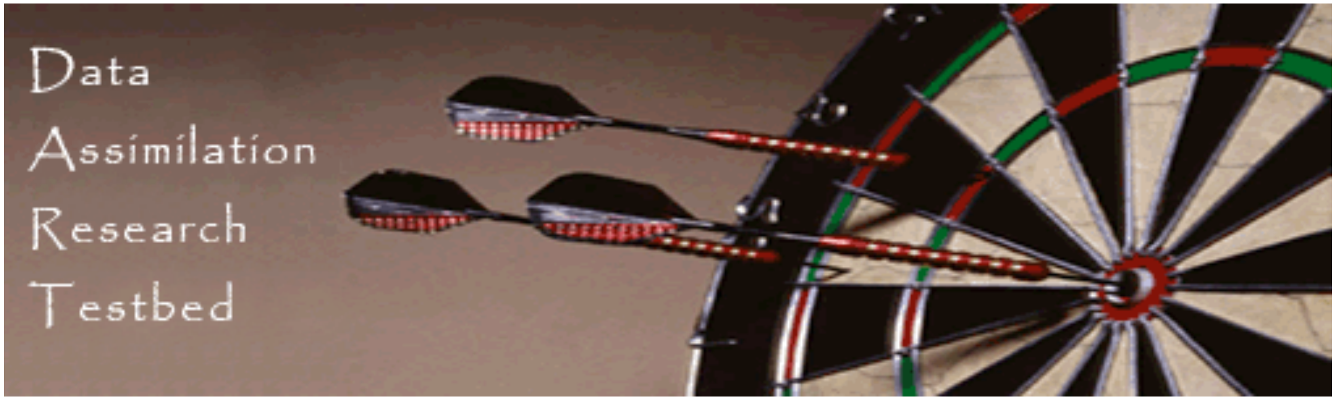
Data Assimilation Research Testbed Tutorial

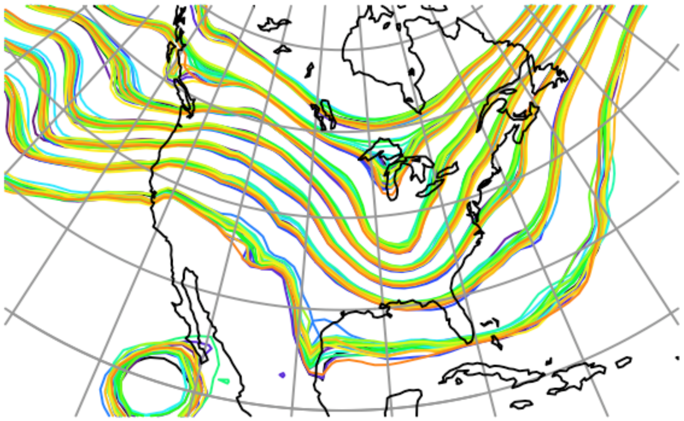


Section 1: Introduction

What is Data Assimilation?

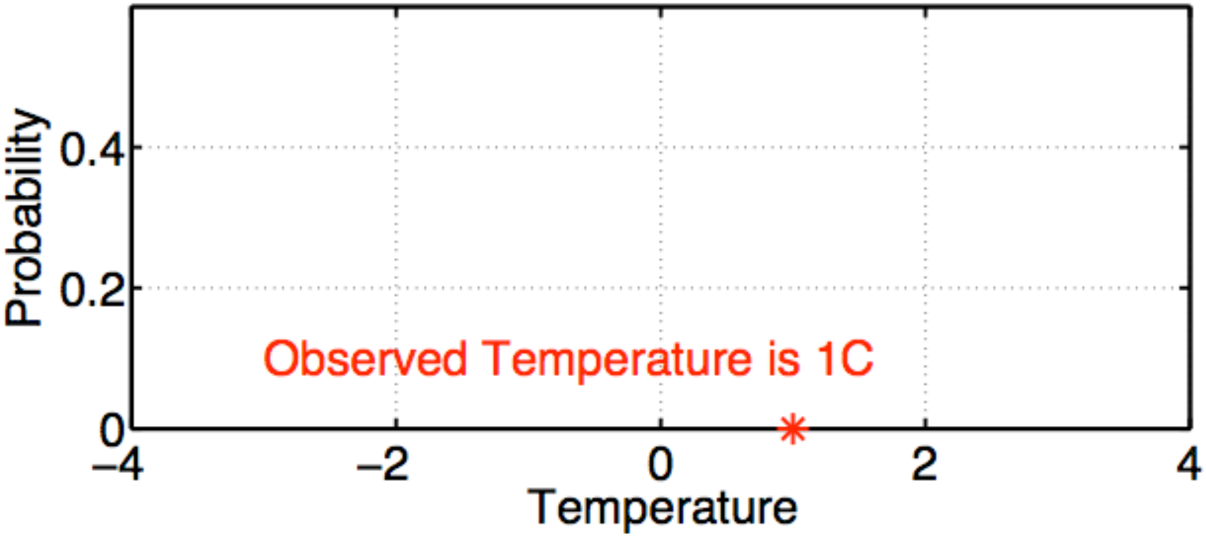
Observations combined with a Model forecast…

…to produce an analysis
(best possible estimate).



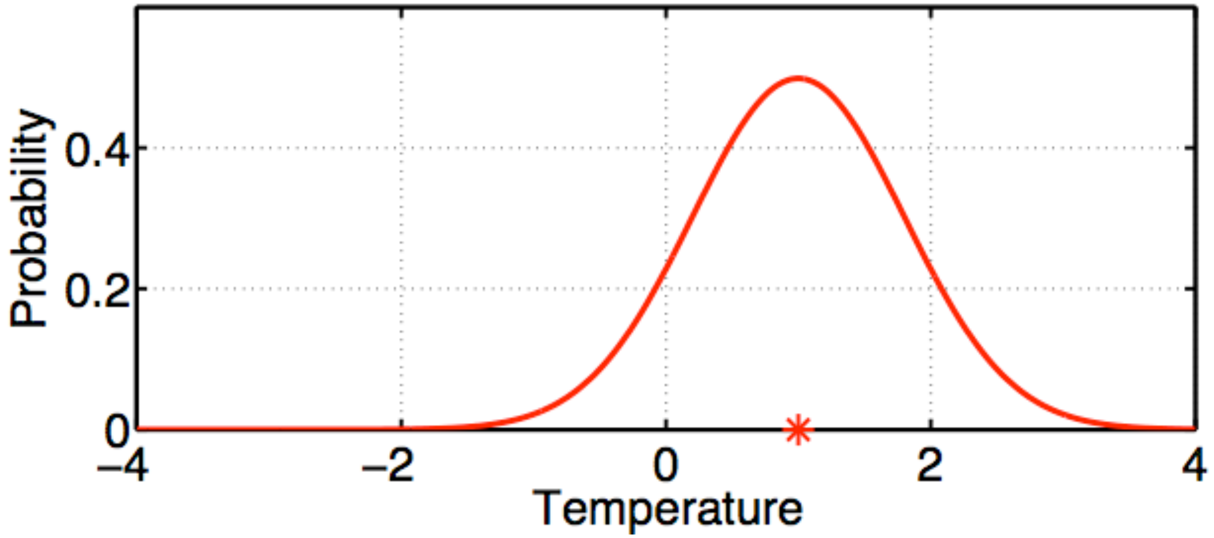
Example: Estimating the Temperature Outside

An observation has a value (∗),



Example: Estimating the Temperature Outside

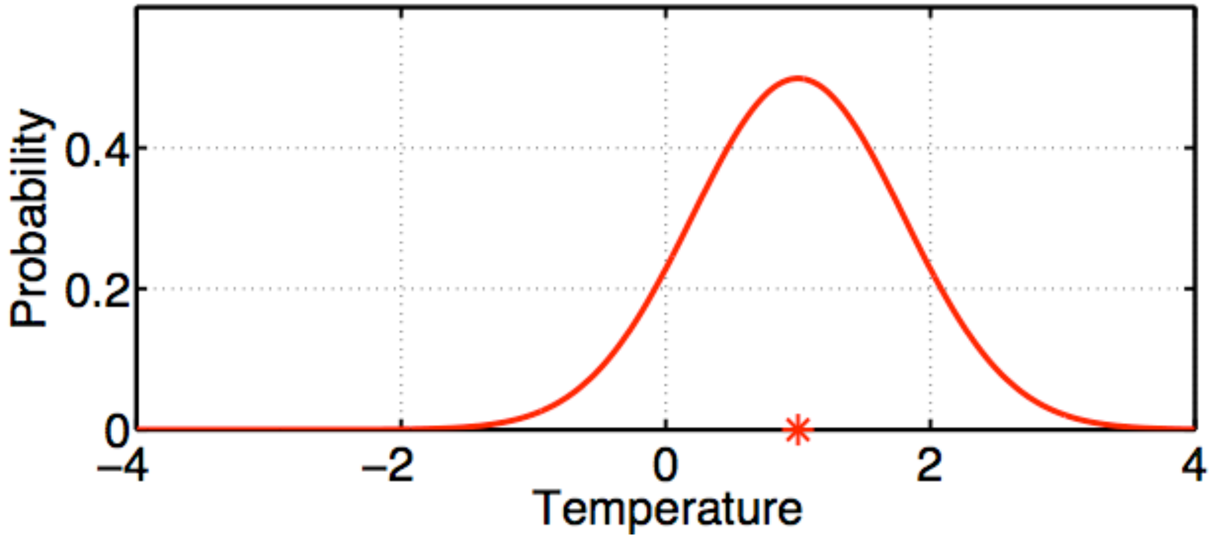
An observation has a value (*),



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

Example: Estimating the Temperature Outside

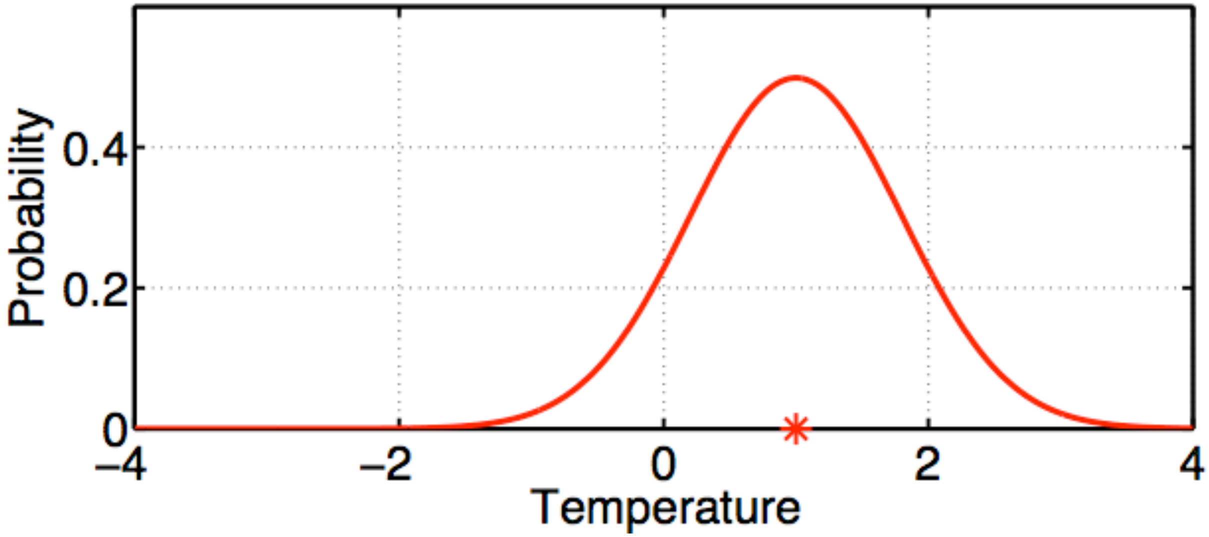
Thermometer outside measures 1C.



Instrument builder says thermometer is unbiased with +/- 0.8C gaussian error.

Example: Estimating the Temperature Outside

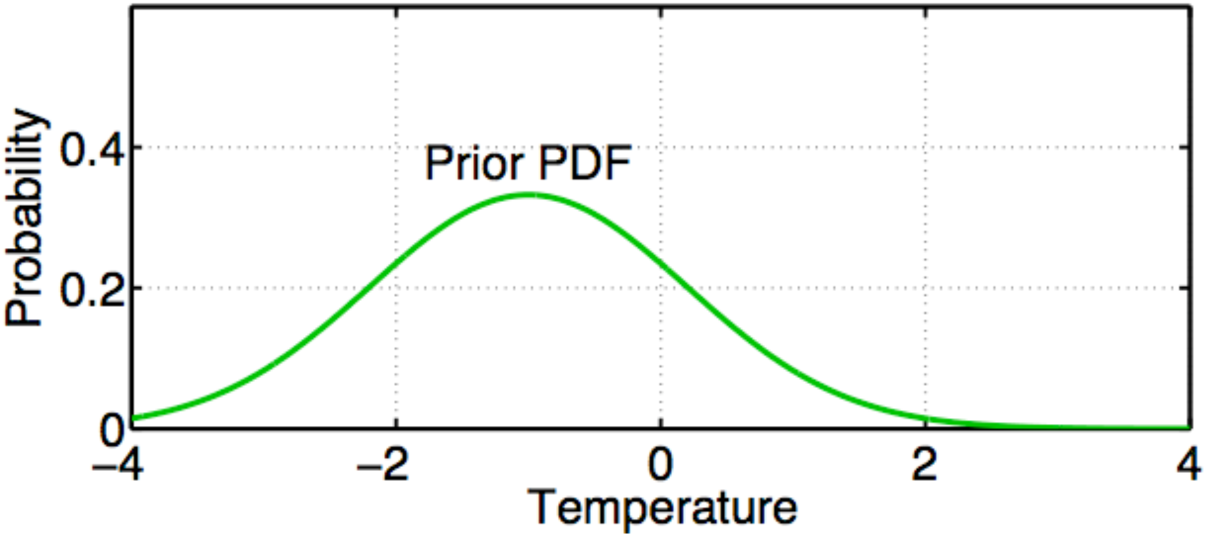
Thermometer outside measures 1C.



The red plot is $P(T \mid T_o)$, 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 $P(T \mid C)$; 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 > -273.15C);
4. Climatological constraints (-30C < T < 40C).

Combining the Prior Estimate and Observation

Bayes Theorem:

$$P\left(T \mid T_o, C\right) = \frac{P\left(T_o \mid T, C\right) P\left(T \mid C\right)}{P\left(T_o \mid C\right)}$$

Prior

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

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

Combining the Prior Estimate and Observation

Rewrite Bayes as:

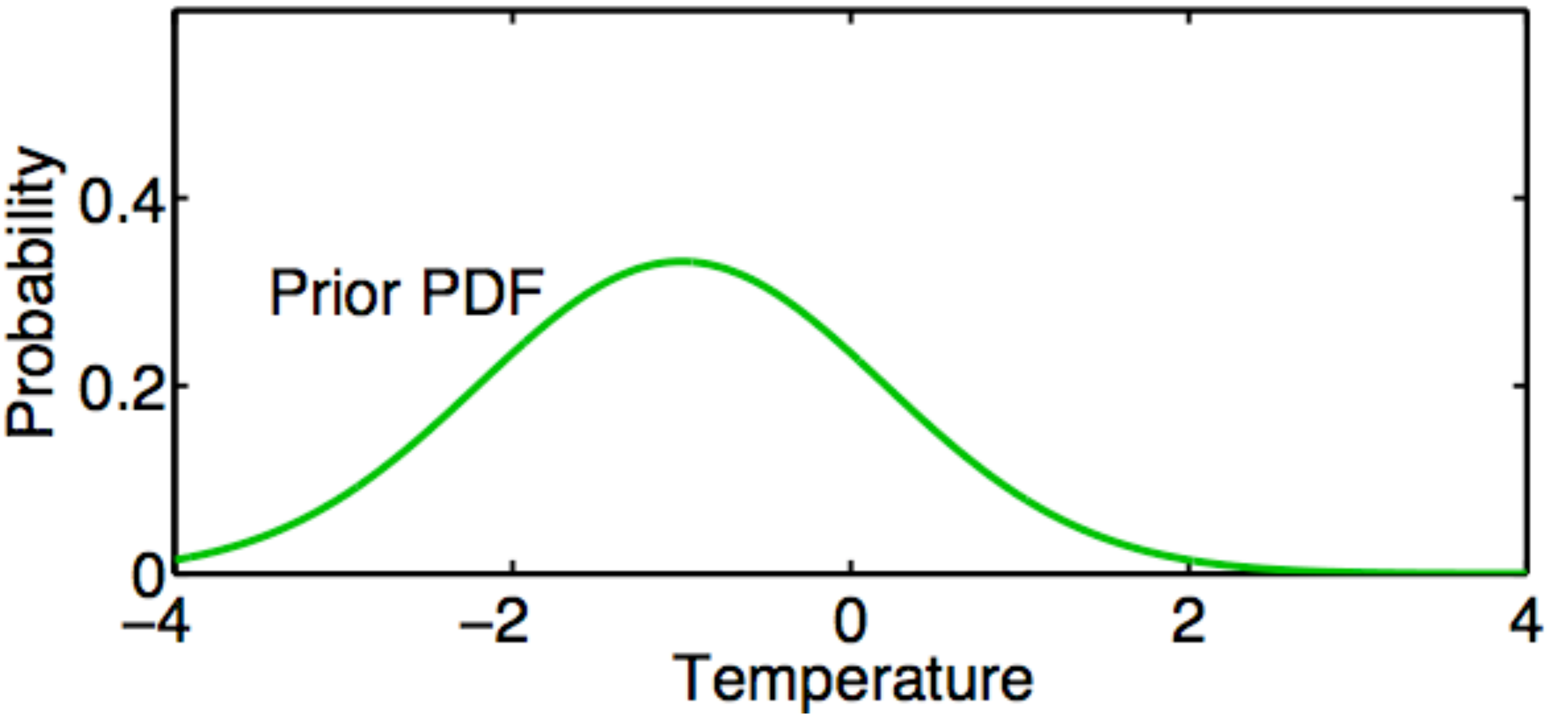
$$\frac{P(T_o \mid T,C)P(T \mid C)}{P(T_o \mid C)} = \frac{P(T_o \mid T,C)P(T \mid C)}{\int P(T_o \mid x)P(x \mid C)dx}$$

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

Denominator normalizes so Posterior is PDF.

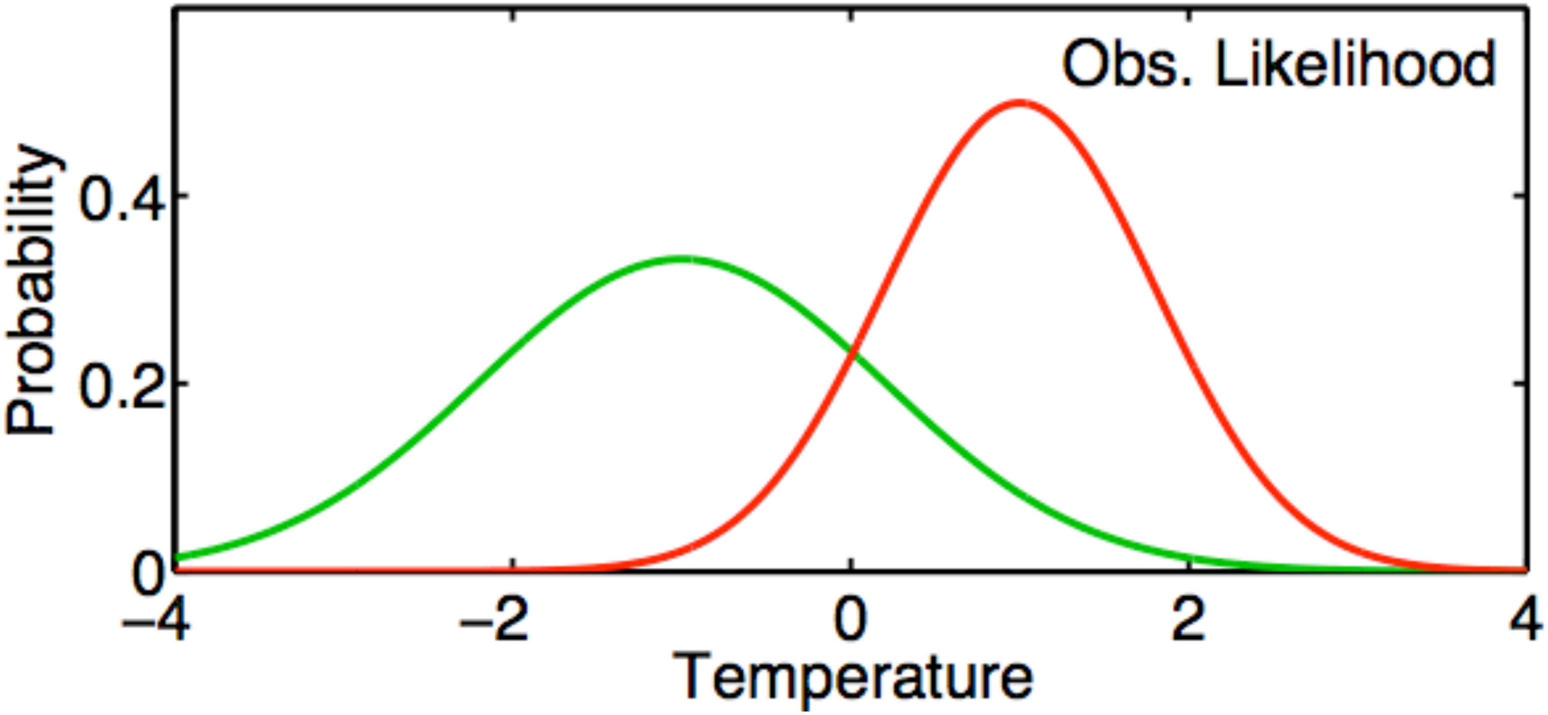
Combining the Prior Estimate and Observation

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



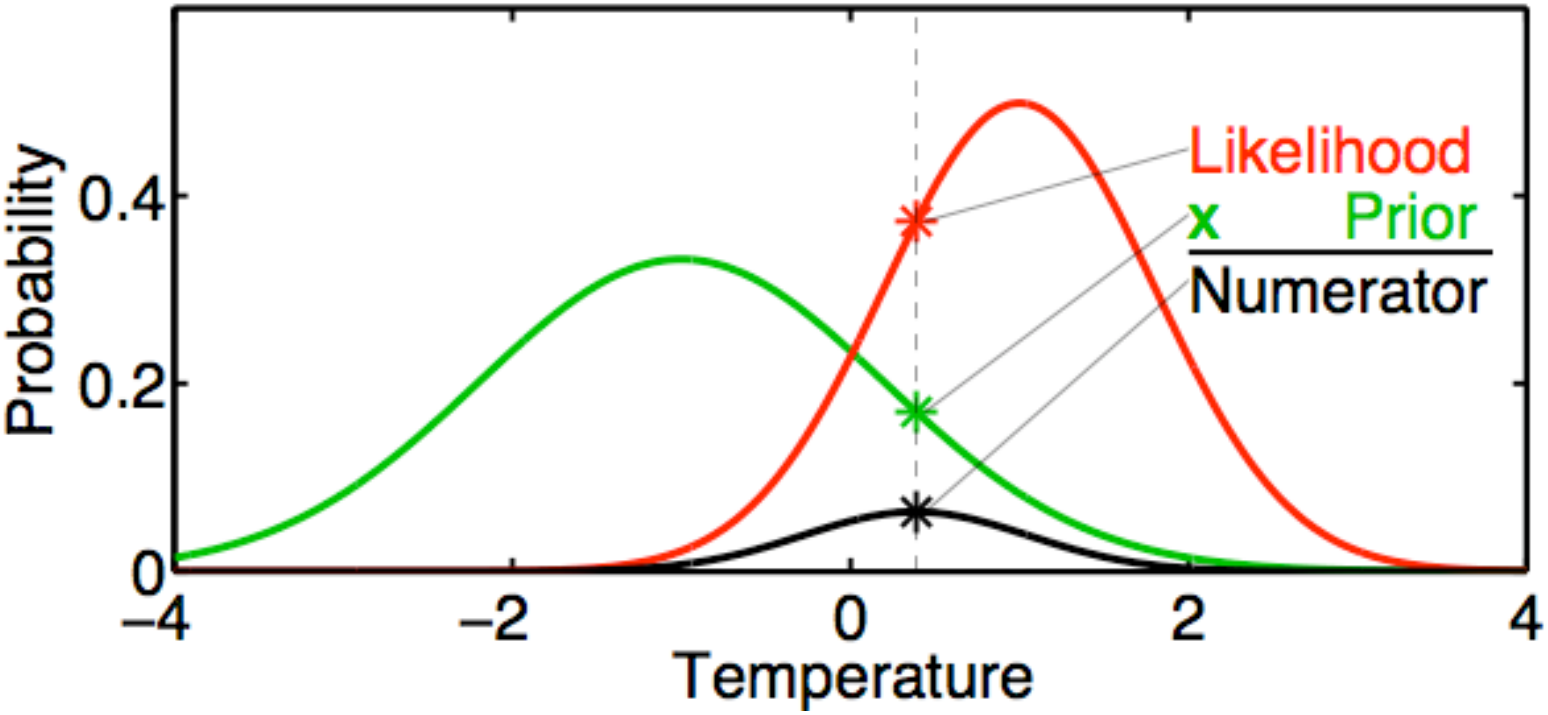
Combining the Prior Estimate and Observation

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



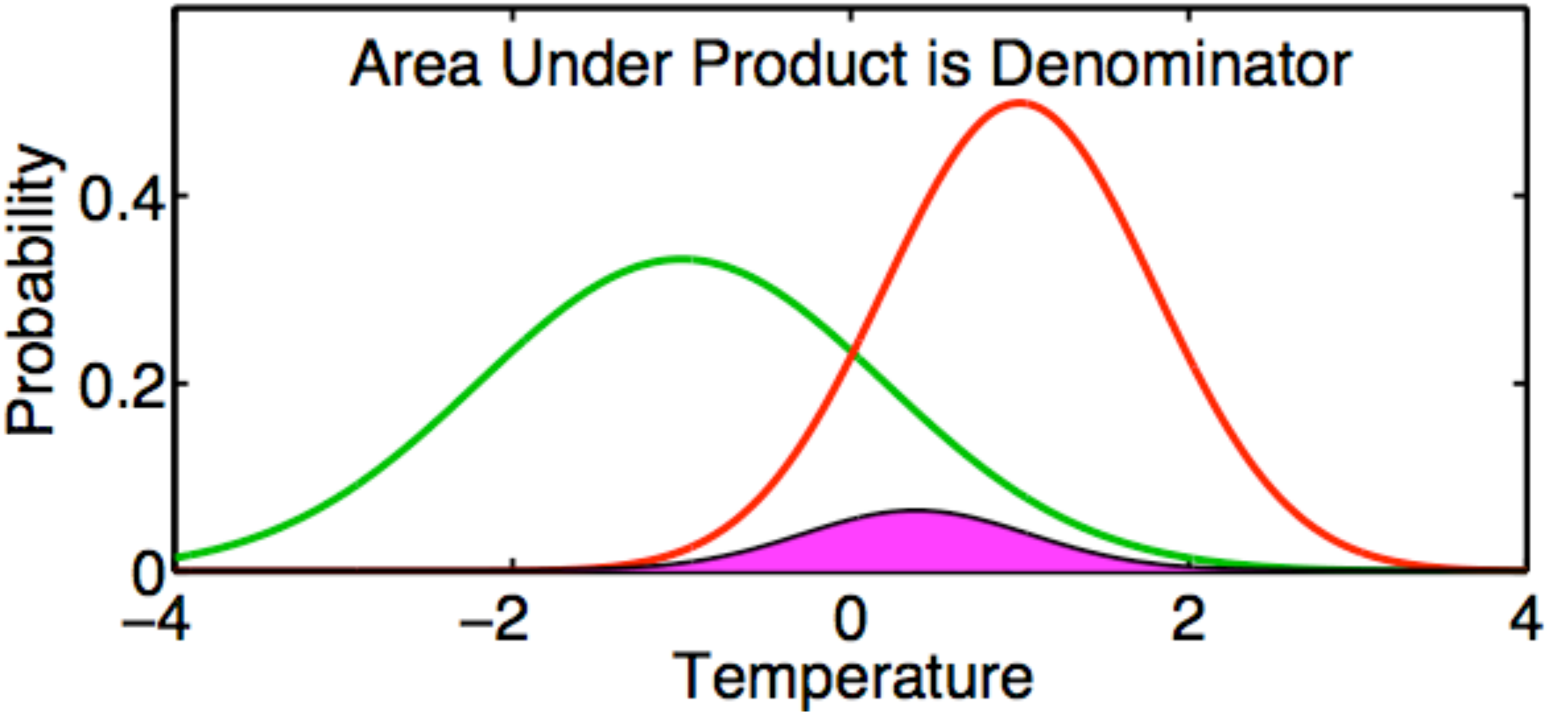
Combining the Prior Estimate and Observation

$$P(T \mid T_o, C) = \frac{\boxed{P(T_o \mid T, C)P(T \mid C)}}{normalization}$$



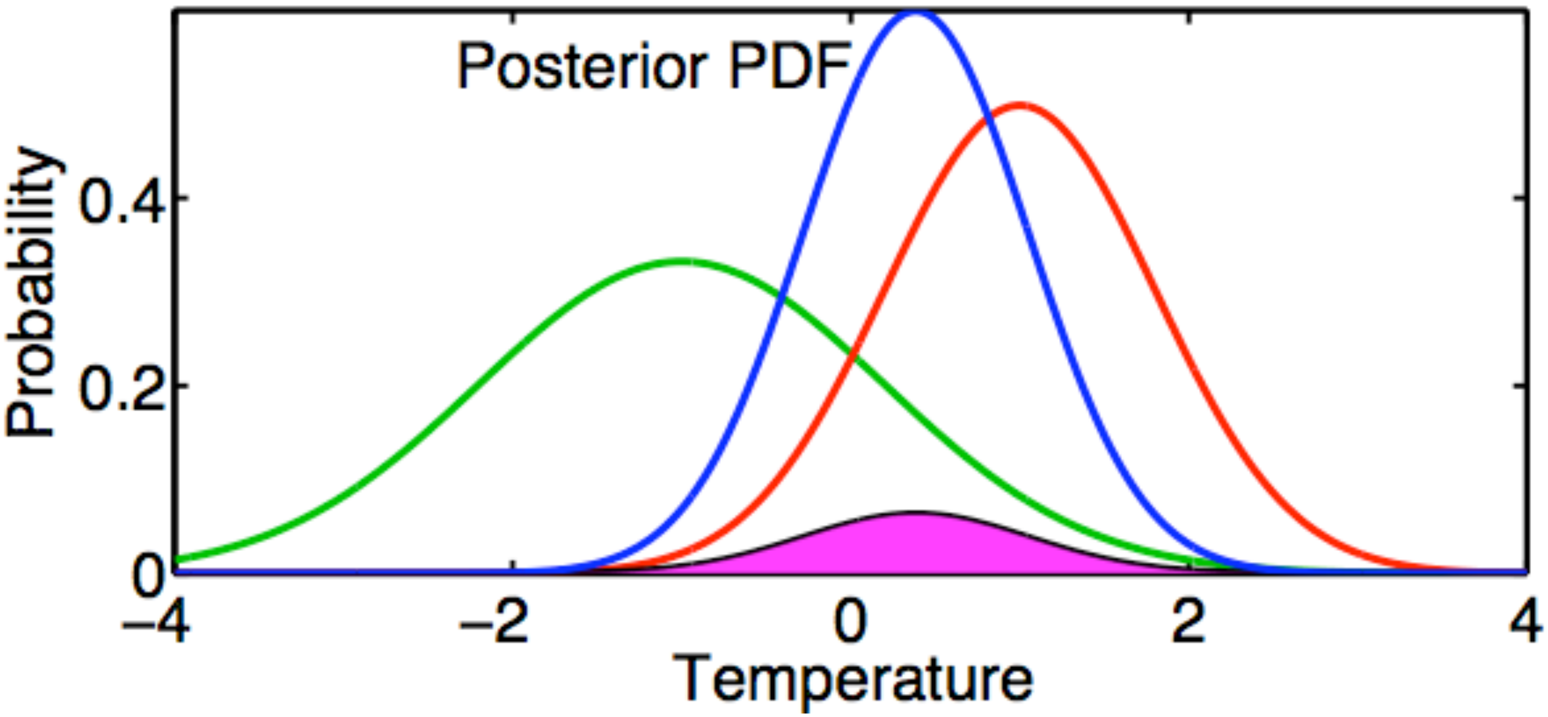
Combining the Prior Estimate and Observation

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



Combining the Prior Estimate and Observation

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



Consistent Color Scheme Throughout Tutorial

Green = Prior

Red = Observation

Blue = Posterior

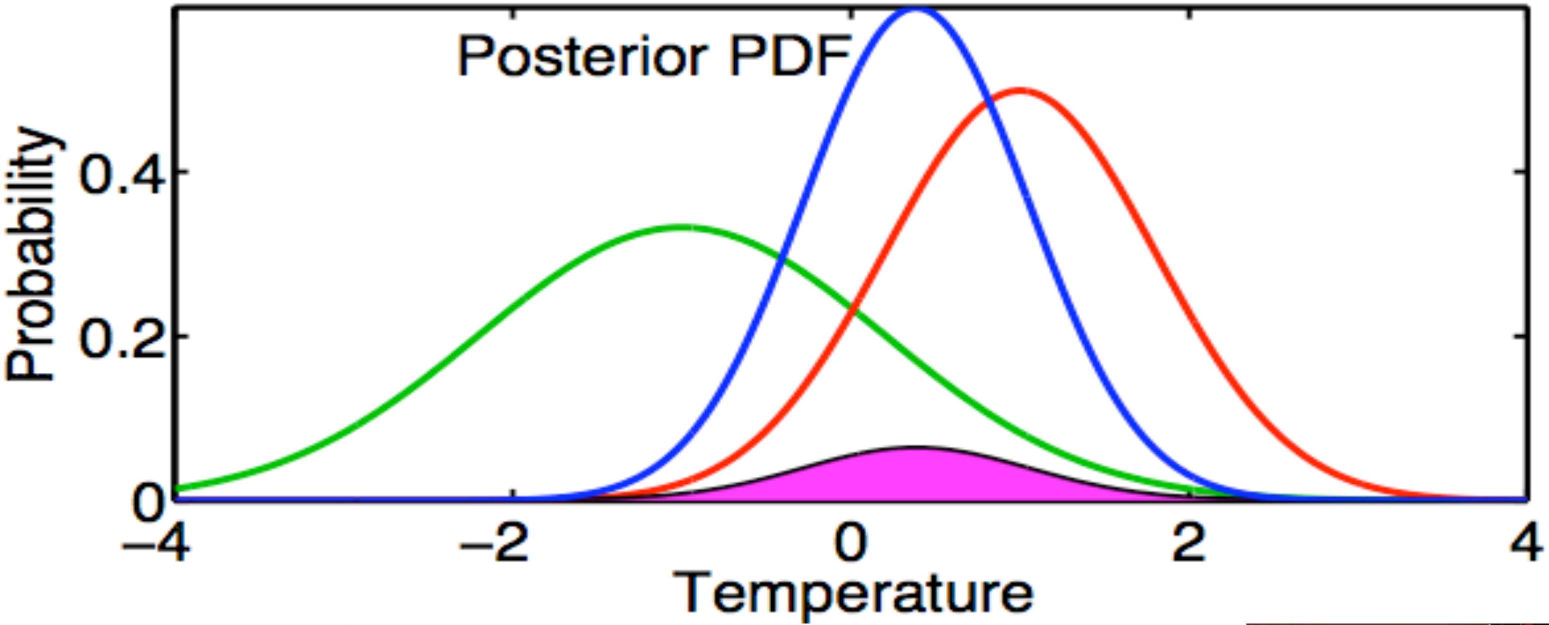
Black = Truth

(truth available only for ‘perfect model’ examples)

Combining the Prior Estimate and Observation

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

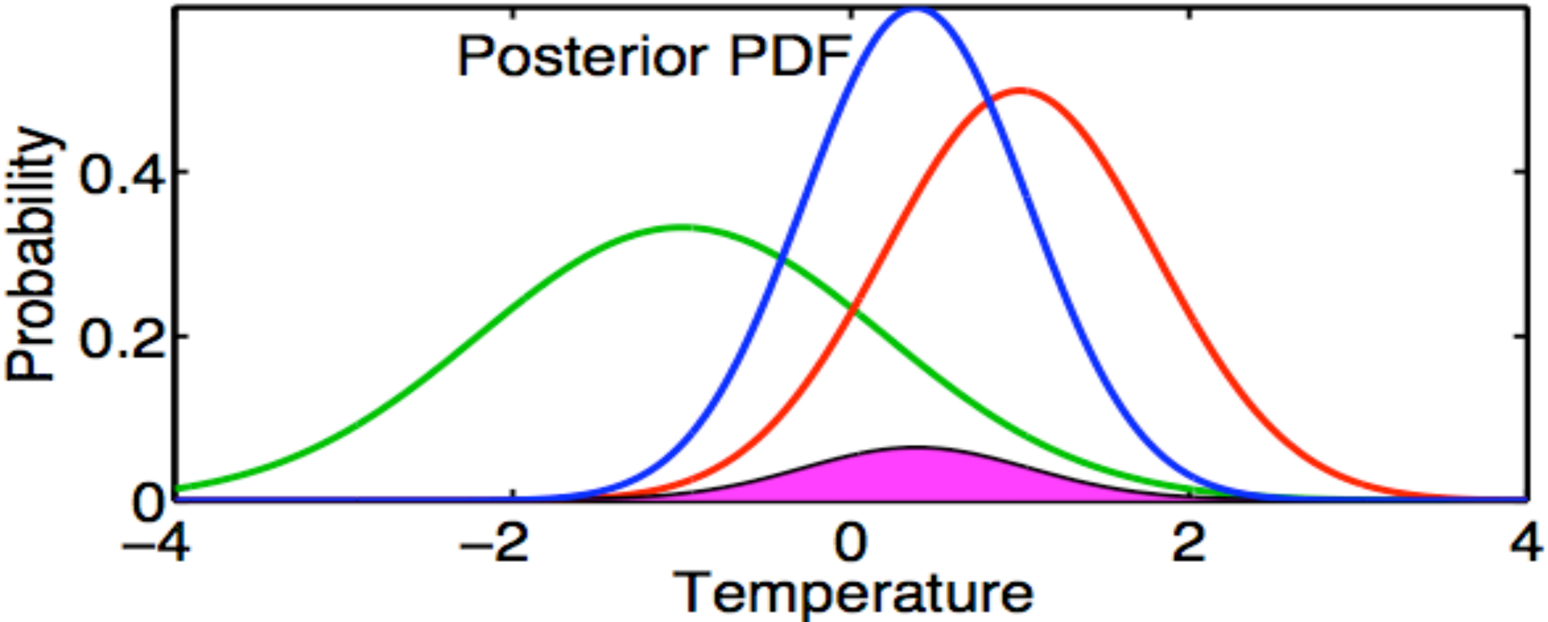
Generally no analytic solution for Posterior.



Combining the Prior Estimate and Observation

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

Gaussian Prior and Likelihood -> Gaussian Posterior



Combining the Prior Estimate and Observation

For Gaussian prior and likelihood…

Prior $P(T \mid C) = Normal(T_p, \sigma_p)$

Likelihood $P(T_o \mid T, C) = Normal(T_o, \sigma_o)$

Then, Posterior $P(T \mid T_o, C) = Normal(T_u, \sigma_u)$

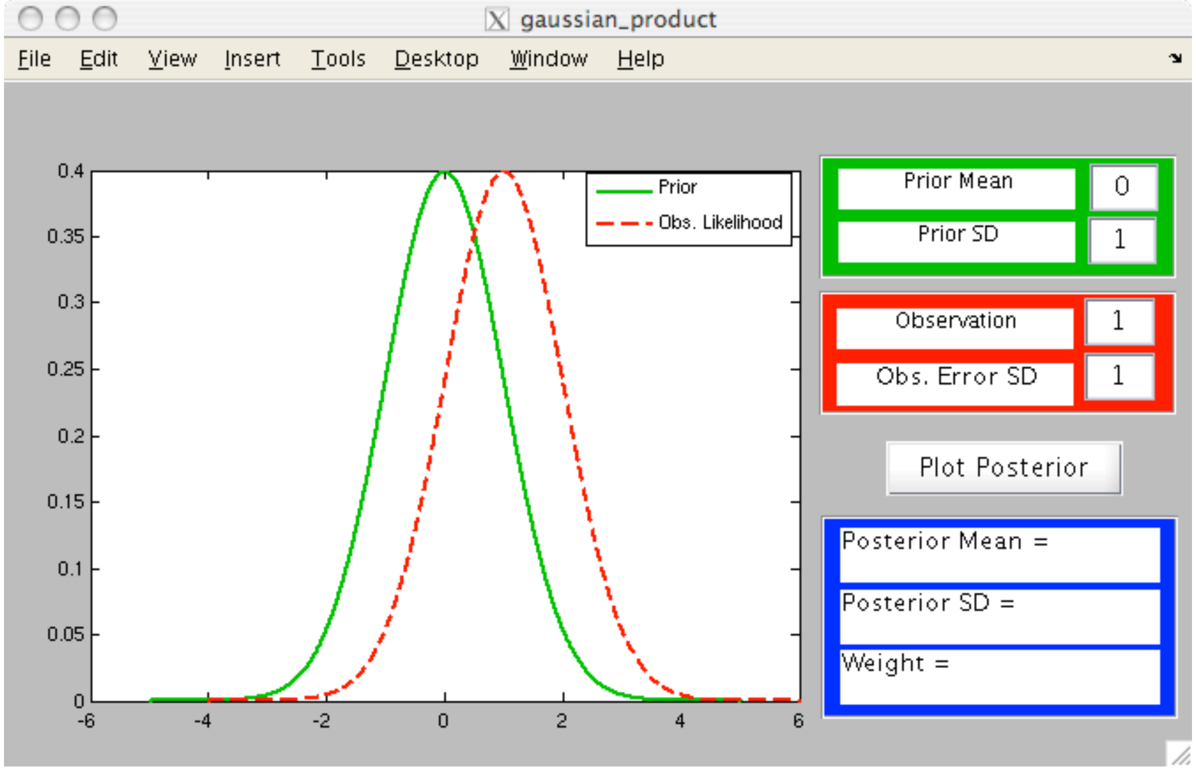
With

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

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

Matlab Hands-On: gaussian_product

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



Prior Mean 0

Prior SD 1

Observation 1

Obs. Error SD 1

Posterior Mean =

Posterior SD =

Weight =

Matlab Hands-On: gaussian_product

Procedure:

1. Use the green dialog boxes to set the prior mean and standard deviation.
2. Use the red dialog boxes to set the observation and the observation error standard deviation.
3. Select Plot Posterior to plot the posterior.

The blue boxes show the posterior mean and standard deviation and the weight for the product.

Matlab Hands-On: gaussian_product

Explorations:

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 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
      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
   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
      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
   prior at t_2 is *Normal*($T_{p,2}$, $\sigma_{p,2}$).

3. Given an observation at t_2 with distribution *Normal*(t_o, σ_o)
   the likelihood is also *Normal*(t_o, σ_o).

The One-Dimensional Kalman Filter

1. Suppose we have a linear forecast model L.

   A. If temperature at time $t_1 = T_{1,}$ then
   temperature at $t_2 = t_1 + \Delta t$ is $T_2 = L(T_1)$.

   B. Example: $T_2 = T_1 + \Delta t T_1$.

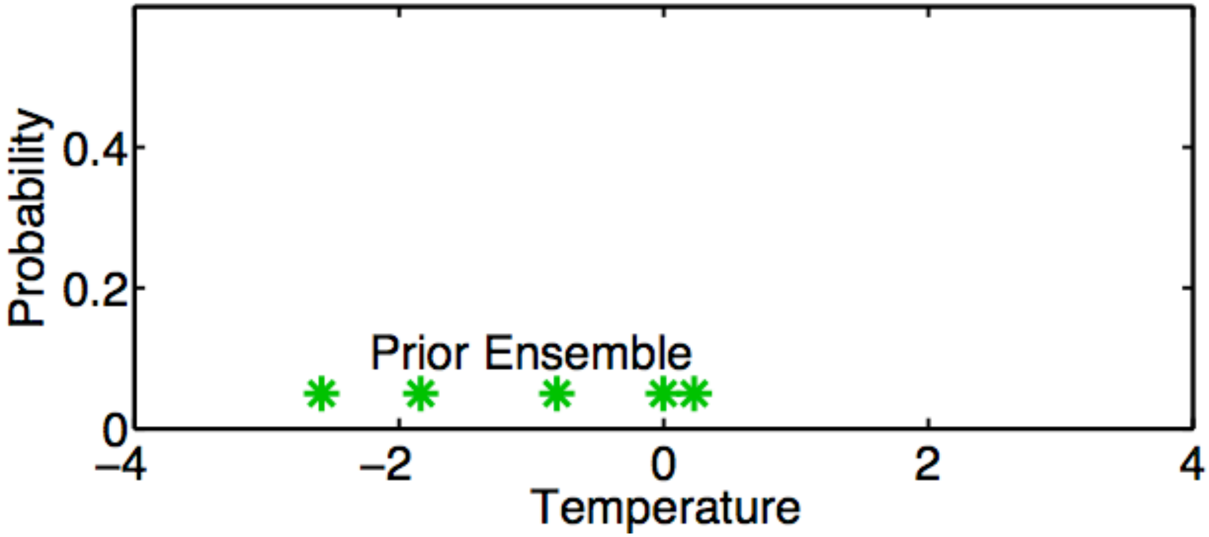
2. If posterior estimate at time t_1 is *Normal*($T_{u,1}$, $\sigma_{u,1}$) then
   prior at t_2 is *Normal*($T_{p,2}$, $\sigma_{p,2}$).

3. Given an observation at t_2 with distribution *Normal*(t_o, σ_o)
   the likelihood is also *Normal*(t_o, σ_o).

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.

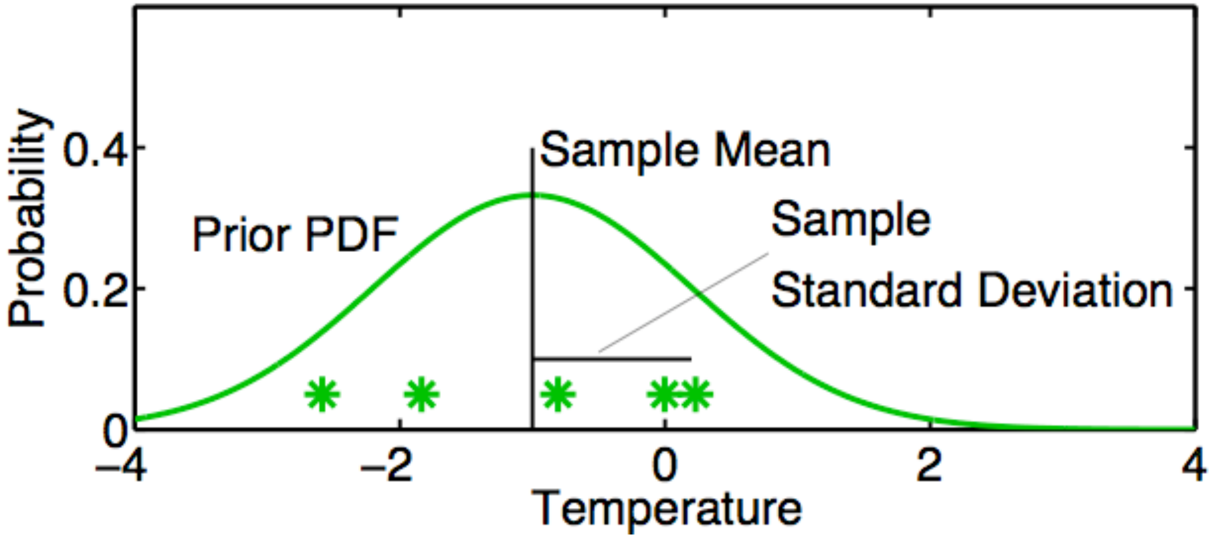
A One-Dimensional Ensemble Kalman Filter

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:



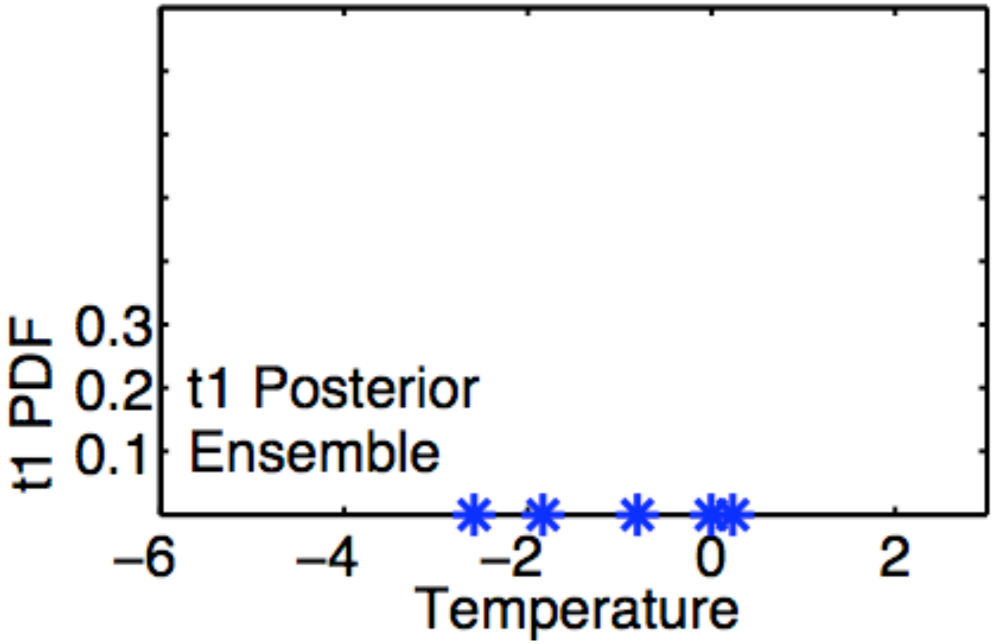
Use sample mean $\overline{T} = \sum_{n=1}^{N} T_n / N$

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

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

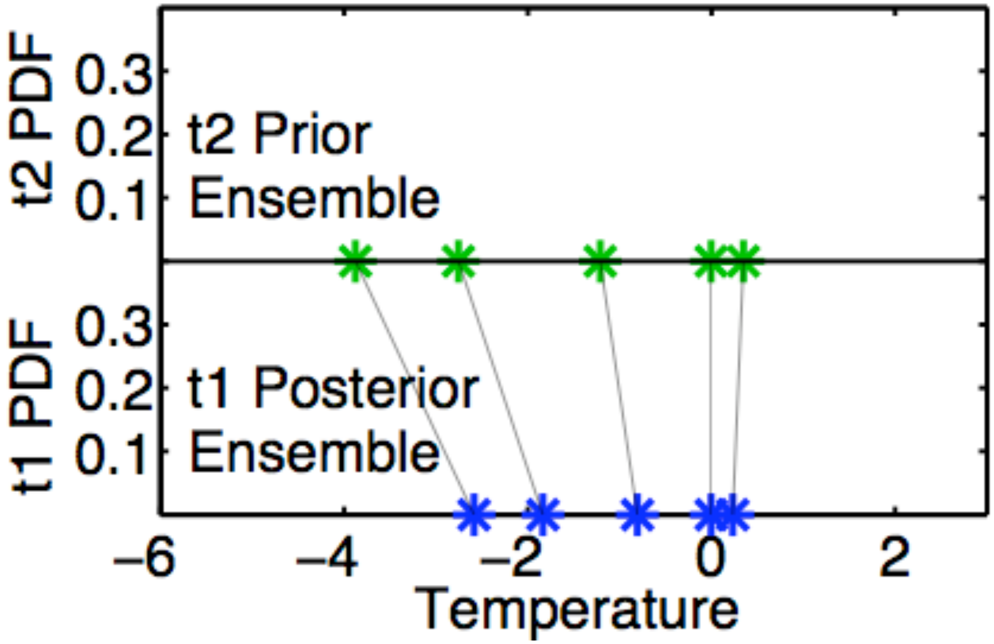
A One-Dimensional Ensemble Kalman Filter: Model Advance

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



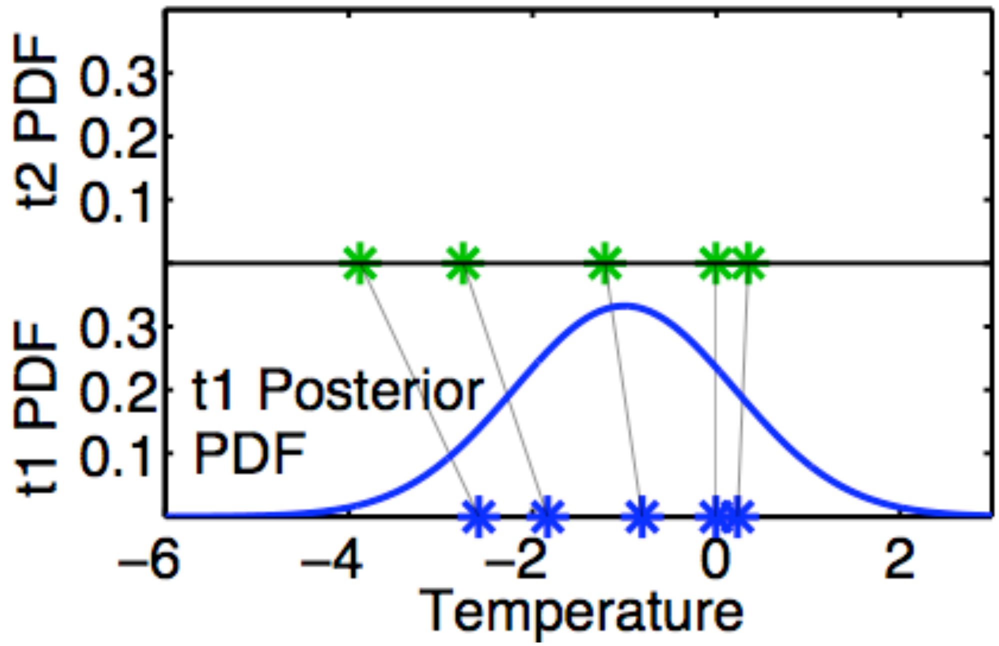
A One-Dimensional Ensemble Kalman Filter: Model Advance

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



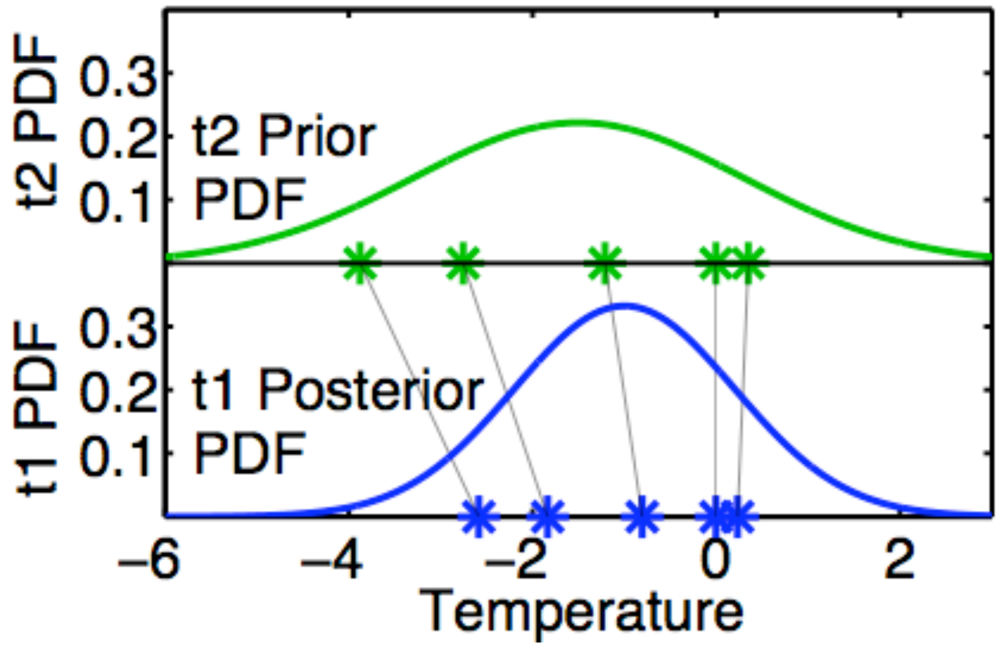
A One-Dimensional Ensemble Kalman Filter: Model Advance

Same as advancing continuous pdf at time t_1 …

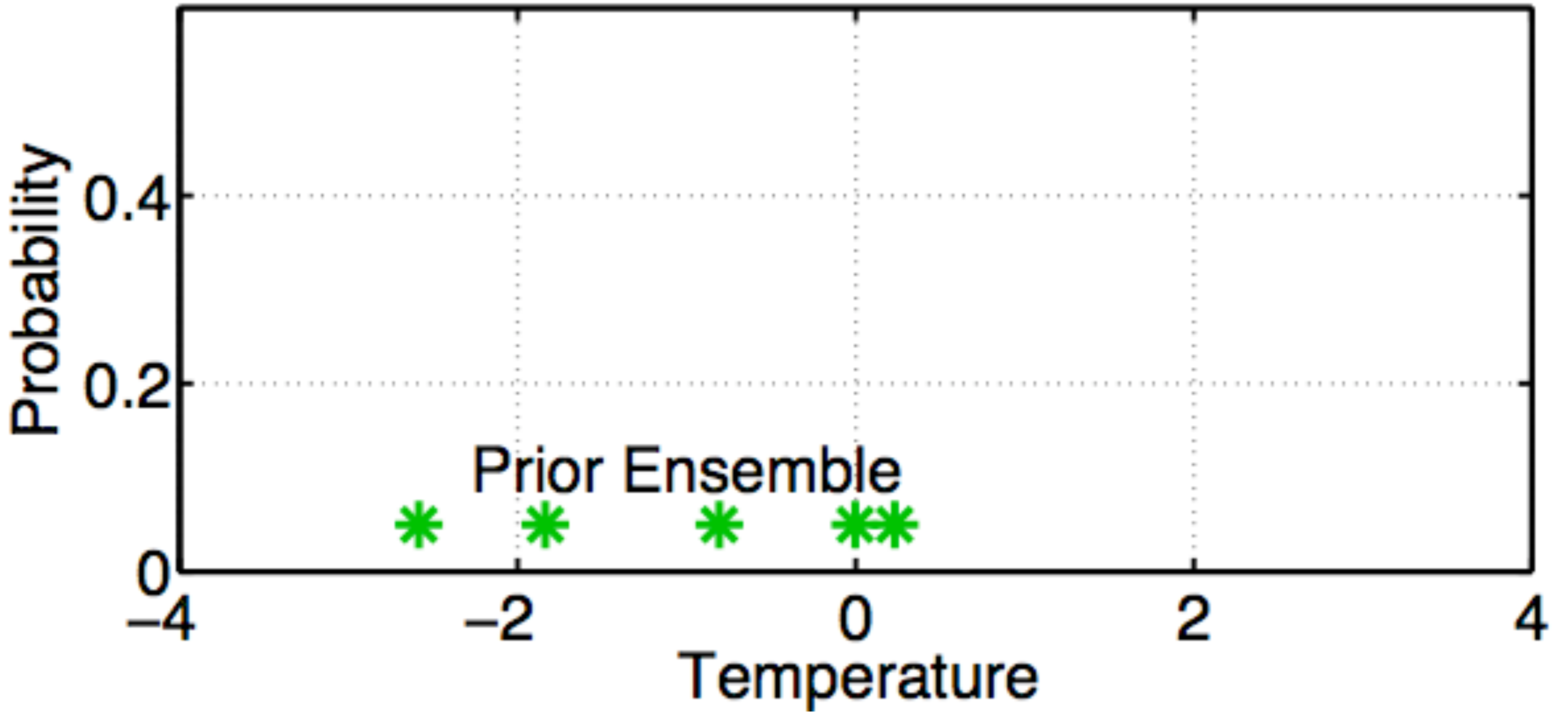


A One-Dimensional Ensemble Kalman Filter: Model Advance

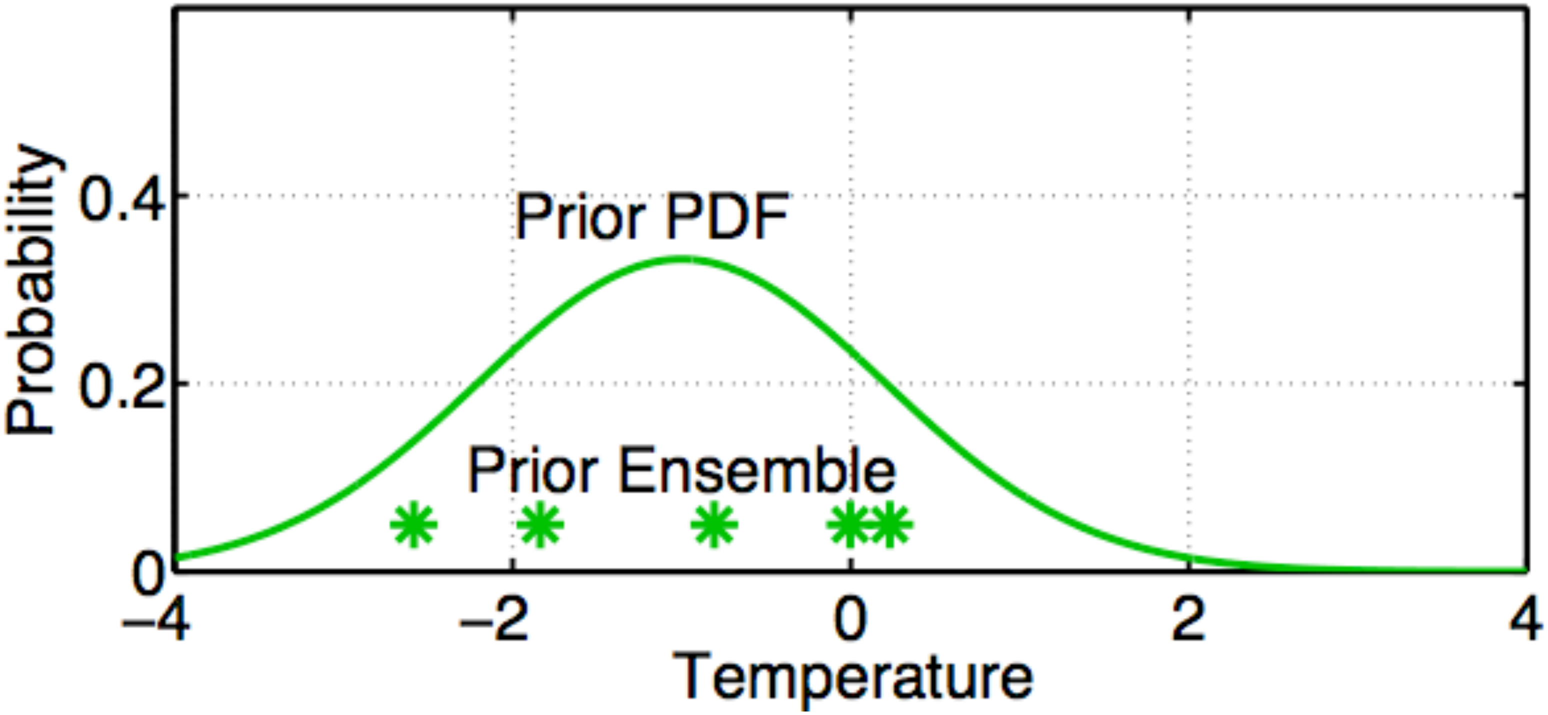
Same as advancing continuous pdf at time t_1 to time t_2 with model L.



A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation

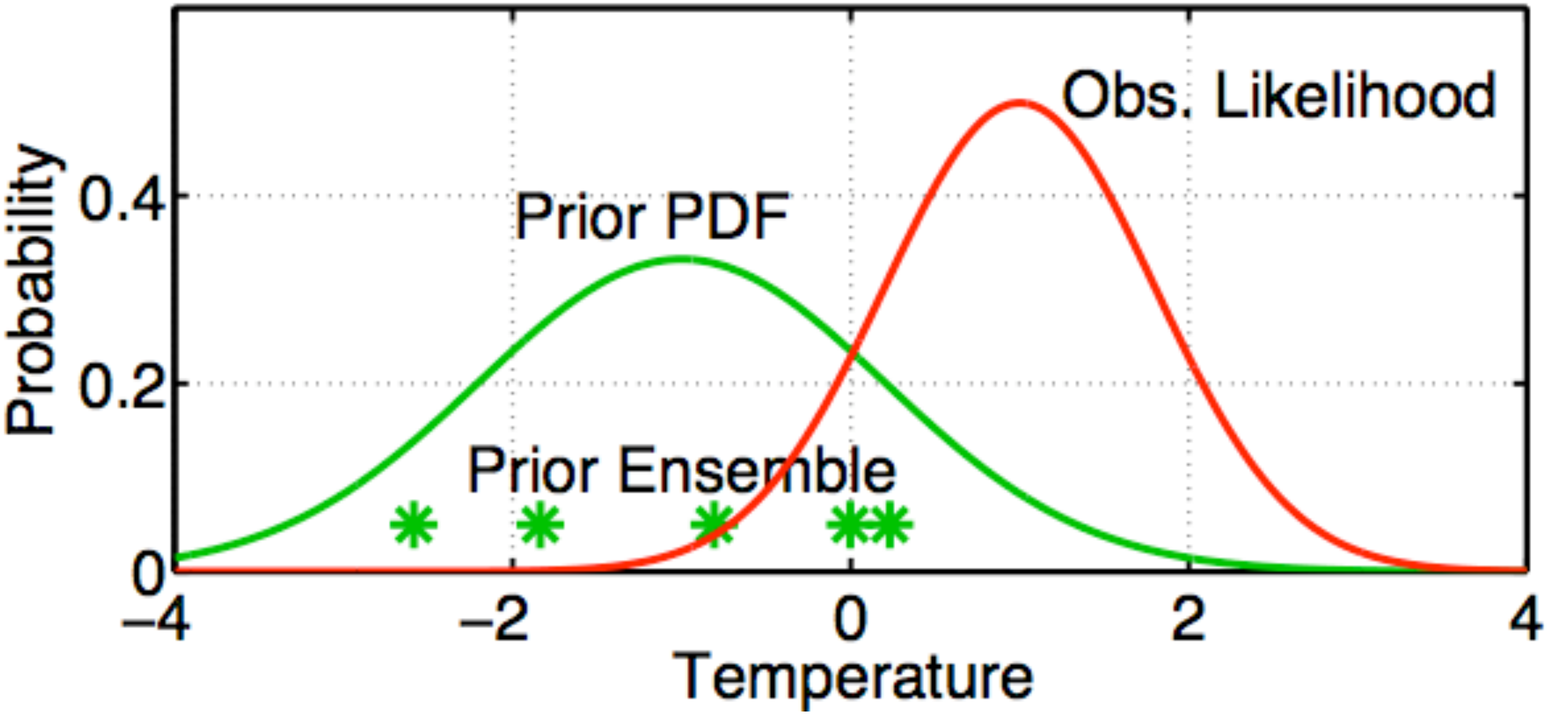


A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



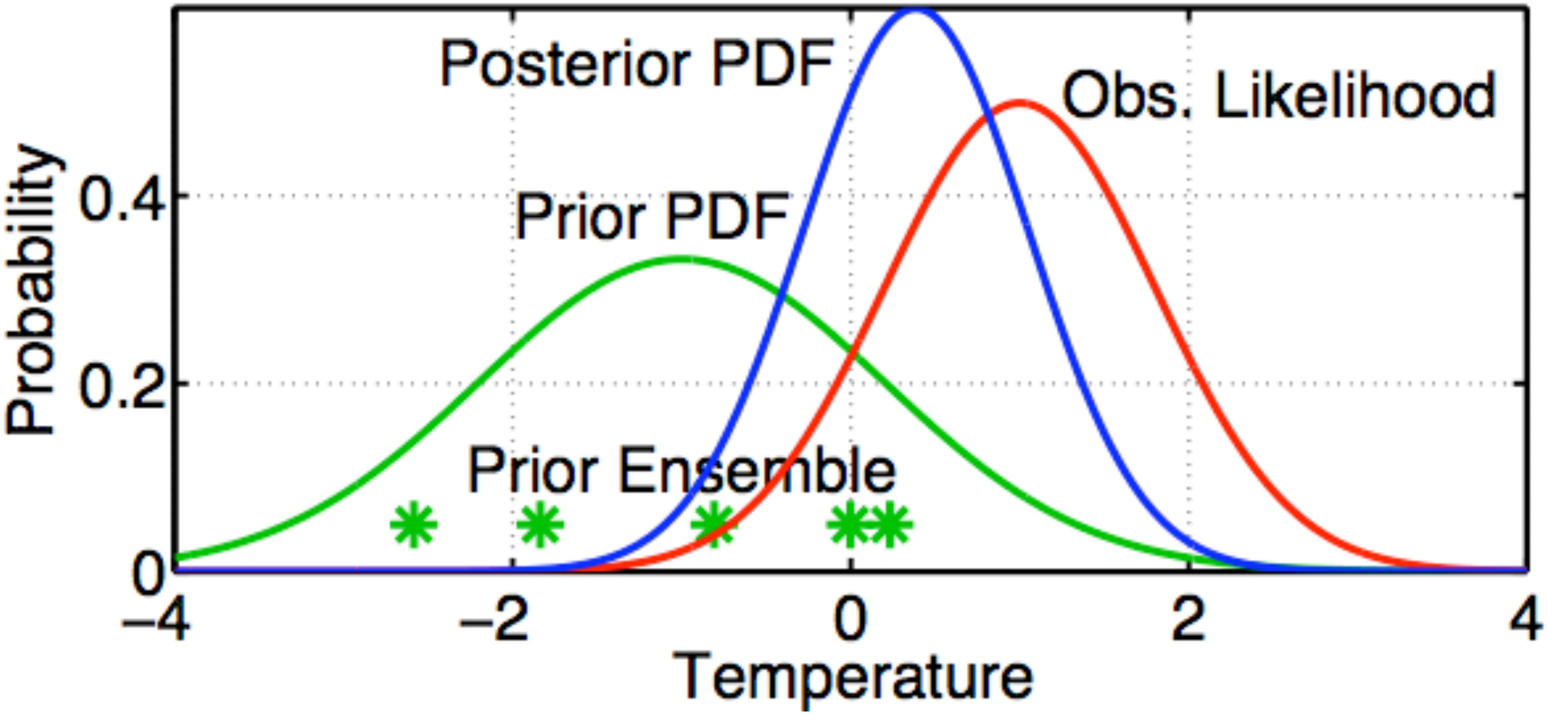
Fit a Gaussian to the sample.

A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



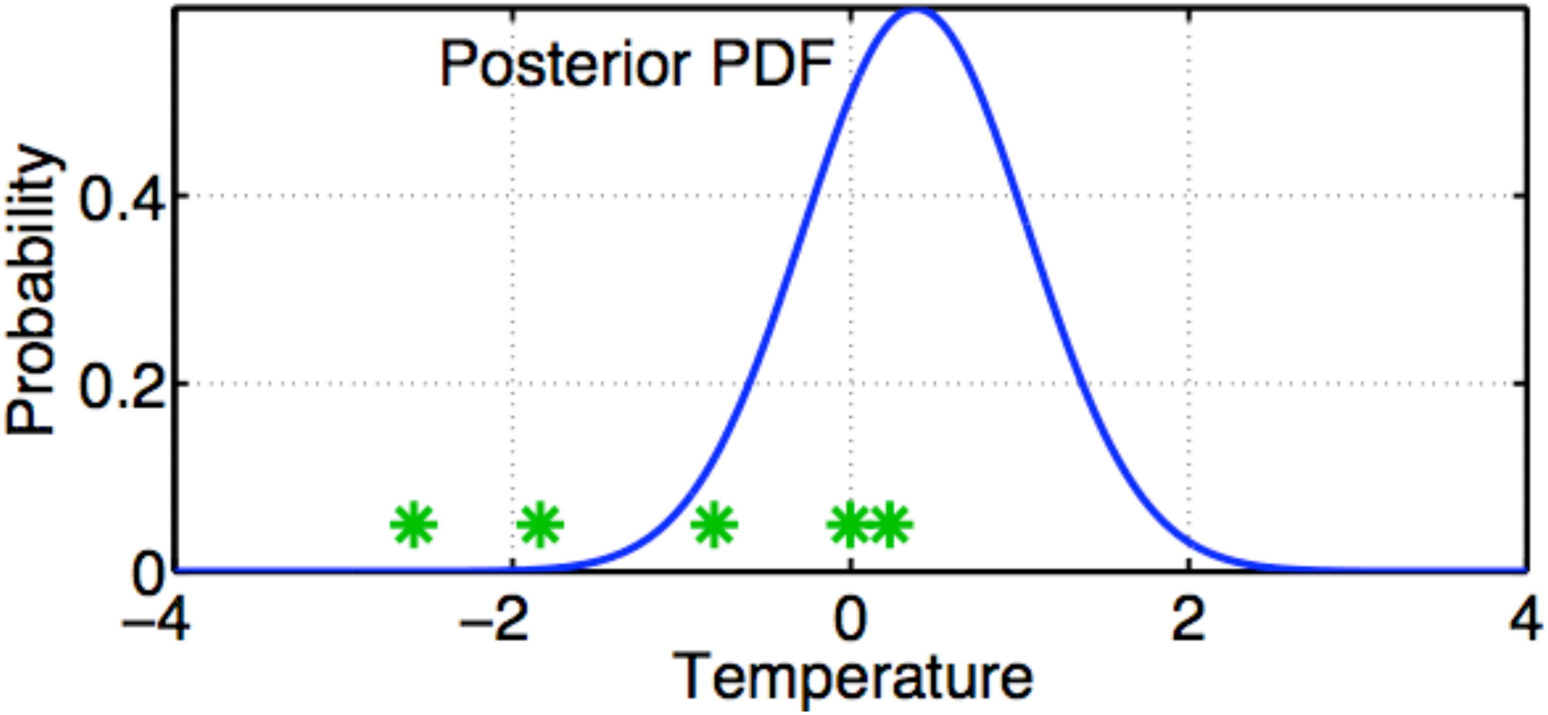
Get the observation likelihood.

A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



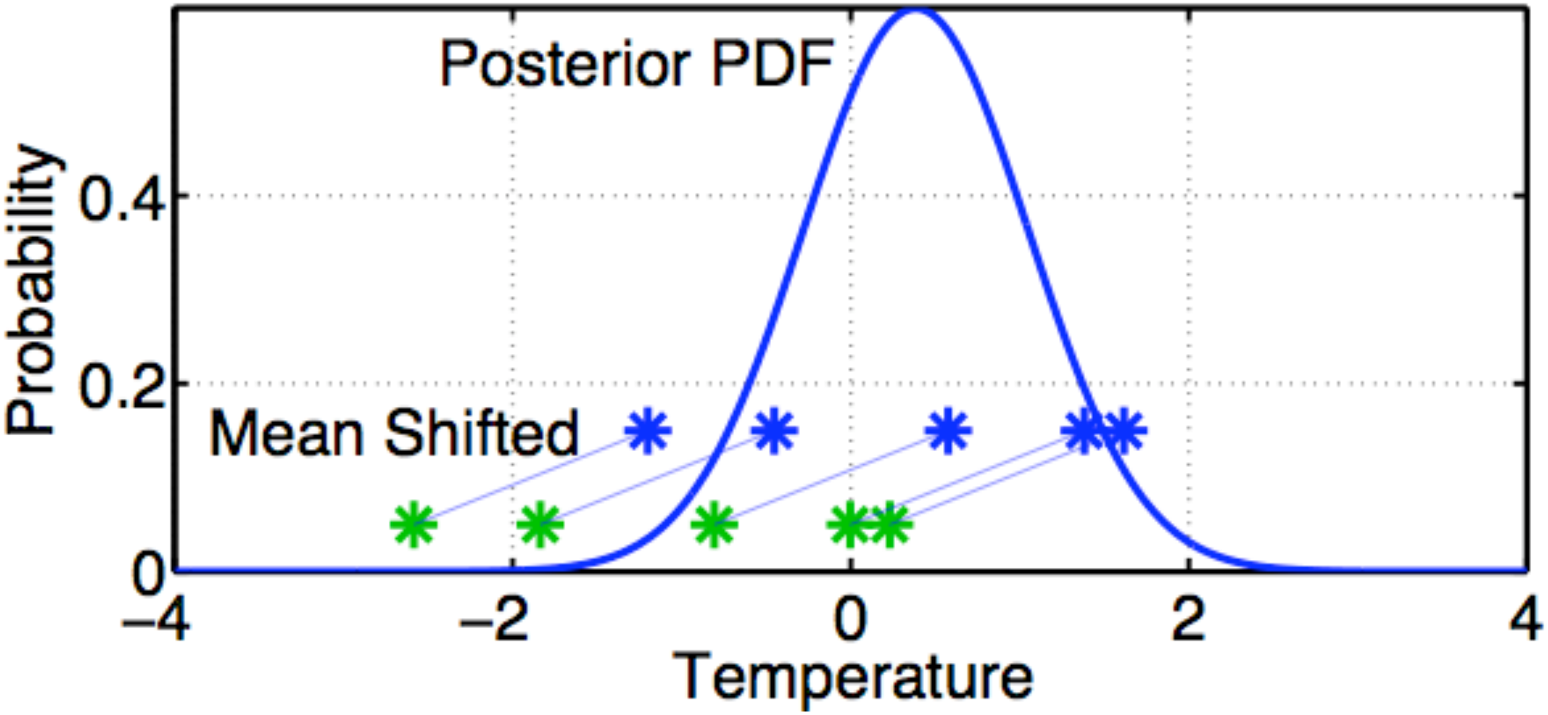
Compute the continuous posterior PDF.

A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



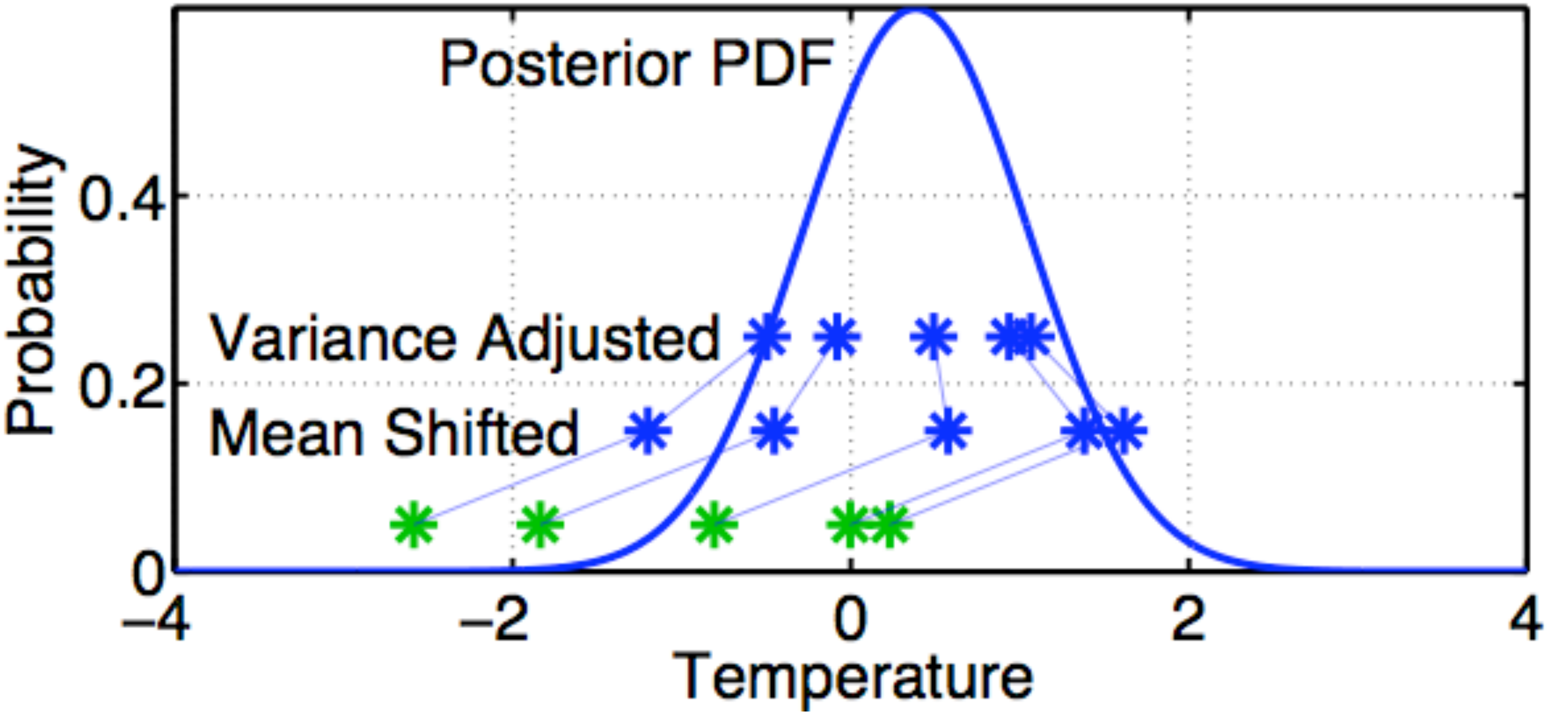
Use a deterministic algorithm to ‘adjust’ the ensemble.

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



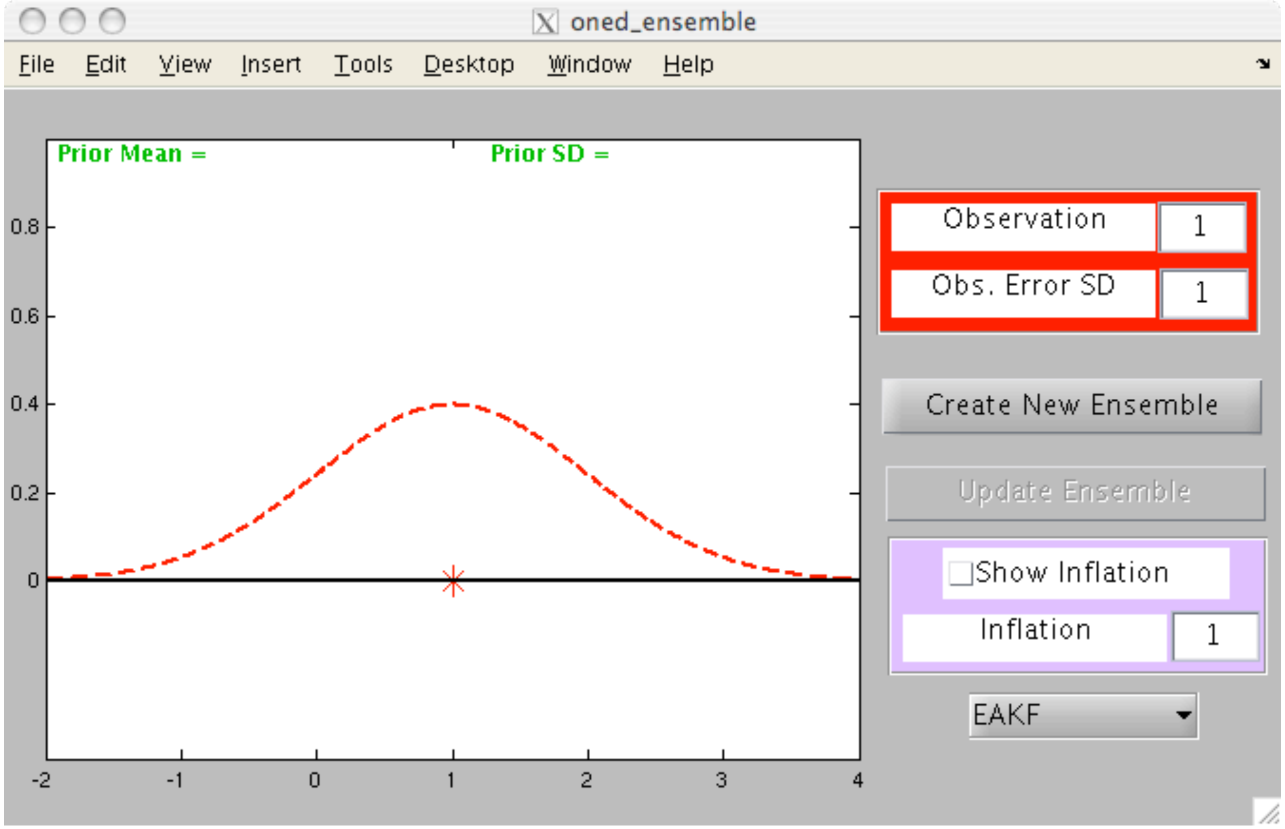
First, ‘shift’ the ensemble to have the exact mean of the posterior.

Second, linearly contract to have the exact variance of the posterior.

Sample statistics are identical to Kalman filter.

Matlab Hands-On: oned_ensemble

Purpose: Explore how ensemble filters update a prior ensemble.



Observation 1

Obs. Error SD 1

□ Show Inflation

Inflation 1

EAKF

Matlab Hands-On: oned_ensemble

Procedure:

1. The observation likelihood mean and standard deviation can be changed with the red dialog boxes.
2. To create a prior ensemble:
   a. Select Create New Ensemble .
   b. Click on the axis in the figure to create an ensemble member. Repeat a few times.
   c. Click on a gray area of the figure to finish ensemble.
   d. Select Update Ensemble to see the updated ensemble.
3. Ignore the inflation box and EAKF pulldown for now.

Matlab Hands-On: oned_ensemble

Explorations:

Keep your ensembles small, less than 10, for easy viewing.

Create a nearly uniformly spaced ensemble. Examine the update.

What happens with an ensemble that is confined to one side of the likelihood?

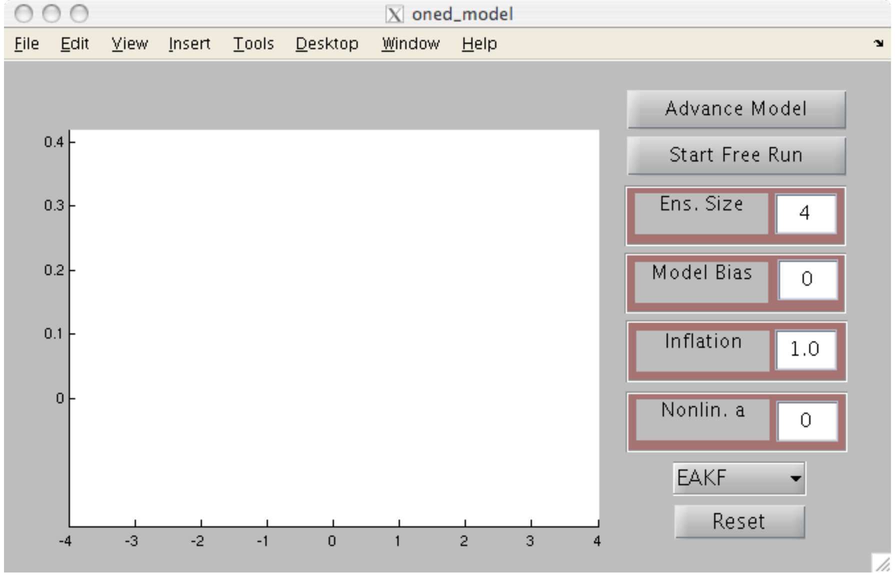
What happens with a bimodal ensemble (two clusters of members on either side)?

What happens with a single outlier in the ensemble?

Matlab Hands-On: oned_model

Purpose: Explore the behavior of a complete 1-dimensional ensemble filter for a linear system.

Look at the behavior of different ensemble sizes.



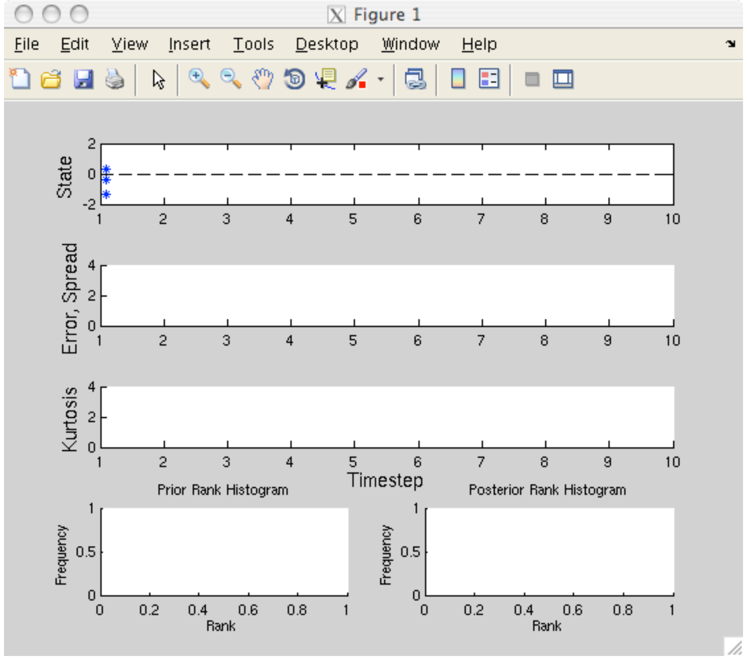
Ens. Size 4

Model Bias 0

Inflation 1.0

Nonlin. a 0

EAKF



Matlab Hands-On: oned_model

Procedure:

This script opens two windows: the menu window and a diagnostic window.

1. To see individual model advance and assimilation steps, select the top button on the menu window (it will alternate between Advance Model and Assimilate Obs).
2. Selecting Start Free Run starts a sequence of advance and assimilation steps.
3. Selecting Stop Free Run stops the sequence of steps.
4. The ensemble size can be changed with a dialog box.
5. Selecting Reset restarts the exercise.

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

The ‘truth’ is always 0.

Observation noise is a draw from a unit normal.

Matlab Hands-On: oned_model

What do I see?

The graphics window on the GUI window displays details of the latest assimilation step. The prior and posterior ensemble, the observation, and the truth are plotted.

The diagnostic window has 5 panels. The top panel shows the evolution of the ensemble with posteriors in blue, model advances in green, and observations in red. The second panel shows the error (absolute value of the difference between the ensemble mean and the truth) in blue and the ensemble spread (standard deviation) in red.

The third panel displays the ensemble kurtosis (more on this later).

The two bottom panels have rank histograms (details later).

Matlab Hands-On: oned_model

Explorations:

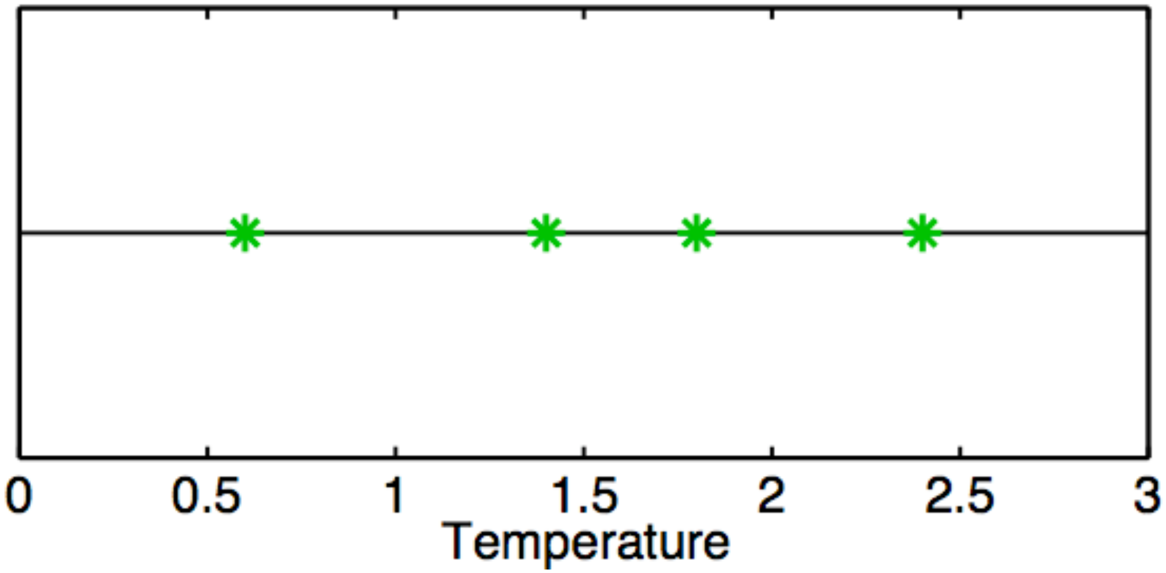
1. Step through a sequence of advances and assimilations with the top button. Watch the evolution of the ensemble, the error and spread.
2. How does a larger ensemble size (< 10 is easiest to see) act?

   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.

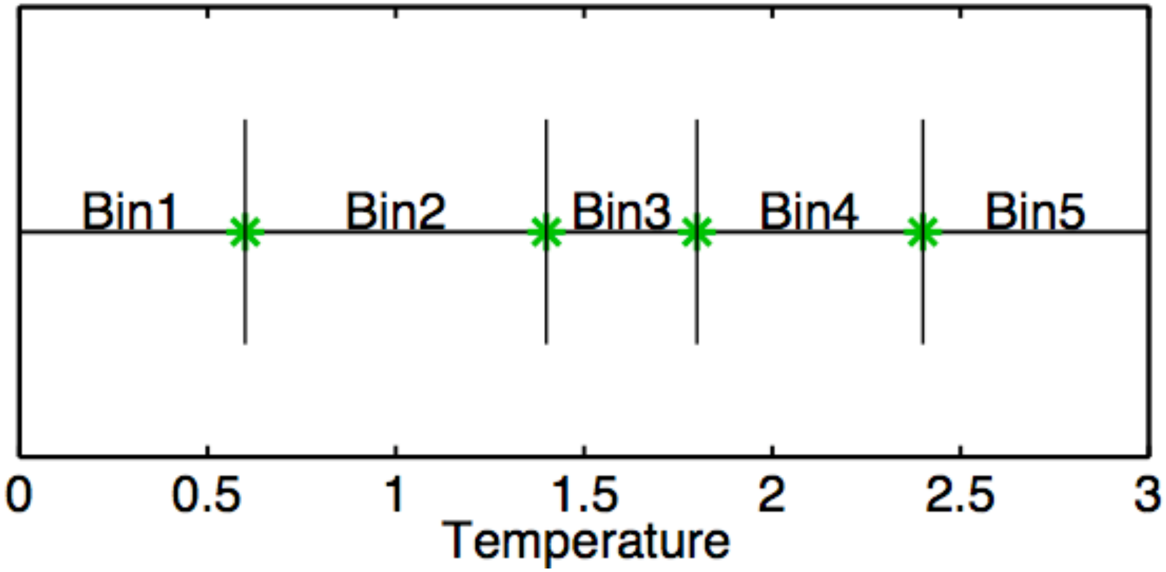
The Rank Histogram: Evaluating Ensemble Performance

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



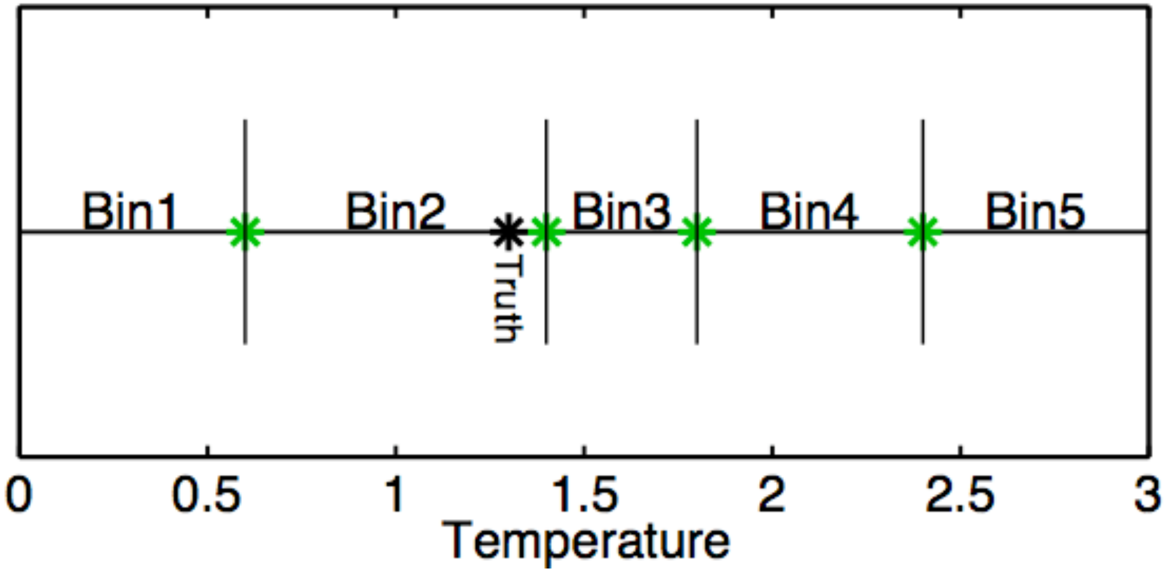
The Rank Histogram: Evaluating Ensemble Performance

These partition the real line into 5 bins.



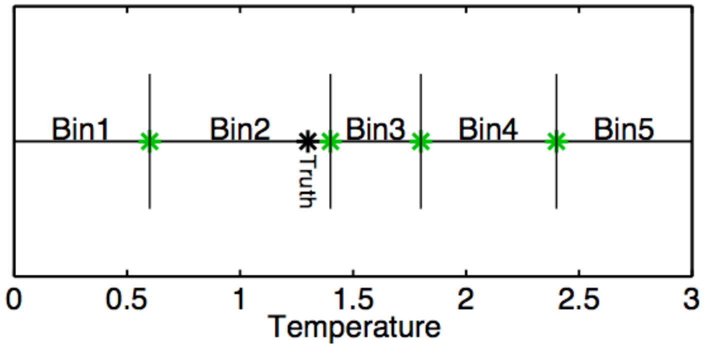
The Rank Histogram: Evaluating Ensemble Performance

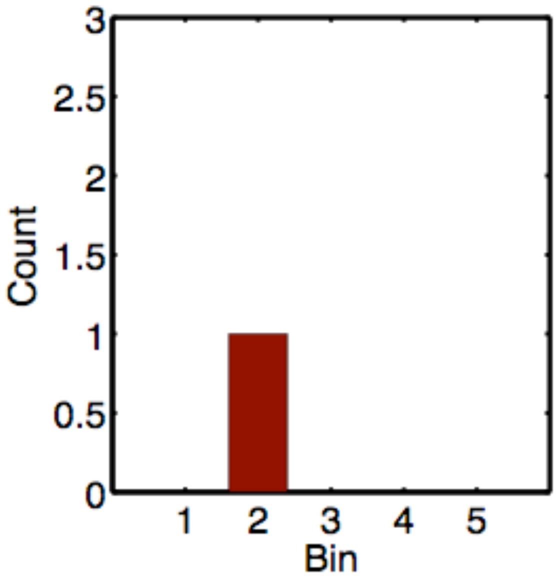
Call the 5th draw the ‘truth’.
1/5 chance that this is in any given bin.



The Rank Histogram: Evaluating Ensemble Performance

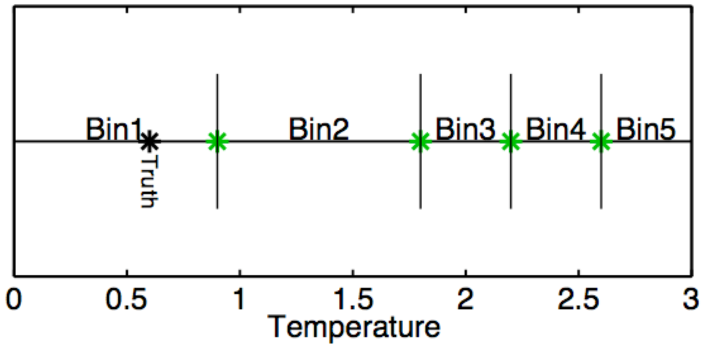
Rank histogram shows the frequency of the truth in each bin over many assimilations.

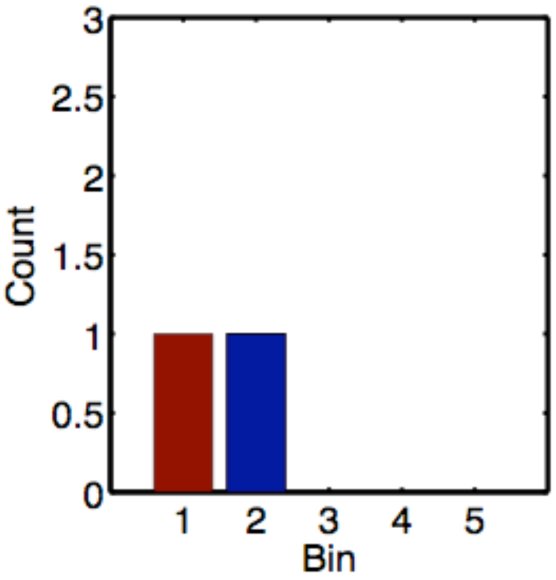




The Rank Histogram: Evaluating Ensemble Performance

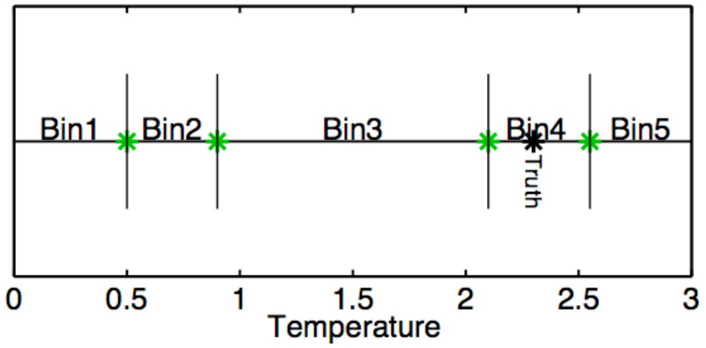
Rank histogram shows the frequency of the truth in each bin over many assimilations.

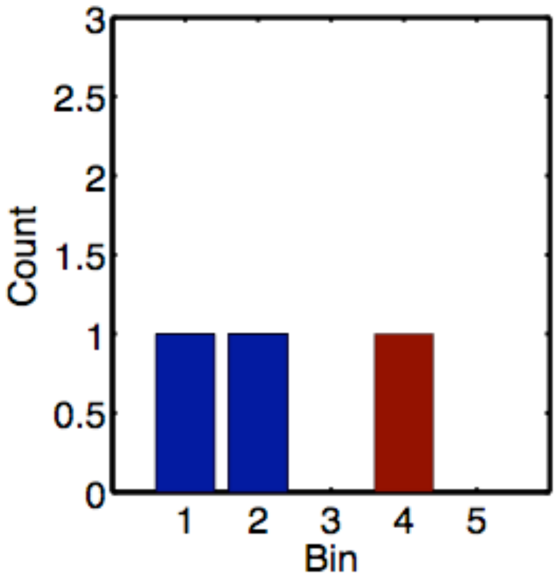




The Rank Histogram: Evaluating Ensemble Performance

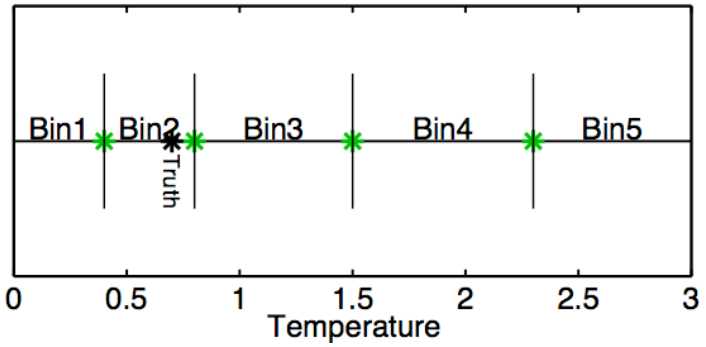
Rank histogram shows the frequency of the truth in each bin over many assimilations.

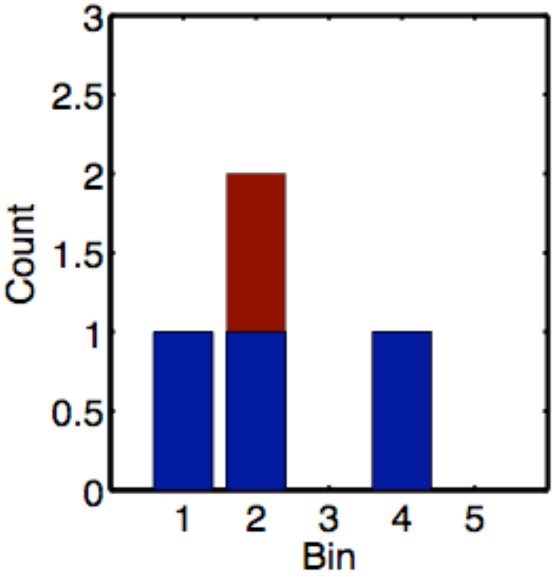




The Rank Histogram: Evaluating Ensemble Performance

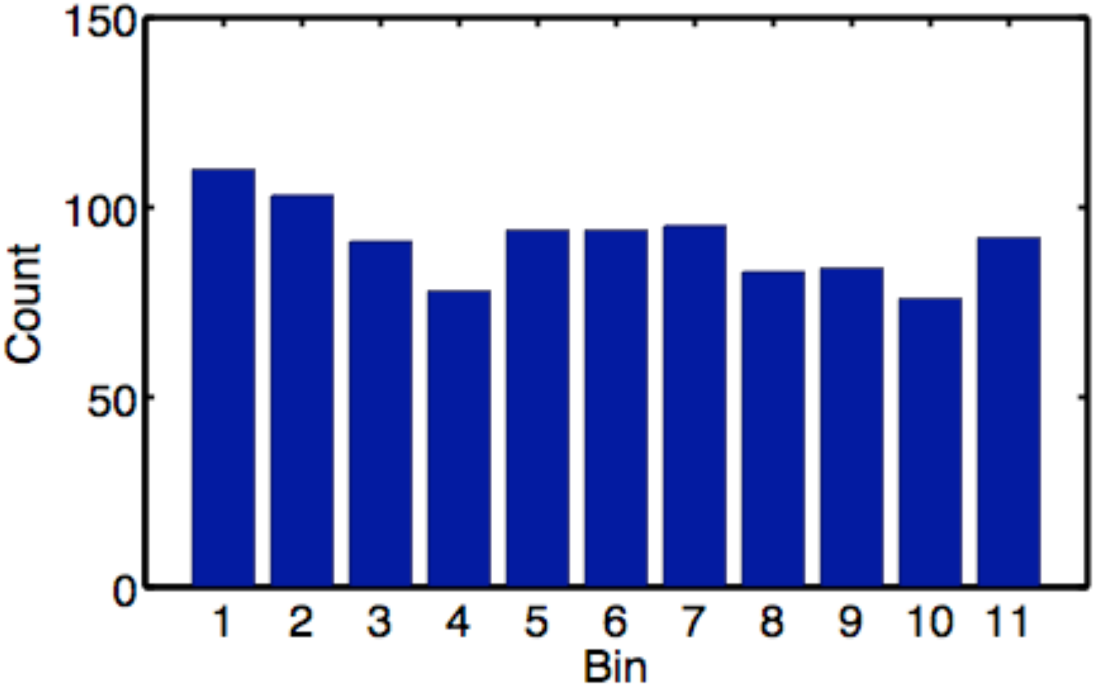
Rank histogram shows the frequency of the truth in each bin over many assimilations.





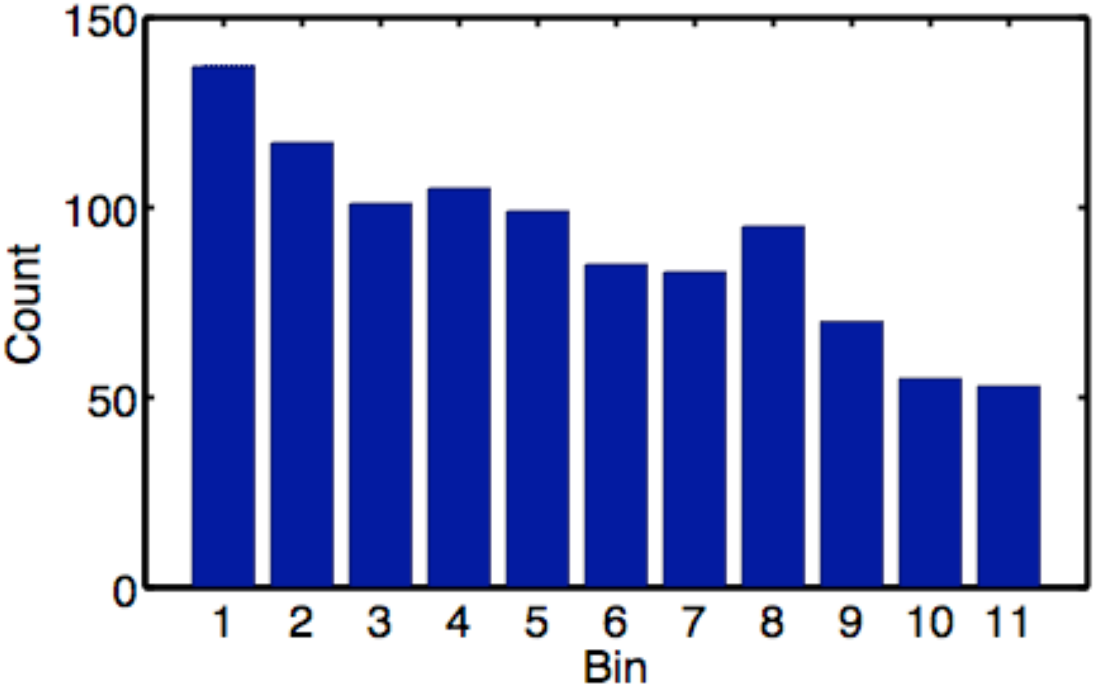
The Rank Histogram: Evaluating Ensemble Performance

Rank histograms for good ensembles should be uniform (caveat sampling noise).
Want truth to look like random draw from ensemble.



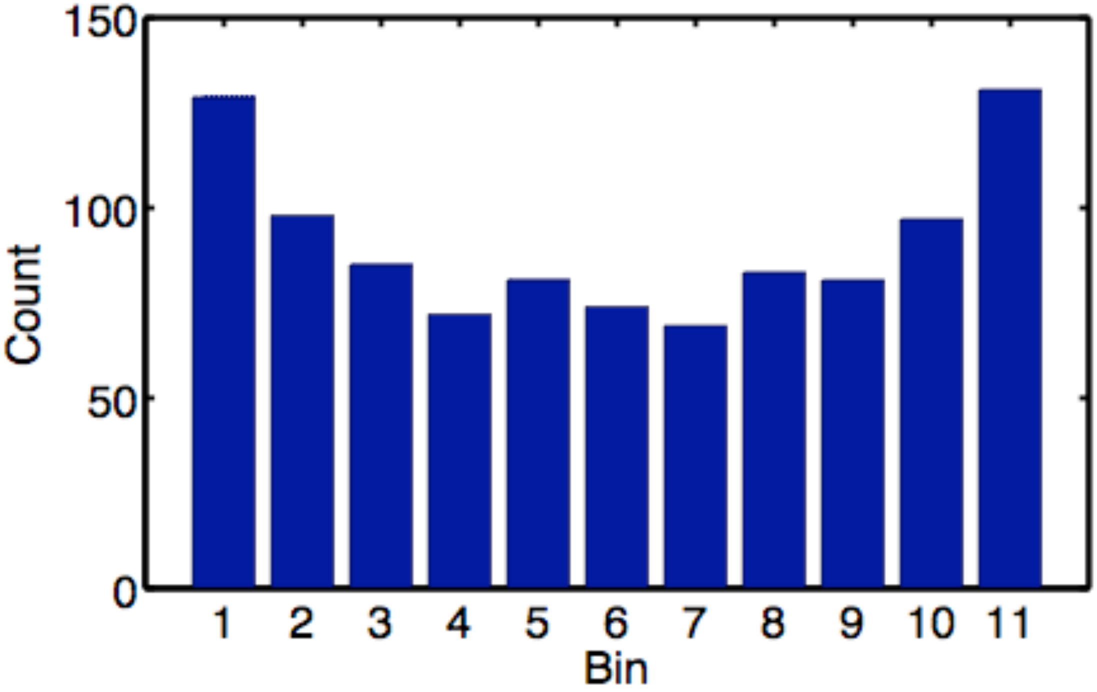
The Rank Histogram: Evaluating Ensemble Performance

A biased ensemble leads to skewed histograms.



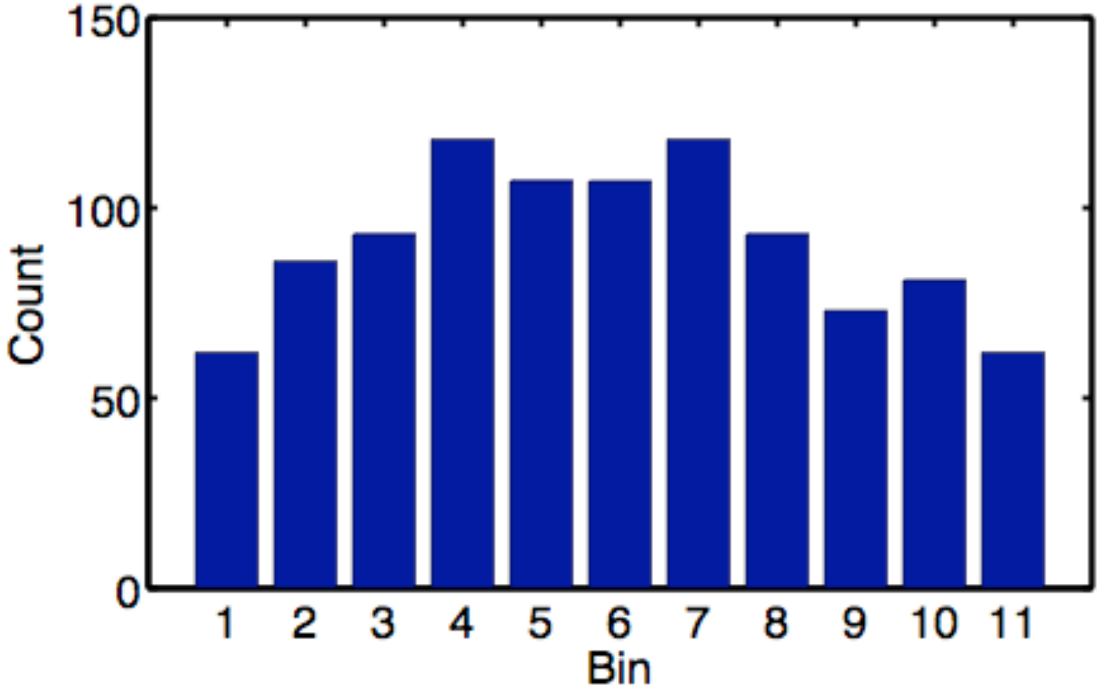
The Rank Histogram: Evaluating Ensemble Performance

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



The Rank Histogram: Evaluating Ensemble Performance

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

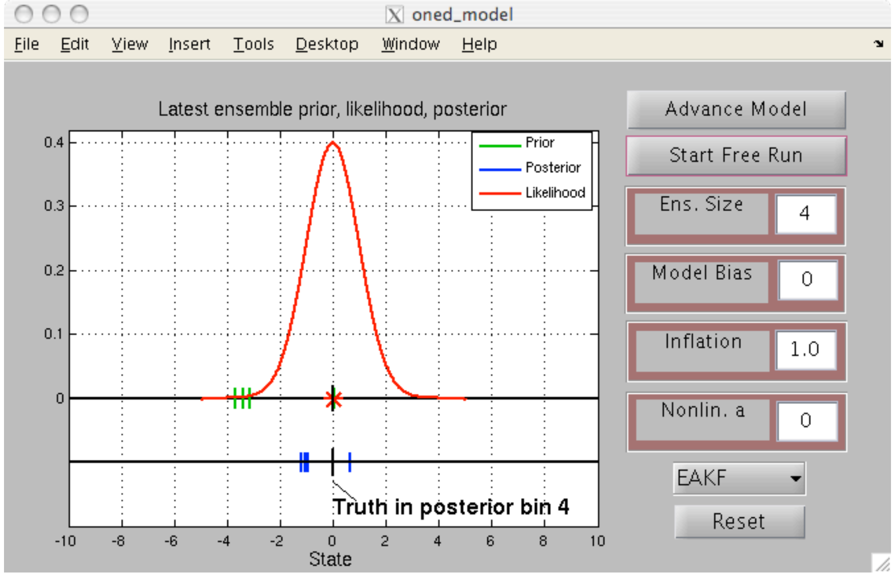


Matlab Hands-On: oned_model (2)

Purpose: Understand rank historgram diagnostics.

Explore the impact of a biased model.

Explore the impact of a nonlinear model.



Ens. Size 4

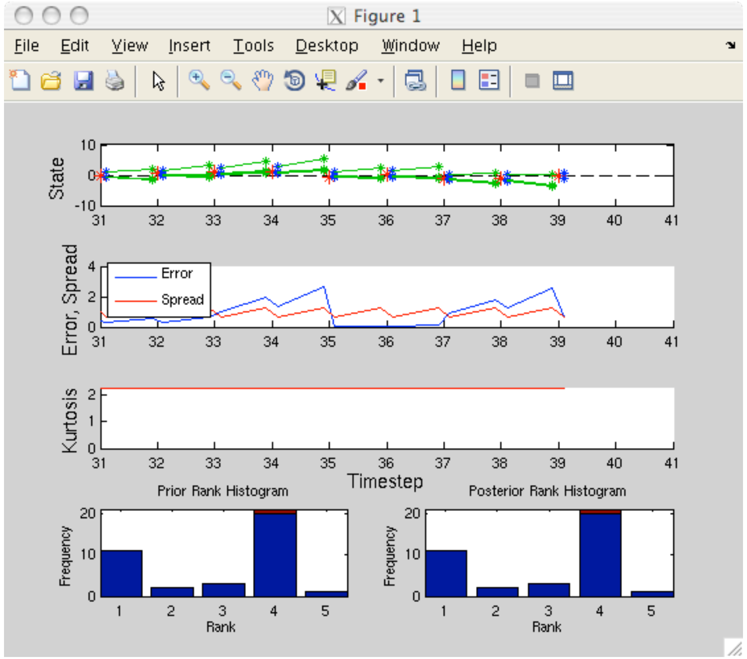
Model Bias 0

Inflation 1.0

Nonlin. a 0

EAKF

Truth in posterior bin 4



Matlab Hands-On: oned_model (2)

Procedure:

This script opens two windows: the menu window and a diagnostic window.

1. To see individual model advance and assimilation steps, select the top button on the menu window (it will alternate between Advance Model and Assimilate Obs).
2. The ensemble size can be changed with a dialog box.
3. A model bias can be set with a dialog box.
4. An additional non-linear term can be added to the model with a dialog box.

Notes: Model bias is dx/dt = x + bias.

Non-linear model is dx/dt = x + a x |x|. Super-exponential growth.

Truth is still always 0.

See matlab script advance_oned.m for details.

Matlab Hands-On: oned_model (2)

What do I see?

The graphics window on the GUI window displays details of the latest assimilation step. Note the rank histogram bin for the truth is labeled.

The diagnostic window has 5 panels. The two bottom panels have rank histograms for the prior and posterior ensembles. The entry for the most recent observation is in red.

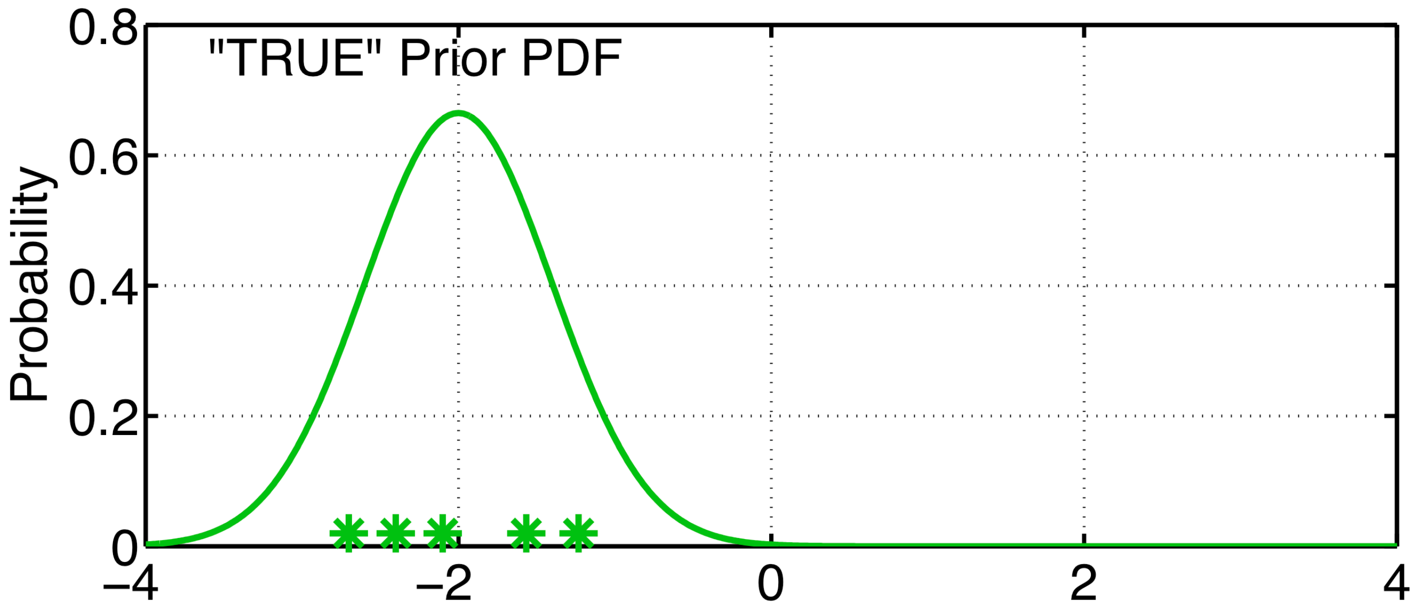
Matlab Hands-On: oned_model (2)

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?

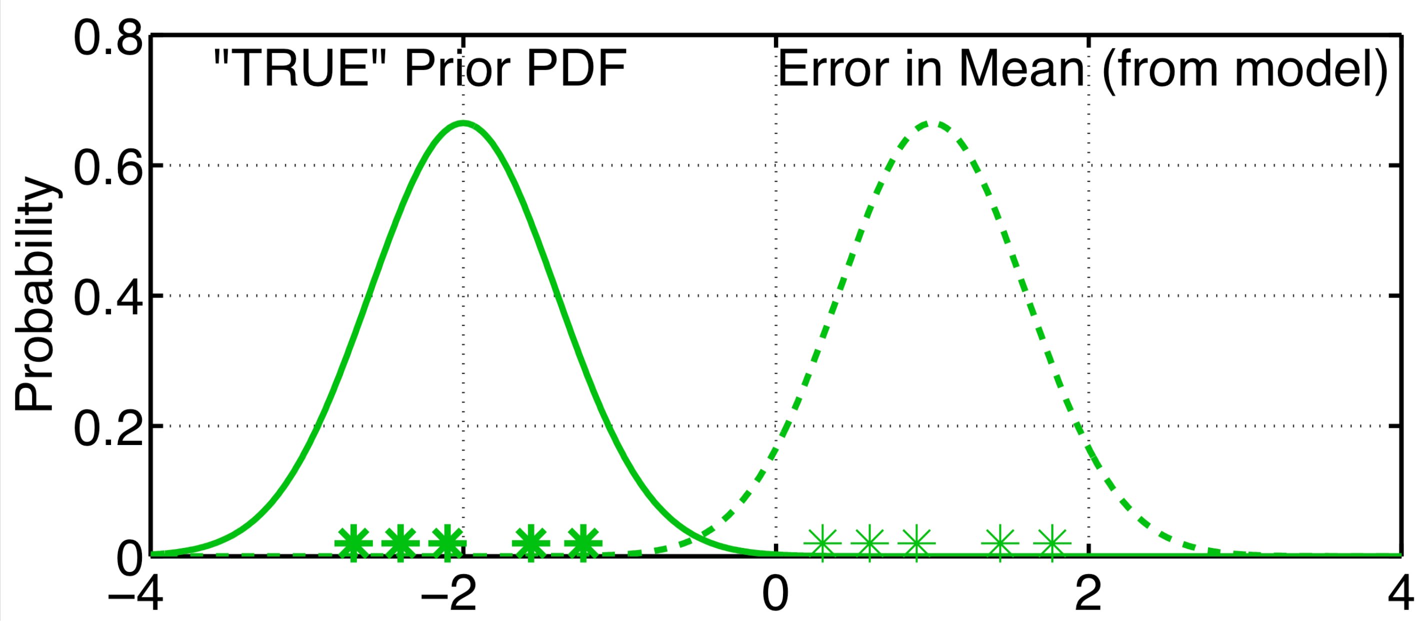
Dealing with systematic error: Variance Inflation

Observations + physical system => ‘true’ distribution.



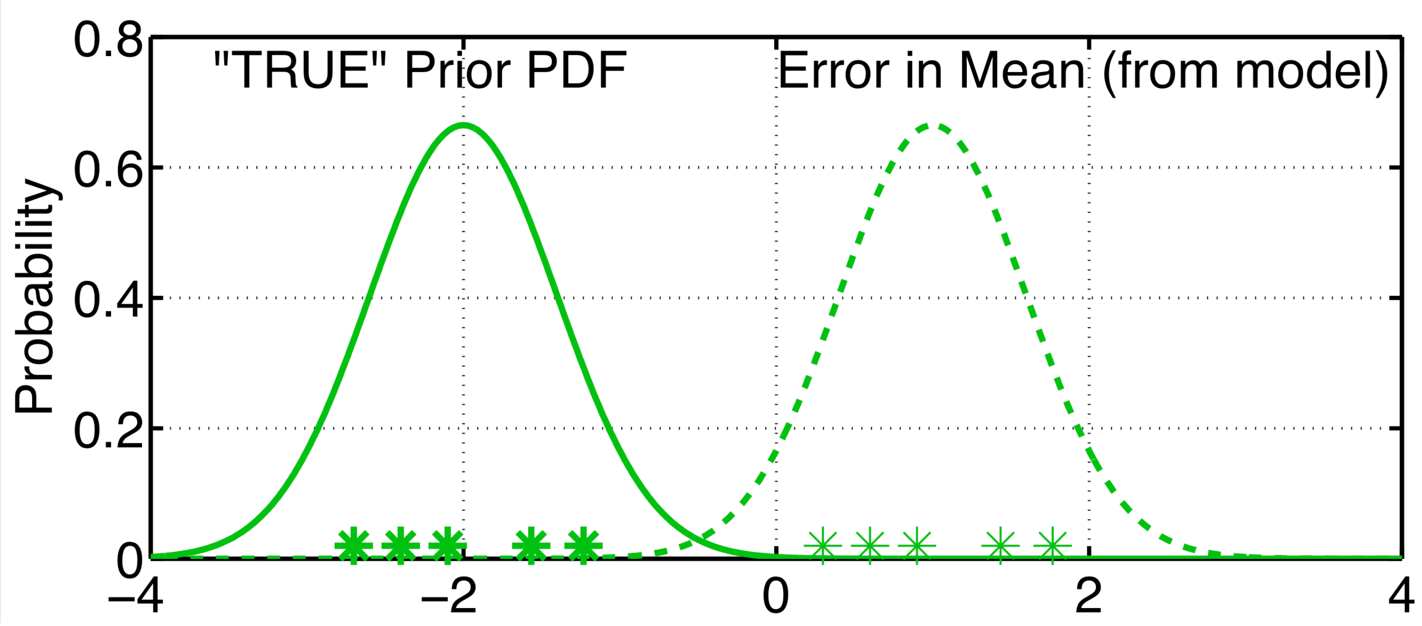
Dealing with systematic error: Variance Inflation

Observations + physical system => ‘true’ distribution.
Model bias (and other errors) can shift actual prior.
Prior ensemble is too certain (needs more spread).



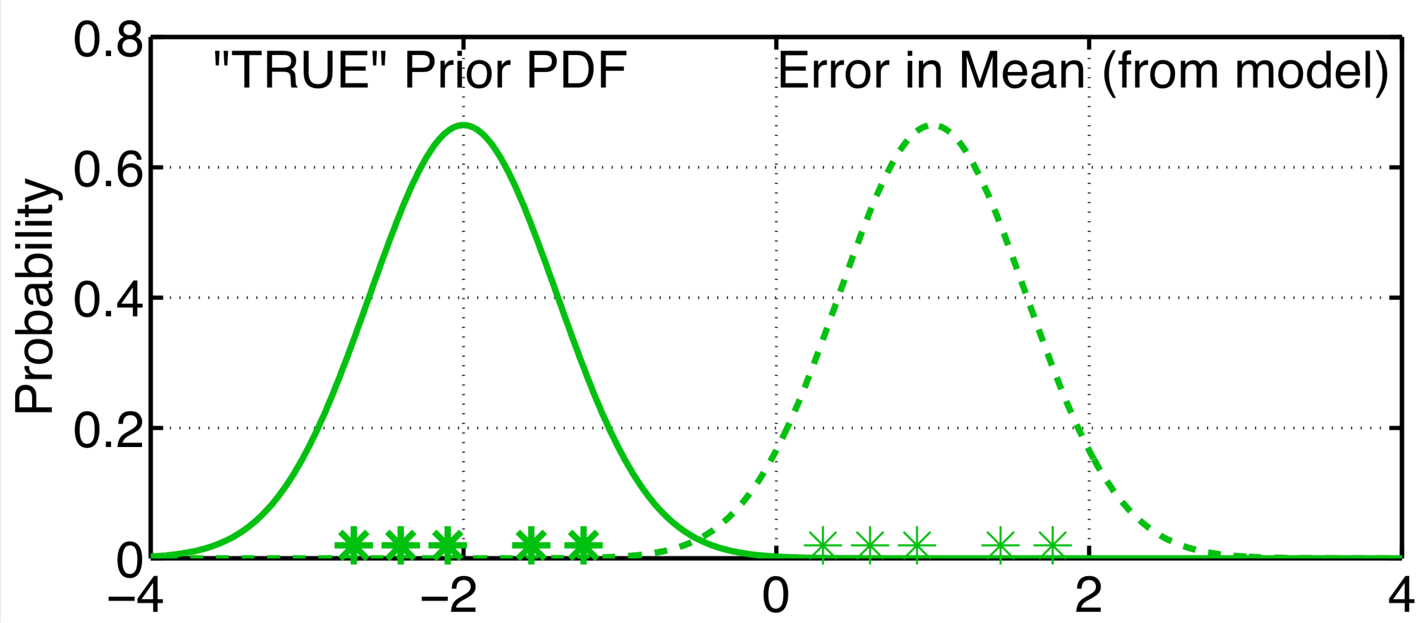
Dealing with systematic error: Variance Inflation

Could correct error if we knew what it was.
With large models, can’t know error precisely.



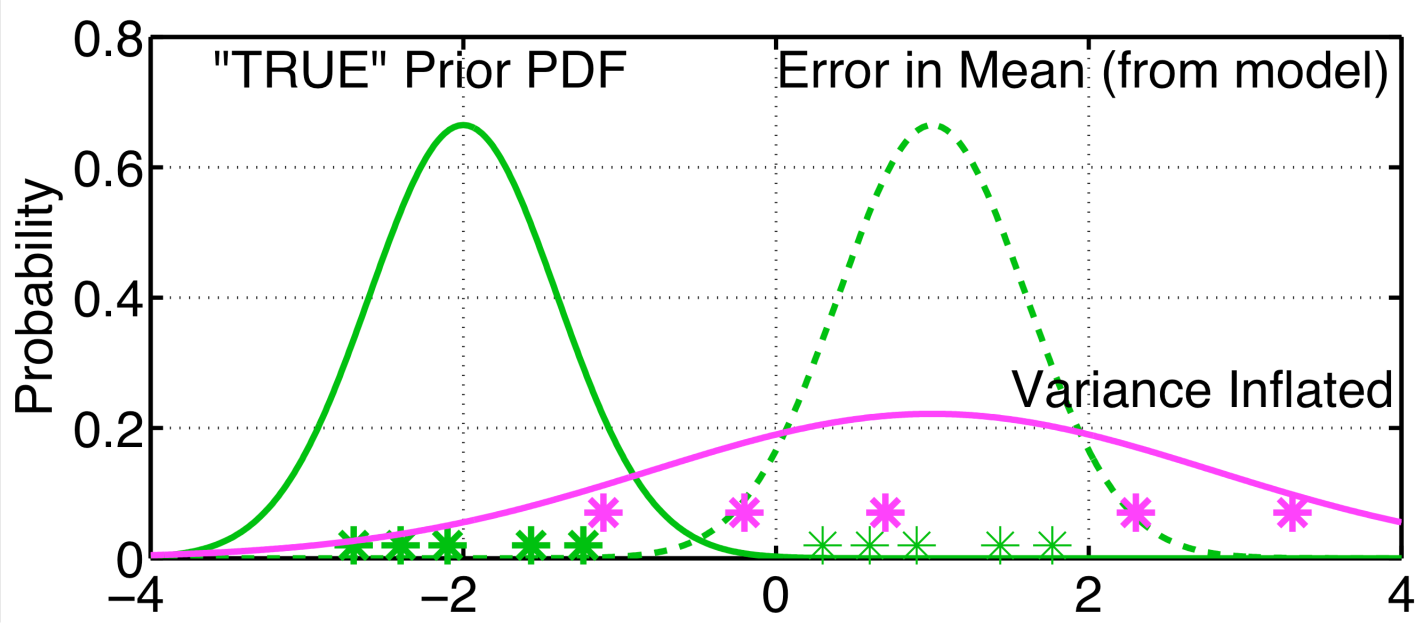
Dealing with systematic error: Variance Inflation

Taking no action can cause observations to be ignored.



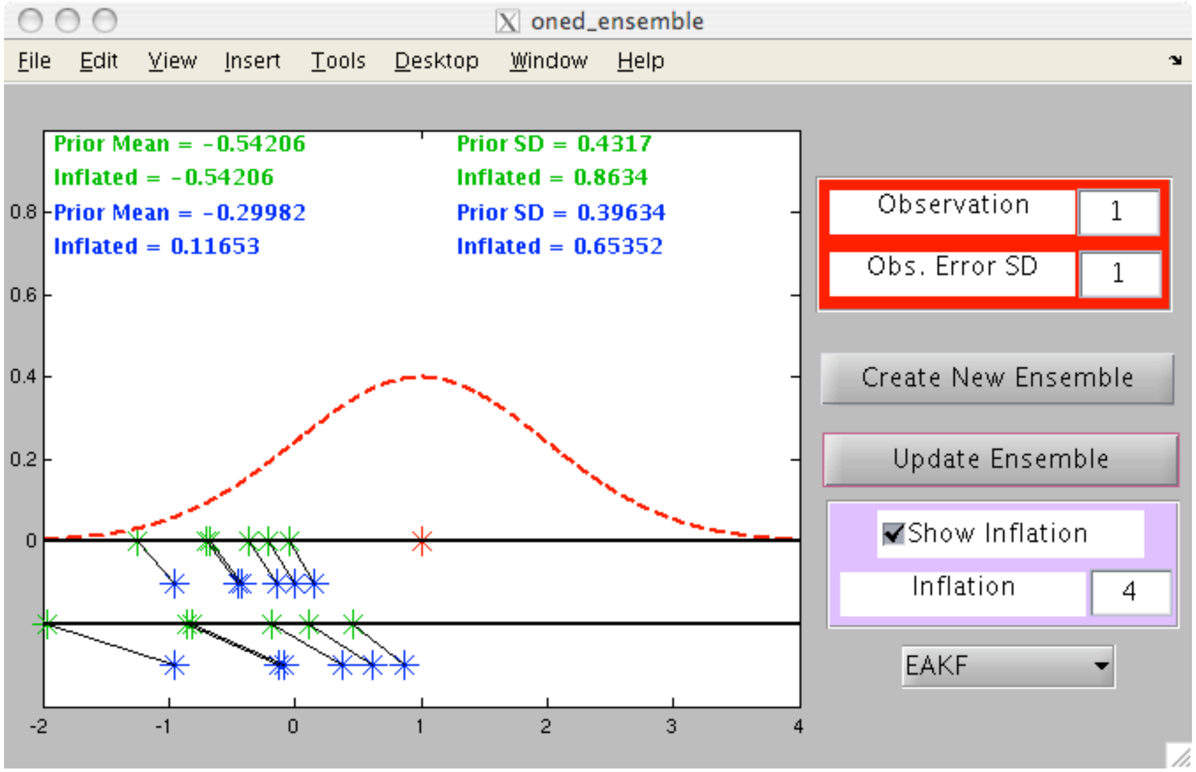
Dealing with systematic error: Variance Inflation

Naïve solution: increase the spread in the prior.
Give more weight to the observation, less to prior.



Matlab Hands-On: oned_ensemble (2)

Purpose: Explore how inflating the prior ensemble impacts the posterior ensemble.



Prior Mean = -0.54206
Inflated = -0.54206
Prior Mean = -0.29982
Inflated = 0.11653

Prior SD = 0.4317
Inflated = 0.8634
Prior SD = 0.39634
Inflated = 0.65352

Observation 1

Obs. Error SD 1

☑ Show Inflation

Inflation 4

EAKF

Matlab Hands-On: oned_ensemble (2)

Procedure:

1. To create a prior ensemble:
   a. Select Create New Ensemble .
   b. Click on the axis in the figure to create an ensemble member. Repeat a few times.
   c. Click on a gray area of the figure to finish ensemble.
   d. Select Update Ensemble to see the updated ensemble.
2. Select Show Inflation before Update Ensemble to see inflated prior and posterior and statistics.

Matlab Hands-On: oned_ensemble (2)

Explorations:

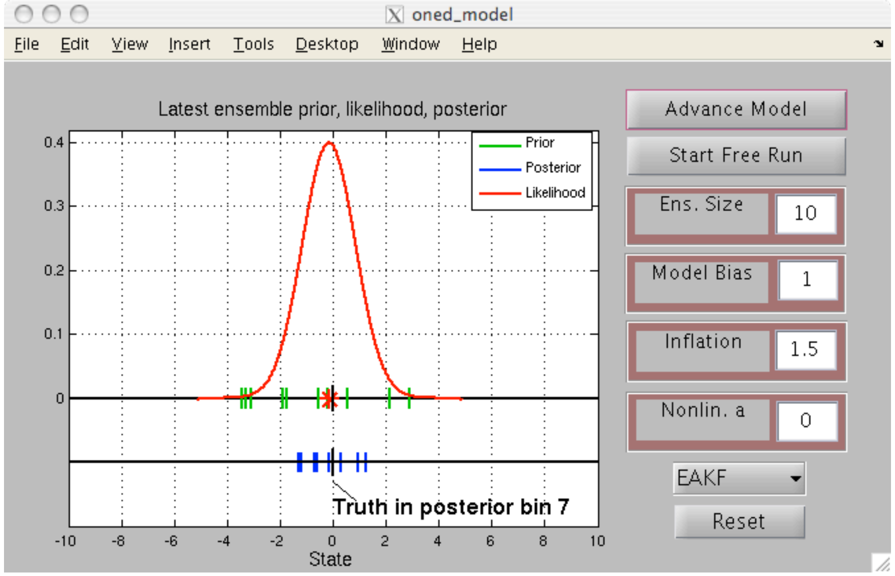
See how increasing inflation (> 1) changes the posterior mean and standard deviation.

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 (3)

Purpose: Explore the use of inflation to deal with model systematic error.



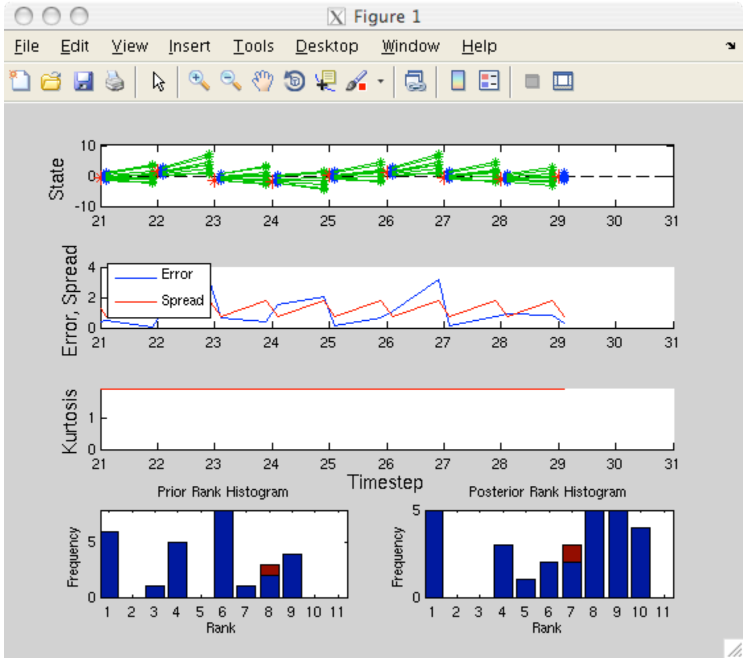
Ens. Size 10

Model Bias 1

Inflation 1.5

Nonlin. a 0

EAKF



Matlab Hands-On: oned_model (3)

Procedure:

This script opens two windows: the menu window and a diagnostic window.

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

Remember: This uses the simple linear model dx/dt = x.
The 'truth' is always 0.
Observation noise is a draw from a unit normal.
The spread is increased by the square root of the inflation.

Matlab Hands-On: oned_model (3)

What do I see?

The graphics window on the GUI window displays details of the latest assimilation step. The prior and posterior ensemble, the observation, and the truth are plotted.

The diagnostic window has 5 panels. The top panel shows the evolution of the ensemble with posteriors in blue, model advances in green, and observations in red. The second panel shows the error (absolute value of the difference between the ensemble mean and the truth) in blue and the ensemble spread (standard deviation) in red.

The third panel displays the ensemble kurtosis (more on this later).

The two bottom panels have rank histograms.

Matlab Hands-On: oned_model (3)

Explorations:

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