

D
a
t
a

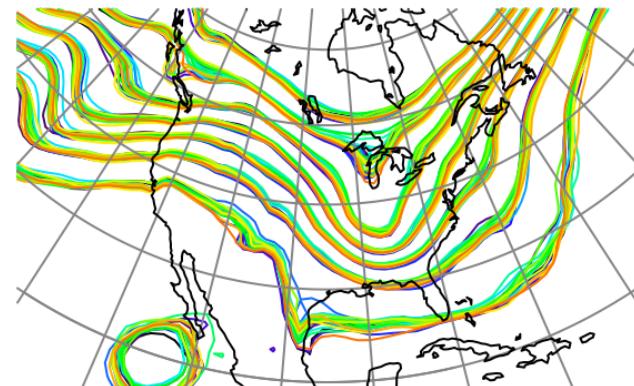
A
s
s
i
m
i
l
o
r
t

R
e
s
e
a
r
c
h

T
e
s
t
b
e
d



DART_LAB Tutorial Section 1: Ensemble Data Assimilation Concepts in 1D



©UCAR 2014

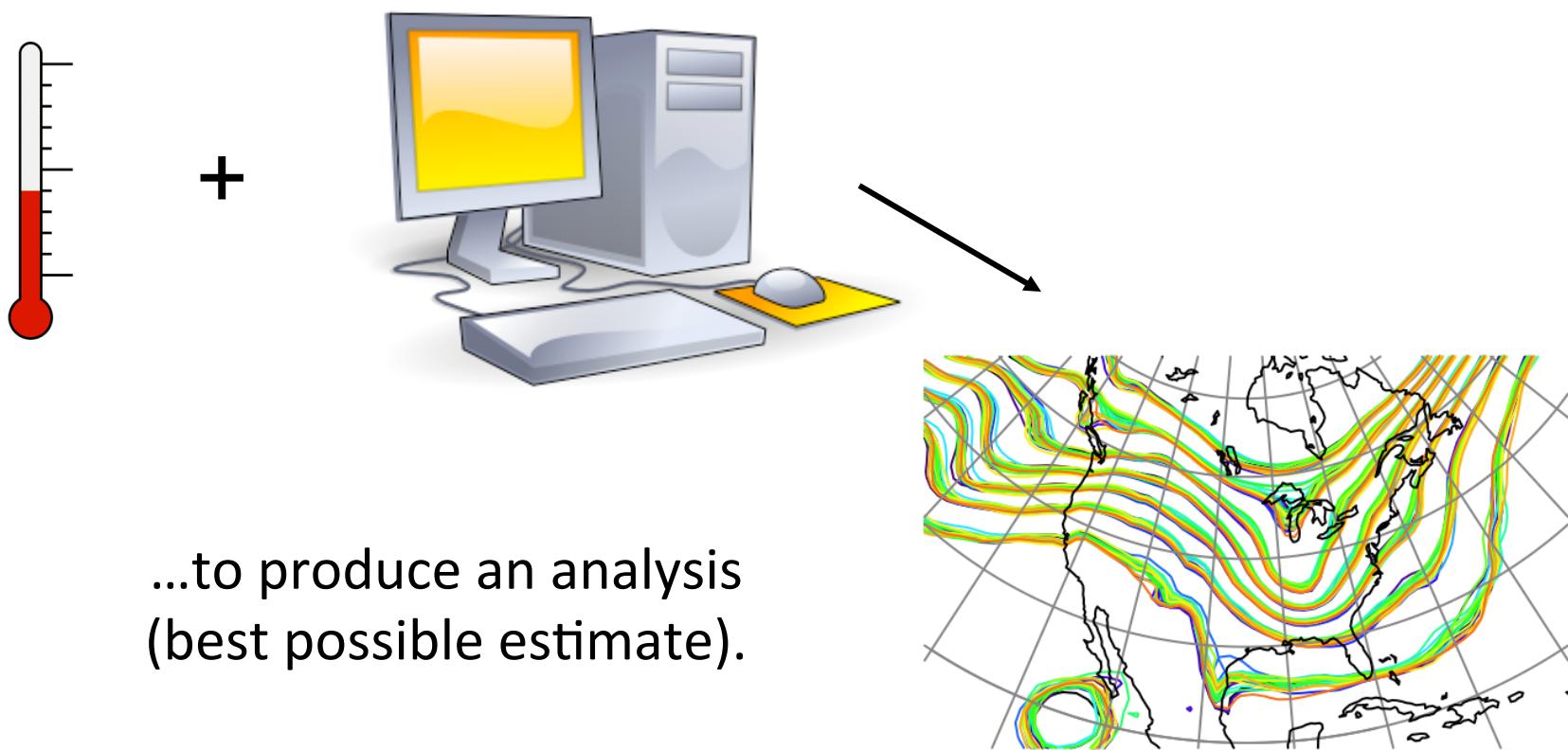
The National Center for Atmospheric Research is sponsored by the National Science Foundation. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

NCAR | National Center for
UCAR Atmospheric Research



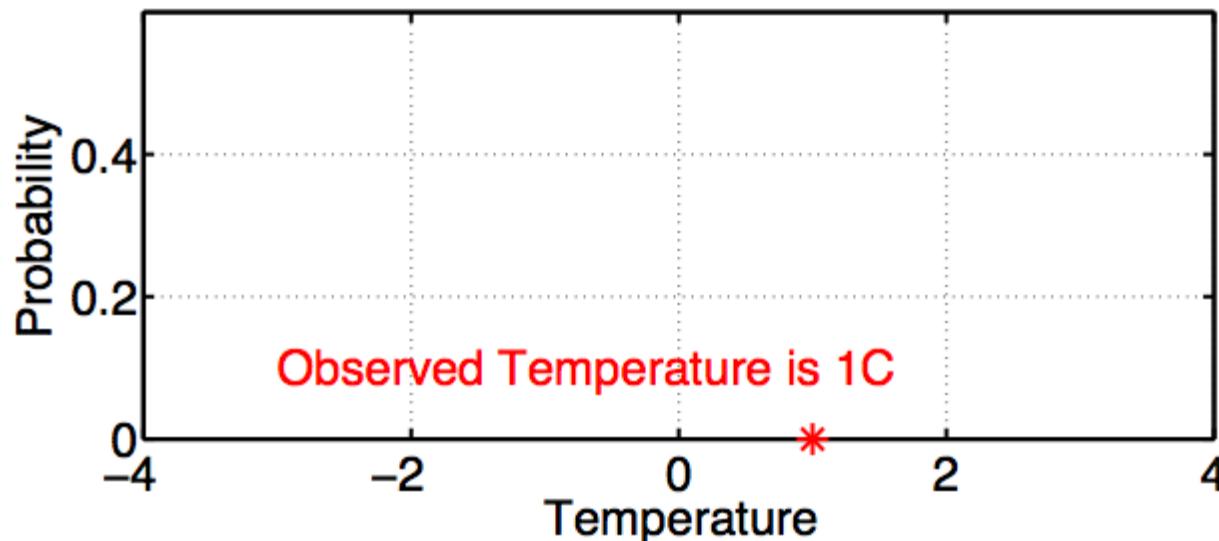
What is Data Assimilation?

Observations combined with a Model forecast ...



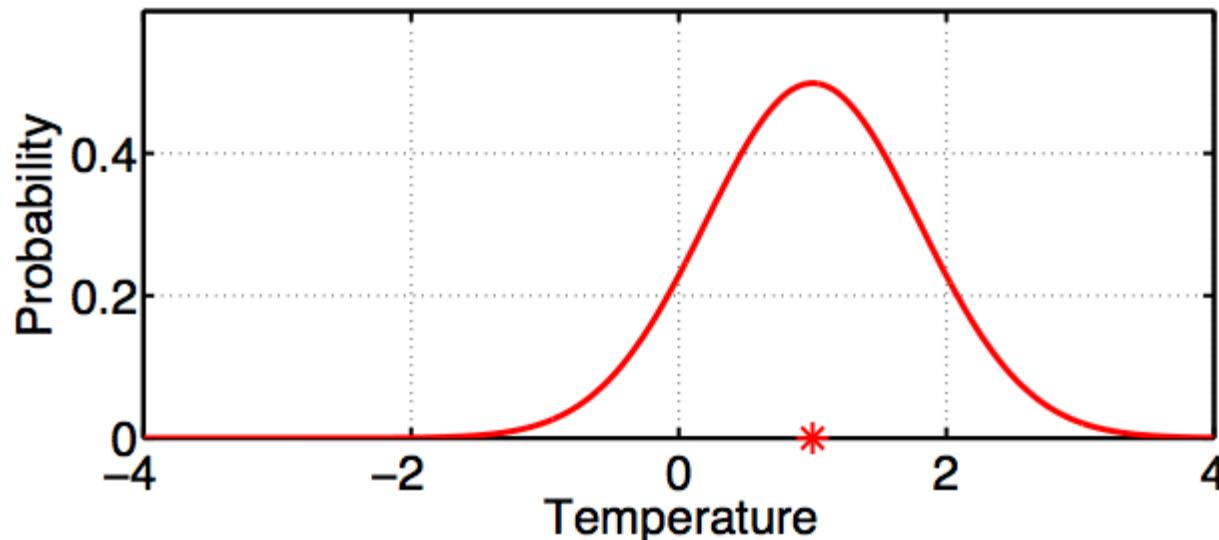
Example: Estimating the Temperature Outside

An observation has a value (*),



Example: Estimating the Temperature Outside

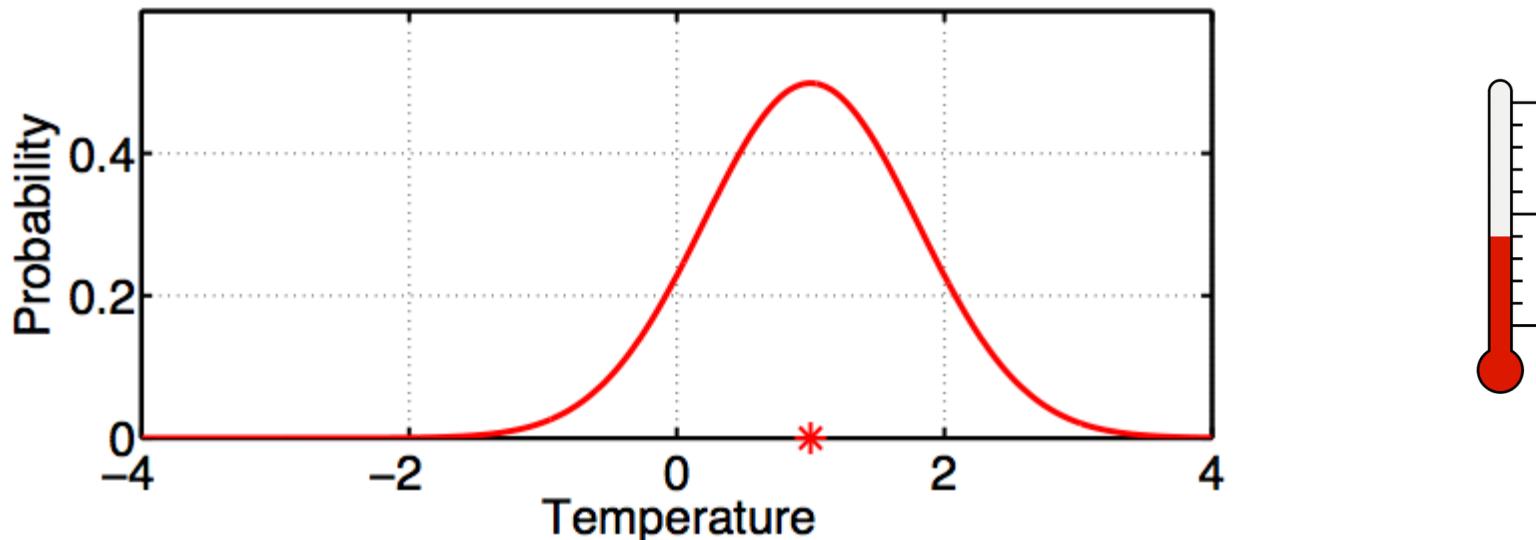
An observation has a value (*),



and an error distribution (red curve) that is associated with the instrument.

Example: Estimating the Temperature Outside

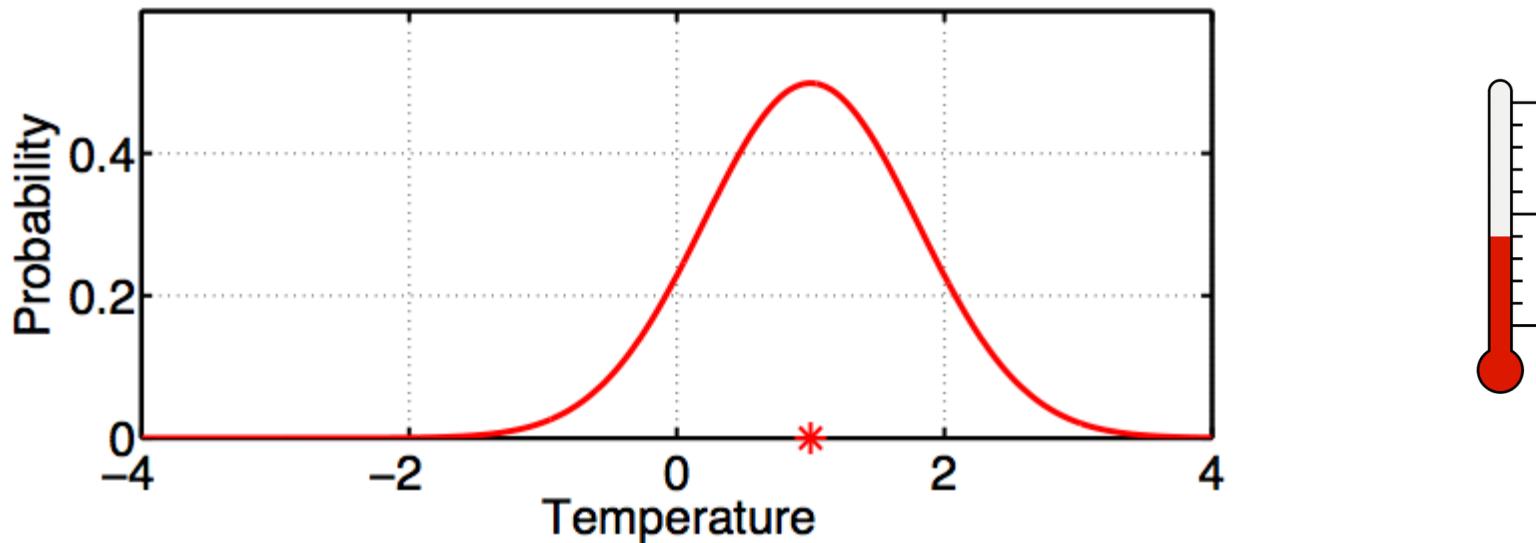
Thermometer outside measures 1°C .



Instrument builder says thermometer is
unbiased with $\pm 0.8^{\circ}\text{C}$ gaussian error.

Example: Estimating the Temperature Outside

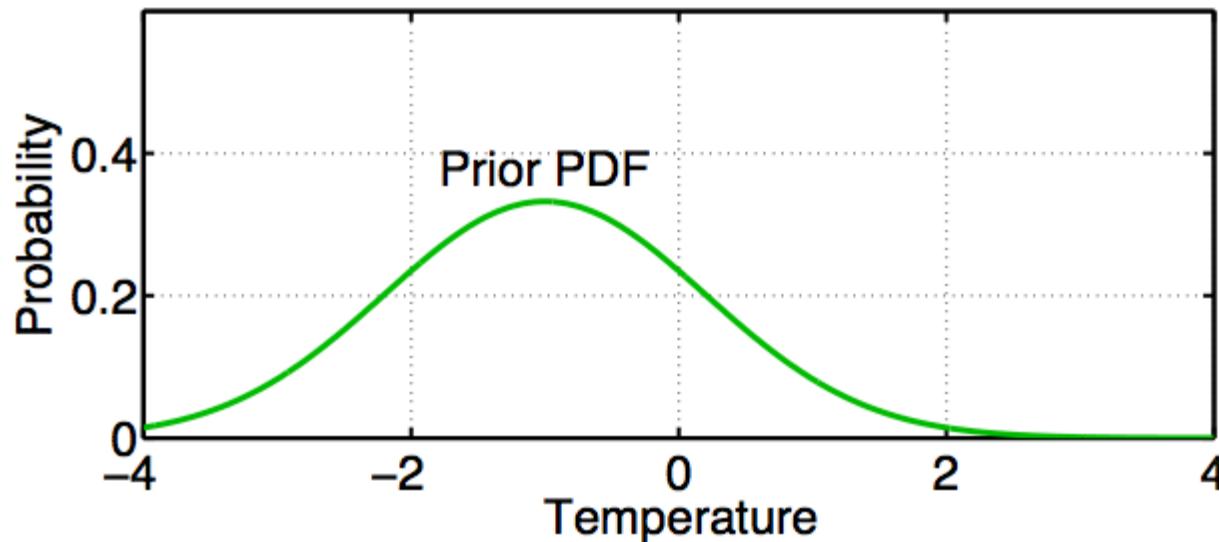
Thermometer outside measures 1°C .



The red plot is $P(T | T_0)$;
probability of temperature *given* that T_o was observed.

Example: Estimating the Temperature Outside

We also have a prior estimate of temperature.



The green curve is $P(T | C)$;
probability of temperature given all available prior information C .

Example: Estimating the Temperature Outside

Prior information C can include:

1. Observations of things besides T ;
2. Model forecast made using observations at earlier times;
3. *a priori* physical constraints ($T > -273.15^\circ C$);
4. Climatological constraints ($-30^\circ C < T < 40^\circ C$).

Combining the Prior Estimate and Observation

Bayes
Theorem:

Likelihood: Probability that T_o is observed if T is true value and given prior information C .

$$P(T|T_o, C) = \frac{P(T_o|T, C)P(T|C)}{P(T_o|C)}$$

Posterior: Probability of T given observations and Prior.
Also called **update** or **analysis**.

Combining the Prior Estimate and Observation

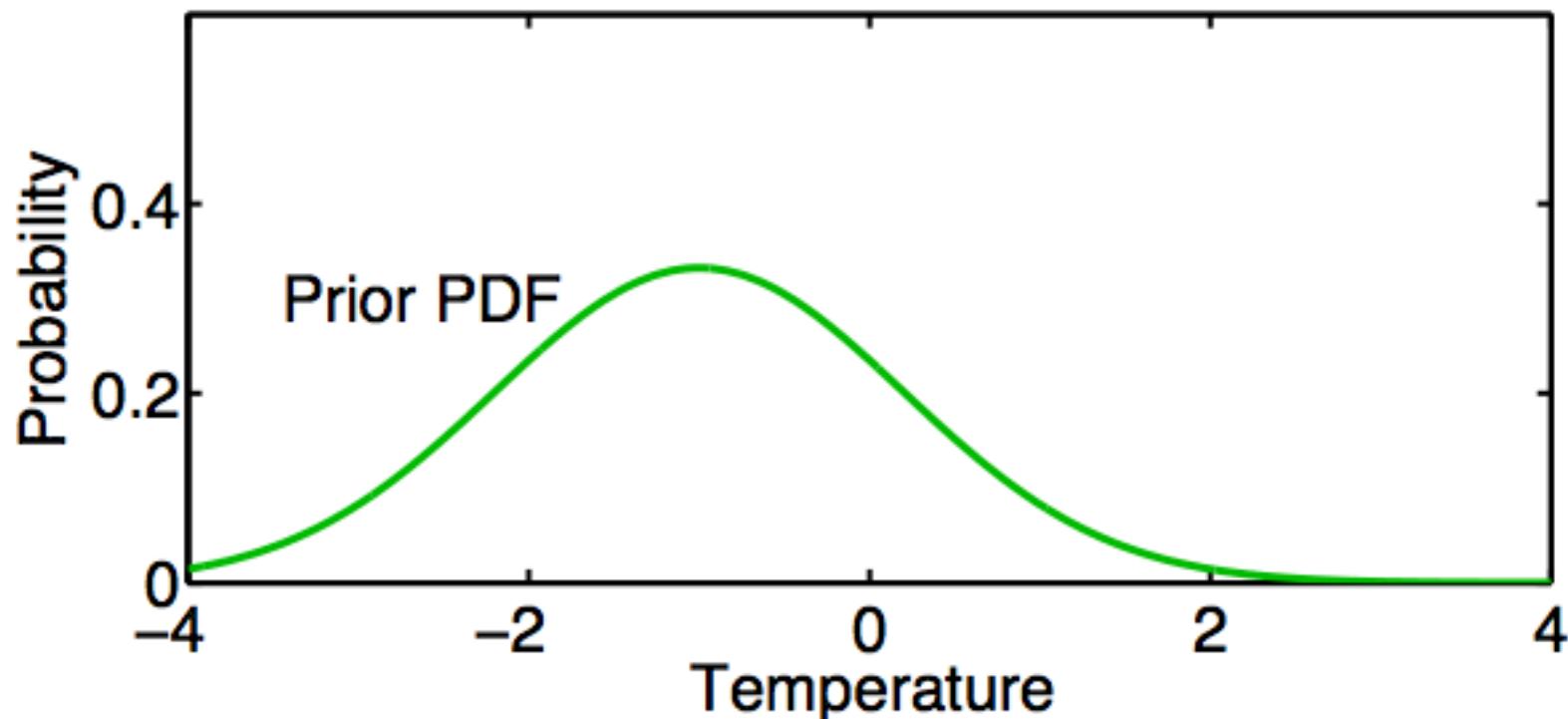
Rewrite Bayes as:

$$\begin{aligned} \frac{P(T_o | T, C) P(T | C)}{P(T_o | C)} &= \frac{P(T_o | T, C) P(T | C)}{\int P(T_o | x) P(x | C) dx} \\ &= \frac{P(T_o | T, C) P(T | C)}{normalization} \end{aligned}$$

Denominator normalizes so Posterior is PDF.

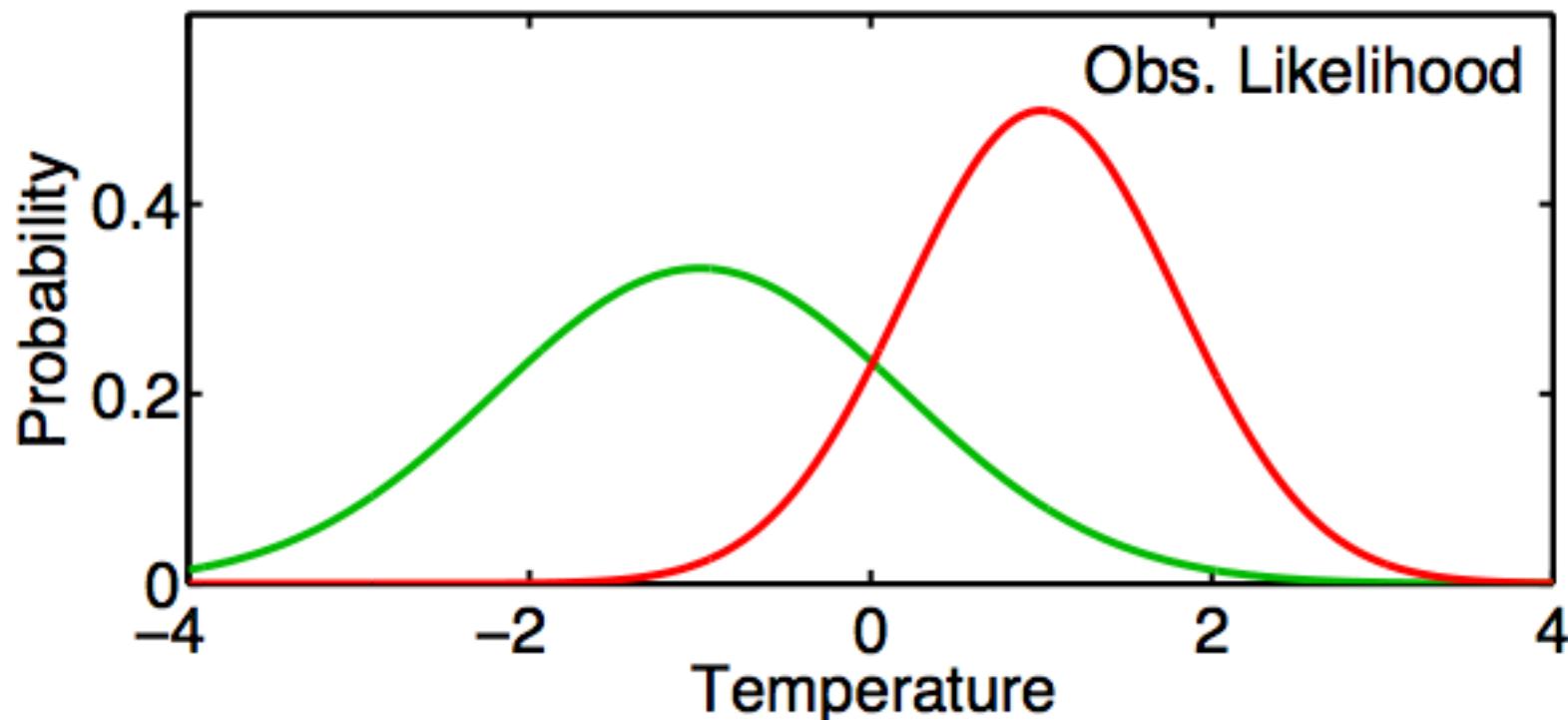
Combining the Prior Estimate and Observation

$$P(T | T_0, C) = \frac{P(T_0 | T, C) P(T | C)}{\text{normalization}}$$



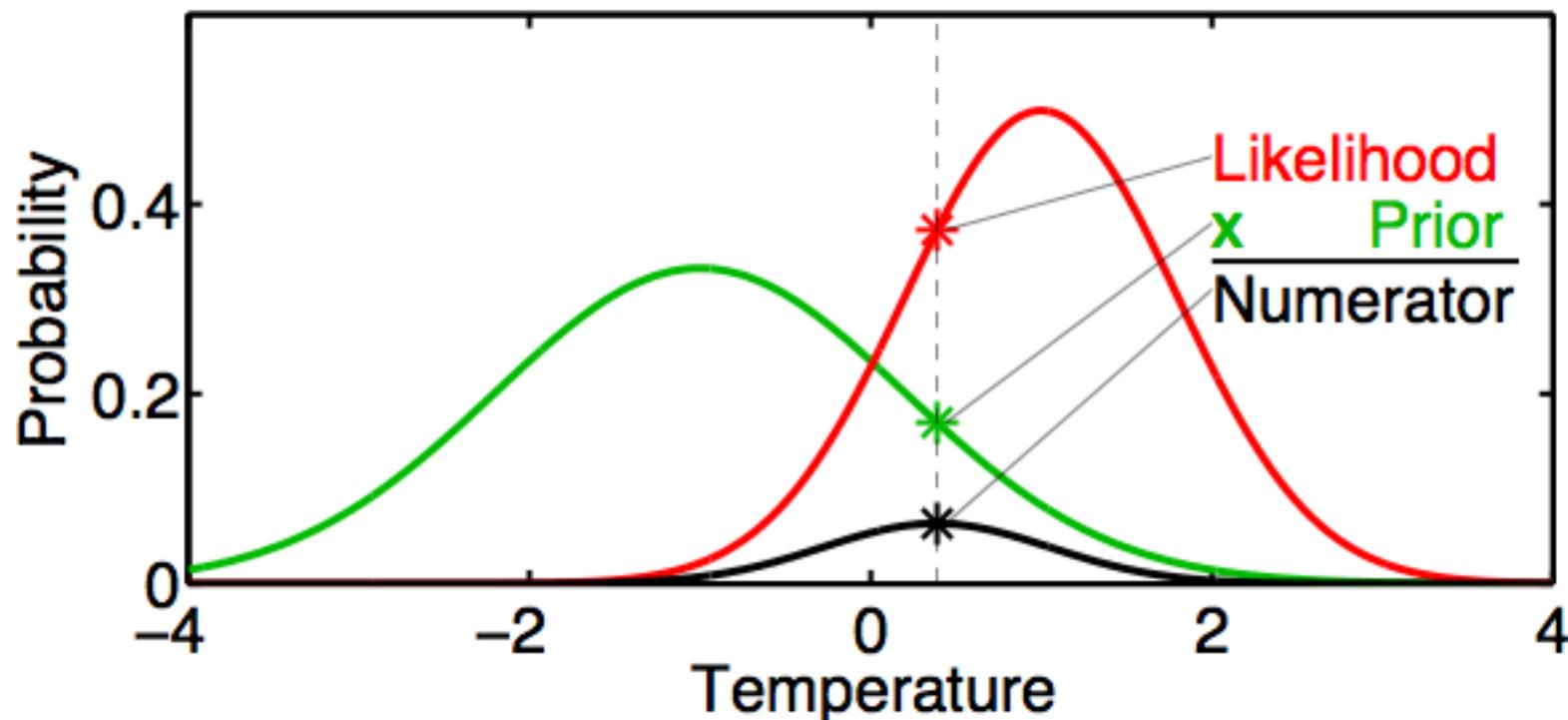
Combining the Prior Estimate and Observation

$$P(T | T_0, C) = \frac{P(T_0 | T, C) P(T | C)}{\text{normalization}}$$



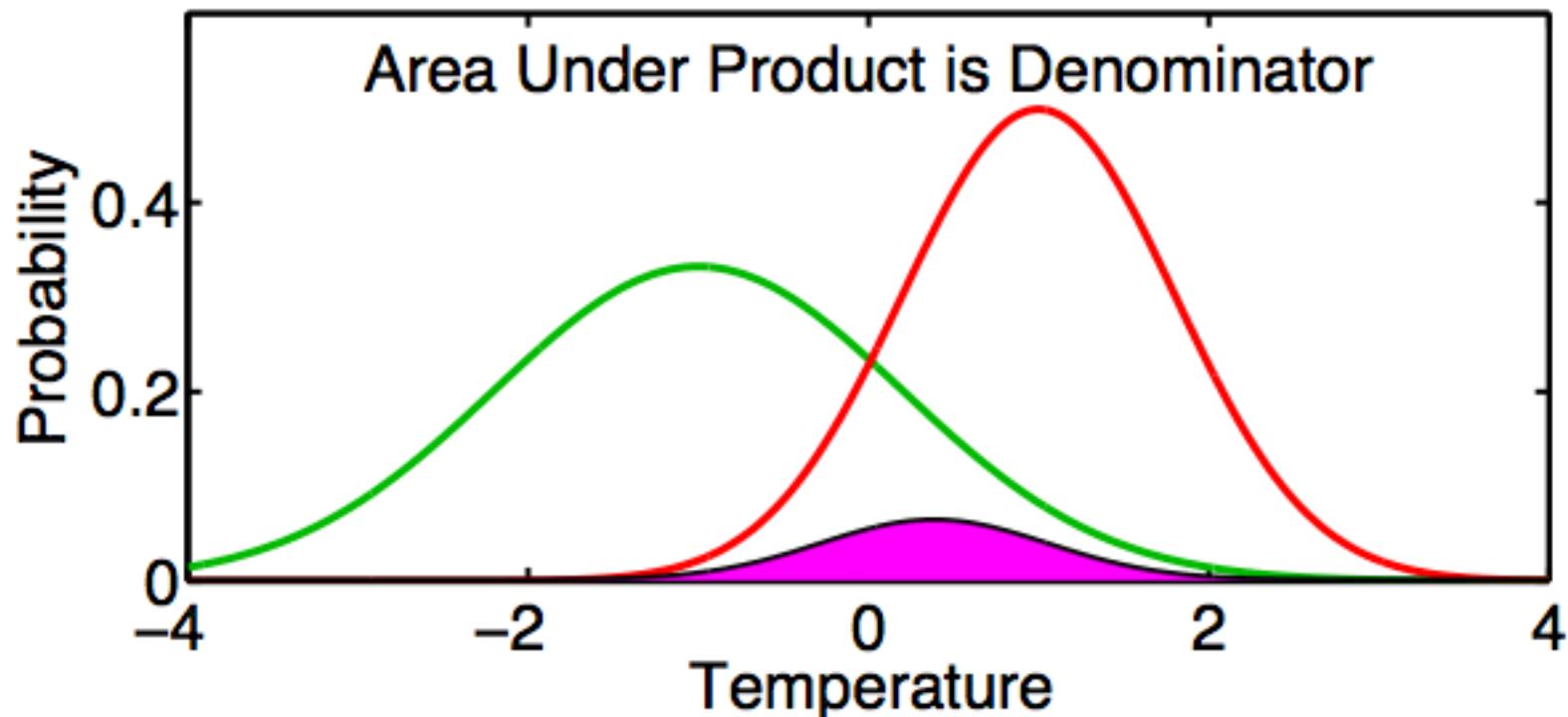
Combining the Prior Estimate and Observation

$$P(T | T_0, C) = \frac{P(T_0 | T, C) P(T | C)}{\text{normalization}}$$



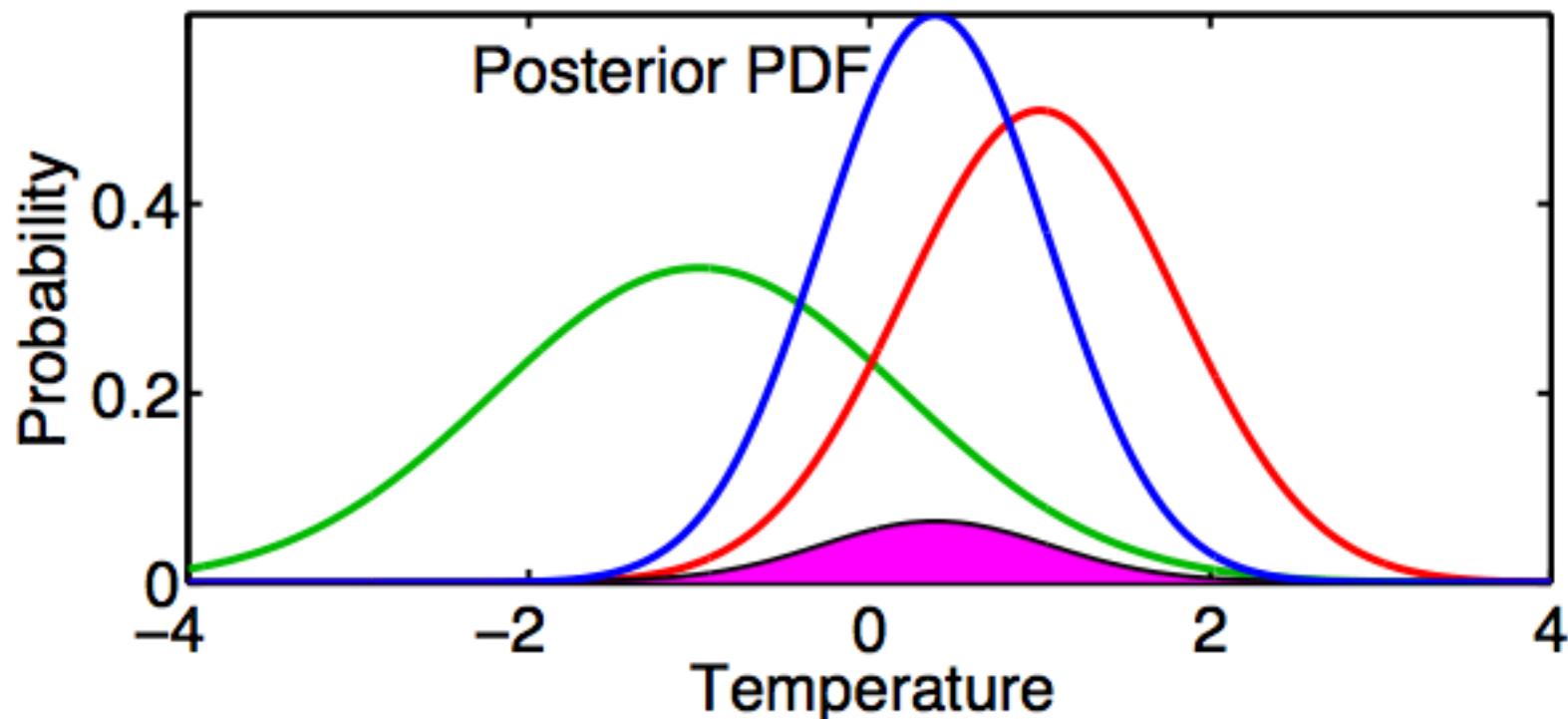
Combining the Prior Estimate and Observation

$$P(T | T_0, C) = \frac{P(T_0 | T, C) P(T | C)}{\text{normalization}}$$



Combining the Prior Estimate and Observation

$$P(T | T_0, C) = \frac{P(T_0 | T, C) P(T | C)}{\text{normalization}}$$



Consistent Color Scheme Throughout Tutorial

Green = Prior

Red = Observation

Blue = Posterior

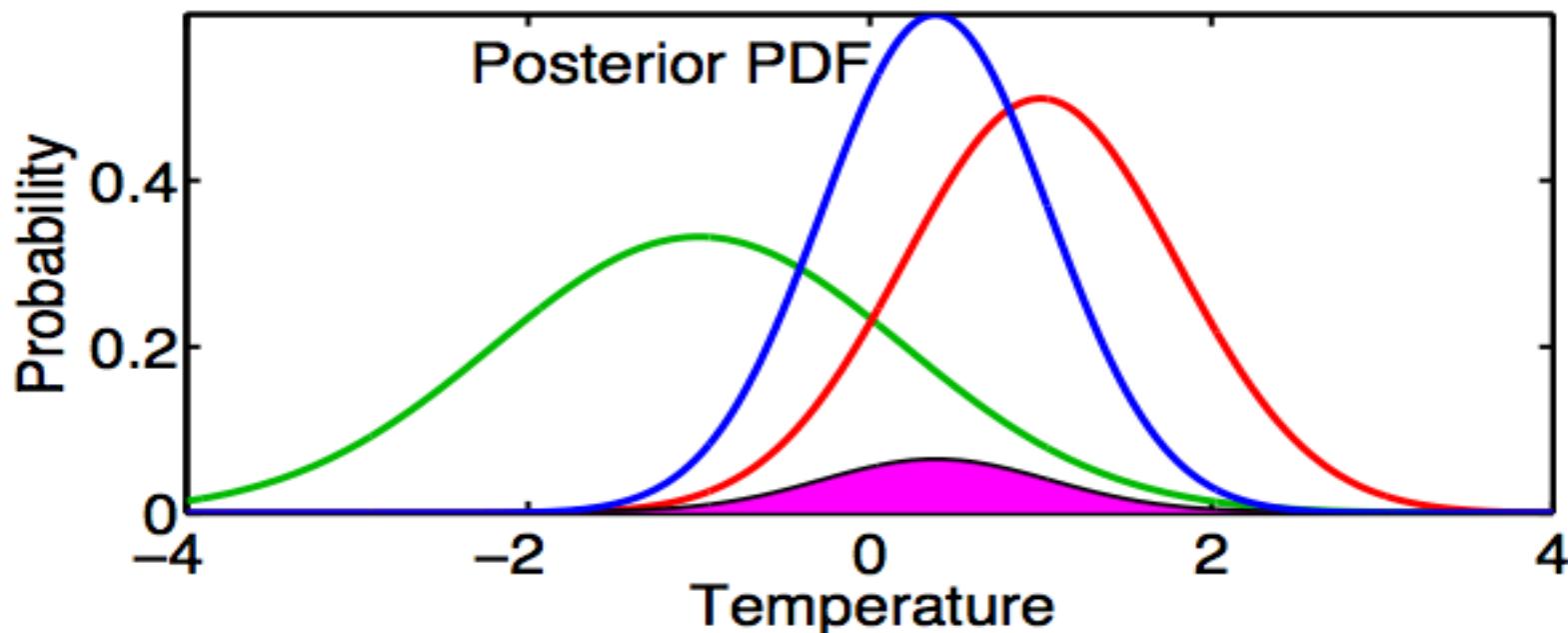
Black = Truth

(truth available only for ‘perfect model’ examples)

Combining the Prior Estimate and Observation

$$P(T | T_0, C) = \frac{P(T_0 | T, C) P(T | C)}{\text{normalization}}$$

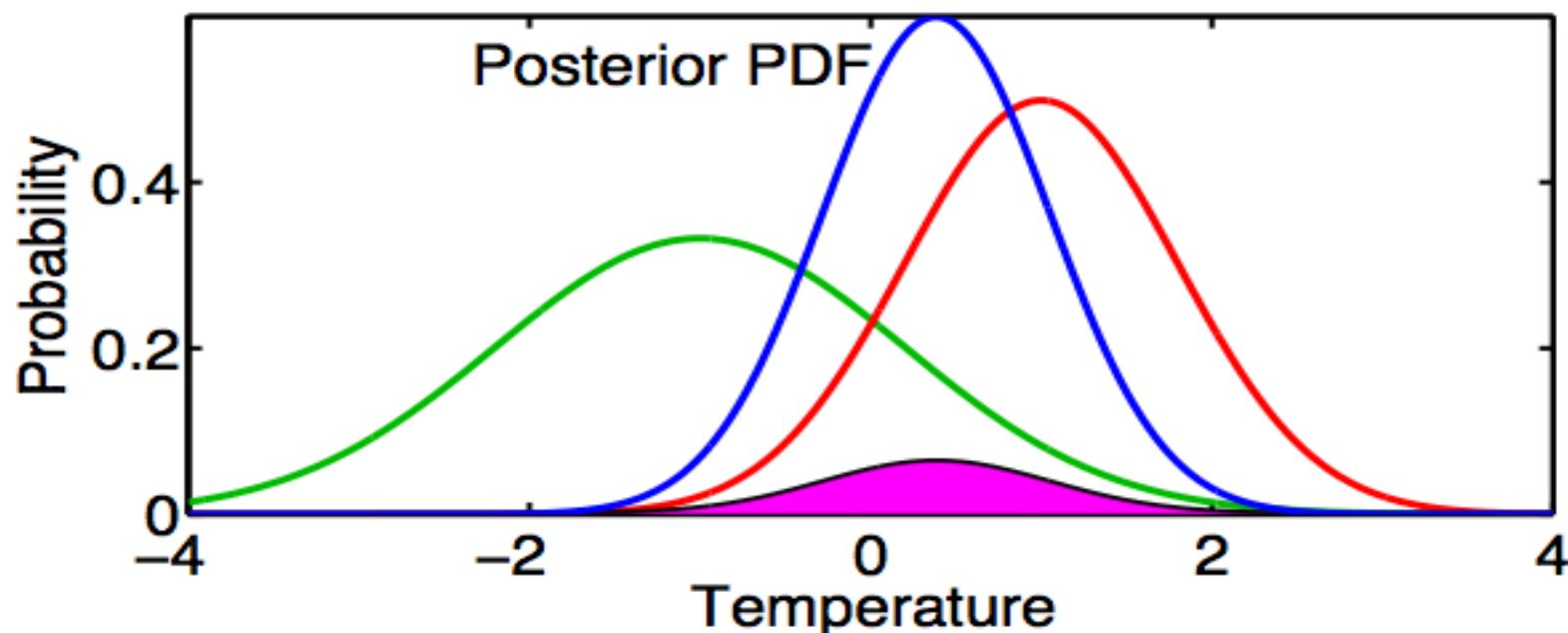
Generally no analytic solution for Posterior.



Combining the Prior Estimate and Observation

$$P(T | T_0, C) = \frac{P(T_0 | T, C) P(T | C)}{\text{normalization}}$$

Gaussian Prior and Likelihood \rightarrow Gaussian Posterior



Combining the Prior Estimate and Observation

For Gaussian prior and likelihood...

Prior

$$P(T|C) = \text{Normal}(T_p, \sigma_p)$$

Likelihood

$$P(T_o|T, C) = \text{Normal}(T_o, \sigma_o)$$

Then, Posterior

$$P(T|T_o, C) = \text{Normal}(T_u, \sigma_u)$$

With

$$\sigma_u = \sqrt{(\sigma_p^{-2} + \sigma_o^{-2})^{-1}}$$

$$T_u = \sigma_u^2 \left[\sigma_p^{-2} T_p + \sigma_o^{-2} T_o \right]$$

Matlab Hands-on: gaussian_product

MATLAB R2014a

HOME PLOTS APPS

New Script New Open Compare Import Data Save Workspace Clear Workspace New Variable Open Variable Analyze Code Run and Time Layout Preferences Help Community Request Support Add-Ons

FILE VARIABLE CODE ENVIRONMENT RESOURCES

Current Folder / Users > thoar > svn > DART > lanai > DART_LAB > matlab

Command Window

```
>> gaussian_product
gaussian_product demonstrates the product of two gaussian distributions.

This is fundamental to Kalman filters and to ensemble
data assimilation. Change the parameters of the
gaussian for the Prior (green) and the Observation (red)
and click on 'Plot Posterior'.
```

The product (in this case, the 'Posterior') of
two gaussians is a gaussian.

If the parameters of the two gaussians are known,
the parameters of the resulting gaussian can be calculated.

See also: [oned_model](#), [oned_ensemble](#), [twod_ensemble](#),
[run_lorenz_63](#), [run_lorenz_96](#)

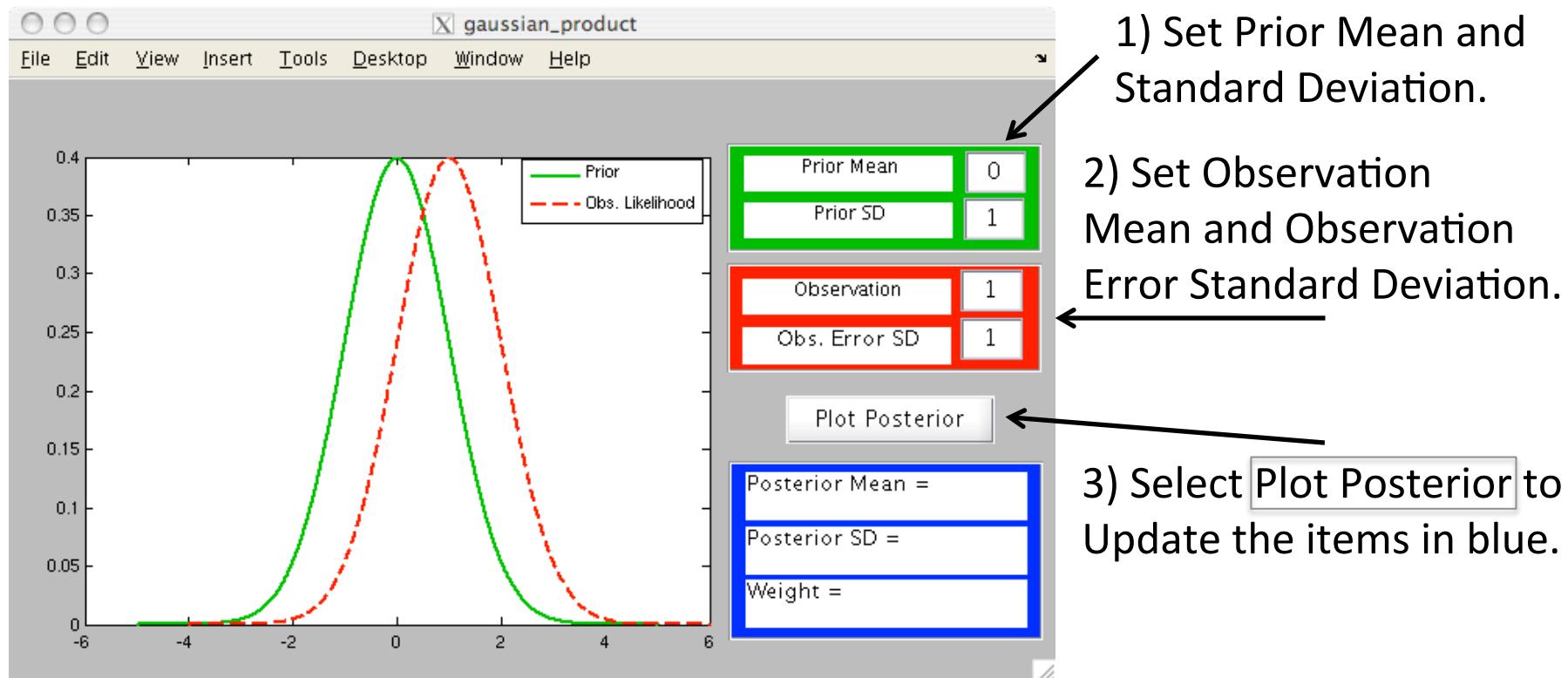
fx >>

Select a file to view details

This will also spawn a GUI that we will work with.

Matlab Hands-on: gaussian_product

Purpose: Explore the gaussian posterior that results from taking the product of a gaussian prior and a gaussian likelihood.



Matlab Hands-on: gaussian_product

Explore!

- Change the mean value of the prior and the observation.
- Change the standard deviation of the prior.
- What is always true for the mean of the posterior?
- What is always true for the standard deviation of the posterior?

The One-Dimensional Kalman Filter

1. Suppose we have a linear forecast model L
 - A. If temperature at time $t_1 = T_1$, then the temperature at $t_2 = t_1 + \Delta t$ is $T_2 = L(T_1)$
 - B. Example: $T_2 = T_1 + \Delta t T_1$

The One-Dimensional Kalman Filter

1. Suppose we have a linear forecast model L .
 - A. If temperature at time $t_1 = T_1$, then the temperature at $t_2 = t_1 + \Delta t$ is $T_2 = L(T_1)$
 - B. Example: $T_2 = T_1 + \Delta t T_1$
2. If posterior estimate at time t_1 is $Normal(T_{u,1}, \sigma_{u,1})$ then the prior at t_2 is $Normal(T_{p,2}, \sigma_{p,2})$.

$$T_{p,2} = T_{u,1} + \Delta t T_{u,1}$$

$$\sigma_{p,2} = (\Delta t + 1) \sigma_{u,1}$$

The One-Dimensional Kalman Filter

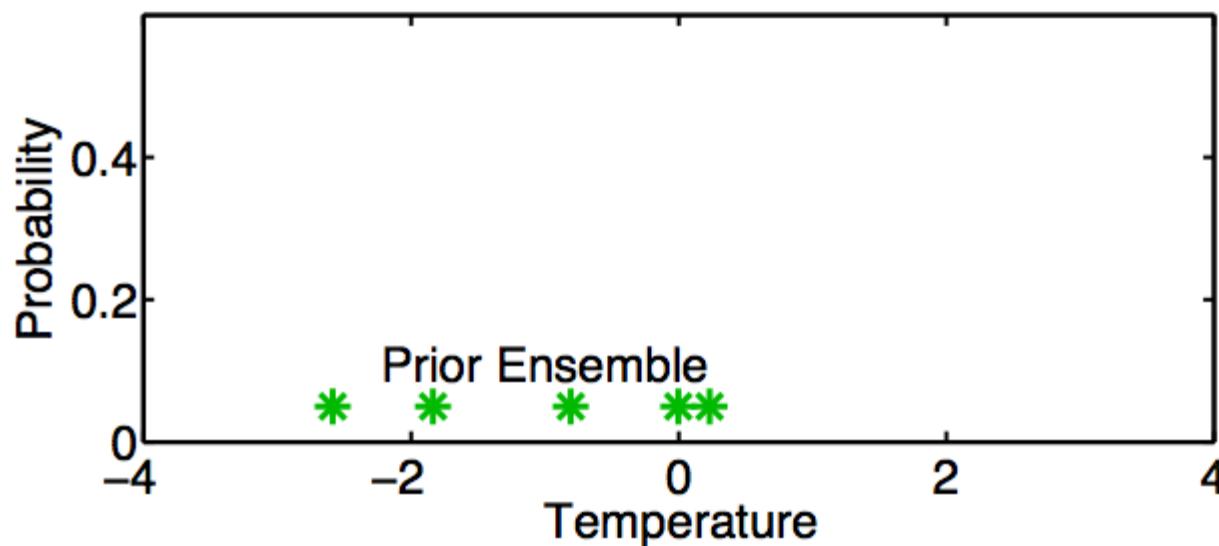
1. Suppose we have a linear forecast model L .
 - A. If temperature at time $t_1 = T_1$, then the temperature at $t_2 = t_1 + \Delta t$ is $T_2 = L(T_1)$
 - B. Example: $T_2 = T_1 + \Delta t T_1$
2. If posterior estimate at time t_1 is $Normal(T_{u,1}, \sigma_{u,1})$ then the prior at t_2 is $Normal(T_{p,2}, \sigma_{p,2})$.
3. Given an observation at t_2 with distribution $Normal(t_0, \sigma_0)$ the likelihood is also $Normal(t_0, \sigma_0)$.

The One-Dimensional Kalman Filter

1. Suppose we have a linear forecast model L .
 - A. If temperature at time $t_1 = T_1$, then the temperature at $t_2 = t_1 + \Delta t$ is $T_2 = L(T_1)$
 - B. Example: $T_2 = T_1 + \Delta t T_1$
2. If posterior estimate at time t_1 is $Normal(T_{u,1}, \sigma_{u,1})$ then the prior at t_2 is $Normal(T_{p,2}, \sigma_{p,2})$.
3. Given an observation at t_2 with distribution $Normal(t_0, \sigma_0)$ the likelihood is also $Normal(t_0, \sigma_0)$.
4. The posterior at t_2 is $Normal(T_{u,2}, \sigma_{u,2})$ where $T_{u,2}$ and $\sigma_{u,2}$ come from page 19.

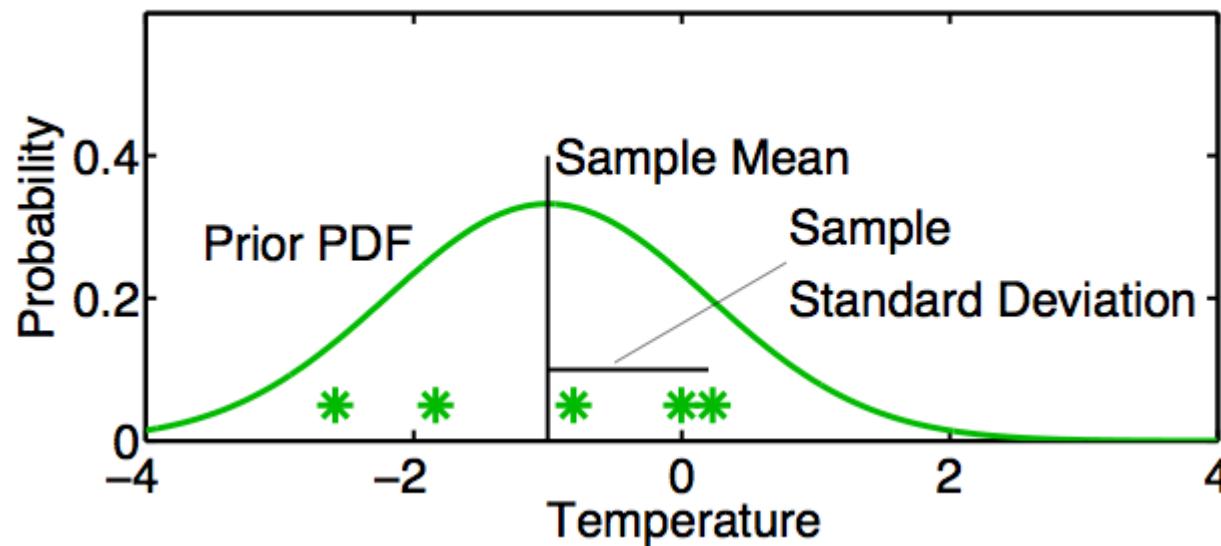
A One-Dimensional Ensemble Kalman Filter

Represent a prior pdf by a sample (ensemble) of N values:



A One-Dimensional Ensemble Kalman Filter

Represent a prior pdf by a sample (ensemble) of N values:



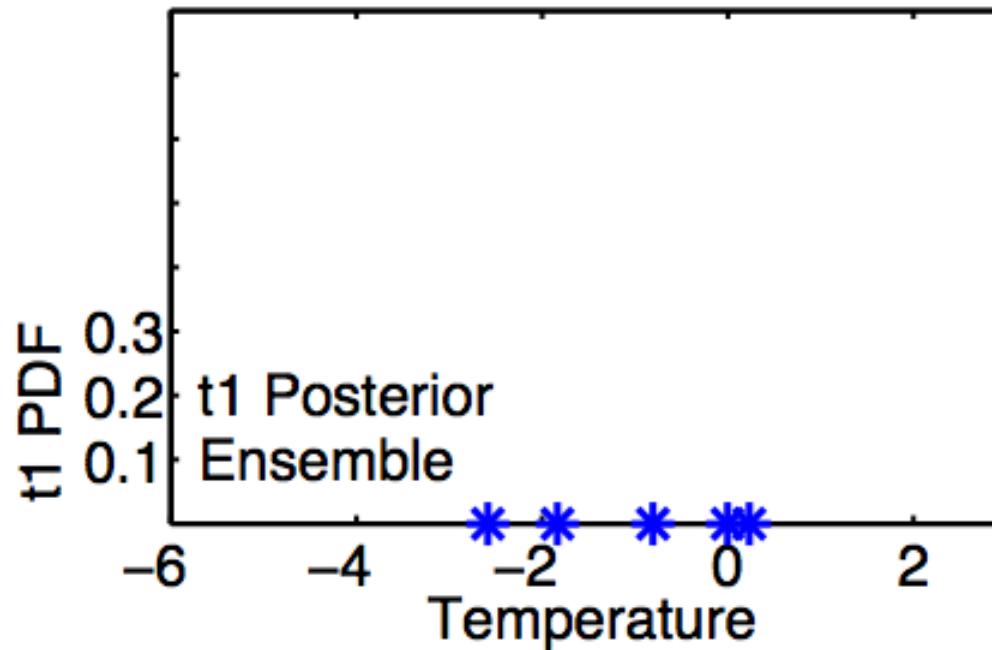
Use sample mean $\bar{T} = \sum_{n=1}^N T_n / N$

and sample standard deviation $\sigma_T = \sqrt{\sum_{n=1}^N (T_n - \bar{T})^2 / (N - 1)}$

to determine a corresponding continuous distribution $Normal(\bar{T}, \sigma_T)$

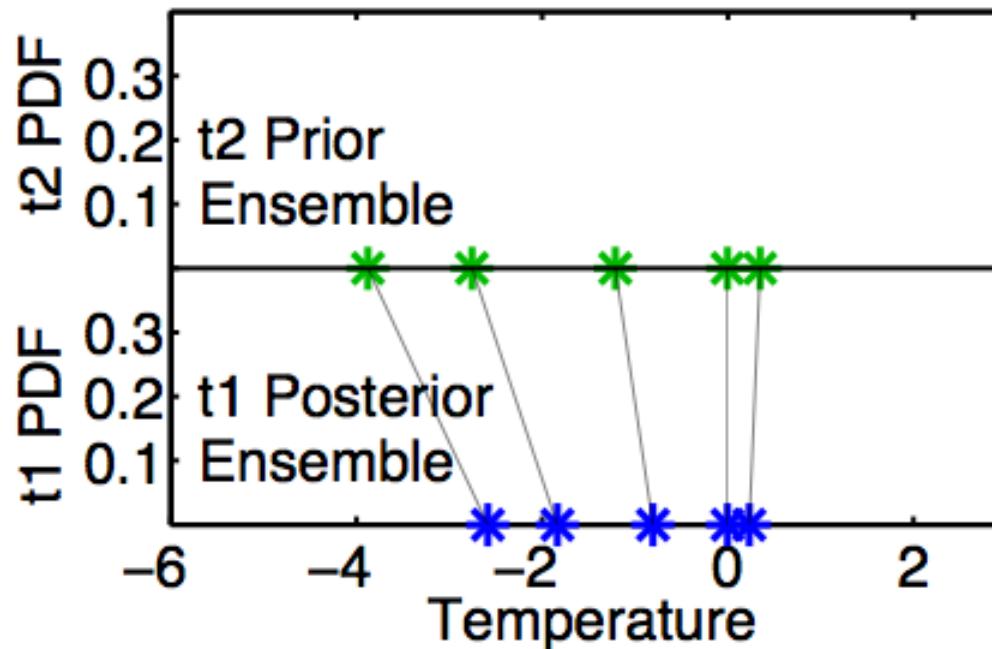
A One-Dimensional Ensemble Kalman Filter: Model Advance

If posterior ensemble at time t_1 is $T_{1,n}$, $n = 1, \dots, N$



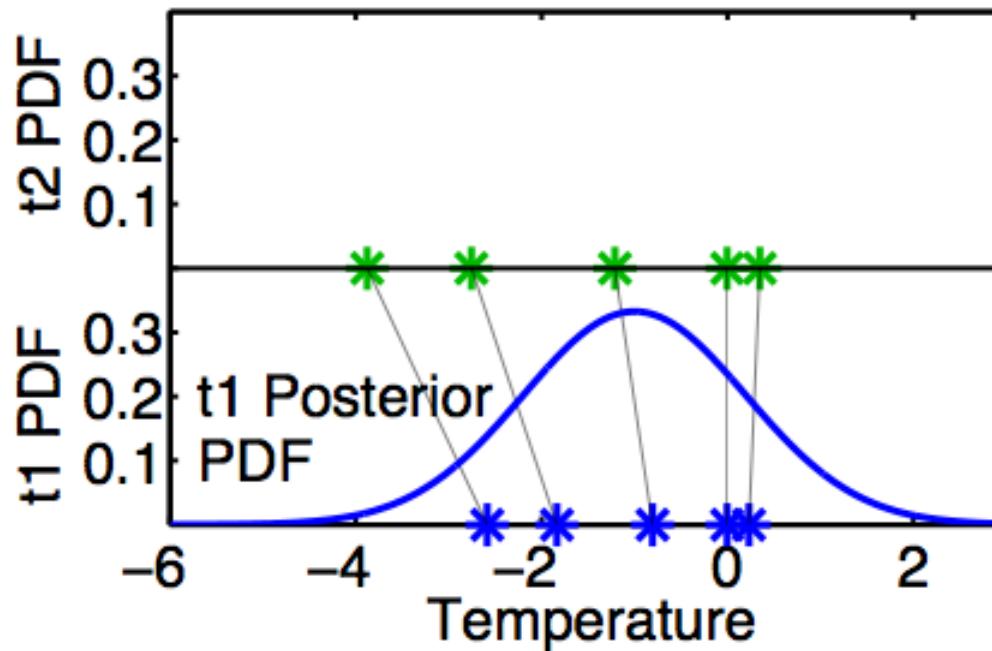
A One-Dimensional Ensemble Kalman Filter: Model Advance

If posterior ensemble at time t_1 is $T_{1,n}$, $n = 1, \dots, N$
advance each member to time t_2 with model, $T_{2,n} = L(T_{1,n})$ $n = 1, \dots, N$.



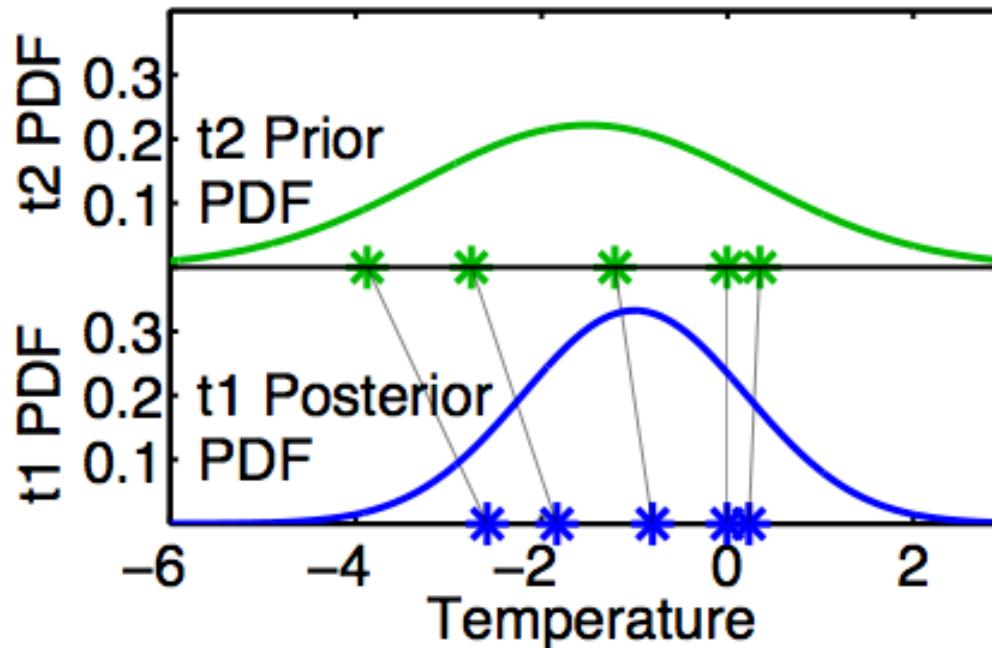
A One-Dimensional Ensemble Kalman Filter: Model Advance

Same as advancing continuous pdf at time t_1 ...

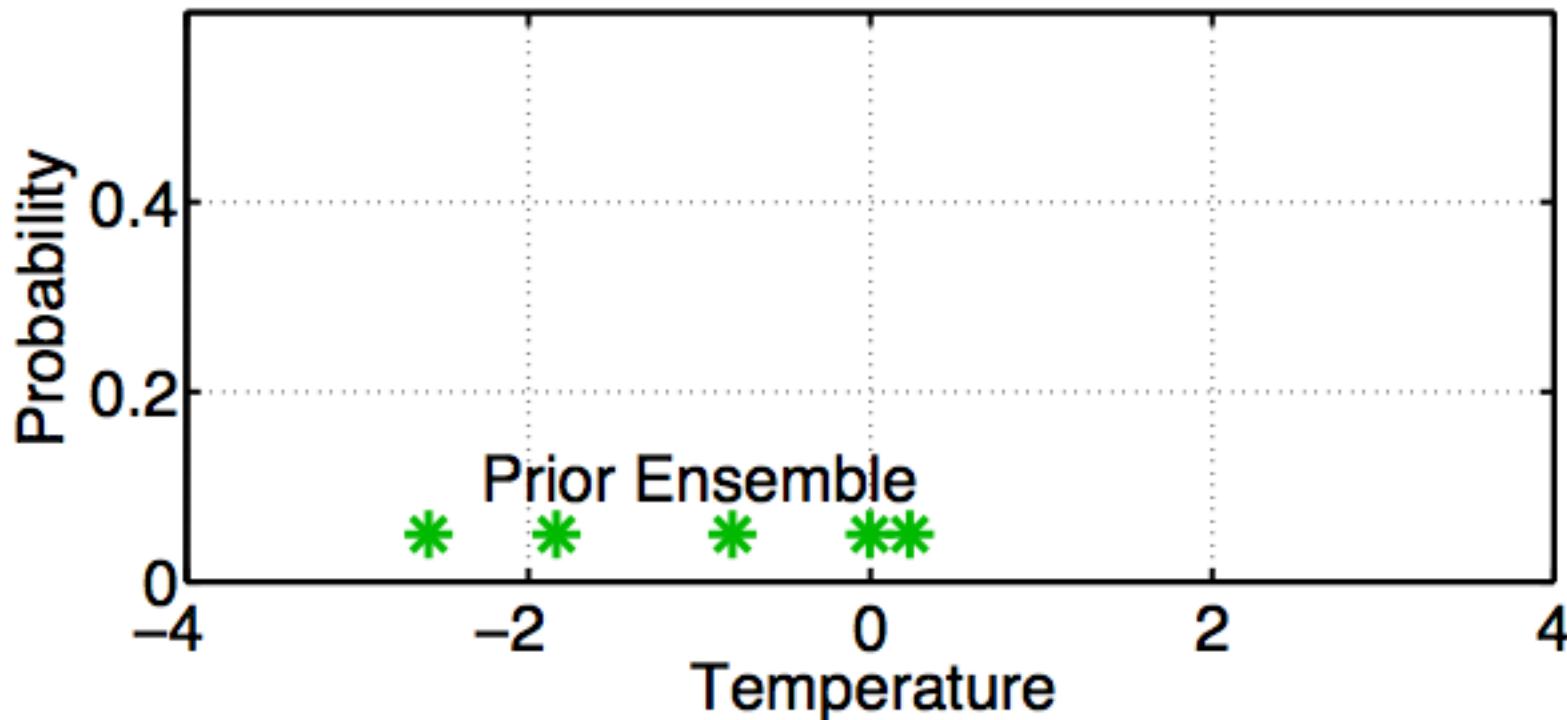


A One-Dimensional Ensemble Kalman Filter: Model Advance

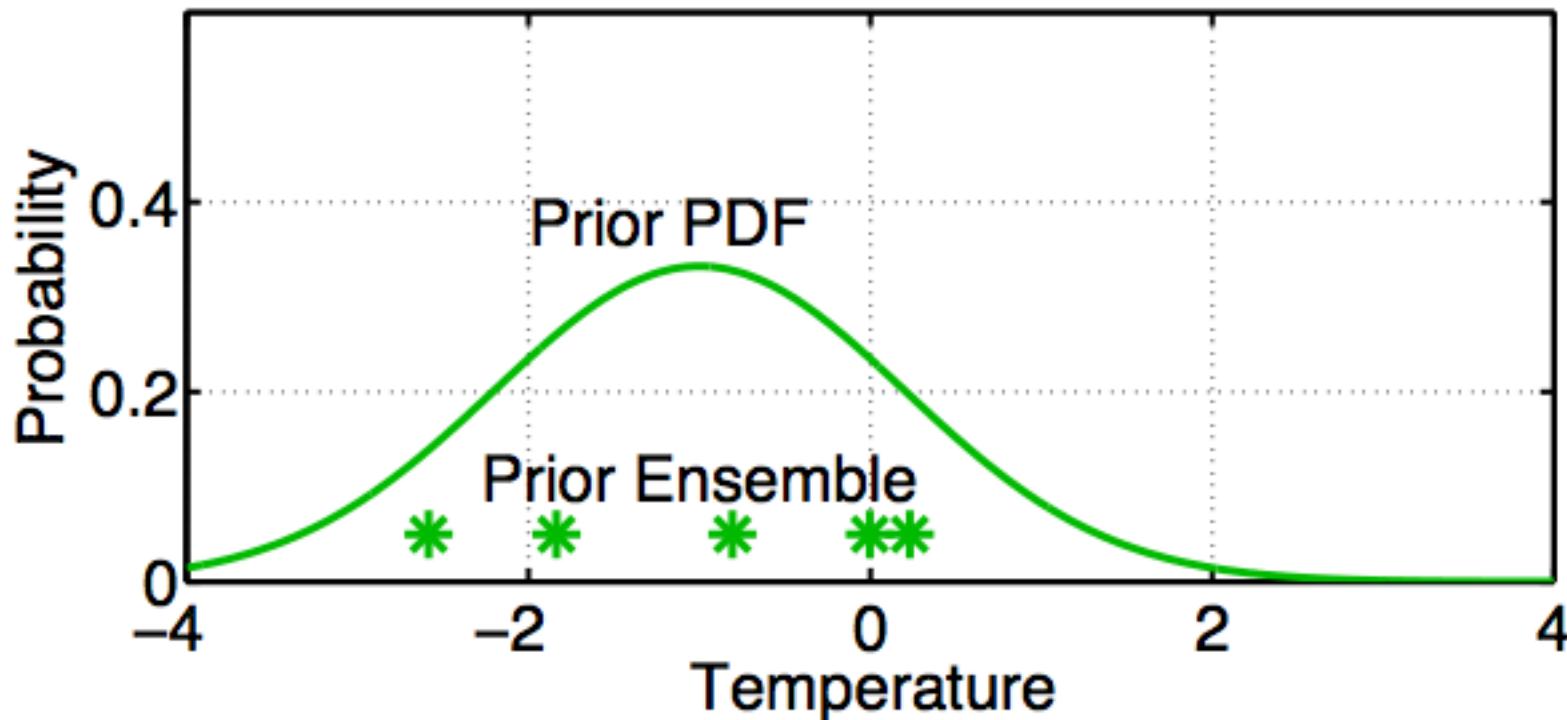
Same as advancing continuous pdf at time t_1 ...
to time t_2 with model L.



A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation

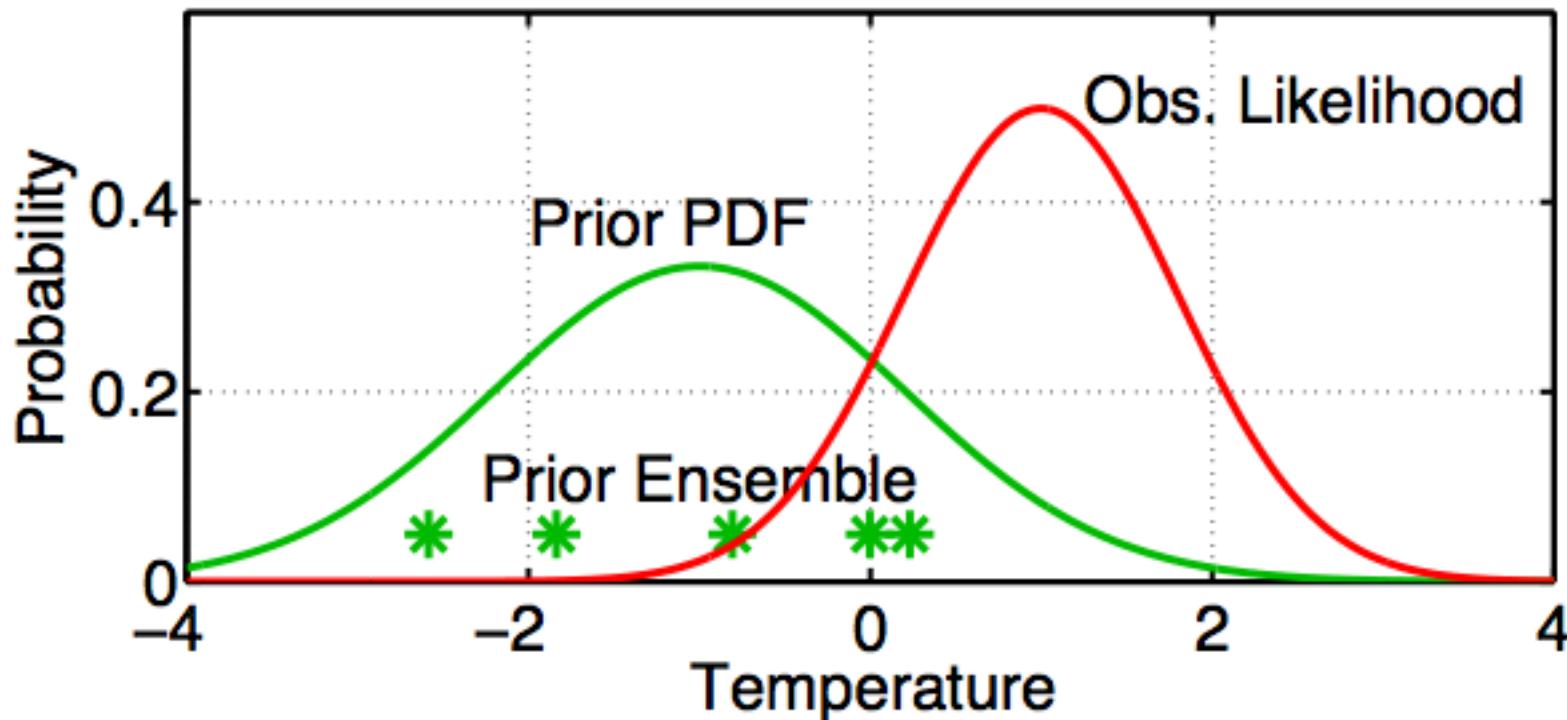


A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



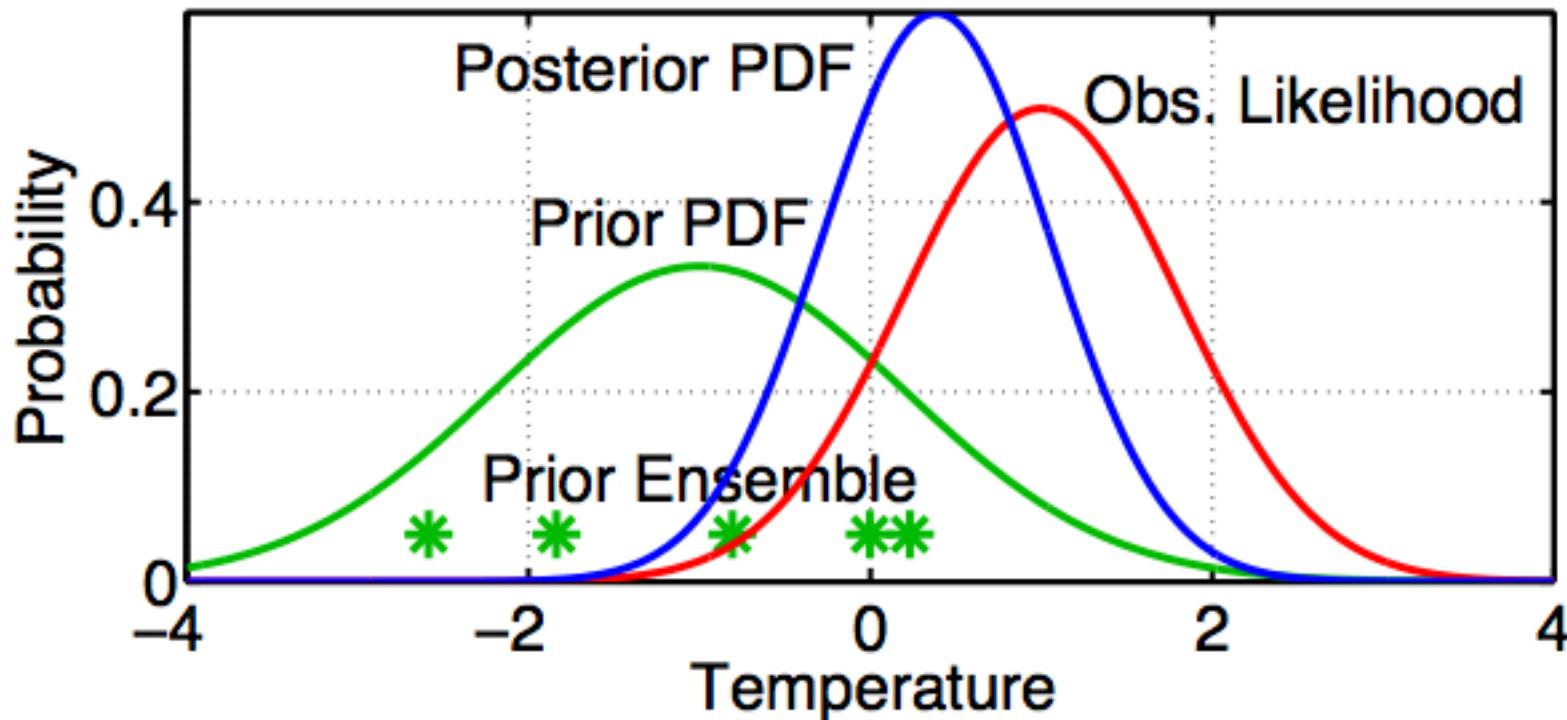
Fit a Gaussian to the sample.

A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



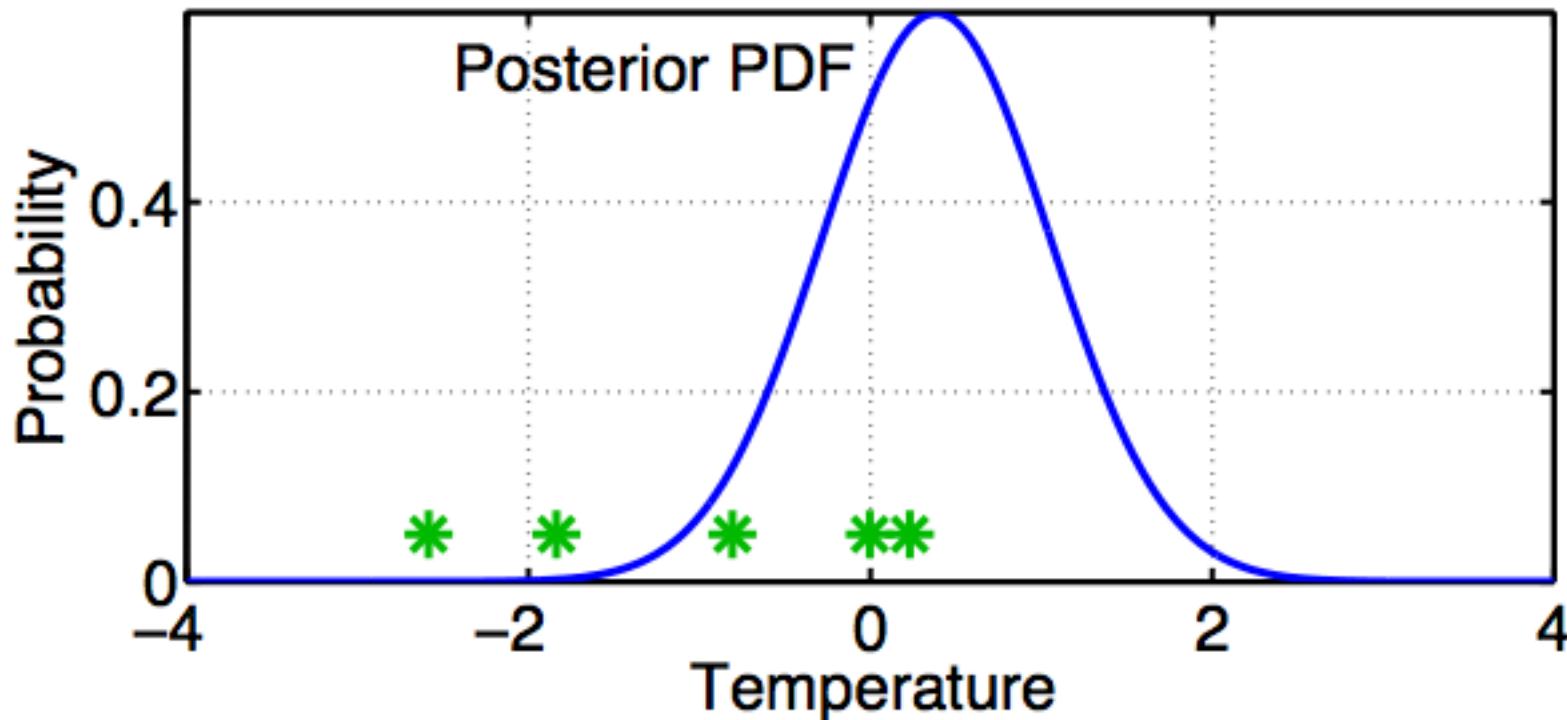
Get the observation likelihood.

A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



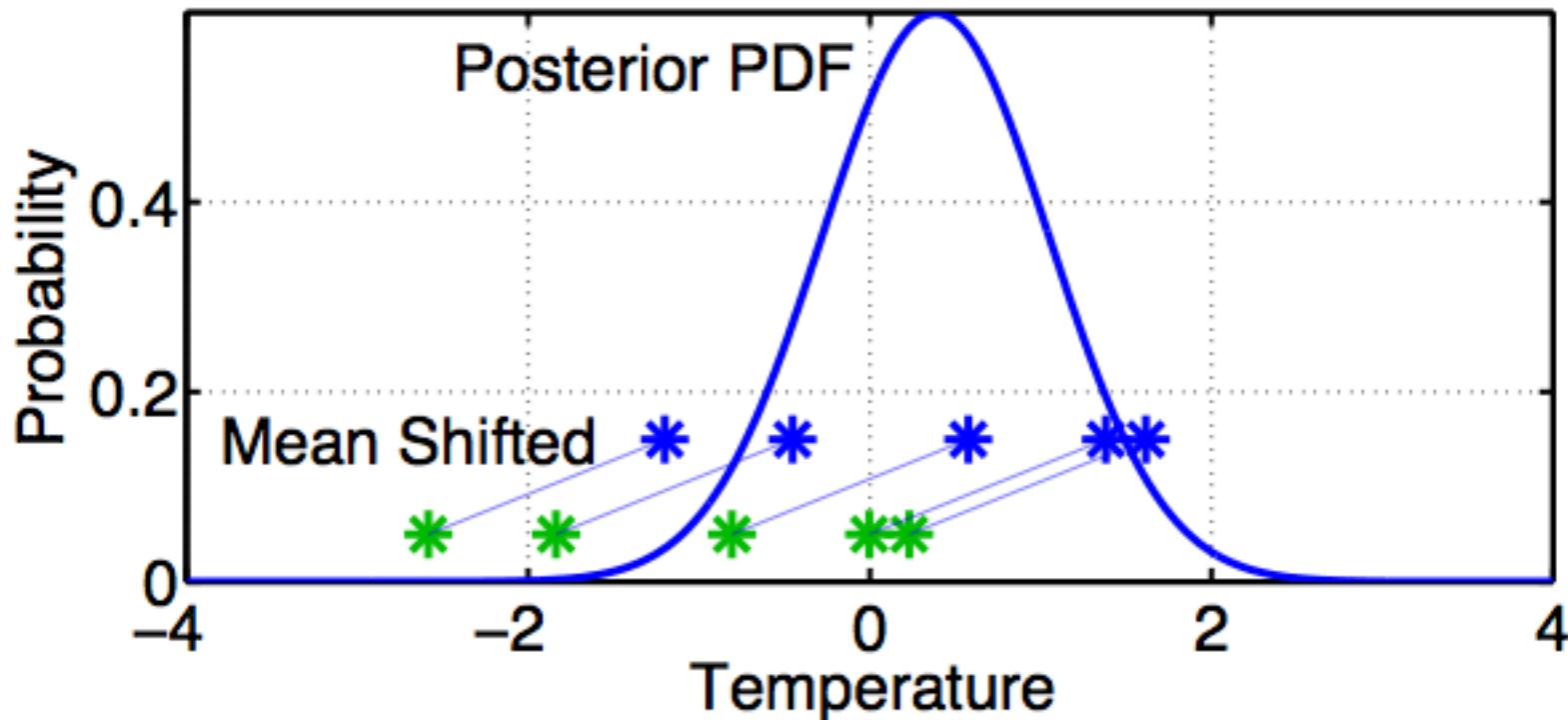
Compute the continuous posterior PDF.

A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



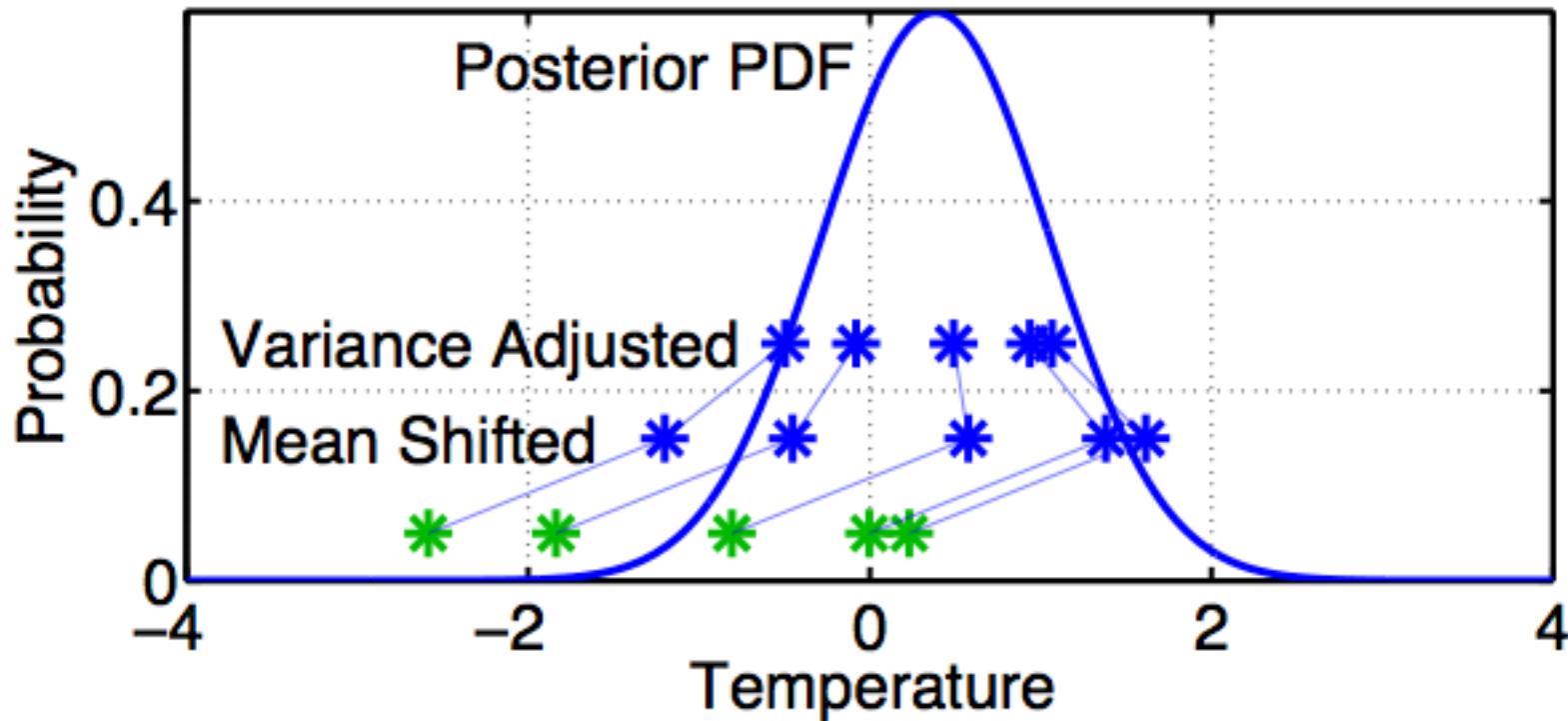
Use a deterministic algorithm to 'adjust' the ensemble.

A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



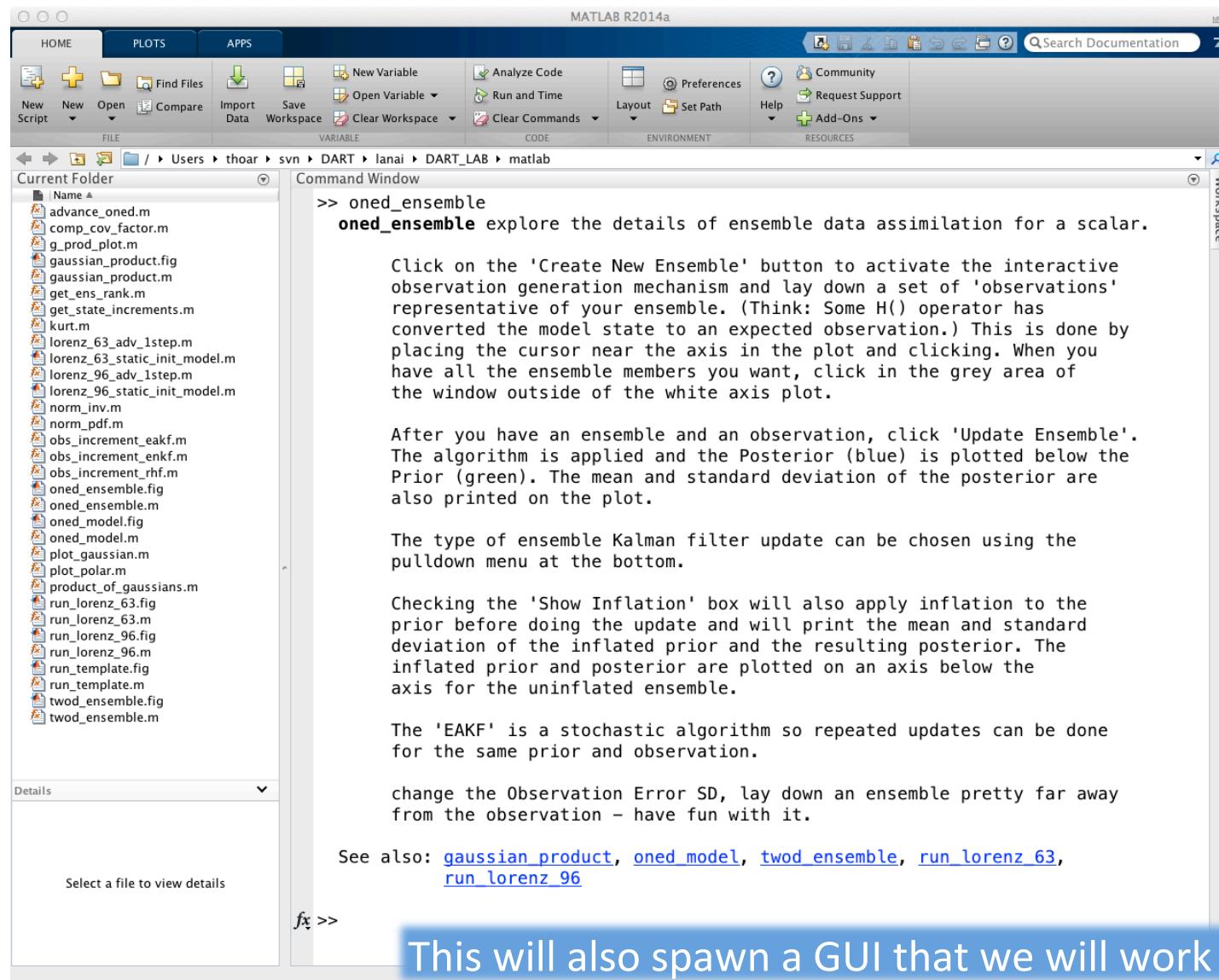
First, ‘shift’ the ensemble to have the exact mean of the posterior.

A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



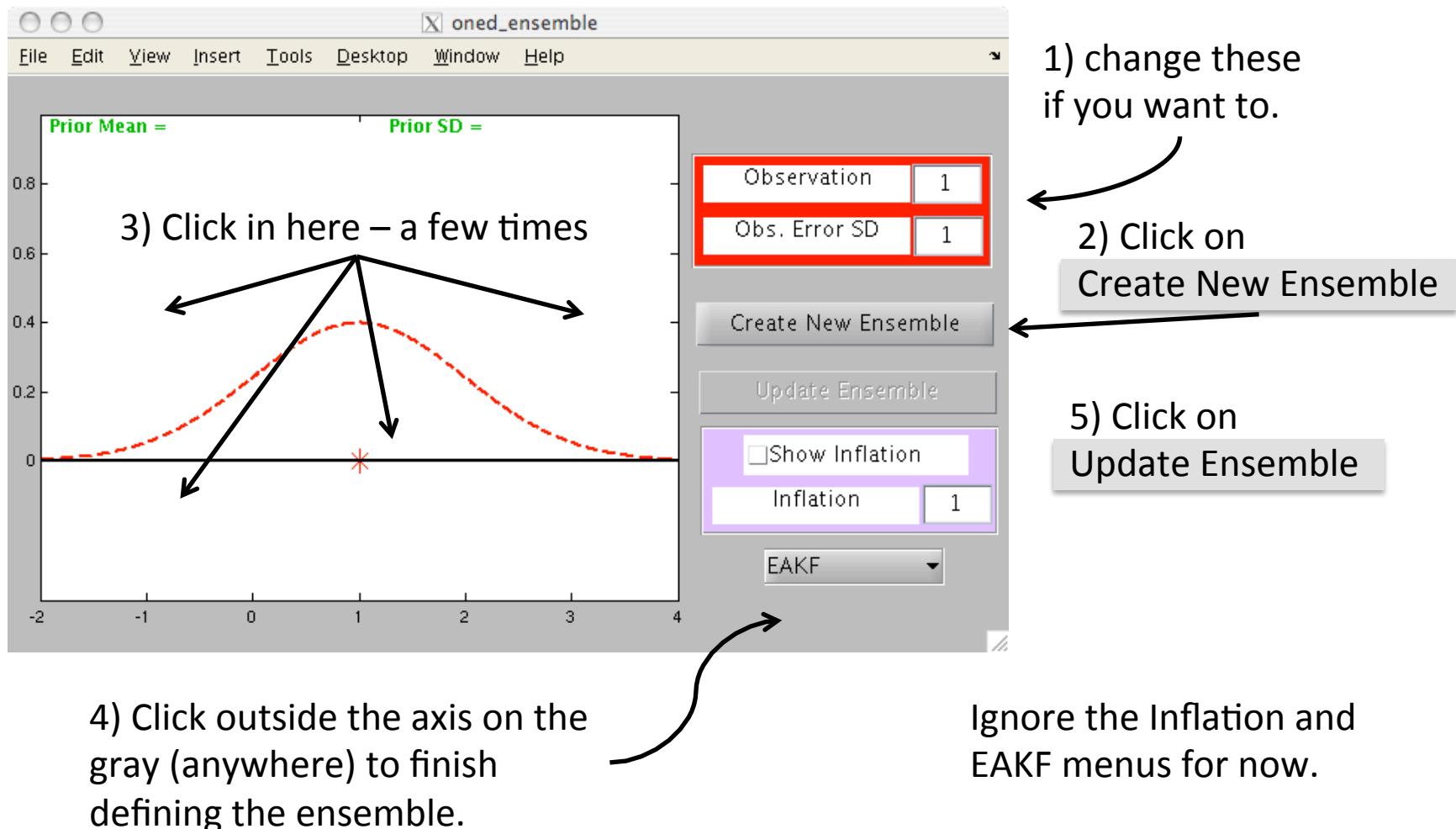
First, ‘shift’ the ensemble to have the exact mean of the posterior.
Second, linearly contract to have the exact variance of the posterior.
Sample statistics are identical to Kalman filter.

Matlab Hands-On: oned_ensemble



Matlab Hands-On: oned_ensemble

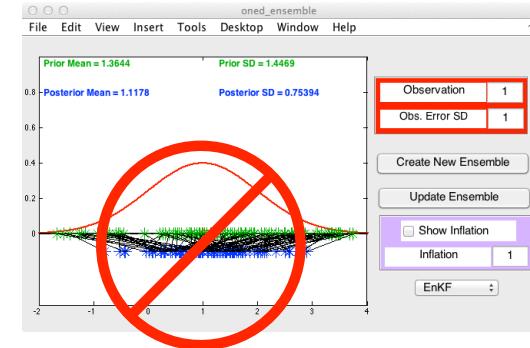
Purpose: Explore how ensemble filters update a prior ensemble.



Matlab Hands-On: oned_ensemble

Explorations:

1. Keep your ensembles small, less than 10, for easy viewing.
2. Create a nearly uniformly spaced ensemble.
Examine the update.
3. What happens with an ensemble that is confined to one side of the likelihood?
4. What happens with a bimodal ensemble (two clusters of members on either side)?
5. What happens with a single outlier in the ensemble?

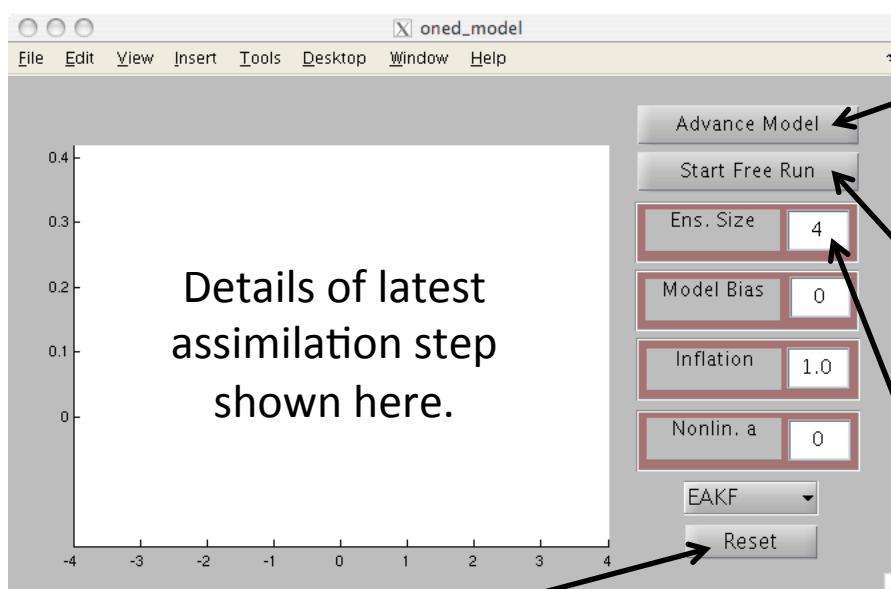


Too Many!

Matlab Hands-On: oned_model

Purpose:

- Explore behavior of a complete 1-D ensemble filter for a linear system.
- Look at the behavior of different ensemble sizes.



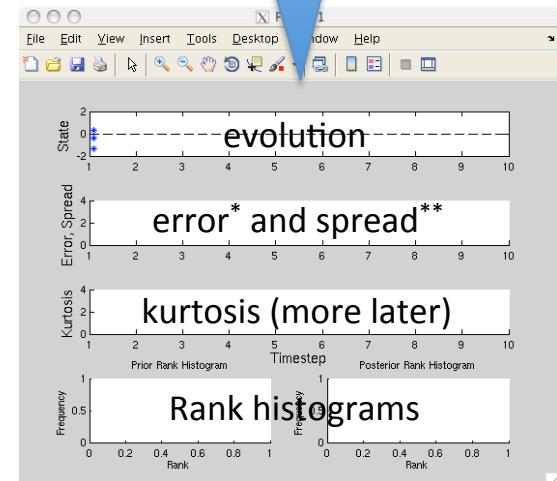
Reset restarts the exercise.

(1) Advance Model
Assimilate Obs

(2) Start Free Run
Stop Free Run

ensemble size can be changed

This window also pops up. More later.



Note:

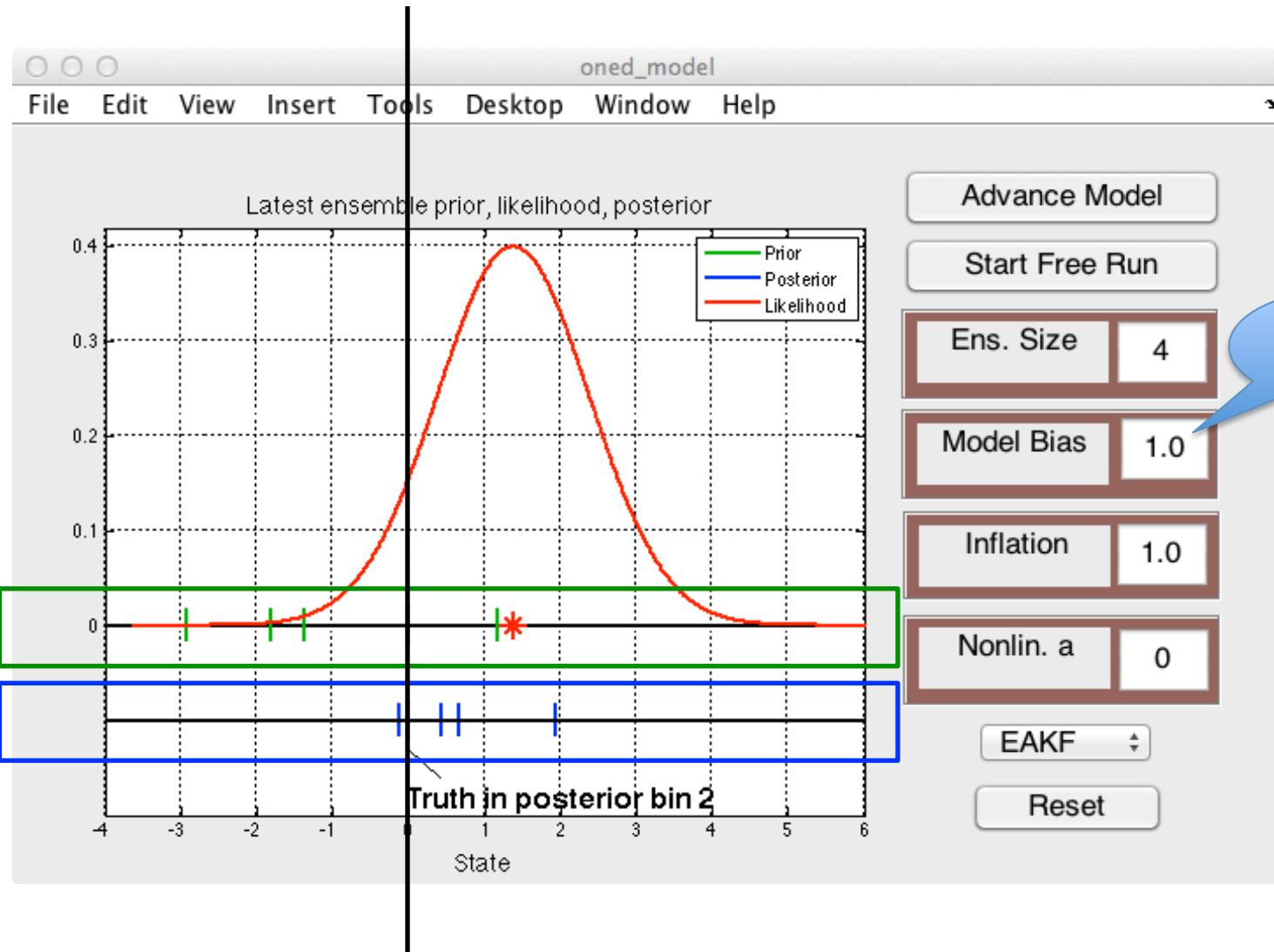
This uses the simple linear model $dx/dt = x$.

The ‘truth’ is always 0.

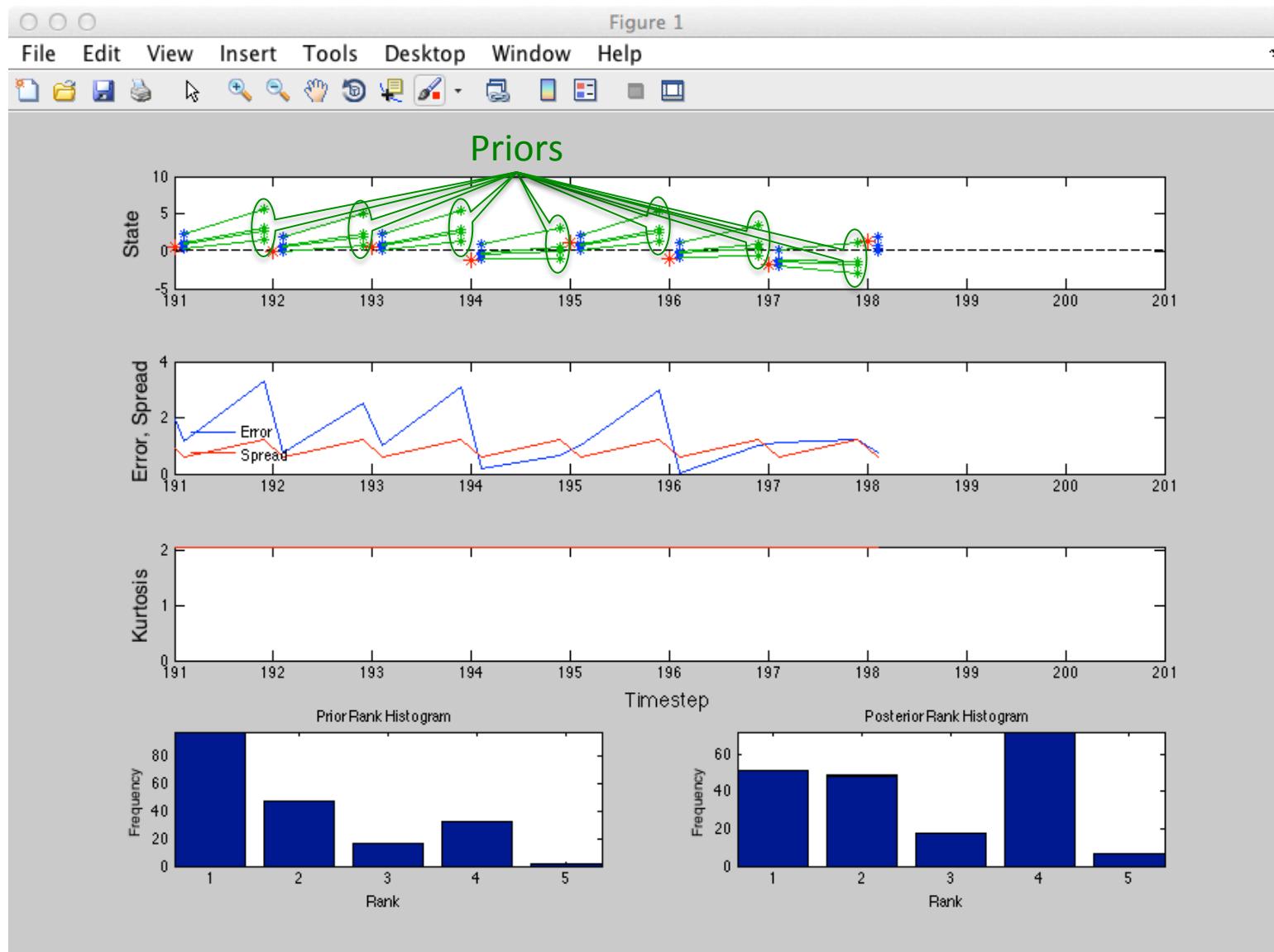
Observation noise is a draw from $N(0,1)$.

Matlab Hands-On: oned_model

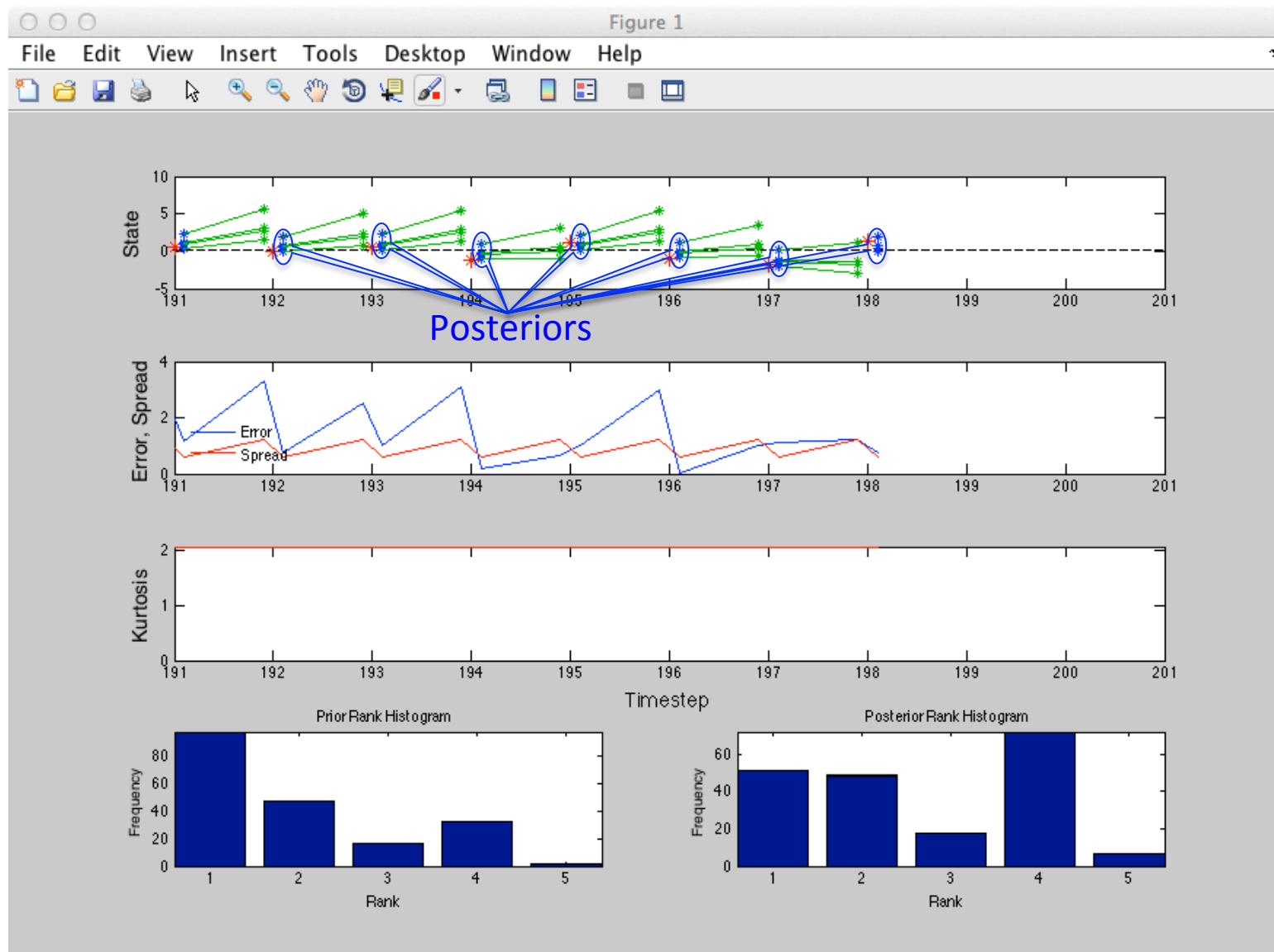
The truth is always 0.0



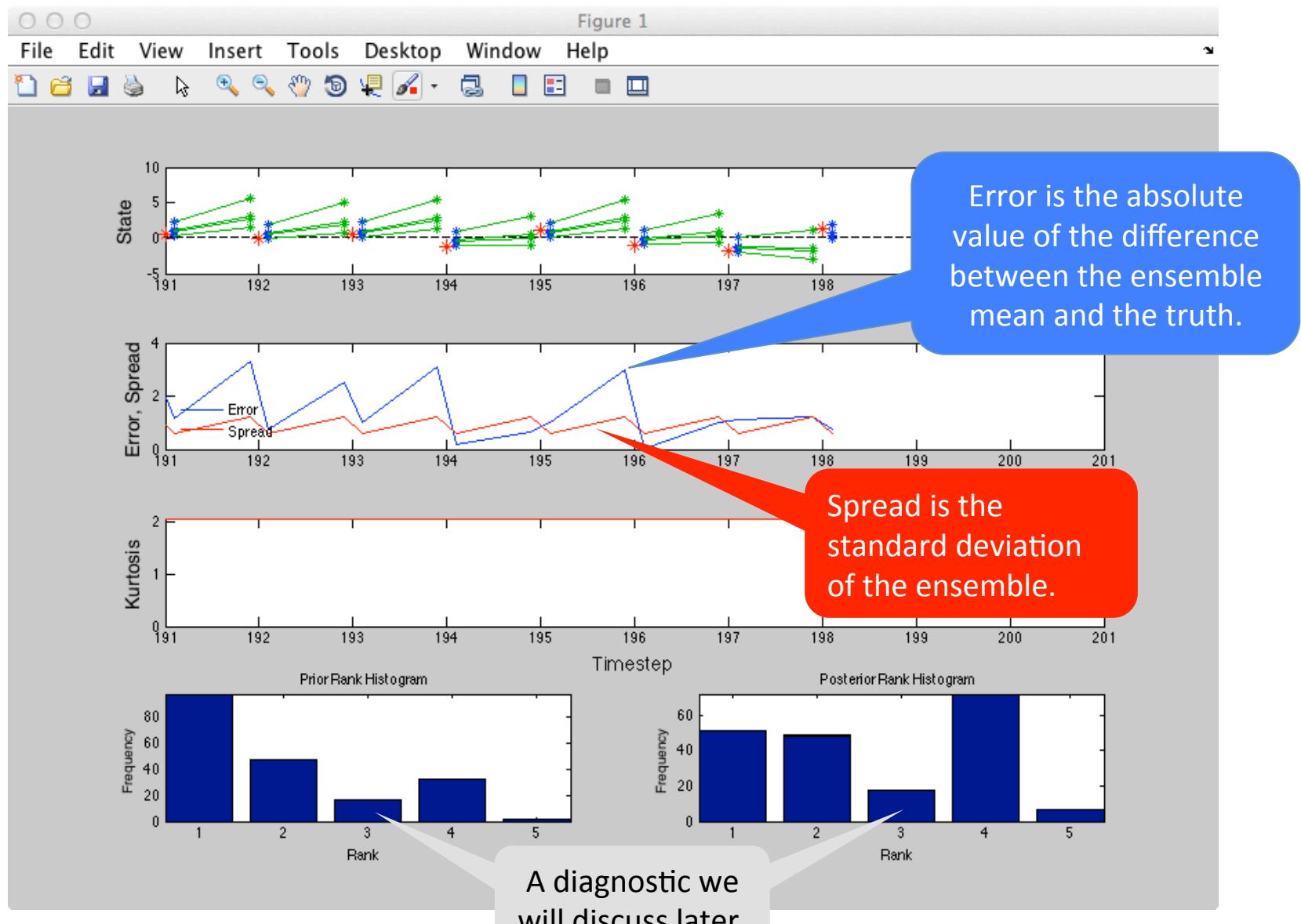
Matlab Hands-On: oned_model



Matlab Hands-On: oned_model



Matlab Hands-On: oned_model



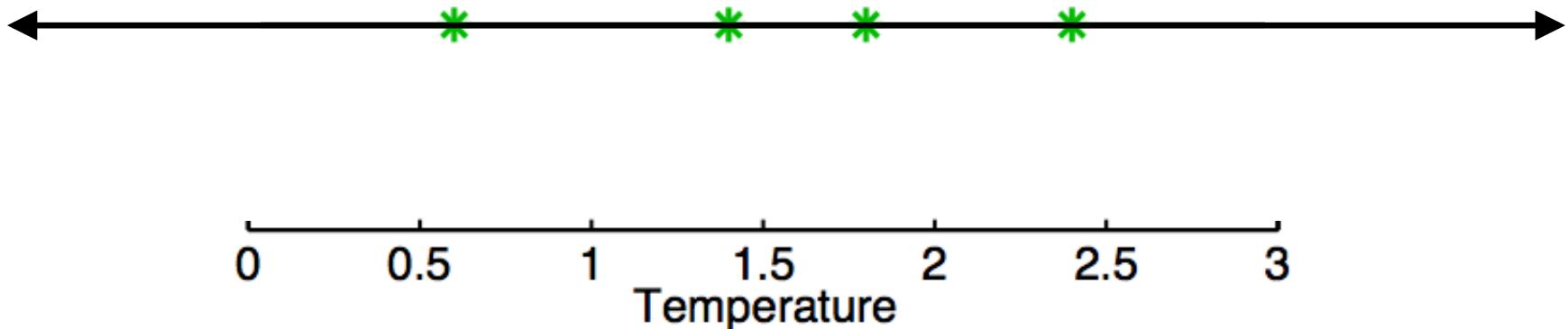
Matlab Hands-On: oned_model

Explorations:

1. Step through a sequence of advances and assimilations with the top button. Watch the evolution of the ensemble, the error and spread.
2. How does a larger ensemble size (< 10 is easiest to see) act?
 - Compare the error and spread for different ensemble sizes.
 - Note the time behavior of the error and spread.
3. Let the model run freely using the second button.

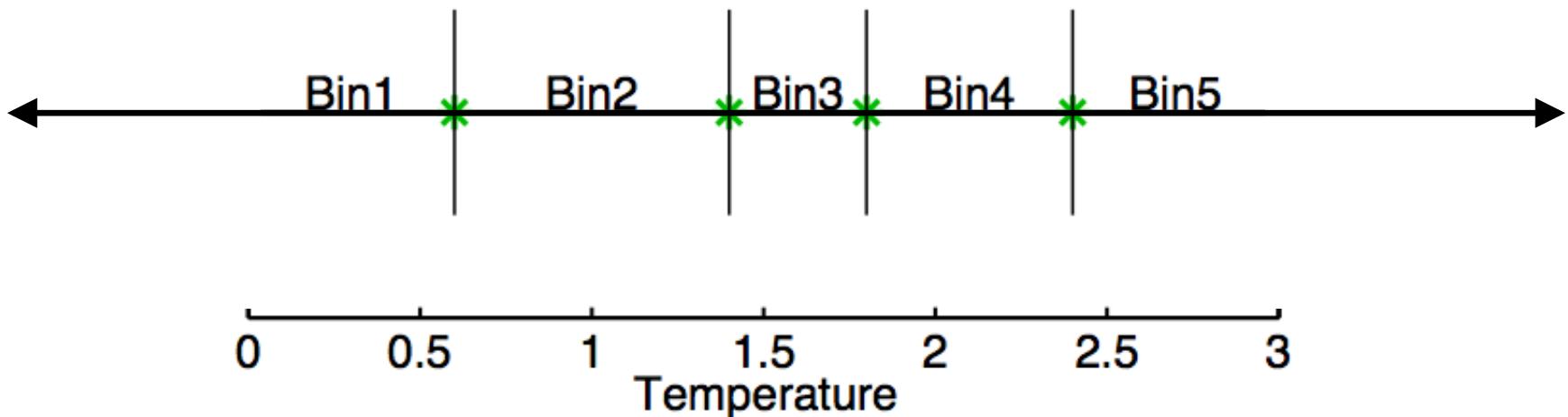
The Rank Histogram: Evaluating Filter Performance

Draw 5 values from a real-valued distribution.
Call the first 4 ‘ensemble members’.



The Rank Histogram: Evaluating Filter Performance

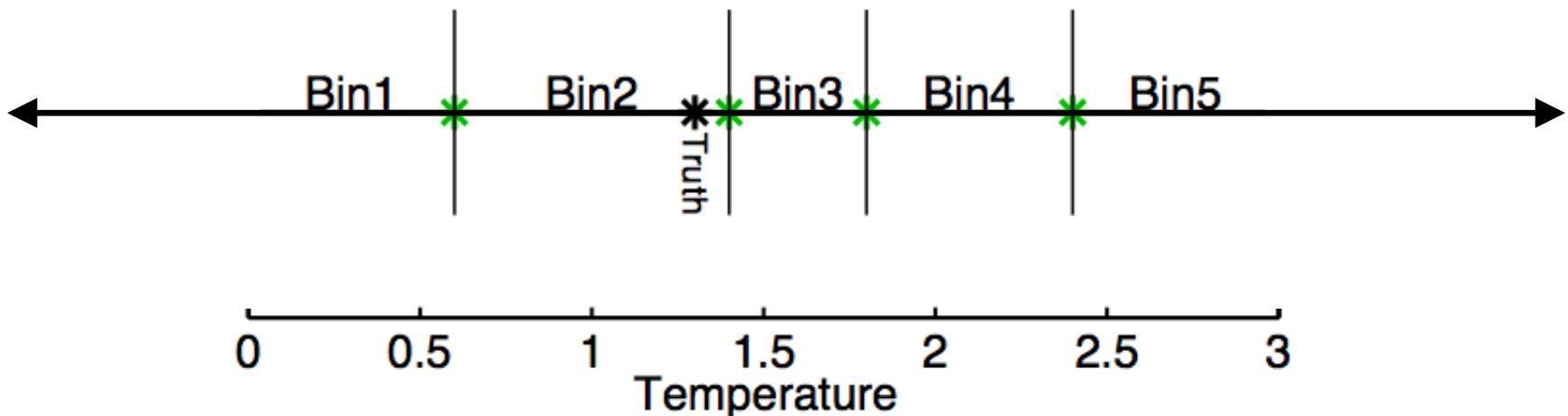
These 4 ‘ensemble members’ partition the real line into 5 bins.



The Rank Histogram: Evaluating Filter Performance

Call the 5th draw the ‘truth’.

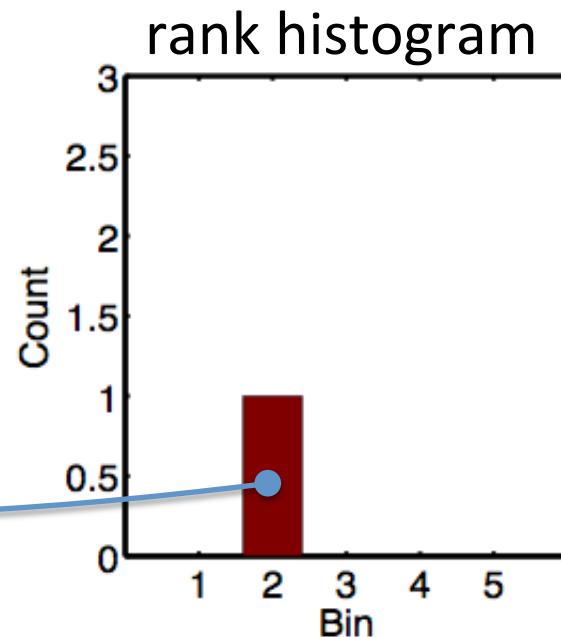
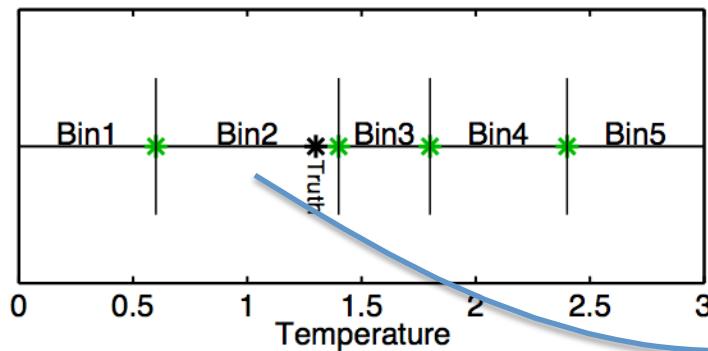
1/5 chance that this is in any given bin.



The Rank Histogram: Evaluating Filter Performance

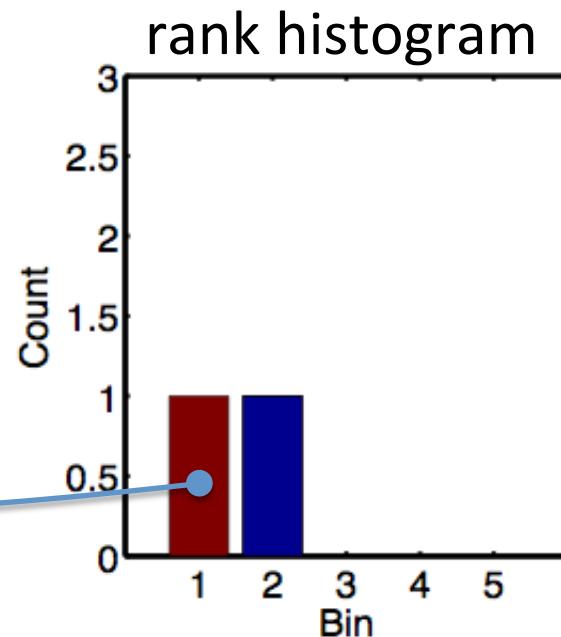
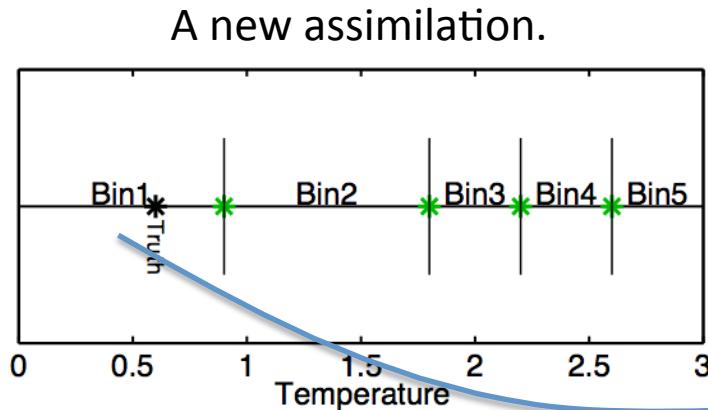
Rank histogram shows the frequency of the truth in each bin over many assimilations.

Same figure as previous slide.



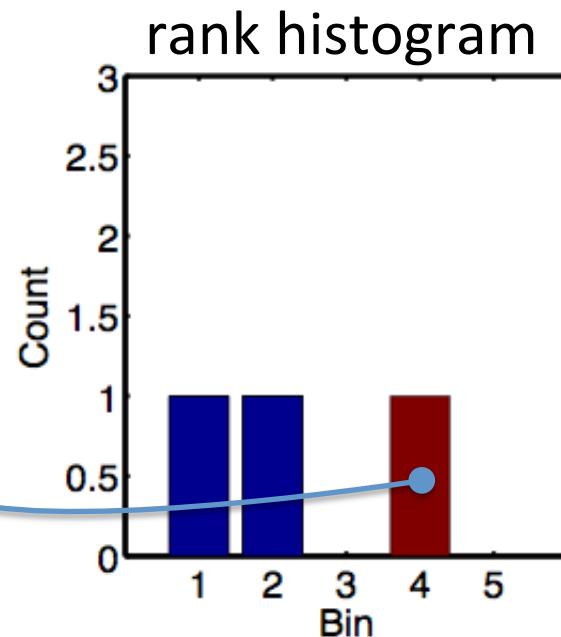
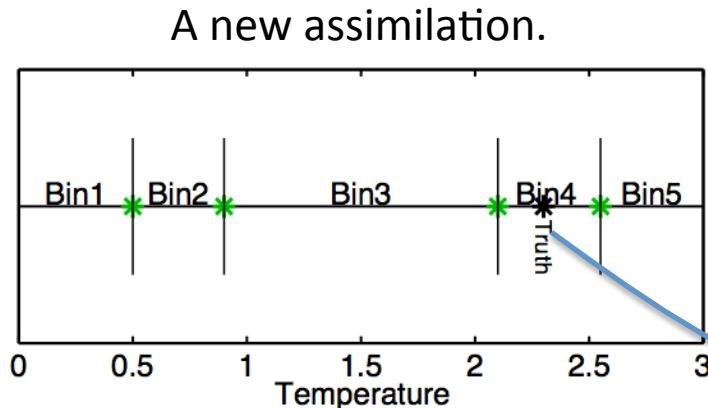
The Rank Histogram: Evaluating Filter Performance

Rank histogram shows the frequency of the truth in each bin over many assimilations.



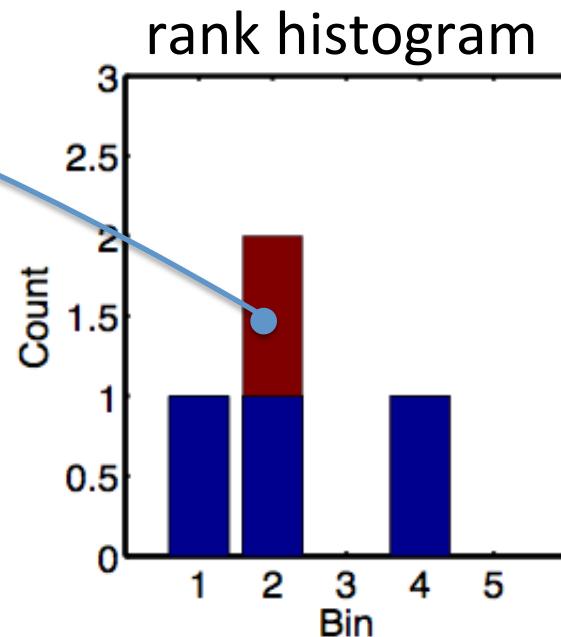
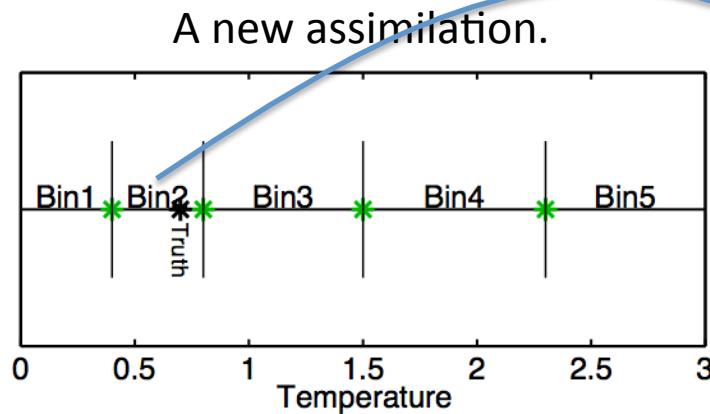
The Rank Histogram: Evaluating Filter Performance

Rank histogram shows the frequency of the truth in each bin over many assimilations.



The Rank Histogram: Evaluating Filter Performance

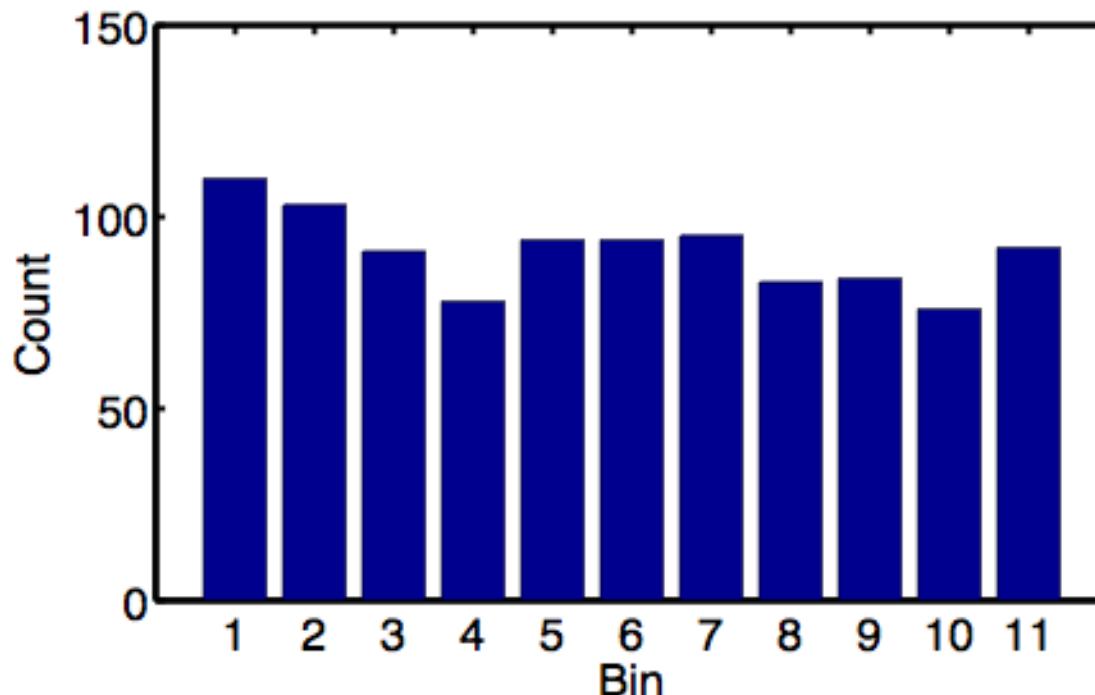
Rank histogram shows the frequency of the truth in each bin over many assimilations.



The Rank Histogram: Evaluating Filter Performance

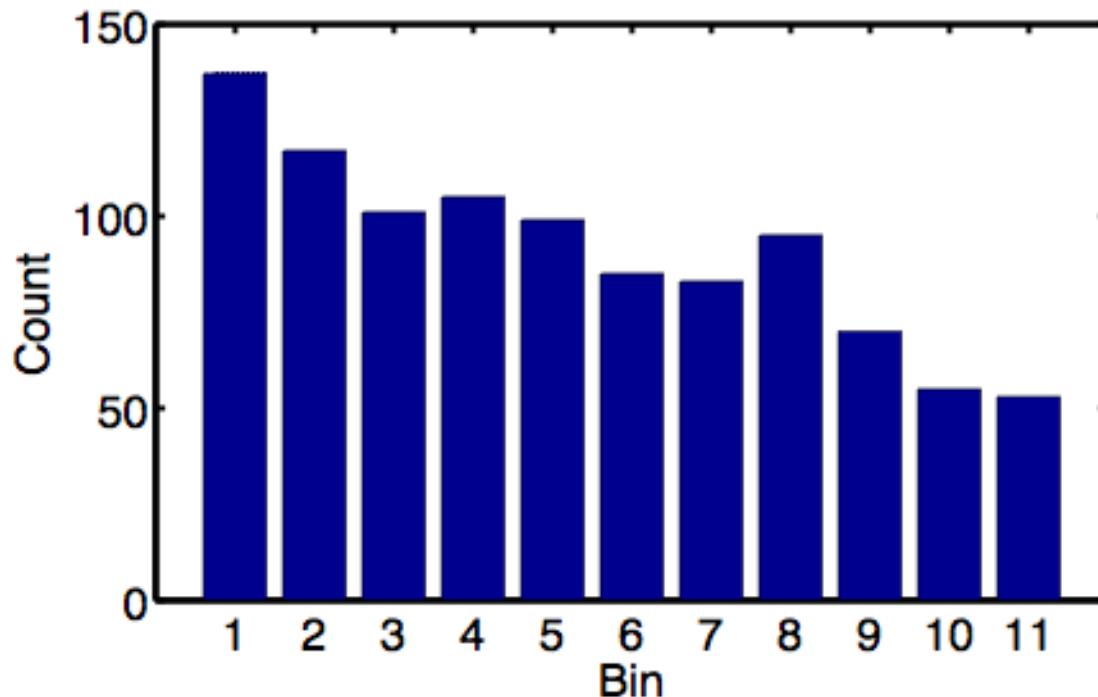
Rank histograms for good ensembles should be uniform (caveat sampling noise).

Want truth to look like random draw from ensemble.



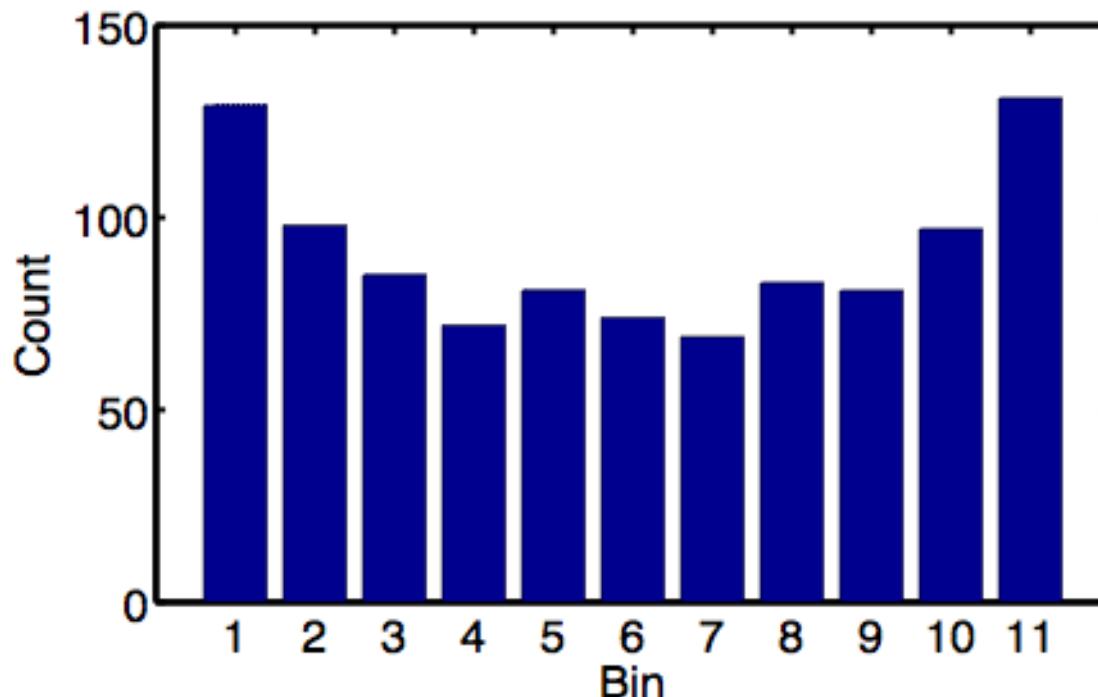
The Rank Histogram: Evaluating Filter Performance

A biased ensemble leads to skewed histograms.



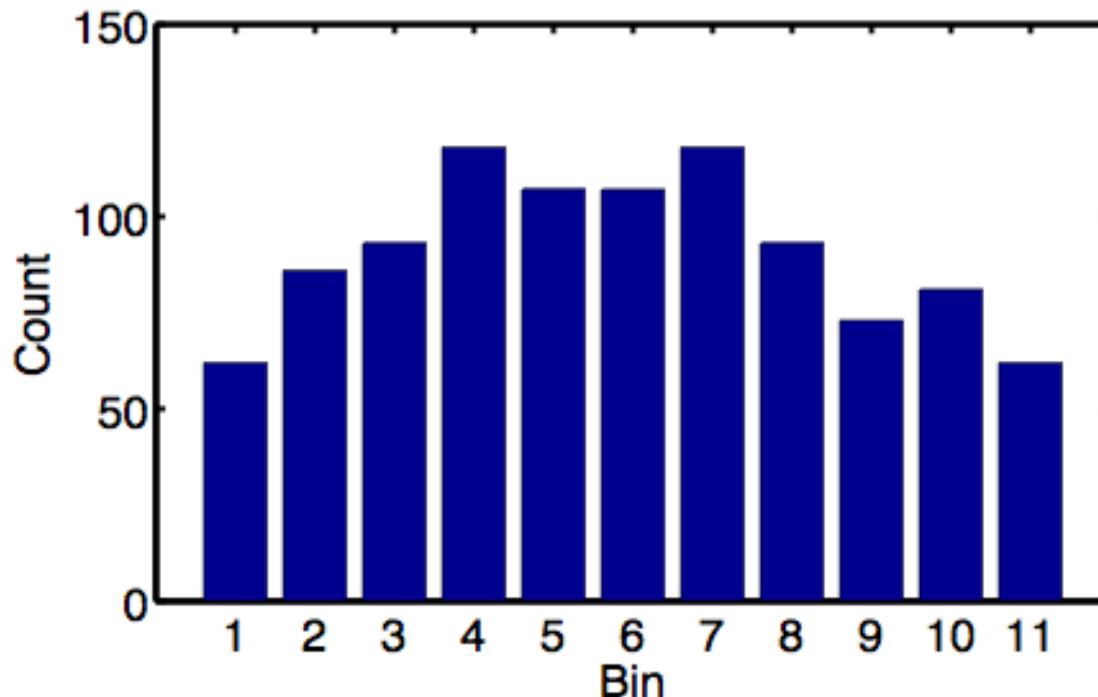
The Rank Histogram: Evaluating Filter Performance

An ensemble with too little spread gives a u-shape.
This is the most common behavior for geophysics.



The Rank Histogram: Evaluating Filter Performance

An ensemble with too much spread is
peaked in the center.



Matlab Hands-On: oned_model

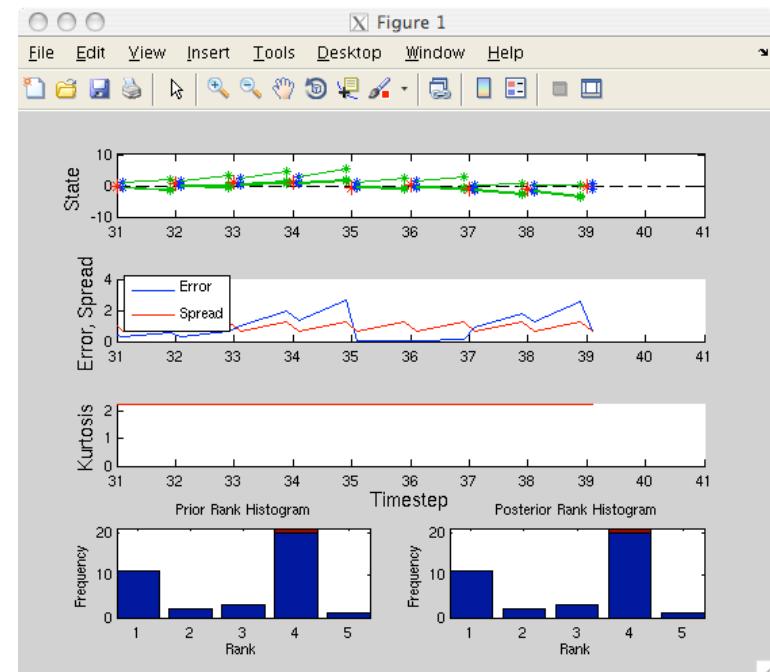
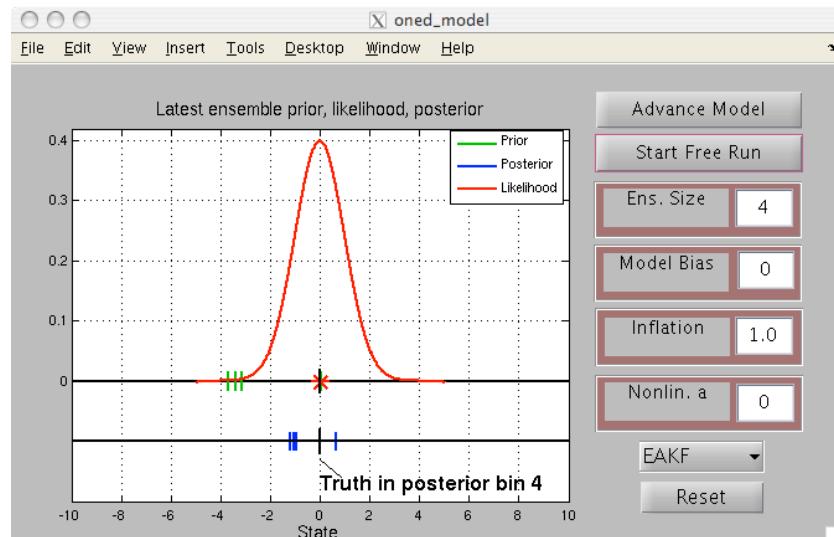
Understanding the Rank Histogram

Procedure:

1. The ensemble size can be changed with a dialog box.
2. A model bias can be set with a dialog box.
3. An additional non-linear term can be added to the model with a dialog box.

Note: Model bias is $dx/dt = x + \text{bias}$.

- Non-linear model is $dx/dt = x + a |x|$. Super-exponential growth.
- Truth is still always 0.
- See matlab script *advance_oned.m* for details.



Matlab Hands-On: oned_model

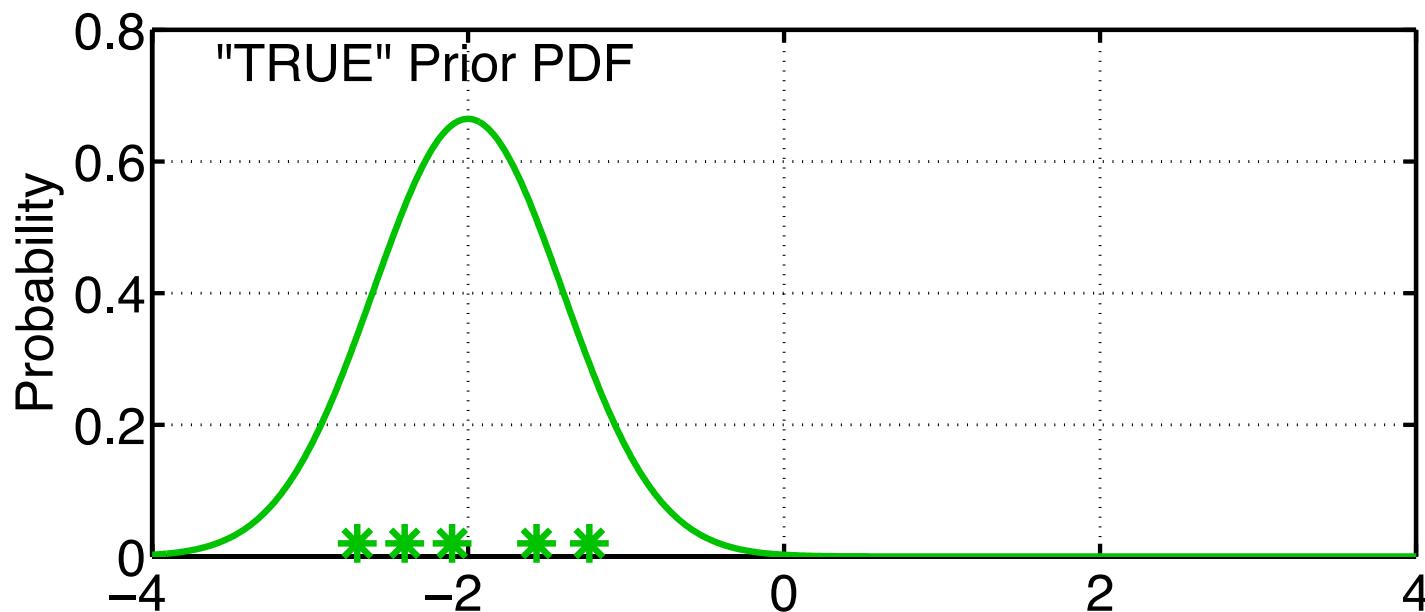
Understanding the Rank Histogram

Explorations:

1. Step through a sequence of advances and assimilations with the top button. Watch the evolution of the rank histogram bins.
2. Add some model bias (less than 1 to start) and see how the filter responds.
3. Add some nonlinearity (< 1) to the model. How do the different filters respond?
4. Can you break the filter (find setting so that the ensemble moves away from zero) with the options explored so far?

Dealing with systematic error: Variance Inflation

Observations + physical system → ‘true’ distribution.

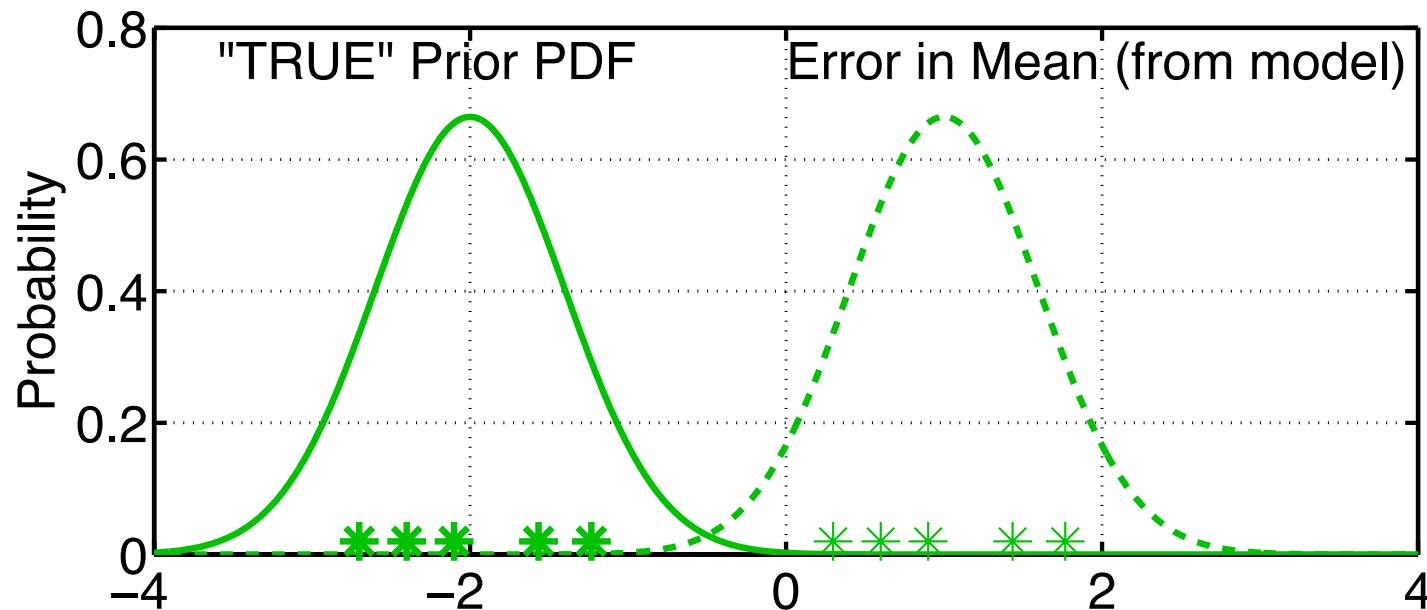


Dealing with systematic error: Variance Inflation

Observations + physical system → ‘true’ distribution.

Model bias (and other errors) can shift actual prior.

Prior ensemble is too certain (needs more spread).

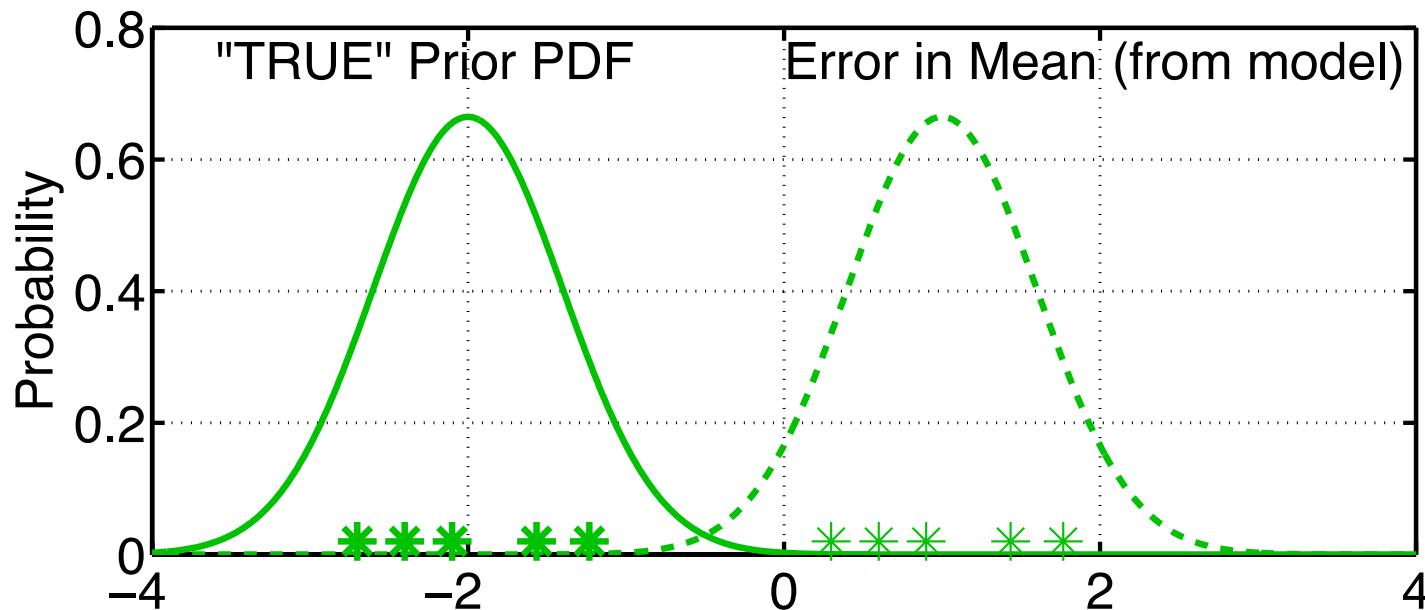


Dealing with systematic error: Variance Inflation

Could correct error if we knew what it was.

With large models, can't know error precisely.

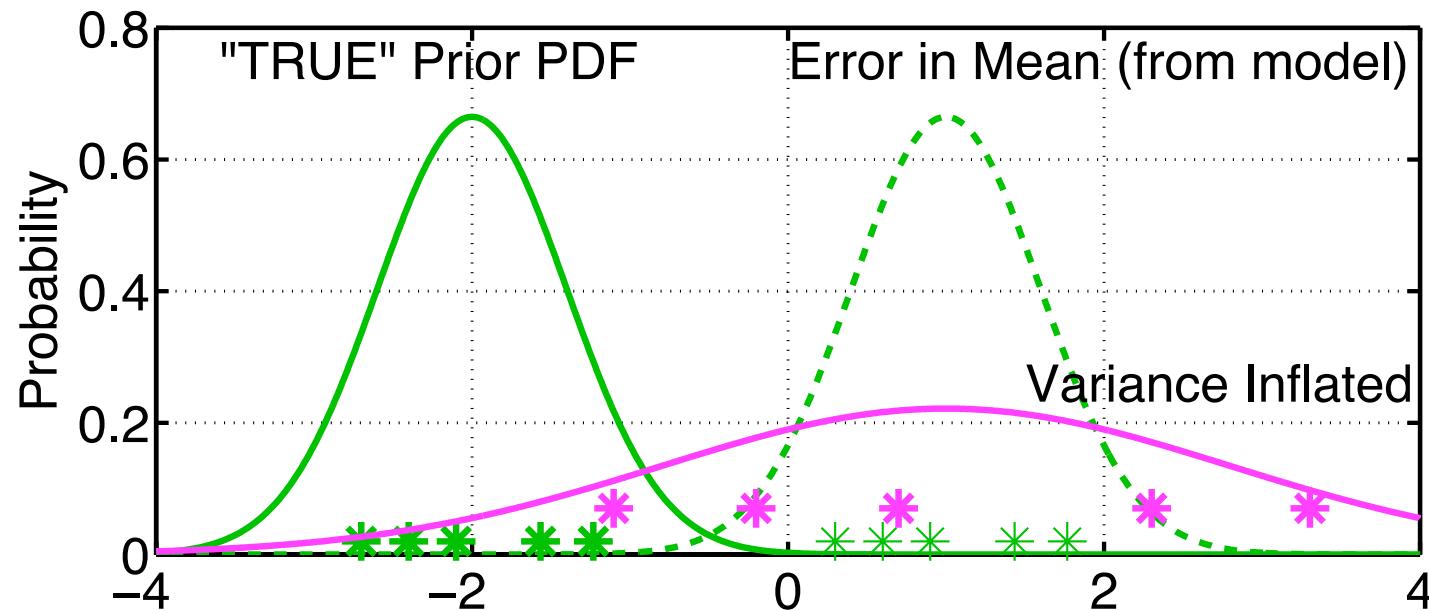
Taking no action can cause observations to be ignored.



Dealing with systematic error: Variance Inflation

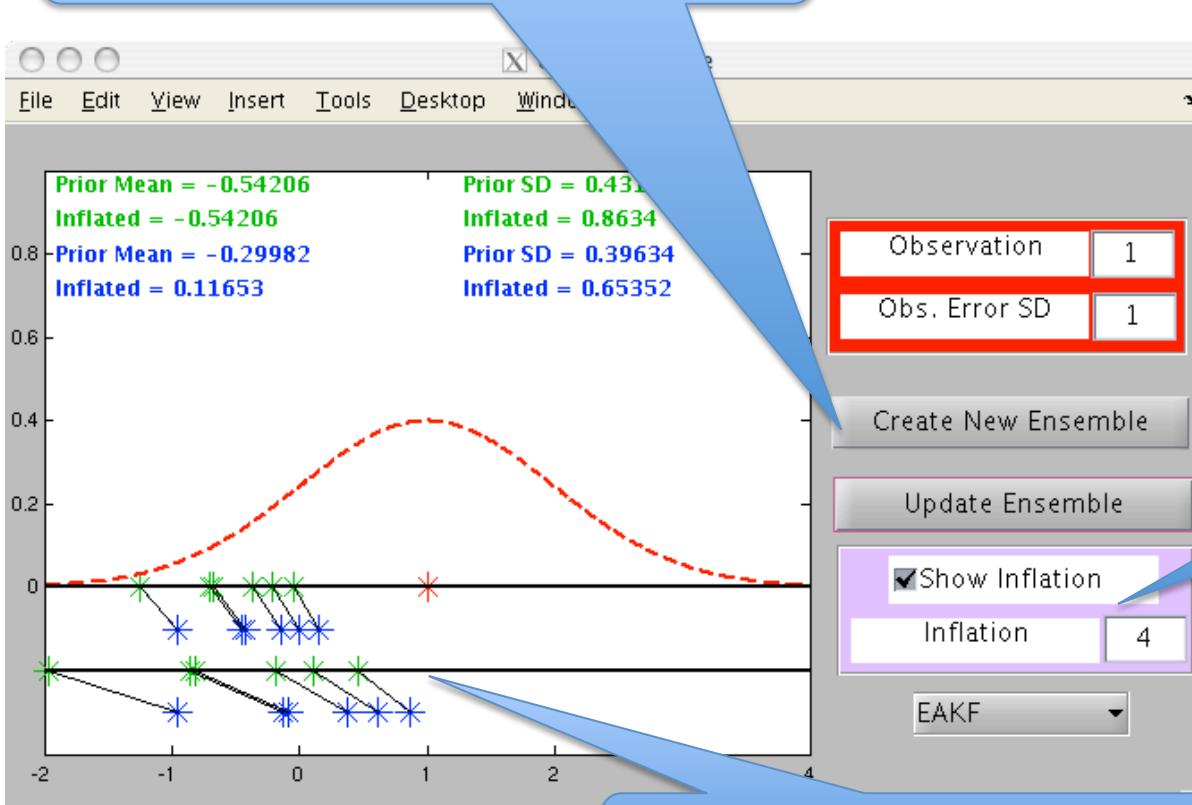
Naïve solution: increase the spread in the prior.

Give more weight to the observation, less to the prior.



Matlab Hands-On: oned_ensemble exploring prior inflation

1) Create a new ensemble.



2) Set an inflation value
and turn on.

3) The inflated prior and
assimilation shows up here.

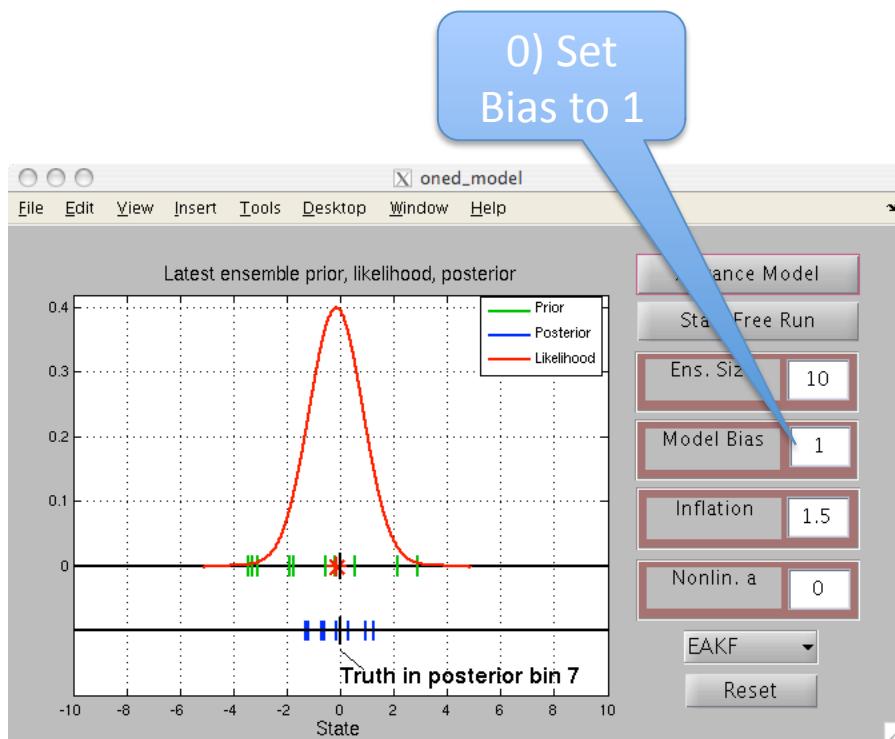
Matlab Hands-On: exploring prior inflation with oned_ensemble

Explorations:

- See how increasing inflation (> 1) changes the posterior mean and standard deviation.
- Look at priors that are not shifted but have small spread compared to the observation error distribution.
- Look at priors that are shifted from the observation.

Matlab Hands-On: oned_model

using inflation to deal with systematic error



Remember:

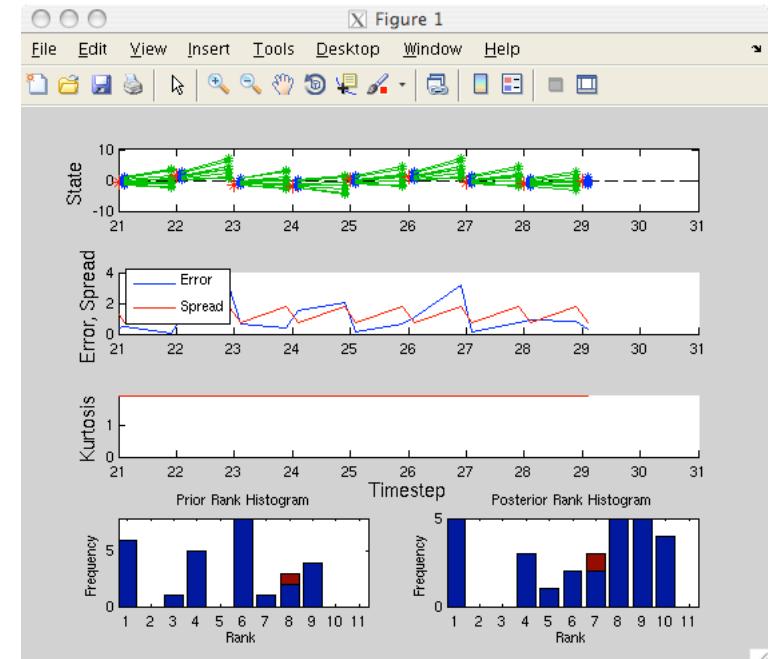
This uses the simple linear model $dx/dt = x$.

The ‘truth’ is always 0.

Observation noise is a draw from a $N(0,1)$.

The spread is increased by the square root of the inflation.

1. Run an assimilation and observe the error, spread, and rank histograms.
2. Add some inflation (try starting with 1.5) and observe how behavior changes.
3. What happens with too much inflation?



Matlab Hands-On: oned_model using inflation to deal with systematic error

Explorations:

- Try a variety of model bias and inflation settings.
- Try using inflation with a nonlinear model.

