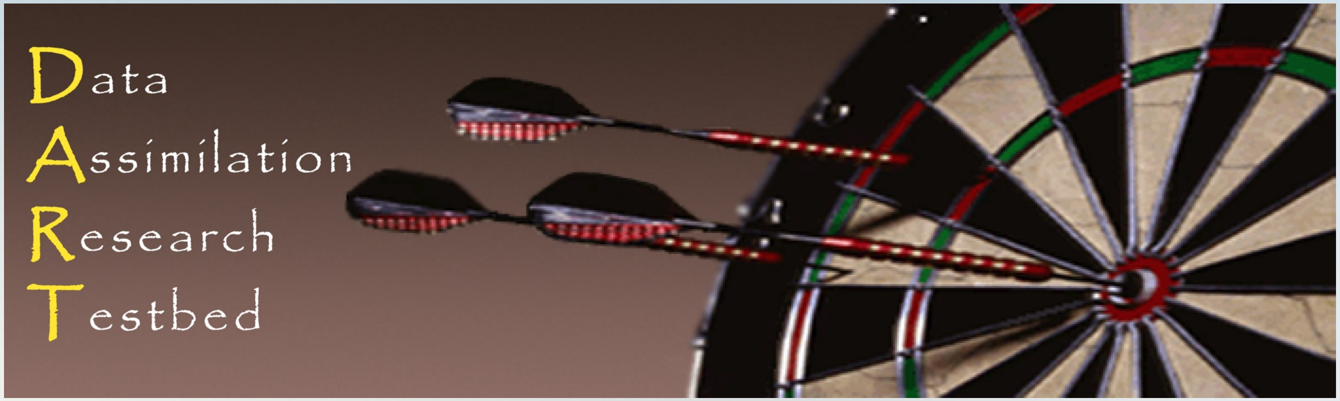
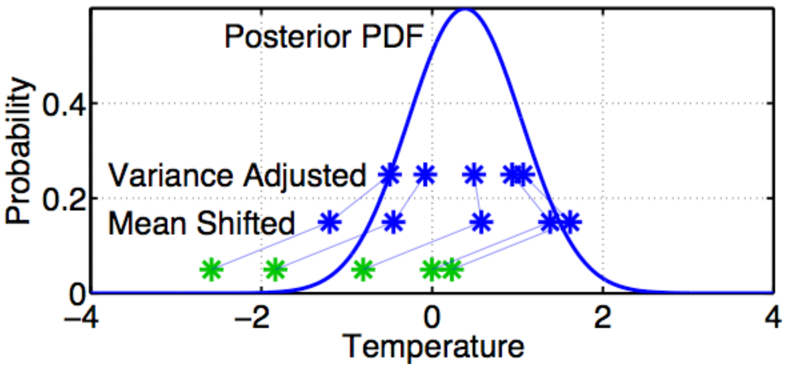


DART_LAB Tutorial Section 2: How Should Observations Impact an Unobserved State Variable? Multivariate Assimilation.

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

Single observed variable, single unobserved variable.

So far, we have a known likelihood for a single variable.

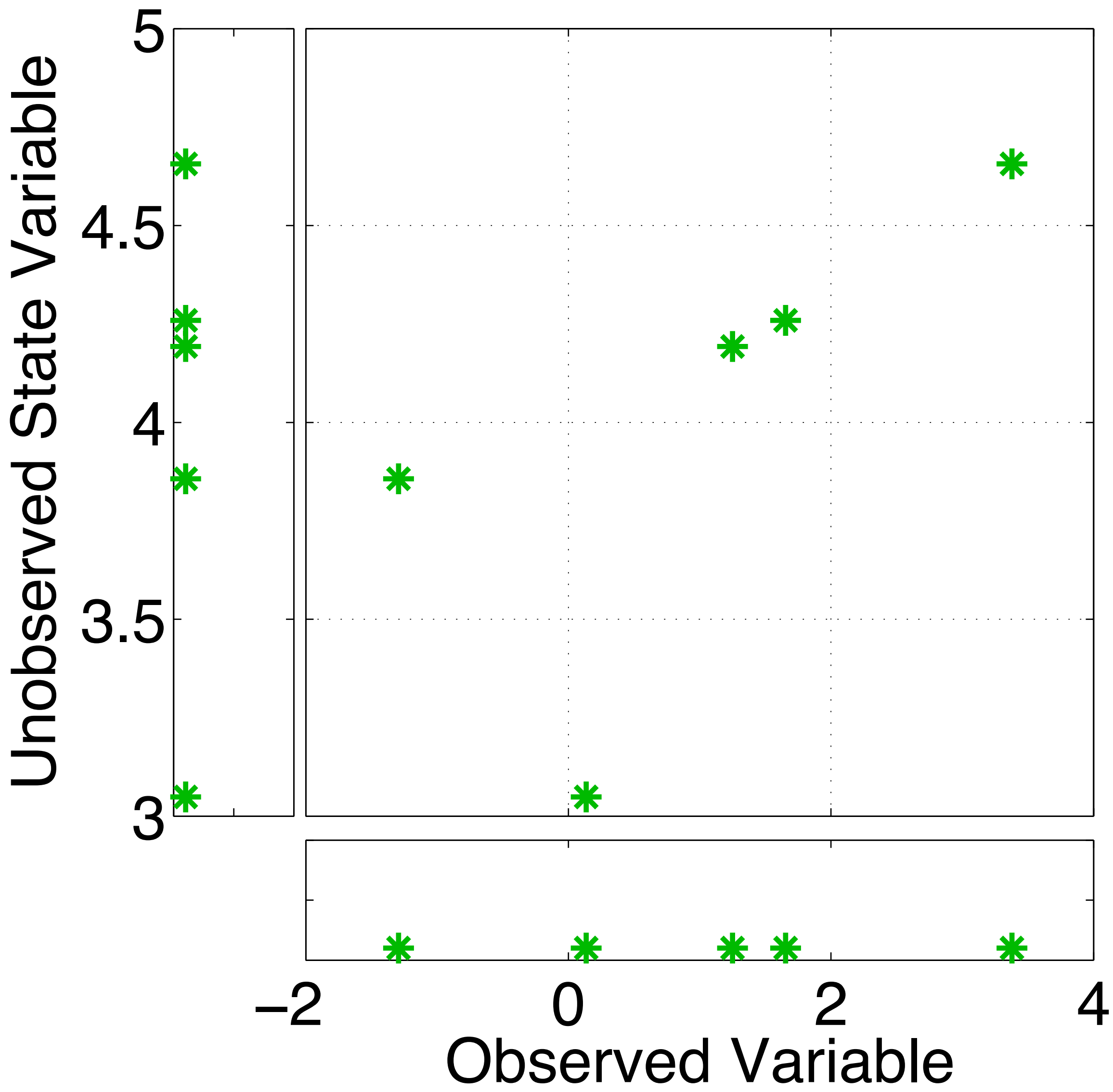


Now, suppose the prior has an additional unobserved variable …

We will examine how ensemble members update the additional variable.

Basic method generalizes to any number of additional variables.

Ensemble filters: Updating additional prior state variables

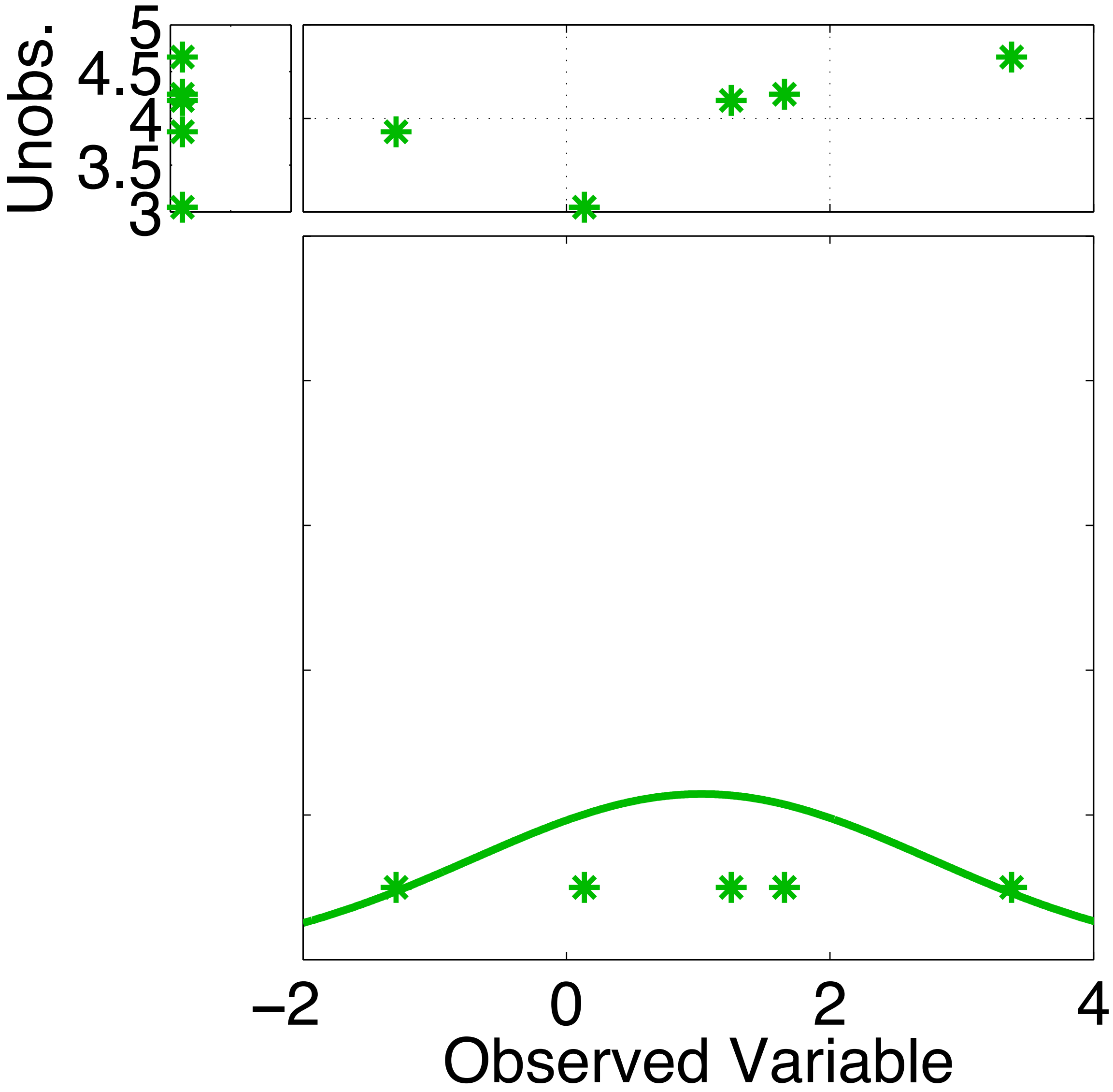


Assume that all we know is the prior joint distribution.

One variable is observed.

What should happen to the unobserved variable?

Ensemble filters: Updating additional prior state variables

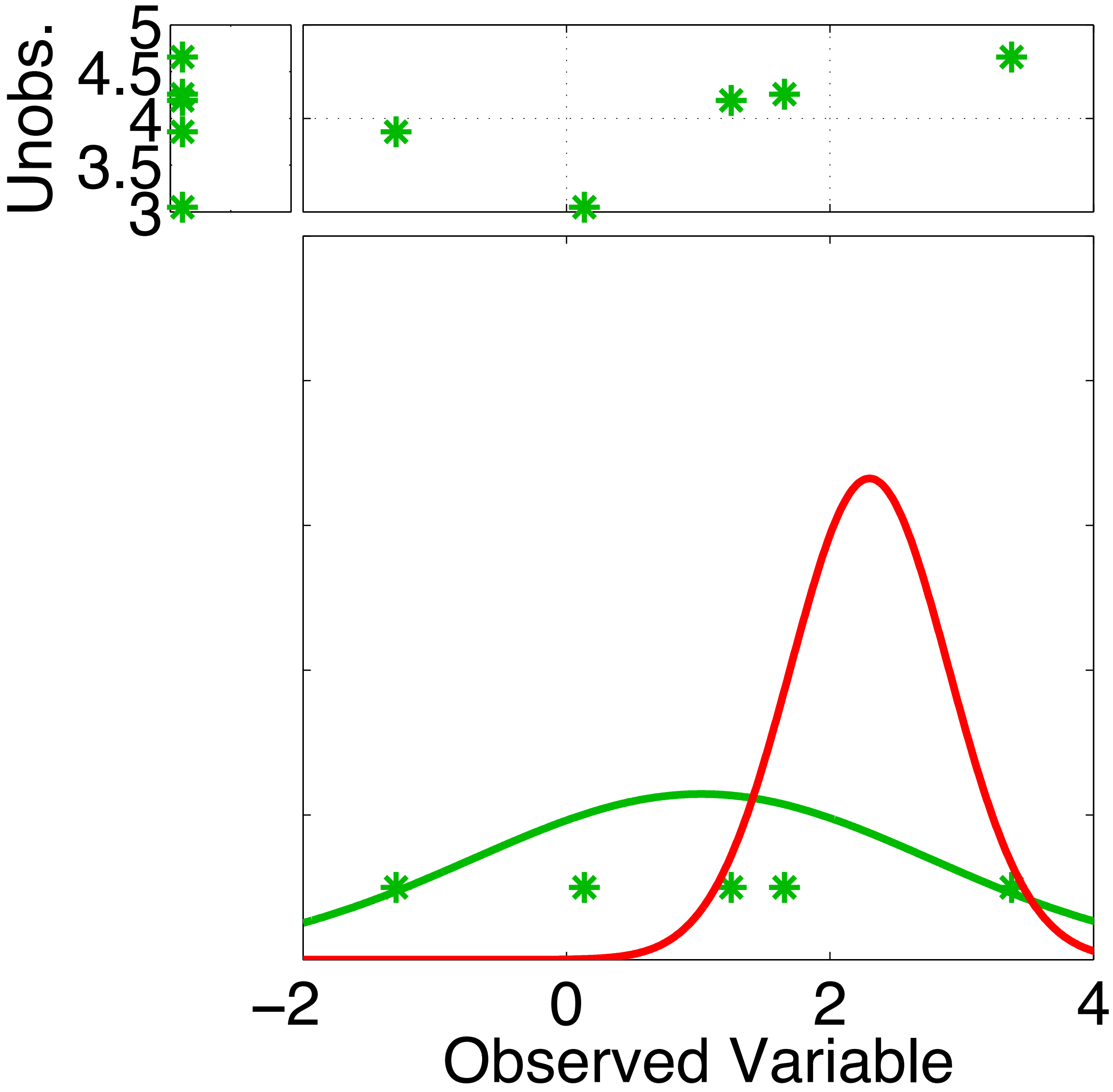


Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable as in section 1.

Ensemble filters: Updating additional prior state variables

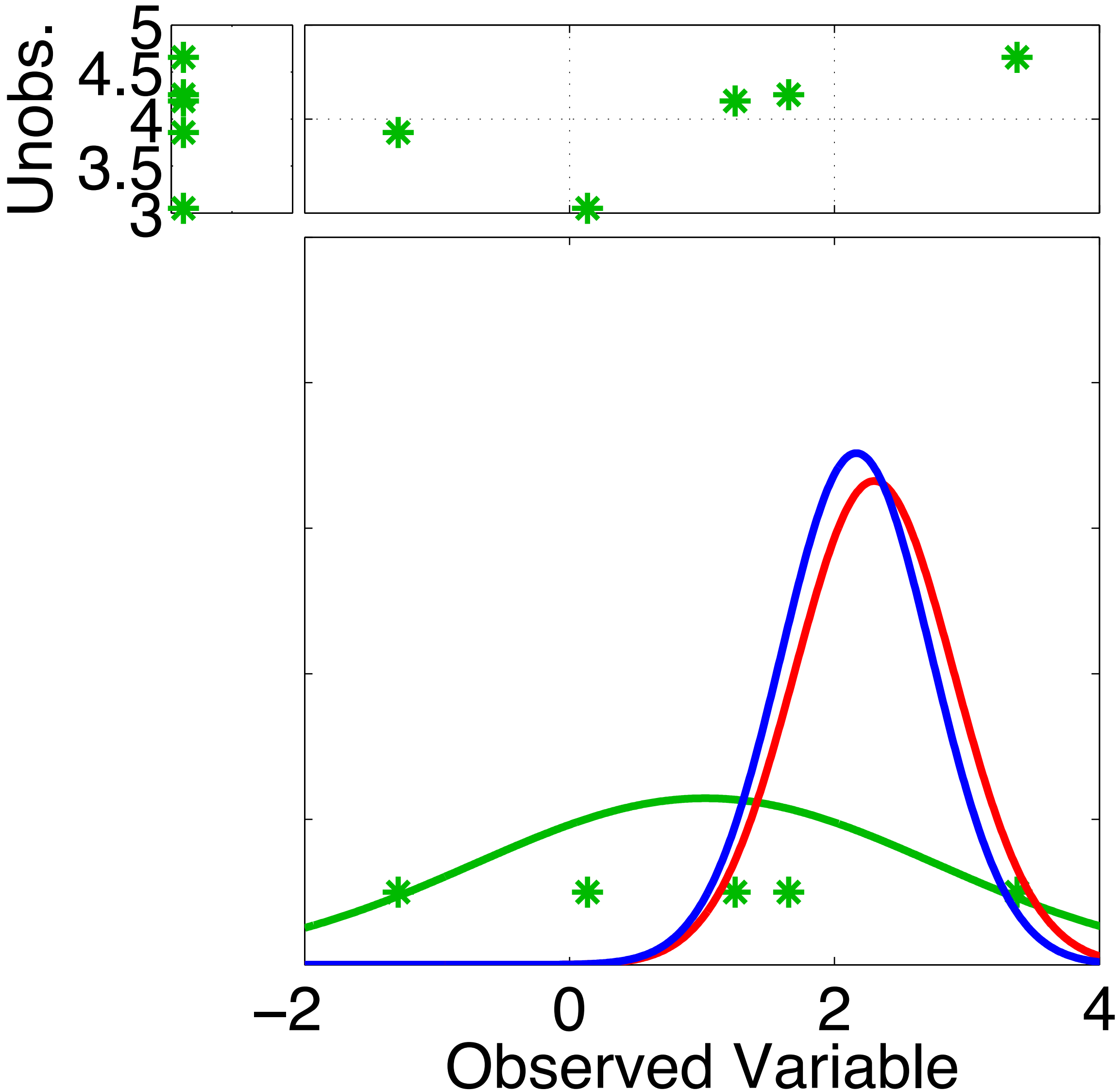


Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable as in section 1.

Ensemble filters: Updating additional prior state variables

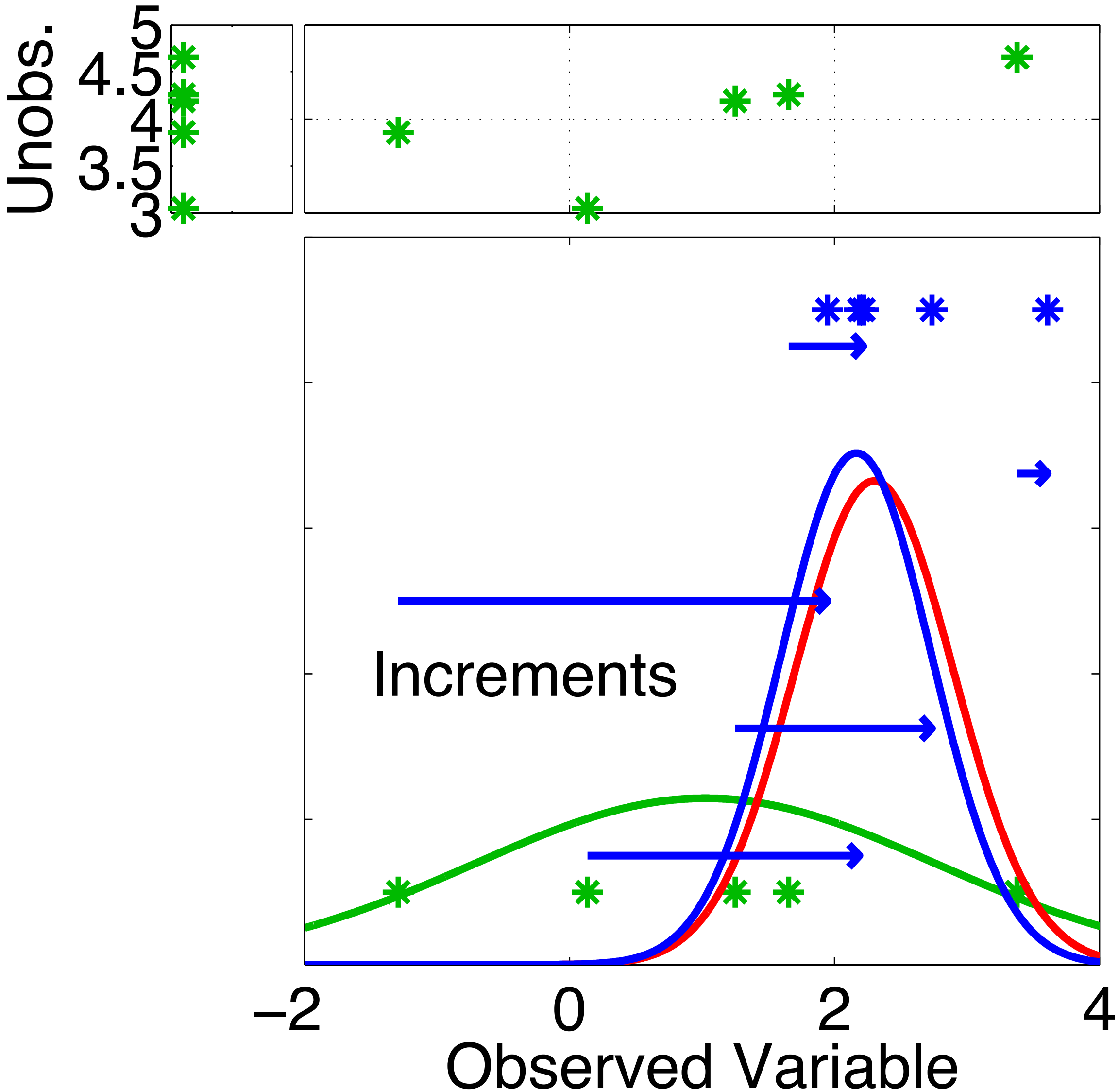


Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable as in section 1.

Ensemble filters: Updating additional prior state variables

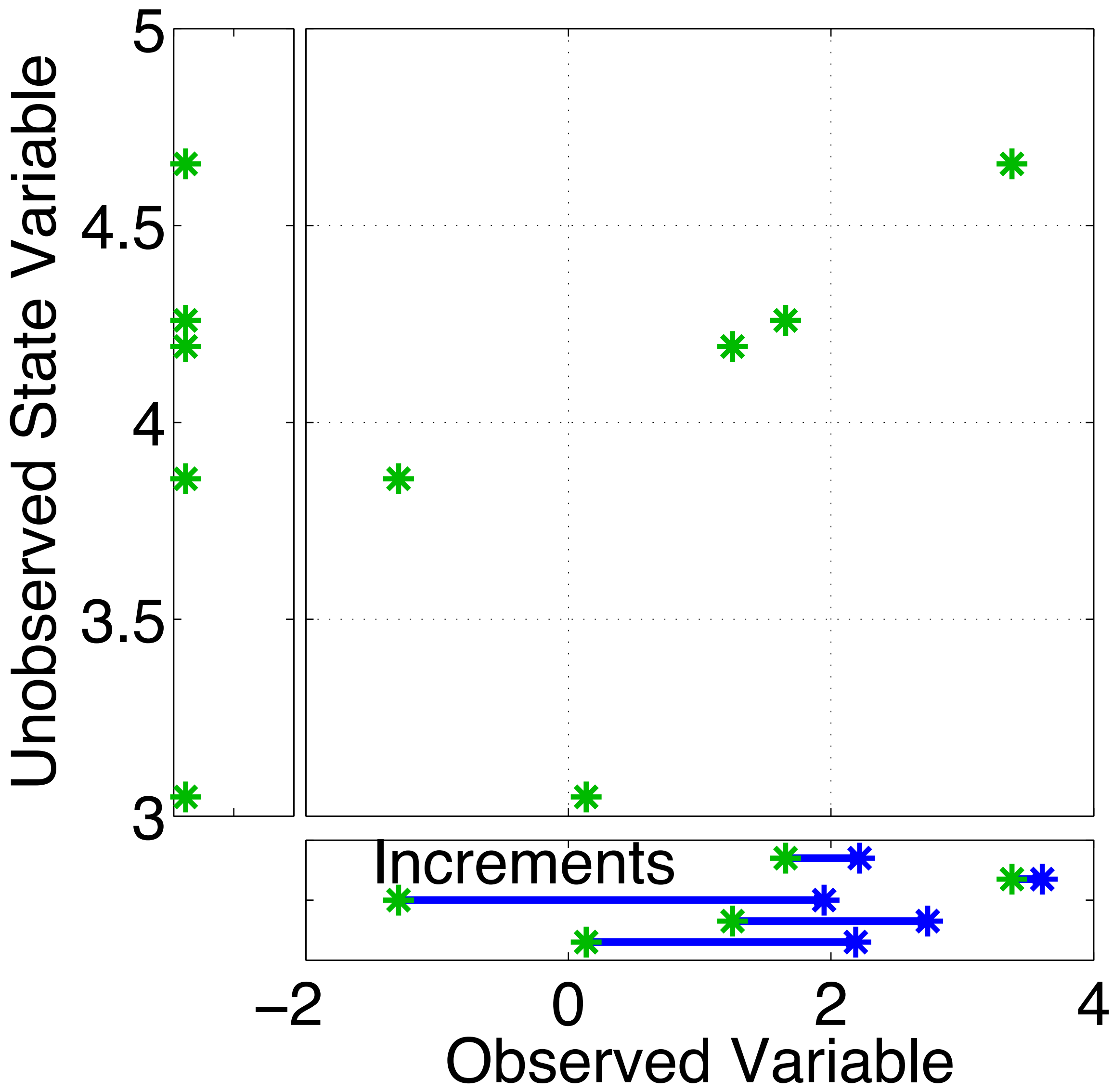


Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.

Ensemble filters: Updating additional prior state variables



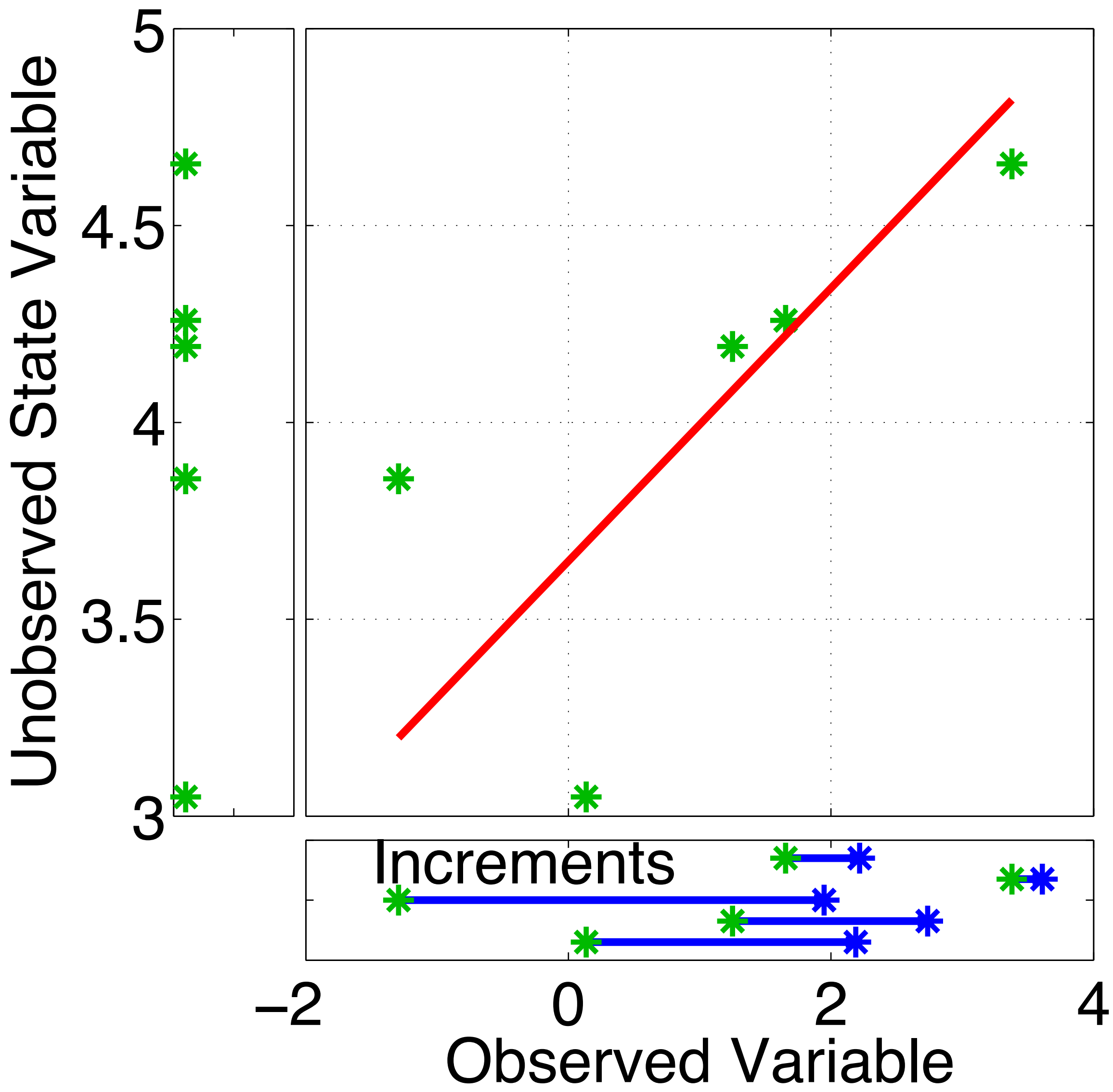
Assume that all we know is the prior joint distribution.

How should the unobserved variable be impacted?

1st choice: least squares.

Equivalent to linear regression.

Same as assuming binormal prior.



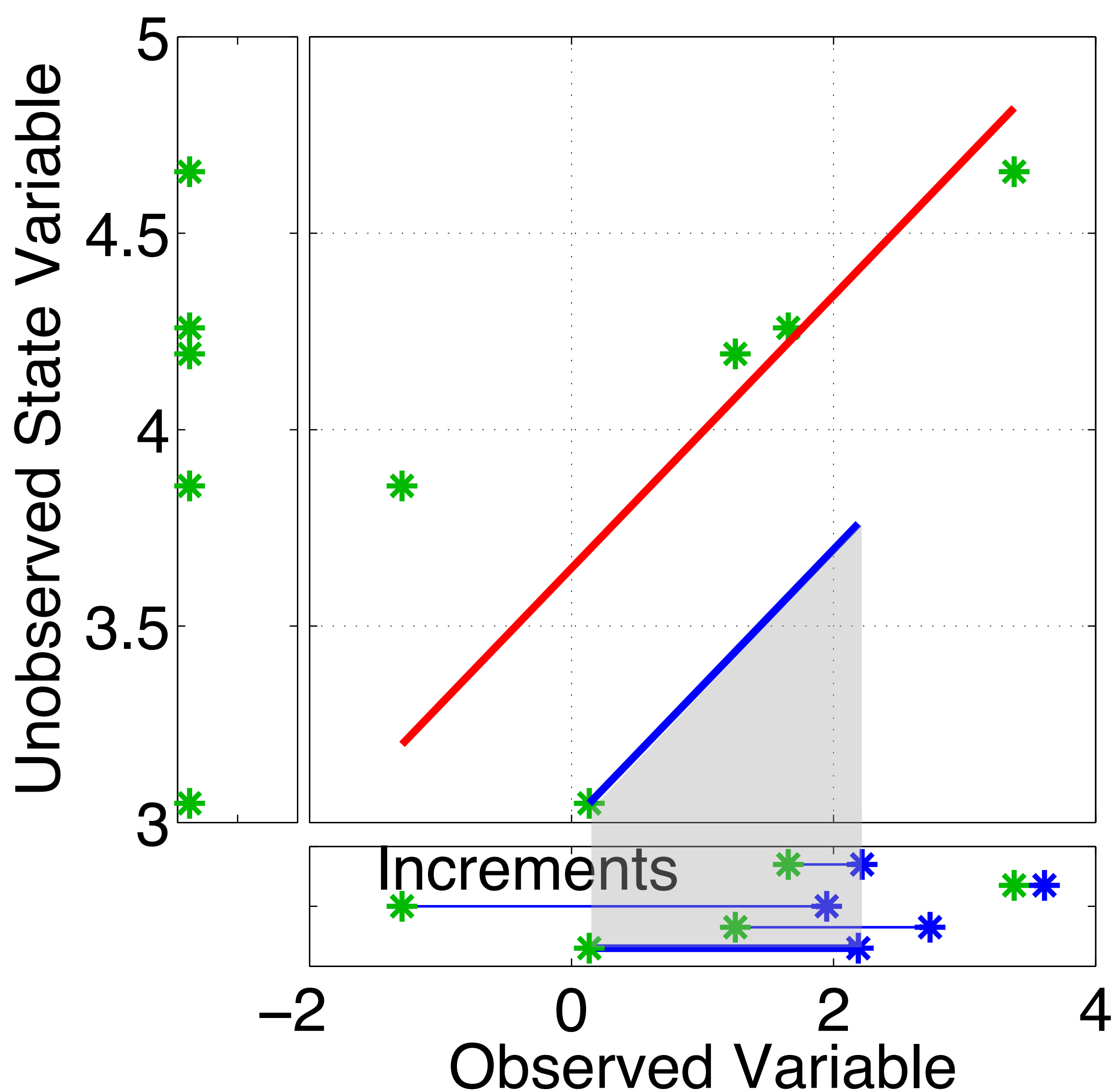
Have joint prior distribution of two variables.

How should the unobserved variable be impacted?

1st choice: least squares.

Begin by finding least squares fit.

Ensemble filters: Updating additional prior state variables

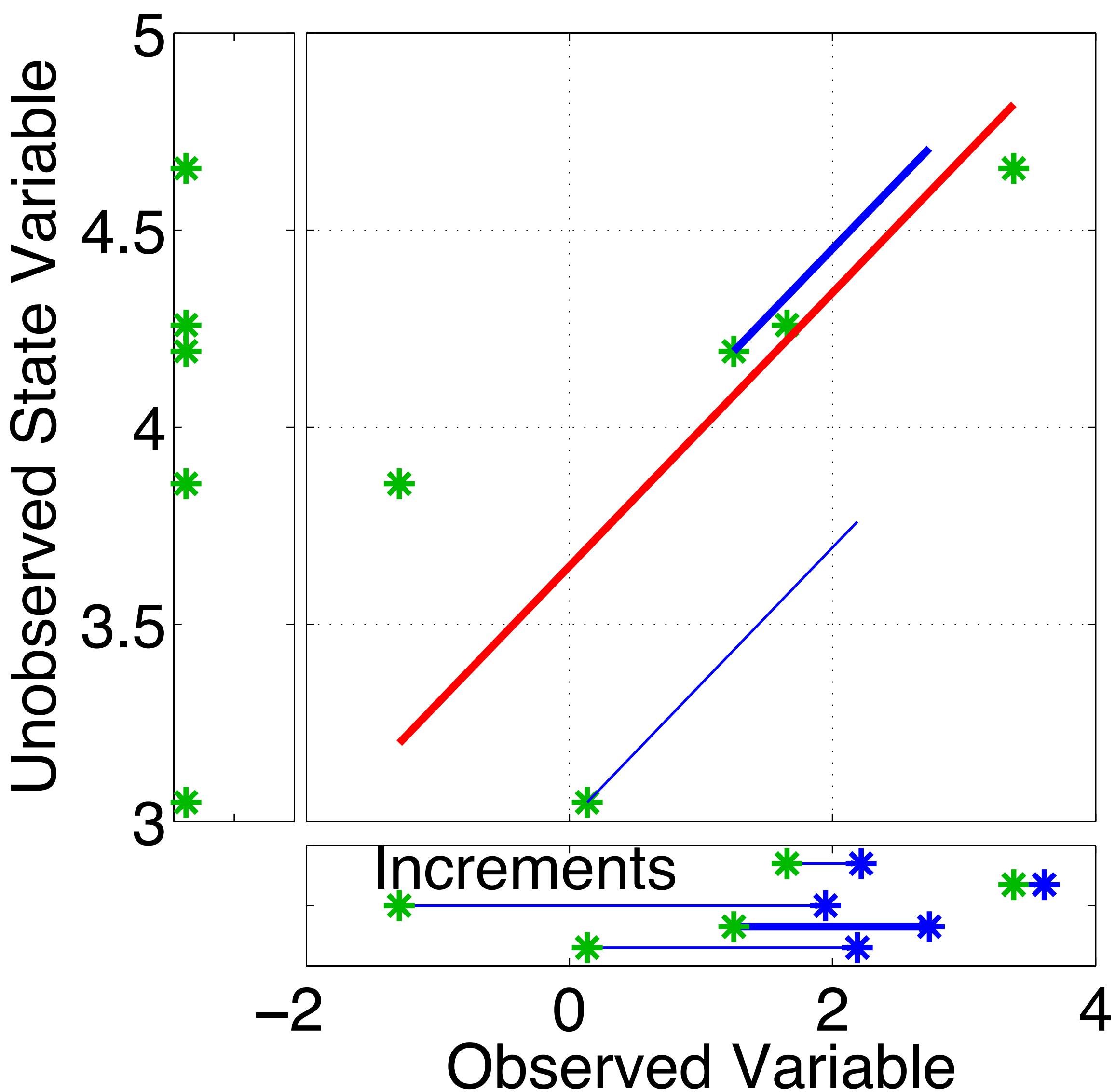


Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in bivariate space.

Ensemble filters: Updating additional prior state variables

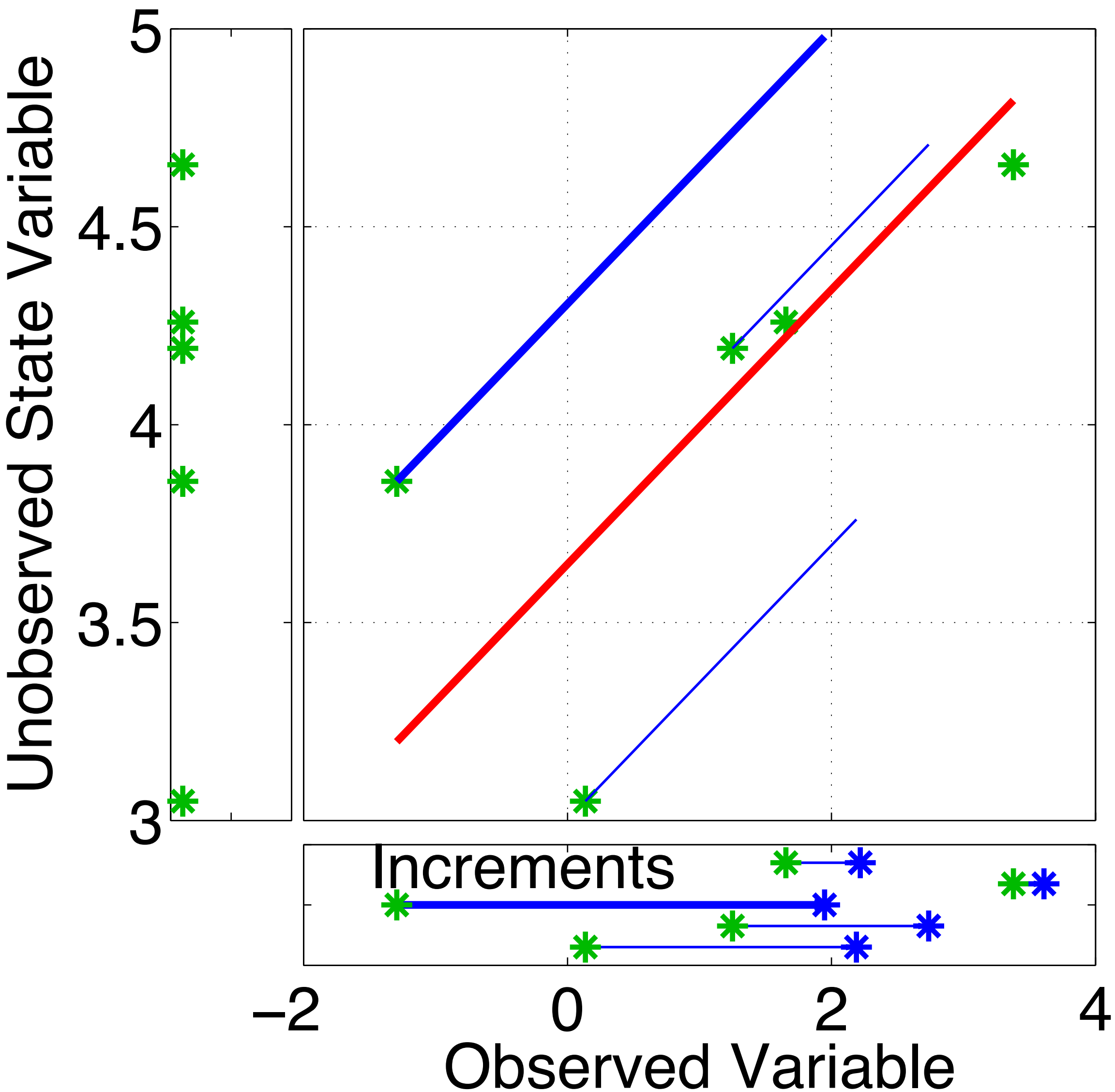


Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in bivariate space.

Ensemble filters: Updating additional prior state variables

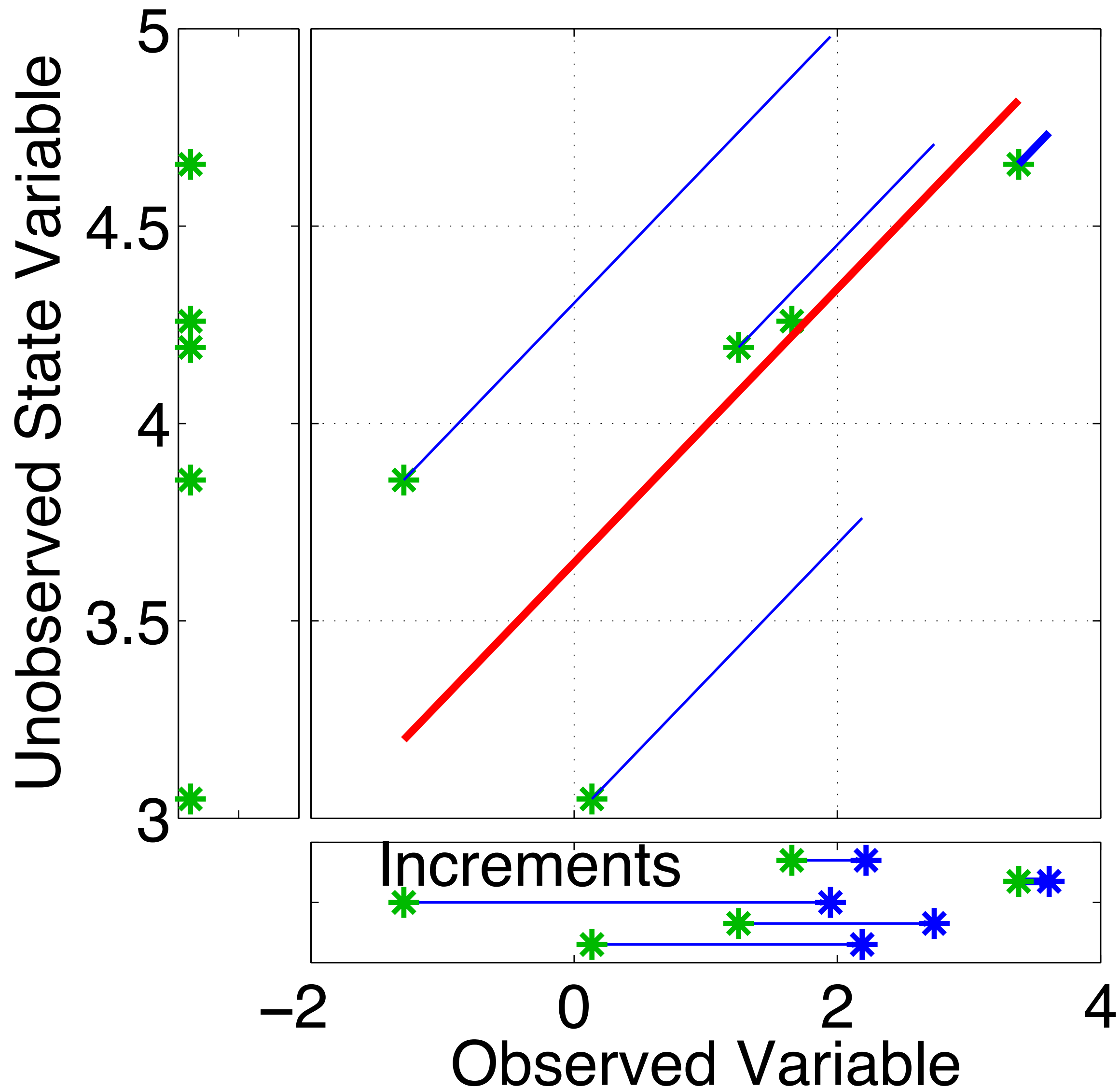


Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in bivariate space.

Ensemble filters: Updating additional prior state variables

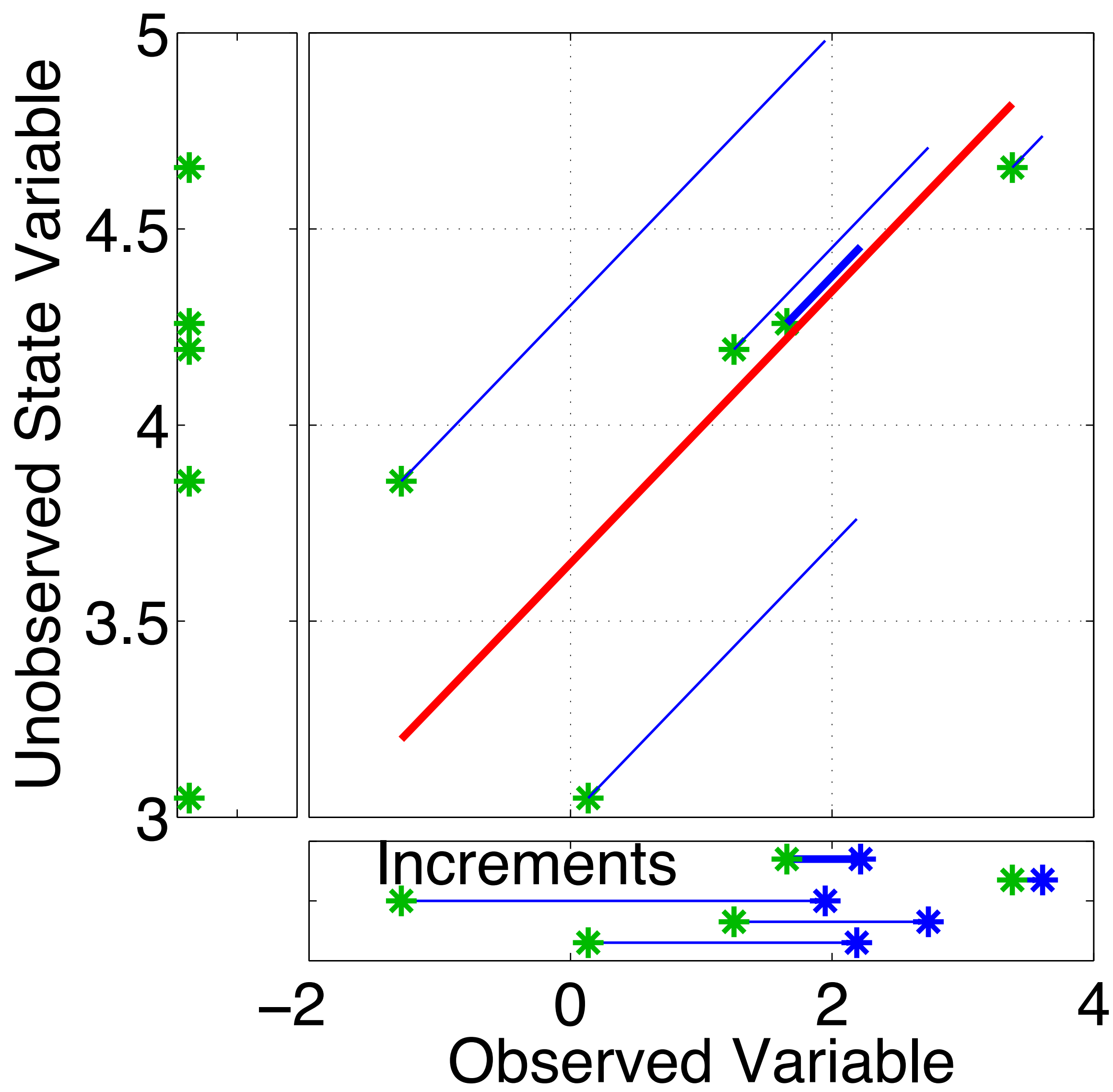


Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in bivariate space.

Ensemble filters: Updating additional prior state variables

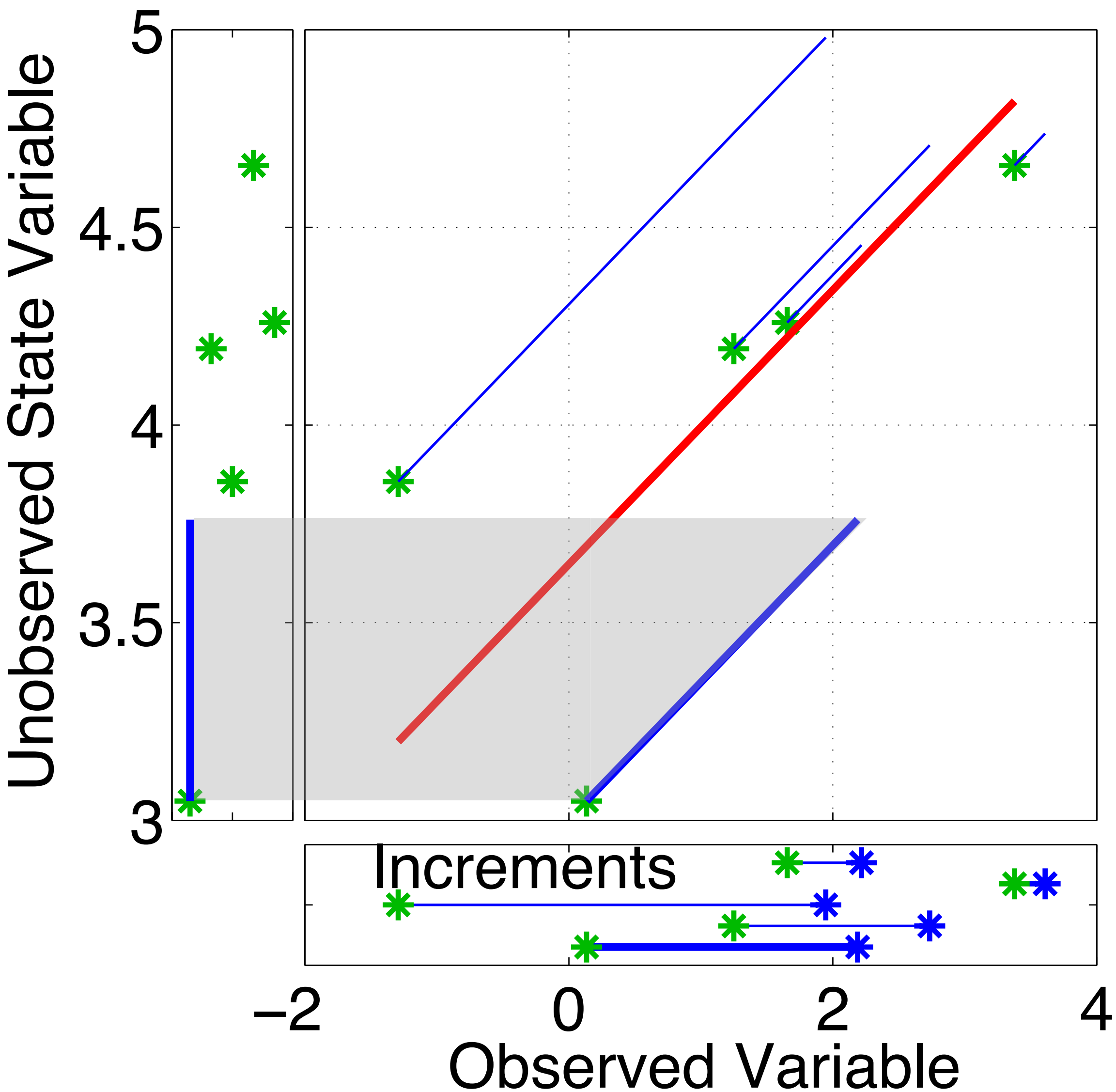


Have joint prior distribution of two variables.

Next, regress the observed variable increments onto increments for the unobserved variable.

Equivalent to first finding image of increment in bivariate space.

Ensemble filters: Updating additional prior state variables

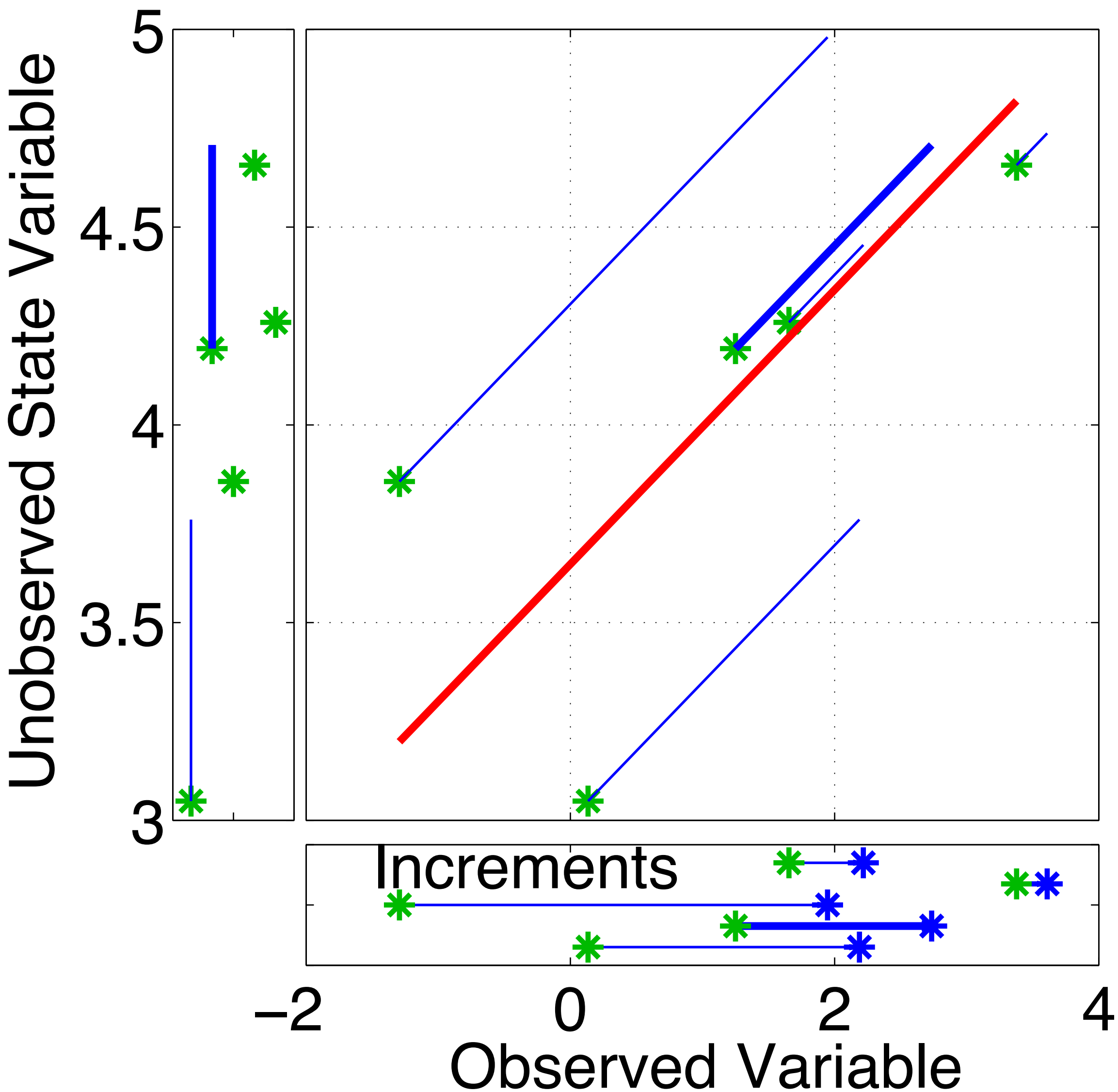


Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in bivariate space.

Then projecting from bivariate space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

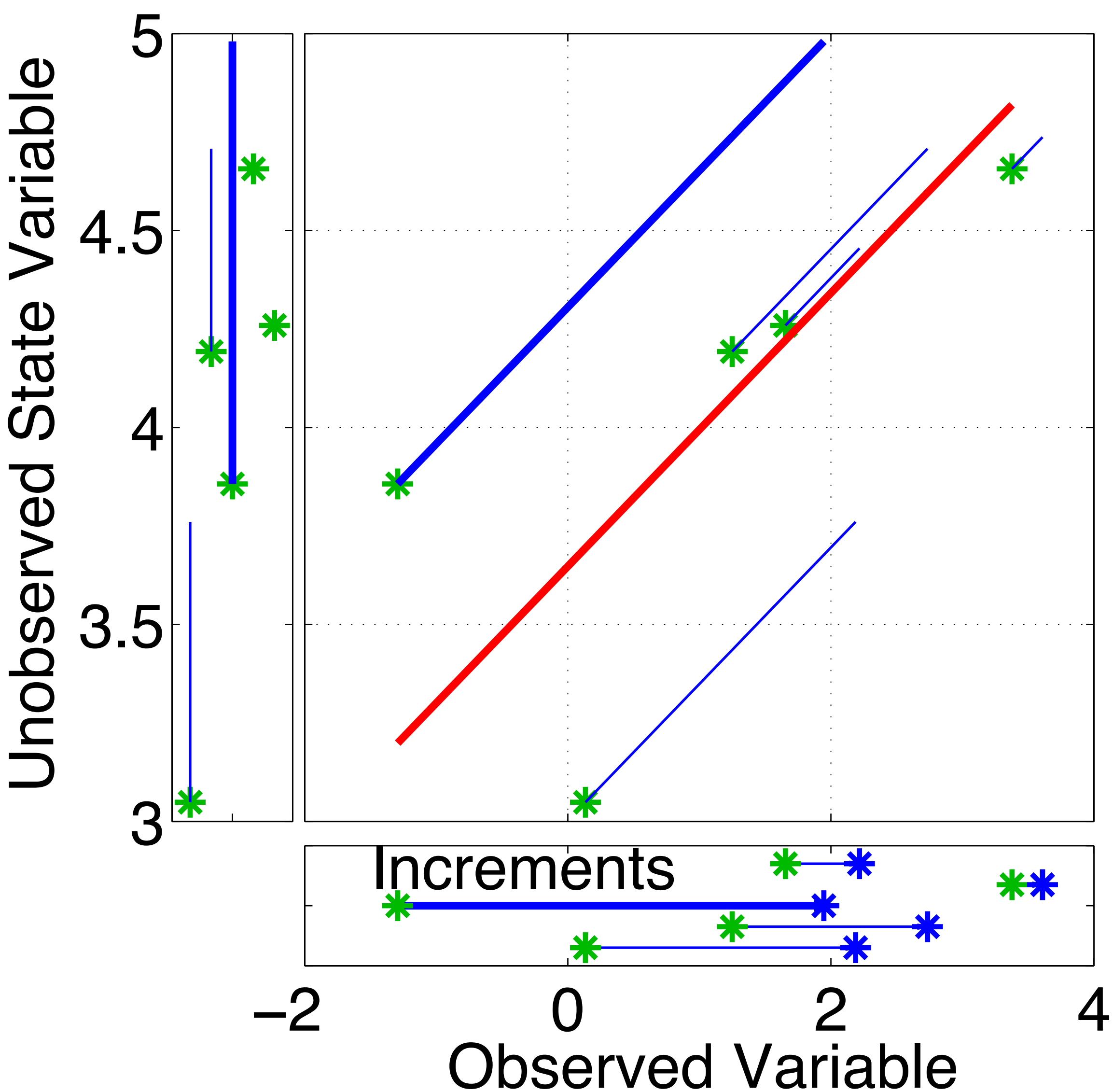


Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in bivariate space.

Then projecting from bivariate space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

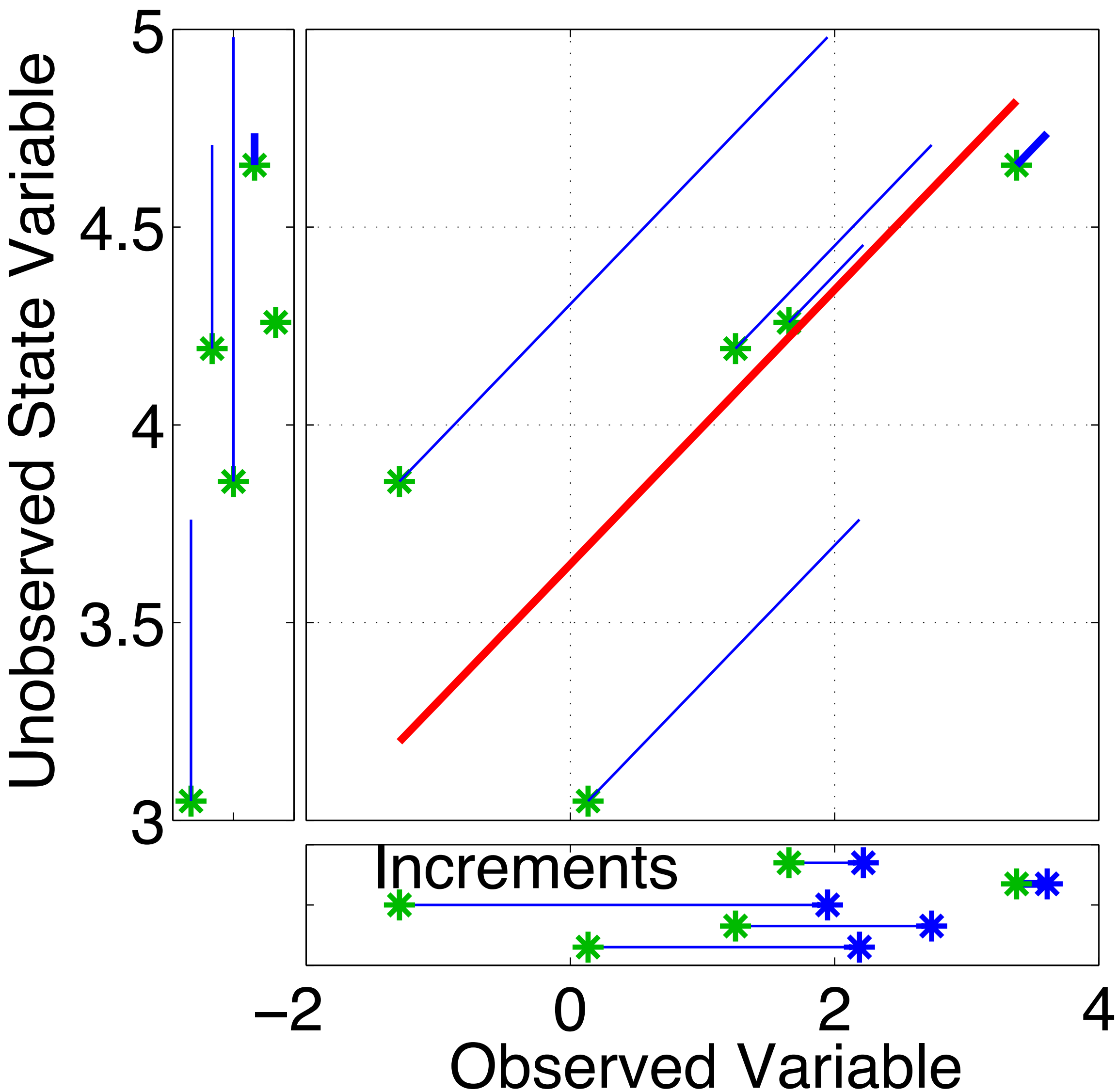


Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in bivariate space.

Then projecting from bivariate space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

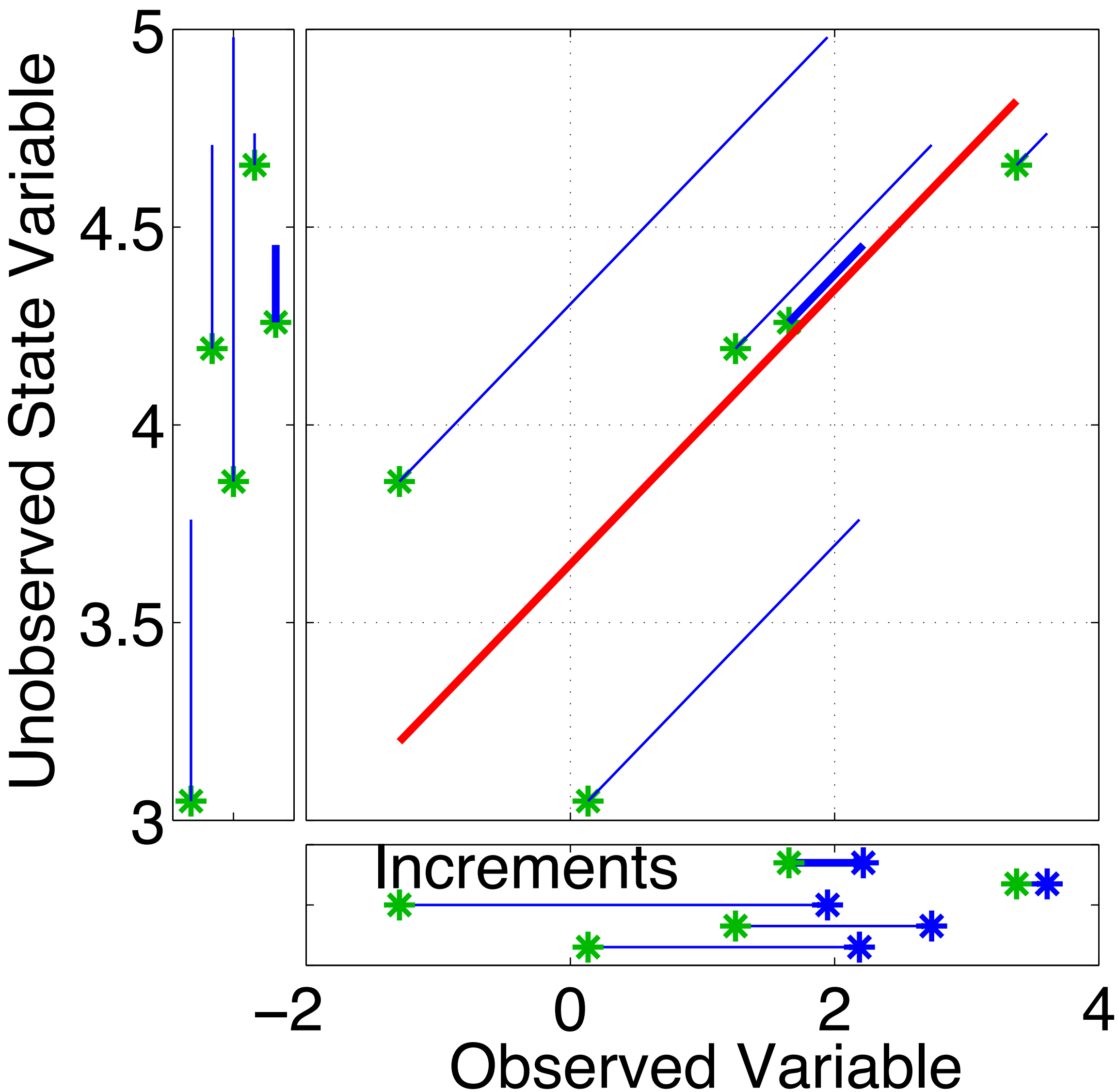


Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in bivariate space.

Then projecting from bivariate space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

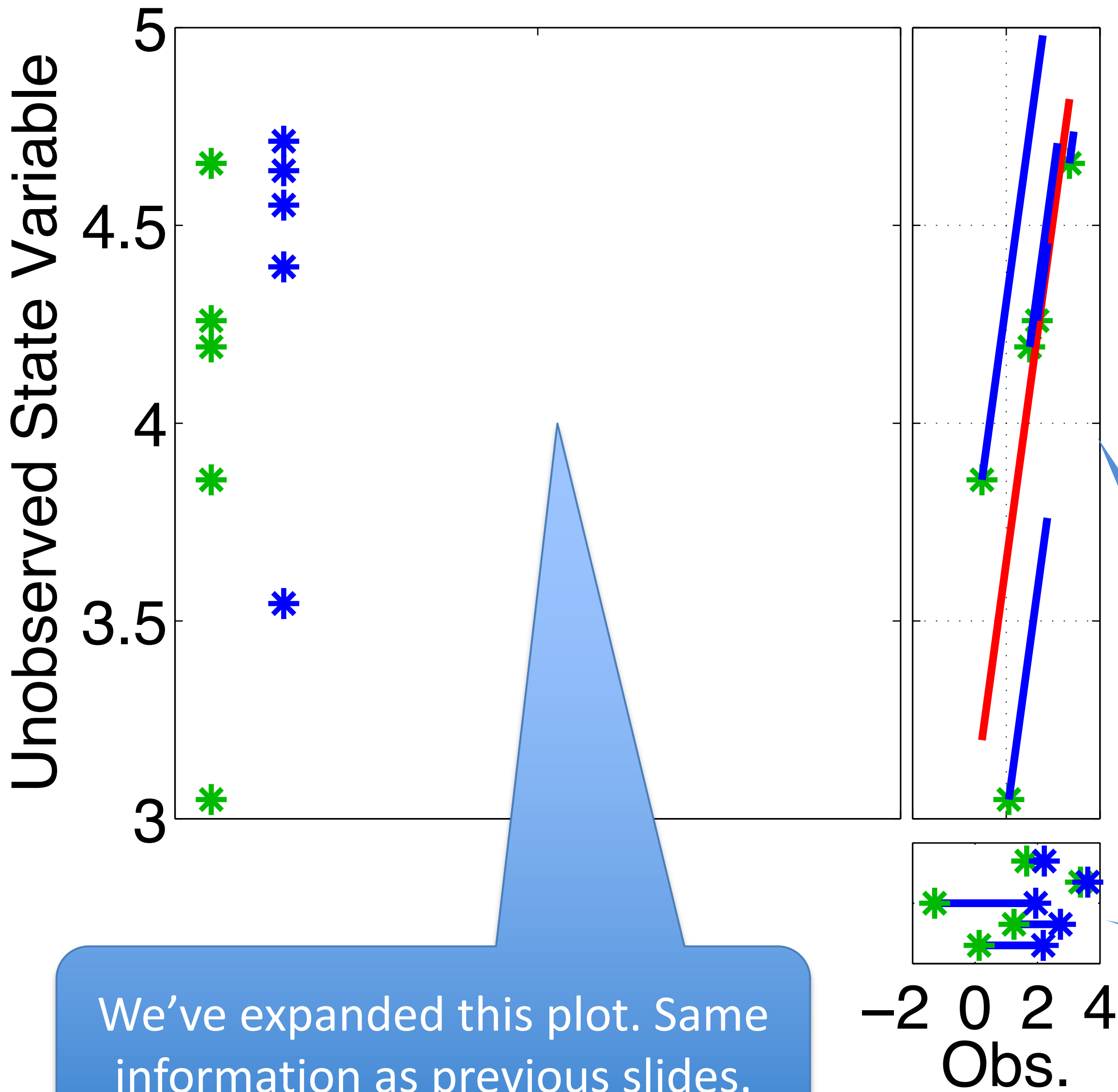


Have joint prior distribution of two variables.

Regression: Equivalent to first finding image of increment in bivariate space.

Then projecting from bivariate space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

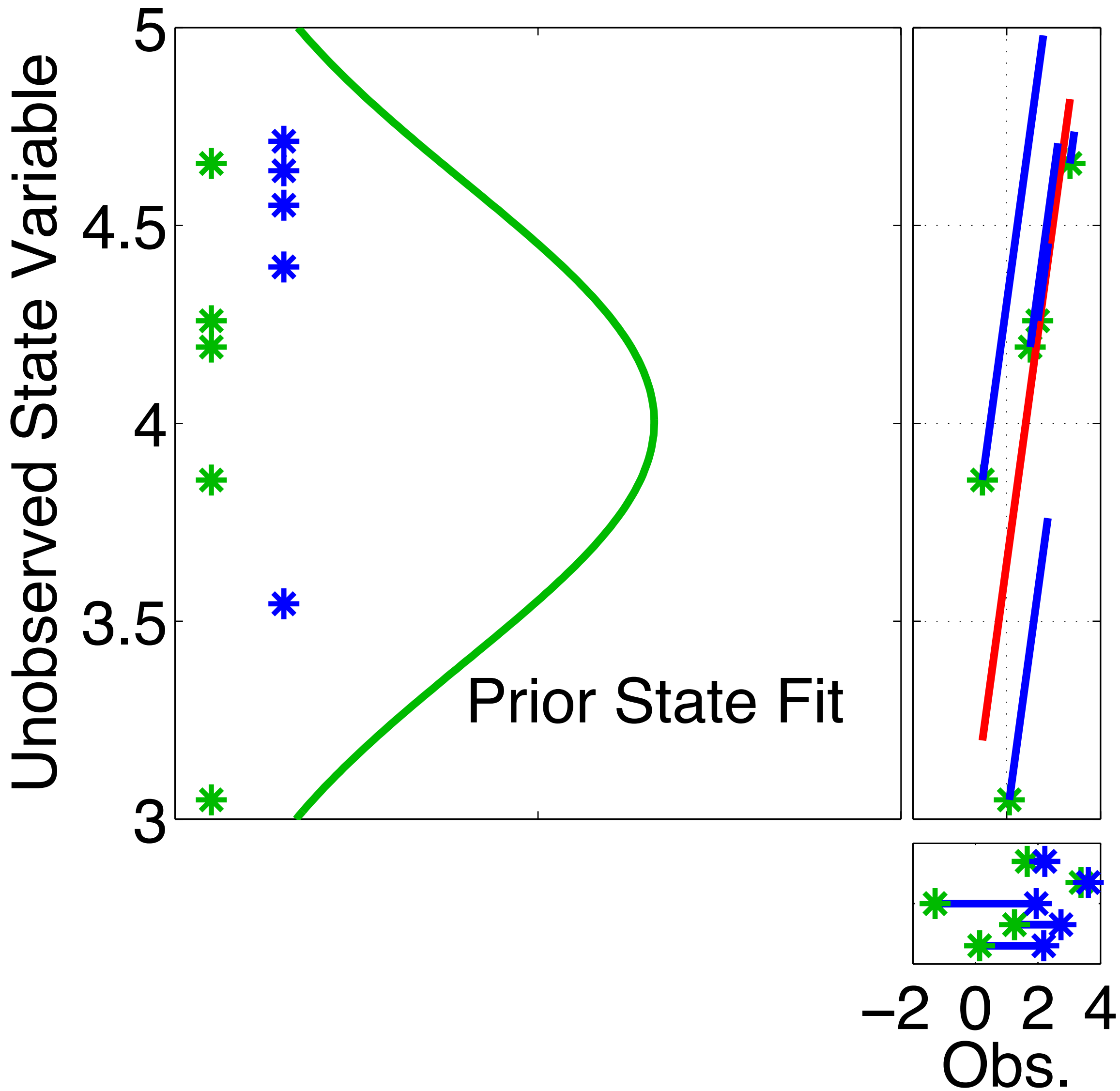


Now have an updated (posterior) ensemble for the unobserved variable.

We've expanded this plot. Same information as previous slides.

Compressed these two.

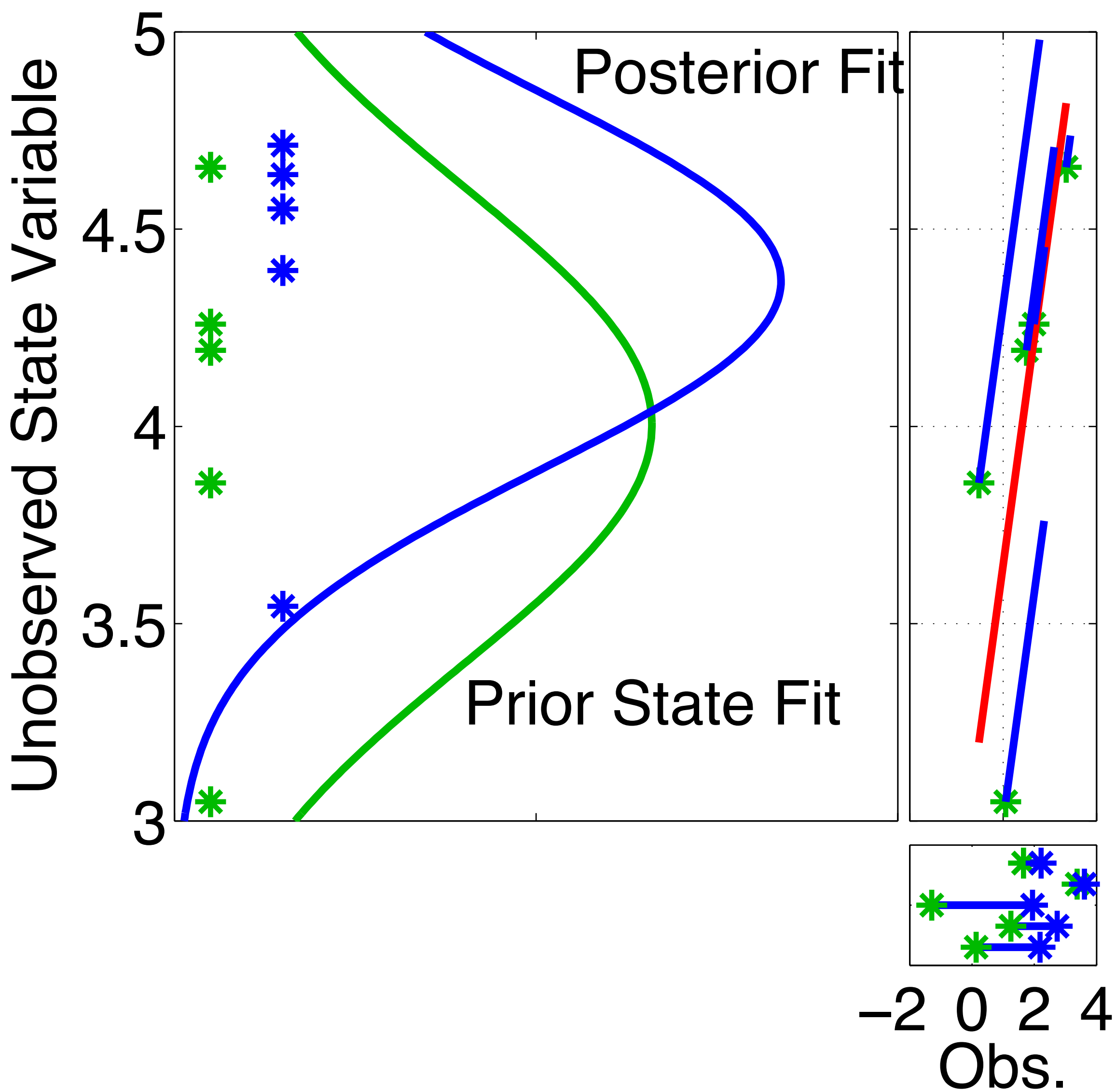
Ensemble filters: Updating additional prior state variables



Now have an updated (posterior) ensemble for the unobserved variable.

Fitting Gaussians shows that mean and variance have changed.

Ensemble filters: Updating additional prior state variables

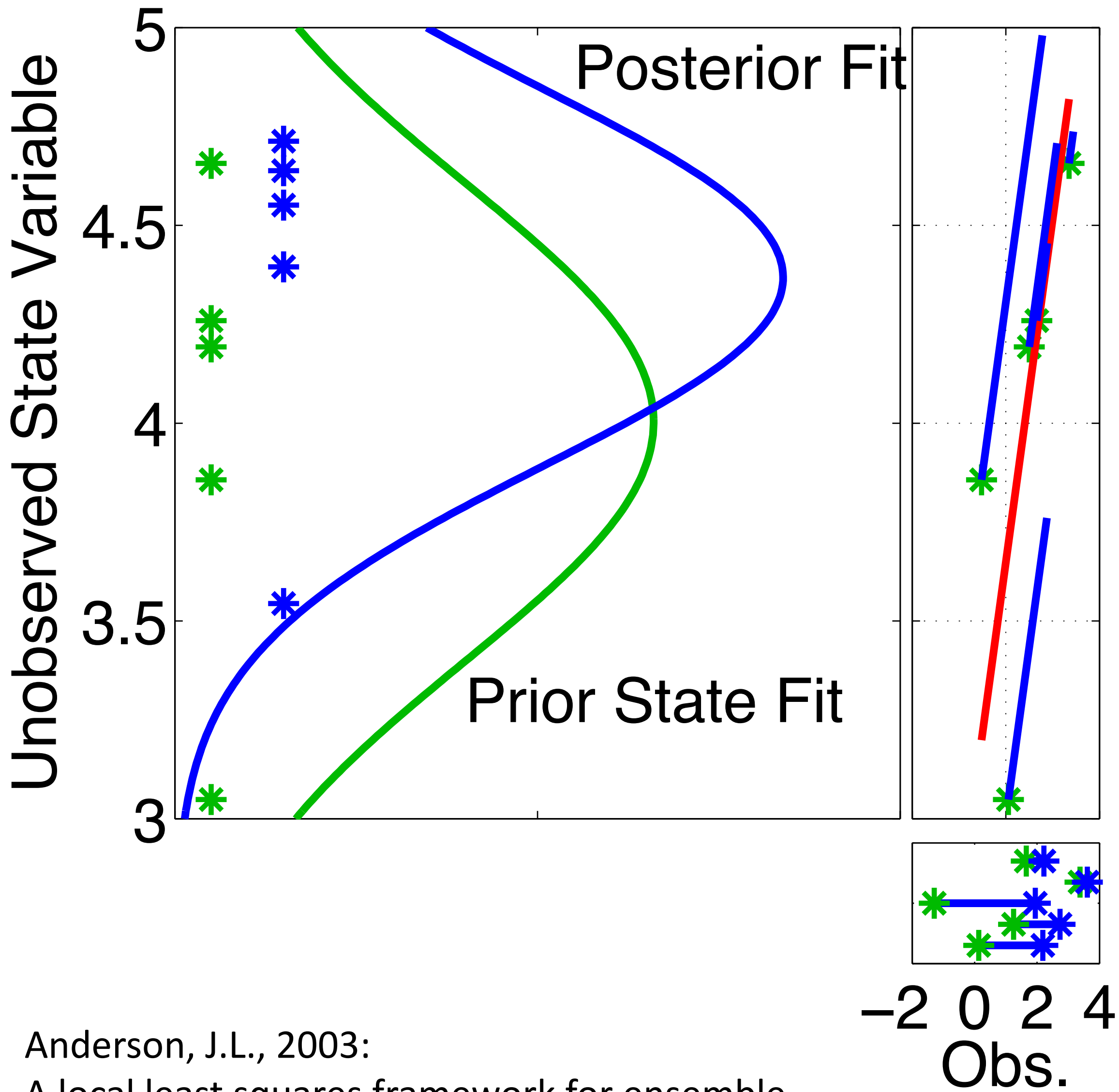


Now have an updated (posterior) ensemble for the unobserved variable.

Fitting Gaussians shows that mean and variance have changed.

Other features of the prior distribution may also have changed.

Ensemble filters: Updating additional prior state variables



CRITICAL POINT:

Since impact on unobserved variable is simply a linear regression, can do this INDEPENDENTLY for any number of unobserved variables!

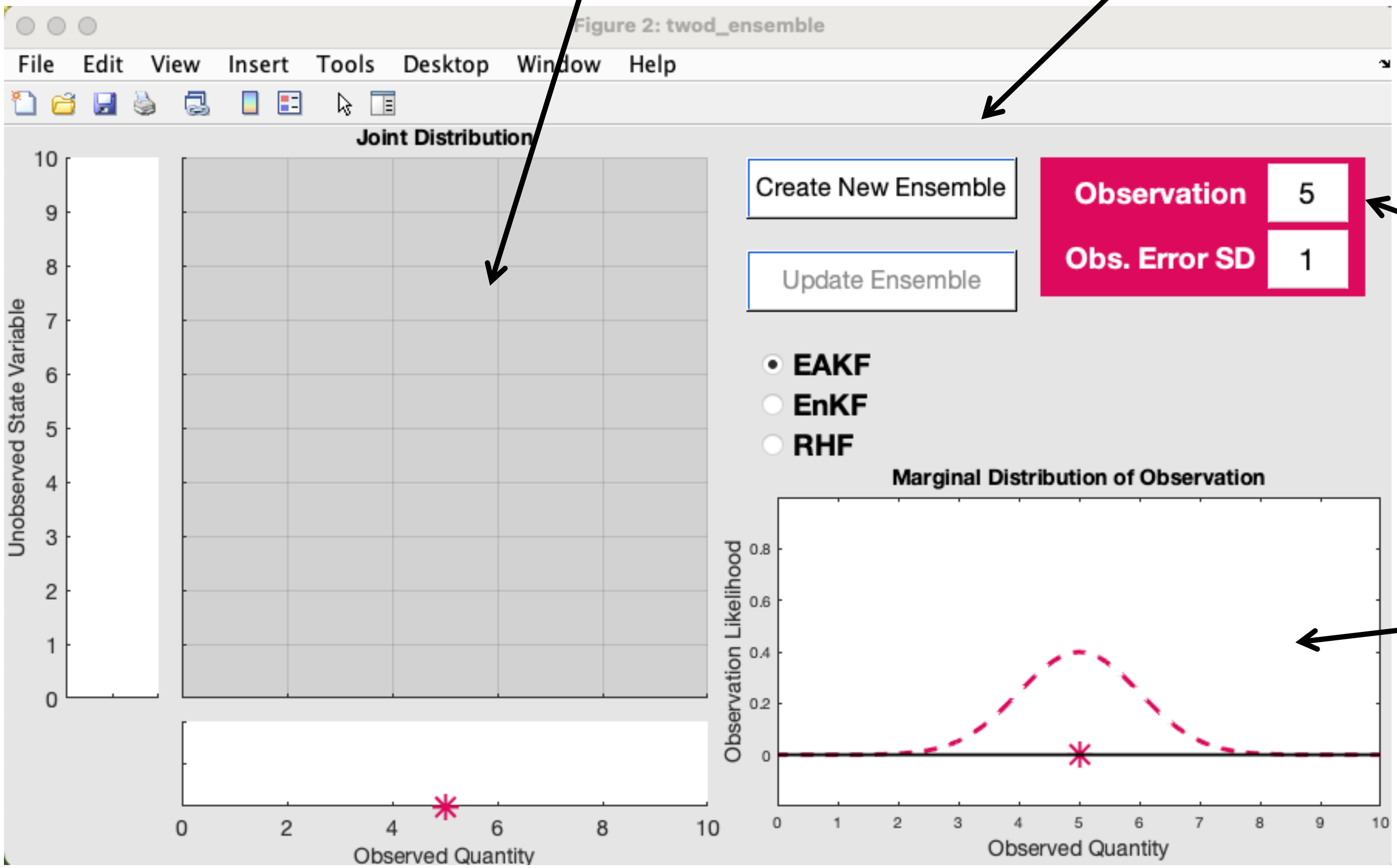
Could also do many at once using matrix algebra as in traditional Kalman Filter.

Anderson, J.L., 2003:
A local least squares framework for ensemble filtering. *Mon. Wea. Rev.*, **131**, 634-642

Matlab Hands-On: twod_ensemble

Bivariate ensemble plot with projected marginals for observed, unobserved variables.

Start creating an ensemble. See next slide.



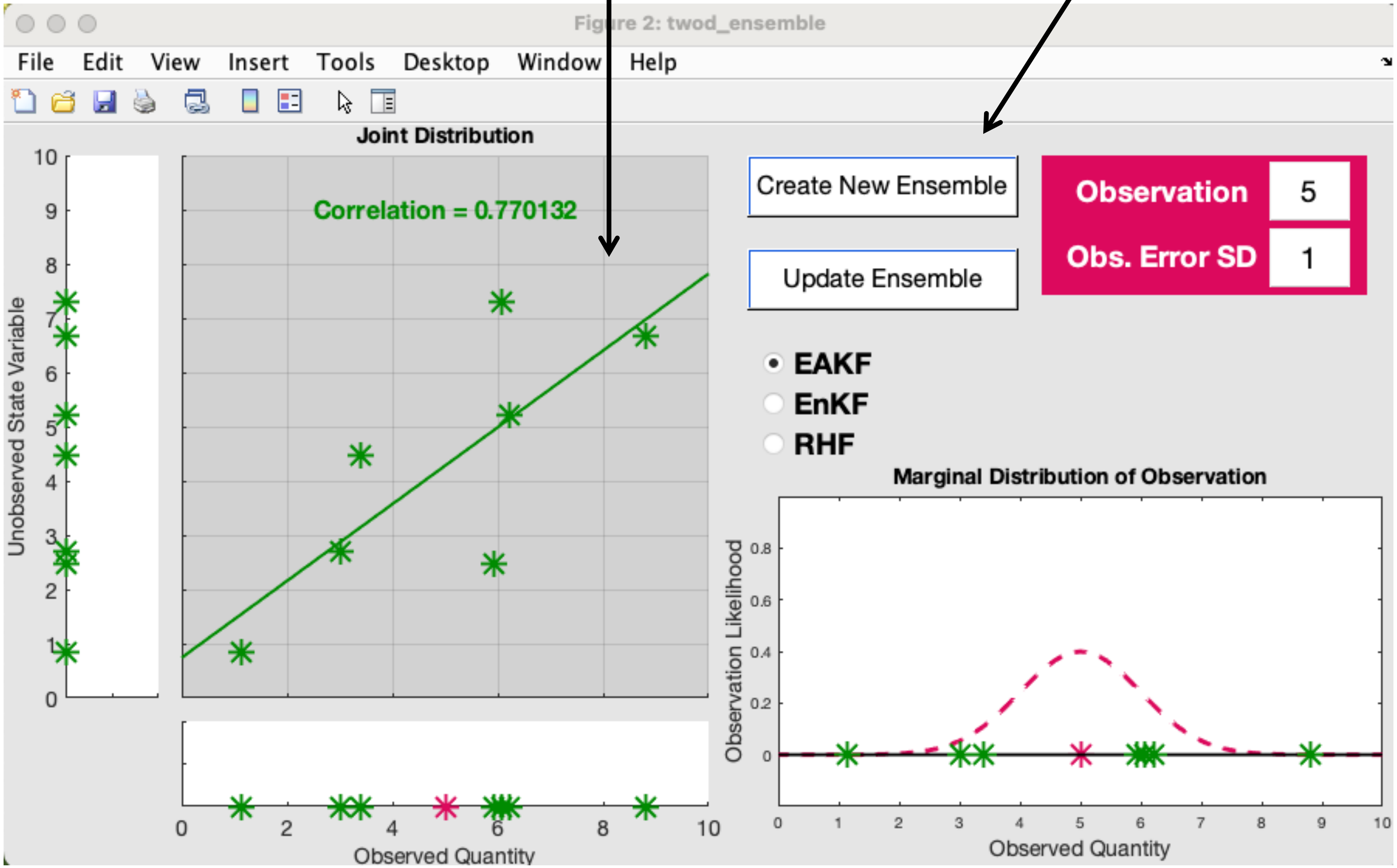
Control observation value and error; same as oned_ensemble.

Detailed plot for observed variable; same as oned_ensemble.

Matlab Hands-On: twod_ensemble

Move cursor and click in this frame to create ensemble members. Click outside this frame when all members are created.

Start creating an ensemble.



Matlab Hands-On: twod_ensemble

Explorations:

- Create ensemble members that are nearly on a line. Explore how the unobserved variable is updated.
- What happens for nearly uncorrelated observed and unobserved variables? Create a roundish cloud of points for the prior.
- What happens with a two-dimensional bimodal distribution?
- Try prior ensembles with various types of outliers.

Summary of Key Points so Far

We know how to:

1. Assimilate a single observation of a single state variable with normal distributions.
2. Cyclically assimilate multiple observations at the same time if their error distributions are independent.
3. Do cycled DA for a single variable and single observation with normal distributions (Kalman Filter).
4. Duplicate cycled DA results using an ensemble of model forecasts for the prior and fitting a normal to the ensemble prior (Ensemble Adjustment Kalman Filter).
5. Update any number of additional variables given an observation using ensemble increment regression.

Combining these things, we have a framework for doing cycled ensemble DA with any number of 'identity' observations of model state variables at each time.

Schematic of an Ensemble Filter for Geophysical Data Assimilation

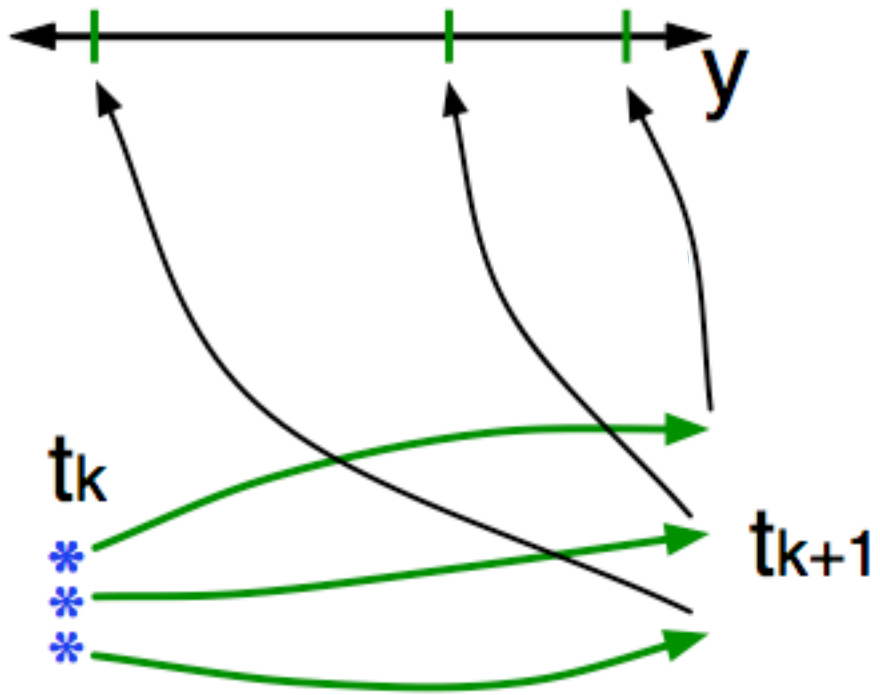
1. Use model to advance ensemble (3 members here) to time at which next observation(s) becomes available.

Ensemble state estimate after using previous observation (analysis)

Ensemble state at time of next observation (prior)

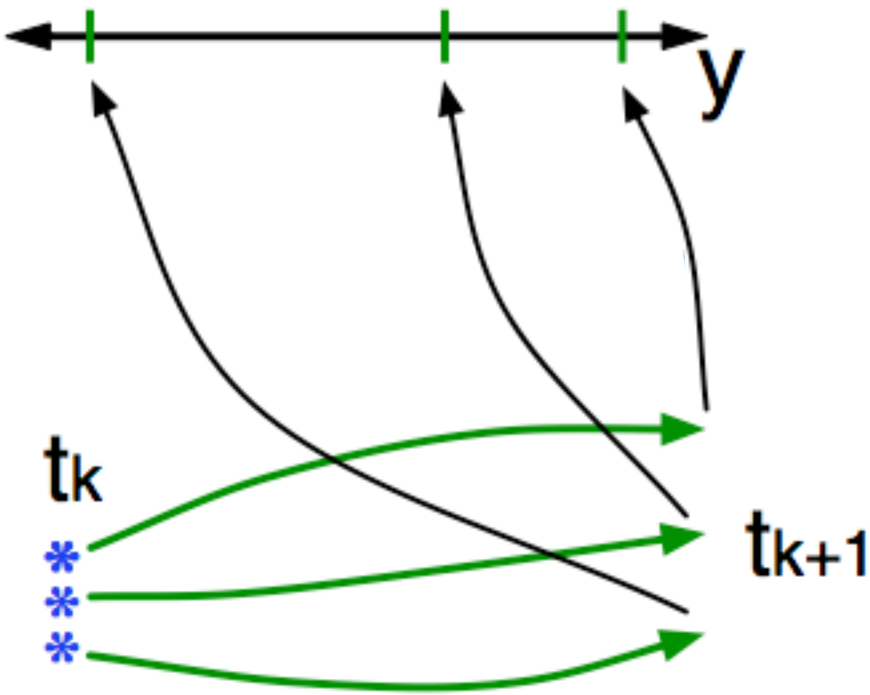
Schematic of an Ensemble Filter for Geophysical Data Assimilation

2. Get the ensemble of values of the first observation to be assimilated at this time (observation is of a state variable).



Schematic of an Ensemble Filter for Geophysical Data Assimilation

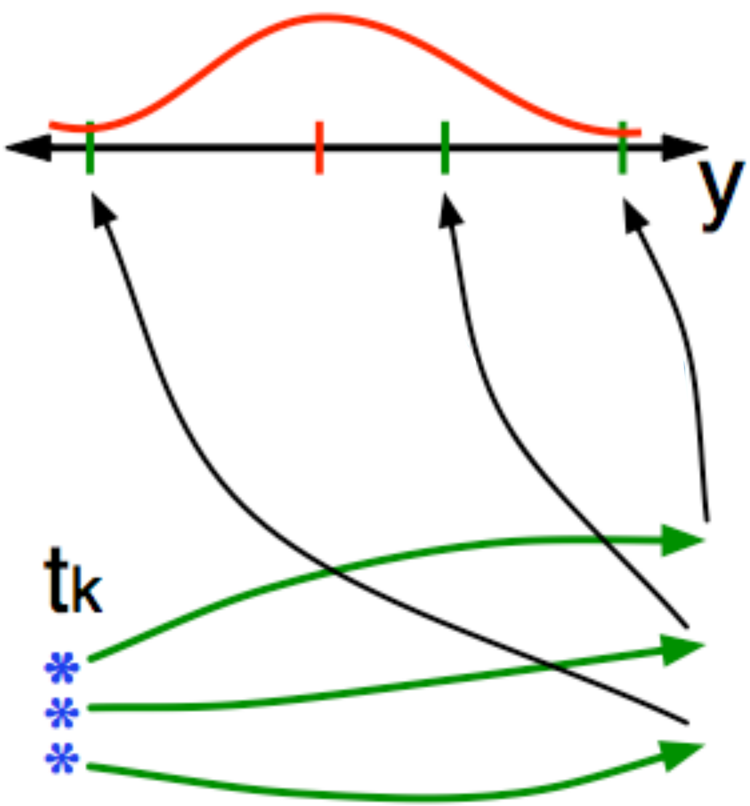
2. Get the ensemble of values of the first observation to be assimilated at this time (observation is of a state variable).



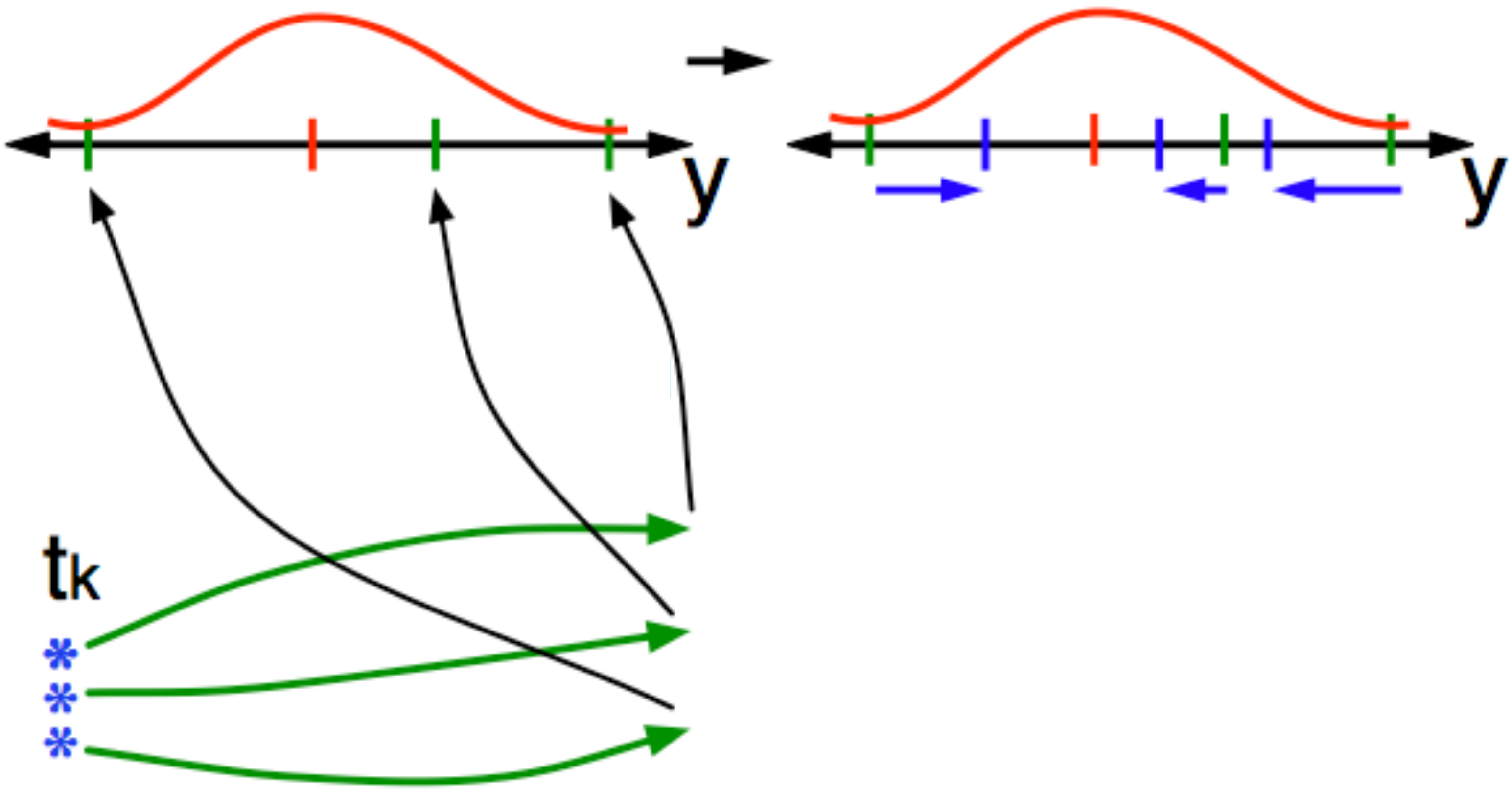
Theory: observations from instruments with uncorrelated errors can be done sequentially.

Houtekamer, P.L. and H.L. Mitchell, 2001: A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.*, **129**, 123-137

3. Get observed value and likelihood from observing system.

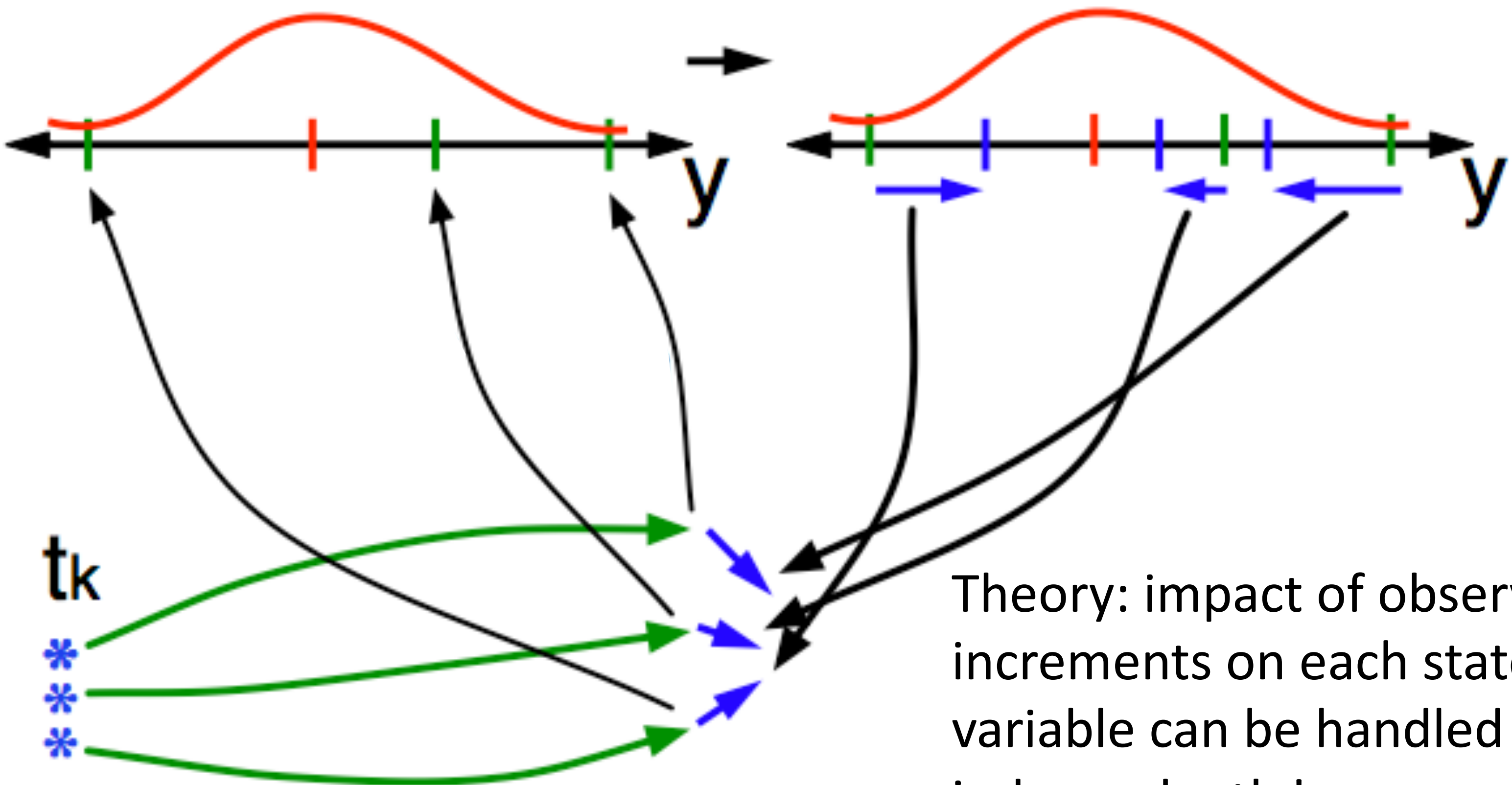


4. Find the increments for the prior observation ensemble, this is a scalar problem.



Schematic of an Ensemble Filter for Geophysical Data Assimilation

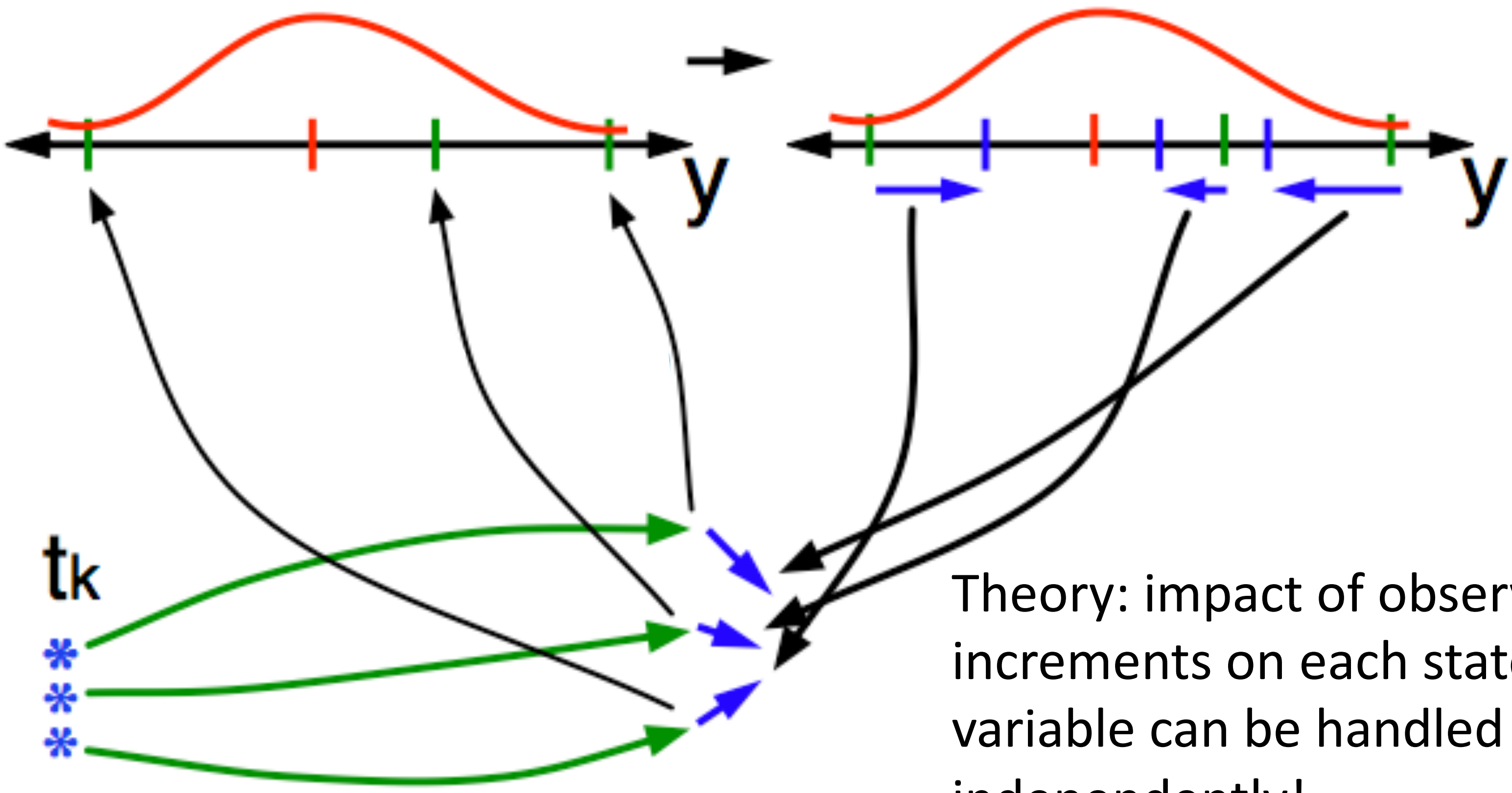
5. Use ensemble samples of y and each state variable to linearly regress observation increments onto state variable increments.



Theory: impact of observation increments on each state variable can be handled independently!

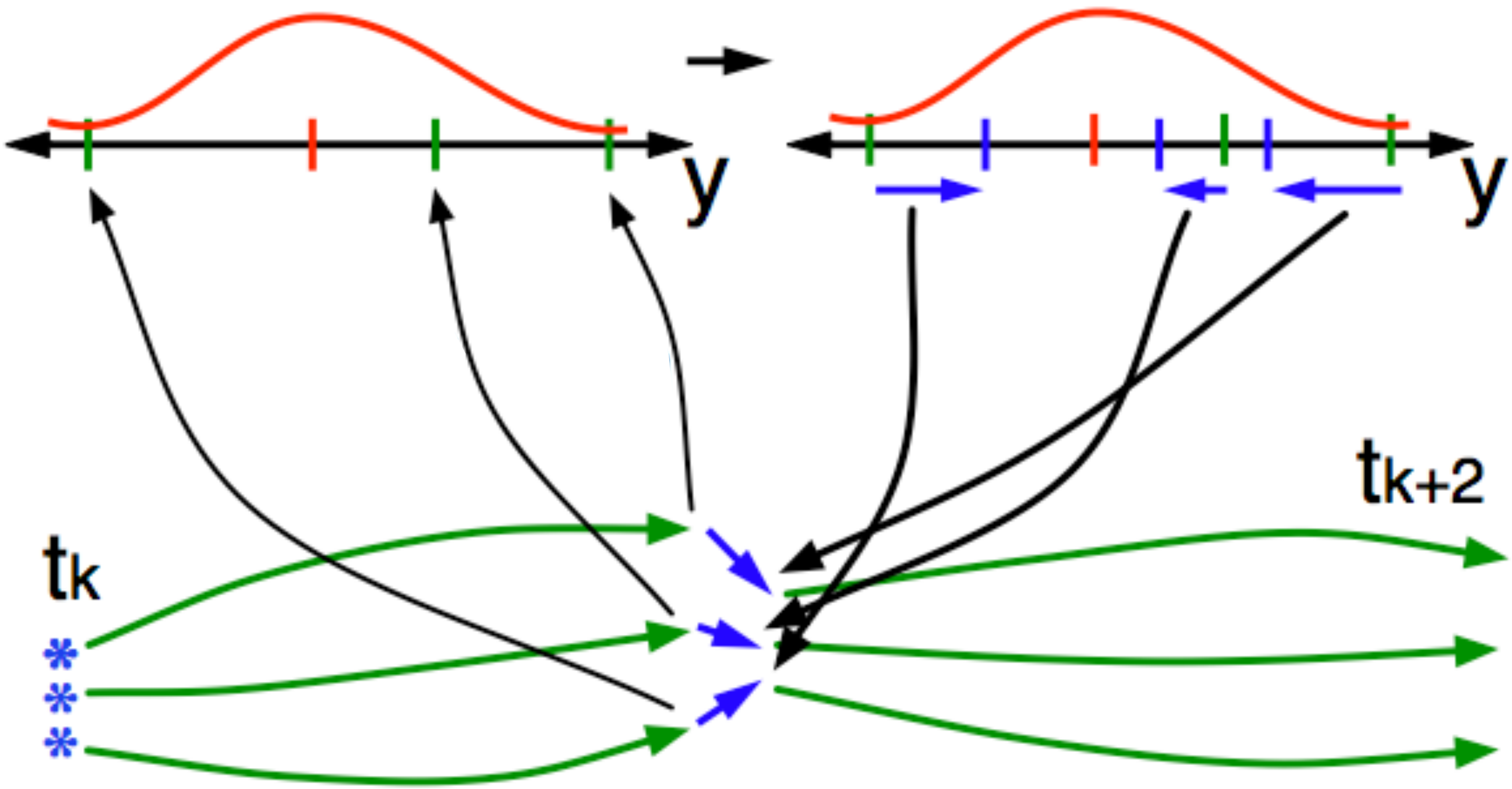
Schematic of an Ensemble Filter for Geophysical Data Assimilation

Repeat steps 2-5 sequentially for each observation at this time.

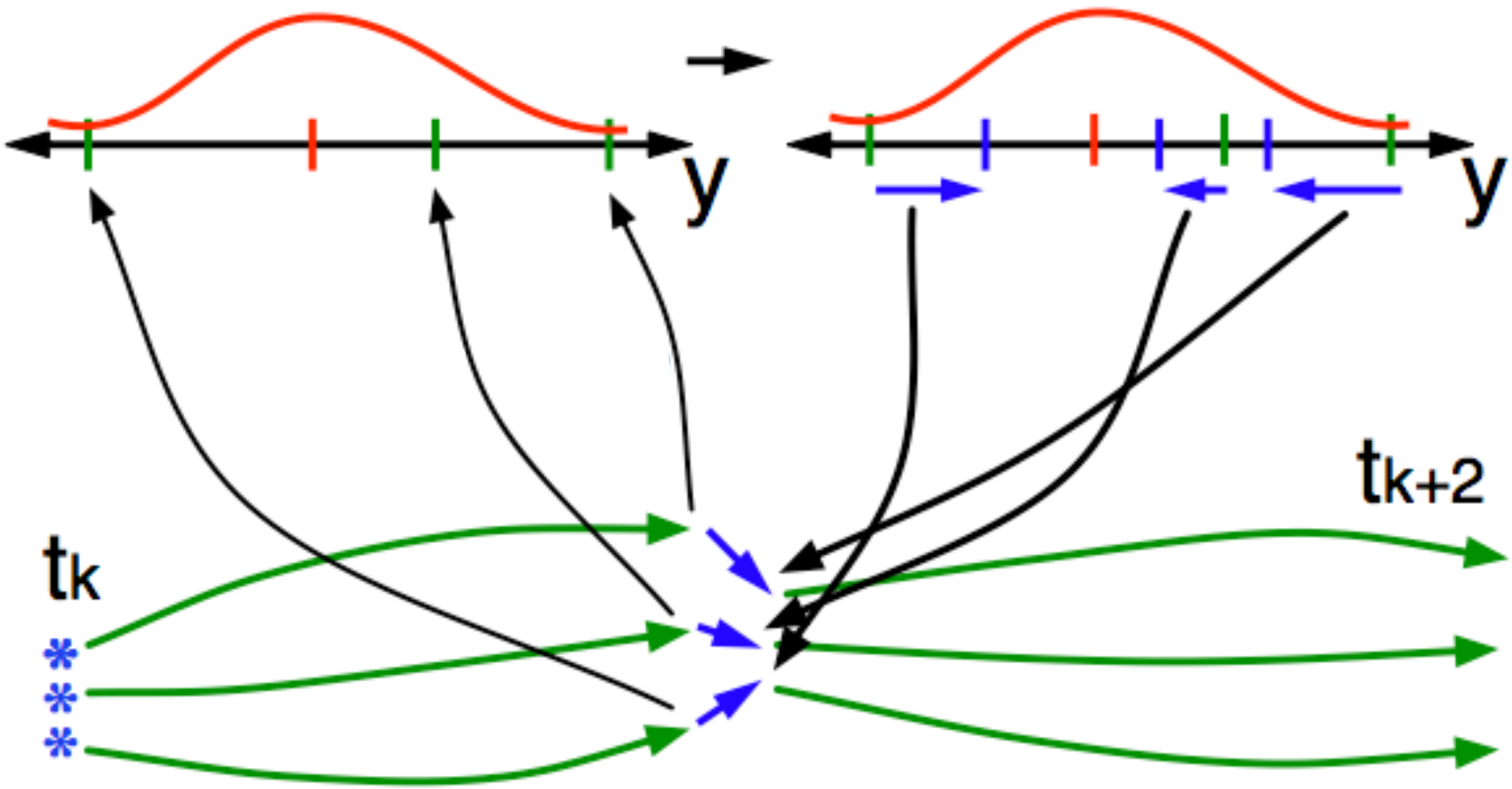


Theory: impact of observation increments on each state variable can be handled independently!

6. When all observations at this time are assimilated, integrate model state to the next time with observations.



Reminder: This schematic has assumed that observations are 'identity' observations of model state variables.



Data Assimilation: A somewhat general description

A time-varying state-vector $\mathbf{x}_t$,

Data Assimilation: A somewhat general description

A time-varying state-vector $\mathbf{x}_t$,

Times t_k with observations: $k = 1, 2, \ldots;\quad t_{k+1} > t_k \geq t_0$,

Data Assimilation: A somewhat general description

A time-varying state-vector $\mathbf{x}_t$,

Times t_k with observations: $k = 1, 2, \ldots;\quad t_{k+1} > t_k \geq t_0$,

Observations at t_k related to $\mathbf{x}_{t_k}$; $\mathbf{y}_k = h_k(\mathbf{x}_{t_k}) + \nu_k$, (1)

Observations

Forward Operator

Observation Error

Data Assimilation: A somewhat general description

A time-varying state-vector $\mathbf{x}_t$,

Times t_k with observations: $k = 1, 2, \ldots;$ $t_{k+1} > t_k \geq t_0$,

Observations at t_k related to $\mathbf{x}_{t_k}$; $\mathbf{y}_k = h_k(\mathbf{x}_{t_k}) + \nu_k$, (1)

Observation error is zero mean, normal, $\nu_k = N(0, \mathbf{R}_k)$, (2)

Observation Error Covariance

We assume $\mathbf{R}_k$ is diagonal (observation errors are uncorrelated for different observations) for the rest of the tutorial.

Data Assimilation: A somewhat general description

A time-varying state-vector $\mathbf{x}_t$,

Times t_k with observations: $k = 1, 2, \ldots;$ $t_{k+1} > t_k \geq t_0$,

Observations at t_k related to $\mathbf{x}_{t_k}$; $\mathbf{y}_k = h_k(\mathbf{x}_{t_k}) + \nu_k$, (1)

Observation error is zero mean, normal, $\nu_k = N(0, \mathbf{R}_k)$, (2)

A forecast model m for the state-vector; $\mathbf{x}_{t_{k+1}} = m_{k:k+1}(\mathbf{x}_{t_k})$ (3)

Data Assimilation: A somewhat general description

A time-varying state-vector $\mathbf{x}_t$,

Times t_k with observations: $k = 1, 2, \ldots;\quad t_{k+1} > t_k \geq t_0$,

Observations at t_k related to $\mathbf{x}_{t_k}$; $\mathbf{y}_k = h_k(\mathbf{x}_{t_k}) + \nu_k$, (1)

Observation error is zero mean, normal, $\nu_k = N(0, \mathbf{R}_k)$, (2)

A forecast model m for the state-vector; $\mathbf{x}_{t_{k+1}} = m_{k:k+1}(\mathbf{x}_{t_k})$ (3)

m can have deterministic and stochastic parts;

$$m_{k:k+1}(\mathbf{x}_{t_k}) = f_{k:k+1}(\mathbf{x}_{t_k}) + g_{k:k+1}(\mathbf{x}_{t_k}). \tag{4}$$

Data Assimilation: A somewhat general description

Define the set of all observations taken no later than time t_k:

$$\mathbf{Y}_k = \{\mathbf{y}_i; i \leq k\} \qquad (5)$$

Problems of interest are:

Analysis: $P(\mathbf{x}_t|\mathbf{Y}_k), \quad t = t_k$ (6)

Forecast: $P(\mathbf{x}_t|\mathbf{Y}_k), \quad t > t_k$ (7)

Smoother: $P(\mathbf{x}_t|\mathbf{Y}_k), \quad t < t_k$ (8)

Data Assimilation: A somewhat general description

Define the set of all observations taken no later than time t_k:

$$\mathbf{Y}_k = \{\mathbf{y}_i; i \leq k\} \quad (5)$$

Problems of interest are:

Analysis: $P(\mathbf{x}_t|\mathbf{Y}_k), \quad t = t_k$ (6)

Forecast: $P(\mathbf{x}_t|\mathbf{Y}_k), \quad t > t_k$ (7)

Smoother: $P(\mathbf{x}_t|\mathbf{Y}_k), \quad t < t_k$ (8)

Note: could also replace $\mathbf{x}_t$ with any of the other things data assimilation can estimate: parameters, initial conditions, …

Data Assimilation: A somewhat general description

Forecasts of state, $\mathbf{x}$ are obtained from model.

Need to update forecast state given new observations:

$$P\left(\mathbf{x}_{t_k} | \mathbf{Y}_k\right) = \mathrm{P}\left(\mathbf{x}_{t_k} | \mathbf{y}_k, \mathbf{Y}_{k-1}\right)$$

Bayes' rule:

$$P\left(\mathbf{x}_{t_k} | \mathbf{Y}_k\right) = \frac{P\left(\mathbf{y}_k | \mathbf{x}_{t_k}, \mathbf{Y}_{k-1}\right) P\left(\mathbf{x}_{t_k} | \mathbf{Y}_{k-1}\right)}{P\left(\mathbf{y}_k | \mathbf{Y}_{k-1}\right)} \qquad (9)$$

Observation errors uncorrelated in time:

$$P\left(\mathbf{y}_k | \mathbf{x}_{t_k}, \mathbf{Y}_{k-1}\right) = P\left(\mathbf{y}_k | \mathbf{x}_{t_k}\right)$$

Denominator in (9) is normalization, makes update a pdf.

Data Assimilation: A somewhat general description

Probability after new observation:

Prior (forecast)

Likelihood

$$P\left(\mathbf{x}_{t_k} | \mathbf{Y}_k\right) = \frac{P(\mathbf{y}_k | \mathbf{x}) P\left(\mathbf{x}_{t_k} | \mathbf{Y}_{k-1}\right)}{Normalization} \quad (10)$$

Posterior (analysis).

Data Assimilation: A somewhat general description

Probability after new observation:

Prior (forecast)

Likelihood

$$P\left(\mathbf{x}_{t_k}|\mathbf{Y}_k\right) = \frac{P(\mathbf{y}_k|\mathbf{x})P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{k-1}\right)}{Normalization} \qquad (10)$$

Posterior (analysis).

Forecasts produced by applying model to analysis.

Smoother can be derived from a similar Bayesian analysis.

Data Assimilation: A revised description

Define extended state vector that combines model state and obs:

$$\hat{\mathbf{x}}_{t_k} \equiv \left[\mathbf{x}_{t_k}, \mathbf{y}_k\right]$$

An extended forecast model $\widehat{m}$;

$$\hat{\mathbf{x}}_{t_{k+1}} = \widehat{m}_{k:k+1}\left(\mathbf{x}_{t_k}\right) \equiv \left[\mathbf{x}_{t_{k+1}}, h_{k+1}\left(\mathbf{x}_{t_{k+1}}\right)\right] =$$

$$\left[m_{k:k+1}\left(\mathbf{x}_{t_k}\right), h_{k+1}\left(m_{k:k+1}\left(\mathbf{x}_{t_k}\right)\right)\right]$$

Observations at t_k related to $\hat{\mathbf{x}}_{t_k}$ by 'identity' forward operator;

$$\mathbf{y}_k = \widehat{H}_k\left(\hat{\mathbf{x}}_{t_k}\right) + \nu_k,$$

$\widehat{H}_k$ has row for each observation, column for each extended state element.

Data Assimilation: A revised description

Observations at t_k related to $\hat{\mathbf{x}}_{t_k}$ by 'identity' forward operator;

$$\mathbf{y}_k = \hat{H}_k\left(\hat{\mathbf{x}}_{t_k}\right) + \nu_k,$$

$\hat{H}_k$ is linear, so it can be represented by a matrix.

Has a row for each observation, a column for each extended state element.

All zeros except a diagonal of 1's in the last number of obs columns.

This example has a 5-element model state vector and 4 observations

$$\hat{H}_k = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{matrix}$$

Data Assimilation: A revised description

A time-varying extended state-vector $\hat{\mathbf{x}}_t$,

Times t_k with observations: $k = 1, 2, \ldots;\quad t_{k+1} > t_k \geq t_0$,

Observations at t_k related to $\hat{\mathbf{x}}_{t_k}$; $\mathbf{y}_k = \widehat{H}_k\left(\hat{\mathbf{x}}_{t_k}\right) + \nu_k$, (1)

Observation error is zero mean, normal, $\nu_k = N(0, \mathbf{R}_k)$, (2)

A forecast model $\widehat{m}$ for the state-vector; $\hat{\mathbf{x}}_{t_{k+1}} = \widehat{m}_{k:k+1}\left(\mathbf{x}_{t_k}\right)$ (3)

$\widehat{m}$ can have deterministic and stochastic parts;

$$\widehat{m}_{k:k+1}\left(\mathbf{x}_{t_k}\right) = \hat{f}_{k:k+1}\left(\mathbf{x}_{t_k}\right) + \hat{g}_{k:k+1}\left(\mathbf{x}_{t_k}\right). \quad (4)$$

Schematic of an Ensemble Filter for Geophysical Data Assimilation (This is implemented in DART Fortran code)

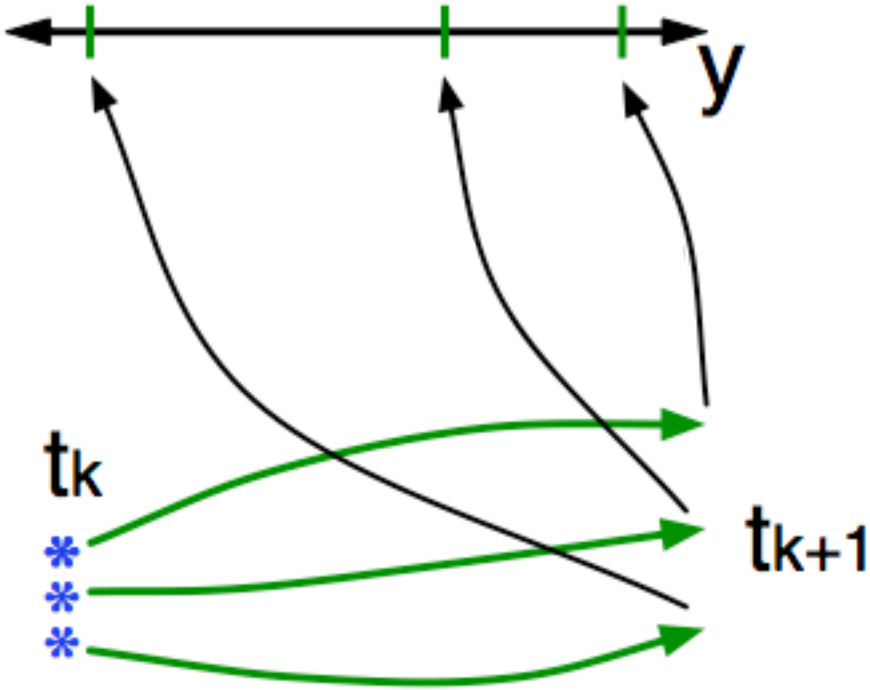
1. Use model to advance ensemble (3 members here) to time at which next observation(s) becomes available, and compute forward operators for all observations for each ensemble.

Ensemble state estimate after using previous observation (analysis)

Extended ensemble state at time of next observation (prior)

Schematic of an Ensemble Filter for Geophysical Data Assimilation (This is implemented in DART Fortran code)

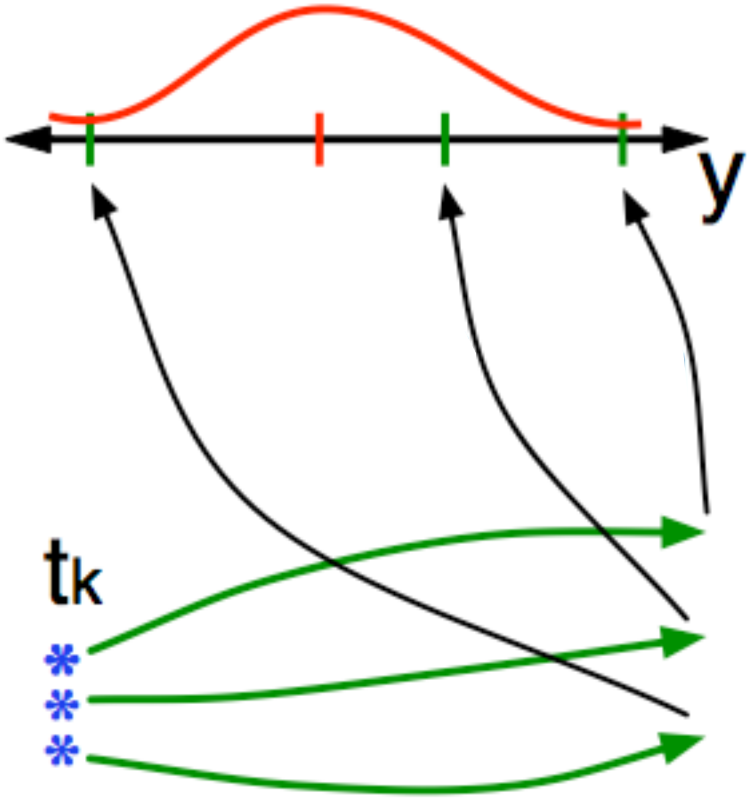
2. Get the ensemble of values of the first observation to be assimilated at this time (observation is of an extended state variable).



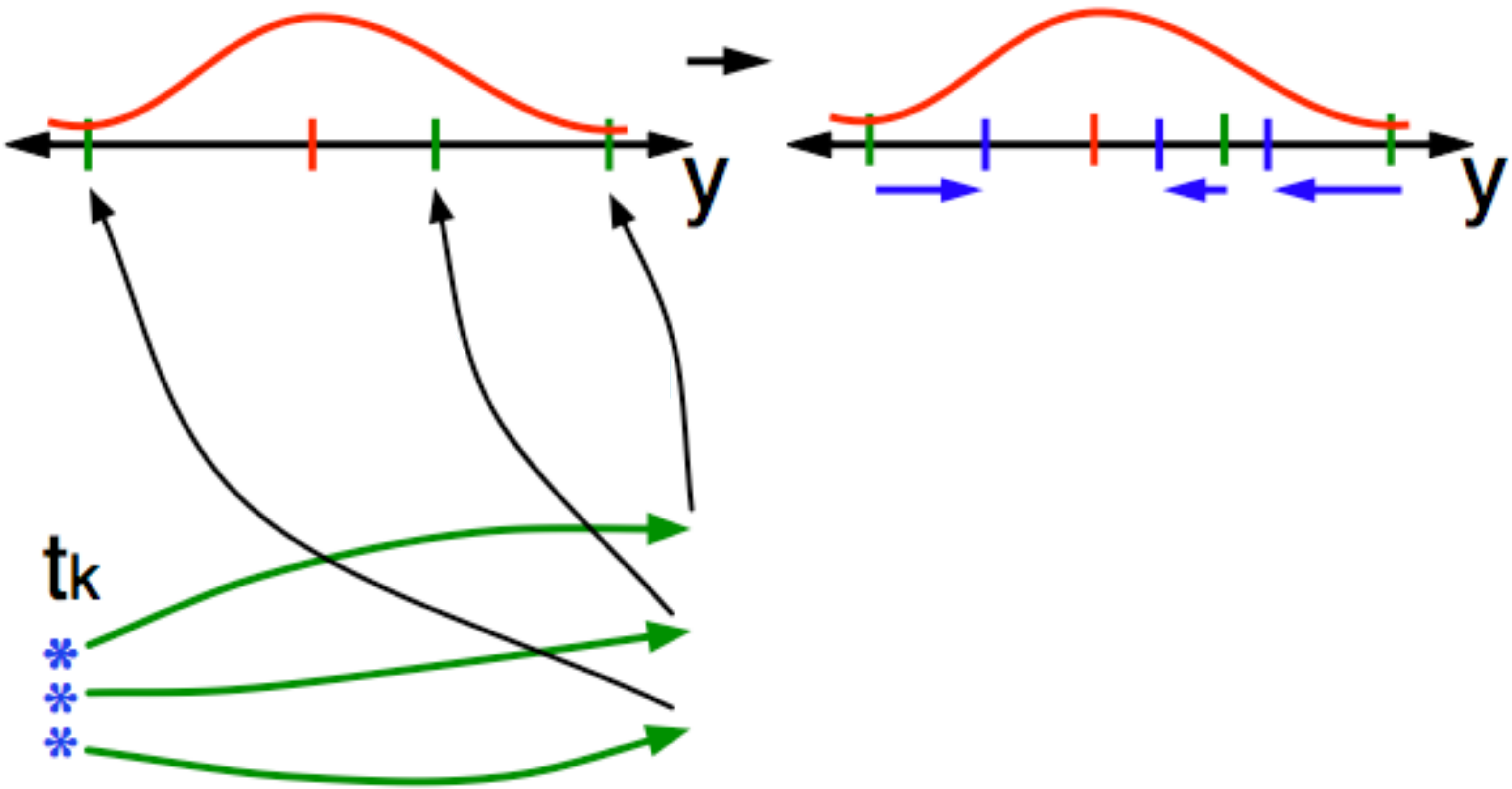
Theory: observations from instruments with uncorrelated errors can be done sequentially.

Houtekamer, P.L. and H.L. Mitchell, 2001: A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.*, **129**, 123-137

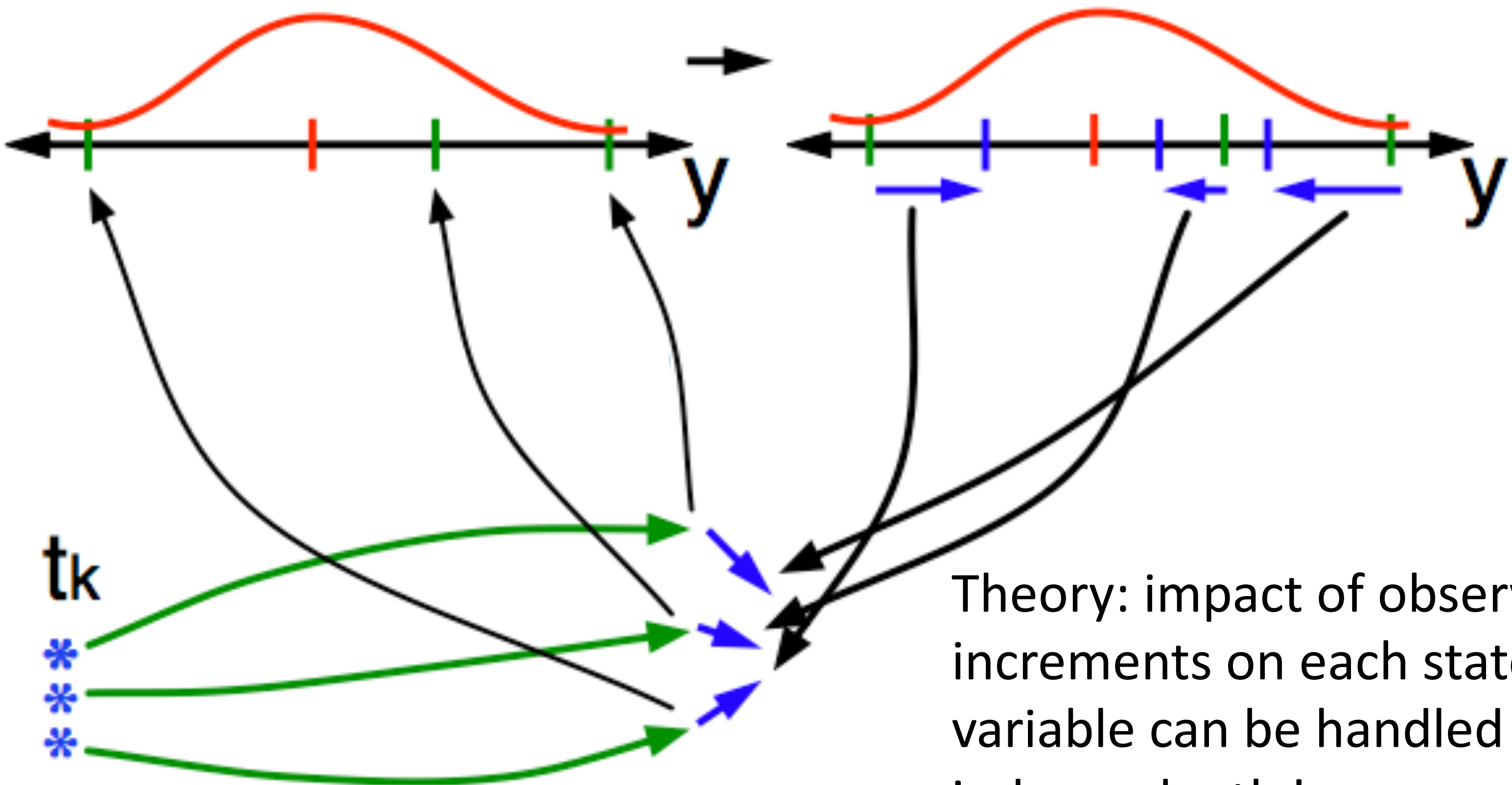
3. Get observed value and likelihood from observing system.



4. Find the increments for the prior observation ensemble, this is a scalar problem.

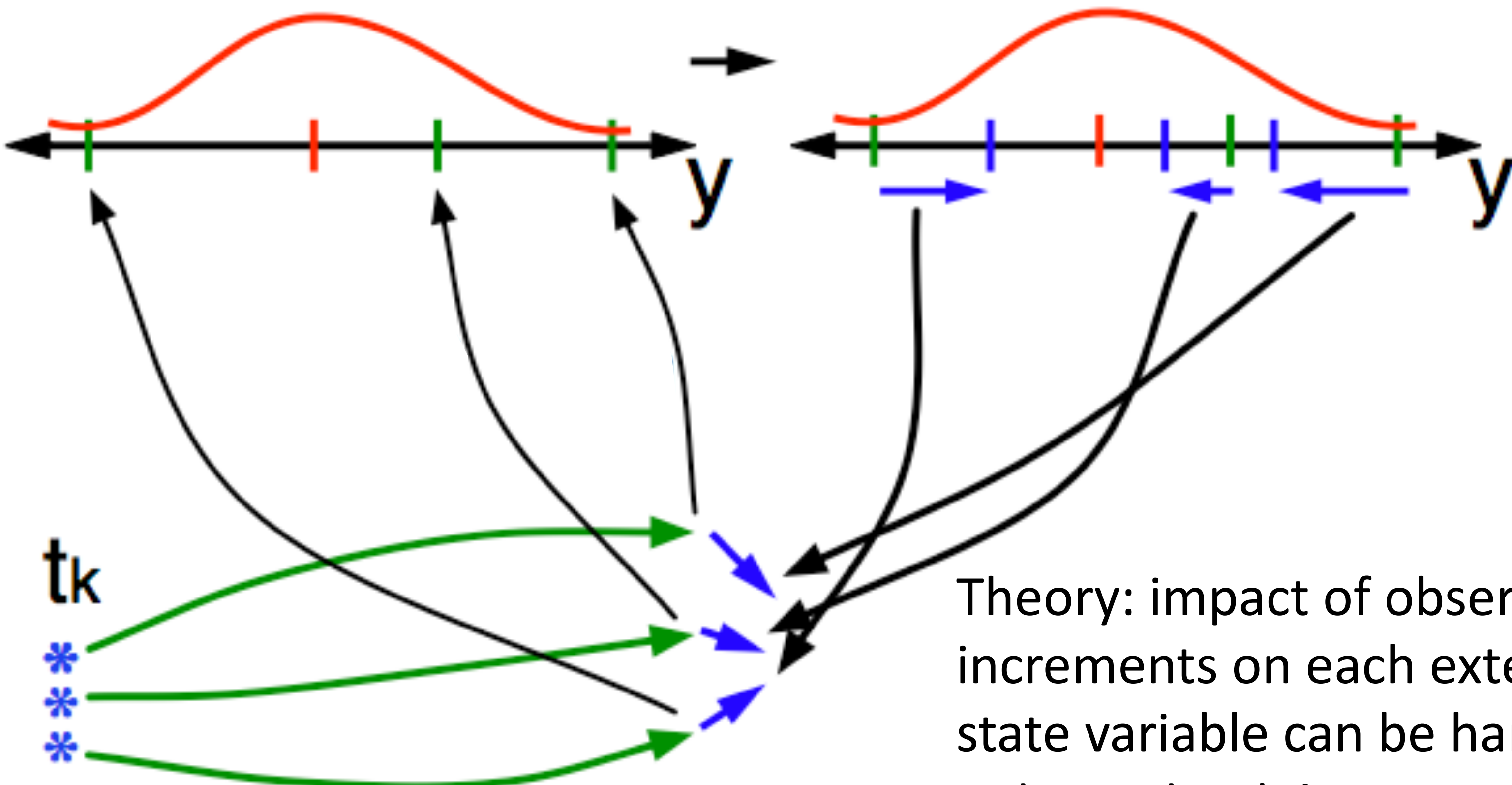


5. Use ensemble samples of **y** and each state variable to linearly regress observation increments onto extended state variable increments.



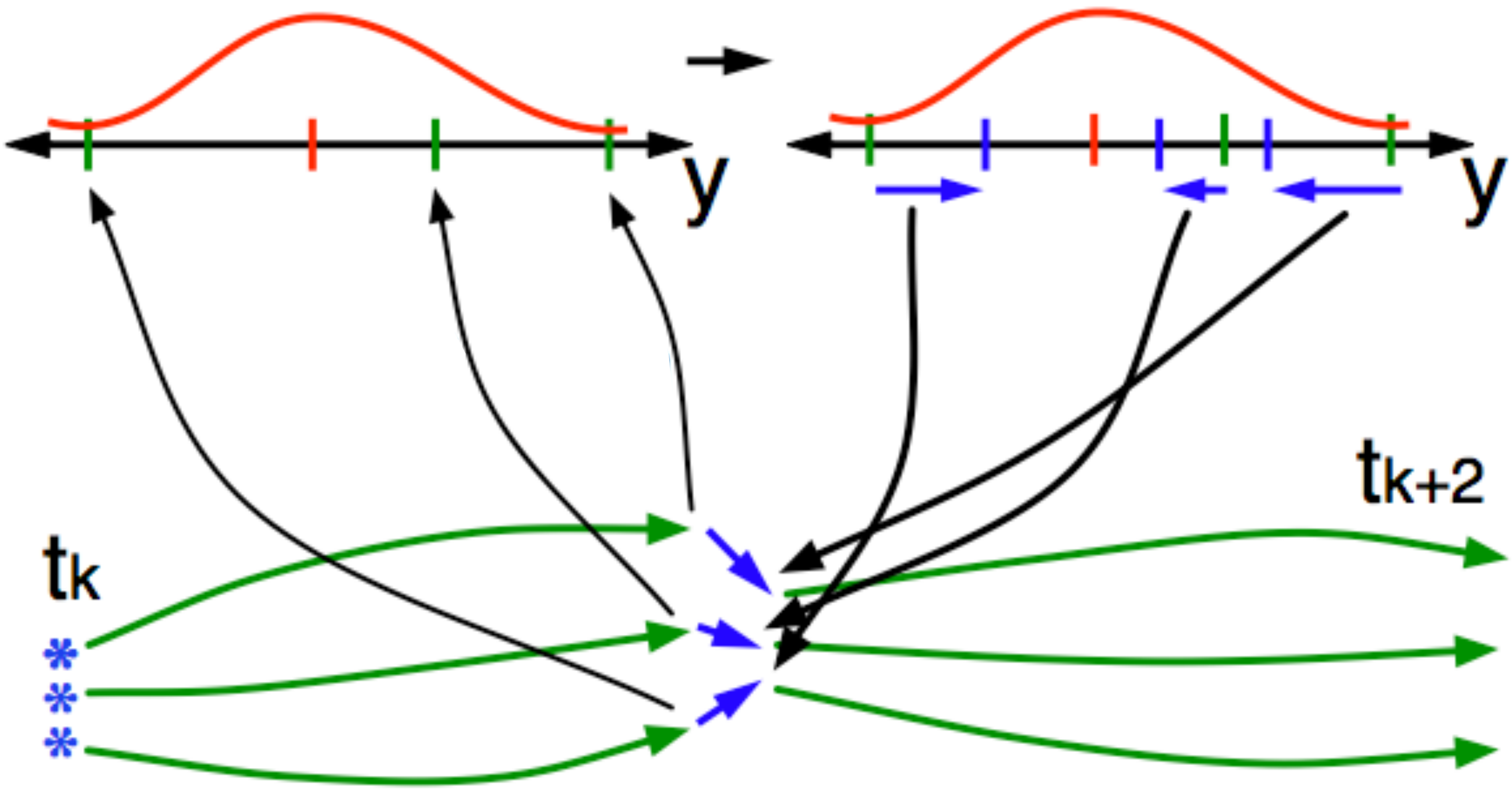
Theory: impact of observation increments on each state variable can be handled independently!

Repeat steps 2-5 sequentially for each observation at this time.



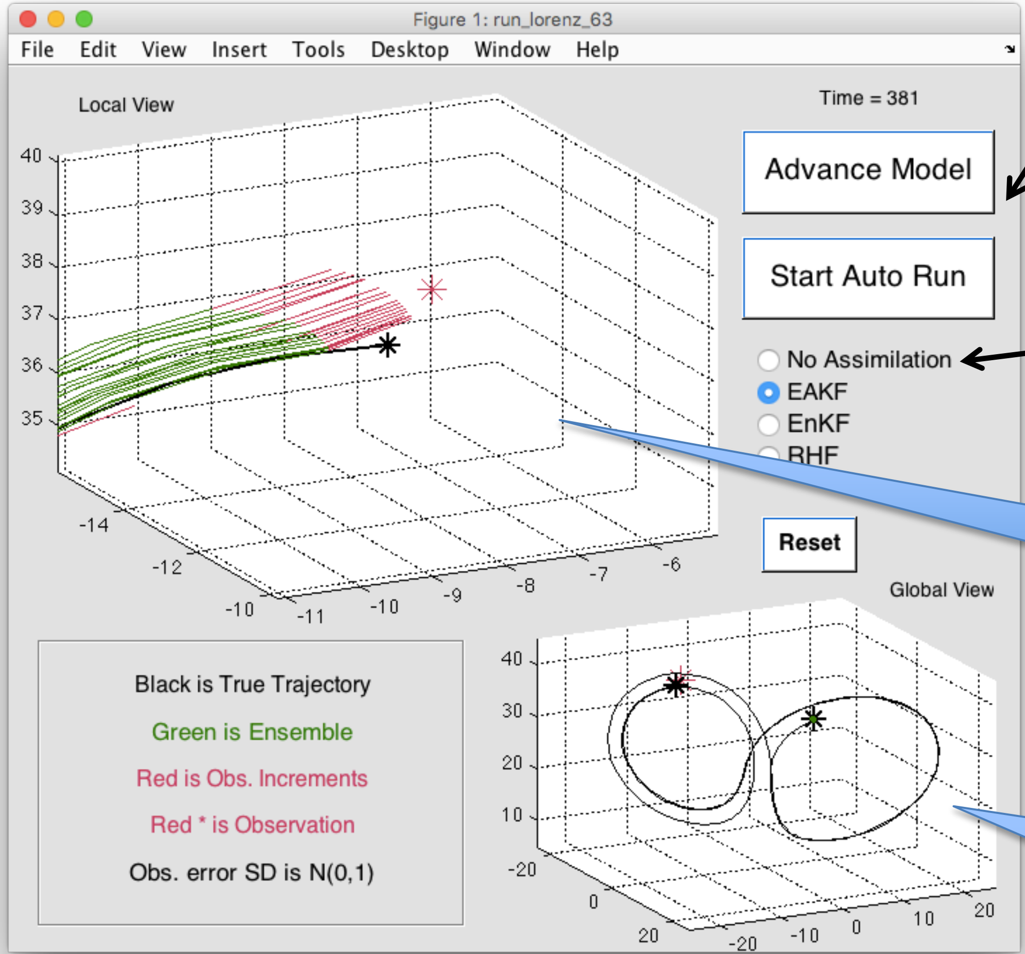
Theory: impact of observation increments on each extended state variable can be handled independently!

6. When all observations are assimilated, integrate model state to the next time that has observations.



Matlab Hands-On: run_lorenz_63

Purpose: Explore behavior of ensemble Kalman filters in a low-order, chaotic dynamical system, the 3-variable Lorenz 1963 model.



These controls work the same as for oned_model.

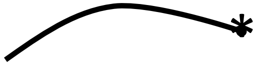
Assimilation can be turned off, just does model advances.

'Local' domain, local timeframe

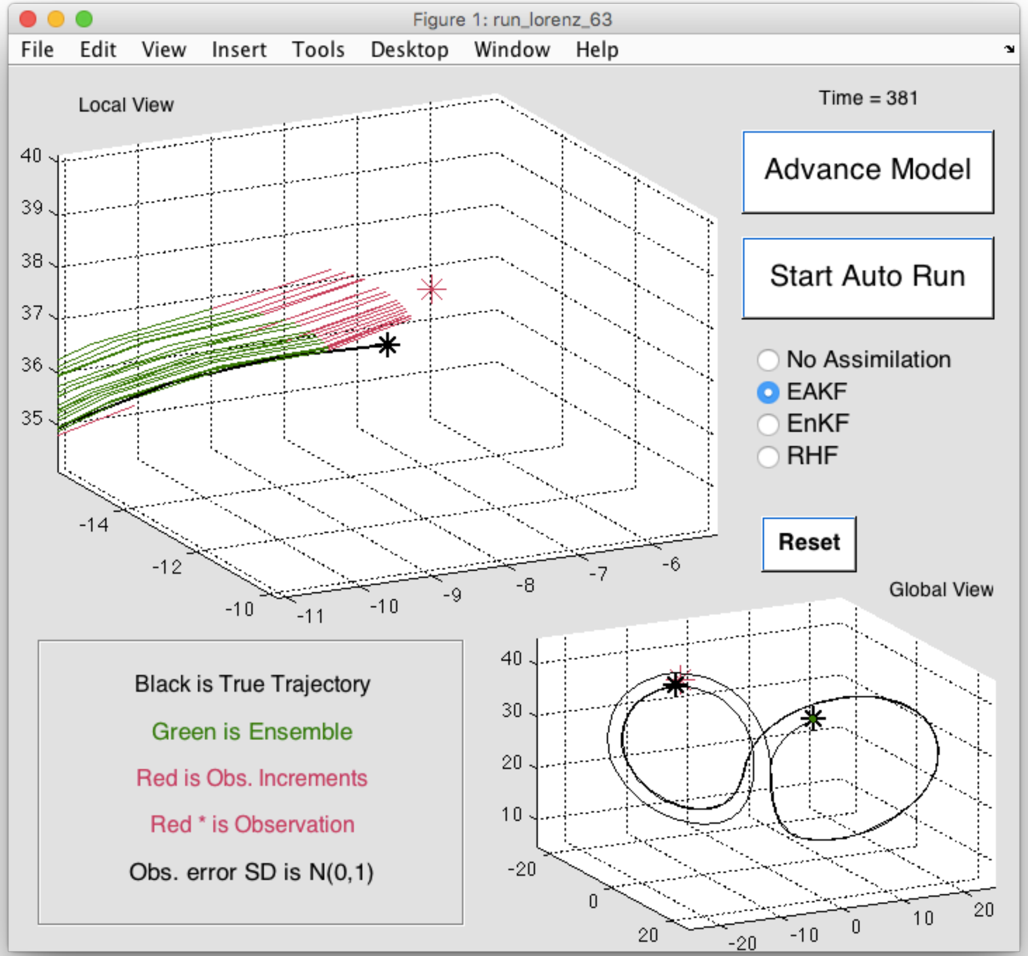
Full domain, full timeframe.

Matlab Hands-On: run_lorenz_63

Both panels show time evolution of true state (black).



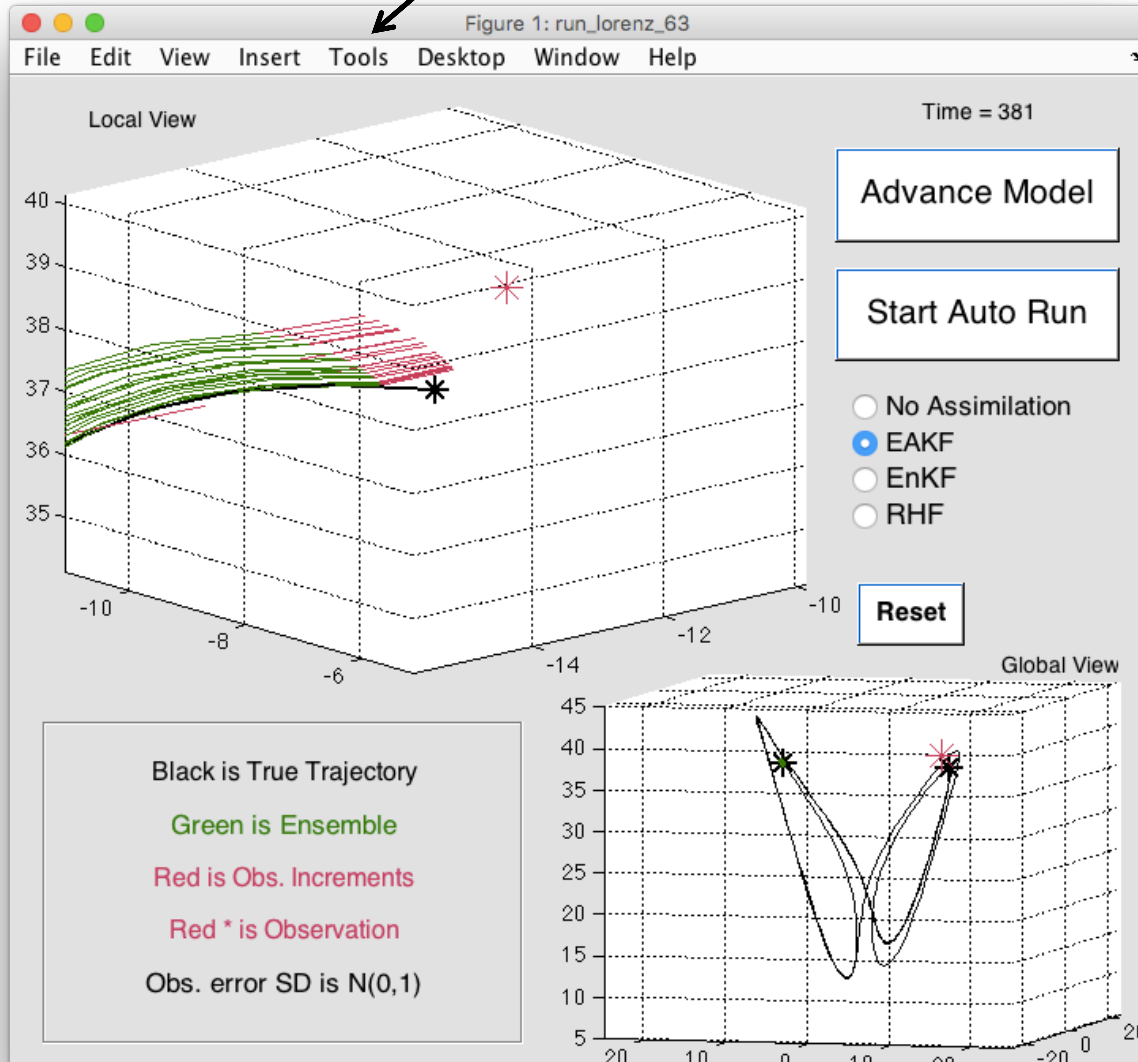
20 ensemble members are shown in green in top window.



At each observation time, the three components of the truth are 'observed' by adding a random draw from a standard normal distribution to the true value.

Matlab Hands-On: run_lorenz_63

You can use Matlab tools to modify plots.



Here, the Rotate 3D tool has been used to change the angle of view of both the local and global views of the assimilation.

Matlab Hands-On: run_lorenz_63

Explorations:

- Select Start Auto Run and watch the evolution of the ensemble. Try to understand how the ensemble spreads out.
- Restart the GUI and select EAKF. Do individual advances and assimilations and observe the behavior.
- Do some auto runs with assimilation turned on.
- Explore how different areas of the attractor have different assimilation behavior.

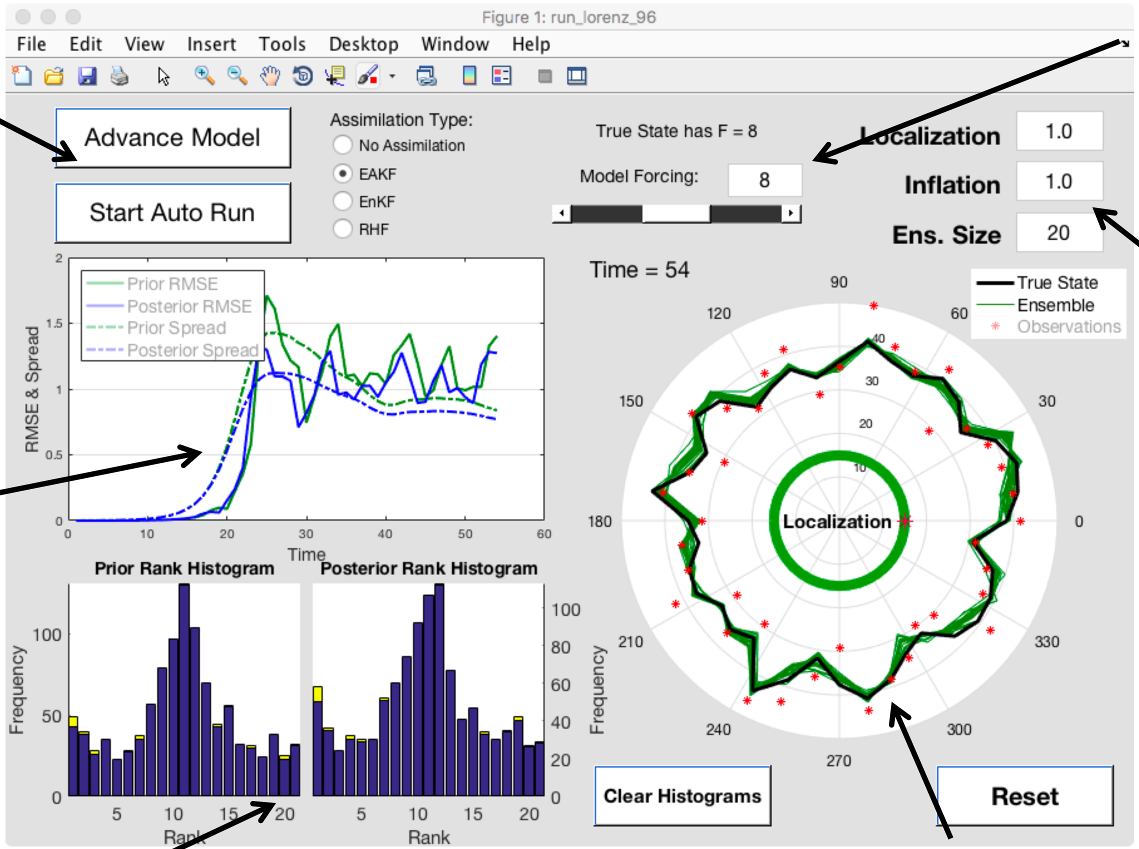
Matlab Hands-On: run_lorenz_96

Purpose: Explore the behavior of ensemble filters in a 40-variable chaotic dynamical system; the Lorenz 1996 model.

These controls work the same as lorenz_63.

Root mean square error from truth and ensemble spread as function of time.

Prior and posterior rank histograms.



Model forcing.

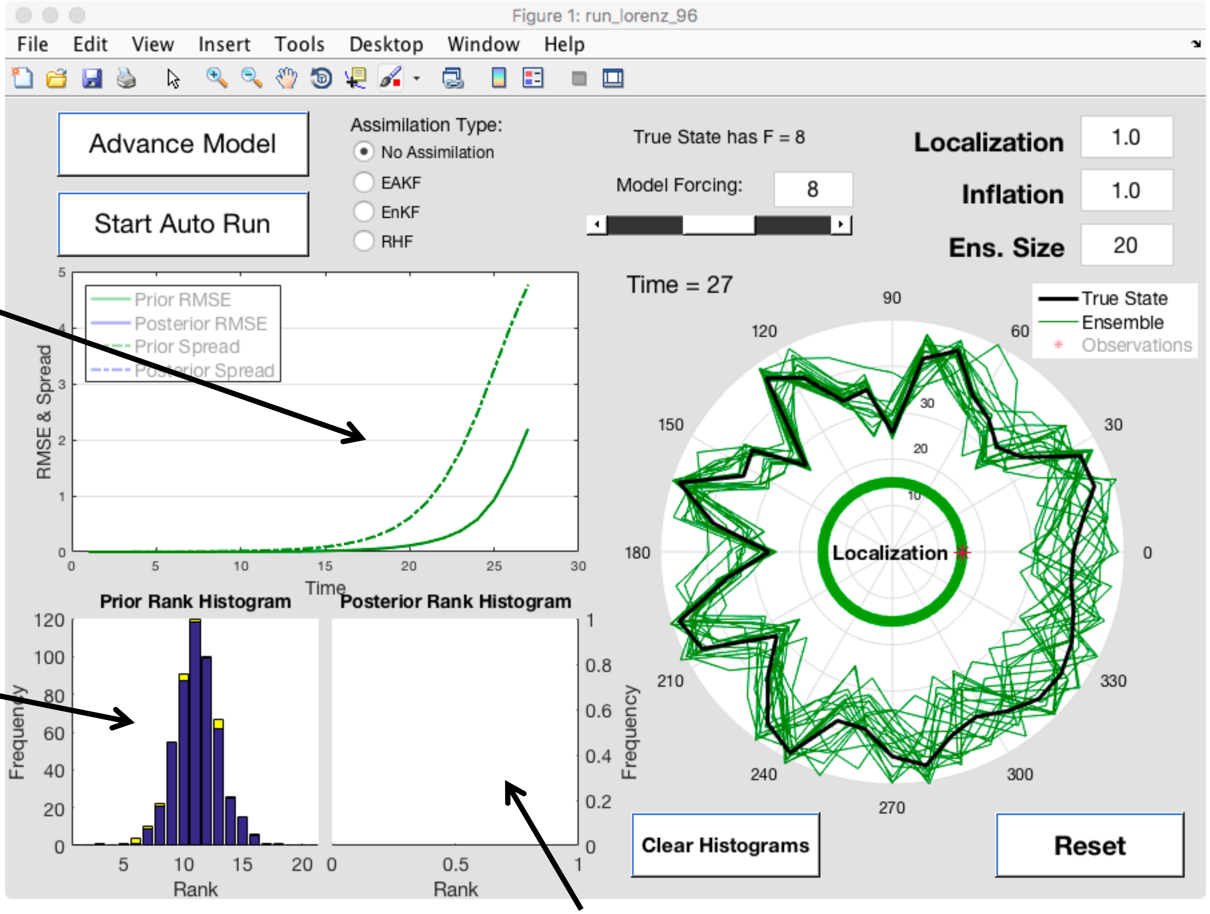
Parameters for ensemble filter.

Ensemble of model contours (spaghetti plot).

Matlab Hands-On: run_lorenz_96

Start a Free Run of the ensemble (No Assimilation). After some time, the minute perturbations in the original states lead to visibly different model states.

Takes a while for the small initial perturbations to grow.



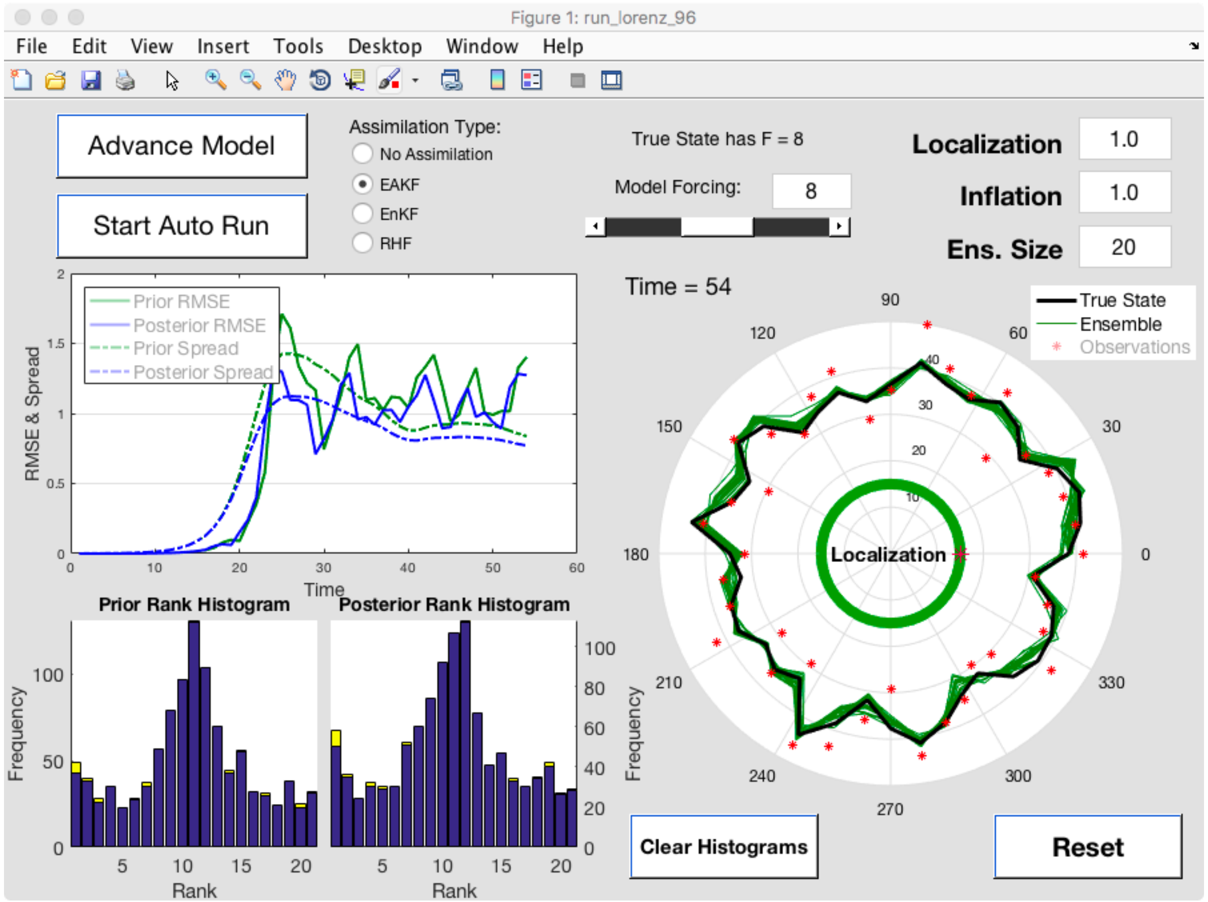
Ensemble size is 20, so there are 21 bins in the rank histogram.

No posterior in a free run.

Matlab Hands-On: run_lorenz_96

1) Stop the free run after some time.
2) Turn on the EAKF.
3) Advance model, assimilate…

Note: All 40 state variables are observed. Observation error standard deviation is 4.0



Your figures will be different depending on your settings. That's OK.

Matlab Hands-On: run_lorenz_96

Explorations:

- Do an extended free run to see error growth in the ensemble. How long does it take to saturate?
- Select EAKF and explore how the assimilation works.