Lecture 2: Unsupervised Learning
Functional PCA, Clustering and Registration

Valeria Vitelli

Oslo Centre for Biostatistics and Epidemiology (OCBE)
Department of Biostatistics, University of Oslo, Norway
valeria.vitelli@medisin.uio.no

UiO University of Oslo

PhD course
An Introduction to Functional Data Analysis:
Theory and Practice

University of Palermo, Italy; March 25th – 28th, 2024

Outline

1. Functional Principal Component Analysis (fPCA)
   - Defining fPCA
   - Visualization
2. Functional Clustering
   - Distance-based Clustering
   - Hierarchical Clustering
3. Registration of Functional Data
   - Motivation & First Definitions
   - Landmark Registration
   - Continuous Data Registration (& Clustering)

Main Reference: Chapters 7-8 in R&S[1], Section 4.1 in Wang et al. (2016)

[1] Ramsay & Silverman, 2005: Functional Data Analysis, 2nd edt, *Springer*

fPCA: Motivation

PCA is a statistical technique to

- find structure / patterns in data of high dimensions
- reduce dimensionality
- interpret modes of variation

Functional Principal Component Analysis (fPCA)

- first method to be considered in the early literature on FDA, extends PCA to the case when data are "curves"
- exploring data to characterize recurring "features" is even more difficult in functional sense
- variance-covariance and correlation functions can be difficult to interpret

fPCA: Link with Lecture 0

Intuition

fPCA allows dimension reduction of infinite-dimensional functional data to manageable finite vectors of functional PC scores

It is based on the **Karhunen-Loeve expansion** (Lecture 0, last slide), which allows expanding a random function $x \sim \mathcal{X}$, $\{x(t), t \in T\}$ as

$$x_i(t) = \mu(t) + \sum_{j=1}^{\infty} \xi_{ij}\phi_j(t), \quad t \in T \tag{1}$$

where $\mu(t)$ is the mean function of $\mathcal{X}$, ϕ_j's are the orthonormal eigenfunctions of the covariance, which satisfy

$$\int_T \phi_j(t)\phi_m(t) = 1 \Leftrightarrow \; j = m, \quad \text{0 otherwise}$$

fPCA: Link with Lecture 0

Recall: Mercer's Theorem

$$Cov(\mathcal{X}(s), \mathcal{X}(t)) = \sum_{j=1}^{\infty} \lambda_j \phi_j(s) \phi_j(t),$$

with decreasing eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq 0$

Moreover:

Properties of the ξ_{ij} in the Karhunen-Loeve expansion (1)

- $\xi_{ij} := \int_T (x_i(t) - \mu(t))\phi_j(t)dt$ are r.v. called x_i's **scores**
- the ξ_{ij}'s are uncorrelated, $\mathbb{E}[\xi_{ij}] = 0$ and $Var(\xi_{ij}) = \lambda_j \; \forall j$
- if $\mathcal{X}$ is a Gaussian random variable $\Rightarrow \xi_{ij} \sim \mathcal{N}(0, \lambda_j)$

fPCA: formal derivation

Idea behind fPCA: find projections of maximal variance (as PCA!)

Sample $\{x_i(t) : t \in T\}_i$ of i.i.d. zero mean curves in $L^2(T)$. Then:

1. **the first fPC** is the function $\phi_1(t)$ such that

$$\xi_{i1} = \langle \phi_1, x_i \rangle = \int_T \phi_1(t) x_i(t) dt$$

   has maximal variance, subject to the constraint $||\phi_1||^2 = 1$

2. **the second fPC** is the function $\phi_2(t)$ such that

$$\xi_{i2} = \langle \phi_2, x_i \rangle = \int_T \phi_2(t) x_i(t) dt$$

   has maximal variance, subject to $\langle \phi_2, \phi_1 \rangle = 0$ and $||\phi_2||^2 = 1$

3. ...

fPCA: formal derivation

Recall the Notation

Sample $\{x_i(t) : t \in T\}_i$ of i.i.d. zero mean curves in $L^2(T)$. Then:

- covariance function $K(s,t) := Cov(\mathcal{X}(s), \mathcal{X}(t)) = \mathbb{E}\left[x_i(s)x_i(t)\right]$
- $K(\cdot,\cdot)$ induces an integral operator $\mathcal{K}$ such that

$$\mathcal{K}f(t) = \int_T K(t,s)f(s)ds, \ \forall \ f(t) \in^2 (T)$$

One can then rewrite the fPCA derivation in the previous slide as

1. 1^{st} fPC: $\phi_1 = \text{argmax}_f \langle \mathcal{K}f, f\rangle$, with $||f||^2 = 1$
2. 2^{nd} fPC: $\phi_2 = \text{argmax}_f \langle \mathcal{K}f, f\rangle$, with $\langle \phi_1, f\rangle = 0$ and $||f||^2 = 1$
3. ...

fPCA: formal derivation

Conclusion

It is then obvious that the functional PCs $\{\phi_j\}'s$ are the *eigenvectors* of the *covariance operator* $\mathcal{K}$:

$$\mathcal{K}\phi_j = \lambda_j \phi_j \qquad \forall j \geq 1$$

Therefore, the following phrasings are **equivalent:**

- functional Principal Components (fPC)
- eigenvectors of the covariance operator $\mathcal{K}$
- eigenfunctions of the covariance function $K(\cdot, \cdot)$

fPCA: Practical Considerations

The usual evaluations of PCA apply also in the functional case!

- **Percentage of Explained Variance (PEV)** by the first K fPCs equal to

$$PEV = \frac{\sum_{j=1}^{K} \lambda_j}{\sum_{j=1}^{\infty} \lambda_j}$$

- Given sample mean $\bar{x}(t)$ and sample covariance function $Cov_{\mathcal{X}}(s,t)$ (see Lecture 1!) one can estimate the eigenvalues and eigenfunctions $\{\hat{\lambda}_j, \hat{\phi}_j(\cdot)\}$ by eigenanalysis ($=$ spectral decomposition) of $Cov_{\mathcal{X}}(s,t)$

,

fPCA: Practical Considerations

- Useful to **select a truncation** K by using the PEV (other information criteria such as AIC, BIC, .. can be used)
- Estimate the **fPC scores** as:

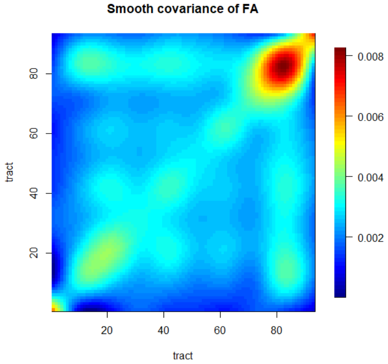
$$\hat{\xi}_{ij} = \int_T (x_i(t) - \bar{x}(t))\hat{\phi}_j(t)dt$$

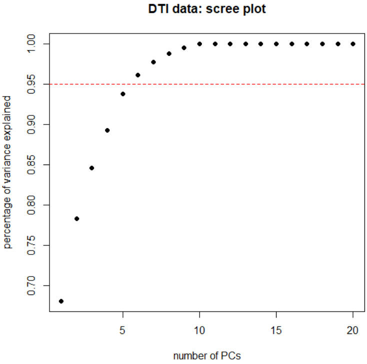
- Estimate the **K-L expansion** as:

$$\hat{x}_i(t) = \bar{x}(t) + \sum_{j=1}^{K} \hat{\xi}_{ij}\hat{\phi}_j(t)$$

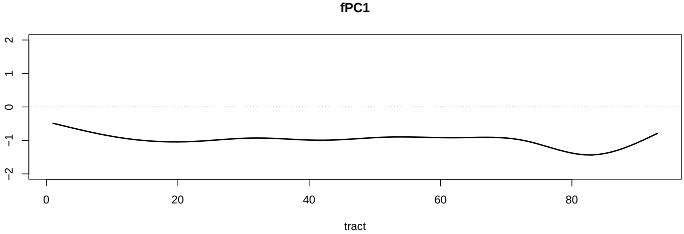
- **Useful additional plots:** $\bar{x}(t) \pm 2\sqrt{\hat{\xi}_{ij}}\hat{\phi}_j(t)$ for some j ("effect" of the single fPC on the mean!)

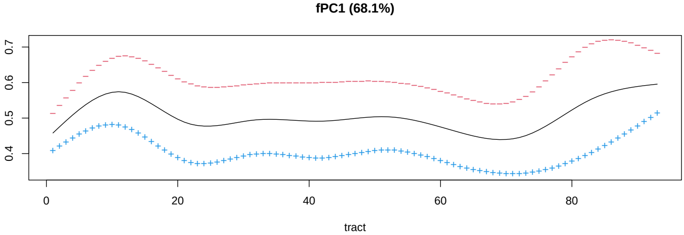
Some useful plots – DTI data example



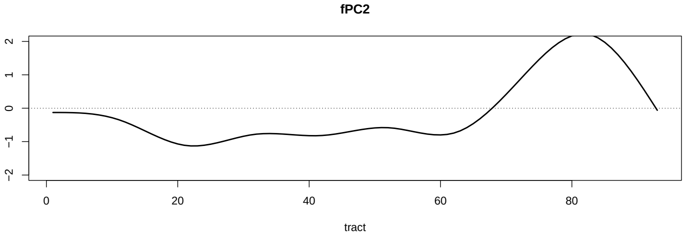


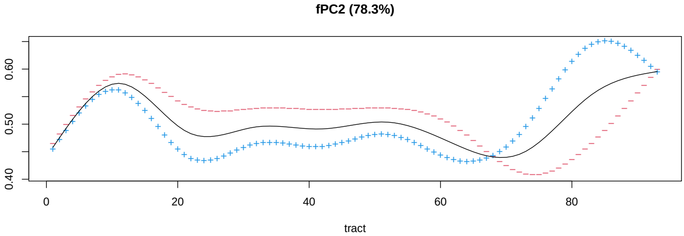
Some useful plots – DTI data example



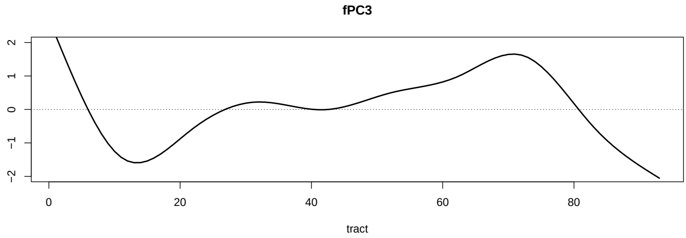


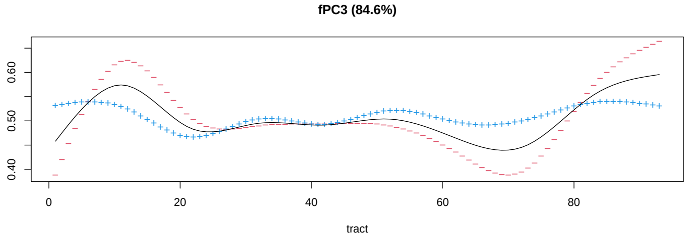
Some useful plots – DTI data example





Some useful plots – DTI data example





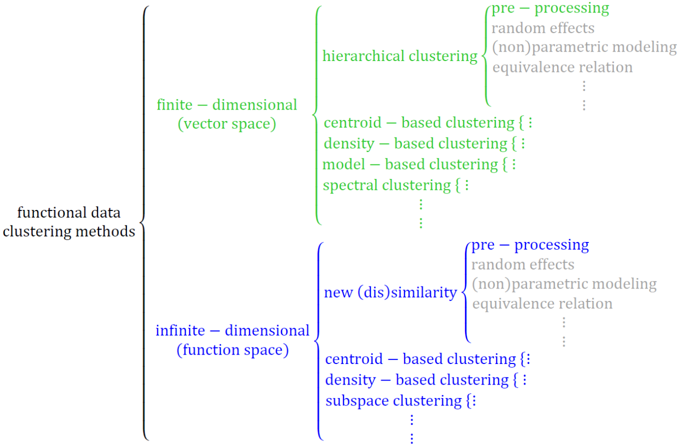
Functional Clustering: Introduction

- Clustering is a useful tool for traditional multivariate data analysis and is equally important yet more challenging in a functional data context
- **Functional data clustering** is an unsupervised learning process that aims at grouping a set of data such that data objects within clusters are more similar than across clusters with respect to a metric
- Classical clustering concepts for multivariate data can typically be extended to functional data, where additional changes arise (e.g., discrete approximations of distance measures, dimension reduction of the infinite-dimensional functional data objects, . . .)

Functional Clustering: Recap from Multivariate Statistics

- Most classical and popular approaches: **hierarchical clustering** and **partitioning approaches** (e.g., k-means)
- **Hierarchical clustering:** either agglomerative (bottom-up) or divisive (top-down) algorithm, requires to define a *functional dissimilarity measure* between sets of observations + *linkage* (rule to decide which clusters to combine or split).
  Number of groups inferred.
- **Partitioning approaches:** usually alternate two steps, calculation of *cluster centers* and assignments of observations to groups via a *metric* measuring distance to centers.
  Number of groups fixed. Optimize a criterion (often within-cluster sum of squared distances from center).

Categorization of Functional Clustering methods



functional data clustering methods

finite − dimensional (vector space)

hierarchical clustering

pre − processing

random effects

(non)parametric modeling

equivalence relation

⋮

centroid − based clustering { ⋮

density − based clustering { ⋮

model − based clustering { ⋮

spectral clustering { ⋮

⋮

infinite − dimensional (function space)

new (dis)similarity

pre − processing

random effects

(non)parametric modeling

equivalence relation

⋮

centroid − based clustering { ⋮

density − based clustering { ⋮

subspace clustering { ⋮

⋮

Figure from Zhang and Parnell (2023)

Finite-dimensional methods

- given a set of basis functions $\{\phi_1, \phi_2, \ldots\}$ of the functional space, the first K projections $\{\xi_{ik}\}$ of the observed trajectories onto the space spanned by the set of basis functions can be used to represent the functional data, where $\xi_{ik} = \langle x_i, \phi_k \rangle$
- **it is then assumed** that the coefficients $\{\xi_{ik}\}$ are a good finite-dimensional approximation of **the clustering structure in the functional space**
- **they typically are not** (unless clustering structure is trivial)

Typical **finite-dimensional functional clustering** approach:

1. represent the functional data by the set of coefficients in the basis expansion
2. apply multivariate clustering algorithms to the coefficients

I will thus skip such methods (interesting as pre-smoothing)

A more interesting finite-dimensional method: Illustration

fPCA + clusering scores

Intuition: estimate a good basis via fPCA + cluster fPCA scores.
Problem? fPCA captures modes of variation; not necessarily groupings! (clustering structure might be masked)

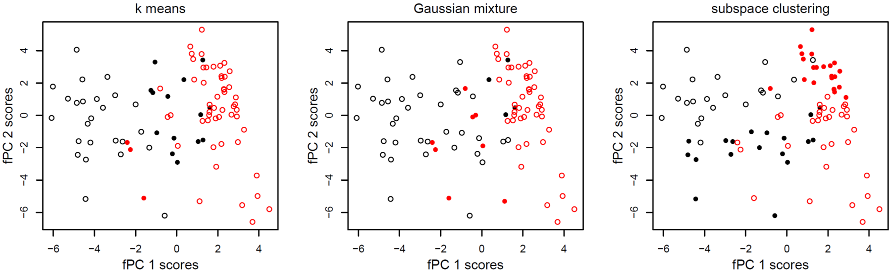


Figure from Zhang and Parnell (2023)

Functional k-means

Starting point: functional observations $x_1(t), \ldots, x_n(t)$ (obtained after smoothing). We assume $x_i(t) \in L^2(T)$ (can be relaxed, to fix ideas). Remember usual internal product & norm.

Functional k-means: intuition

Assuming the number of clusters is L, find a set of cluster centers $\{\mu^c; c = 1, \ldots, L\}$ by minimizing the sum of the squared distances $d(x_i, \mu^{C_i})$ between $x_i(t)$ and the cluster center corresponding to their cluster assignment $C_i \in \{1, \ldots, L\}, i = 1, \ldots, n$.

Functional k-means

Functional k-means: algorithm

- fix L and a *suitable* functional metric $d(\cdot,\cdot)$ (ex. L^2 norm)
- initialize $\{C_1,\dots,C_n\}$ randomly; iterate the two steps:
  1. compute cluster centers as
  $$\mu^c = \sum_{i=1}^{n} x_i(t) 1(C_i = c)/n_c$$
  with $n_c = \sum_{i=1}^{n} 1(C_i = c)$
  2. assign each observation $x_i(t)$ to a cluster such that
  $$C_i = \underset{c=1,\dots,L}{\operatorname{argmin}}\, d(x_i, \mu^c)$$

Choice of the Functional metric

- most obvious & popular choice: **metric induced by the L^2 norm**

$$d_{L^2}(f,g) = \int_T (f(t) - g(t))^2 dt \qquad (2)$$

- **derivative-based functional cosine similarity,** also called "Pearson correlation" (Sangalli et al. 2010)

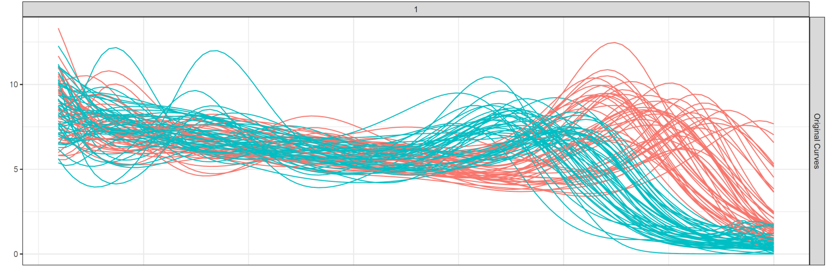
$$\rho(f,g) = \frac{\int_T (f'(t) \cdot g'(t)) dt}{\sqrt{\int_T (f'(t))^2 dt \int_T (g'(t))^2 dt}} \qquad (3)$$

  very good choice in relation to registration (see next Section), but *similarity* NOT metric ⇒ reverse optimization!

- Other popular choices: **fPCA-induced** or **derivative-induced** semi-metrics (see Ferraty and Vieu (2006), Chapter 3)

Functional k-means of the Berkeley Growth Data

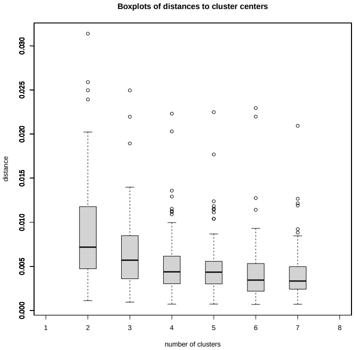
Data: growth velocities smoothed via monotone splines
Algorithmic choices: functional k-means via R package `fdakmeans`, L^2 metric, cluster centers: means, $L = 2$ fixed

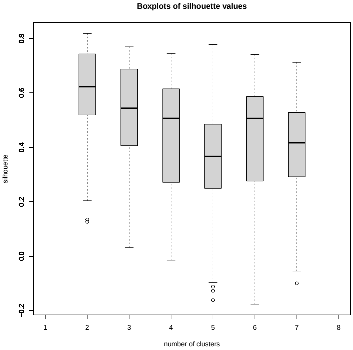


Functional *k*-means of the Berkeley Growth Data

Data: growth velocities smoothed via monotone splines

Algorithmic choices: functional *k*-means via R package `fdakmeans`, L^2 metric, cluster centers: means, $L = 2$ fixed





Functional Hierarchical Clustering

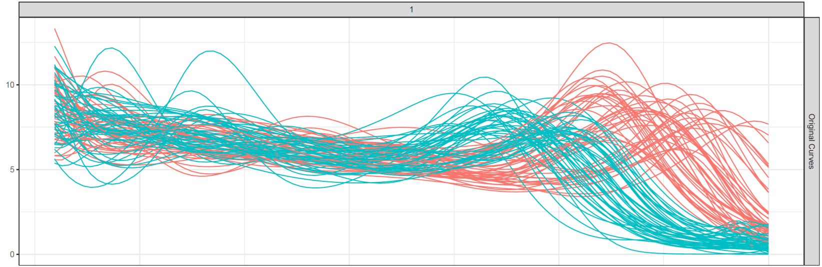
Starting point: functional observations $x_1(t), \ldots, x_n(t)$ (obtained after smoothing). We assume $x_i(t) \in L^2(T)$ (can be relaxed, to fix ideas). Remember usual internal product & norm.

Sketch of the algorithm

- Focus on *Bottom-up algorithm* (top-down methods are less common)
- Start with each observation assigned to its own cluster
- In each iteration, merge the two clusters with the minimal distance from each other – until left with a single cluster containing all observations
- Functional distance / metric same as before
- *Linkage:* same criteria as in multivariate case (common choices: complete, average, single)

Functional hclust of the Berkeley Growth Data

Data: growth velocities smoothed via monotone splines
Algorithmic choices: functional hier clustering via R package `fdakmeans`, L^2 metric, cluster centers: means, complete linkage



A known issue in FDA

A two-fold source of variability in functional data

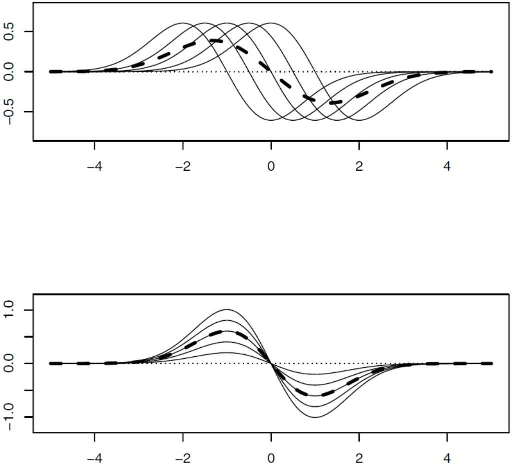
When doing functional data analysis, often displaying functions by plotting $x_i(t)$ versus $t \in T$ shows a visible "two-fold source of variation": one variation *along the y-axis* and one *along the x-axis*, respectively named **amplitude** and **phase** variation

Misalignment

The term **phase variability** is typically used to describe some abscissa variation ancillary to the scopes of the analysis (e.g., biological clock, measurement start, length of the task,...)
Not dealing with phase variability, or not accounting for the fact that it can occur, can severely affect the analysis results

Illustration of the two sources of variability

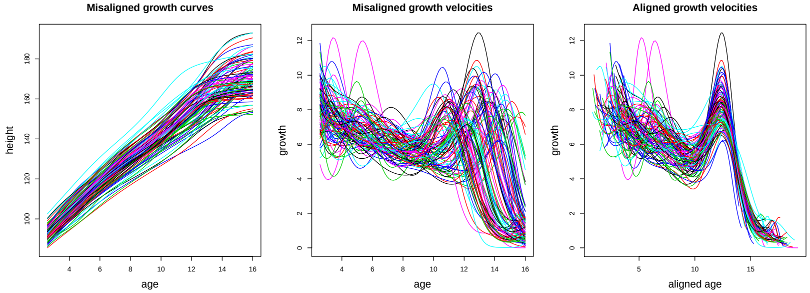
Figure 8.2 from Parmigiani (2009):
Top panel, five curves varying only in phase; bottom panel, five curves varying only in amplitude. ,The dashed line in each panel indicates the functional mean.



Misalignment matters: A known example

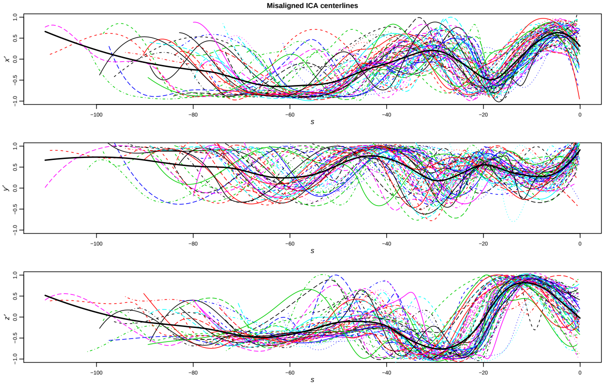
Berkeley Growth Study Data

Evident misalignment in the data due to the kids' biological clocks! (something we have ignored so far)



Misalignment matters: A more complex example

Issue: classical data analyses used in the functional data context can be tremendously affected by the presence of phase variation, since they were designed for simpler data structures without phase changes.



AneuRisk65 data: 3 spatial coordinates (mm) of the Internal Carotid Artery (ICA) centerlines of 65 subjects *suspected* of cerebral aneurysm. `https://statistics.mox.polimi.it/aneurisk/`

A first general definition

Definition: Registration of Functional Data

In most general terms, *registering a functional observation* $x_i(t) \in T$ means finding a *non-linear transformation of the abscissa* $h_i(t) : [0,1] \to T$ (**warping function**) such that

$$\tilde{x}_i(t) := x_i(h_i(t)), t \in [0,1]$$

and such that the set of *registered* (or *aligned*) curves

$$\{\tilde{x}_1(t), \ldots, \tilde{x}_n(t)\}$$

is now such that the ancillary variability along t is removed (i.e., variability in the values of $\tilde{x}_i(t)$ for any fixed $t \in [0,1]$ are only related to *structural amplitude variability*, which is usually related to the process under observation)

Properties of a warping function $h(t)$

Time-warping functions (or simply, **warping functions**) must be

- strictly *monotone increasing* (time can't go backwards)
- *smooth*, in the sense of being differentiable up to at least what applies to the curves being registered
- satisfy the equation: $h^{-1}\left[h(t)\right] = t$

Domain Definition

- When the original curves are observed over a common interval $[0, T]$, the warping functions often satisfy the constraints $h(0) = 0$ and $h(T) = T$
- This does not need to be the case! Each $h(\cdot)$ might also
  - be defined on varying intervals $[0, T_i]$ each transformed to a common interval $[0, T]$, or
  - be defined on a common reference interval $[0, 1]$ and each transformed to a curve-specific interval $[0, T_i]$

A first general definition

Phase Variability

Phase Variability is then *defined* by the estimation procedure that we decide for the set $\{h_1(t), \ldots, h_n(t)\}$ (for example, which functional space we use, how do we define the *optimization problem associated to registration*)

General question: how to estimate the set of non-linear transformations of the abscissa $\{h_1(t), \ldots, h_n(t)\}$?

When the focus is not on *estimating* phase variability but rather on *removing it*, then **landmark registration** is a popular strategy.

Landmark Registration

Definition: Landmark

A *landmark* of a curve is some characteristic that one can associate with a specific argument value t (typically maxima, minima, zero crossings); can be identified on the curves, or on some derivatives.

Given F landmarks, each landmark $f = 1, \dots, F$ is fully identified by a vector $\{t_{0f}; (t_{1f}, \dots, t_{nf})\}$, where

- t_{if} is the occurrence of the landmark on the curve $x_i(\cdot)$, $i = 1, \dots, n$
- t_{0f} is a "reference" for the landmark occurrence in the *registered abscissa system* (often the mean of $\{t_{1f}, \dots, t_{nf}\}$)

Landmark Registration

Landmark Registration Process

Therefore, for each curve $x_i(t)$, one needs to

1. identify the argument values t_{if}, $f = 1, \ldots, F$ associated with each of the F landmarks
2. construct a transformation $h_i(\cdot)$ such that the registered curves with values $\tilde{x}_i(t) = x_i(h_i(t))$ satisfy

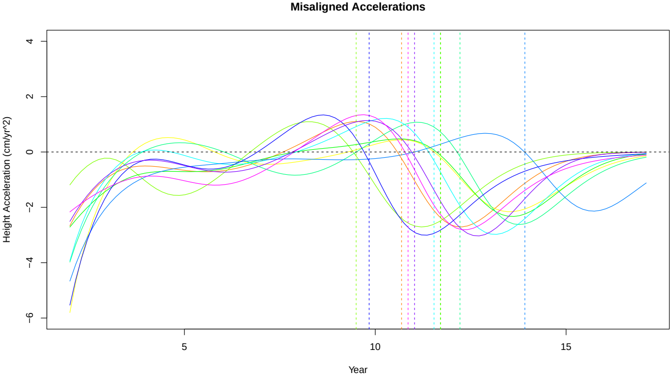
$$x_i(t_{if}) = \tilde{x}_i(t_{0f}),\ \forall f = 1, \ldots, F$$

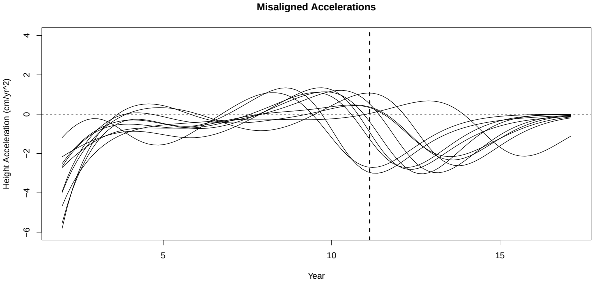
   i.e., each landmark occurs in the same abscissa value for all registered curves

Landmark Registration: Illustration on Growth Data

Data: growth accelerations of the first 10 girls

Algorithmic choices: landmark registration with R package `fda`, 1 single landmark at the pubertal spurt




Misaligned Accelerations
Height Acceleration (cm/yr^2)
4
2
0
−2
−4
−6
5
10
15
Year

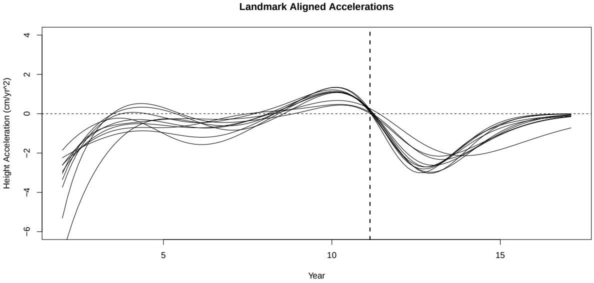
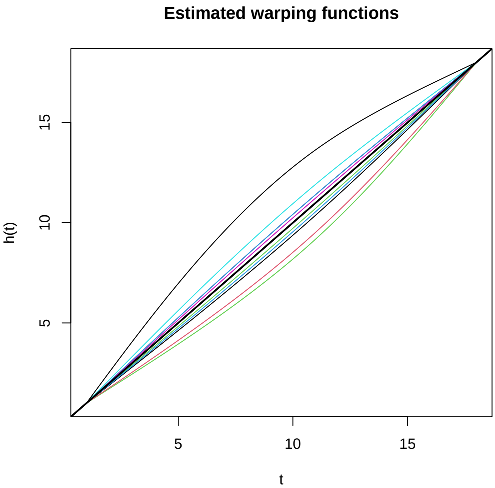

Landmark Aligned Accelerations
Height Acceleration (cm/yr^2)
4
2
0
−2
−4
−6
5
10
15
Year

Illustration on Growth Data: Warping Functions



Continuous Data Registration

Some disadvantages of landmark registration:

- not estimating the argument values t_{if}, $f = 1, \ldots, F$ for each landmark and the warping functions $h_i(\cdot)$ *jointly*
- when wanting to compute *higher order derivatives* of the curves with respect to warped time, the warping function must also be differentiable to the same order, and landmark interpolation would not carry us beyond the first derivative

Continuous Registration methods: do not use landmarks, but rather target the warping function $h_i(t)$ directly in a suitable functional space; for example, minimizing the L^2 distance between x_i aligned via h_i and a second function x_j

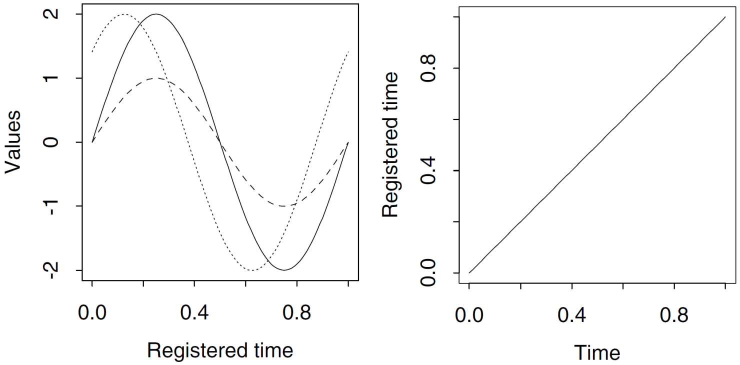
$$h_i = \underset{h \in W}{\operatorname{argmin}} ||x_i \circ h - x_j||_2^2 = \underset{h \in W}{\operatorname{argmin}} \int_T (x_i(h(t)) - x_j(t))^2 dt \quad (4)$$

Continuous Data Registration: Some caveats

The continuous registration optimization problem as stated in (4) carries some issues:

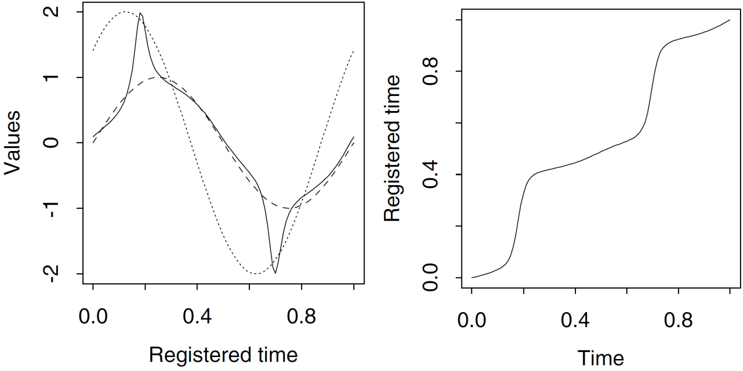
- it is **not symmetric** (registering x_i to x_j is not the same as registering x_j to x_i)
- **pinching problem:** the distance (4) can become small or zero even if x_i and x_j are not warped versions of each other, by using a warping function h_i that compresses areas of the domain where x_i and x_j are dissimilar and expands those where they are close, resulting in *spiky warping functions* (see Marron et al. (2015) for an excellent discussion on the topic)

Pinching problem: Illustration



An artificial registration problem (Figure 7.10 in R&S): left panel, the dotted curve is the misaligned curve to be aligned to the dashed one; result should give the solid curve, using the warping function $h(t) = t$ (right panel)

Pinching problem: Illustration



An artificial registration problem (Figure 7.10 in R&S): left panel, results of aligning the dotted curve (solid curve is the result) by mere application of a criterion similar to (4); right, the obtained warping

Possible Solutions

Solution 1: Keep it simple

“This pinching effect can be mitigated by using warping functions that are constrained to be smooth, either by the use of a regularization strategy or by the use of a small number of basis functions” (Marron et al. 2015)
Penalize so not to go very far from the identity

Solution 2: A rigorous mathematical framework

Even when using simple parametric families, it is crucial to appropriately relate the definitions of amplitude variation and of phase variation, that are jointly described by the loss function to be optimized and the class of warping functions

Continuous Data Registration: A general framework

Decoupling amplitude and phase variation via **equivalence classes**

Srivastava and Klassen (2016) provides a nice introduction: phase variation is incorporated *within equivalence classes*, while amplitude variation appears *across equivalence classes*

We thus need to define:

- a metric $d(\cdot,\cdot): \mathcal{F} \times \mathcal{F} \to \mathbb{R}$ that measures the distance between two curves;
- a class W of warping functions $h: f \circ h \in \mathcal{F}, \forall\ f \in \mathcal{F}$ and $h \in W$, indicating the allowed transformations for the abscissa.

Core idea: the metric must be able to compare equivalence classes, and not just individual functions. This is achieved by requiring

$$d(f_1, f_2) = d(f_1 \circ h, f_2 \circ h), \quad \forall h \in W; f_1, f_2 \in \mathcal{F}, \tag{5}$$

i.e., the metric is *invariant* to variation *within* the equivalence class.

Definition: Continuous Registration

Aligning f_1 to f_2, according to (d, W), means finding $h^* \in W$ that minimizes $d(f_1 \circ h, f_2)$ with respect to h.

- *phase variability* is captured by the optimal warping h^*
- *amplitude variability* is the remaining variability between $f_1 \circ h^*$ and f_2

See Vantini (2012) for a very nice description of this framework

Choices for the metric d and class W coherent to (5)

Warping function class

Strictly increasing affine transformations

$$W = \{h : h(s) = ms + q \ \text{ with } \ m \in \mathbb{R}^+, q \in \mathbb{R}\} \tag{6}$$

Metrics $d(\cdot, \cdot)$ coherent to the class W in (6)

- the normalized L^2 norm (D_1 and D_2 are the domains of f_1 and f_2, respectively)

$$d_{L^2}(f_1, f_2) = \frac{\left(\int_{D_1 \cap D_2} (f_1(x) - f_2(x))^2 dx\right)^{1/2}}{\sqrt{\mu(D_1 \cap D_2)}} \tag{7}$$

- the derivative-based Pearson correlation defined in (3), provided we invert all optimization problems

Continuous Registration via Procrustes procedures

Initialize the warping functions $h_1, \dots, h_n$ to be all equal to the identity; then, iterate until convergence:

1. **Template identification step.** Holding the alignment functions $h_1, \dots, h_n$ fixed, the **aligning template** φ is computed by minimizing, with respect to $\varphi \in \mathcal{F}$, $\sum_{i=1}^{n} d(\varphi, x_i \circ h_i)$
2. **Alignment step.** Holding φ fixed, for $i = 1, \dots, n$ the ith curve x_i is aligned to the template φ to find h_i, i.e., by solving the following optimization

$$h_i = \underset{h \in W}{\text{argmin}}\, d(\varphi, x_i \circ h)$$

Joint registration and clustering of functional data

Motivation

- Scope of registration is to capture *phase variability*
- Scope of clustering is to capture *amplitude variability*
- the two sources of variation are intrinsically connected

Therefore, many authors claim that jointly decoupling the two sources of variation is often convenient (Gaffney and Smyth 2005; Mattar et al. 2012; Zhang and Telesca 2014)

The *Procrustes iterations* algorithm for performing continuous registration is **very well suited to clustering functional data:** we only need to estimate L templates instead of 1, and a **joint clustering and continuous registration** strategy is immediately defined (based for example on k-means)

Joint registration & clustering via Procrustes procedures I

Fix a number of clusters L (this is needed also for hierarchical clustering!). Initialize the cluster assignments $C_1, \ldots, C_L$ randomly and the warping functions $h_1, \ldots, h_n$ to be all equal to the identity; then, iterate until convergence:

1. **Template identification step.** Holding $C_1, \ldots, C_L$ and the alignment functions $h_1, \ldots, h_n$ fixed, the **aligning templates** of the L clusters $\{\varphi_1, \ldots, \varphi_L\}$ are computed by minimizing, with respect to $\varphi \in \mathcal{F}$, $\sum_{i:x_i \in C_l} d(\varphi_l, x_i \circ h_i)$ for all $l = 1, \ldots L$

Joint registration & clustering via Procrustes procedures II

2. **Assignment and alignment step.** Holding $\varphi_1, \dots, \varphi_L$ fixed, for $i = 1, \dots, n$ the ith curve x_i is aligned to all templates $\varphi_1, \dots, \varphi_L$, thus finding the optimal $h_i^1, \dots, h_i^L$, by solving the following optimization

$$h_i^l = \operatorname*{argmin}_{h \in W} d(\varphi_l, x_i \circ h), \quad \forall l = 1, \dots, L$$

Then, x_i is assigned to C_l such that

$$l = \operatorname*{argmin}_{j=1,\dots,L} d(\varphi_j, x_i \circ h_i^j),$$

and h_i^l is the new warping function h_i

Illustration on Growth Data: Continuous Registration

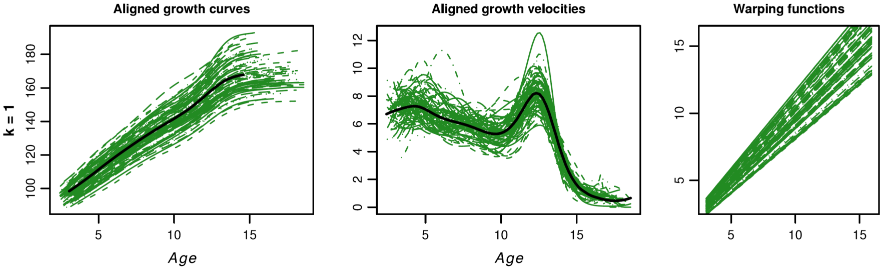


Figure 9 in Sangalli et al. (2010): registered growth curves (left) and corresponding growth velocities (center, estimated template in black), together with warping functions (right). Affine class of warping functions together with functional Pearson similarity were used.

Illustration on Growth Data: Continuous Registration & Clustering

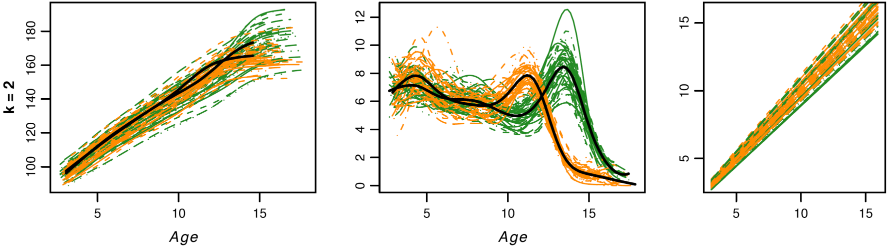


Figure 9 in Sangalli et al. (2010): same as previous slides, but now two clusters.

Illustration on Growth Data: Continuous Registration & Clustering - with gender comparison!

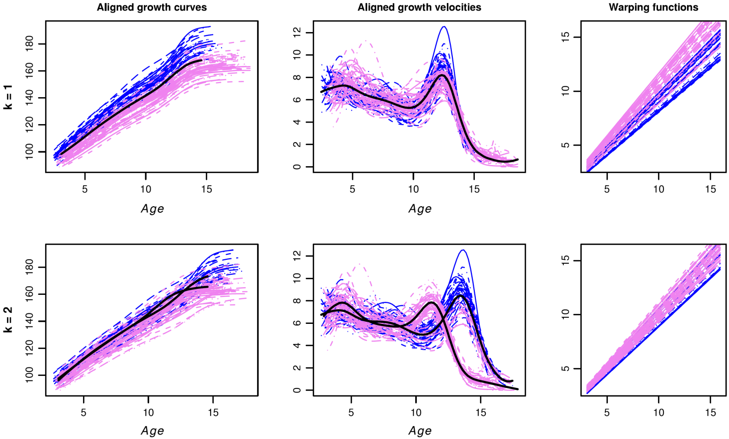


Figure 10 in Sangalli et al. (2010): again continuous registration with respect to two groups, and coloring according to gender.

Illustration on Growth Data: Continuous Registration & Clustering - number of groups!

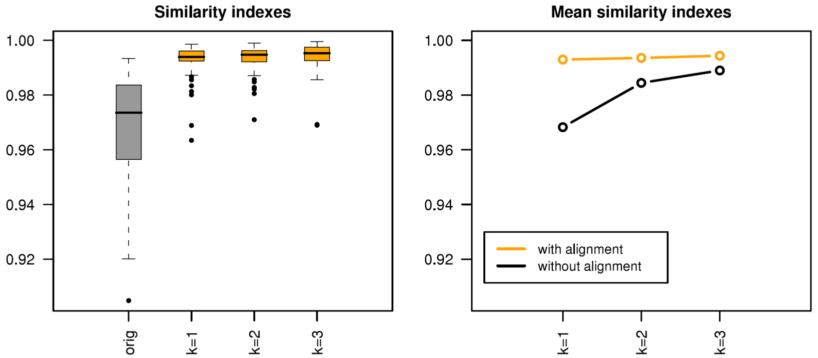


Figure 11 in Sangalli et al. (2010): left, boxplots of similarity indexes for original growth curves and registered growth curves with 1 to 3 clusters. Right, means of similarity indexes comparing with/without registration.

Summary

- fPCA:
  - generalization of PCA to functional data
  - practicalities (trucation and estimation)
  - visualization of results
- Clustering of functional data:
  - distance-based and hierarchical strategies
- Registration of functional data:
  - Why it matters
  - Landmark registration
  - Continuous registration
- Joint registration and clustering of functional data
  - Procrustes iterations approaches

References

F. Ferraty and P. Vieu. *Nonparametric Functional Data Analysis. Theory and Practice*. Springer, 2006.

Scott J Gaffney and Padhraic Smyth. Joint probabilistic curve clustering and alignment. In *Advances in neural information processing systems*, pages 473–480, 2005.

James Stephen Marron, James O Ramsay, Laura M Sangalli, and Anuj Srivastava. Functional data analysis of amplitude and phase variation. *Statistical Science*, pages 468–484, 2015.

Marwan A Mattar, Allen R Hanson, and Erik G Learned-Miller. Unsupervised joint alignment and clustering using bayesian nonparametrics. *arXiv preprint arXiv:1210.4892*, 2012.

Robert Gentleman Kurt Hornik Giovanni Parmigiani. Use r! 2009.

L. M. Sangalli, P. Secchi, S. Vantini, and V. Vitelli. K-mean alignment for curve clustering. *Computational Statistics and Data Analysis*, 54:1219–1233, 2010.

Anuj Srivastava and Eric P Klassen. *Functional and shape data analysis*. Springer, 2016.

Simone Vantini. On the definition of phase and amplitude variability in functional data analysis. *Test*, 21(4): 676–696, 2012.

Jane-Ling Wang, Jeng-Min Chiou, and Hans-Georg Müller. Functional data analysis. *Annual Review of Statistics and its application*, 3:257–295, 2016.

Mimi Zhang and Andrew Parnell. Review of clustering methods for functional data. *ACM Transactions on Knowledge Discovery from Data*, 17(7):1–34, 2023.

Yafeng Zhang and Donatello Telesca. Joint clustering and registration of functional data. *arXiv preprint arXiv:1403.7134*, 2014.