# Serial EnKF

Consider implementing a sequential algorithm for EnKF to avoid large matrix inversion in $K = P^b H^T ( H P^b H^T + R)^{-1}$.

In EnKF, we use $P^b = \mathbb{X}^b \mathbb{X}^{bT}$

Note that $H P^b H^T = (H\mathbb{X}^b)(H\mathbb{X}^b)^T$, and $P^b H^T = \mathbb{X}^b (H\mathbb{X}^b)^T$

It is $H\mathbb{X}^b$ that is actually needed in calculation of $K$.

$$H\mathbb{X}^b = \begin{pmatrix} - & h_1 & - \\ - & h_2 & - \\ & \vdots & \\ - & h_p & - \end{pmatrix} \begin{pmatrix} | & | & & | \\ x_1' & x_2' & \cdots & x_N' \\ | & | & & | \end{pmatrix}^b$$

$h_j x_k^{b\prime}$ is the perturbation ($\overset{k\text{th}}{\text{member}} - \text{mean}$) of observed value associated with $j$th observation

Recall that Linearized observation operator $h_j$ is from assumption:

$$h_j \Delta x = \left.\frac{\partial h_j}{\partial x}\right|_x (x + \Delta x - x) \approx h_j(x+\Delta x) - h_j(x)$$

Now we can use the original difference to replace $H\mathbb{X}^b$ so that linearized observation operators are no longer needed.

$$(H\mathbb{X}^b)_{jk} = \underbrace{h_j x_k^{b\prime}}_{\substack{\text{linearized}\\ \text{observation}\\ \text{operator}}} = \left.\frac{\partial h_j}{\partial x}\right|_{\bar{x}^b} (x_k^b - \bar{x}^b) \approx \underbrace{h_j(x_k^b) - h_j(\bar{x}^b)}_{\substack{\text{nonlinear}\\ \text{observation}\\ \text{function}}} \quad (1)$$

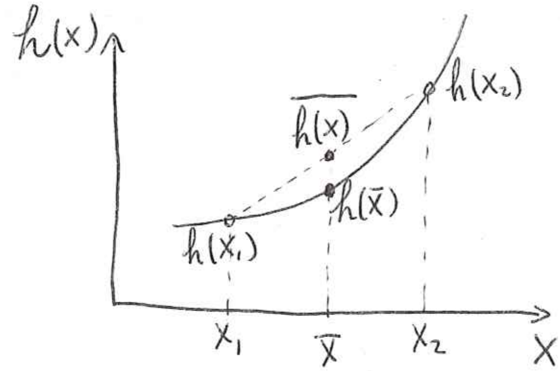
Alternatively, one can use

$$\overline{h_j(x^b)} \equiv \frac{1}{N}\sum_{k=1}^{N} h_j(x_k^b)$$

to replace $h_j(\bar{x}^b)$.

$$h_j x_k^{b\prime} = h_j(x_k^b) - \overline{h_j(x^b)} \qquad (2)$$

If $h()$ is a nonlinear function, $h_j(\bar{x}^b) \neq \overline{h_j(x^b)}$ :




$\bar{x} = \frac{1}{2}(x_1 + x_2)$.

$\overline{h(x)} = \frac{1}{2}\left[ h(x_1) + h(x_2) \right]$

→ In practice, use (1) or (2) ?

$\bar{x}^b$ ensemble mean is not necessarily a natural solution of the dynamic model, therefore sometimes $\overline{h_j(x^b)}$ is a better solution than $h_j(\bar{x}^b)$

- For example, let $x_k^b$ be ensemble members simulating a convective cloud, the location of these clouds are different in each member. $\bar{x}^b$ will have a smoothed cloud field with wide-spread thin cloud instead of convective cloud. Let $h(x)$ be the radiative transfer model (RTM) that gives radiance from cloud top, which is strongly nonlinear (takes only a very thin cloud to reduce the brightness temperature a lot). ⟹ $h(\bar{x}^b)$ will be cold biased.

If $R$ is diagonal (observation errors are uncorrelated) a sequential algorithm can be used to update $\bar{x}$ and $x'$

$\Rightarrow$ Serial EnKF.

Initial input: $x_{(1),k} = x^b_k$ for $k=1,2,\cdots,N$

Assimilate observations one at a time:

for $j=1,2,\cdots,p$

$\bar{x}_{(j)} = \frac{1}{N}\sum_{k=1}^{N} x_{(j),k}$ prior mean

for $k=1,2,\cdots,N$

$x'_{(j),k} = x_{(j),k} - \bar{x}_{(j)}$ prior perturbations

$y_{(j),k} = h_j(x_{(j),k})$ observation priors

$\bar{y}_{(j)} = \frac{1}{N}\sum_{k=1}^{N} y_{(j),k}$ observation prior mean

for $k=1,2,\cdots,N$ : $y'_{(j),k} = y_{(j),k} - \bar{y}_{(j)}$ " " perturbations

scalars

$var(y^o_j) = R_{jj}$ observation error variance

$var(y_{(j)}) = \frac{1}{N-1}\sum_{k=1}^{N}\left(y'_{(j),k}\right)^2$ observation prior variance

$$cov(x_{(j)}, y_{(j)}) = \frac{1}{N-1}\sum_{k=1}^{N}\Big(\underbrace{x'_{(j),k}}_{\text{vector}} \cdot \underbrace{y'_{(j),k}}_{\text{scalar}}\Big)$$

$$K_j = \frac{cov(x_{(j)}, y_{(j)})}{var(y^o_j) + var(y_{(j)})}$$

for $k=1,2,\cdots,N$

$$x_{(j+1),k} = x_{(j),k} + K_j\Big(\underbrace{y^o_{j,k}}_{\substack{\text{perturbed}\\ \text{observation}}} - y_{(j),k}\Big)$$

perturbed observation $\sim N(h_j(x^{tr}), R_{jj})$

output: $x^a_k = x_{(p+1),k}$

Note on Generating random perturbations that fit a Gaussian distribution:

$$\vec{\varepsilon} \sim N(0, R)$$

1. Draw random ensemble $\vec{\eta}_k$, $k = 1, 2, \cdots, N$
2. Recenter ensemble mean (zero in this case)

$$\vec{\delta}_k = \vec{\eta}_k - \frac{1}{N}\sum_{k=1}^{N} \vec{\eta}_k \quad \text{for } k = 1, 2, \cdots, N$$

3. Rescale ensemble covariance:

$$R' = \frac{1}{N-1}\sum_{k=1}^{N} \vec{\delta}_k \vec{\delta}_k^T$$

Take eigenvalue decomposition of $R' = USU^T$

$$\vec{\varepsilon}_k = R^{\frac{1}{2}} U S^{-\frac{1}{2}} U^T \vec{\delta}_k$$

---

Note on update Equation $\delta X = K(y^o - h(x^b))$

Anderson (2003) derived the impact of a single observation $y$ on a single state variable $X$ → this is sufficient to describe all commonly used ensemble filter algorithms without loss of generality:

The update happens in two steps:

① Analysis in observation space: $\delta y = \dfrac{var(y^b)}{var(y^b) + var(y^o)} (y^o - y^b)$

② regression of increment to state space: $\delta X = \dfrac{cov(X^b, y^b)}{var(y^b)} \delta y$