Introducing Functional Data Analysis to Neuroimaging, and Vice Versa

Philip T. Reiss, New York University

Available at: http://works.bepress.com/phil_reiss/25/
Introducing functional data analysis to neuroimaging, and vice versa

Philip T. Reiss
New York University and Nathan Kline Institute
phil.reiss@nyumc.org http://works.bepress.com/phil_reiss

Cornell University Statistics Seminar
November 16, 2011

Research supported in part by grant DMS-0907017,
US National Science Foundation
Acknowledgments

Thanks to

- co-authors Lei Huang (Johns Hopkins), Yin-Hsiu Chen (NYU/Columbia), Thad Tarpey (Wright State University), Maarten Mennes (NYU);
- Eva Petkova and the Biostatistics Division at the NYU Child Study Center;
- Mike Milham, Clare Kelly, Xi-Nian Zuo and Juan Zhou (Institute for Pediatric Neuroscience, NYU Child Study Center);
- Babak Ardekani and the Image Analysis Group at the Nathan Kline Institute;
- Stan Colcombe, Nathan Kline Institute;
- Ciprian Crainiceanu, Johns Hopkins University;
- SAMSI’s 2010–11 Program on Analysis of Object Data
Motivating application
Scientific question: development of homotopic functional connectivity in the brain

- 193 subjects, age 7–50, were scanned with resting-state functional magnetic resonance imaging (fMRI).
- For each subject, we obtain a time series of the BOLD signal (an index of blood oxygenation, and hence of brain activity) at each of a grid of voxels (volume units).
- Neuroscientists have recently used “resting state functional connectivity”—essentially, correlation between time series for different voxels—to study brain networks.
- This particular application focuses on “homotopic” connectivity—correlation between 71287 pairs of voxels in corresponding locations in the left and right hemispheres—and how it develops with age.
Homotopic connectivity as a polynomial function of age (Zuo et al., 2010)
Plan for the talk

I. **Massively parallel nonparametrics**: computational methods to do nonparametric analyses at each voxel

II. Apply **functional data analysis** (FDA) ideas for improved results
I. Massively parallel nonparametrics
Basic objective

Improved characterization of voxel-by-voxel developmental trajectories by penalized $B$-spline smoothing …

… rather than polynomial models.
More formally: say we have performed brain scans for $n$ individuals, yielding some quantity at each of $V \approx 10^5$ voxels. For $i = 1, \ldots, n$, we have collected

- age $x_i$;
- image-derived values $y_{i1}, \ldots, y_{iV}$ (for our data, homotopic correlations derived from functional MRI);
- other demographic / clinical covariates (but we’ll omit these for simplicity).

Standard approach for developmental neuroimaging data: for $v = 1, \ldots, V$, use stepwise testing to choose among

\begin{align*}
\mathcal{M}_{0v} & : \quad E(y_{iv}) = \beta_0 \\
\mathcal{M}_{1v} & : \quad E(y_{iv}) = \beta_0 + \beta_1 x_i \\
\mathcal{M}_{2v} & : \quad E(y_{iv}) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 \\
\mathcal{M}_{3v} & : \quad E(y_{iv}) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3
\end{align*}
Proposed alternative: Massively parallel nonparametrics

1. Massively parallel testing: Test a parametric null model such as $M_0$ (constant) or $M_1$ (linear) vs. a general smooth alternative $M_{*v} : E(y_{iv}) = g_v(x_i)$.

2. Massively parallel smoothing: Estimate $g_v(·)$ in model $M_{*v}$.

Main challenge: Can these nonparametric analyses be done $V \approx 10^5$ times in a computationally feasible manner?
The solution has two key ingredients:

1. the mixed-model formulation of penalized splines
2. some efficient matrix algebra
Penalized splines

- Given data \((x_1, y_1), \ldots, (x_n, y_n)\) (univariate responses), we fit the model
  \[ y_i = g(x_i) + \varepsilon_i, \quad E(\varepsilon_i) = 0 \]
as follows.

1. Assume that \(g(x) = \theta^T b(x)\) where \(b(\cdot) = [b_1(\cdot), \ldots, b_K(\cdot)]^T\); here the \(b_j\) are basis functions such as B-splines.
2. Estimate \(\theta\) by penalized least squares:

   \[ \hat{\theta} = \arg \min_{\theta \in \mathbb{R}^K} \left( \underbrace{\|y - B\theta\|^2}_{\text{sum of squared errors}} + \lambda \underbrace{\theta^T P \theta}_{\text{roughness functional}} \right), \]

where

- \(y = (y_1, \ldots, y_n)^T\),
- \(B = \begin{bmatrix} b_1(x_1) & \cdots & b_K(x_1) \\ \vdots & \ddots & \vdots \\ b_1(x_n) & \cdots & b_K(x_n) \end{bmatrix} \);
- \(\lambda\) is a tuning parameter, and \(P\) is a \(K \times K\) matrix.
Roughness penalty $\lambda \theta^T P \theta$ in criterion $\|y - B\theta\|^2 + \lambda \theta^T P \theta$ prevents overfitting by shrinking $\hat{g}(\cdot) = \hat{\theta}^T b(\cdot)$ toward $\{\theta^T b(\cdot) : \theta^T P \theta = 0\} = \{\theta^T b(\cdot) : \theta \in \text{null} P\}$; smoothing parameter $\lambda \geq 0$ controls the degree of shrinkage. Examples:

- $P = (\int b_i'' b_j'')_{1 \leq i,j \leq K} \rightarrow \theta^T P \theta = \int g''(x)^2 dx \rightarrow$ shrink toward linear $g$

- $P = (\int b_i' b_j')_{1 \leq i,j \leq K} \rightarrow \theta^T P \theta = \int g'(x)^2 dx \rightarrow$ shrink toward constant $g$
Mixed-model formulation of penalized splines

- With a change of basis

\[ B \rightarrow (X | Z) \text{ where } \text{col}(X) = \text{null}(P) \]

(e.g., Wand and Ormerod, 2008), criterion \( \| y - B\theta \|^2 + \lambda \theta^T P\theta \)
becomes

\[ \| y - X\beta - Zu \|^2 + \lambda u^T u, \]

which is \( \propto - \log[f(y|u)f(u)] \) for the linear mixed model

\[ y = X\beta + Zu + \varepsilon \]

with \( u \sim N(0, (\sigma^2/\lambda)I), \varepsilon \sim N(0, \sigma^2 I) \) (cf. Speed, 1991).

- Many authors (e.g., Ruppert, Wand and Carroll, 2003; Wood, 2011) in effect reduce nonparametric inference to mixed model inference, with smoothing parameter \( \lambda \) recast as a variance parameter.
REML-based nonparametric inference

More specifically, the correspondence between the penalized spline problem

\[ \hat{\theta} = \arg \min_{\theta \in \mathcal{R}^K} \left( \| y - B\theta \|^2 + \lambda \theta^T P\theta \right) \]  

(1)

and a linear mixed model

\[ y = X\beta + Z u + \varepsilon \text{ with } u \sim N(0, (\sigma^2 / \lambda) I), \varepsilon \sim N(0, \sigma^2 I), \]  

(2)

motivates basing inference on the restricted log likelihood (REML criterion) \( \ell_R(\beta, \sigma^2, \lambda | y) \) of model (2), or its profile version

\[
\ell_R(\lambda | y) = -\frac{1}{2} (n - p) \log [y^T \{ V_{\lambda}^{-1} - V_{\lambda}^{-1} X (X^T V_{\lambda}^{-1} X)^{-1} X^T V_{\lambda}^{-1} \} y ]
\]

\[- \frac{1}{2} \log |V_{\lambda}| - \frac{1}{2} \log |X^T V_{\lambda}^{-1} X| \quad \text{(with } V_{\lambda} \equiv I_n + \lambda^{-1} ZZ^T) : \]

1. **Testing** \( H_0 : \theta \in \text{null} P \) (e.g., a linear fit vs. a nonparametric alternative) reduces to testing \( \lambda = \infty \) (zero random effect variance); can use the Crainiceanu and Ruppert (2004) restricted likelihood ratio test statistic \( RLRT(y) = \sup_{\lambda \geq 0} 2\ell_R(\lambda | y) - 2\ell_R(\infty | y) \).

2. **Do optimal smoothing** by taking \( \lambda = \arg \max_{\lambda \geq 0} \ell_R(\lambda | y) \) in (1).
The massively parallel setting

For \( v = 1, \ldots, V \), the penalized spline problem

\[
\hat{\theta}_v = \arg \min_{\theta \in \mathbb{R}^K} \left( \|y_v - B\theta\|^2 + \lambda \theta^T P\theta \right)
\]

gives rise to REML criterion

\[
\ell_R(\lambda | y_v) = -\frac{1}{2} (n - p) \log [ y_v^T \{ V^{-1}_\lambda - V^{-1}_\lambda X (X^T V^{-1}_\lambda X)^{-1} X^T V^{-1}_\lambda \} y_v ]
\]

\[
- \frac{1}{2} \log |V_\lambda| - \frac{1}{2} \log |X^T V^{-1}_\lambda X|,
\]

and hence to

1. a problem of testing parametric null \( H_{0v} : \theta \in \text{null} P \) via
   \( \text{RLRT}(y_v) = \sup_{\lambda \geq 0} 2 \ell_R(\lambda | y_v) - 2 \ell_R(\infty | y_v) \).

2. an optimal smoothing problem of finding \( \lambda_v = \arg \max_{\lambda \geq 0} \ell_R(\lambda | y_v) \).

Question: How to solve each of these problems efficiently \( V \approx 10^5 \) of times??
1. MP testing: naïve solution

To approximate

\[
\text{RLRT}(y_v) = \sup_{\lambda \geq 0} 2\ell_R(\lambda|y_v) - 2\ell_R(\infty|y_v) \quad \text{for } v = 1, \ldots, V,
\]

1. choose a grid \( \lambda^{(1)} < \ldots < \lambda^{(G)} \);
2. find the maximum of each column of the \( G \times V \) matrix

\[
\begin{bmatrix}
\ell_R(\lambda^{(1)}|y_1) & \ell_R(\lambda^{(1)}|y_2) & \ldots & \ell_R(\lambda^{(1)}|y_V) \\
\vdots & \vdots & \ddots & \vdots \\
\ell_R(\lambda^{(G)}|y_1) & \ell_R(\lambda^{(G)}|y_2) & \ldots & \ell_R(\lambda^{(G)}|y_V)
\end{bmatrix}.
\]
1. MP testing: fast solution

Main task in computing \( \{ \ell_R(\lambda^g|y_v) : 1 \leq g \leq G, 1 \leq v \leq V \} \)
is to obtain 
\[
y_v^T M_{\lambda^g} y_v \quad \text{for each } g, v,
\]
where 
\[
M_{\lambda} = V_{\lambda}^{-1} X (X^T V_{\lambda}^{-1} X)^{-1} X^T V_{\lambda}^{-1} - V_{\lambda}^{-1};
\]
i.e., compute the \( G \times V \) matrix 
\[
\begin{bmatrix}
y_1^T M_{\lambda^{(1)}} y_1 & \cdots & y_V^T M_{\lambda^{(1)}} y_V \\
\vdots & \ddots & \vdots \\
y_1^T M_{\lambda^{(G)}} y_1 & \cdots & y_V^T M_{\lambda^{(G)}} y_V
\end{bmatrix}
= 
\begin{bmatrix}
1^T [ Y \odot (M_{\lambda^{(1)}} Y) ] \\
\vdots \\
1^T [ Y \odot (M_{\lambda^{(G)}} Y) ]
\end{bmatrix},
\]
where \( Y = (y_1 \ldots y_V) \).
2. MP smoothing

- At \( v \)th point (voxel), the smooth function \( g_v(x) = \theta_v^T b(x) \) is estimated via
  \[
  \hat{\theta}_v = \arg\min_{\theta \in \mathbb{R}^K} \left( \| y_v - B \theta \|^2 + \lambda_v \theta^T P \theta \right),
  \]
  where \( B = [b_j(x_i)]_{1 \leq i \leq n, 1 \leq j \leq K} \).

- Two tasks must be performed for \( v = 1, \ldots, V \) with huge \( V \):
  1. Find the optimal \( \lambda_v \), i.e., \( \lambda_v = \arg\max_{\lambda \geq 0} \ell_R(\lambda | y_v) \).
     - Use same trick as for RLRT to quickly maximize \( \ell_R(\lambda | y_v) \) for \( v = 1, \ldots, V \).
  2. Given the optimal \( \lambda_v \), compute
     \[
     \hat{\theta}_v = \left( B^T B + \lambda_v P \right)^{-1} B^T y_v.
     \]

- Demmler-Reinsch algorithm yields a single matrix expression of the form
  \[
  (\hat{\theta}_1 \ldots \hat{\theta}_V) = G[W_{\lambda_1, \ldots, \lambda_V} \odot (G^T B^T Y)]
  \]
  (no need to invert a \( K \times K \) matrix for each of \( V \) voxels).
Approximate timing comparisons (in minutes) for the homotopic connectivity data

<table>
<thead>
<tr>
<th></th>
<th>RLRT</th>
<th>Smoothing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naive</td>
<td>116</td>
<td>58</td>
</tr>
<tr>
<td>Proposed</td>
<td>1.9</td>
<td>2.1</td>
</tr>
</tbody>
</table>
Voxelwise RLRT results

- $H_0 = \bigcap_{v=1}^{V} H_{0v}$ for $H_{0v}$: homotopic connectivity for $v$th voxel does not change with age, i.e. $g_v(\cdot)$ constant
- FDR, thresholded at 0.1 (dark red $\rightarrow$ nonsignificant)
Smoothing: comparison with mgcv for 200 voxels

(a) Log smoothing parameter

(b) Effective degrees of freedom

(c) Fitted curves for voxel B

(d) REML maxima for voxel B
II. Two roles for functional data analysis
Massively parallel smoothing can be enhanced by ideas from functional data analysis, in two completely different ways:

1. **Functional data clustering**: Treat the MP curve estimates as functional “data,” and cluster them to get prototypical developmental trajectories.

2. **Function-on-scalar regression**: View entire images as *functional responses* being regressed on a scalar predictor (age).
1. Functional data clustering

- Massively parallel smoothing yields an estimate $\hat{g}_v(\cdot)$ of the mean developmental trajectory for voxel $v$ ($v = 1, \ldots, V$).
- How to display a succinct summary of all these function estimates?
- Treat them as functional “data,” and apply FDA clustering methodology (e.g., Tarpey and Kinateder, 2003).
Functional data clustering algorithm

1. Interested in shape, rather than height, of trajectories
   cluster estimated first derivative of trajectories . . .

   \[ \hat{g}_1(\cdot) = \hat{\theta}_1^T b(\cdot) \quad \hat{g}_1'(\cdot) = \hat{\theta}_1^T b'(\cdot) \]
   \[ \vdots \quad \longrightarrow \quad \vdots \]
   \[ \hat{g}_V(\cdot) = \hat{\theta}_V^T b(\cdot) \quad \hat{g}_V'(\cdot) = \hat{\theta}_V^T b'(\cdot) \]

2. Truncated Karhunen-Loève expansion
   \[ \hat{g}_V'(\cdot) \approx \sum_{m=1}^{M} c_{vm}\phi_m(\cdot) \longrightarrow \text{reduce vth function to} \]
   functional principal component scores \( c_{v1}, \ldots, c_{vM} \) (e.g., Silverman, 1996; Ramsay and Silverman, 2005).

3. Apply \( k \)-means clustering to the \( c_{vm} \)'s.
Pointwise estimated developmental trajectories: six-cluster solution for voxels with RLRT FDR < 0.1
2. Function-on-scalar regression

- The massively parallel approach (called “mass-univariate” analysis in neuroimaging) fails to exploit the fact that neighboring voxels $v_1, v_2$ should usually imply similar functions ($g_{v_1} \approx g_{v_2}$).

- We may be able to overcome this limitation by viewing the $i$th image $y_i = (y_{i1}, \ldots, y_{iV})$ as a discretized version of functional response

$$y_i : S \rightarrow \mathcal{R}$$

for a compact set $S \subset \mathcal{R}^3$.

- For simplicity, we’ll consider a different data set with $S \subset \mathcal{R}$ (functional responses are functions of one variable, rather than three)…
Functional responses: fractional anisotropy (FA) profiles, regressed on age

FA along a 107-voxel cross-section in the corpus callosum, in 146 individuals age 7–48.
Analyzing an infant data set, Zhu et al. (2011) view FA profiles as functional responses $y_i(\cdot)$, and regress them on scalar predictors $x_i$ such as age, via a varying-coefficient linear model similar to the Ramsay-Silverman (2005) model $y_i(s) = \mathbf{x}_i^T \beta(s) + \varepsilon_i(s)$.

linear wrt age
But for the much wider age range (7–48) of our sample, FA for some voxels depends **nonlinearly** on age.

Rainbow plots (color-coded by age)
of raw and fitted FA profiles:
Evidence of nonlinearity:
I. Pointwise RLRT
Evidence of nonlinearity:
II. Pointwise smoothing

Voxel 42 (Sensorimotor)

Voxel 97 (Frontal)
Given scalar predictor / functional-response data

\[(x_1, y_1(\cdot)), \ldots, (x_n, y_n(\cdot)),\]

we can consider a general model

\[E[y(s)|x] = h(x, s)\]

for some smooth function \(h\).

The two approaches we’ve seen are (in a sense) opposite approaches to estimating \(h\):

- Varying-coefficient model assumes
  \[h(x, s) = \beta_0(s) + x\beta_1(s) \text{ (linear wrt } x)\]
  \[\longrightarrow \text{ estimate } [\beta_0(s), \beta_1(s)].\]
- Massively parallel smoothing estimates \(h(\cdot, s_v)\), a smooth function of \(x\), for \(v = 1, \ldots, V\).
• Neither of these two approaches to estimating $E[y(s)|x] = h(x, s)$ attains ideal smoothness with respect to both $x$ and $s$: intuitively,
  • varying-coefficient model: $df_v = 2$ for $v = 1, \ldots, V$;
  • mass-univariate model: $(df_1, \ldots, df_V)$ effectively unrestricted.
• To make this more rigorous, and consider alternative estimators, we need a notion of **pointwise effective degrees of freedom** $(df_1, \ldots, df_V)$ for regression with functional responses observed on a grid of $V$ points.
Defining pointwise df, I

• Recall that for univariate responses \( y = (y_1, \ldots, y_n)^T \) and fitted values

\[
\hat{y} = H y,
\]

(e.g., \( H = B(B^T B + \lambda P)^{-1} B^T \) for penalized splines), we define

\[
\text{df} = \text{tr}(H) = \sum_{i=1}^{n} \frac{\partial \hat{y}_i}{\partial y_i}.
\]

• For multivariate or discretized functional responses: letting \( y = \text{vec}(Y_{(n \times V)}) \), (3) holds with an \( nV \times nV \) block matrix \( H \):

\[
\begin{pmatrix}
\hat{y}_1 \\
\vdots \\
\hat{y}_V
\end{pmatrix} =
\begin{pmatrix}
H_{11} & \cdots & H_{1V} \\
\vdots & \ddots & \vdots \\
H_{V1} & \cdots & H_{VV}
\end{pmatrix}
\begin{pmatrix}
y_1 \\
\vdots \\
y_V
\end{pmatrix}.
\]
Defining pointwise df, II

For massively parallel estimation, $H$ is block-diagonal—

$$
\begin{pmatrix}
\hat{y}_1 \\
\vdots \\
\hat{y}_v
\end{pmatrix}
= 
\begin{pmatrix}
H_{11} & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & H_{vv}
\end{pmatrix}
\begin{pmatrix}
y_1 \\
\vdots \\
y_v
\end{pmatrix},
$$

i.e.

$$
\hat{y}_v = H_{vv} y_v \text{ for each } v
$$

—so the natural definition of pointwise df is

$$
df_v = \text{tr}(H_{vv}).
$$
Defining pointwise df, III

But for the varying-coefficient model fit (or other estimators that share information across neighboring points),

$$H = \begin{pmatrix} H_{11} & \cdots & H_{1V} \\ \vdots & \ddots & \vdots \\ H_{V1} & \cdots & H_{VV} \end{pmatrix}$$

is not block-diagonal:
In other words, $\hat{y}_{iv}$ may depend on the whole function $y_i(\cdot)$, not just on $y_{iv}$.

This suggests generalizing the definition

$$df_v = \text{tr}(H_{vv}) = \sum_{i=1}^{n} \text{leverage} \left( \frac{\partial \hat{y}_{iv}}{\partial y_{iv}} \right)$$

to

$$df^*_v = \sum_{k=1}^{V} \text{tr}(H_{vk}) = \sum_{i=1}^{n} \sum_{k=1}^{V} \frac{\partial \hat{y}_{iv}}{\partial y_{ik}}.$$
Pointwise df of the varying-coefficient model

- Ramsay and Silverman (2005) formulate function-on-scalar regression (the varying-coefficient model) via the matrix equation

\[
\begin{bmatrix} Y \\ n \times V \end{bmatrix} = \begin{bmatrix} X \\ n \times q \end{bmatrix} \begin{bmatrix} \Theta \\ q \times K_s \end{bmatrix} + \begin{bmatrix} B_s^T \\ K_s \times V \end{bmatrix} \begin{bmatrix} \Theta \\ q \times K_s \end{bmatrix} + E,
\]

and derive a penalized least squares fit \( \hat{y} = Hy \) with

\[
H = M(M^T M + P_s \otimes \Lambda)^{-1} M^T,
\]

where \( M = B_s \otimes X \) and \( \Lambda = \text{Diag}\{\lambda_1, \ldots, \lambda_q\} \) (cf. Reiss et al., 2010, 2011).

- Pointwise df is as we would expect for a linear model:

**Theorem 1**

*Let \( \text{rank}(X) = q \) and assume*

1. \( B_s 1_{K_s} = 1_V \),
2. \( P_s 1_{K_s} = 0_{K_s} \).

*Then \( H \) as given above yields \( df_v^* = q \) for \( v = 1, \ldots, V \).*

*[So with a single predictor, \( df_v^* = 2 \) for each \( v \).]*
Pointwise df for regressing FA profiles on age

Massively parallel smoothing (○)
vs. varying-coefficient model (―)
Three estimators

Let’s now consider three ways to estimate $h$ in the model

$$y_i(s) = h(x_i, s) + \varepsilon_i(s), \quad i = 1, \ldots, n, \ s \in S,$$

and use the set of pointwise df values

$$
\begin{pmatrix}
\text{df}^*_1 \\
\vdots \\
\text{df}^*_V
\end{pmatrix}
= 
\begin{pmatrix}
\sum_{v=1}^V \text{tr}(H_{1v}) \\
\vdots \\
\sum_{v=1}^V \text{tr}(H_{vv})
\end{pmatrix}
$$

to help assess their suitability.
Approach 0: Tensor product smooth

- Restrict the smooth bivariate function \( E[y(s)|x] = h(x, s) \) to the span of a tensor product basis:

\[
E[y(s)|x] = h(x, s) = \sum_{i=1}^{K_x} \sum_{j=1}^{K_s} \theta_{ij} b_x.i(x) b_s.j(s) = b_x(x)^T \Theta b_s(s).
\]

- I.e., given the \( x \)-direction basis functions \( b_{x.1}, \ldots, b_{x.K_x} \) and the function-direction basis functions \( b_{s.1}, \ldots, b_{s.K_s} \), we assume there exists \( \Theta = (\theta_{ij})_{1 \leq i \leq K_s, 1 \leq j \leq K_x} \) such that

\[
h(x, s) = \sum_{i=1}^{K_x} \sum_{j=1}^{K_s} \theta_{ij} b_x.i(x) b_s.j(s) = b_x(x)^T \Theta b_s(s).
\]

- Standard penalized least squares estimate yields \( \hat{y} = H y \) with

\[
H = M \left[ M^T M + \lambda_x (I_{K_s} \otimes P_x) + \lambda_s (P_s \otimes I_{K_x}) \right]^{-1} M^T,
\]

where \( M = B_s \otimes B_x \) and \( \lambda_x, \lambda_s \) are \( x \)-direction and function-direction smoothing parameters.
Pointwise df curves for different values of log $\lambda_x$
Approach 1: Dependence on $x$ via functional principal components

- Chiou, Müller and Wang (2003) propose the model

$$y_i(s) = \mu(s) + \sum_{j=1}^{J} c_{ij} \psi_j(s) + \varepsilon_i(s),$$

where $\psi_1, \ldots, \psi_J$ are the leading eigenfunctions of the covariance kernel, and $y_i$ depends on $x_i$ only through the PC scores:

$$E(c_{ij}) = g_j(x_i)$$

for some smooth $g_j$.

- The number of PCs $J$ is chosen by cross-validation.
Pointwise df for FA data: mass-univariate vs. FPC-based
Approach 2 (our proposal): Two-stage smoothing

1. (Massively parallel $x$-direction smoothing): Obtain

\[ \tilde{Y} \equiv (\tilde{y}_1 \ldots \tilde{y}_V) = (A_{\lambda_1}^x y_1 \ldots A_{\lambda_V}^x y_V), \]

where $A_{\lambda_v}^x = B_x (B_x^T B_x + \lambda_v P_x)^{-1} B_x^T$, with REML-optimal choice of $\lambda_v$ for $v = 1, \ldots, V$.

2. (Function-direction smoothing)

\[ \hat{Y} = \tilde{Y} A_{\lambda_s}^s \]

\[ \begin{pmatrix}
(A_{\lambda_s}^s \tilde{y}_1.)^T \\
\vdots \\
(A_{\lambda_s}^s \tilde{y}_n.)^T
\end{pmatrix} \]

where $A_{\lambda_s}^s = B_s (B_s^T B_s + \lambda_s P_s)^{-1} B_s^T$, with $\lambda_s$ chosen by cross-validation.

Theorem 2

Let the pointwise df after stages 1 and 2 be $d = (df_1, \ldots, df_V)^T$ and $d^* = (df_1^*, \ldots, df_V^*)^T$, respectively. Then $d^* = A_{\lambda_s}^s d$.

I.e., the effect of the function-direction smoother $A_{\lambda_s}^s$ on pointwise df

\[ \ldots \text{is simply to apply the same smoother to the pointwise df vector.} \]
Pointwise df for FA data:
mass-univariate vs. two-stage
Fitted functions for approaches 1 and 2

FPC

Two-stage

Voxel

FA

0 20 40 60 80 100

0.5 0.6 0.7 0.8 0.9

Voxel

FA

0 20 40 60 80 100

0.5 0.6 0.7 0.8 0.9
Pointwise fits at voxel 42 (sensorimotor)

FPC

Two–stage

Residuals

Residuals
For applications such as ours, in which the shapes of the functions $\{h(\cdot, s) : s \in S\}$ are of primary interest, only the two-stage method smooths appropriately in both directions:
Simulation study

Repeatedly generated data sets

\[ x_i, \quad y_i(\cdot) = h(x_i, \cdot) + \eta_i(\cdot) + \xi_i(\cdot) + \varepsilon_i(\cdot) \quad (i = 1, \ldots, 100) \]

with functions observed at \( s_0 = 0, \ s_1 = \frac{1}{200}, \ldots, s_{200} = 1 \), such that

“true”* pointwise df of \( h(\cdot, s) \approx \begin{cases} 3, & s = 0.5; \\ 2, & \text{otherwise.} \end{cases} \)

* (a non sequitur)
Simulation results

Two-stage had lower estimation error than FPC in 159 of 200 replications.

Pointwise df:
Summary / Discussion

- We have introduced fast algorithms for performing RLRT and smoothing in a massively parallel fashion.
- A new notion of pointwise df motivates a two-stage approach, with MP smoothing followed by smoothing along the function:

$$\underbrace{\mathbf{Y}}_{n \times V} \rightarrow \tilde{\mathbf{Y}} \rightarrow \tilde{\mathbf{Y}} \mathbf{A}^s_{\lambda s}.$$  

  - optimally smooth each column
  - smooth rows

- More specifically, we have shown that stage 2, applying the smoother $\mathbf{A}^s_{\lambda s}$ to each row of $\tilde{\mathbf{Y}}$, corresponds to applying the same smoother to the pointwise df $(df_1, \ldots, df_V)^T$.

- In brain imaging applications:
  - each row of $\tilde{\mathbf{Y}}$ is a 3D image
  - $(df_1, \ldots, df_V)^T$ is a 3D “pointwise df image”
  - may wish to “work backwards”: find a good smoother for $(df_1, \ldots, df_V)^T$, then apply it to each row of $\tilde{\mathbf{Y}}$.

- Future work: extend massively parallel nonparametrics to more complex designs (e.g., additive models, mixed-effect models) and to generalized linear models.
Thank you!
Massively parallel nonparametrics can be adapted to voxelwise estimation of the apparent diffusion coefficient for mapping white matter tracts—which can be formulated as penalized smoothing on the unit sphere (high $\lambda \rightarrow$ near-spherical glyph). Estimates for some example voxels when $\lambda$ is...

Visualized with R package dti (Polzehl and Tabelow, 2009)
References I


