Hidden Markov Bayesian Principal Component Analysis

M. Alvarez malvarez@utp.edu.co
Universidad Tecnológica de Pereira
Pereira, Colombia

R. Henao rhenao@utp.edu.co
Universidad Tecnológica de Pereira
Pereira, Colombia

Abstract

Probabilistic Principal Component Analysis is a reformulation of the common multivariate analysis technique known as Principal Component Analysis. It employs a latent variable model framework similar to factor analysis allowing to establish a maximum likelihood solution for the parameters that comprise the model. One of the main assumptions of Probabilistic Principal Component Analysis is that observed data is independent and identically distributed. This assumption is inadequate for many applications, in particular, for modeling sequential data. In this paper, the authors introduce a temporal version of Probabilistic Principal Component Analysis by using a hidden Markov model in order to obtain optimized representations of observed data through time. Combining Probabilistic Principal Component Analyzers with a hidden Markov model, it is possible to enhance the capabilities of transformation and reduction of time series vectors. In order to find automatically the dimensionality of the principal subspace associated with these Probabilistic Principal Component Analyzers through time, a Bayesian treatment of the Principal Component model is introduced as well.

Keywords:
Hidden Markov models, principal component analysis, bayesian principal component analysis, EM algorithm, model selection

1. Introduction

Principal Component Analysis (PCA) is a popular and powerful technique for feature extraction, dimensionality reduction and probably the most employed of the techniques of multivariate analysis (Jolliffe, 2002b). One of the most common definitions of PCA is that, for a set of observed d-dimensional data vectors $\mathbf{T} = \{\mathbf{t}_n\}_{n=1}^N$, the p principal axes $\mathbf{W} = \{\mathbf{w}_i\}_{i=1}^p$ are those orthonormal axes onto which the retained variance under linear projection is maximal.

However, PCA has several disadvantages, among them, the absence of an associated probability density or generative model and the assumption that observed data is independent and identically distributed (i.i.d.), when modeling time series. The first disadvantage was overcame by Probabilistic Principal Component Analysis (PPCA) (Tipping and Bishop, 1999). PPCA is a probabilistic formulation of a latent variable model with isotropic Gaussian error model, and the maximum likelihood solution for its parameters

indeed corresponds to principal component analysis (Tipping and Bishop, 1999). In this framework, a computational efficient EM algorithm can be derived for the estimation of the PPCA model. At the same time, PPCA can be used to model class-conditional densities and, within Bayes decision theory, to be applied for classification tasks.

About the second disadvantage, this constitutes a non trivial problem because for time series, perhaps the most common type of non-independent data, even a very weak dependence relation between the data makes PCA unappropriate. Several techniques have been developed to exploit the temporal dependencies in order to optimize the representation of the data from a temporal context (Voegtlin, 2005, Ku et al., 1995, Jolliffe, 2002a, Shumway and Stoffer, 2005).

To overcome both disadvantages at the same time, in this paper, the authors introduce a temporal version of Probabilistic Principal Component Analysis by using a hidden Markov model (HMM) in order to obtain optimized representations of observed data through time. Replacing the standard Gaussian mixture model employed as the observation model for the HMM (Huang et al., 2001) with a PPCA model, every data point has an associated local representation corresponding to the most probable state produced by the trained model. This hidden Markov Principal Component Analysis (HMPCA) allows to enhance the capabilities of transformation and reduction of time series vectors. Equally, by exploiting the probabilistic formulation of the PCA model, a Bayesian treatment of the PPCA model (Bishop, 1999) is applied in order to find automatically the dimensionality of the principal subspace associated with these Probabilistic Principal Component Analyzers through time.

The paper is organized as follows: section 2 and section 3 introduces the Hidden Markov Principal Component Analysis Model (HMPCA) and the Hidden Markov Bayesian Principal Component Analysis Model (HMBPCA), respectively. Section 4 contains implementation issues. In section 5 experimental results for synthetic and real data are reported and finally in section 6 the discussion.

2. Hidden Markov Principal Component Analysis

A hidden Markov model is basically a Markov chain where the output observation is a random variable generated according to an output probabilistic function associated with each state (Huang et al., 2001). Formally, a hidden Markov model of Q states is defined by

- $\boldsymbol{\pi}$. Initial state distribution with elements $\pi_k = p(q_0 = k),\ 1 \leq k \leq Q$.

- $\mathbf{A}$. Transition probability matrix which entries a_{ij} denotes the probability of taking a transition from state k to state j, i.e., $a_{kj} = p(q_n = j | q_{n-1} = k),\ n = 1, \ldots, N$.

- $\boldsymbol{\Theta}$. Parameters of an output probability density function $p(\mathbf{t}_n | q_n, \boldsymbol{\Theta})$, where $\mathbf{T} = \{\mathbf{t}_n\}_{n=1}^N$, $\mathbf{t}_n \in \mathbb{R}^d$ is a sequence of observations and $\mathbf{q} = \{q_n\}_{n=1}^N$ is a sequence of states that is not observed.

For simplicity, the set of parameters is denoted as $\widetilde{\boldsymbol{\Lambda}} = (\widetilde{\boldsymbol{\pi}}, \widetilde{\mathbf{A}}, \widetilde{\boldsymbol{\Theta}})$. In an HMM parameterized by some vector $\widetilde{\boldsymbol{\Lambda}}$, the observed variable $\mathbf{t}_n$ depends only on the current state q_n (and not previous states) and the current state depends only on the previous state (and

not on states before that). This allows the joint log likelihood of an observation sequence $\mathbf{T}$ and hidden variable (state) sequence $\mathbf{q}$ to be written as

$$\log p_{\widetilde{\mathbf{\Lambda}}}(\mathbf{T}, \mathbf{q}) = \log p_{\widetilde{\mathbf{\Lambda}}}(q_1) + \sum_{n=2}^{N} \log p_{\widetilde{\mathbf{\Lambda}}}(q_n|q_{n-1}) + \sum_{n=1}^{N} \log p_{\widetilde{\mathbf{\Lambda}}}(\mathbf{t}_n|q_n) \tag{1}$$

A standard procedure for maximizing (1) uses the Expectation-Maximization (EM) algorithm. This algorithm maximizes $Q(\widetilde{\mathbf{\Lambda}}, \mathbf{\Lambda})$, the expectation of the joint log likelihood taken w.r.t. the old distribution of hidden state variables, $p_{\mathbf{\Lambda}}(\mathbf{q}|\mathbf{T})$. In this way,

$$\begin{aligned} Q(\widetilde{\mathbf{\Lambda}}, \mathbf{\Lambda}) = &\sum_{\mathbf{q}} p_{\mathbf{\Lambda}}(\mathbf{q}|\mathbf{T}) \log p_{\widetilde{\mathbf{\Lambda}}}(q_1) + \sum_{\mathbf{q}} p_{\mathbf{\Lambda}}(\mathbf{q}|\mathbf{T}) \sum_{n=2}^{N} \log p_{\widetilde{\mathbf{\Lambda}}}(q_n|q_{n-1}) \\ &+ \sum_{\mathbf{q}} p_{\mathbf{\Lambda}}(\mathbf{q}|\mathbf{T}) \sum_{n=1}^{N} \log p_{\widetilde{\mathbf{\Lambda}}}(\mathbf{t}_n|q_n) \end{aligned} \tag{2}$$

where $\sum_{\mathbf{q}}$ denotes a sum over all posible hidden sequences. The above equation consists of three terms; the first for the initial probability parameters, $\boldsymbol{\pi}$, the second for the state transition probability parameters, $\mathbf{A}$ and the third for the observation model parameters $\mathbf{\Theta}$

$$Q(\widetilde{\mathbf{\Lambda}}, \mathbf{\Lambda}) = Q(\widetilde{\boldsymbol{\pi}}, \boldsymbol{\pi}) + Q(\widetilde{\mathbf{A}}, \mathbf{A}) + Q(\widetilde{\mathbf{\Theta}}, \mathbf{\Theta}) \tag{3}$$

These can be maximized separately giving rise to parameter update equations for different parameters of the model. The third term in equations (2) and (3) can be rearranged as

$$Q(\widetilde{\mathbf{\Theta}}, \mathbf{\Theta}) = \sum_{i} \sum_{n} p_{\mathbf{\Lambda}}(q_n|\mathbf{t}_n) \log p_{\widetilde{\mathbf{\Lambda}}}(\mathbf{t}_n|q_n) \tag{4}$$

In order to obtain $p_{\widetilde{\mathbf{\Lambda}}}(\mathbf{t}_n|q_n)$, it is assumed, for the observation model, that the variable $\mathbf{t}$ is defined by a linear transformation of the p- dimensional latent variable $\mathbf{x}$ plus additive Gaussian noise, so that

$$\mathbf{t} = \mathbf{W}\mathbf{x} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$$

where $p(\mathbf{x}) \sim \mathcal{N}(0, \mathbf{I})$ and $p(\epsilon) \sim \mathcal{N}(0, \sigma^2\mathbf{I})$. With this model, the conditional distribution of the observed variable given the latent variable is of the form,

$$p(\mathbf{t}|\mathbf{x}, \mathbf{W}, \boldsymbol{\mu}, \sigma^2) \sim \mathcal{N}(\mathbf{W}\mathbf{x} + \boldsymbol{\mu}, \sigma^2\mathbf{I})$$

and the marginal distribution is given as

$$p(\mathbf{t}|\boldsymbol{\theta}) = \int p(\mathbf{t}|\mathbf{x}, \boldsymbol{\theta}) p(\mathbf{x}) d\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$$

where $\boldsymbol{\theta} = \{\boldsymbol{\mu}, \sigma^2, \mathbf{W}\}$ and $\mathbf{C} = \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}$. For a mixture model, the marginal distribution is written as

$$p(\mathbf{t}|\mathbf{\Theta}) = \sum_{i=1}^{M} c_i p(\mathbf{t}|\boldsymbol{\theta}_i)$$

where the mixture coefficients follow the restrictions $\sum_i c_i = 1$ and $0 \leq c_i \leq 1,\ \forall i$ and we have defined $\mathbf{\Theta} = \{\boldsymbol{\theta}_i\}_{i=1}^{M}$. Using this mixture model as the observation density in the HMM framework, we can write

$$p(\mathbf{t}_n|q_n = k, \mathbf{\Theta}) = \sum_{i=1}^{M} c_{ik} p(\mathbf{t}_n|\boldsymbol{\theta}_{ik}) = \sum_{i=1}^{M} c_{ik} p(\mathbf{t}_n|\boldsymbol{\mu}_{ik}, \sigma_{ik}^2, \mathbf{W}_{ik})$$

It can be shown that the reestimation formulas for $\boldsymbol{\theta}_{ik}$ and c_{ik} are given as

$$\widetilde{\mathbf{W}}_{ik} = \left[\sum_{n=1}^{N} \gamma_{nik}(\mathbf{t}_n - \widetilde{\boldsymbol{\mu}}_{ik})\langle \mathbf{x}_{nik}\rangle^\top\right]\left[\sum_{n=1}^{N} \gamma_{nik}\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle\right]^{-1} \tag{5}$$

$$\widetilde{\sigma}_{ik}^2 = \left[d\sum_{n=1}^{N}\gamma_{nik}\right]^{-1}\sum_{n=1}^{N}\gamma_{nik}\Bigg[\|\mathbf{t}_n - \widetilde{\boldsymbol{\mu}}_{ik}\| - 2\langle \mathbf{x}_{nik}\rangle^\top \widetilde{\mathbf{W}}_{ik}^\top(\mathbf{t}_n - \widetilde{\boldsymbol{\mu}}_{ik}) + \mathrm{tr}(\widetilde{\mathbf{W}}_{ik}^\top\widetilde{\mathbf{W}}_{ik}\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle)\Bigg] \tag{6}$$

$$\widetilde{\boldsymbol{\mu}}_{ik} = \left[\sum_{n=1}^{N}\gamma_{nik}\right]^{-1}\left[\sum_{n=1}^{N}\gamma_{nik}(\mathbf{t}_n - \widetilde{\mathbf{W}}_{ik}\langle \mathbf{x}_{nik}\rangle)\right] \tag{7}$$

$$\widetilde{c}_{ik} = \frac{1}{N}\sum_{n=1}^{N}\gamma_{nik} \tag{8}$$

where

$$\gamma_{nik} = p(q_n = k|\mathbf{T}, \mathbf{\Lambda})\frac{c_{ik}p(\mathbf{t}_n|\boldsymbol{\theta}_{ik})}{\sum_{j=1}^{M} c_{jk}p(\mathbf{t}_n|\boldsymbol{\theta}_{jk})} \tag{9}$$

$$\langle \mathbf{x}_{nik}\rangle = \mathbf{M}_{ik}^{-1}\mathbf{W}_{ik}^\top(\mathbf{t}_n - \boldsymbol{\mu}_{ik}) \tag{10}$$

$$\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle = \sigma_{ik}^2\mathbf{M}_{ik}^{-1} + \langle \mathbf{x}_{nik}\rangle\langle \mathbf{x}_{nik}\rangle^\top \tag{11}$$

with $\mathbf{M}_{ik} = \mathbf{W}_{ik}^\top\mathbf{W}_{ik} + \sigma_{ik}^2\mathbf{I}$. Quantity γ_{nik} can be efficiently calculated using Forward-Backward algorithm (Rabiner, 1989). Reestimation equations for $\widetilde{\boldsymbol{\pi}}$ and $\widetilde{\mathbf{A}}$ are the same that for a hidden Markov model (Huang et al., 2001).

3. Hidden Markov Bayesian Principal Component Analysis

Once the HMPCA model has been defined, we employ a Bayesian treatment of the Probabilistic Principal Component Analyzer, that we have used as the observation model in the hidden Markov chain, in order to find automatically the dimensionality of the principal

subspace. With an appropriate choice for the prior over $\mathbf{W}_{ik}$, it would be possible to prune noise dimensions out of the principal subspace using an automatic relevance determination framework (ARD) (MacKay, 1992). In particular, a Gaussian prior is defined over each column of $\mathbf{W}_{ik}$,

$$p(\mathbf{W}_{ik}|\boldsymbol{\beta}_{ik}) = \prod_{j=1}^{p} \left(\frac{\beta_{ikj}}{2\pi}\right)^{d/2} \exp\left\{-\frac{1}{2}\beta_{ikj}\mathbf{w}_{ikj}^{\top}\mathbf{w}_{ikj}\right\}$$

where $\mathbf{w}_{ikj}$ is the j^{th} column of $\mathbf{W}_{ik}$ and β_{ikj} are precision hyperparameters. The values of β_{ikj} are reestimated during training maximizing the marginal likelihood that is now given by

$$p(\mathbf{t}|\boldsymbol{\mu}_{ik}, \sigma_{ik}^2, \boldsymbol{\beta}_{ik}) = \int p(\mathbf{t}|\boldsymbol{\mu}_{ik}, \sigma_{ik}^2, \mathbf{W}_{ik}) p(\mathbf{W}_{ik}|\boldsymbol{\beta}_{ik}) d\mathbf{W}_{ik} \tag{12}$$

This integral is intractable. There are several techniques that could be used to approximate it, such as variational inference or sampling methods. Here we use a Laplace approximation. First, we assume that the posterior distribution $p(\mathbf{W}_{ik}|\mathbf{t}, \boldsymbol{\beta}_{ik})$ can be approximated using a Gaussian density function. The evidence term $p(\mathbf{t}|\boldsymbol{\mu}_{ik}, \sigma_{ik}^2, \boldsymbol{\beta}_{ik})$ in equation (12) is obtained then as the normalizing factor of the approximated Gaussian. The distribution $p(\mathbf{t}|\boldsymbol{\mu}_{ik}, \sigma_{ik}^2, \boldsymbol{\beta}_{ik})$ is maximized w.r.t. $\boldsymbol{\beta}_{ik}$, leading to

$$\widetilde{\beta}_{ikj} = \frac{d - \beta_{ikj}\mathrm{tr}_j(\mathbf{H}_{ik})}{\widetilde{\mathbf{w}}_{ikj}^{\top}\widetilde{\mathbf{w}}_{ikj}} \tag{13}$$

where $\mathbf{H}_{ik}$ is the Hessian matrix given by the second derivatives of $\ln p(\mathbf{W}_{ik}|\mathbf{t}, \boldsymbol{\beta}_{ik})$ with respect to the elements of $\mathbf{W}_{ik}$ (evaluated at $\mathbf{W}_{ik}^{\mathrm{MP}}$ which is the mode of $\ln p(\mathbf{W}_{ik}|\mathbf{t}, \boldsymbol{\beta}_{ik})$) and $\mathrm{tr}_j(\cdot)$ denotes the trace of the sub-matrix corresponding to the vector $\widetilde{\mathbf{w}}_{ikj}$ (Bishop, 1999). For the estimation of the parameters $\boldsymbol{\theta}_{ik}$, the only change is in equation (5), which is modified as

$$\widetilde{\mathbf{W}}_{ik} = \left[\sum_{n=1}^{N} \gamma_{nik}(\mathbf{t}_n - \widetilde{\boldsymbol{\mu}}_{ik})\langle \mathbf{x}_{nik}\rangle^{\top}\right] \left[\sum_{n=1}^{N} \gamma_{nik}\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^{\top}\rangle + \sigma_{ik}^2\mathbf{B}_{ik}\right]^{-1} \tag{14}$$

where $\mathbf{B}_{ik} = \mathrm{diag}(\beta_{ikj})$. The effective dimensionality of the principal subspace associated with each latent variable model is determined by the number of finite β_{ikj} values.

4. Implementation Issues

Initialization For parameter initialization we employ the *K-means* algorithm. A PCA model is applied to each data cluster and with a random value of σ_{ik}^2, initial guesses for $\boldsymbol{\theta}_{ik}$ are given. Using these initial parameters, a PPCA model or a BPCA model is trained for each cluster and these models are then associated with the states of the HMPCA model or HMBPCA model, respectively.

Expected Sufficient Statistics Update equations (5),(6), (7) and (8) are in terms of the raw data, so their calculation is dependent of the number N of observations. If N is large, direct implementation of the above formulas might be computational expensive. This is particular true for sequential learning problems where several observation sequences are available and the sums of all above equations must be made over sequences as well. In order to solve this problem, we follow (Murphy, 2003) and formulate estimates for $\boldsymbol{\theta}_{ik}$ in terms of expected sufficient statistics, whose size is independent of the number of samples (see appendix C, equations (33),(34), (35) and (36)).

Approximation of hyperparameters The numerator of $\widetilde{\beta}_{ikj}$ in equation (13) can be interpreted as a measure of how "well-determined" its corresponding parameter vector $\widetilde{\mathbf{w}}_{ikj}$ is by the data (MacKay, 1992). Assuming that all parameters are "well-determined", numerator in equation (13) is approximated with d. This simplification reduces the computational cost since the evaluation of the Hessian matrix is no longer necessary.

Pruning During training, vectors $\widetilde{\mathbf{w}}_{ikj}$ for which there is insufficient support from the data will be driven to zero, with the corresponding $\widetilde{\beta}_{ikj} \to \infty$. To keep the rank of the matrix $\widetilde{\mathbf{W}}_{ik}$ and in order to accelerate the training process, vectors $\widetilde{\mathbf{w}}_{ikj}$ that are zero, are switched off during the reestimation process.

Algorithm For illustrative purposes, algorithm 1 describes the complete training procedure. It should be noticed that for HMPCA there is no need to initialize/reestimate $\mathbf{B}$ nor perform pruning. Besides, equation (5) should be replaced by (14).

Algorithm 1 HMPCA/HMBPCA Algorithms

Require: Training observation $\mathbf{T}$, Q and M
Initialize $\boldsymbol{\Lambda}$, $\mathbf{B}_{ik}$ using *K-means*+BPCA
repeat
  % E step
  Reestimate γ_{nik} using Forward-Backward algorithm, equation (9)
  Calculate $\langle \mathbf{x}_{nik} \rangle$ and $\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^{\top} \rangle$ using equations (10) and (11)
  % M step
  Reestimate $\widetilde{\boldsymbol{\theta}}_{ik}$, $\widetilde{c}_{ik}$, $\widetilde{\mathbf{B}}_{ik}$ using equations (14), (6), (7),(8) and (13)
  Prune $\widetilde{\mathbf{W}}_{ik}$ based on $\widetilde{\mathbf{B}}_{ik}$
until convergence
return Trained model parameters $\boldsymbol{\Lambda}$, $\boldsymbol{\Theta}$ and $\mathbf{B}$

5. Results

5.1 Density modeling and dimensionality recovering

In this first experiment, the objective is to show that the algorithm is able to model and to recover the dimensionality of data. For this purpose, a dataset was generated consisting in 50 observations of length 100 and dimension 10 drawn from a 3 state hidden Markov chain with, uniform distributed transition matrix, uniform initial states probability vector and zero mean gaussian observation models with diagonal covariance matrices. The covariance

matrices are $\mathbf{C}_1 = \text{diag}(2 \times \mathbf{1}_5, 0.1 \times \mathbf{1}_5)$, $\mathbf{C}_2 = \text{diag}(4 \times \mathbf{1}_2, 0.1 \times \mathbf{1}_8)$ and $\mathbf{C}_3 = \text{diag}(1 \times \mathbf{1}_8, 0.1 \times \mathbf{1}_2)$, where $\mathbf{C}_k$ stands for the covariance matrix associated with state k and $\mathbf{1}_m$ is a m dimensional vector of ones. The results after training HMPCA and HMBPCA models[1] are shown in figure 1 by means of Hinton diagrams over resulting $\mathbf{W}$ matrices. For this case, HMBPCA have found that the dimensionality of the data is 5, 2 and 8 which is consistent with the true assumed values. Additionally, given that the data was generated randomly the whole procedure was repeated 50 times and its was found that HMBPCA choose the true dimensionality 72% of the times (particularly 2:74%, 5:100% and 8:72%).

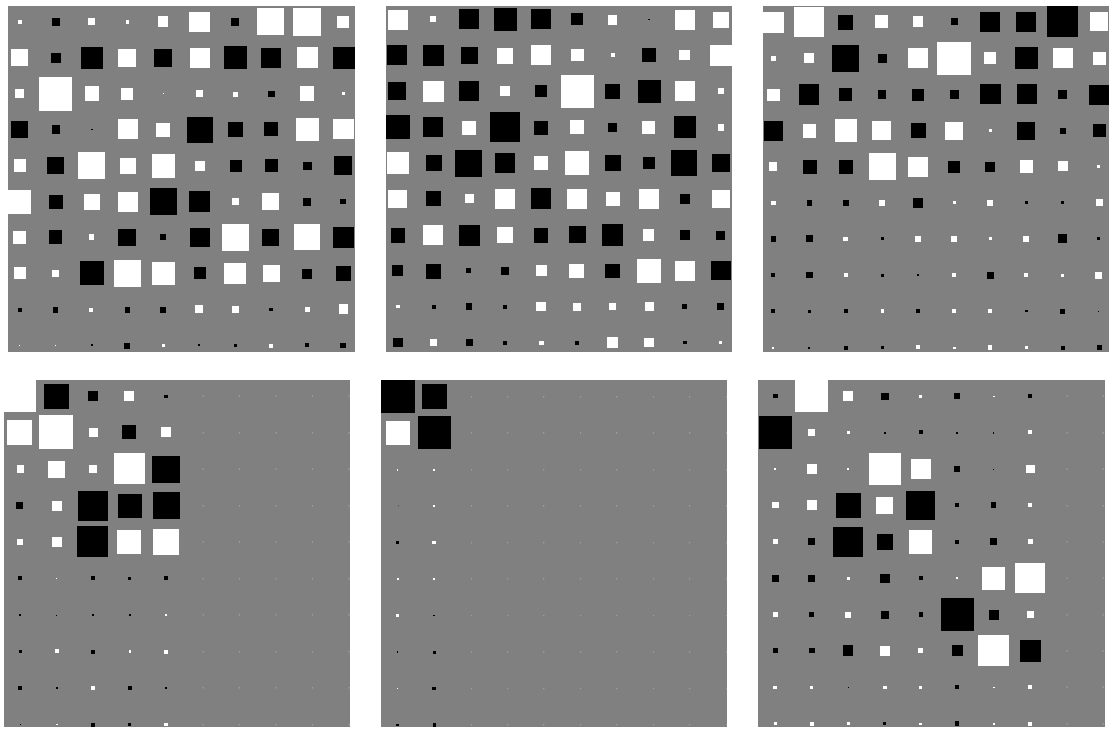

Figure 1: Hinton diagrams of the matrices $\mathbf{W}$ corresponding to each one of the three states estimated model. The top plots are for HMPCA while the bottom plots are for HMBPCA. In the HMBPCA case, the model is clearly able to keep the true dimensionality (5,2 and 8) of the synthetic data

Since this is a case of density model estimation, in figure 2 are shown both, the true eigenvalues of the synthetic data and the eigenvalues from the estimated covariance matrices from a three state HMM, HMPCA and HMBPCA models. In addition, in table 1 it is shown the mean square errors (MSE) calculated between original parameters and those estimated by the models.

From table 1 it can be seen that both HMPCA and HMBPCA are able to capture with a little error, the parameters of the synthetic model in a similar way as HMM does.

1. Algorithms were ran for a number of iterations fixed to 100

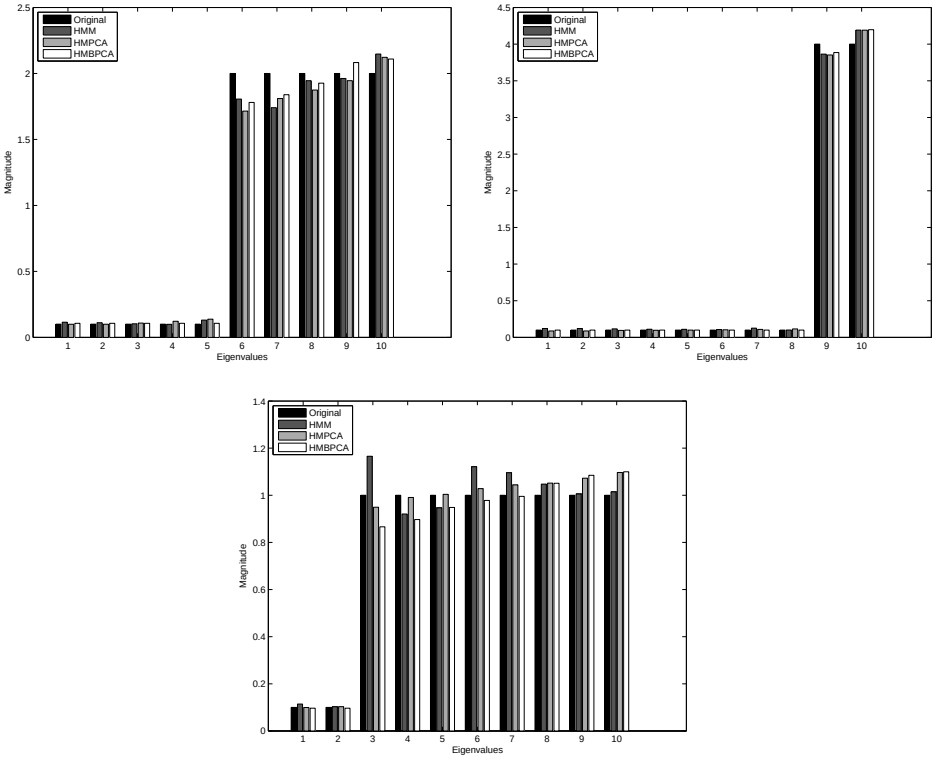

Figure 2: Eigenvalues from true covariance matrices and those estimated by three state HMM, HMPCA and HMBPCA models. Covariance matrices are designed to have 5, 2 and 8 effective dimensions out of 10

Method	$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$	$\boldsymbol{\mu}_1$	$\boldsymbol{\mu}_2$	$\boldsymbol{\mu}_3$
HMM	0.063	0.057	0.132	0.010	0.014	0.009
HMPCA	0.153	0.058	0.022	0.026	0.048	0.155
HMBPCA	0.098	0.052	0.051	0.005	0.016	0.003

Table 1: Mean square errors obtained comparing the true parameters and the ones obtained after training HMM, HMPCA and HMBPCA models. Since estimates $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_3$ are matrices, the MSE was calculated using the its corresponding eigenvalues

5.2 High-dimensional data

In this case, the goal is to gain some insight about the behavior of HMBPCA when the dimensionality of data is high. For this purpose, in the same fashion of the last experiment, 50 sequences of length 100 and dimensionality 100 were generated, for a model containing just two zero mean states with covariances given by $\mathbf{C}_1 = \text{diag}(2 \times \mathbf{1}_5, 0.1 \times \mathbf{1}_{95})$ and

$\mathbf{C}_2 = \text{diag}(1 \times \mathbf{1}_1 0, 0.1 \times \mathbf{1}_{90})$. From figure 3, it is evident that HMBPCA has captured the true dimensionality of data and both $\mathbf{C}_1$ and $\mathbf{C}_2$ depends only on σ_1^2 and σ_2^2 (see table 2).

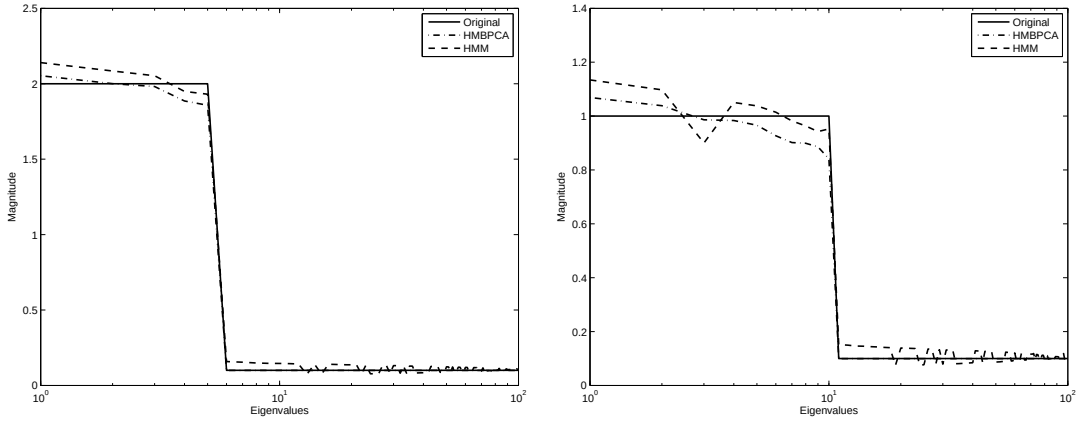

Figure 3: Eigenvalues from true covariance matrices and those estimated by two states HMM and HMBPCA models. Covariance matrices are designed to have 5 and 10 effective dimensions out of 100

Method	$\mathbf{C}_1$	$\mathbf{C}_2$	$\boldsymbol{\mu}_1$	$\boldsymbol{\mu}_2$	σ_1^2	σ_2^2
HMM	0.088	0.095	0.004	0.005	–	–
HMBPCA	0.036	0.071	0.006	0.005	0.997	0.995

Table 2: Mean square errors obtained comparing the true parameters and the ones obtained after training HMM and HMBPCA models. Since estimates $\mathbf{C}_1$ and $\mathbf{C}_2$ are matrices, the MSE was calculated using the its corresponding eigenvalues. Parameters σ_1^2 and σ_2^2 of HMBPCA are also indicated

In table 2 are shown the MSE errors for the parameters estimated using both models, HMM and HMBPCA. It is clear that HMBPCA outperforms HMM because the last one needs to estimate more parameters than HMBPCA, as a consequence of the switched-off dimensions.

5.3 Real world data classification

From a pattern recognition perspective, it is possible to build a classifier using generative models such as HMM, training as many independent models as classes involved and later using a classical Bayes' decision rule. As an illustration, a dataset extracted from European ST-T database consisting in two classes of ECG signals[2] (normal and myocardial ischemia events) was used to build a generative model based classifier. Results in table 3 show the

2. Data is publicly available at: http://www.physionet.org/physiobank/

accuracy and standard deviation as a product of 5-fold crossvalidation scheme also used to find the best parameters Q and M of models. It should be noticed that for the case of HMPCA, the dimension of the principal subspace p should be considered as well.

Method	p	Q	M	Accuracy
HMM	–	3	3	95.00 ± 4.96
HMPCA	6	10	1	96.94 ± 2.85
HMBPCA	–	7	9	98.05 ± 2.70

Table 3: Classification results using ECG dataset. Q is the number of states, M the number of mixtures and p the overall dimensionality of the principal subspace which must be supplied in the case of HMPCA

Results show that HMBPCA outperforms the other two methods in terms of classification accuracy. Selected dimensionalities by HMBPCA are not shown but belongs to a range of 3 to 9. It is clear that HMBPCA requires a more complex model, but since the dimensions on each state and mixture are selected automatically, the model is allowed to incorporate more parameters without the requirement of increasing the dataset size. This is because the number of parameters to be estimated is in average not to far from the needed by a simpler model, and in turn allows it to have a detailed localized representation of observed data, producing better results.

6. Discussion

Results show that HMBPCA is able to determine automatically the effective dimensionality of the observed data even when its dimension is high compared with the dataset size, which provides in turn a practical alternative to exhaustive comparison of dimensionalities using computational expensive techniques such as crossvalidation. In addition, the proposed models, i.e. HMPCA and HMBPCA offer estimations as good as those obtained by a conventional sequence density modeling technique such as HMM. On the other hand, HMBPCA outperforms HMM and HMPCA in terms of the accuracy of the built classifier even when the dimensionalities chosen by the model are low compared with the dimensionality of the dataset, which intuitively could indicate, as in many practical cases that the representation of the signal contains irrelevant or highly correlated features.

From the first experiment it is remarkable that HMBPCA is capable of discovering, in an automatic way, the dimensionalities of the observed data locally, which in fact is a very important property, since it is not to be expected in practice that a complex time series contains the same degrees of freedom everywhere. For instance, ECG signals are more likely to have most of its information near to the central part of the beat (QRS complex) than in the flanks.

When the dimensionality of the data is high and comparable with the dataset size, HMM is not so accurate as HMBPCA due to that, in general terms, it needs to estimate more parameters than HMBPCA because the later eliminates parameters while discarding irrelevant dimensions via pruning and allowing to explain many parameters in the covariance

matrices with single scalar variables σ_{ik}^2. In practice, this is particularly true considering than in high dimensional representations there must be also high amounts of variables that are not informative, selectively relevant on a small set of data or simply background noise, e.g. in handwritten digits images.

From a pattern recognition point of view the dimensionality reduction is good because eliminating irrelevant variables leads in most of the cases to better classifiers, which in this work is done both locally and dynamically. A simple explanation for this phenomenon is that a classifier built using more variables than those strictly necessary is more sensitive to measure/background noise and leads to overfitting.

In computational cost terms, HMBPCA is very efficient when combining the expected sufficient statistics with the pruning step, to accelerate the training process while the unnecessary parameters are eliminated.

As HMM, HMPCA and HMBPCA suffers of two disadvantages, namely, no identifiability and initialization sensitivity. About the later, a possible solution might be to use a mixture of bayesian PCAs instead of *K-means* followed BPCA, mainly due to the limitations of the clustering algorithm, e.g. when the data can be represented as models with means, covariances and dimensionalities that are too similar between states. Equally, since the model uses maximum likelihood and dimensionality reduction via pruning, the likelihood is sensitive to jump between iterations. Albeit this not compromises the convergence of the algorithm, it would be more appropriate to use a convergence criterion non dependent on the dimensionality changes.

Maximum likelihood schemes need high amounts of data and may lead to overtraining, then a possible solution would be to adopt a fully bayesian treatment of HMBPCA based in analytical approximations such as variational or expectation propagation methods. Finally, there are still two parameters that need to be tuned automatically using model selection techniques which are now in fact being investigated.

Acknowledgments

Authors would like to thank to Faculty of Technology and CIE (Centro de Investigaciones y Extensión) under project "Análisis de la variabilidad estocástica en la detección de patologías sobre registros electrocardiográficos y de voz" (contract 6-07-3) for partial support.

References

C.M. Bishop. *Advances in Neural Information Processing Systems*, volume 11, chapter Bayesian PCA, pages 382–388. MIT-Press, 1999.

X. Huang, A. Acero, and H.W. Hon. *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. Prentice Hall, Upper Saddle River. New Jersey, first edition, 2001. ISBN 0-130-22616-5.

I. T. Jolliffe. *Principal Component Analysis*, chapter Principal Component Analysis for time Series and other non Independent data, pages 299–337. Springer Verlag, second edition, 2002a. ISBN 0-387-95442-2.

I.T. Jolliffe. *Principal Component Analysis*. Springer Verlag, second edition, 2002b. ISBN 0-387-98950-1.

W. Ku, R.H. Storer, and C. Georgakis. Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 30(1): 179–196, 1995.

D.J.C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, 1992.

K. Murphy. Fitting a constrained conditional linear gaussian distribution. Technical report, Departments of Computer Science and Statistics, University of British Columbia, 2003.

L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of The IEEE*, 77(2), 1989.

R.H. Shumway and D.S. Stoffer. *Time Series Analysis and Its Applications*, chapter Statistical Methods in Frequency Domain, pages 465–483. Springer Verlag, second edition, 2005. ISBN 0-387-98950-1.

M.E. Tipping and C.M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B*, 21(3):611–622, 1999.

T. Voegtlin. Recursive principal components analysis. *Neural Networks*, 18(8):1040–50, 2005.

Appendix A. Probabilistic Principal Component Analysis

Model:

$$\mathbf{t} = \mathbf{W}\mathbf{x} + \boldsymbol{\mu} + \boldsymbol{\epsilon} \tag{15}$$

Model definition:

$$p(\mathbf{x}) \sim \mathcal{N}(0, \mathbf{I}) \tag{16}$$

$$p(\epsilon) \sim \mathcal{N}(0, \sigma^2\mathbf{I}) \tag{17}$$

Conditional distribution:

$$p(\mathbf{t}|\mathbf{x}, \mathbf{W}, \boldsymbol{\mu}, \sigma^2) \sim \mathcal{N}(\mathbf{W}\mathbf{x} + \boldsymbol{\mu}, \sigma^2\mathbf{I}) \tag{18}$$

Marginal distribution:

$$p(\mathbf{t}|\mathbf{W}, \boldsymbol{\mu}, \sigma^2) = \int p(\mathbf{t}|\mathbf{x}, \mathbf{W}, \boldsymbol{\mu}, \sigma^2)p(\mathbf{x})d\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{C}) \tag{19}$$

where $\mathbf{C} = \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}$.

Latent posterior distribution:

$$p(\mathbf{x}|\mathbf{t}, \mathbf{W}, \boldsymbol{\mu}, \sigma^2) \sim \mathcal{N}(\mathbf{M}^{-1}\mathbf{W}^\top(\mathbf{t} - \boldsymbol{\mu}), \sigma^2\mathbf{M}^{-1}) \tag{20}$$

where $\mathbf{M} = \mathbf{W}^\top\mathbf{W} + \sigma^2\mathbf{I}$, $\langle\mathbf{x}\rangle = \mathbf{M}^{-1}\mathbf{W}^\top(\mathbf{t} - \boldsymbol{\mu})$ and $\langle\mathbf{x}\mathbf{x}^\top\rangle = \sigma^2\mathbf{M}^{-1} + \langle\mathbf{x}\rangle\langle\mathbf{x}\rangle^\top$.

Appendix B. Hidden Markov Principal Component Analysis

Model:

$$p(\mathbf{T}, \mathbf{X}, \mathbf{Z}, \mathbf{S}) = p(\mathbf{T}, \mathbf{X}, \mathbf{Z}|\mathbf{S})p(\mathbf{S}) \tag{21}$$

$$p(\mathbf{X}, \mathbf{Z}, \mathbf{S}|\mathbf{T}) = p(\mathbf{X}, \mathbf{Z}|\mathbf{S}, \mathbf{T})p(\mathbf{S}|\mathbf{T}) \tag{22}$$

$$\langle\mathcal{L}_c\rangle = \int \ln p(\mathbf{T}, \mathbf{X}, \mathbf{Z}, \mathbf{S})p(\mathbf{X}, \mathbf{Z}, \mathbf{S}|\mathbf{T})d\mathbf{X}d\mathbf{Z}d\mathbf{S} \tag{23}$$

Model definitions:

$$p(\mathbf{S}) = p(s_0)\prod_{n=1}^{N} p(s_n|s_{n-1}) \tag{24}$$

$$p(\mathbf{T}, \mathbf{X}, \mathbf{Z}, \mathbf{S}) = p(\mathbf{T}, \mathbf{X}, \mathbf{Z}|\mathbf{S})p(\mathbf{S}) = p(s_0)\prod_{n=1}^{N} p(s_n|s_{n-1})\prod_{i=1}^{M}[c_{ik}p(\mathbf{t}_n|\mathbf{x}_{nik}, \boldsymbol{\theta}_{ik})p(\mathbf{x}_{nik}|\boldsymbol{\theta}_{ik})]^{z_{nik}} \tag{25}$$

where $\boldsymbol{\theta}_{ik} = \{\boldsymbol{\mu}_{ik}, \boldsymbol{\sigma}^2_{ik}, \mathbf{W}_{ik}\}$ and z_{nik} is one if the component is i and the state is k and zero elsewhere.

Expectations:

$$\langle\ln p(\mathbf{S})\rangle = \langle\ln p(s_0)\rangle + \sum_{n=1}^{N}\langle\ln p(s_n|s_{n-1})\rangle \tag{26}$$

$$\langle\ln p(\mathbf{T}, \mathbf{X}, \mathbf{Z}|\mathbf{S})\rangle = \sum_{n=1}^{N}\sum_{i=1}^{M}\langle z_{nik}\rangle[\langle\ln c_{ik}\rangle + \langle\ln p(\mathbf{t}_n|\mathbf{x}_{nik}, \boldsymbol{\theta}_{ik})\rangle + \langle\ln p(\mathbf{x}_{nik}|\boldsymbol{\theta}_{ik})\rangle] \tag{27}$$

$$\langle z_{nik}\rangle = p(s_n = k|\mathbf{T})\frac{c_{ik}p(\mathbf{t}_n|\boldsymbol{\theta}_{ik})}{\sum_{j=1}^{M} c_{jk}p(\mathbf{t}_n|\boldsymbol{\theta}_{jk})} = \gamma(z_{nik}) \tag{28}$$

$$\begin{aligned}\langle\ln c_{ik}\rangle &= \int \ln c_{ik}p(\mathbf{X}, \mathbf{Z}|\mathbf{T})d\mathbf{X} \\ &= \ln c_{ik}\end{aligned} \tag{29}$$

$$\begin{aligned}\langle \ln p(\mathbf{t}_n|\mathbf{x}_{nik},\boldsymbol{\theta}_{ik})\rangle &= \int \ln p(\mathbf{t}_n|\mathbf{x}_{nik},\boldsymbol{\theta}_{ik}) p(\mathbf{X},\mathbf{Z}|\mathbf{T}) d\mathbf{X} \\ &= -\frac{d}{2}\ln(2\pi\sigma_{ik}^2) - \frac{1}{2\sigma_{ik}^2}\|\mathbf{t}_n - \boldsymbol{\mu}_{ik}\|^2 + \frac{1}{\sigma_{ik}^2}\langle \mathbf{x}_{nik}\rangle^\top \mathbf{W}_{ik}^\top(\mathbf{t}_n - \boldsymbol{\mu}_{ik}) \\ &\quad - \frac{1}{2\sigma_{ik}^2}\mathrm{tr}(\mathbf{W}_{ik}^\top \mathbf{W}_{ik}\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle) \end{aligned} \tag{30}$$

$$\begin{aligned}\langle \ln p(\mathbf{x}_{nik}|\boldsymbol{\theta}_{ik})\rangle &= \int \ln p(\mathbf{x}_{nik}|\boldsymbol{\theta}_{ik}) p(\mathbf{X},\mathbf{Z}|\mathbf{T}) d\mathbf{X} \\ &\quad - \frac{q}{2}\ln(2\pi) - \frac{1}{2}\mathrm{tr}(\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle) \end{aligned} \tag{31}$$

From equation (23), the expectation of $\mathcal{L}_c$ with respect to the posterior distribution $p(\mathbf{X},\mathbf{Z},\mathbf{S}|\mathbf{T})$ takes the form

$$\begin{aligned}\langle \mathcal{L}_c\rangle =& \langle p(\mathbf{T},\mathbf{X},\mathbf{Z},\mathbf{S})\rangle \\ =& \sum_{n=1}^{N}\sum_{i=1}^{M}\sum_{k=1}^{Q}\gamma(z_{nik})\left[\ln c_{ik} - \frac{d}{2}\ln(2\pi\sigma_{ik}^2) - \frac{1}{2\sigma_{ik}^2}\|\mathbf{t}_n - \boldsymbol{\mu}_{ik}\|^2\right. \\ &+ \frac{1}{\sigma_{ik}^2}\langle \mathbf{x}_{nik}\rangle^\top \mathbf{W}_{ik}^\top(\mathbf{t}_n - \boldsymbol{\mu}_{ik}) - \frac{1}{2\sigma_{ik}^2}\mathrm{tr}(\mathbf{W}_{ik}^\top \mathbf{W}_{ik}\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle) \\ &\left. -\frac{q}{2}\ln(2\pi) - \frac{1}{2}\mathrm{tr}(\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle)\right] + \langle \ln p(s_0)\rangle + \sum_{n=2}^{N}\langle \ln p(s_n|s_{n-1})\rangle \end{aligned} \tag{32}$$

The M step corresponds to maximize (32) with respect to $\widetilde{\boldsymbol{\mu}}_{ik}$, $\widetilde{\mathbf{W}}_{ik}$, $\widetilde{\sigma}_{ik}^2$ and $\widetilde{c}_{ik}$ to obtain the updated parameters. The derivatives who gives rise to equations (5),(6) and (7) are as follows

$$\frac{\partial\langle \mathcal{L}_c\rangle}{\partial\widetilde{\boldsymbol{\mu}}_{ik}} = \sum_{n=1}^{N}\gamma(z_{nik})\left[\frac{1}{\widetilde{\sigma}_{ik}^2}(\mathbf{t}_n - \widetilde{\boldsymbol{\mu}}_{ik}) - \frac{1}{\widetilde{\sigma}_{ik}^2}\widetilde{\mathbf{W}}_{ik}\langle \mathbf{x}_{nik}\rangle\right]$$

$$\frac{\partial\langle \mathcal{L}_c\rangle}{\partial\widetilde{\mathbf{W}}_{ik}} = \sum_{n=1}^{N}\gamma(z_{nik})\left[\frac{1}{\widetilde{\sigma}_{ik}^2}(\mathbf{t}_n - \widetilde{\boldsymbol{\mu}}_{ik})\langle \mathbf{x}_{nik}\rangle^\top - \frac{1}{\widetilde{\sigma}_{ik}^2}\widetilde{\mathbf{W}}_{ik}^\top\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle\right]$$

$$\begin{aligned}\frac{\partial\langle \mathcal{L}_c\rangle}{\partial\widetilde{\sigma}_{ik}^2} =& \sum_{n=1}^{N}\gamma(z_{nik})\left[-\frac{d}{2\widetilde{\sigma}_{ik}^2} + \frac{1}{2\widetilde{\sigma}_{ik}^2}\|\mathbf{t}_n - \widetilde{\boldsymbol{\mu}}_{ik}\|^2\right. \\ &\left. -\frac{1}{\widetilde{\sigma}_{ik}^2}\langle \mathbf{x}_{nik}\rangle^\top \widetilde{\mathbf{W}}_{ik}^\top(\mathbf{t}_n - \widetilde{\boldsymbol{\mu}}_{ik}) + \frac{1}{2\widetilde{\sigma}_{ik}^2}\mathrm{tr}(\widetilde{\mathbf{W}}_{ik}^\top\widetilde{\mathbf{W}}_{ik}\langle \mathbf{x}_{nik}\mathbf{x}_{nik}^\top\rangle)\right]\end{aligned}$$

In order to maximize $\widetilde{c}_{ik}$, it must be taken into account the constraint that $\sum_i c_{ik}$. This can be handled by using Lagrange a multiplier λ and maximizing

$$L_k = \langle \mathcal{L}_c \rangle_k + \lambda_k \left[\sum_{i=1}^{M} c_{ik} - 1 \right]$$

to finally obtain (8).

Appendix C. Expected Sufficient Statistics

In order to simplify the equations for the M step, the following expected sufficient statistics are introduced

$$\begin{aligned}
\mathbf{S}_{\gamma,ik} &= \sum_{n=1}^{N} \gamma(z_{nik}) \\
\mathbf{S}_{\mathbf{T},ik} &= \sum_{n=1}^{N} \gamma(z_{nik}) \langle \mathbf{t}_n \rangle \\
\mathbf{S}_{\mathbf{X},ik} &= \sum_{n=1}^{N} \gamma(z_{nik}) \langle \mathbf{x}_{nik} \rangle^{\top} \\
\mathbf{S}_{\mathbf{T}\mathbf{X}^{\top},ik} &= \sum_{n=1}^{N} \gamma(z_{nik}) \langle \mathbf{t}_n \rangle \langle \mathbf{x}_{nik} \rangle^{\top} \\
\mathbf{S}_{\mathbf{X}\mathbf{X}^{\top},ik} &= \sum_{n=1}^{N} \gamma(z_{nik}) \langle \mathbf{x}_{nik} \rangle \langle \mathbf{x}_{nik} \rangle^{\top} \\
\mathbf{S}_{\mathbf{T}\mathbf{T}^{\top},ik} &= \sum_{n=1}^{N} \gamma(z_{nik}) \langle \mathbf{t}_n \rangle \langle \mathbf{t}_n \rangle^{\top}
\end{aligned}$$

In the E step of the EM algorithm, $\gamma(z_{nik})$ is calculated using Λ parameters. In the M step, equations (5),(6), (7) and (8) are used to obtain $\widetilde{\Lambda}$ parameters. These equations are in terms of the raw data, so calculation of the M step is dependent of the number N of observations. Using some straightforward algebra, reestimation formulas for $\widetilde{\boldsymbol{\mu}}_{ik}$, $\widetilde{\mathbf{W}}_{ik}$, $\widetilde{\sigma}^2_{ik}$ and $\widetilde{c}_{ik}$ can be rewritten in terms only of the expected sufficient statistics described above, in such way that there are not anymore dependent of the number of samples. The new update equations are

$$\widetilde{\mathbf{W}}_{ik} = \left[\mathbf{S}_{\mathbf{T}\mathbf{X}^{\top},ik} - \widetilde{\boldsymbol{\mu}}_{ik} \mathbf{S}_{\mathbf{X},ik} \right] \left[\sigma^2_{ik} \mathbf{M}^{-1}_{ik} \mathbf{S}_{\gamma,ik} + \mathbf{S}_{\mathbf{X}\mathbf{X}^{\top},ik} \right]^{-1} \tag{33}$$

$$\begin{aligned}
\widetilde{\sigma}^2_{ik} = &\Big[\mathbf{S}_{\mathbf{T}\mathbf{T}^{\top},ik} - 2\mathbf{S}_{\mathbf{T},ik} + \mathbf{S}_{\gamma,ik} \widetilde{\boldsymbol{\mu}}^{\top}_{ik} \widetilde{\boldsymbol{\mu}}_{ik} - 2 \operatorname{tr} \left(\mathbf{S}_{\mathbf{T}\mathbf{X}^{\top},ik} \mathbf{W}^{\top}_{ik} \right) + 2\mathbf{S}_{\mathbf{X},ik} \mathbf{W}^{\top}_{ik} \widetilde{\boldsymbol{\mu}}_{ik} \\
&+ \mathbf{S}_{\gamma,ik} \operatorname{tr} \left(\sigma^2_{ik} \widetilde{\mathbf{W}}^{\top}_{ik} \widetilde{\mathbf{W}}_{ik} \mathbf{M}^{-1}_{ik} \right) + \operatorname{tr} \left(\widetilde{\mathbf{W}}_{ik} \mathbf{S}_{\mathbf{X}\mathbf{X}^{\top},ik} \widetilde{\mathbf{W}}^{\top}_{ik} \right) \Big] \left[d\mathbf{S}_{\gamma,ik} \right]^{-1}
\end{aligned} \tag{34}$$

$$\widetilde{\boldsymbol{\mu}}_{ik} = \left[\mathbf{S}_{\mathbf{T},ik} - \widetilde{\mathbf{W}}_{ik} \mathbf{S}^{\top}_{\mathbf{X},ik} \right] \left[\mathbf{S}_{\gamma,ik} \right]^{-1} \tag{35}$$

$$\widetilde{c}_{ik} = \frac{\mathbf{S}_{\gamma,ik}}{\sum_{j=1}^{M} \mathbf{S}_{\gamma,jk}} \tag{36}$$