The Ensemble filter and Variational problems

Douglas Nychka,
www.image.ucar.edu/~nychka

- The update/forecast cycle
- The Bayes posterior is the update
- Equivalence with the Kalman Filter
- The problems
- An ensemble solution
- Justifying sequential updates of the observations

Supported by the National Science Foundation DMS

The state of the system

The problem:

Estimate the state of a system u_t (or a field) at different times.

u_t can be:

- a vector or field of surface sources of CO_2 varying over time.
- pressure, temperature etc. on 3-d grid describing the state of the atmosphere
- a more general vector that describes the system possibly including parameters.

The basic data assimilation cycle

The problem is to estimate the state of a system x_t (or a field) at different times.

- Forecast for the state at time t
- New data, y_t comes in at time t
- Update x_t in light of the data
- Forecast ahead to time $t+1$ using updated state.

Cycle repeats ...

The DA cycle in Bayes language

- Forecast for the state at time t
  The prior for $[x_t]$
- New data, y_t comes in at time t
  The likelihood $[y_t|x_t]$
- Update x_t in light of the data (Bayes Thm)
  The posterior $[x_t|y_t]$
- Forecast ahead to time $t+1$ using updated state.
  The posterior $[M(x_t)|y_t] = [x_{t+1}|y_t]$

Cycle repeats ...

Today's posterior becomes tomorrows prior.

Doing the update or analysis step

The Gaussian-Gaussian case using Bayes theorem

- **Prior:** $N(\boldsymbol{x}_t^f, P_t^f)$
- **Likelihood:** $[\boldsymbol{y}_t|\boldsymbol{x}_t]$ **is** $N(H\boldsymbol{x}_t, R)$
- **Posterior:**

$$[\boldsymbol{x}_t|\boldsymbol{y}_t] \quad \text{is} \quad N(\boldsymbol{x}_t^a, P_t^a)$$

(products of Gaussians are Gaussian!)

$\boldsymbol{x}_t^a =$ **what you would expect here!**

$P_t^a =$ **messy and large**

Essentially $f \rightarrow a$

All the t **subscripts emphasize this is at a particular time.**

Now forecast ahead and the prior (or background) is computed from $\boldsymbol{x}_t^a$ and P_t^a.

You can not throw away or ignore P_t^a because you will need it to compute the background covariance for the next time step!

The posterior mean is a common estimate of $\boldsymbol{x}$.
It is also the Kalman filter update.

This estimate also maximizes the posterior.
(or minimizes minus log posterior.)

-log Posterior =

$$(\boldsymbol{y} - H\boldsymbol{x})^T R^{-1} (\boldsymbol{y} - H\boldsymbol{x})/2 \; + \; (\boldsymbol{x} - \boldsymbol{x}^f)^T (P^f)^{-1} (\boldsymbol{x} - \boldsymbol{x}^f)/2$$

$+$ *other stuff.*

To minimize take the derivative and set equal to zero.

The Mininizer

By straight forward multivariate calculus:

$$\boldsymbol{x}_t^a = \boldsymbol{x}_t^f + (H^T R^{-1} H \ + \ (P_t^f)^{-1})^{-1} H^T R^{-1} (\boldsymbol{y}_t - H\boldsymbol{x}_t^f)$$

$$\boldsymbol{x}_t^a = \boldsymbol{x}_t^f + \mathcal{W}(\boldsymbol{y}_t - H\boldsymbol{x}_t^f)$$

$\mathcal{W}$= Kalman Gain

The strange case of the two equations for the same thing.

Instead of minimizing over T reparametrize in terms of vectors α and β so that

$$x = x^f + (P^f)^{1/2} H^T \alpha + \beta \quad \textit{and} \; \beta^T H^T \alpha = 0$$

The solution is $\beta = 0$

$$\mathcal{W} = (H^T P_t^f)(H P_t^f H^T + R)^{-1}$$

COV(state,data) $\times$ COV(data)$^{-1}$

The problems with this cycle

- For large state vectors computing the posterior mean involves large linear systems.
- Finding the covariance matrix is even worse: P_a could be $10^7 \times 10^7$
- Even when we know $[x_t|y_t]$ how to we find $[M(x_t)|y_t]$ when M is nonlinear?

The ensemble approximation, 3 steps

Recall

$$\textit{Kalman gain} = (H^T P^f)(H P^f H^T + R)^{-1}$$

Suppose we have a sample (an ensmeble) from $N(x^f, P^f)$:
$\{x_1^f, x_2^f, ...x_M^f\}$

1 Cheat on P^f and x_f

- **Replace the forecast covariance matrix with the sample covariance matrix based on the ensemble.**
- **Replace the forecast mean with the ensemble mean.**

2 Create a new ensemble using the analysis mean and covariance

- $x_j^a = x_j^f + \mathcal{W}(\boldsymbol{y} - H\boldsymbol{x}^f + \epsilon_j)$
- ϵ_j **is a random draw from** $N(0, R)$ **(usually** $\bar{\epsilon} = 0$**)**

Why does this work?

3 Propagate each ensemble member forward using the model

- $M(x_1^a), M(x_2^a), ..., M(x_M^a)$
  The forecast ensemble at time $t + 1$

This step is exact if the ensemble comes from $[\boldsymbol{x}_t|\boldsymbol{y}_t]$**!**

Comments

- There are many ways of creating the ensemble spread in Step 2.
  Adding noise is called perturbed observations.
  A deterministic adjustment is the transform filter.
- The covariance matrix estimated from the ensemble needs to be modified in several ways to reduce the effect of sampling on the Kalman gain.
- In nonlinear systems, like the atmosphere, the filter parameters often must be tuned to be stable.

But this really works!

Justifying a sequential update

Do we have to update the state with all the observations at once?

y_1 and y_2 are independent given x

$$[x|y_1, y_2] = \frac{[y_1, y_2|x][x]}{[y_1, y_2]} = \frac{[y_2|x][y_1|x][x]}{[y_1][y_2]}$$

$$= \frac{[y_2|x][y_1|x][x]}{[y_1][y_2]} \propto [y_2|x][x|y_1]$$

$[x|y_1]$ is the result of updating x with just y_1.

$[x|y_1]$ now becomes the prior for updating just y_2.