CAMBRIDGE UNIVERSITY

ENGINEERING DEPARTMENT

GENERALISED LINEAR GAUSSIAN MODELS

A-V.I. Rosti & M.J.F. Gales

CUED/F-INFENG/TR.420

November 23, 2001

Cambridge University Engineering Department
Trumpington Street
Cambridge. CB2 1PZ
England

E-mail: {avir2, mjfg}@eng.cam.ac.uk

Abstract

This paper addresses the time-series modelling of high dimensional data. Currently, the hidden Markov model (HMM) is the most popular and successful model especially in speech recognition. However, there are well known shortcomings in HMMs particularly in the modelling of the correlation between successive observation vectors; that is, inter-frame correlation. Standard diagonal covariance matrix HMMs also lack the modelling of the spatial correlation in the feature vectors; that is, intra-frame correlation. Several other time-series models have been proposed recently especially in the segment model framework to address the inter-frame correlation problem such as Gauss-Markov and dynamical system segment models. The lack of intra-frame correlation has been compensated for with transform schemes such as semi-tied full covariance matrices (STC). All these models can be regarded as belonging to the broad class of generalised linear Gaussian models. Linear Gaussian models (LGM) are popular as many forms may be trained efficiently using the expectation maximisation algorithm. In this paper, several LGMs and generalised LGMs are reviewed. The models can be roughly categorised into four combinations according to two different state evolution and two different observation processes. The state evolution process can be based on a discrete finite state machine such as in the HMMs or a linear first-order Gauss-Markov process such as in the traditional linear dynamical systems. The observation process can be represented as a factor analysis model or a linear discriminant analysis model. General HMMs and schemes proposed to improve their performance such as STC can be regarded as special cases in this framework.

1 Introduction

Currently, the most popular and successful time-series model is the hidden Markov model (HMM). HMMs can be applied in a broad range of areas such as speech recognition, bioinformatics and stock market analysis. For example HMM based large vocabulary speech recognition systems have dominated the standard evaluation tasks such as Broadcast News Transcription and Switchboard. The benefits of using HMMs include efficient training and recognition algorithms in which the automatic segmentation using Viterbi algorithm [35] is in very significant role. On the other hand, the state conditional independence assumption in HMMs is a major drawback in case of strongly correlated feature vectors such as in speech recognition [6]. This correlation between successive feature vectors is often called inter-frame correlation. To overcome this deficiency of HMMs, several schemes have been proposed from segmental features to segment models [34]. In general, the use of segment models is restricted by the complicated algorithms due to additional duration models.

Another drawback in the standard diagonal covariance matrix HMMs is its weak spatial correlation modelling. Despite the attempts to reduce this intra-frame correlation by front-end transforms such as discrete cosine transform (DCT) in case of standard Mel-frequency cepstral coefficients [42] there is always some correlation present due to the fixed basis functions in DCT. An optimal front-end should employ data dependent transform bases such as in Karhunen-Loève transform [10] but the complexity of the system becomes equivalent to a full covariance matrix HMMs. Several multiple class based transform schemes have been proposed; e.g., semi-tied full covariance matrices (STC) [12]. Linear discriminant analysis and factor analysis have been also proposed to overcome this problem [29, 40]. In addition to correlation modelling they address the problem of high dimensionality of the feature vectors by allowing lower dimensional subspaces to be used.

The machine learning community has been interested in linear Gaussian models (LGM) for some time now [38] due to the efficiency and applicability of the expectation maximisation (EM) algorithm [5] which provides a consistent framework in supervised learning. Despite the attempts to unify the field of LGMs, several interesting models have been omitted; e.g., independent factor analysis [1], linear discriminant analysis and its extensions [11, 29, 26]. The forms of models reviewed in this paper generalise some of the currently used techniques in the HMM framework such as semi-tied full covariance matrices.

Linear Gaussian models are a subset of more general state-space models that consist of state evolution process and observation process. Strictly speaking linear Gaussian models are state-space models in which the state evolution and observation equations are linear and the distributions are Gaussians [38]. Some models presented in this paper deviate from the strict definition in that mixtures of Gaussians are allowed as the distributions. Therefore they are called generalised LGMs. This generalisation is important in case of feature vectors which tend to have multi-modal distributions due to source and environment differences such as in speech recognition.

Linear Gaussian models also allow the investigation of new subspaces of the feature vectors which often tend to be very high in dimensionality. It is well known that the inclusion of dynamic features such as first and second-order differences to the vectors of cepstral coefficients increase the performance of many recognition systems. That is often thought to handle partially the inter-frame correlation. It is not guaranteed, though, that the standard feature space is optimal [11] and one might argue that there exists some optimal subspace where the modelling assumptions hold better. Determining the optimal state-space dimensionality for different models would be an interesting topic by itself but it is not included in this paper.

This paper is organised as follows. In Section 2, the general modelling framework is described by introducing the stochastic processes involved. Bayesian networks are briefly introduced as a way of illustrating the conditional independencies in probabilistic models. Also, the expectation maximisation algorithm used in estimating the model parameters is outlined. Static models based on factor analysis and mixtures of factor analysers are described in Section 3. In Section 4, linear dynamical systems are presented. Various modelling assumptions used to either decrease the number of model parameters or modify them to better suit the nature of complex signals are introduces as well. Hidden Markov models and an extension of HMMs combining it with a generic factor analyser are introduced in Section 5. Section 6 is devoted to linear discriminant analysis based observation processes in the both cases of discrete and continuous state evolution processes. In section 7, implementation issues are discussed. Finally, Section 8 concludes this paper and some future directions are proposed.

2 Generalised Linear Gaussian Models

State-space models are generally based on a k dimensional state vector[1] $\boldsymbol{x}_t$ and a p dimensional observation vector $\boldsymbol{o}_t$ which satisfy the following generative model

$$\boxed{\begin{aligned} \boldsymbol{x}_{t+1} &= f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t, \boldsymbol{w}_t) && (1) \\ \boldsymbol{o}_t &= g(\boldsymbol{x}_t, \boldsymbol{v}_t) && (2) \end{aligned}}$$

where the function $f(\cdot)$ describes the state evolution process and the function $g(\cdot)$ the observation process. Random vectors $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are called state evolution noise and observation noise, respectively. Although the models described by Eqs. 1 and 2 can exhibit any linear or non-linear functions, the models presented in this paper are restricted to linear ones. In strict definition of linear Gaussian models the noise sources $\boldsymbol{w}_t$ and $\boldsymbol{v}_t$ are distributed according to Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_t^{(x)}, \boldsymbol{\Sigma}_t^{(x)})$ and $\mathcal{N}(\boldsymbol{\mu}_t^{(o)}, \boldsymbol{\Sigma}_t^{(o)})$. Since many feature vectors generally do not have unimodal distributions due to source and environment variability, the unimodal assumption is relaxed by including mixture models in this paper. Often the time dependence of the noise source statistics is also omitted so that the distributions have static parameters, $\boldsymbol{\mu}^{(o)}$ and $\boldsymbol{\mu}^{(x)}$. In the same way the noise vectors can be written without the time indices as $\boldsymbol{v}$ and $\boldsymbol{w}$.

The state evolution process may be viewed as some underlying phenomenon which may be inherent for the process to be modelled. Alternatively, it may be viewed only as a compact representation of the high dimensional observation vectors. For example in speech recognition, the state vector may be viewed as representing positions of the articulatory organs and the state evolution process, $f(\cdot)$, describes their movement in time. Although the above generic model fits into this interpretation, the linear Gaussian models are a crude approximation since the movement of articulators is rather non-linear [4]. Due to this non-linear nature, the articulatory interpretation is not often stressed.

The observation equation, $g(\cdot)$, describes the function mapping the current state vector $\boldsymbol{x}_t$ mixed with the observation noise $\boldsymbol{v}_t$ onto the current observation $\boldsymbol{o}_t$ which is usually higher in dimensionality; i.e, $p > k$. For example in speech recognition, the observation process may be viewed as mapping the positions of the articulatory organs onto the measurable acoustic representation where the observation noise $\boldsymbol{v}_t$ represents all the noise induced by the environment and the recording equipment. As in case of state evolution process, this interpretation is only valid for non-linear models [4]. In this paper, the observation process is only viewed as a scheme to carry out dimensionality reduction.

2.1 State Evolution Process

Firstly, all the state evolution processes used in the following sections are presented. Although the dynamic models are more interesting, the static counterparts provide a general framework to

[1] In this paper, bold capital letters are used to denote matrices e.g. $\boldsymbol{A}$, bold letters refer to vectors e.g. $\boldsymbol{a}$ and plain letters represent scalars e.g. c. All vectors are column vectors. Prime is used to denote the transpose of a matrix or a vector e.g. $\boldsymbol{A}', \boldsymbol{a}'$. The determinant of a matrix is denoted by $|\boldsymbol{A}|$. Gaussian distributed vectors e.g. $\boldsymbol{x}$ with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ are denoted by $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The likelihood of a vector $\boldsymbol{z}$ being generated by the above Gaussian; i.e., the Gaussian evaluated at the point $\boldsymbol{z}$, is represented as $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$. Vectors distributed according to a mixture of Gaussians are denoted shortly by $\boldsymbol{x} \sim \sum_m c_m \mathcal{N}(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$. As opposed to the lower case letter p used to represent a continuous distribution, a capital letter P is used to denote a probability mass function of a discrete variables. The probability that a discrete random variable ω equals m is denoted shortly as $P(\omega = m) = P_\omega(m)$ and the subscript is also omitted if not ambiguous.

begin with. The relationship between the static models and the traditional covariance modelling techniques are discussed in the following sections.

The simplest non-trivial stochastic state evolution process is a static multivariate Gaussian with a mean vector $\boldsymbol{\mu}^{(x)}$ and a covariance matrix $\boldsymbol{\Sigma}^{(x)}$. In the case of static models, the time index t in the subscript is omitted. The generative model can be represented as follows

$$\boldsymbol{x} = \boldsymbol{w}, \qquad \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)}) \tag{3}$$

It is often possible to make further assumptions about the distribution of the noise depending on the situation. Especially diagonal covariance matrices can be used if the state evolution noise is assumed to be spatially uncorrelated.

The mixture of Gaussians state process adds a new discrete hidden layer in the model where the individual Gaussians are selected by a hidden discrete mixture indicator variable ω. Generally, mixture of Gaussians model can be represented as follows

$$\boldsymbol{x} = \boldsymbol{w}, \qquad \boldsymbol{w} \sim \sum_n c_n^{(x)} \mathcal{N}(\boldsymbol{\mu}_n^{(x)}, \boldsymbol{\Sigma}_n^{(x)}) \tag{4}$$

where the mixture priors $c_n^{(x)} = P_\omega(n)$ must sum to unity to guarantee $p(\boldsymbol{x})$ being a true distribution. It is also possible to use factorial mixture of Gaussians where individual elements of the noise vector are drawn from a univariate mixture of Gaussians; e.g., see [1]. The factorial mixture includes a stronger assumption on the independence of the individual state vector components but provides more flexible modelling of the component variances. The difference between mixture and factorial mixture models is the same as between vector quantisation and cooperative vector quantisation [43] as well as hidden Markov models and factorial hidden Markov models [18].

The simplest dynamic state evolution process is a linear first-order Gauss-Markov random process. A Markov assumption is made to the general state evolution process in Eq. 1 so that the current state vector $\boldsymbol{x}_t$ depends only on the previous state vector $\boldsymbol{x}_{t-1}$. The new state vector is generated by a linear state evolution process from the current state vector as follows

$$\boldsymbol{x}_{t+1} = \boldsymbol{A}_t \boldsymbol{x}_t + \boldsymbol{w}_t, \qquad \boldsymbol{w}_t \sim \mathcal{N}(\boldsymbol{\mu}_t^{(x)}, \boldsymbol{\Sigma}_t^{(x)}) \tag{5}$$

where $\boldsymbol{A}_t$ is a k by k state transition matrix and $\boldsymbol{w}$ the state evolution noise. Since the initial state vector $\boldsymbol{x}_1$ is assumed to be a Gaussian with a mean vector $\boldsymbol{\mu}^{(i)}$ and a covariance matrix $\boldsymbol{\Sigma}^{(i)}$, all the subsequent state vectors are Gaussian distributed as follows

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{A}_t \boldsymbol{x}_{t-1} + \boldsymbol{\mu}_t^{(x)}, \boldsymbol{\Sigma}_t^{(x)}) \tag{6}$$

and their joint likelihood is a Gaussian. The state evolution noise is often assumed to be zero mean and spatially uncorrelated; i.e., $\boldsymbol{\Sigma}^{(x)}$ is diagonal. Non-zero mean vector corresponds to a constant drift of the state vector in the direction of $\boldsymbol{\mu}^{(x)}$ and diagonality of $\boldsymbol{\Sigma}^{(x)}$ is often used to reduce the number of model parameters. If the state evolution noise is distributed as a mixture of Gaussians, the state vector is also distributed according to a mixture of Gaussians with the number of components growing exponentially. The implications of this growth are discussed later in this paper.

A Gauss-Markov random process with parameters

$$\boldsymbol{\mu}^{(i)} = \begin{bmatrix} -1 \\ -2 \end{bmatrix}, \quad \boldsymbol{\Sigma}^{(i)} = \begin{bmatrix} 1.3 & 0 \\ 0 & 2.5 \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} 0.9 & -0.2 \\ 0.6 & 0.6 \end{bmatrix}, \quad \boldsymbol{\mu}^{(x)} = \begin{bmatrix} 1.8 \\ 1.3 \end{bmatrix}, \quad \boldsymbol{\Sigma}^{(x)} = \begin{bmatrix} 0.3 & 0 \\ 0 & 0.5 \end{bmatrix}$$

is illustrated on the left-hand side of Figure 1. At $t = 1$, the main axes of the k dimensional ellipsoid ($k = 2$ in the figure) representing the covariance matrix of the state vector are parallel to the coordinate axes since a diagonal initial state covariance matrix is used. At every subsequent time step, the covariance ellipsoid of the preceding state vector is stretched and rotated according to the state evolution matrix and the resulting ellipsoid is convolved with the ellipsoid representing the state evolution noise covariance. The state evolution noise mean vector causes the linear drift

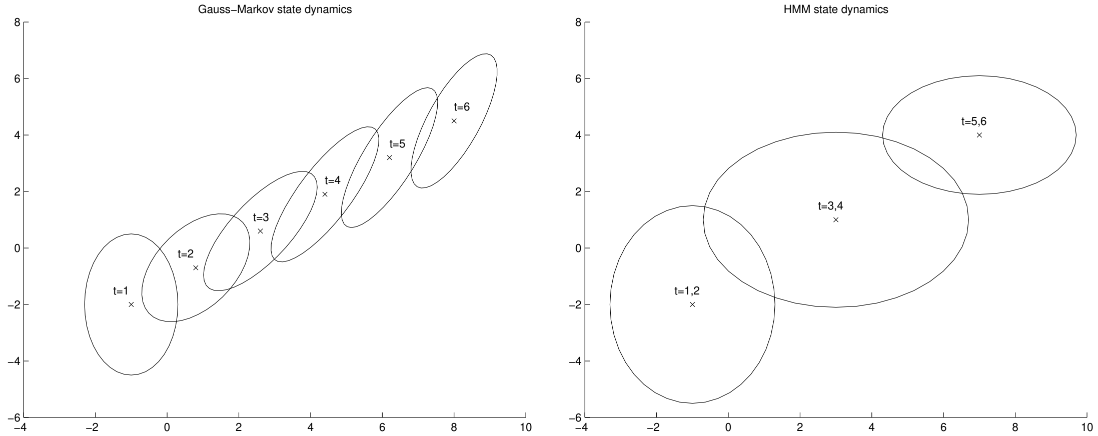

Figure 1: State evolution processes of 6 time steps in length. 1) Linear first-order Gauss-Markov process. 2) Hidden Markov model.

of the ellipsoid centres (marked by x). In modelling of zero mean signals, state evolution noise mean can be omitted.

The last type of the state evolution can be assumed to be generated by a discrete state sequence rather than continuous as above. The most commonly used dynamic discrete state model is the hidden Markov model (HMM). The hidden Markov models are based on N_s hidden states with state conditional densities $b_j(\boldsymbol{x}_t) = p(\boldsymbol{x}_t|j)$, a set of transition probabilities $a_{ij} = P_{q_t|q_{t-1}}(j|i)$ and initial state probabilities $\pi_j = P_{q_1}(j)$. The joint likelihood of the state vector sequence $\boldsymbol{X}$ and the discrete state sequence Q is

$$p(\boldsymbol{X}, Q) = \pi_{q_1} b_{q_1}(\boldsymbol{x}_1) a_{q_1 q_2} b_{q_2}(\boldsymbol{x}_2) a_{q_2 q_3} b_{q_3}(\boldsymbol{x}_3) \dots \tag{7}$$

Since, in practise, the underlying discrete state sequence is unknown (hidden) the likelihood is estimated by summing over all possible discrete state sequences

$$p(\boldsymbol{X}) = \sum_{\{Q_T\}} \pi_{q_1} \prod_{t=2}^{T} b_{q_t}(\boldsymbol{x}_t) a_{q_{t-1} q_t} \tag{8}$$

where $\{Q_T\}$ denotes the set of all possible state sequences of length T. In HMMs the output vectors are generated by one of the N_s discrete states and the transition from one discrete state to the next is assumed to be abrupt. The hidden Markov model as a generative model for $\boldsymbol{x}_t$is represented later in this paper as follows

$$\boldsymbol{x}_t \sim \mathcal{M}^{hmm} \tag{9}$$

A three state hidden Markov model state evolution process with the following state distribution parameters

$$\begin{aligned} \boldsymbol{\mu}_1^{(x)} &= \begin{bmatrix} -1 \\ -2 \end{bmatrix} \qquad \boldsymbol{\mu}_2^{(x)} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \qquad \boldsymbol{\mu}_3^{(x)} = \begin{bmatrix} 7 \\ 4 \end{bmatrix} \\ \boldsymbol{\Sigma}_1^{(x)} &= \begin{bmatrix} 2.3 & 0 \\ 0 & 3.5 \end{bmatrix} \; \boldsymbol{\Sigma}_2^{(x)} = \begin{bmatrix} 3.7 & 0 \\ 0 & 3.1 \end{bmatrix} \; \boldsymbol{\Sigma}_3^{(x)} = \begin{bmatrix} 2.7 & 0 \\ 0 & 2.1 \end{bmatrix} \end{aligned} \tag{10}$$

is depicted on the right-hand side of Figure 1. Since diagonal state conditional densities are used, the main axes of the k dimensional ellipsoid ($k = 2$ in the figure) at a given time are parallel to the coordinate axes. The figure represents a three state HMM and one possible assignment of the state vectors into the states in case of a sequence of six output vectors. The HMM performs clustering of the state vector space in case of full transition probability matrix but different constraints can be set for the transition probabilities according to the nature of the signals to be modelled. In case of causal time-series modelling, so called left to right HMMs are often used so that the discrete states represent parts of the modelling units that can occur only sequentially.

2.2 Observation Process

Two different observation processes are presented here. Both processes can perform dimension reduction in that the state-space is lower in dimensionality than the observation space; $k < p$. The first observation equation is called factor analysis (FA) since in a static case, the model reduces to a standard FA model with some additional assumptions about the state and noise distributions. The factor analysis observation process can be represented as

$$\boldsymbol{o}_t = \boldsymbol{C}_t\boldsymbol{x}_t + \boldsymbol{v}_t, \qquad \boldsymbol{v}_t \sim \mathcal{N}(\boldsymbol{\mu}_t^{(o)}, \boldsymbol{\Sigma}_t^{(o)}) \tag{11}$$

where $\boldsymbol{C}$ is a p by k observation matrix and $\boldsymbol{v}$ the observation noise. The observation noise is independent of the state vector and its covariance matrix is often assumed to be diagonal to capture the correlation into the observation matrix. In case of static data modelling, the time indices in the observation equation can be omitted. If the current state vector is Gaussian distributed, the current observation is a Gaussian with the following distribution

$$p(\boldsymbol{o}_t|\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}_t\boldsymbol{x}_t + \boldsymbol{\mu}_t^{(o)}, \boldsymbol{\Sigma}_t^{(o)}) \tag{12}$$

and all the possible joint distributions are Gaussians. Factor analysis and different mixture assumptions are described in more detail in Section 3.

The second type of the observation equation is called linear discriminant analysis (LDA) because in a restricted static case, the model reduces to a standard LDA. The meaningful dimensions are modelled by the state vector and the nuisance dimensions are modelled by a single Gaussian distributed observation noise as follows

$$\boldsymbol{o}_t = \boldsymbol{C}_t \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{v}_t \end{bmatrix}, \qquad \boldsymbol{v}_t \sim \mathcal{N}(\boldsymbol{\mu}_t^{(o)}, \boldsymbol{\Sigma}_t^{(o)}) \tag{13}$$

where $\boldsymbol{C}_t$ is a p by p matrix and $\boldsymbol{v}$ is a $p-k$ dimensional vector. There are several ways to express the posterior of the observations. The most commonly used is

$$p(\boldsymbol{o}_t|\boldsymbol{x}_t) = \mathrm{abs}|\boldsymbol{C}_t^{-1}|\mathcal{N}(\boldsymbol{C}_t^{-1}\boldsymbol{o}_t; \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{\mu}_t^{(o)} \end{bmatrix}, \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_t^{(o)} \end{bmatrix}) \tag{14}$$

where the first k columns and k rows of the covariance matrix are filled with zeroes because the first k elements of the observation vector are deterministic given the current state vector $\boldsymbol{x}_t$. The first term in the equation is needed to scale the likelihood to be valid and the absolute value is needed to guarantee non-negative likelihoods. Since the deterministic elements do not have any influence on the likelihoods, another way of representing the posterior is

$$p(\boldsymbol{o}_t|\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}_t \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{\mu}_t^{(o)} \end{bmatrix}, \boldsymbol{C}_{t[p-k]}\boldsymbol{\Sigma}_t^{(o)}\boldsymbol{C}'_{t[p-k]}) \tag{15}$$

where $\boldsymbol{C}_{t[p-k]}$ denotes a matrix consisting of the $p-k$ rightmost columns of matrix $\boldsymbol{C}_t$. In this representation, the covariance matrix is clearly non-singular provided the transformation and the observation noise covariance are non-singular as well. Linear discriminant analysis is reviewed in more detail in Section 6.

2.3 Basic Models

Different combinations of the state and observation processes presented in this paper are depicted in Figures[2] 2 and 3. Figure 2 depicts models based on static state evolution process. In the top of

[2]These diagrams do not illustrate all the possible forms of linear Gaussian models. E.g., several combinations using LDA observation process are not present.

the diagram is a single static multivariate Gaussian as the basis for all the subsequent models. A mixture of Gaussians can be regarded as vector quantisation with the Gaussian mean vectors as the cluster centres. Vector quantisation is often used in the initialisation of, e.g., HMM parameters [42]. Instead of using only one code-book to assign a vector into a single cluster, cooperative vector quantisation [43] or factorial mixture of Gaussians uses several independent code-books in a distributed manner. All the three models above correspond to state-space model with identity matrix as the observation matrix and zero observation noise.

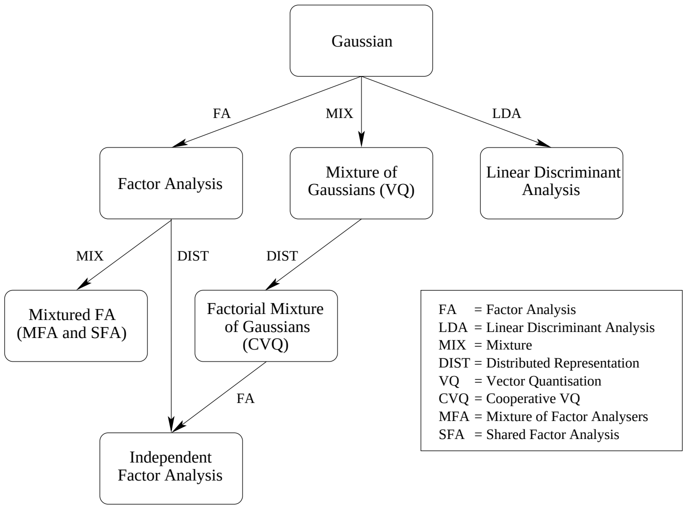

Figure 2: Diagram of static linear Gaussian models. The arrows represent additional properties to the model they are attached to.

Factor analysis [20, 26, 39] is based on a static multivariate Gaussian state process and a factor analysis observation process. In standard factor analysis the state vectors are assumed to be distributed according to a standard Gaussian density, $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Independent factor analysis [1], mixture of factor analysers [15] and shared factor analysis [22] are based on the factor analysis model with different mixture assumptions. These mixture assumptions are described in a unified way in Section 3. Linear discriminant analysis [26] is another static data modelling scheme with linear discriminant analysis observation process.

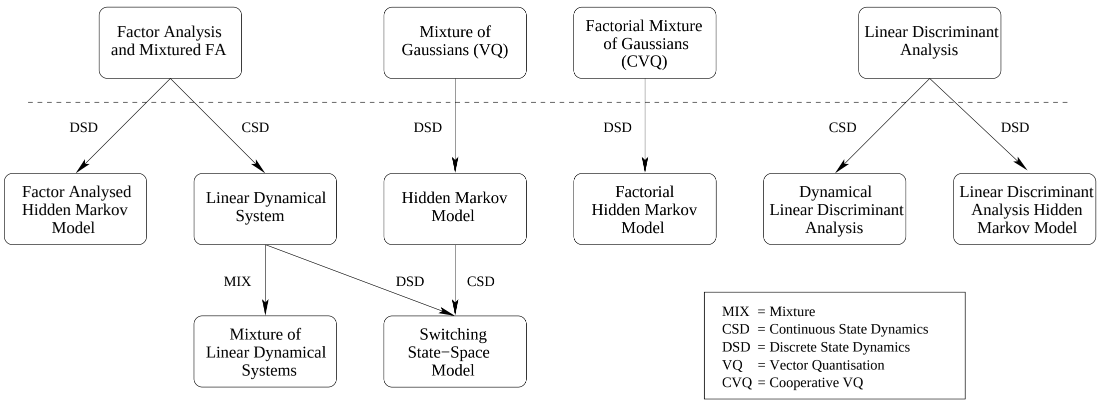

Figure 3: Dynamic linear Gaussian models and how they relate to some of the static models.

Dynamic linear Gaussian models and the corresponding static models are illustrated in Figure 3. Dynamic models with factor analysis observation process include linear dynamical systems [6, 7, 16, 32, 34, 38], mixture of linear dynamical systems and switching state-space model [17, 33] as well as different variations of factor analysed HMMs presented later in this paper and its restricted version in [40]. The linear discriminant observation process is illustrated in case of HMM based [11, 21, 29] and linear first-order Gauss-Markov based state evolution processes. The dynamical linear discriminant analysis is presented later in this paper.

Standard hidden Markov model [35, 42] can be considered as a special case of both the observation processes by just omitting the observation noise and setting the observation matrix to an identity matrix; i.e., $\boldsymbol{C} = \boldsymbol{I}$. Also semi-tied covariance matrix HMMs (STC) [12] can be described by both observation processes when $k = p$ and $\boldsymbol{v} = \boldsymbol{0}$. Factorial hidden Markov models [18] use distributed representation of the discrete state-space so that several independent HMMs can be viewed to have produced the observation vectors.

2.4 Bayesian Networks

In this paper, Bayesian networks [14] are used to illustrate the statistical independencies between different random variables in the probabilistic models. Bayesian networks are directed acyclic graphs, also known as graphical models. The notation is adopted from [33] where round nodes were used to denote continuous and squared nodes discrete random variables. The observable variables are shaded and the lack of an arrow between nodes represents conditional independence.

As an example, consider three continuous random variables $\boldsymbol{z}$,$\boldsymbol{x}$ and $\boldsymbol{o}$ of which $\boldsymbol{o}$ is observed and the others are hidden. The joint likelihood can be factored as a product of conditional likelihoods as follows

$$p(\boldsymbol{z}, \boldsymbol{x}, \boldsymbol{o}) = p(\boldsymbol{z})p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{o}|\boldsymbol{x}, \boldsymbol{z}) \tag{16}$$

This factorisation is perfectly valid in every case. If no assumptions of conditional independence can be made, the corresponding Bayesian network must be illustrated as the first network in Figure 4. There are two arrows pointing the node representing $\boldsymbol{o}$ since in the above factorisation $\boldsymbol{o}$ depends on both $\boldsymbol{z}$ and $\boldsymbol{x}$.

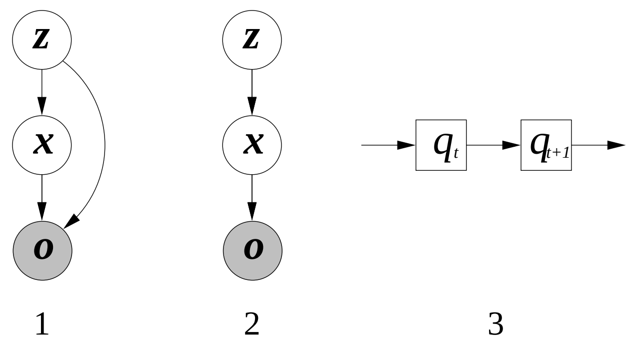

Figure 4: Examples of Bayesian networks representing different assumptions on conditional independence. 1) Continuous random variables $\boldsymbol{z}$, $\boldsymbol{x}$ and $\boldsymbol{o}$ (shading denotes observable) are fully dependent, 2) $\boldsymbol{o}$ is conditionally independent of $\boldsymbol{z}$ given $\boldsymbol{x}$, 3) discrete random variable q_{t+1} is conditionally independent of all its predecessors given q_t (discrete Markov chain).

If the random variable $\boldsymbol{o}$ is assumed to be conditionally independent of $\boldsymbol{z}$ given $\boldsymbol{x}$ the joint likelihood can be rewritten as follows

$$p(\boldsymbol{z}, \boldsymbol{x}, \boldsymbol{o}) = p(\boldsymbol{z})p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{o}|\boldsymbol{x}) \tag{17}$$

The corresponding Bayesian network is depicted as the second graph in Figure 4. The arrow between the nodes representing $\boldsymbol{z}$ and $\boldsymbol{o}$ can be deleted.

As another example, consider an ordered set of discrete random variables, $Q = q_1, \ldots, q_T$, which are encountered several times in the following sections. The joint likelihood of the variables up to time instant $t+1$ can be written using conditional likelihoods as follows

$$P(q_1, \ldots, q_{t+1}) = P(q_1)P(q_2|q_1)P(q_3|q_1, q_2) \ldots P(q_{t+1}|q_1, \ldots, q_t) \tag{18}$$

Often a simplification is achieved by using a Markov assumption which says that the likelihood of the variable q_{t+1} is conditionally independent of all the other previous variables given the immediate predecessor q_t; i.e.,

$$P(q_{t+1}|q_1, \ldots, q_t) = P(q_{t+1}|q_t) \tag{19}$$

This is often called a discrete Markov chain and is illustrated as the third graph of Figure 4.

2.5 The EM Algorithm

Estimation of the parameters in probabilistic models involving latent (hidden) variables is often carried out using the expectation maximisation (EM) algorithm [5]. To find the maximum likelihood (ML) estimates for the model parameters, the joint log-likelihood of the data $\log p(\boldsymbol{O})$ has to be maximised. This can be done in a single run if there are no hidden variables present. The hidden variables, often a collection of hidden state vectors $\boldsymbol{X}$, can be regarded as missing data. In the EM framework the objective is to maximise the log-likelihood of the complete data, $\log p(\boldsymbol{O}, \boldsymbol{X}|\hat{\mathcal{M}})$, given the new model parameters, $\hat{\mathcal{M}}$, via an iterative two step process of estimating the ML solution for the hidden state vectors given the observed data, and then maximising the joint log-likelihood.

By using Jensen's inequality, the joint log-likelihood of the data given the new model parameters is bound from below as follows

$$\log p(\boldsymbol{O}|\hat{\mathcal{M}}) = \log \int p(\boldsymbol{O}, \boldsymbol{X}|\hat{\mathcal{M}})d\boldsymbol{X} \geq \int p^{\circ}(\boldsymbol{X}|\boldsymbol{O}) \log \frac{p(\boldsymbol{O}, \boldsymbol{X}|\hat{\mathcal{M}})}{p^{\circ}(\boldsymbol{X}|\boldsymbol{O})} d\boldsymbol{X} \tag{20}$$

where $p^{\circ}(\boldsymbol{X}|\boldsymbol{O})$ is an arbitrary conditional distribution and $\hat{\mathcal{M}}$ is the set of new parameters. The equality in Eq. 20 is attained exactly when the conditional distribution is chosen to be the posterior of the state sequence given the observation sequence and the set of old model parameters, $p^{\circ}(\boldsymbol{X}|\boldsymbol{O}) = p(\boldsymbol{X}|\boldsymbol{O}, \mathcal{M})$. Since the denominator on the right hand side of Eq. 20 is independent of the new model parameters, the function to be maximised reduces to the expectation of the log-likelihood of the complete data given the new model parameters with respect to the sequence of state vectors given the observation sequence and the old model parameters; i.e.,

$$\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = E\{\log p(\boldsymbol{O}, \boldsymbol{X}|\hat{\mathcal{M}})|\boldsymbol{O}, \mathcal{M}\} = \int p(\boldsymbol{X}|\boldsymbol{O}, \mathcal{M}) \log p(\boldsymbol{O}, \boldsymbol{X}|\hat{\mathcal{M}})d\boldsymbol{X} \tag{21}$$

which is often called as an auxiliary function and is denoted by $\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}})$. The selection of the arbitrary conditional distribution guarantees that in any iteration, the likelihood never decreases.

The derivation of the EM algorithm for a given model always starts off with finding the expected state statistics given the observation sequence and finding the log-likelihood function for the complete data. With linear Gaussian models the statistics of the posterior needed in the maximisation of the auxiliary function are just the mean vector and covariance matrix since the posteriors are also Gaussian. In case of dynamic models based on Gauss-Markov dynamics, the cross covariance matrix between two successive state vectors has to be obtained but it is easy to obtain since the state vectors are also jointly Gaussian.

In the appendices, the auxiliary function is written as a function of the new model parameters marked with caps; e.g. $\hat{\boldsymbol{C}}$. It has to be noted that in the M step, $\hat{\boldsymbol{C}}$ is treated as a variable for which the value that maximises the auxiliary function is determined using standard optimisation methods.

3 Factor Analysis

Factor analysis is a statistical method for modelling the covariance structure of high dimensional static data using a small number of latent (hidden) variables [26]. Traditionally, the latent variables (state vector), $\boldsymbol{x}$, are assumed to be distributed according to a standard Gaussian density, $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, so that the covariance structure would be captured into the factor loading matrix (observation matrix) $\boldsymbol{C}$. If the state vectors had a non-zero mean and its covariance matrix was not an identity matrix, the model would be degenerate since the state vector mean can always be subsumed to the observation noise mean vector and the covariance structure to the observation matrix [38]. The observation noise is assumed to be a Gaussian with mean vector $\boldsymbol{\mu}^{(o)}$ and diagonal covariance matrix $\boldsymbol{\Sigma}^{(o)}$. Diagonality of the observation noise covariance matrix is important since otherwise the parameter estimation could choose the loading matrix to be zero and estimate the observation noise by the sample averages. This would be a maximum likelihood estimate but not very interesting since all the information would be regarded as a p dimensional noise and no gain would be attained by the covariance model.

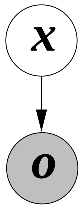

Figure 5: Bayesian network representing factor analysis model.

The generative model of factor analysis can be written as follows

$$\boldsymbol{x} = \boldsymbol{w}, \qquad \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \tag{22}$$

$$\boldsymbol{o} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{v}, \qquad \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}) \tag{23}$$

where the state vector $\boldsymbol{x}$ is k dimensional, the observation noise $\boldsymbol{v}$ as well as the observation $\boldsymbol{o}$ are p dimensional and $\boldsymbol{C}$ is a p by k observation matrix. The generative model of factor analysis can be represented as the Bayesian network in Figure 5. The likelihood of an observation given the state is simply

$$p(\boldsymbol{o}_j|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{o}_j; \boldsymbol{C}\boldsymbol{x} + \boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}) \tag{24}$$

since when the state vector $\boldsymbol{x}$ is given, the product $\boldsymbol{C}\boldsymbol{x}$ is a constant vector added to the observation noise vector $\boldsymbol{v}$. Furthermore, the joint likelihood of the state and observation is Gaussian with the following distribution

$$p(\boldsymbol{o}_j, \boldsymbol{x}) = \mathcal{N}(\begin{bmatrix} \boldsymbol{o}_j \\ \boldsymbol{x} \end{bmatrix}; \begin{bmatrix} \boldsymbol{\mu}^{(o)} \\ \boldsymbol{0} \end{bmatrix}, \begin{bmatrix} \boldsymbol{C}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)} & \boldsymbol{C} \\ \boldsymbol{C}' & \boldsymbol{I} \end{bmatrix}) \tag{25}$$

According to the generative model in Eqs. 22 and 23, the k dimensional factors are distributed as a standard Gaussian distribution as illustrated on the upper left-hand side of Figure 6 where $k = 1$. The covariance matrix of the factors is therefore a k dimensional ball with unit radius. The k dimensional ball is then stretched and rotated according to the observation matrix $\boldsymbol{C}$. The observation noise covariance can be represented as a p dimensional ellipsoid with the main axes parallel to the coordinate axes due to the diagonal covariance matrix assumption. Since the sum of two independent random variables is distributed according to the convolution of the individual distributions, the covariance matrix of the transformed factors is convolved with the ellipsoid

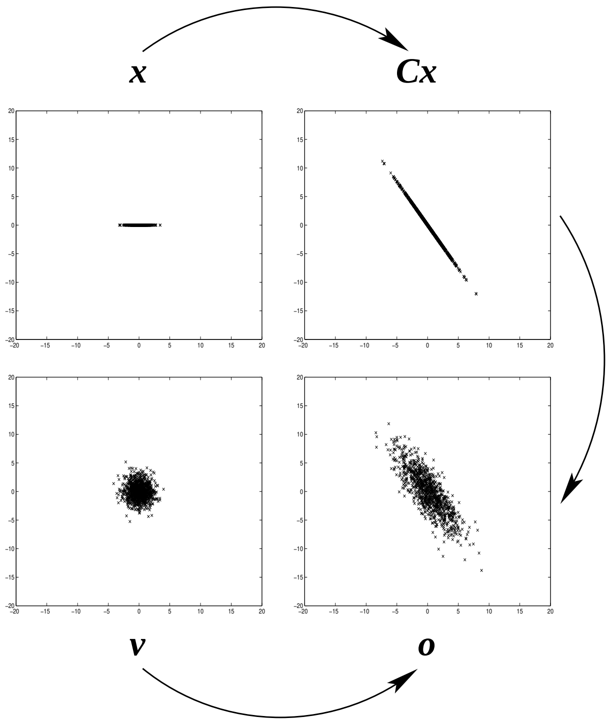

Figure 6: Factor analysis with 1-dimensional state-space and 2-dimensional observation space. $\mathcal{N}(0,1)$ distributed state vectors, x, are stretched and rotated according to the transform $\boldsymbol{C}$, and $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^{(o)})$ distributed noise vectors are added to form the observation vectors. It should be noted that a sum of two random variables is distributed according to the convolution of the individual distributions; in this example $\mathcal{N}(\mathbf{0}, \boldsymbol{C}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)})$.

representing the observation noise covariance as depicted on the bottom right-hand side of Figure 6.

The number of parameters needed to model the observations as a single p dimensional multivariate Gaussian is $p(p+1)/2 + p$ where the first term in the sum corresponds to the symmetric covariance matrix and the second to the mean vector. A factor analysis model requires $pk + 2p$ parameters where the first term of the sum corresponds to the factor loading matrix and the second to the diagonal covariance matrix and the mean vector of the observation noise. A reduction in the number of model parameters by using factor analysis can be attained by choosing the state-space dimensionality according to $k < (p-1)/2$.

There are two different approaches in solving the factor analysis problem [26]. The direct method is computationally very expensive since it requires spectral factorisation of the factor loading matrix. The maximum likelihood (ML) solution suits better the linear Gaussian model framework presented here since the efficient EM algorithm can be used.

3.1 EM Algorithm for Factor Analysis

Let $\boldsymbol{O} = \boldsymbol{o}_1, \ldots, \boldsymbol{o}_N$ be a set of N independent observation vectors. In the case of factor analysis the auxiliary function can be simplified since the log-likelihood of the complete data depends only on the observation vectors due to the distribution assumption on the factors. The auxiliary function can be expressed as follows

$$\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = E\big\{ \log p(\boldsymbol{O}|\boldsymbol{x}, \hat{\mathcal{M}}) | \boldsymbol{O}, \mathcal{M} \big\} = \sum_{j=1}^{N} \int p(\boldsymbol{x}|\boldsymbol{o}_j, \mathcal{M}) \log p(\boldsymbol{o}_j|\boldsymbol{x}, \hat{\mathcal{M}}) d\boldsymbol{x} \tag{26}$$

where $\hat{\mathcal{M}}$ is the set of new model parameters. Since the observation vectors are independent, the posterior of the data given the factors becomes a sum after taking the logarithm.

The E step requires estimating the statistics of the posterior of the state vectors. Since the posterior is also a Gaussian, only the first and second-order statistics are non-zero as follows

$$\hat{\boldsymbol{x}}(j) = \boldsymbol{K}(\boldsymbol{o}_j - \boldsymbol{\mu}^{(o)}) \tag{27}$$

$$\hat{\boldsymbol{R}}(j) = \boldsymbol{I} - \boldsymbol{K}\boldsymbol{C} + \hat{\boldsymbol{x}}(j)\hat{\boldsymbol{x}}'(j) \tag{28}$$

where $\boldsymbol{K} = \boldsymbol{C}'(\boldsymbol{C}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)})^{-1}$ which can be computed in advance for the current model set.

The M step is also very straightforward. However it requires some manipulation to obtain the new parameters using convenient matrix operations. The new model parameters can be obtained by the following two re-estimation formulae

$$\begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix} = \Big(\sum_{j=1}^{N} \begin{bmatrix} \boldsymbol{o}_j \hat{\boldsymbol{x}}'(j) & \boldsymbol{o}_j \end{bmatrix} \Big) \Big(\sum_{j=1}^{N} \begin{bmatrix} \hat{\boldsymbol{R}}(j) & \hat{\boldsymbol{x}}(j) \\ \hat{\boldsymbol{x}}'(j) & 1 \end{bmatrix} \Big)^{-1} \tag{29}$$

$$\hat{\boldsymbol{\Sigma}}^{(o)} = \frac{1}{N} \sum_{j=1}^{N} \operatorname{diag}\Big(\boldsymbol{o}_j \boldsymbol{o}_j' - \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix} \begin{bmatrix} \boldsymbol{o}_j \hat{\boldsymbol{x}}'(j) & \boldsymbol{o}_j \end{bmatrix}' \Big) \tag{30}$$

where $\operatorname{diag}(\cdot)$ denotes setting the elements outside the main diagonal to zeroes. The details of the derivation are presented in Appendix B.

3.2 Mixture of Factor Analysers

The standard factor analysis model works well for correlated data with Gaussian distribution provided the number of factors (state-space dimensionality) is chosen well. In reality the data are not always Gaussian distributed; e.g., speech feature vectors may have bimodal distributions due to the gender variations. A generic mixture of factor analysers (MFA) is based on three random indicator variables ω^x, ω^o and ω^c with priors $c_n^{(x)} = P_{\omega^n}(m)$, $c_m^{(o)} = P_{\omega^o}(m)$ and $c_l^{(c)} = P_{\omega^c}(l)$. The Bayesian network corresponding to this model is illustrated as the first graph in Figure 7. The state-space indicator, ω^x, chooses the state noise parameters $(\boldsymbol{\mu}_n^{(x)}, \boldsymbol{\Sigma}_n^{(x)})$ from $M^{(x)}$ different sets, the observation noise indicator, ω^o, chooses the state noise parameters $(\boldsymbol{\mu}_m^{(o)}, \boldsymbol{\Sigma}_m^{(o)})$ from $M^{(o)}$ different sets and the observation matrix indicator chooses a constant matrix $\boldsymbol{C}_l$ from $M^{(c)}$ alternatives.

It should be noted that in this generic notation, the distribution of the observation matrix is crucial for the model to be a generalised linear Gaussian model. If the observation matrix is Gaussian distributed, the observations are not anymore distributed as Gaussians or even mixtures of Gaussians. To guarantee the observations being Gaussians, the observation matrix is chosen by the mixture indicator from a set of possible constant matrices. Therefore, the observation matrix is distributed according to a weighted delta distribution as presented in the generative model below.

Several parameters of MFA can be assumed to depend on the same mixture indicator. For example, the second graph in Figure 7 represents a model where all the observation parameters $(\boldsymbol{C}_m, \boldsymbol{\mu}_m^{(o)}, \boldsymbol{\Sigma}_m^{(o)})$ depend on the same indicator, ω^o. The third graph depicts a case where all the parameters, $(\boldsymbol{\mu}_m^{(x)}, \boldsymbol{\Sigma}_m^{(x)}, \boldsymbol{C}_m, \boldsymbol{\mu}_m^{(o)}, \boldsymbol{\Sigma}_m^{(o)})$, depend on a single indicator, ω. In this paper only the latter two cases are considered.

The generative model for the generic case without parameter tying can be written as

$$\boldsymbol{x} = \boldsymbol{w}, \qquad \boldsymbol{w} \sim \sum_{n} c_n^{(x)} \mathcal{N}(\boldsymbol{\mu}_n^{(x)}, \boldsymbol{\Sigma}_n^{(x)}) \tag{31}$$

$$\boldsymbol{o} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{v}, \qquad \begin{aligned} \boldsymbol{v} &\sim \sum_{m} c_m^{(o)} \mathcal{N}(\boldsymbol{\mu}_m^{(o)}, \boldsymbol{\Sigma}_m^{(o)}) \\ \boldsymbol{C} &\sim \sum_{l} c_l^{(c)} \delta(\boldsymbol{C} - \boldsymbol{C}_l) \end{aligned} \tag{32}$$

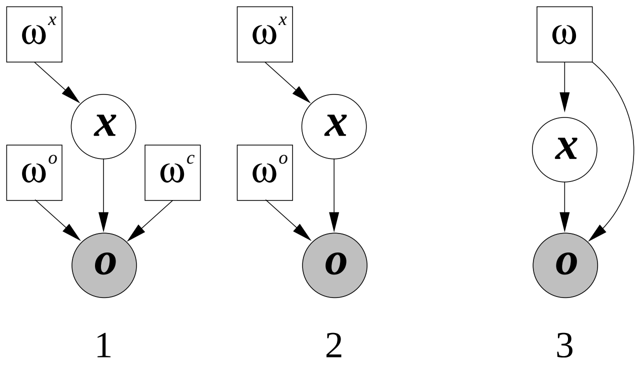

Figure 7: Bayesian networks representing mixture of factor analysers. 1) Generic MFA with three mixture indicators. 2) Observation parameters ($\boldsymbol{C}_m, \boldsymbol{\mu}_m^{(o)}, \boldsymbol{\Sigma}_m^{(o)}$) depend on the same indicator. 3) All parameters ($\boldsymbol{\mu}_m^{(x)}, \boldsymbol{\Sigma}_m^{(x)}, \boldsymbol{C}_m, \boldsymbol{\mu}_m^{(o)}, \boldsymbol{\Sigma}_m^{(o)}$) depend on the same indicator.

which can be easily modified to represent any of the other mixture models just by modifying the corresponding superscripts. Only difference between the training of these models comes with the posterior estimation. Using the conditional independence assumptions, the joint likelihood of the observation, state and the mixture components is simply

$$p(\boldsymbol{o}_j, \boldsymbol{x}, m, n) = c_n^{(x)} p(\boldsymbol{x}|n) c_m^{(o)} p(\boldsymbol{o}_j|\boldsymbol{x}, m) \tag{33}$$

In general, learning the model parameters of mixture of factor analysers can be carried out by the EM algorithm. The normal factor analysis E step has to be modified to take component posteriors into account. Regardless of the model assumptions, the joint component posteriors can be represented as follows

$$\gamma_{mn}(j) = P(m, n|\boldsymbol{o}_j, \mathcal{M}) = \frac{c_m^{(o)} c_n^{(x)} p(\boldsymbol{o}_j|m, n, \mathcal{M})}{\sum_{l=1}^{M^{(o)}} c_l^{(o)} \sum_{i=1}^{M^{(x)}} c_i^{(x)} p(\boldsymbol{o}_j|l, i, \mathcal{M})} \tag{34}$$

where $p(\boldsymbol{o}_j|m, n, \mathcal{M})$ varies according to the choice of the scheme. In the second case of the figure it can be written as

$$p(\boldsymbol{o}_j|m, n, \mathcal{M}) = \mathcal{N}(\boldsymbol{o}_j; \boldsymbol{C}_m \boldsymbol{\mu}_n^{(x)} + \boldsymbol{\mu}_m^{(o)}, \boldsymbol{C}_m \boldsymbol{\Sigma}_n^{(x)} \boldsymbol{C}_m + \boldsymbol{\Sigma}_m^{(o)}) \tag{35}$$

The estimation of the first and second-order component dependent expected state statistics, $\hat{\boldsymbol{x}}_{mn}(j) = E\{\boldsymbol{x}|\boldsymbol{o}_j, m, n, \mathcal{M}\}$ and $\hat{\boldsymbol{R}}_{mn}(j) = E\{\boldsymbol{x}\boldsymbol{x}'|\boldsymbol{o}_j, m, n, \mathcal{M}\}$, has to be modified accordingly.

The re-estimation of the model parameters in the third case of mixture of factor analysers is the same as in case of a standard factor analyser in Eqs. 29 and 30 apart from the inclusion of the mixture posteriors given above. The M steps for the other mixture assumptions are described in the following two sections. The estimation of the new component priors, $\hat{c}_m^{(o)}$ and $\hat{c}_n^{(x)}$, can be carried out as follows

$$\hat{c}_m^{(o)} = \frac{1}{N} \sum_{j=1}^{N} \gamma_m^{(o)}(j) \tag{36}$$

$$\hat{c}_n^{(x)} = \frac{1}{N} \sum_{j=1}^{N} \gamma_n^{(x)}(j) \tag{37}$$

The derivation for this is presented in Appendix C.

3.3 Mixtures of Factors

In this section a mixture of factor analysers with a mixture of Gaussians as the state process is presented. Similar approach is taken in the independent factor analysis (IFA) [1] apart from the independence assumption. The main difference is that IFA is based on factorial mixture of Gaussians where the elements of the state vector (factors) are assumed independent and generated individually by univariate mixtures of Gaussians. Bayesian networks corresponding to a factor analyser with mixtures of factors and IFA are depicted in Figure 8.

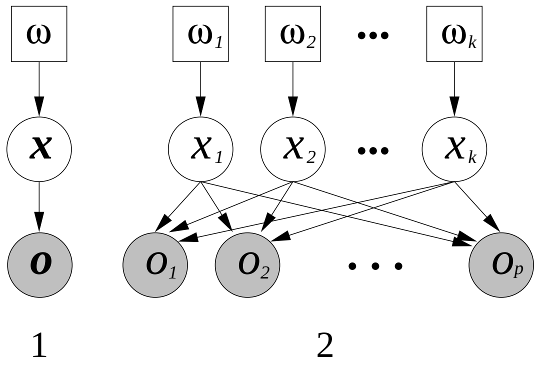

Figure 8: Bayesian networks representing factor analysers with mixtures of factors. 1) One mixture indicator chooses the factors (state vector). 2) In IFA, every factor has its own mixture indicator.

It should be noted that only $(M-1)(2k+1)$ additional parameters are effectively used in this model due to the degeneracy mentioned with the standard factor analysis model. The mean vector and the covariance matrix of one state noise component can always be subsumed to the observation noise mean vector and the observation matrix, respectively. Also, only $M-1$ parameters are needed to define the component priors.

Using a mixture of Gaussians as the state evolution noise makes the E step a bit more complicated. The conditioning of multivariate Gaussians presented in Appendix A.2 becomes useful. Detailed derivation of the E step is presented in Appendix C. The sufficient statistics of the state posterior given the observation and the mixture component are

$$\hat{\boldsymbol{x}}_n^{(x)}(j) = \boldsymbol{\mu}_n^{(x)} + \boldsymbol{K}_n(\boldsymbol{o}_j - \boldsymbol{C}\boldsymbol{\mu}_n^{(x)} - \boldsymbol{\mu}^{(o)}) \tag{38}$$

$$\hat{\boldsymbol{R}}_n^{(x)}(j) = \boldsymbol{\Sigma}_n^{(x)} - \boldsymbol{K}_n\boldsymbol{C}\boldsymbol{\Sigma}_n^{(x)} + \hat{\boldsymbol{x}}_n^{(x)}(j)\hat{\boldsymbol{x}}_n(j)' \tag{39}$$

where $\boldsymbol{K}_n = \boldsymbol{\Sigma}_n^{(x)}\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{\Sigma}_n^{(x)}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)})^{-1}$. Since all the observation parameters are tied over the mixture indicator, the state posteriors given the observation are needed. They can be obtained easily as follows

$$\hat{\boldsymbol{x}}(j) = \sum_{n=1}^{M} \gamma_n^{(x)}(j)\hat{\boldsymbol{x}}_n^{(x)}(j) \tag{40}$$

$$\hat{\boldsymbol{R}}(j) = \sum_{n=1}^{M} \gamma_n^{(x)}(j)\hat{\boldsymbol{R}}_n^{(x)}(j) \tag{41}$$

where $\gamma_n^{(x)}(j)$ corresponds to the posterior of the state-space mixture component. These statistics can be used in the re-estimation of the observation parameters as in normal factor analysis case; see Eqs. 29 and 30.

The re-estimation formulae for the state parameters are

$$\hat{\boldsymbol{\mu}}_n^{(x)} = \frac{\sum_{j=1}^{N} \gamma_n^{(x)}(j)\hat{\boldsymbol{x}}_n^{(x)}(j)}{\sum_{j=1}^{N} \gamma_n^{(x)}(j)} \tag{42}$$

$$\hat{\boldsymbol{\Sigma}}_n^{(x)} = \operatorname{diag}\Big(\frac{\sum_{j=1}^{N} \gamma_n^{(x)}(j)\hat{\boldsymbol{R}}_n^{(x)}(j)}{\sum_{j=1}^{N} \gamma_n^{(x)}(j)} - \hat{\boldsymbol{\mu}}_n^{(x)}\hat{\boldsymbol{\mu}}_n^{(x)\prime}\Big) \tag{43}$$

assuming all the model parameters are updated in the same run. If only covariance matrices are to be re-estimated, Eq. 43 will be more complicated since the substitution of the new mean vector simplifies the formula as seen in Appendix C.

The update formulae for independent factor analysis can be derived in the same way. The advantage of IFA is to have effectively more different state distributions with fewer underlying Gaussian mixture components. The usefulness of IFA depends on the validity of the independence assumption which is not generally true in case of speech feature vectors.

3.4 Mixture of Observation Processes

Mixturing the observation process increases the number of model parameters significantly and therefore tying becomes important. In [15] the noise covariance matrices were chosen to be tied so that the state to observation space transforms are distinct to each mixture component. This model was called mixture of factor analysers (MFA) but it should not be mixed with the generic mixture model in this paper. In [22] the observation matrices were tied and it is called shared factor analysis (SFA) in this paper. Both the models can be illustrated by the same Bayesian network in Figure 9.

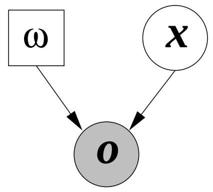

Figure 9: Bayesian network representing factor analyser with mixture of observation processes.

The MFA model provides more freedom in choosing the number of factors, k, in each analyser but the number of model parameters is much higher since M full $\boldsymbol{C}_m$ matrices have to be estimated. In SFA framework the observation noise is modelled as a mixture of Gaussians and the observation matrix is shared so that less model parameters ($M2p$ additional) have to be estimated due to the diagonal covariance matrix assumption.

The parameters for the mixture of factor analysers can be obtained after finding the component and state posteriors given the observation and the mixture component. The derivation for the M step can be found in [15]. The re-estimation formulae for individual parameters of the MFA model can be represented as follows

$$\begin{bmatrix} \hat{\boldsymbol{C}}_m & \hat{\boldsymbol{\mu}}_m^{(o)} \end{bmatrix} = \Big(\sum_{j=1}^{N} \gamma_m^{(o)}(j) \begin{bmatrix} \boldsymbol{o}_j\hat{\boldsymbol{x}}_m^{(o)\prime}(j) & \boldsymbol{o}_j \end{bmatrix}\Big)\Big(\sum_{j=1}^{N} \gamma_m^{(o)}(j) \begin{bmatrix} \hat{\boldsymbol{R}}_m^{(o)}(j) & \hat{\boldsymbol{x}}_m^{(o)}(j) \\ \hat{\boldsymbol{x}}_m^{(o)\prime}(j) & 1 \end{bmatrix}\Big)^{-1} \tag{44}$$

The derivation follows very closely to that of the normal factor analysis since the observation matrix is shared and cancels nicely during the re-estimation of the other parameters. The re-estimation of the shared observation noise covariance matrix is simple and can be represented as follows

$$\hat{\boldsymbol{\Sigma}}^{(o)} = \frac{1}{N}\sum_{j=1}^{N}\sum_{m=1}^{M}\gamma_m^{(o)}(j)\mathrm{diag}\Big(\boldsymbol{o}_j\boldsymbol{o}_j' - \Big[\,\hat{\boldsymbol{C}}_m\ \hat{\boldsymbol{\mu}}_m^{(o)}\,\Big]\Big[\,\boldsymbol{o}_j\hat{\boldsymbol{x}}_m^{(o)\prime}(j)\ \boldsymbol{o}_j\,\Big]'\Big) \tag{45}$$

This is easy to verify since the re-estimation formulae for the other parameters are consistent with the standard factor analysis framework.

Optimisation of the model parameters for the shared factor analysis is more difficult since the re-estimation formulae for any of the model parameters cannot be represented using only the sufficient statistics. Therefore the derivation follows generalised EM algorithm where in addition to the sufficient statistics, the old observation noise parameters are used in updating the observation matrix. This optimisation strategy is the same as used for the optimisation of maximum likelihood linear regression [30] transform matrix in HMM based speaker adaptation. To make the optimisation one row at a time easy, diagonal observation noise covariance matrices have to be used.

To be able to stick to the notation, let $\hat{\boldsymbol{B}} = \hat{\boldsymbol{C}}'$. The rows of the observation matrix can now be optimised through the columns of the $\hat{\boldsymbol{B}}$ matrix. In order to do that, the following statistics have to be defined

$$\boldsymbol{G}_l = \sum_{m=1}^{M}\frac{1}{\sigma_{ml}^{(o)2}}\sum_{j=1}^{N}\gamma_m^{(o)}(j)\hat{\boldsymbol{R}}_m^{(o)}(j) \tag{46}$$

$$\boldsymbol{k}_l = \sum_{m=1}^{M}\frac{1}{\sigma_{ml}^{(o)2}}\sum_{j=1}^{N}\gamma_m^{(o)}(j)(o_{jl} - \mu_{ml}^{(o)})\hat{\boldsymbol{x}}_m^{(o)}(j) \tag{47}$$

where $\sigma_{ml}^{(o)2}$ is the lth diagonal element of the observation noise covariance matrix $\boldsymbol{\Sigma}_m^{(o)}$, o_{jl} is the lth element of the observation vector $\boldsymbol{o}_j$ and $\mu_{ml}^{(o)}$ is the lth element of the observation noise mean vector $\boldsymbol{\mu}_m^{(o)}$. The columns of the $\hat{\boldsymbol{B}}$ matrix can be obtained as follows

$$\hat{\boldsymbol{b}}_l = \boldsymbol{G}_l^{-1}\boldsymbol{k}_l \tag{48}$$

over all the p rows and the new observation matrix is $\hat{\boldsymbol{C}} = \big[\,\hat{\boldsymbol{b}}_1\ \ldots\ \hat{\boldsymbol{b}}_p\,\big]'$.

The observation noise parameters can be re-estimated in the usual way apart from the simplified formula for the observation noise covariance matrix. The individual component parameters can be re-estimated as follows

$$\hat{\boldsymbol{\mu}}_m^{(o)} = \frac{\sum_{j=1}^{N}\gamma_m^{(o)}(j)(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\hat{\boldsymbol{x}}_m^{(o)}(j))}{\sum_{j=1}^{N}\gamma_m^{(o)}(j)} \tag{49}$$

$$\begin{aligned}\hat{\boldsymbol{\Sigma}}_m^{(o)} &= \frac{1}{\sum_{j=1}^{N}\gamma_m^{(o)}(j)}\sum_{j=1}^{N}\gamma_m^{(o)}(j)\mathrm{diag}\Big(\boldsymbol{o}_j\boldsymbol{o}_j' - \Big[\,\hat{\boldsymbol{C}}\ \hat{\boldsymbol{\mu}}_m^{(o)}\,\Big]\Big[\,\boldsymbol{o}_j\hat{\boldsymbol{x}}_m^{(o)\prime}(j)\ \boldsymbol{o}_j\,\Big]' \\ &\quad - \Big[\,\boldsymbol{o}_j\hat{\boldsymbol{x}}_m^{(o)\prime}(j)\ \boldsymbol{o}_j\,\Big]\Big[\,\hat{\boldsymbol{C}}\ \hat{\boldsymbol{\mu}}_m^{(o)}\,\Big]' + \Big[\,\hat{\boldsymbol{C}}\ \hat{\boldsymbol{\mu}}_m^{(o)}\,\Big]\begin{bmatrix}\hat{\boldsymbol{R}}_m^{(o)}(j) & \hat{\boldsymbol{x}}_m^{(o)}(j)\\ \hat{\boldsymbol{x}}_m^{(o)\prime}(j) & 1\end{bmatrix}\Big[\,\hat{\boldsymbol{C}}\ \hat{\boldsymbol{\mu}}_m^{(o)}\,\Big]'\Big)\end{aligned} \tag{50}$$

3.5 Decoding Cost

When using any of the factor analysis models presented above, the likelihood of the observation being generated by a given model can be obtained by using Eq. 35. In case of mixture of factor analysers it has to be weighted by the corresponding mixture priors and the resulting joint likelihoods of the observation and the mixture components have to be summed over the components. The majority of the cost involved in obtaining the likelihood is due to inverting the corresponding full covariance matrices in the observation space, $\boldsymbol{C}_m \boldsymbol{\Sigma}_n^{(x)} \boldsymbol{C}_m + \boldsymbol{\Sigma}_m^{(o)}$. Instead of inverting a p by p matrix there is a useful result from matrix algebra that converts the problem to inverting a k by k matrix as discussed later in Section 7. Furthermore, it should be noted that all the full inverse covariance matrices in the observation space can be pre-computed before starting off with the decoding.

4 Linear Dynamical Systems

Linear dynamical systems (LDS) are the simplest dynamical models with continuous state vectors. The state evolution process is a linear first-order Gauss-Markov random process and the observation process is a factor analyser. The benefits of using linear dynamical systems in speech recognition are the smooth trajectory modelling, improved covariance modelling and it allows subspace modelling [34].

Linear dynamical systems are based on continuous hidden k dimensional state vectors, $\boldsymbol{x}_t$, which evolve according to first-order Markov dynamics. A p dimensional observation vector, $\boldsymbol{o}_t$, is generated from the current state by a linear observation process. LDS can be described by the following two equations

$$\boxed{\begin{aligned} \boldsymbol{x}_{t+1} &= \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{w}, \qquad \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)}) \qquad (51) \\ \boldsymbol{o}_t &= \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{v}, \qquad \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}) \qquad (52) \end{aligned}}$$

where $\boldsymbol{A}$ is a k by k state transition matrix and $\boldsymbol{C}$ is a p by k observation matrix. Vectors $\boldsymbol{w}$ and $\boldsymbol{v}$ are called the state evolution noise and observation noise respectively, and they are independent of each other and of the values $\boldsymbol{x}_t$ and $\boldsymbol{o}_t$. Both of these noise sources are Gaussian distributed and temporally uncorrelated (white) processes; therefore, $\boldsymbol{x}_t$ is a first-order Gauss-Markov random vector process. It should be noted that the observation process is a dynamic version of the factor analysis. The model parameters are $\mathcal{M} = \{\boldsymbol{\mu}^{(i)}, \boldsymbol{\Sigma}^{(i)}, \boldsymbol{A}, \boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)}, \boldsymbol{C}, \boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}\}$. Traditionally, linear dynamical systems do not contain noise mean vectors but they have been included in the generic notation. On top of being more general, mean vectors prove to be essential when dealing with mixture models as already seen in the case of the factor analysis.

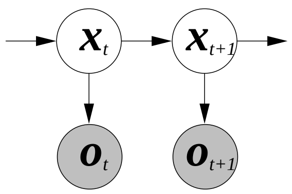

Figure 10: Dynamic Bayesian network representing a linear dynamical system.

The generative model above can be represented as a dynamic Bayesian network depicted in Figure 10. Since the noise sources are Gaussian distributed, $\boldsymbol{x}_t$ and $\boldsymbol{o}_t$ are Gaussian distributed and their conditional densities are

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{A}\boldsymbol{x}_{t-1} + \boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)}) \tag{53}$$
$$p(\boldsymbol{o}_t|\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}) \tag{54}$$

which is consistent with the conditional independencies illustrated in the figure. By assuming a Gaussian initial state density, $\boldsymbol{x}_1 \sim \mathcal{N}(\boldsymbol{\mu}^{(i)}, \boldsymbol{\Sigma}^{(i)})$, and using the Markov property, the joint density of an observation sequence, $\boldsymbol{O} = \boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T$, and state sequence, $\boldsymbol{X} = \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T$, can be represented as

$$p(\boldsymbol{X}, \boldsymbol{O}) = p(\boldsymbol{x}_1) \prod_{t=2}^{T} p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \prod_{t=1}^{T} p(\boldsymbol{o}_t|\boldsymbol{x}_t) \tag{55}$$

The number of model parameters is large in the standard LDS and reliable estimation of the model parameters becomes hard. Noting the same degeneracy regarding the state noise parameters as with the standard factor analysis, the effective number of parameters is $p+p(p+1)/2+pk+k^2+k+k(k+1)/2$ where the first two terms correspond to the observation noise statistics, the next two to the observation matrix and the state evolution matrix, respectively, and the last two to the initial state statistics. The generic model can be modified by introducing several restrictions to the model such as using diagonal noise covariance matrices and tying parameters especially when using mixture models. These restrictions and mixture models are reviewed after the parameter estimation has been presented.

4.1 EM Algorithm for LDS

Learning the parameters (system identification) of a linear dynamical system given only a sequence of observations can be carried out using again the powerful expectation maximisation algorithm [7]. The auxiliary function to be maximised can be represented as follows

$$\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = E\big\{ \log p(\boldsymbol{X}, \boldsymbol{O}|\hat{\mathcal{M}})|\boldsymbol{O}, \mathcal{M}\big\} = \int p(\boldsymbol{X}|\boldsymbol{O}, \mathcal{M}) \log p(\boldsymbol{X}, \boldsymbol{O}|\hat{\mathcal{M}}) d\boldsymbol{X} \tag{56}$$

Due to the Markov assumption in the state dynamics, the observations are dependent on the respective states and the state posteriors become dependent on the entire observation sequence. Therefore, it is essential in estimating the sufficient statistics for the M step to take the entire observation sequence into account. This must be done in two passes by first estimating the state posteriors given the observations up to the current time index; i.e., $E\{\boldsymbol{x}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t\}$ and $E\{\boldsymbol{x}_t\boldsymbol{x}_t'|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t\}$. Using these statistics, the state posteriors given the entire observation sequence can be obtained; i.e., $E\{\boldsymbol{x}_t|\boldsymbol{O}\}$ and $E\{\boldsymbol{x}_t\boldsymbol{x}_t'|\boldsymbol{O}\}$. Since both of these posteriors are Gaussians, it is rather easy to derive the recursions for the estimates using matrix algebra and the generalised forward-backward algorithm as described in Appendix E.1.

By defining $\boldsymbol{x}^{(\tau)}(t) = E\{\boldsymbol{x}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_\tau\}$, $\boldsymbol{R}^{(\tau)}(t) = E\{\boldsymbol{x}_t\boldsymbol{x}_t'|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_\tau\}$ and $\boldsymbol{\Sigma}^{(\tau)}(t) = \boldsymbol{R}^{(\tau)}(t) - \boldsymbol{x}^{(\tau)}(t)\boldsymbol{x}^{(\tau)\prime}(t)$ the forward recursion can be represented as follows

$$\begin{aligned}
\boldsymbol{x}^{(t-1)}(t) &= \boldsymbol{A}\boldsymbol{x}^{(t-1)}(t-1) + \boldsymbol{\mu}^{(x)} && (57)\\
\boldsymbol{\Sigma}^{(t-1)}(t) &= \boldsymbol{A}\boldsymbol{\Sigma}^{(t-1)}(t-1)\boldsymbol{A}' + \boldsymbol{\Sigma}^{(x)} && (58)\\
\boldsymbol{\Sigma}^{(e)}(t) &= \boldsymbol{C}\boldsymbol{\Sigma}^{(t-1)}(t)\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)} && (59)\\
\boldsymbol{K}(t) &= \boldsymbol{\Sigma}^{(t-1)}(t)\boldsymbol{C}'\boldsymbol{\Sigma}^{(e)-1}(t) && (60)\\
\boldsymbol{e}(t) &= \boldsymbol{o}_t - \boldsymbol{C}\boldsymbol{x}^{(t-1)}(t) - \boldsymbol{\mu}^{(o)} && (61)\\
\boldsymbol{x}^{(t)}(t) &= \boldsymbol{x}^{(t-1)}(t) + \boldsymbol{K}(t)\boldsymbol{e}(t) && (62)\\
\boldsymbol{\Sigma}^{(t)}(t) &= \boldsymbol{\Sigma}^{(t-1)}(t) - \boldsymbol{K}(t)\boldsymbol{C}\boldsymbol{\Sigma}^{(t-1)}(t) && (63)
\end{aligned}$$

where $\boldsymbol{x}^{(0)}(1) = \boldsymbol{\mu}^{(i)}$ and $\boldsymbol{\Sigma}^{(0)}(1) = \boldsymbol{\Sigma}^{(i)}$. The matrix $\boldsymbol{K}(t)$ is traditionally called the Kalman gain matrix and the vector $\boldsymbol{e}(t)$ is called the prediction error or the innovation vector [27, 28]. The innovation vector contains the "new" information that cannot be predicted, hence the name.

The generic scaled forward algorithm allows the likelihood of the observation sequence to be obtained via the scaling factors. The scaling factors which represent the posterior of the current observation given all the previous observations, $\kappa_t = p(\boldsymbol{o}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1})$, can be obtained by

$$\kappa_t = \mathcal{N}(\boldsymbol{e}(t); \boldsymbol{0}, \boldsymbol{\Sigma}^{(e)}(t)) \tag{64}$$

and the joint likelihood of an observation sequence is simply $p(\boldsymbol{O}) = \prod_{t=1}^{T} \kappa_t$.

The parallel backward recursion, also known as Kalman or Rauch-Tung-Streibel smoother [36, 37], is derived in Appendix E.2. By defining $\hat{\boldsymbol{x}}(t) = E\{\boldsymbol{x}_t|\boldsymbol{O}\}$ and $\hat{\boldsymbol{R}}(t) = E\{\boldsymbol{x}_t\boldsymbol{x}_t'|\boldsymbol{O}\}$ the estimates of the required statistics can now be initialised as $\hat{\boldsymbol{x}}(T) = \boldsymbol{x}^{(T)}(T)$, $\hat{\boldsymbol{\Sigma}}(T) = \boldsymbol{\Sigma}^{(T)}(T)$ and $\hat{\boldsymbol{R}}(T) = \hat{\boldsymbol{\Sigma}}(T) + \hat{\boldsymbol{x}}(T)\hat{\boldsymbol{x}}'(T)$. The rest of the estimates can be obtained by using the following Kalman smoother recursions,

$$\begin{aligned}
\boldsymbol{J}(t-1) &= \boldsymbol{\Sigma}^{(t-1)}(t-1)\boldsymbol{A}'(\boldsymbol{\Sigma}^{(t-1)}(t))^{-1} && (65)\\
\hat{\boldsymbol{x}}(t-1) &= \boldsymbol{x}^{(t-1)}(t-1) + \boldsymbol{J}(t-1)\big(\hat{\boldsymbol{x}}(t) - \boldsymbol{x}^{(t-1)}(t)\big) && (66)\\
\hat{\boldsymbol{\Sigma}}(t-1) &= \boldsymbol{\Sigma}^{(t-1)}(t-1) + \boldsymbol{J}(t-1)\big(\hat{\boldsymbol{\Sigma}}(t) - \boldsymbol{\Sigma}^{(t-1)}(t)\big)\boldsymbol{J}'(t-1) && (67)\\
\hat{\boldsymbol{\Sigma}}^{(t-1)}(t) &= \hat{\boldsymbol{\Sigma}}(t)\boldsymbol{J}'(t-1) && (68)
\end{aligned}$$

for time indices $t = T, \ldots, 2$. Then $\hat{\boldsymbol{R}}(t) = \hat{\boldsymbol{\Sigma}}(t) + \hat{\boldsymbol{x}}(t)\hat{\boldsymbol{x}}'(t)$ and $\hat{\boldsymbol{R}}^{(t-1)}(t) = \hat{\boldsymbol{\Sigma}}^{(t-1)}(t) + \hat{\boldsymbol{x}}(t)\hat{\boldsymbol{x}}'(t-1)$. This recursion is considerably more efficient than the traditional recursion presented in [38] since the estimate of the cross covariance of two consecutive state vectors, $\hat{\boldsymbol{\Sigma}}^{(t-1)}(t)$, can be obtained via one matrix multiplication in contrast to four in the traditional method. This simplification follows directly from the joint Gaussianity of the state posteriors [32].

The derivation of the re-estimation formulae for linear dynamical systems is outlined in Appendix E.3. Same re-estimation formulae can be found in [34] and can be nicely expressed by the following matrix operations

$$\begin{aligned}
\big[\,\hat{\boldsymbol{C}}\;\hat{\boldsymbol{\mu}}^{(o)}\,\big] &= \Big(\sum_{t=1}^{T}\big[\,\boldsymbol{o}_t\hat{\boldsymbol{x}}'(t)\;\;\boldsymbol{o}_t\,\big]\Big)\Big(\sum_{t=1}^{T}\begin{bmatrix}\hat{\boldsymbol{R}}(t) & \hat{\boldsymbol{x}}(t)\\ \hat{\boldsymbol{x}}'(t) & 1\end{bmatrix}\Big)^{-1} && (69)\\
\hat{\boldsymbol{\Sigma}}^{(o)} &= \frac{1}{T}\sum_{t=1}^{T}\Big(\boldsymbol{o}_t\boldsymbol{o}_t' - \big[\,\hat{\boldsymbol{C}}\;\hat{\boldsymbol{\mu}}^{(o)}\,\big]\big[\,\boldsymbol{o}_t\hat{\boldsymbol{x}}'(t)\;\;\boldsymbol{o}_t\,\big]'\Big) && (70)\\
\big[\,\hat{\boldsymbol{A}}\;\hat{\boldsymbol{\mu}}^{(x)}\,\big] &= \Big(\sum_{t=2}^{T}\big[\,\hat{\boldsymbol{R}}^{(t-1)}(t)\;\;\hat{\boldsymbol{x}}(t)\,\big]\Big)\Big(\sum_{t=2}^{T}\begin{bmatrix}\hat{\boldsymbol{R}}(t-1) & \hat{\boldsymbol{x}}(t-1)\\ \hat{\boldsymbol{x}}'(t-1) & 1\end{bmatrix}\Big)^{-1} && (71)\\
\hat{\boldsymbol{\Sigma}}^{(x)} &= \frac{1}{T-1}\sum_{t=2}^{T}\Big(\hat{\boldsymbol{R}}(t) - \big[\,\hat{\boldsymbol{A}}\;\hat{\boldsymbol{\mu}}^{(x)}\,\big]\big[\,\hat{\boldsymbol{R}}^{(t-1)}(t)\;\;\hat{\boldsymbol{x}}(t)\,\big]'\Big) && (72)\\
\hat{\boldsymbol{\mu}}^{(i)} &= \hat{\boldsymbol{x}}(1) && (73)\\
\hat{\boldsymbol{\Sigma}}^{(i)} &= \hat{\boldsymbol{R}}(1) - \hat{\boldsymbol{\mu}}^{(i)}\hat{\boldsymbol{\mu}}^{(i)\prime} && (74)
\end{aligned}$$

where the noise covariance matrices are full.

4.2 Diagonal Covariance and State Transition Matrices

To reduce the number of model parameters and to guarantee that the state transition matrix and observation matrix model the covariance structure of the data, it is possible to restrict the noise source covariance matrices to be diagonal. The only modifications to the re-estimation formulae are just taking the main diagonal elements of the re-estimated matrices as follows

$$\begin{aligned}
\hat{\boldsymbol{\Sigma}}^{(o)} &= \frac{1}{T}\sum_{t=1}^{T}\mathrm{diag}\Big(\boldsymbol{o}_t\boldsymbol{o}_t' - \big[\,\hat{\boldsymbol{C}}\;\hat{\boldsymbol{\mu}}^{(o)}\,\big]\big[\,\boldsymbol{o}_t\hat{\boldsymbol{x}}'(t)\;\;\boldsymbol{o}_t\,\big]'\Big) && (75)\\
\hat{\boldsymbol{\Sigma}}^{(x)} &= \frac{1}{T-1}\sum_{t=2}^{T}\mathrm{diag}\Big(\hat{\boldsymbol{R}}(t) - \big[\,\hat{\boldsymbol{A}}\;\hat{\boldsymbol{\mu}}^{(x)}\,\big]\big[\,\hat{\boldsymbol{R}}^{(t-1)}(t)\;\;\hat{\boldsymbol{x}}(t)\,\big]'\Big) && (76)\\
\hat{\boldsymbol{\Sigma}}^{(i)} &= \mathrm{diag}\Big(\hat{\boldsymbol{R}}(1) - \hat{\boldsymbol{\mu}}^{(i)}\hat{\boldsymbol{\mu}}^{(i)\prime}\Big) && (77)
\end{aligned}$$

In practise, it is, of course, not necessary to calculate the full covariance matrices but just the diagonal elements. The number of parameters for a diagonal covariance LDS is $2p + pk + k^2 + 2k$.

It is possible to further reduce the number of model parameters by assuming diagonal state transition matrix which corresponds to an independent state trajectory assumption provided that the covariance matrices are diagonal as well. The Kalman filter and smoother recursions do not need any modifications since they give the maximum likelihood estimates for the required statistics regardless the structure of the matrices. The covariance matrices obtained by Kalman smoother will be full matrices even if the state transition matrix is diagonal due to the full observation matrix. Therefore the re-estimation formula for the state transition matrix without state evolution noise mean vector has to be modified as follows

$$\hat{\boldsymbol{A}} = \Big(\mathrm{diag}\Big(\sum_{t=2}^{T} \hat{\boldsymbol{R}}^{(t-1)}(t)\Big)\Big)\Big(\mathrm{diag}\Big(\sum_{t=2}^{T} \hat{\boldsymbol{R}}(t-1)\Big)\Big)^{-1} \tag{78}$$

Unfortunately, the state evolution noise mean vector cannot be re-estimated simultaneously with the diagonal state transition matrix. The simultaneous re-estimation would require major changes in the notation and an inefficient form of Eq. 71 should be used. The re-estimation has to be done separately as follows

$$\begin{aligned}
\hat{\boldsymbol{A}} &= \Big(\mathrm{diag}\Big(\sum_{t=2}^{T} \hat{\boldsymbol{R}}^{(t-1)}(t) - \frac{1}{T-1}\sum_{t=2}^{T} \hat{\boldsymbol{x}}(t)\sum_{t=2}^{T} \hat{\boldsymbol{x}}'(t-1)\Big)\Big) \\
&\quad \Big(\mathrm{diag}\Big(\sum_{t=2}^{T} \hat{\boldsymbol{R}}(t-1) - \frac{1}{T-1}\sum_{t=2}^{T} \hat{\boldsymbol{x}}(t-1)\sum_{t=2}^{T} \hat{\boldsymbol{x}}'(t-1)\Big)\Big)^{-1} &(79) \\
\hat{\boldsymbol{\mu}}^{(x)} &= \frac{1}{T-1}\sum_{t=2}^{T}\Big(\hat{\boldsymbol{x}}(t) - \hat{\boldsymbol{A}}\hat{\boldsymbol{x}}(t-1)\Big) &(80)
\end{aligned}$$

It is easy to verify that Eq. 76 can be used to re-estimate the state evolution noise covariance if diagonal covariance matrices are used as well. In case of full covariance matrices, all the terms in Eq. 255 need to be used.

4.3 Mixture of Linear Dynamical Systems

In linear dynamical systems, the observation vectors are also Gaussian distributed. A Gaussian density is unimodal whereas often the feature vectors are multi-modal as noted already. Therefore, mixture models are more interesting. There are three distinct noise models present in the linear dynamical systems; the initial state, the state evolution noise and the observation noise. If mixtures were used in the state evolution noise, there would be exponential growth in the number of observation components. Propagating M mixture components through the state evolution equations leads to M^2 components after first time instant and in the end of an observation sequence there are M^T components. However, there are several approximate approaches for dealing with this exponential growth as discussed in [33]. Most of the approximate methods work well only in the filtering and the smoothing is more prone to errors. Similar exponential growth occurs in the state posteriors if mixture models are used as the observation noise or the initial state distribution.

The most generic form of a mixture of linear dynamical systems is depicted on the left-hand side of Figure 11. The observation matrix can be distributed according to a mixture model like in the case of a mixture of factor analysers in Section 3.2. The same applies for the state transition matrix for which a mixture indicator ω_t^a is used. The generative model for this model can be written as

$$
\boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{w}, \qquad \begin{aligned} \boldsymbol{w} &\sim \sum_n c_n^{(x)} \mathcal{N}(\boldsymbol{\mu}_n^{(x)}, \boldsymbol{\Sigma}_n^{(x)}) \\ \boldsymbol{A} &\sim \sum_i c_i^{(a)} \delta(\boldsymbol{A} - \boldsymbol{A}_i) \end{aligned} \tag{81}
$$

$$
\boldsymbol{o}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{v}, \qquad \begin{aligned} \boldsymbol{v} &\sim \sum_m c_m^{(o)} \mathcal{N}(\boldsymbol{\mu}_m^{(o)}, \boldsymbol{\Sigma}_m^{(o)}) \\ \boldsymbol{C} &\sim \sum_l c_l^{(c)} \delta(\boldsymbol{C} - \boldsymbol{C}_l) \end{aligned} \tag{82}
$$

As discussed above, the filtering and smoothing are not tractable with this model. Nevertheless, this model is very interesting and some approximate algorithms are mentioned below. Another interesting intractable model is illustrated on the right-hand side of Figure 11. It is often called switching Kalman filter model [33] and it combines a hidden Markov model and a LDS. Generally, the combinations are called switching state-space models [17] since the HMM states can be viewed as switching variables which choose the active parameters of the LDS. All the parameters of switching models are time dependent via the discrete state variable q_t.

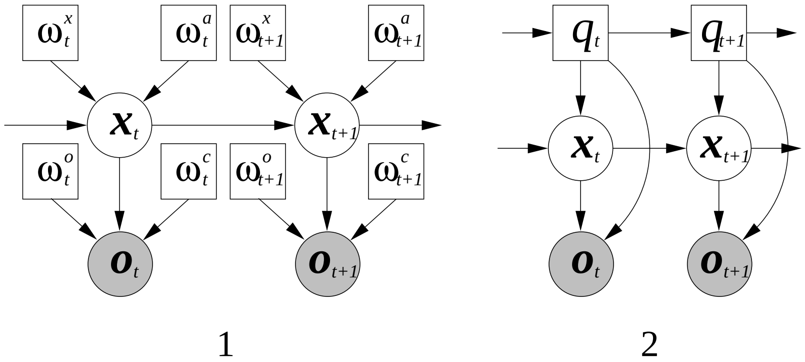

Figure 11: Dynamic Bayesian networks representing mixtures of linear dynamical systems. 1) Generic mixture of LDSs. 2) Switching Kalman filter.

There is only one way to restrict the model to be tractable but still employ a kind of mixture distribution. If the HMM states are restricted to choose the parameters from M sets and the consecutive states are forced to be equal through the entire observation sequence, the learning becomes tractable. This can be viewed as M independent LDSs working in parallel with prior likelihoods $c_m = P_\omega(m)$. Since the number of parameters grows with the factor of M, tying becomes essential to obtain reliable estimates in the re-estimation. If all parameters other than the observation noise are tied, the observation noise is crudely a mixture model. The difference to the standard mixture model is due to the mixture indicator having the same value throughout the observation sequence and therefore not permitting as flexible a model.

To optimise the parameters of mixture of M LDSs with component likelihoods, $c_m = P_\omega(m)$, the following auxiliary function has to be maximised

$$
\begin{aligned} &\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = \\ &E\{\log p(\boldsymbol{O}, \boldsymbol{X}|\hat{\mathcal{M}})|\boldsymbol{O}, \mathcal{M}\} = \sum_{m=1}^{M} P_\omega(m|\boldsymbol{O}, \mathcal{M}) E\{\log p(\boldsymbol{O}, \boldsymbol{X}|m, \hat{\mathcal{M}})|\boldsymbol{O}, m, \mathcal{M}\} \end{aligned} \tag{83}
$$

The E step requires M passes of Kalman filter and smoother, and the likelihoods of the observation sequence given the model, $p(\boldsymbol{O}|\mathcal{M}^m)$, are obtained. The component posteriors are then

$$\gamma_m = P_\omega(m|\boldsymbol{O},\mathcal{M}^1,\ldots,\mathcal{M}^M) = \frac{c_m p(\boldsymbol{O}|\mathcal{M}^m)}{\sum_{l=1}^{M} c_l p(\boldsymbol{O}|\mathcal{M}^l)} \tag{84}$$

The observation process is very similar to the shared factor analysis discussed in Section 3.2 apart from the component posteriors which are conditioned by the entire observation sequence. The re-estimation formula for the observation matrix can be derived in the usual fashion by first defining $\hat{\boldsymbol{B}} = \hat{\boldsymbol{C}}'$ and the following statistics

$$\boldsymbol{G}_l = \sum_{m=1}^{M} \frac{1}{\sigma_{ml}^{(o)2}} \gamma_m \sum_{t=1}^{T} \hat{\boldsymbol{R}}_m(t) \tag{85}$$

$$\boldsymbol{k}_l = \sum_{m=1}^{M} \frac{1}{\sigma_{ml}^{(o)2}} \gamma_m \sum_{t=1}^{T} (o_{tl} - \mu_{ml}^{(o)}) \hat{\boldsymbol{x}}_m(t) \tag{86}$$

where $\sigma_{ml}^{(o)2}$ is the lth diagonal element of the observation noise covariance matrix $\boldsymbol{\Sigma}_m^{(o)}$, o_{tl} is the lth element of the observation vector $\boldsymbol{o}_t$ and $\mu_{ml}^{(o)}$ is the lth element of the observation noise mean vector $\boldsymbol{\mu}_m^{(o)}$. Then the columns of the $\hat{\boldsymbol{B}}$ matrix can be obtained as follows

$$\hat{\boldsymbol{b}}_l = \boldsymbol{G}_l^{-1} \boldsymbol{k}_l \tag{87}$$

over all the p rows and the new observation matrix is $\hat{\boldsymbol{C}} = \begin{bmatrix} \hat{\boldsymbol{b}}_1 & \ldots & \hat{\boldsymbol{b}}_p \end{bmatrix}'$. The component posteriors are also the maximum likelihood estimates for the new component prior likelihoods, $\hat{c}_m = \gamma_m$. Since the priors are subject to constraint, $\sum_m c_m = 1$, this can be derived using Lagrange multipliers as in Appendix C.2.

The re-estimation formulae for the observation noise parameters can be obtained easily noting that the component posteriors cancel since they do not depend on the time, t. The formulae can be represented as follows

$$\hat{\boldsymbol{\mu}}_m^{(o)} = \frac{1}{T} \sum_{t=1}^{T} (\boldsymbol{o}_t - \hat{\boldsymbol{C}} \hat{\boldsymbol{x}}_m(t)) \tag{88}$$

$$\begin{aligned} \hat{\boldsymbol{\Sigma}}_m^{(o)} = \frac{1}{T} \sum_{t=1}^{T} \mathrm{diag}\Big(& \boldsymbol{o}_t \boldsymbol{o}_t' - \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}_m^{(o)} \end{bmatrix} \begin{bmatrix} \boldsymbol{o}_t \hat{\boldsymbol{x}}_m'(t) & \boldsymbol{o}_t \end{bmatrix}' \\ & - \begin{bmatrix} \boldsymbol{o}_t \hat{\boldsymbol{x}}_m'(t) & \boldsymbol{o}_t \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}_m^{(o)} \end{bmatrix}' + \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}_m^{(o)} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{R}}_m(t) & \hat{\boldsymbol{x}}_m(t) \\ \hat{\boldsymbol{x}}_m'(t) & 1 \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}_m^{(o)} \end{bmatrix}' \Big) \end{aligned} \tag{89}$$

The re-estimation formulae for the state evolution process is simply obtained using the component posteriors as weights

$$\begin{bmatrix} \hat{\boldsymbol{A}} & \hat{\boldsymbol{\mu}}^{(x)} \end{bmatrix} = \Big(\sum_{t=2}^{T} \sum_{m=1}^{M} \gamma_m \begin{bmatrix} \hat{\boldsymbol{R}}_m^{(t-1)}(t) & \hat{\boldsymbol{x}}_m(t) \end{bmatrix} \Big) \Big(\sum_{t=2}^{T} \sum_{m=1}^{M} \gamma_m \begin{bmatrix} \hat{\boldsymbol{R}}_m(t-1) & \hat{\boldsymbol{x}}_m(t-1) \\ \hat{\boldsymbol{x}}_m'(t-1) & 1 \end{bmatrix} \Big)^{-1} \tag{90}$$

$$\hat{\boldsymbol{\Sigma}}^{(x)} = \frac{1}{T-1} \sum_{t=2}^{T} \sum_{m=1}^{M} \gamma_m \Big(\hat{\boldsymbol{R}}_m(t) - \begin{bmatrix} \hat{\boldsymbol{A}} & \hat{\boldsymbol{\mu}}^{(x)} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{R}}_m^{(t-1)}(t) & \hat{\boldsymbol{x}}_m(t) \end{bmatrix}' \Big) \tag{91}$$

$$\hat{\boldsymbol{\mu}}^{(i)} = \sum_{m=1}^{M} \gamma_m \hat{\boldsymbol{x}}_m(1) \tag{92}$$

$$\hat{\boldsymbol{\Sigma}}^{(i)} = \sum_{m=1}^{M} \gamma_m \hat{\boldsymbol{R}}_m(1) - \hat{\boldsymbol{\mu}}^{(i)} \hat{\boldsymbol{\mu}}^{(i)\prime} \tag{93}$$

where the noise covariance matrices are full but can be diagonalised as usual.

4.4 Sampling Approaches for Mixture Models

As noted above, using mixture models as the noise processes of a linear dynamical system leads to exponential growth in the state posteriors. Recently, in statistical signal processing [8, 9, 19] and computer vision [25, 24] Monte Carlo sampling methods have been employed in filtering problems with non-linear non-Gaussian models. These methods are based on a "particle cloud" representation in the posterior propagation. A set of N points sampled from the prior distribution are propagated through the system and the resulting N^2 posterior samples are then re-sampled and weighted according to importance functions obtained from the model. Particle filtering can be used instead of Kalman filtering but the smoothing approaches presented in [8, 9] perform poorly in estimating $p(\boldsymbol{x}_t|\boldsymbol{O})$ when $T-t$ is significant. More stable smoothing estimates have to be studied and developed.

An alternative way to deal with the mixture models is to propagate the posterior as usual resulting into M^2 mixture components and then apply Gaussian selection [13] to project these M^2 components back to M components. This is also an approximate method but it takes into account the fact that all the posteriors are always mixtures of Gaussians. Unfortunately, it is not possible to guarantee that the state posteriors obtained this way would increase the likelihood of the data after re-estimation of the model parameters but the validity of the approximation should be evaluated experimentally.

5 Hidden Markov Models

Hidden Markov models are based on a finite N_s state machine with discrete initial state and transition probabilities, π_j and a_{ij} respectively, as well as state dependent observation distributions $b_j(\boldsymbol{o}_t) = p(\boldsymbol{o}_t|j)$. Mixtures of M Gaussians with mixture weights $c_{jm} = P_\omega(m|j)$ are used as the generative model for each state in this paper so that a HMM can be described by the set of parameters $\mathcal{M} = \{\forall j, i \in (1, N_s), m \in (1, M) : \pi_j, a_{ij}, c_{jm}, \boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm}\}$. In speech recognition the initial state and the transition probabilities are restricted to form so called left to right models but in this paper a general formulation is used.

One of the main advantages of HMMs is the efficient segmentation which is carried out using Viterbi algorithm [41]. Training of HMMs is also very efficient using the EM framework and it is often referred to as the Baum-Welch algorithm [2, 3]. The major drawbacks are discontinuous state transitions which prevent trajectory interpretation of the feature elements, the state conditional independence assumption which states that given the state, the subsequent observation vectors are independent, and rather weak covariance modelling if diagonal covariance matrices are used.

Using mixture models as the state dependent observation distributions has yielded better performance in HMM based systems especially in speech recognition. There are three major reasons for that. Firstly, mixtures can model multi-modal distributions that might result from source classes with more than one distinct acoustic characteristics. Secondly, mixtures can model spatial correlation in the feature vectors and therefore several distributions with diagonal covariance matrices can approximate one distribution with full covariance matrix with fewer parameters. Thirdly, mixtures can be used to model non-symmetric distributions. In the limit Gaussian mixture models can represent arbitrary distributions that fulfil certain continuity requirements.

The p dimensional observation vector likelihood $b_j(\boldsymbol{o}_t)$ is defined as the following mixture of Gaussians

$$b_j(\boldsymbol{o}_t) = p(\boldsymbol{o}_t|j) = \sum_{m=1}^{M} c_{jm}\mathcal{N}(\boldsymbol{o}_t; \boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm}) \tag{94}$$

$$\tag{95}$$

where $\boldsymbol{\mu}_{jm}$ and $\boldsymbol{\Sigma}_{jm}$ are the mean vector and covariance matrix associated with the state $q_t = j$ and the mixture component $\omega = m$. The simplest continuous single mixture HMM is illustrated as a dynamic Bayesian network on the left-hand side of Figure 12. On the right-hand side is a mixture of Gaussians HMM.

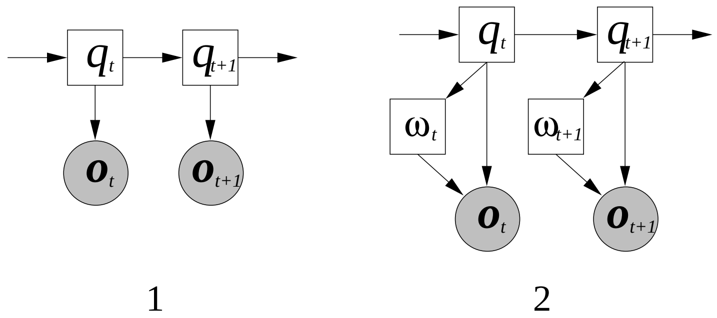

Figure 12: Bayesian networks representing hidden Markov models. 1) Single Gaussian continuous density HMM. 2) Mixture of Gaussians continuous density HMM.

The joint likelihood of an observation sequence, $\boldsymbol{O} = \boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T$, and a HMM state sequence, $Q = q_1, q_2, \ldots, q_T$, can be represented as

$$p(\boldsymbol{O}, Q) = P(q_1) \prod_{t=2}^{T} P(q_t|q_{t-1}) \prod_{t=1}^{T} p(\boldsymbol{o}_t|q_t) \tag{96}$$

where $P(q_1) = \pi_{q_1}$ are the initial state probabilities and $P(q_t|q_{t-1}) = a_{q_{t-1}q_t}$ are the state transition probabilities.

5.1 EM Algorithm for HMM

Learning the model parameters of hidden Markov models given only an observation sequence or a set of sequences can be carried out using the powerful expectation maximisation algorithm also known as Baum-Welch algorithm. The auxiliary function to be maximised can be represented as follows

$$\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = E\{\log p(\boldsymbol{O}, Q|\hat{\mathcal{M}})|\boldsymbol{O}, \mathcal{M}\} = \sum_{\{Q_T\}} P(Q|\boldsymbol{O}, \mathcal{M}) \log p(\boldsymbol{O}, Q|\hat{\mathcal{M}}) \tag{97}$$

Since the state posteriors depend on discrete state transition probabilities, they are not Gaussians any more. Due to the finite number of possible state sequences, efficient likelihood calculation requires the use of dynamic programming. The sufficient statistics $\gamma_j(t) = P_{q_t}(j|\boldsymbol{O}, \mathcal{M})$ and $\xi_{ij}(t) = P_{q_{t-1},q_t}(i, j|\boldsymbol{O}, \mathcal{M})$ for the M step can be obtained using the familiar forward-backward algorithm for HMMs [35]. The forward algorithm starts off as follows

$$\alpha_j(1) = \pi_j b_j(\boldsymbol{o}_1) \tag{98}$$

where $\alpha_j(t) = p(j, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_t)$. The rest of the forward variables are obtained by the following recursion

$$\alpha_j(t) = b_j(\boldsymbol{o}_t) \sum_{i=1}^{N_s} a_{ij}\alpha_i(t-1) \tag{99}$$

$$\tag{100}$$

The joint likelihood of the observation sequence can be obtained using the forward variables at time T; i.e., $p(\boldsymbol{O}) = \sum_{j=1}^{N_s} \alpha_j(T)$.

The backward algorithm starts off with $\beta_i(T) = 1$ and the rest of the backward variables are obtained recursively as follows

$$\beta_i(t-1) = \sum_{j=1}^{N_s} a_{ij} b_j(\boldsymbol{o}_t)\beta_j(t) \tag{101}$$

where $\beta_i(t) = p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T|i)$.

Now, the sufficient statistics can be obtained easily as

$$\gamma_j(t) = \frac{1}{p(\boldsymbol{O})}\alpha_j(t)\beta_j(t) \tag{102}$$

$$\xi_{ij}(t) = \frac{1}{p(\boldsymbol{O})}\alpha_i(t-1)a_{ij}b_t(\boldsymbol{o}_t)\beta_j(t) \tag{103}$$

To update the mixture components the mixture posteriors $\gamma_{jm}(t) = P_{q_t,\omega}(j, m|\boldsymbol{O}, \mathcal{M})$ are needed. Since the mixture components can be regarded as additional states [31], the joint likelihood of being in state j and mixture component m is

$$\gamma_{jm}(t) = \frac{1}{p(\boldsymbol{O})}c_{jm}b_{jm}(\boldsymbol{o}_t)\sum_{i=1}^{N_s} a_{ij}\alpha_i(t-1)\beta_j(t) \tag{104}$$

where $b_{jm}(\boldsymbol{o}_t)$ is the posterior of the observation $\boldsymbol{o}_t$ given the state j and the mixture component m.

The traditional derivation of the Baum-Welch algorithm can be found in [35]. A derivation more consistent with the notation used in this paper can be found in Appendix F by replacing the state dependent observation posteriors properly, and using $\boldsymbol{o}_t$ and $\boldsymbol{o}_t\boldsymbol{o}_t'$ instead of $\hat{\boldsymbol{x}}_t$ and $\hat{\boldsymbol{R}}_t$, respectively. The re-estimation formulae can be represented as follows

$$\hat{\pi}_j = \frac{\gamma_j(1)}{\sum_{i=1}^{N_s}\gamma_i(1)} \tag{105}$$

$$\hat{a}_{ij} = \frac{\sum_{t=2}^{T}\xi_{ij}(t)}{\sum_{t=2}^{T}\gamma_i(t-1)} \tag{106}$$

$$\hat{c}_{jm} = \frac{\sum_{t=1}^{T}\gamma_{jm}(t)}{\sum_{t=1}^{T}\gamma_j(t)} \tag{107}$$

$$\hat{\boldsymbol{\mu}}_{jm} = \frac{\sum_{t=1}^{T}\gamma_{jm}(t)\boldsymbol{o}_t}{\sum_{t=1}^{T}\gamma_{jm}(t)} \tag{108}$$

$$\hat{\boldsymbol{\Sigma}}_{jm} = \frac{\sum_{t=1}^{T}\gamma_{jm}(t)\boldsymbol{o}_t\boldsymbol{o}_t'}{\sum_{t=1}^{T}\gamma_{jm}(t)} - \hat{\boldsymbol{\mu}}_j\hat{\boldsymbol{\mu}}_j' \tag{109}$$

provided all the parameters are update in the same iteration. Otherwise, the update formula 109 for the covariance matrices has to be modified slightly.

To reduce the number of model parameters and the computational complexity, the covariance matrices can be assumed to be diagonal. The validity of the assumption has been argued since there is often some spatial correlation between the feature vectors obtained via currently popular parameterisation methods. Instead of full covariance matrices the correlation of the feature vectors can be modelled, e.g., using semi-tied full covariance matrices [11, 12]. The number of parameters per state in a standard diagonal covariance matrix HMM in speech recognition is $(M-1)+M2p+1$. The first term corresponds to the component priors, the second term to the state distribution parameters and the last to the state transition probability of a left to right HMM.

5.2 Factor Analysed Hidden Markov Models

Factor analysis was combined with HMMs in [40] where all the covariance matrices are factored according to a standard factor analysis model presented in Section 3. This model and the corresponding mixture model can be illustrated as the dynamic Bayesian networks in Figure 13. It was argued that fewer model parameters than with full covariance systems can be used without significantly lowering the system performance. Nevertheless, the evaluation was done using simple alpha-digit and town name tasks. The number of model parameters in this case is huge since a separate observation matrix is attached to every component in the HMM system.

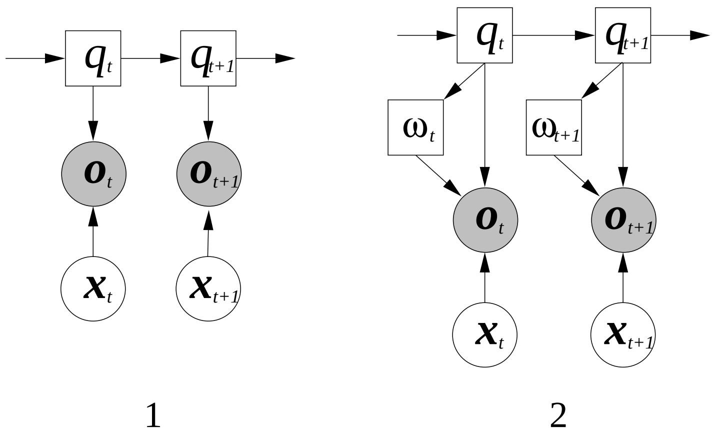

Figure 13: Bayesian networks representing hidden Markov models with factor analysed covariance matrices. 1) Single Gaussian continuous density case. 2) Mixture of Gaussians continuous density case.

Factor analysed HMMs (FAHMM) described in this paper are dynamic versions of the factor analysers with mixtures of factors and shared factor analysis presented in Sections 3.3 and 3.4. FAHMM is based on a HMM as a state evolution process and a factor analyser as an observation process. All the benefits of conventional HMMs are present and a better covariance modelling is achieved since the assumption of spatially uncorrelated feature vectors is removed. FAHMM allows also subspace modelling. The drawbacks of FAHMM include virtually the same as HMMs apart from the weak spatial covariance modelling. The generative model of FAHMM can be represented as follows

$$\boldsymbol{x}_t \sim \mathcal{M}^{hmm}, \qquad \mathcal{M}^{hmm} = \{a_{ij}, c_{jn}^{(x)}, \boldsymbol{\mu}_{jn}^{(x)}, \boldsymbol{\Sigma}_{jn}^{(x)}\} \tag{110}$$

$$\boldsymbol{o}_t = \boldsymbol{C}_t \boldsymbol{x}_t + \boldsymbol{v}_t, \qquad \begin{aligned} \boldsymbol{v}_t &\sim \sum_m c_{jm}^{(o)} \mathcal{N}(\boldsymbol{\mu}_{jm}^{(o)}, \boldsymbol{\Sigma}_{jm}^{(o)}) \\ \boldsymbol{C}_t &\sim \sum_l c_{jl}^{(c)} \delta(\boldsymbol{C} - \boldsymbol{C}_{jl}) \end{aligned} \tag{111}$$

where the observation noise distributions and the observation matrices are state dependent. The observation matrices are delta distributed to guarantee the model to be a generalised linear Gaussian model as discussed in the case of a mixture of factor analysers in Section 3.2. Only a single component observation matrix distribution is used later in this paper. Therefore the observation matrix prior cancels out in the formulation. The dynamic Bayesian network corresponding to this model is illustrated in Figure 14.

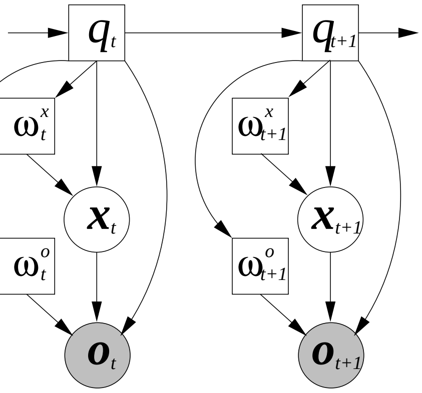

Figure 14: Bayesian network representing a generic factor analysed hidden Markov model.

Since there are now two mixtures of Gaussians, the mixture weights for the state vector distributions are denoted as $c_{jn}^{(x)} = P_{\omega^x|q_t}(n|j)$ and for the observation noise distributions as $c_{jm}^{(o)} = P_{\omega^o|q_t}(m|j)$. The model parameters of FAHMM are $\mathcal{M} = \{\forall j,i \in (1,N_s), n \in (1,M^{(x)}), m \in (1,M^{(o)}) : \pi_j, a_{ij}, c_{jn}^{(x)}, \boldsymbol{\mu}_{jn}^{(x)}, \boldsymbol{\Sigma}_{jn}^{(x)}, \boldsymbol{C}_j, c_{jm}^{(o)}, \boldsymbol{\mu}_{jm}^{(o)}, \boldsymbol{\Sigma}_{jm}^{(o)}\}$. Since the reliable estimation of all the parameters requires a lot of data, sharing becomes important. The general model allows arbitrary sharing of the parameters to be used in many different levels. For example the observation matrix can be tied between several states, several models or even globally.

The joint likelihood of an observation sequence, $\boldsymbol{O} = \boldsymbol{o}_1, \boldsymbol{o}_2, \dots, \boldsymbol{o}_T$, state vector sequence, $\boldsymbol{X} = \boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_T$, and HMM state sequence, $Q = q_1, q_2, \dots, q_T$, can be represented as

$$p(\boldsymbol{O}, \boldsymbol{X}, Q) = P(q_1) \prod_{t=2}^{T} P(q_t|q_{t-1}) \prod_{t=1}^{T} p(\boldsymbol{x}_t|q_t) p(\boldsymbol{o}_t|\boldsymbol{x}_t) \tag{112}$$

where $P(q_1) = \pi_{q_1}$ is the initial state probability and $P(q_t|q_{t-1}) = a_{q_{t-1}q_t}$ are the state transition probabilities.

5.3 EM Algorithm for FAHMM

In the expectation maximisation algorithm for factor analysed HMM the following auxiliary function has to be maximised

$$\begin{aligned} \mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) &= E\{\log p(\boldsymbol{O}, \boldsymbol{X}, Q|\hat{\mathcal{M}})|\boldsymbol{O}, \mathcal{M}\} \\ &= \sum_{\{Q_T\}} \int p(\boldsymbol{X}|\boldsymbol{O}, Q, \mathcal{M}) P(Q|\boldsymbol{O}, \mathcal{M}) \log p(\boldsymbol{O}, \boldsymbol{X}, Q|\hat{\mathcal{M}}) d\boldsymbol{X} \end{aligned} \tag{113}$$

where the posteriors for both the state vector and the discrete HMM state are needed.

The posterior probability of being in state j at time t $\gamma_j(t) = P_{q_t}(j|\boldsymbol{O}, \mathcal{M})$, and being in state j at time t and in state i at time $t-1$ $\xi_{ij}(t) = P_{q_{t-1},q_t}(i,j|\boldsymbol{O}, \mathcal{M})$ for the transition parameter re-estimation can be obtained using the same forward-backward algorithm as for HMMs presented in the previous section and using the following posterior of an observation

$$b_j(\boldsymbol{o}_t) = \sum_{m=1}^{M^{(o)}} c_{jm}^{(o)} \sum_{n=1}^{M^{(x)}} c_{jn}^{(x)} \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}_j \boldsymbol{\mu}_{jn}^{(x)} + \boldsymbol{\mu}_{jm}^{(o)}, \boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}_j' + \boldsymbol{\Sigma}_{jm}^{(o)}) \tag{114}$$

The posterior of being in state j, state-space component n and observation space component m at time t $\gamma_{jmn}(t)$ is needed for the re-estimation of the other parameters. This can be obtained as in the standard HMMs using the forward and backward variables as follows

$$\gamma_{jmn}(t) = \frac{1}{p(\boldsymbol{O})} c_{jm}^{(o)} c_{jn}^{(x)} b_{jmn}(\boldsymbol{o}_t) \sum_{i=1}^{N_s} a_{ij} \alpha_i(t-1) \beta_j(t) \tag{115}$$

where $b_{jmn}(\boldsymbol{o}_t)$ is the posterior of an observation given both the mixture components; i.e., the Gaussian in Eq. 114.

The first and second order state vector posteriors are needed in the update formulae. The posteriors given both mixture components can be obtained as follows

$$\hat{\boldsymbol{x}}_{jmn}(t) = \boldsymbol{\mu}_{jn}^{(x)} + \boldsymbol{K}_{jmn}(\boldsymbol{o}_t - \boldsymbol{C}_j \boldsymbol{\mu}_{jn}^{(x)} - \boldsymbol{\mu}_{jm}^{(o)}) \tag{116}$$

$$\hat{\boldsymbol{R}}_{jmn}(t) = \boldsymbol{\Sigma}_{jn}^{(x)} - \boldsymbol{K}_{jmn} \boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} + \hat{\boldsymbol{x}}_{jmn}(t) \hat{\boldsymbol{x}}'_{jmn}(t) \tag{117}$$

where $\boldsymbol{K}_{jmn} = \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}'_j (\boldsymbol{C}_j \boldsymbol{\Sigma}_{jn}^{(x)} \boldsymbol{C}'_j + \boldsymbol{\Sigma}_{jm}^{(o)})^{-1}$. Now, $\hat{\boldsymbol{x}}_{jmn}(t)$ corresponds to the expected state vector given the state, the mixture components, the observation sequence and the old model parameters $E\{\boldsymbol{x}_t | j, m, n, \boldsymbol{O}, \mathcal{M}\}$. The similarity to the mixtures of factors in Section 3.3 should be noted.

Re-estimation formulae are derived in Appendix F. The formulae for the state parameters are very similar to the normal HMM parameter estimation and can be written as follows

$$\hat{\pi}_j = \frac{\gamma_j(1)}{\sum_{i=1}^{N_s} \gamma_i(1)} \tag{118}$$

$$\hat{a}_{ij} = \frac{\sum_{t=2}^{T} \xi_{ij}(t)}{\sum_{t=2}^{T} \gamma_i(t-1)} \tag{119}$$

$$\hat{c}_{jn}^{x} = \frac{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t)}{\sum_{t=1}^{T} \gamma_j(t)} \tag{120}$$

$$\hat{\boldsymbol{\mu}}_{jn}^{(x)} = \frac{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t) \hat{\boldsymbol{x}}_{jmn}(t)}{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t)} \tag{121}$$

$$\hat{\boldsymbol{\Sigma}}_{jn}^{(x)} = \operatorname{diag}\Big(\frac{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t) \hat{\boldsymbol{R}}_{jmn}(t)}{\sum_{t=1}^{T} \sum_{m=1}^{M^{(o)}} \gamma_{jmn}(t)} - \hat{\boldsymbol{\mu}}_{jn}^{(x)} \hat{\boldsymbol{\mu}}_{jn}^{(x)\prime} \Big) \tag{122}$$

The observation parameter update formulae are the dynamic version of shared factor analysis and a new matrix $\hat{\boldsymbol{B}}_j = \hat{\boldsymbol{C}}_j'$ can be obtained defining the following statistics

$$\boldsymbol{G}_{jl} = \sum_{m=1}^{M^{(o)}} \frac{1}{\sigma_{jml}^{(o)2}} \sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t) \hat{\boldsymbol{R}}_{jmn}(t) \tag{123}$$

$$\boldsymbol{k}_{jl} = \sum_{m=1}^{M^{(o)}} \frac{1}{\sigma_{jml}^{(o)2}} \sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)(o_{tl} - \mu_{jml}^{(o)}) \hat{\boldsymbol{x}}_{jmn}(t) \tag{124}$$

where $\sigma_{jml}^{(o)2}$ is the lth diagonal element of the observation noise covariance matrix $\boldsymbol{\Sigma}_{jm}^{(o)}$, o_{tl} is the lth element of the observation vector $\boldsymbol{o}_t$ and $\mu_{jml}^{(o)}$ is the lth elements of the observation noise mean vector $\boldsymbol{\mu}_{jm}^{(o)}$. The columns of the $\hat{\boldsymbol{B}}$ matrix can be obtained as follows

$$\hat{\boldsymbol{b}}_{jl} = \boldsymbol{G}_{jl}^{-1} \boldsymbol{k}_{jl} \tag{125}$$

over all the p rows and the new observation matrix is $\hat{\boldsymbol{C}}_j = \left[\, \hat{\boldsymbol{b}}_{j1} \, \ldots \, \hat{\boldsymbol{b}}_{jp} \, \right]'$.

The re-estimation formulae for the observation noise parameters can be obtained in the usual way as follows

$$\hat{c}_{jm}^{(o)} = \frac{\sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)}{\sum_{t=1}^{T} \gamma_j(t)} \tag{126}$$

$$\hat{\boldsymbol{\mu}}_{jm}^{(o)} = \frac{\sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)(\boldsymbol{o}_t - \hat{\boldsymbol{C}}_j \hat{\boldsymbol{x}}_{jmn}(t))}{\sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)} \tag{127}$$

$$\begin{aligned} \hat{\boldsymbol{\Sigma}}_{jm}^{(o)} = & \frac{1}{\sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)} \sum_{t=1}^{T} \sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t) \mathrm{diag}\Big(\boldsymbol{o}_t \boldsymbol{o}_t' - \left[\, \hat{\boldsymbol{C}}_j \;\; \hat{\boldsymbol{\mu}}_{jm}^{(o)} \, \right] \left[\, \boldsymbol{o}_t \hat{\boldsymbol{x}}_{jmn}'(t) \;\; \boldsymbol{o}_t \, \right]' \\ & - \left[\, \boldsymbol{o}_t \hat{\boldsymbol{x}}_{jmn}'(t) \;\; \boldsymbol{o}_t \, \right] \left[\, \hat{\boldsymbol{C}}_j \;\; \hat{\boldsymbol{\mu}}_{jm}^{(o)} \, \right]' + \left[\, \hat{\boldsymbol{C}}_j \;\; \hat{\boldsymbol{\mu}}_{jm}^{(o)} \, \right] \begin{bmatrix} \hat{\boldsymbol{R}}_{jmn}(t) & \hat{\boldsymbol{x}}_{jmn}(t) \\ \hat{\boldsymbol{x}}_{jmn}'(t) & 1 \end{bmatrix} \left[\, \hat{\boldsymbol{C}}_j \;\; \hat{\boldsymbol{\mu}}_{jm}^{(o)} \, \right]' \Big) \end{aligned} \tag{128}$$

Now, diagonal covariance matrices have to be used to avoid a vast amount of parameters to be optimised and to make the easy observation parameter update possible. The diagonality of the HMM state dependent covariance matrices is not such a bad assumption anymore since the observation matrix is used to model the spatial covariance structure of the observations. The number of parameters per state in a FAHMM in speech recognition is $(M^{(x)} - 1)(1 + 2k) + pk + (M^{(o)} - 1) + M^{(o)}2p + 1$. The first term correspond to the state-space mixture priors and HMM state distribution statistics. One state-space distribution component can be again omitted due to the degeneracy of the factor analysis model as discussed in Section 3.3. The second term corresponds to the state dependent observation matrix, the third to the observation noise priors and the fourth for the distribution. The last term corresponds to the transition probability of a left to right HMM.

5.4 Relationship Between FAHMM and STC

As noted in the section about factor analysis the observation noise spans the $p - k$ nuisance dimensions making it a significant part of the model when $k < p$. If the dimensionality of the observation and the state-space are the same, the observation noise can be cancelled without losing the identifiability of the system. However, in that case the observation equation becomes deterministic and it does not have any influence on the joint likelihood of the observation and state sequences. The re-estimation formulae change their form dramatically as well.

Since the observation equation is deterministic, the transform $\boldsymbol{C}_j$ can be subsumed to the state dependent HMM output covariance matrix and the usual semi-tied full covariance matrix [12] system results with a transform class for every unique state in the system. This model can also be regarded as a noiseless independent factor analysis [1] with the factorial mixture model replaced by a standard mixture of Gaussians. Since arbitrary sharing is possible in the FAHMM framework, the transform classes can be defined more reasonably. The optimisation and further characteristics of semi-tied covariance matrices is discussed in the following section dealing with linear discriminant analysis.

6 Linear Discriminant Analysis

An alternative dimension reduction scheme to factor analysis is linear discriminant analysis [26]. LDA is based on a linear transformation which projects the samples so that the within class variance is minimised and the between class variance maximised. Provided the classes are separable in some lower dimensional space, LDA finds the optimal projection and thus can be regarded as a dimension reduction scheme. However, not much is known about the generalisation of LDA from two dimensional case and the performance of such a scheme must be evaluated experimentally.

In traditional linear discriminant analysis, the within class covariance matrices are assumed to be the same. In [29] an extension to LDA was presented in which the restriction of the equal covariance matrices was removed and it is called heteroscedastic LDA (HLDA). The generative model for LDA can be represented in the framework of this paper by allowing more general densities at the state process than a single Gaussian. For example, mixture of Gaussians can represent multiple classes with mean vectors as the cluster centres.

The traditional linear discriminant analysis is illustrated in Figure 15 where two classes in p dimensional space ($p = 2$ in the figure) are not easy to separate along the axis in the original space. Fortunately, there exists a transformation to lower dimensional space ($k = 1$ in the figure) where the classes are easily separable. Dimension $\boldsymbol{x}_1$ in the figure represents the new space and dimension $\boldsymbol{x}_2$ represents the nuisance dimension where both classes are represented by a single Gaussian.

The optimisation of a static HLDA model parameters culminates into finding the new transformation matrix $\hat{\boldsymbol{C}}$. Defining $\hat{\boldsymbol{B}} = (\hat{\boldsymbol{C}}^{-1})'$, the transformation can be optimised by maximising the following auxiliary function

$$\begin{aligned}\mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}}) = & \frac{1}{2}\sum_{j,m}\gamma_m(j)\log\left(\frac{\text{abs}(|\hat{\boldsymbol{B}}'|)^2}{\text{abs}(|\text{diag}(\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{W}_m\hat{\boldsymbol{B}}_{[k]})|)\text{abs}(|\text{diag}(\hat{\boldsymbol{B}}'_{[p-k]}\hat{\boldsymbol{\Sigma}}\hat{\boldsymbol{B}}_{[p-k]})|)}\right) \\ & -\frac{p}{2}\sum_{j,m}\gamma_m(j)\big(\log(2\pi)+1\big)\end{aligned} \tag{129}$$

where $\gamma_m(j)$ denotes the posterior of the mixture component m given the jth observation and the current model set. The sufficient statistics $\boldsymbol{W}_m$ and $\hat{\boldsymbol{\Sigma}}$ are defined as follows

$$\boldsymbol{W}_m = \frac{\sum_j \gamma_m(j)\big(\boldsymbol{o}_j - \hat{\boldsymbol{\mu}}_m\big)\big(\boldsymbol{o}_j - \hat{\boldsymbol{\mu}}_m\big)'}{\sum_j \gamma_m(j)} \tag{130}$$

$$\hat{\boldsymbol{\Sigma}} = \frac{\sum_{j,m} \gamma_m(j)\big(\hat{\boldsymbol{\mu}}_m - \hat{\boldsymbol{\mu}}\big)\big(\hat{\boldsymbol{\mu}}_m - \hat{\boldsymbol{\mu}}\big)'}{\sum_{j,m} \gamma_m(j)} + \frac{\sum_{j,m} \gamma_m(j)\big(\boldsymbol{o}_j - \hat{\boldsymbol{\mu}}_m\big)\big(\boldsymbol{o}_j - \hat{\boldsymbol{\mu}}_m\big)'}{\sum_{j,m} \gamma_m(j)} \tag{131}$$

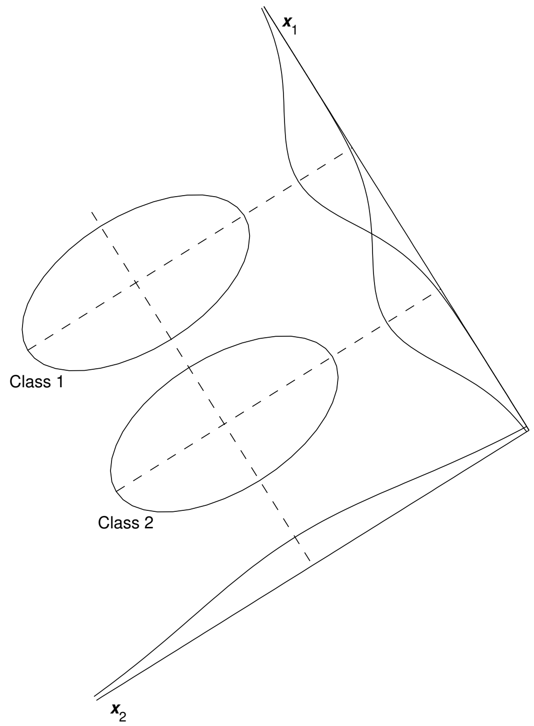

Figure 15: Linear discriminant analysis in case of two classes for which the original axes orientation is not optimal.

where

$$\hat{\boldsymbol{\mu}}_m = \frac{\sum_j \gamma_m(j)\boldsymbol{o}_j}{\sum_j \gamma_m(j)} \tag{132}$$

$$\hat{\boldsymbol{\mu}} = \frac{\sum_{j,m} \gamma_m(j)\boldsymbol{o}_j}{\sum_{j,m} \gamma_m(j)} \tag{133}$$

There exists a simple iterative scheme to optimise the columns of the matrix $\hat{\boldsymbol{B}}$ which was presented in [12]. The ML solutions for the state density mean vector and covariance matrix are

$$\hat{\boldsymbol{\mu}}_m^{(x)} = \hat{\boldsymbol{B}}_{[k]}^{'}\hat{\boldsymbol{\mu}}_m \tag{134}$$

$$\hat{\boldsymbol{\Sigma}}_m^{(x)} = \operatorname{diag}\left(\hat{\boldsymbol{B}}_{[k]}^{'}\boldsymbol{W}_m\hat{\boldsymbol{B}}_{[k]}\right) \tag{135}$$

and the new observation noise parameters are simply

$$\hat{\boldsymbol{\mu}}^{(o)} = \hat{\boldsymbol{\mu}}_{[p-k]} \tag{136}$$

$$\hat{\boldsymbol{\Sigma}}^{(o)} = \hat{\boldsymbol{\Sigma}}_{[p-k]} \tag{137}$$

The benefit of the LDA and HLDA is that the likelihood is calculated only in the useful subspace and therefore, storage of the observation noise parameters is not needed.

6.1 Linear Discriminant Analysis Hidden Markov Models

Finding a better subspace in HMM framework using linear discriminant analysis and HLDA was studied in [11, 21, 29]. In [11], the close relationship between semi-tied full covariance matrices

(STC) and HLDA was presented. STC is based on class specific transformations which are applied to the state dependent output covariance matrices of a HMM. HMM with HLDA can be viewed as a STC model when the state-space and observation space have the same dimensionality, $k = p$, and there exists only one transform class.

A linear discriminant analysis HMM can be expressed with the following generative equations

$$\boxed{\begin{aligned} \boldsymbol{x}_t &\sim \mathcal{M}^{hmm} && (138) \\ \boldsymbol{o}_t &= \boldsymbol{C} \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{v} \end{bmatrix}, \qquad \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}) && (139) \end{aligned}}$$

where a single Gaussian observation noise is used. The generative model above can also be modified to represent other interesting models. If the observation noise is replaced by another state vector distributed according to a HMM, the equations define a two stream HMM with a linear mixing process. Depending on the form of the observation matrix, the model can represent an independent stream HMM or a factorial HMM [18]. As noted before, many interesting models based on linear discriminant analysis observation process are omitted in this paper.

The optimisation of the parameters of a HMM with HLDA observation process is the same as in the static case presented above. Only the observation indices must be replaced by time indices and the component posterior replaced by the joint likelihood of the current HMM state and the current mixture component given the observation sequence and the current model parameters. The posteriors can be obtained using the standard forward-backward recursions by replacing the posteriors of the observation given the HMM state accordingly.

6.2 Dynamical Linear Discriminant Analysis

Dynamical Linear Discriminant Analysis (DLDA) is based on continuous hidden k dimensional state variable vectors, $\boldsymbol{x}_t$, which evolve according to linear first-order Markov dynamics. A p dimensional observation vector, $\boldsymbol{o}_t$ is generated from the current state by a linear observation process. DLDA can be described by the following two equations

$$\boxed{\begin{aligned} \boldsymbol{x}_{t+1} &= \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{w}, \qquad \boldsymbol{w} \sim \mathcal{N}(\boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)}) && (140) \\ \boldsymbol{o}_t &= \boldsymbol{C} \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{v} \end{bmatrix}, \qquad \boldsymbol{v} \sim \mathcal{N}(\boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}) && (141) \end{aligned}}$$

where the relationship between the state vectors and the observation vectors is deterministic. Again, only a simple observation noise distribution is used. Defining $\boldsymbol{B} = (\boldsymbol{C}^{-1})'$ the likelihood of two consecutive observations can be represented as

$$p(\boldsymbol{o}_t|\boldsymbol{o}_{t-1}) = \text{abs}|\boldsymbol{B}'|\mathcal{N}(\boldsymbol{B}'_{[k]}\boldsymbol{o}_t; \boldsymbol{A}\boldsymbol{B}'_{[k]}\boldsymbol{o}_{t-1} + \boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)})\mathcal{N}(\boldsymbol{B}'_{[p-k]}\boldsymbol{o}_t; \boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)}) \tag{142}$$

where $\boldsymbol{B}_{[k]}$ denotes the first k columns of the matrix $\boldsymbol{B}$ and an absolute value of the determinant of the transformation matrix has to be taken to guarantee non-negative likelihoods. The transposition in the definition of $\boldsymbol{B}$ is needed to stick to our notation in the optimisation below. The likelihood of an observation sequence, $\boldsymbol{O} = \boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T$, is simply

$$p(\boldsymbol{O}) = \prod_{t=2}^{T} p(\boldsymbol{o}_t|\boldsymbol{o}_{t-1}) \tag{143}$$

and the auxiliary function to be maximised is therefore defined by

$$\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = \sum_{t=2}^{T} \log p(\boldsymbol{o}_t|\boldsymbol{o}_{t-1}) \tag{144}$$

Again, in the likelihood calculations the observation noise term can be ignored since it does not carry any class discriminating information.

6.2.1 Parameter Optimisation

If diagonal state transition and observation noise covariance matrices are used, the parameter optimisation can be done easily row by row. It is shown in Appendix G that the column vectors $1 \leq i \leq k$ of the matrix $\hat{\boldsymbol{B}} = (\hat{\boldsymbol{C}}^{-1})'$ are given by

$$\hat{\boldsymbol{b}}_i = \boldsymbol{k}_i \boldsymbol{G}_i^{-1}(\boldsymbol{k}_i + \delta \boldsymbol{c}_i) \tag{145}$$

where $\boldsymbol{c}_i$ is a vector of the cofactors of the matrix $\hat{\boldsymbol{B}}$ corresponding to the elements in the vector $\hat{\boldsymbol{b}}_i$, δ satisfies a usual quadratic expression in Eq. 289 and

$$\boldsymbol{G}_j = \frac{1}{\sigma_j^{(x)2}} \sum_{t=2}^{T} (\boldsymbol{o}_t \boldsymbol{o}_t' - a_{jj} \boldsymbol{o}_t \boldsymbol{o}_{t-1}' - a_{jj} \boldsymbol{o}_{t-1} \boldsymbol{o}_t + a_{jj}^2 \boldsymbol{o}_{t-1} \boldsymbol{o}_{t-1}') \tag{146}$$

$$\boldsymbol{k}_j = \frac{\mu_j^{(x)2}}{\sigma_j^{(x)2}} \sum_{t=2}^{T} (\boldsymbol{o}_t - a_{jj} \boldsymbol{o}_{t-1}) \tag{147}$$

where a_{jj} is the jth diagonal element of the state transition matrix $\boldsymbol{A}$. The rest $p-k$ columns of the matrix $\hat{\boldsymbol{B}}$ can be obtained by just replacing the statistics as follows

$$\boldsymbol{G}_j = \frac{1}{\sigma_{(j-k)}^{(o)2}} \sum_{t=2}^{T} \boldsymbol{o}_t \boldsymbol{o}_t' \tag{148}$$

$$\boldsymbol{k}_j = \frac{\mu_{(j-k)}^{(o)2}}{\sigma_{(j-k)}^{(o)2}} \sum_{t=2}^{T} \boldsymbol{o}_t \tag{149}$$

The state evolution parameters of DLDA can be obtained as follows

$$\begin{aligned}\hat{\boldsymbol{A}} = & \left(\operatorname{diag}\left(\hat{\boldsymbol{B}}_{[k]}'\Big(\sum_{t=2}^{T} \boldsymbol{o}_t \boldsymbol{o}_{t-1}'\Big)\hat{\boldsymbol{B}}_{[k]} - \frac{1}{T-1}\hat{\boldsymbol{B}}_{[k]}'\Big(\sum_{t=2}^{T} \boldsymbol{o}_t\Big)\Big(\sum_{t=2}^{T} \boldsymbol{o}_{t-1}\Big)'\hat{\boldsymbol{B}}_{[k]}\right)\right) \\ & \left(\operatorname{diag}\left(\hat{\boldsymbol{B}}_{[k]}'\Big(\sum_{t=2}^{T} \boldsymbol{o}_{t-1} \boldsymbol{o}_{t-1}'\Big)\hat{\boldsymbol{B}}_{[k]} - \frac{1}{T-1}\hat{\boldsymbol{B}}_{[k]}'\Big(\sum_{t=2}^{T} \boldsymbol{o}_{t-1}\Big)\Big(\sum_{t=2}^{T} \boldsymbol{o}_{t-1}\Big)'\hat{\boldsymbol{B}}_{[k]}\right)\right)^{-1}\end{aligned} \tag{150}$$

$$\hat{\boldsymbol{\mu}}^{(x)} = \frac{1}{T-1} \sum_{t=2}^{T} \left(\hat{\boldsymbol{B}}_{[k]}' \boldsymbol{o}_t - \hat{\boldsymbol{A}} \hat{\boldsymbol{B}}_{[k]}' \boldsymbol{o}_{t-1}\right) \tag{151}$$

$$\hat{\boldsymbol{\Sigma}}^{(x)} = \frac{1}{T-1} \sum_{t=2}^{T} \operatorname{diag}\left(\hat{\boldsymbol{B}}_{[k]}' \boldsymbol{o}_t \boldsymbol{o}_t' \hat{\boldsymbol{B}}_{[k]} - \left[\, \hat{\boldsymbol{A}} \;\; \hat{\boldsymbol{\mu}}^{(x)} \,\right] \left[\, \hat{\boldsymbol{B}}_{[k]}' \boldsymbol{o}_t \boldsymbol{o}_{t-1}' \hat{\boldsymbol{B}}_{[k]} \;\; \hat{\boldsymbol{B}}_{[k]}' \boldsymbol{o}_t \,\right]'\right) \tag{152}$$

and finally the observation noise parameters can be re-estimated as follows

$$\hat{\boldsymbol{\mu}}^{(o)} = \frac{1}{T-1} \hat{\boldsymbol{B}}_{[p-k]}' \sum_{t=2}^{T} \boldsymbol{o}_t \tag{153}$$

$$\hat{\boldsymbol{\Sigma}}^{(o)} = \frac{1}{T-1} \sum_{t=2}^{T} \operatorname{diag}\left(\hat{\boldsymbol{B}}_{[p-k]}' \boldsymbol{o}_t \boldsymbol{o}_t' \hat{\boldsymbol{B}}_{[p-k]} - \hat{\boldsymbol{\mu}}^{(o)} \hat{\boldsymbol{\mu}}^{(o)\prime}\right) \tag{154}$$

7 Implementation Issues

In this section, some implementation issues are discussed. Firstly, there exists a nice result from matrix algebra that can be exploited in the inversion of the large matrices in case of models based on factor analysis observation process. Secondly, some issues in variance flooring and parameter tying are presented.

7.1 Efficient E Step Calculation

A p by p matrix of the form $\boldsymbol{R}+\boldsymbol{STU}$ has to be inverted in the E step of factor analysis based models reviewed in this paper; namely, $\boldsymbol{K}=\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{C}'+\boldsymbol{\Sigma}^{(o)})^{-1}$ in case of factor analysis, $\boldsymbol{K}_m=\boldsymbol{\Sigma}^{(x)}\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{\Sigma}^{(x)}\boldsymbol{C}'+\boldsymbol{\Sigma}^{(o)})^{-1}$ in case of mixtures of factors, $\boldsymbol{K}_t=\boldsymbol{\Sigma}_t^{t-1}\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{\Sigma}_t^{t-1}\boldsymbol{C}'+\boldsymbol{\Sigma}^{(o)})^{-1}$ in case of linear dynamical systems and $\boldsymbol{K}_{jmn}=\boldsymbol{\Sigma}_{jn}^{(x)}\boldsymbol{C}_j'(\boldsymbol{C}_j\boldsymbol{\Sigma}_{jn}^{(x)}\boldsymbol{C}_j'+\boldsymbol{\Sigma}_{jm}^{(o)})^{-1}$ in case of factor analysed HMMs. Since mostly diagonal covariance matrices are used, it is possible to use the following matrix inversion lemma to implement the inversion efficiently [23]

$$(\boldsymbol{R}+\boldsymbol{STU})^{-1} = \boldsymbol{R}^{-1}-\boldsymbol{R}^{-1}\boldsymbol{S}(\boldsymbol{T}^{-1}+\boldsymbol{U}\boldsymbol{R}^{-1}\boldsymbol{S})^{-1}\boldsymbol{U}\boldsymbol{R}^{-1} \tag{155}$$

where $\boldsymbol{R}$ is replaced by a diagonal output noise covariance matrix in all the previous cases and is therefore easy to invert, and the k by k matrix $(\boldsymbol{T}^{-1}+\boldsymbol{U}\boldsymbol{R}^{-1}\boldsymbol{S})$ can be inverted more efficiently since $k<p$ in case of subspace modelling.

Similar result for the determinants of the matrices of the form $\boldsymbol{R}+\boldsymbol{STU}$ can be employed in the likelihood calculations. Since the matrices corresponding to $\boldsymbol{R}$ and $\boldsymbol{T}$ are often diagonal apart from LDSs where $\boldsymbol{\Sigma}_t^{t-1}$ is unfortunately not diagonal but is still in the subspace, the determinant

$$|\boldsymbol{R}+\boldsymbol{STU}| = |\boldsymbol{R}||\boldsymbol{T}||\boldsymbol{T}^{-1}+\boldsymbol{U}\boldsymbol{R}^{-1}\boldsymbol{S}| \tag{156}$$

can be obtained efficiently. The inverse $(\boldsymbol{T}^{-1}+\boldsymbol{U}\boldsymbol{R}^{-1}\boldsymbol{S})^{-1}$ in the subspace can be computed using Cholesky decomposition since the matrices are symmetric positive definite and the determinant $|\boldsymbol{T}^{-1}+\boldsymbol{U}\boldsymbol{R}^{-1}\boldsymbol{S}|$ can be obtained as a by-product.

7.2 Variance Flooring for FAHMMs

The number of Gaussian components in a large system based on FAHMMs is often huge. To guarantee that the covariance matrices are invertible after re-estimation, flooring of the variance components has to be considered. In a standard diagonal covariance HMM the global variance of the training data is first computed. If the ith component of a re-estimated variance, $\sigma_{jmi}^{(o)2}$, falls below a certain fraction of the corresponding global variance component, σ_{fi}^2, it is floored to σ_{fi}^2 [42]. The floor is often chosen to be one hundredth of the global variance.

In case of factor analysed HMMs, there are two mixture distributions, one in state level and the other in observation level. In the experiments carried out so far, flooring was found to be an issue only with very few observation noise variance elements. The full covariance matrices $\boldsymbol{C}_j\boldsymbol{\Sigma}_{jn}^{(x)}\boldsymbol{C}_j'+\boldsymbol{\Sigma}_{jm}^{(o)}$ have to be invertible for every combination of observation and state-space noise components. Since the matrices $\boldsymbol{C}_j\boldsymbol{\Sigma}_{jn}^{(x)}\boldsymbol{C}_j'$ are generally singular, even one zero variance element in a single observation noise component can make the full covariance matrix singular. This was found to happen very seldom with a context dependent large vocabulary speech recognition system when there was not enough training examples assigned to a particular state. The standard variance flooring scheme of HMM systems on the observation noise variances was found to be working without any dramatic influence on the training and recognition likelihood scores.

7.3 Parameter Tying

As the number of model parameters is increased, more data is required to obtain reliable estimates in the re-estimation process. As discussed earlier, tying of the model parameters becomes important as the complexity of the models is increased. The parameter tying has only been seen in this paper in the re-estimation of the mixture of linear dynamical systems and the factor analysers with mixtures of observation processes. It is though possible to arbitrarily share any parameters in different levels of every model presented in this paper.

In general, the parameter tying requires simply accumulation of the sufficient statistics in the E step of the EM algorithm and the M step equations remain as usual. For example, if the observation matrices of single FAHMM over all the states are to be tied, the statistics $\boldsymbol{G}_l$ and $\boldsymbol{k}_l$ have to be summed over all the states $j \in (1, N_s)$ before updating the parameters in the usual way. Also, parameters over several models can be tied similarly. For example, entire state parameters within several models can be tied as is often done in the tied state hidden Markov models in continuous speech recognition.

8 Conclusion

Several interesting linear Gaussian models have been presented in this paper. The models fall into category of general state-space models with assumptions on linearity and distributions which are either Gaussians or mixtures of Gaussians. The state vectors can be regarded as generated by a single distribution in case of static models or either linear first-order Gauss-Markov random process or hidden Markov (HMM) model in case of dynamic models.

This paper complements the attempt to unify the set of linear Gaussian models presented in [38] although both these papers have still not covered all the possible forms. Some variants of linear discriminant analysis (LDA) [11] have been omitted as well as all the possible combinations of LDA with different mixture assumptions. Instead, factor analysis models and factor analysed hidden Markov models were presented in very generic fashion combining the way of representing mixtures of factors such as used in independent factor analysis [1], and mixtures of factor analysis observation processes with different parameter sharing schemes such as used in mixtures of factor analysers [15] and shared factor analysis [22].

Factor analysed hidden Markov model (FAHMM) provides a very generic extension to currently very popular time-series model, the hidden Markov model. FAHMM is based on state vectors generated by a HMM and an observation process that performs shared factor analysis using the state vector elements as factors. FAHMMs provide a flexible framework for time-series modelling since all the advantages of HMMs are present such as efficient segmentation using Viterbi algorithm but, in addition, a variety of possible mixturing and tying schemes of the model parameters can be employed. An important benefit of FAHMMs in comparison to the standard diagonal covariance HMMs is the modelling of spatial correlation which is often present due to non-ideal feature extraction algorithms.

Several models suitable in segment model (SM) [34] framework have also been presented. It has been noted that the major drawback in the segment models using linear dynamical systems (LDSs) as the models for feature vector dynamics is the unimodal nature of the observations. It is well known that many feature vectors such as in speech recognition have multi-modal distributions due to the source and environment variability. In this paper, a form of mixture of LDSs where the parameters of several independent LDSs other than the observation noise parameters are tied so representing crudely a LDS with a mixture of Gaussians as the observation noise distribution. Other forms of LDSs with mixture model noise sources do not have tractable training schemes and some sort of approximate methods have to be considered such as particle filtering [8]. There are still many interesting alternatives to extend the class of models for feature dynamics in SM framework. It will be a worthwhile to investigate these models since all the promises of SMs have not yet been utilised.

Acknowledgements

A-V.I. Rosti is funded by the Cambridge European Trust, EPSRC, Finnish Cultural Foundation, Nokia Foundation, Research Scholarship Foundation of Tampere, Tampere Chamber of Commerce

and Industry, and Tampere Graduate School in Information Science and Engineering.

References

[1] H. Attias. Independent factor analysis. *Neural Computation*, 11(4):803–851, 1999.

[2] L.E. Baum and T. Petrie. Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics*, 37:1554–1563, 1966.

[3] L.E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[4] C.S. Blackburn. *Articulatory Methods for Speech Production and Recognition*. PhD thesis, University of Cambridge, 1996.

[5] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[6] V. Digalakis. *Segment-Based Stochastic Models of Spectral Dynamics for Continuous Speech Recognition*. PhD thesis, Boston University, 1992.

[7] V. Digalakis, J.R. Rohlicek, and M. Ostendorf. ML estimation of a stochastic linear system with the EM algorithm and its application to speech recognition. *IEEE Transactions on Speech and Audio Processing*, 1(4):431–442, 1993.

[8] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[9] A. Doucet, S.J. Godsill, and M. West. Monte carlo filtering and smoothing with application to time-varying spectral estimation. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 701–704, 2000.

[10] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, 2 edition, 1990.

[11] M.J.F. Gales. Maximum likelihood multiple projection schemes for hidden Markov models. Technical Report CUED/F-INFENG/TR.365, Cambridge University Engineering Department, 1999. Available via anonymous ftp from `ftp://svr-ftp.eng.cam.ac.uk/pub/reports/`.

[12] M.J.F. Gales. Semi-tied covariance matrices for hidden Markov models. *IEEE Transactions on Speech and Audio Processing*, 7(3):272–281, 1999.

[13] M.J.F. Gales, K.M. Knill, and S.J. Young. State-based Gaussian selection in large vocabulary continuous speech recognition using HMMs. *IEEE Transactions on Speech and Audio Processing*, 7(2):152–161, 1999.

[14] Z. Ghahramani. Learning dynamic Bayesian networks. In C.L. Giles and M. Gori, editors, *Adaptive Processing of Sequences and Data Structures*, volume 1387 of *Lecture Notes in Computer Science*, pages 168–197. Springer, 1998.

[15] Z. Ghahramani and G. Hinton. The EM algorithm for mixtures of factor analyzers. Technical Report CRG-TR-96-1, Department of Computer Science, University of Toronto, 1996.

[16] Z. Ghahramani and G. Hinton. Parameter estimation for linear dynamical systems. Technical Report CRG-TR-96-2, Department of Computer Science, University of Toronto, 1996.

[17] Z. Ghahramani and G.E. Hinton. Variational learning for switching state-space models. *Neural Computation*, 12(4):963–996, 1998.

[18] Z. Ghahramani and M.I. Jordan. Factorial hidden Markov models. *Machine Learning*, 29:245–273, 1997.

[19] S. Godsill, A. Doucet, and M. West. Maximum a posteriori sequence estimation using Monte Carlo particle filters. *Annals of the Institute of Statistical Mathematics*, 52, 2001.

[20] R.A. Gopinath. Constrained maximum likelihood modeling with gaussian distributions. In *Proceedings of the Broadcast News Transcription and Understanding Workshop*, 1998.

[21] R.A. Gopinath. Maximum likelihood modeling with Gaussian distributions for classification. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 661–664, 1998.

[22] R.A. Gopinath, B. Ramabhadran, and S. Dharanipragada. Factor analysis invariant to linear transformations of data. In *Proceedings of the International Conference on Speech and Language Processing*, pages 397–400, 1998.

[23] D.A. Harville. *Matrix Algebra from a Statistician's Perspective.* Springer, 1997.

[24] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proceedings of the European Conference on Computer Vision*, volume 1, pages 343–356, 1996.

[25] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[26] R.A. Johnson and D.W. Wichern. *Applied Multivariate Statistical Analysis.* Prentice Hall, 4 edition, 1998.

[27] R.E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the American Society of Mechanical Engineering, Series D, Journal of Basic Engineering*, 82:35–45, 1960.

[28] R.E. Kalman and R.S. Bucy. New results in linear filtering and prediction theory. *Transactions of the American Society of Mechanical Engineering, Series D, Journal of Basic Engineering*, 83:95–108, 1961.

[29] N. Kumar. *Investigation of Silicon-Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition.* PhD thesis, Johns Hopkins University, 1997.

[30] C.J. Leggetter and P.C. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density HMMs. *Computer Speech and Language*, 9(2):171–185, 1995.

[31] L.A. Liporace. Maximum likelihood estimation for multivariate observations of Markov sources. *IEEE Transactions on Information Theory*, IT-28(5):729–734, 1982.

[32] T.P. Minka. From hidden Markov models to linear dynamical systems. Technical Report #531, Vision and Modeling Group, MIT Media Laboratory, 1999. Available at `http://vismod.www.media.mit.edu/~tpminka/papers/learning.html`.

[33] K. Murphy. Learning switching kalman filter models. Technical Report 98-10, Compaq Cambridge Research Lab., 1998. Available at `http://www.cs.berkeley.edu/~murphyk/publ.html`.

[34] M. Ostendorf, V. Digalakis, and O. Kimball. From HMM's to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4(5):360–378, 1996.

[35] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, 1989.

[36] H.E. Rauch. Solutions to the linear smoothing problem. *IEEE Transactions on Automatic Control*, 8:371–372, 1963.

[37] H.E. Rauch, F. Tung, and C.T. Striebel. Maximum likelihood estimates of linear dynamic systems. *American Institute of Aeronautics and Astronautics Journal*, 3(8):1445–1450, 1965.

[38] S. Roweis and Z. Ghahramani. A unifying review of linear Gaussian models. *Neural Computation*, 11(2):305–345, 1999.

[39] D.B. Rubin and D.T. Thayer. EM algorithms for ML factor analysis. *Psychometrika*, 47(1):69–76, 1982.

[40] L. Saul and M. Rahim. Maximum likelihood and minimum classification error factor analysis for automatic speech recognition. *IEEE Transactions on Speech and Audio Processing*, 8(2):115–125, 1999.

[41] A.J. Viterbi. Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory*, IT-13:260–269, 1967.

[42] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book (for HTK Version 3.0).* Cambridge University, 2000.

[43] R.S. Zemel. *A Minimum Description Length Framework for Unsupervised Learning.* PhD thesis, University of Toronto, 1994.

A Useful Results from Matrix Algebra

This appendix reviews some useful results from the matrix algebra for the derivations in the following appendices. First, inversion formulae for partitioned matrices are presented and then conditional density of two multivariate Gaussians is derived.

A.1 Inverting Partitioned Matrices

Let $\boldsymbol{A}$ represent an arbitrary $(n+m)$ by $(n+m)$ matrix with the following partitioning

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 \\ \boldsymbol{A}_3 & \boldsymbol{A}_4 \end{bmatrix} \tag{157}$$

where $\boldsymbol{A}_1$ is an n by n matrix, $\boldsymbol{A}_2$ an n by m matrix, $\boldsymbol{A}_3$ an m by n matrix and $\boldsymbol{A}_4$ an m by m matrix. Suppose that $\boldsymbol{A}_1$ is non-singular. The matrix $\boldsymbol{A}$ is singular if and only if the n by n matrix

$$(\boldsymbol{A}|\boldsymbol{A}_4) = \boldsymbol{A}_1 - \boldsymbol{A}_2\boldsymbol{A}_4^{-1}\boldsymbol{A}_3 \tag{158}$$

is non-singular [23]. The matrix $(\boldsymbol{A}|\boldsymbol{A}_4)$ is called the Schur complement of $\boldsymbol{A}_4$ in $\boldsymbol{A}$ and the following two identities apply

$$\begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 \\ \boldsymbol{A}_3 & \boldsymbol{A}_4 \end{bmatrix}^{-1} = \begin{bmatrix} (\boldsymbol{A}|\boldsymbol{A}_4)^{-1} & -(\boldsymbol{A}|\boldsymbol{A}_4)^{-1}\boldsymbol{A}_2\boldsymbol{A}_4^{-1} \\ -\boldsymbol{A}_4^{-1}\boldsymbol{A}_3(\boldsymbol{A}|\boldsymbol{A}_4)^{-1} & \boldsymbol{A}_4^{-1} + \boldsymbol{A}_4^{-1}\boldsymbol{A}_3(\boldsymbol{A}|\boldsymbol{A}_4)^{-1}\boldsymbol{A}_2\boldsymbol{A}_4^{-1} \end{bmatrix} \tag{159}$$

$$= \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{A}_4^{-1} \end{bmatrix} + \begin{bmatrix} \boldsymbol{I}_n \\ -\boldsymbol{A}_4^{-1}\boldsymbol{A}_3 \end{bmatrix} (\boldsymbol{A}|\boldsymbol{A}_4)^{-1} \begin{bmatrix} \boldsymbol{I}_n \\ -\boldsymbol{A}_2\boldsymbol{A}_4^{-1} \end{bmatrix}' \tag{160}$$

A.2 Conditioning Multivariate Gaussians

Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be p and k dimensional Gaussian distributed random vectors with mean vectors, $\boldsymbol{\mu_x}$ and $\boldsymbol{\mu_y}$, and covariance matrices, $\boldsymbol{\Sigma_x}$ and $\boldsymbol{\Sigma_y}$, respectively. Suppose also that $\boldsymbol{x}$ and $\boldsymbol{y}$ are also jointly Gaussian with $\boldsymbol{\Sigma_{yx}}$ and $\boldsymbol{\Sigma_{xy}}$ as their cross covariance matrices. The joint distribution can be represented as follows

$$p(\boldsymbol{x},\boldsymbol{y}) = \mathcal{N}(\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix}; \begin{bmatrix} \boldsymbol{\mu_x} \\ \boldsymbol{\mu_y} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma_x} & \boldsymbol{\Sigma_{xy}} \\ \boldsymbol{\Sigma_{yx}} & \boldsymbol{\Sigma_y} \end{bmatrix}) \tag{161}$$

It should be noted that $\boldsymbol{\Sigma_{yx}} = \boldsymbol{\Sigma}'_{\boldsymbol{xy}}$.

The posterior distribution of $\boldsymbol{x}$ given $\boldsymbol{y}$ is obtained by the definition, $p(\boldsymbol{x}|\boldsymbol{y}) = p(\boldsymbol{x},\boldsymbol{y})/p(\boldsymbol{y})$,

$$p(\boldsymbol{x}|\boldsymbol{y}) = (2\pi)^{-\frac{(p+k)}{2}} \left| \begin{bmatrix} \boldsymbol{\Sigma_x} & \boldsymbol{\Sigma_{xy}} \\ \boldsymbol{\Sigma_{yx}} & \boldsymbol{\Sigma_y} \end{bmatrix} \right|^{-\frac{1}{2}} \exp\Big\{ -\frac{1}{2} \begin{bmatrix} \boldsymbol{x}-\boldsymbol{\mu_x} \\ \boldsymbol{y}-\boldsymbol{\mu_y} \end{bmatrix}' \begin{bmatrix} \boldsymbol{\Sigma_x} & \boldsymbol{\Sigma_{xy}} \\ \boldsymbol{\Sigma_{yx}} & \boldsymbol{\Sigma_y} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{x}-\boldsymbol{\mu_x} \\ \boldsymbol{y}-\boldsymbol{\mu_y} \end{bmatrix} \Big\}$$

$$\Big/ \Big((2\pi)^{-\frac{k}{2}} |\boldsymbol{\Sigma_y}|^{-\frac{1}{2}} \exp\Big\{ -\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu_y})'\boldsymbol{\Sigma_y}^{-1}(\boldsymbol{y}-\boldsymbol{\mu_y}) \Big\} \Big) \tag{162}$$

$$= (2\pi)^{-\frac{p}{2}} |\boldsymbol{\Sigma_{xx}} - \boldsymbol{\Sigma_{xy}}\boldsymbol{\Sigma_y}^{-1}\boldsymbol{\Sigma_{yx}}|^{-\frac{1}{2}} \exp\Big\{ -\frac{1}{2}\gamma \Big\} \tag{163}$$

where the last determinant is obtained using the following identity for partitioned matrices [23]

$$\left| \begin{bmatrix} \boldsymbol{\Sigma_x} & \boldsymbol{\Sigma_{xy}} \\ \boldsymbol{\Sigma_{yx}} & \boldsymbol{\Sigma_y} \end{bmatrix} \right| = |\boldsymbol{\Sigma_y}||\boldsymbol{\Sigma_{xx}} - \boldsymbol{\Sigma_{xy}}\boldsymbol{\Sigma_y}^{-1}\boldsymbol{\Sigma_{yx}}| \tag{164}$$

and the definition of the scalar γ is

$$\gamma = \begin{bmatrix} \boldsymbol{x}-\boldsymbol{\mu_x} \\ \boldsymbol{y}-\boldsymbol{\mu_y} \end{bmatrix}' \begin{bmatrix} \boldsymbol{\Sigma_x} & \boldsymbol{\Sigma_{xy}} \\ \boldsymbol{\Sigma_{yx}} & \boldsymbol{\Sigma_y} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{x}-\boldsymbol{\mu_x} \\ \boldsymbol{y}-\boldsymbol{\mu_y} \end{bmatrix} - (\boldsymbol{y}-\boldsymbol{\mu_y})'\boldsymbol{\Sigma_y}^{-1}(\boldsymbol{y}-\boldsymbol{\mu_y}) \tag{165}$$

The partitioned joint covariance matrix, denoted by $\boldsymbol{\Sigma}$ below, can be inverted by using the identity in Eq. 160. Obviously, the first term in the right hand side of the identity cancels the last term in Eq. 165 and therefore it simplifies to the following form

$$\gamma = \Big((\boldsymbol{x}-\boldsymbol{\mu}_{\boldsymbol{x}})' - (\boldsymbol{y}-\boldsymbol{\mu}_{\boldsymbol{y}})'\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1}\boldsymbol{\Sigma}_{\boldsymbol{yx}}\Big)(\boldsymbol{\Sigma}|\boldsymbol{\Sigma}_{\boldsymbol{y}})^{-1}\Big((\boldsymbol{x}-\boldsymbol{\mu}_{\boldsymbol{x}}) - \boldsymbol{\Sigma}_{\boldsymbol{xy}}\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}_{\boldsymbol{y}})\Big) \tag{166}$$

which is the Mahalanobis distance between the vectors $\boldsymbol{x}$ and $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}} = \boldsymbol{\mu}_{\boldsymbol{x}} + \boldsymbol{\Sigma}_{\boldsymbol{xy}}\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}_{\boldsymbol{y}})$ with covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}} = (\boldsymbol{\Sigma}|\boldsymbol{\Sigma}_{\boldsymbol{y}}) = \boldsymbol{\Sigma}_{\boldsymbol{x}} - \boldsymbol{\Sigma}_{\boldsymbol{xy}}\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1}\boldsymbol{\Sigma}_{\boldsymbol{yx}}$.

Indeed, by substituting γ back into Eq. 163 it is obvious that the posterior distribution of $\boldsymbol{x}$ given $\boldsymbol{y}$ is Gaussian distributed with mean vector $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$; i.e.,

$$p(\boldsymbol{x}|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{x};\boldsymbol{\mu}_{\boldsymbol{x}} + \boldsymbol{\Sigma}_{\boldsymbol{xy}}\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}_{\boldsymbol{y}}), \boldsymbol{\Sigma}_{\boldsymbol{x}} - \boldsymbol{\Sigma}_{\boldsymbol{xy}}\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1}\boldsymbol{\Sigma}_{\boldsymbol{yx}}) \tag{167}$$

B Derivation of the EM Algorithm for Factor Analysis

The expectation maximisation algorithm for factor analysis is presented in this appendix. The derivation of this standard algorithm was first done in [39]. Here the derivation is presented to keep the paper self contained and using generic matrix manipulation. In the E step, the sufficient statistics required in the M step are obtained. The M step is very generic in that only few modifications are needed in the EM algorithms for the other models with factor analysis observation process.

B.1 E Step for Factor Analysis

The state posteriors have to be obtained in the E step of the EM algorithm. It is easy to see from the factor analysis observation equation in Eq. 11 that the likelihood of the observation vector,$\boldsymbol{o}_j$, can be obtained as follows

$$p(\boldsymbol{o}_j) = \mathcal{N}(\boldsymbol{o}_j;\boldsymbol{\mu}^{(o)}, \boldsymbol{C}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)}) \tag{168}$$

and since the observation and the state vector are also jointly Gaussian with the following joint density

$$p(\boldsymbol{o}_j,\boldsymbol{x}) = \mathcal{N}(\begin{bmatrix}\boldsymbol{o}_j\\ \boldsymbol{x}\end{bmatrix};\begin{bmatrix}\boldsymbol{\mu}^{(o)}\\ \boldsymbol{0}\end{bmatrix},\begin{bmatrix}\boldsymbol{C}\boldsymbol{C}'+\boldsymbol{\Sigma}^{(o)} & \boldsymbol{C}\\ \boldsymbol{C}' & \boldsymbol{I}\end{bmatrix}) \tag{169}$$

and the state posteriors can be obtained using Eq. 167

$$p(\boldsymbol{x}|\boldsymbol{o}_j) = \mathcal{N}\big(\boldsymbol{x};\boldsymbol{K}(\boldsymbol{o}_j - \boldsymbol{\mu}^{(o)}), \boldsymbol{I} - \boldsymbol{K}\boldsymbol{C}\big) \tag{170}$$

where $\boldsymbol{K} = \boldsymbol{C}'(\boldsymbol{C}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)})^{-1}$.

The required first and second-order sufficient statistics can be now represented as follows

$$\hat{\boldsymbol{x}}(j) = E\{\boldsymbol{x}|\boldsymbol{o}_j,\mathcal{M}\} = \boldsymbol{K}(\boldsymbol{o}_j - \boldsymbol{\mu}^{(o)}) \tag{171}$$

$$\hat{\boldsymbol{R}}(j) = E\{\boldsymbol{x}\boldsymbol{x}'|\boldsymbol{o}_j,\mathcal{M}\} = \boldsymbol{I} - \boldsymbol{K}\boldsymbol{C} + \hat{\boldsymbol{x}}(j)\hat{\boldsymbol{x}}'(j) \tag{172}$$

where $\mathcal{M}$ is added to show explicitly that the old model parameters are used.

B.2 M Step for Factor Analysis

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{C}}, \hat{\boldsymbol{\mu}}^{(o)}, \hat{\boldsymbol{\Sigma}}^{(o)})$, for the observation process the following auxiliary function has to be maximised

$$\begin{aligned}&\mathcal{Q}_o(\mathcal{M},\hat{\mathcal{M}}) = \\ &-\frac{1}{2}\sum_{j=1}^{N}\Big(\log|\hat{\boldsymbol{\Sigma}}^{(o)}| + E\Big\{(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\boldsymbol{x} - \hat{\boldsymbol{\mu}}^{(o)})'\boldsymbol{\Sigma}^{(o)-1}(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\boldsymbol{x} - \hat{\boldsymbol{\mu}}^{(o)})\Big|\boldsymbol{O},\mathcal{M}\Big\}\Big)\end{aligned} \tag{173}$$

Differentiating Eq. 173 with respect to $\hat{\boldsymbol{\mu}}^{(o)}$ and equating the resulting expression to zero

$$\frac{\partial \mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\mu}}^{(o)}} = \hat{\boldsymbol{\Sigma}}^{(o)-1} \sum_{j=1}^{N} \big(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\hat{\boldsymbol{x}}(j) - \hat{\boldsymbol{\mu}}^{(o)}\big) = \mathbf{0} \tag{174}$$

$$\hat{\boldsymbol{\mu}}^{(o)} = \frac{1}{N} \sum_{j=1}^{N} \big(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\hat{\boldsymbol{x}}(j)\big) \tag{175}$$

Differentiating Eq. 173 with respect to $\hat{\boldsymbol{C}}$, substituting the above ML estimate of the observation noise mean and setting the resulting equation to zero, the new observation matrix can be solved using only the sufficient statistics from the E step as follows

$$\frac{\partial \mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{C}}} = \hat{\boldsymbol{\Sigma}}^{(o)-1} \sum_{j=1}^{N} \big(\boldsymbol{o}_j\hat{\boldsymbol{x}}'(j) - \hat{\boldsymbol{\mu}}^{(o)}\hat{\boldsymbol{x}}'(j) - \hat{\boldsymbol{C}}\hat{\boldsymbol{R}}(j)\big) = \mathbf{0} \tag{176}$$

$$\hat{\boldsymbol{C}} = \Big(\sum_{j=1}^{N} \boldsymbol{o}_j\hat{\boldsymbol{x}}'(j) - \frac{1}{N}\sum_{j=1}^{N}\boldsymbol{o}_t \sum_{j=1}^{N}\hat{\boldsymbol{x}}'(j)\Big)\Big(\sum_{j=1}^{N}\hat{\boldsymbol{R}}(j) - \frac{1}{N}\sum_{j=1}^{N}\hat{\boldsymbol{x}}(j)\sum_{j=1}^{N}\hat{\boldsymbol{x}}'(j)\Big)^{-1} \tag{177}$$

When re-estimating the model parameters, the observation matrix has to be updated before the observation noise mean vector or they can be re-estimated simultaneously using matrix notation. By redefining the following sums of the sufficient statistics

$$\boldsymbol{\Gamma}_1 = \sum_{j=1}^{N} \boldsymbol{o}_j\hat{\boldsymbol{x}}'(j) \quad , \boldsymbol{\Gamma}_2 = \sum_{j=1}^{N}\hat{\boldsymbol{R}}(j)$$

$$\boldsymbol{\zeta}_1 = \sum_{j=1}^{N}\boldsymbol{o}_j \quad , \boldsymbol{\zeta}_2 = \sum_{j=1}^{N}\hat{\boldsymbol{x}}(j) \tag{178}$$

it can be checked that the parameters $\hat{\boldsymbol{C}}$ and $\hat{\boldsymbol{\mu}}^{(o)}$ can be maximised simultaneously as follows

$$\begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix} = \Big(\sum_{j=1}^{N} \begin{bmatrix} \boldsymbol{o}_j\hat{\boldsymbol{x}}'(j) & \boldsymbol{o}_j \end{bmatrix}\Big)\Big(\sum_{j=1}^{N}\begin{bmatrix} \hat{\boldsymbol{R}}(j) & \hat{\boldsymbol{x}}(j) \\ \hat{\boldsymbol{x}}'(j) & 1 \end{bmatrix}\Big)^{-1} = \begin{bmatrix} \boldsymbol{\Gamma}_1 & \boldsymbol{\zeta}_1 \end{bmatrix}\begin{bmatrix} \boldsymbol{\Gamma}_2 & \boldsymbol{\zeta}_2 \\ \boldsymbol{\zeta}_2' & N \end{bmatrix}^{-1} \tag{179}$$

Now, the Schur's complement, $(\boldsymbol{\Gamma}|N)$, in Eq. 160 for the matrix to be inverted is

$$(\boldsymbol{\Gamma}|N) = \boldsymbol{\Gamma}_2 - \frac{1}{N}\boldsymbol{\zeta}_2\boldsymbol{\zeta}_2' \tag{180}$$

and the first element of the matrix product in Eq. 179 is

$$\boldsymbol{\Gamma}_1(\boldsymbol{\Gamma}|N)^{-1} - \frac{1}{N}\boldsymbol{\zeta}_1\boldsymbol{\zeta}_2'(\boldsymbol{\Gamma}|N)^{-1} = \Big(\boldsymbol{\Gamma}_1 - \frac{1}{N}\boldsymbol{\zeta}_1\boldsymbol{\zeta}_2'\Big)\Big(\boldsymbol{\Gamma}_2 - \frac{1}{N}\boldsymbol{\zeta}_2\boldsymbol{\zeta}_2'\Big)^{-1} \tag{181}$$

which is exactly the ML estimate for $\hat{\boldsymbol{C}}$. The second element of the matrix product in Eq. 179 is

$$\begin{aligned} &-\frac{1}{N}\boldsymbol{\Gamma}_1(\boldsymbol{\Gamma}|N)^{-1}\boldsymbol{\zeta}_2 + \frac{1}{N}\boldsymbol{\zeta}_1 + \frac{1}{N^2}\boldsymbol{\zeta}_1\boldsymbol{\zeta}_2'(\boldsymbol{\Gamma}|N)^{-1}\boldsymbol{\zeta}_2 \\ &= \frac{1}{N}\Big(\boldsymbol{\zeta}_1 - \big(\boldsymbol{\Gamma}_1(\boldsymbol{\Gamma}|N)^{-1} - \frac{1}{N}\boldsymbol{\zeta}_1\boldsymbol{\zeta}_2'(\boldsymbol{\Gamma}|N)^{-1}\big)\boldsymbol{\zeta}_2\Big) \end{aligned} \tag{182}$$

which is exactly the ML estimate for $\hat{\boldsymbol{\mu}}^{(o)}$.

To find the new observation noise covariance matrix, it is easier to differentiate Eq. 173 with respect to the inverse of $\hat{\boldsymbol{\Sigma}}^{(o)}$ as follows

$$\frac{\partial \mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\Sigma}}^{(o)-1}} = \frac{1}{2}\sum_{j=1}^{N}\Big(\hat{\boldsymbol{\Sigma}}^{(o)} - E\Big\{\big(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\boldsymbol{x} - \hat{\boldsymbol{\mu}}^{(o)}\big)\big(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\boldsymbol{x} - \hat{\boldsymbol{\mu}}^{(o)}\big)'\Big|\boldsymbol{O}, \mathcal{M}\Big\}\Big) = \mathbf{0} \tag{183}$$

$$\hat{\boldsymbol{\Sigma}}^{(o)} = \frac{1}{N}\sum_{j=1}^{N}\mathrm{diag}\Big(\boldsymbol{o}_j\boldsymbol{o}_j' - \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix}\begin{bmatrix} \boldsymbol{o}_j\hat{\boldsymbol{x}}'(j) & \boldsymbol{o}_j \end{bmatrix}'\Big) \tag{184}$$

where the ML estimates of the observation matrix and noise mean vector must be used. The last form is obtained by noting that the terms inside the expectation in Eq. 183 can be reorganised as follows

$$\boldsymbol{o}_j\boldsymbol{o}_j' - \tag{185}$$
$$\begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix} \begin{bmatrix} \boldsymbol{o}_j\hat{\boldsymbol{x}}'(j) & \boldsymbol{o}_j \end{bmatrix}' - \begin{bmatrix} \boldsymbol{o}_j\hat{\boldsymbol{x}}'(j) & \boldsymbol{o}_j \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix}' + \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{R}}(j) & \hat{\boldsymbol{x}}(j) \\ \hat{\boldsymbol{x}}'(j) & 1 \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix}'$$

and substituting the first term of Eq. 179 in place of $\begin{bmatrix} \hat{\boldsymbol{C}} & \hat{\boldsymbol{\mu}}^{(o)} \end{bmatrix}$.

C Derivation of the EM Algorithm for Mixtures of Factors Factor Analyser

The expectation maximisation algorithm for factor analysis with mixtures of factors is presented in this appendix. The E step is a bit more complicated than in general factor analysis. The M step is very generic and can be applied to the factor analysed hidden Markov models as well.

C.1 E Step for Mixtures of Factors

The joint posterior of the observation and the state vectors given the mixture component can be written as

$$p(\boldsymbol{o}_j, \boldsymbol{x}|n) = \mathcal{N}\left(\begin{bmatrix} \boldsymbol{o}_j \\ \boldsymbol{x} \end{bmatrix}; \begin{bmatrix} \boldsymbol{C}\boldsymbol{\mu}_n^{(x)} + \boldsymbol{\mu}^{(o)} \\ \boldsymbol{\mu}_n^{(x)} \end{bmatrix}, \begin{bmatrix} \boldsymbol{C}\boldsymbol{\Sigma}_n^{(x)}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)} & \boldsymbol{C}\boldsymbol{\Sigma}_n^{(x)} \\ \boldsymbol{\Sigma}_n^{(x)}\boldsymbol{C}' & \boldsymbol{\Sigma}_n^{(x)} \end{bmatrix}\right) \tag{186}$$

Using the conditioning of multivariate Gaussians presented in Appendix A.2 the posterior of the state vector given the observation and the mixture component is

$$p(\boldsymbol{x}|\boldsymbol{o}_j, n) = \mathcal{N}\big(\boldsymbol{x}; \boldsymbol{\mu}_n^{(x)} + \boldsymbol{K}_n(\boldsymbol{o}_j - \boldsymbol{C}\boldsymbol{\mu}_n^{(x)} - \boldsymbol{\mu}^{(o)}), \boldsymbol{\Sigma}_n^{(x)} - \boldsymbol{K}_n\boldsymbol{C}\boldsymbol{\Sigma}_n^{(x)}\big) \tag{187}$$

where $\boldsymbol{K}_n = \boldsymbol{\Sigma}_n^{(x)}\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{\Sigma}_n^{(x)}\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)-1}$ and the sufficient statistics needed for the E step are

$$\hat{\boldsymbol{x}}_{jn} = E\{\boldsymbol{x}|\boldsymbol{o}_j, n\} = \boldsymbol{\mu}_n^{(x)} + \boldsymbol{K}_n(\boldsymbol{o}_j - \boldsymbol{C}\boldsymbol{\mu}_n^{(x)} - \boldsymbol{\mu}^{(o)}) \tag{188}$$
$$\hat{\boldsymbol{R}}_{jn} = E\{\boldsymbol{x}\boldsymbol{x}'|\boldsymbol{o}_j, n\} = \boldsymbol{\Sigma}_n^{(x)} - \boldsymbol{K}_n\boldsymbol{C}\boldsymbol{\Sigma}_n^{(x)} + \hat{\boldsymbol{x}}_{jn}\hat{\boldsymbol{x}}_{jn}' \tag{189}$$

In case of fully tied observation process the posteriors of the state vector given the observation has to be obtained. These are easily obtained using the mixture component posteriors and the above state posteriors as follows

$$\hat{\boldsymbol{x}}_j = E\{\boldsymbol{x}|\boldsymbol{o}_j\} = \sum_{n=1}^{M^{(x)}} P(n|\boldsymbol{o}_j)E\{\boldsymbol{x}|\boldsymbol{o}_j, n\} = \sum_{n=1}^{M^{(x)}} \gamma_{jn}\hat{\boldsymbol{x}}_{jn} \tag{190}$$
$$\hat{\boldsymbol{R}}_j = E\{\boldsymbol{x}\boldsymbol{x}'|\boldsymbol{o}_j\} = \sum_{n=1}^{M^{(x)}} P(n|\boldsymbol{o}_j)E\{\boldsymbol{x}\boldsymbol{x}'|\boldsymbol{o}_j, n\} = \sum_{n=1}^{M^{(x)}} \gamma_{jn}\hat{\boldsymbol{R}}_{jn} \tag{191}$$

C.2 M Step for the Mixture Component Priors

To find the new mixture component priors, $\hat{c}_n^{(x)}$, the following auxiliary function has to be maximised

$$\mathcal{Q}_\omega(\mathcal{M}, \hat{\mathcal{M}}) = \sum_{j=1}^{N}\sum_{n=1}^{M^{(x)}} \gamma_{jn}\log\hat{c}_n^{(x)} \tag{192}$$

which is not possible by just setting the first derivative to zero since the logarithm is monotonically increasing function. The constraint $\sum_{n=1}^{M^{(x)}} \hat{c}_n^{(x)} = 1$ can be used to find the maximum using Lagrange method. By introducing a Lagrange multiplier λ, the maximisation of Eq. 192 is equivalent to maximising the following function

$$\mathcal{L}(\hat{c}_n^{(x)}, \lambda) = \sum_{j=1}^{N} \sum_{n=1}^{M^{(x)}} \gamma_{jn} \log \hat{c}_n^{(x)} - \lambda\Big(\sum_{n=1}^{M^{(x)}} \hat{c}_n^{(x)} - 1\Big) \tag{193}$$

Differentiating the above equation with respect to $\hat{c}_n^{(x)}$ and λ and setting the derivatives to zeroes lead to the following pair of equations

$$\sum_{j=1}^{N} \gamma_{jn} \frac{1}{\hat{c}_n^{(x)}} - \lambda = 0 \tag{194}$$

$$\sum_{n=1}^{M^{(x)}} \hat{c}_n^{(x)} - 1 = 0 \tag{195}$$

By eliminating λ, the new component priors can be found as follows

$$\hat{c}_n^{(x)} = \frac{1}{N} \sum_{j=1}^{N} \gamma_{jn} \tag{196}$$

C.3 M Step for Mixtures of Factors

To find the new state parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{\mu}}_n^{(x)}, \hat{\boldsymbol{\Sigma}}_n^{(x)})$, the following auxiliary function has to be maximised

$$\mathcal{Q}_s(\mathcal{M}, \hat{\mathcal{M}}) = -\frac{1}{2} \sum_{j=1}^{N} \sum_{n=1}^{M^{(x)}} \gamma_{jn} \Big(\log |\hat{\boldsymbol{\Sigma}}_n^{(x)}| + E\Big\{ (\boldsymbol{x} - \hat{\boldsymbol{\mu}}_n^{(x)})' \hat{\boldsymbol{\Sigma}}_n^{(x)-1} (\boldsymbol{x} - \hat{\boldsymbol{\mu}}_n^{(x)}) \Big| \boldsymbol{o}_j, n, \mathcal{M} \Big\} \Big) \tag{197}$$

Differentiating Eq. 197 with respect to the mixture component mean vector, $\hat{\boldsymbol{\mu}}_n^{(x)}$, and setting the resulting equation to zero, the new mean vectors can be solved using the sufficient statistics from the E step as follows

$$\frac{\partial \mathcal{Q}_s(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\mu}}_n^{(x)}} = \hat{\boldsymbol{\Sigma}}_n^{(x)-1} \sum_{j=1}^{N} \gamma_{jn} (\hat{\boldsymbol{\mu}}_n^{(x)} - \hat{\boldsymbol{x}}_{jn}) = \mathbf{0} \tag{198}$$

$$\hat{\boldsymbol{\mu}}_n^{(x)} = \frac{\sum_{j=1}^{N} \gamma_{jn} \hat{\boldsymbol{x}}_{jn}}{\sum_{j=1}^{N} \gamma_{jn}} \tag{199}$$

To find the new state noise covariance matrix, it is easier to differentiate Eq. 197 with respect to the inverse of $\hat{\boldsymbol{\Sigma}}_n^{(x)}$ as follows

$$\frac{\partial \mathcal{Q}_s(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\Sigma}}_n^{(x)-1}} = \frac{1}{2} \sum_{j=1}^{N} \gamma_{jn} \Big(\hat{\boldsymbol{\Sigma}}_n^{(x)} - (\hat{\boldsymbol{R}}_{jn} - \hat{\boldsymbol{x}}_{jn} \hat{\boldsymbol{\mu}}_n^{(x)\prime} - \hat{\boldsymbol{\mu}}_n^{(x)} \hat{\boldsymbol{x}}'_{jn} + \hat{\boldsymbol{\mu}}_n^{(x)} \hat{\boldsymbol{\mu}}_n^{(x)\prime}) \Big) = \mathbf{0} \tag{200}$$

$$\hat{\boldsymbol{\Sigma}}_n^{(x)} = \frac{\sum_{j=1}^{N} \gamma_{jn} \hat{\boldsymbol{R}}_{jn}}{\sum_{j=1}^{N} \gamma_{jn}} - \hat{\boldsymbol{\mu}}_n^{(x)} \hat{\boldsymbol{\mu}}_n^{(x)\prime} \tag{201}$$

D Derivation of the EM Algorithm for Shared Factor Analysis

The expectation maximisation algorithm for shared factor analysis is presented in this appendix. It was first introduced in [22] but the derivation is in a simple consistent form. The E step is very similar to the factor analysis but the statistics have to be estimated for each component and the component posteriors have to be obtained as well. The M step is again very generic and can be applied to the mixture of linear dynamical systems as well as factor analysed hidden Markov models.

D.1 E Step for Shared Factor Analysis

In addition to the state posteriors, the component posteriors have to be estimated as follows

$$\gamma_{jm} = P_\omega(m|\boldsymbol{o}_j, \mathcal{M}) = \frac{c_m^{(o)} p(\boldsymbol{o}_j|m, \mathcal{M})}{\sum_{l=1}^{M^{(o)}} c_l^{(o)} p(\boldsymbol{o}_j|l, \mathcal{M})} \tag{202}$$

where the component posterior of the observation is simply

$$p(\boldsymbol{o}_j|m, \mathcal{M}) = \mathcal{N}(\boldsymbol{o}_j; \boldsymbol{\mu}_m^{(o)}, \boldsymbol{C}\boldsymbol{C}' + \boldsymbol{\Sigma}_m^{(o)}) \tag{203}$$

The state posteriors can be obtained like in the factor analysis by defining the state posteriors given the observation and the mixture component as follows

$$\hat{\boldsymbol{x}}_{jm} = \boldsymbol{K}_m(\boldsymbol{o}_j - \boldsymbol{\mu}_m) \tag{204}$$

$$\hat{\boldsymbol{R}}_{jm} = \boldsymbol{I} - \boldsymbol{K}_m\boldsymbol{C} + \hat{\boldsymbol{x}}_{jm}\hat{\boldsymbol{x}}_{jm}' \tag{205}$$

where $\boldsymbol{K}_m = \boldsymbol{C}'(\boldsymbol{C}\boldsymbol{C}' + \boldsymbol{\Sigma}_m^{(o)})^{-1}$.

D.2 M Step for Shared Factor Analysis

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{C}}, \hat{\boldsymbol{\mu}}_m^{(o)}, \hat{\boldsymbol{\Sigma}}_m^{(o)})$, for the observation process the following auxiliary function has to be maximised

$$\mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}}) = \tag{206}$$
$$-\frac{1}{2}\sum_{j=1}^{N}\sum_{m=1}^{M^{(o)}} \gamma_{jm}\Big(\log|\hat{\boldsymbol{\Sigma}}_m^{(o)}| + E\Big\{(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\boldsymbol{x} - \hat{\boldsymbol{\mu}}_m^{(o)})'\hat{\boldsymbol{\Sigma}}_m^{(o)-1}(\boldsymbol{o}_j - \hat{\boldsymbol{C}}\boldsymbol{x} - \hat{\boldsymbol{\mu}}_m^{(o)})\Big|\boldsymbol{O}, \mathcal{M}\Big\}\Big)$$

The M step in case of mixture models is not as elegant as before since the optimisation of the observation matrix cannot be done without using the old model parameters. In case of full covariance matrices the optimisation is a bit difficult. Fortunately, if diagonal covariance matrices were used, the auxiliary function can be rewritten ignoring all terms independent of the observation matrix and defining $\hat{\boldsymbol{B}} = \hat{\boldsymbol{C}}'$ as follows

$$\mathcal{Q}_o'(\mathcal{M}, \hat{\mathcal{M}}) = -\frac{1}{2}\sum_{l=1}^{p}\Big(\hat{\boldsymbol{b}}_l'\boldsymbol{G}_l\hat{\boldsymbol{b}}_l - 2\hat{\boldsymbol{b}}_l'\boldsymbol{k}_l\Big) \tag{207}$$

where $\hat{\boldsymbol{b}}_l$ is the lth column of the matrix $\hat{\boldsymbol{B}}$ and

$$\boldsymbol{G}_l = \sum_{m=1}^{M^{(o)}} \frac{1}{\sigma_{ml}^{(o)2}} \sum_{j=1}^{N} \gamma_{jm}\hat{\boldsymbol{R}}_{jm} \tag{208}$$

$$\boldsymbol{k}_l = \sum_{m=1}^{M^{(o)}} \frac{1}{\sigma_{ml}^{(o)2}} \sum_{j=1}^{N} \gamma_{jm}(o_{jl} - \mu_{ml}^{(o)})\hat{\boldsymbol{x}}_{jm} \tag{209}$$

where $\sigma^{(o)2}_{ml}$ is the lth diagonal element of the observation noise covariance matrix $\boldsymbol{\Sigma}^{(o)}_m$, o_{jl} is the lth element of the observation vector $\boldsymbol{o}_j$ and $\mu^{(o)}_{ml}$ is the lth element of the observation noise mean vector $\boldsymbol{\mu}^{(o)}_m$.

Differentiating the auxiliary function in Eq. 207 with respect to $\hat{\boldsymbol{b}}_l$ and equating the resulting expression to zero, the vectors $\hat{\boldsymbol{b}}_l$ can be re-estimated as follows

$$\frac{\partial \mathcal{Q}'_o(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{b}}_l} = \boldsymbol{k}_l - \boldsymbol{G}_l \hat{\boldsymbol{b}}_l = \boldsymbol{0} \tag{210}$$

$$\hat{\boldsymbol{b}}_l = \boldsymbol{G}_l^{-1} \boldsymbol{k}_l \tag{211}$$

which are the row vectors of the new observation matrix $\hat{\boldsymbol{C}}$.

Differentiating Eq. 206 with respect to the component observation mean vectors and equating the resulting expression to zero, the new estimates for the means can be obtained as follows

$$\frac{\partial \mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\mu}}^{(o)}_m} = \hat{\boldsymbol{\Sigma}}^{(o)-1}_m \sum_{j=1}^{N} \gamma_{jm} (\boldsymbol{o}_j - \hat{\boldsymbol{C}} \hat{\boldsymbol{x}}_{jm} - \hat{\boldsymbol{\mu}}^{(o)}_m) = \boldsymbol{0} \tag{212}$$

$$\hat{\boldsymbol{\mu}}^{(o)}_m = \frac{\sum_{j=1}^{N} \gamma_{jm} (\boldsymbol{o}_j - \hat{\boldsymbol{C}} \hat{\boldsymbol{x}}_{jm})}{\sum_{j=1}^{N} \gamma_{jm}} \tag{213}$$

To find the new component observation noise covariance matrices, it is easier to differentiate Eq. 206 with respect to the inverse of $\boldsymbol{\Sigma}^{(o)}_m$ as follows

$$\frac{\partial \mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\Sigma}}^{(o)-1}_m} = \tag{214}$$

$$\frac{1}{2} \sum_{j=1}^{N} \gamma_{jm} \Big(\hat{\boldsymbol{\Sigma}}^{(o)}_m - E\Big\{ (\boldsymbol{o}_j - \hat{\boldsymbol{C}} \boldsymbol{x} - \hat{\boldsymbol{\mu}}^{(o)}_m)(\boldsymbol{o}_j - \hat{\boldsymbol{C}} \boldsymbol{x} - \hat{\boldsymbol{\mu}}^{(o)}_m)' \Big| \boldsymbol{O}, \mathcal{M} \Big\} \Big) = \boldsymbol{0}$$

$$\begin{aligned} \hat{\boldsymbol{\Sigma}}^{(o)}_m = {} & \frac{1}{\sum_{j=1}^{N} \gamma_{jm}} \sum_{j=1}^{N} \gamma_{jm} \mathrm{diag}\Big(\boldsymbol{o}_j \boldsymbol{o}'_j - \big[\, \hat{\boldsymbol{C}} \;\; \hat{\boldsymbol{\mu}}^{(o)}_m \,\big] \big[\, \boldsymbol{o}_j \hat{\boldsymbol{x}}'_{jm} \;\; \boldsymbol{o}_j \,\big]' \\ & - \big[\, \boldsymbol{o}_j \hat{\boldsymbol{x}}'_{jm} \;\; \boldsymbol{o}_j \,\big] \big[\, \hat{\boldsymbol{C}} \;\; \hat{\boldsymbol{\mu}}^{(o)}_m \,\big]' + \big[\, \hat{\boldsymbol{C}} \;\; \hat{\boldsymbol{\mu}}^{(o)}_m \,\big] \begin{bmatrix} \hat{\boldsymbol{R}}_{jm} & \hat{\boldsymbol{x}}_{jm} \\ \hat{\boldsymbol{x}}'_{jm} & 1 \end{bmatrix} \big[\, \hat{\boldsymbol{C}} \;\; \hat{\boldsymbol{\mu}}^{(o)}_m \,\big]' \Big) \end{aligned} \tag{215}$$

Unfortunately, this expression cannot be further simplified due to the form of the observation matrix re-estimation formula.

E Derivation of the EM Algorithm for Linear Dynamical System

Derivation of the Kalman filter and smoother recursions as well as the M step of the EM algorithm for a linear dynamical system are presented in this appendix. The Kalman filter and smoother recursions are derived using the generic forward-backward algorithm and conditioning multivariate Gaussians as described in App. A.2. The M step is rather straightforward maximisation of quadratic matrix forms. The derivation presented here is very similar to [16] but more attention is paid to the matrix notation used in this paper.

E.1 Kalman Filter

Following the derivation in [32] so called generic scaled forward-backward algorithm is used. Let the forward variable, $\alpha_{\boldsymbol{x}}(t)$, be defined as follows

$$\alpha_{\boldsymbol{x}}(t) = p(\boldsymbol{x}_t, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_t) = p(\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t)p(\boldsymbol{x}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t) = \Big(\prod_{\tau=1}^{t} \kappa_\tau\Big)\hat{\alpha}_{\boldsymbol{x}}(t) \tag{216}$$

where $\kappa_t = p(\boldsymbol{o}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1})$ are the scaling factors and $\hat{\alpha}_{\boldsymbol{x}}(t) = p(\boldsymbol{x}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t)$ are the scaled forward variables. Both the scaling factors and forward variables are Gaussian distributed due to the linear Gaussian model assumption. Let $\boldsymbol{x}^{(\tau)}(t)$ denote $E\{\boldsymbol{x}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_\tau\}$ and $\boldsymbol{\Sigma}^{(\tau)}(t)$ denote $E\{(\boldsymbol{x}_t - \boldsymbol{x}^{(\tau)}(t))(\boldsymbol{x}_t - \boldsymbol{x}^{(\tau)\prime}(t))|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_\tau\}$ then the scaled forward variable is distributed as

$$\hat{\alpha}_{\boldsymbol{x}}(t) = \mathcal{N}\big(\boldsymbol{x}_t; \boldsymbol{x}^{(t)}(t), \boldsymbol{\Sigma}^{(t)}(t)\big) \tag{217}$$

Using the first-order Markov property and the state conditional independence of the observations in Eqs. 51 and 52 the forward variable, $\alpha_{\boldsymbol{x}}(t)$, can be rewritten in the following recursive form

$$\begin{aligned}
\alpha_{\boldsymbol{x}}(t) &= p(\boldsymbol{x}_t, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_t) = p(\boldsymbol{o}_t|\boldsymbol{x}_t)p(\boldsymbol{x}_t, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1}) && (218)\\
&= p(\boldsymbol{o}_t|\boldsymbol{x}_t)\int_{\boldsymbol{z}} p(\boldsymbol{x}_t, \boldsymbol{x}_{t-1} = \boldsymbol{z}, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1}) && (219)\\
&= p(\boldsymbol{o}_t|\boldsymbol{x}_t)\int_{\boldsymbol{z}} p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1} = \boldsymbol{z})p(\boldsymbol{o}_{t-1}|\boldsymbol{x}_{t-1} = \boldsymbol{z})p(\boldsymbol{x}_{t-1} = \boldsymbol{z}, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1}) && (220)\\
&= p(\boldsymbol{o}_t|\boldsymbol{x}_t)\int_{\boldsymbol{z}} p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1} = \boldsymbol{z})\Big(\prod_{\tau=1}^{t-1} \kappa_\tau\Big)\hat{\alpha}_{\boldsymbol{z}}(t-1) && (221)\\
\kappa_t\hat{\alpha}_{\boldsymbol{x}}(t) &= p(\boldsymbol{o}_t|\boldsymbol{x}_t)\int_{\boldsymbol{z}} p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1} = \boldsymbol{z})\hat{\alpha}_{\boldsymbol{z}}(t-1) && (222)
\end{aligned}$$

which starts off with the initial value $\hat{\alpha}_{\boldsymbol{x}}(1) = p(\boldsymbol{x}_1|\boldsymbol{o}_1)$. All the terms in Eq. 222 can be expressed with Gaussians as follows

$$\begin{aligned}
&\kappa_t\hat{\alpha}_{\boldsymbol{x}}(t) = && (223)\\
&\quad \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)})\int_{\boldsymbol{z}} \mathcal{N}\big(\boldsymbol{x}_t; \boldsymbol{A}\boldsymbol{z} + \boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)}\big)\mathcal{N}\big(\boldsymbol{z}; \boldsymbol{x}^{(t-1)}(t-1), \boldsymbol{\Sigma}^{(t-1)}(t-1)\big)
\end{aligned}$$

where the both terms inside the integral are also jointly Gaussian as follows

$$\begin{aligned}
&p(\boldsymbol{z}, \boldsymbol{x}_t) = && (224)\\
&\quad \mathcal{N}\Big(\begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{x}_t \end{bmatrix}; \begin{bmatrix} \boldsymbol{x}^{(t-1)}(t-1) \\ \boldsymbol{A}\boldsymbol{x}^{(t-1)}(t-1) + \boldsymbol{\mu}^{(x)} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}^{(t-1)}(t-1) & \boldsymbol{\Sigma}^{(t-1)}(t-1)\boldsymbol{A}' \\ \boldsymbol{A}\boldsymbol{\Sigma}^{(t-1)}(t-1) & \boldsymbol{A}\boldsymbol{\Sigma}^{(t-1)}(t-1)\boldsymbol{A}' + \boldsymbol{\Sigma}^{(x)} \end{bmatrix}\Big)
\end{aligned}$$

and since the integration is carried out with respect to $\boldsymbol{z}$ the term on the right hand side reduces to the probability of $\boldsymbol{x}_t$ and by defining $\boldsymbol{x}^{(t-1)}(t) = \boldsymbol{A}\boldsymbol{x}^{(t-1)}(t-1) + \boldsymbol{\mu}^{(x)}$ and $\boldsymbol{\Sigma}^{(t-1)}(t) = \boldsymbol{A}\boldsymbol{\Sigma}^{(t-1)}(t-1)\boldsymbol{A}' + \boldsymbol{\Sigma}^{(x)}$, Eq. 223 becomes

$$\begin{aligned}
\kappa_t\hat{\alpha}_{\boldsymbol{x}}(t) &= \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{\mu}^{(o)}, \boldsymbol{\Sigma}^{(o)})\mathcal{N}\big(\boldsymbol{x}_t; \boldsymbol{x}^{(t-1)}(t), \boldsymbol{\Sigma}^{(t-1)}(t)\big) && (225)\\
&= p(\boldsymbol{o}_t|\boldsymbol{x}_t, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1})p(\boldsymbol{x}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1}) && (226)
\end{aligned}$$

which is not yet in the required form but can be transformed to it by using the Bayes' formula as follows

$$p(\boldsymbol{o}_t|\boldsymbol{x}_t, \boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1})p(\boldsymbol{x}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1}) = p(\boldsymbol{o}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1})p(\boldsymbol{x}_t|\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t) \tag{227}$$

and noticing that $\boldsymbol{o}_t$ and $\boldsymbol{x}_t$ are also jointly Gaussian

$$p(\boldsymbol{o}_t, \boldsymbol{x}_t) = \mathcal{N}\Big(\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{o}_t \end{bmatrix}; \begin{bmatrix} \boldsymbol{x}^{(t-1)}(t) \\ \boldsymbol{C}\boldsymbol{x}^{(t-1)}(t) + \boldsymbol{\mu}^{(o)} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}^{(t-1)}(t) & \boldsymbol{\Sigma}^{(t-1)}(t)\boldsymbol{C}' \\ \boldsymbol{C}\boldsymbol{\Sigma}^{(t-1)}(t) & \boldsymbol{C}\boldsymbol{\Sigma}^{(t-1)}(t)\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)} \end{bmatrix} \Big) \tag{228}$$

The latter form in Eq. 227 can be easily obtained by applying the conditioning of two multivariate Gaussians in Eq. 167. Therefore Eq. 225 becomes

$$\begin{aligned} \kappa_t \hat{\alpha}_{\boldsymbol{x}}(t) &= \mathcal{N}\big(\boldsymbol{o}_t; \boldsymbol{C}\boldsymbol{x}^{(t-1)}(t) + \boldsymbol{\mu}^{(o)}, \boldsymbol{C}\boldsymbol{\Sigma}^{(t-1)}(t)\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)}\big) \qquad (229) \\ & \quad \mathcal{N}\big(\boldsymbol{x}_t; \boldsymbol{x}^{(t-1)}(t) + \boldsymbol{K}(t)(\boldsymbol{o}_t - \boldsymbol{C}\boldsymbol{x}^{(t-1)}(t) - \boldsymbol{\mu}^{(o)}), \boldsymbol{\Sigma}^{(t-1)}(t) - \boldsymbol{K}(t)\boldsymbol{C}\boldsymbol{\Sigma}^{(t-1)}(t)\big) \end{aligned}$$

where $\boldsymbol{K}(t) = \boldsymbol{\Sigma}^{(t-1)}(t)\boldsymbol{C}'\big(\boldsymbol{C}\boldsymbol{\Sigma}^{(t-1)}(t)\boldsymbol{C}' + \boldsymbol{\Sigma}^{(o)}\big)^{-1}$ and which is now the product of the probability of the current observation given the history up to the time instant t and the scaled forward variable as defined earlier.

E.2 Kalman Smoother

The backward variable is defined as usual

$$\begin{aligned} & \beta_{\boldsymbol{x}}(t) = \qquad (230) \\ & p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T | \boldsymbol{x}_t) = p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T | \boldsymbol{o}_1, \ldots, \boldsymbol{o}_t) \frac{p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T | \boldsymbol{x}_t)}{p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T | \boldsymbol{o}_1, \ldots, \boldsymbol{o}_t)} \Big(\prod_{\tau=t+1}^{T} \kappa_\tau \Big) \hat{\beta}_{\boldsymbol{x}}(t) \end{aligned}$$

where $\kappa_t = p(\boldsymbol{o}_t | \boldsymbol{o}_1, \ldots, \boldsymbol{o}_{t-1})$ are the same scaling factors as the ones used with the forward variables. Using again the first-order Markov property and the state conditional independence of the observations in Eqs. 51 and 52 the backward variable, $\beta_{\boldsymbol{x}}(t-1)$, can be rewritten in the following recursive form

$$\begin{aligned} \beta_{\boldsymbol{x}}(t-1) &= \int_{\boldsymbol{z}} p(\boldsymbol{x}_t = \boldsymbol{z}, \boldsymbol{o}_t, \ldots, \boldsymbol{o}_T | \boldsymbol{x}_{t-1}) & (231) \\ &= \int_{\boldsymbol{z}} p(\boldsymbol{x}_t = \boldsymbol{z} | \boldsymbol{x}_{t-1}) p(\boldsymbol{o}_t | \boldsymbol{x}_t = \boldsymbol{z}) p(\boldsymbol{o}_{t+1} \ldots, \boldsymbol{o}_T | \boldsymbol{x}_t = \boldsymbol{z}) & (232) \\ &= \int_{\boldsymbol{z}} p(\boldsymbol{x}_t = \boldsymbol{z} | \boldsymbol{x}_{t-1}) p(\boldsymbol{o}_t | \boldsymbol{x}_t = \boldsymbol{z}) \Big(\prod_{\tau=t+1}^{T} \kappa_\tau \Big) \hat{\beta}_{\boldsymbol{z}}(t) & (233) \\ \hat{\beta}_{\boldsymbol{x}}(t-1) &= \frac{1}{\kappa_t} \int_{\boldsymbol{z}} p(\boldsymbol{x}_t = \boldsymbol{z} | \boldsymbol{x}_{t-1}) p(\boldsymbol{o}_t | \boldsymbol{x}_t = \boldsymbol{z}) \hat{\beta}_{\boldsymbol{z}}(t) & (234) \end{aligned}$$

The Kalman smoother estimates, $\hat{\boldsymbol{x}}(t)$ and $\hat{\boldsymbol{\Sigma}}(t)$, are the mean vector and covariance matrix of the state vector, $\boldsymbol{x}_t$, given the entire observation sequence, $\boldsymbol{O}$. Therefore it can be represented as the product of the scaled forward and backward variables as follows

$$\begin{aligned} \hat{\alpha}_{\boldsymbol{x}}(t)\hat{\beta}_{\boldsymbol{x}}(t) &= p(\boldsymbol{x}_t | \boldsymbol{o}_1, \ldots, \boldsymbol{o}_t) \frac{p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T | \boldsymbol{x}_t)}{p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T | \boldsymbol{o}_1, \ldots, \boldsymbol{o}_t)} & (235) \\ &= p(\boldsymbol{x}_t | \boldsymbol{O}) = \mathcal{N}\big(\boldsymbol{x}_t; \hat{\boldsymbol{x}}(t), \hat{\boldsymbol{\Sigma}}(t)\big) & (236) \end{aligned}$$

The backward recursion can be derived by substituting 53, 54, 217, 225, 234 and 236 into the product of the scaled forward and backward variables and doing some algebra as follows

$$
\begin{aligned}
\hat{\alpha}_{\boldsymbol{x}}(t)\hat{\beta}_{\boldsymbol{x}}(t) &= \mathcal{N}\big(\boldsymbol{x}_t; \boldsymbol{x}^{(t)}(t), \boldsymbol{\Sigma}^{(t)}(t)\big) \int_{\boldsymbol{z}} \mathcal{N}(\boldsymbol{z}; \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)}) \mathcal{N}(\boldsymbol{o}_{t+1}; \boldsymbol{C}\boldsymbol{z}, \boldsymbol{\Sigma}^{(o)}) && (237) \\
&\quad \frac{\mathcal{N}\big(\boldsymbol{z}; \hat{\boldsymbol{x}}(t+1), \hat{\boldsymbol{\Sigma}}(t+1)\big)}{\kappa_{t+1}\hat{\alpha}_{\boldsymbol{z}}(t+1)} \\
&= \int_{\boldsymbol{z}} \mathcal{N}\big(\boldsymbol{x}_t; \boldsymbol{x}^{(t)}(t) + \boldsymbol{J}(t)(\boldsymbol{z} - \boldsymbol{A}\boldsymbol{x}^{(t)}(t) - \boldsymbol{\mu}^{(x)}), \boldsymbol{\Sigma}^{(t)}(t) - \boldsymbol{J}(t)\boldsymbol{A}\boldsymbol{\Sigma}^{(t)}(t)\big) && (238) \\
&\quad \mathcal{N}\big(\boldsymbol{z}; \hat{\boldsymbol{x}}(t+1), \hat{\boldsymbol{\Sigma}}(t+1)\big) \\
&= \mathcal{N}\big(\boldsymbol{x}_t; \boldsymbol{x}^{(t)}(t) + \boldsymbol{J}(t)(\hat{\boldsymbol{x}}(t+1) - \boldsymbol{x}^{(t)}(t+1)), && (239) \\
&\quad \boldsymbol{\Sigma}^{(t)}(t) + \boldsymbol{J}(t)(\hat{\boldsymbol{\Sigma}}(t+1) - \boldsymbol{\Sigma}^{(t)}(t+1))\boldsymbol{J}'(t)\big) && (240)
\end{aligned}
$$

where $\boldsymbol{J}(t) = \boldsymbol{\Sigma}^{(t)}(t)\boldsymbol{A}'(\boldsymbol{\Sigma}^{(t)-1}(t+1))$ and Kalman smoother recursions result obviously.

The cross covariance of two consecutive state vectors is also required. It can be obtained by the scaled forward and backward variables as follows

$$
p(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t|\boldsymbol{O}) = \frac{1}{\kappa_t}\hat{\alpha}_{\boldsymbol{x}}(t-1)p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})p(\boldsymbol{o}_t|\boldsymbol{x}_t)\hat{\beta}_{\boldsymbol{x}_t}(t) \tag{241}
$$

Since the current state and the previous state are jointly Gaussian it is easy to obtain the cross covariance matrix $\hat{\boldsymbol{\Sigma}}^{(t-1)}(t) = \hat{\boldsymbol{\Sigma}}(t)\boldsymbol{J}'(t-1)$.

E.3 M Step

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{C}}, \hat{\boldsymbol{\mu}}^{(o)}, \hat{\boldsymbol{\Sigma}}^{(o)})$, for the observation process the following auxiliary function has to be maximised

$$
\begin{aligned}
&\mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}}) = && (242) \\
&-\frac{1}{2}\sum_{t=1}^{T}\Big(\log|\hat{\boldsymbol{\Sigma}}^{(o)}| + E\Big\{\big(\boldsymbol{o}_t - \hat{\boldsymbol{C}}\boldsymbol{x}_t - \hat{\boldsymbol{\mu}}^{(o)}\big)'\hat{\boldsymbol{\Sigma}}^{(o)-1}\big(\boldsymbol{o}_t - \hat{\boldsymbol{C}}\boldsymbol{x}_t - \hat{\boldsymbol{\mu}}^{(o)}\big)\Big|\boldsymbol{O}, \mathcal{M}\Big\}\Big)
\end{aligned}
$$

which is exactly the same as Eq. 173 in the M step for the factor analysis except the index is time, t, rather than a set index, j. So, the derivation follows directly from Appendix B.2 by replacing N with T and j with t.

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{A}}, \hat{\boldsymbol{\mu}}^{(x)}, \hat{\boldsymbol{\Sigma}}^{(x)})$, for the state evolution process the following auxiliary function has to be maximised

$$
\begin{aligned}
&\mathcal{Q}_x(\mathcal{M}, \hat{\mathcal{M}}) = && (243) \\
&-\frac{1}{2}\sum_{t=2}^{T}\Big(\log|\hat{\boldsymbol{\Sigma}}^{(x)}| + E\Big\{\big(\boldsymbol{x}_t - \hat{\boldsymbol{A}}\boldsymbol{x}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big)'\hat{\boldsymbol{\Sigma}}^{(x)-1}\big(\boldsymbol{x}_t - \hat{\boldsymbol{A}}\boldsymbol{x}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big)\Big|\boldsymbol{O}, \mathcal{M}\Big\}\Big)
\end{aligned}
$$

To find the new state evolution noise mean vector the auxiliary function in Eq. 243 can be differentiated with respect to $\hat{\boldsymbol{\mu}}^{(x)}$ as follows

$$
\begin{aligned}
\frac{\partial \mathcal{Q}_x(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\mu}}^{(x)}} &= \hat{\boldsymbol{\Sigma}}^{(x)-1}\sum_{t=2}^{T}\big(\hat{\boldsymbol{x}}(t) - \hat{\boldsymbol{A}}\hat{\boldsymbol{x}}(t-1) - \hat{\boldsymbol{\mu}}^{(x)}\big) = \boldsymbol{0} && (244) \\
\hat{\boldsymbol{\mu}}^{(x)} &= \frac{1}{T-1}\sum_{t=2}^{T}\big(\hat{\boldsymbol{x}}(t) - \hat{\boldsymbol{A}}\hat{\boldsymbol{x}}(t-1)\big) && (245)
\end{aligned}
$$

Differentiating Eq. 243 with respect to $\hat{\boldsymbol{A}}$ and setting the resulting equation to zero the new state transition matrix can be solved using only the sufficient statistics from the E step as follows

$$\frac{\partial \mathcal{Q}_x(\mathcal{M},\hat{\mathcal{M}})}{\partial \hat{\boldsymbol{A}}} = \hat{\boldsymbol{\Sigma}}^{(x)-1} \sum_{t=2}^{T} \big(\hat{\boldsymbol{R}}^{(t-1)}(t) - \hat{\boldsymbol{\mu}}^{(x)}\hat{\boldsymbol{x}}'(t-1) - \hat{\boldsymbol{A}}\hat{\boldsymbol{R}}(t-1)\big) = \mathbf{0} \tag{246}$$

$$\begin{aligned} \hat{\boldsymbol{A}} = & \Big(\sum_{t=2}^{T} \hat{\boldsymbol{R}}^{(t-1)}(t) - \frac{1}{T-1}\sum_{t=2}^{T}\hat{\boldsymbol{x}}(t)\sum_{t=2}^{T}\hat{\boldsymbol{x}}'(t-1)\Big) \\ & \Big(\sum_{t=2}^{T} \hat{\boldsymbol{R}}(t-1) - \frac{1}{T-1}\sum_{t=2}^{T}\hat{\boldsymbol{x}}(t-1)\sum_{t=2}^{T}\hat{\boldsymbol{x}}'(t-1)\Big)^{-1} \end{aligned} \tag{247}$$

When re-estimating the model parameters, the state evolution matrix has to be updated before the state evolution noise mean vector or they can be re-estimated simultaneously using matrix notation. By defining the following sums of the sufficient statistics

$$\begin{aligned} \boldsymbol{\Gamma}_3 &= \sum_{t=2}^{T} \hat{\boldsymbol{R}}^{(t-1)}(t) \quad , \boldsymbol{\Gamma}_4 = \sum_{t=2}^{T} \hat{\boldsymbol{R}}(t-1) \\ \boldsymbol{\zeta}_3 &= \sum_{t=2}^{T} \hat{\boldsymbol{x}}(t) \quad , \boldsymbol{\zeta}_4 = \sum_{t=2}^{T} \hat{\boldsymbol{x}}(t-1) \end{aligned} \tag{248}$$

it can can be checked that the parameters $\hat{\boldsymbol{A}}$ and $\hat{\boldsymbol{\mu}}^{(x)}$ can be estimated simultaneously as follows

$$\begin{aligned} & \big[\, \hat{\boldsymbol{A}} \ \hat{\boldsymbol{\mu}}^{(x)} \big] = \\ & \Big(\sum_{t=2}^{T} \big[\, \hat{\boldsymbol{R}}^{(t-1)}(t) \ \hat{\boldsymbol{x}}(t) \big]\Big)\Big(\sum_{t=2}^{T} \begin{bmatrix} \hat{\boldsymbol{R}}(t-1) & \hat{\boldsymbol{x}}(t-1) \\ \hat{\boldsymbol{x}}'(t-1) & 1 \end{bmatrix}\Big)^{-1} = \big[\, \boldsymbol{\Gamma}_3 \ \boldsymbol{\zeta}_3 \big] \begin{bmatrix} \boldsymbol{\Gamma}_4 & \boldsymbol{\zeta}_4 \\ \boldsymbol{\zeta}_4' & T-1 \end{bmatrix}^{-1} \end{aligned} \tag{249}$$

Now, the Schur's complement, $(\boldsymbol{\Gamma}|T-1)$, in Eq. 159 for the matrix to be inverted is

$$(\boldsymbol{\Gamma}|T-1) = \boldsymbol{\Gamma}_4 - \frac{1}{T-1}\boldsymbol{\zeta}_4\boldsymbol{\zeta}_4' \tag{250}$$

and the first element of the matrix product in Eq. 249 is

$$\boldsymbol{\Gamma}_3(\boldsymbol{\Gamma}|T-1)^{-1} - \frac{1}{T-1}\boldsymbol{\zeta}_3\boldsymbol{\zeta}_4'(\boldsymbol{\Gamma}|T-1)^{-1} = \Big(\boldsymbol{\Gamma}_3 - \frac{1}{T-1}\boldsymbol{\zeta}_3\boldsymbol{\zeta}_4'\Big)\Big(\boldsymbol{\Gamma}_4 - \frac{1}{T}\boldsymbol{\zeta}_4\boldsymbol{\zeta}_4'\Big)^{-1} \tag{251}$$

which is exactly the ML estimate for $\hat{\boldsymbol{A}}$. The second element of the matrix product in Eq. 249 is

$$\begin{aligned} & -\frac{1}{T-1}\boldsymbol{\Gamma}_3(\boldsymbol{\Gamma}|T-1)^{-1}\boldsymbol{\zeta}_4 + \frac{1}{T-1}\boldsymbol{\zeta}_3 + \frac{1}{(T-1)^2}\boldsymbol{\zeta}_3\boldsymbol{\zeta}_4'(\boldsymbol{\Gamma}|T-1)^{-1}\boldsymbol{\zeta}_4 = \\ & \frac{1}{T-1}\Big(\boldsymbol{\zeta}_3 - \big(\boldsymbol{\Gamma}_3(\boldsymbol{\Gamma}|T-1)^{-1} - \frac{1}{T-1}\boldsymbol{\zeta}_3\boldsymbol{\zeta}_4'(\boldsymbol{\Gamma}|T-1)^{-1}\big)\boldsymbol{\zeta}_4\Big) \end{aligned} \tag{252}$$

which is exactly the ML estimate for $\hat{\boldsymbol{\mu}}^{(x)}$.

To find the new state evolution noise covariance matrix, the auxiliary function in Eq. 243 can be differentiated with respect to the inverse of $\hat{\boldsymbol{\Sigma}}^{(x)}$ as follows

$$\begin{aligned} & \frac{\partial \mathcal{Q}_x(\mathcal{M},\hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\Sigma}}^{(x)-1}} = \\ & \frac{1}{2}\sum_{t=2}^{T}\Big(\hat{\boldsymbol{\Sigma}}^{(x)} - E\Big\{\big(\boldsymbol{x}_t - \hat{\boldsymbol{A}}\boldsymbol{x}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big)\big(\boldsymbol{x}_t - \hat{\boldsymbol{A}}\boldsymbol{x}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big)'\Big|\boldsymbol{O},\mathcal{M}\Big\}\Big) = \mathbf{0} \end{aligned} \tag{253}$$

$$\hat{\mathbf{\Sigma}}^{(x)} = \frac{1}{T-1}\sum_{t=2}^{T}\Big(\hat{\boldsymbol{R}}_t - \Big[\ \hat{\boldsymbol{A}}\ \hat{\boldsymbol{\mu}}^{(x)}\Big]\Big[\ \hat{\boldsymbol{R}}^{(t-1)}(t)\ \hat{\boldsymbol{x}}(t)\Big]'\Big) \tag{254}$$

where the last form is obtained by noting that the terms inside the expectation in Eq. 254 can be reorganised as follows

$$\begin{aligned}&\hat{\boldsymbol{R}}_t - \Big[\ \hat{\boldsymbol{A}}\ \hat{\boldsymbol{\mu}}^{(x)}\Big]\Big[\ \hat{\boldsymbol{R}}^{(t-1)}(t)\ \hat{\boldsymbol{x}}(t)\Big]' \\ &- \Big[\ \hat{\boldsymbol{R}}^{(t-1)}(t)\ \hat{\boldsymbol{x}}(t)\Big]\Big[\ \hat{\boldsymbol{A}}\ \hat{\boldsymbol{\mu}}^{(x)}\Big]' + \Big[\ \hat{\boldsymbol{A}}\ \hat{\boldsymbol{\mu}}^{(x)}\Big]\begin{bmatrix}\hat{\boldsymbol{R}}(t-1) & \hat{\boldsymbol{x}}(t-1)\\ \hat{\boldsymbol{x}}'(t-1) & 1\end{bmatrix}\Big[\ \hat{\boldsymbol{A}}\ \hat{\boldsymbol{\mu}}^{(x)}\Big]'\end{aligned} \tag{255}$$

and substituting the first term in Eq. 249 into the place of $\Big[\ \hat{\boldsymbol{A}}\ \hat{\boldsymbol{\mu}}^{(o)}\Big]$.

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{\mu}}^{(i)}, \hat{\mathbf{\Sigma}}^{(i)})$, for the initial state distribution the following auxiliary function has to be maximised

$$\mathcal{Q}_i(\mathcal{M}, \hat{\mathcal{M}}) = -\frac{1}{2}\log|\hat{\mathbf{\Sigma}}^{(i)}| - \frac{1}{2}E\Big\{\big(\boldsymbol{x}(1) - \hat{\boldsymbol{\mu}}^{(i)}\big)'\hat{\mathbf{\Sigma}}^{(i)-1}\big(\boldsymbol{x}(1) - \hat{\boldsymbol{\mu}}^{(i)}\big)\Big\} \tag{256}$$

Differentiating 256 with respect to the initial state mean vector $\hat{\boldsymbol{\mu}}^{(i)}$ yields

$$\frac{\partial \mathcal{Q}_i(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\mu}}^{(i)}} = \hat{\mathbf{\Sigma}}^{(i)-1}(\hat{\boldsymbol{x}}(1) - \hat{\boldsymbol{\mu}}^{(i)}) = \mathbf{0} \tag{257}$$

$$\hat{\boldsymbol{\mu}}^{(i)} = \hat{\boldsymbol{x}}(1) \tag{258}$$

and with respect to the inverse of the initial state covariance matrix $\hat{\mathbf{\Sigma}}^{(i)}$ yields

$$\frac{\partial \mathcal{Q}_i(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\mathbf{\Sigma}}^{(i)-1}} = \frac{1}{2}\hat{\mathbf{\Sigma}}^{(i)} - \frac{1}{2}(\hat{\boldsymbol{R}}(1) - \hat{\boldsymbol{x}}(1)\hat{\boldsymbol{\mu}}^{(i)\prime} - \hat{\boldsymbol{\mu}}^{(i)}\hat{\boldsymbol{x}}(1)' + \hat{\boldsymbol{\mu}}^{(i)}\hat{\boldsymbol{\mu}}^{(i)\prime}) = \mathbf{0} \tag{259}$$

$$\hat{\mathbf{\Sigma}}^{(i)} = \hat{\boldsymbol{R}}(1) - \hat{\boldsymbol{\mu}}^{(i)}\hat{\boldsymbol{\mu}}^{(i)\prime} \tag{260}$$

F Derivation of the EM Algorithm for FAHMM

The expectation maximisation algorithm for factor analysed hidden Markov models is presented in this appendix. The E step is very similar to the hidden Markov models except for the different observation posteriors given the current state. The M step is also a combination of the hidden Markov model, the mixtures of factors and shared factor analysis provided the correct statistics are used.

F.1 E Step

The state posterior $p(\boldsymbol{x}_t|j,n)$ and the observation posterior $p(\boldsymbol{o}_t|\boldsymbol{x}_t,m)$ are defined as the following two Gaussians

$$p(\boldsymbol{x}_t|j,n) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_{jn}^{(x)}, \mathbf{\Sigma}_{jn}^{(x)}) \tag{261}$$

$$p(\boldsymbol{o}_t|\boldsymbol{x}_t,j,m) = \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}_j\boldsymbol{x}_t + \boldsymbol{\mu}_{jm}^{(o)}, \mathbf{\Sigma}_{jm}^{(o)}) \tag{262}$$

where $\boldsymbol{\mu}_{jn}^{(x)}$ and $\mathbf{\Sigma}_{jn}^{(x)}$ are the mean vector and covariance matrix associated with the state j and mixture component n. $\boldsymbol{C}_j$ is a p by k observation matrix and $\boldsymbol{\mu}_{jm}^{(o)}$, $\mathbf{\Sigma}_{jm}^{(o)}$ are the observation noise mean vector and covariance matrix. Since the state vectors and observations are also jointly Gaussian given the state and the mixture components it is easy to obtain the posterior distribution of an observation $\boldsymbol{o}_t$ using again the conditioning of multivariate Gaussians as follows

$$p(\boldsymbol{o}_t|j,m,n) = \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{C}_j\boldsymbol{\mu}_{jn}^{(x)} + \boldsymbol{\mu}_{jm}^{(o)}, \boldsymbol{C}_j\mathbf{\Sigma}_{jn}^{(x)}\boldsymbol{C}_j' + \mathbf{\Sigma}_{jm}^{(o)}) \tag{263}$$

It should be noted that various independence assumptions have been applied. Now, the likelihood of the observation given only the state can be marginalised by using the mixture priors and summing over the mixture components as follows

$$b_j(\boldsymbol{o}_t) = \sum_{m=1}^{M^{(o)}} c_{jm}^{(o)} \sum_{n=1}^{M^{(x)}} c_{jn}^{(x)} p(\boldsymbol{o}_t|j,m,n) \tag{264}$$

The state posteriors, $\gamma_j(t)$ and $\xi_{ij}(t)$, can be obtained using the forward-backward algorithm in the usual way. To find the joint likelihood of the state and the state and observation mixture components, $\gamma_{jmn}(t)$, the forward and backward variables are used as usual

$$\gamma_{jmn}(t) = \frac{1}{p(\boldsymbol{O})} c_{jm}^{(o)} c_{jn}^{(x)} b_{jmn}(\boldsymbol{o}_t) \sum_{i=1}^{N_s} a_{ij}\alpha_i(t-1)\beta_j(t) \tag{265}$$

where $b_{jmn}(\boldsymbol{o}_t) = p(\boldsymbol{o}_t|j,m,n)$.

The first and second-order sufficient statistics $\hat{\boldsymbol{x}}_{jmn}(t) = E\{\boldsymbol{x}_t|j,m,n,\boldsymbol{O}\}$ and $\hat{\boldsymbol{R}}_{jmn}(t) = E\{\boldsymbol{x}_t\boldsymbol{x}_t'|j,m,n,\boldsymbol{O}\}$ for the M step can be obtained by using the independence assumption of HMMs as follows

$$\begin{aligned} & p(\boldsymbol{x}_t|j,m,n,\boldsymbol{O}) = \\ & \quad \frac{p(\boldsymbol{o}_t,\boldsymbol{x}_t|j,m,n)}{p(\boldsymbol{o}_t|j,m,n)} = \mathcal{N}\big(\boldsymbol{x}_t; \boldsymbol{\mu}_{jn}^{(x)} + \boldsymbol{K}_{jmn}(\boldsymbol{o}_t - \boldsymbol{C}_j\boldsymbol{\mu}_{jn}^{(x)} - \boldsymbol{\mu}_{jm}^{(o)}), \boldsymbol{\Sigma}_{jn}^{(x)} - \boldsymbol{K}_{jmn}\boldsymbol{C}_j\boldsymbol{\Sigma}_{jn}^{(x)}\big) \end{aligned} \tag{266}$$

where $\boldsymbol{K}_{jmn} = \boldsymbol{\Sigma}_{jn}^{(x)}\boldsymbol{C}_j'(\boldsymbol{C}_j\boldsymbol{\Sigma}_{jn}^{(x)}\boldsymbol{C}_j' + \boldsymbol{\Sigma}_{jm}^{(o)})^{-1}$. Again, the joint Gaussianity of the observations and the state vectors were used as well as the conditioning of multivariate Gaussians as described in Appendix A.2. The statistics are then

$$\hat{\boldsymbol{x}}_{jmn}(t) = \boldsymbol{\mu}_{jn}^{(x)} + \boldsymbol{K}_{jmn}(\boldsymbol{o}_t - \boldsymbol{C}_j\boldsymbol{\mu}_{jn}^{(x)} - \boldsymbol{\mu}_{jm}^{(o)}) \tag{267}$$

$$\hat{\boldsymbol{R}}_{jmn}(t) = \boldsymbol{\Sigma}_{jn}^{(x)} - \boldsymbol{K}_{jmn}\boldsymbol{C}_j\boldsymbol{\Sigma}_{jn}^{(x)} + \hat{\boldsymbol{x}}_{jmn}(t)\hat{\boldsymbol{x}}_{jmn}'(t) \tag{268}$$

F.2 M Step

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\pi}_j, \hat{a}_{ij})$, for the HMM state transition matrix the following auxiliary function has to be maximised

$$\mathcal{Q}_s(\mathcal{M},\hat{\mathcal{M}}) = \sum_{j=1}^{N_s} P_{q_1}(j|\boldsymbol{O},\mathcal{M})\log\hat{\pi}_j + \sum_{t=2}^{T}\sum_{i=1}^{N_s}\sum_{j=1}^{N_s} P_{q_{t-1},q_t}(i,j|\boldsymbol{O},\mathcal{M})\log\hat{a}_{ij} \tag{269}$$

It is not possible to maximise the above expression by setting its first derivative with respect to the parameters to zero but Lagrange multipliers can be used since $\hat{\pi}_j$ and $\hat{a}_{ij}$ are subject to constraints $\sum_{j=1}^{N_s}\hat{\pi}_j = 1$ and $\forall i : \sum_{j=1}^{N_s}\hat{a}_{ij} = 1$, respectively. The derivation follows closely to the derivation of the new mixture priors in a mixture of factor analysers presented in Appendix C.2. Defining $\gamma_j(t) = P_{q_t}(j|\boldsymbol{O},\mathcal{M})$ and $\xi_{ij}(t) = P_{q_{t-1},q_t}(i,j|\boldsymbol{O},\mathcal{M})$ the ML estimates for $\hat{\pi}_j$ and $\hat{a}_{ij}$ are

$$\hat{\pi}_j = \frac{\gamma_j(1)}{\sum_{i=1}^{N_s}\gamma_i(1)} \tag{270}$$

$$\hat{a}_{ij} = \frac{\sum_{t=2}^{T}\xi_{ij}(t)}{\sum_{t=2}^{T}\gamma_i(t-1)} \tag{271}$$

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{\mu}}_{jn}^{(x)}, \hat{\boldsymbol{\Sigma}}_{jn}^{(x)})$, for the HMM state vector distributions the following auxiliary function has to be maximised

$$\mathcal{Q}_x(\mathcal{M}, \hat{\mathcal{M}}) = \tag{272}$$
$$-\frac{1}{2}\sum_{t=1}^{T}\sum_{j=1}^{N_s}\sum_{m=1}^{M^{(o)}}\sum_{n=1}^{M^{(x)}} \gamma_{jmn}(t)\Big(\log|\hat{\boldsymbol{\Sigma}}_{jn}^{(x)}| + E\Big\{(\boldsymbol{x}_t - \hat{\boldsymbol{\mu}}_{jn}^{(x)})'\hat{\boldsymbol{\Sigma}}_{jn}^{(x)-1}(\boldsymbol{x}_t - \hat{\boldsymbol{\mu}}_{jn}^{(x)})\Big|\boldsymbol{O}, j, m, n, \mathcal{M}\Big\}\Big)$$

where $\gamma_{jmn}(t) = P_{q_t,\omega^o,\omega^x}(j, m, n|\boldsymbol{O}, \mathcal{M})$.

Differentiating the auxiliary function with respect to the state mean vectors yields

$$\frac{\partial \mathcal{Q}_x(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\mu}}_{jn}^{(x)}} = \hat{\boldsymbol{\Sigma}}_{jn}^{(x)-1}\sum_{t=1}^{T}\sum_{m=1}^{M^{(o)}}\gamma_{jmn}(t)\big(\hat{\boldsymbol{\mu}}_{jn}^{(x)} - \hat{\boldsymbol{x}}_{jmn}(t)\big) = \boldsymbol{0} \tag{273}$$

$$\hat{\boldsymbol{\mu}}_{jn}^{(x)} = \frac{\sum_{t=1}^{T}\sum_{m=1}^{M^{(o)}}\gamma_{jmn}(t)\hat{\boldsymbol{x}}_{jmn}(t)}{\sum_{t=1}^{T}\sum_{m=1}^{M^{(o)}}\gamma_{jmn}(t)} \tag{274}$$

and with respect to the inverse covariance matrices yields

$$\frac{\partial \mathcal{Q}_x(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\Sigma}}_{jn}^{(x)-1}} = \tag{275}$$
$$\frac{1}{2}\sum_{t=1}^{T}\sum_{m=1}^{M^{(o)}}\gamma_{jmn}(t)\Big(\hat{\boldsymbol{\Sigma}}_{jn}^{(x)} - \big(\hat{\boldsymbol{R}}_{jmn}(t) - \hat{\boldsymbol{x}}_{jmn}(t)\hat{\boldsymbol{\mu}}_{jn}^{(x)\prime} - \hat{\boldsymbol{\mu}}_{jn}^{(x)}\hat{\boldsymbol{x}}_{jmn}'(t) + \hat{\boldsymbol{\mu}}_{jn}^{(x)}\hat{\boldsymbol{\mu}}_{jn}^{(x)\prime}\big)\Big) = \boldsymbol{0}$$

$$\hat{\boldsymbol{\Sigma}}_{jn}^{(x)} = \frac{\sum_{t=1}^{T}\sum_{m=1}^{M^{(o)}}\gamma_{jmn}(t)\hat{\boldsymbol{R}}_{jmn}(t)}{\sum_{t=1}^{T}\sum_{m=1}^{M^{(o)}}\gamma_{jmn}(t)} - \hat{\boldsymbol{\mu}}_{jn}^{(x)}\hat{\boldsymbol{\mu}}_{jn}^{(x)\prime} \tag{276}$$

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{C}}_j, \hat{\boldsymbol{\mu}}_{jm}^{(o)}, \hat{\boldsymbol{\Sigma}}_{jm}^{(o)})$, for the observation process, exactly the same auxiliary function as in case of shared factor analysis in Eq. 206 has to be maximised provided the sufficient statistics are replaced accordingly and the state-space component is marginalised by summing over n.

G Optimisation of Dynamical Linear Discriminant Analysis

The algorithm for the optimisation of the parameters of a dynamical linear discriminant analysis is presented in this appendix. The algorithm differs from the expectation maximisation algorithm presented in previous chapters in that there are no latent variables present.

G.1 Observation Matrix Optimisation

To find the new transformation matrix, $\hat{\boldsymbol{C}}$, the following auxiliary function has to be maximised

$$\mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}}) = \tag{277}$$
$$\frac{1}{2}\sum_{t=2}^{T}\big(\log(|\hat{\boldsymbol{B}}'|^2) - (\hat{\boldsymbol{B}}_{[k]}'\boldsymbol{o}_t - \boldsymbol{A}\hat{\boldsymbol{B}}_{[k]}'\boldsymbol{o}_{t-1} - \boldsymbol{\mu}^{(x)})'\boldsymbol{\Sigma}^{(x)-1}(\hat{\boldsymbol{B}}_{[k]}'\boldsymbol{o}_t - \boldsymbol{A}\hat{\boldsymbol{B}}_{[k]}'\boldsymbol{o}_{t-1} - \boldsymbol{\mu}^{(x)})$$
$$-(\hat{\boldsymbol{B}}_{[p-k]}'\boldsymbol{o}_t - \boldsymbol{\mu}^{(o)})'\boldsymbol{\Sigma}^{(o)-1}(\hat{\boldsymbol{B}}_{[p-k]}'\boldsymbol{o}_t - \boldsymbol{\mu}^{(o)})\big)$$

where $\hat{\boldsymbol{B}} = (\hat{\boldsymbol{C}}^{-1})'$. The optimisation can now be done one column of $\hat{\boldsymbol{B}}$ at a time easily if diagonal transition matrix is used. The auxiliary function can be then rewritten as follows

$$\mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}}) = \tag{278}$$
$$(T-1)\log(\hat{\boldsymbol{b}}_i'\boldsymbol{c}_i) - \frac{1}{2}\sum_{j=1}^{k}(\hat{\boldsymbol{b}}_j'\boldsymbol{G}_j^{(x)}\hat{\boldsymbol{b}}_j - 2\hat{\boldsymbol{b}}_j'\boldsymbol{k}_j^{(x)}) - \frac{1}{2}\sum_{j=k+1}^{p}(\hat{\boldsymbol{b}}_j'\boldsymbol{G}_j^{(o)}\hat{\boldsymbol{b}}_j - 2\hat{\boldsymbol{b}}_j'\boldsymbol{k}_j^{(o)})$$

where $\boldsymbol{c}_i$ is a vector of the cofactors of the matrix $\hat{\boldsymbol{B}}$ corresponding to the elements in the vector $\hat{\boldsymbol{b}}_i$ and

$$\boldsymbol{G}_j^{(x)} = \frac{1}{\sigma_j^{(x)2}}\sum_{t=2}^{T}(\boldsymbol{o}_t\boldsymbol{o}_t' - a_{jj}\boldsymbol{o}_t\boldsymbol{o}_{t-1}' - a_{jj}\boldsymbol{o}_{t-1}\boldsymbol{o}_t + a_{jj}^2\boldsymbol{o}_{t-1}\boldsymbol{o}_{t-1}') \tag{279}$$

$$\boldsymbol{k}_j^{(x)} = \frac{\mu_j^{(x)2}}{\sigma_j^{(x)2}}\sum_{t=2}^{T}(\boldsymbol{o}_t - a_{jj}\boldsymbol{o}_{t-1}) \tag{280}$$

$$\boldsymbol{G}_j^{(o)} = \frac{1}{\sigma_{(j-k)}^{(o)2}}\sum_{t=2}^{T}\boldsymbol{o}_t\boldsymbol{o}_t' \tag{281}$$

$$\boldsymbol{k}_j^{(o)} = \frac{\mu_{(j-k)}^{(o)2}}{\sigma_{(j-k)}^{(o)2}}\sum_{t=2}^{T}\boldsymbol{o}_t \tag{282}$$

where $\mu_j^{(x)2}$ and $\mu_j^{(o)2}$ are the jth elements of the observation and state evolution noise mean vectors $\boldsymbol{\mu}^{(x)}$ and $\boldsymbol{\mu}^{(o)}$, respectively, $\sigma_j^{(x)2}$, $\sigma_j^{(o)2}$ and a_{jj} are the jth diagonal elements of the observation and state evolution noise covariance, and the state transition matrices, respectively.

For the rows $1 \leq i \leq k$, differentiating the auxiliary function with respect to $\hat{\boldsymbol{b}}_i$ yields

$$\frac{\partial\mathcal{Q}_o(\mathcal{M}, \hat{\mathcal{M}})}{\partial\hat{\boldsymbol{b}}_i} = (T-1)\frac{\boldsymbol{c}_i}{\hat{\boldsymbol{b}}_i'\boldsymbol{c}_i} - \boldsymbol{G}_i^{(x)}\hat{\boldsymbol{b}}_i + \boldsymbol{k}_i^{(x)} = \boldsymbol{0} \tag{283}$$

and after rearranging yields

$$\hat{\boldsymbol{b}}_i\hat{\boldsymbol{b}}_i'\boldsymbol{c}_i = \boldsymbol{G}_i^{(x)-1}\boldsymbol{k}_i^{(x)}\hat{\boldsymbol{b}}_i'\boldsymbol{c}_i + (T-1)\boldsymbol{G}_i^{(x)-1}\boldsymbol{c}_i \tag{284}$$

Since $\hat{\boldsymbol{b}}_i'\boldsymbol{c}_i$ is a scalar and considering the direction of the vector $\hat{\boldsymbol{b}}_i$, it can be represented as follows

$$\hat{\boldsymbol{b}}_i = \delta\boldsymbol{G}_i^{(x)-1}(\lambda\boldsymbol{k}_i^{(x)} + \boldsymbol{c}_i) \tag{285}$$

and the values for δ and λ have to be found. Substituting this expression into Eq. 284, multiplying by $\boldsymbol{G}_i^{(x)}$ from the left and rearranging yields

$$\delta(\lambda\delta - 1)(\lambda\boldsymbol{k}_i^{(x)\prime} + \boldsymbol{c}_i')\boldsymbol{G}_i^{(x)-1}\boldsymbol{c}_i\boldsymbol{k}_i^{(x)} = \big((T-1) - \delta^2(\lambda\boldsymbol{k}_i^{(x)\prime} + \boldsymbol{c}_i')\boldsymbol{G}_i^{(x)-1}\boldsymbol{c}_i\big)\boldsymbol{c}_i \tag{286}$$

which holds always if and only if

$$\lambda\delta = 1 \tag{287}$$

$$(T-1) = \delta^2(\lambda\boldsymbol{k}_i^{(x)\prime} + \boldsymbol{c}_i')\boldsymbol{G}_i^{(x)-1}\boldsymbol{c}_i \tag{288}$$

and these can be expressed as one simple quadratic equation

$$\delta^2\boldsymbol{c}_i'\boldsymbol{G}_i^{(x)-1}\boldsymbol{c}_i + \delta\boldsymbol{k}_i^{(x)\prime}\boldsymbol{G}_i^{(x)-1}\boldsymbol{c}_i - (T-1) = 0 \tag{289}$$

It can be shown that both roots of this equation are maxima. Now that knowing δ and λ, the vector $\hat{\boldsymbol{b}}_i$ can be expressed as

$$\hat{\boldsymbol{b}}_i = \boldsymbol{k}_i^{(x)} \boldsymbol{G}_i^{(x)-1} (\boldsymbol{k}_i^{(x)} + \delta \boldsymbol{c}_i) \tag{290}$$

and the auxiliary function for the ith row ignoring all the terms independent of δ can be represented as follows

$$\mathcal{Q}_o^{(i)}(\mathcal{M}, \hat{\mathcal{M}}) = (T-1)\log(|\delta\epsilon_1 + \epsilon_2|) - \frac{1}{2}\delta^2\epsilon_1 \tag{291}$$

where $\epsilon_1 = \boldsymbol{c}_i' \boldsymbol{G}_i^{(x)-1} \boldsymbol{c}_i$ and $\epsilon_2 = \boldsymbol{k}_i^{(x)\prime} \boldsymbol{G}_i^{(x)-1} \boldsymbol{c}_i$. Since it is not possible to ensure that $\epsilon_2 > 0$, the value of δ is selected that maximises $\mathcal{Q}_o^{(i)}(\mathcal{M}, \hat{\mathcal{M}})$.

For the rows $k+1 \leq i \leq p$, the optimisation is exactly the same as above except for using the values $\boldsymbol{G}_i^{(o)}$ and $\boldsymbol{k}_i^{(o)}$ instead of $\boldsymbol{G}_i^{(x)}$ and $\boldsymbol{k}_i^{(x)}$.

G.2 Observation Noise Parameter Optimisation

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{\Sigma}}^{(o)}, \hat{\boldsymbol{\mu}}^{(o)})$, for the observation noise, the following auxiliary function has to be maximised

$$\mathcal{Q}_v(\mathcal{M}, \hat{\mathcal{M}}) = -\frac{1}{2}\sum_{t=2}^{T}\Big(\log|\hat{\boldsymbol{\Sigma}}^{(o)}| + \big(\hat{\boldsymbol{B}}_{[p-k]}'\boldsymbol{o}_t - \hat{\boldsymbol{\mu}}^{(o)}\big)'\hat{\boldsymbol{\Sigma}}^{(o)-1}\big(\hat{\boldsymbol{B}}_{[p-k]}'\boldsymbol{o}_t - \hat{\boldsymbol{\mu}}^{(o)}\big)\Big) \tag{292}$$

To find the new observation noise mean vector the auxiliary function in Eq. 292 has to be differentiated with respect to $\hat{\boldsymbol{\mu}}^{(o)}$ as follows

$$\frac{\partial \mathcal{Q}_v(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\mu}}^{(o)}} = \hat{\boldsymbol{\Sigma}}^{(o)-1}\sum_{t=2}^{T}\big(\hat{\boldsymbol{B}}_{[p-k]}'\boldsymbol{o}_t - \hat{\boldsymbol{\mu}}^{(o)}\big) = \boldsymbol{0} \tag{293}$$

$$\hat{\boldsymbol{\mu}}^{(o)} = \frac{1}{T-1}\hat{\boldsymbol{B}}_{[p-k]}'\sum_{t=2}^{T}\boldsymbol{o}_t \tag{294}$$

To find the new observation noise covariance matrix, the auxiliary function in Eq. 292 has to be differentiated with respect to the inverse of $\hat{\boldsymbol{\Sigma}}^{(o)}$ as follows

$$\frac{\partial \mathcal{Q}_v(\mathcal{M}, \hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\Sigma}}^{(o)-1}} = \tag{295}$$

$$\frac{1}{2}\sum_{t=2}^{T}\Big(\hat{\boldsymbol{\Sigma}}^{(o)} - \big(\hat{\boldsymbol{B}}_{[p-k]}'\boldsymbol{o}_t\boldsymbol{o}_t'\hat{\boldsymbol{B}}_{[p-k]} - \hat{\boldsymbol{\mu}}^{(o)}\boldsymbol{o}_t'\hat{\boldsymbol{B}}_{[p-k]} - \hat{\boldsymbol{B}}_{[p-k]}'\boldsymbol{o}_t\hat{\boldsymbol{\mu}}^{(o)\prime} + \hat{\boldsymbol{\mu}}^{(o)}\hat{\boldsymbol{\mu}}^{(o)\prime}\big)\Big) = \boldsymbol{0}$$

$$\hat{\boldsymbol{\Sigma}}^{(o)} = \frac{1}{T-1}\sum_{t=2}^{T}\mathrm{diag}\big(\hat{\boldsymbol{B}}_{[p-k]}'\boldsymbol{o}_t\boldsymbol{o}_t'\hat{\boldsymbol{B}}_{[p-k]} - \boldsymbol{\mu}^{(o)}\boldsymbol{\mu}^{(o)\prime}\big) \tag{296}$$

G.3 State Evolution Process Parameter Optimisation

To find the new parameters, $\hat{\mathcal{M}} = (\hat{\boldsymbol{A}}, \hat{\boldsymbol{\Sigma}}^{(x)}, \hat{\boldsymbol{\mu}}^{(x)})$, for the state evolution process the following auxiliary function has to be maximised

$$\mathcal{Q}_s(\mathcal{M}, \hat{\mathcal{M}}) = \tag{297}$$

$$-\frac{1}{2}\sum_{t=2}^{T}\Big(\log|\hat{\boldsymbol{\Sigma}}^{(x)}| + \big(\hat{\boldsymbol{B}}_{[k]}'\boldsymbol{o}_t - \hat{\boldsymbol{A}}\hat{\boldsymbol{B}}_{[k]}'\boldsymbol{o}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big)'\hat{\boldsymbol{\Sigma}}^{(o)-1}\big(\hat{\boldsymbol{B}}_{[k]}'\boldsymbol{o}_t - \hat{\boldsymbol{A}}\hat{\boldsymbol{B}}_{[k]}'\boldsymbol{o}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big)\Big)$$

To find the new state evolution noise mean vector, the auxiliary function in Eq. 297 can be differentiated with respect to $\hat{\boldsymbol{\mu}}^{(x)}$ as follows

$$\frac{\partial \mathcal{Q}_x(\mathcal{M},\hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\mu}}^{(x)}} = \hat{\boldsymbol{\Sigma}}^{(x)-1} \sum_{t=2}^{T} \big(\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_t - \hat{\boldsymbol{A}}\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big) = \boldsymbol{0} \tag{298}$$

$$\hat{\boldsymbol{\mu}}^{(x)} = \frac{1}{T-1}\sum_{t=2}^{T} \big(\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_t - \hat{\boldsymbol{A}}\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_{t-1}\big) \tag{299}$$

Differentiating Eq. 297 with respect to $\hat{\boldsymbol{A}}$ and setting the resulting equation to zero, the new state transition matrix can be solved remembering the diagonality assumption

$$\frac{\partial \mathcal{Q}_x(\mathcal{M},\hat{\mathcal{M}})}{\partial \hat{\boldsymbol{A}}} = \hat{\boldsymbol{\Sigma}}^{(x)-1} \sum_{t=2}^{T} \big(\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_t\boldsymbol{o}'_{t-1}\hat{\boldsymbol{B}}_{[k]} - \hat{\boldsymbol{\mu}}^{(x)}\boldsymbol{o}'_{t-1}\hat{\boldsymbol{B}}_{[k]} - \hat{\boldsymbol{A}}\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_{t-1}\boldsymbol{o}'_{t-1}\hat{\boldsymbol{B}}_{[k]}\big) = \boldsymbol{0} \tag{300}$$

$$\begin{aligned}\hat{\boldsymbol{A}} = {} & \Big(\mathrm{diag}\Big(\hat{\boldsymbol{B}}'_{[k]}\big(\sum_{t=2}^{T}\boldsymbol{o}_t\boldsymbol{o}'_{t-1}\big)\hat{\boldsymbol{B}}_{[k]} - \frac{1}{T-1}\hat{\boldsymbol{B}}'_{[k]}\big(\sum_{t=2}^{T}\boldsymbol{o}_t\big)\big(\sum_{t=2}^{T}\boldsymbol{o}_{t-1}\big)'\hat{\boldsymbol{B}}_{[k]}\Big)\Big) \\ & \Big(\mathrm{diag}\Big(\hat{\boldsymbol{B}}'_{[k]}\big(\sum_{t=2}^{T}\boldsymbol{o}_{t-1}\boldsymbol{o}'_{t-1}\big)\hat{\boldsymbol{B}}_{[k]} - \frac{1}{T-1}\hat{\boldsymbol{B}}'_{[k]}\big(\sum_{t=2}^{T}\boldsymbol{o}_{t-1}\big)\big(\sum_{t=2}^{T}\boldsymbol{o}_{t-1}\big)'\hat{\boldsymbol{B}}_{[k]}\Big)\Big)^{-1}\end{aligned} \tag{301}$$

which follows like the state transition matrix in case of linear dynamical systems in Appendix E.3.

To find the new state evolution noise covariance matrix, the auxiliary function in Eq. 297 can be differentiated with respect to the inverse of $\hat{\boldsymbol{\Sigma}}^{(x)}$ as follows

$$\begin{aligned}&\frac{\partial \mathcal{Q}_x(\mathcal{M},\hat{\mathcal{M}})}{\partial \hat{\boldsymbol{\Sigma}}^{(x)-1}} = \\ &\frac{1}{2}\sum_{t=2}^{T}\Big(\hat{\boldsymbol{\Sigma}}^{(x)} - \big(\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_t - \hat{\boldsymbol{A}}\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big)\big(\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_t - \hat{\boldsymbol{A}}\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_{t-1} - \hat{\boldsymbol{\mu}}^{(x)}\big)'\Big) = \boldsymbol{0}\end{aligned} \tag{302}$$

$$\hat{\boldsymbol{\Sigma}}^{(x)} = \frac{1}{T-1}\sum_{t=2}^{T}\mathrm{diag}\Big(\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_t\boldsymbol{o}'_t\hat{\boldsymbol{B}}_{[k]} - \Big[\,\hat{\boldsymbol{A}}\;\;\hat{\boldsymbol{\mu}}^{(x)}\Big]\Big[\,\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_t\boldsymbol{o}'_{t-1}\hat{\boldsymbol{B}}_{[k]}\;\;\hat{\boldsymbol{B}}'_{[k]}\boldsymbol{o}_t\Big]'\Big) \tag{303}$$

which of course holds only if diagonal covariance matrices are used.