Machine Learning

Dimensionality Reduction

Dariush Hosseini

dariush.hosseini@ucl.ac.uk
Department of Computer Science
University College London

Lecture Overview

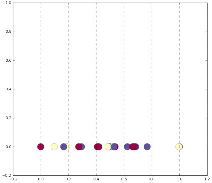
Lecture Overview

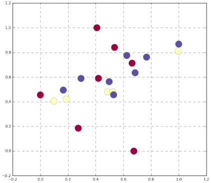
By the end of this lecture you should:

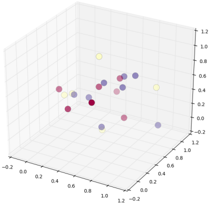
1. Understand the problem of the **curse of dimensionality** and further motivations for **dimensionality reduction**
2. Know how **Principal Components Analysis** can be used to project data onto a lower-dimensional **subspace**
3. Be familiar with **manifold learning** as a means to uncover the non-linear low-dimensional structure in high dimensional data

Lecture Overview

The Curse of Dimensionality







- As the dimensionality of our input space increases, the number of instances that we need to ‘fill’ that space increases

Specific Motivations

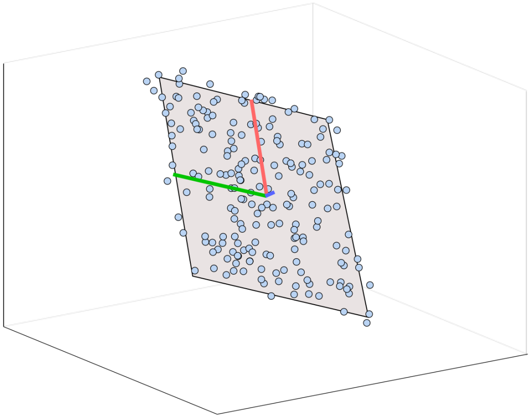
- **Data Visualisation**
- **Data Compression**
- **Denoising**

Dimensionality Reduction

- At the heart of dimensionality reduction techniques is the idea that, although our data is high-dimensional, it actually *lies* on or near a low-dimensional **subspace** or **manifold**
- If we assume our data lies on a subspace, we can use **linear** techniques to obtain a low-dimensional estimation
- If we assume our data lies on a manifold, then we will need to use **non-linear techniques**

Lecture Overview

Linear Subspaces



The data shown is $\mathbf{x} \in \mathbb{R}^3$

Along with this data is drawn a hyperplane that passes through both the origin and the data. The hyperplane is referred to as a **subspace** of $\mathbb{R}^3$

Linear Subspaces

- The hyperplane is a subset of $\mathbb{R}^3$:
  - Each point on the plane is represented by a single point in $\mathbb{R}^3$. But...
  - ...We only require $\mathbb{R}^2$ co-ordinates to describe a single point on the hyperplane
- So if we have a subspace $\mathbb{R}^d$ within $\mathbb{R}^m$, where $d \ll m$, then we can reduce the dimensionality of our data

Principal Components Analysis

- **Principal Component Analysis** (**PCA**), is one of the oldest methods for linear dimensionality reduction
- One view of PCA is that of **Projected Variance Maximisation** (other, equivalent views exist):
    - Here the idea is to project our high-dimensional data onto a lower-dimensional subspace (defined up to a rotation)**...**
    - **...**Such that the **sample variance** is maximally preserved
    - This subspace encapsulates the directions along which the data varies the most

PCA: Setting

- As usual, we assume that our input data consists of n instances: $\{\mathbf{x}^{(i)}\}_{i=1}^{n}$ where: $\mathbf{x}^{(i)} \in \mathbb{R}^m$

- Our goal is to project this data (linearly) onto a space $\mathbb{R}^d$, where $d < m$, such that the variance of projected data is maximised

- This space is spanned by the set of basis vectors $\{\mathbf{u}^{[i]}\}_{i=1}^{d}$ where: $\mathbf{u}^{[i]} \in \mathbb{R}^m$

- Since this will not fully define the space in which we are interested, we remove some degeneracy (and ease our mathematical analysis) by requiring **orthonormality**:

$$\mathbf{u}^{[i]} \cdot \mathbf{u}^{[j]} = \delta_{ij}$$

PCA: Setting

- For each basis vector $\mathbf{u}^{[j]}$, each data point $\mathbf{x}^{(i)}$ is then **projected** onto a scalar value $\mathbf{u}^{[j]} \cdot \mathbf{x}^{(i)}$
- The mean of the projected data onto this basis vector is $\mathbf{u}^{[j]} \cdot \overline{\mathbf{x}}$, where $\overline{\mathbf{x}}$ is the sample mean:

$$\overline{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}^{(i)}$$

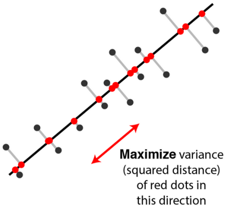
Projected Variance Maximisation

- The sample variance of the projected data is given by:

$$\frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{u}^{[j]}\cdot\mathbf{x}^{(i)}-\mathbf{u}^{[j]}\cdot\overline{\mathbf{x}}\right)^2=\mathbf{u}^{[j]\,T}\mathbf{S}\mathbf{u}^{[j]}$$

- Where S is the **sample covariance matrix** defined by:

$$\mathbf{S}=\frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}^{(i)}-\overline{\mathbf{x}}\right)\left(\mathbf{x}^{(i)}-\overline{\mathbf{x}}\right)^T=\frac{1}{n}\mathbf{X}^T\mathbf{X}$$



Projected Variance Maximisation

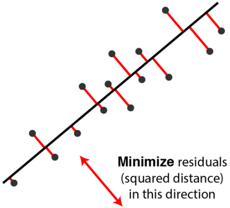
- Here we note that $\mathbf{X}$ is a **centred design matrix**:

$$\mathbf{X} = \begin{bmatrix} (\mathbf{x}^{(1)} - \overline{\mathbf{x}})^T \\ (\mathbf{x}^{(2)} - \overline{\mathbf{x}})^T \\ \cdot \\ \cdot \\ (\mathbf{x}^{(n)} - \overline{\mathbf{x}})^T \end{bmatrix}$$

Aside: Reconstruction Error Minimisation

- We can also view PCA as a search for the d dimensional subspace which minimises the **reconstruction error**:

$$\sum_{i=1}^{n} \left\| (\mathbf{x}^{(i)} - \overline{\mathbf{x}}) - \sum_{j=1}^{d} \left(\mathbf{u}^{[j]} \cdot (\mathbf{x}^{(i)} - \overline{\mathbf{x}}) \right) \mathbf{u}^{[j]} \right\|_2^2$$



PCA: Problem Formulation

- We are interested in finding $\left\{\mathbf{u}^{[j]}\right\}_{j=1}^{d}$ such that the sum of the variance of the projected sample data is maximised
- In other words we wish to solve the following optimisation problem:

$$\text{argmax}_{\left\{\mathbf{u}^{[j]}\right\}_{j=1}^{d}} \mathsf{L} \qquad \text{where:} \qquad \mathsf{L} = \sum_{j=1}^{d} \mathbf{u}^{[j]T}\mathbf{S}\mathbf{u}^{[j]} \tag{1}$$

Subject to orthonormal $\left\{\mathbf{u}^{[j]}\right\}_{j=1}^{d}$

Eigendecomposition

- Consider a square symmetric matrix, $\mathbf{S}$, with **eigenvalues** $\{\lambda_i\}_{i=1}^m$, and associated (orthonormal) **eigenvectors** $\{\mathbf{q}_i \in \mathbb{R}^m\}_{i=1}^m$:

$$\mathbf{S}\mathbf{q}_i = \lambda_i \mathbf{q}_i \qquad \text{where:} \qquad \mathbf{q}_i \cdot \mathbf{q}_j = \delta_{ij}$$

- Then the **eigendecomposition** of $\mathbf{S}$ is given by:

$$\mathbf{S} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^T$$

where:

$$\boldsymbol{\Lambda} = \text{diag}(\lambda_1, ..., \lambda_m)$$
$$\mathbf{Q} = [\mathbf{q}_1, ..., \mathbf{q}_m]$$

PCA: Greedy Solution

- Let us proceed in a step-wise fashion and attempt to learn each dimension **greedily**

- First, let us search for the direction of highest variance, and write our objective as the following Lagrangian, $\mathcal{L}$:

$$\mathcal{L}(\mathbf{u}^{[1]}, \lambda^{[1]}) = \mathbf{u}^{[1]T}\mathbf{S}\mathbf{u}^{[1]} - \lambda^{[1]}(\mathbf{u}^{[1]} \cdot \mathbf{u}^{[1]} - 1)$$

  Where $\lambda^{[1]}$ is a Lagrange multiplier associated with the orthogonality constraint.

- Seeking stationarity wrt $\mathbf{u}^{[1]}$ gives:

$$2\mathbf{S}\mathbf{u}^{[1]} - 2\lambda^{[1]}\mathbf{u}^{[1]} = 0$$
$$\implies \mathbf{S}\mathbf{u}^{[1]} = \lambda^{[1]}\mathbf{u}^{[1]}$$

PCA: First Basis vector

- Since $\mathbf{u}^{[1]}$ satisfies the eigenvalue equation this lets us equate it with some eigenvalue $\mathbf{q}_i$
- But which eigenvalue? Let us calculate the variance of the projected data:

$$\mathbf{u}^{[1]T}\mathbf{S}\mathbf{u}^{[1]} = \lambda^{[1]}$$

- We want to maximise this variance so we select $\mathbf{u}^{[1]} = \mathbf{q}_1$, the eigenvector associated with λ_1, the largest eigenvalue. Thus:

$$\mathbf{u}^{[1]} = \mathbf{q}_1$$
$$\lambda^{[1]} = \lambda_1$$

PCA: Second Basis Vector

- Now let us find another direction, $\mathbf{u}^{[2]}$, to further increase the projected variance, such that $\mathbf{u}^{[2]} \cdot \mathbf{u}^{[2]} = 1$ and $\mathbf{u}^{[1]} \cdot \mathbf{u}^{[2]} = 0$:

- We write a Lagrangian, $\mathcal{L}$, for this problem as follows:

$$\mathcal{L}(\mathbf{u}^{[2]}, \lambda^{[2]}, \lambda^{[1][2]}) = \mathbf{u}^{[2]T}\mathbf{S}\mathbf{u}^{[2]} - \lambda^{[2]}(\mathbf{u}^{[2]} \cdot \mathbf{u}^{[2]} - 1) - \lambda^{[1][2]}(\mathbf{u}^{[2]} \cdot \mathbf{u}^{[1]} - 0)$$

  Where $\lambda^{[2]}$ and $\lambda^{[1][2]}$ are Lagrange multipliers.

- Seeking stationarity wrt $\mathbf{u}^{[2]}$ gives:

$$2\mathbf{S}\mathbf{u}^{[2]} - 2\lambda^{[2]}\mathbf{u}^{[2]} - \lambda^{[1][2]}\mathbf{u}^{[1]} = 0 \qquad (2)$$

PCA: Second Basis Vector

- Left multiply equation (2) by $\mathbf{u}^{[1]T}$ gives:

$$
\begin{aligned}
\implies 2\mathbf{u}^{[1]T}\mathbf{S}\mathbf{u}^{[2]} - 2\lambda^{[2]}\mathbf{u}^{[1]}\cdot\mathbf{u}^{[2]} - \lambda^{[1][2]}\mathbf{u}^{[1]}\cdot\mathbf{u}^{[1]} &= 0 \\
\implies 2\mathbf{u}^{[2]T}\mathbf{S}\mathbf{u}^{[1]} - \lambda^{[1][2]} &= 0 \\
\implies 2\lambda^{[1]}\mathbf{u}^{[2]}\cdot\mathbf{u}^{[1]} - \lambda^{[1][2]} &= 0 \\
\implies \lambda^{[1][2]} &= 0
\end{aligned}
$$

- Therefore:

$$\mathbf{S}\mathbf{u}^{[2]} = \lambda^{[2]}\mathbf{u}^{[2]}$$

PCA: Second Basis vector

- Since $\mathbf{u}^{[2]}$ satisfies the eigenvalue equation this lets us equate it with some eigenvalue $\mathbf{q}_i$
- But which eigenvalue? Let us calculate the variance of the projected data:

$$\mathbf{u}^{[2]T}\mathbf{S}\mathbf{u}^{[2]} = \lambda^{[2]}$$

- We want to maximise this variance so we select $\mathbf{u}^{[2]} = \mathbf{q}_2$, the eigenvector associated with λ_2, the largest remaining eigenvalue. Thus:

$$\mathbf{u}^{[2]} = \mathbf{q}_2$$
$$\lambda^{[2]} = \lambda_2$$

PCA: Subsequent Basis Vectors

- The solution proceeds in a similar step-wise fashion, to give:

$$\left\{\mathbf{u}^{[j]*} = \mathbf{q}_j\right\}_{j=1}^{d}$$

- And this implies a total projected variance of

$$\mathsf{L}^* = \sum_{j=1}^{d} \mathbf{q}_j^T \mathbf{S} \mathbf{q}_j = \sum_{j=1}^{d} \lambda_j$$

PCA: Non-Uniqueness

- Note that we have made a number of choices in this derivation...
  - Orthogonality of basis vectors
  - Normality of different basis vectors
  - Orientation of the first basis vector
- Other choices are possible which would also yield similar solutions
- This suggests that we should check if a particular similar solution is globally optimal

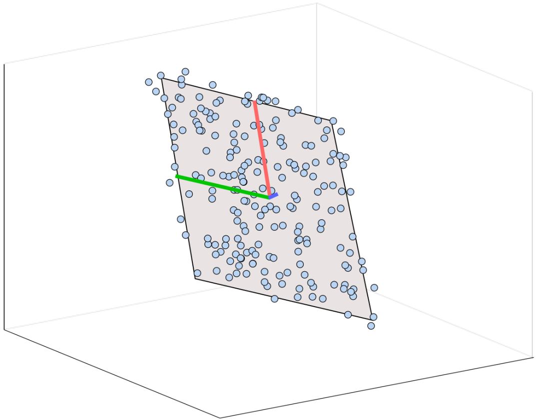
PCA: Non-Convexity

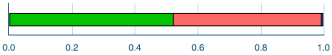
- The trouble is that our original problem is **non-convex**
- However we can show that the greedy local optima we have investigated give rise to globally optimal points
- It turns out that PCA is a rare example of a non-convex optimisation problem which we can solve globally!

PCA: Examining the Eigenvalues

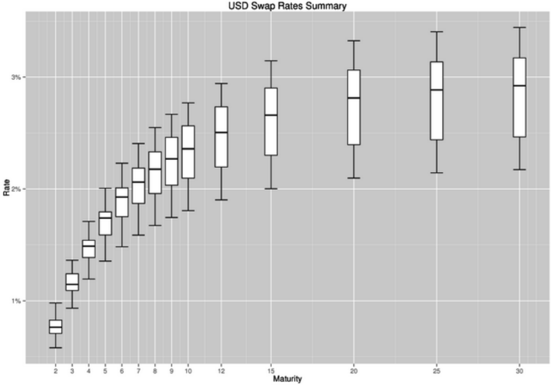
- Our PCA solution allows us to estimate the subspace as the space spanned by the first d eigenvectors of the sample covariance matrix $\mathbf{S}$ ordered by size of eigenvalue
- If we refer to the dimensionality of the input space as the **ambient dimensionality** of the data, then the **intrinsic dimensionality** is the dimensionality of the subspace upon which the data is assumed to lie

Examining the Eigenvalues

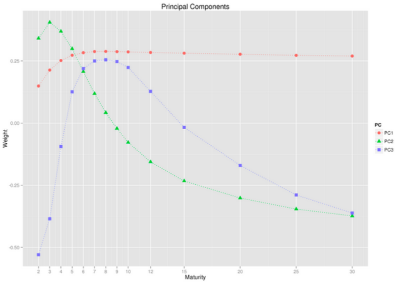


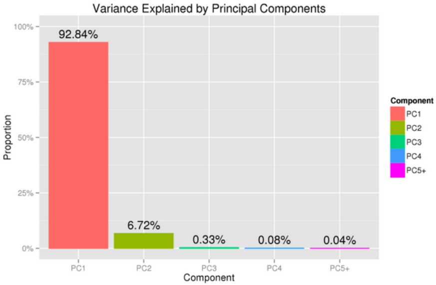


Application: Swaps Curve



Application: Swaps Curve





- ‘Shift’, ‘Tilt’, ‘Curvature’ effects dominate

Pattern Stability

- PCA is intuitively plausible, but can we be sure that our training sample has allowed us to discern a reliable subspace?

- At least two possible responses to this, which flow from different ML paradigms:

Pattern Stability

- **Generative Modelling:**
  - *Probabilistic PCA* - Seeks a Gaussian latent variable model
  - Learn parameters using MLE or Bayesian treatment

- **PAC Approach:**
  - Seeks to bound generalisation error resulting from projection: $\mathbb{E}_{\mathcal{D}}\left[\mathbf{x} - \sum_{j=1}^{d}(\mathbf{u}^{[j]} \cdot \mathbf{x})\mathbf{u}^{[j]}\right]^2$, with high probability
  - Bound contains terms in sample residual eigenspectrum and in complexity of data space
  - Indicates low generalisation error if subspace captures a high proportion of the variance in a dimensionality small compared to training data size

Lecture Overview

Probabilistic PCA

- Let's investigate the probabilistic generalisation of PCA
- Here it's valuable to recall the probabilistic generalisation of the *k*-means model to the Mixture of Gaussians model for clustering
- Can we build a Latent Variable version of PCA?

UCL

PPCA: Setting

- As before, our input data consists of n instances, $\{\mathbf{x}^{(i)} \in \mathbb{R}^m\}_{i=1}^n$, with sample mean $\bar{\mathbf{x}}$, and associated sample covariance matrix, $\mathbf{S} = \frac{1}{n}\mathbf{X}^T\mathbf{X}$
- And as before, we seek some d-dimensional principal component subspace, where $d < m$

PPCA: Model

- Each point has an unknown, latent, variable, $\mathbf{z} \in \mathbb{R}^d$ associated with it, corresponding to its position in the principal component subspace. This variable is the outcome of a Gaussian random variable, $\mathcal{Z}$, such that:

$$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$$

- Contingent on the principal component subspace variable, each $\mathbf{x}$ is the outcome of a Gaussian random variable, $\mathcal{X}$, such that:

$$\mathbf{x}|\mathbf{z} \sim \mathcal{N}(\mathbf{Wz} + \mu, \sigma^2 \mathbf{I}_m)$$

where: $\mathbf{W} \in \mathbb{R}^{m \times d}$, which defines the directions of the principal subspace, $\mu \in \mathbb{R}^m$, $\sigma \in \mathbb{R}^+$

PPCA: Model

- We can view the model from a generative standpoint:
  - First a value is drawn for the latent variable, $\mathbf{z}$
  - Then the observed variable is sampled, conditional on this latent variable, $\mathbf{x}|\mathbf{z}$
  - And residual noise is captured by ε, which is an outcome of an m-dimensional random variable, ϵ, distributed like a zero-mean Gaussian:

$$\begin{aligned} \mathbf{x} &= \mathbf{W}\mathbf{z} + \mu + \varepsilon \\ \mathbf{z} &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d) \\ \varepsilon &\sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_m) \end{aligned} \tag{3}$$

where $\mathfrak{Z}$ and ϵ are uncorrelated

PPCA: Model

- Recall the properties of the **marginal & conditional distributions** associated with **Linear Gaussian Models** that we encountered in the *Probability Lecture*:
  - Given a marginal distribution for $\widetilde{\mathbf{x}}$ and a conditional Gaussian distribution for $\widetilde{\mathbf{y}}$ given $\widetilde{\mathbf{x}}$:

$$\widetilde{\mathbf{x}} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda}^{-1})$$
$$\widetilde{\mathbf{y}}|\widetilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{A}\widetilde{\mathbf{x}} + \mathbf{b}, \mathbf{L}^{-1})$$

Where: $\widetilde{\mathbf{x}} \in \mathbb{R}^n$, $\widetilde{\mathbf{y}} \in \mathbb{R}^m$, $\boldsymbol{\mu} \in \mathbb{R}^n$, $\boldsymbol{\Lambda} \in \mathbb{R}^{n\times n}$, $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{L} \in \mathbb{R}^{m\times m}$
Then:

$$\widetilde{\mathbf{y}} \sim \mathcal{N}(\mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{L}^{-1} + \mathbf{A}\boldsymbol{\Lambda}^{-1}\mathbf{A}^T)$$
$$\widetilde{\mathbf{x}}|\widetilde{\mathbf{y}} \sim \mathcal{N}(\boldsymbol{\Sigma}[\mathbf{A}^T\mathbf{L}(\widetilde{\mathbf{y}} - \mathbf{b}) + \boldsymbol{\Lambda}\boldsymbol{\mu}], \boldsymbol{\Sigma})$$

Where: $\boldsymbol{\Sigma} = (\boldsymbol{\Lambda} + \mathbf{A}^T\mathbf{L}\mathbf{A})^{-1}$

PPCA: Model

- From this, we can see that **x** is drawn from a Gaussian distribution with:
  - **Mean:**

$$\begin{aligned}\mathbb{E}[\mathcal{X}] &= \mathbf{W}\mathbb{E}[\mathcal{Z}] + \boldsymbol{\mu} + \mathbb{E}[\boldsymbol{\epsilon}] \\ &= \boldsymbol{\mu}\end{aligned}$$

  - **Covariance:**

$$\begin{aligned}\mathbf{C} &= \mathbb{E}[(\mathbf{W}\mathcal{Z} + \boldsymbol{\mu} + \boldsymbol{\epsilon} - \mathbb{E}[\mathcal{X}])(\mathbf{W}\mathcal{Z} + \boldsymbol{\mu} + \boldsymbol{\epsilon} - \mathbb{E}[\mathcal{X}])^T] \\ &= \mathbb{E}[(\mathbf{W}\mathcal{Z} + \boldsymbol{\epsilon})(\mathbf{W}\mathcal{Z} + \boldsymbol{\epsilon})^T] \\ &= \mathbb{E}[\mathbf{W}\mathcal{Z}\mathcal{Z}^T\mathbf{W}^T] + \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] + \mathbb{E}[\mathbf{W}\mathcal{Z}\boldsymbol{\epsilon}^T] + \mathbb{E}[\boldsymbol{\epsilon}\mathcal{Z}^T\mathbf{W}^T] \\ &= \mathbf{W}\mathbb{E}[\mathcal{Z}\mathcal{Z}^T]\mathbf{W}^T + \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] \\ &= \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}_m\end{aligned}$$

- Thus:

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$$

PPCA: Interpretation

- We can interpret $p_{\mathcal{X}}(\mathbf{x})$ as a density defined by taking an isotropic Gaussian 'spray can'...
- ...Then moving across the principal subspace, spraying Gaussian ink with density determined by σ^2...
- ...And weighted by the prior distribution, $p_{\mathcal{Z}}(\mathbf{z})$
- This results in an ink density which has a pancake shaped distribution, which represents $p_{\mathcal{X}}(\mathbf{x})$

PPCA: Interpretation

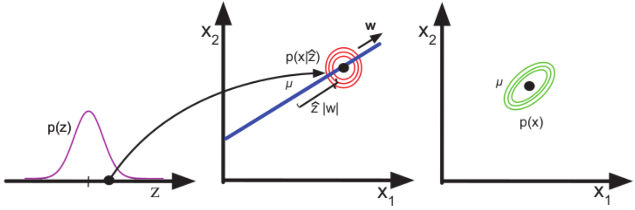


Figure 12.1 Illustration of the PPCA generative process, where we have $L = 1$ latent dimension generating $D = 2$ observed dimensions. Based on Figure 12.9 of (Bishop 2006b).

UCL

PPCA: Rotational Invariance

- Let $\mathbf{R}$ be an orthogonal (rotation) matrix (for which $\mathbf{R}^T\mathbf{R} = \mathbf{I}$)

- Now apply this rotation to the latent space coordinate matrix, $\mathbf{W}$:

$$\begin{aligned} \widetilde{\mathbf{W}} &= \mathbf{W}\mathbf{R} \\ \implies \widetilde{\mathbf{W}}\widetilde{\mathbf{W}}^T &= \mathbf{W}\mathbf{R}\mathbf{R}^T\mathbf{W}^T \\ &= \mathbf{W}\mathbf{W}^T \end{aligned}$$

- Thus $p_{\mathcal{X}}(\mathbf{x})$ is as well characterised by any $\widetilde{\mathbf{W}}$ as it is by $\mathbf{W}$

- This is the analogue of the non-uniqueness which we encountered in PCA

PPCA: Learning Problem

- Now that we have defined our generative model we need to learn its parameters: $\boldsymbol{\mu}$, $\mathbf{W}$, σ^2
- Let’s use a Maximum Likelihood approach:

PPCA: Log Likelihood

$$\begin{aligned}
\ln \mathbb{P}\left(\{\mathbf{x}^{(i)}\}_{i=1}^{n}\right) &= \sum_{i=1}^{n} \ln p_{\mathcal{X}}(\mathbf{x}^{(i)}; \mathbf{W}, \boldsymbol{\mu}, \sigma^2) \\
&= -\frac{nm}{2}\ln(2\pi) - \frac{n}{2}\ln|\mathbf{C}| - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}^{(i)} - \boldsymbol{\mu})^T \mathbf{C}^{-1}(\mathbf{x}^{(i)} - \boldsymbol{\mu})
\end{aligned}$$

PPCA: Optimisation

- Tipping & Bishop ('99) demonstrate the following closed form solutions which flow from the optimisation of this function:
  - $$\mu_{\mathrm{MLE}} = \bar{\mathbf{x}}$$
  - $$\mathbf{W}_{\mathrm{MLE}} = \mathbf{Q}(\Lambda - \sigma^2 \mathbf{I})^{1/2}\mathbf{R}$$

    Here:
    $\mathbf{Q}$ is the $m \times d$ matrix whose columns are given by the leading d eigenvectors of the covariance matrix $\mathbf{S}$
    Λ is the diagonal matrix of the d leading eigenvalues associated with these eigenvectors

    $\mathbf{R}$ is an arbitrary orthogonal matrix
  - $$\sigma^2_{\mathrm{MLE}} = \frac{1}{m-d} \sum_{i=d+1}^{m} \lambda_i$$

PPCA: Interpretation of Solution

- We can set $\mathbf{R} = \mathbf{I}$ without loss of generality, in which case the columns of $\mathbf{W}$ are the principal component eigenvectors scaled by the square root of the variance parameters $(\lambda_i - \sigma^2)^{1/2}$

- So the variance of $p_{\mathcal{X}}(\mathbf{x})$ in the $\mathbf{q}_i$ direction is given by:

$$\begin{aligned} \mathbf{q}_i^T \mathbf{C} \mathbf{q}_i &= \mathbf{q}_i^T \mathbf{W} \mathbf{W}^T \mathbf{q}_i + \sigma^2 \mathbf{q}_i^T \mathbf{q}_i \\ &= \lambda_i - \sigma^2 + \sigma^2 \\ &= \lambda_i \end{aligned}$$

- So this model captures the variance of the data in the direction of the principal axes

- While σ^2_{MLE} is the average variance associated with the discarded dimensions

PPCA: Connection with PCA

- Recall the distributional forms for $\mathbf{z}$, and $\mathbf{x}|\mathbf{z}$:

$$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$$
$$\mathbf{x}|\mathbf{z} \sim \mathcal{N}(\mathbf{W}\mathbf{z} + \mu, \sigma^2\mathbf{I}_n)$$

- Then using the conditional distribution property for Linear Gaussian Models once again:

$$\mathbf{z}|\mathbf{x} \sim \mathcal{N}(\mathbf{M}^{-1}\mathbf{W}_{\mathrm{MLE}}^T(\mathbf{x} - \mu), \sigma^2\mathbf{M}^{-1})$$

  Where: $\mathbf{M} = \mathbf{W}_{\mathrm{MLE}}^T\mathbf{W}_{\mathrm{MLE}} + \sigma^2\mathbf{I}$

- This is the particular distribution of $\mathbf{z}$ given a point $\mathbf{x}$ in the input data space.

PPCA: Connection with PCA

- The expectation of $\mathbb{Z}|\mathbf{x}$ gives a summary of the point $\mathbf{x}$ in latent space:

$$\mathbb{E}[\mathbb{Z}|\mathbf{x}] = \mathbf{M}^{-1}\mathbf{W}_{\text{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu})$$

PPCA: Connection with PCA

- And this latent variable point projects back to a point in the input data space the expectation of which is given by:

$$\mathbf{W}_{\mathrm{MLE}}\mathbb{E}[\mathbb{Z}|\mathbf{x}] + \boldsymbol{\mu}$$
$$= \mathbf{W}_{\mathrm{MLE}}\mathbf{M}^{-1}\mathbf{W}_{\mathrm{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu}$$

PPCA: Connection with PCA

- And this latent variable point projects back to a point in the input data space the expectation of which is given by:

$$
\begin{aligned}
&\mathbf{W}_{\mathrm{MLE}}\mathbb{E}[\mathbb{Z}|\mathbf{x}] + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\mathrm{MLE}}\mathbf{M}^{-1}\mathbf{W}_{\mathrm{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\mathrm{MLE}}\left(\mathbf{W}_{\mathrm{MLE}}^{T}\mathbf{W}_{\mathrm{MLE}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\mathrm{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu}
\end{aligned}
$$

PPCA: Connection with PCA

- And this latent variable point projects back to a point in the input data space the expectation of which is given by:

$$
\begin{aligned}
&\mathbf{W}_{\text{MLE}}\mathbb{E}[\mathbb{Z}|\mathbf{x}] + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\mathbf{M}^{-1}\mathbf{W}_{\text{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\left(\mathbf{W}_{\text{MLE}}^{T}\mathbf{W}_{\text{MLE}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\text{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\left((\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}}\mathbf{Q}^{T}\mathbf{Q}(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\text{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu}
\end{aligned}
$$

PPCA: Connection with PCA

- And this latent variable point projects back to a point in the input data space the expectation of which is given by:

$$
\begin{aligned}
&\mathbf{W}_{\mathrm{MLE}}\mathbb{E}[\mathbb{Z}|\mathbf{x}] + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\mathrm{MLE}}\mathbf{M}^{-1}\mathbf{W}_{\mathrm{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\mathrm{MLE}}\left(\mathbf{W}_{\mathrm{MLE}}^{T}\mathbf{W}_{\mathrm{MLE}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\mathrm{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\mathrm{MLE}}\left((\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}}\mathbf{Q}^{T}\mathbf{Q}(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\mathrm{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\mathrm{MLE}}\boldsymbol{\Lambda}^{-1}\mathbf{W}_{\mathrm{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu}
\end{aligned}
$$

PPCA: Connection with PCA

- And this latent variable point projects back to a point in the input data space the expectation of which is given by:

$$
\begin{aligned}
&\mathbf{W}_{\text{MLE}}\mathbb{E}[\mathbb{Z}|\mathbf{x}] + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\mathbf{M}^{-1}\mathbf{W}_{\text{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\left(\mathbf{W}_{\text{MLE}}^{T}\mathbf{W}_{\text{MLE}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\text{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\left((\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}}\mathbf{Q}^{T}\mathbf{Q}(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\text{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\boldsymbol{\Lambda}^{-1}\mathbf{W}_{\text{MLE}}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{Q}(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}}\boldsymbol{\Lambda}^{-1}(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}}\mathbf{Q}^{T}(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu}
\end{aligned}
$$

PPCA: Connection with PCA

- And this latent variable point projects back to a point in the input data space the expectation of which is given by:

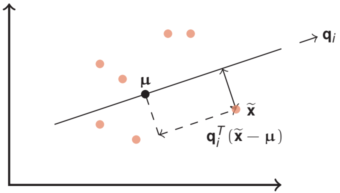
$$
\begin{aligned}
&\mathbf{W}_{\text{MLE}}\mathbb{E}[\mathbb{Z}|\mathbf{x}] + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\mathbf{M}^{-1}\mathbf{W}_{\text{MLE}}^T(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\left(\mathbf{W}_{\text{MLE}}^T\mathbf{W}_{\text{MLE}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\text{MLE}}^T(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\left((\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}}\mathbf{Q}^T\mathbf{Q}(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{W}_{\text{MLE}}^T(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{W}_{\text{MLE}}\boldsymbol{\Lambda}^{-1}\mathbf{W}_{\text{MLE}}^T(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{Q}(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}}\boldsymbol{\Lambda}^{-1}(\boldsymbol{\Lambda} - \sigma^2\mathbf{I})^{\frac{1}{2}}\mathbf{Q}^T(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu} \\
&\quad = \mathbf{Q}\quad \text{diag}\left(\frac{\lambda_1 - \sigma^2}{\lambda_1}, \ldots, \frac{\lambda_d - \sigma^2}{\lambda_d}\right)\mathbf{Q}^T(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu}
\end{aligned}
$$

PPCA: Connection with PCA

- Now, as $\sigma^2 \to 0$, then $(\mathbf{W}_{\text{MLE}}\mathbb{E}[\mathcal{Z}|\mathbf{x}] + \boldsymbol{\mu}) \to \left(\mathbf{Q}\mathbf{Q}^T(\mathbf{x} - \boldsymbol{\mu}) + \boldsymbol{\mu}\right)$

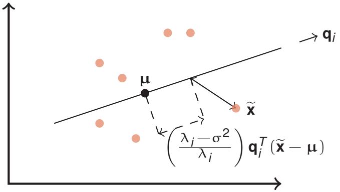
- Thus each data point is approximated by a mapping into a linear subspace defined by the eigenvectors of **S**, given by **Q**, such that each point is orthogonally projected into this subspace

- ...Just as in PCA:



PPCA: Connection with PCA

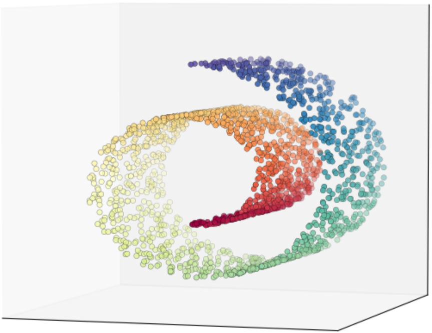
- But for $\sigma^2 > 0$, each projection is scaled by $\frac{\lambda_i - \sigma^2}{\lambda_i} < 1$, thus the projection is shrunk towards $\boldsymbol{\mu}$:

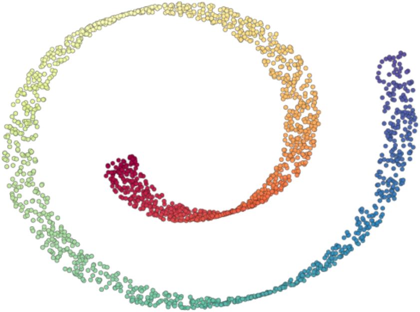


PPCA: Setting d

- Just as in the Mixture of Gaussians model for clustering, we can use our generative model to select d in a principled way:
- Evaluate the likelihood of data on a validation set for various settings of d and select the one which gives rise to the maximal one
- A Bayesian treatment of PPCA or the maximisation of the PAC bound offer different resolutions to this problem

Where PCA Fails

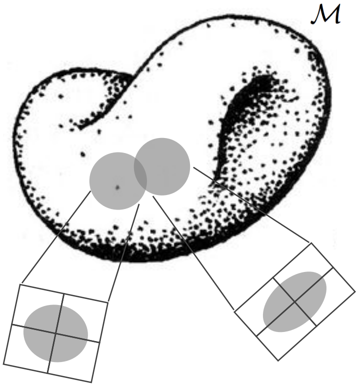




- One of the restrictions of PCA is that it assumes the data lies on or near a linear subspace
- What happens if that assumption fails?

Lecture Overview

From Subspaces to (Sub)Manifolds



Manifold Learning Techniques

- Kernel PCA
- Isomap
- Locally Linear Embedding (LLE)
- Autoencoder
- ...

Lecture Overview

Summary

1. **Dimensionality Reduction** falls into two broad categories depending on whether the low-dimensional space we are mapping to is linear or non-linear
2. **Principal Component Analysis** is a linear technique for reducing the dimensionality of the data by projecting it onto the maximum covariance subspace
3. Manifold learning techniques allow for the linear assumption inherent in PCA to be relaxed