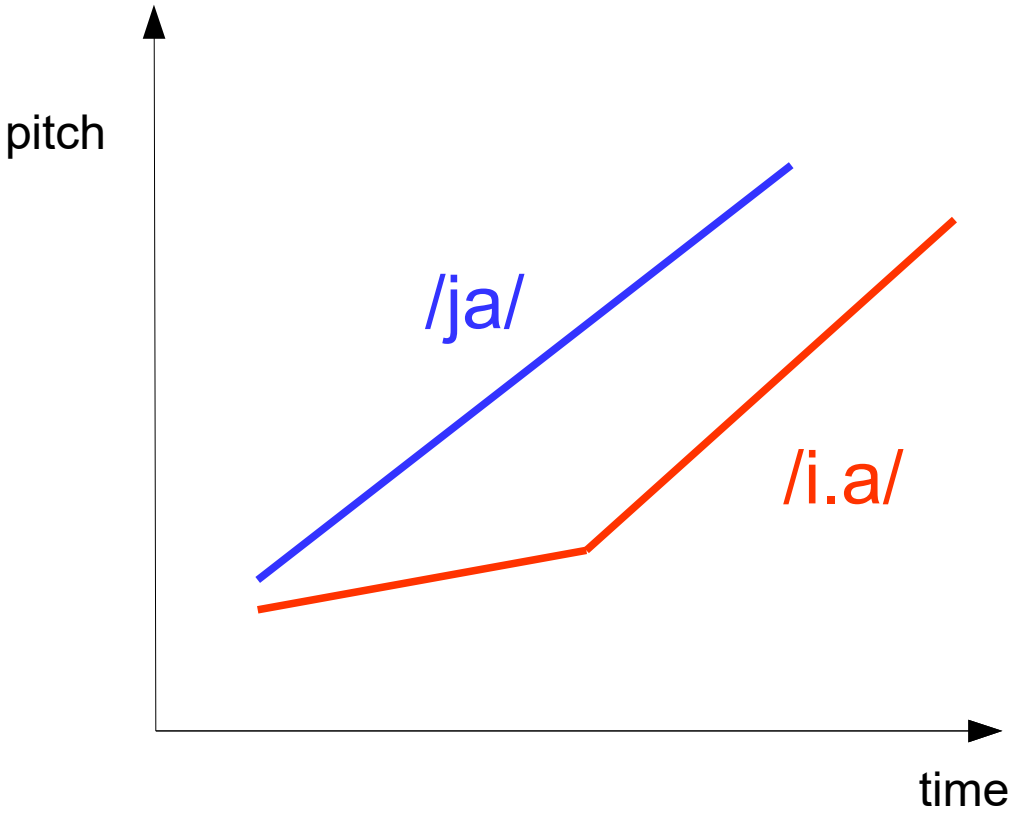
Modelling time-varying features of speech: tools and methods

Michele Gubian

Institute of Phonetics and Speech Processing
LMU Munich, Germany

Graz, October 2020

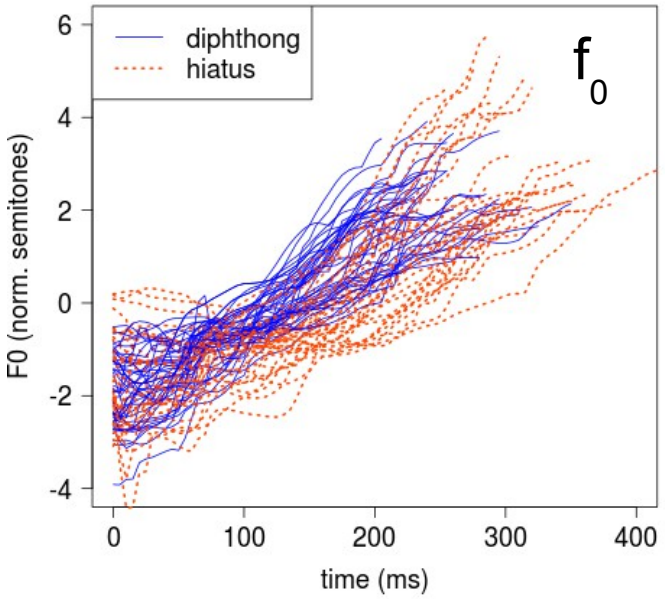
Alignment of rising pitch accents in Spanish



- European Spanish
- **Diphthong**: /ja/

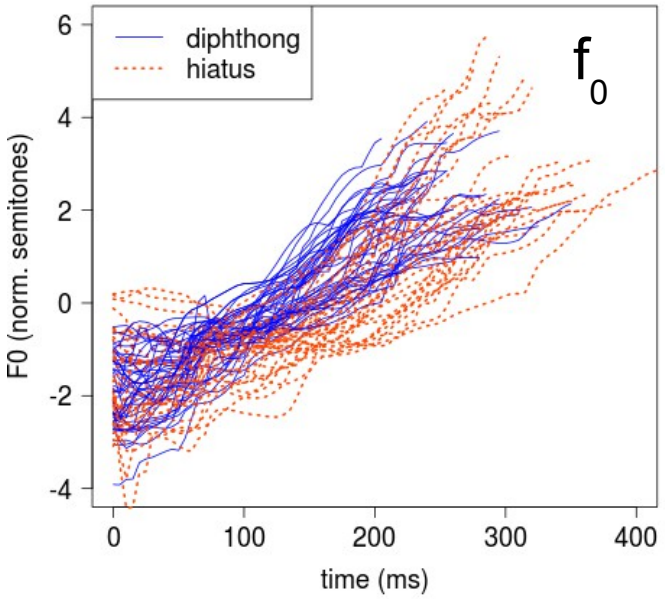
  e.g. *Emili<u>a</u>na*
- **Hiatus** /i.a/

  e.g. *pi<u>a</u>no*
- Rising pitch accent should align to syllabic structure



- Read speech
- 9 participants
- 20 Diphthongs +

  20 Hiatuses each

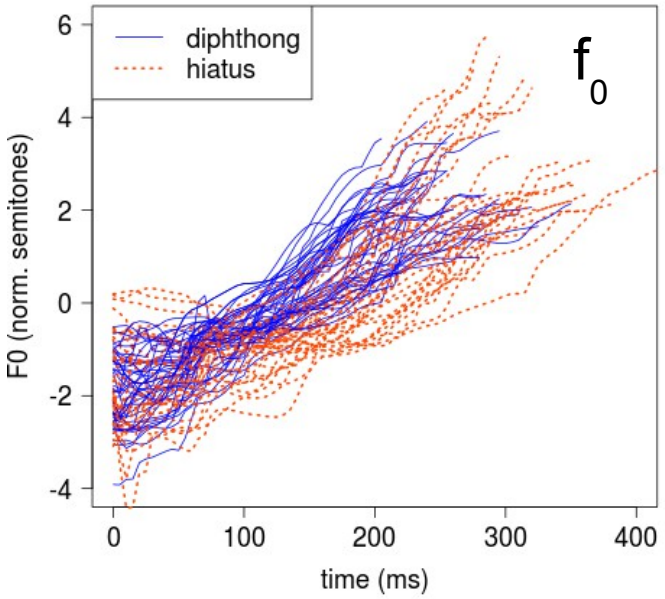


LMER

- Read speech
- 9 participants
- 20 Diphthongs +

  20 Hiatuses each

CURVES



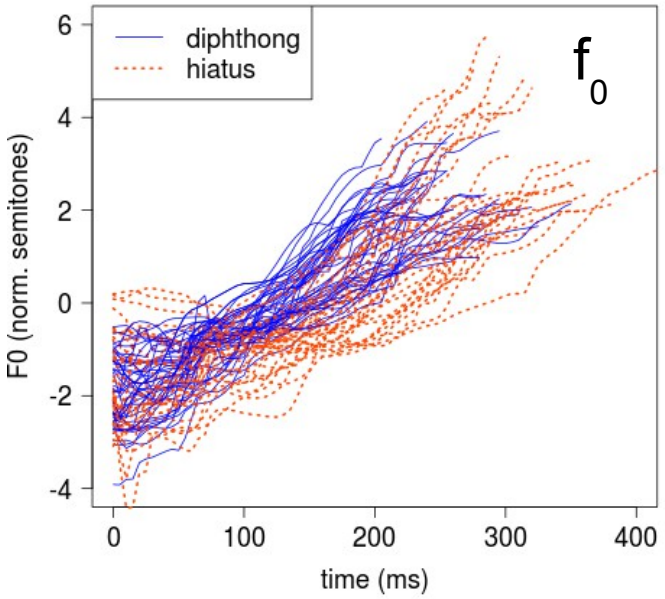
- Read speech
- 9 participants
- 20 Diphthongs +

  20 Hiatuses each

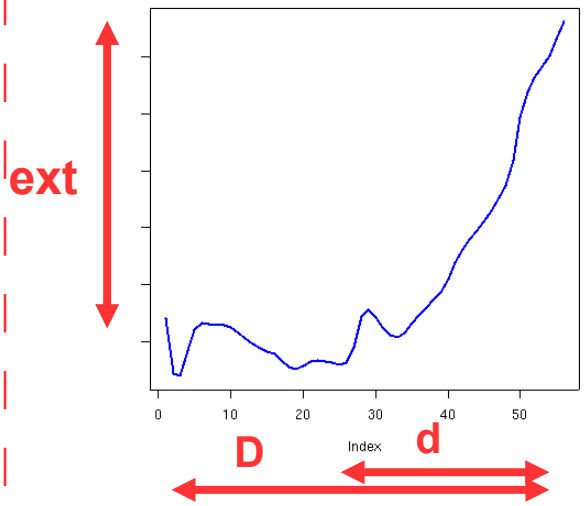
NUMBERS

LMER

CURVES



- Read speech
- 9 participants
- 20 Diphthongs +
  20 Hiatuses each

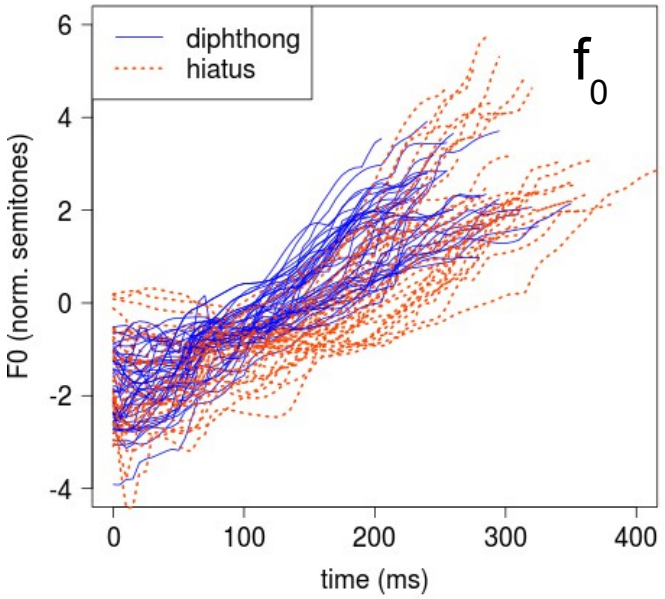


ext (st)	d/D	Cat.
5.3	0.9	D
4.6	0.7	H
...	...	...

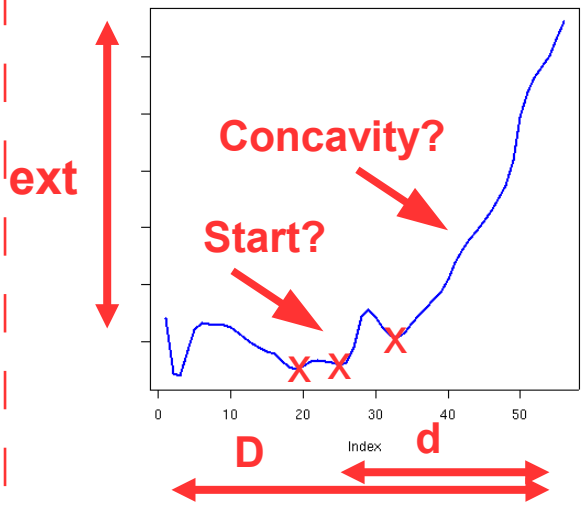
NUMBERS

LMER

CURVES



- Read speech
- 9 participants
- 20 Diphthongs +
  20 Hiatuses each

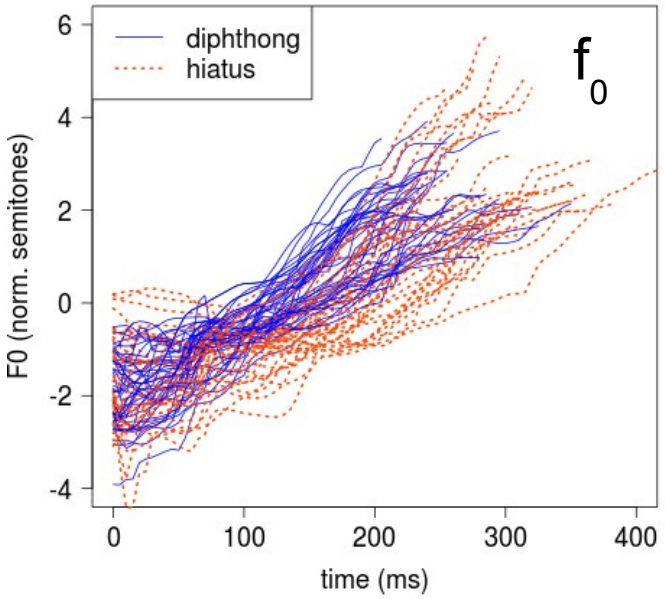


ext (st)	d/D	Cat.
5.3	0.9	D
4.6	0.7	H
…	…	…

NUMBERS

LMER

CURVES



- Read speech
- 9 participants
- 20 Diphthongs +

  20 Hiatuses each

DCT

k0	k1	k2	...
...	...	...	...
...	...	...	...
...	...	...	...

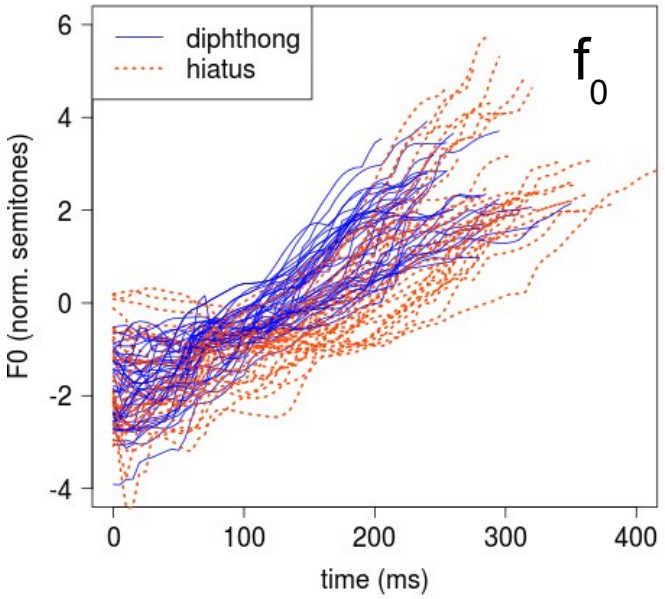
NUMBERS

LMER

DCT limitations

- DCT does not (easily) encode time-localised information, e.g. a small hump
- Typically only k0, k1 and k2 are used, which have a geometric interpretation (mean, slope, curvature)
- Extracting several k's brings up the need of PCA
- In general, not effective to encode long signals

CURVES



- Read speech
- 9 participants
- 20 Diphthongs +

  20 Hiatuses each

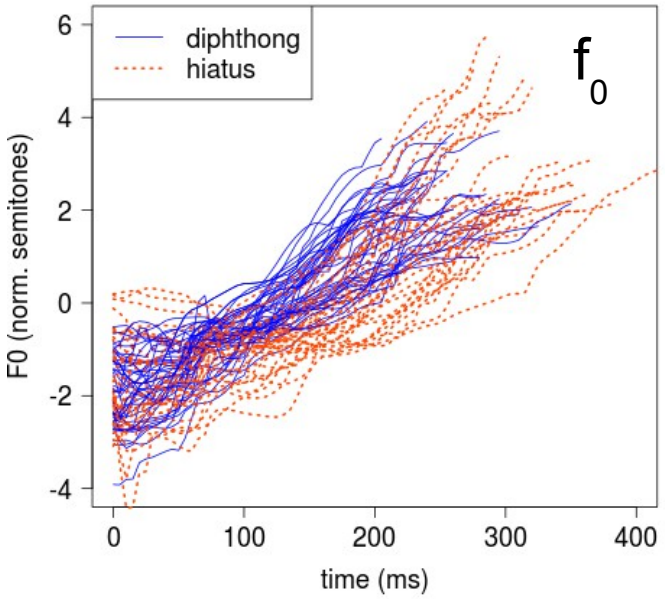
NUMBERS

GAMM

GAMMs

- **PRO**
  - LMER directly on curves
  - Good R packages (e.g. mgcv)
  - Good tutorials (e.g. Wieling, Soskuthy)
- **CON**
  - No easy way to analyse multidimensional signals
  - Computationally heavy
  - LMER directly on curves :D

CURVES



- Read speech
- 9 participants
- 20 Diphthongs +
  20 Hiatuses each

FPCA

s1	s2	s3	...
...	...	...	...
...	...	...	...
...	...	...	...

NUMBERS

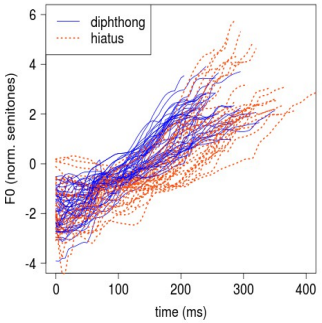
LMER

FPCA

- **PRO**
  - Computationally light
  - Interpretable
  - Easy to analyse multidimensional signals
  - Allows you to use LMER
- **CON**
  - Suboptimal with respect to GAMMs
  - Can fail with many categories
  - Can fail when noise is linearly related to variation of interest

Road map

CURVES



Interpolate using a function basis

- Data driven

Dimensionality reduction tool

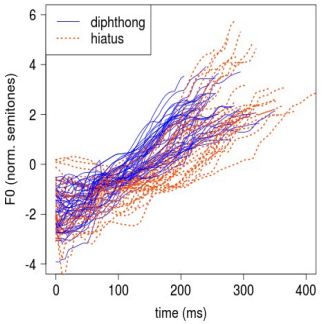
- Few parameters
- Interpretable

NUMBERS

LMER

Road map

CURVES



Interpolate using a function basis

- Data driven

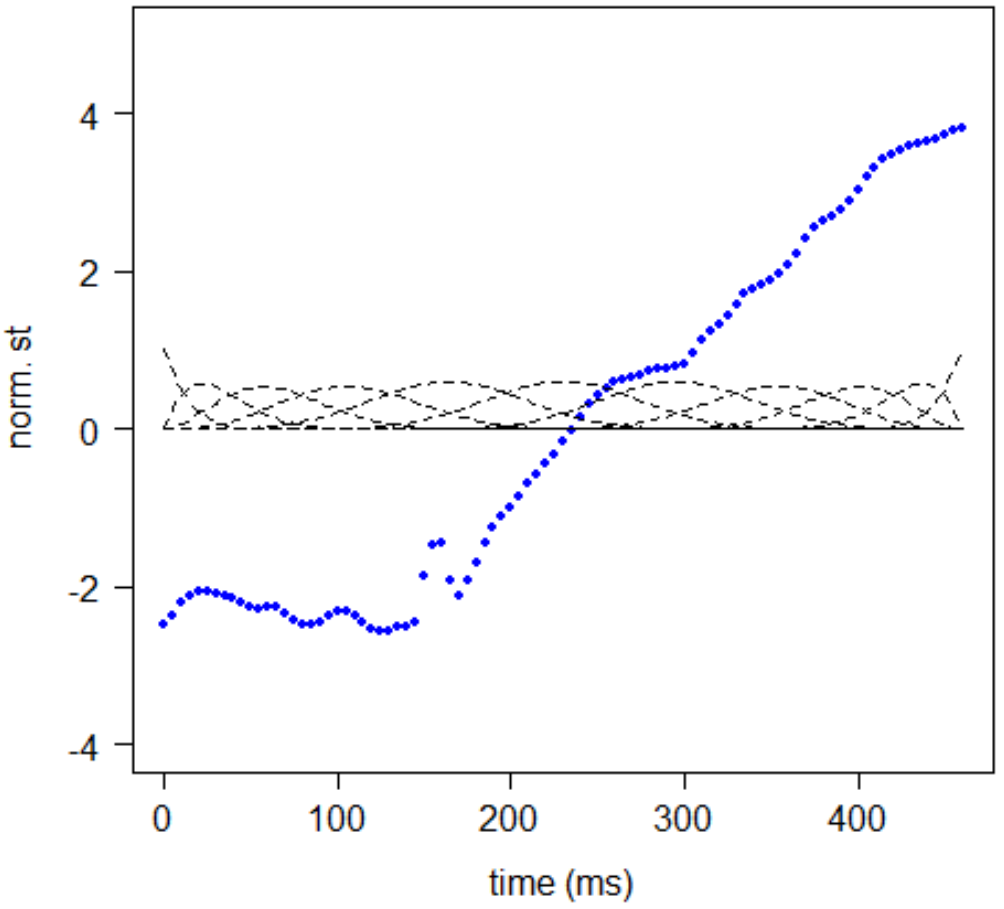
Dimensionality reduction tool

- Few parameters
- Interpretable

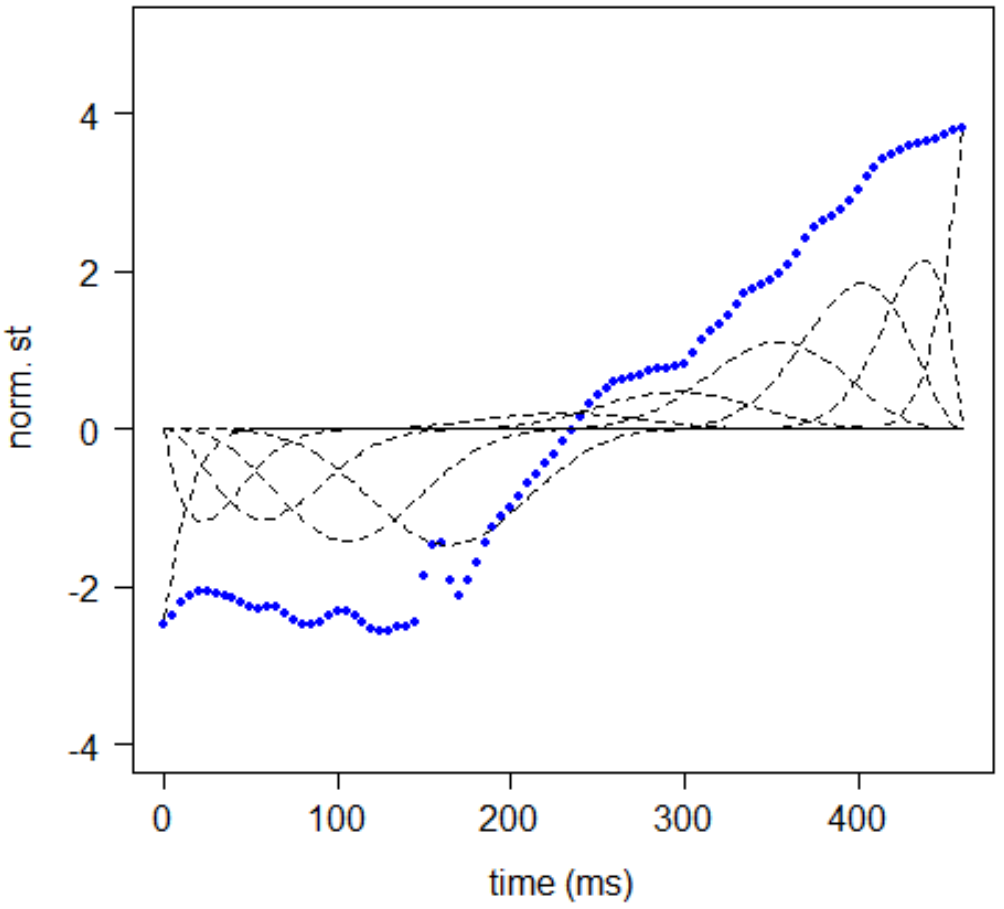
NUMBERS

LMER

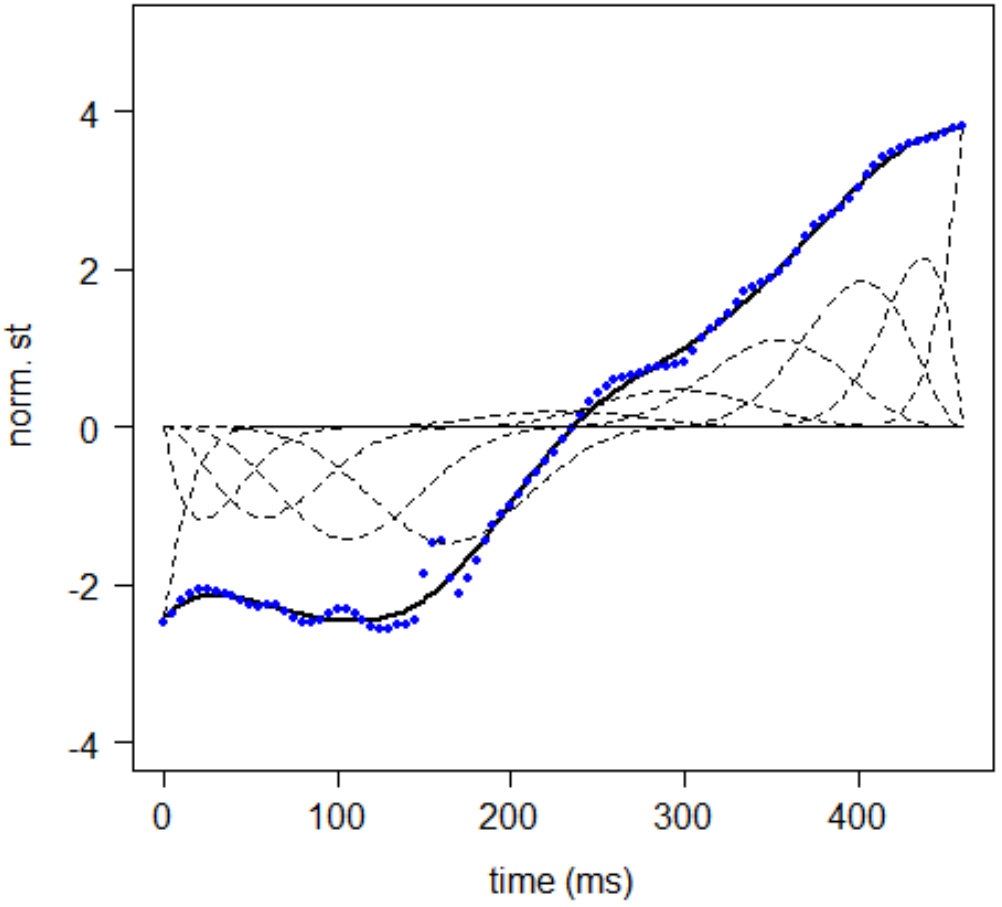
Interpolation with B-splines



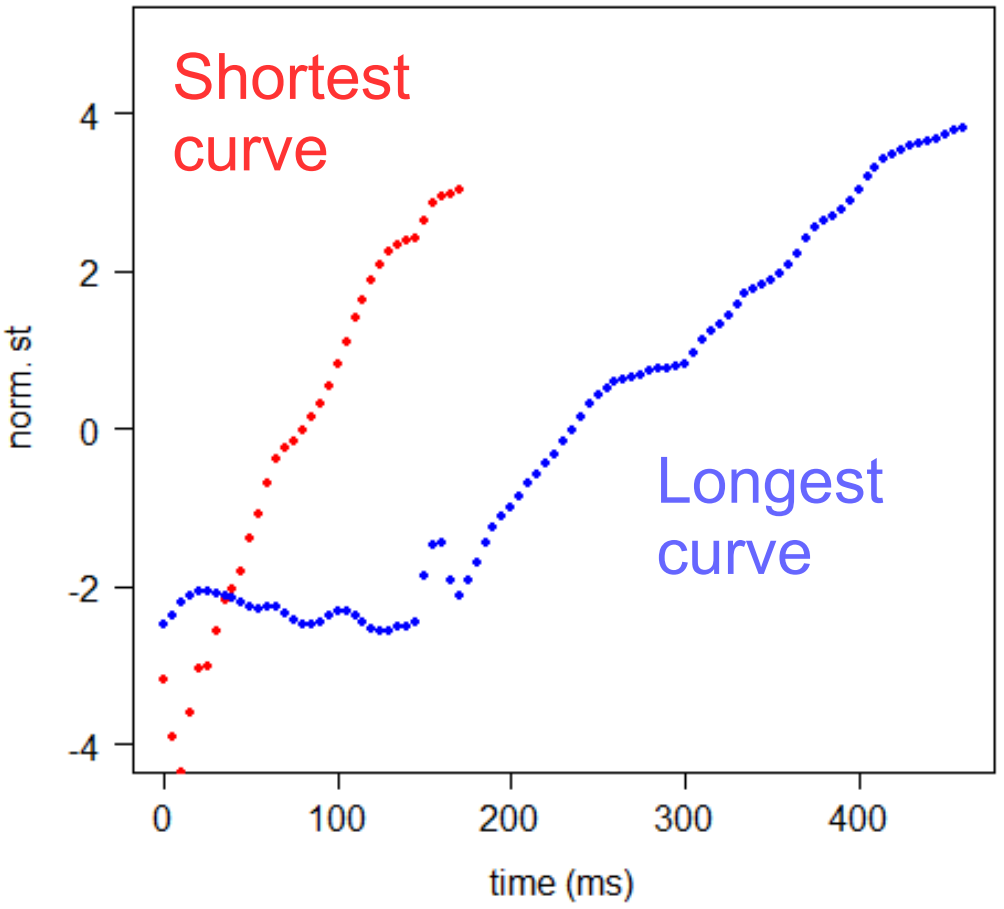
Interpolation with B-splines



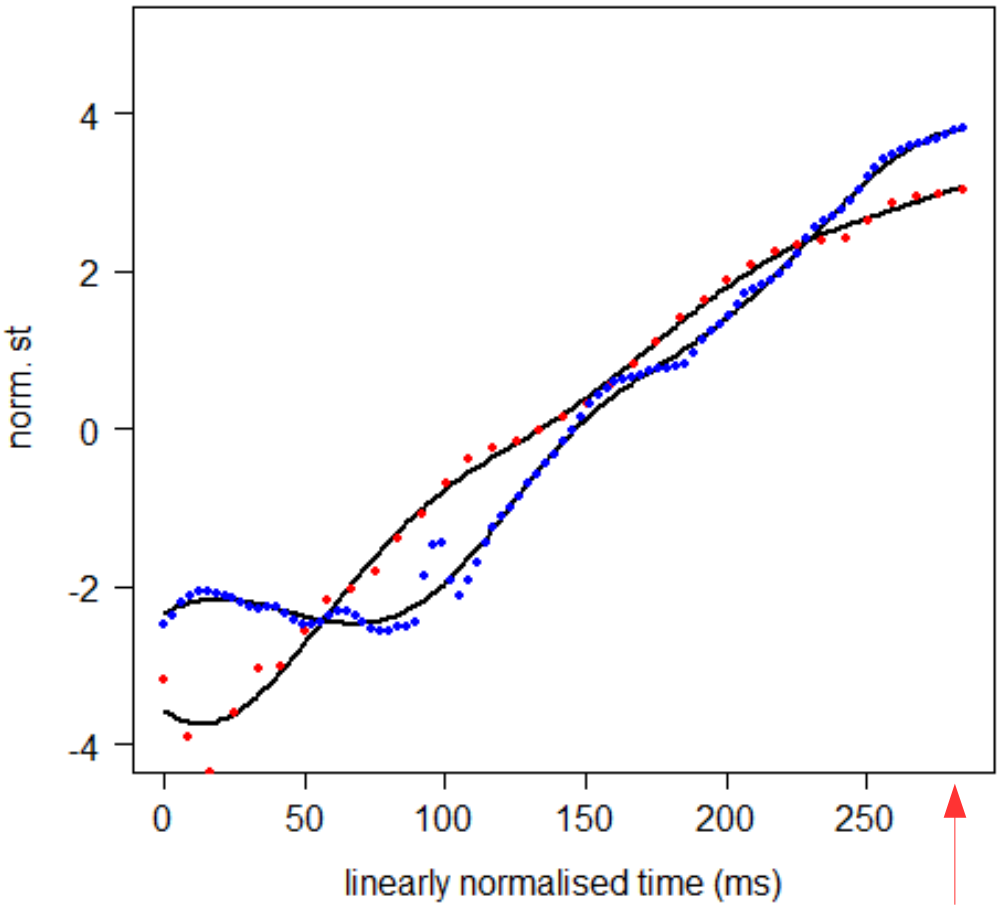
Interpolation with B-splines



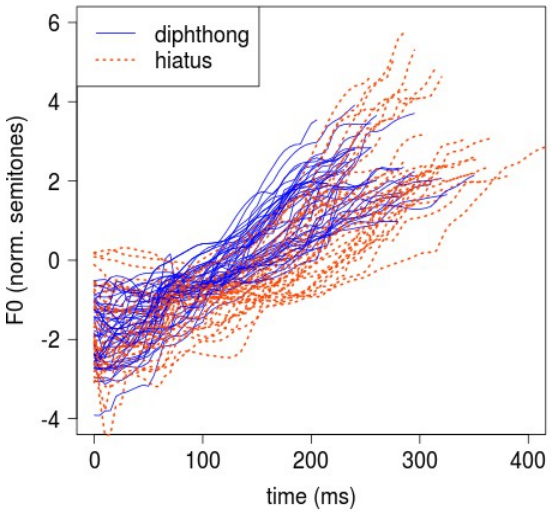
Different durations



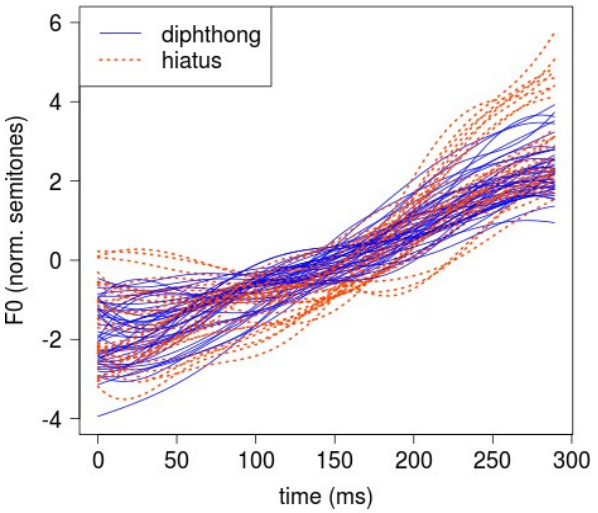
Linear time normalisation



Linear time normalisation



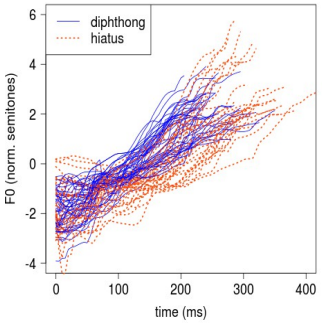
Interpolate to
the same
time interval



- We must use the same time interval
- This implies linear time normalisation
- Durations have to be reintroduced at the end of the analysis

Road map

CURVES



Interpolate to
<u>the same</u>
time interval

- Data driven

Dimensionality
reduction tool

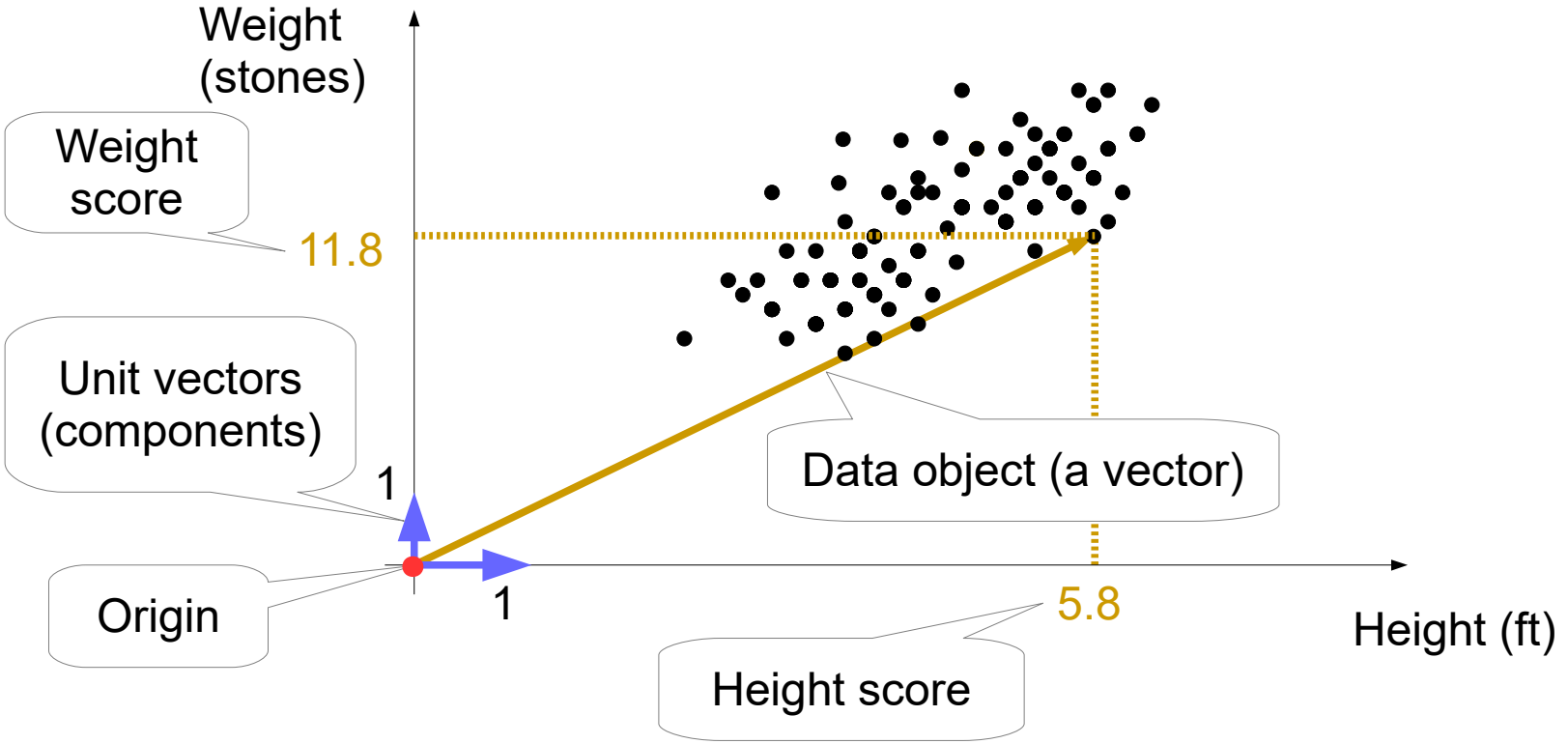
- Few parameters
- Interpretable

NUMBERS

LMER

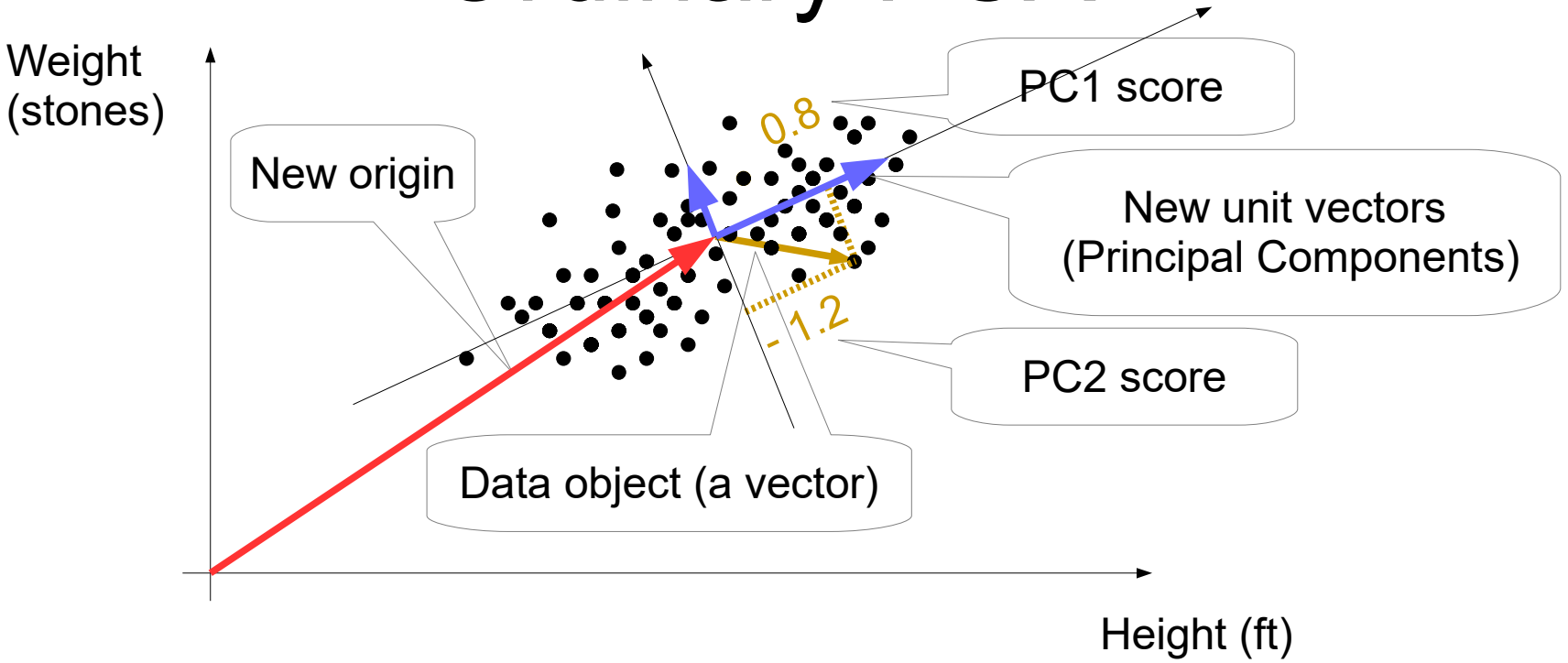
Introducing Functional PCA

Vectors



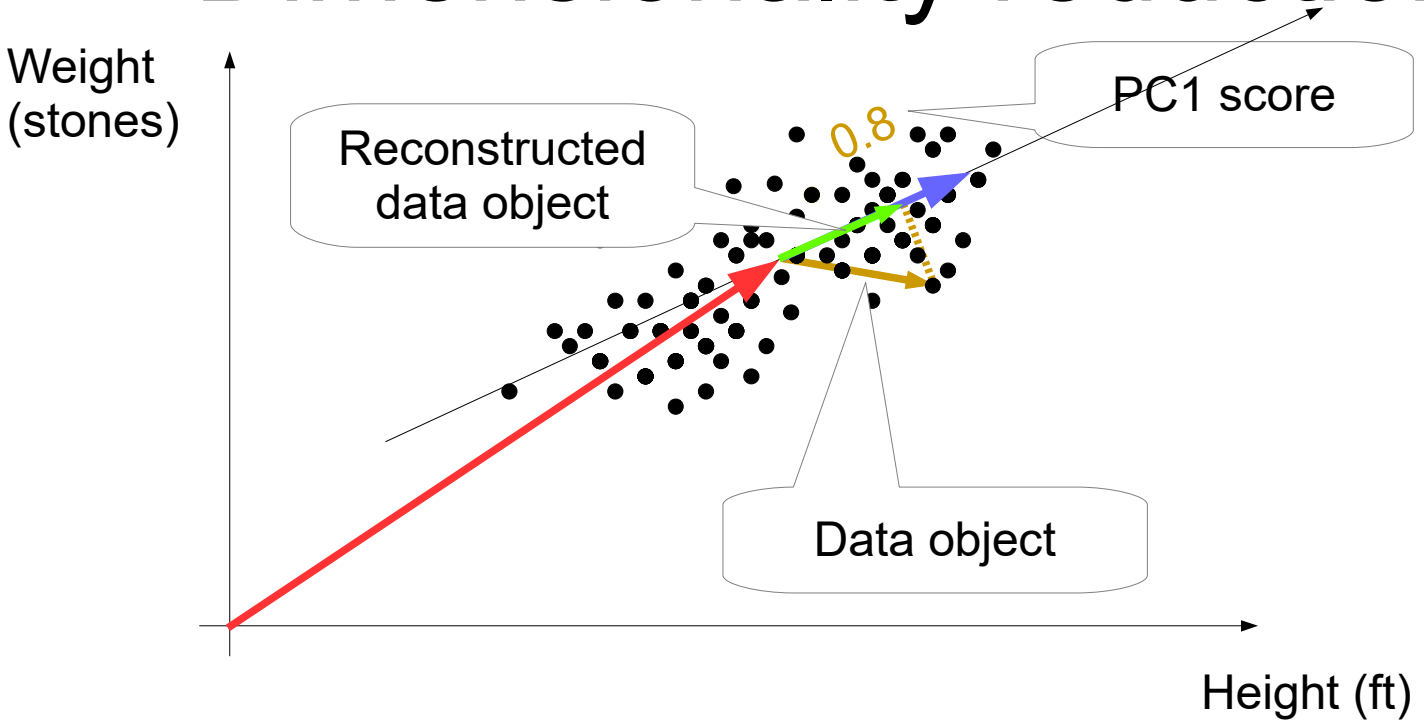
- Data objects and components are vectors
- From scores (numbers) we can reconstruct data objects (vectors)

Ordinary PCA

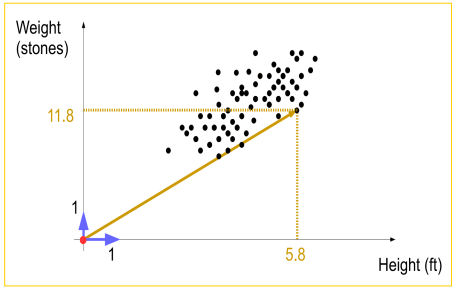


- PCA computes new origin and unit vectors which best suit the data
- From PC scores we can reconstruct data objects

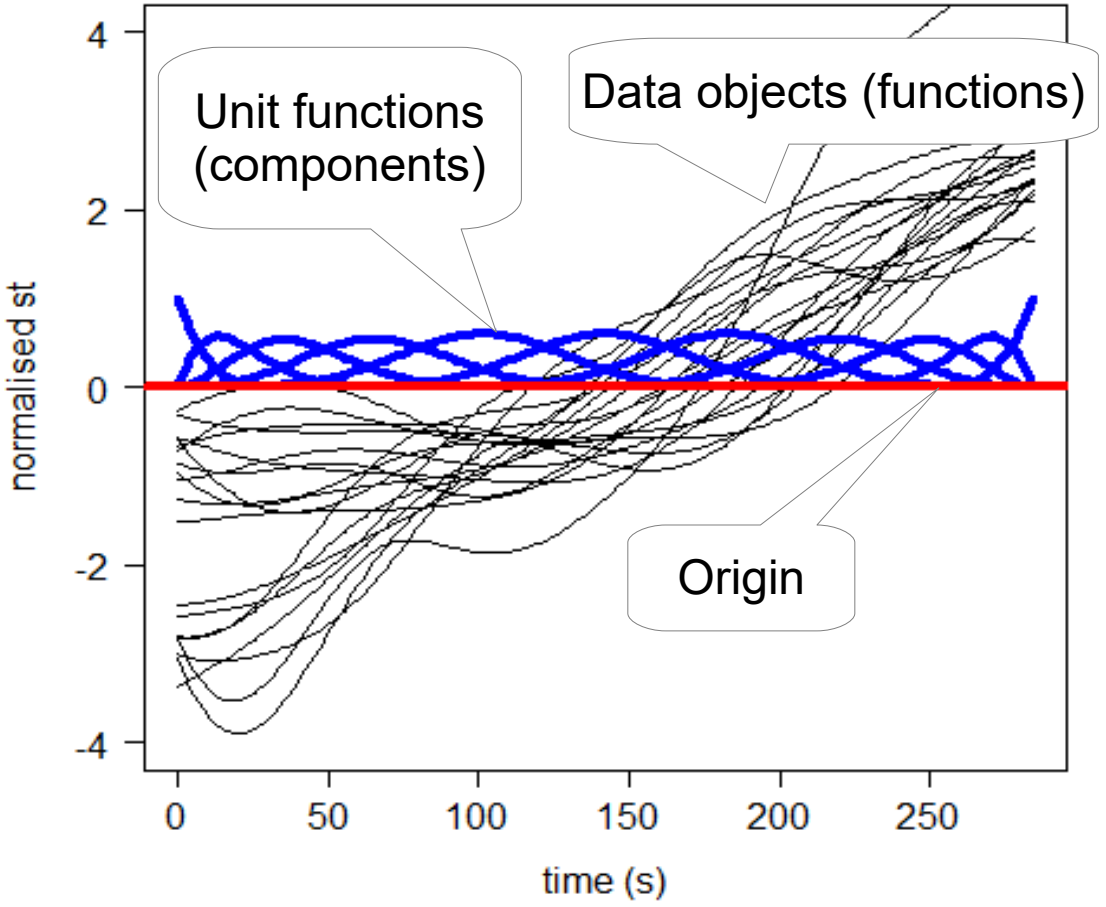
Dimensionality reduction



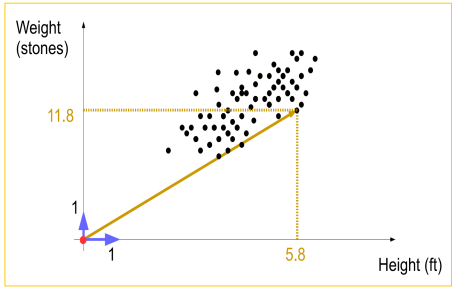
- We can use only part of the PCs
- This reduces the data dimensionality
- But introduces reconstruction errors too



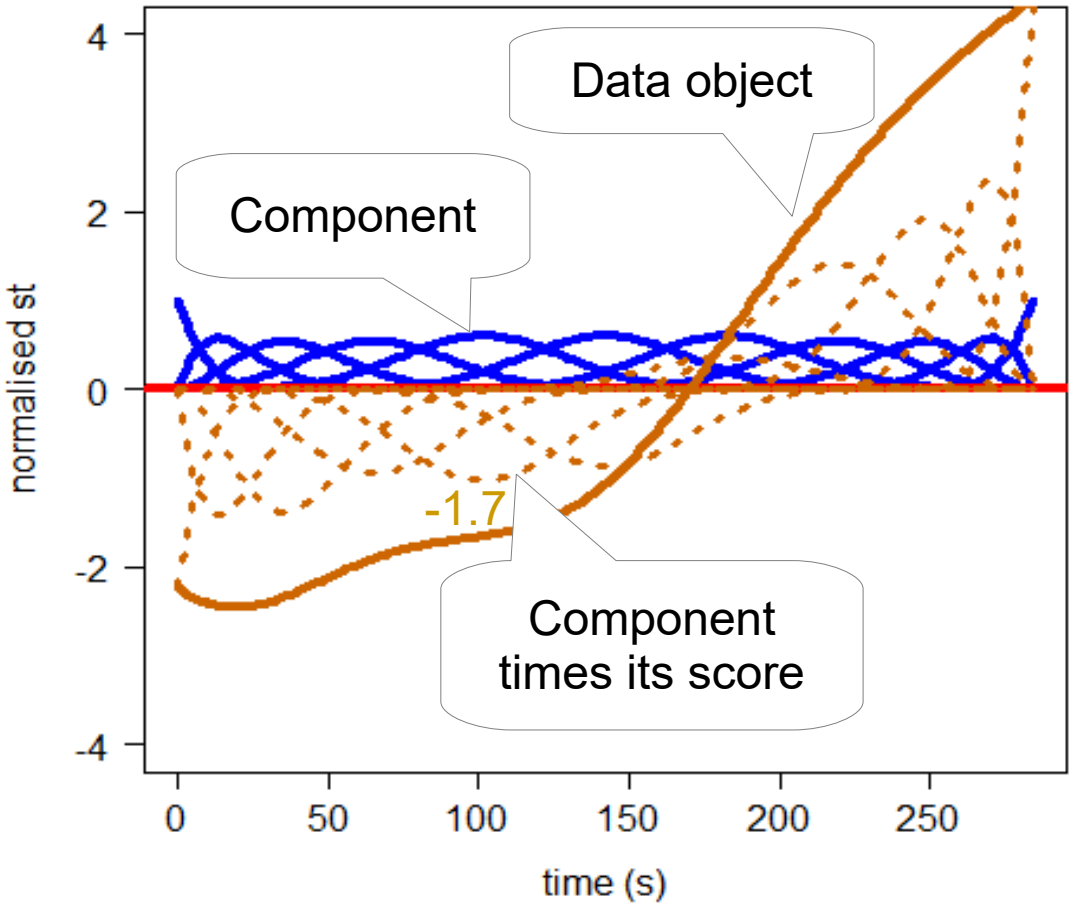
Functions (curves)



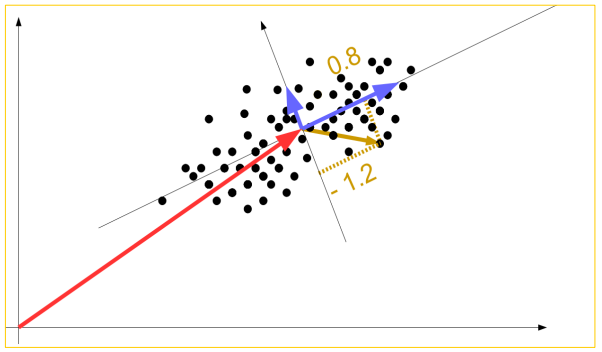
- Origin, components and data objects are functions
- Origin is a flat line
- Components are 11 B-spline curves



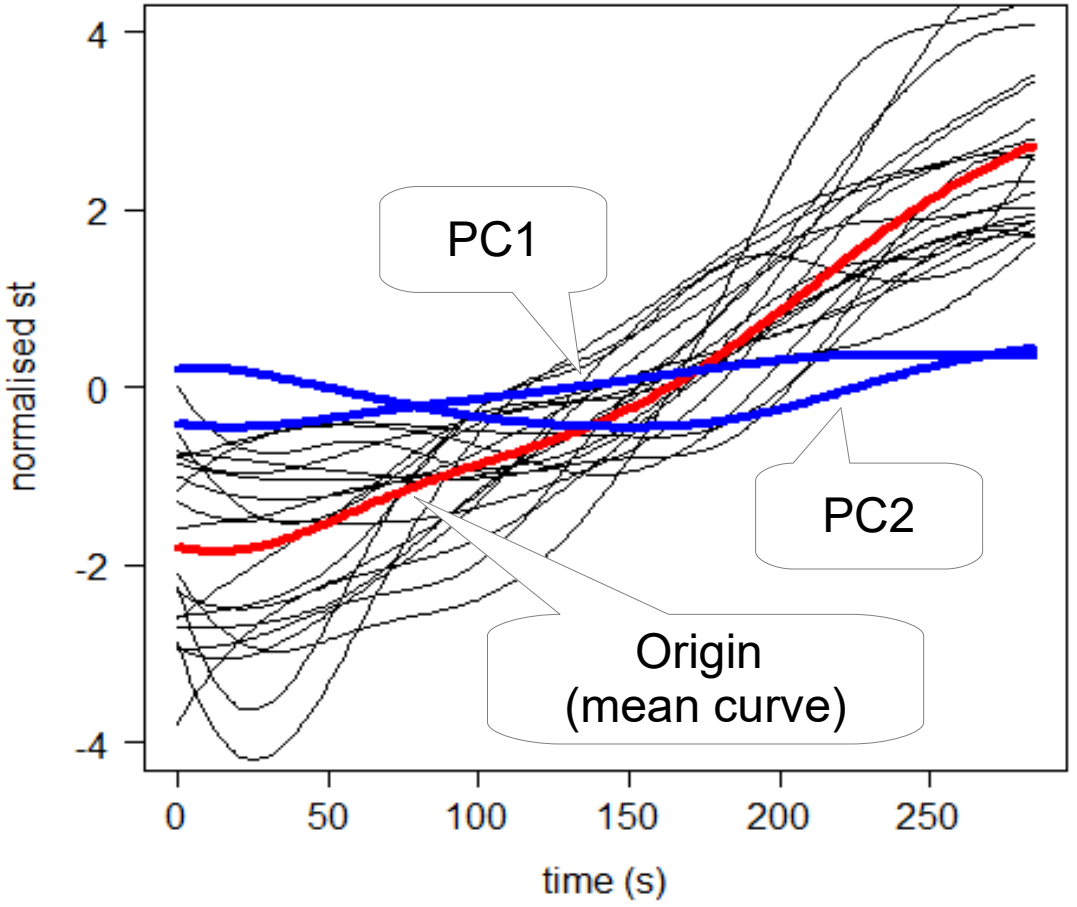
Functions (curves)



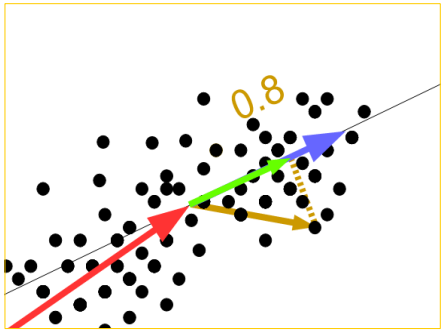
- Each of the 11 components is multiplied by a score
- These are summed together to obtain a data object



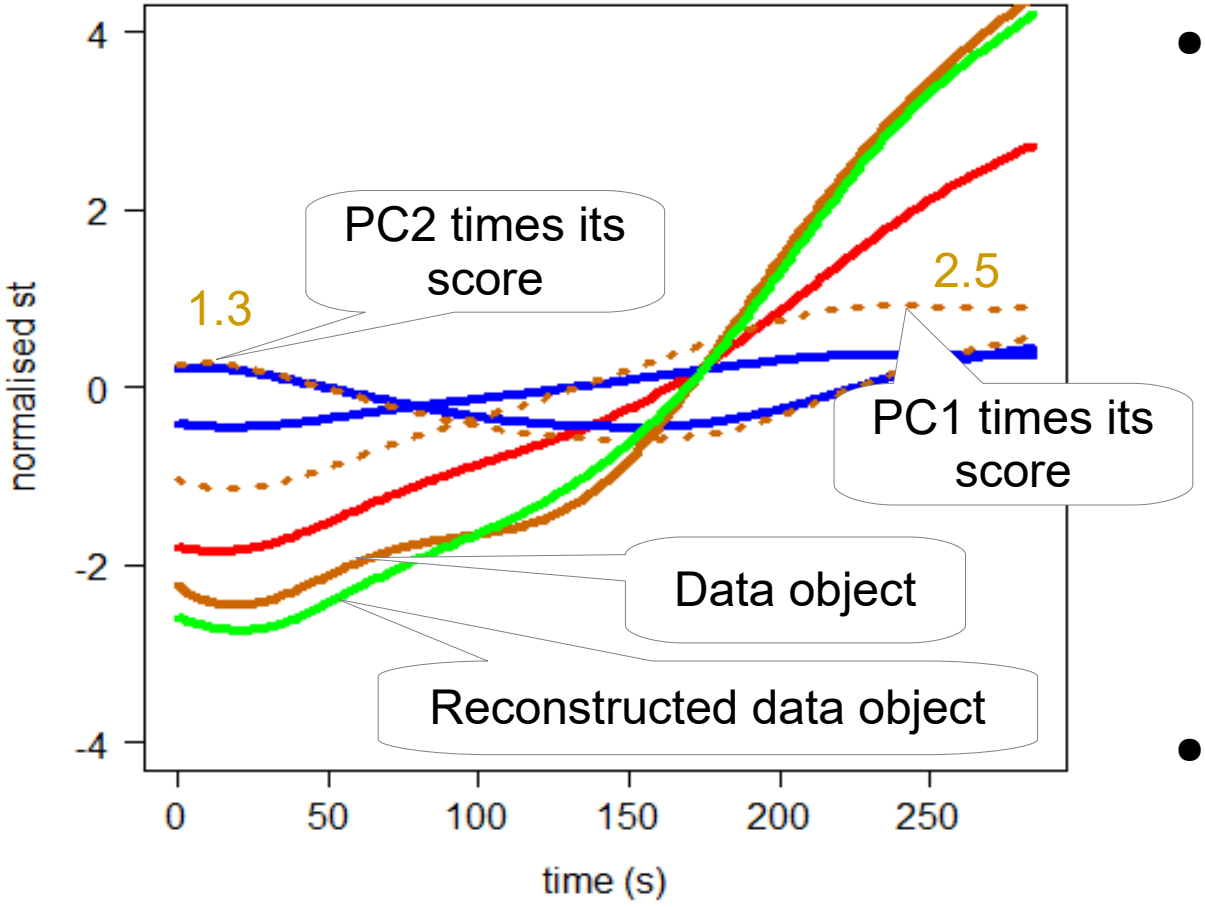
Functional PCA



- FPCA computes new origin and component functions which best suit the data



Functional PCA



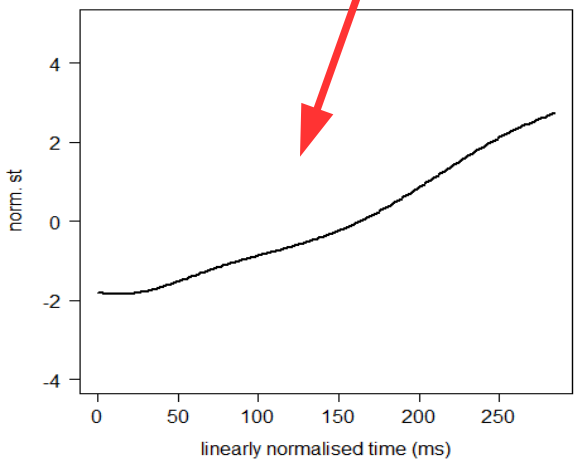
- The sum of origin (mean) curve + PCs times their scores gives an approx reconstruction of the original curve
- Dimensions from 11 (B-splines) down to 2 (PCs)

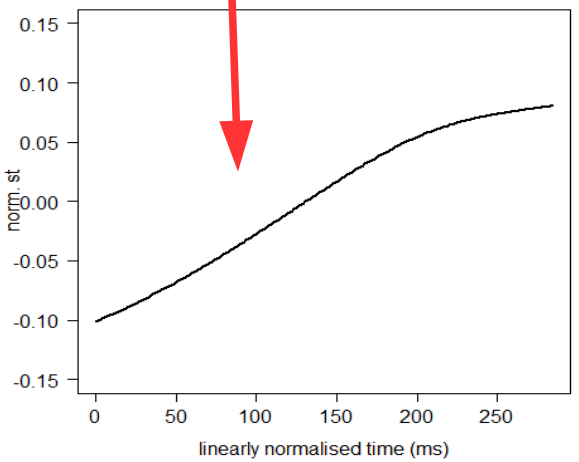
Functional PCs

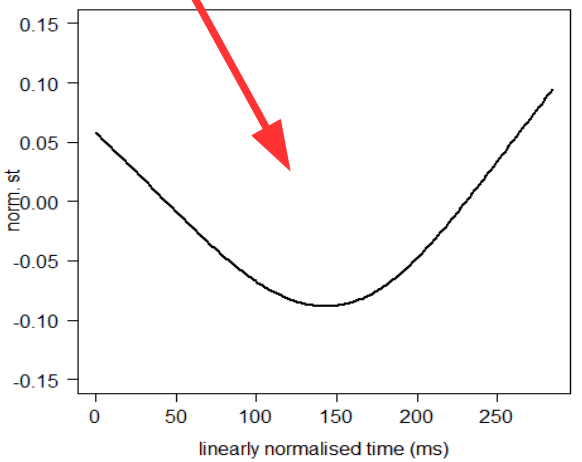
$$f(t) \approx \mu(t) + s_1 \cdot PC1(t) + s_2 \cdot PC2(t) + \cdots$$

PC1 score

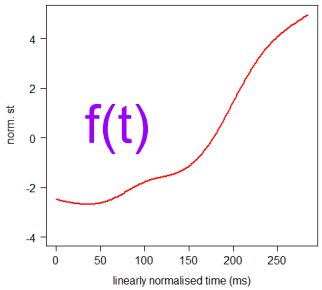
PC2 score



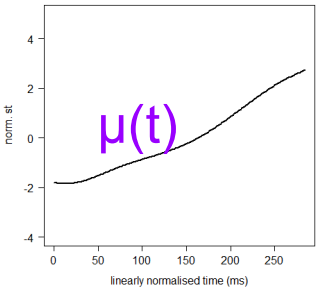




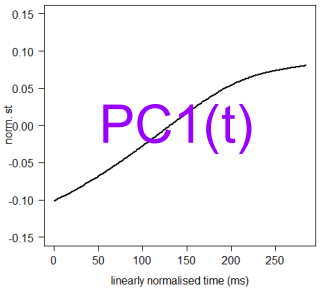
Curve reconstruction



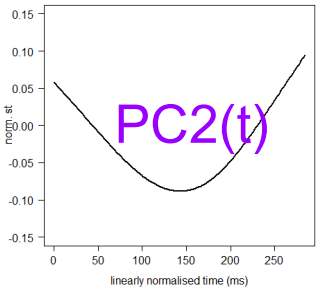
≈

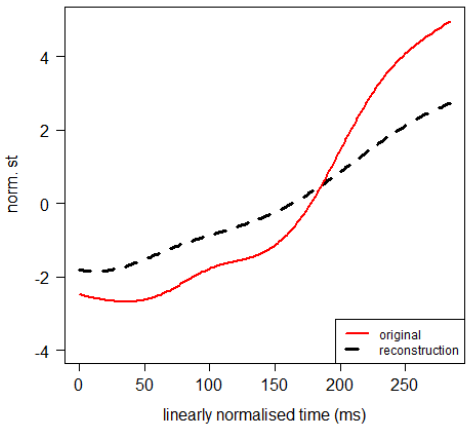


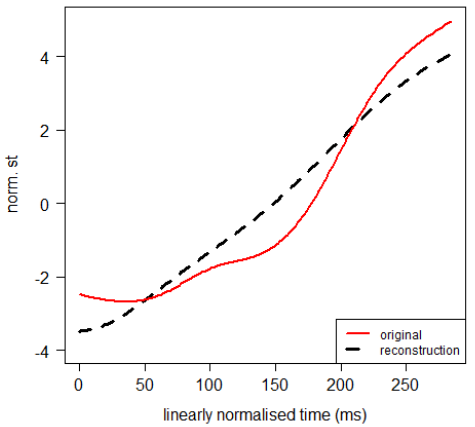
+ 16.5 *

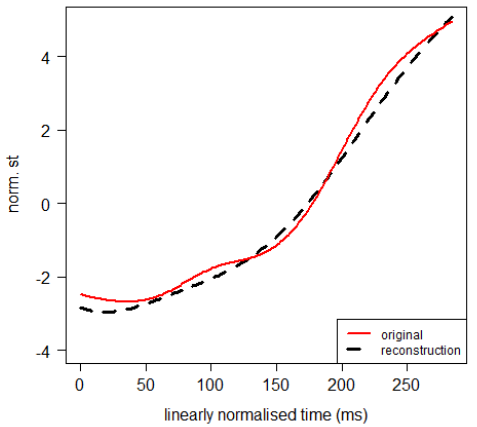


+ 10.8 *

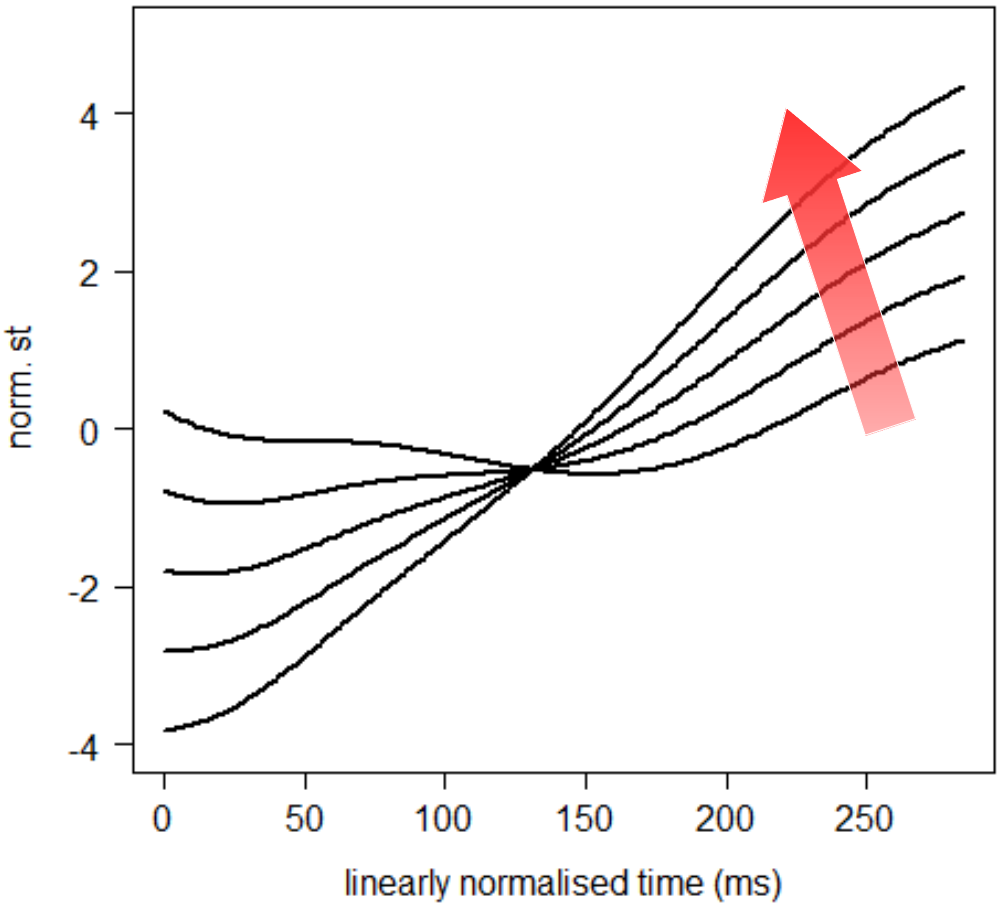






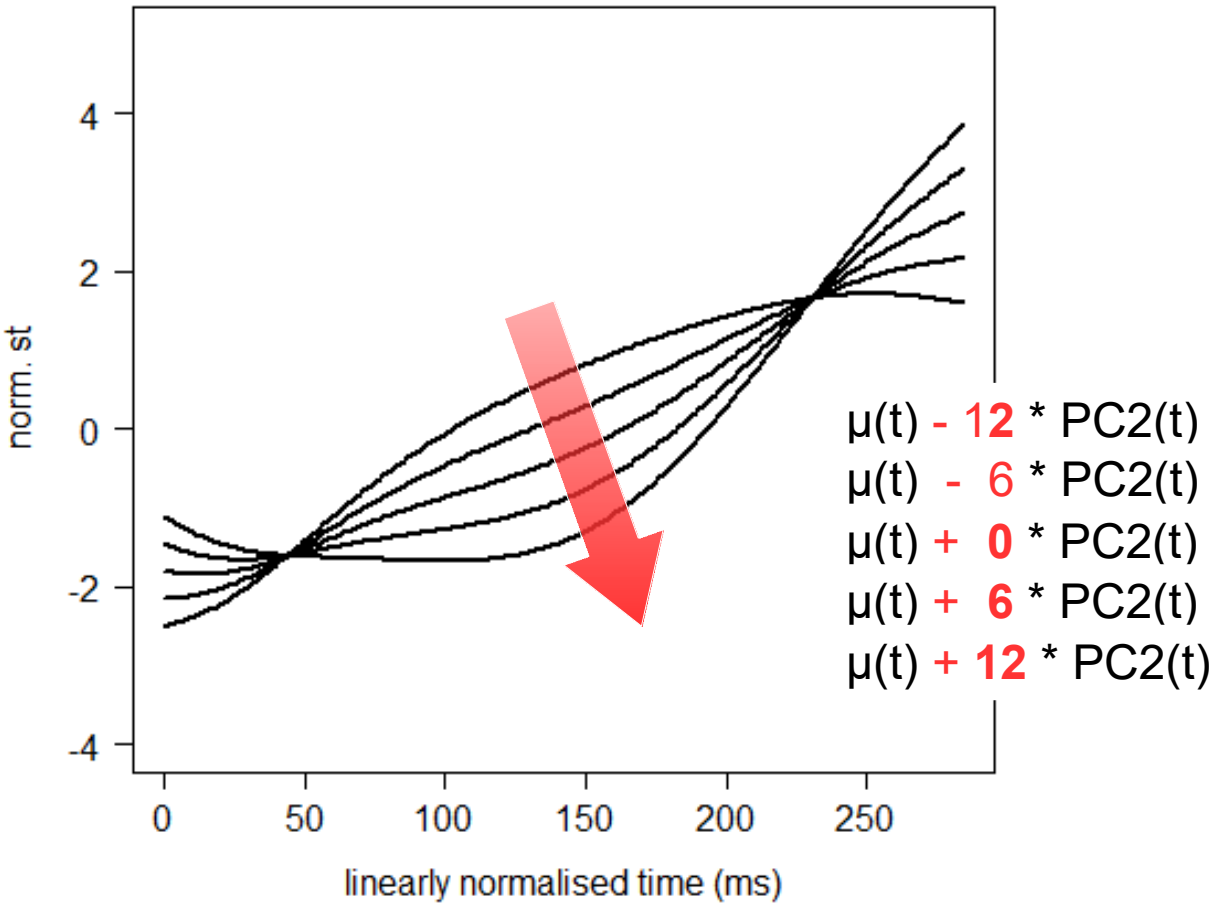


PC1 scores

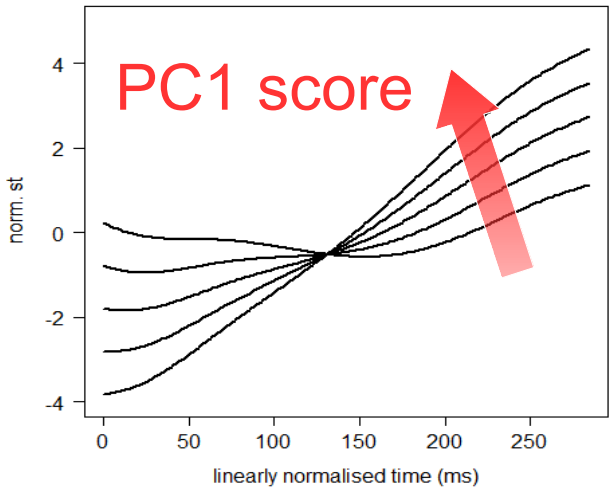


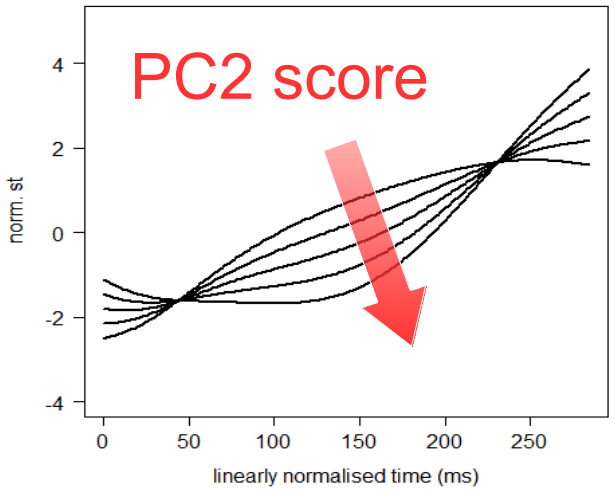
μ(t) + **20** * PC1(t)
μ(t) + **10** * PC1(t)
μ(t) + **0** * PC1(t)
μ(t) - **10** * PC1(t)
μ(t) - **20** * PC1(t)

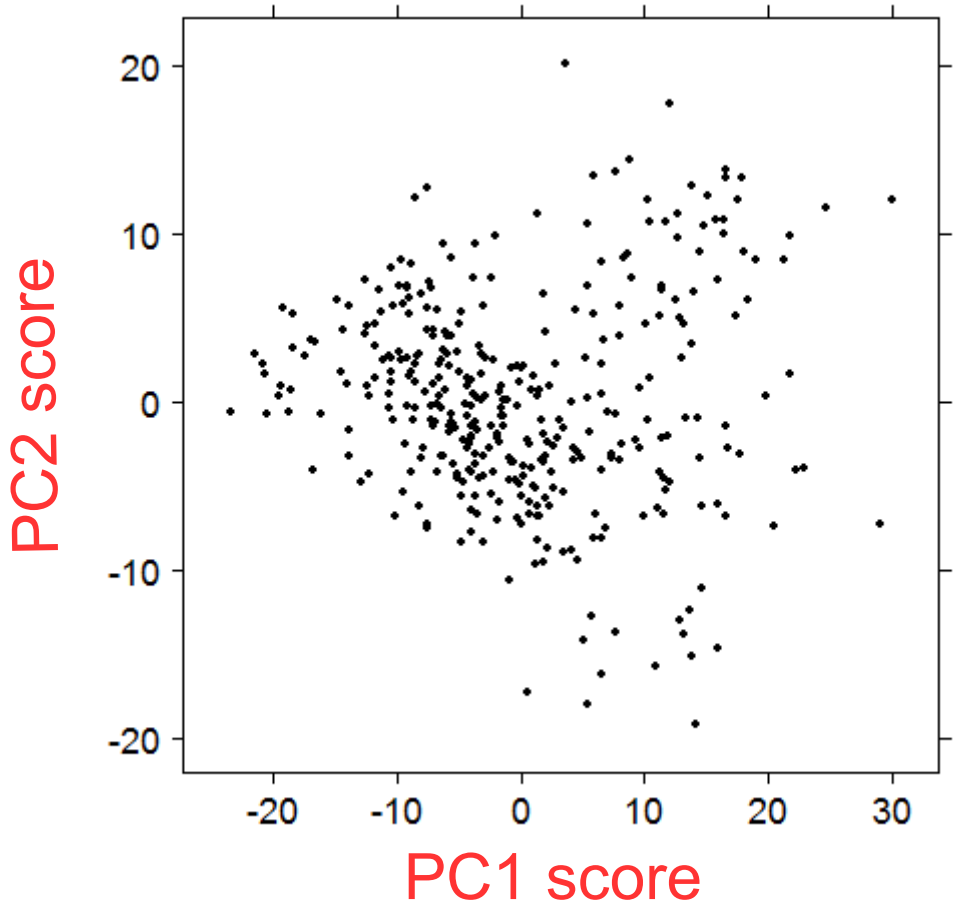
PC2 scores


4
2
0
-2
-4
norm. st
0
50
100
150
200
250
linearly normalised time (ms)
μ(t) - 12 * PC2(t)
μ(t) - 6 * PC2(t)
μ(t) + 0 * PC2(t)
μ(t) + 6 * PC2(t)
μ(t) + 12 * PC2(t)

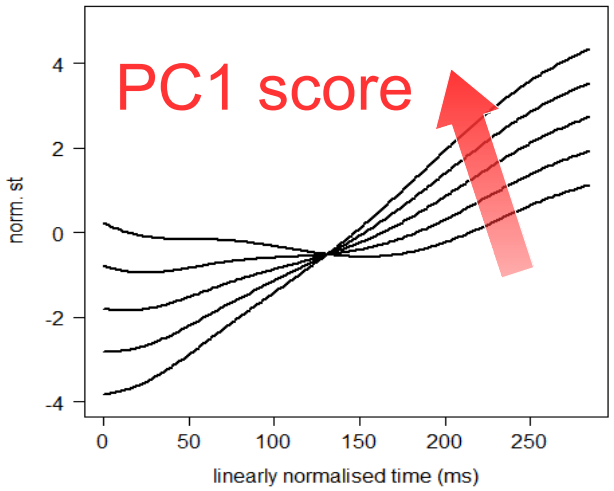
Curve parametrisation

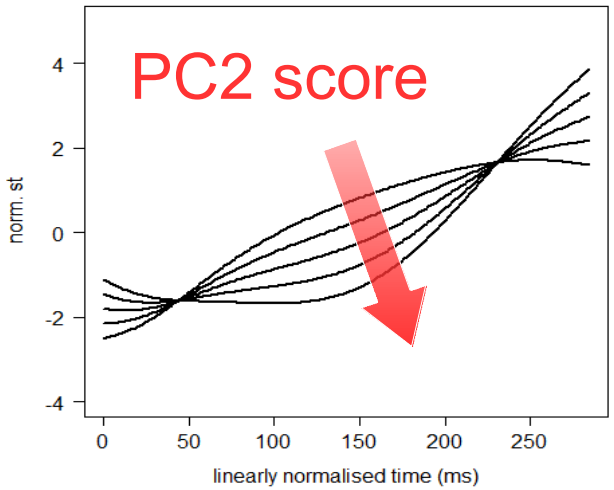


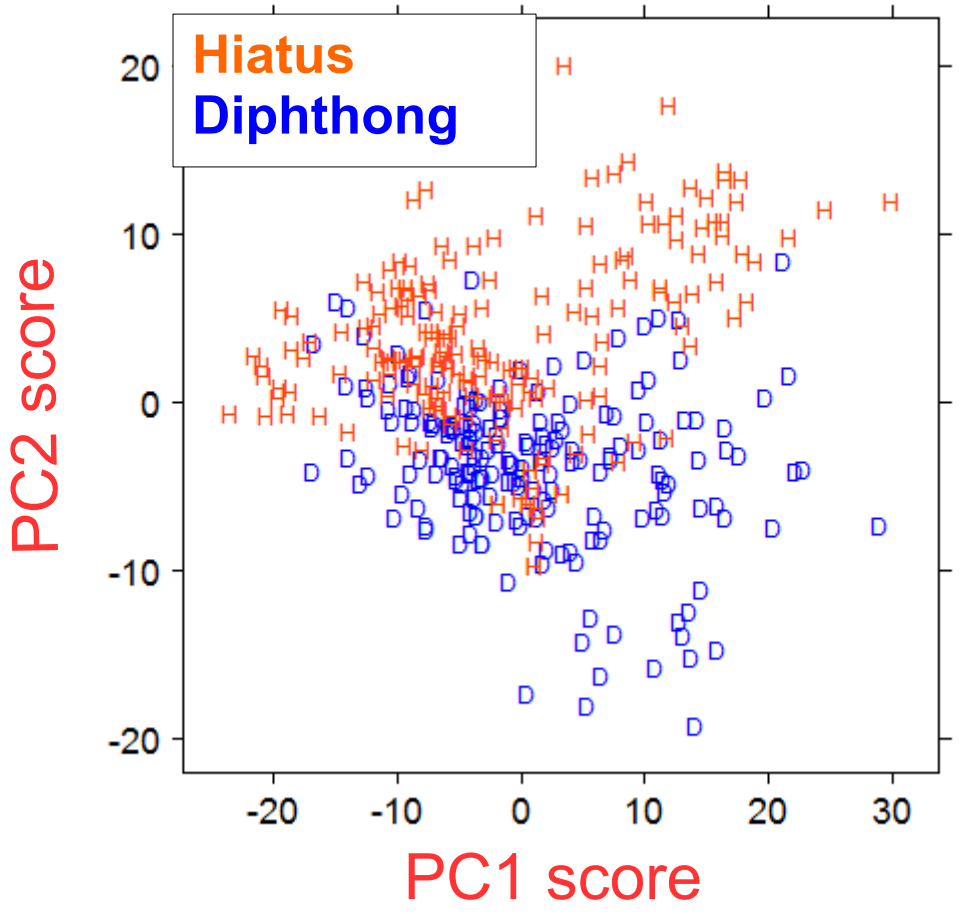




Curve parametrisation



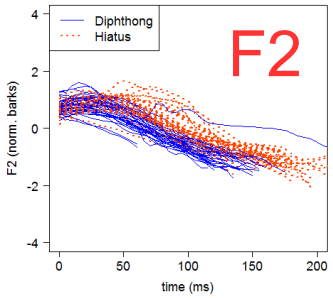


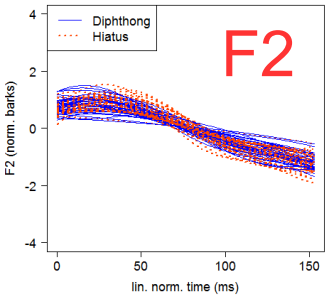


Multidimensional signals

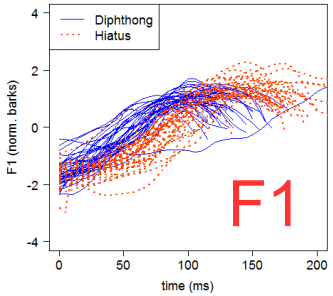
Formants

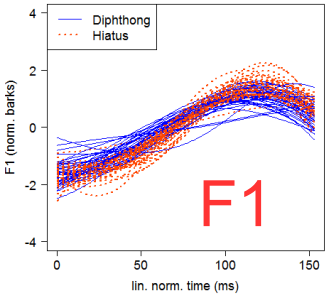
2D CURVES





FPCA





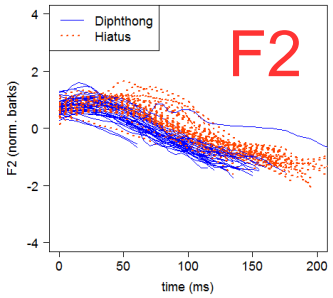
FPCA

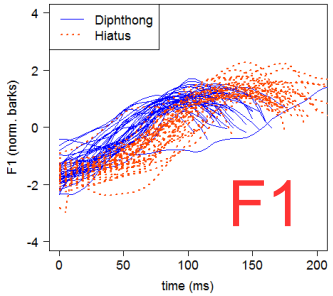
NUMBERS

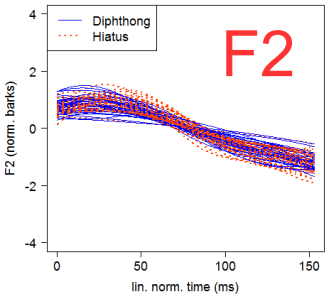
LMER

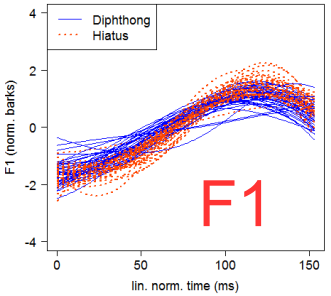
Formants

2D CURVES









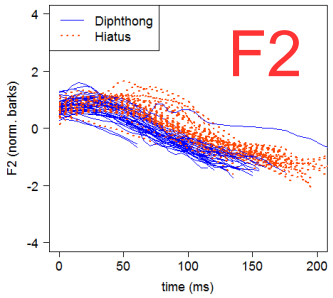
2D
FPCA

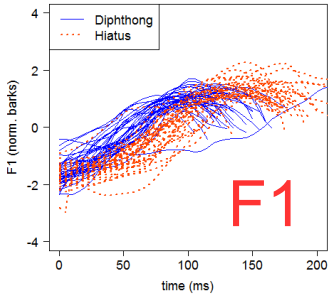
NUMBERS

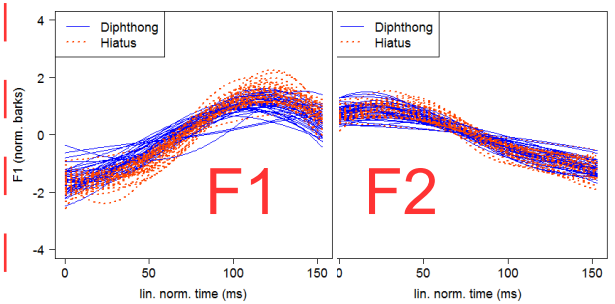
LMER

Trick

2D CURVES





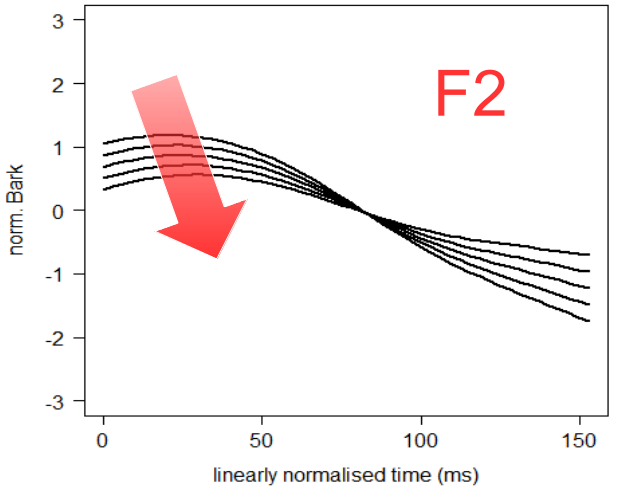


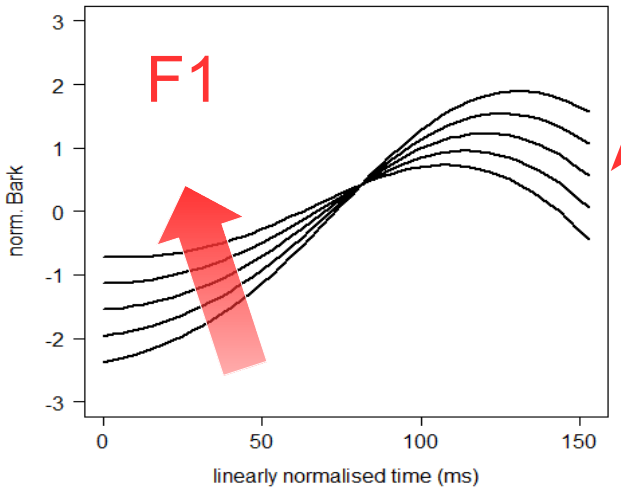
FPCA

NUMBERS

LMER

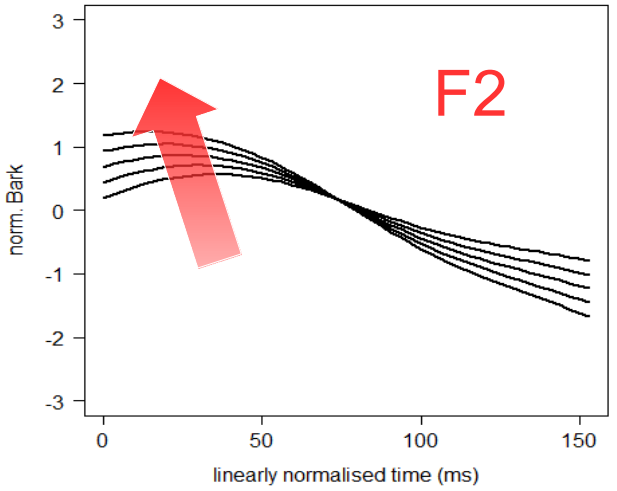
PC1 scores

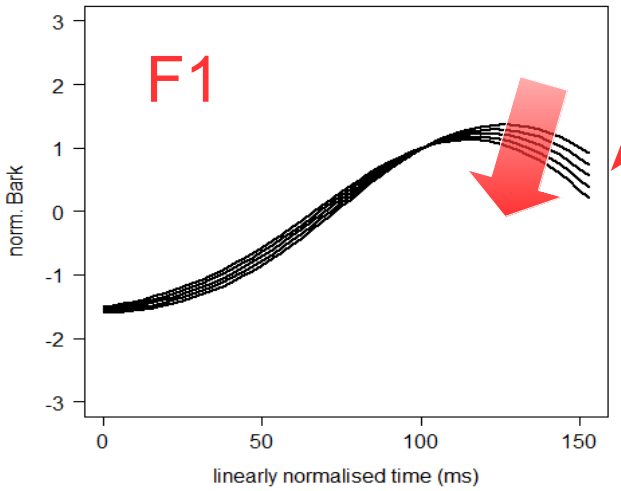




μ(t) + **8** * PC1(t)
μ(t) + **4** * PC1(t)
μ(t) + **0** * PC1(t)
μ(t) - **4** * PC1(t)
μ(t) - **8** * PC1(t)

PC2 scores

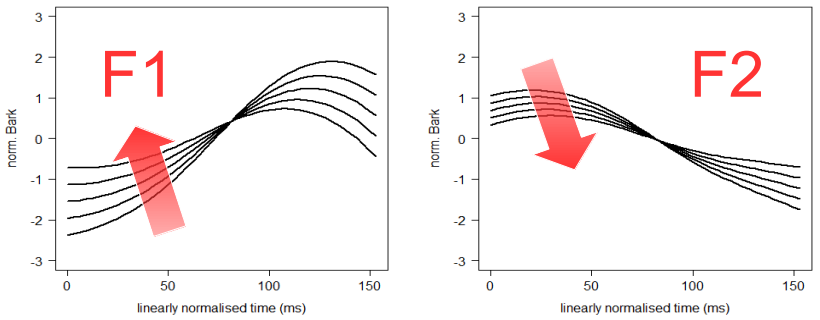




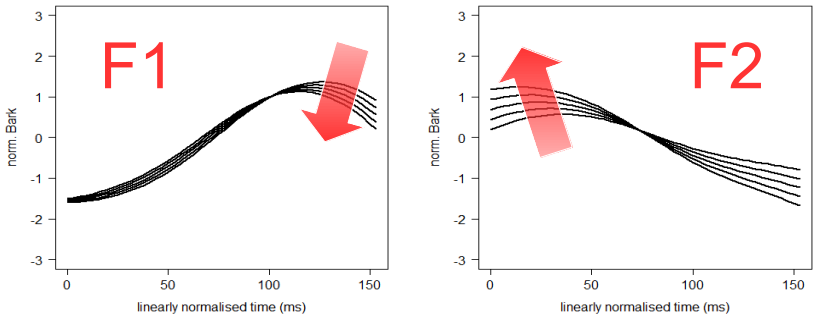
μ(t) + **4** * PC1(t)
μ(t) + **2** * PC1(t)
μ(t) + **0** * PC1(t)
μ(t) - **2** * PC1(t)
μ(t) - **4** * PC1(t)

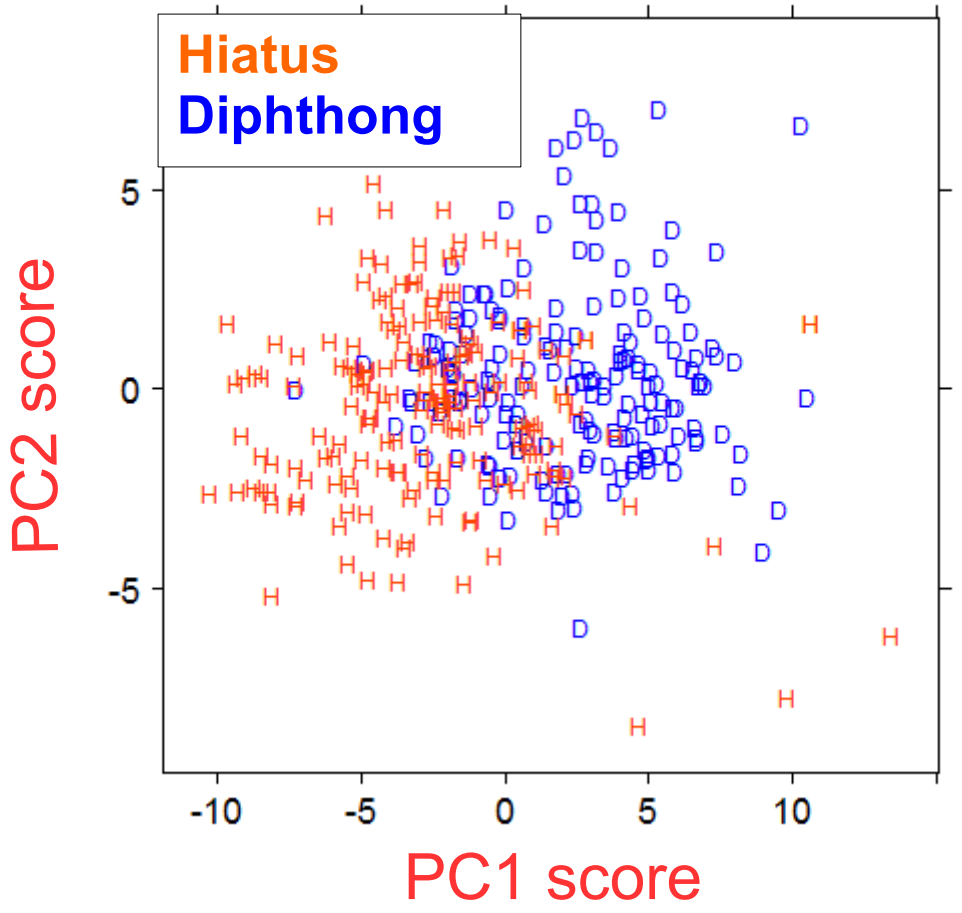
2D curve parametrisation

PC1 score



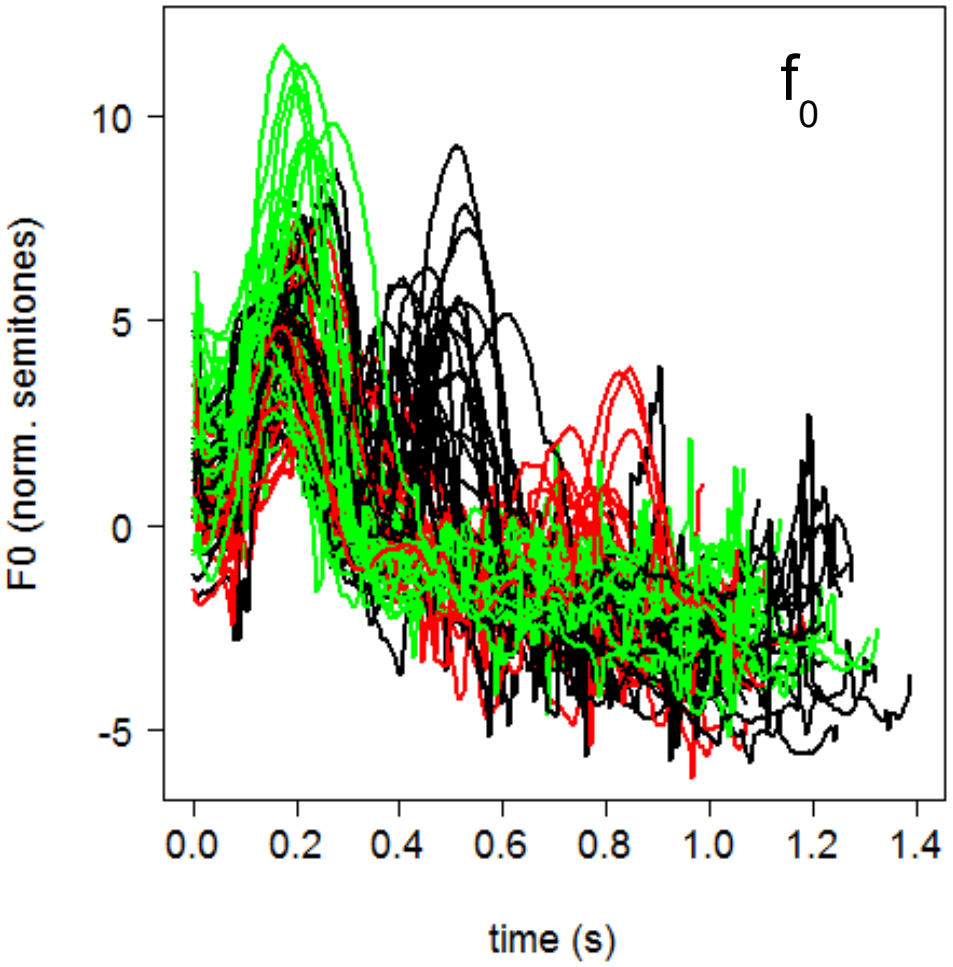
PC2 score





Long signals

Many segments



- Narrow focus in Neapolitan Italian
- Focus on

  Subject, **Verb** or **Prop. Phrase**

  Danilo vola da Roma

  (*Danilo flies from Rome*)
- 8 CV syllables

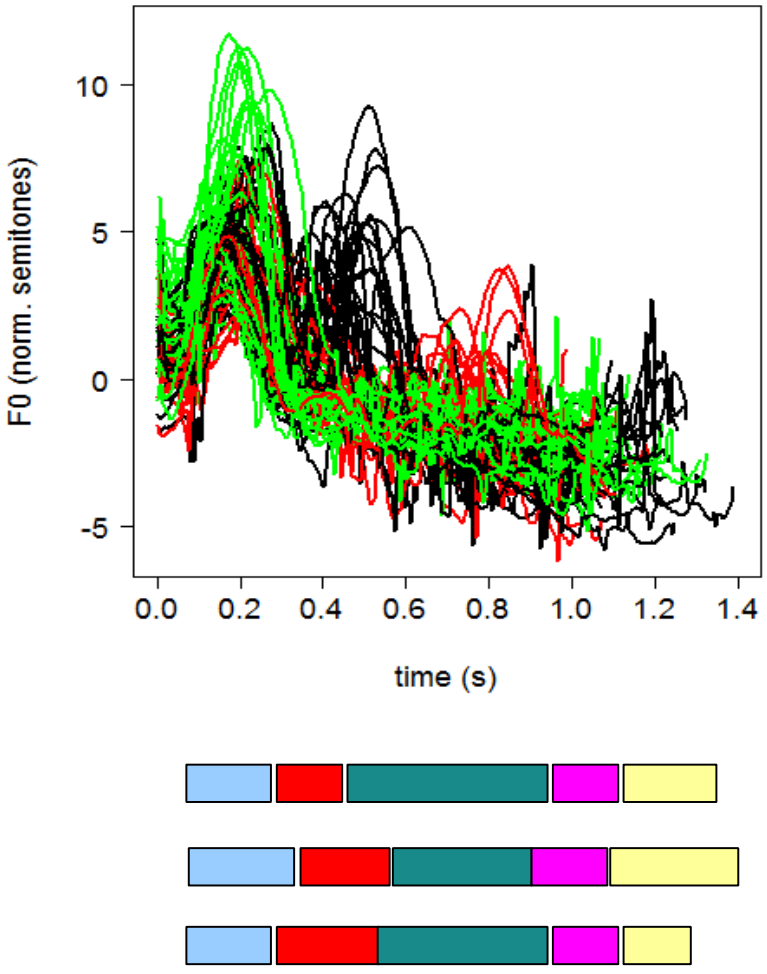
  first C was excluded (too short)

  VCVCV CVCV CV CVCV

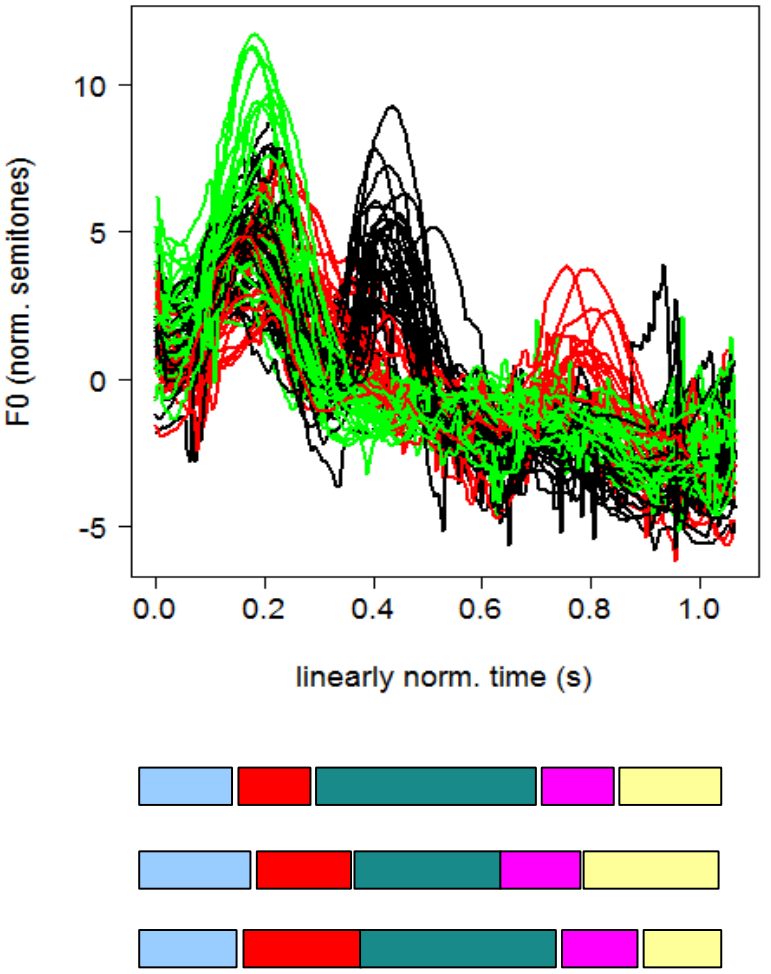
… 15 segments!

Linear time normalisation

BEFORE

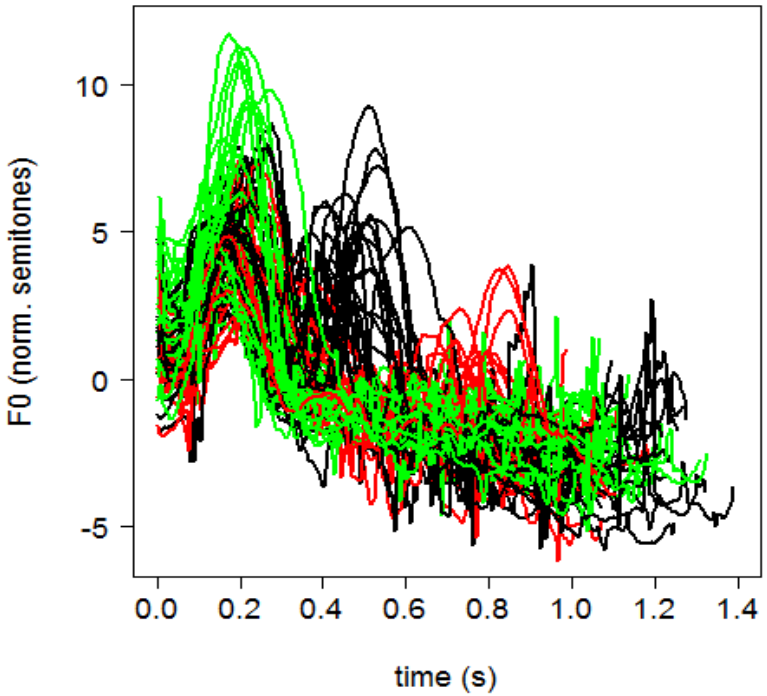


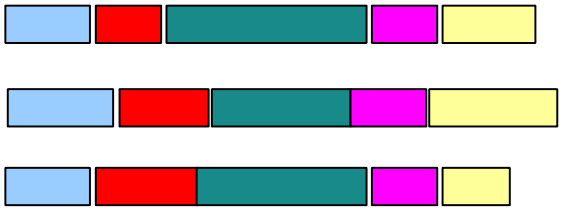
AFTER



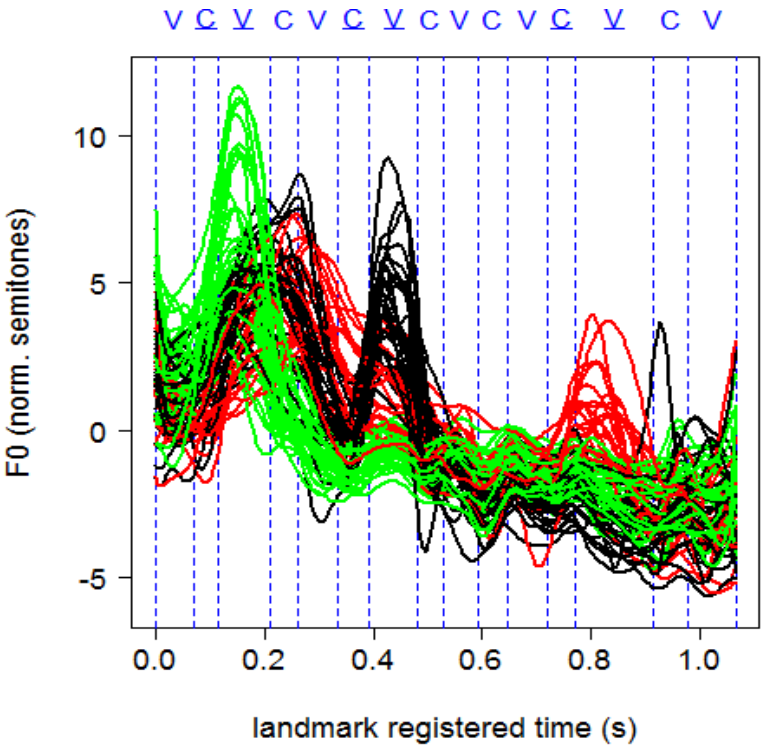
Landmark registration

BEFORE



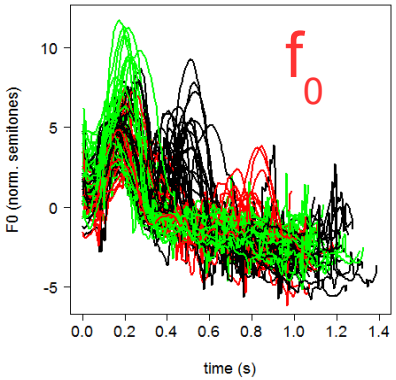


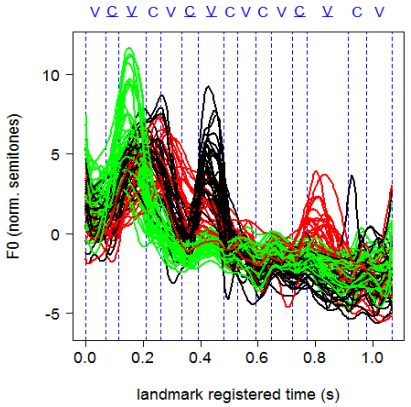
AFTER



Using landmark registration

CURVES





segment durations

d1	d2	...	d15
...	...	...	...
...	...	...	...
...	...	...	...

FPCA

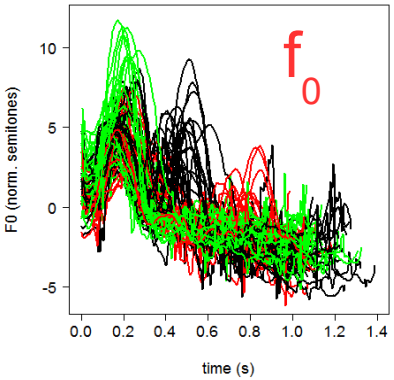
PCA

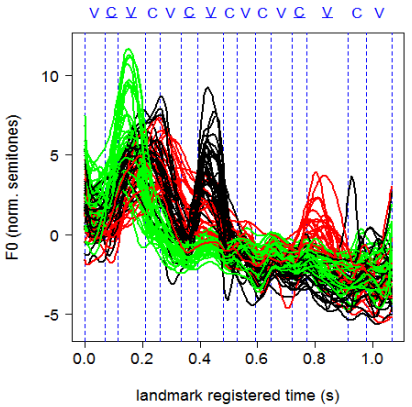
NUMBERS

LMER

Using landmark registration

CURVES





segment durations

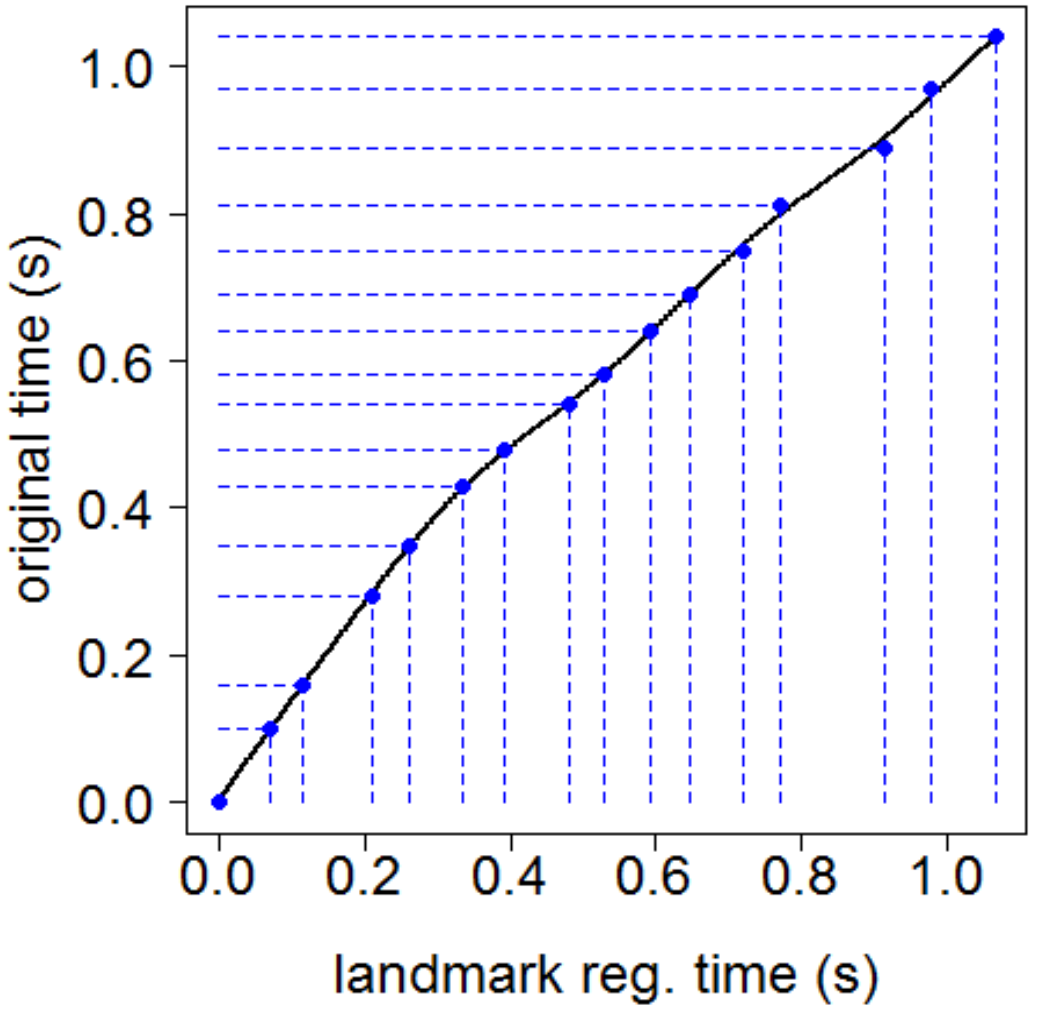
d1	d2	...	d15
...	...	...	...
...	...	...	...
...	...	...	...

FPCA

NUMBERS

LMER

Inside landmark registration

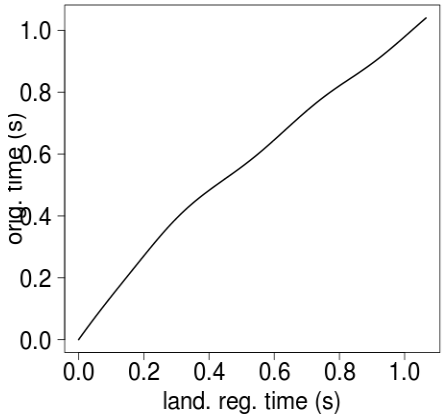


h(t)

Relative log rate

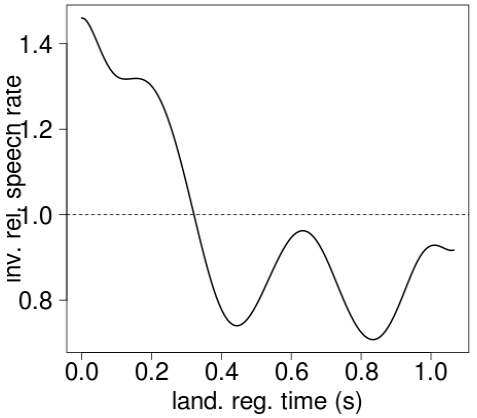
1

h(t)



2

dh(t)/dt

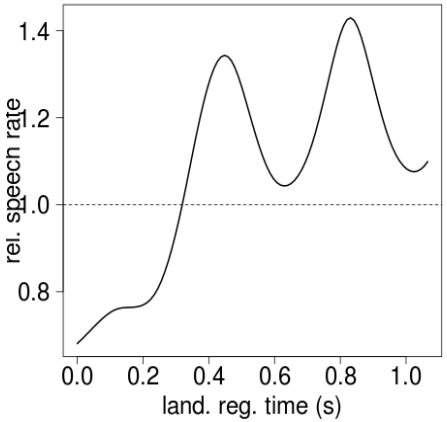


REVERSIBLE!

1 ⟷ 4

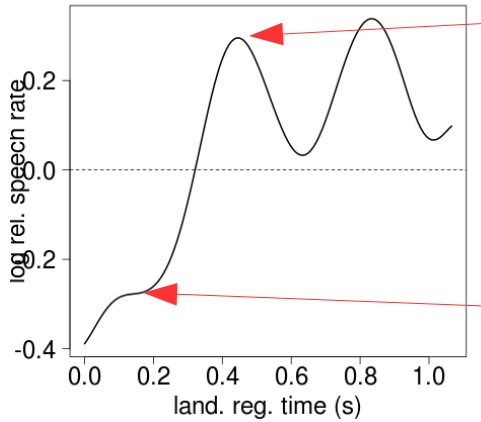
3

- dh(t)/dt



4

- log dh(t)/dt



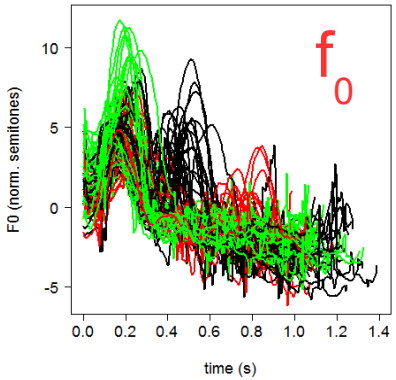
+ 0.25 → duration / 1.28

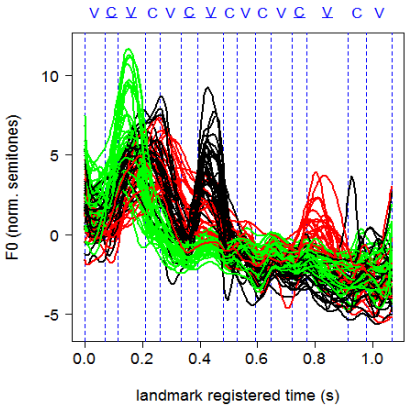
0 → same duration

- 0.25 → duration * 1.28

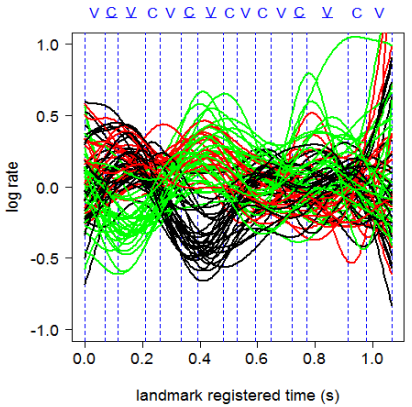
Using log rates

CURVES





log rates

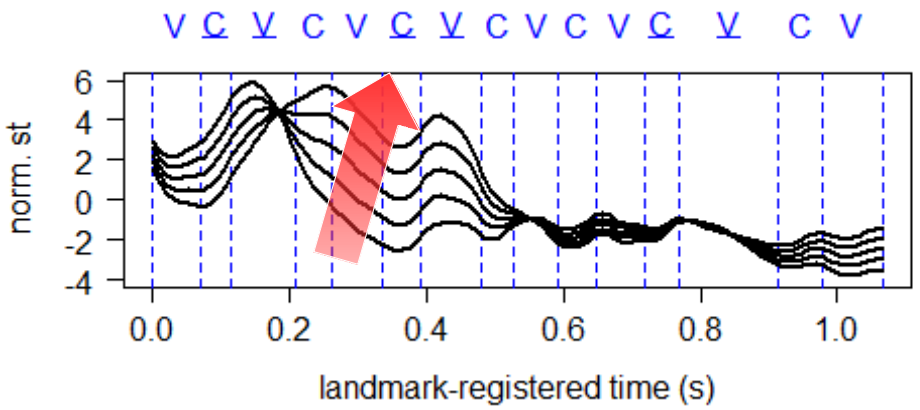


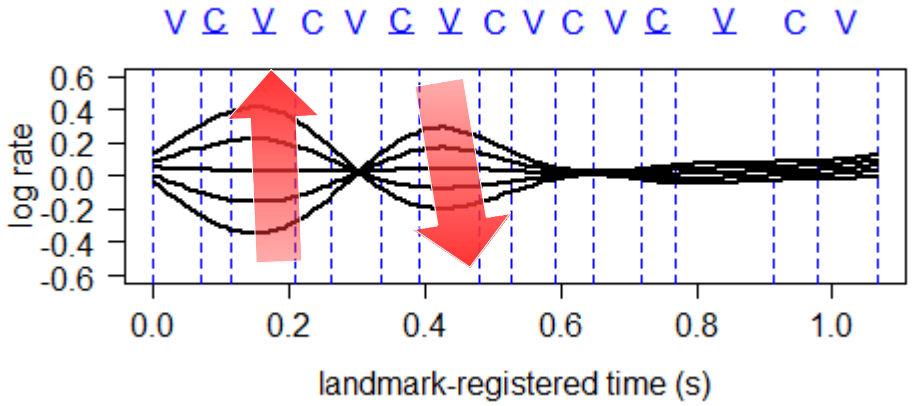
2D FPCA

NUMBERS

LMER

PC1 scores



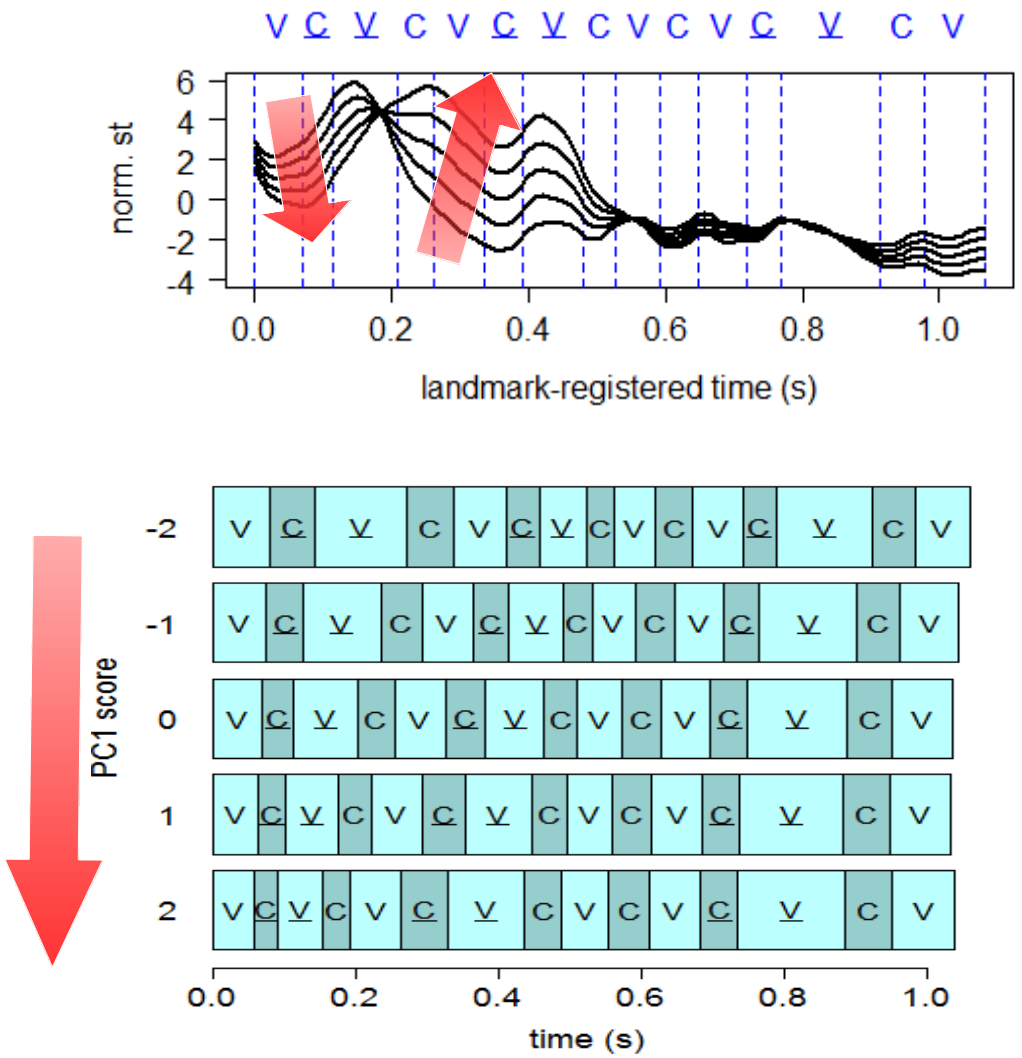


f_0

log rates

$\mu(t) + 2 * PC1(t)$
$\mu(t) + 1 * PC1(t)$
$\mu(t) + 0 * PC1(t)$
$\mu(t) - 1 * PC1(t)$
$\mu(t) - 2 * PC1(t)$

PC1 scores

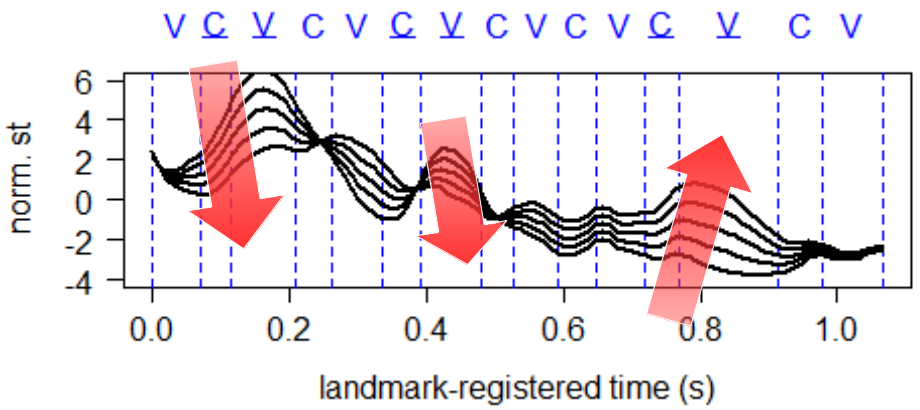


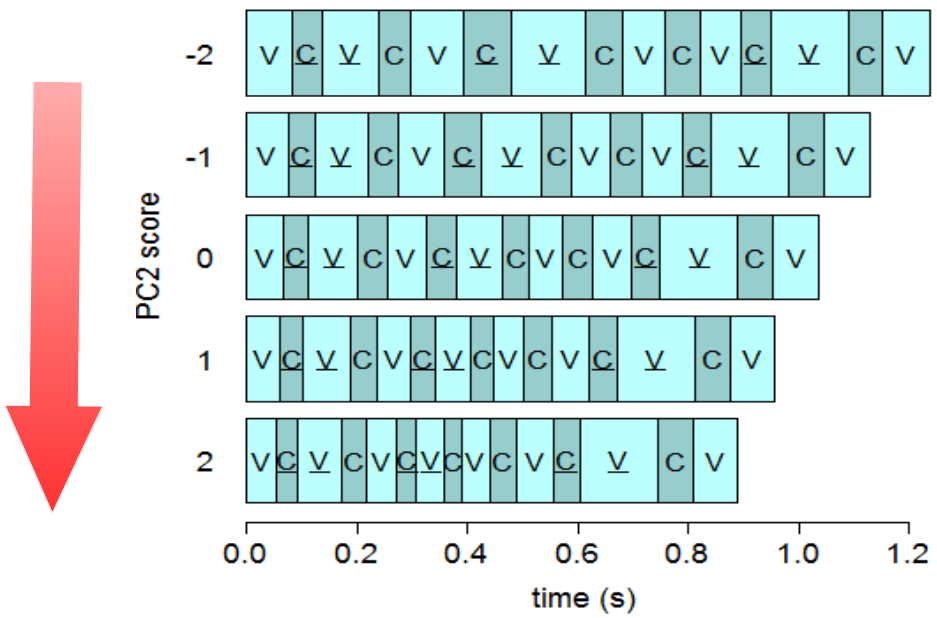
f_0

$\mu(t) + 2 * PC1(t)$
$\mu(t) + 1 * PC1(t)$
$\mu(t) + 0 * PC1(t)$
$\mu(t) - 1 * PC1(t)$
$\mu(t) - 2 * PC1(t)$

segment durations

PC2 scores





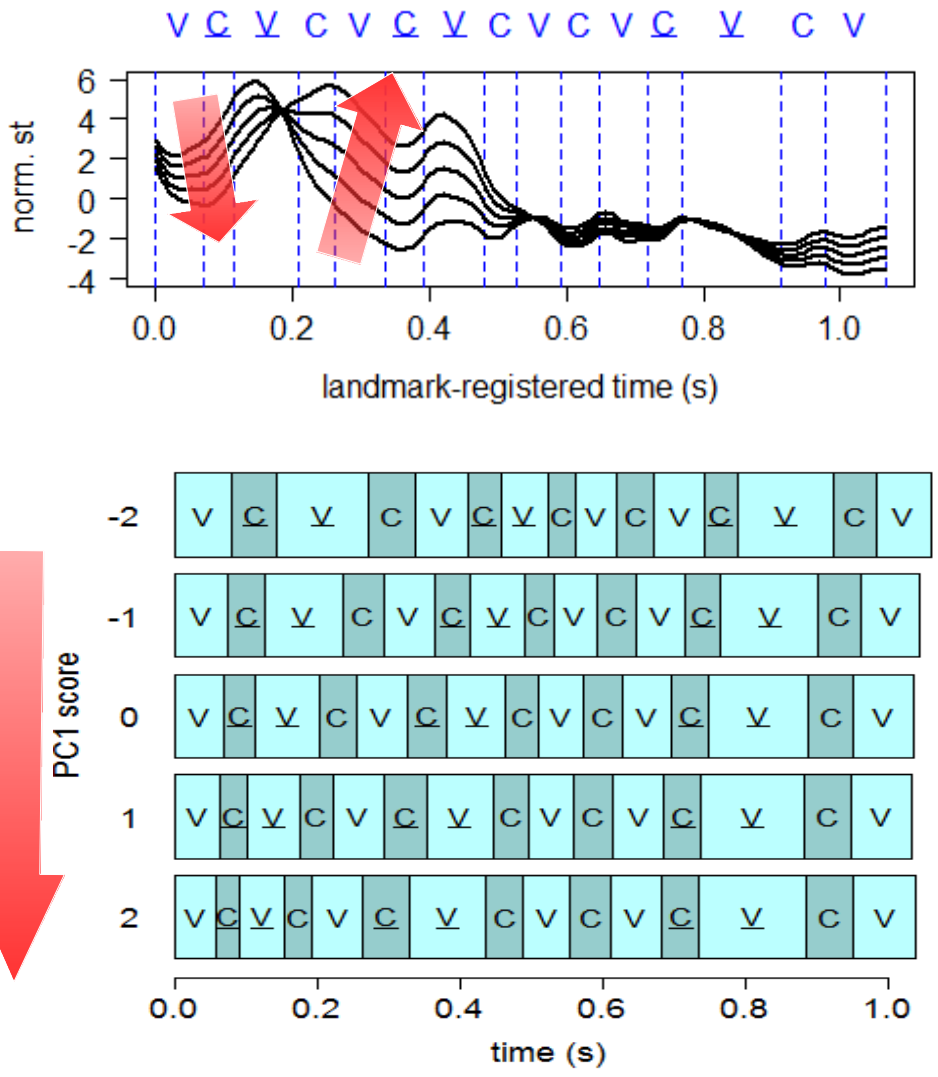
f_0

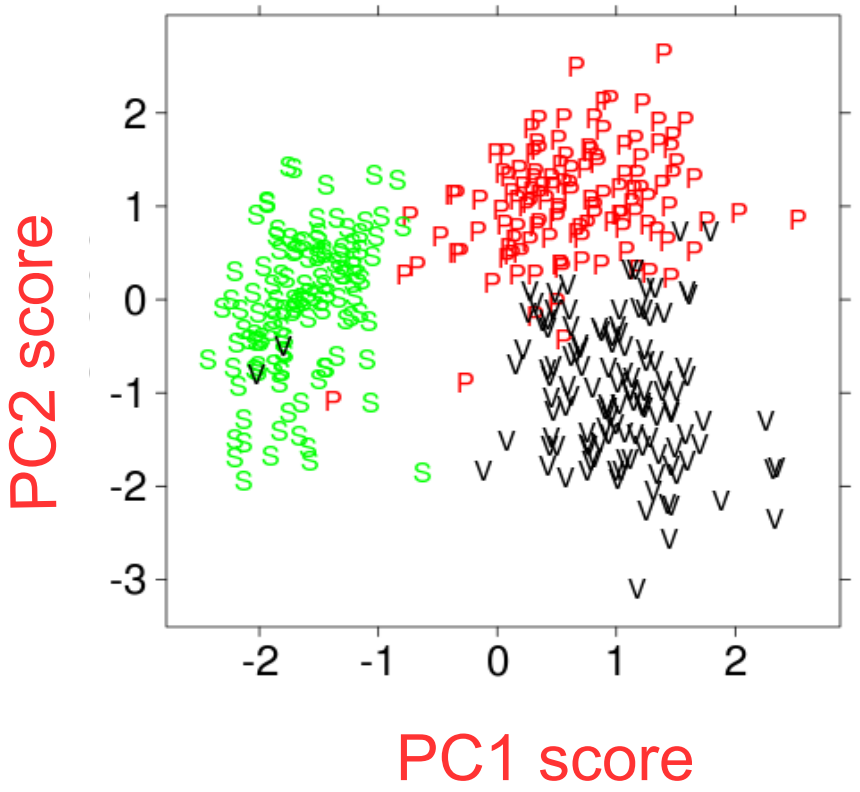
$\mu(t) + 2 * PC1(t)$
$\mu(t) + 1 * PC1(t)$
$\mu(t) + 0 * PC1(t)$
$\mu(t) - 1 * PC1(t)$
$\mu(t) - 2 * PC1(t)$

segment durations

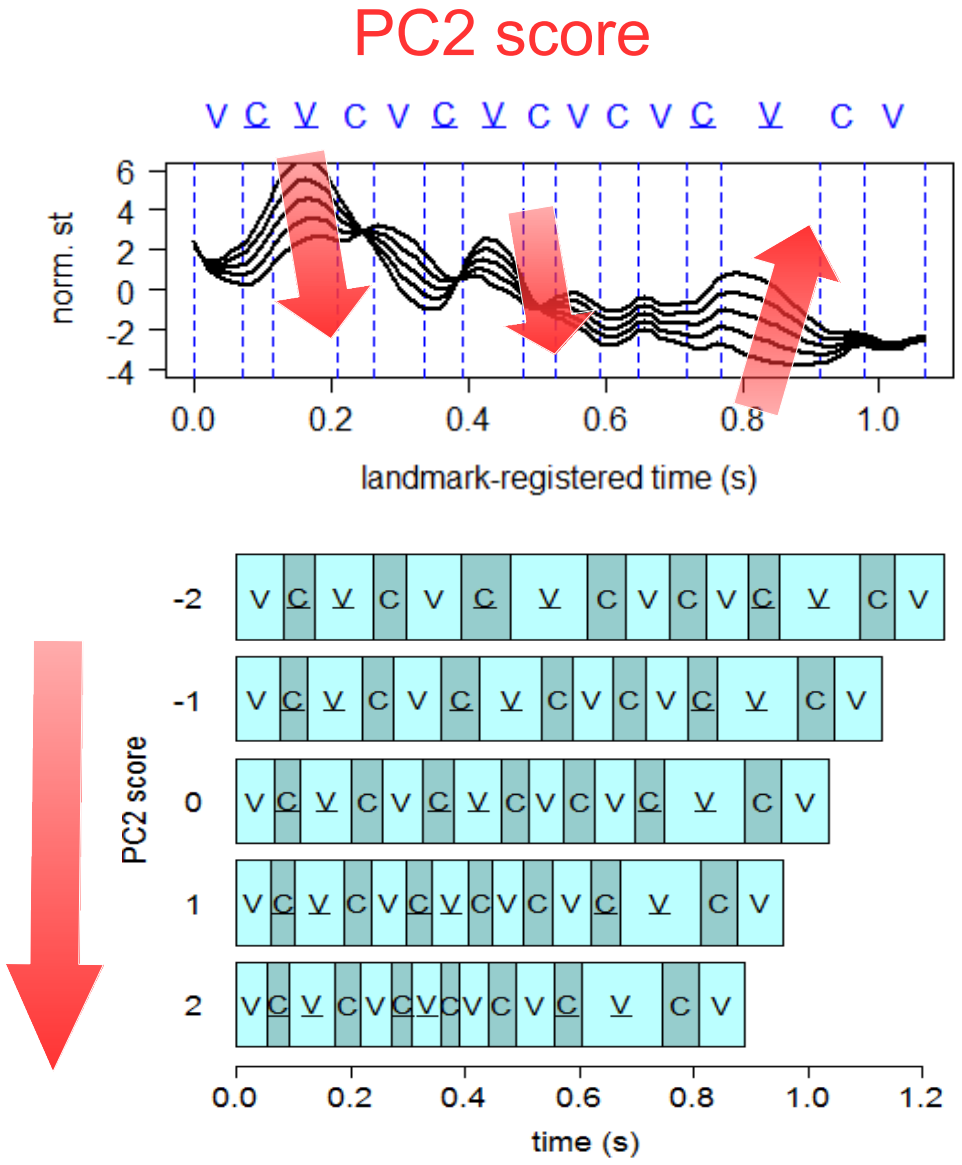
multi-segment curve parametrisation

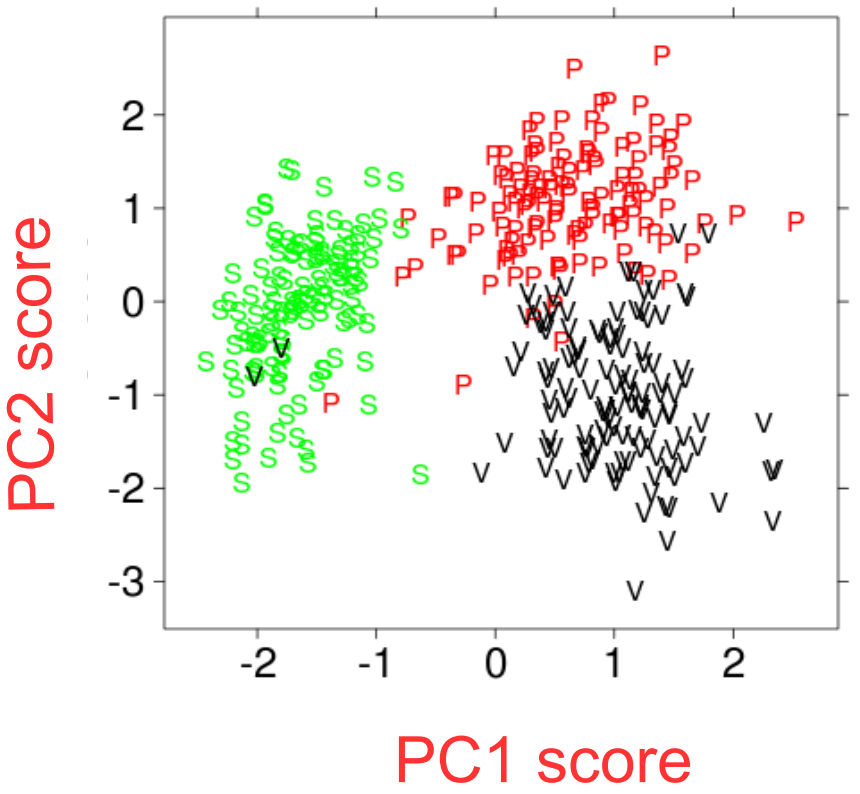
PC1 score





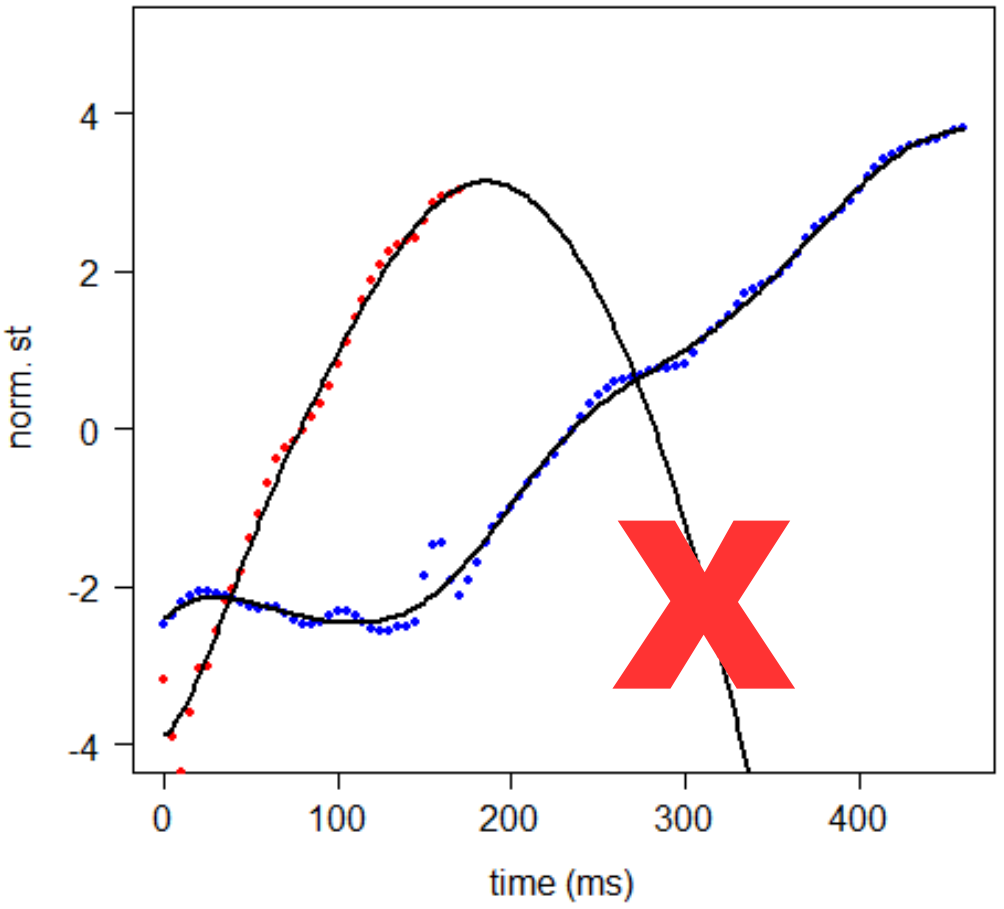
multi-segment curve parametrisation



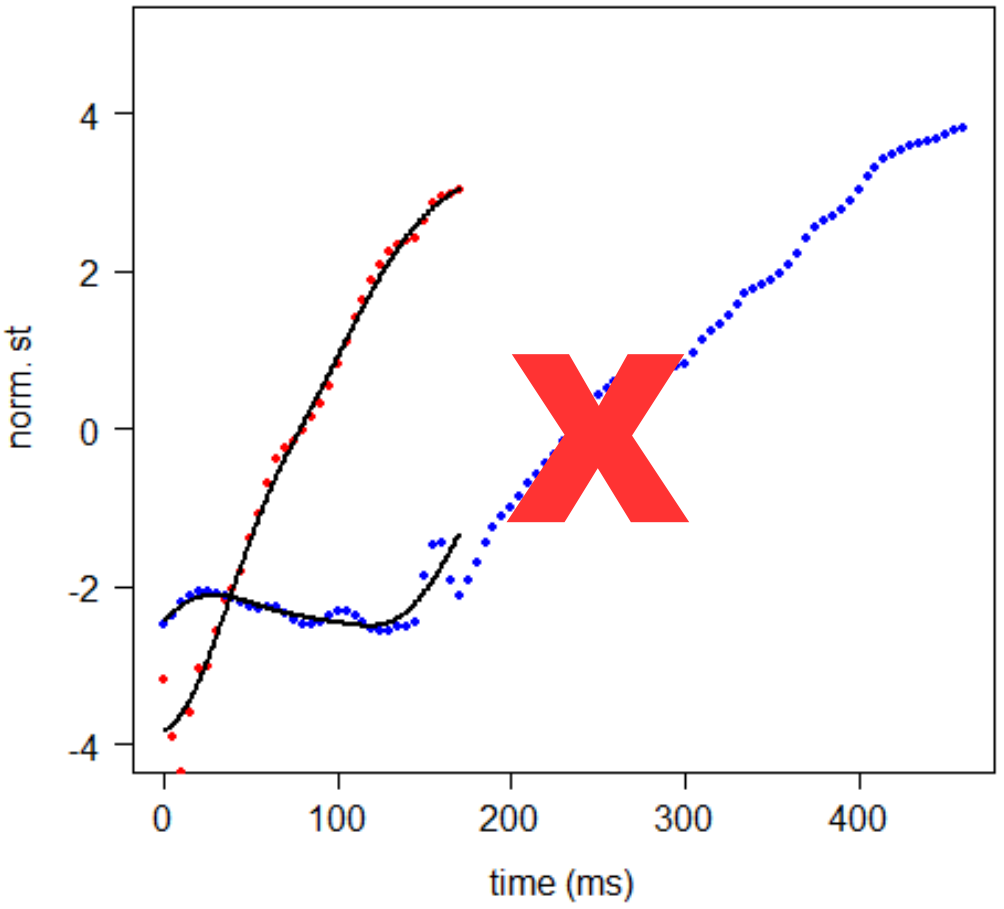


Extra slides

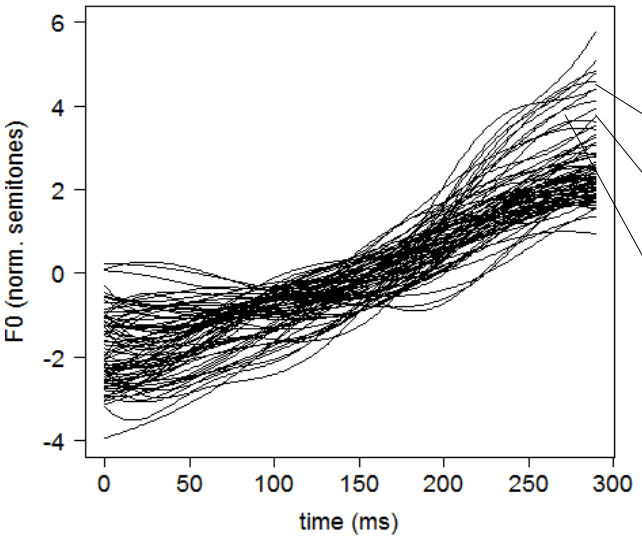
Take longest duration

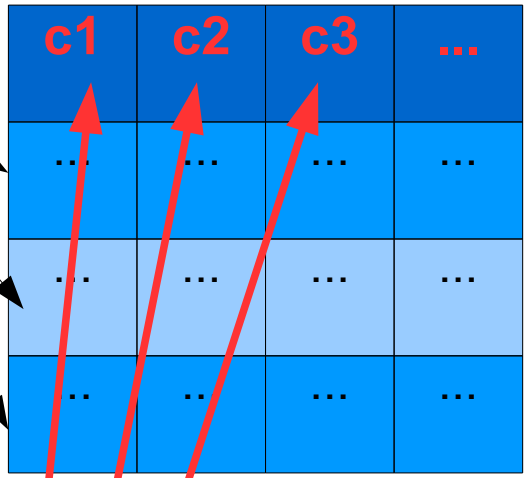


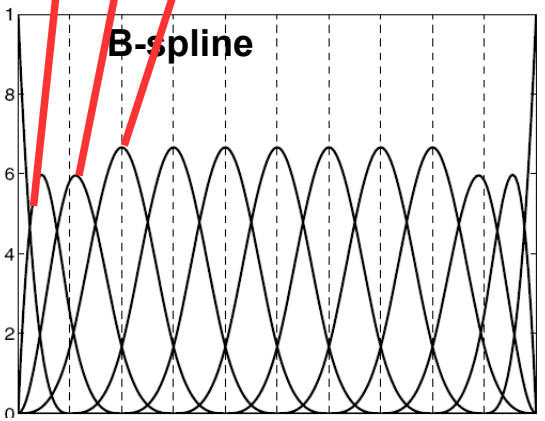
Take shortest duration



Principal Component Analysis





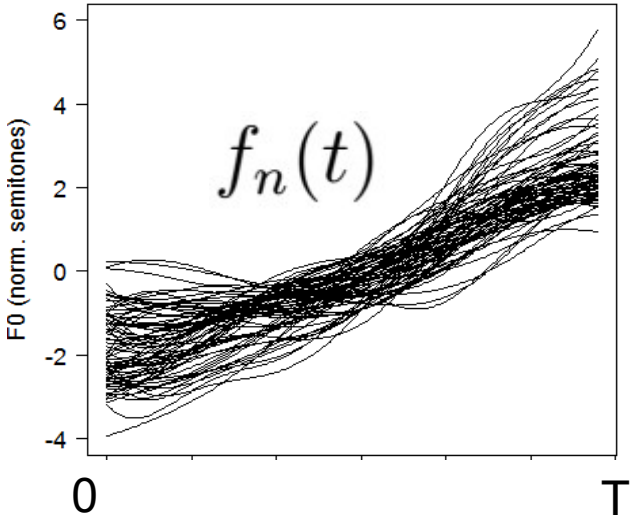


PCA

PCA limitations

- PCA does not use any explicit information related to the curve shapes or the B-splines shapes
- e.g. the sequence of coefficients c1, c2,.. reflects time adjacency of polynomial components, i.e. overlapping 'hills'

Functional PCA



$$\max\left\{var_n\left(\int_0^T PC1(t)f_n(t)dt\right)\right\}$$

$$\text{subject to}\int_0^T PC1^2(t)=1$$

- FPCA definition uses the input curves $f_n(t)$
- FPCA is independent of the B-splines used to smooth $f_n(t)$