Reconstruction of sequential data with probabilistic models and continuity constraints

Miguel Á. Carreira-Perpiñán

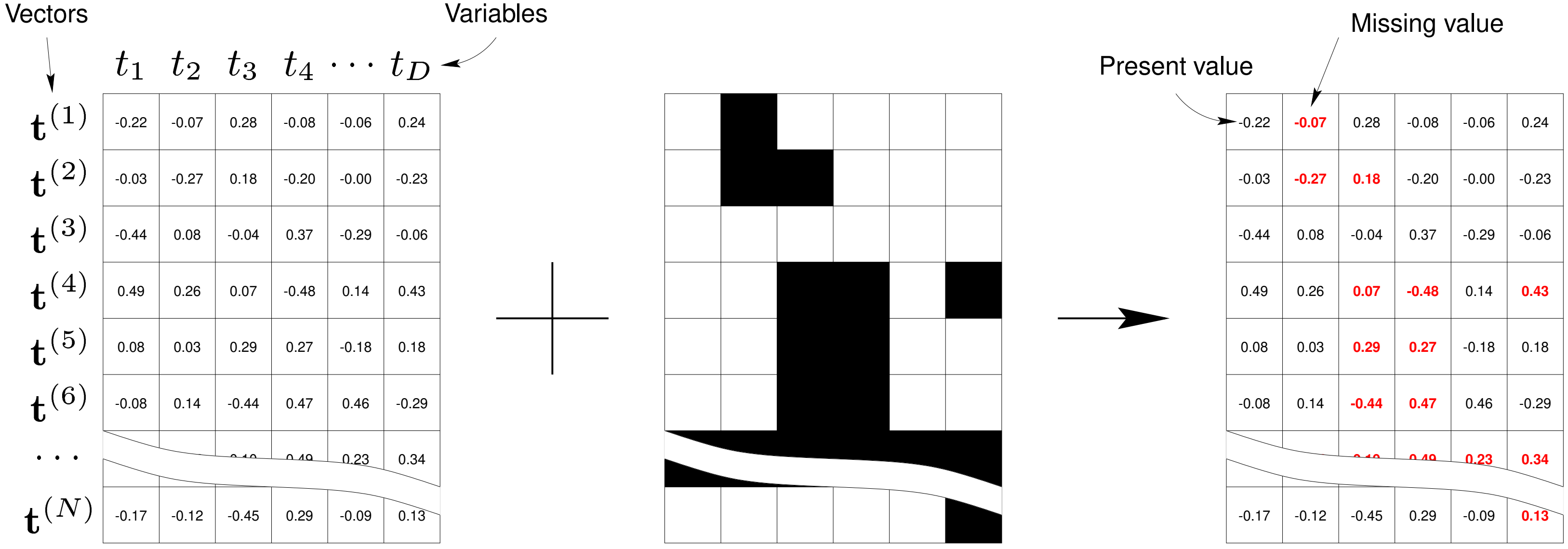
Dept. of Computer Science, University of Sheffield

miguel@dcs.shef.ac.uk http://www.dcs.shef.ac.uk/~miguel

Contents

Definition of the problem of data reconstruction

- Given a sequence $\{\mathbf{t}^{(1)}, \dots, \mathbf{t}^{(N)}\} \subset \mathbb{R}^D$ where part of the data are missing, reconstruct the whole sequence to minimise an error criterion.



	t_1	t_2	t_3	t_4	$\cdots$	t_D
$\mathbf{t}^{(1)}$	-0.22	-0.07	0.28	-0.08	-0.06	0.24
$\mathbf{t}^{(2)}$	-0.03	-0.27	0.18	-0.20	-0.00	-0.23
$\mathbf{t}^{(3)}$	-0.44	0.08	-0.04	0.37	-0.29	-0.06
$\mathbf{t}^{(4)}$	0.49	0.26	0.07	-0.48	0.14	0.43
$\mathbf{t}^{(5)}$	0.08	0.03	0.29	0.27	-0.18	0.18
$\mathbf{t}^{(6)}$	-0.08	0.14	-0.44	0.47	0.46	-0.29
$\cdots$					0.23	0.34
$\mathbf{t}^{(N)}$	-0.17	-0.12	-0.45	0.29	-0.09	0.13

Original sequence | Missing data pattern (mask) | Observed sequence

- Assume $\{\mathbf{t}^{(n)}\}_{n=1}^N$ are samples from a continuous function $\mathfrak{F}$ of an independent variable $\mathbf{z}$ (the *experimental conditions*) at points $\{\mathbf{z}^{(n)}\}_{n=1}^N$.

Examples of problem	$\mathbf{z}$	$\mathbf{t}$
Trajectory of a mobile point	time, 1D	spatial coordinates, 3D (Cartesian, spherical)
Utterance	time, 1D	speech feature vector, 13D (e.g. LSPs, MFCCs, PLPs)
Wind field on the ocean surface	spatial coordinates, 2D	wind velocity vector, 2D

- Reconstruction is possible due to two kinds of redundancy in the data:
  - *pointwise*: low intrinsic dimensionality of the observed variables $t_1, \dots, t_D$ due to correlations
  - across a whole *sequence*: due to the continuity of $\mathbf{t} = \mathfrak{F}(\mathbf{z})$.
- Reconstruction can be a previous step to some further processing, e.g. classification.
- Types of reconstruction:
  - *Pointwise (or local)*: reconstruction of a vector $\mathbf{t}^{(n)}$ given information present in $\mathbf{t}^{(n)}$ only.
  - *Multiple pointwise*: like pointwise, but several candidate reconstructions for $\mathbf{t}^{(n)}$ are provided.
  - *Global*: reconstruction of the whole sequence $\{\mathbf{t}^{(n)}\}_{n=1}^{N}$ given information present in it.

Statistical mapping approximation

- Definition: construct a function $\mathbf{x} \rightarrow \mathbf{y}$ given a data set of input-output pairs $\{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$.
- Efficiently solved by *universal function approximators*, e.g. MLPs.
- Statistically, the optimal *univalued* mapping (in the L_2 sense) is the *conditional mean* or *regression* $\mathrm{E}\{\mathbf{y}|\mathbf{x}\}$, to which an MLP trained to minimise least squares will converge.
- **Mapping inversion**: given a known forward function, obtain its inverse. Examples: inverse kinematics of a robot arm, acoustic-to-articulatory mapping.
- Statistically, it is just a mapping approximation problem with data $\{(\mathbf{y}_n, \mathbf{x}_n)\}_{n=1}^N$. There is no need to know the forward mapping.
- If the forward mapping is injective, the inverse will be multivalued and standard MLPs can't deal with it. A possible method is to use an *assembly* of MLPs:
  - Train several MLPs so that each one approximates a different univalued branch of the mapping.
  - This requires clustering the data, which is difficult–e.g. how many MLPs are necessary? Joint density estimation achieves this naturally.
- Mapping approximation and inversion are particular cases of missing data reconstruction.

Imputation modelling

- **Imputation modelling** is predictive modeling: create plausible imputations of the missing values. The goal is not:
  - to predict the missing values with the greatest accuracy
  - to describe the data in a scientifically meaningful way

  but:
  - to properly reflect uncertainty
  - to preserve important aspects of the data distribution
  - to preserve important relationships.
- The statistical analysis of incomplete data is based in a predictive distribution of the missing data obtained from a model for the complete data: $p(\text{missing}|\text{present})$.
- Two kinds of imputation exist:
  - *Pointwise imputation*: each missing datum is replaced by a single value from the predictive distribution, e.g. the mean.
  - *Multiple imputation*: each missing datum is replaced by $M > 1$ simulated values. The resulting M versions of the complete data are analysed by standard complete-data methods and the results combined to produce inferential statements (e.g. interval estimates or p-values). It is achieved by Markov Chain Monte Carlo methods, i.e., simulated draws from the predictive distribution—due to its intractability.

Notation

Vectors are boldface lowercase, scalars are italics lowercase, sets are calligraphic uppercase.

Symbol	Meaning
$\mathbf{t} = (t_1, \dots, t_D) \in \mathcal{T} \subset \mathbb{R}^D$ $\mathbf{x} = (x_1, \dots, x_L) \in \mathcal{X} \subset \mathbb{R}^L$	D-dimensional vector in observed space $\mathcal{T}$ L-dimensional vector in latent space $\mathcal{X}$
$\mathcal{I}, \mathcal{J} \in \{1, \dots, D\}$	Sets of indices for variable selection. Example: if $\mathcal{I} = \{1, 7, 3\}$ and $\mathcal{J} = \{2, 5\}$ then: $\mathbf{t}_{\mathcal{I}}$ is (t_1, t_7, t_3) $\mathbf{t}_{\mathcal{J}}$ is (t_2, t_5) $p(\mathbf{t}_{\mathcal{I}})$ is $p(t_1, t_3, t_7)$ $p(\mathbf{t}_{\mathcal{J}} \vert \mathbf{t}_{\mathcal{I}})$ is $p(t_2, t_5 \vert t_1, t_3, t_7)$ $p(\mathbf{t}_{\mathcal{I}}, \mathbf{t}_{\mathcal{J}})$ is $p(t_1, t_2, t_3, t_5, t_7)$
$\mathbf{t}_{\mathcal{I}} \rightarrow \mathbf{t}_{\mathcal{J}}$	$\mathbf{t}_{\mathcal{J}}$ as a function of $\mathbf{t}_{\mathcal{I}}$
$\{\mathbf{t}_n\}_{n=1}^N$ $\{\mathbf{t}^{(n)}\}_{n=1}^N$	Set of N vectors not necessarily in sequence Set of N vectors in (temporal) sequence

My approach in a nutshell

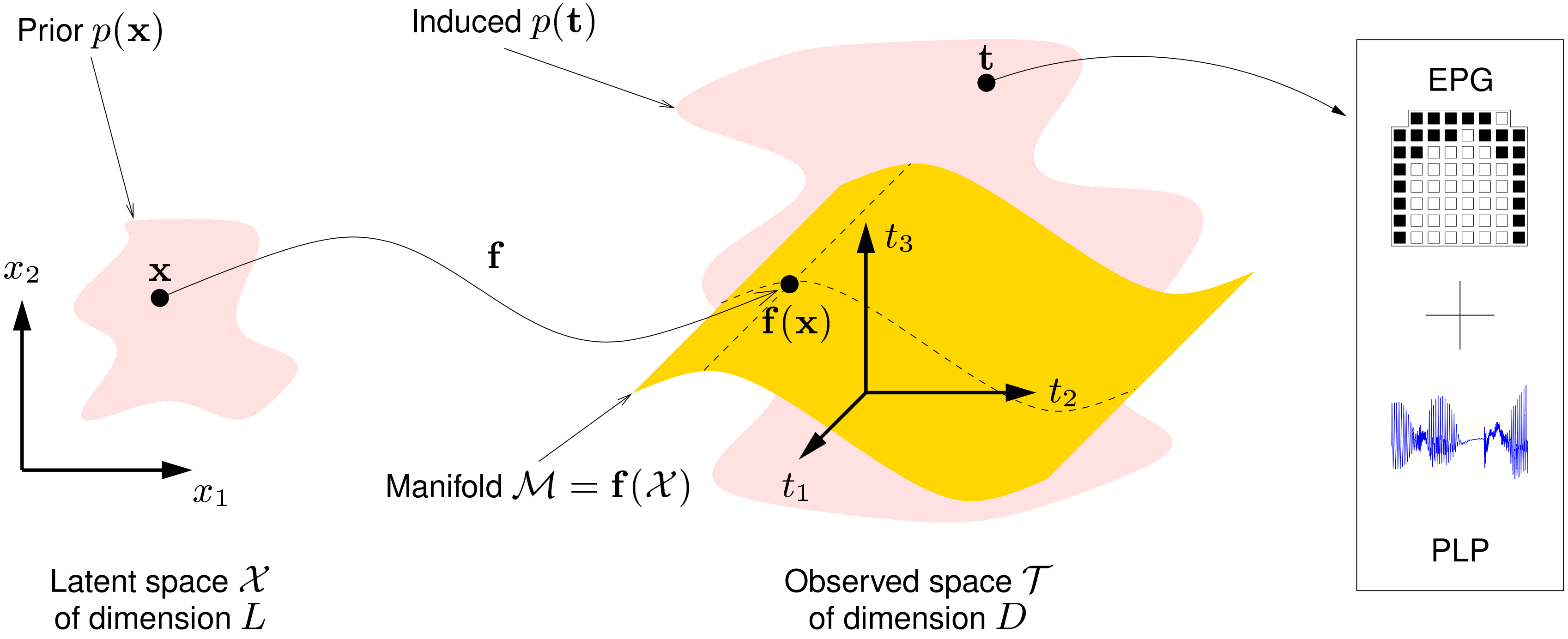
- Joint density model $p(\mathbf{t})$ of the observed variables $t_1, \dots, t_D$ estimated offline from a (complete) training set $\{\mathbf{t}_n\}_{n=1}^N$.
- At reconstruction time, given an incomplete sequence $\{\mathbf{t}^{(n)}\}_{n=1}^N$:
  - Multiple pointwise reconstruction at point n by the modes of the conditional distribution
  $$p\big(\text{missing variables}^{(n)} \big| \text{present variables}^{(n)}\big).$$
  - The reconstructed sequence is obtained by minimising a global continuity constraint using dynamic programming.

Offline estimation of joint density model

- We want to estimate a *parametric model* for the pdf of the observed variables $t_1, \dots, t_D$ from a training set consisting of (complete) data $\{\mathbf{t}_n\}_{n=1}^N$ not necessarily sequential.
- Several techniques exist with *universal approximation properties*: mixtures of Gaussians, kernel density estimation, RBFs, etc.
- It is also possible to estimate the model from incomplete data using an EM algorithm.
- It ought to allow efficient computation of *conditional probabilities* of the form $p(t_3 t_5 | t_1 t_2 t_4)$.
- The low intrinsic dimensionality of the data suggests using *latent variable models*.

Continuous latent variable models

Aim: to infer a *low-dimensional representation* of an observed, high-dimensional process.



The data distribution in observed space $\mathcal{T}$ is modelled using a low-dimensional representation in latent space $\mathcal{X}$. In the figure the observed space consists of 62-dimensional EPG patterns plus 12-dimensional PLP coefficients, thus $D = 74$.

Continuous latent variable models (cont.)

- The **prior** in latent space $p(\mathbf{x})$, the **mapping** $\mathbf{f}(\mathbf{x})$ and the **noise model** in observed space $p(\mathbf{t}|\mathbf{x})$ are equipped with parameters $\mathbf{\Theta}$.
- *Marginalisation* in latent space (often difficult): $p(\mathbf{t}) = \int p(\mathbf{t}|\mathbf{x})p(\mathbf{x})\,d\mathbf{x}$
  - $\mathbf{f}$ linear, $p(\mathbf{t}|\mathbf{x})$ normal centred in $\mathbf{t} = \mathbf{f}(\mathbf{x})$, $p(\mathbf{x})$ normal: analytically *solvable*
  - $\mathbf{f}$ nonlinear: analytically *insolvable*; approximate integral by a Monte Carlo method.
- *Maximum likelihood parameter estimation* from sample $\{\mathbf{t}_n\}_{n=1}^N$, usually via an *EM algorithm*: $\arg\max_{\mathbf{\Theta}} l(\mathbf{\Theta}) = \sum_{n=1}^N \log p(\mathbf{t}_n|\mathbf{\Theta})$.
- **Dimensionality reduction mapping** $\mathbf{g}$ via posterior distribution in latent space:

$$p(\mathbf{x}|\mathbf{t}) = \frac{p(\mathbf{t}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{t})} \quad \leadsto \quad \mathbf{x} = \mathbf{g}(\mathbf{t}).$$

Continuous latent variable models (cont.)

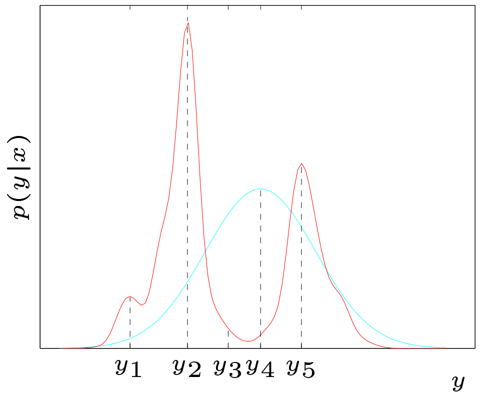
Latent variable model	*Prior in latent space* $p(\mathbf{x})$	*Mapping* $\mathbf{f}$	*Noise model* $p(\mathbf{t}\vert\mathbf{x})$	*Density in observed space* $p(\mathbf{t})$
Factor analysis (FA)	$\mathcal{N}(\mathbf{0}, \mathbf{I})$	linear	diagonal normal	constrained Gaussian
Principal component analysis (PCA)	$\mathcal{N}(\mathbf{0}, \mathbf{I})$	linear	spherical normal	constrained Gaussian
Generative topographic mapping (GTM)	discrete uniform	generalised linear model	spherical normal	constrained Gaussian mixture[a]
Independent component analysis (ICA)	unknown but factorised	linear	Dirac delta	depends

[a] The accuracy can be improved by using more components without risk of overfitting (but at a computational cost).

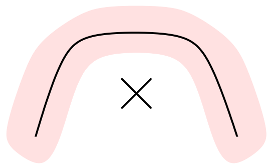
Problems of latent variable models:

- The latent dimension L must be fixed in advance—could use model selection.
- The complexity of GTM is $\mathcal{O}(e^L)$.

Deriving a functional relationship from a conditional distribution



- Consider a conditional distribution $f(y) \stackrel{\text{def}}{=} p(y|x=x_0)$.
- The **entropy** $h(f) \stackrel{\text{def}}{=} -\int f(y) \log f(y)\, dy$ tells how informative the distribution is: flat (high entropy) or (multiply) peaked (low entropy).
- If the entropy is low, we can pick representative points of $p(y|x)$:
  - Single choice (*univalued mapping*):
    - $*$ Mean: may not be good if the solution domain is not convex.
    - $*$ Global mode: not much better than the other modes.
  - Multiple choice (*multivalued mapping*): select all the modes; one of them should be the correct one—knowing which one requires additional information.



- For unimodal, symmetric distributions (e.g. factor analysis) nothing of this matters!

Mixtures of Gaussians: conditional distributions and entropy

- Consider a D-dimensional mixture of Gaussians with M components:

$$p(\mathbf{t}) = \sum_{m=1}^{M} p(m)p(\mathbf{t}|m) \qquad p(\mathbf{t}|m) \sim \mathcal{N}_D(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \qquad \mathbf{t} \in \mathbb{R}^D.$$

- Any conditional distribution of a mixture of Gaussians is a mixture of Gaussians itself, trivially computable if the components are diagonal (where many mixture components can be discarded):

$$p(\mathbf{t}_\mathcal{J}|\mathbf{t}_\mathcal{I}) = \sum_{m=1}^{M} \underbrace{p(m|\mathbf{t}_\mathcal{I})}_{\text{negligible for most } m} \; p(\mathbf{t}_\mathcal{J}|m) \qquad p(\mathbf{t}_\mathcal{J}|m) \sim \mathcal{N}_J(\boldsymbol{\mu}_{m\mathcal{J}}, \boldsymbol{\Sigma}_{m\mathcal{J}\mathcal{J}}) \qquad \mathbf{t}_\mathcal{J} \in \mathbb{R}^J.$$

- **Entropy**: no exact formula, but bounds: $\max(\mathsf{LB}_1, \mathsf{LB}_2) \le h(p(\mathbf{t})) \le \mathsf{UB}_1$, where:

$$\mathsf{LB}_1 \stackrel{\text{def}}{=} \frac{1}{2}\log\left\{(2\pi e)^D \prod_{m=1}^{M} |\boldsymbol{\Sigma}_m|^{\pi_m}\right\} \qquad \mathsf{UB}_1 \stackrel{\text{def}}{=} \frac{1}{2}\log\left((2\pi e)^D\,|\boldsymbol{\Sigma}|\right)$$

$$\mathsf{LB}_2 \stackrel{\text{def}}{=} -\log\left\{\sum_{m,n=1}^{M} p(m)p(n)\,|2\pi(\boldsymbol{\Sigma}_m+\boldsymbol{\Sigma}_n)|^{-\frac{1}{2}}\, e^{-\frac{1}{2}(\boldsymbol{\mu}_m-\boldsymbol{\mu}_n)^T(\boldsymbol{\Sigma}_m+\boldsymbol{\Sigma}_n)^{-1}(\boldsymbol{\mu}_m-\boldsymbol{\mu}_n)}\right\}$$

and $\boldsymbol{\Sigma} = \sum_{m=1}^{M} \pi_m \boldsymbol{\Sigma}_m + \sum_{m=1}^{M} \pi_m \boldsymbol{\mu}_m \boldsymbol{\mu}_m^T - \boldsymbol{\mu}\boldsymbol{\mu}^T$ is the mixture covariance.

Mixtures of Gaussians: exhaustive mode finding

- Aim: find *all the modes* of the Gaussian mixture $p(\mathbf{t})$, $\mathbf{t} \in \mathbb{R}^D$ (no direct methods exist).
- $p(\mathbf{t})$ has at most M modes and they all lie in the convex hull of the mean vectors $\{\boldsymbol{\mu}_m\}_{m=1}^M$.
- Algorithm: use an iterative maximisation method starting from each mean vector:
  - *Gradient descent*: $\mathbf{t} = \mathbf{t}^{(\tau)} + s\mathbf{g}(\mathbf{t}^{(\tau)})$ with the gradient

$$\mathbf{g} \stackrel{\text{def}}{=} \nabla p(\mathbf{t}) = \sum_{m=1}^M p(\mathbf{t}, m)\boldsymbol{\Sigma}_m^{-1}(\boldsymbol{\mu}_m - \mathbf{t}).$$

  - *Quadratic maximisation*: $\mathbf{t}^{(\tau+1)} = \mathbf{t}^{(\tau)} - \mathbf{H}^{-1}(\mathbf{t}^{(\tau)})\mathbf{g}(\mathbf{t}^{(\tau)})$ with the Hessian

$$\mathbf{H} \stackrel{\text{def}}{=} (\nabla\nabla^T)p(\mathbf{t}) = \sum_{m=1}^M p(\mathbf{t}, m)\boldsymbol{\Sigma}_m^{-1}\left((\boldsymbol{\mu}_m - \mathbf{t})(\boldsymbol{\mu}_m - \mathbf{t})^T - \boldsymbol{\Sigma}_m\right)\boldsymbol{\Sigma}_m^{-1}.$$

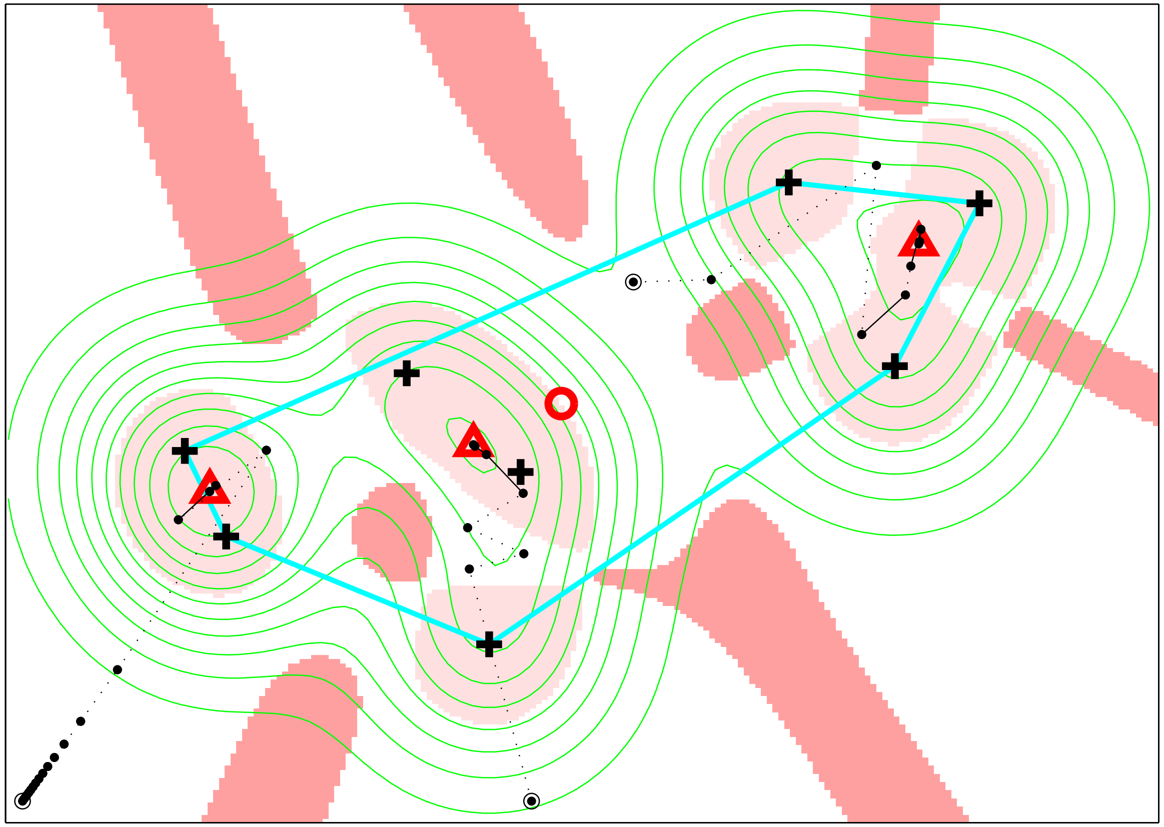
  - *Fixed-point iteration*: $\mathbf{t} = \mathbf{f}(\mathbf{t})$ with

$$\mathbf{f}(\mathbf{t}) \stackrel{\text{def}}{=} \left(\sum_{m=1}^M p(m|\mathbf{t})\boldsymbol{\Sigma}_m^{-1}\right)^{-1} \sum_{m=1}^M p(m|\mathbf{t})\boldsymbol{\Sigma}_m^{-1}\boldsymbol{\mu}_m.$$

  From the stationary points found, remove minima, saddle points and coincident maxima.

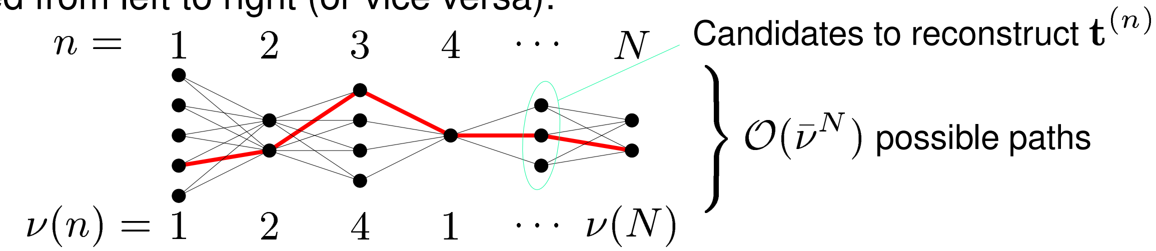
Mixtures of Gaussians: exhaustive mode finding (cont.)

- The formulas simplify considerably for $\mathbf{\Sigma}_m = \sigma^2\mathbf{I}$ (GTM's case).
- *Confidence sets*: approximate the pdf at each mode by a Gaussian (using the Hessian).



Global reconstruction: constraining the solutions

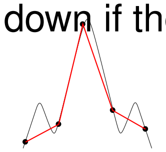
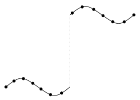
- Applying multiple pointwise reconstruction to each point in the sequence produces a *layered graph* which must be traversed from left to right (or vice versa):



- To select a unique candidate for each point we need external info. For *continuous sequential data*, we apply a *local continuity constraint*: consecutive points must be close to each other.
- Define a *distance* $\delta(\mathbf{x}, \mathbf{y})$, e.g. the weighted Euclidean distance $\sum_{d=1}^{D} w_d (x_d - y_d)^2$.
- Extend to a *global continuity constraint*: $\mathscr{C}\left(\{\mathbf{t}^{(n)}\}_{n=1}^{N}\right) = \sum_{n=1}^{N-1} \delta(\mathbf{t}^{(n)}, \mathbf{t}^{(n+1)})$. If δ is the Euclidean distance then $\mathscr{C}$ is the curve length: shortest path algorithm.
- In the mapping inversion case, add a *fitness* term based in the forward mapping $\mathbf{t}_{\mathcal{J}} \xrightarrow{\mathbf{g}} \mathbf{t}_{\mathcal{I}}$:

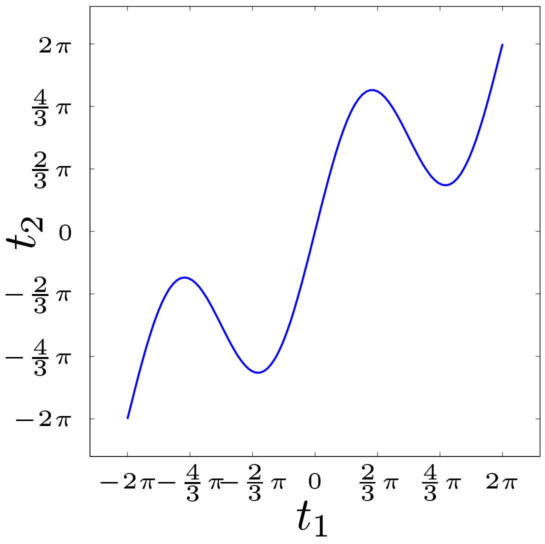
$$\mathscr{C}\left(\{\mathbf{t}^{(n)}\}_{n=1}^{N}\right) = \underbrace{\sum_{n=1}^{N-1} \delta(\mathbf{t}^{(n)}, \mathbf{t}^{(n+1)})}_{\text{continuity}} + \lambda \underbrace{\sum_{n=1}^{N} \Delta(\mathbf{t}_{\mathcal{I}}^{(n)}, \mathbf{g}(\mathbf{t}_{\mathcal{J}}^{(n)}))}_{\text{fitness}}.$$

Global reconstruction: constraining the solutions (cont.)

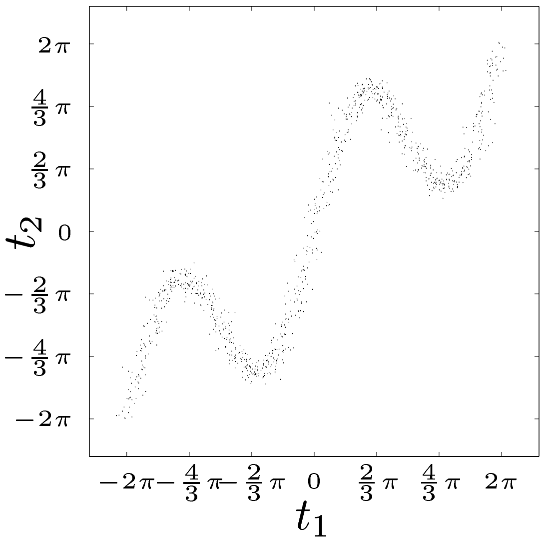
- The search problem $\boxed{\min_{\text{all combinations}} \mathscr{C}\left(\{\mathbf{t}^{(n)}\}_{n=1}^{N}\right)}$ can be efficiently solved by **dynamic programming** in $\mathcal{O}(N\bar{\nu})$ time. A suboptimal solution can be found even faster by a *greedy* algorithm.
- Additional information may be easily included, e.g. bounds on the missing variables given by the present ones.
- The dynamic programming search can break down if there are isolated *discontinuities*:
  - due to artifacts, such as *undersampling* 
  - due to the *intrinsic* nature of the data (e.g. fronts in wind fields). 
- Dynamic programming is independent of the speed at which the sequence is travelled (i.e., to reparametrisations of it) and so is robust to time warping (e.g. fast/slow speech).

Experiments with a toy problem

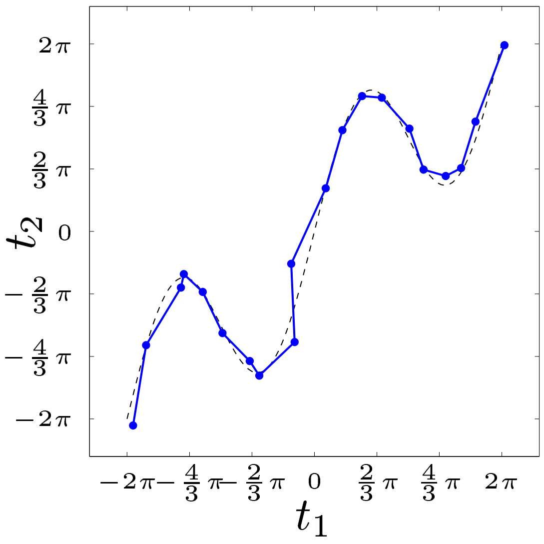
Consider one-dimensional data ($L = 1$) observed in $D = 2$ dimensions, i.e., $\mathbf{t} = (t_1, t_2)$:



Curve $\mathbf{t} = (x, x + 3\sin x)$ for $x \in [-2\pi, 2\pi]$.



Point cloud $\{\mathbf{t}_n\}_{n=1}^N$, sampled from the curve with additive $\mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ noise.



Sample noisy trajectory $\{\mathbf{t}^{(n)}\}_{n=1}^N$.

Univalued forward mapping $t_2 = g(t_1) = t_1 + 3\sin t_1$ but multivalued inverse $t_1 = g^{-1}(t_2)$.

Masks

P0 Complete sequence (0% missing)

$n=1$	2	3	4	5	6	7	8	9	10	n	N	
-6.1	-5.6	-4.5	-4.4	-3.7	-3.1	-2.2	-1.8	-0.7	-0.8		6.4	t_1
-6.5	-3.8	-1.9	-1.4	-2.0	-3.4	-4.3	-4.8	-3.7	-1.1		6.2	t_2

P1 t_2 always missing: regression $t_1 \to t_2$ (50% missing)

-6.1	-5.6	-4.5	-4.4	-3.7	-3.1	-2.2	-1.8	-0.7	-0.8		6.4	t_1
												t_2

P2 t_1 always missing: regression $t_2 \to t_1$ (50% missing)

												t_1
-6.5	-3.8	-1.9	-1.4	-2.0	-3.4	-4.3	-4.8	-3.7	-1.1		6.2	t_2

P3 t_1 or t_2 missing at random (75% missing)

			-4.4				-1.8					t_1
-6.5		-1.9				-4.3	-4.8				6.2	t_2

P4 t_1 or t_2 missing at random (50% missing)

-6.1	-5.6	-4.5		-3.7		-2.2	-1.8		-0.8		6.4	t_1
-6.5				-2.0	-3.4				-1.1			t_2

P5 t_1 or t_2 missing at random (25% missing)

-6.1	-5.6	-4.5	-4.4		-3.1	-2.2	-1.8		-0.8		6.4	t_1
-6.5	-3.8		-1.4		-3.4	-4.3		-3.7	-1.1		6.2	t_2

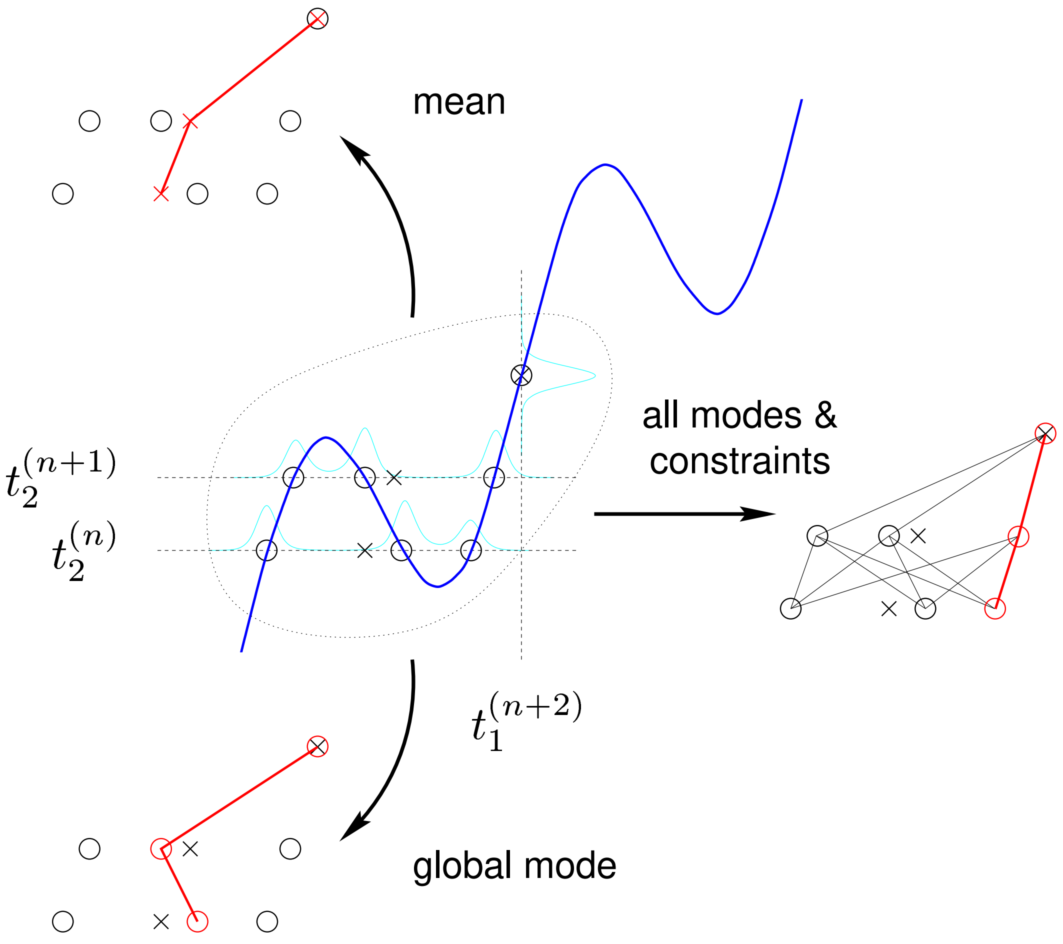
P6 One of $\{t_1, t_2\}$ missing at random (50% missing)

		-4.5		-3.7	-3.1	-2.2		-0.7			6.4	t_1
-6.5	-3.8		-1.4				-4.8		-1.1			t_2

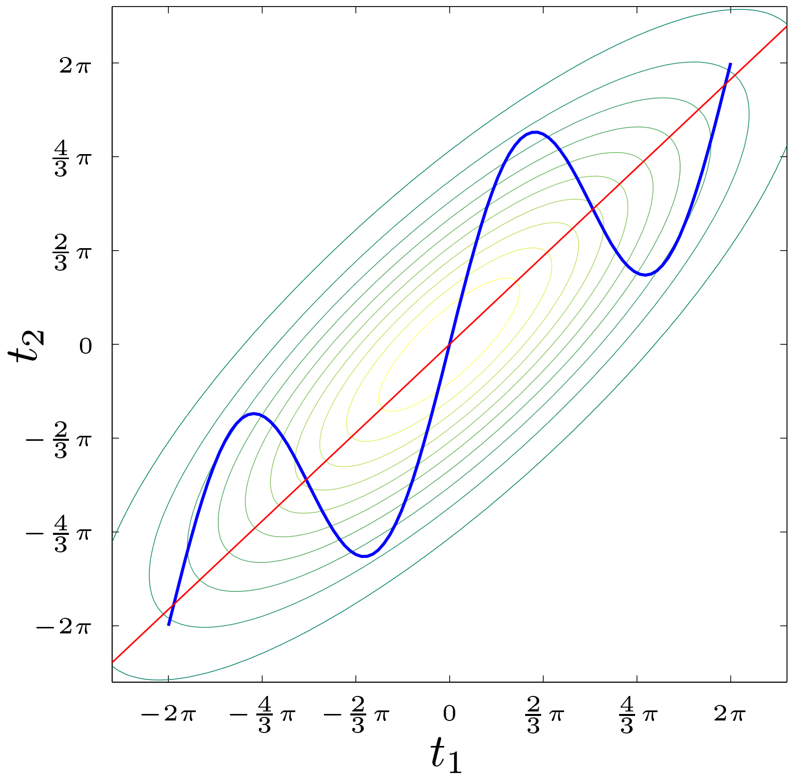
P7 Random blocks of missing data (10% missing)

			-4.4	-3.7	-3.1		-1.8	-0.7	-0.8		6.4	t_1
-6.5	-3.8	-1.9	-1.4	-2.0	-3.4		-4.8	-3.7	-1.1		6.2	t_2

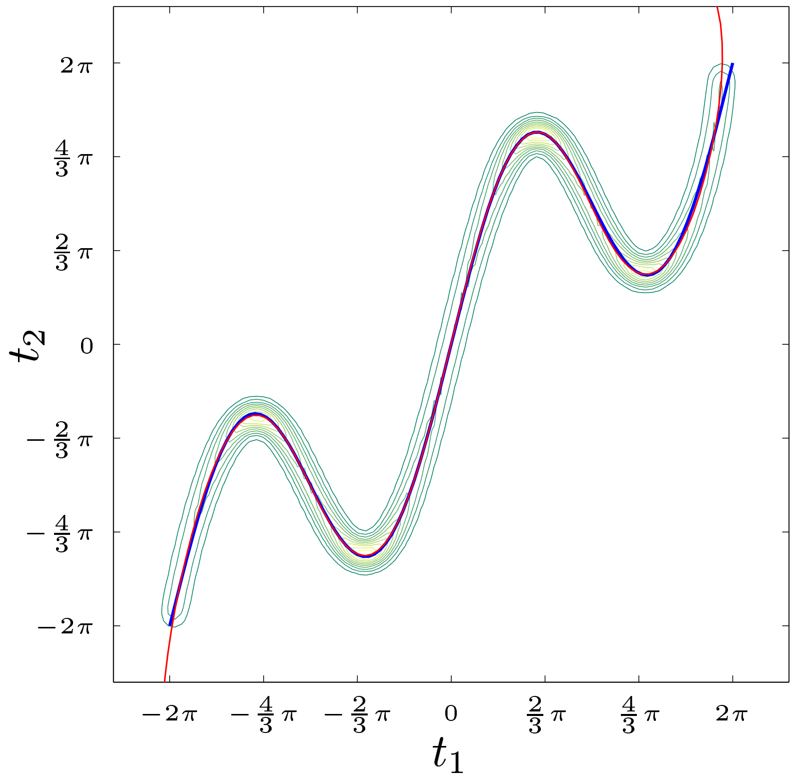
Demonstration of the method



Density models used

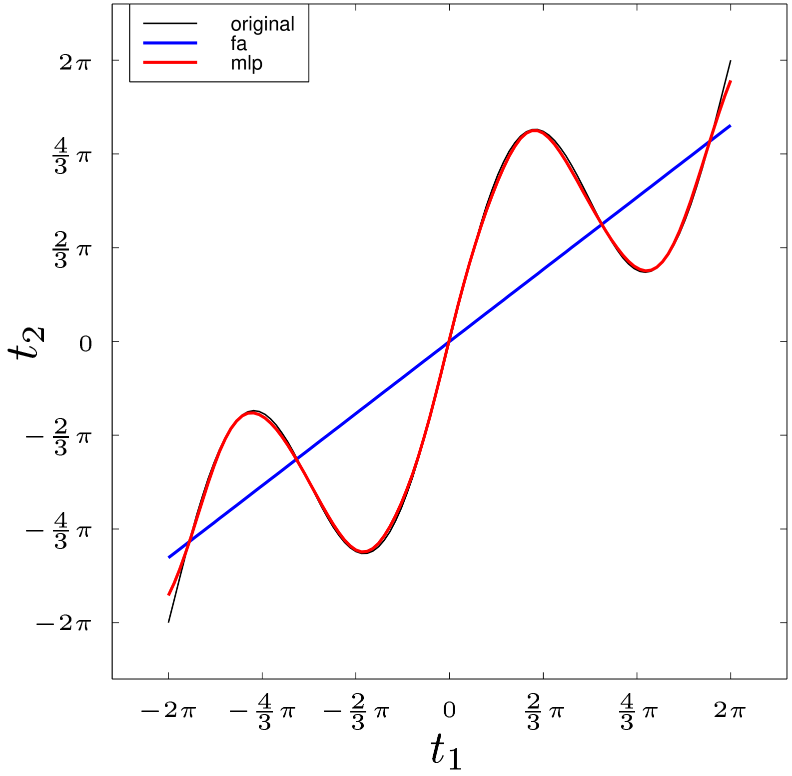


Factor analysis: $L = 1$, 6 parameters, log-likelihood $= -4807$

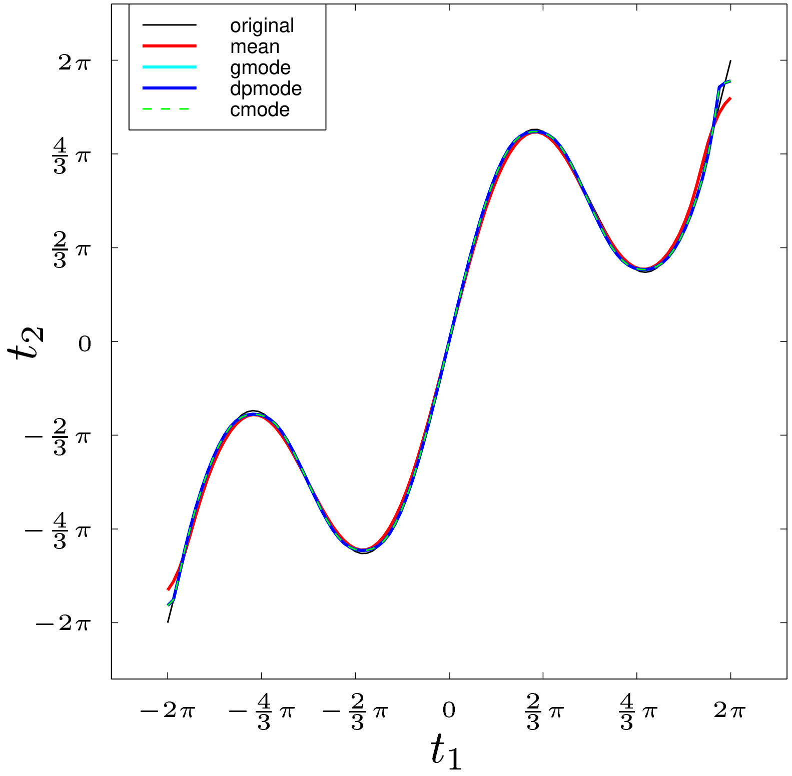


GTM: $L = 1$, $K = 200$, 21 parameters, log-likelihood $= -3109$

Reconstruction results: forward problem (mask P1)

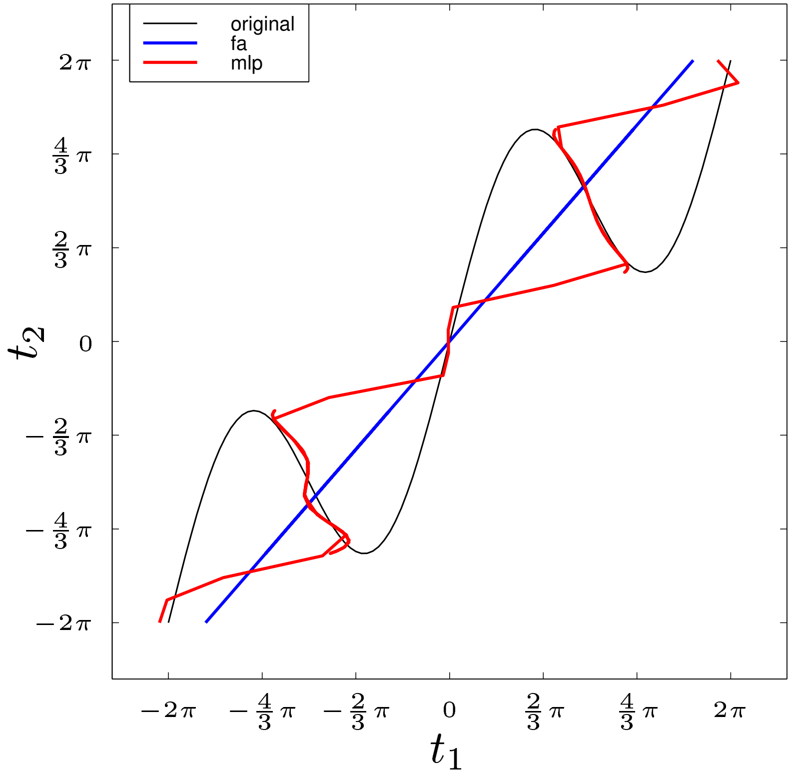


Factor analysis and multilayer perceptron

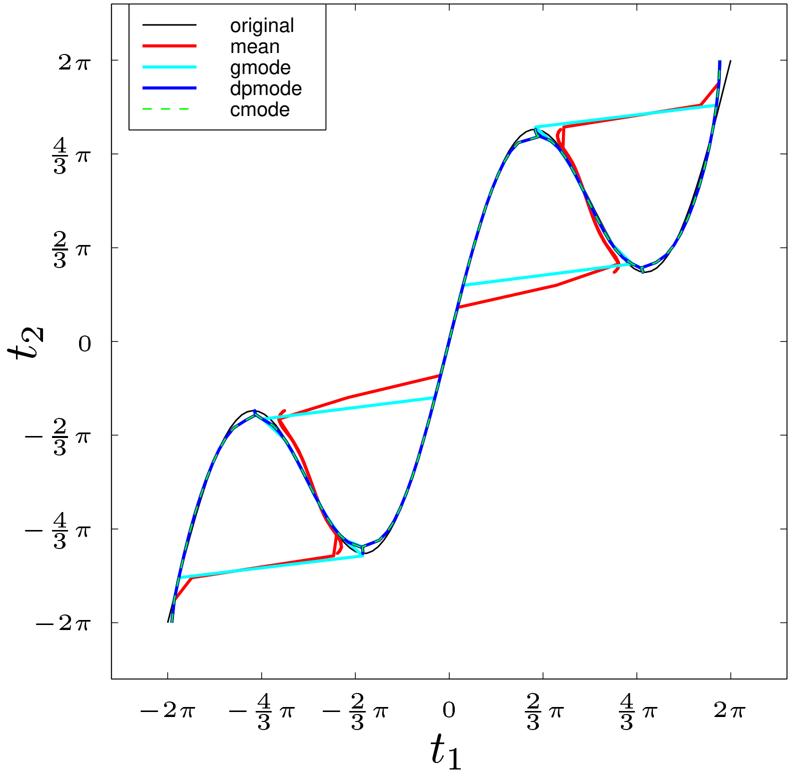


GTM with $K = 200$, various methods

Reconstruction results: inverse problem (mask P2)

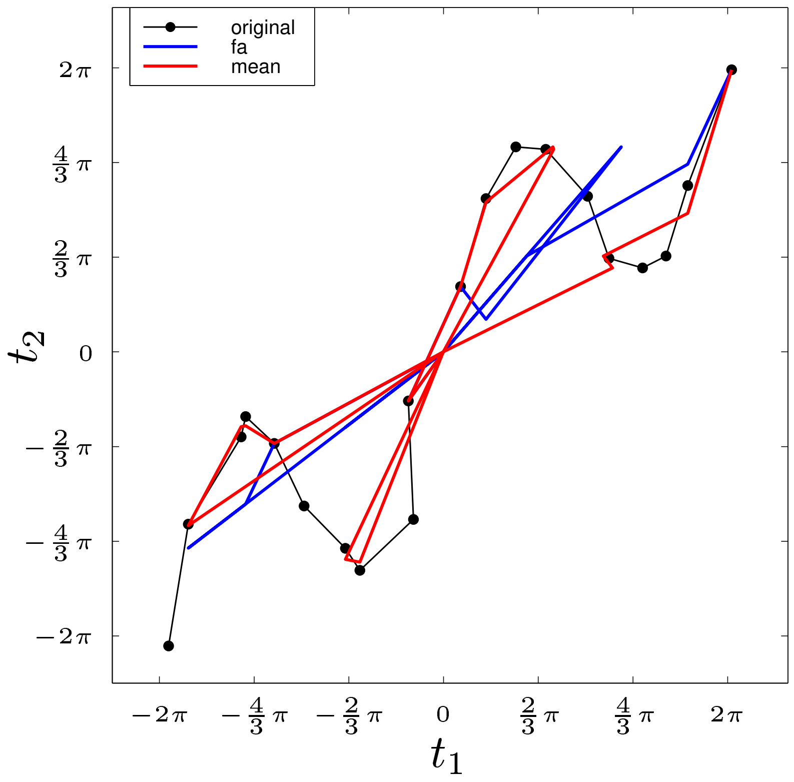


Factor analysis and multilayer perceptron

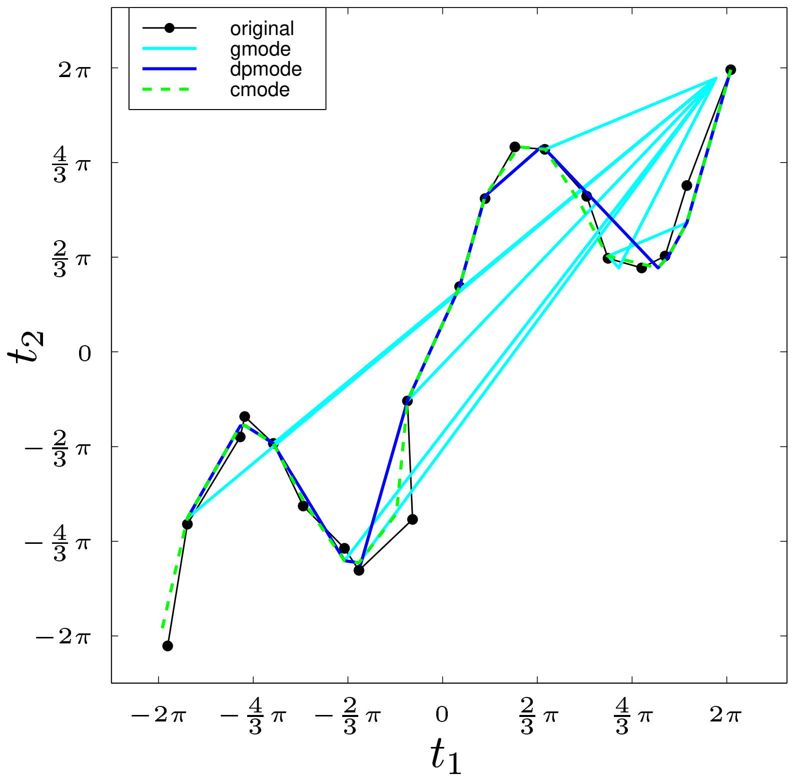


GTM with $K = 200$, various methods

Rec. results: noisy trajectory, 50% missing data (mask P4)



Factor analysis and GTM$_{K=200}$ (`mean`)



GTM$_{K=200}$ (`gmode`, `dpmode`, `cmode`)

NB: the multilayer perceptron can't deal with varying patterns of missing data.

Reconstruction results: average squared error

Average squared error $\frac{1}{N}\sum_{n=1}^{N}\left\|\mathbf{t}_n - \hat{\mathbf{t}}_n\right\|^2$ for a sequence with $N = 100$ points.

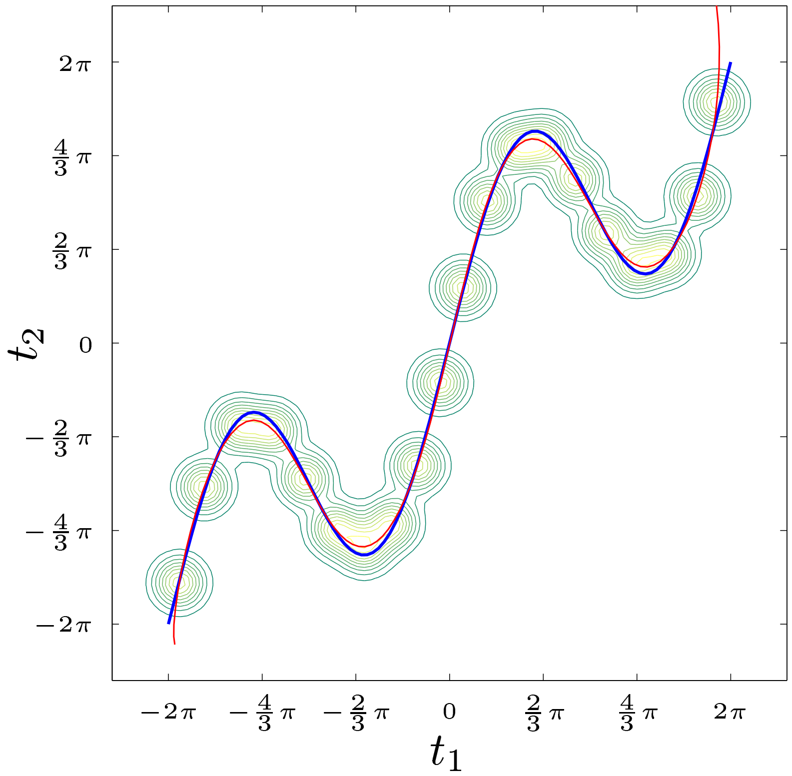
mask (% missing)	Factor analysis	MLP	GTM with $K = 200$				
			`mean`	`gmode`	`rmode`	`cmode`	`dpmode`
P1 (50%)	3.8011	0.0129	0.0196	0.0120	0.0120	0.0120	0.0120
P2 (50%)	4.2702	2.1475	2.1184	2.0878	7.7086	0.0129	0.0129
P3 (76%)	15.5385		14.6874	62.8203	36.1892	0.0069	0.1936
P4 (56%)	11.4116		9.7848	31.4508	14.8545	0.0058	0.0746
P5 (25%)	2.9049		1.7891	6.5224	3.1629	0.0040	0.0066
P6 (50%)	4.1377		0.9555	0.8104	4.7174	0.0122	0.0122
P7 (8%)	1.8476		1.5252	5.0694	2.5553	0.0007	0.0029

Problem type (mask)	Low error $\longrightarrow$ High error
Forward (P1)	$\texttt{cmode} = \texttt{dpmode} = \texttt{mean} = \text{MLP} = \texttt{gmode} = \texttt{rmode} < \text{FA}$
Inverse (P2)	$\texttt{cmode} \lesssim \texttt{dpmode} \ll \texttt{mean} = \text{MLP} < \texttt{gmode} < \text{FA} < \texttt{rmode}$
General (P3-P7)	$\texttt{cmode} < \texttt{dpmode} \ll \texttt{mean} \ll \texttt{gmode} = \texttt{rmode} = \text{FA}$

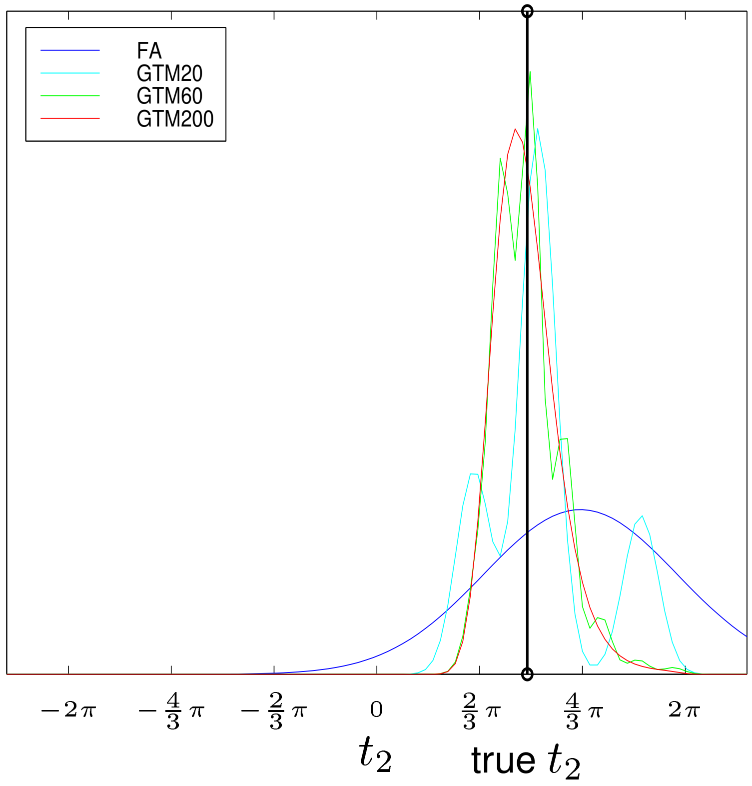
Reconstruction results: average squared error (cont.)

- `gmode` can become `rmode` depending on the implementation of the maximisation algorithm.
- Other mode-based strategies:
  - Greedy algorithm implementation of constraint: large error.
  - Picking S samples from the conditional distribution: large error.
  - Combination of `mean` (if the conditional distribution is unimodal) and `dpmode` (if it is multimodal): nearly no change over `dpmode`.
- When does `dpmode` break down?
  - For *non-sequential data* (e.g. a shuffled sequence): the continuity condition is not satisfied.
  - When the number of points in the sequence N is very large compared with the data noise: *oversampling* (for regression-type problems only).
  - When the joint density model is inaccurate, notably when it lacks *smoothness* (for regression-type problems only).

Smoothness of the density model

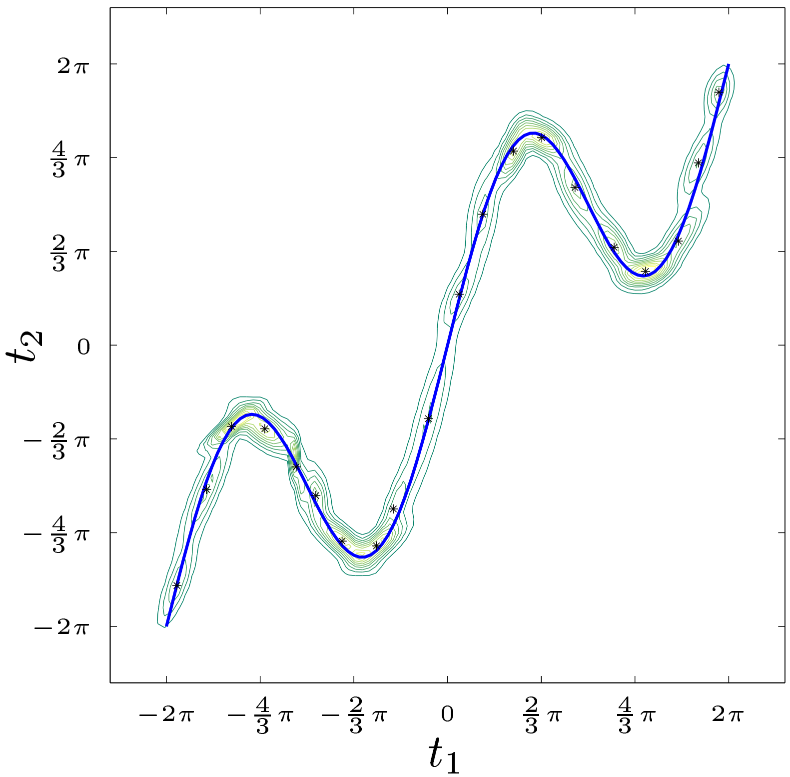


GTM: $L = 1$, $K = 20$, 21 parameters, log-likelihood $= -3468$

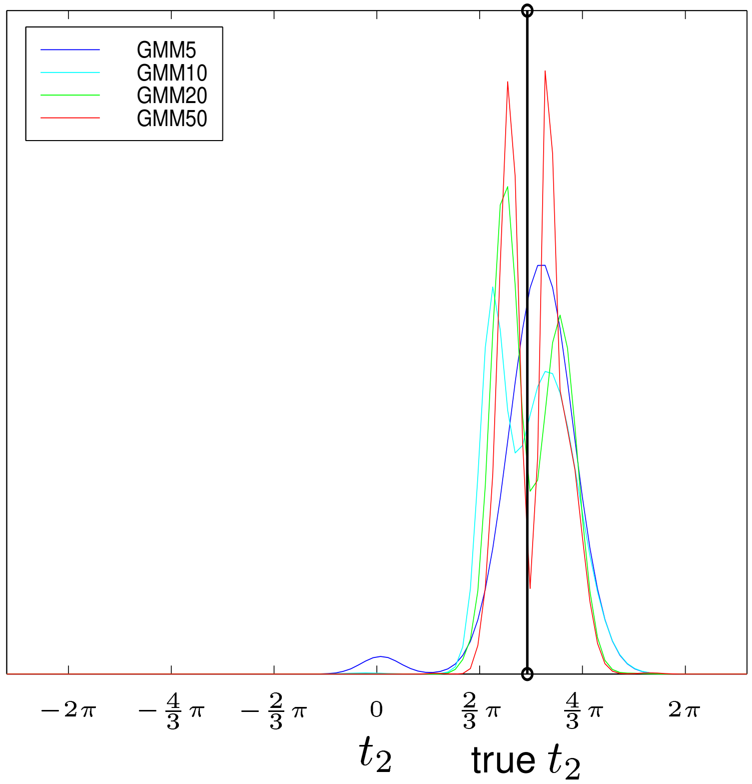


Conditional distribution $p(t_2|t_1 = 5.39)$ for each model

Smoothness of the density model (cont.)

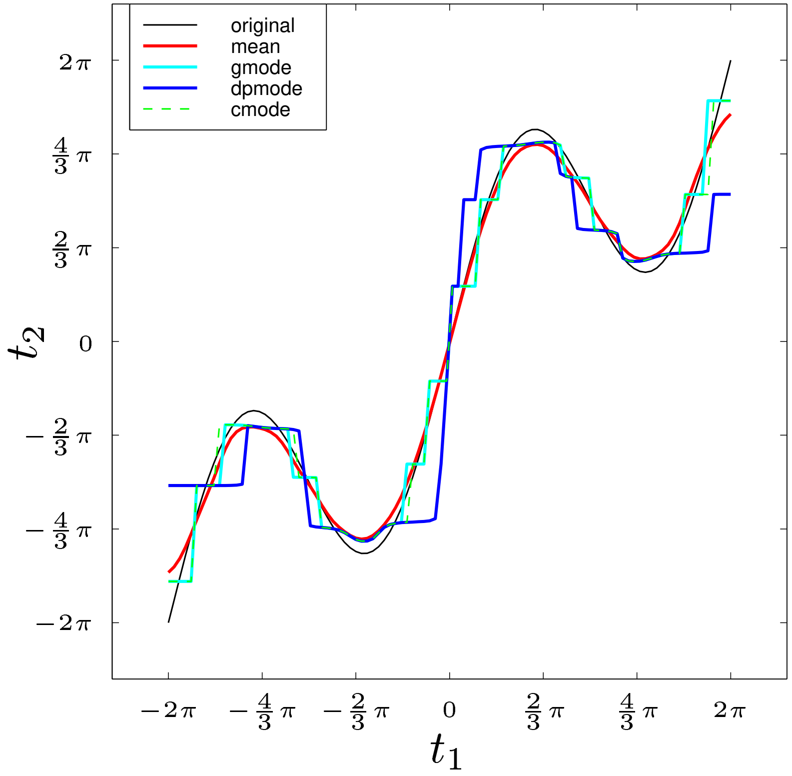


Mixture of $M = 20$ **full-covariance Gaussians**: 119 parameters, log-likelihood $= -3053$

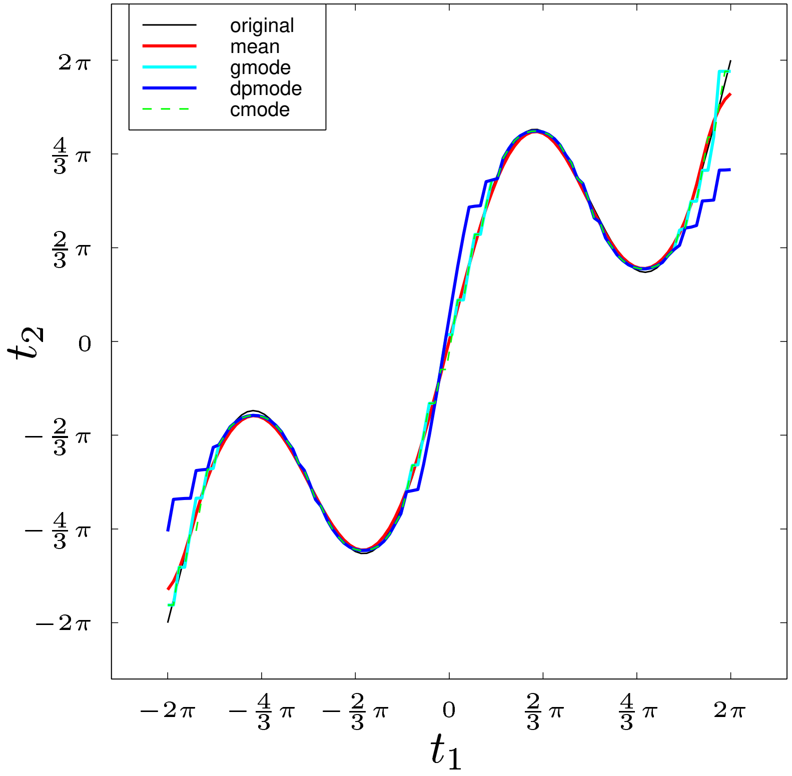


Conditional distribution $p(t_2|t_1 = 5.39)$ for various mixtures

Reconstruction results: forward problem (mask P1)



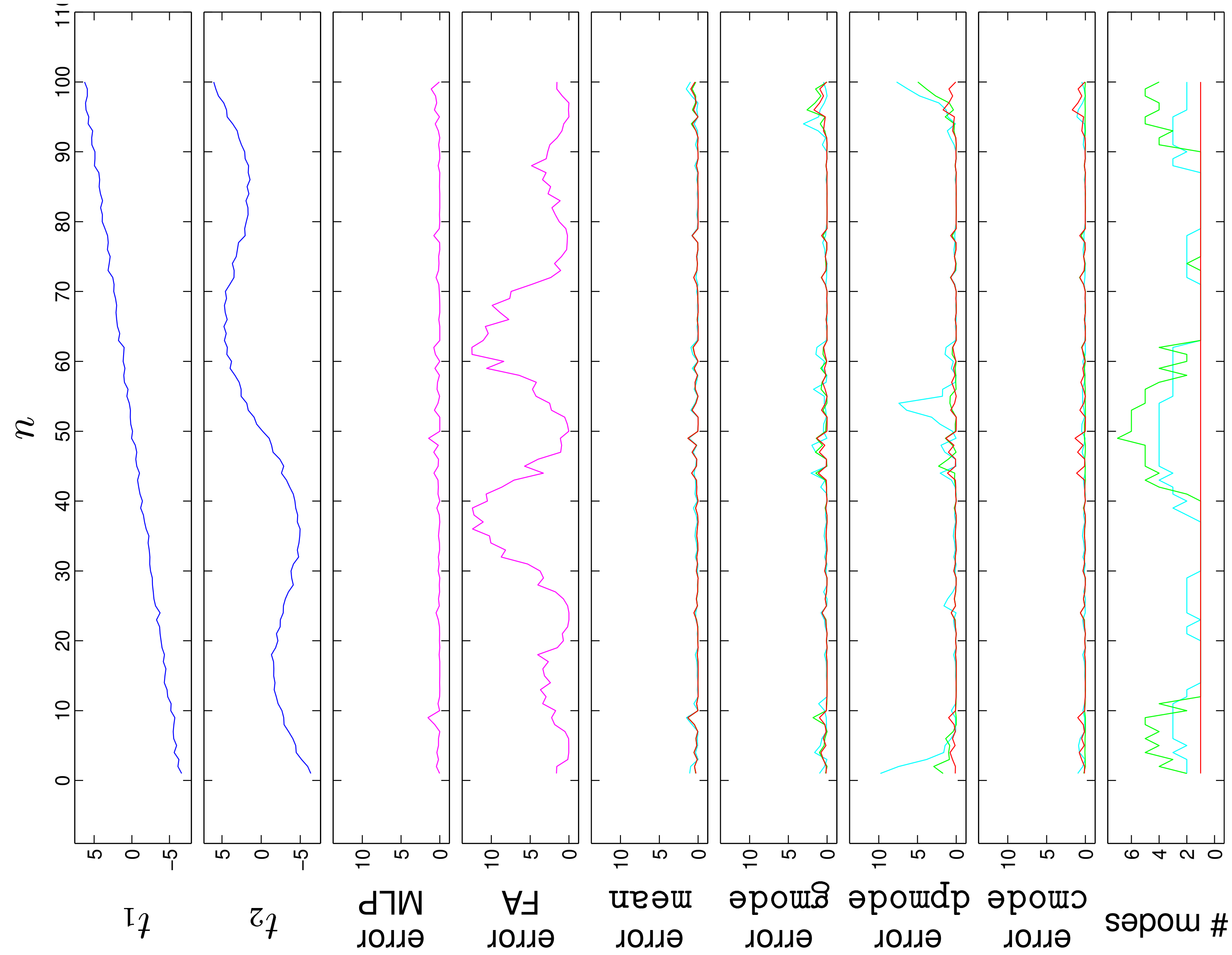
GTM with $K = 20$, various methods



GTM with $K = 60$, various methods

Rec. error: forward problem (mask P1)

GTM models: $K = 20$ (cyan), $K = 60$ (green) and $K = 200$ (red).

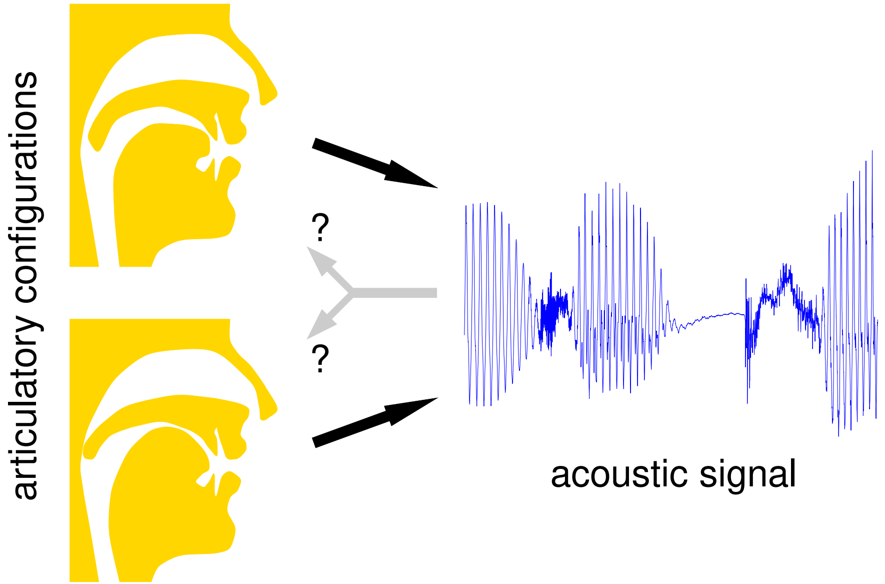


Smoothness of the density model: discussion

- The modes are very sensitive to the joint density model, while the mean is more robust.
- If the *density model is not smooth*, the conditional distribution presents *spurious modes* which may give rise to wrong solutions of the dynamic programming search.
- In this case, the reconstruction error for regression problems (P1, P2) usually exceeds that of the conditional mean. For general patterns of missing data (P3-P7) the error increase is small.
- Removing “low-probability” modes does not help.
- For GTM, increasing the number of mixture components K smoothens the density model at a higher computational cost. For mixtures of full-covariance Gaussians, overfitting limits the number of components.

The acoustic-to-articulatory mapping problem

The positions, speeds, etc. of supraglottal mechanical elements of the vocal tract (tongue, jaw . . .) are called *articulatory variables* and are continuous functions of the time.



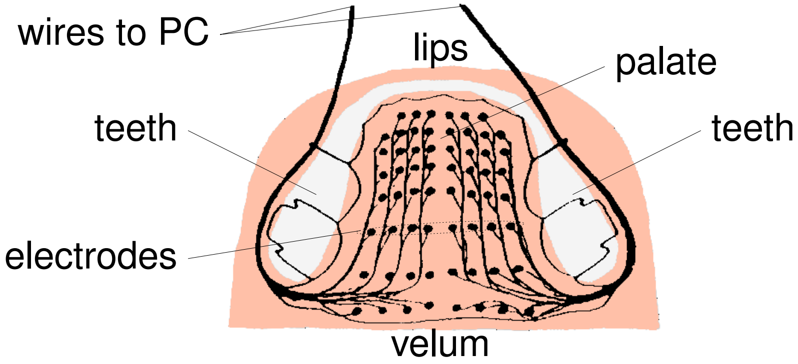
- ▷ A vocal tract configuration produces a unique acoustic signal. Thus the mapping articulatory $\longrightarrow$ acoustic is univalued (*forward problem*).
- ▷ An acoustic signal can be produced by different vocal tract configurations. The mapping acoustic $\longrightarrow$ articulatory is multivalued (*inverse problem*).

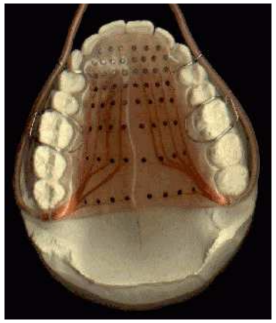
Approaches:

- ▷ Dynamic programming search in a large articulatory codebook: best performance, but very slow.
- ▷ Neural network (trained with the codebook): faster and more compact, but worse performance.
- ▷ Carefully prepared assembly of neural networks: approaches codebook performance and is fast.

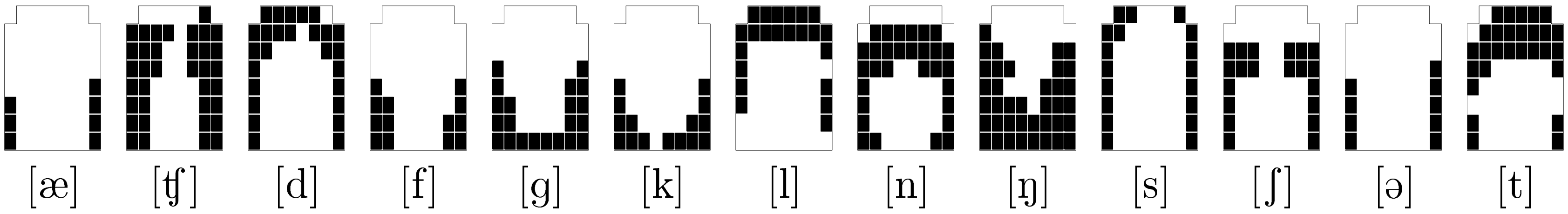
Electropalatography

A plastic pseudopalate fitted to a person's mouth detects the presence or absence of contact between the tongue and the palate in 62 different locations during an utterance.





The Reading pseudopalate



Representative EPG frames for the typical stable phase of different phonemes, pictured rowwise from alveoli (top) to velum (bottom).

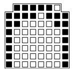
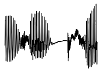
Prediction of PLP coefficients and EPG patterns

Two characteristics of the EPG variables:

▷ They are an incomplete representation of the vocal tract.

▷ They are binary—but temporal continuity as proximity between consecutive frames still applies.

The dataset

▷ 62-dimensional EPG patterns  plus 12-dimensional PLP coefficients  sampled at 200 Hz. Both mappings, EPG $\rightarrow$ PLP and PLP $\rightarrow$ EPG, are one-to-many.

▷ 10 shuffled utterances from the ACCOR database for speaker RK were used for training (over 7500 74-dimensional vectors). 4 utterances were used for testing.

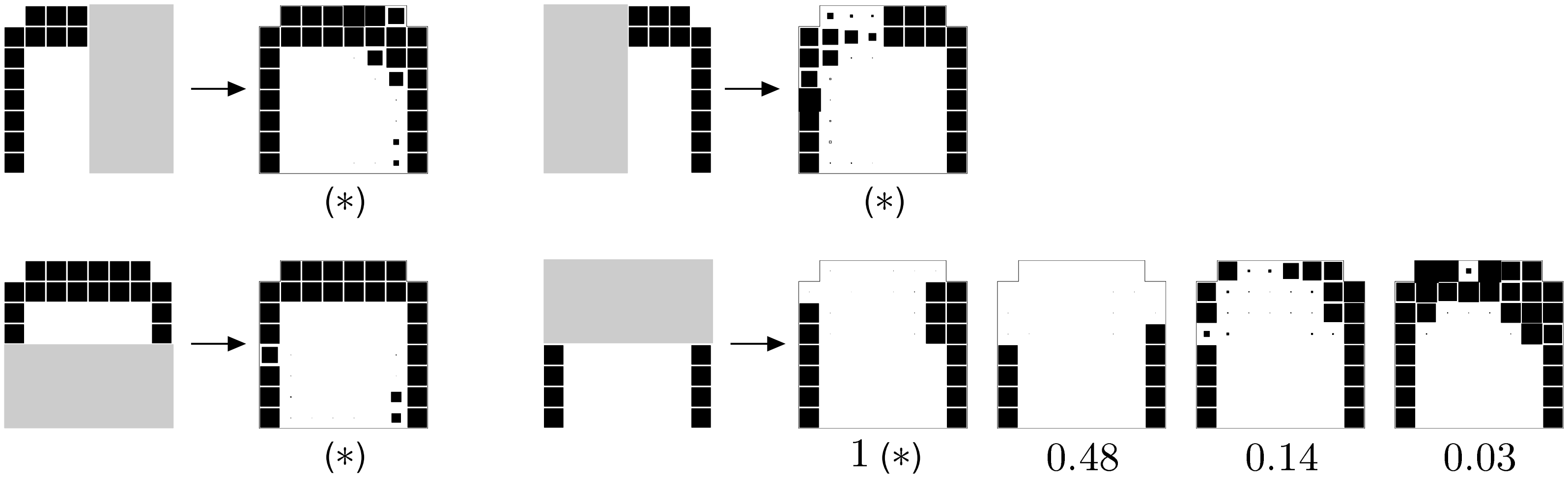
▷ The silence intervals at the beginning and end (but not inside) of each utterance were removed.

The models

▷ Factor analysis with a latent space of dimension $L = 9$ (total: 825 parameters).

▷ GTM with a latent space of dimension $L = 2$, a 30×30 latent grid and a 7×7 RBF grid (total: 3751 parameters).

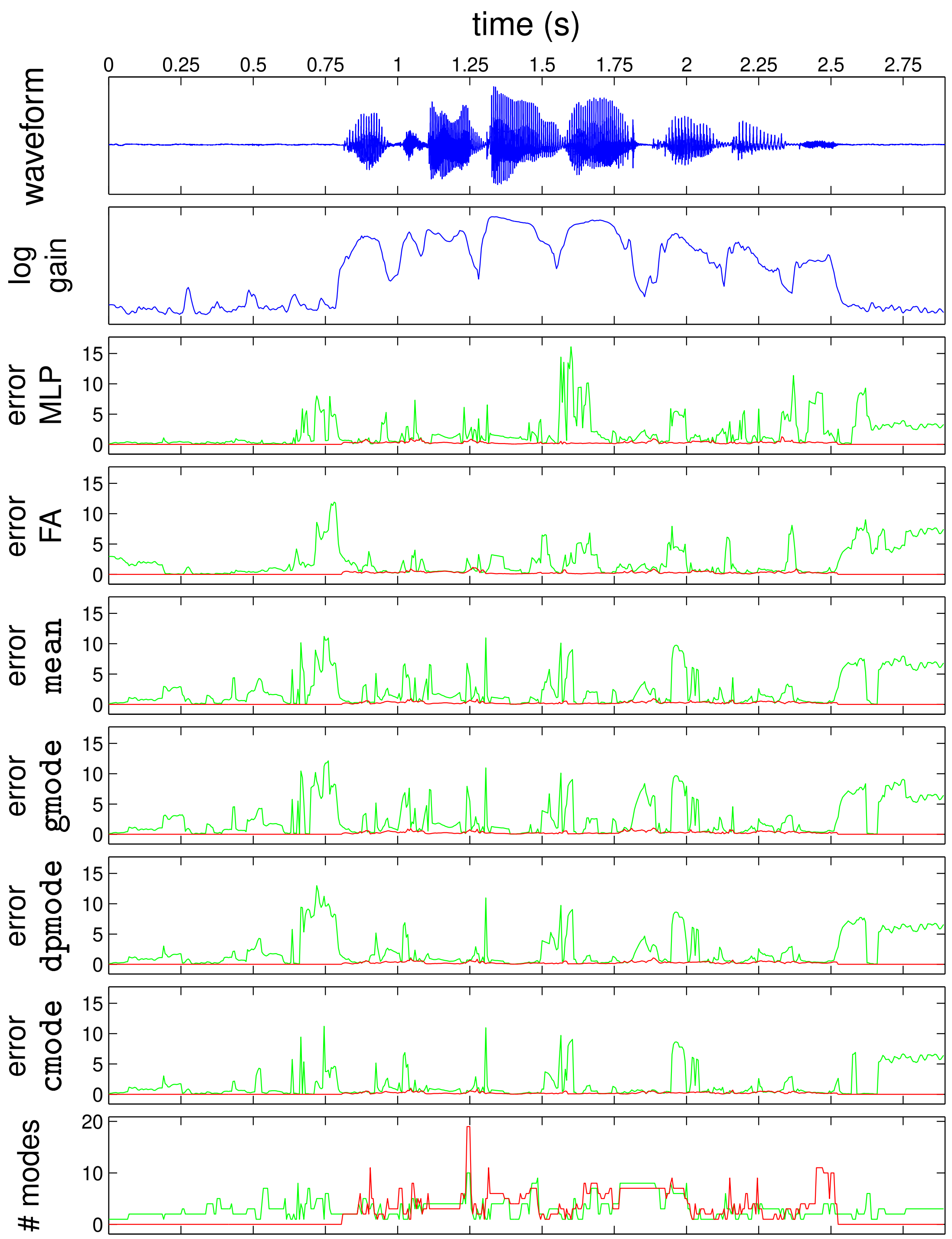
Reconstruction of single EPG frames



Use of the conditional distribution modes to predict, or reconstruct, variables in observed space. Here, we use the GTM model to compute the distribution of the EPG part greyed out (the unknown values) conditional on the EPG part which is not greyed out (the known values). The modes are given to the right of the arrow, labelled with their normalised probability if there is more than one mode. In all four cases, the mean (marked $*$) coincided approximately with one of the modes.

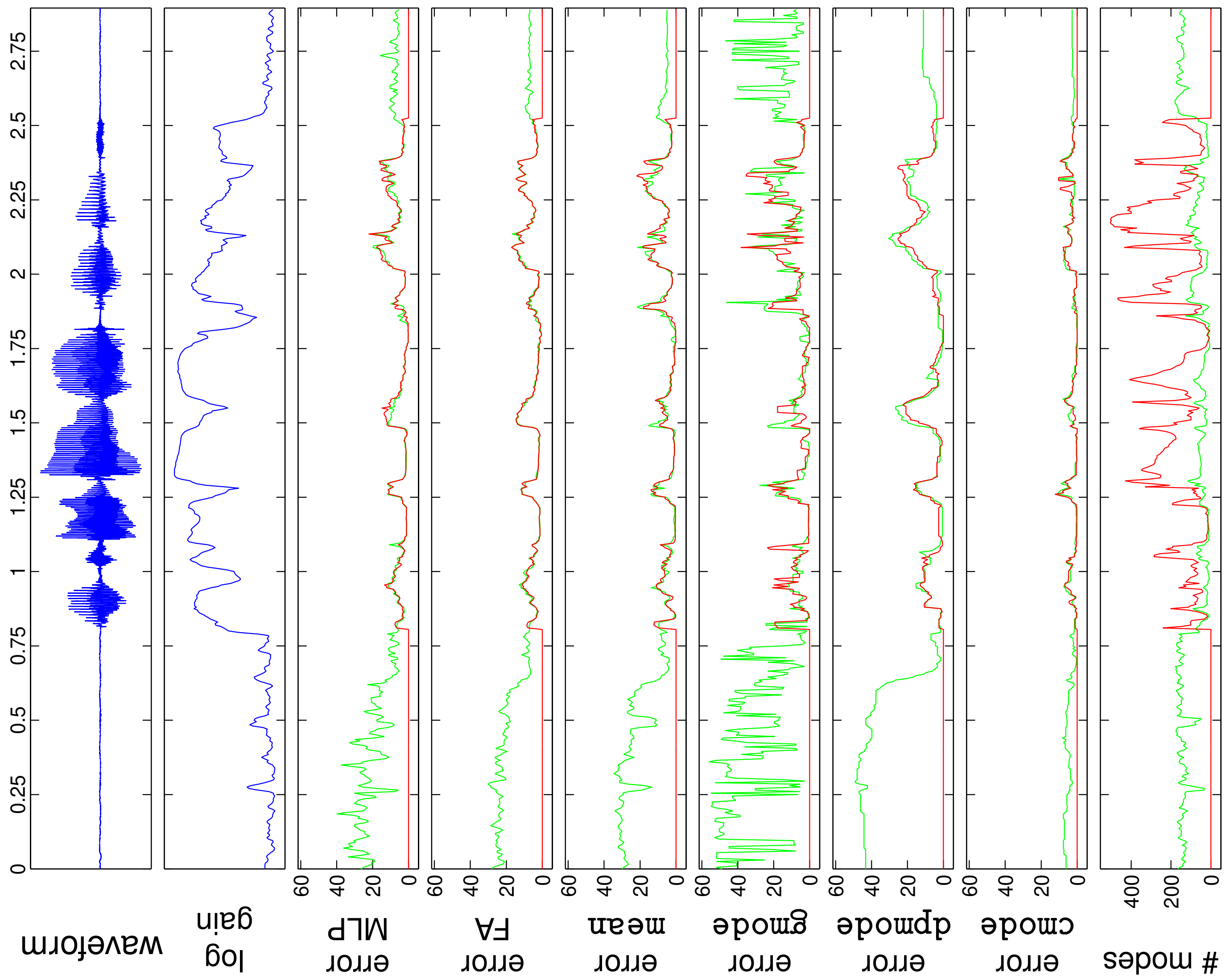
Rec. error: EPG → PLP (mask P1)

Utterance “We tore down the outbuildings” reconstructed by models trained without (red) and with silence and log gain (green).



Rec. error: PLP → EPG (mask P2)

Utterance "We tore down the outbuildings" reconstructed by models trained without (red) and with silence and log gain (green).



Reconstruction results: average squared error

Average squared error $\frac{1}{N}\sum_{n=1}^{N}\left\|\mathbf{t}_n - \hat{\mathbf{t}}_n\right\|^2$ for utterance "We tore down the outbuildings" with $N = 579$ points, without start and end silence intervals and without gain.

mask (% missing)	Factor analysis	MLP	GTM with $K = 900$				
			`mean`	`gmode`	`rmode`	`cmode`	`dpmode`
P1 (16%)	0.3185	0.3063	0.2843	0.3040	0.2983	0.2399	0.2821
P2 (84%)	5.7824	5.6305	5.5576	7.8321	9.2272	2.4032	8.4136
P3 (75%)	2.9885		3.1947	3.6480	4.0173	2.2315	2.4717
P4 (51%)	1.7364		1.8605	1.9608	2.1005	1.5297	1.6100
P5 (25%)	0.8000		0.8151	0.8498	0.8832	0.6948	0.7178
P6 (51%)	3.1221		3.0183	4.2263	5.0106	1.4511	1.6953
P7 (7%)	0.3889		0.4398	0.5347	0.5477	0.2100	0.2208

Problem type (mask)	Low error $\longrightarrow$ High error
EPG $\rightarrow$ PLP (P1)	$\texttt{cmode} \approx \text{MLP} < \texttt{dpmode} \lesssim \texttt{mean} \lesssim \texttt{gmode} \lesssim \text{FA} < \texttt{rmode}$
PLP $\rightarrow$ EPG (P2)	$\texttt{cmode} \ll \text{MLP} < \texttt{mean} \lesssim \text{FA} < \texttt{dpmode} < \texttt{gmode} < \texttt{rmode}$
General (P3-P7)	$\texttt{cmode} < \texttt{dpmode} \ll \texttt{mean} \lesssim \text{FA} \approx \texttt{gmode} < \texttt{rmode}$

Reconstruction results: average squared error (cont.)

- The continuity constraint was implemented with the Euclidean distance and no fitness term. Various weighted distances (e.g. zero weight for the PLP variables) did not alter the errors in general (less than 5%).
- Rounding the reconstructed EPG values to $\{0, 1\}$ increases considerably the error.
- Removing the gain from the density model and the silence intervals from the data reduces the error of all methods considerably. Reason: the gain is independent of the articulatory variables.
- Using a GTM model with fewer components ($K = 400$) increases the error of the mode-based strategies but does not affect much the `mean`.
- Factor analysis performs almost as well as the `mean`! Possible reason: the GTM model is inadequate due to the low latent dimensionality assumed ($L = 2$) for computational reasons.
- The number of modes per point can be very high for PLP $\rightarrow$ EPG, which indicates that the model is not smooth.

Summary

- General approach to reconstruction of sequential data based in:
  - Multiple pointwise reconstruction via a joint density model of the observed variables, using exhaustive mode finding in the conditional distributions (here implemented with latent variable models or Gaussian mixtures).
  - Sequence reconstruction via a continuity constraint on the set of pointwise candidates (here implemented with a weighted Euclidean distance and dynamic programming).

  The modes of the conditional distribution contain the *potentially* correct values. The continuity constraint—in those cases where it is applicable—may recover the *actually* correct ones.

Summary (cont.)

- Advantages:
  - Applicable to varying patterns of missing data: the observed variables are treated symmetrically, unlike methods based in function approximators or conditional distribution approximators.
  - Abstract: no physical knowledge of the problem is required, e.g. no need for a forward mapping—as long as powerful models are used.
  - Insensitive to time warping.
- Disadvantages:
  - Sensitivity to the smoothness of the density model for regression problems.
  - High computational cost at reconstruction time due to the mode-finding step.
  - Difficulty of density estimation in high dimensions (the curse of the dimensionality).
- Dynamical modelling: unlike hidden Markov models or Kalman filters, this approach does not model the temporal evolution of the data.

Computational considerations

- At reconstruction time, the computations required are:
  - Exhaustive mode finding in the conditional distribution: slow.
  - Dynamic programming search in the space of candidate reconstructions: fast.
- The mode search is slow when the dimensionality of the missing variables is high.
- For the acoustic-to-articulatory problem, reconstructing an utterance takes from some seconds (EPG $\rightarrow$ PLP, 16% data missing) to one hour (PLP $\rightarrow$ EPG, 84% data missing) in our Matlab implementation.
- Acceleration is possible by discarding low-probability mixture components in the conditional distribution ($4\times$ speedup can be obtained with a 10% reconstruction error increase).
- Using full-covariance mixtures reduces the computation time compared with GTM because the number of mixture components is limited by overfitting.

General probabilistic interpretation

- By interpreting the constraint in a probabilistic way, we can view the approach as a grand distribution $p(\mathbf{t}^{(1)}, \dots, \mathbf{t}^{(N)})$ in terms of:
  - The joint density model, independent of the sequence index n: $p(\mathbf{t})$.
  - A Markovian continuity constraint model (e.g. normal for the weighted Euclidean distance): $p(\mathbf{t}^{(n+1)}|\mathbf{t}^{(n)})$.

  The parameters of this grand distribution are fixed prior to the reconstruction stage.
- This grand distribution is both a pointwise generative model (the joint density model) and a sequence generative model.

Further work

- Other constraints:
  - *Continuity*: first-order finite differences.
  - *Smoothness*: second-order finite differences (stronger condition).
- Multidimensional constraints, e.g. reflecting spatial structure (1- to 3D) as well as temporal (1D).
- Unbounded horizon reconstruction: $N \to \infty$.
- Smooth density model estimation.
- Applications:
  - *Acoustic-to-articulatory mapping* using data from the Wisconsin X-ray microbeam database.
  - *Multimodal speech processing*: derive facial expression from acoustics (face animation) or vice versa (speech reconstruction).
  - *Hyppocampal place cell reconstruction*: obtain the trajectory of a rat moving in a 2D maze from the activity of an ensemble of hyppocampal place cells.
  - Oceanic *wind velocity vector retrieval* from satellite scatterometer data.