Estimator Selection and Combination in Scalar-on-Function Regression

Jeff Goldsmith[1,*] and Fabian Scheipl[2]

[1]Department of Biostatistics, Columbia Mailman School of Public Health
[2]Department of Statistics, Ludwig-Maximilians-Universität Munich
[*]*jeff.goldsmith@columbia.edu*

October 6, 2013

Abstract

Scalar-on-function regression problems with continuous outcomes arise naturally in many settings, and a wealth of estimation methods now exist. Despite the clear differences in regression model assumptions, tuning parameter selection, and the incorporation of functional structure, it remains common to apply a single method to any data set of interest. In this paper we develop tools for estimator selection and combination in the context of continuous scalar-on-function regression based on minimizing the cross-validated prediction error of the final estimator. A broad collection of functional and high-dimensional regression methods is used as a library of candidate estimators. We find that the performance of any single method relative to others can vary dramatically across data sets, but that the proposed cross-validation procedure is consistently among the top performers. Four real-data analyses using publicly available benchmark datasets are presented; code implementing these analyses and facilitating the application of proposed methods on future data sets is available in a web supplement.

Key Words: Cross validation; Functional linear model; Model stacking; Super learning.

1 Introduction

The problem of predicting continuous scalar outcomes from functional predictors has received high levels of interest in recent years, driven in part by a proliferation of complex datasets and by an increase in computational power. Although there are now many approaches to this problem, including several techniques for the popular functional linear model and methods for a number of data-generating scenarios, it is rare for a practitioner to apply more than one scalar-on-function regression method to any dataset. Doing so would potentially yield improved predictions of outcomes or new insights into scientific processes; indeed, the choice of regression model and estimation technique (a process we will refer to as estimator selection) is important and can dramatically affect prediction of outcomes and interpretation of results.

In this manuscript we develop approaches to facilitate the comparison and combination of many scalar-on-function estimation methods. We first focus on estimator selection, or the choice of a single estimator from a large collection of candidates, and then on the dynamic combination of approaches to yield an optimal ensemble estimator of the association between a scalar outcome and functional predictor. Our proposed approaches are based on estimator selection through minimizing cross-validated loss (Breiman, 1996; Dudoit and van der Laan, 2005; van der Laan and Dudoit, 2003; Wolpert, 1992), referred to variously in the literature as model stacking and super learning. We adapt these strategies

to the setting in which predictors are both high dimensional and spatially structured. Publicly available software allows easy comparison and selection of methods for predicting scalar outcomes from functional predictors.

Many approaches to scalar-on-function regression for continuous outcomes are now available. In the context of the functional linear model (described below), techniques include functional principal components regression (Ramsay and Silverman, 2005) and partial least squares (Reiss and Ogden, 2007), and penalized spline methods (Cardot et al., 2003; Goldsmith et al., 2011; Marx and Eilers, 1999; James et al., 2009). Extensions of the functional linear model include the functional generalized additive model (McLean et al., pted), the functional additive model (James and Silverman, 2005), and single-index regressions (Eilers et al., 2009). A point-impact model was proposed in (Lindquist and McKeague, 2009) and a Bayesian hierarchical regression kernel method was developed in (Woodard et al., ress). In addition, high-dimensional regression and machine learning methods that are not specifically designed for structured functional data can nonetheless be applied to such datasets. Such methods include ridge regression, lasso and elastic net (Friedman et al., 2010), classification and regression trees (Breiman et al., 1984), boosting (Freund and Schapire, 1995), random forests (Breiman, 2001), and support vector machines (Suykens and Vandewalle, 1999). These methods are not necessarily designed for functional data, but could be applied to scores resulting from a truncated functional principal component analysis or from other reduced rank basis representations to give hybrid functional methods. Many of the approaches mentioned above are accompanied by software implementations.

We are motivated by a desire to optimally predict continuous scalar outcomes from

functional data, acknowledging that no one method will be universally superior, and therefore pursue estimator selection and ensembling to compare and combine competing methods. To demonstrate the practical significance of this approach, we consider four real-data examples in this manuscript. First we consider the standard Canadian weather dataset, in which daily temperature measurements are used to estimate log annual precipitation at 35 monitoring stations. Next we analyze the Tecator dataset, which consists of 215 near-infrared (NIR) absorbance spectra of meat samples used as predictors of fat content of the sample. Third we analyze a diffusion tensor imaging (DTI) dataset, where the goal is predicting a scalar measure of cognitive function from functional summaries of intracranial white matter microstructure using 334 observations. Finally we examine an additional NIR spectra dataset, which consists of 72 samples of cookie dough in which the sucrose content is of interest. These examples illustrate several practical issues related to the application of scalar-on-function regression methods, including the differential performance of individual methods across datasets, the value of applying, selecting, and ensembling multiple methods, and the computational concerns in the proposed techniques. All datasets considered are publicly available, and code implementing each analysis is available as a web supplement.

The remainder of the manuscript is organized as follows. A broad selection of approaches for functional regression is discussed in Section 2, while estimator selection and ensembling are detailed in Section 3. Real data analyses are presented in Section 4. We close with a discussion in Section 5.

2 Existing Methods for Continuous Scalar-on-Function Regression

We observe data $[Y_i, W_i(s)]$ for subjects $1 \leq i \leq I$ where Y_i is a continuous outcome and $W_i(s)$, without loss of generality assuming $s \in [0, 1]$, is the functional predictor of interest. In practice the curves $W_i(s)$ are observed on a discrete grid $\{s_{ij}\}_{j=1}^{J_i}$ that is potentially sparse and subject-specific, and often observations are subject to measurement error. Preprocessing steps such as smoothing or functional principal components analysis (FPCA) can be used to reduce the effect of measurement error and obtain curves on a dense common grid $\{s_j\}_{j=1}^{J}$; in this exposition we will assume data in this form. This section reviews existing methods for estimating the regression function $\psi_0(W(s)) = \mathrm{E}[Y \mid W(s)]$. While we attempt to be thorough, this review is not exhaustive. Our discussion focuses only on a single functional predictor although several of the approaches discussed allow multiple functional predictors or the inclusion of non-functional covariates, both of which are important in practice.

2.1 Functional Linear Model

The functional linear model (FLM) extends the standard multiple linear regression model to functional predictors. Thus we assume

$$Y_i = \int_0^1 W_i(s)\beta(s)ds + \epsilon_i \tag{1}$$

where $\epsilon_i \sim \mathrm{N}\left[0, \sigma^2\right]$ and $\beta(s)$ is the coefficient function. The FLM seeks to minimize the sum of squared errors $\left\| \boldsymbol{Y} - \int_0^1 \boldsymbol{W}(s)\beta(s)ds \right\|^2$ where $\|\boldsymbol{v}\| = \sqrt{\boldsymbol{v}^T\boldsymbol{v}}$. It is additionally assumed, either implicitly or explicitly, that the coefficient function $\beta(s)$ is smooth in some sense over its domain; such an assumption respects the local structure inherent in the predictor and avoids the problem of an ill-posed regression when $J \geq I$. The functional linear model is perhaps the most common approach for scalar-on-function regression, and many techniques have been proposed to estimate the coefficient function $\beta(s)$ based on different assumed forms of this function. The coefficient function $\beta(s)$ is an interpretable object: locations with large $|\beta(s)|$ are influential for the outcome, and the direction of the association is given by the sign of the coefficient function.

Functional principal components regression (FPCR) is based on an FPCA decomposition of the functional predictors (Ramsay and Silverman, 2005). Specifically, curves are approximated using $W_i(s) \approx \sum_{k=1}^{K_W} c_{ik}\phi_k(s)$ where $\boldsymbol{c}_i = \{c_{ik}\}_{k=1}^{K_W}$ are subject-specific principal component loadings, and $\boldsymbol{\phi}(s)$ and K_W are respectively the shared orthonormal basis functions and truncation lag estimated from the spectral decomposition of the covariance operator $\Sigma^W(s,t) = cov\left[W_i(s), W_i(t)\right]$. FPCR uses the FPC basis functions to express $\beta(s) = \boldsymbol{\phi}(s)\boldsymbol{\beta}$, so that the integral appearing in (1) becomes $\int_0^1 W_i(s)\beta(s)ds = \int_0^1 \left(\boldsymbol{\phi}(s)\boldsymbol{c}_i\right)^T \boldsymbol{\phi}(s)\boldsymbol{\beta}ds = \boldsymbol{c}_i^T\boldsymbol{\beta}$. In effect, FPCR poses a standard regression model in which the FPC loadings $\boldsymbol{c}_i$ are used as predictors and the parameters of interest are $\boldsymbol{\beta}$; the truncation lag K_W acts as an implicit tuning parameter that controls the shape and smoothness of $\beta(s)$. The effect of quadratic penalties on the regression coefficients $\boldsymbol{\beta}$ in FPCR is discussed in Randolph et al. (2012); the authors establish that the resulting estimates are based on "partially empirical" basis functions that depend on the singular value decom-

position of both the predictors and the specific penalty matrix, and conclude that when prior information is available informative penalties can improve estimation. L_1 penalties for coefficients $\boldsymbol{\beta}$ are considered in Lee and Park (2011), which allows a variable selection approach to choosing the most informative FPC basis functions for the outcome.

Broadly speaking, penalized spline approaches to the FLM expand $\beta(s)$ using a rich spline basis and impose explicit penalties to enforce smoothness in the coefficient function. These methods minimize $\left\|Y - \int_0^1 W(s)\beta(s)ds\right\|^2 + \lambda P(\beta(s))$ where λ is a tuning parameter that controls the amount of penalization and $P(\beta(s))$ is the penalty function. Both Marx and Eilers (1999) and Cardot et al. (2003) express the coefficient function in terms of a B-spline basis of size K_B and impose a difference penalty on the B-spline coefficients. Thus $\beta(s) = \boldsymbol{B}_{K_B}(s)\boldsymbol{\beta}$ where $\boldsymbol{B}_{K_B}(s)$ are the B-spline basis functions, and $P(\beta(s)) = \boldsymbol{\beta}^T \boldsymbol{P}_d^T \boldsymbol{P}_d \boldsymbol{\beta}$ where $\boldsymbol{P}_d\boldsymbol{\beta}$ gives the d-th order differences of $\boldsymbol{\beta}$; the tuning parameter λ is chosen via cross validation. For $d = 2$ this penalizes deviations of the coefficient function from linearity, and for $d = 0$ this penalizes deviations from $\beta(s) = 0$. The FPCR$_R$ method of Reiss and Ogden (2007) first projects the curves $W_i(s)$ onto a B-spline basis and then, using the smoothed predictors, carries out FPCR subject to the d-th order difference penalty. The tuning parameters K_W and λ, which control the number of FPC basis functions and penalty size respectively, are chosen using a nested m-fold cross validation (λ can also be estimated using a mixed model within folds for K_W). In Goldsmith et al. (2011), predictors are preprocessed using FPCA and the coefficient function is expressed using a flexible spline basis. Spline coefficients are treated as random effects whose distribution depends on the penalty structure; all parameters, including the tuning parameter λ, are estimated using a mixed model framework.

The penalized spline approaches discussed above focus on smoothness in the estimated coefficient function $\beta(s)$. An additional goal is the interpretability of the coefficient function, which is often achieved by inducing sparsity; that is, forcing $\beta(s) = 0$ for a large proportion of $s \in [0, 1]$. With interpretability in mind, James et al. (2009) introduces L_1 penalties on the derivatives of the coefficient function so that $P(\beta(s)) = \|\boldsymbol{P}_d\boldsymbol{\beta}\|_1$. The specific case of $d = 0$ is considered in Lee and Park (2011), who consider several penalization schemes other than the lasso and Dantzig selector (both of which appear in James et al. (2009)). A wavelet-based method for the FLM is considered in Zhao et al. (2012): similarly to FPCR, both the predictors and coefficient function are expanded using a rich orthonormal wavelet basis. Wavelet basis coefficients for $\beta(s)$ are penalized using a lasso penalty to enforce sparsity in the estimated coefficient function. Methods that promote interpretability through enforcing sparsity, either in the coefficient function or its derivatives, often use cross-validation to choose tuning parameter values.

2.1.1 Generalizations of the FLM

Several extensions of the FLM have been proposed to allow for more complex data generating mechanisms. Single index regression models for functional predictors are considered in Ait-Saïdi et al. (2008) and Eilers et al. (2009). These approaches add a non-linear index function to the traditional FLM:

$$Y_i = f\left(\int_0^1 W_i(s)\beta(s)ds\right) + \epsilon_i. \tag{2}$$

In Ait-Saïdi et al. (2008), a kernel estimator is used for $f(\cdot)$ conditionally on $\beta(s)$ and the optimal $\beta(s)$ is chosen via cross-validation. In Eilers et al. (2009), $\beta(\cdot)$ and $f(\cdot)$ are alternately held fixed while the other is optimized subject to smoothness constraints.

Modifications of projection pursuit regression (Chen, 1991) are considered for functional predictors in James and Silverman (2005), Amato et al. (2006), and Ferraty et al. (2013). As an example, James and Silverman (2005) proposes the model

$$Y_i = \beta_0 + \sum_{k=1}^{r} f_k \left(\int_0^1 W_i(s)\beta_k(s)ds \right) + \epsilon_i \tag{3}$$

for an arbitrary r. Constraints ensuring the identifiability of $\beta_k(s)$ and $f_k(\cdot)$ are needed for both (2) and (3), and constraints preventing correlation of f_k and $f_{k'}$ are needed for (3). In James and Silverman (2005), the authors expand predictors using a truncated principal component expansion and consider a penalty $P(\beta(s))$ that reduces variation that is orthogonal to the first PC basis functions, thereby penalizing variations in $\beta(s)$ that do not coincide with variations in the $W_i(s)$. Amato et al. (2006) uses a wavelet decomposition of functional predictors, and estimates the regression using minimum average variance estimation. Ferraty et al. (2013) relaxes the distributional assumptions of previous methods and introduces a criterion for choosing the number of projections. Chen et al. (2011) describe fully nonparametric extensions to estimate $\beta(s)$ and $f(\cdot)$ in both (2) and (3).

A coefficient surface is estimated in the functional generalized additive model of McLean et al. (pted), which poses the regression

$$Y_i = \int_0^1 f\left(W_i(s), s\right) ds + \epsilon_i. \tag{4}$$

Here $f(\cdot,\cdot)$ is a bivariate function that allows the effect at location s to vary by the value of $W_i(s)$. Penalized tensor product B-splines are used to estimate the surface $f(\cdot,\cdot)$. Both (2) and (4) include the FLM (1) as a special case.

A nonparametric kernel-based regression approach for functional covariates is described in (Ferraty and Vieu, 2006; Febrero-Bande and Oviedo de la Fuente, 2012):

$$Y_i = \frac{\sum_{j\neq i} K(h^{-1}d(W_j(s), W_i(s)))Y_j}{\sum_{j\neq i} K(h^{-1}d(W_j(s), W_i(s)))} + \epsilon_i, \tag{5}$$

where $K(\cdot)$ is a kernel function, h is a tuning parameter and $d(\cdot,\cdot)$ is a suitable (semi-)metric for function spaces. In the implementation we use, $K(\cdot)$ is an asymmetric normal kernel and the metric is given by $d(x(s), y(s)) = \sqrt{\int (x(s) - y(s))^2 ds}$.

Recently, Ferraty and Vieu (2009) proposed using a collection of semi-metrics d_j, $j = 1, \ldots, J$, as learners in a boosting algorithm. In this method, regressions of the form in (5) are applied sequentially, with subsequent models being fit to the residuals of previous estimates; information from each regression is included additively in predicting outcomes Y.

2.2 High-Dimensional Regression Methods

Because functional data are in practice observed discretely, methods that have been developed for high-dimensional regression can be applied to functional observations. Such methods include penalized linear models (with ridge, lasso, bridge, elastic net or other penalties), boosting (Freund and Schapire, 1995), classification and regression trees (Breiman et al., 1984), random forests (Breiman, 2001), Bayesian variable selection (Mitchell and

Beauchamp, 1988), generalized additive models and many others. In a few cases, these methods have been proposed for use with functional data. An example is the point-impact model described in Lindquist and McKeague (2009), which assumes that only a very small number of locations contain information regarding the outcome. The point-impact outcome model is

$$Y_i = \sum_{j=1}^{J} \beta_j W_i(s_j) + \epsilon_i. \tag{6}$$

where regression coefficients β_j are estimated for each observed grid point in the domain of $W_i(\cdot)$. This model is estimated using a lasso-type penalty on the discretized observations $W_i(s_j)$ and is particularly suitable for predictors that lack strong local structure.

It is important to note that methods based on treating observed data as an unstructured vector of covariates do not incorporate spatial structure in the estimates of regression coefficients, and therefore have very different interpretations from the approaches described in Section 2.1. For the point-impact model the interpretive focus is on the discrete collection of informative time points s_j with non-zero β_j rather than on a continuous coefficient function $\beta(s)$. While the use of some specific high-dimensional regression methods for functional observations has been proposed, many methods (such as random forests) have not to our knowledge been used for scalar-on-function regression.

Finally we note that the hybridization of functional and high-dimensional methods is also possible by focusing on FPC loadings as the unstructured predictor vector. This can effectively result in alternate methods for estimating the FLM, such as in the use of a lasso penalty for FPC scores described in Lee and Park (2011). Some hybrid methods have been

proposed but others are, to the best of our knowledge, novel.

3 Estimator Selection and Ensembling

While each of the methods in Section 2 focus on the problem of scalar-on-function regression, predictions made by these methods can be quite different. Fundamentally different assumptions in the FLM, for instance regarding smoothness of $\beta(s)$ or choices in how to enforce sparsity, often result in very distinct coefficient function estimates. Methods based on high-dimensional regression techniques pose a regression model that can be quite different from that given by the FLM. The relative performance of these methods in practice depends on the unknowable true data generating mechanism.

In this Section we introduce the use of estimator selection and of ensembling for scalar-on-function regression. Intuitively, we assess multiple models and estimating procedures for their predictive performance on the data set of interest. In Section 3.1 we choose the estimator with the lowest cross-validated prediction error; in Section 3.2 we combine methods into an ensemble predictor. Methods described in this section are based on approaches for estimator selection by minimizing cross-validated loss (Polley and van der Laan, 2010; Wolpert, 1992; van der Laan et al., 2007).

3.1 Estimator Selection by Minimizing Cross-Validated Loss

Given data $[Y_i, W_i(s)]$ for subjects $1 \leq i \leq I$, we propose to estimate the regression function $\psi_0^W(W(s)) = \mathrm{E}[Y \mid W(s)]$ from a large collection of candidate estimators. This collection can include any of those discussed in Section 2, but of course we are not limited

to these methods. Our goal is to minimize the expected squared error loss $\psi_0^W(W(s)) = \arg\min_\psi \mathrm{E}\left\{\left[Y - \psi^W(W(s))\right]^2\right\}$, where the argument ψ represents distinct estimators for the regression of scalar outcomes on functional predictors. Let $\boldsymbol{C} = \{\psi_k^W\}_{k=1}^K$ represent the collection of candidate estimators, and for each estimator let $\hat{\psi}_k^W$ be the estimator fitted to the observed data.

M-fold cross-validation is used to select the estimator with the minimum cross-validated squared error prediction loss. Subjects i are partitioned into M exclusive and exhaustive sets of roughly equal size. Each set and its complement in turn are the validation and training sets $V(m)$ and $T(m)$, respectively, for $m = 1, \ldots, M$. For each split, all estimators $\psi_k^W \in \boldsymbol{C}$ are fit on the training data $T(m)$ and applied to the validation data $V(m)$, giving predictions $\hat{\psi}_{k,T(m)}^W(W_i(s))$ for $i \in V(m)$. Let n_m be the size of $V(m)$. The averaged cross-validated loss $\mathrm{CV}_{avg}(k) = \frac{1}{M}\sum_{m=1}^M \left\{\frac{1}{n_m}\sum_{i\in V(m)}\left\{\left[Y_i - \hat{\psi}_{k,T(m)}^W(W_i(s))\right]^2\right\}\right\}$ is computed for each estimator $\psi_k^W \in \boldsymbol{C}$. We then select the estimator with the smallest cross-validated loss by choosing $\hat{k} = \arg\min_k \mathrm{CV}_{avg}(k)$, and define the cross-validation selected estimator $\hat{\psi}_{\hat{k}}^W$. Theoretical arguments in (van der Laan et al., 2006) show that the cross-validation approach to estimator selection will perform asymptotically as well as the oracle selector (which chooses the unknown estimator that minimizes loss under the true data-generating mechanism).

3.2 Ensemble Prediction

The efficient combination of estimation methods can achieve better performance than the selection of a single estimator. As above we are interested in minimizing the cross-validated squared error loss, but we propose to do this by combining the estimators ψ_k^W

in $\boldsymbol{C}$ based on their individual performances to create an ensemble estimator. We develop an approach based on existing ensembling techniques to unify and combine methods for scalar-on-function regression.

To begin, construct vectors $\boldsymbol{Z}_i = \left\{\hat{\psi}^W_{k,T(m(i))}(W_i(s))\right\}_{k=1}^K$ where $m(i)$ denotes the validation set such that $i \in V(m)$. Thus Z_i contains the predictions of Y_i from each candidate estimator based on training data that excludes subject i. Next, we estimate the regression $\mathrm{E}[Y \mid \boldsymbol{Z}] = \psi_0^Z(\boldsymbol{Z})$ of Y onto $\boldsymbol{Z}$ using observations $(Y_i, \boldsymbol{Z}_i)$. There are, of course, many possible procedures to estimate ψ_0^Z; a common choice is to use a linear model so that $\mathrm{E}[Y \mid \boldsymbol{Z}] = \boldsymbol{Z}\boldsymbol{\beta}$, possibly subject to sparsity or convexity constraints. Using a linear model the ensemble estimate of $\psi_0^Z(Z)$ is $\hat{\psi}_0^Z(Z) = \sum_{k=1}^K \hat{\beta}_k \hat{\psi}_k(W(s))$, which is constructed using estimators ψ_k fit to the full data. The estimator selection method in Section 3.1 is a special case of the ensembler in which $\beta_{\hat{k}} = 1$ and remaining coefficients are zero; a second special case sets $\beta_k = \frac{1}{K}$ for all k and thus assigns equal weight to all candidate estimators. Asymptotic properties of the ensemble approach are discussed in van der Laan et al. (2007).

In practice it is common to have a large and diverse collection of candidate estimators $\psi_k^W(W(s))$ in $\boldsymbol{C}$ that result in heterogeneous predictions of Y, as well as to have a diverse collection of potential ensemblers to estimate $\psi_0^Z(\boldsymbol{Z})$. This may necessitate a second level of cross-validation to choose the optimal ensembler.

4 Numerical Studies

We now implement the proposed estimator selection and combination methods on three benchmark datasets. Of the potential methods for scalar-on-function regression, we limit our library of candidate estimators to those with readily available R implementations. Thus we consider FPCR fitted using a linear model on the first few FPC loadings $\max(K_W) = 15$ (`wrap.pclm`) and FPCR using a spike-and-slab prior (`wrap.spikeslabGAM`); partial least squares (`wrap.plsreg2`); several penalized FLM implementations, including methods using PC bases (`wrap.fpcr`), smoothing splines (`wrap.flm`, `wrap.pfr`, `wrap.flirti`) and wavelet expansions (`wrap.wnet`); generalizations of the FLM (`wrap.fgam`, `wrap.sisr`); non-functional methods (`wrap.lasso`, `wrap.rf`, `wrap.gbm`); and simple in-sample and out-of-sample means (`benchmark.is`, `benchmark.oos`) used to gauge the improvement in predictions based on functional covariates. We consider several ensemblers, including the linear model (`lm`), the linear model with lasso penalty (`lasso`) and non-negative least squares (`nnls`), random forests (`rf`) and choosing the single estimator with minimum cross-validated loss (`best`). Table 1 lists each method along with a short description, the implementation source, and appropriate references.

For each data set, we partition the data into training and validation sets with 70% and 30% of the full data, respectively. We train all methods using the ten-fold cross-validation procedure described in Section 3 using the training data only, and apply the resulting prediction methods to the validation data. Because this process can be sensitive to the original partitioning into training and validation datasets, we replicate the entire procedure twenty times. We report relative root MSEs calculated using the validation set for all

methods, where the relative RMSE is standardized to the best performer within a replicate. Average computation time to conduct a full replicate using a laptop computer is reported for each data set. All datasets are publicly available, and the code implementing the analysis is published in a web supplement.

4.1 Application to Canadian Weather Data

We begin by analyzing a classic dataset in the FDA literature that contains daily temperature and rainfall observations over the course of a year. Data come from 35 geographically diverse Canadian weather monitoring stations, and it is of interest to predict the log annual precipitation from the observations of temperature. These data have been widely analyzed as a test case for scalar-on-function regression methods James et al. (2009); Lee and Park (2011); Ramsay and Silverman (2005). The data set appears in the R package `fda` (Ramsay et al., 2012), available on CRAN. Daily temperature measurements for the thirty five stations are shown in the left panel of Figure 1, sorted according to the log annual precipitation.

The results of the analysis are shown in the right panel of Figure 1. There is substantial heterogeneity in the performance of any particular method across replications of the partitioning and fitting procedure, potentially due to both the small sample size for training estimators and the small validation sample for estimating MSEs. The `wrap.flirti` approach in this example underperforms compared to other FLMs, a possible result of the assumed data generating mechanism or of the default tuning parameter choices of this implementation. Most approaches seem to improve upon the benchmark mean estimators, indicating that the functional covariates contain useful information in predicting the

Method	Description	Implementation (package:function)	References
Prediction Algorithms			
wrap.pclm	linear model on first K FPC loadings, optimal K estimated by 20-fold bootstrap	our implementation	Ramsay and Silverman (2005)
wrap.spikeslabGAM	Bayesian additive model with variable selection on first K_W FPC loadings	spikeSlabGAM: spikeSlabGAM	Scheipl (2011); Scheipl et al. (2012)
wrap.fpcr	functional principal component regression on first K_W FPC loadings	refund:fpcr	Crainiceanu et al. (2012); Reiss and Ogden (2007)
wrap.flm	REML-based functional linear model with a locally adaptive penalty	our implementation, using mgcv:gam	Wood (2011); Cardot et al. (2003)
wrap.pfr	penalized functional regression	refund:pfr	Crainiceanu et al. (2012); Goldsmith et al. (2011)
wrap.flirti	functional linear model with sparse coefficient function	code downloaded from author's website	James et al. (2009)
wrap.plsreg2	penalized partial least squares	ppls: penalized.pls.cv	Krämer et al. (2008); Krämer and Boulesteix (2011)
wrap.wnet	functional linear model in a wavelet basis with elastic net penalty	refund:wnet	Crainiceanu et al. (2012); Zhao et al. (2012)
wrap.fgam	functional additive model	refund:fgam	Crainiceanu et al. (2012); McLean et al. (pted)
wrap.sisr	REML-based single-index signal regression with locally adaptive penalty	our implementation, using mgcv:gam	Wood (2011); Eilers et al. (2009)
wrap.psr	CV-based single-index signal regression	R-script by Brian Marx & Bin Li	Eilers et al. (2009)
wrap.lasso	LASSO penalized linear model on first K_W FPC loadings	glmnet:cv.glmnet	Friedman et al. (2010)
wrap.rf	random forest on first K_W FPC loadings	randomForest: randomForest	Liaw and Wiener (2002)
wrap.gbm	Friedman's gradient boosting machine applied to first K_W FPC loadings	gbm:gbm	Ridgeway (2013)
wrap.fregrenp	non-parametric kernel-based functional regression	fda.usc:fregre.np	Febrero-Bande and Oviedo de la Fuente (2012); Ferraty and Vieu (2006)
Ensembling Algorithms			
lm	simple linear model	base:lm	
lasso	see wrap.lasso		
nnls	ensembling individual predictions via non-negative least squares	nnls:nnls	Mullen and van Stokkum (2012)
rf	see wrap.rf		
svd	LASSO model on the principal components of the matrix of single algorithm predictions	base:svd, glmnet:cv.glmnet	Friedman et al. (2010)
best	use predictions of best-performing algorithm on training data		

Table 1: Algorithms and ensembling methods used in the applications.

outcome. Each replication took on average 186 seconds on a laptop computer.

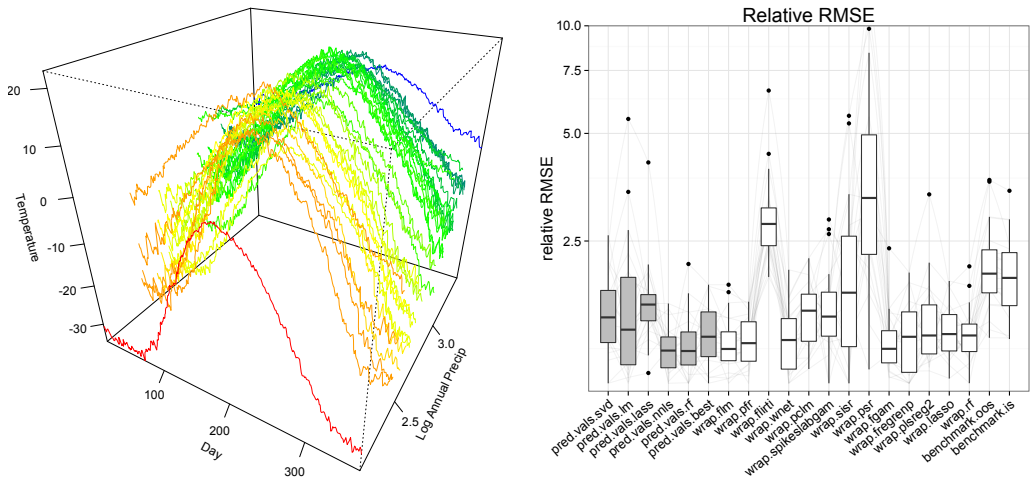

Figure 1: Canadian weather application data and results. Left panel shows daily temperature measurements for the recording locations sorted and color-coded by log annual precipitation. Right panel shows boxplots of relative root mean square errors on the test set for the 20 replications. Relative root mean square error is defined as RMSE divided by RMSE of the best (single or ensemble) algorithm for that replication. Boxplots for ensemble methods in grey, for single algorithms in white. Note that some outliers for `wrap.psr` have been cut off.

Several points follow from the analysis of the Canadian weather data. When comparing approaches for their predictive performance, it is important to repeat the analysis for multiple partitions into training and validation sets to understand the variation in relative performance. In this example many methods, including the ensemble prediction algorithms, have similar predictive performances. Finally we see that very similar models (here using the FLM) can have very different performances due to differing forms for penalization and default choices for tuning parameters.

4.2 Application to Tecator Data

Our second dataset focuses on the prediction of the fat content of meat samples based on near-infrared (NIR) absorbance spectra. These data were originally used by Borggaard & Thodberg (Borggaard and Thodberg, 1992) and have since appeared often in the functional data literature (Eilers et al., 2009; Yao and Müller, 2010; Zhao et al., 2012). There are 215 observations consisting of the fat content outcome and a 100-channel spectrum of absorbances measured using a Tecator Infrared spectrometer. These data can be found in the R package `fda.usc` (Febrero-Bande and Oviedo de la Fuente, 2012), available on CRAN. The NIR spectra are are shown in the left panel of Figure 2.

The right panel of Figure 2 displays the results for the Tecator data analysis. Here `wrap.sisr` clearly outperforms all other individual estimation approaches. Other approaches based on the functional linear model or FPCA work much less well than `wrap.sisr`, but seem to improve upon the non-functional methods. One notable exception is `wrap.fgam`, which for this dataset has very unstable performance across replications. The "pick the best" selector correctly identifies the single index regression in each replication, while the relative performance of the single index regression and other ensemble predictors depends on the replication. Computation time for each replication was on average 447 seconds.

The Tecator data analysis emphasizes that the cross-validated selector will generally identify a clearly superior estimator if one exists. Moreover ensemble prediction methods can often outperform a single method, although the relative performance depends to some extent on the training and validation data. On the other hand, some ensemblers

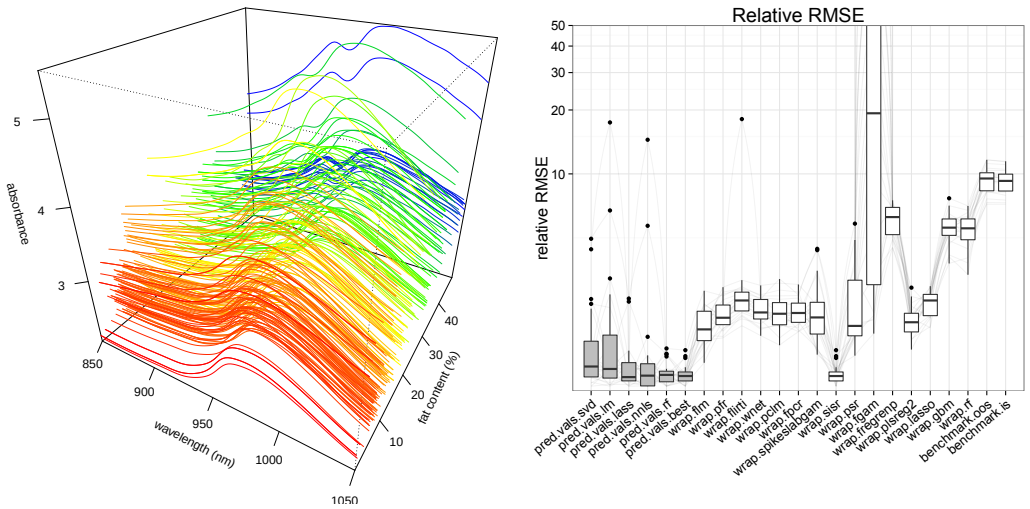

Figure 2: Tecator application data and results. Left panel shows NIR absorbance spectra for the meat samples sorted and color-coded by their fat content. Right panel shows boxplots of relative root mean square errors on the test set for the 20 replications. Relative root mean square error is defined as RMSE divided by RMSE of the best (single or ensemble) algorithm for that replication. Boxplots for ensemble methods in grey, for single algorithms in white. Note that some outliers for the FGAM fits are cut off.

can be sensitive to the effects of a single poor prediction algorithm, as demonstrated by the large outliers for several of these approaches. This may be alleviated by removing the problematic single estimator (in this case `wrap.fgam`), but the sensitivity of ensemble predictors is important to keep in mind.

4.3 Application to DTI Data

Multiple sclerosis (MS) is an immune-mediated disease that is associated with the incidence of demyelinating lesions. Damage to the myelin sheath of white matter fibers can disrupt the transmission of electrical signals in individual fibers and bundles of fibers, or tracts, and thereby result in patient disability. Diffusion tensor imaging (DTI) is an MRI-

based technique that traces the diffusion of water to assess white matter microstructure. In this study we focus on the corpus callosum, a major collection of white matter tissue, and use the fractional anisotropy measure of white matter microstructure parameterized by distance along the tract as the functional predictor of interest; tracts are registered across subjects using anatomical landmarks. We are interested in the association between white matter in the corpus callosum and a scalar measure of cognitive performance using 334 observations. These data have been previously analyzed using several methods Goldsmith et al. (2011); McLean et al. (pted); Randolph et al. (2012). The left panel of Figure 3 shows the tract summaries sorted according to the associated cognitive outcome. This data set is available in the R package `refund` (Crainiceanu et al., 2012), available on CRAN.

Results for the scalar-on-function regression analysis using the DTI data set are shown in the right panel of Figure 2. Many methods have similar performances, although random forests seem to outperform many functional data methods. The improvement compared to the benchmark mean approaches is relatively modest. Several possibilities for the failure to generate substantially improved outcome prediction are possible: the cognitive function score is a proxy for a complex process and may be only loosely associated with patient function, or the tract profiles may oversimplify a complex three-dimensional structure and lose important anatomical information. Average computation time for each replication was 650 seconds.

This data set illustrates three important points related to scalar-on-function regression: i. there is little or no penalty on performance from using an estimator selection or ensembling procedure, demonstrated by the reasonable performance of the selection and en-

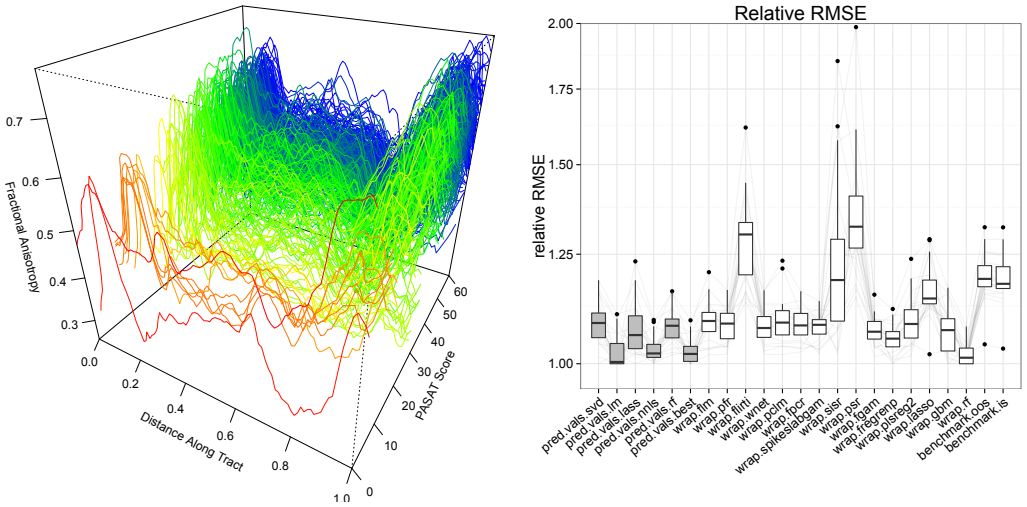

Figure 3: DTI application data and results. Left panel shows fractional anisotropy (FA) curves along the right corticospinal tract ordered and color-coded according to the cognitive performance outcome (PASAT score). Right panel shows boxplots of relative root mean square errors on the test set for the 20 replications. Relative root mean square error is defined as RMSE divided by RMSE of the best (single or ensemble) algorithm for that replication. Boxplots for ensemble methods in grey, for single algorithms in white. Note that some outliers for the SISR fits are cut off.

sembling by non-negative least squares or random forests; $ii.$ of the single approaches the best performer is random forests, which is a useful reminder that for scalar-on-function regression non-functional approaches can perform well; and $iii.$ the poor performance of `wrap.sisr` (which was clearly the best method for use with the Tecator data) and the reasonable performance of `wrap.fgam` (which performed poorly on the Tecator data) for this data set reinforce that the use of a single method, even if it has performed well on other data sets, can lead to very poor predictions.

4.4 Application to Cookie Data

Our final example arises from a near infrared spectroscopy study of cookie dough samples with the goal of assessing sucrose content. There are 72 spectra measured from 1100 to 2498 nanometers (nm) in 2 nm increments, giving functional predictors densely observed on grids of length 700. This data set is contained in the R package `ppls` (Krämer and Boulesteix, 2011), available on CRAN. Cookie dough spectra are shown in the left panel of Figure 4, and the relative root MSEs across replications of the training/validation split are shown in the right panel. For these data most individual methods provide substantial improvements over benchmark means. However there is a lack of uniformity across FLMs, in terms of both average performance and variability. As in Tecator data in Section 4.2 (the other NIR example) the better performers among individual methods tend to be functional in nature. Average computation time across replications was 605 seconds.

Among the examples we consider, this dataset most clearly demonstrates the potential usefulness of ensemble predictors – the random forest ensemble predictor tends to moderately outperform competing approaches. Again we have an example of the instability of single predictors across datasets. Here `wrap.pfr` is outperformed by all other FLMs, while `wrap.flirti` is comparable to remaining methods; this reverses the relative performance of these methods for the Canadian weather and DTI datasets. Finally we note that while ensemble predictors tend to be among the better methods across datasets, their relative ordering can vary. For this dataset, the random forest ensembler outperforms the "pick the best" approach, while the opposite was true for the DTI dataset.

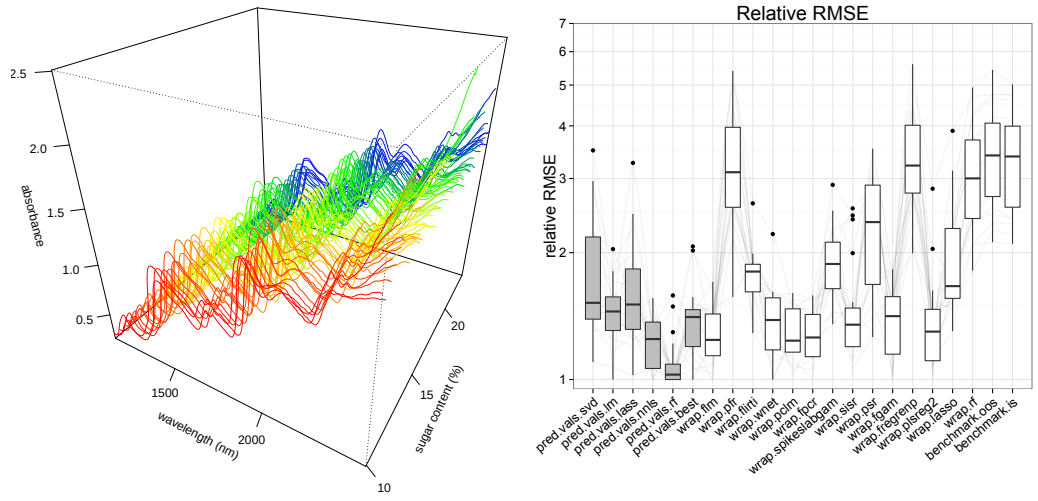

Figure 4: Cookie dough application data and results. Left panel shows NIR absorbance spectra for the cookie dough samples sorted and color-coded by their sucrose content. Right panel shows boxplots of relative root mean square errors on the test set for the 20 replications. Relative root mean square error is defined as RMSE divided by RMSE of the best (single or ensemble) algorithm for that replication. Boxplots for ensemble methods in grey, for single algorithms in white.

5 Concluding remarks

Recent years have seen intense development of new methods for scalar-on-function regression, resulting in a plethora of estimation procedures representing a wide range of model assumptions and estimation choices. Because of these disparities, the estimation procedures can have dramatically different performances when applied to the same data set. Despite the availability of multiple approaches, it remains common to use only a single method for any real-data analysis. In this paper we have extended estimator selection and ensembling to scalar-on-function regression; our publicly available code provides tools to facilitate the implementation of the proposed methods to new data sets.

Our applications to three benchmark data sets indicate some useful observations and guidelines:

- Estimator selection and ensembling often outperforms any single estimation method for a given dataset, and are more consistently among the best performers across data sets than single estimators.
- Single estimation methods can have very different performances across data sets, and the relative ordering of methods is not fixed.
- Comparison of multiple methods through cross validation increases computational burden, but is reasonable for many situations.
- Cross validation reduces the reliance on user choices for estimator and model selection.
- Replication of the training/validation step illustrates variability of results depending on this split.
- Models based on very different assumptions regarding the data generating mechanism can produce similar predictions.

In this manuscript we have focused on the specific problem of continuous scalar-on-function regression with a single functional predictor, due to the commonality of this context and the plethora of estimation techniques available. In practice, however, a wide array of problems arise in functional regression. For continuous outcomes, it is commonly of interest to include both a functional predictor and standard scalar covariates, often referred to as partial functional regression (Aneiros-Perez and Vieu, 2006, 2008), or to include multiple functional predictors. Binary outcomes are regularly of interest, in addition to other non-continuous scalar responses (Goldsmith et al., 2011). Functional-response models have been considered in many applications; see, for example, Reiss and Huang (2010), Ferraty et al. (2011, 2012) or Scheipl et al. (2013) for recent developments.

In all of these cases, we consider it important to compare multiple estimation techniques and believe the proposed framework suitable for this end.

A drawback of the proposed is the loss of interpretation from ensemble predictors, particularly given that interpretability is traditionally considered an advantage of functional regression methods. On the other hand, our final point above indicates that very different model assumptions can provide similar fits; in such cases reliance on a particular model to provide scientific insights may be misleading and the interpretations of several similar models should be considered. In the case that a single method uniformly outperforms its competitors, it may useful to focus on that approach and use the interpretations that are derived from its application. A second disadvantage of the proposed method is the difficulty of obtaining inference for predicted values compared to individual model-based approaches. Inference based on targeted maximum likelihood may be possible for ensemble predictors and is the topic of future work. Finally we acknowledge the computational effort required to compare many methods through cross-validation, and anticipate that this will be exacerbated as larger data sets, such as two- and three-dimensional imaging studies, become more common and methods for scalar-on-image regression more numerous.

6 Supplementary Materials

All supplementary materials are contained in a zipped Web Appendix available on the first author's website. Supplements consist of code used to access and analyze data described in Section 4.

7 Acknowledgments

Fabian Scheipl's research was financially supported by the German Research Foundation (DFG) through the Emmy Noether Program (grant GR 3739/1-1 for Dr. Sonja Greven).

References

L. Breiman, Heuristics of instability and stabilization, Annals of Statistics 24 (1996) 2350–2383.

S. Dudoit, M. J. van der Laan, Asymptotics of cross-validated risk estimation in estimator selection and performance assessment, Statistical Methodology 2 (2005) 131–154.

M. J. van der Laan, S. Dudoit, Unified cross-validation methodology for selection among estimators and a general cross-validated adaptive epsilon-net estimator: Finite sample oracle inequalities and examples, Technical Report 13 (2003).

D. H. Wolpert, Stacked generalization, Neural Networks 5 (1992) 241–259.

J. O. Ramsay, B. W. Silverman, Functional Data Analysis, New York: Springer, 2005.

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

H. Cardot, F. Ferraty, P. Sarda, Spline estimators for the functional linear model, Statistica Sinica 13 (2003) 571–591.

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

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

G. M. James, J. Wang, J. Zhu, Functional linear regression that's interpretable, Annals of Statistics 37 (2009) 2083–2108.

M. W. McLean, G. Hooker, A.-M. Staicu, F. Scheipl, D. Ruppert, Functional generalized additive models, Journal of Computational and Graphical Statistics (Accepted).

G. M. James, B. Silverman, Functional adaptive model estimation, Journal of the American Statistical Association 100 (2005) 565–576.

P. H. C. Eilers, B. Li, B. D. Marx, Multivariate calibration with single-index signal regression, Chemometrics and Intelligent Laboratory Systems 96 (2009) 196–202.

M. Lindquist, I. McKeague, Logistic regression with Brownian-like predictors, Journal of the American Statistical Association 104 (2009) 1575–1585.

D. B. Woodard, C. M. Crainiceanu, D. Ruppert, Hierarchical adaptive regression kernels for regression with functional predictors, Journal of Computational and Graphical Statistics (In Press).

J. Friedman, T. Hastie, R. Tibshirani, Regularization paths for generalized linear models via coordinate descent, Journal of Statistical Software 33 (2010) 1–22.

L. Breiman, J. H. Freidman, R. A. Olshen, C. J. Stone, Classification and Regression Trees, Belmont, CA: Wadsworth, 1984.

Y. Freund, R. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, in: Computational Learning Theory, Springer Berlin / Heidelberg, 1995, pp. 23–37.

L. Breiman, Random forests, Machine Learning 24 (2001) 123–140.

J. A. K. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural Processing Letters 9 (1999) 293–300.

T. W. Randolph, J. Harezlak, Z. Feng, Structured penalties for functional linear models-partially empirical eigenvectors for regression, Electronic Journal of Statistics 6 (2012) 323–353.

E. R. Lee, B. U. Park, Sparse estimation in functional linear regression, Journal of Multivariate Analysis 105 (2011) 1–17.

Y. Zhao, R. T. Ogden, P. T. Reiss, Wavelet-based LASSO in functional linear regression, Journal of Computational and Graphical Statistics 21 (2012) 600–617.

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

H. Chen, Estimation of a projection-pursuit type regression model, Annals of Statistics 19 (1991) 142–157.

U. Amato, A. Antoniadis, I. De Feis, Dimension reduction in functional regression with applications, Computational Statistics and Data Analysis 50 (2006) 2422–2446.

F. Ferraty, A. Goia, E. Salinelli, P. Vieu, Functional projection pursuit regression, Test 22 (2013) 293–320.

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

F. Ferraty, P. Vieu, Nonparametric Functional Data Analysis: Theory and Practice, New York: Springer, 2006.

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

F. Ferraty, P. Vieu, Additive prediction and boosting for functional data, Computational Statistics and Data Analysis 53 (2009) 1400–1413.

T. Mitchell, J. Beauchamp, Bayesian variable selection in linear regression, Journal of the American Statistical Association 83 (1988) 1023–1032.

E. C. Polley, M. J. van der Laan, Super learner in prediction, COBRA (2010).

M. J. van der Laan, E. C. Polley, A. E. Hubbard, Super learner, Statistical applications in genetics and molecular biology 6 (2007).

M. J. van der Laan, S. Dudoit, van der Vaart A W, The cross-validated adaptive epsilon-net estimator, Statistics and Decisions 24 (2006) 373–395.

F. Scheipl, `spikeSlabGAM`: Bayesian variable selection, model choice and regularization for generalized additive mixed models in r, Journal of Statistical Software 43 (2011) 3283–3299.

F. Scheipl, L. Fahrmeir, T. Kneib, Spike-and-slab priors for function selection in structured additive regression models, Journal of the American Statistical Association 107 (2012) 1518–1532.

C. Crainiceanu, P. Reiss, J. Goldsmith, L. Huang, L. Huo, F. Scheipl, refund: Regression with Functional Data, 2012. URL: http://CRAN.R-project.org/package=refund, r package version 0.1-6.

S. N. Wood, Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models, Journal of the Royal Statistical Society: Series B 73 (2011) 3–36.

N. Krämer, A.-L. Boulesteix, G. Tutz, Penalized partial least squares with applications to b-spline transformations and functional data, Chemometrics and Intelligent Laboratory Systems 94 (2008) 60–69.

N. Krämer, A.-L. Boulesteix, ppls: Penalized Partial Least Squares, 2011. URL: http://CRAN.R-project.org/package=ppls, r package version 1.05.

A. Liaw, M. Wiener, Classification and Regression by randomForest, R News 2 (2002) 18–22.

G. Ridgeway, gbm: Generalized Boosted Regression Models, 2013. URL: http://CRAN.R-project.org/package=gbm, r package version 2.0-8.

K. M. Mullen, I. H. M. van Stokkum, nnls: The Lawson-Hanson algorithm for non-negative least squares (NNLS), 2012. URL: http://CRAN.R-project.org/package=nnls, r package version 1.4.

J. O. Ramsay, H. Wickham, S. Graves, G. Hooker, fda: Functional Data Analysis, 2012. URL: http://CRAN.R-project.org/package=fda, r package version 2.3.2.

C. Borggaard, H. H. Thodberg, Optimal minimal neural interpretation of spectra, Analytical Chemistry 64 (1992) 545–551.

F. Yao, H.-G. Müller, Functional quadratic regression, Biometrika 97 (2010) 49–64.

G. Aneiros-Perez, P. Vieu, Semi-functional partial linear regression, Statistics and Probability Letters 76 (2006) 1102–1110.

G. Aneiros-Perez, P. Vieu, Nonparametric time series prediction: A semi-functional partial linear modeling, Journal of Multivariate Analysis 99 (2008) 834–857.

J. Goldsmith, C. M. Crainiceanu, B. Caffo, D. Reich, Penalized functional regression analysis of white-matter tract profiles in multiple sclerosis, NeuroImage 57 (2011) 431–439.

P. T. Reiss, L. Huang, Fast function-on-scalar regression with penalized basis expansions, International Journal of Biostatistics 6 (2010) Article 28.

F. Ferraty, A. Laksaci, A. Tadj, P. Vieu, Kernel regression with functional response, Electronic Journal of Statistics 5 (2011) 159–171.

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

F. Scheipl, A.-M. Staicu, S. Greven, Functional additive mixed models, Under Review (2013).