Adaptive ensemble Kalman filtering of nonlinear systems

Tyrus Berry

September 29, 2016
Background:

- PhD Mathematics at GMU, Advisor: Tim Sauer
- Postdoc at PSU with John Harlim
- NSF Big Data Postdoc at GMU (current)

Research Interests:

- Geometry of data and nonparametric statistics
- Data-driven and model-free forecasting
- Filtering/forecasting with model error

This is also joint work with Franz Hamilton (postdoc at NC State)
What is the filtering problem?

Consider a discrete time dynamical system:

\[ x_k = f_k(x_{k-1}, \omega_k) \]
\[ y_k = h_k(x_k, \nu_k) \]

Where \( x_k \) is the state variable, \( \omega_k \) is stochastic forcing, and the maps \( f_k \) define the dynamics.

The maps \( h_k \) are called the observation functions, \( \nu_k \) is observation noise, and \( y_k \) is a noisy observation.
What is the filtering problem?

Consider a discrete time dynamical system:

\[ x_k = f_k(x_{k-1}, \omega_k) \]
\[ y_k = h_k(x_k, \nu_k) \]

Given the observations \( y_1, \ldots, y_k \) we define three problems:

- **Filtering**: Estimate the current state \( p(x_k \mid y_1, \ldots, y_k) \)
- **Forecasting**: Estimate a future state \( p(x_{k+\ell} \mid y_1, \ldots, y_k) \)
- **Smoothing**: Estimate a past state \( p(x_{k-\ell} \mid y_1, \ldots, y_k) \)
What is the filtering problem?

\[ \frac{dx^i}{dt} = -x^{i-2}x^{i-1} + x^{i-1}x^{i+1} - x^i + F \]
Two Step Filtering to Find $p(x_k \mid y_1, ..., y_k)$

- Assume we have $p(x_{k-1} \mid y_1, ..., y_{k-1})$

- **Forecast Step:** Find $p(x_k \mid y_1, ..., y_{k-1})$

- **Assimilation Step:** Perform a Bayesian update,

$$p(x_k \mid y_1, ..., y_k) \propto p(x_k \mid y_1, ..., y_{k-1}) p(y_k \mid x_k, y_1, ..., y_{k-1})$$

**Posterior** $\propto$ **Prior** $\times$ **Likelihood**
Assume linear dynamics/obs and additive Gaussian noise

\[ x_k = F_{k-1}x_{k-1} + \omega_k \quad \omega_k \sim \mathcal{N}(0, Q) \]
\[ y_k = H_k x_k + \nu_k \quad \nu_k \sim \mathcal{N}(0, R) \]

For linear systems, easy observability condition:

\[
\tilde{H}_k^\ell = \begin{pmatrix}
H_k \\
H_{k+1}F_k \\
\vdots \\
H_{k+\ell+1}F_{k+\ell} \cdots F_k
\end{pmatrix}
\]

Must be full rank for some \( \ell \)
Assume linear dynamics/obs and additive Gaussian noise

\[
\begin{align*}
    x_k &= F_{k-1}x_{k-1} + \omega_k & \omega_k &\sim \mathcal{N}(0, Q) \\
    y_k &= H_k x_k + \nu_k & \nu_k &\sim \mathcal{N}(0, R)
\end{align*}
\]

Assume current estimate is Gaussian:

\[
p(x_{k-1} \mid y_1, \ldots, y_{k-1}) = \mathcal{N}(\hat{x}_{k-1}^a, P_{k-1}^a)
\]
Kalman Filter: Forecast Step

- At time $k - 1$ we have mean $\hat{x}_{k-1}^a$ and covariance $P_{k-1}^a$
- Linear combinations of Gaussians are still Gaussian so:
  - $p(F_{k-1}x_{k-1} \mid y_1, \ldots, y_{k-1}) = \mathcal{N}(F_{k-1}\hat{x}_{k-1}^a, F_{k-1}P_{k-1}F_{k-1}^\top)$
  - $p(x_k \mid y_1, \ldots, y_{k-1}) = \mathcal{N}(F_{k-1}\hat{x}_{k-1}^a, F_{k-1}P_{k-1}F_{k-1}^\top + Q)$
- Define the **Forecast mean**: $\hat{x}_k^f \equiv F_{k-1}\hat{x}_{k-1}^a$
- Define the **Forecast covariance**: $P_k^f \equiv F_{k-1}P_{k-1}F_{k-1}^\top + Q$
Kalman Filter: Defining the Likelihood function

- Recall that $y_k = H_k x_k + \nu_k$ where $\nu_k \sim \mathcal{N}(0, R)$ is Gaussian
- The forecast distribution: $p(x_k | y_1, \ldots, y_{k-1}) = \mathcal{N}(\hat{x}_k^f, P_k^f)$

**Likelihood:** $p(y_k | x_k, y_1, \ldots, y_{k-1}) = \mathcal{N}(H_k \hat{x}_k^f, H_k P_k^f H_k^\top + R)$

- Define the **Observation mean:** $y_k^f = H_k \hat{x}_k^f$
- Define the **Observation covariance:** $P_k^y = H_k P_k^f H_k^\top + R$
Kalman Filter: Assimilation Step

- Gaussian prior $\times$ Gaussian likelihood $\Rightarrow$ Gaussian posterior

$$p(y|x)p(x) \propto \exp \left\{ -\frac{1}{2} (y - Hx)^\top (P^y)^{-1} (y - Hx) 
- \frac{1}{2} (x - \hat{x}^f)^\top (P^f)^{-1} (x - \hat{x}^f) \right\}$$

$$\propto \exp \left\{ -\frac{1}{2} x^\top \left( (P^y)^{-1} + H(P^f)^{-1} H^\top \right) x 
+ x^\top \left( H^\top (P^y)^{-1} y - (P^f)^{-1} \hat{x}^f \right) \right\}$$

- Posterior Covariance: $P^a = ((P^f)^{-1} + H^\top (P^y)^{-1} H)^{-1}$
- Posterior Mean: $x^a = P^a \left( H^\top (P^y)^{-1} y - (P^f)^{-1} \hat{x}^f \right)$
Kalman Filter: Assimilation Step

- **Kalman Equations**: (after some linear algebra...)
  - Kalman Gain: $K_k = P_k^f H_k^T (P_k^y)^{-1}$
  - Innovation: $\epsilon_k = y_k - y_k^f$
  - Posterior Mean: $\hat{x}_k^a = \hat{x}_k^f + K_k \epsilon_k$
  - Posterior Covariance: $P_k^a = (I - K_k H_k) P_k^f$

- $\hat{x}_k^a$ is the least squares/minimum variance estimator of $x_k$
Kalman Filter Summary

\[ x_f^k = F_{k-1} x_{a}^{k-1} \]

\[ P_f^k = F_{k-1} P_{a}^{k-1} F_{k-1}^T + Q_{k-1} \]

\[ P_y^k = H_k P_f^k H_k^T + R_{k-1} \]

\[ K_k = P_k^f H_k^T (P_y^k)^{-1} \]

\[ P_a^k = (I - K_k H_k) P_f^k \]

\[ \epsilon_k = y_k - y_f^k = y_k - H_k x_f^k \]

\[ x_a^k = x_f^k + K_k \epsilon_k \]
What about nonlinear systems?

- Consider a system of the form:

  \[ x_{k+1} = f(x_k) + \omega_{k+1} \quad \omega_{k+1} \sim \mathcal{N}(0, Q) \]
  \[ y_{k+1} = h(x_{k+1}) + \nu_{k+1} \quad \nu_{k+1} \sim \mathcal{N}(0, R) \]

- More complicated observability condition (Lie derivatives)

- **Extended Kalman Filter (EKF):**
  - Linearize \( F_k = Df(\hat{x}_k^a) \) and \( H_k = Dh(\hat{x}_k^f) \)
  - Problem: State estimate \( \hat{x}_k^a \) may not be well localized

- Solution: Ensemble Kalman Filter (EnKF)
Generate an ensemble with the current statistics (use matrix square root):

\begin{align*}
    x_t^i &= \text{“sigma points” on semimajor axes} \\
    x_t^f &= \frac{1}{2n} \sum F(x_t^i) \\
    P_{xx}^f &= \frac{1}{2n-1} \sum (F(x_t^i) - x_t^f)(F(x_t^i) - x_t^f)^T + Q
\end{align*}
Ensemble Kalman Filter (EnKF)

Calculate \( y_t^i = H(F(x_t^i)) \). Set \( y_t^f = \frac{1}{2n} \sum_i y_t^i \).

\[
\begin{align*}
P_{yy} &= (2n - 1)^{-1} \sum_i (y_t^i - y_t^f)(y_t^i - y_t^f)^T + R \\
P_{xy} &= (2n - 1)^{-1} \sum_i (F(x_t^i) - x_t^f)(y_t^i - y_t^f)^T \\
K &= P_{xy}P_{yy}^{-1} \quad \text{and} \quad P_{xx}^a = P_{xx}^f - KP_{yy}K^T \\
x_{t+1}^a &= x_t^f + K(y_t - y_t^f)
\end{align*}
\]
Parameter Estimation

- When the model has parameters $p$,

$$x_{k+1} = f(x_k, p) + \omega_{k+1}$$

- Can augment the state $\tilde{x}_k = [x_k, p_k]$

- Introduce trivial dynamics for $p$

$$x_{k+1} = f(x_k, p_k) + \omega_{k+1}$$

$$p_{k+1} = p_k + \omega^p_{k+1}$$

- Need to tune the covariance of $\omega^p_{k+1}$
Example of Parameter Estimation

Consider the Hodgkin-Huxley neuron model, expanded to a network of $n$ equations

\[ \dot{V}_i = -g_{Na} m^3 h(V_i - E_{Na}) - g_K n^4 (V_i - E_K) - g_L (V_i - E_L) + I + \sum_{j \neq i} \Gamma_{HH}(V_j) V_j \]

\[ \dot{m}_i = a_m(V_i)(1 - m_i) - b_m(V_i)m_i \]

\[ \dot{h}_i = a_h(V_i)(1 - h_i) - b_h(V_i)h_i \]

\[ \dot{n}_i = a_n(V_i)(1 - n_i) - b_n(V_i)n_i \]

\[ \Gamma_{HH}(V_j) = \frac{\beta_{ij}}{1 + e^{-10(V_j+40)}} \]

Only observe the voltages $V_i$, recover the hidden variables and the connection parameters $\beta$. 
Example of Parameter Estimation

Can even turn connections on and off (grey dashes)
Variance estimate $\Rightarrow$ statistical test (black dashes)

![Parameter Estimation Graph](image)
Nonlinear Kalman-type Filter: Influence of $Q$ and $R$

- Simple example with full observation and diagonal noise covariances
- Red indicates RMSE of unfiltered observations
- Black is RMSE of ‘optimal’ filter (true covariances known)
Standard Kalman Update:

\[ P_{k}^{f} = F_{k-1}P_{k-1}^{a}F_{k-1}^T + Q_{k-1} \]
\[ P_{k}^{y} = H_{k}P_{k}^{f}H_{k}^T + R_{k-1} \]
\[ K_{k} = P_{k}^{f}H_{k}^T(P_{k}^{y})^{-1} \]
\[ P_{k}^{a} = (I - K_{k}H_{k})P_{k}^{f} \]
\[ \epsilon_{k} = y_{k} - y_{k}^{f} = y_{k} - H_{k}x_{k}^{f} \]
\[ x_{k}^{a} = x_{k}^{f} + K_{k}\epsilon_{k} \]
Adaptive Filter: Estimating $Q$ and $R$

- Innovations contain information about $Q$ and $R$

$$\epsilon_k = y_k - y_k^f$$

$$= h(x_k) + \nu_k - h(x_k^f)$$

$$= h(f(x_{k-1}) + \omega_k) - h(f(x_{a,k-1})) + \nu_k$$

$$\approx H_k F_{k-1} (x_{k-1} - x_{a,k-1}) + H_k \omega_k + \nu_k$$

- IDEA: Use innovations to produce samples of $Q$ and $R$:

$$\mathbb{E}[\epsilon_k \epsilon_k^T] \approx H P^f H^T + R$$

$$\mathbb{E}[\epsilon_{k+1} \epsilon_k^T] \approx H F P^e H^T - H F K \mathbb{E}[\epsilon_k \epsilon_k^T]$$

$$P^e \approx F P^a F^T + Q$$

- In the linear case this is rigorous and was first solved by Mehra in 1970.
Adaptive Filter: Estimating $Q$ and $R$

- To find $Q$ and $R$ we estimate $H_k$ and $F_{k-1}$ from the ensemble and invert the equations:

$$
\mathbb{E}[\epsilon_k\epsilon_k^T] \approx HP^f H^T + R
$$
$$
\mathbb{E}[\epsilon_{k+1}\epsilon_k^T] \approx HFP^e H^T - HFK\mathbb{E}[\epsilon_k\epsilon_k^T]
$$

- This gives the following empirical estimates of $Q_k$ and $R_k$:

$$
P_k^e = (H_{k+1}F_k)^{-1}(\epsilon_{k+1}\epsilon_k^T + H_{k+1}F_kK_k\epsilon_k\epsilon_k^T)H_k^{-T}
$$
$$
Q_k^e = P_k^e - F_{k-1}P_{k-1}F_{k-1}^T
$$
$$
R_k^e = \epsilon_k\epsilon_k^T - H_kP_k^f H_k^T
$$

- Note: $P_k^e$ is an empirical estimate of the background covariance
An Adaptive Kalman-Type Filter for Nonlinear Problems

We combine the estimates of $Q$ and $R$ with a moving average

Original Kalman Eqs.

$$
P_k^f = F_{k-1}P_{k-1}^a F_{k-1}^T + Q_{k-1}
$$

$$
P_k^y = H_k P_k^a H_k^T + R_{k-1}
$$

$$
K_k = P_k^f H_k^T (P_k^y)^{-1}
$$

$$
P_k^a = (I - K_k H_k) P_k^f
$$

$$
\epsilon_k = y_k - y_k^f
$$

$$
x_k^a = x_k^f + K_k \epsilon_k
$$

Our Additional Update

$$
P_{k-1}^e = F_{k-1}^{-1} H_{k-1}^{-1} \epsilon_k \epsilon_k^T H_{k-1}^{-T}
$$

$$
+ K_{k-1} \epsilon_{k-1} \epsilon_{k-1}^T H_{k-1}^{-T}
$$

$$
Q_{k-1}^e = P_{k-1}^e - F_{k-2} P_{k-2}^a F_{k-2}^T
$$

$$
R_{k-1}^e = \epsilon_{k-1} \epsilon_{k-1}^T H_{k-1} - H_{k-1} P_{k-1}^f H_{k-1}^T
$$

$$
Q_k = Q_{k-1} + (Q_{k-1}^e - Q_{k-1}) / \tau
$$

$$
R_k = R_{k-1} + (R_{k-1}^e - R_{k-1}) / \tau
$$
How does this compare to inflation?

- We extend Kalman's equations to estimate $Q$ and $R$
- Estimates converge for linear models with Gaussian noise
- When applied to nonlinear, non-Gaussian problems
  - We interpret $Q$ as an additive inflation
  - $Q$ can have complex structure, possibly more effective than multiplicative inflation?
  - Downside: many more parameters than multiplicative inflation?
- Somewhat less ad hoc than other inflation techniques?
Observability and Parameterization of Q

Recall:

\[ P_{k-1}^e = F_{k-1}^{-1} H_k^{-1} \epsilon_k \epsilon_{k-1}^T H_{k-1}^{-T} + K_{k-1} \epsilon_k \epsilon_{k-1}^T H_{k-1}^{-T} \]

\[ Q_{k-1}^e = P_{k-1}^e - F_{k-2} P_{k-2}^a F_{k-2}^T \]

Together these equations imply that:

\[ H_k F_{k-1} Q_{k-1}^e H_{k-1}^T = \epsilon_k \epsilon_{k-1}^T + H_k F_{k-1} K_{k-1} \epsilon_k \epsilon_{k-1}^T H_{k-1}^T - H_k F_{k-1} P_{k-1}^a F_{k-1}^T H_{k-1}^T \]

Set \( C_k \) equal to the right hand side (we simply compute \( C_k \)).

Parameterize \( Q_k^e = \sum_{i=1}^{s} q_i \hat{Q}_i \) where \( q_i \) are scalar parameters and \( \hat{Q}_i \) are ‘shape’ matrices.
We now need to solve:

\[ C_k = \sum_{i=1}^{s} q_i H_k F_{k-1} \hat{Q}_i H_{k-1}^T \]

We vectorize the equation as

\[ \text{vec}(C_k) = \sum_{i=1}^{s} q_i \text{vec}(H_k F_{k-1} \hat{Q}_i H_{k-1}^T) = A_k [q_1, ..., q_s]^T \]

where \( A_k \) is an \( m^2 \)-by-\( l \) matrix where the \( i \)-th row is given by \( \text{vec}(H_k F_{k-1} \hat{Q}_i H_{k-1}^T) \).

We can the solve for the parameters \([q_1, ..., q_s]^T\) by least squares.
We will apply the adaptive EnKF to the 40-dimensional Lorenz96 model integrated over a time step $\Delta t = 0.05$

$$\frac{dx^i}{dt} = -x^{i-2}x^{i-1} + x^{i-1}x^{i+1} - x^i + F$$

We augment the model with Gaussian white noise

$$x_k = f(x_{k-1}) + \omega_k \quad \omega_k = \mathcal{N}(0, Q)$$
$$y_k = h(x_k) + \nu_k \quad \nu_k = \mathcal{N}(0, R)$$

We will consider full and sparse observations

The Adaptive EnKF uses $F = 8$

We will consider model error where the true $F^i = \mathcal{N}(8, 16)$
Recovering $Q$ and $R$, Full Observability

RMSE shown for the initial guess covariances (red) the true $Q$ and $R$ (black) and the adaptive filter (blue)
Recovering $Q$ and $R$, Sparse Observability

Observing 10 sites results in divergence with the true $Q$ and $R$

<table>
<thead>
<tr>
<th>True Covariance</th>
<th>Initial Guess</th>
<th>Final Estimate</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$R$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

RMSE shown for the initial guess covariances (red) the true $Q$ and $R$ (black) and the adaptive filter (blue)
The adaptive filter compensates for errors in the forcing $F_i$

<table>
<thead>
<tr>
<th>True Covariance</th>
<th>Initial Guess</th>
<th>Final Estimate</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

RMSE shown for the initial guess covariances (red) an Oracle EnKF (black) and the adaptive filter (blue)
Integration with the LETKF

Simply find a local $Q$ and $R$ for each region

<table>
<thead>
<tr>
<th>True Covariance</th>
<th>Initial Guess</th>
<th>Final Estimate</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

RMSE shown for the initial guess covariances (red) the true $Q$ and $R$ (black) and the adaptive filter (blue)
Kalman-Takens Filter: Throwing out the model...

- Starting with historical observations \( \{y_0, \ldots, y_n\} \)

- Form Takens delay-embedding state vectors
  \[ x_i = (y_i, y_{i-1}, \ldots, y_{i-d})^\top \]

- Build an EnKF:
  - Apply analog forecast to each ensemble member
  - Use the observation function \( Hx_i = y_i \)
  - Crucial to estimate \( Q \) and \( R \)
Kalman-Takens applied to L96

Adaptive ensemble Kalman filtering of nonlinear systems
Papers with Franz Hamilton and Tim Sauer

http://math.gmu.edu/~berry/


Related/Background Material