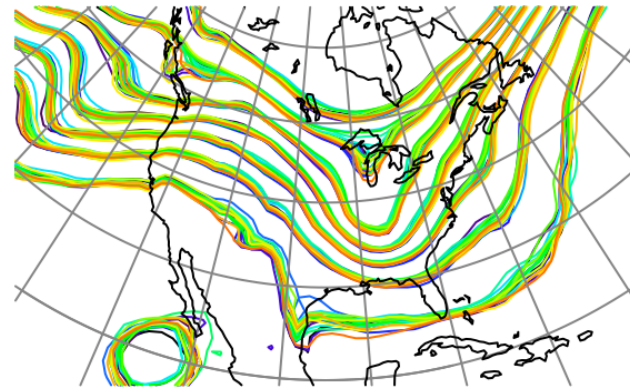


D  
A  
R  
T

ata  
ssimilation  
esearch  
estbed



# DART\_LAB Tutorial Section 1: Ensemble Data Assimilation Concepts in 1D



©UCAR 2014

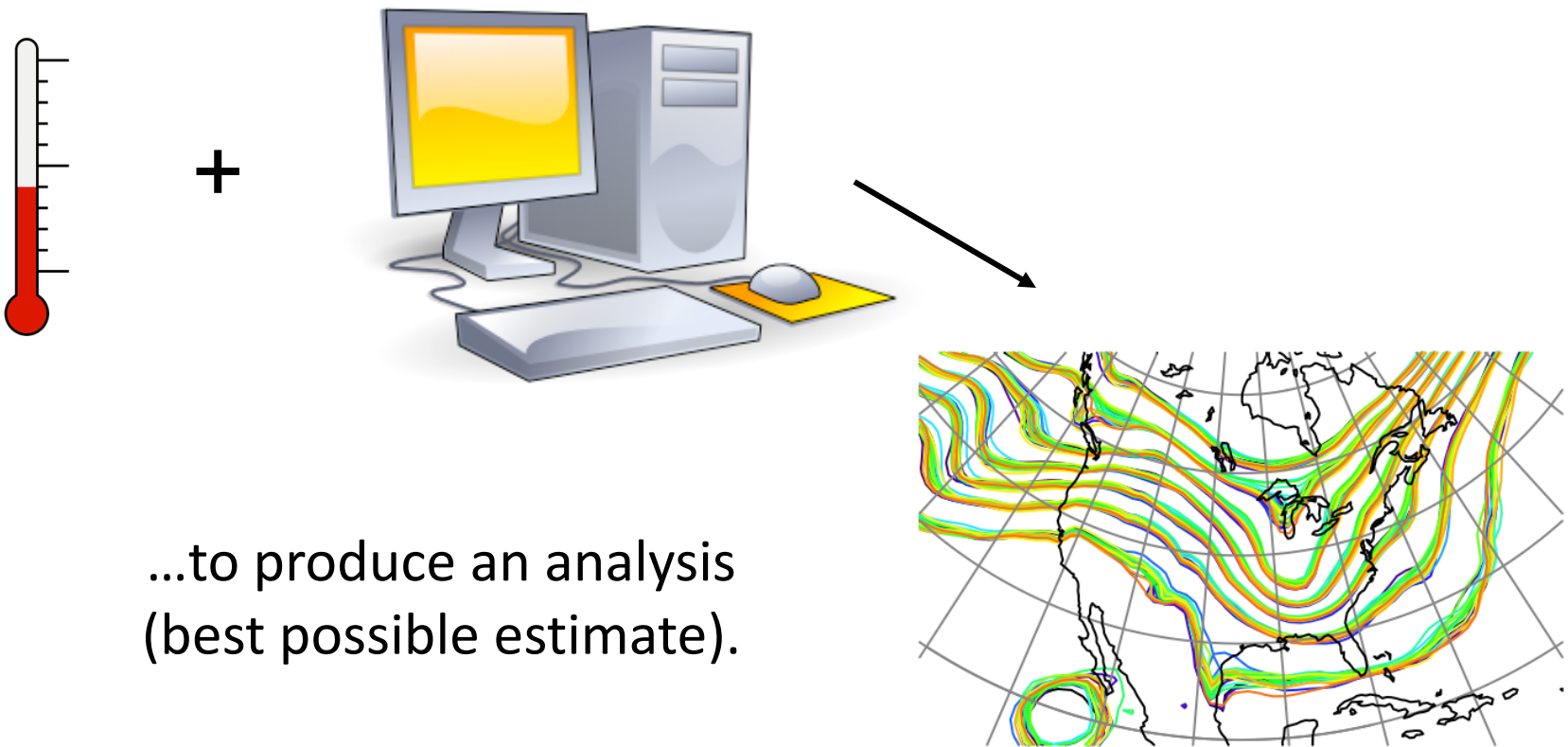


The National Center for Atmospheric Research is sponsored by the National Science Foundation. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

NCAR | National Center for  
UCAR | Atmospheric Research

# What is Data Assimilation?

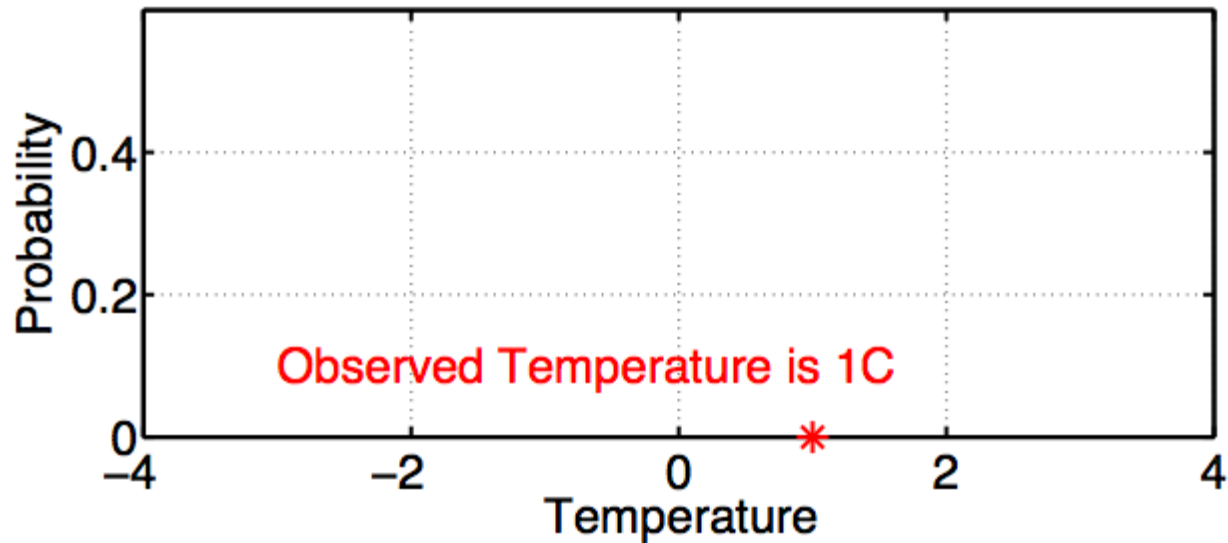
Observations combined with a Model forecast ...



...to produce an analysis  
(best possible estimate).

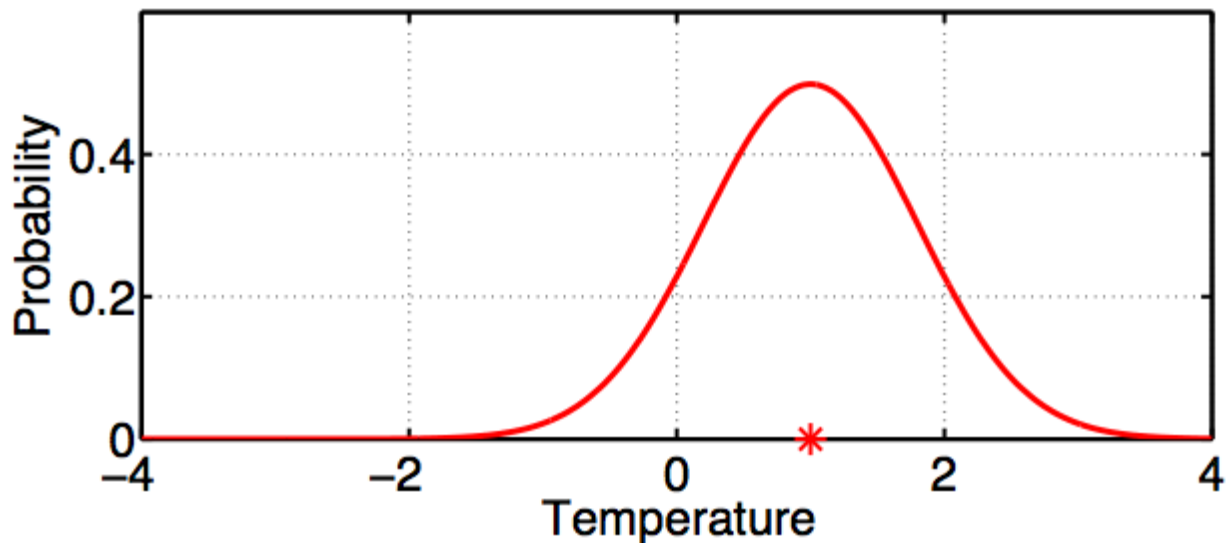
# Example: Estimating the Temperature Outside

An observation has a value ( \* ),



# Example: Estimating the Temperature Outside

An observation has a value ( \* ),

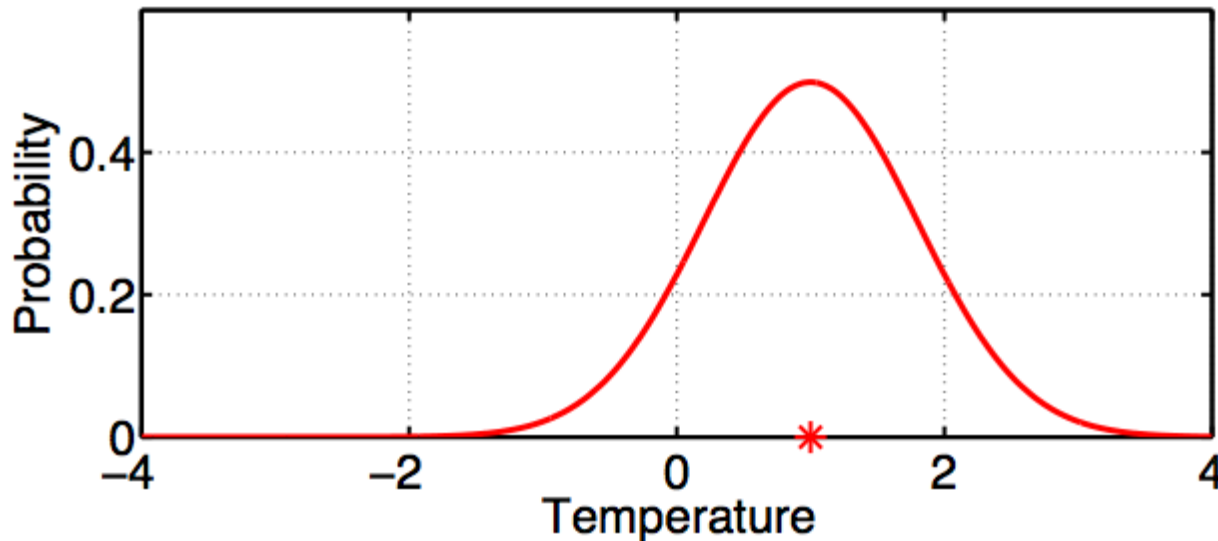


and an error distribution (red curve) that is associated with the instrument.



# Example: Estimating the Temperature Outside

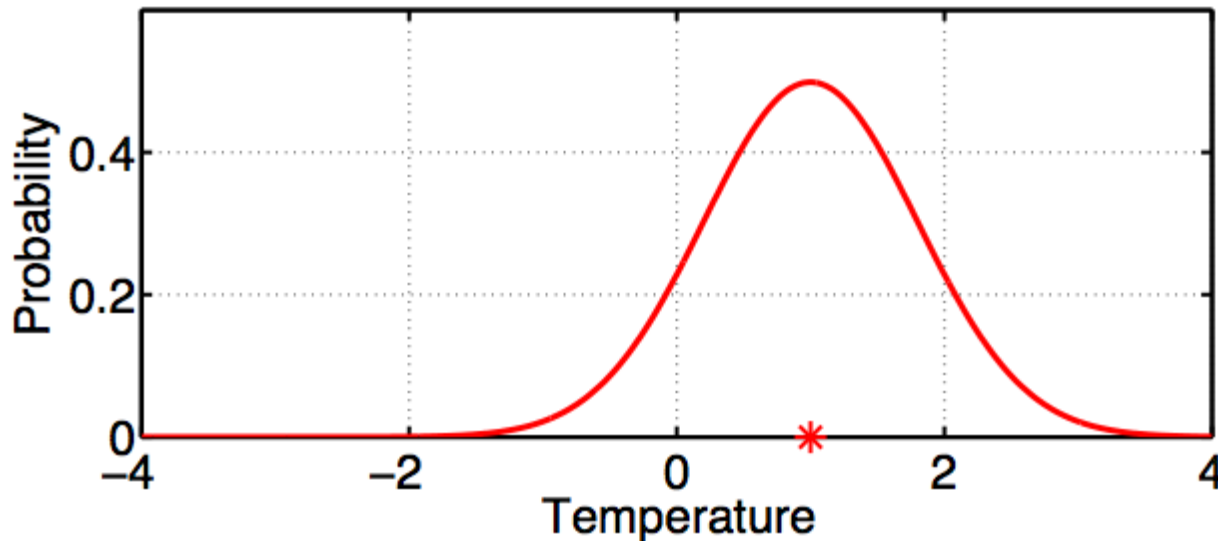
Thermometer outside measures  $1^{\circ}\text{C}$ .



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

# Example: Estimating the Temperature Outside

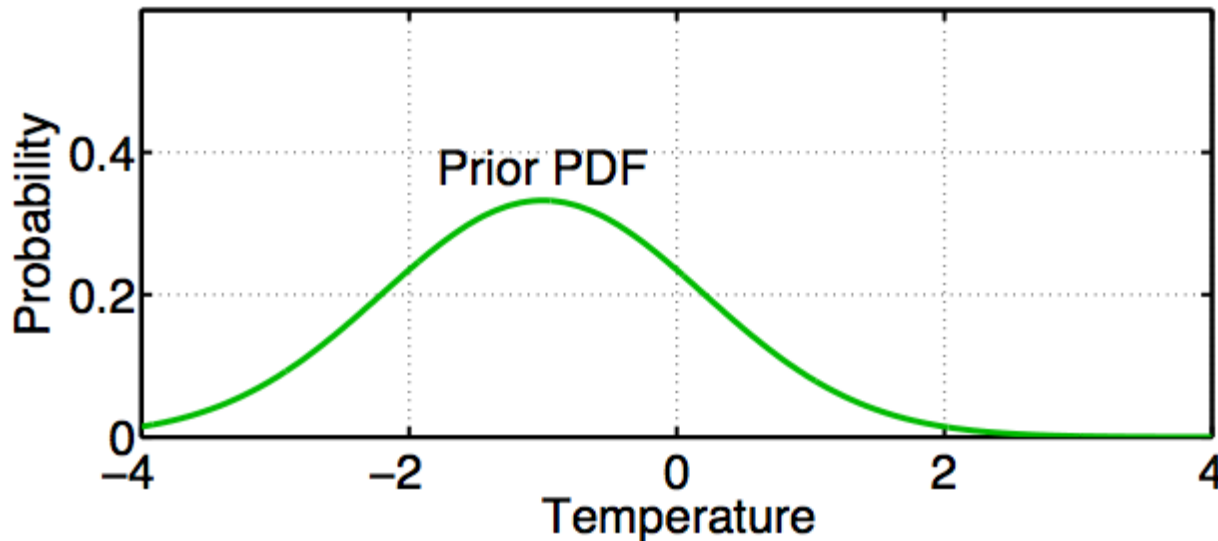
Thermometer outside measures  $1^\circ\text{C}$ .



The red plot is  $P(T | T_0)$ ;  
probability of temperature *given* that  $T_0$  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 (  $T > -273.15^\circ \text{C}$  );
4. Climatological constraints (  $-30^\circ \text{C} < T < 40^\circ \text{C}$  ).

# Combining the Prior Estimate and Observation

Bayes  
Theorem:

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

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

Diagram annotations: An arrow points from the text "Likelihood" to the term  $P(T_o | T, C)$  in the numerator. Another arrow points from the word "Prior" to the term  $P(T | C)$  in the numerator. A third arrow points from the left towards the term  $P(T | T_o, C)$  in the numerator.

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

# Combining the Prior Estimate and Observation

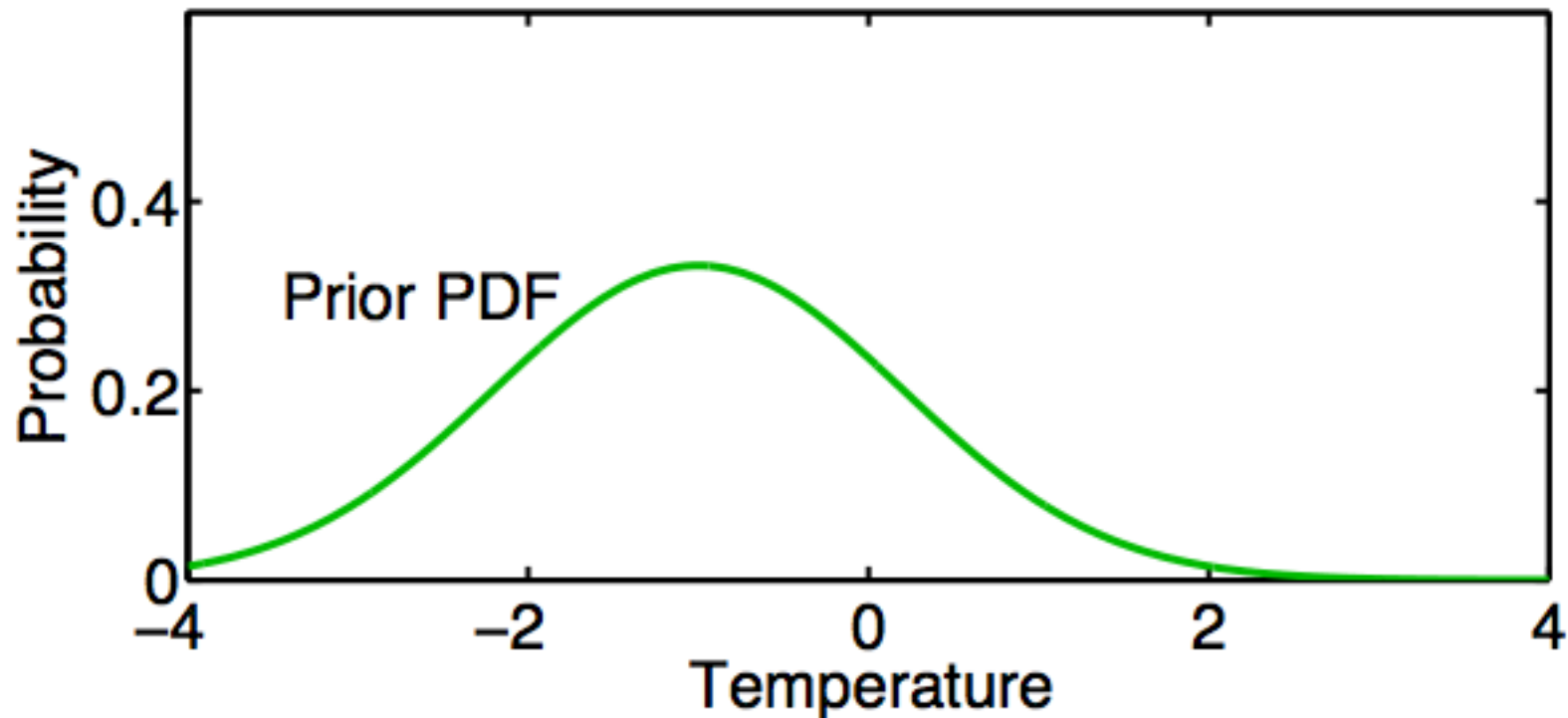
Rewrite Bayes as:

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

Denominator normalizes so Posterior is PDF.

# Combining the Prior Estimate and Observation

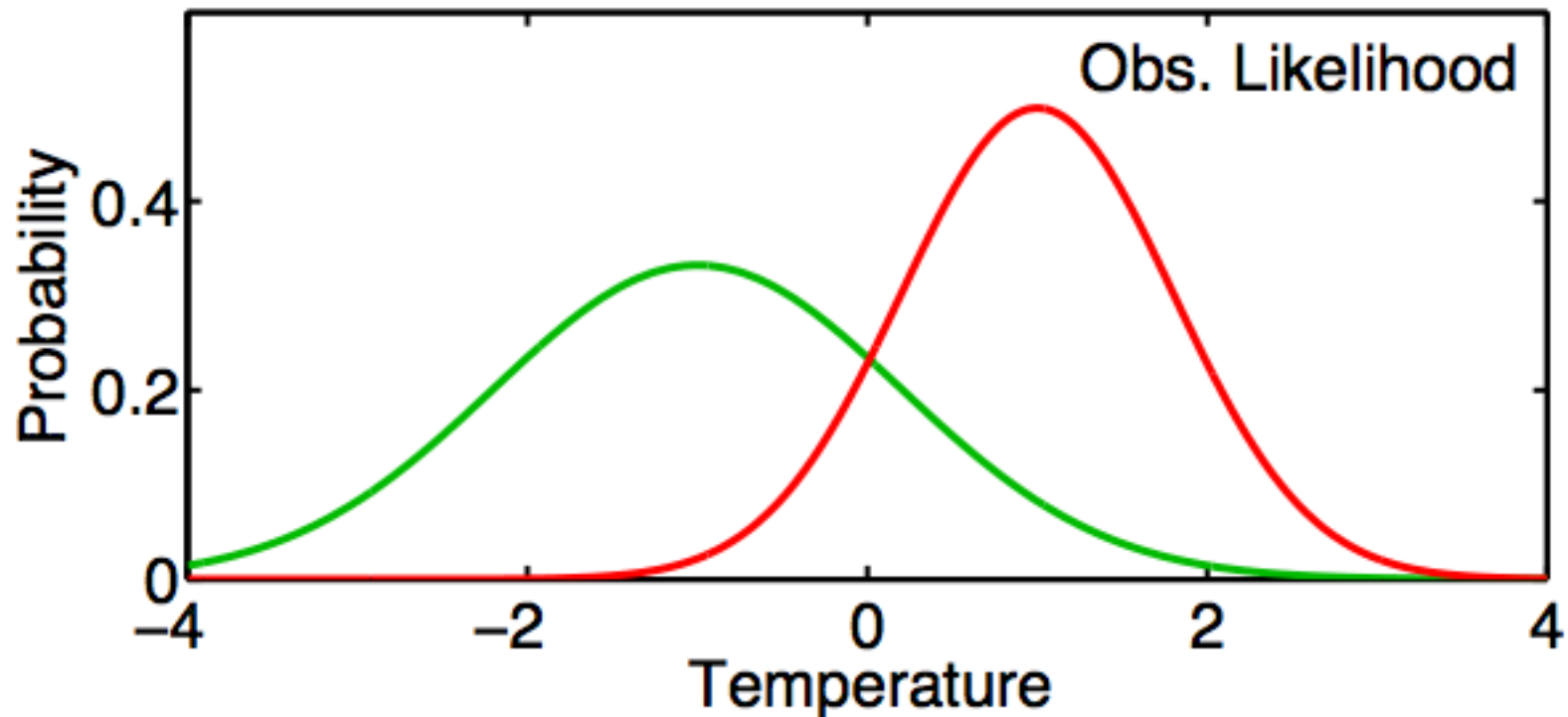
$$P(T | T_0, C) = \frac{P(T_0 | T, C)P(T | C)}{\textit{normalization}}$$





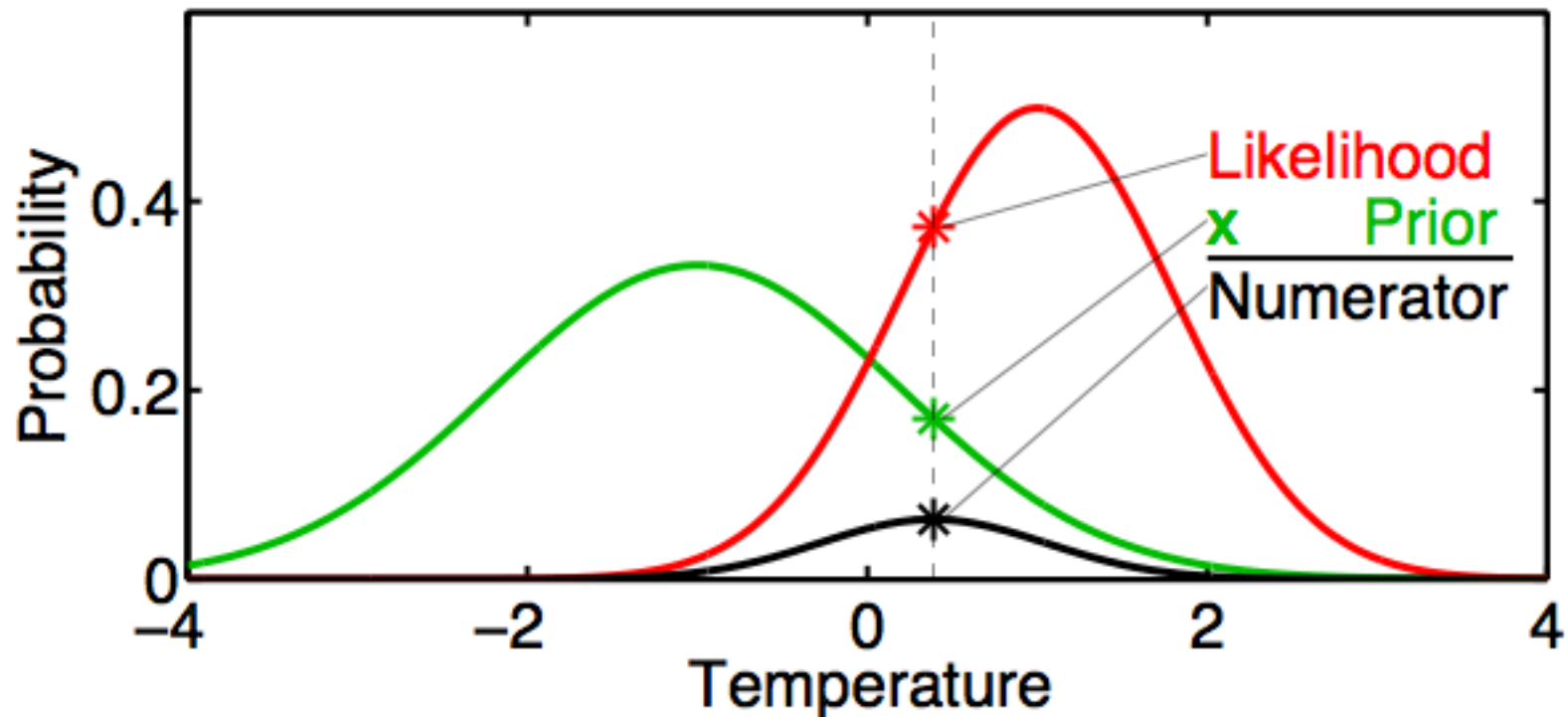
# Combining the Prior Estimate and Observation

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



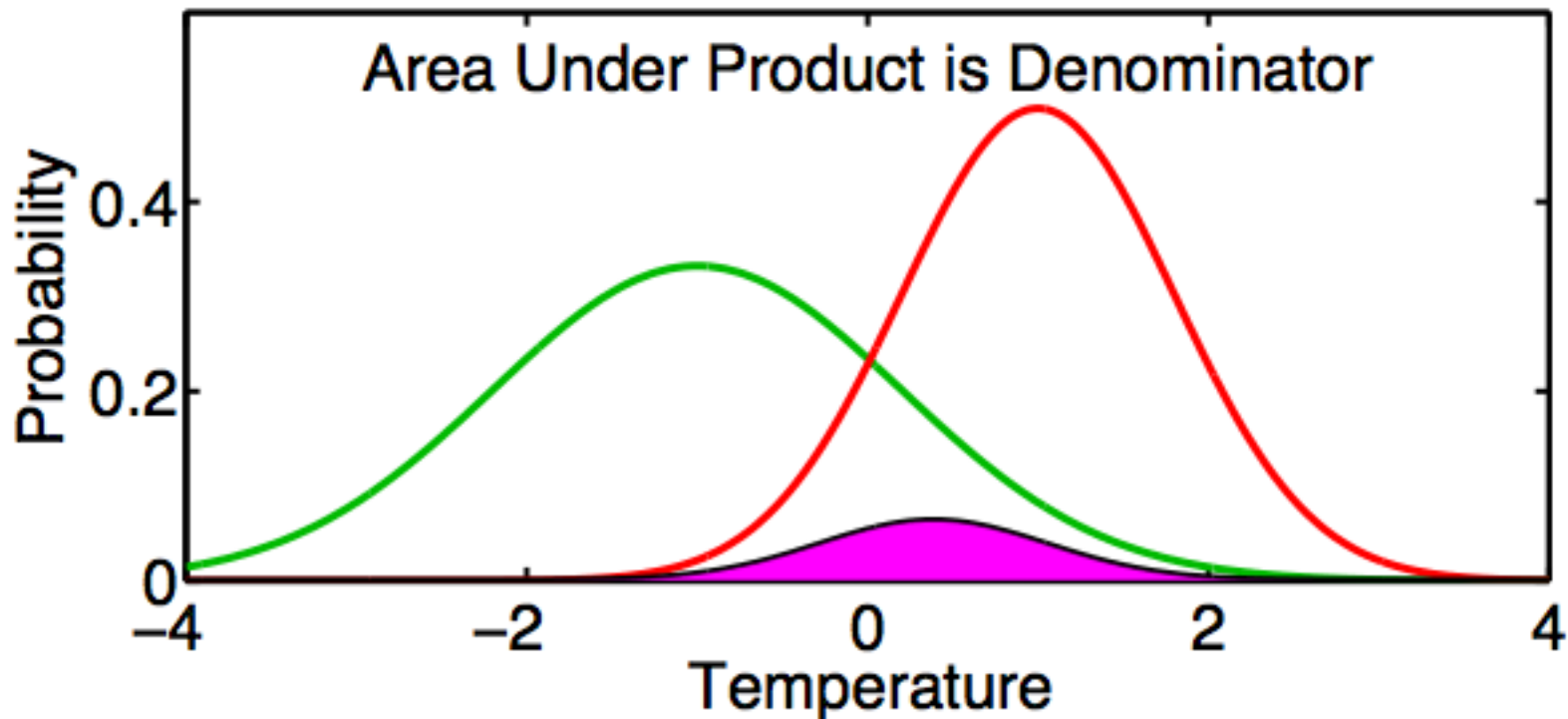
# Combining the Prior Estimate and Observation

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



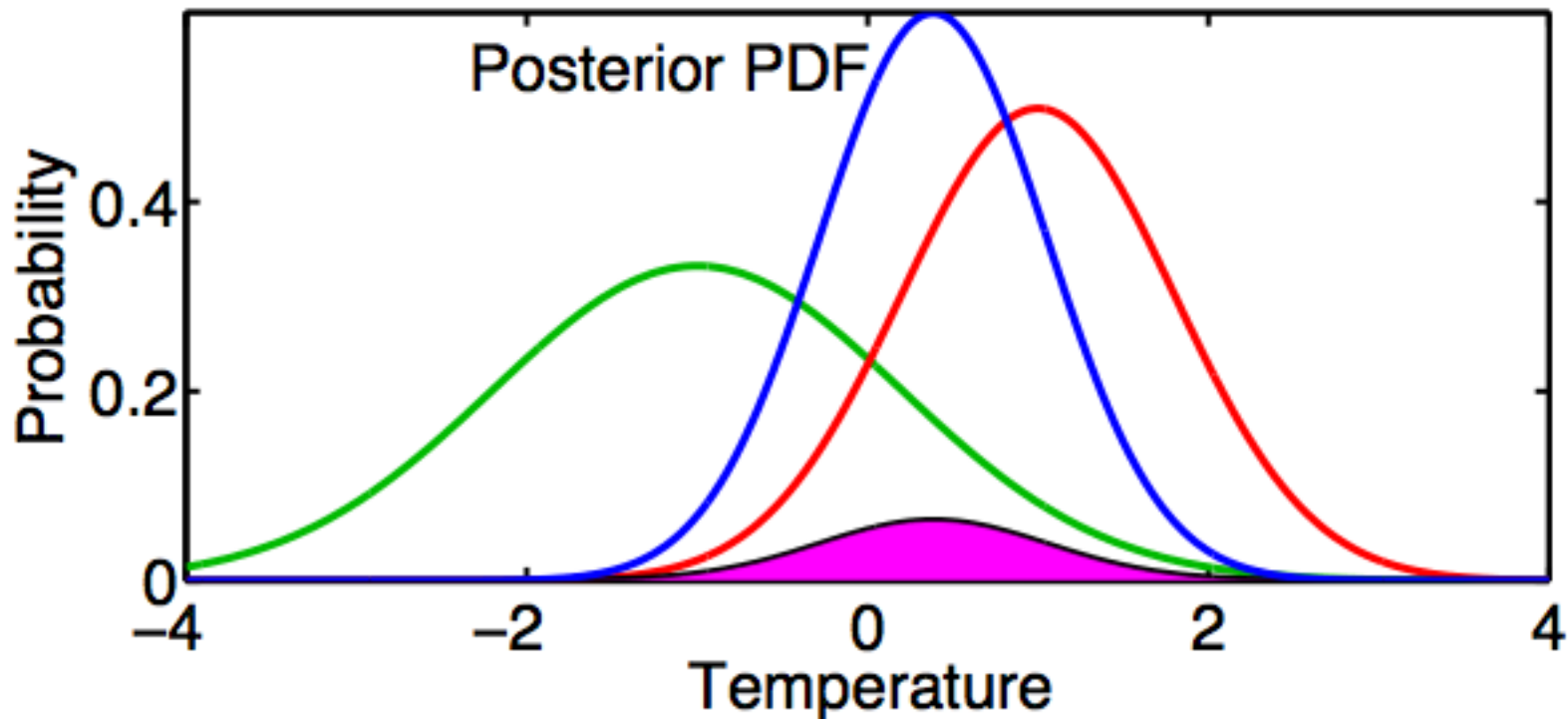
# Combining the Prior Estimate and Observation

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



# Combining the Prior Estimate and Observation

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



# Consistent Color Scheme Throughout Tutorial

Green = Prior

Red = Observation

Blue = Posterior

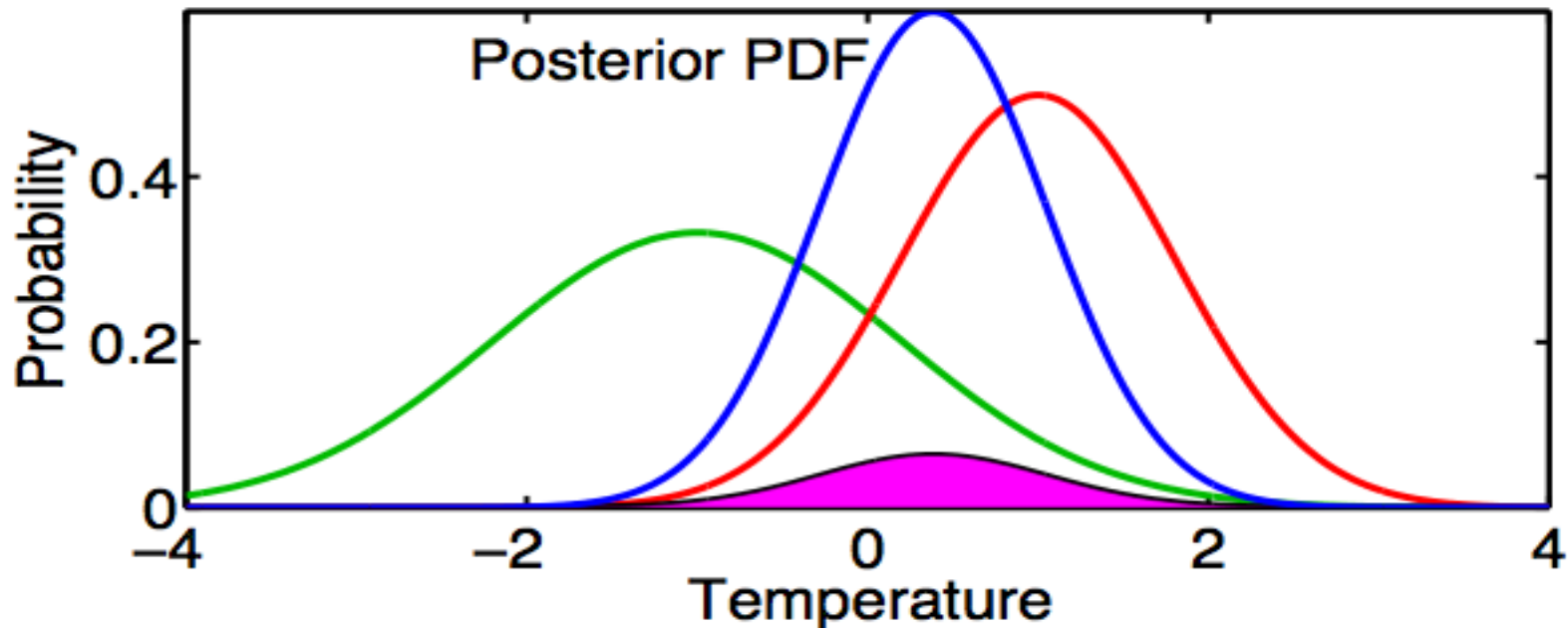
Black = Truth

(truth available only for ‘perfect model’ examples)

# Combining the Prior Estimate and Observation

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

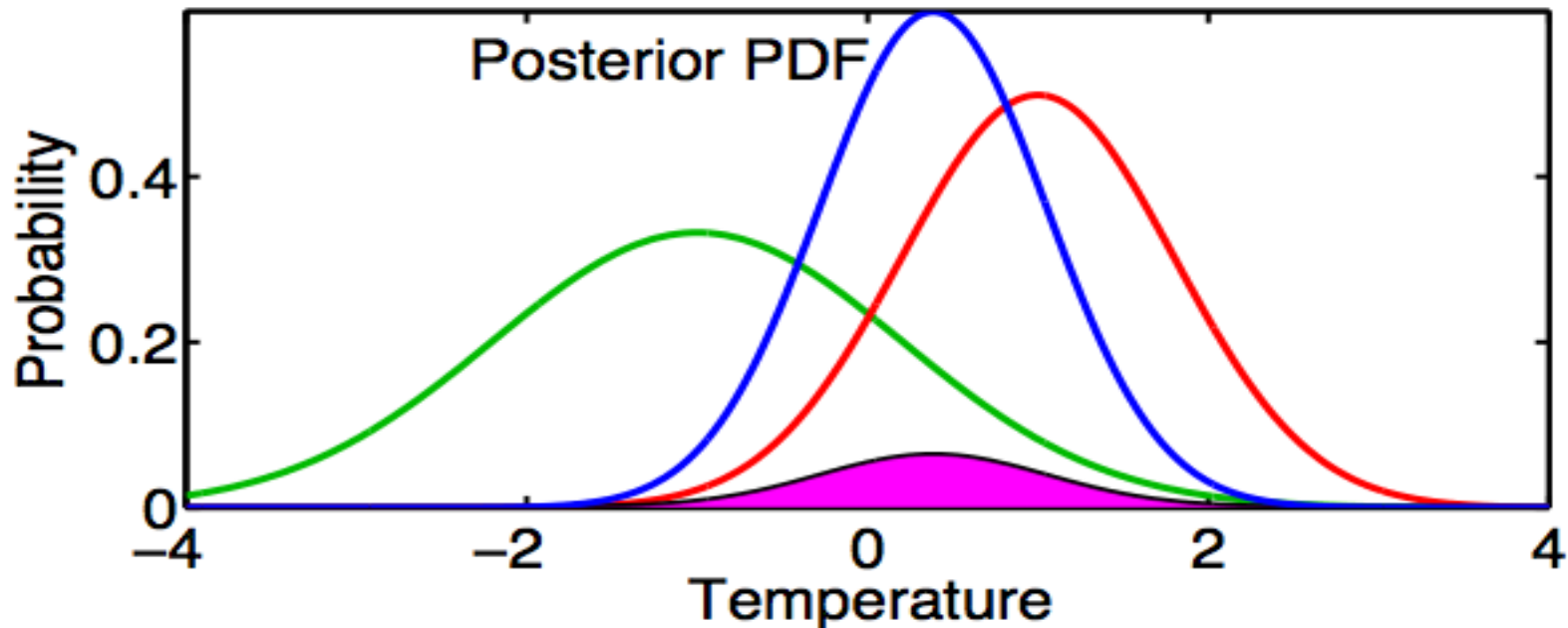
Generally no analytic solution for Posterior.



# Combining the Prior Estimate and Observation

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

Gaussian Prior and Likelihood -> Gaussian Posterior





# Combining the Prior Estimate and Observation

For Gaussian prior and likelihood...

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

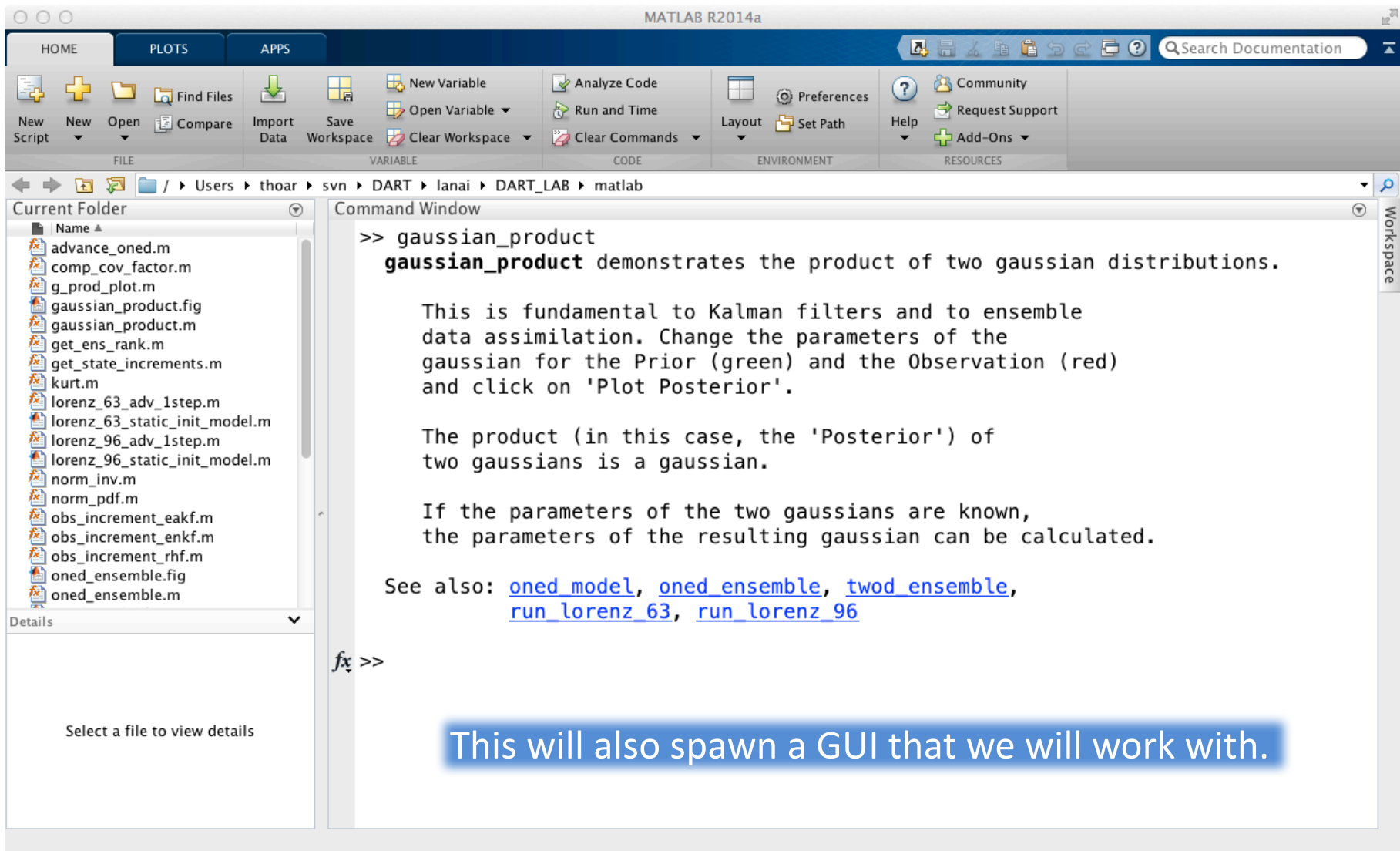
Likelihood  $P(T_o | T, C) = \text{Normal}(T_o, \sigma_o)$

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

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

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

# Matlab Hands-on: gaussian\_product



MATLAB R2014a

HOME PLOTS APPS

Search Documentation

FILE VARIABLE CODE ENVIRONMENT RESOURCES

Users > thoar > svn > DART > lanai > DART\_LAB > matlab

Current Folder

- advance\_oned.m
- comp\_cov\_factor.m
- g\_prod\_plot.m
- gaussian\_product.fig
- gaussian\_product.m
- get\_ens\_rank.m
- get\_state\_increments.m
- kurt.m
- lorenz\_63\_adv\_1step.m
- lorenz\_63\_static\_init\_model.m
- lorenz\_96\_adv\_1step.m
- lorenz\_96\_static\_init\_model.m
- norm\_inv.m
- norm\_pdf.m
- obs\_increment\_eakf.m
- obs\_increment\_enkf.m
- obs\_increment\_rhf.m
- oned\_ensemble.fig
- oned\_ensemble.m

Details

Select a file to view details

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.

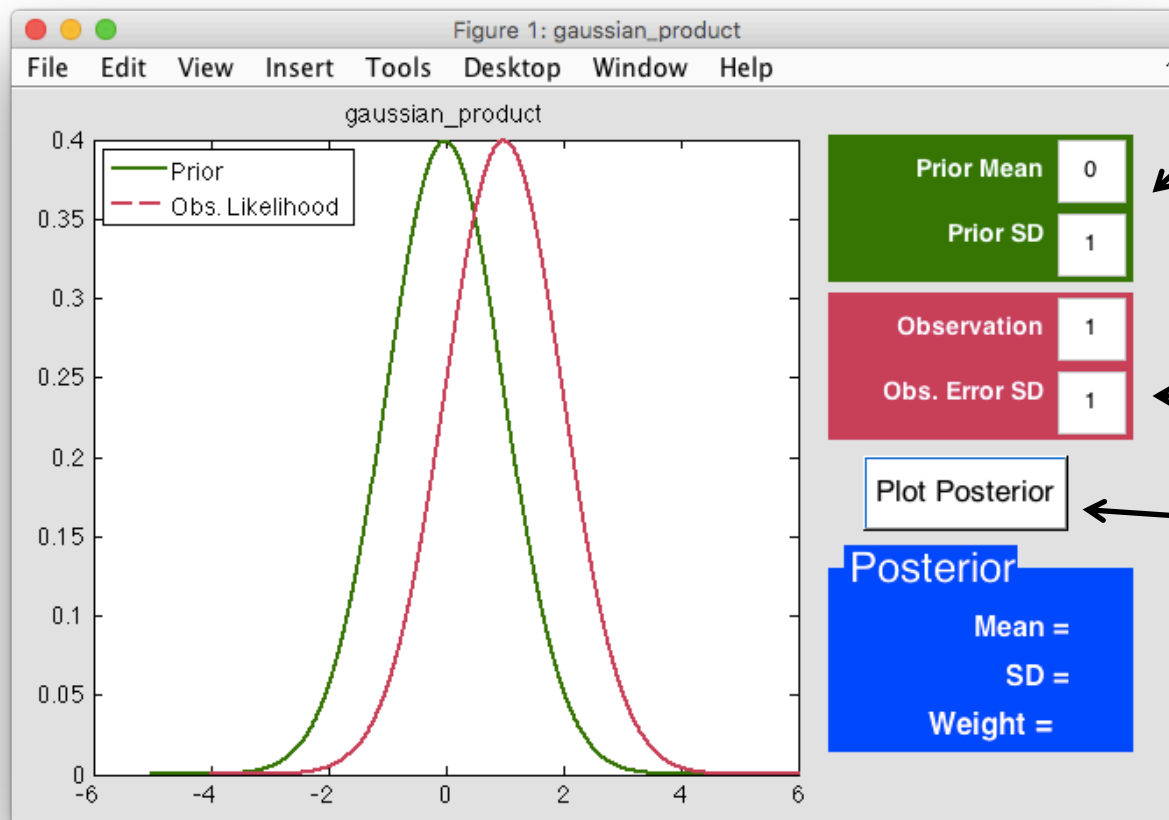
See also: oned\_model, oned\_ensemble, twod\_ensemble,
run\_lorenz\_63, run\_lorenz\_96
```

*fx* >>

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.



1) Set Prior Mean and Standard Deviation.

2) Set Observation Mean and Observation Error Standard Deviation.

3) Select Plot Posterior to Update the items in blue.

## 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  $Normal(t_0, \sigma_0)$ .

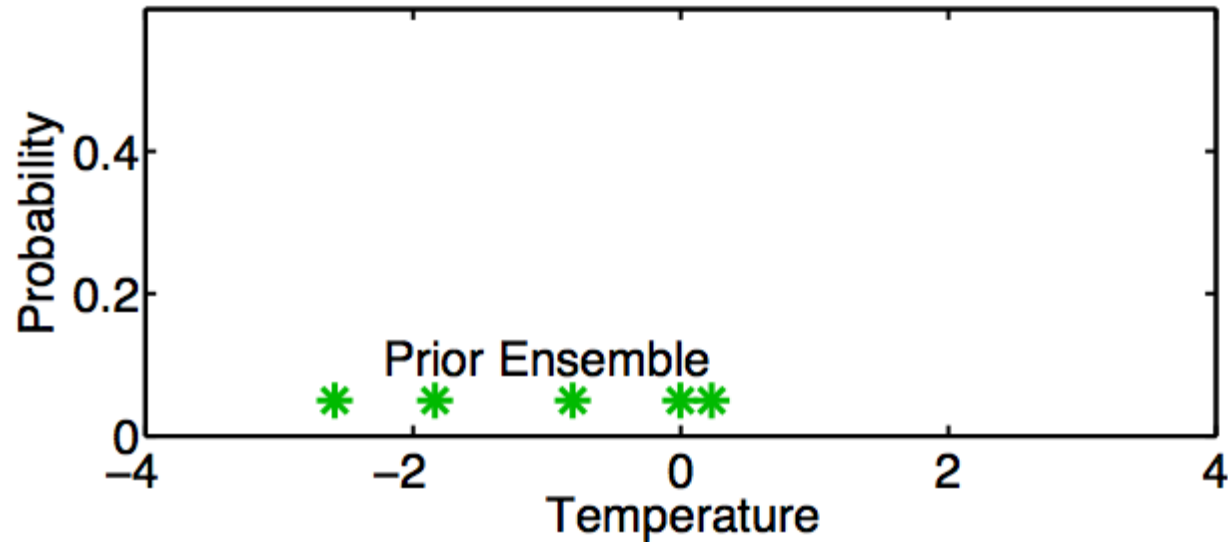


# The One-Dimensional Kalman Filter

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.

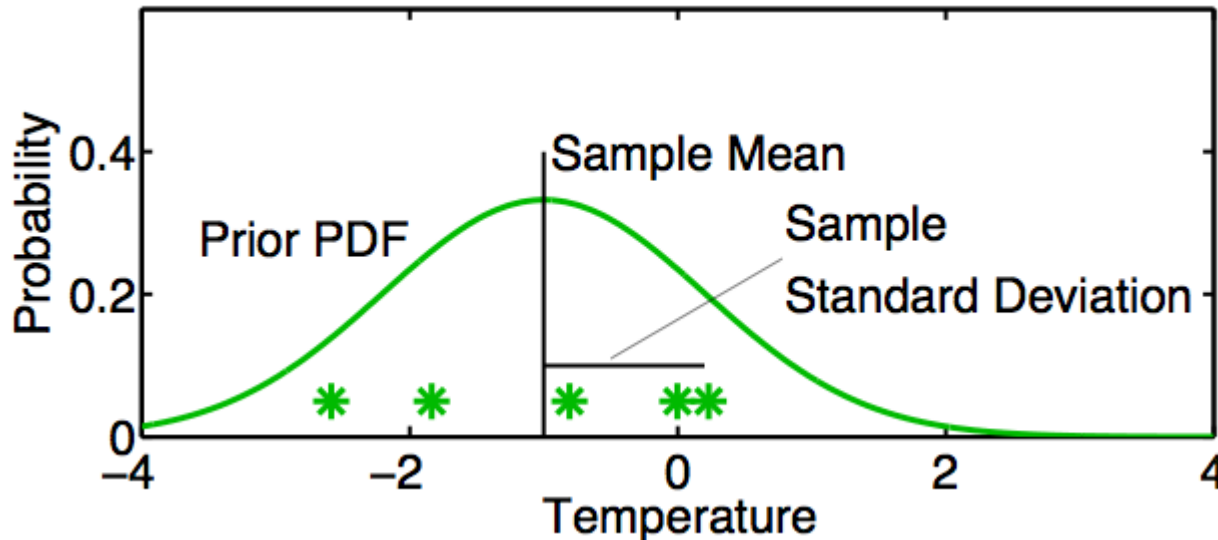
# A One-Dimensional Ensemble Kalman Filter

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:



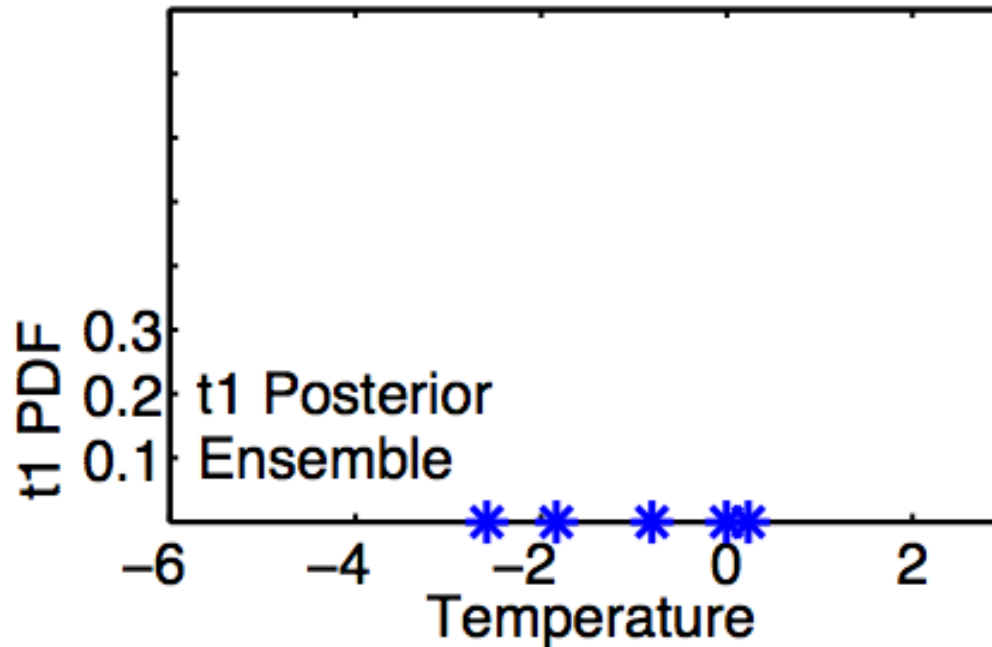
Use sample mean  $\bar{T} = \sum_{n=1}^N T_n / N$

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

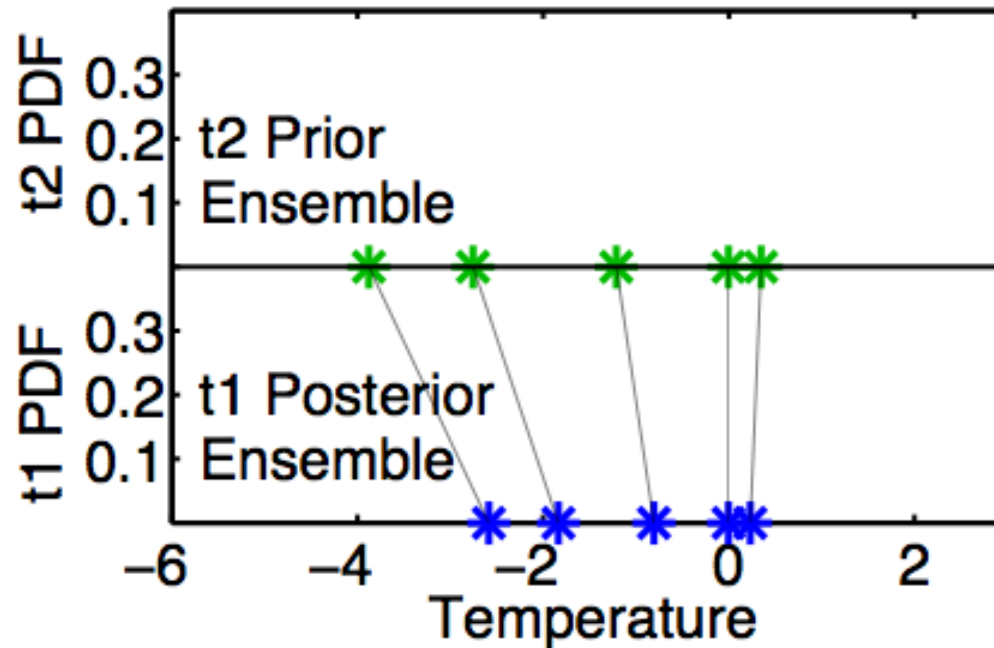
If posterior ensemble at time  $t_1$  is  $T_{1,n}$ ,  $n = 1, \dots, N$



# A One-Dimensional Ensemble Kalman Filter: Model Advance

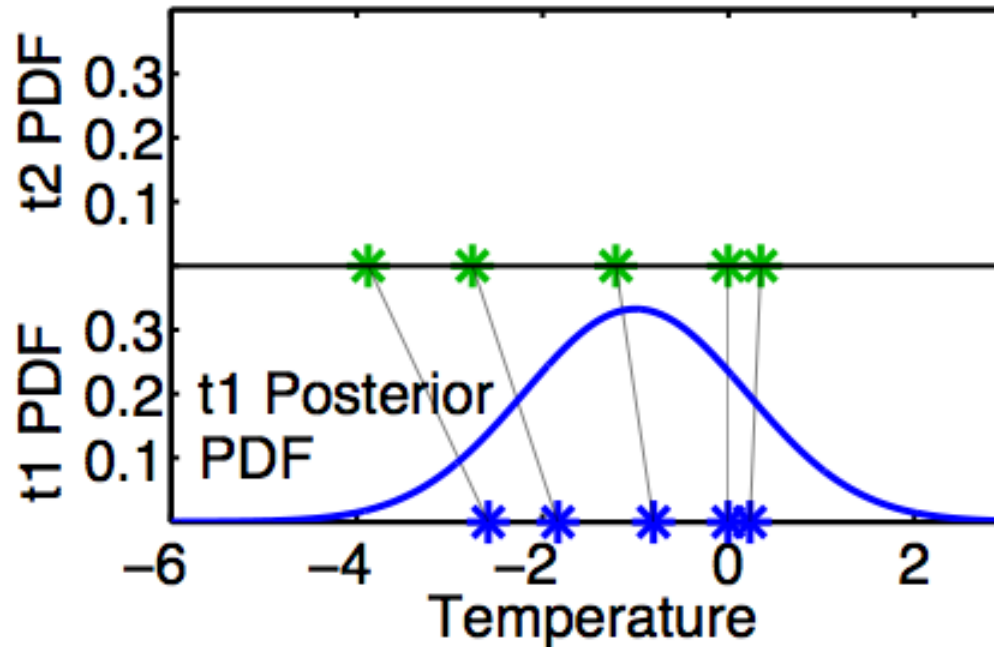
If posterior ensemble at time  $t_1$  is  $T_{1,n}$ ,  $n = 1, \dots, N$

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



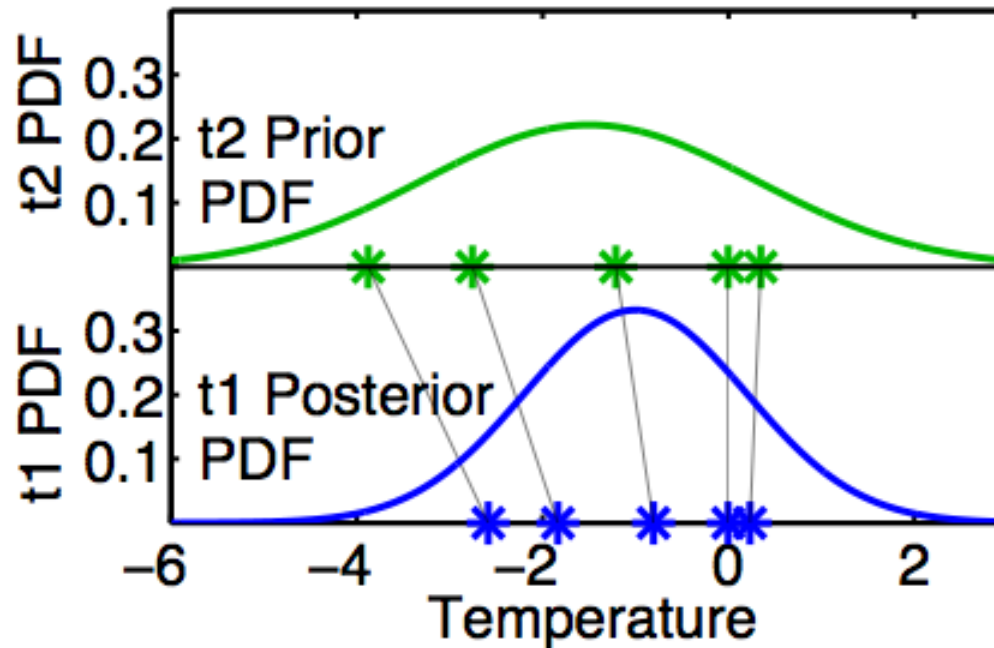
# A One-Dimensional Ensemble Kalman Filter: Model Advance

Same as advancing continuous pdf at time  $t_1$  ...



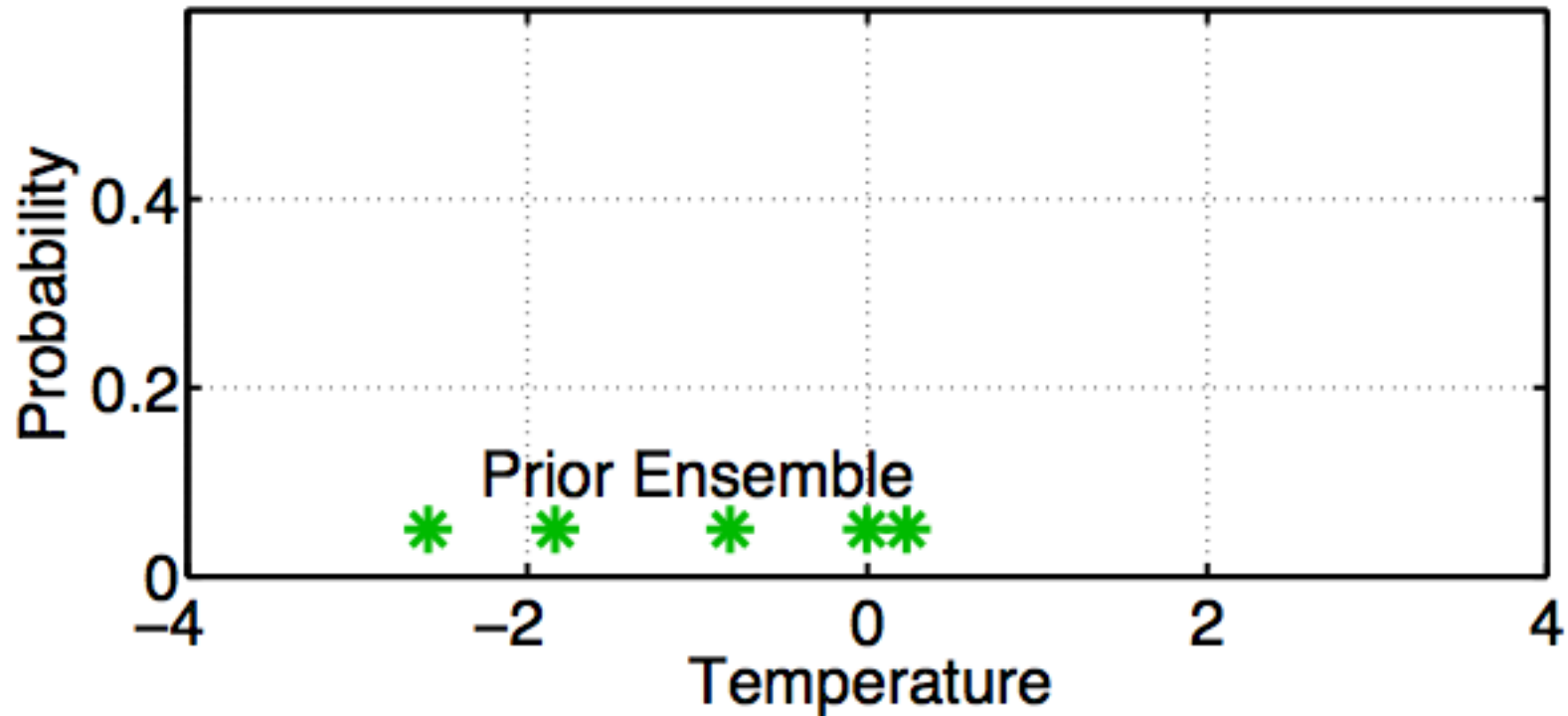
# A One-Dimensional Ensemble Kalman Filter: Model Advance

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

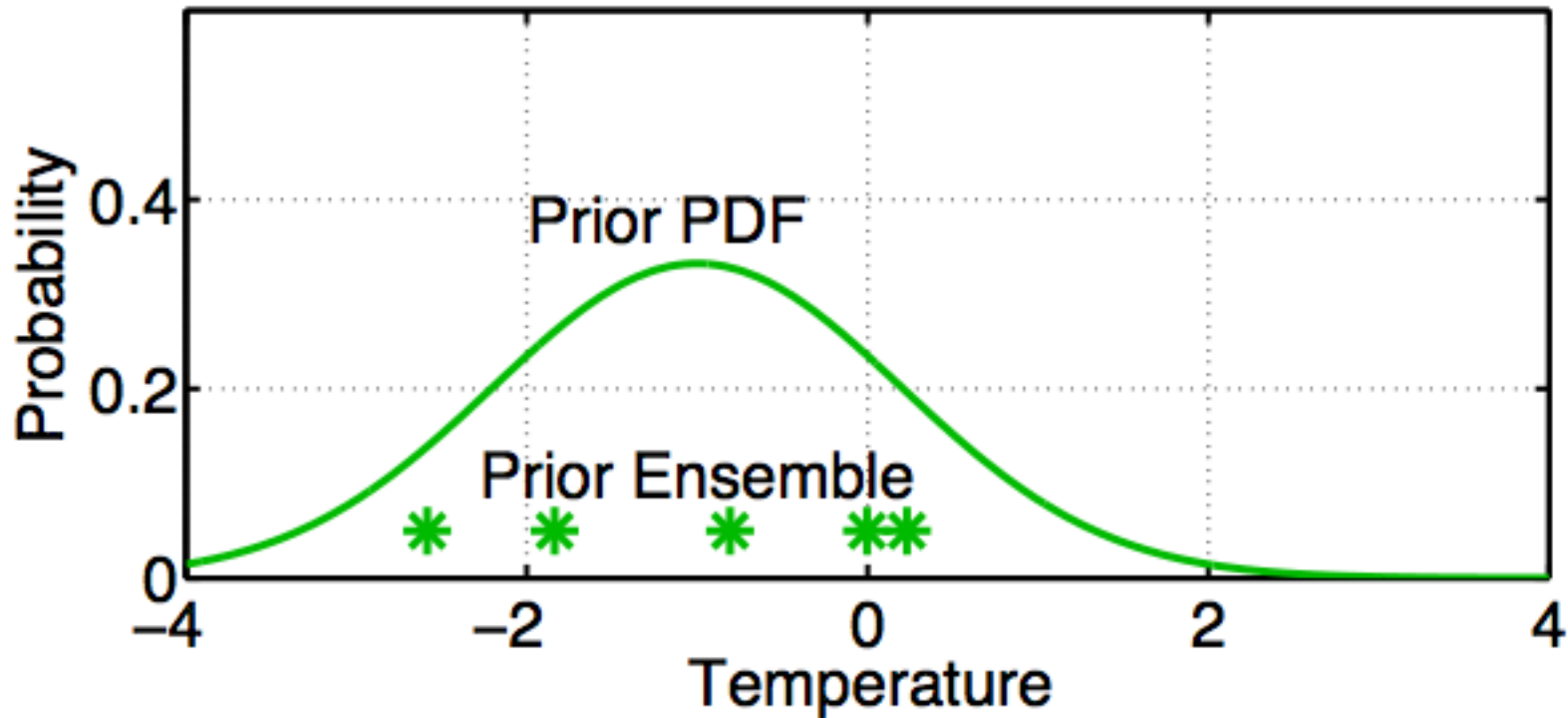




# A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation

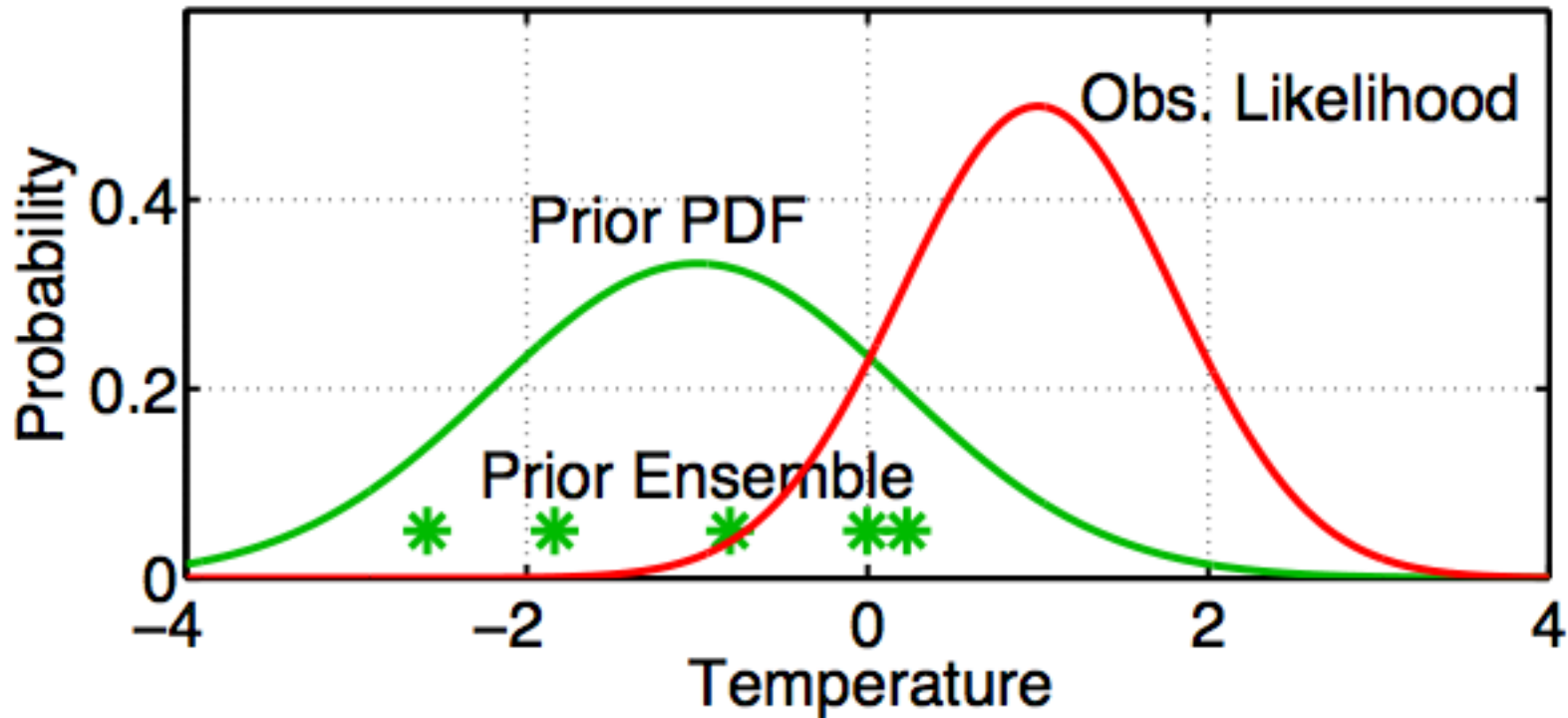


# A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



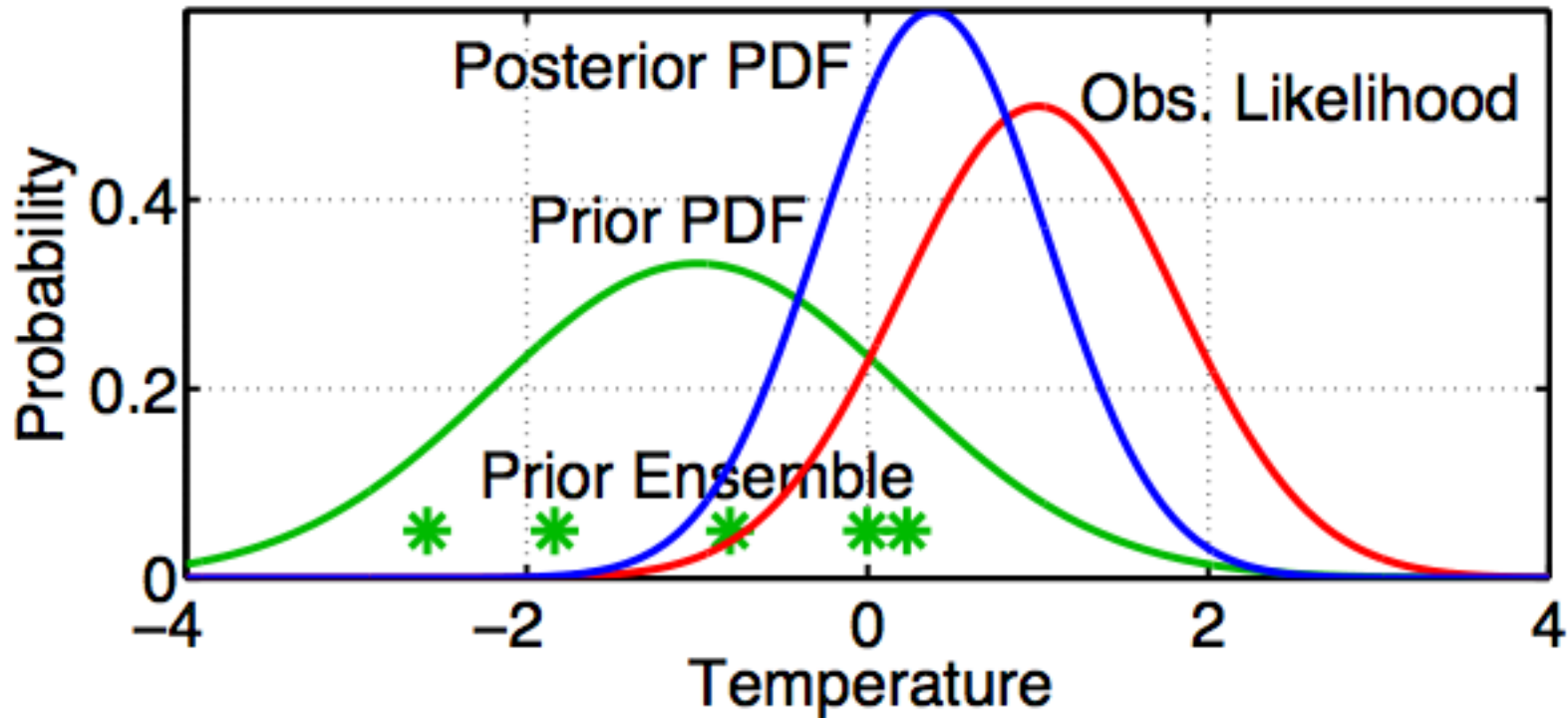
Fit a Gaussian to the sample.

# A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



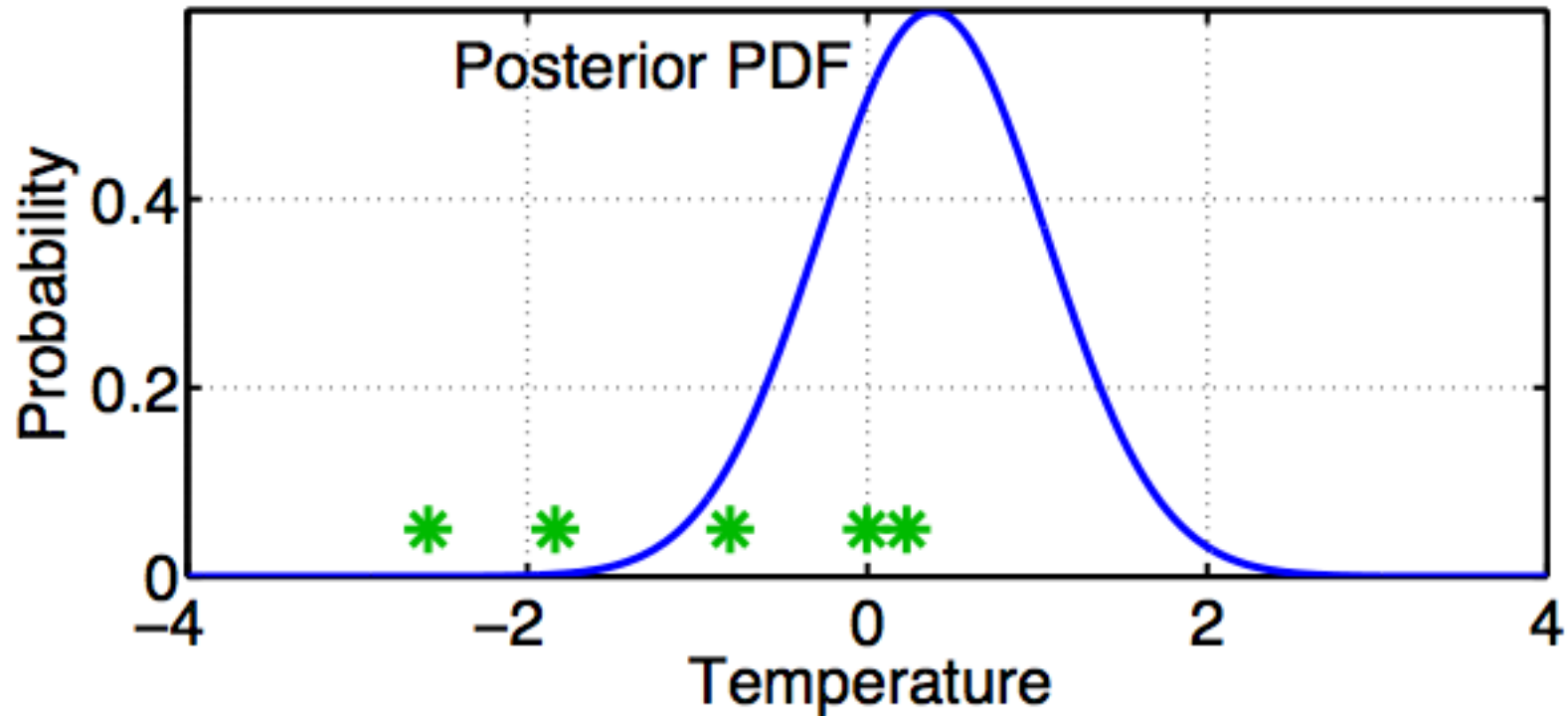
Get the observation likelihood.

# A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



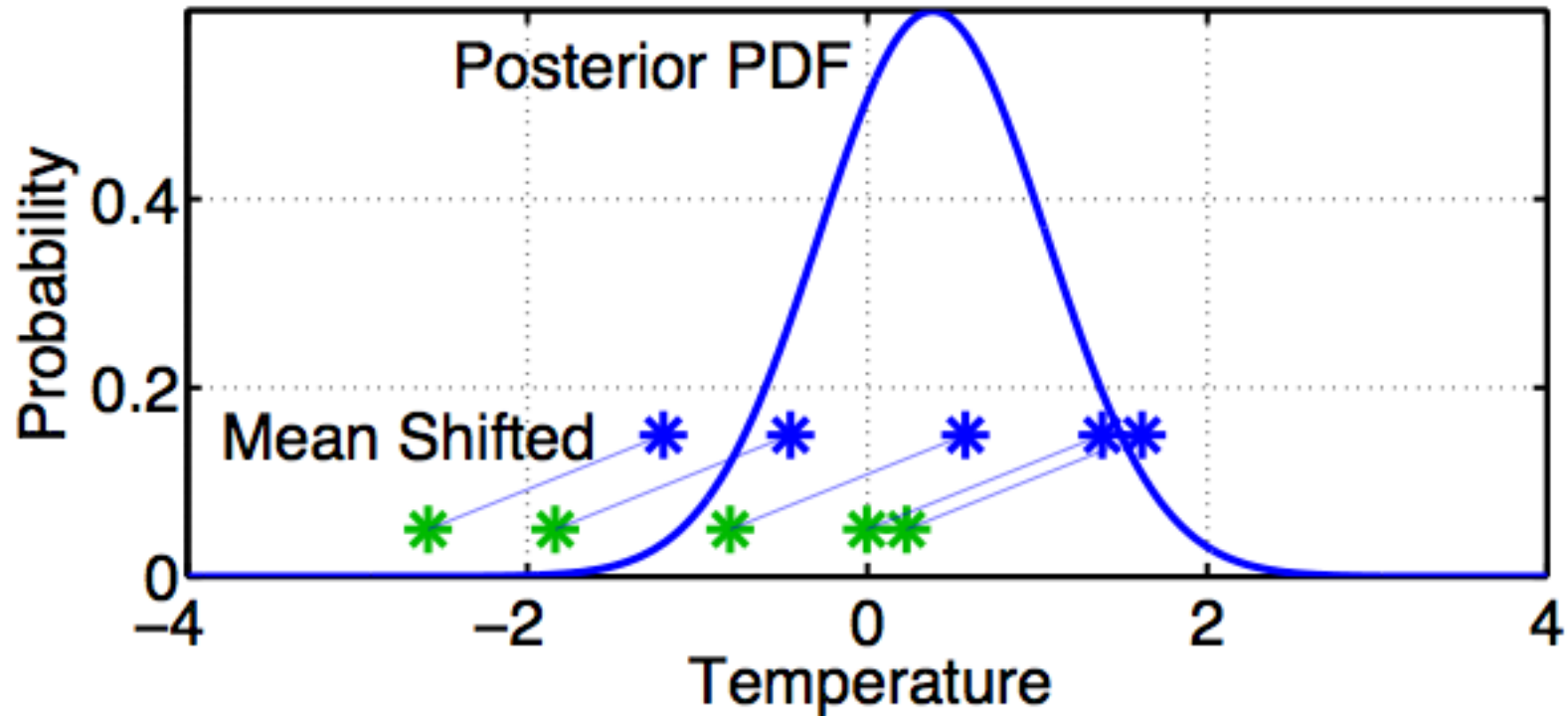
Compute the continuous posterior PDF.

# A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



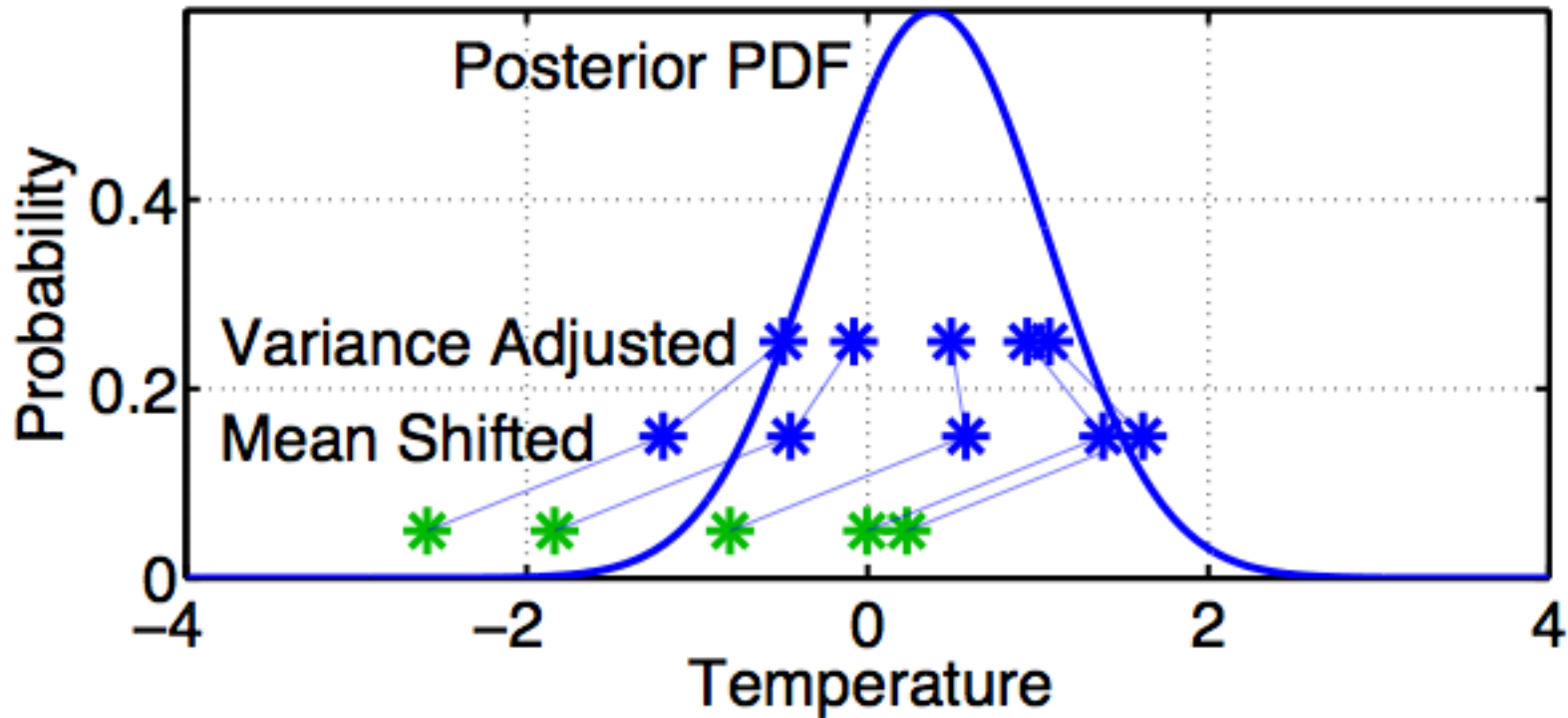
Use a deterministic algorithm to 'adjust' the ensemble.

# A One-Dimensional Ensemble Kalman Filter: Assimilating an Observation



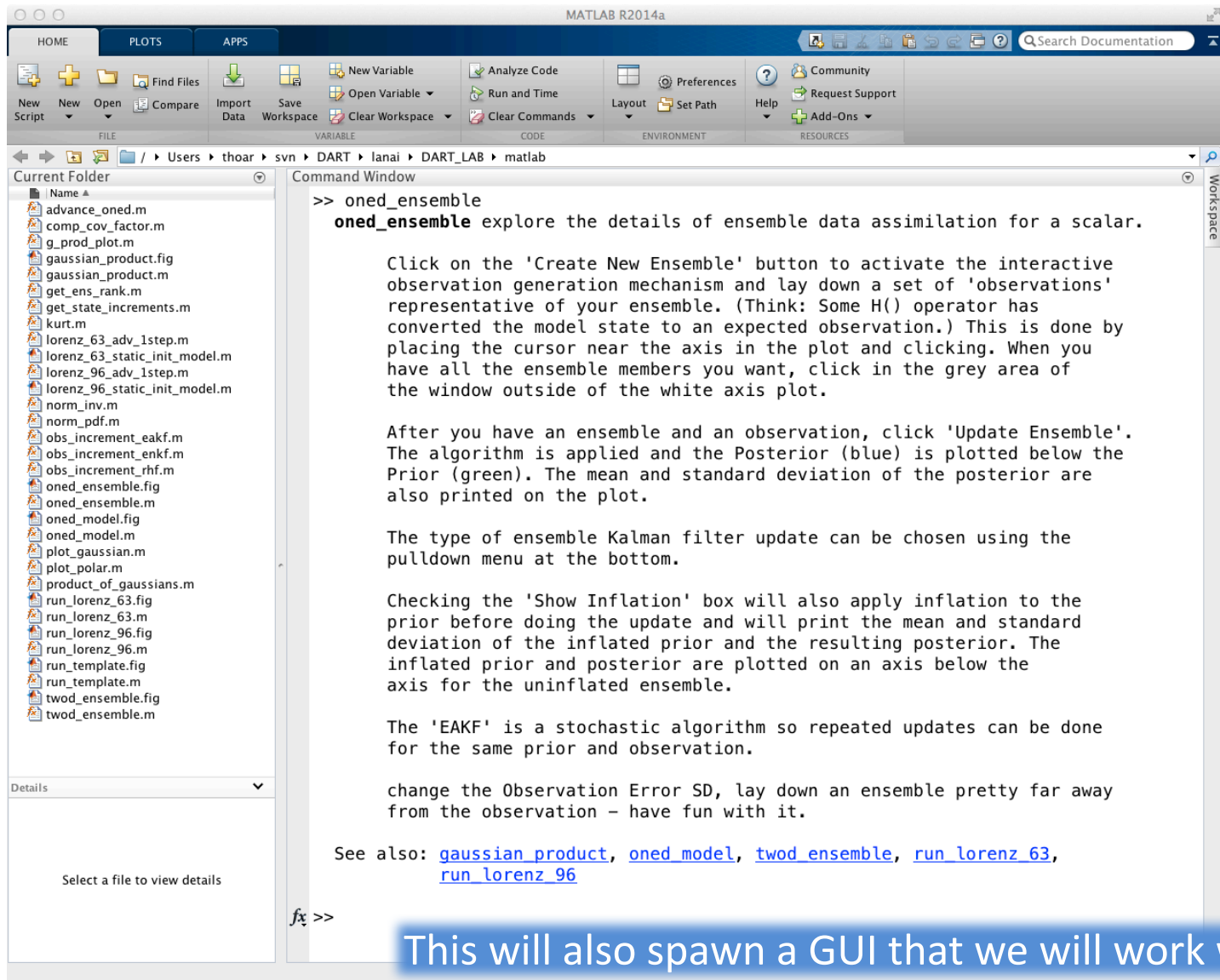
First, 'shift' the ensemble to have the exact mean of the posterior.

# 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



The screenshot displays the MATLAB R2014a environment. The Command Window is active, showing the following text:

```
>> oned_ensemble
oned_ensemble explore the details of ensemble data assimilation for a scalar.
```

Click on the 'Create New Ensemble' button to activate the interactive observation generation mechanism and lay down a set of 'observations' representative of your ensemble. (Think: Some H() operator has converted the model state to an expected observation.) This is done by placing the cursor near the axis in the plot and clicking. When you have all the ensemble members you want, click in the grey area of the window outside of the white axis plot.

After you have an ensemble and an observation, click 'Update Ensemble'. The algorithm is applied and the Posterior (blue) is plotted below the Prior (green). The mean and standard deviation of the posterior are also printed on the plot.

The type of ensemble Kalman filter update can be chosen using the pulldown menu at the bottom.

Checking the 'Show Inflation' box will also apply inflation to the prior before doing the update and will print the mean and standard deviation of the inflated prior and the resulting posterior. The inflated prior and posterior are plotted on an axis below the axis for the uninflated ensemble.

The 'EAKF' is a stochastic algorithm so repeated updates can be done for the same prior and observation.

change the Observation Error SD, lay down an ensemble pretty far away from the observation - have fun with it.

See also: [gaussian\\_product](#), [oned\\_model](#), [twod\\_ensemble](#), [run\\_lorenz\\_63](#), [run\\_lorenz\\_96](#)

fx >>

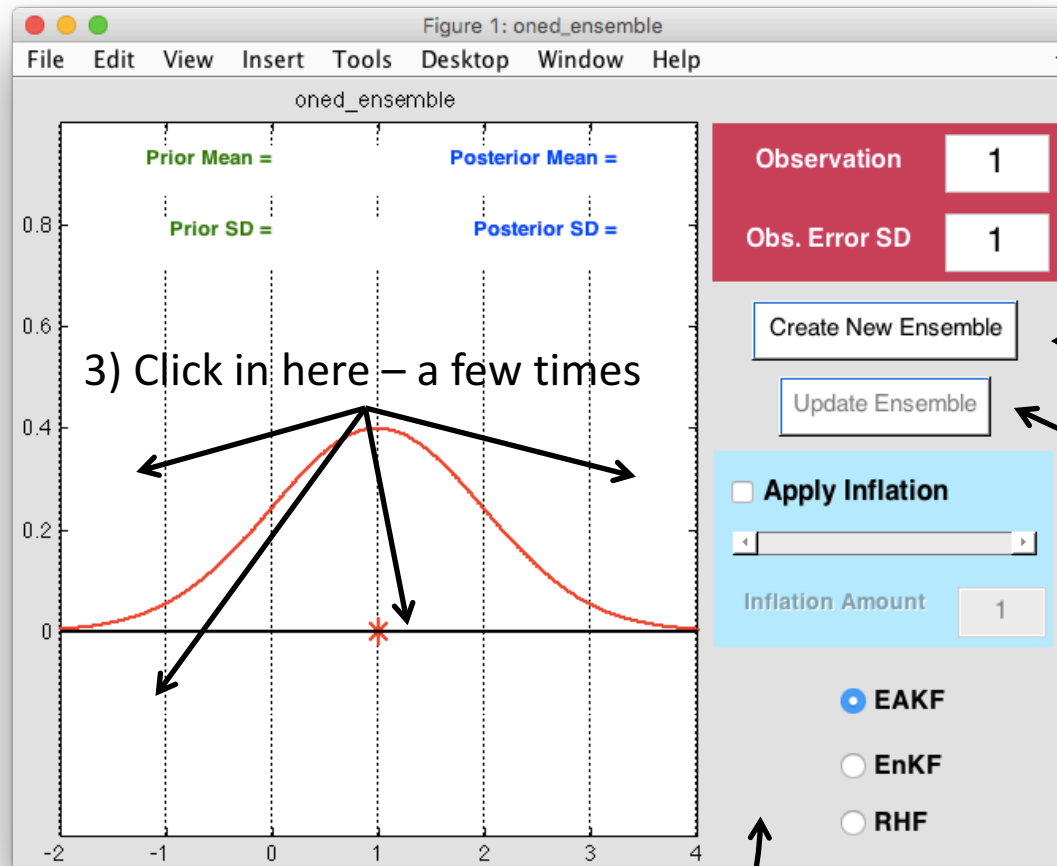
The interface also shows a file explorer on the left with a list of files in the current folder, including `advance_oned.m`, `comp_cov_factor.m`, `g_prod_plot.m`, `gaussian_product.fig`, `gaussian_product.m`, `get_ens_rank.m`, `get_state_increments.m`, `kurt.m`, `lorenz_63_adv_1step.m`, `lorenz_63_static_init_model.m`, `lorenz_96_adv_1step.m`, `lorenz_96_static_init_model.m`, `norm_inv.m`, `norm_pdf.m`, `obs_increment_eakf.m`, `obs_increment_eknf.m`, `obs_increment_rhf.m`, `oned_ensemble.fig`, `oned_ensemble.m`, `oned_model.fig`, `oned_model.m`, `plot_gaussian.m`, `plot_polar.m`, `product_of_gaussians.m`, `run_lorenz_63.fig`, `run_lorenz_63.m`, `run_lorenz_96.fig`, `run_lorenz_96.m`, `run_template.fig`, `run_template.m`, `twod_ensemble.fig`, and `twod_ensemble.m`.

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



# Matlab Hands-On: oned\_ensemble

Purpose: Explore how ensemble filters update a prior ensemble.



1) change these if you want to.

2) Click on Create New Ensemble

3) Click in here - a few times

5) Click on Update Ensemble

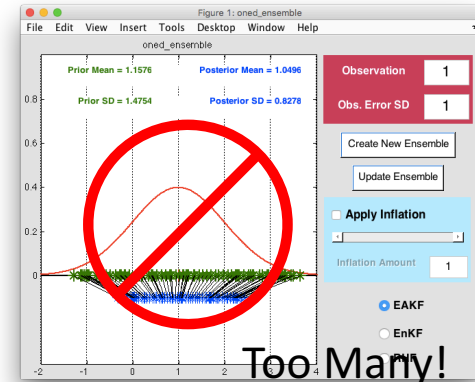
4) Click outside the axis on the gray (anywhere) to finish defining the ensemble.

Ignore the Inflation and EAKF menus for now.

# 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?



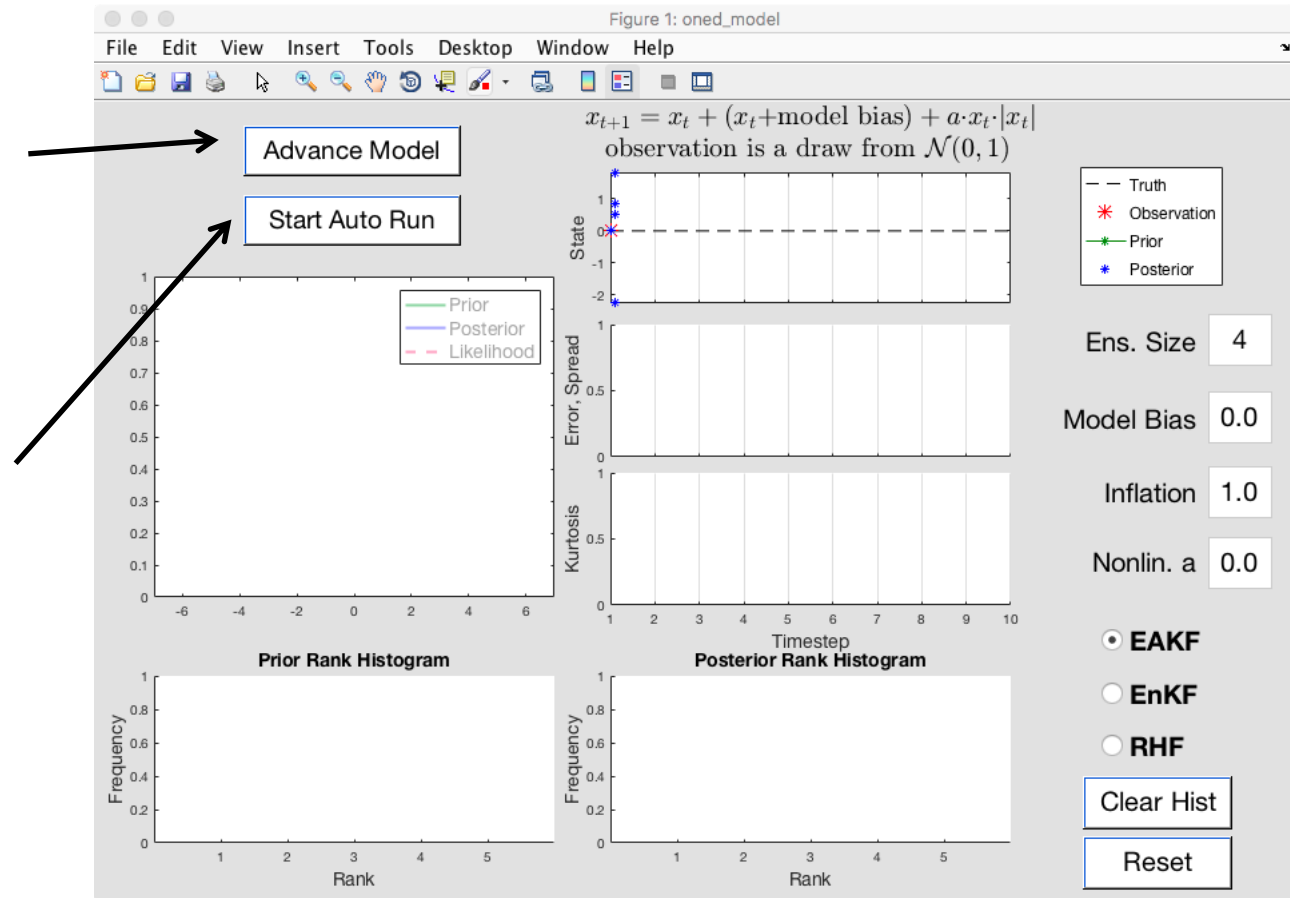
# Matlab Hands-On: oned\_model

Purpose:

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

Top button allows alternating model advance and assimilation steps.

Or automatically sequence advances and assimilations.



Notes:

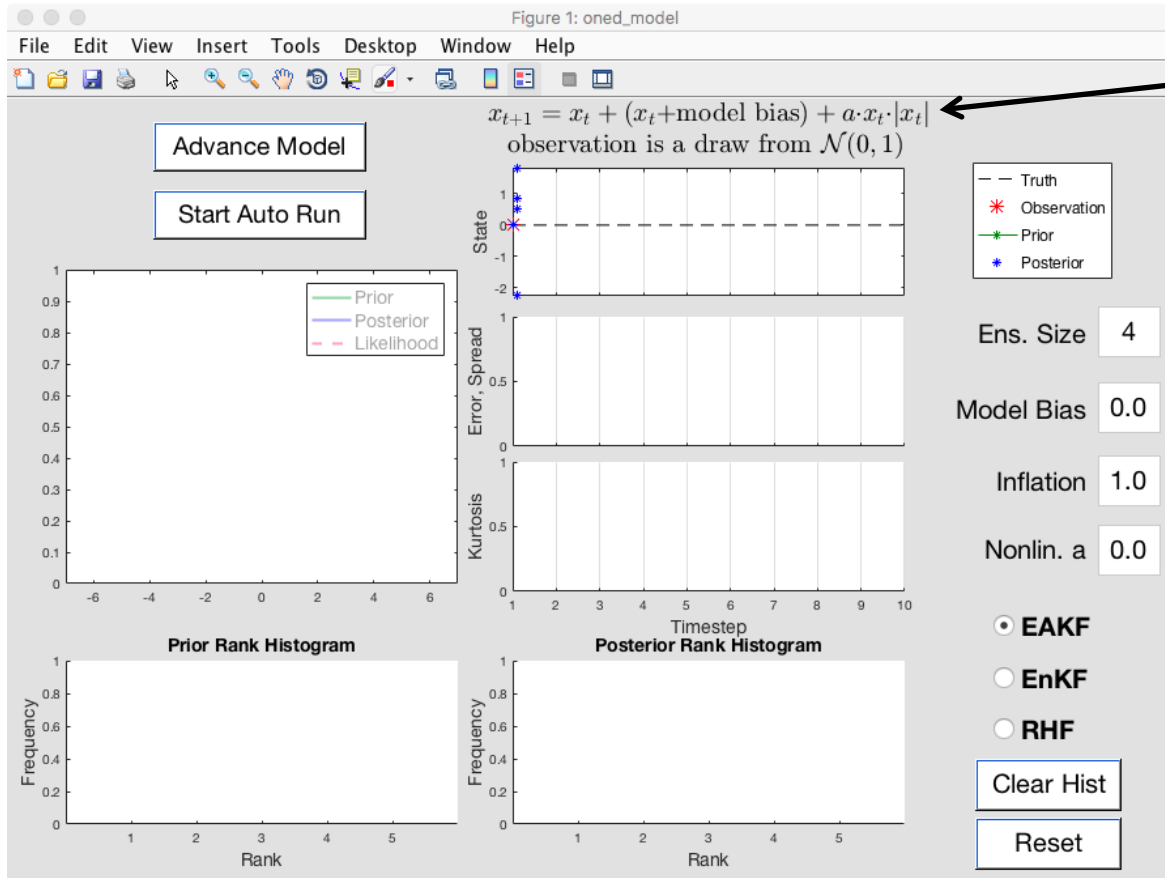
The 'truth' is always 0.

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

# 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.



This is the equation for the model time tendency.

Change the ensemble size or the model parameters.

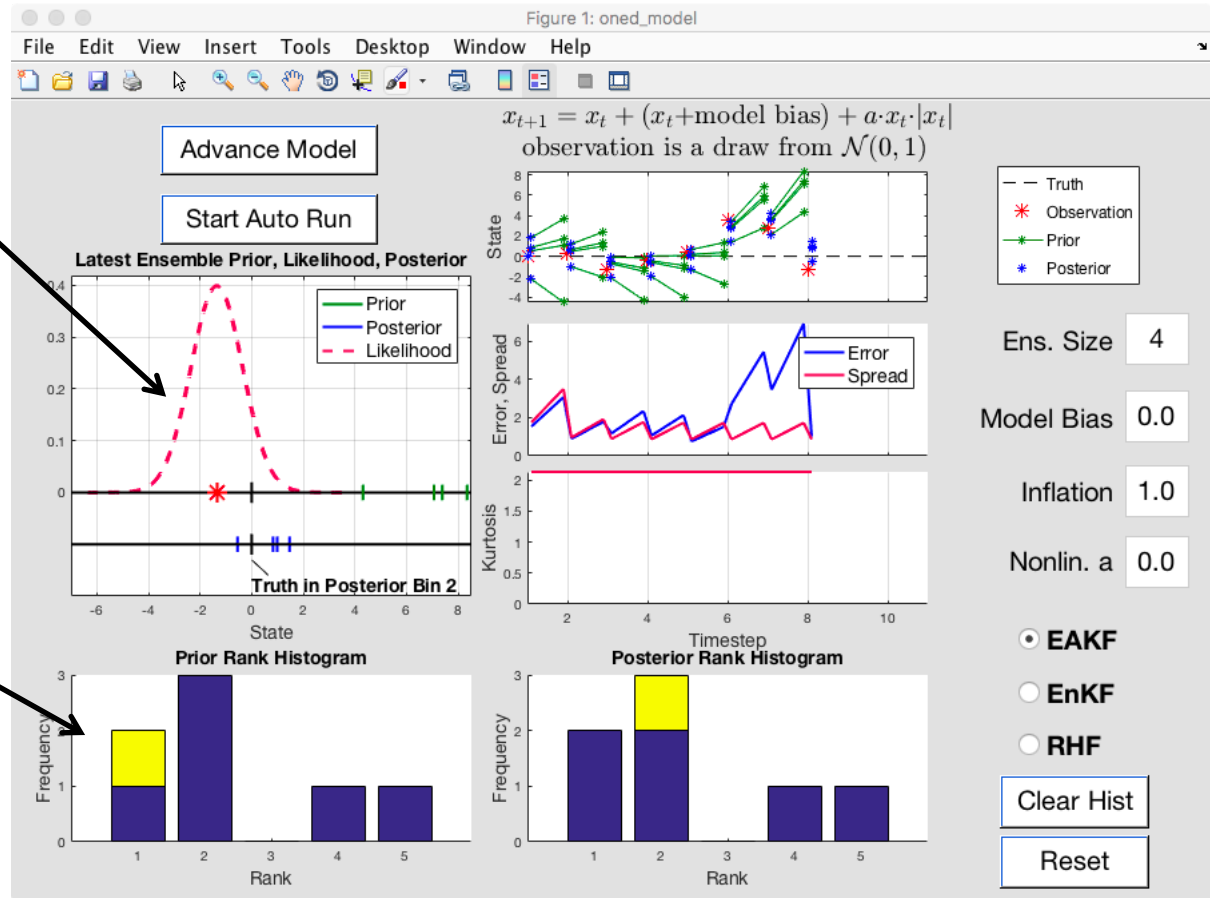
# Matlab Hands-On: oned\_model

Purpose:

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

Prior ensemble members green ticks, Posterior blue, observation is \*. Truth is always 0.

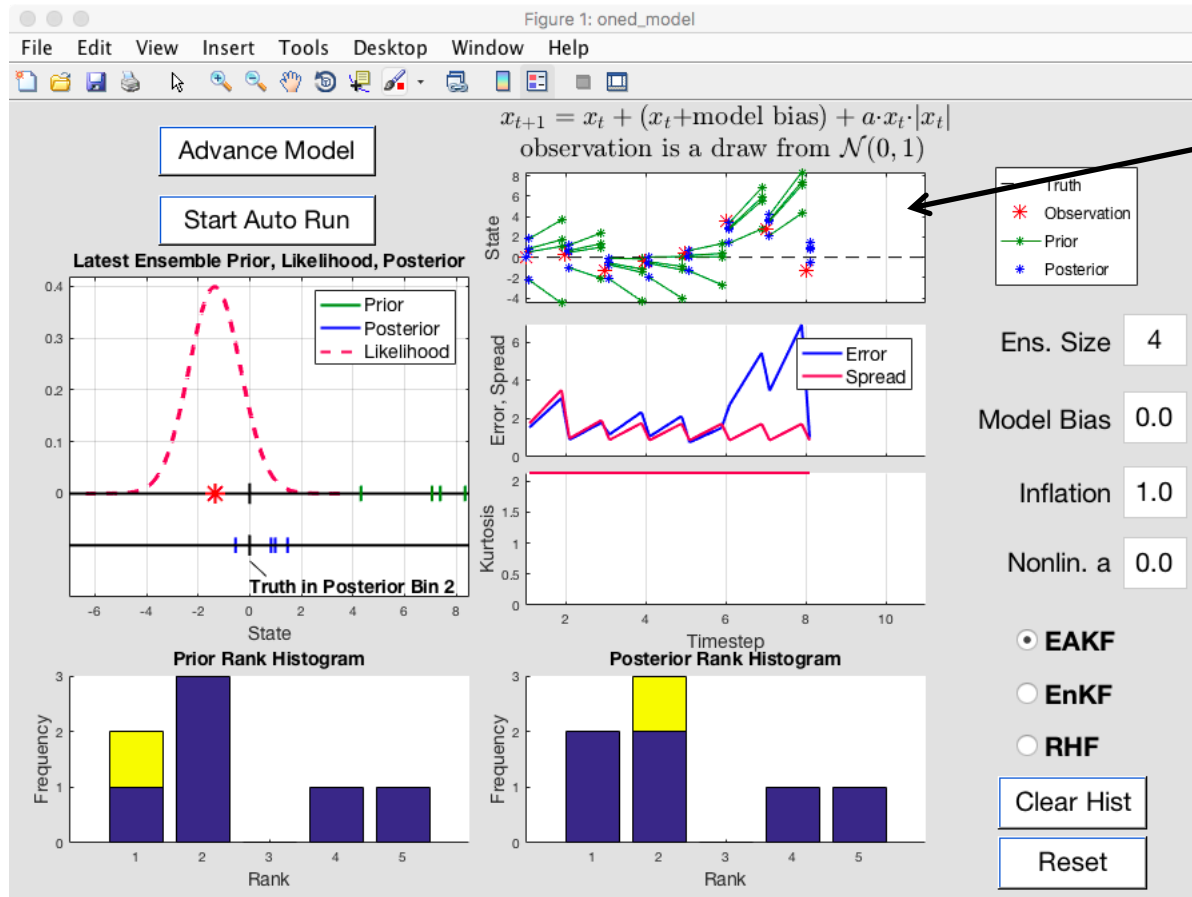
More on rank histograms shortly.



# 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.

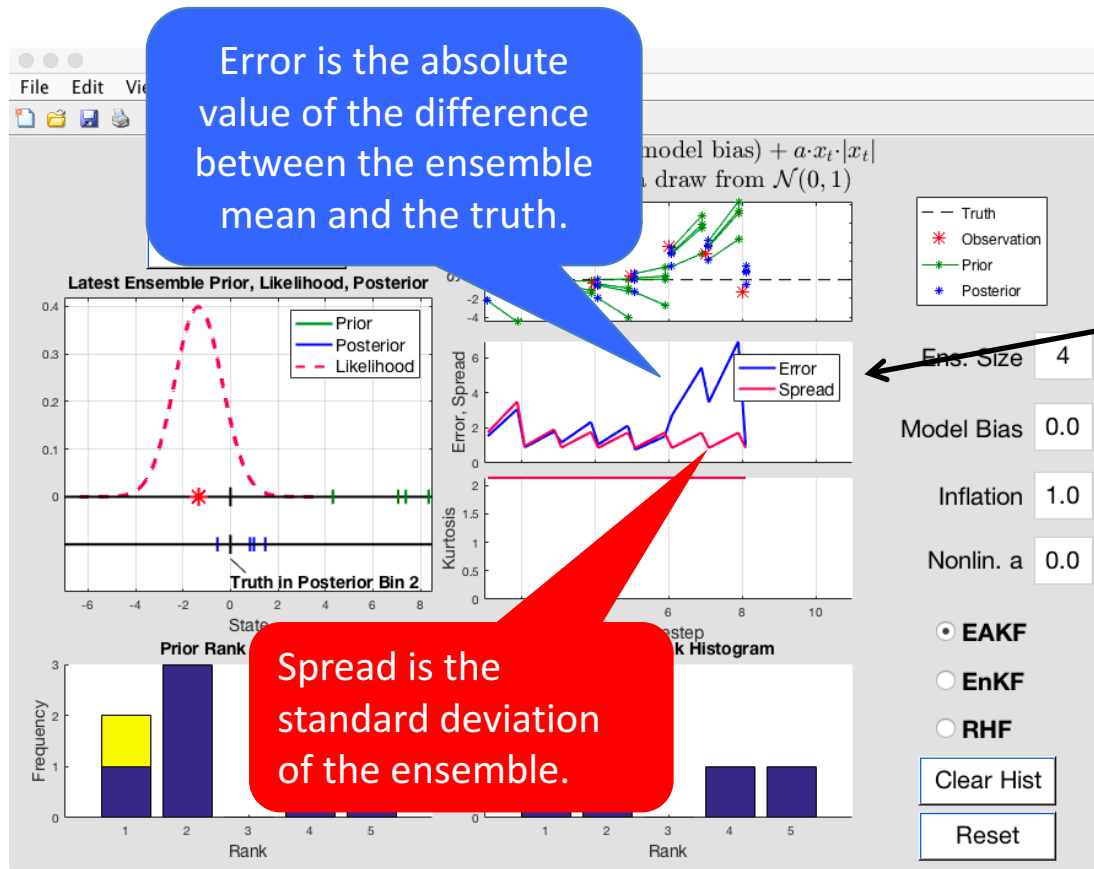


A time series of the assimilation. Line segments show forecast evolution. Most recent prior, observation, and posterior are same as in upper right window but plotted with a vertical axis.

# 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.

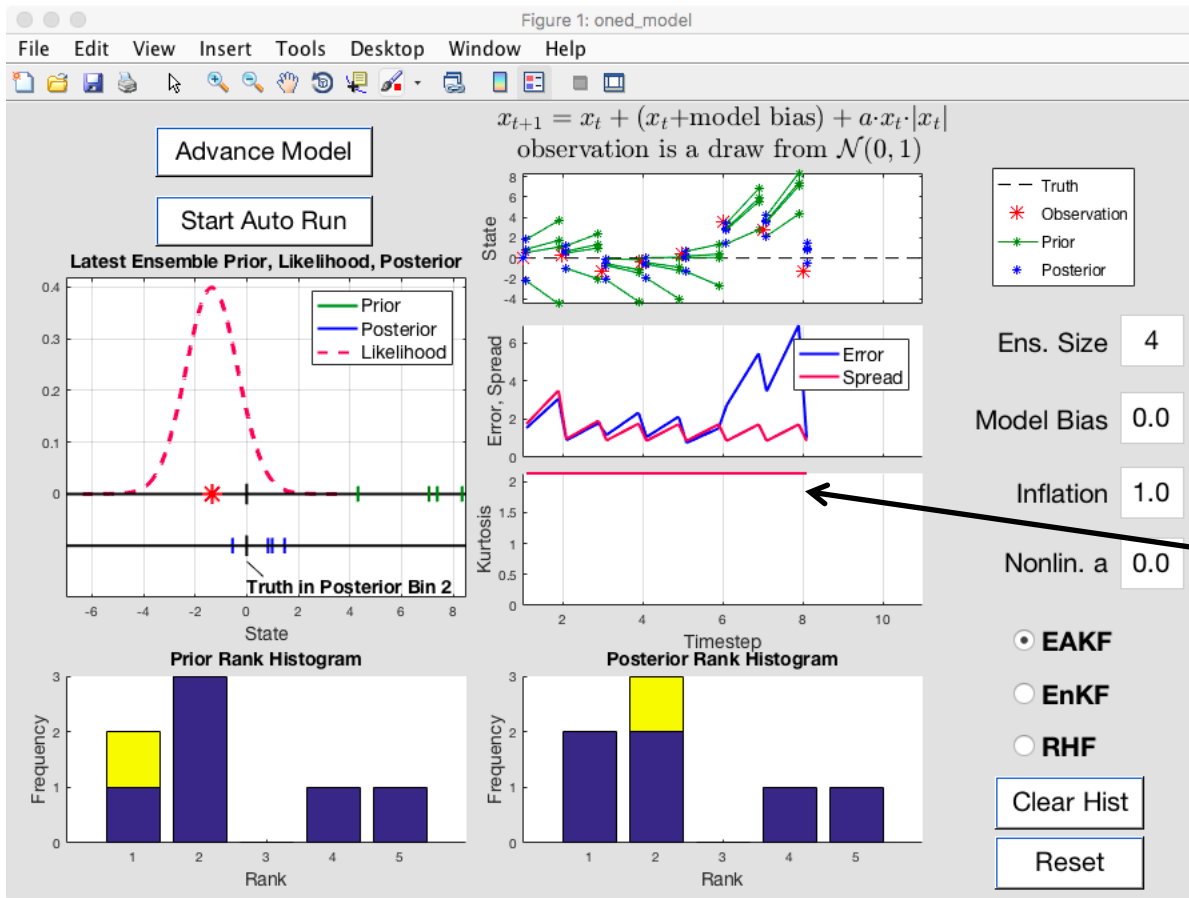


Time series of error and spread. Sawtooth pattern because prior and posterior values are shown for each time.

# 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.



Kurtosis as function of time. Stays constant here.



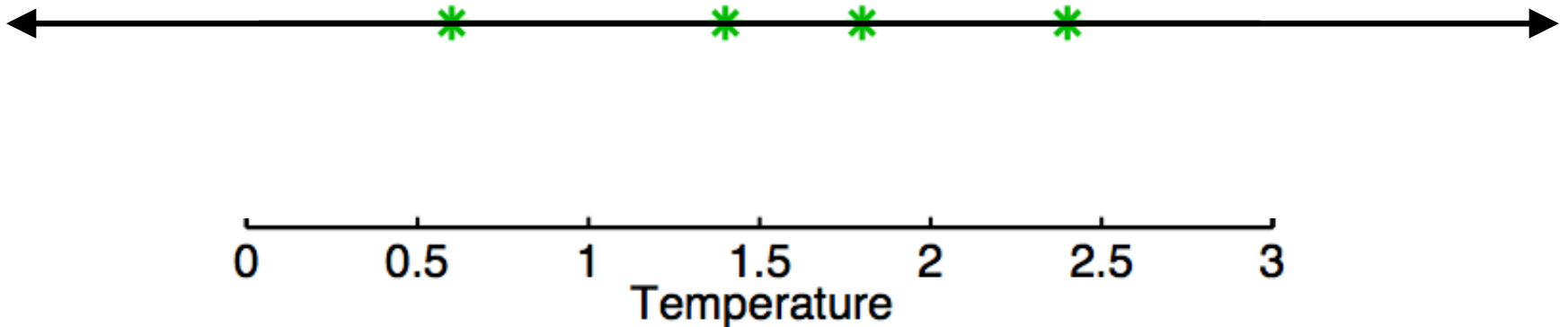
# Matlab Hands-On: oned\_model

## 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.

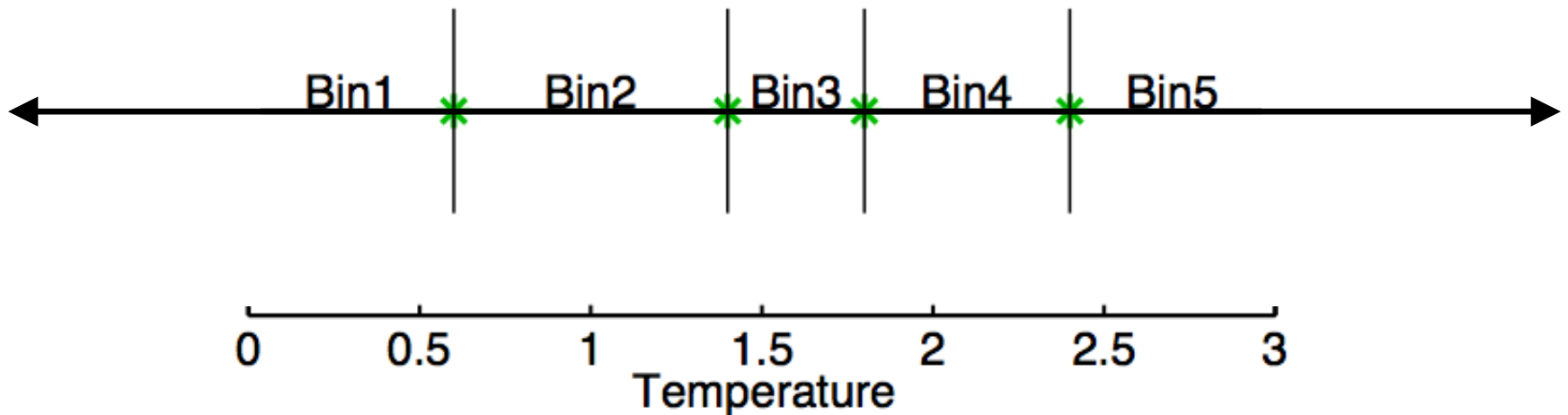
# The Rank Histogram: Evaluating Filter Performance

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



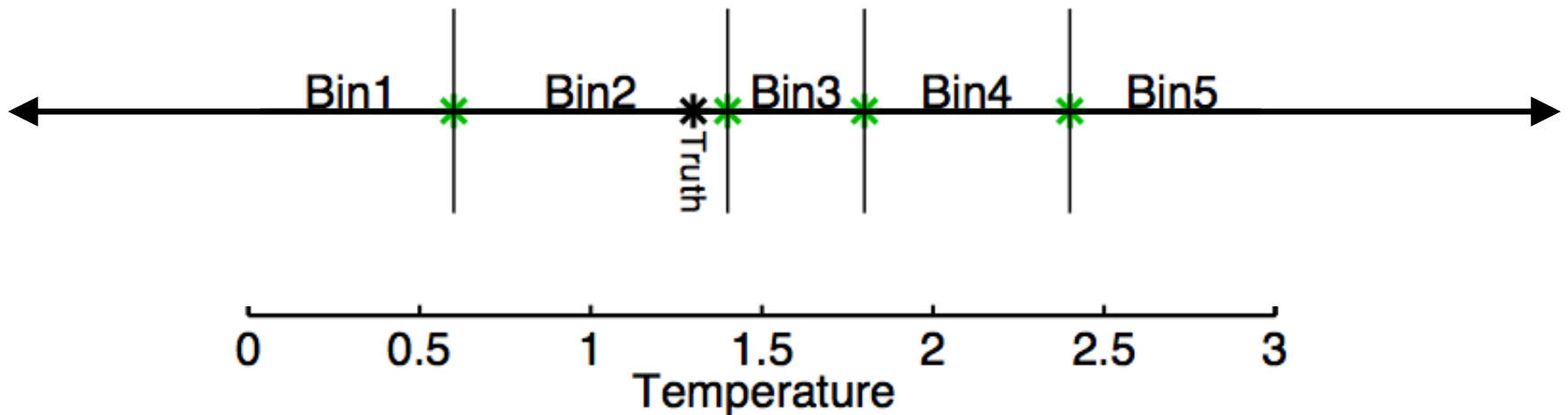
# The Rank Histogram: Evaluating Filter Performance

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



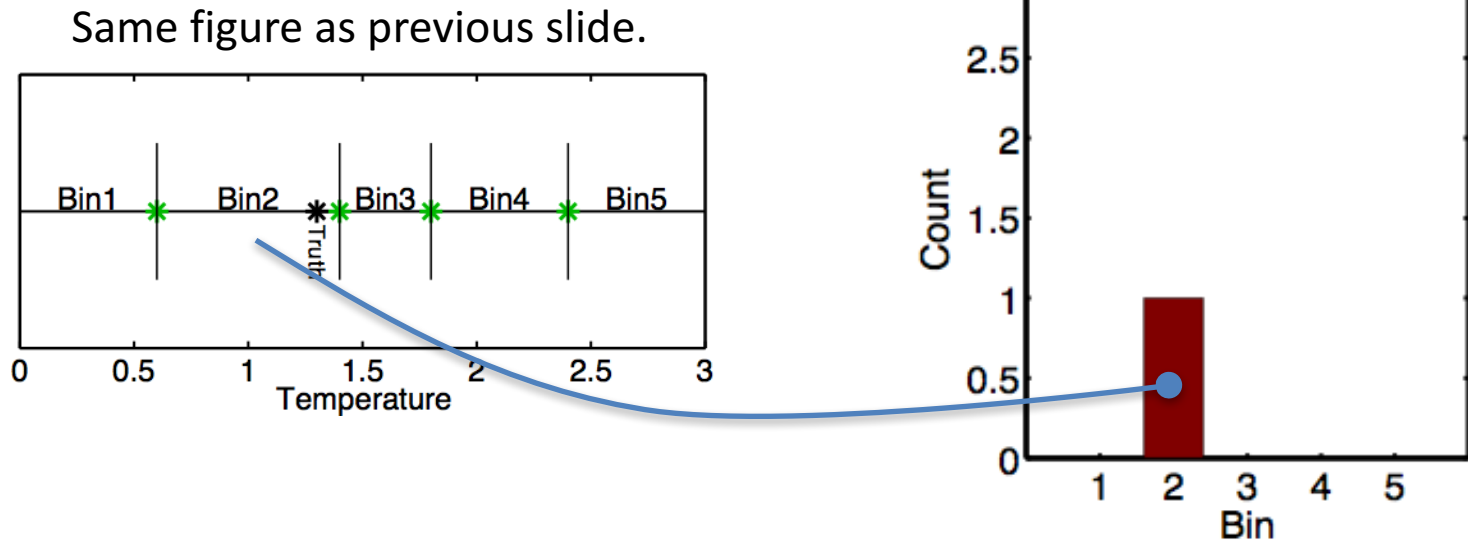
# The Rank Histogram: Evaluating Filter Performance

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



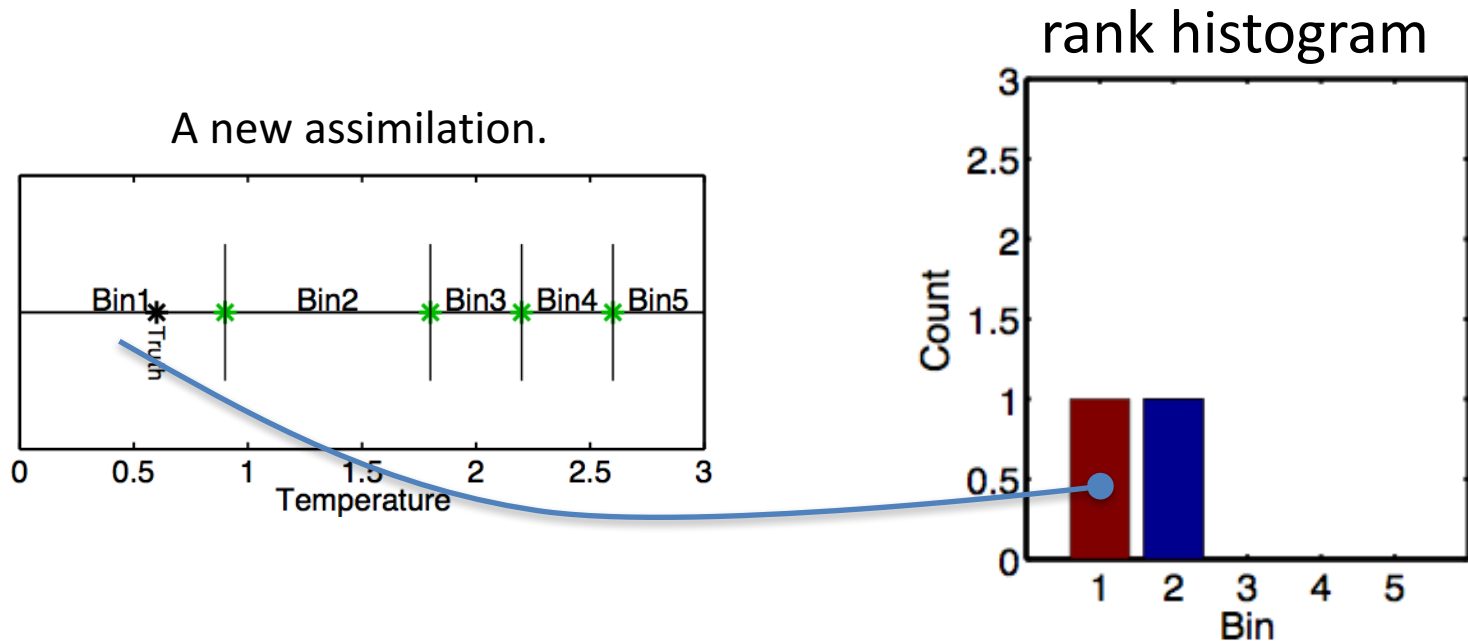
# The Rank Histogram: Evaluating Filter Performance

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



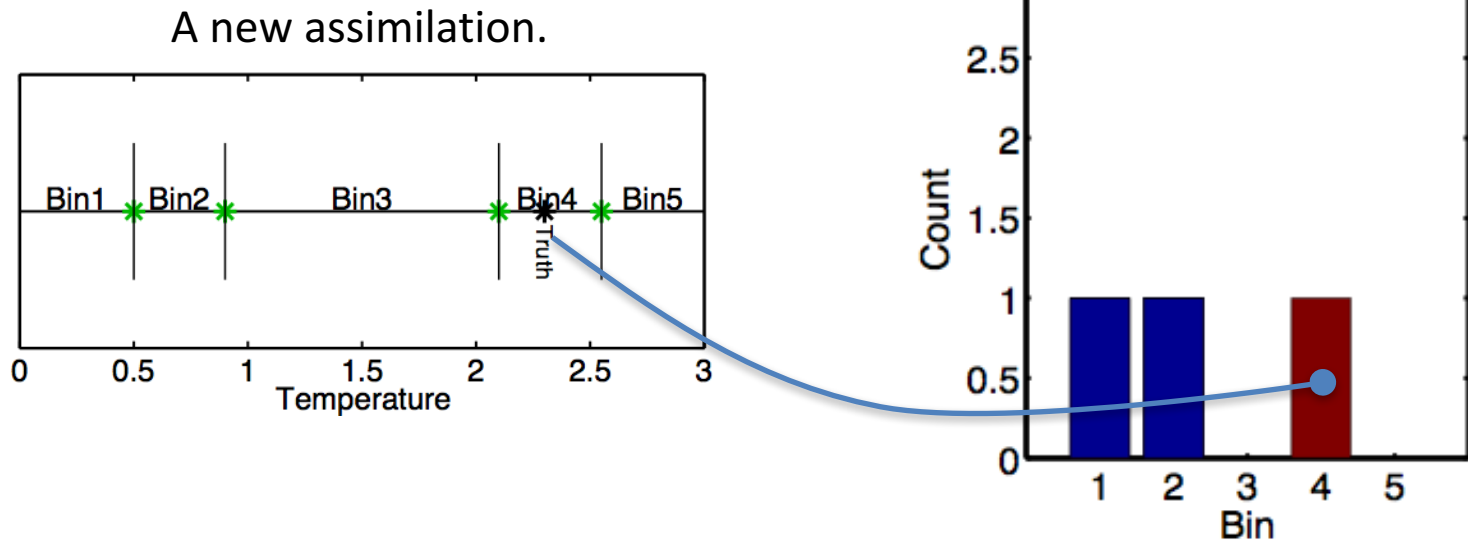
# The Rank Histogram: Evaluating Filter Performance

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



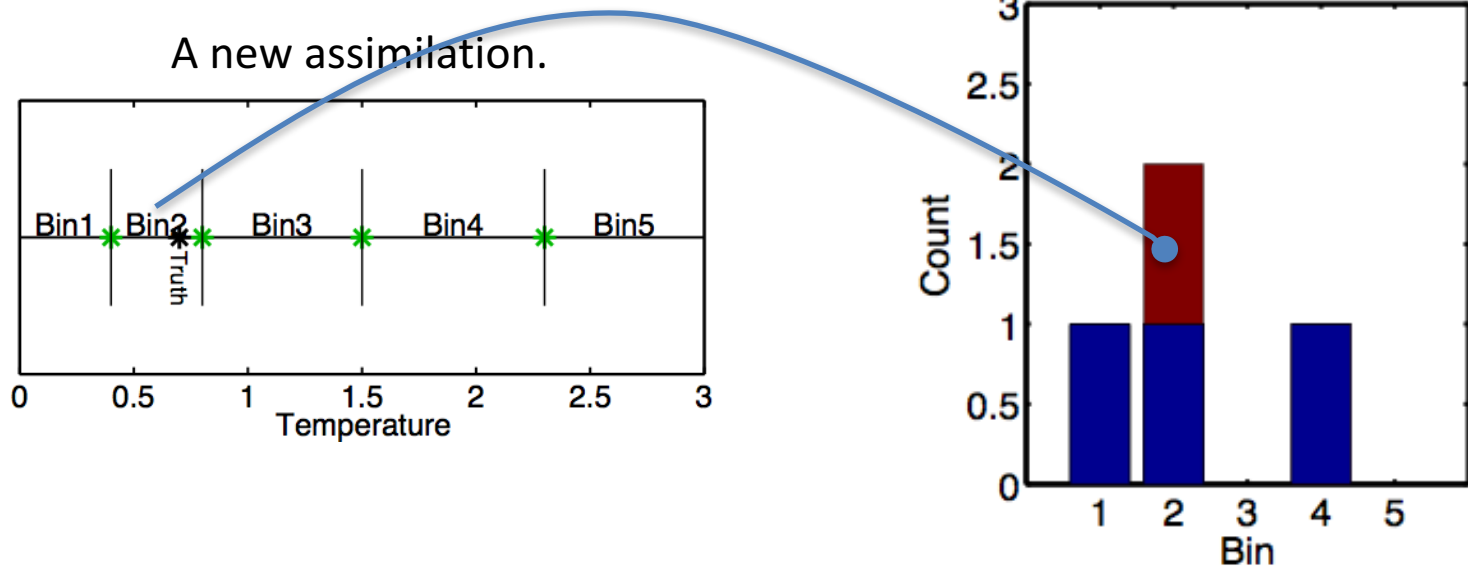
# The Rank Histogram: Evaluating Filter Performance

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.

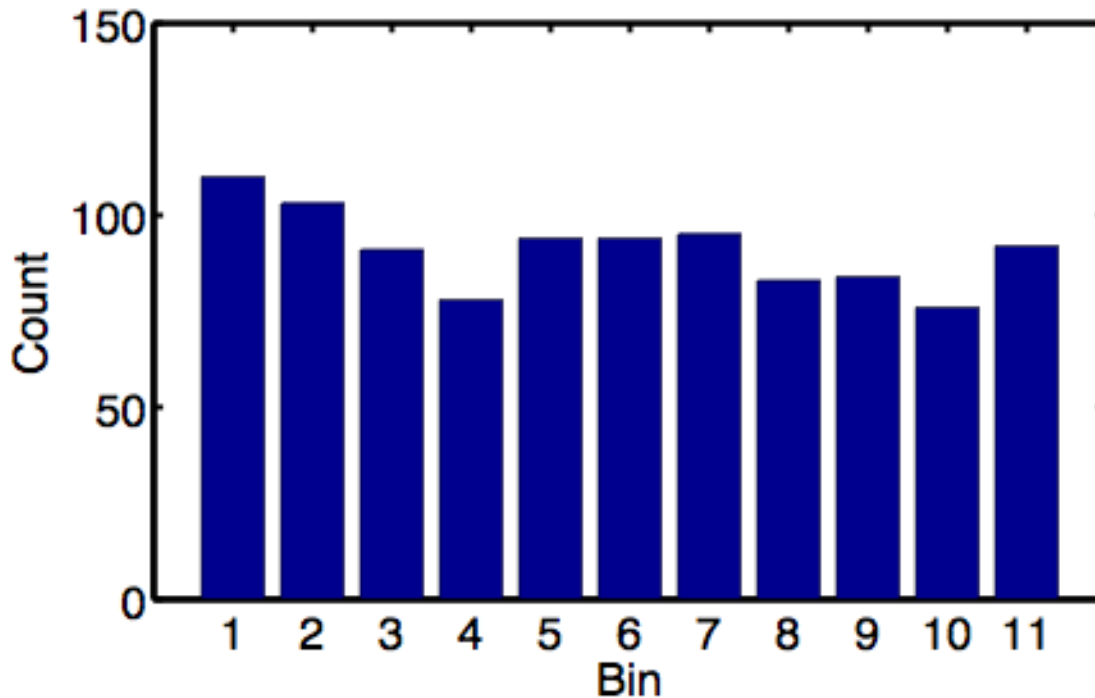




# The Rank Histogram: Evaluating Filter Performance

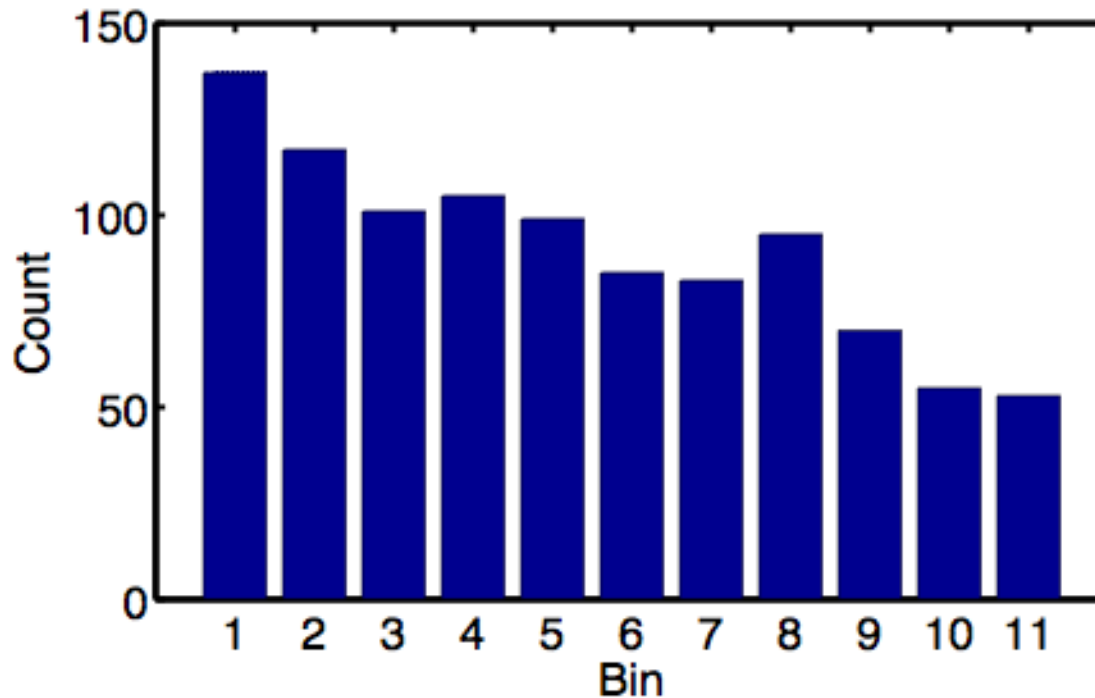
Rank histograms for good ensembles should be uniform (caveat sampling noise).

Want truth to look like random draw from ensemble.



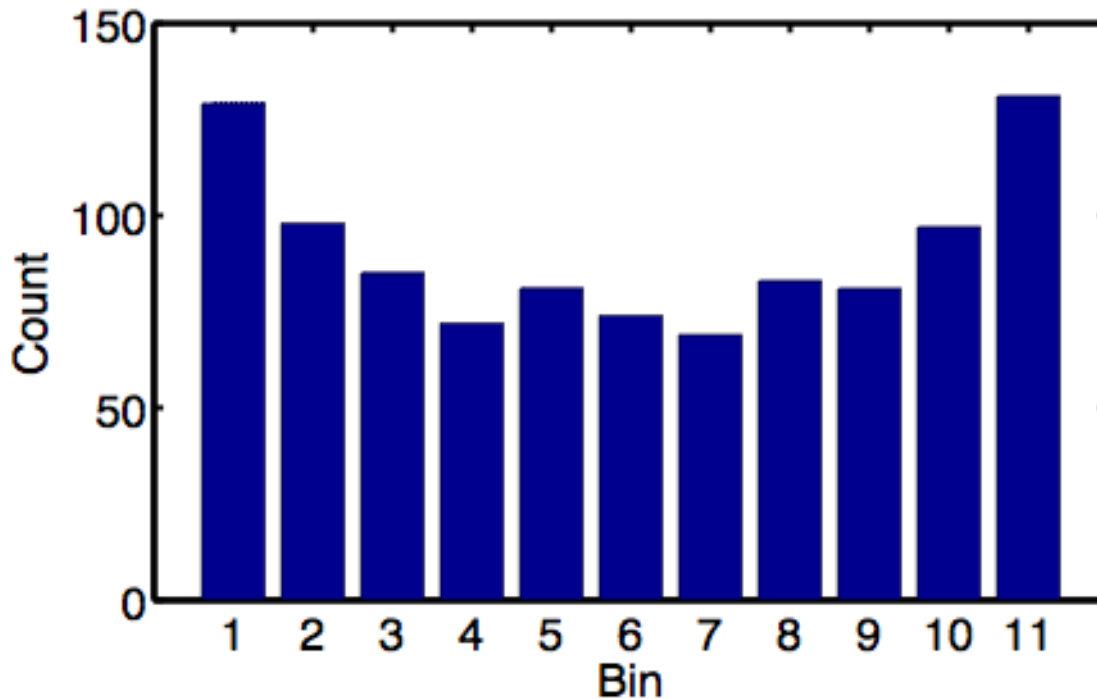
# The Rank Histogram: Evaluating Filter Performance

A biased ensemble leads to skewed histograms.



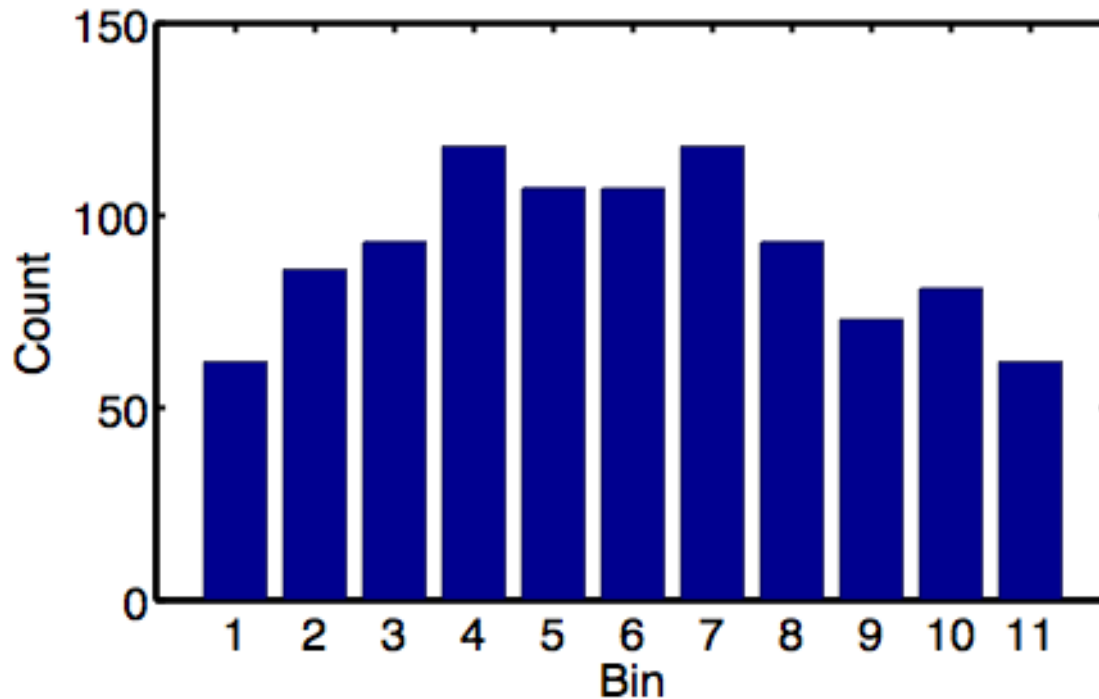
# The Rank Histogram: Evaluating Filter Performance

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



# The Rank Histogram: Evaluating Filter Performance

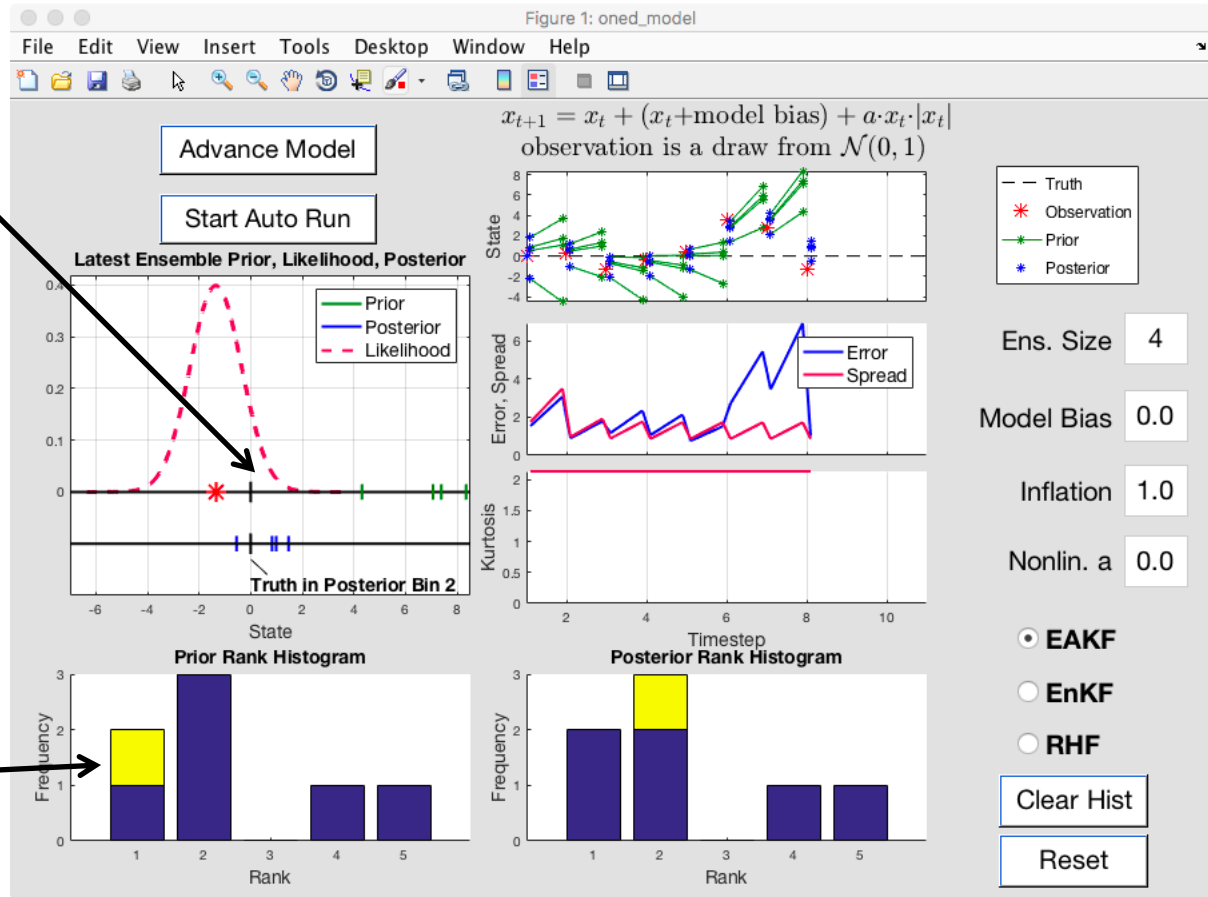
An ensemble with too much spread is peaked in the center.



# Matlab Hands-On: oned\_model

## Understanding the Rank Histogram

Truth is always 0. For this time, truth is between 1<sup>st</sup> and 2<sup>nd</sup> ensemble members; that's the second bin.



Prior (left) and Posterior (right) rank histograms.

Yellow is for current time.

# Matlab Hands-On: oned\_model

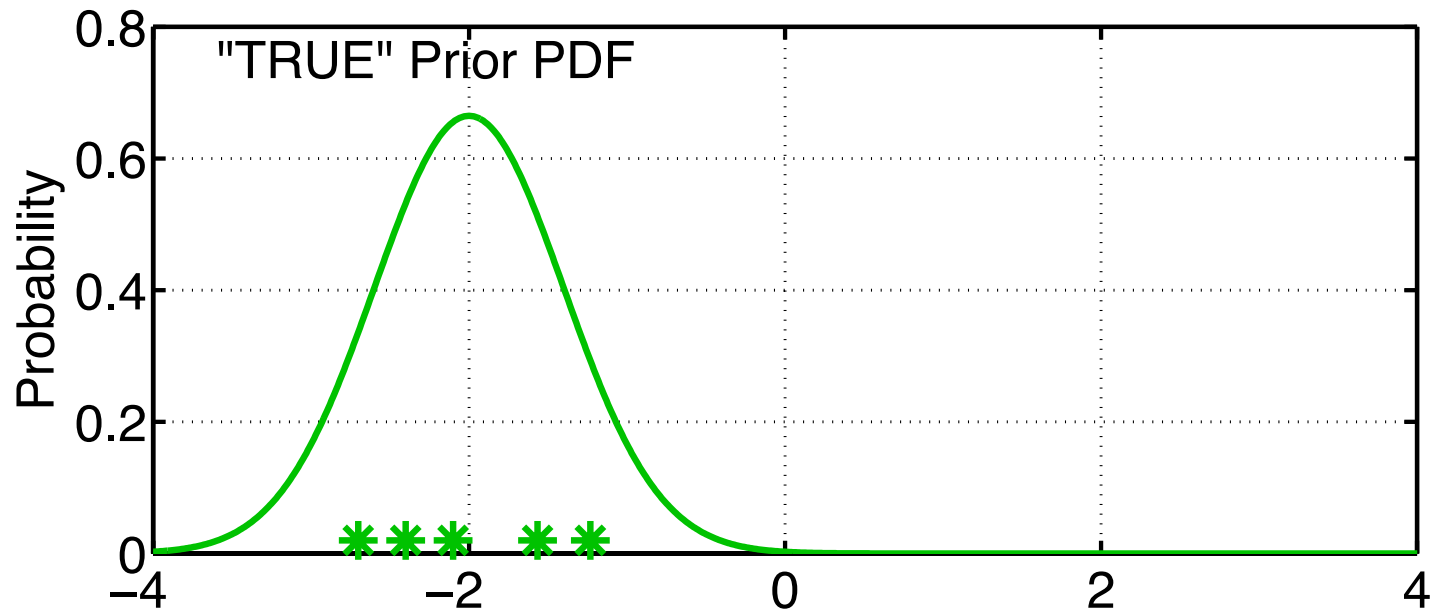
## Understanding the Rank Histogram

### 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 (  $a < 1$  ) 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?

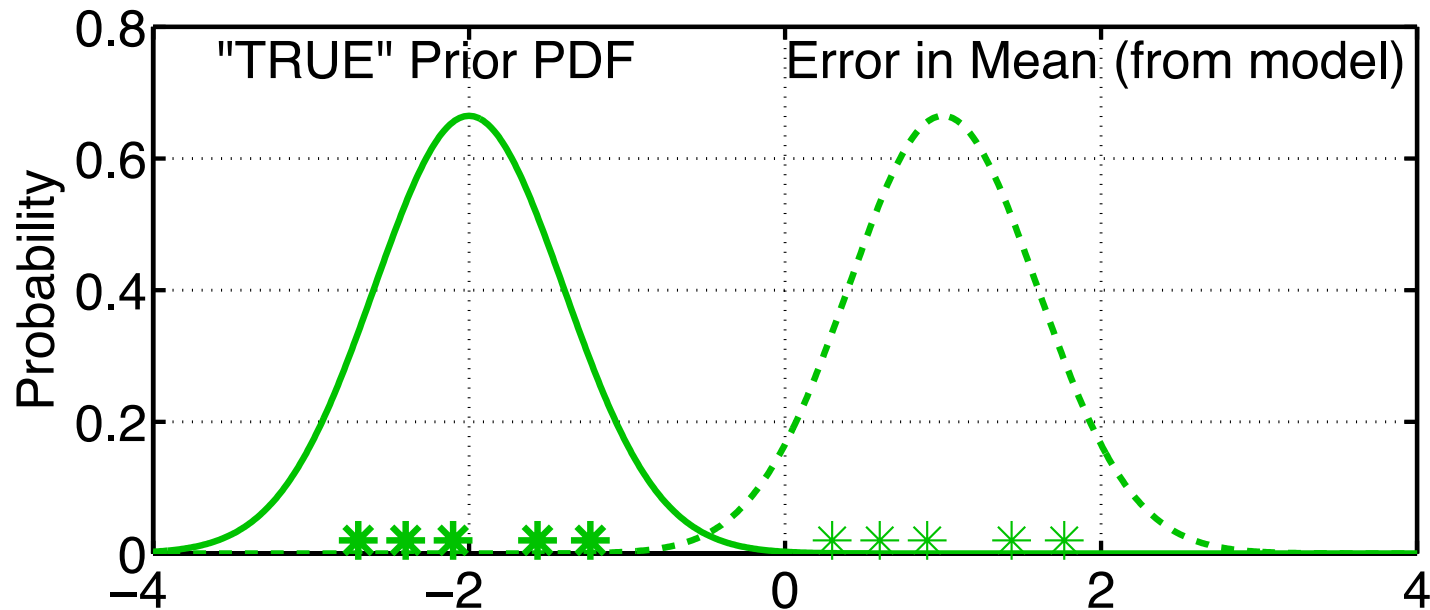
# Dealing with systematic error: Variance Inflation

Observations + physical system  $\longrightarrow$  'true' distribution.



# Dealing with systematic error: Variance Inflation

Observations + physical system  $\longrightarrow$  'true' distribution.  
Model bias (and other errors) can shift actual prior.  
Prior ensemble is too certain (needs more spread).



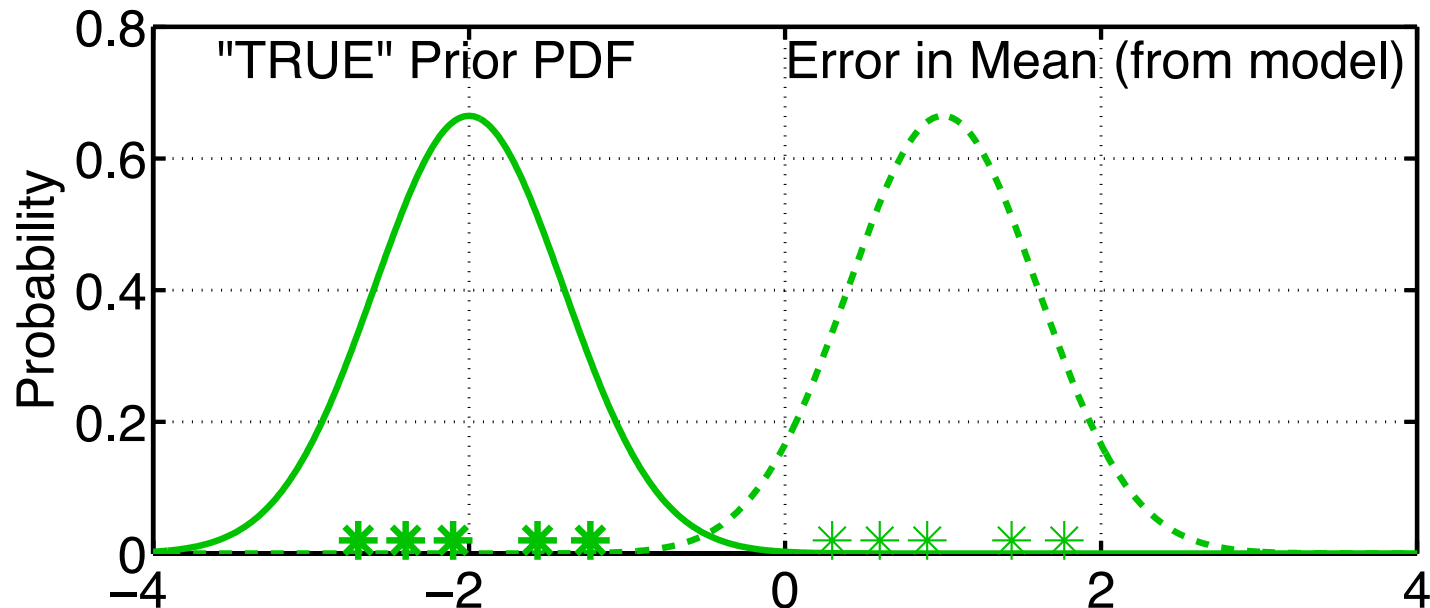


# Dealing with systematic error: Variance Inflation

Could correct error if we knew what it was.

With large models, can't know error precisely.

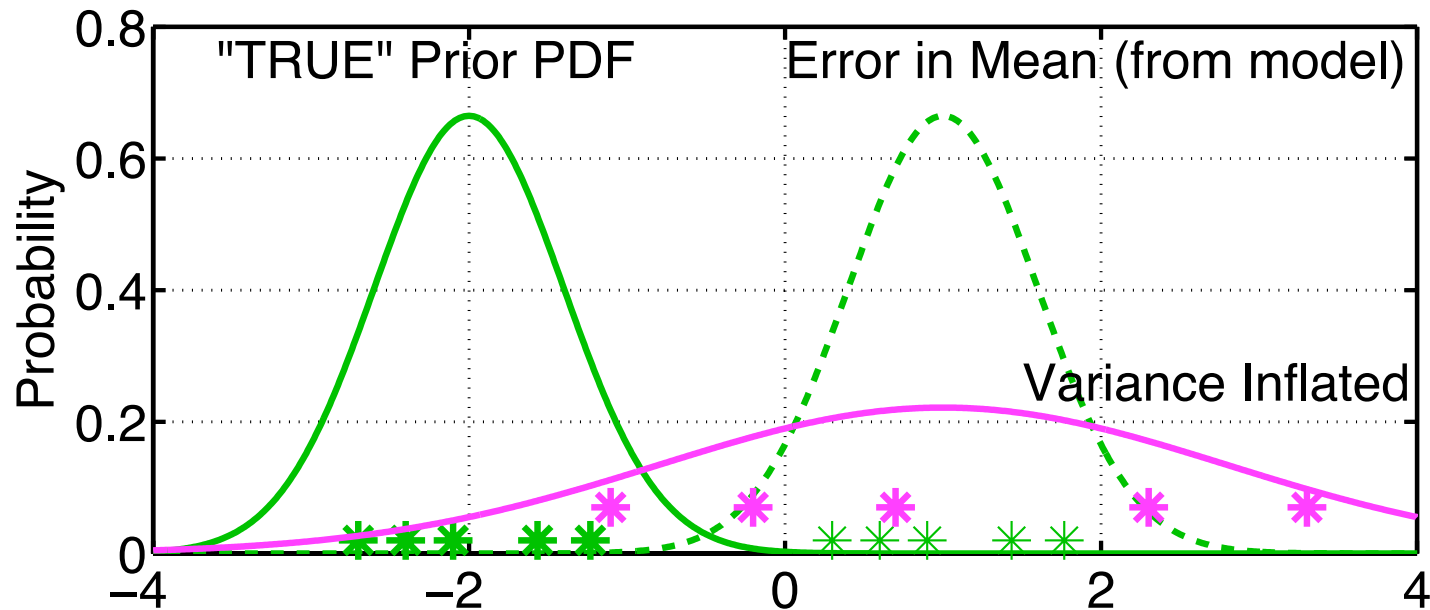
Taking no action can cause observations to be ignored.



# Dealing with systematic error: Variance Inflation

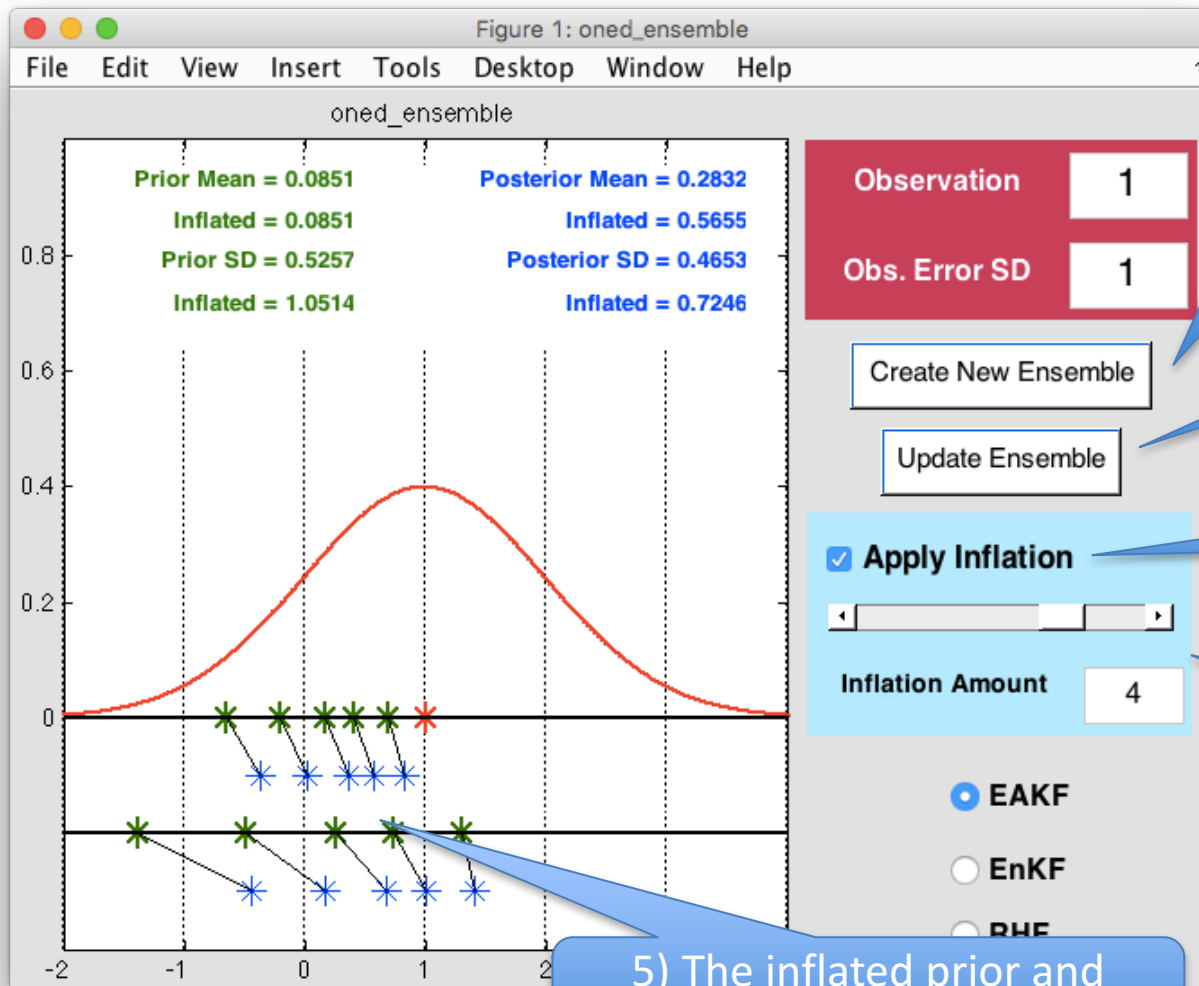
Naïve solution: increase the spread in the prior.

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



# Matlab Hands-On: oned\_ensemble

## exploring prior inflation



1) Create a new ensemble.

4) Do assimilation.

2) Turn inflation on.

3) Set an inflation value.

5) The inflated prior and posterior shows up here.

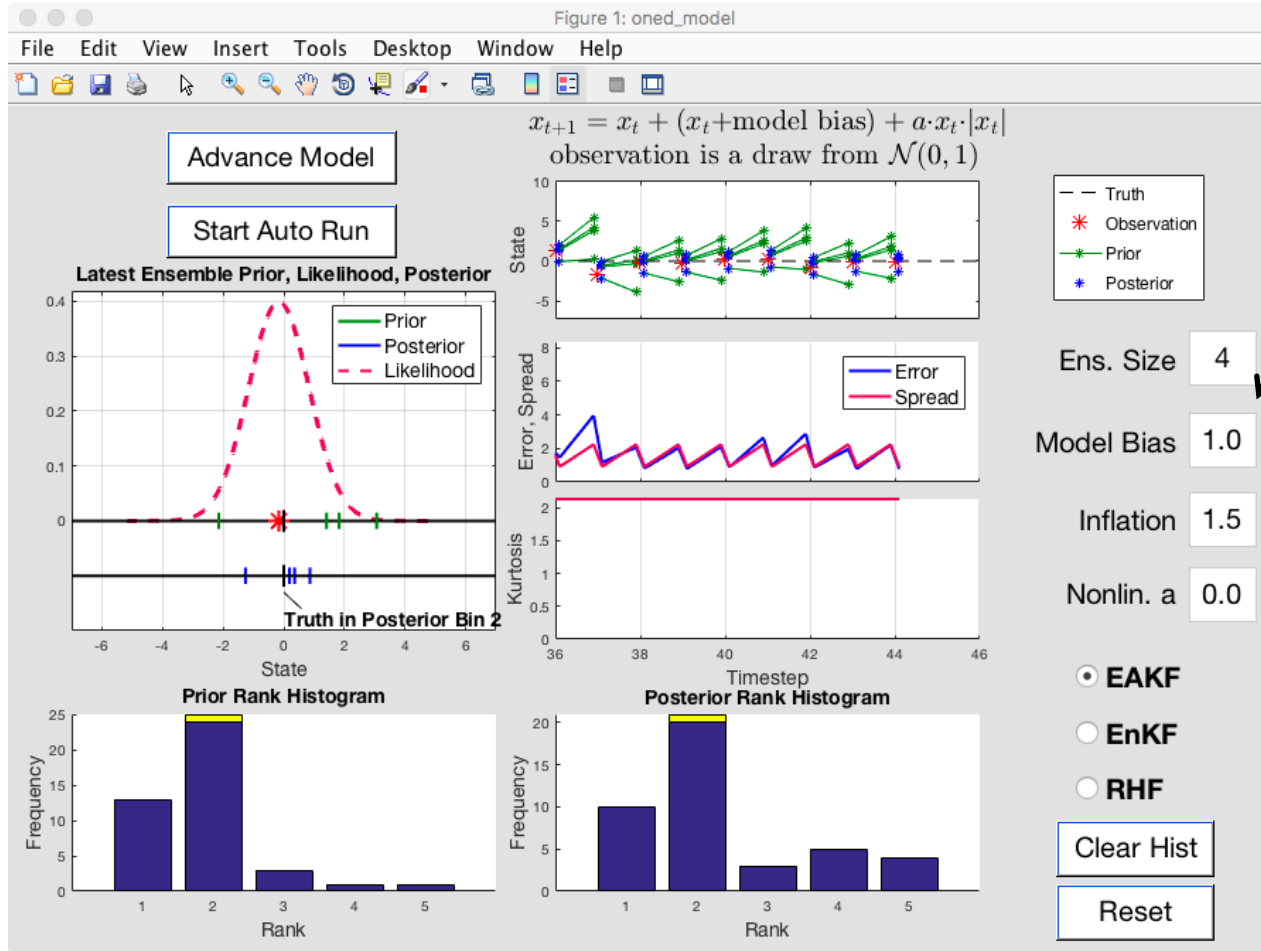
# Matlab Hands-On: exploring prior inflation with oned\_ensemble

## 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



1. Add some model bias to simulate systematic error.
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?

Note: The spread is increased by the square root of the 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.

