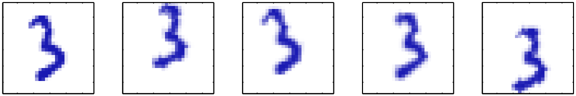
Chris Bishop's PRML
Ch. XII: Continuous Latent Variables

Caroline Bernard-Michel & Hervé Jegou

June 12, 2008

Introduction

- Aim of this chapter: dimensionality reduction
- Can be interesting for lossy data compression, feature extraction and data visualization.
- Example: synthetic data set
- Choice of one of the off-line digit images
- Creation of multiple copies with a random displacement and rotation
- Individuals= images ($28 \times 28 = 784$)
- Variables= pixels grey levels
- Only two latent variables: the translation and the rotation

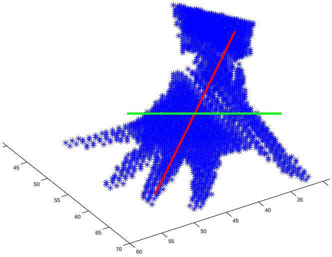


Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

Maximum variance formulation

- Consider a data set of observations $\{x_n\}$ where $n = 1, \ldots, N$ (x_n with dimensionality D).
- Idea of PCA: Project this data onto a space of lower dimensionality $M < D$, called *the principal subspace*, while maximizing the variance of the projected data



Notations

We will denote by:

- D the dimensionality
- M the fixed dimension of the principal subspace
- $\{u_i\}$, $i = 1, \ldots, M$ the basis vectors ($(D \times 1)$ vectors) of the principal subspace
- The sample mean ($(D \times 1)$ vector) by:

$$\bar{x} = \frac{1}{N} \sum_{n=1}^{N} x_n \tag{1.90}$$

- The sample variance/covariance matrix ($(D \times D)$ matrix) by:

$$S = \frac{1}{N} \sum_{n=1}^{N} (x_n - \bar{x})(x_n - \bar{x})^T \tag{1}$$

Idea of PCA with one-dimensional principal subspace

- Let us consider a unit D-dimensional normalized vector u_1 ($u_1^T u_1 = 1$)
- Each point x_n is then projected onto a scalar is $u_1^T x_n$
- The mean of the projected data is:

$$u_1^T \bar{x} \tag{2}$$

- The variance of the projected data is:

$$\frac{1}{N} \sum_{n=1}^{N} u_1^T x_n - u_1^T \bar{x}^2 = u_1^T S u_1 \tag{3}$$

Idea of PCA: Maximize the projected variance $u_1^T S u_1$ with respect to u_1 under the normalization constraint $u_1^T u_1 = 1$

Idea of PCA with one-dimensional principal subspace

- Trick: introduce the Lagrange multiplier λ_1
- Unconstrained maximization of $u_1^T S u_1 + \lambda_1(1 - u_1^T u_1)$
- Solution must verify:

$$S u_1 = \lambda_1 u_1 \tag{4}$$

- u_1 must be an eigenvector of S having eigenvalue λ_1!
- The variance of the projected data is λ_1 $(u_1^T S u_1 = \lambda_1)$, so λ_1 has to be the largest eigenvalue!
- Additional principal components are obtained maximizing the projected variance amongst all possible directions orthogonal to those already considered!
- PCA =calculating the eigenvectors of the data covariance matrix corresponding to the largest eigenvalues!
- Note: $\sum_{i=1}^{D} \lambda_i$ is generally called the total inerty or the total variance. The percentage of inerty explained by one component u_i is then $\frac{\lambda_i}{\sum_{i=1}^{D} \lambda_i}$.

Chapter content

Minimum-error formulation

- Based on projection error minimization
- Consider a D-dimensional basis vectors $\{u_i\}$ where $i = 1, \ldots, D$ satisfying $u_i^T u_j = \delta_{ij}$
- Each data point x_n can be represented by:

$$x_n = \sum_{i=1}^{D} \alpha_{ni} u_i \texttt{ where } \alpha_{ni} = x_n^T u_i \tag{5}$$

- x_n can be approximated by

$$\tilde{x}_n = \sum_{i=1}^{M} z_{ni} u_i + \sum_{i=M+1}^{D} b_i u_i \tag{6}$$

- Idea of PCA: Minimize the distortion J introduced by the reduction in dimensionality

$$J = \frac{1}{N} \sum_{n=1}^{N} \| x_n - \tilde{x}_n \|^2 \tag{7}$$

Minimum-error formulation (2)

- Setting the derivative with respect to z_{nj} and to b_j, one obtains that:

$$z_{nj} = x_n^T u_j \ \texttt{and} \ b_j = \bar{x}^T u_j \tag{8}$$

- J can then be expressed as:

$$J = \frac{1}{N} \sum_{i=M+1}^{D} u_i^T S u_i \tag{9}$$

- The minimum is obtained when $\{u_i\}$, $i = M+1, \ldots, D$ are the eigenvectors of S associated to the smallest eigenvalues.
- The distortion is then given by $J = \sum_{i=M+1}^{D} \lambda_i$
- x_n is approximated by:

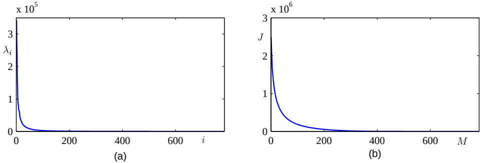
$$\tilde{x}_n = \sum_{i=1}^{M} (x_n^T u_i) u_i + \sum_{i=M+1}^{D} (\bar{x}^T u_i) u_i = \bar{x} + \sum_{i=1}^{M} (x_n^T - \bar{x}^T u_i) u_i \tag{10}$$

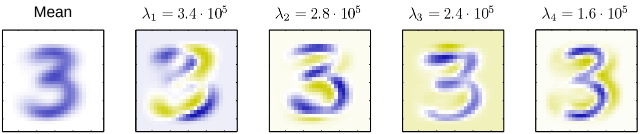
Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

Application of PCA: data compression

- Individuals = images
- Variables = grey levels of each pixel (784)



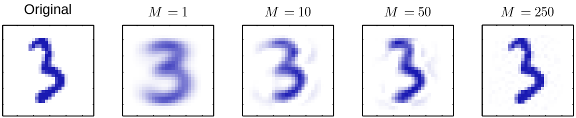


Application of PCA: data compression (2)

- Compression using the PCA approximation for x_n

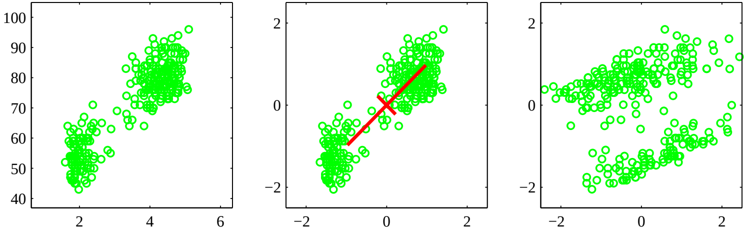
$$\tilde{x}_n = \bar{x} + \sum_{i=1}^{M} \{x_n u_i - \bar{x} u_i\} u_i \tag{11}$$

- For each data point we have replaced the D-dimensional vector x_n with an M-dimensional vector having components $(x_n^T u_i - \bar{x}^T u_i)$



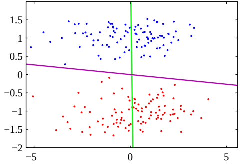
Application of PCA: data pre-processing

- Usually, individual standardization of each variable: each variable has zero mean and unit variance. Variables still correlated.
- Use of PCA for standardization:
  - writing the eigenvector equation $SU = UL$ where L is a $D \times D$ diagonal matrix with element λ_i and U is a $D \times D$ orthogonal matrix with columns given by u_i
  - And defining by: $y_n = L^{-1/2}U^T(x_n - \bar{x})$
  - y_n has zero mean and identity covariance matrix (new variables are decorrelated)

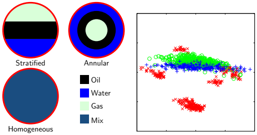


Application of PCA: data pre-processing

- Comparison: PCA chooses the direction of maximum variance whereas the Fisher's linear discriminant takes account of the class labels (see Chap. 4).



- Vizualization: projection of the oil data flow onto the first two principal factors. Three geometrical configurations of the oil, water and gas phases.



Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

PCA for high-dimensional data

- Number of data points N is smaller than the dimensionality D
- At least $D - N + 1$ of the eigenvalues equal to zero!
- Generally computationally infeasible.

- Let us denote X the $(N \times D)$-dimensional centred matrix.
- The covariance matrix can be writen as $S = N^{-1}X^TX$
- It can be shown that S has $D - N + 1$ eigenvalues of value zero and $N - 1$ eigenvalues as XX^T
- If we denote the eigenvectors of XX^T by v_i, the normalized eigenvectors u_i for S can be deduced by:

$$u_i = \frac{1}{(N\lambda_i)^{\frac{1}{2}}} X^T v_i \qquad (12)$$

Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

Probabilistic PCA

Advantages:

- Derive an EM algorithm for PCA that is computationally efficient
- Allows to deal with missing values in the dataset
- Mixture of probabilistic PCA models
- Basis for the Bayesian treatment of PCA in which the dimensionality of the principal subspace can be found automatically.
- ...

Probabilistic PCA (2)

Related to factor analysis:

- A latent variable model seeks to relate a D-dimensional observation vector x to a corresponding M-dimensional Gaussian latent variable z

$$x = Wz + \mu + \epsilon \tag{13}$$

  where
  - z is an M-dimensional Gaussian latent variable
  - W is an $(D \times M)$ matrix (the latent space)
  - ϵ is a D-dimensional Gaussian noise
  - ϵ and z are independent
  - μ is a parameter vector that permits the model to have non zero mean
- Factor analysis: $\epsilon \backsim N(O, \Psi)$
- Probabilistic PCA: $\epsilon \backsim N(O, \sigma^2 I)$

Probabilistic PCA (3)

- The use of the isotropic Gaussian noise model for ϵ implies that the z-conditional probability distribution over x-space is given by

$$x/z \backsim \mathcal{N}(Wz + \mu, \sigma^2 I) \tag{14}$$

- Defining $z \backsim \mathcal{N}(0, I)$, the marginal distribution of x is obtained by integrating out the latent variables and is likewise Gaussian

$$x \backsim N(\mu, C) \tag{15}$$

  with $C = WW^T + \sigma^2 I$

To do: estimate the parameters: μ, W and σ^2

Chapter content

Maximum likelihood PCA

Given a data set $X = \{x_n\}$ of observed data points, the log likelihood is given by:

$$L = -\frac{ND}{2}\ln(2\pi) - \frac{N}{2}\ln \mid C \mid -\frac{1}{2}\sum_{n=1} N(x_n - \mu)^T C^{-1}(x_n - \mu) \tag{16}$$

Setting the derivative with respect to μ gives

$$\mu = \bar{x} \tag{17}$$

Back-substituting,we can write:

$$L = -\frac{ND}{2}\ln(2\pi) + \ln \mid C \mid + Tr(C^{-1}S) \tag{18}$$

This solution represents the unique maximum

Maximum likelihood PCA (2)

Maximization with respect to W and σ^2 is more complex but has an exact closed-form solution

$$W_{ML} = U_M(L_M - \sigma^2 I)^{\frac{1}{2}} R \tag{19}$$

where

- U_M is a $(D \times M)$ matrix whose columns are given by the eigenvectors of S whose eigenvalues are the M largest
- L_M is an $(M \times M)$ diagonal matrix given by the corresponding eigenvalues λ_i
- R is an arbitrary $(M \times M)$ orthogonal matrix.

$$\sigma^2_{ML} = \frac{1}{D-M} \sum_{i=M+1}^{D} \lambda_i \tag{20}$$

Average variance of the discarded dimensions

Maximum likelihood PCA (3)

- R can be interpreted as a rotation matrix in the $M \times M$ latent space
- The predictive density is unchanged by rotations
- If $R = I$, the columns of W are the principle component eigenvectors scaled by the variance $\lambda_i - \sigma^2$
- The model correctly captures the variance of the data along the principal axes and approximates the variance in all remaining directions with a single average value σ^2. Variance 'lost' in the projections.

Maximum likelihood PCA (4)

- PCA generally expressed as a projection of points from the D-dimensional dataspace onto an M-dimensional subspace
- Use of the posterior distribution

$$z/x \backsim N(M^{-1}W^T(x-\mu), \sigma^{-2}M) \tag{21}$$

where $M = W^T W + \sigma^2 I$
The mean is given by

$$E(z/x) = M^{-1}W_{ML}^T(x-\bar{x}) \tag{22}$$

<u>Note</u>: Takes the same form as the solution of a regularized linear regression!
This projects to a point in data space given by

$$WE(z/x) + \mu \tag{23}$$

Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

EM algorithm for PCA

- In spaces of high dimensionality, computational advantages using EM!
- Can be extended to factor analysis for which there is no closed-form solution
- Can be used when values are missing, for mixture models...

Requires the complete-data log likelihood function that takes the form:

$$L_c = \sum_{n=1}^{N} \ln p(x_n/z_n) + \ln p(z_n) \tag{24}$$

In the followings, μ is substituted by the sample mean $\bar{x}$

EM algorithm for PCA

- Initialize the parameters W and σ^2
- E-step

$$\mathbb{E}[L_c] = -\sum_{n=1}^{N}\{\frac{D}{2}\ln(2\pi\sigma^2) + \frac{1}{2}\mathrm{Tr}(\mathbb{E}[z_n z_n^T])$$

$$+ \frac{1}{2\sigma^2} \mid\mid x_n - \mu \mid\mid^2 - \frac{1}{\sigma^2}\mathbb{E}[z_n^T]W^T(x_n - \mu) + \frac{1}{2\sigma^2}\mathrm{Tr}(\mathbb{E}[z_n Z_n^T]W^T W)\}$$

with

$$\mathbb{E}[z_n] = M^{-1}W(x_n - \bar{x})$$
$$\mathbb{E}[z_n z_n^T] = \sigma^2 M^{-1} + \mathbb{E}[z_n]\mathbb{E}[z_n]^T$$

- M-step

$$W_{new} = [\sum_{n=1}^{N}(x_n - \bar{x})\mathbb{E}[z_n]^T][\sum_{n=1}^{N}\mathbb{E}[z_n z_n^T]]^{-1} \tag{25}$$

$$\sigma_{new}^2 = \frac{1}{ND}\sum_{n=1}^{N}\{\mid\mid x_n - \bar{x} \mid\mid^2 - 2\mathbb{E}[z_n]^T W_{new}^T(x_n - \bar{x})$$
$$+ \mathrm{Tr}(\mathbb{E}[z_n z_n^T]W_{new}W_{new})\}$$

- Check for convergence

EM algorithm for PCA

- When $\sigma^2 \to 0$, EM approach corresponds to standard PCA
- Defining $\tilde{X}$ a matrix of size $N \times D$ whose n^{th} row is given by $x_n - \bar{x}$
- Defining Ω a matrix of size $D \times M$ whose n^{th} row is given by the vector $\mathbb{E}[z_n]$
- The E-step becomes

$$\Omega = (W_{old}^T W_{old})^{-1} W_{old}^T \tilde{X} \tag{26}$$

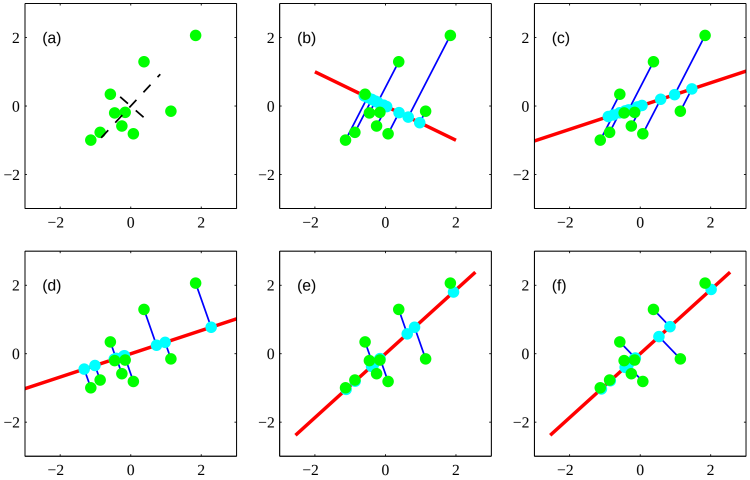
Orthogonal projection on the current estimate for the principal subspace

- The M-step takes the form

$$W_{new} = \tilde{X}^T \Omega^T (\Omega \Omega^T)^{-1} \tag{27}$$

Re-estimation of the principal subspace minimizing the squared reconstruction errors in which the projections are fixed

EM algorithm for PCA



Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

Idea of Bayesian PCA

- Usefull to choose the dimensionality M of the principal subspace
- Cross validation with a validation data set: computationally costly!
- Define an independent Gaussian prior over each column w_i of W. The variance is governed by a precision parameter α_i

$$p(W/\alpha) = \prod_{i=1}^{M} (\frac{\alpha_i}{2\pi})^{D/2} exp\{-\frac{1}{2}\alpha_i \omega_i^T w_i\} \tag{28}$$

- Values of α_i are estimated iteratively by maximizing the logarithm of the marginal likelihood function:

$$p(X/\alpha, \mu, \sigma^2) = \int p(X|W, \mu, \sigma^2) p(W|\alpha) dW \tag{29}$$

- The effective dimensionality of the principal subspace is determined by the number of finite α_i values. Principal subspace = the corresponding w_i.

Idea of Bayesian PCA

- Maximization with respect to α_i:

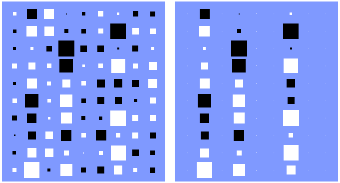
$$\alpha_i^{new} = \frac{D}{w_i^T w_i} \tag{30}$$

- These estimations are intervealed with EM algorithm with W_{new} modified

$$W_{new} = [\sum_{n=1}^{N}(x_n - \bar{x})\mathbb{E}[z_n]^T][\sum_{n=1}^{N}\mathbb{E}[z_n z_n^T] + \sigma^2 A]^{-1} \tag{31}$$

with $A = \mathrm{diag}(\alpha_i)$

- Example: 300 point in dimension D sampled from a Gaussian distribution having $M = 3$ directions with larger variance



Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

Factor analysis

- Closely related to Bayesian PCA
  $\rightarrow$ but the covariance of $p(x|z)$ diagonal instead of isotropic:

  $$x/z \backsim \mathcal{N}(Wz + \mu, \Psi) \tag{64}$$

  where Ψ is a $D \times D$ diagonal matrix.

- The components' variance of natural axes is explained by Ψ
- Observed covariance structured is captured by W
- Consequences
  - For PCA,
    rotation of data space $\Rightarrow$ same fit with W rotated with the same matrix
  - For Factor Analysis, the analogous property is: component-wise re-scaling is absorbed into the re-scaling elements of Ψ

Factor analysis

- The marginal density of the observed variable is

$$x \backsim \mathcal{N}(\mu, WW^T + \Psi) \tag{65}$$

- As in probabilistic PCA, the model is invariant w.r.t the latent space
- μ, W and Ψ can be determined by maximum likehood
- $\mu = \bar{x}$, as in probabilistic PCA
- But no closed-form ML solution for W
  $\rightarrow$ iteratively estimated using EM

Parameters estimation using EM

- $\mathbb{E}$ step

$$\mathbb{E}[z_n] = GW^T\Psi^{-1}(x - \bar{x}) \tag{66}$$

$$\mathbb{E}[z_n z_n^T] = G + \mathbb{E}[z_n]\ \mathbb{E}[z_n]^T \tag{67}$$

where $G = (I + W^T\Psi^{-1}W)^{-1}$

- $\mathbb{M}$ step

$$W^{\text{new}} = \left[\sum_{n=1}^{N}(x - \bar{x})\mathbb{E}[z_n]^T\right]\left[\sum_{n}^{N}\mathbb{E}[z_n z_n^T]\right]^{-1} \tag{69}$$

$$\Psi^{\text{new}} = \text{diag}\left\{S - W^{\text{new}}\frac{1}{N}\sum_{n=1}^{N}\mathbb{E}[z_n](x_n - \bar{x})^T\right\} \tag{70}$$

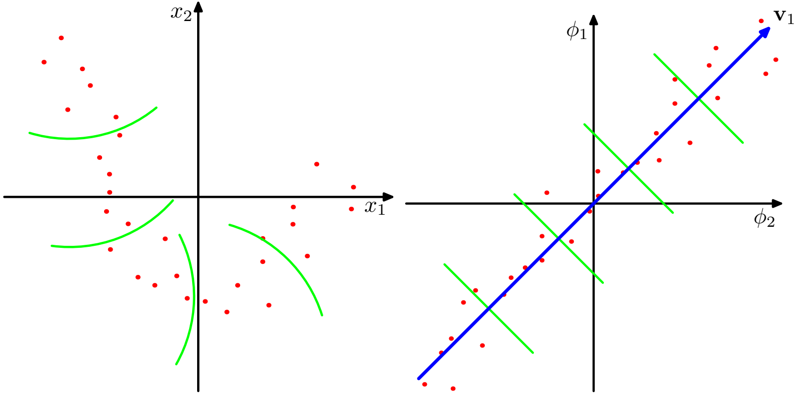
where the "diag" operator zeros all non diagonal elements

Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

Kernel PCA

Applying the ideas of Kernel substitution (see Chapter 5) to PCA



Kernel PCA: preliminaries

Kernel substitution: express each step of PCA in terms of the inner product $x^T x$ between data vectors to generalize the inner product

- Recall that the principal components are defined as

$$Su_i = \lambda_i u_i \tag{71}$$

with $||u_i||_2 = u_i^T u_i = 1$ and covariance Matrix S defined as

$$S = \frac{1}{N} \sum_{n=1}^{N} x_n x_n^T \tag{72}$$

- Consider a nonlinear mapping transformation Φ into a M-dimensional feature space
$\rightarrow$ maps any data point x_n onto $\Phi(x_n)$

Kernel PCA

- Let assume that $\sum_n \Phi(x_n) = 0$
- the $M \times M$ sample covariance matrix C in feature space is given by

$$C = \frac{1}{N} \sum_{n=1}^{N} \Phi(x_n)\Phi(x_n)^T \qquad (73)$$

with eigenvector expansion as

$$Cv_i = \lambda_i v_i, \quad i = 1, ..., M \qquad (74)$$

- Goal: solve this eigenvalue problem without working explicitly in the feature space
- Eigenvector v_i can be written as a linear combination of the $\Phi(x_n)$, of the form

$$v_i = \sum_{n=1}^{M} a_{in} \Phi(x_n) \qquad (76)$$

Kernel PCA

Note: typo in (12.78)

- The eigenvector equation can then be defined in terms of the kernel function as

$$K^2 a_i = \lambda_i N K a_i \tag{79}$$

where $a_i = (a_{1i}, \ldots, a_{Ni})^T$, unknown at this point.

- The a_i can be found by solving the eigenvalue problem:

$$K a_i = \lambda_i N a_i \tag{80}$$

- The a_i's normalization condition is obtained by requiring that the eigenvectors in feature space be normalized:

$$1 = v_i^T v_i = a_i^T K a_i = \lambda_i N a_i^T a_i \tag{81}$$

Kernel PCA

- The resulting principal component projections can also be cast in terms of the kernel function
- A point x is "projected" onto eigenvector i as

$$y_i(x) = \Phi(x)^T \; v_i = \sum_{n=1}^{N} a_{in} \; k(x, x_n) \qquad (82)$$

- Remarks:
  - At most D linear principal components
  - The number of nonlinear principal components can exceed D
  - The number of nonzero eigenvalues cannot exceed the number of data points N

Kernel PCA

- Up to now, we assumed that the projected data has zero mean

$$\sum_{i=1}^{N} \Phi(x_n) = 0$$

- This mean can't be simply computed and subtracted
- However the projected data points after centralizing can be obtained as

$$\tilde{\Phi}(x_n) = \Phi(x_n) - \frac{1}{N} \sum_{l=1}^{N} \Phi(x_l) \tag{83}$$

Kernel PCA

- The corresponding elements of the Gram matrix are given by

$$\begin{aligned}\tilde{K}_{nm} &= \tilde{\Phi}(x_n)^T\ \tilde{\Phi}(x_m)\\ &= k(x_n, x_m) - \frac{1}{N}\sum_{l=1}^{N} k(x_l, x_m)\\ &- \frac{1}{N}\sum_{l=1}^{N} k(x_n, x_l) + \frac{1}{N^2}\sum_{j=1}^{N}\sum_{l=1}^{N} k(x_j, x_l)\end{aligned} \tag{84}$$

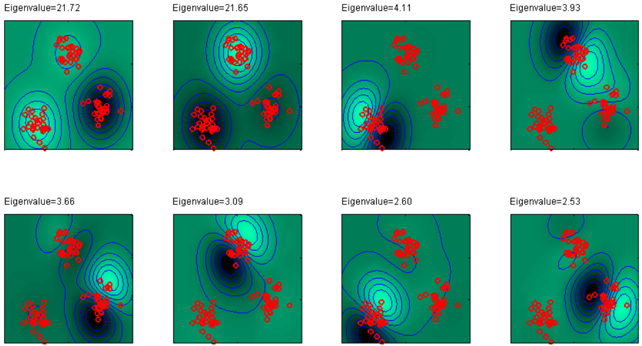
i.e.,

$$\tilde{K} = K - 1_N K - K 1_N + 1_N K 1_N, \tag{85}$$

where

$$1_N = \begin{bmatrix} 1/N & ... & 1/N \\ \dots & & \dots \\ 1/N & ... & 1/N \end{bmatrix}$$

Kernel PCA: example and remark



- PCA is often used to reconstruct a sample x_n with good accuracy from its projections on the first principal components
- In kernel PCA, this is not possible in general, as we can't map points explicitly from the feature space to the data space

Chapter content

Independent component analysis

- Consider models in which
  - the observed variables are related linearly to the latent variables
  - for which the latent distribution is non-Gaussian
- Important class of such models: independent component analysis, for which

$$p(z) = \prod_{j=1}^{M} p(z_j) \tag{86}$$

$\rightarrow$ the distribution of latent variables factorize

Application case: blind source separation

- Setup:
  - Two people talking at the same time
  - their voices recorded using two microphones
- Objective: to reconstruct the two signal separately
  $\rightarrow$ "blind" because we are given only the mixed data. We haven't observed
  - the original sources
  - the mixing coefficients
- under some assumptions (no time delay and echoes)
  - the signals received by the microphone are linear combinations of the voice amplitudes
  - the coefficient of this linear combination are constant

Application case: blind source separation

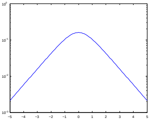
- Hereafter: a possible approach (see Mackay'03)
  $\longrightarrow$ that does not consider the temporal aspect of the problem
- Consider generative model with
  - the latent variables: unobserved speech signal amplitudes
  - the two observed signal values $o = [o_1 \ o_2]^T$ at the microphones
- Distribution of latent variables factorizes as $p(z) = p(z_1)p(z_2)$
- No need to include noise: observed variables = deterministic linear combinations of latent variables as

$$o = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} z$$

Application case: blind source separation

- Given a set of observation
  - the likehood function is a function of the coefficients a_{ij}
  - log likehood maximized using gradient-based optimization $\rightarrow$ particular case of independent analysis
- This requires that the latent variables have non Gaussian distributions
  - Probabilistic PCA: latent-space distribution = zero-mean isotropic Gaussian
  - No way to distinguish between two choices for the latent variables $\rightarrow$ these differ by a rotation in the latent space
- Common choice for the latent-variable distribution:

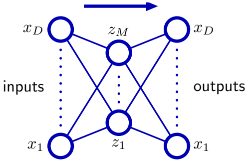
$$p(z_j) = \frac{1}{\pi \cosh(z_j)} = \frac{1}{\pi(e^{z_j} + e^{-z_j})} \tag{90}$$



Chapter content

Autoassociative neural networks

- Chapter 5: Neural networks for predicting outputs given inputs
- They can also used for dimensionality reduction



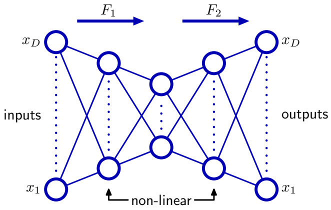
- Network that perform an *autoassociative* mapping
  - #outputs = #inputs > number of hidden units
    $\Rightarrow$ no perfect reconstruction
  - find networks parameters w minimizing a given error function
    $\rightarrow$ for instance sum-of-square errors

$$E(w) = \frac{1}{2} \sum_{n=1}^{N} ||y(x_n, w) - x_n||^2 \tag{91}$$

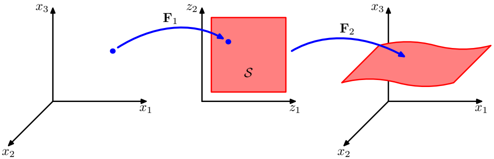
Autoassociative neural networks

- Linear activations functions $\Rightarrow$
  - unique global minimum
  - the network performs projections onto the M-dimensional principal component subspace
  - this subspace is spanned by the vector of weights
- Even with nonlinear hiddens units, minimum error obtained by principal component subspace
  $\Rightarrow$ no advantage of using two-layer neural networks to perform dimensionality reduction: use standard PCA techniques

Autoassociative neural networks



- Using more hidden layers (4 here), the approach is worthful



- Training the network involves nonlinear optimization techniques (with risk of suboptimally)

Chapter content

- Principal Component Analysis
  - Maximum variance formulation
  - Minimum-error formulation
  - Applications of PCA
  - PCA for high-dimensional data
- Probabilistic PCA
  - Problem setup
  - Maximum likelihood PCA
  - EM algorithm for PCA
  - Bayesian PCA
  - Factor analysis
- Kernel PCA
- Nonlinear Latent Variable Models
  - Independent component analysis
  - Autoassociative neural networks
  - Modelling nonlinear manifolds

Modelling nonlinear manifolds

- Data may lie in a manifold of lower dimensionality than the observed data space
- capture this property explicitly may improve the density modelling
- Possible approach: non-linear manifold modelled by piece-wise linear approximation, e.g.,
  - k-means + PCA for each cluster
  - better: use reconstruction error for cluster assignment
- These are limited by not having an overall density model
- Tipping and Bishop: full probablistic model using a mixture distribution in which components are probabilistic PCA
  $\Rightarrow$ both discrete latent variables and continuous ones

Modelling nonlinear manifolds

Alternative approach: to use a single nonlinear model

- Principal curves:
  - Extension of PCA (that finds a linear subspace)
  - A curve is described by a vector-valued function $f(\lambda)$
  - Natural parametrization: the arc length along the curve
  - Given a point $\hat{x}$, we can find the closest point $\lambda = g_f(x)$ on the curve in terms of the Euclidean distance
  - A principal curve is a curve for which every point on the curve is the mean of all points in data space to project to it, so that

    $$\mathbb{E}[x|g_f(x) = \lambda] = f(\lambda) \tag{92}$$

    $\rightarrow$ there may be many principal curves for a continuous distribution
  - Hastie et al: two-stage iterative procedure for finding principal curve

Modelling nonlinear manifolds: MDS

- PCA is often used for the purpose of visualization
- Another technique with a similar aim: multidimensional scaling (MDS, Cox and Cox 2000)
  - preserve as closely as possible the pairwise distances between data points
  - involves finding the eigenvectors of the distance matrix
  - equivalent results to PCA when the distance is Euclidean
    $\rightarrow$ but can be extended to a wide variety of data types specified in terms of a similarity matrix

Modelling nonlinear manifolds: LLE

- Locally linear embedding (LLE, Roweis and Saul 2000)
- Compute the set of coefficients that best reconstruct each data point from its neighbours
- coefficients arranged to be invariant to rotation, translations, scaling
  $\rightarrow$ characterize the local geometrical properties of the neighborhood
- LLE maps the high-dimensional data to a lower dimensional subspace while preserving these coefficients
- These weights are used to reconstruct the data points in low-dimensional space as in the high dimensional space
- Albeit non linear, LLE does not exhibit local minima

Modelling nonlinear manifolds: ISOMAP

- Isometric feature mapping (ISOMAP, Tenenbaum et al. 2000)
- Goal: data projected to a lower-dimensional space using MDS
  $\rightarrow$ but dissimilarities defined in terms of the *geodesic distances* on the manifold
- Algorithm:
  - First defines the neighborhood using KNN or ε-search
  - Construct a neighborhood graph with weights corresponding to the Euclidean distances
  - Geodesic distance approximated by the sum of Euclidean distances along the shortest path connecting two points
  - Apply MDS to the geodesic distances

Modelling nonlinear manifolds: other techniques

- Latent traits: Models having continous latent variables together with discrete observed variables
  $\longrightarrow$ can be used to visualize binary vectors analogously to PCA for continuous variables
- Density network: nonlinear function governed by a multilayered neural network
  $\longrightarrow$ flexible model but computationally intensive
- Generative topographic mapping (GTM): restricted forms for the nonlinear function $\Rightarrow$ nonlinear and efficient to train
  - latent distribution defined by a finite regular grid over the latent space (of dimensionality 2, typically)
  - can be seen as a probabilistic version of the self-organizing map (SOM, Kohonen)