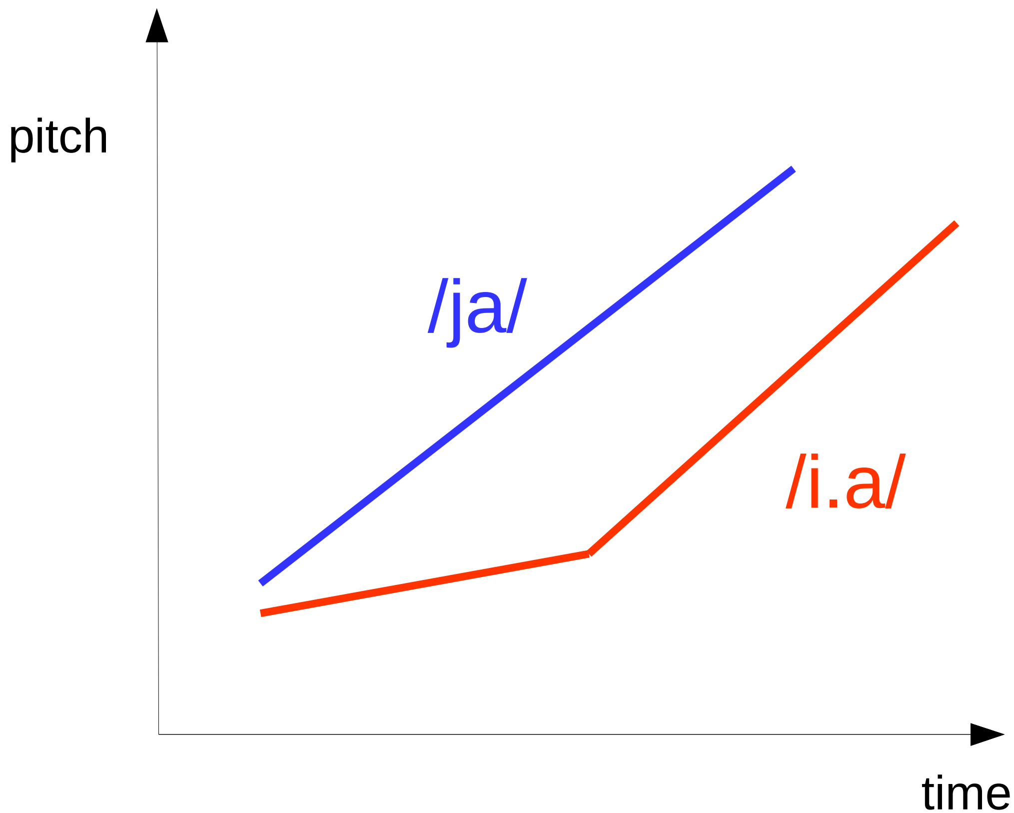
Functional Principal Component Analysis (FPCA) for Phonetic Research

Michele Gubian

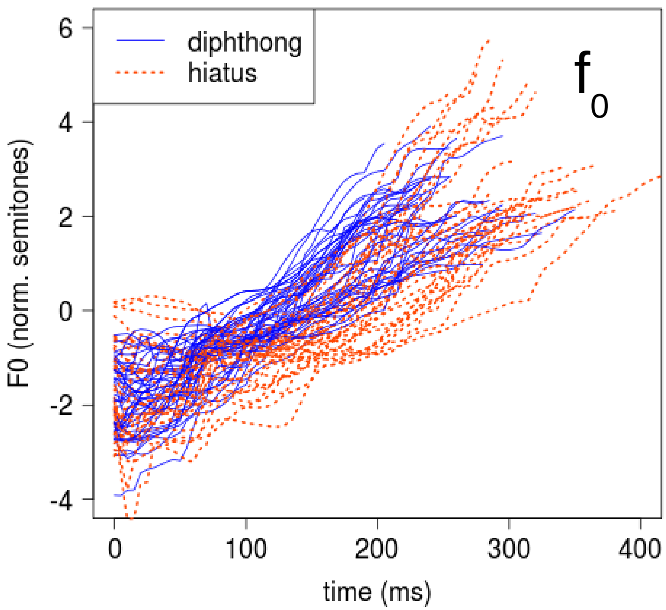
LMU Munich, Germany

November 15th, 2018

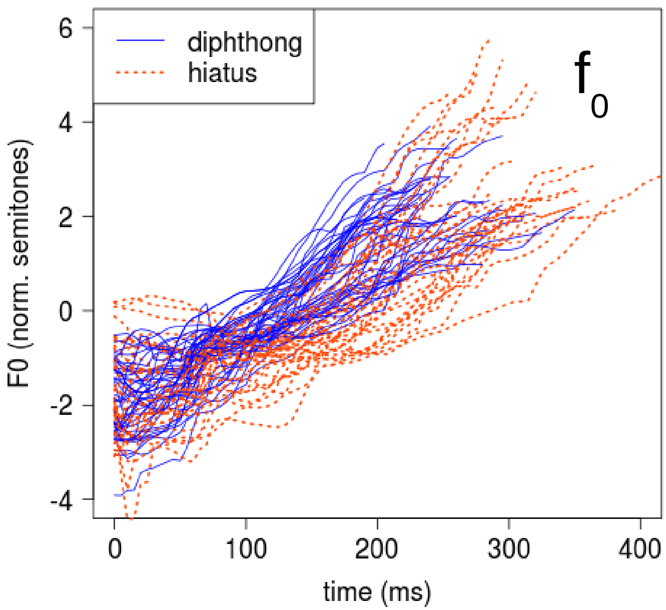
Alignment of rising pitch accents in Spanish



- European Spanish
- **Diphthong**: /ja/
  e.g. *Emil<u>ia</u>na*
- **Hiatus** /i.a/
  e.g. *p<u>ia</u>no*
- Rising pitch accent should align to syllabic structure



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



ANOVA

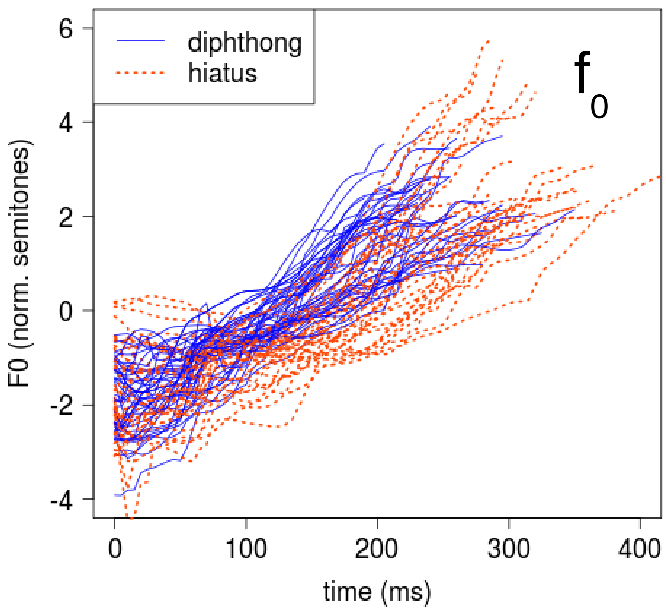
LR

LMER

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

CURVES

NUMBERS



MIND THE GAP

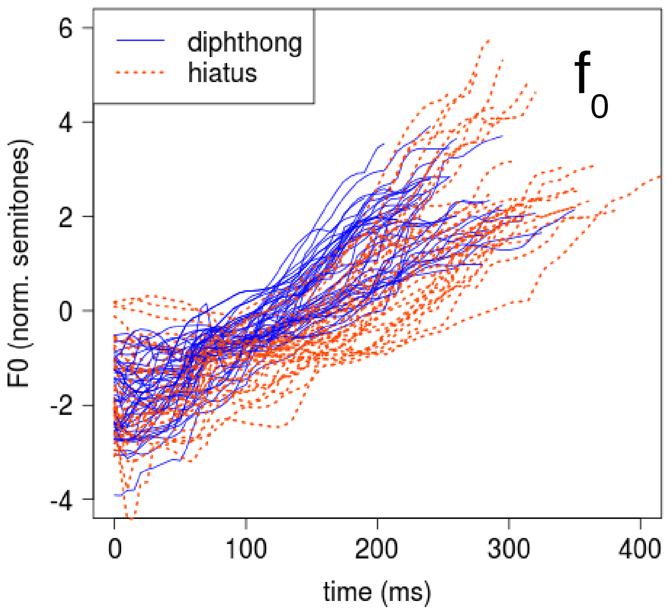
ANOVA

LR

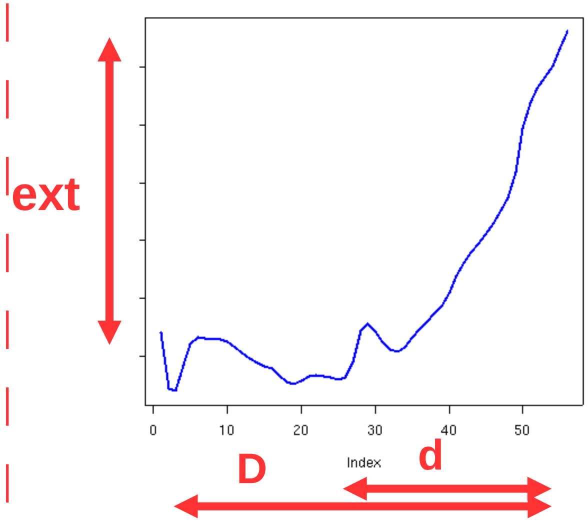
LMER

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

CURVES



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

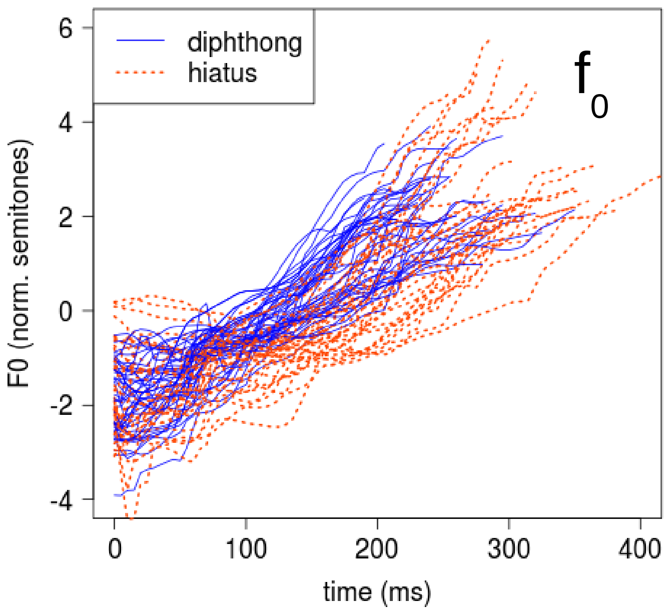


ext (st)	d/D	Cat.
5.3	0.9	D
4.6	0.7	H
...	...	...

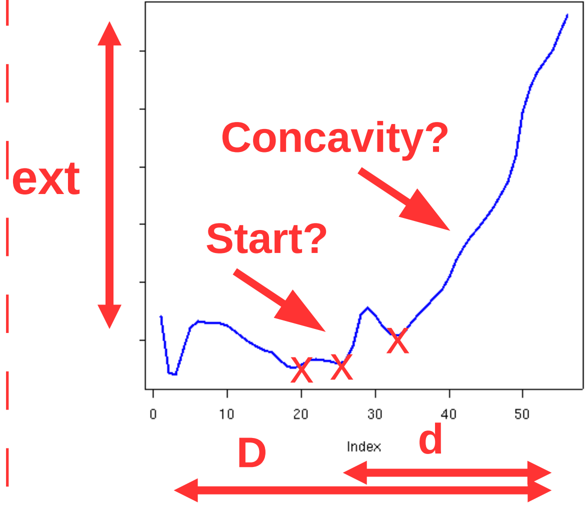
NUMBERS

- ANOVA
- LR
- LMER

CURVES



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

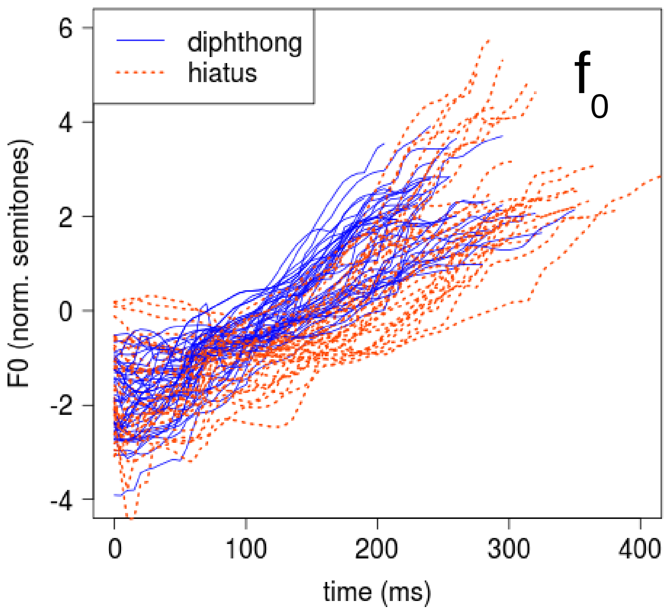


ext (st)	d/D	Cat.
5.3	0.9	D
4.6	0.7	H
...	...	...

NUMBERS

- ANOVA
- LR
- LMER

CURVES



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

DCT

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

NUMBERS

ANOVA

LR

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

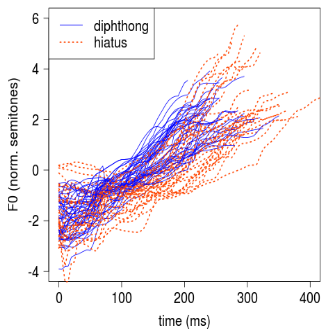
MISSION

automate curve parametrisation

- Data driven
- Few parameters
- Interpretable

Road map

CURVES



Interpolate using a function basis

- Data driven

Dimensionality reduction tool

- Few parameters
- Interpretable

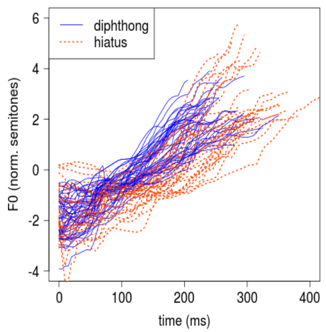
NUMBERS

- ANOVA
- LM
- LMER

Road map

CURVES

NUMBERS



Interpolate using a function basis

Dimensionality reduction tool

ANOVA

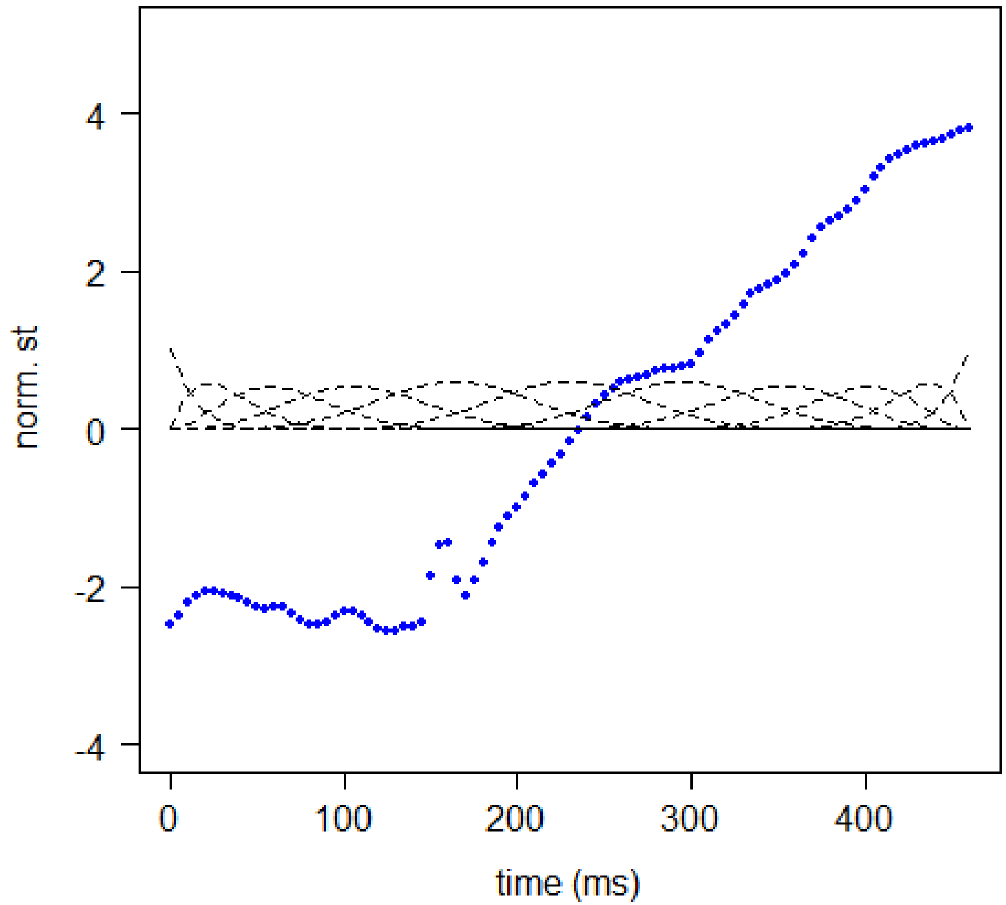
LM

LMER

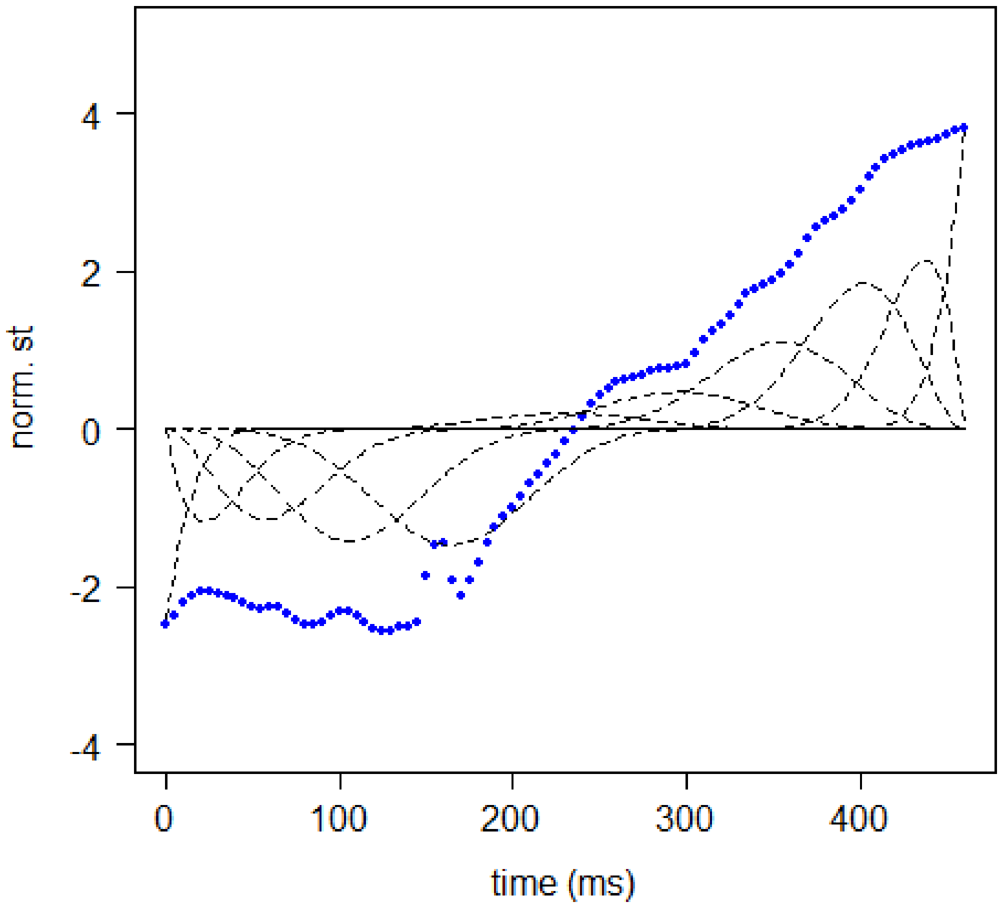
- Data driven

- Few parameters
- Interpretable

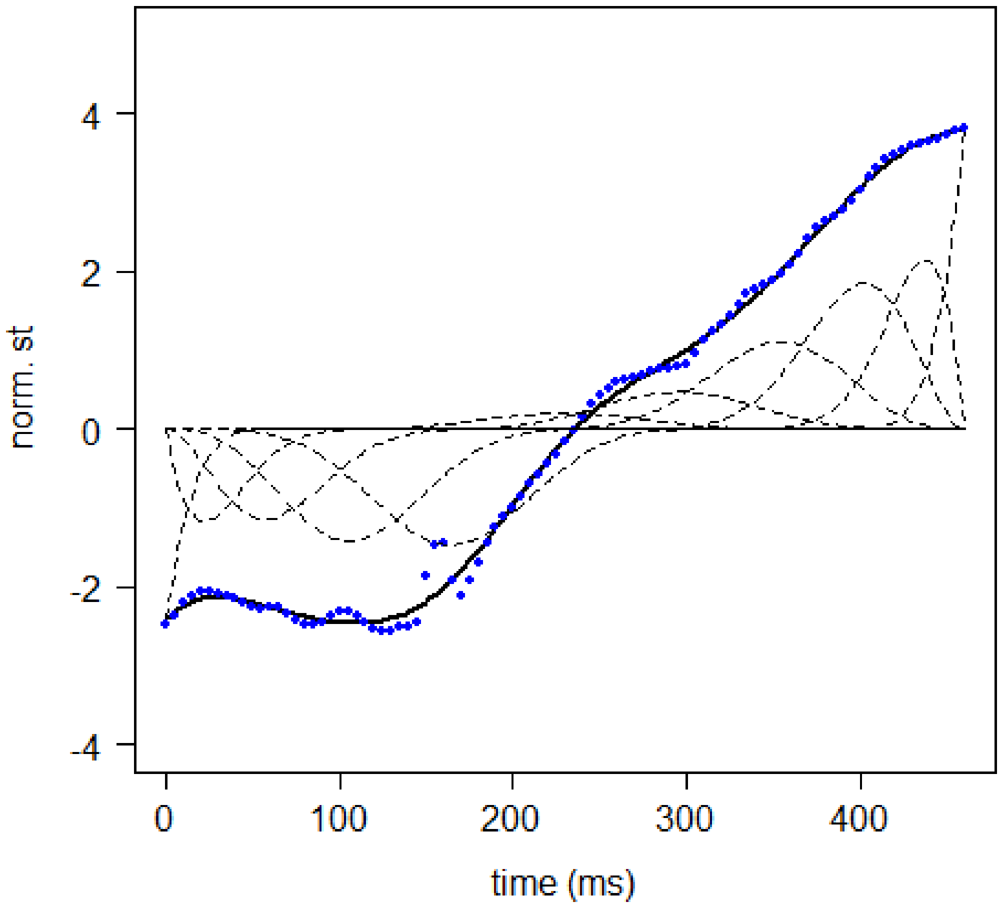
Interpolation with B-splines



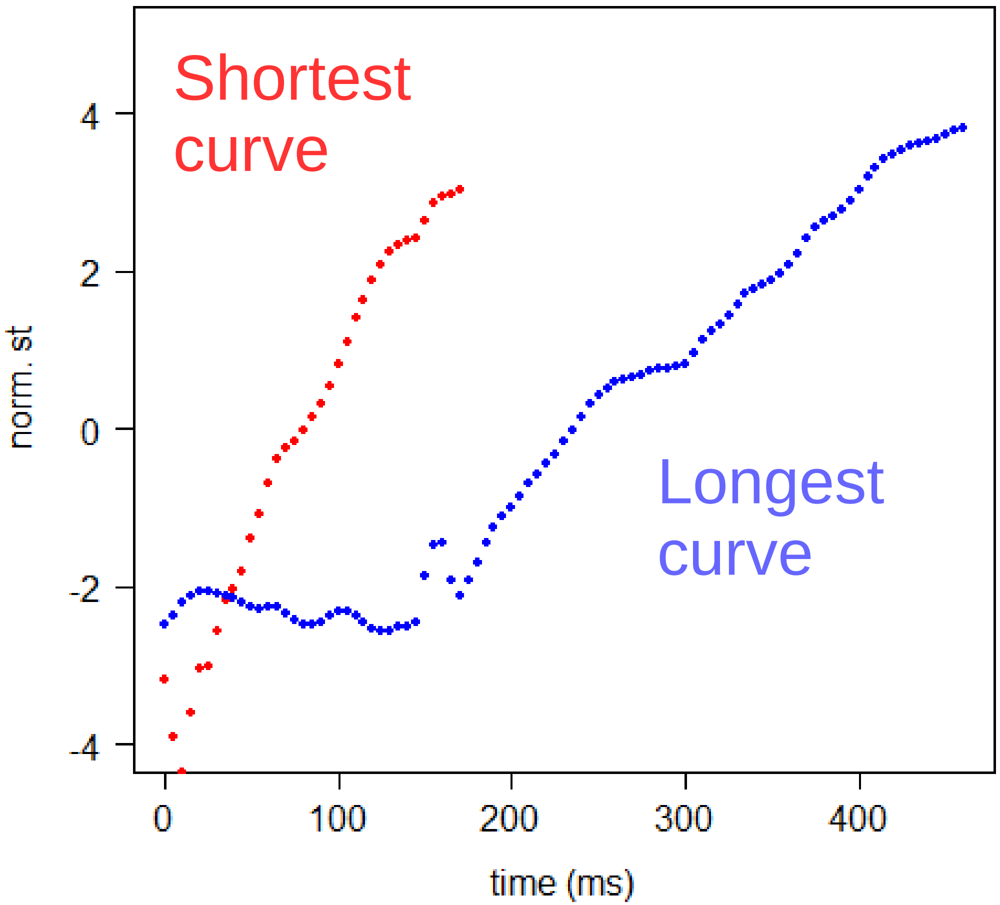
Interpolation with B-splines



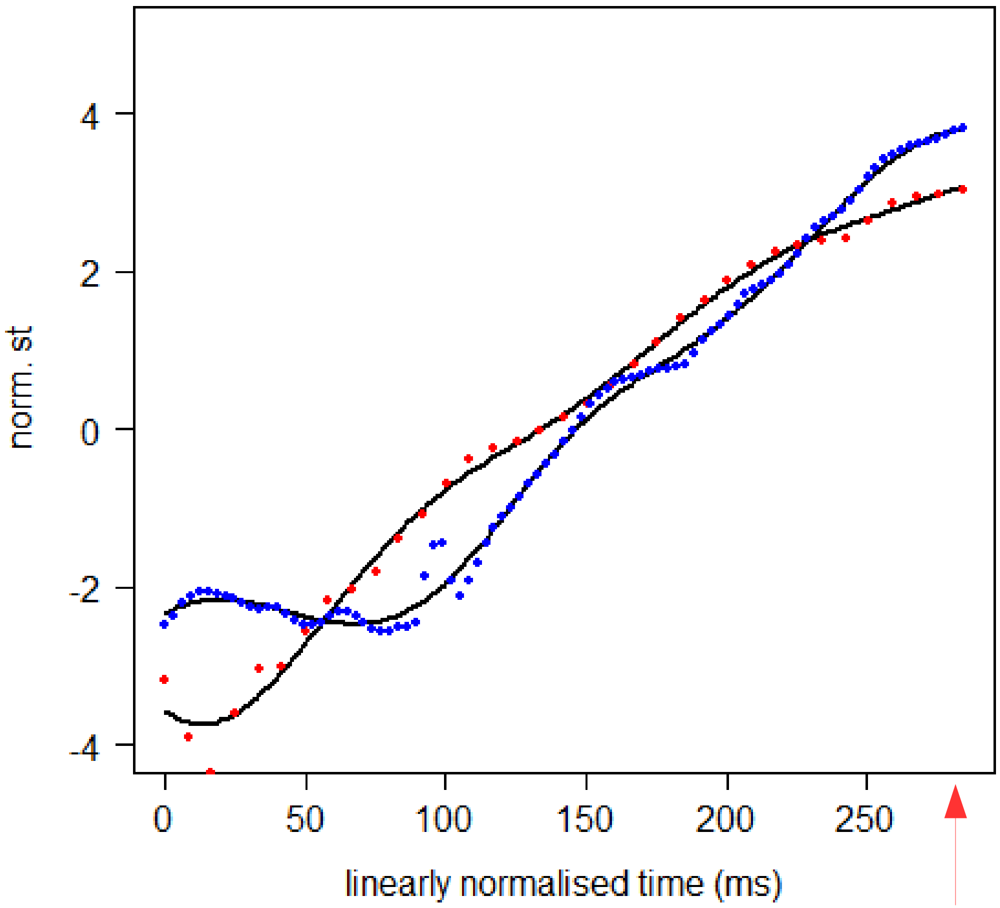
Interpolation with B-splines



Different durations

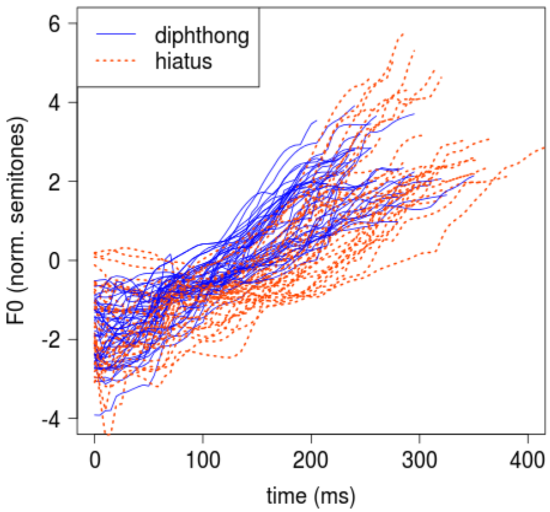


Linear time normalisation

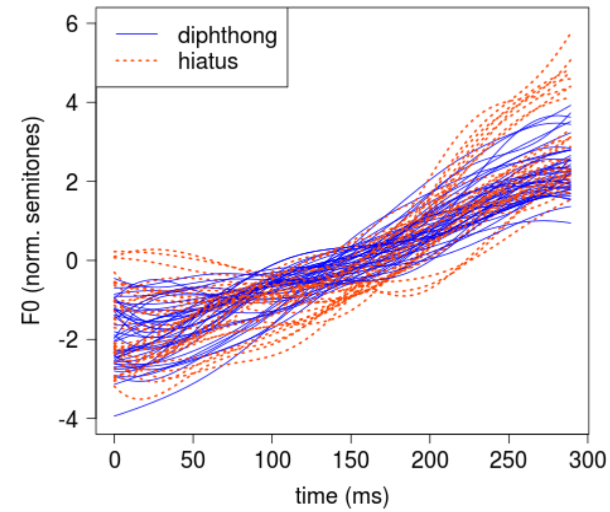


average duration

Linear time normalisation



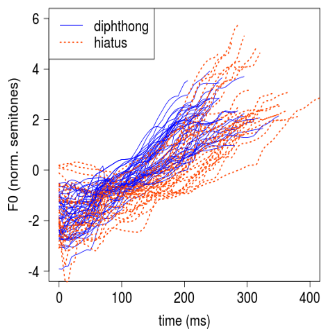
Interpolate to
<u>the same</u>
time interval



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

Road map

CURVES



Interpolate to
the same
time interval

- Data driven

Dimensionality
reduction tool

- Few parameters
- Interpretable

NUMBERS

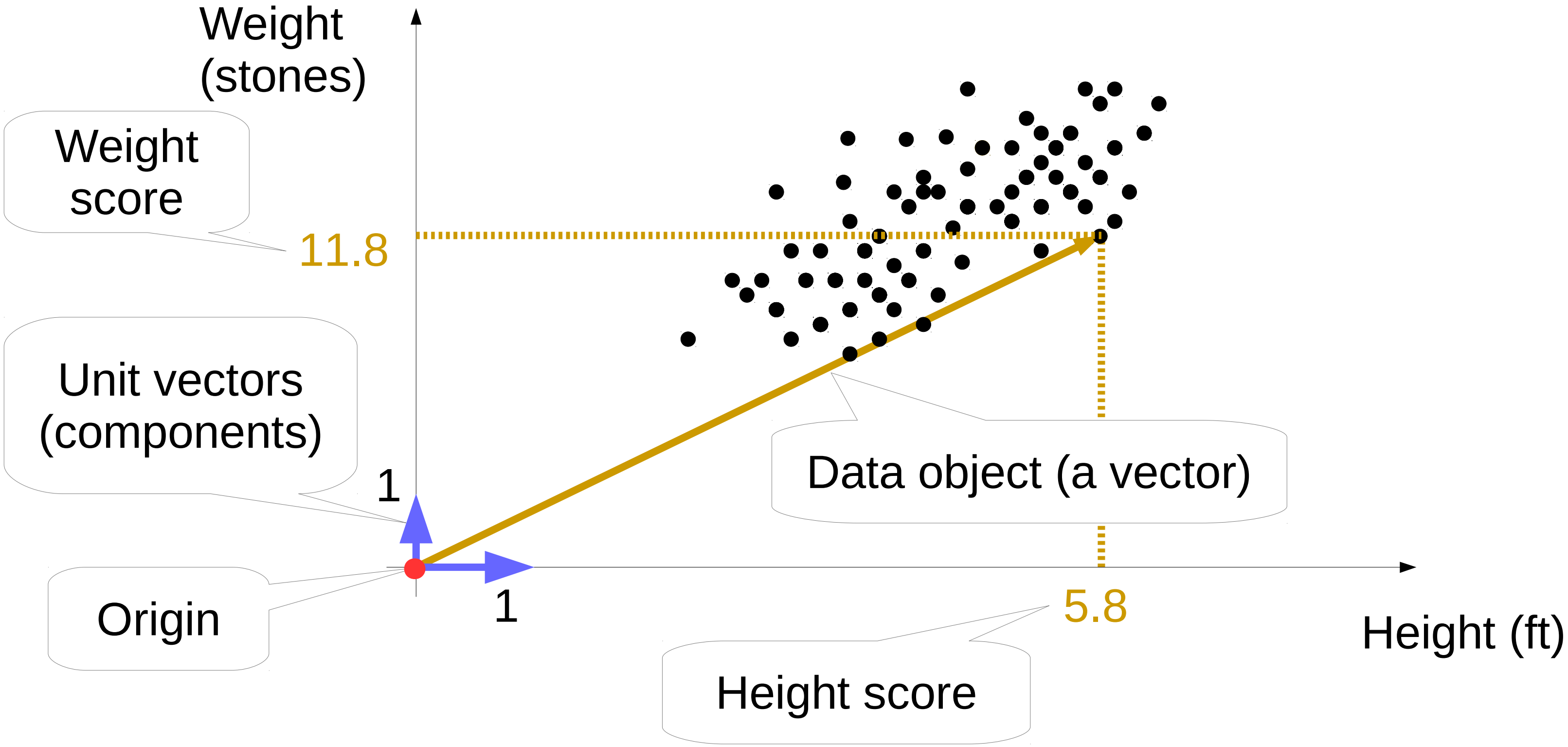
ANOVA

LM

LMER

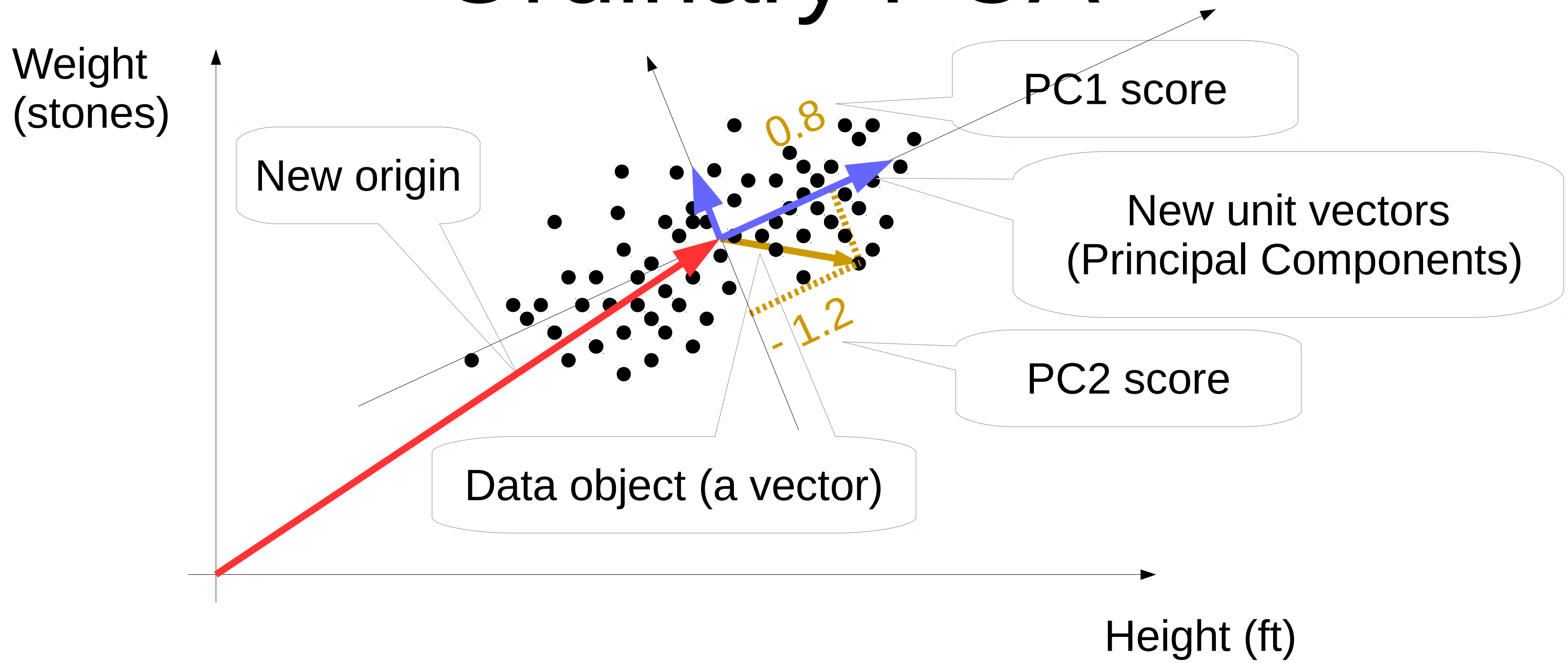
Introducing
Functional PCA

Vectors



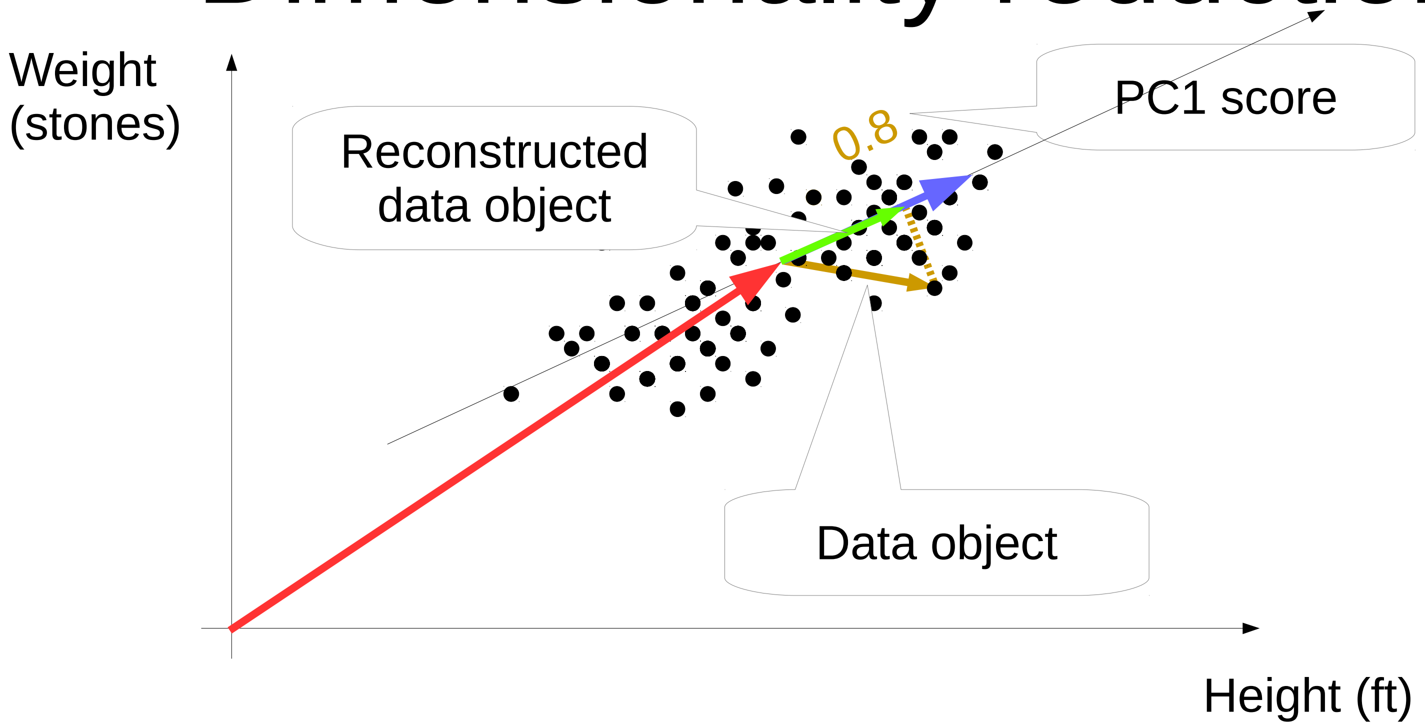
- Data objects and components are vectors
- From scores (numbers) we can reconstruct data objects (vectors)

Ordinary PCA

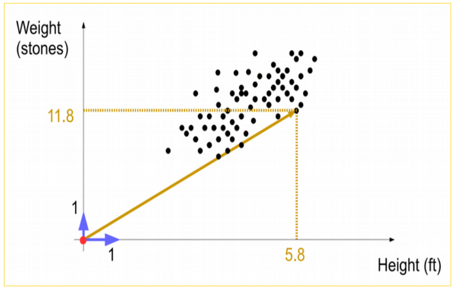


- PCA computes new origin and unit vectors which best suit the data
- From PC scores we can reconstruct data objects

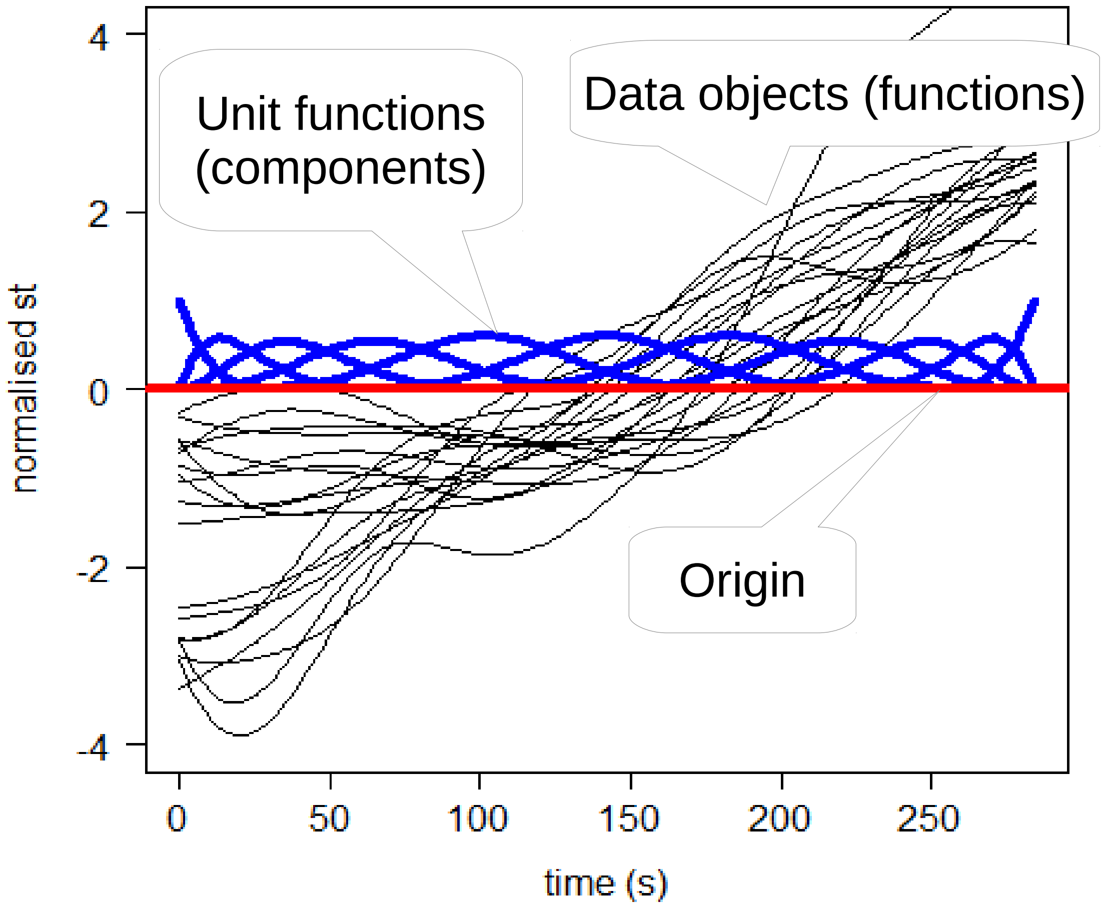
Dimensionality reduction



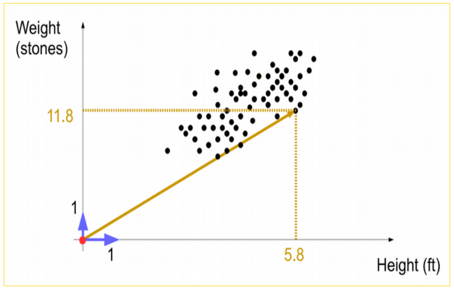
- We can use only part of the PCs
- This reduces the data dimensionality
- But introduces reconstruction errors too



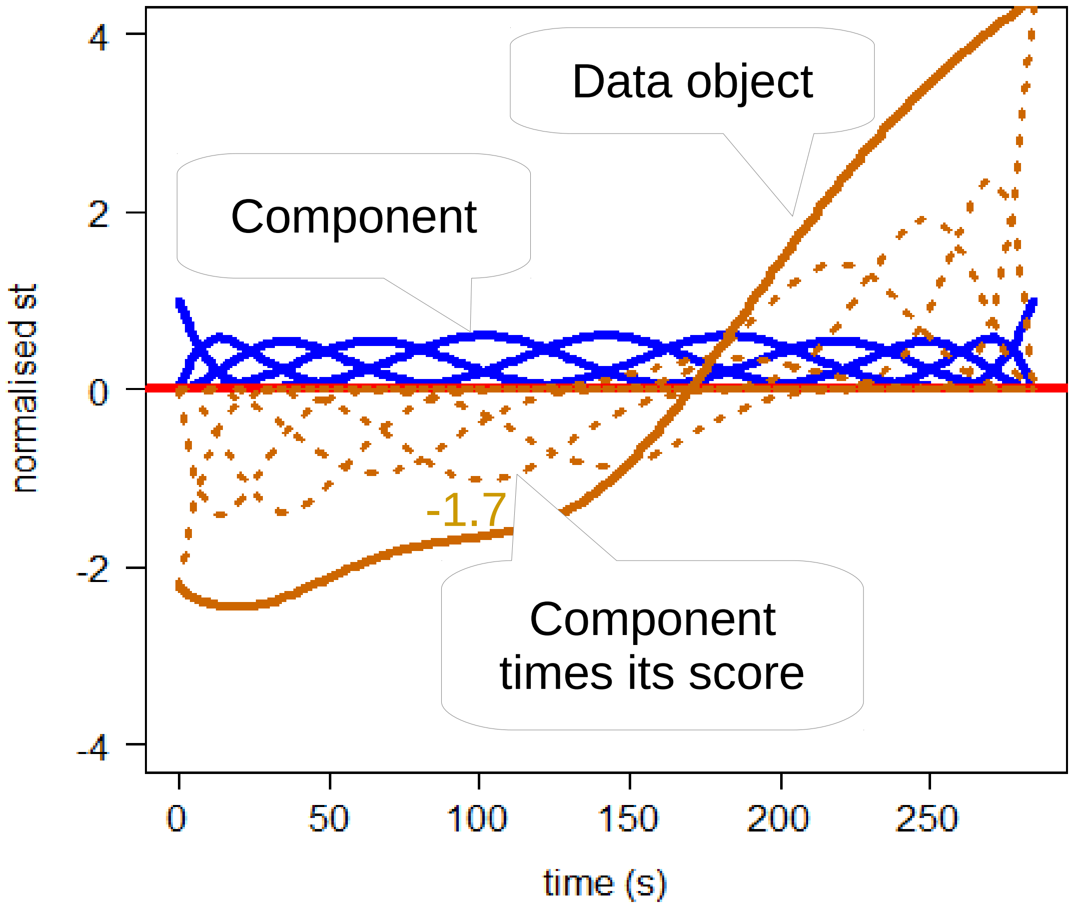
Functions (curves)



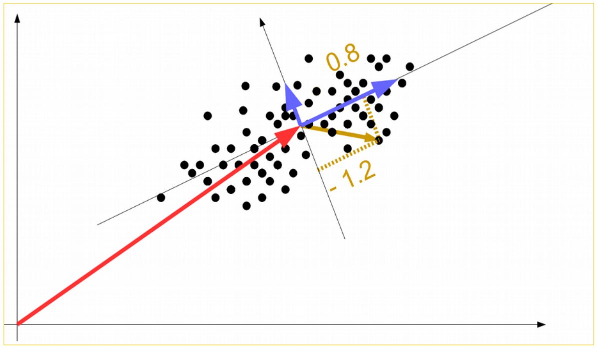
- Origin, components and data objects are functions
- Origin is a flat line
- Components are 11 B-spline curves



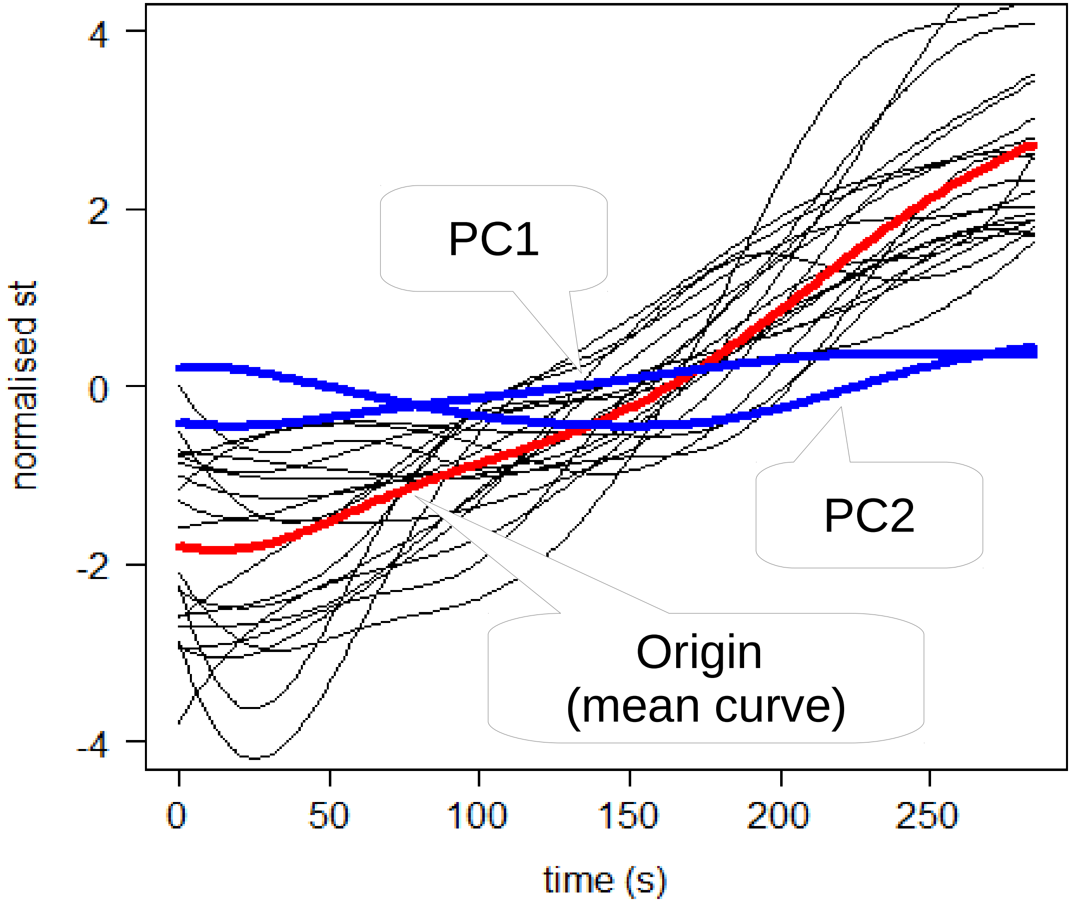
Functions (curves)



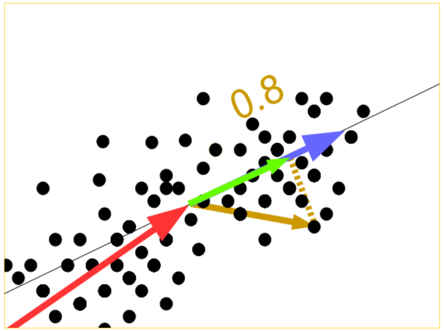
- Each of the 11 components is multiplied by a score
- These are summed together to obtain a data object



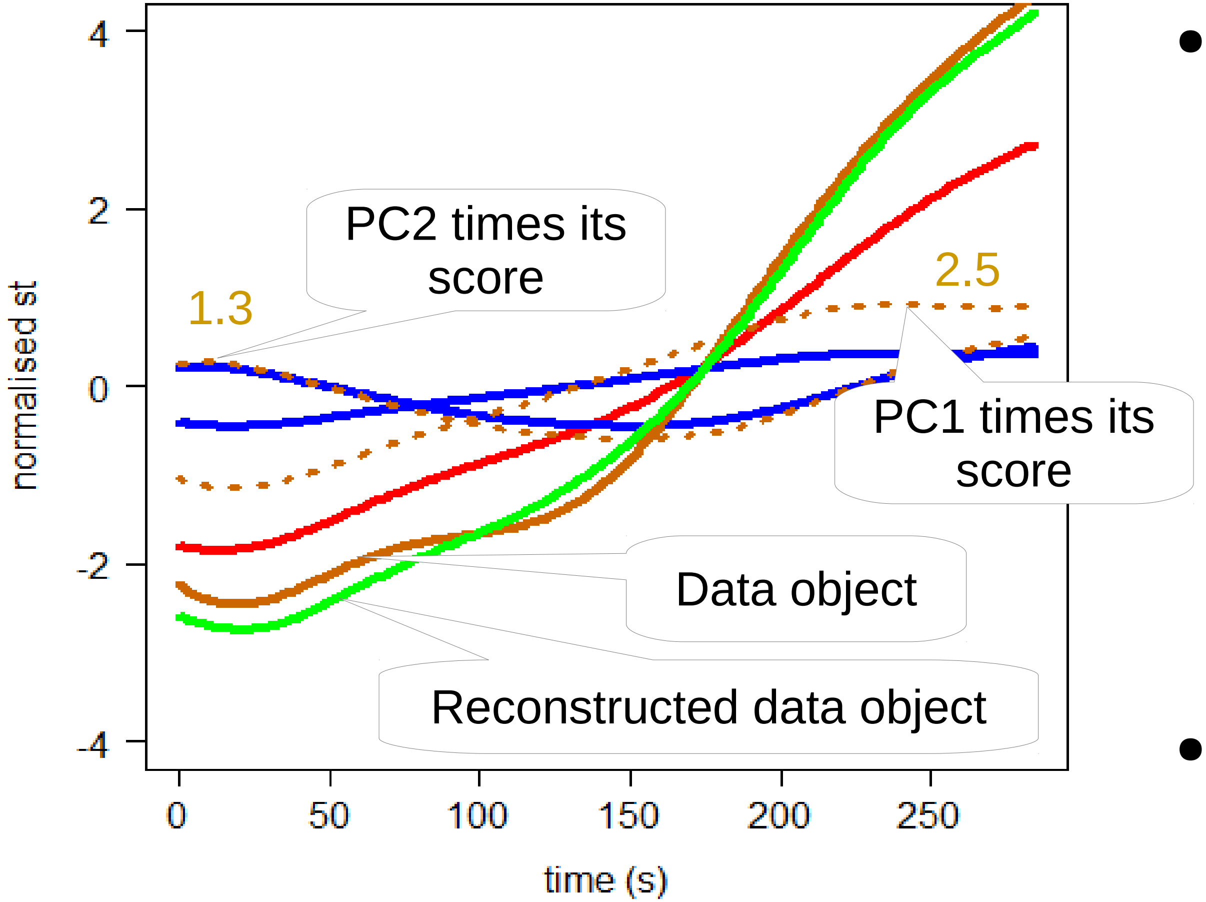
Functional PCA



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



Functional PCA



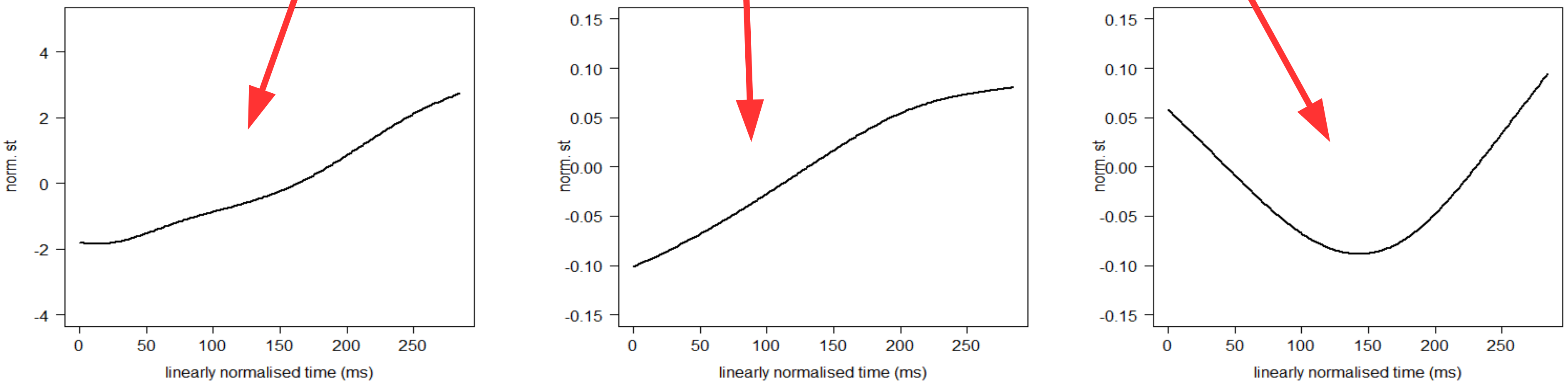
- The sum of origin (mean) curve + PCs times their scores gives an approx reconstruction of the original curve
- Dimensions from 11 (B-splines) down to 2 (PCs)

Functional PCs

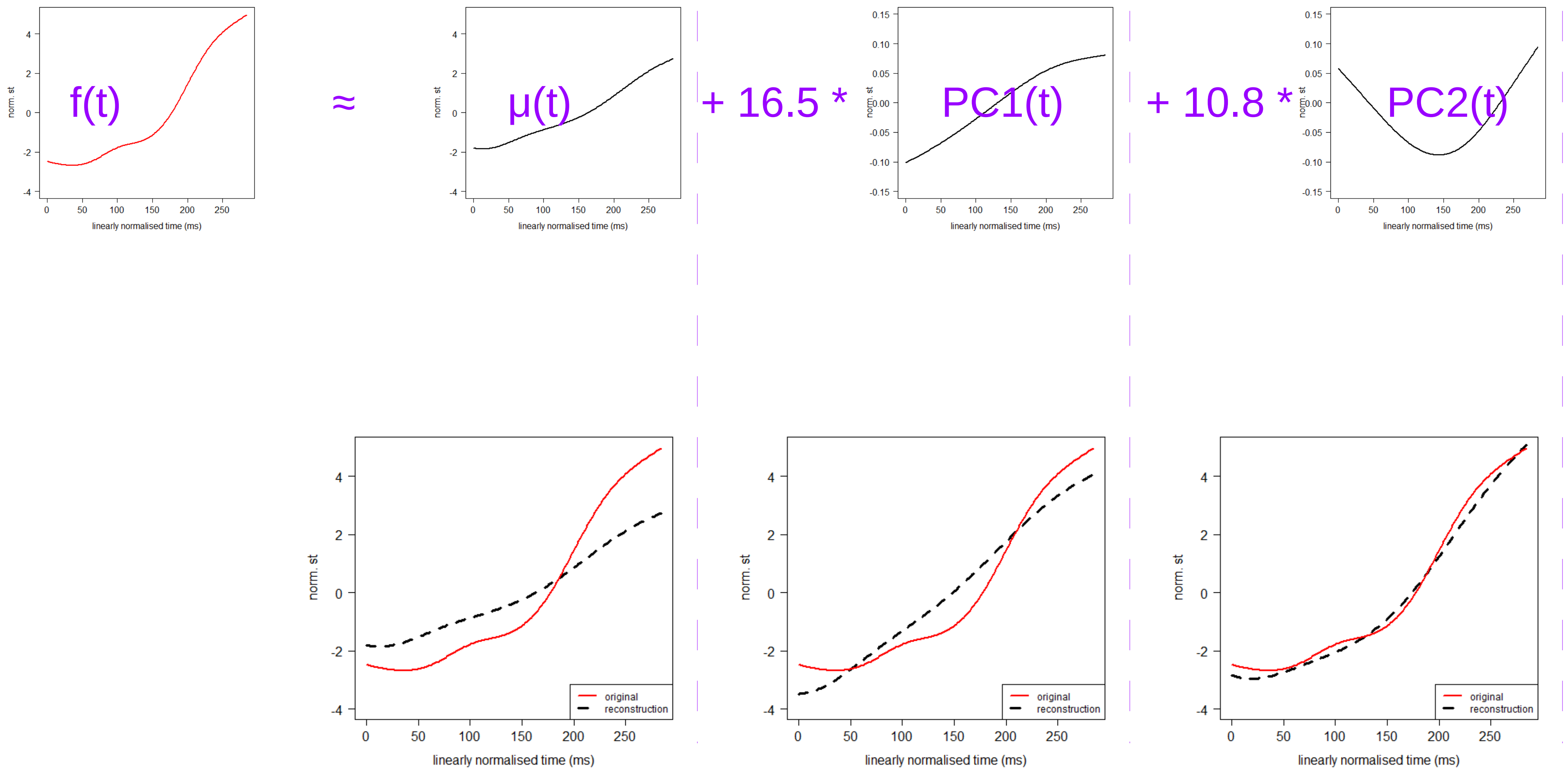
$$\boxed{f(t) \approx \mu(t) + s_1 \cdot PC1(t) + s_2 \cdot PC2(t) + \cdots}$$

PC1 score

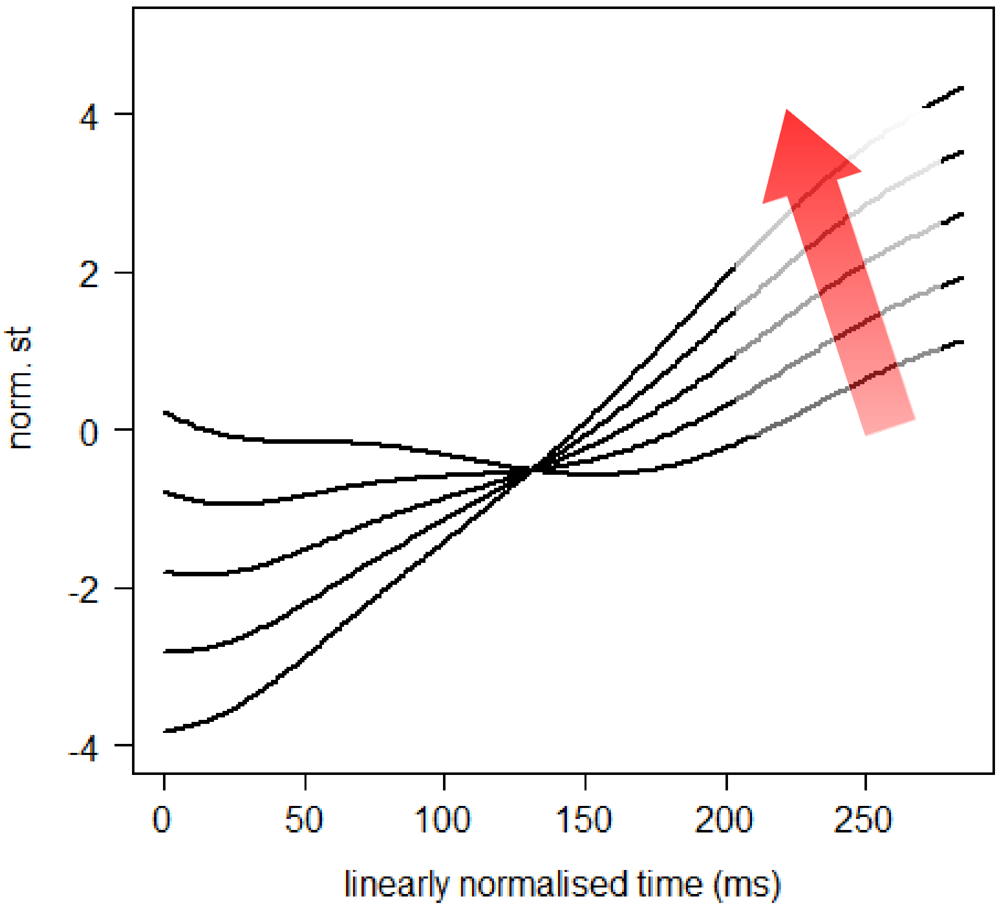
PC2 score



Curve reconstruction

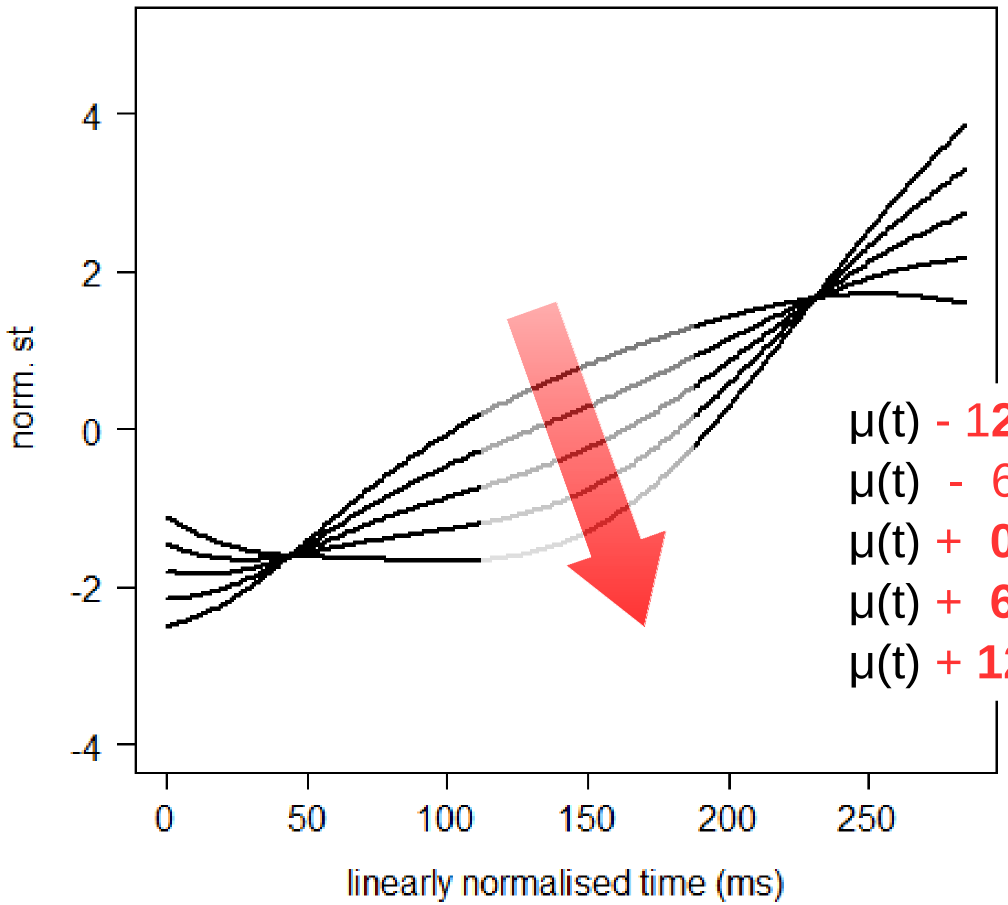


PC1 scores



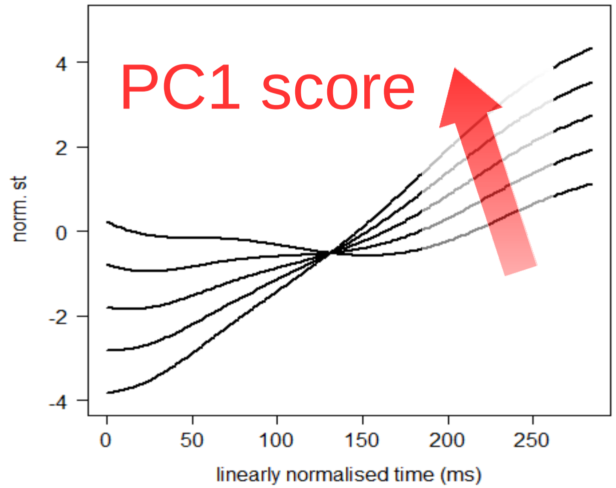
μ(t) + **20** * PC1(t)
μ(t) + **10** * PC1(t)
μ(t) + **0** * PC1(t)
μ(t) - **10** * PC1(t)
μ(t) - **20** * PC1(t)

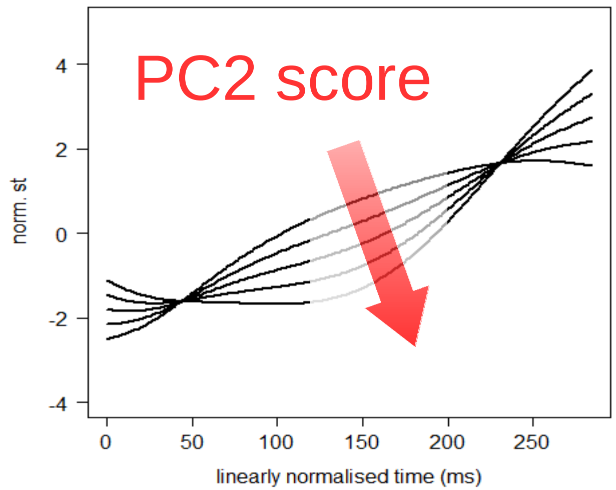
PC2 scores

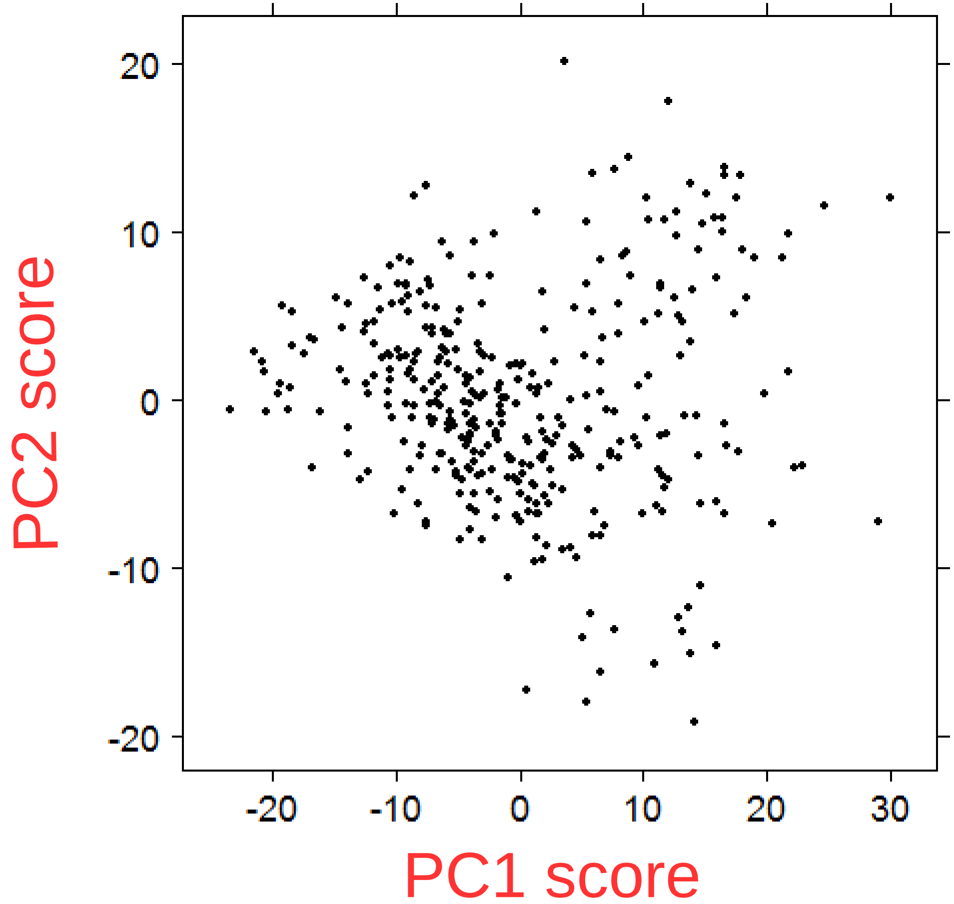


$\mu(t) - 12 * PC2(t)$
$\mu(t) - 6 * PC2(t)$
$\mu(t) + 0 * PC2(t)$
$\mu(t) + 6 * PC2(t)$
$\mu(t) + 12 * PC2(t)$

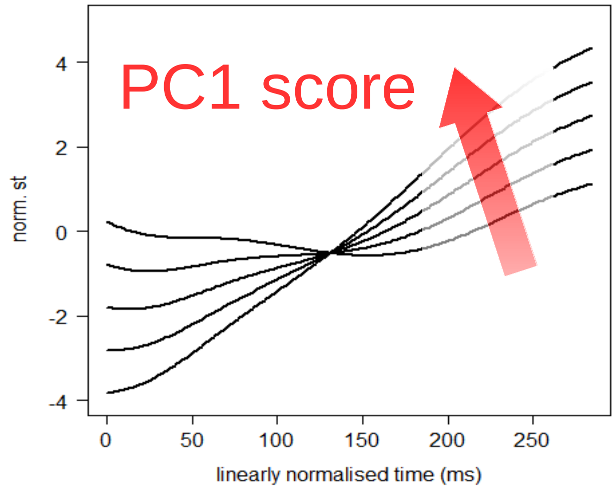
Curve parametrisation

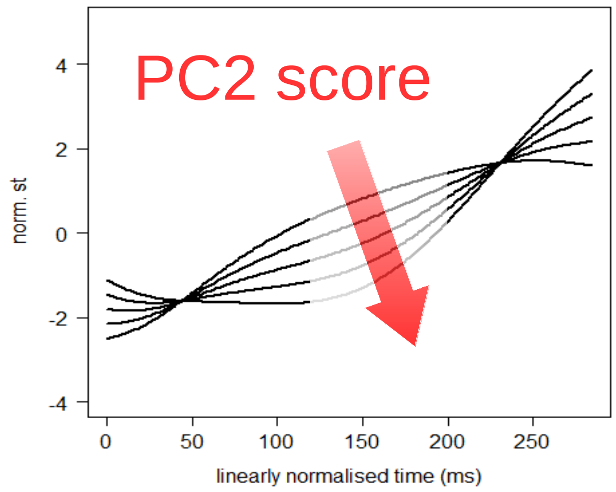


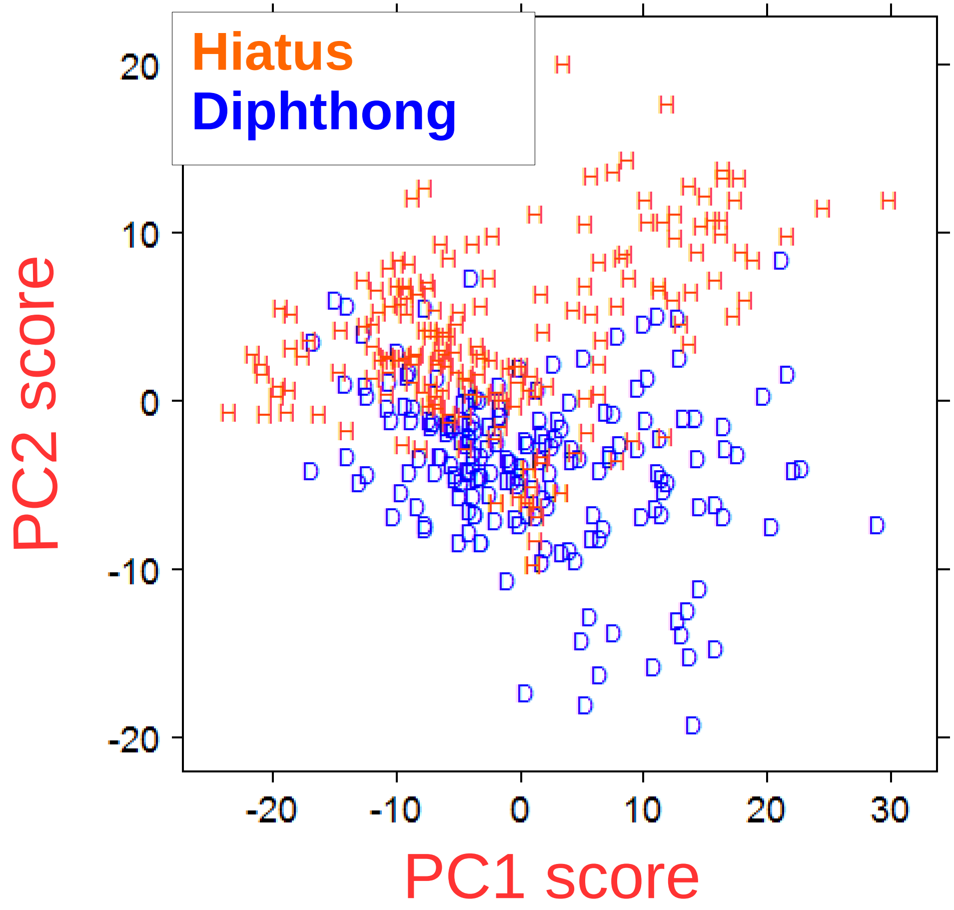




Curve parametrisation





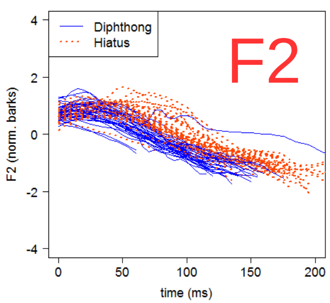


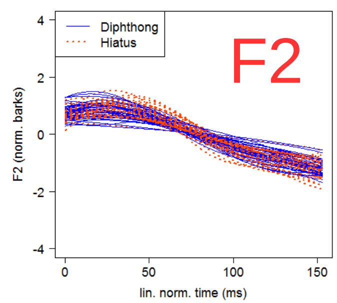
Multidimensional signals

Formants

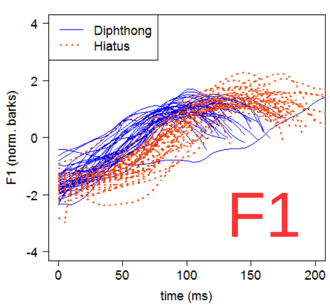
2D CURVES

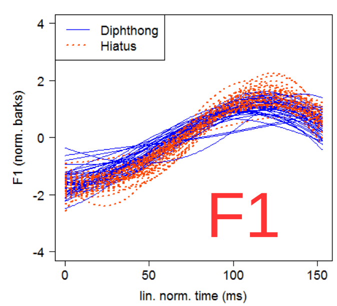
NUMBERS





FPCA





FPCA

ANOVA

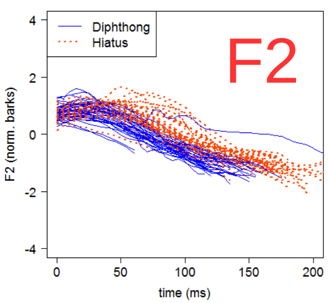
LM

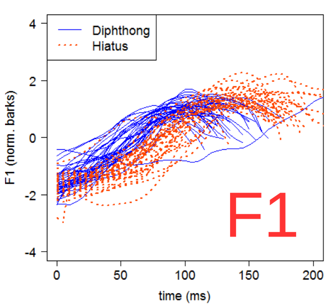
LMER

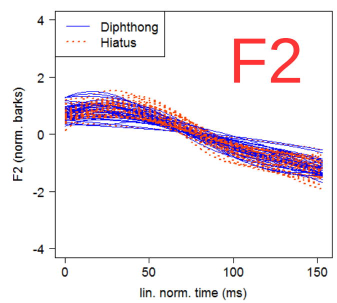
Formants

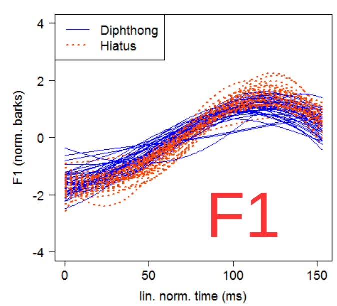
2D CURVES

NUMBERS









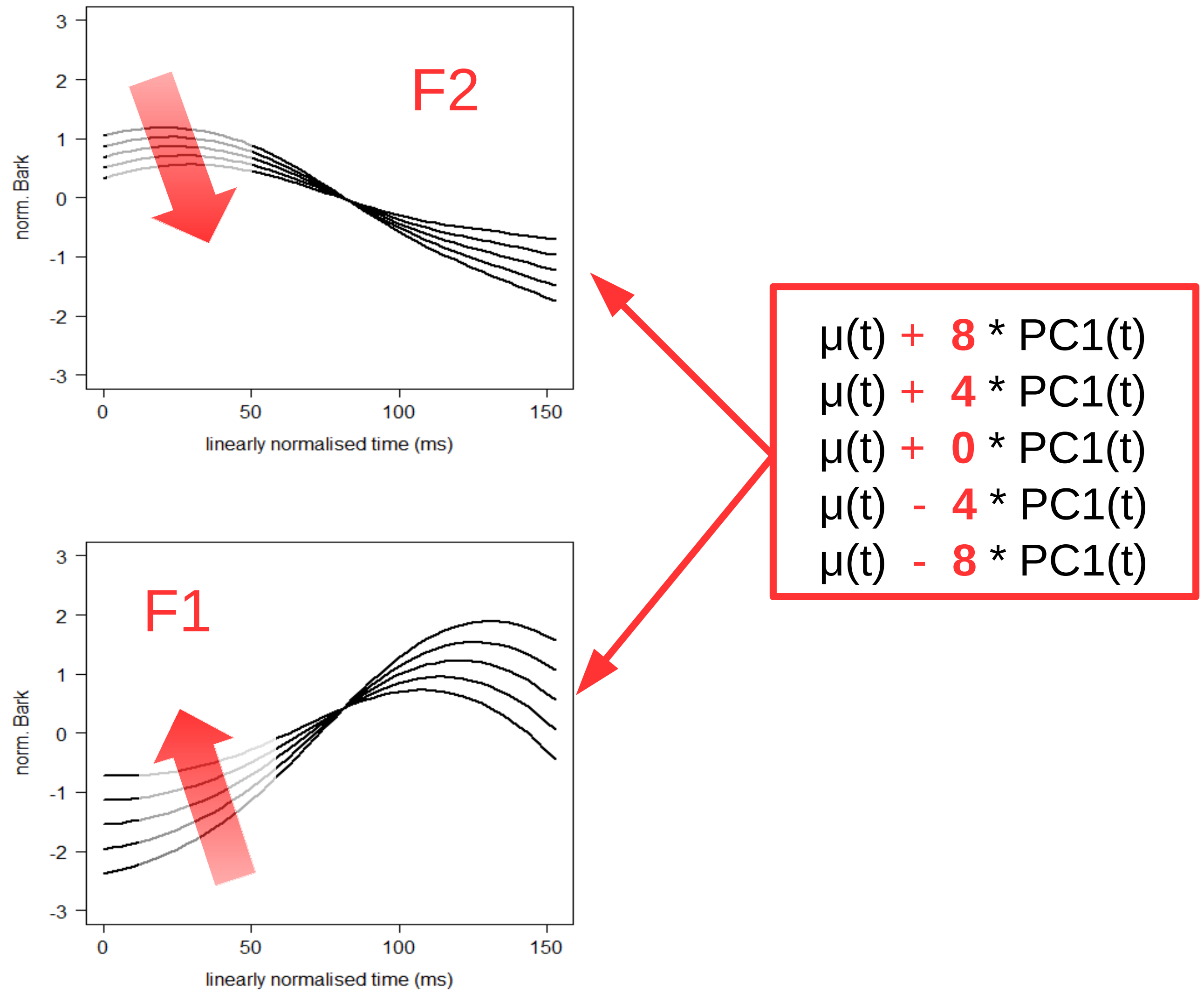
2D FPCA

ANOVA

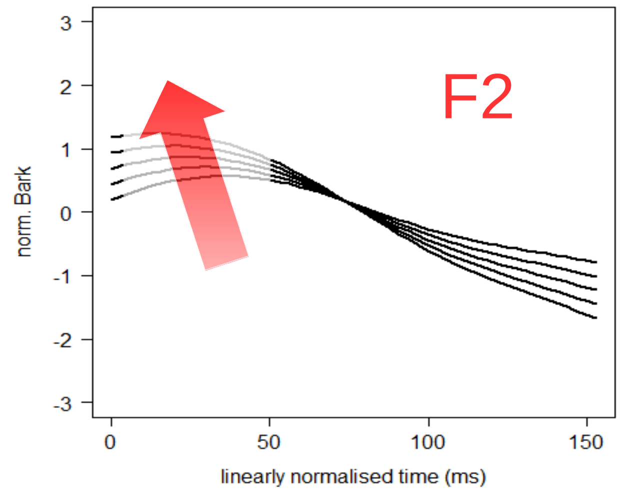
LM

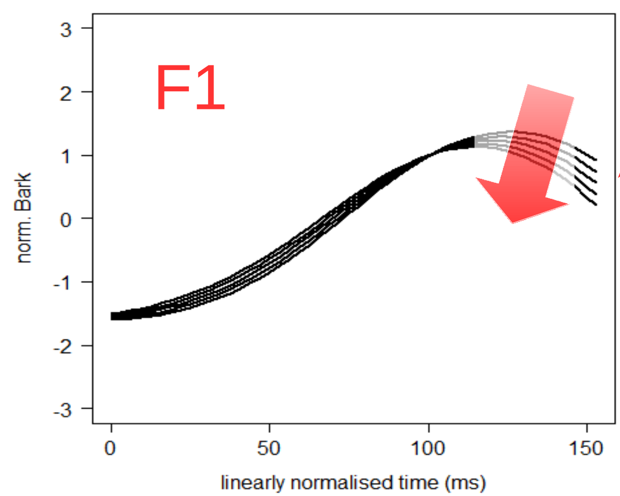
LMER

PC1 scores


F2
3
2
1
0
-1
-2
-3
norm. Bark
0
50
100
150
linearly normalised time (ms)
F1
3
2
1
0
-1
-2
-3
norm. Bark
0
50
100
150
linearly normalised time (ms)
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)

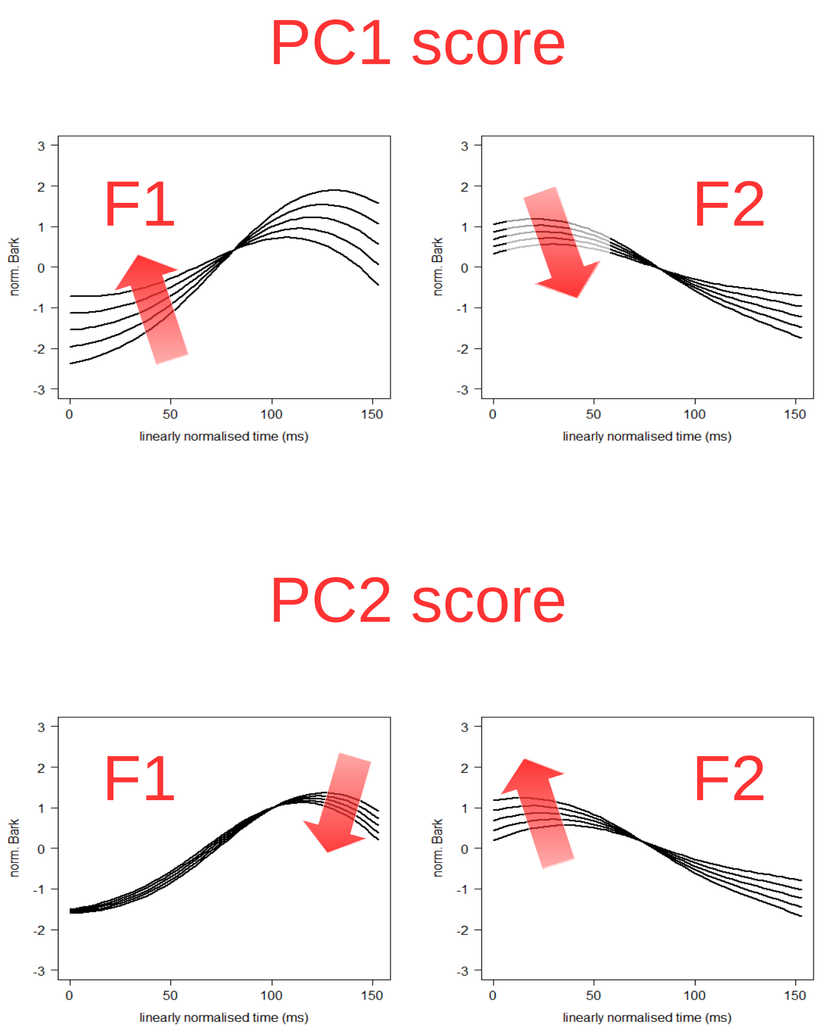
PC2 scores

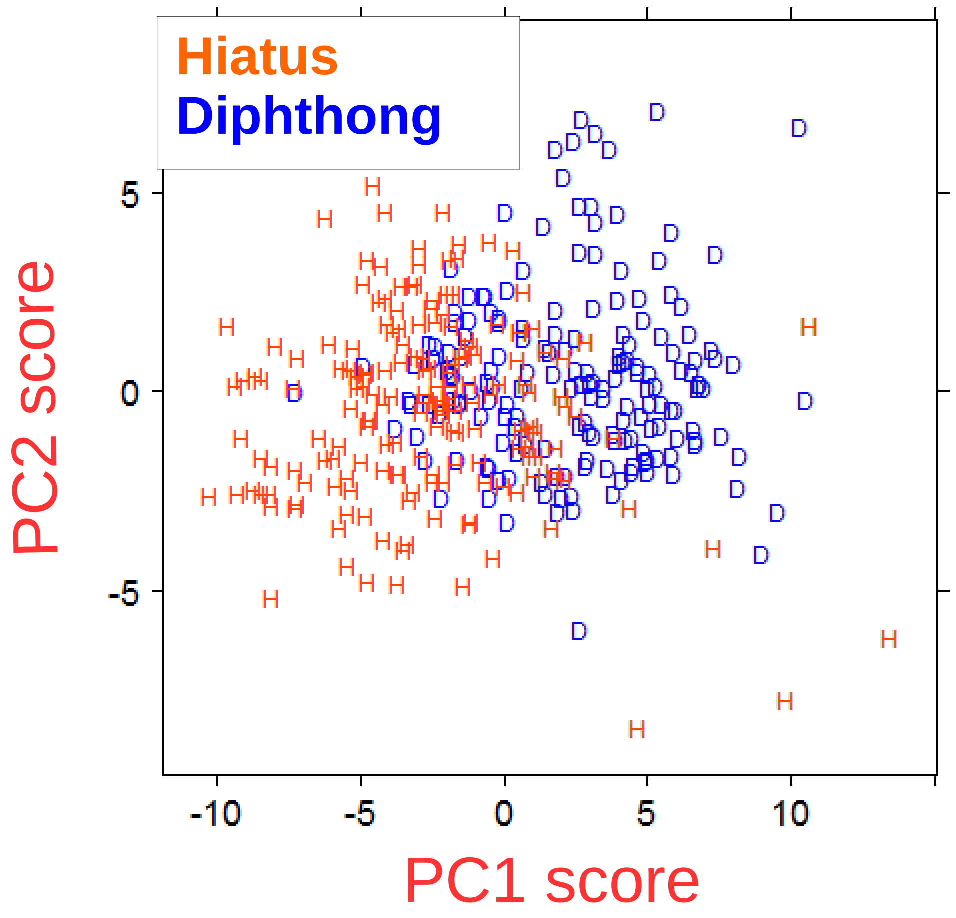




$$\mu(t) + 4 * PC1(t)$$
$$\mu(t) + 2 * PC1(t)$$
$$\mu(t) + 0 * PC1(t)$$
$$\mu(t) - 2 * PC1(t)$$
$$\mu(t) - 4 * PC1(t)$$

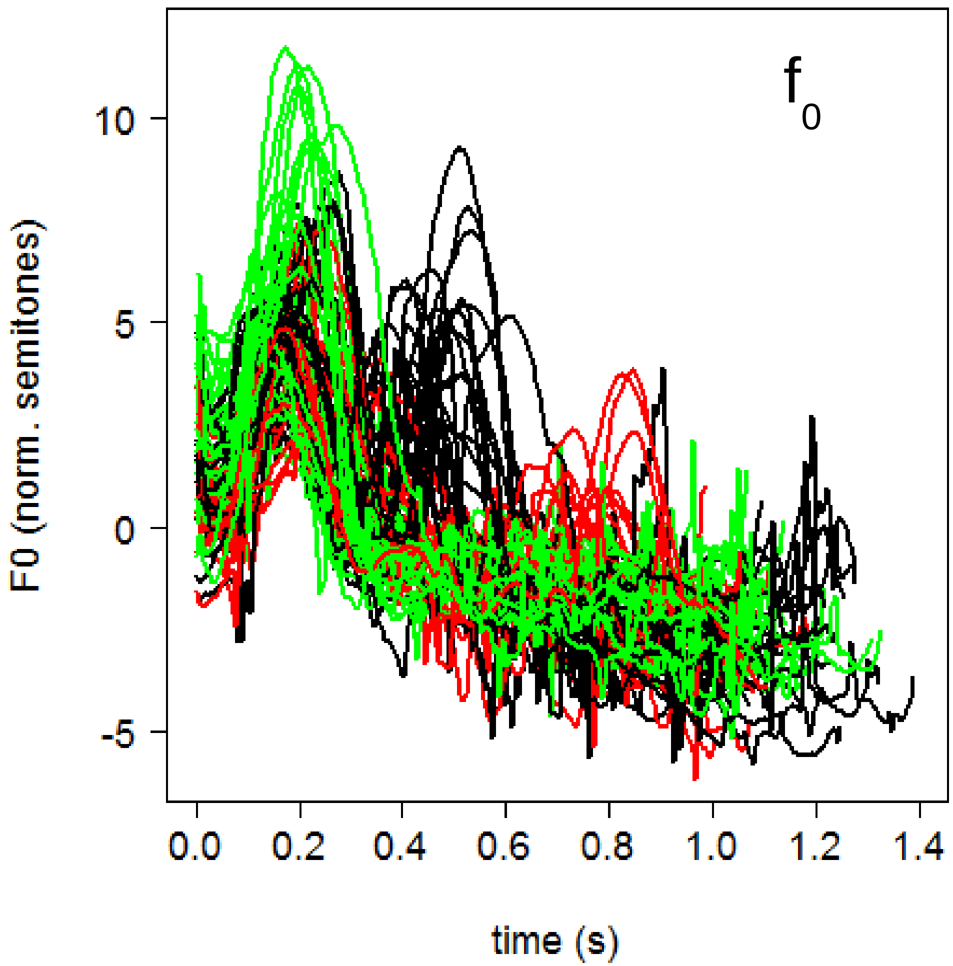
2D curve parametrisation





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)

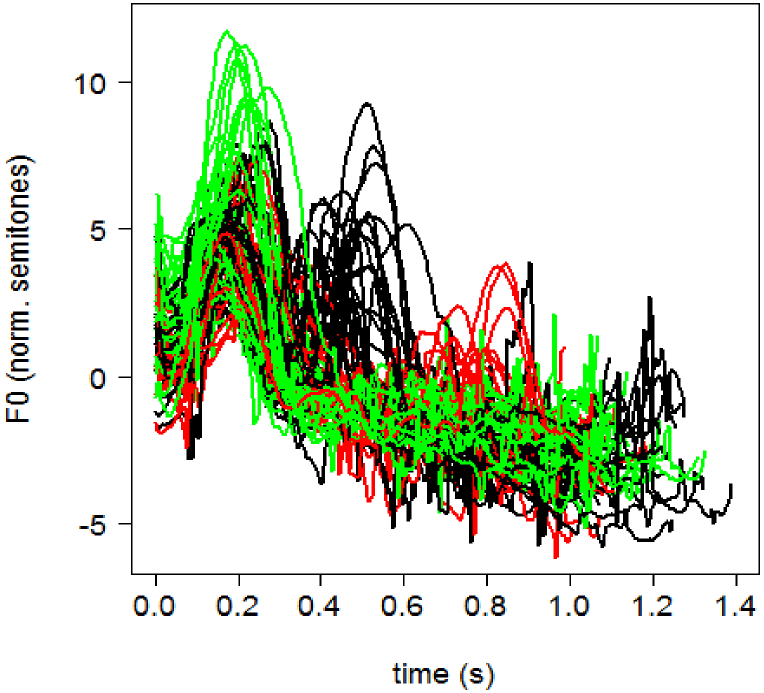
  VCVCV CVCV CV CVCV

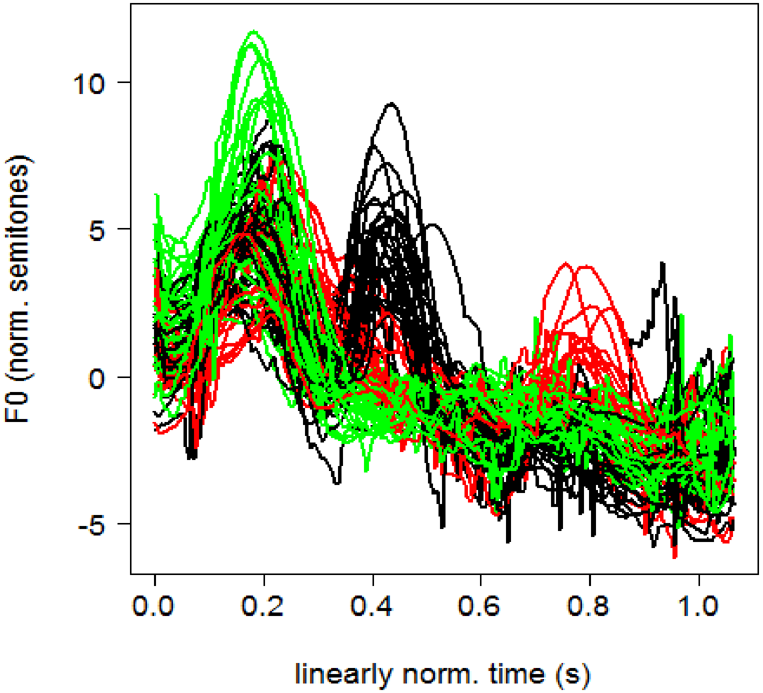
… 15 segments!

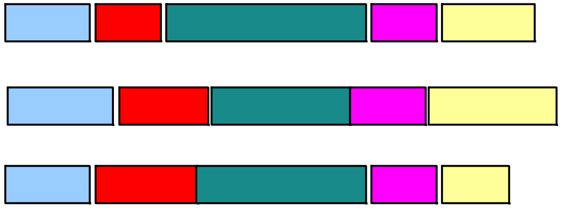
Linear time normalisation

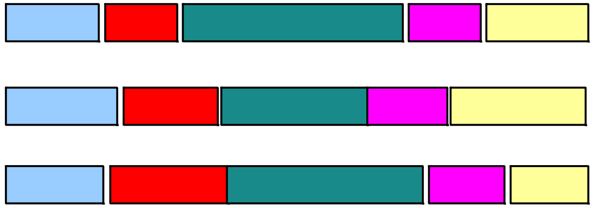
BEFORE

AFTER



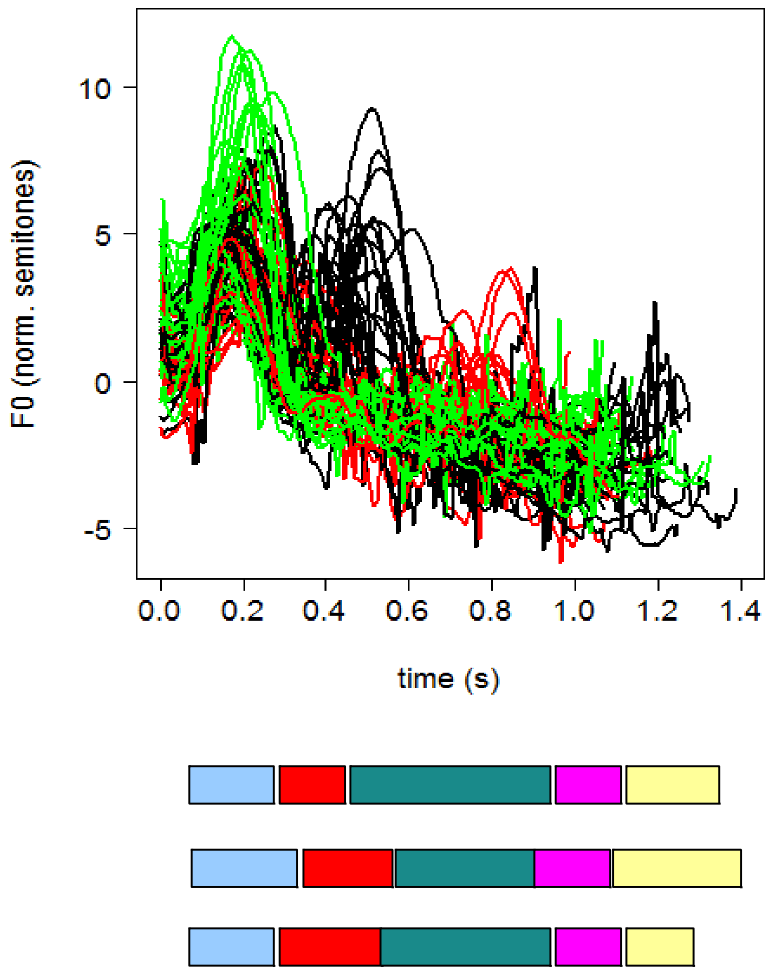




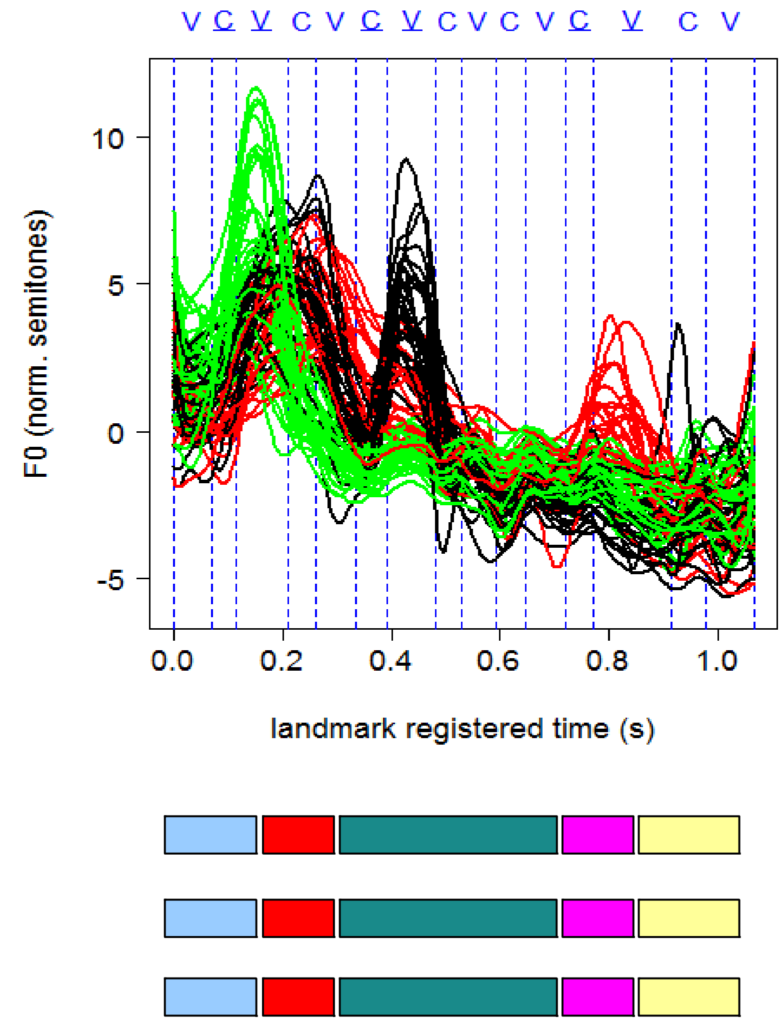


Landmark registration

BEFORE



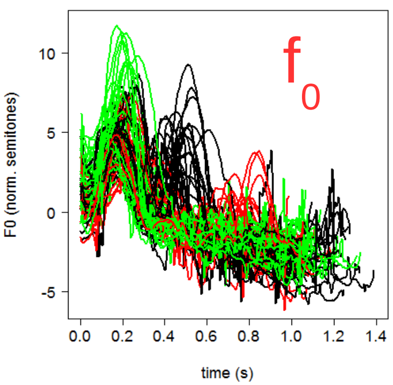
AFTER

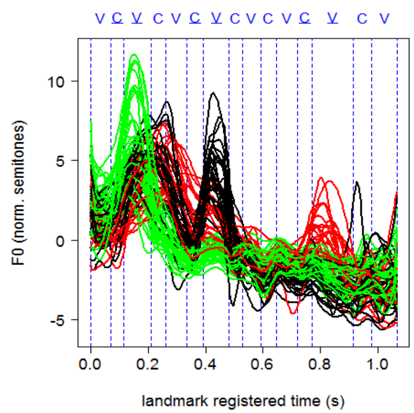


Using landmark registration

CURVES

NUMBERS





segment durations

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

FPCA

PCA

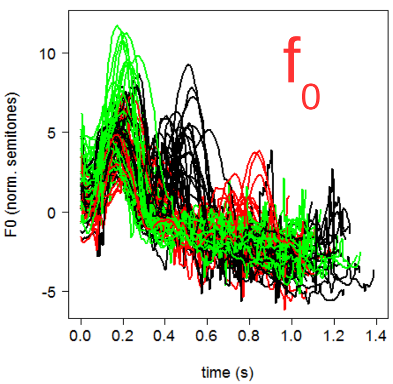
ANOVA

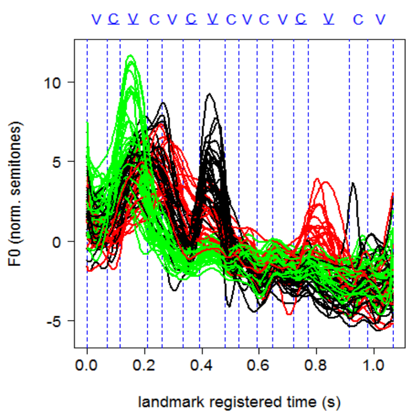
LM

LMER

Using landmark registration

CURVES





segment durations

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

MIND THE GAP

FPCA

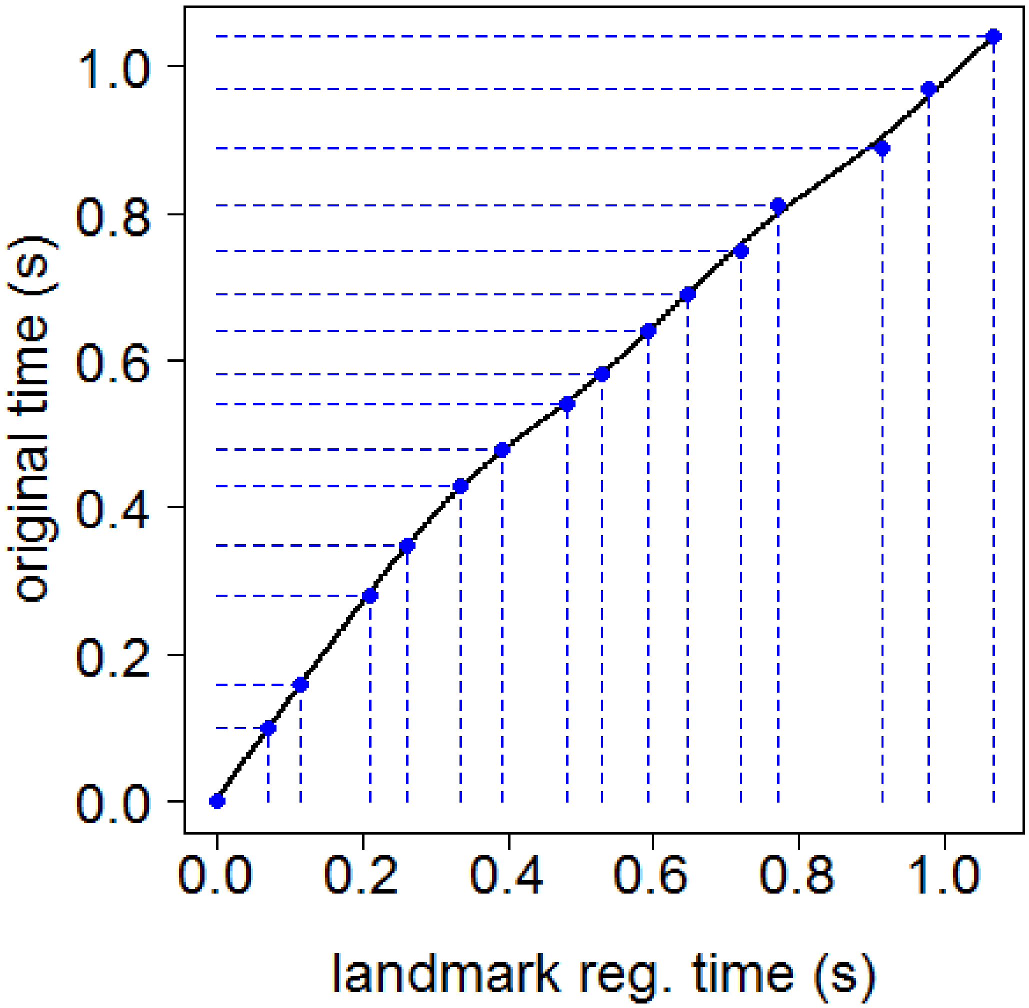
NUMBERS

ANOVA

LM

LMER

Inside landmark registration

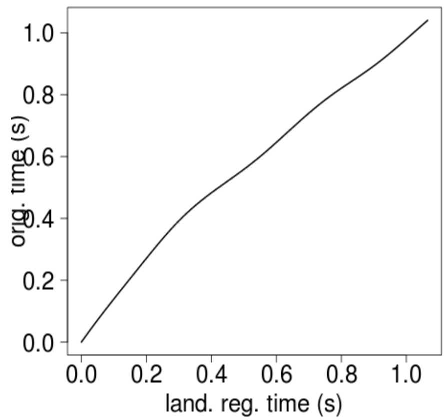


h(t)

Relative log rate

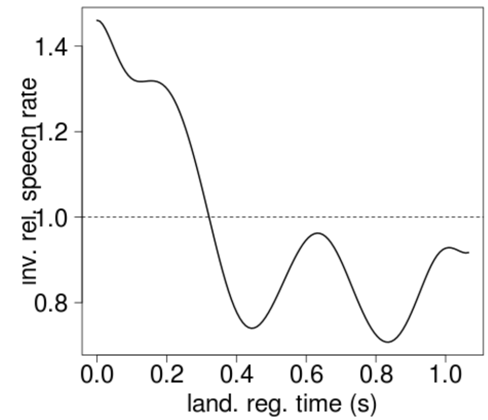
1

h(t)



2

dh(t)/dt

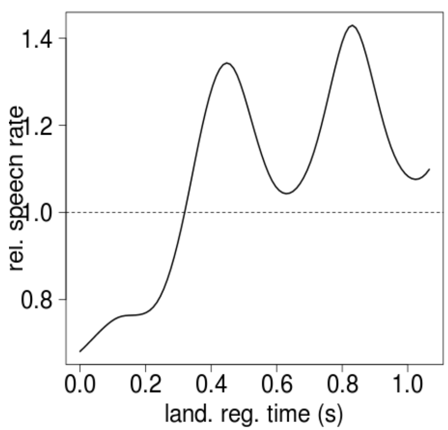


REVERSIBLE!

1 ⟷ 4

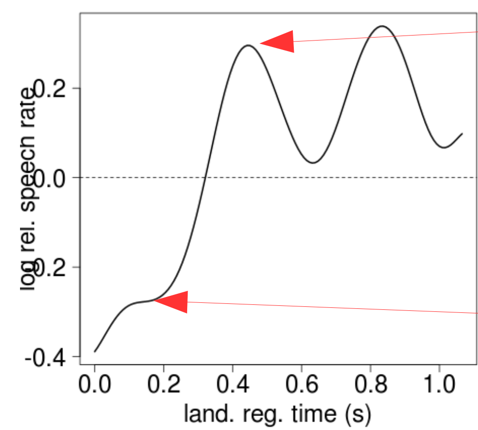
3

- dh(t)/dt



4

- log dh(t)/dt



+ 0.25 → duration / 1.28

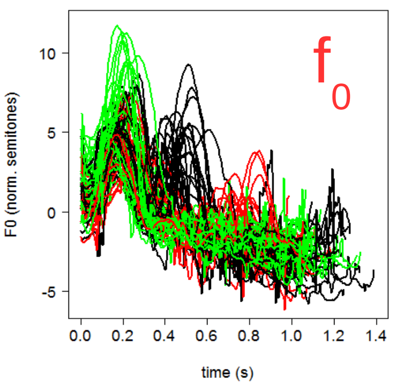
0 → same duration

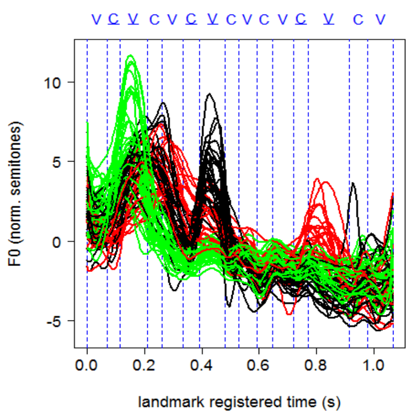
- 0.25 → duration * 1.28

Using log rates

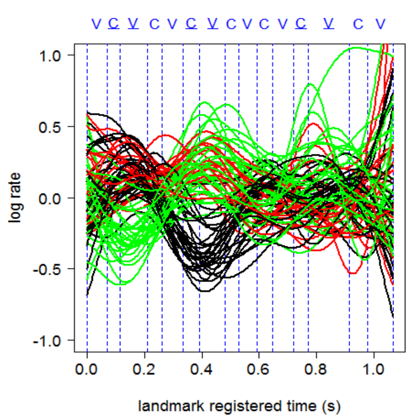
CURVES

NUMBERS





log rates



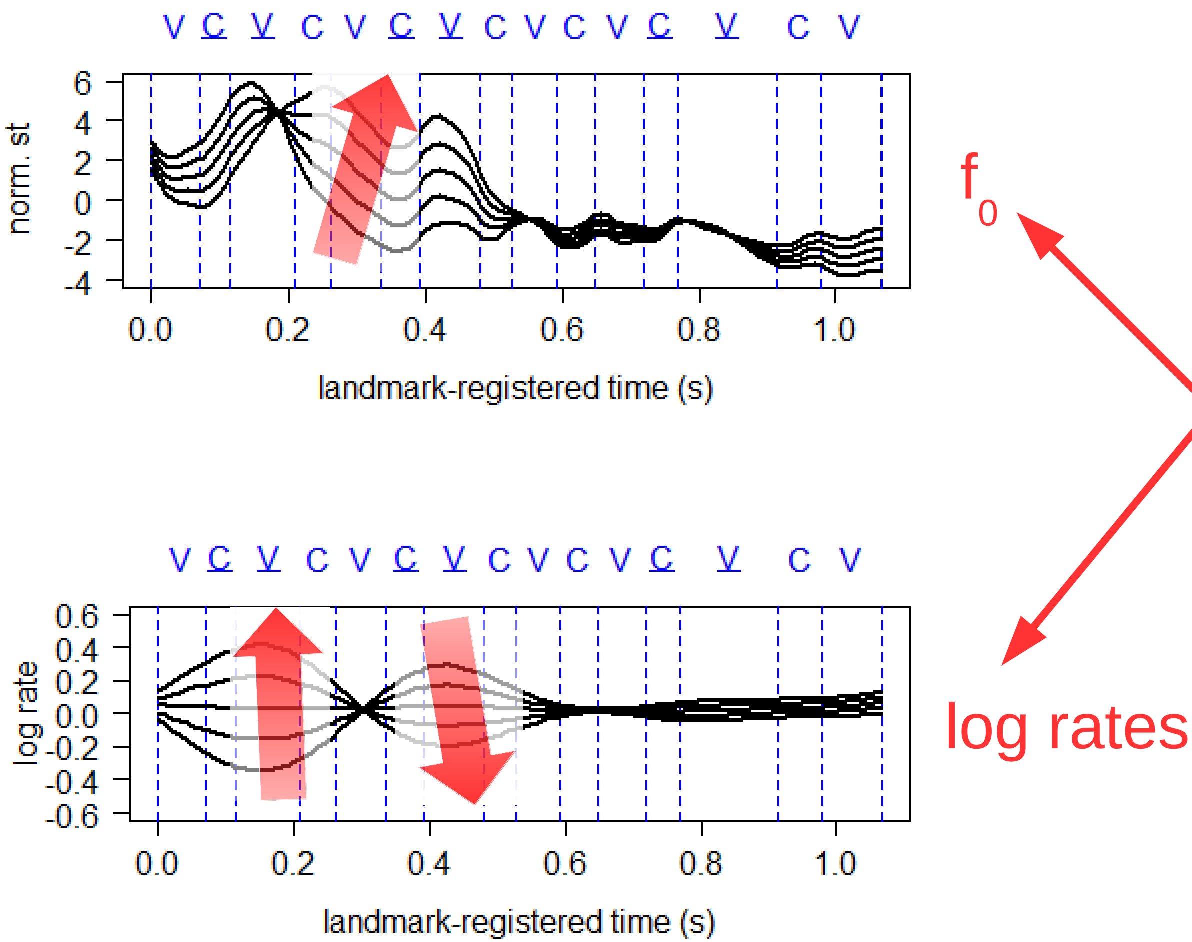
2D FPCA

ANOVA

LM

LMER

PC1 scores



$\mu(t) + 2 * PC1(t)$

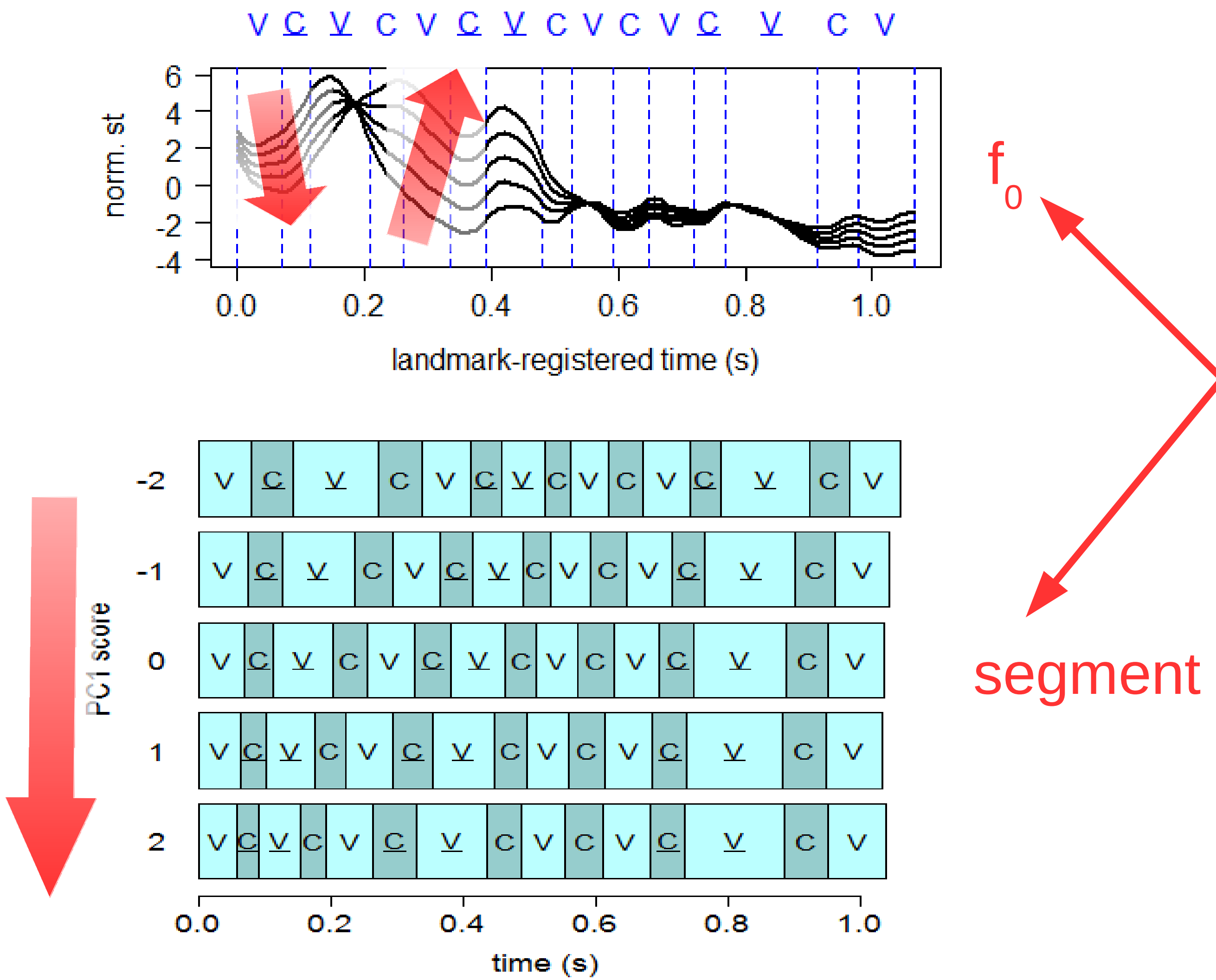
$\mu(t) + 1 * PC1(t)$

$\mu(t) + 0 * PC1(t)$

$\mu(t) - 1 * PC1(t)$

$\mu(t) - 2 * PC1(t)$

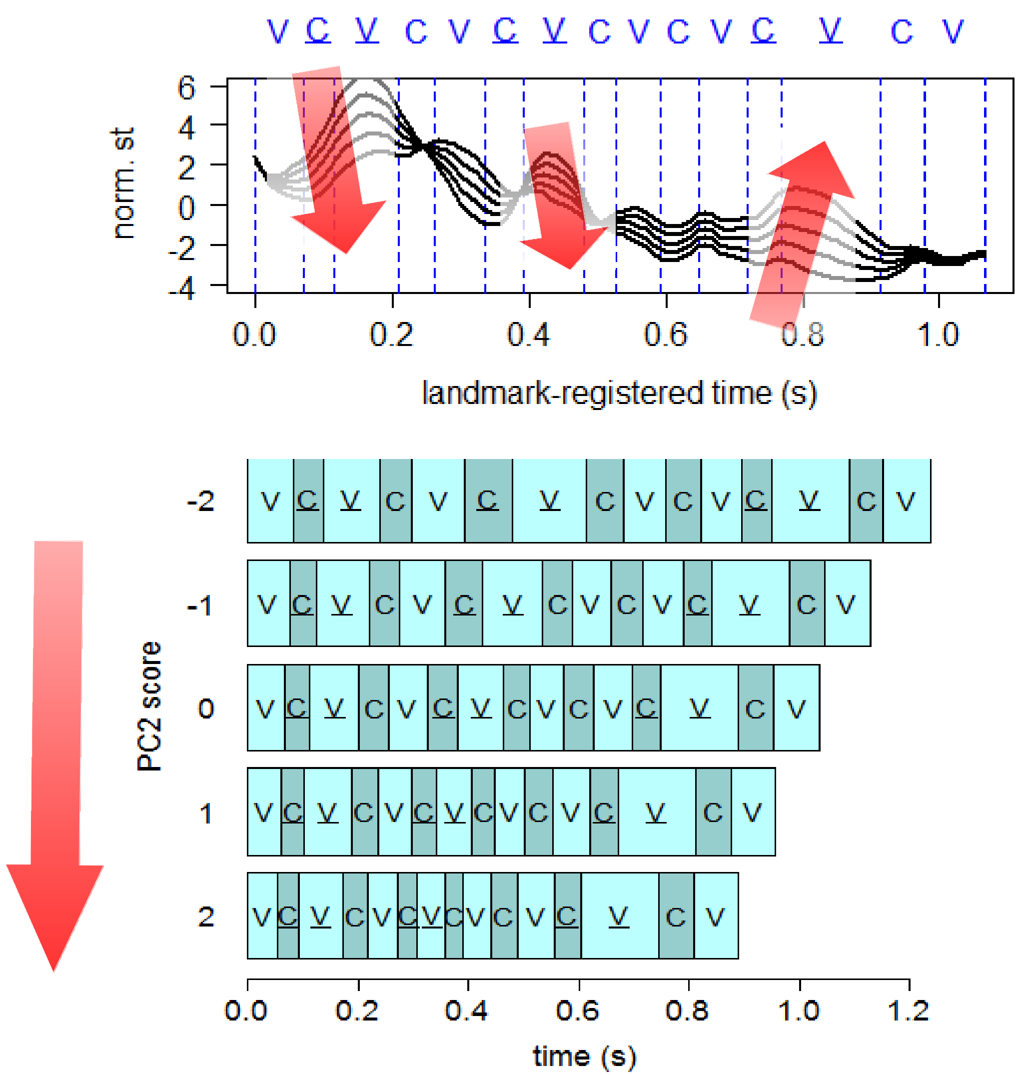
PC1 scores



$\mu(t)$ + **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

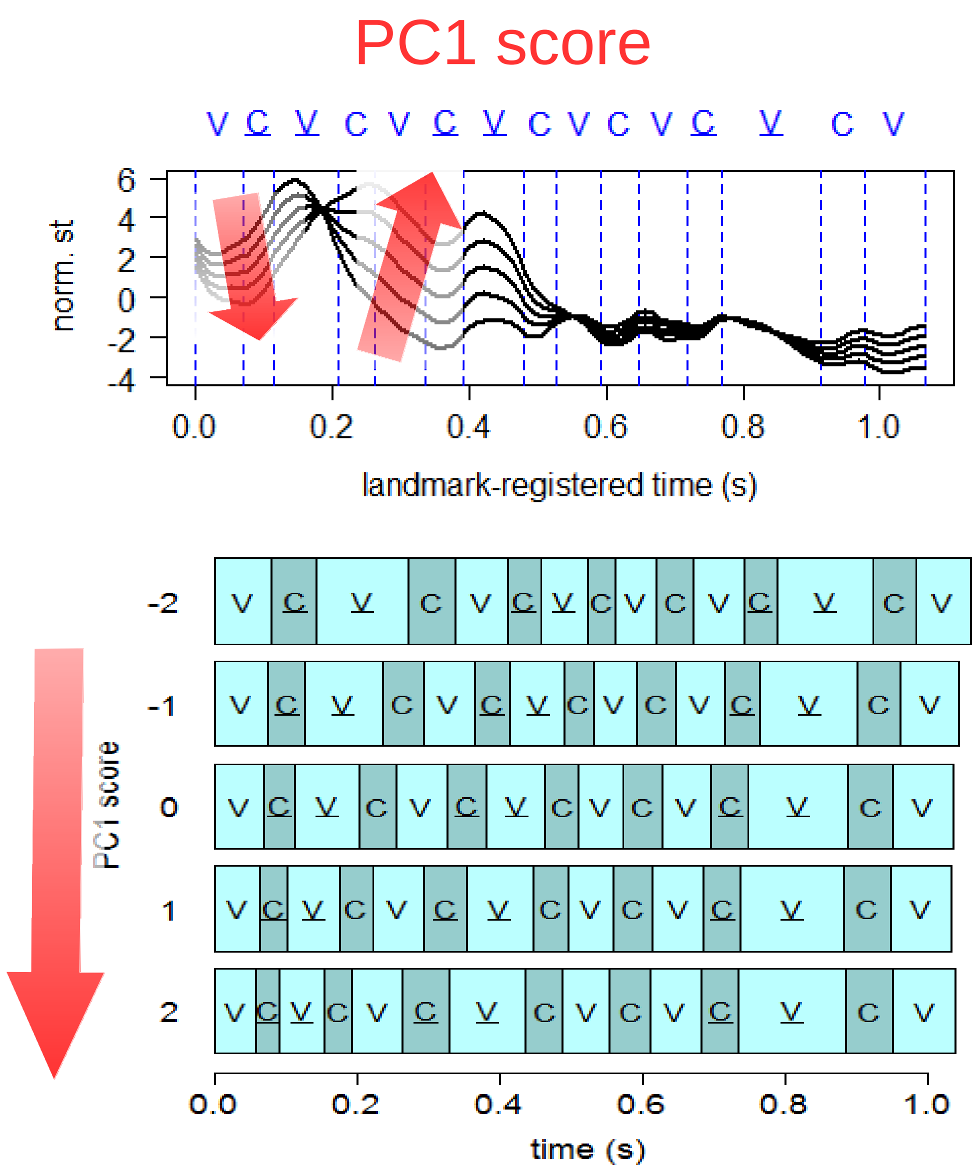


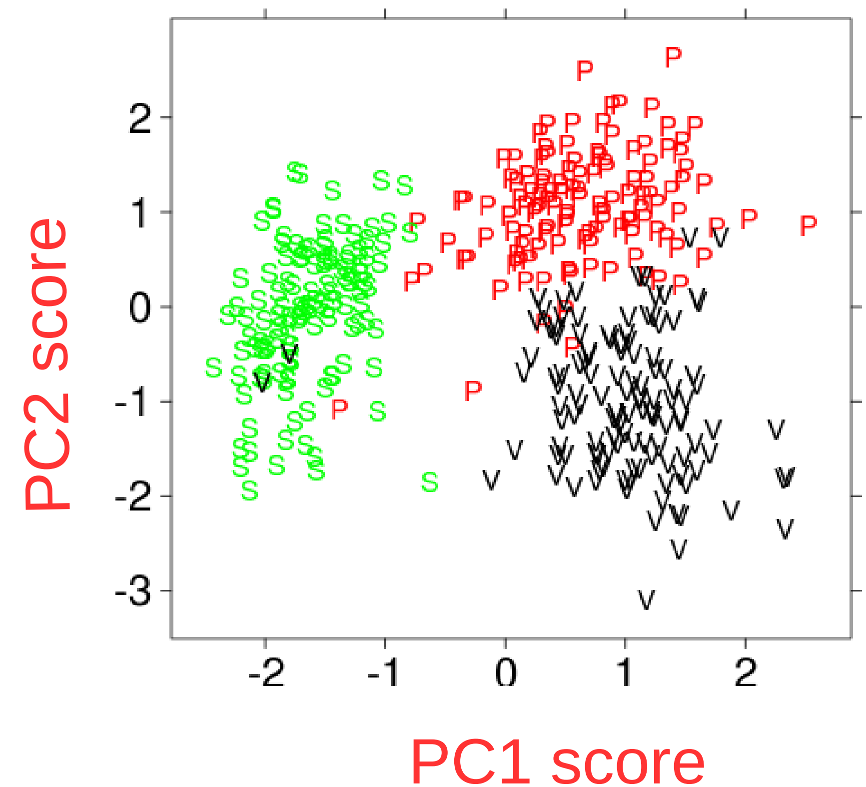
f_0

μ(t) + 2 * PC1(t)
μ(t) + 1 * PC1(t)
μ(t) + 0 * PC1(t)
μ(t) - 1 * PC1(t)
μ(t) - 2 * PC1(t)

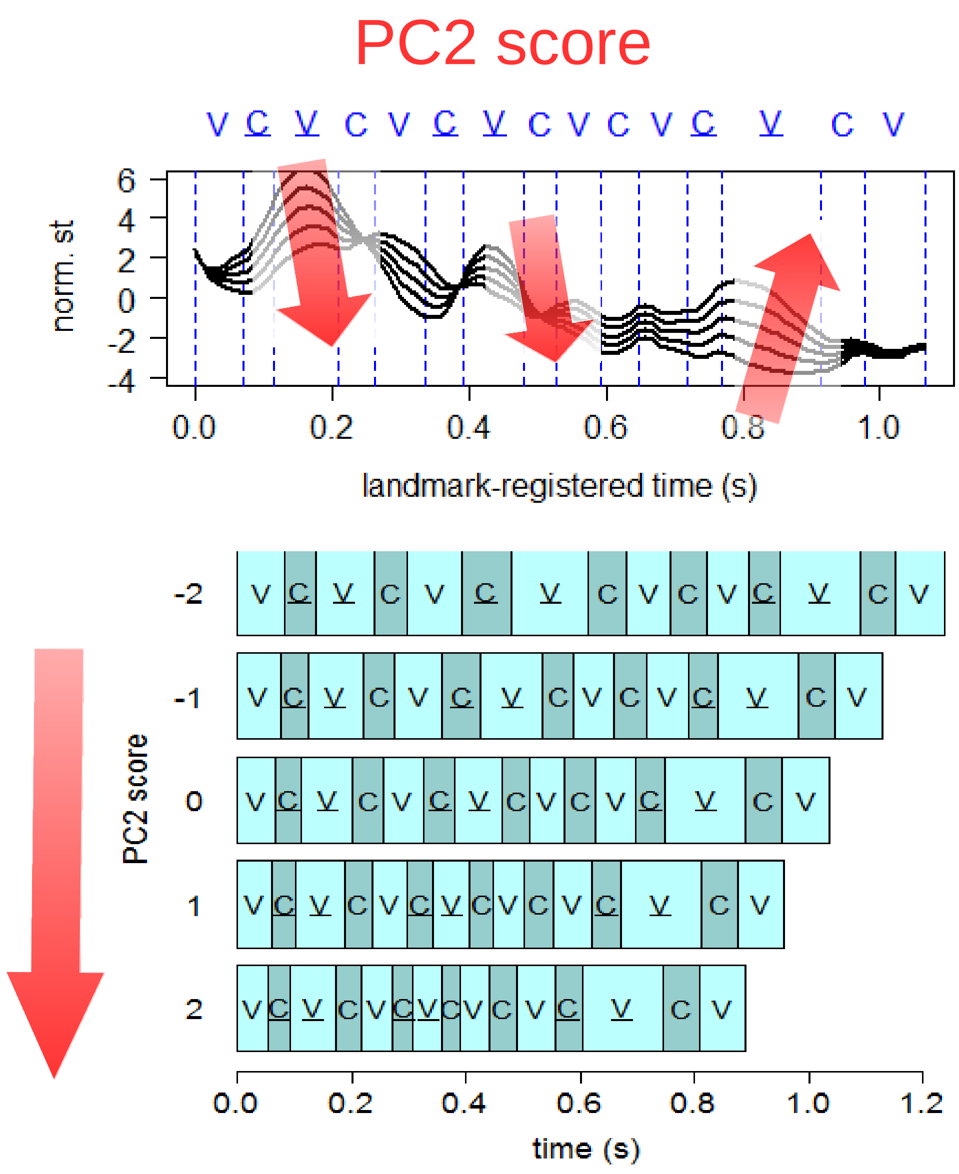
segment durations

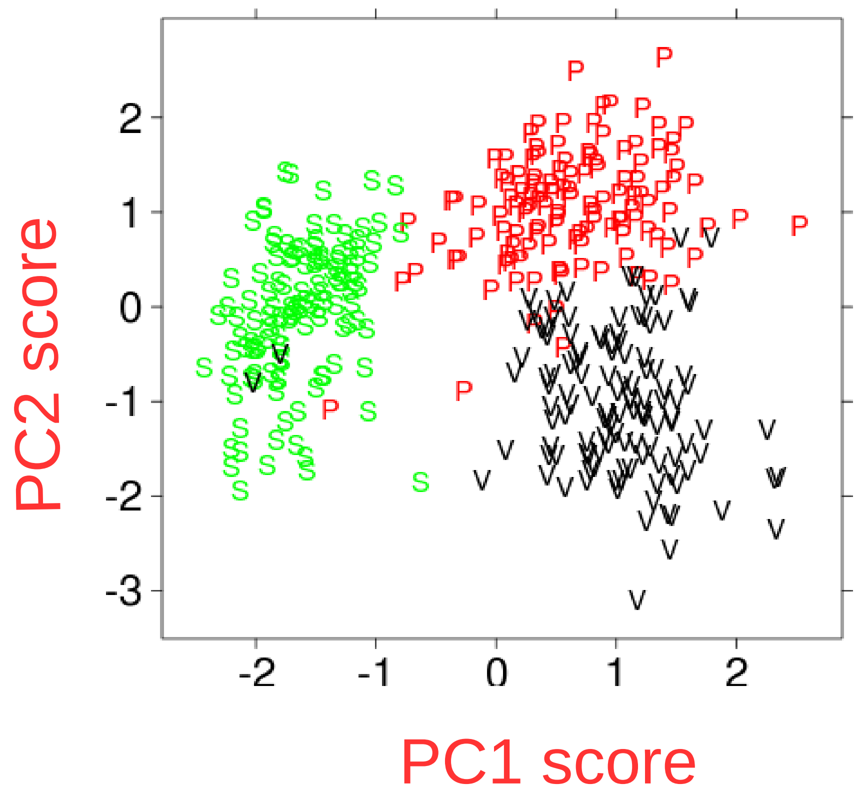
multi-segment curve parametrisation





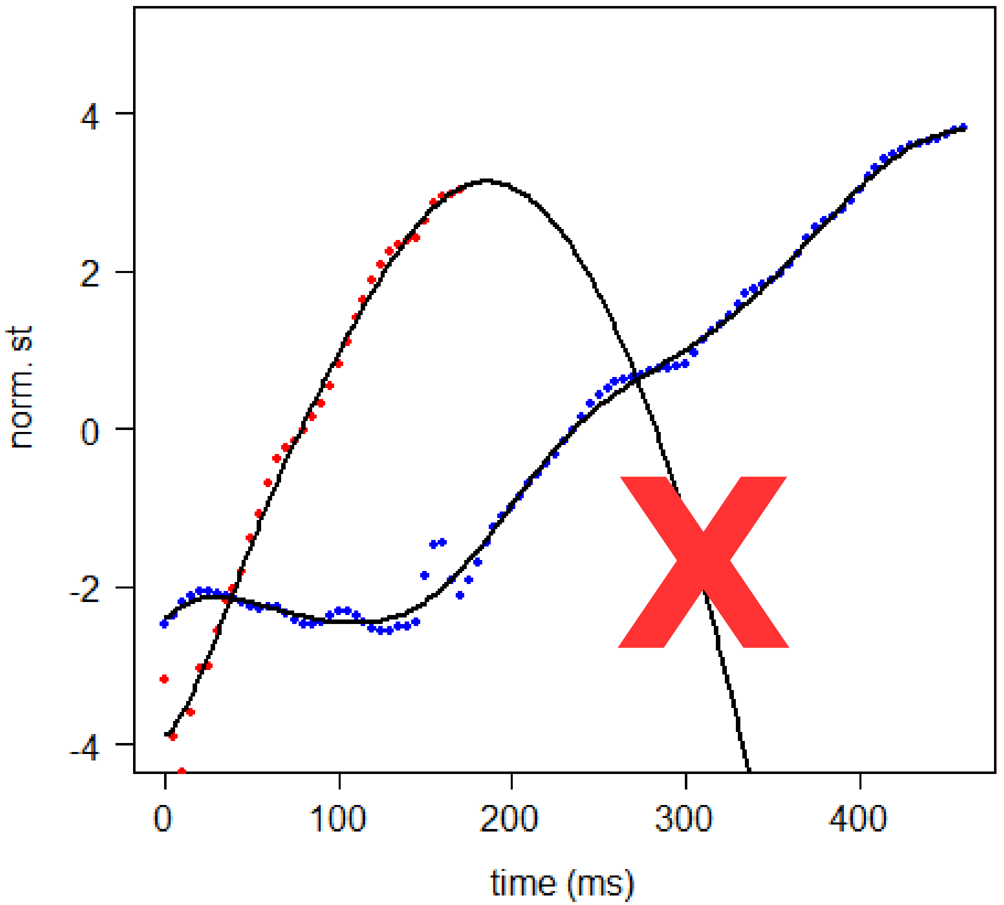
multi-segment curve parametrisation



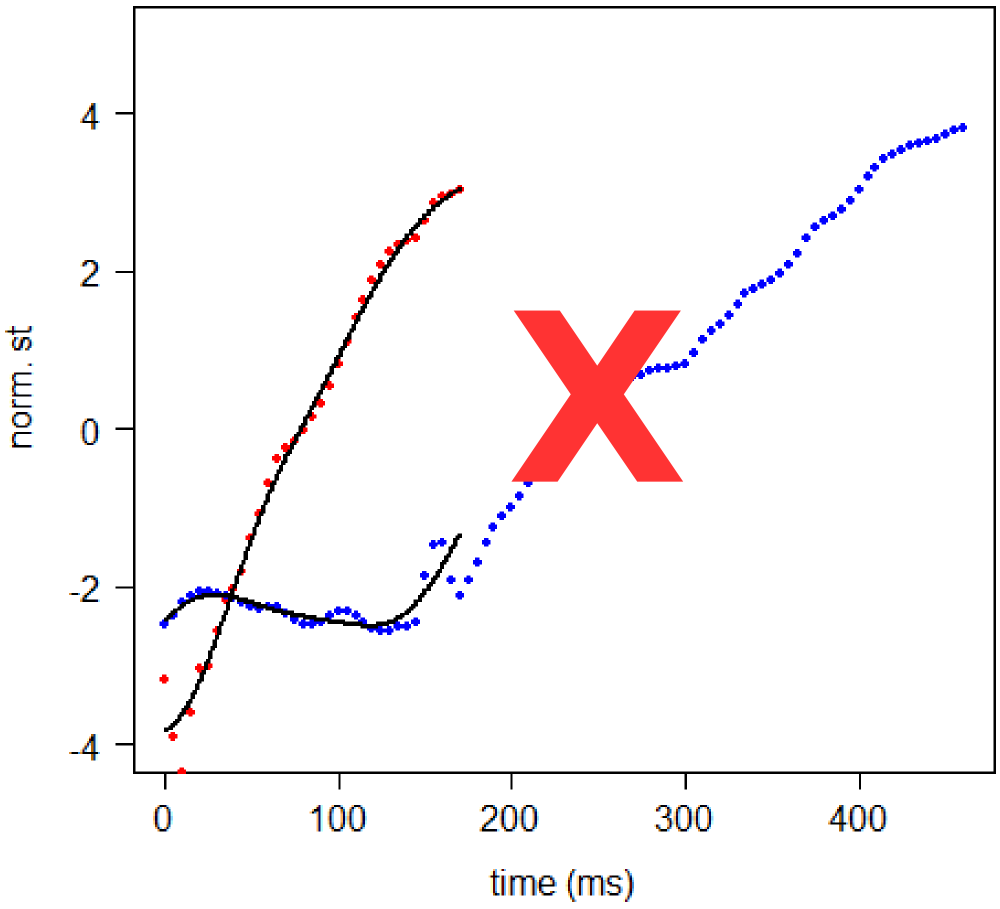


Extra slides

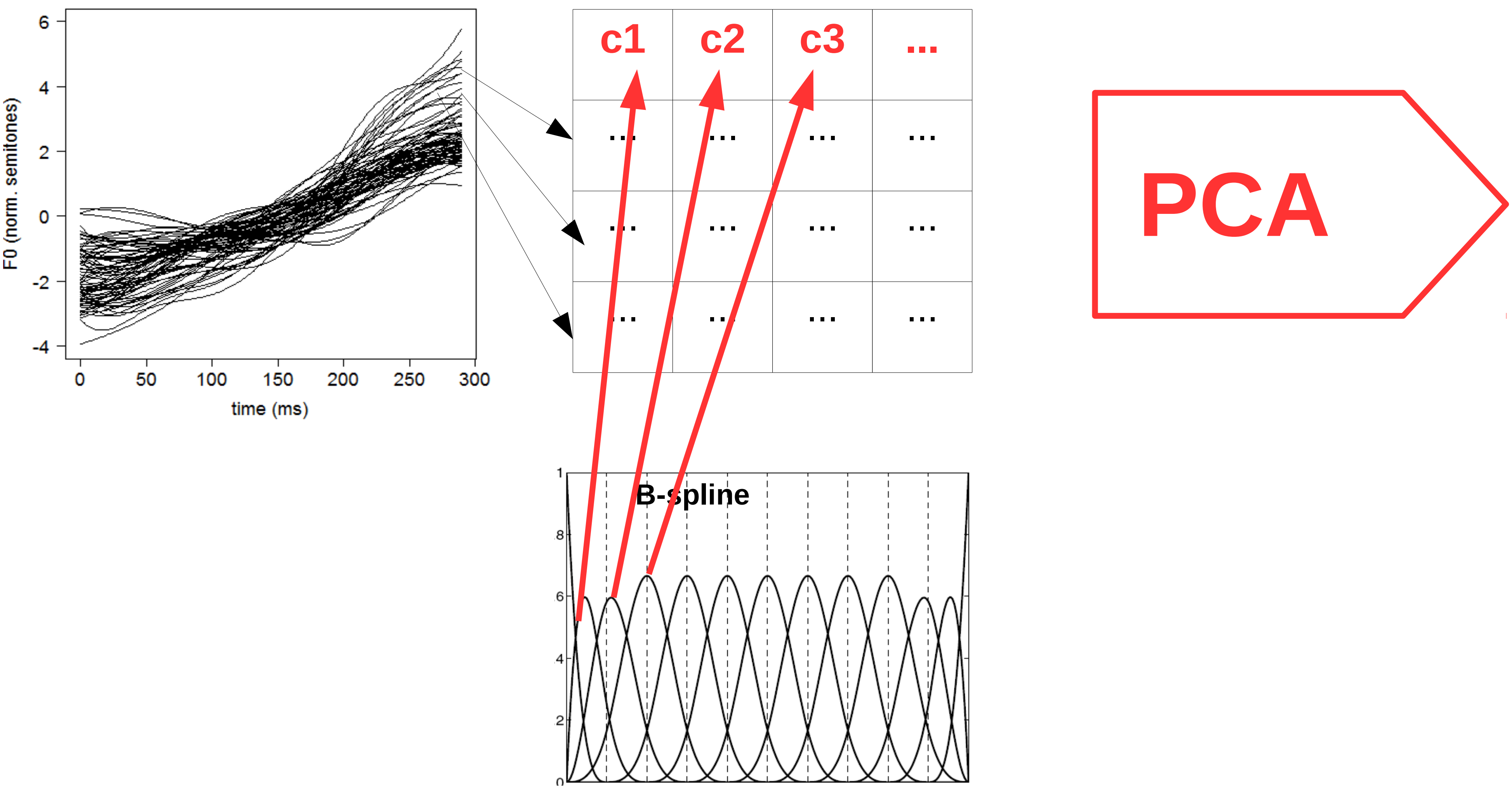
Take longest duration



Take shortest duration



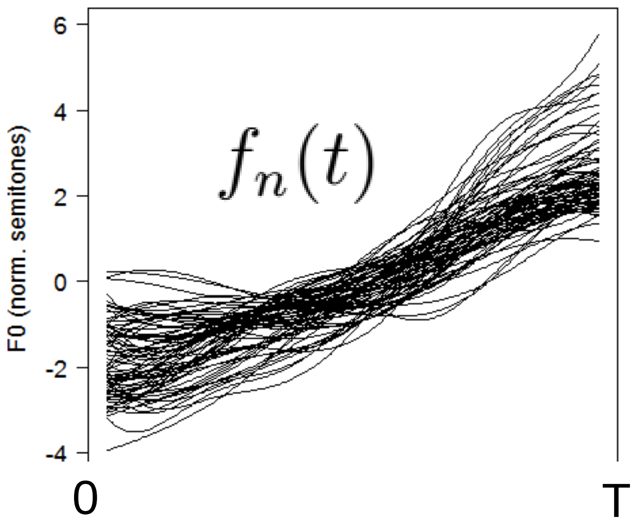
Principal Component Analysis



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 $f_n(t)$