlled by the three scalar smoothing parameters λ_0, λ_1, and λ_2. We employ quadratic penalties, $P_0(\boldsymbol{\beta}_0) = \boldsymbol{\beta}_0^t\boldsymbol{D}_0\boldsymbol{\beta}_0$, $P_1(\boldsymbol{\beta}_1) = \boldsymbol{\beta}_1^t\boldsymbol{D}_1\boldsymbol{\beta}_1$, $P_2(\boldsymbol{\beta}_2) = \boldsymbol{\beta}_2^t\boldsymbol{D}_2\boldsymbol{\beta}_2$, where $\boldsymbol{D}_0$, $\boldsymbol{D}_1$, $\boldsymbol{D}_2$ are known penalty matrices associated with the chosen basis. Among the possible criteria for selection of the smoothing parameters, (Generalized) Cross Validation (GCV),

AIC and Restricted Maximum Likelihood (REML) are most popular. In this paper we favor REML (Reiss and Ogden 2009; Wood 2011) in a particular mixed model; our preference is also motivated by theoretical results that a restricted maximum likelihood-based selection of the smoothing parameters is more robust to moderate violations of the independent error assumption, see Krivobokova and Kauermann (2007). Estimation of smoothing parameters and model parameters are described next.

2.2 Mixed model representation of function on function regression

We note that the penalized criterion (3) with the quadratic penalty choices described in the previous section becomes

$$\frac{1}{\sigma_\epsilon^2}\sum_{i,j}||Y_i(t_j)-\mu_i(t_j;\boldsymbol{\gamma},\boldsymbol{\beta}_0,\boldsymbol{\beta}_1,\boldsymbol{\beta}_2)||^2+\frac{\lambda_0}{\sigma_\epsilon^2}\boldsymbol{\beta}_0^t\boldsymbol{D}_0\boldsymbol{\beta}_0+\frac{\lambda_1}{\sigma_\epsilon^2}\boldsymbol{\beta}_1^t\boldsymbol{D}_1\boldsymbol{\beta}_1+\frac{\lambda_2}{\sigma_\epsilon^2}\boldsymbol{\beta}_2^t\boldsymbol{D}_2\boldsymbol{\beta}_2,$$

where σ_ϵ^2 is the variance of the errors $\epsilon_i(t)$ in model (1). By denoting $\sigma_0^2=\sigma_\epsilon^2/\lambda_0$, $\sigma_1^2=\sigma_\epsilon^2/\lambda_1$, $\sigma_2^2=\sigma_\epsilon^2/\lambda_2$ and following arguments identical to those in Ruppert et al. (2003, sec 4.9) we conclude that the solution to the penalized criterion (3) is the best linear unbiased predictor in the mixed model

$$\begin{aligned} Y_i(t) &\sim N\{\mu_i(t;\boldsymbol{\gamma},\boldsymbol{\beta}_0,\boldsymbol{\beta}_1,\boldsymbol{\beta}_2),\sigma_\epsilon^2\};\\ \boldsymbol{\beta}_0 &\sim N(\mathbf{0},\sigma_0^2\boldsymbol{D}_0^{-1});\ \ \boldsymbol{\beta}_1\sim N(\mathbf{0},\sigma_1^2\boldsymbol{D}_1^{-1});\ \ \boldsymbol{\beta}_2\sim N(\mathbf{0},\sigma_2^2\boldsymbol{D}_2^{-1}), \end{aligned} \tag{4}$$

where $\boldsymbol{D}_0$, $\boldsymbol{D}_1$, and $\boldsymbol{D}_2$ are known, and shrinkage of the functional parameters is controlled by σ_0^2, σ_1^2, and σ_2^2, respectively. There is a slight abuse of notation in model (4), as matrices $\boldsymbol{D}_0$, $\boldsymbol{D}_1$, and $\boldsymbol{D}_2$ are typically not invertible. Indeed, in many cases, only a subset of the coefficients are being penalized or the penalty matrix is rank deficient. These are well known problems in penalized regression and the standard solution (Wood 2006, ch 6.6) is to either replace the inverse with a particular generalized inverse or separate the coefficients that are penalized from the ones that are not.

Replacing the penalized approach with the mixed model (4) has many favorable intended

consequences. First, models can naturally be extended in a likelihood framework to adapt to different levels of data complexity, such as the addition of smooth effects of scalar covariates. This enables us to expand the scope of function-on-function regression substantially. Second, inferential methods originally developed for mixed models transfer to function-on-function regression. In particular, approximate confidence intervals (Wahba 1983; Nychka 1988; Ruppert et al. 2003) can be obtained as a by-product of the fitting algorithm. This provides a statistically sound solution to an important problem that is currently unaddressed in function-on-function regression. We discuss in this paper implementation of this inferential tool when independent identically distributed (i.i.d.) errors in model (1) are assumed. Appendix A discusses first steps towards related inference tools for settings with non-i.i.d. errors. Moreover, tests on the shape of the association between responses and covariates are available as well. For suitably chosen bases and penalties, likelihood ratio tests of linearity versus non-linearity, or constancy versus non-constancy, of the coefficient surfaces along the lines described in Crainiceanu and Ruppert (2004), Greven, Crainiceanu, Küchenhoff, and Peters (2008) can be performed with the `R` package `RLRsim` (Scheipl, Greven and Küchenhoff, 2008).

We propose to fit model (4) using frequentist model software based on REML estimation of the variance components. We decided to use the robust `mgcv` package (Wood 2012) in `R` (R Development Core Team 2012), which is designed for penalized regression and has a built-in capability to construct the penalty matrices that are appropriate for the specified spline bases. We describe the fitting implementation in the next section.

2.3 Function-on-function regression via mixed models software

We now turn our attention to implementation. In particular, we explain how model (4) can be fit using the `mgcv` package (Wood 2012) in `R`. Note that our representation of the function on function regression model (2) and (3) is a penalized additive model where the original basis functions are re-weighted via the expressions $A_{1,l,i}(t) = \sum_{k=1}^{K} a_{1,l}(t, s_k)\widetilde{X}_{i1}(s_k)$ and $A_{2,l,i}(t) = \sum_{q=1}^{Q} a_{2,l}(t, r_q)\widetilde{X}_{i2}(r_q)$. If $a_{0,l}(\cdot)$, $a_{1,l}(\cdot, \cdot)$, and $a_{2,l}(\cdot, \cdot)$ are, for example, thin

plate spline bases, then the `mgcv` fit can simply be expressed as

```
fit <- gam(Y ~ W + s(t_0) + s(t_1, s, by=DX1) + s(t_2, r, by=DX2), method="REML").
```

This very short software line deserves an in-depth explanation to illustrate its direct connection to our mixed model representation of function on function regression. In order to fit the model, the outcome functions $Y_i(t)$ are stacked in an $nm \times 1$ dimensional vector $\boldsymbol{Y} = \{Y_1(t_1), \ldots, Y_1(t_m), Y_2(t_1), \ldots, Y_n(t_m)\}^t$, where n is the number of curves (or subjects) and m is the number of observations per curve. This vector is labeled `Y`. The $p \times 1$ dimensional vectors of covariates $\boldsymbol{W}_i$ are also stacked using the same rules used for $Y_i(t)$. More precisely, for every subject i the vector $\boldsymbol{W}_i$ is repeated m times and rows are row-stacked in an $m \times p$ dimensional matrix $\widetilde{\boldsymbol{W}}_i$. These subject specific matrices are further row-stacked across subjects to form an $mn \times p$ dimensional matrix $\boldsymbol{W}$. This matrix is labeled `W`. The next step is to define a grid of points for the smooth function $\beta_0(t)$ in model (1). The grid corresponds exactly to the stacking of the outcome functions and is defined as the $mn \times 1$ dimensional vector $\boldsymbol{t}_0 = (t_1, \ldots, t_m, \ldots, t_1, \ldots, t_m)^t$ obtained by stacking n repetitions of the grid vector $(t_1, \ldots, t_m)^t$. This vector is labeled `t_0`. The expression `s(t_0)` thus fits a penalized univariate thin-plate spline at the grid points in the vector $\boldsymbol{t}_0$.

So far, data manipulation and labeling have been quite straightforward. However, fitting the functional part of the model requires some degree of software customization. We focus on the first functional component and build three $nm \times K$ dimensional matrices: 1) $\boldsymbol{t}_1$, obtained by column-binding K copies of the $nm \times 1$ dimensional vector $\boldsymbol{t}_0$; this matrix is labeled `t_1`; 2) $\boldsymbol{s}$ obtained by row-binding nm copies of the $1 \times K$ dimensional vector $(s_1, \ldots, s_K)$; this

matrix is labeled `s`; and 3) $\widetilde{\boldsymbol{X}}_1$ the $nm \times K$ dimensional matrix

$$\begin{bmatrix} \widetilde{X}_{11}(s_1), \widetilde{X}_{11}(s_2), \ldots, \widetilde{X}_{11}(s_K) \\ \widetilde{X}_{11}(s_1), \widetilde{X}_{11}(s_2), \ldots, \widetilde{X}_{11}(s_K) \\ \ldots \\ \widetilde{X}_{11}(s_1), \widetilde{X}_{11}(s_2), \ldots, \widetilde{X}_{11}(s_K) \\ \widetilde{X}_{21}(s_1), \widetilde{X}_{21}(s_2), \ldots, \widetilde{X}_{21}(s_K) \\ \ldots \\ \widetilde{X}_{n1}(s_1), \widetilde{X}_{n1}(s_2), \ldots, \widetilde{X}_{n1}(s_K) \end{bmatrix},$$

where each row $\{\widetilde{X}_{i1}(s_1), \ldots, \widetilde{X}_{i1}(s_K)\}$ is repeated m times, for $i = 1, \ldots, n$; this matrix is labeled `DX1`. With these notations, the expression `s(t_1,s,by=DX1)` is essentially building $\sum_l A_{1,l,i}(t)\beta_{1,l}(t)$, where $A_{1,l,i}(t) = \sum_{k=1}^{K} a_{1,l}(t, s_k)\widetilde{X}_{i1}(s_k)$. Without the option `by=DX1` the expression `s(t_1,s)` would build the bivariate thin-plate spline basis $a_{1,l}(t, s_k)$, whereas adding `by=DX1` is averaging these bivariate bases along the second dimension using the weights $\widetilde{X}_{i1}(s_k)$ derived from the first functional predictor. A similar construction is done for the second functional predictor. Smoothing for all three penalized splines is done via REML, as indicated by the option `method=`"REML".

The implementation presented above is simple, but the great flexibility of the `gam` function allows multiple useful extensions. First, it is obvious that the methods and software can be adapted to a larger number of functional predictors. The number of basis functions used for each smooth can be adjusted. For example, requiring $k_0 = 10$ basis functions for a univariate thin-plate penalized spline for the $\beta_0(\cdot)$ function can be obtained by replacing `s(t_0)` with `s(t_0,k=10)`. Moreover, the implementation can accommodate unevenly sampled grid points both for the functional response and predictors. Indeed, nothing in the modeling or implementation requires equally spaced grids. Changing from (isotropic) bivariate thin-plate splines to (anisotropic) tensor product splines based on two univariate bases can be done by replacing `s(t_1,s,by=DX1)` with `te(t_1,s,by=DX1)`. This changes criterion (3) only slightly, such that $\boldsymbol{\beta}_1$ then has two additive penalties in s and t direction, respectively. We

can also easily incorporate linear effects of scalar covariates allowed to vary smoothly along t (varying coefficients), that is $z_i\beta_3(t)$, using expressions of the type `s(t_0,by=Z)`, where `Z` is obtained using a strategy similar to the one for functional predictors. For large datasets the function `bam` is more computationally efficient than `gam`.

In practice, it is useful to have a dedicated, user-friendly interface that automatically takes care of the stacking, concatenation and multiplication operations described here, calls the appropriate estimation routines in `mgcv`, and returns a rich model object that can easily be summarized, visualized, and validated. `pffr` offers a formula-based interface that accepts functional predictors and responses in standard matrix form, i.e., the i^{th} row contains the function evaluations for subject i on an index vector like $\boldsymbol{t}_0$ or $\boldsymbol{s}$. It returns a model object whose fit can be summarized, plotted and compared with other model formulations without any programming effort by the user through the convenient and fully documented functions `summary`, `plot` and `predict`. The model formula syntax used to specify models is very similar to the established formula syntax to lower the barrier to entry for users familiar with the `mgcv`-package, i.e. to specify model (2), we use

```
fit.pffr <- pffr(Ymat ~ c(W) + ff(X1mat, yind=t) + ff(X2mat, yind=t))
```

where `Ymat, X1mat, X2mat` are matrices containing the function evaluations and `ff(X, yind=t)` denotes a linear function-on-function term $\int X(s)\beta(t,s)ds$. A functional intercept $\beta_0(t)$ is included by default. The term `c(W)` corresponds to a constant effect of the covariates in $\boldsymbol{W}$, i.e., $\boldsymbol{W}\boldsymbol{\gamma}$. By default, `pffr` associates scalar covariates with an effect varying smoothly on the domain of the response, i.e., `~W` yields an effect $\boldsymbol{W}\boldsymbol{\gamma}(t)$. Our implementation also supports non-linear effects of scalar covariates z_i that may or may not be constant over the domain of the response, i.e., $f(z_i,t)$ or $f(z_i)$, specified as `s(z)` or `c(s(z))`, respectively, as well as multivariate non-linear effects of scalar covariates z_{1i}, z_{2i} that may or may not be constant over the domain of the response, i.e., $f(z_{1i},z_{2i},t)$ or $f(z_{1i},z_{2i})$, specified as `te(z_1, z_2)` or `c(te(z1, z2))`, respectively.

Prior to applying our `pffr` procedure, we recommend to center the functional predictors. For each functional predictor, the mean function is estimated (e.g. Ramsay, Hooker and

Graves 2009, sec 6.1; Bunea, Ivanescu and Wegkamp 2011) and subtracted from curve i. When the functional predictors are centered, the functional intercept has the interpretation of the overall mean outcome for observations with functional predictor values at the respective mean values.

3 Extensions

We have already seen in Section 2.3 that the class of models we can cover in our framework is much broader than model (1), including additional functional predictors, varying coefficients, non-linear effects of one or multiple scalar covariates. Here we present extensions to a few other realistic settings.

3.1 Sparsely observed functional response

Our approach is able to accommodate sparseness of the observed response trajectories. This work is the first, to the best of our knowledge, to consider a sparsely observed functional response and the general setting of multiple functional and scalar predictors. Let $\{t_{ij} : j = 1, \ldots, m_i\}$ be the set of time points at which the response for curve i is observed such that $\cup_{i=1}^{n}[\{t_{ij}\}_{j=1}^{m_i}]$ is dense in $\mathcal{T}$. PFFR can easily accommodate such a scenario with only few modifications. First, the vector of responses, labeled `Y`, has the form $\boldsymbol{Y} = \{Y_1(t_{11}), \ldots, Y_1(t_{1m_1}), Y_2(t_{21}), \ldots, Y_2(t_{2m_2}), \ldots, Y_n(t_{nm_n}))\}^t$ to accommodate subject-responses observed at different time points. Second, the vector labeled `t_0` has the form $\boldsymbol{t}_0 = (t_{11}, \ldots, t_{1m_1}, \ldots, t_{nm_n})^t$. Third, the matrix of covariates, labeled `W`, is obtained by taking m_i copies of the $1 \times p$ row vector of covariates $\boldsymbol{W}_i$ for subject i, column-stacking these into a $m_i \times p$-dimensional matrix $\widetilde{\boldsymbol{W}}_i$, and then further column-stacking these matrices across subjects. The functional components labeled `DX1` and `DX2` are constructed using a similar logic. There is no modification in the definitions of the vectors labeled $\boldsymbol{t}_1$ and $\boldsymbol{s}$.

With these adjustments, the fitting procedure can proceed with `gam` as described in Section 2.3. Section 4.2 provides simulation results obtained in this setting.

3.2 Corruptly observed functional predictors

When the functional predictors are not observed uncorrupted on a fine grid, the underlying smooth curves need to be estimated first. If data are not too sparse, pre-smoothing the trajectories using spline-based methods (Ramsay and Silverman 2005; Ramsay, Hooker and Graves 2009, ch 5) can produce good approximations for the smooth profiles. If data are sparse, it may be necessary to pool information across curves. We follow an approach similar to Yao et al. (2005a), which is built on sharing information across curves to estimate the common covariance function and uses smoothing of this covariance to take out measurement error, which induces an offset on the diagonal of the covariance. We use smoothing based on penalized splines, as previously used by Goldsmith et al. (2011), to obtain centered and reconstructed functional predictors $\widehat{X}_{i1}(s)$ and $\widehat{X}_{i2}(r)$, and use them as input values for our PFFR procedure. We present numerical results for our implementations in Section 4.3.

4 Simulation study

We conducted a simulation study to evaluate the performance of PFFR in realistic scenarios including both densely and sparsely observed functional predictors, and for a functional response that is densely or sparsely sampled.

To the best of our knowledge, there is no publicly available software for fitting the function-on-function regression with two functional predictors and scalar covariates as in model (1) other than our `pffr` function. The `fda` package (Ramsay, Wickham, Graves and Hooker 2011) in `R` includes the function `linmod`, see Ramsay, Hooker and Graves (2009, sec. 10.3), which cannot handle multiple functional predictors or effects of scalar covariates, however. Also, the `fRegress` function in the `fda` package is restricted to the concurrent model.

We thus compare PFFR to a sequence of scalar on function regressions. More precisely, for every fixed t, the function on function regression model (1) becomes a scalar on function regression that can be fit using, for example, PFR (Goldsmith et al. 2011), as implemented in

the `R` function `pfr` from the `refund` package. We only compare our approach to PFR because, to the best of our knowledge, there is no other implemented method available that can handle our general model, which incorporates multiple functional predictors and multiple scalar covariates. Thus, function on function regression can be done using a sequence of scalar on function regressions at every t. PFR provides estimates of the functional parameters $\beta_1(t,s)$ and $\beta_2(t,r)$ for every fixed t. Aggregating these functions over t leads to surface estimators that are then smoothed using a bivariate penalized spline smoother. This approach is labeled modified-PFR in the remainder of the paper. Without bivariate smoothing the modified-PFR was found not to be competitive.

4.1 Densely sampled functional predictors

The curves $Y_i(t)$ were observed on an equally spaced grid $t \in \{j/10 : j = 1, 2, ..., 60\}$ and the functional predictors were observed on equally spaced grids, but in different domains: $X_{i1}(s)$ on $s \in \{k/10 : k = 1, 2, ..., 50\}$ and $X_{i2}(r)$ on $r \in \{q/10 : q = 1, 2, ..., 70\}$. The bivariate functional parameters $\beta_1(t,s) = \cos(t\pi/3)\sin(s\pi/5)$ and $\beta_2(t,r) = \sqrt{tr}/4.2$ have comparable range and are displayed in the left panels of Figure 2. The functional intercept is $\beta_0(t) = 2e^{-(t-2.5)^2}$ and the random errors $\varepsilon_i(t_j)$ were simulated i.i.d. $N(0, \sigma^2_\epsilon)$.

We generated n functional outcomes $Y_i(t)$ from model (1) by approximating the integrals via Riemann sums with a dense grid for each domain $\mathcal{S}$ and $\mathcal{R}$. For the first functional predictor we considered the following mean zero process $V_{i1}(s) = X_{i1}(s) + \delta_{i1}(s)$, where $X_{i1}(s) = \sum_{k=1}^{E_s}\{v_{ik_1}\sin(k\pi s/5) + v_{ik_2}\cos(k\pi s/5)\}$ with $E_s = 10$, and where $v_{ik_1}, v_{ik_2} \sim$ N(0,1/k^4) are independent across subjects i. For the second functional predictor we considered $V_{i2}(r) = X_{i2}(r) + \delta_{i2}(r)$, where $X_{i2}(r) = \sum_{k=1}^{E_r}(2\sqrt{2}/(k\pi))U_{ik}\sin(k\pi r/7)$ with $E_r = 40$, and where $U_{ik} \sim$ N(0,1), and $\delta_{i1}(s), \delta_{i2}(r) \sim$ N$(0, \sigma^2_X)$ are mutually independent. More precisely, $X_{i1}(s)$ and $X_{i2}(r)$ were the true underlying processes used to generate $Y_i(t)$ from model (1) and $V_{i1}(s)$ and $V_{i2}(r)$ were the actually observed functional predictors. The choice of $X_{i1}(s)$, $\beta_1(\cdot,\cdot)$ and $\beta_2(\cdot,\cdot)$ is similar to the choices in Goldsmith et al. (2011), whereas $X_{i2}(r)$ is a modification of a Brownian bridge. For the scalar covariates, we considered a

binary variable $W_1 = 1\{\text{Unif}[0,1] \geq .75\}$, and a continuous variable $W_2 \sim N(10, 5^2)$. A univariate cubic B-spline basis with $\kappa_0 = 10$ basis functions with second-order difference penalty was used to fit $\beta_0(\cdot)$. Tensor products of cubic B-splines with $\kappa_1 = \kappa_2 = 25$ basis functions and second-order difference penalties in both directions were used to fit $\beta_1(\cdot,\cdot)$ and $\beta_2(\cdot,\cdot)$. However, for more complex functional parameters, increasing the number of basis functions may be necessary to capture the increased complexity. Increasing the number of basis functions for the bivariate smoothers from 25 to 100 did not significantly affect the fit or computation times.

We considered all possible combinations of the following choices:

1. Number of subjects: (a) $n = 100$ and (b) $n = 200$.

2. Functional predictors: (a) noiseless $\sigma_X = 0$ and (b) noisy $\sigma_X = 0.2$.

3. Standard deviation for the error: $\sigma_\epsilon = 1$.

4. Effects of the scalar covariates:

    (a) no scalar covariates, and

    (b) scalar covariates W_1 and W_2 with $\gamma_1 = 1$ and $\gamma_2 = -0.5$.

For illustration, Figure 1 displays one simulated data set for scenario 4(a). Curves for three subjects are highlighted in color, with each color representing one subject across the three panels.

The combination of the various choices provides eight different scenarios and for each scenario we simulated 500 data sets. PFFR uses two steps for case 2(b). First, the functional predictors were estimated as $\widehat{X}_{i1}(s)$ and $\widehat{X}_{i2}(r)$ using the smoothing approach previously used in Goldsmith et al. (2011). Second, these estimated functions were used instead of $X_{i1}(s)$ and $X_{i2}(r)$ when fitting model (1).

We computed the integrated mean squared error (IMSE), integrated squared bias (IBIAS^2), and integrated variance (IVAR), where $\text{IMSE}\{\widehat{\beta}(t,s)\} = \int_{\mathcal{T}}\int_{\mathcal{S}} \text{E}[\{\widehat{\beta}(t,s) - \beta(t,s)\}^2]dtds$, $\text{IBIAS}^2\{\widehat{\beta}(t,s)\} = \int_{\mathcal{T}}\int_{\mathcal{S}}[\text{E}\{\widehat{\beta}(t,s)\} - \beta(t,s)]^2 dtds$, and $\text{IVAR}\{\widehat{\beta}(t,s)\} = \int_{\mathcal{T}}\int_{\mathcal{S}} \text{Var}\{\widehat{\beta}(t,s)\}dtds$.

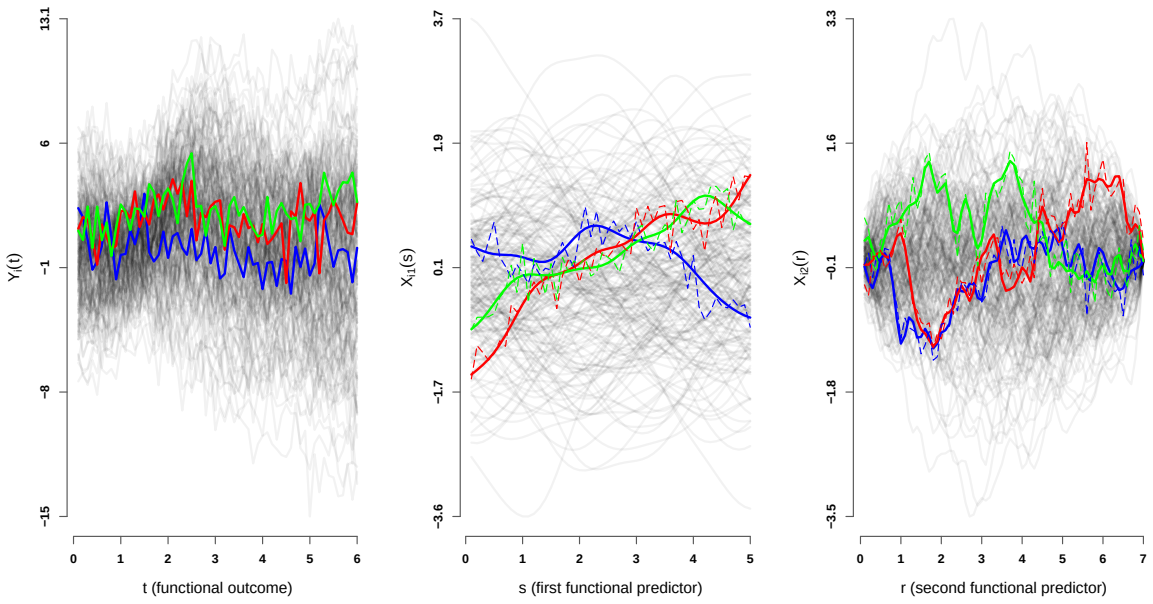

Figure 1: The left panel displays a sample of 200 simulated functional responses $Y_i(t)$ with $\sigma_\epsilon = 1$. The middle and right panels display 200 simulated functions from $X_{i1}(s)$ and $X_{i2}(r)$, respectively, highlighting three examples of the functional predictors with no error (solid) and with measurement error $\sigma_X = 0.2$ (dashed).

Here $\mathrm{E}\{\widehat{\beta}(t,s)\}$ and $\mathrm{Var}\{\widehat{\beta}(t,s)\}$ were estimated by the empirical mean and variance of $\widehat{\beta}(t,s)$ in 500 simulations. To characterize the properties of the pointwise confidence intervals we report the integrated actual pointwise coverage (IAC) and integrated actual width (IAW), where $\mathrm{IAC} = \mathrm{E}[\int_{\mathcal{T}}\int_{\mathcal{S}} 1\{\beta(t,s) \in CI_p(t,s)\}dtds]$, where CI_p is the pointwise approximate confidence interval for the model parameter $\beta(t,s)$. For example, an approximate 95% pointwise confidence interval $CI_p(t_0,s_0)$ for $\beta_1(t_0,s_0)$ can be constructed as $\widehat{\beta}_1(t_0,s_0) \pm 1.96\ \widehat{\mathrm{sd}}\{\widehat{\beta}_1(t_0,s_0)\}$. As $\widehat{\beta}_1(t_0,s_0) = \sum_{l=1}^{\kappa_1} a_{1,l}(t_0,s_0)\widehat{\beta}_{1,l}$, for every (t_0,s_0), $\widehat{\mathrm{sd}}\{\widehat{\beta}_1(t_0,s_0)\} = \sqrt{\boldsymbol{a_1(t_0,s_0)}\widehat{\boldsymbol{\Sigma}}_1\boldsymbol{a_1^t(t_0,s_0)}}$, where $\widehat{\boldsymbol{\Sigma}}_1$ is the estimated covariance matrix of $\widehat{\boldsymbol{\beta}}_1$, and $\boldsymbol{a_1(t_0,s_0)} = \{a_{1,l}(t_0,s_0)\}_l$. We use the Bayesian posterior covariance matrix, see Ruppert et al. (2003). The length of this confidence interval is $3.92\ \widehat{\mathrm{sd}}\{\widehat{\beta}_1(t_0,s_0)\}$ and $\mathrm{IAW} = \mathrm{E}[3.92\int_{\mathcal{T}}\int_{\mathcal{S}}\widehat{\mathrm{sd}}\{\widehat{\beta}_1(t,s)\}dtds]$. As a measure of accuracy of the fit we provide the functional R^2, denoted by fR^2, and computed as $\mathrm{fR}^2 = 1 - \{\sum_i\sum_j[Y_i(t_j) - \widehat{Y}_i(t_j)]^2\}/\{\sum_i\sum_j[Y_i(t_j) - \widehat{\mu}_Y(t_j)]^2\}$, where $\widehat{\mu}_Y(t_j) = \sum_{i=1}^n Y_i(t_j)/n$, and $\widehat{Y}_i(t_j)$ are the predicted $Y_i(t_j)$ using the fitted model. Table 1 compares the average of these measures over simulations, whereas Appendix B displays boxplots of these measures for the bivariate functional parameters calculated for each data set. Overall, our results indicate that PFFR outperforms the modified-PFR. To provide the intuition behind these results, Figure 2 displays the fits using PFFR and PFR obtained in one simulation for the setting $n = 200$, $\sigma_\epsilon = 1$, $\sigma_X = 0$ and case 4(a). Both methods capture the general features of the true parameter surfaces well, but the modified-PFR method does not borrow strength between the neighboring values of the functional outcomes since it is derived from PFR. As our results show, this causes unnecessary roughness of the estimates along the t dimension. In contrast, PFFR provides a much smoother surface that better approximates the shape of the true underlying functions.

Results in Table 1 can be summarized as follows. PFFR performs better than modified-PFR in terms of IMSE for all the estimated parameters irrespective of the noise level in the functional predictors and the presence of additional non-functional covariates; see the

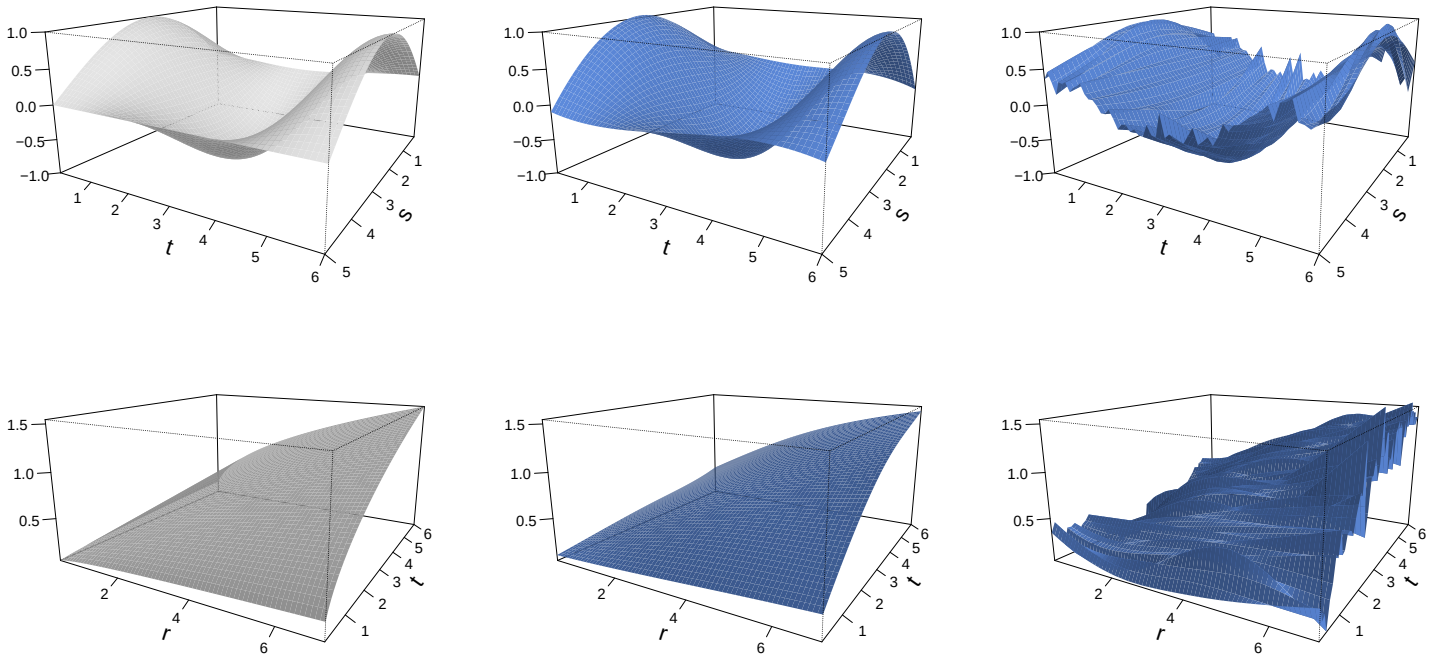

Figure 2: Displayed are the bivariate functional parameters $\beta_1(t,s)$ (top panels) and $\beta_2(t,r)$ (bottom panels): true values (gray, left panels), estimates via PFFR (blue, middle panels), and estimates via PFR (blue, right panels). Scenario: $n = 200$, $\sigma_\epsilon = 1$, $\sigma_X = 0$ and case 4(a).

Table 1: Results for function-on-function regression with densely sampled functional predictors, based on 500 simulations.

		$\sigma_X = 0$						$\sigma_X = 0.2$					
			($\times 10^3$)						($\times 10^3$)				
	Method	$\sqrt{IMSE}$	$\sqrt{IVAR}$	$\sqrt{IBIAS^2}$	IAC	IAW	fR^2	$\sqrt{IMSE}$	$\sqrt{IVAR}$	$\sqrt{IBIAS^2}$	IAC	IAW	fR^2
Scenario 4(a)													
	n=100												
$\beta_0(t)$													
	PFFR	39.20	35.80	15.97	0.92	0.14		42.12	38.97	15.99	0.90	0.14	
	modified-PFR	40.43	40.06	5.46	1.00	0.40		43.86	43.47	5.78	0.99	0.40	
$\beta_1(t,s)$													
	PFFR	42.54	41.64	8.70	0.98	0.17	92.55%	49.49	48.72	8.71	0.96	0.18	92.40%
	modified-PFR	134.68	26.13	132.12	0.77	0.28	92.34%	137.56	27.95	134.69	0.76	0.28	92.19%
$\beta_2(t,r)$													
	PFFR	37.34	25.07	27.67	0.88	0.08		41.63	31.82	26.84	0.83	0.09	
	modified-PFR	65.00	28.59	58.37	0.97	0.35		68.18	31.38	60.53	0.96	0.35	
	n=200												
$\beta_0(t)$													
	PFFR	29.49	25.53	14.76	0.90	0.10		31.29	27.62	14.69	0.88	0.10	
	modified-PFR	27.69	27.46	3.58	1.00	0.28		29.84	29.64	3.45	0.99	0.28	
$\beta_1(t,s)$													
	PFFR	32.92	32.08	7.38	0.97	0.13	92.59%	37.89	37.17	7.34	0.96	0.14	92.44%
	modified-PFR	99.40	23.51	96.58	0.83	0.25	92.45%	103.25	25.14	100.14	0.82	0.25	92.29%
$\beta_2(t,r)$													
	PFFR	30.22	18.50	23.89	0.89	0.06		34.46	24.60	24.14	0.83	0.07	
	modified-PFR	46.30	22.62	40.39	0.98	0.30		49.03	25.16	42.08	0.98	0.30	
Scenario 4(b)													
	n=100												
γ_1													
	PFFR	30.40	30.40	0.21	0.96	0.12		42.95	42.93	1.18	0.85	0.12	
	modified-PFR	33.90	33.90	0.27	1.00	0.94		44.45	44.42	1.49	1.00	0.95	
γ_2													
	PFFR	2.81	2.80	0.12	0.92	0.01		3.78	3.77	0.27	0.85	0.01	
	modified-PFR	3.10	3.10	0.05	1.00	0.08		3.96	3.95	0.35	1.00	0.08	
$\beta_0(t)$													
	PFFR	53.93	51.43	16.23	0.79	0.14	94.94%	65.21	63.06	16.58	0.70	0.14	94.84%
	modified-PFR	115.68	112.44	27.17	0.99	1.15	94.75%	122.83	119.73	27.43	0.99	1.17	94.64%
$\beta_1(t,s)$													
	PFFR	43.42	42.40	9.33	0.97	0.17		51.31	50.53	8.87	0.95	0.18	
	modified-PFR	135.36	26.70	132.70	0.77	0.28		138.43	28.72	135.42	0.76	0.28	
$\beta_2(t,r)$													
	PFFR	36.54	24.59	27.02	0.89	0.08		42.70	32.64	27.53	0.82	0.08	
	modified-PFR	65.72	29.16	58.89	0.97	0.35		68.74	31.66	61.02	0.96	0.35	
	n=200												
γ_1													
	PFFR	21.41	21.40	0.34	0.94	0.08		29.83	29.78	1.59	0.86	0.08	
	modified-PFR	23.25	23.25	0.56	1.00	0.65		30.15	30.10	1.74	1.00	0.66	
γ_2													
	PFFR	1.83	1.83	0.08	0.95	0.01		2.57	2.57	0.05	0.85	0.01	
	modified-PFR	1.94	1.94	0.10	1.00	0.05		2.65	2.65	0.03	1.00	0.05	
$\beta_0(t)$													
	PFFR	38.42	35.45	14.81	0.80	0.10	94.97%	47.04	44.60	14.96	0.71	0.10	94.86%
	modified-PFR	80.07	78.79	14.25	1.00	0.80	94.84%	84.52	83.03	15.78	0.99	0.81	94.74%
$\beta_1(t,s)$													
	PFFR	33.00	32.22	7.10	0.97	0.13		39.09	38.34	7.65	0.95	0.14	
	modified-PFR	100.16	23.09	97.46	0.82	0.25		103.74	25.79	100.49	0.82	0.25	
$\beta_2(t,r)$													
	PFFR	30.39	18.91	23.79	0.89	0.06		34.45	25.33	23.35	0.84	0.07	
	modified-PFR	46.65	22.85	40.67	0.98	0.30		49.86	25.62	42.77	0.98	0.30	

columns labeled IMSE. Appendix B shows that, for the bivariate functional parameters, the variability of the MSEs of PFFR and modified-PFR is comparable, and that the median MSE of PFFR is smaller than that of modified-PFR. As the number of subjects increases, IMSE for both methods decreases, confirming that in the settings considered, the methodology yields consistent estimators; this is expected to hold more generally, see, for example Claeskens, Krivobokova, and Opsomer (2009), for the asymptotics of penalized splines. In terms of the performance of the pointwise approximate confidence intervals, the PFFR intervals are reasonably narrow and have a coverage probability that is relatively close to the nominal level; see the columns corresponding to IAW and IAC in Table 1. For PFFR confidence intervals, as the number of subjects increases, IAW decreases, as expected, while maintaining IAC close to the nominal level. Overall, when functional predictors are measured with error, the estimation of parameters tends to deteriorate slightly. However, PFFR continues to outperform modified-PFR. The accuracy of the fit seems similar for PFFR and modified-PFR, as illustrated by fR^2.

The only available alternative to PFFR yields inferior results. First, fit results tend to be rougher and may require additional and specialized smoothing. Second, running a large number of scalar on function regressions leads to longer computation times. Third, confidence intervals are not easy to obtain, which may further affect computation times. For these reasons, studying their properties via simulations is computationally hard.

4.2 Sparsely sampled functional response

Consider again scenario 4(a) described in Section 4.1 and assume that for each subject i, the functional response is observed at randomly sampled m_i points from $t \in \{j/10, j = 1, 2, ..., 60\}$. Estimation of the parameter functions was carried out using our PFFR approach with $\kappa_0 = 10$ basis functions for the univariate spline basis, and $\kappa_1 = \kappa_2 = 25$ for the bivariate spline bases. The estimates were then evaluated using the same measures as described in Section 4.1. In the situation of sparsely sampled functional response, our method does not have any competitors.

Table 2 shows the results for two sparsity levels, $m_i = 20$ and $m_i = 6$. As expected, the sparsity of the functional response affects both the bias and the variance of the parameter estimators, as well as the width of the confidence intervals; compare Table 2 with the PFFR results of Table 1 corresponding to scenario 4(a) and $n = 200$. Nevertheless PFFR continues to show very good performance for many levels of missingness on the functional response with coverage of the confidence intervals similar to before.

Table 2: Results for function-on-function regression with sparsely sampled functional response and densely sampled functional predictors, based on 500 simulations.

		$\sigma_X = 0$					$\sigma_X = 0.2$				
		($\times 10^3$)					($\times 10^3$)				
	Method	$\sqrt{IMSE}$	$\sqrt{IVAR}$	$\sqrt{IBIAS^2}$	IAC	IAW	$\sqrt{IMSE}$	$\sqrt{IVAR}$	$\sqrt{IBIAS^2}$	IAC	IAW
Scenario 4(a)											
	n=200										
$m_i = 20$											
$\beta_0(t)$	PFFR	48.00	44.76	17.31	0.91	0.17	49.07	45.87	17.42	0.91	0.17
$\beta_1(t,s)$	PFFR	49.06	47.83	10.90	0.97	0.20	52.47	51.33	10.91	0.97	0.20
$\beta_2(t,r)$	PFFR	40.49	28.91	28.35	0.89	0.10	43.78	31.68	30.21	0.85	0.10
$m_i = 6$											
$\beta_0(t)$	PFFR	83.77	79.32	26.92	0.92	0.29	85.13	80.71	27.06	0.92	0.30
$\beta_1(t,s)$	PFFR	79.39	75.62	24.15	0.97	0.29	81.99	78.20	24.63	0.97	0.30
$\beta_2(t,r)$	PFFR	55.93	43.67	34.93	0.88	0.15	57.81	44.68	36.68	0.87	0.15

4.3 Functional predictors sampled with moderate sparsity

The sparse design for the functional predictors was generated by starting with the scenario 4(a) described in Section 4.1 with few changes. The number of eigenfunctions for the two functional predictors was set to $E_s = 2$ and $E_r = 4$ respectively. For each functional predictor curve i, $1 \leq i \leq n$, we randomly sampled K_i points from $s \in \{k/10, k = 1, 2, ..., 50\}$ and Q_i points from $r \in \{q/10, q = 1, 2, ..., 70\}$. We apply the same smoothing approach used in Goldsmith et al. (2011) to reconstruct the trajectories of each functional predictor, by making use of the `R` code supplied with the paper.

For illustration, Figure 3 displays the simulated data and predicted values for 4 curves, 2 for process $X_{i1}(s)$ displayed in the two leftmost columns, and 2 for process $X_{i2}(r)$ displayed in the two rightmost columns. The true underlying data are the black dotted lines, the actual observed data are the red dots and the predicted curves are the solid red lines. The visual inspection of these plots indicates that the smoothing procedure recovers the underlying

signal quite well. This is one of the main reasons PFFR continues to perform well for these scenarios.

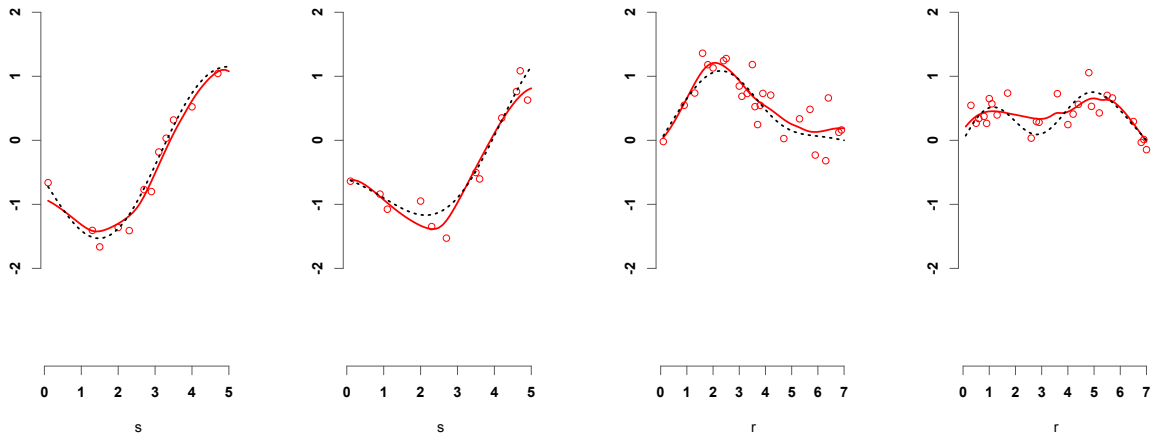

Figure 3: Prediction of trajectories for functional predictors in simulation settings with $K_i = 12$, $Q_i = 25$ points per curve, and $\sigma_X = 0.20$. Shown are: $X_{i1}(s_{ik})$ for two subjects (left two panels) and $X_{i2}(r_{iq})$ for the same subjects (right two panels). For each panel we have the true signal (black, dotted lines), the observed signal (red points), and the predicted signal (red, solid lines).

Table 3 displays the results for our model parameters for the case of functional predictors observed with moderate sparsity, $n = 200$ subjects and different sparsity levels. Both the IMSE and the coverage performance of the PFFR are affected by the sparsity of the predictor function. Due to the sparsity of the functional predictors, a reduced number of basis functions is used for estimating the bivariate parameters. Throughout this simulation exercise we used $\kappa_0 = 10$ basis functions for the univariate basis, and $\kappa_1 = \kappa_2 = 20$ basis functions for the two bivariate bases.

5 Application to DTI-MRI tractography

We consider a brain tractography study of multiple sclerosis (MS) patients. White matter tracts consist of axons that connect nerve cells and transmit information via electrical nerve

Table 3: Results for function-on-function regression with densely sampled functional response and functional predictors sampled with moderate sparsity, based on 500 simulations.

		$K_i = 15$, $Q_i = 30$						$K_i = 12$, $Q_i = 25$					
		($\times 10^3$)						($\times 10^3$)					
	Method	$\sqrt{IMSE}$	$\sqrt{IVAR}$	$\sqrt{IBIAS^2}$	IAC	IAW	fR^2	$\sqrt{IMSE}$	$\sqrt{IVAR}$	$\sqrt{IBIAS^2}$	IAC	IAW	fR^2
Scenario 4(a)													
	n=200												
$\beta_0(t)$													
	PFFR	35.67	32.46	14.78	0.84	0.10		38.42	35.43	14.83	0.82	0.11	
	modified-PFR	33.93	33.73	3.65	0.99	0.29		36.61	36.42	3.74	0.99	0.29	
$\beta_1(t,s)$													
	PFFR	69.83	68.32	14.44	0.88	0.18	92.13%	80.75	79.38	14.80	0.86	0.19	91.96%
	modified-PFR	123.87	35.44	118.69	0.78	0.27	91.97%	131.04	38.28	125.33	0.76	0.27	91.81%
$\beta_2(t,r)$													
	PFFR	51.51	49.71	13.48	0.72	0.08		55.99	54.48	12.94	0.70	0.08	
	modified-PFR	61.97	33.82	51.93	0.97	0.32		64.94	35.41	54.43	0.97	0.32	

impulses. Axons are covered with a white fatty coating called myelin, which facilitates the transmission of neuronal signals. Multiple sclerosis is a demyelinating autoimmune-mediated disease that is associated with brain lesions and results in severe disability. Little is known about in-vivo demyelination including whether it is a global or local phenomenon, whether certain areas of the brain demyelinate faster, or whether lesion formation in certain areas is associated with demyelination in other areas of the brain. Here we attempt to provide an answer to some of these questions using function on function regression.

Diffusion tensor imaging (DTI) is one way to measure proxies of demyelination by quantifying the water diffusion. Changes in water diffusion in the brain could potentially be associated with demyelination. DTI is a magnetic resonance imaging (MRI) technique that, at some level of output complexity, estimates the water diffusion at every voxel using its first three directions of variation (Basser, Mattiello and LeBihan 1994; Basser, Pajevic, Pierpaoli and Duda 2000). This information is then collected and integrated with biological knowledge to produce tracts, or bundles of axons thought to serve a similar purpose. Various water diffusion properties along these tracts are then estimated and are thought to provide localized information about changes in water diffusion and, implicitly, of demyelination processes. Some measures of diffusion are fractional anisotropy, parallel diffusivity, and perpendicular diffusivity. For example, fractional anisotropy is a function of the three eigenvalues of the estimated diffusion process that is equal to zero if water diffuses perfectly isotropically (Brownian motion) and to one if water diffuses anisotropically (perfectly organized and synchronized movement of all water molecules in one direction). In our study we focus on the

water diffusion properties along three major and well identified white matter tracts: corticospinal, corpus callosum, and optical radiation. Tracts are registered within and between subjects using biological landmarks identified by an experienced neuroradiologist. For the purpose of this application we consider averages of water diffusion measurements along two of the dimensions, which results in a functional observation with scalar argument.

Very little is known about the microstructure changes in the white matter of people who suffer from MS. We focus on the corpus callosum (CCA), corticospinal (CTS) and optical radiation (OPR) tracts because they are all major tracts that are easy to recognize and identify on the MRI scans. Thus, our approaches are less prone to location and replication errors typically associated with brain tractography based on diffusion tensor images (DTI-MRI).

Our goal here is mainly exploratory, as we are trying to further understand the spatial and temporal course of the disease. Indeed, MS is typically associated with lesions and axon demyelination in the corpus callosum. In advanced stages of MS, there is evidence for significant neuronal loss in the corpus callosum (Evangelou et al. 2000). Following Tievsky et al. (1999) and Song et al. (2002), we use fractional anisotropy (FA) as our proxy variable for demyelination of the white matter tracts, and assume larger FA values are closely associated with less demyelination and fewer lesions. Thus, in our first regression model we are investigating whether spatial associations between demyelination along the CTS, OPR and CCA are observed. For example, an association between the inferior CTS and the posterior CCA (splenium) may indicate specific spatial propagation of the disease. Such findings could generate a new set of targeted, well-defined spatial associations hypotheses. We also explore the potential temporal associations between fractional anisotropy at different visits.

Our study comprises of 160 MS patients and 42 healthy controls, who are observed at multiple visits; see Greven, Crainiceanu, Caffo and Reich (2010), Goldsmith at el. (2011), Staicu, Crainiceanu, Ruppert and Reich (2011). The work in Goldsmith at el. (2011) used parallel diffusivity within left intracranial CTS tracts to predict MS cases and controls. In

this paper we consider fractional anisotropy (FA) for several tracts, observed at the first two visits. For illustration, the top panels in Figure 4 display the FA in the MS group for three tracts - CCA tract (left), CTS tract (middle) and OPR tract (right) - at the baseline visit. Depicted in red/blue/green are the FA measurements corresponding to three subjects (subjects are color coded). Our first objective is to regress the FA for the CCA tract (e.g. the red curve in the left panel) onto the FA for the CTS tract (e.g. the red curve in the middle panel) and the FA for the OPR tract (e.g. the red curve in the right panel). This will allow us to estimate the local associations between FA of the various tracts. Our second objective is to study how the FA for one tract, at a current visit, relates to the FA for the same tract and for a different tract at the previous visit. By way of example, we focus on the MS group and consider regression models of the FA for the left CTS tract at a second visit on the FA for the left CTS tract and the FA for the left OPR tract at the baseline visit; see Figure 6.

Consider first the situation when the outcome of interest is the FA for the CCA tract, and the functional predictors are the FA for the left CTS and left OPR tracts. The first step is to de-noise and deal with missing data in the functional predictors, which is discussed in Section 3. Following the notation in Section 2, $Y_i(t)$ denotes the FA for the CCA tract at location t, $X_{i1}(s)$ and $X_{i2}(r)$ denote the centered de-noised FA for the left CTS tract at location s, and for the left OPR tract at location r, respectively, of subject i. We use regression model (1), where $\boldsymbol{W}_i$ is the vector of the covariates age and gender of subject i. To obtain the estimates of the regression functions $\beta_0(\cdot)$, $\beta_1(\cdot,\cdot)$ and $\beta_2(\cdot,\cdot)$ as well as the regression parameters $\boldsymbol{\gamma}$ we use the methodology described in Section 2.

Figure 5 shows the regression function estimates along with information on their pointwise 95% confidence intervals for each of the two groups: MS subjects (top row) and controls (bottom row). We begin with $\widehat{\beta}_0(t)$, which accounts for a relative R^2 of 28.9% for the MS group and 38% for the control group. For the MS group, the estimated overall mean of the FA for the CCA tract has a wavy shape, with two main peaks, one near the beginning of

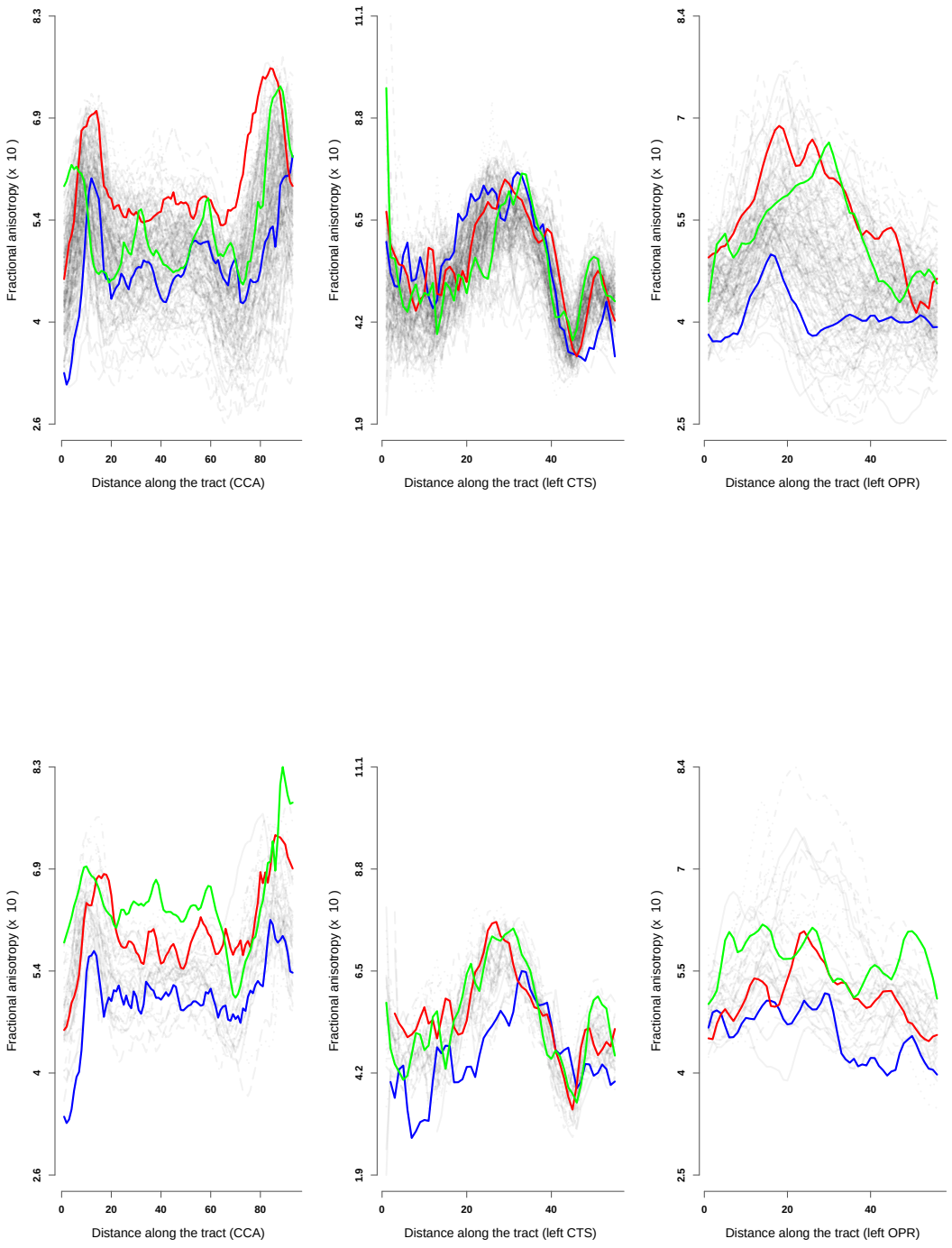

Figure 4: FA in the CCA tract (left panels), FA in the left CTS tract (middle) and FA in the left OPR tract (right panels) for the MS patients (top panels) and controls (bottom panels) observed at the baseline visit. In each of the top and bottom panels, depicted with red/green/blue are the FA measurements for three subjects, with each color representing a subject.

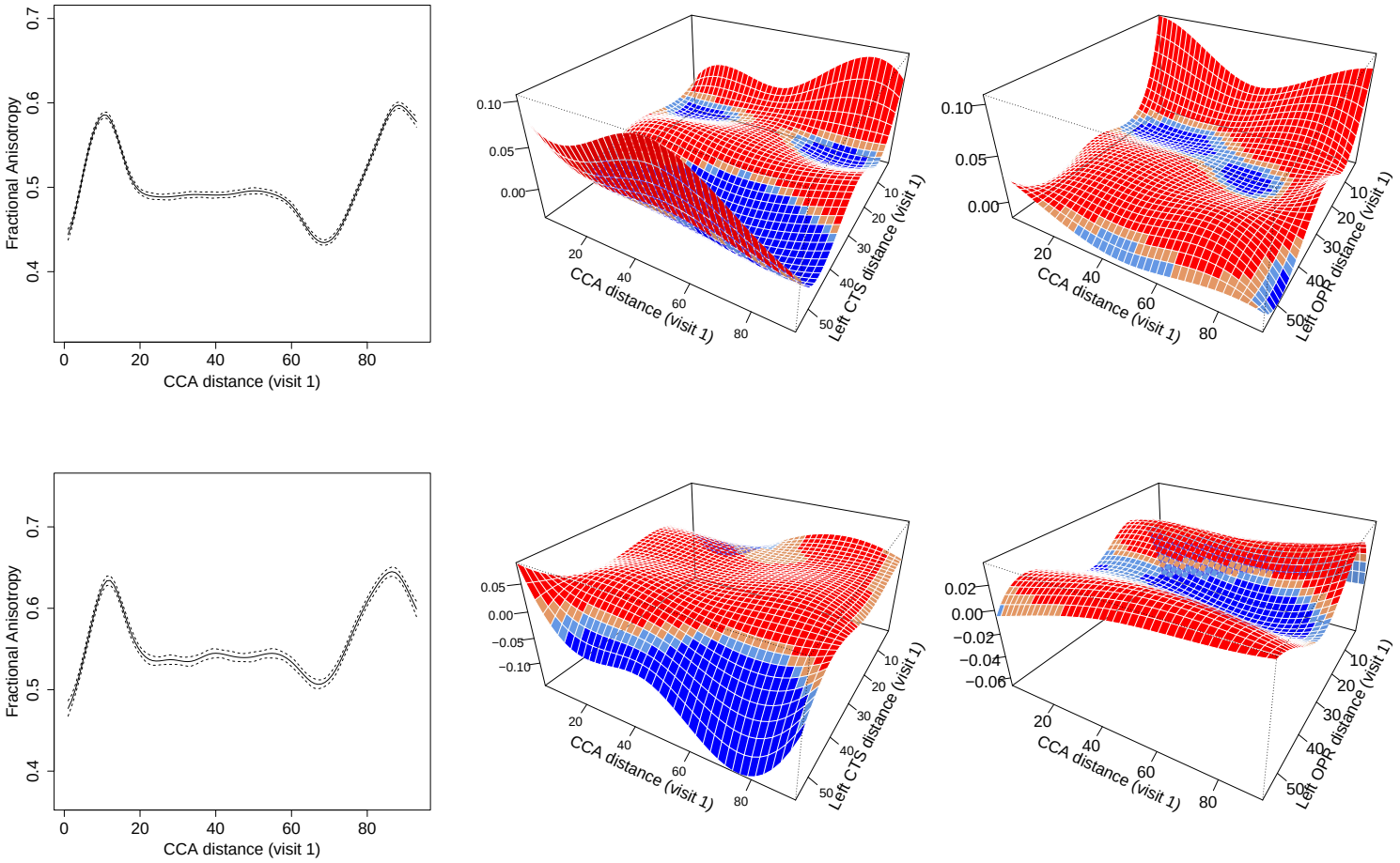

Figure 5: Estimates of the regression functions $\widehat{\beta}_0$ (left colum), $\widehat{\beta}_1$ (middle column) and $\widehat{\beta}_2$ (right colum) corresponding to the MS group (top row) and the control group (bottom row). The dashed lines (left column) correspond to the pointwise 95% confidence intervals. Also displayed are different colors of the facets of the surface mesh: red/blue for more than approximately two standard errors above/below zero, and light red/blue for less than approximately two standard errors above/below zero.

the tract, around location 10, and one at the end, around 90. For the control group, the estimated mean function shows similar patterns and we observe lower FA values for the MS group than for the control; this is biologically plausible, as FA values tend to decrease in MS-affected subjects due to lesions or demyelination. Next, we consider the effect of the FA for the left CTS tract; the relative R^2 corresponding to this predictor is 10.8% for the MS group and 15.9% for the control group. The estimated function $\widehat{\beta}_1(t, s)$, displayed in the middle column of Figure 5, shows the association between the FA for the CTS tract at location s and the FA of the CCA tract at location t.

We display information on the pointwise 95% confidence intervals under independence assumption of the residuals by using different colors of the facets of the surface mesh. Specifically, 'red' represents positive values, $\widehat{\beta}_1(t, s) > 0$, that are found significant, 'blue' represents significant negatives, $\widehat{\beta}_1(t, s) < 0$; 'light red' and 'light blue' correspond to the remaining positive and negative values, respectively. These effects provide guidance concerning the strength of the association for exploratory purposes.

We can see that, for the control group, most of the CTS (locations below 40) FA values are positively associated with FA values in the CCA, indicating that demyelination, e.g. due to age might occur simultaneously in those tracts. The coefficient surface for the control group is mostly constant, indicating a spatially homogeneous association for most regions of these two tracts. In MS patients, on the other hand, there are areas (30 to 45, and around 10) of the CTS for which FA measures show negative associations with the FA values in the CCA. This might indicate regions of those tracts where demyelination due to lesions typically occurs only in one tract at a time, while stronger positive associations may suggest that lesions often occur simultaneously in those areas. Our results support the expected pattern that the pathological demyelination in MS patients is a more localized phenomenon than demyelination in control patients.

These findings, while exploratory, may yield new insights into the motivating question of whether lesion formation in certain areas of the brain is associated with demyelination in other areas of the brain and can lead to hypotheses for further studies.

We examine the effect of the FA for the left OPR tract, as estimated by $\widehat{\beta}_2(t,r)$. The profile of the left OPR tract seems to be less predictive for the response for the control group (relative R^2 is 12.6%) and more predictive for the MS group (relative R^2 is 25.7%). Here, the associations between the FA for the OPR and the FA for the CCA seem somewhat similar between the two groups, but more pronounced for the MS patients, especially for the inferior OPR; see Figure 5, right panel.

The estimates of the additional covariates, age and gender (reference group: female), are $0.0002^*(0.00004)$ and $0.0008(0.0009)$, for the MS group, and $-0.0003^*(0.00005)$ and $-0.0052^*(0.0018)$ for the control group, respectively, where the asterisks indicate significance at level 0.01. Standard errors are displayed within brackets. Negative age effects as seen in the controls are expected, as myelination and FA values tend to decrease with age even in healthy subjects.

We further consider the situation when the outcome of interest is the FA for the left CTS at the current visit, and the functional predictors are the FA for the same tract and the FA for the left OPR tract, measured at the previous visit. We focus on the MS group of 106 patients with at least two visits, due to the lack of longitudinal data in the control group. As an initial step, the functional predictors are de-noised as discussed earlier. The second step is to apply function on function regression. The overall mean function, $\widehat{\beta}_0(t)$ accounts for 61.4% of the variability in the response. The FA measurement in the CTS is predictive of the FA measurement at the next visit to some extent (12.0% relative R^2), while the FA in the OPR doesn't seem to contain much additional information (1.5% relative R^2). Figure 7 shows the estimates of the three regression functions, and information on the pointwise 95% confidence intervals under independence assumption of the residuals by using different colors. The estimated regression function $\widehat{\beta}_1(t,s)$ looks relatively flat with the diagonal ends pulled upward. As expected, high values in the CTS's FA at the previous visit are associated with high values in the CTS's FA at the current visit for the tract locations that are close to one another. The effect of the OPR's FA at the previous visit on the CTS's FA at the current visit is considerably smaller; see the right plot of Figure 7. The estimates of the additional

covariates, age and gender, are $-0.0001(0.0001)$ and $-0.0063^{*}(0.0016)$ respectively, where the asterisk indicates significance at level 0.01.

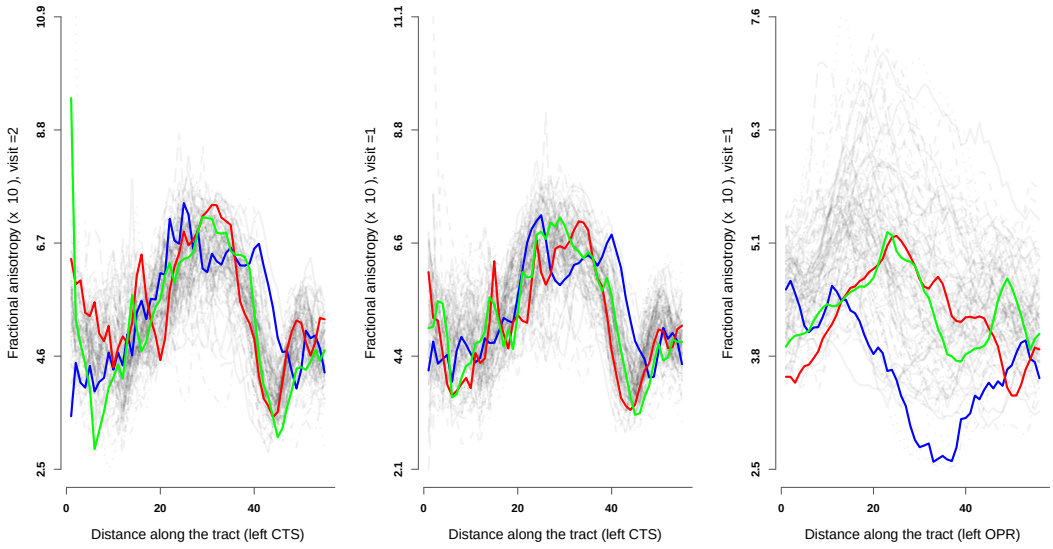

Figure 6: FA in the left CTS tract at the second visit (left panel), FA in the left CTS tract at the first visit (middle panel) and FA in the left OPR at the first visit (right panel) for the MS patients.

6 Discussion

Relating a functional predictor to the functional response through a bivariate parameter function implies a cumulative effect. There are some special cases when such an assumption is not warranted. For example, suppose that both the outcome, $Y(t)$, and a predictor process, $X(s)$, are observed on the same time domain $[0, T]$. In such a case, one can reasonably assume that the value of the outcome process $Y(t)$ depends on the past values of the predictor process $X(s), s \leq t$, see, for example Malfait and Ramsay (2003). However, it may not be reasonable to argue that $Y(t)$ is predicted by the future values $X(s), s > t$. This case is

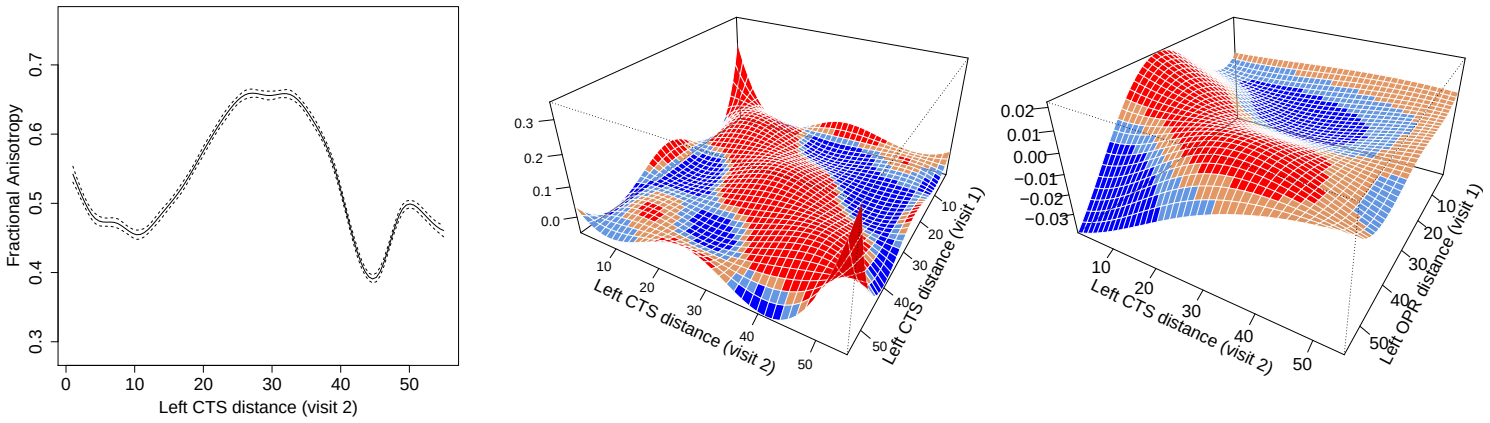

Figure 7: Estimates of the regression functions $\widehat{\beta}_0$ (left), $\widehat{\beta}_1$ (middle) and $\widehat{\beta}_2$ (right), when the outcome is FA for the left CTS tract on the second visit, and the predictors are FA for the left CTS and FA for the left OPR at the baseline visit. The dashed lines (left panel) correspond to the pointwise 95% confidence intervals. Also displayed are different colors of the facets of the surface mesh: red/blue for more than approximately two standard errors above/below zero, and light red/blue for less than approximately two standard errors above/below zero.

special, because both processes are observed on the same scale, time, which is inherently irreversible. In many cases, including the one considered in this paper, the processes are observed on different domains and the arguments of the processes are reversible. Thus, we consider the association between any FA value on the corpus callosum and any value on the corticospinal tract or optical radiations tract. In simpler contexts, this cumulative model has been considered before, see, for example, Ramsay and Silverman (2005, ch 16) and Yao, Müller and Wang (2005b).

We would like to emphasize that our approach can easily be adapted to cases where time-irreversibility-like problems occur. Indeed, we have already developed and implemented models that accommodate limited integration ranges, e.g. to allow for cumulative effects up to the current index value $\int_{s \leq t_{ij}} \beta_1(t_{ij}, s) X_{i1}(s)\, ds$. Such extensions allow for temporal covariates and responses observed on the same domain where features of the responses cannot be influenced by future covariate values. An implementation of this extension will be included in a future version of `pffr`.

This paper develops a novel method for function-on-function regression that: 1) applies

to one or multiple functional predictors observed on the same or different domains as the functional response; 2) incorporates scalar predictors; 3) is fitted automatically using existing software; 4) produces likelihood-based approximate confidence intervals as a by-product of the mixed model framework; 5) allows likelihood-based testing; 6) applies to sparsely and/or noisily observed functional predictors; and 7) accommodates a sparsely sampled functional response. Our approach to function-on-function regression provides a framework to further consider other settings than the ones considered in this paper. For example, resampling methods applied in the context of our proposed methodology, discussed in Appendix A, provide inference tools for non-i.i.d. errors scenarios.

PFFR was applied and tested in a variety of scenarios and showed good results in both simulations and a medical application. Equally important PFFR is an easy to use automatic fitting procedure implemented in the `pffr` function of the `R` package `refund`.

Acknowledgements

Staicu's research was supported by U.S. National Science Foundation grant number DMS 1007466 and by the NCSU Faculty Research and Professional Development grant. Fabian Scheipl and Sonja Greven were funded by Emmy Noether grant GR 3793/1-1 from the German Research Foundation. We thank Daniel Reich and Peter Calabresi for the DTI tractography data. We would like to thank Ciprian Crainiceanu for helpful discussions and comments on earlier versions of the paper.

References

[1] Basser, P., Mattiello, J., and LeBihan, D. (1994), MR Diffusion Tensor Spectroscopy and Imaging, *Biophysical Journal*, 66, 259-267.

[2] Basser, P., Pajevic, S., Pierpaoli, C., and Duda, J. (2000), In Vivo Fiber Tractography using DT-MRI Data, *Magnetic Resonance in Medicine*, 44, 625-632.

[3] Bunea, F., Ivanescu, A. E., and Wegkamp, M. H. (2011), Adaptive Inference for the Mean of a Gaussian Process in Functional Data, *Journal of the Royal Statistical Society, Series B*, 73(4), 531-558.

[4] Claeskens, G., Krivobokova, T., and Opsomer, J. D. (2009), Asymptotic Properties of Penalized Splines Estimators. *Biometrika*, 96(3), 529-544.

[5] Crainiceanu, C. M., and Reiss, P. (Coordinating authors), Goldsmith, J., Greven, S., Huang, L., and Scheipl, F. (Contributors) (2012), *refund: Regression with Functional Data*, R package version 0.1-5.

[6] Crainiceanu, C. M., and Ruppert, D. (2004), Likelihood Ratio Tests in Linear Mixed Models with one Variance Component, *Journal of the Royal Statistical Society, Series B*, 66(1), 165-185.

[7] Crainiceanu, C. M., Staicu, A.-M., and Di, C. (2009), Generalized Multilevel Functional Regression, *Journal of the American Statistical Association*, 104(488), 177-194.

[8] Evangelou, N., Konz, D., Esiri, M. M., Smith, S., Palace, J., and Matthews, P. M. (2000), Regional axonal loss in the corpus callosum correlates with cerebral white matter lesion volume and distribution in multiple sclerosis, *Brain*, 123(9): 1845-1849.

[9] Goldsmith, J., Bobb, J., Crainiceanu, C. M., Caffo, B. S., and Reich, D. (2011), Penalized Functional Regression, *Journal of Computational and Graphical Statistics*, 20(4), 830-851.

[10] Greven, S., Crainiceanu, C. M., Caffo, B. S., and Reich, D. (2010), Longitudinal Functional Principal Component Analysis, *Electronic Journal of Statistics*, 4, 1022-1054.

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

[12] He, G., Müller, H.-G., Wang, J.-L., and Wang, W. (2010), Functional Linear Regression via Canonical Analysis, *Bernoulli*, 16(3), 705-729.

[13] Jiang, C.-R. and Wang, J.-L. (2011), Functional Single Index Models for Longitudinal Data, *The Annals of Statistics*, 39, 362-388.

[14] Krivobokova, T. and Kauermann, G. (2007), A note on penalized spline smoothing with correlated errors, *JASA*, 102 (480), 1328-1337.

[15] Malfait, N. and Ramsay, J. O. (2003), The historical functional linear model, *The Canadian Journal of Statistics*, 31(2), 185-201.

[16] Nychka, D. (1988), Confidence Intervals for Smoothing Splines, *Journal of the American Statistical Association*, 83, 1134-1143.

[17] R Development Core Team (2012), *R: a Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, *www.R-project.org*.

[18] Ramsay, J. O., Wickham, H., Graves, S., and Hooker, G. (2011), *fda: Functional Data Analysis*, R package version 2.2.6.

[19] Ramsay, J. O., Hooker, G., and Graves, S. (2009), *Functional Data Analysis with R and Matlab*, Springer, New York.

[20] Ramsay, J. O., and Silverman, B. W. (2005), *Functional Data Analysis*, Springer-Verlag, New York.

[21] Reiss, P., and Ogden, T. (2009), Smoothing Parameter Selection for a Class of Semiparametric Linear Models, *Journal of the Royal Statistical Society, Series B*, 71(2), 505-523.

[22] Ruppert, D., Wand, M. P., and Caroll, R. J. (2003), *Semiparametric Regression*, Cambridge University Press, Cambridge, UK.

[23] Scheipl, F., Greven, S., and Küchenhoff, H. (2008), Size and Power of Tests for a Zero Random Effect Variance or Polynomial Regression in Additive and Linear Mixed Models, *Computational Statistics and Data Analysis*, 52(7), 3283-3299.

[24] Song, S. K., Sun, S. W., Ramsbottom, M. J., Chang, C., Russell, J. and Cross, A. H. (2002), Dysmyelination revealed through MRI as increased radial (but unchanged axial) diffusion of water, *Neuroimage, 17(3), 1429-1436.*

[25] Staicu, A.-M., Crainiceanu, C. M., Reich, D.S. and Ruppert, D. (2012), Modeling Functional Data with Spatially Heterogeneous Shape Characteristics, *Biometrics*, 68(2), 331-343.

[26] Tievsky, A. L., Ptak, T. and Farkas, J. (1999), Investigation of apparent diffusion coefficient and diffusion tensor anisotropy in acute and chronic multiple sclerosis lesions, *American Journal of Neuroradiology, 20(8), 1491-1499.*

[27] Wahba, G. (1983), Bayesian 'Confidence Intervals' for the Cross-Validated Smoothing Spline, *Journal of the Royal Statistical Society, Series B*, 45, 133-150.

[28] Wood, S. N. (2006), *Generalized Additive Models: An Introduction with R*, Chapman & Hall/CRC, New York.

[29] Wood, S. N. (2011), Fast Stable Restricted Maximum Likelihood and Marginal Likelihood Estimation of Semiparametric Generalized Linear Models, *Journal of the Royal Statistical Society, Series B*, 73(1), 3-36.

[30] Wood, S. N. (2012), *mgcv: GAMs with GCV/AIC/REML Smoothness Estimation and GAMMs by PQL*, R package version 1.7-13.

[31] Wu, Y., Fan, J., and Müller, H.-G. (2010), Varying-coefficient Functional Linear Regression, *Bernoulli*, 16(3), 730-758.

[32] Yao, F., Müller, H.-G., and Wang, J.-L. (2005a), Functional Data Analysis for Sparse Longitudinal Data, *Journal of the American Statistical Association*, 100(740), 577–590.

[33] Yao, F., Müller, H.-G., and Wang, J.-L. (2005b), Functional Linear Regression Analysis for Longitudinal Data, *The Annals of Statistics*, 33(6), 2873-2903.

Appendix A

For function-on-function regression scenarios with not independently identically distributed errors, pointwise approximate confidence intervals can be constructed using a resampling procedure. We provide boot-`PFFR`, a bootstrap procedure applied in the framework of PFFR. Our bootstrap approach resamples subject indices and estimates PFFR model parameters for each such resampled dataset, followed by extracting the pointwise bootstrap confidence intervals for the coefficient surface estimates from the bootstrap samples. This procedure is made available via the `bootpffr` function in the `refund` package.

Table 4: Results for function-on-function regression with densely sampled functional predictors, based on 50 simulations.

		AR(1)				heteroscedastic			
		$n = 100$		$n = 50$		$n = 100$		$n = 50$	
	Method	IAC	IAW	IAC	IAW	IAC	IAW	IAC	IAW
Scenario 4(a)									
$\beta_0(t)$									
	boot-PFFR	0.92	0.25	0.90	0.36	0.90	0.13	0.92	0.19
	modified-PFR	0.99	0.40	0.98	0.58	1.00	0.38	1.00	0.54
$\beta_1(t,s)$									
	boot-PFFR	0.99	0.59	0.99	0.86	0.94	0.13	0.94	0.17
	modified-PFR	0.77	0.29	0.77	0.34	0.80	0.28	0.82	0.33
$\beta_2(t,r)$									
	boot-PFFR	0.98	0.33	0.98	0.50	0.85	0.07	0.87	0.10
	modified-PFR	0.97	0.35	0.95	0.42	0.97	0.33	0.97	0.41

We tested `bootpffr` with $B = 200$ bootstrap samples for data generated using two different settings, AR(1) and heteroscedastic, for the error terms in the context of case 4(a) and $\sigma_X = 0$ in Section 4.1. In the AR(1) setting, residual vectors were generated from an AR(1) process with auto-correlation parameter 0.6. In the heteroscedastic setting, residuals were generated with a variance that increases linearly in t, i.e. $\epsilon_i(t) \sim N(0, 10t)$, followed by scaling to obtain unit average variance for the errors. The nominal level is set to 0.95 and results show that, for sample sizes $n = 50$ and $n = 100$, and for the AR(1) scenario, the coverage of the boot-`PFFR` confidence intervals for our model parameters approached or exceeded the nominal level; see columns IAC in Table 4. For the heteroscedastic scenario the width of the boot-`PFFR` confidence intervals were narrow and coverage was close to the nominal level. The `bootpffr` procedure is especially useful for smaller sample sizes n, as computation time increases in n (per dataset: compare 919 seconds for $n = 50$, to 1756 seconds for $n = 100$).

Appendix B

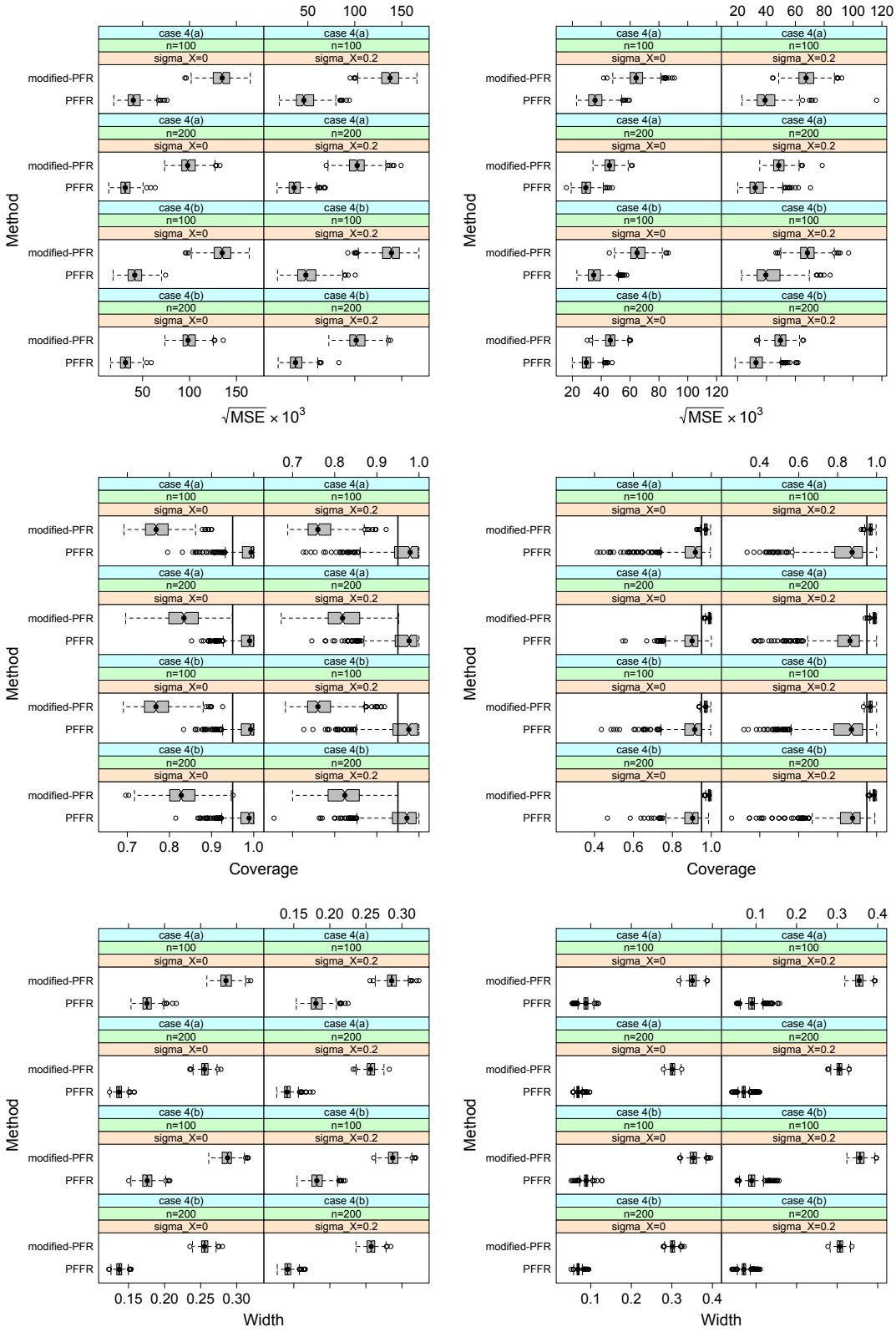

Figure 8: Displayed are box plots of the root-mean-squared errors (top), and box plots of coverage (middle) and width (bottom) of 95% pointwise approximate confidence intervals for $\beta_1(t,s)$ (left panels) and $\beta_2(t,r)$ (right panels) for all 8 scenarios in Section 4.1. Results are based on 500 simulated datasets.