Adaptive Kalman Filtering for Syntactic Learning Processes

This document seeks to develop an adaptive Kalman filtering method by integrating the Bayes' rule and the Chapman-Kolmogorov framework, focusing on a linear state equation and syntactic observation models. The objective is to design adaptive filters suitable for estimating sequences from songs made up of specific phrases.

For a given alphabet $A = \{'a',' b',' c'\}$, we establish a set of parameters to determine the transition probability between letters using a parameter vector, under the assumption of a first-order Markov process:

$$P\left(y_m^k \middle| y_{m-1}^k; \{\theta\}^k\right) = \begin{pmatrix} \theta_{\{y_m=a, y_{m-1}=1\}} \\ \cdots \\ \theta_{\{y_m=c, y_{m-1}=c\}} \end{pmatrix}$$

We further represent this vector using a function $f: \mathbb{R} \to [0:1]$ through the softmax function (the significance of this function will be elaborated later). Hence, for every sequence 'k' (represented with a superscript), consisting of characters $\{Y_n^k\} = \{y_1^k, y_2^k, \dots, y_n^k\}$:

$$P\left(y_m^k \middle| y_{m-1}^k; \{x\}^k\right) = f\left(x_{\{y_m, y_{m-1}\}}^k\right) \equiv \frac{e^{x_{\{y_m, y_{m-1}\}}^k}}{Z_{y_{m-1}}}; Z_{y_{m-1}} = \sum_{i=1}^{|A|} e^{x_{\{y_m=A_i, y_{m-1}\}}^k}$$

This ensures the generation of transition probabilities as the denominator aggregates over the index i in alphabet A; For instance:

$$P(a|a; \{x\}^k) = f\left(x_{\{y_m='a', y_{m-1}='a'\}}^k\right); then\ Z_{y_{m-1}='a'} = \sum\nolimits_{y_m='a','b','c'} \left[e^{x_{\{y_m,a\}}^k}\right]$$

We hypothesize that the vector of logits $\vec{x}$ experiences a "random walk" after each sequence 'k', such that:

$$P(x^{k+1}|x^k) \sim \mathcal{N}(x^k, \Sigma)$$

Here Σ denotes the noise parameter, which aligns with the number of parameters:

$$\Sigma = \begin{pmatrix} \sigma_{a1}^2 & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & \sigma_{cc}^2 \end{pmatrix}$$

The likelihood of K sequences, each with $n(k \in K)$ letters, given the parameters, can be expressed using the Markov property as:

$$L(\{y\}^K|\{x\}^K, \Sigma) = \prod_{k=1}^{K}\prod_{m=1}^{n(k)}[P(\{y\}_m^k; x^k)] \cdot P(x^k|x^{k-1})$$

$$= \prod_{k=1}^{K}\prod_{m=1}^{n(k)}\left[P\left(y_m^k|y_{m-1}^k; x^k\right)\right] \cdot P(x^k|x^{k-1})$$

By taking the log of each side, we obtain the log-likelihood, which we'll denote as $\mathcal{L}$:

$$(1)\ \mathcal{L}(\{y\}^K|\{x\}^K, \Sigma) = \sum_{k=1}^{K}\left[\sum_{m=1}^{n(k)}\log\left(P\left(y_m^k|y_{m-1}^k; x^k\right)\right)\right]\log\left(P(x^k|x^{k-1})\right)$$

$$= \sum_{k=1}^{K}\sum_{m=1}^{n(k)}\left[\log\left(f\left(x^k_{\{y_m,y_{m-1}\}}\right)\right)\right] - \frac{1}{2}\left(x^k - x^{k-1}\right)^T\Sigma^{-1}\left(x^k - x^{k-1}\right) + K \cdot \log\left((2\pi)^{-\frac{||\vec{x}||}{2}} \cdot |\Sigma|^{-\frac{1}{2}}\right)$$

The subscript m denotes the m^{th} letter in the k^{th} sequence.

Our primary objective is to optimize this likelihood function to determine the most accurate estimation for the parameters. To achieve this, we will execute N simulations. During each iteration i (where $i < N$) of the simulation, we utilize the parameter estimates denoted by $\psi^i = \left[x_0^i, \Sigma^i\right]$. From this, we derive an estimation for ψ^{i+1} for the subsequent simulation using the Expectation-Maximization (EM) algorithm.

To accomplish this optimization, we need to compute:

$$Q\left(\psi, \psi'\right) = E_{p(\{x\}|\{y\},\Sigma)}\mathcal{L}$$

$$= E\left[\sum_{k=1}^{K}\left[\sum_{m=1}^{n(k)}\left(\log\left(f\left(x^k_{\{y_m,y_{m-1}\}}\right)\right)\right)\right]||\{y\}, x_0^i, \Sigma^i\right]$$

$$- E\left[\sum_{k=1}^{K}\frac{1}{2}(x^k - x^{k-1})^T\Sigma^{-1}(x^k - x^{k-1})\ ||\{y\}, x_0^i, \Sigma^i\right] + \frac{K\left||\vec{x}|\right|}{2} \cdot \log(2\pi) - \frac{K}{2}\log|\Sigma|$$

Expanding the right-hand side (RHS) necessitates the computation of the following quantities:

a. The expected value of x^k given all the sequences:

$$x^{k|K} \equiv \int p(x|y, \Sigma)x^k d^K x$$

b. The covariance between x^k and x^{k+1} given all the sequences:

$$W^{k,k+1|K} \equiv \int p(x|y, \Sigma)\left(x^k - x^{k|K}\right) \cdot \left(x^{k+1} - x^{k+1|K}\right) d^K x$$

c. ~~The expected value of the square of x^k given all the sequences:~~

$$\cancel{\int p(x|y,\Sigma) {x^k}^2 d^K x}$$

d. The expected value of the logarithm of the sum over all possible values of y_m given all the sequences:

$$\int p(x|y,\Sigma) \log\left(\sum_{m=1}^{n(k)} e^{x^k_{\{y_m, y_{m-1}\}}}\right) d^K x$$

These quantities will be computed in the following steps:

1. E step – forward filter: This step involves computing the expected sufficient statistics of the hidden states (in this case, x^k) given the observed data up to time k.
2. E step – filter smoothing: After the forward filter, we'll refine our estimates of the hidden states by incorporating observations from the entire sequence.
3. M step: Here, we'll maximize the expected log-likelihood with respect to the parameters. This involves updating our estimates of the parameters based on the expected sufficient statistics computed in the E step.

Each of these steps will involve specific computations and algorithms to accurately estimate the parameters and hidden states.

---

1. We'll start by describing the forward filter step, We want to compute the following:

$$x^{k|k-1}, x^{k|k}, W^{k|k-1}, W^{k|k}$$

These are the mean values and covariance matrices used in the forward filtering approach.

The idea is to start with

$$x_{1|0} = x_0 \;\&\; W_{0|0} = 0$$

and iterate the following steps:

From the Chapman-Kolmogorov identity

$$\# \; p(x^k|y^{1:k-1},\Sigma) = \int p(x^k|x^{k-1},\Sigma)\cdot p(x^{k-1}|y^{1:k-1},\Sigma)dx^{k-1}$$

we assume all densities to be gaussian;

$p(x^k|y^{1:k-1},\Sigma) \sim \mathcal{N}\left(x^{k|k-1}, W^{k|k-1}\right)$, $p(x^{k-1}|y^{1:k-1},\Sigma) \sim \mathcal{N}\left(x^{k-1|k-1}, W^{k-1|k-1}\right)$, $p(x^k|x^{k-1},\Sigma) \sim \mathcal{N}(x^{k-1},\Sigma)$

The convolution of two gaussians is also a gaussian so we get #:

$$x^{k|k-1} = x^{k-1|k-1};\, W^{k|k-1} = W^{k-1|k-1} + \Sigma$$

Next, we incorporate the observed k'th sequence of letters:

We use bayes law and the independence of $y^k and\ y^{1:k-1} given\ x^k$ to incorporate the observed letters in the sequence:

$$\#\#\ p(x^k|y^{1:k},\Sigma) = \frac{p(y^{1:k-1}|x^k,\Sigma)\cdot p(x^k|\Sigma)}{p(y^{1:k-1})} = \frac{p(y^k|x^k,\Sigma)\cdot p(y^{1:k-1}|x^k,\Sigma)\cdot p(x^k|\Sigma)}{p(y^k|y^{1:k-1})\cdot p(y^{1:k-1})}$$

$$= \frac{p(x^k|\theta^k,\Sigma)\cdot p(y^{1:k-1})\cdot p(x^k|y^{1:k-1},\Sigma)}{p(y^k|y^{1:k-1})\cdot p(y^{1:k-1})}$$

We end up with:

$$\#\#\ \frac{p(y^k|x^k,\Sigma)\cdot p(x^k|y^{1:k-1},\Sigma)}{p(y^k|y^{1:k-1},\Sigma)} = p(x^k|y^{1:k},\Sigma) \sim \mathcal{N}(x^{k|k},W^{k|k})$$

With:

$$p(y^k|x^k,\Sigma) = \prod_{m=1}^{n(k)} f\left(x^k_{\{y_m,y_{m-1}\}}\right)\ ;\ p(x^k|y^{1:k-1},\Sigma) \sim \mathcal{N}\left(x^{k|k-1}\ ,W^{k|k-1}\right)$$

We take the log of both sides of ## and receive **(2):**

$$\$ - \frac{1}{2}\left(x^k - x^{k|k}\right)^T W^{-1}_{k|k}\left(x^k - x^{k|k}\right) = \sum_{m=1}^{n(k)} \left[\log\left(f\left(x^k_{\{y_m,y_{m-1}\}}\right)\right)\right] - \frac{1}{2}\left(x^k - x^{k|k-1}\right)^T W^{-1}_{k|k-1}\left(x^k - x^{k|k-1}\right)$$

Where $\$ = \log p\left(y^k\middle|y^{1:k-1},\Sigma\right)$ independent of x.

This is where the use of the softmax function comes in handy, we defined:

$$\theta_i = f\left(x^k_{\{y_m,y_{m-1}\}}\right) \equiv \frac{e^{x^k_{\{y_m,y_{m-1}\}}}}{Z_{y_{m-1}}}$$

Let's look at the derivative of the log of this function with respect to the parameters:

$$\frac{\partial\left(\log f\left(x^k_{a,b}\right)\right)}{\partial x^k_{c,d}} \equiv \frac{\partial \log\left(f^k_{a,b}\right)}{\partial x^k_{c,d}} = \frac{\partial}{\partial x^k_{c,d}} \log\left(\frac{e^{x^k_{a,b}}}{\mathrm{Z_b}}\right) = \frac{\partial}{\partial x^k_{c,d}}\left(\mathrm{x}^k_{a,b} - \log\left(\sum_a e^{x^k_{a,b}}\right)\right)$$

$$= \delta_{a,b,c,d} - \frac{1}{\mathrm{Z_b}}\left(\frac{\partial}{\partial x^k_{c,d}}\left(\sum_a e^{x^k_{a,b}}\right)\right) = \delta_{a,b,c,d} - \frac{e^{x^k_{a,b}}\delta_{a,b,c,d}}{\mathrm{Z_b}} \equiv$$

$$\boldsymbol{J_{a,b,c,d} = \delta_{a,b,c,d} - f\left(x^k_{c,d}\right)}$$

Which is the log-softmax's Jacobian, it is a bit confusing but the notation $[a,b,c,d]$ denotes the pairs (a,b), (c,d) which corresponds with the parameters (e.g. $(a,b) \to$ ('a', 1) and $(c,d) \to$ ('a', 'b') for the parameters x_{a1} & x_{ab}, '1' being the start of the sequence), as this gets even more confusing, we will switch to $(a,b) \to i$ and $(c,d) \to j$ for the rest of the log-softmax derivation.

$$\boldsymbol{J_{i,j} = \delta_{i,j} - f\left(x_j^k\right)}$$

Let's also look at the second derivative:

$$\frac{\partial^2}{\partial x_l \partial x_j} \log\left(f(x_i)\right) = \frac{\partial}{\partial x_l}\left(\delta_{i,j} - f\left(x_j\right)\right) = -\frac{\partial f\left(x_j\right)}{\partial x_l} = -f\left(x_j\right) \cdot \frac{\partial \log\left(f\left(x_j\right)\right)}{\partial x_l}$$

$$\equiv \boldsymbol{H_{i,l,j} = -f\left(x_j\right)\left(\delta_{j,l} - f(x_l)\right)}$$

Producing the log-softmax's Hessian, which is simply given by the derivative of the log and is very efficient to calculate.

Let's look at the derivative of (2) with regards to x:

$$\frac{\partial}{\partial x^k} \to \left[W_{k|k}^{-1}\left(x^k - x^{k|k}\right)\right] = \left[W_{k|k-1}^{-1}\left(x^k - x^{k|k-1}\right)\right] - \sum_{m=1}^{n(k)} \frac{\partial}{\partial {x_{a,b}}^k}\left[\log f\left(x_{\{y_m, y_{m-1}\}}^k\right)\right]$$

$$= \left[W_{k|k-1}^{-1}\left(x^k - x^{k|k-1}\right)\right] - \sum_{m=1}^{n(k)}\left(\delta_{y_m, y_{m-1}, a, b} - f\left(x_{a,b}^k\right)\right)$$

This should hold for $x^k = x^{k|k-1}$ so we insert it and get:

$$(2.1)\ x^{k|k} = x^{k|k-1} + W_{k|k}\sum_{m=1}^{n(k)}\left(\delta_{y_m, y_{m-1}, a, b} - f\left(x_{a,b}^{k|k-1}\right)\right)$$

We can derive (2) again and receive the update step for $W_{k|k}^{-1}$ in the same manner:

$$\frac{\partial^2}{\partial^2 x^k} \to \left[W_{k|k}^{-1}\right] = \left[W_{k|k-1}^{-1}\right] - \sum_{m=1}^{n(k)} \frac{\partial^2}{\partial x_{c,d}^k \partial x_{a,b}^k}\left[\log f\left(x_{\{y_m, y_{m-1}\}}^k\right)\right]$$

$$\left[W_{k|k}^{-1}\right] = \left[W_{k|k-1}^{-1}\right] + \sum_{m=1}^{n(k)} \boldsymbol{f\left(x_{c,d}^k\right)\left(\delta_{a,b,c,d} - f\left(x_{a,b}^k\right)\right)}$$

And we can inset $x^k = x^{k|k-1}$again and get:

$$(2.2)\ \left[W_{k|k}^{-1}\right] = \left[W_{k|k-1}^{-1}\right] + \sum_{m=1}^{n(k)} \boldsymbol{f\left(x_{c,d}^{k|k-1}\right)\left(\delta_{a,b,c,d} - f\left(x_{a,b}^{k|k-1}\right)\right)}$$

This concludes the 1st step of the algorithm (forward filter), the procedure is:

a. **Start with random $x_{0|0}$ and $W_{0|0} = 0$**
b. **Compute next step with: $x^{k|k-1} = x^{k-1|k-1}; W^{k|k-1} = W^{k-1|k-1} + \Sigma$**
c. **Compute (2.1) and (2.2) and receive $x^{k|k}, W^{k|k}\ \forall\ k \in K$**

2nd step will be the smoothing part. The idea is to refine our estimates of previous states, in the light of later observations.

After performing the 1st step we obtained $x^{k|k}, W^{k|k}\ \forall\ k \in K$; We start the next step by assuming $x^{k|l}, x^{k+1|l} (l > k)$ both maximize the Gaussian joint distribution:

$$p(x^k, x^{k+1}, y^{k+1:l}|y^{1:k}, \Sigma)$$

from this point:

$$\#\#\#\ p(x^k, x^{k+1}|y^{1:l}, \Sigma) = \frac{p(x^k, x^{k+1}, y^{1:l}|\Sigma)}{p(y^{1:l})} = \frac{p(y^{1:k})}{p(y^{1:l})} \cdot p(x^k, x^{k+1}, y^{k+1:l}|y^{1:k}, \Sigma)$$

$$= \frac{p(y^{1:k})}{p(y^{1:l})} \cdot p(y^{k+1:l}|x^k, x^{k+1}, y^{1:k}, \Sigma) \cdot p(x^k, x^{k+1}|y^{1:k}, \Sigma)$$

Since the process is Markovian, we will use:

$$(*)\ p(y^{k+1:l}|x^k, x^{k+1}, y^{1:k}, \Sigma) = p(y^{k+1:l}|x^{k+1}, \Sigma)$$

$$(**)\ p(x^k, x^{k+1}|y^{1:k}, \Sigma) = p(x^{k+1}|x^k, y^{1:k}, \Sigma) \cdot p(x^k|y^{1:k}, \Sigma) = p(x^{k+1}|x^k, \Sigma) \cdot p(x^k|y^{1:k}, \Sigma).$$

We can put it back in ### above and receive (3):

$$(3)\ p(x^k, x^{k+1}|y^{1:l}, \Sigma) = c(x^{k+1}) p(x^{k+1}|x^k, \Sigma) \cdot p(x^k|y^{1:k}, \Sigma)$$

$$with\ c(x^{k+1}) = \frac{p(y^{1:k})}{p(y^{1:l})} \cdot p(y^{k+1:l}|x^{k+1}, \Sigma)\ which\ is\ indpendent\ of\ x^k$$

Now we introduce our assumptions $p(x^{k+1}|x^k, \Sigma) \sim \mathcal{N}(x^k, \Sigma)$ and $p(x^k|y^{1:k}, \Sigma) \sim \mathcal{N}\left(x^{k|k}, W^{k|k}\right)$, we also require that $x^{k|l}, x^{k+1|l}$ minimize log(3):

$$\log(3): d(x^{k+1}) + (x^{k+1} - x^k)^T \Sigma^{-1} (x^{k+1} - x^k) + \left(x^k - x^{k|k}\right)^T \mathrm{W}_{\mathrm{k|k}}^{-1} \left(x^k - x^{k|k}\right)$$

Where $d = \log\left(c(x^{k+1})\right)$ and is again independent of x^k.

If we assume that $x^{k+1|l}$ is known, we only need to minimize:

$$\log(3): \left(x^{k+1|l} - x^k\right)^T \Sigma^{-1} \left(x^{k+1|l} - x^k\right) + \left(x^k - x^{k|k}\right)^T \mathrm{W}_{\mathrm{k|k}}^{-1} \left(x^k - x^{k|k}\right)$$

w.r.t x^k. So, we take the derivative, set it to zero and note that all matrices are symmetric:

$$0 = \left(-\Sigma^{-1}\left(x^{k+1|l} - x^k\right) - \left(x^{k+1|l} - x^k\right)^T \Sigma^{-1} + \mathrm{W}_{\mathrm{k|k}}^{-1}\left(x^k - x^{k|k}\right) + \left(x^k - x^{k|k}\right)^T \mathrm{W}_{\mathrm{k|k}}^{-1}\right)|_{x^k = x^{k|l}}$$

$$= \left(\Sigma^{-1} + W_{k|k}^{-1}\right) \cdot x^{k|l} - \Sigma^{-1} \cdot x^{k+1|l} - W_{k|k}^{-1} \cdot x^{k|k}$$

Which solves to:

$$x^{k|l} = \left(\Sigma^{-1} + W_{k|k}^{-1}\right)^{-1} \cdot \left(\Sigma^{-1} \cdot x^{k+1|l} + W_{k|k}^{-1} \cdot x^{k|k}\right)$$

Based on the identity: $(I + PM^T R^{-1} M)^{-1} = I - PM^T (MPM^T + R)^{-1} M$ for square matrices P,M,R, we identify $M = I, R = \Sigma, P = W_{k|k}$:

$$x^{k|l} = \left(\Sigma^{-1} + W_{k|k}^{-1}\right)^{-1} \cdot \Sigma^{-1} \cdot x^{k+1|l} + \left(\Sigma^{-1} + W_{k|k}^{-1}\right)^{-1} \cdot W_{k|k}^{-1} \cdot x^{k|k}$$

$$= W^{k|k} \cdot \left(W^{k|k} + \Sigma\right)^{-1} x^{k+1|l} + \left(I - W^{k|k}\left(W^{k|k} + \Sigma\right)^{-1}\right) x^{k|k}$$

With $x^{k|k} = x^{k+1|k}$ *and* $W^{k+1|k} = W^{k|k} + \Sigma$ we finally get:

$$(4) \quad x^{k|l} = x^{k|k} + M^k\left(x^{k+1|l} - x^{k+1|k}\right); M^k = W^{k|k} W_{k+1|k}^{-1}$$

As the smoothing step for parameters x. next we develop the smoothing for the covariance matrix, we begin by defining $x^{\widetilde{k|q}} = x^k - x^{k|q}$ and subtracting (4) from x^k:

$$x^{\widetilde{k|l}} + M^k x^{k+1|l} = x^{\widetilde{k|k}} + M^k x^{k|k}$$

we square both sides and compute the expectation (<var>) to get:

$$W^{k|l} + M^k \underbrace{\left\langle x^{k+1|l} x^{\widetilde{k|l}^T} \right\rangle}_{=0} + \underbrace{\left\langle x^{\widetilde{k|l}} x^{k+1|l^T} \right\rangle}_{=0} M^{k^T} + M^k \left\langle x^{k+1|l} x^{k+1|l^T} \right\rangle M^{k^T}$$

$$= W^{k|k} + M^k \underbrace{\left\langle x^{k|k} x^{\widetilde{k|k}^T} \right\rangle}_{=0} + \underbrace{\left\langle x^{\widetilde{k|k}} x^{k|k^T} \right\rangle}_{=0} M^{k^T} + M^k \left\langle x^{k|k} x^{k|k^T} \right\rangle M^{k^T}$$

By identity $\left\langle x^k x^{k^T} \right\rangle = W^{k|q} + \langle x^{k|q} x^{k|q^T} \rangle$:

$$(5) \; W^{k|l} = W^{k|k} + M^k\left(W^{k+1|l} - W^{k+1|k}\right) M^{k^T}$$

This identity also gives us:

$$E\left(x^{k^2} \middle| y^k, \Sigma\right) = \left\langle x^k x^{k^T} \right\rangle = W^{k|K} + x^{k|K} x^{k|K^T}$$

This concludes the smoothing algorithm:

a. **Start with $x^{K|K}$ and $W^{K|K}$ we received from the filtering algorithm.**
b. **Compute (4) and (5) from K to 1, and receive $x^{k|K}$ & $W^{k|K}$ $\forall\, k \in K$**

We have 2 more quantities to compute for the E step:

$$W^{k,u|K} \equiv \int p(x|y,\Sigma)\left(x^k - x^{k|K}\right)\cdot\left(x^u - x^{u|K}\right)d^K x\,;\ 1 \le k \le u \le K$$

This can be given by the orthogonal projection of x^k on the subspace $y^1, \dots, y^k, x^{k+1} - x^{k+1|k}, \epsilon^{s+1}, \dots, \epsilon^K (\epsilon^{s+1} = x^{s+2} - x^{s+1})$ which is $\hat{x}^k = x^{k|k} + M^k\left(x^{k+1} - x^{k+1|k}\right)$. the proof for this due to $\langle t, v\rangle = E(tv)$.

$$\hat{x}^k = x^{k|k} + Cov\left(x^k, x^{k+1} - x^{k+1|k}\right)W_{k+1|k}^{-1}\left(x^{k+1} - x^{k+1|k}\right)$$

And because $x^{k+1} - x^{k+1|k}, \epsilon^{s+1}, \dots, \epsilon^K$ have mean zero and are uncorrelated to each other and to the observations $\{y\}^k$ the projection is then:

$$Cov\left(x^k, x^{k+1} - x^{k+1|k}\right)W_{k+1|k}^{-1} = Cov(x^k, x^k)W_{k+1|k}^{-1} = M^k$$

Hence,

$$W^{k,u|K} \equiv Cov\left[x^k - x^{k|K}, x^u - x^{u|K}\right] = Cov\left[x^k - x^{k|K}, x^u\right] = Cov\left[x^k - \hat{x}^k + \hat{x}^k - x^{k|K}, x^u\right]$$
$$= Cov\left[\hat{x}^k - x^{k|K}, x^u\right] = Cov\left[M^k\left(x^{k+1} - x^{k+1|K}\right), x^u\right] = M^k W^{k+1,u|K}$$

Replace u with k+1 to get:

$$(6)\ W^{k,k+1|K} = M^k W^{k+1,k+1|K}$$

We now must calculate the second quantity:

$$\left\langle \log\left(\sum_{i=1}^{|A|} e^{x^K_{\{y_m = A_i, y_{m-1}\}}}\right)\right\rangle \equiv \int p(x|y,\Sigma)\log\left(\sum_{i=1}^{|A|} e^{x^k_{\{y_m = A_i, y_{m-1}\}}}\right)d^K x$$

But we might even do better: we can formulate $E[\log\left(f\left(x^k_{\{a,b\}}\right)\right) || \{y\}^K, x_0, \Sigma]$ using a second order expansion[1] which we already derived twice:

$$\langle \log\left(f\left(x^k_{\{a,b\}}\right)\right)\rangle \approx \log\left(f\left(\mu_{x^k_{\{a,b\}}}\right)\right) - \frac{1}{2}f\left(\mu_{x^k_{\{c,d\}}}\right)\left(1 - f\left(\mu_{x^k_{\{c,d\}}}\right)\right)\sigma_{x^k_{\{c,d\}}}$$

Where μ, σ are the mean and standard deviation assuming $x^k_{\{a,b\}} \sim \mathcal{N}\left(\mu_{x^k}, \sigma_{x^k}\right)$, we can further approximate μ, σ to be:

$$\mu \approx E[x^k || \{y\}^K, \Sigma] = x^{k|K}\ and\ Cov[x^k, x^k || \{y\}^K, \Sigma] = W^{k,k|K} \rightarrow \sigma \approx W_{k,k|K}^{-\frac{1}{2}}$$

[1]Semi-analytical approximations to statistical moments of sigmoid and softmax mappings of normal variables, J. Daunizeau, Brain and Spine Institute, Paris, France. https://arxiv.org/pdf/1703.00091.pdf

$$\Rightarrow (7)\ \langle \log(f(x^{k|K})\rangle \approx \log\left(f\left(x^{k|K}\right)\right) - \frac{1}{2} f\left(x^{k|K}\right)\left(1 - f\left(x^{k|K}\right)\right) W_{k,k|K}^{-\frac{1}{2}}$$

One issue which arises from this is that the approximation does not ensure a proper normalization (i.e. $e^{\left\langle \log\left(f\left(x_{\{a,b\}}^{k}\right)\right)\right\rangle} + e^{\left\langle \log\left(1-f\left(x_{\{a,b\}}^{k}\right)\right)\right\rangle} = 1$ may not be satisfied) .

We can now move forward to the M-step.

The M – step requires us to find the values which maximize the expected value of the likelihood function from iteration i to be used as the new parameters in iteration i+1, so finding:

$$\psi^{i+1} = \left[x_0^{i+1}, \Sigma^{i+1}\right];\ \psi^{i+1} = \underset{\psi^i}{argmax}\left(Q\left(\psi, \psi^i\right)\right)$$

Updating Σ:

$$\Sigma^{i+1} = \underset{\Sigma}{argmax}\left(Q\left(\psi, \psi^i\right)\right)$$

$$= \underset{\Sigma}{argmax}\left[E\left[\sum_{k=1}^{K} \frac{1}{2}(x^k - x^{k-1})^T \Sigma^{-1} (x^k - x^{k-1})\ ||\{y\}, x_0^i, \Sigma^i\right] + \frac{K}{2}\log|\Sigma|\right]$$

We'll derive the expression by σ_j ; $\Sigma_{jj} = diag(\sigma_j)$:

$$\frac{\partial Q}{\partial \sigma_j} = \frac{K}{\sigma_j} + E_{p(\{x\}|\{y\},\Sigma)}\left[\sum_{k=1}^{K}(x^k - x^{k-1})^T \begin{pmatrix} 0 & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & -\frac{1}{\sigma_j^2} \end{pmatrix} (x^k - x^{k-1})\right]$$

$$= \frac{K}{\sigma_j} - \frac{1}{\sigma_j^2} E_{p(\{x\}|\{y\},\Sigma)}\left[\sum_{k=1}^{K}(x^k - x^{k-1})_j{}^T (x^k - x^{k-1})_j\right]$$

We will equate this to 0 and receive:

$$(8)\ \sigma_j = \frac{1}{K} E_{p(\{x\}|\{y\},\Sigma)}\left[\sum_{k=1}^{K}(x^k - x^{k-1})_j{}^T (x^k - x^{k-1})_j\right]$$

Updating x_0:

Any Development of the point process filtering and SSGLM done by Yarden or by Czanner et al. 2008 (Eden & Brown 2004 doesn't mention EM algorithms at all) mentions a closed form solution for the initial state of this form:

We will look at the parts of Q which contains instances of x^0:

$$\Theta(x^0,\Sigma) = \frac{-1}{2} E[(x^1 - x^0)^T \Sigma^{-1} (x^1 - x^0) \,||\{y\}, x_0^i, \Sigma^i]$$

Thus, the solution for x_0 will be (following the same steps):

$$\frac{\partial\Theta}{\partial x_{0q}} = E_{p(\{x\}|\{y\},\Sigma)}[\Sigma^{-1} (x^1 - x^0)]_q = 0$$

$$(9)\ x_0^{i+1} = E_{p(\{x\}|\{y\},\Sigma)} x^1$$

This was done by taking k = 1 in Q, which ignores the part of the innovation term containing λ_1, if that is the case, without ignoring this term, we will get:

$$\Theta(x^0,\Sigma) = \frac{-1}{2} E[(x^1 - x^0)^T \Sigma^{-1} (x^1 - x^0) \,||\{y\}, x_0^i, \Sigma^i] + E\left[\sum_{m=1}^{n(k=1)} \left(\log\left(f\left(x_{\{y_m,y_{m-1}\}}^{k=1}\right)\right)\right) ||\{y\}, x_0^i, \Sigma^i\right]$$

And we will have to derive:

$$\frac{\partial\Theta}{\partial x_{0q}} = E_{p(\{x\}|\{y\},\Sigma)}\left[\Sigma^{-1} (x^1 - x^0)_q + \sum_{m=1}^{n(k=1)} \left(\delta_{y_m,y_{m-1},0,q} - f\left(x_{0,q}^{k=1}\right)\right)\right] = 0$$

Which might not have a closed form solution.

References:

1. Development of the point-process filtering and SS-GLM, Yarden Cohen, 2016.
2. Analysis of Between-Trial and Within-Trial of Neural Spiking Dynamics, Czanner et al. 2008.
3. Dynamic Analysis of Neural Encoding by Point Process Adaptive Filtering, Eden at al. 2004.
4. Comparison of Expectation – Maximization based parameter estimation using Particle Filter, Unscented and Extended Kalman Filtering techniques, Chitralekha et al. 2009