The Gaussian Process Latent Variable Model

Neil D. Lawrence

27th January 2006

Oxford

Introduction

- The Gaussian process latent variable model (GP-LVM)
  - ➥ powerful approach to probabilistic non-linear dimensionality reduction.
- This review:
  - ➥ Review Probabilistic PCA [18].
  - ➥ Review Gaussian Processes.
  - ➥ Derive GP-LVM.
  - ➥ Present some Results.

Examples that can be recreated: code from `http://www.dcs.shef.ac.uk/~neil/fgplvm` & `http://www.dcs.shef.ac.uk/~neil/oxford`.

Motivation

- Many data sets are high dimensional.
- 'Curse of dimensionality' implies that we need many data points.
- In practice we often do very well with smaller data sets.
- Perhaps many data sets of interest seem high dimensional but are intrinsically low dimensional.

Digits Data

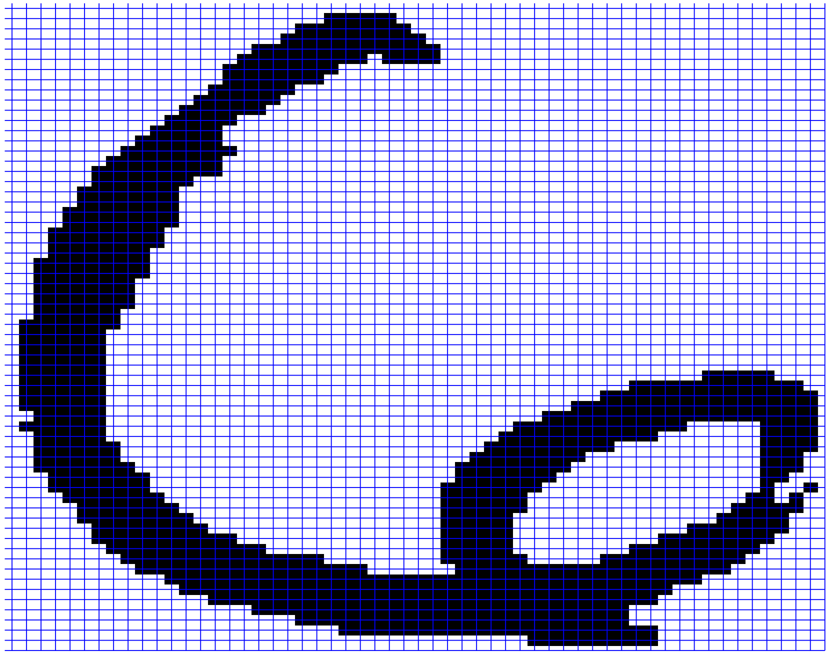


Figure 1: Digit 6 from the USPS Cedar CD-ROM. The digit is 64 pixels by 57 pixels giving it 3,648 dimensions.

Lower Intrinsic Dimensionality

- This data point is 3,648 dimensional.
- Data won't span all 3,648 dimensions of the space.
- Consider digit rotations.
- Let's consider rotations of the digit:
  - Create a data set by rotating the original digit 360 times.
  - Project data onto its 2nd & 3rd principal components.

Rotated Digit

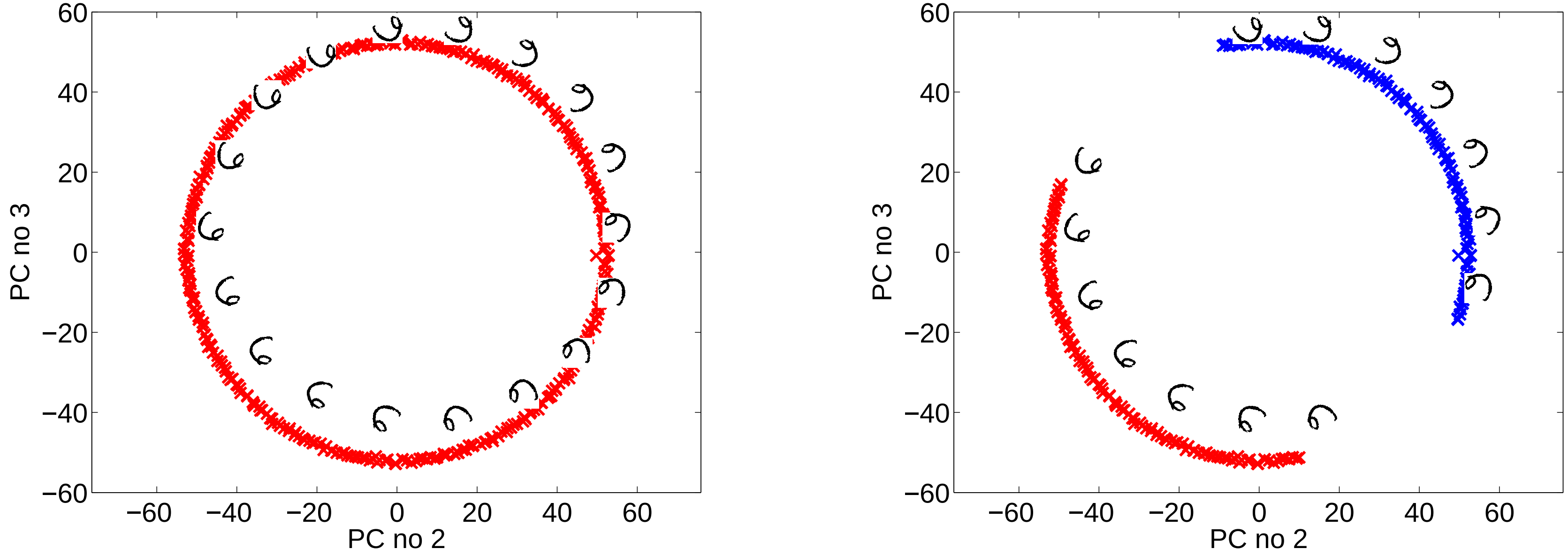


Figure 2: Rotation of handwritten 6. Data set generated by rotating the original image 360 times .

More Transformations

- Real data sets not generated by a simple rotation of a one dimensional space.
- Reasonable to assume a data set:
  - has fixed number of ‘prototypes’
  - each undergoes a limited number of transformations
  - and is, perhaps, corrupted by some noise.
- Makes sense to model high dimensional data by seeking a low dimensional ‘embedding’.

Traditional Approaches

- Standard approach in statistics: multi-dimensional scaling (MDS, see *e.g.* [8]).
- Recently in Machine Learning several spectral approaches. ([17, 14, 21])
- Some can be seen as classical MDS.
- We seek a probabilistic approach:
  - ➥ Ease of extension, for the GP-LVM see ([2, 19, 20, 15]).

Coming Up

- Review of Probabilistic PCA.
- Review of Gaussian Processes.
- Dual Probabilistic PCA and the GP-LVM.
- Results from the GP-LVM + back constraints, dynamics etc.

Probabilistic PCA

- Given points in a latent space $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^{\mathrm{T}}$ and a *centred data set*, $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^{\mathrm{T}}$ assume

$$\mathbf{y}_n = \mathbf{W}\mathbf{x}_n + \boldsymbol{\eta}_n,$$

$\mathbf{W} \in \Re^{D \times q}$, D – data space dim, q – latent space dim., $\boldsymbol{\eta}_n$ noise term.

- For probabilistic PCA,

$$p(\boldsymbol{\eta}_n|\beta) = N\left(\boldsymbol{\eta}_n|\mathbf{0}, \beta^{-1}\mathbf{I}\right),$$

β is an inverse variance or precision.

Probabilistic PCA contd

- Conditional probability of the data given the latent space written as

$$p\left(\mathbf{y}_n|\mathbf{x}_n, \mathbf{W}, \beta\right) = N\left(\mathbf{y}_n|\mathbf{W}\mathbf{x}_n, \beta^{-1}\mathbf{I}\right),$$

assume independence across data points

$$p\left(\mathbf{Y}|\mathbf{X}, \mathbf{W}, \beta\right) = \prod_{n=1}^{N} N\left(\mathbf{y}_n|\mathbf{W}\mathbf{x}_n, \beta^{-1}\mathbf{I}\right). \tag{1}$$

This term can be seen as a *likelihood*.

Gaussian Prior

- The Gaussian prior over $\mathbf{X}$ is zero mean and unit covariance,

$$p(\mathbf{X}) = \prod_{n=1}^{N} p(\mathbf{x}_n) = \prod_{n=1}^{N} N(\mathbf{x}_n|\mathbf{0}, \mathbf{I}) \,. \tag{2}$$

- The marginal likelihood is then,

$$p(\mathbf{Y}|\mathbf{W}, \beta) = \prod_{n=1}^{N} N(\mathbf{y}_n|\mathbf{0}, \mathbf{C}) \,, \tag{3}$$

where $\mathbf{C} = \mathbf{W}\mathbf{W}^{\mathrm{T}} + \beta^{-1}\mathbf{I}$.

Reduced Rank Covariance

- $\mathbf{C}$ is recognised as a reduced rank representation of the covariance.

$$\mathbf{C} = \mathbf{W}\mathbf{W}^{\mathrm{T}} + \beta^{-1}\mathbf{I}$$

- Since $\mathbf{W} \in \Re^{D\times q}$ the matrix $\mathbf{W}\mathbf{W}^{\mathrm{T}} \in \Re^{D\times D}$ will have rank of at most q.

- The term $\beta^{-1}\mathbf{I}$ then acts as a 'regulariser'.

Maximum Likelihood Solution

- Model was suggested simultaneously by [13, 18].
- But [18] also proved that maximum likelihood solution for $\mathbf{W}$ spans the principal sub-space.
- Solution is
$$\hat{\mathbf{W}} = \mathbf{U}_q'\mathbf{L}\mathbf{V}^{\mathrm{T}}$$
$\mathbf{U}_q'$ are q eigenvectors of $N^{-1}\mathbf{Y}^{\mathrm{T}}\mathbf{Y}$ associated with q largest eigenvalues, $\{\lambda_i\}_{i=1}^{q}$.
- $\mathbf{L}$ is diagonal, its ith element is $l_i = \left(\lambda_i - \beta^{-1}\right)^{\frac{1}{2}}$.

Eigenvalue Problem

- *I.e.* we solve this eigenvalue problem

$$N^{-1}\mathbf{Y}^{\mathrm{T}}\mathbf{Y}\mathbf{U}' = \mathbf{U}'\Lambda. \tag{4}$$

- Solution for probabilistic PCA spans the q-dimensional principal sub-space of the data.

Gaussian Processes

- Gaussian processes: [9, 10, 25, 24, 7, 12], probability distributions over functions.

- Functions are infinite dimensional objects.

- Consider a finite Gaussian distribution over instantiated values $\mathbf{f} = \{f_n\}_{n=1}^{N} \in \Re^{N\times 1}$.

- Assume that these values are drawn from a Gaussian distribution.

Gaussian Processes

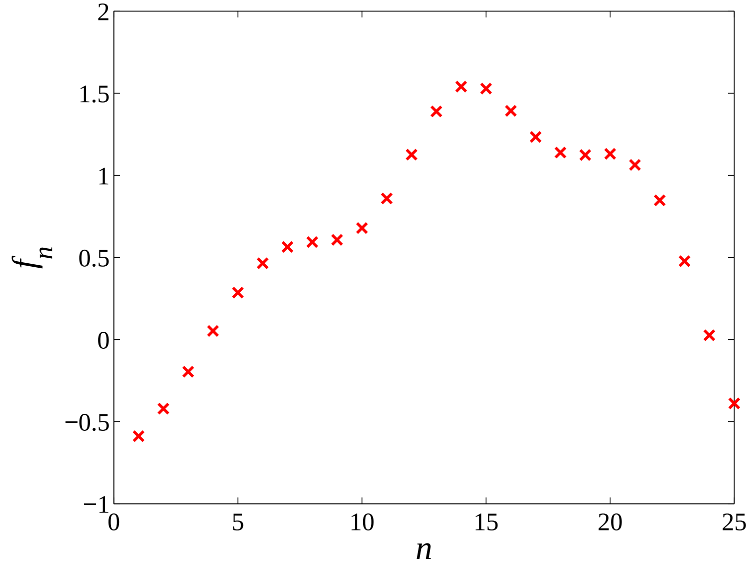
- Typically assume mean zero and covariance $\mathbf{K}$,

$$
\begin{aligned}
p(\mathbf{f}|\mathbf{K}) &= N(\mathbf{f}|\mathbf{0}, \mathbf{K}) \\
&= \frac{1}{(2\pi)^{\frac{N}{2}} |\mathbf{K}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\mathbf{f}\mathbf{K}^{-1}\mathbf{f}\right).
\end{aligned}
$$

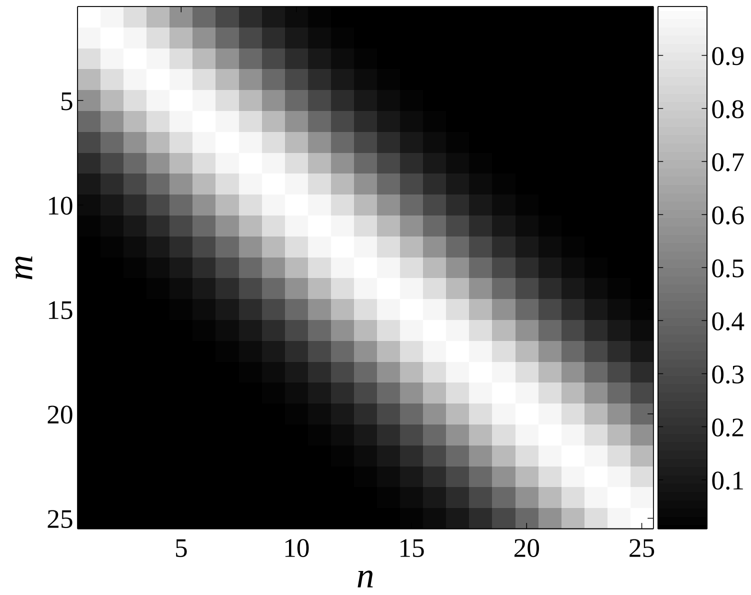
- In the next slide, take *one sample* from a Gaussian with this covariance matrix.

- In this sample there will be $N = 25$ instantiations.

GP Sample



(a)



(b)

Figure 3: (a) 25 instantiations of a function, f_n, (b) covariance matrix as a greyscale plot.

GP Sample

- Covariance function shows correlation between points f_m and f_n if n is near to m.
- Less correlation if n is distant from m.
- The function therefore appears smooth.
- Let's maker predictions given the covariance and 1 data point.

Point Prediction 1 - 2

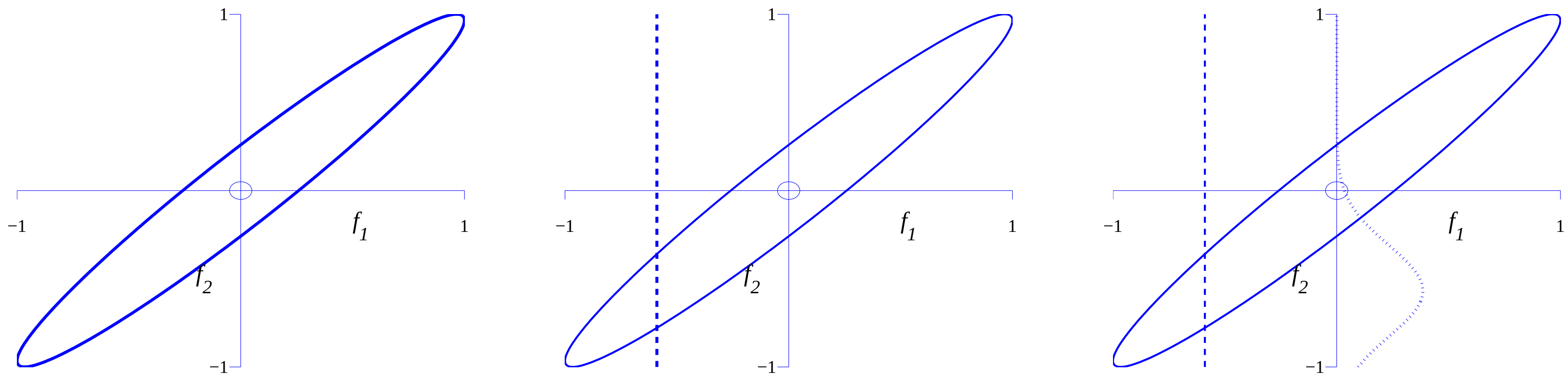


Figure 4: Joint distribution between the values of f_1 and f_2 , $\mathbf{K}_{12} = \begin{bmatrix} 1 & 0.966 \\ 0.966 & 1 \end{bmatrix}$.

Point Prediction 1 - 5

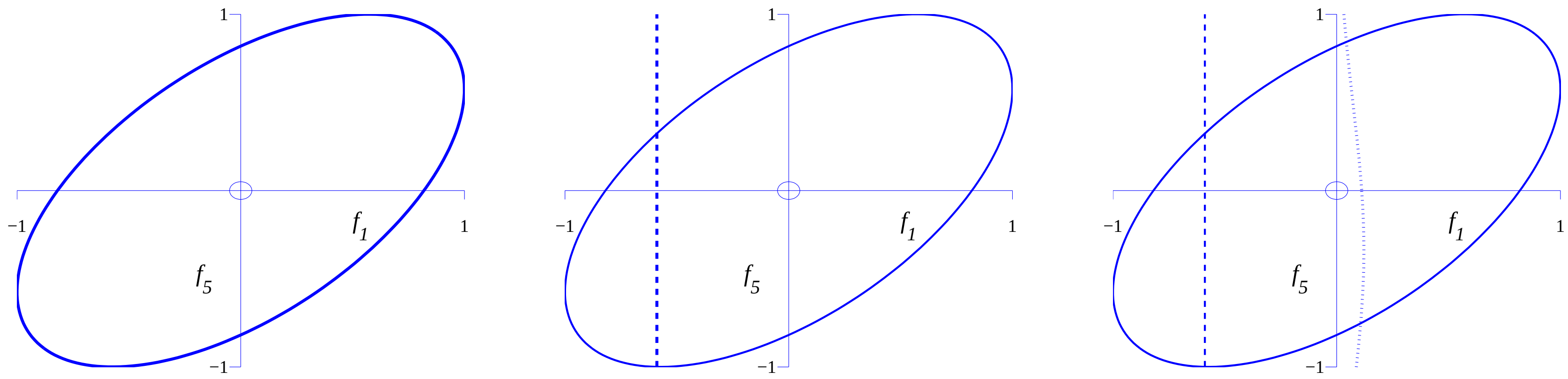


Figure 5: Joint distribution between the values of f_1 and f_5, $\mathbf{K}_{15} = \begin{bmatrix} 1 & 0.574 \\ 0.574 & 1 \end{bmatrix}$.

Whence Covariance?

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

- Based on Euclidean distance

$$k(\mathbf{x}_m, \mathbf{x}_n) = \exp\left(-\frac{\gamma}{2}(\mathbf{x}_m - \mathbf{x}_n)^{\mathrm{T}}(\mathbf{x}_m - \mathbf{x}_n)\right), \tag{5}$$

- Also known as a kernel.

Joint Distribution

- Covariance function provides the joint distribution over the instantiations.
- Conditional distribution provides predictions.
- Denote the training set as $\mathbf{f}$ and test set as $\mathbf{f}_*$, predict using $p\left(\mathbf{f}_*|\mathbf{f}\right)$.
- Find conditional from joint using partiitoned inverse

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{\mathbf{f},\mathbf{f}} & \mathbf{K}_{\mathbf{f},*} \\ \mathbf{K}_{*,\mathbf{f}} & \mathbf{K}_{*,*} \end{bmatrix}$$

Partitioned Inverse

- Partitioned inverse is then

$$\mathbf{K}^{-1} = \left[\begin{array}{cc} \mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1} + \mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}\Sigma^{-1}\mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1} & -\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}\Sigma^{-1} \\ -\Sigma^{-1}\mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1} & \boldsymbol{\Sigma}^{-1} \end{array} \right]$$

where

$$\Sigma = \mathbf{K}_{*,*} - \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}.$$

Joint Distribution

- Logarithm of the joint distribution:

$$\begin{aligned}\log p\left(\mathbf{f}, \mathbf{f}_*\right) &= -\frac{1}{2}\mathbf{f}^{\mathrm{T}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{f} - \frac{1}{2}\mathbf{f}^{\mathrm{T}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}\Sigma^{-1}\mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{f} \\ &\quad +\mathbf{f}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}\Sigma^{-1}\mathbf{f}_* - \frac{1}{2}\mathbf{f}_*^{\mathrm{T}}\Sigma^{-1}\mathbf{f}_* + \mathrm{const}_1\end{aligned}$$

- Conditional is found by dividing joint by the prior, $p\left(\mathbf{f}\right) = N\left(\mathbf{f}|\mathbf{0}, \mathbf{K}_{\mathbf{f},\mathbf{f}}\right)$.

Conditional Distribution

- In log space this is equivalent to subtraction of

$$\log p(\mathbf{f}) = -\frac{1}{2}\mathbf{f}^{\mathrm{T}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{f} + \mathrm{const}_2$$

giving

$$\log p(\mathbf{f}_*|\mathbf{f}) = \log p(\mathbf{f}_*, \mathbf{f}) - \log p(\mathbf{f}) = \log N\left(\mathbf{f}_*|\bar{\mathbf{f}}_*, \Sigma\right).$$

where $\bar{\mathbf{f}} = \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{f}$ and $\Sigma = \mathbf{K}_{*,*} - \mathbf{K}_{*,\mathbf{f}}\mathbf{K}_{\mathbf{f},\mathbf{f}}^{-1}\mathbf{K}_{\mathbf{f},*}$.

Prediction

- If we observe points from the function, $\mathbf{f}$.
- We can predict the locations of functions at as yet unseen locations.
- The prediction is also a Gaussian process, with mean $\bar{\mathbf{f}}$ and covariance Σ.
- Often observe corrupted version of function.

GP Graphical Model

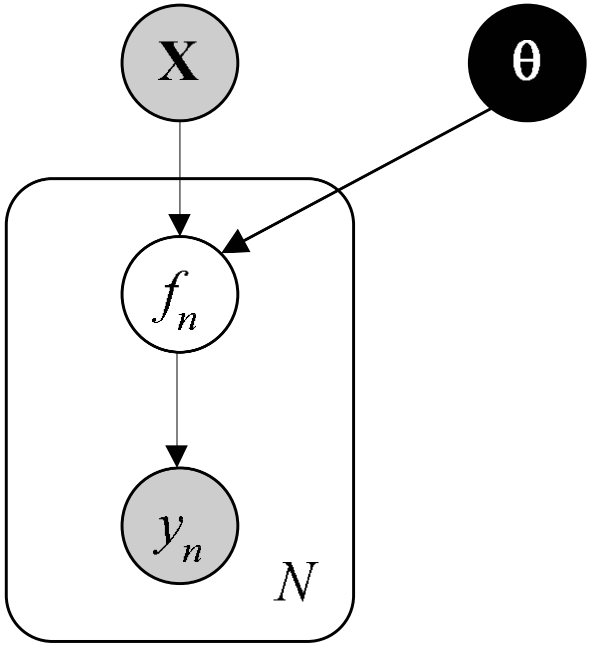


Figure 6: The Gaussian process depicted graphically, here θ represents model parameters.

Noise Model

- Observations are corrupted by noise.
- Define a noise model $p(\mathbf{y}|\mathbf{f})$
- In regression

$$p(\mathbf{y}|\mathbf{f}) = \prod_{n=1}^{N} p(y_n|f_n) = \prod_{n=1}^{N} N\left(y_n|f_n, \beta^{-1}\right), \tag{6}$$

Marginal Likelihood

- Maximise marginal likelihood,

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

$$p(\mathbf{y}) \quad = \quad N\left(\mathbf{y}|\mathbf{0}, \mathbf{K}_{\mathbf{f},\mathbf{f}} + \beta^{-1}\mathbf{I}\right), \tag{7}$$

Result is a Gaussian process on $\mathbf{y}$ with covariance $\mathbf{K}_{\mathbf{f},\mathbf{f}} + \beta^{-1}\mathbf{I}$.

Covariance Functions

- RBF Covariance has two parameters:

$$k(\mathbf{x}_m, \mathbf{x}_n) = \alpha \exp\left(-\frac{\gamma}{2}(\mathbf{x}_m - \mathbf{x}_n)^{\mathrm{T}}(\mathbf{x}_m - \mathbf{x}_n)\right), \tag{8}$$

control signal and and length scale

- Linear Covariance Function

$$k(\mathbf{x}_m, \mathbf{x}_n) = \alpha \mathbf{x}_m^{\mathrm{T}} \mathbf{x}_n$$

$$\mathbf{K}_{\mathbf{f},\mathbf{f}} = \mathbf{X}\mathbf{X}^{\mathrm{T}}$$

Different Covariance Functions

- Multi-layer perceptron covariance [23]

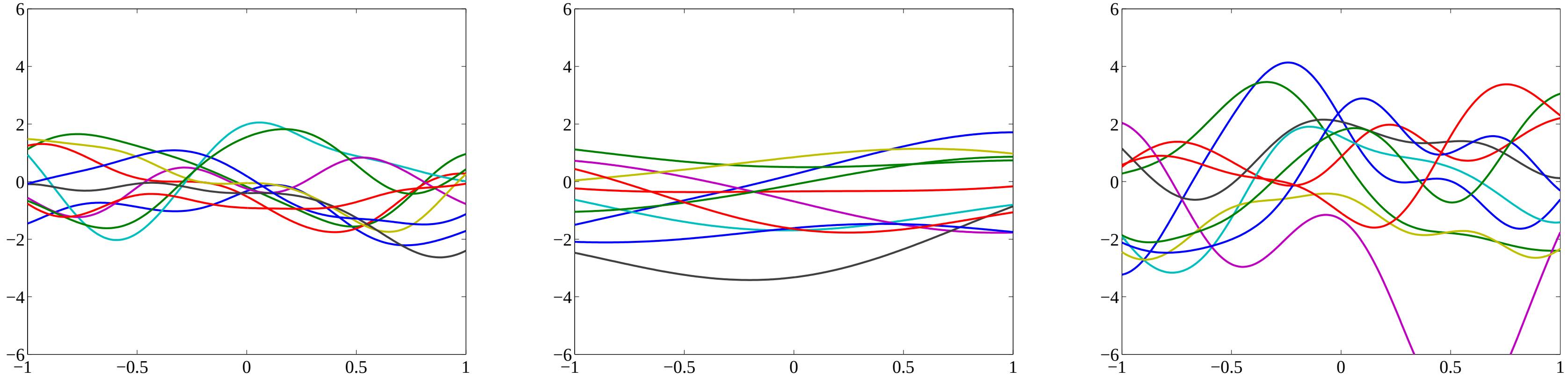
$$k\left(\mathbf{x}_m, \mathbf{x}_n\right) = \alpha \sin^{-1}\left(\frac{w\mathbf{x}_m^{\mathrm{T}}\mathbf{x}_n + b}{\sqrt{w\mathbf{x}_m^{\mathrm{T}}\mathbf{x}_m + b + 1}\sqrt{w\mathbf{x}_n^{\mathrm{T}}\mathbf{x}_n + b + 1}}\right),$$

- Bias

$$k\left(\mathbf{x}_m, \mathbf{x}_n\right) = \alpha,$$

- Parameters of the covariance function found through maximisation of marginal likelihood.

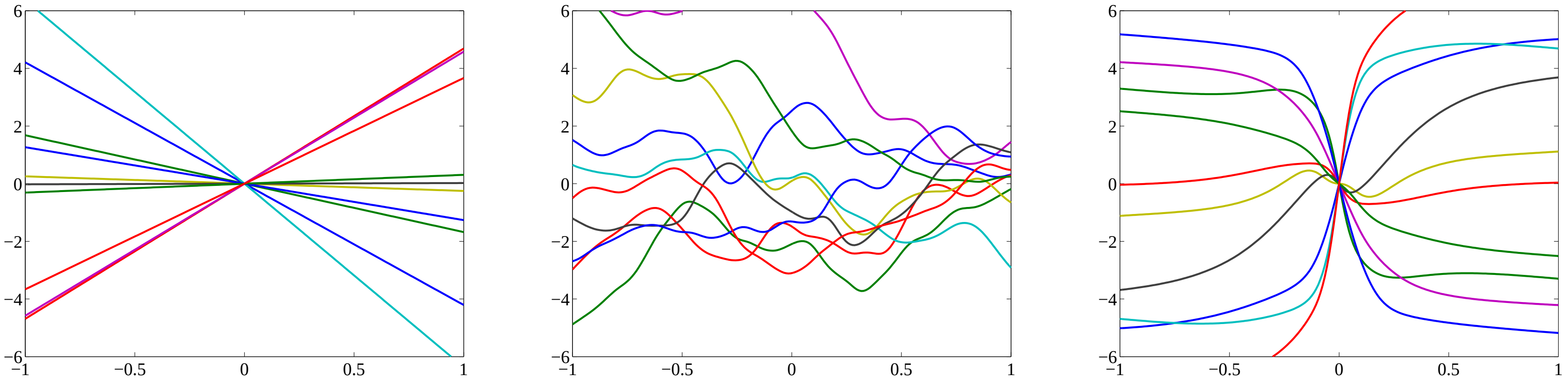
Covariance Samples



(a) (b) (c)

Figure 7: Samples from different covariance functions. (a) RBF kernel with $\gamma = 10$, $\alpha = 1$, (b) RBF kernel with $\gamma = 1$, $\alpha = 1$ (c) RBF kernel with $\gamma = 10$, $\alpha = 4$.

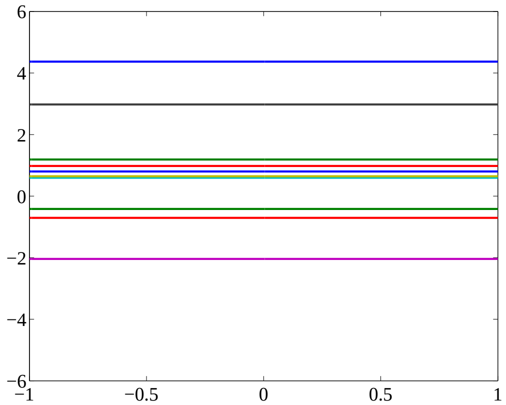
Covariance Samples



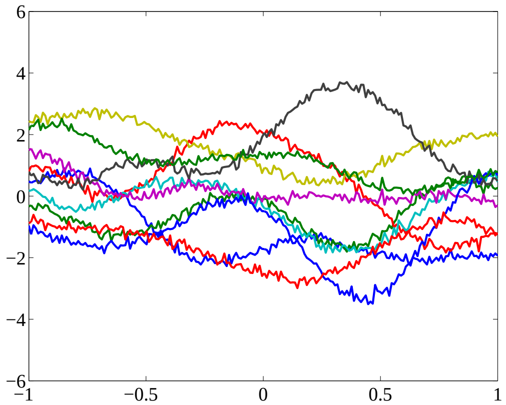
(a) (b) (c)

Figure 8: (a) linear kernel with $\alpha = 16$, (b) MLP kernel with $\alpha = 8$, $w = 100$ and $b = 100$, (c) MLP kernel with $\alpha = 8$, $b = 0$ and $w = 100$.

Covariance Samples



(a)



(b)

Figure 9: (a) bias kernel with $\alpha = 1$ and (b) Summed combination of: RBF kernel, $\alpha = 1$, $\gamma = 10$; bias kernel, α =1; and white noise kernel, $\beta = 100$. Samples can be recreated with the script `demCovFuncSample`.

Consistency

- Predictions remain the same regardless of the number and location of the test points.

$$p(\mathbf{f}_*|\mathbf{f}) = \int p(\mathbf{f}_*, \mathbf{f}_+|\mathbf{f})\, d\mathbf{f}_+,$$

- For the system to be consistent this conditional probability must be independent of the length of $\mathbf{f}_+$.

- In other words.

$$p(\mathbf{f}_*|\mathbf{f}) = \int p(\mathbf{f}_*, \mathbf{f}_+|\mathbf{f})\, d\mathbf{f}_+ = \int p\left(\mathbf{f}_*, \hat{\mathbf{f}}_+|\mathbf{f}\right) d\hat{\mathbf{f}}_+$$

The GP-LVM

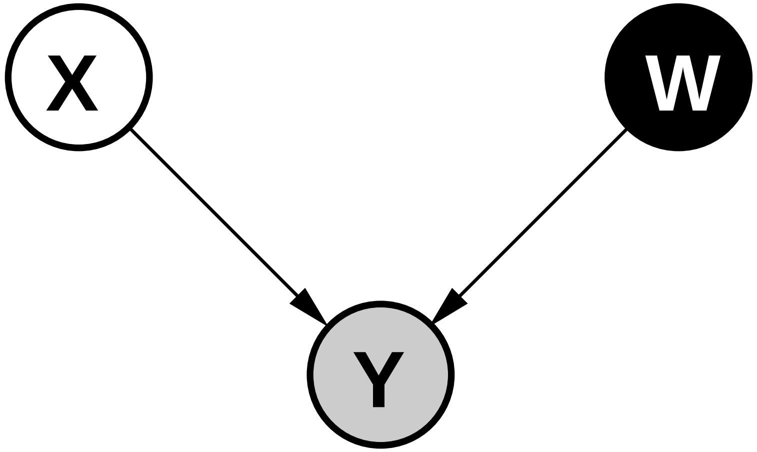
- Probabilistic PCA uses Gaussian likelihood,

$$p\left(\mathbf{Y}|\mathbf{W},\mathbf{X},\beta\right)=\prod_{n=1}^{N}N\left(\mathbf{y}_n|\mathbf{W}\mathbf{x}_n,\beta^{-1}\mathbf{I}\right)$$

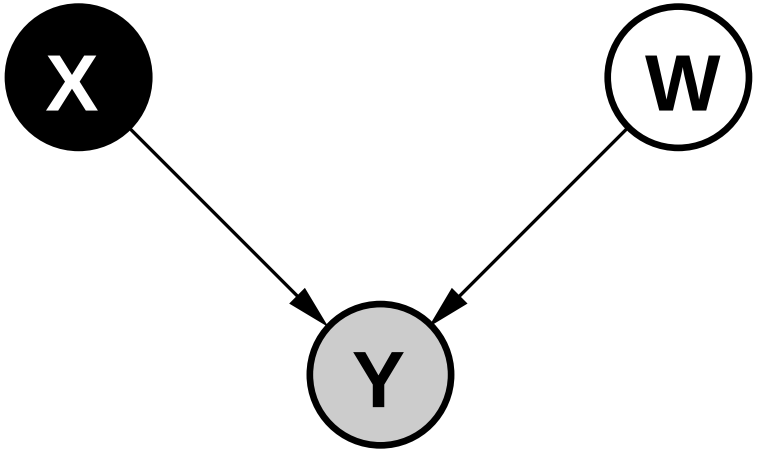
with a Gaussian prior on the latent variables, $\mathbf{X}$.

- GP-LVM: a different perspective on latent variable models.
  - ➥ Rather than marginalising the latent variables
  - ➥ We seek to marginalise the mapping.

Probabilistic PCA vs GP-LVM



(a) Standard Probabilistic PCA



(b) GP-LVM model representation

Linear Mappings and PPCA

- If mappings are constrained linear
  - ➥ dual representation of probabilistic PCA.
- The required marginalisation now takes the form

$$p(\mathbf{Y}|\mathbf{X},\beta) = \int \prod_{n=1}^{N} p(\mathbf{y}_n|\mathbf{x}_n,\mathbf{W},\beta)\, p(\mathbf{W})\, d\mathbf{W}.$$

- Using Gaussian prior distribution

$$p(\mathbf{W}) = \prod_i N(\mathbf{w}_i|\mathbf{0},\mathbf{I}),$$

$\mathbf{w}_i$ is ith row of $\mathbf{W}$.

Marginal Likelihood

- The marginal likelihood is then found as

$$p\left(\mathbf{Y}|\mathbf{X},\beta\right)=\frac{1}{\left(2\pi\right)^{\frac{DN}{2}}\left|\mathbf{K}\right|^{\frac{D}{2}}}\exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\right)\right), \qquad (9)$$

where $\mathbf{K}=\mathbf{X}\mathbf{X}^{\mathrm{T}}+\beta^{-1}\mathbf{I}$ and $\mathbf{X}=\left[\mathbf{x}_1^{\mathrm{T}}\ldots\mathbf{x}_N^{\mathrm{T}}\right]^{\mathrm{T}}$.

Duality

- Note that by taking $\mathbf{C} = \mathbf{W}\mathbf{W}^{\mathrm{T}} + \beta^{-1}\mathbf{I}$ we express PPCA likelihood as

$$p\left(\mathbf{Y}|\mathbf{W},\beta\right) = \frac{1}{(2\pi)^{\frac{DN}{2}}\left|\mathbf{C}\right|^{\frac{N}{2}}}\exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{C}^{-1}\mathbf{Y}^{\mathrm{T}}\mathbf{Y}\right)\right),$$

- Compare with our new model

$$p\left(\mathbf{Y}|\mathbf{X},\beta\right) = \frac{1}{(2\pi)^{\frac{DN}{2}}\left|\mathbf{K}\right|^{\frac{D}{2}}}\exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\right)\right),$$

where $\mathbf{K} = \mathbf{X}\mathbf{X}^{\mathrm{T}} + \beta^{-1}\mathbf{I}$.

GP-LVM Graph

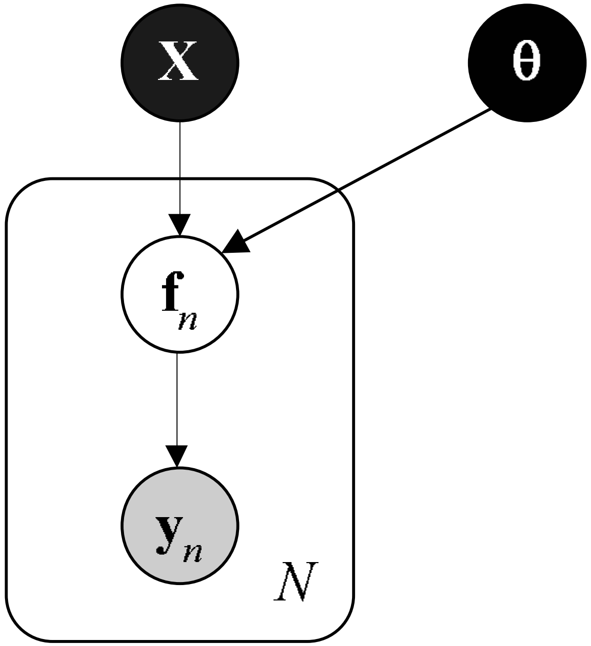


Figure 10: The Gaussian process as a latent variable model, now both kernel parameters, $\boldsymbol{\theta}$and latent positions are optimised.

Gaussian Process

- Optimisation of the new marginal is clearly related to optimisation of the previous likelihood.

- New likelihood is of the form

$$p\left(\mathbf{Y}|\mathbf{X},\beta\right)=\prod_{i=1}^{D}\frac{1}{\left(2\pi\right)^{\frac{N}{2}}\left|\mathbf{K}\right|^{\frac{1}{2}}}\exp\left(-\frac{1}{2}\mathbf{y}_{:,i}^{\mathrm{T}}\mathbf{K}^{-1}\mathbf{y}_{:,i}\right), \tag{10}$$

where $\mathbf{y}_{:,i}$ is the ith column of $\mathbf{Y}$.

- This is recognised as a product of D independent Gaussian processes.

Maximisation of the Marginal Likelihood

- Proof of optimum is the dual of the proof given in [18].
- Maximising log likelihood is equivalent to minimising

$$L = \frac{N}{2}\ln 2\pi + \frac{1}{2}\ln\left|\mathbf{K}\right| + \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{S}\right), \quad (11)$$

where $\mathbf{S} = D^{-1}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}$.

- Gradient of the likelihood wrt $\mathbf{X}$ is

$$\frac{\partial L}{\partial \mathbf{X}} = -\mathbf{K}^{-1}\mathbf{S}\mathbf{K}^{-1}\mathbf{X} + \mathbf{K}^{-1}\mathbf{X},$$

setting the equation to zero and pre-multiplying by $\mathbf{K}$ gives

$$\mathbf{S}\left[\beta^{-1}\mathbf{I} + \mathbf{X}\mathbf{X}^{\mathrm{T}}\right]^{-1}\mathbf{X} = \mathbf{X}.$$

Singular Value Decomposition

- Substitute $\mathbf{X}$ with its SVD, $\mathbf{X} = \mathbf{U}\mathbf{L}\mathbf{V}^{\mathrm{T}}$, giving

$$\mathbf{S}\mathbf{U}\left[\mathbf{L} + \beta^{-1}\mathbf{L}^{-1}\right]^{-1}\mathbf{V}^{\mathrm{T}} = \mathbf{U}\mathbf{L}\mathbf{V}^{\mathrm{T}}$$

- Right multiplying both sides by $\mathbf{V}$ giving

$$\mathbf{S}\mathbf{U} = \mathbf{U}\left(\beta^{-1}\mathbf{I} + \mathbf{L}^2\right),$$

- Since $\left(\beta^{-1}\mathbf{I} + \mathbf{L}^2\right)$ is diagonal, this is an eigenvalue problem.

- $\mathbf{U}$ are eigenvectors of $\mathbf{S}$ and $\Lambda = \left(\beta^{-1}\mathbf{I} + \mathbf{L}^2\right)$ are the eigenvalues.

- Thus elements from diagonal of $\mathbf{L}$ are

$$l_i = \left(\lambda_i - \beta^{-1}\right)^{\frac{1}{2}}.$$

The Retained Eigenvalues

- If $q < D$ need to select which eigenvectors to retain.
- All eigenvectors are associated with stationary points.
- Can rewrite objective as a difference of geometric and arithmetic mean.
  - This implies eigenvalues must be neighbouring.
- Solution for β becomes negative if largest eigenvalues aren't retained.

Equivalence of Eigenvalue Problems

- For DPPCA the eigenvalue problem is of the form

$$\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{U} = \mathbf{U}\Lambda.$$

Premultiplying by $\mathbf{Y}^{\mathrm{T}}$ gives

$$\mathbf{Y}^{\mathrm{T}}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{U} = \mathbf{Y}^{\mathrm{T}}\mathbf{U}\Lambda \qquad (12)$$

- $\mathbf{U}$ are the eigenvectors of $\mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ so $\mathbf{U}^{\mathrm{T}}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{U} = \Lambda$,
  - Matrix $\mathbf{U}' = \mathbf{Y}^{\mathrm{T}}\mathbf{U}\Lambda^{-\frac{1}{2}}$ is orthonormal.
- Post multiplying both sides of (12) by $\Lambda^{-\frac{1}{2}}$ gives

$$\mathbf{Y}^{\mathrm{T}}\mathbf{Y}\mathbf{U}' = \mathbf{U}'\Lambda$$

which is the form of the eigenvalue problem associated with PPCA.

Non-linear GP-LVM

- PCA can be interpreted as a product of linear Gaussian processes.
- Can replace the linear kernel and obtain non-linear latent variable models.
- Can no-longer use an eigenvalue problem to solve.

Optimisation of the Non-linear Model

- No closed form solution for non-linear model.

- Gradients of kernel required

$$\frac{\partial L}{\partial \mathbf{K}} = \mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{K}^{-1} - D\mathbf{K}^{-1}, \tag{13}$$

And combined with $\frac{\partial \mathbf{K}}{\partial x_{n,j}}$ via chain rule.

Illustration of GP-LVM via SCG

- Oil Data
  - Twelve dimensional data set.
  - Oil flow in a pipeline.
  - Stratified, annular and homogeneous.

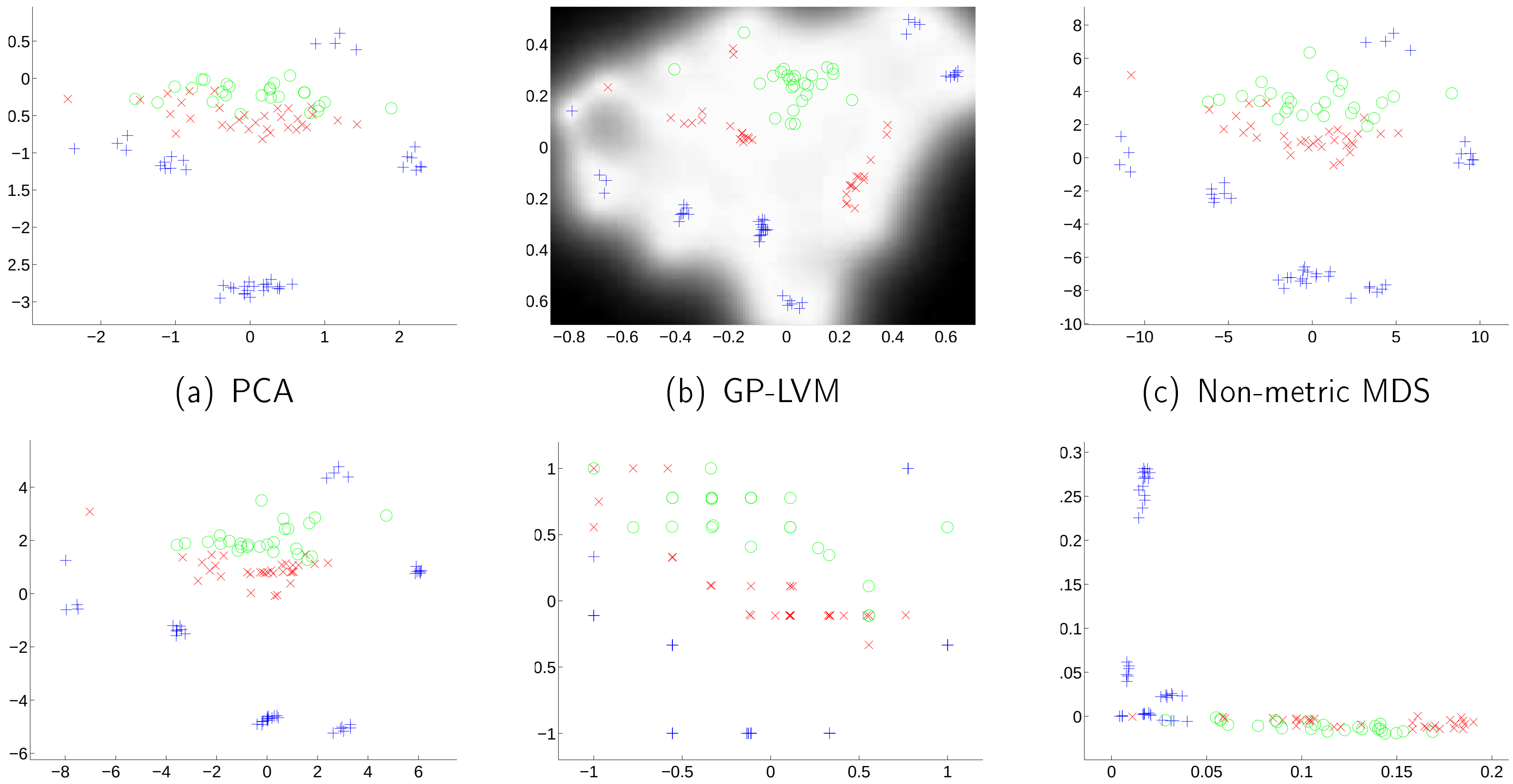
Oil Data Contd

- Compare GP-LVM with MDS methods.

- Use RBF kernel for GP-LVM

$$k(\mathbf{x}_i, \mathbf{x}_j) = \alpha_{\mathrm{rbf}} \exp\left(-\frac{\gamma}{2}(\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}}(\mathbf{x}_i - \mathbf{x}_j)\right) + \alpha_{\mathrm{bias}} + \beta^{-1}\delta_{ij}.$$

optimise jointly with respect to $\mathbf{X}$, α_{bias}, α_{rbf}, β and γ.



(a) PCA (b) GP-LVM (c) Non-metric MDS

(d) Metric MDS (e) GTM (f) kernel PCA

Nearest Neighbour Errors

Method	PCA	GP-LVM	Non-metric MDS	Metric MDS	GTM*	kernel PCA*
Errors	20	4	13	6	7	13

Table 1: Errors made by the different methods when using the latent-space for nearest neighbour classification in the latent space. Both the GTM and kernel PCA are given asterisks as the result shown is the best obtained for each method from a range of different parameterisations.

Visualising the Uncertainty

- Likelihood (10) a product of D separate Gaussian processes.
- We have maintained the implicit assumption in PCA that *a priori* each dimension is identically distributed.
- This leads to an *a posteriori* shared level of uncertainty in each process.
- This allows us to visualise the uncertainty in the latent space.
- The uncertainty is visualised by varying the intensity of the background pixels.

Computational Complexity

- Each gradient step requires an inverse of the kernel matrix.
- This is an $O\left(N^3\right)$ operation.
- Renders the algorithm impractical for many data sets of interest.
- Seek to maximise a sparse approximation to the full likelihood.

Large Data Sets

- In [3, 4] a sub-set of data approach is suggested.
- This approach has two main drawbacks:
  - ➥ It suffers from the lack of a convergence criterion.
  - ➥ It discards information in the data set.
- A more promising approach is suggested in [16] and developed in [11].
- This approach is now available in the FGPLVM toolbox and documented in [5, in preparation].

Sub-set of Data

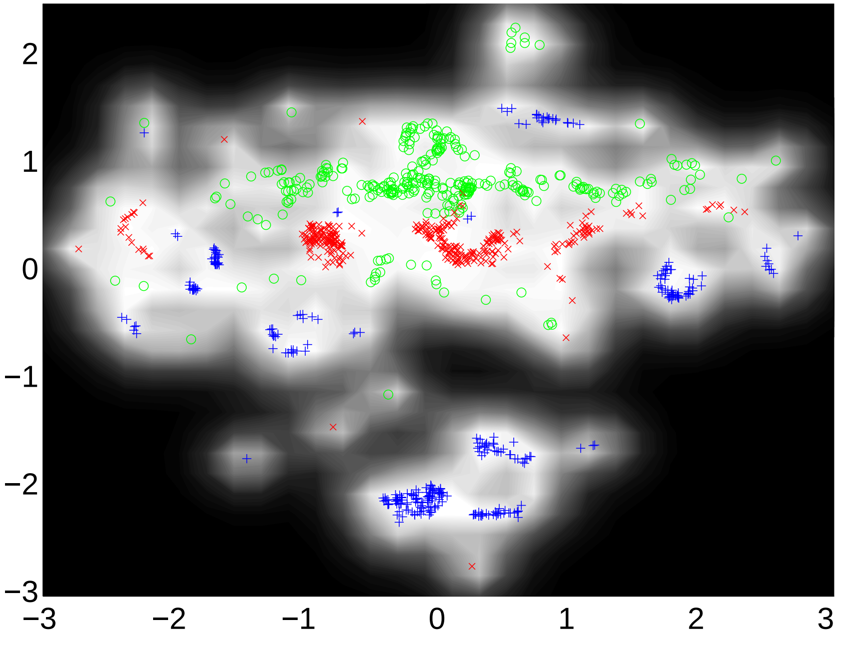


Figure 11: The full oil flow data set visualised with an RBF based kernel using sub-set of data approximations.

Full GP-LVM

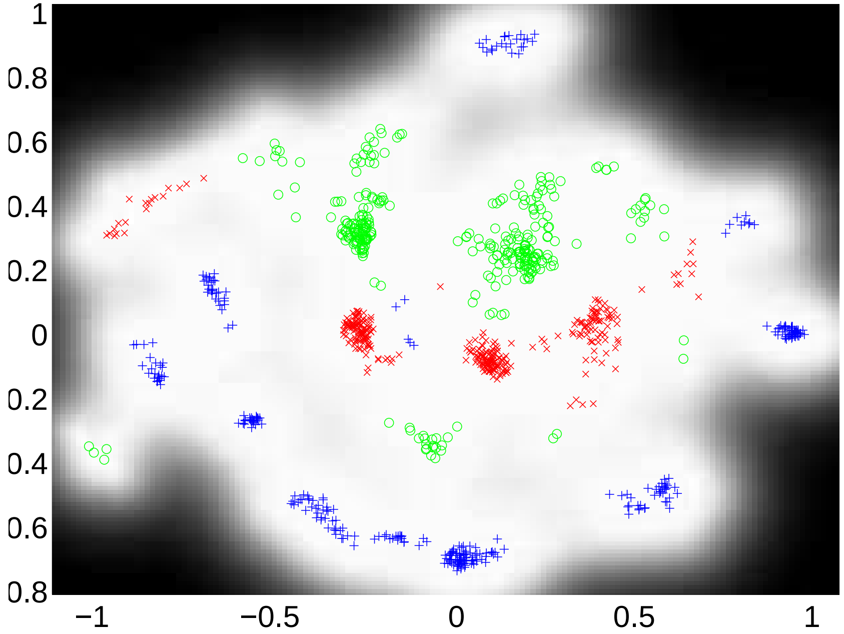


Figure 12: The full GP-LVM algorithm with RBF kernel on the oil flow data (uses the GPLVMCPP toolbox).

Nearest Neighbour in $\mathbf{X}$

Model	PCA	Sparse GP-LVM (IVM)	GP-LVM (RBF)	GTM	Y
Errors	162	24	1	11	2

Table 2: Number of errors for nearest neighbour classification in the latent-space for the full oil data set (1000 points). Far right column contains result for nearest neighbour in the data space, also presented is a result for the GTM algorithm.

Back Constraints

- Joint work with Joaquin Quinoñero Candela
- GP-LVM provides a smooth mapping from latent space to the data space.
- Points close in latent space will be close in data space.
- Does not imply that points which are close in data space will be close in latent space.
- In recent work [6, in preparation] use of back constraints is suggested.

Back Constraints II

- Back constraints constrain latent points to be a smooth function of data points.

$$x_{n,i} = f_i\left(\mathbf{y}_n, \mathbf{a}_i\right)$$

  where $\mathbf{w}$ are parameters.

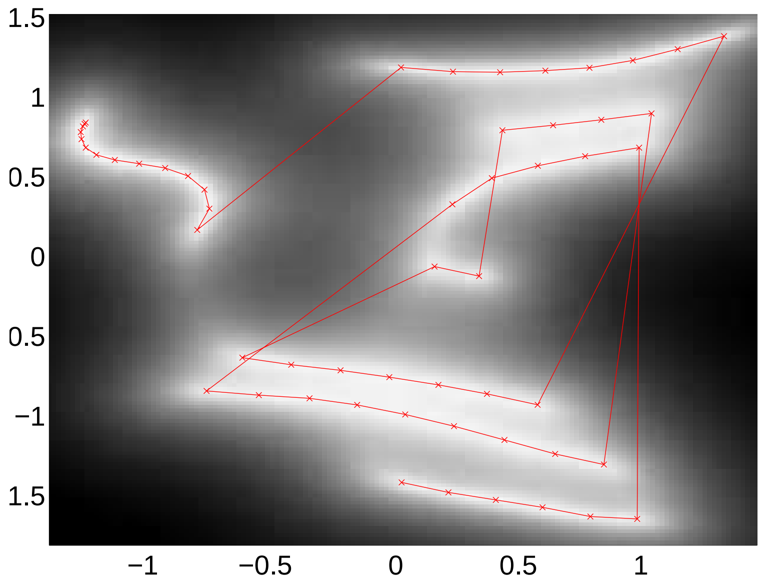
- Instead of maximising wrt $\mathbf{X}$, maximise wrt $\mathbf{A} = \left[\mathbf{a}_1, \dots, \mathbf{a}_q\right]^{\mathrm{T}}$.

- This forces points which are close in data space to be close in latent space.

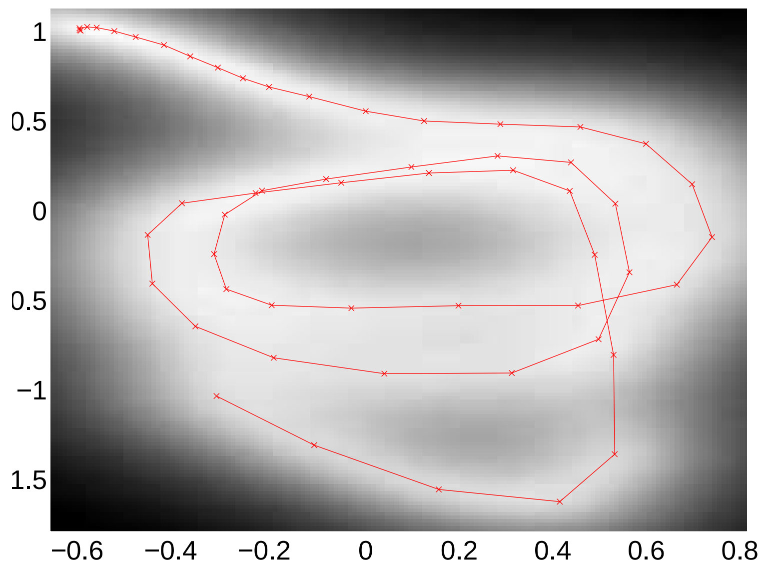
Motion Capture Data

- Motion capture of a man running.
- Subject breaking into a run from standing.
- Approximately three full strides in the sequence.
- Data is mean subtracted so subject runs 'in place'.
  - ➥ The data is therefore somewhat periodic in nature.
  - ➥ Angle of run changes during sequence.
- Compare pure GP-LVM with GP-LVM with back constraints.
- The back constraint was implemented through an RBF based kernel mapping with $\gamma = 1 \times 10^{-3}$.

Running Man



(a) Pure GP-LVM $L = 1,543$

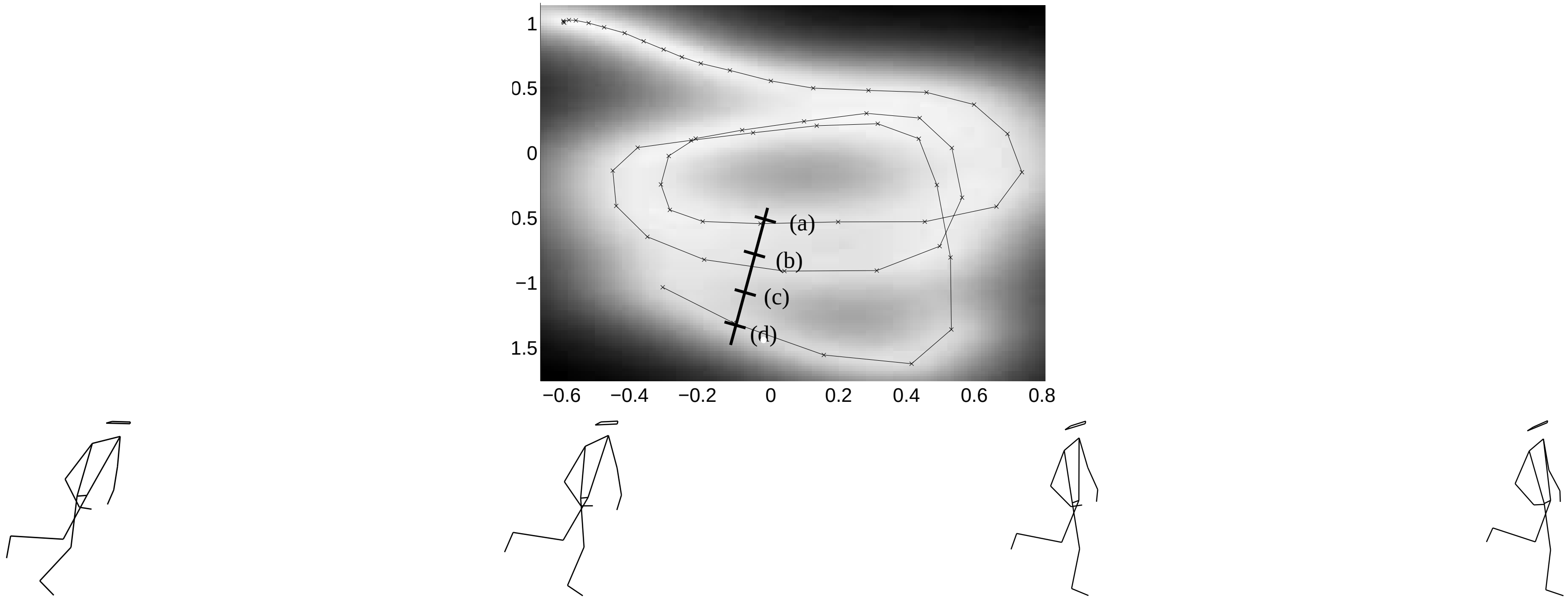


(b) Back constraints $L = 1,000$

Running Man

- The likelihood of the pure model is higher.
- However the sequence is split across sub-sequences.
- A circular structure is necessary for periodic nature of data.
- Squashed spiral has either
  - less representational power in the inner rings (inner groove distortion) in gramophone records)
  - or will crosses over itself (not consistent with data).
- Note: using a three dimensional latent space alleviates the problem.

Run Angle



(c) (d) (e) (f)

Figure 13:

Vowel Data

- Single speaker vowel data set (collaboration with Jon Malkin and Jeff Bilmes).
- Cepstral coefficients and deltas of ten different vowel phonemes.
- Data acquired as part of a vocal joystick system [1].
- PCA fails to separate the vowels.
- PCA initialised GP-LVM therefore fragments the vowels.
- Back constraints fix this.

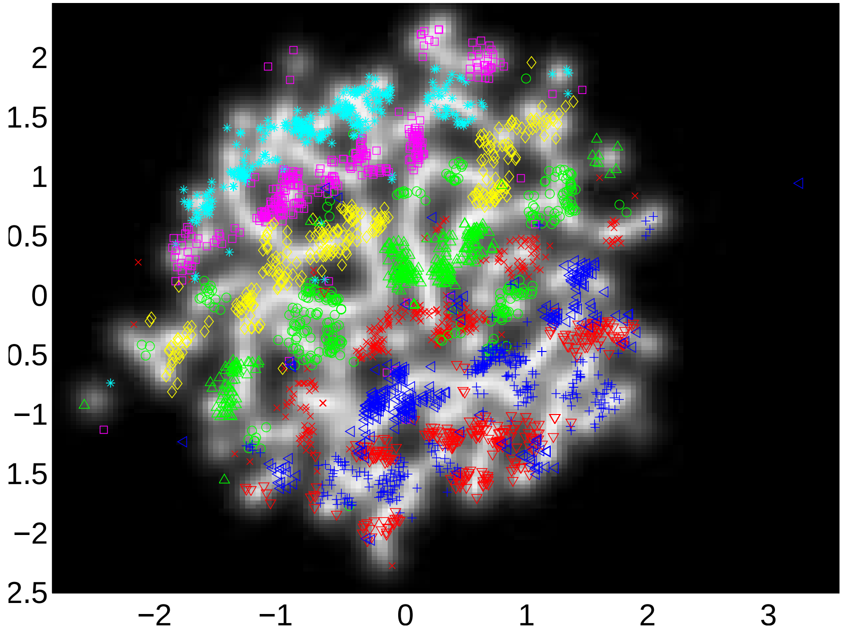


Figure 14: Pure GP-LVM. */a/* red cross */ae/* green circle */ao/* blue plus */e/* cyan asterix */i/* magenta square */ibar/* yellow diamond */o/* red down triangle */schwa/* green up triangle and */u/* blue left triangle.

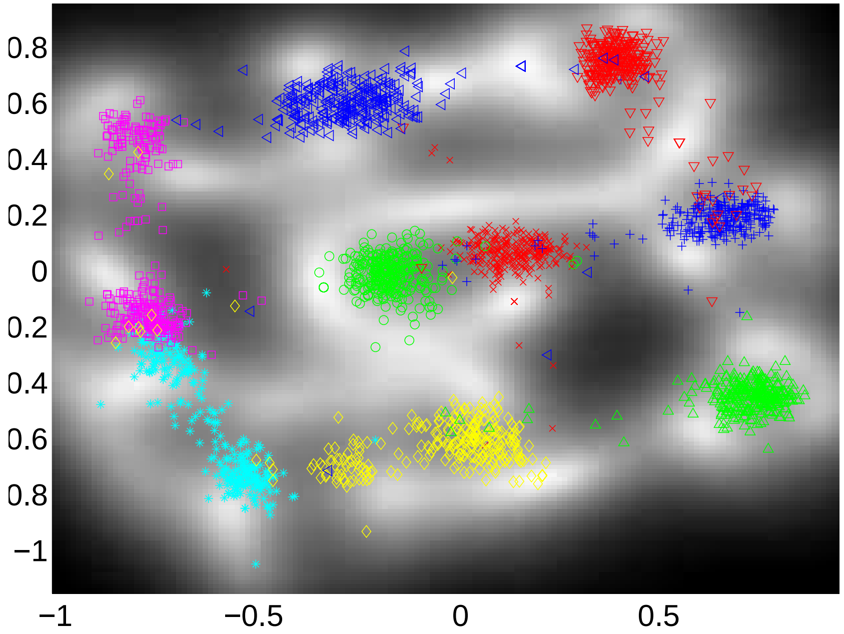


Figure 15: Back constrained. */a/* red cross */ae/* green circle */ao/* blue plus */e/* cyan asterix */i/* magenta square */ibar/* yellow diamond */o/* red down triangle */schwa/* green up triangle and */u/* blue left triangle.

GP-LVM with Dynamics

- Recently [20] described an approach to adding dynamics to the GP-LVM.
- Assume data is presented in temporal order.
- Place a Markov chain distribution over the latent space by defining $p\left(\mathbf{x}_n|\mathbf{x}_{n-1}\right)$.
- Leads to a prior distribution $p\left(\mathbf{X}\right) = p\left(\mathbf{x}_1\right)\prod_{n=2}^{N} p\left(\mathbf{x}_n|\mathbf{x}_{n-1}\right)$.
- Marginalising $\mathbf{X}$ is now intractable.

MAP Solution

- It is straightforward to obtain maximum *a posteriori* (MAP) estimates of the solution.
- In [20] using a Gaussian process to relate $\mathbf{x}_n$ to $\mathbf{x}_{n-1}$ is suggested.
- Joint likelihood is then given by

$$
\begin{aligned}
p(\mathbf{Y}, \mathbf{X}) = & -\frac{DN}{2}\log 2\pi - \frac{D}{2}\log|\mathbf{K}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\right) \\
& -\frac{qN}{2}\log 2\pi - \frac{q}{2}\log|\mathbf{K}_x| - \frac{1}{2}\mathrm{tr}\left(\mathbf{K}_x^{-1}\left(\hat{\mathbf{X}} - \tilde{\mathbf{X}}\right)\left(\hat{\mathbf{X}} - \tilde{\mathbf{X}}\right)^{\mathrm{T}}\right),
\end{aligned}
$$

  where $\hat{\mathbf{X}} = \left[\mathbf{x}_2 \ldots \mathbf{x}_N\right]^{\mathrm{T}}$ and $\tilde{\mathbf{X}} = \left[\mathbf{x}_1 \ldots \mathbf{x}_{N-1}\right]^{\mathrm{T}}$.

- $\mathbf{K}_x$ is the dynamics covariance function, constructed on $\tilde{\mathbf{X}}$.

Implementation

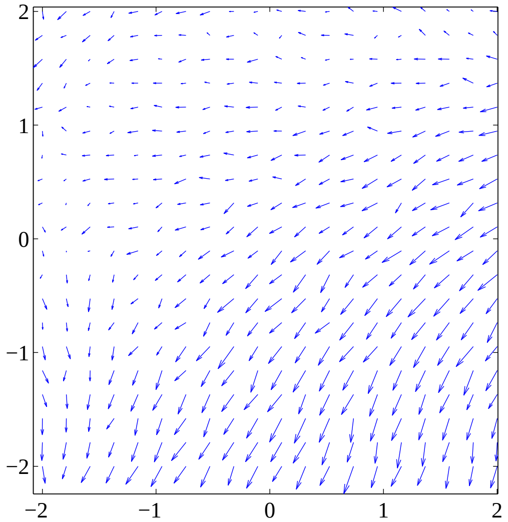
- For dynamics, use an RBF kernel and a white noise term,

$$k\left(\mathbf{x}_n, \mathbf{x}_m\right) = \alpha'_{\text{rbf}} \exp\left(-\frac{\gamma'}{2}\left(\mathbf{x}_n - \mathbf{x}_m\right)^{\mathrm{T}}\left(\mathbf{x}_n - \mathbf{x}_m\right)\right) + \beta'^{-1}\delta_{nm}.$$

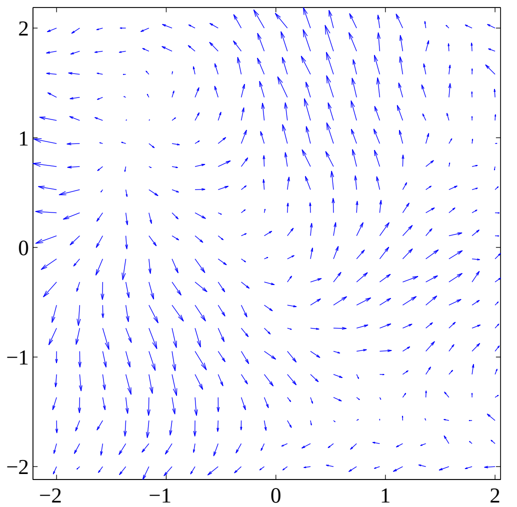
  δ_{nm} is the Kronecker delta function.

- Can fix the dynamics model parameters by hand.
- The signal variance is given by α'_{rbf} and the noise variance by β'^{-1},
- γ controls the smoothness.
- We now show some samples from dynamics covariances.

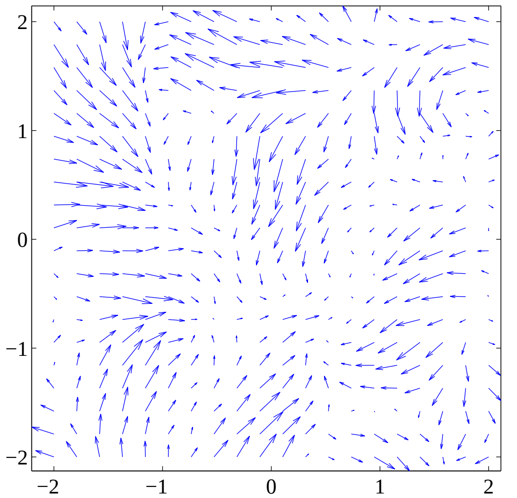
Dynamics Samples I



(a) $\gamma' = 0.2, \beta'^{-1} = 4 \times 10^{-4}$



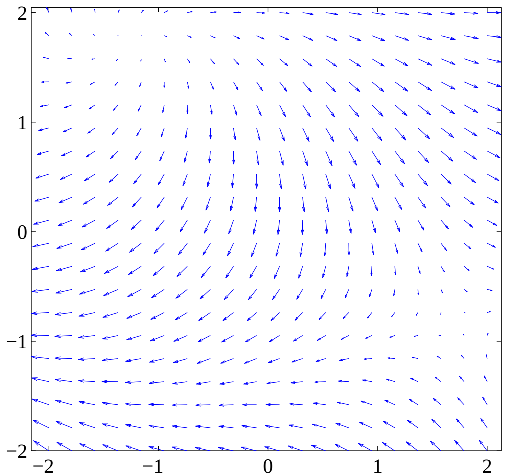
(b) $\gamma' = 1, \beta' = 4 \times 10^{-4}$



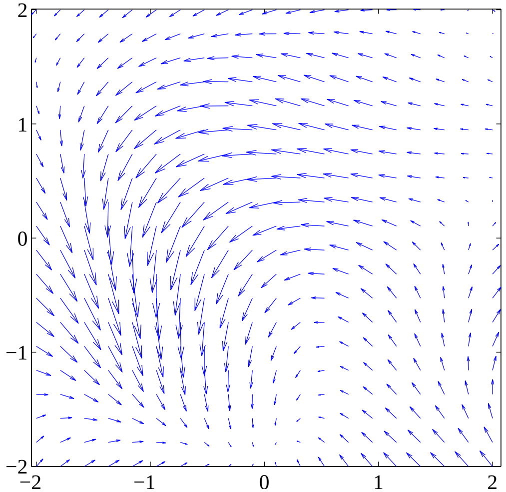
(c) $\gamma' = 5, \beta'^{-1} = 4 \times 10^{-4}$

Figure 16: $\alpha'_{\mathrm{rbf}} = 0.1$

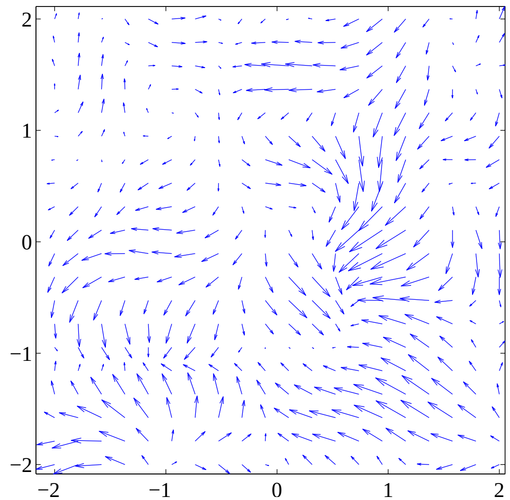
Dynamics Samples II



(a) $\gamma' = 0.2, \beta'^{-1} = 1 \times 10^{-6}$



(b) $\gamma' = 1, \beta'^{-1} = 1 \times 10^{-6}$



(c) $\gamma' = 5, \beta'^{-1} = 4 \times 10^{-6}$

Figure 17: $\alpha'_{\mathrm{rbf}} = 0.1$

Running Man + Dynamics

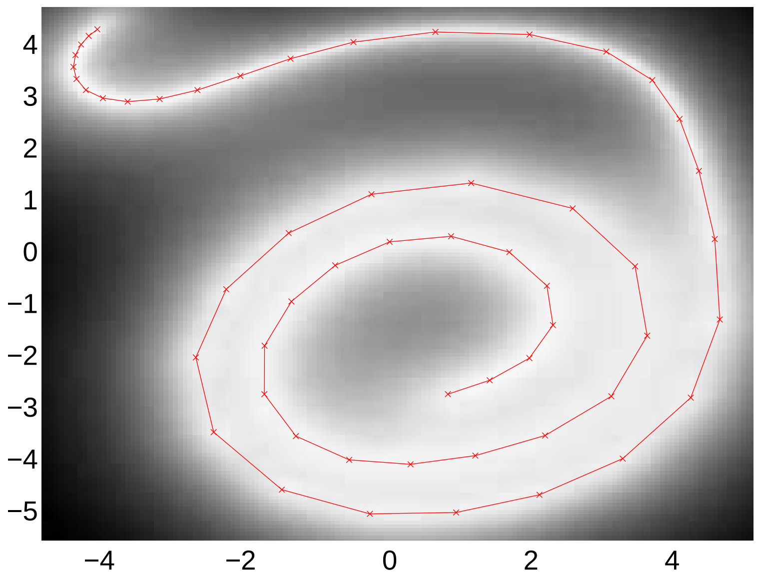
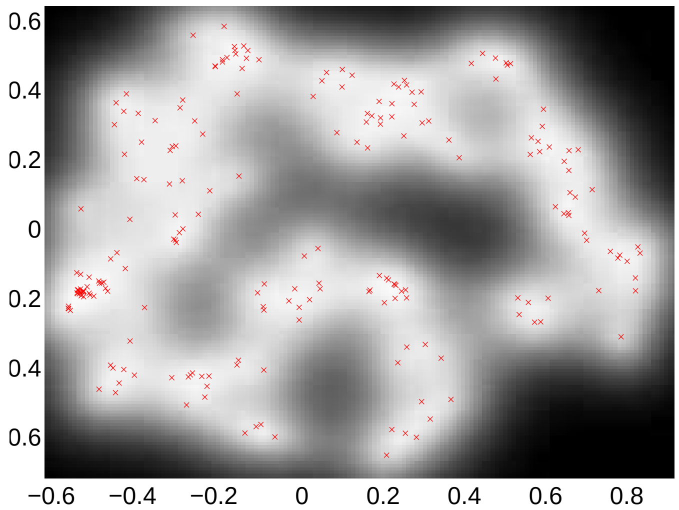


Figure 18: Using dynamics from previous slide (a)

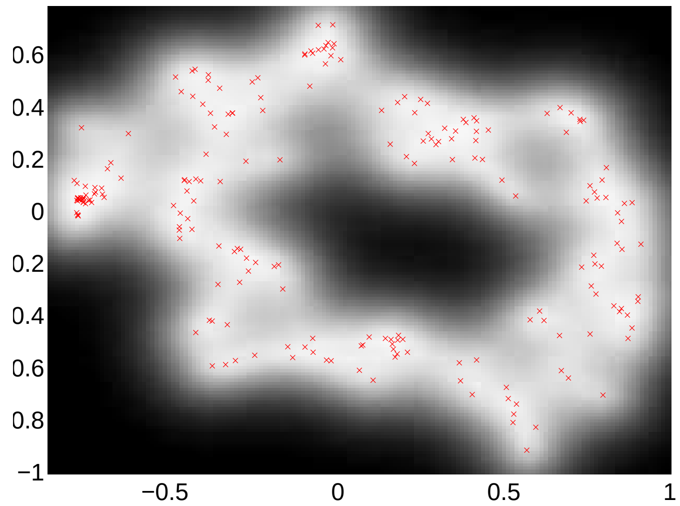
Loop Closure in Robotics

- On-going work with Dieter Fox and Brian Ferris at the University of Washington.
- Robot navigation via wireless access points.
- Robot completes one loop ... space is inherently 2d.

Loop Closure

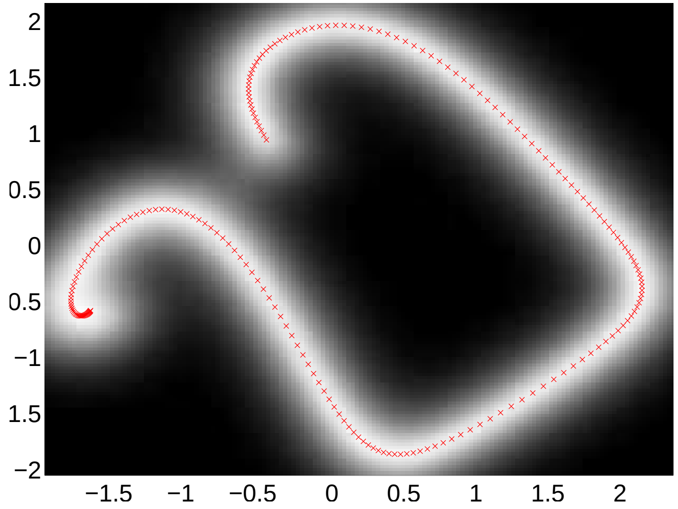


(a) Pure GP-LVM

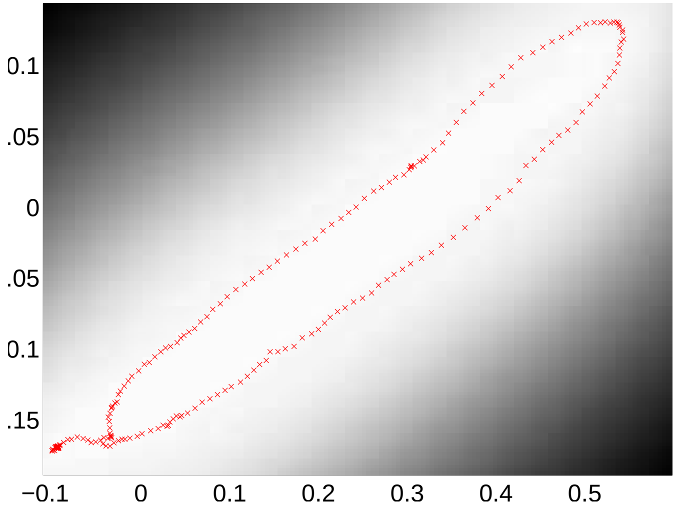


(b) Back Constraints

Loop Closure II



(c) Pure GP-LVM



(d) Back Constraints

Conclusions

- GP-LVM
  - Probabilistic model: combines PPCA and GPs
- Computational issues for larger data sets.
  - Sub-set of data methods.
  - Code available for more advanced approaches.
- Extensions - Dynamics
  - Forces temporal continuity in latent space.
  - We advocated manual setting of kernel parameters & sampling.
- Extension - Back Constraints
  - Force local distance preservation.
  - Can be combined with back constraints.

References

[1] J. Bilmes, J. Malkin, X. Li, S. Harada, K. Kilanski, K. Kirchhoff, R. Wright, A. Subramanya, J. Landay, P. Dowden, and H. Chizeck. The vocal joystick. In *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing*. IEEE, May 2006. To appear.

[2] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popovic. Style-based inverse kinematics. In *ACM Transactions on Graphics (SIGGRAPH 2004)*, 2004.

[3] 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.

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

[5] N. D. Lawrence. Large scale learning with the Gaussian process latent variable model. Technical report, University of Sheffield, 2006.

[6] N. D. Lawrence and J. Quiñonero Candela. Local distance preservation in the GP-LVM through back constraints.

Technical report, University of Sheffield, 2006. In preparation.

[7] 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.

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

[9] A. O'Hagan. Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society, B*, 40:1–42, 1978.

[10] A. O'Hagan. Some Bayesian numerical analysis. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, editors, *Bayesian Statistics 4*, pages 345–363, Valencia, 1992. Oxford University Press.

[11] 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.

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

[13] 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.

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

[15] 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. [22].

[16] E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In Weiss et al. [22].

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

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

[19] 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, 17-20 Oct. 2005. IEEE Computer Society Press.

[20] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models. In Weiss et al. [22].

[21] K. Q. Weinberger, F. Sha, and L. K. Saul. Learning a kernel matrix for nonlinear dimensionality reduction. In R. Greiner and D. Schuurmans, editors, *Proceedings of the Interna-*

tional Conference in Machine Learning, volume 21, pages 839–846, San Francisco, CA, 2004. Morgan Kauffman.

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

[23] C. K. I. Williams. Computing with infinite networks. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, Cambridge, MA, 1997. MIT Press.

[24] C. K. I. Williams. Prediction with Gaussian processes: From linear regression to linear prediction and beyond. In M. I. Jordan, editor, *Learning in Graphical Models*, volume 89 of *Series D: Behavioural and Social Sciences*. Kluwer, Dordrecht, The Netherlands, 1998.

[25] C. K. I. Williams and C. E. Rasmussen. Gaussian processes for regression. In D. Touretzky, M. Mozer, and M. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 514–520, Cambridge, MA, 1996. MIT Press.