EM for a mixture of drifting t-distributions

Kevin Shan

2014-03-12

1 EM for a mixture of stationary t-distributions

For this I will follow the notation in *Finite Mixture Models* by McLachlan and Peel (2000). This section is a review of chapter 2 ("ML fitting of mixture models") and chapter 7 ("Multivariate t mixtures").

We have a mixture-of-(stationary)-t-distributions where the PDF f_{mot} of a single observation y_j is:

$$f_{\text{mot}}(y_j; \Psi) = \sum_{k=1}^{K} \alpha_k f_{\text{mvt}}(y_j; \mu_k, \Sigma_k, \nu)$$

where α_k is the relative contribution of component k and f_{mvt} is the PDF of the multivariate t-distribution:

$$f_{\text{mvt}}(y_j; \mu_k, \Sigma_k, \nu) = \frac{\Gamma\left(\frac{\nu+p}{2}\right) |\Sigma_k|^{-1/2}}{(\pi\nu)^{\frac{1}{2}p}\Gamma(\frac{\nu}{2})\big(1 + (y_j - \mu_k)' \, \Sigma_k^{-\frac{1}{2}}(y_j - \mu_k)\big)^{\frac{1}{2}(\nu+p)}}$$

where p is the number of dimensions, and the parameters μ, Σ, ν are called the location, scale, and degrees-of-freedom, respectively.

Our goal is to fit parameters α, μ, Σ given the set of observations y and assuming ν is known.

1.1 Formulation as an incomplete-data problem

Given the set of observations y, the overall log likelihood for a parameter set $\Psi = \{\alpha, \mu, \Sigma\}$ is:

$$\log L(\Psi) = \sum_{j=1}^{N} \log \left(\sum_{k=1}^{K} \alpha_k f_{\text{mvt}}(y_j; \mu_k, \Sigma_k, \nu) \right)$$

which is difficult to optimize directly. So instead we introduce indicator variables $Z \in \{0, 1\}$ such that $Z_{kj} = 1$ if spike j came from component k and 0 otherwise. Now if we treat both y_j and $Z_{kj} = z_{kj}$ as known, we get the "complete-data" log likelihood $\log L_c(\Psi)$

$$\log L_c(\Psi) = \sum_{j=1}^{N} \sum_{k=1}^{K} z_{kj} \left(\log \alpha_k + \log f_{\text{mvt}}(y_j; \mu_k, \Sigma_k, \nu)\right) \tag{1}$$

1.2 Reformulation of the t-distribution

Unfortunately, the expression for f_{mvt} is a mess to deal with. However, it does have a convenient factorization, i.e. given a gamma-distributed random variable U (shape-rate parametrization):

$$U \sim \text{gamma}(\frac{1}{2}\nu, \frac{1}{2}\nu)$$

and a random variable Y whose distribution conditional on $U = u$ is Gaussian:

$$Y \mid U \sim \mathcal{N}(\mu, \Sigma/u)$$

the marginal distribution of Y will be t-distributed with location μ, scale Σ, and degrees-of-freedom ν. This is a common method of generating samples from a multivariate t-distribution, see e.g. MATLAB's `mvtrnd` function.

This gives us a joint distribution of Y_j and U_j (assuming that they come from component k):

$$\begin{aligned}
f_{\mathrm{mvt}}(y_j, u_j; \mu_k, \Sigma_k, \nu) &= f_{\mathrm{mvn}}(y_j; \mu_k, \Sigma_k/u_j) f_{\mathrm{gamma}}(u_j; \frac{1}{2}\nu, \frac{1}{2}\nu) \\
\log f_{\mathrm{mvt}}(y_j, u_j; \mu_k, \Sigma_k, \nu) &= -\frac{1}{2}p\log(2\pi) - \frac{1}{2}\log|\Sigma_k/u_j| - \frac{1}{2}(y_j - \mu_k)'(\Sigma_k/u_j)^{-1}(y_j - \mu_k) \\
&\quad - \log\Gamma(\frac{1}{2}\nu) + \frac{1}{2}\nu\log(\frac{1}{2}\nu) + \frac{1}{2}\nu(\log u_j - u_j) - \log u_j
\end{aligned}$$

So we add this U as an additional latent variable, then substitute into (1) to get:

$$\begin{aligned}
\log L_c(\Psi) = \sum_{j=1}^{N}\sum_{k=1}^{K} z_{kj}\Big[&\log\alpha_k - \frac{1}{2}p\log(2\pi) - \frac{1}{2}\log|\Sigma_k| + \frac{1}{2}p\log u_j - \frac{1}{2}u_j(y_j - \mu_k)'\Sigma_k^{-1}(y_j - \mu_k) \\
&- \log\Gamma(\frac{1}{2}\nu) + \frac{1}{2}\nu\log(\frac{1}{2}\nu) + \frac{1}{2}\nu(\log u_j - u_j) - \log u_j\Big] \qquad (2)
\end{aligned}$$

I will note that this differs from the expression in the book (eq 7.11–7.14) by the inclusion of this $\frac{1}{2}p\log u_j$ term. I'm pretty sure it belongs there but ultimately it doesn't matter because it doesn't interact with the parameters we're optimizing over.

1.3 E-step

Now we take the expectation of (2) over the latent variables Z and U, conditional on the observations y, and treating our current parameter estimates $\hat{\Psi} = \{\hat{\alpha}, \hat{\mu}, \hat{\Sigma}\}$ as fixed:

$$\begin{aligned}
Q(\Psi|\hat{\Psi}) &= \mathrm{E}_{Z,U}\big(\log L_c(\Psi) \mid y, \hat{\Psi}\big) \\
&= \sum_{j=1}^{N}\sum_{k=1}^{K} \mathrm{E}_{Z_{kj},U_j}\big(z_{kj}[\cdots] \mid y_j, \hat{\Psi}\big) \\
&= \sum_{j=1}^{N}\sum_{k=1}^{K} \mathrm{P}(Z_{kj} = 1 \mid y_j, \hat{\Psi})\mathrm{E}_{U_j}\big([\cdots] \mid Z_{kj} = 1, y_j, \hat{\Psi}\big) \qquad (3)
\end{aligned}$$

where $[\cdots]$ represents the long bracketed expression in (2).

Relying on MacLachlan and Peel for the math here, we'll introduce the following variables:

$$\begin{aligned}
\tau_{kj} = P(Z_{kj} = 1 \mid y_j, \hat{\Psi}) \qquad &= \frac{\hat{\alpha}_k f_{\mathrm{mvt}}(y_j; \hat{\mu}_k, \hat{\Sigma}_k, \nu)}{f_{\mathrm{mot}}(y_j; \hat{\Psi})} \qquad (4) \\
u_{kj} = \mathrm{E}_{U_j}(U_j \mid Z_{kj} = 1, y_j, \hat{\Psi}) &= \frac{\nu + p}{\nu + (y_j - \hat{\mu}_k)'\hat{\Sigma}_k^{-1}(y_j - \hat{\mu}_k)} \qquad (5)
\end{aligned}$$

τ_{kj} is the familiar expression for the posterior membership, and u_{kj} will turn out to be a sort of correction term for the non-Gaussian-ness of the observations. It accounts for the longer tails by weighting the faraway points less. In the Gaussian limit ($\nu \to \infty$), we get $u_{kj} \to 1$.

I'll also note that there exists an expression for $\mathrm{E}_{U_j}(\log U_j \mid Z_{kj} = 1, y_j, \hat{\Psi})$, but it's not important to us because we're not optimizing over ν in the M-step.

Substituting τ_{kj} and u_{kj} into (3), we get the objective function for the M-step optimization:

$$Q(\Psi|\hat{\Psi}) = \sum_{j=1}^{N} \sum_{k=1}^{K} \tau_{kj} \left[\log \alpha_k - \frac{1}{2} \log |\Sigma_k| - \frac{1}{2} u_{kj} (y_j - \mu_k)' \Sigma_k^{-1} (y_j - \mu_k) + \cdots \right] \tag{6}$$

where we have eliminated terms not involving α, μ, or Σ.

1.4 M-step

The objective function in (6) is fairly straightforward. Our update is simply:

$$\hat{\alpha}_k = \arg\max_{\alpha_k} Q(\Psi|\hat{\Psi}) = \frac{1}{N} \sum_{j=1}^{N} \tau_{kj} \tag{7}$$

$$\hat{\mu}_k = \arg\max_{\mu_k} Q(\Psi|\hat{\Psi}) = \frac{\sum_{j=1}^{N} \tau_{kj} u_{kj}\, y_j}{\sum_{j=1}^{N} \tau_{jk} u_{kj}} \tag{8}$$

$$\hat{\Sigma}_k = \arg\max_{\Sigma_k} Q(\Psi|\hat{\Psi}) = \frac{\sum_{j=1}^{N} \tau_{kj} u_{kj} (y_j - \hat{\mu}_k)(y_j - \hat{\mu}_k)'}{\sum_{j=1}^{N} \tau_{kj}} \tag{9}$$

which is simply a weighted version of the Mixture-of-Gaussians M-step.

2 Adaptation to a mixture of drifting t-distributions

For convenience of notation let us assume there is exactly one observation per time step. We will relax this later. Our underlying model for this mixture-of-drifting-t-distributions is:

$$f_{\mathrm{modt}}(y_t; \Psi) = \sum_{k=1}^{K} \alpha_k f_{\mathrm{mvt}}(y_t; \mu_{kt}, \Sigma_k, \nu)$$
$$\mu_{kt} \sim \mu_{k(t-1)} + \mathcal{N}(0, Q)$$

where we assume the drift covariance Q is known. Adding the drift turns our complete-data log-likelihood (c.f. eq 1) into

$$\log L_c(\Psi) = \sum_{t=1}^{T} \sum_{k=1}^{K} z_{kt} (\log \alpha_k + \log f_{\mathrm{mvt}}(y_t; \mu_k, \Sigma_k, \nu)) + \sum_{t=2}^{T} \sum_{k=1}^{K} \log f_{\mathrm{mvn}}(\mu_{kt}; \mu_{k(t-1)}, Q)$$

where f_{mvn} is the multivariate normal PDF.

Adding the additional latent variable U, we get (c.f. eq 2):

$$\begin{aligned}
\log L_c(\Psi) \;=& \sum_{t=1}^{T} \sum_{k=1}^{K} z_{kj} \Big[\quad \log \alpha_k - \frac{1}{2} p \log(2\pi) - \frac{1}{2} \log |\Sigma_k| - \frac{1}{2} u_t (y_t - \mu_{kt})' \Sigma_k^{-1} (y_t - \mu_{kt}) \\
&\qquad - \log \Gamma(\frac{1}{2}\nu) + \frac{1}{2}\nu \log(\frac{1}{2}\nu) + \frac{1}{2}\nu (\log u_t - u_t) - \log u_t \Big] \\
&+ \sum_{t=2}^{T} \sum_{k=1}^{K} \Big[-\frac{1}{2} p \log(2\pi) - \frac{1}{2} \log |Q| - \frac{1}{2} (\mu_{kt} - \mu_{k(t-1)})' Q^{-1} (\mu_{kt} - \mu_{k(t-1)}) \Big]
\end{aligned} \tag{10}$$

The additional term has no Z or U in it, so it doesn't affect the E-step. Our objective function is then:

$$\begin{aligned} Q(\Psi|\hat{\Psi}) = &\sum_{t=1}^{T}\sum_{k=1}^{K} \tau_{kt}\left[\log \alpha_k - \frac{1}{2}\log|\Sigma_k| - \frac{1}{2}u_{kt}(y_t - \mu_{kt})'\Sigma_k^{-1}(y_t - \mu_{kt}) + \cdots\right] \\ &+ \sum_{t=2}^{T}\sum_{k=1}^{K}\left[-\frac{1}{2}(\mu_{kt} - \mu_{k(t-1)})'Q^{-1}(\mu_{kt} - \mu_{k(t-1)}) + \cdots\right] \end{aligned}$$

We can make this a little more compact by defining μ_{k0} and letting the second sum start at $t = 1$. We will discuss how to define μ_{k0} later.

$$\begin{aligned} Q(\Psi|\hat{\Psi}) = \sum_{t=1}^{T}\sum_{k=1}^{K} \Bigg(&\tau_{kt}\Big[\log \alpha_k - \frac{1}{2}\log|\Sigma_k| - \frac{1}{2}u_{kt}(y_t - \mu_{kt})'\Sigma_k^{-1}(y_t - \mu_{kt}) + \cdots\Big] \\ &- \frac{1}{2}(\mu_{kt} - \mu_{k(t-1)})'Q^{-1}(\mu_{kt} - \mu_{k(t-1)}) + \cdots\Bigg) \end{aligned} \tag{11}$$

2.1 M-step

Unlike the stationary case, our M-step update for μ will depend on Σ. So we will first estimate μ holding Σ constant, and then (as in the stationary case) use the updated μ to estimate Σ.

The additional drift term in (11) only affects μ, so our update equations for α and Σ remain unchanged:

$$\hat{\alpha}_k = \arg\max_{\alpha_k} Q(\Psi|\hat{\Psi}) = \frac{1}{T}\sum_{t=1}^{T}\tau_{kt} \tag{12}$$

$$\hat{\Sigma}_k = \arg\max_{\Sigma_k} Q(\Psi|\hat{\Psi}) = \frac{\sum_{t=1}^{T}\tau_{kt}u_{kt}(y_t - \hat{\mu}_{kt})(y_t - \hat{\mu}_{kt})'}{\sum_{t=1}^{T}\tau_{kt}} \tag{13}$$

Each component k is independent of the rest, so our $\hat{\mu}_k$ update will solve the optimization problem:

$$\underset{\{\mu_1,\ldots,\mu_T\}}{\text{minimize}} \quad \sum_{t=1}^{T}\left[\tau_t u_t(y_t - \mu_t)'\Sigma^{-1}(y_t - \mu_t) + (\mu_t - \mu_{t-1})'Q^{-1}(\mu_t - \mu_{t-1})\right] \tag{14}$$

This is an unconstrained quadratic optimization problem, so we could solve this by inverting a single $(Tp) \times (Tp)$ matrix. Instead we will exploit the block-tridiagonal structure to give us a recursive algorithm that solves the problem with T inversions of $p \times p$ matrices.

2.2 Kalman filter (forward pass) – derivation

The Kalman filter is often derived as the minimum mean-squared-error estimator for a linear system (as Kálmán did in 1960), or as a Bayesian update on a linear system with Gaussian noise (which is a 2-line proof, plus a lemma about conditional distributions). In this section we will derive it as a dynamic program for solving an optimization problem related to (14).

Let us introduce "cost-to-go" functions $J_{1|1}, \ldots, J_{T|T}$:

$$J_{t|t}(\mu_t) = \min_{\{\mu_1,\ldots,\mu_{t-1}\}} \sum_{s=1}^{t}\left[\tau_s u_s(y_s - \mu_s)'\Sigma^{-1}(y_s - \mu_s) + (\mu_s - \mu_{s-1})'Q^{-1}(\mu_s - \mu_{s-1})\right] \tag{15}$$

So $J_{t|t}$ tells us how our cumulative cost (from the start of time to time t) is affected by our choice of μ_t, assuming that we make the optimal choice for the other $\mu_1, \ldots, \mu_{t-1}$.

We will assume that these cost functions are quadratic (we will justify this assumption later), and so can be defined in terms of a $\mu_{t|t}$ and a positive definite $P_{t|t}$:

$$J_{t|t}(\mu_t) = (\mu_t - \mu_{t|t})' P_{t|t}^{-1} (\mu_t - \mu_{t|t}) + \text{constants} \tag{16}$$

For the sake of brevity, we will ignore the additive constants in $J_{t|t}$ from here on out, and just say that $J_{t|t}(\mu_t) = (\mu_t - \mu_{t|t})' P_{t|t}^{-1} (\mu_t - \mu_{t|t})$. These constants are irrelevant because we don't care about the actual value of the objective function, just that we are minimizing it.

Now we will derive the recursion equations. If $J_{t|t}$ is known (i.e. $\mu_{t|t}$ and $P_{t|t}$ are known), we can define a new cost function $J_{t+1|t}$ that includes the drift penalty for the next time step:

$$\begin{aligned}
J_{t+1|t}(\mu_{t+1}) &= \min_{\{\mu_1,\ldots,\mu_t\}} \left[\sum_{s=1}^{t} [\cdots] + (\mu_{t+1} - \mu_t)' Q^{-1} (\mu_{t+1} - \mu_t) \right] \\
&= \min_{\mu_t} \left[J_{t|t}(\mu_t) + (\mu_{t+1} - \mu_t)' Q^{-1} (\mu_{t+1} - \mu_t) \right] \\
&= \min_{\mu_t} \left[(\mu_t - \mu_{t|t})' P_{t|t}^{-1} (\mu_t - \mu_{t|t}) + (\mu_{t+1} - \mu_t)' Q^{-1} (\mu_{t+1} - \mu_t) \right]
\end{aligned} \tag{17}$$

Taking the derivative with respect to μ_t and setting it to zero, we get the optimal choice of μ_t, which we will denote $\mu_t^\star$:

$$\begin{aligned}
0 &= 2(P_{t|t}^{-1} + Q^{-1})\mu_t^\star - 2P_{t|t}^{-1}\mu_{t|t} - 2Q^{-1}\mu_{t+1} \\
\mu_t^\star &= \arg\min_{\mu_t} [\cdots] = (P_{t|t}^{-1} + Q^{-1})^{-1} (P_{t|t}^{-1}\mu_{t|t} + Q^{-1}\mu_{t+1})
\end{aligned} \tag{18}$$

Substituting $\mu_t^\star$ into (17) and applying some matrix inversion identities, we get:

$$J_{t+1|t}(\mu_{t+1}) = (\mu_{t+1} - \mu_{t|t})' (P_{t|t} + Q)^{-1} (\mu_{t+1} - \mu_{t|t})$$

To harmonize with the standard Kalman notation, let us introduce $\mu_{t+1|t}$ and $P_{t+1|t}$ such that:

$$J_{t+1|t}(\mu_{t+1}) = (\mu_{t+1} - \mu_{t+1|t})' P_{t+1|t}^{-1} (\mu_{t+1} - \mu_{t+1|t})$$

$$\mu_{t+1|t} = \mu_{t|t} \tag{19}$$

$$P_{t+1|t} = P_{t|t} + Q \tag{20}$$

This corresponds to the Kalman filter "prediction" step.

Now let us shift our indexing up by one (so that $t+1$ becomes t and t becomes $t-1$) and then redefine (15) in terms of $J_{t|t-1}$:

$$\begin{aligned}
J_{t|t}(\mu_t) &= \min_{\{\mu_1,\ldots,\mu_{t-1}\}} \left[\sum_{s=1}^{t} [\cdots] \right] \\
&= \tau_t u_t (y_t - \mu_t)' \Sigma^{-1} (y_t - \mu_t) + J_{t|t-1}(\mu_t) \\
&= \tau_t u_t (y_t - \mu_t)' \Sigma^{-1} (y_t - \mu_t) + (\mu_t - \mu_{t|t-1})' P_{t|t-1}^{-1} (\mu_t - \mu_{t|t-1})
\end{aligned}$$

We can collect terms and ignore terms not involving μ_t:

$$J_{t|t}(\mu_t) = \mu_t' (\tau_t u_t \Sigma^{-1} + P_{t|t-1}^{-1}) \mu_t - 2\mu_t' (\tau_t u_t \Sigma^{-1} y_t + P_{t|t-1}^{-1} \mu_{t|t-1}) + \text{constants}$$

Completing the square and ignoring the additive constants, we get:

$$J_{t|t}(\mu_t) = (\mu_t - \mu_{t|t})' P_{t|t}^{-1} (\mu_t - \mu_{t|t})$$

$$P_{t|t} = \left(\tau_t u_t \Sigma^{-1} + P_{t|t-1}^{-1} \right)^{-1} \tag{21}$$

$$\mu_{t|t} = \left(\tau_t u_t \Sigma^{-1} + P_{t|t-1}^{-1} \right)^{-1} (\tau_t u_t \Sigma^{-1} y_t + P_{t|t-1}^{-1} \mu_{t|t-1}) \tag{22}$$

This corresponds to the Kalman filter "update" step. We can see that this update for $J_{t|t}$ maintains the quadratic form we assumed in (16).

If a single time step has multiple observations $y_t^{(i)}$ with corresponding $\tau_t^{(i)}, u_t^{(i)}$, then equations (21) and (22) generalize to:

$$
\begin{aligned}
P_{t|t} &= \left(\sum_i \left[\tau_t^{(i)} u_t^{(i)} \Sigma^{-1} \right] + P_{t|t-1}^{-1} \right)^{-1} \\
\mu_{t|t} &= P_{t|t} \left(\sum_i \left[\tau_t^{(i)} u_t^{(i)} \Sigma^{-1} y_t^{(i)} \right] + P_{t|t-1}^{-1} \mu_{t|t-1} \right)
\end{aligned}
$$

We could also rewrite (21) and (22) in terms of the Kalman gain K_t:

$$
\begin{aligned}
K_t &= P_{t|t-1} \left(\frac{1}{\tau_t u_t} \Sigma + P_{t|t-1} \right)^{-1} \\
P_{t|t} &= (I - K_t) P_{t|t-1} \\
\mu_{t|t} &= \mu_{t|t-1} + K_t (y_t - \mu_{t|t-1})
\end{aligned}
$$

This is the more commonly-encountered form of the Kalman filter, but it is not as efficient in handling multiple observations per time step.

2.3 Kalman filter (forward pass) – summary

Our original problem was to find $\{\hat{\mu}_1, \ldots, \hat{\mu}_T\}$ that minimizes the objective function (14):

$$
\underset{\{\mu_1,\ldots,\mu_T\}}{\text{minimize}} \quad \sum_{t=1}^{T} \left[\tau_t u_t (y_t - \mu_t)' \Sigma^{-1} (y_t - \mu_t) + (\mu_t - \mu_{t-1})' Q^{-1} (\mu_t - \mu_{t-1}) \right]
$$

As we showed in the previous section, the Kalman filter computes a related function:

$$
J_{t|t}(\mu_t) = \min_{\{\mu_1,\ldots,\mu_{t-1}\}} \sum_{s=1}^{t} \left[\tau_s u_s (y_s - \mu_s)' \Sigma^{-1} (y_s - \mu_s) + (\mu_s - \mu_{s-1})' Q^{-1} (\mu_s - \mu_{s-1}) \right]
$$

where $J_{t|t}$ has a quadratic form:

$$
J_{t|t}(\mu_t) = (\mu_t - \mu_{t|t})' P_{t|t}^{-1} (\mu_t - \mu_{t|t})
$$

and the values for $\mu_{t|t}$ and $P_{t|t}$ are given by the recursive update:

$$
P_{t|t} = \left(\tau_t u_t \Sigma^{-1} + (P_{t-1|t-1} + Q)^{-1} \right)^{-1} \tag{23}
$$

$$
\mu_{t|t} = P_{t|t} \left(\tau_t u_t \Sigma^{-1} y_t + (P_{t-1|t-1} + Q)^{-1} \mu_{t-1|t-1} \right) \tag{24}
$$

So far we have not yet talked about the initialization of this forward pass, i.e. defining $P_{0|0}$ and $\mu_{0|0}$. We don't want $\mu_{0|0}$ to affect our estimate because it's not part of our original log-likelihood function (10). So we can either set $P_{0|0}$ to be very large (and then the value of $\mu_{0|0}$ doesn't matter), or set $\mu_{0|0}$ to be equal to $\mu_{1|1}$. The `MoKsm` code strives for the latter by setting $\mu_{0|0}$ equal to $\hat{\mu}_1$ from the previous EM iteration.

In the next section, we will show how the backwards pass uses $\mu_{t|t}$ and $P_{t|t}$ to determine $\{\hat{\mu}_1, \ldots, \hat{\mu}_T\}$ that solve our original optimization problem.

2.4 Rauch–Tung–Striebel smoother (backwards pass)

The algorithm for the backwards pass is originally due to Rauch, Tung, and Striebel (1965).

Suppose we knew the values for $\mu_{t+1}, \ldots, \mu_T$ that minimize our objective function, i.e.

$$\{\hat{\mu}_{t+1}, \ldots, \hat{\mu}_T\} = \underset{\{\mu_{t+1}, \ldots, \hat{\mu}_T\}}{\arg\min} \sum_{s=1}^{T} \left[\tau_s u_s (y_s - \mu_s)' \Sigma^{-1} (y_s - \mu_s) + (\mu_s - \mu_{s-1})' Q^{-1} (\mu_s - \mu_{s-1})\right]$$

Our task now is to choose the optimal value for μ_t. Since the optimal values for $\mu_{t+1}, \ldots, \mu_T$ are known, we can treat them as constants:

$$\begin{aligned}
\hat{\mu}_t &= \arg\min_{\mu_t} \sum_{s=1}^{T} \left[\tau_s u_s (y_s - \mu_s)' \Sigma^{-1} (y_s - \mu_s) + (\mu_s - \mu_{s-1})' Q^{-1} (\mu_s - \mu_{s-1})\right] \\
&= \arg\min_{\mu_t} \sum_{s=1}^{t} \left[\tau_s u_s (y_s - \mu_s)' \Sigma^{-1} (y_s - \mu_s) + (\mu_s - \mu_{s-1})' Q^{-1} (\mu_s - \mu_{s-1})\right] \\
&\qquad + (\hat{\mu}_{t+1} - \mu_t)' Q^{-1} (\hat{\mu}_{t+1} - \mu_t) + \text{constants}
\end{aligned}$$

This is an expression we've already encountered in equation (17). We've even determined the optimal value for μ_t in (18):

$$\begin{aligned}
\hat{\mu}_t &= \arg\min_{\mu_t} \left[J_{t|t}(\mu_t) + (\hat{\mu}_{t+1} - \mu_t)' Q^{-1} (\hat{\mu}_{t+1} - \mu_t)\right] \\
&= (P_{t|t}^{-1} + Q^{-1})^{-1} (P_{t|t}^{-1} \mu_{t|t} + Q^{-1} \hat{\mu}_{t+1}) \\
&= \left(I - P_{t|t}(P_{t|t} + Q)^{-1}\right) \mu_{t|t} + \left(I - Q(P_{t|t} + Q)^{-1}\right) \hat{\mu}_{t+1} \\
&= \mu_{t|t} + P_{t|t}(P_{t|t} + Q)^{-1} (\hat{\mu}_{t+1} - \mu_{t|t})
\end{aligned} \tag{25}$$

To initialize the recursion, we note that $J_{T|T}$ is in fact the overall objective function we are trying to minimize. So we start by setting:

$$\hat{\mu}_T = \arg\min_{\mu_T} J_{T|T}(\mu_T) = \mu_{T|T} \tag{26}$$

And then use (25) to get the rest of $\{\hat{\mu}_1, \ldots, \hat{\mu}_T\}$. Repeating for each component k gives us our M-step update for $\hat{\mu}_{kt}$.