Variational Gaussian process latent variable models for high dimensional image data

Andreas Damianou[1]
joint work with Neil Lawrence[1], Michalis Titsias[2] and Carl Henrik Ek[3]

[1] Department of Neuro- and Computer Science, University of Sheffield, UK
[2] Wellcome Trust Centre for Human Genetics, University of Oxford
[3] Computer Vision and Active Perception Lab, KTH

The Rank Prize symposium 2012

Outline

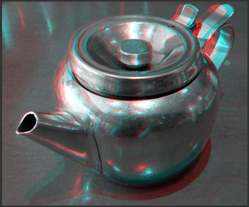
Gaussian process latent variable model (GP-LVM)

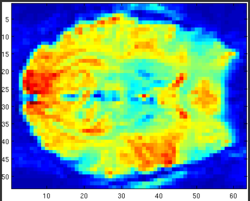
Bayesian GP-LVM

Modelling temporal data

Multi-modal modelling

Real-world image datasets in computer vision are usually high-dimensional, complex and noisy





Probabilistic methods for dim. reduction

- Observed (high-dimensional) data: $Y \in \mathbb{R}^{N \times D}$
- Actual (low-dimensional) data: $X \in \mathbb{R}^{N \times Q}$, $Q \ll D$

Probabilistic methods for dim. reduction

- Observed (high-dimensional) data: $Y \in \mathbb{R}^{N \times D}$
- Actual (low-dimensional) data: $X \in \mathbb{R}^{N \times Q}$, $Q \ll D$
- Model:

$$y_{nd} = f_d(\mathbf{x}_n) + \epsilon_n \;, \quad \epsilon_n \sim \mathcal{N}(0, \beta^{-1})$$

Gaussian process latent variable model

GP-LVM: Places a *GP prior* $f \sim \mathcal{GP}(\mathbf{0}, k_f(x, x'))$ directly on the mapping function so that:

$$p(F|X) \sim \mathcal{N}(\mathbf{0}, k_f(X, X))$$

from which we can compute the likelihood

$$p(Y|X) = \int p\,(Y|F)\,p(F|X)\mathrm{d}F = \mathcal{N}(Y|\mathbf{0}, k_f(X, X) + \beta^{-1} I_N)$$

Gaussian process latent variable model

GP-LVM: Places a *GP prior* $f \sim \mathcal{GP}(\mathbf{0}, k_f(x, x'))$ directly on the mapping function so that:

$$p(F|X) \sim \mathcal{N}(\mathbf{0}, k_f(X, X))$$

from which we can compute the likelihood

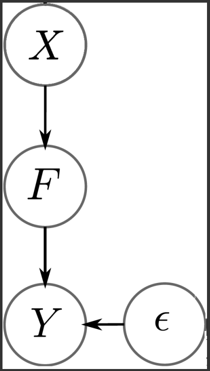
$$p(Y|X) = \int p\,(Y|F)\,p(F|X)\mathrm{d}F = \mathcal{N}(Y|\mathbf{0}, k_f(X, X) + \beta^{-1} I_N)$$

This allows for *non-linear mappings* if the covariance function k_f is non-linear. For example:

$$k_f\left(\mathbf{x}_i, \mathbf{x}_j\right) = \sigma^2 \exp\left(-\frac{1}{2}\sum_{q=1}^{Q} w_q \left(x_{i,q} - x_{j,q}\right)^2\right)$$

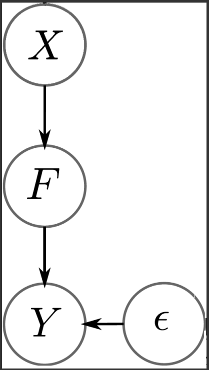
[Lawrence 2005]

Optimising the GP-LVM



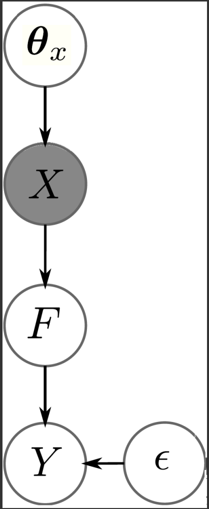
- Objective function for optimisation is $p(Y|X)$

Optimising the GP-LVM



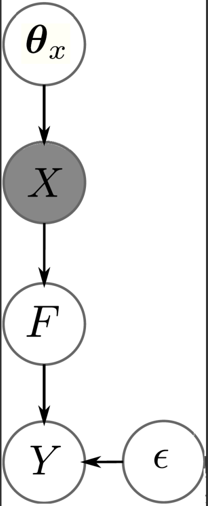
- Objective function for optimisation is $p(Y|X)$
- Problem: this finds a single point *(MAP)* estimate for X

Optimising the GP-LVM



- Objective function for optimisation is $p(Y|X)$
- Problem: this finds a single point *(MAP)* estimate for X
- We would prefer to instead find a *distribution* over $X \Rightarrow$ *Bayesian GP-LVM*

Optimising the GP-LVM



- Objective function for optimisation is $p(Y|X)$
- Problem: this finds a single point *(MAP)* estimate for X
- We would prefer to instead find a *distribution* over $X \Rightarrow$ *Bayesian GP-LVM*
- This allows for:
  - training robust to overfitting
  - automatic detection for the dimensionality of X
  - incorporating known structure on the latent space

Bayesian GP-LVM

- GP-LVM objective:

$$p(Y|X) = \int p\left(Y|\mathbf{f}\right) p(\mathbf{f}|X) d\mathbf{f} = \mathcal{N}(Y|\mathbf{0}, k_f(X,X) + \beta^{-1} I_N)$$

  The GP-LVM is trained by maximizing $p(Y|X)$ w.r.t the mapping's parameters and X (jointly) $\Rightarrow$ *MAP* estimate

- Bayesian GP-LVM: Also integrate out X's:

$$p(Y) = \int p\left(Y|X\right) p(X|\boldsymbol{\theta}_x) \mathrm{d}X$$

Bayesian GP-LVM

- GP-LVM objective:

$$p(Y|X) = \int p\,(Y|\mathbf{f})\,p(\mathbf{f}|X)d\mathbf{f} = \mathcal{N}(Y|\mathbf{0}, k_f(X,X) + \beta^{-1}I_N)$$

  The GP-LVM is trained by maximizing $p(Y|X)$ w.r.t the mapping's parameters and X (jointly) $\Rightarrow$ *MAP* estimate

- Bayesian GP-LVM: Also integrate out X's:

$$p(Y) = \int p\,(Y|X)\,p(X|\boldsymbol{\theta}_x)\mathrm{d}X$$

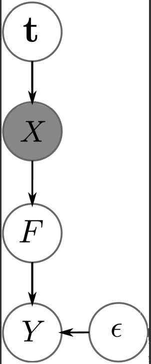
- Tractability: The marginal likelihood as well as the posterior $p(X|Y)$ are intractable $\Rightarrow$ variational framework in an expanded probability model [Titsias and Lawrence, 2010] and find a bound:

$$\mathcal{F}_v \leq \log p(Y)$$

Incorporating prior assumptions

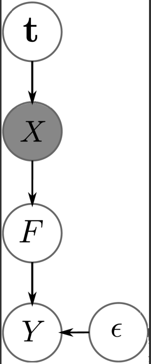
- Unconstrained X: $p(X) \sim \mathcal{N}(\mathbf{0}, I_Q)$

Incorporating prior assumptions



- Unconstrained X: $p(X) \sim \mathcal{N}(\mathbf{0}, I_Q)$
- Model dynamics: $\mathbf{x} = x(t) \sim \mathcal{GP}(\mathbf{0}, k_x)$

Incorporating prior assumptions



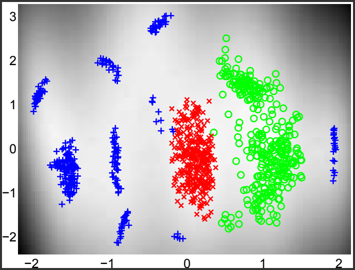
- Unconstrained X: $p(X) \sim \mathcal{N}(\mathbf{0}, I_Q)$
- Model dynamics: $\mathbf{x} = x(t) \sim \mathcal{GP}(\mathbf{0}, k_x)$
- $\mathbf{x}$'s coupled in $p(X) \Rightarrow \mathcal{O}(N^2)$ var. parameters in the approximate posterior $q(X) \sim \mathcal{N}(\boldsymbol{\mu}, S)$
  - Reparametrization using fixed point equations $\Rightarrow \mathcal{O}(N)$ actual parameters: $S = (K_x^{-1} + \text{diag}(\boldsymbol{\lambda}))^{-1}$

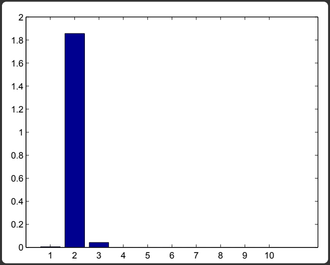
Automatic dimensionality detection

- Achieved by employing *automatic relevance determination (ARD)* priors for the mapping f.
- $f \sim \mathcal{GP}(\mathbf{0}, k_f)$ with:

$$k_f\left(\mathbf{x}_i, \mathbf{x}_j\right) = \sigma^2 \exp\left(-\frac{1}{2}\sum_{q=1}^{Q} w_q \left(x_{i,q} - x_{j,q}\right)^2\right)$$

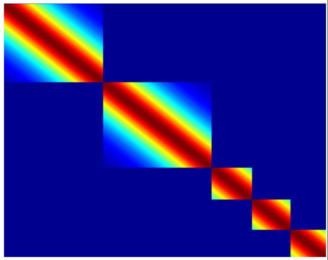
- Example:





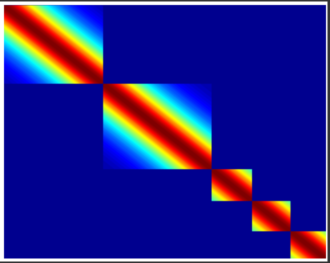
More on dynamics

- Dynamics are encoded in the covariance matrix $K_x = k_x(\mathbf{t}, \mathbf{t})$, e.g. forcing K_x to be block-diagonal allows to jointly model individual sequences



More on dynamics

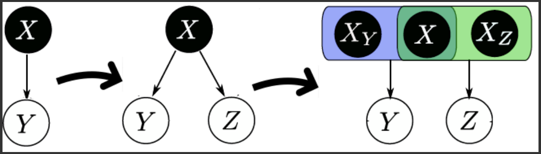
- Dynamics are encoded in the covariance matrix $K_x = k_x(\mathbf{t}, \mathbf{t})$, e.g. forcing K_x to be block-diagonal allows to jointly model individual sequences



- *Video examples...*
  `http://www.youtube.com/watch?v=i9TEoYxaBxQ`
  `http://www.youtube.com/watch?v=mUY1XHPnoCU`

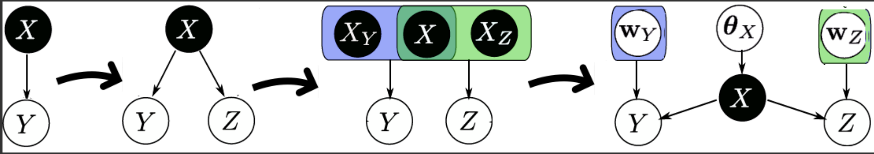
Multi-modal modelling

- Several observation modalities for the same underlying phenomenon
- Challenge: factorise the latent space into parts that are either private or shared for all modalities



Multi-modal modelling

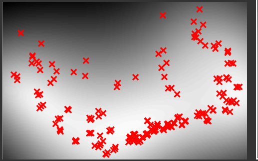
- Several observation modalities for the same underlying phenomenon
- Challenge: factorise the latent space into parts that are either private or shared for all modalities
- Bayesian solution: use a separate set of *ARD* parameters for each modality
- The ARD weights are optimised to learn the responsibility of each latent dimension for generating each of the observation spaces $\Rightarrow$ *soft* segmentation



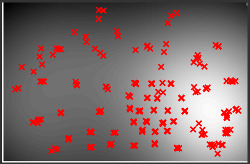
Example: Yale faces

- Dataset Y: 3 persons under all illumination conditions
- Dataset Z: As above for 3 different persons
- Align datapoints $\mathbf{y}_n$ and $\mathbf{z}_n$ only based on the lighting direction
- Show MATLAB demo / video results...

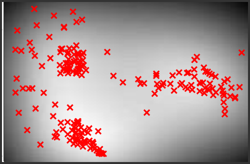
Results



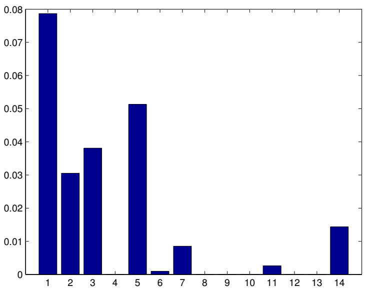
Dims 1 vs 2



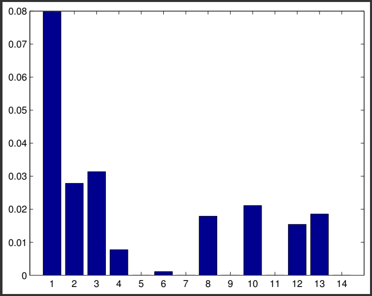
Dims 1 vs 3



Dims 5 vs 14



ARD weights for Y



ARD weights for Z

Applications

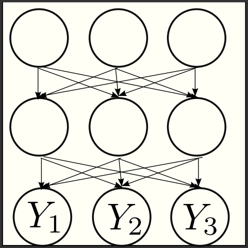
- Exploring the structure of the latent space
- Generation of novel observations
- Supervised dimensionality reduction
- Transfer information between modalities

Applications

- Exploring the structure of the latent space
- Generation of novel observations

- Supervised dimensionality reduction
- Transfer information between modalities

- Extension to hierarchical scenarios (deep architectures)



Summary

- GP-LVM: probabilistic non-linear dimensionality reduction
- Bayesian GP-LVM: placing a prior over and marginalising the latent space
- Dynamical framework: constraining the latent space to be a timeseries
- Multi-modal framework: automatically segment the latent space to shared and private subspaces

Thanks

Univ. of Oxford
Michalis Titsias

KTH
Carl Henrik Ek

Univ. of Sheffield
Neil Lawrence

Funding

- University of Sheffield Moody endowment fund
- Greek State Scholarships Foundation (IKY)