BAYESIAN STATE ANALYSIS ON LINEAR GAUSSIAN DYNAMICAL SYSTEMS

Outline:

- **Bayesian linear regression**
- **Bayesian state analysis on linear Gaussian dynamical systems**
    - **Kalman filter**
    - **RTS (Rauch-Tung-Striebel) smoother**
    - **Backward sampler**
- **Analytical approximation on Bayesian state analysis of nonlinear dynamical systems**
    - **Uncertainty propagation**
    - **Extended Kalman filter**
    - **Unscented Kalman filter**

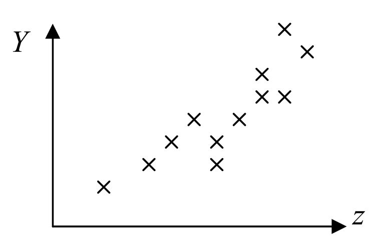
Bayesian linear regression

Introduction:

Consider the following model:

$$Y = aX + E$$

where $E \sim N(0, \Sigma)$, $X \sim N(\mu_X, \Sigma_X)$ and $E \perp X$. We observe $Y = \hat{Y}$ (data), and we'd like to know $f(x \mid \hat{Y})$. It is called linear regression because the problem is linear in the uncertain variable X. Also, note that all uncertainties are Gaussian. Although this problem is linear, but it can actually handle nonlinearity, e.g.:



$z_i \in R, \hat{Y}_i \in R, i = 1 \cdots N$ $\qquad$ $g_X(z) = X_0 + X_1 z + X_2 z^2$

$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_N \end{bmatrix} = \begin{bmatrix} 1 & z_1 & z_1^2 \\ & \vdots & \\ 1 & z_N & z_N^2 \end{bmatrix} \begin{bmatrix} X_0 \\ X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} E_1 \\ \vdots \\ E_N \end{bmatrix} \Rightarrow Y = aX + E$$

It turns out that $f\left(x \mid \hat{Y}\right)$ is also Gaussian, i.e. $N\left(\mu_X, \Sigma_X\right)$ is a conjugate prior. To see this, observe that

$$f\left(x \mid \hat{Y}\right) = \frac{f\left(\hat{Y} \mid x\right) f\left(x\right)}{f\left(\hat{Y}\right)} \propto e^{-\frac{1}{2}\left(\hat{Y}-ax\right)^T \Sigma^{-1}\left(\hat{Y}-ax\right)} e^{-\frac{1}{2}\left(x-\mu_X\right)^T \Sigma_X^{-1}\left(x-\mu_X\right)}$$

is log-quadratic in X, so $f\left(x \mid \hat{Y}\right)$ is Gaussian. Therefore, $f\left(x \mid \hat{Y}\right) = N\left(\mu_{X|\hat{Y}}, \Sigma_{X|\hat{Y}}\right)$, where

$$\mu_{X|\hat{Y}} = \mu_X + Cov\left(X,Y\right) Cov\left(Y\right)^{-1} \left(\hat{Y} - E\left(Y\right)\right) = \mu_X + \Sigma_X a^T \left(a \Sigma_X a^T + \Sigma\right)^{-1} \left(\hat{Y} - a\mu_X\right)$$

$$\Sigma_{X|\hat{Y}} = \Sigma_X - Cov\left(X,Y\right) Cov\left(Y\right)^{-1} Cov\left(Y,X\right) = \Sigma_X - \Sigma_X a^T \left(\Sigma + a\Sigma_X a^T\right)^{-1} a\Sigma_X$$

Proof:

Note that at the mean value of the Gaussian PDF $f\left(x \mid \hat{Y}\right)$, $\nabla_x f(x \mid \hat{Y}) = 0$. That means the mean of the Gaussian PDF can be found by solving $\nabla_x f(x \mid \hat{Y}) = 0$. Also, the covariance matrix of the Gaussian PDF $f\left(x \mid \hat{Y}\right)$ is equal to $\left(-\nabla_x^2 \log[f\left(x \mid \hat{Y}\right)]\right)^{-1}$. Therefore, we have

$$\begin{aligned}
\mu_{X|\hat{Y}} &= \left(a^T \Sigma^{-1} a + \Sigma_X^{-1}\right)^{-1} \left(\Sigma_X^{-1} \mu_X + a^T \Sigma^{-1} \hat{Y}\right) \\
&= \left(\Sigma_X a^T \Sigma^{-1} a + I\right)^{-1} \Sigma_X \left(\Sigma_X^{-1} \mu_X + a^T \Sigma^{-1} \hat{Y}\right) \\
&= \left(\Sigma_X a^T \Sigma^{-1} a + I\right)^{-1} \left(\mu_X + \Sigma_X a^T \Sigma^{-1} \hat{Y}\right) \\
&= \left(\Sigma_X a^T \Sigma^{-1} a + I\right)^{-1} \left(\mu_X + \Sigma_X a^T \Sigma^{-1} \hat{Y} - \Sigma_X a^T \Sigma^{-1} a \mu_X + \Sigma_X a^T \Sigma^{-1} a \mu_X\right)
\end{aligned}$$

$$= \left(\Sigma_X a^T \Sigma^{-1} a + I\right)^{-1} \left[\left(\Sigma_X a^T \Sigma^{-1} a + I\right)\mu_X + \Sigma_X a^T \Sigma^{-1}\left(\hat{Y} - a\mu_X\right)\right]$$

$$= \mu_X + \left(\Sigma_X a^T \Sigma^{-1} a + I\right)^{-1} \Sigma_X a^T \Sigma^{-1}\left(\hat{Y} - a\mu_X\right)$$

$$= \mu_X + \left(\Sigma_X a^T \Sigma^{-1} a + I\right)^{-1} \Sigma_X a^T \Sigma^{-1}\left(\hat{Y} - a\mu_X\right)$$

$$\because (PQ + I)^{-1} P = P(QP + I)^{-1}$$

$$= \mu_X + \Sigma_X a^T \left(a\Sigma_X a^T + \Sigma\right)^{-1}\left(\hat{Y} - a\mu_X\right)$$

$$\Sigma_{X|\hat{Y}} = \left(-\nabla_x^2 \log\left[f\left(x \mid \hat{Y}\right)\right]\right)^{-1} = \left(a^T \Sigma^{-1} a + \Sigma_X^{-1}\right)^{-1}$$

$$\because \left(VC^{-1}V^T + A^{-1}\right)^{-1} = A - AV\left(C + V^T AV\right)^{-1} V^T A$$

$$= \Sigma_X - \Sigma_X a^T \left(\Sigma + a\Sigma_X a^T\right)^{-1} a\Sigma_X$$

Bayesian state analysis on linear Gaussian dynamical systems

Introduction:

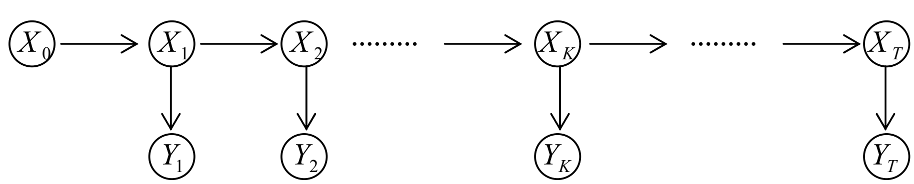
For linear dynamical systems with Gaussian uncertainties, the Bayesian state PDF updating can be done analytically. This is because the updated state PDF is Gaussian due to the conjugate priors. Moreover, the state PDF can be updated sequentially. The resulting Bayesian state estimation algorithm is called the Kalman filter.

Consider the following discrete-time linear state-space dynamical system:

$$X_k = a_{k-1}X_{k-1} + b_{k-1}u_{k-1} + W_{k-1} \qquad X_0 \sim N(\mu_{0|0}, \Sigma_{0|0}) \qquad \text{(state equation)}$$

$$Y_k = c_k X_k + d_k u_k + V_k \qquad k = 1, \ldots, T \qquad \text{(observation equation)}$$

where $u_k =$known system input at time *k*, $W_k, V_k =$modeling error, $W_k \sim N\left(0, \Sigma_W\right)$, $V_k \sim N\left(0, \Sigma_V\right)$, $W_i \perp W_j\left(i \neq j\right)$, $V_i \perp V_j\left(i \neq j\right)$, $W_i \perp V_j$, $X_0 \perp W_j$, $X_0 \perp V_j$; a_k, b_k, c_k, d_k, Σ_W, Σ_V, $\mu_{0|0}, \Sigma_{0|0}$ are known vectors and matrices. This model class creates a hidden Markov chain.



Note that under this setting, all uncertain variables are jointly Gaussian; also, the model class is linear in all uncertain variables. This implies that if we observe data $\hat{Y}=\{\hat{Y}_1,...,\hat{Y}_T\}$, the posterior PDF $f(x\,|\,\hat{Y})\equiv f(x_0,...,x_T\,|\,\hat{Y})$ is jointly Gaussian, where

$$f(x_0,...,x_T\,|\,\hat{Y})=\frac{f(\hat{Y}\,|\,x_0,...,x_T)f(x_0,...,x_T)}{f(\hat{Y})}=\frac{\prod_{k=1}^{T}f(\hat{Y}_k\,|\,x_k)\cdot\prod_{k=1}^{T}f(x_k\,|\,x_{k-1})}{f(\hat{Y})}$$

$$=const\cdot\prod_{k=1}^{T}e^{-\frac{1}{2}(\hat{Y}_k-c_kx_k-d_ku_k)^T\Sigma_V^{-1}(\hat{Y}_k-c_kx_k-d_ku_k)}\cdot\prod_{k=1}^{T}e^{-\frac{1}{2}(x_k-a_{k-1}x_{k-1}-b_{k-1}u_{k-1})^T\Sigma_W^{-1}(x_k-a_{k-1}x_{k-1}-b_{k-1}u_{k-1})}$$

$$\log\left[f(x_0,...,x_T\,|\,\hat{Y})\right]$$

$$=const-\frac{1}{2}\sum_{k=1}^{T}\left[(\hat{Y}_k-c_kx_k-d_ku_k)^T\Sigma_V^{-1}(\hat{Y}_k-c_kx_k-d_ku_k)\right]$$

$$-\frac{1}{2}\sum_{k=1}^{T}\left[(x_k-a_{k-1}x_{k-1}-b_{k-1}u_{k-1})^T\Sigma_W^{-1}(x_k-a_{k-1}x_{k-1}-b_{k-1}u_{k-1})\right]$$

But how do we obtain the mean and the covariance matrix of $f(x\,|\,\hat{Y})$? The mean may be simply, i.e. differentiate $\log\left[f(x_0,...,x_T\,|\,\hat{Y})\right]$ w.r.t. x_k and solve for zero, we get $E(X_k\,|\,\hat{Y})$. But how about the covariance matrix or even $Cov(X_k\,|\,\hat{Y})$? Get the Hessian of $-\log\left[f(x_0,...,x_T\,|\,\hat{Y})\right]$ and calculate the inverse? No, we don't want to do so since the inversion will be on a huge matrix.

Another class of interesting problems is to obtain $f(x_k\,|\,\hat{Y}_{1:k})$, where $\hat{Y}_{1:k}=\{\hat{Y}_1,...,\hat{Y}_k\}$. One can see that this posterior PDF is also Gaussian, where the mean and covariance matrix can, again, be obtained by solving the gradient of $\log\left[f(x_0,...,x_k\,|\,\hat{Y}_{1:k})\right]$ for zero and also by calculating the inverse of the Hessian of $-\log\left[f(x_0,...,x_k\,|\,\hat{Y}_{1:k})\right]$. But again, we are required to do an inversion on a huge matrix.

Terminology:

A Bayesian filtering problem is to obtain $f(x_k\,|\,\hat{Y}_{1:k})$ for all *k*, while a Bayesian smoothing problem is to obtain $f(x_k \mid \hat{Y})$ for all k.

Kalman filter:

Kalman filter provides a smart way of calculating the mean and covariance matrix of $f(x_k \mid \hat{Y}_{1:k})$ without inverting huge matrices. Basically, Kalman filter is an algorithm that derives $f(x_{k+1} \mid \hat{Y}_{1:k+1})$ based on the prior $f(x_k \mid \hat{Y}_{1:k})$ and the new data $\hat{Y}_{k+1}$, or equivalently, derives $E(X_{k+1} \mid \hat{Y}_{1:k+1})$ and $Cov(X_{k+1} \mid \hat{Y}_{1:k+1})$ based on $E(X_k \mid \hat{Y}_{1:k})$, $Cov(X_k \mid \hat{Y}_{1:k})$ and the new data $\hat{Y}_{k+1}$. One can see once this algorithm is finished, we can obtain $f(x_k \mid \hat{Y}_{1:k})$ recursively starting from $f(x_0 \mid \hat{Y}_{1:0}) \equiv f(x_0) = N(\mu_{0|0}, \Sigma_{0|0})$. Let us denote $\mu_{p|q} \equiv E(X_p \mid \hat{Y}_{1:q})$ and $\Sigma_{p|q} \equiv Cov(X_p \mid \hat{Y}_{1:q})$.

Algorithm: Kalman filter

1. Starting from $\mu_{0|0}$ and $\Sigma_{0|0}$
2. $$\begin{aligned} \mu_{k+1|k} &= a_k \mu_{k|k} + b_k u_k \\ \Sigma_{k+1|k} &= a_k \Sigma_{k|k} a_k^T + \Sigma_W \end{aligned}$$ (Uncertainty propagation)
3. $$\begin{aligned} &\mu_{k+1|k+1} \\ &= \mu_{k+1|K} + Cov\left(X_{k+1}, Y_{k+1} \mid \hat{Y}_{1:k}\right) Cov\left(Y_{k+1} \mid \hat{Y}_{1:k}\right)^{-1} \left(\hat{Y}_{k+1} - E\left(Y_{k+1} \mid \hat{Y}_{1:k}\right)\right) \\ &= \mu_{k+1|k} + \Sigma_{k+1|k} c_{k+1}^T \left(c_{k+1} \Sigma_{k+1|k} c_{k+1}^T + \Sigma_V\right)^{-1} \left(\hat{Y}_{k+1} - c_{k+1}\mu_{k+1|k} - d_{k+1}u_{k+1}\right) \\ &\Sigma_{k+1|k+1} \\ &= \Sigma_{k+1|k} - Cov\left(X_{k+1}, Y_{k+1} \mid \hat{Y}_{1:k}\right) Cov\left(Y_{k+1} \mid \hat{Y}_{1:k}\right)^{-1} Cov\left(Y_{k+1}, X_{k+1} \mid \hat{Y}_{1:k}\right) \\ &= \Sigma_{k+1|k} - \Sigma_{k+1|k} c_{k+1}^T \left(c_{k+1} \Sigma_{k+1|k} c_{k+1}^T + \Sigma_V\right)^{-1} c_{k+1} \Sigma_{k+1|k}^T \end{aligned}$$ (Bayesian update)

Note:

$$E\left(Y_{k+1} \mid \hat{Y}_{1:k}\right) = E\left(c_{k+1}X_{k+1} + d_{k+1}u_{k+1} + V_{k+1} \mid \hat{Y}_{1:k}\right) = c_{k+1}\mu_{k+1|k} + d_{k+1}u_{k+1}$$

$$Cov\left(Y_{k+1} \mid \hat{Y}_{1:k}\right) = Cov\left(c_{k+1}X_{k+1} + d_{k+1}u_{k+1} + V_{k+1} \mid \hat{Y}_{1:k}\right) = c_{k+1}\Sigma_{k+1|k}c_{k+1}^T + \Sigma_V$$

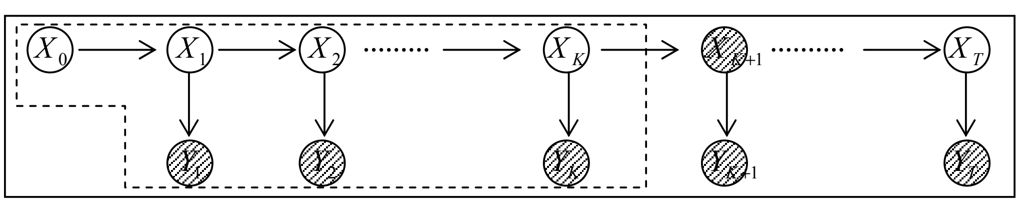
$$
\begin{aligned}
&Cov\left(X_{k+1}, Y_{k+1} \mid \hat{Y}_{1:k}\right) \\
&= E\left\{\left[X_{k+1} - E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right]\left[Y_{k+1} - E\left(Y_{k+1} \mid \hat{Y}_{1:k}\right)\right]^T \middle| \hat{Y}_{1:k}\right\} \\
&= E\left\{\left[X_{k+1} - E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right]\left[c_{k+1}X_{k+1} + d_{k+1}u_{k+1} + V_{k+1} - c_{k+1}E\left(X_{k+1} \mid \hat{Y}_{1:k}\right) - d_{k+1}u_{k+1}\right]^T \middle| \hat{Y}_{1:k}\right\} \\
&= E\left\{\left[X_{k+1} - E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right]\left[c_{k+1}X_{k+1} - c_{k+1}E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right]^T \middle| \hat{Y}_{1:k}\right\} \\
&\qquad + E\left\{\left[X_{k+1} - E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right]V_{k+1}^T \middle| \hat{Y}_{1:k}\right\} \\
&= E\left\{\left[X_{k+1} - E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right]\left[X_{k+1} - E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right]^T \middle| \hat{Y}_{1:k}\right\} c_{k+1}^T = \Sigma_{k+1|k} c_{k+1}^T
\end{aligned}
$$

RTS smoother

RTS smoother provides a smart way of calculating the mean and covariance matrix of $f(x_k \mid \hat{Y})$ (recall that $\hat{Y} = \hat{Y}_{1:T}$) without inverting huge matrices. Basically, RTS smoother starts from the results from Kalman filter and operates backwards in time, i.e. obtain $\mu_{k|T}$ and $\Sigma_{k|T}$ based on $\mu_{k+1|T}$ and $\Sigma_{k+1|T}$. One can see once this algorithm is finished, we can obtain $\mu_{k|T}$ and $\Sigma_{k|T}$ for all *k* recursively starting from $f(x_T \mid \hat{Y}) = N(\mu_{T|T}, \Sigma_{T|T})$. Note that $\mu_{T|T}$ and $\Sigma_{T|T}$ are obtained from Kalman filter.

Now we establish the backward recursive equation that relates $\mu_{k|T}$ and $\Sigma_{k|T}$ to $\mu_{k+1|T}$ and $\Sigma_{k+1|T}$. First note that

$$f\left(x_k \mid X_{k+1}, \hat{Y}_{1:T}\right) = f\left(x_k \mid X_{k+1}, \hat{Y}_{1:k}\right) \equiv N\left(\mu_{k|k}^*(X_{k+1}), \Sigma_{k|k}^*(X_{k+1})\right)$$



Moreover,

$$\mu^*_{k|k}\left(X_{k+1}\right) = \mu_{k|k} + Cov\left(X_k, X_{k+1} \mid \hat{Y}_{1:k}\right) Cov\left(X_{k+1} \mid \hat{Y}_{1:k}\right)^{-1} \left(X_{k+1} - E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right)$$

$$= \mu_{k|k} + \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} \left(X_{k+1} - \mu_{k+1|k}\right)$$

$$\Sigma^*_{k|k}\left(X_{k+1}\right) = \Sigma_{k|k} - Cov\left(X_k, X_{k+1} \mid \hat{Y}_{1:k}\right) Cov\left(X_{k+1} \mid \hat{Y}_{1:k}\right)^{-1} Cov\left(X_{k+1}, X_k \mid \hat{Y}_{1:k}\right)$$

$$= \Sigma_{k|k} - \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} a_K \Sigma_{k|k} = \Sigma^*_{k|k}$$

Note:

$$Cov\left(X_k, X_{k+1} \mid \hat{Y}_{1:k}\right)$$

$$= E\left\{\left[X_k - E\left(X_k \mid \hat{Y}_{1:k}\right)\right]\left[X_{k+1} - E\left(X_{k+1} \mid \hat{Y}_{1:k}\right)\right]^T \middle| \hat{Y}_{1:k}\right\}$$

$$= E\left\{\left[X_k - E\left(X_k \mid \hat{Y}_{1:k}\right)\right]\left[a_k X_k + b_k u_k + W_k - a_k E\left(X_k \mid \hat{Y}_{1:k}\right) - b_k u_k\right]^T \middle| \hat{Y}_{1:k}\right\}$$

$$= E\left\{\left[X_k - E\left(X_k \mid \hat{Y}_{1:k}\right)\right]\left[a_k X_k - a_k E\left(X_k \mid \hat{Y}_{1:k}\right)\right]^T \middle| \hat{Y}_{1:k}\right\} + E\left\{\left[X_k - E\left(X_k \mid \hat{Y}_{1:k}\right)\right] W_k^T \middle| \hat{Y}_{1:k}\right\}$$

$$= E\left\{\left[X_k - E\left(X_k \mid \hat{Y}_{1:k}\right)\right]\left[X_k - E\left(X_k \mid \hat{Y}_{1:k}\right)\right]^T \middle| \hat{Y}_{1:k}\right\} a_k^T = \Sigma_{k|k} a_k^T$$

Implementing the following identity:

$$E_Y\left(E_X\left(X \mid Y\right)\right) = E_X\left(X\right) \qquad Cov_X\left(X\right) = E_Y\left[Cov_X\left(X \mid Y\right)\right] + Cov_Y\left[E_X\left(X \mid Y\right)\right]$$

One can see that

$$\mu_{k|T} = E\left(X_k \mid \hat{Y}_{1:T}\right) = E\left[E\left(X_k \mid X_{k+1}, \hat{Y}_{1:T}\right) \mid \hat{Y}_{1:T}\right] = E\left[E\left(X_k \mid X_{k+1}, \hat{Y}_{1:k}\right) \mid \hat{Y}_{1:T}\right]$$

$$= E\left(\mu^*_{k|k}(X_{k+1}) \mid \hat{Y}_{1:T}\right) = E\left(\mu_{k|k} + \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} \left(X_{k+1} - \mu_{k+1|k}\right) \mid \hat{Y}_{1:T}\right)$$

$$= \mu_{k|k} + \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} \left(\mu_{k+1|T} - \mu_{k+1|k}\right)$$

$$\Sigma_{k|T} = Cov\left(X_k \mid \hat{Y}_{1:T}\right) = E\left[Cov\left(X_k \mid X_{k+1}, \hat{Y}_{1:T}\right) \mid \hat{Y}_{1:T}\right] + Cov\left[E\left(X_k \mid X_{k+1}, \hat{Y}_{1:T}\right) \mid \hat{Y}_{1:T}\right]$$

$$= E\left[Cov\left(X_k \mid X_{k+1}, \hat{Y}_{1:k}\right) \mid \hat{Y}_{1:T}\right] + Cov\left[E\left(X_k \mid X_{k+1}, \hat{Y}_{1:k}\right) \mid \hat{Y}_{1:T}\right]$$

$$= \Sigma^*_{k|k} + Cov\left[\mu^*_{k|k}(X_{k+1}) \mid \hat{Y}_{1:T}\right]$$

$$= \Sigma_{k|k} - \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} a_K \Sigma_{k|k} + \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} \Sigma_{k+1|T} \left(\Sigma_{k+1|k}\right)^{-1} a_k \Sigma_{k|k}$$

Note that the data is not needed for the RTS smoother.

Algorithm: RTS smoother

1. Run Kalman filter first
2. Starting from $\mu_{T|T}$ and $\Sigma_{T|T}$
3. Operate backwards in time

$$\mu_{k|T} = \mu_{k|k} + \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} \left(\mu_{k+1|T} - \mu_{k+1|k}\right)$$
$$\Sigma_{k|T} = \Sigma_{k|k} - \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} a_K \Sigma_{k|k} + \Sigma_{k|k} a_k^T \left(\Sigma_{k+1|k}\right)^{-1} \Sigma_{k+1|T} \left(\Sigma_{k+1|k}\right)^{-1} a_k \Sigma_{k|k}$$

Backward sampler:

In many times, we are interested in the maximum state response over the entire time interval [0,T]. However, from the results of Kalman filter and RTS smoother, we lose the correlation information between the states of different time. This correlation information is essential for understanding the maximum state response over time. We describe an algorithm that draws state time history samples from the posterior PDF $f(x_0,...,x_T \mid \hat{Y})$. This algorithm requires us to run Kalman filter first.

Note that

$$\begin{aligned}
&f\left(x_0,\cdots,x_T \mid \hat{Y}\right) \\
&= f\left(x_T \mid \hat{Y}\right) f\left(x_{T-1} \mid x_T, \hat{Y}\right) \cdots f\left(x_K \mid x_{K+1},\cdots,x_T,\hat{Y}\right) \ldots f\left(x_0 \mid x_1,\cdots,x_T,\hat{Y}\right) \\
&= f\left(x_T \mid \hat{Y}_{1:T}\right) f\left(x_{T-1} \mid x_T, \hat{Y}_{1:T-1}\right) \cdots f\left(x_k \mid x_{k+1}, \hat{Y}_{1:k}\right) \ldots f\left(x_1 \mid x_2, \hat{Y}_1\right) f\left(x_0 \mid x_1\right)
\end{aligned}$$

A strategy is to first sample $\hat{X}_T$ from $f(x_T \mid \hat{Y}) = N(\mu_{T|T}, \Sigma_{T|T})$, then sample $\hat{X}_{T-1}$ from $f\left(x_{T-1} \mid \hat{X}_T, \hat{Y}_{1:T-1}\right) = N\left(\mu^*_{T-1|T-1}(\hat{X}_T), \Sigma^*_{T-1|T-1}\right)$, then sample $\hat{X}_{T-2}$ from $N\left(\mu^*_{T-2|T-2}(\hat{X}_{T-1}), \Sigma^*_{T-2|T-2}\right)$ and so on to get a sample of the state time history. Do this many times independently to get independent state time history samples. Afterwards, we can use these samples to estimate the expected value of the maximum state response based on the Law of Large Number. Note that the data is not needed for the backward sampler.

Algorithm: Backward sampler

1. Do Kalman filter first
   $$\hat{X}_T \sim N\left(\mu_{T|T}, \Sigma_{T|T}\right)$$
2. Sample $\hat{X}_{T-1} \sim N\begin{pmatrix} \mu_{T-1|T-1} + \Sigma_{T-1|T-1} a_{T-1}^T \left(\Sigma_{T|T-1}\right)^{-1} \cdot \left(\hat{X}_T - x_{T|T-1}\right), \\ \Sigma_{T-1|T-1} + \Sigma_{T-1|T-1} a_{T-1}^T \left(\Sigma_{T|T-1}\right)^{-1} a_{T-1} \Sigma_{T-1|T-1} \end{pmatrix}$
   $$\vdots$$
3. Do (2) N times to get N i.i.d. samples from $f(x_0,...,x_T \mid \hat{Y})$

Analytical approximation on Bayesian state analysis of nonlinear dynamical systems

Introduction:

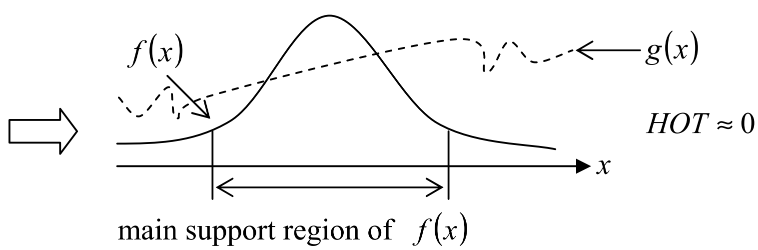
Many dynamical models are nonlinear or non-Gaussian. In this case, the above analysis breaks down. However, we can linearize and Gaussianize those models so Kalman filter, RTS smoother and backward sampler can still work approximately.

Extended Kalman filter: linearization on uncertainty propagation for $Y = g(X)$

Do Taylor expansion on $g(X)$ around $E(X)$, we get

$$Y = g(X) = g\left(E(X)\right) + \nabla_x g\left(E(X)\right)\cdot\left(X - E(X)\right) + HOT$$

Assuming $g(\cdot)$ is roughly linear in the main support region of the PDF $f(x)$,



we have

$$Y = g(X) \approx g\left(E(X)\right) + \nabla_x g\left(E(X)\right)\cdot\left(X - E(X)\right)$$

$E(Y) \approx g\left(E(X)\right)$ and $Cov(Y) \approx \nabla_x g\Big|_{x=E(X)} \cdot Cov(X) \cdot \nabla_x g\Big|_{x=E(X)}^{T}$

Note that the truncation error is of 2nd order, and the approximation will be poor if $g(\cdot)$ is highly nonlinear in the main support region of the PDF $f(x)$. This method of approximately propagating the first two moments is sometimes called the First-Order-Second-Moment (FOSM) method. Under this approximation, any nonlinear models can be linearized. The resulting Bayesian filtering algorithm is called extended Kalman filter. We'll skip the smoothing and backward sampling part.

For non-linear models, we can linearize the state and observation equations:

$$\begin{aligned}
X_{k+1} &= \phi_k\left(X_k, u_k, W_k\right) \\
&\approx \phi_k\left(\mu_{k|k}^{EKF}, u_k, 0\right) + \nabla_{X_k}\phi_k\left(\mu_{k|k}^{EKF}, u_k, 0\right)\cdot\left(X_k - \mu_{k|k}^{EKF}\right) + \nabla_{W_k}\phi_k\left(\mu_{k|k}^{EKF}, u_k, 0\right)\cdot W_k \\
&= \underbrace{\nabla_{X_k}\phi_k\left(\mu_{k|k}^{EKF}, u_k, 0\right)}_{a_k}\cdot X_k + \underbrace{\left[\phi_k\left(\mu_{k|k}^{EKF}, u_k, 0\right) - \nabla_{X_k}\phi_k\left(\mu_{k|k}^{EKF}, u_k, 0\right)\cdot \mu_{k|k}^{EKF}\right]}_{b_k u_k} + W_k^{EKF}
\end{aligned}$$

$$
\begin{aligned}
Y_k &= \varphi_k\left(X_k, u_k, V_k\right) \\
&\approx \varphi_k\left(\mu_{k|k-1}^{EKF}, u_k, 0\right) + \nabla_{X_k}\varphi_k\left(\mu_{k|k-1}^{EKF}, u_k, 0\right)\cdot\left(X_k - \mu_{k|k-1}^{EKF}\right) + \nabla_{V_k}\varphi_k\left(\mu_{k|k-1}^{EKF}, u_k, 0\right)\cdot V_k \\
&= \underbrace{\nabla_{X_k}\varphi_k\left(\mu_{k|k-1}^{EKF}, u_k, 0\right)}_{c_k}\cdot X_k + \underbrace{\left[\varphi_k\left(\mu_{k|k-1}^{EKF}, u_k, 0\right) - \nabla_{X_k}\varphi_k\left(\mu_{k|k-1}^{EKF}, u_k, 0\right)\cdot \mu_{k|k-1}^{EKF}\right]}_{d_k u_k} + V_k^{EKF}
\end{aligned}
$$

Then proceed with the Kalman filter algorithm. The resulting algorithm is called the extended Kalman filter.

Unscented Kalman filter: propagate moments by matching moments

Let $s = (X - EX)/\|X - EX\|$, consider the Taylor series expansion in the *s* direction:

$$
Y = g\left(X\right) = g\left(E\left(X\right)\right) + \sum_{i=1}^{\infty}\frac{1}{i!}\frac{\partial^i g(x)}{\partial s^i}\bigg|_{x=E(X)}\cdot\left\|X - E\left(X\right)\right\|^i
$$

where

$$
\begin{aligned}
&\frac{\partial^i g(x)}{\partial s^i}\bigg|_{x=E(X)} = \left(s\cdot\nabla_x\right)^i g(x)\Big|_{x=E(X)} \\
&= \left(s_1\frac{\partial}{\partial x_1} + \ldots + s_n\frac{\partial}{\partial x_n}\right)^i g(x)\bigg|_{x=E(X)} = \left(\sum_{j=1}^{n}\frac{X_j - E(X_j)}{\|X - EX\|}\cdot(\partial/\partial x_j)\right)^i g(x)\bigg|_{x=E(X)}
\end{aligned}
$$

So we get

$$
\begin{aligned}
&Y = g\left(X\right) \\
&= g\left(E\left(X\right)\right) + \sum_{i=1}^{\infty}\frac{1}{i!}\left(\sum_{j=1}^{n}\frac{X_j - E(X_j)}{\|X - EX\|}\cdot(\partial/\partial x_j)\right)^i\cdot g(E(X))\cdot\left\|X - E\left(X\right)\right\|^i \\
&= g\left(E\left(X\right)\right) + \sum_{i=1}^{\infty}\frac{1}{i!}\left(\sum_{j=1}^{n}\left[X_j - E(X_j)\right]\cdot(\partial/\partial x_j)\right)^i\cdot g(E(X))
\end{aligned}
$$

Therefore,

$$
E(Y) = g\left(E\left(X\right)\right) + \sum_{i=1}^{\infty}\frac{1}{i!}E\left[\left(\sum_{j=1}^{n}\left[X_j - E(X_j)\right]\cdot(\partial/\partial x_j)\right)^i\right]\cdot g(E(X))
$$

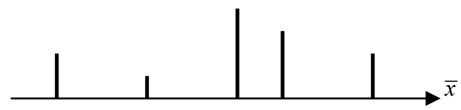
If $g(\cdot)$ is an $(2p-1)^{th}$ order polynomial and if we can find another uncertain variable $\overline{X}(\neq X)$ whose first $2p-1$ moments are identical to those of *X*. Define $\overline{Y} = g\left(\overline{X}\right)$. It is clear that $E\left(\overline{Y}\right) = E\left(Y\right)$. This is because when $g(\cdot)$ is an $(2p-1)^{th}$ order polynomial and the first $2p-1$ moment of $\overline{X}$ and X are identical,

$$E(Y)=g\left(E\left(X\right)\right)+\sum_{i=1}^{2p-1}\frac{1}{i!}E\left[\left(\sum_{j=1}^{n}\left[X_j-E(X_j)\right]\cdot(\partial/\partial x_j)\right)^i\right]\cdot g(E(X))$$

$$=g\left(E\left(\bar{X}\right)\right)+\sum_{i=1}^{2p-1}\frac{1}{i!}E\left[\left(\sum_{j=1}^{n}\left[\bar{X}_j-E(\bar{X}_j)\right]\cdot(\partial/\partial x_j)\right)^i\right]\cdot g(E(\bar{X}))=E(\bar{Y})$$

Note that given the $2p-1$ moments of *X*, $\bar{X}$ is not unique. A convenient choice is to take $\bar{X}$ with the following PDF:

$$f(\bar{x})=\sum_{i=1}^{p}w_i\delta(\bar{x}-\lambda_i),\ \ \sum_{i=1}^{p}w_i=1$$



where $\{(\lambda_i,w_i):i=1,...,p\}$ are the locations and weights of the delta functions. We can adjust the $2p-1$ free parameters to match the first $2p-1$ moments of *X*. Now it is clear that

$$E(Y)=E(\bar{Y})=\sum_{i=1}^{p}w_i g\left(\lambda_i\right)$$

is an exact solution!! We name this approach as the moment matching method (MM).

Consider the case that we would like to estimate the mean and variance of a scalar function h(X), i.e. we want to estimate E[h(X)] and Var[h(X)], where X is also a scalar. Also consider the usual case that we cannot analytically determine the gradient of h(X). For the linearization approach (FOSM), we usually need to evaluate h(.) function at three points to find E[h(X)] and Var[h(X)] since

$$E\left(h(X)\right)\approx h\left(E\left(X\right)\right)\ \text{ and }\ Var\left(h(X)\right)\approx\left(\frac{h(E(X)+\Delta x)-h(E(X)-\Delta x)}{2\Delta x}\right)^2\cdot Var\left(X\right)$$

The FOSM estimates are exact if the h(.) function is linear in the main support region of the PDF of X.

With the same computation cost, we can employ a three-point moment matching method, i.e. let $\bar{X}$ be the uncertain variable with the following PDF:

$$f\left(\bar{x}\right)=\sum_{i=1}^{3}w_i\delta\left(\bar{x}-\lambda_i\right),\ \ \sum_{i=1}^{3}w_i=1$$

As discussed in the above, we can match the first five moments of *X* using $\bar{X}$. So we have

$$E\left(h(X)\right) \approx \sum_{i=1}^{3} w_i h\left(\lambda_i\right) \text{ and } E\left(h(X)^2\right) \approx \sum_{i=1}^{3} w_i h\left(\lambda_i\right)^2$$

The MM estimates for $E\left(h(X)\right)$ is exact if the h(.) function is a fifth-order polynomial (or less) in the main support region of the PDF of X; the $E\left(h(X)^2\right)$ estimate is exact if the h(.) function is a quadratic polynomial (or less) in the main support region of the PDF of X.

You may wonder how to match moments by selecting appropriate $\left\{\left(\lambda_i, w_i\right): i=1,\ldots,p\right\}$? It is really simple: just solve the following equations:

$$\sum_{i=1}^{p} w_i = 1 \qquad \sum_{i=1}^{p} w_i \lambda_i = E\left(X\right) \qquad \cdots \qquad \sum_{i=1}^{p} w_i \lambda_i^{2p-1} = E\left(X^{2p-1}\right)$$

When X is some standard uncertain variable, we don't even need to solve them:

1. If X is uniform, $\left\{\left(\lambda_i, w_i\right): i=1,\ldots,p\right\}$ are related to the locations and weights of Gauss-Legendre quadrature. See the following link:
   http://mathworld.wolfram.com/Legendre-GaussQuadrature.html
2. If X is Gaussian, $\left\{\left(\lambda_i, w_i\right): i=1,\ldots,p\right\}$ are related to the locations and weights of Gauss-Hermite quadrature. See the following link:
   http://mathworld.wolfram.com/Hermite-GaussQuadrature.html
3. If X is exponential, $\left\{\left(\lambda_i, w_i\right): i=1,\ldots,p\right\}$ are related to the locations and weights of Gauss-Laguerre quadrature. See the following link:
   http://mathworld.wolfram.com/Laguerre-GaussQuadrature.html

If we use MM to propagate the first two moments $E(X_k \mid \hat{Y}_{1:k})$ and $Cov(X_k \mid \hat{Y}_{1:k})$ for approximate Bayesian filtering, the resulting algorithm is called the unscented Kalman filter [1].

Maximum entropy argument

We can think of the extended/unscented Kalman filters as first-two-moment Bayesian filters based on the maximum entropy principle. In the case that we only want to propagate the first two moments $E(X_k \mid \hat{Y}_{1:k})$ and $Cov(X_k \mid \hat{Y}_{1:k})$ for nonlinear dynamical systems, we can argue that the maximum entropy PDF constrained by the two moments $E(X_k \mid \hat{Y}_{1:k})$ and $Cov(X_k \mid \hat{Y}_{1:k})$ is Gaussian.

Reference:

[1] Julier, S.J, and Uhlmann, J.K. (1997) "A new extension of the Kalman filter to nonlinear systems." In *Proceedings of AeroSense*: *The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls*.