The Kalman filter and extensions

Filtering without simulation methods

Introduction

- Linear Gaussian State Space models (LGSSM) still useful
  - econometrics, engineering, neuroscience,...
- The filtering problem is tractable
  - Kalman filtering and other deterministic methods
  - very useful as **benchmark**
- More general question?
  - for which models are there tractable filtering solutions?
  - or approximations not based on simulation?

Filtering Recursions

- Prediction

$$
\begin{aligned}
p_\theta\left(x_n \mid y_{0:n-1}\right) &= \int f_\theta\left(x_n \mid x_{n-1}\right) p_\theta\left(x_{n-1} \mid y_{0:n-1}\right) dx_{n-1} \\
&= \int p_\theta\left(x_n, x_{n-1} \mid y_{0:n-1}\right) dx_{n-1}
\end{aligned}
$$

- Update

$$
\begin{aligned}
p_\theta\left(x_n \mid y_{0:n}\right) &= \frac{p_\theta\left(x_n \mid y_{0:n-1}\right) g_\theta\left(y_n \mid x_n\right)}{p_\theta\left(y_n \mid y_{0:n-1}\right)} \\
&= \frac{p_\theta\left(x_n \mid y_{0:n-1}\right) g_\theta\left(y_n \mid x_n\right)}{\int p_\theta\left(x_n \mid y_{0:n-1}\right) g_\theta\left(y_n \mid x_n\right) dx_n} \\
&= \frac{p_\theta(x_n, y_n \mid y_{0:n-1})}{p_\theta(y_n \mid y_{0:n-1})}
\end{aligned}
$$

Finite and discrete state spaces

- Let $\mathcal{X} = \{1, \ldots, d_x\}$ and $\mathcal{Y} = \{1, \ldots, d_y\}$
- Filtering computations can be done analytically
- Integrals are sums, densities (row) vectors and kernels matrices.

Linear Gaussian HMMs

- Multidimensional case:

$$\begin{aligned} X_n &= AX_{n-1} + BW_n, \\ Y_n &= CX_n + DV_n, \end{aligned}$$

  W_n, V_n iid zero mean Gaussian vectors.
- Some constraints need to be placed for $\theta = \{A, B, C, D\}$ to achieve controllability, observability etc.

The Filtering problem

Objective 1: Compute $\pi_n(\cdot) = \mathbb{P}[X_n \in \cdot | Y_{0:n}]$ on-line as we receive y_n.

- LGSSM: if $\pi_0 = \mathcal{N}(\mu_0, \Sigma_0)$ or δ_{x_0} then π_n is also Gaussian; let:

$$\pi_n = \mathcal{N}(\mu_{n|n}, \Sigma_{n|n})$$

Objective 2: Compute $p_\theta(y_{0:n})$ or $p_\theta(y_n | y_{0:n-1})$ on-line.

- Point above holds for also for $p_\theta(y_{0:n})$; recall:

$$p_\theta(y_{0:n}) = \prod_{k=0}^{n} p_\theta(y_k | y_{0:k-1})$$

The Kalman filter: the history

- Before 1960 filtering was a frequency domain business
  - Nobert Wiener, Andrey Kolmogorov
- Importance of time-domain analysis took of after Rudolf Kalman papers
  - Kalman filter: A new approach to linear filtering and prediction problems (1960)
  - Kalman-Bucy filter: Kalman & Bucy (1961) New Results in Linear Filtering and Prediction Theory (cont. time)
- Some historical precursors
  - Thorvald N. Thiele 1880
  - Ruslan Stratonovich 1958 (general continuous time work on filtering, Kalman-Bucy is a special case)
  - Stanley F. Schmidt 1950-1965 (navigation, square-root formulations Kalman-Schmidt filters)
  - Peter Swerling 1959 (optimal estimation of target tracks, fluctuating target models)

The Kalman filter in practice

- A nice historical account:
    - D. Crisan, The stochastic filtering problem: a brief historical account, 2014
- Importance of Kalman filter:
    - time-domain analysis took over then traditional frequency domain analysis
- Applications:
    - all purpose guidance & navigation systems
    - aerospace & military
        - tracking, SONAR and RADAR
        - ballistic systems
        - naval vessels
        - aircraft navigation, tracking and air traffic control
        - NASA Space Shuttle/ISS, NASA uses EKF still today!
    - robotics
    - economics (monetary policy models), finance & volatility models
    - ....

The model

- Linear Gaussian state space model

$$
\begin{aligned}
X_n &= AX_{n-1} + BW_n, \\
Y_n &= CX_n + DV_n,
\end{aligned}
$$

  $W_n, V_n \sim \mathcal{N}(0, I)$ i.i.d.

- Transition density and conditional likelihood (omitting θ-s)

$$f(x_n | x_{n-1}) = \mathcal{N}_{x_n}(Ax_{n-1}, BB^T)$$

$$g(y_n | x_n) = \mathcal{N}_{y_n}(Cx_n, DD^T)$$

Filtering Recursions

- We want to compute (again omitting θ-s)

$$
\begin{aligned}
p(x_n|y_{1:n-1}) &= \int f(x_n|x_{n-1})p(x_{n-1}|y_{1:n-1})dx_{n-1} \\
&= \mathcal{N}_{x_n}(\mu_{n|n-1}, \Sigma_{n|n-1}) \\
p(y_n|y_{1:n-1}) &= \int g_n(y_n|x_n)p(x_n|y_{1:n-1})dx_n \\
&= \mathcal{N}_{y_n}(m_n, S_n) \\
p(x_n|y_{1:n}) &= \frac{g_n(y_n|x_n)p(x_n|y_{0:n-1})}{\int g_n(y_n|x_n)p(x_n|y_{0:n-1})dx_n} \\
&= \mathcal{N}_{x_n}(\mu_{n|n}, \Sigma_{n|n})
\end{aligned}
$$

The Kalman filter

- The Kalman filter computes $\mu_{n|n}, \Sigma_{n|n}, m_n, S_n, \mu_{n|n-1}, \Sigma_{n|n-1}$ recursively

$$
\begin{aligned}
\mu_{n|n-1} &= A\mu_{n-1|n-1} \\
\Sigma_{n|n-1} &= A\Sigma_{n-1|n-1}A^T + BB^T \\
m_n &= C\mu_{n|n-1} \\
S_n &= C\Sigma_{n|n-1}C^T + DD^T \\
K_n &= \Sigma_{n|n-1}C^T S_n^{-1} \\
\mu_{n|n} &= \mu_{n|n-1} + K_n(Y_n - m_n) \\
\Sigma_{n|n} &= \Sigma_{n|n-1} - K_n C\Sigma_{n|n-1}
\end{aligned}
$$

The Kalman filter: sketch of probabilistic derivation

- Key elements
  1. Gaussian integral

$$\int_{-\infty}^{\infty} e^{-ax^2+bx+c}\,dx = \sqrt{\frac{\pi}{a}}\,e^{\frac{b^2}{4a}+c}$$

     or in d_x dimensions

$$\int e^{-\frac{1}{2}x^T Ax+x^T b}dx = \sqrt{\frac{(2\pi)^{d_x}}{\det A}}e^{\frac{1}{2}b^T A^{-1} b}.$$

  2. taking log and rearranging terms in exp and gather in quadratic form $-\frac{1}{2}x^T\Sigma_{n|n}^{-1}x + x^T\mu_{n|n}$

The Kalman filter: sketch of probabilistic derivation

- Gaussian integral is the key to computing prediction

$$p_\theta\left(x_n \middle| y_{0:n-1}\right) = \int f_\theta\left(x_n \middle| x_{n-1}\right) p_\theta\left(x_{n-1} \middle| y_{0:n-1}\right) dx_{n-1}$$

- The update

$$p_\theta\left(x_n \middle| y_{0:n}\right) \propto p_\theta\left(x_n \middle| y_{0:n-1}\right) g_\theta\left(y_n \middle| x_n\right)$$

  can be computed by 2.

- For recursive likelihood use both 1. and 2. (with 2. acting on Y_n)

$$p_\theta\left(y_n \middle| y_{0:n-1}\right) = \int p_\theta\left(x_n \middle| y_{0:n-1}\right) g_\theta\left(y_n \middle| x_n\right) dx_n$$

The Extended Kalman filter (EKF)

- Consider nonlinear case with zero mean additive noise

$$X_n = \psi_\theta\left(X_{n-1}\right) + BV_n,\ \ Y_n = \phi_\theta\left(X_n\right) + DW_n,$$

- Linearise ψ, ϕ around $\mu_{n-1|n-1}$ and $\mu_{n|n-1}$ resp. to get 1st order Taylor approx.

$$X_n = a_n + A_n X_{n-1} + BW_n,\ Y_n = c_n + C_n X_n + DV_n$$

with

$$A_n = \nabla_x \psi_\theta|_{\mu_{n-1|n-1}}, \quad C_n = \nabla_x \phi_\theta|_{\mu_{n|n-1}}$$

$$a_n = \psi_\theta\left(\mu_{n-1|n-1}\right) - A_n \mu_{n-1|n-1}, \quad c_n = \phi_\theta\left(\mu_{n|n-1}\right) - C_n \mu_{n|n-1}$$

- Run Kalman filter for linearised model

The Extended Kalman filter (EKF)

- General non-linear case

$$X_{n+1} = \psi_\theta\left(X_n, V_{n+1}\right), \;\; Y_n = \phi_\theta\left(X_n, W_n\right), \qquad (1)$$

- Replace (1) with an approximate linear Gaussian model

$$X_n = a_n + A_n X_{n-1} + B_n V_n, \;\; Y_n = c_n + C_n X_n + D_n W_n,$$

- 1st order Taylor approx.: linearisation of ψ, ϕ around $\mu_{n-1|n-1}$ and $\mu_{n|n-1}$ resp.

$$A_n = \nabla_x \psi_\theta|_{\left(\mu_{n-1|n-1},0\right)}, \quad C_n = \nabla_x \phi_\theta|_{\left(\mu_{n|n-1},0\right)}$$

and 0 for the noises (assuming they are zero mean)

$$B_n = \nabla_V \psi_\theta|_{\left(\mu_{n-1|n-1},0\right)}, \quad D_n = \nabla_W \phi_\theta|_{\left(\mu_{n1|n-1},0\right)}$$

so

$$a_n = \psi_\theta\left(\mu_{n-1|n-1}, 0\right) - A_n \mu_{n-1|n-1}, \quad c_n = \phi_\theta\left(\mu_{n|n-1}, 0\right) - C_n \mu_{n|n-1}$$

The extended Kalman filter

- EKF is Kalman filter for linearised model
- Computes $\mu_{n|n}, \Sigma_{n|n}, m_n, S_n, \mu_{n|n-1}, \Sigma_{n|n-1}$ recursively:

$$\begin{aligned}
\mu_{n|n-1} &= a_n + A_n \mu_{n-1|n-1} \\
\Sigma_{n|n-1} &= A_n \Sigma_{n-1|n-1} A_n^T + B_n B_n^T \\
m_n &= c_n + C_n \mu_{n|n-1} \\
S_n &= C_n \Sigma_{n|n-1} C_n^T + D_n D_n^T \\
K_n &= \Sigma_{n|n-1} C_n^T S_n^{-1} \\
\mu_{n|n} &= \mu_{n|n-1} + K_n (Y_n - m_n) \\
\Sigma_{n|n} &= \Sigma_{n|n-1} - K_n C_n \Sigma_{n|n-1}
\end{aligned}$$

The extended Kalman filter: discussion

- Pros:
  - The recursion is deterministic
    - update requires propagation of the the moments of π_n, i.e. deterministic quantities
  - This is easy to implement, cheap computationally
    - (derivations of derivatives are done off-line)
  - sometimes it works, for smooth and "close to linear models" it can give reasonable answers

The extended Kalman filter: discussion

- Cons:
  - approximation is hard to justify
    - higher order terms in Taylor series can be significant
    - so often is very inaccurate
  - approximation works for small class of models
    - makes sense only when ψ, ϕ are close to being linear.
    - hard non-linearities and discontinuities are an issue.
    - inapplicable when one cannot take gradients
  - we are still assuming V_n, W_n are Gaussian, if not still the approximation can be quite bad.

Towards non-linear filtering

- Basic approach of the Kalman filter:
  - propagates the first two moments of π_n recursively.
  - in the linear Gaussian case this is sufficient to capture π_n in the full filtering recursion.

- In general π_n might have a much higher number of non-zero moments.
  - so could we go ahead and try to update a fixed number of moments and/or sufficient statistics in closed form?

Towards non-linear filtering

- These filters are often called **finite dimensional**
  - makes sense when filtering distribution has a finite number of moments
  - can exploit Bayesian Conjugacy
  - note propagating infinite number of moments captures the full filter distribution.
- In principle is a good idea but very few examples in practice:
  - Vidoni, Exponential family state space models based on a conjugate latent process, JRSSB, 1999
  - Vidoni & Ferrante, Finite dimensional filters for nonlinear stochastic difference equations with multiplicative noises, SPA, 1998
  - Runggaldier & Fabio Spizzichino, Sufficient conditions for finite dimensionality of filters in discrete time: a Laplace transform-based approach, Bernoulli, 2001

Towards non-linear filtering

- Finite filters also would require finite computational cost
  - **on-line** filtering requires a **fixed number of computations (and memory) per time**
- Could possibly violate this, to see what the cost could be to update the filter:
  - look at super-linear computational costs and non-parametric filters
  - Papaspiliopoulos & Ruggiero. Optimal filtering and the dual process. Bernoulli, 2014.
- Possible for some models:
  - but the computational cost of the recursion would increase with time: exponentially or polynomially in some cases.
  - In most cases this is not practical as n can be quite large.

Towards non-linear filtering

- It is worth to mention that Initially KF was used for linear models and non-Gaussian i.i.d sequence for the noise:
  - it is an approximation, but can work if higher moments are very small (hard to know a priori)
- Still intuition so far is useful
  - We can construct approximations by **approximately** computing the first two moments recursively
  - could go beyond EKF

Extending the EKF

- Can we improve the performance of this basic idea?
  - using propagation of moments or other sufficient statistics
  - using numerical integration methods
  - without using any simulation based methods
- Some methods:
  - Gaussian Sum filter
  - Unscented Kalman filter
  - Quadrature/Cubature based Kalman recursions
  - .....

Gaussian Sum filter

- Alspach and Sorenson, Nonlinear Bayesian estimation using Gaussian sum approximations, IEEE Trans. Automat. Contr., 1972
- Key idea:
  - propagate a mixture of Gaussians, i.e. weights, means, covariances and number of components.
- Use a linearisation similar to EKF, but construct an approximation based on the following **mixture**

$$\pi_{n|n-1} = \sum_{i=1}^{q'_n} w'_{n,i} \mathcal{N}_{x_n}(\mu'_{n,i}, \Sigma'_{n,i})$$

$$\pi_n = \sum_{i=1}^{q_n} w_{n,i} \mathcal{N}_{x_n}(\mu_{n,i}, \Sigma_{n,i})$$

  where $\sum_{i=1}^{q'_n} w'_{n,i} = \sum_{i=1}^{q_n} w_{n,i} = 1$.
- Construct recursions for $w_n, q_n, \mu_{i,n}, \Sigma_{i,n}$ and $w'_n, q'_n, \mu'_{i,n}, \Sigma'_{i,n}$; see paper for details.

Gaussian Sum filter

- Pros
  - addresses some limitations related to EKF approximations such as dealing with multimodality
  - can be used for a bigger class of models
- Cons
  - is still based on linearisation
  - q_n increases linearly with time so increasing computational cost per time.

Using deterministic integration

- Chapman Kolmogorov is an integral

$$p_\theta\left(x_n \mid y_{0:n-1}\right) = \int f_\theta\left(x_n \mid x_{n-1}\right) p_\theta\left(x_{n-1} \mid y_{0:n-1}\right) dx_{n-1}$$

  so could be numerically computed.
- The update is

$$p_\theta\left(x_n \mid y_{0:n}\right) = \frac{p_\theta\left(x_n \mid y_{0:n-1}\right) g_\theta\left(y_n \mid x_n\right)}{p_\theta\left(y_n \mid y_{0:n-1}\right)}$$

  where the recursive likelihood is another integral

$$p_\theta\left(y_n \mid y_{0:n-1}\right) = \int p_\theta\left(x_n \mid y_{0:n-1}\right) g_\theta\left(y_n \mid x_n\right) dx_n$$

- Deterministic integration could be useful but the question is at what cost.

Using deterministic integration

- Some selected papers: the Unscented KF and the cubature KF
  - (UKF) Julier & Uhlmann (1995). A new extension of the Kalman filter to nonlinear systems. Int. Symp. Aerospace/Defense Sensing, Simul. and Controls
  - Arasaratnam & Haykin, Cubature kalman filters, IEEE Trans. Automat. Contr., 2009
- Key idea: complement the propagation of the moments with deterministic integration (quadrature/cubature methods).
- To a very basic level one can compute numerically integrals of the form:

$$\int_{\mathcal{X}} f(x)dx \approx \sum_{i=1}^{N} \alpha_k f(X_k)$$

  by good choice/placement of $\{\alpha_k, X_k\}_{k=1}^{N}$ based on structure and smoothness of f.

The Unscented tranform

- The Unscented Transform (UT) is a quadrature method:
  - it computes the mean and covariance of a random variable that undergoes a nonlinear transformation.
- Let X be a Gaussian random variable

$$X \sim \mathcal{N}(\mu, \Sigma)$$

  with $\Sigma = QQ^T$

- We want to approximate the <u>mean</u> and <u>covariance</u> of

$$g(X)$$

- We will form a set of $2L+1$ "sigma-points" to capture most of the mass in the support of the distr. of X.

The Unscented tranform

- Form a set of $2L + 1$ "sigma-points". Initialise

$$X^0 = \mu$$

  and for $i = 1, \ldots, L$ set

$$\begin{aligned} X^i &= \mu + \sqrt{n+\lambda}\,[Q]_i \\ X^{i+L} &= \mu - \sqrt{n+\lambda}\,[Q]_i \end{aligned}$$

  with $[Q]_i$ denoting the i-th column of the matrix.
- Then propagate the sigma points through $g(\cdot)$

$$Y^i = g(X^i), \quad i = 0, \ldots, 2L$$

The Unscented tranform

- Use quadrature estimates

$$
\begin{aligned}
\mathbb{E}\left[g(X)\right] = \mu_g \quad &\approx \quad \sum_{i=0}^{2L} W_m^i Y^i \\
\mathbb{C}ov\left[g(X)\right] = \Sigma_g \quad &\approx \quad \sum_{i=0}^{2L} W_c^i \left(Y^i - \mu_g\right)\left(Y^i - \mu_g\right)^T
\end{aligned}
$$

with

$$
\begin{aligned}
W_m^0 \quad &= \quad \frac{\lambda}{L+\lambda} \\
W_c^0 \quad &= \quad \frac{\lambda}{L+\lambda} + \kappa \\
W_m^i \quad &= \quad W_m^i = \frac{1}{2\left(L+\lambda\right)}, \quad i = 1, \ldots, L
\end{aligned}
$$

- λ, κ are tuning parameters

The Unscented KF

- For the model:

$$X_n = \psi_\theta\left(X_{n-1}, V_n\right) \;\; Y_n = \phi_\theta\left(X_n, W_n\right)$$

  Note that instead of X, Y, g one can use
  - (X_{n-1}, V_n), X_n, ψ_θ to get $\mu_{n|n-1}, \Sigma_{n|n-1}$
  - (X_n, W_n), Y_n, ϕ_θ to get m_n, Σ_n
- Update filter then as in KF

$$K_n = \Sigma_{n|n-1} C_n^T S_n^{-1}$$

$$\mu_{n|n} = \mu_{n|n-1} + K_n(Y_n - m_n)$$

$$\Sigma_{n|n} = \Sigma_{n|n-1} - K_n C_n \Sigma_{n|n-1}$$

Discussion on using numerical integration

- Pros
  - UKF/CKF can work better than EKF
  - UKF can work well when filter is close to a Gaussian distribution
  - CKF can improve on UKF
- Cons
  - again justification requires strong assumptions on model that are hard to quantify
    - how close are you to a Gaussian model or how close ϕ_θ, ψ_θ are to polynomials
  - numerical integration
    - is cumbersome in higher dimensions than $2-3$
    - can be difficult to tune
    - scales poorly with dimension of X_n, Y_n
  - naive implementation might result to increasing computational cost per time, e.g. Gaussian sum filter

Towards Monte Carlo filtering

- The issues with deterministic approximations often are very limiting in practice
  - e.g. consider models like stochastic volatility, Lotka-Volterra, epidemics, ...
- In the heart of filtering lies the problem of numerical integration
  - different direction is to use simulation and Monte Carlo
  - and take advantage of more computational power available.
- Particle filters are the state of the art!