All you want to know about GPs: Gaussian Process Latent Variable Model

Raquel Urtasun and Neil Lawrence

TTI Chicago, University of Sheffield

June 16, 2012

Motivation for Non-Linear Dimensionality Reduction

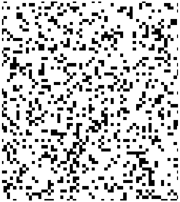
USPS Data Set Handwritten Digit

- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.

Motivation for Non-Linear Dimensionality Reduction

USPS Data Set Handwritten Digit

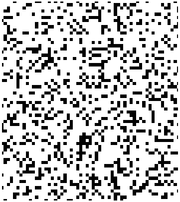
- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!



Motivation for Non-Linear Dimensionality Reduction

USPS Data Set Handwritten Digit

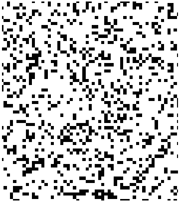
- 3648 Dimensions
  - 64 rows by 57 columns
  - Space contains more than just this digit.
  - Even if we sample every nanosecond from now until the end of the universe, you won't see the original six!



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

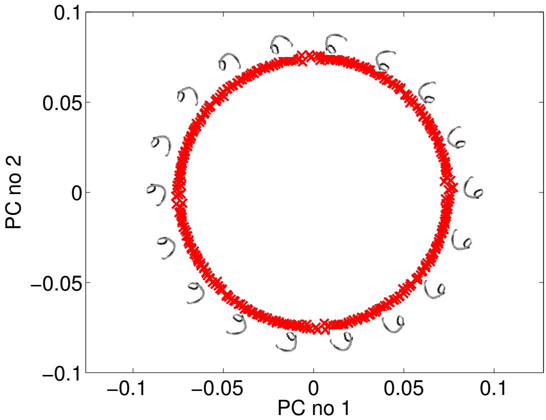
Rotate a 'Prototype'

MATLAB Demo

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

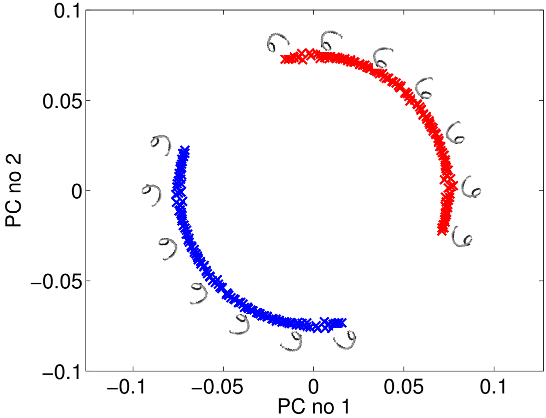
MATLAB Demo

```
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.

Feature Selection

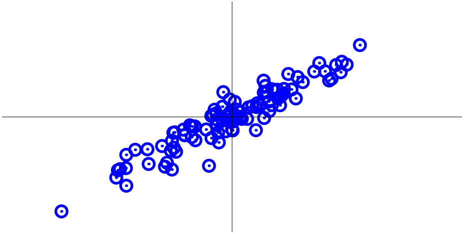


Figure: `demRotationDist`. Feature selection via distance preservation.

Feature Selection

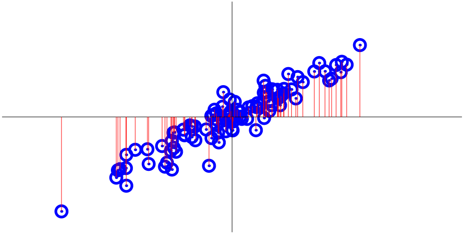


Figure: `demRotationDist`. Feature selection via distance preservation.

Feature Selection

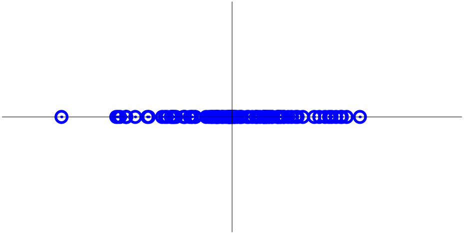


Figure: `demRotationDist`. Feature selection via distance preservation.

Feature Extraction

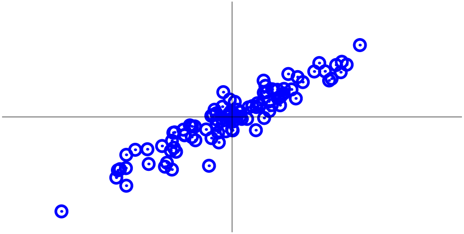


Figure: `demRotationDist`. Rotation preserves interpoint distances.

Feature Extraction

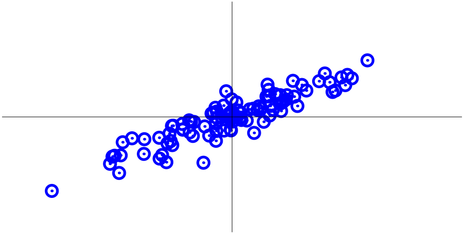


Figure: `demRotationDist`. Rotation preserves interpoint distances.

Feature Extraction

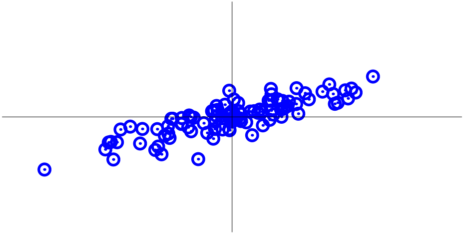


Figure: `demRotationDist`. Rotation preserves interpoint distances.

Feature Extraction

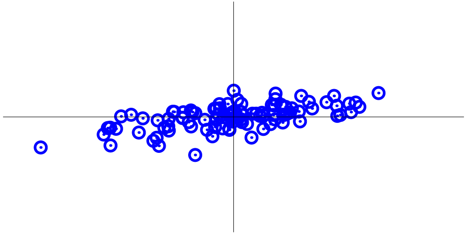


Figure: `demRotationDist`. Rotation preserves interpoint distances.

Feature Extraction

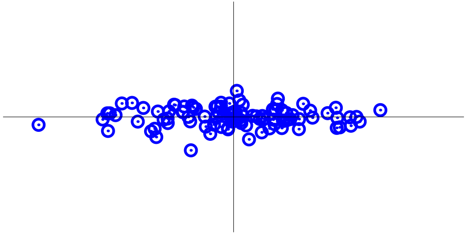


Figure: `demRotationDist`. Rotation preserves interpoint distances.

Feature Extraction

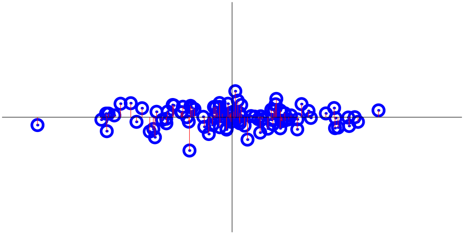


Figure: `demRotationDist`. Rotation preserves interpoint distances. Residuals are much reduced.

Feature Extraction

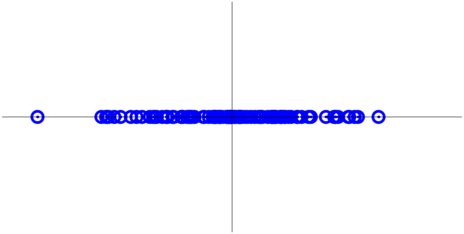


Figure: `demRotationDist`. Rotation preserves interpoint distances. Residuals are much reduced.

Which Rotation?

- We need the rotation that will minimise residual error.
- Retain features/directions with *maximum variance*.

Which Rotation?

- We need the rotation that will minimise residual error.
- Retain features/directions with *maximum variance*.
- Error is then given by the sum of residual variances.

$$E(\mathbf{X}) = \frac{2}{p} \sum_{k=q+1}^{p} \sigma_k^2.$$

Which Rotation?

- We need the rotation that will minimise residual error.
- Retain features/directions with *maximum variance*.
- Error is then given by the sum of residual variances.

$$E(\mathbf{X}) = \frac{2}{p} \sum_{k=q+1}^{p} \sigma_k^2.$$

- Rotations of data matrix *do not* effect this analysis.
- Rotate data so that largest variance directions are retained.

Which Rotation?

- We need the rotation that will minimise residual error.
- Retain features/directions with *maximum variance*.
- Error is then given by the sum of residual variances.

$$E(\mathbf{X}) = \frac{2}{p} \sum_{k=q+1}^{p} \sigma_k^2.$$

- Rotations of data matrix *do not* effect this analysis.
- Rotate data so that largest variance directions are retained.

Reminder: Principal Component Analysis

- How do we find these directions?
- Find directions in data with maximal variance.
  - That's what PCA does!
- **PCA**: rotate data to extract these directions.
- **PCA**: work on the sample covariance matrix $\mathbf{S} = n^{-1}\hat{\mathbf{Y}}^{\top}\hat{\mathbf{Y}}$.

Principal Coordinates Analysis

- The rotation which finds directions of maximum variance is the eigenvectors of the covariance matrix.
- The variance in each direction is given by the eigenvalues.
- **Problem:** working directly with the sample covariance, $\mathbf{S}$, may be impossible.

Equivalent Eigenvalue Problems

- Principal Coordinate Analysis operates on $\hat{\mathbf{Y}}^T\hat{\mathbf{Y}}$.
- Two eigenvalue problems are equivalent. One solves for the rotation, the other solves for the location of the rotated points.
- When $p < n$ it is easier to solve for the rotation, $\mathbf{R}_q$. But when $p > n$ we solve for the embedding (principal coordinate analysis). from distance matrix.
- Can we compute $\hat{\mathbf{Y}}\hat{\mathbf{Y}}^\top$ instead?

Equivalent Eigenvalue Problems

- Principal Coordinate Analysis operates on $\hat{\mathbf{Y}}^T\hat{\mathbf{Y}}$.
- Two eigenvalue problems are equivalent. One solves for the rotation, the other solves for the location of the rotated points.
- When $p < n$ it is easier to solve for the rotation, $\mathbf{R}_q$. But when $p > n$ we solve for the embedding (principal coordinate analysis). from distance matrix.
- Can we compute $\hat{\mathbf{Y}}\hat{\mathbf{Y}}^\top$ instead?

The Covariance Interpretation

- $n^{-1}\hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}$ is the data covariance.
- $\hat{\mathbf{Y}}\hat{\mathbf{Y}}^\top$ is a centred inner product matrix.
  - Also has an interpretation as a covariance matrix (Gaussian processes).
  - It expresses correlation and anti correlation between *data points*.
  - Standard covariance expresses correlation and anti correlation between *data dimensions*.

Summary up to know on dimensionality reduction

- Distributions can behave very non-intuitively in high dimensions.
- Fortunately, most data is not really high dimensional.
- Probabilistic PCA exploits linear low dimensional structure in the data.
  - Probabilistic interpretation brings with it many advantages: extensibility, Bayesian approaches, missing data.
- Didn't deal with the non-linearities highlighted by the six example!
- Let's look at *non linear* dimensionality reduction.

Spectral methods

- LLE (Roweis & Saul, 00) , ISOMAP (Tenenbaum et al. 00), Laplacian Eigenmaps (Belkin &Niyogi, 01)
- Based on local distance preservation

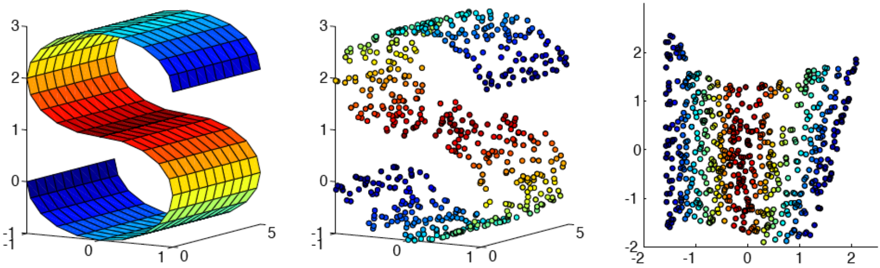
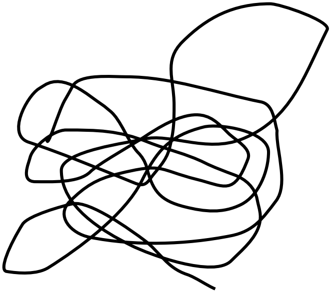


Figure: LLE embeddings from densely sampled data

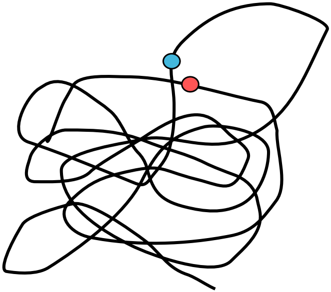
Tangled String

- Sometimes local distance preservation in data space is wrong.
- The pink and blue ball should be separated.



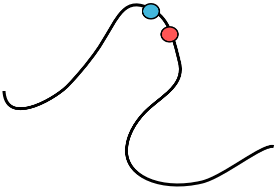
Tangled String

- Sometimes local distance preservation in data space is wrong.
- The pink and blue ball should be separated.
- But the assumption makes the problem simpler (for spectral methods it is convex).



Tangled String

- Sometimes local distance preservation in data space is wrong.
- The pink and blue ball should be separated.
- But the assumption makes the problem simpler (for spectral methods it is convex).



Generative Models

- Directly model the generating process.
- Map from string to position in space.
- How to model observation "generation"?

Example of data generation

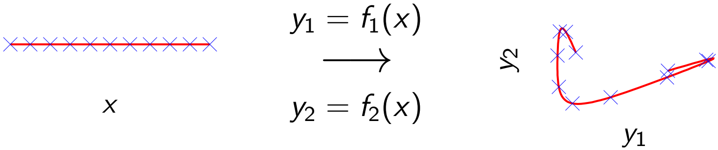


Figure: A string in two dimensions, formed by mapping from one dimension, x, line to a two dimensional space, $[y_1,\ y_2]$ using nonlinear functions $f_1(\cdot)$ and $f_2(\cdot)$.

Difficulty for Probabilistic Approaches

- Propagate a probability distribution through a non-linear mapping.
- Normalisation of distribution becomes intractable.

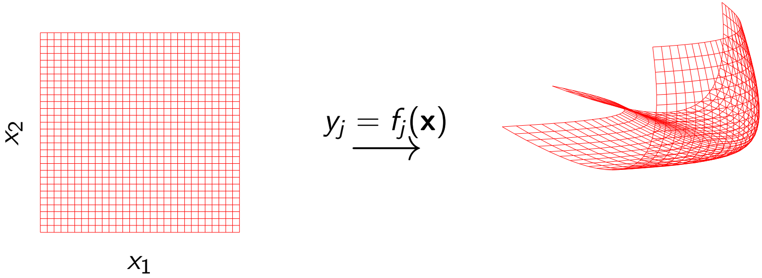


Figure: A three dimensional manifold formed by mapping from a two dimensional space to a three dimensional space.

Difficulty for Probabilistic Approaches

- Propagate a probability distribution through a non-linear mapping.
- Normalisation of distribution becomes intractable.

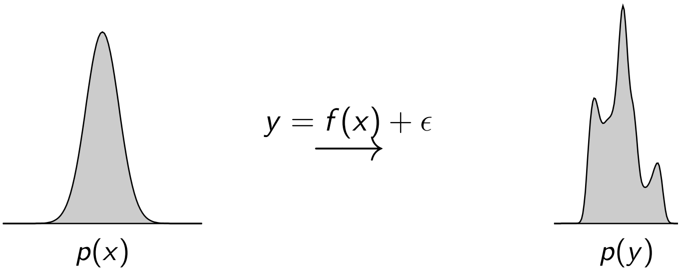


Figure: A Gaussian distribution propagated through a non-linear mapping. $y_i = f(x_i) + \epsilon_i$. $\epsilon \sim \mathcal{N}\left(0, 0.2^2\right)$ and $f(\cdot)$ uses RBF basis, 100 centres between -4 and 4 and $\ell = 0.1$. New distribution over y (right) is multimodal and difficult to normalize.

Mapping of Points

- Mapping points to higher dimensions is easy.

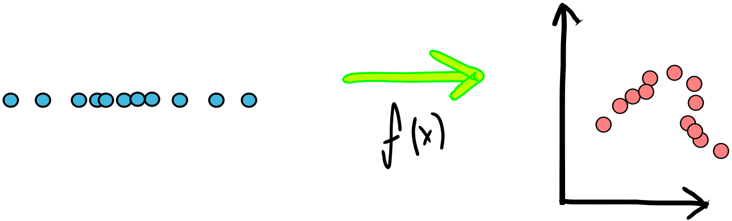


Figure: One dimensional Gaussian mapped to two dimensions.

Mapping of Points

- Mapping points to higher dimensions is easy.

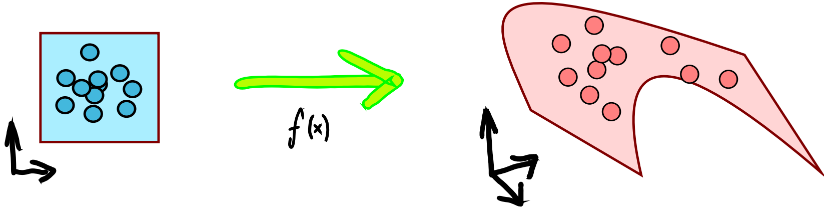


Figure: Two dimensional Gaussian mapped to three dimensions.

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

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

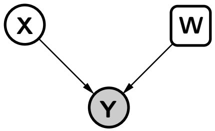
where

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

Linear Latent Variable Model

Probabilistic PCA

- Define *linear-Gaussian relationship* between latent variables and data.
- **Standard** Latent variable approach:

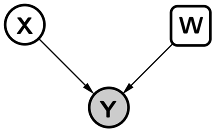


$$p\left(\mathbf{Y}|\mathbf{X},\mathbf{W}\right) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

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*, $\mathbf{X}$.

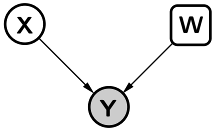


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right)$$

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*.



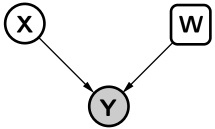
$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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

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*.



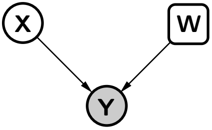
$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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

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

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln **(Tipping 99)**



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

Linear Latent Variable Model II

Probabilistic PCA Max. Likelihood Soln (Tipping 99)

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

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

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

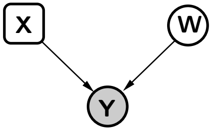
$$\mathbf{W} = \mathbf{U}_q\mathbf{L}\mathbf{R}^\top, \quad \mathbf{L} = \left(\mathbf{\Lambda}_q - \sigma^2\mathbf{I}\right)^{\frac{1}{2}}$$

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:

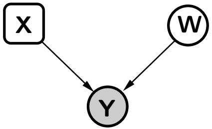


$$p(\mathbf{Y}|\mathbf{X},\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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*, $\mathbf{W}$.

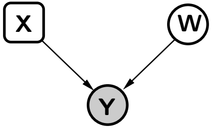


$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

Linear Latent Variable Model III

Dual Probabilistic PCA

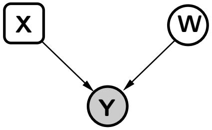
- Define *linear-Gaussian relationship* between latent variables and data.
- **Novel** Latent variable approach:
  - Define Gaussian prior over *parameters*, $\mathbf{W}$.
  - Integrate out *parameters*.



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

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

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*.

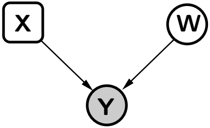
$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \sigma^2\mathbf{I}\right)$$

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

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

Linear Latent Variable Model IV

Dual Probabilistic PCA Max. Likelihood Soln (Lawrence 03, Lawrence 05)



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

Linear Latent Variable Model IV

Dual Probabilistic PCA Max. Likelihood Soln (Lawrence 03, Lawrence 05)

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

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

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

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

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

Linear Latent Variable Model IV

Probabilistic PCA Max. Likelihood Soln (Tipping 99)

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

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

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)

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

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

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

- Equivalence is from

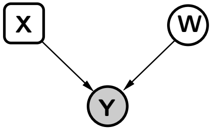
$$\mathbf{U}_q = \mathbf{Y}^\top \mathbf{U}'_q \boldsymbol{\Lambda}_q^{-\frac{1}{2}}$$

- You have probably used this trick to compute PCA efficiently when number of dimensions is much higher than number of points.

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*, **W**.
  - Integrate out *parameters*.



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

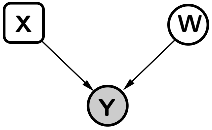
$$p(\mathbf{W}) = \prod_{i=1}^{p} \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0},\mathbf{I}\right)$$

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

Non-Linear Latent Variable Model

Dual Probabilistic PCA

- Inspection of the marginal likelihood shows ...
  - The covariance matrix is a covariance function.

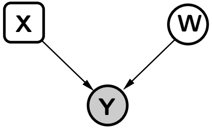


$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}\right)$$

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'.



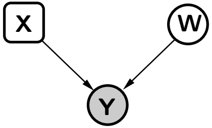
$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}\right)$$

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

Non-Linear Latent Variable Model

Dual Probabilistic PCA

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



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

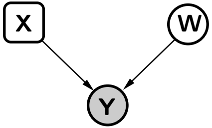
$$\mathbf{K} = \mathbf{X}\mathbf{X}^{\top} + \sigma^2\mathbf{I}$$

This is a product of Gaussian processes with linear kernels.

Non-Linear Latent Variable Model

Dual Probabilistic PCA

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



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

$$\mathbf{K} = ?$$

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

Non-linear Latent Variable Models

Exponentiated Quadratic (EQ) Covariance

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

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

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

Non-linear Latent Variable Models

Exponentiated Quadratic (EQ) Covariance

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

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

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

Stick Man

Generalization with 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*.

Stick Man

Generalization with 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*.
- Example: Modelling a stick man in 102 dimensions with 55 data points!

Stick Man

Generalization with 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*.
- Example: Modelling a stick man in 102 dimensions with 55 data points!

Stick Man II

demStick1

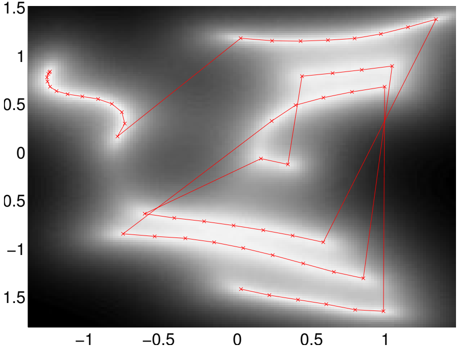
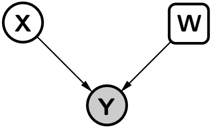


Figure: The latent space for the stick man motion capture data.

Let's look at some applications and extensions of the GPLVM

Maximum Likelihood Solution

Probabilistic PCA Max. Likelihood Soln (Tipping 99)

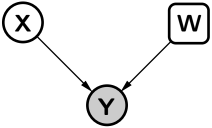


$$p(\mathbf{Y}|\mathbf{W},\boldsymbol{\mu}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\boldsymbol{\mu}, \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}\right)$$

Gradient of log likelihood

Maximum Likelihood Solution

Probabilistic PCA Max. Likelihood Soln (Tipping 99)



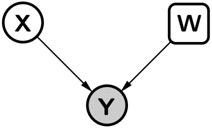
$$p\left(\hat{\mathbf{Y}}|\mathbf{W}\right) = \prod_{i=1}^{n} \mathcal{N}\left(\hat{\mathbf{y}}_{i,:}|\mathbf{0}, \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}\right)$$

Gradient of log likelihood

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

Maximum Likelihood Solution

Probabilistic PCA Max. Likelihood Soln (Tipping 99)

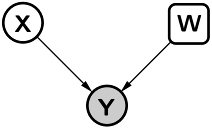


$$p\left(\hat{\mathbf{Y}}|\mathbf{W}\right) = \prod_{i=1}^{n} \mathcal{N}\left(\hat{\mathbf{y}}_{i,:}|\mathbf{0}, \mathbf{C}\right), \quad \mathbf{C} = \mathbf{W}\mathbf{W}^{\top} + \sigma^2\mathbf{I}$$

Gradient of log likelihood

Maximum Likelihood Solution

Probabilistic PCA Max. Likelihood Soln (Tipping 99)



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

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

Gradient of log likelihood

Optimization

Seek fixed points

$$\mathbf{0} = -\frac{n}{2}\mathbf{C}^{-1}\mathbf{W} + \frac{1}{2}\mathbf{C}^{-1}\hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}\mathbf{C}^{-1}\mathbf{W}$$

pre-multiply by $2\mathbf{C}$

$$\mathbf{0} = -n\mathbf{W} + \hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}\mathbf{C}^{-1}\mathbf{W}$$

$$\frac{1}{n}\hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}\mathbf{C}^{-1}\mathbf{W} = \mathbf{W}$$

Substitute W with singular value decomposition

$$\mathbf{W} = \mathbf{U}\mathbf{L}\mathbf{R}^\top$$

which implies

$$\begin{aligned}\mathbf{C} &= \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}\\ &= \mathbf{U}\mathbf{L}^2\mathbf{U}^\top + \sigma^2\mathbf{I}\end{aligned}$$

Using matrix inversion lemma

$$\mathbf{C}^{-1}\mathbf{W} = \mathbf{U}\mathbf{L}\left(\sigma^2 + \mathbf{L}^2\right)^{-1}\mathbf{R}^\top$$

Solution given by

$$\frac{1}{n}\hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}\mathbf{U} = \mathbf{U}\left(\sigma^2 + \mathbf{L}^2\right)$$

which is recognised as an eigenvalue problem.

- This implies that the columns of $\mathbf{U}$ are the eigenvectors of $\frac{1}{n}\hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}$ and that $\sigma^2 + \mathbf{L}^2$ are the eigenvalues of $\frac{1}{n}\hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}$.
- $l_i = \sqrt{\lambda_i - \sigma^2}$ where λ_i is the ith eigenvalue of $\frac{1}{n}\hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}$.
- Further manipulation shows that if we constrain $\mathbf{W} \in \Re^{p\times q}$ then the solution is given by the largest q eigenvalues.

Probabilistic PCA Solution

- If $\mathbf{U}_q$ are first q principal eigenvectors of $n^{-1}\hat{\mathbf{Y}}^\top\hat{\mathbf{Y}}$ 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.
- Some further work shows that the *principal* eigenvectors need to be retained.
- The maximum likelihood value for σ^2 is given by the average of the discarded eigenvalues.

Return