Uncertainty in compositional models of alignment

Ieva Kazlauskaite, University of Bath

Neill D.F. Campbell, University of Bath
Carl Henrik Ek, University of Bristol
Ivan Ustyuzhaninov, University of Tübingen
Tom Waterson, Electronic Arts

September, 2019

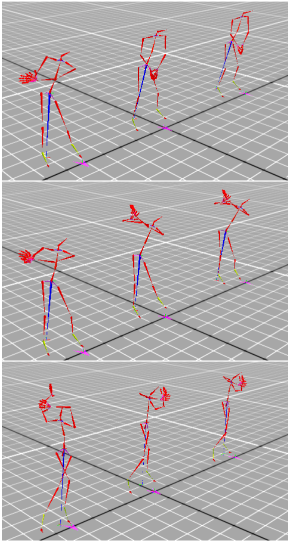
Motivation

Data:

- Motion capture sequences, e.g. a jump or a golf swing.
- Each motion corresponds to a different style or mood.

Goal: Generate new motions by interpolating between the captured clips.

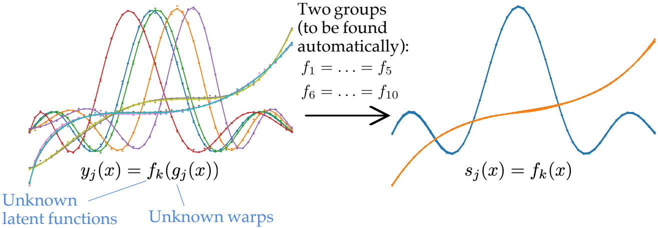
Pre-processing: The clips need to be temporally aligned.



Motivation

Assume we are given some time-series data with inputs $\mathbf{x} \in \mathbb{R}^N$ and J output sequences $\{\mathbf{y}_j \in \mathbb{R}^N\}$.
We know that there are multiple underlying function that generated this data, say K such functions, $f_k(\cdot)$, and the observed data was generated by warping the inputs to the true functions using some warping function $g_j(\mathbf{x})$ such that:

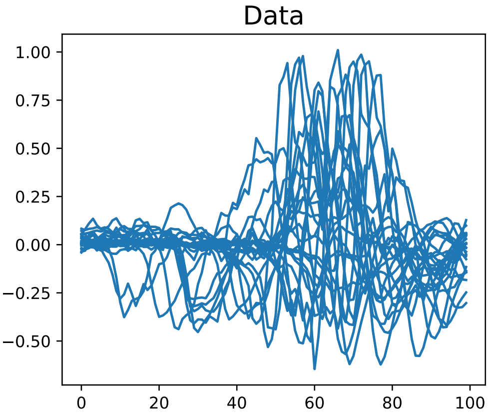
$$\mathbf{y}_j = f_k(g_j(\mathbf{x})) + \text{noise}. \tag{1}$$



Motivation

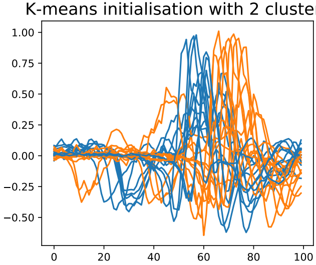
Unknowns:

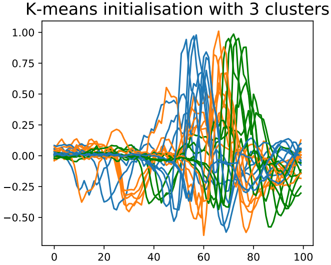
- Number of underlying functions K
- Underlying functions $f_k(\cdot)$
- Warps $g_j(\cdot)$ for each sequence



Motivation

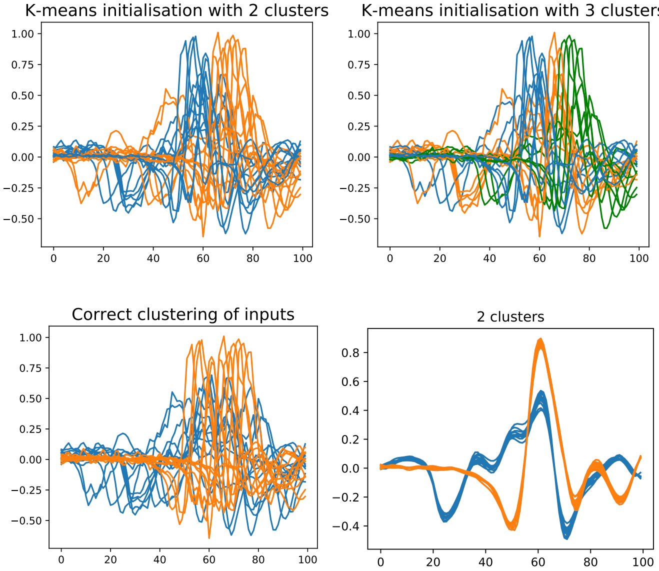
Let's try to find K using K-means clustering:





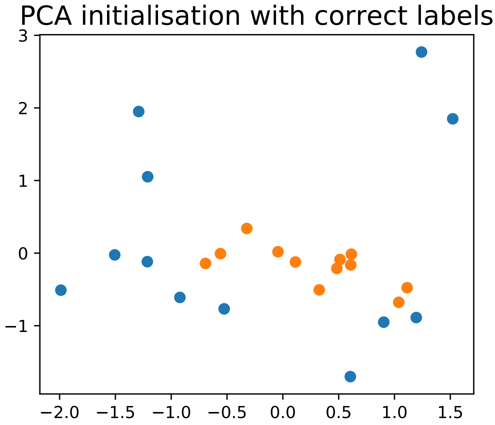
Motivation

K-means clustering vs. correct labels:



Motivation

A PCA scatter plot of the data:

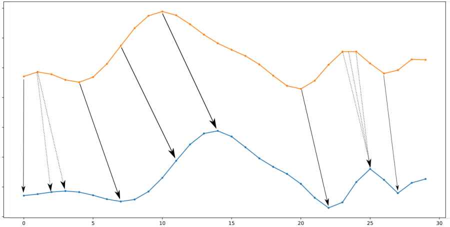


Alignment model

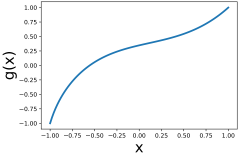
Three constituent parts:

- Model of transformations (warps), g_j
- Model of sequences, f_k
- Alignment objective

Model of transformations (warps)



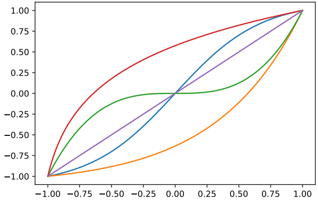
Observed sequences



Example warp

- Parametric warps.
  $\sum_{i \in I} w_i = 1, w_i \geq 0 \quad \forall\, i \in I$
- Nonparametric warps.
  For example, monotonic GPs

In general, we prefer warps that are close to an identity



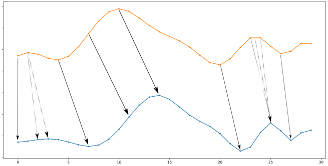
Riihimäki & Vehtari. Gaussian processes with monotonicity information (2010)
K. et al. Monotonic Gaussian Process Flow (2019)

Model of sequences

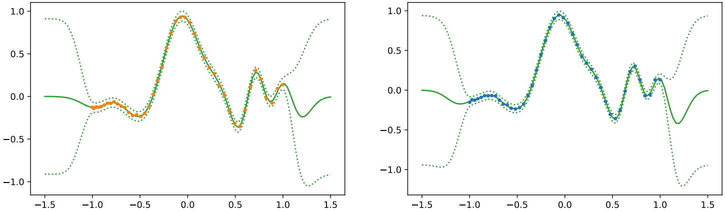
Option 1: interpolate sequences using linear interpolation or splines.

Option 2: fit GPs to the sequences.

- principled way to handle observational noise
- can impose priors of f_k



Observed sequences



GP regression

Notation

Assume that the observed data was generated as:

$$\mathbf{y}_j = f_k(g_j(\mathbf{x})) + \epsilon_j, \quad \epsilon_j \sim \mathcal{N}(0, \beta_j^- 1) \tag{2}$$

where $\mathbf{x}$ are fixed linearly spaced input locations (or evenly sampled time).

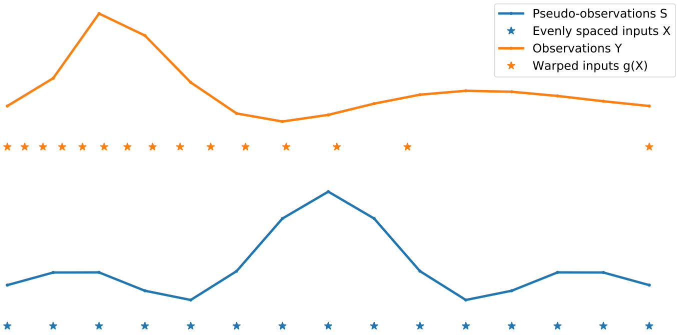
Then the corresponding aligned sequences are:

$$\mathbf{s}_j := f_k(\mathbf{x}) \tag{3}$$

The joint conditional likelihood is:

$$p\left(\begin{bmatrix}\mathbf{s}_j \\ \mathbf{y}_j\end{bmatrix} \middle| G_j, X_j, \theta_j\right) \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} k_{\theta_j}(X, X) & k_{\theta_j}(X, G_j) \\ k_{\theta_j}(G_j, X) & k_{\theta_j}(G_j, G_j) + \beta_j^{-1}\end{bmatrix}\right) \tag{4}$$

Model of sequences



Then the goal is to:

- Fit GPs to observations and pseudo-observations $\{[g(\mathbf{X}), \mathbf{X}], [\mathbf{Y}, \mathbf{S}]\}$ for each sequence
- Impose alignment constraint on pseudo-observations $\{\mathbf{X}, \mathbf{S}\}$

Alignment objective

We want an alignment objective that:

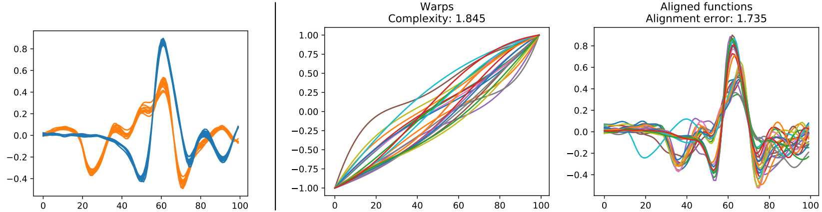
- infers the number of clusters (underlying functions) K
- aligns sequences within these clusters

We aim to design a clustering or dim. reduction objective that is invariant to the transformation (warps) of the inputs

Pairwise distance alignment objective

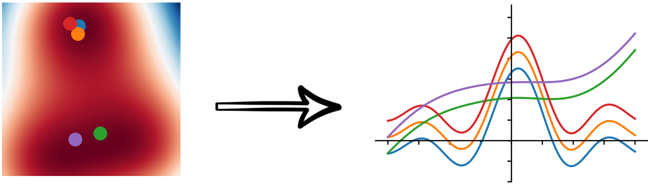
Minimise the pairwise distance between all sequences (irrespective of the underlying clusters of functions):

$$\mathcal{L} = \sum_{n=1}^{J} \sum_{m=n+1}^{J} ||\mathbf{s}_n(\mathbf{x}) - \mathbf{s}_m(\mathbf{x})||^2 \tag{5}$$



Traditional GP-LVM

- Observe high-dimensional data $\mathbf{S}$.
- Find low-dim representation $\mathbf{Z}$ that captures the structure of $\mathbf{S}$.
- Find a mapping $\mathbf{f}$ from $\mathbf{Z}$ to $\mathbf{S}$.



Latent space $\mathbf{Z}$ Mapping $\mathbf{h}$ Inputs $\mathbf{S}$

$$\mathbf{s}_j = \mathbf{h}\left(\, \mathbf{z}_j, \theta \,\right) \,+\, \text{noise},$$

where θ are parameters of $\mathbf{h}$.

Traditional GP-LVM

In a GP-LVM, GPs are taken to be independent across the features and the likelihood function is:

$$p(\mathbf{S} \mid \mathbf{x}) = \prod_{d=1}^{D} p(\mathbf{s}_d \mid \mathbf{x}) = \prod_{d=1}^{D} \mathcal{N}(\mathbf{s}_d \mid 0, K + \gamma^{-1} I) \qquad (6)$$

Observed data $\mathbf{Y}$ in matrix form

Aligned data $\mathbf{S}$ in matrix form

GP-LVM as alignment objective

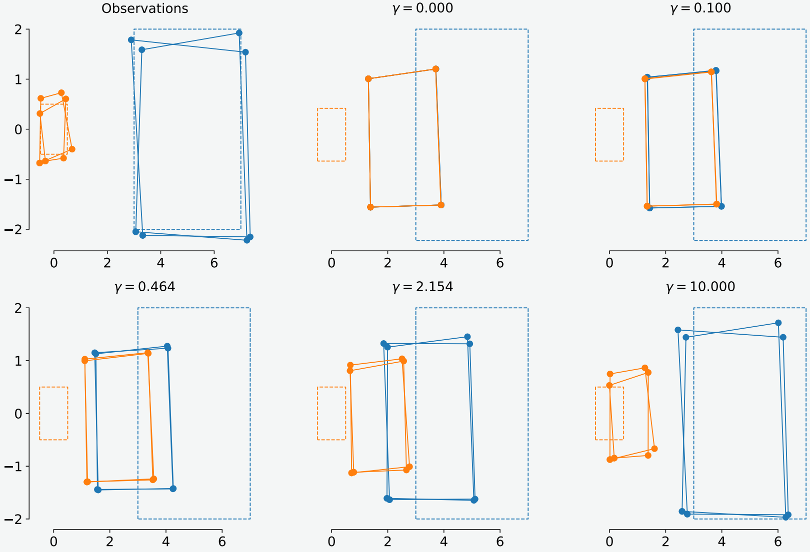
We impose the alignment objective by learning a low-dimensional representation $\mathbf{Z}$ of the pseudo-observations $\mathbf{S}$.

$$
\begin{aligned}
\mathcal{L}_{\text{GP-LVM}} &= \log p(\mathbf{S} \mid \mathbf{Z}, \theta_h, \theta_z, \beta) \\
&= \underbrace{\frac{N}{2} \log |\mathbf{K}_{zz}|}_{\text{complexity terms}} - \underbrace{\frac{1}{2} \mathrm{Tr}(\mathbf{K}_{zz}^{-1} \mathbf{S}\mathbf{S}^T)}_{\text{data fitting terms}} \\
&\quad + \underbrace{\log(p(\mathbf{Z} \mid \theta_z))}_{\text{prior over latent variables}} + \underbrace{\log(p(\theta_h))}_{\text{prior over GP mappings}} + \text{const}
\end{aligned} \tag{7}
$$

As an alignment objective, it is controlled by:

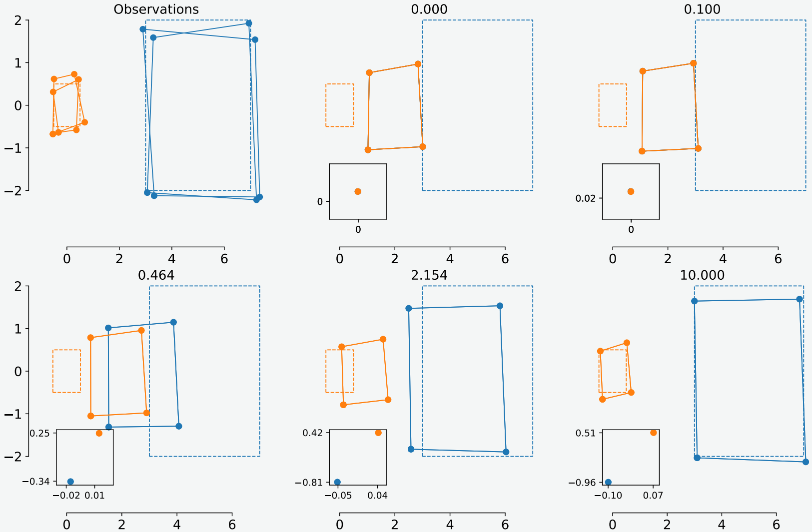
1. prior over the latent variables $\mathbf{Z}$, $p(\mathbf{Z}) \sim \mathcal{N}(\mathbf{0}, \theta_z I)$
2. lengthscale in the GP-LVM mapping (part of θ_h))

Aside: Pairwise distance alignment objective



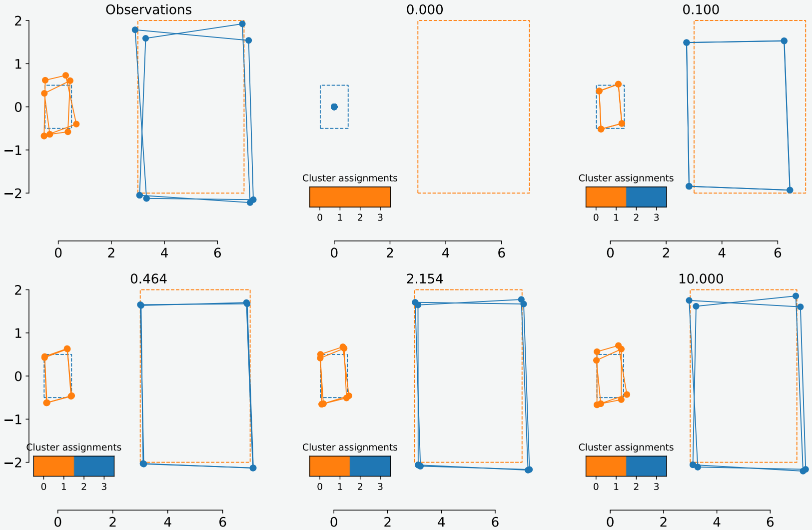
$$\mathbf{y}^i_{transformed} = \mathbf{y}^i_{\text{input}} + \mathbf{w}^i, \quad \mathbf{y}^i, \mathbf{w}^i \in \mathbb{R}^8 \text{ with } \gamma\,||\mathbf{w}||^2,\ i = 1, 2, 3, 4$$

Aside: GP-LVM as alignment objective



$$\mathbf{y}^i_{transformed} = \mathbf{y}^i_{\text{input}} + \mathbf{w}^i, \quad \mathbf{y}^i, \mathbf{w}^i \in \mathbb{R}^8 \text{ with } \gamma \, ||\mathbf{w}||^2, \; i = 1, 2, 3, 4$$

Aside: Bayesian Mixture Model as alignment objective



$$\mathbf{y}^i_{transformed} = \mathbf{y}^i_{\text{input}} + \mathbf{w}^i, \quad \mathbf{y}^i, \mathbf{w}^i \in \mathbb{R}^8 \text{ with } \gamma\,||\mathbf{w}||^2,\ i = 1, 2, 3, 4$$

Full objective for sequence alignment

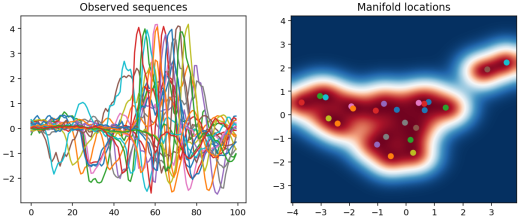
1. For each of the J sequences we perform standard GP regression on the observed data $\mathbf{y}_j$ and the pseudo-observations $\mathbf{s}_j$ by learning the hyperparameters of the GPs and the parameters of the warpings.
2. Impose the alignment objective on the pseudo-observations $\mathbf{S}$

The sum of the log-likelihoods is:

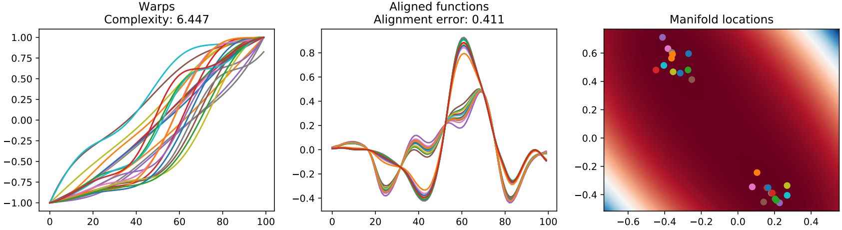
$$\begin{aligned}\mathcal{L} &= \sum_{j=1}^{J} \mathcal{L}_{\mathrm{GP}_i} + \mathcal{L}_{\text{GP-LVM}} + \sum_{j=1}^{J} \log p(g_j) \\ &= \sum_{j=1}^{J} \log p([\mathbf{s}_j, \mathbf{y}_j]^T \mid \mathbf{x}, g_j, \theta_j, \beta_j) + \mathcal{L}_{\text{GP-LVM}}(\mathbf{Z}, \psi_h, \psi_z, \gamma) + \sum_{j=1}^{J} \log p(g_j)\end{aligned} \tag{8}$$

Results on ECG data

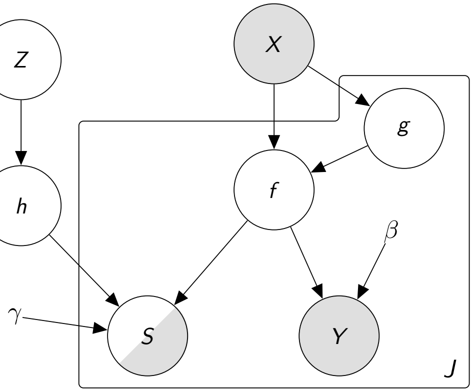
Input data:



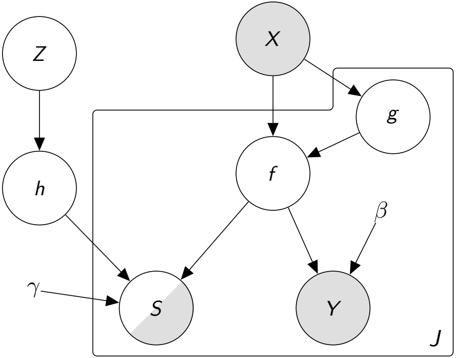
Alignment with GP-LVM objective:



Competing objectives and joint model



Competing objectives and joint model



Likelihood $p(\mathbf{S} \mid \mathbf{H}, \mathbf{F}^{\mathrm{X}})$ as an equal mixture (where S_j and S_n refer to rows and columns of $\mathbf{S}$):

$$p(\mathbf{S} \mid \mathbf{H}, \mathbf{F}^{\mathrm{X}}) = \frac{1}{2}\left(\prod_n \mathcal{N}(S_n|\mathbf{H}_n, \gamma^{-1} I_J) + \prod_j \mathcal{N}(S_j|\mathbf{F}_j^X, \beta_j^{-1} I_N)\right)$$

Multi-task learning and Matrix distributions

Given data $Y \in \mathbb{R}^{J \times N}$:

1. each sequence (row) has a GP prior and there's a free-form matrix C that models the covariances between the sequences[1].

2. learn sparse inverse covariance between features while accounting for a low-rank confounding covariance between samples using GP-LVM[2]:

$$p(Y \mid R, C^{-1}) = \mathcal{N}(\text{vec}(Y) \mid 0_{N \times D}, C \otimes R + \sigma^2 I_{N \times D}) \quad (9)$$

[1] Bonilla et al. Multi-task Gaussian Process Prediction (2008)
[2] Stegle et al. Efficient inference in matrix-variate Gaussian models with iid observation noise (2011)

More generally...

These types of constructions are useful when:

1. The data has a hierarchical structure with additional constraints:

$$\mathbf{y}_j = f_k(g_j(\mathbf{x})) + \epsilon_j, \quad \epsilon_j \sim \mathcal{N}(0, \beta_j^- 1)$$

2. We want to perform dim. reduction or clustering that is invariant to a specific transformation

Uncertainty in alignment model

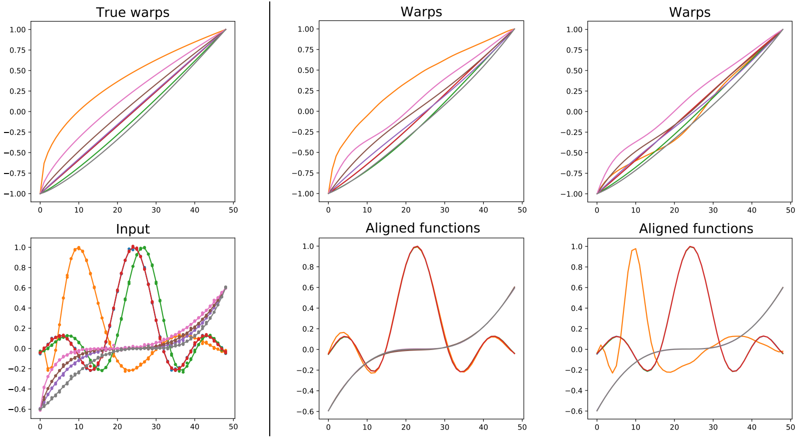
Uncertainty in alignment model

While the alignment model is probabilistic, so far we only considered point estimates and ignored the uncertainties associated with warpings and group assignments.

Uncertainty in the alignment model contains:

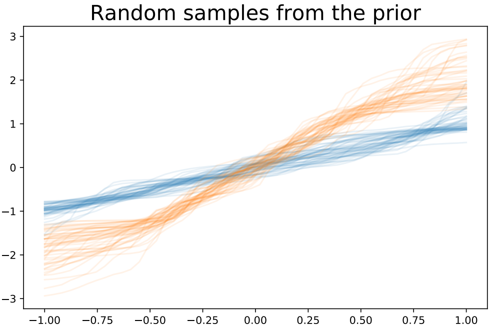
1. Observed sequences are often noisy
2. Warping uncertainty
3. Assignment of sequences to groups is ambiguous

Uncertainty in alignment model



Going beyond the point estimates of the warps

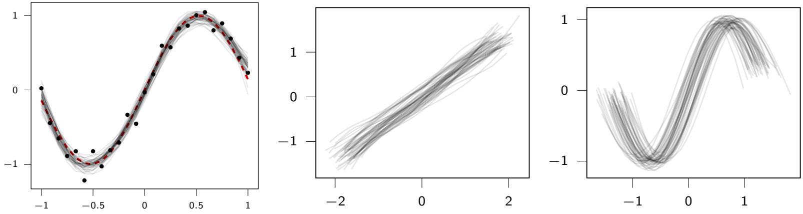
- So far we have been computing point estimates of the warps (by optimising G_j directly).
- To model warping uncertainty we developed a nonparametric model[1] of monotonic warps based on the Gaussian process differential flow model[2].



[1] Hegde et al. Deep learning with differential Gaussian process flows (2019)
[2] K. et al. Monotonic Gaussian Process Flow (2019)

Fully probabilistic model - Mean-field

- The composition of a warp (g-function) and a GP (f-function) is similar to a two-layer DGP
- Exact inference is also intractable, so we augment both layers with inducing points $\{\mathbf{U}^g\}$ and $\{\mathbf{U}^f\}$
- Inducing points effectively define mappings in each layer. If they are independent, the mappings do not match each other to fit the observations



Observations Layer 1 Layer 2

Aside: Girard et al. Gaussian Process Priors With Uncertain Inputs (2013)

Beyond mean-field variational distribution

Use optimal distribution of inducing points[1]

Two components of a variational distribution:

1. Free-form variational distribution $q(\{\mathbf{U}^g\})$ for the inducing points of the warp
2. For a given output G of the warp, we define $q(\{\mathbf{U}^f\})$ to be the optimal variational distribution[1] of inducing points in a GP mapping G to the observations

[1] M. Titsias. Variational Learning of Inducing Variables in Sparse Gaussian Processes, 2009

Beyond mean-field variational distribution

Use optimal distribution of inducing points

Fitting the model:

1. Sample $\{\mathbf{U}^g\} \sim q(\{\mathbf{U}^g\})$
2. Conditioned on this sample, sample (again) the output the warps $G \sim p(G \mid \{\mathbf{U}^g\})$
3. Conditioned on G, compute the optimal distribution of inducing points $q(\{\mathbf{U}^f\})$ and the likelihood

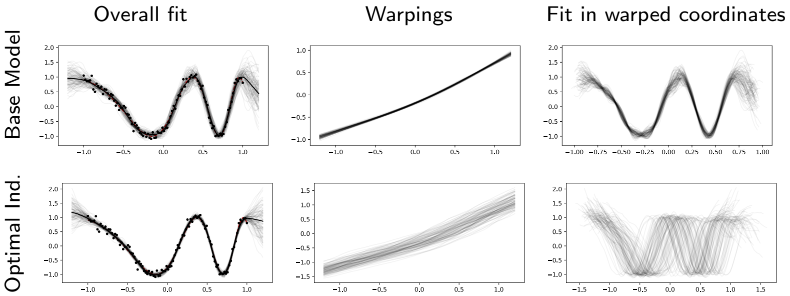
$$p(Y \mid G) = \int p(Y \mid G, \{\mathbf{U}^f\}) q(\{\mathbf{U}^f\}) d\mathbf{U}^f$$

The only variational parameters to optimise are those of $q(\{\mathbf{U}^g\})$, which we can do by maximising $p(Y \mid G)$ (using the reparametrisation trick)

Salimbeni & Deisenroth. Doubly Stochastic Variational Inference for Deep Gaussian Processes (2017)

2-layer DGP

Consider 2-layer DGP where first layer is monotonic:



Thank you

I. Kazlauskaite, C. H. Ek, N. D. F. Campbell. Gaussian Process Latent Variable Alignment Learning. *AISTATS (2019)*

I. Kazlauskaite, I. Ustyuzhaninov, C. H. Ek, N. D. F. Campbell. Sequence Alignment with Dirichlet Process Mixtures. *Bayesian Nonparametrics Workshop at NIPS (2018)*

I. Ustyuzhaninov*, I. Kazlauskaite*, C. H. Ek, N. D. F. Campbell. Monotonic Gaussian Process Flow. *arXiv (2019)*

I. Ustyuzhaninov*, I. Kazlauskaite*, M. Kaiser, E. Bodin, C. H. Ek, N. D. F. Campbell. Compositional uncertainty in deep Gaussian processes. *arXiv (2019)*