Generalized Additive Models for Regression With Functional Data

University of Auckland

Mathew McLean

Postdoctoral Research Fellow
University of Technology Sydney
School of Mathematical and Physical Sciences
mathew.mclean@uts.edu.au

August 7, 2017

Outline

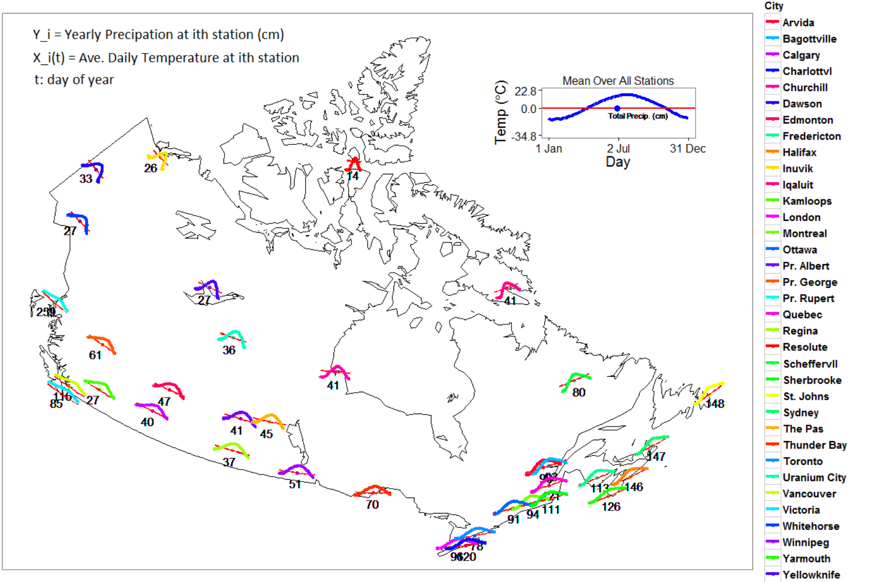
Functional Data

- Each data point is sample path of $\mathcal{L}^2$ stochastic process
  $\{X(t) : t \in \mathcal{T}\}$
- Each data point/trajectory/curve is assumed smooth
- First part of talk:
  X observed on dense grid of points and presmoothed

Functional Data

- Each data point is sample path of $\mathcal{L}^2$ stochastic process
  $\{X(t) : t \in \mathcal{T}\}$
- Each data point/trajectory/curve is assumed smooth
- First part of talk:
  X observed on dense grid of points and presmoothed
- Goal: From N samples predict Y using smooth function $X(t)$
- $\mathcal{T}$ closed interval. $\mathcal{T} = [0, 1]$ w.l.o.g. Often, t is time
- R.v. Y is continuous and normally distributed

Canadian Weather Data



Functional Linear Model (FLM)

- Bad ideas: $Y_i = \beta_0 +$

The most commonly used functional regression model:

FLM

$$E(Y_i|X_i) = \beta_0 + \int_{\mathcal{T}} \beta(t) X_i(t)\, dt \qquad i = 1, \ldots, N$$

- $\beta(\cdot)$ is unknown smooth coefficient function
- $\mathrm{Var}(Y_i|X_i) = \sigma^2$

- Effect of X on Y is linear for each t (Easy to interpret)

Functional Linear Model (FLM)

- Bad ideas: $Y_i = \beta_0 +$

The most commonly used functional regression model:

FLM

$$E(Y_i|X_i) = \beta_0 + \int_{\mathcal{T}} \beta(t) X_i(t)\, dt \qquad i = 1, \ldots, N$$

- $\beta(\cdot)$ is unknown smooth coefficient function
- $\mathrm{Var}(Y_i|X_i) = \sigma^2$

- Effect of X on Y is linear for each t (Easy to interpret)
- Two separate smooths: 1) $X(t)$ (ignored), 2) $\beta(t)$

Functional Linear Model (FLM)

- Bad ideas: $Y_i = \beta_0 +$

The most commonly used functional regression model:

FLM

$$E(Y_i|X_i) = \beta_0 + \int_{\mathcal{T}} \beta(t) X_i(t)\, dt \qquad i = 1, \dots, N$$

- $\beta(\cdot)$ is unknown smooth coefficient function
- $\mathrm{Var}(Y_i|X_i) = \sigma^2$

- Effect of X on Y is linear for each t (Easy to interpret)
- Two separate smooths: 1) $X(t)$ (ignored), 2) $\beta(t)$
- Coefficient function commonly estimated in one of two ways
  1) Using B-splines and roughness penalty
  2) Using functional principal components analysis (fPCA)

Is the FLM "good enough"?

- FLM is easy to understand, easy to fit, well-understood
- Is it flexible/general enough?

Is the FLM "good enough"?

- FLM is easy to understand, easy to fit, well-understood
- Is it flexible/general enough?

Previous attempts at extensions:

1) FDA extension of Nadaraya-Watson (1964) estimator:

$$\widehat{r}(X) = \frac{\sum_{i=1}^{N} Y_i \, K\left\{\lambda^{-1} d(X, X_i)\right\}}{\sum_{i=1}^{N} K\left\{\lambda^{-1} d(X, X_i)\right\}}, \quad \text{Ferraty and Vieu (2006)}$$

- K is an asymmetrical kernel with bandwidth λ
- d is a semimetric

Is the FLM "good enough"?

- FLM is easy to understand, easy to fit, well-understood
- Is it flexible/general enough?

Previous attempts at extensions:

1) FDA extension of Nadaraya-Watson (1964) estimator:

$$\widehat{r}(X) = \frac{\sum_{i=1}^{N} Y_i \, K\left\{\lambda^{-1} d(X, X_i)\right\}}{\sum_{i=1}^{N} K\left\{\lambda^{-1} d(X, X_i)\right\}}, \quad \text{Ferraty and Vieu (2006)}$$

   - K is an asymmetrical kernel with bandwidth λ
   - d is a semimetric

- "Black box" - hard to interpret how $X_i(t)$ affects Y_i

Is the FLM "good enough"?

- FLM is easy to understand, easy to fit, well-understood
- Is it flexible/general enough?

Previous attempts at extensions:

1) FDA extension of Nadaraya-Watson (1964) estimator:
2) Additive model in some projection of the data:

$$Y_i = \beta_0 + \sum_{j=1}^{p} f_j(\xi_{ij}) + \epsilon_i$$

- $\xi_{ij} = \int_{\mathcal{T}} \beta_j(t) X_i(t)\, dt$ (James & Silverman, 2005)
- $\xi_{ij} = j$th eigenvalue of $\text{cov}\{X(s), X(t)\}$ (Yao & Müller, 2008)

Is the FLM "good enough"?

- FLM is easy to understand, easy to fit, well-understood
- Is it flexible/general enough?

Previous attempts at extensions:

1) FDA extension of Nadaraya-Watson (1964) estimator:
2) Additive model in some projection of the data:

$$Y_i = \beta_0 + \sum_{j=1}^{p} f_j(\xi_{ij}) + \epsilon_i$$

  - $\xi_{ij} = \int_{\mathcal{T}} \beta_j(t) X_i(t)\, dt$ (James & Silverman, 2005)
  - $\xi_{ij} = j$th eigenvalue of $\mathrm{cov}\{X(s), X(t)\}$ (Yao & Müller, 2008)

- We'd like a model that incorporates $X(t)$ directly

An Additive Model With Functional Predictor - FGAM

The model we propose is

FGAM

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

unknown bivariate function $F : \mathcal{X} \times \mathcal{T} \to \mathbb{R}$

An Additive Model With Functional Predictor - FGAM

The model we propose is

FGAM

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

unknown bivariate function $F : \mathcal{X} \times \mathcal{T} \to \mathbb{R}$

- Need to impose smoothness of $F(\cdot, \cdot)$ in x and t
  - Two parameters, λ_x and λ_t control function complexity

An Additive Model With Functional Predictor - FGAM

The model we propose is

FGAM

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

unknown bivariate function $F : \mathcal{X} \times \mathcal{T} \to \mathbb{R}$

- Need to impose smoothness of $F(\cdot,\cdot)$ in x and t
  - Two parameters, λ_x and λ_t control function complexity
- If $F(x,t) = \beta(t)x$, we get the FLM
- Interpretability - Functional predictor directly incorporated

An Additive Model With Functional Predictor - FGAM

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

- Define

$$x_{ij} \equiv X_i(t_j) \quad f_j(\cdot) \equiv F(\cdot, t_j)J^{-1}$$

An Additive Model With Functional Predictor - FGAM

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

- Define

$$x_{ij} \equiv X_i(t_j) \quad f_j(\cdot) \equiv F(\cdot, t_j)J^{-1}$$

- Consider the additive model

$$E(Y_i|X_{i1}, \ldots, X_{iJ}) = \theta_0 + \sum_{j=1}^{J} f_j\{x_{ij}\} = \theta_0 + \sum_{j=1}^{J} F\{x_{ij}, t_j\}J^{-1}$$

An Additive Model With Functional Predictor - FGAM

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

- Define

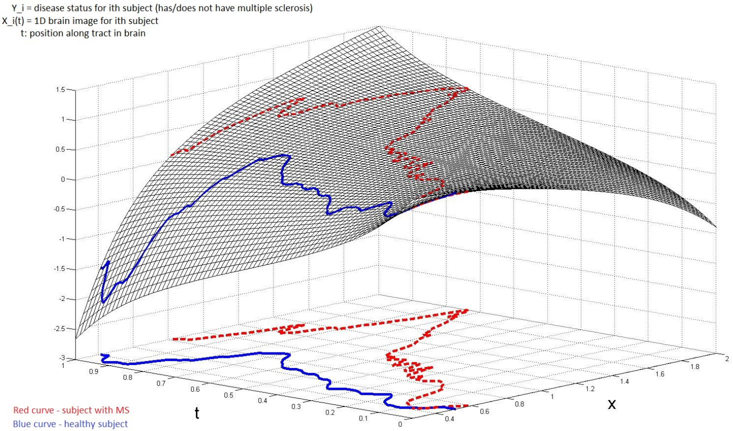
$$x_{ij} \equiv X_i(t_j) \quad f_j(\cdot) \equiv F(\cdot, t_j)J^{-1}$$

- Consider the additive model

$$E(Y_i|X_{i1}, \ldots, X_{iJ}) = \theta_0 + \sum_{j=1}^{J} f_j\{x_{ij}\} = \theta_0 + \sum_{j=1}^{J} F\{x_{ij}, t_j\}J^{-1}$$

- Obtain FGAM in limit as $J \to \infty$

Example Estimated Surface



Estimated surface $\hat{F}(x, t)$ and two predictor curves.

Model for $F(x, t)$

Simple way to represent bivariate surface $F(x, t)$:

Take products of univariate spline bases

Model for $F(x,t)$

Simple way to represent bivariate surface $F(x,t)$:

Take products of univariate spline bases

We use bivariate tensor product B-splines for $F(x,t)$

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

- $\{B_j^X(x) : j = 1, \ldots, K_x\}$ and $\{B_k^T(x) : k = 1, \ldots, K_t\}$ are low-rank, univariate B-spline bases
- Equally spaced knots, must specify degree of the spline and number of basis functions

Putting It Together

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

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

- Define $Z_{jk}(i) = \int_{\mathcal{T}} B_j^X\{X_i(t)\} B_k^T(t)\, dt$ and

  $\mathbb{Z}$, the $N \times (1 + K_x K_t)$ matrix of $Z_{jk}(i)$'s with first column $\mathbf{1}$

Putting It Together

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

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

- Define $Z_{jk}(i) = \int_{\mathcal{T}} B_j^X\{X_i(t)\} B_k^T(t)\, dt$ and

  $\mathbb{Z}$, the $N \times (1 + K_x K_t)$ matrix of $Z_{jk}(i)$'s with first column $\mathbf{1}$
- The model becomes

$$E(Y_i|X_i) = \theta_0 + \sum_{j=1}^{K_x} \sum_{k=1}^{K_t} \theta_{jk} Z_{jk}(i) = \mathbb{Z}\boldsymbol{\theta}$$

Putting It Together

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

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

- Define $Z_{jk}(i) = \int_{\mathcal{T}} B_j^X\{X_i(t)\} B_k^T(t)\, dt$ and

  $\mathbb{Z}$, the $N \times (1 + K_x K_t)$ matrix of $Z_{jk}(i)$'s with first column $\mathbf{1}$
- The model becomes

$$E(Y_i|X_i) = \theta_0 + \sum_{j=1}^{K_x} \sum_{k=1}^{K_t} \theta_{jk} Z_{jk}(i) = \mathbb{Z}\boldsymbol{\theta}$$

- Must approx. Z_{jk}'s: Choose grid $\mathbf{t}$ and quadrature weights $\mathbf{L}$

  $Z_{jk}(i) \approx \mathbf{L}^T \mathbb{B}_{\boldsymbol{\xi}_i} \equiv \mathbf{b}_{\boldsymbol{\xi}_i}^T$, $\mathbb{B}_{\boldsymbol{\xi}_i}$ has columns $B_j^X\{\hat{X}_i(\mathbf{t})\} B_k^T(\mathbf{t})$

Smoothing Via Mixed Models (Ruppertanism)

Key idea: We can reparametrize FGAM as mixed model

Smoothing Via Mixed Models (Ruppertanism)

Key idea: We can reparametrize FGAM as mixed model

- Semiparametric model - Explicitly separate $F(x, t)$ into
  1) unpenalized (parametric) fixed effect part
  2) penalized (nonparametric) random effect part

Smoothing Via Mixed Models (Ruppertanism)

Key idea: We can reparametrize FGAM as mixed model

- Semiparametric model - Explicitly separate $F(x,t)$ into
  1) unpenalized (parametric) fixed effect part
  2) penalized (nonparametric) random effect part
- Shrinkage/smoothing via variance components
- Allows use of mixed model machinery to estimate λ's

Smoothing Via Mixed Models (Ruppertanism)

Key idea: We can reparametrize FGAM as mixed model

- Semiparametric model - Explicitly separate $F(x, t)$ into
  1) unpenalized (parametric) fixed effect part
  2) penalized (nonparametric) random effect part
- Shrinkage/smoothing via variance components
- Allows use of mixed model machinery to estimate λ's
- Each part of talk uses different mixed model representation

Mixed Model Representation

Consider one scalar covariate additive model

$$\mathbf{Y} = f(\mathbf{x}) + \boldsymbol{\epsilon} = \mathbb{B}\boldsymbol{\theta} + \epsilon; \quad \boldsymbol{\epsilon} \sim N(0, \sigma_e^2 \mathbb{I}_N);$$

- $\mathbf{Y}$, $\mathbf{x}$ - N-vectors of observed data
- $\mathbb{B}$: $N \times K$ matrix of B-splines evaluated at $\mathbf{x}$

Mixed Model Representation

Consider one scalar covariate additive model

$$\mathbf{Y} = f(\mathbf{x}) + \boldsymbol{\epsilon} = \mathbb{B}\boldsymbol{\theta} + \epsilon; \quad \boldsymbol{\epsilon} \sim N(0, \sigma_e^2 \mathbb{I}_N);$$

- $\mathbf{Y}$, $\mathbf{x}$ - N-vectors of observed data
- $\mathbb{B}$: $N \times K$ matrix of B-splines evaluated at $\mathbf{x}$
- Parameter estimates given by

$$\underset{\boldsymbol{\theta},\lambda,\sigma^2}{\arg\min}\ (\mathbf{Y} - \mathbb{B}\boldsymbol{\theta})^T(\mathbf{Y} - \mathbb{B}\boldsymbol{\theta}) + \lambda\boldsymbol{\theta}^T\mathbb{P}\boldsymbol{\theta}$$

- $\mathbb{P}$ - Penalty matrix; $\boldsymbol{\theta}^T\mathbb{P}\boldsymbol{\theta}$ represents penalty $\int\{f''(x)\}^2\,dx$

$$\mathbb{P} = \mathbb{U}\mathbb{D}\mathbb{U}^T, \quad \mathbb{U}^T\mathbb{U} = \mathbb{I}, \quad \mathbb{D} = \mathrm{diag}(d_1, \cdots, d_{K-2}, 0, 0).$$

$$\mathbb{U} = [\mathbb{U}_n : \mathbb{U}_z] \text{ and } \mathbb{D}_+ = \mathrm{diag}(d_1, \cdots, d_{K-2})$$

Mixed Model Representation

Use eigendecomposition of $\mathbb{P}$ and reparametrize

$$\begin{aligned}\mathbf{Y} = f(\mathbf{x}) + \boldsymbol{\epsilon} = \mathbb{B}\boldsymbol{\theta} + \boldsymbol{\epsilon} &= \mathbb{B}[\mathbb{U}_n : \mathbb{U}_z][\mathbb{U}_n : \mathbb{U}_z]^T\boldsymbol{\theta} + \epsilon \\ &= [\mathbb{Z} : \mathbb{X}]\begin{pmatrix}\boldsymbol{\delta} \\ \boldsymbol{\beta}\end{pmatrix} + \boldsymbol{\epsilon} = \mathbb{X}\boldsymbol{\beta} + \mathbb{Z}\boldsymbol{\delta} + \epsilon; \\ \boldsymbol{\epsilon} \sim N(0, \sigma_e^2\mathbb{I}_N); \quad \boldsymbol{\delta} &\sim N(0, \sigma_u^2\mathbb{D}_+^{-1}); \quad \lambda = \sigma_u^2/\sigma_e^2\end{aligned}$$

where $\mathbb{X} = [\mathbf{1} : \mathbf{x}]$ if using 2nd order penalty, $\int[f''(x)]^2\,dx$

Mixed Model Representation

Use eigendecomposition of $\mathbb{P}$ and reparametrize

$$\begin{aligned} \mathbf{Y} &= f(\mathbf{x}) + \boldsymbol{\epsilon} = \mathbb{B}\boldsymbol{\theta} + \boldsymbol{\epsilon} = \mathbb{B}[\mathbb{U}_n : \mathbb{U}_z][\mathbb{U}_n : \mathbb{U}_z]^T\boldsymbol{\theta} + \epsilon \\ &= [\mathbb{Z} : \mathbb{X}]\begin{pmatrix}\boldsymbol{\delta} \\ \boldsymbol{\beta}\end{pmatrix} + \boldsymbol{\epsilon} = \mathbb{X}\boldsymbol{\beta} + \mathbb{Z}\boldsymbol{\delta} + \epsilon; \\ \boldsymbol{\epsilon} &\sim N(0, \sigma_e^2\mathbb{I}_N); \quad \boldsymbol{\delta} \sim N(0, \sigma_u^2\mathbb{D}_+^{-1}); \quad \lambda = \sigma_u^2/\sigma_e^2 \end{aligned}$$

where $\mathbb{X} = [\mathbf{1} : \mathbf{x}]$ if using 2nd order penalty, $\int[f''(x)]^2\,dx$

- New penalty: $\boldsymbol{\theta}^T\mathbb{P}\boldsymbol{\theta} = \boldsymbol{\delta}^T\mathbb{D}_+\boldsymbol{\delta}$. $\boldsymbol{\beta}$ unpenalized

Mixed Model Representation

Use eigendecomposition of $\mathbb{P}$ and reparametrize

$$\mathbf{Y} = f(\mathbf{x}) + \boldsymbol{\epsilon} = \mathbb{B}\boldsymbol{\theta} + \boldsymbol{\epsilon} = \mathbb{B}[\mathbb{U}_n : \mathbb{U}_z][\mathbb{U}_n : \mathbb{U}_z]^T\boldsymbol{\theta} + \epsilon$$
$$= [\mathbb{Z} : \mathbb{X}]\begin{pmatrix}\boldsymbol{\delta}\\ \boldsymbol{\beta}\end{pmatrix} + \boldsymbol{\epsilon} = \mathbb{X}\boldsymbol{\beta} + \mathbb{Z}\boldsymbol{\delta} + \epsilon;$$
$$\boldsymbol{\epsilon} \sim N(0, \sigma_e^2\mathbb{I}_N); \quad \boldsymbol{\delta} \sim N(0, \sigma_u^2\mathbb{D}_+^{-1}); \quad \lambda = \sigma_u^2/\sigma_e^2$$

where $\mathbb{X} = [\mathbf{1} : \mathbf{x}]$ if using 2nd order penalty, $\int [f''(x)]^2\, dx$

- New penalty: $\boldsymbol{\theta}^T\mathbb{P}\boldsymbol{\theta} = \boldsymbol{\delta}^T\mathbb{D}_+\boldsymbol{\delta}$. $\boldsymbol{\beta}$ unpenalized

Notice: $\sigma_u^2 = 0 \Rightarrow \boldsymbol{\delta} = \mathbf{0} \Rightarrow \mathbf{Y} = [\mathbf{1} : \mathbf{x}]\binom{\beta_0}{\beta_1} + \boldsymbol{\epsilon}$

- To test parametric (linear) model vs. nonparametric model

$$H_0 : \ \sigma_u = 0 \quad \text{vs.} \quad H_1 : \ \sigma_u > 0$$

How to choose smoothing parameters?

The smoothing parameters are chosen by minimizing the GCV score

$$GCV(\lambda_x, \lambda_t) = \frac{N||\mathbf{y} - \mathbb{H}\mathbf{y}||^2}{[N - \gamma\, \mathrm{tr}\,(\mathbb{H})]^2} = \frac{N^{-1}||(\mathbb{I} - \mathbb{H})\mathbf{y}||^2}{[N^{-1}\, \mathrm{tr}\,(\mathbb{I} - \gamma\mathbb{H})]^2}$$

- Efficient, rotation invariant version of ordinary cross validation
- $\gamma \geq 1$ is tuning parameter usually selected to be 1.2-1.4 to force GCV to do more smoothing
- Code uses Newton's method for the minimization

Is the FLM "good enough"?

Is the true regression relationship linear?

- Goal: formally test H_0 : FLM vs. H_1 : FGAM

Is the FLM “good enough”?

Is the true regression relationship linear?

- Goal: formally test H_0 : FLM vs. H_1 : FGAM
- Know FLM is special case of FGAM
- Want hypotheses in terms of model parameters
- Not obvious how for our parametrization of $F(x, t)$

Previous work on this problem

Very little

- Almost all work in literature is for testing no functional effect

$$H_0: \ \beta(t) \equiv 0$$

Previous work on this problem

Very little

- Almost all work in literature is for testing no functional effect

  $$H_0: \ \beta(t) \equiv 0$$

Exceptions

- Cramér-von Mises statistic (García-Portugués et al, in press)
  - No penalization for $\beta(t)$. Assumes $\beta(t) = \sum_{j=1}^{p} \theta_j B_j(t)$
- use norm of cross covariance of (X, Y) - Cardot et al. (2003)
  - Never implemented

Mixed Model Representation for FGAM

Using ideas from SS-ANOVA our surface can be expressed as

Term	[Penalty]
$F(x,t)$	$[\lambda_t \int (\frac{\partial^2}{\partial t^2} F)^2 + \lambda_x \int (\frac{\partial^2}{\partial x^2} F)^2]$
$= \ \beta_0 + \beta_1 x + \beta_2 t + \beta_3 x \cdot t$	[unpenalized]
$+ \ f_1(t) + x \cdot f_2(t)$	$[\lambda_1 \int (\frac{\partial^2}{\partial t^2} f_1)^2 + (\frac{\partial^2}{\partial t^2} f_2)^2]$
$+ \ g_1(x) + t \cdot g_2(x)$	$[\lambda_2 \int (\frac{\partial^2}{\partial x^2} g_1)^2 + (\frac{\partial^2}{\partial x^2} g_2)^2]$
$+ \ h(x,t)$	$[\lambda_3 \int (\frac{\partial^4}{\partial x^2 \partial t^2} f)^2]$

Why is this useful?

- Random effect covariances are constant times identity matrix
  - Tensor product decomposed into independent components

Why is this useful?

- Random effect covariances are constant times identity matrix
  - Tensor product decomposed into independent components
- Each component easy to interpret in terms of penalty
  - Identifiability easy to enforce – drop terms from basis

Why is this useful?

- Random effect covariances are constant times identity matrix
  - Tensor product decomposed into independent components
- Each component easy to interpret in terms of penalty
  - Identifiability easy to enforce – drop terms from basis
- Disadvantage: more var. components/smoothing parameters

New Mixed Model Representation for FGAM

It can be shown FGAM has the following LMM representation

$$\mathbf{Y} = \mathbb{L}\left(\mathbb{X}\boldsymbol{\beta} + \sum_{j=1}^{3}\mathbb{Z}_j\boldsymbol{\delta}_j\right) + \epsilon; \quad \boldsymbol{\epsilon} \sim N(0, \sigma_e^2\mathbb{I}_N);$$

$$\boldsymbol{\delta}_j \sim N(0, \sigma_j^2\mathbb{I}); \quad \lambda_j = \sigma_j^2/\sigma_e^2; \quad j = 1, 2, 3;$$

where $\mathbb{L}$ is matrix of quadrature weights,

$$\int F(X_i(t), t)\, dt \approx \sum_{j=1}^{J} \ell_{ij} F\{X_i(t_{ij}), t_{ij}\}$$

New Mixed Model Representation for FGAM

It can be shown FGAM has the following LMM representation

$$\mathbf{Y} = \mathbb{L}\left(\mathbb{X}\boldsymbol{\beta} + \sum_{j=1}^{3} \mathbb{Z}_j \boldsymbol{\delta}_j\right) + \epsilon; \quad \boldsymbol{\epsilon} \sim N(0, \sigma_e^2 \mathbb{I}_N);$$

$$\boldsymbol{\delta}_j \sim N(0, \sigma_j^2 \mathbb{I}); \quad \lambda_j = \sigma_j^2/\sigma_e^2; \quad j = 1, 2, 3;$$

where $\mathbb{L}$ is matrix of quadrature weights,

$$\int F(X_i(t), t)\, dt \approx \sum_{j=1}^{J} \ell_{ij} F\{X_i(t_{ij}), t_{ij}\}$$

Our test for FLM vs. FGAM becomes

$$H_0: \ \sigma_2 = \sigma_3 = 0 \quad \text{vs.} \quad H_1: \ \text{at least one of } \sigma_2 > 0 \text{ or } \sigma_3 > 0$$

I.e. must test two variance components being simultaneously zero

- Also have one nuisance variance component

Tests for Zero Variance Components

- Difficult due to σ_2, σ_3 on boundary of parameter space under H_0
- Standard asymptotics fail because y_i's are not independent
    - Tests are too conservative for spline smoothing
- Exact distribution under null known for one smoothing parameter (Crainiceanu & Ruppert, 2005)

Two Groups of Approaches

- Likelihood Ratio Tests (LRTs) or Restricted LRTs
  - Greven et al., 2008: Fix nuisance effects at BLUPs, use one λ results

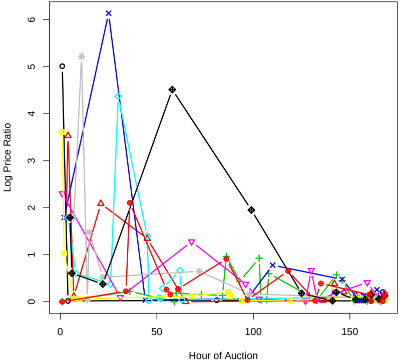
Two Groups of Approaches

- Likelihood Ratio Tests (LRTs) or Restricted LRTs
  - Greven et al., 2008: Fix nuisance effects at BLUPs, use one λ results
- Approximate F tests
  - Wang & Chen, 2012: Quickly compute test stat. over grid of λ's; avoids bootstrap

Two Groups of Approaches

- Likelihood Ratio Tests (LRTs) or Restricted LRTs
  - Greven et al., 2008: Fix nuisance effects at BLUPs, use one λ results
- Approximate F tests
  - Wang & Chen, 2012: Quickly compute test stat. over grid of λ's; avoids bootstrap
- Will these work for testing two components simultaneously zero?
- What about generalized case?

What if $X(t)$ is not fully observed?



From Functional to Longitudinal Data

- Have n_i noisy measurements of each $x_i(t)$

$$\tilde{x}_i(t_{ij}) = x_i(t_{ij}) + e_i(t_{ij}); \qquad e_i(t_{ij}) \overset{\text{i.i.d.}}{\sim} N(0, \sigma_x^2);\ j = 1, \ldots, n_i$$

- n_i's can be very small and t_{ij}'s are irregularly spaced

From Functional to Longitudinal Data

- Have n_i noisy measurements of each $x_i(t)$

$$\tilde{x}_i(t_{ij}) = x_i(t_{ij}) + e_i(t_{ij}); \qquad e_i(t_{ij}) \overset{\text{i.i.d.}}{\sim} N(0, \sigma_x^2);\ j = 1, \ldots, n_i$$

- n_i's can be very small and t_{ij}'s are irregularly spaced
- Can't pre-smooth each curve separately as before
  - Instead, pool data then estimate mean and covariance function
  - "Borrow strength" across curves
  - Represent $X(t)$ in terms of its main modes of variation

Karhunen-Loève Decomposition

Define mean and covariance function

$$\mu_X(t) = E[X(t)], \quad G(s,t) = E\left\{[X(s) - \mu_X(s)][X(t) - \mu_X(t)]\right\}$$

Karhunen-Loève Decomposition

Define mean and covariance function

$$\mu_X(t) = E[X(t)], \quad G(s,t) = E\left\{[X(s) - \mu_X(s)][X(t) - \mu_X(t)]\right\}$$

By Mercer's theorem $G(s,t) = \sum_{m=1}^{\infty} \nu_m \phi_m(s) \phi_m(t)$

- ν's are eigenvalues, ϕ's orthonormal eigenfunctions

Karhunen-Loève Decomposition

Define mean and covariance function

$$\mu_X(t) = E[X(t)], \quad G(s,t) = E\left\{[X(s) - \mu_X(s)][X(t) - \mu_X(t)]\right\}$$

By Mercer's theorem $G(s,t) = \sum_{m=1}^{\infty} \nu_m \phi_m(s)\phi_m(t)$

- ν's are eigenvalues, ϕ's orthonormal eigenfunctions

By Karhunen-Loève theorem

$$X(t) = \mu_X(t) + \sum_{m=1}^{\infty} \xi_{im}\phi_m(t)$$

- ξ_{im} are principal component scores, $\xi_{im} \overset{\text{ind.}}{\sim} (0, \nu_m)$
- $X(t)$ will have estimated ,

PACE - Yao, Müller, Wang (2005)

1. Fit penalized spline to pooled data to estimate $\mu(t)$
2. Est. covariance surface, $\hat{G}(s,t)$, fitting a bivariate smoother to

$$G_i(t_{il}, t_{is}) \equiv [\widetilde{x}_i(t_{il}) - \hat{\mu}_x(t_{il})][\widetilde{x}_i(t_{is}) - \hat{\mu}_x(t_{is})];\ l \neq s;\ i = 1, \dots, N$$

3. σ_x^2 estimated by $\int_0^1 \left[\hat{V}(s) - \hat{G}(s,s)\right] ds$
   - $\hat{V}(s)$ is univariate smooth of $G_i(t_{il}, t_{il})$
4. Estimate ν's and ϕ's from eigendecomposition of estimate in 2.
5. PC scores estimates are BLUPs for normal model, $E[\boldsymbol{\xi}_i | \widetilde{\mathbf{x}}_i]$
   - Avoids numerical integration done by classical FPCA methods

Alt. Mixed Model Formulation of FGAM

- Our penalized spline model has equivalent Bayesian formulation
    - Improper Guassian prior on spline coefficients

Alt. Mixed Model Formulation of FGAM

- Our penalized spline model has equivalent Bayesian formulation
  - Improper Guassian prior on spline coefficients
- Alternative: Separate $F\{\hat{X}_i(\mathbf{t}), \mathbf{t}\} = \mathbb{B}_{\boldsymbol{\xi}_i}\boldsymbol{\theta}$ into

$$F(\hat{X}_i(\mathbf{t}), \mathbf{t}) = \mathbb{B}_{i,0}\boldsymbol{\beta} + \mathbb{B}_{i,p}\boldsymbol{\delta}$$

  - $\mathbb{B}_{i,0}\boldsymbol{\beta}$: unpenalized, fixed effect part
  - $\mathbb{B}_{i,p}\boldsymbol{\delta}$: penalized, random effect part

Alt. Mixed Model Formulation of FGAM

- Our penalized spline model has equivalent Bayesian formulation
    - Improper Guassian prior on spline coefficients
- Alternative: Separate $F\{\hat{X}_i(\mathbf{t}), \mathbf{t}\} = \mathbb{B}_{\boldsymbol{\xi}_i}\boldsymbol{\theta}$ into

$$F(\hat{X}_i(\mathbf{t}), \mathbf{t}) = \mathbb{B}_{i,0}\boldsymbol{\beta} + \mathbb{B}_{i,p}\boldsymbol{\delta}$$

    - $\mathbb{B}_{i,0}\boldsymbol{\beta}$: unpenalized, fixed effect part
    - $\mathbb{B}_{i,p}\boldsymbol{\delta}$: penalized, random effect part
- Decomposition we use gives diagonal, pos. def. penalty matrix
    - $\mathbb{P}(\lambda_x, \lambda_t) \equiv \lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t$, with $\boldsymbol{\Psi}_t$, $\boldsymbol{\Psi}_t$ diag.
    - *Proper* Gaussian prior on $\boldsymbol{\delta}$ with precision matrix $\mathbb{P}(\lambda_x, \lambda_t)$

Alt. Mixed Model Formulation of FGAM

- Our penalized spline model has equivalent Bayesian formulation
  - Improper Guassian prior on spline coefficients
- Alternative: Separate $F\{\hat{X}_i(\mathbf{t}), \mathbf{t}\} = \mathbb{B}_{\boldsymbol{\xi}_i}\boldsymbol{\theta}$ into

$$F(\hat{X}_i(\mathbf{t}), \mathbf{t}) = \mathbb{B}_{i,0}\boldsymbol{\beta} + \mathbb{B}_{i,p}\boldsymbol{\delta}$$

  - $\mathbb{B}_{i,0}\boldsymbol{\beta}$: unpenalized, fixed effect part
  - $\mathbb{B}_{i,p}\boldsymbol{\delta}$: penalized, random effect part
- Decomposition we use gives diagonal, pos. def. penalty matrix
  - $\mathbb{P}(\lambda_x, \lambda_t) \equiv \lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t$, with $\boldsymbol{\Psi}_t$, $\boldsymbol{\Psi}_t$ diag.
  - *Proper* Gaussian prior on $\boldsymbol{\delta}$ with precision matrix $\mathbb{P}(\lambda_x, \lambda_t)$
  - Diffuse prior on $\boldsymbol{\beta}$

Bayesian Hierarchical Model For FGAM

$$Y_i \sim N(\eta_{0i} + \mathbb{Z}_0\boldsymbol{\beta} + \mathbb{Z}_p\boldsymbol{\delta},;\sigma^2); \quad \sigma^2 \sim \mathrm{IG}(a_e, b_e);$$

$$\widetilde{x}_i(t) \sim N\left(\mu(t) + \sum_{m=1}^{M} \xi_{im}\phi_m(t), \sigma_x^2\right); \quad \sigma_x^2 \sim \mathrm{IG}(a_x, b_x);$$

$$\xi_{im} \sim N(0, \nu_m),\ m = 1, \ldots, M;$$

$$\boldsymbol{\delta} \sim N\left(0, [\lambda_t \boldsymbol{\Psi}_t + \lambda_x \boldsymbol{\Psi}_x]^{-1}\right); \quad \lambda_x, \lambda_t \sim \mathrm{Gamma}(a_l, b_l);$$

$$\boldsymbol{\beta} \sim N(0, \sigma_\beta^2 \mathbb{I}); \quad \eta_{0i} \sim N(0, \sigma_\eta^2 \mathbf{I})$$

Why not just use PACE once then fit FGAM?

One could just take two-stage approach

1) Estimate $X(t)$ for sparse observations using PACE
2) Fit FGAM using approach at start of talk

Why not just use PACE once then fit FGAM?

One could just take two-stage approach

1) Estimate $X(t)$ for sparse observations using PACE
2) Fit FGAM using approach at start of talk

However, performance is bad if data highly sparse

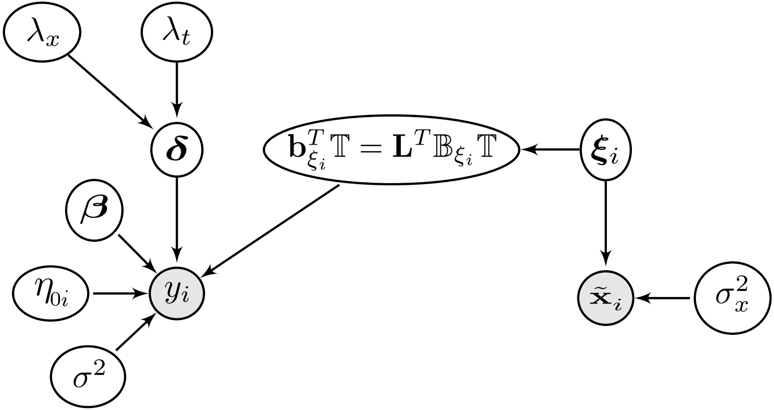
- Does not account for variability in estimated curves
- Can sampled Y's help estimate $X(\cdot)$'s?
- Initial PACE estimates **occasionally** *very* poor

Pseudocode for fitting FGAM to Sparse Data

- Obtain initial estimates for the trajectories $\tilde{\mathbf{x}}$ using "PACE"
- Specify penalties, bases for $F(x, t)$ use above decomposition
- Initialize other parameters

repeat
  for $i = 1 \to N$ **do**
    Update principal component scores, $\boldsymbol{\xi}_i$
    Update $\tilde{\mathbf{x}}_i$
    Update $\mathbb{B}_{i,p}$
  end for
  for $i = 1 \to N$ **do**
    Update terms involving scalar covariates, η_{0i}
  end for
  Update unpenalized spline coefficients, $\boldsymbol{\beta}$
  Update penalized spline coefficients, $\boldsymbol{\delta}$
  Update smoothing parameters, $\lambda_x, \ \lambda_t$
  Update measurement error variance, σ_x^2
  Update response error variance, σ^2
until Max. # iterations reached *OR* [for VB] convergence criteria met

Directed Acyclic Graph



- $p(\boldsymbol{\theta}_l|\text{rest}) = p(\boldsymbol{\theta}_l|\text{Markov blanket of } \boldsymbol{\theta}_l)$
- Markov blanket: all children, parents, and co-parents of node

Complications

- No closed-form for full conditionals for λ_x and λ_t
  - full conditionals are $\propto |\text{Penalty Matrix}|^{1/2}\text{Gamma}(a, b)$

Complications

- No closed-form for full conditionals for λ_x and λ_t
    - full conditionals are $\propto |\text{Penalty Matrix}|^{1/2}\text{Gamma}(a, b)$
- No closed-form for full conditional for the PC scores, $\boldsymbol{\xi}_i$
    - $\boldsymbol{\xi}_i$'s appear in likelihood as arguments to B-splines

Complications

- No closed-form for full conditionals for λ_x and λ_t
  - full conditionals are $\propto |\text{Penalty Matrix}|^{1/2}\text{Gamma}(a, b)$
- No closed-form for full conditional for the PC scores, $\boldsymbol{\xi}_i$
  - $\boldsymbol{\xi}_i$'s appear in likelihood as arguments to B-splines
- Conjugate priors used for all other model parameters

MCMC algorithm

- Standard Gibbs sampling for $\sigma^2,\ \sigma_x^2,\ \boldsymbol{\eta}_{0i},\ \boldsymbol{\beta},\ \boldsymbol{\delta}$
- Independent Metropolis step for updating PC scores
  - Gaussian proposal density, simple form for acceptance probability
- Slice sampling used to update λ_x and λ_t
  - Sample r.v. by uniformly sampling area under its density
  - Easier to tune than a Metropolis update

Idea of Variational Approximation

- Used to make approximate inference about model parameters

Idea of Variational Approximation

- Used to make approximate inference about model parameters
- Regularly used in CS, just now catching on in statistics

Idea of Variational Approximation

- Used to make approximate inference about model parameters
- Regularly used in CS, just now catching on in statistics
- Used in statistics mostly to approximate posterior distributions

Idea of Variational Approximation

- Used to make approximate inference about model parameters
- Regularly used in CS, just now catching on in statistics
- Used in statistics mostly to approximate posterior distributions
- Easy to apply when using conjugate priors

Idea of Variational Approximation

- Used to make approximate inference about model parameters
- Regularly used in CS, just now catching on in statistics
- Used in statistics mostly to approximate posterior distributions
- Easy to apply when using conjugate priors
- Much faster than MCMC,

Idea of Variational Approximation

- Used to make approximate inference about model parameters
- Regularly used in CS, just now catching on in statistics
- Used in statistics mostly to approximate posterior distributions
- Easy to apply when using conjugate priors
- Much faster than MCMC,
    - Can't be made arbitrarily accurate

Idea of Variational Approximation

- Used to make approximate inference about model parameters
- Regularly used in CS, just now catching on in statistics
- Used in statistics mostly to approximate posterior distributions
- Easy to apply when using conjugate priors
- Much faster than MCMC,
    - Can't be made arbitrarily accurate
    - Allows for C.I.s for model parameters to be obtained by resampling as in Goldsmith et al. (2011)
    - Use VB estimates as initial estimates for MCMC algorithm

Idea of Variational Bayes

Approximate posterior, $p(\mathbf{\Theta}|\mathbf{y})$, using "nicer" density, $q(\mathbf{\Theta})$

Idea of Variational Bayes

Approximate posterior, $p(\boldsymbol{\Theta}|\mathbf{y})$, using "nicer" density, $q(\boldsymbol{\Theta})$

- Usually assume $q(\boldsymbol{\Theta})$ factors to form $q(\boldsymbol{\Theta}) = \prod_{l=1}^{G} q_l(\boldsymbol{\theta}_l)$
    - Accuracy of approximation determined by how reasonable this is

Idea of Variational Bayes

Approximate posterior, $p(\boldsymbol{\Theta}|\mathbf{y})$, using “nicer” density, $q(\boldsymbol{\Theta})$

- Usually assume $q(\boldsymbol{\Theta})$ factors to form $q(\boldsymbol{\Theta}) = \prod_{l=1}^{G} q_l(\boldsymbol{\theta}_l)$
  - Accuracy of approximation determined by how reasonable this is
- Using theory of Kullback-Leibler divergence, it can be shown

$$q_l^*(\boldsymbol{\theta}_l) \propto \exp\left\{E_{-\boldsymbol{\theta}_l}[\log p(\boldsymbol{\theta}_l|\text{rest})]\right\},$$

gives the optimal densities for approximating $p(\boldsymbol{\theta}|\mathbf{y})$

Idea of Variational Bayes

Approximate posterior, $p(\mathbf{\Theta}|\mathbf{y})$, using “nicer” density, $q(\mathbf{\Theta})$

- Usually assume $q(\mathbf{\Theta})$ factors to form $q(\mathbf{\Theta}) = \prod_{l=1}^{G} q_l(\boldsymbol{\theta}_l)$
  - Accuracy of approximation determined by how reasonable this is
- Using theory of Kullback-Leibler divergence, it can be shown

$$q_l^*(\boldsymbol{\theta}_l) \propto \exp\left\{E_{-\boldsymbol{\theta}_l}[\log p(\boldsymbol{\theta}_l|\text{rest})]\right\},$$

  gives the optimal densities for approximating $p(\boldsymbol{\theta}|\mathbf{y})$

- Iteratively update parameters deterministically using $q_l^*(\boldsymbol{\theta}_l)$'s
- Convergence monitored via lower bound on marginal likelihood

Factorization

We approximate $p(\boldsymbol{\eta}_{i0}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_N, \lambda_x, \lambda_t, \sigma_x^2, \sigma^2 | \text{data})$ with

$$q(\boldsymbol{\eta}_{i0}, \boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_N, \lambda_x, \lambda_t, \sigma_x^2, \sigma^2) = \\ q^*(\boldsymbol{\eta}_{i0}) q^*(\boldsymbol{\beta}, \boldsymbol{\delta}) q^*(\lambda_x, \lambda_t, \sigma_x^2, \sigma^2) \prod_{i=1}^{N} q^*(\boldsymbol{\xi}_i),$$

which simplifies to

$$q^*(\boldsymbol{\eta}_{i0}) q^*(\boldsymbol{\beta}) q^*(\boldsymbol{\delta}) q^*(\lambda_x) q^*(\lambda_t) q^*(\sigma_x^2) q^*(\sigma^2) \prod_{i=1}^{N} q^*(\boldsymbol{\xi}_i)$$

VB Algorithm For FGAM

- Obtain $q^*(\lambda_x)$ and $q^*(\lambda_x)$ using Gauss-Laguerre quadrature
  - Must be careful to avoid underflow
  - Must approximate: $E_{\lambda_t}\left[|\lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t|\right] \approx |\lambda_x \boldsymbol{\Psi}_x + E_{\lambda_t}[\lambda_t] \boldsymbol{\Psi}_t|$

VB Algorithm For FGAM

- Obtain $q^*(\lambda_x)$ and $q^*(\lambda_x)$ using Gauss-Laguerre quadrature
  - Must be careful to avoid underflow
  - Must approximate: $E_{\lambda_t}\left[|\lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t|\right] \approx |\lambda_x \boldsymbol{\Psi}_x + E_{\lambda_t}[\lambda_t]\boldsymbol{\Psi}_t|$
- Approximate $q^*(\boldsymbol{\xi}_i)$ using a Laplace approximation, $q^*(\boldsymbol{\xi}_i) = N(\boldsymbol{\xi}_{i,0}, \mathbb{H}^{-1})$
  - $\boldsymbol{\xi}_{i,0}$ is mode of unnormalized optimal density, $\tilde{q}(\boldsymbol{\xi}_i)$
  - $\mathbb{H}$ is Hessian of $\tilde{q}(\boldsymbol{\xi}_i)$ evaluated at $\boldsymbol{\xi}_{i,0}$

VB Algorithm For FGAM

- Obtain $q^*(\lambda_x)$ and $q^*(\lambda_x)$ using Gauss-Laguerre quadrature
  - Must be careful to avoid underflow
  - Must approximate: $E_{\lambda_t}\left[|\lambda_x \boldsymbol{\Psi}_x + \lambda_t \boldsymbol{\Psi}_t|\right] \approx |\lambda_x \boldsymbol{\Psi}_x + E_{\lambda_t}[\lambda_t] \boldsymbol{\Psi}_t|$

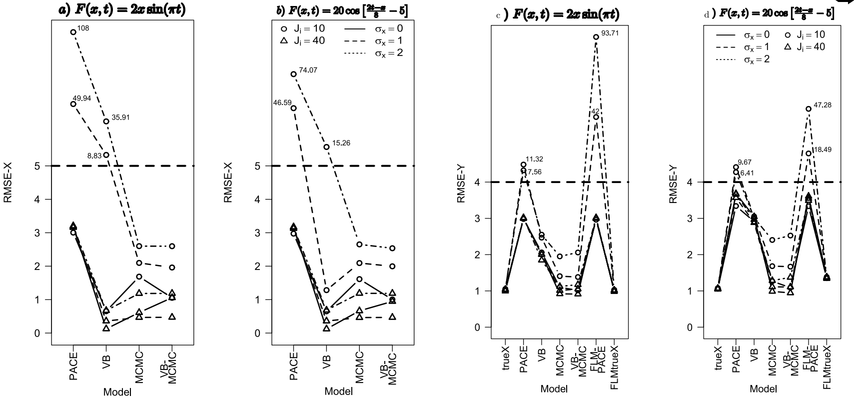
- Approximate $q^*(\boldsymbol{\xi}_i)$ using a Laplace approximation, $q^*(\boldsymbol{\xi}_i) = N(\boldsymbol{\xi}_{i,0}, \mathbb{H}^{-1})$
  - $\boldsymbol{\xi}_{i,0}$ is mode of unnormalized optimal density, $\tilde{q}(\boldsymbol{\xi}_i)$
  - $\mathbb{H}$ is Hessian of $\tilde{q}(\boldsymbol{\xi}_i)$ evaluated at $\boldsymbol{\xi}_{i,0}$

- Also need $E_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i}]$ and $E_{\boldsymbol{\xi}_i}[\mathbf{b}_{\boldsymbol{\xi}_i} \mathbf{b}_{\boldsymbol{\xi}_i}^T]$ for other q^*'s
  - Use 2nd-order Taylor approximation

Simulated Data

- Generate 100 trajectories each with 50 measurements
- Consider three levels of measurement error, $\sigma_x^2 = 0, 1, 2$
- Consider two sparsity levels, $J_i = 10$ or 40
  - 10 or 40 of 50 time points randomly observed for each subject
- Two different true surfaces
  - FLM True Model - $F(x,t) = 2x\sin(\pi t)$
  - Nonlinear True Model - $F(x,t) = 20\cos\left(\frac{2t-x}{8} - 5\right)$
- Four nonzero principal component scores
  - Each method estimates exactly four scores

Results For 100 Simulations



a), b) Mean ISE for recovering trajectories, X

c), d) Mean Out-of-Sample RMSE for predicting Y

- Speed-up of ≈ an order of magnitude for VB vs. MCMC

Real Data - Ebay Auctions

- All received bids from 155 7-day Ebay auctions
- Must convert bids to hourly prices

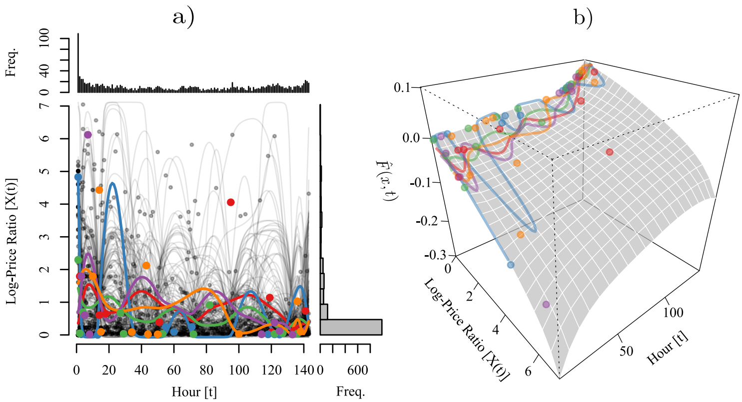
Real Data - Ebay Auctions

- All received bids from 155 7-day Ebay auctions
- Must convert bids to hourly prices
- Each auction usually has three parts
  1. Initial period with some bidding
  2. Middle period with very few bids
  3. Rapid bidding at end of auction (bid sniping)

Real Data - Ebay Auctions

- All received bids from 155 7-day Ebay auctions
- Must convert bids to hourly prices
- Each auction usually has three parts
  1. Initial period with some bidding
  2. Middle period with very few bids
  3. Rapid bidding at end of auction (bid sniping)
- X(t): log-price ratios from first six days of auction as covariates
- Y: closing price

Estimated Surface and Trajectories Using MCMC



a) Observed data, estimated trajectories, & "rug plots"

b) As a) plus estimated surface

Summary

Extended FGAM to handle sparse functional covariates measured with error

FGAM is

- Intuitive extension of additive models to functional data
- Highly flexible AND highly interpretable
- Easily estimated using penalized regression splines
- Serves as useful diagnostic for checking FLM
- Extensions to sparse functional covariates available

References

- For details on the fully-observed predictor case see

M.W. McLean, G. Hooker, A.-M. Staicu, F. Scheipl, D. Ruppert. Functional Generalized Additive Models. *Journal of Computational and Graphical Statistics* 23.1, pp. 249–269.

- For details on the sparse predictor case see

M.W. McLean, F. Scheipl, G. Hooker, S. Greven, D. Ruppert. Bayesian Functional Generalized Additive Models with Sparsely Observed Covariates. *Submitted*. `arXiv:1305.3585v2`.

- M.W. McLean, G. Hooker, D. Ruppert. Restricted Likelihood Ratio Tests for Linearity in Scalar on Function Regression. In: *Statistics and Computing* 25.5, pp. 997-1008.
- A copy of the papers and R code can be obtained from

`https://mwmclean.github.io/`