Insperata accident magis saepe quam quae speres.
(Things you do not expect happen more often than
things you do expect) Plautus (ca 200(B.C.)

Project no: 027787

DIRAC

Detection and Identification of Rare Audio-visual Cues

Integrated Project
IST - Priority 2

DELIVERABLE NO: D2.3
Features of Audio Signals, Obtained with Different Modeling Techniques

Date of deliverable: 31.12.2006
Actual submission date: 02.02.2007

Start date of project: 01.01.2006 Duration: 60 months

Organization name of lead contractor for this deliverable: **IDIAP Research Institute**

Revision [1]

Project co-funded by the European Commission within the Sixth Framework Program (2002-2006)		
Dissemination Level		
PU	Public	
PP	Restricted to other program participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	X
CO	Confidential, only for members of the consortium (including the Commission Services)	

DIRAC

Detection and Identification of Rare Audiovisual Cues

Insperata accident magis saepe quam quae speres.
(Things you do not expect happen more often than
things you do expect) Plautus (ca 200(B.C.)

D2.3 FEATURES OF AUDIO SIGNALS, OBTAINED WITH DIFFERENT MODELING TECHNIQUES

IDIAP Research Institute (IDIAP)

Abstract:
Understanding the statistical regularities in an environment may be cast as a probabilistic model-building task. However, a key difficulty is that in many situations the quantity of interest is only a part of the complex environment, and therefore finding an appropriate signal decomposition is of considerable interest.

One case is when the signal of interest has been linearly mixed with other signals, such as might occur when several people are speaking concurrently. A theoretical contribution is made in which the signal is decomposed into approximately disjoint subunits, and is shown to outperform classical Independent Components Analysis approaches when the sources exhibit statistical dependencies.

Another important case is noise corruption. In order to tackle this issue one approach is to build a model of both the signal of interest and also any corrupting `noise' signals. In our approach we explicitly construct a forward-model and then use Bayes rule to infer the feature distribution. One may interpret the latent variables in the model as `features'. A potential advantage of this approach over the more traditional approach is that, provided a sufficiently flexible noise model is incorporated, it should be possible to separate features which are responsible for generating signal from features which are responsible for generating noise.

Table of Content

1 Overview

Understanding the statistical regularities in an environment may be cast as a probabilistic model-building task. However, a key difficulty is that in many situations the quantity of interest is only a part of the complex environment, and therefore finding an appropriate signal decomposition is of considerable interest.

One case considered in the work here is when the signal of interest has been linearly mixed with other signals, such as might occur in several people speaking concurrently. A theoretical contribution is made in which the signal is decomposed into approximately disjoint subunits, and is shown to outperform classical Independent Components Analysis approaches when the sources exhibit statistical dependencies.

Another important case is noise corruption. For example, a speech signal may be embedded within a corrupting noise signal. In order to tackle this issue one approach is to build a model of both the signal of interest and also any corrupting 'noise' signals. In this way the joint signal-noise model enables the extraction of signal from noise. In order to proceed with this framework, explicit models of both the signal and noise processes need to be made.

'Inverse' versus 'forward' Feature Modelling

The traditional viewpoint of a feature is based on an inverse-model $p(\text{features}|\text{signal})$, see Figure (1). Whilst this approach is undoubtedly successful, when the signal is heavily corrupted with noise, finding 'noise' free features is difficult. Similarly, modelling explicitly the dynamics which are corrupted with noise is highly complex. For this reason, many traditional approaches are not robust to corrupting effects in the environment.

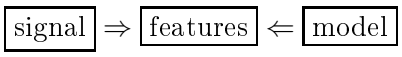


Figure 1: The traditional 'forward' approach to feature extraction and modelling. First features are extracted from the signal which are subsequently modelled, $p(\text{features}|\text{signal}, \text{model})$. Noise effects are dealt with by choosing appropriate features which are as 'noise' free as possible.

In the work carried out within the DIRAC project, the aim has been to study a different modelling strategy in which a single consistent model of the joint noise and signal environment is made, see Figure (2). In our approach we explicitly construct a forward-model $p(\text{signal}|\text{features})$ and use Bayes rule to then infer the feature distribution. In this sense, one can interpret the latent variables in the model as 'features'. A potential advantage of this approach over the more traditional approach is that, provided a sufficiently flexible noise model is incorporated, it should be possible to separate features which are responsible for generating signal from features which are responsible for generating noise.

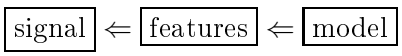


Figure 2: The 'inverse' feature technique. Here a single consistent model $p(\text{signal}|\text{features})p(\text{features}|\text{model})$ generates features which in turn generate components of the signal. If desired, features may be then extracted using Bayes rule $p(\text{features}|\text{signal}, \text{model}) \propto p(\text{signal}|\text{features})p(\text{features}|\text{model})$.

It should perhaps be emphasized that this 'forward' technique ultimately corresponds to 'throwing away useless information' about the signal, as in the more standard feature extraction technique since, in both cases, ultimately the underlying model of interest for tasks such as ASR is the one generating the feature distribution. The two approaches are therefore just different ways at extracting relevant features from a signal.
A particular focus of the work carried out has been to make specific models of acoustic signals, based mainly on extensions of linear dynamical systems. In order to improve robustness and flexibility, additional parameter priors have been incorporated leading to Bayesian treatments of the related models.

The work is composed of novel theoretical components, including

- State-of-the-art method for inference in Switching Linear Dynamical Systems
- Bayesian Switching Autoregressive Model
- State-of-the-art technique for dealing with Bayesian Linear Dynamical Systems
- Identification of independent dynamical processes underlying signal generation
- Stable inference techniques in large-scale Gaussian distributions

This work is described in detail in the following DIRAC publications:

1. D. Barber. Expectation Correction for smoothing in Switching Linear Gaussian State Space models. Journal of Machine Learning Research, 2006.
2. S. Chiappa and D. Barber. Bayesian Linear Gaussian State Space Models for Biosignal Decomposition. Signal Processing Letters, 2007.
3. D. Barber and B. Mesot. A Novel Gaussian Sum Smoother for Approximate Inference in Switching Linear Dynamical Systems. In Advances in Neural Information Processing Systems (NIPS), volume 20, 2006.
4. B. Mesot and D. Barber. A Bayesian Treatment of Gain Adaptation in Switching AR-HMMs. ICASSP 2007
5. D. Barber and S. Chiappa. Unified Inference for Variational Bayesian Linear Gaussian State-Space Models. In Advances in Neural Information Processing Systems (NIPS), volume 20, 2006.

A brief summary of the projects comprising these works is detailed below.

Switching Linear Dynamical Systems for Noisy ASR

Most modern Automatic Speech Recognition systems perform pre-processing to extract features. However, in noisy environments, such methods are often brittle since they are fine-tuned to work only in clean-speech environments. A key issue therefore in advancing recognition in real-world environments is to separate signal from noise. In our approach we use strong prior knowledge of speech at the waveform level to help separate speech from noise and perform more robust classification.

Our models are based on Switching Linear Dynamical Systems (SLDS), whereby each linear system is responsible for generating the waveform over a period of roughly 100ms. In this sense, each 'feature' is a linear signal generator. The traditional viewpoint of a feature is based on an inverse-model p(feature|signal). In our approach we explicitly construct a forward-model p(signal|feature) and use Bayes rule to then infer the feature distribution. In this sense, one can interpret the latent variables in the SLDS as 'features' that are responsible for signal segments lasting up to several hundred milliseconds.

To deal with noise, we explicitly include additional additive components to the waveform, resulting in a Dynamical Bayesian Network to model both speech and noise processes simultaneously. The advantage of this general approach is that the features for the signal can be isolated from noise corrupting effects. We then trained such models to perform ASR on a simple problem, isolated TI-DIGITS, but corrupted with very large amounts of additive Gaussian noise.

Whilst standard ASR systems perform well in low-noise environments, but degrade rapidly with increasing noise, our system degrades gracefully under very large noise. A future direction for such work is to extend the models to deal with more generic speech and noise environments in order to make a more flexible technique for separating speech from noise.

Inference in Switching Linear Dynamical Systems

Whilst the SLDS framework is conceptually straightforward, learning and inference in an SLDS is formally computationally intractable, and approximate techniques need to be developed. We presented a state-of-the-art method for inference in the generic class of Switching Linear Dynamical Systems. The method is based on a novel mixture-of-Gaussians smoother.

From a theoretical perspective, of interest in this work is the approach taken to form an approximation. In most approximation schemes an objective criterion is proposed, from which an algorithm may be developed to optimise the criterion. In our approach we start rather from the exact recursions that would result from intractable inference in the full system, and approximate these recursions. Whilst this does not derive from a simple global objective, it leads to a simple scheme that strives to remain faithful to the exact inference procedure. Extensive experiments show that the method outperforms a wide-range of competing methods.

Bayesian Switching Autoregressive Model

One of the known difficulties in the implementation of waveform level models in acoustic signal processing is the issue of gain adaptation, which refers to the changing volume levels in the environment and also in the speech recordings. We discussed how to improve flexibility of the switching linear models by placing a distribution on the noise levels, improving robustness

to the varying noise levels and thereby improving generalization performance. Our aim is to extend this class of models to deal with a wider class of speech signals in order to deal more effectively with changes in the environment.

Bayesian Linear Dynamical Systems

Linear Dynamical Systems (LDS) are one of the central tools in signal analysis. A Bayesian treatment of this extensive class of models is therefore of considerable general interest. The approximate Variational Bayesian method applied to these models is an attractive approach, used successfully in acoustics applications. The most challenging aspect of implementing the method is in performing inference on the hidden state sequence of the model. We show how to convert the inference problem so that standard and stable Kalman Filtering/Smoothing recursions from the literature may be applied. This is in contrast to previously approaches based on Belief Propagation. Our framework both simplifies and unifies the inference problem, so that future applications may be more easily developed. We hope that our approach will be the standard technique for implementing the variational approximation to Bayesian Linear Dynamical Systems.

Independent Component Signal Analysis

We applied our Bayesian Linear Dynamical System framework to a factorized latent space, which corresponds to analyzing a signal into independent components. In particular, we used our Bayesian procedure to bias each independent component to be restricted to a particular frequency band. This results in an analysis that breaks the signal into separate frequency components. Unlike the FFT, the method is flexible in that there is a prior preference to for each independent component to remain close to a desired frequency, resulting in a signal decomposition that is able to adapt to moderate changes in frequencies in the signal.

Blind Signal Separation by Disjoint Component Analysis

A novel method for blind signal separation and analysis, disjoint component analysis (DCA), is proposed which is based on minimizing the overlap of output signals, thereby making their support maximally disjoint. Performance of DCA alone and in comparison to ICA is evaluated in dependence to source overlap and source independence. It is concluded that DCA may be of particular value in applications where independence is not fulfilled and where measurement data is positive-valued.

A more detailed report of the above contributions follows.

2 Bayesian Autoregressive Hidden Markov Models

Models dealing with the raw acoustic speech signal directly are an alternative to conventional feature-based Hidden Markov Models (HMMs). One of the most popular examples is the *Autoregressive (AR) Process* which models a sample y_t of a speech signal—represented as a sequence of samples $y_{1:T}$—as a linear combination of the R previous samples plus a Gaussian distributed innovation η

$$y_t = \sum_{r=1}^{R} c_r y_{t-r} + \eta_t \quad \text{with} \quad \eta_t \sim \mathcal{N}(0, \sigma^2) \tag{1}$$

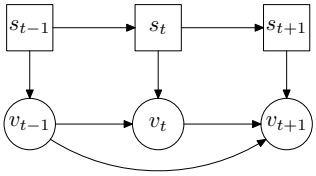


Figure 3: DBN representation of the SAR-HMM. The squares and circles represent discrete and continuous variables respectively—s_t symbolises the state at time t and y_t the observed sample at time t.

where σ^2 is the variance of the innovation and c_r are the AR coefficients. However, an AR process is too simple to model the strong non-stationarities typically encountered in speech signals. A possible way to deal with non-stationarity is to select at each timestep t a setting of the AR parameters from a discrete set of possible parameter values, with the switching between the parameters controlled by a Markov Model. This approach is at the root of the *AR Hidden Markov Model* (AR-HMM) proposed by Poritz [1] and its modern-day counterpart the *Switching AR-HMM* (SAR-HMM), proposed by Ephraim and Roberts [2].

At the heart of the above models lies a standard AR-process. However, a fundamental limitation of such AR models is that the innovation variance σ^2 does not scale properly with the signal. In particular, if the signal is scaled by a factor α, we would expect the innovation variance to scale by a factor α^2 as well. In other words, the 'gain' of the sequence, σ, needs to be set for each sequence, and has a strong impact on the likelihood of an observed sequence. A straightforward solution is to *gain normalise* the signal such that it always has unit variance. An alternate and more effective solution is to replace σ^2 in Equation 1 by the variance which maximises the likelihood of the speech signal $y_{1:T}$

$$\sigma^2_{\mathrm{ML}} = \arg\max_{\sigma^2} p(y_{1:T} \,|\, \sigma^2). \tag{2}$$

This approach, called *Gain Adaptation* (GA), has been successfully used for isolated digit recognition with AR-HMMs in clean and noisy environments [2, 3, 4]. Whilst useful in practice, GA does not fit into the usual machine learning framework since, formally, model parameters may only be set on the basis of *training* data. Otherwise, in flexible models, setting model parameters on the basis of test data may lead to overfitting. We therefore consider a statistically principled alternative Bayesian approach to gain adaptation which consists in specifying a prior probability distribution on the model parameters. This approach has two potential benefits over standard GA(i) the variations of the gain can be explicitly controlled, and (ii) the AR coefficients are allowed to change, which may be useful to model inter and intra speaker variations for example.
Here we present the Bayesian SAR-HMM which generalises the standard acoustic level SAR-HMM, concurrently dealing with the issues of GA and parameter uncertainty in a computationally efficient and principled manner.

2.1 The SAR-HMM

The standard SAR-HMM[2, 3, 4] has a discrete switch variable which can be in S different states, each state representing a particular setting of the AR coefficients c_r and innovation variance σ^2 used in Equation 1. From a probabilistic viewpoint, the model defines a joint

distribution over the sequences of observed samples $y_{1:T}$ and switch states $s_{1:T}$ of the form

$$p(y_{1:T}, s_{1:T}) = \prod_{t=1}^{T} p(y_t \,|\, y_{t-R:t-1}, s_t)\, p(s_t \,|\, s_{t-1}) \tag{3}$$

where $p(y_t \,|\, y_{t-R:t-1}, s_t) \equiv p(y_t \,|\, y_{1:t-1}, s_t)$ if $t \leq R$ and $p(s_1 \,|\, s_0) \equiv p(s_1)$. The emission probability, corresponding to Equation 1, is given by

$$p(y_t \,|\, y_{t-R:t-1}, s_t) \propto \exp\left\{-\frac{1}{2\sigma_{s_t}^2}\big(y_t - \tilde{\mathbf{y}}_t^{\mathsf{T}} \mathbf{c}_{s_t}\big)^2\right\} \tag{4}$$

where $\tilde{\mathbf{y}}_t^{\mathsf{T}} = [y_{t-1} \ldots y_{t-R}]$ and $c_{s_t} = [c_1(s_t) \ldots c_R(s_t)]^{\mathsf{T}}$.
In practice it is not desirable to allow the switch state to change at each time step because we expect the dynamics to last for a minimal amount of time—1.75 ms in our case[1]. In the SAR-HMM, the speech signal is therefore considered as the concatenation of N fixed-length segments over which the state cannot change. This corresponds to the joint distribution

$$p(y_{1:T}, s_{1:N}) = \prod_{n=1}^{N} p(s_n \,|\, s_{n-1}) \prod_{t=t_n}^{t_{n+1}-1} p(y_t \,|\, \tilde{\mathbf{y}}_t, s_n) \tag{5}$$

where t_n is the time step at which the n-th segment starts[2].

Gain Adaptation in the SAR-HMM

Given a sequence of samples $y_{1:T}$, GA is performed in the SAR-HMM by replacing the state innovation variances σ_s^2 in Equation 4 by the *per segment and state* variances σ_{ns}^2 which maximise the likelihood of the observed sequence $y_{1:T}$, i.e.,

$$\sigma_{ns}^2 = \frac{1}{T_n} \sum_{t=t_n}^{t_{n+1}-1} \big(y_t - \tilde{\mathbf{y}}_t \mathbf{c}_s\big)^2$$

where $T_n = t_{n+1} - t_n - 1$ is the length of the n-th segment.

2.2 The Bayesian SAR-HMM

In the SAR-HMM the AR coefficients $\mathbf{c}_s$ and innovation variances σ_s^2 are considered as free parameters that have to be learned from data. In the proposed Bayesian approach we treat them as random variables whose probability distributions are controlled by hyper-parameters. Figure 4 shows the *Dynamical Bayesian Network* (DBN) representation of the Bayesian SAR-HMM. A particular segment n is modelled by an R-th order AR process whose coefficients $\mathbf{c}_n$ and inverse innovation variance[3] ν_n are drawn randomly from a prior distribution conditioned on the switch state s_n. Formally the Bayesian SAR-HMM defines the joint distribution

$$\begin{aligned} &p(y_{1:T}, s_{1:N}, \mathbf{c}_{1:N}, \nu_{1:N}) = \\ &\quad \prod_{n=1}^{N} p(y_n \,|\, \tilde{\mathbf{y}}_{t_n}, \mathbf{c}_n, \nu_n)\, p(\mathbf{c}_n, \nu_n \,|\, s_n)\, p(s_n \,|\, s_{n-1}) \end{aligned} \tag{6}$$

[1] This corresponds to 140 samples at a sampling frequency of 8 kHz.
[2] To save space, we replaced $y_{t-R:t-1}$ by $\tilde{\mathbf{y}}_t$ in Equation 5. Hence $p(y_t \,|\, \tilde{\mathbf{y}}_t, s_t)$ is shorthand for $p(y_t \,|\, y_{t-R:t-1}, s_t)$.
[3] To ease notation we prefer using the inverse variance $\nu = 1/\sigma^2$.

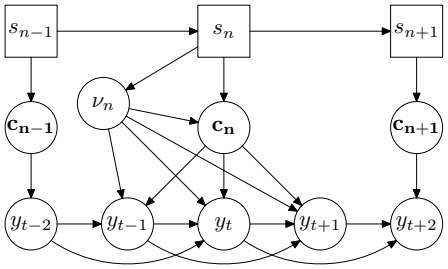


Figure 4: DBN representation of the Bayesian SAR-HMM. The graph represents a model with segments of 3 samples and an AR process of order 2. The index n represents the segment number. Squares and circles represent discrete and continuous variables respectively.

which is a temporal extension of [5]. Explicitly,

$$p(y_n \,|\, \tilde{\mathbf{y}}_{t_n}, \mathbf{c}_n, \nu_n) = \prod_{t=t_n}^{t_{n+1}-1} p(y_t \,|\, \tilde{\mathbf{y}}_t, \mathbf{c}_n, \nu_n). \tag{7}$$

The new factor

$$p(\mathbf{c}_n, \nu_n \,|\, s_n) = p(\mathbf{c}_n \,|\, \nu_n, s_n)\, p(\nu_n \,|\, s_n)$$

defines priors on the AR coefficients and the inverse innovation variance of the n-th segment. In order to keep the model tractable, we use the conjugate priors[4]

$$\mathbf{c} \,|\, \nu, s \sim \mathcal{N}(\mu_s, \nu^{-1}\Sigma_s) \quad \text{and} \quad \nu \,|\, s \sim \gamma(\alpha_s, \beta_s)$$

where $\mathcal{N}(\mu, \Sigma)$ is the multivariate normal distribution with mean μ and covariance Σ, and $\gamma(\alpha, \beta)$ is the gamma distribution defined as

$$\gamma(\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \nu^{\alpha-1} e^{-\beta\nu}.$$

2.3 Training

The free parameters of the Bayesian SAR-HMM are, μ_s, Σ_s, α_s, β_s, for each state s, and the transition probability $a_{ij} \equiv p(s_n = j \,|\, s_{n-1} = i)$ for each pair (i, j) of switch states. Training the model consists of maximising the likelihood of the observed training data

$$p(y_{1:T}) = \sum_{s_{1:N}, \mathbf{c}_{1:N}, \nu_{1:N}} p(y_{1:T}, s_{1:N}, \mathbf{c}_{1:N}, \nu_{1:N}). \tag{8}$$

To achieve this, we use the standard *Expectation Maximisation* (EM) algorithm. Given the current setting of the model parameters ϕ, an updated setting $\hat{\phi}$ is found by maximising (M-step) the expected complete log-likelihood (E-step)

$$\Big\langle \log p\big(y_{1:T}, s_{1:N}, \mathbf{c}_{1:N}, \nu_{1:N} \,|\, \hat{\phi}\big) \Big\rangle_q \tag{9}$$

where $\langle \cdot \rangle_q$ is the average with respect to the posterior

$$q \equiv p(s_{1:N}, \mathbf{c}_{1:N}, \nu_{1:N} \,|\, y_{1:T}, \phi). \tag{10}$$

[4]The segment number has been dropped to simplify the notation.

Model	Word Accuracy
HMM (HTK)	100%
SAR-HMM (no gain)	88.3%
SAR-HHM (gain)	97.2% (98.5%)
Bayesian SAR-HMM	98.4%

Table 1: Word accuracy of three different models on a single digit recognition task on the TI-DIGITS database; *gain* and *no gain* indicates whether or not gain adaptation has been used. The performance of the gain adapted SAR-HMM reported in [2] is indicated between parenthesis.

2.4 Performance

We compared the Bayesian SAR-HMM to the original SAR-HMM proposed in [2], with and without gain adaptation, and a standard feature-based HMM. The task was to recognise isolated digits pronounced by various male speakers from the TI-DIGITS database [6]. The training/test sets were composed of 110/112 utterances for each of the eleven digits (1–9, 'zero' and 'oh'), spoken by 55/56 different speakers respectively. Each digit class was modelled by a separate SAR-HMM and recognition performed by associating the utterance to the digit whose model had the highest likelihood. Whilst this speech classification problem is relatively easy, the effective volume on each utterance is different so that, for AR-based models, some form of GA is crucial for good performance.

All SAR-HMMs were composed of 10 states with a left-right transition matrix and a 10-th order AR process per state, in keeping with the optimal values found in [2]. The Bayesian SAR-HMM was initialised with hyperparameter $\alpha_s = 10$. The transition matrix of Bayesian SAR-HMM was set to the trained standard SAR-HMM and, for each state, the priors on the AR coefficients and the inverse variance were set such that their means (α/β) corresponds to the AR coefficients and inverse variance obtained from the trained SAR-HMM. The covariance matrix Σ_s was initialised to $\frac{1}{\langle \nu_s \rangle} I$, where I is the identity matrix.

The feature-based HMM was composed of 18 states with a left-right transition matrix, a mixture of three Gaussians per state and used 13 MFCC features, including energy and implemented using HTK [7].

Table 1 shows the word accuracy of each model. The performance of the gain adapted SAR-HMM is reproduced from [2]. All the other results have been obtained by our own implementation of the respective models. Note that the accuracy we obtained for the gain adapted SAR-HMM is slightly below that reported in [2]. The Bayesian and gain adapted SAR-HMM have a word accuracy which is 10% higher than that of the non gain adapted SAR-HMM. This demonstrates that dealing with the gain problem is crucial to ensure good performance. The similar performance of the Bayesian and gain adapted SAR-HMM demonstrates that the Bayesian approach is a sensible principled alternative to gain adaptation.

2.5 Discussion

Modeling the raw acoustic signal is an alternative strategy to using feature based HMMs for speech recognition. A motivation for this is that strong signal models may be used to remove noise, and can also form the basis of powerful hierarchical models of the signal. However,

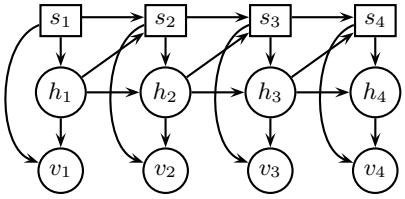


Figure 5: The independence structure of the aSLDS. Square nodes denote discrete variables, round nodes continuous variables. In the SLDS links from h to s are not normally considered.

signal models based on AR-processes are over-sensitive to signal amplitude, and this problem is typically healed using ad-hoc Gain Adaptation. In contrast, our Bayesian approach provides a statistically principled and straightforward exact alternative to standard Maximum Likelihood Gain Adaptation. The result is a simple update formula which correctly deals with the uncertainty in the parameter estimates from the training set, and automatically computes the posterior distribution of parameters in light of test data. This is an encouraging step towards the development of more complex signal and noise models, in which the flexibility of the models is every increasing.

3 Switching Linear Dynamical Systems

The Linear Dynamical System (LDS) [8] is a key temporal model in which a latent linear process generates the observed series. For complex time-series which are not well described globally by a single LDS, we may break the time-series into segments, each modeled by a potentially different LDS. This is the basis for the Switching LDS (SLDS) [9, 10, 11, 12] where, for each time t, a switch variable $s_t \in 1, \ldots, S$ describes which of the LDSs is to be used. The observation (or 'visible') $v_t \in \mathcal{R}^V$ is linearly related to the hidden state $h_t \in \mathcal{R}^H$ with additive noise η by

$$v_t = B(s_t)h_t + \eta^v(s_t) \quad \equiv \quad p(v_t|h_t, s_t) = \mathcal{N}\left(B(s_t)h_t, \Sigma^v(s_t)\right) \tag{11}$$

where $\mathcal{N}(\mu, \Sigma)$ denotes a Gaussian distribution with mean μ and covariance Σ. The transition dynamics of the continuous hidden state h_t is linear,

$$h_t = A(s_t)h_{t-1} + \eta^h(s_t), \quad \equiv \quad p(h_t|h_{t-1}, s_t) = \mathcal{N}\left(A(s_t)h_{t-1}, \Sigma^h(s_t)\right) \tag{12}$$

The switch s_t may depend on both the previous s_{t-1} and h_{t-1}. This is an augmented SLDS (aSLDS), and defines the model

$$p(v_{1:T}, h_{1:T}, s_{1:T}) = \prod_{t=1}^{T} p(v_t|h_t, s_t)p(h_t|h_{t-1}, s_t)p(s_t|h_{t-1}, s_{t-1})$$

The standard SLDS[11] considers only switch transitions $p(s_t|s_{t-1})$. At time $t = 1$, $p(s_1|h_0, s_0)$ simply denotes the prior $p(s_1)$, and $p(h_1|h_0, s_1)$ denotes $p(h_1|s_1)$. The aim of this article is to address how to perform inference in the aSLDS. In particular we desire the *filtered* estimate $p(h_t, s_t|v_{1:t})$ and the *smoothed* estimate $p(h_t, s_t|v_{1:T})$, for any $1 \leq t \leq T$. Both filtered and smoothed inference in the SLDS is intractable, scaling exponentially with time [11].

3.1 Expectation Correction

Our approach to approximate $p(h_t, s_t|v_{1:T})$ mirrors the Rauch-Tung-Striebel 'correction' smoother for the simpler LDS [8].The method consists of a single forward pass to recursively find the filtered posterior $p(h_t, s_t|v_{1:t})$, followed by a single backward pass to correct this into a smoothed

posterior $p(h_t, s_t|v_{1:T})$. The forward pass we use is equivalent to standard Assumed Density Filtering (ADF) [13]. The main contribution of this work is a novel form of backward pass, based only on collapsing the smoothed posterior to a mixture of Gaussians. Together with the ADF forward pass, we call the method Expectation Correction, since it corrects the moments found from the forward pass.

3.1.1 *Forward Pass (Filtering)*

Readers familiar with ADF may wish to continue directly to Section (3.1.2). Our aim is to form a recursion for $p(s_t, h_t|v_{1:t})$, based on a Gaussian mixture approximation of $p(h_t|s_t, v_{1:t})$. Without loss of generality, we may decompose the filtered posterior as

$$p(h_t, s_t|v_{1:t}) = p(h_t|s_t, v_{1:t})p(s_t|v_{1:t}) \tag{13}$$

The exact representation of $p(h_t|s_t, v_{1:t})$ is a mixture with $O(S^t)$ components. We therefore approximate this with a smaller I-component mixture

$$p(h_t|s_t, v_{1:t}) \approx \sum_{i_t=1}^{I} p(h_t|i_t, s_t, v_{1:t})p(i_t|s_t, v_{1:t})$$

where $p(h_t|i_t, s_t, v_{1:t})$ is a Gaussian parameterized with mean $f(i_t, s_t)$ and covariance $F(i_t, s_t)$. To find a recursion for these parameters, consider

$$p(h_{t+1}|s_{t+1}, v_{1:t+1}) = \sum_{s_t, i_t} p(h_{t+1}|s_t, i_t, s_{t+1}, v_{1:t+1})p(s_t, i_t|s_{t+1}, v_{1:t+1}) \tag{14}$$

Evaluating $p(h_{t+1}|s_t, i_t, s_{t+1}, v_{1:t+1})$

We find $p(h_{t+1}|s_t, i_t, s_{t+1}, v_{1:t+1})$ by first computing the joint distribution $p(h_{t+1}, v_{t+1}|s_t, i_t, s_{t+1}, v_{1:t})$, which is a Gaussian with covariance and mean elements,

$$\begin{aligned}
&\Sigma_{hh} = A(s_{t+1})F(i_t, s_t)A^{\mathsf{T}}(s_{t+1}) + \Sigma^h(s_{t+1}), \quad \Sigma_{vv} = B(s_{t+1})\Sigma_{hh}B^{\mathsf{T}}(s_{t+1}) + \Sigma^v(s_{t+1}) \\
&\Sigma_{vh} = B(s_{t+1})F(i_t, s_t), \quad \mu_v = B(s_{t+1})A(s_{t+1})f(i_t, s_t), \quad \mu_h = A(s_{t+1})f(i_t, s_t)
\end{aligned} \tag{15}$$

and then conditioning on v_{t+1}[5]. For the case $S = 1$, this forms the usual Kalman Filter recursions[8].

Evaluating $p(s_t, i_t|s_{t+1}, v_{1:t+1})$

The mixture weight in Equation (14) can be found from the decomposition

$$p(s_t, i_t|s_{t+1}, v_{1:t+1}) \propto p(v_{t+1}|i_t, s_t, s_{t+1}, v_{1:t})p(s_{t+1}|i_t, s_t, v_{1:t})p(i_t|s_t, v_{1:t})p(s_t|v_{1:t}) \tag{16}$$

The first factor in Equation (16), $p(v_{t+1}|i_t, s_t, s_{t+1}, v_{1:t})$ is a Gaussian with mean μ_v and covariance Σ_{vv}, as given in Equation (15). The last two factors $p(i_t|s_t, v_{1:t})$ and $p(s_t|v_{1:t})$ are given from the previous iteration. Finally, $p(s_{t+1}|i_t, s_t, v_{1:t})$ is found from

$$p(s_{t+1}|i_t, s_t, v_{1:t}) = \langle p(s_{t+1}|h_t, s_t)\rangle_{p(h_t|i_t, s_t, v_{1:t})} \tag{17}$$

where $\langle\cdot\rangle_p$ denotes expectation with respect to p. In the SLDS, Equation (17) is replaced by the Markov transition $p(s_{t+1}|s_t)$. In the aSLDS, however, Equation (17) will generally need to be computed numerically.

[5] $p(x|y)$ is a Gaussian with mean $\mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y)$ and covariance $\Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx}$.

Closing the recursion

We are now in a position to calculate Equation (14). For each setting of the variable s_{t+1}, we have a mixture of $I \times S$ Gaussians which we numerically collapse back to I Gaussians to form

$$p(h_{t+1}|s_{t+1}, v_{1:t+1}) \approx \sum_{i_{t+1}=1}^{I} p(h_{t+1}|i_{t+1}, s_{t+1}, v_{1:t+1}) p(i_{t+1}|s_{t+1}, v_{1:t+1})$$

Any method of choice may be supplied to collapse a mixture to a smaller mixture; our code simply repeatedly merges low-weight components. In this way the new mixture coefficients $p(i_{t+1}|s_{t+1}, v_{1:t+1})$, $i_{t+1} \in 1, \ldots, I$ are defined, completing the description of how to form a recursion for $p(h_{t+1}|s_{t+1}, v_{1:t+1})$ in Equation (13). A recursion for the switch variable is given by

$$p(s_{t+1}|v_{1:t+1}) \propto \sum_{s_t, i_t} p(v_{t+1}|s_{t+1}, i_t, s_t, v_{1:t}) p(s_{t+1}|i_t, s_t, v_{1:t}) p(i_t|s_t, v_{1:t}) p(s_t|v_{1:t})$$

where all terms have been computed during the recursion for $p(h_{t+1}|s_{t+1}, v_{1:t+1})$.
The likelihood $p(v_{1:T})$ may be found by recursing $p(v_{1:t+1}) = p(v_{t+1}|v_{1:t}) p(v_{1:t})$, where

$$p(v_{t+1}|v_t) = \sum_{i_t, s_t, s_{t+1}} p(v_{t+1}|i_t, s_t, s_{t+1}, v_{1:t}) p(s_{t+1}|i_t, s_t, v_{1:t}) p(i_t|s_t, v_{1:t}) p(s_t|v_{1:t})$$

3.1.2 *Backward Pass (Smoothing)*

The main contribution of our work is to find a suitable way to 'correct' the filtered posterior $p(s_t, h_t|v_{1:t})$ obtained from the forward pass into a smoothed posterior $p(s_t, h_t|v_{1:T})$. We derive this for the case of a single Gaussian representation. We approximate the smoothed posterior $p(h_t|s_t, v_{1:T})$ by a Gaussian with mean $g(s_t)$ and covariance $G(s_t)$ and our aim is to find a recursion for these parameters. A useful starting point for a recursion is:

$$p(h_t, s_t|v_{1:T}) = \sum_{s_{t+1}} p(s_{t+1}|v_{1:T}) p(h_t|s_t, s_{t+1}, v_{1:T}) p(s_t|s_{t+1}, v_{1:T})$$

The term $p(h_t|s_t, s_{t+1}, v_{1:T})$ may be computed as

$$p(h_t|s_t, s_{t+1}, v_{1:T}) = \int_{h_{t+1}} p(h_t|h_{t+1}, s_t, s_{t+1}, v_{1:t}) p(h_{t+1}|s_t, s_{t+1}, v_{1:T}) \tag{18}$$

The recursion therefore requires $p(h_{t+1}|s_t, s_{t+1}, v_{1:T})$, which we can write as

$$p(h_{t+1}|s_t, s_{t+1}, v_{1:T}) \propto p(h_{t+1}|s_{t+1}, v_{1:T}) p(s_t|s_{t+1}, h_{t+1}, v_{1:t}) \tag{19}$$

The difficulty here is that the functional form of $p(s_t|s_{t+1}, h_{t+1}, v_{1:t})$ is not squared exponential in h_{t+1}, so that $p(h_{t+1}|s_t, s_{t+1}, v_{1:T})$ will not be Gaussian[6]. One possibility would be to approximate the non-Gaussian $p(h_{t+1}|s_t, s_{t+1}, v_{1:T})$ by a Gaussian (or mixture thereof) by minimizing the Kullback-Leilbler divergence between the two, or performing moment matching in the case of a single Gaussian. A simpler alternative (which forms 'standard' EC) is to make the assumption $p(h_{t+1}|s_t, s_{t+1}, v_{1:T}) \approx p(h_{t+1}|s_{t+1}, v_{1:T})$, where $p(h_{t+1}|s_{t+1}, v_{1:T})$ is already known from the previous backward recursion. Under this assumption, the recursion becomes

$$p(h_t, s_t|v_{1:T}) \approx \sum_{s_{t+1}} p(s_{t+1}|v_{1:T}) p(s_t|s_{t+1}, v_{1:T}) \left\langle p(h_t|h_{t+1}, s_t, s_{t+1}, v_{1:t}) \right\rangle_{p(h_{t+1}|s_{t+1}, v_{1:T})} \tag{20}$$

[6] In the *exact* calculation, $p(h_{t+1}|s_t, s_{t+1}, v_{1:T})$ *is* a mixture of Gaussians. However, since in Equation (19) the two terms $p(h_{t+1}|s_{t+1}, v_{1:T})$ will only be approximately computed during the recursion, our approximation to $p(h_{t+1}|s_t, s_{t+1}, v_{1:T})$ will not be a mixture of Gaussians.

Evaluating $\langle p(h_t|h_{t+1}, s_t, s_{t+1}, v_{1:t})\rangle_{p(h_{t+1}|s_{t+1},v_{1:T})}$

$\langle p(h_t|h_{t+1}, s_t, s_{t+1}, v_{1:t})\rangle_{p(h_{t+1}|s_{t+1},v_{1:T})}$ is a Gaussian in h_t, whose statistics we will now compute. First we find $p(h_t|h_{t+1}, s_t, s_{t+1}, v_{1:t})$ which may be obtained from the joint distribution

$$p(h_t, h_{t+1}|s_t, s_{t+1}, v_{1:t}) = p(h_{t+1}|h_t, s_{t+1})p(h_t|s_t, v_{1:t}) \tag{21}$$

which itself can be found from a forward dynamics from the filtered estimate $p(h_t|s_t, v_{1:t})$. The statistics for the marginal $p(h_t|s_t, s_{t+1}, v_{1:t})$ are simply those of $p(h_t|s_t, v_{1:t})$, since s_{t+1} carries no extra information about h_t. The remaining statistics are the mean of h_{t+1}, the covariance of h_{t+1} and cross-variance between h_t and h_{t+1}, which are given by

$$\langle h_{t+1}\rangle = A(s_{t+1})f_t(s_t),\ \Sigma_{t+1,t+1} = A(s_{t+1})F_t(s_t)A^{\mathsf{T}}(s_{t+1}) + \Sigma^h(s_{t+1}),\ \Sigma_{t+1,t} = A(s_{t+1})F_t(s_t)$$

Given the statistics of Equation (21), we may now condition on h_{t+1} to find $p(h_t|h_{t+1}, s_t, s_{t+1}, v_{1:t})$. Doing so effectively constitutes a reversal of the dynamics,

$$h_t = \overleftarrow{A}(s_t, s_{t+1})h_{t+1} + \overleftarrow{\eta}(s_t, s_{t+1})$$

where $\overleftarrow{A}(s_t, s_{t+1})$ and $\overleftarrow{\eta}(s_t, s_{t+1}) \sim \mathcal{N}(\overleftarrow{m}(s_t, s_{t+1}), \overleftarrow{\Sigma}(s_t, s_{t+1}))$ are easily found using conditioning. Averaging the above reversed dynamics over $p(h_{t+1}|s_{t+1}, v_{1:T})$, we find that $\langle p(h_t|h_{t+1}, s_t, s_{t+1}, v_{1:t})\rangle_{p(h_{t+1}}$ is a Gaussian with statistics

$$\mu_t = \overleftarrow{A}(s_t, s_{t+1})g(s_{t+1}) + \overleftarrow{m}(s_t, s_{t+1}),\quad \Sigma_{t,t} = \overleftarrow{A}(s_t, s_{t+1})G(s_{t+1})\overleftarrow{A}^{\mathsf{T}}(s_t, s_{t+1}) + \overleftarrow{\Sigma}(s_t, s_{t+1})$$

These equations directly mirror the standard RTS backward pass[8].

Evaluating $p(s_t|s_{t+1}, v_{1:T})$

The main departure of EC from previous methods is in treating the term

$$p(s_t|s_{t+1}, v_{1:T}) = \langle p(s_t|h_{t+1}, s_{t+1}, v_{1:t})\rangle_{p(h_{t+1}|s_{t+1},v_{1:T})} \tag{22}$$

The term $p(s_t|h_{t+1}, s_{t+1}, v_{1:t})$ is given by

$$p(s_t|h_{t+1}, s_{t+1}, v_{1:t}) = \frac{p(h_{t+1}|s_{t+1}, s_t, v_{1:t})p(s_t, s_{t+1}|v_{1:t})}{\sum_{s'_t} p(h_{t+1}|s_{t+1}, s'_t, v_{1:t})p(s'_t, s_{t+1}|v_{1:t})} \tag{23}$$

Here $p(s_t, s_{t+1}|v_{1:t}) = p(s_{t+1}|s_t, v_{1:t})p(s_t|v_{1:t})$, where $p(s_{t+1}|s_t, v_{1:t})$ occurs in the forward pass, Equation (17). In Equation (23), $p(h_{t+1}|s_{t+1}, s_t, v_{1:t})$ is found by marginalizing Equation (21).

Computing the average of Equation (23) with respect to $p(h_{t+1}|s_{t+1}, v_{1:T})$ may be achieved by any numerical integration method desired. A simple approximation is to evaluate the integrand at the mean value of the averaging distribution $p(h_{t+1}|s_{t+1}, v_{1:T})$.

Closing the Recursion

We have now computed both the continuous and discrete factors in Equation (18), which we wish to use to write the smoothed estimate in the form $p(h_t, s_t|v_{1:T}) = p(s_t|v_{1:T})p(h_t|s_t, v_{1:T})$. The distribution $p(h_t|s_t, v_{1:T})$ is readily obtained from the joint Equation (18) by conditioning on s_t to form the mixture

$$p(h_t|s_t, v_{1:T}) = \sum_{s_{t+1}} p(s_{t+1}|s_t, v_{1:T})p(h_t|s_t, s_{t+1}, v_{1:T})$$

which may then be collapsed to a single Gaussian (the mixture case is discussed in the published version of this work). The smoothed posterior $p(s_t|v_{1:T})$ is given by

$$p(s_t|v_{1:T}) = \sum_{s_{t+1}} p(s_{t+1}|v_{1:T})\,\langle p(s_t|h_{t+1}, s_{t+1}, v_{1:t})\rangle_{p(h_{t+1}|s_{t+1},v_{1:T})}\,. \tag{24}$$

3.1.3 *Relation to other methods*

The EC Backward pass is closely related to Kim's method [14]. In both EC and Kim's method, the approximation $p(h_{t+1}|s_t, s_{t+1}, v_{1:T}) \approx p(h_{t+1}|s_{t+1}, v_{1:T})$, is used to form a numerically simple backward pass. The other 'approximation' in EC is to numerically compute the average in Equation (24). In Kim's method, however, an update for the discrete variables is formed by replacing the required term in Equation (24) by

$$\langle p(s_t|h_{t+1}, s_{t+1}, v_{1:t})\rangle_{p(h_{t+1}|s_{t+1},v_{1:T})} \approx p(s_t|s_{t+1}, v_{1:t}) \tag{25}$$

Since $p(s_t|s_{t+1}, v_{1:t}) \propto p(s_{t+1}|s_t)p(s_t|v_{1:t})/p(s_{t+1}|v_{1:t})$, this can be computed simply from the filtered results alone. The fundamental difference therefore between EC and Kim's method is that the approximation, Equation (25), is not required by EC. The EC backward pass therefore makes fuller use of the future information, resulting in a recursion which intimately couples the continuous and discrete variables. The resulting effect on the quality of the approximation can be profound, as we will see in the experiments.

The Expectation Propagation (EP) algorithm makes the central assumption of collapsing the posteriors to a Gaussian family [12]; the collapse is defined by a consistency criterion on overlapping marginals. In our experiments, we take the approach in [15] of collapsing to a single Gaussian. Ensuring consistency requires frequent translations between moment and canonical parameterizations, which is the origin of potentially severe numerical instability [16]. In contrast, EC works largely with moment parameterizations of Gaussians, for which relatively few numerical difficulties arise. Unlike EP, EC is not based on a consistency criterion and a subtle issue arises about possible inconsistencies in the Forward and Backward approximations for EC. For example, under the conditional independence assumption in the Backward Pass, $p(h_T|s_{T-1}, s_T, v_{1:T}) \approx p(h_T|s_T, v_{1:T})$, which is in contradiction to Equation (15) which states that the approximation to $p(h_T|s_{T-1}, s_T, v_{1:T})$ *will* depend on s_{T-1}. Such potential inconsistencies arise because of the approximations made, and should not be considered as separate approximations in themselves.

Rather than using a global (consistency) objective, EC attempts to faithfully approximate the exact Forward and Backward propagation routines. For this reason, as in the exact computation, only a single Forward and Backward pass are required in EC.

In [17] a related dynamics reversed is proposed. However, the singularities resulting from incorrectly treating $p(v_{t+1:T}|h_t, s_t)$ as a density are heuristically finessed.

In [18] a variational method approximates the joint distribution $p(h_{1:T}, s_{1:T}|v_{1:T})$ rather than the marginal inference $p(h_t, s_t|v_{1:T})$. This is a disadvantage when compared to other methods that directly approximate the marginal.

Sequential Monte Carlo methods (Particle Filters)[19], are essentially mixture of delta-function approximations. Whilst potentially powerful, these typically suffer in high-dimensional hidden spaces, unless techniques such as Rao-Blackwellization are performed. ADF is generally preferential to Particle Filtering since in ADF the approximation is a mixture of non-trivial distributions, and is therefore more able to represent the posterior.

3.2 Application to Noise Robust ASR

Here we briefly present an application of the SLDS to robust Automatic Speech Recognition (ASR), for which the intractable inference is performed by EC, and serves to demonstrate

how EC scales well to a large-scale application. Fuller details are given in [20]. The standard approach to noise robust ASR is to provide a set of noise-robust features to a standard Hidden Markov Model (HMM) classifier, which is based on modeling the acoustic feature vector. For example, the method of Unsupervised Spectral Subtraction (USS) [21] provides state-of-the-art performance in this respect. Incorporating noise models directly into such feature-based HMM systems is difficult, mainly because the explicit influence of the noise on the features is poorly understood. An alternative is to model the raw speech signal directly, such as the SAR-HMM model [2] for which, under *clean* conditions, isolated spoken digit recognition performs well. However, the SAR-HMM performs poorly under noisy conditions, since no explicit noise processes are taken into account by the model.

The approach we take here is to extend the SAR-HMM to include an explicit noise process, so that the observed signal v_t is modeled as a noise corrupted version of a clean *hidden* signal v_t^h:

$$v_t = v_t^h + \tilde{\eta}_t \quad \text{with} \quad \tilde{\eta}_t \sim \mathcal{N}(0, \tilde{\sigma}^2)$$

The dynamics of the clean signal is modeled by a switching AR process

$$v_t^h = \sum_{r=1}^{R} c_r(s_t) v_{t-r}^h + \eta_t^h(s_t), \qquad \eta_t^h(s_t) \sim \mathcal{N}(0, \sigma^2(s_t))$$

where $s_t \in \{1, \ldots, S\}$ denotes which of a set of AR coefficients $c_r(s_t)$ are to be used at time t, and $\eta_t^h(s_t)$ is the so-called *innovation* noise. When $\sigma^2(s_t) \equiv 0$, this model reproduces the SAR-HMM of [2], a specially constrained HMM. Hence inference and learning for the SAR-HMM are tractable and straightforward. For the case $\sigma^2(s_t) > 0$ the model can be recast as an SLDS. To do this we define h_t as a vector which contains the R most recent clean hidden samples

$$h_t = \begin{bmatrix} v_t^h & \ldots & v_{t-r+1}^h \end{bmatrix}^\mathsf{T} \tag{26}$$

and we set $A(s_t)$ to be an $R \times R$ matrix where the first row contains the AR coefficients $-c_r(s_t)$ and the rest is a shifted down identity matrix. For example, for a third order ($R = 3$) AR process,

$$A(s_t) = \begin{bmatrix} -c_1(s_t) & -c_2(s_t) & -c_3(s_t) \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \tag{27}$$

The hidden covariance matrix $\Sigma_h(s)$ has all elements zero, except the top-left most which is set to the innovation variance. To extract the first component of h_t we use the (switch independent) $1 \times R$ projection matrix $B = \begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix}$. The (switch independent) visible scalar noise variance is given by $\Sigma_v \equiv \sigma_v^2$.

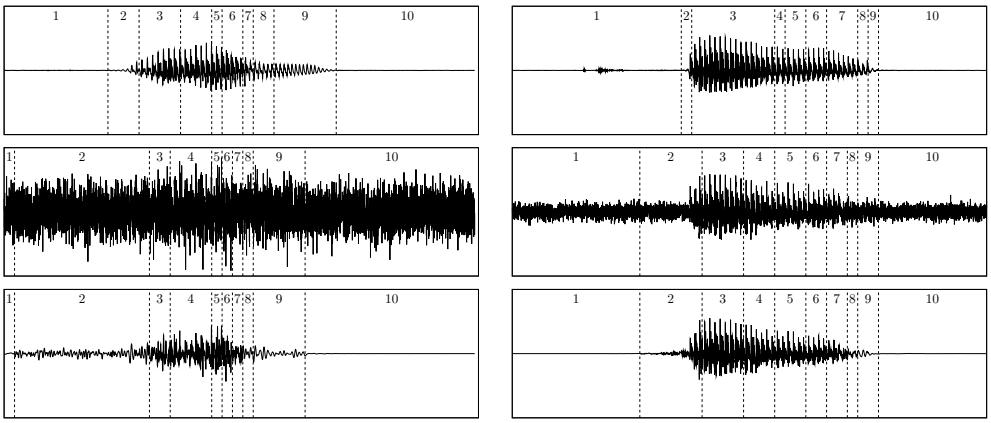
A well-known issue with raw speech signal models is that the energy of a signal may vary from one speaker to another or because of a change in recording conditions. For this reason the innovation Σ_h is adjusted by maximizing the likelihood of an observed sequence with respect to the innovation covariance, a process called *Gain Adaptation* [2].

3.2.1 *Training & Evaluation*

Following [2], we trained a separate SAR-HMM for each of the eleven digits (0–9 and 'oh') from the TI-DIGITS database [6]. The training set for each digit was composed of 110 single digit

Noise Variance	SNR (dB)	HMM	SAR-HMM	AR-SLDS
0	26.5	100.0%	97.0%	96.8%
10^{-7}	26.3	100.0%	79.8%	96.8%
10^{-6}	25.1	90.9%	56.7%	96.4%
10^{-5}	19.7	86.4%	22.2%	94.8%
10^{-4}	10.6	59.1%	9.7%	84.0%
10^{-3}	0.7	9.1%	9.1%	61.2%

Table 2: Comparison of the recognition accuracy of three models when the test utterances are corrupted by various levels of Gaussian noise.



"one" at SNR 0.7 dB "five" at SNR 10.6 dB

Figure 6: Two examples of signal reconstruction using the AR-SLDS; (top) original clean signal taken from the TI-DIGITS database, (middle) noisy signal, i.e., clean signal artificially corrupted by Gaussian noise, (bottom) reconstructed clean signal. The dashed lines and the numbers show the most-likely state segmentation.

utterances down-sampled to 8 kHz, each one pronounced by a male speaker. Each SAR-HMM was composed of ten states with a left-right transition matrix. Each state was associated with a 10th-order AR process and the model was constrained to stay an integer multiple of $K = 140$ time steps (0.0175 seconds) in the same state. We refer the reader to [2] for a detailed explanation of the training procedure used with the SAR-HMM.

An AR-SLDS was built for each of the eleven digits by copying the parameters of the corresponding trained SAR-HMM, i.e., the AR coefficients $c_r(s)$ are copied into the first row of the hidden transition matrix $A(s)$ and the same discrete transition distribution $p(s_t \mid s_{t-1})$ is used. The models were then evaluated on a test set composed of 112 corrupted utterances of each of the eleven digits, each pronounced by different male speakers than those used in the training set. The recognition accuracy obtained by the models on the corrupted test sets is presented in Table 2. As expected, the performance of the SAR-HMM rapidly decreases with noise. The feature-based HMM with USS has high accuracy only for high SNR levels. In contrast, the AR-SLDS achieves a recognition accuracy of 61.2% at a SNR close to 0 dB, while the performance of the two other methods is equivalent to random guessing (9.1%). Whilst other inference methods may also perform well in this case, we found that EC performs admirably, without numerical instabilities, even for time-series with several thousand time-steps.

3.3 Discussion

We presented a method for approximate smoothed inference in an augmented class of switching linear dynamical systems. Our approximation is based on the idea that due to the forgetting which commonly occurs in Markovian models, a finite number of mixture components may provide a reasonable approximation. Clearly, in systems with very long correlation times our method may require too many mixture components to produce a satisfactory result, although we are unaware of other techniques that would be able to cope well in that case. The main benefit of EC over Kim smoothing is that future information is more accurately dealt with. Whilst EC is not as general as EP, EC carefully exploits the properties of singly-connected distributions, such as the aSLDS, to provide a numerically stable procedure. We hope that the ideas presented here may therefore help facilitate the practical application of dynamic hybrid networks.

4 Bayesian Linear Gaussian State-Space Models

Linear Gaussian State-Space Models (LGSSMs)[7] are fundamental in time-series analysis [22, 23, 24]. In these models the observations $v_{1:T}$[8] are generated from an underlying dynamical system on $h_{1:T}$ according to

$$v_t = Bh_t + \eta_t^v, \;\; \eta_t^v \sim \mathcal{N}(\mathbf{0}_V, \Sigma_V); \qquad h_t = Ah_{t-1} + \eta_t^h, \;\; \eta_t^h \sim \mathcal{N}\left(\mathbf{0}_H, \Sigma_H\right) ,$$

where $\mathcal{N}(\mu, \Sigma)$ denotes a Gaussian with mean μ and covariance Σ, and $\mathbf{0}_X$ denotes an X-dimensional zero vector. The observation v_t has dimension V and the hidden state h_t dimension H. Probabilistically, the LGSSM is defined by:

$$p(v_{1:T}, h_{1:T}|\Theta) = p(v_1|h_1)p(h_1)\prod_{t=2}^{T} p(v_t|h_t)p(h_t|h_{t-1}) ,$$

with $p(v_t|h_t) = \mathcal{N}\left(Bh_t, \Sigma_V\right)$, $p(h_t|h_{t-1}) = \mathcal{N}\left(Ah_{t-1}, \Sigma_H\right)$, $p(h_1) = \mathcal{N}(\mu, \Sigma)$ and where $\Theta = \{A, B, \Sigma_H, \Sigma_V, \mu, \Sigma\}$ denotes the model parameters. Because of the widespread use of these models, a Bayesian treatment of parameters is of considerable interest [25, 26, 27, 28, 29].
An exact implementation of the Bayesian LGSSM is formally intractable [29], and recently a Variational Bayesian (VB) approximation has been studied [25, 26, 27, 28, 30]. The most challenging part of implementing the VB method is performing inference over $h_{1:T}$, and previous authors have developed their own specialized routines, based on Belief Propagation, since standard LGSSM inference routines appear, at first sight, not to be applicable.

A key contribution of this work is to show how the Variational Bayesian treatment of the LGSSM *can* be implemented using standard inference routines. Based on the insight we provide, any standard inference method may be applied, including those specifically addressed to improve numerical stability [31, 32, 23]. In this article, we decided to describe the standard predictor-corrector and Rauch-Tung-Striebel recursions [23], and also suggest a small modification that reduces computational cost.

The Bayesian LGSSM is particularly of interest when strong prior constraints are needed to find adequate solutions. One such case is in EEG signal analysis, whereby we wish to extract sources that evolve independently through time. Since EEG is particularly noisy [33], a prior that encourages sources to have preferential spectral properties is advantageous in recovering

[7] Also called Kalman Filters/Smoothers, Linear Dynamical Systems.
[8] $v_{1:T}$ denotes $v_1, \ldots, v_T$.

meaningful sources. This application is discussed in Section (4.3), and demonstrates the ease of applying our VB framework.

4.1 A Bayesian treatment

In the Bayesian treatment of the LGSSM, instead of considering the model parameters Θ as fixed, we define a prior distribution $p(\Theta|\hat{\Theta})$, where $\hat{\Theta}$ is a set of hyperparameters. Then:

$$p(v_{1:T}|\hat{\Theta}) = \int_{\Theta} p(v_{1:T}|\hat{\Theta}, \Theta) p(\Theta|\hat{\Theta}) \,. \tag{28}$$

In a full Bayesian treatment we would define additional prior distributions over the hyperparameters $\hat{\Theta}$. Here we take instead the ML-II ('evidence') framework, in which the optimal set of hyperparameters is found by maximizing $p(v_{1:T}|\hat{\Theta})$ with respect to $\hat{\Theta}$ [27, 28, 30].

For the parameter priors, we define Gaussians on the columns of A and B:

$$p(A|\alpha, \Sigma_H) \propto \prod_{j=1}^{H} e^{-\frac{\alpha_j}{2}(A_j - \hat{A}_j)^{\mathsf{T}} \Sigma_H^{-1} (A_j - \hat{A}_j)}, \qquad p(B|\beta, \Sigma_V) \propto \prod_{j=1}^{H} e^{-\frac{\beta_j}{2}(B_j - \hat{B}_j)^{\mathsf{T}} \Sigma_V^{-1} (B_j - \hat{B}_j)} \,,$$

which has the effect of biasing the transition and emission matrices to desired forms $\hat{A}$ and $\hat{B}$. The conjugate priors for the covariances Σ_H and Σ_V are Inverse Wishart distributions [28][9]. In the simpler and more common case of assuming diagonal covariances these become Inverse Gamma distributions [28, 26]. The hyperparameters are then $\hat{\Theta} = \{\alpha, \beta\}$[10].

Variational Bayes

Optimizing Equation (28) with respect to $\hat{\Theta}$ is difficult due to the intractability of the integrals. Instead, in VB, one considers the lower bound [27, 28, 30][11]:

$$\mathcal{L} = \log p(v_{1:T}|\hat{\Theta}) \geq H_q(\Theta, h_{1:T}) + \left\langle \log p(\Theta|\hat{\Theta}) \right\rangle_{q(\Theta)} + \left\langle E(h_{1:T}, \Theta|\hat{\Theta}) \right\rangle_{q(\Theta, h_{1:T})} \equiv \mathcal{F} \,,$$

where

$$E(h_{1:T}, \Theta|\hat{\Theta}) \equiv \log p(v_{1:T}, h_{1:T}|\Theta, \hat{\Theta}).$$

The notation $H_d(x)$ signifies the entropy of the distribution $d(x)$, and $\langle \cdot \rangle_{d(x)}$ denotes the expectation operator.

The key approximation in VB is $q(\Theta, h_{1:T}) \equiv q(\Theta) q(h_{1:T})$, from which one may show that, for optimality of $\mathcal{F}$,

$$q(h_{1:T}) \propto e^{\left\langle E(h_{1:T}, \Theta|\hat{\Theta}) \right\rangle_{q(\Theta)}}, \qquad q(\Theta) \propto p(\Theta) e^{\left\langle E(h_{1:T}, \Theta|\hat{\Theta}) \right\rangle_{q(h_{1:T})}}.$$

These coupled equations need to be iterated to convergence. The updates for the parameters $q(\Theta)$ are straightforward and are given in Appendices **??** and **??**. Once converged, the hyperparameters are updated by maximizing $\mathcal{F}$ with respect to $\hat{\Theta}$, which lead to simple update formulae [28].

Our main concern is with the update for $q(h_{1:T})$, for which this work makes a departure from treatments previously presented.

[9] For expositional simplicity, we do not put priors on μ and Σ.

[10] For simplicity, we keep the parameters of the Inverse Wishart priors fixed.

[11] Strictly we should write throughout $q(\cdot|v_{1:T})$. We omit the dependence on $v_{1:T}$ for notational convenience.

4.2 Unified Inference on $q(h_{1:T})$

Optimally $q(h_{1:T})$ is Gaussian since $\left\langle E(h_{1:T}, \Theta|\hat{\Theta}) \right\rangle_{q(\Theta)}$ is quadratic in $h_{1:T}$, being namely[12]

$$-\frac{1}{2}\sum_{t=1}^{T}\left[\left\langle (v_t - Bh_t)^\mathsf{T}\Sigma_V^{-1}(v_t - Bh_t)\right\rangle_{q(B,\Sigma_V)} + \left\langle (h_t - Ah_{t-1})^\mathsf{T}\,\Sigma_H^{-1}\,(h_t - Ah_{t-1})\right\rangle_{q(A,\Sigma_H)}\right]. \quad (29)$$

Optimally, $q(A|\Sigma_H)$ and $q(B|\Sigma_V)$ are Gaussians (see the DIRAC publication for fuller details), so we can easily carry out the averages. The further averages over $q(\Sigma_H)$ and $q(\Sigma_V)$ are also easy due to conjugacy. Whilst this defines the distribution $q(h_{1:T})$, quantities such as $q(h_t)$, which are required for the parameter updates (see the Appendices), need to be inferred from this distribution. Clearly, in the non-Bayesian case, the averages over the parameters are not present, and the above simply represents an LGSSM whose visible variables have been clamped into their evidential states. In that case, inference can be performed using any standard method. Our aim, therefore, is to try to represent the *averaged* Equation (29) directly as an LGSSM $\tilde{q}(h_{1:T}|\tilde{v}_{1:T})$, for some suitable parameter settings.

Mean + Fluctuation Decomposition

A useful decomposition is to write

$$\left\langle (v_t - Bh_t)^\mathsf{T}\Sigma_V^{-1}(v_t - Bh_t)\right\rangle_{q(B,\Sigma_V)} = \underbrace{(v_t - \langle B\rangle h_t)^\mathsf{T}\left\langle\Sigma_V^{-1}\right\rangle(v_t - \langle B\rangle h_t)}_{mean} + \underbrace{h_t^\mathsf{T} S_B h_t}_{fluctuation},$$

and similarly

$$\left\langle (h_t - Ah_{t-1})^\mathsf{T}\Sigma_H^{-1}(h_t - Ah_{t-1})\right\rangle_{q(A,\Sigma_H)} = \underbrace{(h_t - \langle A\rangle h_{t-1})^\mathsf{T}\left\langle\Sigma_H^{-1}\right\rangle(h_t - \langle A\rangle h_{t-1})}_{mean} + \underbrace{h_{t-1}^\mathsf{T} S_A h_{t-1}}_{fluctuation},$$

where the parameter covariances are $S_B = VH_B^{-1}$ and $S_A = HH_A^{-1}$ (see publication for details). The mean terms simply represent a clamped LGSSM with averaged parameters. However, the extra contributions from the fluctuations mean that Equation (29) cannot be written as a clamped LGSSM with averaged parameters. In order to deal with these extra terms, our idea is to treat the fluctuations as arising from an augmented visible variable, for which Equation (29) can then be considered as a clamped LGSSM.

Inference Using an Augmented LGSSM

To represent Equation (29) as a LGSSM $\tilde{q}(h_{1:T}|\tilde{v}_{1:T})$, we augment v_t and B as[13]:

$$\tilde{v}_t = vert(v_t, \mathbf{0}_H, \mathbf{0}_H), \qquad \tilde{B} = vert(\langle B\rangle, U_A, U_B),$$

where U_A is the Cholesky decomposition of S_A, so that $U_A^\mathsf{T}U_A = S_A$. Similarly, U_B is the Cholesky decomposition of S_B. The equivalent LGSSM $\tilde{q}(h_{1:T}|\tilde{v}_{1:T})$ is then completed by specifying[14]

$$\tilde{A} \equiv \langle A\rangle, \quad \tilde{\Sigma}_H \equiv \left\langle\Sigma_H^{-1}\right\rangle^{-1}, \quad \tilde{\Sigma}_V \equiv diag(\left\langle\Sigma_V^{-1}\right\rangle^{-1}, I_H, I_H), \quad \tilde{\mu} \equiv \mu, \quad \tilde{\Sigma} \equiv \Sigma.$$

[12] For simplicity of exposition, we ignore the first time-point here.

[13] The notation $vert(x_1, \ldots, x_n)$ stands for vertically concatenating the arguments $x_1, \ldots, x_n$.

[14] Strictly, we need a time-dependent emission $\tilde{B}_t = \tilde{B}$, for $t = 1, \ldots, T-1$. For time T, $\tilde{B}_T$ has the Cholesky factor U_A replaced by $\mathbf{0}_{H,H}$.

The validity of this parameter assignment can be checked by showing that, up to negligible constants, the exponent of this augmented LGSSM has the same form as Equation (29). Now that this has been written as an LGSSM $\tilde{q}(h_{1:T}|\tilde{v}_{1:T})$, standard inference routines in the literature may be applied to compute $q(h_t) = \tilde{q}(h_t|\tilde{v}_{1:T})$ [22, 32, 23][15].

Algorithm 1 LGSSM: Forward and backward recursive updates. The smoothed posterior $p(h_t|v_{1:T})$ is returned in the mean $\hat{h}_t^T$ and covariance P_t^T.

procedure FORWARD
1a: $P \leftarrow \Sigma$
1b: $P \leftarrow (\Sigma^{-1} + S_A + S_B)^{-1} = (I - \Sigma U_{AB}\left(I + U_{AB}^\mathsf{T}\Sigma U_{AB}\right)^{-1} U_{AB}^\mathsf{T}) \equiv D\Sigma$
2a: $\hat{h}_1^0 \leftarrow \mu$
2b: $\hat{h}_1^0 \leftarrow D\mu$
3: $K \leftarrow PB^\mathsf{T}(BPB^\mathsf{T} + \Sigma_V)^{-1}$, $P_1^1 \leftarrow (I - KB)P$, $\hat{h}_1^1 \leftarrow \hat{h}_1^0 + K(v_t - B\hat{h}_1^0)$
for $t \leftarrow 2, T$ **do**
4: $P_t^{t-1} \leftarrow AP_{t-1}^{t-1}A^T + \Sigma_H$
5a: $P \leftarrow P_t^{t-1}$
5b: $P \leftarrow D_t P_t^{t-1}$, where $D_t \equiv (I - P_t^{t-1}U_{AB}\left(I + U_{AB}^\mathsf{T}P_t^{t-1}U_{AB}\right)^{-1} U_{AB}^\mathsf{T})$
6a: $\hat{h}_t^{t-1} \leftarrow A\hat{h}_{t-1}^{t-1}$
6b: $\hat{h}_t^{t-1} \leftarrow D_t A\hat{h}_{t-1}^{t-1}$
7: $K \leftarrow PB^\mathsf{T}(BPB^\mathsf{T} + \Sigma_V)^{-1}$, $P_t^t \leftarrow (I - KB)P$, $\hat{h}_t^t \leftarrow \hat{h}_t^{t-1} + K(v_t - B\hat{h}_t^{t-1})$
end for
end procedure
procedure BACKWARD
for $t \leftarrow T-1, 1$ **do**
$\overleftarrow{A_t} \leftarrow P_t^t A^\mathsf{T}(P_{t+1}^t)^{-1}$
$P_t^T \leftarrow P_t^t + \overleftarrow{A_t}(P_{t+1}^T - P_{t+1}^t)\overleftarrow{A_t}^\mathsf{T}$
$\hat{h}_t^T \leftarrow \hat{h}_t^t + \overleftarrow{A_t}(\hat{h}_{t+1}^T - A\hat{h}_t^t)$
end for
end procedure

For completeness, we decided to describe the standard predictor-corrector form of a Kalman filter, together with the Rauch-Tung-Striebel recursions [23] for performing inference in an LGSSM. These are given in Algorithm 1. To compute $\tilde{q}(h_t|\tilde{v}_{1:T})$, we then call the FORWARD and BACKWARD procedures.

We present two variants of the FORWARD pass. Either we may call procedure FORWARD in Algorithm 1 with parameters $\tilde{A}, \tilde{B}, \tilde{\Sigma}_H, \tilde{\Sigma}_V, \tilde{\mu}, \tilde{\Sigma}$ and the augmented visible variables $\tilde{v}_t$ in which we use steps 1a, 2a, 5a and 6a. This is exactly the predictor-corrector form of a Kalman filter [23]. Otherwise, in order to reduce the computational cost, we may call procedure FORWARD with the parameters $\langle A\rangle, \langle B\rangle, \left\langle \Sigma_H^{-1}\right\rangle^{-1}, \left\langle \Sigma_V^{-1}\right\rangle^{-1}, \mu, \Sigma$ and the original visible variable v_t in which we use steps 1b (where $U_{AB}^\mathsf{T}U_{AB} \equiv S_A + S_B$), 2b, 5b and 6b. The two algorithms are mathematically equivalent. Computing $q(h_t) = \tilde{q}(h_t|\tilde{v}_{1:T})$ is then completed by calling the common BACKWARD pass.

The important point here is that the reader may supply any standard Kalman Filtering/Smoothing routine, and simply call it with the appropriate parameters. In some parameter regimes, or in very long time series, numerical stability may be a serious concern, for which several stabilized

[15]Note that, since the augmented LGSSM $\tilde{q}(h_{1:T}|\tilde{v}_{1:T})$ is designed to match the *fully* clamped distribution $q(h_{1:T})$, filtering $\tilde{q}(h_{1:T}|\tilde{v}_{1:T})$ does not correspond to filtering $q(h_{1:T})$.

algorithms have been developed over the years, for example the square-root forms [31, 32, 23]. By converting the problem to a standard form, we have therefore unified and simplified inference, so that future applications may be more readily developed.

4.2.1 *Relation to Previous Approaches*

An alternative approach to the one above, and taken in [28, 26], is to recognize that the posterior is

$$\log q(h_{1:T}) = \sum_{t=2}^{T} \phi_t(h_{t-1}, h_t) + const.$$

for suitably defined quadratic forms $\phi_t(h_{t-1}, h_t)$. Here the potentials $\phi_t(h_{t-1}, h_t)$ encode the averaging over the parameters A, B, Σ_H, Σ_V. The approach taken in [28] is to recognize this as a pairwise Markov chain, for which the Belief Propagation recursions may be applied. The backward pass from Belief Propagation makes use of the observations $v_{1:T}$, so that any approximate online treatment would be difficult. The approach in [26] is based on a Kullback-Leibler minimization of the posterior with a chain structure, which is algorithmically equivalent to Belief Propagation. Whilst mathematically valid procedures, the resulting algorithms do not correspond to any of the standard forms in the Kalman Filtering/Smoothing literature, whose properties have been well studied [34].

4.3 An Application to Bayesian ICA

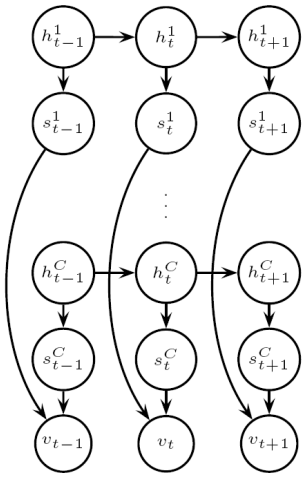


Figure 7: The structure of the LGSSM for ICA.

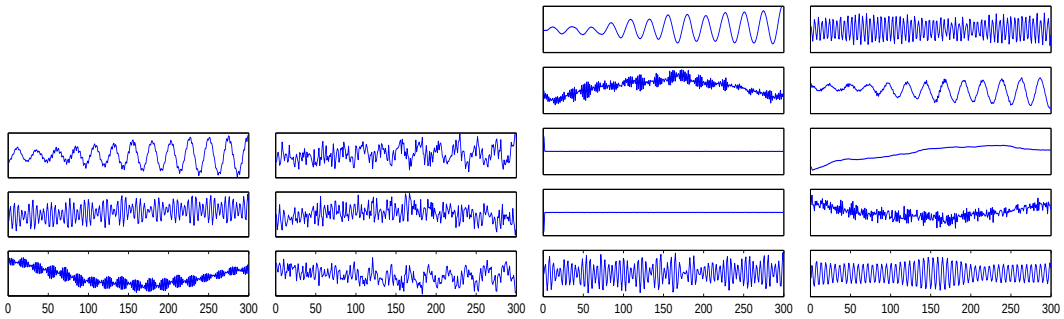
A particular case for which the Bayesian LGSSM is of interest is in extracting independent source signals underlying a multivariate time-series [35, 26]. This will demonstrate how the approach developed in Section (4.2) makes VB easily to apply. The sources s^i are modeled as independent in the following sense:

$$p(s^i_{1:T}, s^j_{1:T}) = p(s^i_{1:T})p(s^j_{1:T}), \qquad \text{for } i \neq j, \qquad i, j = 1, \ldots, C.$$

Independence implies block diagonal transition and state noise matrices A, Σ_H and Σ, where each block c has dimension H_c. A one dimensional source s^c_t for each independent dynamical subsystem is then formed from $s^c_t = \mathbf{1}_c^\mathsf{T} h^c_t$, where $\mathbf{1}_c$ is a unit vector and h^c_t is the state of dynamical system c. Combining the sources, we can write $s_t = Ph_t$, where $P = diag(\mathbf{1}_1^\mathsf{T}, \ldots, \mathbf{1}_C^\mathsf{T})$, $h_t = vert(h^1_t, \ldots, h^C_t)$. The resulting emission matrix is constrained to be of the form $B = WP$, where W is the $V \times C$ mixing matrix. This means that the observations are formed from linearly mixing the sources, $v_t = Ws_t + \eta^v_t$. The graphical structure of this model is presented in Figure (7). To encourage redundant components to be removed, we place a zero mean Gaussian prior on W. In this case, we do not define a prior for the parameters Σ_H and Σ_V which are instead considered as hyperparameters. More details of the model are given in [35].

4.4 Demonstration

As a simple demonstration, we used a LGSSM to generate 3 sources s^c_t with random 5×5 transition matrices A^c, $\mu = \mathbf{0}_H$ and $\Sigma \equiv \Sigma_H \equiv I_H$. The sources were mixed into three observations $v_t = Ws_t + \eta^v_t$, for W chosen with elements from a zero mean unit variance Gaussian distribution, and $\Sigma_V = I_V$. We then trained a Bayesian LGSSM with 5 sources and



(a) (b) (c) (d)

Figure 8: (a) Original sources s_t. (b) Observations resulting from mixing the original sources, $v_t = W s_t + \eta_t^v$, $\eta_t^v \sim \mathcal{N}(0, I)$. (c) Recovered sources using the Bayesian LGSSM. (d) Sources found with MAP LGSSM.

7×7 transition matrices A^c. To bias the model to find the simplest sources, we used $\hat{A}^c \equiv \mathbf{0}_{H_c,H_c}$ for all sources. In Figure (8a) and Figure (8b) we see the original sources and the noisy observations respectively. In Figure (8c) we see the estimated sources from our method after convergence of the hyperparameter updates. Two of the 5 sources have been removed, and the remaining three are a reasonable estimation of the original sources. Another possible approach for introducing prior knowledge is to use a Maximum a Posteriori (MAP) procedure by adding a prior term to the original log-likelihood $\log p(v_{1:T}|A, W, \Theta) + \log p(A|\alpha) + \log p(W|\beta)$. However, it is not clear how to reliably find the hyperparameters α and β in this case. One solution is to estimate them by optimizing the new objective function jointly with respect to the parameters and hyperparameters (this is the so-called joint map estimation – see for example [36]). A typical result of using this joint MAP approach on the artificial data is presented in Figure (8d). The joint MAP does not estimate the hyperparameters well, and the incorrect number of sources is identified.

4.5 Discussion

We considered the application of Variational Bayesian learning to Linear Gaussian State-Space Models. This is an important class of models with widespread application, and finding a simple way to implement this approximate Bayesian procedure is of considerable interest. The most demanding part of the procedure is inference of the hidden states of the model. Previously, this has been achieved using Belief Propagation, which differs from inference in the Kalman Filtering/Smoothing literature, for which highly efficient and stabilized procedures exist. A central contribution of this work is to show how inference *can* be written using the standard Kalman Filtering/Smoothing recursions by augmenting the original model. Additionally, a minor modification to the standard Kalman Filtering routine may be applied for computational efficiency. We demonstrated the elegance and unity of our approach by showing how to easily apply a Variational Bayes analysis of temporal ICA. We hope that this simple and unifying interpretation of Variational Bayesian LGSSMs may therefore facilitate the further application to related models.

5 Blind Signal Separation by Disjoint Component Analysis

5.1 Introduction

Representation of measured data in terms of a number of generating causes or underlying "sources" is an important problem that has gained widespread attention in recent years, either

with the goal of extracting known-to-exist sources from measurements (blind source separation), or in order to find an efficient—possibly lower-dimensional—description of given data (exploratory data analysis).

We propose and investigate a novel technique, "disjoint component analysis" (DCA) that is based on the goal of extracting components with maximally disjoint support from given data. I.e., it is sought to describe the data in terms of components of which as few as possible should be activated at any single time (or sample) point. Ideally, only a single source process would account for a single sample of measured data, a goal that clearly is too strong for real-world data. We demonstrate that it can be significantly relaxed while still retaining the beneficial characteristics of the method.

Disjoint support between generating source processes may constitute a relevant general principle in domains where other assumptions, e.g., statistical independence and the implied effective physical separation of generating source processes, have to be postulated or justified post-hoc rather than deduced a-priori. In some cases such as communicating speakers or densely interconnected nervous cells in the brain, theoretical considerations argue in favor of dependencies between source processes. Even though such dependencies might turn out to be largely negligible in some domains, it does appear to be worthwhile to consider the implications of incorporating such dependencies into the models.

In the opposite direction (and with a different intention than ours), some authors have argued that sources that are often regarded as independent can effectively be modeled as being "w-disjoint orthogonal" [37].

5.2 Disjoint component analysis

5.2.1 Derivation of algorithm

We consider N observed signals $\mathbf{x}(t) = [x_1(t), \ldots, x_N(t)]^T$ which may be thought to have been generated from (for convenience) N underlying sources $\mathbf{s}(t) = [s_1(t), \ldots, s_N(t)]^T$ by multiplication with a mixing system $\mathbf{A}$ as

$$\mathbf{x}(t) = \mathbf{A}\,\mathbf{s}(t) \tag{30}$$

It is sought to linearly transform the observations by a matrix $\mathbf{W}$ to obtain output signals

$$\mathbf{y}(t) = \mathbf{W}\,\mathbf{x}(t) \tag{31}$$

with components $\mathbf{y}(t) = [y_1(t), \ldots, y_N(t)]^T$. When source reconstruction is desired, these should resemble the sources up to arbitrary rescaling and permutation. When an exploratory data analysis view is adopted, the output signals should convey a signal representation that is meaningful in some to-be-specified sense.

A central notion in our approach is the overlap

$$o_{ij} = E(|y_i|\,|y_j|) \tag{32}$$

between two output signals y_i and y_j with $i \neq j$, where $E(\cdot)$ denotes expectation and sample index t is omitted where convenient. With $o_{ij} \geq 0$ and $o_{ij} = 0$ if and only if $y_i(t)\,y_j(t) = 0$ for all t and $i \neq j$, two signals y_i and y_j have *disjoint support* if $o_{ij} = 0$. In this case, y_i and y_j are called *disjoint*, i.e., at most one of the signals is non-zero at any time.

For strictly disjoint source signals $\mathbf{s}(t)$ and a non-singular matrix $\mathbf{A}$, strictly disjoint outputs can be obtained that resemble the sources up to arbitrary permutation and rescaling.

Note that in this case sources are not mutually *independent* but exhibit statistical *dependencies* through the negative correlations of their signal envelopes or signal power time-courses.

While it is not possible in general to linearly transform an arbitrary signal $\mathbf{x}(t)$ into a signal $\mathbf{y}(t)$ with only disjoint components, finding minimally overlapping outputs is a natural goal as it corresponds to a signal description in terms of processes out of which only a small number is active at any given time. In this sense, disjoint component analysis bears similarities with both parts-based approaches and sparse coding assumptions. A natural choice to obtain maximally disjoint, minimally overlapping output signals is minimization of the function

$$H = \frac{1}{2}\sum_{i \neq j} o_{ij} = \frac{1}{2}\sum_{i \neq j} E(|y_i|\,|y_j|) \tag{33}$$

The global minimum $H = 0$ is attained only for strictly disjoint signals where for all t any signal $y_i(t) \neq 0$ if and only if $y_j(t) = 0$ for all $j \neq i$. Substituting 31 into 33, the partial derivatives are given by

$$\frac{\partial H}{\partial w_{ij}} = E\Big(\mathrm{sign}(y_i)\, x_j \sum_{k \neq i} |y_k|\Big) \tag{34}$$

which in matrix notation reads

$$\nabla H = E\left(-\mathbf{y}\mathbf{x}^H + ||\mathbf{y}||_1 \mathrm{sign}(\mathbf{y})\mathbf{x}^H\right) \tag{35}$$

where $||\mathbf{y}||_1 = \sum_i |u_i|$ denotes the 1-norm of $\mathbf{y}$.
A natural gradient [38] may be derived by right-multiplication with $\mathbf{W}^T\mathbf{W}$, yielding

$$\tilde{\nabla} H = E\left(-\mathbf{y}\mathbf{y}^H + ||\mathbf{y}||_1 \mathrm{sign}(\mathbf{y})\mathbf{y}^H\right)\mathbf{W} \tag{36}$$

Without regularization the gradients converge to the trivial solution $\mathbf{W} = \mathbf{0}$. To remove the scaling ambiguity each row $\mathbf{w}_i$ of matrix $\mathbf{W}$ is fixed to unit-norm $||\mathbf{w}_i||_2 = 1$. Hence, each row $\boldsymbol{\Delta}_i$ of ∇H is projected according to

$$\boldsymbol{\Delta}_i^{\perp} = \boldsymbol{\Delta}_i - (\boldsymbol{\Delta}_i^H \mathbf{w}_i)\,\mathbf{w}_i \tag{37}$$

resulting in the projected gradient matrix $\boldsymbol{\Delta}^{\perp}$ that is then used for gradient descent. The final update rule for matrix $\mathbf{W}$ with a step size of η is

$$\mathbf{W} \leftarrow \mathbf{W} - \eta\,\boldsymbol{\Delta}^{\perp} \tag{38}$$

for the ordinary gradient and similarly for the natural gradient. Periodic row re-normalization of $\mathbf{W}$ is applied to keep it on the constraint manifold for non-infinitesimal η.

5.3 Evaluation

5.3.1 Synthetic data generation

Disjoint sources $s_i(t)$ may be generated from mutually independent signals $\zeta_i(t)$ by multiplying them with disjoint masking functions $\mu_i(t) \in \{0, 1\}$ for all i, t and

$$s_i(t) = \mu_i(t)\,\zeta_i(t) \tag{39}$$

$$E(\mu_i\,\mu_j) = 0 \quad \text{if} \quad i \neq j \tag{40}$$

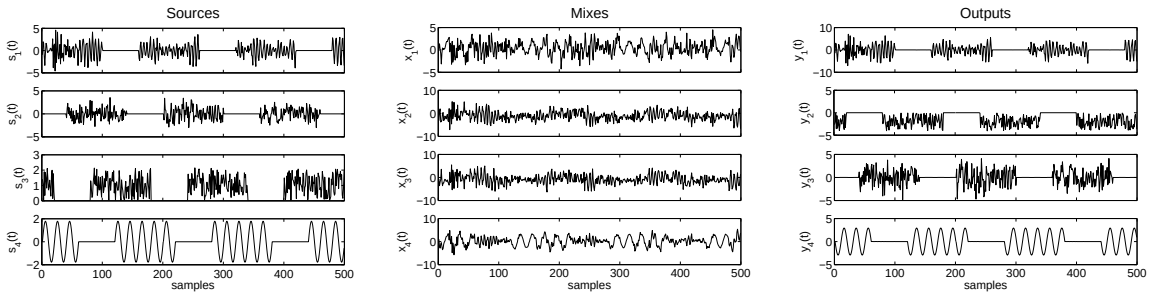


Figure 9: Disjoint component analysis of four sources (left panel) which are not strictly disjoint but exhibit significant overlap. Sources were mixed with a randomly chosen 4×4 mixing matrix to yield observation signals (center panel) which were successfully separated into the original sources up to arbitrary permutation, rescaling and sign flip (right panel) using DCA.

These sources may then be used to generate observations by multiplication with a matrix $\mathbf{A}$ according to Eq. 30.

Strictly disjoint sources with zero overlap are not expected to be an appropriate model for real data. Hence, sources with variable masker overlap γ_{ij}, which may depend on the source pair (i, j),

$$\gamma_{ij} = E(\mu_i \mu_j) \,/\, E(\mu_i^2) \tag{41}$$

with $E(\mu_i^2) = \text{const}$ for all i are also generated. In the experiments reported below masker overlap γ_{ij} is chosen such that a value of $\gamma_{ij} = 1$ corresponds to a source pair (s_i, s_j) exhibiting mutual statistical dependence through maskers with *positive* correlation. The value $\gamma_{ij} = 0$ corresponds to strictly disjoint sources that exhibit mutual statistical dependence through maskers with *negative* correlation. Finally, a value of $\gamma_{ij} = 0.5$ coincides with statistically *independent* sources (s_i, s_j) because of uncorrelated maskers (and statistically independent $\zeta_i(t)$).

The signal generation scheme was inspired by a functional magnetic resonance imaging (fMRI) experiment design [39].

5.3.2 Separation of synthetic sources

Four sources were generated according to the scheme described above, mixed with a randomly chosen mixing matrix and processed with the natural gradient disjoint component analysis algorithm (Eq. 36) with regularization (Eq. 37). The underlying mutually independent signals $\zeta_i(t)$ were chosen as a speech signal (ζ_1), i.i.d. noise from a normal distribution with zero-mean and unit-variance (ζ_2), i.i.d. noise from a uniform distribution on the interval $[0, 1]$ (ζ_3), and a sine wave (ζ_4). The maskers $\mu_i(t)$ were chosen such that $\gamma_{ij} = 0.6$ for source pairs $(1, 2)$, $(2, 3)$, $(3, 4)$, $(1, 4)$, and $\gamma_{ij} = 0.4$ for source pairs $(1, 3)$, $(2, 4)$. Source signals, observed (mixed) signals and output signals are displayed in Fig. 9, demonstrating that the algorithm performs successful separation even though sources are not strictly disjoint but show significant overlap. Similarly, the algorithm successfully separates mixtures of four strictly disjoint sources with $\gamma_{ij} = 0$ for all $i \neq j$ (data not shown here).

5.3.3 Variable degree of overlap

The goal of this experiment was to systematically study the influence of the degree of overlap on the performance of the disjoint component analysis algorithm. Results are reported for the

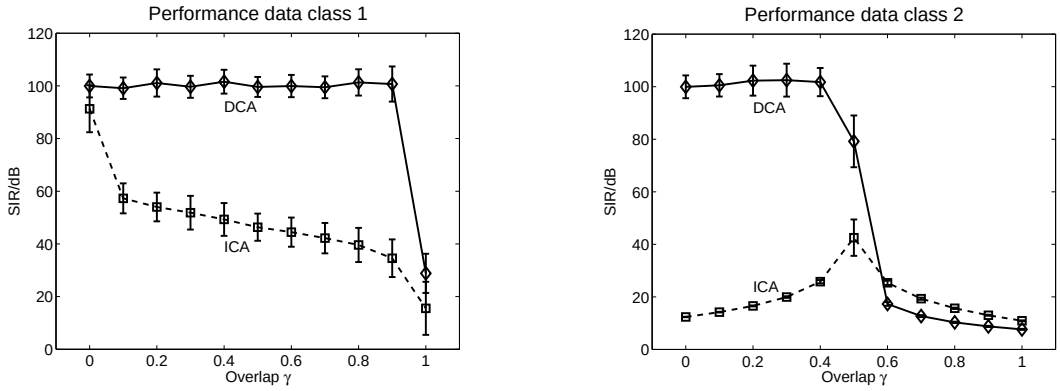


Figure 10: Separation performance of DCA and ICA in terms of signal-to-interference ratio (SIR) in dB after separation. Performance is given for data class 1 (left panel, sources with positive and negative observation values) and data class 2 (right panel, sources with positive only observation values) as a function of overlap γ. A value of $\gamma = 0$ corresponds to strictly disjoint sources (statistical dependencies between sources through negative correlation of signal envelopes); $\gamma = 0.5$ corresponds to statistically independent sources; and $\gamma = 1.0$ corresponds to fully overlapping, not disjoint sources (statistical dependencies through positive correlation of signal envelopes). Mean and variance of performance for 100 separation runs, each with independently generated data, are given for each condition.

gradient version of the algorithm (Eq. 35) with regularization (Eq. 37). Results for the natural gradient version are virtually identical and not reported separately.

Sources were generated based on two different underlying signal classes. In the first part of the experiment ("data class 1"), two sources s_1 and s_2 were generated from ζ_1 and ζ_2 that were drawn as i.i.d. signals from a zero-mean and unit-variance normal distribution, hence containing positive and negative values.

In the second part of the experiment ("data class 2"), ζ_1 and ζ_2 were chosen to be i.i.d. signals from a uniform distribution on the interval $[0, 1]$, hence containing only positive values.

For both data sets the single overlap parameter γ was varied from 0 (no overlap, source dependence through negative masker correlation) via 0.5 (50% overlap, statistically independent sources) to 1.0 (full overlap, source dependence through positive masker correlation) in steps of 0.1.

Hence, 11 data set conditions were generated for each of the two data classes. For each condition, disjoint component analysis was performed on 100 individual datasets drawn independently according to the description above. This resulted in a total of 2200 datasets each with 10000 samples for each of the two sources.

Fig. 10 shows the results with mean and variance of signal separation in dB signal-to-interference ratio (SIR) after separation separately for data class 1 (left panel) and data class 2 (right panel). For data class 1 with sources that adopt positive and negative values, DCA separation performance shows no significant dependence on the overlap parameter γ except (as expected) for complete overlap at $\gamma = 1$ where the algorithm essentially attempts to separate two i.i.d. normally distributed sources which is ill-posed. In all other cases of data class 1, DCA separation is excellent with about 100 dB SIR.

The results look different for data class 2 with positive only source values. Separation remains excellent for data sets with a small overlap ($0.0 \leq \gamma \leq 0.4$), with again about 100 dB SIR. In the case of independent sources at $\gamma = 0.5$, separation is still very good at 80 dB. Performance breaks down for large overlaps ($1.0 \geq \gamma \geq 0.6$), an effect which we attribute to the positivity of the sources.

5.3.4 Comparison with independent component analysis

The same data generated for subsection 5.3.3 was re-analyzed with natural gradient infomax ICA [40, 38] using the ICA toolbox [41, 42] with logistic function non-linearity. For comparison, a simple gradient approach with fixed step size and sign function non-linearity was also used and gave virtually identical results for data class 1. On data class 2, the fixed step gradient approach gave qualitatively similar results but was outperformed by the referenced ICA toolbox in terms of SIR separation performance. All source signals have been checked to have positive kurtosis.

Results in Fig. 10 show that in most cases ICA results in a poorer SIR than DCA. For data class 2, ICA performs best when sources are independent ($\gamma = 0.5$) with a drop off in performance towards both lower and higher source overlaps, which is plausible due to ICA's independence assumption.

For data class 1, ICA shows excellent signal separation for strictly disjoint sources ($\gamma = 0.0$). Performance is significantly lower, though still good, for independent sources, which seems to stand in contradiction to the independence assumption. As expected, performance decreases towards sources with strong overlap ($\gamma = 1.0$).

5.4 Conclusion

Disjoint component analysis (DCA) has been shown to yield excellent performance for strictly disjoint and moderately disjoint data sets. For data with high overlap between sources (weakly disjoint), performance depends on the specific type of data, with excellent performance for data sets with sources that take positive and negative observation values, and a break-down of performance in case of purely positive source data.

The empirical algorithm evaluation showed a better separation performance for DCA than for ICA under most conditions. Interestingly, ICA produced the best performance not for statistically independent sources but for strictly disjoint ones (cf. also ICA 2006 presentation of I.C. Daubechies).

While far from being conclusive, the results presented here appear to warrant a closer investigation of the differences and similarities of both algorithm classes. It would be desirable to gain experience with a wider range of synthetic and natural data than could be presented here.

We are tempted to speculate that DCA might be appropriate in particular for analyzing data where the independence assumption is not strictly fulfilled, where a data representation in terms of disjoint components is preferable to independent components, and where signals are comprised of positive only measurement values. This could be the case, e.g., for brain signals such fMRI, for data from dialog speech signals, and for comparably short signal sequences where independence cannot be fully attained due to finite sample effects.

References

[1] A. B. Poritz. Linear predictive hidden Markov models and the speech signal. In *Proceedings of the IEEE International Confererence on Acoustics, Speech, and Signal Processing*, volume 7, pages 1291–1294, May 1982.

[2] Y. Ephraim and W. J. J. Roberts. Revisiting autoregressive hidden Markov modeling of speech signals. *IEEE Signal Processing Letters*, 12(2):166–169, February 2005.

[3] K. Y. Lee and J. Lee. Recognition of noisy speech by a nonstationary AR HMM with gain adaptation under unknown noise. *IEEE Transactions on Speech and Audio Processing*, 9(7):741–746, October 2001.

[4] Y. Ephraim. Gain–adapted hidden Markov models for recognition of clean and noisy speech. *IEEE Transaction on Signal Processing*, 40(6):1303–1316, June 1992.

[5] H. Attias, J. C. Platt, A. Acero, and L. Deng. Speech denoising and dereverberation using probabilistic models. In *Proceedings of NIPS 2001*, 2001.

[6] R.G. Leonard. A database for speaker independent digit recognition. In *Proceedings of ICASSP84*, volume 3, 1984.

[7] The Hidden Markov Model Toolkit. Available at: `http://htk.eng.cam.ac.uk/`.

[8] Y. Bar-Shalom and Xiao-Rong Li. *Estimation and Tracking : Principles, Techniques and Software*. Artech House, Norwood, MA, 1998.

[9] V. Pavlovic, J. M. Rehg, and J. MacCormick. Learning switching linear models of human motion. In *Advances in Neural Information Processing systems (NIPS 13)*, pages 981–987, 2001.

[10] A. T. Cemgil, B. Kappen, and D. Barber. A Generative Model for Music Transcription. *IEEE Transactions on Audio, Speech and Language Processing*, 14(2):679 – 694, 2006.

[11] U. N. Lerner. *Hybrid Bayesian Networks for Reasoning about Complex Systems*. PhD thesis, Stanford University, 2002.

[12] O. Zoeter. *Monitoring non-linear and switching dynamical systems*. PhD thesis, Radboud University Nijmegen, 2005.

[13] T. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, MIT Media Lab, 2001.

[14] C-J. Kim. Dynamic linear models with Markov-switching. *Journal of Econometrics*, 60:1–22, 1994.

[15] T. Heskes and O. Zoeter. Expectation Propagation for approximate inference in dynamic Bayesian networks. In A. Darwiche and N. Friedman, editors, *Uncertainty in Artificial Intelligence*, pages 216–223, 2002.

[16] S. Lauritzen and F. Jensen. Stable local computation with conditional Gaussian distributions. *Statistics and Computing*, 11:191–203, 2001.

[17] G. Kitagawa. The Two-Filter Formula for Smoothing and an implementation of the Gaussian-sum smoother. *Annals of the Institute of Statistical Mathematics*, 46(4):605–623, 1994.

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

[19] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[20] B. Mesot and D. Barber. Switching linear dynamical systems for noise robust speech recognition. IDIAP-RR 08, 2006.

[21] G. Lathoud, M. Magimai-Doss, B. Mesot, and H. Bourlard. Unsupervised spectral subtraction for noise-robust ASR. In *Proceedings of ASRU 2005*, pages 189–194, November 2005.

[22] Y. Bar-Shalom and X.-R. Li. *Estimation and Tracking: Principles, Techniques and Software*. Artech House, 1998.

[23] M. S. Grewal and A. P. Andrews. *Kalman Filtering: Theory and Practice Using MATLAB*. John Wiley and Sons, Inc., 2001.

[24] R. H. Shumway and D. S. Stoffer. *Time Series Analysis and Its Applications*. Springer, 2000.

[25] M. J. Beal, F. Falciani, Z. Ghahramani, C. Rangel, and D. L. Wild. A Bayesian approach to reconstructing genetic regulatory networks with hidden factors. *Bioinformatics*, 21:349–356, 2005.

[26] A. T. Cemgil and S. J. Godsill. Probabilistic phase vocoder and its application to interpolation of missing values in audio signals. In *13th European Signal Processing Conference*, 2005.

[27] H. Valpola and J. Karhunen. An unsupervised ensemble learning method for nonlinear dynamic state-space models. *Neural Computation*, 14:2647–2692, 2002.

[28] M. J. Beal. *Variational Algorithms for Approximate Bayesian Inference*. Ph.D. thesis, Gatsby Computational Neuroscience Unit, University College London, 2003.

[29] Davy. M. and S. J. Godsill. Bayesian harmonic models for musical signal analysis (with discussion). In J.O. Bernardo, J.O. Berger, A.P Dawid, and A.F.M. Smith, editors, *Bayesian Statistics VII*. Oxford University Press, 2003.

[30] D. J. C. MacKay. Ensemble learning and evidence maximisation. Unpublished manuscipt: www.variational-bayes.org, 1995.

[31] M. Morf and T. Kailath. Square-root algorithms for least-squares estimation. *IEEE Transactions on Automatic Control*, 20:487–497, 1975.

[32] P. Park and T. Kailath. New square-root smoothing algorithms. *IEEE Transactions on Automatic Control*, 41:727–732, 1996.

[33] E. Niedermeyer and F. Lopes Da Silva. *Electroencephalography: basic principles, clinical applications and related fields*. Lippincott Williams and Wilkins, 1999.

[34] M. Verhaegen and P. Van Dooren. Numerical Aspects of Different Kalman Filter Implementations. *IEEE Transactions of Automatic Control*, 31(10):907–917, 1986.

[35] Anonymous. Bayesian Linear Gaussian State-Space Models for Biosignal Decomposition. 2006.

[36] S. S. Saquib, C. A. Bouman, and K. Sauer. ML parameter estimation for Markov random fields with applications to Bayesian tomography. *IEEE Transactions on Image Processing*, 7:1029–1044, 1998.

[37] S. Rickard and Z. Yilmaz, "On the approximate w-disjoint orthogonality of speech," in *ICASSP '02*, 2002, pp. I–529–I–532.

[38] S.-I. Amari, "Natural gradient works efficiently in learning," *Neural Computation*, vol. 10, pp. 251–276, 1998.

[39] M.S. Benharrosh, S. Takerkart, J.D. Cohen, I.C. Daubechies, and W. Richter, "Using ICA on fMRI: Does independence matter?," in *Human Brain Mapping*, 2003, abstract no. 784.

[40] A.J. Bell and T.J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129–1159, 1995.

[41] S. Makeig et al., "EEGLAB: ICA toolbox for psychophysical research," Swartz Center for Computational Neuroscience, Institute for Neural Computation, University of California San Diego, http://www.sccn.ucsd.edu:/eeglab, 2000.

[42] S. Makeig, A.J. Bell, T.-P. Jung, and T.J. Sejnowski, "Independent component analysis of electroencephalographic data," in *Advances in neural information processing system*, 1996, vol. 8, pp. 145–151.