

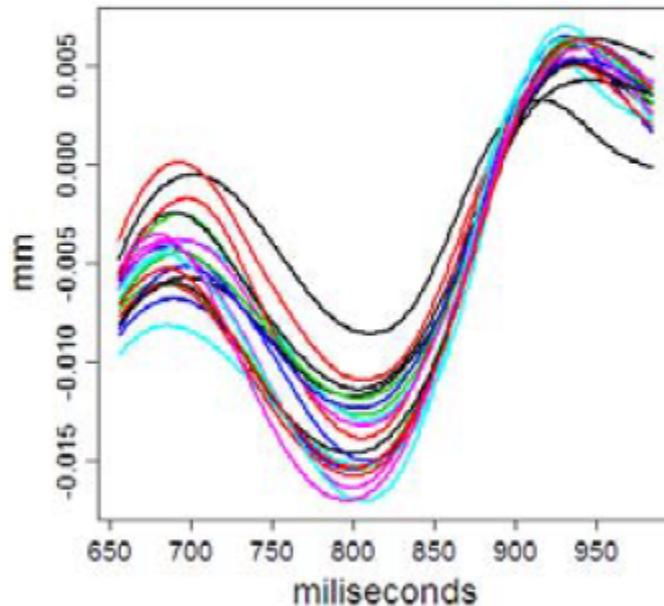


A JOURNEY THROUGH FUNCTIONAL DATA WITH R

Mehdi Maadooliat
Dept. of Mathematical and Statistical Sciences
Marquette University
Milwaukee, WI (USA)
Email: mehdi.maadooliat@mu.edu
URL: <http://www.mscs.mu.edu/~mehdi>

What is Functional Data?

20 replications

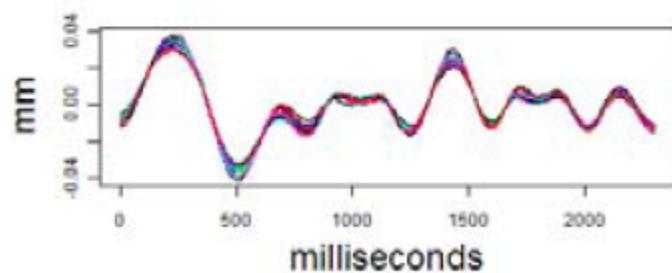
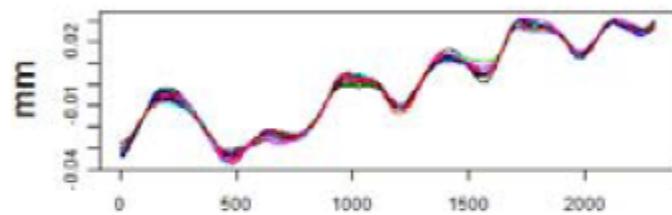


Functional data analysis involves repeated measures of the same process.

fda

What is Functional Data?

20 replications, 1401 observations within replications, 2 dimensions



Functional data is often
complex:

- often a large number of related quantities
- viewing each replication as a *single observation* can make the data easier to think about (once we have the right machinery)

- What are these data, anyway?

Some References

Three references for this course

- Ramsay & Silverman, 2005, “Functional Data Analysis”
- Ramsay & Silverman, 2002, “Applied Functional Data Analysis”
- Ramsay, Hooker & Graves, 2009, “Functional Data Analysis in R and Matlab”
- <https://fPCA.shinyapps.io/fPCA/>
(SHINY APP – IMPLEMENTED USING R)
- <http://faculty.bscb.cornell.edu/~hooker/ShortCourseHandout.pdf>
(MAINLY USED THIS ONE TO PREPARE THE CURRENT SLIDE)
- <https://www.stat.tamu.edu/~jianhua/paper/EJS-2008-218.pdf> (Functional PCA)
- <https://www.math.ntnu.no/emner/TMA4215/2008h/cubicsplines.pdf> (Some Theory)
- http://www.cs.cmu.edu/~aarti/Class/10701_Spring14/ (ICA)

Characteristics

- Data are measurements of smooth processes over time
- We usually do not want to make parametric assumptions about those processes.
- Often have multiple measurements of the same process
- We are interested in describing the variation of *processes*.
- Frequently, collected data have high resolution and low noise.
- Two (conflicting) goals: “fidelity” to y and “smoothness”
 - Fidelity, sum of squares: $\sum_{i=1}^n (y_i - x(t_i))^2$
 - How to quantify smoothness?

What Is Functional Data?

Functional data is multivariate data with an ordering on the dimensions. (Müller, (2006))

Key assumption is *smoothness*:

$$y_{ij} = x_i(t_{ij}) + \epsilon_{ij}$$

with t in a continuum (usually time), and $x_i(t)$ smooth
Functional data = the functions $x_i(t)$.

medical monitoring: EEG, ECG, fMRI, blood pressure ...

medical tests: HIV antibodies, flu screens...

biology: animal behavior (whale songs, fly egg-laying...)

environmental monitoring: weather, pollution, solar radiation, traffic ...

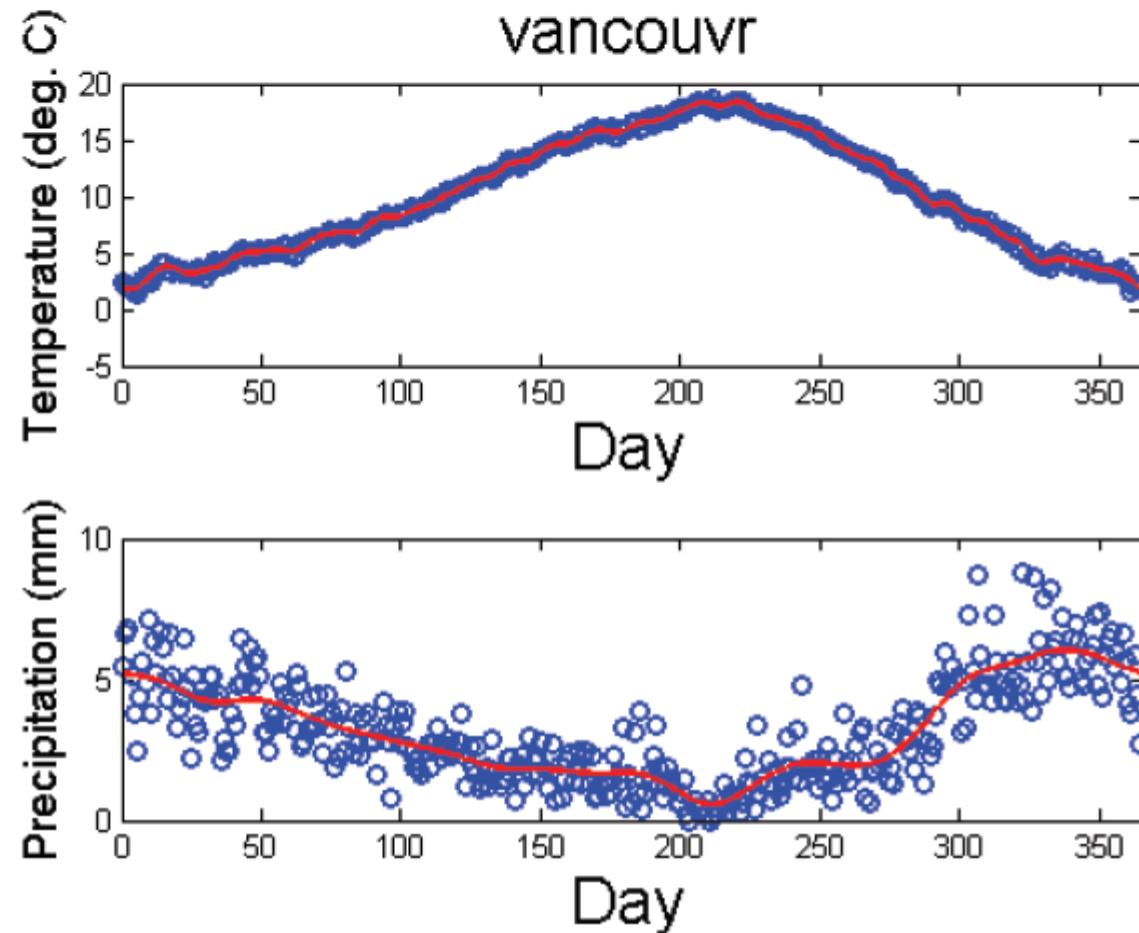


MARQUETTE
UNIVERSITY

Be The Difference.

Weather In Vancouver

Measure of climate: daily precipitation and temperature in Vancouver, BC averaged over 40 years.



fda



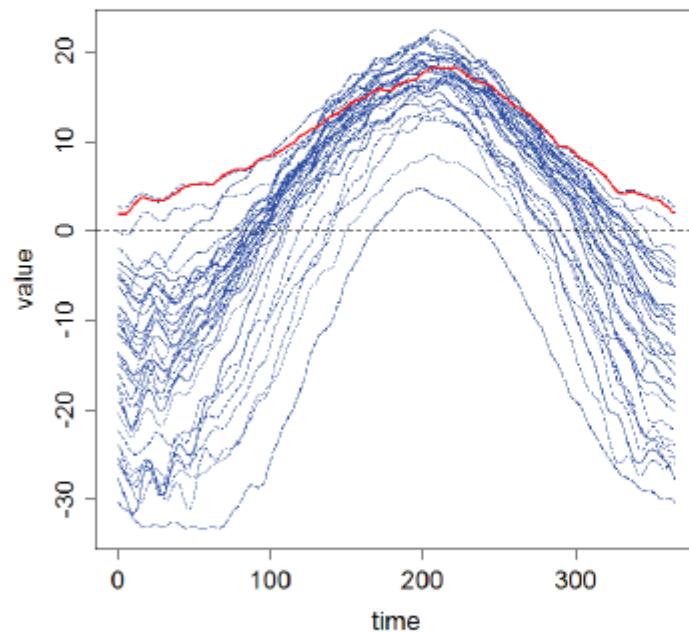
MARQUETTE
UNIVERSITY

Be The Difference.

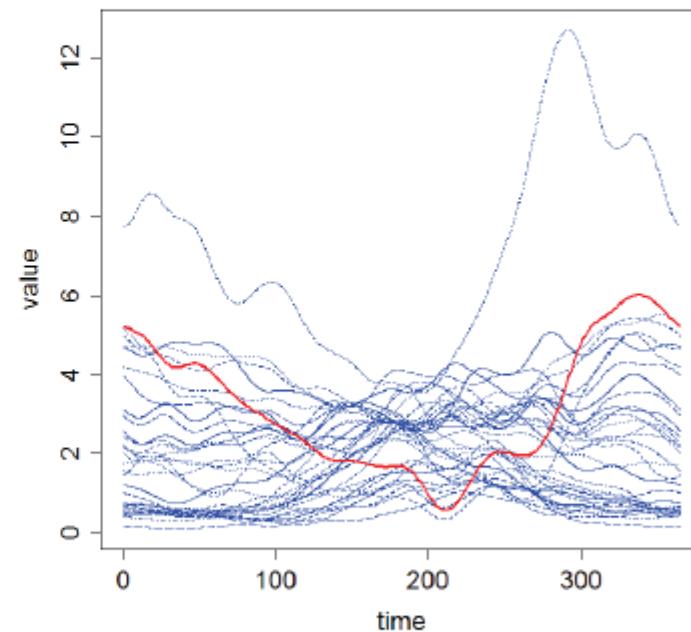
Canadian Weather Data

Average daily temperature and precipitation records in 35 weather stations across Canada (classical and much over-used)

Temperature



Precipitation



Interest is in variation in and relationships between smooth, underlying processes.

What Are The Challenges?

- Estimation of functional data from noisy, discrete observations.
- Numerical representation of infinite-dimensional objects
- Representation of variation in infinite dimensions.
- Description of statistical relationships between infinite dimensional objects.
- $n < p = \infty$, and use of smoothness.
- Measures of variation in estimates.

fda

Representing Functional Data

fda



From Discrete to Functional Data

Two problems/two methods

- 1 Representing non-parametric continuous-time functions.

- Basis-expansion methods:

$$x(t) = \sum_{i=1}^K \phi_i(t) c_i$$

for pre-defined $\phi_i(t)$ and coefficients c_i .

- 2 Reducing noise in measurements

- Smoothing penalties:

$$c = \operatorname{argmin}_c \sum_{i=1}^n (y_i - x(t_i))^2 + \lambda \int [Lx(t)]^2 dt$$

- $Lx(t)$ measures “roughness” of x
- λ a “smoothing parameter” that trades-off fit to the y_i and roughness; must be chosen.



MARQUETTE
UNIVERSITY

Be The Difference.

1. Basis Expansions

Univariate Case

fda

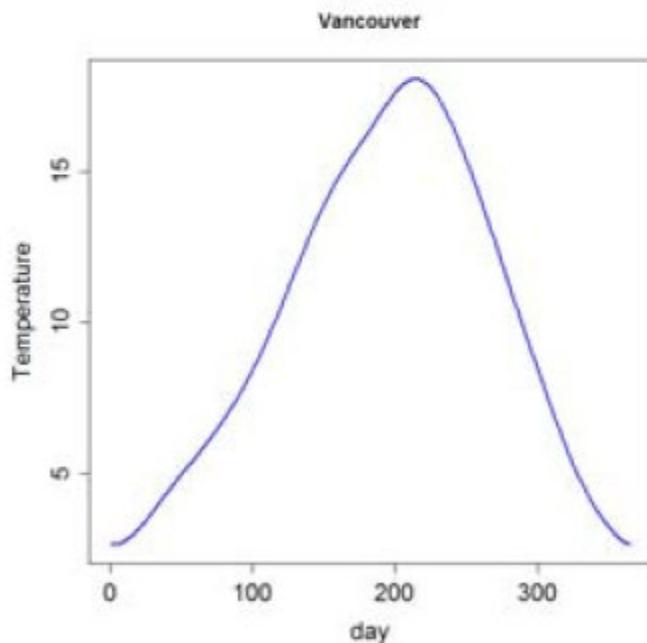
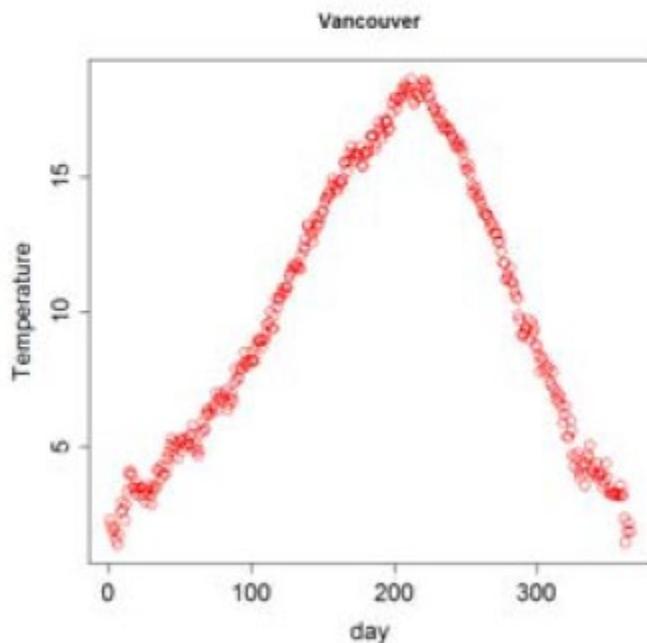
From Data To Functions

How do we go from

data

to

functions?



fda



MARQUETTE
UNIVERSITY

Be The Difference.

Basis Expansions

From multiple linear regression:

$$y_i = \beta_0 + x_{1i}\beta_1 + x_{2i}\beta_2 + \cdots + \epsilon_i$$

Or if there is curvature:

$$y_i = \beta_0 + \beta_1 t_i + \beta_2 t_i^2 + \beta_3 t_i^3 + \cdots + \epsilon_i$$

Consider only one record

$$y_i = x(t_i) + \epsilon_i$$

represent $x(t)$ as

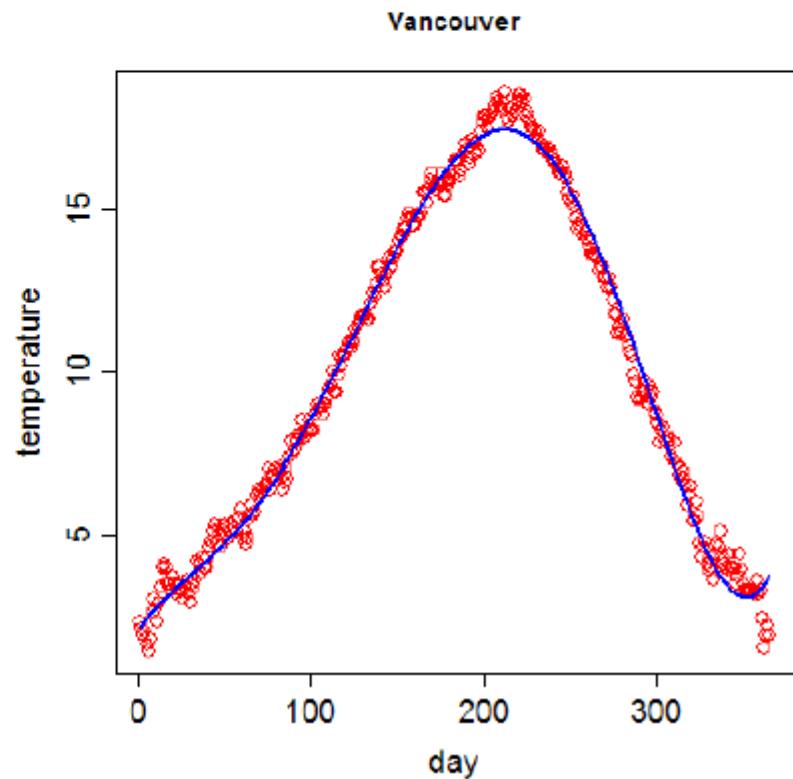
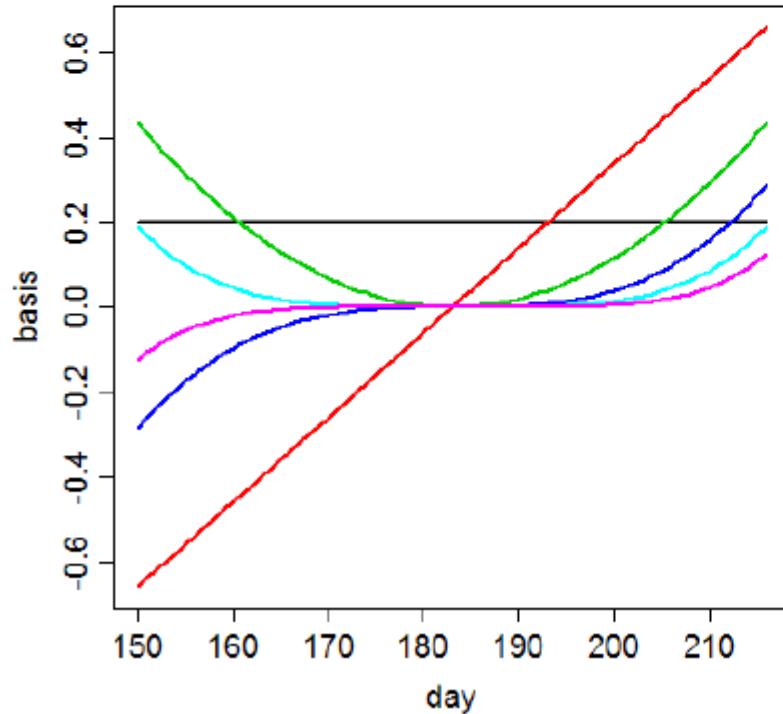
$$x(t) = \sum_{j=1}^K c_j \phi_j(t) = \Phi(t)\mathbf{c}$$

We say $\Phi(t)$ is a *basis system* for x .

$$x(t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \cdots$$

The Monomial Basis

$$\Phi(t) = (1, t, t^2, t^3, t^4, t^5, \dots)$$

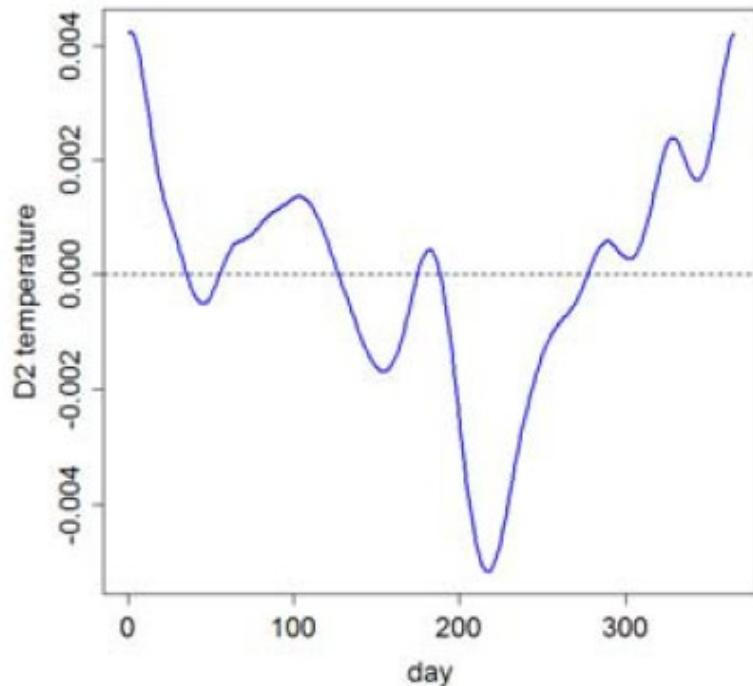


fda

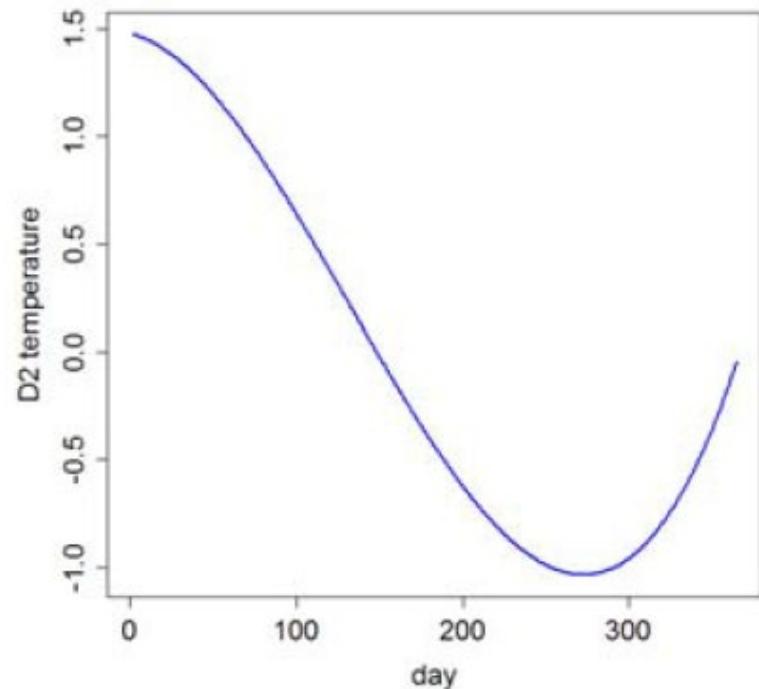
Problems with the Monomial Basis

Whereas the opposite happens in most real-world data:

Second Derivative



Estimate



fda

R code



MARQUETTE
UNIVERSITY

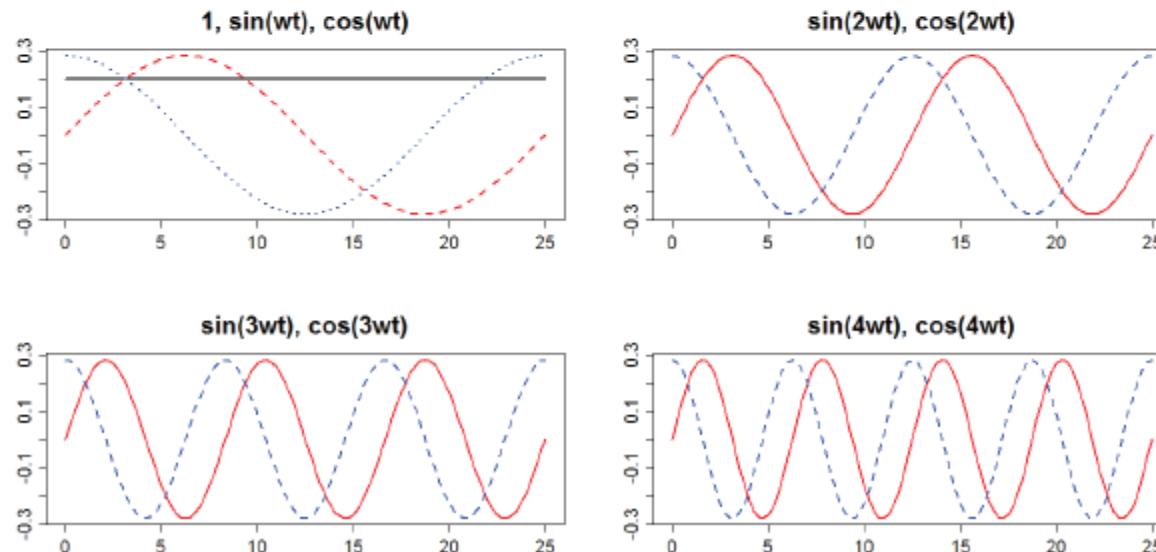
Be The Difference.

The Fourier Basis

- basis functions are sine and cosine functions of increasing frequency:

$$1, \sin(\omega t), \cos(\omega t), \sin(2\omega t), \cos(2\omega t), \dots$$
$$\sin(m\omega t), \cos(m\omega t), \dots$$

- constant $\omega = 2\pi/P$ defines the period P of oscillation of the first sine/cosine pair.



fda



MARQUETTE
UNIVERSITY

Be The Difference.

Advantages of Fourier Bases

- Only alternative to polynomials until the middle of the 20th century
- Excellent computational properties, especially if the observations are equally spaced.
- Natural for describing periodic data, such as the annual weather cycle

BUT representations are periodic; this can be a problem if the data are not.

Fourier basis is first choice in many fields, eg signal processing.

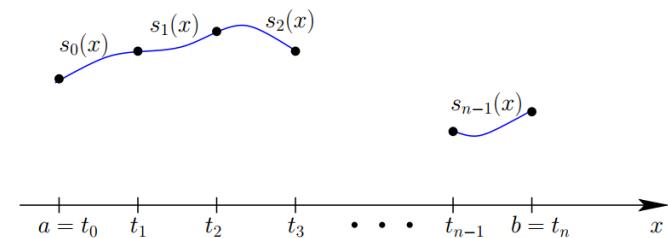
Splines

- Splines are polynomial segments joined end-to-end
- Segments are constrained to be smooth at the join
- The points at which the segments join are called *knots*
- The order m (order = degree+1) of the polynomial segments and
- the location of the knots define the system.

A function $S(x)$ is a spline of degree k on $[a, b]$ if

- $S \in C^{k-1}[a, b]$.
- $a = t_0 < t_1 < \dots < t_n = b$ and

$$S(x) = \begin{cases} S_0(x), & t_0 \leq x \leq t_1 \\ S_1(x), & t_1 \leq x \leq t_2 \\ \vdots \\ S_{n-1}(x), & t_{n-1} \leq x \leq t_n \end{cases}$$



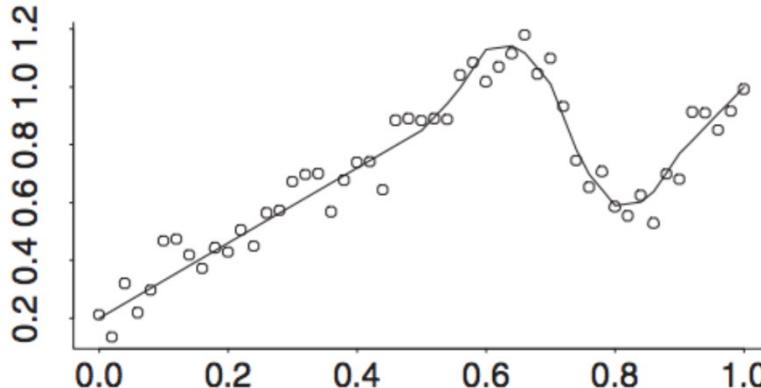
fda



MARQUETTE
UNIVERSITY

Be The Difference.

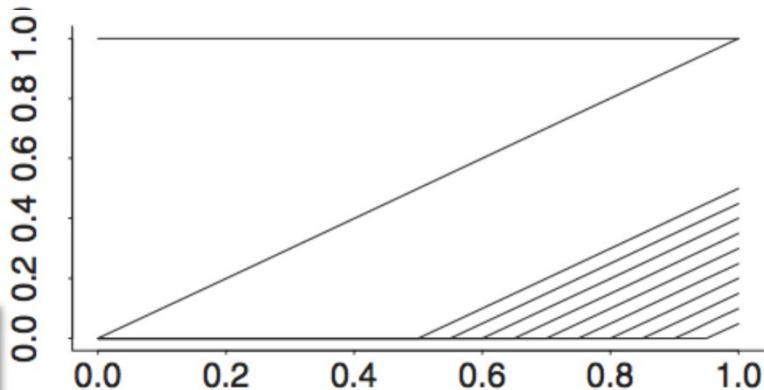
Truncated Linear Spline



■ Corresponding Basis:

$$X = \begin{bmatrix} 1 & x_1 & (x_1 - 0.5)_+ & (x_1 - 0.55)_+ & \cdots & (x_1 - 0.95)_+ \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & (x_n - 0.5)_+ & (x_n - 0.55)_+ & \cdots & (x_n - 0.95)_+ \end{bmatrix}$$

$$f(x) = \beta_0 + \beta_1 x + \sum_{k=1}^K b_k (x - \kappa_k)_+$$



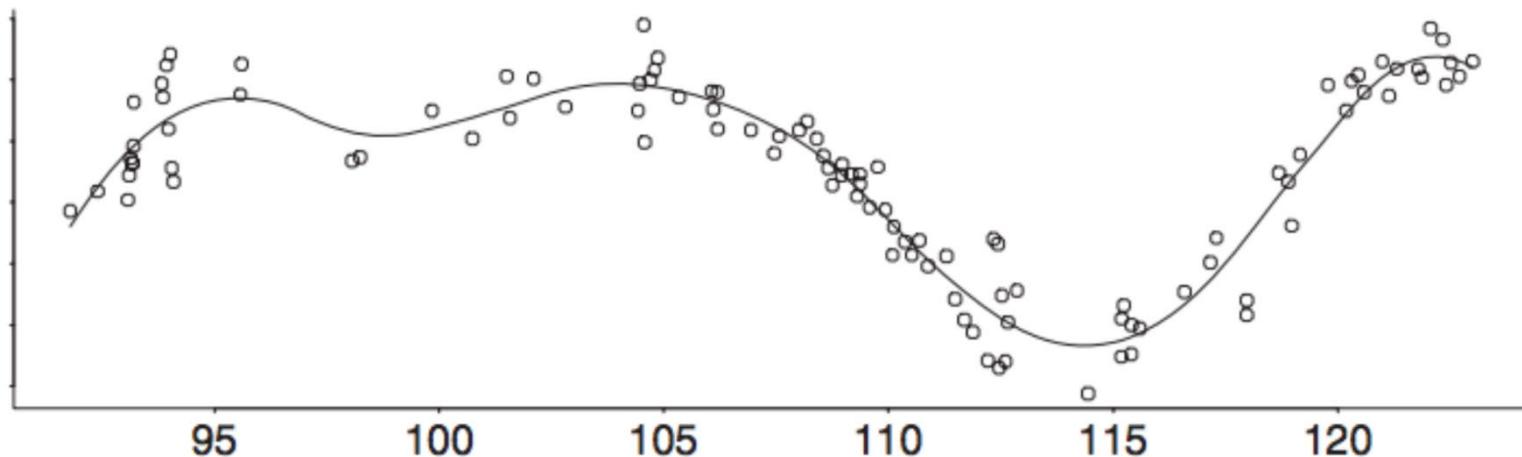
fda



MARQUETTE
UNIVERSITY

Be The Difference.

Truncated Power Spline



- Basis:

$$1, x, \dots, x^p, (x - \kappa_1)_+^p, \dots, (x - \kappa_k)_+^p,$$

- Model:

$$f(x) = \beta_0 + \beta_1 x + \cdots + \beta_p x^p + \sum_{k=1}^p \beta_{pk} (x - \kappa_k)_+^p.$$

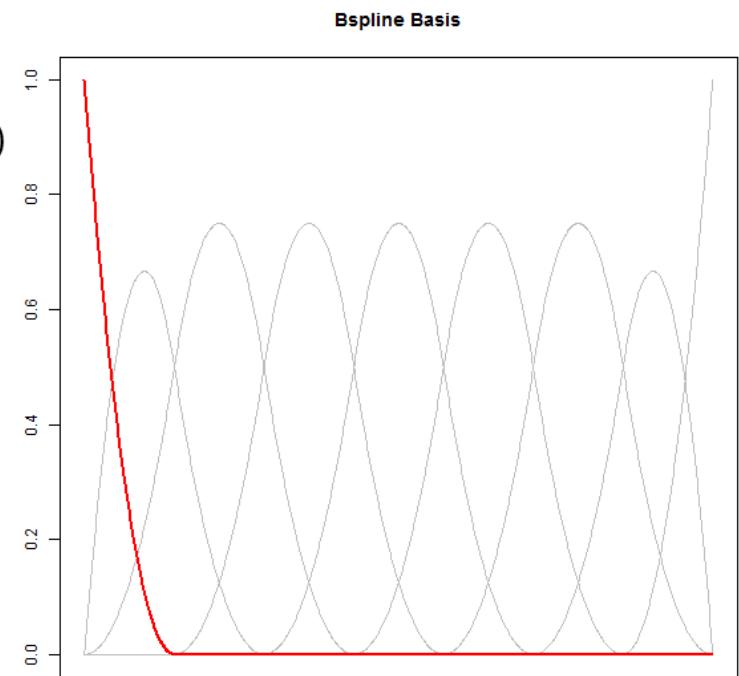
fda

B-spline Bases

$$B_{i,0}(x) := \begin{cases} 1 & \text{if } t_i \leq x < t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$B_{i,k}(x) := \frac{x - t_i}{t_{i+k} - t_i} B_{i,k-1}(x) + \frac{t_{i+k+1} - x}{t_{i+k+1} - t_{i+1}} B_{i+1,k-1}(x)$$

$$\begin{matrix} & & 0 \\ & & 0 \\ 0 & & B_{i-2,2} \\ & B_{i-1,1} & \\ B_{i,0} & & B_{i-1,2} \\ & B_{i,1} & \\ 0 & & B_{i,2} \\ & 0 & \\ & & 0 \end{matrix}$$



Properties of B-splines

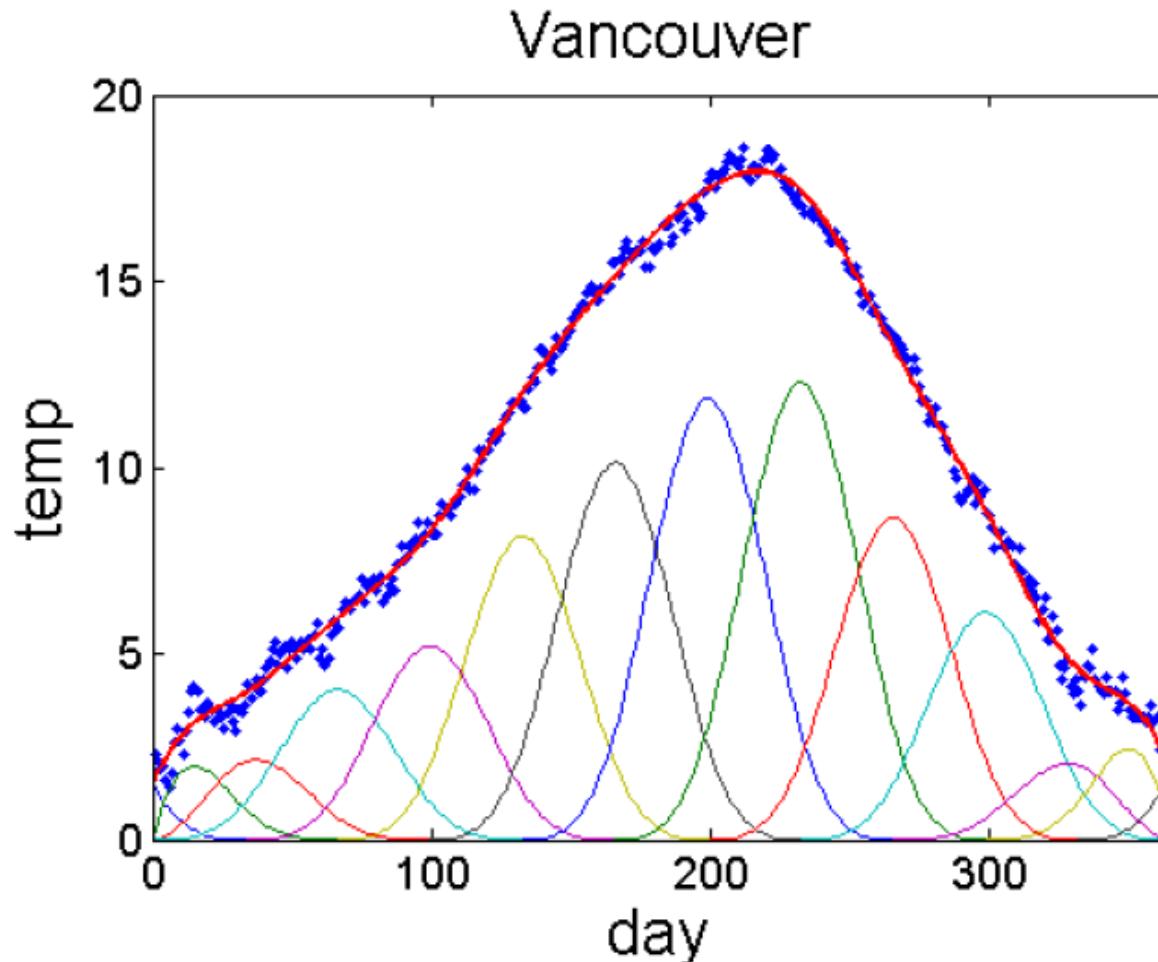
- Number of basis functions:

order + number interior knots

- Order m splines: derivatives up to $m - 2$ are continuous.
- Support on m adjacent intervals – highly sparse design matrix.

B-spline Bases

An illustration of basis expansions for *local* basis functions



fda



MARQUETTE
UNIVERSITY

Be The Difference.

Other Bases in fda Library

Constant $\phi(t) = 1$, the simplest of all.

Monomial $1, x, x^2, x^3, \dots, \dots$, mostly for legacy reasons.

Power $t^{\lambda_1}, t^{\lambda_2}, t^{\lambda_3}, \dots$, powers are distinct but not necessarily integers or positive.

Exponential $e^{\lambda_1 t}, e^{\lambda_2 t}, e^{\lambda_3 t}, \dots$

Other possible bases to represent $x(t)$:

Wavelets especially for sharp, local features (not in fda)

Empirical functional Principal Components (special topics)

1. Basis Expansions

Multi(Bi)variate Case

fda

From 1D to 2D Basis Functions using Tensor Product

- Suppose for each dimension we have a basis of function

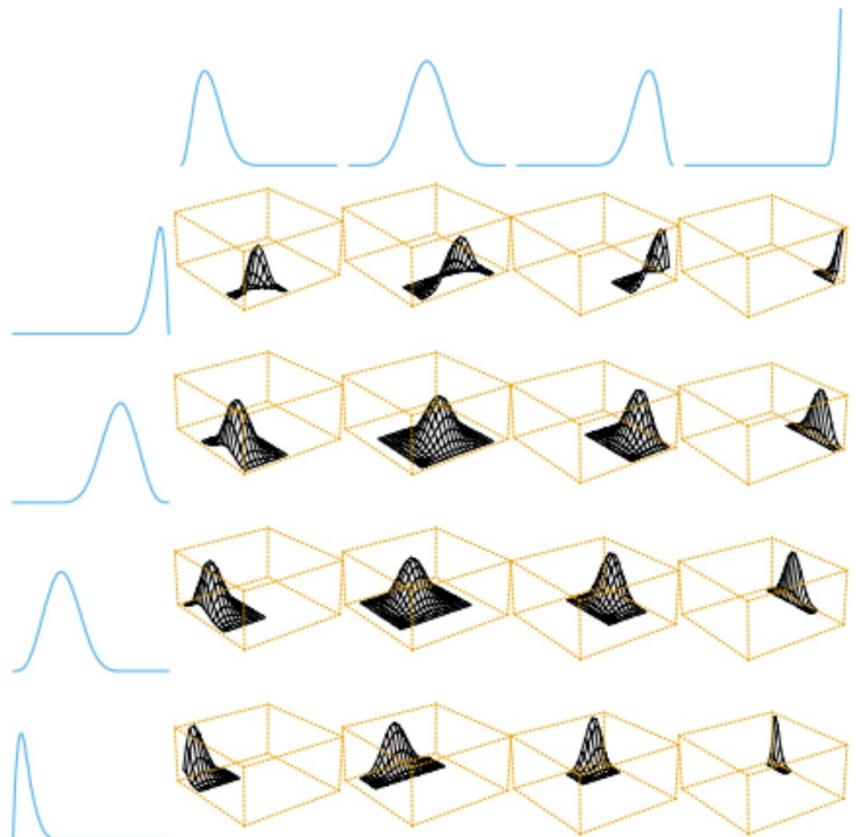
- We can write

$$f(x, y) = \sum_{i=1}^N \sum_{j=1}^M c_{ij} \phi_i(x) \psi_j(y)$$

- The $N \times M$ dimensional ***tensor product basis*** is:

$$B = B_x \otimes B_y$$

- R code



From Hastie, Tibshirani, Friedman book

Triangulation

- **Barycentric coordinates:** Given a triangle A , any point $v \in \mathbb{R}^2$ can be written as

$$v = b_1 v_1 + b_2 v_2 + b_3 v_3,$$

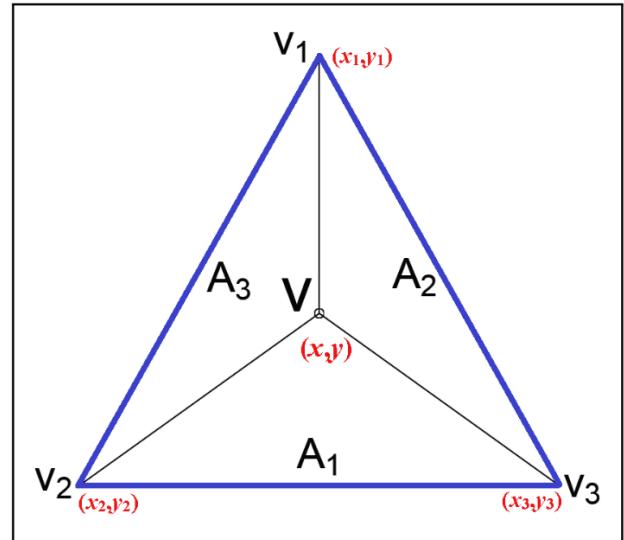
where (b_1, b_2, b_3) is the barycentric coordinates of v with respect to A

- **Some facts:**

$$\blacktriangleright b_1 + b_2 + b_3 = 1$$

$$\blacktriangleright b_i = \frac{\text{Area of } A_i}{\text{Area of } A}, \quad i = 1, 2, 3$$

$$\blacktriangleright v \in A \iff b_i \geq 0, \quad i = 1, 2, 3$$



fda

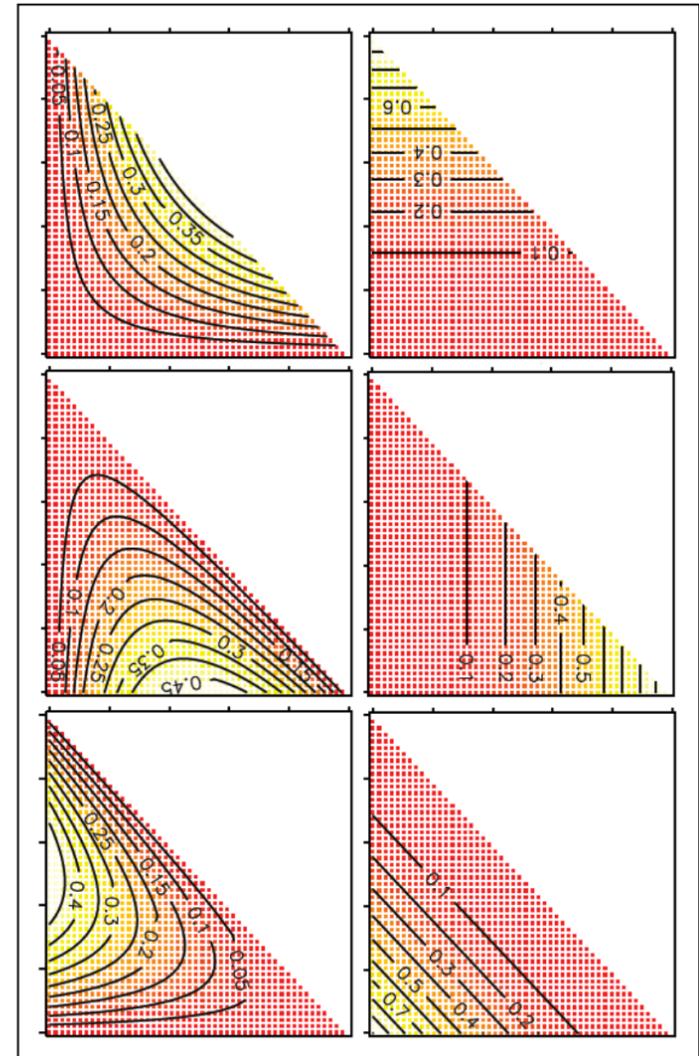
Triangulation (Bivariate B-splines Basis)

- **Bivariate B-splines:** Given a triangle A and a point $v \in A$, define:

$$B_{d,ijk}(v) := \frac{d!}{i!j!k!} b_1^i b_2^j b_3^k, \quad i+j+k=d$$

- For a given triangle A and $d = 2$, B_d contains six functions (see right panel):
- **B_d is a basis for $P_d(A)$:** $\forall g \in P_d(A)$,
 $\exists c_{ijk}$:

$$g(v) = \sum_{i+j+k=d} c_{ijk} B_{d,ijk}(v)$$



fda

Triangulation (Smoothness conditions)

Theorem

Suppose there are two triangles A_1 and A_2 sharing edge e . ω is any direction not parallel to common edge e and $D_\omega^n p(v)$ is n th order derivative in direction ω at point v . Then

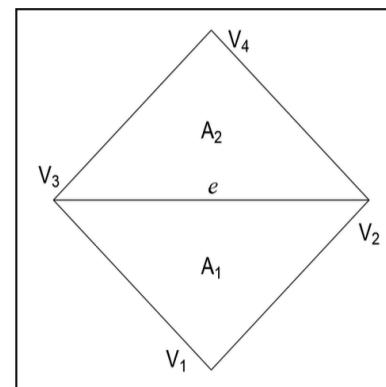
$$D_\omega^\ell p_1(v) = D_\omega^\ell p_2(v), \quad \forall v \in e \text{ and } \ell = 0, \dots, r \quad \text{iff}$$

$$c_{ijk}^{(2)} = \sum_{\nu+\mu+\kappa=\ell} c_{\nu,k+\mu,j+\kappa}^{(1)} B_{\nu\mu\kappa}^{(1)n}(v_4), \quad j+k = d-\ell, \quad \ell = 0, \dots, r.$$

- Hint: $\mathbf{Ac} = 0$ is the linear constraint for smoothness of the boundaries



spanning set for the null space of \mathbf{A} is a basis that satisfies the constraint



Summary

- 1 Basis expansions: just like adding different independent variables in linear regression
- 2 Monomial basis: direct extension of adding interaction and quadratic terms. Poor numerics, bad for derivatives.
- 3 Fourier basis: classical, common in signal processing etc. Great for periodic functions. Must be infinitely differentiable.
- 4 B-spline basis: locally polynomial. Allows control of smoothness and accuracy. Local definition \Rightarrow good numerics.
- 5 Other basis systems also exist.
- 6 What is best depends on the data.
 - The order of the spline should be at least $k + 2$ if you are interested in k derivatives.
 - Place more knots where you know there is strong curvature, and fewer where the function changes slowly.
 - Be sure there is at least one data point in every interval.
 - Knots are often equally spaced (a useful default)
 - Later, we'll discuss placing a knot at each point of observation.



MARQUETTE
UNIVERSITY

Be The Difference.

2. Smoothing Penalties

Univariate Case

fda

Ordinary Least-Squares Estimates

Assume we have observations for a single curve

$$y_i = x(t_i) + \epsilon$$

and we want to estimate

$$x(t) \approx \sum_{j=1}^K c_j \phi_j(t)$$

Minimize the sum of squared errors:

$$SSE = \sum_{i=1}^n (y_i - x(t_i))^2 = \sum_{i=1}^n (y_i - \Phi(t_i)\mathbf{c})^2$$

This is just linear regression!

Linear Regression on Basis Functions

- If the N by K matrix Φ contains the values $\phi_j(t_k)$, and \mathbf{y} is the vector (y_1, \dots, y_N) , we can write

$$SSE(\mathbf{c}) = (\mathbf{y} - \Phi\mathbf{c})^T(\mathbf{y} - \Phi\mathbf{c})$$

- The error sum of squares is minimized by the *ordinary least squares estimate*

$$\hat{\mathbf{c}} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{y}$$

- Then we have the estimate

$$\hat{x}(t) = \Phi(t)\hat{\mathbf{c}} = \Phi(t)(\Phi^T \Phi)^{-1} \Phi^T \mathbf{y}$$



MARQUETTE
UNIVERSITY

Be The Difference.

The Standard Model for Residual Distribution

- least squares is optimal for residuals that are independently and identically normal with mean 0 and variance σ^2 .
- That is

$$E\mathbf{y} = \Phi\mathbf{c} \text{ and } \text{Var}[\mathbf{y}] = \sigma^2\mathbf{I}$$

- Call this the *standard model* for the distribution of residuals.
- An unbiased estimate is

$$\hat{\sigma}^2 = \frac{1}{N - K} MSSE$$

- We know that $\hat{\mathbf{c}} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{y} = C\mathbf{y}$
- Then under the standard model

$$\text{Var}[\hat{\mathbf{c}}] = \sigma^2 C I C^T = \sigma^2 (\Phi^T \Phi)^{-1}$$



MARQUETTE
UNIVERSITY

Be The Difference.

The Standard Model for Residual Distribution

- Then we have the estimate

$$\hat{x}(t) = \Phi(t)\hat{c} = \Phi(t) \left(\Phi^T \Phi \right)^{-1} \Phi^T y$$

When we look at the values of \hat{x} at the observation points we have

$$\hat{y} = \Phi \left(\Phi^T \Phi \right)^{-1} \Phi^T y = S y$$

S is referred to as the *smoothing matrix*.

- And the variance-covariance matrix of the fitted values is

$$\text{Var} [\hat{y}] = \Phi C \Sigma C^T \Phi^T = \sigma^2 S$$



MARQUETTE
UNIVERSITY

Be The Difference.

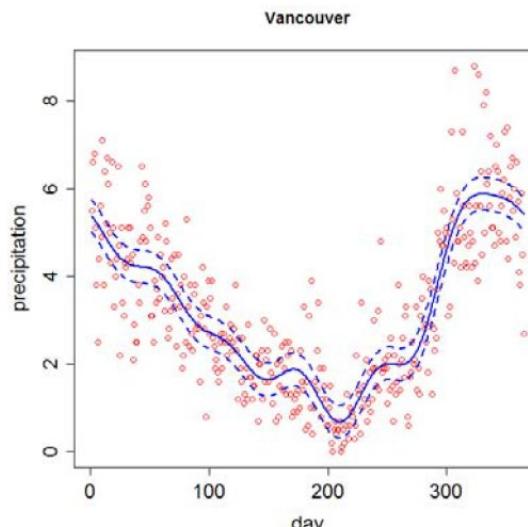
Pointwise Confidence Bands

- For each point we calculate lower and upper bands for $\hat{y}(t)$ by

$$\hat{y}(t) \pm 2\sqrt{\text{Var}[\hat{y}(t)]}$$

- These bands are not confidence bands for the entire curve, but only for the value of the curve at a fixed point.
- Ignores bias in the estimated curve

Fitted Vancouver Precipitation Data with 13 Fourier Bases

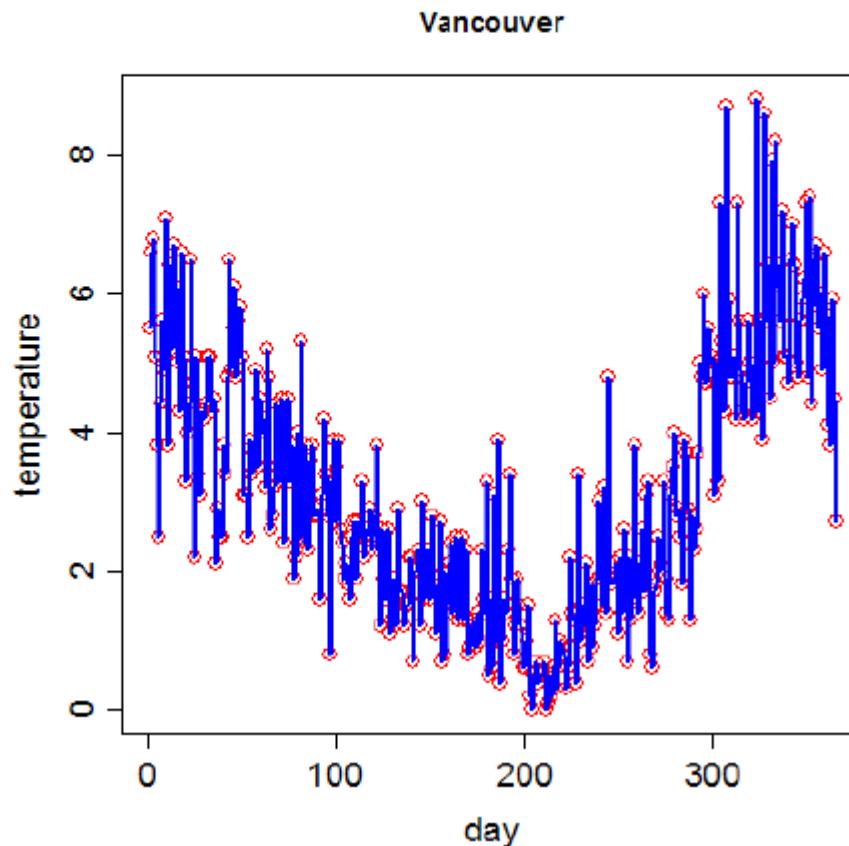


MARQUETTE
UNIVERSITY

Be The Difference.

Choosing the Number of Basis Functions

Vancouver Precipitation: 365 Fourier Bases



- Too many basis functions over-fits the data and reflect errors of measurement
- Too few basis functions fails to capture interesting features of the curves.

Cross-Validation

One method of choosing a model:

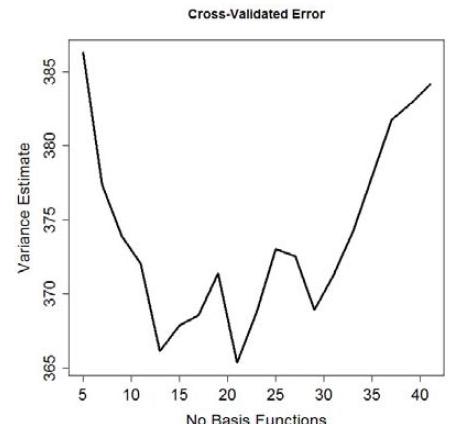
- leave out one observation (t_i, y_i)
- estimate $\hat{x}_{-i}(t)$ from remaining data
- measure $y_i - \hat{x}_{-i}(t)$
- Choose K to minimize the *ordinary cross-validation* score:

$$\text{OCV}[\hat{x}] = \sum (y_i - \hat{x}_{-i}(t_i))^2$$

- for a linear smooth $\hat{y} = Sy$,

Cross Validation for Vancouver Precipitation

$$\text{OCV}[\hat{x}] = \sum \frac{(y_i - \hat{x}(t_i))^2}{(1 - s_{ii})^2}$$



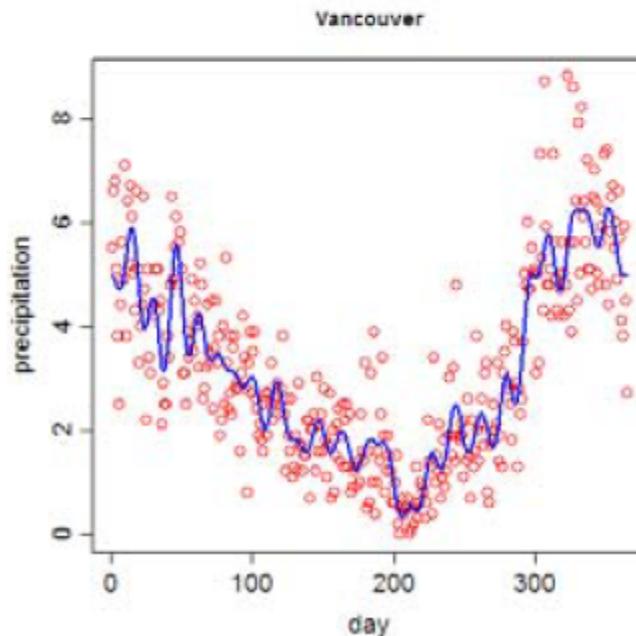
fda

What do we mean by smoothness?

Some things are fairly clearly smooth:

- constants
- straight lines

What we really want to do is eliminate small “wiggles” in the data.



fda

The D Operator

We use the notation that for a function $x(t)$,

$$Dx(t) = \frac{d}{dt}x(t)$$

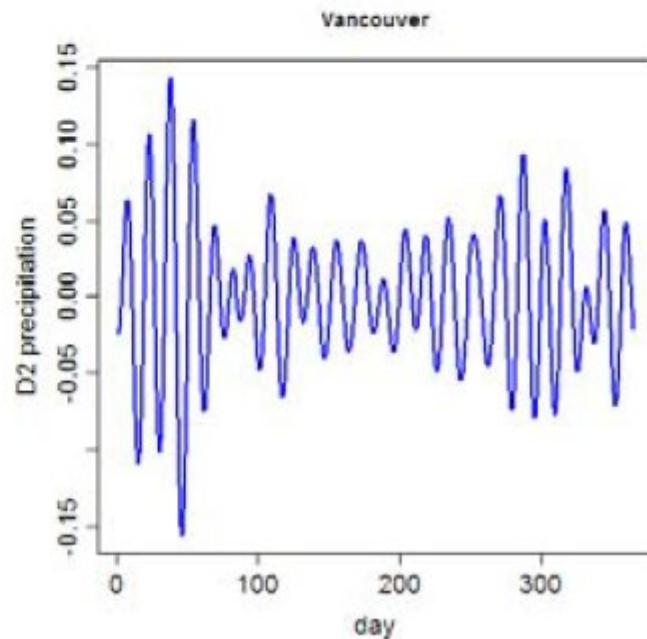
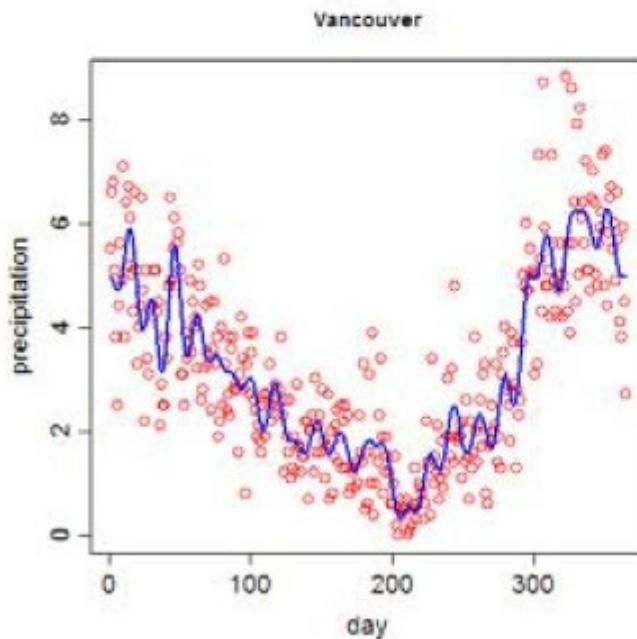
We can also define further derivatives in terms of powers of D :

$$D^2x(t) = \frac{d^2}{dt^2}x(t), \dots, D^kx(t) = \frac{d^k}{dt^k}x(t), \dots$$

- $Dx(t)$ is the instantaneous *slope* of $x(t)$; $D^2x(t)$ is its *curvature*.

Curvature of Vancouver Precipitation

53 Fourier bases:



The D Operator

We use the notation that for a function $x(t)$,

$$Dx(t) = \frac{d}{dt}x(t)$$

We can also define further derivatives in terms of powers of D :

$$D^2x(t) = \frac{d^2}{dt^2}x(t), \dots, D^kx(t) = \frac{d^k}{dt^k}x(t), \dots$$

- $Dx(t)$ is the instantaneous *slope* of $x(t)$; $D^2x(t)$ is its *curvature*.
- We measure the size of the curvature for all of f by

$$J_2(x) = \int [D^2x(t)]^2 dt$$



MARQUETTE
UNIVERSITY

Be The Difference.

Smoothing Penalties

- Problem: how to choose a basis? Large affect on results.
- Finesse this by specifying a very rich basis, but then imposing smoothness.
- In particular, add a penalty to the least-squares criterion:

$$\text{PENSSE} = \sum_{i=1}^n (y_i - x(t_i))^2 + \lambda J[x]$$

- $J[x]$ measures “roughness” of x .
- λ represents a continuous tuning parameter (to be chosen):
 - $\lambda \uparrow \infty$: roughness increasingly penalized, $x(t)$ becomes smooth.
 - $\lambda \downarrow 0$: penalty reduces, $x(t)$ fits data better.

The Smoothing Spline Theorem

Consider the “usual” penalized squared error:

$$PENSSE_{\lambda}(x) = \sum (y_i - x(t_i))^2 + \lambda \int [D^2 x(t)]^2 dt$$

- The function $x(t)$ that minimizes $PENSSE_{\lambda}(x)$ is
 - a spline function of order 4 (piecewise cubic)
 - with a knot at each sample point t_i

Cubic B-splines are exact; other systems will approximate solution as close as desired.

Calculating the Penalized Fit

When $x(t) = \Phi(t)\mathbf{c}$, we have that

$$\int [D^2x(t)]^2 dt = \int \mathbf{c}^T [D^2\Phi(t)] [D^2\Phi(t)]^T \mathbf{c} dt = \mathbf{c}^T R_2 \mathbf{c}$$

$[R_2]_{jk} = \int [D^2\phi_j(t)][D^2\phi_k(t)]dt$ is the *penalty matrix*.

The penalized least squares estimate for \mathbf{c} is

$$\hat{\mathbf{c}} = [\Phi^T \Phi + \lambda R_2]^{-1} \Phi^T \mathbf{y}$$

This is still a linear smoother:

$$\hat{\mathbf{y}} = \Phi [\Phi^T \Phi + \lambda R_2]^{-1} \Phi^T \mathbf{y} = S(\lambda) \mathbf{y}$$



Be The Difference.

Linear Smooths and Degrees of Freedom

- In least squares fitting, the degrees of freedom used to smooth the data is exactly K , the number of basis functions
- In penalized smoothing, we can have $K > n$.
- The smoothing penalty reduces the flexibility of the smooth
- The degrees of freedom are controlled by λ . A natural measure turns out to be

$$df(\lambda) = \text{trace}[S(\lambda)], \quad S(\lambda) = \Phi \left[\Phi^T \Phi + \lambda R_L \right]^{-1} \Phi^T$$

Choosing Smoothing Parameters: Cross Validation

There are a number of data-driven methods for choosing smoothing parameters.

- Ordinary Cross Validation: leave one point out and see how well you can predict it:

$$\text{OCV}(\lambda) = \frac{1}{n} \sum \left(y_i - x_{\lambda}^{-i}(t_i) \right)^2 = \frac{1}{n} \sum \frac{(y_i - x_{\lambda}(t_i))^2}{(1 - S(\lambda))_{ii}^2}$$

- Generalized Cross Validation tends to smooth more:

$$\text{GCV}(\lambda) = \frac{\sum (y_i - x_{\lambda}(t_i))^2}{[\text{trace}(\mathbb{I} - S(\lambda))]^2}$$

will be used here.

- Other possibilities: AIC, BIC, ...



MARQUETTE
UNIVERSITY

Be The Difference.

Constrained Functions

There are some situations in which we want to include known restrictions about $x(t)$.

- $x(t)$ is always positive

$$x(t) = e^{W(t)}$$

- $x(t)$ is always increasing (or decreasing)

$$Dx(t) = e^{W(t)} \quad \xrightarrow{Dx(t) > 0} \quad x(t) = \alpha + \int_{t_0}^t e^{W(s)} ds$$

- $x(t)$ is a density

$$x(t) \text{ a density} \Rightarrow \text{positive, integrates to 1} \rightarrow \quad x(t) = e^{W(t)} / \int e^{W(t)} dt$$

Idea: Enforce these conditions by transforming $x(t)$.

Define $W(t) = \Phi(t)\mathbf{c}$

2. Smoothing Penalties

Multi(Bi)variate Case

fda

3. Further Exploration of the Functional Data

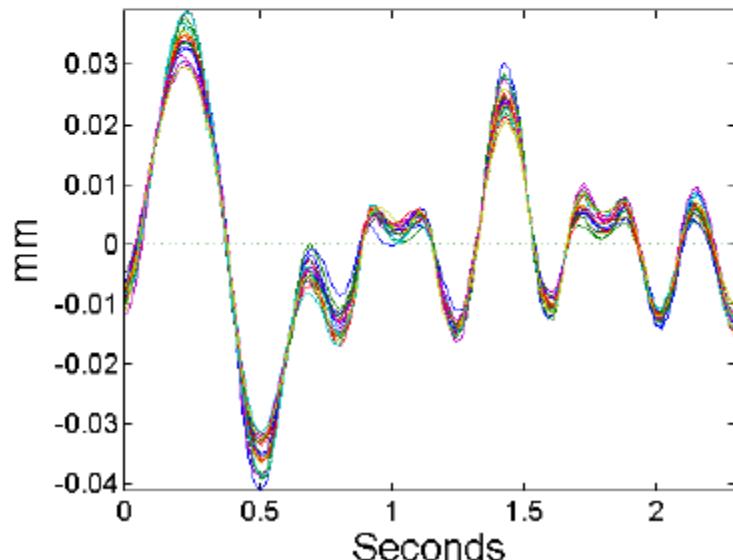
Dimension Reduction

Exploring Functional Data

How do we understand

- 1 mean
- 2 variance
- 3 covariance
- 4 important parts of covariance

in functional data?

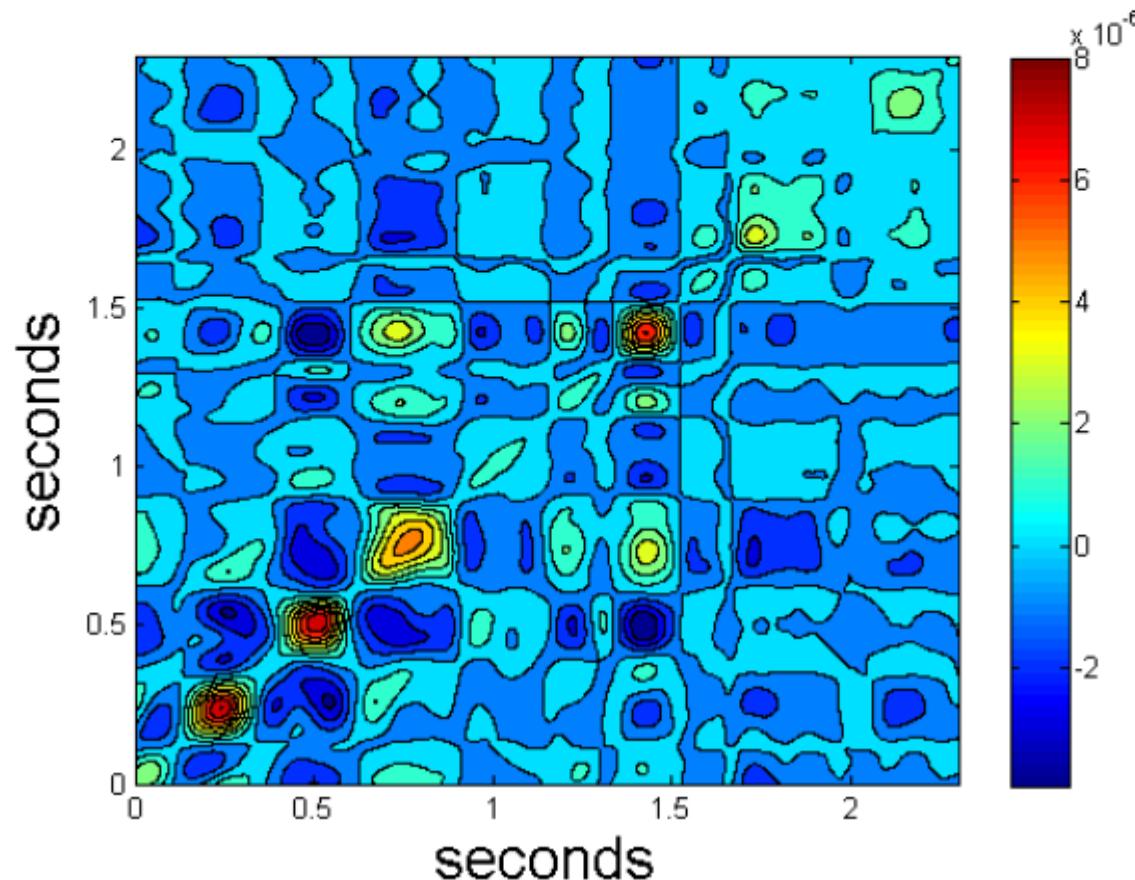


fda

The Covariance Surface

The covariance between $x(s)$ and $x(t)$ is a *surface*

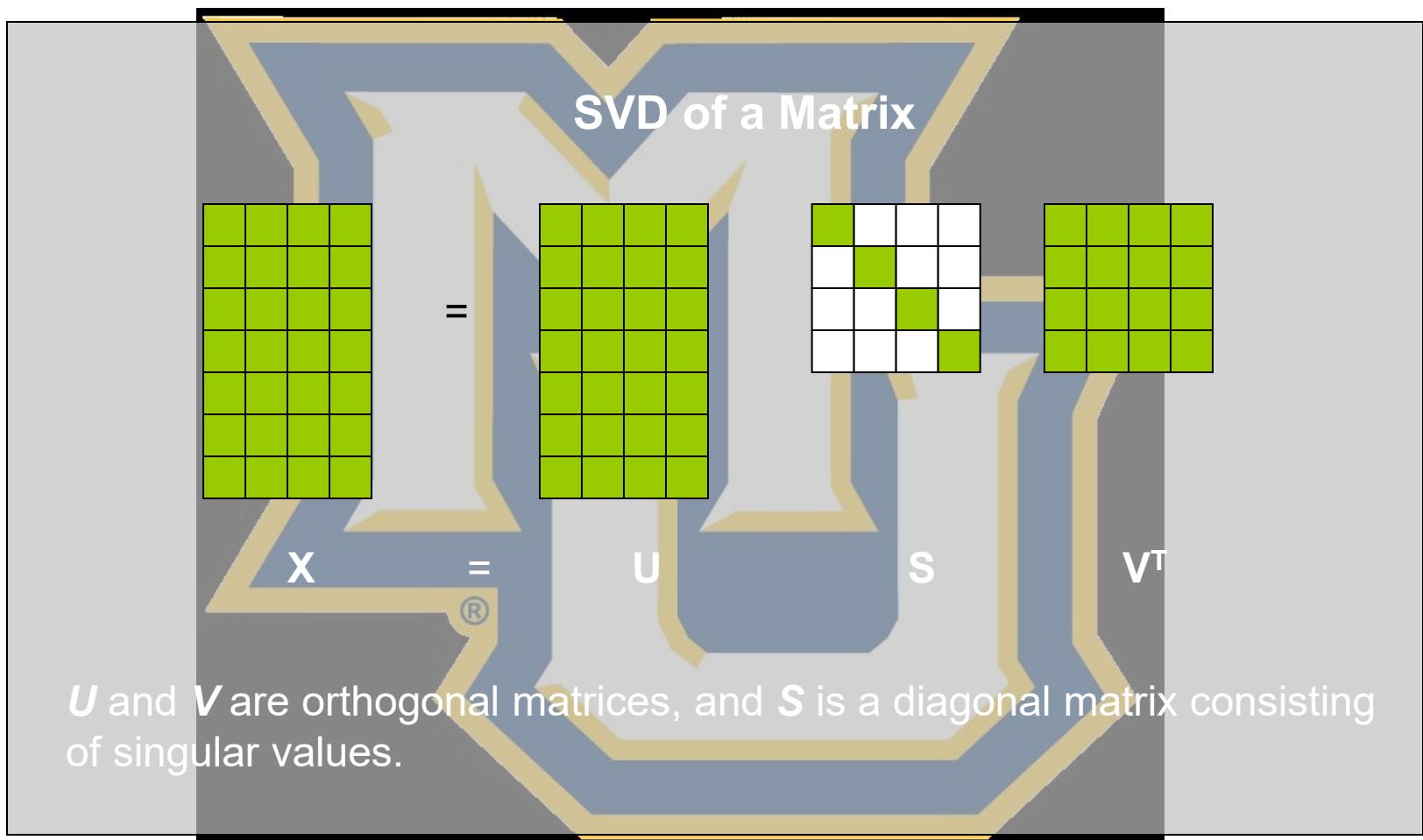
$$C(s, t) = \frac{1}{n-1} \sum (x_i(s) - \bar{x}(s))(x_i(t) - \bar{x}(t))$$



Be The Difference.

fda

Singular Value Decomposition(SVD)



fda

Truncated Singular Value Decomposition (TSVD)

$$X = U \cdot S \cdot V^T$$

Matrix dimensions:

- X : $n \times m$ (orange square)
- U : $n \times r$ (green rectangle)
- S : $r \times r$ (red diagonal matrix)
- V^T : $r \times m$ (green rectangle)

$$X_k = U_k \cdot S_k \cdot V_k^T$$

Matrix dimensions:

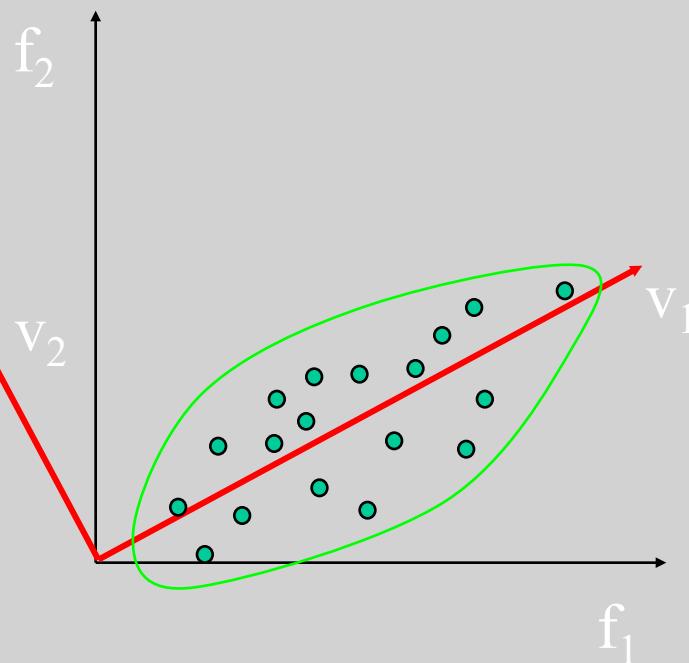
- X_k : $n \times m$ (teal square)
- U_k : $n \times k$ (teal vertical bar)
- S_k : $k \times k$ (red diagonal matrix)
- V_k^T : $k \times m$ (white rectangle)

The reconstructed matrix $X_k = U_k \cdot S_k \cdot V_k^T$ is the closest *rank-k* matrix to the original matrix X .

fda

Geometric Interpretation of PCA

- Intuition: find the axis that shows the greatest variation, and project all points onto this axis



$$\underset{\mathbf{v}}{\operatorname{argmax}} \operatorname{Var}(\mathbf{v}^T \mathbf{X}_r) = \underset{\mathbf{v}}{\operatorname{argmax}} \mathbf{v}^T \operatorname{Var}(\mathbf{X}_r) \mathbf{v}$$

But note that the sample covariance is given by $S=n^{(-1)}(\mathbf{X}^T \mathbf{X})$. Hence

We want to find the direction \mathbf{v} such that:

$$\underset{\mathbf{v}}{\operatorname{argmax}} \mathbf{v}^T (\mathbf{X}^T \mathbf{X}) \mathbf{v} = \underset{\mathbf{v}}{\operatorname{argmax}} \|\mathbf{Xv}\|_2$$

It can be shown that \mathbf{v} is the eigenvector associated to the largest eigenvalue of $\mathbf{X}^T \mathbf{X}$

Connection between SVD and Eigen Decomposition

SVD of a Matrix: observations

$$X = U S V^T$$

Multiply both sides by X^T

Multiplying on the left

$$X^T X = (U S V^T)^T U S V^T$$

$$X^T X = (V S U^T) U S V^T$$

remember $U^T U = I$

$$X^T X = V S^2 V^T$$

Multiplying on the right

$$X X^T = U S V^T (U S V^T)^T$$

$$X X^T = U S V^T (V S U^T)$$

$$V^T V = I$$

$$X X^T = U S^2 U^T$$

$$U = X V S^{-1}$$

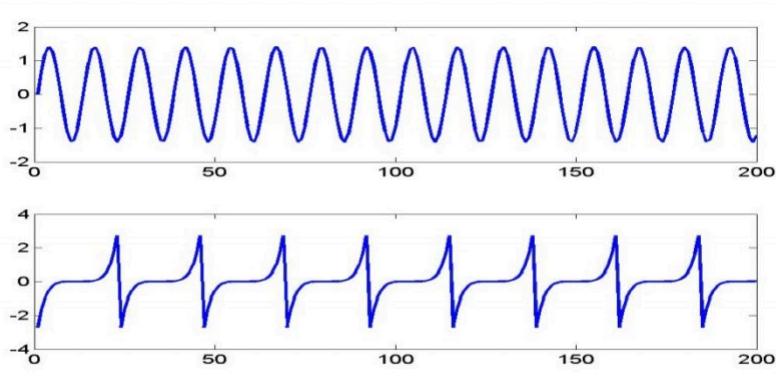
fda



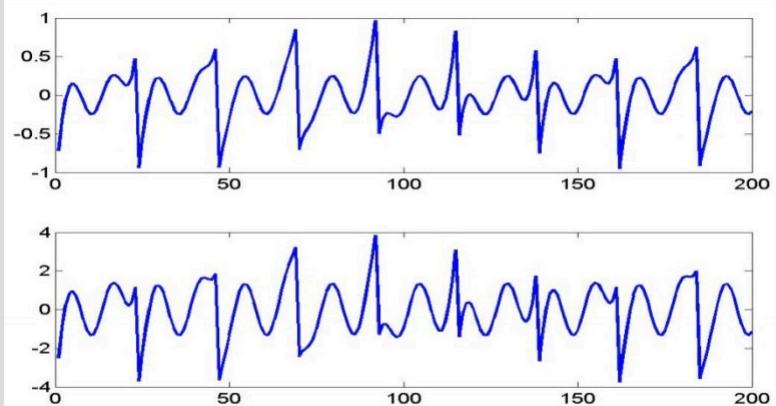
MARQUETTE
UNIVERSITY

Be The Difference.

Independent Component Analysis (ICA)



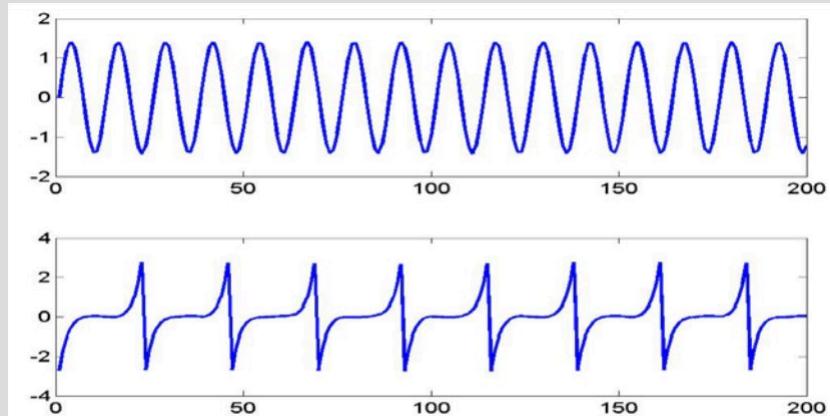
original signals



Observations (Mixtures)

$$\begin{aligned}x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) \\x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t)\end{aligned}$$

Model



ICA estimated signals

fda

Independent Component Analysis (ICA)

We observe

$$\begin{pmatrix} x_1(1) \\ x_2(1) \end{pmatrix}, \begin{pmatrix} x_1(2) \\ x_2(2) \end{pmatrix}, \dots, \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}$$

Model

$$\begin{aligned} x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) \\ x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t) \end{aligned}$$

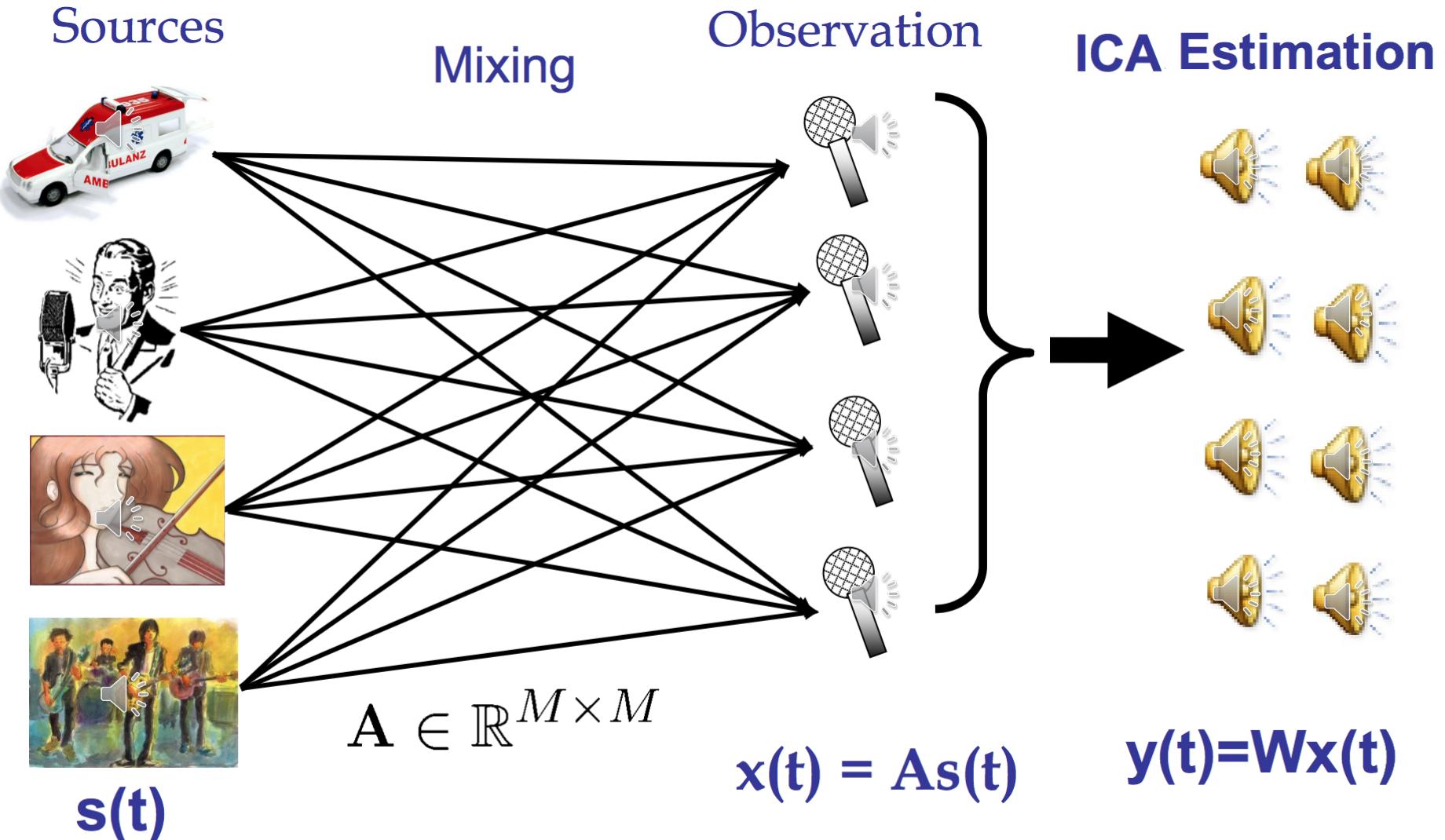
We want

$$\begin{pmatrix} s_1(1) \\ s_2(1) \end{pmatrix}, \begin{pmatrix} s_1(2) \\ s_2(2) \end{pmatrix}, \dots, \begin{pmatrix} s_1(t) \\ s_2(t) \end{pmatrix}$$

But we don't know $\{a_{ij}\}$, nor $\{s_i(t)\}$

Goal: Estimate $\{s_i(t)\}$, (and also $\{a_{ij}\}$)

The Cocktail Party Problem

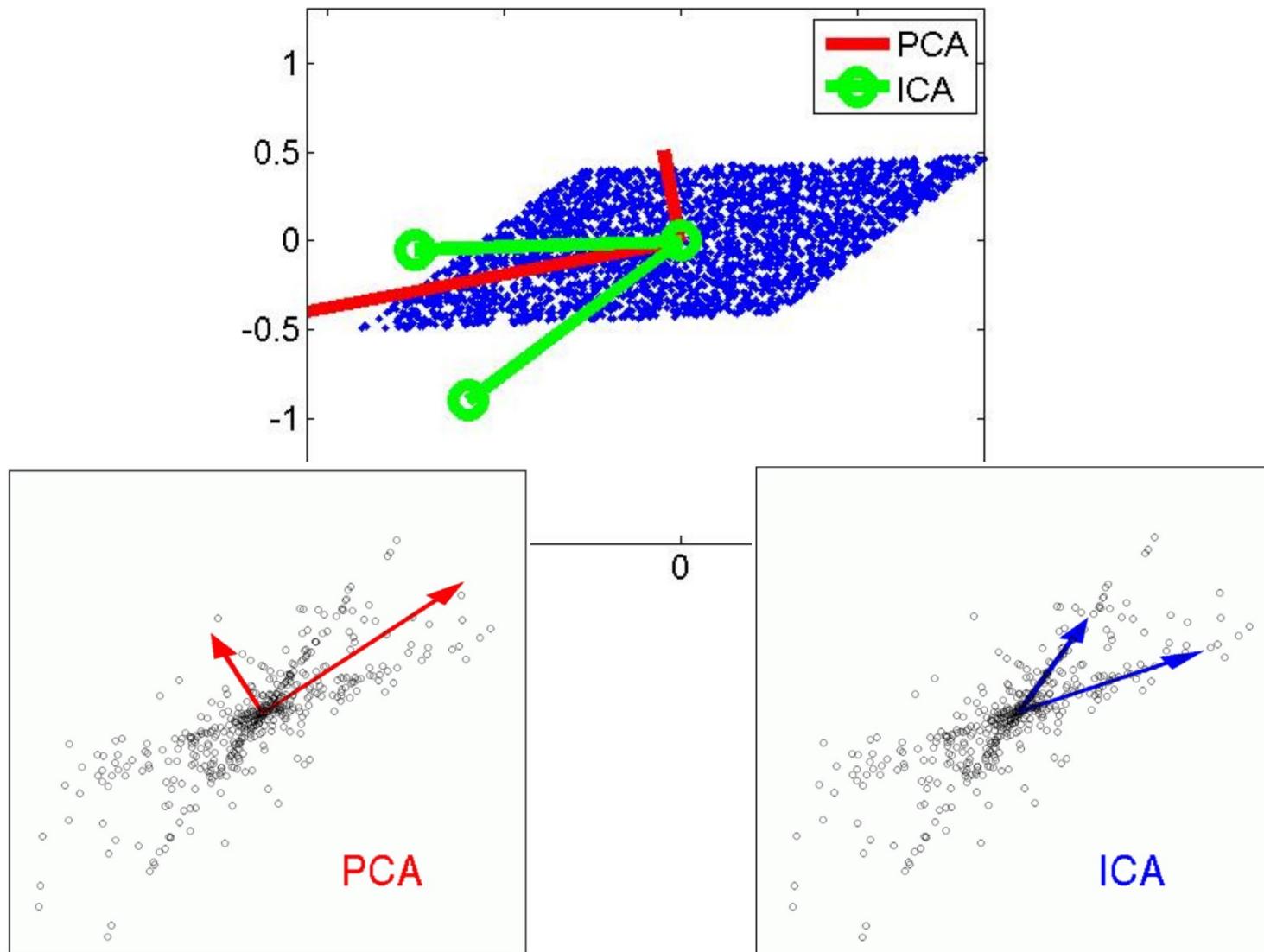


ICA vs PCA - Similarities

- PCA: $\mathbf{X} = \mathbf{US}$, where $\mathbf{S} = \Sigma \mathbf{V}^T$ and $\mathbf{U}^T \mathbf{U} = \mathbf{I}$
- ICA: $\mathbf{X} = \mathbf{AS}$, where \mathbf{A} is invertible
- PCA: just removes correlations, **not** higher order dependence
- ICA: removes correlations, **and** higher order dependence
- PCA: Some components are **more important** than others
- ICA: Components are **equally important**
- PCA: vectors are orthogonal
- ICA: vectors are **not** orthogonal

fda

ICA vs PCA – Cont...



fda



MARQUETTE
UNIVERSITY

Be The Difference.

Functional PCA

- Instead of covariance matrix Σ , we have a surface $\sigma(s, t)$.
- Would like a low-dimensional summary/interpretation.
- Multivariate PCA, use Eigen-decomposition:

$$\Sigma = U^T D U = \sum_{j=1}^p d_j u_j u_j^T$$

and $u_i^T u_j = I(i = j)$.

- For functions: use Karhunen-Loève decomposition:

$$\sigma(s, t) = \sum_{j=1}^{\infty} d_j \xi_j(s) \xi_j(t)$$

for $\int \xi_i(t) \xi_j(t) dt = I(i = j)$



MARQUETTE
UNIVERSITY

Be The Difference.

PCA and Karhunen-Loèvre

$$\sigma(s, t) = \sum_{i=1}^{\infty} d_i \xi_i(s) \xi_i(t)$$

- The $\xi_i(t)$ maximize $\text{Var} \left[\int \xi_i(t) x_j(t) dt \right]$.
- $d_i = \text{Var} \left[\int \xi_i(t) x_j(t) dt \right]$
- $d_i / \sum d_i$ is proportion of variance explained
- Principal component scores are

$$f_{ij} = \int \xi_j(t) [x_i(t) - \bar{x}(t)] dt$$

- Reconstruction of $x_i(t)$:

$$x_i(t) = \bar{x}(t) + \sum_{j=1}^{\infty} f_{ij} \xi_j(t)$$



MARQUETTE
UNIVERSITY

Be The Difference.

fda

4. Functional PCA

**Integrating Data Transformation
and
Handling the Missing Data**

fda

The fda Package

fda



fda Objects

The `fda` package provides utilities based on basis expansions and smoothing penalties.

`fda` works by defining objects that can be manipulated with pre-defined functions.

In particular

`basis objects` define basis systems that can be used

`fd objects` store functional data objects

`bifd objects` store functions of two-dimensions

`Lfd objects` define smoothing penalties

`fdPar objects` collect all three plus a smoothing parameter

Each of these are lists with prescribed elements.

Basis Objects

Define basis systems of various types. They have elements

`range|eval` Range of values for which basis is defined.

`nbasis` Number of basis functions.

Specific basis systems require other arguments.

Basis objects created by `create...basis` functions. eg

```
fbasis = create.fourier.basis(c(0,365),21)
```

creates a fourier basis on [0 365] with 21 basis functions.

Bspline Basis Objects

Bspline bases also require

`norder` Order of the splines.

`breaks` Knots (or break-points) for the splines.

```
nbasis = 17
```

```
norder = 6
```

```
months = cumsum(c(0,31,28,31,30,31,30,31,31,30,31,30,31))
```

```
bbasis = create.bspline.basis(c(0,365),nbasis,norder,months)
```

Creates a B-spline basis of order 6 on the year ([0 365]) with knots at the months.

Note that

```
nbasis = length(knots)+norder-2
```

`nbasis` is fragile in case of conflict.

Manipulating Basis Objects

Some functions that work with bases:

`plot(bbasis)`

plots `bbasis`.

`eval.basis(0:365,fbasis)`

evaluates `fbasis` at times `0:365`.

`inprod(bbasis,fbasis)`

produces the inner product matrix $J_{ij} = \int \phi_i(t)\psi_j(t)dt$.

Additional arguments allow use of $L\Phi$ for linear differential operators L .

Functional Data (fd) Objects

Stores functional data: a list with elements

`coefs` array of coefficients

`basis` basis object

`fdnames` defines dimension names

```
fdobj = fd(coefs,bbasis)
```

creates a functional data object with coefficients `coefs` and basis
`bbasis` `coefs` has three dimensions corresponding to

- 1 index of the basis function
- 2 replicate
- 3 dimension

Functional Arithmetic

fd objects can be manipulated arithmetically

`fdobj1+fdobj2, fdobj1^k, fdobj1*fdobj2`

are defined pointwise.

fd objects can also be subset

`fdobj[3,2]`

gives the 2nd dimension of the 3rd observation

Additionally

`eval.fd(0:365,fdobj)` returns an array of values of `fdobj` on 0:365.

`deriv.fd(fdobj,nderiv)` gives the `nderiv`-th derivative of `fdobj`.

`plot(fdobj)` plots `fdobj`

`eval.fd` and `plot` can also take argument `nderiv`.



MARQUETTE
UNIVERSITY

Be The Difference.

Lfd Objects

Define smoothing penalties

$$Lx = D^m x - \sum_{j=0}^{m-1} \alpha_j(t) D^j x$$

and require the α_j to be given as a list of fd objects.

Two common shortcuts:

`int2Lfd(k)` creates an Lfd object $Lx = D^k x$

`vec2Lfd(a)` for vector a of length m creates an Lfd object

$$Lx = D^m x - \sum_{j=1}^m a_j D^{j-1} x.$$

In particular

```
vec2Lfd(c(0,-2*pi/365,0))
```

creates a Harmonic acceleration penalty $Lx = D^3 x + \frac{2\pi}{365} D x$.



MARQUETTE
UNIVERSITY

Be The Difference.

fda

fdPar Objects

This is a utility for imposing smoothing. It collects

`fdobj` an `fd` (or a `basis`) object.

`Lfdobj` a `Lfd` object.

`lambda` a smoothing parameter.

fda

bifd Objects

Represents functions of two dimensions s and t as

$$x(s, t) = \sum_{i=1}^{K_1} \sum_{j=1}^{K_2} \phi_i(s) \psi_j(t) c_{ij}$$

requires

`coefs` for the matrix of c_{ij} .

`sbasis` basis object defining the $\phi_i(s)$.

`tbasis` basis object defining the $\psi_j(t)$.

Can also be evaluated (but not plotted).

`bifdPar` objects store `bifd` plus `Lfd` objects and λ for each of s and t .

fda

Smoothing Functions

Main smoothing function is `smooth.basis`

```
data(daily)
argvals = (1:365)-0.5
fdParobj = fdPar(fbasis,int2Lfd(2),1e-2)
tempSmooth =
smooth.basis(argvals,daily$tempav,fdParobj)
```

smooths the Canadian temperature data with a second derivative penalty, $\lambda = 0.01$. Along with an `fd` object it returns

`df` equivalent degrees of freedom

`SSE` total sum of squared errors

`gcv` vector giving GCV for each smooth

Typically, λ is chosen to minimize average `gcv`.

Note: numerous other smoothing functions, `Data2fd` just returns the `fd` and can avoid the `fdPar` object, `data2fd` is deprecated.



MARQUETTE
UNIVERSITY

Be The Difference.

Functional Statistics

Basic utilities:

`mean.fd` mean fd object

`var.fd` Variance or covariance (bifd object)

`cor.fd` Correlation (given as a matrix)

`sd.fd` Standard deviation (root diagonal of var.fd)

In addition, fPCA obtained through

```
temppca=pca.fd(tempfd$fd,nharm=4,fdParobj)
```

(Smoothing not strictly necessary). `pca.fd` output:

`harmonics` fd objects giving eigen-functions

`values` eigen values

`scores` PCA scores

`varprop` Proportion of variance explained

Diagnostics plots given by `plot(temppca)`



MARQUETTE
UNIVERSITY

Be The Difference.

Resources

- Ramsay and Silverman, 2005, "Functional Data Analysis", Springer
- Ramsay and Silverman, 2002, "Applied Functional Data Analysis", Springer
- Ramsay, Hooker and Graves, 2009, "Functional Data Analysis in R and Matlab", Springer
- www.functionaldata.org
- fda packages for both R and Matlab.
- In R package: code available to produce examples from all three books.
- Huang, Shen, and Buja, 2008, "Functional Principal Components Analysis via Penalized Rank One Approximation", Electronic Journal of Statistics 2, 678-695.

Thank you

■ ANY QUESTION?

fda