Data assimilation with and without a model

Tyrus Berry
George Mason University

NJIT
Feb. 28, 2017

Postdoc supported by NSF
This work is in collaboration with:

Tim Sauer, GMU

Franz Hamilton, Postdoc, NCSU
DATA ASSIMILATION

\[ x_k = f(x_{k-1}) + \eta_k \quad \eta_k \in \mathcal{N}(0, Q) \]
\[ y_k = h(x_k) + \nu_k \quad \nu_k \in \mathcal{N}(0, R) \]

**Main Problem:** Given the model above plus observations \( y_k \),

- **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 \( p(x_{k-\ell} \mid y_1, \ldots, y_k) \)
- **Parameter estimation**

Apply ensemble Kalman filter (EnKF) to achieve these goals.
**DATA ASSIMILATION**

\[
x_k = f(x_{k-1}) + \eta_k \quad \eta_k \in \mathcal{N}(0, Q)
\]
\[
y_k = h(x_k) + \nu_k \quad \nu_k \in \mathcal{N}(0, R)
\]

Possible obstructions:

- Observations \( y_k \) mix system noise \( \eta_k \) with obs noise \( \nu_k \)
- Observations may be sparse in space or time
- Model error
  - \( Q \) and \( R \) may be unknown
  - Known model with unknown parameters
  - Wrong model, even with best fit parameters
  - Have model for some, not all of the variables
EXAMPLE 1. LORENZ 96

\[
\frac{dx^i}{dt} = -x^{i-2}x^{i-1} + x^{i-1}x^{i+1} - x^i + F
\]
EXAMPLE 2. MEA RECORDINGS

**Given:** Voltages + Model

**Want to find:**

- **State:**
  - Sodium
  - Potassium
  - Currents

- **Parameters**
  - Neuron
  - Network
**TWO STEP FILTERING TO FIND** $p(x_k | y_1, \ldots, y_k)$

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

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

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

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

  Posterior $\propto$ Prior $\times$ Likelihood

- **Alternative:** Variational methods (e.g. 3DVAR, 4DVAR)
  - Minimize a cost functional $\Rightarrow$ Hard optimization problem
BEST POSSIBLE SCENARIO

\[
\begin{align*}
    x_k &= f(x_{k-1}) + \eta_k & \eta_k &\in \mathcal{N}(0, Q) \\
    y_k &= h(x_k) + \nu_k & \nu_k &\in \mathcal{N}(0, R)
\end{align*}
\]

\textit{f} and \textit{h} are linear, all parameters known.

\[
\begin{align*}
    x_k &= F_{k-1}x_{k-1} + \eta_k & \eta_k &\in \mathcal{N}(0, Q) \\
    y_k &= H_kx_k + \nu_k & \nu_k &\in \mathcal{N}(0, R)
\end{align*}
\]
Kalman Filter

- Assume linear dynamics/obs and additive Gaussian noise

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

- 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 \( \ell \) \( \Rightarrow \) KF guaranteed to work!
**Kalman Filter**

- Assume linear dynamics/obs and additive Gaussian noise

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

\[
y_k = H_k x_k + \nu_k \quad \nu_k \sim \mathcal{N}(0, R)
\]

- Assume current estimate is Gaussian:

\[
p(x_{k-1} \mid y_1, \ldots, y_{k-1}) = \mathcal{N}(x_{k-1}^a, P_{k-1}^a)
\]

- **Forecast:** Linear combinations of Gaussians

  - **Prior:** \( p(x_k \mid y_1, \ldots, y_{k-1}) = \mathcal{N}(x_k^f, P_k^f) \)

    - \( x_k^f = F_{k-1} x_{k-1}^a \)
    - \( P_k^f = F_{k-1} P_{k-1} F_{k-1}^\top + Q \)

  - **Likelihood:** \( p(y_k \mid x_k, y_1, \ldots, y_{k-1}) = \mathcal{N}(y_k^f, P_k^y) \)

    - \( y_k^f = H_k x_k^f \)
    - \( P_k^y = H_k P_k^f H_k^\top + R \)
**Kalman Filter**

- Forecast: Linear combinations of Gaussians
  - Prior: \( p(x_k \mid y_1, \ldots, y_{k-1}) = \mathcal{N}(x_k^f, P_k^f) \)
    - \( x_k^f = F_{k-1}x_{k-1}^a \)
    - \( P_k^f = F_{k-1}P_{k-1}F_{k-1}^T + Q \)
  - Likelihood: \( p(y_k \mid x_k, y_1, \ldots, y_{k-1}) = \mathcal{N}(y_k^f, P_k^y) \)
    - \( y_k^f = H_kx_k^f \)
    - \( P_k^y = H_kP_k^fH_k^T + R \)

- Assimilation: Product of Gaussians (complete the square)
  \[
p(x_k \mid y_1, \ldots, y_k) = \mathcal{N}(x_k^f, P_k^f) \times \mathcal{N}(y_k^f, P_k^y) = \mathcal{N}(x_k^a, P_k^a)
\]
- Define the Kalman gain: \( K_k = P_k^fH_k^T(P_k^y)^{-1} \)
  - \( x_k^a = x_k^f + K_k(y_k - y_k^f) \)
  - \( P_k^a = (I - K_kH_k)P_k^f \)
**Kalman Filter Summary**

**Forecast**

\[ x_k^f = F_{k-1} x_{k-1} \]
\[ y_k^f = H_k x_k \]

**Covariance update**

\[ P_k^f = F_{k-1} P_{k-1}^a F_{k-1}^T + Q \]
\[ P_k^y = H_k P_k^f H_k^T + R \]

**Kalman gain & Innovation**

\[ K_k = P_k^f H_k^T (P_k^y)^{-1} \]
\[ \epsilon_k = y_k - y_k^f \]

**Assimilation**

\[ x_k^a = x_k^f + K_k \epsilon_k \]
\[ P_k^a = (I - K_k H_k) P_k^f \]
What about nonlinear systems?

- Consider a system of the form:
  \[
  x_{k+1} = f(x_k) + \omega_{k+1} \\
  y_{k+1} = h(x_{k+1}) + \nu_{k+1}
  \]
  \[\omega_{k+1} \sim \mathcal{N}(0, Q)\]
  \[\nu_{k+1} \sim \mathcal{N}(0, R)\]

- More complicated observability condition (Lie derivatives)

- Extended Kalman Filter (EKF):
  - Linearize \( F_k = Df(x_k^a) \) and \( H_k = Dh(x_k^f) \)

- Problem: State estimate \( x_k^a \) may not be well localized

- Solution: Ensemble Kalman Filter (EnKF)
ENSEMBLE KALMAN FILTER (EnKF)

Generate an ensemble with the current statistics (use matrix square root):

\[ x_t^i = \text{“sigma points” on semimajor axes} \]
\[ x_t^f = \frac{1}{2n} \sum F(x_t^i) \]
\[ P_{xx}^f = \frac{1}{2n-1} \sum (F(x_t^i) - x_t^f)(F(x_t^i) - x_t^f)^T + Q \]
**ENSEMBLE KALMAN FILTER (EnKF)**

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

\[
P_{yy} = (2n - 1)^{-1} \sum (y_t^i - y_t^f)(y_t^i - y_t^f)^T + R
\]

\[
P_{xy} = (2n - 1)^{-1} \sum (F(x_t^i) - x_t^f)(y_t^i - y_t^f)^T
\]

\[
K = P_{xy}P_{yy}^{-1} \text{ and } P_{xx}^a = P_{xx}^f - KP_{yy}K^T
\]

\[
x_{t+1}^a = x_t^f + K(y_t - y_t^f)
\]
PARAMETER ESTIMATION (STATE AUGMENTATION)

- When the model has parameters $\theta$,
  \[ x_{k+1} = f(x_k, \theta) + \omega_{k+1} \]
- Augment the state $\tilde{x}_k = [x_k, \theta_k]$, $\tilde{Q} = \begin{bmatrix} Q & 0 \\ 0 & Q^\theta \end{bmatrix}$
- Introduce trivial dynamics $d\theta = Q^\theta d\omega^\theta$
  \[ x_{k+1} = f(x_k, \theta_k) + \omega_{k+1} \]
  \[ \theta_{k+1} = \theta_k + \omega^\theta_{k+1} \]
- Need to tune the covariance $Q^\theta$ of $\omega^\theta$
- Can track slowly varying parameters, $Q^\theta \approx \text{var}(\theta)$
EXAMPLE OF PARAMETER ESTIMATION

Consider a network of $n$ Hodgkin-Huxley neurons

\[
\begin{align*}
\dot{V}_i &= -g_{Na}m^3 h(V_i - E_{Na}) - g_K n^4 (V_i - E_K) - g_L (V_i - E_L) \\
&\quad + I + \sum_{j \neq i} \Gamma_{HH}(V_j) V_j \\
\dot{m}_i &= a_m(V_i)(1 - m_i) - b_m(V_i)m_i \\
\dot{h}_i &= a_h(V_i)(1 - h_i) - b_h(V_i)h_i \\
\dot{n}_i &= a_n(V_i)(1 - n_i) - b_n(V_i)n_i \\
\Gamma_{HH}(V_j) &= \beta_{ij}/(1 + e^{-10(V_j+40)})
\end{align*}
\]

Only observe the voltages $V_i$

Recover all variables and the connection parameters $\beta$
EXAMPLE OF PARAMETER ESTIMATION

Can even turn connections on and off (grey dashes)

Variance estimate ⇒ statistical test (black dashes)
ROBUSTNESS TO MODEL ERROR?

Fit a generic spiking model (Hindmarsh-Rose)

\[
\begin{align*}
\dot{V}_i &= a_i V_i^2 - V_i^3 - y_i - z_i + I_i + \sum_{j \neq i}^{n} \Gamma_{HH}(V_j) V_j \\
\dot{y}_i &= (a_i + \alpha_i) V_i^2 - y_i \\
\dot{z}_i &= \mu_i (b_i V_i + c_i - z_i) \\
\Gamma_{HH}(V_j) &= \beta_{ij}/(1 + e^{-10(V_j+40)})
\end{align*}
\]

Observe voltages \( V_i \) from Hodgkin-Huxley!

Fit parameters to match neuron characteristics

Recover the connection parameters \( \beta \)
LINK DETECTION FROM NETWORKS OF MODEL NEURONS

Network of Hindmarsh-Rose neurons, modeled by Hindmarsh-Rose

Network of Hodgkin-Huxley neurons, modeled by Hindmarsh-Rose
L I N K  D E T E C T I O N  F R O M  M E A  R E C O R D I N G S
LINK DETECTION FROM MEA RECORDINGS

MEA Recording

Recovered Network

% of 160 sec each connection was statistically significant
**ENKF: 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)
**ENKF: INFLUENCE OF $Q$ AND $R$**

Standard Kalman Update:

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

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

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

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

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

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

![Graph showing the influence of $Q$ and $R$ on RMSE]
A D A P T I V E  F I L T E R: E S T I M A T I N G  Q  A N D  R

- Innovations contain information about Q and R

\[ \epsilon_k = y_k - y_k^f \]
\[ = h(x_k) + \nu_k - h(x_k^f) \]
\[ = h(f(x_{k-1}) + \omega_k) - h(f(x_{k-1}^a)) + \nu_k \]
\[ \approx H_k F_{k-1}(x_{k-1} - x_{k-1}^a) + H_k \omega_k + \nu_k \]

- 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 $R_k$:

\[
P^e_k = (H_{k+1} F_k)^{-1} (\epsilon_{k+1} \epsilon_k^T + H_{k+1} F_k K_k \epsilon_k \epsilon_k^T) H_k^{-T} \\
Q^e_k = P^e_k - F_{k-1} P^a_{k-1} F_{k-1}^T \\
R^e_k = \epsilon_k \epsilon_k^T - H_k P^f_k H_k^T
\]

- Note: $P^e_k$ is an empirical estimate of the background covariance
**ADAPTIVE ENKF**

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

**Original Kalman Eqs.**

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

\[
P_k^y = H_k P_k^f H_k^T + R_{k-1}
\]

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

\[
P_k^a = (I - K_k H_k) P_k^f
\]

\[
\epsilon_k = y_k - y_k^f
\]

\[
x_k^a = x_k^f + K_k \epsilon_k
\]

**Our Additional Update**

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

\[
+ K_{k-1} \epsilon_k \epsilon_{k-1}^T H_k^{-T}
\]

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

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

\[
Q_k = Q_{k-1} + (Q_{k-1} - Q_{k-1})/\tau
\]

\[
R_k = R_{k-1} + (R_{k-1} - R_{k-1})/\tau
\]
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

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

The Adaptive EnKF uses $F = 8$

We will consider model error where the true $F^i = \mathcal{N}(8, 16)$
RECOVERING $Q$ AND $R$, PERFECT MODEL

<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 (bottom right) for the initial guess covariances (red) the true $Q$ and $R$ (black) and the adaptive filter (blue)
COMPENSATING FOR MODEL ERROR

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 (bottom right) for the initial guess covariances (red) the perfect model (black) and the adaptive filter (blue)
OPEN PROBLEM

- Determining $Q^\theta$ for parameters fails: $d\theta = Q^\theta d\omega^\theta$
- Often $Q^\theta$ increases unrealistically or diverges
- Brownian motion $\Rightarrow$ Non-identifiability?
- Ornstein-Uhlenbeck works: $d\theta = \alpha(\bar{\theta} - \theta)d\theta + Q^\theta d\omega^\theta$
- But now need to estimate $\alpha, \bar{\theta}$!
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 RECONSTRUCTION
ANALOG FORECAST
ANALOG FORECAST
**Kalman-Takens filter: Lorenz-63**

EnKF w/ model

Kalman-Takens
**Kalman-Takens Filter**

Comparing K-T (red) with full model (blue)
KALMAN-TAKENS FILTER: LORENZ-96
MODEL ERROR

Lorenz-63

Lorenz-96
Kalman-Takens filter

Forecast error: El Nino index

![Graph showing forecast error for El Nino index](image-url)
LIMITS OF PARAMETER ESTIMATION

- Depends on model complexity and observability
- Typically estimate $\approx 4$ parameters per observation
- New work: (F. Hamilton et al., Hybrid modeling and prediction of dynamical systems)
  - Problem: Too many parameters, model is useless
  - Solution: To fit params, replace other equations with K-T
  - Extends the boundaries of parameter estimation
ESTIMATING UNMODELED DYNAMICS (FRANZ)

Want to track un-modeled variable $S$

\[
\dot{w} = F(w) + \omega_t \\
\begin{bmatrix}
y \\
\vdots \\
y \\
S 
\end{bmatrix} = H(w) + \nu_t
\]

Run $m$ model with different parameters $p_i$

\[
w = \begin{bmatrix}
x^1 \\
\vdots \\
x^m \\
c^1 \\
\vdots \\
c^m \\
d
\end{bmatrix}, \quad F = \begin{bmatrix}
f(x, p_1) \\
\vdots \\
f(x, p_m) \\
0 \\
\vdots \\
0 \\
0
\end{bmatrix}, \quad H(w) = \\
\begin{bmatrix}
h(x^1) \\
\vdots \\
h(x^m) \\
\sum_{i,j} c^i x^j + d
\end{bmatrix}
\]

Fit regression params $c^i_j$ and $d$ from training data $S$
RECONSTRUCTING UNMODELED IONIC DYNAMICS

Observing seizure voltage, reconstruct unmodeled potassium and sodium dynamics (assimilation model is Hindmarsh-Rose)

RECONSTRUCTING UNMODELED IONIC DYNAMICS
Observing seizure voltage, reconstruct unmodeled potassium and sodium dynamics (assimilation model is Hindmarsh-Rose)
RECONSTRUCTING UNMODELED IONIC DYNAMICS
Observing seizure voltage, reconstruct unmodeled potassium and sodium dynamics (assimilation model is Hindmarsh-Rose)
RECONSTRUCTING EXTRACELLULAR POTASSIUM FROM AN *In Vitro* NETWORK

We want to track extracellular potassium dynamics in a network but measurements are difficult and spatially limited.

Extracellular potassium is an **unmodeled variable**
RECONSTRUCTING EXTRACELLULAR POTASSIUM FROM AN *In Vitro* NETWORK

The local extracellular potassium in an MEA network can be reconstructed and predicted using our approach (assimilation model is Hindmarsh-Rose)
RECONSTRUCTING EXTRACELLULAR POTASSIUM FROM AN *In Vitro* NETWORK

The local extracellular potassium in an MEA network can be reconstructed and predicted using our approach (assimilation model is Hindmarsh-Rose)


**RECONSTRUCTING EXTRACELLULAR POTASSIUM FROM AN In Vitro NETWORK**

The local extracellular potassium in an MEA network can be reconstructed and predicted using our approach (assimilation model is Hindmarsh-Rose)

![Graph showing Actual, Observed, and Predicted potassium levels over time](image)
SUMMARY

- EnKF is a useful data assimilation technique for neurodynamics and other types of data
- Parameter estimation
- Adaptive QR is helpful when Q and R are unknown
- Difficulties
  - Model error
  - Unmodeled variables
  - No model
- Kalman-Takens filter
- Multimodel data assimilation
REFERENCES


▶ F. Hamilton, A. Lloyd, K. Flores, Hybrid modeling and prediction of dynamical systems. Submitted.
Kalman Filter: Forecast Step

- At time $k-1$ we have mean $x_{k-1}^a$ and covariance $P_{k-1}^a$

$$x_k = F_{k-1}x_{k-1} + \omega_k$$

- Linear combinations of Gaussians are still Gaussian so:

  - $p(F_{k-1}x_{k-1} \mid y_1, \ldots, y_{k-1}) = \mathcal{N}(F_{k-1}x_{k-1}^a, F_{k-1}P_{k-1}F_{k-1}^T)$
  - $p(x_k \mid y_1, \ldots, y_{k-1}) = \mathcal{N}(F_{k-1}x_{k-1}^a, F_{k-1}P_{k-1}F_{k-1}^T + Q)$

- Define the **Forecast mean**: $x_k^f \equiv F_{k-1}x_{k-1}^a$

- Define the **Forecast covariance**: $P_k^f \equiv F_{k-1}P_{k-1}F_{k-1}^T + Q$
Recall that \( y_k = H_k x_k + \nu_k \) where \( \nu_k \sim \mathcal{N}(0, R) \) is Gaussian.

The forecast distribution: \( p(x_k | y_1, ..., y_{k-1}) = \mathcal{N}(x^f_k, P^f_k) \)

Likelihood:
\[
p(y_k | x_k, y_1, ..., y_{k-1}) = \mathcal{N}(H_k x^f_k, H_k P^f_k H_k^\top + R)
\]

Define the **Observation mean**: \( y^f_k = H_k x^f_k \)

Define the **Observation covariance**: \( P^y_k = H_k P^f_k H_k^\top + R \)
Kalman Filter: Assimilation Step

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

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

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

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

- **Kalman Equations:** (after some linear algebra...)
  - Kalman Gain: $K_k = P_k^f H_k^\top (P_k^y)^{-1}$
  - Innovation: $\epsilon_k = y_k - y_k^f$
  - Posterior Mean: $x_k^a = x_k^f + K_k \epsilon_k$
  - Posterior Covariance: $P_k^a = (I - K_k H_k) P_k^f$
  - $x_k^a$ is the least squares/minimum variance estimator of $x_k$