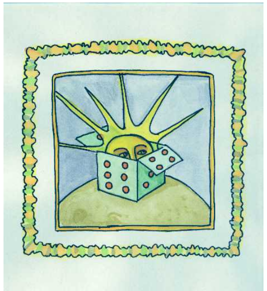
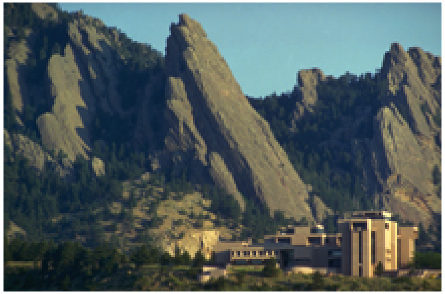
Forecasting and data assimilation

Douglas Nychka, Thomas Bengtsson, Chris Snyder
Geophysical Statistics Project
National Center for Atmospheric Research

- **Numerical models**
- **Kalman Filter**
- **Ensembles**
- **Local-Local filter for non-Gaussian distributions**





Supported by the National Science Foundation DMS

Overview

Data Assimilation (DA):

Combining predictions made by a numerical model with observed data to estimate the state of a system, $\boldsymbol{x}$.
This is also called a *filter*: We filter the noisy observations to estimate the state.

The statistical foundation is Bayes Theorem and the uncertainty in the state of the system is represented by a probability distribution.

PRIOR for $\boldsymbol{x}$ + observations → POSTERIOR for $\boldsymbol{x}$

Usually the assimilation is done at many consecutive time points and the practical implementation involves many shortcuts to approximate a posterior distribution.

Why do this?

- **Forecast the weather**
- **Forecast air quality**
- **Assimilating data for a given geophysical model may be one of the few ways to test it.**
- **Solve an** *inverse* **problem such as estimating sources and sinks of pollutant precursors.**

Some key ideas:

- **Represent a continuous distribution by a random sample.**
- **Only update state variables ”local” to the observations**
- **Use local regression to update the state variables**

Contribution:

A Local-Local filter to handle non-Gaussian data assimilation problems.

40-Dimensional Lorenz System (Lorenz, 1996)

- **Atmospheric system describing k values of an atmospheric variable at k longitudes:** $x_1, \ldots, x_{40}$. **(Subscript denotes spatial location.)**
- **Equations: for** $j = 1, \ldots, 40$,

$$\dot{x}_j = x_{j-1}(x_{j+1} - x_{j-2}) - x_j + F,$$

  where F **represents forcing.**

- **The equations contain quadratic nonlinearities mimicking advection:**

  $\dot{u}_i \;\propto\; u_i \frac{\partial u_i}{\partial x} \;\approx\; u_i(u_{i'} - u_{i^\star})/\delta$**.**

- F **is chosen so that phase space is bounded and the system exhibits chaotic behavior.**
- **'observe'** $z_2, z_4, \ldots, z_{40}$**:** $y_j = z_j + \epsilon_j$**,** $\epsilon_j \sim N(0, .5^2)$ **and** $\delta t = .20$**.**

Atmospheric models 101

- **A deterministic numerical model that describes the circulation of the atmosphere.**
- **State of system, $\boldsymbol{x}_t$ at time t defined on a 3-d grid of the the atmosphere.**
  Community Atmospheric Model (CAM) $128 \times 64 \times 30$ **boxes (**$\approx 280km$**)**
  Rapid Update Cycle Model (RUC) **is run on part of the earth but on a** $40km$ **grid.**
- **Evolution of the model is governed by a discretizing the nonlinear equations of motion derived from fluid dynamics, usually deterministic.**

$$\boldsymbol{x}_{t+1} = g(\boldsymbol{x}_t)$$

g **is nonlinear, complicated and fairly expensive to evaluate**

Making a deterministic forecast:

$$\hat{\boldsymbol{x}}_{t+1} = g(\hat{\boldsymbol{x}}_t)$$

The Bayes cycle

$$p(\mathbf{x}_t),\ \ \mathbf{y}_t \xrightarrow{\textbf{Bayes}} p(\mathbf{x}_t|\mathbf{y}_t) \qquad \xrightarrow{g(.)} p(\mathbf{x}_{t+1}|\mathbf{y}_t) = p(\mathbf{x}_{t+1}),\ \ \mathbf{y}_{t+1}$$

Yesterday's posterior becomes today's prior!

Standard Kalman Filter/ conditional multivariate normal distributions

This is easy in closed form if everything is *multivariate normal* **and** *linear***.**

Observation Model

$$\boldsymbol{y} = H\boldsymbol{x}_t + \boldsymbol{e}$$

with

$$\boldsymbol{e} \sim MN(0, R)$$

Prior

$$\boldsymbol{x}_t \sim MN(\boldsymbol{\mu}_t, \Sigma)$$

Kalman update for state

$$\hat{\boldsymbol{x}}_t = E(\boldsymbol{x}_t|\boldsymbol{y}) = \boldsymbol{\mu}_t + \Sigma H^T(H\Sigma H^T + R)^{-1}(\boldsymbol{y} - H\boldsymbol{\mu}_t)$$

Kalman update for covariance

$$VAR(\boldsymbol{x}_t|\boldsymbol{y}) = P_a^t = \Sigma - H^T(H\Sigma H^T + R)^{-1}H$$

Forecast mean:

Assume that g is linear

$$\hat{\boldsymbol{x}}_{t+1} = G\hat{\boldsymbol{x}}_t$$

Forecast covariance:

$$P_{f,t}^{t+1} = GP_a^t G^t$$

A qualifier problem:

These are just results based on the conditional distributions for the multivariate normal because everything is assumed to be Gaussian or a linear transformation.

Problems

- $\boldsymbol{x} \approx 10^6 - 10^7$ **and** $\boldsymbol{y} \approx 10^5 - 10^6$
  So even with closed form expressions the computations may not be feasible because the linear systems are *huge*.

- **Finding**

  $$p(\boldsymbol{x}_{t+1}|\boldsymbol{y}_t) = p(g(\boldsymbol{x}_t)|\boldsymbol{y}_t)$$

  from

  $$p(\boldsymbol{x}_t|\boldsymbol{y}_t)$$

  is the mother of all change of variable problems!

- P_f **can not be stored or directly propagated.**

Ensembles

Each distribution is represented by a random sample of the states called an *ensemble*.

In place of

$$\pi(\boldsymbol{x}_t|y_t) \longrightarrow \pi(g(\boldsymbol{x}_t)|y_t)$$

propagate each ensemble member.

$$\begin{array}{ccc} \boldsymbol{x}_{t,1} & & g(\boldsymbol{x}_{t,1}) = \boldsymbol{x}_{t+1,1} \\ \boldsymbol{x}_{t,2} & g \longrightarrow & g(\boldsymbol{x}_{t,2}) = \boldsymbol{x}_{t+1,2} \\ \vdots & & \vdots \\ \boldsymbol{x}_{t,M} & & g(\boldsymbol{x}_{t,M}) = \boldsymbol{x}_{t+1,M} \end{array}$$

By elementary probability:

$\{\boldsymbol{x}_{t,j}\}$ **is a random sample from** $p(\boldsymbol{x}_t|y_t)$ **implies** $\{\boldsymbol{x}_{t+1,j}\}$ **will be a random sample from** $p(\boldsymbol{x}_{t+1}|y_t)$

Ensemble Kalman filter (EKF)

- If the observations have independent errors, the observations can be assimilated sequentially to get the same result.
- Wherever a covariance matrix or mean vector appears replace these by the *sample* quantities from the ensemble. The covariance matrix is tapered to be a better estimate and inflated to make the filter stable.
- Sampling to get the new ensemble for posterior is computed in a very similar way as the standard update (perturbed observation method).

Ensemble Kalman filter (EKF) (continued)

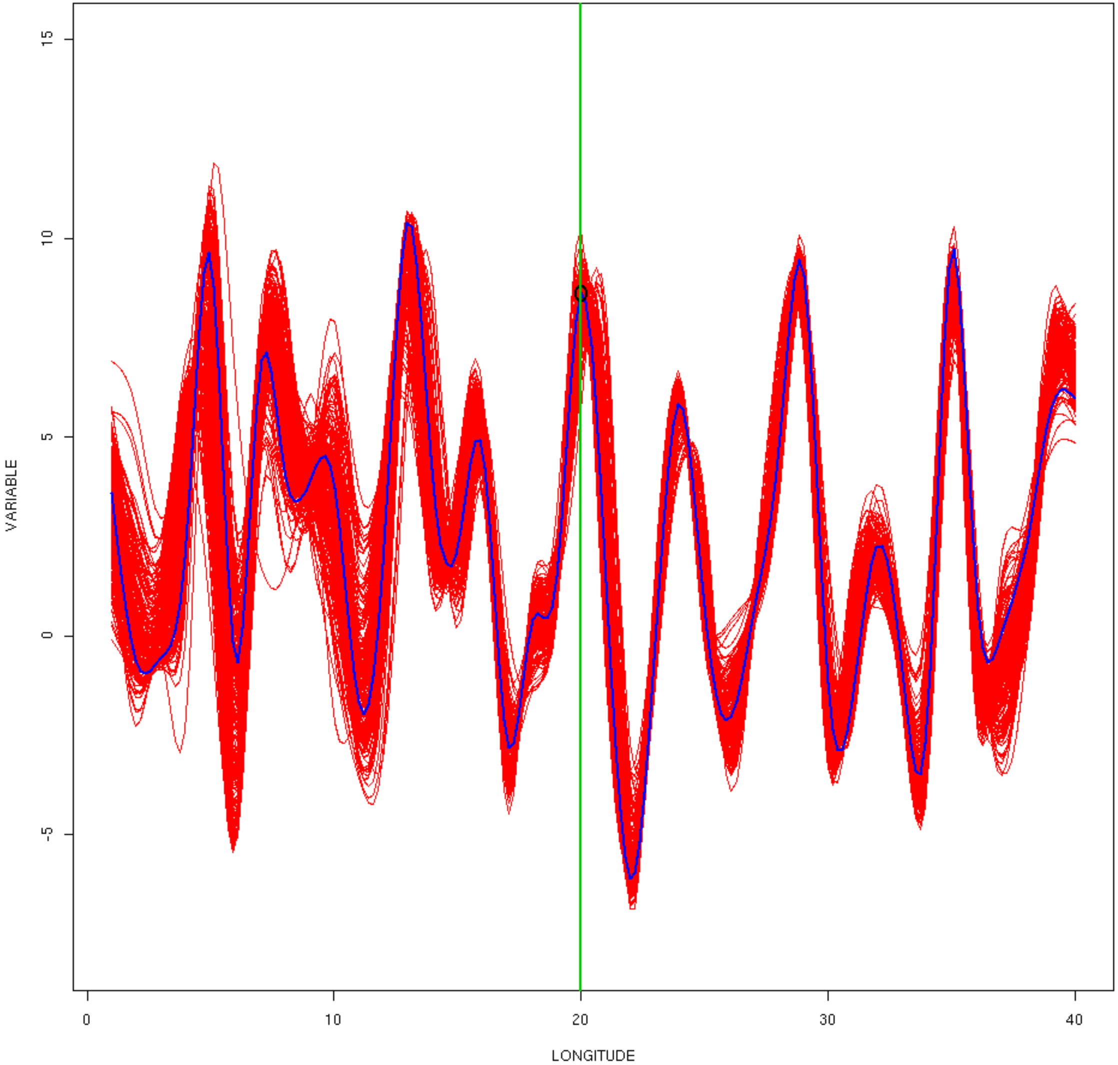
The first point suggests a double loop algorithm:

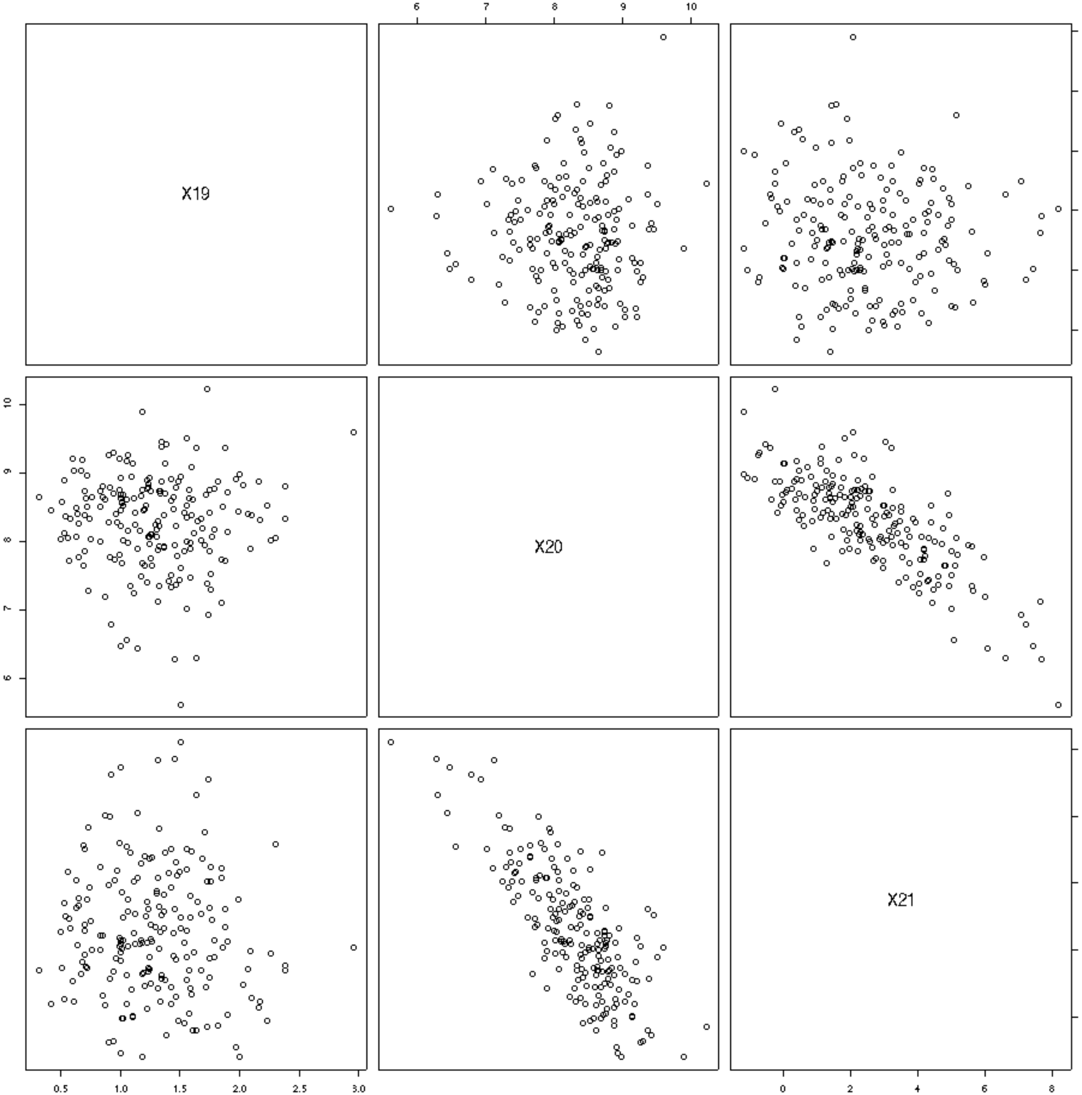
Assimilating at a given time:

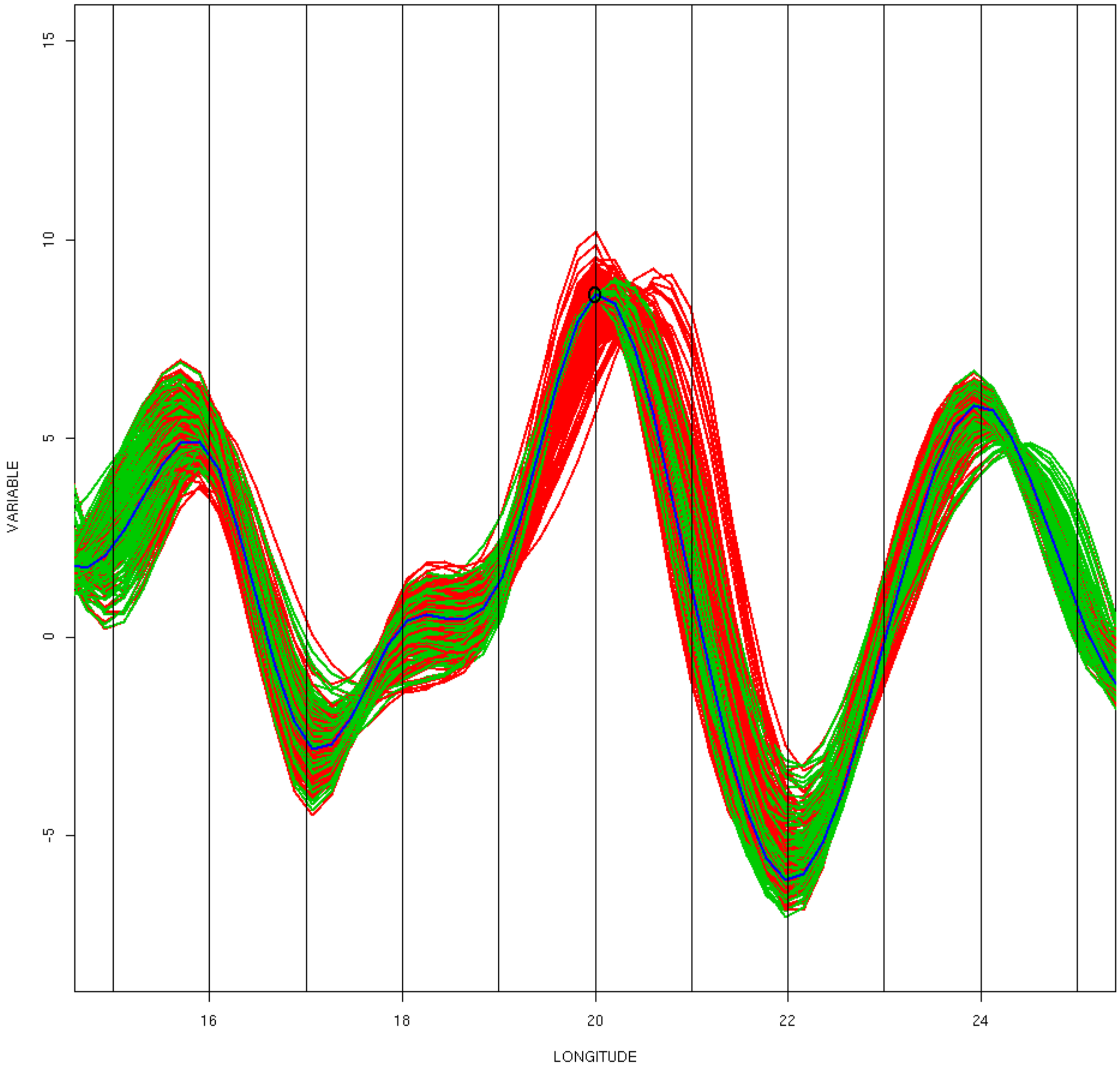
```
Loop over observations $\{y_1, y_2, ...y_n\}$
    Loop over ensemble members: $\{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_M\}$
        Update ensemble member $\boldsymbol{x}_i$ based on $y_j$
        Modify components of $\boldsymbol{x}_i$ “close” to $y_j$
```

A key aspect is that the observation only changes part of the state vector that is ”close” to it due to the covariance tapering.

Given an observation in Washington,DC update a Baltimore grid point ...
but a grid point near Moscow is unchanged.


15
10
5
0
-5
VARIABLE
0
10
20
30
40
LONGITUDE


X19
X20
X21
6
7
8
9
10
3.0
2.5
2.0
1.5
1.0
0.5
10
9
8
7
6
8
6
4
2
0
0.5
1.0
1.5
2.0
2.5
3.0
0
2
4
6
8


15
10
5
0
-5
VARIABLE
16
18
20
22
24
LONGITUDE

Tapering a covariance

Suppose the variables x_i and x_j are separated by physical distance d_{ij} and $\psi(d)$ is a positive definite function, with compact support.

Use

$$\hat{\Sigma}_{ij}\psi(d_{ij})$$

It is an open question how/why the approximations and tuning parameters in the EKF change its statistical performance.

BTW:

By tapering with a compactly supported kernel we (Furrer, Genton and Nychka (2004)) can accurately ”krig” 5000 points and evaluate the surface on a 1000×1000 **grid in** $\approx$ **15 seconds**

... in R!

This taper method will scale linearly with the number of observations.

Non-Gaussian distributions

Given that g is nonlinear one can not expect Gaussian distributions.
Represent the prior distributions as mixtures of multivariate normals
$p(\mathbf{x}_t) = \sum_{i=1}^{k} p_i \mathbf{MN}(\boldsymbol{\mu}_i, \mathbf{P}_i)$

The posterior distribution is also a mixture:
$p(\mathbf{x}_t|\mathbf{y}_t) = \sum_{i=1}^{k} p_i^\star \mathbf{MN}(\boldsymbol{\mu}_i^\star, \mathbf{P}_i^\star)$

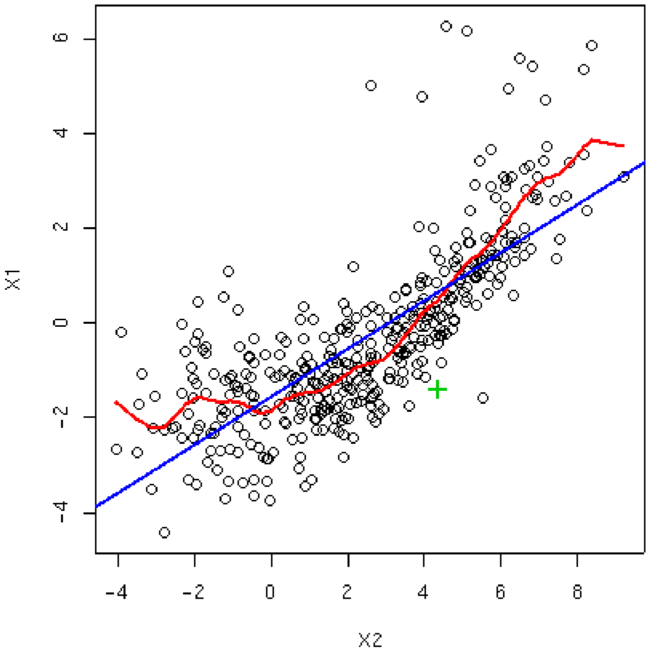
Ensembles as a mixture distribution:

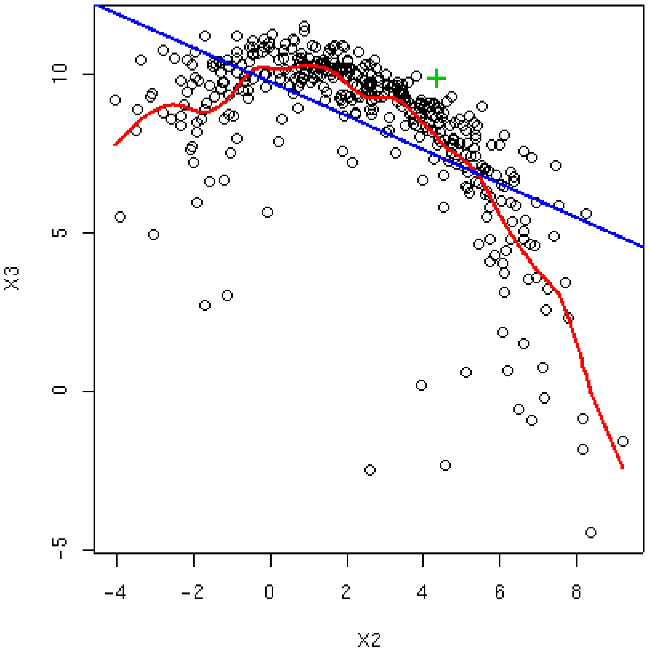
Each ensemble member is the center of a mixture where the covariance is the sample covariance of its nearest neighbors.

From the form for the posterior mean using ensembles the posterior probabilities look like weights based on a normal kernel. The use of neighborhoods to find the covariance results in a local linear regression.

A non-Gaussian update

Observe $X2$ with error and wish to update $X1$ and $X3$.





Local-local filter

Curse of dimensionality:

**Even 40 dimensions is too large a state space to apply the mixture ensemble filter directly.
In 40 dimensions every state vector is far away from every other!**

Basic idea is to only use the mixture model for components of the state vector close to the observations. Otherwise use the usual EKF for updating components.
For example
Given an observation of Y_2 consider non-Gaussian model to update

(X_1, X_2, X_3)

Complement, $(X_4,$ ..., $X_{40})$ is updated using the Gaussian EKF.

LOCAL-LOCAL

LOCAL in physical space:

Only update components close to the observation location.

LOCAL in state space:

only use ensemble members that are ”close” to the observed value.

We have worked on this for two years and in the process rediscovered non-parametric regression kernel smoothing!

The reason this was not obvious is because it is the numerical model that is generating the approximate regression relationship through the ensemble members. Also we got hung up on the Bayesian thing ...

Summary

Results

- We have some evidence that the practical version of the EKF actually handles non-Gaussian distributions better than an exact Kalman filter.
- The Local-Local filter clearly out performs EKF in a simple 3-d system especially in places where g is very nonlinear.
- A version of the L-L filter also performs better than EKF with about 5% improvement (without any extensive tuning) for the 40 variable model.

Some Future Work

- Adaptive estimates of tuning parameters
- Robust estimators

- **Exploring more realistic test systems, e.g. primitive equation models for a dry atmosphere.**