Learning EnKF

Lecture 5

GOALS:

- Onlinear Probabilistic Estimation
- Mean Field Perfect Filter
- EnKF
- Learning Improved EnKF.
- Attention Map

WORK UNDER Data Assumption ⊛, Lecture 4

ONLINE PROBABILISTIC ESTIMATION //

ⓐ $v^\dagger_{j+1} = \Psi(v^\dagger_j) + \xi^\dagger_j$, $j \in \mathbb{Z}^+$; $v^\dagger_0 \sim N(m_0, C_0)$

ⓑ $y^\dagger_{j+1} = Hv^\dagger_{j+1} + \eta^\dagger_{j+1}$, $j \in \mathbb{N}$

$\xi^\dagger_j \sim N(0, \Sigma)$, $\eta^\dagger_{j+1} \sim N(0, \Gamma)$ i.i.d $\xi, \eta, v^\dagger_0 \perp\!\!\!\perp$

$\mu_j = \mathbb{P}(v^\dagger_j \mid Y^\dagger_j)$; $Y^\dagger_j = (y^\dagger_1, \ldots, y^\dagger_j)$
$\in \mathbb{R}^{kj}$

ⓐ: $\hat{\mu}_{j+1} = P\mu_j$ predict state

ⓑ: $\pi_{j+1} = Q\hat{\mu}_{j+1}$ predict obs

ⓒ: $\mu_{j+1} = B(\pi_{j+1}; y^\dagger_{j+1})$ condition

MEAN FIELD PERFECT FILTER

arxiv: 2209.11371 / Acta Numerica 2025.

ⓐ $\hat{v}_{j+1} = \Psi(v_j) + \xi_j$

ⓑ $\hat{y}_{j+1} = H\hat{v}_{j+1} + \eta_{j+1}$ $\pi_{j+1} = \mathrm{Law}(\hat{v}_{j+1}, \hat{y}_{j+1})$

ⓒ $v_{j+1} = T(\hat{v}_{j+1}, \hat{y}_{j+1}; \pi_{j+1}, y^{\dagger}_{j+1})$

Then $\mathrm{Law}(v_j) = \mu_j$, $\mathrm{Law}(\hat{v}_{j+1}) = \hat{\mu}_{j+1}$.

ENSEMBLE KALMAN FILTER (EnKF)

Mean field EnKF:

- Approximate $T(\cdot, \cdot; \pi_{j+1}, y^{\dagger}_{j+1})$ in affine class, $T_{affine}(\cdot, \cdot; \pi_{j+1}, y^{\dagger}_{j+1})$

- Make T_{affine} exact in Gaussian setting.

ⓐ $\hat{v}_{j+1} = \Psi(v_j) + \xi_j$

ⓑ $\hat{y}_{j+1} = H\hat{v}_{j+1} + \eta_{j+1}$ $\hat{\mu}_{j+1} = \mathrm{Law}(\hat{v}_{j+1})$

ⓒ $v_{j+1} = \hat{v}_{j+1} + K_{j+1}(\hat{\mu}_{j+1})(y^{\dagger}_{j+1} - \hat{y}_{j+1})$

$$K_{j+1}(\hat{\mu}_{j+1}) = \hat{C}_{j+1} H^T (H \hat{C}_{j+1} H^T + \Gamma)$$ 3

$\hat{C}_{j+1} =$ covariance of $\hat{v}_{j+1}$.

Particle Approximation ($n = 1, \dots, N$):

(a) $\hat{v}^{(n)}_{j+1} = \Psi(v^{(n)}_j) + \xi^{(n)}_j$ $\qquad \hat{\mu}^N_{j+1} = \frac{1}{N} \sum_{j=1}^{N} \delta_{\hat{v}^{(n)}_{j+1}}$

(b) $\hat{y}^{(n)}_{j+1} = H \hat{v}^{(n)}_{j+1} + \eta^{(n)}_{j+1}$

(c) $v^{(n)}_{j+1} = \hat{v}^{(n)}_{j+1} + K_{j+1}(\hat{\mu}^N_{j+1})(y^+_{j+1} - \hat{y}^{(n)}_{j+1})$

$$K_{j+1}(\hat{\mu}^N_{j+1}) = \hat{C}^N_{j+1} H^T (H \hat{C}^N_{j+1} H^T + \Gamma)$$

$\hat{C}^N_{j+1} =$ empiral covariance of $\{\hat{v}^{(n)}_{j+1}\}_{n=1}^{N}$

LEARNING IMPROVED EnKF //

The EnKF is accurate when near-Gaussian:

arXiv: 2212.13239 (SINUM 2024)

arXiv: 2409.09800 (SINUM to Appear)

Use machine learning to improve it beyond this setting.

arXiv: 2504.17836

(a) $\hat{v}_{j+1}^{(n)} = \Psi(v_j^{(n)}) + \xi_j^{(n)}$ $\qquad \hat{\mu}_{j+1}^N = \frac{1}{N}\sum_{n=1}^{N} \delta_{\hat{v}_{j+1}^{(n)}}$

(b) $\hat{y}_{j+1}^{(n)} = H\hat{v}_{j+1}^{(n)} + \eta_{j+1}^{(n)}$

(c) $v_{j+1}^{(n)} = T(\hat{v}_{j+1}^{(n)}, \hat{y}_{j+1}^{(n)}; \hat{\mu}_{j+1}^N, y_{j+1}^{+}; \theta)$

$$J(\theta) = \frac{1}{J}\sum_{j=1}^{J} \| \bar{v}_j(\theta) - v_j^{+} \|^2$$

$$\bar{v}_j(\theta) = \frac{1}{N}\sum_{j=1}^{N} v_j^{(n)}(\theta) = \mathbb{E}^{\mu_j^N} v. \qquad \theta^* = \operatorname{arginf} J.$$

Key challenges for architecture:

- Neural networks that take probability measures as input, including empirical measures
- want to use same parameter θ for all N.
- In practice fine-tune localization parameter for different N.

ATTENTION MAP //

Let $\mu^N = \frac{1}{N}\sum_{n=1}^{N} \delta_{u^{(n)}}$

5

$u_{1:N} = (u^{(1)}, \dots, u^{(N)}) \in F^N = \{f : \{1, \dots, N\} \to \mathbb{R}^d\}$

$ATT : F^N \to F^N$ defined by

$$ATT(u_{1:N})_k = h_k + \frac{1}{Z_k} \sum_{\ell=1}^{N} \exp(\langle Qu_k, Ku_\ell \rangle) V_\ell$$

$$Z_k = \sum_{\ell=1}^{N} \exp(\langle Qu_k, Ku_\ell \rangle)$$

Main component of set transformer.