13 BP on Gaussian Hidden Markov Models: Kalman Filtering and Smoothing

As we have seen, when our variables of interest are jointly Gaussian, the sum-product algorithm for inference on trees takes a special form in which the messages themselves are Gaussian. As we also saw earlier, a simple but very important class of trees are the Hidden Markov Models (HMMs). In this section of the notes we specialize our Gaussian BP algorithm to the class of Gaussian HMMs, which are equivalently characterized as linear dynamical systems (LDSs). As we will see, what results is a version of the forward-backward algorithm for such models that is referred to as Kalman filtering and smoothing.

13.1 Gaussian HMM (State Space Model)

Consider the Gaussian HMM depicted in Fig. 1, where to simplify our development we restrict our attention to zero-mean variables and homogeneous models. In this model, the states $\mathbf{x}_t$ and observations $\mathbf{y}_t$ are d- and d'-dimensional vectors, respectively.

The Gaussian HMM can be expressed as an LDS by exploiting the representation of the variables in *innovation form*.[1] In particular, first, the states evolve according to the linear dynamics

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{v}_t, \qquad \mathbf{v}_t \sim \mathtt{N}(\mathbf{0}, \mathbf{Q}), \qquad \mathbf{x}_0 \sim \mathtt{N}(\mathbf{0}, \mathbf{\Lambda}_0), \tag{1a}$$

where $\mathbf{A}$ is a constant matrix, $\mathbf{\Lambda}_0$ and $\mathbf{Q}$ are covariance (i.e., positive semidefinite) matrices, and where $\{\mathbf{v}_t\}$ is a sequence of independent random vectors that are also independent of $\mathbf{x}_0$. Second, the observations depend on the state according to the linear measurements

$$\mathbf{y}_t = \mathbf{C}\mathbf{x}_t + \mathbf{w}_t, \qquad \mathbf{w}_t \sim \mathtt{N}(\mathbf{0}, \mathbf{R}), \tag{1b}$$

where $\mathbf{C}$ is a constant matrix and $\mathbf{R}$ is a covariance matrix, and where $\{\mathbf{w}_t\}$ is a sequence of independent random variables that is also independent of $\mathbf{x}_0$ and $\{\mathbf{v}_t\}$.

Hence, in terms of our notation, we have the conditional distributions

$$\begin{aligned} \mathbf{x}_0 &\sim \mathtt{N}(\mathbf{0}, \mathbf{\Lambda}_0) \\ \mathbf{x}_{t+1}|\mathbf{x}_t &\sim \mathtt{N}(\mathbf{A}\mathbf{x}_t, \mathbf{Q}) \\ \mathbf{y}_t|\mathbf{x}_t &\sim \mathtt{N}(\mathbf{C}\mathbf{x}_t, \mathbf{R}). \end{aligned}$$

[1] Recall that any pair of jointly Gaussian random vectors $\mathbf{u}, \mathbf{v}$ can be expressed in innovation form: there exists a matrix $\mathbf{G}$ and a Gaussian random vector $\mathbf{w}$ (the innovation) that is independent of $\mathbf{v}$ such that

$$\mathbf{u} = \mathbf{G}\mathbf{v} + \mathbf{w}.$$

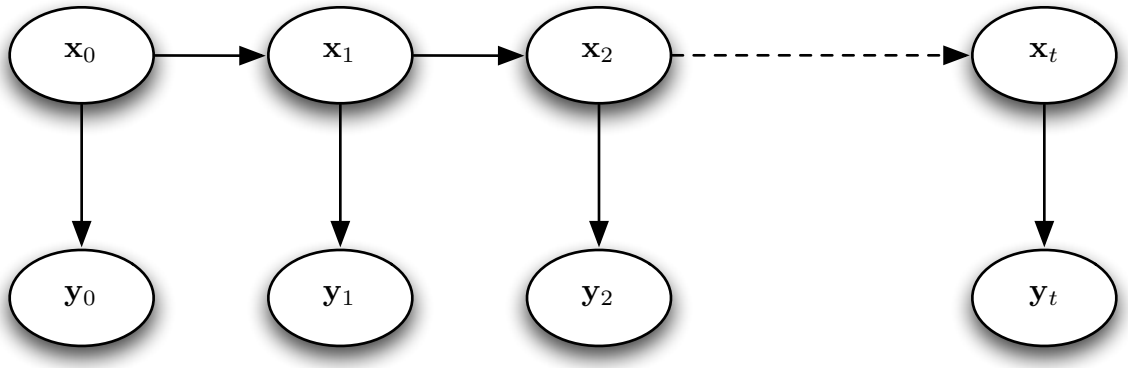

Figure 1: A Gaussian HMM as a directed tree. In this model, the variables are jointly Gaussian.

Such Gaussian HMMs are used to model an extraordinarily broad range of phenomena in practice.

13.2 Inference with Gaussian HMMs

In our development of Gaussian BP for trees, we expressed the algorithm in terms of the information form representation of the Gaussian variables involved. In this section, we further specialize our results to the HMM, and express the quantities involved directly in terms of the parameters of the LDS (1).

To obtain our results, we start by observing that the joint distribution factors as

$$p(\mathbf{x}_0^t, \mathbf{y}_0^t) = p(\mathbf{x}_0)\, p(\mathbf{y}_0|\mathbf{x}_0)\, p(\mathbf{x}_1|\mathbf{x}_0)\, p(\mathbf{y}_1|\mathbf{x}_1) \cdots p(\mathbf{y}_t|\mathbf{x}_t),$$

where we have used $\mathbf{x}_0^t$ as a shorthand to denote the entire sequence $\mathbf{x}_0, \ldots, \mathbf{x}_t$.

Substituting the Gaussian form of the constituent conditional distributions and matching corresponding terms, we then obtain

$$\begin{aligned}
p(\mathbf{x}_0^t, \mathbf{y}_0^t) &\propto \exp\left(-\frac{1}{2}\mathbf{x}_0^{\mathrm{T}}\boldsymbol{\Lambda}_0\mathbf{x}_0\right) \exp\left(-\frac{1}{2}(\mathbf{y}_0 - \mathbf{C}\mathbf{x}_0)^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{y}_0 - \mathbf{C}\mathbf{x}_0)\right) \\
&\qquad \cdot \exp\left(-\frac{1}{2}(\mathbf{x}_1 - \mathbf{A}\mathbf{x}_0)^{\mathrm{T}}\mathbf{Q}^{-1}(\mathbf{x}_1 - \mathbf{A}\mathbf{x}_0)\right) \cdots \\
&\propto \exp\left(-\frac{1}{2}\mathbf{x}_0^{\mathrm{T}}\boldsymbol{\Lambda}_0\mathbf{x}_0 - \frac{1}{2}\mathbf{y}_0^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{y}_0 - \frac{1}{2}\mathbf{x}_0^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{C}\mathbf{x}_0 + \mathbf{x}_0^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{y}_0\right) \\
&\qquad \cdot \exp\left(-\frac{1}{2}\mathbf{x}_1^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{x}_1 - \frac{1}{2}\mathbf{x}_0^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{A}\mathbf{x}_0 + \mathbf{x}_1^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{A}\mathbf{x}_0 + \cdots\right) \\
&\propto \exp\left(-\frac{1}{2}\mathbf{x}_0^{\mathrm{T}}(\boldsymbol{\Lambda}_0 + \mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{C} + \mathbf{A}^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{A})\mathbf{x}_0 - \frac{1}{2}\mathbf{y}_0^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{y}_0 - \frac{1}{2}\mathbf{x}_1^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{x}_1\right)
\end{aligned}$$

$$\cdot \exp\left(\mathbf{x}_0^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{y}_0 + \mathbf{x}_1^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{A}\mathbf{x}_0 + \cdots\right)$$
$$= \prod_{k=0}^{t}\phi_k(\mathbf{x}_k)\prod_{k=1}^{t}\psi_{k-1,k}(\mathbf{x}_{k-1},\mathbf{x}_k)\prod_{k=0}^{t}\varphi_k(\mathbf{y}_k)\prod_{k=t}^{t}\tilde{\varphi}_k(\mathbf{x}_k,\mathbf{y}_k),$$

where

$$\log\phi_k(\mathbf{x}_k) = \begin{cases} -\frac{1}{2}\mathbf{x}_0^{\mathrm{T}}(\underbrace{\mathbf{\Lambda}_0 + \mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{C} + \mathbf{A}^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{A}}_{\triangleq\mathbf{J}_0})\mathbf{x}_0 & k=0 \\ -\frac{1}{2}\mathbf{x}_k^{\mathrm{T}}(\underbrace{\mathbf{Q}^{-1} + \mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{C} + \mathbf{A}^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{A}}_{\triangleq\mathbf{J}_k})\mathbf{x}_k & 1\le k\le t-1 \\ -\frac{1}{2}\mathbf{x}_k^{\mathrm{T}}(\underbrace{\mathbf{Q}^{-1} + \mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{C}}_{\triangleq\mathbf{J}_t})\mathbf{x}_k & k=t \end{cases}$$

$$\log\psi_{k-1,k}(\mathbf{x}_{k-1},\mathbf{x}_k) = \mathbf{x}_k^{\mathrm{T}}\underbrace{\mathbf{Q}^{-1}\mathbf{A}}_{\triangleq\mathbf{L}_k}\mathbf{x}_{k-1},$$
$$\log\varphi_k(\mathbf{y}_k) = -\frac{1}{2}\mathbf{y}_0^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{y}_0,$$
$$\log\tilde{\varphi}_k(\mathbf{x}_k,\mathbf{y}_k) = \mathbf{x}_k^{\mathrm{T}}\underbrace{\mathbf{C}^{\mathrm{T}}\mathbf{R}^{-1}}_{\triangleq\mathbf{M}_k}\mathbf{y}_k.$$

Next note that $\mathbf{y}_0^t$ are observations, so we condition on these variables. After conditioning on $\mathbf{y}_0^t$, the joint distribution continues to be Gaussian, and simplifies to the following form

$$\begin{aligned} p_{\mathbf{x}_0^t|\mathbf{y}_0^t}(\mathbf{x}_0^t|\mathbf{y}_0^t) &\propto p_{\mathbf{x}_0^t,\mathbf{y}_0^t}(\mathbf{x}_0^t,\mathbf{y}_0^t) \\ &\propto \prod_{k=0}^{t}\phi_k(\mathbf{x}_k)\prod_{k=0}^{t}\tilde{\varphi}_k(\mathbf{x}_k,\mathbf{y}_k)\prod_{k=1}^{t}\psi_{k-1,k}(\mathbf{x}_{k-1},\mathbf{x}_k) \\ &\propto \prod_{k=0}^{t}\exp\Big(-\frac{1}{2}\mathbf{x}_k^{\mathrm{T}}\mathbf{J}_k\mathbf{x}_k + \mathbf{x}_k^{\mathrm{T}}\underbrace{\mathbf{M}_k\mathbf{y}_k}_{\triangleq\mathbf{h}_k}\Big)\prod_{k=1}^{t}\exp\left(-\mathbf{x}_k^{\mathrm{T}}(-\mathbf{L}_k)\mathbf{x}_{k-1}\right). \end{aligned}$$

Thus the potential for node k is a Gaussian with information parameters $(\mathbf{h}_k, \mathbf{J}_k)$, and the potential for the edge from node $k-1$ to node k is a Gaussian with information parameters $(0, \mathbf{L}_k)$; see Fig. 2.

Having now expressed the joint distribution as a product of potentials in Gaussian information form, we can easily read off the information matrices and potential vectors required to apply Gaussian BP. In particular, the initial messages are

$$\mathbf{J}_{0\to1} = -\mathbf{L}_1\mathbf{J}_0^{-1}\mathbf{L}_1^{\mathrm{T}}$$
$$\mathbf{h}_{0\to1} = \mathbf{L}_1\mathbf{J}_0^{-1}\mathbf{h}_0.$$

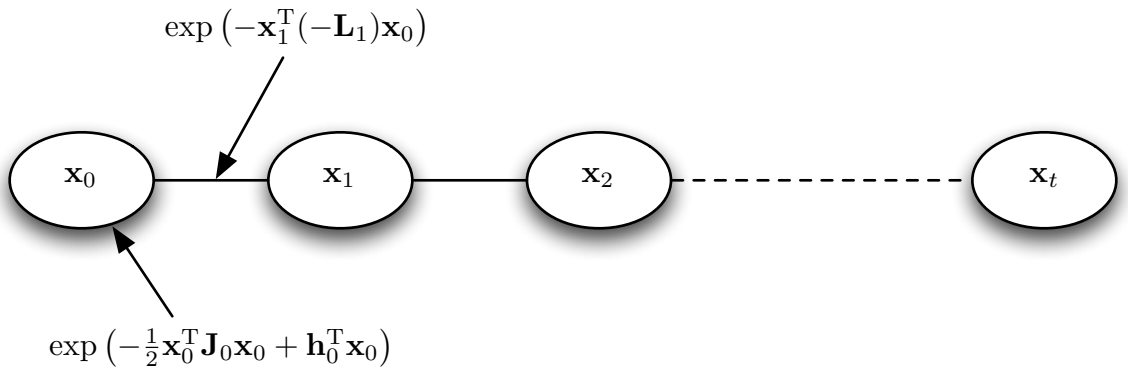

Figure 2: The equivalent undirected tree for Gaussian HMMs, with only the first node and edge potentials shown.

Then, the "forward" pass is given by the recursion

$$\begin{aligned}\mathbf{J}_{i\to i+1} &= -\mathbf{L}_{i+1}\left(\mathbf{J}_i + \mathbf{J}_{i-1\to i}\right)^{-1}\mathbf{L}_{i+1}^{\mathrm{T}} \\ \mathbf{h}_{i\to i+1} &= \mathbf{L}_{i+1}\left(\mathbf{J}_i + \mathbf{J}_{i-1\to i}\right)^{-1}\left(\mathbf{h}_i + \mathbf{h}_{i-1\to i}\right)\end{aligned}, \qquad i = 0, 1, \ldots, t-1.$$

In turn, the "backward" pass is given by the recursion

$$\begin{aligned}\mathbf{J}_{i+1\to i} &= -\mathbf{L}_{i+1}\left(\mathbf{J}_{i+1} + \mathbf{J}_{i+2\to i+1}\right)^{-1}\mathbf{L}_{i+1}^{\mathrm{T}} \\ \mathbf{h}_{i+1\to i} &= \mathbf{L}_{i+1}\left(\mathbf{J}_{i+1} + \mathbf{J}_{i+2\to i+1}\right)^{-1}\left(\mathbf{h}_{i+1} + \mathbf{h}_{i+2\to i+1}\right).\end{aligned}, \qquad i = t-1, t-2, \ldots, 0.$$

Finally, the parameters for the marginals for each i are obtained via

$$\begin{aligned}\hat{\mathbf{J}}_i &= \mathbf{J}_i + \mathbf{J}_{i-1\to i} + \mathbf{J}_{i+1\to i} \\ \hat{\mathbf{h}}_i &= \mathbf{h}_i + \mathbf{h}_{i-1\to i} + \mathbf{h}_{i+1\to i},\end{aligned}$$

from which the mean vectors and covariance matricses associated with the t marginals are given by

$$\boldsymbol{\mu}_i = \hat{\mathbf{J}}_i^{-1}\hat{\mathbf{h}}_i \quad \text{and} \quad \boldsymbol{\Sigma}_i = \hat{\mathbf{J}}_i^{-1}.$$

As a reminder, the complexity of this algorithm scales as $O(td^3)$ compared to $O((td)^3)$ for a naive implementation of the marginalization. The savings is particularly significant because $t \gg d$ in typical applications.

13.3 Kalman Filtering and Smoothing

The implementation of BP for Gaussian HMMs developed in the previous section has a convenient two-pass structure, consisting of a forward and a backward pass. Note that the data $\mathbf{y}_k$ enters into the computation through the potential vector $\mathbf{h}_k$, and thus in the serial version of BP each piece of the data is used three times: once

during the forward pass, once during the backward pass, and once during the marginal computation.

There are a variety of ways to rearrange such forward-backward computation for efficiently producing marginals. One such variant corresponds to what is referred to as the Rauch-Tung-Striebel (RTS) algorithm. In this algorithm, the forward and backward passes take the form of what are referred to as Kalman filtering and smoothing, respectively.

The Kalman filter was introduced by R. Kalman in his extraordinarily influential work in 1959.[2] In contrast to the forward pass of the Gaussian BP procedure developed in the previous section, the Kalman filter directly generates a particular set of marginals. Specifically, it generates the marginals $p(\mathbf{x}_i|\mathbf{y}_0^i)$ for $i = 0, 1, \ldots, t$, i.e., a marginal at node i based on the data only through index i; this is what is meant by the term "filtering." Each step in the forward pass is typically implemented in two substeps: a prediction substep that generates $p(\mathbf{x}_i|\mathbf{y}_0^{i-1})$, followed by an update substep that generates $p(\mathbf{x}_i|\mathbf{y}_0^i)$.

The backward pass of the RTS algorithm, which implements Kalman smoothing, directly generates the desired marginals $p(\mathbf{x}_i|\mathbf{y}_0^t)$, i.e., a marginal at node i based on *all* the data. Moreover, it does so in a manner that requires only access to the marginals obtained in the forward pass in the computation—once the forward pass is complete, the data is no longer needed.

From the above perspectives, the RTS algorithm can be viewed as the Gaussian version of the so-called α, γ variant of the forward-backward algorithm for discrete HMMs. Indeed, suitably normalized, the α messages are the "filtered" marginals, and the γ messages are the "smoothed" marginals. The α, γ forward-backward algorithm and its relation to the α, β forward-backward algorithm we introduced as BP for discrete HMMs are developed in Chapter 12 of Jordan's notes.

There are several versions of the Kalman filter, both in covariance and information forms. As an illustration, we summarize a standard version of the former. As notation, we use $\boldsymbol{\mu}_{i|j}$ and $\boldsymbol{\Sigma}_{i|j}$ to refer to the mean vector and covariance matrix, respectively, of the distribution $p(\mathbf{x}_i|\mathbf{y}_0^j)$. We omit the derivation, which involves substituting the Gaussian forms into the α, γ version of the forward-backward algorithm, with summations replaced by integrals, and evaluating the integrals, the same manner that Gaussian BP was obtained from our original BP equations.

More specifically, the forward (filtering) recursion is

$$\alpha(\mathbf{x}_{i+1}) = \int \alpha(\mathbf{x}_i)\, p(\mathbf{x}_{i+1}|\mathbf{x}_i)\, p(\mathbf{y}_{i+1}|\mathbf{x}_{i+1})\, \mathrm{d}\mathbf{x}_i,$$

[2] You are strongly encouraged to read at least the first two pages to appreciate the broader context as well as significance of this work.

and the backward (smoothing) recursion is

$$\gamma(\mathbf{x}_i) = \int \gamma(\mathbf{x}_{i+1}) \left[\frac{\alpha(\mathbf{x}_i)\, p(\mathbf{x}_{i+1}|\mathbf{x}_i)}{\int \alpha(\mathbf{x}'_i)\, p(\mathbf{x}_{i+1}|\mathbf{x}'_i)\, \mathrm{d}\mathbf{x}'_i} \right] \mathrm{d}\mathbf{x}_{i+1},$$

where, for reference, recall that the γ marginals are expressed in terms of our original β messages via

$$\gamma(\mathbf{x}_i) = \frac{\alpha(\mathbf{x}_i)\, \beta(\mathbf{x}_i)}{p(\mathbf{y}_0^t)}.$$

13.3.1 Filtering

The filtering pass produces the mean and covariance parameters $\boldsymbol{\mu}_{i|i}$ and $\boldsymbol{\Sigma}_{i|i}$, respectively, in sequence for $i = 0, 1, \ldots, t$. Each step consists of the following two substeps:

Prediction Substep: In this substep, we predict the next state $\mathbf{x}_{i+1}$ from the current observations $\mathbf{y}_0^i$, whose impact is summarized in the filtered marginals $p(\mathbf{x}_i|\mathbf{y}_0^i)$ and thus does not require re-accessing the data:

$$p(\mathbf{x}_i|\mathbf{y}_0^i) \to p(\mathbf{x}_{i+1}|\mathbf{y}_0^i), \qquad i = 0, 1, \ldots, t-1.$$

In terms of the model parameters (1), the prediction recursion for the associated marginal parameters can be derived to be

$$\begin{aligned} \boldsymbol{\mu}_{i+1|i} &= \mathbf{A}\boldsymbol{\mu}_{i|i} \\ \boldsymbol{\Sigma}_{i+1|i} &= \mathbf{A}\boldsymbol{\Sigma}_{i|i}\mathbf{A}^{\mathrm{T}} + \mathbf{Q}. \end{aligned} \qquad i = 0, 1, \ldots, t-1,$$

where the initialization of the recursion is

$$\boldsymbol{\mu}_{0|-1} = \mathbf{0} \tag{2}$$

$$\boldsymbol{\Sigma}_{0|-1} = \boldsymbol{\Lambda}_0. \tag{3}$$

Update Substep: In this substep, we update the prediction at step i by incorporating the new data $\mathbf{y}_{i+1}$, i.e.,

$$p(\mathbf{x}_{i+1}|\mathbf{y}_0^i) \to p(\mathbf{x}_{i+1}|\mathbf{y}_0^{i+1}), \qquad i = 0, 1, \ldots, t-1.$$

In terms of the model parameters (1), the update recursion for the associated marginal parameters can be derived to be

$$\begin{aligned} \boldsymbol{\mu}_{i+1|i+1} &= \boldsymbol{\mu}_{i+1|i} + \mathbf{G}_{i+1}(\mathbf{y}_{i+1} - \mathbf{C}\boldsymbol{\mu}_{i+1|i}) \\ \boldsymbol{\Sigma}_{i+1|i+1} &= \boldsymbol{\Sigma}_{i+1|i} - \mathbf{G}_{i+1}\mathbf{C}\boldsymbol{\Sigma}_{i+1|i} \end{aligned} \qquad i = 0, 1, \ldots, t-1,$$

where

$$\mathbf{G}_{i+1} = \boldsymbol{\Sigma}_{i+1|i}\mathbf{C}^{\mathrm{T}}(\mathbf{C}\boldsymbol{\Sigma}_{i+1|i}\mathbf{C}^{\mathrm{T}} + \mathbf{R})^{-1}$$

is a precomputable quantity referred to as the *Kalman gain*.

13.3.2 Smoothing

The filtering pass produces the mean and covariance parameters $\boldsymbol{\mu}_{i|t}$ and $\boldsymbol{\Sigma}_{i|t}$, respectively, in sequence for $i = t, t-1, \ldots, 0$. Each step implements

$$p(\mathbf{x}_{i+1}|\mathbf{y}_0^t) \to p(\mathbf{x}_i|\mathbf{y}_0^t), \qquad i = t, t-1, \ldots, 0.$$

In terms of the model parameters (1), the smoothing recursion for the associated marginal parameters can be derived to be

$$\begin{aligned} \boldsymbol{\mu}_{i|t} &= \boldsymbol{\mu}_{i|i} + \mathbf{F}_i(\boldsymbol{\mu}_{i+1|t} - \boldsymbol{\mu}_{i+1|i}) \\ \boldsymbol{\Sigma}_{i|t} &= \mathbf{F}_i(\boldsymbol{\Sigma}_{i+1|t} - \boldsymbol{\Sigma}_{i+1|i})\mathbf{F}_i^{\mathrm{T}} + \boldsymbol{\Sigma}_{i|i} \end{aligned} \qquad i = t, t-1, \ldots, 0.$$

where

$$\mathbf{F}_i = \boldsymbol{\Sigma}_{i|i}\mathbf{A}^{\mathrm{T}}\boldsymbol{\Sigma}_{i+1|i}^{-1}.$$

6.438 Algorithms for Inference

Fall 2014