Functional Data Analysis and Applications

Introduction

Haipeng Shen

Innovation and Information Management
Faculty of Business and Economics
University of Hong Kong

July 2016

Outline

FDA

- acronym for Functional Data Analysis
  - not for *Food and Drug Administration*,
  - Google does not work here, try Ramsay's FDA site: `http://www.psych.mcgill.ca/misc/fda/`
- J. S. Marron's personal view: "atom" of the statistical analysis
  - 1st statistical course: "atoms" are numbers
  - multivariate analysis: "atoms" are vectors
  - FDA: "atoms" are curves or functions, often smooth
  - Object oriented data analysis (OODA): "atoms" are more complex data objects such as images, shapes, blood vessel trees, ...

FDA: some examples

- collection of independent curves
- two sets of curves
- time series of curves
- two-dimensional smooth surface
- spatial-temporal process
- ...

Example: girl height curve (RS, 2005)

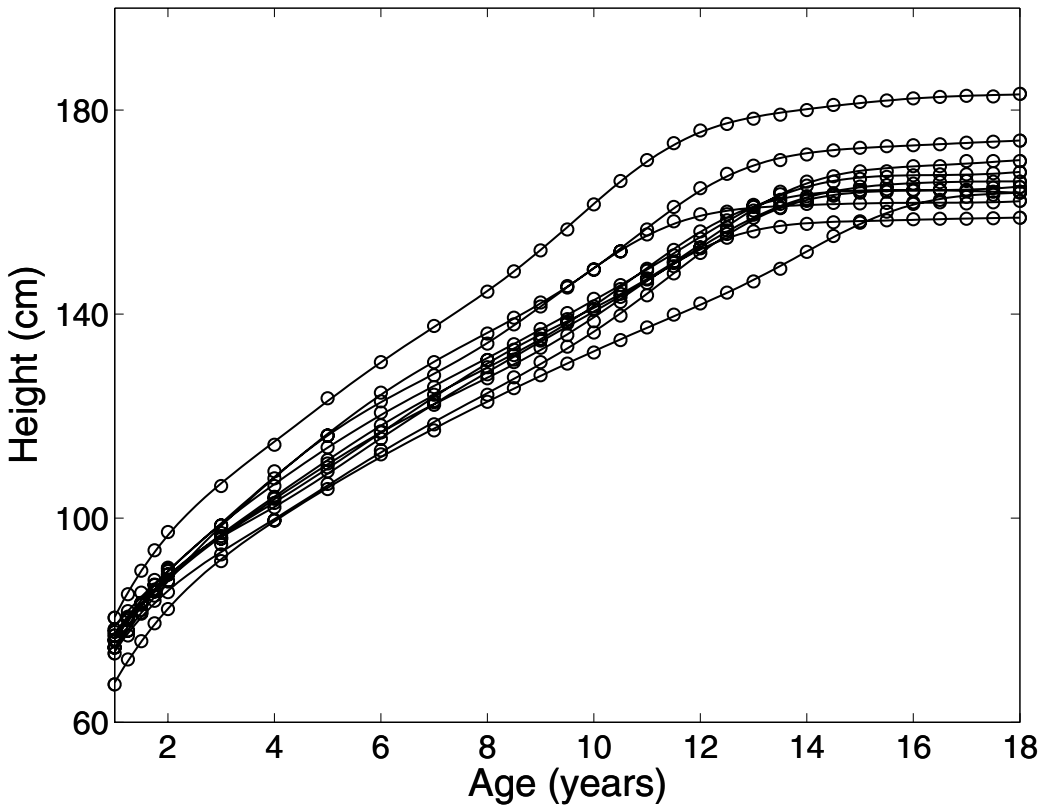


Figure 1.1. The heights of 10 girls measured at 31 ages. The circles indicate the unequally spaced ages of measurement.

Underlying growth process, smooth, monotone increasing, ...

Example: human walking motion (RS, 2005) - two sets of curves

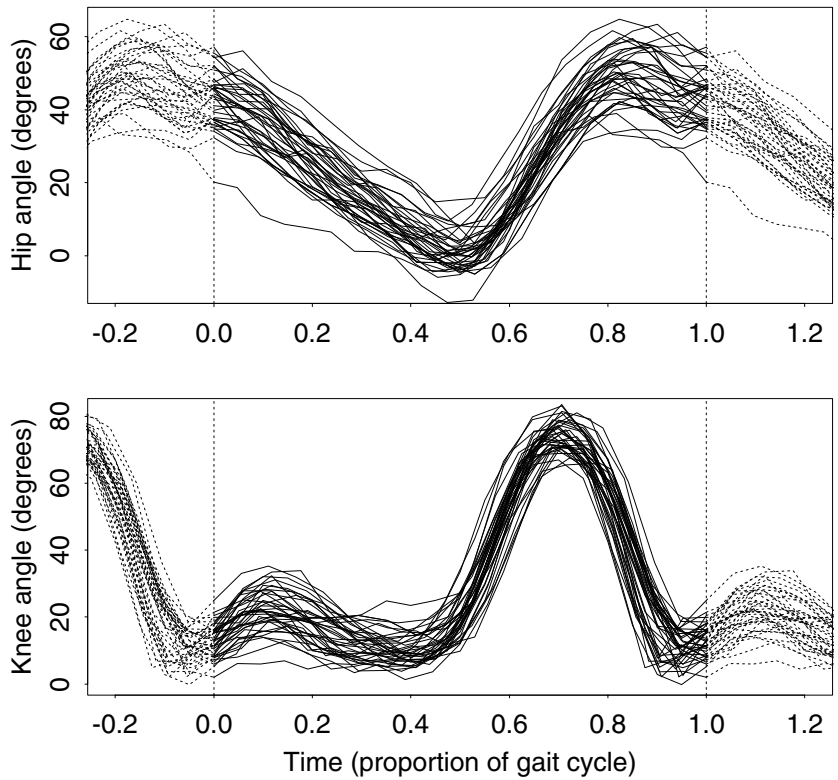


Figure 1.8. The angles in the sagittal plane formed by the hip and by the knee as 39 children go through a gait cycle. The interval $[0, 1]$ is a single cycle, and the dotted curves show the periodic extension of the data beyond either end of the cycle.

Example: human walking motion - interaction between hip and knee)

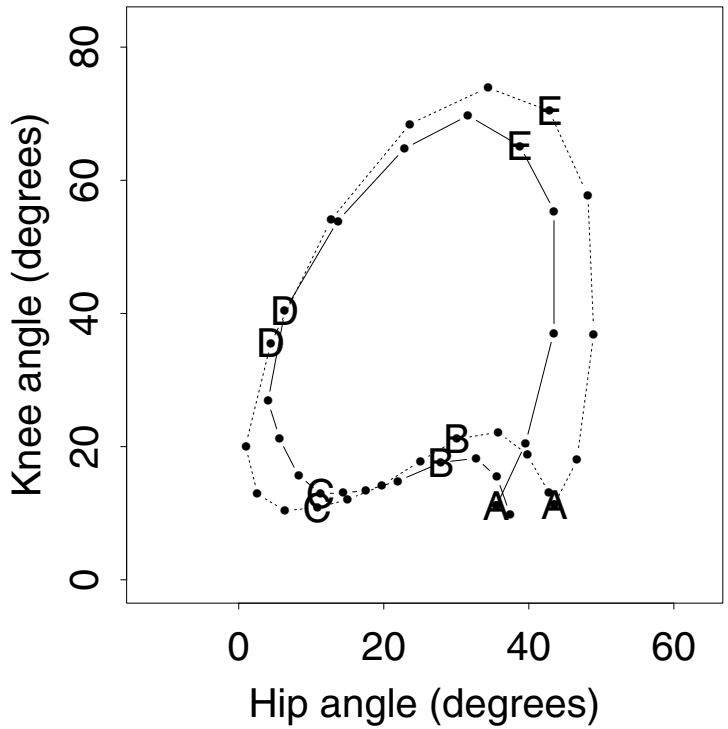


Figure 1.9. Solid line: The angles in the sagittal plane formed by the hip and by the knee for a single child plotted against each other. Dotted line: The corresponding plot for the average across children. The points indicate 20 equally spaced time points in the gait cycle, and the letters are plotted at intervals of one-fifth of the cycle, with A marking the heel strike.

Example: pinch force (RS, 2005) - curve registration

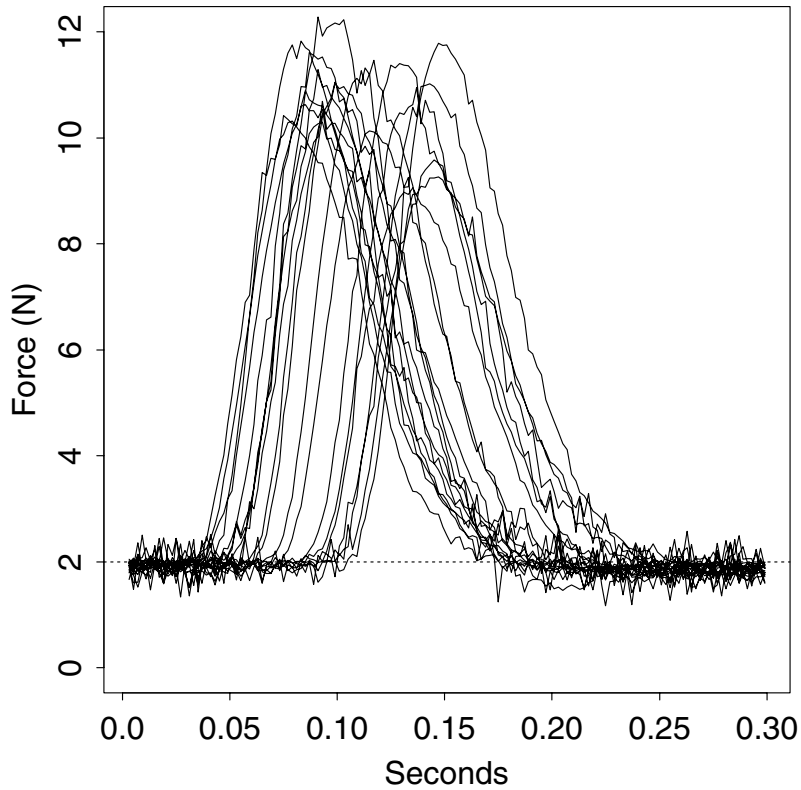
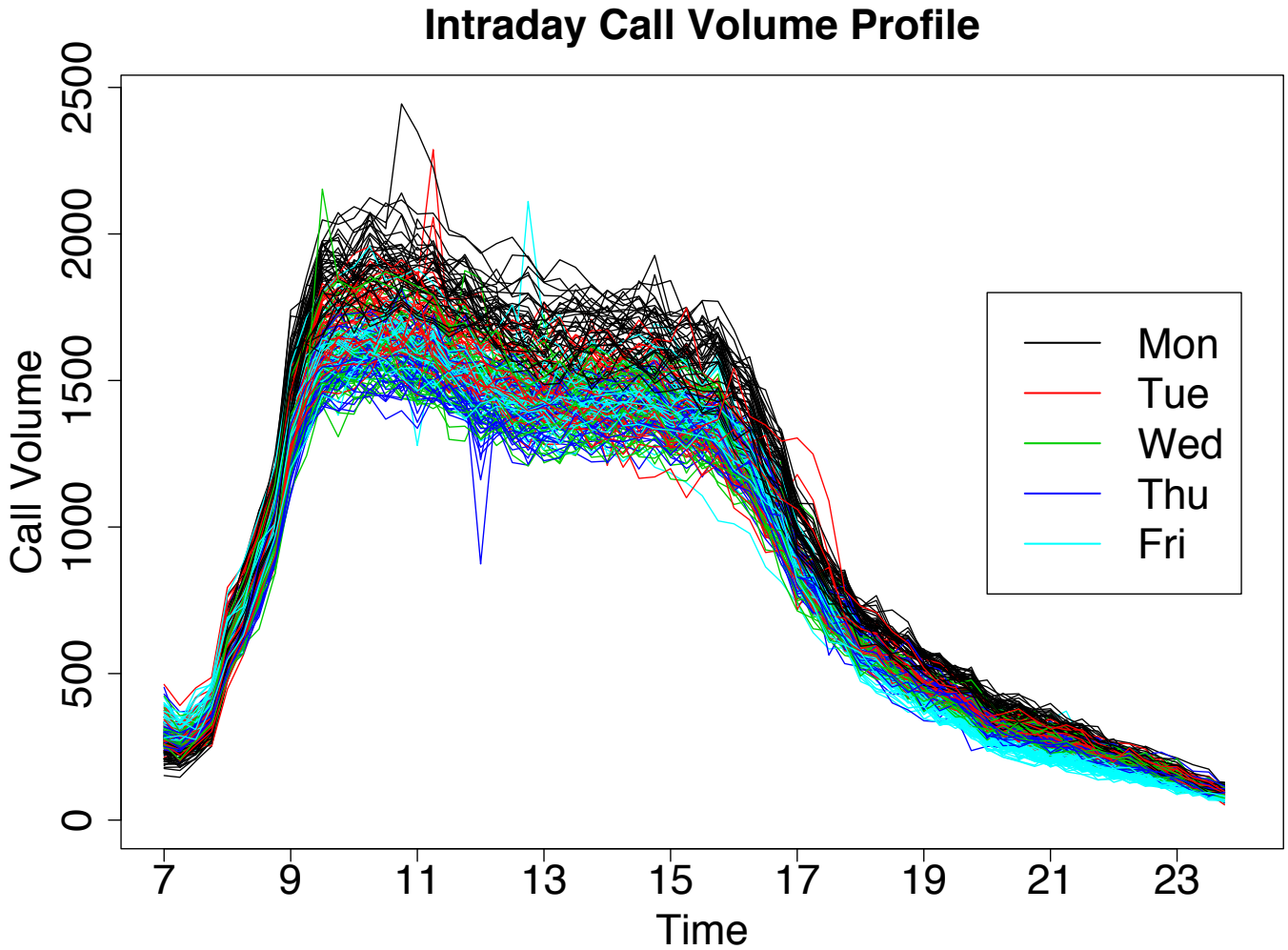
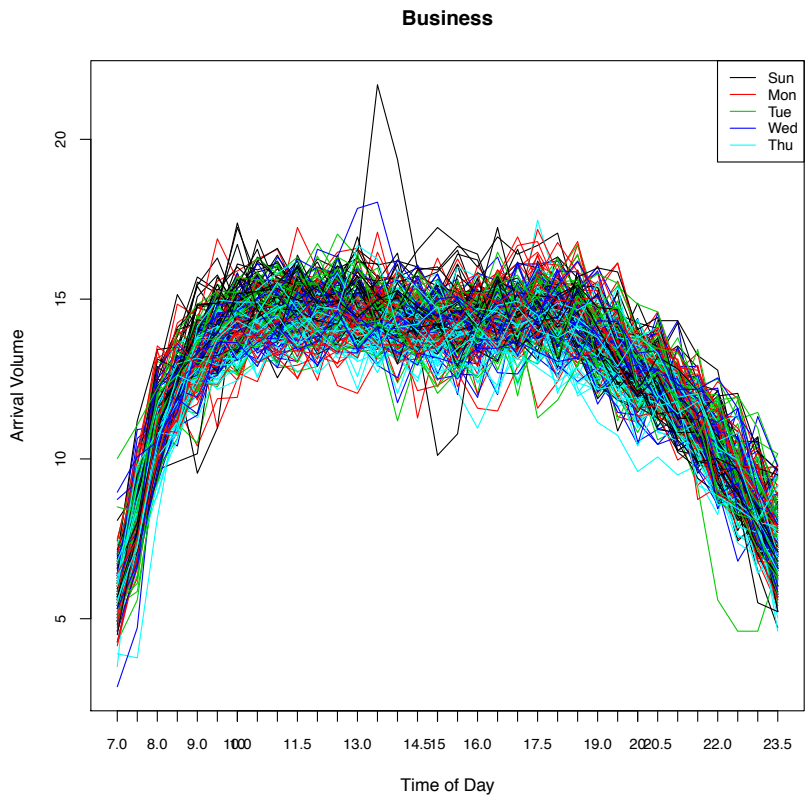


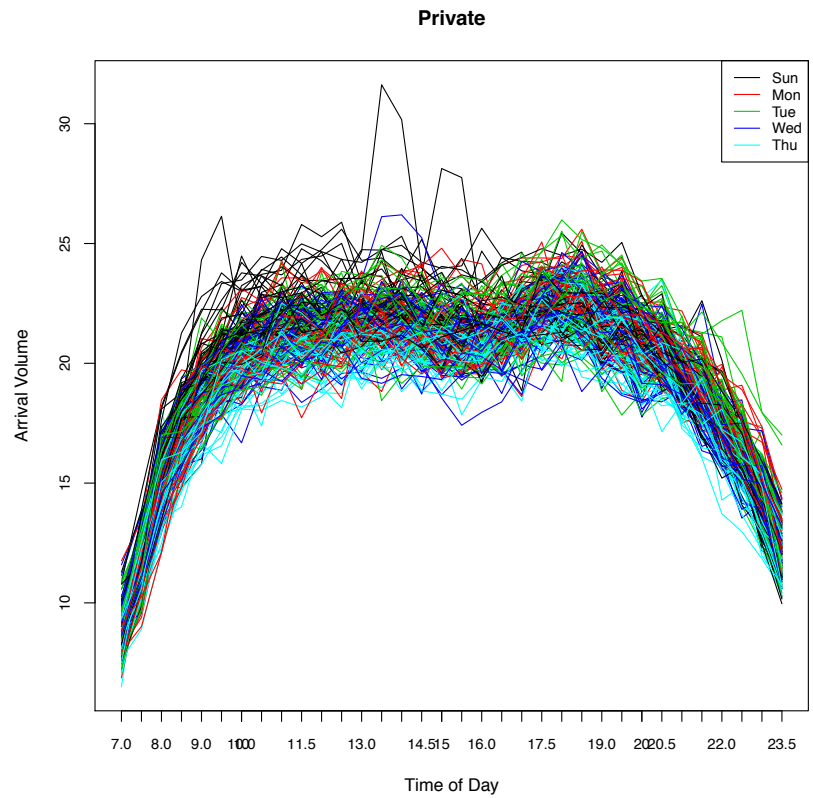
Figure 1.11. Twenty recordings of the force exerted by the thumb and forefinger where a constant background force of two newtons was maintained prior to a brief impulse targeted to reach 10 newtons. Force was sampled 500 times per second.

Example: intraday call volume profiles - time series of curves (Shen, 2008)

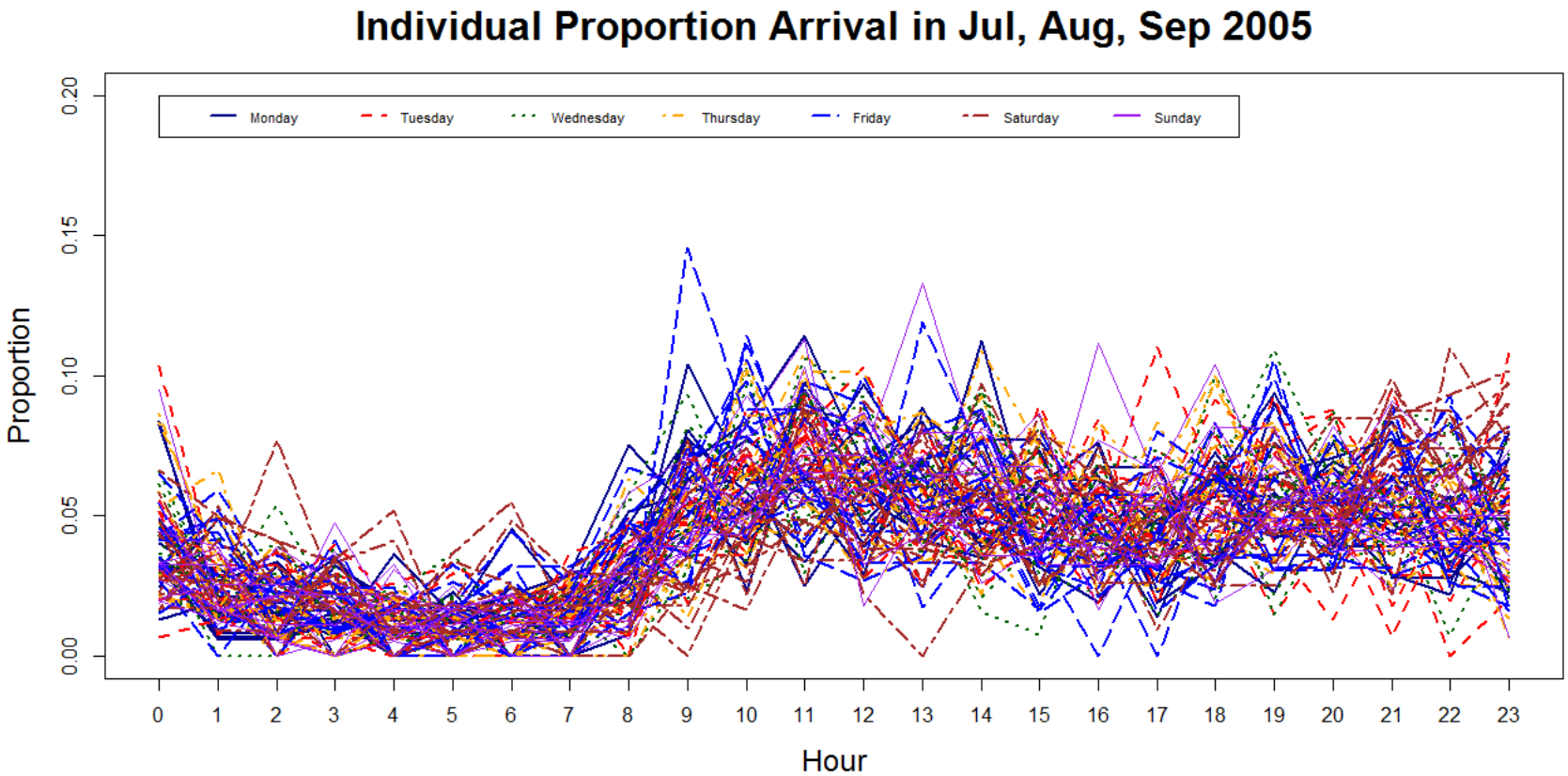


Example: intraday call volume profiles - two sets of time series of curves (Han Ye, 2016)

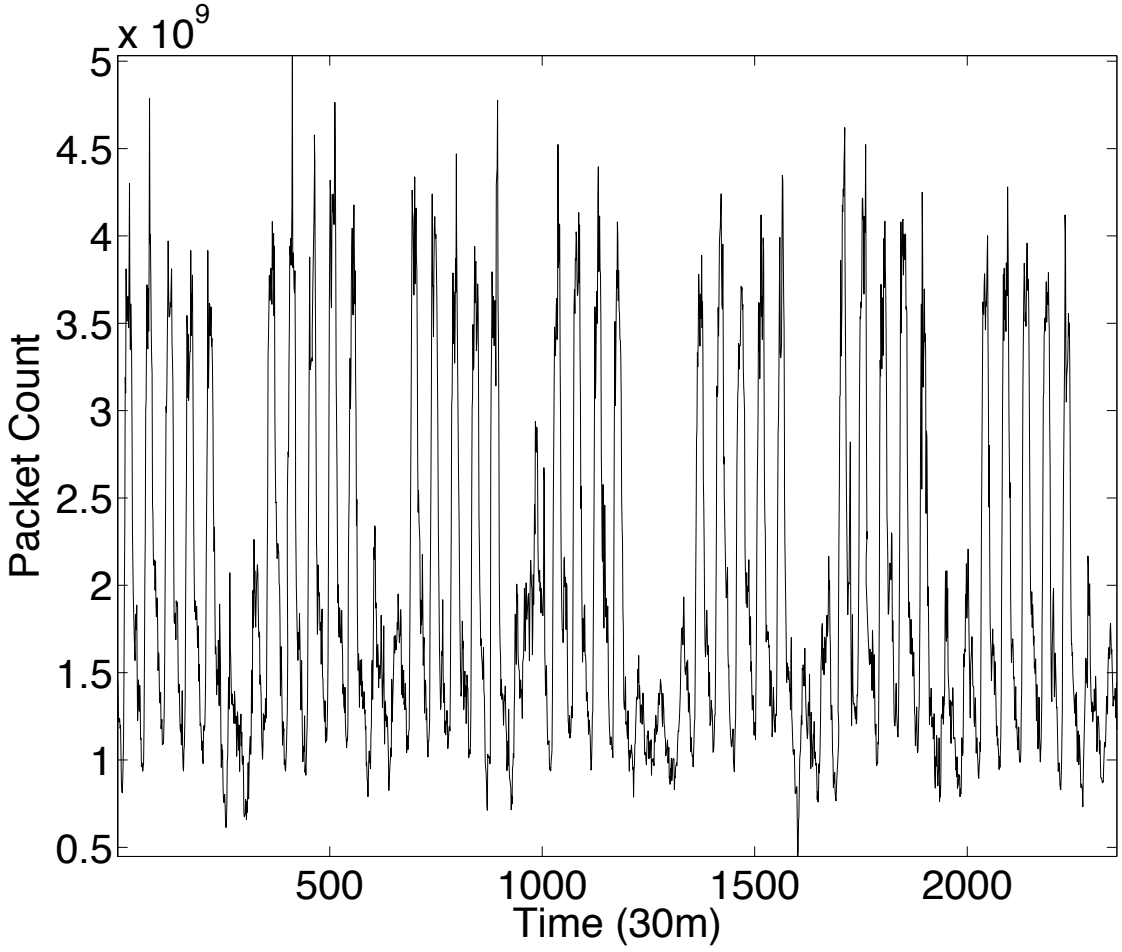




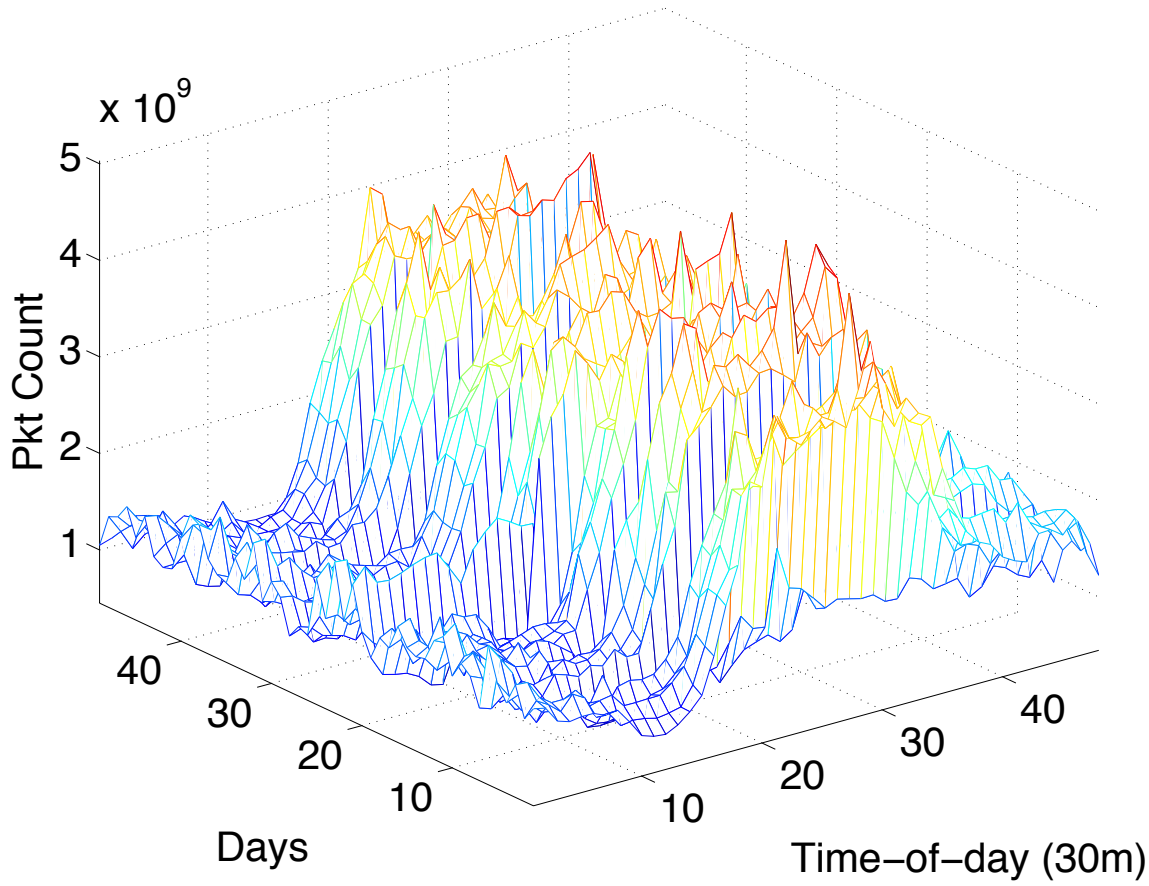
Example: Hospital patient arrivals - time series of curves (Dongqing Yu, 2016)



Example: Internet traffic volume - time series of curves (Lingsong Zhang, 2009)

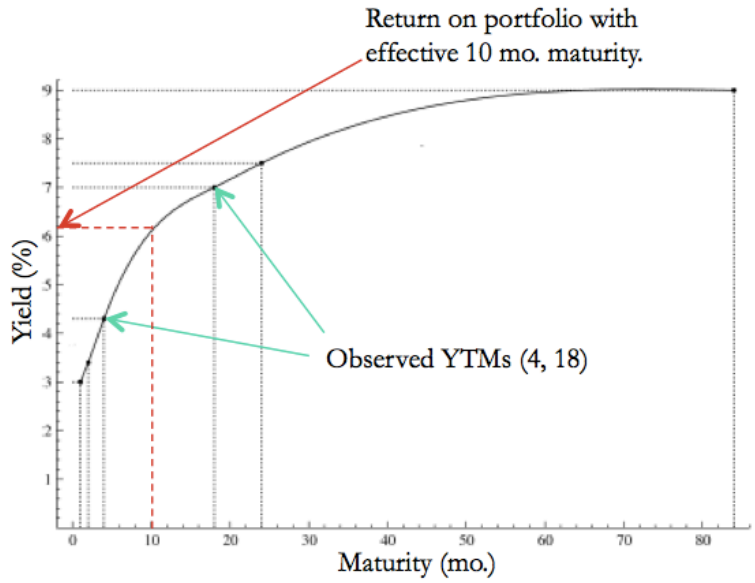


Example: Internet traffic volume - time series of curves



The Yield Curve: Functional Time Series

- Zero coupon bonds:
  - \$1,000 face /future value (FV)
  - Payable at maturity t months ahead
- Current day price: P < FV.
  - Invest P today, what rate of return generates FV t months from now?
  - Driven by market
- Yield to maturity (YTM): the rate of return that relates current price to future value

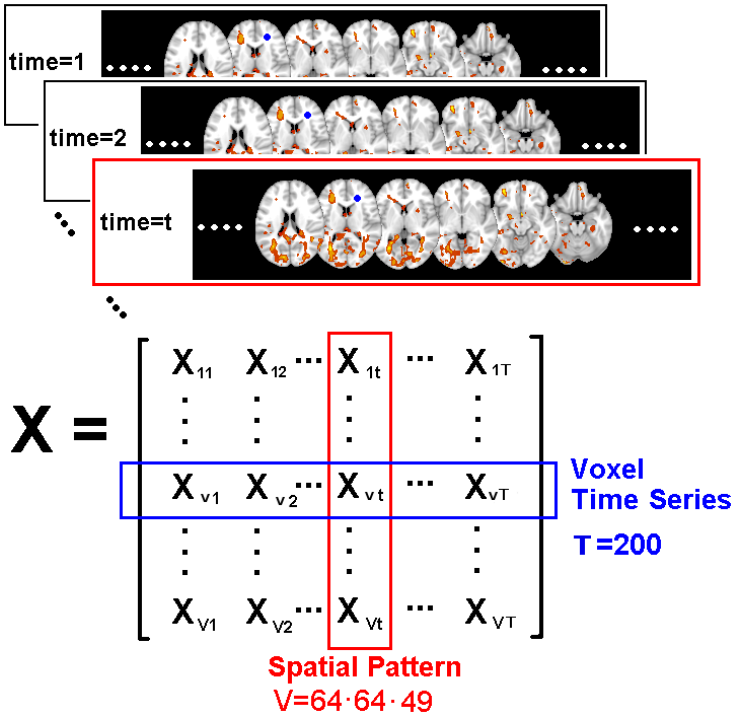


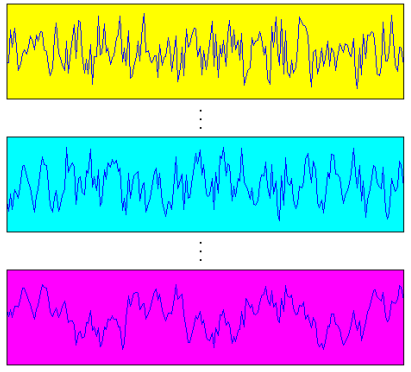
The Yield Curve $x_i(t)$:

- Continuous in maturity t
- Change over time, $i = 1, \ldots, n$

Example: Brain imaging (Seonjoo Lee, 2011)

Figure: MRI Scanner

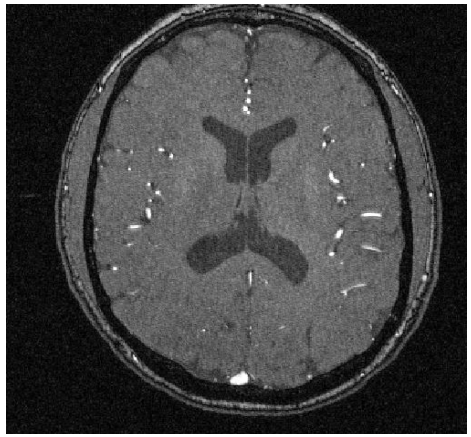




Collection of functions, spatially distributed, dependent, ...

Example: Blood vessel trees (Dan Shen, 2015)

Panel (A)



Panel (B)

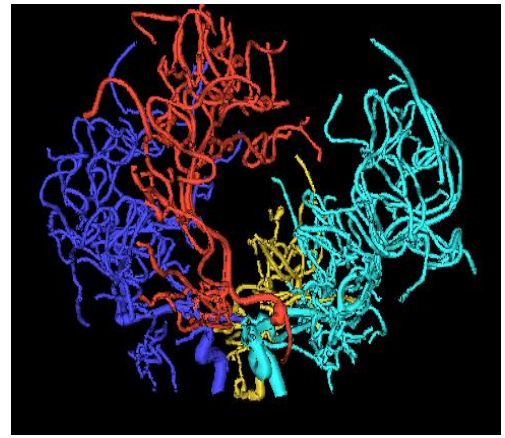


Figure A: Panel (A) is a single slice MRA image for one person with white regions indicating blood flow. The white regions in the 3 dimensional collection of MRA slices generate brain artery trees as shown in Panel (B). The colors in Panel (B) indicate the regions of the brain: Front (red), Back (gold), Left (cyan), Right (blue).

Collection of objects, non-Euclidean, ...

Goals of FDA

- Represent the data in ways that aid further analysis
  - dimension reduction, functional principal component analysis, ...
- Display the data so as to highlight features
  - smoothing, interpolation, registration, ...
- Study important sources of pattern and variation
  - center, variance, clusters, ...
- Explain variation in a dependent variable by using independent variable information
  - functional linear regression models, functional additive models, ...
- Compare sets of data about certain types of variation
  - curve clustering/classification, functional canonical correlation, ...
- Predict future curves
  - time series of curves, ...

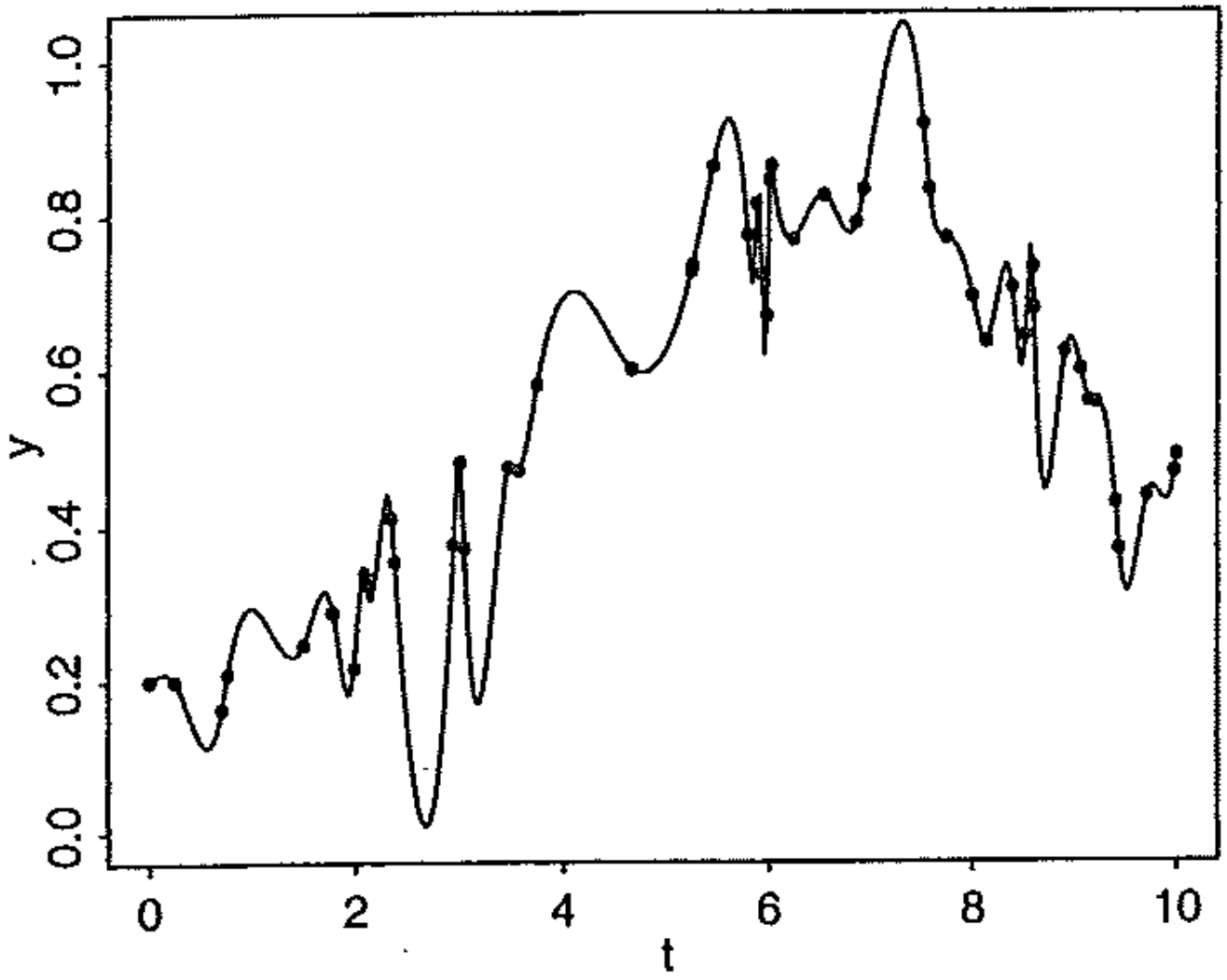
Data representation: smoothing and interpolation

- Curves: $f_1(t), \ldots, f_i(t), \ldots, f_n(t)$
- Discrete observations - y_{ij}, $j = 1, \ldots, n_i$:

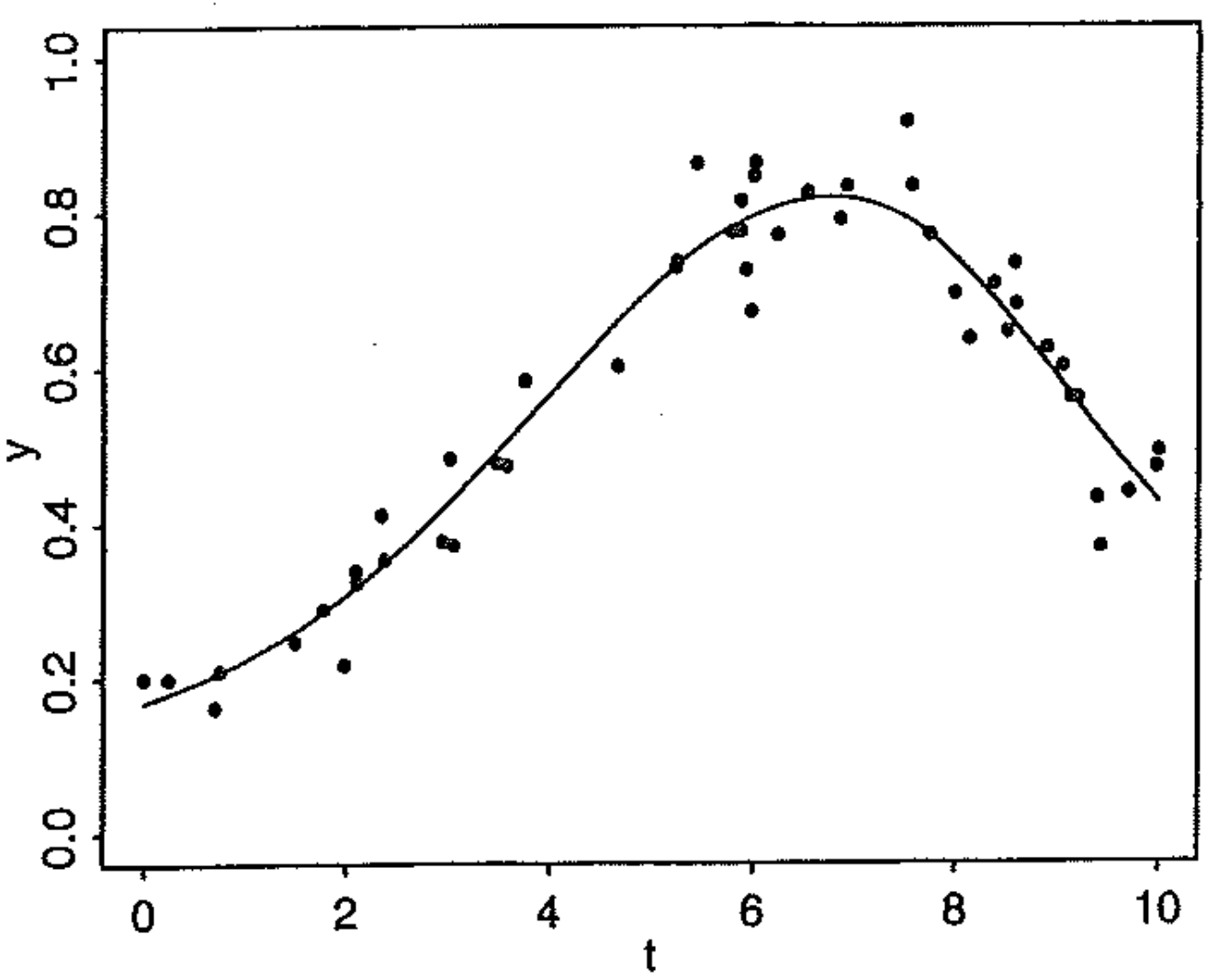
$$y_{ij} = f_i(x_{ij}) + \epsilon_{ij},$$

  where
  - x_{ij}: observation points, common or different for different i
  - n_i: number of observations for curve $f_i(t)$
- To uncover the underlying curve $f_i(t)$:
  - interpolation, when $\epsilon_{ij} = 0$
  - smoothing, when $\epsilon_{ij} \neq 0$

Example: Interpolation



Example: Smoothing



Outline

Regression, I

- Data: (x_i, Y_i), $i = 1, \ldots, n$
- Goal: dependence of Y_i on x_i
- Linear regression:

$$Y = \beta_0 + \beta_1 x + \epsilon$$

  - simplest model
  - elegant least squares theory
  - too rigid
- Generalized linear model
  - relax the response dist. to exponential family dist.
- Mixed-effects model
  - random parameters β_0/β_1

Regression, II

- Polynomial regression:

$$Y = \beta_0 + \beta_1 x + \ldots + \beta_k x^k + \epsilon$$

- Nonparametric regression, or *scatterplot smoother*:

$$Y = f(x) + \epsilon,$$

  where $f(x)$ is a smooth function

- Additive model for multiple predictors:

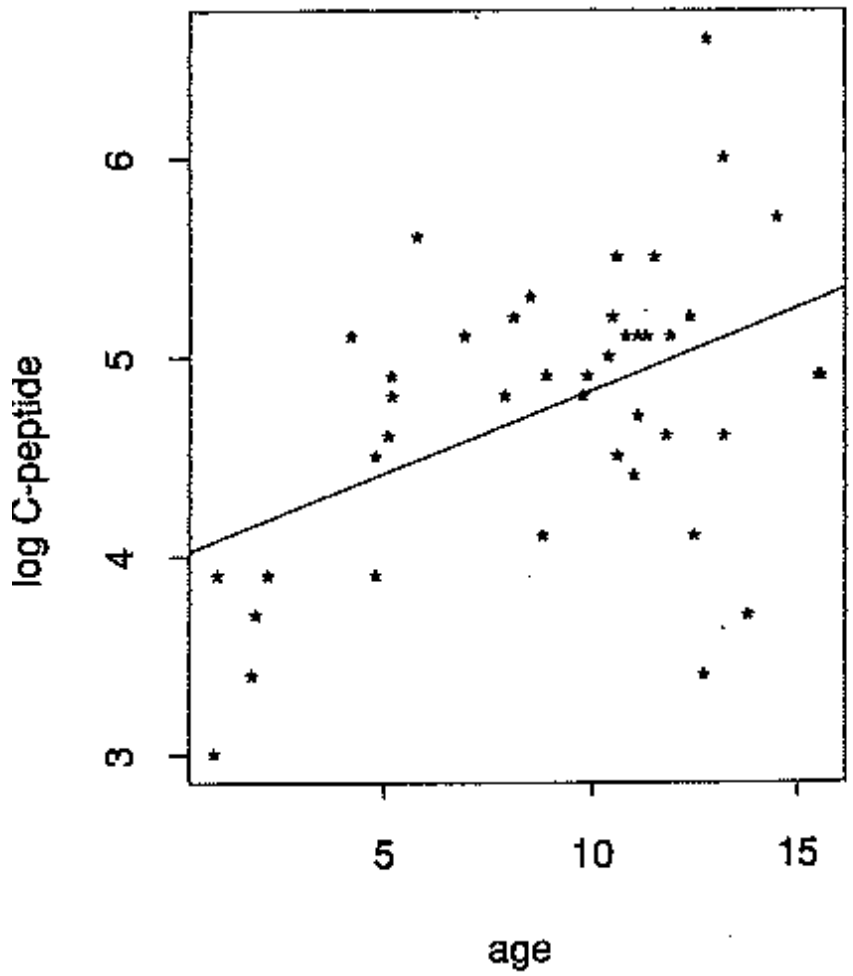
$$Y = f_1(x_1) + \ldots + f_k(x_k) + \epsilon$$

- Semi-parametric model

$$Y = Z\beta + f_1(x_1) + \ldots + f_k(x_k) + \epsilon$$

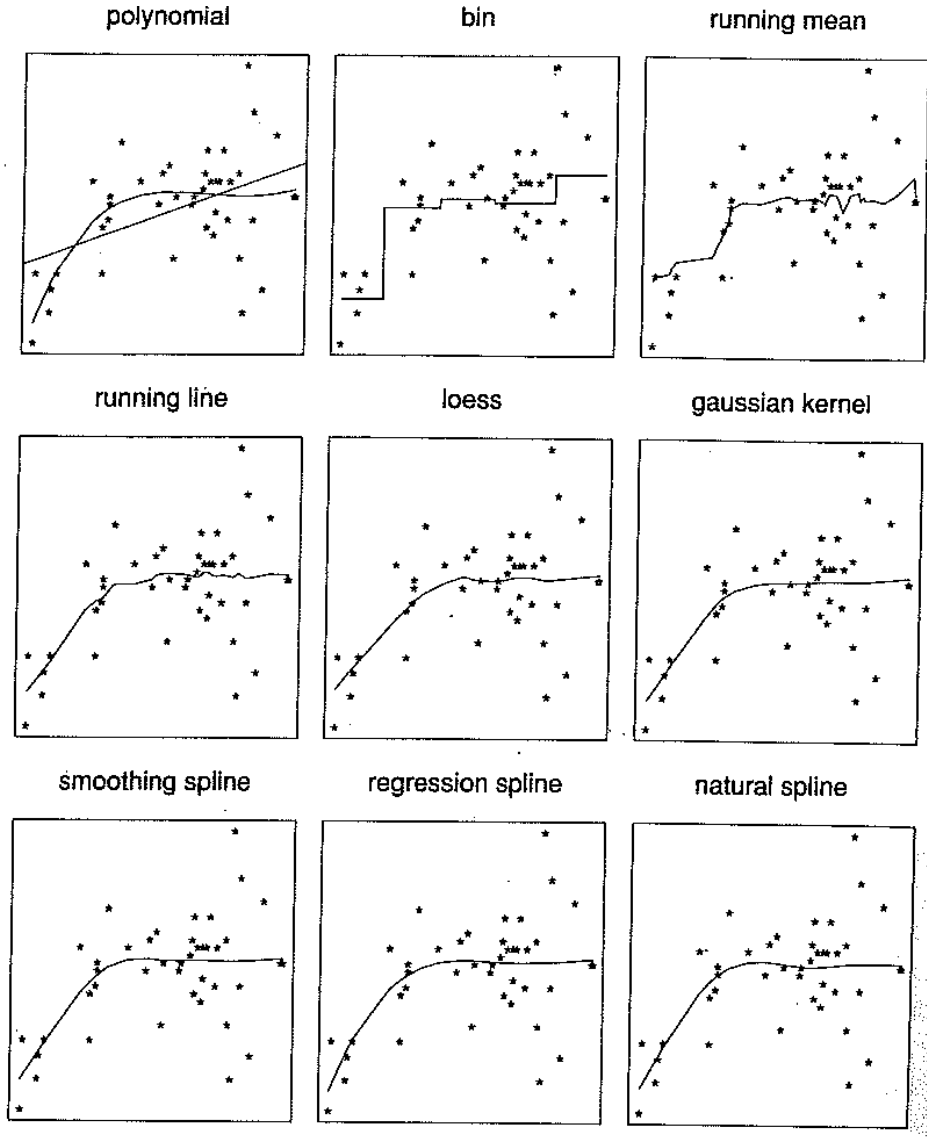
- Generalized additive model

Example: log(C-peptide) vs. age, linear regression



Diabetes patients: log(C-peptide) vs. the Age of diagnosis

Example: log(C-peptide) vs. age, various smoothers



Smoothing degree of freedom: 5

Linear: 2

Bin/quartic-poly: 5

(Cubic) regre. spline: one interior knot

Natural cubic spline: 3 interior knots and 2 endpoints constraints

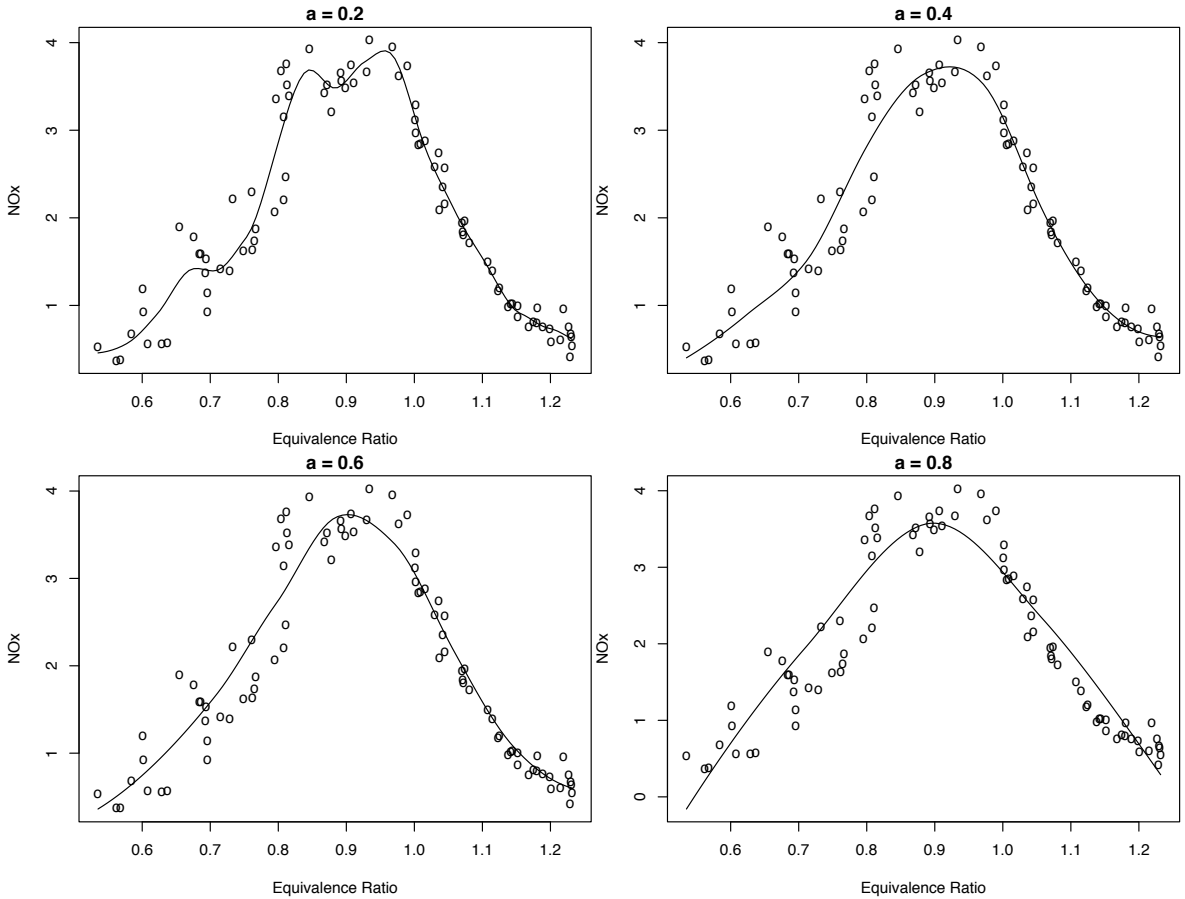
Smoothing Methods

- Basis expansion
  - polynomial splines (or regression splines)
  - Fourier basis expansion
  - wavelets
- Regularization (or roughness penalization)
  - smoothing splines
  - penalized splines
- Kernel regression
- Local polynomial regression

Smoothing Parameter

- Every smoother has some smoothing parameter to tune
- Its choice affects the final smooth
- Polynomial splines: order of the spline basis, number of knots
- Smoothing splines: order of the splines, penalization parameter
- Kernel regression: bandwidth
- Local polynomial regression: order of the polynomials, bandwidth

Local Polynomial Regression: nearest neighbor bandwidth



NOx: concentration of certain pollutants in the emissions

Eq. Ratio: richness of air/fuel mix in the engine

a: percent of obs used in a local window

Smoothing Parameter Selection

- subjective choice
- automatic selection
  - cross-validation
  - generalized cross-validation
  - 1st generation bandwidth selection
  - 2nd generation bandwidth selection
- scale-space approach
  - consider a range of smoothing parameters
  - SiZer (Chaudhuri and Marron, 2000)

Smoothing Context

- Nonparametric regression (our focus)
- Density function estimation
- Hazard function estimation
- ...

Polynomial/Regression Splines

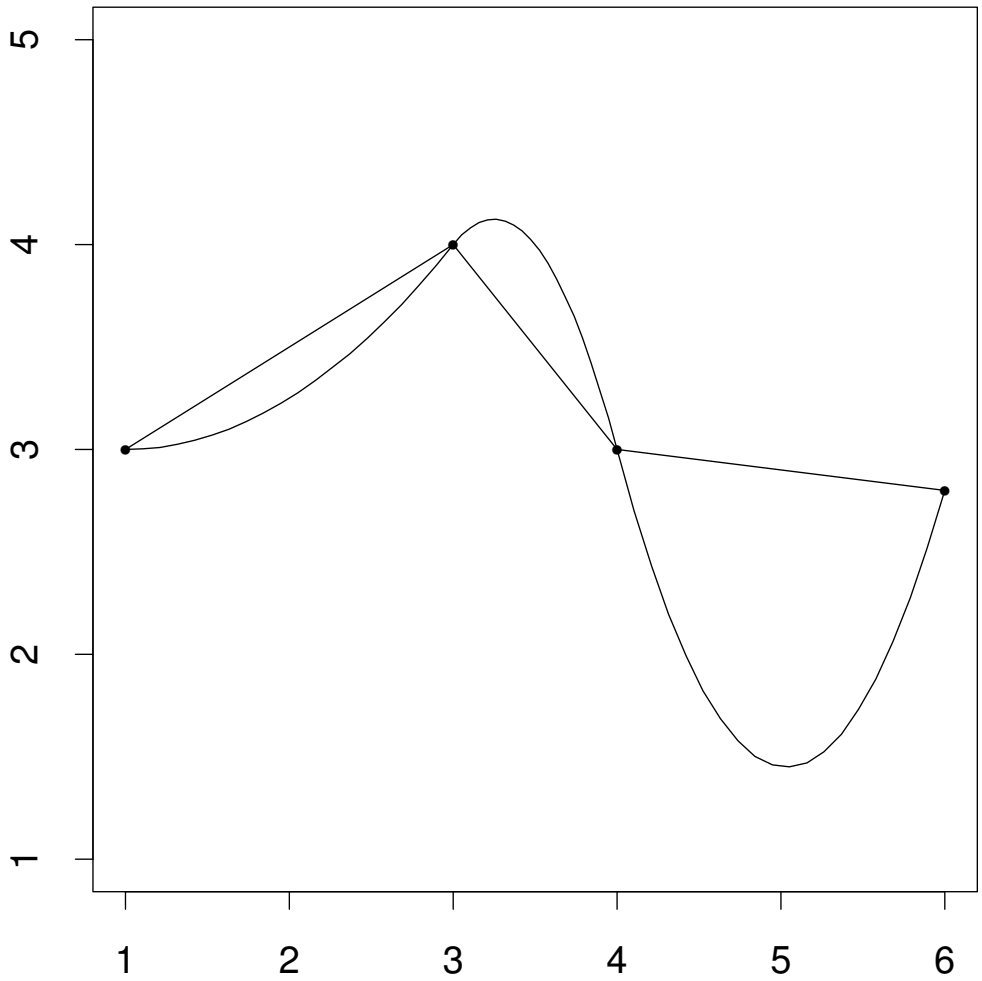
July 2016

Outline

Splines

- Spline: a curve formed by pasting several smaller pieces of curves together
- Named after the draftman's tool - spline, a thin flexible rod made of metal, plastic or wood held in place by weights for drawing smooth curves that go through certain points
- To define a spline, need
  - knots, locations where the joining or pasting occurs
  - smaller curves, used to form the larger one

Examples: linear and quadratic splines with knots at 1,3,4,6



Polynomials and Piecewise Polynomials

- A polynomial of order k:

$$p(x) = a_0 + a_1 x + \ldots + a_{k-1} x^{k-1},$$

  where k: positive integer
  - k coefficients
  - degree $k - 1$

- Consider $t_1 < t_1 < \cdots < t_m$.

- A piecewise polynomial of order k with knots $t_1, \ldots, t_m$: function whose restriction to each of the $m - 1$ intervals $[t_1, t_2)$, ..., $[t_{m-1}, t_m)$ is a polynomial of order k.

Polynomial Splines

- A (polynomial) spline function of order k having distinct knots $t_1, t_2, \ldots, t_m$:
  - piecewise polynomial of order k with the above knots
  - for $k \geq 2$, $k-2$ times continuously differentiable
- Examples:
  - spline of order one: piecewise constant
  - linear spline ($k = 2$): continuous piecewise linear
  - quadratic spline ($k = 3$): continuously differentiable piecewise quadratic
  - cubic spline ($k = 4$): piece cubic with continuous second derivatives (common choice in practice)

Splines: truncated power basis

- A spline function of order k can be written as

$$s(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{k-1} x^{k-1} + \sum_{i=1}^{m} b_i (x - t_i)_+^{k-1}, \quad k \geq 2,$$

  where $(\cdot)_+$: nonnegative part of the argument.

- $m + k$ coefficients: $a_0, \ldots, a_{k-1}$ and $b_1, \ldots, b_m$

- Truncated power basis:

$$1, x, x^2, \ldots, x^{k-1}, (x - t_1)_+^{k-1}, \ldots, (x - t_m)_+^{k-1}.$$

  - ill-conditioned, highly collinear

  - design matrix illustration

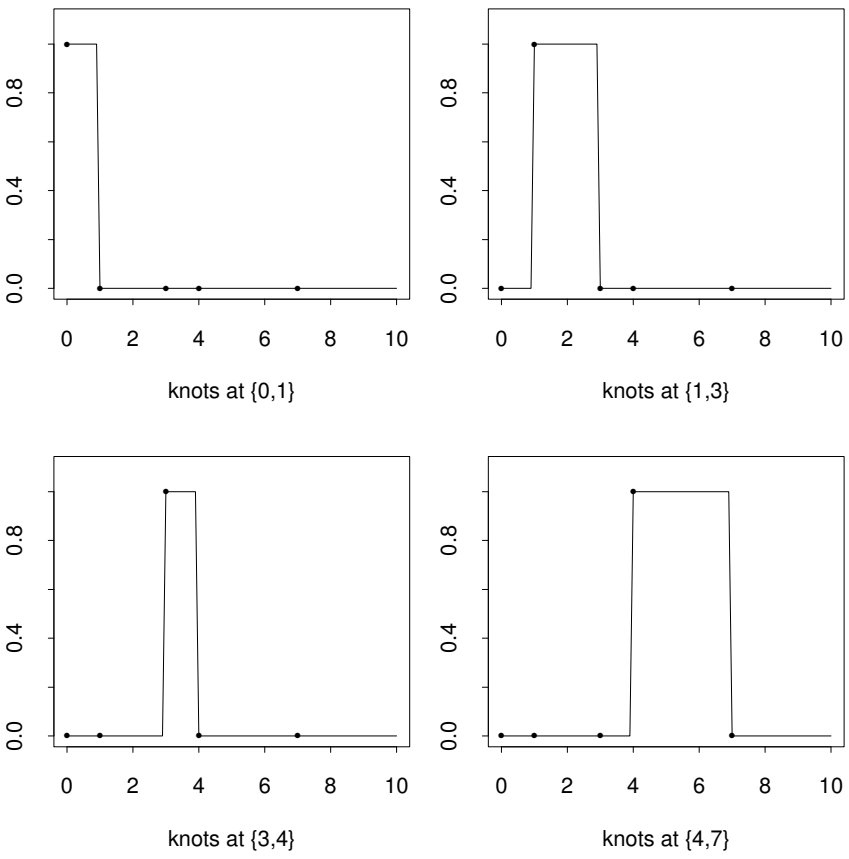
Splines: B-splines basis

- *B* stands for *basis*
- Alternative basis for spline functions
  - excellent numerical and theoretical properties
- The recursive definition (Kincaid and Cheney, 1996)
  - simpler
  - easier to understand most important properties of B-splines
- Consider the doubly infinite knot sequence

$$\cdots < t_{-2} < t_{-1} < t_0 < t_1 < t_2 < \cdots .$$

B-splines of order 1

$$B_i^1(x) = 1_{[t_i, t_{i+1})}(x) = \begin{cases} 1, & t_i \leq x < t_{i+1}, \\ 0, & \text{otherwise.} \end{cases}$$



Knots: 0, 1, 3, 4, 7

B-splines of order 1: Properties

- The support of $B_i^1(x)$ is $[t_i, t_{i+1})$;
  - i.e. two knots are needed to construct a B-spline of order one;
- $B_i^1(x)$ is nonnegative for all x and i;
- $B_i^1(x)$ is continuous from the right on the entire line;
- $\sum_i B_i^1(x) = 1$ for all x.

B-splines of order 2

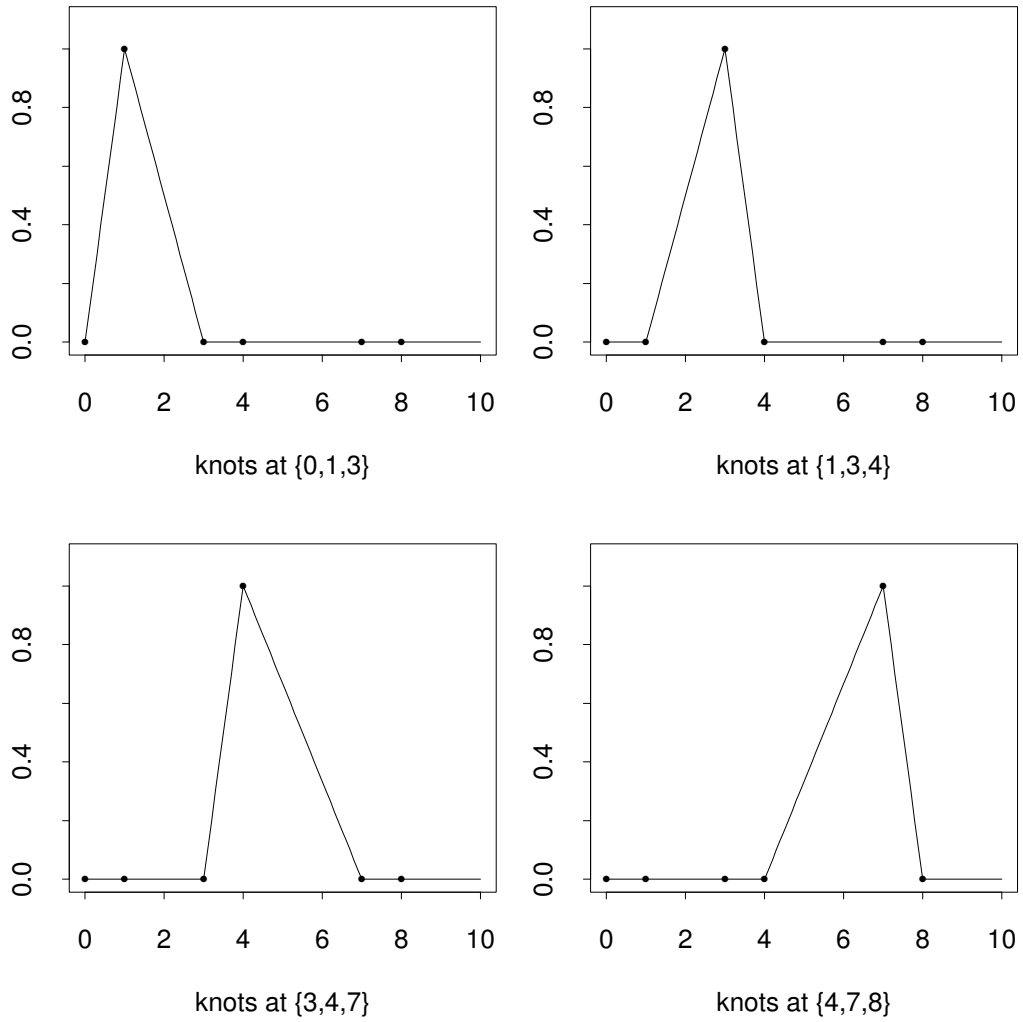
Recursive Definition:

$$B_i^2(x) = \frac{x - t_i}{t_{i+1} - t_i} \cdot B_i^1(x) + \frac{t_{i+2} - x}{t_{i+2} - t_{i+1}} \cdot B_{i+1}^1(x),$$

which is equivalent to

$$B_i^2(x) = \begin{cases} \frac{x - t_i}{t_{i+1} - t_i}, & t_i \leq x < t_{i+1}, \\ \frac{t_{i+2} - x}{t_{i+2} - t_{i+1}}, & t_{i+1} \leq x < t_{i+2}, \\ 0, & \text{otherwise.} \end{cases}$$

B-splines of order 2: Plot with knots at 0, 1, 3, 4, 7, 8



B-splines of order 2: Properties

- The support of $B_i^2(x)$ is $[t_i, t_{i+2})$;
  - i.e. three knots are involved;
- $B_i^2(x)$ is nonnegative for all x and i;
- $B_i^2(x)$ is continuous on the entire line;
- $\sum_i B_i^2(x) = 1$ for all x;
- $B_i^2(x)$ is unimodal.

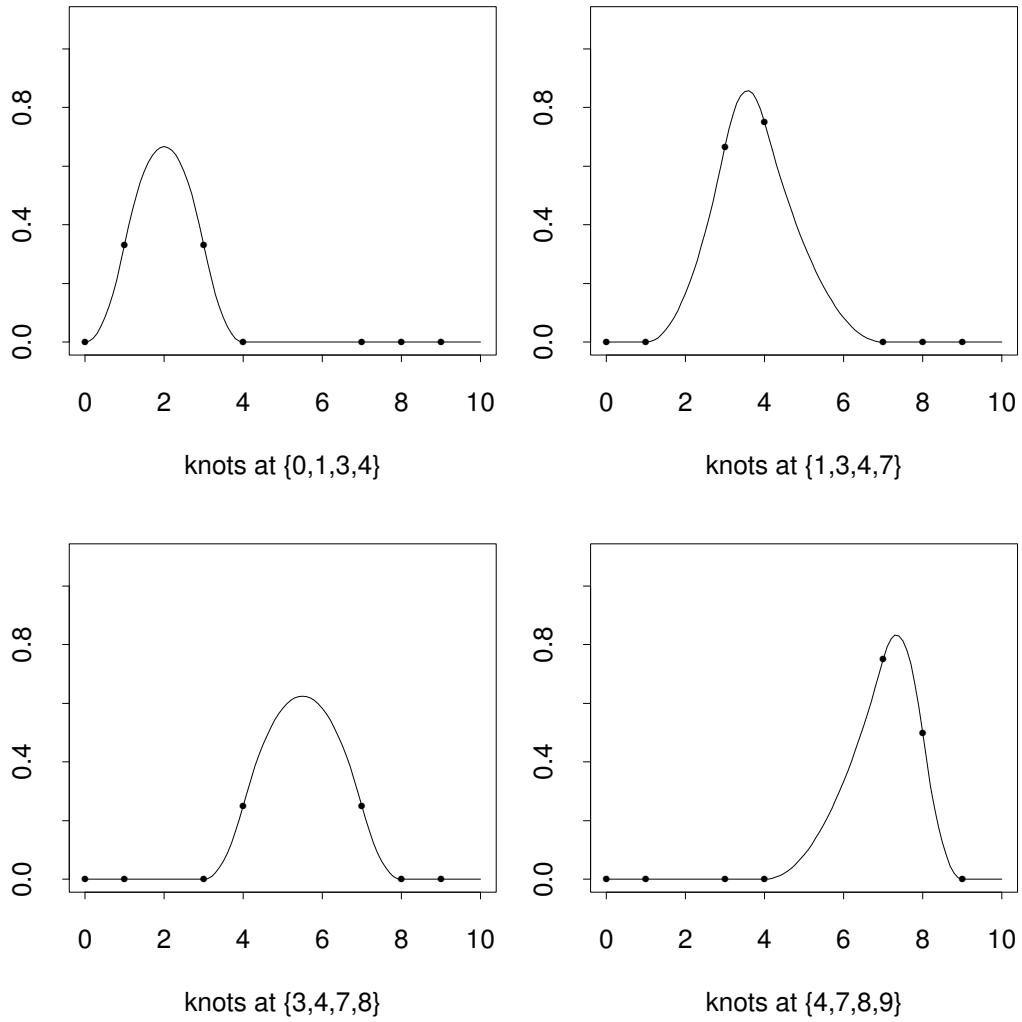
B-splines of order k

Recursive Definition:

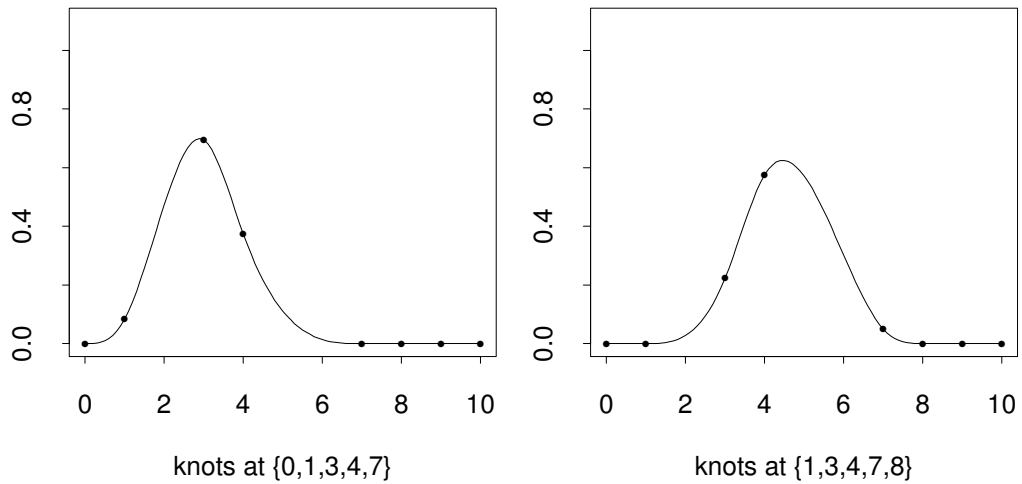
$$B_i^k(x) = \frac{x - t_i}{t_{i+k-1} - t_i} \cdot B_i^{k-1}(x) + \frac{t_{i+k} - x}{t_{i+k} - t_{i+1}} \cdot B_{i+1}^{k-1}(x).$$

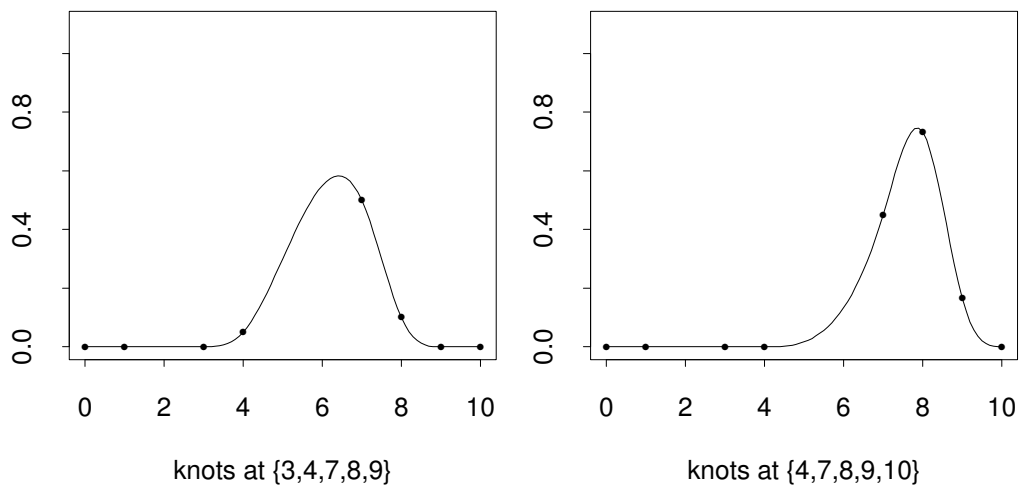
- $k = 3$: quadratic B-splines
- $k = 4$: cubic B-splines

B-splines of order 3: Plot with knots at 0, 1, 3, 4, 7, 8, 9



B-splines of order 4: Plot with knots at 0, 1, 3, 4, 7, 8, 9, 10





B-splines of order k: Properties

- Support of B-splines:
  - the support increases with order
  - $B_i^1(x) = 1_{[t_i, t_{i+1})}(x)$
  - $B_i^k(x) > 0 \text{ for } x \in (t_i, t_{i+k}), \quad k > 1$
  - for a given x, finite B-splines are nonzero at x
  - for example, for $[t_i, t_{i+1})$, only k B-splines might be nonzero:

$$B_{i-k+1}^k, B_{i-k+2}^k, \ldots, B_i^k.$$

- $\sum_i B_i^k(x) = 1$ for all x;
- B-splines of order two or higher are unimodal;

B-splines of order k: Properties

- Derivatives of B-splines:

$$\frac{d}{dx}B_i^k(x) = \frac{k-1}{t_{i+k-1}-t_i}\cdot B_i^{k-1}(x) - \frac{k-1}{t_{i+k}-t_{i+1}}\cdot B_{i+1}^{k-1}(x).$$

- The set $\{B_{i-k+1}^k, B_{i-k+2}^k, \ldots, B_i^k\}$ of B-splines is linearly independent on (t_i, t_{i+1});

- The set $\{B_{-k+1}^k, B_{-k+2}^k, \ldots, B_m^k\}$ of B-splines is linearly independent on (t_0, t_{m+1});

- On the interval $[t_0, t_{m+1}]$, the set $\{B_{-k+1}^k, B_{-k+2}^k, \ldots, B_m^k\}$ of B-splines forms a basis for all order k splines having knots $t_1, t_2, \ldots, t_m$.

Why B-splines?

- their support is finite, which provides a stable numerical method
- in regression setting, the B-spline representation - a banded design matrix, more efficient for computation
- on the other hand, the truncated power basis - non local, ill-conditioned, highly collinear design matrix, unstable
- B-splines: a useful and powerful tool in spline theory
- R: bs; Matlab: bspline

References: de Boor (1978), Schumaker (1981)

Regression Splines

- Data: (x_i, Y_i), $i = 1, \ldots, n$
- Goal: dependence of Y on x
- Two end points of x: t_0, t_{m+1}; interior knots: $t_1, \ldots, t_m$
- Nonparametric regression, or *scatterplot smoother*:

$$Y_i = f(x_i) + \epsilon_i,$$

where $f(\cdot)$ is a smooth function

- Nice approximation property of splines:
exist a spline function $f^*(x)$ on $[t_0, t_{m+1}]$ such that

$$\sup_{x \in [t_0, t_{m+1}]} |f(x) - f^*(x)| \to 0$$

as the number of the knots of the spline tends to infinity.

Regression Splines

- The approximation property suggests that there exists a set of B-splines $B_j(x)$ and constants β_j, $j = 1, \ldots, J$, such that

$$f(x) \approx f^*(x) = \sum_{j=1}^{J} \beta_j B_j(x).$$

- The coefficients $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_J)^T$ can be estimated via least squares method:

$$\hat{\boldsymbol{\beta}} = \text{argmin}_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left(Y_i - \sum_{j=1}^{J} \beta_j B_j(x_i) \right)^2.$$

Regression Splines: comments

- the spline fit of a function is totally characterized by the finite number of coefficients in the basis expansion
- consider m interior knots, order k splines: the number of parameters is

$$(m+1)k - m(k-1) = m + k.$$

- provides an explicit expression of the fitted model - a global smoothing method, yet the B-spline basis is local
- given order and knot sequence, the B-spline basis can be obtained
- the observations are only needed to estimate the coefficients

Regression Splines

- The number of B-spline bases J is determined by the order of the B-splines k and the number of interior knots m, in that
$$J = k + m.$$
- Usually cubic B-splines are used, i.e. $k = 4$
- Some choice needs to be made regarding the number of knots as well as their locations
  - fixed knots:
    - only need to choose m
    - equal spaced knots
    - quantile knots
  - adaptive (or free) knots
    - need to choose both m and the knot locations $\mathbf{t}$

Fixed knots splines

- For a fixed m, essentially a linear regression model with the number of parameters being $m + k$
- Can use model selection methods for linear models to choose m
  - Akaike Information Criterion (Akaike, 1974)

$$\mathrm{AIC} = \log\left(\frac{\mathrm{RSS}}{n}\right) + 2 \cdot \frac{m+k}{n},$$

    where RSS is the minimized residual sum of squares

  - Bayesian Information Criterion (Schwarz, 1978)

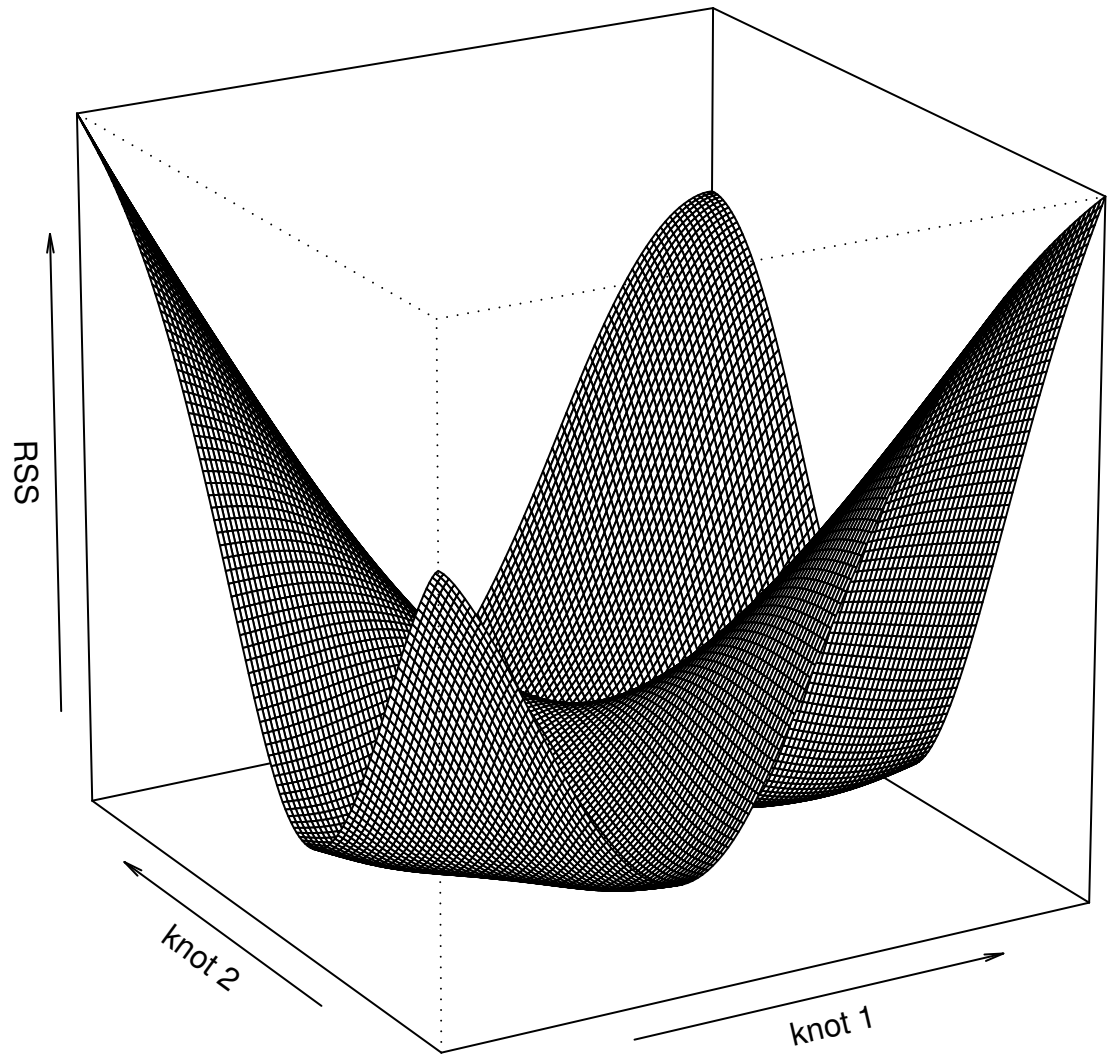
$$\mathrm{BIC} = \log\left(\frac{\mathrm{RSS}}{n}\right) + \log(n) \cdot \frac{m+k}{n}$$

  - BIC tends to choose smaller models
- Examples: functional coefficient models, nonparametric regression with lognormal errors

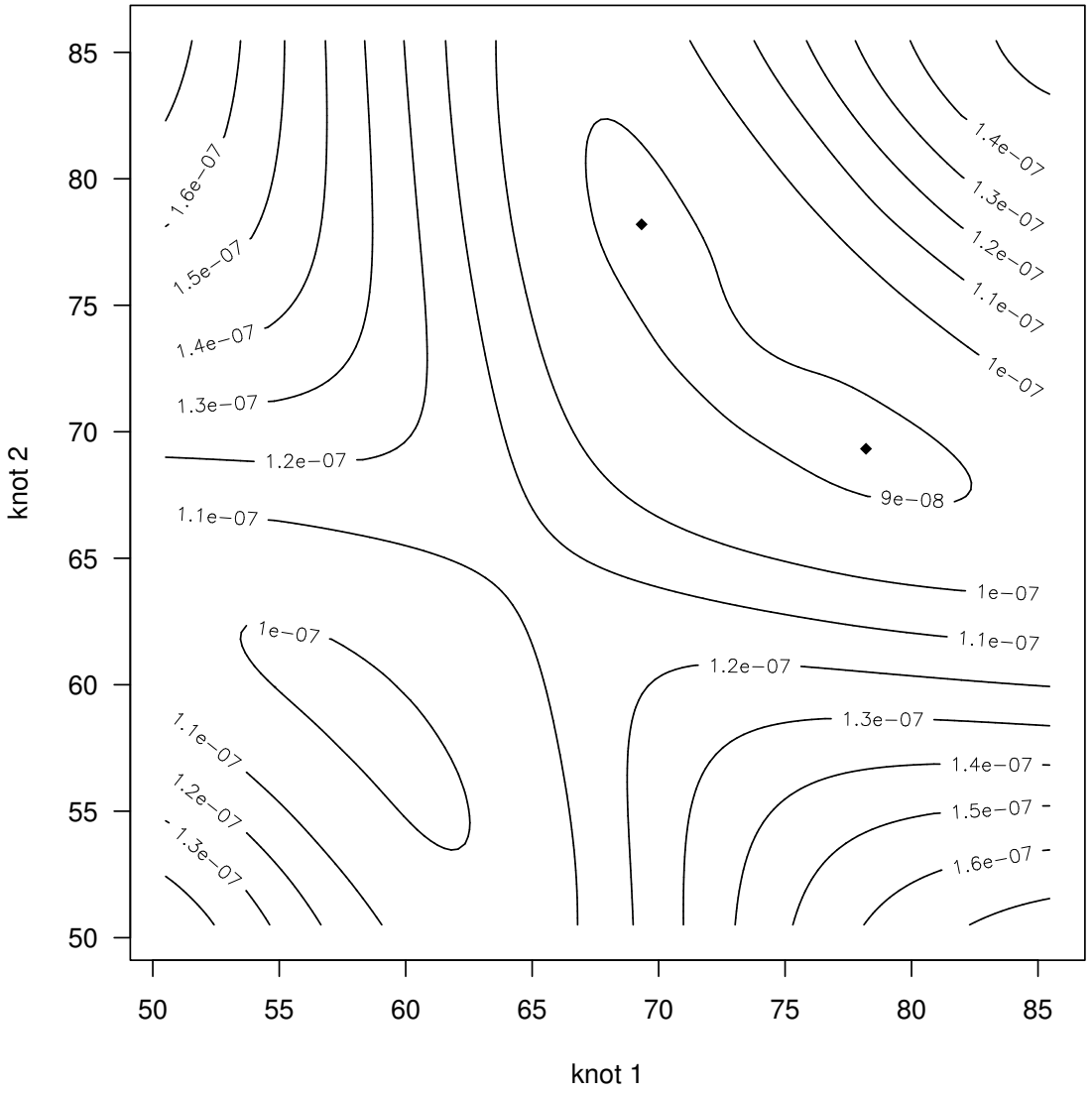
Free knots splines

- For each fixed model dimension (or equivalently m), need to look for the knot sequence **t** that minimizes the RSS.
- RSS depends nonlinearly on **t** and is known to have local minima.
  - Example: RSS(t_1, t_2) (next slides)
- Standard optimization techniques have trouble
- Regularization techniques (later)
- Stepwise addition/deletion (later)

Example: RSS(t_1, t_2) - perspective plot



Example: RSS(t_1, t_2) - contour plot



Comments:

1. Symmetric along $t_1 = t_2$

2. Black points: 69.3, 78.2

3. Local minima: 56.7, 60.2

Autoregressive time series models

- Linear autoregressive model of order p, AR(p):

$$x_t = a_1 x_{t-1} + \ldots + a_p x_{t-p} + \epsilon_t$$

- Exponential autoregressive model, EXPAR:

$$x_t = \sum_{i=1}^{p} \{\alpha_i + (\beta_i + \gamma_i x_{t-d}) \exp(-\theta_i x_{t-d}^2)\} x_{t-i} + \epsilon_t, \quad \theta_i \geq 0$$

Autoregressive time series models

- Threshold autoregressive model, TAR:

$$x_t = \phi_1^{(i)} x_{t-1} + \cdots + \phi_p^{(i)} x_{t-p} + \epsilon_t^{(i)} \quad \text{if} \quad x_{t-d} \in \Omega_i$$

  where $\{\Omega_i\}$ form a partition of the real line.

- Functional-coefficient autoregressive model, FAR:

$$x_t = a_1(\mathbf{X}_{t-1}^*) x_{t-1} + \cdots + a_p(\mathbf{X}_{t-1}^*) x_{t-p} + \epsilon_t$$

  where $\mathbf{X}_{t-1}^* = (x_{t-i_1}, \ldots, x_{t-i_k})^T$.

Functional varying coefficient models for time series

- $\{Y_t, \mathbf{X}_t, \mathbf{U}_t\}_{-\infty}^{\infty}$ jointly strictly stationary
- $\mathbf{X}_t = (X_{t1}, \ldots, X_{td})$ in $\mathbb{R}^d$
- $\mathbf{U}_t$ in $\mathbb{R}^k$, k small
- $\mathbf{X}_t$ and $\mathbf{U}_t$ can be lagged values of Y_t
- Regression function:

$$f(\mathbf{x}, \mathbf{u}) = E(Y_t | \mathbf{X}_t = \mathbf{x}, \mathbf{U}_t = \mathbf{u}) = a_1(\mathbf{u})x_1 + \cdots + a_d(\mathbf{u})x_d$$

Spline approximation and least squares

- Use polynomial splines to approximate the smooth functions
- $a_j(u) \approx \sum_{s=1}^{K_j} \beta_{js}^* B_{js}(u)$.
- $f(\mathbf{x}, u) \approx \sum_{j=1}^{d} \left\{ \sum_{s=1}^{K_j} \beta_{js}^* B_{js}(u) \right\} x_j$
- minimize

$$\ell(\beta) = \sum_{i=1}^{n} \left(Y_i - \sum_{j=1}^{d} \left\{ \sum_{s=1}^{K_j} \beta_{js} B_{js}(U_i) \right\} X_{ij} \right)^2$$

- $\widehat{a}_j(u) = \sum_{s=1}^{K_j} \widehat{\beta}_{js} B_{js}(u)$

Model selection

- $\mathrm{AIC} = \log(\mathrm{RSS}/n) + 2 * p/n$, where $p = \sum_j K_j$
- $\mathrm{AIC_C} = \mathrm{AIC} + \frac{2(p+1)(p+2)}{n(n-p-2)}$
- $\mathrm{BIC} = \log(\mathrm{RSS}/n) + \log(n) * p/n$
- Modified Cross-validation (MCV)
  - m and Q two positive integers, $n > mQ$
  - Use Q sub-series of lengths $n - qm (q = 1, ..., Q)$ to estimate the unknown coefficient functions a_j
    - compute the one-step forecasting errors of the next section of the time series of length m based on the estimated models
    - AMS_q: average mean squared prediction error
  - $\mathrm{AMS} = \sum_{q=1}^{Q} \mathrm{AMS}_q$

Threshold variable and significant lags

- consider the functional autoregressive models
  $Y_t = \sum_{j \in S_d} a_j(Y_{t-d}) Y_{t-j} + \epsilon_t,\ 1 \le d \le p_{\max},$
  $S_d \subset \{1, \cdots, p_{\max}\}$
- for a given candidate threshold lag d, decide on an optimal subset S_d^* of significant lags
  - stepwise addition followed by stepwise deletion
- the final model is determined by the pair $\{d, S_d^*\}$ that produces the smallest AIC (or $\mathrm{AIC_C}$ or BIC or MCV)

Stepwise procedure for choosing S_d^*

- Addition: add one significant lag at a time, choosing among all candidate lags not yet selected in the model by minimizing the MSE (mean square error)
- Stop if the number of lags selected equals a pre-specified number $q_{\max} \leq p_{\max}$
- Deletion: delete one lag at a time from the collection of lags selected in the addition stage, also by minimizing the MSE until there is no lag variable left in the model
- Addition-Deletion results in a sequence of subsets of lag indices, the one that minimizes AIC is chosen as S_d^*

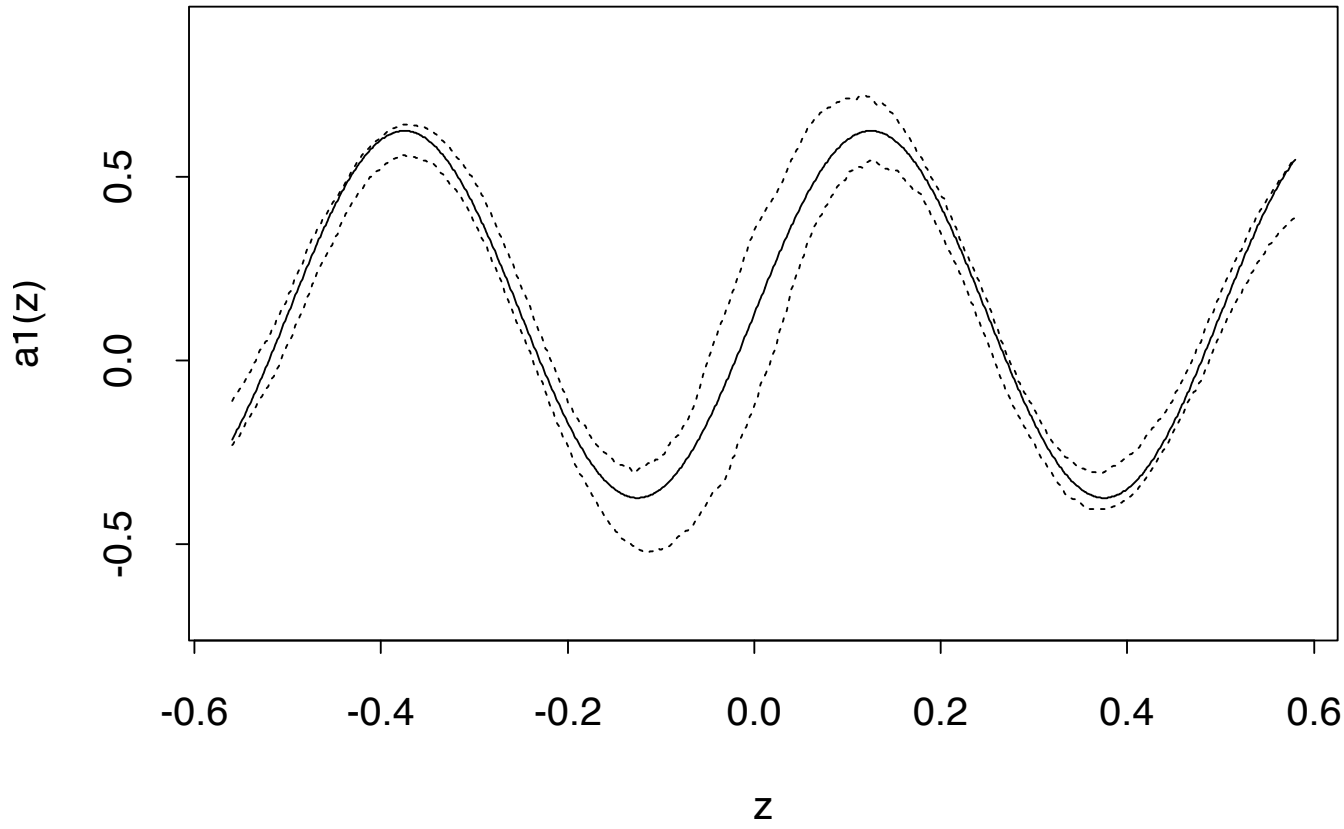
A Simulation Study

- $Y_t = a_1(Y_{t-1})Y_{t-1} + a_2(Y_{t-1})Y_{t-2} + \epsilon_t$
  - $a_1(u) = \frac{1}{8} + \frac{1}{2}\sin(4\pi u)$
  - $a_2(u) = -\frac{1}{8} - (\frac{1}{2} + u)e^{-4u^2}$
  - $\{\epsilon_t\}$ are i.i.d. $N(0, 0.2^2)$
- $a_1(u)$ and $a_2(u)$ are of different smoothness
- the simulation is replicated 100 times
- length of the time series is 1000

Knot positions

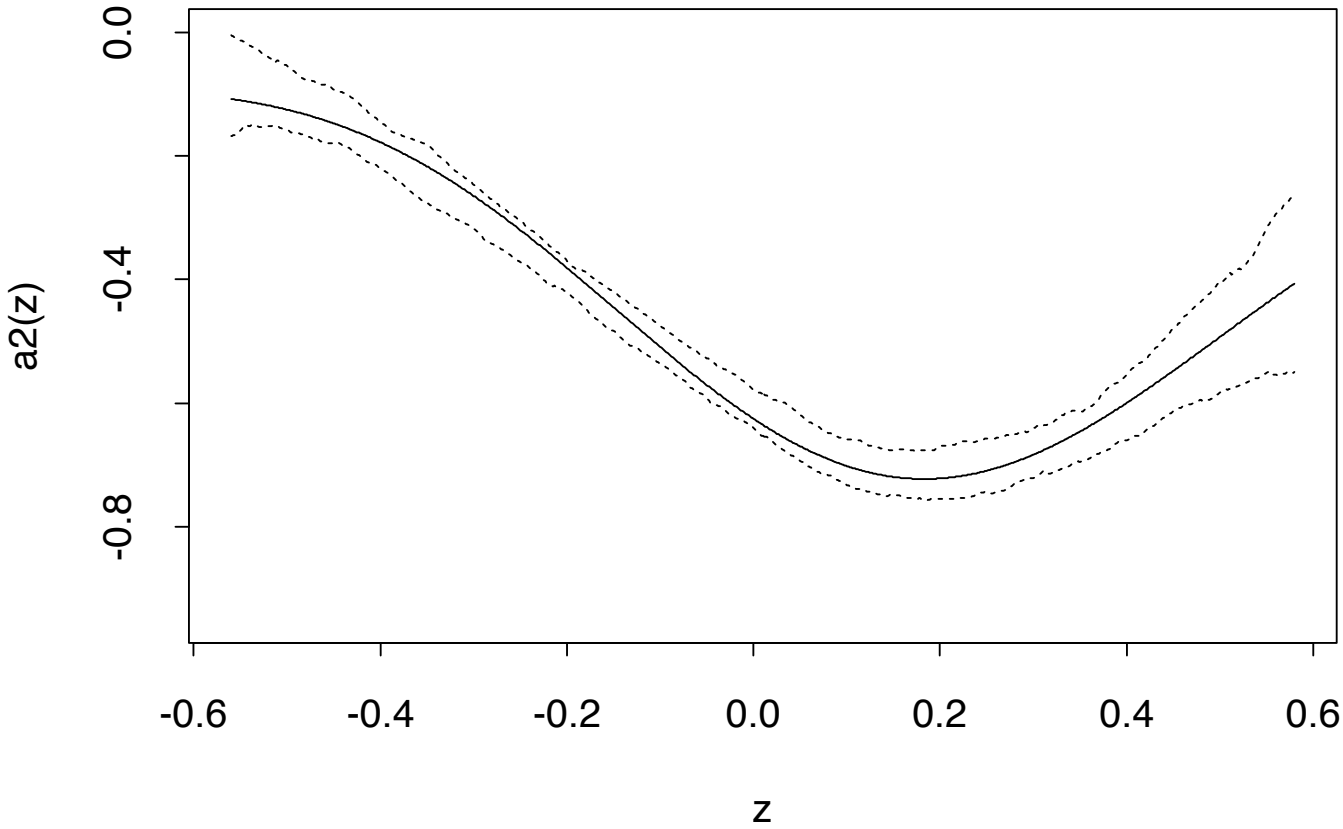
- the data range may vary substantially across simulations
- left boundary knot: the 0.5 percentile of each data set
- right boundary: the 99.5 percentile of the data
- interior knots: equally spaced

Results: $a_1(z)$



Solid: real, Dashed: quartiles

Results: $a_2(z)$



Application to US GNP

- the data are quarterly, seasonally adjusted, real US GNP in 1982 dollars from 1947Q1 to 1991Q1
- 177 raw observations $\{z_t\}$.
- the transformed data $y_t = 100 \log (z_t / z_{t-1})$
- Goal: model the dynamics of US GNP and make predictions
- traditional approach: to fit a linear autoregressive (AR) model to the series $\{y_t\}$
- evidence of nonlinearity of US GNP is provided and nonlinear models are proposed in the econometrics literature (Potter 1995)
- fit a functional coefficient model and perform multi-step (out-of-sample) forecasts based on the fitted model

Application Glossary

- GNP (gross national product): the market value of final goods and services newly produced *by domestic factors of production* during the current period
- real GNP: GNP taking inflation into account, nominal GNP
- GDP (gross domestic product): *production taking place within a country*

US GNP: AR model

- select the best model from the class of models,
  $y_t = a_0 + \sum_{j=1}^{p} a_j y_{t-j} + \epsilon_t$, $1 \leq p \leq 8$
- using AIC, an AR(3) model is chosen
- the fitted model is
  $\hat{y}_t = 0.508 + 0.342 y_{t-1} + 0.178 y_{t-2} - 0.148 y_{t-3}$

US GNP: Functional coefficient model

- $p_{max} = q_{max} = 4$
- select the threshold lag and significant lags
  $\hat{y}_t = \hat{a}_1(y_{t-2})y_{t-1} + \hat{a}_2(y_{t-2})y_{t-2}$
- quadratic splines with equally spaced knots
- put the boundary knots at some sample quantiles

Table: *Mean Squared Prediction Error (MSPE) of multi-step Forecasts: 60 Series.*

forecast horizon	1	2	3	4	5	6
AR MSPE	1.064	1.181	1.254	1.229	1.224	1.145
ANN/AR	1.114	0.970	0.976	0.966	1.000	1.046
FC/AR	0.999	0.869	0.902	0.913	0.938	0.955

forecast horizon	7	8	9	10	11	12
AR MSPE	1.045	0.891	0.896	0.890	0.905	0.934
ANN/AR	0.775	1.138	1.228	1.230	1.555	2.866
FC/AR	1.001	1.024	1.013	1.005	0.991	0.963

ANN: artificial neural network

Lognormal nonparametric regression models

- Z is a lognormal (LN) random variable $\Leftrightarrow$ $Y = \log(Z)$ is normal.
- $Z \sim \mathrm{LN}(\mu, \sigma^2) \quad \Leftrightarrow \quad Y = \log(Z)$ is normal with mean μ and variance σ^2.
- Then $\nu = \mathrm{E}(Z) = \exp(\mu + \sigma^2/2)$.
- Observe data $\{X_i, Z_i\}_{i=1}^n$ where $Z_i | X_i \sim \mathrm{LN}(f(X_i), g(X_i))$.
- Let $Y_i = \log Z_i$, then the following model is true:

$$Y_i = f(X_i) + \sqrt{g(X_i)}\epsilon_i,$$

  where ϵ_i's are $i.i.d.$ standard normal.
- Goal: estimate $\nu(x) = \exp(f(x) + g(x)/2)$

Spline approximation and maximum likelihood

- Use polynomial splines to approximate the smooth functions
- $$f(x) \approx \sum_{s=1}^{K^f} \beta_s^f B_s^f(x),$$
$$\log g(x) \approx \sum_{s=1}^{K^g} \beta_s^g B_s^g(x),$$
or
$$g(x) \approx \exp(\sum_{s=1}^{K^g} \beta_s^g B_s^g(x)).$$

Spline approximation and maximum likelihood

- The (approximate) likelihood function:

$$
\begin{aligned}
L(\beta^f, \beta^g) &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi g(x_i)}} e^{-\frac{(y_i - f(x_i))^2}{2g(x_i)}} \\
&\approx \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\exp\left(\sum_{s=1}^{K^g} \beta_s^g B_s^g(x_i)/2\right)} e^{-\frac{\left[y_i - \sum_{s=1}^{K^f} \beta_s^f B_s^f(x_i)\right]^2}{2\exp\left(\sum_{s=1}^{K^g} \beta_s^g B_s^g(x_i)\right)}}.
\end{aligned}
$$

- maximize the above likelihood function:
  - alternate between β^f and β^g
  - Newton methods

A Simulation Study

- Setup:

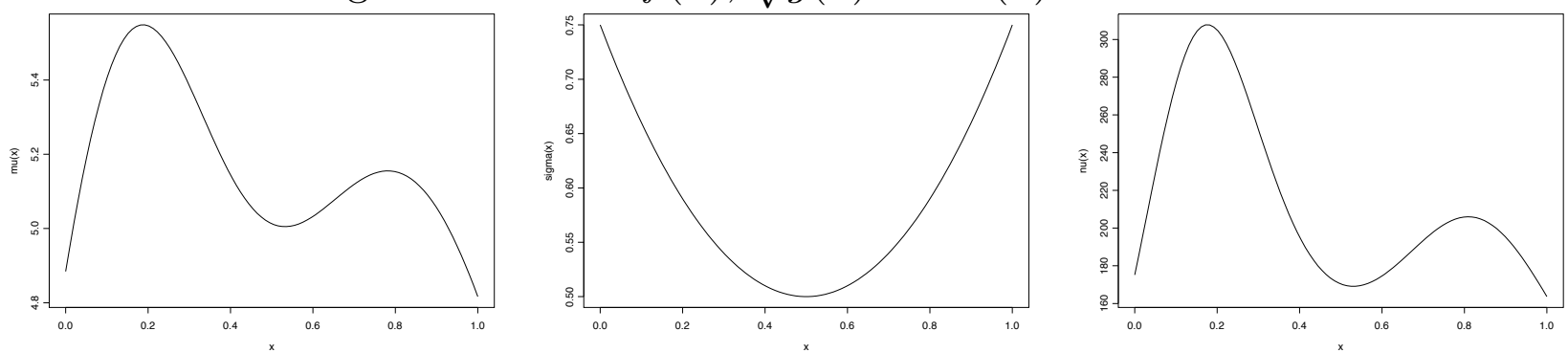
$$f(x) = 3 + 6(x + 0.3)e^{-8x^2} + 2(x + 0.3)e^{-4(x-0.7)^2},$$

$$g(x) = (a + (x - 0.5)^2)^2,$$

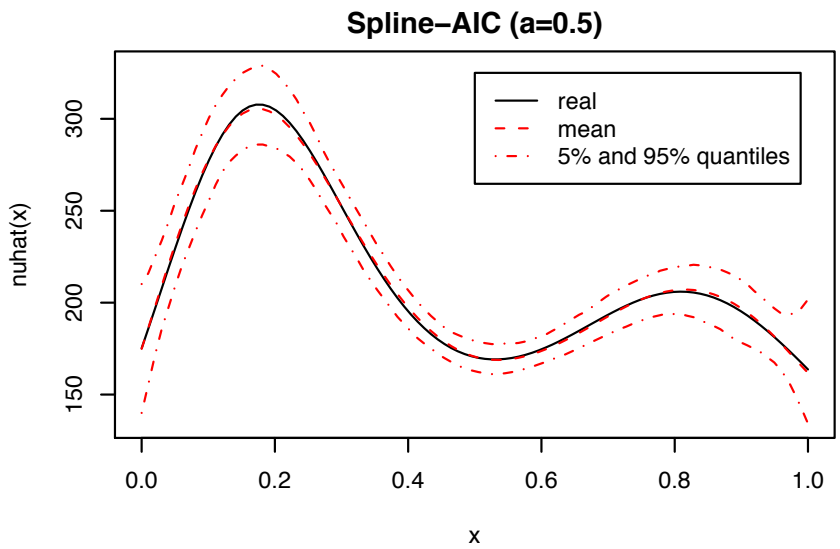
where $a = 0.5$ or 0.75.

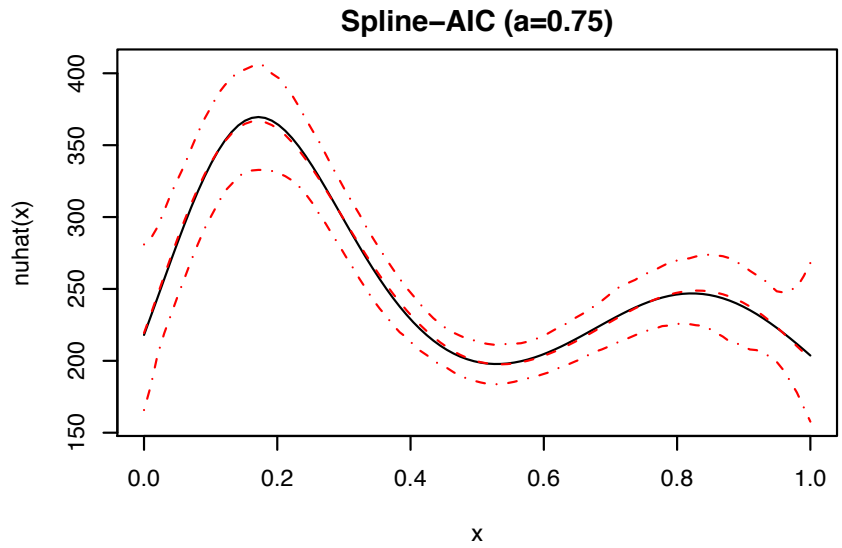
Sample size: $n = 2000$

Figure 1: Plot of $f(x)$, $\sqrt{g}(x)$ and $\nu(x)$ for $a = 0.5$



Simulation Results

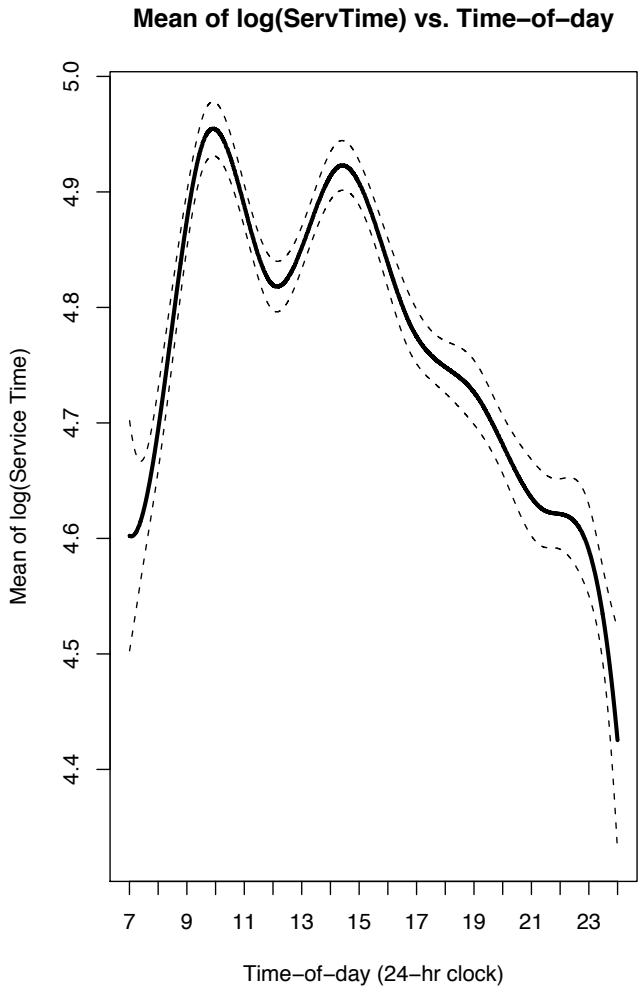


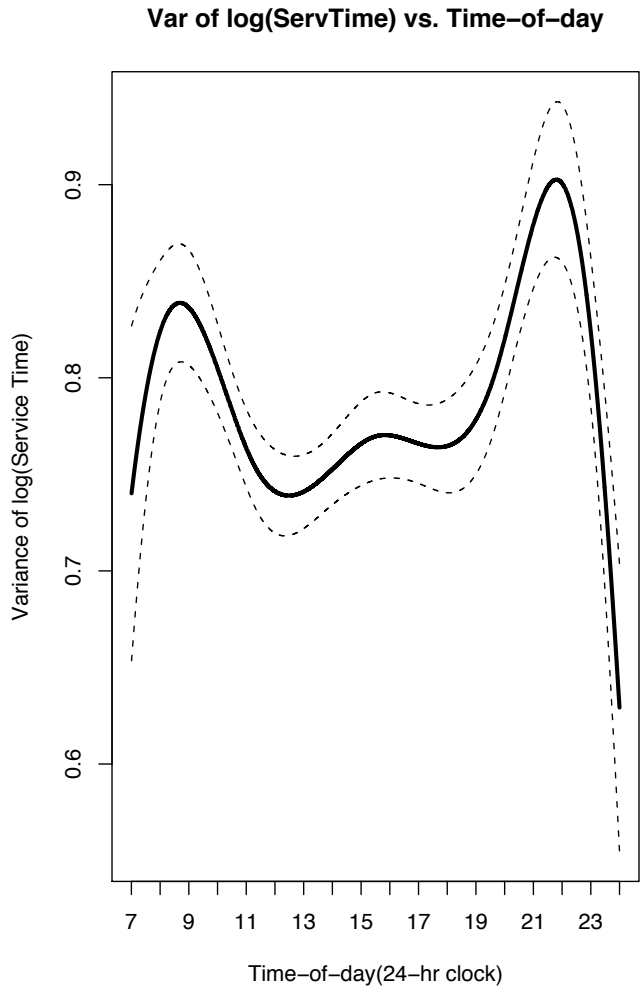


Application to Call Center Service Times

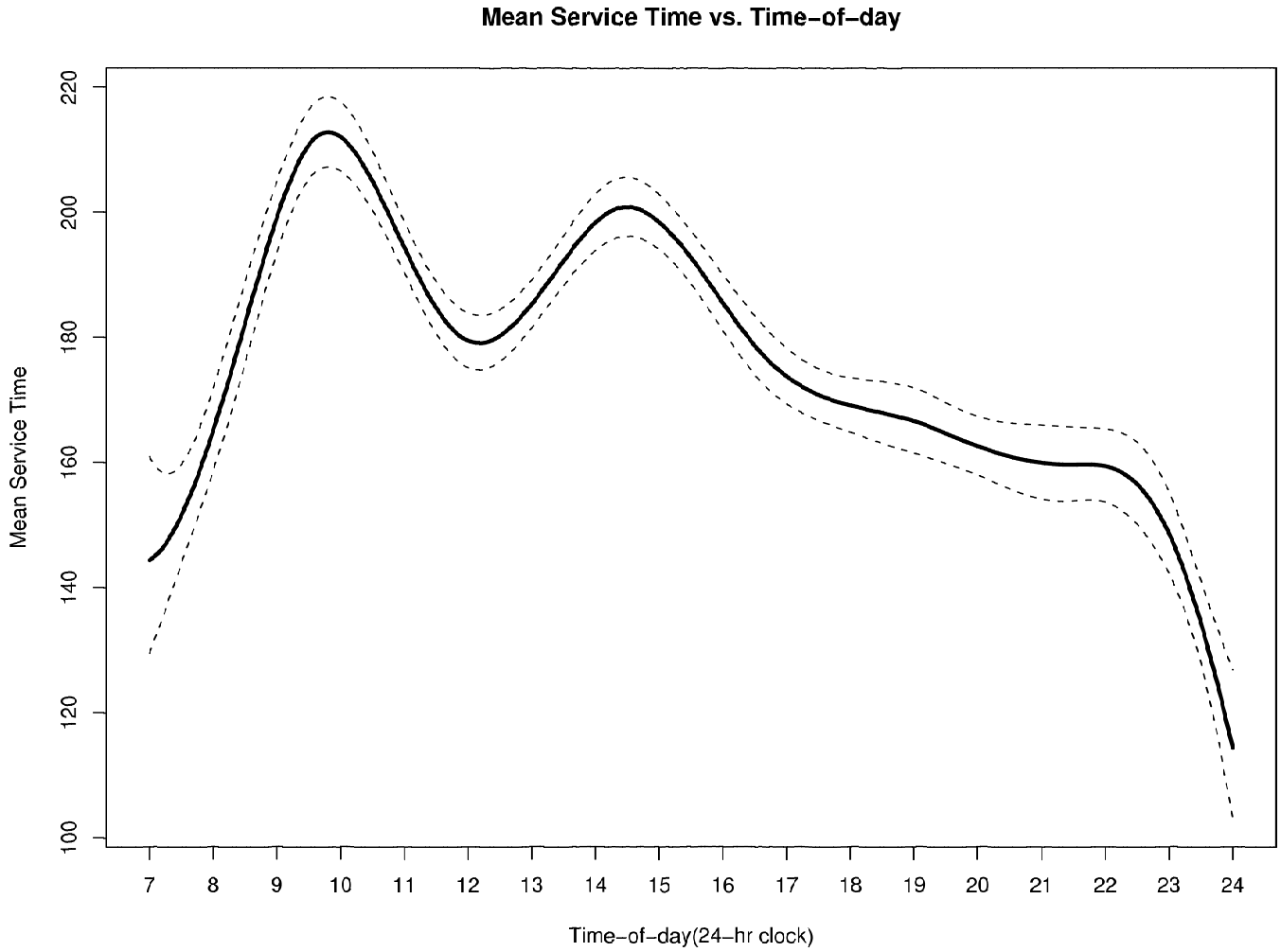
- Israeli banking call center
- Customers talk with service representatives about their problems.
- The conversation times are found to be lognormally distributed (Brown et al. 2005).
- Goal: estimate the average conversation time as a function of time-of-day.

Mean/Variance Functions of Log(Service Time)





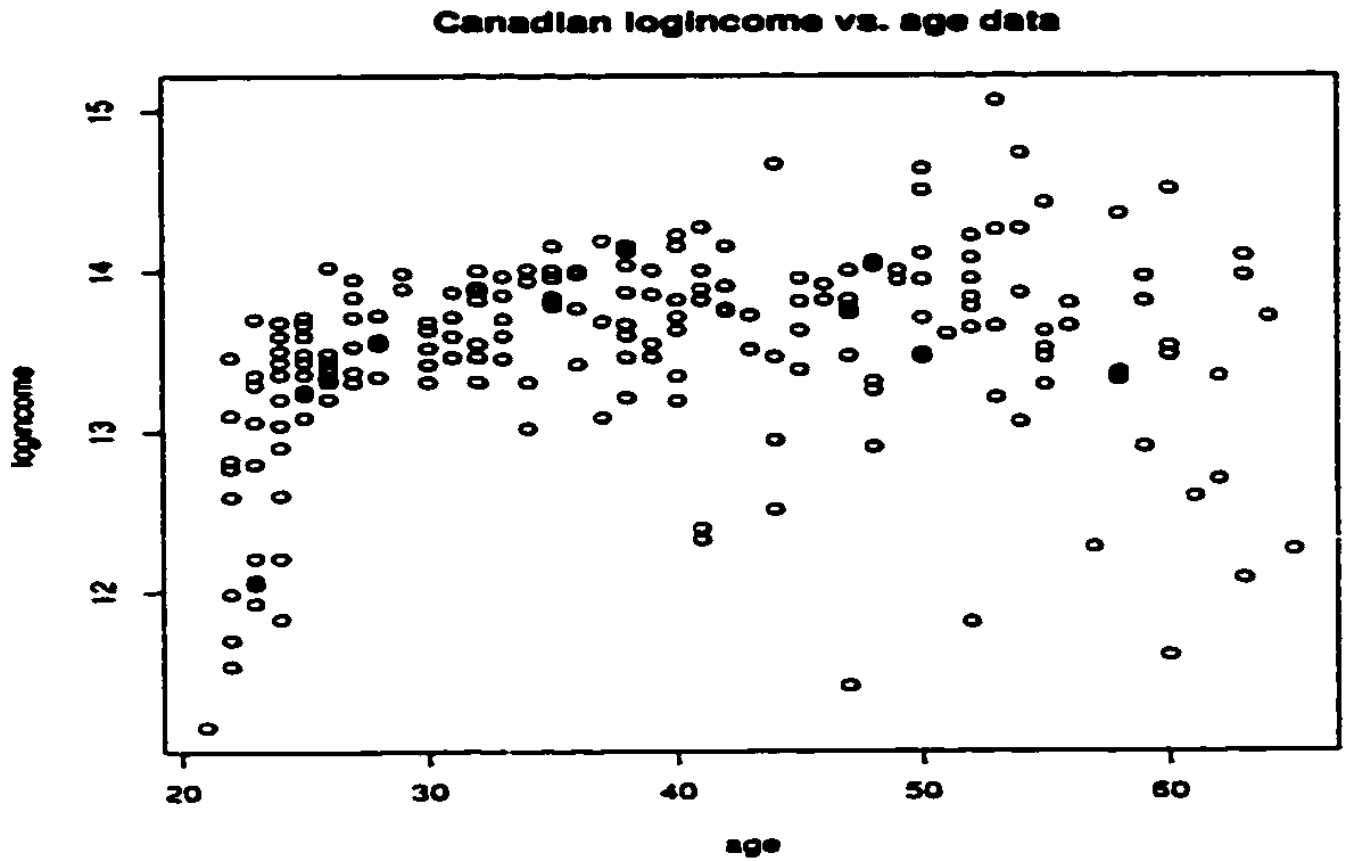
Mean Service Time Function



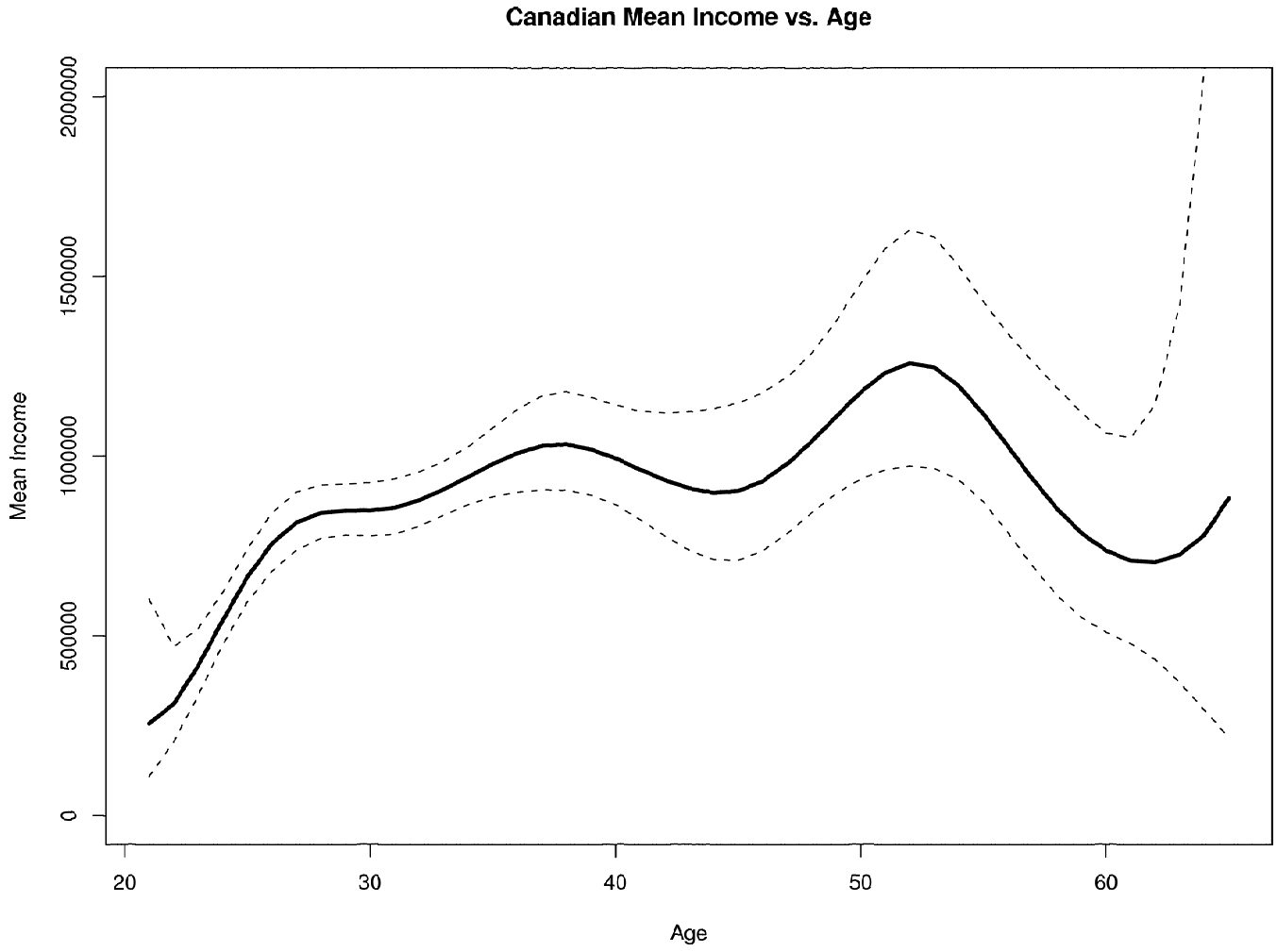
Application of Canadian Income vs. Age

- Age and yearly income of 205 residents of Canada (Ullah, 1985; Chu and Marron, 1991)
- Income is found to be lognormally distributed
- Goal: estimate mean income as a function of age

Scatterplot: Log(Income) vs. Age



Mean Income Function



Natural Cubic Splines

A natural cubic spline (NCS) g on $[a, b]$ satisifes

- g is a cubic spline on $[a, b]$
- g is linear on $[a, t_1]$ and $[t_m, b]$
- dimension: # of knots + 4 (for cubic splines) - 4 (for constraints) = # of knots
- in practice, it is common to add two boundary knots at the extremes of the data
- $[a, b]$ is arbitrary: NCS is linear beyond the data extremes

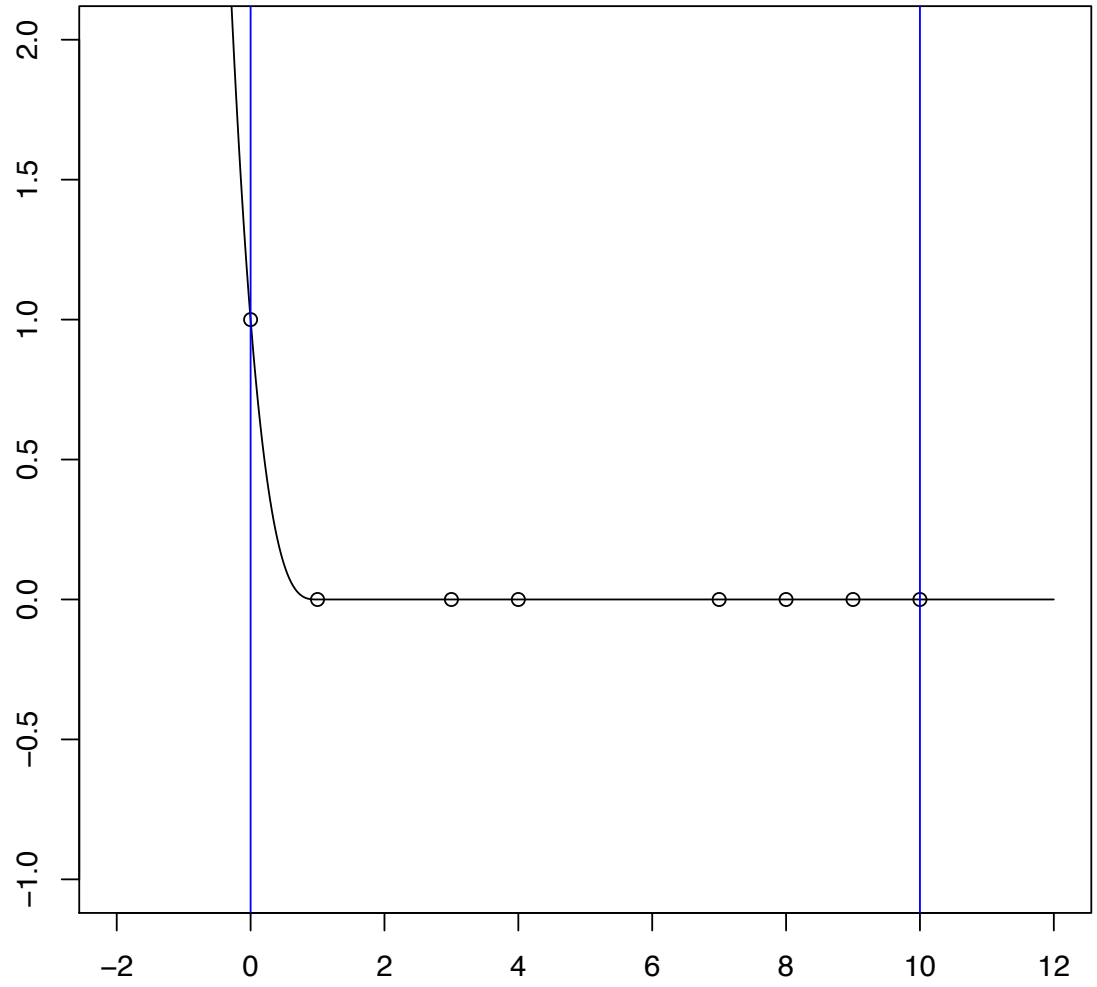
Cubic B-spline Basis and Natural Spline Basis

- NCS lives in a subspace of the linear space spanned by the cubic B-spline basis
- Note the boundaries effects: cubic vs. linear
- Knots at 0, 1, 3, 4, 7, 8, 9, 10

Comparison of Cubic B-spline Basis and Natural Spline Basis

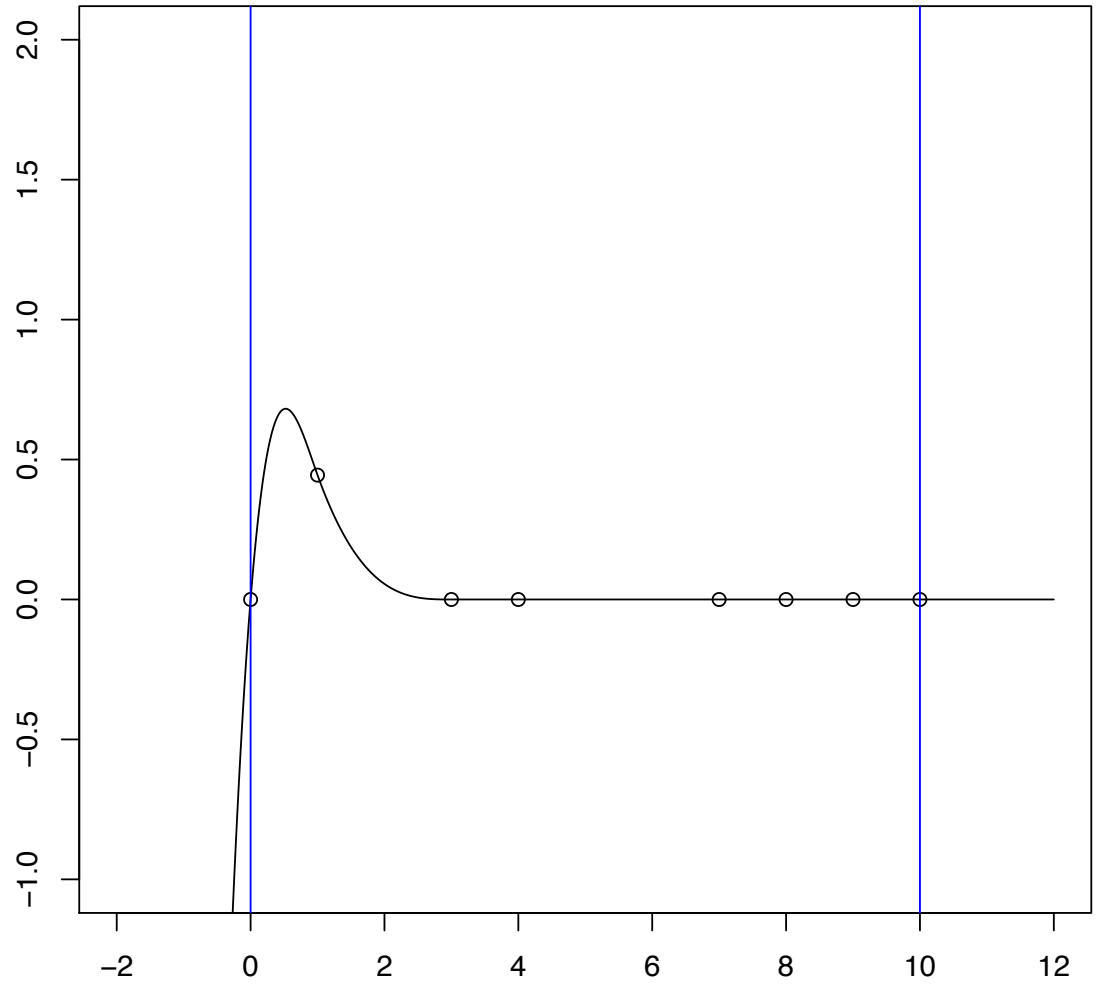
- # of B-spline basis fns: $k + 4 = 6 + 4 = 10$
- # of natural spline basis fns: $k + 2 = 6 + 2 = 8$
- Note the boundaries effects: cubic vs. linear

Comparison of Cubic B-splines and NCS



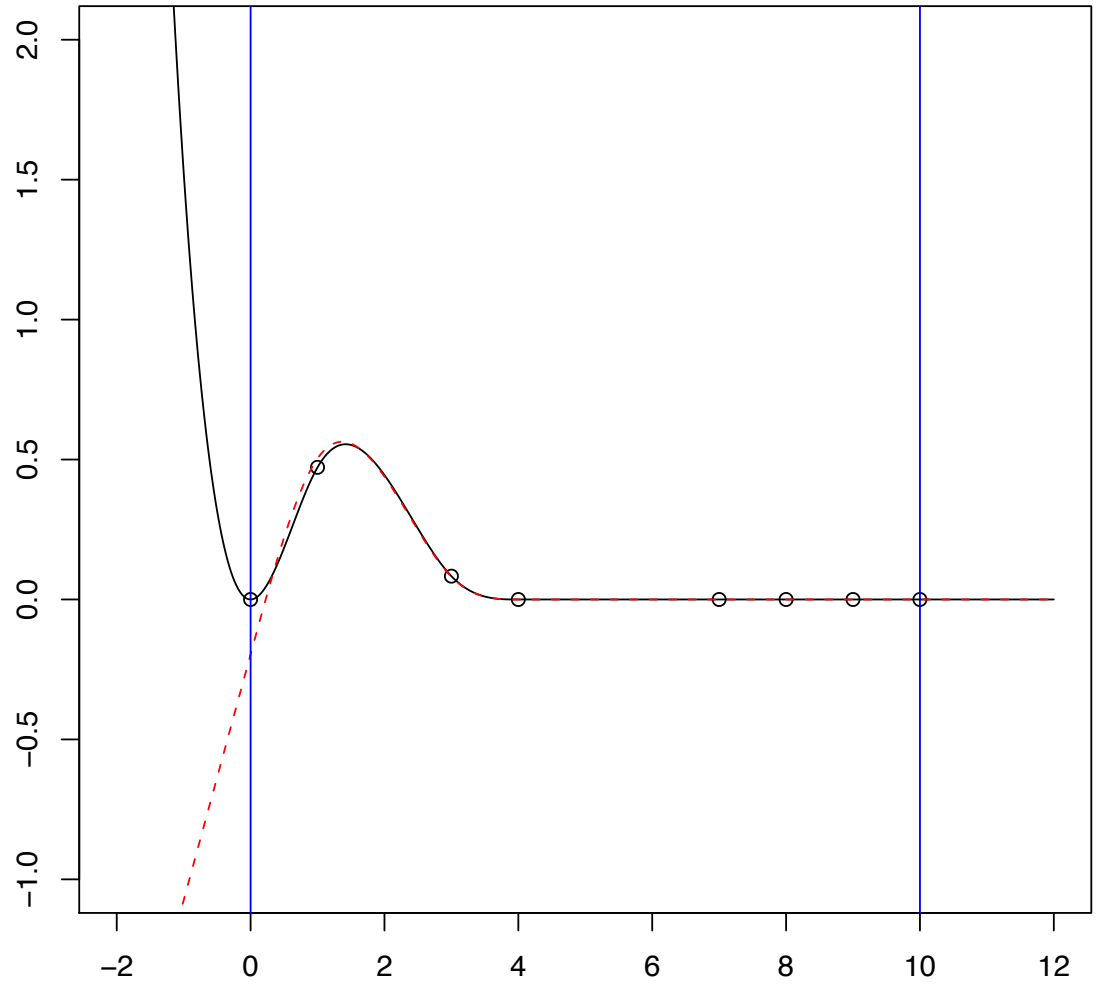
Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



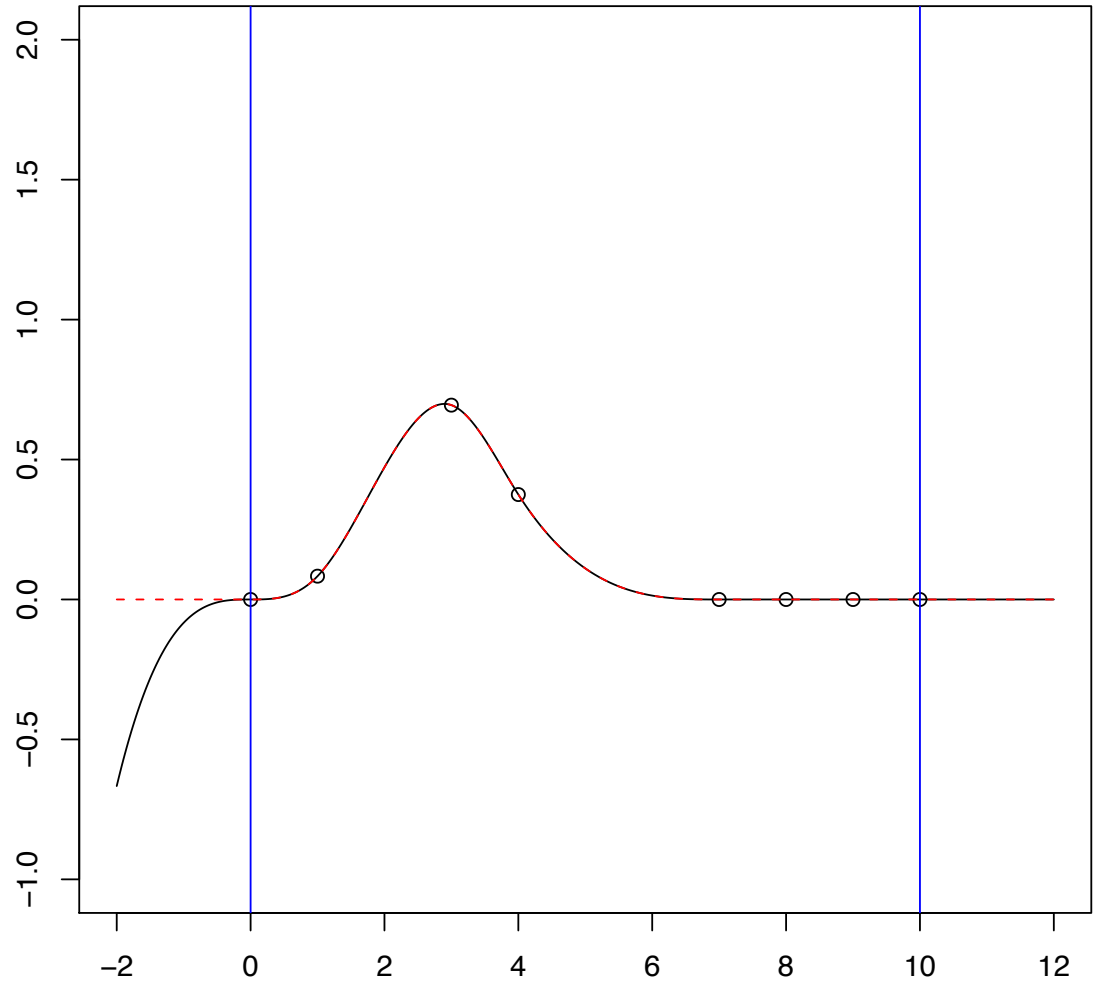
Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



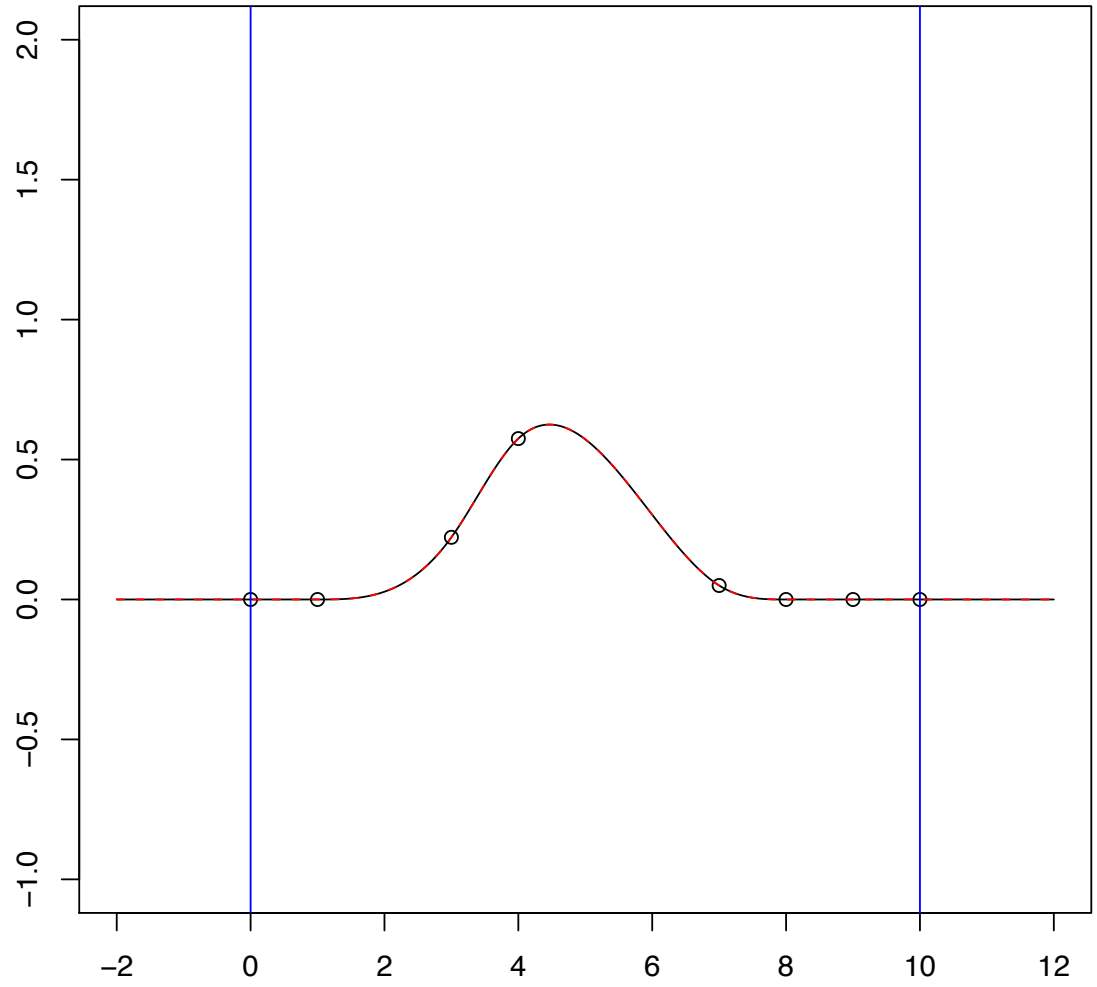
Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



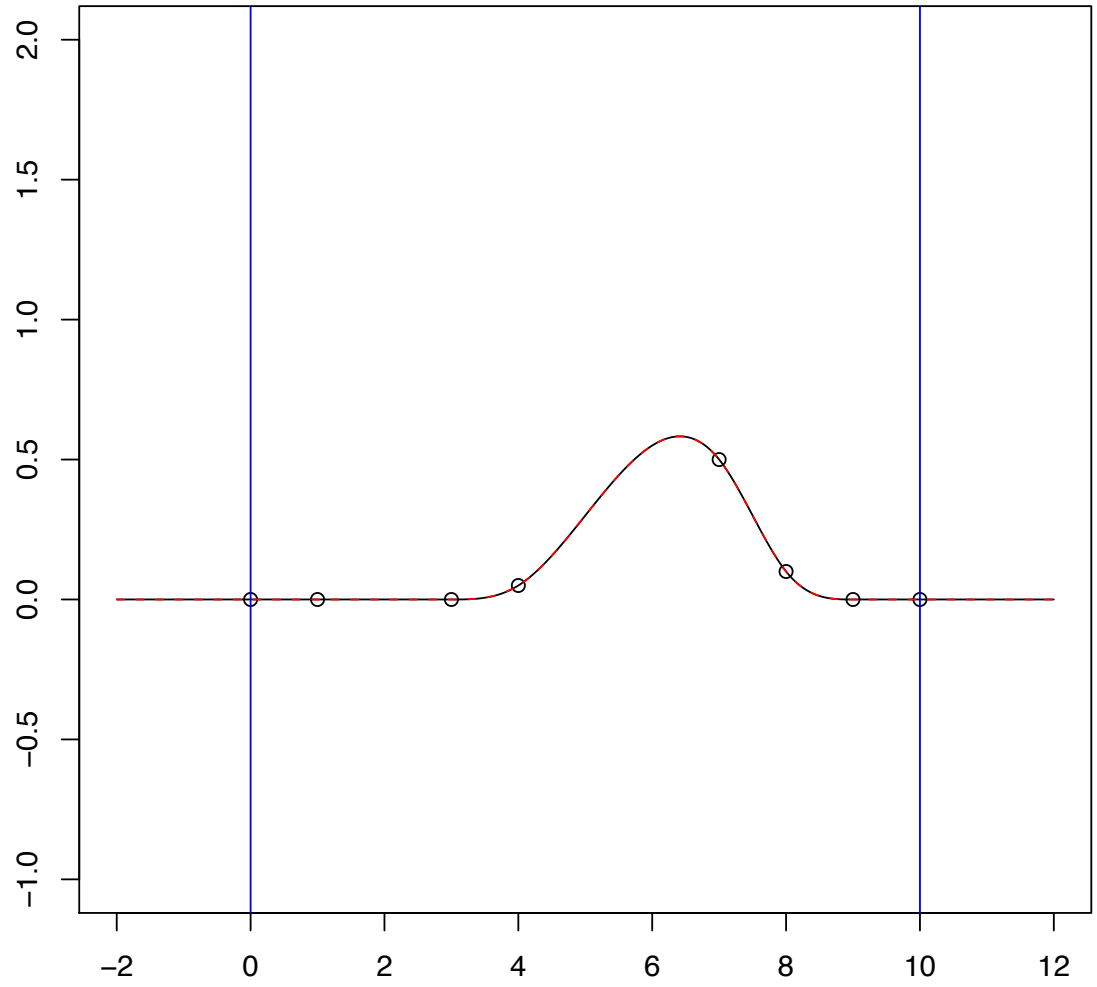
Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



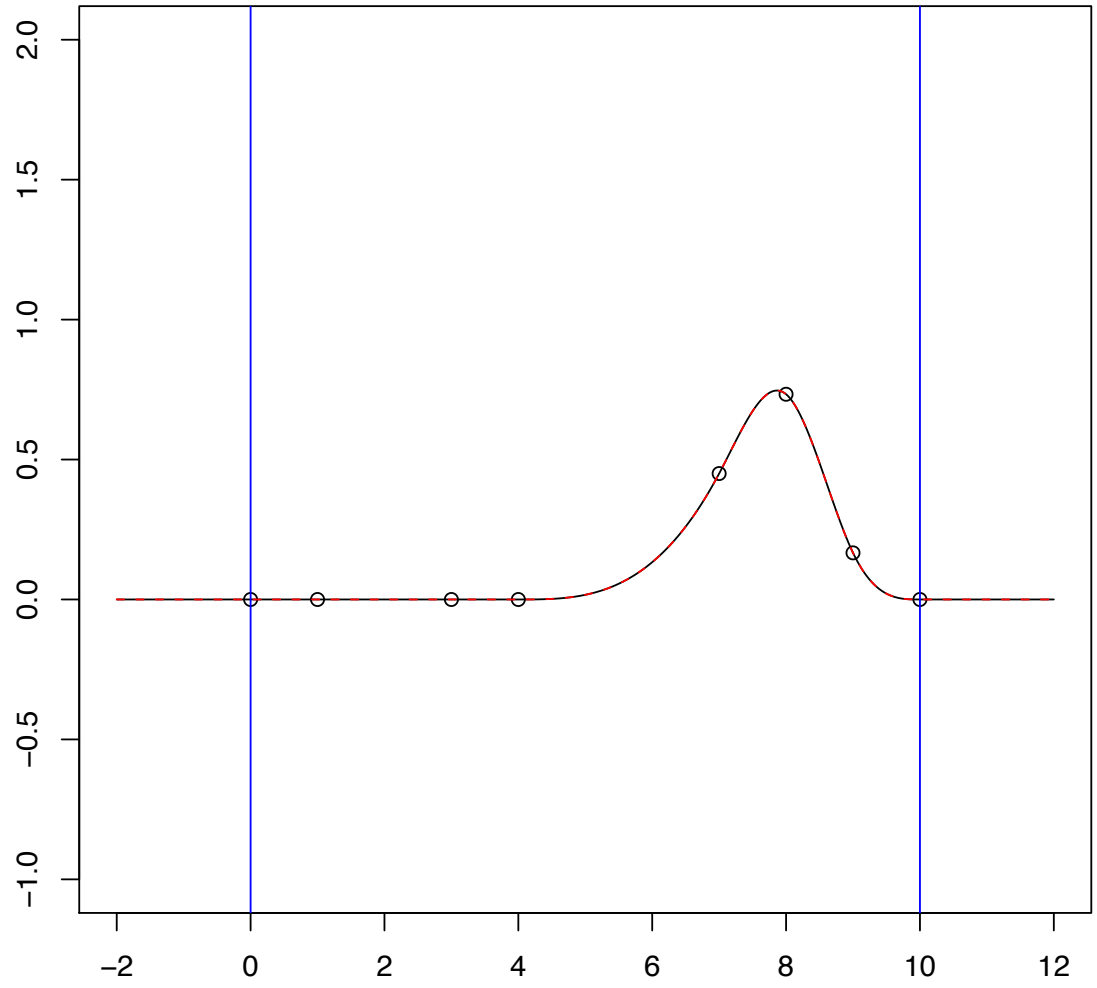
Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



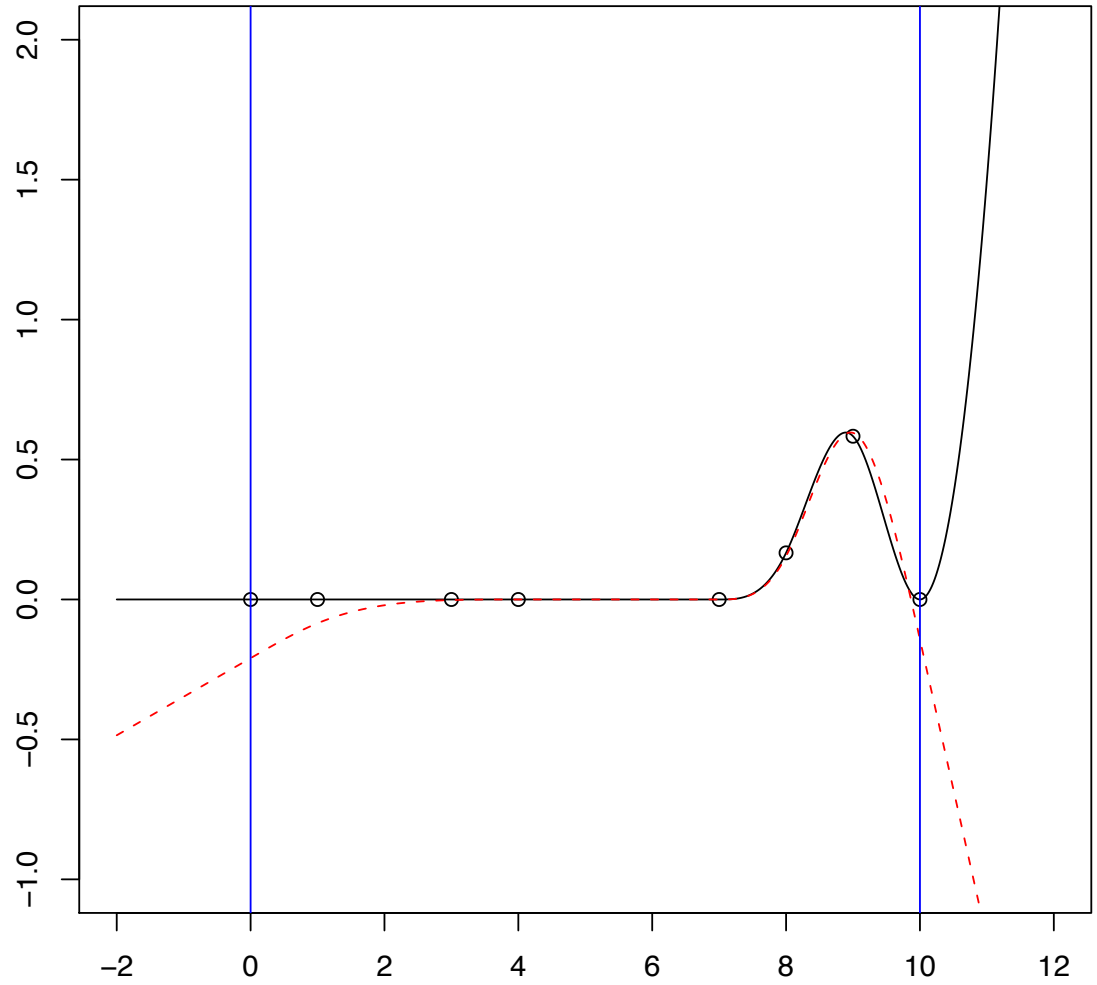
Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



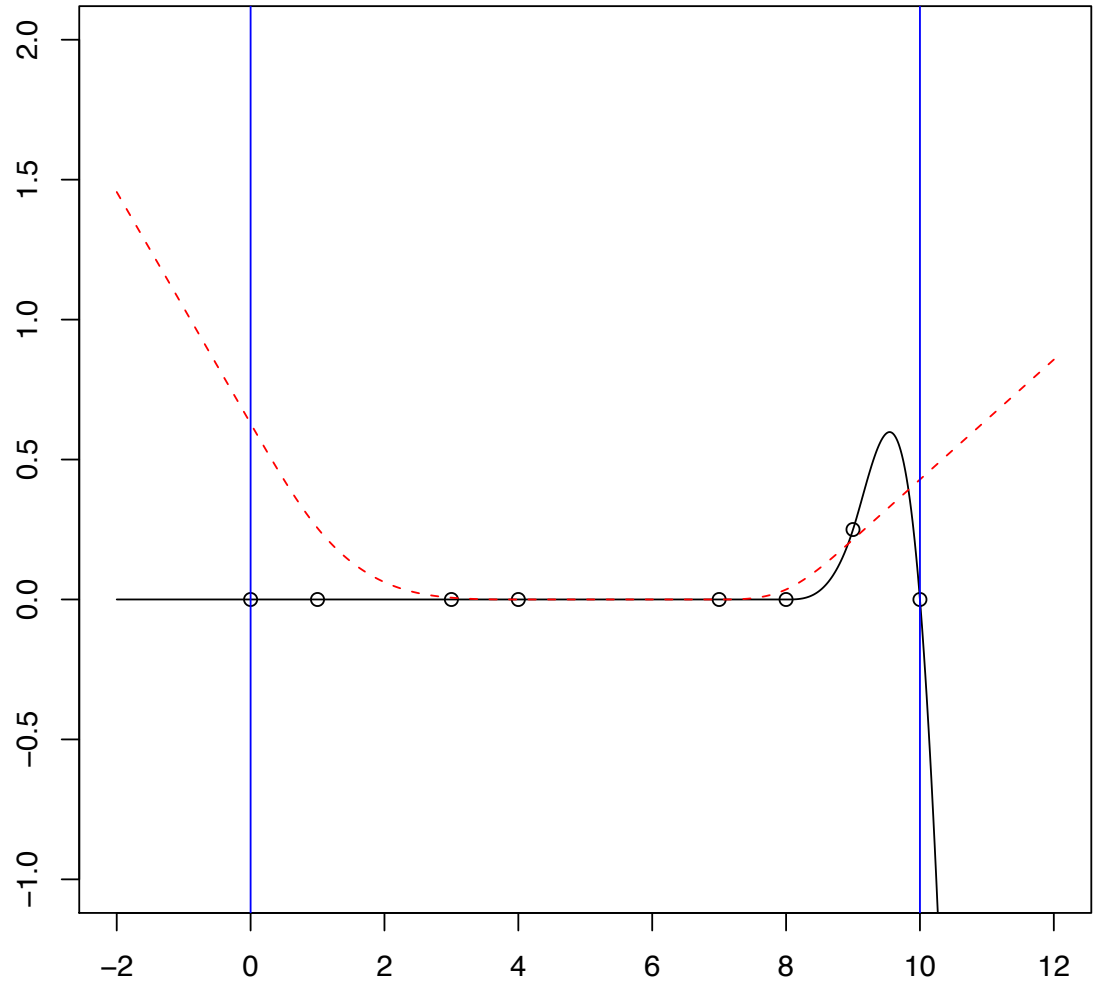
Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



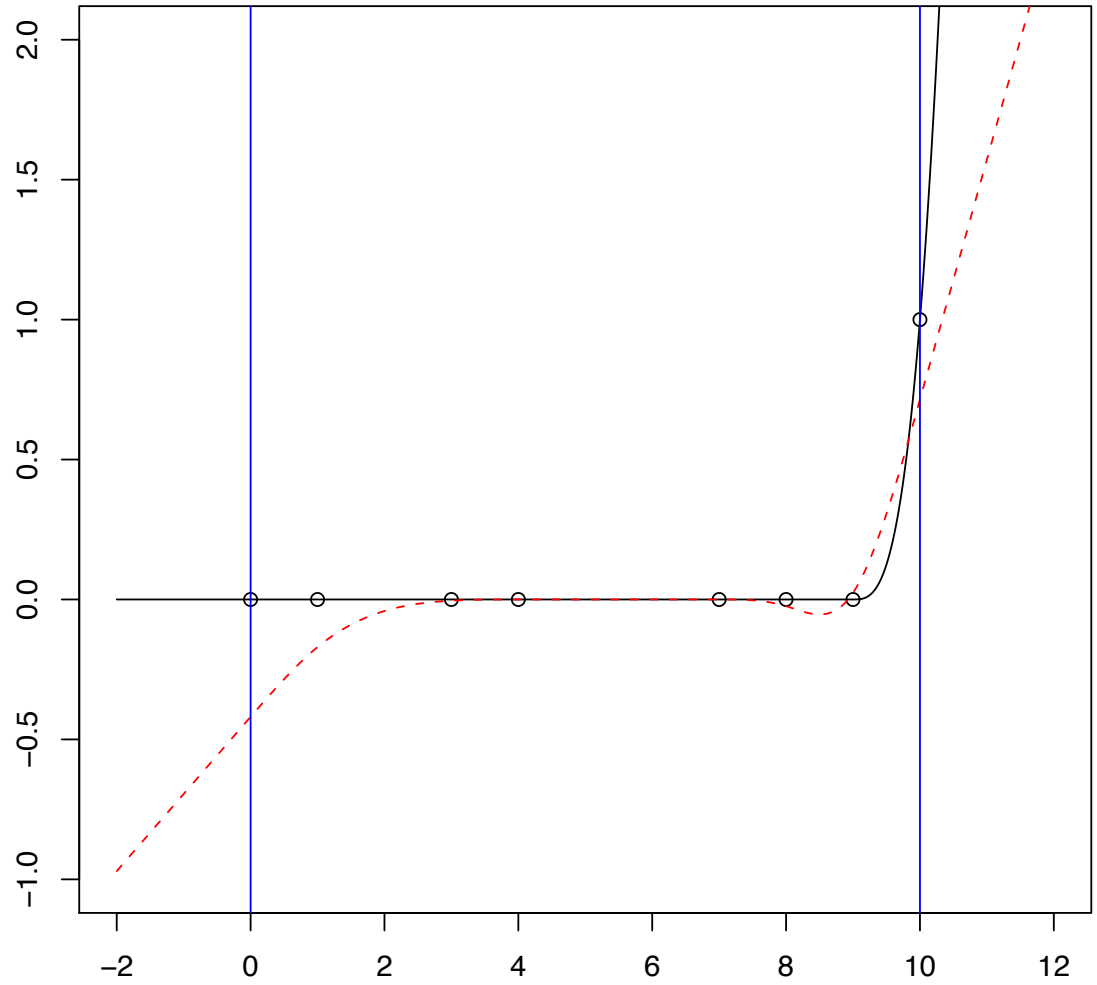
Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



Black: Cubic B-spline, Red: NCS

Comparison of Cubic B-splines and NCS



Black: Cubic B-spline, Red: NCS

Smoothing Splines

July 2016

Smoothing Splines

Given observations (t_i, Y_i), $i = 1, \ldots, n$:

$$Y = g(t) + \epsilon,$$

where $g(\cdot)$ is a curve that is twice differentiable.

To estimate $g(\cdot)$, we minimize

$$\sum_{i=1}^{n} (Y_i - g(t_i))^2 \quad + \quad \alpha \int_a^b \{g''(t)\}^2 dt,$$

where α: smoothing parameter, controlling the balance between goodness of fit and smoothness of the estimate.

Connection between Smooth Splines and NCS

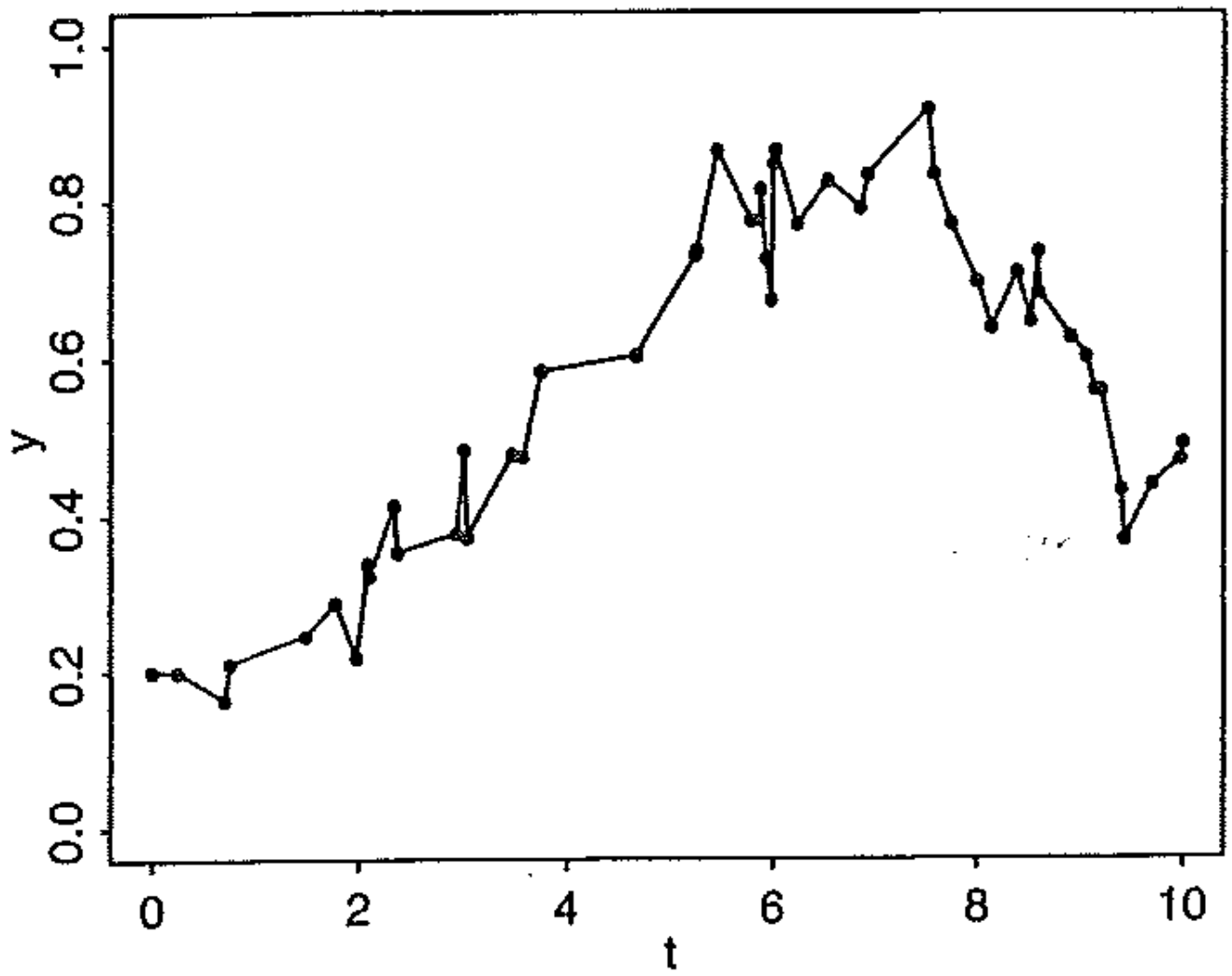
- Green and Sliverman (1994)
- The minimizer $\hat{g}(t)$ is a NCS with knots at $t_1,\ t_2,\ \ldots,\ t_n$.
- Define banded matrices $Q_{n\times(n-2)}$ and $R_{(n-2)\times(n-2)}$ according to Green and Silverman (1994), using the gaps between two consecutive knots.
- Define $\Omega = QR^{-1}Q^T$.
- Let $\nu = (\nu_1, \ldots, \nu_n)^T$. Consider the minimizer $\hat{\nu}$ of

$$\sum_{i=1}^{n}(Y_i - \nu_i)^2 \quad + \quad \alpha\, \nu^T \Omega \nu.$$

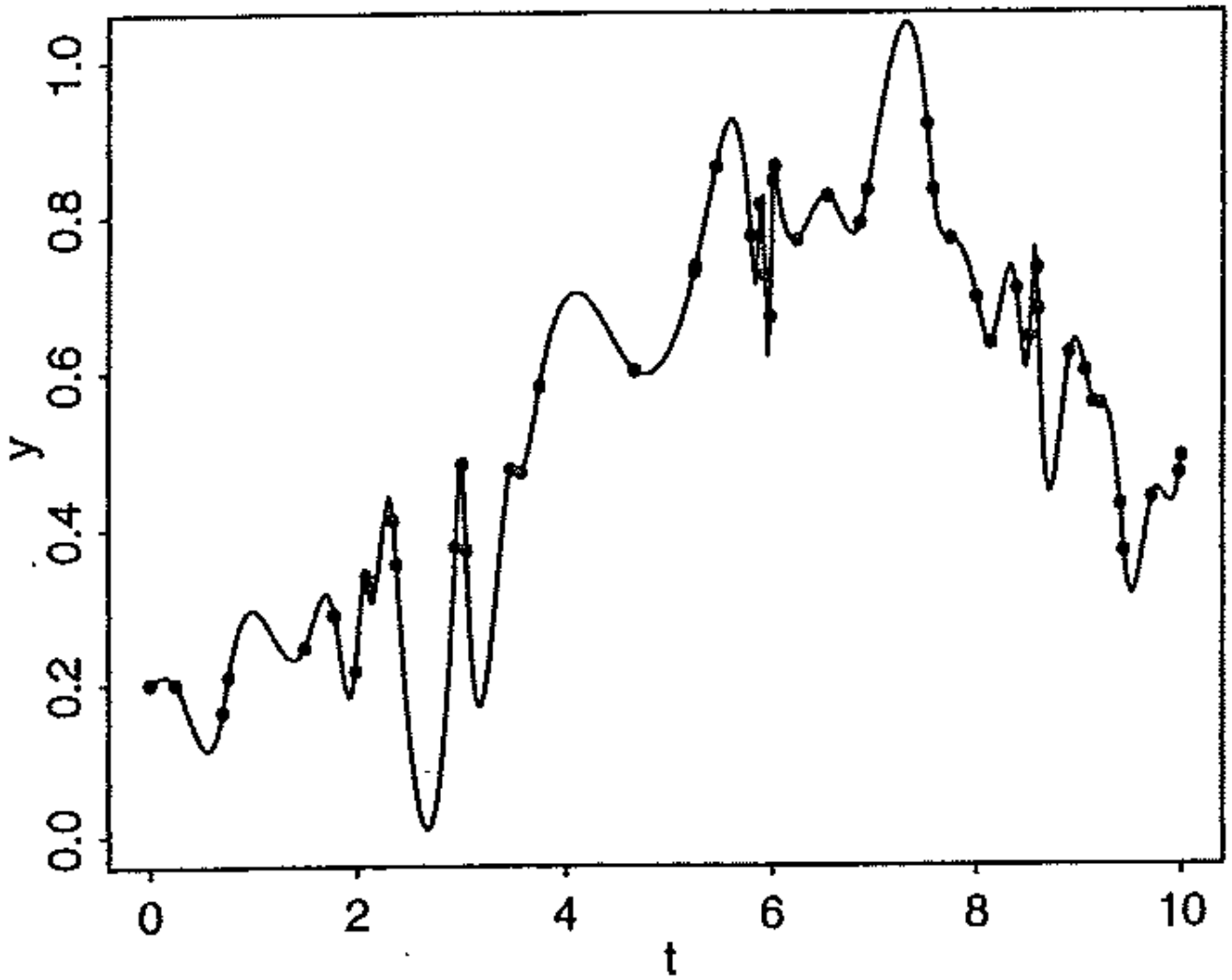
- Then, $\hat{\nu} = (I + \alpha\Omega)^{-1} Y$.
- Furthermore,

$$\hat{g}(t_i) = \hat{\nu}_i.$$

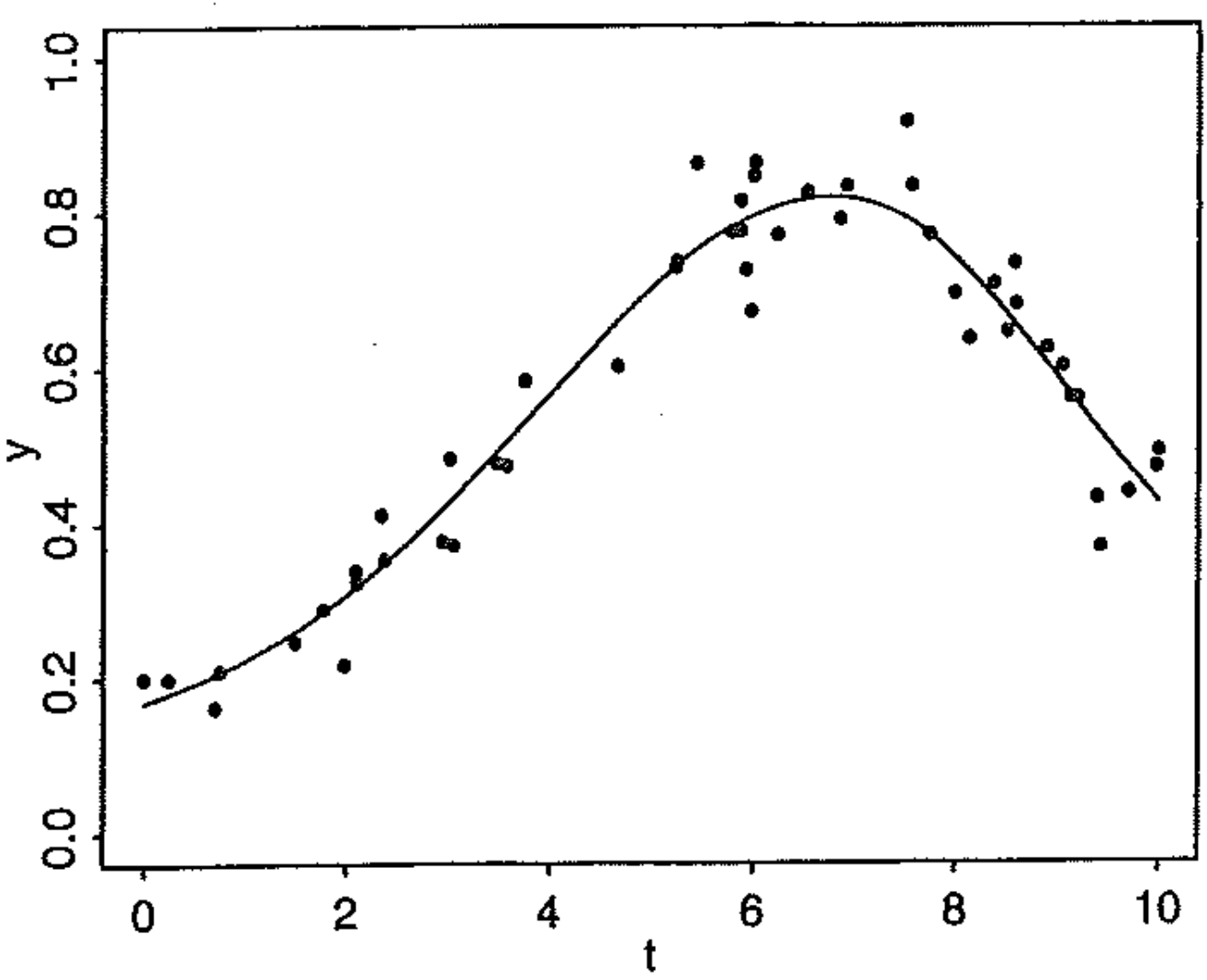
Example: Linear Interpolation



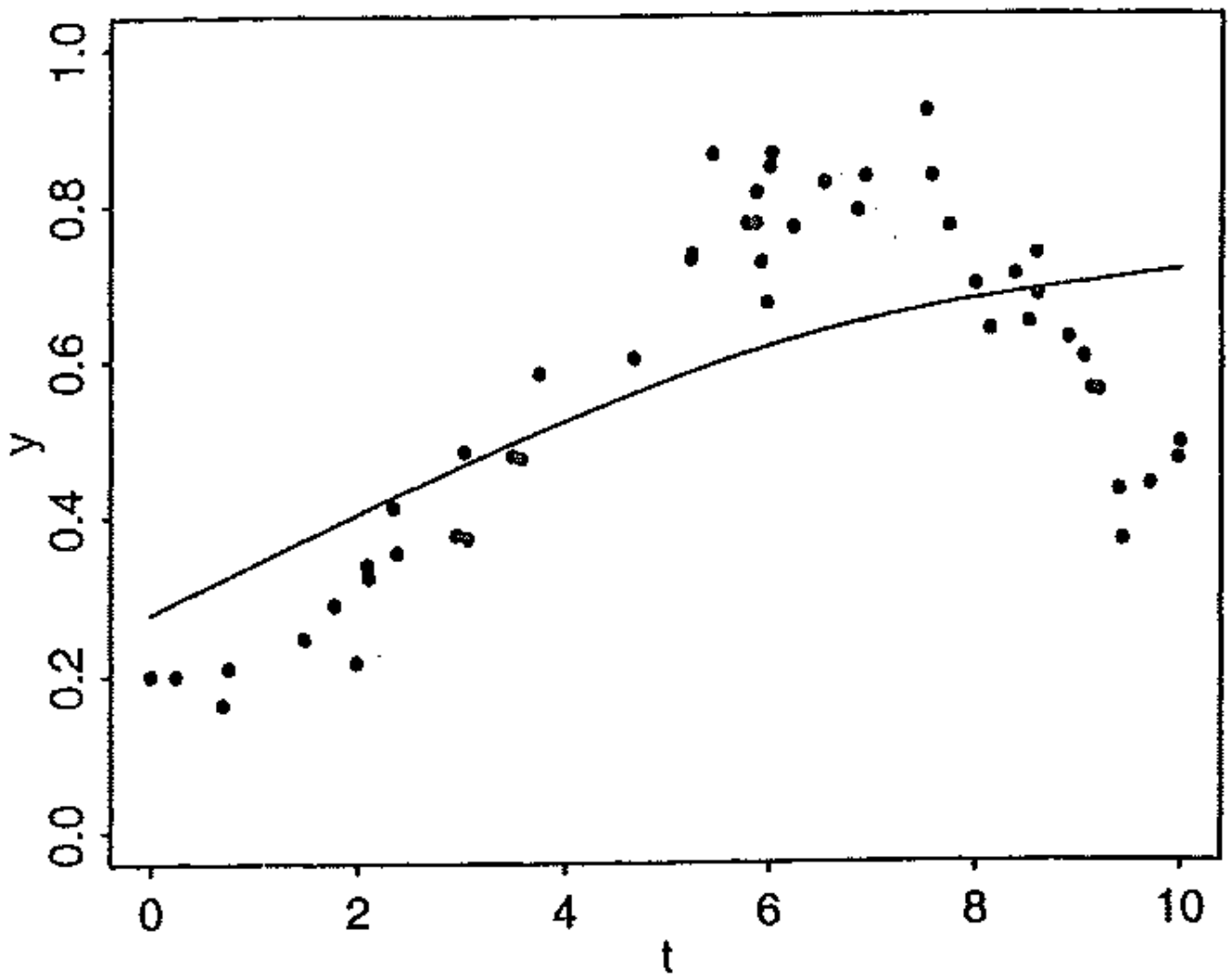
Example: Smoothing Spline Interpolation with $\alpha = 0$



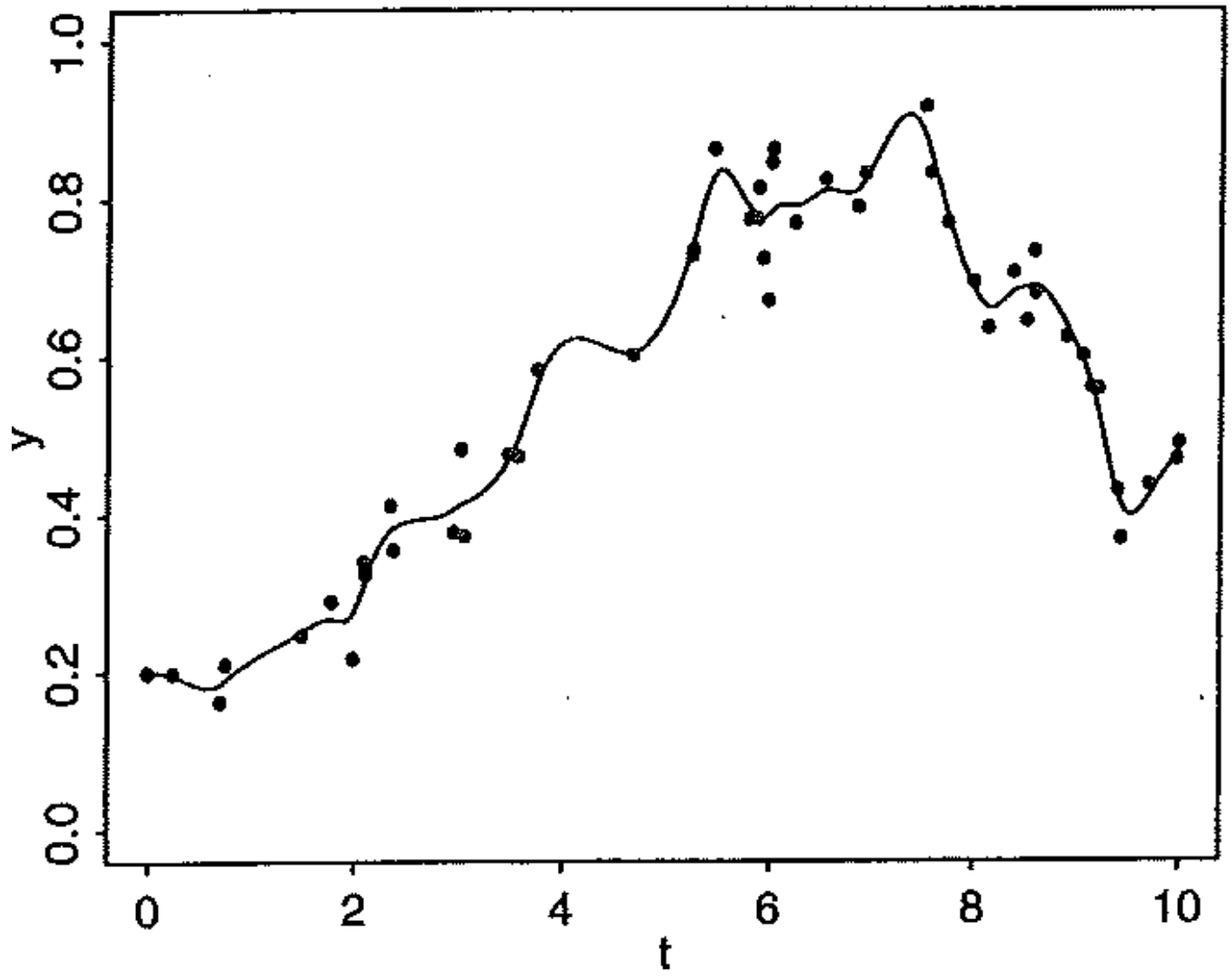
Example: Smoothing Spline with $\alpha = 1$



Example: Smoothing Spline with a large α



Example: Smoothing Spline with a small α



Application of NCS in Functional Data Analysis

July 2016

Outline

Principal Component Analysis (PCA)

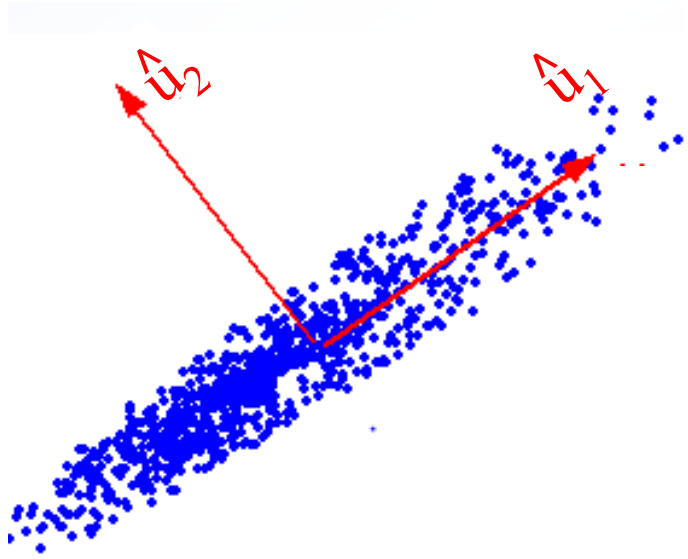
- History: Pearson (1901), Hotelling (1933)
- Consider $n \times d$ matrix $\mathbf{X}$: d variables on n observations. PCA *sequentially* solves the following optimization problem

$$\mathbf{v}_j = \underset{\{\mathbf{v}:\|\mathbf{v}\|=1\}}{\operatorname{argmax}} \operatorname{Var}(\mathbf{X}\mathbf{v}), \quad \text{subject to} \quad \mathbf{v}_j^T \mathbf{v}_{j'} = 0 \ \text{ for } \ j' < j. \tag{1}$$

  - $\mathbf{v}_j$: jth principal component (PC) loading vector
  - $\mathbf{X}\mathbf{v}_j$: jth PC, linear combination of the original variables
  - Interpretation: PCs ordered by importance, or variance explained
    - use first few PCs to explain most of variability in data
  - Calculation: $\mathbf{v}_j$ are eigenvectors of the covariance matrix of $\mathbf{X}$ or its sample version

Principal Component Analysis (PCA)

- Goal: obtain a few linear combinations of the raw variables to explain majority of the data variation



- Try Google Scholar search: how many????
  - Regularized siblings: functional PCA, sparse PCA, ...

Application 1: RNAseq

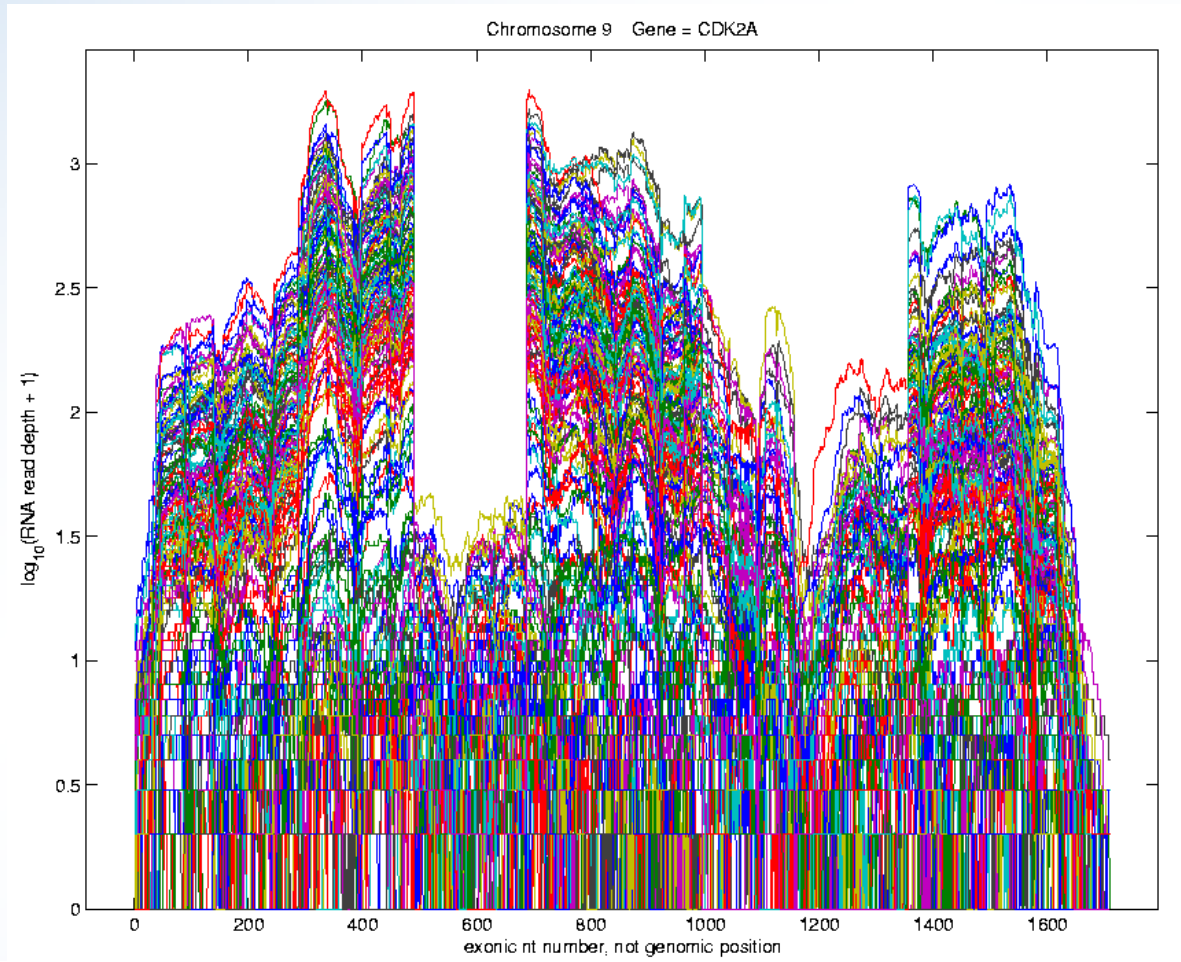
- Interesting Real Data Example
  - Genetics (Cancer Research)
  - RNAseq (Next Generation Sequencing)
  - Deep look at “gene components"

- Microarrays: single number per gene
- RNAseq: thousands of measurements

Application 1: RNAseq

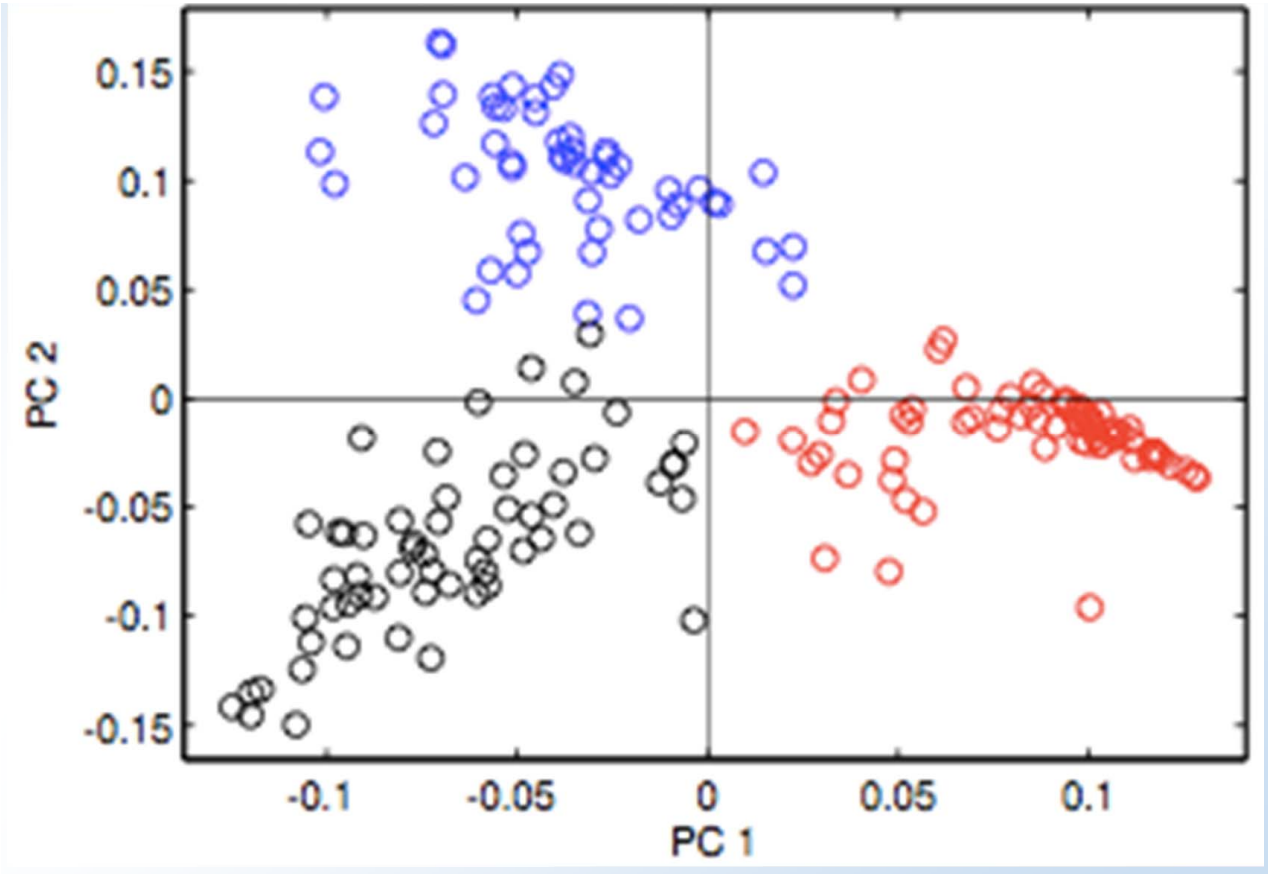
- Interesting Real Data Example
  - Genetics (Cancer Research)
  - RNAseq (Next Generation Sequencing)
  - Deep look at “gene components"
- Gene studied here: CDK2A
- Goal: Study Alternate Splicing
- Sample size: $n = 180$
- Dimension: $d = \sim 1700$

Application 1: RNAseq



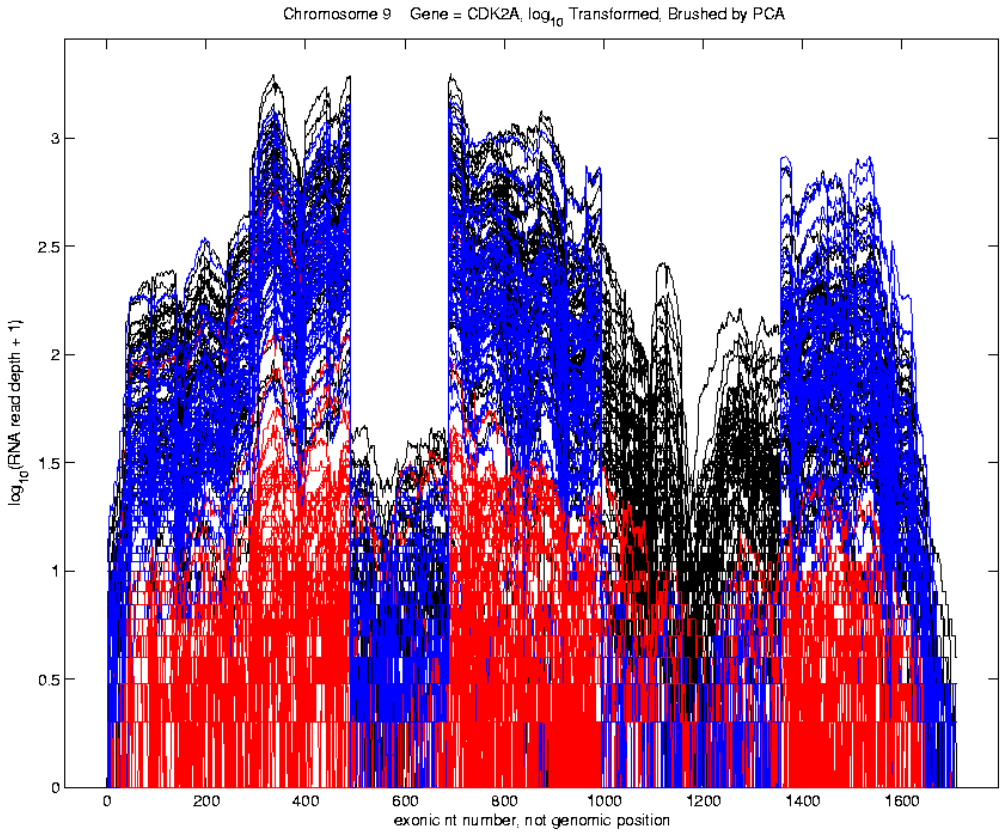
- Simple 1st View: Curve Overlay (log scale)

Application 1: RNAseq, PCA



- Manually brush clusters ...

Application 1: RNAseq



- Clear Alternate Splicing
  - Blue samples deletion in 4th exonic region ...

Movie Rental: Netflix

- A US-based DVD retail company, established in 1997
- As of September 2013, 40.4 million global streaming subscribers

- Good recommendation = happy customer

Recommender System: The Netflix Prize

- October 2006:
  - Offers $1,000,000 for an improved recommender algorithm
- Training data
  - 100 million ratings
  - 480, 000 viewers
  - 17,770 movies
  - 6 years of data: 2000-2005
- Test data
  - Last few ratings of each viewer (2.8 million)
- Winner
  - BellKor's Pragmatic Theory, using a combination of $>$ 800 models
  - Two main classes: nearest neighbors and principal component analysis (PCA)

PCA in Netflix

10,000 user ratings in Netflix Data

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6
Miss Congeniality	-0.296	0.463	-0.195			-0.146
Independence Day	-0.235	-0.233	-0.179		0.100	
Patriot	-0.241	-0.185	0.365	-0.385	0.136	-0.334
Day After Tomorrow	-0.301	-0.149		0.663	0.357	-0.185
Pirates Caribbean	-0.145		-0.209	-0.120	0.329	-0.525
Pretty Woman	-0.231	0.360		-0.188		0.469
Forrest Gump	-0.104			-0.289	0.321	0.197
The Green Mile	-0.153		0.148	-0.278	0.274	
Con Air	-0.285	-0.265	-0.285	-0.124	-0.443	
Twister	-0.289	-0.112	-0.216	0.146	0.283	0.450
Sweet Home Alabama	-0.296	0.517			-0.181	-0.236
Pearl Harbor	-0.345		0.716	0.225	-0.274	0.110
Armageddon	-0.313	-0.232			-0.183	
The Rock	-0.228	-0.271	-0.253	-0.224	-0.296	-0.167
What Women Want	-0.280	0.224		-0.228	0.207	

Singular Value Decomposition (SVD)

- History: Beltrami (1873), Jordan (1874)
- Consider $n \times d$ matrix $\mathbf{X}$ and $\mathrm{rank}(\mathbf{X}) = r$. Then the SVD of $\mathbf{X}$ is

$$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^T = \sum_{k=1}^{r} s_k \mathbf{u}_k \mathbf{v}_k^T, \tag{2}$$

  - left singular vector matrix: $\mathbf{U}_{n\times r} = \{\mathbf{u}_1, \ldots, \mathbf{u}_r\}$ with $\mathbf{u}_i^T \mathbf{u}_j = \delta_{ij}$;
  - right singular vector matrix: $\mathbf{V}_{d\times r} = \{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$ with $\mathbf{v}_i^T \mathbf{v}_j = \delta_{ij}$;
  - singular value matrix: $\mathbf{S}_{r\times r} = \mathrm{diag}(s_1, \ldots, s_r)$ with $s_1 \geq s_2 \geq \cdots \geq s_r > 0$;
  - Interpretation: pairs of singular vectors ordered by singular values.

Connection between PCA and SVD

- Assume $\mathbf{X}$ is column-centered
- Via SVD,

$$\mathbf{X} = \mathbf{USV}^T$$

- Then,

$$\mathbf{X}^T\mathbf{X} = \mathbf{VS}^2\mathbf{V}^T$$

- Also note, $\mathbf{X}^T\mathbf{X}$ is proportional to the covariance matrix of $\mathbf{X}$
- Hence,
  - columns of $\mathbf{V}$: the PC loading vectors
  - columns of $\mathbf{US}$: the PCs
  - squared singular values: variances of the PCs
  - the PCs are ordered

PCA and SVD

- SVD provides a better way to calculate PCA
  - Computational effort for SVD is determined by $\text{rank}(\mathbf{X}) = r$, (could be) much less than the dimension of the matrix
  - Can be much faster for high-dimension-low-sample-size (HDLSS) data, i.e. when $d \gg n$, for example, in microarray data
- SVD provides PCAs for both $\mathbf{X}$ and $\mathbf{X}^T$
  - columns $\mathbf{x}_j$ as variables: *Primal PCA*, $\mathbf{V}$ = PC loadings
  - rows $\mathbf{x}_{(i)}$ as variables: *Dual PCA*, $\mathbf{U}$ = PC loadings
  - both can make sense sometimes (Zhang et al. 2007)

Another view of SVD: low-rank approximation

- Eckart and Young (1936)
- For an integer $K \leq r$, consider an arbitrary rank-K matrix $\mathbf{X}^*$
- Consider the squared distance between $\mathbf{X}$ and $\mathbf{X}^*$, measured by the Frobenius norm,

$$\|\mathbf{X} - \mathbf{X}^*\|_F^2 = \mathrm{tr}\{(\mathbf{X} - \mathbf{X}^*)(\mathbf{X} - \mathbf{X}^*)^T\} = \sum_{i,j}(x_{ij} - x_{ij}^*)^2 \quad (3)$$

- Then, SVD provides the best rank-K approximation of $\mathbf{X}$:

$$\mathbf{X}^{(K)} \equiv \sum_{k=1}^{K} s_k \mathbf{u}_k \mathbf{v}_k^T = \underset{\{\mathbf{X}^*:\ \mathrm{rank}(\mathbf{X}^*)=K\}}{\mathrm{argmin}} \|\mathbf{X} - \mathbf{X}^*\|_F^2 \quad (4)$$

SVD: bilinear model

- Consider rank-1 approximation.
- Any rank-1 $n \times d$ matrix can be written as

$$s\mathbf{u}\mathbf{v}^T,$$

where

  - s: a positive scalar
  - $\mathbf{u}$: a norm-1 n-vector
  - $\mathbf{v}$: a norm-1 d-vector

- The first SVD triplet $\{s_1, \mathbf{u}_1, \mathbf{v}_1\}$ is the solution for

$$\min_{\{s,\mathbf{u},\mathbf{v}\}} \|\mathbf{X} - s\mathbf{u}\mathbf{v}^T\|_F^2 = \sum_{i,j}(x_{ij} - su_i v_j)^2$$

  - bilinear model
  - also true for rank-K approximation

SVD: alternating least squares

Equivalently, the first SVD triplet $\{s_1, \mathbf{u}_1, \mathbf{v}_1\}$ is the solution for

$$\min_{\{s,\mathbf{u},\mathbf{v}\}} \|\mathbf{X} - s\mathbf{u}\mathbf{v}^T\|_F^2 = \sum_i (\mathbf{x}_{(i)} - su_i\mathbf{v}^T)^2 \tag{5}$$

or,

$$\min_{\{s,\mathbf{u},\mathbf{v}\}} \|\mathbf{X} - s\mathbf{u}\mathbf{v}^T\|_F^2 = \sum_j (\mathbf{x}_j - sv_j\mathbf{u})^2 \tag{6}$$

- (5): regress rows $\mathbf{x}_{(i)}$ of $\mathbf{X}$ on $\mathbf{v}$
- (6): regress columns $\mathbf{x}_j$ of $\mathbf{X}$ on $\mathbf{u}$
- alternating least squares (Gabriel and Zamir, 1979)

Some motivations to modify SVD

- Consider rank-1 approximation $s\mathbf{u}\mathbf{v}^T$
- Goals: want the “singular vectors" to have some additional properties:
  - Sparseness: some entries of **u** or **v** or **both** are exactly zero
    - “variable” selection: column or row or both
    - better interpretation, ...
  - Continuity: entries of **u** or **v** or **both** are continuous
    - functional data analysis: one-way or two-way
    - robustness, ...
- How can we modify SVD to achieve such goals?

Roughness regularization: function estimation/smoothing

- $\{x_i, y_i\}$, $x_i \in R$
- $y_i = f(x_i) + \epsilon_i$, $i = 1, \ldots, n$
- $f(x)$: smooth function
  - yield curve, growth curve, ...
  - time-varying intensity function for non-homogeneous Poisson arrival process, ...
- penalized sum of squares

$$\sum_{i=1}^{n} \{y_i - f(x_i)\}^2 + \lambda \int \{f''(x)\}^2 dx$$

- penalized splines

Sparsity regularization: feature selection

- $\{x_i, y_i\}$, $x_i \in R^p$
- $y_i = x_i^T \beta + \epsilon_i$, $i = 1, \ldots, n$
  - linear model with a large number of predictors
  - even with $p >> n$
- penalized sum of squares

$$\sum_{i=1}^{n} \{y_i - x_i^T \beta\}^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- Enforce sparsity of β, i.e. entries of 0

A regularization framework to modify SVD

- The core for SVD is the following optimization problem,

$$\min_{\{s,\mathbf{u},\mathbf{v}\}} \|\mathbf{X} - s\mathbf{u}\mathbf{v}^T\|_F^2 \tag{7}$$

  - Viewed as a bilinear least squares problem
  - Solved via alternating least squares

- How about adding some regularization (or penalization) in (7)?

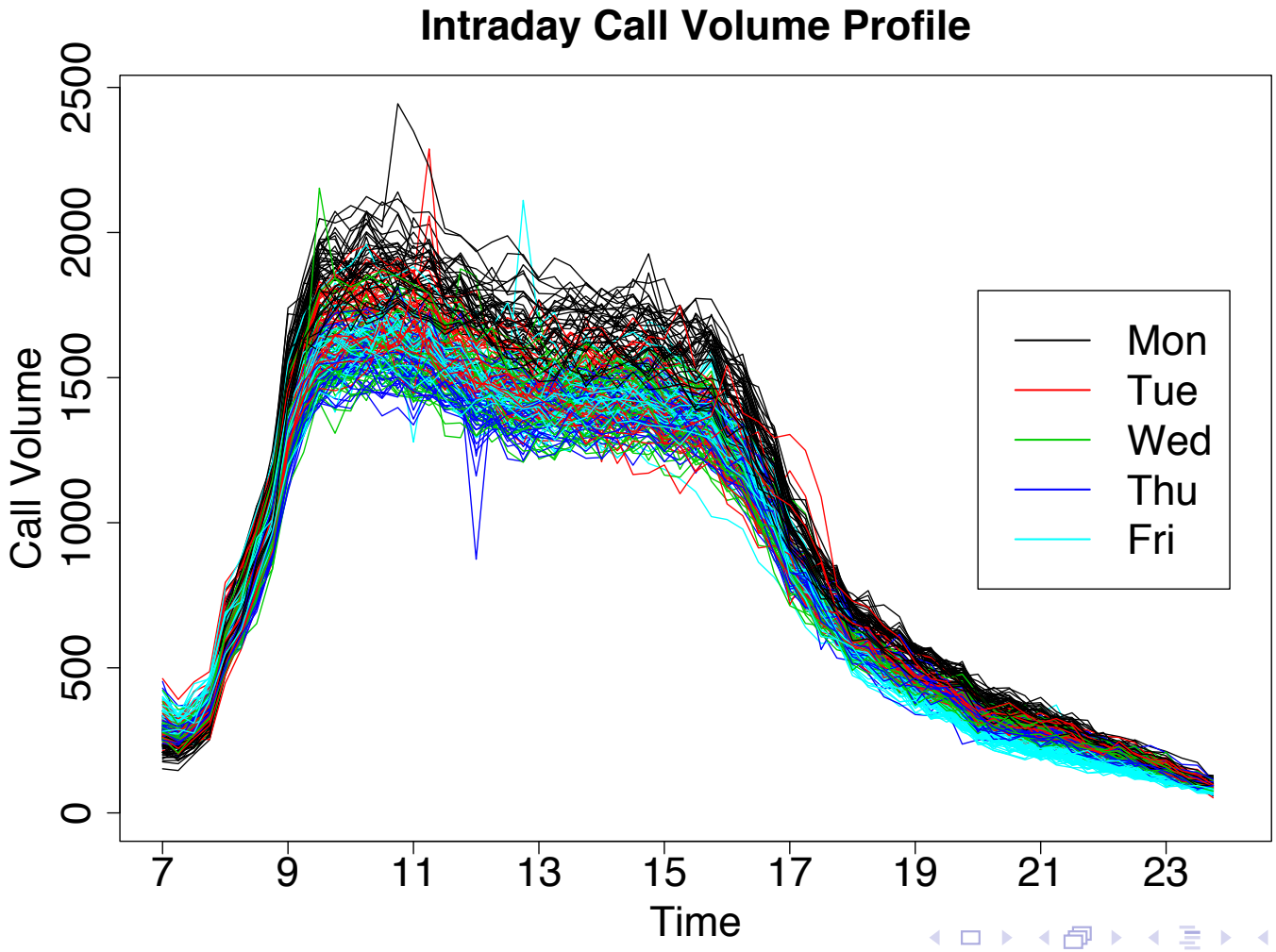
- Instead of (7), minimize a penalized version

$$\|\mathbf{X} - s\mathbf{u}\mathbf{v}^T\|_F^2 + P_{\lambda_1,\lambda_2}(s,\mathbf{u},\mathbf{v}) \tag{8}$$

  - two regularization parameters are allowed
  - sparsity-inducing penalty or smoothness penalty or a hybrid
  - form of penalty flexible; some make more sense than others

Functional data analysis (FDA)

- FDA: statistical analysis of populations of functions or curves
- Example: call center arrival data



Functional principal component analysis (FPCA): main workhorse

- Rao (1958) - growth curves, Ramsay and Silverman (2005)
- $X(\cdot)$: random function, observed repeatedly and as a whole
- α: smoothing penalty parameter
- To find the jth FPC $\gamma_j(\cdot)$, Rice and Silverman (1991) maximize

$$\frac{\text{var}(\int \gamma X) - \alpha \int \gamma''^2}{\int \gamma^2} \tag{9}$$

subject to $\int \gamma \hat{\gamma}_k = 0$ for $k < j$, where $\hat{\gamma}_k$ is the estimated kth FPC

- Silverman (1996) maximizes

$$\frac{\text{var}(\int \gamma X)}{\int \gamma^2 + \alpha \int \gamma''^2} \tag{10}$$

subject to $\int \gamma \hat{\gamma}_k + \alpha \int \gamma'' \hat{\gamma}''_k = 0$ for $k < j$.

A new approach (Huang, Shen and Buja, 2008)

- Rice-Silverman and Silverman: maximize some variance
- How about the low-rank-approximation view?
- Consider a sample of functional data $x_i(\cdot)$, $i = 1, \ldots, n$, observed at common grid points $t_1, \ldots, t_m$
- The observed data are $x_{ij} = x_i(t_j)$, stored in a matrix $\mathbf{X} = (x_{ij})$
- Suppose $\mathbf{X}$ is column centered
- Reminder: our general regularization framework

$$\|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|_F^2 + P_{\lambda_1,\lambda_2}(\mathbf{u}, \mathbf{v})$$

- Note that the scale s is absorbed into either $\mathbf{u}$ or $\mathbf{v}$.

Penalization and transformation invariance

- Following FDA, assume underlying smooth function $\gamma(\cdot)$ such that $v_j = \gamma(t_j)$
- To estimate $\mathbf{v}$, propose to minimize

$$\|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|_F^2 + \alpha\, \mathbf{u}^T\mathbf{u}\, \mathbf{v}^T\mathbf{\Omega}\mathbf{v}, \tag{11}$$

  - $\mathbf{\Omega}$: roughness penalty matrix
  - for equi-spaced t_j, could be the second difference matrix:

$$\mathbf{v}^T\mathbf{\Omega}\mathbf{v} = \sum_{j=2}^{m-1} (v_{j+1} - 2v_j + v_{j-1})^2$$

  - could be carefully chosen for $\gamma(\cdot)$ to be spline interpolation of $\mathbf{v}$

Penalization and transformation invariance

- (11) is equivalent to maximizing

$$\frac{\mathbf{v}^T \mathbf{X}^T \mathbf{X} \mathbf{v}}{\mathbf{v}^T \mathbf{v} + \alpha \, \mathbf{v}^T \mathbf{\Omega} \mathbf{v}}, \tag{12}$$

  discrete version of the Silverman criterion expressed in (10).

- (11) is invariant under
  - scale transformations

$$\mathbf{u} \to c\mathbf{u}, \quad \mathbf{v} \to \mathbf{v}/c \tag{13}$$

  - scale transformation of the measurements:

$$\mathbf{X} \to c\mathbf{X}, \quad \mathbf{u} \to c\mathbf{u} \tag{14}$$

Failed attempts

- Criterion 1:

$$\|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|_F^2 + \alpha\, \mathbf{v}^T \mathbf{\Omega} \mathbf{v}, \tag{15}$$

  - not invariant under scale transformation (13)

- Criterion 2:

$$\|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|_F^2 + \alpha\, \frac{\mathbf{v}^T \mathbf{\Omega} \mathbf{v}}{\mathbf{v}^T \mathbf{v}}. \tag{16}$$

  - equivalent to maximizing

$$\frac{\mathbf{v}^T \mathbf{X}^T \mathbf{X} \mathbf{v} - \alpha\, \mathbf{v}^T \mathbf{\Omega} \mathbf{v}}{\mathbf{v}^T \mathbf{v}}, \tag{17}$$

    discrete version of the Rice-Silverman criterion expressed in (9)
  - invariant under the scale transformation (13)
  - not invariant under scale transformation of the measurements (14)
  - prefer Silverman over Rice-Silverman

Efficient computing: iteration

- Iterative algorithm:
  1. Initialize $\mathbf{v}$, for example first right singular vector of $\mathbf{X}$.
  2. Repeat until convergence:
     (a) $\mathbf{u} \leftarrow \mathbf{X}\mathbf{v}$,
     (b) $\mathbf{v} \leftarrow (\mathbf{I} + \alpha\,\mathbf{\Omega})^{-1}\mathbf{X}^T\mathbf{u}$,
     (c) $\mathbf{v} \leftarrow \mathbf{v}/\|\mathbf{v}\|$.
- When $\alpha = 0$, reduce to the power algorithm for computing PCA (Jolliffe, 2002)

Efficient computing: half-smoothing using SVD

- Denote $\mathbf{S}(\alpha) = (\mathbf{I} + \alpha\,\mathbf{\Omega})^{-1}$, $\tilde{\mathbf{v}} = \mathbf{S}^{-1/2}(\alpha)\,\mathbf{v}$, and $\tilde{\mathbf{X}} = \mathbf{X}\,\mathbf{S}^{1/2}(\alpha)$
- Then, (11) is equivalent to

$$\|\mathbf{X}\|_F^2 - \|\tilde{\mathbf{X}}\|_F^2 + \|\tilde{\mathbf{X}} - \mathbf{u}\tilde{\mathbf{v}}^T\|_F^2. \tag{18}$$

Because,

$$\|\mathbf{X}-\mathbf{u}\mathbf{v}^T\|_F^2+\alpha\,\mathbf{u}^T\mathbf{u}\,\mathbf{v}^T\mathbf{\Omega}\mathbf{v} = \|\mathbf{X}\|_F^2-2\mathbf{u}^T\mathbf{X}\mathbf{v}+\mathbf{u}^T\mathbf{u}\,\mathbf{v}^T(\mathbf{I}+\alpha\,\mathbf{\Omega})\mathbf{v},$$

which equals

$$\|\mathbf{X}\|_F^2 - \|\tilde{\mathbf{X}}\|_F^2 + \|\tilde{\mathbf{X}}\|_F^2 - 2\mathbf{u}^T\,\tilde{\mathbf{X}}\,\tilde{\mathbf{v}} + \mathbf{u}^T\mathbf{u}\,\tilde{\mathbf{v}}^T\tilde{\mathbf{v}}.$$

  - SVD of the row-half-smoothed $\tilde{\mathbf{X}}$ to get $\mathbf{u}$ and $\tilde{\mathbf{v}}$
  - then half-smooth $\tilde{\mathbf{v}}$ to obtain $\mathbf{v} = \mathbf{S}^{1/2}(\alpha)\tilde{\mathbf{v}}$

Efficient calculation of $\mathbf{S}(\alpha)$ and $\mathbf{S}^{1/2}(\alpha)$

- Eigen decomposition of $\boldsymbol{\Omega}$:

$$\boldsymbol{\Omega} = \boldsymbol{\Gamma}\boldsymbol{\Lambda}\boldsymbol{\Gamma}^T,$$

  where $\boldsymbol{\Gamma}$ is an orthogonal matrix containing the eigenvectors, and $\boldsymbol{\Lambda}$ is a diagonal matrix of the eigenvalues.

- Then,

$$\mathbf{S}(\alpha) = \boldsymbol{\Gamma}\left(\mathbf{I} + \alpha\,\boldsymbol{\Lambda}\right)^{-1}\boldsymbol{\Gamma}^T \quad \text{and} \quad \mathbf{S}^{1/2}(\alpha) = \boldsymbol{\Gamma}\left(\mathbf{I} + \alpha\,\boldsymbol{\Lambda}\right)^{-1/2}\boldsymbol{\Gamma}^T.$$

- Minimization of $\|\tilde{\mathbf{X}} - \mathbf{u}\tilde{\mathbf{v}}^T\|_F^2$ is equivalent to minimizing $\|\mathbf{X}\boldsymbol{\Gamma}(\mathbf{I} + \alpha\boldsymbol{\Lambda})^{-1/2} - \mathbf{u}\bar{\mathbf{v}}^T\|_F^2$ where $\bar{\mathbf{v}} = \boldsymbol{\Gamma}^T\tilde{\mathbf{v}}$.
- Finally, $\mathbf{v} = \mathbf{S}^{1/2}(\alpha)\tilde{\mathbf{v}} = \boldsymbol{\Gamma}(\mathbf{I} + \alpha\boldsymbol{\Lambda})^{-1/2}\bar{\mathbf{v}}$.

Parameter selection: cross-validation

- For fixed $\mathbf{u}$, $\mathbf{v}$ can be derived through a smoothing step
  - $\mathbf{v} \leftarrow (\mathbf{I} + \alpha\,\mathbf{\Omega})^{-1}\mathbf{X}^T\mathbf{u}$
  - response: $\mathbf{X}^T\mathbf{u}$
  - smoothing (or hat) matrix: $\mathbf{S}(\alpha) = (\mathbf{I} + \alpha\,\mathbf{\Omega})^{-1}$
- Similar to the smoothing spline case, can rigorously derive Leave-one-column cross-validation:

$$\mathrm{CV}(\alpha) = \frac{1}{m}\sum_{j=1}^{m} \frac{\left[\{(\mathbf{I} - \mathbf{S}(\alpha))(\mathbf{X}^T\mathbf{u})\}_{jj}\right]^2}{\left(1 - \{\mathbf{S}(\alpha)\}_{jj}\right)^2}, \tag{19}$$

- The same holds for Generalized CV.
- Efficient computation of CV/GCV.
- Rice/Silverman: leave-one-row CV, no computational shortcut such as (19).

Para. selection: connection with smooth. splines

- Suppose $\mathbf{X}$ has only one row, denoted by $\mathbf{y}^T$, then $\mathbf{u} = 1$ due to norm constraint and identifiability.
- Then (11) becomes

$$\|\mathbf{y} - u\mathbf{v}\|^2 + \alpha\, u^2 \mathbf{v}^T \mathbf{\Omega} \mathbf{v} = \|\mathbf{y} - \mathbf{v}\|^2 + \alpha\, \mathbf{v}^T \mathbf{\Omega} \mathbf{v},$$

the penalized least squares criterion for smoothing splines.

- In general, denote

$$\bar{\mathbf{y}} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \dots \\ \mathbf{x}_m \end{pmatrix}, \ \bar{\mathbf{X}} = \begin{pmatrix} \mathbf{u} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{u} & \dots & \mathbf{0} \\ \dots & \dots & \dots & \dots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{u} \end{pmatrix}, \ \mathbf{\Omega}_{v|u} = \alpha \|\mathbf{u}\|^2 \mathbf{\Omega},$$

where $\mathbf{x}_j$: j-th column of $\mathbf{X}$, $\bar{\mathbf{y}}$: size $mn \times 1$, $\bar{\mathbf{X}}$: size $mn \times m$.

Parameter selection: CV and GCV

- The penalized sum of squares (11) equals

$$\| \bar{\mathbf{y}} - \bar{\mathbf{X}}\mathbf{v} \|^2 \; + \; \mathbf{v}^T \, \Omega_{v|u} \, \mathbf{v} \, . \tag{20}$$

- Then, leaving out the jth column of $\mathbf{X}$ is equivalent to deleting the jth block of $\bar{\mathbf{y}}$ and the corresponding rows of $\bar{\mathbf{X}}$.
- Prediction error sum of squares:

$$\|\mathbf{u}\hat{v}_j^{(-j)} - \mathbf{x}_j\|^2 = \mathbf{x}_j^T\mathbf{x}_j - \frac{(\mathbf{x}_j^T\mathbf{u})^2}{\|\mathbf{u}\|^2} + \frac{\left(\|\mathbf{u}\|\hat{v}_j - \frac{\mathbf{u}^T\mathbf{x}_j}{\|\mathbf{u}\|}\right)^2}{(1-\mathbf{S}_{jj})^2}, \tag{21}$$

  where $\mathbf{v}_j = \{\mathbf{S}(\alpha)\mathbf{X}^T\mathbf{u}\}_{jj}$, and $\mathbf{v}_j^{(-j)}$ minimizes (20) when deleting the jth block of $\bar{\mathbf{y}}$ and the corresponding rows of $\bar{\mathbf{X}}$.
- The first two items on the RHS are irrelevant

Natural cubic spline (NCS) interpolation

- To extract the underlying function $\gamma(\cdot)$, propose to minimize, with respect to u_i and $\gamma(\cdot)$, the penalized sum of squares,

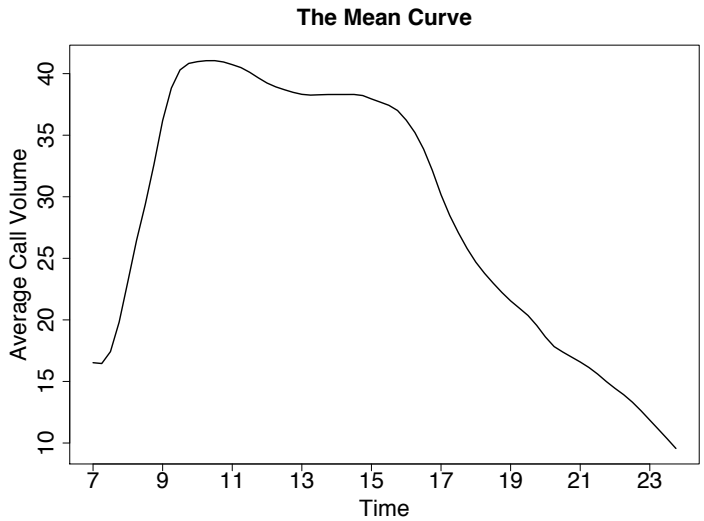
$$\sum_{i=1}^{n}\sum_{j=1}^{m}\{x_{ij} - u_i\gamma(t_j)\}^2 + \alpha\left(\sum_{i=1}^{n} u_i^2\right)\int\{\gamma''(t)\}^2\,dt, \quad (22)$$

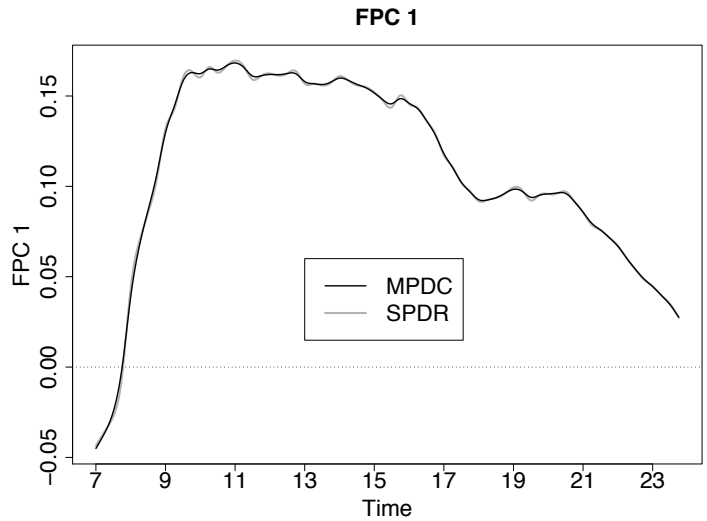
  subject to $\int\{\gamma''(t)\}^2\,dt < \infty$.
- Define

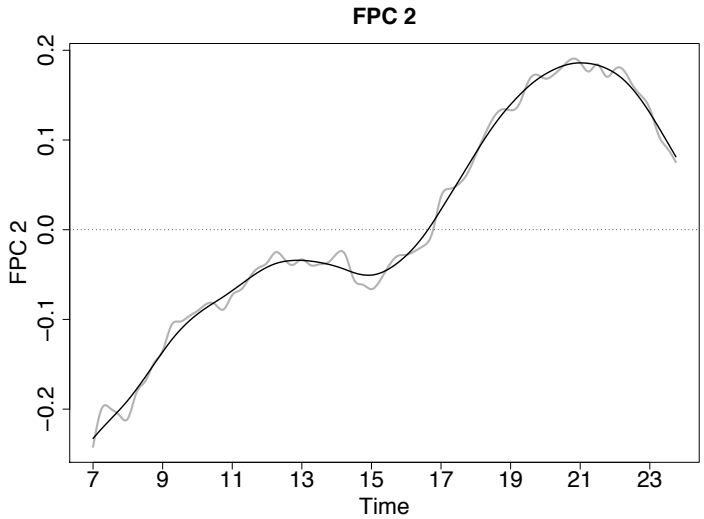
$$\mathbf{\Omega} = QR^{-1}Q^T$$

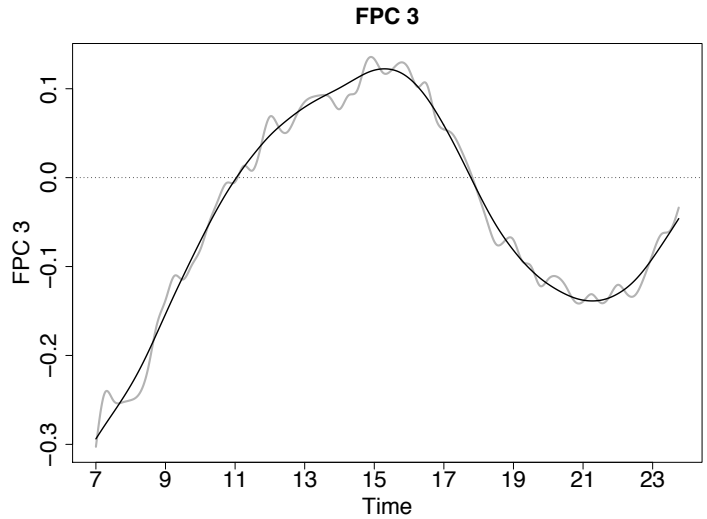
  for some banded matrices Q and R (Green and Silverman, 1994)
- Suppose $\hat{\mathbf{v}}$ minimizes the discretized problem (11) with the penalty matrix $\mathbf{\Omega}$
- Then, the $\hat{\gamma}(\cdot)$ optimizing (22) is the NCS with knots at t_j, and $\hat{\gamma}(t_j) = \hat{v}_j$.

Call center arrival data









Call center arrival data: message

- Our approach:
  - leave-out-one-column CV, computational shortcut
  - multiple smoothing parameters allowed
  - $\alpha = 0.44, 25.63, 38.44$ for first three FPCs
- The approach in Silverman (1996):
  - leave-out-one-row CV, no computational shortcut
  - single smoothing parameter
  - $\alpha = 0.13$ for first three FPCs
  - longer to compute: 1 minute vs 2 seconds

Simulation study: multiple smooth parameters matter

- The data generating model:

$$X_{ij} = u_{i1} v_1(t_j) + u_{i2} v_2(t_j) + \epsilon_{ij}, \quad i = 1, \ldots, n; \;\; j = 1, \ldots, m, \tag{23}$$

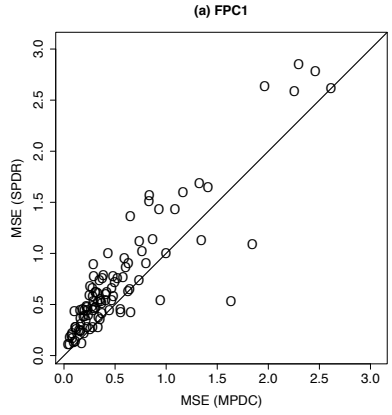
  - $u_{i1} \overset{i.i.d.}{\sim} N(0, \sigma_1^2)$, $u_{i2} \overset{i.i.d.}{\sim} N(0, \sigma_2^2)$, and $\epsilon_{ij} \overset{i.i.d.}{\sim} N(0, \sigma^2)$;
  -

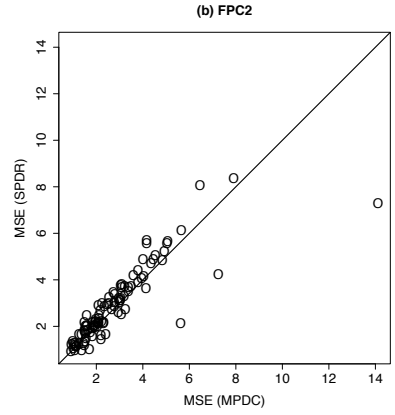
$$v_1(t) = \frac{1}{s_1}\{t + \sin(\pi t)\} \quad \text{and} \quad v_2(t) = \frac{1}{s_2}\cos(3\pi t),$$

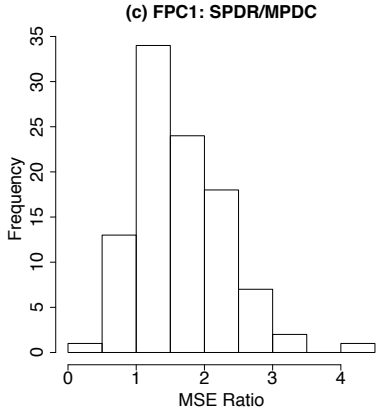
    where s_1 and s_2: normalizing constants.

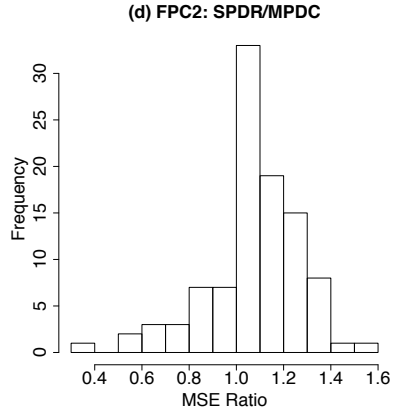
- Parameters: $n = m = 101$, $\sigma_1 = 20$, $\sigma_2 = 10$, $\sigma = 4$.
- 101 grid points t_j equally spaced in $[-1, 1]$.
- Compare mean squared errors (MSE) over the 101 grid points for each FPC.

Simulation study: multiple smooth parameters matter









	FPC 1				FPC 2			
	$Q1$	Median	Mean	$Q3$	$Q1$	Median	Mean	$Q3$
SPDR/MPDC	1.17	1.51	1.64	2.03	1.01	1.08	1.07	1.20

FDA: main messages

- invariance under scale transformation of the measurements
- support Silverman (1996) over Rice and Silverman (1991)
- efficient computing algorithm
- naturally incorporate spline smoothing
- suggest CV/GCV for parameter selection
- allow different amount of smoothing for different FPC

Two-way Functional Data Analysis via Two-way Regularized SVD

July 2016

Outline

Some motivations to modify SVD

- Consider rank-1 approximation $s\mathbf{u}\mathbf{v}^T$
- Goals: want the “singular vectors" to have some additional properties:
  - Continuity: entries of **u** or **v** or **both** are continuous
    - functional data analysis: one-way or two-way
    - robustness, ...
  - Sparseness: some entries of **u** or **v** or **both** are exactly zero
    - “variable” selection: column or row or both
    - better interpretation, ...
- How can we modify SVD to achieve such goals?

A regularization framework to modify SVD

- The core for SVD is the following optimization problem,

$$\min_{\{s,\mathbf{u},\mathbf{v}\}} \|\mathbf{X} - s\mathbf{u}\mathbf{v}^T\|_F^2 \qquad (1)$$

  - Viewed as a bilinear least squares problem
  - Solved via alternating least squares

- How about adding some regularization (or penalization) in (1)?

- Instead of (1), minimize a penalized version

$$\|\mathbf{X} - s\mathbf{u}\mathbf{v}^T\|_F^2 + P_{\lambda_1,\lambda_2}(s,\mathbf{u},\mathbf{v}) \qquad (2)$$

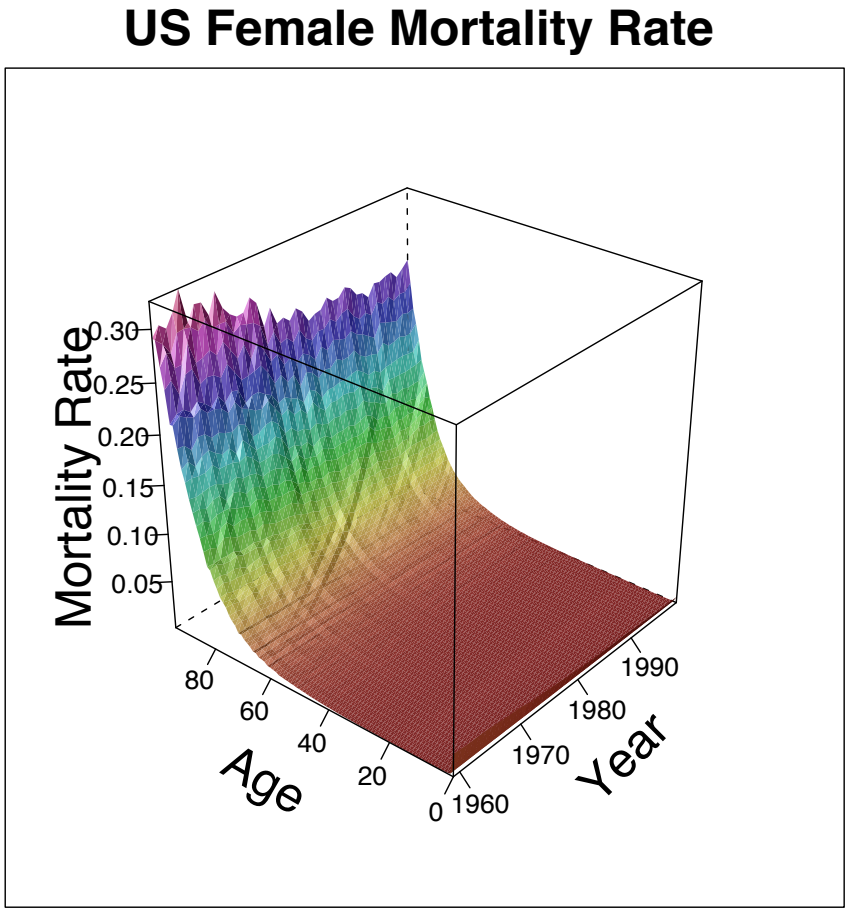
  - two regularization parameters are allowed
  - smoothness penalty or sparsity-inducing penalty or a hybrid

Two-way functional data analysis

- Deal with data that are functional in two ways
- $\mathbf{X} = (x_{i,j})_{i \in I, j \in J}$: both index domains I and J are structured with notions of smoothness
- As a comparison, traditional FDA: rows are considered as iid samples, realizations of curves
- Some examples

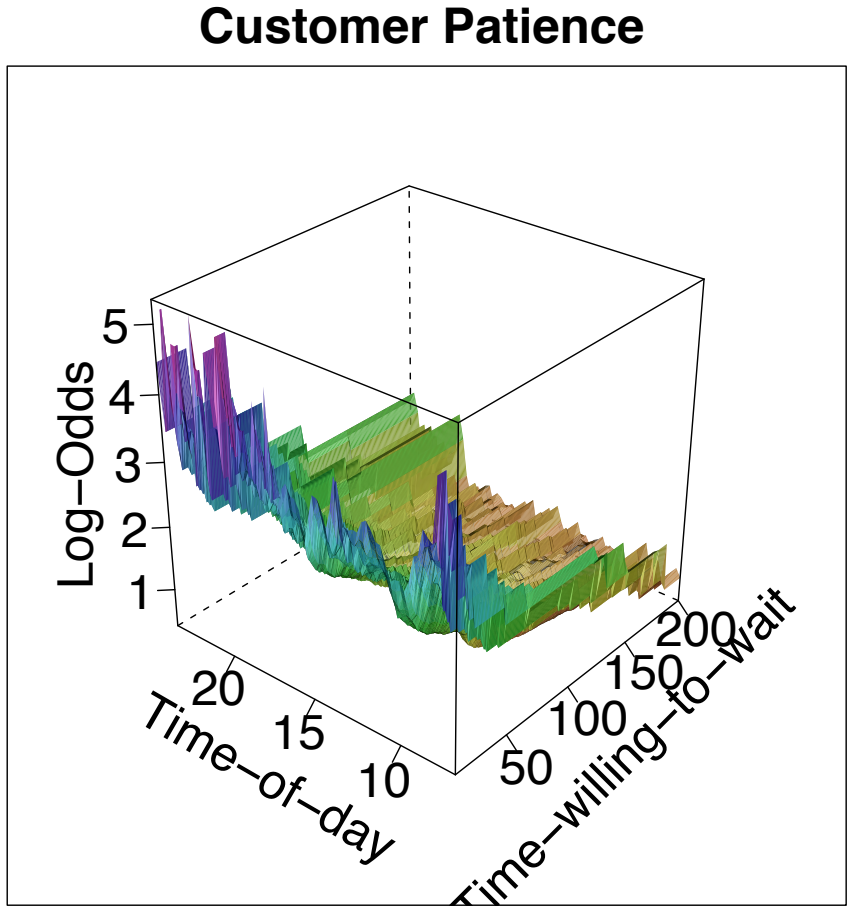
Example 1: Mortality rate

- Female mortality rates (US): *http* : *//www.mortality.org/*
- Rows: years from 1959 to 1999
- Columns: ages from 0 to 95



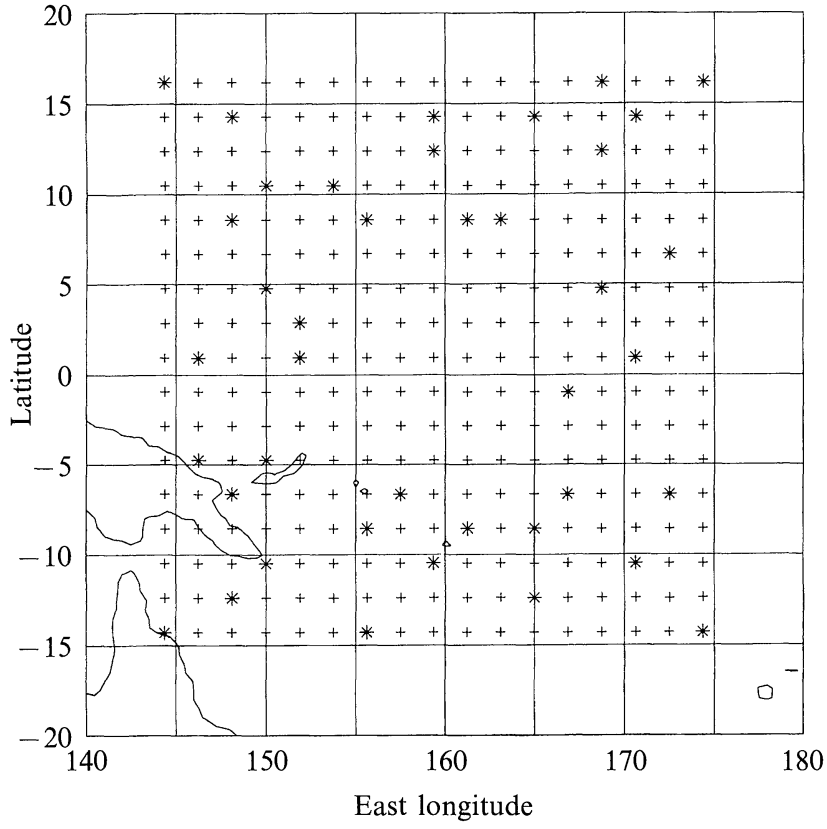
Example 2: Call center customer patience

- Customer patience at an Israeli call center
- Data: log-odds of time-willing-to-wait
- Survival analysis: censored due to getting service
- Rows: waiting times between 11 and 200 seconds
- Columns: quarter hours between 7am and midnight

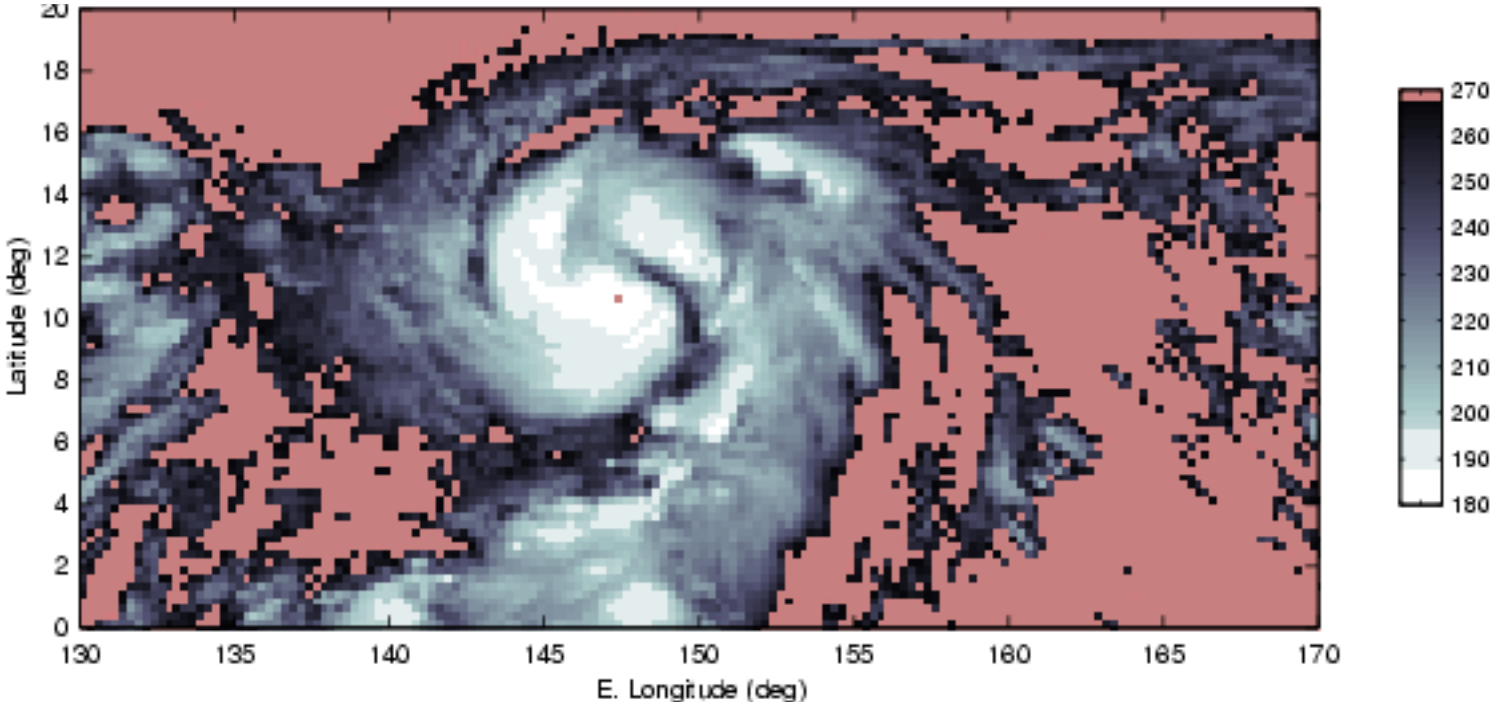


Example 3: Tropical wind

- East-west component of the wind velocity vector from a region over the tropical western Pacific ocean (Wikle and Cressie, 1999)
- Rows: a spatial resolution of 2 degrees in latitude(14°S-16°N) and longitude (145°E-175°E): 17 by 17 grid = 289 locations
- Columns: every 6 hours between November 1992

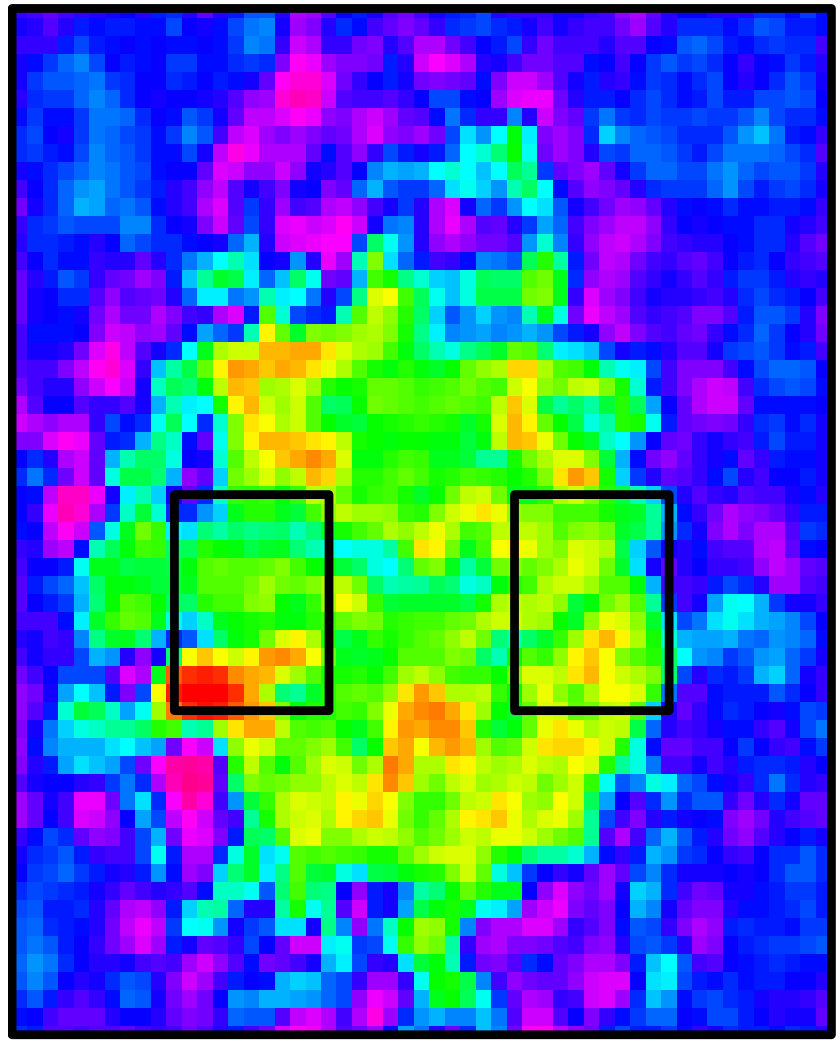


Example 3: Tropical wind



Example 4: Functional magnetic resonance imaging

- Functional images of one slice of the brain
- Rows: 53 by 60 voxels
- Columns: time points within two identical experiment cycles, each generating 100 images



Goal of Two-way FDA

- View the element x_{ij} as evaluation of an underlying function $X(\cdot, \cdot)$ on a rectangular grid of sampling points (y_i, z_j),
  - y_i $(i = 1, \ldots, n)$ from a domain $\mathcal{Y}$
  - z_j $(j = 1, \ldots, m)$ from a domain $\mathcal{Z}$
- Cannot rely on PCA and asymmetric treatment of row/column
- Led to the SVD which offers symmetric treatment
- Two-way FDA model:

$$X(y, z) = U_1(y)V_1(z) + U_2(y)V_2(z) + \cdots + U_r(y)V_r(z) + \epsilon(y, z)\,, \tag{3}$$

  - absorbed the singular value λ_k into $U_k(y)$ and/or $V_k(z)$,
  - the error is iid white noise
  - assume smooth $U_k(y)$ and $V_k(z)$ on their respective domains

Regularized SVD

- Roughness penalization
- Basis expansion
- Hybrid between penalization and basis expansion

Unpenalized LS for rank-one approximation

- Consider rank-one approximations to $\mathbf{X}$ as $\mathbf{u}\mathbf{v}^T$.
- Identified up to a scale factor:

$$\mathbf{u} \mapsto c\mathbf{u}\,, \quad \mathbf{v} \mapsto \mathbf{v}/c \quad (c \neq 0). \tag{4}$$

- The unpenalized LS criterion for rank-one approximations is

$$\mathcal{C}_0(\mathbf{u}, \mathbf{v}) \;=\; \|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|^2 \;=\; \|\mathbf{X}\|^2 - 2\mathbf{u}^T\mathbf{X}\mathbf{v} + \|\mathbf{u}\|^2\|\mathbf{v}\|^2. \tag{5}$$

- Can be cast as two conditional LS problems with solutions

$$\operatorname{argmin}_{\mathbf{u}} \mathcal{C}_0(\mathbf{u}, \mathbf{v}) \;=\; \frac{\mathbf{X}\mathbf{v}}{\|\mathbf{v}\|^2} \quad \text{and} \quad \operatorname{argmin}_{\mathbf{v}} \mathcal{C}_0(\mathbf{u}, \mathbf{v}) \;=\; \frac{\mathbf{X}^T\mathbf{u}}{\|\mathbf{u}\|^2}\,. \tag{6}$$

- Lead to the power algorithm

$$\mathbf{u} \leftarrow \mathbf{X}\mathbf{v}\,, \quad \mathbf{v} \leftarrow \mathbf{X}^T\mathbf{u}\,, \quad \text{followed by normalizations.} \tag{7}$$

Penalized LS for rank-one approximation

- Consider domain-specific penalty matrices $\mathbf{\Omega}_u$ and $\mathbf{\Omega}_v$
- Need to balance goodness-of-fit measure $\mathcal{C}_0(\mathbf{u}, \mathbf{v})$ against smoothness measures $\mathbf{u}^T \mathbf{\Omega}_u \mathbf{u}$ and $\mathbf{v}^T \mathbf{\Omega}_v \mathbf{v}$
- For now, absorb smoothing parameters α_u and α_v into the penalty matrices
- General penalized LS criterion:

$$\mathcal{C}(\mathbf{u}, \mathbf{v}) \;=\; \|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|^2 + \mathcal{P}(\mathbf{u}, \mathbf{v})\,. \tag{8}$$

- To determine the (???) $\mathcal{P}(\mathbf{u}, \mathbf{v})$, impose

$$\operatorname{argmin}_{\mathbf{u}} \mathcal{C}(\mathbf{u}, \mathbf{v}) \;\propto\; \mathbf{S}_u \mathbf{X} \mathbf{v} \quad \text{and} \quad \operatorname{argmin}_{\mathbf{v}} \mathcal{C}(\mathbf{u}, \mathbf{v}) \;\propto\; \mathbf{S}_v \mathbf{X}^T \mathbf{u}\,, \tag{9}$$

  where $\mathbf{S}_u = (\mathbf{I} + \mathbf{\Omega}_u)^{-1}, \quad \mathbf{S}_v = (\mathbf{I} + \mathbf{\Omega}_v)^{-1}$.
- Can prove that the unique form is

$$\mathcal{P}(\mathbf{u}, \mathbf{v}) \;=\; \mathbf{u}^T \mathbf{\Omega}_u \mathbf{u} \cdot \|\mathbf{v}\|^2 \;+\; \|\mathbf{u}\|^2 \cdot \mathbf{v}^T \mathbf{\Omega}_v \mathbf{v} \;+\; \mathbf{u}^T \mathbf{\Omega}_u \mathbf{u} \cdot \mathbf{v}^T \mathbf{\Omega}_v \mathbf{v}$$

Desirable properties of the criterion

- Scale invariance under (4):

$$\mathcal{C}(c\mathbf{u}, \mathbf{v}) = \mathcal{C}(\mathbf{u}, c\mathbf{v});$$

- Equivariance under rescaling of $\mathbf{X}$ and the fit $\mathbf{u}\mathbf{v}^T$:

$$\mathcal{C}(c\mathbf{u}, \mathbf{v}; c\mathbf{X}) = \mathcal{C}(\mathbf{u}, c\mathbf{v}; c\mathbf{X}) = c^2\,\mathcal{C}(\mathbf{u}, \mathbf{v}; \mathbf{X});$$

- For $\boldsymbol{\Omega}_u = 0$, the penalty specializes to

$$\|\mathbf{u}\|^2 \cdot \mathbf{v}^T \boldsymbol{\Omega}_v \mathbf{v},$$

  the one-way penalty of Silverman (1996) and Huang et al. (2008);

- The stationary equations of $\mathcal{C}(\mathbf{u}, \mathbf{v})$ involve smoothing with penalties $\boldsymbol{\Omega}_u$ and $\boldsymbol{\Omega}_v$:

$$\operatorname{argmin}_{\mathbf{u}} \mathcal{C}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{S}_u \mathbf{X}\mathbf{v}}{\mathbf{v}^T(\mathbf{I} + \boldsymbol{\Omega}_v)\mathbf{v}}, \quad \operatorname{argmin}_{\mathbf{v}} \mathcal{C}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{S}_v \mathbf{X}^T\mathbf{u}}{\mathbf{u}^T(\mathbf{I} + \boldsymbol{\Omega}_u)\mathbf{u}}. \tag{10}$$

Flawed penalized LS approaches

- $$\mathcal{C}_1(\mathbf{u},\mathbf{v}) \;=\; \|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|_F^2 + \mathbf{u}^T\Omega_u\mathbf{u} + \mathbf{v}^T\Omega_v\mathbf{v}\,. \qquad (11)$$
  - not scale invariant
- $$\mathcal{C}_2(\mathbf{u},\mathbf{v}) \;=\; \|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|_F^2 + \mathbf{u}^T\Omega_u\mathbf{u}\cdot\mathbf{v}^T\Omega_v\mathbf{v}\,. \qquad (12)$$
  - scale invariant, but does not specialize to one-way penalization
- $$\mathcal{C}_3(\mathbf{u},\mathbf{v}) \;=\; \|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|_F^2 + \mathbf{u}^T\Omega_u\mathbf{u}\cdot\|\mathbf{v}\|^2 + \|\mathbf{u}\|^2\cdot\mathbf{v}^T\Omega_v\mathbf{v}\,. \qquad (13)$$
  - scale invariant, specialize to Silverman's penalty
  - but the amount of smoothing for $\mathbf{u}$ depends on $\mathbf{v}$, and vice

Efficient computing: iteration

1. Initialize $\mathbf{v}$, for example first right singular vector of $\mathbf{X}$.

2. Repeat until convergence:
   (a) $\mathbf{u} \leftarrow (\mathbf{I} + \mathbf{\Omega}_u)^{-1}\mathbf{X}\mathbf{v}$,
   (b) $\mathbf{v} \leftarrow (\mathbf{I} + \mathbf{\Omega}_v)^{-1}\mathbf{X}^T\mathbf{u}$,
   (c) normalization.

Efficient computing: half-smoothing using SVD

- The penalized SVD based on $\mathcal{C}(\mathbf{u}, \mathbf{v})$ is a plain SVD in a nonstandard coordinate system.
- The new coordinates $\widetilde{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{v}}$ are

$$\mathbf{S}_u^{1/2}\,\widetilde{\boldsymbol{u}} \;=\; \mathbf{u} \quad \text{and} \quad \mathbf{S}_v^{1/2}\,\widetilde{\boldsymbol{v}} \;=\; \mathbf{v}\,. \tag{14}$$

  where the “half-smoothers" are

$$\mathbf{S}_u^{1/2} = (\mathbf{I} + \boldsymbol{\Omega}_u)^{-1/2}$$

$$\mathbf{S}_v^{1/2} = (\mathbf{I} + \boldsymbol{\Omega}_v)^{-1/2}$$

- Denote $\widetilde{\mathbf{X}} = \mathbf{S}_u^{1/2}\mathbf{X}\,\mathbf{S}_v^{1/2}$
- Then, $\mathcal{C}(\mathbf{u}, \mathbf{v})$ is equivalent to

$$\begin{aligned} \mathcal{C}(\mathbf{u}, \mathbf{v}) &= \|\mathbf{X}\|^2 \;-\; 2\widetilde{\boldsymbol{u}}^T\widetilde{\mathbf{X}}\widetilde{\boldsymbol{v}} \;+\; \|\widetilde{\boldsymbol{u}}\|^2 \;\cdot\; \|\widetilde{\boldsymbol{v}}\|^2 \\ &= \|\mathbf{X}\|^2 \;-\; \|\widetilde{\mathbf{X}}\|^2 \;+\; \|\widetilde{\mathbf{X}} - \widetilde{\boldsymbol{u}}\widetilde{\boldsymbol{v}}^T\|^2, \end{aligned}$$

Efficient computing: half-smoothing algorithm

1. half-smooth the data matrix according to $\widetilde{\mathbf{X}} = \mathbf{S}_u^{1/2} \mathbf{X} \mathbf{S}_v^{1/2}$,
2. obtain a plain SVD of the half-smoothed data matrix $\widetilde{\mathbf{X}}$, and
3. half-smooth the singular vectors according to $\mathbf{S}_u^{1/2} \widetilde{\boldsymbol{u}} = \mathbf{u}$ and $\mathbf{S}_v^{1/2} \widetilde{\boldsymbol{v}} = \mathbf{v}$.

Comments

- Potential application of efficient SVD algorithms (Golub and van Loan, 1996) for calculating the penalized SVD.
- However, the earlier iterative algorithm is critical
  - to identify an appropriate penalized criterion
  - to develop a cross-validation criterion for smoothing parameter selection

Notions of orthogonality and length

- The notions are nonstandard under penalization.
- For $\widetilde{\boldsymbol{u}}$ and $\widetilde{\boldsymbol{v}}$, the inner products and squared norms are Euclidean.
- While for $\mathbf{u}$ and $\mathbf{v}$, they are:

$$
\begin{aligned}
\langle\langle \mathbf{u}_1, \mathbf{u}_2 \rangle\rangle &= \widetilde{\boldsymbol{u}}_1^T \widetilde{\boldsymbol{u}}_2 = \mathbf{u}_1^T (\mathbf{I} + \boldsymbol{\Omega}_u) \mathbf{u}_2 , \\
\langle\langle \mathbf{v}_1, \mathbf{v}_2 \rangle\rangle &= \widetilde{\boldsymbol{v}}_1^T \widetilde{\boldsymbol{v}}_2 = \mathbf{v}_1^T (\mathbf{I} + \boldsymbol{\Omega}_v) \mathbf{v}_2 , \\
[\![\mathbf{u}]\!]^2 &= \widetilde{\boldsymbol{u}}^T \widetilde{\boldsymbol{u}} = \mathbf{u}^T (\mathbf{I} + \boldsymbol{\Omega}_u) \mathbf{u} , \\
[\![\mathbf{v}]\!]^2 &= \widetilde{\boldsymbol{v}}^T \widetilde{\boldsymbol{v}} = \mathbf{v}^T (\mathbf{I} + \boldsymbol{\Omega}_v) \mathbf{v} ,
\end{aligned}
$$

- The notion of norm extends another of Silverman's observations to the two-way case.

Smoothing parameter (bandwidth) selection

- Make bandwidths explicit as α_u and α_v in $\mathcal{C}(\mathbf{u}, \mathbf{v})$,

$$\begin{aligned}\mathcal{C}(\mathbf{u}, \mathbf{v}; \alpha_u, \alpha_v) &= \|\mathbf{X} - \mathbf{u}\mathbf{v}^T\|^2 + \mathbf{u}^T(\alpha_u \boldsymbol{\Omega}_u)\mathbf{u} \cdot \|\mathbf{v}\|^2 \\ &\quad + \|\mathbf{u}\|^2 \cdot \mathbf{v}^T(\alpha_v \boldsymbol{\Omega}_v)\mathbf{v} + \mathbf{u}^T(\alpha_u \boldsymbol{\Omega}_u)\mathbf{u} \cdot \mathbf{v}^T(\alpha_v \boldsymbol{\Omega}_v)\mathbf{v} ,\end{aligned} \tag{15}$$

- Denote the smoothers by, respectively,

$$\mathbf{S}_u(\alpha_u) = (\mathbf{I} + \alpha_u \boldsymbol{\Omega}_u)^{-1}, \qquad \mathbf{S}_v(\alpha_v) = (\mathbf{I} + \alpha_v \boldsymbol{\Omega}_v)^{-1}.$$

- Nest bandwidth selection inside each updating (or conditional optimization) step of the alternating algorithm, to avoid simultaneous selection of two bandwidths

Smoothing parameter (bandwidth) selection

- The alternating updates are

$$\mathbf{u} = \frac{\mathbf{S}_u(\alpha_u)\,\mathbf{X}\mathbf{v}}{\mathbf{v}^T(\mathbf{I}+\alpha_v\boldsymbol{\Omega}_v)\mathbf{v}} = \frac{\mathbf{S}_u(\alpha_u)}{1+\alpha_v\mathcal{R}_v(\mathbf{v})}\,\frac{\mathbf{X}\mathbf{v}}{\|\mathbf{v}\|^2}, \quad (16)$$

$$\mathbf{v} = \frac{\mathbf{S}_v(\alpha_v)\,\mathbf{X}^T\mathbf{u}}{\mathbf{u}^T(\mathbf{I}+\alpha_u\boldsymbol{\Omega}_u)\mathbf{u}} = \frac{\mathbf{S}_v(\alpha_v)}{1+\alpha_u\mathcal{R}_u(\mathbf{u})}\,\frac{\mathbf{X}^T\mathbf{u}}{\|\mathbf{u}\|^2}, \quad (17)$$

  where $\mathcal{R}_u(\mathbf{u}) = \mathbf{u}^T\boldsymbol{\Omega}_u\mathbf{u}/\|\mathbf{u}\|^2$ and $\mathcal{R}_v(\mathbf{v}) = \mathbf{v}^T\boldsymbol{\Omega}_v\mathbf{v}/\|\mathbf{v}\|^2$ are the plain Rayleigh quotients of $\boldsymbol{\Omega}_u$ and $\boldsymbol{\Omega}_v$.

- The unpenalized updates are

$$\mathbf{u} = \frac{\mathbf{X}\mathbf{v}}{\|\mathbf{v}\|^2}, \qquad \mathbf{v} = \frac{\mathbf{X}^T\mathbf{u}}{\|\mathbf{u}\|^2},$$

- Each updating: smoothing + shrinking

Cross-validation

- Analogously,

$$\mathrm{CV}_u(\alpha_u;\alpha_v) = \frac{1}{n}\sum_{i=1}^{n}\frac{(\hat{u}_i - \mathbf{x}_{(i)}\mathbf{v}/\|\mathbf{v}\|^2)^2}{\{1-[\mathbf{S}_u(\alpha_u)]_{ii}/(1+\alpha_v\mathcal{R}_v(\mathbf{v}))\}^2} \tag{18}$$

$$\mathrm{CV}_v(\alpha_v;\alpha_u) = \frac{1}{m}\sum_{j=1}^{m}\frac{(\hat{v}_j - \mathbf{x}_j^T\mathbf{u}/\|\mathbf{u}\|^2)^2}{\{1-[\mathbf{S}_v(\alpha_v)]_{jj}/(1+\alpha_u\mathcal{R}_u(\mathbf{u}))\}^2} \tag{19}$$

- The CV scores are based on residuals from the projections $\mathbf{Xv}$ and $\mathbf{X}^T\mathbf{u}$, not the data matrix $\mathbf{X}$.
- Can prove the above is equivalent to leaving-out-one-row and leaving-out-one-column of $\mathbf{X}$.
- Similar to the leaving-out-one-column in the one-way FDA case.

Efficient calculation

- Eigen decomposition of $\mathbf{\Omega}_u$ and $\mathbf{\Omega}_v$
- Efficient calculation for
  - $\mathbf{S}_u$, $\mathbf{S}_v$
  - $\mathbf{S}_u^{1/2}$, $\mathbf{S}_v^{1/2}$
  - GCV and CV scores
- Natural cubic spline (NCS) interpolation for both directions

A simulation study

- The data generating model:

$$X_{ij} = U_1^*(s_i)\, V_1^*(t_j) + U_2^*(s_i)\, V_2^*(t_j) + \epsilon_{ij}, \quad i = 1, \ldots, n; \quad j = 1, \ldots, m \tag{20}$$

where

$$\begin{aligned} U_1^*(s) &= \sin(2\pi s), & V_1^*(t) &= -3 + 8\exp(-4(t-0.25) \\ U_2^*(s) &= \sin(2\pi(s-0.25)), & V_2^*(t) &= -3 + 8\exp(-4(t-0.75) \end{aligned}$$

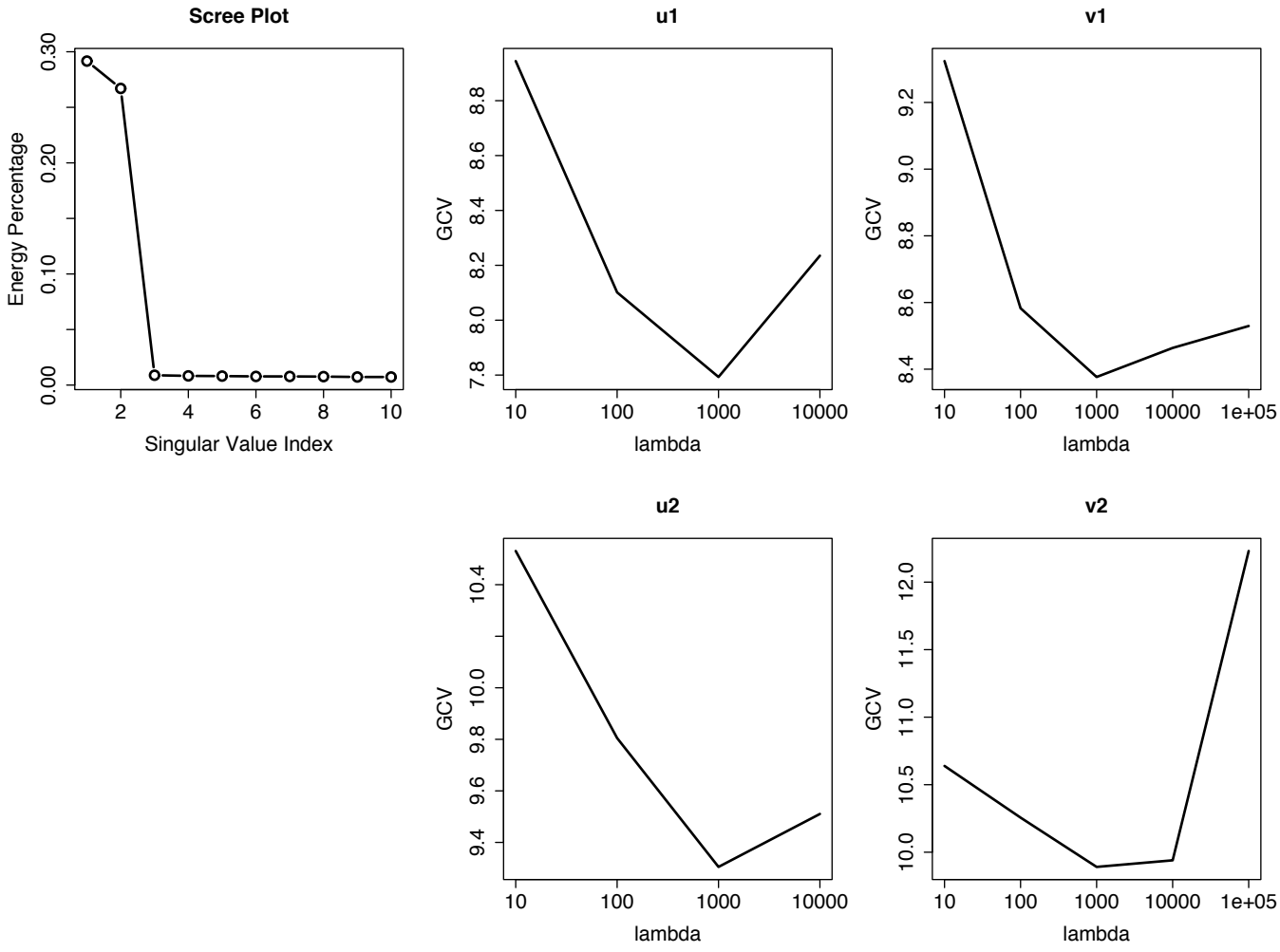
- s_i and t_j are each 201 equi-spaced points in $[0, 1]$;
- Parameters: $n = m = 201$, $\sigma = 3$ or 6.
- 100 simulations
- Apply SVD to

$$x_{ij}^* = U_1^*(s_i)\, V_1^*(t_j) + U_2^*(s_i)\, V_2^*(t_j)\,.$$

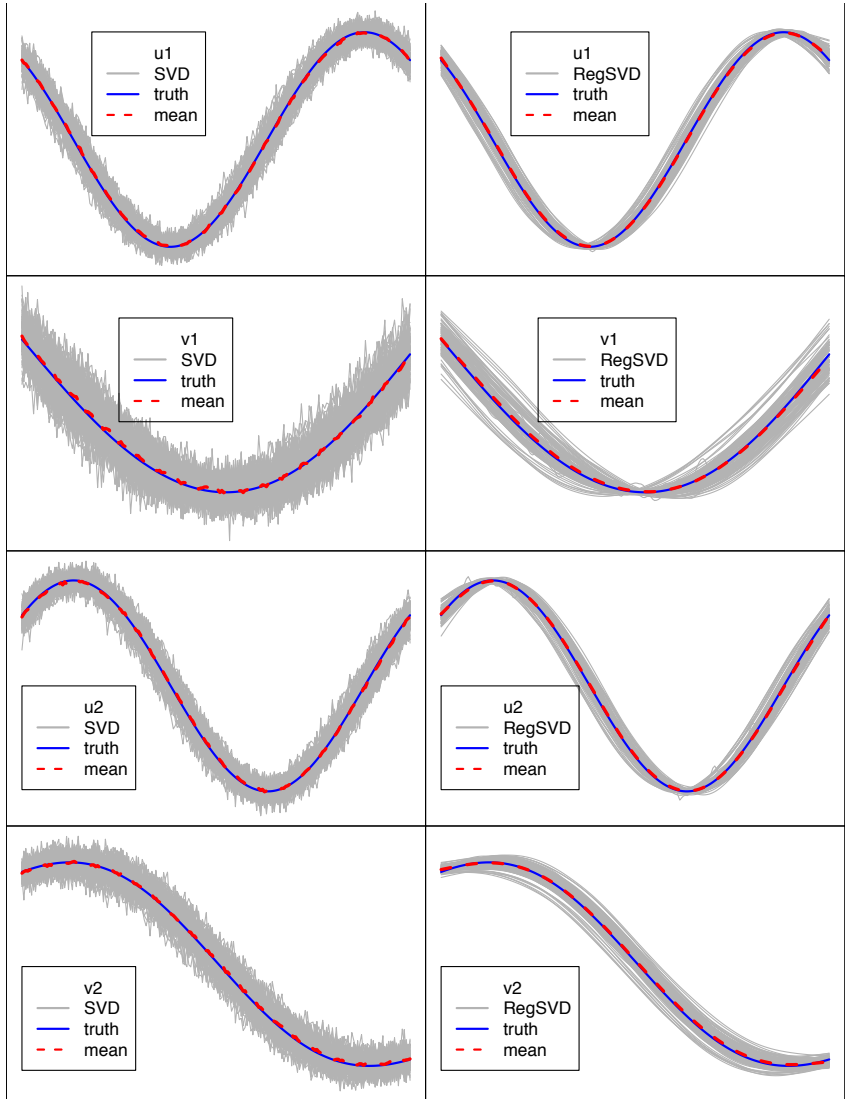
to obtain the true SVD

$$x_{ij}^* = d_1\, U_1(s_i)\, V_1(t_j) + d_2\, U_2(s_i)\, V_2(t_j)\,. \tag{21}$$

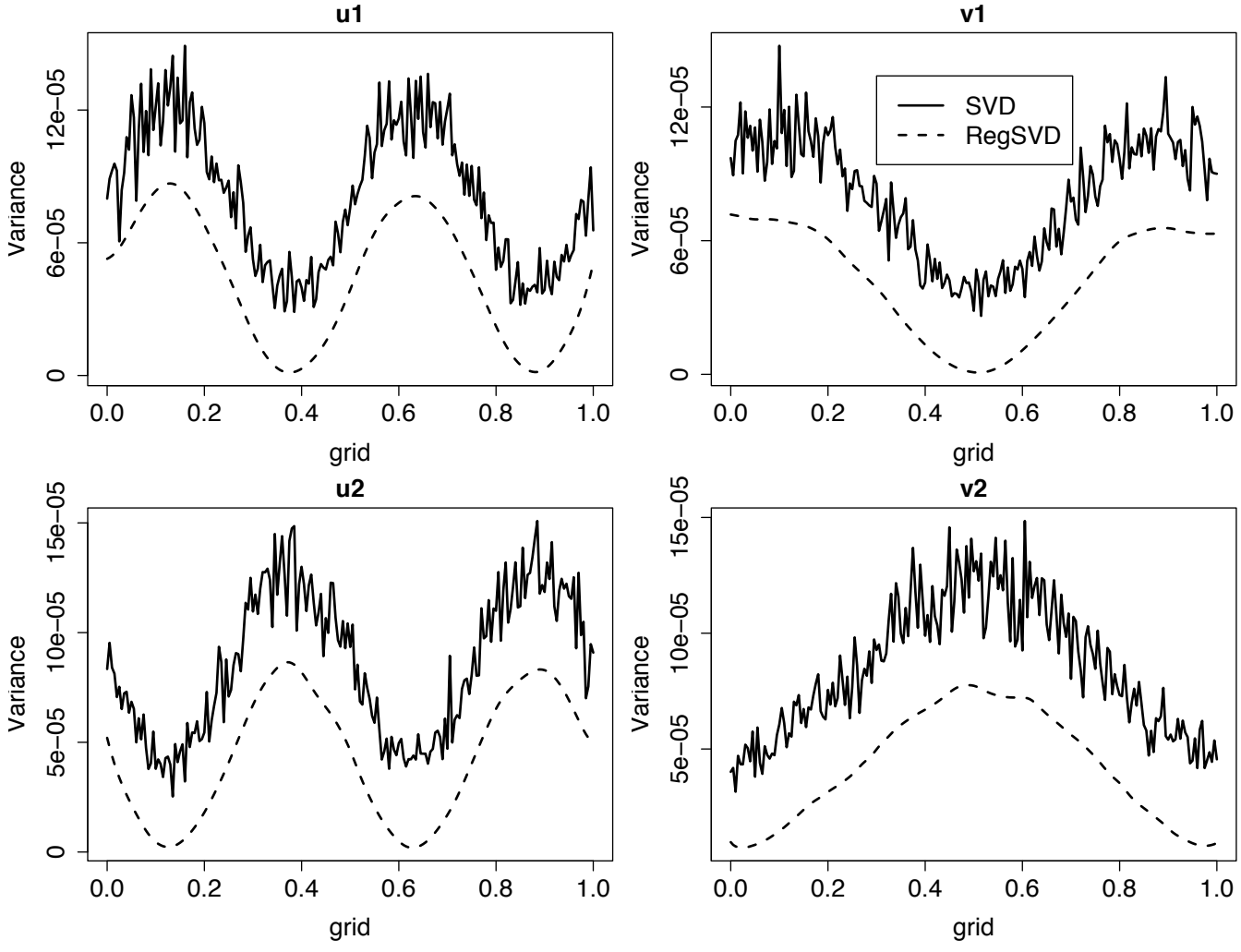
A simulation study: Scree Plot



A simulation study: SVD vs RegSVD Individual Curves



A simulation study: SVD vs RegSVD Variance (55% average reduction)



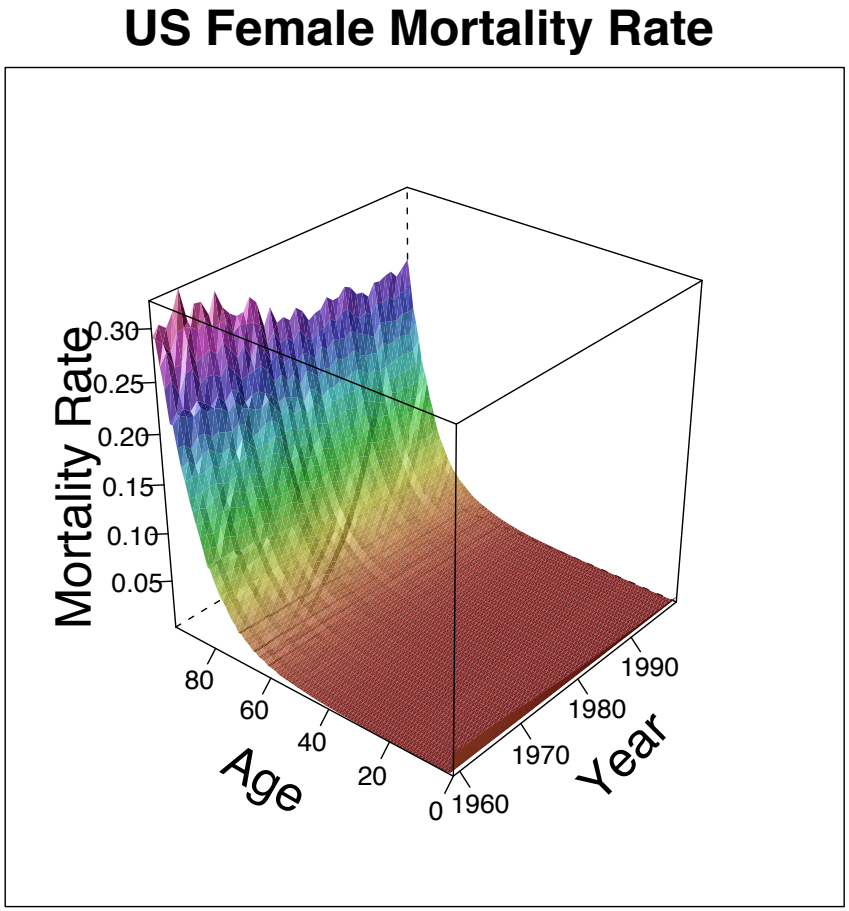
A simulation study: Comparison of Integrated Square Error

Methods	noise level σ	$\mathbf{u}_1$	$\mathbf{u}_2$	$\mathbf{v}_1$	$\mathbf{v}_2$
SVD	3	7.48 (0.89)	7.69 (0.81)	12.31 (2.76)	9.50 (1.18)
	6	8.08 (0.94)	9.26 (1.04)	15.33 (3.07)	11.94 (1.57)
sSVD	3	1.09 (0.03)	1.07 (0.03)	1.06 (0.04)	1.10 (0.04)
	6	1.38 (0.14)	1.38 (0.14)	1.64 (0.28)	1.56 (0.22)
uSVD	3	1.07 (0.03)	1.06 (0.02)	12.17 (2.70)	9.39 (1.17)
	6	1.22 (0.06)	1.19 (0.05)	14.49 (2.84)	11.29 (1.45)
vSVD	3	7.37 (0.88)	7.57 (0.80)	1.03 (0.03)	1.02 (0.03)
	6	7.65 (0.89)	8.74 (1.00)	1.30 (0.16)	1.24 (0.12)
rSVD-basis	3	1.05 (0.01)	1.08 (0.02)	1.10 (0.03)	1.10 (0.02)
	6	1.08 (0.05)	1.14 (0.05)	1.22 (0.09)	1.19 (0.06)

Benchmark: Integrated Square Error from rSVD

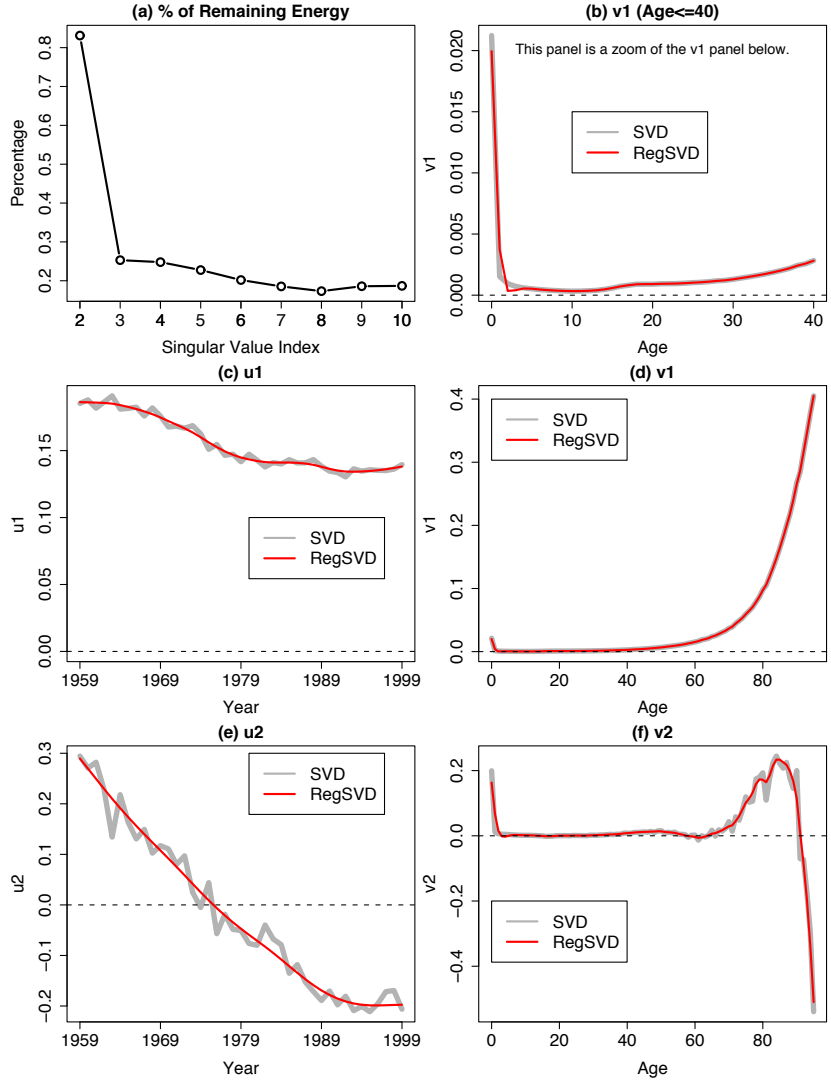
Example 1: Mortality rate

- Female mortality rates (US): *http* : *//www.mortality.org/*
- Rows: years from 1959 to 1999
- Columns: ages from 0 to 95
- Model:

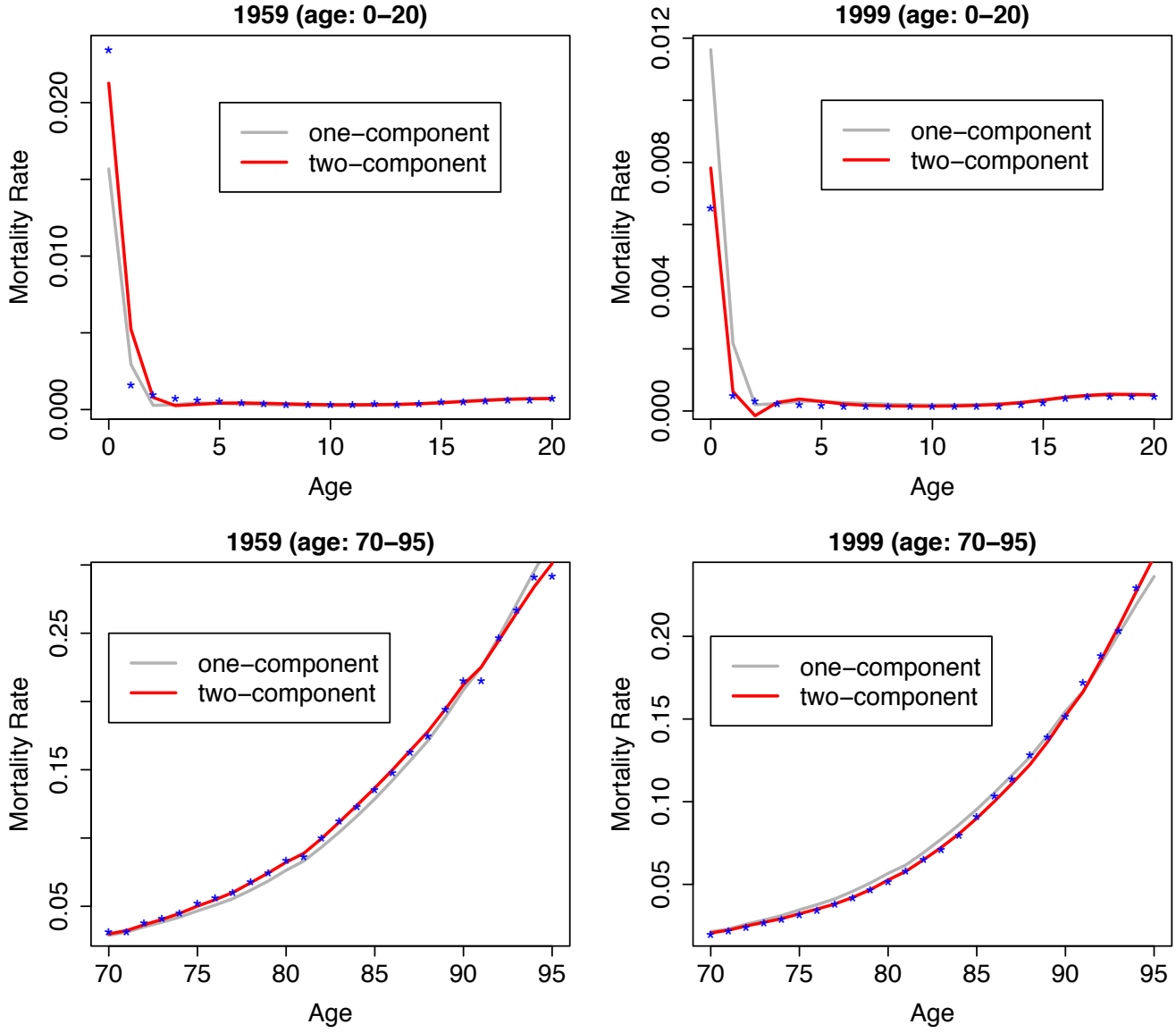


$$X(\text{Period}, \text{Age}) = d_1 U_1(\text{Period}) V_1(\text{Age}) + \dots$$

Example 1: Mortality rate



Example 1: Mortality rate, second FPC

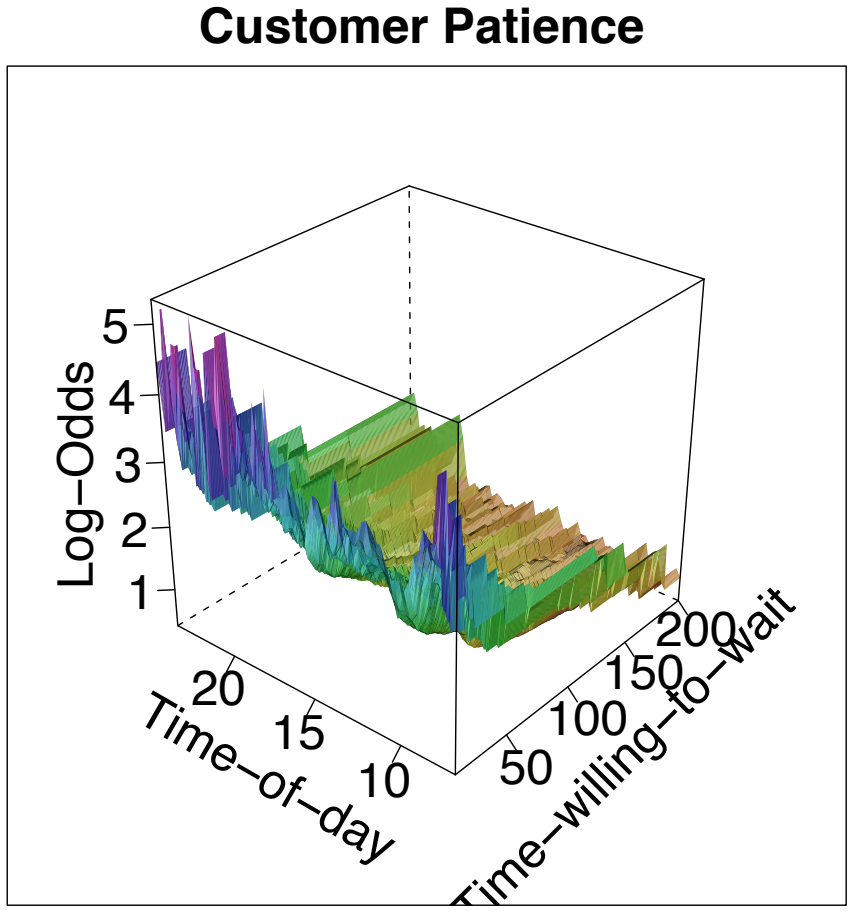


Example 1: Mortality rate

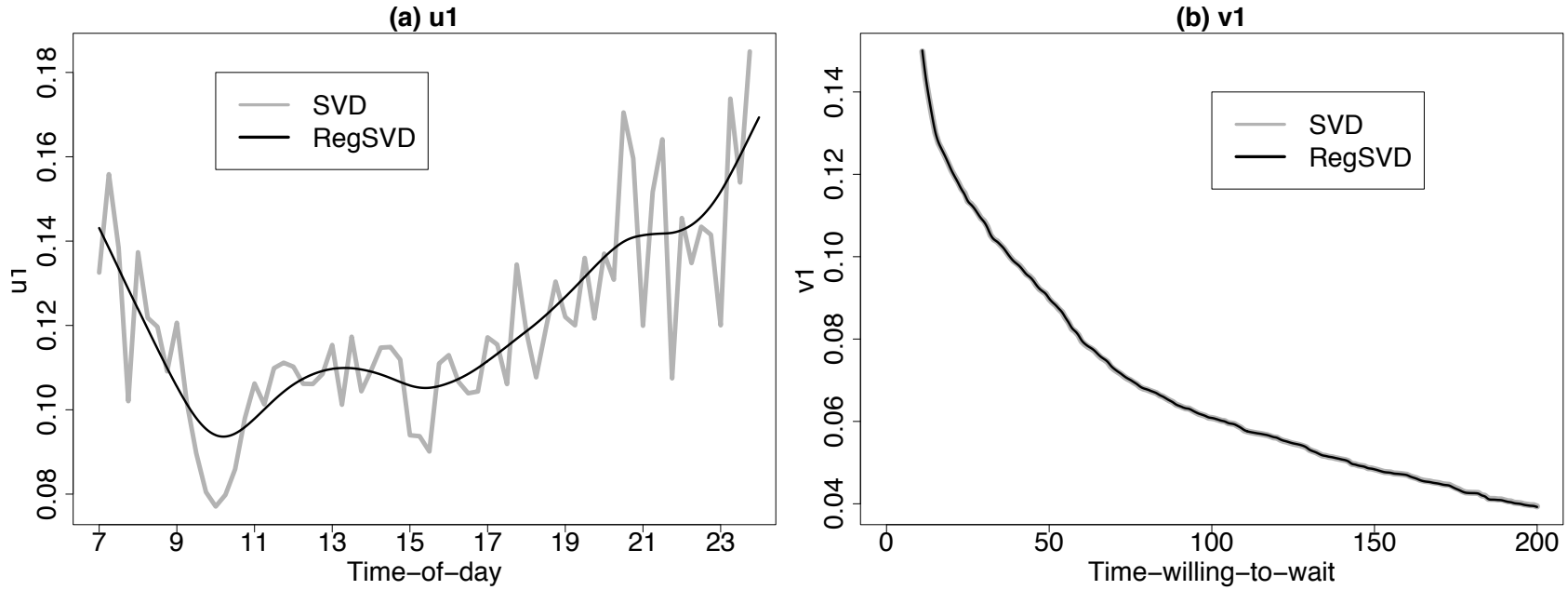
- The first pair $(\mathbf{u}_1, \mathbf{v}_1)$ explains about 99.87% of the total energy
- The second pair $(\mathbf{u}_2, \mathbf{v}_2)$ explains 83.11% of the remaining energy.
- $\mathbf{v}_1$: well-known pattern of mortality age curves
- $\mathbf{u}_1$: smooth average time trend across periods
- The second component focuses mainly on the early and late ages:
  - for ages < 2 and between 70 and 90, the mortality rate is higher in the 1960s and lower in the 1990s than what can be explained by the one component model.

Example 2: Call center customer patience

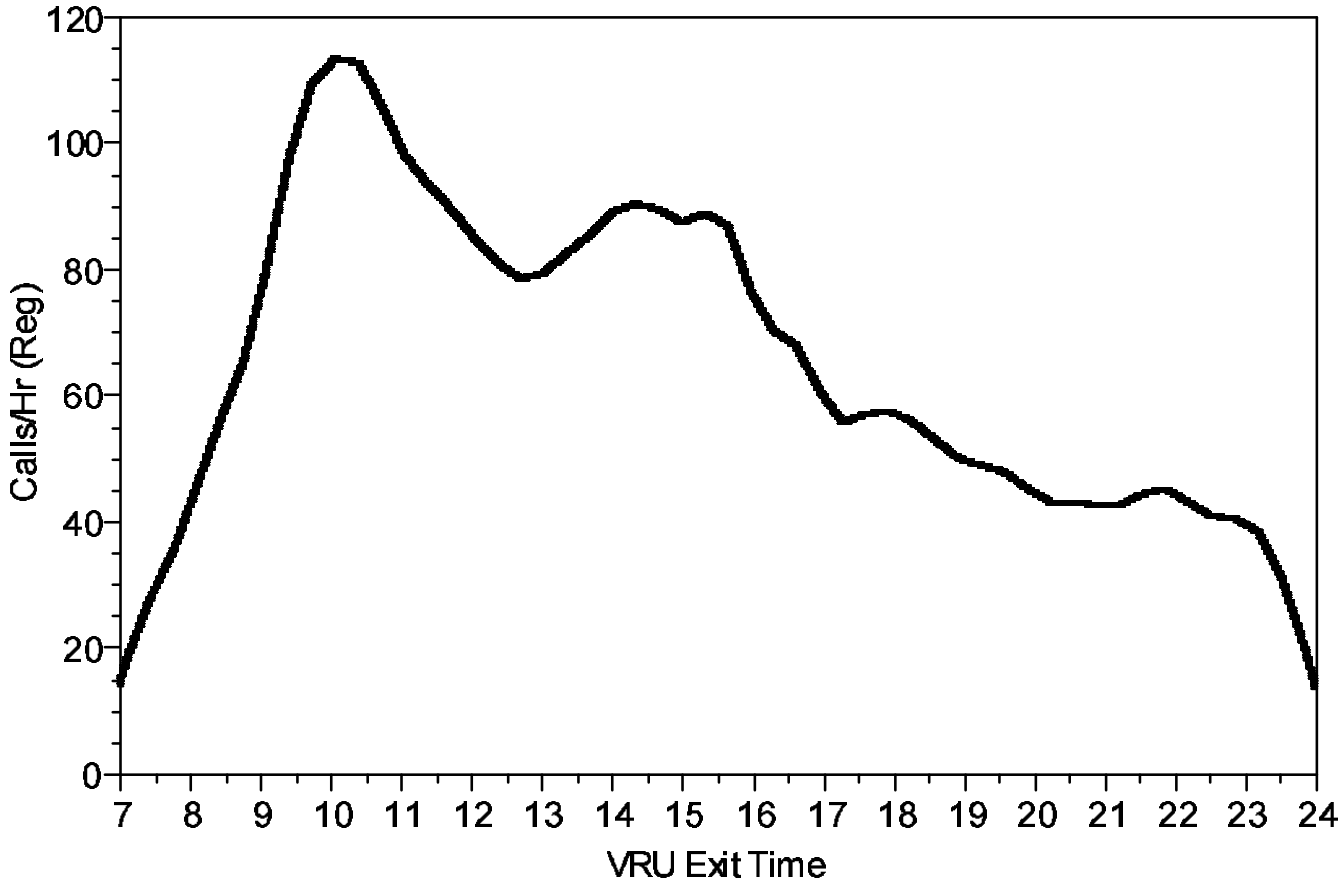
- Customer patience at an Israeli call center
- Data: log-odds of time-willing-to-wait
- Survival analysis: censored due to getting service
- Rows: waiting times between 11 and 200 seconds
- Columns: quarter hours between 7am and midnight



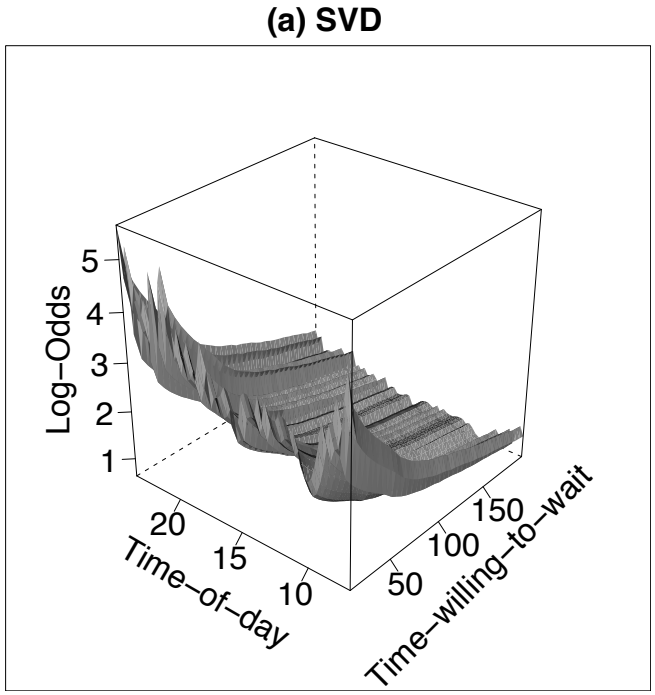
Example 2: Customer patience

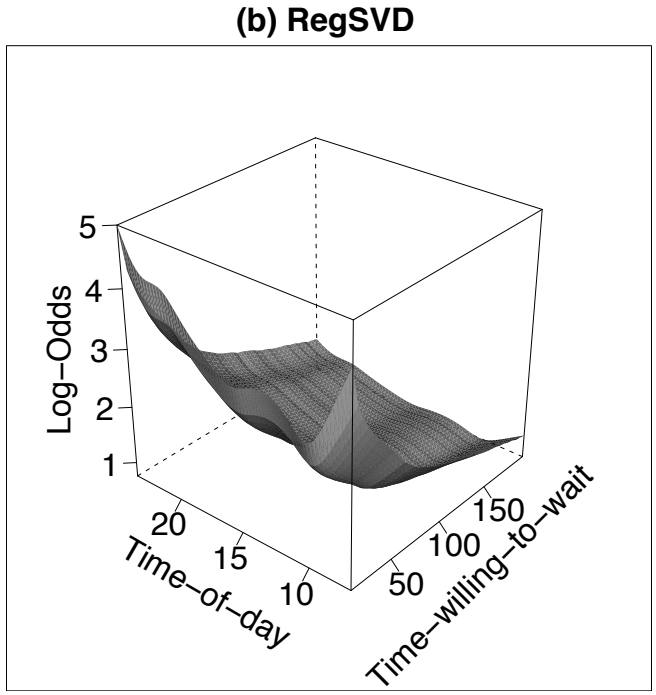


Example 2: Customer patience, call arrival rate



Example 2: Customer patience, surface





Example 2: Customer patience

- The first pair, $(\mathbf{u}_1, \mathbf{v}_1)$, explains about 98.93% of the total energy
- The second pair is not separated from the rest higher order pairs
- $U_1(t)$: reverse of the call arrival pattern; during peak hours (10am and 3pm), customers are less patient
- The model with one component: a proportional log-odds model,
  - $V_1(w)$: the baseline pattern
  - $d_1 U_1(t)$: the time-of-day specific scale adjustment
  - Customers seem to have the same aggregate behavior in terms of time-willing-to-wait at different times of day.

Two-way FDA: main messages

- two-way smooth regularization (spatial-temporal, imaging, ...)
- unique “optimal” two-way penalty
- efficient computing algorithm
- naturally incorporate spline smoothing
- suggest CV/GCV for parameter selection
- allow different amount of smoothing for different FPC
- shrinkage to some subspaces (periodic functions, ...)