# Non-linear Bayesian Image Modelling

Christopher M. Bishop[1] and John M. Winn[2]

[1] Microsoft Research
St. George House, 1 Guildhall Street
Cambridge CB2 3NH, U.K.
cmbishop@microsoft.com

[2] University of Cambridge
Department of Engineering
Trumpington Street
Cambridge, CB2 1PZ, U.K.
jmw39@cam.ac.uk

**Abstract.** In recent years several techniques have been proposed for modelling the low-dimensional manifolds, or 'subspaces', of natural images. Examples include principal component analysis (as used for instance in 'eigen-faces'), independent component analysis, and auto-encoder neural networks. Such methods suffer from a number of restrictions such as the limitation to linear manifolds or the absence of a probablistic representation. In this paper we exploit recent developments in the fields of variational inference and latent variable models to develop a novel and tractable probabilistic approach to modelling manifolds which can handle complex non-linearities. Our framework comprises a mixture of sub-space components in which both the number of components and the effective dimensionality of the sub-spaces are determined *automatically* as part of the Bayesian inference procedure. We illustrate our approach using two classical problems: modelling the manifold of face images and modelling the manifolds of hand-written digits.

## 1 Introduction

Interest in image subspace modelling has grown considerably in recent years in contexts such as recognition, detection, verification and coding. Although an individual image can be considered as a point in a high-dimensional space described by the pixel values, an ensemble of related images, for example faces, lives on a (noisy) non-linear manifold having much a much lower *intrinsic* dimensionality. One of the simplest approaches to modelling such manifolds involves finding the principal components of the ensemble of images, as used for example in 'eigen-faces' [15].

However, simple principal component analysis suffers from two key limitations. First, it does not directly define a probability distribution, and so it is difficult to use standard PCA as a natural component in a probabilistic solution to a computer vision problem. Second, the manifold defined by PCA is necessarily linear. Techniques which address the first of these problems by constructing a density model include Gaussians and mixtures of Gaussians [12]. The second problem has been addressed by considering non-linear projective methods such as principal curves and auto-encoder neural networks

D. Vernon (Ed.): ECCV 2000, LNCS 1842, pp. 3–17, 2000.

[11]. Bregler and Omohundro [5] and Heap and Hogg [9] use mixture representations to try to capture the non-linearity of the manifold. However, their model fitting is based on simple clustering algorithms (related to $K$-means) and lacks the fully probabilistic approach as discussed in this paper.

A central problem in density modelling in high dimensional spaces concerns model complexity. Models fitted using maximum likelihood are particularly prone to severe over-fitting unless the number of free parameters is restricted to be much less than the number of data points. For example, it is clearly not feasible to fit an unconstrained mixture of Gaussians directly to the data in the original high-dimensional space using maximum likelihood due to the excessive number of parameters in the covariance matrices. Moghaddam and Pentland [12] therefore project the data onto a PCA sub-space and then perform density estimation within this lower dimensional space using Gaussian mixtures. While this limits the number of free parameters in the model, the non-linearity of the manifold requires the PCA space to have a significantly higher dimensionality than that of the manifold itself, and so again the model is prone to over-parameterization.

One important aspect of model complexity concerns the dimensionality of the manifold itself, which is typically not known in advance. Moghaddam [11], for example, arbitrarily fixes the model dimensionality to be 20.

In this paper we present a sophisticated Bayesian framework for modelling the manifolds of images. Our approach constructs a probabilistically consistent density model which can capture essentially arbitrary non-linearities and which can also discover an appropriate dimensionality for modelling the manifold. A key feature is the use of a fully Bayesian formulation in which the appropriate model complexity, and indeed the dimensionality of the manifold itself, can be discovered *automatically* as part of the inference procedure [2]. The model is based on a mixture of components each of which is a latent variable model whose dimensionality can be inferred from the data. It avoids a discrete model search over dimensionality , involving instead the use of continuous hyper-parameters to determine an *effective* dimensionality for the components in the mixture model.

Our approach builds on recent developments in latent variable models and variational inference. In Section 2 we describe the probabilistic model, and in Section 3 we explain the variational framework used to fit it to the data. Results from face data and from images of hand-written digits are presented in Section 4 and conclusions given in Section 5.

Note that several authors have explored the use of non-linear warping of the image, for example in the context of face recognition, in order to take account of changes of pose or of interpersonal variation [4,6,7]. In so far as such distortions can be accurately represented, these transformations should be of significant benefit in tackling the subspace modelling problem, albeit at increased computational expense. It should be emphasised that such approaches can be used to augment virtually any sub-space modelling algorithm, including those discussed in this paper, and so they will not be considered further.

## 2 Models for Manifolds

Our approach to modelling the manifolds of images builds upon recent developments in latent variable models and can be seen as a natural development of PCA and mixture modelling frameworks leading to a highly flexible, fully probabilistic framework. We begin by showing how conventional PCA can be reformulated probabilistically and hence used as the component distribution in a mixture model. Then we show how a Bayesian approach allows the model complexity (including the number of components in the mixture as well as the effective dimensionality of the manifold) to be inferred from the data.

### 2.1 Maximum Likelihood PCA

Principal component analysis (PCA) is a widely used technique for data analysis. It can be defined as the linear projection of a data set into a lower-dimensional space under which the retained variance is a maximum, or equivalently under which the sum-of-squares reconstruction cost is minimized.

Consider a data set $D$ of observed $d$-dimensional vectors $D = \{\mathbf{t}_n\}$ where $n \in \{1, \ldots, N\}$. Conventional PCA is obtained by first computing the sample covariance matrix given by

$$\mathbf{S} = \frac{1}{N} \sum_{n=1}^{N} (\mathbf{t}_n - \bar{\mathbf{t}})(\mathbf{t}_n - \bar{\mathbf{t}})^{\mathrm{T}} \tag{1}$$

where $\bar{\mathbf{t}} = N^{-1} \sum_n \mathbf{t}_n$ is the sample mean. Next the eigenvectors $\mathbf{u}_i$ and eigenvalues $\lambda_i$ of $\mathbf{S}$ are found, where $\mathbf{S}\mathbf{u}_i = \lambda_i \mathbf{u}_i$ and $i = 1, \ldots, d$. The eigenvectors corresponding to the $q$ largest eigenvalues (where $q < d$) are retained, and a reduced-dimensionality representation of the data set is defined by $\mathbf{x}_n = \mathbf{U}_q^{\mathrm{T}}(\mathbf{t}_n - \bar{\mathbf{t}})$ where $\mathbf{U}_q = (\mathbf{u}_1, \ldots, \mathbf{u}_q)$.

A significant limitation of conventional PCA is that it does not define a probability distribution. Recently, however, Tipping and Bishop [14] showed how PCA can be reformulated as the maximum likelihood solution of a specific latent variable model, as follows. We first introduce a $q$-dimensional latent variable $\mathbf{x}$ whose prior distribution is a zero mean Gaussian $P(\mathbf{x}) = \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$ and $\mathbf{I}_q$ is the $q$-dimensional unit matrix. The observed variable $\mathbf{t}$ is then defined as a linear transformation of $\mathbf{x}$ with additive Gaussian noise $\mathbf{t} = \mathbf{W}\mathbf{x} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$ where $\mathbf{W}$ is a $d \times q$ matrix, $\boldsymbol{\mu}$ is a $d$-dimensional vector and $\boldsymbol{\epsilon}$ is a zero-mean Gaussian-distributed vector with covariance $\tau^{-1}\mathbf{I}_d$ (where $\tau$ is an inverse variance, often called the 'precision'). Thus $P(\mathbf{t}|\mathbf{x}) = \mathcal{N}(\mathbf{W}\mathbf{x} + \boldsymbol{\mu}, \tau^{-1}\mathbf{I}_d)$. The marginal distribution of the observed variable is then given by the convolution of two Gaussians and is itself Gaussian

$$P(\mathbf{t}) = \int P(\mathbf{t}|\mathbf{x}) P(\mathbf{x})\, d\mathbf{x} = \mathcal{N}(\boldsymbol{\mu}, \mathbf{C}) \tag{2}$$

where the covariance matrix $\mathbf{C} = \mathbf{W}\mathbf{W}^{\mathrm{T}} + \tau^{-1}\mathbf{I}_d$. The model (2) represents a constrained Gaussian distribution governed by the parameters $\boldsymbol{\mu}$, $\mathbf{W}$ and $\tau$.

It was shown by Tipping and Bishop [14] that the stationary points of the log likelihood with respect to $\mathbf{W}$ satisfy

$$\mathbf{W}_{\mathrm{ML}} = \mathbf{U}_q(\boldsymbol{\Lambda}_q - \tau^{-1}\mathbf{I}_q)^{1/2} \tag{3}$$

where the columns of $\mathbf{U}_q$ are eigenvectors of $\mathbf{S}$, with corresponding eigenvalues in the diagonal matrix $\boldsymbol{\Lambda}_q$. It was also shown that the *maximum* of the likelihood is achieved when the $q$ largest eigenvalues are chosen, so that the columns of $\mathbf{U}_q$ correspond to the *principal* eigenvectors, with all other choices of eigenvalues corresponding to saddle points. The maximum likelihood solution for $\tau$ is then given by

$$\frac{1}{\tau_{\mathrm{ML}}} = \frac{1}{d-q}\sum_{i=q+1}^{d}\lambda_i \tag{4}$$

which has a natural interpretation as the average variance lost per discarded dimension. The density model (2) thus represents a probabilistic formulation of PCA. It is easily verified that conventional PCA is recovered by treating $\tau$ as a parameter and taking the limit $\tau \to \infty$.

Probabilistic PCA has been successfully applied to problems in data compression, density estimation and data visualization, and has been extended to mixture and hierarchical mixture models [13,14,3]. As with conventional PCA, however, the model itself provides no mechanism for determining the value of the latent-space dimensionality $q$. For $q = d-1$ the model is equivalent to a full-covariance Gaussian distribution[1], while for $q < d-1$ it represents a constrained Gaussian in which the variance in the remaining $d-q$ directions is modelled by the single parameter $\tau$. Thus the choice of $q$ corresponds to a problem in model complexity optimization. In principal cross-validation to compare all possible values of $q$ offers a possible approach. However, maximum likelihood estimation is highly biased (leading to 'overfitting') and so in practice excessively large data sets would be required and the procedure would become computationally intractable.

### 2.2 Bayesian PCA

The issue of model complexity can be handled naturally within a Bayesian paradigm. Armed with the probabilistic reformulation of PCA defined in Section 2.1, a Bayesian treatment of PCA is obtained by first introducing prior distributions over the parameters $\boldsymbol{\mu}$, $\mathbf{W}$ and $\tau$. A key goal is to control the effective dimensionality of the latent space (corresponding to the number of retained principal components). Furthermore, we seek to avoid discrete model selection and hence we introduce continuous hyperparameters to determine automatically an appropriate *effective* dimensionality for the latent space as part of the process of Bayesian inference. This is achieved by introducing a *hierarchical* prior $P(\mathbf{W}|\boldsymbol{\alpha})$ over the matrix $\mathbf{W}$, governed by a $q$-dimensional

[1] This follows from the fact that the $q-1$ linearly independent columns of $\mathbf{W}$ have independent variances along $q-1$ directions, while the variance along the remaining direction is controlled by $\tau$.

vector of hyper-parameters $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_q\}$. Each hyper-parameter controls one of the columns of the matrix $\mathbf{W}$ through a conditional Gaussian distribution of the form

$$P(\mathbf{W}|\boldsymbol{\alpha}) = \prod_{i=1}^{q} \left(\frac{\alpha_i}{2\pi}\right)^{d/2} \exp\left\{-\frac{1}{2}\alpha_i \|\mathbf{w}_i\|^2\right\} \tag{5}$$

where $\{\mathbf{w}_i\}$ are the columns of $\mathbf{W}$. This form of prior is motivated by the framework of *automatic relevance determination* (ARD) introduced in the context of neural networks by Neal and MacKay (see MacKay, 1995). Each $\alpha_i$ controls the inverse variance of the corresponding $\mathbf{w}_i$, so that if a particular $\alpha_i$ has a posterior distribution concentrated at large values, the corresponding $\mathbf{w}_i$ will tend to be small, and that direction in latent space will be effectively 'switched off'. The dimensionality of the latent space is set to its maximum possible value $q = d - 1$.

We complete the specification of the Bayesian model by defining the remaining priors to have the form

$$P(\boldsymbol{\mu}) = \mathcal{N}(\boldsymbol{\mu}|\mathbf{0}, \beta^{-\mathbf{1}}\mathbf{I}) \tag{6}$$

$$P(\boldsymbol{\alpha}) = \prod_{i=1}^{q} \Gamma(\alpha_i|a_\alpha, b_\alpha) \tag{7}$$

$$P(\tau) = \Gamma(\tau|c_\tau, d_\tau). \tag{8}$$

Here $\mathcal{N}(\mathbf{x}|\mathbf{m}, \boldsymbol{\Sigma})$ denotes a multivariate normal distribution over $\mathbf{x}$ with mean $\mathbf{m}$ and covariance matrix $\boldsymbol{\Sigma}$. Similarly, $\Gamma(x|a, b)$ denotes a Gamma distribution over $x$ given by

$$\Gamma(x|a, b) = \frac{b^a x^{a-1} e^{-bx}}{\Gamma(a)} \tag{9}$$

where $\Gamma(a)$ is the Gamma function. We obtain broad priors by setting $a_\alpha = b_\alpha = a_\tau = b_\tau = 10^{-3}$ and $\beta = 10^{-3}$.

As an illustration of the role of the hyperparameters in determining model complexity, we consider a data set consisting of 300 points in 10 dimensions, in which the data is drawn from a Gaussian distribution having standard deviation 1.0 in 3 directions and standard deviation $0.5$ in the remaining 7 directions. The result of fitting both maximum likelihood and Bayesian PCA models is shown in Figure 1. (The Bayesian model was trained using the variational approach discussed in Section 3.) In this case the Bayesian model has an effective dimensionality of $q_{\text{eff}} = 3$ as expected.

### 2.3 Mixtures of Bayesian PCA Models

Given a probabilistic formulation of PCA we can use it to construct a mixture distribution comprising a linear superposition of principal component analyzers. If we were to fit such a model to data using maximum likelihood we would have to choose both the number $M$ of components and the latent space dimensionality $q$ of the components. For moderate numbers of components and data spaces of several dimensions it quickly becomes computationally costly to use cross-validation.

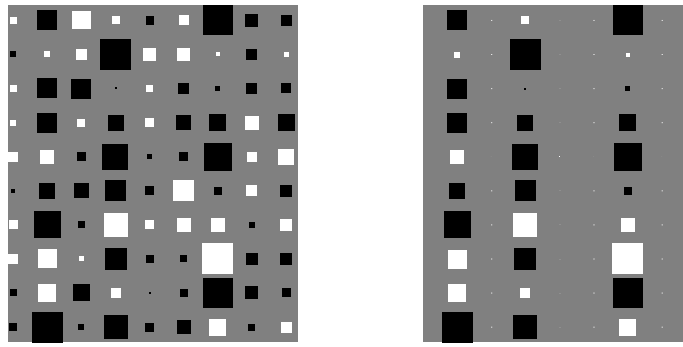

**Fig. 1.** Hinton diagrams of the matrix $\mathbf{W}$ for a data set in 10 dimensions having $m = 3$ directions with larger variance than the remaining 7 directions. The area of each square is proportional to the magnitude of the corresponding matrix element, and the squares are white for positive values and black for negative values. The left plot shows $\mathbf{W}_{\mathrm{ML}}$ from maximum likelihood PCA while the right plot shows the posterior mean $\langle \mathbf{W} \rangle$ from the Bayesian approach, showing how the model is able to discover the appropriate dimensionality by suppressing the 6 surplus degrees of freedom.

Here Bayesian PCA offers a significant advantage in allowing the effective dimensionalities of the models to be determined automatically. Furthermore, we also wish to determine the appropriate number of components in the mixture. We do this by Bayesian model comparison [1] as an integral part of the learning procedure as discussed in the next section.

To formulate the probabilistic model we introduce, for each data point $\mathbf{t}_n$, an additional $M$-dimensional binary latent variable $\mathbf{s}_n$ which has one non-zero element denoting which of the $M$ components in the mixture is responsible for generating $\mathbf{t}_n$. These discrete latent variables have distributions governed by hyperparameters $\boldsymbol{\pi} = \{\pi_m\}$ where $m = 1, \ldots, M$,

$$P(\mathbf{s} = \delta_m | \boldsymbol{\pi}) = \pi_m \tag{10}$$

where $\delta_m$ denotes a vector with all elements zero except element $m$ whose value is 1. The parameters $\boldsymbol{\pi}$ are given a Dirichlet distribution

$$P(\boldsymbol{\pi}) = \mathrm{Dir}(\boldsymbol{\pi}|\mathbf{u}) = \frac{1}{Z(\mathbf{u})} \prod_{i=1}^{M} \pi_i^{u_i - 1} \delta \left( \sum_{i=1}^{M} \pi_i - 1 \right) \tag{11}$$

with $\mathbf{u}$ are parameters of the distribution, and $Z(\mathbf{u})$ is the normalization constant.

In a simple mixture of Bayesian PCA models, each component would be free to determine its own dimensionality. A central goal of this work, however, is to model a continuous non-linear manifold. We therefore wish the components in the mixture to have a common dimensionality whose value is a-priori unknown and which should be inferred from the data. This can be achieved within our framework by using a single set of $\boldsymbol{\alpha}$ hyper-parameters which are *shared* by all of the components in the mixture. The probabilistic structure of the resulting model is displayed diagrammatically in Figure 2.

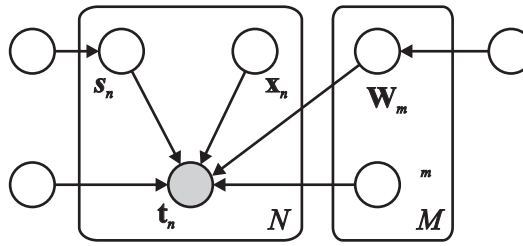


**Fig. 2.** Representation of a Bayesian PCA mixture as a probabilistic graphical model (directed acyclic graph) showing the hierarchical prior over $\mathbf{W}$ governed by the vector of shared hyperparameters $\boldsymbol{\alpha}$. The boxes denote 'plates' comprising $N$ independent observations of the data vector $\mathbf{t}_n$ (shown shaded) together with the corresponding hidden variables $\mathbf{x}_n$ and $\mathbf{s}_n$, with a similar plate denoting the $M$ copies of the parameters associated with each component in the mixture.

## 3 Variational Inference

In common with many complex probabilistic models, exact computation cannot be performed analytically. We avoid the computational complexity, and difficulty of convergence assessment, associated with Markov chain Monte Carlo methods by using variational inference [10]. For completeness we first give a brief overview of variational methods and then describe the variational solution for the Bayesian Mixture PCA model.

In order to motivate the variational approach, consider a general probabilistic model with parameters $\boldsymbol{\theta} = \{\theta_i\}$ and observed data $D$, for which the marginal probability of the data is given by

$$P(D) = \int P(D, \boldsymbol{\theta})\, d\boldsymbol{\theta}. \tag{12}$$

We have already noted that integration with respect to the parameters is analytically intractable. Variational methods involve the introduction of a distribution $Q(\boldsymbol{\theta})$ which, as we shall see shortly, provides an approximation to the true posterior distribution. Consider the following transformation applied to the log marginal likelihood

$$\ln P(D) = \ln \int P(D, \boldsymbol{\theta})\, d\boldsymbol{\theta} \tag{13}$$

$$= \ln \int Q(\boldsymbol{\theta}) \frac{P(D, \boldsymbol{\theta})}{Q(\boldsymbol{\theta})}\, d\boldsymbol{\theta} \tag{14}$$

$$\geq \int Q(\boldsymbol{\theta}) \ln \frac{P(D, \boldsymbol{\theta})}{Q(\boldsymbol{\theta})}\, d\boldsymbol{\theta} = \mathcal{L}(Q) \tag{15}$$

where we have applied Jensen's inequality. We see that the function $\mathcal{L}(Q)$ forms a rigorous lower bound on the true log marginal likelihood. The significance of this transformation is that, through a suitable choice for the $Q$ distribution, the quantity $\mathcal{L}(Q)$ may be tractable to compute, even though the original log likelihood function is not.

From (15) it is easy to see that the difference between the true log marginal likelihood $\ln P(D)$ and the bound $\mathcal{L}(Q)$ is given by

$$\mathrm{KL}(Q\|P) = -\int Q(\boldsymbol{\theta}) \ln \frac{P(\boldsymbol{\theta}|D)}{Q(\boldsymbol{\theta})}\, d\boldsymbol{\theta} \tag{16}$$

which is the Kullback-Leibler (KL) divergence between the approximating distribution $Q(\boldsymbol{\theta})$ and the true posterior $P(\boldsymbol{\theta}|D)$. The relationship between the various quantities is shown in Figure 3.

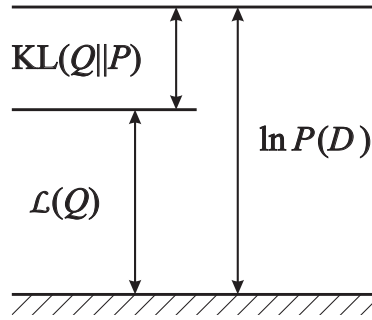


**Fig. 3.** The quantity $\mathcal{L}(Q)$ provides a rigorous lower bound on the true log marginal likelihood $\ln P(D)$, with the difference being given by the Kullback-Leibler divergence $\mathrm{KL}(Q\|P)$ between the approximating distribution $Q(\boldsymbol{\theta})$ and the true posterior $P(\boldsymbol{\theta}|D)$.

Suppose we consider a completely free-form optimization over $Q$, allowing for all possible $Q$ distributions. Using the well-known result that the KL divergence between two distributions $Q(\boldsymbol{\theta})$ and $P(\boldsymbol{\theta})$ is minimized by $Q(\boldsymbol{\theta}) = P(\boldsymbol{\theta})$ we see that the optimal $Q$ distribution is given by the true posterior, in which case the KL divergence is zero, the bound becomes exact and $\mathcal{L}(Q) = \ln P(D)$. However, this will not lead to any simplification of the problem since, by assumption, direct evaluation of $\ln P(D)$ is intractable. In order to make progress it is necessary to restrict the range of $Q$ distributions.

The goal in a variational approach is to choose a suitable form for $Q(\boldsymbol{\theta})$ which is sufficiently simple that the lower bound $\mathcal{L}(Q)$ can readily be evaluated and yet which is sufficiently flexible that the bound is reasonably tight. We generally choose some family of $Q$ distributions and then seek the best approximation within this family by maximizing the lower bound $\mathcal{L}(Q)$. Since the true log likelihood is independent of $Q$ we see that this is equivalent to minimizing the Kullback-Leibler divergence.

One approach is to consider a parametric family of $Q$ distributions of the form $Q(\boldsymbol{\theta}; \psi)$ governed by a set of parameters $\psi$. We can then adapt $\psi$ by minimizing the KL divergence to find the best approximation within this family. Here we consider an alternative approach which is to restrict the functional form of $Q(\boldsymbol{\theta})$ by assuming that it

factorizes over the component variables $\{\theta_i\}$ in $\boldsymbol{\theta}$, so that

$$Q(\boldsymbol{\theta}) = \prod_i Q_i(\theta_i). \tag{17}$$

The KL divergence can then be minimized over all possible factorial distributions by performing a free-form minimization over each of the $Q_i$, leading to the following result

$$Q_i(\theta_i) = \frac{\exp\left\langle \ln P(D, \boldsymbol{\theta})\right\rangle_{k\neq i}}{\int \exp\left\langle \ln P(D, \boldsymbol{\theta})\right\rangle_{k\neq i} d\theta_i} \tag{18}$$

where $\langle\ \cdot\ \rangle_{k\neq i}$ denotes an expectation with respect to the distributions $Q_k(\theta_k)$ for all $k \neq i$. For models having suitable conjugate choices for prior distributions, the right hand side of (18) can be expressed as a closed-form analytic distribution. Note, however, that it still represents a set of coupled implicit solutions for the factors $Q_k(\theta_k)$. In practice, therefore, these factors are suitably initialized and are then cyclically updated using (18).

It is worth emphasizing that, for models such as the one discussed in this paper for which this framework is tractable, it is also possible to calculate the lower bound $\mathcal{L}(Q)$ itself in closed form. Numerical evaluation of this bound during the optimization process allows convergence to be monitored, and can also be used for Bayesian model comparison since it approximates the log model probability $\ln P(V)$. It also provides a check on the accuracy of the mathematical solution and its numerical implementation, since the bound can never decrease as the result of updating one of the $Q_i$.

### 3.1 Variational Solution for Bayesian PCA Mixtures

In order to apply this framework to Bayesian PCA we assume a $Q$ distribution of the form

$$Q(S, X, \boldsymbol{\pi}, \mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\mu}, \tau) = Q(S)Q(X|S)Q(\boldsymbol{\pi})Q(\mathbf{W})Q(\boldsymbol{\alpha})Q(\boldsymbol{\mu})Q(\tau) \tag{19}$$

where $X = \{\mathbf{x}_n\}$. The joint distribution of data and parameters is given by

$$\left[\prod_{n=1}^{N} P(\mathbf{t}_n|\mathbf{x}_n, \mathbf{W}, \boldsymbol{\mu}, \tau, S)\right] P(X)P(S|\boldsymbol{\pi})P(\boldsymbol{\pi})P(\mathbf{W}|\boldsymbol{\alpha})P(\boldsymbol{\alpha})P(\boldsymbol{\mu})P(\tau). \tag{20}$$

Using (19) and (20) in (18), and substituting for the various $P(\cdot)$ distributions, we obtain the following results for the component distributions of $Q(\cdot)$

$$Q(X|S) = \prod_{n=1}^{N} Q(\mathbf{x}_n|\mathbf{s}_n) \tag{21}$$

$$Q(\mathbf{x}_n|\mathbf{s}_n = \delta_m) = \mathcal{N}(\mathbf{x}_n|\mathbf{m}_{\mathbf{x}}^{(nm)}, \boldsymbol{\Sigma}_{\mathbf{x}}^{(m)}) \tag{22}$$

$$Q(\boldsymbol{\mu}) = \prod_{m=1}^{M} \mathcal{N}(\boldsymbol{\mu}_m|\mathbf{m}_{\boldsymbol{\mu}}^{(m)}, \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{(m)}) \tag{23}$$

$$Q(\mathbf{W}) = \prod_{m=1}^{M} \prod_{k=1}^{d} \mathcal{N}(\widetilde{\mathbf{w}}_{km} | \mathbf{m}_{\mathbf{w}}^{(km)}, \boldsymbol{\Sigma}_{\mathbf{w}}^{(m)}) \tag{24}$$

$$Q(\boldsymbol{\alpha}) = \prod_{m=1}^{M} \prod_{i=1}^{q} \mathit{\Gamma}(\alpha_{mi} | \widetilde{a}_{\alpha}, \widetilde{b}_{\alpha}^{(mi)}) \tag{25}$$

$$Q(\tau) = \mathit{\Gamma}(\tau | \widetilde{a}_{\tau}, \widetilde{b}_{\tau}) \tag{26}$$

$$Q(\mathit{\Pi}) = \prod_{m=1}^{M} \mathrm{Dir}(\pi_m | \widetilde{u}^{(m)}) \tag{27}$$

$$Q(S) = \prod_{n=1}^{N} Q(\mathbf{s}_n) \tag{28}$$

where $\widetilde{\mathbf{w}}_k$ denotes a column vector corresponding to the $k$th row of $\mathbf{W}$. Here we have defined

$$\mathbf{m}_{\mathbf{x}}^{(nm)} = \langle \tau \rangle \boldsymbol{\Sigma}_{\mathbf{x}}^{(m)} \langle \mathbf{W}_m^{\mathrm{T}} \rangle (\mathbf{t}_n - \langle \boldsymbol{\mu}_m \rangle) \tag{29}$$

$$\boldsymbol{\Sigma}_{\mathbf{x}}^{(m)} = \left( \mathbf{I}_q + \langle \tau \rangle \langle \mathbf{W}_m^{\mathrm{T}} \mathbf{W}_m \rangle \right)^{-1} \tag{30}$$

$$\mathbf{m}_{\boldsymbol{\mu}}^{(m)} = \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{(m)} \langle \tau \rangle \sum_{n=1}^{N} \langle s_{nm} \rangle \left( \mathbf{t}_n - \langle \mathbf{W}_m \rangle \langle \mathbf{x}_n | m \rangle \right) \tag{31}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{(m)} = \left( \beta + \langle \tau \rangle \sum_{n=1}^{N} \langle s_{nm} \rangle \right)^{-1} \mathbf{I}_d \tag{32}$$

$$\mathbf{m}_{\mathbf{w}}^{(km)} = \boldsymbol{\Sigma}_{\mathbf{w}} \langle \tau \rangle \sum_{n=1}^{N} \langle s_{nm} \rangle \langle \mathbf{x}_n | m \rangle (t_{nk} - \langle \mu_k \rangle) \tag{33}$$

$$\boldsymbol{\Sigma}_{\mathbf{w}}^{(m)} = \left( \mathrm{diag} \langle \boldsymbol{\alpha}_m \rangle + \langle \tau \rangle \sum_{n=1}^{N} \langle s_{nm} \rangle \langle \mathbf{x}_n \mathbf{x}_n^{\mathrm{T}} | m \rangle \right)^{-1} \tag{34}$$

$$\widetilde{a}_{\alpha} = a_{\alpha} + \frac{d}{2} \qquad \widetilde{b}_{\alpha}^{(mj)} = b_{\alpha} + \frac{\langle \|\mathbf{w}_{mj}\|^2 \rangle}{2} \qquad \widetilde{a}_{\tau} = a_{\tau} + \frac{Nd}{2} \tag{35}$$

$$\widetilde{b}_{\tau} = b_{\tau} + \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{M} \langle s_{nm} \rangle \{ \|\mathbf{t}_n\|^2 + \langle \|\boldsymbol{\mu}_m\|^2 \rangle + \mathrm{Tr}(\langle \mathbf{W}_m^{\mathrm{T}} \mathbf{W}_m \rangle$$

$$\langle \mathbf{x}_n \mathbf{x}_n^{\mathrm{T}} | m \rangle). \qquad + 2 \langle \boldsymbol{\mu}_m^{\mathrm{T}} \rangle \langle \mathbf{W}_m \rangle \langle \mathbf{x}_n | m \rangle - 2 \mathbf{t}_n^{\mathrm{T}} \langle \mathbf{W}_m \rangle \langle \mathbf{x}_n | m \rangle - 2 \mathbf{t}_n^{\mathrm{T}} \langle \boldsymbol{\mu}_m \rangle \} \tag{36}$$

$$\widetilde{u}^{(m)} = u_m + \sum_{n=1}^{N} \langle s_{nm} \rangle \tag{37}$$

$$\ln Q(\mathbf{s}_n = \delta_m) = \langle \ln \pi_m \rangle - \frac{1}{2} \langle \mathbf{x}_n^{\mathrm{T}} \mathbf{x}_n | m \rangle - \frac{1}{2} \langle \tau \rangle \left\{ \|\mathbf{t}_n\|^2 + \langle \|\boldsymbol{\mu}_m\|^2 \rangle \right.$$

$$+ \mathrm{Tr}(\langle \mathbf{W}_m^{\mathrm{T}} \mathbf{W}_m \rangle \langle \mathbf{x}_n \mathbf{x}_n^{\mathrm{T}} | m \rangle) + 2 \langle \boldsymbol{\mu}_m^{\mathrm{T}} \rangle \langle \mathbf{W}_m \rangle \langle \mathbf{x}_n | m \rangle \tag{38}$$

$$\left. - 2 \mathbf{t}_n^{\mathrm{T}} \langle \mathbf{W}_m \rangle \langle \mathbf{x}_n | m \rangle - 2 \mathbf{t}_n^{\mathrm{T}} \langle \boldsymbol{\mu}_m \rangle \right\} + \frac{1}{2} \ln |\boldsymbol{\Sigma}_{\mathbf{x}}^{(m)}| + \mathrm{const.} \tag{39}$$

where $\mathrm{diag}\langle\boldsymbol{\alpha}\rangle$ denotes a diagonal matrix whose diagonal elements are given by $\langle\alpha_i\rangle$. The constant in $\ln Q(\mathbf{s}_n = \delta_n)$ is found simply by summing and normalizing. Note also that $\langle\mathbf{x}_n|m\rangle$ denotes an average with respect to $Q(\mathbf{x}_n|\mathbf{s}_n = \delta_m)$.

The solution for the optimal factors in the $Q(\boldsymbol{\theta})$ distribution is, of course, an implicit one since each distribution depends on moments of the other distributions. We can find a solution numerically by starting with a suitable initial guess for the distributions and then cycling through the groups of variables in turn, re-estimating each distribution using the above results. The required moments are easily evaluated using the standard results for normal and Gamma distributions.

Our framework also permits a direct evaluation of the posterior distribution over the number $M$ of components in the mixture (assuming a suitable prior distribution, for example a uniform distribution up to some maximum value). However, in order to reduce the computational complexity of the inference problem we adopt an alternative approach based on model comparison using the numerically evaluated lower bound $\mathcal{L}(Q)$ which approximates the log model probability $\ln P(V)$. Our optimization mechanism dynamically adapts the value of $M$ through a scheme involving the addition and deletion of components [16,8].

One of the limitations of fitting conventional Gaussian mixture models by maximum likelihood is that there are singularities in the likelihood function in which a component's mean coincides with one of the data points while its covariance shrinks to zero. Such singularities do not arise in the Bayesian framework due to the implicit integration over model parameters.

## 4 Results

In order to demonstrate the operation of the algorithm, we first explore its behaviour using synthetic data. The example on the left of Figure 4 shows synthetic data in two

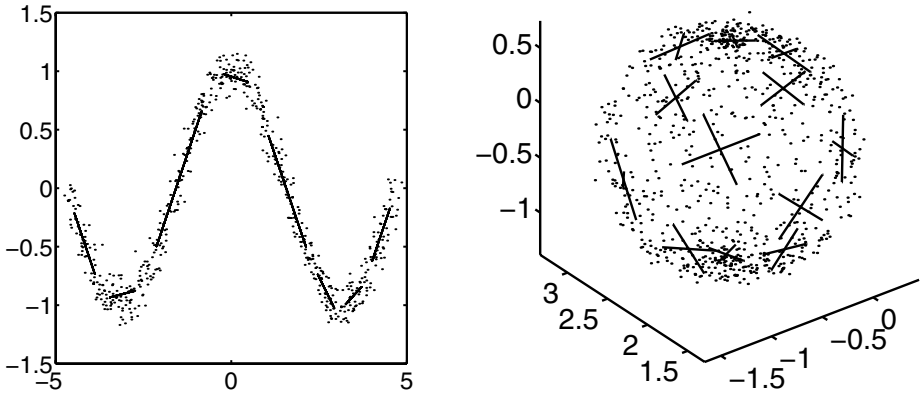


**Fig. 4.** Examples of Bayesian PCA mixture models fitted to synthetic data.

dimensions, together with the result of fitting a Bayesian PCA mixture model. The lines represent the non-zero principal directions of each component in the mixture. At convergence the model had 8 components, having a common effective dimensionality of 1. The right hand plot in Figure 4 shows synthetic data from a noisy 2-dimensional sphere in 3 dimensions together with the converged model, which has 12 components having effective dimensionality of 2. Similar results with synthetic data are robustly obtained when embedding low-dimensional non-linear manifolds in spaces of higher dimensionality.

We now apply our framework to the problem of modelling the manifold of a data set of face images. The data used is a combination of images from the Yale face database and the University of Stirling database. The training set comprises 276 training images, which have been cropped, subsampled to $26 \times 15$, and normalized pixelwise to zero mean and unit variance. A test set consisting of a further 100 face images, together with 200 non-face images, taken from the Corel database, all of which were pre-processed in the same way as the training data.

The converged Bayesian PCA mixture model has 4 components, having a common dimensionality of 5, as emphasized by the Hinton diagram of the shared $\boldsymbol{\alpha}$ hyper-parameters shown in Figure 5.

**Fig. 5.** Hinton diagram showing the inverses of the $\boldsymbol{\alpha}$ hyper-parameters (corresponding to the variances of the principal components) indicating a manifold of intrinsic dimensionality 5.

In order to see how well the model has captured the manifold we first run the model generatively to give some sample synthetic images, as shown in Figure 6. Synthetic faces

**Fig. 6.** Synthetic faces obtained by running the learned mixture distribution generatively.

generated from a single probabilistic PCA model are less noticeably distinct.

We can quantify the extent to which we have succeeded in modelling the manifold of faces by using the density model to classify the images in the test set as faces versus non-faces. To do this we evaluate the density under the model for each test image and if this density exceeds some threshold the image is classified as a face. The threshold value determines the trade-off between false negatives and false positives, leading to an ROC curve, as shown in Figure 7. For comparison we also show the corresponding ROC

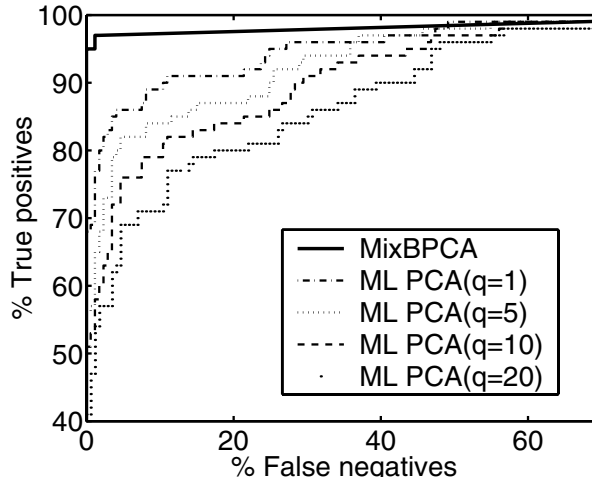


**Fig. 7.** ROC curves for classifying images as faces versus non-faces for the Bayesian PCA mixture model, together with the corresponding results for a maximum likelihood PCA model with various values for $q$ the number of retained principal components. This highlights the significant improvement in classification performance in going from a linear PCA model to a non-linear mixture model.

curves for a single maximum likelihood PCA model for a range of different $q$ values. We see that moving from a linear model (PCA) to a non-linear model (a Bayesian PCA mixture) gives a significant improvement in classification performance. This result also highlights the fact that the Bayesian approach avoids the need to set parameter values such as $q$ by exhaustive exploration.

As a second application of our framework we model the manifolds of images of handwritten digits. We use a data set taken from the CEDAR U.S. Postal Service database, and comprising 11,000 images (equally distributed over the ten digits) each of which is $8 \times 8$ grayscale, together with a similar independent test set of 2711 images. Synthetic images generated from a Bayesian PCA mixture model fitted to the training set are shown in Figure 8.

The learned model achieved 4.83% error rate on the test set. For comparison we note that Tipping and Bishop [13] used the same training and test sets with a maximum likelihood mixture of probabilistic principal component analysers. The training set in

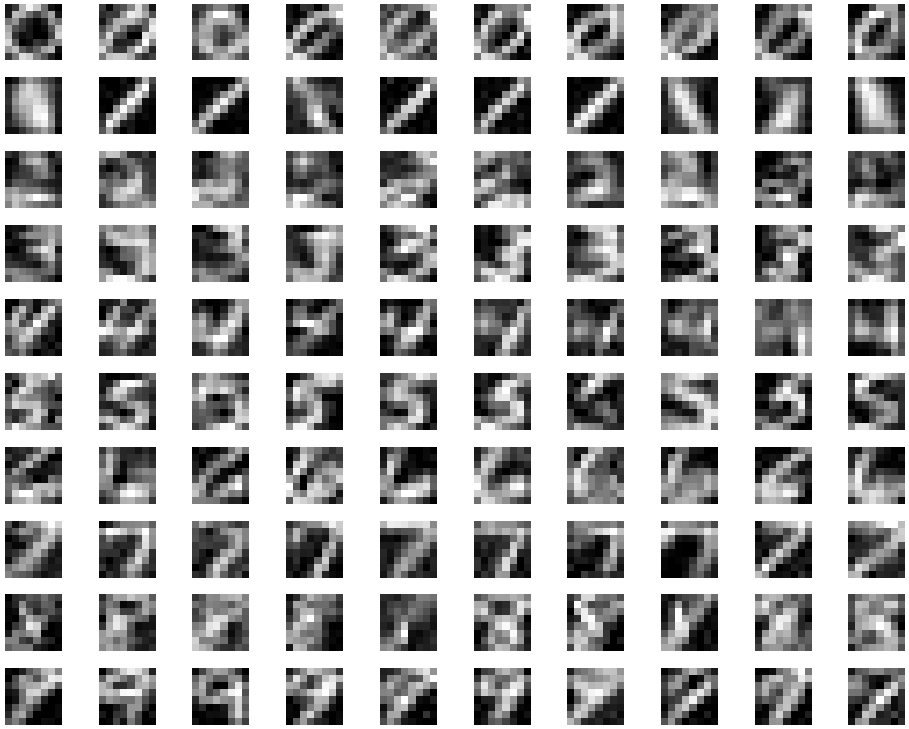

**Fig. 8.** Digits synthesized from each of the ten trained Bayesian PCA mixture model by running the models generatively.

this case was itself subdivided into training plus validation sets. For each of the ten digit models considerable computational effort was expended in finding the optimum values of $M$ (the number of components in the mixture) and $q$ (the dimensionality of the latent spaces) by evaluation of performance on the validation set. This approach achieved 4.61% error rate on the test set, which is comparable with the result obtained from the single run of the Bayesian PCA mixture model.

## 5 Discussion

In this paper we have introduced a fully probabilistic approach to modelling the manifolds of images in which an appropriate model complexity, as well as the manifold intrinsic dimensionality, can be inferred automatically from the data. Preliminary results on data sets of face images and hand-written digits demonstrate both the practical feasiblity of the framework as well as improved performance compared to previous approaches.

An important advantage of our framework is that there are no significant adjustable parameters in the model to be set by the user. The model complexity is inferred from

the data, and since no model optimization is required the model can be run once on the training data, without the need for computationally intensive cross-validation.

**Acknowledgements**

We are grateful to Andrew Blake, Phil Torr and Colin Davidson for useful discussions and helpful advice throughout this project.

## References

1. H. Attias. Learning parameters and structure of latent variables by variational Bayes, 1999. Submitted to UAI.
2. C. M. Bishop. Bayesian PCA. In S. A. Solla M. S. Kearns and D. A. Cohn, editors, *Advances in Neural Information Processing Systems*, volume 11, pages 382–388. MIT Press, 1999.
3. C. M. Bishop and M. E. Tipping. A hierarchical latent variable model for data visualization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):281–293, 1998.
4. M. J. Black and Y. Yacoob. Recognizing facial expressions under rigid and non-rigid facial motions. In *International Workshop on Automatic Face and Gesture Recognition, Zurich*, pages 12–17, 1995.
5. C. Bregler and S.M. Omohundro. Nonlinear manifold learning for visual speech recognition. In *Fifth International Conference on Computer Vision*, pages 494–499, Boston, Jun 1995.
6. T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models — their training and application. In *Computer vision, graphics and image understanding*, volume 61, pages 38–59, 1995.
7. B. Frey and N. Jojic. Transformed component analysis: joint estimation of spatial transformations and image components. In *Seventh International Conference on Computer Vision*, pages 1190–1196, 1999.
8. Z. Ghahramani and M. J. Beal. Variational inference for Bayesian mixture of factor analysers. In *Advances in Neural Information Processing Systems*, volume 12, 1999.
9. T. Heap and D. Hogg. Wormholes in shape space: Tracking through discontinuous changes in shape. In *Sixth International Conference on Computer Vision*, pages 344–349, 1998.
10. M. I. Jordan, Z. Gharamani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 105–162. Kluwer, 1998.
11. B. Moghaddam. Principal manifolds and Bayesian subspaces for visual recognition. In *Seventh International Conference on Computer Vision*, pages 1131–1136, 1999.
12. B. Moghaddam and A. Pentland. Probabilistic visual learning for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):696–710, 1997.
13. M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11(2):443–482, 1999.
14. M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B*, 21(3):611–622, 1999.
15. M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
16. N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton. SMEM algorithm for mixture models. In *Advances in Neural Information Processing Systems*, volume 11, 1999.