Deep Gaussian Processes

Learning Abstract Features with Gaussian Process Models

Neil D. Lawrence
Departments of Neuro- and Computer Science, University of Sheffield, U.K.

University College London

30th January 2013

Outline

Deep Motivation

Bayesian GP-LVM

Deep GPs

Conclusions

Outline

Deep Motivation

Bayesian GP-LVM

Deep GPs

Conclusions

direction for further research.

11.1. HAVE WE THROWN THE BABY OUT WITH THE BATH WATER?

According to the hype of 1987, neural networks were meant to be intelligent models which discovered features and patterns in data. Gaussian processes in contrast are simply smoothing devices. How can Gaussian processes possibly replace neural networks? What is going on?

I think what the work of Williams and Rasmussen (1996) shows is that many real–world data modelling problems are perfectly well solved by sensible smoothing methods. The most interesting problems, the task of feature discovery for example, are not ones which Gaussian processes will solve. But maybe multilayer perceptrons can't solve them either. On the other hand, it may be that the limit of an infinite number of hidden units, to which Gaussian processes correspond, was a bad limit to take; maybe we should backtrack, or modify the prior on neural network parameters, so as to create new models more interesting than Gaussian processes. Evidence that this infinite limit has lost something compared with finite neural networks comes from the observation that in a finite neural network with more than one output, there are non–trivial correlations between the outputs (since they share inputs from common hidden units); but in the limit of an infinite number of hidden units, these correlations vanish. Radford Neal has suggested the use of non–Gaussian priors in networks with multiple hidden layers. Or perhaps a completely fresh start is needed, approaching the problem of machine learning from a paradigm different from the supervised feedforward mapping.

Structure of Priors

MacKay: NIPS Tutorial 1997 “Have we thrown out the baby with the bathwater?” (Published as MacKay, 1998) Also noted by (Wilson et al., 2012)

Deep Models

- Universal approximator arguments ignore interesting priors.
- Gaussian process priors are amazing, but still limited.
    - Struggle to learn unusual long range correlations
    - Makes covariance functions inappropriate for ‘multitask learning’.

Restricted Boltzman Machine

Linear Latent Variable Model

- Represent data, $\mathbf{Y}$, with a set of latent variables $\mathbf{X}$.
- Assume a linear-logistic relationship of the form

$$P(\mathbf{y}_{i,j}) = \sigma_{i,j}^{y_{i,j}}(1-\sigma_{i,j})^{(1-\sigma_{i,j})}$$

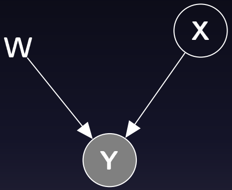
where

$$\sigma_{i,j} = \frac{1}{1+\exp\left(-\mathbf{w}_{j,i}^{\top}\mathbf{x}_{i,:}\right)},$$

Restricted Boltzman Machine

RBM

- Define *linear-logistic relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define binomial prior over *latent space*, **X**.
  - Integrate out *latent variables* ... need to sample ...

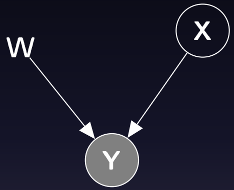


$$p\left(\mathbf{Y}|\mathbf{X},\mathbf{W}\right) = \prod_{i=1}^{n}\prod_{j=1}^{p} \sigma_{i,j}^{y_{i,j}} \left(1-\sigma_{i,j}\right)^{\left(1-y_{i,j}\right)}$$

Restricted Boltzman Machine

RBM

- Define *linear-logistic relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define binomial prior over *latent space*, **X**.
  - Integrate out *latent variables* ... need to sample ...

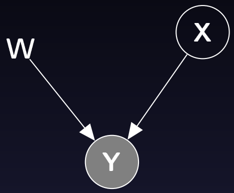


$$p\left(\mathbf{Y}|\mathbf{X},\mathbf{W}\right)=\prod_{i=1}^{n}\prod_{j=1}^{p}\sigma_{i,j}^{y_{i,j}}\left(1-\sigma_{i,j}\right)^{(1-y_{i,j})}$$

Restricted Boltzman Machine

RBM

- Define *linear-logistic relationship* between latent variables and data.
- **Standard** Latent variable approach:
    - Define binomial prior over *latent space*, $\mathbf{X}$.
    - Integrate out *latent variables* ... need to sample ...



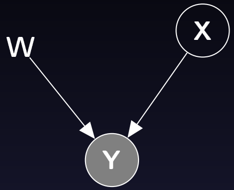
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n}\prod_{j=1}^{p}\sigma_{i,j}^{y_{i,j}}(1-\sigma_{i,j})^{(1-y_{i,j})}$$

$$p(\mathbf{X}) = \prod_{i=1}^{n}\prod_{i=1}^{q}h_i^{x_{i,j}}(1-h_i)^{(1-x_{i,j})}$$

Restricted Boltzman Machine

RBM

- Define *linear-logistic relationship* between latent variables and data.
- **Standard** Latent variable approach:
    - Define binomial prior over *latent space*, $\mathbf{X}$.
    - Integrate out *latent variables* ... need to sample ...

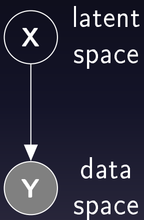


$$p\left(\mathbf{Y}|\mathbf{X},\mathbf{W}\right)=\prod_{i=1}^{n}\prod_{j=1}^{p}\sigma_{i,j}^{y_{i,j}}\left(1-\sigma_{i,j}\right)^{\left(1-y_{i,j}\right)}$$

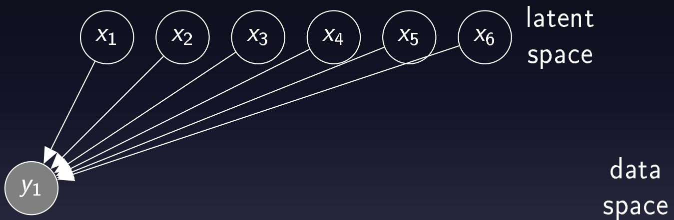
$$p\left(\mathbf{X}\right)=\prod_{i=1}^{n}\prod_{i=1}^{q}h_{i}^{x_{i,j}}\left(1-h_{i}\right)^{\left(1-x_{i,j}\right)}$$

$$p\left(\mathbf{Y}|\mathbf{W}\right)=??$$

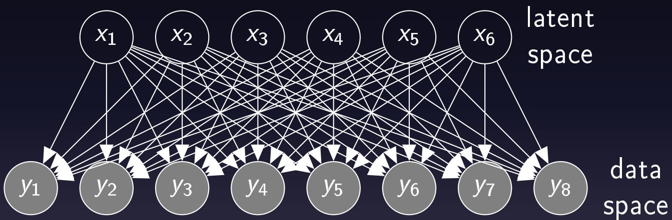
Shallow to Deep



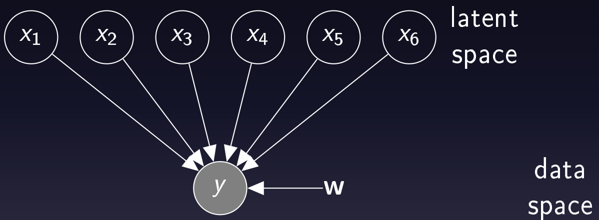
Shallow to Deep



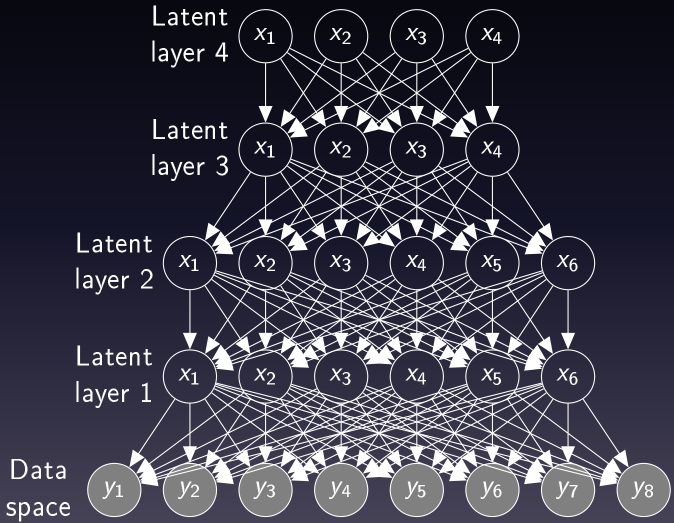
Shallow to Deep



Shallow to Deep



Deep Models



Latent layer 4
x_1 x_2 x_3 x_4
Latent layer 3
x_1 x_2 x_3 x_4
Latent layer 2
x_1 x_2 x_3 x_4 x_5 x_6
Latent layer 1
x_1 x_2 x_3 x_4 x_5 x_6
Data space
y_1 y_2 y_3 y_4 y_5 y_6 y_7 y_8

Deep Gaussian Processes

Work with Andreas Damianou

- Deep architectures allow abstraction of features (Bengio, 2009; Hinton and Osindero, 2006; Salakhutdinov and Murray, 2008).
- We use variational approach to stack GP models.
- Similar to GPDS, but apply recursively.

Linear Dimensionality Reduction

Linear Latent Variable Model

- Represent data, $\mathbf{Y}$, with a lower dimensional set of latent variables $\mathbf{X}$.
- Assume a linear relationship of the form

$$\mathbf{y}_{i,:} = \mathbf{W}\mathbf{x}_{i,:} + \boldsymbol{\epsilon}_{i,:},$$

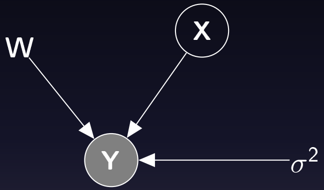
where

$$\boldsymbol{\epsilon}_{i,:} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right).$$

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.

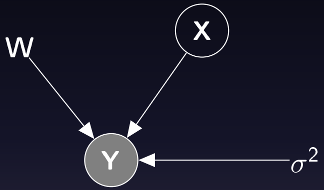


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.

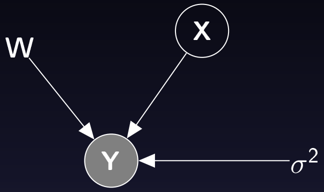


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.



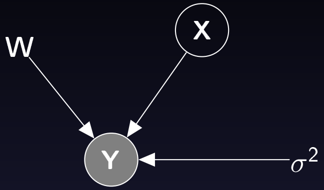
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

$$p(\mathbf{X}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{x}_{i,:}|\mathbf{0},\mathbf{I}\right)$$

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:
    - Define Gaussian prior over *latent space*, $\mathbf{X}$.
    - Integrate out *latent variables*.



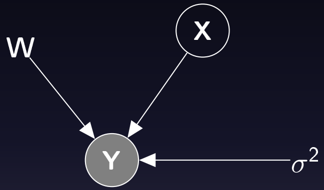
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

$$p(\mathbf{X}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{x}_{i,:}|\mathbf{0},\mathbf{I}\right)$$

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0},\mathbf{W}\mathbf{W}^{\top}+\sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)



$$p\left(\mathbf{Y}|\mathbf{W}\right)=\prod_{i=1}^{n}\mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0},\mathbf{W}\mathbf{W}^{\top}+\sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^{\top} + \sigma^2 \mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{W}) = -\frac{n}{2}\log|\mathbf{C}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{C}^{-1}\mathbf{Y}^{\top}\mathbf{Y}\right) + \mathrm{const.}$$

If $\mathbf{U}_q$ are first q principal eigenvectors of $n^{-1}\mathbf{Y}^{\top}\mathbf{Y}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

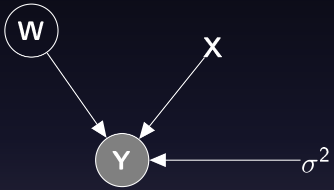
$$\mathbf{W} = \mathbf{U}_q \mathbf{L} \mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2 \mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model III

Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, $\mathbf{W}$.
  - Integrate out *parameters*.

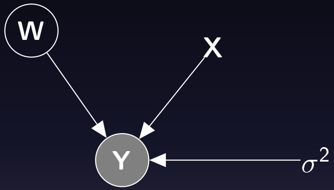


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model III

Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, $\mathbf{W}$.
  - Integrate out *parameters*.

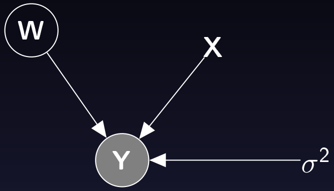


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model III

Dual Probabilistic PCA

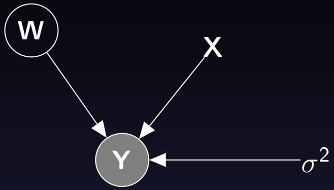
- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, $\mathbf{W}$.
  - Integrate out *parameters*.



$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

$$p(\mathbf{W}) = \prod_{i=1}^{p} \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0}, \mathbf{I}\right)$$

Linear Latent Variable Model III



Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, **W**.
  - Integrate out *parameters*.

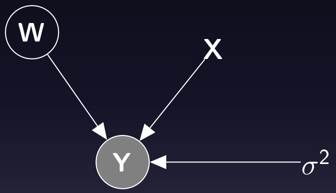
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

$$p(\mathbf{W}) = \prod_{i=1}^{p} \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0}, \mathbf{I}\right)$$

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model IV

Dual Probabilistic PCA Max. Likelihood Soln (Lawrence, 2004, 2005)



$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model IV

Dual Probabilistic PCA Max. Likelihood Soln (Lawrence, 2004, 2005)

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right), \quad \mathbf{K} = \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{X}) = -\frac{p}{2}\log|\mathbf{K}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\top}\right) + \text{const.}$$

If $\mathbf{U}_q'$ are first q principal eigenvectors of $p^{-1}\mathbf{Y}\mathbf{Y}^{\top}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{X} = \mathbf{U}_q'\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Linear Latent Variable Model IV

Probabilistic PCA Max. Likelihood Soln (Tipping and Bishop, 1999)

$$p(\mathbf{Y}|\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}\right), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^{\top} + \sigma^2\mathbf{I}$$

$$\log p(\mathbf{Y}|\mathbf{W}) = -\frac{n}{2}\log|\mathbf{C}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{C}^{-1}\mathbf{Y}^{\top}\mathbf{Y}\right) + \mathrm{const.}$$

If $\mathbf{U}_q$ are first q principal eigenvectors of $n^{-1}\mathbf{Y}^{\top}\mathbf{Y}$ and the corresponding eigenvalues are $\mathbf{\Lambda}_q$,

$$\mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

where $\mathbf{R}$ is an arbitrary rotation matrix.

Equivalence of Formulations

The Eigenvalue Problems are equivalent

- Solution for Probabilistic PCA (solves for the mapping)

$$\mathbf{Y}^{\top}\mathbf{Y}\mathbf{U}_q = \mathbf{U}_q\boldsymbol{\Lambda}_q \qquad \mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^{\top}$$

- Solution for Dual Probabilistic PCA (solves for the latent positions)

$$\mathbf{Y}\mathbf{Y}^{\top}\mathbf{U}'_q = \mathbf{U}'_q\boldsymbol{\Lambda}_q \qquad \mathbf{X} = \mathbf{U}'_q\mathbf{L}\mathbf{R}^{\top}$$

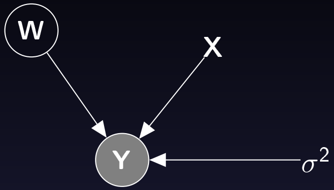
- Equivalence is from

$$\mathbf{U}_q = \mathbf{Y}^{\top}\mathbf{U}'_q\boldsymbol{\Lambda}_q^{-\frac{1}{2}}$$

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
    - Define Gaussian prior over *parameteters*, $\mathbf{W}$.
    - Integrate out *parameters*.



$$p\left(\mathbf{Y}|\mathbf{X},\mathbf{W}\right)=\prod_{i=1}^{n}\mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

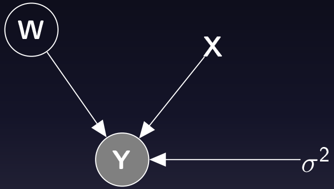
$$p\left(\mathbf{W}\right)=\prod_{i=1}^{p}\mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0},\mathbf{I}\right)$$

$$p\left(\mathbf{Y}|\mathbf{X}\right)=\prod_{j=1}^{p}\mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\mathbf{X}\mathbf{X}^\top+\sigma^2\mathbf{I}\right)$$

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
  - The covariance matrix is a covariance function.
  - We recognise it as the 'linear kernel'.
  - We call this the Gaussian Process Latent Variable model (GP-LVM).

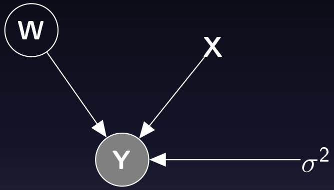


$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right)$$

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
    - The covariance matrix is a covariance function.
    - We recognise it as the 'linear kernel'.
    - We call this the Gaussian Process Latent Variable model (GP-LVM).



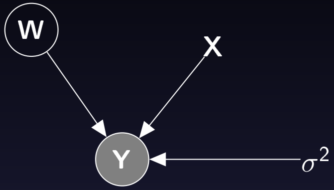
$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

$$\mathbf{K} = \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}$$

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
  - The covariance matrix is a covariance function.
  - We recognise it as the ‘linear kernel’.
  - We call this the Gaussian Process Latent Variable model (GP-LVM).



$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

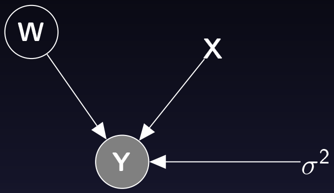
$$\mathbf{K} = \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}$$

This is a product of Gaussian processes with linear kernels.

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
  - The covariance matrix is a covariance function.
  - We recognise it as the 'linear kernel'.
  - We call this the Gaussian Process Latent Variable model (GP-LVM).



$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

$$\mathbf{K} = ?$$

Replace linear kernel with non-linear kernel for non-linear model.

Exponentiated Quadratic (EQ) Covariance

- The EQ covariance has the form $k_{i,j} = k\left(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}\right)$, where

$$k\left(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}\right) = \alpha \exp\left(-\frac{\left\|\mathbf{x}_{i,:} - \mathbf{x}_{j,:}\right\|_2^2}{2\ell^2}\right).$$

- No longer possible to optimise wrt $\mathbf{X}$ via an eigenvalue problem.
- Instead find gradients with respect to $\mathbf{X}, \alpha, \ell$ and σ^2 and optimise using conjugate gradients.

Outline

Deep Motivation

Bayesian GP-LVM

Deep GPs

Conclusions

Learning in Larger Datasets

(Lawrence, 2007; Titsias, 2009)

- Complexity of standard GP:
  - $O(n^3)$ in computation.
  - $O(n^2)$ in storage.
- Via low rank representations of covariance:
  - $O(nm^2)$ in computation.
  - $O(nm)$ in storage.
- Where m is user chosen number of *inducing* variables. They give the rank of the resulting covariance.

Inducing Variable Approximations

- Date back to (Williams and Seeger, 2001; Smola and Bartlett, 2001; Csató and Opper, 2002; Seeger et al., 2003; Snelson and Ghahramani, 2006). See Quiñonero Candela and Rasmussen (2005) for a review.
- We follow variational perspective of (Titsias, 2009).
- This is an augmented variable method, followed by a collapsed variational approximation (King and Lawrence, 2006; Hensman et al., 2012).

Augmented Variable Model

Augment standard GP model with a set of m new inducing variables, $\mathbf{u}$.

$$p(\mathbf{y}) = \int p(\mathbf{y}, \mathbf{u}) \mathrm{d}\mathbf{u}$$

Augmented Variable Model

Augment standard GP model with a set of m new inducing variables, $\mathbf{u}$.

$$p(\mathbf{y}) = \int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u}$$

Augmented Variable Model

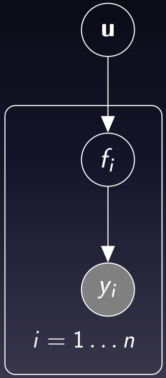
Assume that relationship is through $\mathbf{f}$.

$$p(\mathbf{y}) = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{f}\mathrm{d}\mathbf{u}$$

Augmented Variable Model

Very often likelihood factorizes.

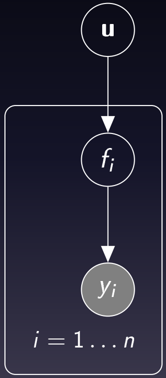
$$p(\mathbf{y}) = \int \prod_{i=1}^{n} p(y_i|f_i) p(\mathbf{f}|\mathbf{u}) p(\mathbf{u}) \mathrm{d}\mathbf{f} \mathrm{d}\mathbf{u}$$



Augmented Variable Model

Focus on integral over $\mathbf{f}$.

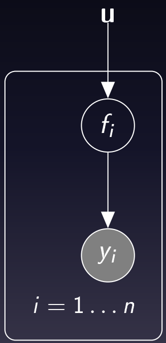
$$p(\mathbf{y}) = \int \int \prod_{i=1}^{n} p(y_i|f_i)p(\mathbf{f}|\mathbf{u})\mathrm{d}\mathbf{f}\, p(\mathbf{u})\mathrm{d}\mathbf{u}$$



Augmented Variable Model

Focus on integral over $\mathbf{f}$.

$$p(\mathbf{y}|\mathbf{u}) = \int \prod_{i=1}^{n} p(y_i|f_i)p(\mathbf{f}|\mathbf{u})\mathrm{d}\mathbf{f}$$



Variational Bound on $p(\mathbf{y}|\mathbf{u})$

$$\begin{aligned}\log p(\mathbf{y}|\mathbf{u}) &= \log \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})\mathrm{d}\mathbf{f} \\ &\geq \int q(\mathbf{f}) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})}{q(\mathbf{f})}\mathrm{d}\mathbf{f}\end{aligned}$$

- For variational approximation of (Titsias, 2009) set $q(\mathbf{f}) = p(\mathbf{f}|\mathbf{u})$,

$$\log p(\mathbf{y}|\mathbf{u}) \geq \log \int p(\mathbf{f}|\mathbf{u}) \log p(\mathbf{y}|\mathbf{f})\mathrm{d}\mathbf{f}.$$

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \log p(\mathbf{y}|\mathbf{f})\mathrm{d}\mathbf{f}.$$

Deterministic Training Conditional

- The variational bound factorizes over data points.
- Marginalizing over $p(\mathbf{u})$ is analytic.
  - This results in a modified variant of the projected process approximation (Rasmussen and Williams, 2006) or DTC (Quiñonero Candela and Rasmussen, 2005). Proposed by (Smola and Bartlett, 2001; Seeger et al., 2003; Csató and Opper, 2002; Csató, 2002).

$$L \geq \sum_{i=1}^{n} \log c_i + \log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}^\top \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{K}_{\mathbf{u},\mathbf{f},}\right)$$

Deterministic Training Conditional

- The variational bound factorizes over data points.
- Marginalizing over $p(\mathbf{u})$ is analytic.
    - This results in a modified variant of the projected process approximation (Rasmussen and Williams, 2006) or DTC (Quiñonero Candela and Rasmussen, 2005). Proposed by (Smola and Bartlett, 2001; Seeger et al., 2003; Csató and Opper, 2002; Csató, 2002).

$$L \approx \log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \sigma^2\mathbf{I} + \mathbf{K}_{\mathbf{f},\mathbf{u}}^{\top}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f},}\right)$$

Selecting Data Dimensionality

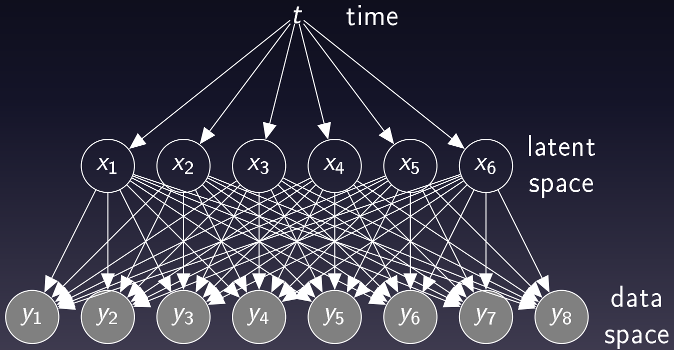
- GP-LVM Provides probabilistic non-linear dimensionality reduction.
- How to select the dimensionality?
- Need to estimate marginal likelihood.
- In standard GP-LVM it increases with increasing q.

Variational Latent Variables

- Variational marginalizing of $\mathbf{X}$ is *also* analytic.
- Need to assume Gaussian $q(\mathbf{X})$.
- Compute expectations of $q(\mathbf{X})$ then analytically marginalize $p(\mathbf{u})$ as before. (Titsias and Lawrence, 2010; Hensman et al., 2012)
    - Requires expectations of $\mathbf{K}_{\mathbf{f},\mathbf{u}}$ and $\mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{f}}$.

Gaussian Process Dynamical Systems

Work with Andreas Damianou and Michalis Titsias



Gaussian Process over Latent Space

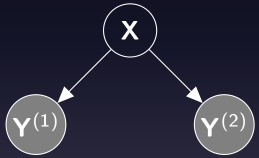
- Assume a GP prior for $p(\mathbf{X})$.
- Input to the process is time, $p(\mathbf{X}|t)$.

Gaussian Process over Latent Space

- Allows to interpret high dimensional video.
- Examples: Missa and Dog Generation.

Modeling Multiple 'Views'

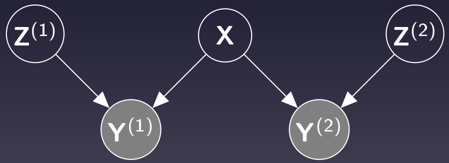
- Single space to model correlations between two different data sources, e.g., images & text, image & pose.
- Shared latent spaces: (Shon et al., 2006; Navaratnam et al., 2007; Ek et al., 2008b)



- Effective when the 'views' are correlated.
- But not all information is shared between both 'views'.
- PCA applied to concatenated data vs CCA applied to data.

Shared-Private Factorization

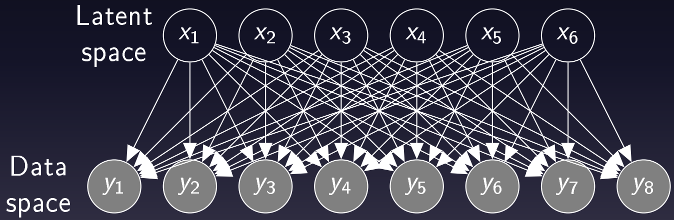
- In real scenarios, the ‘views’ are neither fully independent, nor fully correlated.
- Shared models
  - either allow information relevant to a single view to be mixed in the shared signal,
  - or are unable to model such private information.
- Solution: Model shared and private information (Virtanen et al., 2011; Ek et al., 2008a; Leen and Fyfe, 2006; Klami and Kaski, 2007, 2008; Tucker, 1958)



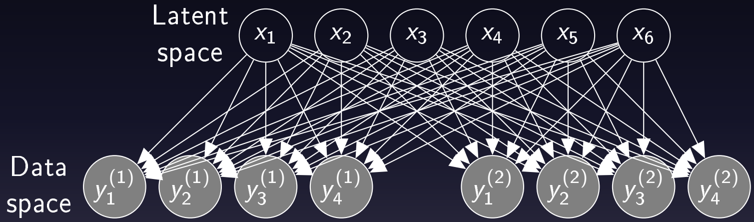
- Probabilistic CCA is case when dimensionality of $\mathbf{Z}$ matches $\mathbf{Y}^{(i)}$ (cf Inter Battery Factor Analysis (Tucker, 1958)).

Manifold Relevance Determination

Work with Andreas Damianou and Carl Henrik Ek



Shared GP-LVM



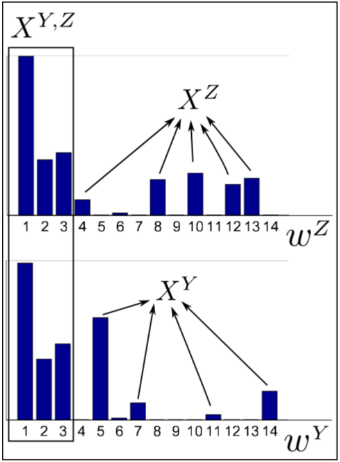
Separate ARD parameters for mappings to $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(2)}$.

Example: Yale faces

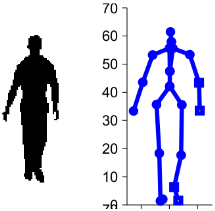
- Dataset Y: 3 persons under all illumination conditions
- Dataset Z: As above for 3 different persons
- Align datapoints $\mathbf{x}_n$ and $\mathbf{z}_n$ only based on the lighting direction

Results

- Latent space X initialised with 14 dimensions
- Weights define a segmentation of X
- Video / demo…



Potential applications..?



Outline

Deep Motivation

Bayesian GP-LVM

Deep GPs

Conclusions

Hierarchical GP-LVM

(Lawrence and Moore, 2007)

Stacking Gaussian Processes

- Regressive dynamics provides a simple hierarchy.
    - The input space of the GP is governed by another GP.
- By stacking GPs we can consider more complex hierarchies.
- Ideally we should marginalise latent spaces
    - In practice we seek MAP solutions.

Two Correlated Subjects

(Lawrence and Moore, 2007)

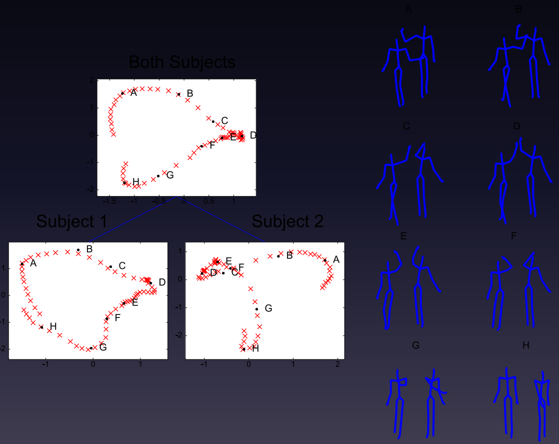


Figure: Hierarchical model of a 'high five'.

Within Subject Hierarchy

(Lawrence and Moore, 2007)

Decomposition of Body

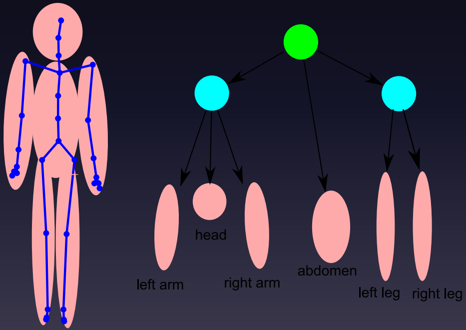
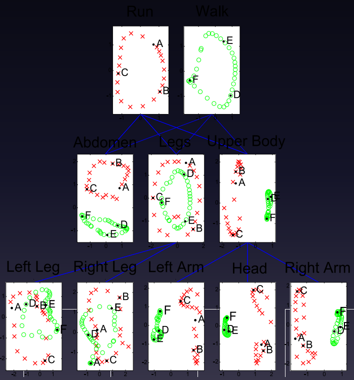


Figure: Decomposition of a subject.

Single Subject Run/Walk

(Lawrence and Moore, 2007)



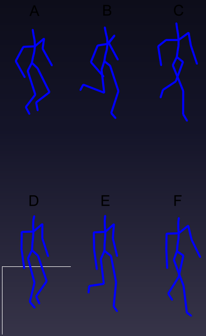


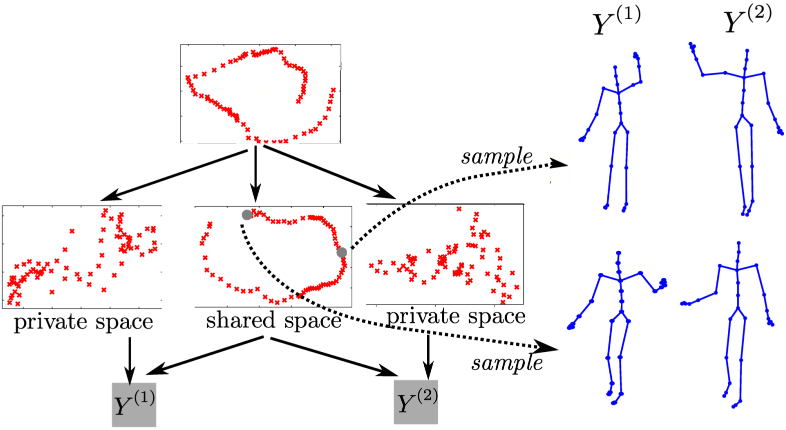
Figure: Hierarchical model of a walk and a run.

Return

Motion Capture

- Revisit 'high five' data.
- This time allow model to learn structure, rather than imposing it.

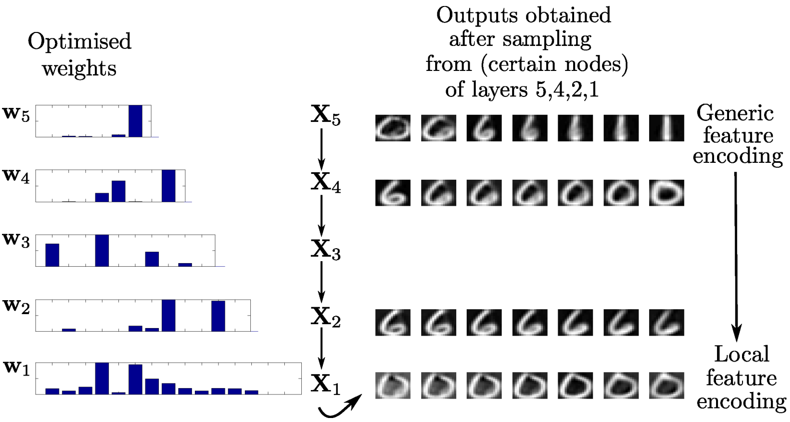
Deep hierarchies – motion capture



Digits Data Set

- Are deep hierarchies justified for small data sets?
- We can lower bound the evidence for different depths.
- For 150 6s, 0s and 1s from MNIST we found at least 5 layers are required.

Deep hierarchies – MNIST



Summary

- Variational GP-LVM gives dimensionality estimation in non linear PCA.
- Shared models use structure learning to do manifold relevance determination.
- Temporal models place a GP prior on the latent space to ensure time dependence of variables.
- Deep GPs place GP-LVM priors on each layer recursively.

References I

Y. Bengio. Learning Deep Architectures for AI. *Found. Trends Mach. Learn.*, 2(1):1–127, Jan. 2009. ISSN 1935-8237. [DOI].

L. Csató. *Gaussian Processes — Iterative Sparse Approximations*. PhD thesis, Aston University, 2002.

L. Csató and M. Opper. Sparse on-line Gaussian processes. *Neural Computation*, 14(3):641–668, 2002.

C. H. Ek, J. Rihan, P. Torr, G. Rogez, and N. D. Lawrence. Ambiguity modeling in latent spaces. In A. Popescu-Belis and R. Stiefelhagen, editors, *Machine Learning for Multimodal Interaction (MLMI 2008)*, LNCS, pages 62–73. Springer-Verlag, 28–30 June 2008a. [PDF].

C. H. Ek, P. H. Torr, and N. D. Lawrence. Gaussian process latent variable models for human pose estimation. In A. Popescu-Belis, S. Renals, and H. Bourlard, editors, *Machine Learning for Multimodal Interaction (MLMI 2007)*, volume 4892 of *LNCS*, pages 132–143, Brno, Czech Republic, 2008b. Springer-Verlag. [PDF].

Z. Ghahramani, editor. *Proceedings of the International Conference in Machine Learning*, volume 24, 2007. Omnipress. [Google Books] .

J. Hensman, M. Rattray, and N. D. Lawrence. Fast variational inference in the exponential family. *NIPS 2012*, 2012.

G. E. Hinton and S. Osindero. A fast learning algorithm for deep belief nets. *Neural Computation*, 18: 2006, 2006.

N. J. King and N. D. Lawrence. Fast variational inference for Gaussian Process models through KL-correction. In *ECML, Berlin, 2006*, Lecture Notes in Computer Science, pages 270–281, Berlin, 2006. Springer-Verlag. [PDF].

A. Klami and S. Kaski. Local dependent components analysis. In Ghahramani (2007). [Google Books] .

A. Klami and S. Kaski. Probabilistic approach to detecting dependencies between data sets. *Neurocomputing*, 72:39–46, 2008.

N. D. Lawrence. Gaussian process models for visualisation of high dimensional data. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems*, volume 16, pages 329–336, Cambridge, MA, 2004. MIT Press.

References II

N. D. Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *Journal of Machine Learning Research*, 6:1783–1816, 11 2005.

N. D. Lawrence. Learning for larger datasets with the Gaussian process latent variable model. In M. Meila and X. Shen, editors, *Proceedings of the Eleventh International Workshop on Artificial Intelligence and Statistics*, pages 243–250, San Juan, Puerto Rico, 21-24 March 2007. Omnipress. [PDF].

N. D. Lawrence and A. J. Moore. Hierarchical Gaussian process latent variable models. In Ghahramani (2007), pages 481–488. [Google Books] . [PDF].

G. Leen and C. Fyfe. A Gaussian process latent variable model formulation of canonical correlation analysis. Bruges (Belgium), 26-28 April 2006 2006.

T. K. Leen, T. G. Dietterich, and V. Tresp, editors. *Advances in Neural Information Processing Systems*, volume 13, Cambridge, MA, 2001. MIT Press.

D. J. C. MacKay. Introduction to Gaussian Processes. In C. M. Bishop, editor, *Neural Networks and Machine Learning*, volume 168 of *Series F: Computer and Systems Sciences*, pages 133–166. Springer-Verlag, Berlin, 1998.

R. Navaratnam, A. Fitzgibbon, and R. Cipolla. The joint manifold model for semi-supervised multi-valued regression. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE Computer Society Press, 2007.

J. Quiñonero Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006. [Google Books] .

R. Salakhutdinov and I. Murray. On the quantitative analysis of deep belief networks. In S. Roweis and A. McCallum, editors, *Proceedings of the International Conference in Machine Learning*, volume 25, pages 872–879. Omnipress, 2008.

M. Seeger, C. K. I. Williams, and N. D. Lawrence. Fast forward selection to speed up sparse Gaussian process regression. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, Key West, FL, 3–6 Jan 2003.

References III

A. P. Shon, K. Grochow, A. Hertzmann, and R. P. N. Rao. Learning shared latent structure for image synthesis and robotic imitation. In Weiss et al. (2006).

A. J. Smola and P. L. Bartlett. Sparse greedy Gaussian process regression. In Leen et al. (2001), pages 619–625.

E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In Weiss et al. (2006).

M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, B*, 6(3):611–622, 1999. [PDF]. [DOI].

M. K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In D. van Dyk and M. Welling, editors, *Proceedings of the Twelfth International Workshop on Artificial Intelligence and Statistics*, volume 5, pages 567–574, Clearwater Beach, FL, 16-18 April 2009. JMLR W&CP 5.

M. K. Titsias and N. D. Lawrence. Bayesian Gaussian process latent variable model. In Y. W. Teh and D. M. Titterington, editors, *Proceedings of the Thirteenth International Workshop on Artificial Intelligence and Statistics*, volume 9, pages 844–851, Chia Laguna Resort, Sardinia, Italy, 13-16 May 2010. JMLR W&CP 9. [PDF].

L. R. Tucker. An inter-battery method of factor analysis. *Psychometrika*, 23(2):111–136, 1958.

S. Virtanen, A. Klami, and S. Kaski. Bayesian CCA via group sparsity. In L. Getoor and T. Scheffer, editors, *Proceedings of the International Conference in Machine Learning*, volume 28, 2011.

Y. Weiss, B. Schölkopf, and J. C. Platt, editors. *Advances in Neural Information Processing Systems*, volume 18, Cambridge, MA, 2006. MIT Press.

C. K. I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In Leen et al. (2001), pages 682–688.

A. G. Wilson, D. A. Knowles, and Z. Ghahramani. Gaussian process regression networks. In J. Langford and J. Pineau, editors, *Proceedings of the International Conference in Machine Learning*, volume 29, San Francisco, CA, 2012. Morgan Kauffman. To appear.