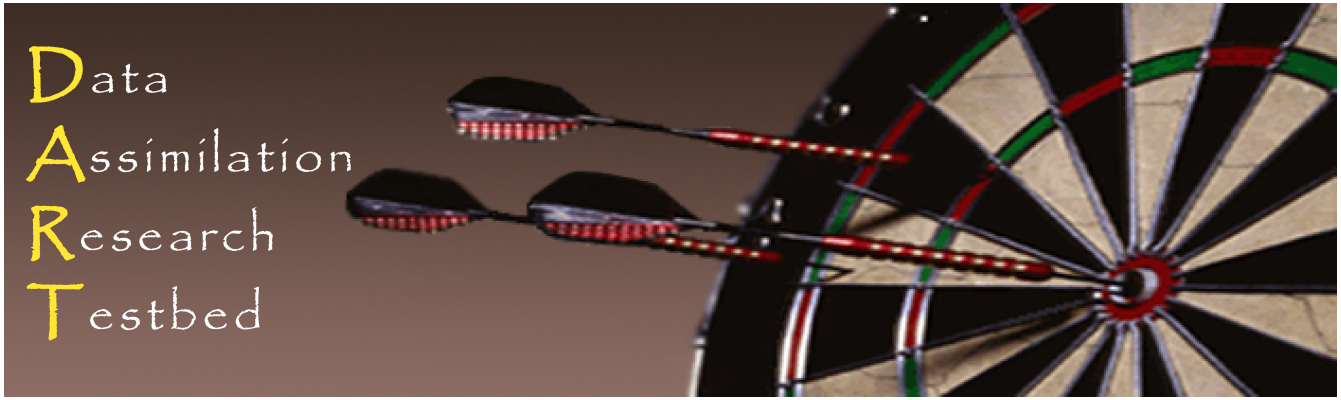
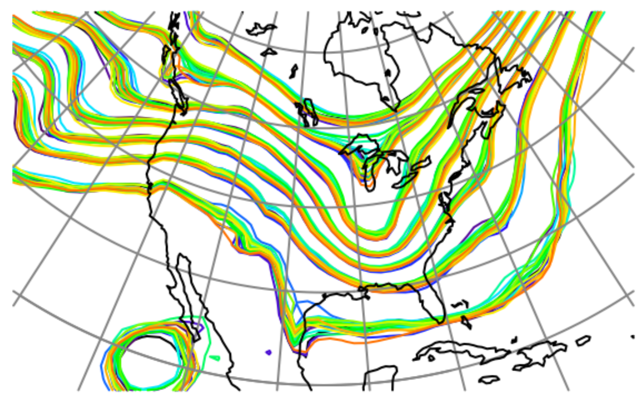


DART_LAB Tutorial Section 2: How should observations impact an unobserved state variable? Multivariate assimilation.

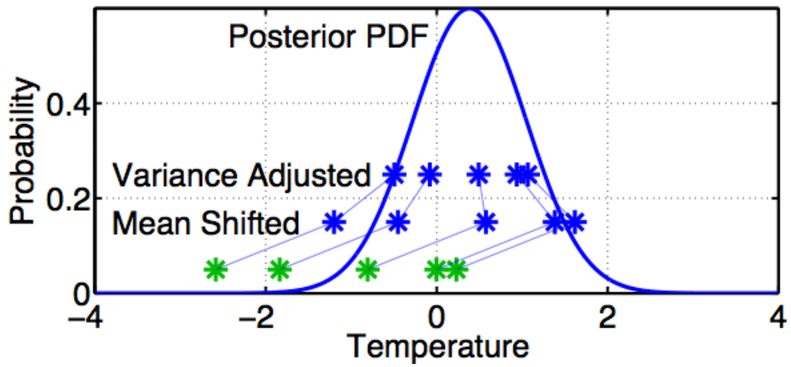


The National Center for Atmospheric Research is sponsored by the National Science Foundation. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

NCAR | UCAR National Center for Atmospheric Research

Single observed variable, single unobserved variable.

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

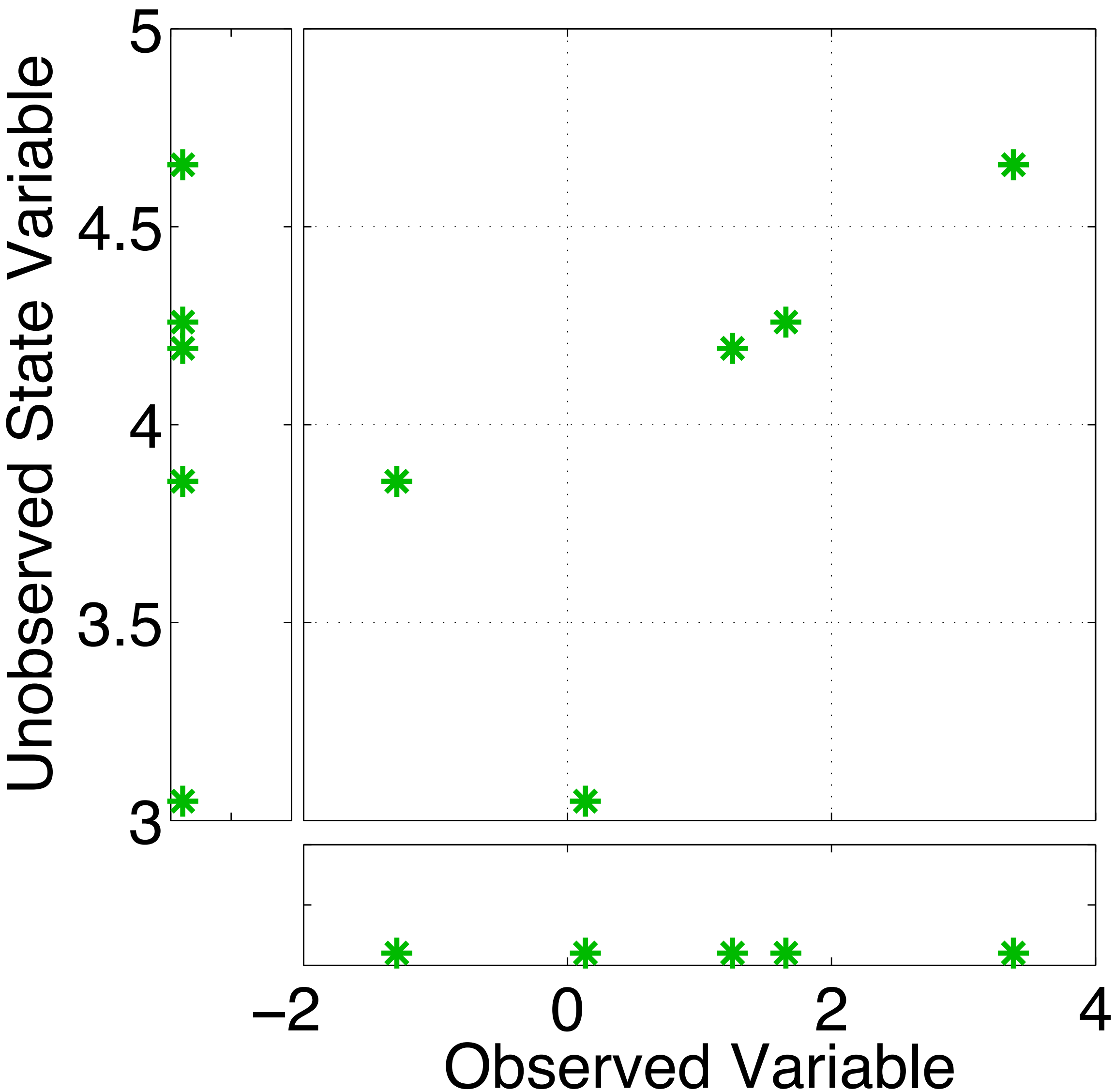


Now, suppose the prior has an additional variable …

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

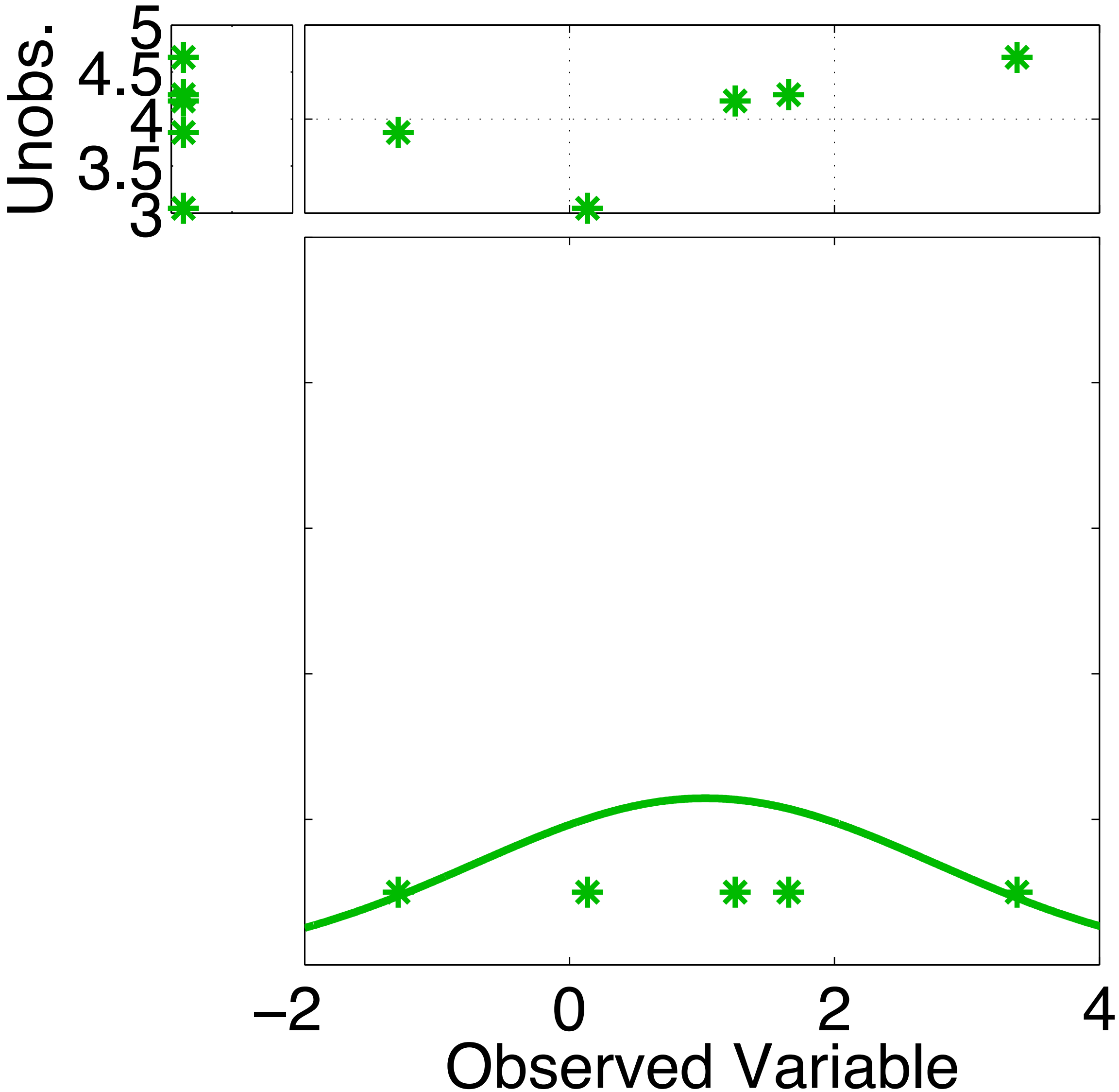


Assume that all we know is the prior joint distribution.

One variable is observed.

What should happen to the unobserved variable?

Ensemble filters: Updating additional prior state variables

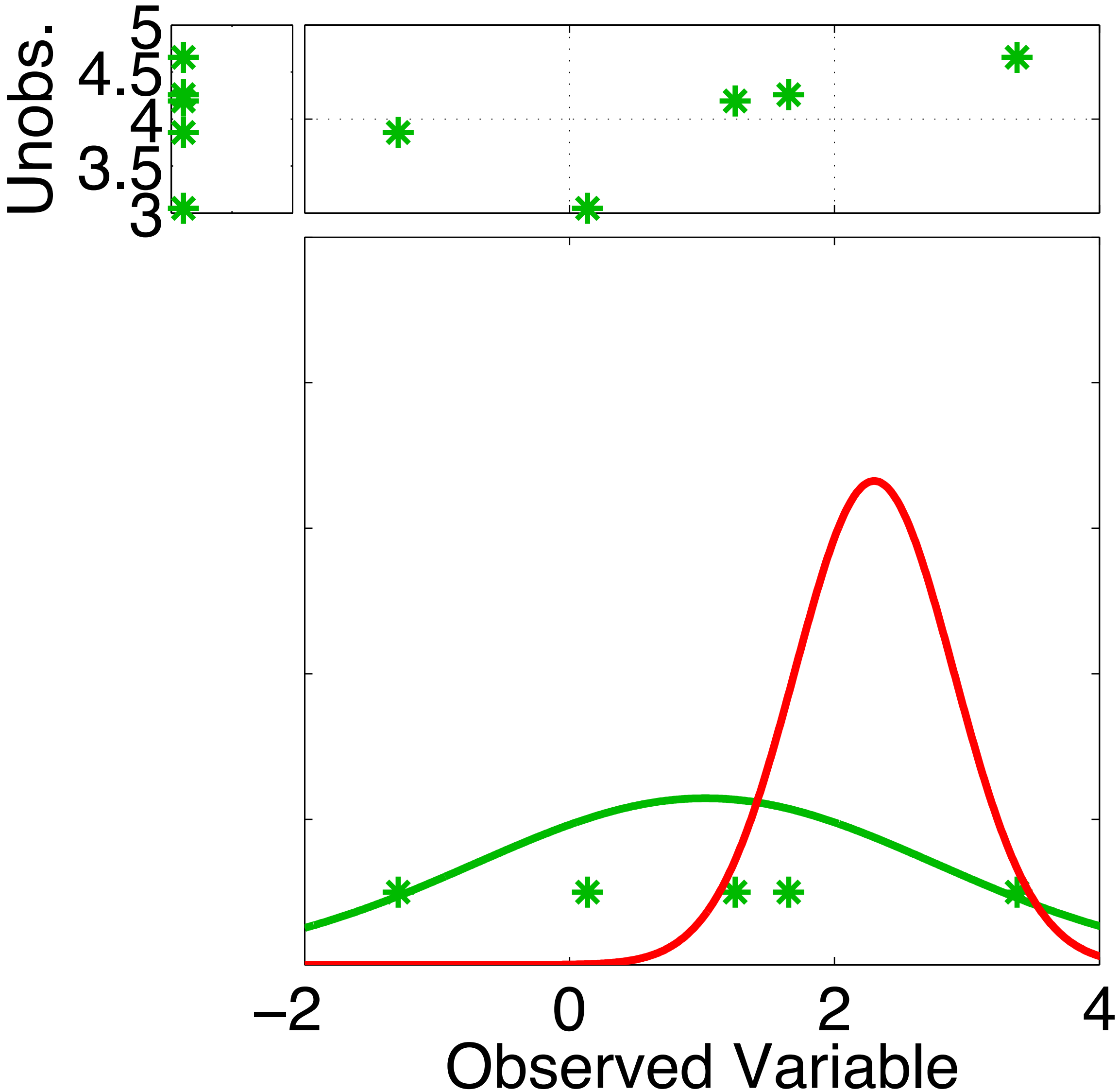


Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable with one of the previous methods.

Ensemble filters: Updating additional prior state variables

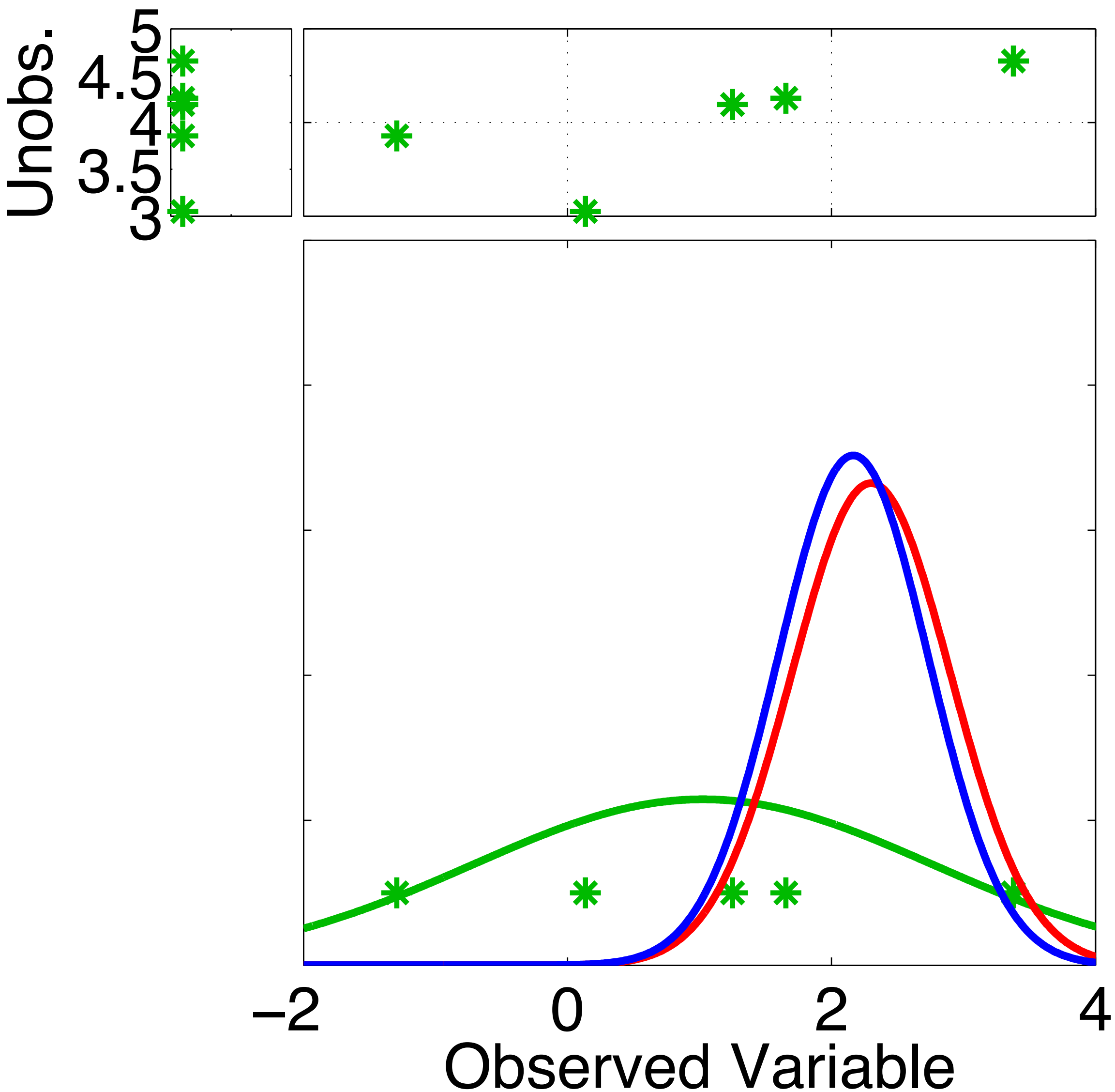


Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable with one of the previous methods.

Ensemble filters: Updating additional prior state variables

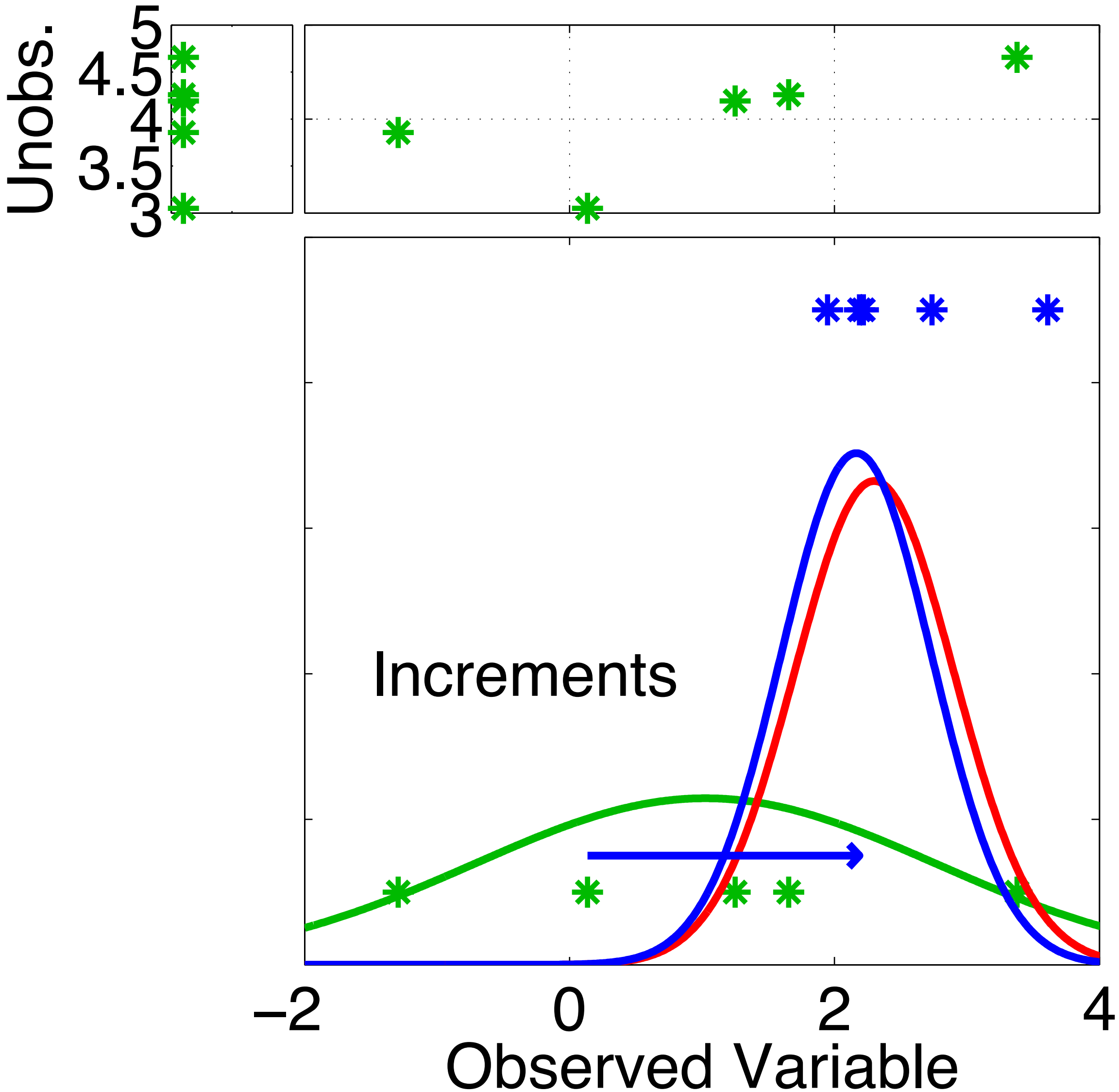


Assume that all we know is the prior joint distribution.

One variable is observed.

Update observed variable with one of the previous methods.

Ensemble filters: Updating additional prior state variables

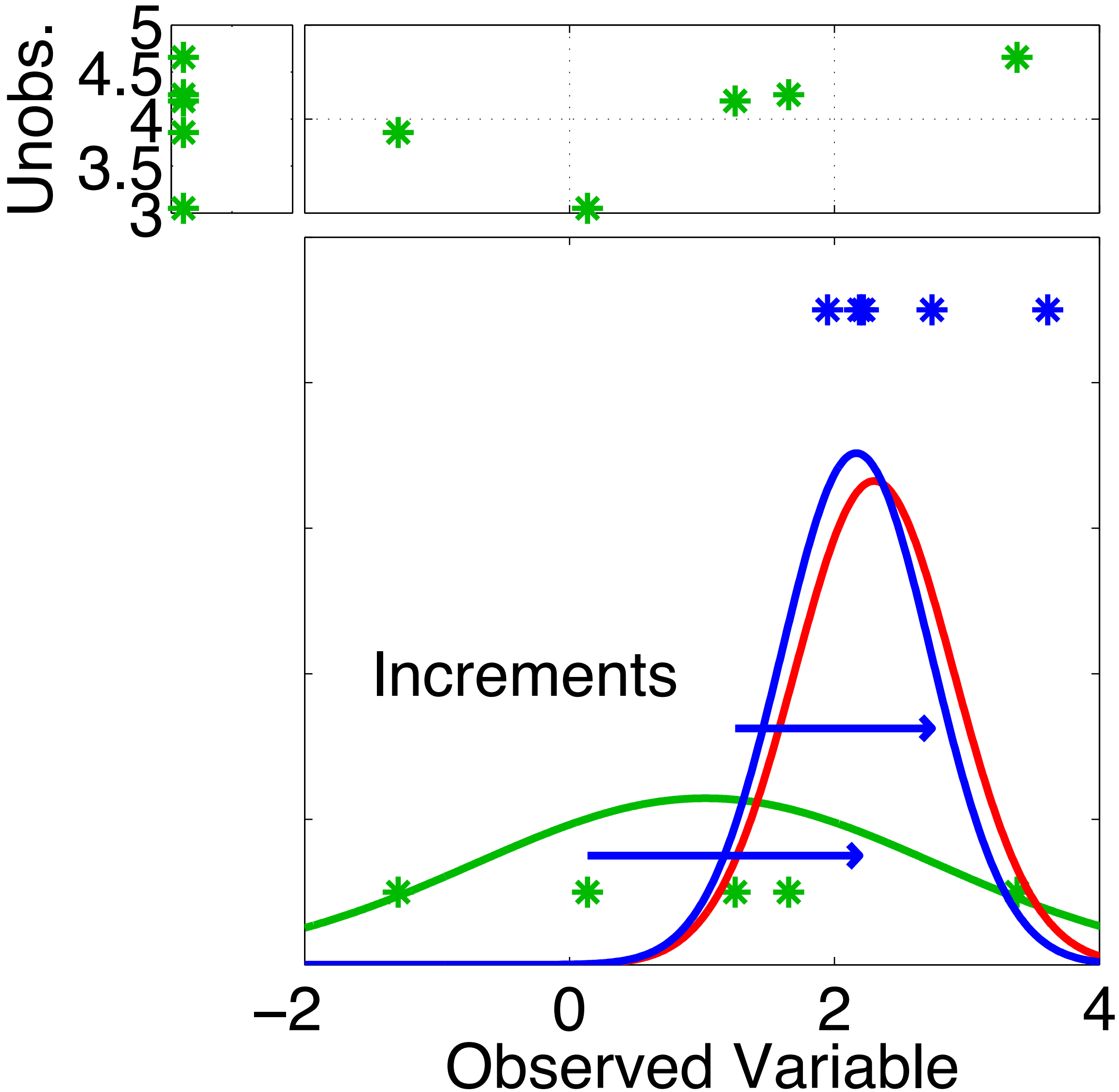


Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.

Ensemble filters: Updating additional prior state variables

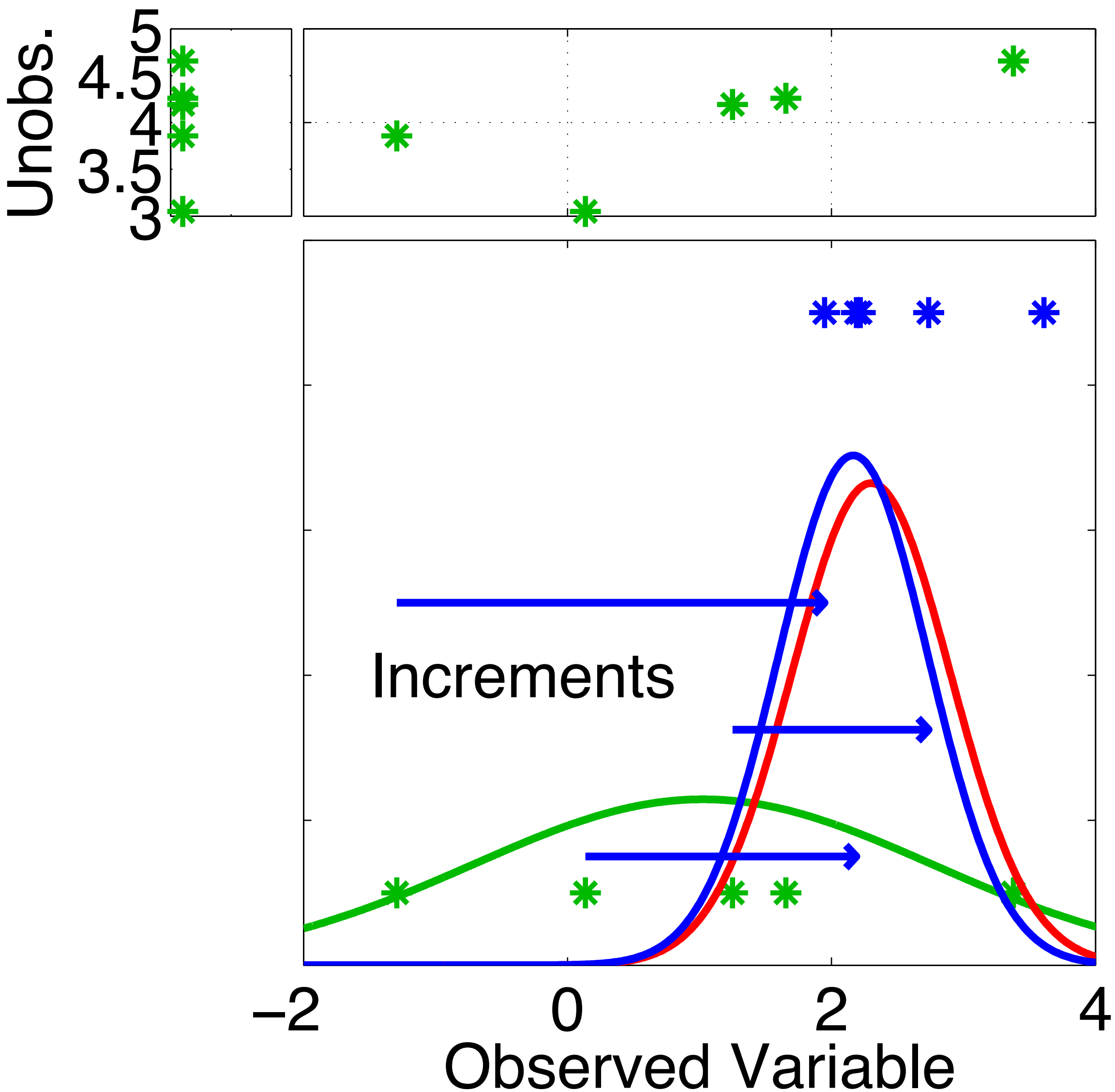


Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.

Ensemble filters: Updating additional prior state variables

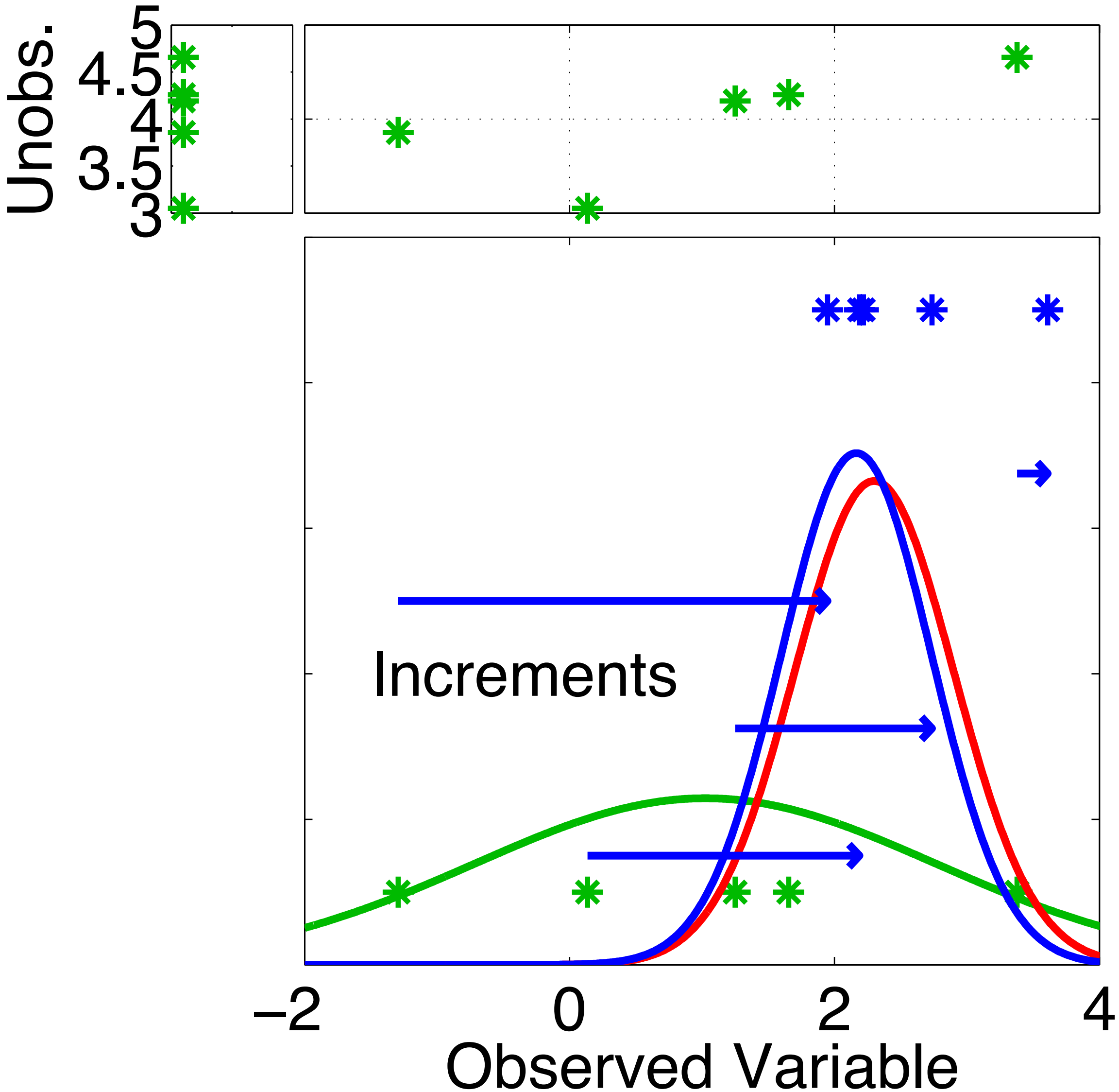


Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.

Ensemble filters: Updating additional prior state variables

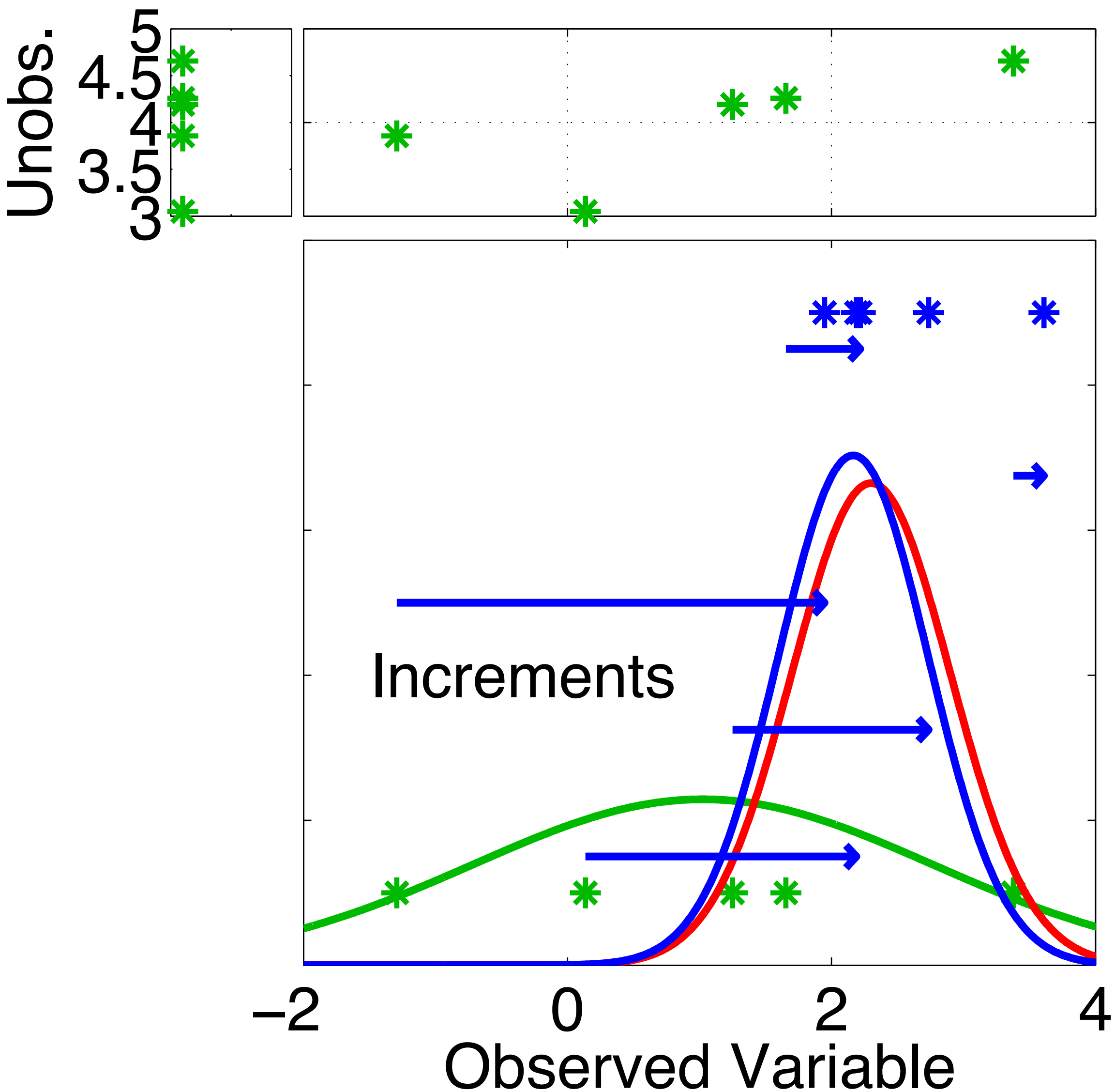


Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.

Ensemble filters: Updating additional prior state variables

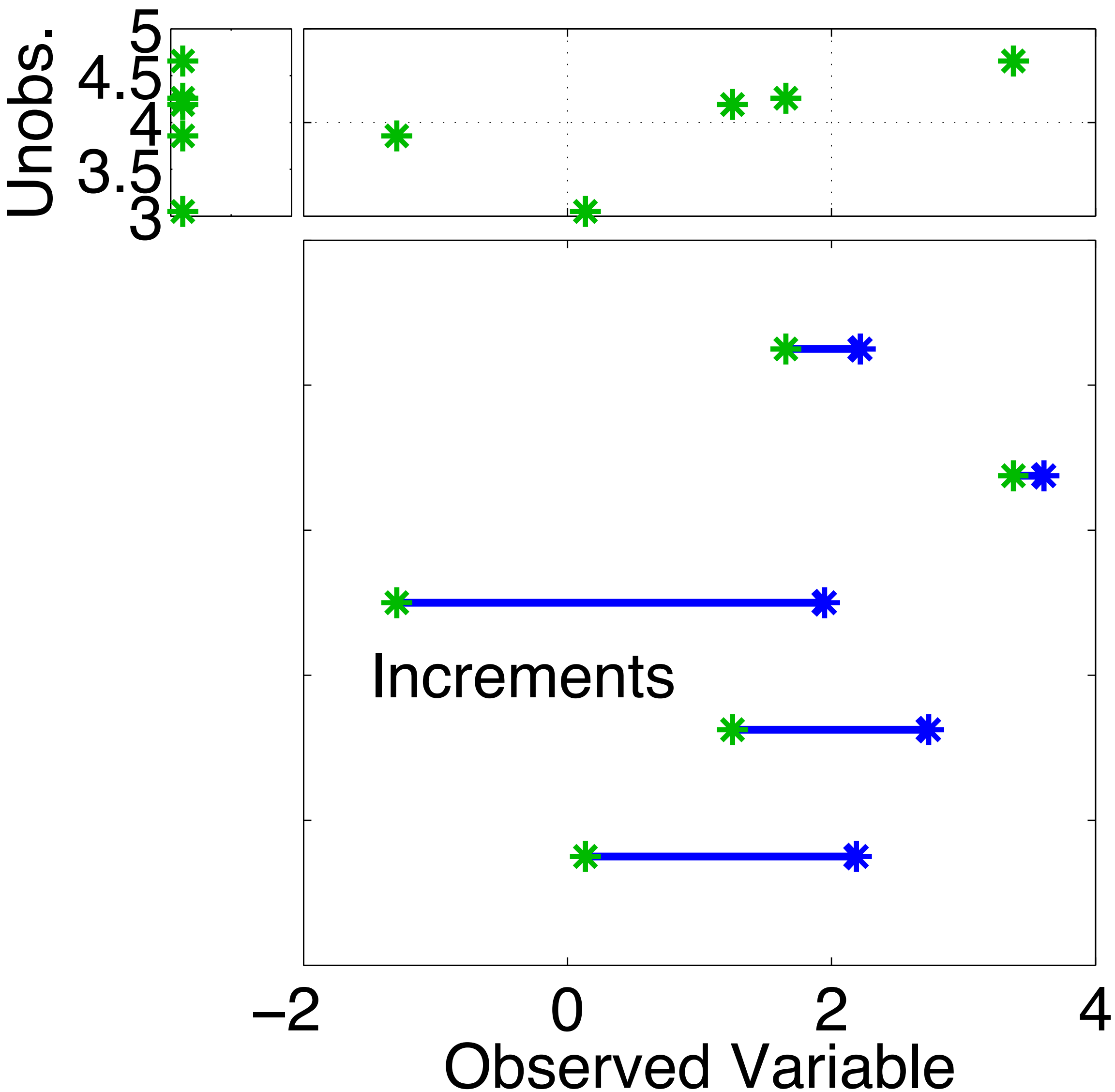


Assume that all we know is the prior joint distribution.

One variable is observed.

Compute increments for prior ensemble members of observed variable.

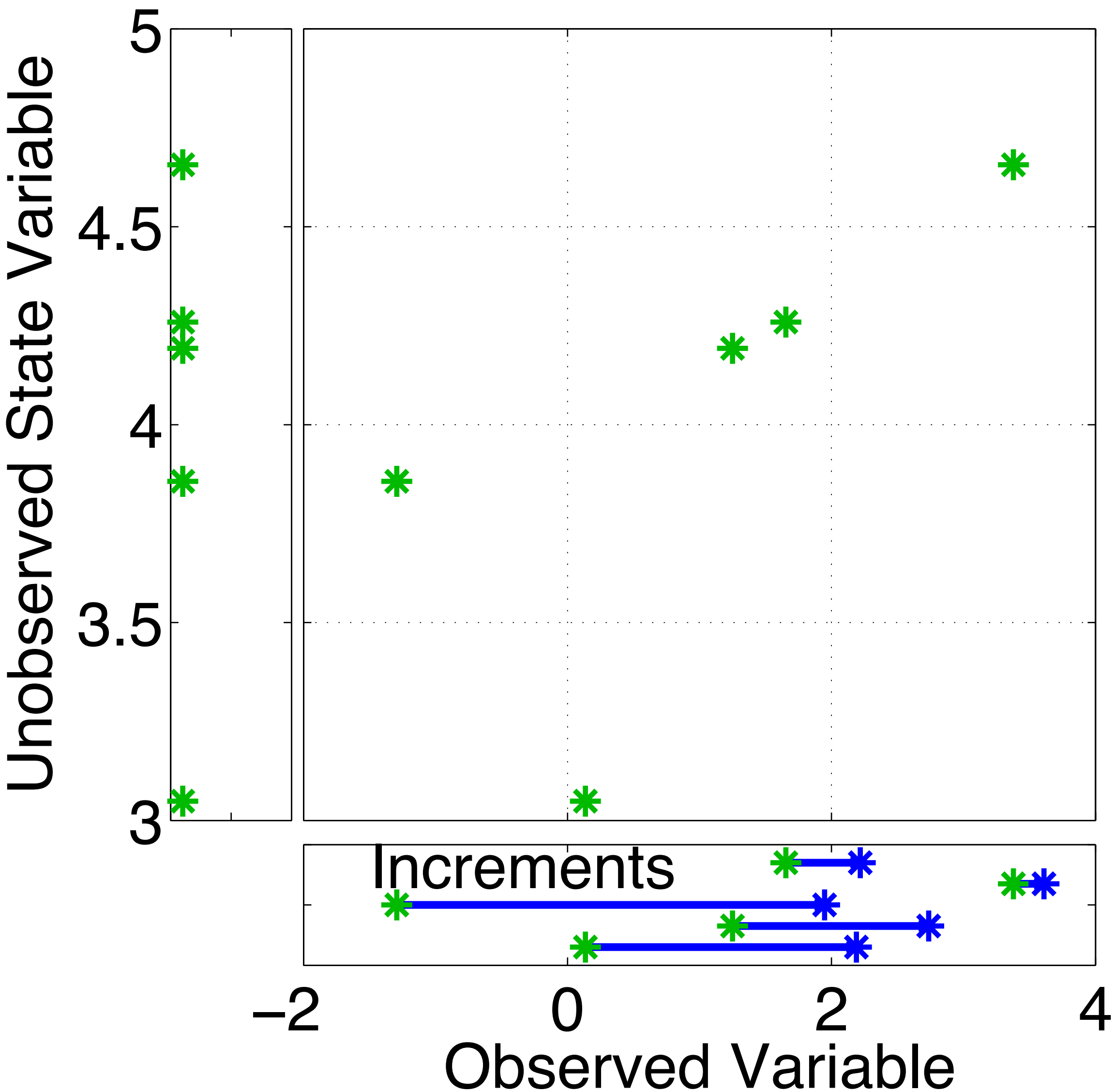
Ensemble filters: Updating additional prior state variables



Using only increments guarantees that if observation had no impact on observed variable, the unobserved variable is unchanged.

Highly desirable!

Ensemble filters: Updating additional prior state variables



Assume that all we know is the prior joint distribution.

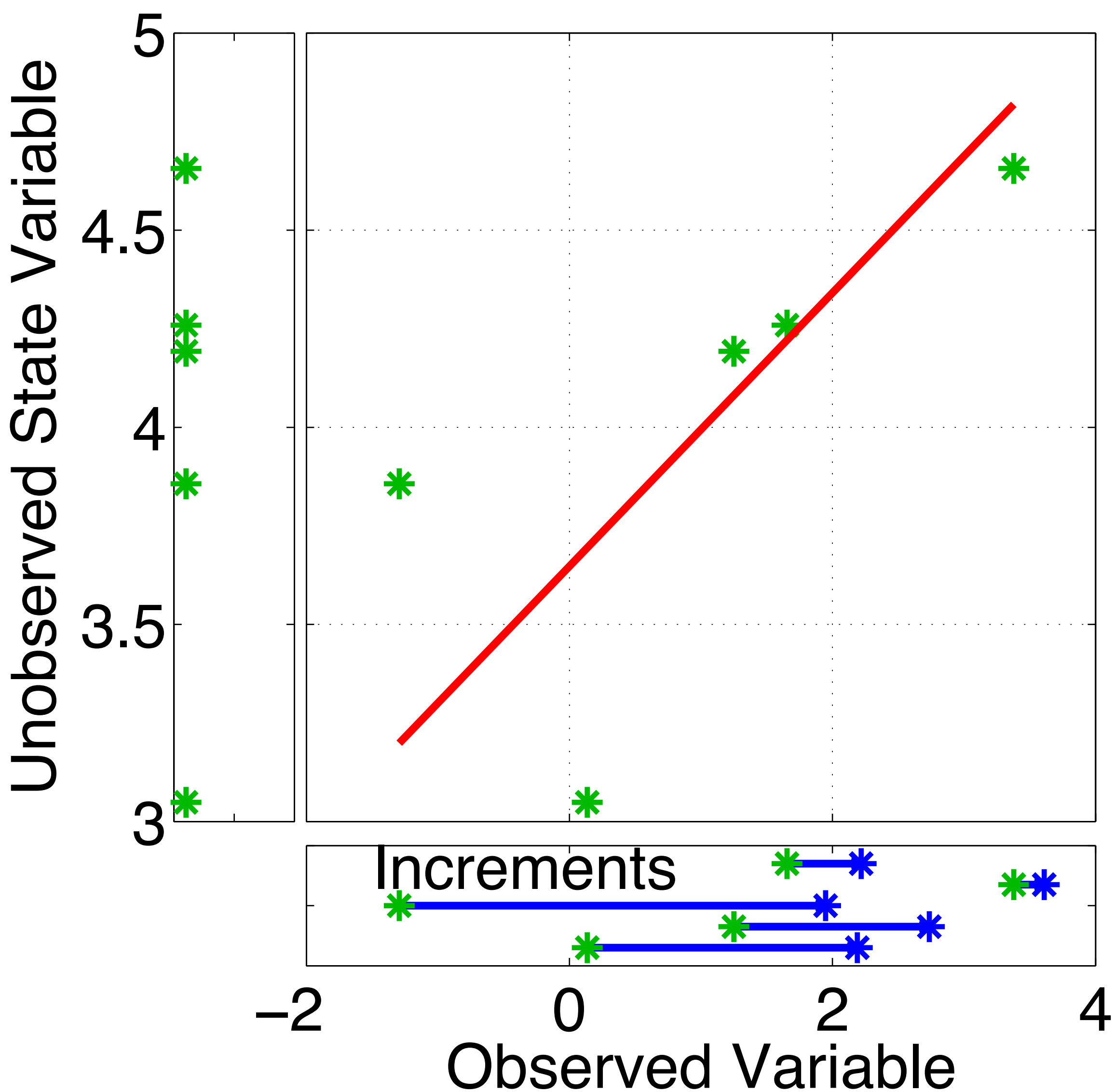
How should the unobserved variable be impacted?

1st choice: least squares

Equivalent to linear regression.

Same as assuming binormal prior.

Ensemble filters: Updating additional prior state variables



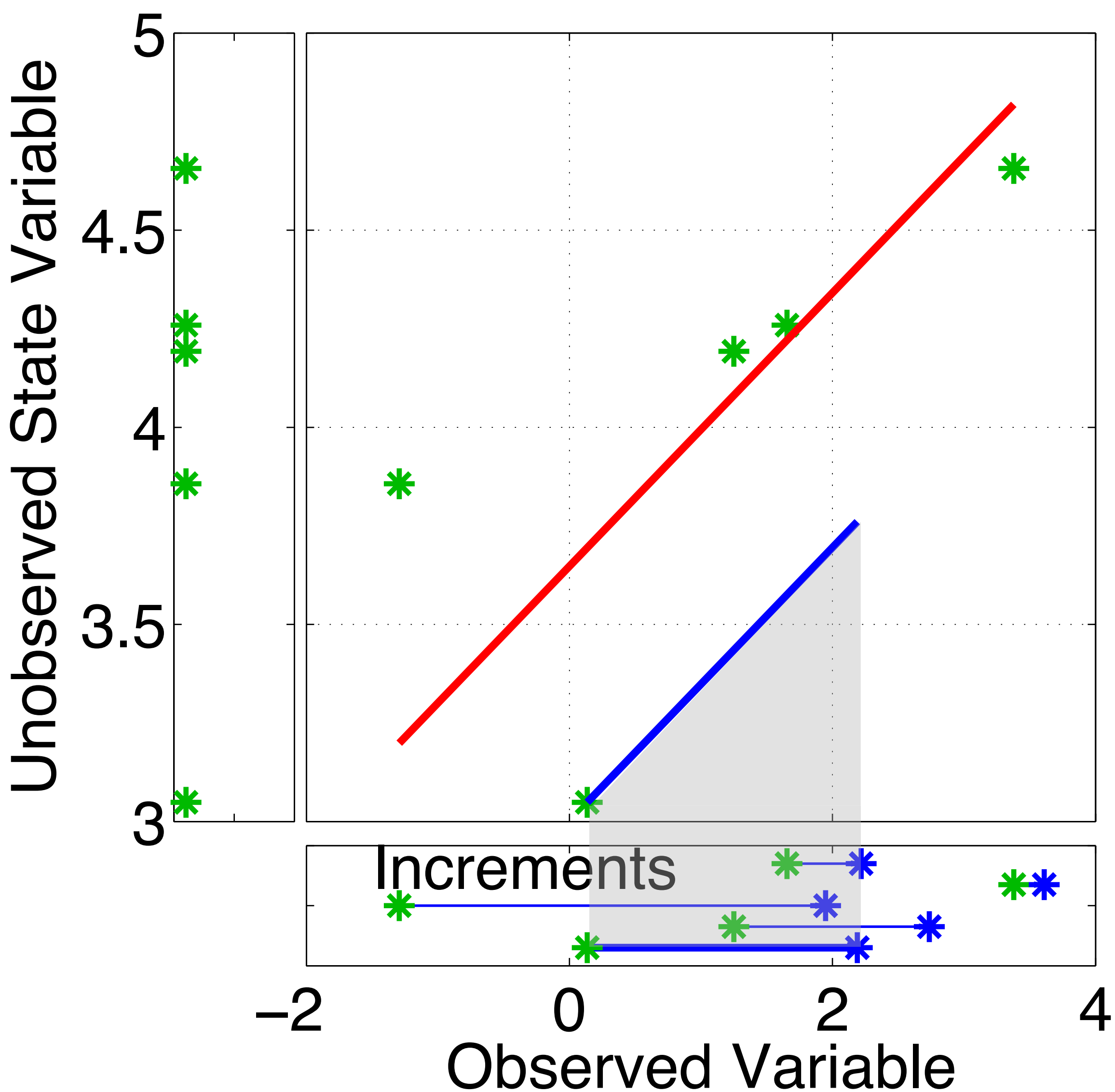
Have joint prior distribution of two variables.

How should the unobserved variable be impacted?

1st choice: least squares

Begin by finding least squares fit.

Ensemble filters: Updating additional prior state variables

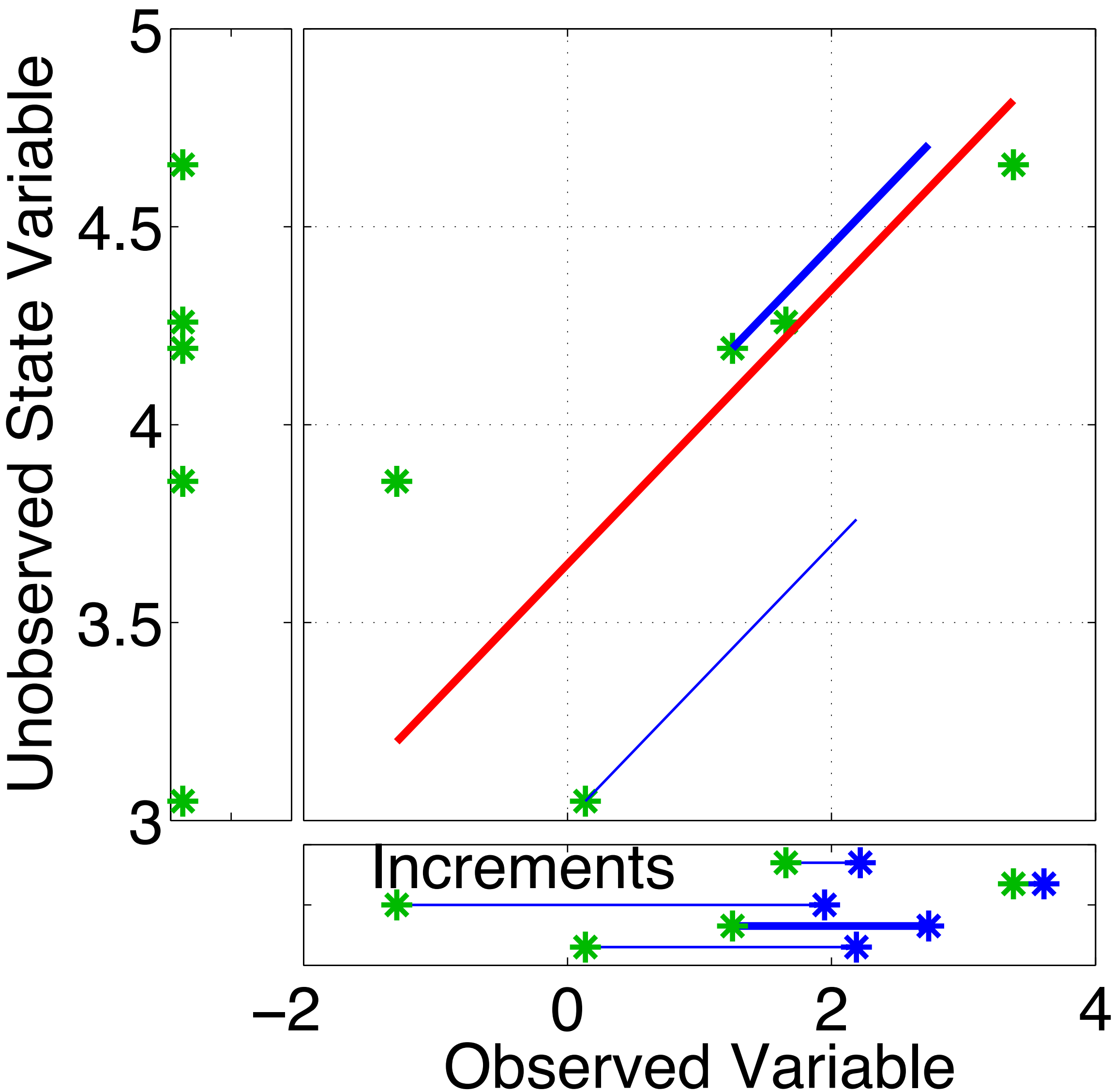


Have joint prior distribution of two variables.

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

Equivalent to first finding image of increment in joint space.

Ensemble filters: Updating additional prior state variables

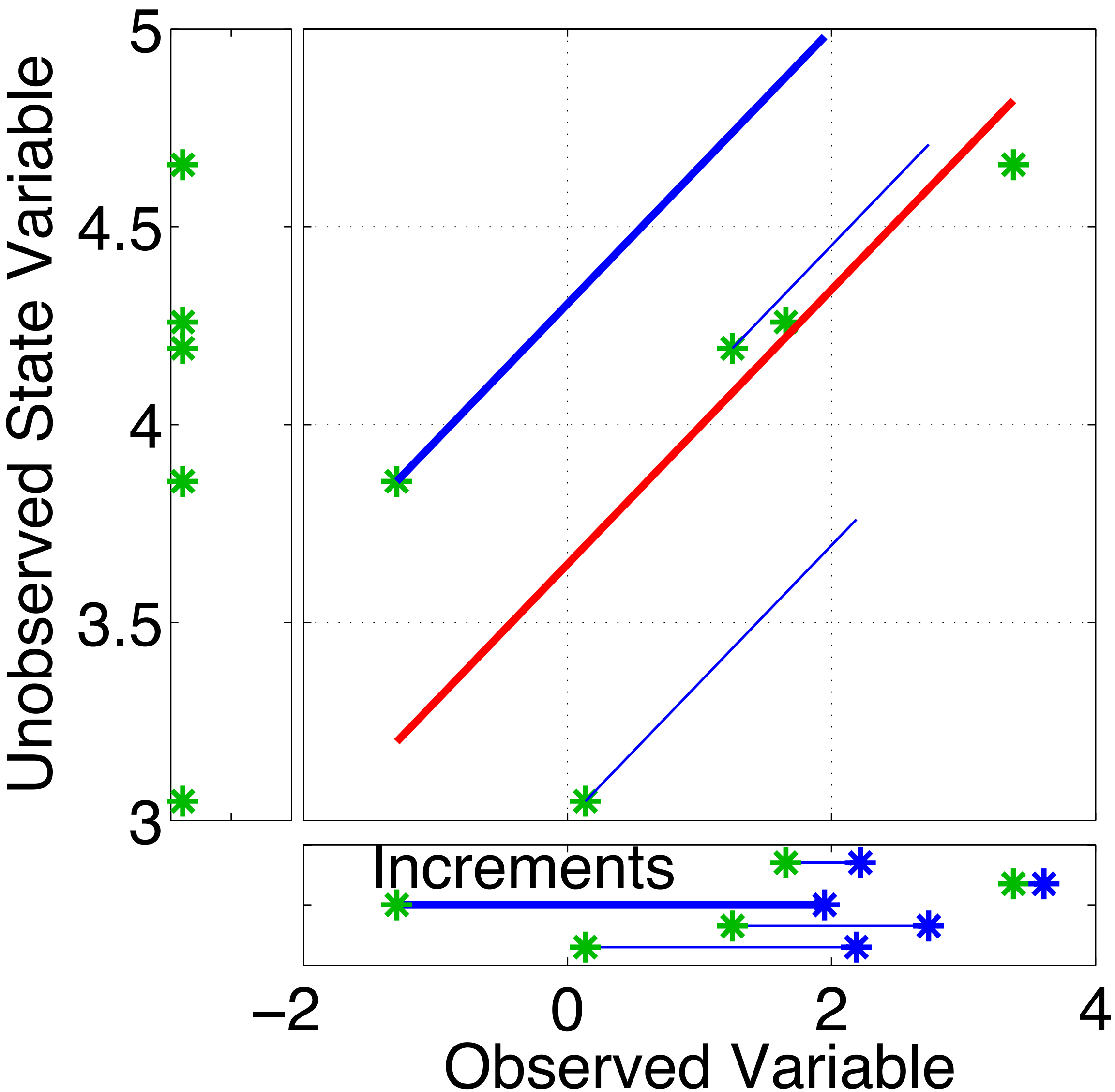


Have joint prior distribution of two variables.

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

Equivalent to first finding image of increment in joint space.

Ensemble filters: Updating additional prior state variables

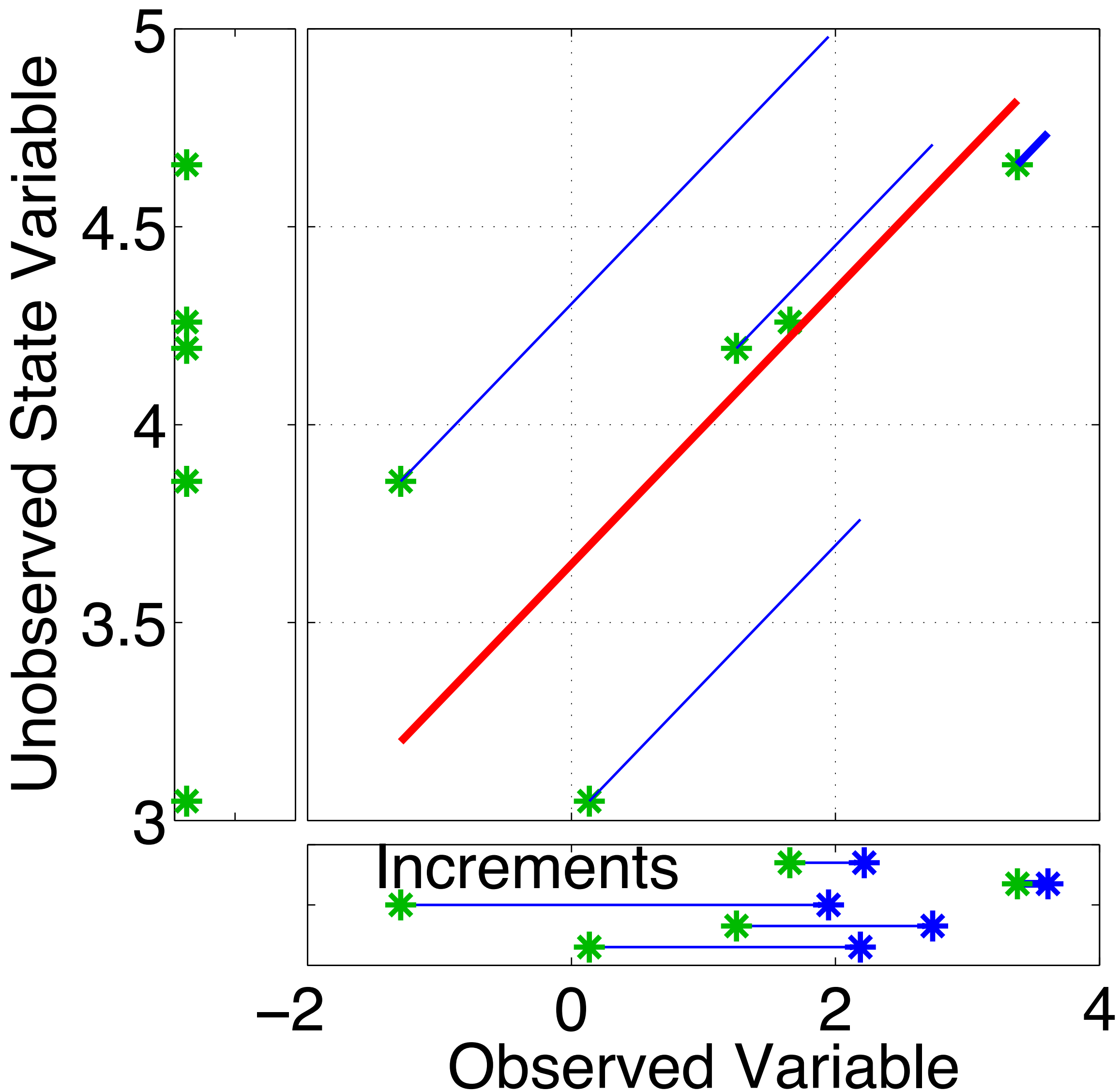


Have joint prior distribution of two variables.

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

Equivalent to first finding image of increment in joint space.

Ensemble filters: Updating additional prior state variables

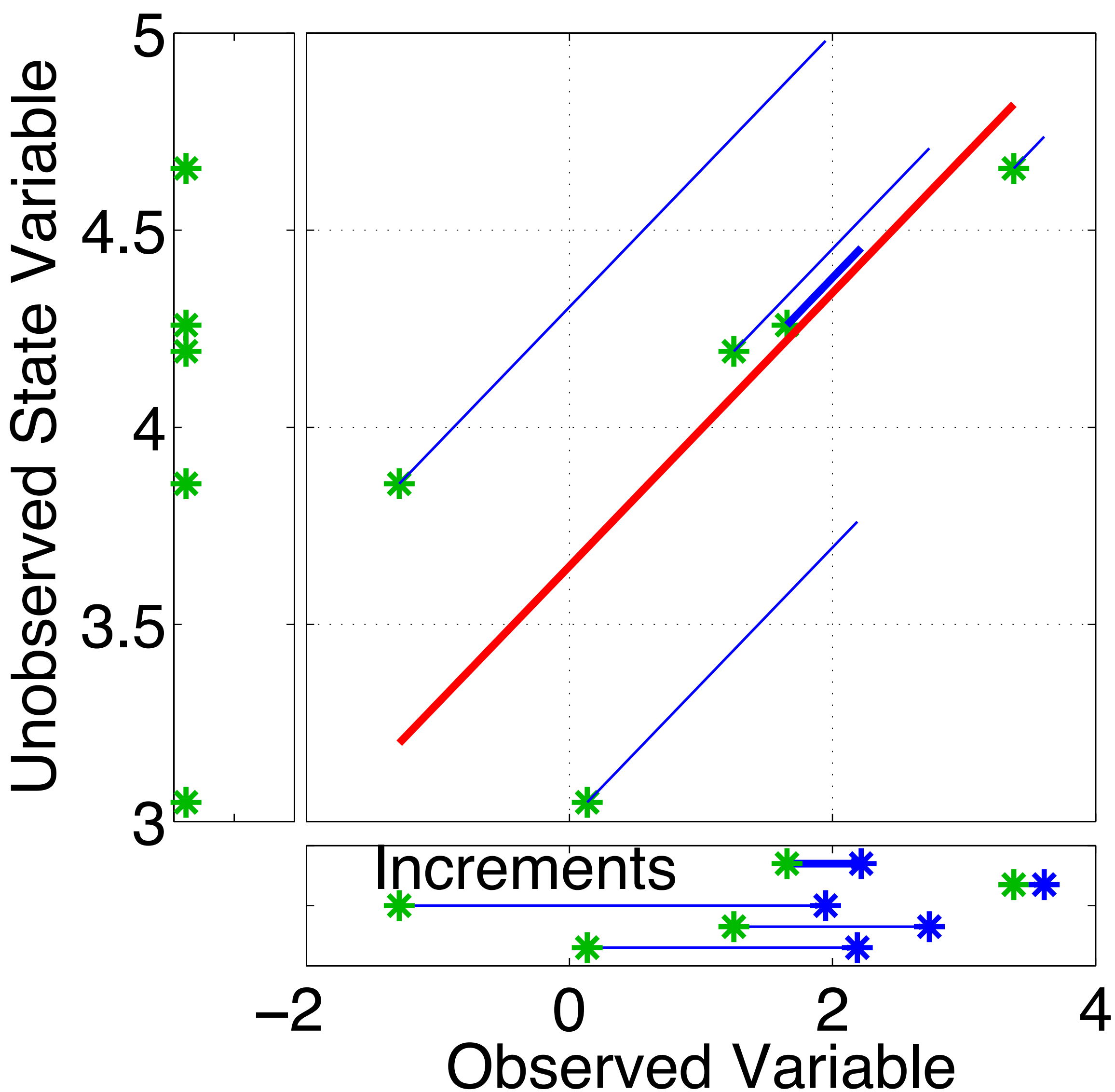


Have joint prior distribution of two variables.

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

Equivalent to first finding image of increment in joint space.

Ensemble filters: Updating additional prior state variables

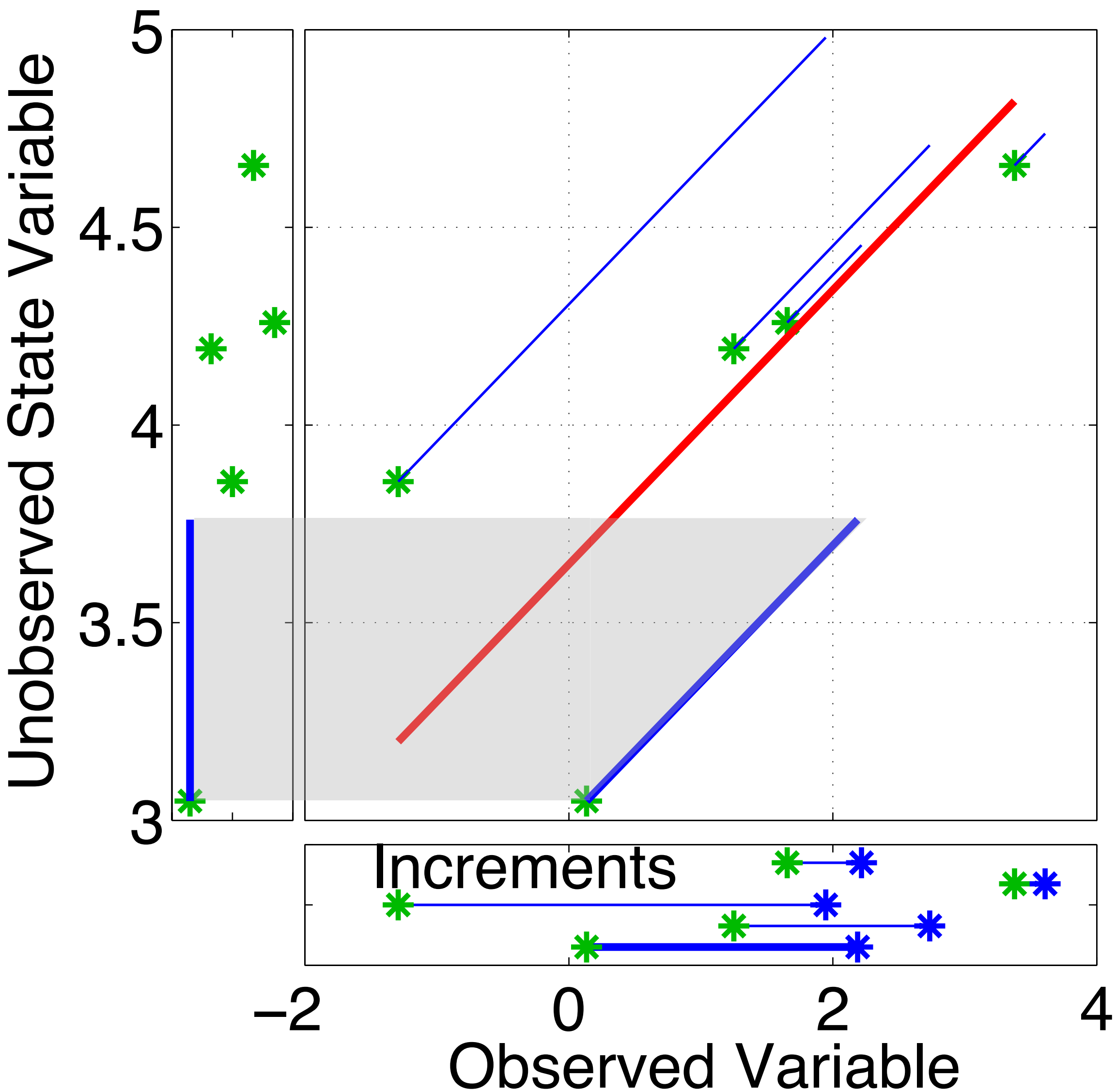


Have joint prior distribution of two variables.

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

Equivalent to first finding image of increment in joint space.

Ensemble filters: Updating additional prior state variables

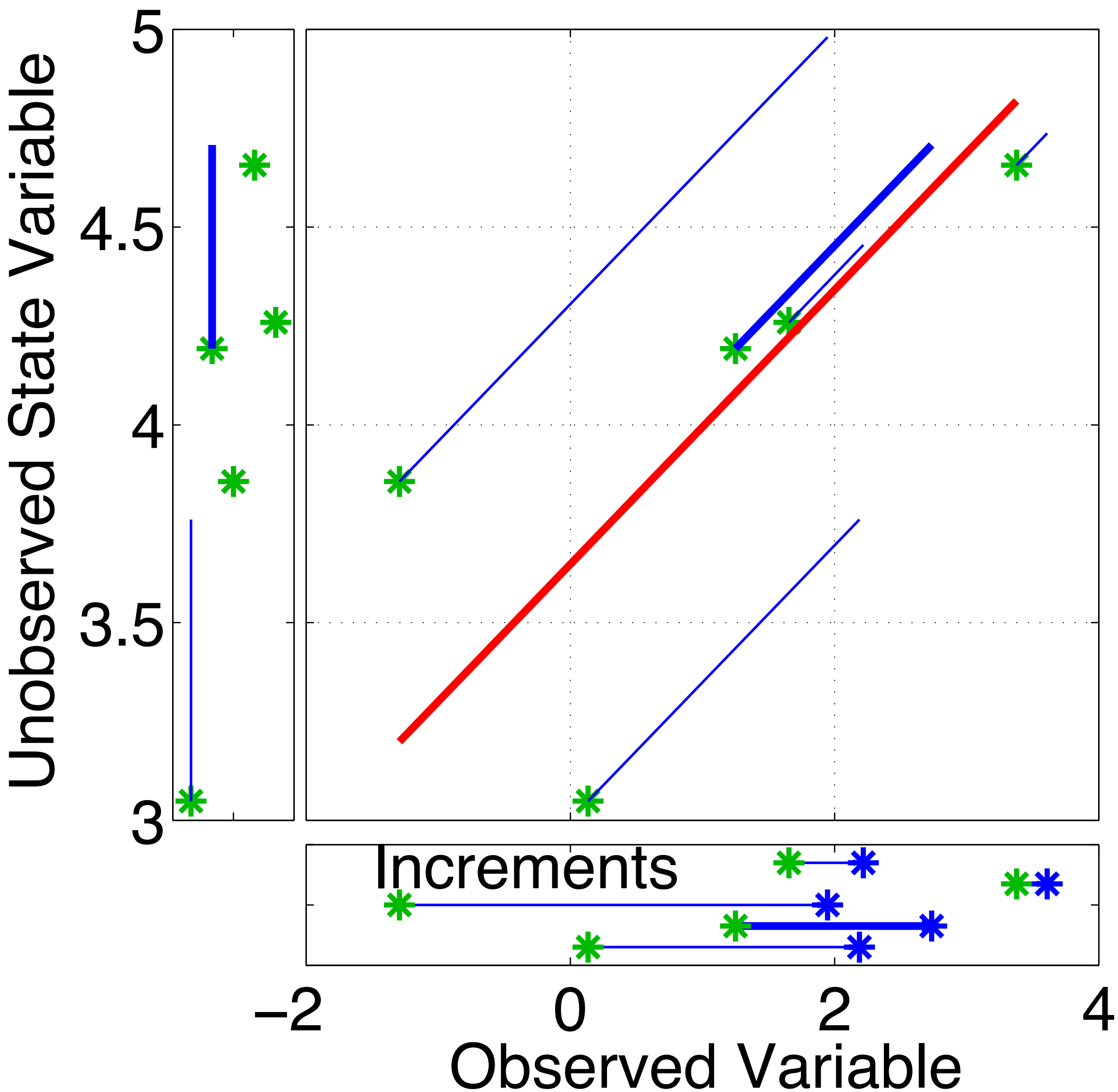


Have joint prior distribution of two variables.

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

Then projecting from joint space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

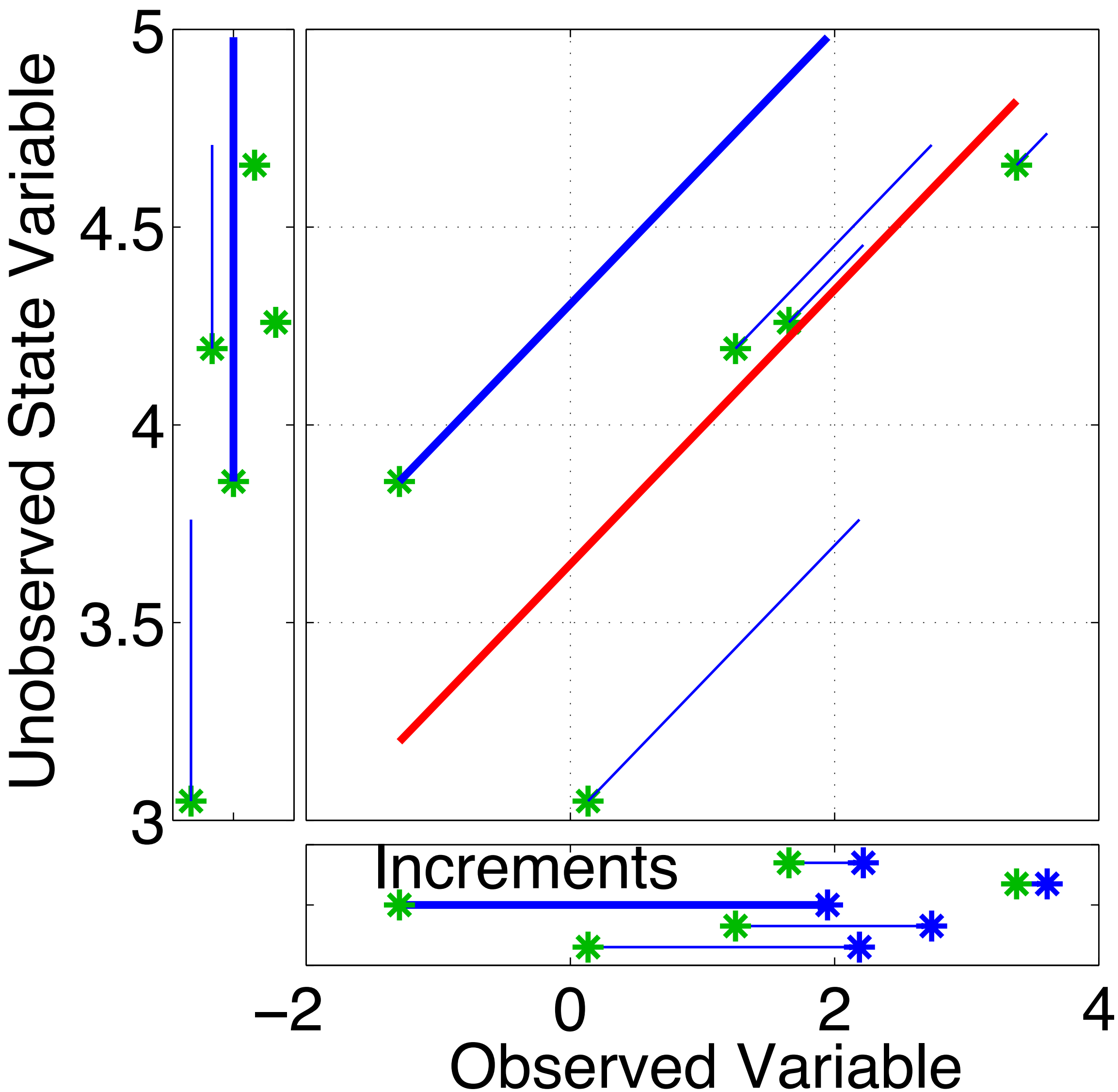


Have joint prior distribution of two variables.

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

Then projecting from joint space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

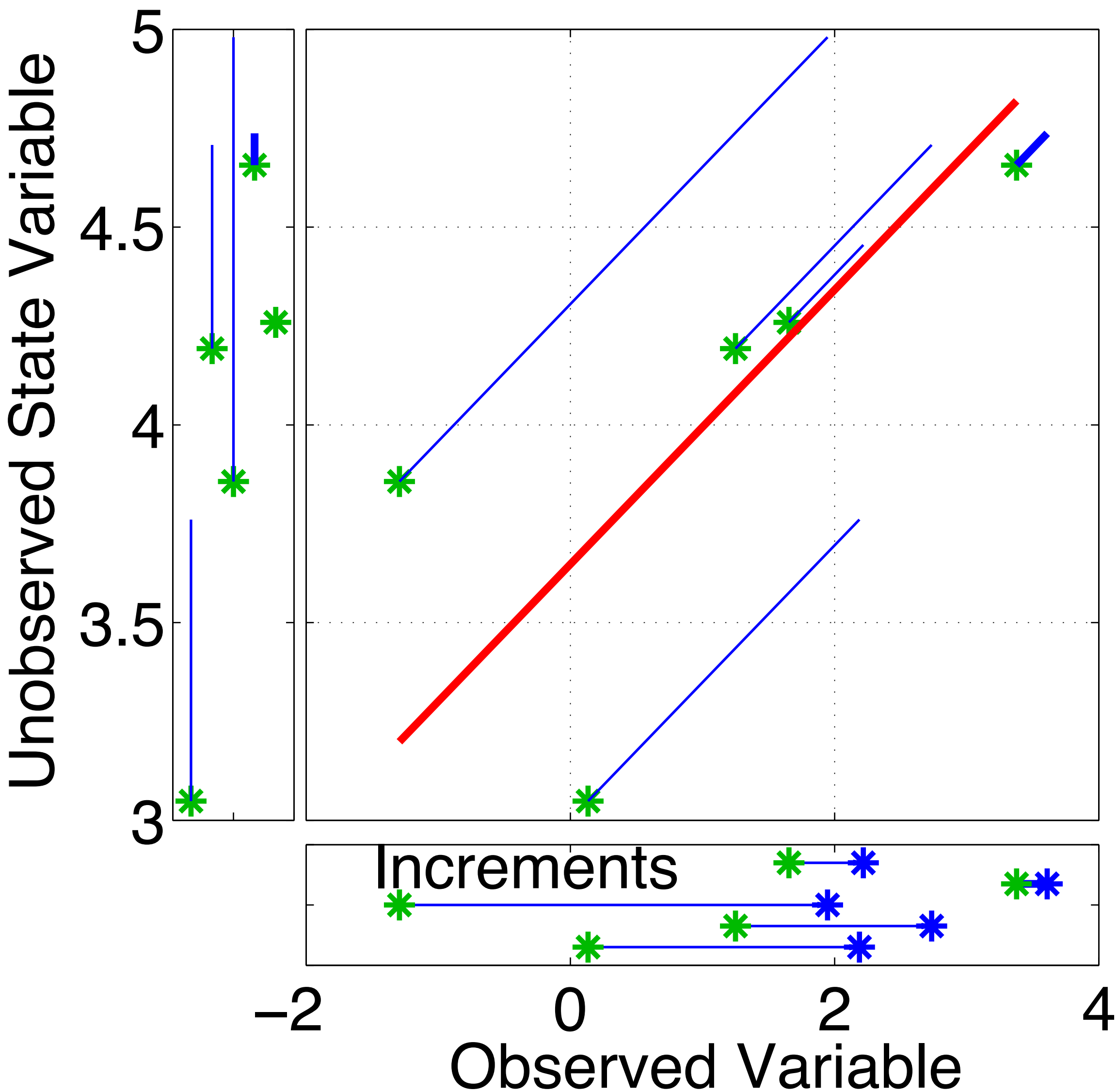


Have joint prior distribution of two variables.

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

Then projecting from joint space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

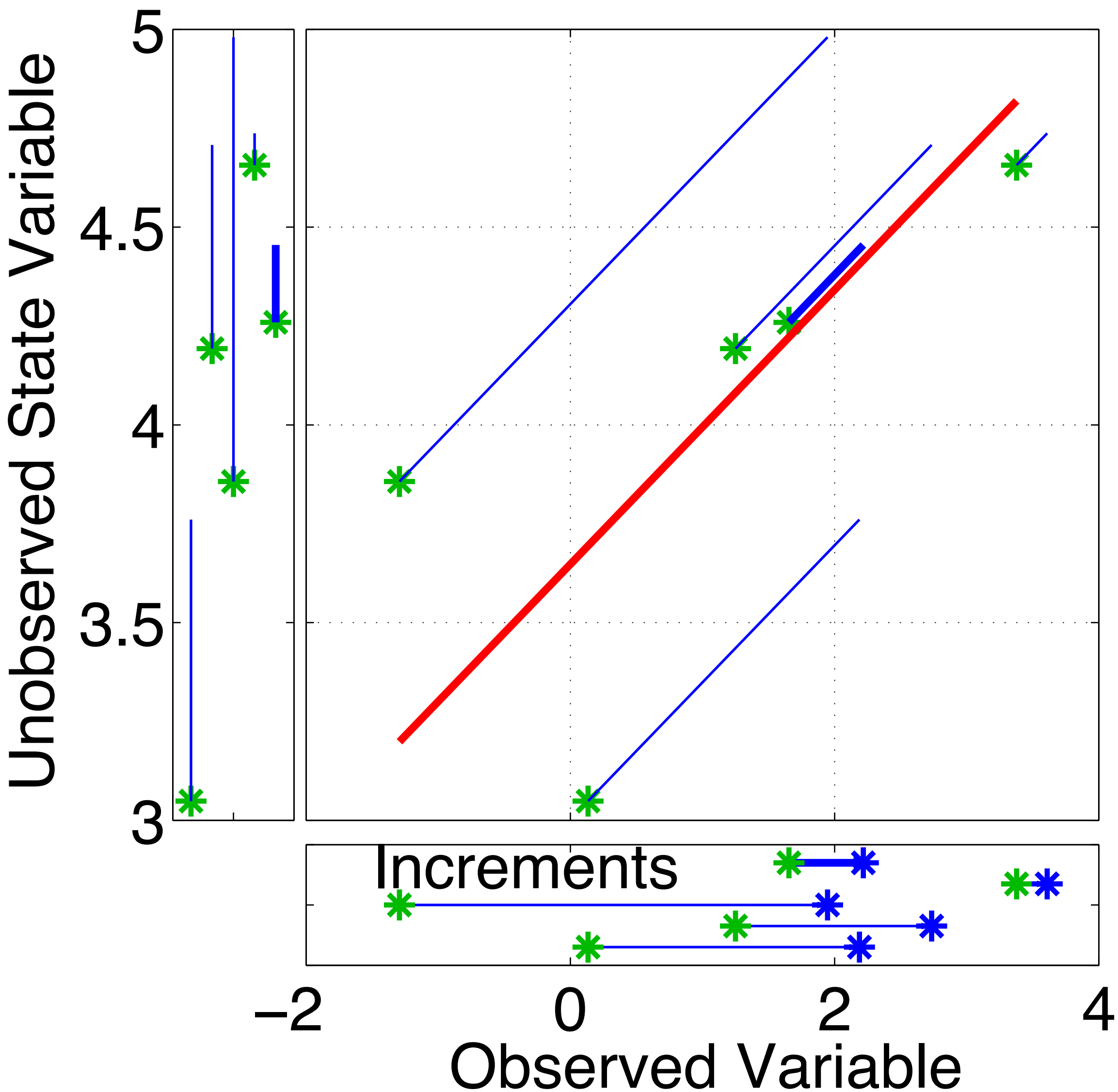


Have joint prior distribution of two variables.

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

Then projecting from joint space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

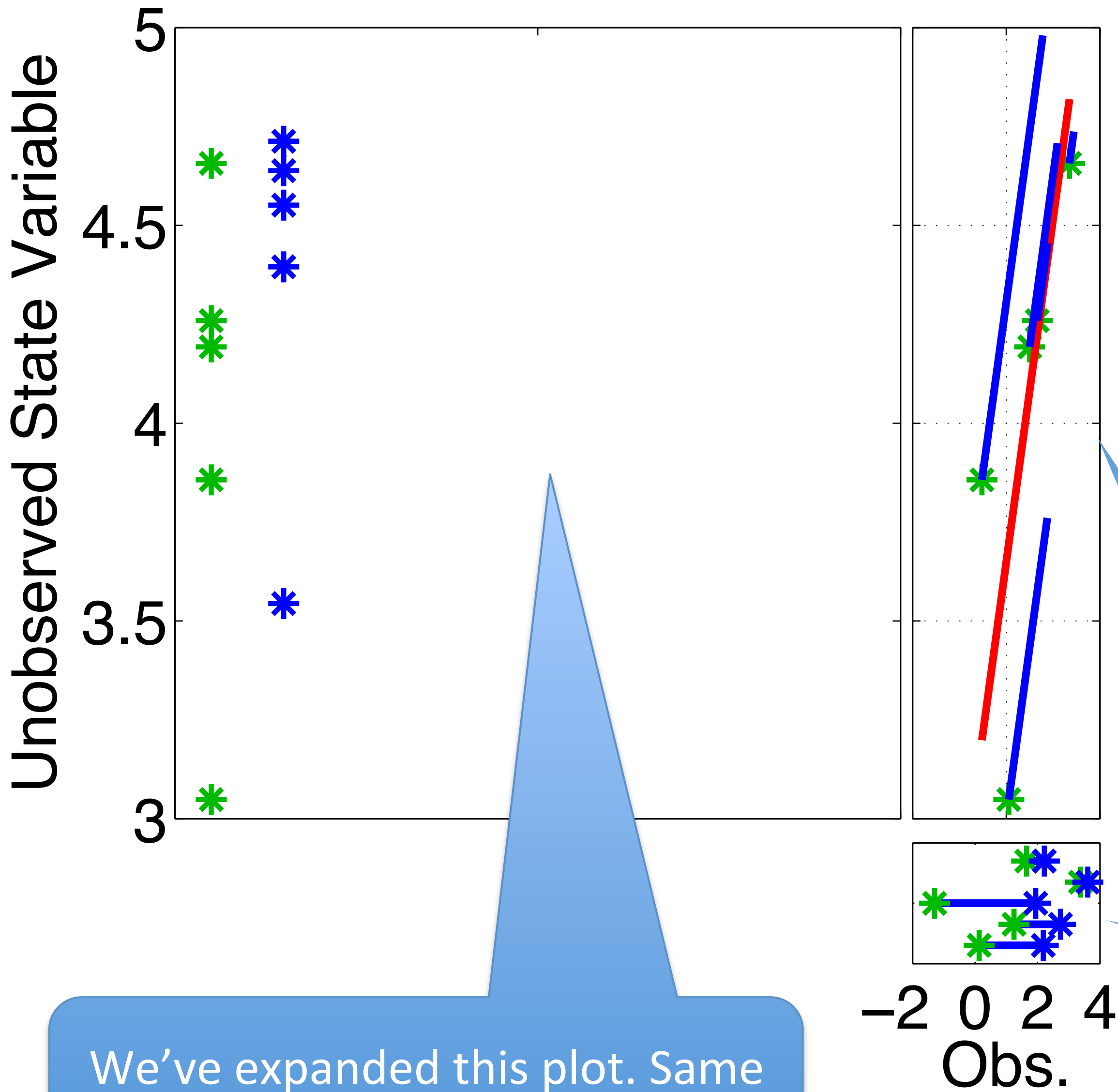


Have joint prior distribution of two variables.

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

Then projecting from joint space onto unobserved priors.

Ensemble filters: Updating additional prior state variables

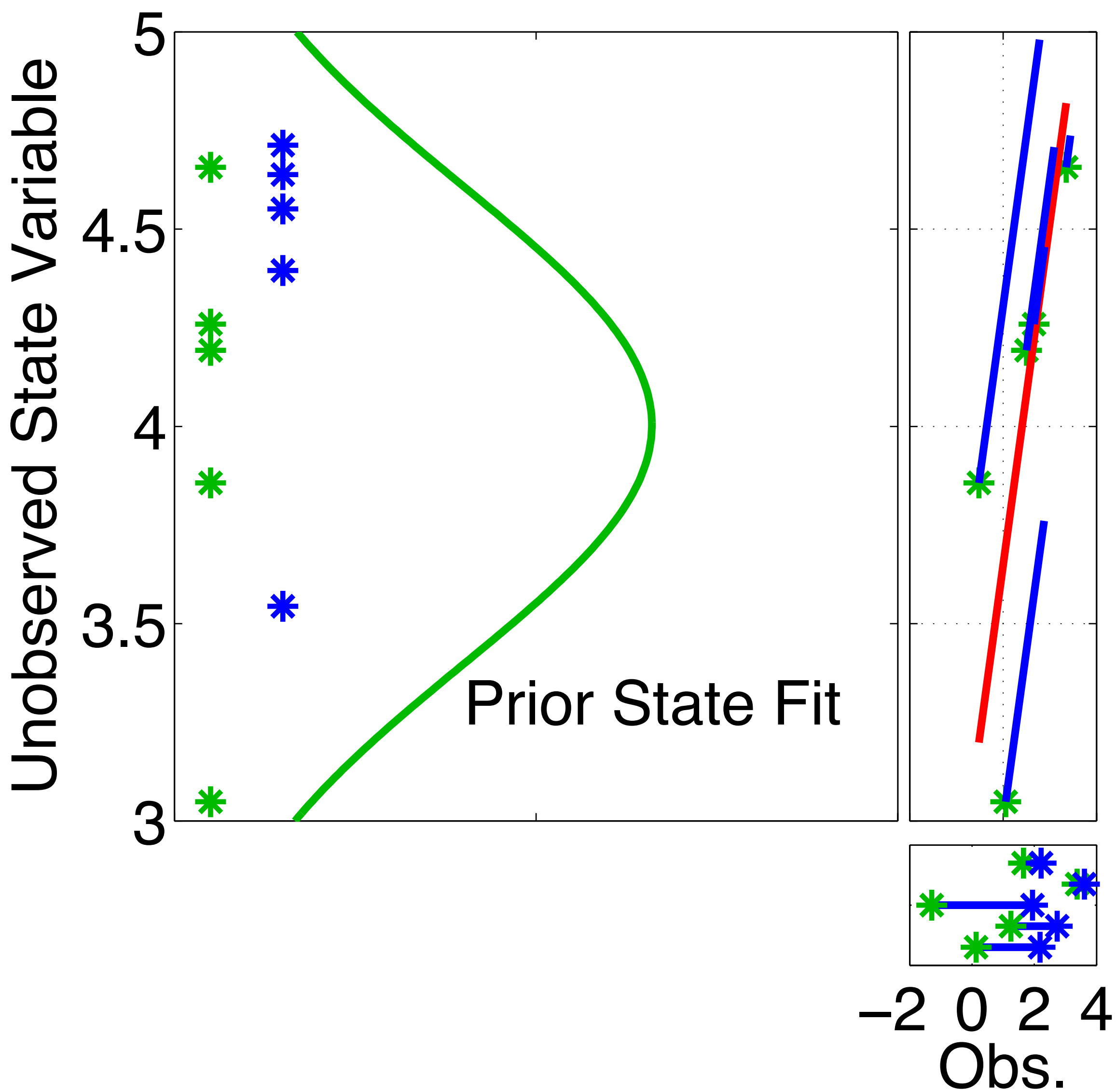


Now have an updated (posterior) ensemble for the unobserved variable.

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

Compressed these two.

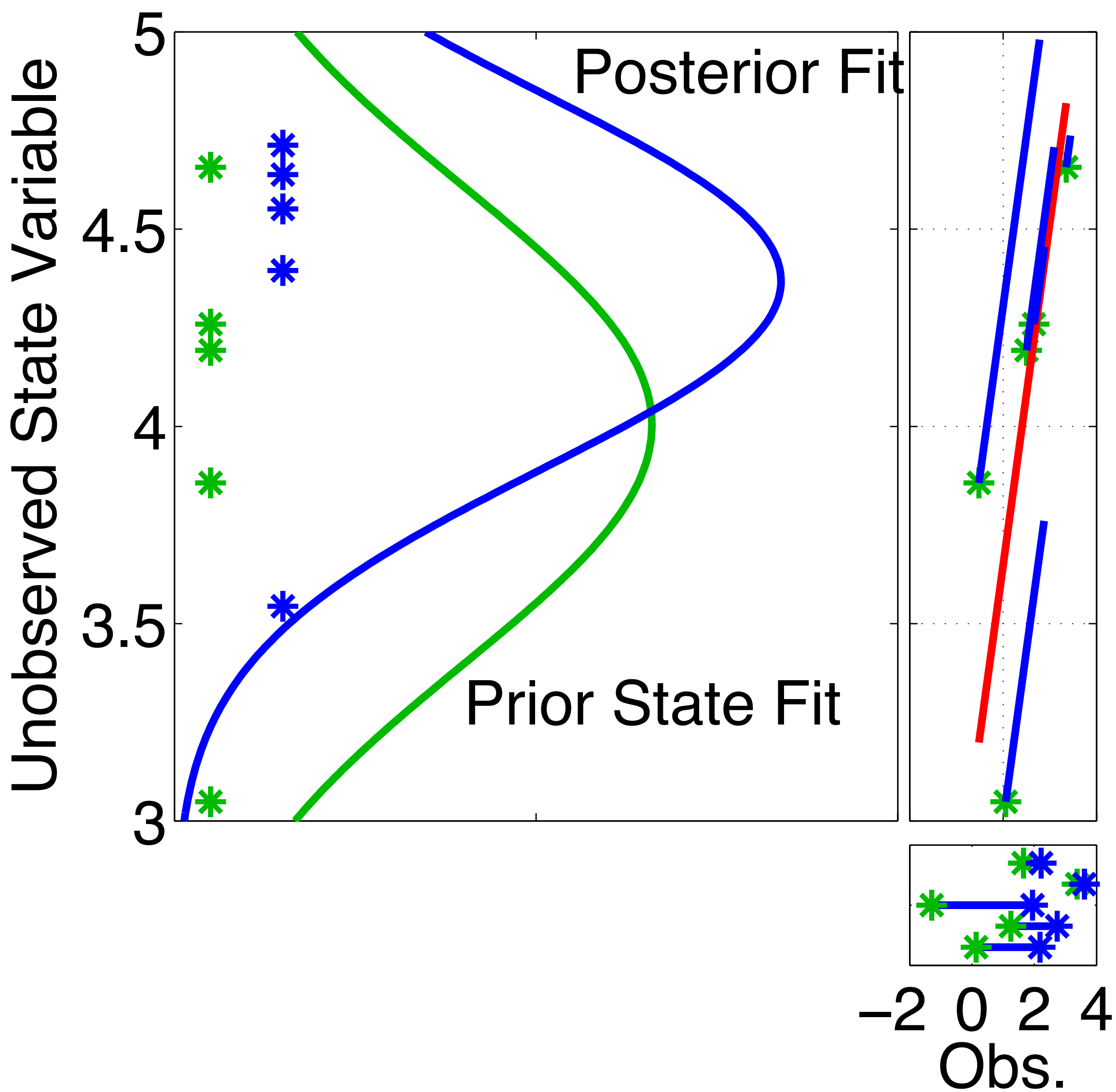
Ensemble filters: Updating additional prior state variables



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

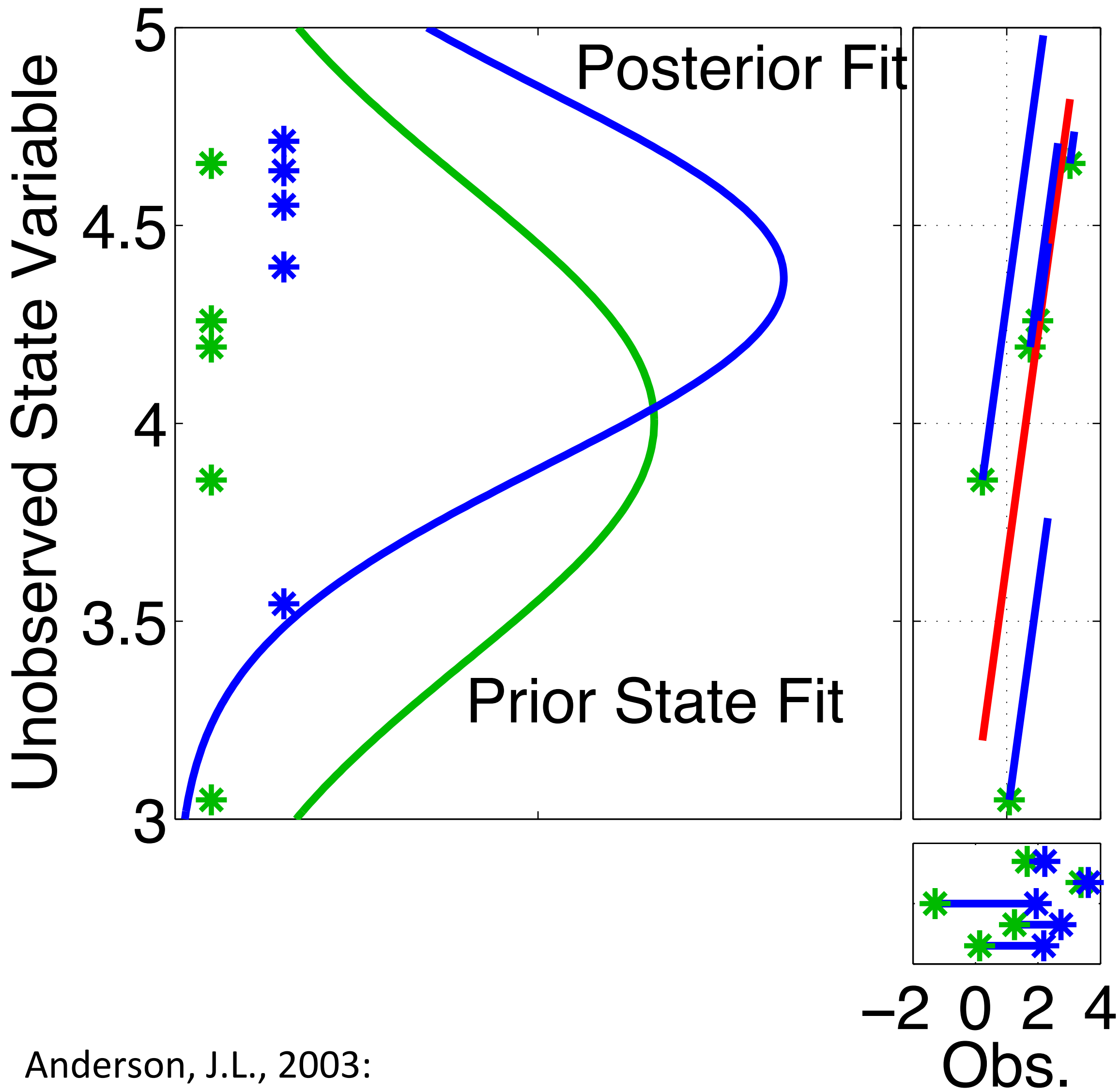


Now have an updated (posterior) ensemble for the unobserved variable.

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:

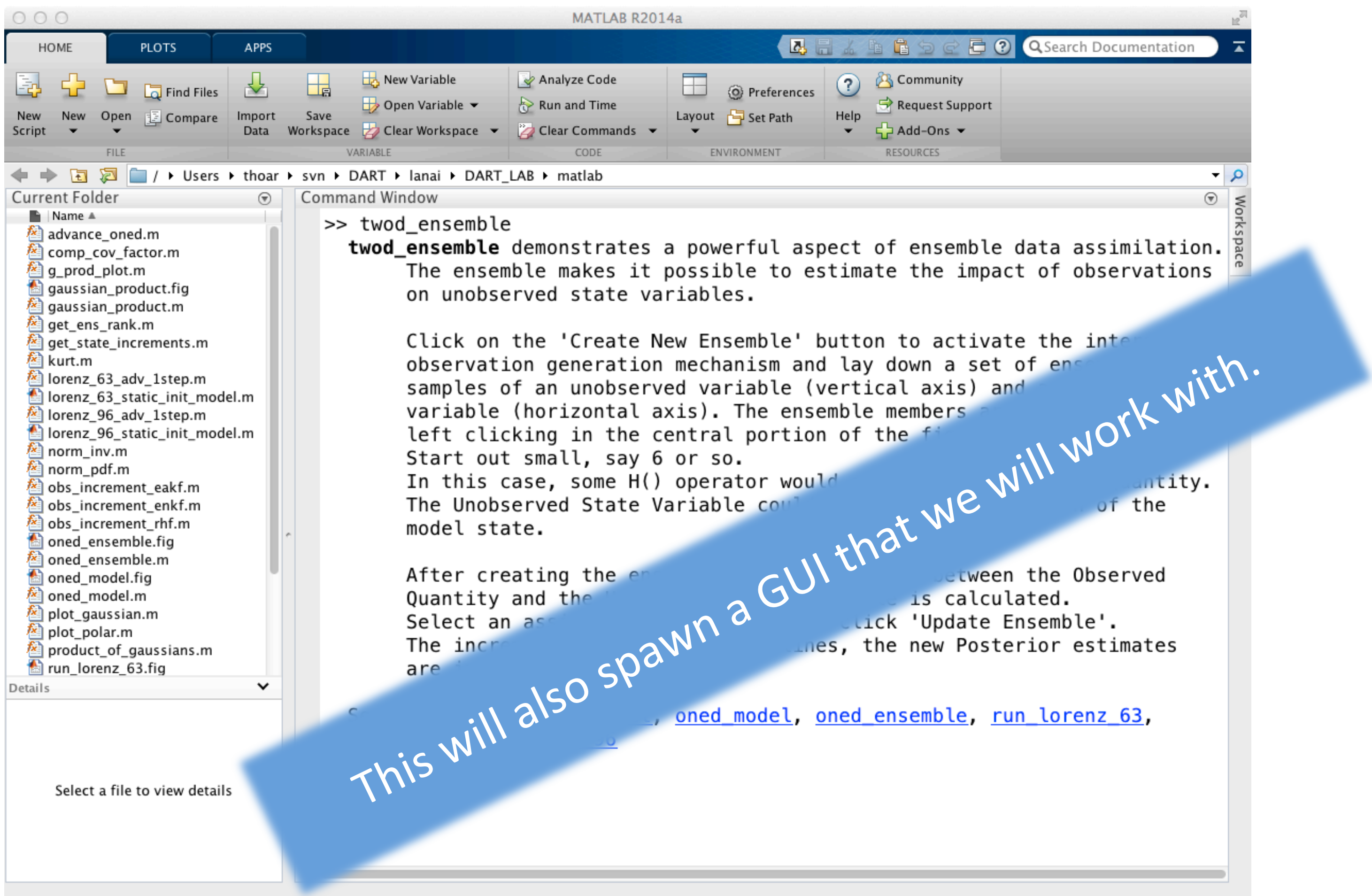
Since impact on unobserved variable is simply a linear regression, can do this INDEPENDENTLY for any number of unobserved variables!

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**

Purpose: Explore how an unobserved state variable is updated by an observation of another state variable.

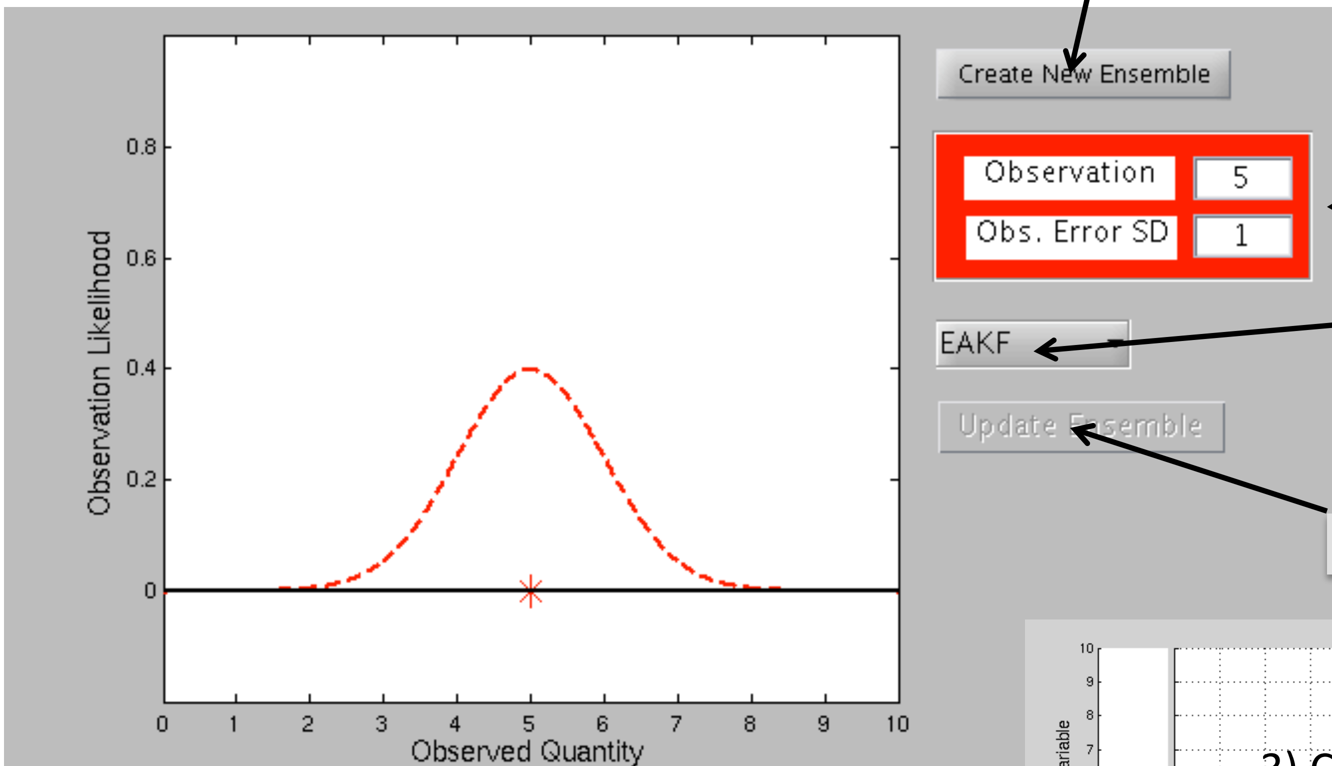


Matlab Hands-On: twod_ensemble

2) Click on

Create New Ensemble

1) change these if you want to.



Ignore the EAKF menu for now.

5) Click on

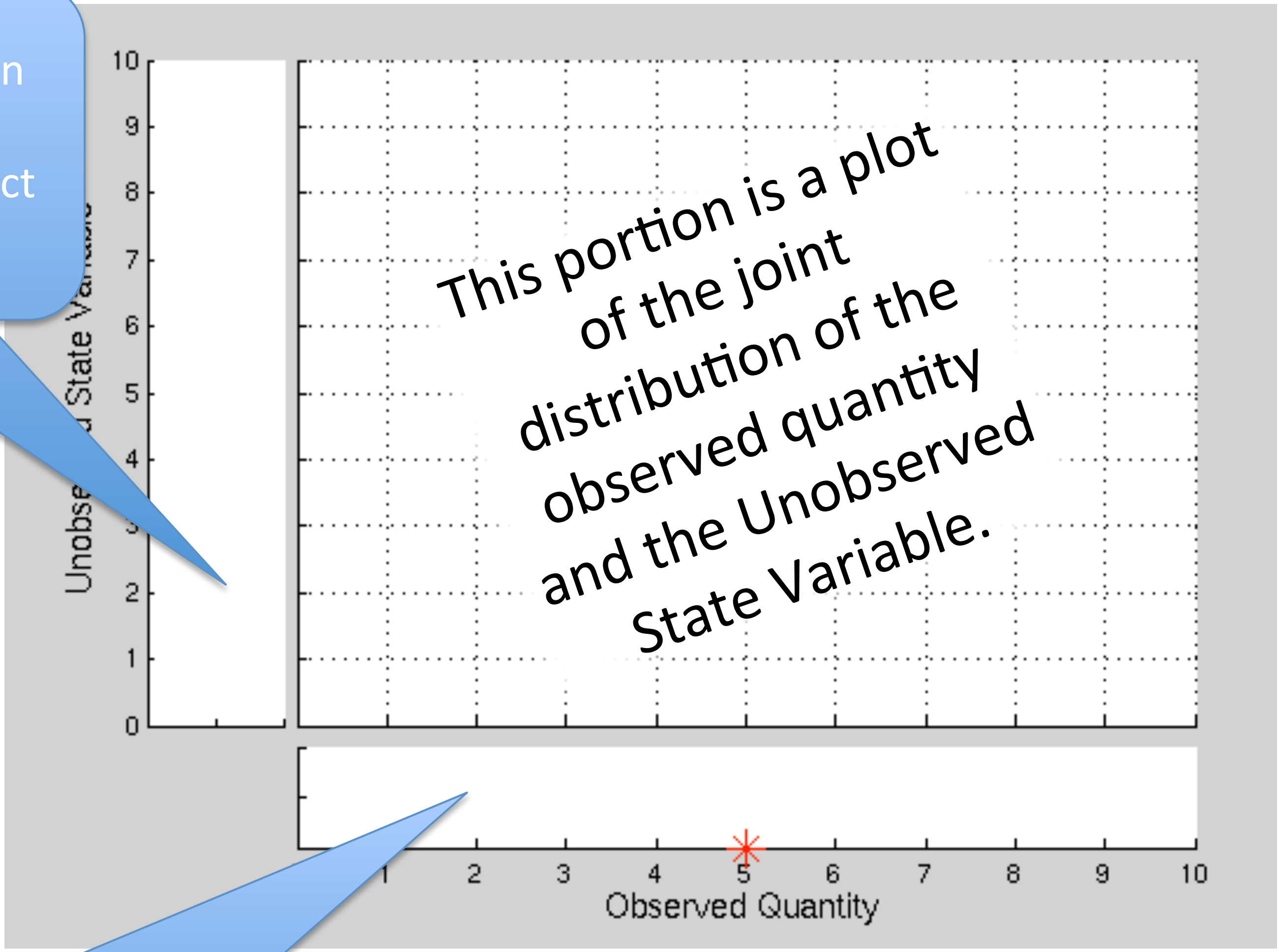
Update Ensemble

3) Click in here a few times. Anywhere!

4) Click outside the axis on the gray (anywhere) to finish defining the ensemble.

Matlab Hands-On: twod_ensemble

The marginal distribution for the unobserved variable. Will show impact of assimilation.



This portion is a plot of the joint distribution of the observed quantity and the Unobserved State Variable.

The marginal distribution for just the observed variable. Same information as corresponding marginal in GUI window.

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.

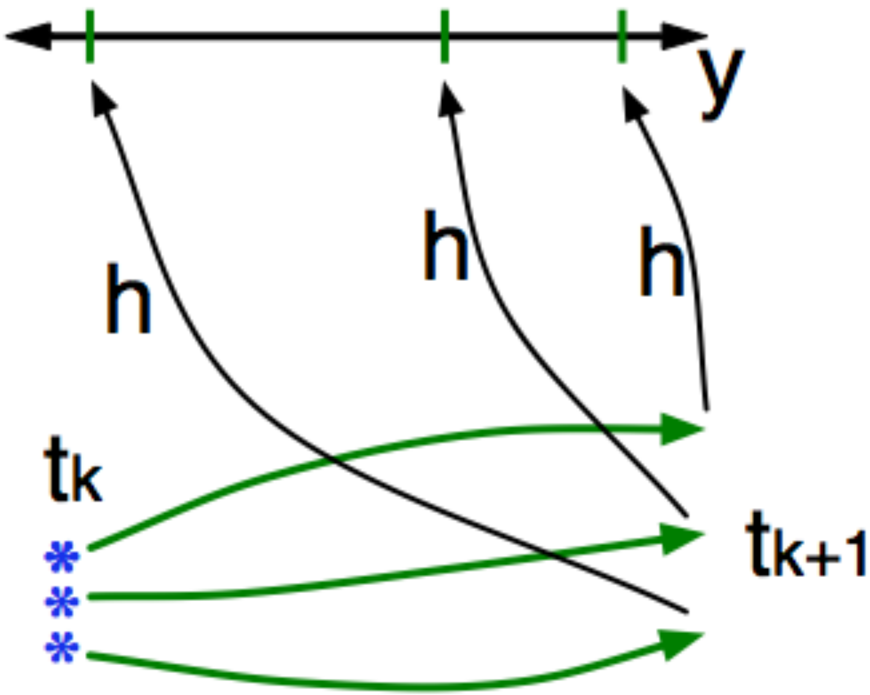
Schematic of an Ensemble Filter for Geophysical Data Assimilation

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

Ensemble state estimate after using previous observation (analysis)

Ensemble state at time of next observation (prior)

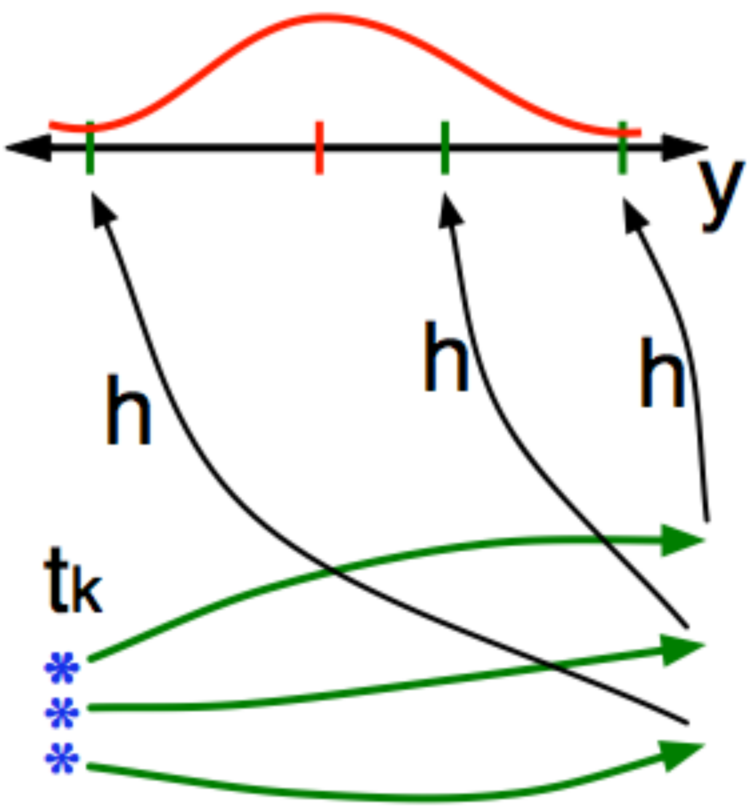
2. Get prior ensemble sample of observation, $y = h(x)$, by applying forward operator **h** to each ensemble member.



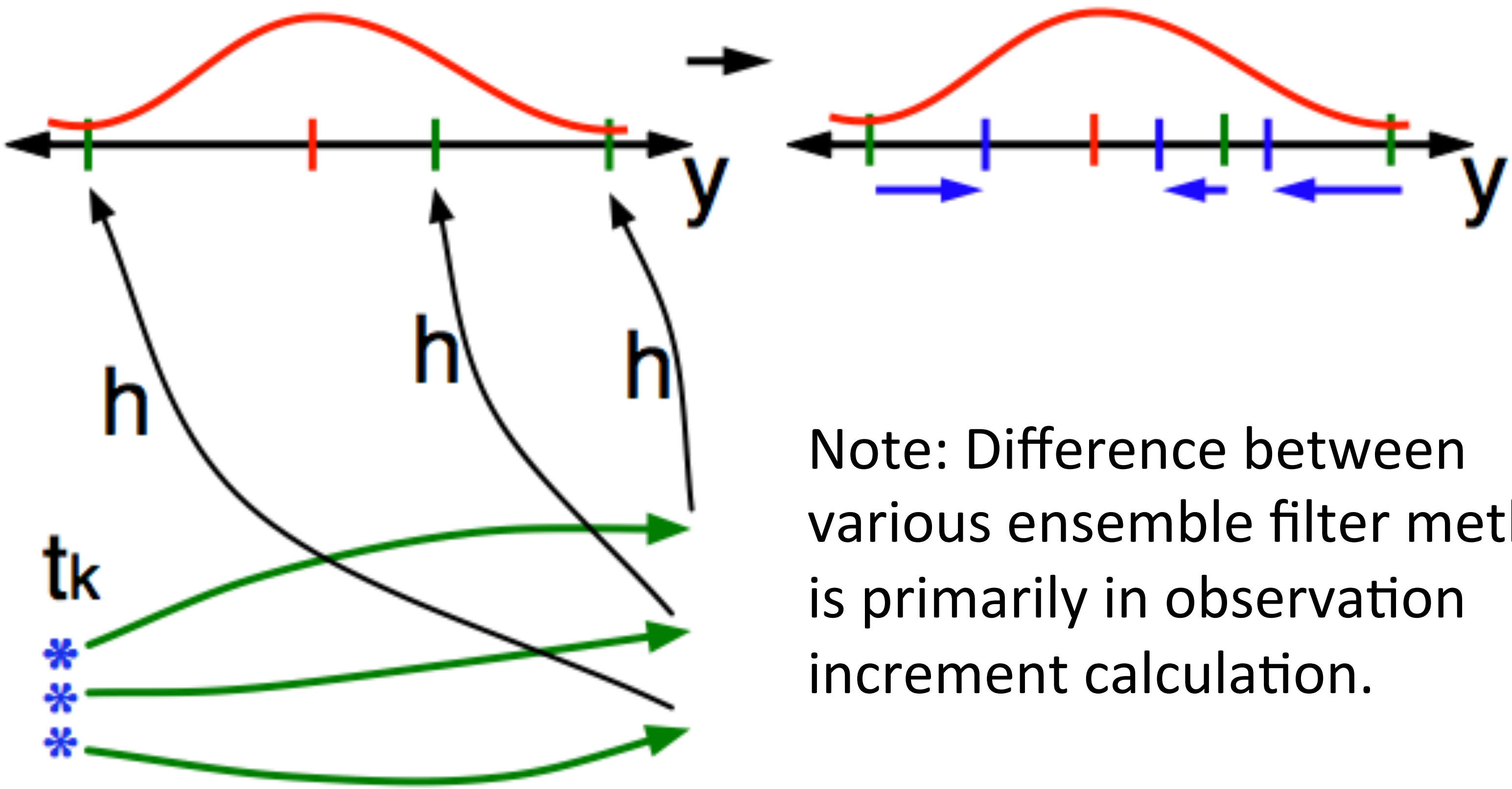
Theory: observations from instruments with uncorrelated errors can be done sequentially.

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 observational error distribution from observing system.

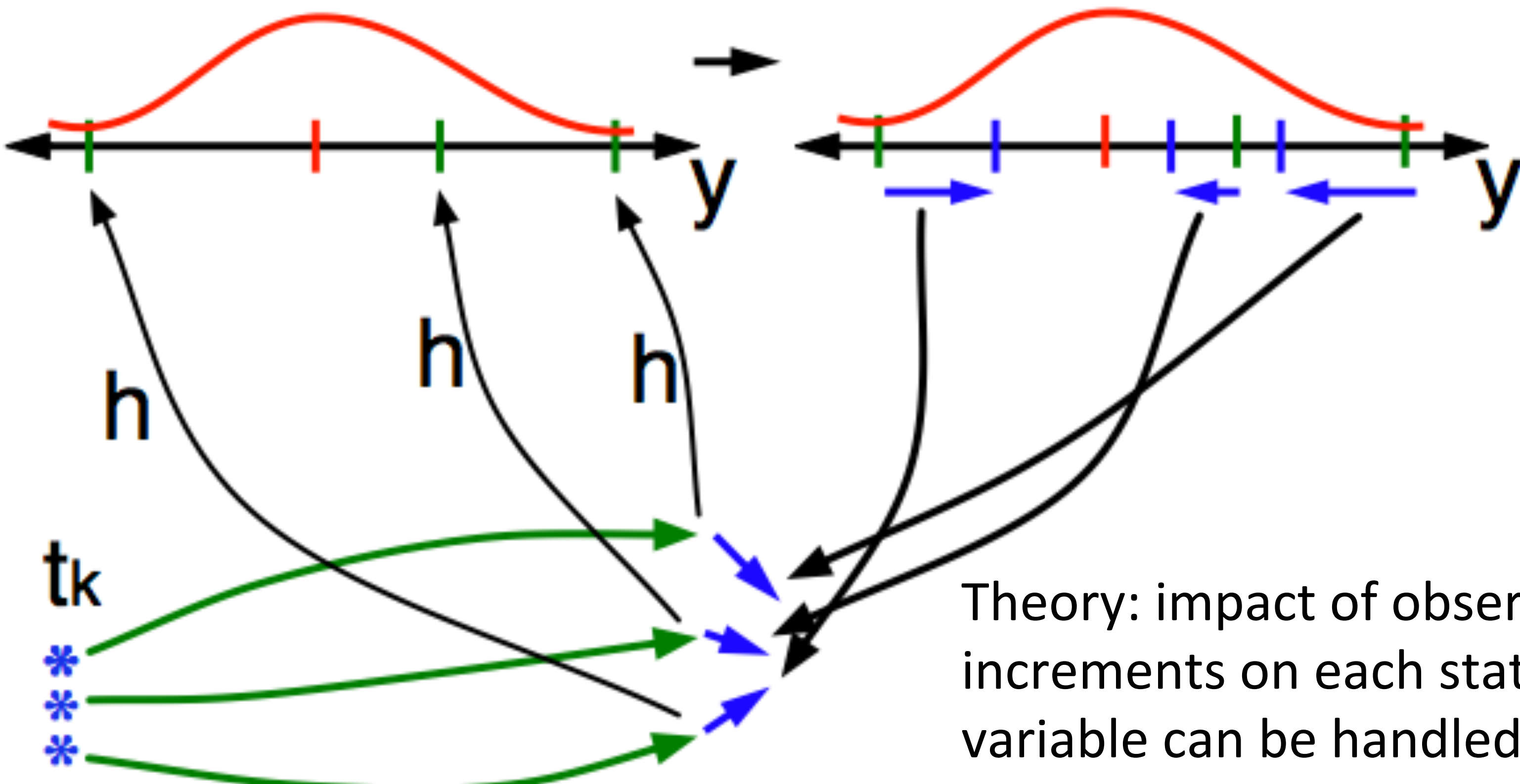


4. Find the increments for the prior observation ensemble (this is a scalar problem for uncorrelated observation errors).



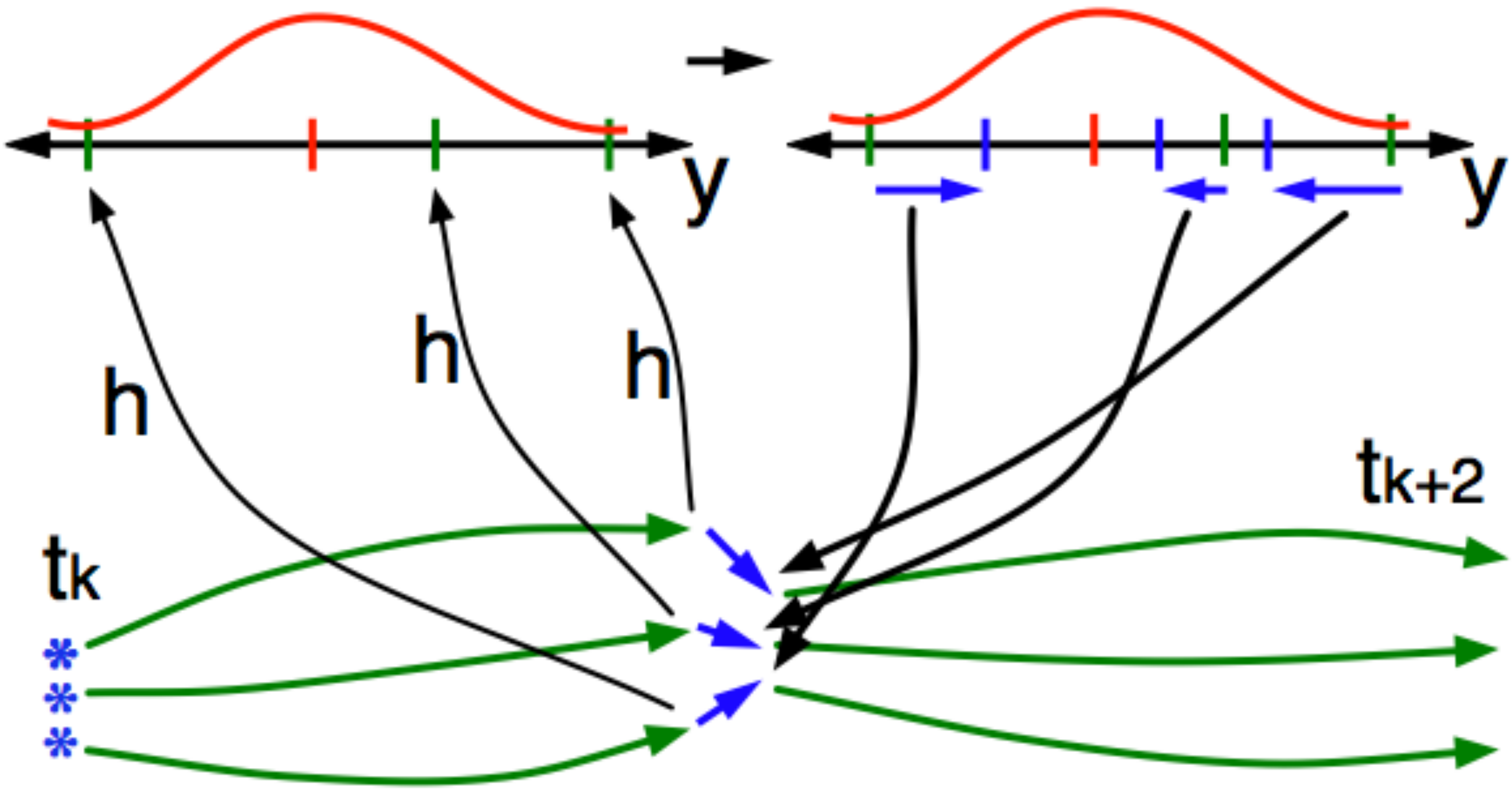
Note: Difference between various ensemble filter methods is primarily in observation increment calculation.

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



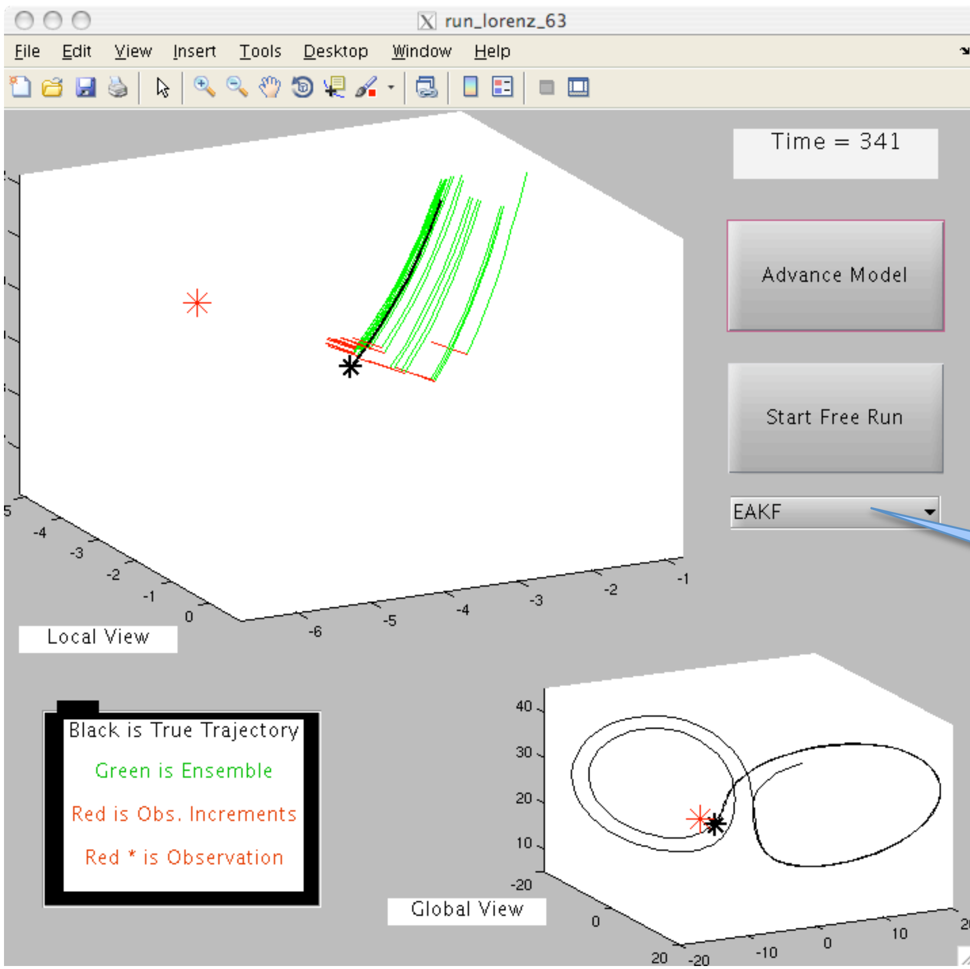
Theory: impact of observation increments on each state variable can be handled independently!

6. When all ensemble members for each state variable are updated, there is a new analysis. Integrate to time of next observation …



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.



1) Advance Model
Assimilate Obs

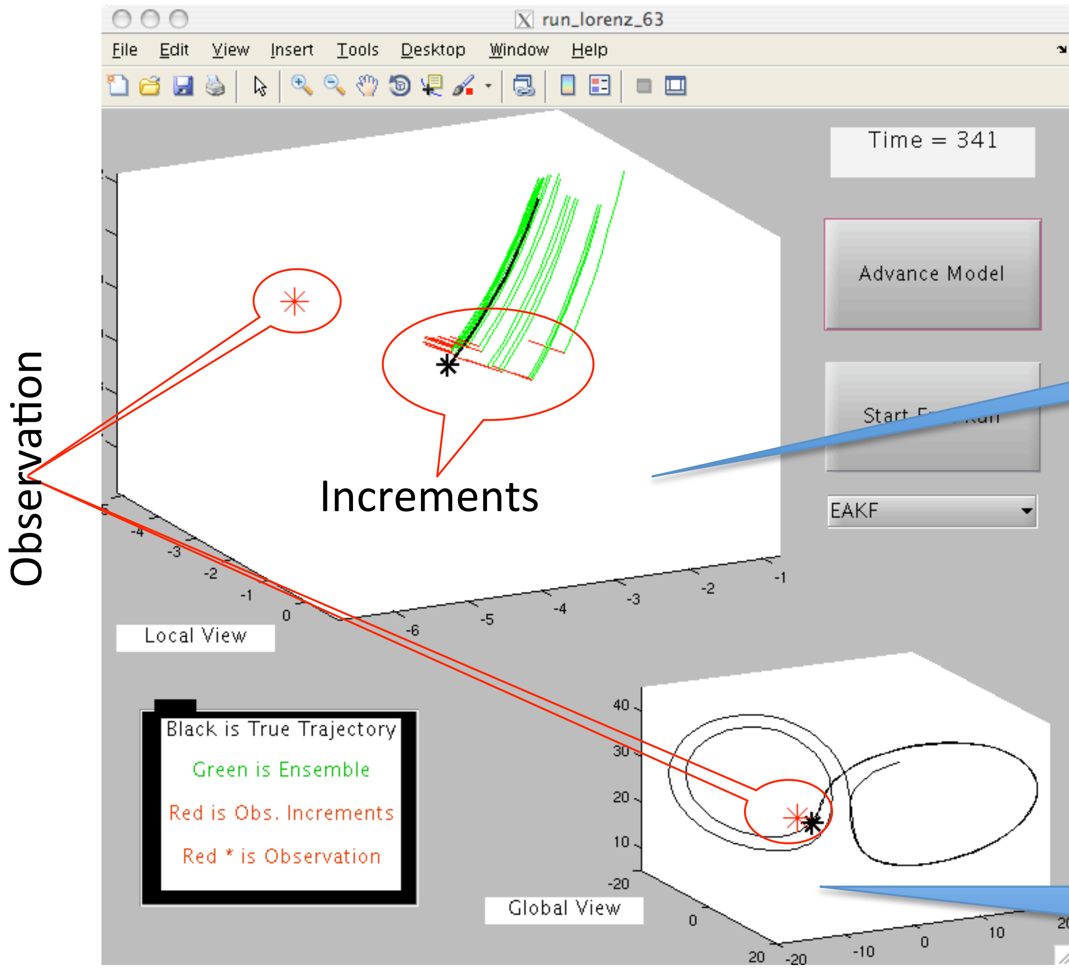
2) Start Free Run
Stop Free Run

Explore the filtering algorithms!

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.



'Local' domain, local timeframe

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.

Full domain, full timeframe.

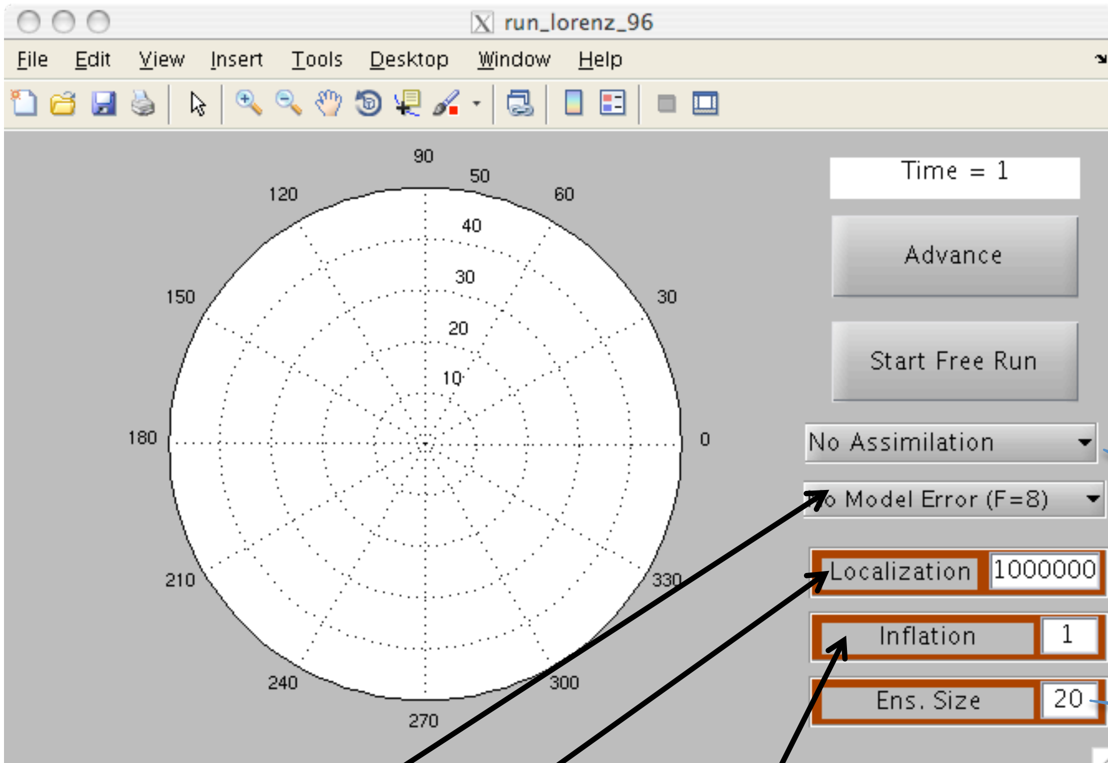
Matlab Hands-On: run_lorenz_63

Explorations:

- Select Start Free 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 free 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.



1) Advance
Assimilate Obs

2) Start Free Run
Stop Free Run

No Assimilation: free run
EAKF: deterministic

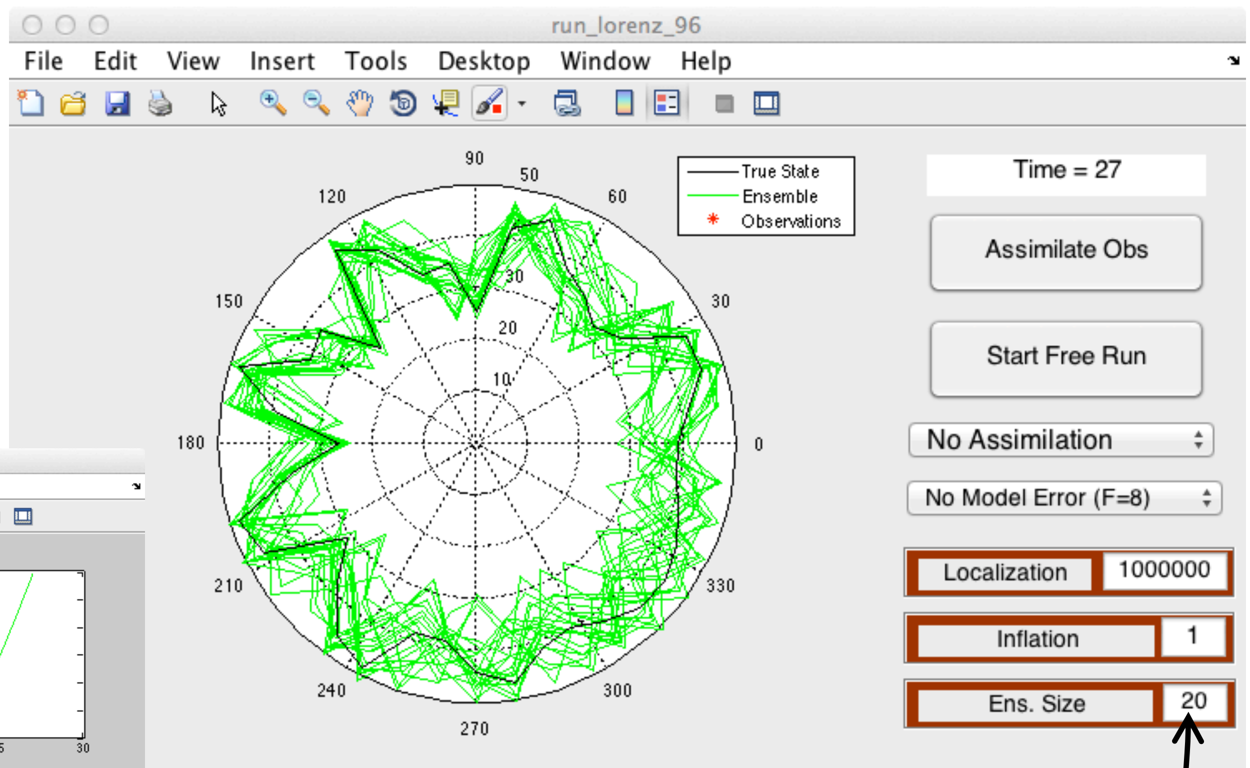
Choose your ensemble size.

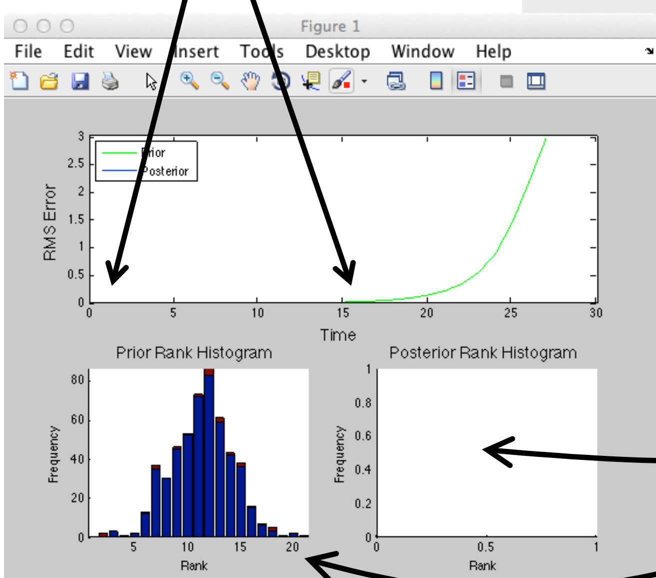
Model error, Localization, and Inflation all come later.

Matlab Hands-On: run_lorenz_96

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

Takes a while for the perturbations to manifest themselves.





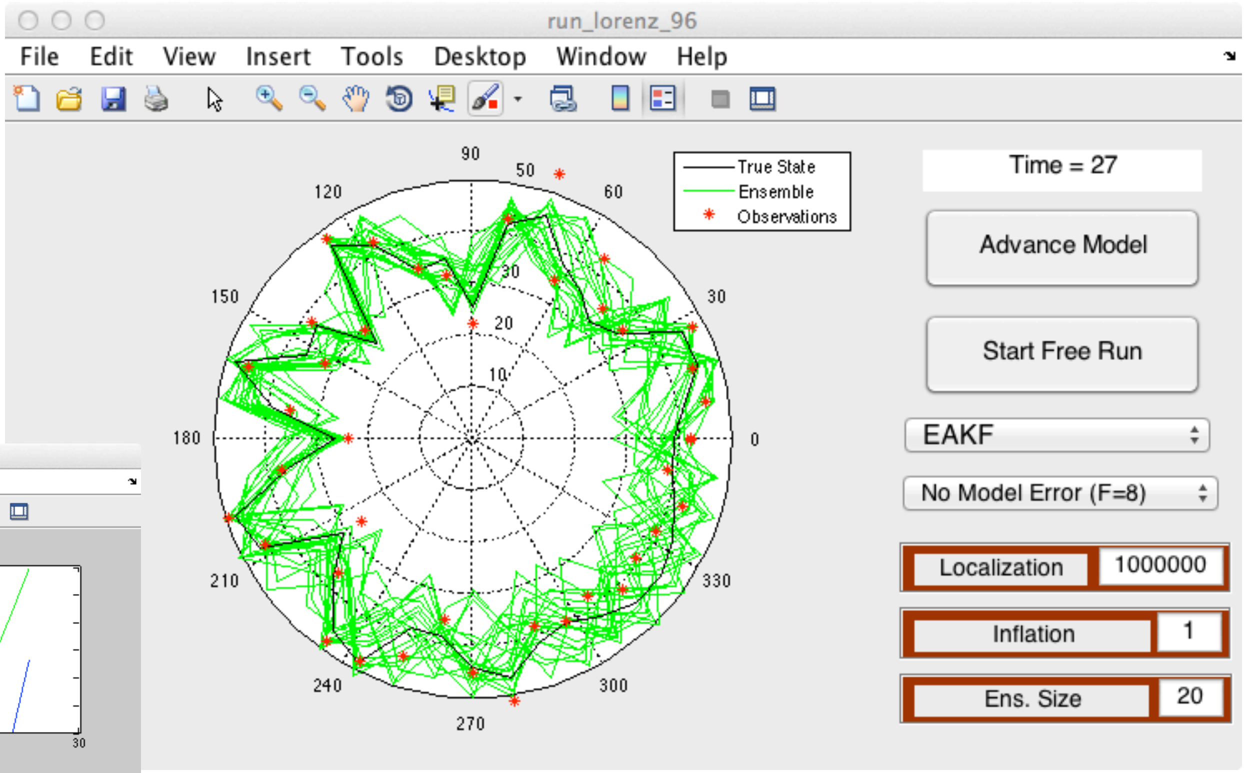
No such thing as posterior in a free run!

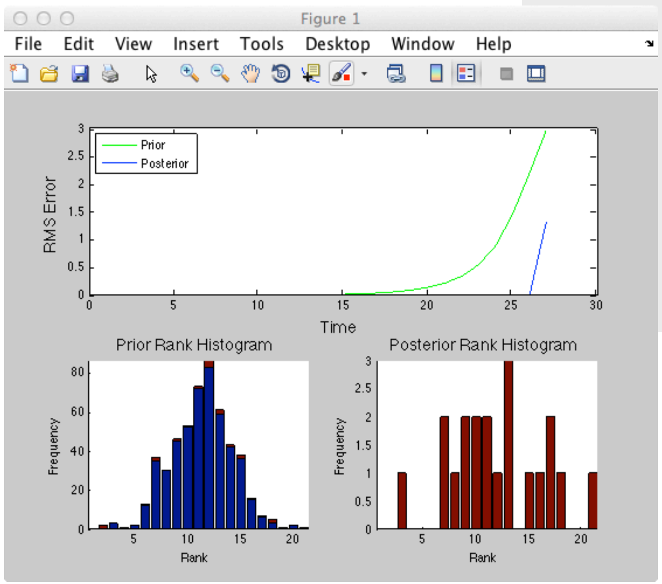
Ensemble size is 20, so there are 21 bins in the rank histogram.

Matlab Hands-On: run_lorenz_96

1) Stop the free run after some time.
2) Turn on the EAKF
3) Advance model … once!
4) Figure it out!

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.

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.
- Try adding inflation (maybe 1.4) and repeat.