The Variational Kalman Smoother

Matthew J. Beal and Zoubin Ghahramani

Gatsby Computational Neuroscience Unit

{m.beal,zoubin}@gatsby.ucl.ac.uk

http://www.gatsby.ucl.ac.uk

May 22, 2000. last revision April 6, 2001

Abstract

In this note we outline the derivation of the variational Kalman smoother, in the context of Bayesian Linear Dynamical Systems. The smoother is an efficient algorithm for the E-step in the Expectation-Maximisation (EM) algorithm for linear-Gaussian state-space models. However, inference approximations are required if we hold distributions over parameters. We derive the E-step updates for the hidden states (the variational smoother), and the M-step updates for the parameter distributions. We show that inference of the hidden state is tractable for *any* distribution over parameters, provided the expectations of certain quantities are available, analytically or otherwise.[1]

1 Introduction to variational Linear Dynamical Systems

The reader is referred to [1] and [2] for the theoretical framework and motivation for variational Bayesian learning. The joint probability for the state of the hidden, $\mathbf{x}_{1:T}$, and observed, $\mathbf{y}_{1:T}$, variables for a Markov process is given by

$$P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T}) = P(\mathbf{x}_1)P(\mathbf{y}_1|\mathbf{x}_1)\prod_{t=2}^{T} P(\mathbf{x}_t|\mathbf{x}_{t-1})P(\mathbf{y}_t|\mathbf{x}_t) \tag{1}$$

In the case of a linear dynamical system with Gaussian noise, the state dynamics and output distributions are governed by

$$\mathbf{x}_t = A\mathbf{x}_{t-1} + \mathbf{w}_t, \qquad \mathbf{y}_t = C\mathbf{x}_t + \mathbf{v}_t, \qquad \text{with } \mathbf{w}_t \sim \mathrm{N}(0, E), \text{ and } \mathbf{v}_t \sim \mathrm{N}(0, R) \tag{2}$$

where A is the transition matrix, C is the output matrix, and $\mathbf{w}_t$ and $\mathbf{v}_t$ are the Gaussian noise vectors added at time t. Without loss of generality, the state noise E can be set to the identity (arbitrary rescaling of the state noise can be achieved through changes to A, C and R, the latter a diagonal matrix with entries $1/\boldsymbol{\rho}$, although a full covariance version is not difficult to implement). Thus the full joint for parameters, hidden variables and observed data is given by

$$\begin{aligned} P(A, C, \boldsymbol{\rho}, \mathbf{x}_{1:T}, \mathbf{y}_{1:T}) \;\; = \;\; & P(A|\boldsymbol{\alpha})P(\boldsymbol{\rho}|a,b)P(C|\boldsymbol{\gamma},\boldsymbol{\rho})P(\mathbf{x}_1|\boldsymbol{\pi})P(\mathbf{y}_1|\mathbf{x}_1, C, \boldsymbol{\rho}) \cdot \\ & \prod_{t=2}^{T} P(\mathbf{x}_t|\mathbf{x}_{t-1}, A)P(\mathbf{y}_t|\mathbf{x}_t, C, \boldsymbol{\rho}) \; . \end{aligned} \tag{3}$$

[1] Matlab code which fully implements the variational linear dynamical system model (with inputs) described in this report can be obtained from the tar file at http://www.gatsby.ucl.ac.uk/~beal/papers/vks.tar.gz

By applying Jensen's inequality we can lower bound the log evidence

$$\begin{aligned}
\ln P(\mathbf{y}_{1:T}) &= \ln \int dA\, dB\, dC\, dD\, d\boldsymbol{\rho}\, d\mathbf{x}_{1:T}\; P(A, B, C, D, \boldsymbol{\rho}, \mathbf{x}_{1:T}, \mathbf{y}_{1:T}) && (4)\\
&\geq \int dA\, dB\, dC\, dD\, d\boldsymbol{\rho}\, d\mathbf{x}_{1:T}\; Q(A, B, C, D, \boldsymbol{\rho}, \mathbf{x}_{1:T}) \ln \frac{P(A, B, C, D, \boldsymbol{\rho}, \mathbf{x}_{1:T}, \mathbf{y}_{1:T})}{Q(A, B, C, D, \boldsymbol{\rho}, \mathbf{x}_{1:T})} && (5)\\
&= \mathcal{F}\ .
\end{aligned}$$

We make the approximation to the posterior $Q(A, B, C, D, \boldsymbol{\rho}, \mathbf{x}_{1:T}) = Q(A, B)Q(C, D, \boldsymbol{\rho})Q(\mathbf{x}_{1:T})$. The factorisation of the parameters from the hidden variables is the initial assumption to make inference tractable, and the second factorisation, that of the parameters, falls out of the conditional independencies in the graphical model (there are no terms in the parameter posterior that couple either A or B with either C or B ρ). We choose to write the joint as $Q(B)Q(A|B)Q(\boldsymbol{\rho})Q(D|\boldsymbol{\rho})Q(C|\boldsymbol{\rho}, D)Q(\mathbf{x}_{1:T})$. The integral of (5) then separates into

$$\begin{aligned}
\mathcal{F} &= \int dB\; Q(B) \ln \frac{P(B|\boldsymbol{\beta})}{Q(B)} + \int dB\; Q(B) \int dA\; Q(A|B) \ln \frac{P(A|\boldsymbol{\alpha})}{Q(A|B)}\\
&\quad + \int d\boldsymbol{\rho}\; Q(\boldsymbol{\rho}) \ln \frac{P(\boldsymbol{\rho}|a, b)}{Q(\boldsymbol{\rho})} + \int d\boldsymbol{\rho}\; Q(\boldsymbol{\rho}) \int dD\; Q(D|\boldsymbol{\rho}) \ln \frac{P(D|\boldsymbol{\rho}, \boldsymbol{\delta})}{Q(D|\boldsymbol{\rho})}\\
&\quad + \int d\boldsymbol{\rho}\; Q(\boldsymbol{\rho}) \int dD\; Q(D|\boldsymbol{\rho}) \int dC\; Q(C|\boldsymbol{\rho}, D) \ln \frac{P(C|\boldsymbol{\rho}, \boldsymbol{\gamma})}{Q(C|\boldsymbol{\rho}, D)}\\
&\quad - \int d\mathbf{x}_{1:T}\; Q(\mathbf{x}_{1:T}) \ln Q(\mathbf{x}_{1:T})\\
&\quad + \int dB\; Q(B) \int dA\; Q(A|B) \int d\boldsymbol{\rho}\; Q(\boldsymbol{\rho}) \int dD\; Q(D|\boldsymbol{\rho}) \int dC\; Q(C|\boldsymbol{\rho}, D)\ \cdot\\
&\qquad \ln P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T}|A, B, C, D, \boldsymbol{\rho}). \qquad (6)
\end{aligned}$$

For variational Bayesian learning $\mathcal{F}$ is the key quantity that we work with. Learning proceeds with iterative updates of the variational posteriors. The optimum forms of these approximate posteriors can be found by taking functional derivatives of $\mathcal{F}$ with respect to each distribution, $Q(\cdot)$. The forms for these are given in Section 3; in the case of $Q(\mathbf{x}_{1:T})$, there exists an efficient algorithm, the Kalman smoother, for finding the expected statistics of the hidden state in time $\mathcal{O}(T)$. The smoother is also known as the forward-backward algorithm for Linear Dynamical Systems, and is similar in spirit to the E-step in the Baum-Welch algorithm for HMMs. In the following section we rederive the filter (forward) and smoother (backward) recursions, and then incorporate the variational methodology in to these results to obtain $Q(\mathbf{x}_{1:T})$ in the Bayesian framework.

2 The forward-backward algorithm

In the standard point-parameter linear-Gaussian dynamical system, given the settings of the parameters, the hidden state posterior is jointly Gaussian over the time steps. Reassuringly, when we differentiate $\mathcal{F}$ with respect to $Q(\mathbf{x}_{1:T})$, the variational posterior for $\mathbf{x}_{1:T}$ is also Gaussian:

$$\begin{aligned}
\ln Q(\mathbf{x}_{1:T}) &= -\ln Z + \langle \ln P(A, B, C, D, \boldsymbol{\rho}, \mathbf{x}_{1:T}, \mathbf{y}_{1:T}) \rangle_{Q(A,B,C,D,\boldsymbol{\rho})}\\
&= -\ln Z' + \langle \ln P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T}|A, B, C, D, \boldsymbol{\rho}) \rangle_{Q(A,B,C,D,\boldsymbol{\rho})} \qquad (7)
\end{aligned}$$

$$\text{where } Z' = \int d\mathbf{x}_{1:T}\; \exp\left[\langle \ln P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T}|A, B, C, D, \boldsymbol{\rho}) \rangle_{Q(A,B,C,D,\boldsymbol{\rho})} \right]\ . \qquad (8)$$

In this expression, the expectations w.r.t the approximate parameter posteriors are performed on the *logarithm* of the joint likelihood and, even though this leaves the coefficients on the $\mathbf{x}_t$ terms in a somewhat

unorthodox state, the approximate posterior for $\mathbf{x}_{1:T}$ is still Gaussian. We can therefore use an algorithm very similar indeed to the Kalman smoother for inference of the hidden states' sufficient stastistics (the E-like step). However we can no longer plug in parameters to the filter and smoother, but have to work with the parameter sufficient statistics throughout the implementation.

The following derivations take us through the foward and backward recursions, without using simplifying steps such as the matrix inversion lemma (see Appendix B), which would invalidate a Bayesian approach. For the time being we will set aside the Bayesian implementational details and concentrate on the derivations. We will incorporate the Bayesian scheme in Section 3.

2.1 Filter: forward recursion

$$\begin{aligned}
\alpha_t(\mathbf{x}_t) \equiv P(\mathbf{x}_t|\mathbf{y}_{1:t}) &= \int d\mathbf{x}_{t-1}\; P(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})\, P(\mathbf{x}_t|\mathbf{x}_{t-1})\, P(\mathbf{y}_t|\mathbf{x}_t) \cdot /P(\mathbf{y}_t|\mathbf{y}_{1:t-1}) && (9)\\
&= \frac{1}{\zeta_t}\int d\mathbf{x}_{t-1}\; \alpha_{t-1}(\mathbf{x}_{t-1})\, P(\mathbf{x}_t|\mathbf{x}_{t-1})\, P(\mathbf{y}_t|\mathbf{x}_t) && (10)\\
&= \frac{1}{\zeta_t}\int d\mathbf{x}_{t-1}\; \mathcal{N}(\mathbf{x}_{t-1};\boldsymbol{\mu}_{t-1},\Sigma_{t-1})\, \mathcal{N}(\mathbf{x}_t;A\mathbf{x}_{t-1},E)\, \mathcal{N}(\mathbf{y}_t;C\mathbf{x}_t,R) && (11)\\
&= \mathcal{N}(\mathbf{x}_t;\boldsymbol{\mu}_t,\Sigma_t) && (12)
\end{aligned}$$

where we have defined $\zeta_t = P(\mathbf{y}_t|\mathbf{y}_{1:t-1})$ as the filtered output probability (we explain its usage a little later). The marginal probability for $\mathbf{x}_t$ is obtained by integrating out $\mathbf{x}_{t-1}$, by completing the square within the exponent of the integrand. The quadratic terms in $\mathbf{x}_{t-1}$ form the Gaussian $\mathrm{N}(\mathbf{x}_{t-1};\overline{\mathbf{x}}_{t-1},\Sigma^*_{t-1})$ with

$$\begin{aligned}
\Sigma^*_{t-1} &= \left(\Sigma^{-1}_{t-1} + A^\top E^{-1} A\right)^{-1} && (13)\\
\overline{\mathbf{x}}_{t-1} &= \Sigma^*_{t-1}\left[\Sigma^{-1}_{t-1}\boldsymbol{\mu}_{t-1} + A^\top E^{-1}\mathbf{x}_t\right]\ . && (14)
\end{aligned}$$

Marginalising out $\mathbf{x}_{t-1}$ gives the filtered estimates of the mean and covariance of the hidden state as

$$\begin{aligned}
\Sigma_t &= \left[E^{-1} + C^\top R^{-1} C - E^{-1} A\left(\Sigma^{-1}_{t-1} + A^\top E^{-1} A\right)^{-1} A^\top E^{-1}\right]^{-1} && (15)\\
\boldsymbol{\mu}_t &= \Sigma_t\left[C^\top R^{-1}\mathbf{y}_t + E^{-1} A\left(\Sigma^{-1}_{t-1} + A^\top E^{-1} A\right)^{-1}\Sigma^{-1}_{t-1}\boldsymbol{\mu}_{t-1}\right] && (16)
\end{aligned}$$

or rewritten as

$$\begin{aligned}
\Sigma_t &= \left[E^{-1} + C^\top R^{-1} C - E^{-1} A\Sigma^*_{t-1} A^\top E^{-1}\right]^{-1} && (17)\\
\boldsymbol{\mu}_t &= \Sigma_t\left[C^\top R^{-1}\mathbf{y}_t + E^{-1} A\Sigma^*_{t-1}\Sigma^{-1}_{t-1}\boldsymbol{\mu}_{t-1}\right]\ . && (18)
\end{aligned}$$

At each step, the normalising constant obtained as the denominator in (9), ζ_t, contributes to the calculation of the likelihood of the data

$$\begin{aligned}
P(\mathbf{y}_{1:T}) &= P(\mathbf{y}_1)P(\mathbf{y}_2|\mathbf{y}_1)\ldots P(\mathbf{y}_t|\mathbf{y}_{1:t-1})\ldots P(\mathbf{y}_T|\mathbf{y}_{1:T-1}) && (19)\\
&= P(\mathbf{y}_1)\prod_{t=2}^{T} P(\mathbf{y}_t|\mathbf{y}_{1:t-1}) \equiv \prod_{t=1}^{T}\zeta_t\ . && (20)
\end{aligned}$$

It is not difficult to show that ζ_t is $\mathrm{N}(\mathbf{y}_t;\boldsymbol{\varpi}_t,\varsigma_t)$ with

$$\begin{aligned}
\varsigma_t &= \left(R^{-1} - R^{-1} C\Sigma_t C^\top R^{-1}\right)^{-1} && (21)\\
\boldsymbol{\varpi}_t &= \varsigma_t R^{-1} C\Sigma_t E^{-1} A\Sigma^*_{t-1}\Sigma^{-1}_{t-1}\boldsymbol{\mu}_{t-1}\ . && (22)
\end{aligned}$$

With these distributions at hand we can evaluate the likelihood of each observation $\mathbf{y}_t$ as the filter progresses, whilst taking into account previous observed data automatically. These distributions must obviously be recalculated to evaluate the likelihood of a different observation time series (a test set), unless we want to clamp the hidden state from the training data. Further simplification of any of these results using the matrix inversion lemma is not undertaken (with the exception of (21)), as later we will be rewriting these equations with the necessary averages in place, and certain averages cannot be inverted simply.

2.2 Smoother: backward recursion

The Kalman smoother makes use of the backward recursion

$$\gamma_t(\mathbf{x}_t) \equiv P(\mathbf{x}_t|\mathbf{y}_{1:T}) = \int d\mathbf{x}_{t+1} \left[\frac{P(\mathbf{x}_t|\mathbf{y}_{1:t})P(\mathbf{x}_{t+1}|\mathbf{x}_t)}{\int d\mathbf{x}'_t\, P(\mathbf{x}'_t|\mathbf{y}_{1:t})P(\mathbf{x}_{t+1}|\mathbf{x}'_t)}\right] P(\mathbf{x}_{t+1}|\mathbf{y}_{1:T}) \tag{23}$$

$$\gamma_t(\mathbf{x}_t) = \int d\mathbf{x}_{t+1} \left[\frac{\alpha_t(\mathbf{x}_t)P(\mathbf{x}_{t+1}|\mathbf{x}_t)}{\int d\mathbf{x}'_t\, \alpha_t(\mathbf{x}'_t)P(\mathbf{x}_{t+1}|\mathbf{x}'_t)}\right] \gamma_{t+1}(\mathbf{x}_{t+1}) \ . \tag{24}$$

Once the integral in the denominator is done, the terms in the exponent of the integrand are (multiplied by -2)

$$(\mathbf{x}_t - \boldsymbol{\mu}_t)^\top \Sigma_t^{-1}(\mathbf{x}_t - \boldsymbol{\mu}_t) + (\mathbf{x}_{t+1} - A\mathbf{x}_t)^\top E^{-1}(\mathbf{x}_{t+1} - A\mathbf{x}_t) + (\mathbf{x}_{t+1} - \boldsymbol{\eta}_{t+1})\Psi_{t+1}^{-1}(\mathbf{x}_{t+1} - \boldsymbol{\eta}_{t+1}) + (\Sigma_t^{-1}\boldsymbol{\mu}_t + A^\top E^{-1}\mathbf{x}_{t+1})^\top \left(\Sigma_t^{-1} + A^\top E^{-1} A\right)^{-1} (\Sigma_t^{-1}\boldsymbol{\mu}_t + A^\top E^{-1}\mathbf{x}_{t+1}) - \mathbf{x}_{t+1}^\top E^{-1}\mathbf{x}_{t+1} \ . \tag{25}$$

Integrating out $\mathbf{x}_{t+1}$ yields Gaussian distributions for the smoothed estimates of the hidden state at each time step

$$\text{defining} \quad \Sigma_t^* = \left(\Sigma_t^{-1} + A^\top E^{-1} A\right)^{-1} \quad \text{and} \quad K_t = \left(\Psi_{t+1}^{-1} + E^{-1} A \Sigma_t^* A^\top E^{-1}\right)^{-1} \tag{26}$$

$$\Psi_t = \left[\Sigma_t^{*-1} - A^\top E^{-1} K_t E^{-1} A\right]^{-1} \tag{27}$$

$$\boldsymbol{\eta}_t = \Psi_t \left[\Sigma_t^{-1}\boldsymbol{\mu}_t + A^\top E^{-1} K_t \left(\Psi_{t+1}^{-1}\boldsymbol{\eta}_{t+1} - E^{-1} A \Sigma_t^* \Sigma_t^{-1}\boldsymbol{\mu}_t\right)\right] \ . \tag{28}$$

This result differs from the traditional β-pass, in that it does not require any more information of the data $\mathbf{y}_{1:T}$ — in essence all necessary information carried in the observed data has already been assimilated into the $\alpha_{1:T}$ messages in the filter's forward recursion. This method is useful in online scenarios because we can throw away past observations once they have been filtered.

2.3 Cross-time covariance

For learning the parameters of the transition matrix, or equivalently for the calculation of the variational posterior for $Q(A)$, we require the cross-time covariance. This can best be calculated by looking at the inverse covariance terms in the joint in (25), and making use of Schur complements. Prior to the integration over $\mathbf{x}_{t+1}$ in (23) we have

$$P(\mathbf{x}_t, \mathbf{x}_{t+1}|\mathbf{y}_{1:T}) = \frac{P(\mathbf{x}_t|\mathbf{y}_{1:t})P(\mathbf{x}_{t+1}|\mathbf{x}_t)}{\int d\mathbf{x}'_t\, P(\mathbf{x}'_t|\mathbf{y}_{1:t})P(\mathbf{x}'_{t+1}|\mathbf{x}_t)} P(\mathbf{x}_{t+1}|\mathbf{y}_{1:T}) \ . \tag{29}$$

Terms that couple $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$ are best represented with the product

$$-\ln P(\mathbf{x}_t, \mathbf{x}_{t+1}|\mathbf{y}_{1:T}) \stackrel{+}{=} \begin{pmatrix} \mathbf{x}_t^\top & \mathbf{x}_{t+1}^\top \end{pmatrix} \begin{pmatrix} \Sigma_t^{*-1} & -A^\top E^{-1} \\ -E^{-1} A & K_t^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{x}_t \\ \mathbf{x}_{t+1} \end{pmatrix} . \tag{30}$$

Defining $\Upsilon_{t,t+1}$ to be the cross-covariance between the hidden states at times t and $t+1$, we can make use of Schur complements in (93) and (94) to give

$$\begin{aligned}
\Upsilon_{t,t+1} &\equiv \langle (\mathbf{x}_t - \boldsymbol{\eta}_t)(\mathbf{x}_{t+1} - \boldsymbol{\eta}_{t+1})^\top \rangle_{Q(\mathbf{x}_{1:T})} && (31)\\
&= \Sigma_t^* A^\top E^{-1} \Psi_{t+1} \qquad \text{using (94)} && (32)\\
&= \Psi_t A^\top E^{-1} K_t \qquad \text{using (93)} && (33)
\end{aligned}$$

The second expression is the standard result, the first an equivalent expression, together giving a recursion relation for Ψ_t. Up to this point we have followed the standard procedures to derive the forward and backward elements of the smoother, only stopping short of using matrix inversion lemmas.

2.4 Required hidden state sufficient statistics

We will need to use the results of inference over the hidden states to update the distributions over parameters. The hidden state sufficient statistics and some related quantities are defined as

$$\begin{aligned}
S &= \sum_{t=1}^{T-1} \langle \mathbf{x}_t \mathbf{x}_{t+1}^\top \rangle = \sum_{t=1}^{T-1} \left(\Upsilon_{t,t+1} + \boldsymbol{\eta}_t \boldsymbol{\eta}_{t+1}^\top \right) && (34)\\
W &= \sum_{t=1}^{T-1} \langle \mathbf{x}_t \mathbf{x}_t^\top \rangle = \sum_{t=1}^{T-1} \left(\Psi_t + \boldsymbol{\eta}_t \boldsymbol{\eta}_t^\top \right) && (35)\\
W' &= \sum_{t=1}^{T} \langle \mathbf{x}_t \mathbf{x}_t^\top \rangle = \sum_{t=1}^{T} \left(\Psi_t + \boldsymbol{\eta}_t \boldsymbol{\eta}_t^\top \right) && (36)\\
M &= \sum_{t=1}^{T-1} \boldsymbol{\eta}_t \mathbf{u}_{t+1}^\top && (37)\\
U_\mu &= \sum_{t=1}^{T} \mathbf{u}_t \boldsymbol{\eta}_t^\top && (38)\\
\tilde{M} &= U_\mu - \mathbf{u}_1 \boldsymbol{\eta}_1^\top + \mathbf{u}_1 \left(\boldsymbol{\eta}_1 - \boldsymbol{\pi}_0 \right)^\top \Sigma_0^{-1} && (39)\\
\tilde{Y} &= \sum_{t=1}^{T} \boldsymbol{\eta}_t \mathbf{y}_t^\top . && (40)
\end{aligned}$$

3 Bayesian implementation

In the last term of (6), the log-likelihood terms are averaged over distributions of the variational posteriors. Therefore in the E-step of learning, the effective parameters are just their expectations as they appear in the exponent of this likelihood term.

3.1 Parameter priors

In the analysis so far we have not yet needed to specify our prior distributions over the parameters, which appear in the full joint of (3), and consequently in our evidence lower bound $\mathcal{F}$ of (6). The forms we

choose are *conjugate*, in that they have the same functional form as the likelihood:

$$
\begin{aligned}
A &: A_i \sim \mathrm{N}(A_i; \mathbf{0}, \mathrm{diag}\,(\boldsymbol{\alpha})) && (41)\\
B &: B_i \sim \mathrm{N}(B_i; \mathbf{0}, \mathrm{diag}\,(\boldsymbol{\beta})) && (42)\\
C &: C_i \sim \mathrm{N}(C_i; \mathbf{0}, \rho_i\ \mathrm{diag}\,(\boldsymbol{\gamma})) && (43)\\
D &: D_i \sim \mathrm{N}(D_i; \mathbf{0}, \rho_i\ \mathrm{diag}\,(\boldsymbol{\delta})) && (44)\\
\boldsymbol{\rho} &: \rho_i \sim \mathrm{Ga}(\rho_i; a, b) && (45)
\end{aligned}
$$

where the single subscript index i on a matrix denotes the transpose of its i^{th} *row*, i.e. a column vector. The effect of the Gaussian priors on the transition (A) and output (C) matrices will be to perform automatic relevance determination (ARD) on the hidden states; i.e. during learning only those hidden dimensions that are required to model the data will remain active (with corresponding non-zero weights). If a dimension is not required then to increase the likelihood its emanating weights to both the next hidden state (A) and to the data (C) will move towards the zero mean of the prior, and they can be removed from the model. Similarly the Gaussian priors on the input matrices (B) and (D) should prune those inputs that are irrevelant to predicting the data.

Below we provide pseudocode for an implementation of Bayesian Linear Dynamical Systems, which uses the variational Kalman smoother as a subroutine for the E-step of inference. As the parameters have distributions rather than being point estimates, we calculate these distributions' sufficient statistics. The $\langle\cdot\rangle$ notation denotes expectation under the relevant variational posterior(s). The parameter expectations (sufficient statistics) are given below. As mentioned above, without loss of generality we can set the hidden state noise E to the identity.

3.2 Parameter posterior approximations

Given the approximating factorisation of the posterior distribution over hidden variables and parameters, the approximate posterior over the parameters can be factorised without further assumption into $Q(A, B, C, D, \boldsymbol{\rho}) = \prod_{j=1}^{k} Q(B_j) Q(A_j|B_j) \prod_{i=1}^{p} Q(\rho_i) Q(D_i|\rho_i) Q(C_i|\rho_i, D_i)$. Note there is no need for the prior on C_i to be a function of D_i, even though the posterior distribution factorisation involves this dependence.

$$
\begin{aligned}
Q(B_i) &= \mathrm{N}\left(B_i;\ \Sigma^B \dot{M}_i,\ \Sigma^B_i\right) && (46)\\
Q(A_i|B_i) &= \mathrm{N}\left(A_i;\ \left[S^\top - B^\top M\right] \Sigma^A,\ \Sigma^A\right) && (47)\\
Q(\rho_i) &= \mathrm{Ga}\left(\rho_i;\ a + \frac{T}{2},\ b + \frac{1}{2} G_i\right) && (48)\\
Q(D_i|\rho_i) &= \mathrm{N}\left(D_i;\ \ddot{Y}^\top \Sigma^D,\ \rho_i^{-1} \Sigma^D\right) && (49)\\
Q(C_i|\rho_i, D_i) &= \mathrm{N}\left(C_i;\ \left[\tilde{Y}^\top - D^\top U_\mu\right] \Sigma^C,\ \rho_i^{-1} \Sigma^C\right) && (50)
\end{aligned}
$$

where

$$\hat{U} = \sum_{t=1}^{T} \mathbf{u}_t \mathbf{u}_t^\top \tag{51}$$

$$\tilde{U} = \hat{U} + \mathbf{u}_1 \left(V_0 - 1\right) \mathbf{u}_1^\top \tag{52}$$

$$\Sigma^{A^{-1}} = \mathrm{diag}\left(\boldsymbol{\alpha}\right) + W \tag{53}$$

$$\Sigma^{B^{-1}} = \tilde{U} + \mathrm{diag}\left(\boldsymbol{\beta}\right) - M^\top \Sigma^A M \tag{54}$$

$$\Sigma^{C^{-1}} = \mathrm{diag}\left(\boldsymbol{\gamma}\right) + W' \tag{55}$$

$$\Sigma^{D^{-1}} = \hat{U} + \mathrm{diag}\left(\boldsymbol{\delta}\right) - U_\mu \Sigma^C U_\mu^\top \tag{56}$$

$$\hat{Y} = \sum_{t=1}^{T} \mathbf{u}_t \mathbf{y}_t^\top \tag{57}$$

$$\dot{Y} = \sum_{t=1}^{T} \mathbf{y}_t \mathbf{y}_t^\top \tag{58}$$

and

$$\ddot{Y} = \hat{Y} - U_\mu \Sigma^C \tilde{Y} \tag{59}$$

$$\breve{Y} = \tilde{Y} - U_\mu^\top \Sigma^D \ddot{Y} \tag{60}$$

$$\dot{M} = \tilde{M} - M^\top \Sigma^A S \tag{61}$$

$$G_i = \left[\dot{Y} - \tilde{Y}^\top \Sigma^C \tilde{Y} - \ddot{Y}^\top \Sigma^D \ddot{Y}\right]_{ii} \tag{62}$$

3.3 Required parameter sufficient statistics

We require the following parameter sufficient statistics for the implementation of the variational Kalman smoother:

$$\langle A \rangle = \left[S^\top - \dot{M}^\top \Sigma^B M^\top\right] \Sigma^A \tag{63}$$

$$\langle A^\top A \rangle = k\Sigma^A + \Sigma^A \Big[SS^\top - S\dot{M}^\top \Sigma^B M^\top - M\Sigma^B \dot{M} S^\top + M\Sigma^B M^\top + M\Sigma^B \dot{M}\dot{M}^\top \Sigma^B M^\top\Big] \Sigma^A \tag{64}$$

$$\langle B \rangle = \dot{M}^\top \Sigma^B \tag{65}$$

$$\langle A^\top B \rangle = \Sigma^A \left[S\langle B \rangle - M\left\{k\Sigma^B + \langle B \rangle^\top \langle B \rangle\right\}\right] \tag{66}$$

$$\langle R^{-1} \rangle = \mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right) \tag{67}$$

$$\langle C^\top R^{-1} C \rangle = p\Sigma^C + \Sigma^C \left(pU_\mu^\top \Sigma^D U_\mu + \breve{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\breve{Y}^\top\right) \Sigma^C \tag{68}$$

$$\langle R^{-1} C \rangle = \mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right) \breve{Y}^\top \Sigma^C \tag{69}$$

$$\langle C^\top R^{-1} D \rangle = \Sigma^C \left(\tilde{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\ddot{Y}^\top - pU_\mu^\top - U_\mu^\top \Sigma^D \ddot{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\ddot{Y}^\top\right) \Sigma^D \tag{70}$$

$$\langle R^{-1} D \rangle = \mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right) \ddot{Y}^\top \Sigma^D \tag{71}$$

$$\langle \rho_i \rangle = \overline{\rho_i} = \frac{a_{\boldsymbol{\rho}} + T/2}{b_{\boldsymbol{\rho}} + G_i/2} \tag{72}$$

$$\langle \ln \rho_i \rangle = \overline{\ln \rho_i} = \psi(a_{\boldsymbol{\rho}} + T/2) - \ln(b_{\boldsymbol{\rho}} + G_i/2) \tag{73}$$

The following are not required but are useful to look at

$$\langle C \rangle = \left[\tilde{Y}^\top - \ddot{Y}^\top \Sigma^D U_\mu\right] \Sigma^D \tag{74}$$

$$\langle D \rangle = \ddot{Y}^\top \Sigma^D \tag{75}$$

3.4 Hyperparameter updates

The hyperparameters $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}, a, b\}$ and the prior parameters, Σ_0 and $\boldsymbol{\mu}_0$, as point estimates can be updated so as to maximise the lower bound on the evidence (6). The following four updates are only valid if we have an isotropic variance prior on the hidden state, such that $\Sigma_j^B = \Sigma^B$ for every j.

$$\alpha_j^{-1} = \Sigma_{jj}^A + \frac{1}{k}\Sigma_j^{A\top}\left[SS^\top - 2M\Sigma^B\dot{M}S^\top + kM\Sigma^BM^\top + M\Sigma^B\dot{M}\dot{M}^\top\Sigma^BM^\top\right]\Sigma_j^A \tag{76}$$

$$\beta_j^{-1} = \Sigma_{jj}^B + \frac{1}{k}\Sigma_j^{B\top}\dot{M}\dot{M}^\top\Sigma_j^B \tag{77}$$

$$\gamma_j^{-1} = \frac{1}{p}\left\{p\Sigma_{jj}^C + \Sigma_j^{C\top}\left[\tilde{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\tilde{Y}^\top - 2\tilde{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\ddot{Y}^\top\Sigma^DU_\mu + pU_\mu^\top\Sigma^DU_\mu + U_\mu^\top\Sigma^D\ddot{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\ddot{Y}^\top\Sigma^DU_\mu\right]\Sigma_j^C\right\} \tag{78}$$

$$\delta_j^{-1} = \frac{1}{p}\left\{p\Sigma_{jj}^D + \Sigma_j^{D\top}\ddot{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\ddot{Y}^\top\Sigma_j^D\right\} \tag{79}$$

Provided we have calculated the expectations of B and D, simpler forms for the hyperparameter updates are given by

$$\alpha_j^{-1} = \frac{1}{k}\left[k\Sigma^A + \Sigma^A\left[SS^\top - 2M\langle B\rangle^\top S^\top + M\left\{k\Sigma^B + \langle B\rangle^\top\langle B\rangle\right\}M^\top\right]\Sigma^A\right]_{jj} \tag{80}$$

$$\beta_j^{-1} = \frac{1}{k}\left[k\Sigma^B + \langle B\rangle^\top\langle B\rangle\right]_{jj} \tag{81}$$

$$\gamma_j^{-1} = \frac{1}{p}\left[p\Sigma^C + \Sigma^C\left[\tilde{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\tilde{Y}^\top - 2\tilde{Y}\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\langle D\rangle U_\mu + pU_\mu^\top\Sigma^DU_\mu + U_\mu^\top\langle D\rangle^\top\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\langle D\rangle U_\mu\right]\Sigma^C\right]_{jj} \tag{82}$$

$$\delta_j^{-1} = \frac{1}{p}\left[p\Sigma^D + \langle D\rangle^\top\mathrm{diag}\left(\overline{\boldsymbol{\rho}}\right)\langle D\rangle\right]_{jj} \tag{83}$$

where again the subscript indexing $\Sigma_j^{\cdot}$ denotes the j^{th} column of $\Sigma^{\cdot}$, or equivalently the transpose of its j^{th} row, and $\Sigma_{jj}^{\cdot}$ denotes the $(j,j)^{\text{th}}$ element of $\Sigma^{\cdot}$. The hyperparameters a and b are set to the fixed point of the equations

$$\psi(a) = \ln b + \frac{1}{p}\sum_{i=1}^{p}\overline{\ln\rho_i} \tag{84}$$

$$\frac{1}{b} = \frac{1}{pa}\sum_{i=1}^{p}\overline{\rho_i} \tag{85}$$

where $\psi(x) = \partial/\partial x \ln\Gamma(x)$ is the *digamma* function. These fixed point equations can be easily solved using gradient following techniques (Newton's method etc.) in just a few iterations.

3.5 Calculation of F

Some of the nested integrals in (6) can be removed as in most cases the inner Kullback-Leibler divergence is not a function of the outer integration variables, thanks partly to the parameter priors having conjugate forms. This makes the first 5 terms simpler evaluations, leaving

$$\mathcal{F} = -\mathrm{KL}(B) - \mathrm{KL}(A|B) - \mathrm{KL}(\boldsymbol{\rho}) - \mathrm{KL}(D|\boldsymbol{\rho}) - \mathrm{KL}(C|\boldsymbol{\rho}, D) - \int d\mathbf{x}_{1:T}Q(\mathbf{x}_{1:T})\ln Q(\mathbf{x}_{1:T}) + \langle\ln P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T}|A, B, C, D, \boldsymbol{\rho})\rangle_{Q(A,B,C,D,\boldsymbol{\rho})Q(\mathbf{x}_{1:T})} \tag{86}$$

$$= -\mathrm{KL}(B) - \mathrm{KL}(A) - \mathrm{KL}(\boldsymbol{\rho}) - \mathrm{KL}(D) - \mathrm{KL}(C) + H(\mathbf{x}_{1:T}) + \langle\ln P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T}|A, B, C, D, \boldsymbol{\rho})\rangle_{Q(A,B,C,D,\boldsymbol{\rho})Q(\mathbf{x}_{1:T})} \tag{87}$$

where $\mathrm{KL}(s)$ is the Kullback-Leibler divergence between the variational posterior and the prior distributions of variable s, and $H(s)$ is the entropy of the variational posterior over s. Calculating $\mathcal{F}$ at each iteration still looks problematic due to the entropy term of the hidden state, $H(\mathbf{x}_{1:T})$ in (87). Fortunately, straight after a variational E-Step, we know the form of $Q(\mathbf{x}_{1:T})$ from (7). This gives

$$\begin{aligned} H(\mathbf{x}_{1:T}) &= -\int d\mathbf{x}_{1:T}\, Q(\mathbf{x}_{1:T}) \ln Q(\mathbf{x}_{1:T}) && (88)\\ &= -\int d\mathbf{x}_{1:T}\, Q(\mathbf{x}_{1:T}) \left[-\ln Z' + \langle \ln P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T} | A, B, C, D, \boldsymbol{\rho}) \rangle_{Q(A,B,C,D,\boldsymbol{\rho})} \right] && (89)\\ &= \ln Z' - \langle \ln P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T} | A, B, C, D, \boldsymbol{\rho}) \rangle_{Q(A,B,C,D,\boldsymbol{\rho}) Q(\mathbf{x}_{1:T})} \ . && (90) \end{aligned}$$

Substituting this into (87) cancels both equations' last terms to yield a simple expression for the lower bound

$$\mathcal{F} = -\mathrm{KL}(B) - \mathrm{KL}(A) - \mathrm{KL}(\boldsymbol{\rho}) - \mathrm{KL}(D) - \mathrm{KL}(C) + \ln Z' \ . \quad (91)$$

We still have to be able to evaluate the partition function for $\mathbf{x}_{1:T}$, Z'. It is not as complicated as the integral in equation (8) purports: at least in the point-parameter scenario we showed that this was just $\prod_{t=1}^{T} \zeta_t$, (see (20)). With some care we can translate the required calculations from (21) and (22) over into the Bayesian scheme; the expressions that evaluate this Bayesian ζ_t at each time step are given in the variational filter pseudocode; to summarise each ζ_t is a slightly modified Gaussian. In the Matlab code accompanying this report, $\mathcal{F}$ is calculated after the variational E-Step, at which point equation (91) is correct. To be precise $\mathcal{F}$ is actually computed immediately after the filter (foward pass). The KL divergence terms are also surprisingly simple to calculate: each row of A contributes the same to $\mathrm{KL}(A)$; $\mathrm{KL}(C, \boldsymbol{\rho})$ factorises into $\mathrm{KL}(C) + \mathrm{KL}(\boldsymbol{\rho})$ as a result of the divergence between two Gaussian densities being only a function of the *ratio* of covariance determinants, thus cancelling the dependence of $\mathrm{KL}(C|\boldsymbol{\rho})$ on $\boldsymbol{\rho}$ (this requires the *prior* on C to have $\boldsymbol{\rho}$ dependence).

4 Extensions and further work

Unfortunately, as it currently stands, the Bayesian scheme is not as complete as it could be. It is true that for a proposed hidden state dimension the algorithm above *does* perform ARD and can reveal an appropriate hidden state-space dimensionality with some success [2]. However, we cannot at this stage compare two models' lower bounds on the evidence. The reason for this is that we have not yet integrated out all those variables whose cardinality increases with model complexity. The ARD parameters, $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$, are such variables and so we introduce prior distributions and variational posteriors over these variables. Incorporating this further level in the hierarchy requires changes and additions to the terms in $\mathcal{F}$ (6), yielding the following fully Bayesian $\mathcal{F}$:

$$\begin{aligned} \mathcal{F} = &-\int dA\, Q(A) \ln Q(A) + \int d\boldsymbol{\alpha}\, Q(\boldsymbol{\alpha}) \left[\int dA\, Q(A) \ln P(A|\boldsymbol{\alpha}) - \ln \frac{Q(\boldsymbol{\alpha})}{P(\boldsymbol{\alpha}|a_{\boldsymbol{\alpha}}, b_{\boldsymbol{\alpha}})} \right] \\ &- \int d\boldsymbol{\rho}\, Q(\boldsymbol{\rho}) \ln \frac{Q(\boldsymbol{\rho})}{P(\boldsymbol{\rho}|a_{\boldsymbol{\rho}}, b_{\boldsymbol{\rho}})} \\ &- \int d\boldsymbol{\rho}\, Q(\boldsymbol{\rho}) \int dC\, Q(C|\boldsymbol{\rho}) \left[\ln Q(C|\boldsymbol{\rho}) - \int d\boldsymbol{\gamma}\, Q(\boldsymbol{\gamma}) \ln P(C|\boldsymbol{\gamma}, \boldsymbol{\rho}) \right] \\ &- \int d\boldsymbol{\gamma}\, Q(\boldsymbol{\gamma}) \ln \frac{Q(\boldsymbol{\gamma})}{P(\boldsymbol{\gamma}|a_{\boldsymbol{\gamma}}, b_{\boldsymbol{\gamma}})} \\ &- \int d\mathbf{x}_{1:T}\, Q(\mathbf{x}_{1:T}) \ln Q(\mathbf{x}_{1:T}) \\ &+ \int dA\, Q(A) \int dC\, d\boldsymbol{\rho}\, Q(C, \boldsymbol{\rho}) \int d\mathbf{x}_{1:T}\, Q(\mathbf{x}_{1:T}) \ln P(\mathbf{x}_{1:T}, \mathbf{y}_{1:T} | A, C, \boldsymbol{\rho}) \ . \end{aligned} \quad (92)$$

To create this $\mathcal{F}$ from (6) we have just placed integrals around any expressions involving $\boldsymbol{\alpha}$ or $\boldsymbol{\gamma}$, and also introduced KL-divergence penalties for these distributions. All the existing updates remain essentially the same, except that we can now also optimise the hyperparameters $a_{\boldsymbol{\alpha}}$, $b_{\boldsymbol{\alpha}}$, $a_{\boldsymbol{\gamma}}$ and $b_{\boldsymbol{\gamma}}$ as well. These updates will follow the usual moment matching theme: the variational posteriors $Q(\boldsymbol{\alpha})$ and $Q(\boldsymbol{\gamma})$ are products of Gamma distributions with their first and first logarithmic moments matching those of the entries in A and C respectively. With this cost function we can be sure that we are penalising over-parameterisation of the state-space with more than just an ARD scheme. The authors have yet to implement this "more Bayesian" model, for the moment simply leaving it to rest as theoretically desirable.

There are a few other areas and aspects of the model that can be improved, at no serious cost to the methodology. One of these is to incorporate inputs into the system, as an autonomous linear dynamical system is not nearly as representationally powerful as a driven one. The dynamics would then obey $\mathbf{x}_t = A\mathbf{x}_{t-1} + B\mathbf{u}_t + \mathbf{w}_t$, and $\mathbf{y}_t = C\mathbf{x}_t + D\mathbf{u}_t + \mathbf{v}_t$, where $\mathbf{u}_t$ is the user input to the system at time t. By augmenting the inputs with a constant bias the dynamics can then be displaced if need be. Similar priors could be placed on the entries of B and D, and even ARD for these matrix elements might be meaningful by showing which inputs to the system are relevant to predicting the output and which are noise. This extension is quite trivial, and will be included in a revision of this note shortly.

Additionally it is possible for the 1st order Markov model to emulate the dynamics of a higher order model by feeding back concatenated observed data $\mathbf{y}_{t-n:t-1}$ into the future input u_t. By generalising this and using an ARD construction we should be able to shed light on the order of the system that actually generated the data.

Tests with gene expression micro-array data are anticipated.

5 Acknowledgements

We wish to thank Ali Taylan Cemgil for helpful comments on an intial draft of this report, discussed at the NIPS 2000 workshop *Real-time modeling for complex learning tasks*.

References

[1] H. Attias. A variational Bayesian framework for graphical models. In S. A. Solla, T. K. Leen, and K. Müller, editors, *Advances in Neural Information Processing Systems 12*, Cambridge, MA, 2000. MIT Press.

[2] Z. Ghahramani and M. J. Beal. Propagation algorithms for variational Bayesian learning. In *Adv. Neur. Inf. Proc. Sys. 13*, Cambridge, MA, 2001. MIT Press.

6 Appendices

A Schur complements and inverting partitioned matrices

Inverting partitioned matrices can be quite difficult. If A is of 2 by 2 block form, we can use Schur complements to obtain the following results for the partitioned inverse, and the determinant of A in terms of its constituents.

Pseudocode: Bayesian linear dynamical systems

Initialise parameters. Initialise hidden variables and state priors Σ_0, $\boldsymbol{\mu}_0$

for $n = 1 : max\_its$

Variational M-Step

- parameter $Q(A)$ suff. stats.
  $S = \sum_{t=1}^{T-1} \langle \mathbf{x}_t \mathbf{x}_{t+1}^\top \rangle = \sum_{t=1}^{T-1} \left(\Upsilon_{t,t+1} + \boldsymbol{\eta}_t \boldsymbol{\eta}_{t+1}^\top \right)$
  $W = \sum_{t=1}^{T-1} \langle \mathbf{x}_t \mathbf{x}_t^\top \rangle = \sum_{t=1}^{T-1} \left(\Psi_t + \boldsymbol{\eta}_t \boldsymbol{\eta}_t^\top \right)$
- parameter $Q(B)$
  update
- parameter $Q(\boldsymbol{\rho})$ suff. stats. $\forall\ i$
  $\mathbf{G} : G_i = \sum_{t=1}^{T} \mathbf{y}_{ti}^2 - U_i^\top (\mathrm{diag}\,(\boldsymbol{\gamma}) + W')^{-1} U_i$
- parameter $Q(C|\boldsymbol{\rho})$ suff. stats.
  $W' = \sum_{t=1}^{T} \langle \mathbf{x}_t \mathbf{x}_t^\top \rangle = \sum_{t=1}^{T} \left(\Psi_t + \boldsymbol{\eta}_t \boldsymbol{\eta}_t^\top \right)$
  $U = \sum_{t=1}^{T} \boldsymbol{\eta}_t \mathbf{y}_t^\top$
- parameter $Q(D)$
  update
- calculate parameter sufficient statistics
  $\{pss\} \leftarrow$ parameter suff. stats. (see (**??**) - (**??**))

Variational E-Step

- hidden variables $Q(\mathbf{x}_{1:T})$ suff. stats.

$\{\boldsymbol{\eta}_{1:T}, \Psi_{1:T}, \mathcal{F}\} \leftarrow$ **variational Kalman smoother**$(\mathbf{y}_{1:T}, pss)$

Hyperparameters

- hyperparameters $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$, $\forall\ k$
  $\boldsymbol{\alpha} : \alpha_k = K / \langle A^\top A \rangle_{kk}$
  $\boldsymbol{\gamma} : \gamma_k = D / \langle C^\top \mathrm{diag}\,(\boldsymbol{\rho})\, C \rangle_{kk}$
- hyperparameters a and b, at fixed point of
  $\psi(a) = \ln b + \frac{1}{D} \sum_{i=1}^{D} \langle \ln \rho_i \rangle$
  $\frac{1}{b} = \frac{1}{aD} \sum_{i=1}^{D} \langle \rho_i \rangle$
- state priors
  $\Sigma_0 =$
  $\boldsymbol{\mu}_0 =$

end for

Pseudocode: variational Kalman smoother

- hidden variables $Q(\mathbf{x}_{1:T})$ suff. stats.

- Foward recursion
  $\Sigma_0^* = \left(\Sigma_0^{-1} + \langle A^\top A\rangle\right)^{-1}$
  for $t = 1 : T$
    $\Sigma_t = \left(I + \langle C^\top R^{-1} C\rangle - \langle A\rangle \Sigma_{t-1}^* \langle A\rangle^\top\right)^{-1}$
    $\boldsymbol{\mu}_t = \Sigma_t \left(\langle C^\top R^{-1}\rangle \mathbf{y}_t + \langle A\rangle \Sigma_{t-1}^* \Sigma_{t-1}^{-1} \boldsymbol{\mu}_{t-1}\right)$
    $\Sigma_t^* = \left(\Sigma_t^{-1} + \langle A^\top A\rangle\right)^{-1}$
    $\varsigma_t = \left(\langle R^{-1}\rangle - \langle R^{-1} C\rangle \Sigma_t \langle R^{-1} C\rangle^\top\right)^{-1}$
    $\boldsymbol{\varpi}_t = \varsigma_t \langle R^{-1} C\rangle \Sigma_t \langle A\rangle \Sigma_{t-1}^* \Sigma_{t-1}^{-1} \boldsymbol{\mu}_{t-1}$
  end for

Calculate $\mathcal{F}$

Set $\boldsymbol{\eta}_T = \boldsymbol{\mu}_T$ and $\Psi_T = \Sigma_T$

- Backward recursion
  for $t = (T-1) : 1$
    $K_t = \left(\Psi_{t+1}^{-1} + \langle A\rangle \Sigma_t^* \langle A\rangle^\top\right)^{-1}$
    $\Psi_t = \left(\Sigma_t^{*-1} - \langle A\rangle^\top K_t \langle A\rangle\right)^{-1}$
    $\Upsilon_{t,t+1} = \Psi_t \langle A\rangle^\top K_t$
    $\boldsymbol{\eta}_t = \Sigma_t^* \Sigma_t^{-1} \boldsymbol{\mu}_t + \Upsilon_{t,t+1} \Psi_{t+1}^{-1} \boldsymbol{\eta}_{t+1}$
  end for

Pseudocode: variational Kalman smoother with inputs $\mathbf{u}_{1:T}$

- hidden variables $Q(\mathbf{x}_{1:T})$ suff. stats.

- Foward recursion

  $\Sigma_0^* = \left(\Sigma_0^{-1} + \langle A^\top A\rangle\right)^{-1}$

  for $t = 1 : T$

  $\Sigma_t = \left(I + \langle C^\top R^{-1} C\rangle - \langle A\rangle \Sigma_{t-1}^* \langle A\rangle^\top\right)^{-1}$

  $\boldsymbol{\mu}_t = \Sigma_t \left[\langle C^\top R^{-1}\rangle \mathbf{y}_t + \langle A\rangle \Sigma_{t-1}^* \Sigma_{t-1}^{-1} \boldsymbol{\mu}_{t-1} + \left(\langle B\rangle - \langle A\rangle \Sigma_{t-1}^* \langle A^\top B\rangle - \langle C^\top R^{-1} D\rangle\right) \mathbf{u}_t\right]$

  $\Sigma_t^* = \left(\Sigma_t^{-1} + \langle A^\top A\rangle\right)^{-1}$

  $\varsigma_t = \left(\langle R^{-1}\rangle - \langle R^{-1} C\rangle \Sigma_t \langle R^{-1} C\rangle^\top\right)^{-1}$

  $\boldsymbol{\varpi}_t = \varsigma_t \left[\langle R^{-1} C\rangle \Sigma_t \langle A\rangle \Sigma_{t-1}^* \Sigma_{t-1}^{-1} \boldsymbol{\mu}_{t-1} + \left(\langle R^{-1} D\rangle + \langle R^{-1} C\rangle \Sigma_t \left\{\langle B\rangle - \langle C^\top R^{-1} D\rangle - \langle A\rangle \Sigma_{t-1}^* \langle A^\top B\rangle\right\}\right) \mathbf{u}_t\right]$

  end for

Calculate $\mathcal{F}$

Set $\boldsymbol{\eta}_T = \boldsymbol{\mu}_T$ and $\Psi_T = \Sigma_T$

- Backward recursion

  for $t = (T-1) : 1$

  $K_t = \left(\Psi_{t+1}^{-1} + \langle A\rangle \Sigma_t^* \langle A\rangle^\top\right)^{-1}$

  $\Psi_t = \left(\Sigma_t^{*-1} - \langle A\rangle^\top K_t \langle A\rangle\right)^{-1}$

  $\Upsilon_{t,t+1} = \Psi_t \langle A\rangle^\top K_t$

  $\boldsymbol{\eta}_t = \Sigma_t^* \left[\Sigma_t^{-1} \boldsymbol{\mu}_t - \langle A^\top B\rangle \mathbf{u}_{t+1}\right] + \Upsilon_{t,t+1} \Psi_{t+1}^{-1} \boldsymbol{\eta}_{t+1}$

  end for

The partitioned inverse is given by

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}^{-1} = \begin{pmatrix} F_{11}^{-1} & -A_{11}^{-1}A_{12}F_{22}^{-1} \\ -F_{22}^{-1}A_{21}A_{11}^{-1} & F_{22}^{-1} \end{pmatrix} \tag{93}$$

$$= \begin{pmatrix} A_{11}^{-1} + A_{11}^{-1}A_{12}F_{22}^{-1}A_{21}A_{11}^{-1} & -F_{11}^{-1}A_{12}A_{22}^{-1} \\ -A_{22}^{-1}A_{21}F_{11}^{-1} & A_{22}^{-1} + A_{22}^{-1}A_{21}F_{11}^{-1}A_{12}A_{22}^{-1} \end{pmatrix} \tag{94}$$

and the determinant by

$$\begin{vmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{vmatrix} = |A_{22}| \cdot |F_{11}| = |A_{11}| \cdot |F_{22}|$$

where

$$F_{11} = A_{11} - A_{12}A_{22}^{-1}A_{21}, \qquad F_{22} = A_{22} - A_{21}A_{11}^{-1}A_{12}.$$

Notice that inverses of A_{12} or A_{21} do not appear in these results. There are other Schur complements that are defined in terms of these "off-diagonal" terms, but for our purposes it is inadvisable to use them (Cemgil, private communication).

B Matrix inversion lemma

This proof, or derivation, is included for reference only and is not used in the above work. It plainly shows that the expectations cannot be carried through a matrix inversion in any reasonable way. The lemma is most useful when A is a large diagonal matrix and B has few columns.

$$(A + BCB^\top)^{-1} = A^{-1} - A^{-1}B(C^{-1} + B^\top A^{-1}B)^{-1}B^\top A^{-1} \ .$$

To derive this lemma we use the Taylor series expansion of the matrix inverse

$$(A + M)^{-1} = A^{-1}(I + MA^{-1})^{-1} = A^{-1}\sum_{i=0}^{\infty}(-1)^i(MA^{-1})^i \ ,$$

where the series is only well-defined when the spectral radius of MA^{-1} is less than unity. We can easily check that this series is indeed the inverse by directly multiplying by $(A + M)$, yielding the identity,

$$\begin{aligned} (A+M)A^{-1}\sum_{i=0}^{\infty}(-1)^i(MA^{-1})^i &= AA^{-1}\left[I - MA^{-1} + (MA^{-1})^2 - (MA^{-1})^3 + \dots\right] \\ &\quad + MA^{-1}\left[\ I \ - \ MA^{-1} \ + (MA^{-1})^2 - \dots\right] \\ &= I \ . \end{aligned}$$

In the series expansion we find an embedded expansion, which forms the inverse matrix term on the right hand side, as follows

$$\begin{aligned} (A + BCB^\top)^{-1} &= A^{-1}\sum_{i=0}^{\infty}(-1)^i(BCB^\top A^{-1})^i \\ &= A^{-1}\left(I + \sum_{i=1}^{\infty}(-1)^i(BCB^\top A^{-1})^i\right) \\ &= A^{-1}\left(I - BC\left[\sum_{i=0}^{\infty}(-1)^i(B^\top A^{-1}BC)^i\right]B^\top A^{-1}\right) \\ &= A^{-1}\left(I - BC(I + B^\top A^{-1}BC)^{-1}B^\top A^{-1}\right) \\ &= A^{-1} - A^{-1}B(C^{-1} + B^\top AB)^{-1}B^\top A^{-1} \ . \end{aligned}$$

In this proof we have had to put constraints on A and M for the Taylor expansion to be well-defined. However straight multiplication of the expression by its proposed inverse does in fact yield the identity. This suffices as a proof in itself.