Deep Gaussian Processes

Learning Abstract Features with Gaussian Process Models

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

Aalto University

25th January 2013

Outline

Motivation

Larger Datasets

Bayesian GP-LVM

Deep GPs

Conclusions

Outline

Motivation for Non-Linear Dimensionality Reduction

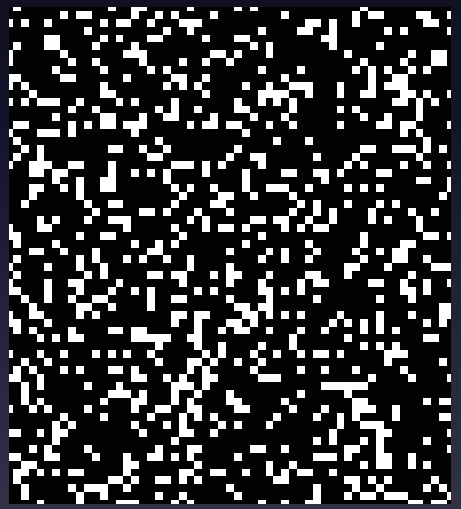
USPS Data Set Handwritten Digit

- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!

Motivation for Non-Linear Dimensionality Reduction

USPS Data Set Handwritten Digit

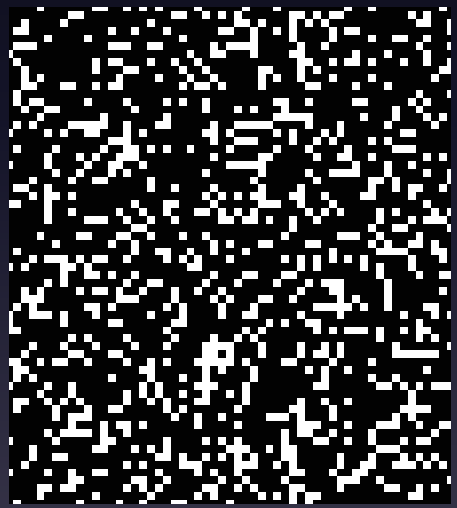
- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!



Motivation for Non-Linear Dimensionality Reduction

USPS Data Set Handwritten Digit

- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!



Motivation for Non-Linear Dimensionality Reduction

USPS Data Set Handwritten Digit

- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

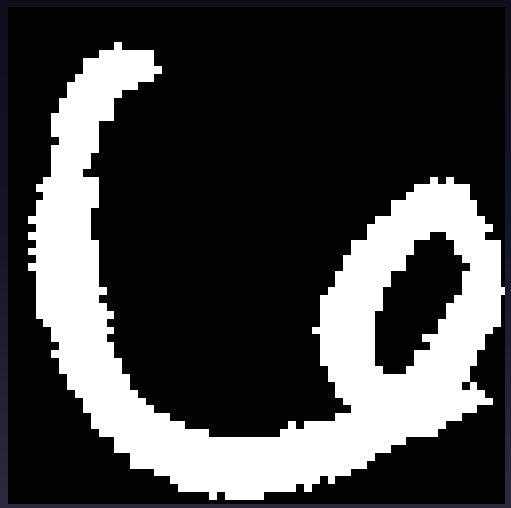
Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

Simple Model of Digit

Rotate a 'Prototype'

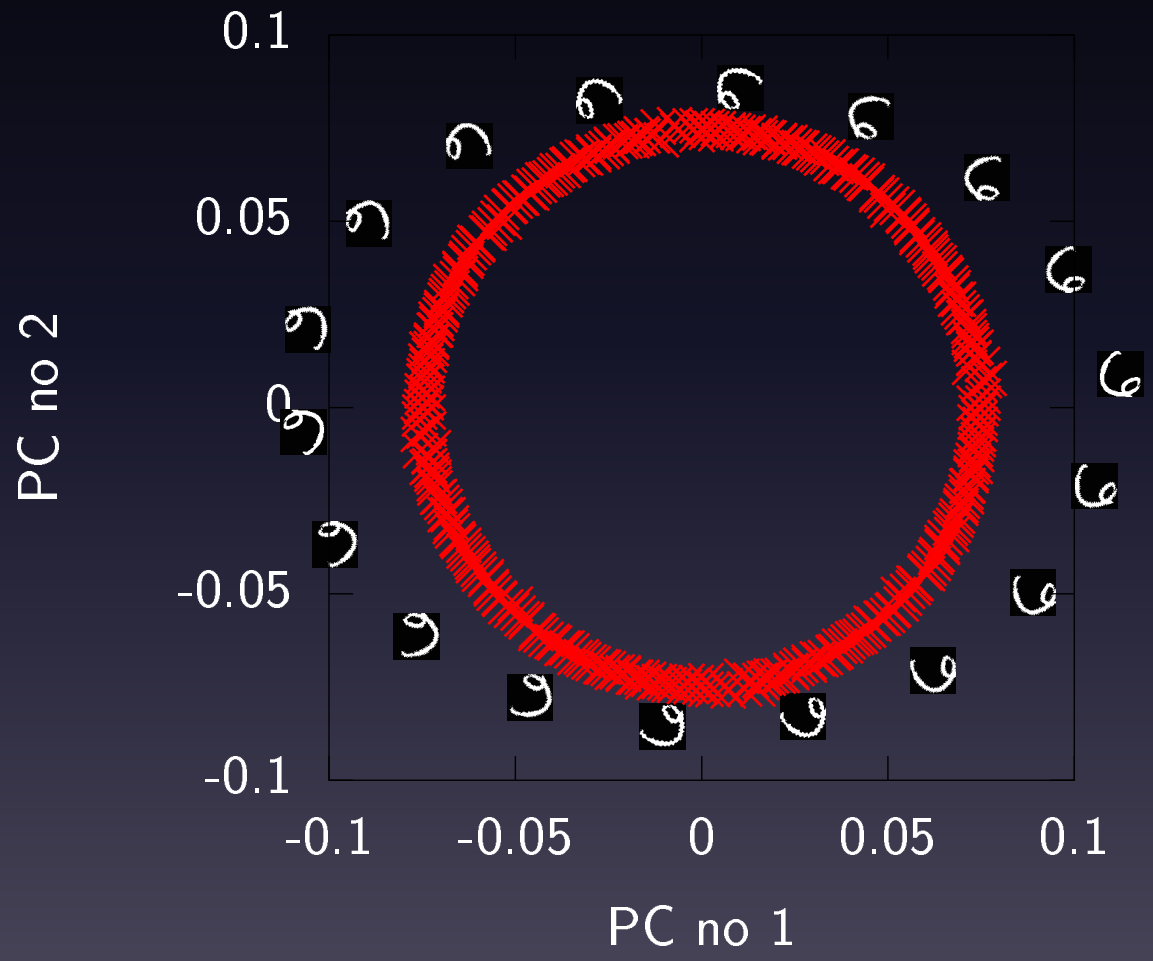


MATLAB Demo

```
demDigitsManifold([1 2], 'all')
```

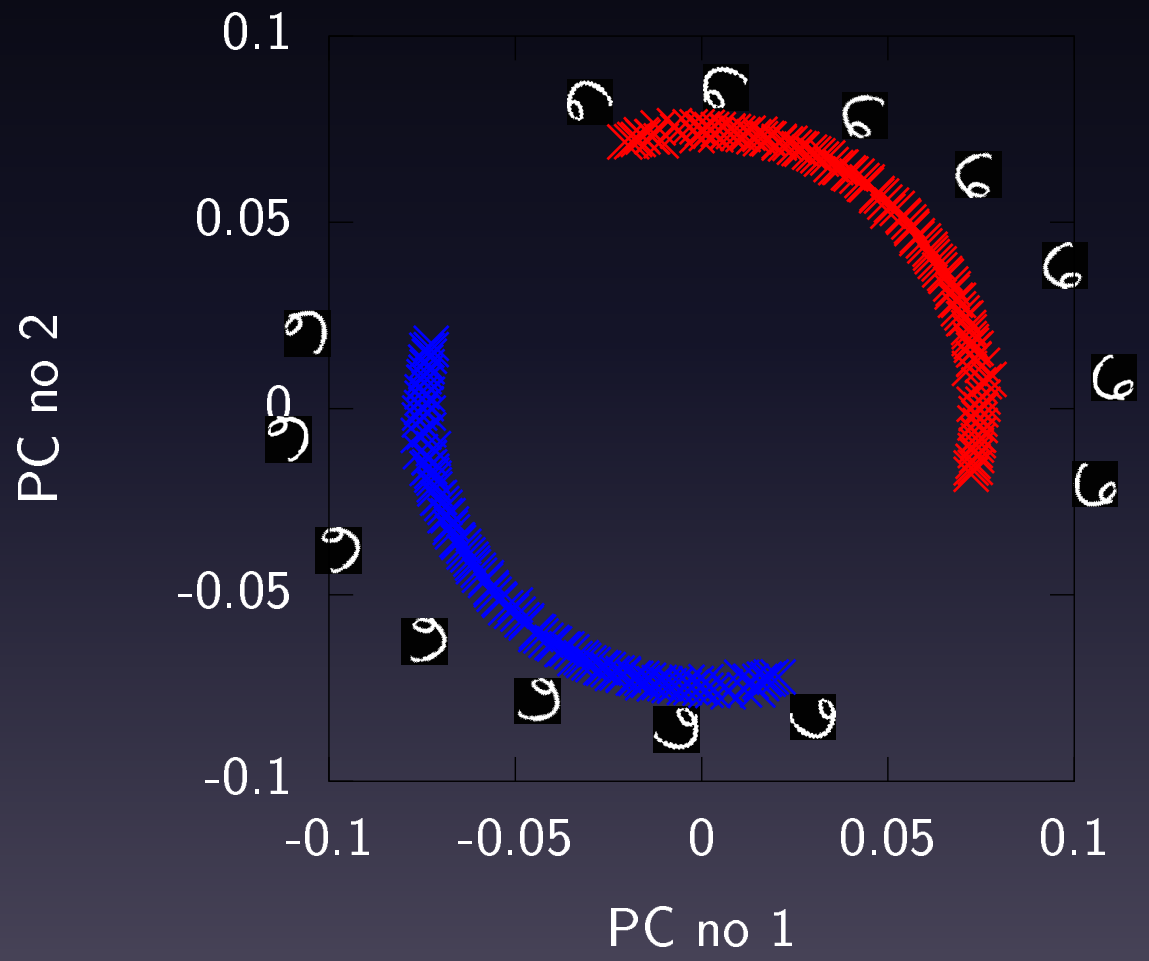
MATLAB Demo

```
demDigitsManifold([1 2], 'all')
```



MATLAB Demo

```
demDigitsManifold([1 2], 'sixnine')
```



Low Dimensional Manifolds

Pure Rotation is too Simple

- In practice the data may undergo several distortions.
  - *e.g.* digits undergo 'thinning', translation and rotation.
- For data with 'structure':
  - we expect fewer distortions than dimensions;
  - we therefore expect the data to live on a lower dimensional manifold.
- Conclusion: deal with high dimensional data by looking for lower dimensional non-linear embedding.

Existing Methods

Spectral Approaches

- Classical Multidimensional Scaling (MDS) (Mardia et al., 1979).
  - Uses eigenvectors of similarity matrix.
    - Isomap (Tenenbaum et al., 2000) is MDS with a particular proximity measure.
  - Kernel PCA (Schölkopf et al., 1998)
    - Provides a representation and a mapping — dimensional expansion.
    - Mapping is implied throught he use of a kernel function as a similarity matrix.
  - Locally Linear Embedding (Roweis and Saul, 2000).
    - Looks to preserve locally linear relationships in a low dimensional space.

Existing Methods II

Iterative Methods

- Multidimensional Scaling (MDS)
  - Iterative optimisation of a stress function (Kruskal, 1964).
  - Sammon Mappings (Sammon, 1969).
    - Strictly speaking not a mapping — similar to iterative MDS.
- NeuroScale (Lowe and Tipping, 1997)
  - Augmentation of iterative MDS methods with a mapping.

Existing Methods III

Probabilistic Approaches

- Probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998)
  - A linear method.
- Density Networks (MacKay, 1995)
  - Use importance sampling and a multi-layer perceptron.
- Generative Topographic Mapping (GTM) (Bishop et al., 1998)
  - Uses a grid based sample and an RBF network.

Existing Methods III

Probabilistic Approaches

- Probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998)
  - A linear method.
- Density Networks (MacKay, 1995)
  - Use importance sampling and a multi-layer perceptron.
- Generative Topographic Mapping (GTM) (Bishop et al., 1998)
  - Uses a grid based sample and an RBF network.

Existing Methods III

Probabilistic Approaches

- Probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998)
  - A linear method.
- Density Networks (MacKay, 1995)
  - Use importance sampling and a multi-layer perceptron.
- Generative Topographic Mapping (GTM) (Bishop et al., 1998)
  - Uses a grid based sample and an RBF network.

Existing Methods III

Probabilistic Approaches

- Probabilistic PCA (Tipping and Bishop, 1999; Roweis, 1998)
  - A linear method.
- Density Networks (MacKay, 1995)
  - Use importance sampling and a multi-layer perceptron.
- Generative Topographic Mapping (GTM) (Bishop et al., 1998)
  - Uses a grid based sample and an RBF network.

Difficulty for Probabilistic Approaches

- Propagate a probability distribution through a non-linear mapping.

The New Model

A Probabilistic Non-linear PCA

- PCA has a probabilistic interpretation (Tipping and Bishop, 1999; Roweis, 1998).
- It is difficult to 'non-linearise'.

Dual Probabilistic PCA

- We present a new probabilistic interpretation of PCA (Lawrence, 2005).
- This interpretation can be made non-linear.
- The result is non-linear probabilistic PCA.

Notation

q— dimension of latent/embedded space

p— dimension of data space

n— number of data points

centred data, $\mathbf{Y} = [\mathbf{y}_{1,:}, \ldots, \mathbf{y}_{n,:}]^\top = [\mathbf{y}_{:,1}, \ldots, \mathbf{y}_{:,p}] \in \Re^{n \times p}$

latent variables, $\mathbf{X} = [\mathbf{x}_{1,:}, \ldots, \mathbf{x}_{n,:}]^\top = [\mathbf{x}_{:,1}, \ldots, \mathbf{x}_{:,q}] \in \Re^{n \times q}$

mapping matrix, $\mathbf{W} \in \Re^{p \times q}$

$\mathbf{a}_{i,:}$ is a vector from the ith row of a given matrix $\mathbf{A}$

$\mathbf{a}_{:,j}$ is a vector from the jth row of a given matrix $\mathbf{A}$

Reading Notation

X and Y are *design matrices*

- Covariance given by $n^{-1}\mathbf{Y}^\top\mathbf{Y}$.
- Inner product matrix given by $\mathbf{Y}\mathbf{Y}^\top$.

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

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

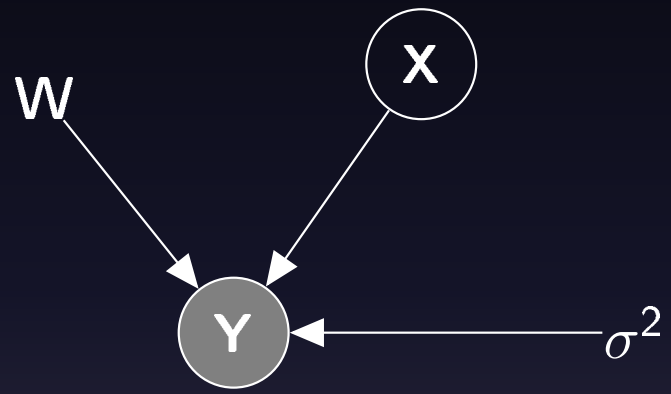
where

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

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*.

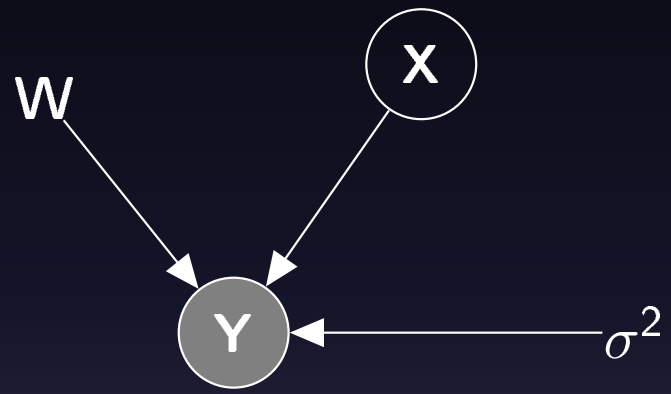


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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*.

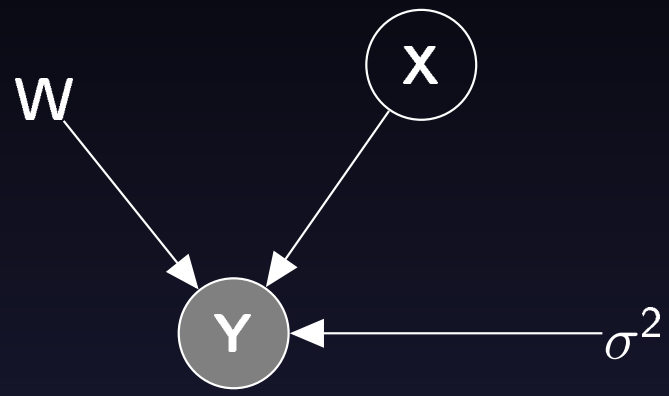


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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*.



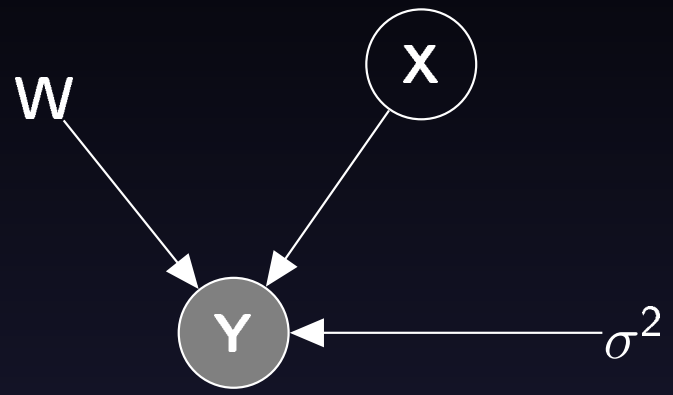
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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

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*.



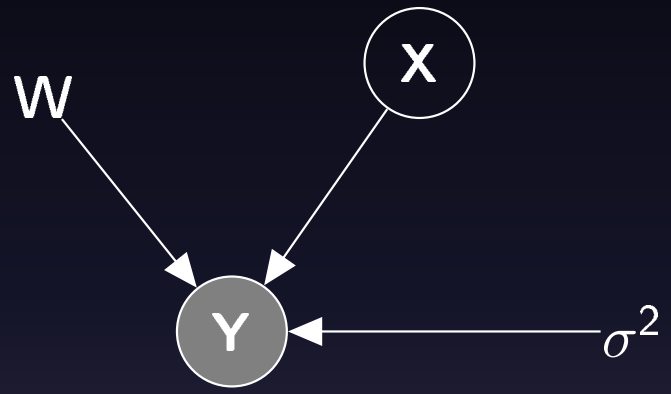
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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

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

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)

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

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

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

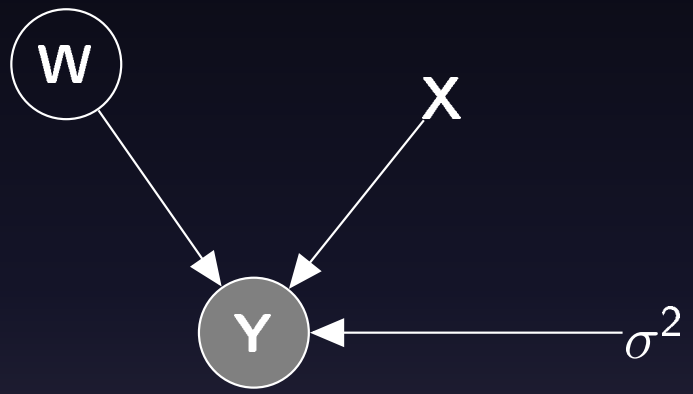
$$\mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^{\top}, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

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*.

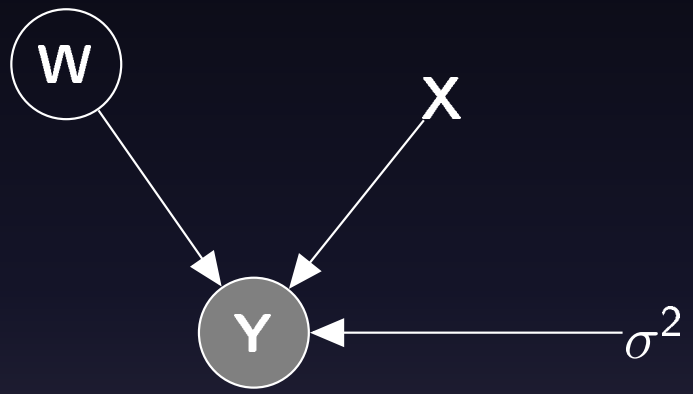


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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*.

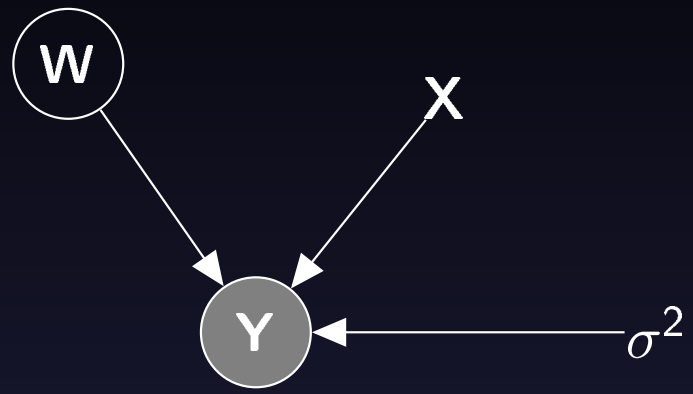


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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*.



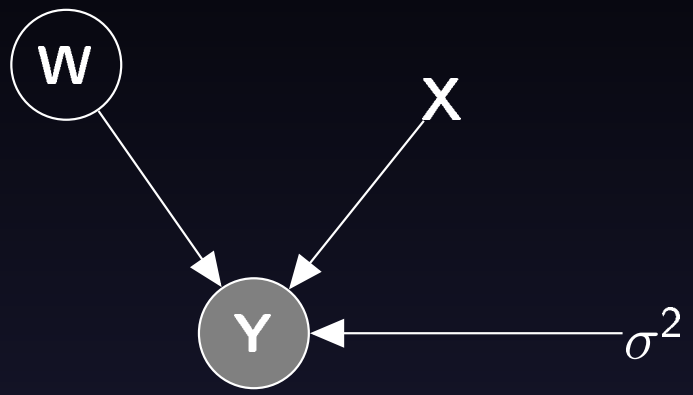
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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

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*, $\mathbf{W}$.
  - Integrate out *parameters*.



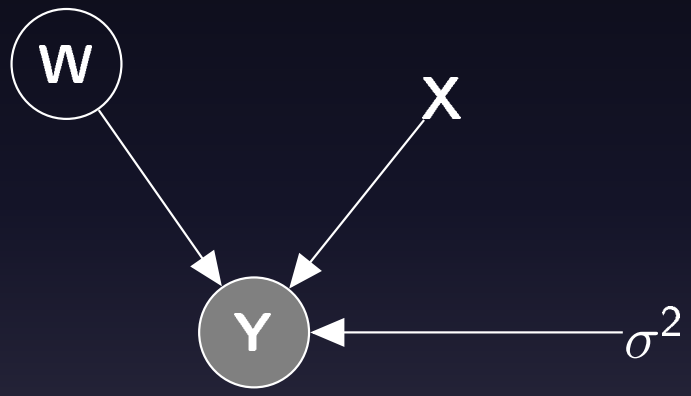
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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

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

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)

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

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

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)

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

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

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)

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

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

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

- Equivalence is from

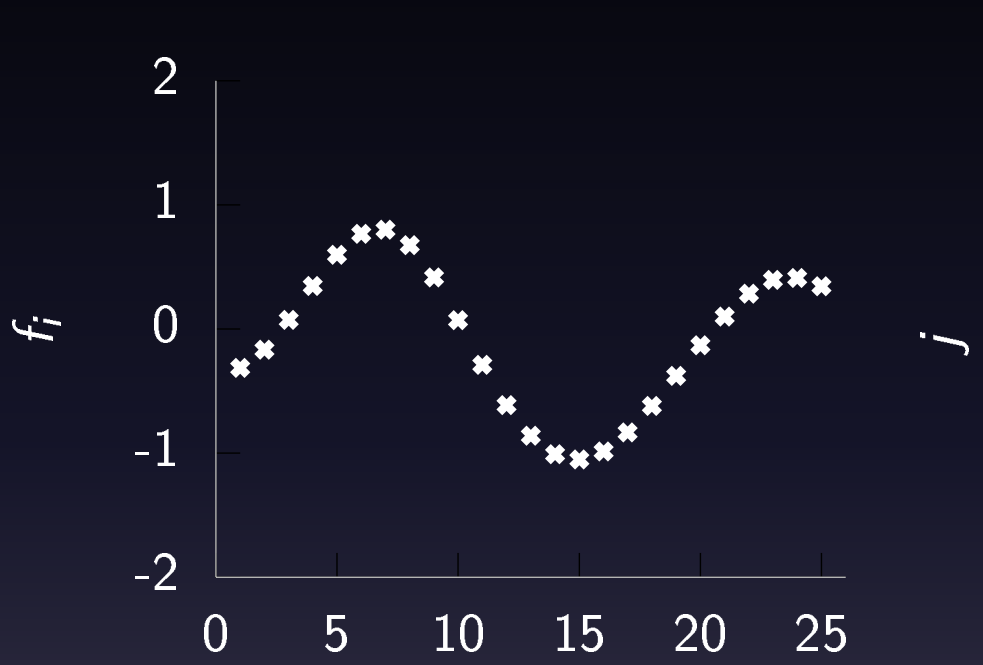
$$\mathbf{U}_q = \mathbf{Y}^\top \mathbf{U}'_q \mathbf{\Lambda}_q^{-\frac{1}{2}}$$

Sampling a Function

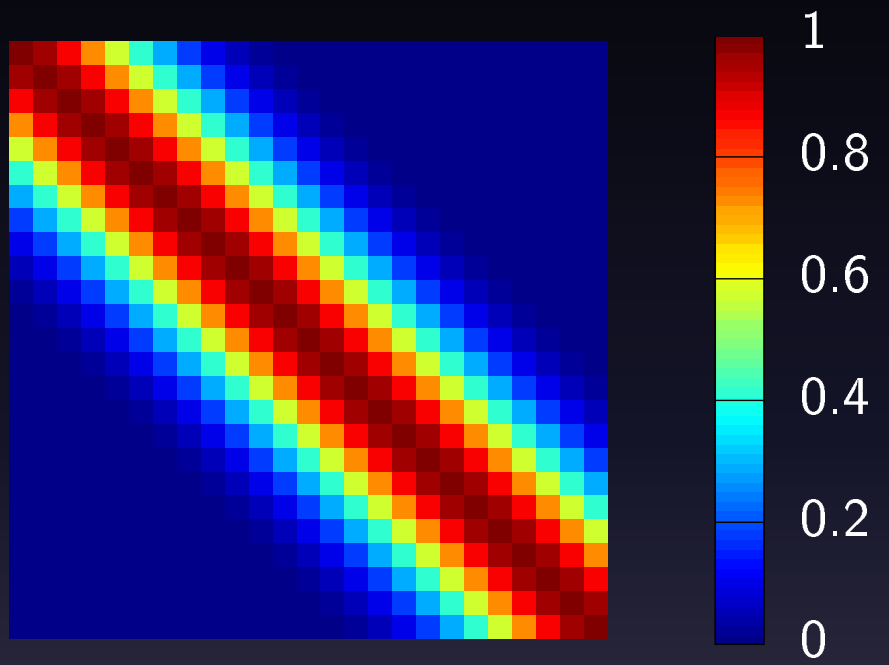
Multi-variate Gaussians

- We will consider a Gaussian with a particular structure of covariance matrix.
- Generate a single sample from this 25 dimensional Gaussian distribution, $\mathbf{f} = [f_1, f_2 \dots f_{25}]$.
- We will plot these points against their index.

Gaussian Distribution Sample



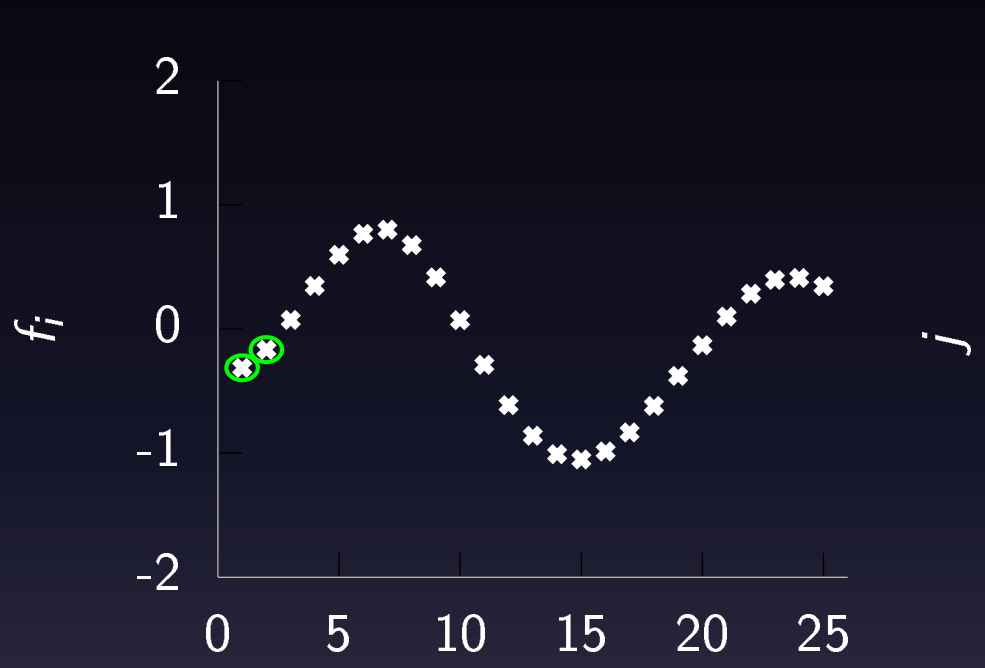
(a) A 25 dimensional correlated random variable (values ploted against index)



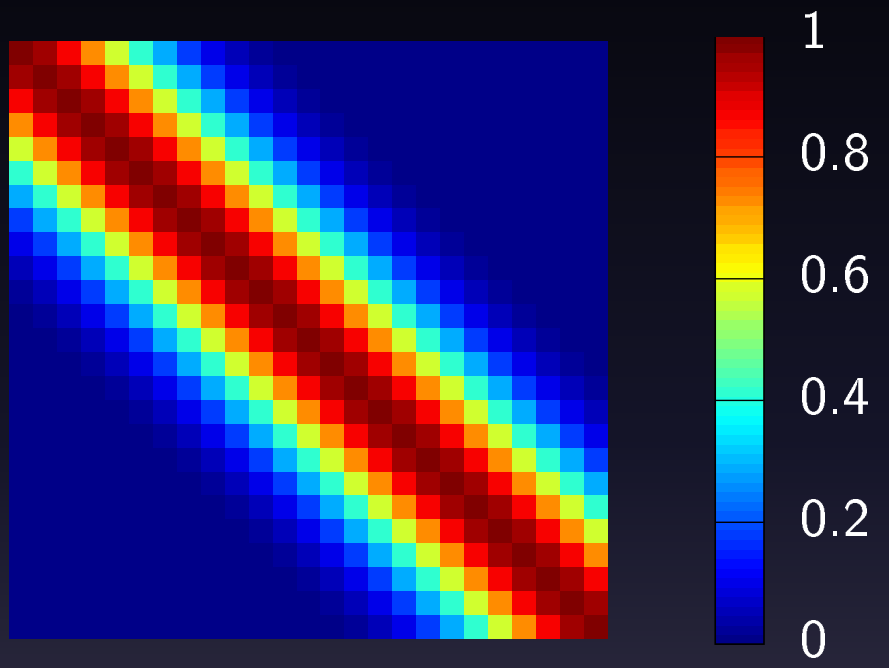
(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample



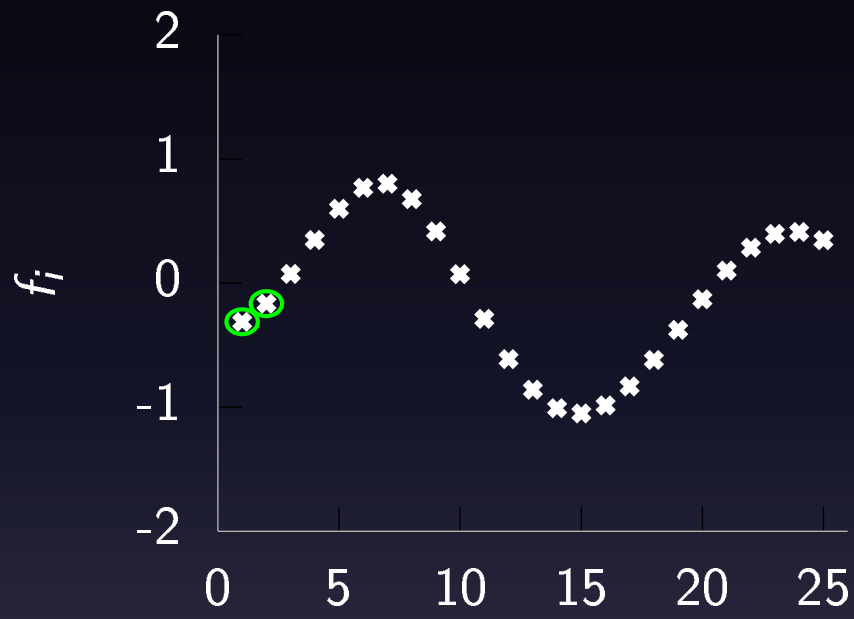
(a) A 25 dimensional correlated random variable (values ploted against index)



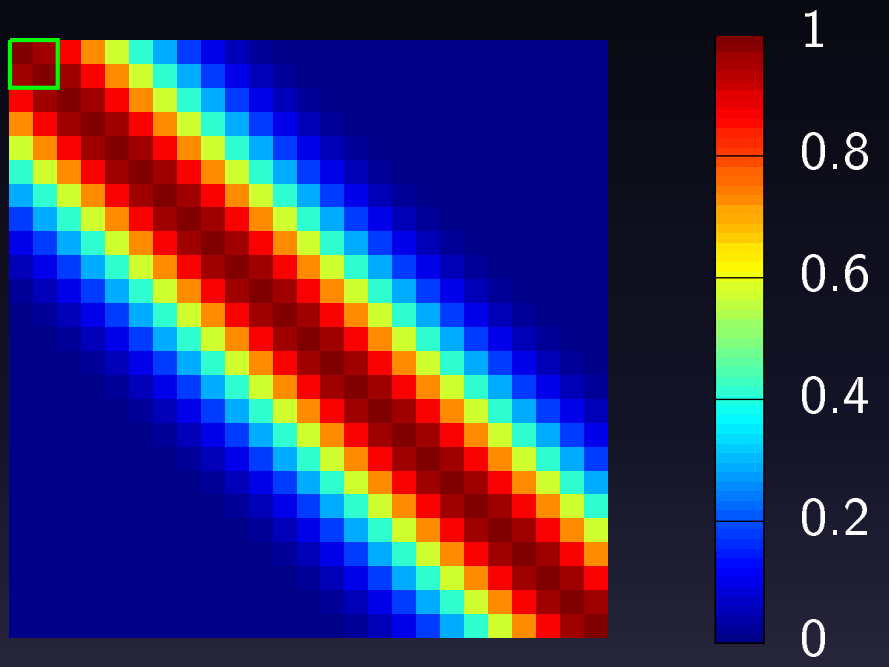
(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample



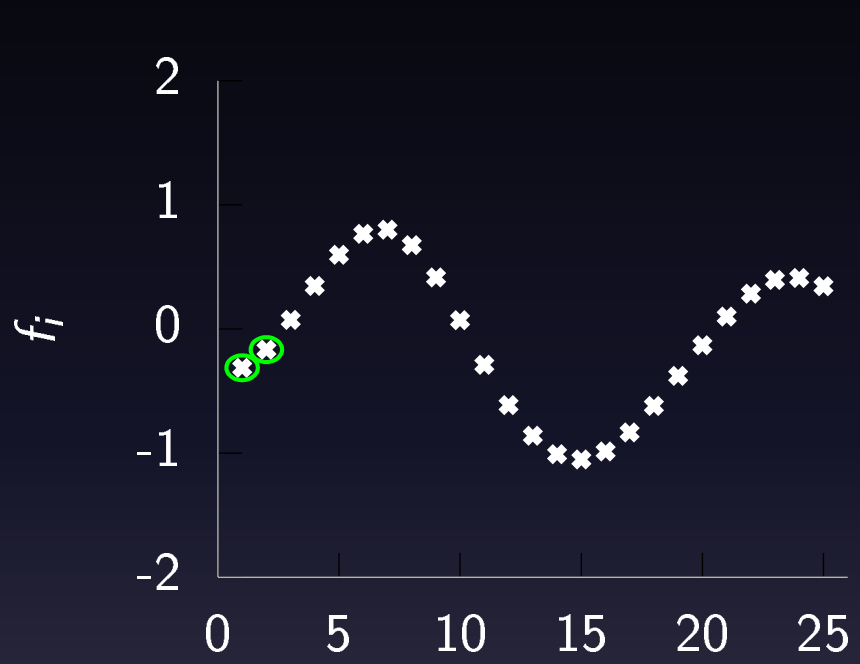
(a) A 25 dimensional correlated random variable (values ploted against index)



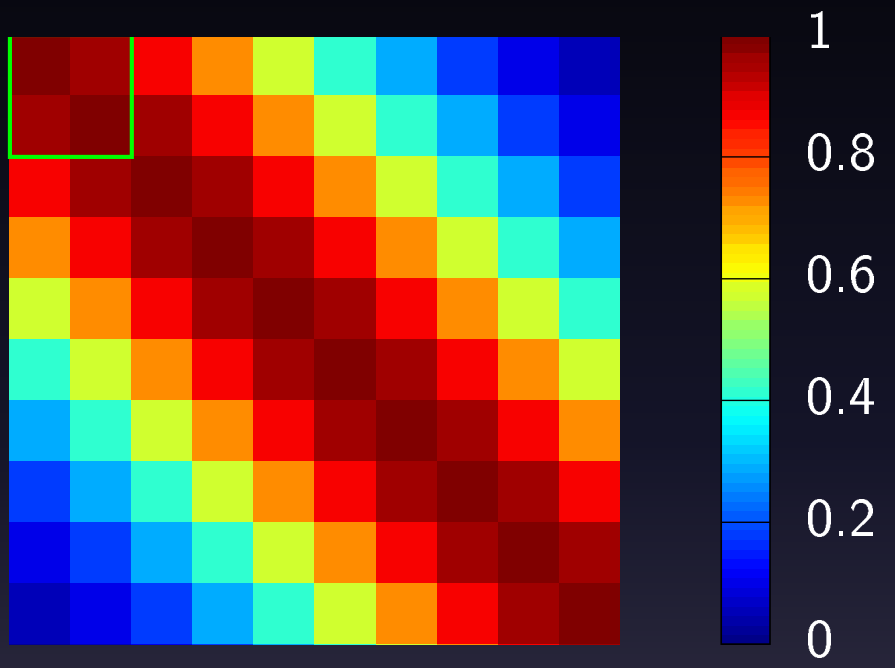
(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample



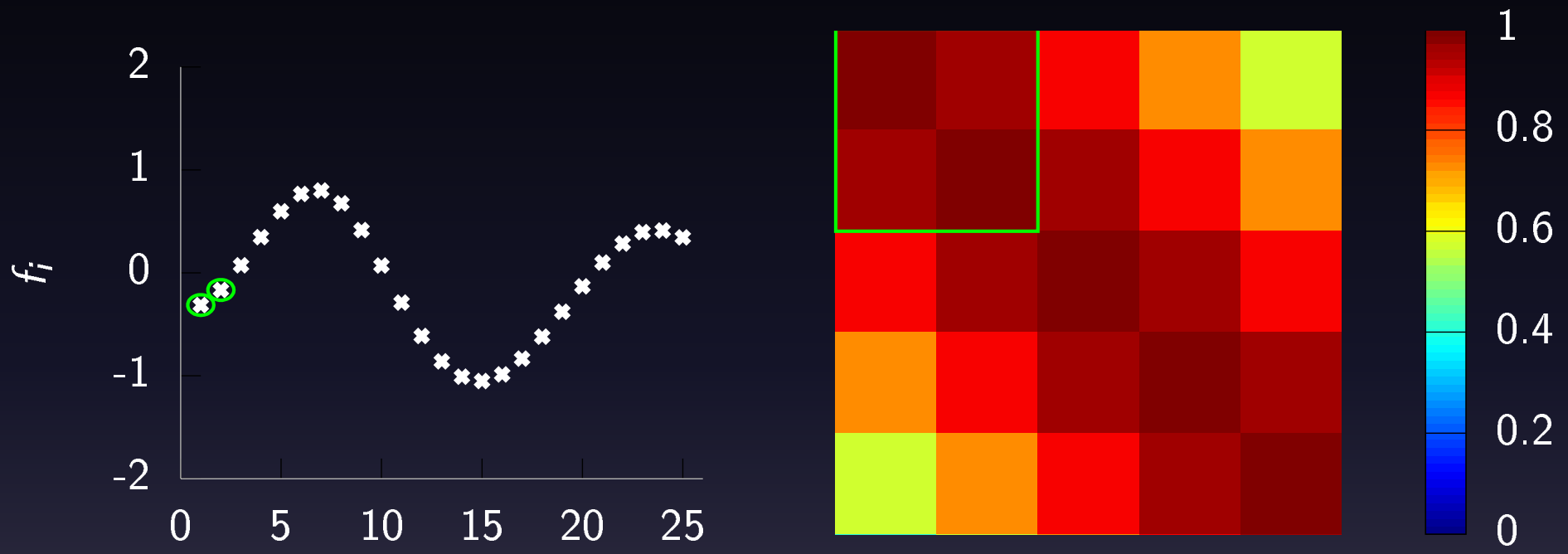
(a) A 25 dimensional correlated random variable (values ploted against index)



(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

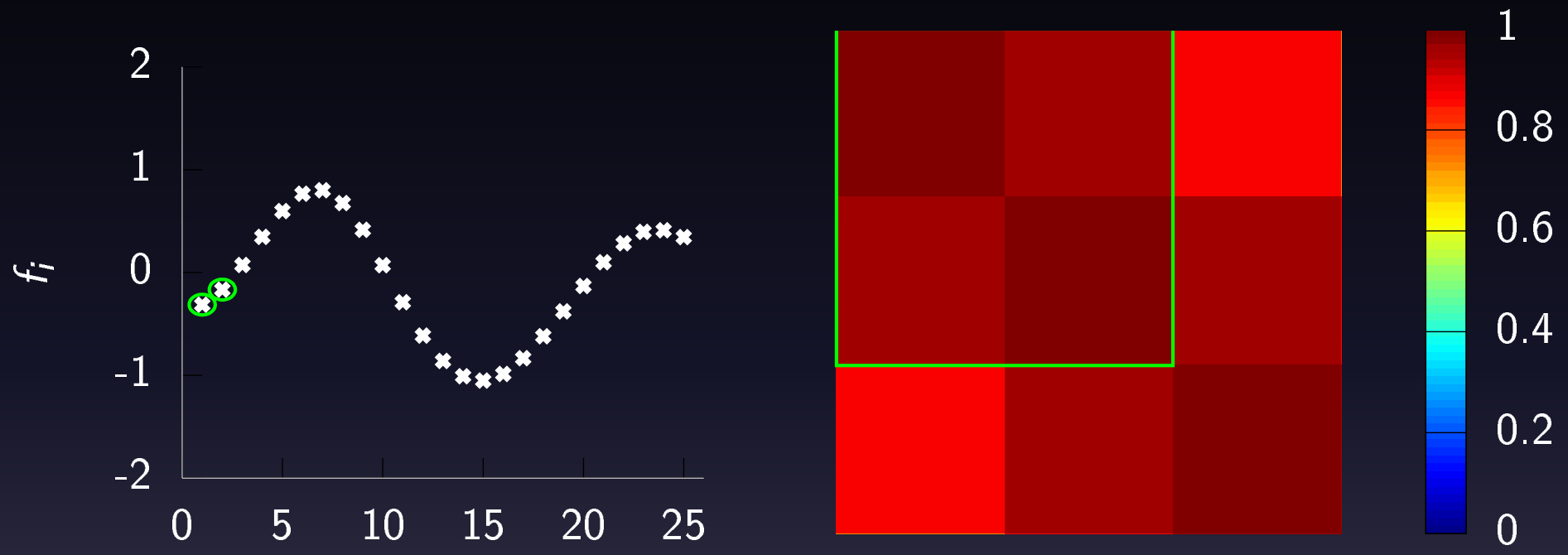


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

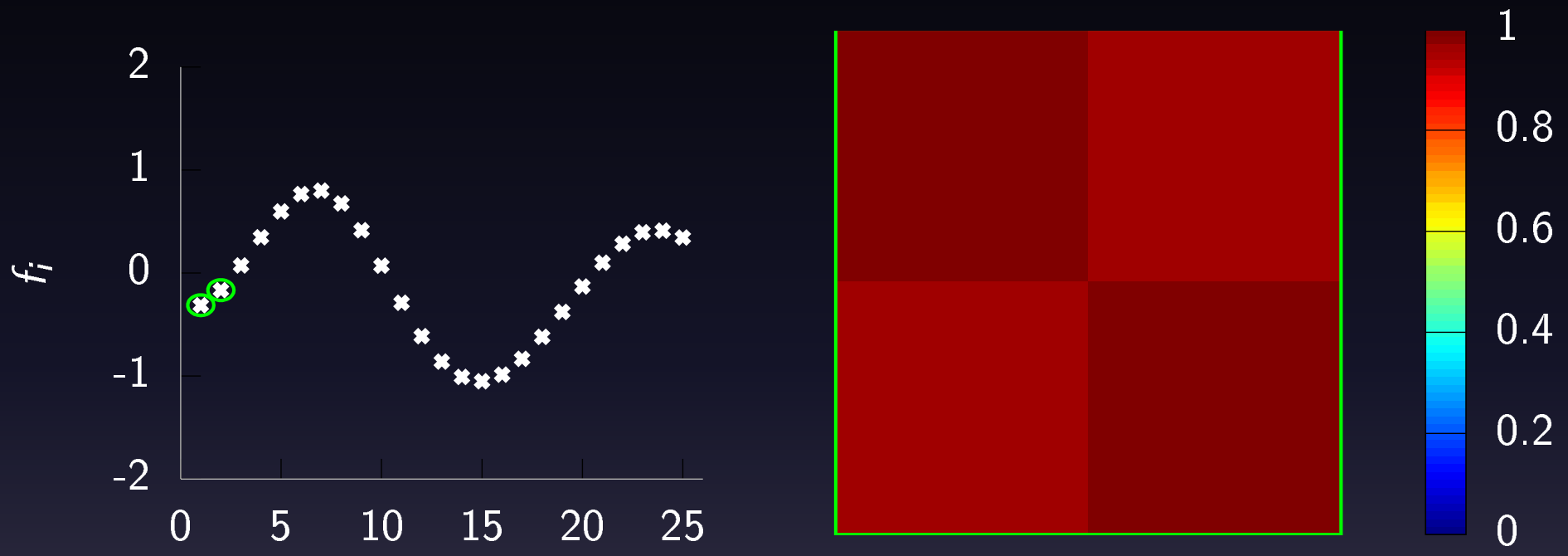


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample

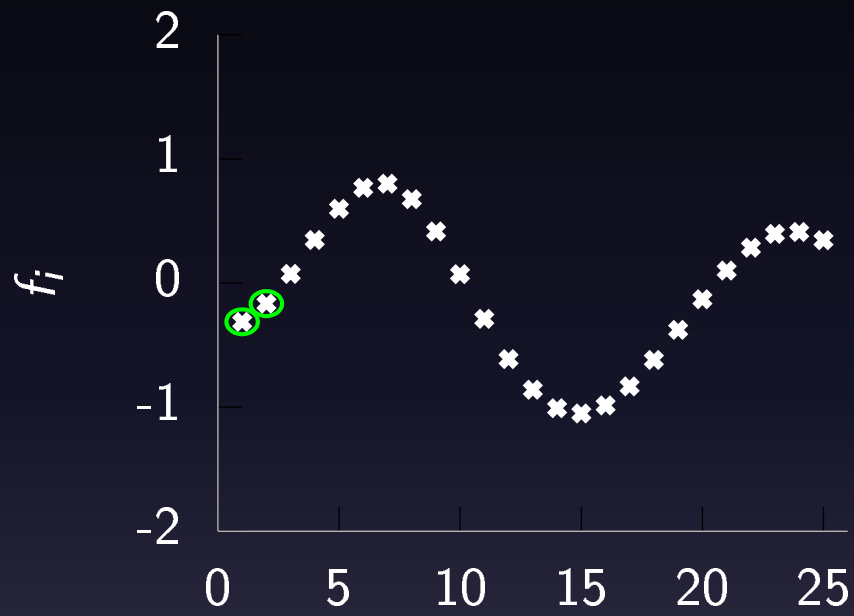


(a) A 25 dimensional correlated random variable (values ploted against index)

(b) colormap showing correlations between dimensions.

Figure: A sample from a 25 dimensional Gaussian distribution.

Gaussian Distribution Sample



(a) A 25 dimensional correlated random variable (values ploted against index)

$$\begin{bmatrix} 1 & 0.96587 \\ 0.96587 & 1 \end{bmatrix}$$

(b) correlation between f_1 and f_2.

Figure: A sample from a 25 dimensional Gaussian distribution.

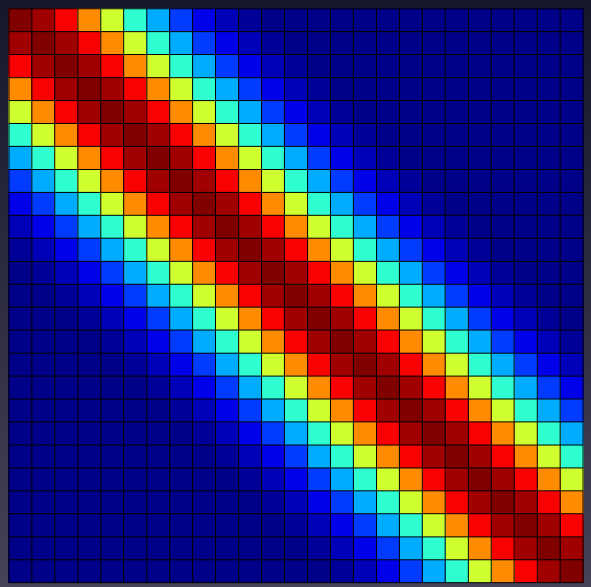
Covariance Functions

Where did this covariance matrix come from?

Exponentiated Quadratic Kernel Function (RBF, Squared Exponential, Gaussian)

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \alpha \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function $\mathbf{x}$.
- For the example above it was based on Euclidean distance.
- The covariance function is also know as a kernel.



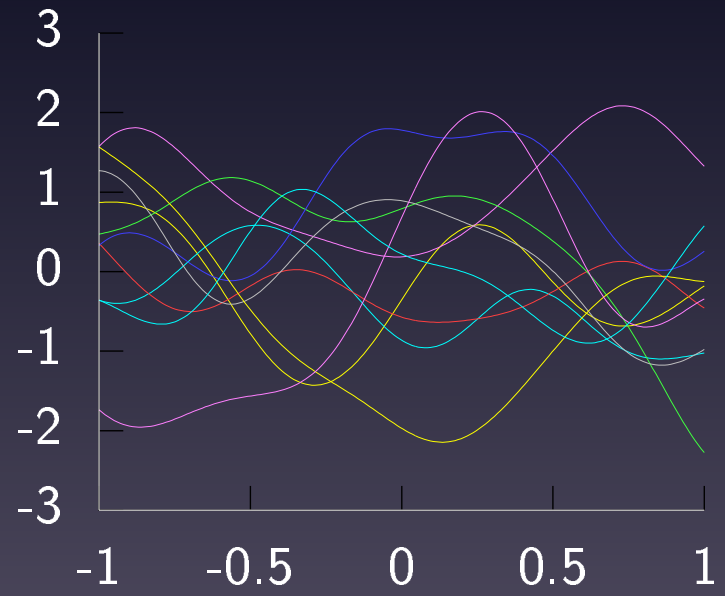
Covariance Functions

Where did this covariance matrix come from?

Exponentiated Quadratic Kernel Function (RBF, Squared Exponential, Gaussian)

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \alpha \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function $\mathbf{x}$.
- For the example above it was based on Euclidean distance.
- The covariance function is also know as a kernel.



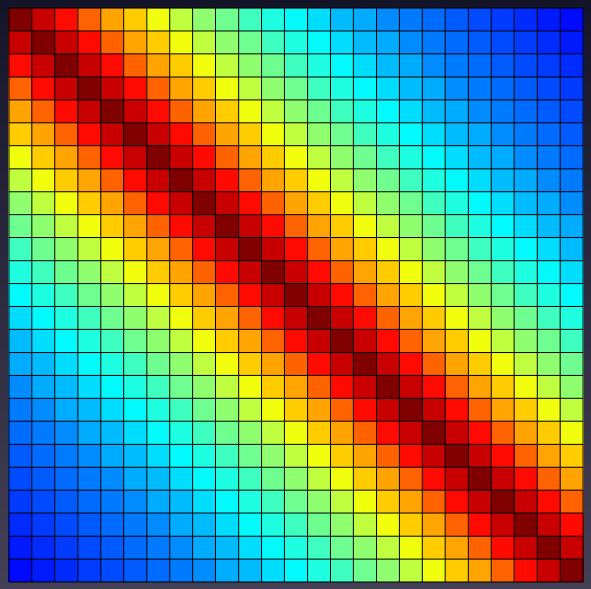
Covariance Functions

Where did this covariance matrix come from?

Ornstein-Uhlenbeck (stationary Gauss-Markov) covariance function

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \alpha \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|}{2\ell^2}\right)$$

- Covariance matrix is built using the *inputs* to the function $\mathbf{x}$.



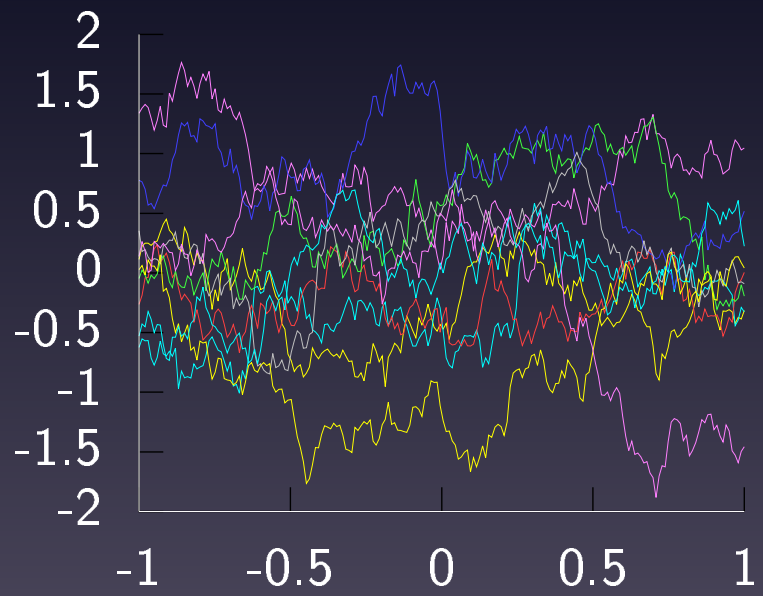
Covariance Functions

Where did this covariance matrix come from?

Ornstein-Uhlenbeck (stationary Gauss-Markov) covariance function

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \alpha \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|}{2\ell^2}\right)$$

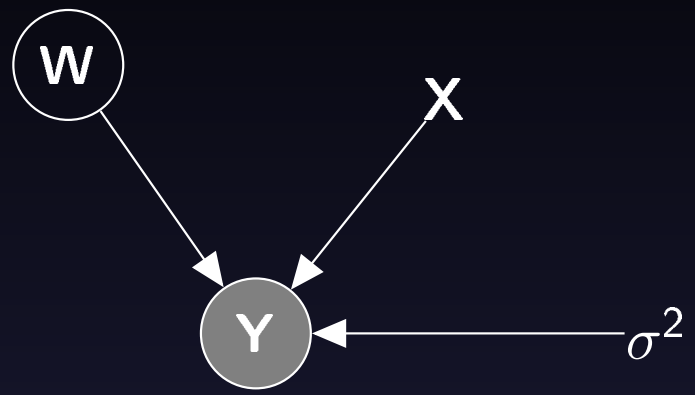
- Covariance matrix is built using the *inputs* to the function $\mathbf{x}$.



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*.



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

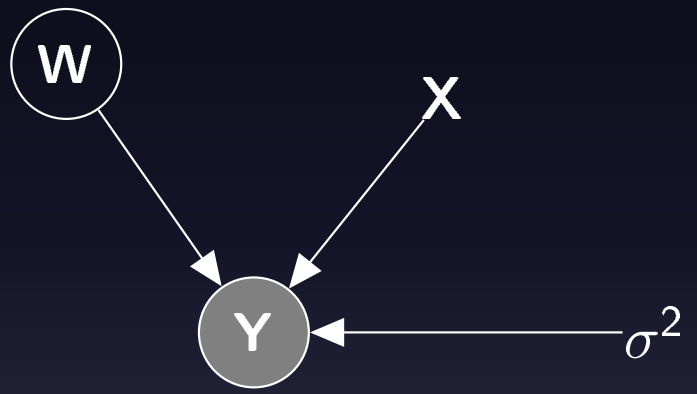
$$p(\mathbf{W}) = \prod_{i=1}^{p} \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0}, \mathbf{I}\right)$$

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

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).

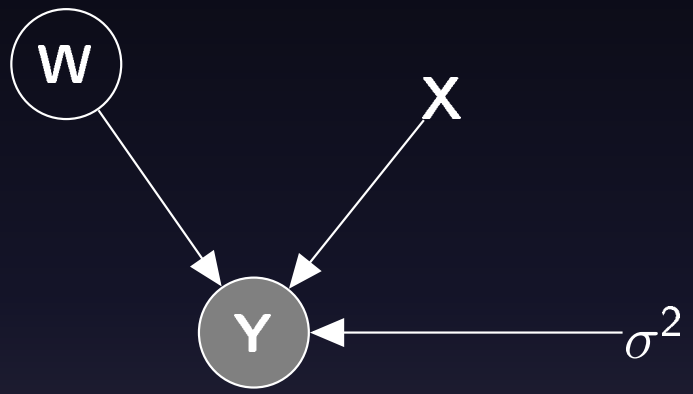


$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right)$$

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).



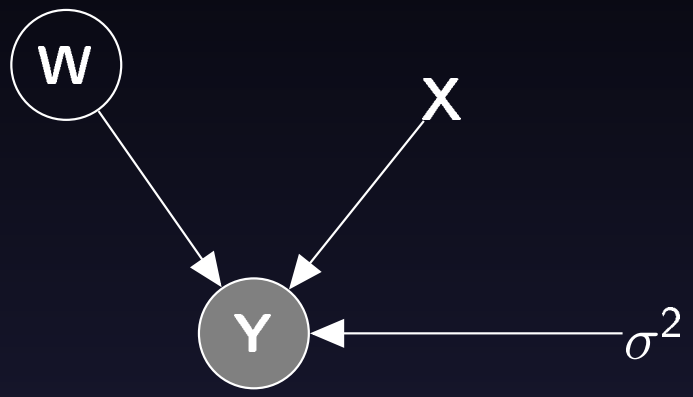
$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K})$$

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

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).



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

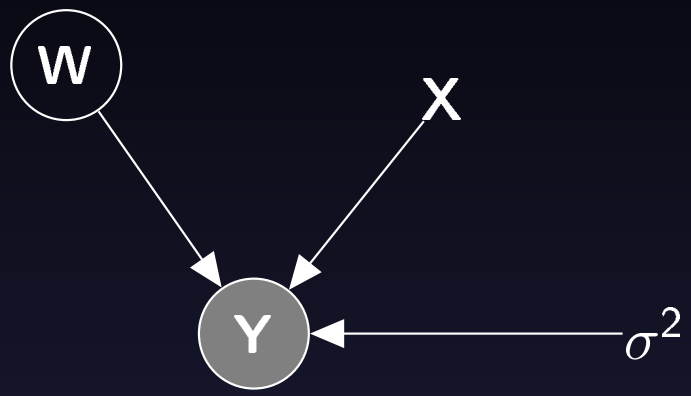
$$\mathbf{K} = \mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}$$

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).



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

$$\mathbf{K} = ?$$

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

Non-linear Latent Variable Models

Exponentiated Quadratic (EQ) Covariance

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

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

- 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.

Applications

Style Based Inverse Kinematics

- Facilitating animation through modeling human motion with the GP-LVM (Grochow et al., 2004)

Tracking

- Tracking using models of human motion learnt with the GP-LVM (Urtasun et al., 2005, 2006)

Assisted Animation

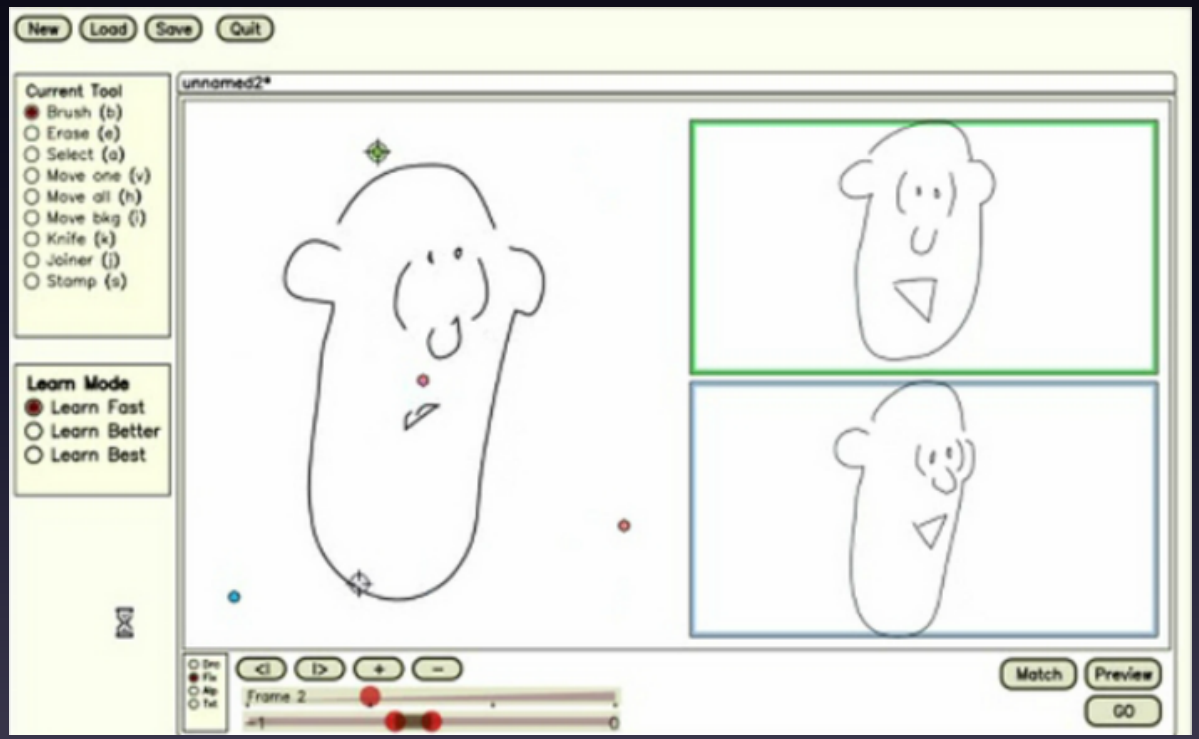
- Generalizing drawings to animate them. (Baxter and Anjyo, 2006)

Shape Models

- Inferring shape (e.g. pose from silhouette). (Ek et al., 2008b,a; Priacuriu and Reid, 2011a,b)

Example: Latent Doodle Space

(Baxter and Anjyo, 2006)



http://vimeo.com/3235882

Example: Latent Doodle Space

(Baxter and Anjyo, 2006)

Generalization with much less Data than Dimensions

- Powerful uncertainly handling of GPs leads to surprising properties.
- Non-linear models can be used where there are fewer data points than dimensions *without overfitting*.

Outline

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

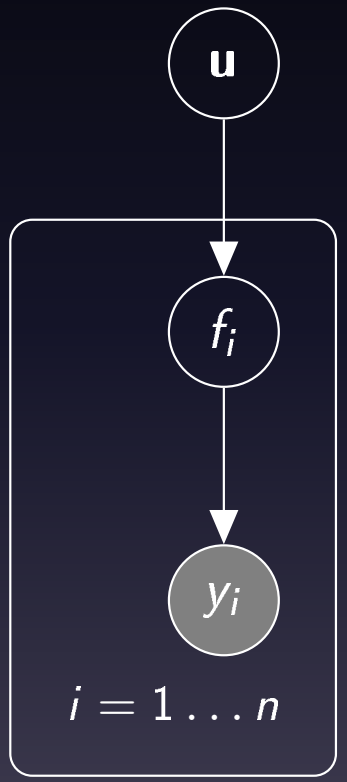
Assume that relationship is through $\mathbf{f}$.

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

Augmented Variable Model

Very often likelihood factorizes.

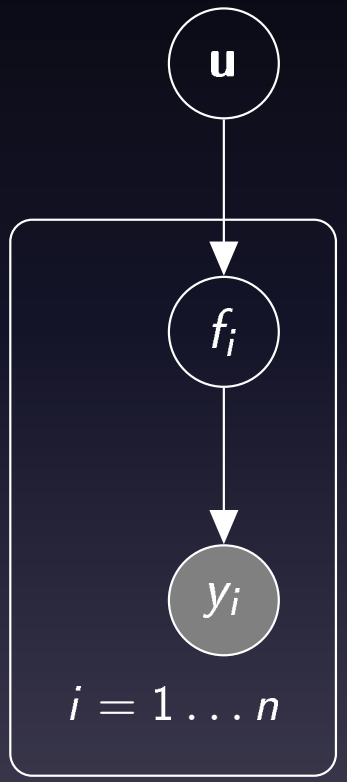
$$p(\mathbf{y}) = \int \prod_{i=1}^{n} p(y_i|f_i)p(\mathbf{f}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{f}\mathrm{d}\mathbf{u}$$



Augmented Variable Model

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

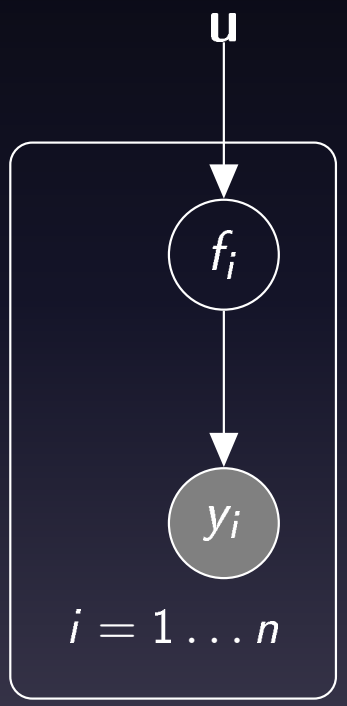
$$p(\mathbf{y}) = \int \int \prod_{i=1}^{n} p(y_i|f_i)p(\mathbf{f}|\mathbf{u})\mathrm{d}\mathbf{f}p(\mathbf{u})\mathrm{d}\mathbf{u}$$



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 $p(\mathbf{y}|\mathbf{u})$

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

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

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

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

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

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

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

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

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \sum_{i=1}^{n} \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \int p(\mathbf{f}|\mathbf{u}) \sum_{i=1}^{n} \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \sum_{i=1}^{n} \int p(f_i|\mathbf{u}) \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \exp \sum_{i=1}^{n} \int p(f_i|\mathbf{u}) \log p(y_i|f_i) \mathrm{d}\mathbf{f}.$$

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

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

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

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

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \prod_{i=1}^{n} \exp \left\langle \log p(y_i|f_i) \right\rangle_{p(f_i|\mathbf{u})}$$

- Then the bound factorizes.

Factorizing Likelihoods

- If the likelihood, $p(\mathbf{y}|\mathbf{f})$, factorizes

$$p(\mathbf{y}|\mathbf{u}) \geq \prod_{i=1}^{n} \exp \left\langle \log p(y_i|f_i) \right\rangle_{p(f_i|\mathbf{u})}$$

- Then the bound factorizes.

Gaussian $p(y_i|f_i)$

For Gaussian likelihoods:

$$\langle \log p(y_i|f_i) \rangle_{p(f_i|\mathbf{u})} = -\frac{1}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}\left(y_i - \langle f_i \rangle\right)^2 - \frac{1}{2\sigma^2}\left(\langle f_i^2 \rangle - \langle f_i \rangle^2\right)$$

Gaussian $p(y_i|f_i)$

For Gaussian likelihoods:

$$\langle \log p(y_i|f_i) \rangle_{p(f_i|\mathbf{u})} = -\frac{1}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}\left(y_i - \langle f_i \rangle\right)^2 - \frac{1}{2\sigma^2}\left(\langle f_i^2 \rangle - \langle f_i \rangle^2\right)$$

Implying:

$$p(y_i|\mathbf{u}) \geq \exp \langle \log c_i \rangle \, \mathcal{N}\left(y_i | \langle f_i \rangle, \sigma^2\right)$$

Gaussian Process Over $\mathbf{f}$ and $\mathbf{u}$

Define:

$$q_{i,i} = \text{var}_{p(f_i|\mathbf{u})}\left(f_i\right) = \left\langle f_i^2 \right\rangle_{p(f_i|\mathbf{u})} - \left\langle f_i \right\rangle^2_{p(f_i|\mathbf{u})}$$

We can write:

$$c_i = \exp\left(-\frac{q_{i,i}}{2\sigma^2}\right)$$

If joint distribution of $p(\mathbf{f}, \mathbf{u})$ is Gaussian then:

$$q_{i,i} = k_{i,i} - \mathbf{k}_{i,\mathbf{u}}^\top \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{k}_{i,\mathbf{u}}$$

c_i is not a function of $\mathbf{u}$ but *is* a function of $\mathbf{X}_\mathbf{u}$).

Lower Bound on Likelihood

Substitute variational bound into marginal likelihood:

$$p(\mathbf{y}) \geq \prod_{i=1}^{n} c_i \int \mathcal{N}\left(\mathbf{y} | \left\langle \mathbf{f} \right\rangle, \sigma^2 \mathbf{I}\right) p(\mathbf{u}) \mathrm{d}\mathbf{u}$$

Note that:

$$\left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u})} = \mathbf{K}_{\mathbf{f},\mathbf{u}} \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1} \mathbf{u}$$

is *linearly* dependent on $\mathbf{u}$.

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

$$\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathrm{d}\mathbf{u} \geq \prod_{i=1}^{n} c_i \int \mathcal{N}\left(\mathbf{y}|\mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{u}, \sigma^2\right) \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right) \mathrm{d}\mathbf{u}$$

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

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

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

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

Maximize log of the bound to find covariance function parameters,

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

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

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

Maximize log of the bound to find covariance function parameters,

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

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

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

Maximize log of the bound to find covariance function parameters,

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

- If the bound is normalized, the c_i terms are removed.
- This results in 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).

Deterministic Training Conditional

Making the marginalization of $\mathbf{u}$ straightforward. In the Gaussian case:

$$p(\mathbf{u}) = \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u},\mathbf{u}}\right)$$

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

Maximize log of the bound to find covariance function parameters,

- If the bound is normalized, the c_i terms are removed.
- This results in 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).

Outline

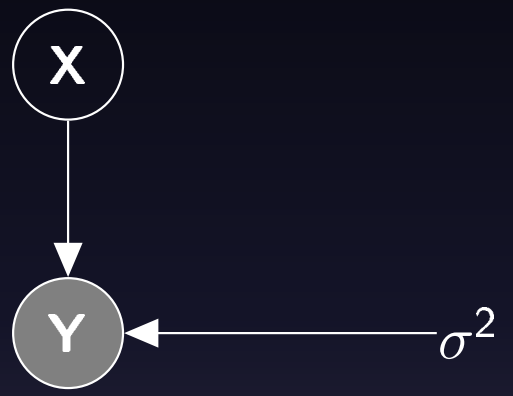
Selecting Data Dimensionality

- 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.

Integrate Mapping Function and Latent Variables

Bayesian GP-LVM

- Start with a standard GP-LVM.
- Apply standard latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.
  - Unfortunately integration is intractable.

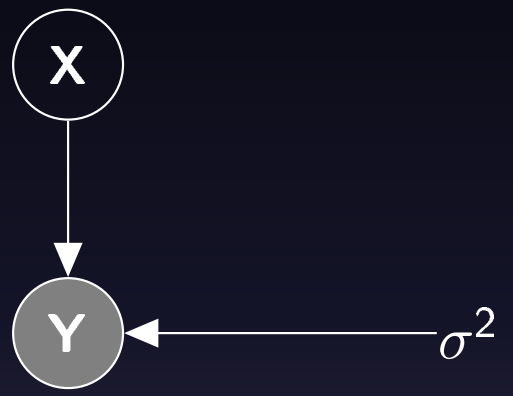


$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K})$$

Integrate Mapping Function and Latent Variables

Bayesian GP-LVM

- Start with a standard GP-LVM.
- Apply standard latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.
  - Unfortunately integration is intractable.

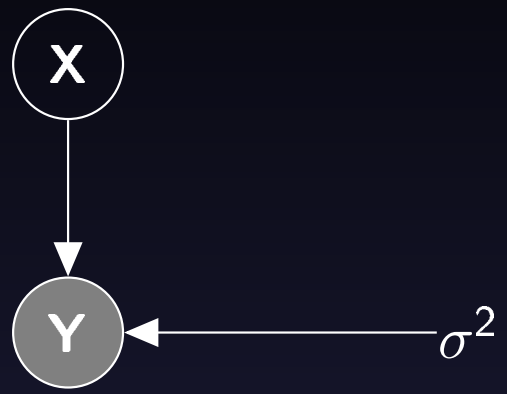


$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K})$$

Integrate Mapping Function and Latent Variables

Bayesian GP-LVM

- Start with a standard GP-LVM.
- Apply standard latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.
  - Unfortunately integration is intractable.



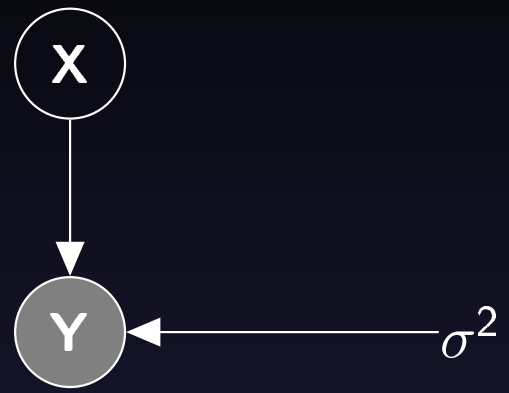
$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K})$$

$$p(\mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}\left(\mathbf{x}_{:,j}|\mathbf{0}, \alpha_i^{-2}\mathbf{I}\right)$$

Integrate Mapping Function and Latent Variables

Bayesian GP-LVM

- Start with a standard GP-LVM.
- Apply standard latent variable approach:
  - Define Gaussian prior over *latent space*, $\mathbf{X}$.
  - Integrate out *latent variables*.
  - Unfortunately integration is intractable.



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

$$p(\mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}\left(\mathbf{x}_{:,j}|\mathbf{0}, \alpha_i^{-2}\mathbf{I}\right)$$

$$p(\mathbf{Y}|\boldsymbol{\alpha}) = ??$$

Standard Variational Approach Fails

- Standard variational bound has the form:

$$\mathcal{L} = \left\langle \log p(\mathbf{y}|\mathbf{X}) \right\rangle_{q(\mathbf{X})} + \mathsf{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Requires expectation of $\log p(\mathbf{y}|\mathbf{X})$ under $q(\mathbf{X})$.

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^\top \left(\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{y} - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right| - \frac{n}{2}\log 2\pi$$

- Extremely difficult to compute because $\mathbf{K}_{\mathbf{f},\mathbf{f}}$ is dependent on $\mathbf{X}$ and appears in the inverse.

Standard Variational Approach Fails

- Standard variational bound has the form:

$$\mathcal{L} = \left\langle \log p(\mathbf{y}|\mathbf{X}) \right\rangle_{q(\mathbf{X})} + \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Requires expectation of $\log p(\mathbf{y}|\mathbf{X})$ under $q(\mathbf{X})$.

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^\top \left(\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{y} - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right| - \frac{n}{2}\log 2\pi$$

- Extremely difficult to compute because $\mathbf{K}_{\mathbf{f},\mathbf{f}}$ is dependent on $\mathbf{X}$ and appears in the inverse.

Standard Variational Approach Fails

- Standard variational bound has the form:

$$\mathcal{L} = \left\langle \log p(\mathbf{y}|\mathbf{X}) \right\rangle_{q(\mathbf{X})} + \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Requires expectation of $\log p(\mathbf{y}|\mathbf{X})$ under $q(\mathbf{X})$.

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^\top \left(\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{y} - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I}\right| - \frac{n}{2}\log 2\pi$$

- Extremely difficult to compute because $\mathbf{K}_{\mathbf{f},\mathbf{f}}$ is dependent on $\mathbf{X}$ and appears in the inverse.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$p(\mathbf{y}) \geq \prod_{i=1}^{n} c_i \int \mathcal{N}\left(\mathbf{y} | \left\langle \mathbf{f} \right\rangle, \sigma^2 \mathbf{I}\right) p(\mathbf{u}) \mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.
- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$p(\mathbf{y}|\mathbf{X}) \geq \prod_{i=1}^{n} c_i \int \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.
- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$\int p(\mathbf{y}|\mathbf{X})p(\mathbf{X})\mathrm{d}\mathbf{X} \geq \int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) \mathrm{d}\mathbf{X} p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.
- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$\int p(\mathbf{y}|\mathbf{X})p(\mathbf{X})\mathrm{d}\mathbf{X} \geq \int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2\mathbf{I}\right) \mathrm{d}\mathbf{X} p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.
- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$\int p(\mathbf{y}|\mathbf{X})p(\mathbf{X})\mathrm{d}\mathbf{X} \geq \int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}_i \left| \left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2 \mathbf{I}\right.\right) \mathrm{d}\mathbf{X} p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.

$$\int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}_i \left| \left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2 \mathbf{I}\right.\right) p(\mathbf{X})\mathrm{d}\mathbf{X}$$

$$\geq \left\langle \sum_{i=1}^{n} \log c_i \right\rangle_{q(\mathbf{X})} + \left\langle \log \mathcal{N}\left(\mathbf{y}_i \left| \left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2 \mathbf{I}\right.\right) \right\rangle_{q(\mathbf{X})} + \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Variational Bayesian GP-LVM

- Consider collapsed variational bound,

$$\int p(\mathbf{y}|\mathbf{X})p(\mathbf{X})\mathrm{d}\mathbf{X} \geq \int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}_i \left| \left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2 \mathbf{I}\right.\right) \mathrm{d}\mathbf{X} p(\mathbf{u})\mathrm{d}\mathbf{u}$$

- Apply variational lower bound to the inner integral.

$$\int \prod_{i=1}^{n} c_i \mathcal{N}\left(\mathbf{y}_i \left| \left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2 \mathbf{I}\right.\right) p(\mathbf{X})\mathrm{d}\mathbf{X}$$

$$\geq \left\langle \sum_{i=1}^{n} \log c_i \right\rangle_{q(\mathbf{X})} + \left\langle \log \mathcal{N}\left(\mathbf{y}_i \left| \left\langle \mathbf{f} \right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{X})}, \sigma^2 \mathbf{I}\right.\right) \right\rangle_{q(\mathbf{X})} + \mathrm{KL}\left(q(\mathbf{X}) \,\|\, p(\mathbf{X})\right)$$

- Which is analytically tractable for Gaussian $q(\mathbf{X})$ and some covariance functions.

Required Expectations

- Need expectations under $q(\mathbf{X})$ of:

$$\log c_i = \frac{1}{2\sigma^2}\left[k_{i,i} - \mathbf{k}_{i,\mathbf{u}}^\top \mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{k}_{i,\mathbf{u}}\right]$$

and

$$\log \mathcal{N}\left(\mathbf{y}|\left\langle \mathbf{f}\right\rangle_{p(\mathbf{f}|\mathbf{u},\mathbf{Y})}, \sigma^2\mathbf{I}\right) = -\frac{1}{2}\log 2\pi\sigma^2 - \frac{1}{2\sigma^2}\left(y_i - \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{u}\right)^2$$

- This requires the expectations

$$\left\langle \mathbf{K}_{\mathbf{f},\mathbf{u}}\right\rangle_{q(\mathbf{X})}$$

and

$$\left\langle \mathbf{K}_{\mathbf{f},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{f}}\right\rangle_{q(\mathbf{X})}$$

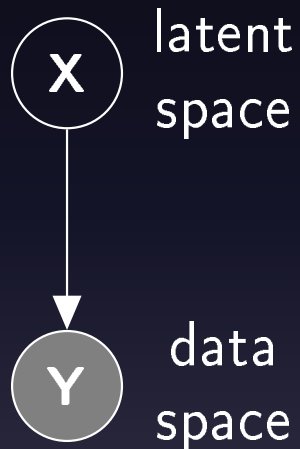
which can be computed analytically for some covariance functions.

Priors for Latent Space

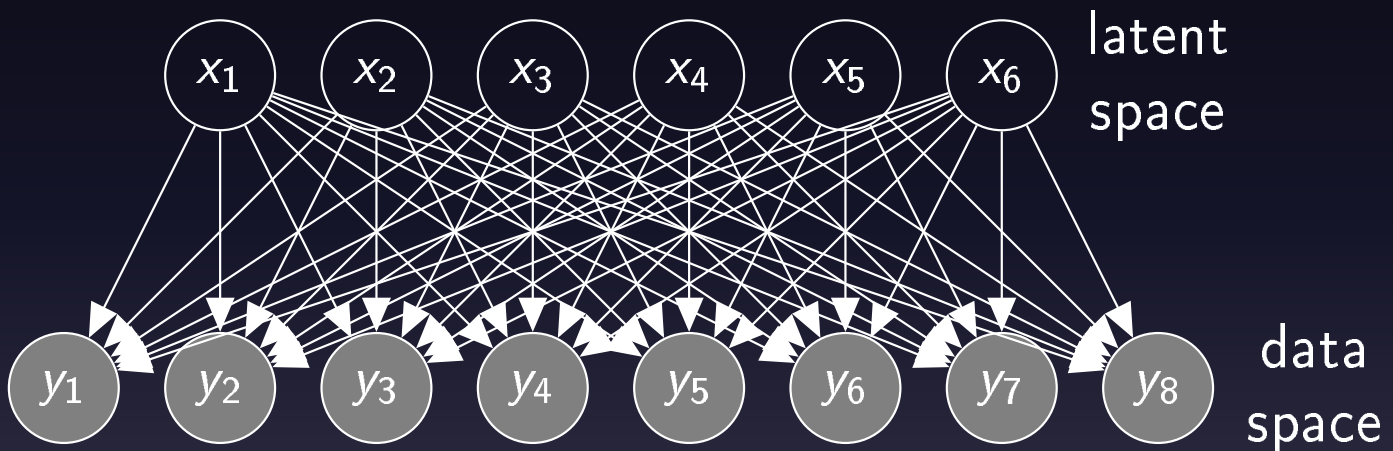
(Titsias and Lawrence, 2010)

- Variational marginalization of $\mathbf{X}$ allows us to learn parameters of $p(\mathbf{X})$.
- Standard GP-LVM where $\mathbf{X}$ learnt by MAP, this is not possible (see e.g. Wang et al., 2008).
- First example: learn the dimensionality of latent space.

Graphical Representations of GP-LVM

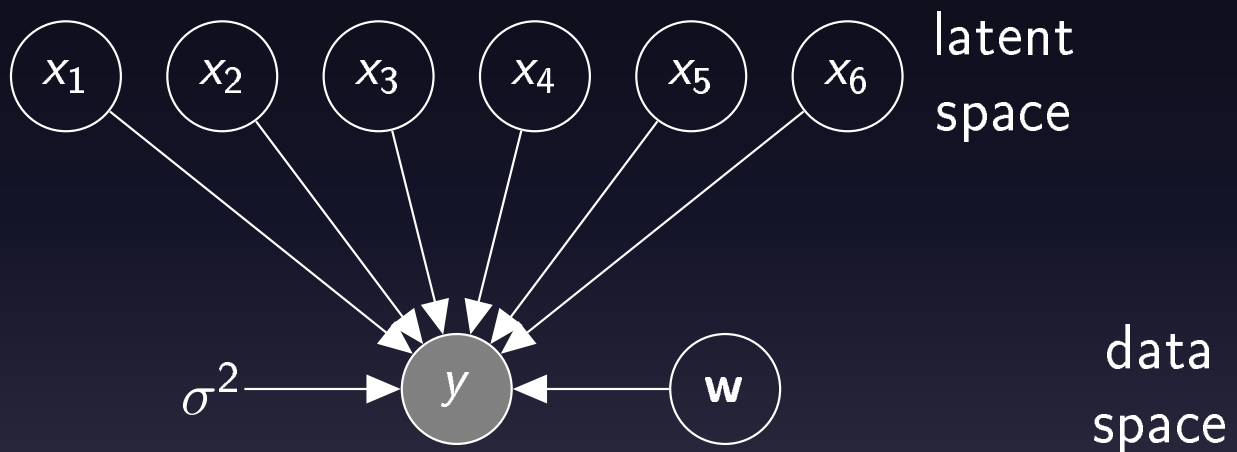


Graphical Representations of GP-LVM

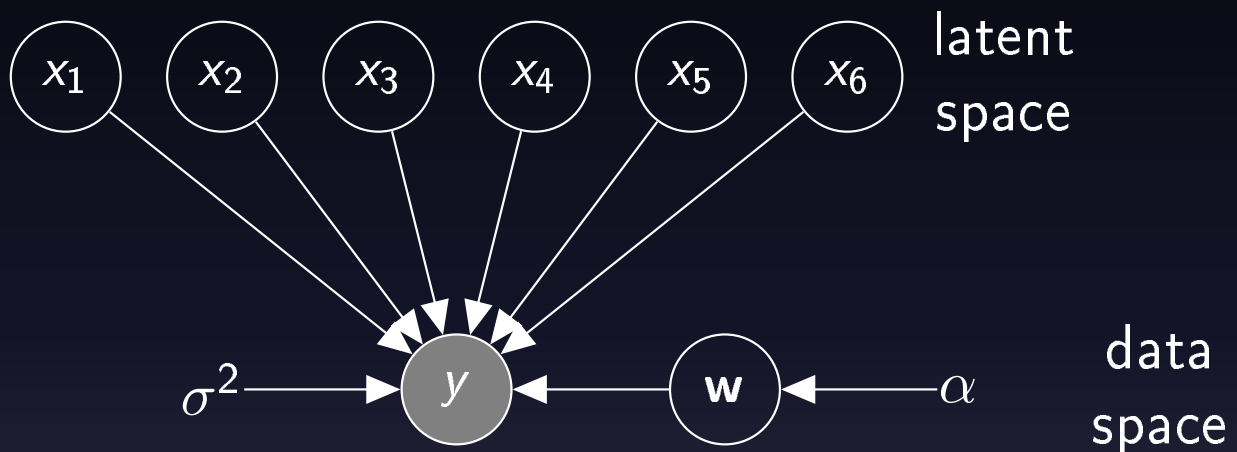


Graphical Representations of GP-LVM

Graphical Representations of GP-LVM



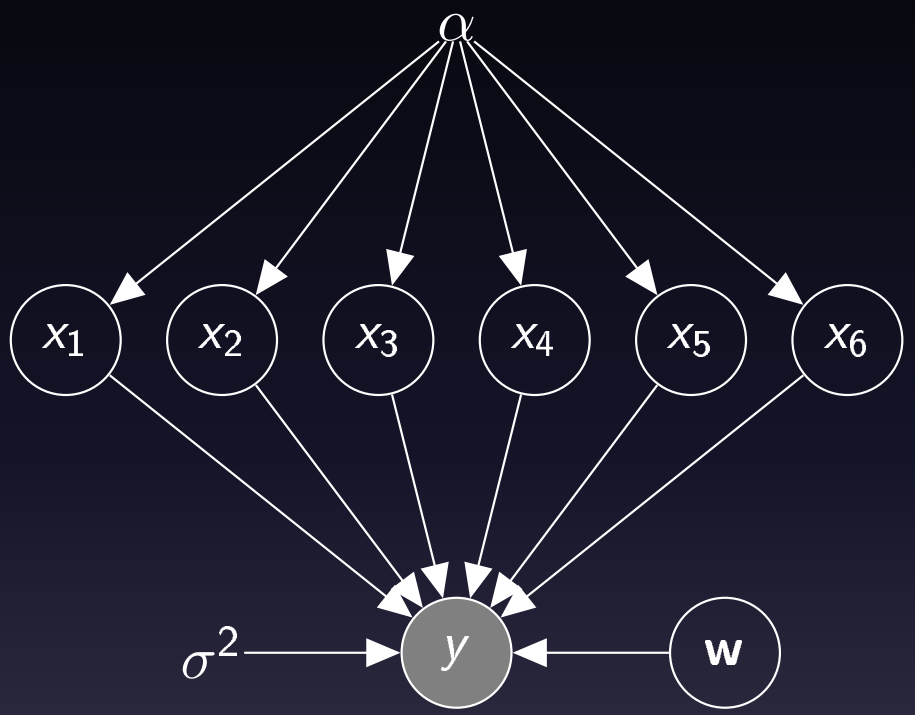
Graphical Representations of GP-LVM



$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha \mathbf{I}) \quad \mathbf{x} \sim \mathcal{N}(0, \mathbf{I})$$

$$y \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w}, \sigma^2)$$

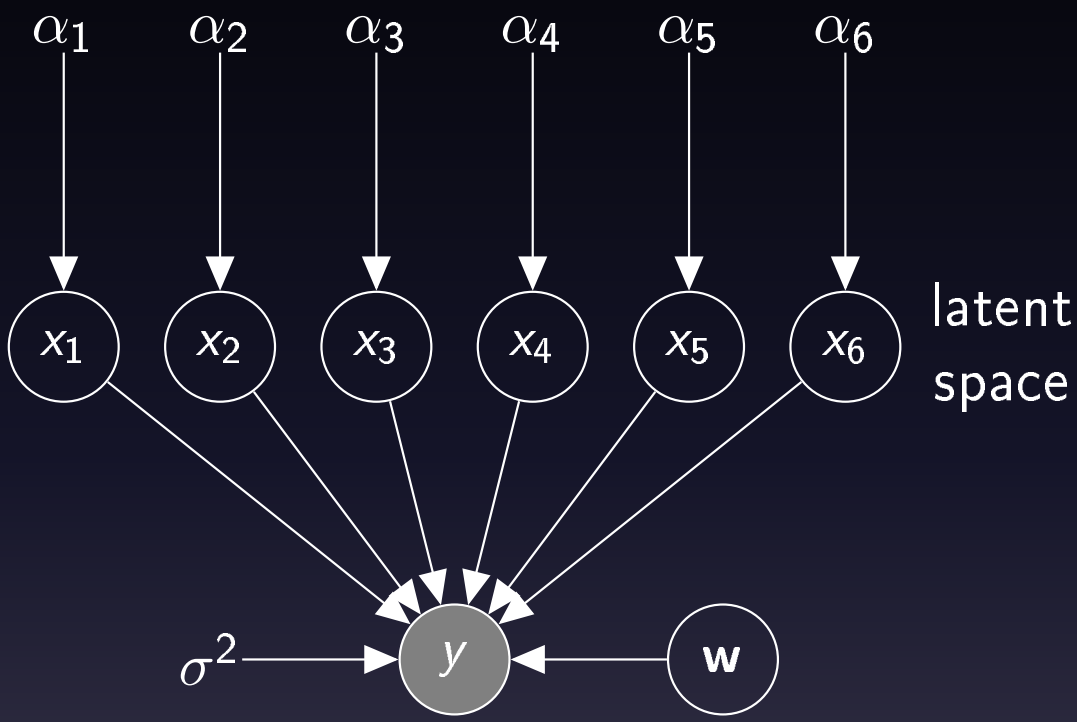
Graphical Representations of GP-LVM



$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \mathbf{x} \sim \mathcal{N}(0, \alpha\mathbf{I})$$

$$y \sim \mathcal{N}\left(\mathbf{x}^\top \mathbf{w}, \sigma^2\right)$$

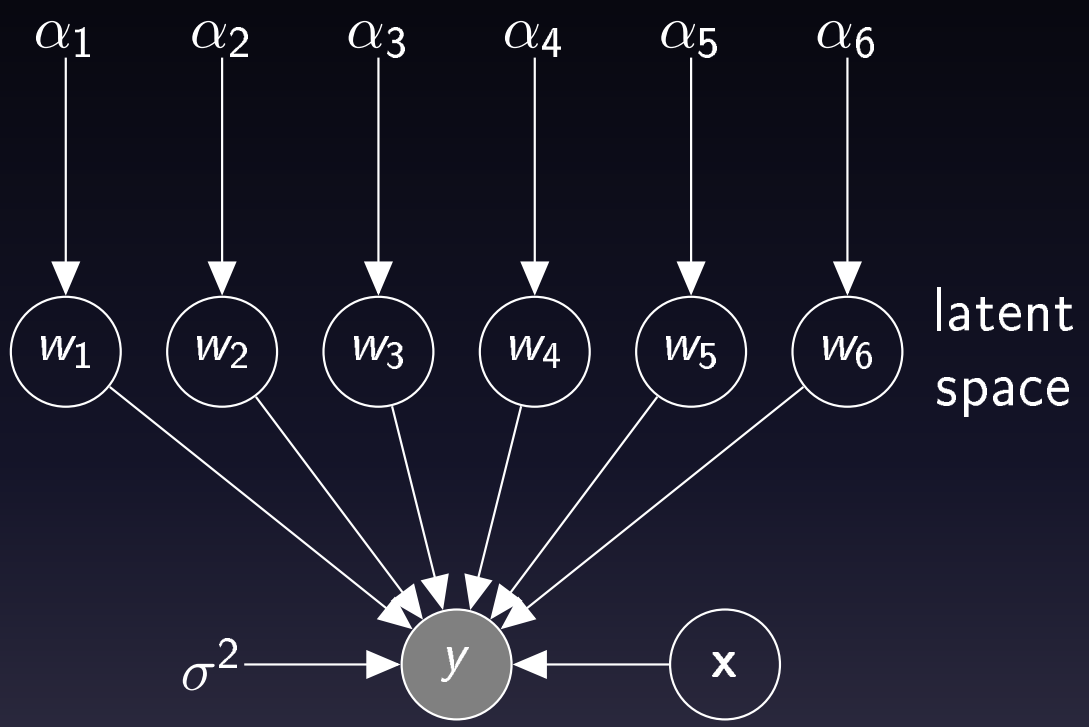
Graphical Representations of GP-LVM



$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad x_i \sim \mathcal{N}(0, \alpha_i)$$

$$y \sim \mathcal{N}\left(\mathbf{x}^\top \mathbf{w}, \sigma^2\right)$$

Graphical Representations of GP-LVM



$$w_i \sim \mathcal{N}(0, \alpha_i) \quad \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

$$y \sim \mathcal{N}\left(\mathbf{x}^\top \mathbf{w}, \sigma^2\right)$$

Learning Dimensionality: Automatic Relevance Determination

$\{\alpha_i\}_{i=1}^{q}$, softly switch off latent dimensions either through

$$p(\mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}(\mathbf{x}_{:,j}|\mathbf{0}, \alpha_j \mathbf{I})$$

or

$$p(\mathbf{W}) = \prod_{i=1}^{q} \mathcal{N}(\mathbf{x}_{i,:}|\mathbf{0}, \alpha_i \mathbf{I})$$

Non-linear $f(\mathbf{x})$

- In linear case equivalence because $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$

$$p(w_i) \sim \mathcal{N}(\mathbf{0}, \alpha_i)$$

- In non linear case, need to scale columns of $\mathbf{X}$ in prior for $f(\mathbf{x})$.
- This implies scaling columns of $\mathbf{X}$ in covariance function

$$k(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}) = \exp\left(-\frac{1}{2}(\mathbf{x}_{:,i} - \mathbf{x}_{:,j})^\top \mathbf{A}(\mathbf{x}_{:,i} - \mathbf{x}_{:,j})\right)$$

$\mathbf{A}$ is diagonal with elements α_i^2. Now keep prior spherical

$$p(\mathbf{X}) = \prod_{j=1}^{q} \mathcal{N}(\mathbf{x}_{:,j} | \mathbf{0}, \mathbf{I})$$

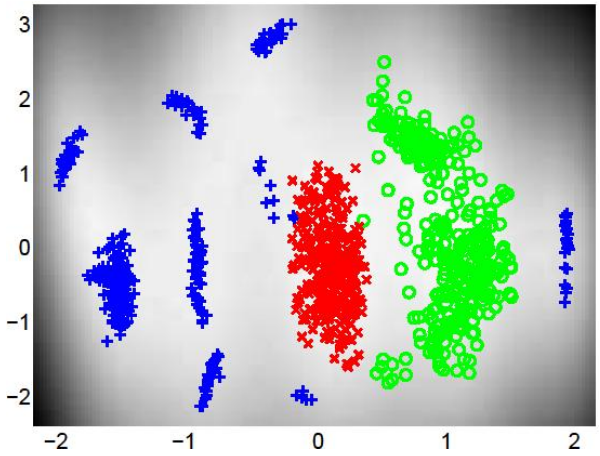
- Covariance functions of this type are known as ARD (see e.g. Neal, 1996; MacKay, 2003; Rasmussen and Williams, 2006).

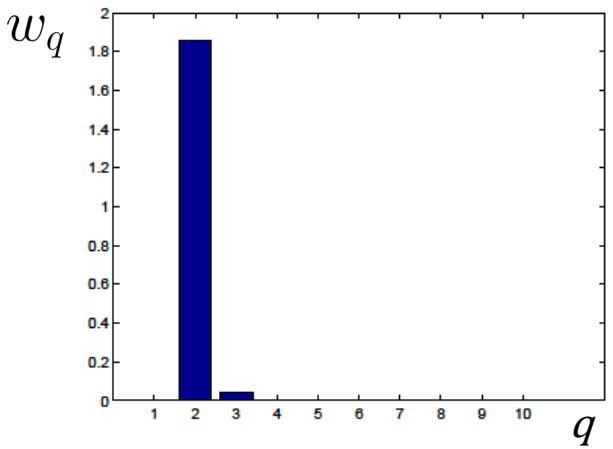
Automatic dimensionality detection

- Achieved by employing an *Automatic Relevance Determination (ARD)* covariance function for the prior on the GP mapping
- $f \sim GP(\mathbf{0}, k_f)$ with

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

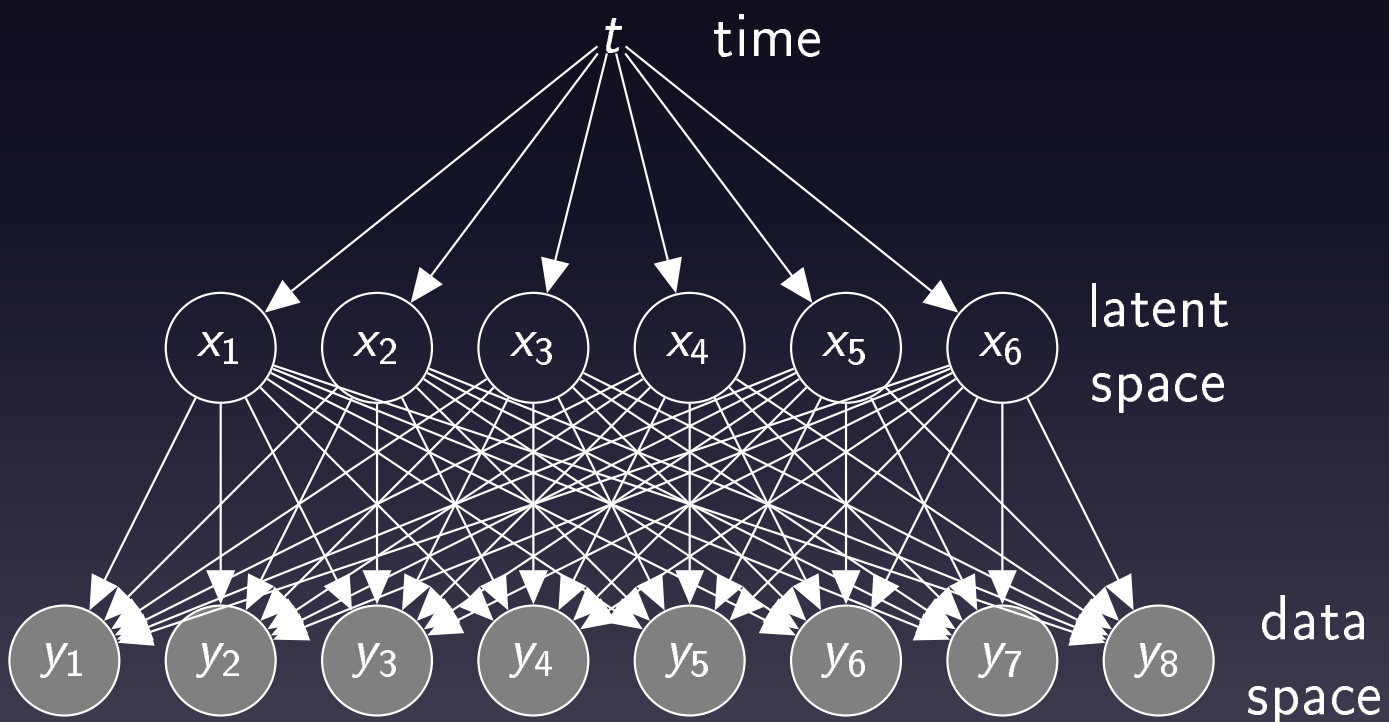
- Example





Gaussian Process Dynamical Systems

Work with Andreas Damianou and Michalis Titsias



Gaussian Process over Latent Space

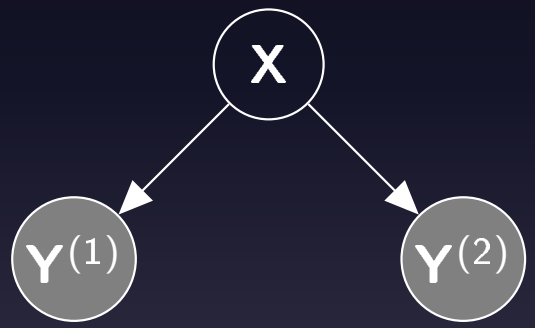
- Assume a GP prior for $p(\mathbf{X})$.
- Input to the process is time, $p(\mathbf{X}|t)$.

Gaussian Process over Latent Space

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

Modeling Multiple 'Views'

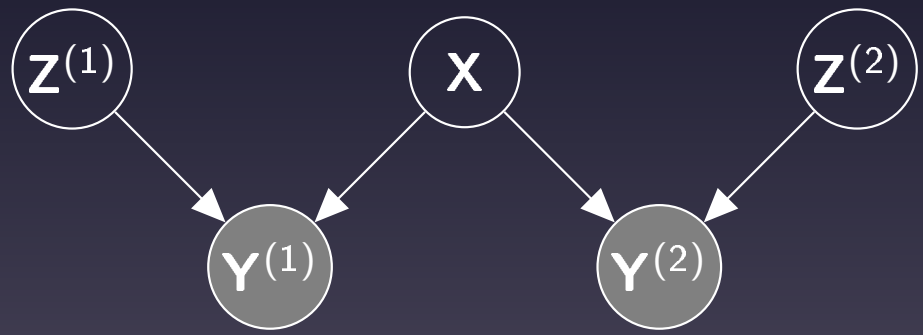
- Single space to model correlations between two different data sources, e.g., images & text, image & pose.
- Shared latent spaces: (Shon et al., 2006; Navaratnam et al., 2007; Ek et al., 2008b)



- 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

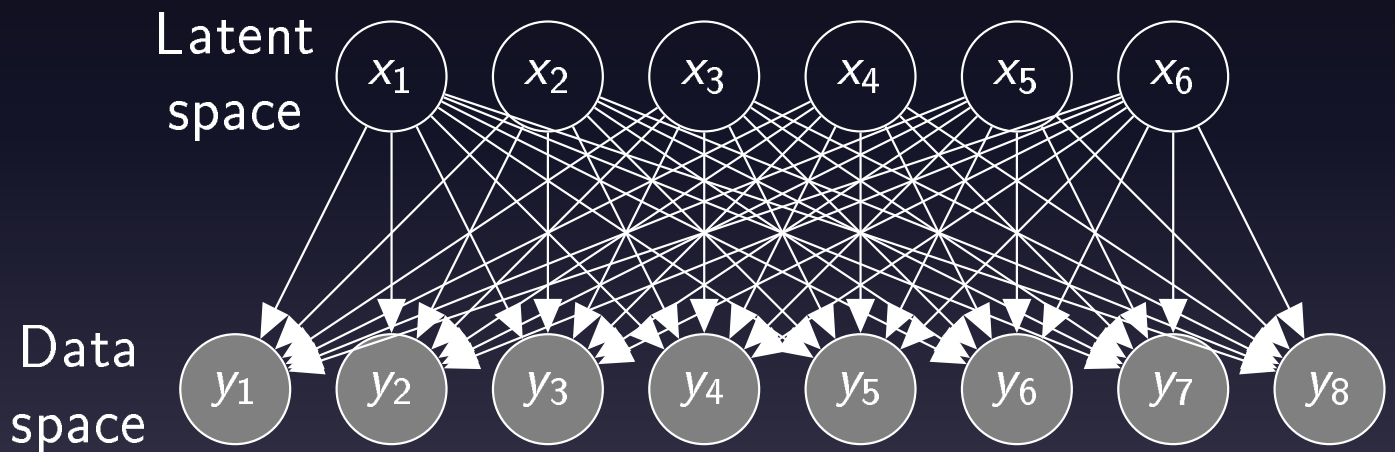
- In real scenarios, the 'views' are neither fully independent, nor fully correlated.
- Shared models
  - either allow information relevant to a single view to be mixed in the shared signal,
  - or are unable to model such private information.
- Solution: Model shared and private information (Virtanen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



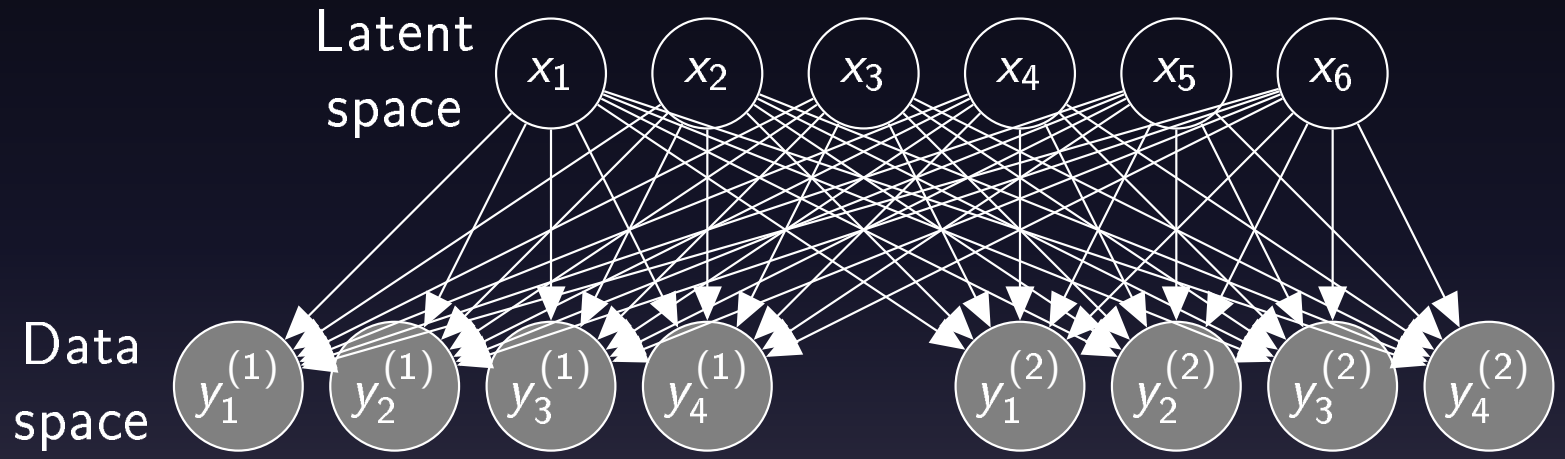
- Probabilistic CCA is case when dimensionality of $\mathbf{Z}$ matches $\mathbf{Y}^{(i)}$ (cf Inter Battery Factor Analysis (Tucker, 1958)).

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

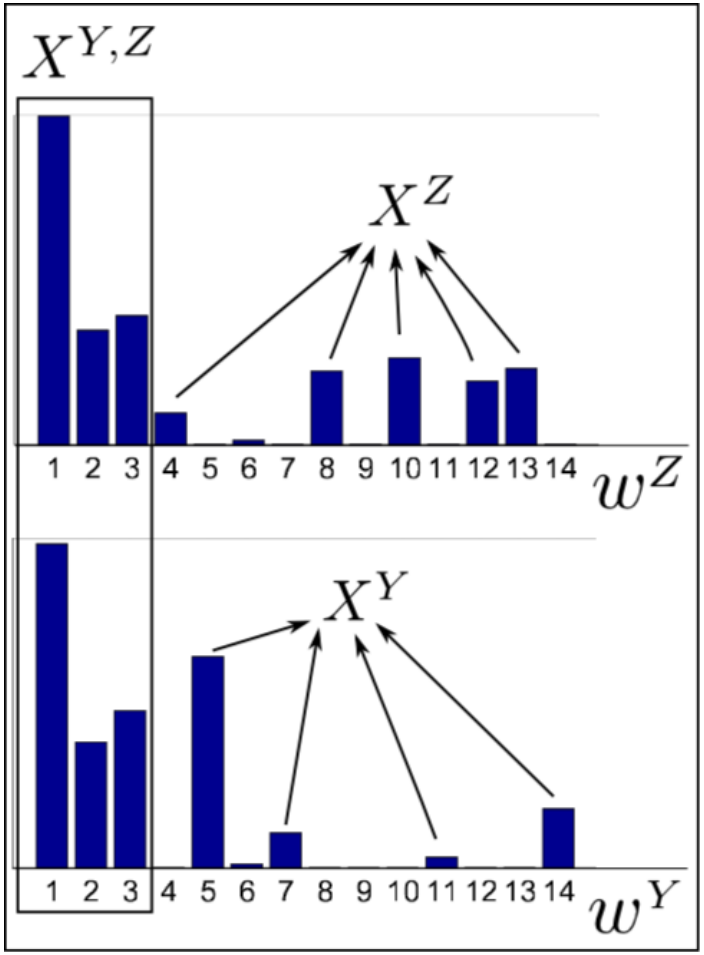
- Dataset Y: 3 persons under all illumination conditions
- Dataset Z: As above for 3 different persons
- Align datapoints $\mathbf{x}_n$ and $\mathbf{z}_n$ only based on the lighting direction

Results

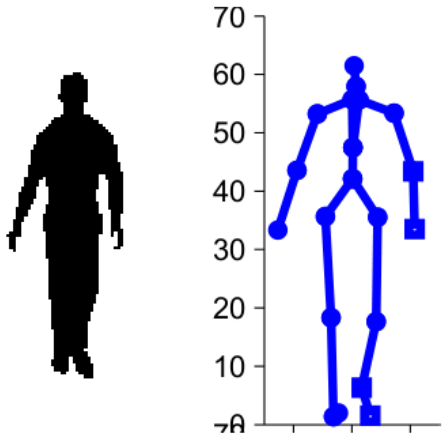
- Latent space X initialised with 14 dimensions

- Weights define a segmentation of X

- Video / demo…



Potential applications..?



Outline

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)

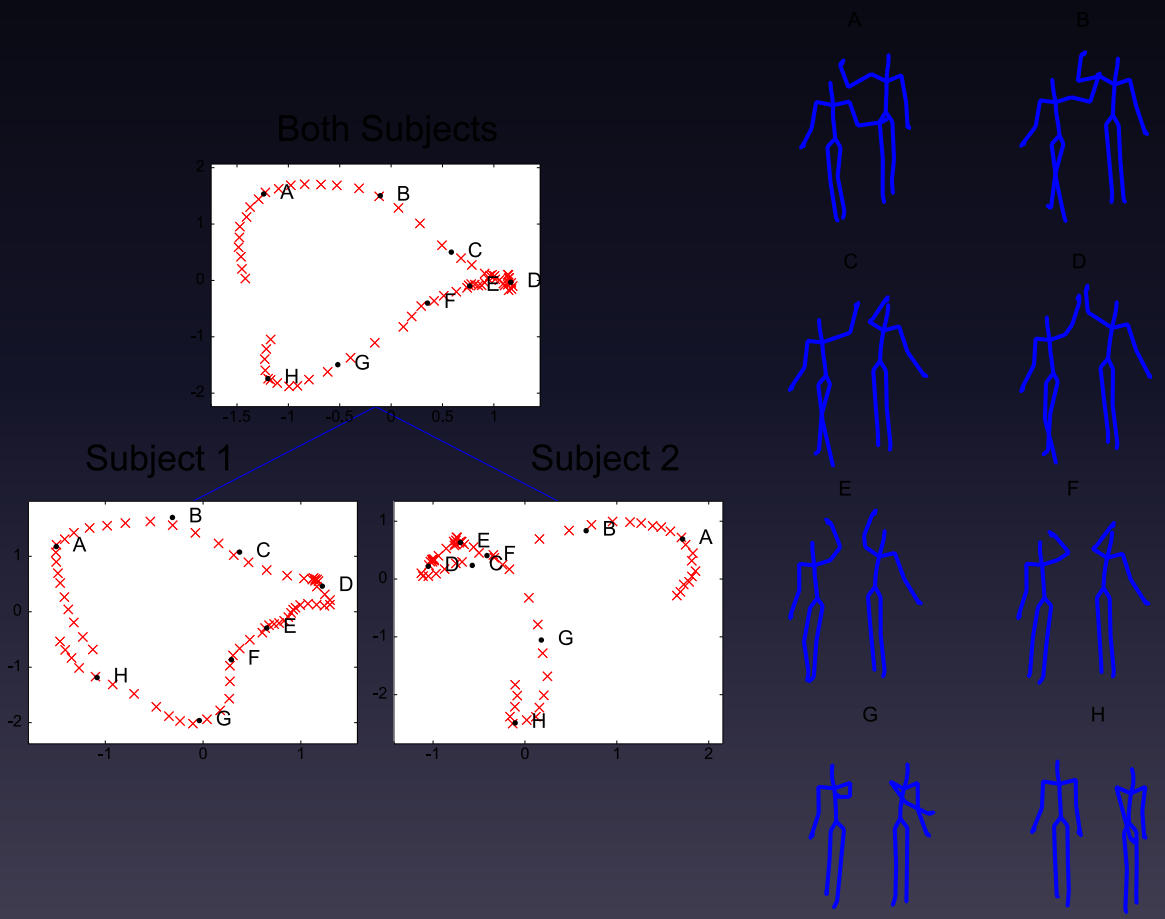


Figure: Hierarchical model of a 'high five'.

Within Subject Hierarchy

(Lawrence and Moore, 2007)

Decomposition of Body

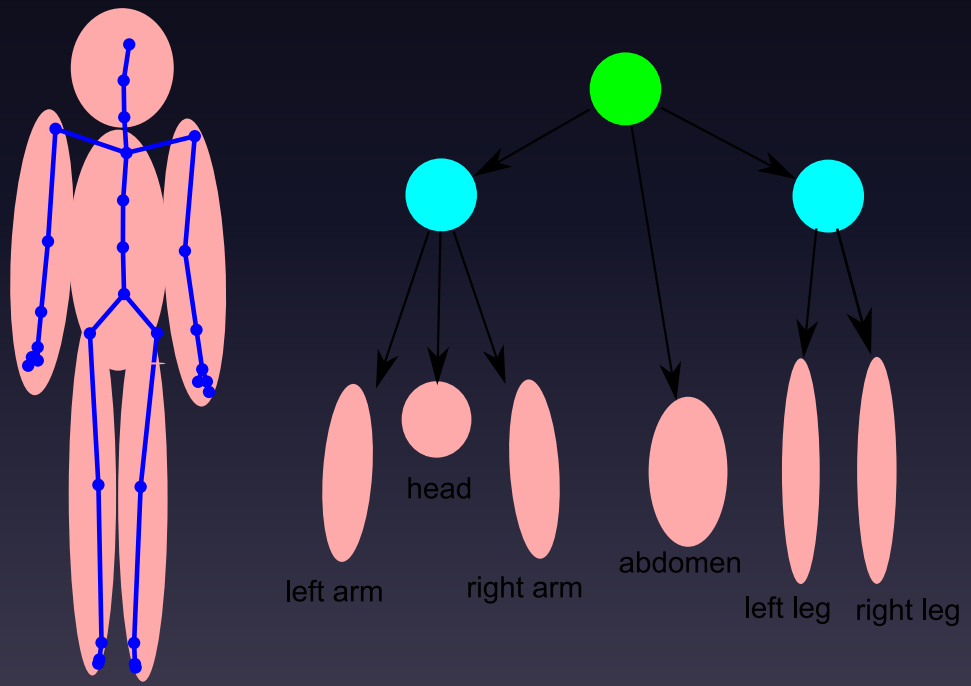


Figure: Decomposition of a subject.

Single Subject Run/Walk

(Lawrence and Moore, 2007)

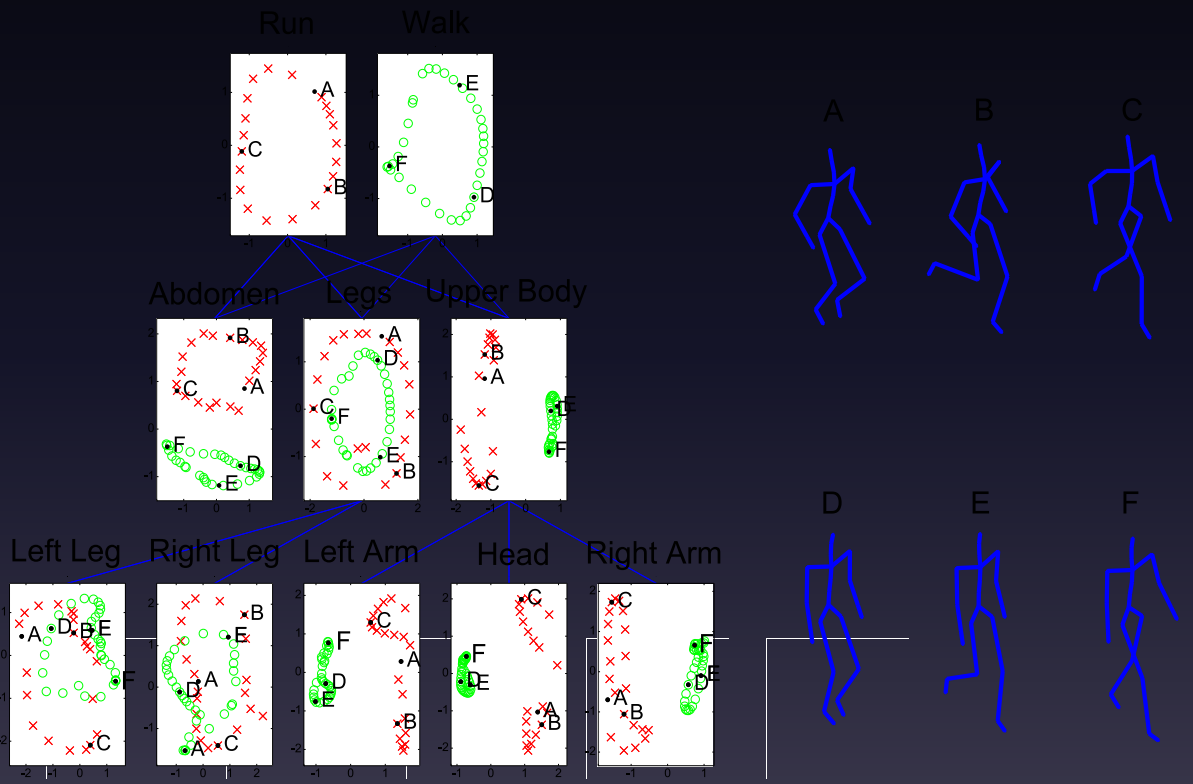


Figure: Hierarchical model of a walk and a run.

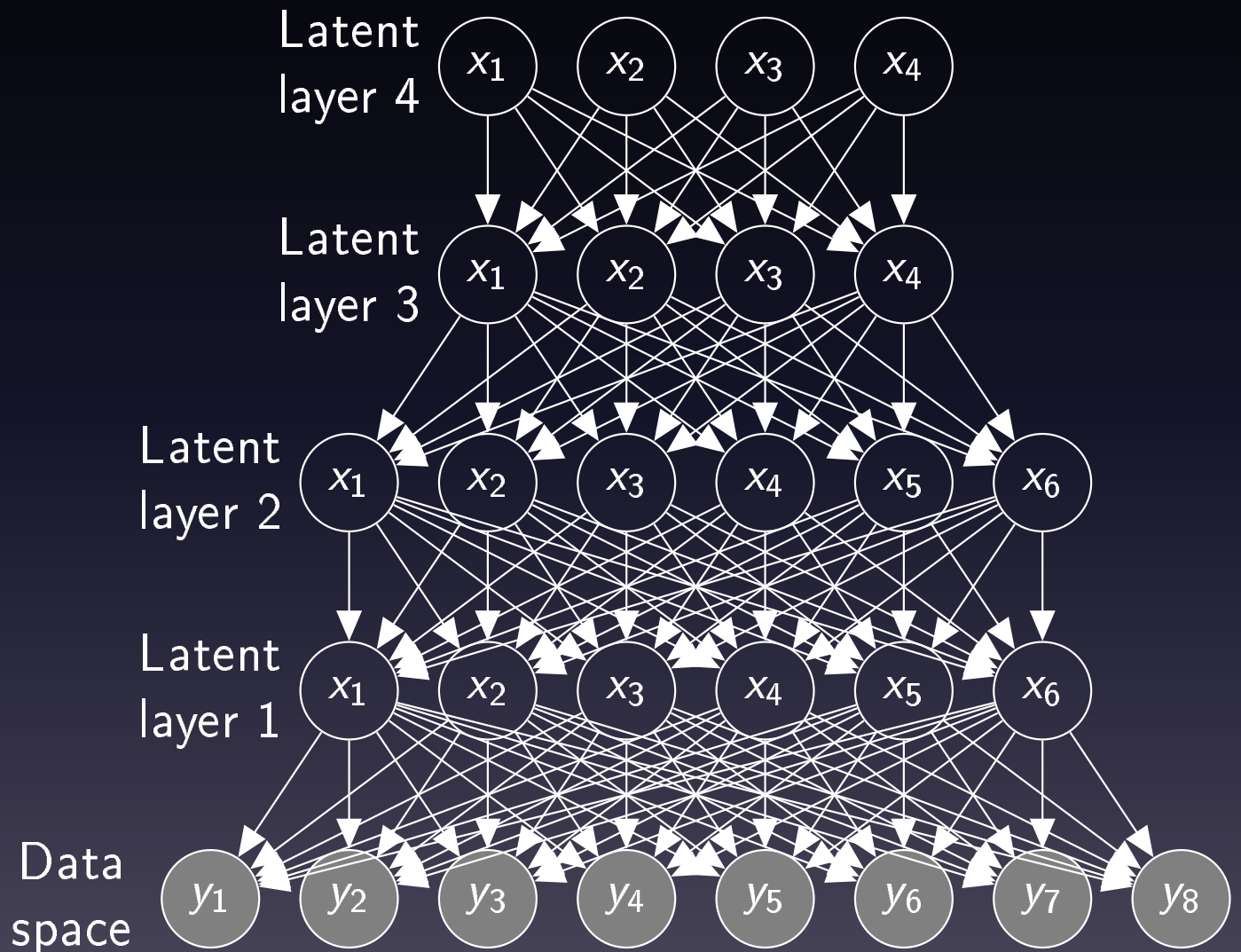
▸ Return

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.

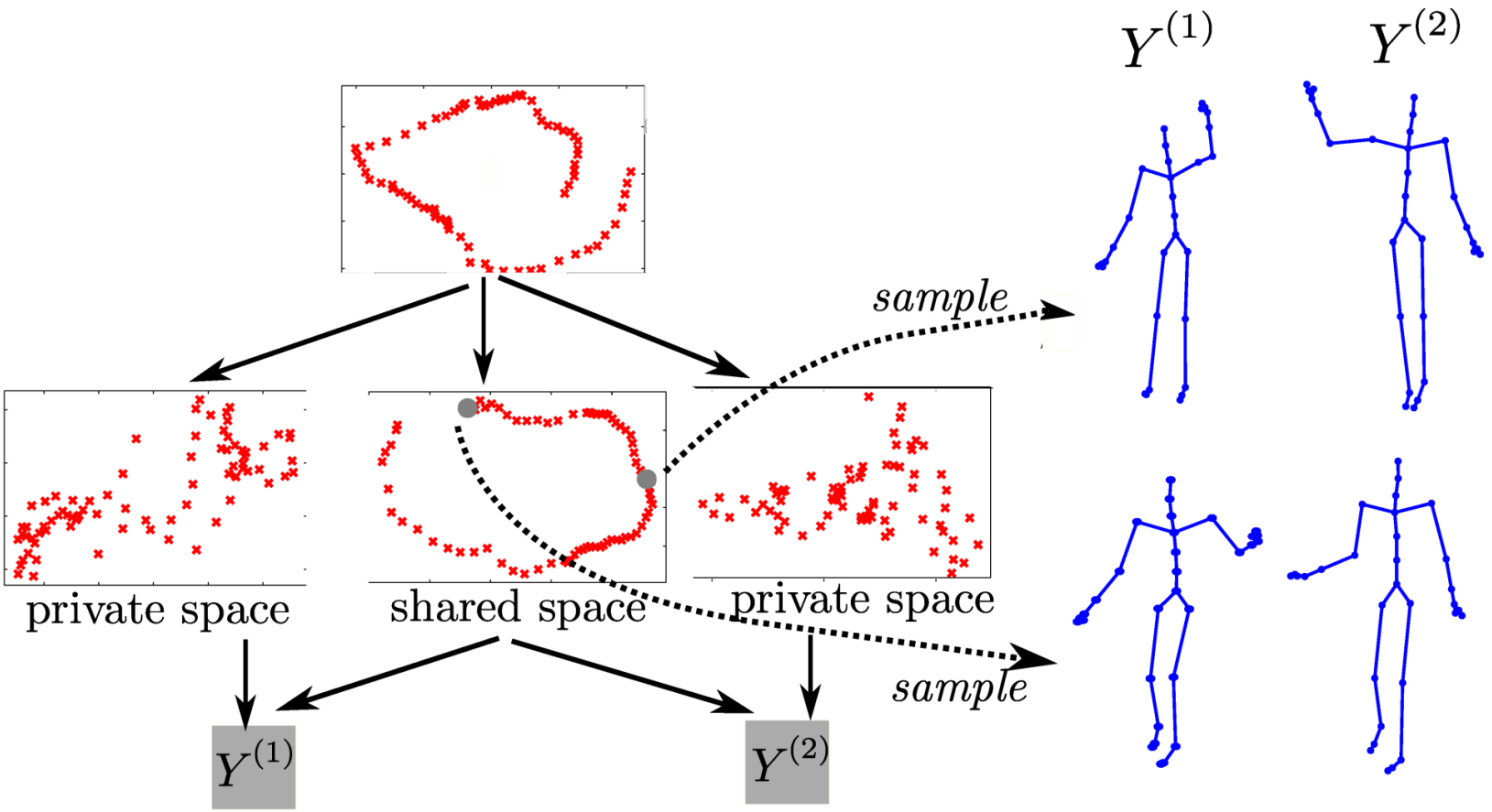
Deep Gaussian Processes



Motion Capture

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

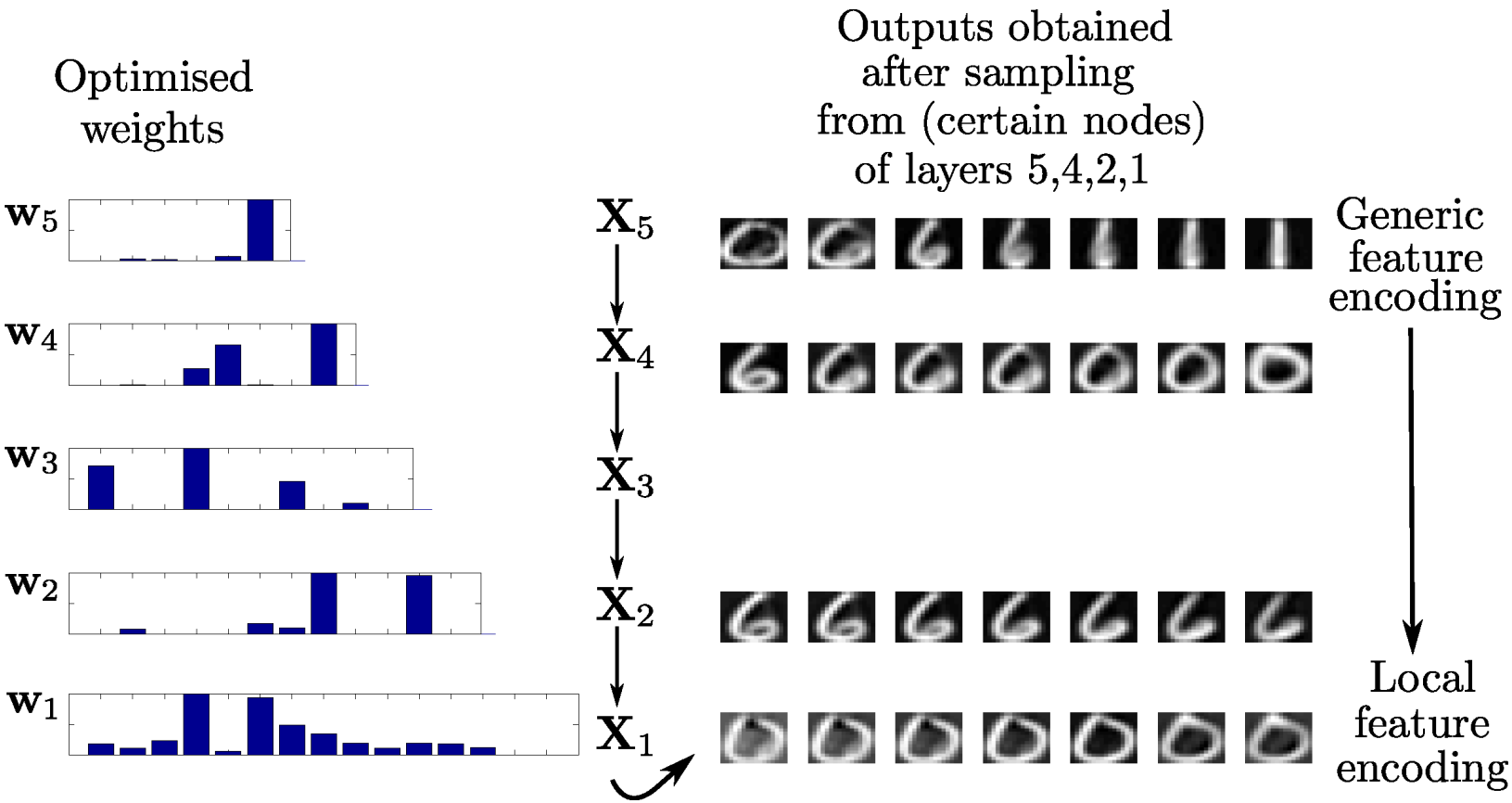
Deep hierarchies – motion capture



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

W. V. Baxter and K.-I. Anjyo. Latent doodle space. In *EUROGRAPHICS*, volume 25, pages 477–485, Vienna, Austria, September 4-8 2006.

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

C. M. Bishop, M. Svensén, and C. K. I. Williams. GTM: the Generative Topographic Mapping. *Neural Computation*, 10(1):215–234, 1998. [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] .

K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popovic. Style-based inverse kinematics. In *ACM Transactions on Graphics (SIGGRAPH 2004)*, pages 522–531, 2004.

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] .

References II

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

J. B. Kruskal. Multidimensional scaling by optimizing goodness-of-fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–28, 1964. [DOI].

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.

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. Lowe and M. E. Tipping. Neuroscale: Novel topographic feature extraction with radial basis function networks. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 543–549, Cambridge, MA, 1997. MIT Press.

D. J. C. MacKay. Bayesian neural networks and density networks. *Nuclear Instruments and Methods in Physics Research, A*, 354(1):73–80, 1995. [DOI].

D. J. C. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, Cambridge, U.K., 2003. [Google Books] .

References III

K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate analysis*. Academic Press, London, 1979. [Google Books] .

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.

R. M. Neal. *Bayesian Learning for Neural Networks*. Springer, 1996. Lecture Notes in Statistics 118.

V. Priacuriu and I. D. Reid. Nonlinear shape manifolds as shape priors in level set segmentation and trackign. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011a.

V. Priacuriu and I. D. Reid. Shared shape spaces. In *IEEE International Conference on Computer Vision (ICCV)*, 2011b.

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] .

S. T. Roweis. EM algorithms for PCA and SPCA. In M. I. Jordan, M. J. Kearns, and S. A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10, pages 626–632, Cambridge, MA, 1998. MIT Press.

S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000. [DOI].

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.

J. W. Sammon. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, C-18(5):401–409, 1969. [DOI].

B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998. [DOI].

References IV

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.

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).

J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. [DOI].

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.

R. Urtasun, D. J. Fleet, and P. Fua. 3D people tracking with Gaussian process dynamical models. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 238–245, New York, U.S.A., 17–22 Jun. 2006. IEEE Computer Society Press.

R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua. Priors for people tracking from small training sets. In *IEEE International Conference on Computer Vision (ICCV)*, pages 403–410, Bejing, China, 17–21 Oct. 2005. IEEE Computer Society Press.

References V

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.

J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298, 2008. ISSN 0162-8828. [DOI].

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.