Functional Generalized Additive Models

A new model for regression with functional predictors

Mathew McLean

Cornell University
School of Operations Research and Information Engineering
mwm79@cornell.edu

August 31, 2012

1. Introduction to Nonparametric Regression
2. Overview of Functional Data Analysis
3. Functional Generalized Additive Models
   - Estimation
   - Approximate Inference
4. Numerical Results
   - Simulations
   - Diffusion Tensor Imaging Data
5. Extensions
   - Non-Identity Link GAMs
   - Multiple Predictors
   - Sparse, Noisy Predictor Functions - Current Work

Linear Model

- Observe N data vectors: $(y_i, x_{i1}, \ldots, x_{ip}) \in \mathbb{R}^{p+1};\; i = 1 \ldots, N$
  - $p < N$, less predictors than samples
- Want to predict r.v. Y given $x_1, \ldots, x_p$
- Simplest approach: Linear Model

LM

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i$$

where $\epsilon_i \overset{\text{i.i.d.}}{\sim} N(0, \sigma^2)$ and β's are unknown coefficients

Linear Model (LM)

- In matrix notation:

$$\mathbf{Y} = \mathbb{X}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

  Design matrix: $\mathbb{X} = [\mathbf{1}\ \mathbf{x}_1 \cdots \mathbf{x}_p]$
- Using least squares (or equivalently maximum likelihood)

$$\widehat{\boldsymbol{\theta}} = (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\mathbf{Y},$$

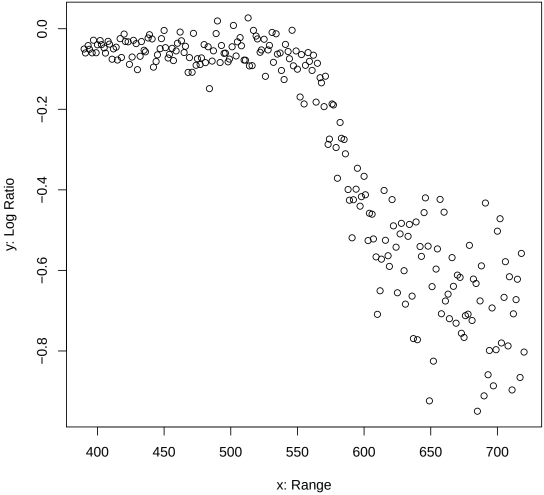
- Predicted Values:

$$\hat{\mathbf{Y}} = \mathbb{X}\widehat{\boldsymbol{\theta}} = \mathbb{H}\mathbf{Y},$$

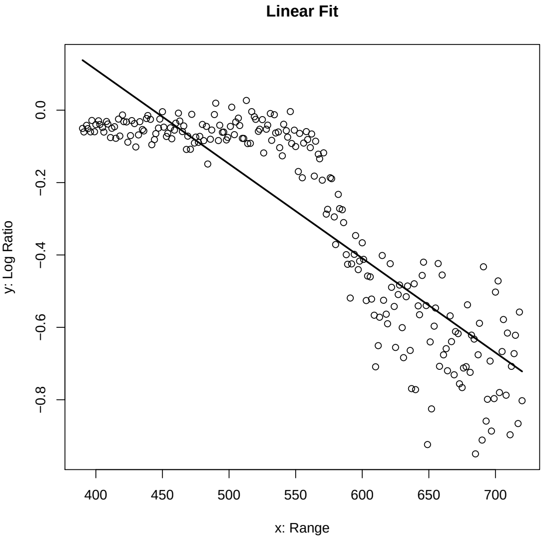
  Hat Matrix: $\mathbb{H} = \mathbb{X}(\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T$

LIDAR Data: $p = 1$

- Y: log-ratio of received light from two lasers
- X: distance travelled before light is reflected back to source



LIDAR Data


Linear Fit
y: Log Ratio
0.0
−0.2
−0.4
−0.6
−0.8
400
450
500
550
600
650
700
x: Range

Need something more general

- Could try

$$Y_i = f(x_{i1}, \ldots, x_{ip}) + \epsilon_i$$

  f is unknown surface estimated from data
- Very hard to estimate for even moderately large p
  - Known as curse of dimensionality
  - Need more and more data to avoid huge variance in estimates
- Need to restrict class of f's we consider

Additive Model (AM)

AM

$$Y_i = \theta_0 + \sum_{j=1}^{p} f_j(x_{ji}) + \epsilon_i$$

f_j's are unknown, smooth (f'' cont.) functions estimated from data

- Note: f_j's only identified up to a constants
- Need constraint, $E[f_j(X_j)] = 0$ for all j

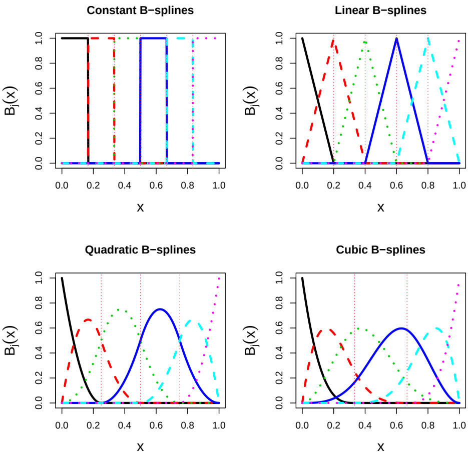
How to Represent f_j's?

- Using linear combination of "basis functions", $B_k(\cdot)$,

$$f(x) \equiv \sum_{k=1}^{K} \theta_k B_k(x) = \boldsymbol{\theta}^T \mathbf{B}_x$$

- $\mathbf{B}_x$ could be polynomials, Fourier series, wavelets, splines, etc.
  - Most common: splines - piecewise polynomials
  - Need to specify knots and order of the polynomials
  - Many varieties - B-splines are most popular

Univariate B-splines of Diff. Order



- Choose order at least 2 greater than highest deriv. of interest
- Can have poor fits at boundary

Regression Splines

One predictor AM fit with B-splines:

$$Y_i = \theta_0 + f(x_i) + \epsilon_i = \theta_0 + \sum_{k=1}^{K} \theta_k B_k(x_i) + \epsilon_i$$

Regression Splines

One predictor AM fit with B-splines:

$$\mathbf{Y} = \mathbb{B}\boldsymbol{\theta} + \boldsymbol{\epsilon}$$

Design matrix: $\mathbb{B} = [\mathbf{1}\ \mathbf{B}_1(\mathbf{x}) \cdots \mathbf{B}_K(\mathbf{x})]$

- Least squares estimates:

$$\widehat{\boldsymbol{\theta}} = (\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T\mathbf{Y},$$

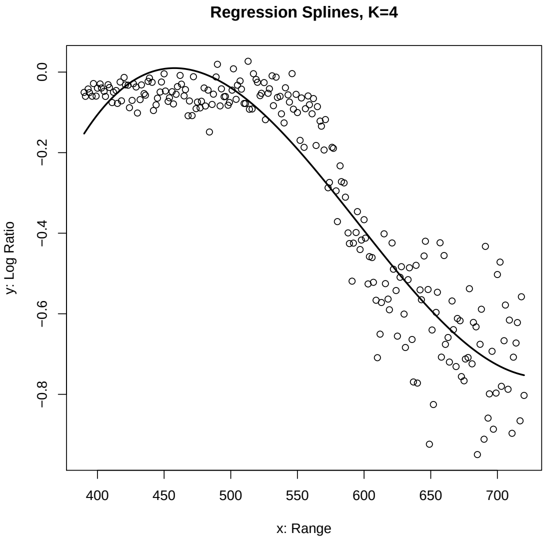
- Predicted Values:

$$\hat{\mathbf{Y}} = \mathbb{B}\widehat{\boldsymbol{\theta}} = \mathbb{H}\mathbf{Y},$$

  Hat Matrix: $\mathbb{H} = \mathbb{B}(\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T$

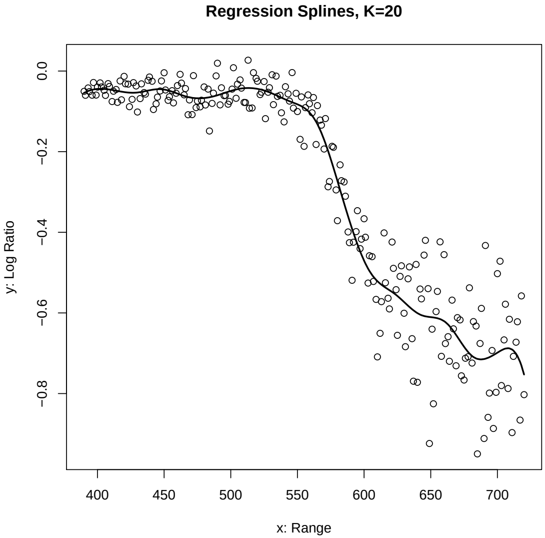
- LM vs. regression splines: $\mathbb{X} \leftrightarrow \mathbb{B} \qquad p \leftrightarrow K$

LIDAR Data - Too Few Splines

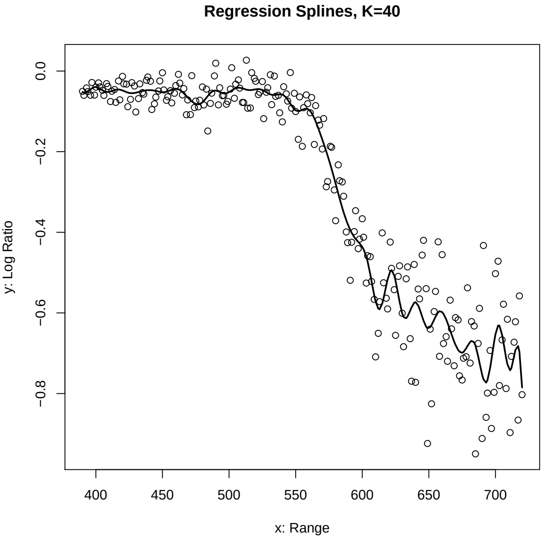


- Over-smoothing aka under-fitting

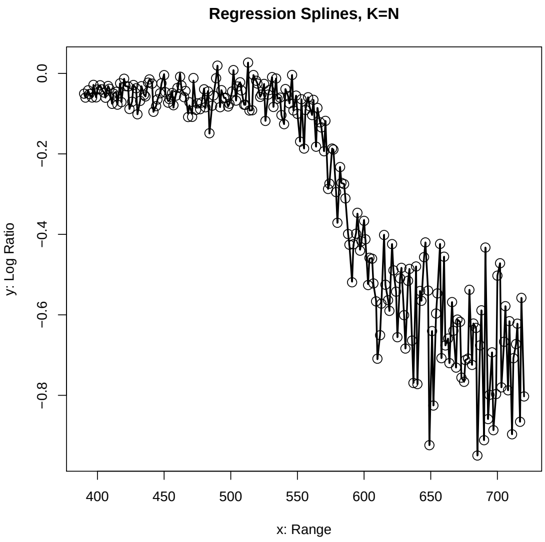
LIDAR Data - Better



LIDAR Data - Under-smoothing



LIDAR Data - Interpolating Data



Penalized Regression Splines

Idea

Control smoothness by penalizing some measure of complexity of f

- Often used penalty is: $\int [f''(t)]^2 \, dt$
- Our objective function is (quadratic program):

$$L(\boldsymbol{\theta}) = (\mathbf{Y} - \mathbb{B}\boldsymbol{\theta})^T(\mathbf{Y} - \mathbb{B}\boldsymbol{\theta}) + \lambda\boldsymbol{\theta}^T\mathbb{P}\boldsymbol{\theta}$$

  where $\mathbb{P}$ is positive semi-definite matrix incorporating penalty
- λ controls amount of smoothing
- $\lambda \to 0$: R.Spline fit; $\lambda \to \infty$: polynomial fit
- Bias-Variance trade-off: Introducing bias to reduce variance

Penalized Regression Splines

- Objective function to be minimized:

$$L(\boldsymbol{\theta}) = (\mathbf{Y} - \mathbb{B}\boldsymbol{\theta})^T(\mathbf{Y} - \mathbb{B}\boldsymbol{\theta}) + \lambda\boldsymbol{\theta}^T\mathbb{P}\boldsymbol{\theta}$$

- Solution is

$$\widehat{\boldsymbol{\theta}} = (\mathbb{B}^T\mathbb{B} + \lambda\mathbb{P})^{-1}\mathbb{B}^T\mathbf{Y}$$

- Hat Matrix

$$\mathbb{H} = \mathbb{B}(\mathbb{B}^T\mathbb{B} + \lambda\mathbb{P})^{-1}\mathbb{B}^T$$

  $\text{tr}\,(\mathbb{H}) =$ effective degrees of freedom.
    - Measures effective number of parameters in fit
    - Value of λ not informative for quantifying amount of smoothing
    - $q + 1 < \text{tr}(\mathbb{H}) < q + 1 + K$ where q is degree of spline

Penalized Regression Splines

- Objective function to be minimized:

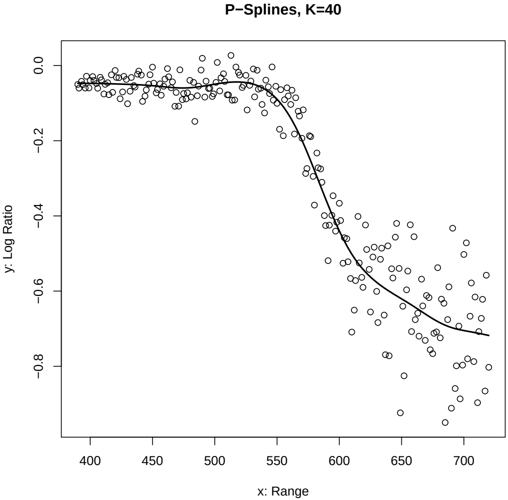
$$L(\boldsymbol{\theta}) = (\mathbf{Y} - \mathbb{B}\boldsymbol{\theta})^T(\mathbf{Y} - \mathbb{B}\boldsymbol{\theta}) + \lambda\boldsymbol{\theta}^T\mathbb{P}\boldsymbol{\theta}$$

- Solution is

$$\widehat{\boldsymbol{\theta}} = (\mathbb{B}^T\mathbb{B} + \lambda\mathbb{P})^{-1}\mathbb{B}^T\mathbf{Y}$$

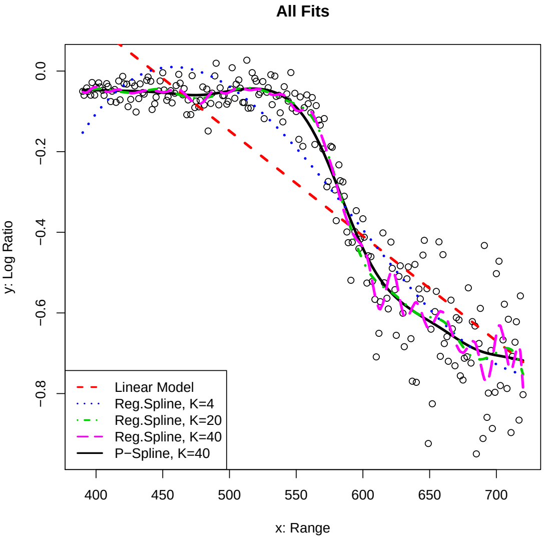
- Note: Can handle "$p > N$"
    - No model selection
- Possible to formulate as a mixed effects model

LIDAR Data



- P-splines: Specific type of penalized spline smooth w\ B-splines
- No boundary effects

LIDAR Data


All Fits
y: Log Ratio
x: Range
0.0
−0.2
−0.4
−0.6
−0.8
400
450
500
550
600
650
700
Linear Model
Reg.Spline, K=4
Reg.Spline, K=20
Reg.Spline, K=40
P−Spline, K=40

Generalized Linear Model (GLM)

- Y comes from any exponential family distribution,

$$Y_i \overset{\text{i.i.d.}}{\sim} EF(\mu_i, \phi),$$

  $\mu_i = E(Y_i)$
  ϕ: Dispersion parameter ($\phi = \sigma^2$ for Normal data)

- e.g. Bernoulli, Binomial, Poisson, Gamma, etc.

GLM

$$g(\mu_i) = \mathbb{X}\boldsymbol{\theta}$$

Link function, g: monotonic, differentiable (usually known)

- g is identity function for Normal data

1. Introduction to Nonparametric Regression
2. Overview of Functional Data Analysis
3. Functional Generalized Additive Models
   - Estimation
   - Approximate Inference
4. Numerical Results
   - Simulations
   - Diffusion Tensor Imaging Data
5. Extensions
   - Non-Identity Link GAMs
   - Multiple Predictors
   - Sparse, Noisy Predictor Functions - Current Work

FDA Intro

- Sampling units are functions, $X_i(t)$, instead of scalars/vectors
- Key assumption: functions are smooth
- In practise: $X(t)$ observed on finite grid and pre-smoothed.
- Often derivatives of $X(t)$ are of interest
- Multivariate analysis with sums replaced by integrals

Examples of Functional Data

- Time Series: $X(t)$ is temperature on day t at a weather station
- DTI: $X(t)$ is some measure of diffusion at position t in tract
- Tracking movements of points in space: $X(t),\ Y(t)$
    - X-Y coordinates of pen on paper at time t
- $p(x)$ is a probability density
- Image analysis: Bivariate functional data

Common Tools for FDA

- Functional descriptive statistics, e.g. mean function $\mu_x(t)$
- Registration - line up features (e.g. zero crossings) of curve
- Functional Principal Components Analysis (fPCA)
  - Exploratory technique for identifying important features
- Dynamics - Differential Equations models involving $X(t)$
- Functional Regression - response and/or predictor are functions

Functional Regression - Setup

- Goal: predict Y using function $X : \mathcal{T} \to \mathcal{X}$; $\mathcal{T}$ closed interval
- For now continuous, normally distributed response with one functional predictor (will be relaxed later)
- $X(t)$ observed at finite number of points in $\mathcal{T}$ and presmoothed
- $\mathcal{T} = [0, 1]$ w.l.o.g.

Functional Linear Model (FLM)

- The most commonly used functional regression model:

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

  $\beta(t)$ is unknown coefficient function to be estimated from data
- Effect of X on Y is linear for each t (Easy to interpret)

- Is goal prediction or estimating $\beta(\cdot)$?

FLM as limit of LM

- Back to multivariate data: Observe function at finite number of points

$$x_{ij} \equiv X_i(t_j);\ j = 1\ldots, J$$

- Linear Model:

$$E(Y_i|X_{i1},\ldots,X_{iK}) = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} = \beta_0 + \sum_{j=1} \beta_j^* X_i(t_j)\Delta t_j$$

(think Riemann sum)

- Letting $J \to \infty$ we arrive at

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

Functional Linear Model (FLM)

- The most commonly used functional regression model:

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

- Effect of X on Y is linear for each t (Easy to interpret)
- Linear Model with an infinite number of predictors (limit of Riemann sum approximation)
- Coefficient function commonly estimated in of two ways
  1) Using B-splines and roughness penalty
  2) Using function principal components

Extending the FLM

- Linear Model not general enough to model complex relationships between response and predictor functions
- How to improve FLM in a way that is:
  1) Highly flexible (low bias)
  2) Avoids curse of dimensionality (low variance)
  3) Easy to interpret (not a "black box")
  4) Has FLM as a special case

An Additive Model With Functional Predictor - FGAM

- The model we propose is

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

  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 smoothing parameters (Using only one not justified here)
- If $F(x,t) = \beta(t)x$, we get the FLM

An Additive Model With Functional Predictor - FGAM

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

- Compare with the additive model

$$x_{ij} \equiv X_i(t_j) \quad f_j(\cdot) \equiv F(\cdot, t_j)\Delta t_j$$

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

- Let $J \to \infty$ arrive at FGAM

How to represent $F(x,t)$?

- We will use tensor products of univariate B-splines
  - Instead of:

$$F(x,t) = \sum_{j=1}^{K} \theta_j B_j(x,t)$$

  - We have:

$$F(x,t) = \sum_{j=1}^{K_1} \sum_{j=1}^{K_2} \theta_{jk} B_j^X(x) B_j^T(t)$$

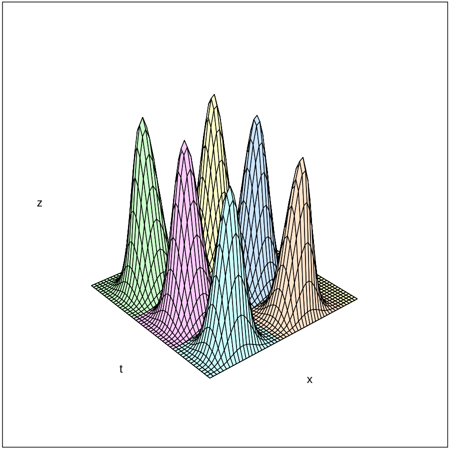
How to represent $F(x, t)$?

$F(x, t)$ becomes

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

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

Tensor Product B-splines



Putting It Together

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

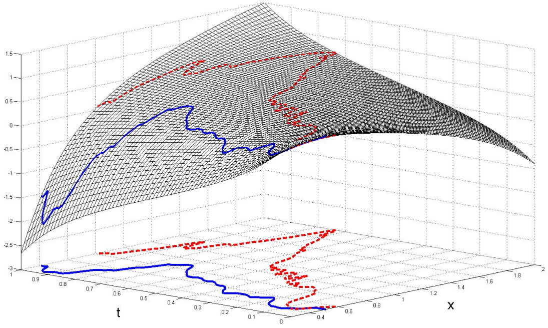
$$F(x,t) = \sum_{j=1}^{K_x} \sum_{k=1}^{K_t} \theta_{jk} B_j^X(x) B_k^T(t)$$

- The model becomes

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

  - $Z_{jk}(i) = \int_{\mathcal{T}} B_j^X\{X_i(t)\} B_k^T(t)\, dt$
  - $\mathbb{Z}$ is $N \times (1 + K_x K_t)$ matrix of $Z_{jk}(i)$ with first column $\mathbf{1}$

Example Estimated Surface



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

- The solid curve belongs to a control and the dashed curve belongs to an MS patient.

Identifiability

- Define $F^*(x,t) = F(x,t) + g(t)$, where $\int_{\mathcal{T}} g(t)\,dt = 0$
Notice that
$$\int_{\mathcal{T}} F^*(x,t)\,dt = \int_{\mathcal{T}} F(x,t)\,dt$$

BAD! Model is not identifiable

- Need to use constraints to ensure identifiability and interpretability of our model.
- Also check for numerical rank deficiency during fitting
- Specific constraint not too important, except when constructing confidence bands

Transforming the Functional Predictor

Idea: Transform functional predictor, $X(t)$, to say, $G_t(x) = G \circ X(t)$

- The new surface to be estimated is $F(g, t)$
- Estimation procedure is the same
- Why?
  - Improve predictive performance
    - E.g. Use lth order derivative $\frac{d^l}{dt^l} X(t)$ instead of $X(t)$
  - Ensure new predictor data falls inside range of marginal basis for X
    - E.g. Quantile transformation: $\widehat{G}_t(x) = n^{-1} \sum_{i=1}^{n} \mathbb{1}_{\{X_i(t) < x\}}$

Comments on Penalized Regression Splines

- Multiple smoothing parameters estimated simultaneously
- No iteration necessary (Note: backfitting not possible here)
- Fast, numerically stable fitting methods
- Easily extends to additional predictors and other exponential family distributions
- We use a specific type known as P-splines (Marx & Eilers, 1996)
  - Other types of bases and penalties possible
  - Not as lacking in theory as they used to be

Penalized Likelihood Estimation

Penalized least squares objective function:

$$(\mathbf{Y} - \mathbb{Z}\boldsymbol{\theta})^T(\mathbf{Y} - \mathbb{Z}\boldsymbol{\theta}) + \boldsymbol{\theta}^T\mathbb{P}\boldsymbol{\theta}$$

- $\mathbb{P} = \lambda_x\mathbb{P}_x + \lambda_t\mathbb{P}_t$ incorporates difference penalties on X and t
- Closed form solution for the unconstrained parameters is given by
$$\widehat{\boldsymbol{\theta}} = (\mathbb{Z}^T\mathbb{Z} + \mathbb{P})^{-1}\mathbb{Z}^T\mathbf{Y}$$
- Check for rank deficiency during fitting

Generalized Cross Validation - GCV

The smoothing parameters are chosen by minimizing the GCV score

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

- $\mathbb{H} = \mathbb{Z}(\mathbb{Z}^T\mathbb{Z} + \mathbb{P})^{-1}\mathbb{Z}^T$ is the hat matrix ($\hat{\mathbf{y}} = \mathbb{H}\mathbf{y}$)
- $\gamma \geq 1$ is tuning parameter usually selected to be 1.2-1.4 to force GCV to do more smoothing

Variance of the estimated surface

Some possibilities

- Sandwich estimator: okay if bias is small
- (empirical) Bayesian estimator: attempts to account for bias
- Use bootstrap: account for bias and uncertainty λ's

Confidence Bands for True Surface

- Interval from Bayesian estimator recommended for our implementation,
- C.I. based on SW estimator under-covered for nonlinear $F(\cdot,\cdot)$
- Bayesian interval has good "average" performance
  - coverage close to nominal when averaged across all $\mathbf{x}$ and $\mathbf{t}$
- Coverage can still be poor at individual x_i and t_j values

Testing for Constant Surface or FLM

- Can test $H_0 : \boldsymbol{\theta} = \mathbf{0}$ and $H_0 : \mathbf{F} = \mathbf{0}$ using sandwich estimator
- Notice $\frac{\partial^2}{\partial x^2} F(x,t) = 0$ for all x and t implies

$$F(x,t) = \beta(t)x$$

- Can construct confidence bands for $\frac{\partial^2}{\partial x^2} F(x,t)$ to check FLM
- Easy to do since derivatives of B-splines are easy to compute

Implementation

- Our code will soon be available in R package `refund`
- Estimation is done using the `mgcv` package of Wood

1. Introduction to Nonparametric Regression
2. Overview of Functional Data Analysis
3. Functional Generalized Additive Models
   - Estimation
   - Approximate Inference
4. Numerical Results
   - Simulations
   - Diffusion Tensor Imaging Data
5. Extensions
   - Non-Identity Link GAMs
   - Multiple Predictors
   - Sparse, Noisy Predictor Functions - Current Work

Other Functional Regression Models

- FLM with roughness penalty (FLM1)
- FLM with fPCA (FLM2)
- Functional Additive Model of Yao+Müller: GAM in f.p.c. scores (FAM)
- Fully nonparametric kernel estimator of Ferraty+Vieu (FV):

$$\widehat{r}(X) = \frac{\sum_{i=1}^{N} Y_i \, K\left\{h^{-1} d(X, X_i)\right\}}{\sum_{i=1}^{N} K\left\{h^{-1} d(X, X_i)\right\}},$$

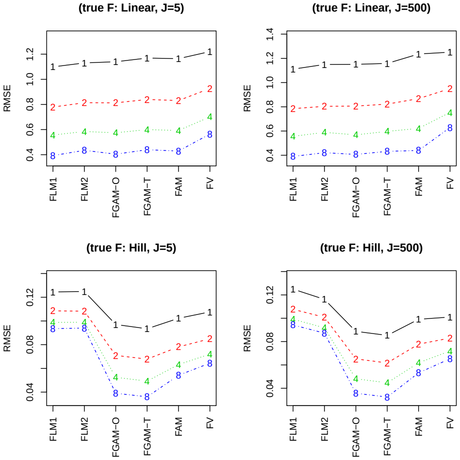
where K is an asymmetrical kernel with bandwidth h and d is a semimetric.

Data Generation

- 1000 simulations, $N = 100$ curves (67 for training, 33 for testing) sampled at 200 points in $\mathcal{T} = [0, 1]$
- $X_i(t) = \sum_{j=1}^{J} \gamma_j [Z_{1ij}\phi_{1j}(t) + Z_{2ij}\phi_{2j}(t)]$ where

  $Z_{hij} \sim N(0, \frac{4}{j^2})$, $\phi_{1j}(t) = \sqrt{2}\cos(\pi j t)$, $\phi_{1j}(t) = \sqrt{2}\sin(\pi j t)$
- J controls smoothness of X
- 1) $F(x,t) = xt$ and 2) $F(x,t) = -.5 + \exp\left[-(\frac{x}{5})^2 - (\frac{t-.5}{.3})^2\right]$.
- The error variance changes each sample so that the empirical signal to noise ratio (SNR) remains constant.
- $K_x = 6,\ K_t = 7,\ d_x = d_t = 2,\ \gamma = 1.0$ and cubic B-splines

Predictive Performance - Median RMSE

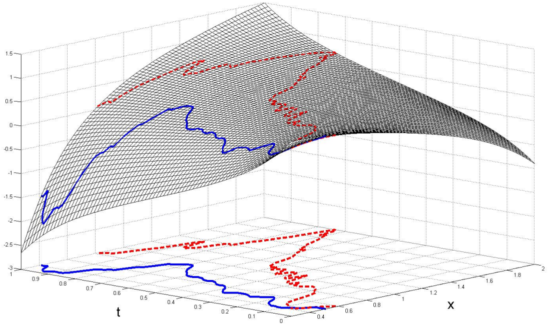


- Four different empirical signal to noise ratios: 1, 2, 4, 8
- Rough (J=500) and smooth (J=5) predictor functions.

Diffusion Tensor Imaging

- Study comparing brains images of subjects with Multiple Sclerosis with healthy controls
- At each of 93 locations in several tracts of the brain, measure diffusion of water which is summarized by a 3×3 symmetric, positive-definite matrix
- 3 functional measurements summarizing the diffusion:
  - Parallel diffusivity - largest eigenvalue
  - Perpendicular diffusivity - average of two other eigenvalues
  - Fractional anisotropy (=0 if isotropic diffusion)
- Response is PASAT score: a cognitive test scored from 0-60, administered to MS patients only
- MS patients are known to perform poorly on this test

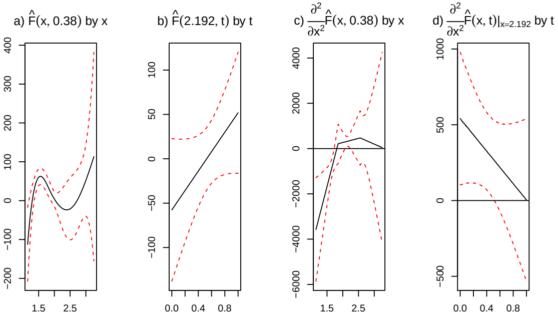
Estimated Surface



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

- The solid curve belongs to a control and the dashed curve belongs to an MS patient.
- $K_x = 6,\ K_t = 7,\ d_x = d_t = 2,\ \gamma = 1.4$ and cubic B-splines

Fixed slices of $\widehat{F}$ and $\partial^2/\partial x^2 \widehat{F}(x,t)$



- Untransformed parallel diffusivity with PASAT score as the response variable.

Leave-One-Curve-Out Prediction Error

- RMSE$=\left[N^{-1}\sum_{i=1}^{N}(y_i-\widehat{y}_{(i)})^2\right]^{1/2}$,

$\widehat{y}_{(i)}$ is the predicted value of the ith response when that sample is left out of the estimation

Measurement	FGAM-O	FGAM-T	FLM1	FLM2	FV	FAM
Perp. Diffusivity	12.22	10.46	10.98	11.27	11.16	11.71
Frac. Anisotropy	12.55	11.60	11.87	11.91	12.11	12.70
Para. Diffusivity	11.94	12.09	12.32	12.24	11.97	11.86

- FGAM with quantile transformation seems to perform best for this example

1 Introduction to Nonparametric Regression

2 Overview of Functional Data Analysis

3 Functional Generalized Additive Models
Estimation
Approximate Inference

4 Numerical Results
Simulations
Diffusion Tensor Imaging Data

5 Extensions
Non-Identity Link GAMs
Multiple Predictors
Sparse, Noisy Predictor Functions - Current Work

Extension to Other Link Functions

- Easy to extend to Y from any exponential family distribution
- P-ILRS now used for fitting
- GCV score uses deviance in numerator
- Use outer iteration: Penalized GLM fit for each pair of smoothing parameters

Adding Additional Predictors

Fitting a model such as

$$g\{E(Y_i|X_{i,1}, X_{i,2}, W_i)\} = \theta_0 + \int_{\mathcal{T}_1} F_1\{X_{i,1}(t), t\}dt + \int_{\mathcal{T}_2} F_2\{X_{i,2}(t), t\}dt + f_3(W_i),$$

is easy due to the modularity of penalized splines

- This model would have three constraints and five smoothing parameters

Idea of Variational Approximation

- Approximate solution to optimization problem by restricting class of functions being considered
- Used in statistics mostly to approximate posterior distributions, usually by assuming density factors
- Easy to apply in same situations where Gibbs Sampler can be used.
- Much faster than MCMC, but cannot be made arbitrarily accurate

Why use Variational Bayes with FGAM?

- A Bayesian Mixed Model approach will allow for the handling of partially observed predictor curves measured with error
- Using a Variational Approximation avoids the computational burden of MCMC
- Bootstrap Confidence Intervals can be obtained for all model parameters

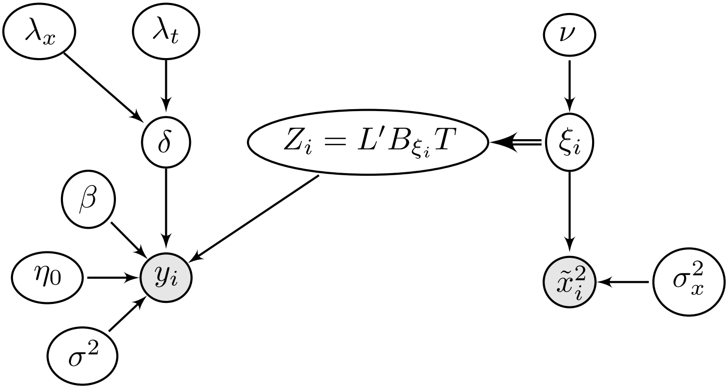
New Setup

- $\tilde{x}_i(t) = \mu_x(t) + \sum_{m=1}^{M} \xi_{im}\phi_m(t), \quad \xi_{im} \sim N(0, \nu_m)$
  - All initially estimated using fPCA: PACE (Yao, Müller & Wang, 2005)
- Improper Gaussian prior for $\boldsymbol{\theta}$: $p(\boldsymbol{\theta}|\lambda_x, \lambda_t) \propto \exp\left(-\frac{1}{2}\boldsymbol{\theta}^T \mathbb{P}\boldsymbol{\theta}\right)$
- Use mixed model representation to avoid numerical issues due to rank deficiency of penalty

$$\int_{\mathcal{T}} F(\tilde{x}_i(t), t) \approx \mathbf{L}^T \mathbb{B}_{\boldsymbol{\xi}_i}\boldsymbol{\theta} = \mathbf{L}^T \mathbb{B}_{\boldsymbol{\xi}_i}\mathbb{T}_0\boldsymbol{\beta} + \mathbf{L}^T \mathbb{B}_{\boldsymbol{\xi}_i}\mathbb{T}_p\boldsymbol{\delta}$$

- $\mathbf{L}$ is vector of quadrature weights
- $\mathbb{B}_{\boldsymbol{\xi}_i}$ is matrix of tensor product B-spline evaluations
- $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ are coefficients for unpenalized and penalized parts of $F(\cdot, \cdot)$, respectively

Directed Acyclic Graph



Complications

- The two smoothing parameters are difficult to separate
  - Use numerical integration
- Density for Y depends nonlinearly on the principal component scores, $\boldsymbol{\xi}_i$.
  - Use Laplace Approximation for optimal density
  - Use Newton's method to find mode. Scaling important to speed convergence
- Other parameters have closed-form optimal densities due to use of conjugate priors

Current Status

- Full algorithm for updating all parameters developed and implemented in R
- Issues updating p.c. scores
- Difficulty in choosing step size for optimizer, numerical errors

Acknowledgements

- Thanks to the Natural Sciences and Engineering Research Council of Canada for support
- Thanks to James Davis for the LaTeX Beamer theme
- Like “Jim Sucks” on Facebook