POLITECNICO MILANO 1863

MOX

An Introduction to Functional Data Analysis

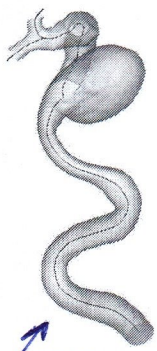
Piercesare Secchi

Applied Statistics – year 2018/2019

MOX, Department of Mathematics, Politecnico di Milano
piercesare.secchi@polimi.it

Introduction

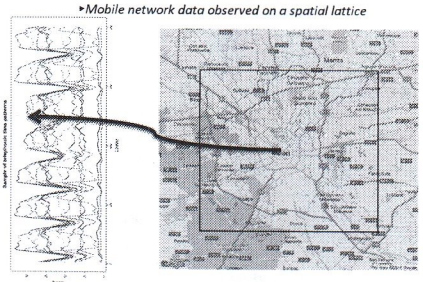
Functional data: where do they come from?



Explosive growth in recording complex and high-dimensional data having a functional nature (i.e., representable by curves, surfaces, dynamic curves and surfaces)

2D and 3D images and measures captured in time and space

▸ *Mobile network data observed on a spatial lattice*



Secchi, Vantini, Vitelli (2015), Statistical Methods and Applications

▸ *Reconstruction of an inner carotid artery with aneurysm, from angiographic images*

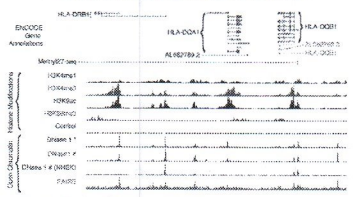
Sangalli, Secchi, Vantini, Veneziani (2009) J. R. Stat. Soc. Ser. C

here the data is a 3D curve, in patients with a 3D curve ⇒ functional data

more than $f: \mathbb{R} \to \mathbb{R}$

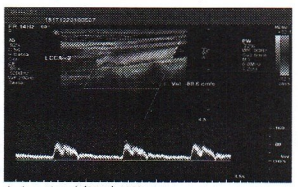
Functional data: where do they come from?

▸ Measurements of gene expression levels



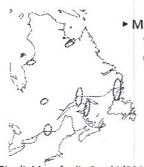
Cremona et al. (2015) BMC Bioinformatics

▸ ECD images for blood flow velocity field estimation



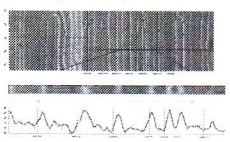
Azzimonti et al. (2014), JASA

▸ Manifold valued object data: temperature-precipitaion covariances in Quebec



Pigoli, Menafoglio Secchi (2016), JMVA

▸ Identification of past seasonal climates through the analysis of varves



Abramowicz et al. i (2016), SERRA

the ellipses are representing variance/covariance matrices (temperature / precipitations)

Functional data: where do they come from?

The analysis of complex and high dimensional data poses new and challenging problems in research

It is fueling one of the most fascinating and fast growing research fields of modern statistics

we receive data as discrete (from a grid) but we believe that this data comes from a function ⇒ **SMOOTHING**: creating (more "extrapolating") the function from data

how to move from data (always discrete) to a function representation

What are functional data?

- Informally, **functional data** are entities that can be described through a function, e.g., a curve, a surface, a image
- A **functional dataset** consists of a sample of functional observations
- Even though observations are actually discrete, the observed values reflect a **smooth variation of the phenomenon**. One might be interested not only in **point-wise** values, but also in **differential properties** of the data

Example: Berkeley Growth study
Observation of the height of 10 girls measured along 31 ages

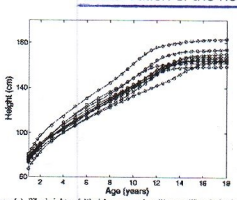


Figure 1.1 The heights of 10 girls measured at 31 ages. The circles indicate the unequally spaced ages of measurement.

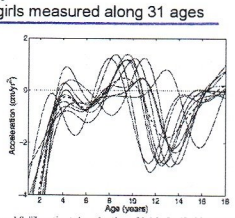


Figure 1.2 The estimated accelerations of height for 10 girls, measured in centimeters per year. The heavy dashed line is the cross-sectional mean, and is a rather poor summary of the curves.

Ramsay Silverman 2005 Springer

better to think about it as 10 functions

If we consider the finite case we have 10 stat. units with 31 features (years) ⇒ not good ($p \gg n$)

Berkeley Growth Curves as functional data

- Data reflect **smooth** variation of height over time: *h(t)*
- Some interesting features are only visible if **derivatives** are analyzed (e.g., mid-spurt and pubertal growth spurt)
- The grid spacing on the **time axis** is non-uniform. The underlying function might have been observed on different time points for different individuals
- **Large *p* small *n* problems**: classical multivariate methods fail when the number of variables is larger than the sample size (in this case, p=31, n=10)

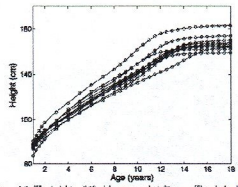


Figure 1.1. The heights of 10 girls measured at 31 ages. The circles indicate the unequally spaced ages of measurement.

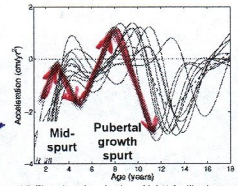


Figure 1.2. The estimated accelerations of height for 10 girls, measured in centimeters per year. The heavy dashed line is the cross-sectional mean, and is a rather poor summary of the curves.

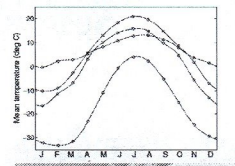
Ramsay Silverman 2005 Springer

MIS-ALLINEAMENT

it seems like there are periods where there is an acceleration of growing. This acceleration doesn't happen at the same time for all of them. we can conceptualize it only if we consider the data as FUNCTIONS and NOT VECTORS.

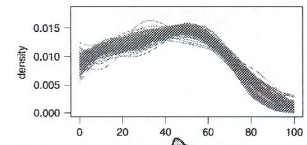
More simple examples of functional data

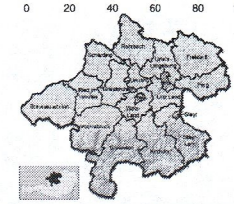
Example: Temperature curves in four locations in Canada (periodic data)



Ramsay Silverman 2005 Springer

Example: Density functions of Age Distribution in Austria (constrained data)





Hron et al. 2015, CSDA

Functional Data Analysis

- **Functional Data Analysis** is concerned with the statistical analysis of functional data
- **Typical goals of FDA**
  - Represent the data in ways that aid further analysis
  - Display the data to highlight their salient features
  - Study the main sources of pattern and variation among the data
  - Explain an outcome (response) using input/independent variable information. Here either the input or the output (or both) might be functional.
  - Classify the data or compare groups of data with respect to certain type of variations
- In this **short course**, we will be concerned with:
  - Representing data: given raw/discrete observations, represent the data through a functional form
  - Reducing the dimensionality of the representation space and highlights the main sources of variability (as in Principal Component Analysis)
  - Aligning (registration) and clustering (unsupervised classification) data

Course Agenda

1. **Hilbert space model for functional data**
   - 1.1. Basics notions on Hilbert spaces
   - 1.2. Hilbert space embedding for functional data
   - 1.3. Formal definition of functional data
2. **Smoothing and interpolation of functional data**
   - 2.1. Basis function
   - 2.2. Least square smoothing
   - 2.3. Smoothing with a differential penalization
3. **FDA & Dimensionality reduction in Hilbert spaces**
   - 3.1. Functional Principal Components in Hilbert spaces
   - 3.2. Examples in L2
4. **Data alignment and clustering**
   - 4.1 Phase and amplitude variability
   - 4.2 Landmark and continuous registration
   - 4.3 Decoupling phase and amplitude variability
   - 4.4 K-mean alignment

General references on FDA

Books:

- **Ramsay, J.O. and Silverman, B.W. (2005). *Functional Data Analysis*, Springer, 2nd ed.**
- Ramsay, J.O. and Silverman, B.W. (2002). *Applied Functional Data Analysis*, Springer.
- Ramsay, J.O., Hooker, G. and Graves, S. (2009). *Functional Data Analysis with R and Matlab*, Springer.
- Ferraty, F. and Vieu, P. (2006). *Nonparametric Functional Data Analysis: Theory and Practice*, Springer.
- Horvath, L. and Kokoszka P. (2012). Inference for Functional Data with Applications, Springer.

Software: (available from CRAN)

- R package fda (corresponding Matlab code available from http://www.psych.mcgill.ca/misc/fda/)
- R package Refund
- Matlab code PACE
- R package mgcv
- R package fdakma (alignment and clustering)
- R package fdaPDE (surfaces)

simplest space we can work with (almost like $\mathbb{R}^p$; Hilbert spaces have the same architecture/structure as $\mathbb{R}^p$)

1. Hilbert space model for functional data

Course Agenda

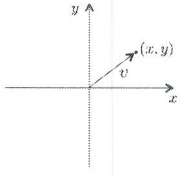
1.1. Basics notions on Hilbert spaces: a reminder

A Hilbert Space approach to the analysis of Functional Data

The notion of **Hilbert space** generalizes the concept of Euclidean space to spaces of any (even infinite) dimension

- Vectorial structure (linear combinations)
- Distance, angles, projections (measure of dependence, best approximations)

Euclidean space $\mathbb{R}^2$



- Sum: $v_1 + v_2 = (x_1 + x_2, y_1 + y_2)$
- Product by a constant: $c \cdot v = (c \cdot x, c \cdot y)$

} Operations (+, ·)

- Norm (lenght of a vector): $\|v\| = (x^2 + y^2)^{1/2}$
- Distance: $\|v_1 - v_2\|^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$
- Angle: $\vartheta = \arccos \frac{\langle v_1, v_2 \rangle}{\|v_1\| \|v_2\|}$

} Inner product $\langle v_1, v_2 \rangle = (x_1 \cdot x_2) + (y_1 \cdot y_2)$

we use Hilbert spaces so we can rely on the structure (similar to the $\mathbb{R}^p$ structure) and recall all the tools of the "$\mathbb{R}^p$ - statistics"

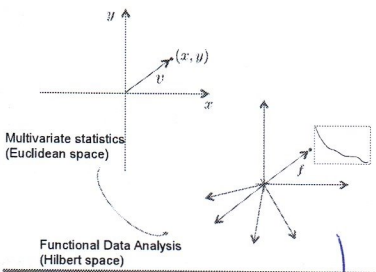
it implies the measure of angles, a norm, a distance

1.1. Basics notions on Hilbert spaces

A Hilbert Space approach to the analysis of Functional Data

The notion of **Hilbert space** generalizes the concept of Euclidean space to spaces of any (even infinite) dimension

- Vectorial structure (linear combinations)
- Distance, angles, projections (measure of dependence, best approximations)



Why Hilbert spaces?

- We understand functional data as **points of a space of functions**
- Many methods of **multivariate statistics can be extended to data embedded in a Hilbert space**, through the notions of inner product and norm

every point of a Hilbert space is a function

1.1. Basics notions on Hilbert spaces

Inner product spaces

Let H be a linear space. An inner product on H is a bilinear, symmetric, positive definite form

$$\langle \cdot, \cdot \rangle : H \times H \to \mathbb{R}$$

that satisfies

(i) $\langle \lambda x + y, z \rangle = \lambda \langle x, z \rangle + \langle y, z \rangle \qquad \forall \lambda \in \mathbb{R},\ \forall x, y, z \in H$

(ii) $\langle x, y \rangle = \langle y, x \rangle \qquad \forall x, y \in H$

(iii) $\langle x, x \rangle \geq 0 \qquad \forall x \in H$

(iv) $\langle x, x \rangle = 0 \iff x = 0$

In particular:

- The inner product allows to measure lengths and angles
- It allows to define orthogonality: two vectors in H are orthogonal if $\langle x, y \rangle = 0$
- The inner product induces a norm and a metric
- The inner product allows generalizing the Pythagoras' Theorem:

$$\|x + y\|^2 = \|x\|^2 + \|y\|^2 \quad \text{if and only if } \langle x, y \rangle = 0$$

1.1. Basics notions on Hilbert spaces

Hilbert spaces

A (real) Hilbert space H is an inner product space that is complete, in the norm induced by the inner product.

- A Hilbert space is complete in the sense that it contains all the limit points of its Cauchy sequences;
- A Hilbert space is separable if it contains a dense countable subset; ⇒ ∃ countable orthonormal basis for the space
- Useful properties:
  - In a Hilbert space one has the notion of orthogonal projection and of best approximations
  - A Hilbert space H is separable iff it has an orthonormal basis $\{u_n\}_{n \in \mathbb{N}}$
  - If H is separable Hilbert space, $\{u_n\}_{n \in \mathbb{N}}$ is an orthonormal basis and $x \in H$. Then

$$x = \sum_{n=1}^{\infty} \langle x, u_n \rangle u_n. \qquad \text{Basis expansion}$$

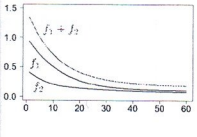
1.1. Basics notions on Hilbert spaces

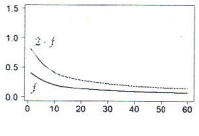
An Example: the Hilbert space L^2

L^2: space of real-valued square-integrable functions

- Sum: $(f_1 + f_2)(t) = f_1(t) + f_2(t)$
- Product by a constant: $(c \cdot f)(t) = c \cdot f(t)$

} Operations (+, ·)





- Norm: $\|f\|^2 = \int (f(t))^2 dt$
- Distance: $\|f_1 - f_2\|^2 = \int (f_1(t) - f_2(t))^2 dt$
- Angle: $\vartheta = \arccos \frac{\langle f_1, f_2 \rangle}{\|f_1\| \|f_2\|}$

} Inner product $\langle f_1, f_2 \rangle = \int (f_1(t) \cdot f_2(t)) dt$

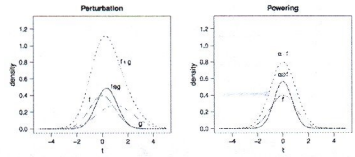
More precisely, L^2 is a quotient space with respect to the equivalence relation: $x = y$ if $\int [x(t) - y(t)]^2 dt = 0$

REMEMBER that every point is not a function but an equivalence class: all functions that differ only on a zero-measure set are the same in this space

good space when the important thing is the ratio between data ($\mathbb{P}(\text{event 1}) = 3\,\mathbb{P}(\text{event 2})$) or when we have densities

B^2: space of density functions on a close interval *I*, with log in L^2

- Equivalence relation: f, g are equivalent if they are proportional (*scale invariance*)
- Sum (perturbation): $(f \oplus g)(t) = \frac{f(t)g(t)}{\int_I f(s)g(s)\,ds}$ ← product normalized
- Product by a constant (powering): $(\alpha \odot f)(t) = \frac{f(t)^\alpha}{\int_I f(s)^\alpha\,ds}, \quad t \in I.$
- Inner product: $\langle f, g\rangle_{B^2} = \frac{1}{2\eta}\int_I\int_I \ln\frac{f(t)}{f(s)} \ln\frac{g(t)}{g(s)}\,dt\,ds$
- Norm: $\|f\|_{B^2} = \left[\frac{1}{2\eta}\int_I\int_I \ln^2\frac{f(t)}{f(s)}\,dt\,ds\right]^{1/2}$



Note: the geometry of L^2 wouldn't make sense for density functions

if we sum two densities in L^2 we obtain a function that is not a density, in B^2 we obtain another density

1.1. Basics notions on Hilbert spaces
An Example: the Bayes Hilbert space B^2

B^2: space of density functions on a close interval *I*, with log in L^2

- Equivalence relation: f, g are equivalent if they are proportional (*scale invariance*)
- Sum (perturbation): $(f \oplus g)(t) = \frac{f(t)g(t)}{\int_I f(s)g(s)\,ds}$
- Product by a constant (powering): $(\alpha \odot f)(t) = \frac{f(t)^\alpha}{\int_I f(s)^\alpha\,ds}, \quad t \in I.$
- Inner product: $\langle f, g\rangle_{B^2} = \frac{1}{2\eta}\int_I\int_I \ln\frac{f(t)}{f(s)} \ln\frac{g(t)}{g(s)}\,dt\,ds$
- Norm: $\|f\|_{B^2} = \left[\frac{1}{2\eta}\int_I\int_I \ln^2\frac{f(t)}{f(s)}\,dt\,ds\right]^{1/2}$

- B^2 is isomorphic to L^2 (in fact, all the Hilbert spaces are isomorphic). An isometric isomorphism is provided, e.g., by the **centred log-ratio transformation**

$$\mathrm{clr}(f)(t) = f_c(t) = \ln f(t) - \frac{1}{\eta}\int_I \ln f(s)\,ds.$$

Exercise: prove that

$$\mathrm{clr}(f \oplus g)(t) = f_c(t) + g_c(t), \qquad \mathrm{clr}(\alpha \odot f)(t) = \alpha \cdot f_c(t), \qquad \langle f, g\rangle_{B^2} = \langle f_c, g_c\rangle_{L^2} = \int_I f_c(t) g_c(t)\,dt.$$

we can transform the data and move in L^2, work in L^2 and then go back to B^2 (there are many possible isomorphism to move from B^2 to L^2 or contrary)

1.1. Basics notions on Hilbert spaces
An Example: the Bayes Hilbert space B^2

B^2: space of density functions on a close interval *I*, with log in L^2

- Equivalence relation: f, g are equivalent if they are proportional (*scale invariance*)

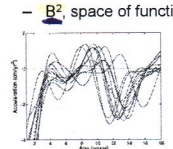
- Hilbert space structure for functional compositional data (e.g., probability density functions)
- Account for the key properties of compositional data: scale invariance, relative scale, sub-compositional coherence
- Meaningful interpretations in mathematical statistics, e.g.,
  - Exponential families as affine finite-dimensional subspaces
  - Perturbation $\oplus$ as a Bayes update of information

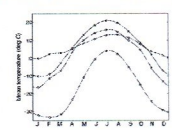
Exercise: prove that

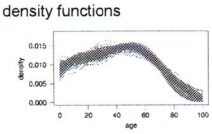
exponential families are linear spaces in B^2

1.2. Hilbert space embedding for functional data

- As a first step of any functional data analysis, one need to **choose the embedding** for the data (already a modeling operation)
- **Separable Hilbert spaces are a convenient choice** (projections, best approximations).
  Note: Not all the interesting spaces are Hilbert: e.g., the space of continuous functions is not a Hilbert space. Other interesting spaces: Riemannian manifolds (OODA)
- Examples of Hilbert spaces for FDA:
  - L^2, space of square integrable functions: OK for most data analyses (especially if data are unconstrained)
  - B^2, space of functional compositions: useful for density functions


Acceleration in Berkeley Growth data


Temperatures in Canada



Population Age densities

1.3. Formal definition of functional data
Functional random variables and functional data

- Let *H* be a Hilbert space, whose points are functions defined on a closed interval $T = [t_{min}, t_{max}]$ (e.g., range of time during which the data are collected)
- Hereafter, we will always consider functional data in Hilbert spaces

Ω = space, $\mathfrak{F}$ = σ-field, $\mathbb{P}$ = prob. def. on $\mathfrak{F}$

Definition 1:
A **functional random variable** is a random element on a probability space $(\Omega, \mathfrak{F}, \mathbb{P})$ in the space *H*: $X : \Omega \to H$

Definition 2:
A **functional datum** x is a realization of a functional random variable, i.e., for $\omega \in \Omega$

$$x = X(\omega) : T = [t_{min}, t_{max}] \to \mathbb{R}$$

Definition 3:
A **functional dataset** is a collection of functional data.

1.3. Formal definition of functional data
Mean and covariance operator

Let $X : \Omega \to H$ be a functional random variable in *H*. Hereafter, we always assume that $\mathbb{E}[\|X\|_H^4] < \infty$.

Definition 4:
We call Fréchet mean of X the (unique) element μ of *H* that solves

$$\mu = \operatorname*{arginf}_{x \in H} \mathbb{E}[\|X - x\|_H^2].$$

- If $H=L^2$ (space of square-integrable functions), the Fréchet mean coincides a.e. with the point-wise mean

$$\mathbb{E}[X(t)] = \mu(t), \quad t \in T$$

- If $H=B^2$ (Bayes space of PDFs), the Fréchet mean can be computed as

$$\mu = clr^{-1}(\mathbb{E}[clr(X)])$$

(in particular, one can define the mean of the clr-transformed variable point-wise)

- In any *H*, one can **estimate the mean via the sample estimator**

$$\overline{X} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

In $H=L^2$, this is the point-wise sample mean

1.3. Formal definition of functional data
Mean and covariance operator

Let $X : \Omega \to H$ be a **zero-mean** functional random variable in H, such that $\mathbb{E}[\|X\|_H^4] < \infty$.

Definition 5:
We call covariance operator of X the operator from H to H defined as

$$Cx = \mathbb{E}[\langle X, x\rangle X], \quad x \in H$$

- If $H=L^2$ (space of square-integrable functions), the covariance operator can be equivalently defined through a kernel operator

$$[Cx](t) = \int_T c(s,t)x(s)d(s)\,, \quad x \in L^2$$

where the covariance kernel is precisely the point-wise covariance

$$c(s,t) = \mathbb{E}[X(s)X(t)]$$

- In $H=\mathbb{R}^p$, the covariance operator coincides with the linear operator defined by the covariance matrix

1.3. Formal definition of functional data
Mean and covariance operator

Let $X : \Omega \to H$ be a **zero-mean** functional random variable in H, such that $\mathbb{E}[\|X\|_H^4] < \infty$

Definition 5:
We call covariance operator of X the operator from H to H defined as

$$Cx = \mathbb{E}[\langle X, x\rangle X], \quad x \in H$$

- In any H, the covariance operator can be estimated through the sample covariance operator

$$Sx = \frac{1}{N}\sum_{i=1}^{N}\langle X_i, x\rangle X_i, \quad x \in H$$

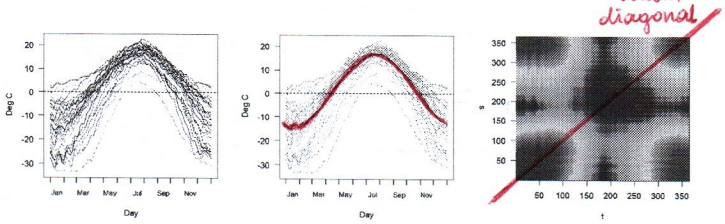
- If $H=L^2$, one can use the alternative definition

$$[Sx](t) = \int_T \hat{c}(s,t)x(s)d(s)\,, \quad x \in L^2 \text{ with } \quad \hat{c}(s,t) = \frac{1}{N}\sum_{i=1}^{N} X(s)X(t)$$

1.3. Formal definition of functional data
An example in L^2

Ramsay Silverman 2005 Springer

Dataset of Temperatures in Canada (35 observations)

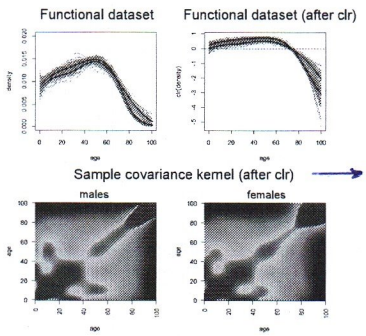


Functional dataset | Sample mean | Sample covariance kernel

1.3. Formal definition of functional data
An example in B^2

Hron et al. 2015, CSDA

Dataset of Age Densities (114 observations)



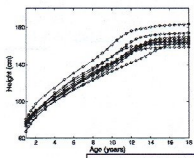
2. Smoothing and interpolation of functional data

Course Agenda

Smoothing is considered as a pre-process, so we smooth functions separately (even if the smoothing criterium is the same and the embedding space is the same) ⟹ we focus on a single datum (single function)

2. Smoothing and interpolation of functional data
From raw observations to functional data

- Typical observations of functional data are **discrete** and **noisy**. Indeed, the record of each function x_i usually consists of n_i pairs (t_{ij}, y_{ij}), with $j=1,\dots,n_i$.
- We model these pairs as $y_j = x(t_j) + \epsilon_j$. regression problem where the goal is to find $x(\cdot)$
  Note. The argument values t_{ij} may or may not be the same for each datum.
- For each i, we aim to reconstruct the underlying functional observation function x_i from the records (t_{ij}, y_{ij}), with $j=1,\dots,n_i$
  Note: The assumptions on the properties of x_i (e.g., the smoothness) will reflect on the way we proceed to reconstruct the data



Example: Height of 10 girls in Berkeley Growth data
- Raw data are depicted as symbols
- Reconstructed functional data are depicted as lines

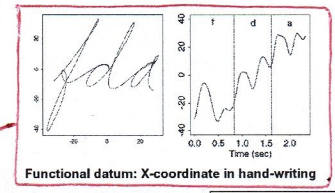
Ramsay Silverman 2005 Springer

From what we know about linear models, we already know that most of the focus will be on the error (what is the structure, what can we assume):

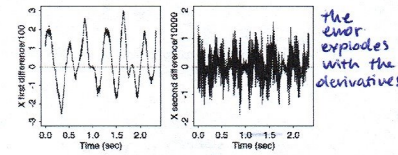
2. Smoothing and interpolation of functional data
From raw observations to functional data

- Depending on our prior knowledge on the measurement error (i.e., on the properties of the noise ϵ_j, we can decide to perform
  - Interpolation: the functional form reconstructed actually interpolates its discrete observations (noiseless measurements)
  - Smoothing: the functional form is smoother than the actual observations (noisy measurement)

In most cases, smoothing is preferred to interpolation.
Note. Differential operations (i.e., derivatives) amplify the effect of noise. Smoothing raw data actually enhances the estimation of derivatives



Functional datum: X-coordinate in hand-writing



First and second derivatives estimated via finite-differences

the error explodes with the derivatives

Ramsay Silverman 2005 Springer

the object "fda" is not a function but we transform it into a function: we track the x and y positions of the hand that is writing ⟹ we get two functions from it (one is represented (the x position))

when we interpolate we take into the model the noise. If we obtain a function with the noise inside, the derivative will be not precise.
⟹ interpolation is not a good idea if we think that there is some noise in our data and we want to use derivatives.

2. Smoothing and interpolation of functional data
Basic steps

the choice is based on what we want to do with the data (the analysis)

If we aim to interpolate or smooth discrete data we typically perform the following steps:
either:
- Choose a **target functional form** for x_i, that possibly depends on parameters
- **Estimate** the functional form, based on the pair (t_{ij}, y_{ij})

The **choice of the functional form** depends on various factors:
- **Features that we want to extract**: e.g., regularity of the functional form if the target is the differential information of the function (first, second derivatives)
- **Functional space embedding**: when we choose a Hilbert space embedding, we automatically identify possible orthonormal bases

In most cases:
- Hilbert space embedding is employed (especially, $H=L^2$)
- Functions are represented by basis functions

2.1. Basis functions
Representing data via basis functions

Most of the times what we do is think about a representation in terms of basis functions:
- Roughly speaking: a **system of basis functions** is a set of known functions that are linearly independent and allows us to approximate arbitrarily well any function as a linear combination of (a sufficiently large number of) K of these functions

- More precisely: given a system of basis functions ϕ_k, we will express a function x by the linear expansion

$$x(t) = \sum_{k=1}^{K} c_k \phi_k(t) \qquad [\phi_1 \cdots \phi_K] \begin{bmatrix} c_1 \\ \vdots \\ c_K \end{bmatrix}$$

or in matrix notation $x = \mathbf{c}'\boldsymbol{\phi} = \boldsymbol{\phi}'\mathbf{c}$.

Recall. In **Hilbert spaces**, we can always find an **orthonormal basis** that allows approximating, with any desired precision, any element of the space through the expansion

$$x = \sum_{n=1}^{K} \langle x, u_n \rangle u_n.$$

Note. In the following, we will mainly refer to L^2

Every separable Hilbert space has a countable basis. The representations we will work with will be FINITE representations (for computational purposes) ⟹ finite number of elements in the basis ⟹ we choose a sufficiently large number of basis functions and we project the data on the linear space generated by this finite number of basis functions

2.1. Basis functions
Fourier basis functions

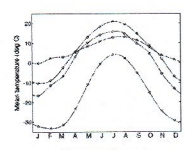
- One of the best known basis expansion in L^2 is provided by the **Fourier series**

$$\hat{x}(t) = c_0 + c_1 \sin\omega t + c_2 \cos\omega t + c_3 \sin 2\omega t + c_4 \cos 2\omega t + \dots$$

i.e., with the previous notation

$$\phi_0(t) = 1, \phi_{2r-1}(t) = \sin r\omega t, \text{ and } \phi_{2r}(t) = \cos r\omega t.$$

- **Properties**:
  - The basis is periodic, of period $2\pi/\omega$
  - If the values of t_j are equally spaced in T and the period is equal to the length of T than the basis is orthogonal (it can be made orthonormal via a proper rescaling)
- **Useful for:**
  - Extremely stable functions (i.e., no strong local features), for which uniformly smooth behavior is expected
  - Periodic data
- **Inappropriate for:**
  - Discontinuous functions (or with discontinuous derivatives)



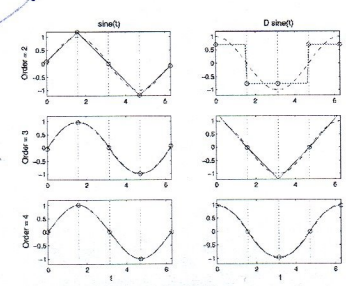
Temperatures in Canada

Ramsay Silverman 2005 Springer

2.1. Basis functions
Spline functions

or discontinuous functions (!)

- Spline functions are widely-used as approximation system for **non-periodic functional data**
- Construction of a ***m*-order spline**:
  - Divide the interval of definition T into L subintervals, i.e. fix a set of **knots**
  - Over each interval, the spline is defined as a polynomial or order m (# of constant to define the polynomial)
  - The polynomials are constrained as to guarantee that **adjacent polynomials join with continuity** in their values and in those of the derivatives up to order $m-2$



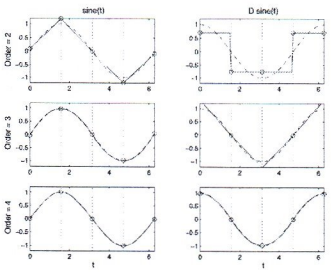
Ramsay Silverman 2005 Springer

the order m is equal to the degree of the polynomial function +1:
splines of order 3 ⟹ quadratic functions

2.1. Basis functions
Spline functions

- To gain **flexibility** in a spline one can increase the number of its knots, e.g., by locating more knots where the function exhibits more variability
- The **number of parameters** required to define a spline function with non-overlapping knots is the order plus the number of interior knots *m+L-1*

= degrees of freedom once we already stabilized all the constraints



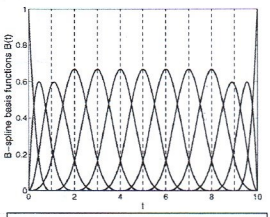
Ramsay Silverman 2005 Springer

2.1. Basis functions
B-Spline basis functions

- B-spline basis functions are systems of spline basis functions ϕ_k with the key properties:
  - Each basis function is a spline
  - A linear combination of the basis elements is a spline function
  - Any spline function can be expressed as a linear combination of these basis functions
- We call $B_k(t,\tau)$ a B-spline basis function in t with sequence of knots τ. A spline function is then defined as

$$S(t) = \sum_{k=1}^{m+L-1} c_k B_k(t,\tau)$$

- **Smoothing splines**: knots are placed at each argument value



Ramsay Silverman 2005 Springer

we have $\phi_1, \dots, \phi_K$ basis and (y_j, t_j) $j=1,\dots,n$. What we're trying to do is to evaluate the basis at every point t_j and then try to minimize the distance between the values of the basis at t_j with what we have observed.

2.2. Least square smoothing

- We defined basis systems, that allows us to express a functional datum as a linear combinations of these basis elements

$$x(t) = \sum_{k=1}^{K} c_k \phi_k(t)$$

- Our next goal is to estimate the parameters c_k from the observed pairs (t_j, y_j) under the model

$$y_j = x(t_j) + \epsilon_j.$$

or, in matrix form $x = \mathbf{c}'\boldsymbol{\phi} = \boldsymbol{\phi}'\mathbf{c}$.

Note 1. We can interpret this problem in the framework of classical linear models, and apply least square estimators.

Note 2. We smooth/interpolate one datum at a time, hence we here omit the index *i* of the statistical unit.

2.2. Least square smoothing
Ordinary least square fit

Goal: estimate the coefficient c_k of the linear model

$$x(t) = \sum_{k=1}^{K} c_k \phi_k(t) + \epsilon_j.$$

from the pairs (t_j, y_j).

- **Solution 1**: We minimize the sum of squared errors between fitted values and observations:

$$\text{SMSSE}(\mathbf{y}|\mathbf{c}) = \sum_{j=1}^{n} [y_j - \sum_{k}^{K} c_k \phi_k(t_j)]^2.$$

UNKNOWNS (β_k)

- From the theory of linear regression we know the solution of this problem

$$\hat{\mathbf{c}} = (\boldsymbol{\Phi}'\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{y}$$
$$\hat{\mathbf{y}} = \boldsymbol{\Phi}\hat{\mathbf{c}} = \boldsymbol{\Phi}(\boldsymbol{\Phi}'\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{y}.$$

Note. $\hat{\mathbf{y}}$ is the projection of $\mathbf{y}$ over the space generated by the columns of $\boldsymbol{\Phi}$, that are the evaluations of the basis functions in the measurement points.

It's exactly a linear model!

we have:

but it's not that the only thing that we can do is fit a line! We're transforming the $\mathbf{t}$'s using the basis functions and then we're fitting the transformed data

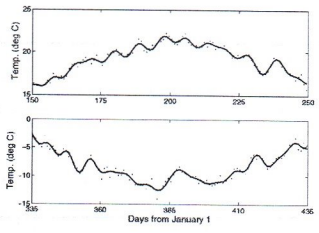
The design matrix $\boldsymbol{\Phi}$ is simply the transformation of the datum $\mathbf{t}_j$ through the basis function:

$$\boldsymbol{\Phi} = \begin{bmatrix} \phi_1(t_1) & \dots & \phi_K(t_1) \\ \phi_1(t_2) & \dots & \phi_K(t_2) \\ \vdots & & \\ \phi_1(t_n) & \dots & \phi_K(t_n) \end{bmatrix} \in \mathbb{R}^{n \times k}$$

2.2. Least square smoothing
Ordinary least square fit

- Ordinary least squares are appropriate if the measurement error may be assumed to be iid
- The degree of smoothness of the estimated curve depends on the number of basis functions employed

Example. Smoothing of temperature data in Montreal using 109 Fourier basis functions



- We choose a Fourier basis because data are periodic
- We truncate the basis to 109 basis functions, that allows to catch (109-1)/2=54 different harmonic frequencies (about 1 per week)
- Performing OLS estimate means that the noise in the observations is iid across the days

Ramsay Silverman 2005 Springer

choosing **k** it's like choosing the variables in the linear model ⇒ we can use the tools for variables selection

2.2. Least square smoothing
Weighted least square fit

If data are not iid (e.g., there is autocorrelation in the measurement process), we can use a weighted least squares.

- **Solution 2**: We minimize the weighted sum of squared errors between fitted values and observations:

$$\text{SMSSE}(\mathbf{y}|\mathbf{c}) = (\mathbf{y} - \boldsymbol{\Phi}\mathbf{c})'\mathbf{W}(\mathbf{y} - \boldsymbol{\Phi}\mathbf{c})$$

- Matrix $\mathbf{W}$ is assumed to be positive definite, and can be set e.g. to the covariance matrix of the errors $\mathbf{W} = \boldsymbol{\Sigma}_e^{-1}$.
- The solution of this minimization problem is found as

$$\hat{\mathbf{c}} = (\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{W}\mathbf{y}$$

2.2. Least square smoothing
Sampling variances and confidence limits

- Approximate point-wise confidence intervals can be built based upon the estimated model
- As in classical linear models, the variance of the estimator for the coefficients is

$$\mathtt{Var}[\mathbf{c}] = (\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}'\mathbf{W}\mathbf{\Sigma}_e\mathbf{W}\mathbf{\Phi}(\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi})^{-1}$$

which in case of unweighted least squares and iid errors reduces to

$$\mathtt{Var}[\mathbf{c}] = \sigma^2(\mathbf{\Phi}'\mathbf{\Phi})^{-1}$$

- The variance of the point-wise estimate of the curve is then obtained as the diagonal of the matrix

$$\mathtt{Var}[\hat{\mathbf{y}}] = \mathbf{\Phi}\mathtt{Var}[\mathbf{c}]\mathbf{\Phi}'$$

which in case of unweighted least squares and iid errors reduces to

$$\mathtt{Var}[\hat{\mathbf{y}}] = \sigma^2\mathbf{\Phi}(\mathbf{\Phi}'\mathbf{\Phi})^{-1}\mathbf{\Phi}' = \sigma^2\mathbf{S}$$

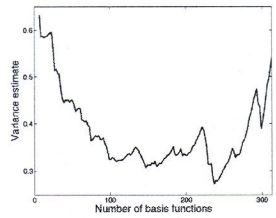
- The variance of the errors can be estimated from the residual sum of squares

$$s^2 = \frac{1}{n-K}\sum_j^n (y_j - \hat{y}_j)^2$$

2.2. Least square smoothing
Sampling variances and confidence limits

- To choose K, one may evaluate when a drop in sampling variance occurs for a range of candidate K

Example. Smoothing of temperature data in Montreal



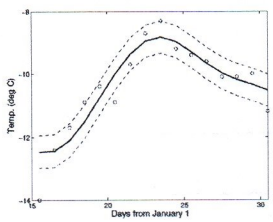
- A drop in variance is obtained around 100 basis functions
- We truncated the basis to 109 basis functions, that allowed to catch (109-1)/2=54 different harmonic frequencies (about 1 per week)
- Lower variances may be obtained but overfitting might occurs then

Ramsay Silverman 2005 Springer

2.2. Least square smoothing
Sampling variances and confidence limits

- Based on previous expressions, approximate confidence limits may be built

Example. Smoothing of temperature data in Montreal



- Confidence limits are build summing/subtracting 2 standard deviations
- Quantiles of the normal can be used instead
- Confidence limits must be interpreted point-wise

Ramsay Silverman 2005 Springer

2.2. Least square smoothing
Bias variance trade-off

- A key point of least square smoothing is how to set the order of the basis expansion. Algorithms to set K can be borrowed from the context of linear regression (e.g., step-wise algorithms). Nevertheless, one should pay close attention to the fact that:
  - The larger K, the better the fit to the data, but higher risk to fit the noise (or non-interesting variations)
  - If K is too small we may miss important features of the underlying function that we wish to estimate
- In fact, as in linear regression, we have a bias/variance trade-off

$$\mathtt{Bias}[\hat{x}(t)] = x(t) - \mathrm{E}[\hat{x}(t)],$$

$$\mathtt{Var}[\hat{x}(t)] = \mathrm{E}[\{\hat{x}(t) - \mathrm{E}[\hat{x}(t)]\}^2]$$

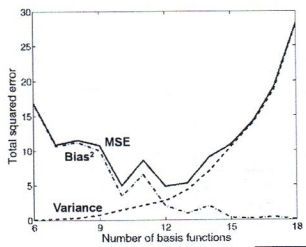
  - For large values of K the bias is small, the variance is high
  - For small values of K, the bias is high, the variance is low

2.2. Least square smoothing
Mean-squared error

- The mean-squared error summaries what we actually would like to minimize

$$\mathtt{MSE}[\hat{x}(t)] = \mathrm{E}[\{\hat{x}(t) - x(t)\}^2] = \mathtt{Bias}^2[\hat{x}(t)] + \mathtt{Var}[\hat{x}(t)]$$

Example. Bias/variance trade of a simulated example inspired by the Berkeley Growth Study



Ramsay Silverman 2005 Springer

We can think in terms of how smooth we would the function to be and we introduce a penalization when we are far from the smoothed situation (this is in the same style of Ridge/Lasso)

2.3. Smoothing with a differential penalization
Penalized regression

- We now focus on estimating a non-periodic function $x(t)$ on the basis of a vector $\mathbf{y}$ of discrete and noisy observations.
  Note: we are not (yet) assuming any functional form for $x(t)$
- A way to approach the bias/variance trade-off is to impose a certain degree of smoothing on the curve (this reduces the variance at the expense of increasing bias)
- A popular way to do this is to quantify the notion of **roughness** through a **differential property of the curve**, and perform a **regression with** the corresponding **penalization**

Let us quantify *roughness* through the second derivative

$$\mathtt{PEN}_2(x) = \int [D^2x(s)]^2\,ds$$

Measure of ***curvature*** of the function (*$PEN_2(x)$=0* if x is a straight line)

Given λ, find $x(t)$ that minimizes

$$\mathtt{PENSSE}_\lambda(x|\mathbf{y}) = [\mathbf{y} - x(\mathbf{t})]'\mathbf{W}[\mathbf{y} - x(\mathbf{t})]^2 + \lambda \times \mathtt{PEN}_2(x)$$

Penalized SSE

more basis functions ⟹ more overfitting but more smoothness: the obvious thing to do is to set a criterium that'll consider (how far we are from data)^2 and a piece taking care of smoothness and a penalization will be to penalize the 2nd derivative of the curve

2.3. Smoothing with a differential penalization
Penalized regression

- Let's give a closer look to the penalized SSE functional

$$\texttt{PENSSE}_\lambda(x|\mathbf{y}) = [\mathbf{y} - x(\mathbf{t})]'\mathbf{W}[\mathbf{y} - x(\mathbf{t})]^2 + \lambda \times \texttt{PEN}_2(x)$$

- Parameter λ is called *smoothing parameter* and controls the importance of the penalization with respect to the residual sum of squares:
  - If $\lambda \to \infty$ the functional gives emphasis to the penalization and the fitted curve will be a straight line ($\texttt{PEN}_2(x) = 0$.)
  - If $\lambda \to 0$. the curve approaches the smoothest twice-differentiable curve that interpolates the data

- **Key result** (de Boor, 2002): the curve x that minimizes $\texttt{PENSSE}_\lambda(x|\mathbf{y})$ is a cubic spline with knots at the data points t_j
  → *the functional form of x is a consequence of the objective function!*

- **Common computational technique:** use a four order B-spline basis (called *cubic spline*) and minimize the $\texttt{PENSSE}_\lambda(x|\mathbf{y})$ with respect to the coefficients of the expansion

$$x(t) = \sum_{k=1}^{K} c_k \phi_k(t)$$

λ set via GCV.

2.3. Smoothing with a differential penalization
Penalized regression – computational details

- We can re-express the penalization as

$$\begin{aligned}\texttt{PEN}_m(x) &= \int [D^m x(s)]^2 \, ds \\ &= \int [D^m \mathbf{c}'\boldsymbol{\phi}(s)]^2 \, ds \\ &= \mathbf{c}'\mathbf{R}c \,,\end{aligned}$$

with $\mathbf{R} = \int D^m\boldsymbol{\phi}(s) D^m\boldsymbol{\phi}'(s)\, ds$.

- Plugging this expression in the objective function yields

$$\texttt{PENSSE}_m(\mathbf{y}|\mathbf{c}) = (\mathbf{y} - \boldsymbol{\Phi}\mathbf{c})'\mathbf{W}(\mathbf{y} - \boldsymbol{\Phi}\mathbf{c}) + \lambda \mathbf{c}'\mathbf{R}\mathbf{c} \,.$$

that is minimized for

$$\hat{\mathbf{c}} = (\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi} + \lambda\mathbf{R})^{-1}\boldsymbol{\Phi}'\mathbf{W}\mathbf{y} \,.$$

same form that we get for $\hat{\beta}$ with ridge regression (with GLS) (except that $R = I$)

by using an optimization criterium when we fit our data we end up with a representation of the data which automatically embedds our data in a space (!)

Example: if we want to penalize the second derivative → we know that there is a result that says the curve that penalizes the least square criterium + penalize the second derivative is cubic splines if we put the knots in every point we have observed

2.3. Smoothing with a differential penalization
Penalized regression

Generalized cross-validation

$$\texttt{GCV}(\lambda) = \frac{n^{-1}\,\texttt{SSE}}{[n^{-1}\,\texttt{trace}\,(\mathbf{I} - \mathbf{S}_{\phi,\lambda})]^2}$$

with

$$\hat{\mathbf{y}} = \boldsymbol{\Phi}(\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi} + \lambda\mathbf{R})^{-1}\boldsymbol{\Phi}'\mathbf{W}\mathbf{y} = \mathbf{S}_{\phi,\lambda}\mathbf{y}$$

2.3 Closing remarks

- We have seen basis expansions as a way to smooth raw functional observations
- Many other bases and smoothing techniques are available in the literature, e.g.,
  - Local polynomial smoothing: LS smoothing on neighborhoods of the point t_j through polynomial basis

$$\texttt{SMSSE}_t(\mathbf{y}|\mathbf{c}) = \sum_{j=1}^{n} w_j(t)[y_j - \sum_{k=1}^{K} c_k\phi_k(t_j)]^2,$$

$$w_t(t) = \texttt{Kern}\left(\frac{t_t - t_j}{h}\right)$$

Uniform:	$\texttt{Kern}(u) = 0.5$ for $\lvert u\rvert \le 1$,	0 otherwise
Quadratic:	$\texttt{Kern}(u) = 0.75(1-u^2)$ for $\lvert u\rvert \le 1$,	0 otherwise
Gaussian:	$\texttt{Kern}(u) = (2\pi)^{-1/2}\exp(-u^2/2)$.	

2.3 Closing remarks

- We have seen basis expansions as a way to smooth raw functional observations
- Many other bases and smoothing techniques are available in the literature, e.g.,
  - Local polynomial smoothing: LS smoothing on neighborhoods of the point t_j through polynomial basis
  - Wavelet bases
- Ad hoc smoothing techniques need to be employed in case of constrained data, e.g.,
  - Monotonically increasing functions
  - Probability density functions
- Smoothing or interpolation is the very first step of a functional data analysis and all the subsequent results depend on this step.
  → One should pay close attention in applying the most appropriate technique for smoothing the data

3. FDA & Dimensionality reduction in Hilbert spaces

Course Agenda

1. Hilbert space model for functional data
   - 1.1. Basics notions on Hilbert spaces
   - 1.2. Hilbert space embedding for functional data
   - 1.3. Formal definition of functional data
2. Smoothing and interpolation of functional data
   - 2.1. Basis function
   - 2.2. Least square smoothing
   - 2.3. Smoothing with a differential penalization
3. **FDA & Dimensionality reduction in Hilbert spaces**
   - **3.1. Functional Principal Components in Hilbert spaces**
   - **3.2. Examples in L2**
4. Data alignment and clustering
   - 4.1 Phase and amplitude variability
   - 4.2 Landmark and continuous registration
   - 4.3 Decoupling phase and amplitude variability
   - 4.4 K-mean alignment

Find the directions s.t. the scores on the directions have a maximized variability (w.r.t. all other directions)

Recall: Principal Component Analysis

Problem: Given a dataset of *N* zero-mean multivariate observations in $\mathbb{R}^p$, $X_1, \dots, X_N$, find the orthonormal directions $a_1, \dots, a_p$ of maximum variability (for the dataset).

Equivalently, for *k= 1,...,p*, find:

$$\mathbf{a}_k = argmax_{\mathbf{a}\in\mathbb{R}^p} Var(\mathbf{a}'\mathbf{X})$$
$$\text{subject to:} \quad \mathbf{a}'\mathbf{a} = 1,\ \mathbf{a}_j'\mathbf{a} = 0 \text{ for } j < k$$

- We can re-write the problem as

$$\mathbf{a}_k = argmax_{\mathbf{a}\in\mathbb{R}^p} \frac{1}{N}\sum_{i=1}^{N}(\mathbf{a}'\mathbf{X}_i)^2$$
$$\text{subject to:} \quad \mathbf{a}'\mathbf{a} = 1,\ \mathbf{a}_j'\mathbf{a} = 0 \text{ for } j < k$$

or, equivalently

$$\mathbf{a}_k = argmax_{\mathbf{a}\in\mathbb{R}^p} \frac{1}{N}\sum_{i=1}^{N} < \mathbf{a}, \mathbf{X}_i >^2$$
$$\text{subject to:} \quad \|\mathbf{a}\| = 1,\ < \mathbf{a}_j, \mathbf{a} >= 0 \text{ for } j < k$$

Note 1. We assume *N>p* and absence of collinearity, i.e. the data matrix is full rank.

Note 2 . If $X_1, \dots, X_N$ are not zero-mean, they can be centered by subtracting the (sample) mean. For unbisasedness, divide by *N-1* instead of *N*.

this is something we'll have to change since functional data are belonging to infinite dimensional space

Recall: Principal Component Analysis

Problem: Given a dataset of *N* zero-mean multivariate observations in $\mathbb{R}^p$, $X_1, \dots, X_N$, find the orthonormal directions $a_1, \dots, a_n$ of maximum variability, i.e., those solving for k=1,...p,

$$\mathbf{a}_k = argmax_{\mathbf{a}\in\mathbb{R}^p} \frac{1}{N}\sum_{i=1}^{N} < \mathbf{a}, \mathbf{X}_i >^2$$
$$\text{subject to:} \quad \|\mathbf{a}\| = 1,\ < \mathbf{a}_j, \mathbf{a} >= 0 \text{ for } j < k$$

Solution: Call *S* the sample covariance matrix of $X_1, \dots, X_N$. Then, the **principal components** are found as the eigenvectors of the matrix *S*; for *k=1,..., p*, they solve the eigen-equation

$$Sc_k = \lambda_k c_k$$

The eigenvalue λ_k associated with the eigenvector c_k represents the variability along the direction c_k .

Note. We call *score* x_{ik} the projection of the observation X_i along the direction c_k, i.e.,

$$x_{ik} = \langle X_i, e_k \rangle = X_i' e_k$$

Recall: Principal Component Analysis

Problem: Given a dataset of *n* zero-mean multivariate observations in $\mathbb{R}^p, X_1, \dots, X_N$, find the directions of maximum variability of the dataset, i.e., those maximizing

$$\frac{1}{N}\sum_{i=1}^{N} \langle a, X_i \rangle^2 \quad \text{subject to } \|a\| = 1$$

Can we do the same in *any* Hilbert space, using its inner product?

The eigenvalue λ_k associated with the eigenvector c_k represents the variability along the direction c_k.

Note. We call score x_{ik} the projection of the observation X_i along the direction c_k, i.e., $x_{ik} = \langle X_i, e_k \rangle = X_i' e_k$

Yes, since we're only using inner products

Given a dataset of observations in a Hilbert space (zero-mean) we want to find the direction (which is a vector in Hilbert space) s.t. when we project our data on this direction and we compute the variability of the scores, this variability is maximized.

Functional Principal Component Analysis
Problem statement

Problem: Given a dataset of *N* zero-mean functional observations in *H*, $X_1, \dots, X_N$, find the directions of maximum variability (in *H*) of the dataset, i.e., for *k=1,...,N*, find ξ_k maximizing

$$\frac{1}{N}\sum_{i=1}^{N} < \xi, \mathbf{X}_i >_H^2$$
$$\text{subject to:} \quad \|\xi\| = 1,\ < \xi_j, \xi >_H = 0 \text{ for } j < k$$

- We look for an orthonormal system in *H* maximizing the *variability* of the corresponding projections
- Indeed, $\langle \xi, X_i \rangle_H$ is the projection of X_i «along the direction» ξ (i.e., a «direction» in *H*). Note that $\langle \xi, X_i \rangle_H$ is a scalar, hence $\frac{1}{N}\sum_{i=1}^{N} \langle \xi, X_i \rangle_H^2$ is a sample variance in the usual sense.

Note 1. If the data are not zero-mean, they can be centered by subtracting the (sample) mean. *N* should then be replaced by *N-1*.

Note 2. If data are linearly independent and centered on the sample mean, we can find at most *N-1* principal components.

here the directions (ξ_k) are elements of the Hilbert space

Functional Principal Component Analysis
Sample covariance operator

Problem: Given a dataset of *n* zero-mean functional observations in *H*, $X_1, \dots, X_N$, find the directions of maximum variability (in *H*) of the dataset, i.e., for *k=1,...,N*, find ξ_k maximizing

$$\frac{1}{N}\sum_{i=1}^{N} < \xi, \mathbf{X}_i >_H^2$$
$$\text{subject to:} \quad \|\xi\| = 1,\ < \xi_j, \xi >_H = 0 \text{ for } j < k$$

- As in multivariate principal component analysis, **functional principal components** are related with the eigen-decomposition of the functional counterpart of the (sample) covariance matrix
- Recall that the **sample covariance operator** is defined as

$$Sx = \frac{1}{N}\sum_{i=1}^{N} \langle X_i, x \rangle X_i, \quad x \in H$$

In L^2 it is equivalently defined as

$$[Sx](t) = \int_T \widehat{c}(s,t)x(s)d(s)], \quad x \in L^2 \quad \text{with} \quad \widehat{c}(s,t) = \frac{1}{N}\sum_{i=1}^{N} X(s)X(t)$$

Note. If data are centered on the sample mean, divide by *N-1* for unbiasedness.

Functional Principal Component Analysis
FPCA and sample covariance operator

Problem: Given a dataset of *n* zero-mean functional observations in *H*, $X_1, \dots, X_N$, find the directions of maximum variability (in *H*) of the dataset, i.e., for *k=1,…,N*, find ξ_k maximizing

$$\frac{1}{N}\sum_{i=1}^{N} < \xi, \mathbf{X}_i >_H^2$$

$$\text{subject to:} \quad ||\xi|| = 1, \ < \xi_j, \xi >_H = 0 \text{ for } j < k$$

Solution: Let *S* be the sample covariance operator of $X_1, \dots, X_N$. Then, the **functional principal components** $\xi_1, \dots, \xi_N$ are found as the eigenfunctions of the operator *S*, i.e., they solve the eigen-equations

$$S\xi_k = \lambda_k \xi_k$$

The eigenvalue λ_k associated with the eigenvector ξ_k represents the variability along the direction ξ_k.

We call *functional score* x_{ik} the projection of the observation X_i along the direction ξ_k, i.e.,

$$x_{ik} = < X_i, \xi_k >$$

Note. If data are centered on the sample mean, we can find at most *N-1* principal components

The directions identifying principal components will be functions ⟹ we can look at them as they were the extension (∞-dimension extension) of our loadings vector

Functional Principal Component Analysis
Dimensionality reduction and Interpretation of the results

- To reduce the dimensionality of the dataset one can proceed as in the multivariate setting, e.g., by looking for an elbow in the cumulative percentage of total variance explained by the first *p* functional principal components.

$$CPV(p) = \frac{\sum_{k=1}^{p} \hat{\lambda}_k}{\sum_{k=1}^{N} \hat{\lambda}_k}.$$

- Other useful plots are the boxplots of the scores along the first *p* directions, to investigate the possible presence (and influence) of outliers on the results
- Interpretation of the loadings can be performed by:
  - Plotting the loadings themselves (*only for expert users*)
  - Plotting the mean +/- the eigenfunctions multiplied by a proper constant, e.g., the std. along the component, which corresponds to the sqrt of the eigenvalue:

$$\overline{X} \pm \sqrt{\lambda_k}\xi_k$$

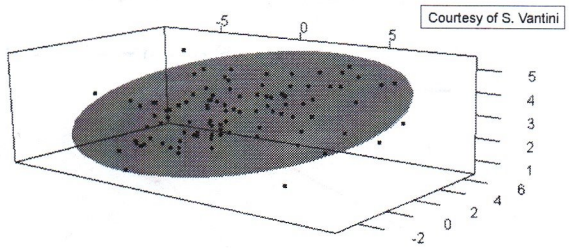
  - Plotting the projection of the dataset along each component or along the first *p* components

$$\overline{X} + x_{ik}\xi_k$$

$$\overline{X} + \sum_{k=1}^{p} x_{ik}\xi_k$$

we can look directly at how the PC's modify the information that is given by the mean: we can plot the mean function and then we add a principal component (for example ξ_1) multiplied by a number ⟹ we have an interpretation about what happens if the score of a statistical unit is positive or negative w.r.t. that component

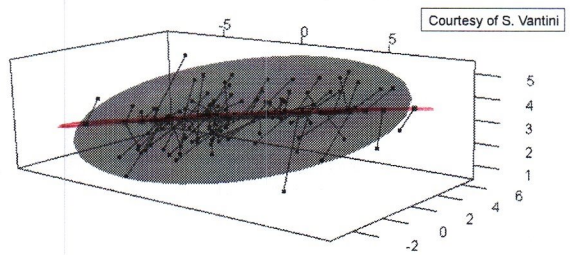
Functional Principal Component Analysis
FPCA as space of best approximation



Problem: find the space of dimension *k* that best approximate the data in the mean square sense

If k=0: sample mean

Functional Principal Component Analysis
FPCA as space of best approximation



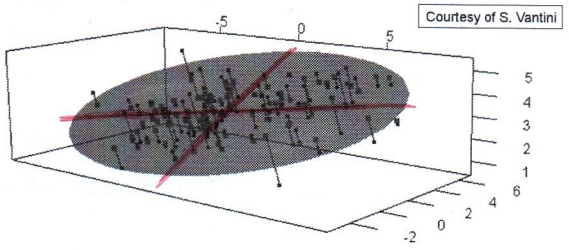
Problem: find the space of dimension *k* that best approximate the data in the mean square sense

If k=1: linear space generated by the first FPC

Principal components can be interpreted as the basis system which identifies the linear space in **H** which is closest to the dataset:

- 0 linear space: mean
- 1 linear space: 1st principal component
- 2 linear space: plane identified by PC1, PC2

Functional Principal Component Analysis
FPCA as space of best approximation



Problem: find the space of dimension *k* that best approximate the data in the mean square sense

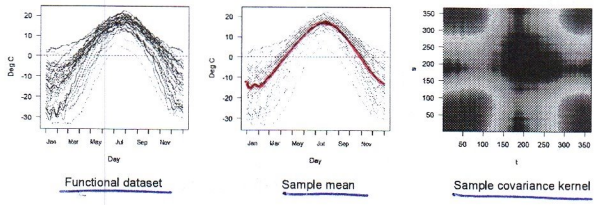
If k=2: linear space generated by the first two FPCs

…

3.2 Examples in L^2
Dataset of Canadian temperatures

Ramsay Silverman 2005 Springer

Example. Dataset of Temperatures in Canada (35 observations)



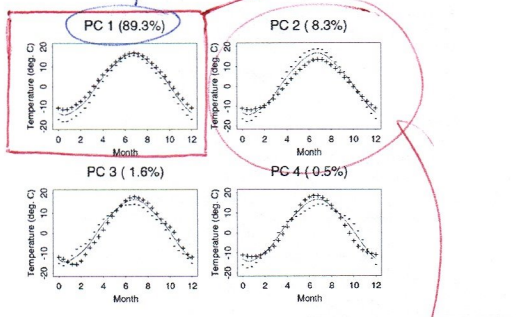
Functional dataset | Sample mean | Sample covariance kernel

PC1 is explaining 89.3% of the variability of the dataset

3.2 Examples in L^2

Dataset of Canadian temperatures

Ramsay Silverman 2005 Springer



Ramsay Silverman 2005 Springer

the continuous line is the mean. If we take the mean and we add one standard deviation in the direction of the first principal component we get the line identified by "+++" (⇒ more or less the same shape except that is higher). if we move one standard deviation but with the minus sign we obtain the line identified by "---", so the first principal component is discriminating cold places (--) vs. hot places (++) (since the mean is the temperature)

the second principal component needs to explain the contrast between the zones that have temperature at first below the mean, then above and then below again (--) vs. the zones that have temperature at first above then below then above again (++). It's capturing 8.3% of variability

4. Data alignment and clustering

→ only with functional data

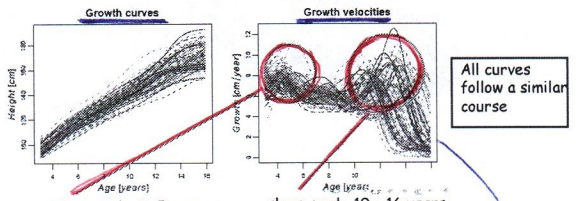
Course Agenda

1. Hilbert space model for functional data
   - 1.1. Basics notions on Hilbert spaces
   - 1.2. Hilbert space embedding for functional data
   - 1.3. Formal definition of functional data
2. Smoothing and interpolation of functional data
   - 2.1. Basis function
   - 2.2. Least square smoothing
   - 2.3. Smoothing with a differential penalization
3. FDA & Dimensionality reduction in Hilbert spaces
   - 3.1. Functional Principal Components in Hilbert spaces
   - 3.2. Examples in L2
4. **Data alignment and clustering**
   - 4.1 Phase and amplitude variability
   - 4.2 Landmark and continuous registration
   - 4.3 Decoupling phase and amplitude variability
   - 4.4 K-mean alignment

4.1 Misalingned data: the Berkeley growth study data

(Tuddenham and Snyder, 1954)

- the data include the heights of 93 children, 54 girls and 39 boys, measured on 31 time instances, not equally spaced;
- the functional form of these data has been reconstructed using monotone smoothing splines. (to pass from discrete to continue data)



All curves follow a similar course

minor peak, 2 -5 years (mid-spurt)

sharp peak, 10 - 16 years (pubertal-spurt)

However, each child follows his/her own biological clock

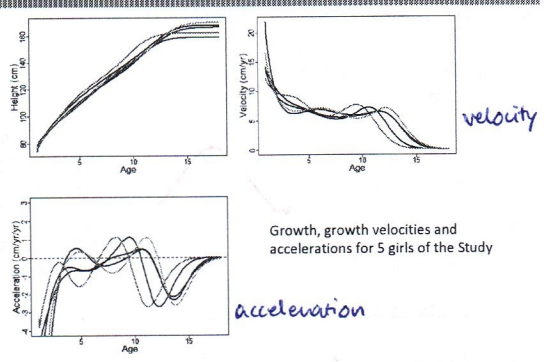
it looks like every child is going through the same pattern of growing but at different time (every child has his/her biological clock)

although the curves have a similar shape, it's clear that there are differences in when some landmarks happen:

similar but "translated"

we can see it even better if we focus on the growth of 5 girls:

4.1 The growth curves of 5 girls



growth

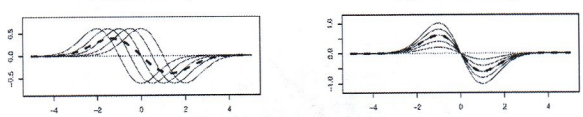
velocity

acceleration

Growth, growth velocities and accelerations for 5 girls of the Study

4.1 Phase and amplitude variability

Ramsay Silverman 2005 Springer



Phase variability: different curves exhibit more or less the same features but these features occur at different times or space locations for different statistical units.

If not taken properly into account, the misalignment acts as a confounding factor and may blur subsequent analyses.

→ with functional data we have 2 variabilities:

- the one we know (the variability of heights in this case)
- variability along the ascissa, the variability related to WHEN this phenomenon happens (the time/point in space where this phenomenon happens)

→ variability on the x axes

PHASE variability and AMPLITUDE variability

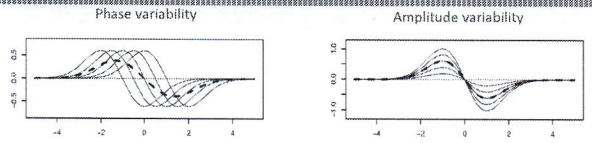
→ variability on the y axes

in general: transformation of the abscissa

If we have n curves: $c_1(t), \dots, c_n(t)$ we need to find a transformation for the x axes (WARPING FUNCTIONS) in order to make the curves the more similar possible

4.1 Phase and amplitude variability

Ramsay Silverman 2005 Springer



Registration of a set of functions

Given n curves $c_1(t), \dots, c_n(t)$, find suitable warping functions $h_1(t), \dots, h_n(t)$ such that $c_1(h_1(t)), \dots, c_n(h_n(t))$ are the most similar.

The functions h_i should be increasing; they capture the phase variability. Amplitude variability is the remaining variability among the aligned curves in the vertical direction.

In some cases, time or location is merely shifted from curve to curve, for example, because the measurements are started at random time points. For these situations, it is natural to use $h_i(t)=t+delta_i$. In other situations, phase variation is a matter of dilation, in which case $h_i(t)=alpha_i t$ is a natural choice of warping function. In yet other situations, the time or space deformation is more complex.

4.1 Phase and amplitude variability

Ramsay Silverman 2005 Springer

Example:

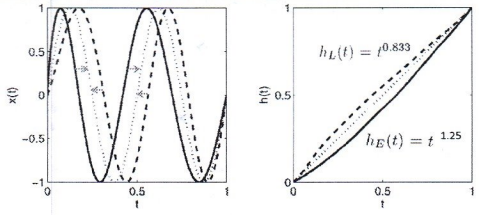
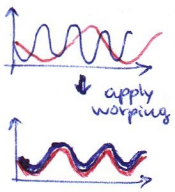


Figure 7.9. The left panel shows the target function, $x_0(t) = \sin(4\pi t)$, as a dotted line; an early function, $x_E(t) = \sin(4\pi t^{0.8})$, as a solid line; and a late function, $x_L(t) = \sin(4\pi t^{1.2})$, as a dashed line. The corresponding warping functions that register the early and late curves to the target are shown in the right panel.

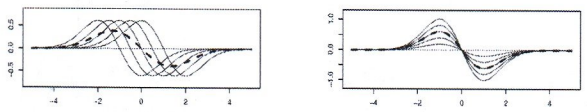


apply warping fcts.

one warping function is making the curve to move more slowly and one warping function is making the curve to move faster
⟹ the curves will be aligned

4.1 Phase and amplitude variability

Ramsay Silverman 2005 Springer



Registration of a set of functions

Find **suitable** warping functions $h_1, \dots, h_n$ such that $c_1 \circ h_1, \dots, c_n \circ h_n$ are the most **similar**.

⟹ **Landmark Approach**: known **landmarks** along the curves that are aligned so that landmarks occurs at the same abscissa points.

⟹ **Continuous Approach**: define a measure of similarity/dissimilarity between curves, that are aligned in order to maximize/minimize their similarity/dissimilarity.

4.2 Landmark registration

Ramsay Silverman 2005 Springer

Landmarks: significant (univocally identifiable) shape-events in a curve, e.g. crossings of zero, peaks, valleys, points of inflection.

$c_1, \dots, c_n$, where $c_i : [0, T] \to \mathbb{R}^d$

Suppose

- L landmarks; for the i–th curve, located at $t_{i1}, \dots, t_{iL}$
- a template curve c_0 is available with landmark locations $t_{01}, \dots, t_{0L}$
  If not, we can define t_{0j} as the average of the t_{ij}'s

Warping function for the $i-th$ curve: any strictly increasing function h_i s.t.

- $h_i(0) = 0$
- $h_i(t_{0j}) = t_{ij}$, for $j = 1, \dots, L$
- $h_i(T) = T$

Notation: the warping functions will be the inverse of these h_i

▶ $(0, 0), (t_{01}, t_{i1}), \dots, (t_{1L}, t_{iL}), (T, T)$: interpolated by a piece-wise line, a polygon or higher order monotone splines (strictly increasing)

template curve to which we want to align all the curves in the sample (if we don't have a template we take the average position for every landmark)

any warping function that will allignc these landmarks

clear what is going on but it's not so clear how to do it; maybe we have functions for which there is no peak or maybe the peaks are so small that many of them are candidates to become the peak that should be aligned with the others
⟹ it's a difficult approach to use

4.2 Continuous registration

- Landmark-based registration may require significant user input and can be sensitive to the accuracy of the landmark identification.

- In some applications it is not possible to identify well-defined features that can be taken as landmarks

Alternative strategy: Continuous registration

Main idea:

- definition of a suitable distance (or closeness) measure between curves, which measures dissimilarity (or similarity) between curves.

- the curves are thus aligned by warping their time or space abscissa parameters choosing the optimal warping function in some class of admissible warping functions in order to minimize the final distance among the curves or, equivalently, maximize their final similarity.

= group of transformations that are allowed as warping transformations

4.2 Continuous registration

The problem of decoupling amplitude and phase variability is not univocally defined as different measures of distance or similarity between curves can be considered, as well as different classes of admissible warping functions (e.g., simple translations or dilations, increasing linear transformations or more complex increasing transformations), leading to different registration results.

The choice of the couple formed by dissimilarity/similarity measure and admissible warping functions defines the distinction between phase variability and amplitude variability in the specific problem under analysis.

This choice must thus be problem specific.

Decoupling amplitude and phase variability depends on the choices that we make on dissimilarities between curves and warping functions that are allowed

⟹ there is not just one solution to the decoupling problem because we can always chose different type of dissimilarity and different transformations

We have some independent requirements for ρ and W ((1)) and then we have to put together the two things ((2)) $\Rightarrow$ the two things (ρ, W) shouldn't be chosen independently

4.3 Decoupling phase and amplitude variabilities

Sangalli, Secchi, Vantini, Veneziani 2009 JASA

(ρ, W) must satisfy properties that ensure that the aligning problem is well-posed and the corresponding procedure is coherent

(1)
- ρ
  - Bounded
  - Reflexive
  - Symmetric
  - Transitive
- W
  - Convex vector space
  - Group structure with respect to function composition

ρ → dissimilarities between curves

W → group of functions that are allowed for warping → two warping combined make another warping

(2)
- (ρ, W) Properties of coherence

$$\rho(c_1, c_2) = \rho(c_1 \circ h, c_2 \circ h), \quad \forall\, h \in W$$

W-invariance of the index (Isometry of the group, parallel orbits)

→ $\rho(c_1 \circ h_1, c_2 \circ h_2) = \rho(c_1 \circ h_1 \circ h_2^{-1}, c_2) = \rho(c_1, c_2 \circ h_2 \circ h_1^{-1})$ = aligning c_1 to c_2 is the same as aligning c_2 to c_1

(ρ, W) defines on the considered set of functions $\mathcal{C}$ a partition in equivalence classes

4.3 Decoupling phase and amplitude variabilities

Vantini 2012 TEST
Sangalli Secchi Vantini 2014b EJS

dissimilarity d	warpings W
$\|c_1 - c_2\|$	W_{shift}
$\|c_1' - c_2'\|$	W_{shift}
$\|(c_1 - \bar{c_1}) - (c_2 - \bar{c_2})\|$	W_{shift}
$\left\|(c_1' - \bar{c_1'}) - (c_2' - \bar{c_2'})\right\|$	W_{shift}
$\left\|\frac{c_1}{\|c_1\|} - \frac{c_2}{\|c_2\|}\right\|$	$W_{affinity}$
$\left\|\frac{c_1'}{\|c_1'\|} - \frac{c_2'}{\|c_2'\|}\right\|$	$W_{affinity}$
$\left\|\text{sign}(c_1')\sqrt{\|c_1'\|} - \text{sign}(c_2')\sqrt{\|c_2'\|}\right\|$	$W_{diffeomorphism}$

couples of dissimilarities and warping functions that are coherent (= that respect all the ties we mentioned before)

if we change $c_1(t)$ and $c_2(t)$ with the same warping function $h(\cdot)$ we want that

$$\text{dissimilarity}\left(c_1(t), c_2(t)\right) = \text{dissimilarity}\left(c_1(h(t)), c_2(h(t))\right)$$

$$\left(\rho(c_1, c_2) = \rho(c_1 \circ h,\ c_2 \circ h)\right)$$

4.3 Curve alignment: iterative procedure

Sangalli Secchi Vantini Vitelli 2010 CSDA

- If a template (prototype) curve φ is known, then it is enough to align each curve to this template
- If the template is unknown then it must be estimated from the data, leading to a complex optimization problem

$$\text{find } \varphi \in \mathcal{C} \text{ and } \underline{\mathbf{h}} = \{h_1, \dots, h_N\} \subset W \text{ such that}$$
$$\frac{1}{N}\sum_{i=1}^{N} \rho(\varphi, c_i \circ h_i) \geq \frac{1}{N}\sum_{i=1}^{N} \rho(\psi, c_i \circ g_i)$$
$$\text{for any other } \psi \in \mathcal{C} \text{ and } \mathbf{g} = \{g_1, \dots, g_N\} \subset W$$

we have a set of mis-aligned curves, we want to find a template to which align our curves but we want to find the warping functions! $\Rightarrow$ we have to find
- template
- warping functions

such that: the average similarity when we warp the functions to the template (unknown) is larger than what we would get for any other possible choice of template and warping

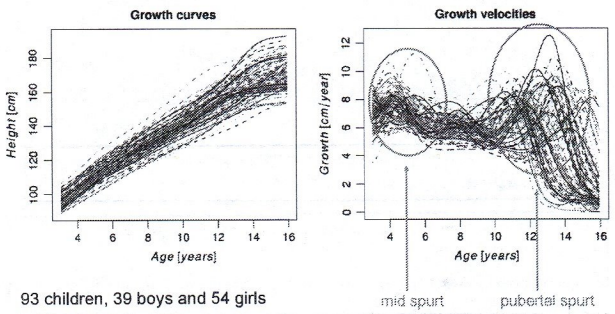
it can be done iteratively: Iterative PROCRUSTES procedure that alternates:
- template estimation step: the template centerline is estimated from the curves obtained in the previous alignment step
- alignment step: the centerlines are aligned to the temple centerline estimated in the previous template estimation step

Iterate until the warping fct's become almost the identity!

4.3 Example: Berkeley Growth Study data

(Tuddenham and Snyder, 1954)

Sangalli Secchi Vantini Vitelli 2010 CSDA

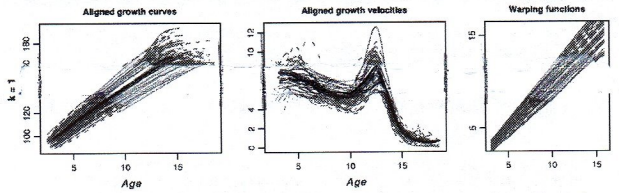


93 children, 39 boys and 54 girls

Curves estimated by monotonic cubic regression splines, implemented using the R package *fda*

4.3 Berkeley Growth Study data

Sangalli Secchi Vantini Vitelli 2010 CSDA



Results of continuous alignment using the following similarity index and class of warping functions (ρ, W):

$$\rho(c_i, c_j) = \frac{\int_{S_{ij}} c_i'(s) c_j'(s)\, ds}{\sqrt{\int_{S_{ij}} c_i'(s)^2 ds}\sqrt{\int_{S_{ij}} c_j'(s)^2 ds}}$$

the focus is on growth patterns, rather than on the absolute heights of the children or on their more or less pronounced growths

$$\rho(c_i, c_j) = 1 \Leftrightarrow \exists a \in \mathbb{R}, b \in \mathbb{R}^+ : c_i(t) = a + b c_j(t) \quad (*)$$

$c_i(t) \approx c_j(t)$

$$W = \{h : h(t) = mt + q \text{ with } m \in \mathbb{R}^+, q \in \mathbb{R}\}$$

constant modifications of the running speeds of the children biological clocks

this is the cosenus of the angle between the derivatives of the curves (between velocities)

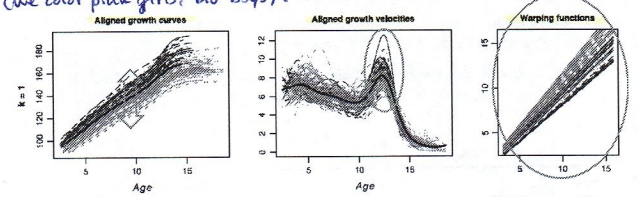
(*): if $\cos(c_i, c_j) = 1 \Rightarrow$ the angle is zero, so the two curves are very similar (one is the multiple of the other)

Now that we have aligned the functions we wonder what happens between boys and girls? We had 2 lables:

4.3 Berkeley Growth Study data

Sangalli Secchi Vantini Vitelli 2010 CSDA

(we color pink girls, blu boys):



Once the biological clocks are aligned

the height of boys stochastically dominates the one of girls for any registered biological age

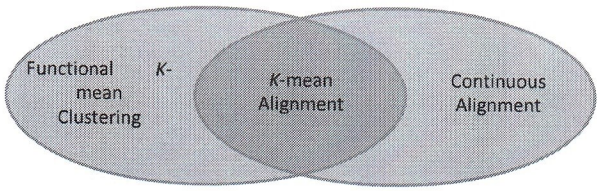
boys have a more pronounced growth during puberty (more prominent growth velocity peak)

Neat separation of boys and girls in the phase.

The biological clocks of boys and girls run at different speeds

4.4 Simultaneous registration and classification: the K-mean Alignment algorithm

Sangalli Secchi Vantini Vitelli 2010 CSDA



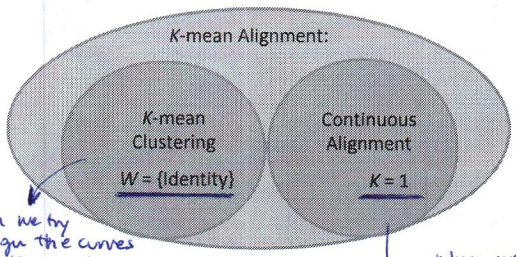
→ *K*-mean Clustering with warping allowed

→ Continuous Alignment with *K* templates

Code for *K*-mean alignment: R package fdakma, available from CRAN

4.4 K-mean Alignment

Sangalli Secchi Vantini Vitelli 2010 CSDA



When we try to align the curves to K different templates (not only 1) but we do NOT ALLOW WARPING

when we're trying to do K-mean but with only 1 group and we ALLOW WARPING

Code for *K*-mean alignment: R package fdakma, available from CRAN

if we allow:
- K different templates
- warping

⟹ K-mean alignment

4.4 K-mean Alignment

Sangalli Secchi Vantini Vitelli 2010 CSDA

Goal of Alignment:
Decoupling Phase and Amplitude Variability

+

Goal of *K*-mean Clustering:
Decoupling Within and Between-cluster (Amplitude) Variability

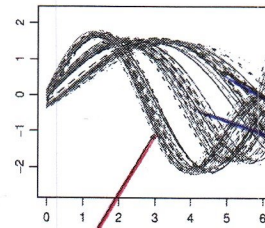
=

Goal of *K*-mean Alignment:
Identifying Phase Variability, Within-cluster Amplitude Variability and Between-cluster Amplitude Variability

(disclosing clustering in the phase)

4.4 A small part of a larger simulation study...

Sangalli Secchi Vantini Vitelli 2010 CSDA

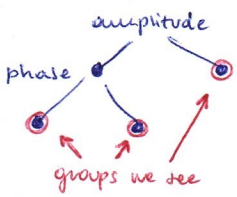


2 AMPLITUDE CLUSTERS (2 templates were) generated these data?

ONE has associated a further CLUSTERING IN THE PHASE

$1 * \sin(s) + 1 * \sin\left(\frac{s^2}{2\pi}\right)$

$2 * \sin(s) - 1 * \sin\left(\frac{s^2}{2\pi}\right) + (1+\varepsilon_{4i})s) + (1+\varepsilon_{2i}) * \sin\left(\frac{(\varepsilon_{3i}+(1+\varepsilon_{4i})s)^2}{2\pi}\right)$

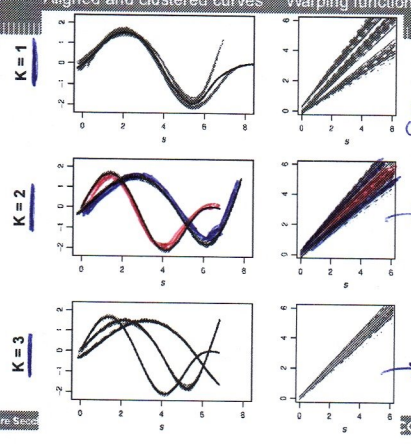


2 amplitude clusters (2 template curves) and there is a 3rd cluster obtained by modifying the phase and not the amplitude (Qualitatively is very difficult to imagine)

How many clusters there are? Qualitatively we can think that there are 3 clusters (3 "strings")
How did this simulated data been generated?
1 curve + gaussian noise for the central group
1 curve (different) + gaussian noise for the "left" group
The 3rd group (the "right" one) has been generated by warping the center group (it's not another template!)

Aligned and clustered curves | Warping functions

Let's apply K-means alignment:



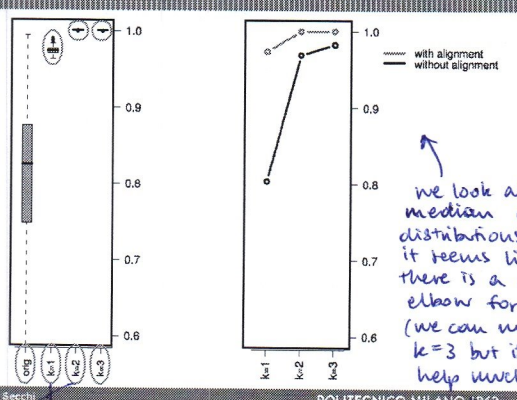
we try only the alignment, there is a clear cluster in the warping functions (it seems that there are more groups)

we see a grouping structure (those used to align the red are in the middle, those used to align the blue are above and below)

not very different from the identity (we cluster without alignement)

How can we decide? We look at the distribution of the dissimilarities:

4.4 A small part of a larger simulation study...



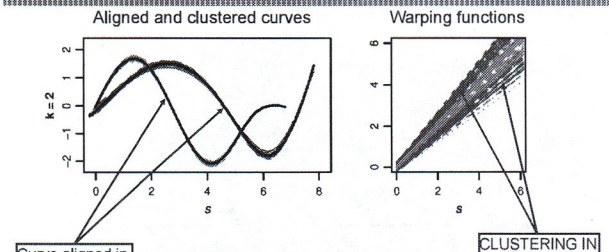
we look at the median of these distributions and it seems like there is a clear elbow for k = 2 (we can move to k = 3 but it doesn't help much)

k=1 the curves became more similar than the original

k=2 even more similar than with k=1

k = 2 with alignment is the best we can do

4.4 A small part of a larger simulation study…



Curve aligned in 2 AMPLITUDE CLUSTERS

CLUSTERING IN THE PHASE

k-mean alignment is able to efficiently detect true amplitude clusters and also to disclose clustering structures in the phase