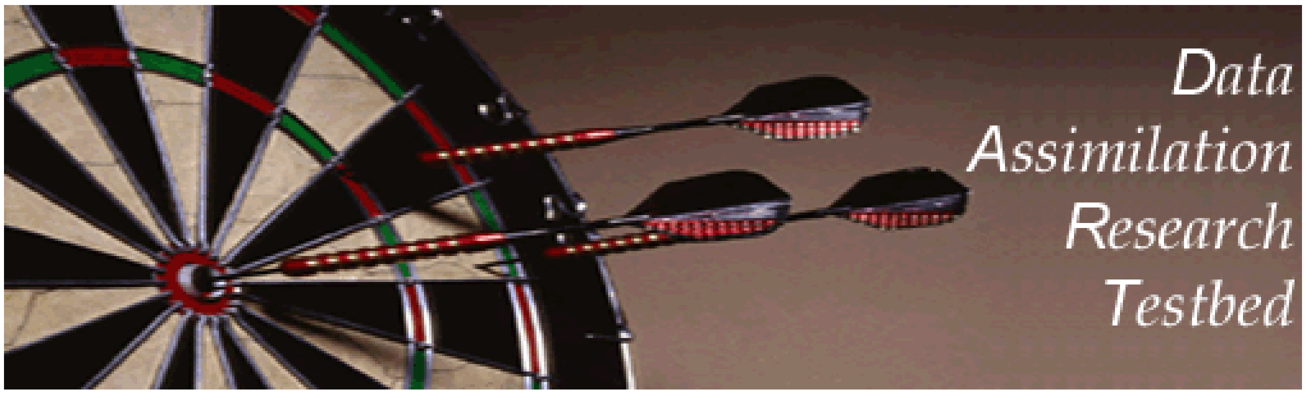
A Non-Gaussian Ensemble Filter for Nonlinear Models and Discrete Structures

Jeffrey Anderson
NCAR Data Assimilation Research Section (DAReS)



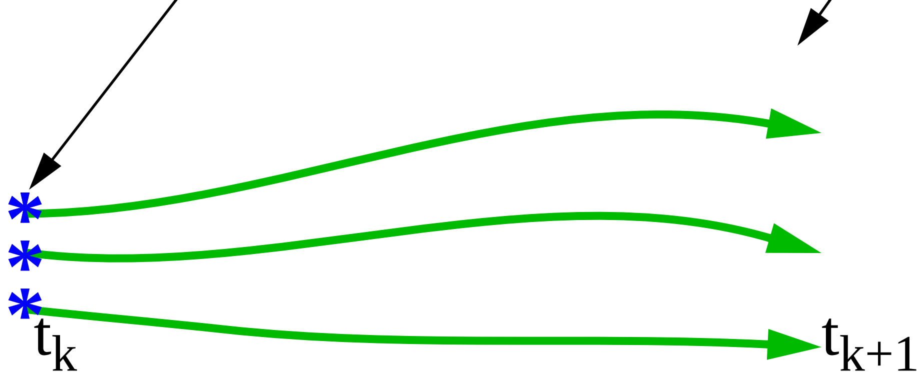
University of Maryland
25 July, 2008

Ensemble Filter Overview.

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

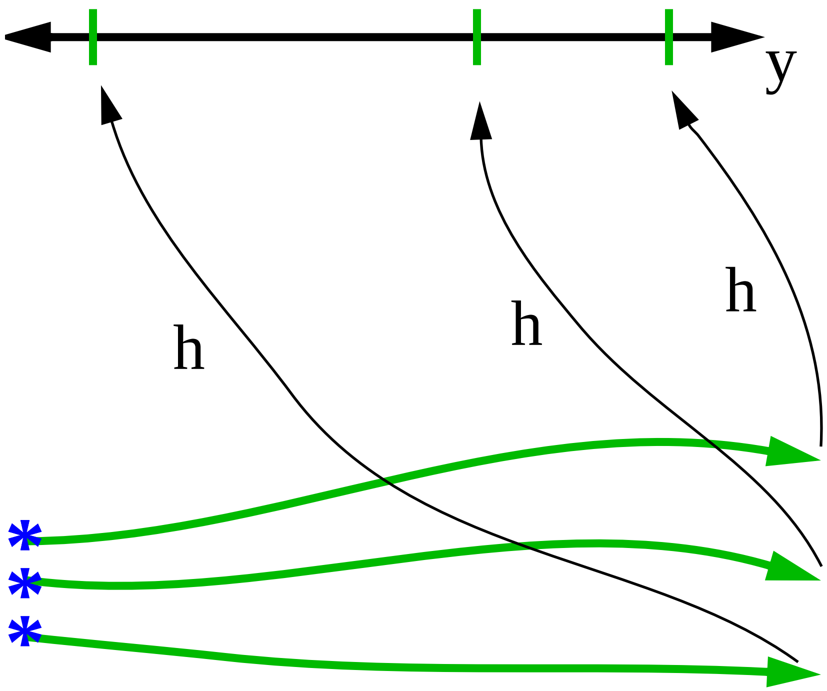
Ensemble state estimate after using previous observation (analysis).

Ensemble state at time of next observation (prior).



Ensemble Filter Overview.

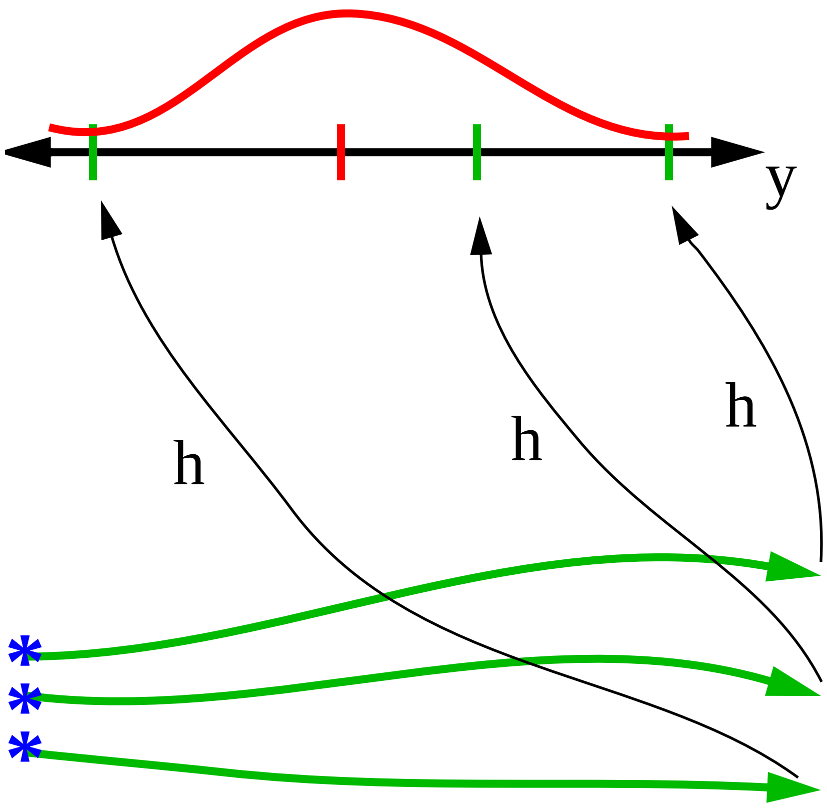
2. Get prior ensemble sample of observation, y=h(x), by applying forward operator h to each ensemble member.



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

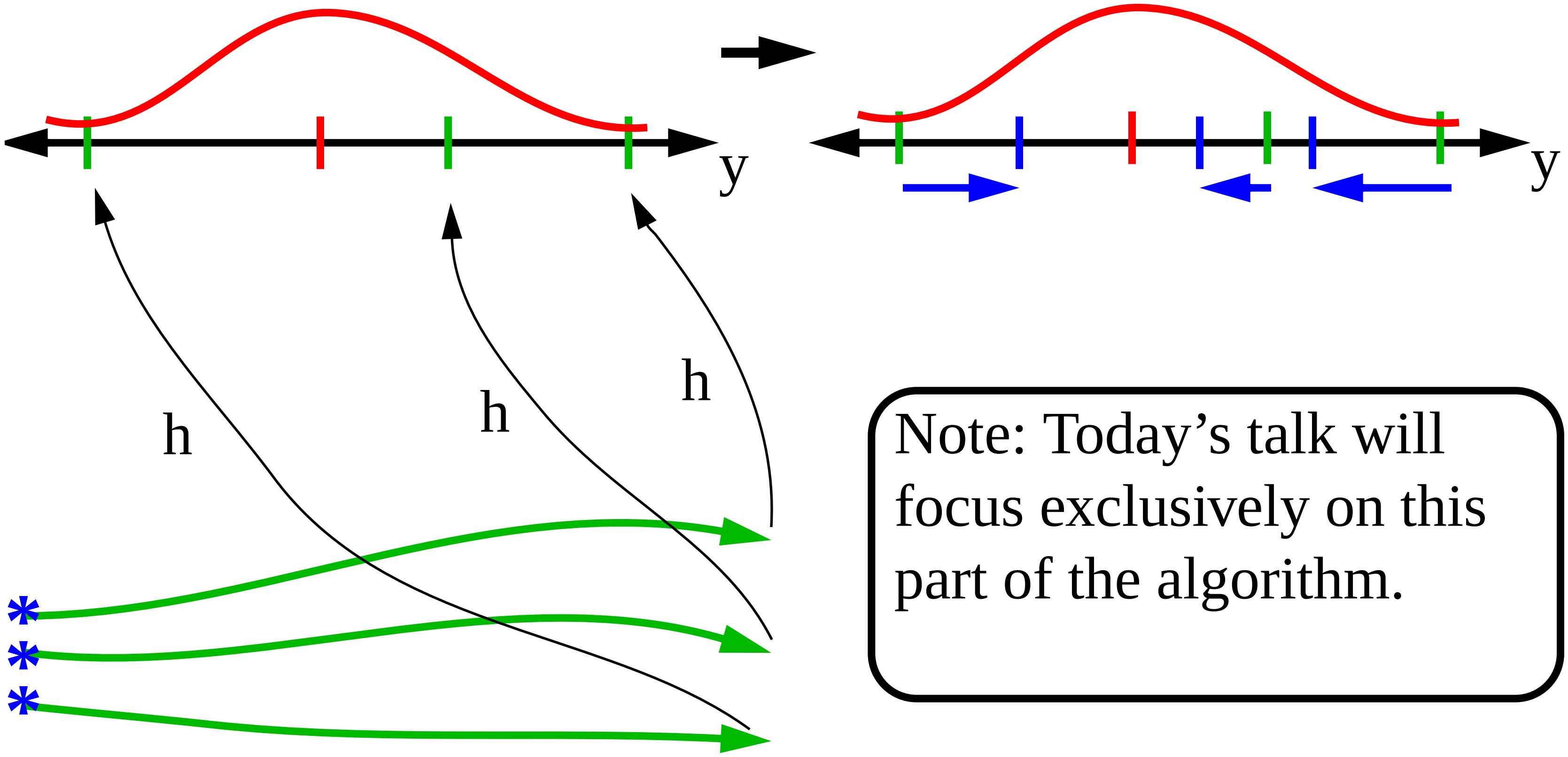
Ensemble Filter Overview.

3. Get observed value and observational error distribution from observing system.



Ensemble Filter Overview.

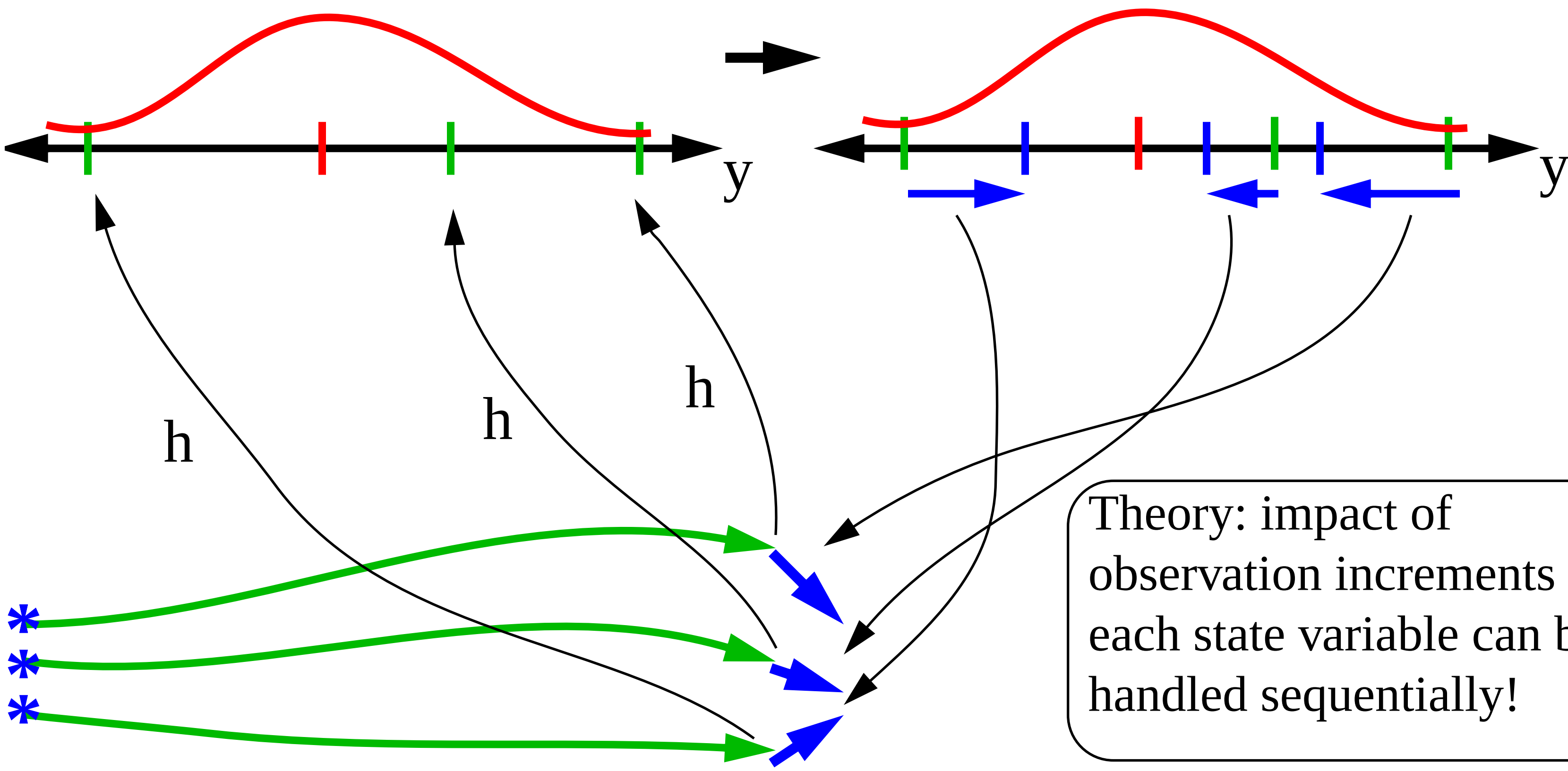
4. Find increment for each prior observation ensemble
(this is a scalar problem for uncorrelated observation errors).



Note: Today's talk will focus exclusively on this part of the algorithm.

Ensemble Filter Overview.

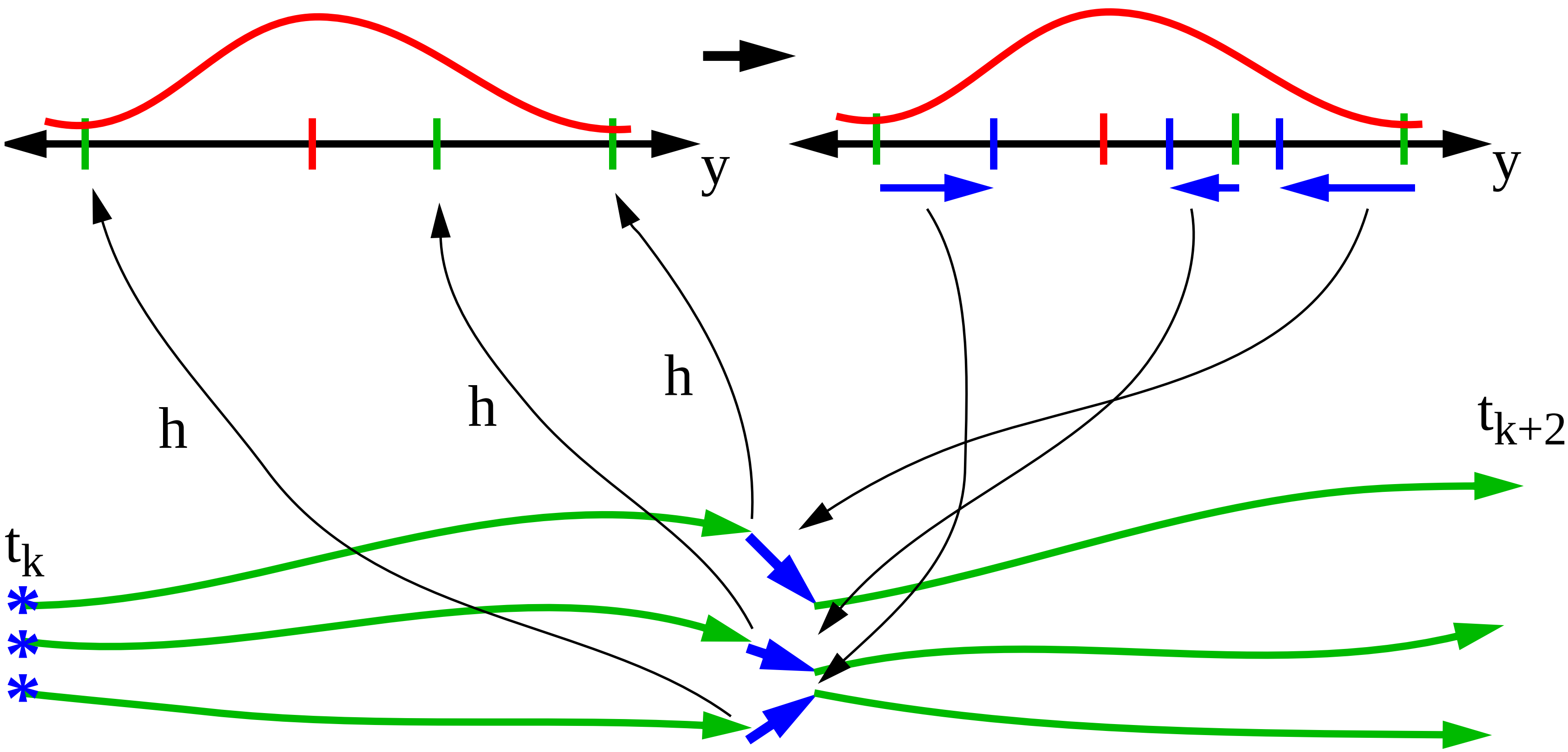
5. Use ensemble samples of y and each state variable to linearly regress observation increments onto state variable increments.



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

Ensemble Filter Overview.

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



Begin with two most common observation space update algorithms:

1. EAKF: Ensemble Adjustment KF (deterministic square root);

2. EnKF: Ensemble KF (Monte Carlo approximation).

Note: Consistent Color Scheme Throughout

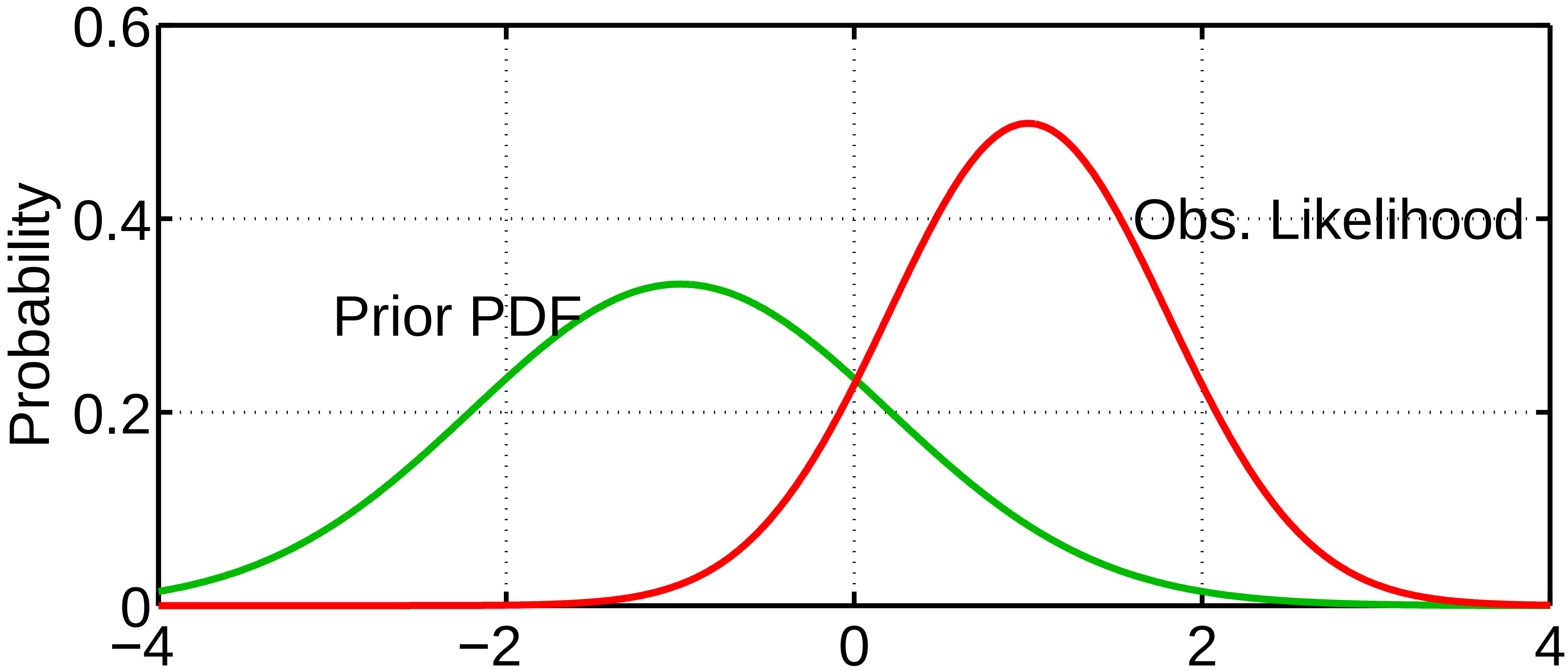
Green = Prior

Red = Observation

Blue = Posterior

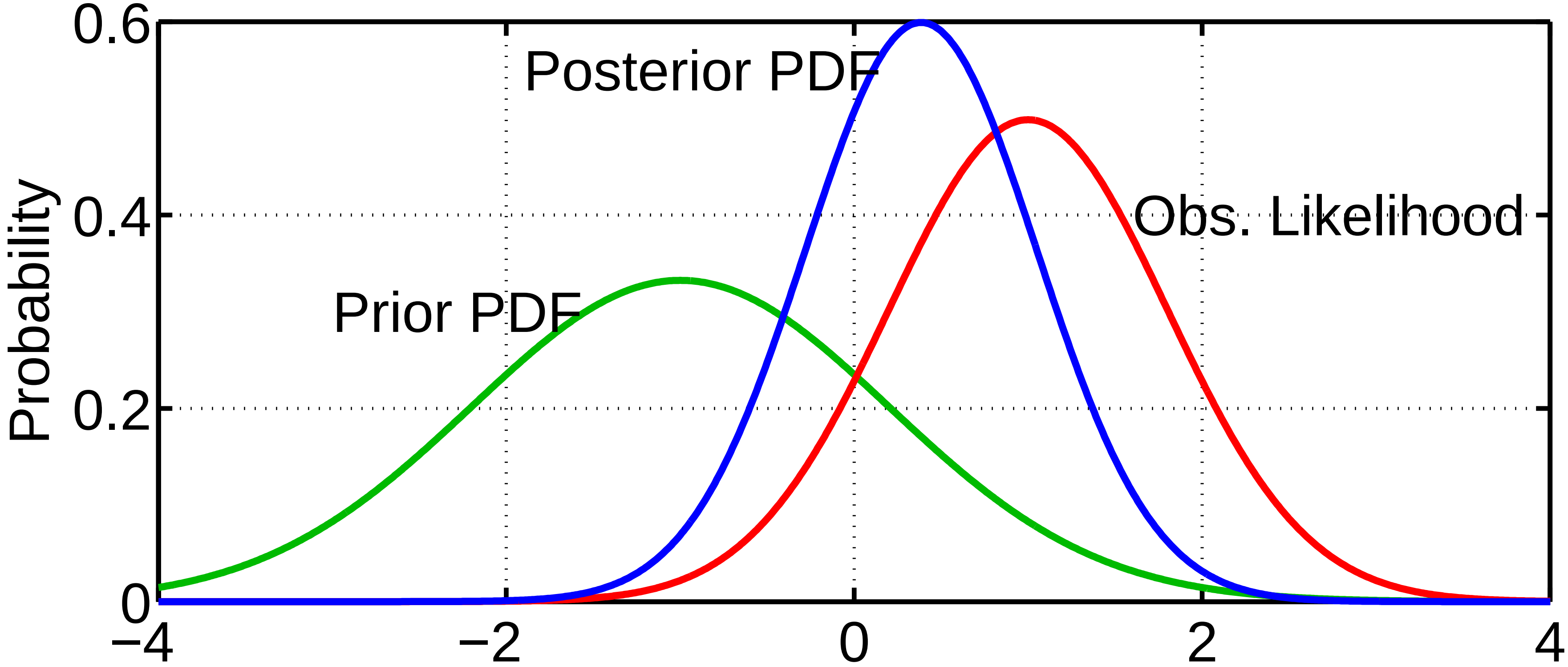
Bayes rule: $p(A|BC) = \frac{p(B|AC)p(A|C)}{p(B|C)} = \frac{p(B|AC)p(A|C)}{\int p(B|x)p(x|C)dx}$

This product is closed for Gaussian distributions.



Bayes rule: $p(A|BC) = \frac{p(B|AC)p(A|C)}{p(B|C)} = \frac{p(B|AC)p(A|C)}{\int p(B|x)p(x|C)dx}$

This product is closed for Gaussian distributions.



Product of two Gaussians:

Product of d-dimensional normals with means μ_1 and μ_2 and covariance matrices Σ_1 and Σ_2 is normal.

$$\mathbf{N}(\mu_1, \Sigma_1) \boldsymbol{N}(\mu_2, \Sigma_2) = c\mathbf{N}(\mu, \Sigma)$$

Product of two Gaussians:

Product of d-dimensional normals with means μ_1 and μ_2 and covariance matrices Σ_1 and Σ_2 is normal.

$$\mathbf{N}(\mu_1, \Sigma_1)\boldsymbol{N}(\mu_2, \Sigma_2) = c\mathbf{N}(\mu, \Sigma)$$

Covariance: $\Sigma = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$

Mean: $\mu = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1}\mu_1 + \Sigma_2^{-1}\mu_2)$

Product of two Gaussians:

Product of d-dimensional normals with means μ_1 and μ_2 and covariance matrices Σ_1 and Σ_2 is normal.

$$\mathbf{N}(\mu_1, \Sigma_1)N(\mu_2, \Sigma_2) = c\mathbf{N}(\mu, \Sigma)$$

Covariance: $\Sigma = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$

Mean: $\mu = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}(\Sigma_1^{-1}\mu_1 + \Sigma_2^{-1}\mu_2)$

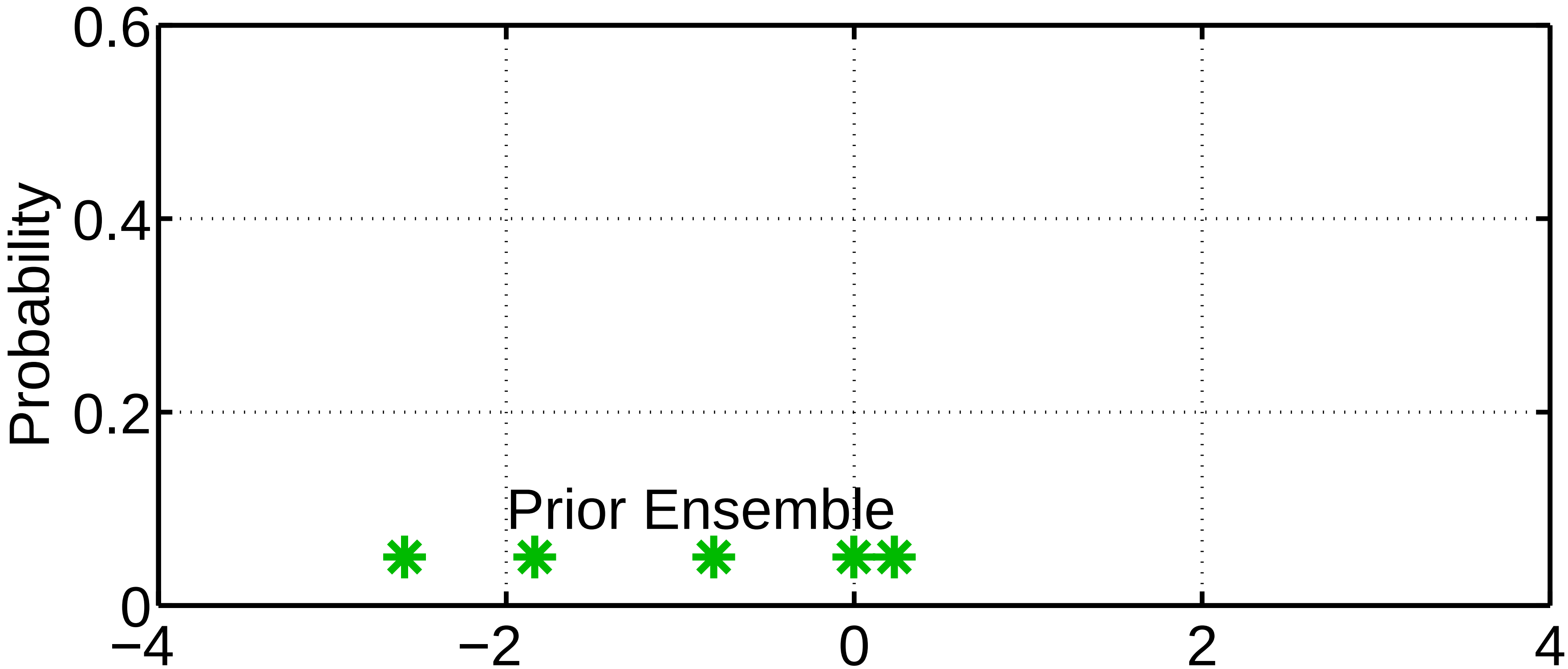
Weight: $c = \frac{1}{(2\Pi)^{d/2}|\Sigma_1 + \Sigma_2|^{1/2}} \exp\left\{-\frac{1}{2}[(\mu_2 - \mu_1)^T(\Sigma_1 + \Sigma_2)^{-1}(\mu_2 - \mu_1)]\right\}$

Ignore the weight for now; normalize products to be PDFs.

But it is used in the new algorithm...

Bayes rule: $p(A|BC) = \dfrac{p(B|AC)p(A|C)}{p(B|C)} = \dfrac{p(B|AC)p(A|C)}{\int p(B|x)p(x|C)dx}$

Ensemble filters: <u>Prior is available as finite sample.</u>

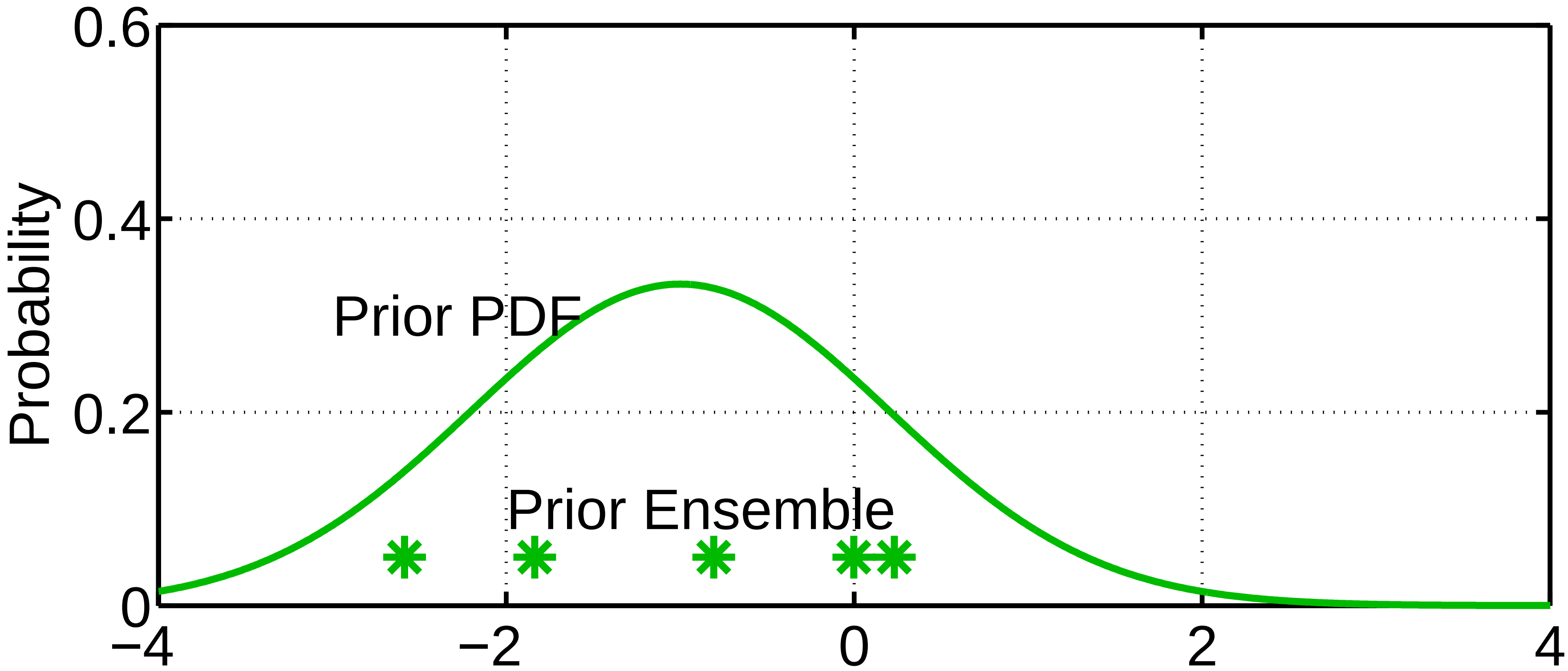


Don't know much about properties of this sample.

May naively assume it is random draw from 'truth'.

Bayes rule: $p(A|BC) = \frac{p(B|AC)p(A|C)}{p(B|C)} = \frac{p(B|AC)p(A|C)}{\int p(B|x)p(x|C)dx}$

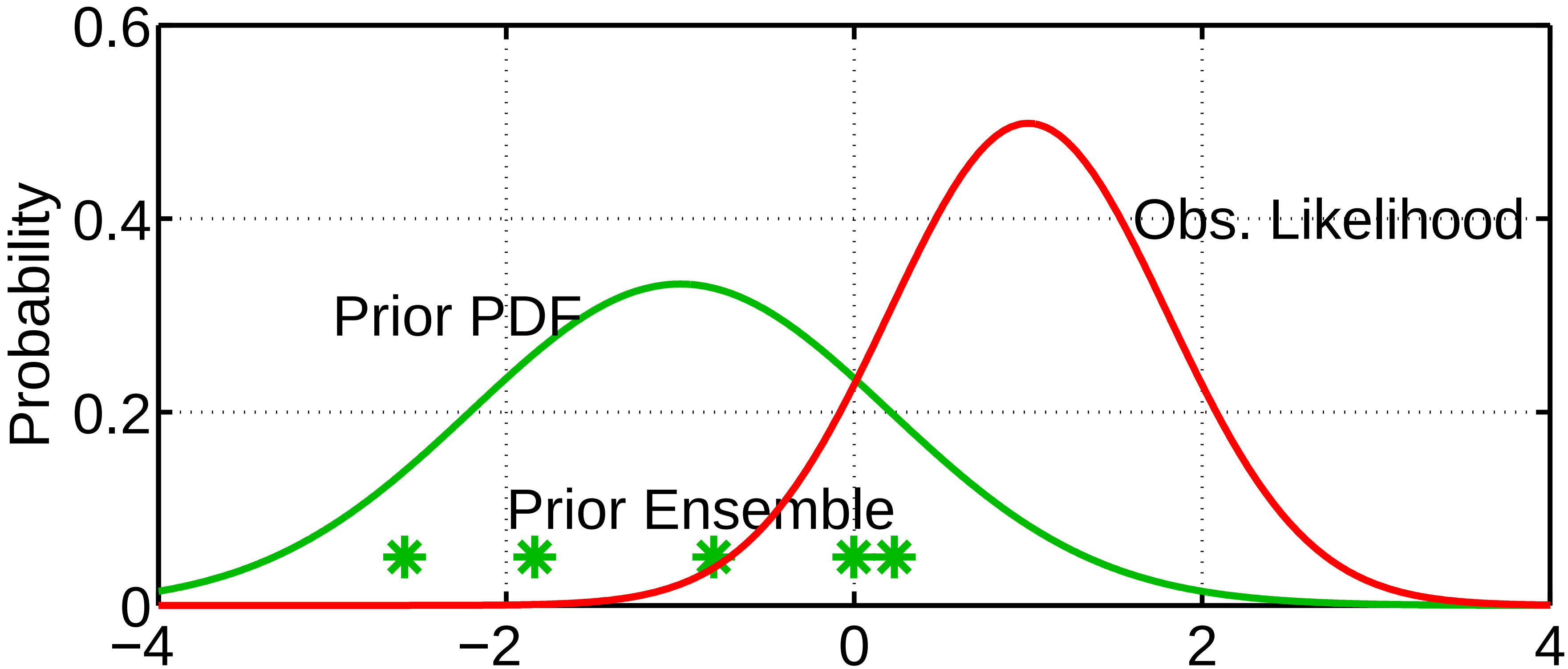
How can we take product of sample with continuous likelihood?



Fit a continuous (Gaussian for now) distribution to sample.

Bayes rule: $p(A|BC) = \dfrac{p(B|AC)p(A|C)}{p(B|C)} = \dfrac{p(B|AC)p(A|C)}{\int p(B|x)p(x|C)dx}$

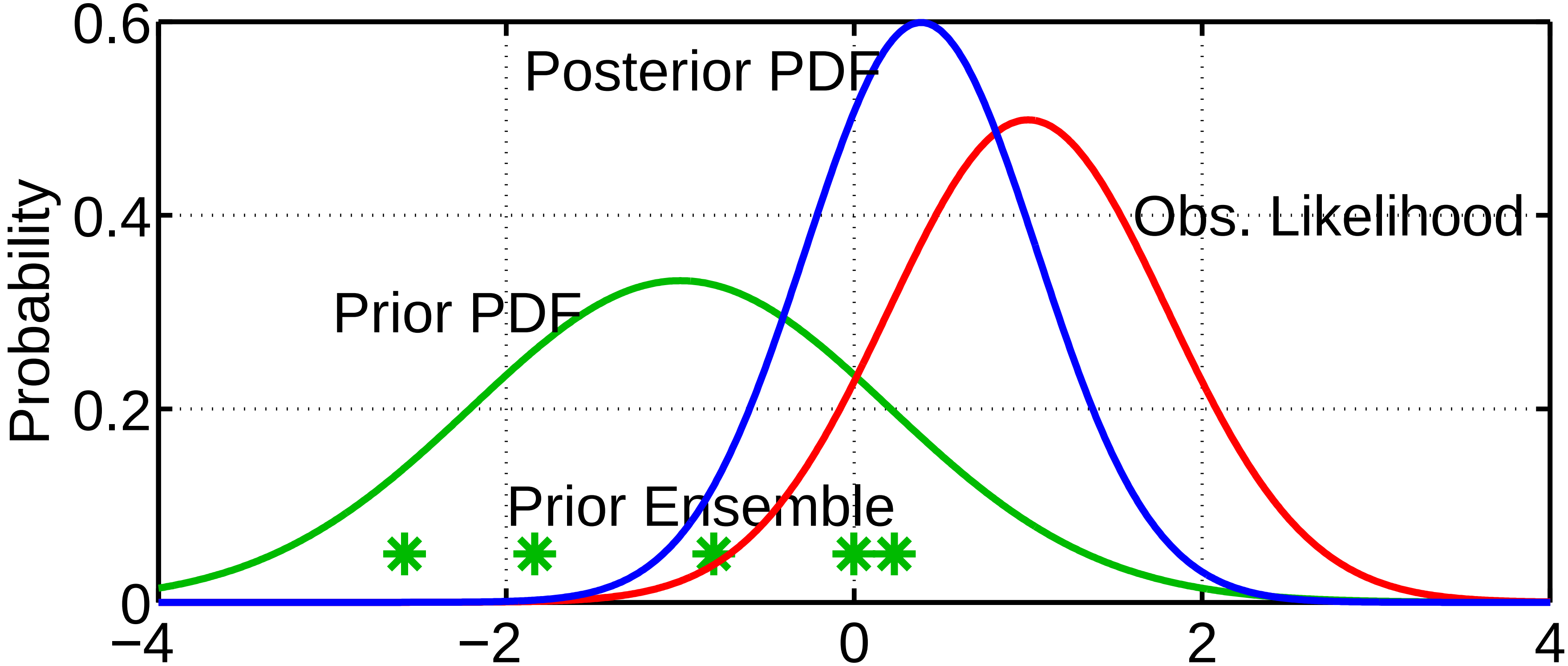
Observation likelihood usually continuous (nearly always Gaussian).



If Obs. Likelihood isn't Gaussian, can generalize methods below.
For instance, can fit set of Gaussian kernels to obs. likelihood.

Bayes rule: $p(A|BC) = \dfrac{p(B|AC)p(A|C)}{p(B|C)} = \dfrac{p(B|AC)p(A|C)}{\int p(B|x)p(x|C)dx}$

Product of prior Gaussian fit and Obs. likelihood is Gaussian.

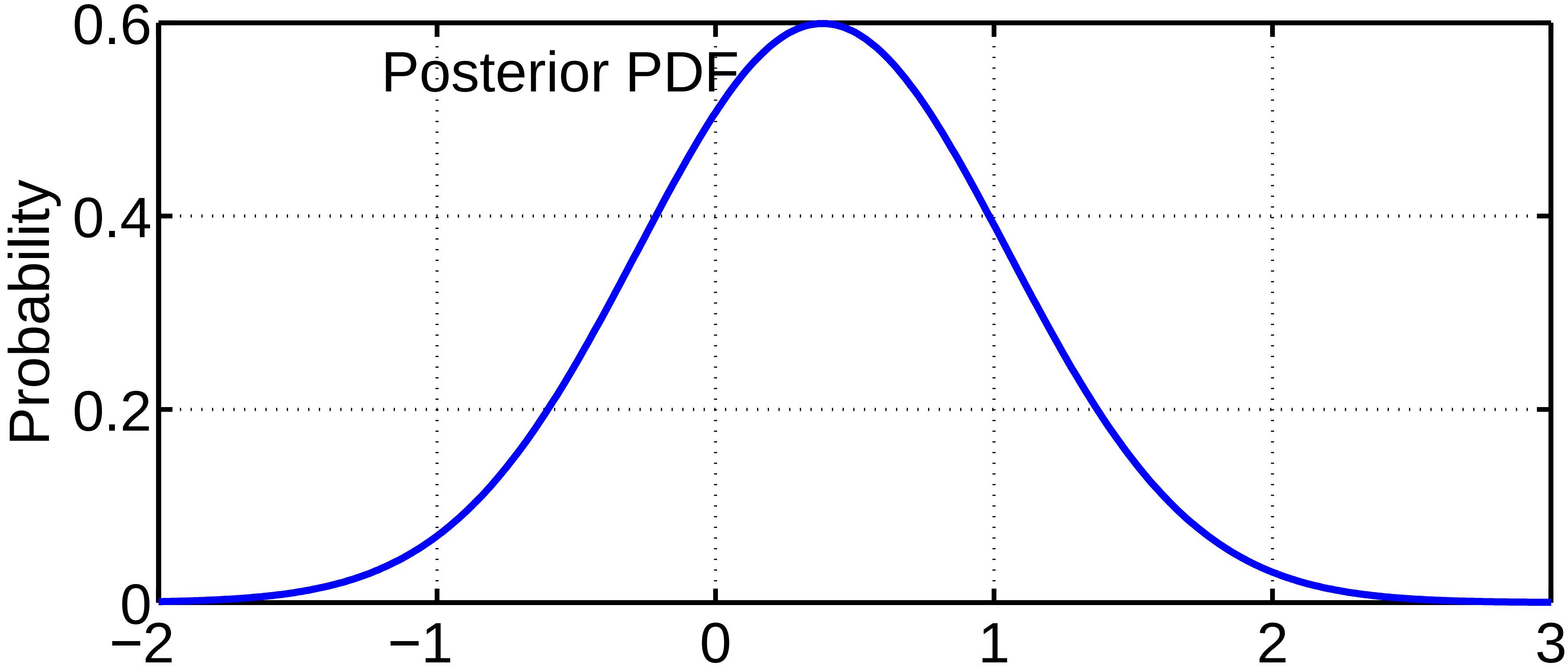


Computing continuous posterior is simple.

BUT, need to have a SAMPLE of this PDF.

Sampling Posterior PDF:

There are many ways to do this.



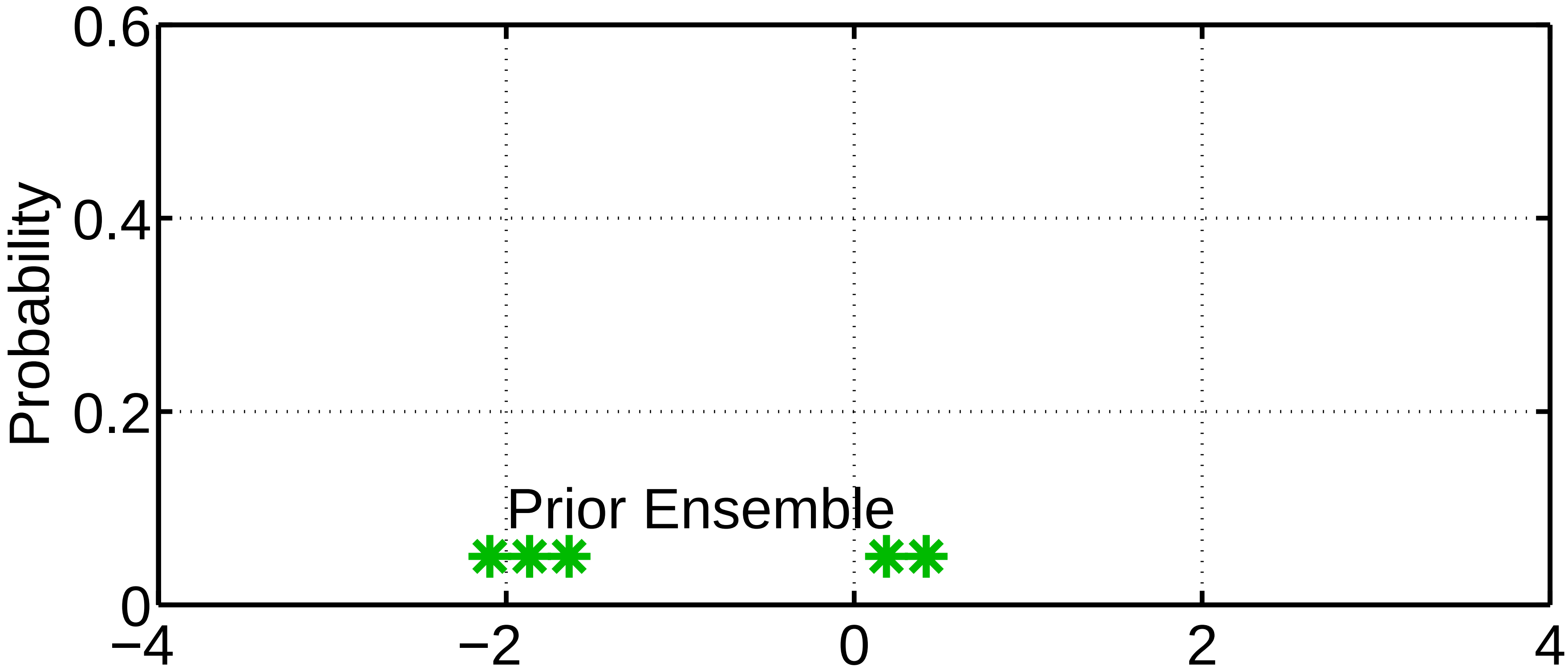
Exact properties of different methods may be unclear.

Trial and error still best way to see how they perform.

Will interact with properties of prediction models, etc.

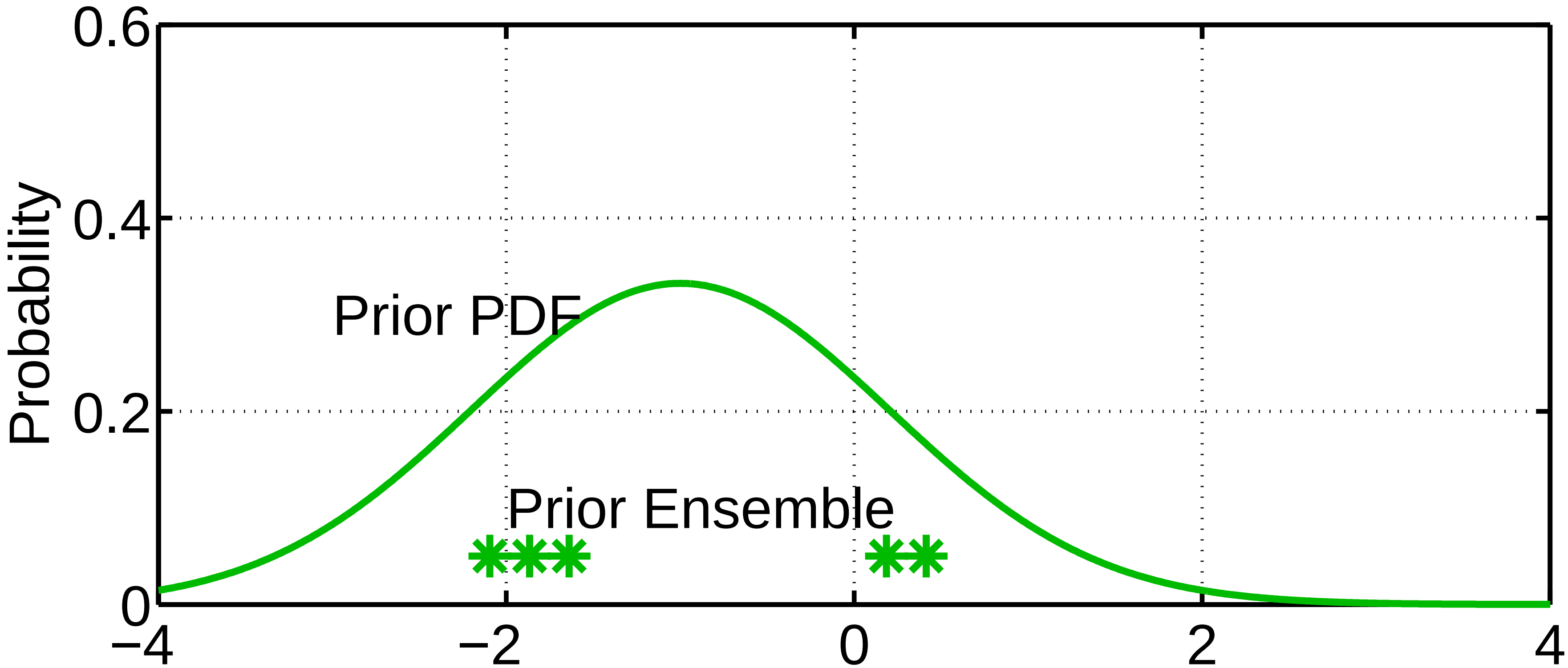
Ensemble Filter Algorithms:

Ensemble Adjustment Filter (a deterministic square root filter).



Ensemble Filter Algorithms:

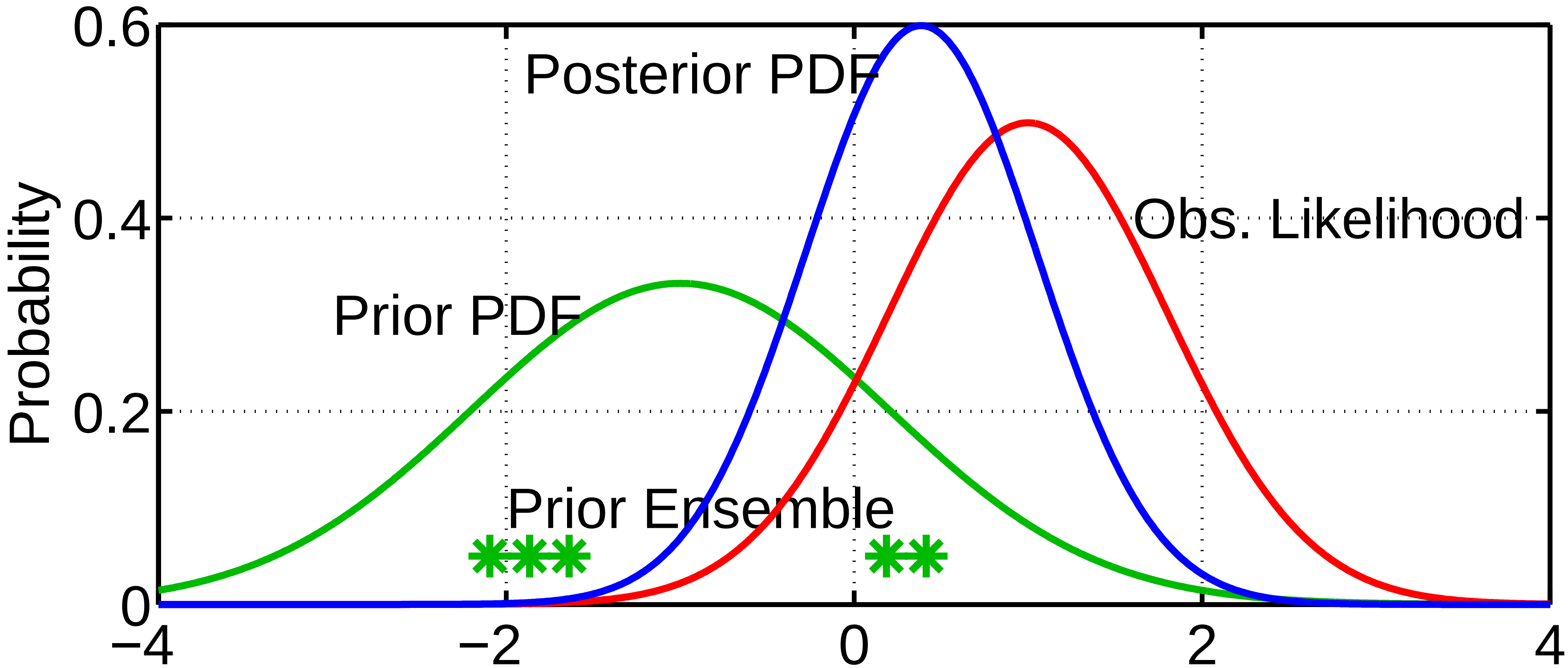
Ensemble Adjustment Filter (a deterministic square root filter).



Again, fit a Gaussian to sample.

Ensemble Filter Algorithms:

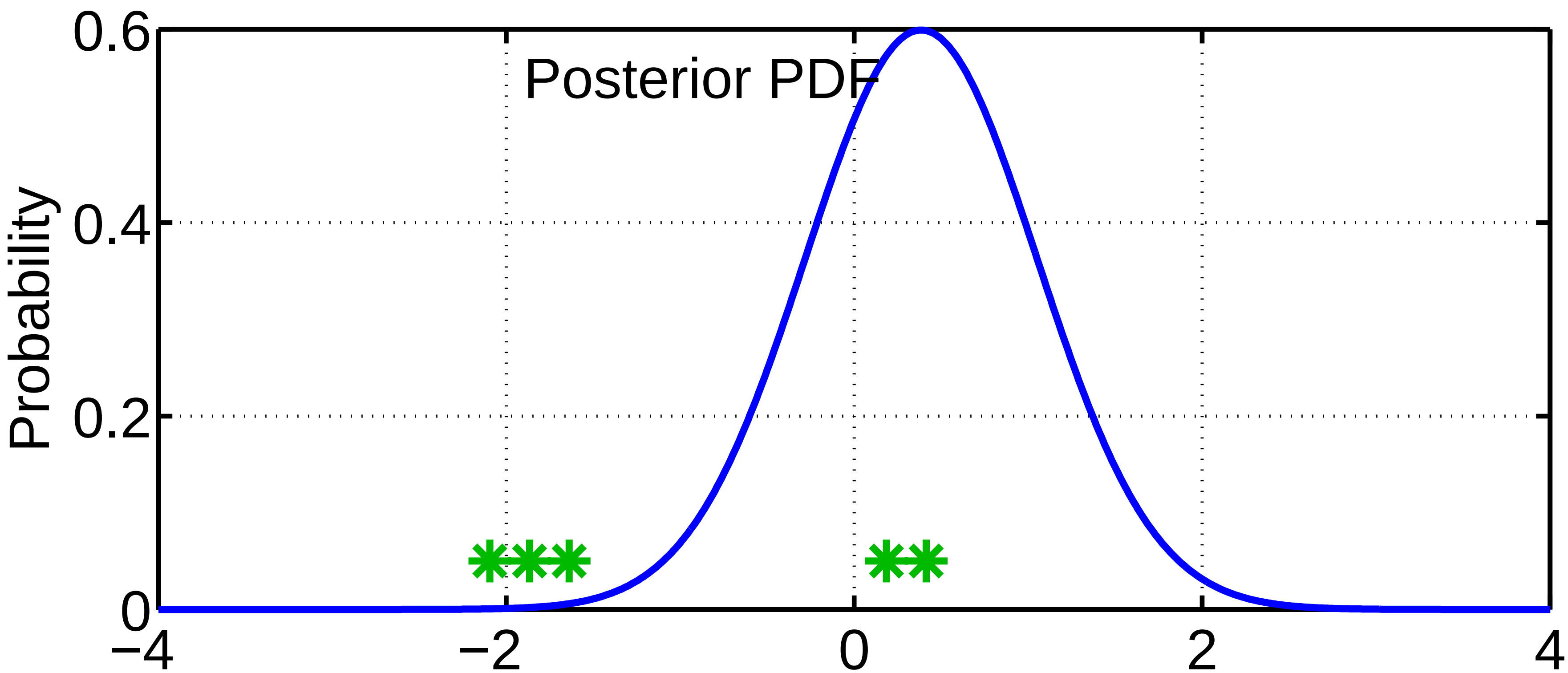
Ensemble Adjustment Filter (a deterministic square root filter).



Compute posterior PDF.

Ensemble Filter Algorithms:

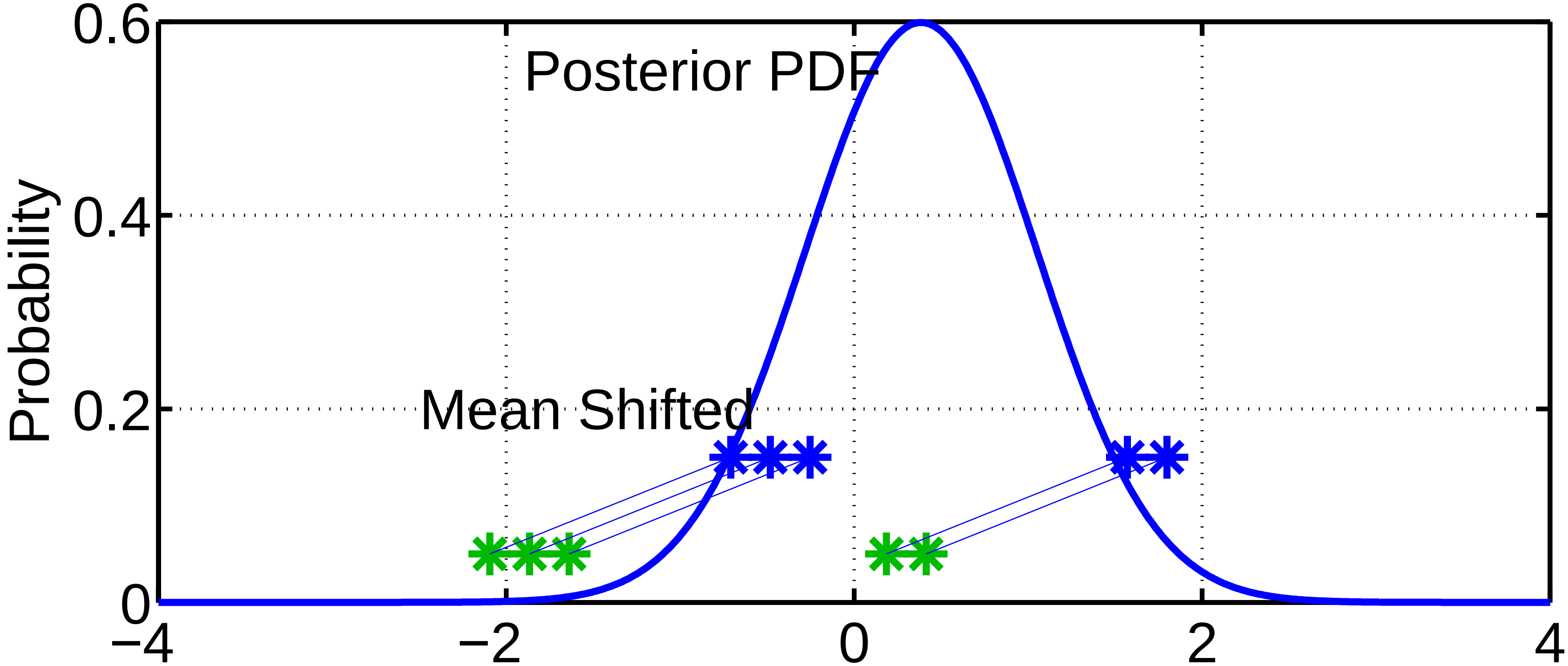
Ensemble Adjustment Filter (a deterministic square root filter).



Use deterministic algorithm to ‘adjust’ ensemble.

Ensemble Filter Algorithms:

Ensemble Adjustment Filter (a deterministic square root filter).

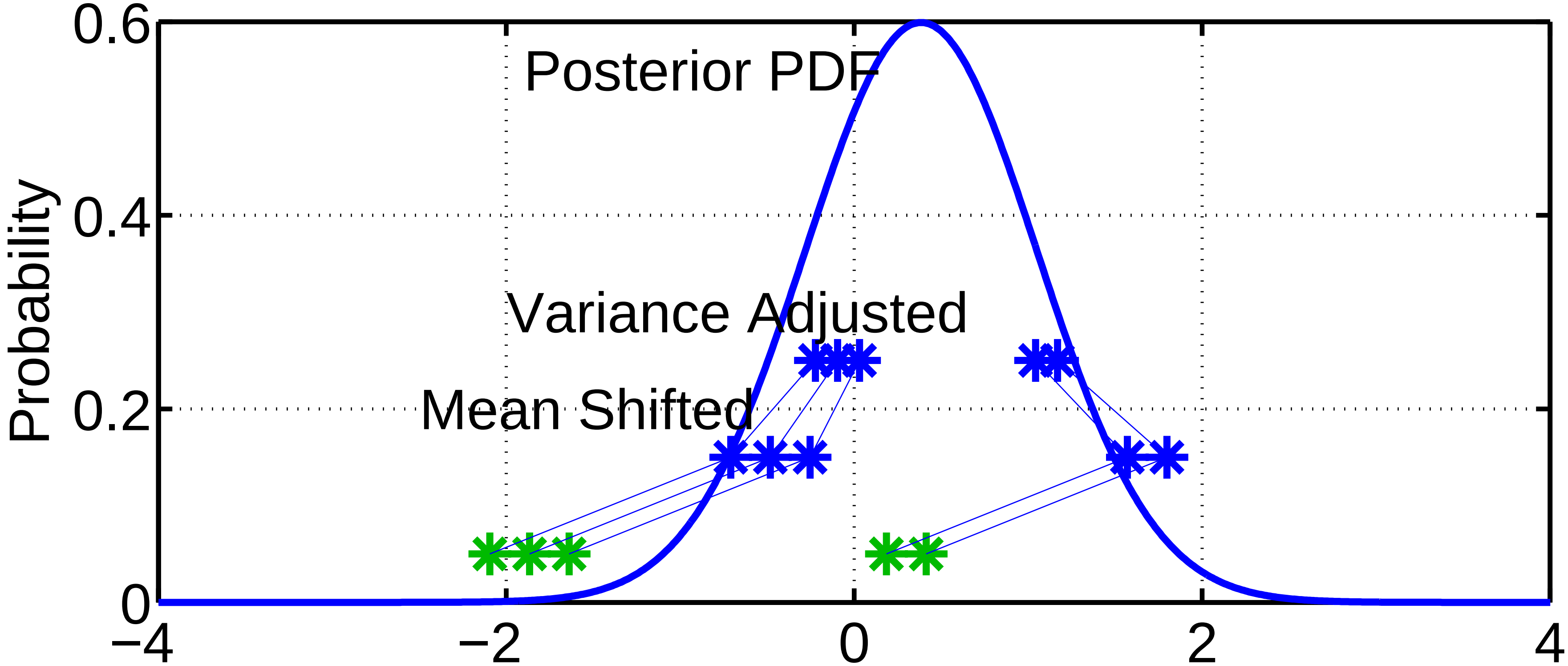


Use deterministic algorithm to 'adjust' ensemble.

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

Ensemble Filter Algorithms:

Ensemble Adjustment Filter (a deterministic square root filter).



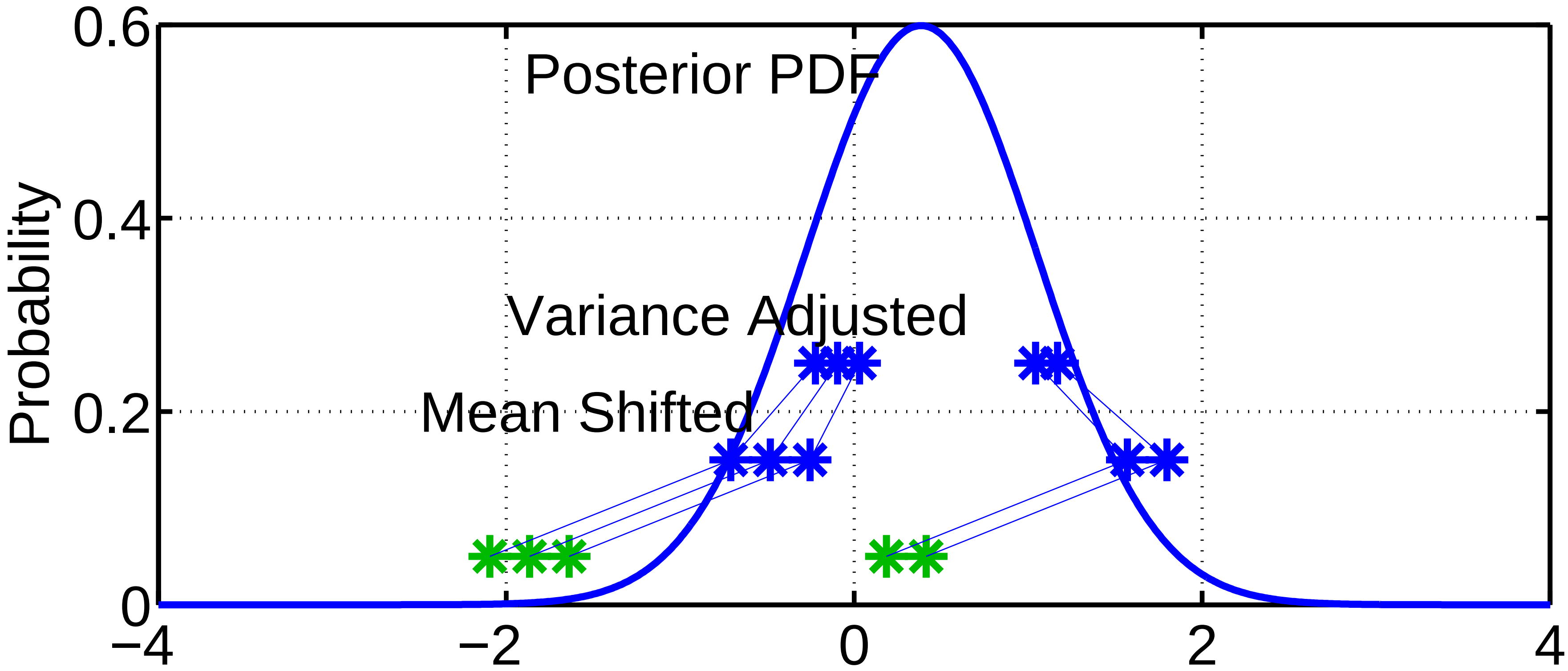
Use deterministic algorithm to ‘adjust’ ensemble.

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

Second, use linear contraction to have exact variance of posterior.

Ensemble Filter Algorithms:

Ensemble Adjustment Filter (a deterministic square root filter).



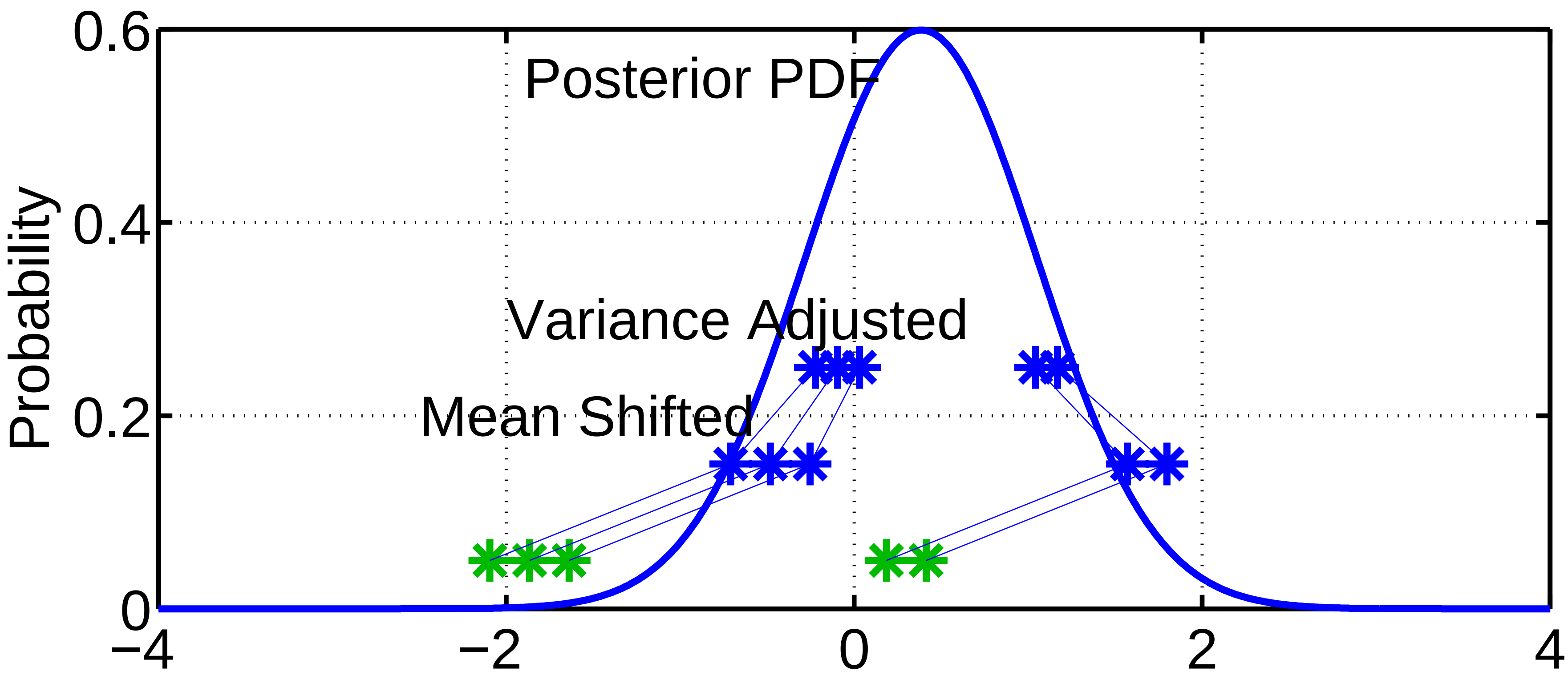
$$x_i^u = (x_i^p - \overline{x^p}) \cdot (\sigma^u / \sigma^p) + \overline{x^u} \quad \text{i = 1,..., ensemble size.}$$

p is prior, u is update (posterior), overbar is ensemble mean,

σ is standard deviation.

Ensemble Filter Algorithms:

Ensemble Adjustment Filter (a deterministic square root filter).



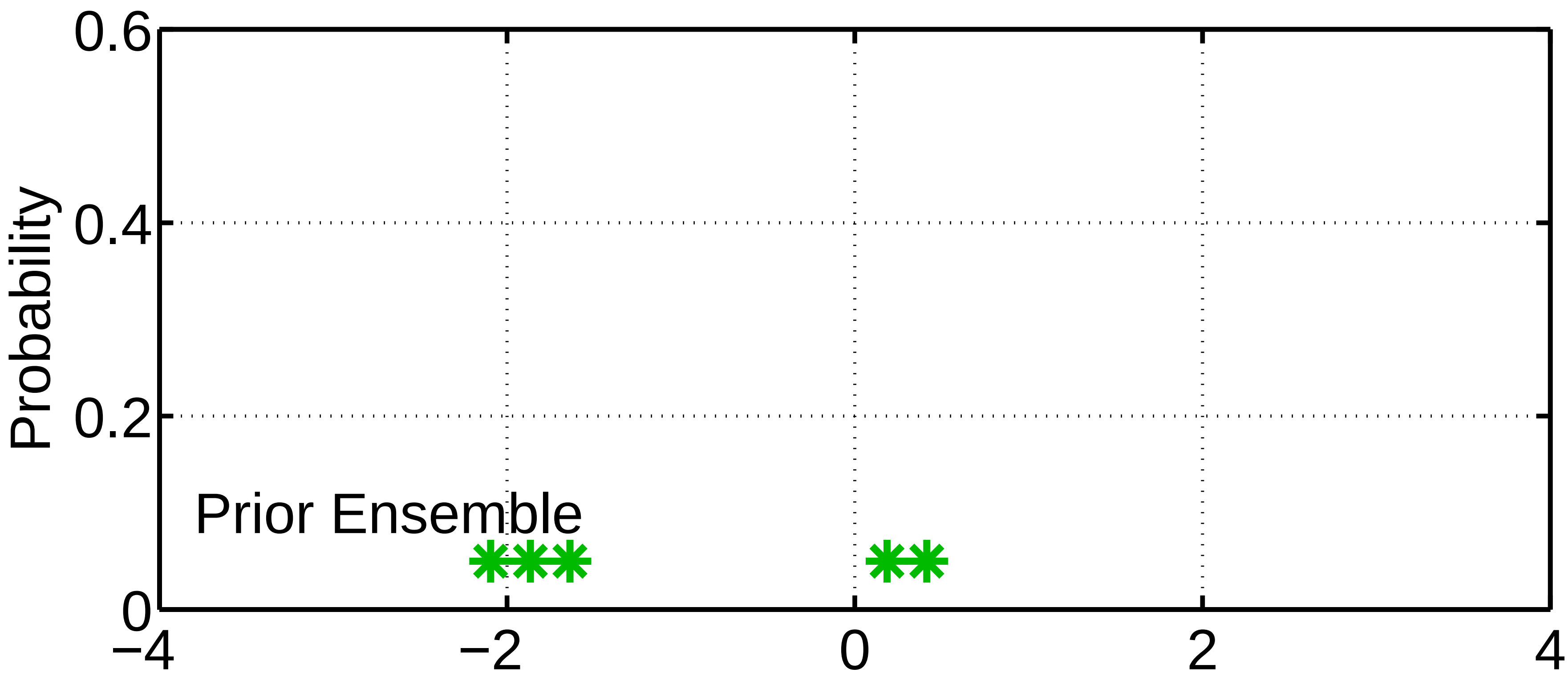
Bimodality maintained, but not appropriately positioned or weighted.

No problem with random outliers.

<u>For linear, gaussian, large enough ensemble, this is EXACTLY KF.</u>

Ensemble Filter Algorithms:

Ensemble Kalman Filter (EnKF).

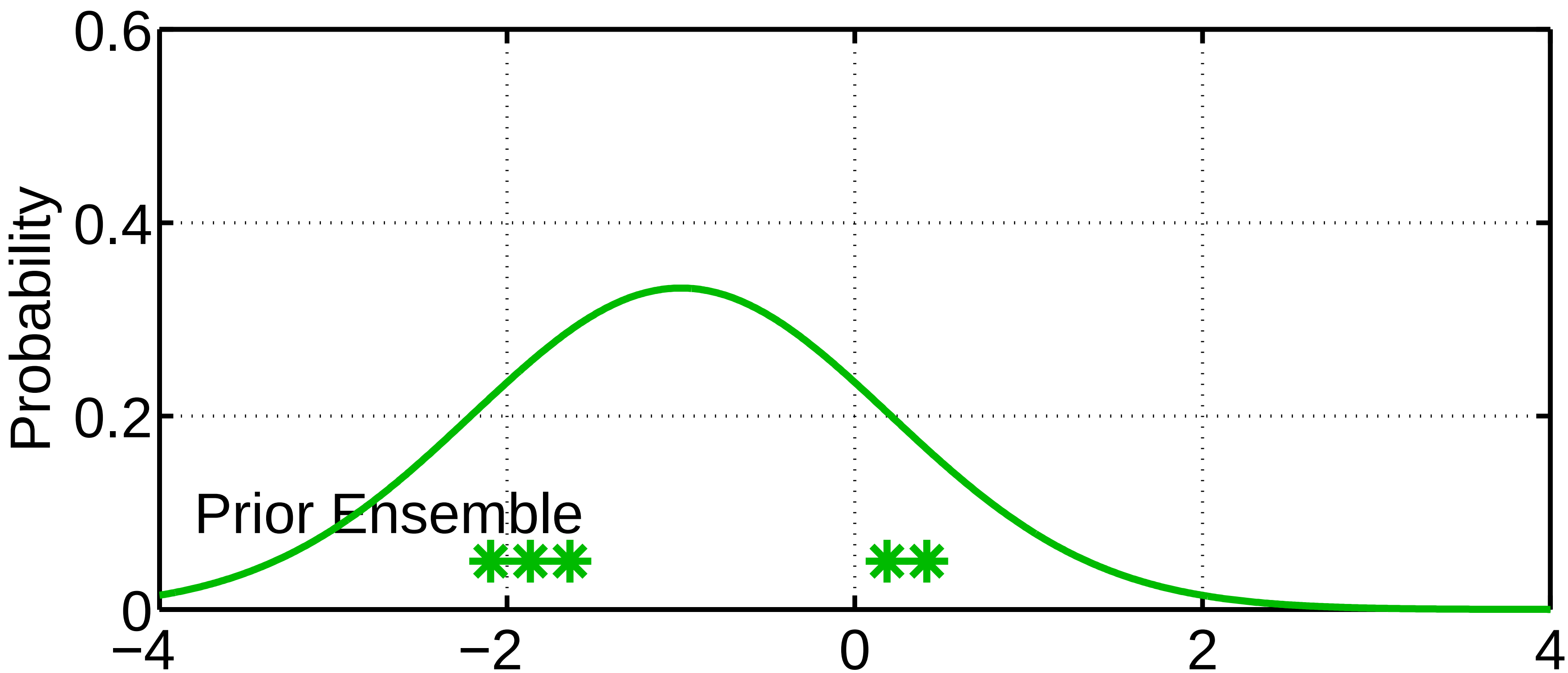


'Classical' Monte Carlo Algorithm for Data Assimilation.

Note: earliest refs have incorrect algorithm.

Ensemble Filter Algorithms:

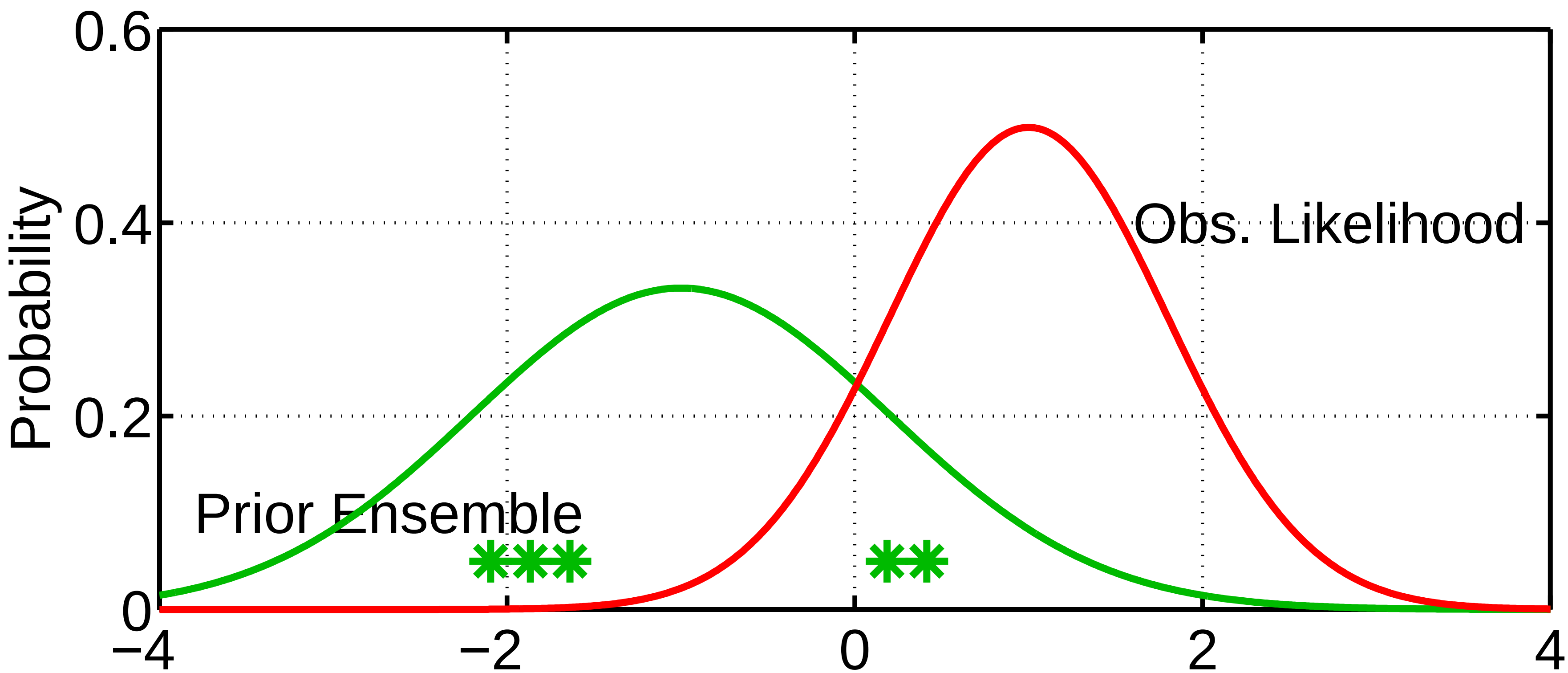
Ensemble Kalman Filter (EnKF).



Again, fit a Gaussian to sample.

Ensemble Filter Algorithms:

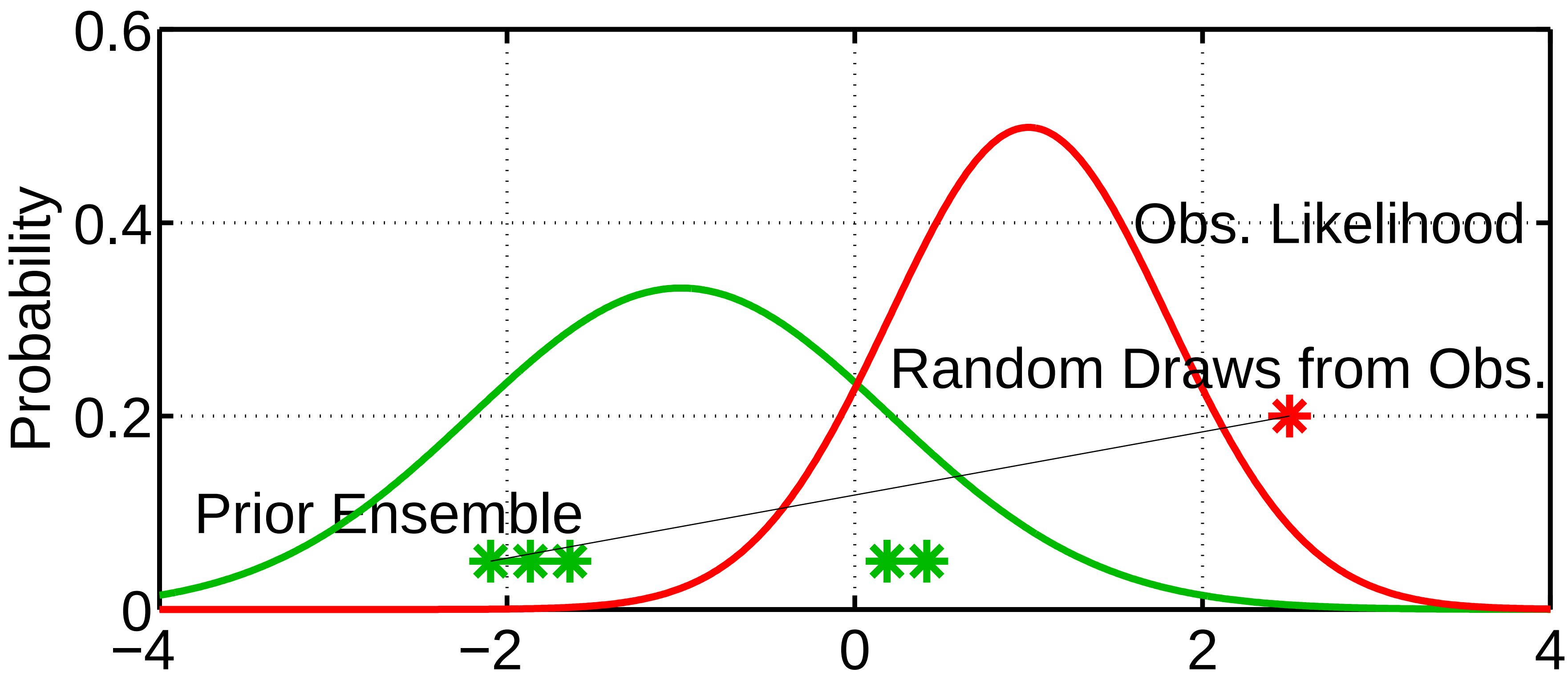
Ensemble Kalman Filter (EnKF).



Again, fit a Gaussian to sample.

Ensemble Filter Algorithms:

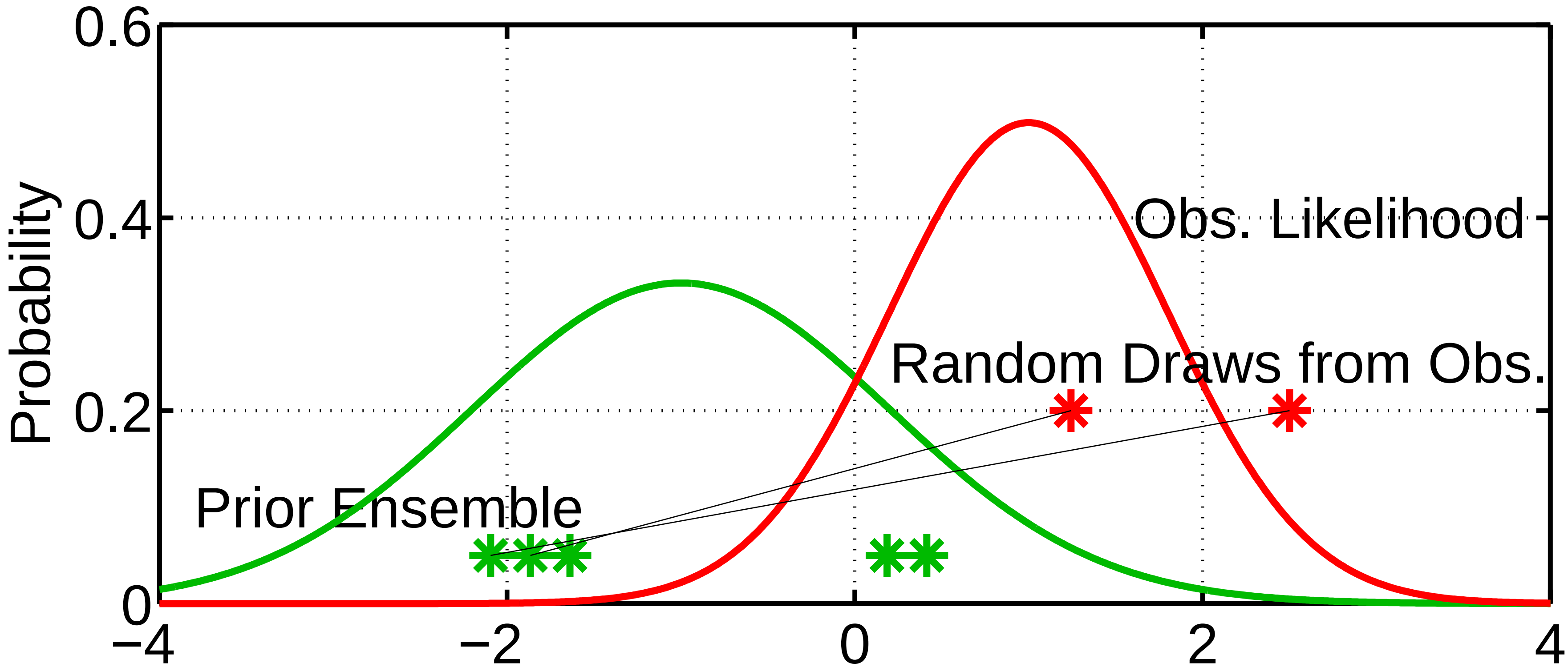
Ensemble Kalman Filter (EnKF).



Generate a random draw from the obs. likelihood.

Associate it with the first sample of prior ensemble.

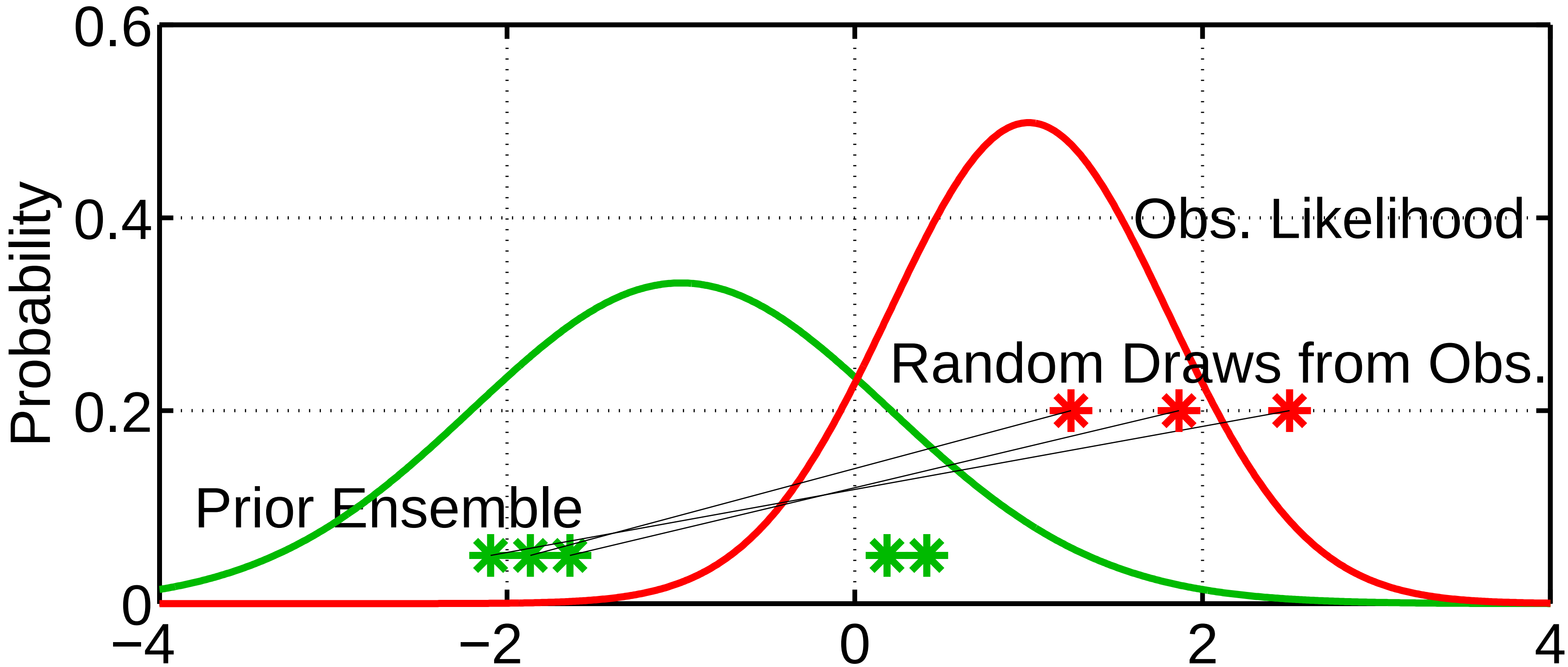
Ensemble Kalman Filter (EnKF).



Proceed to associate a random draw from obs. with each prior sample.
This has been called ‘perturbed’ observations.
Algorithm sometimes called ‘perturbed obs.’ ensemble Kalman filter.

Ensemble Filter Algorithms:

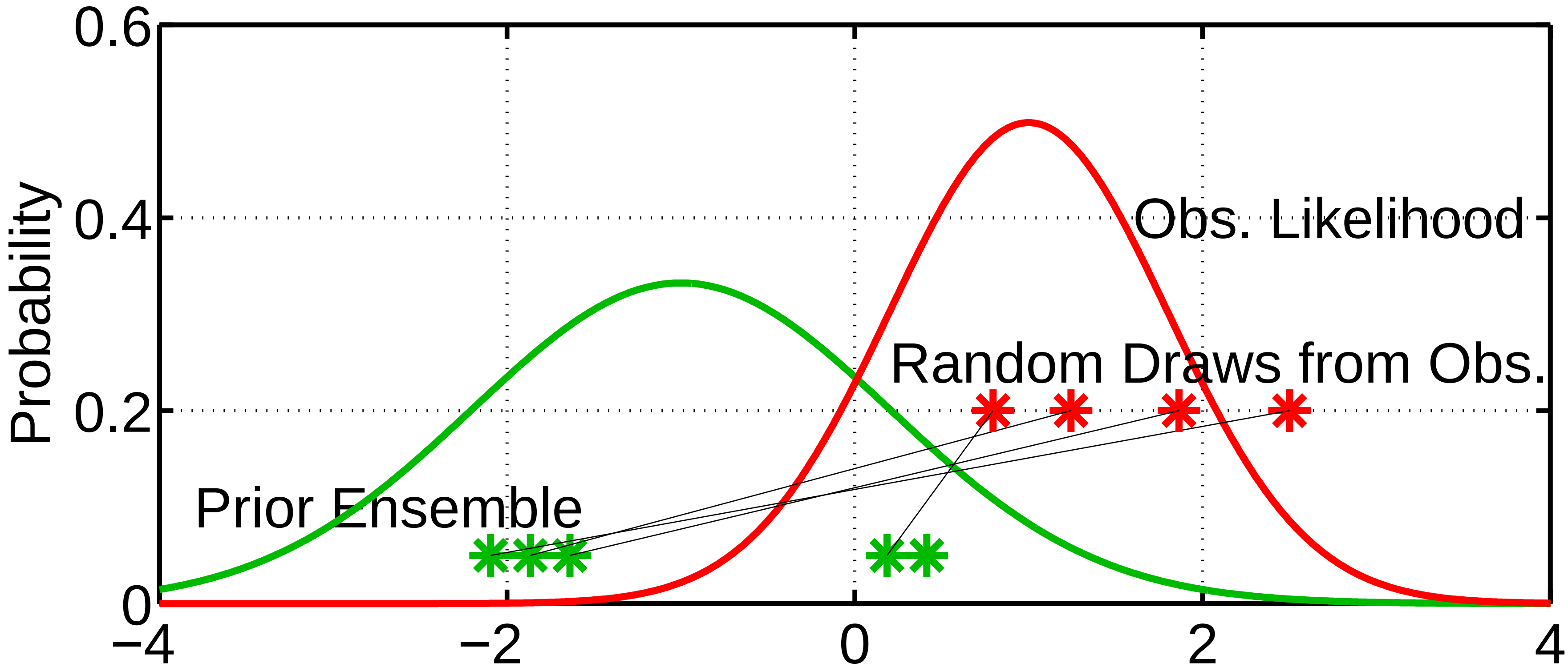
Ensemble Kalman Filter (EnKF).



Proceed to associate a random draw from obs. with each prior sample.
This has been called 'perturbed' observations.
Algorithm sometimes called 'perturbed obs.' ensemble Kalman filter.

Ensemble Filter Algorithms:

Ensemble Kalman Filter (EnKF).



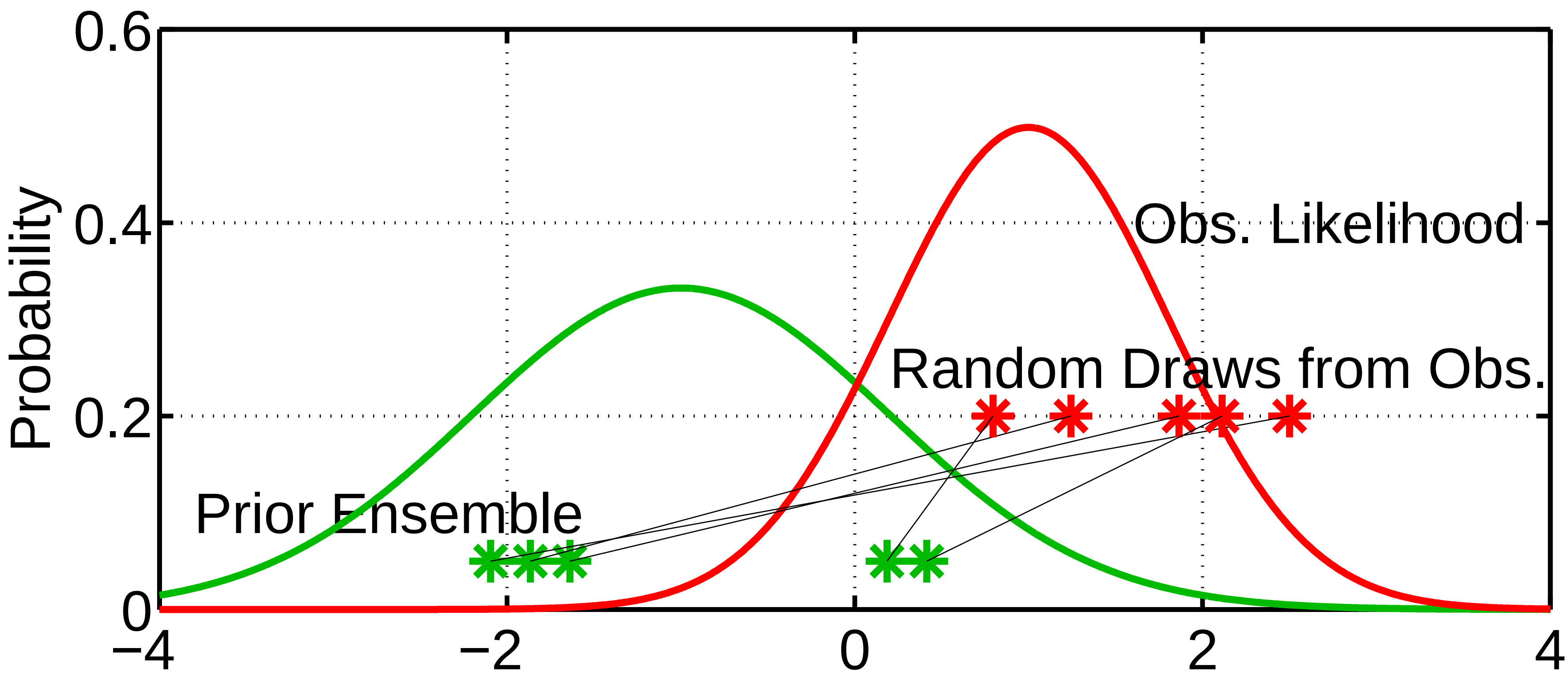
Proceed to associate a random draw from obs. with each prior sample.

Earliest publications associated mean of obs. likelihood with each prior

This resulted in insufficient variance in posterior.

Ensemble Filter Algorithms:

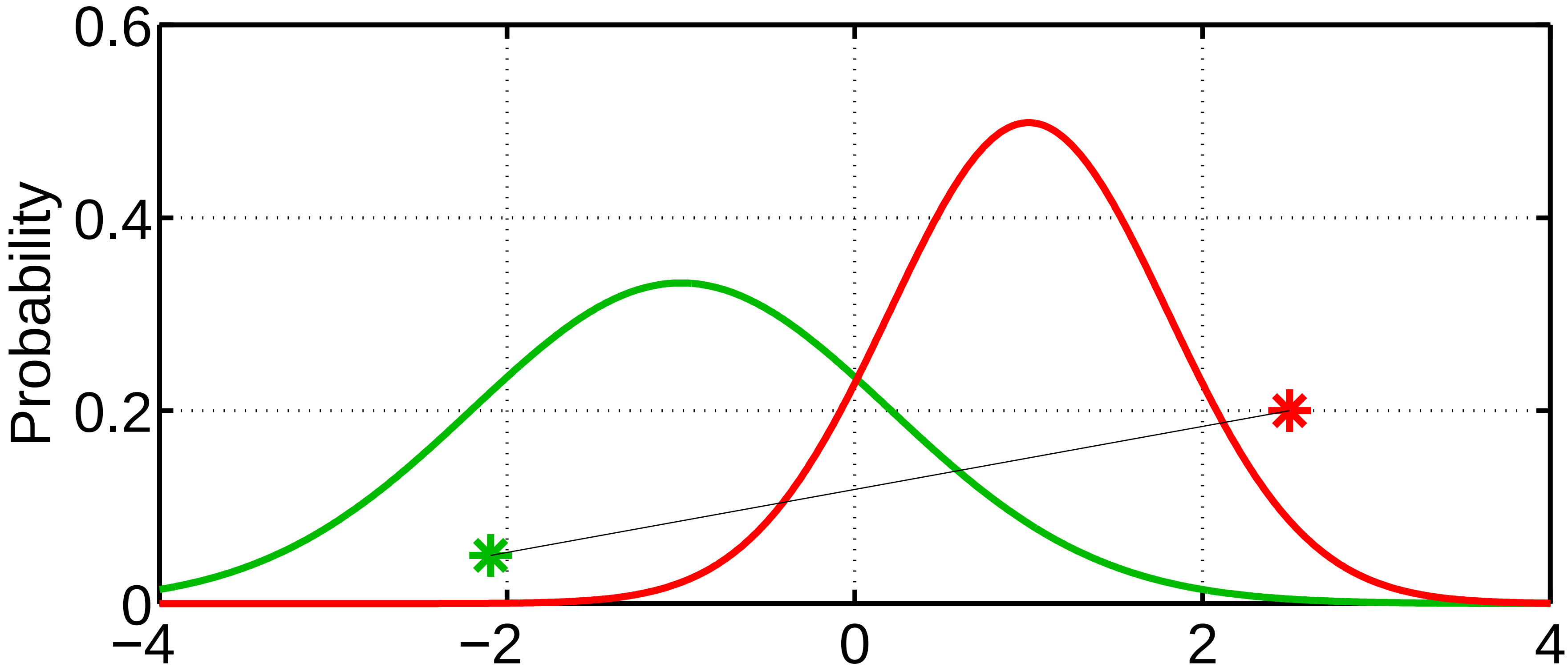
Ensemble Kalman Filter (EnKF).



Adjusting the mean of obs. sample to be exact improves performance.
Adjusting the variance may further improve performance.
Outliers are potential problem, but can be removed.

Ensemble Filter Algorithms:

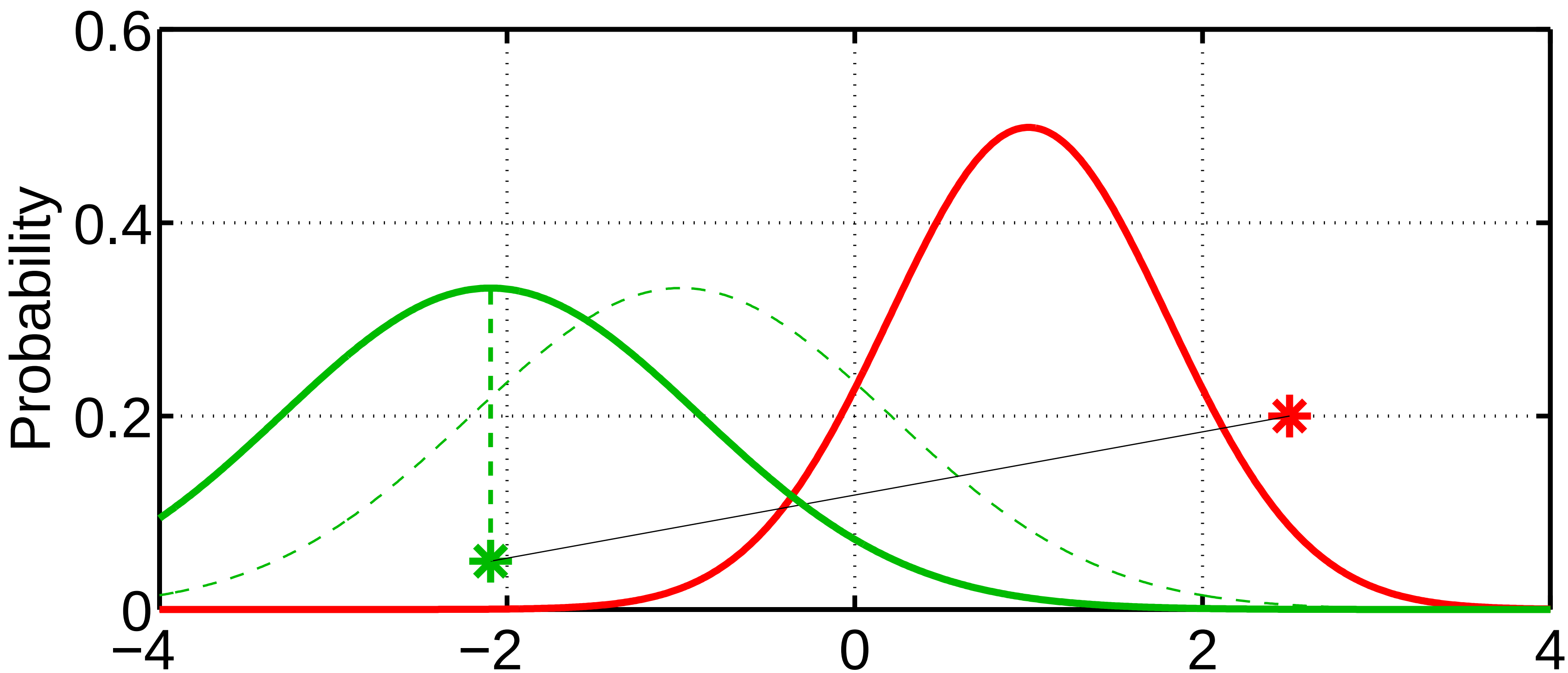
Ensemble Kalman Filter (EnKF).



For each prior mean/obs. pair, find mean of posterior PDF.

Ensemble Filter Algorithms:

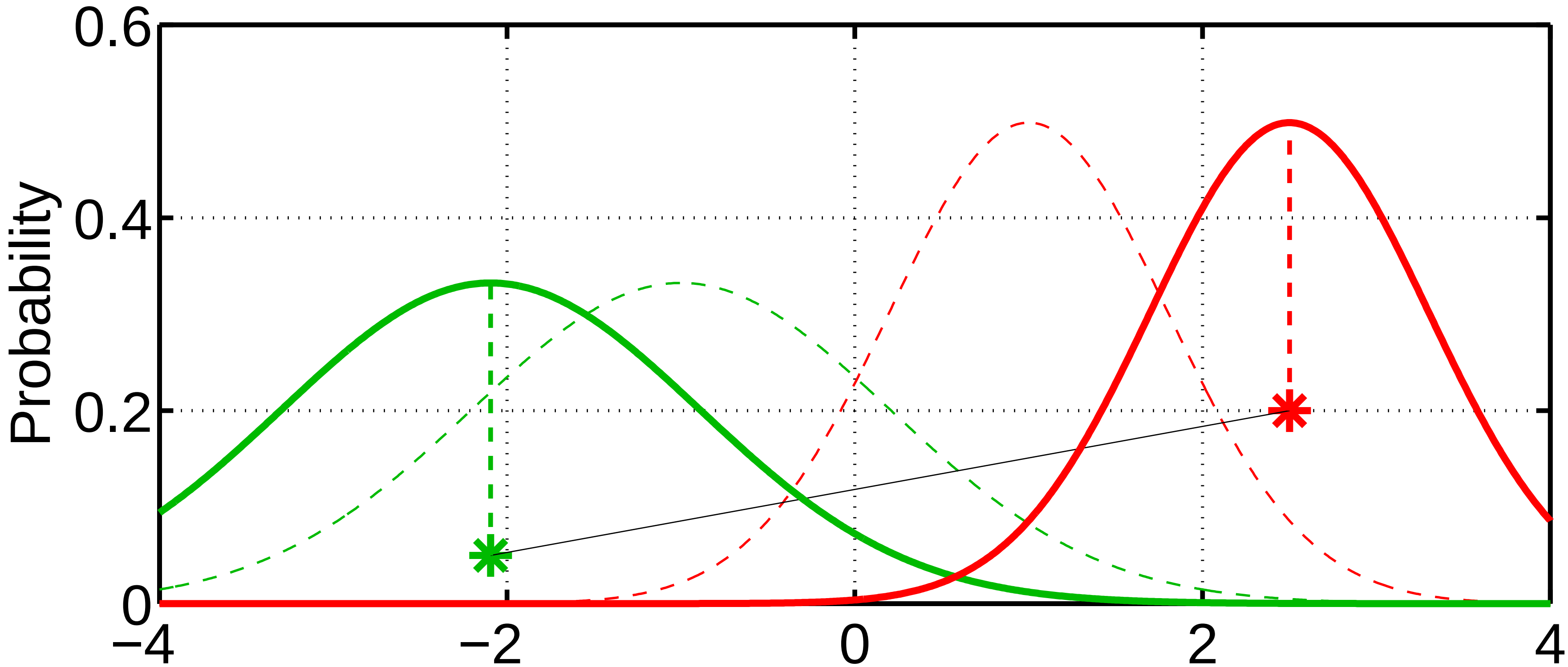
Ensemble Kalman Filter (EnKF).



Prior sample standard deviation still measures uncertainty of prior mean estimate.

Ensemble Filter Algorithms:

Ensemble Kalman Filter (EnKF).

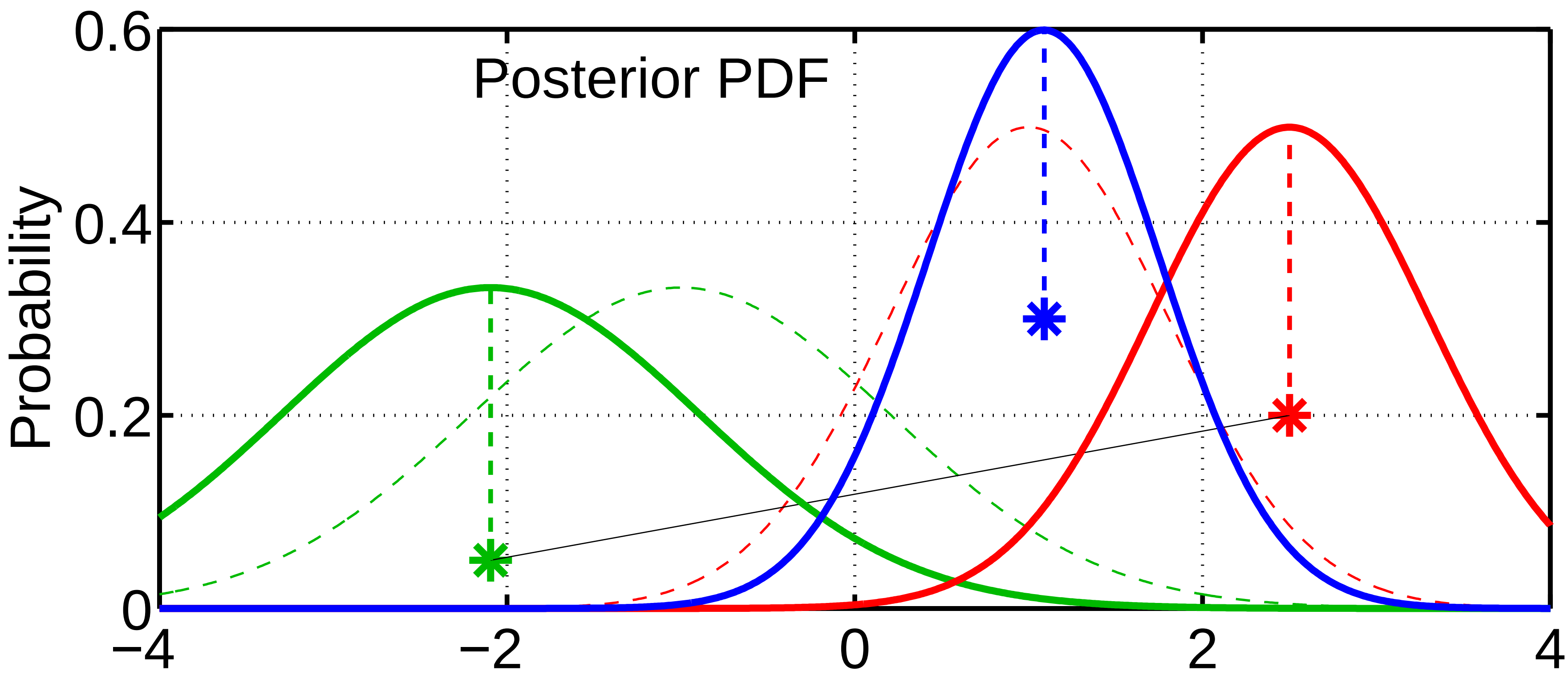


Prior sample standard deviation still measures uncertainty of prior mean estimate.

Obs. likelihood standard deviation measures uncertainty of obs. estimate.

Ensemble Filter Algorithms:

Ensemble Kalman Filter (EnKF).

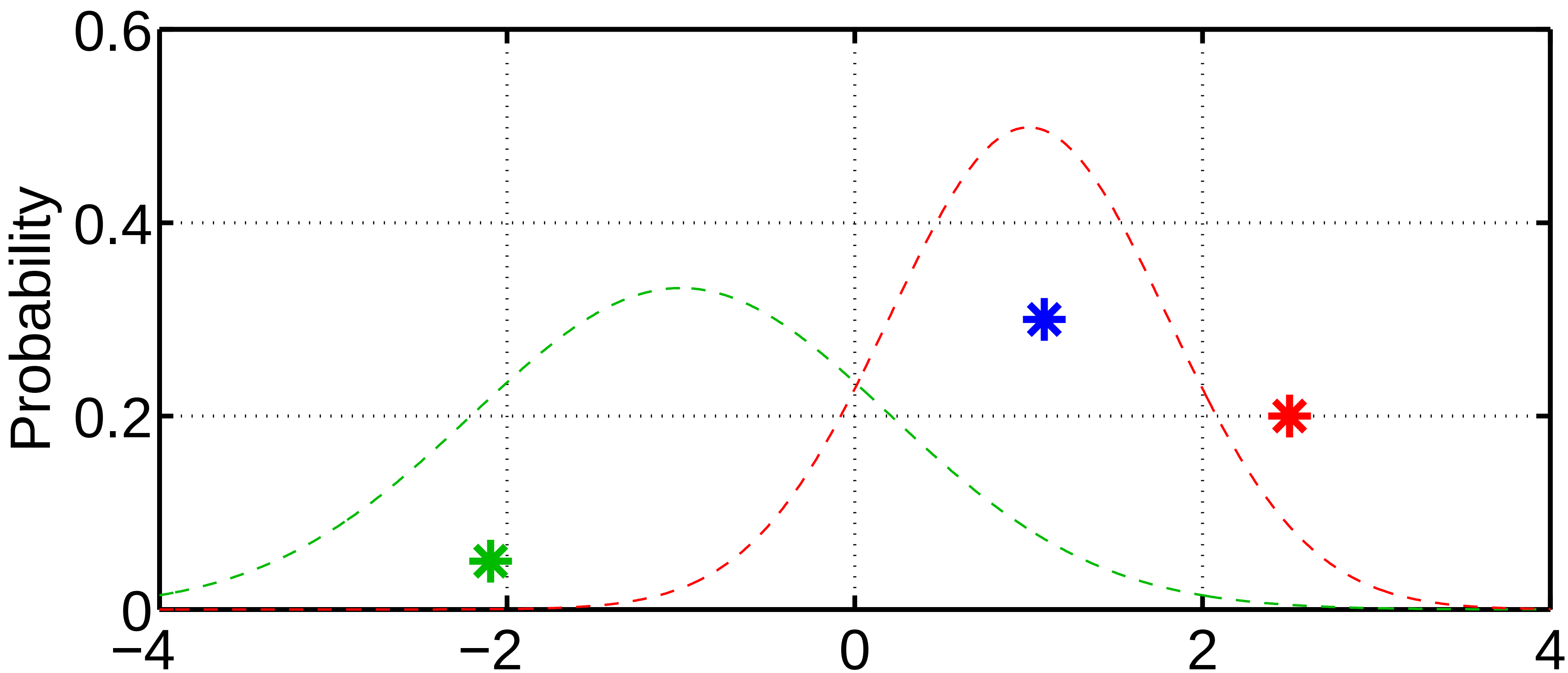


Take product of the prior/obs distributions for first sample.

This is standard Gaussian product.

Ensemble Filter Algorithms:

Ensemble Kalman Filter (EnKF).

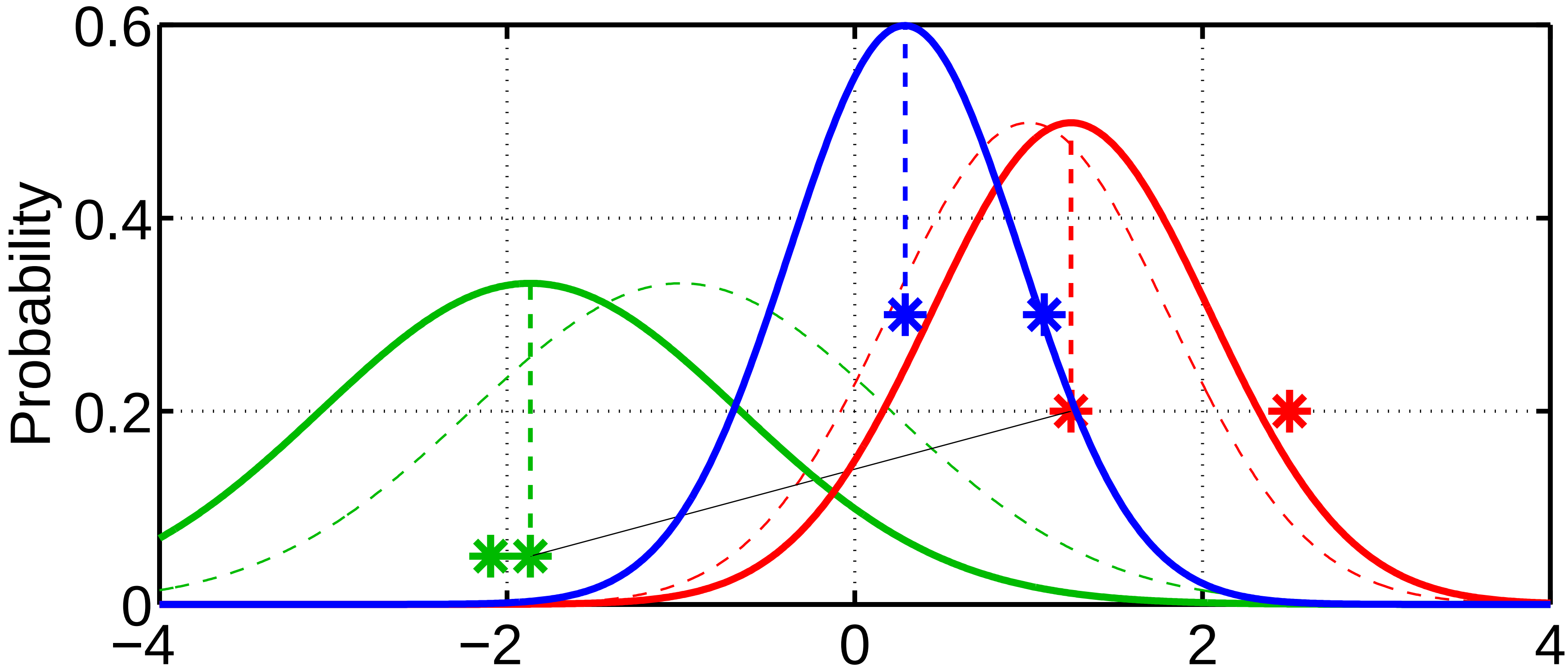


Mean of product is random sample of posterior.

Product of random samples is random sample of product.

Ensemble Filter Algorithms:

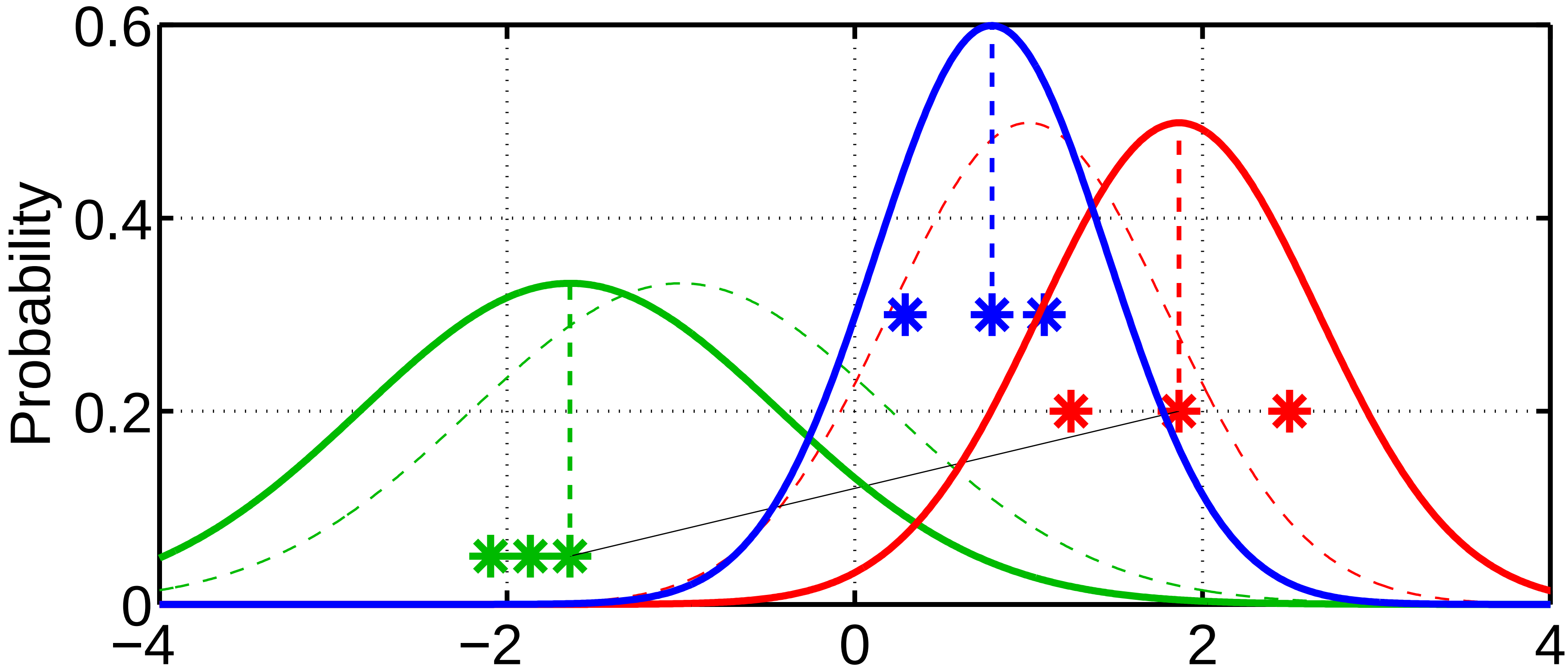
Ensemble Kalman Filter (EnKF).



Repeat this operation for each joint prior pair.

Ensemble Filter Algorithms:

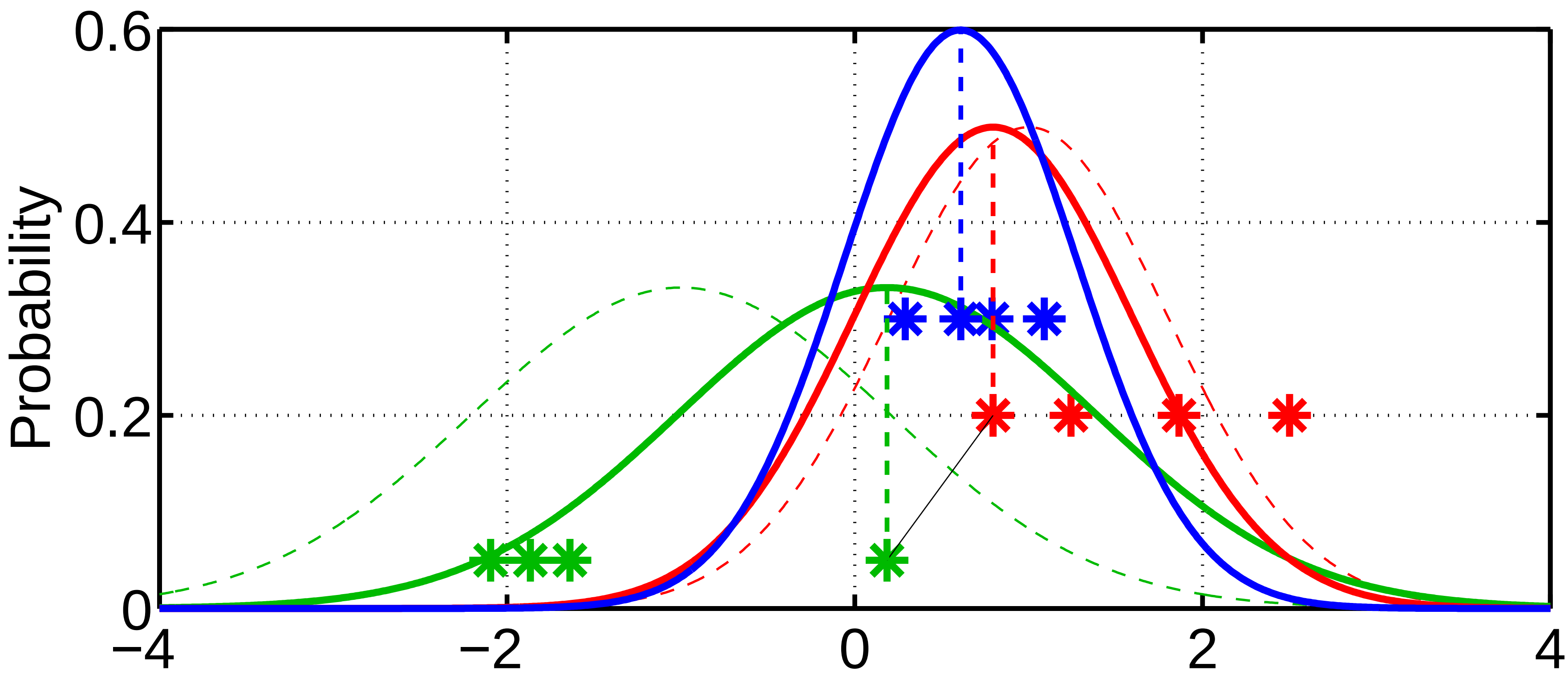
Ensemble Kalman Filter (EnKF).



Repeat this operation for each joint prior pair.

Ensemble Filter Algorithms:

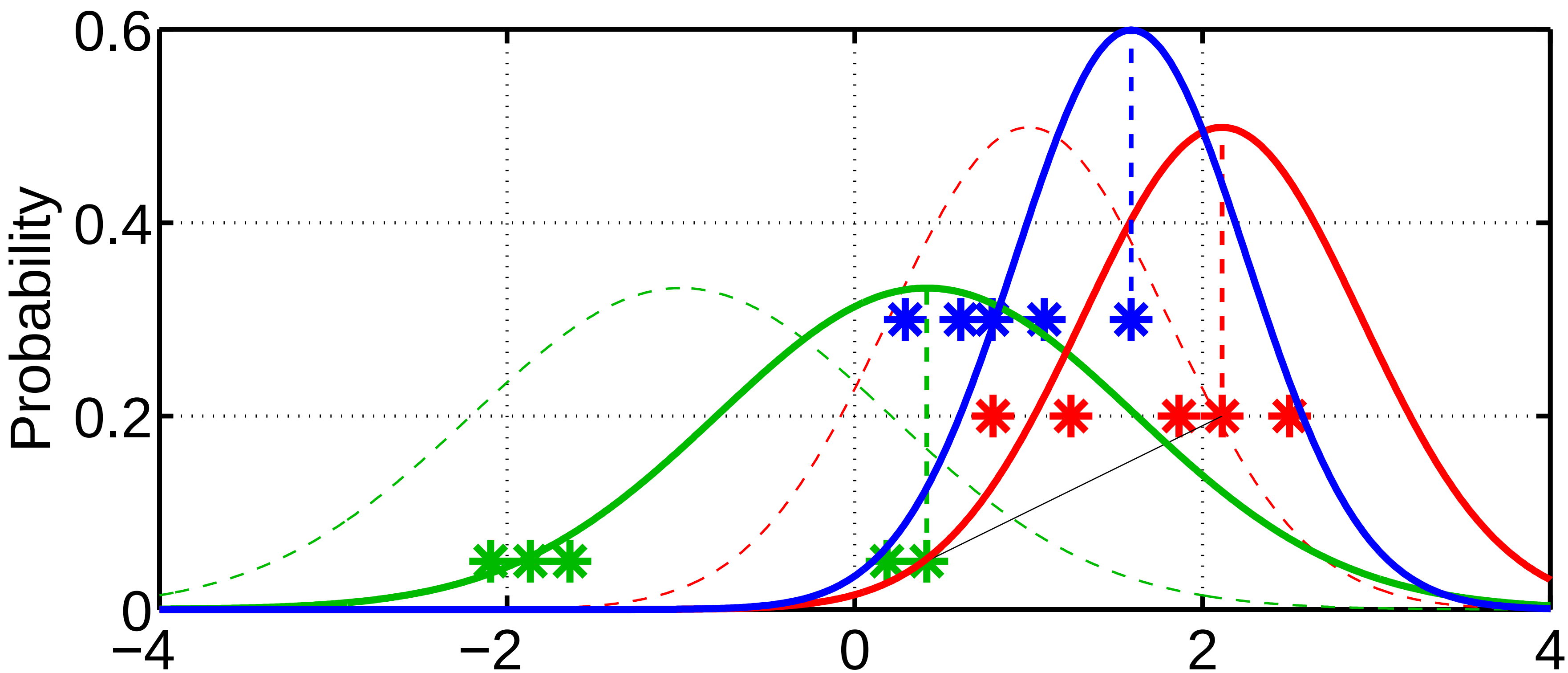
Ensemble Kalman Filter (EnKF)



Repeat this operation for each joint prior pair.

Ensemble Filter Algorithms:

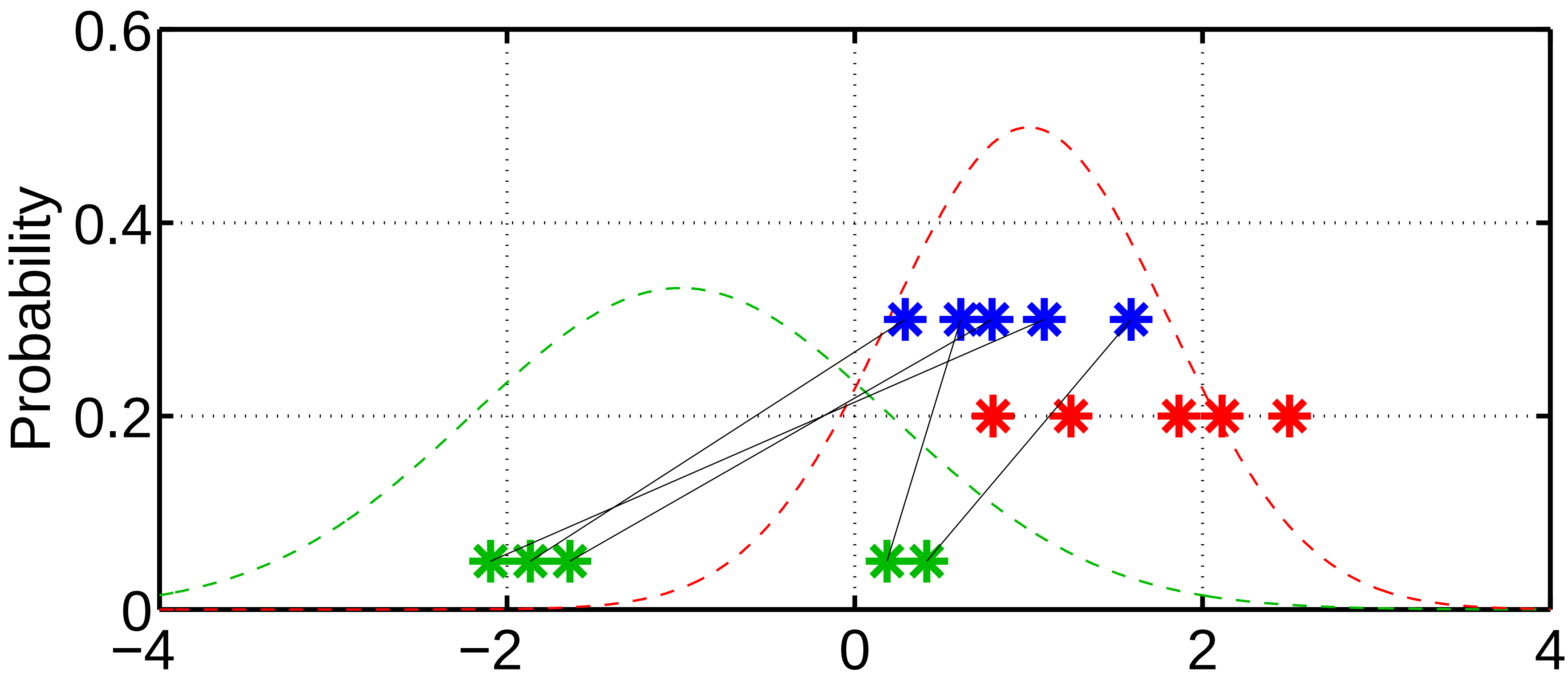
Ensemble Kalman Filter (EnKF).



Repeat this operation for each joint prior pair.

Ensemble Filter Algorithms:

Ensemble Kalman Filter (EnKF).



Posterior sample maintains much of prior sample structure.

(This is more apparent for larger ensemble sizes).

Posterior sample mean and variance converge to 'exact' for large samples.

Sample is mixed by some introduced noise.

A One-Variable Test Model

$$\frac{dx}{dt} = x + \alpha|x|x$$

Assume 'true' trajectory is just x=0.

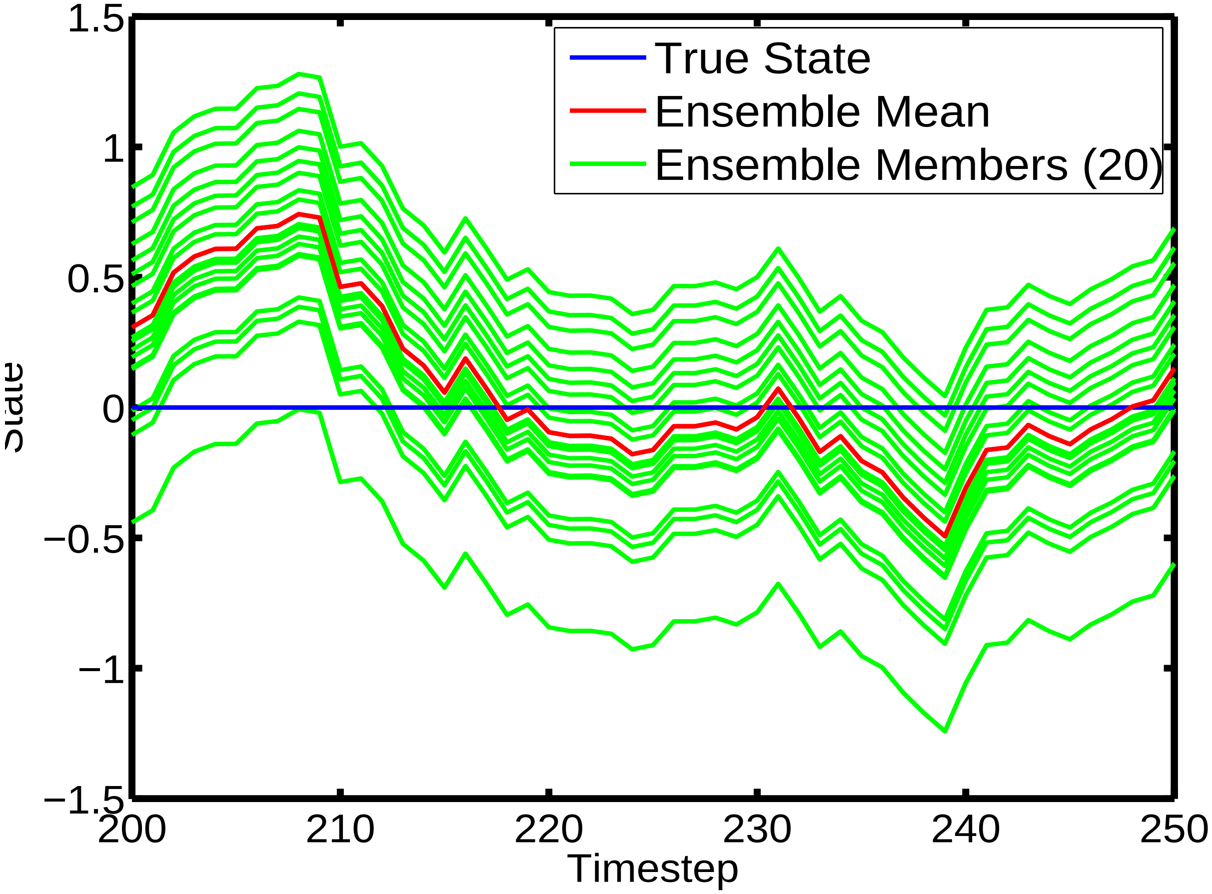
(Same as linearizing around an arbitrary trajectory).

$\alpha = 0$: linear model (exponential growth).

$\alpha > 0$: have additional expansion.

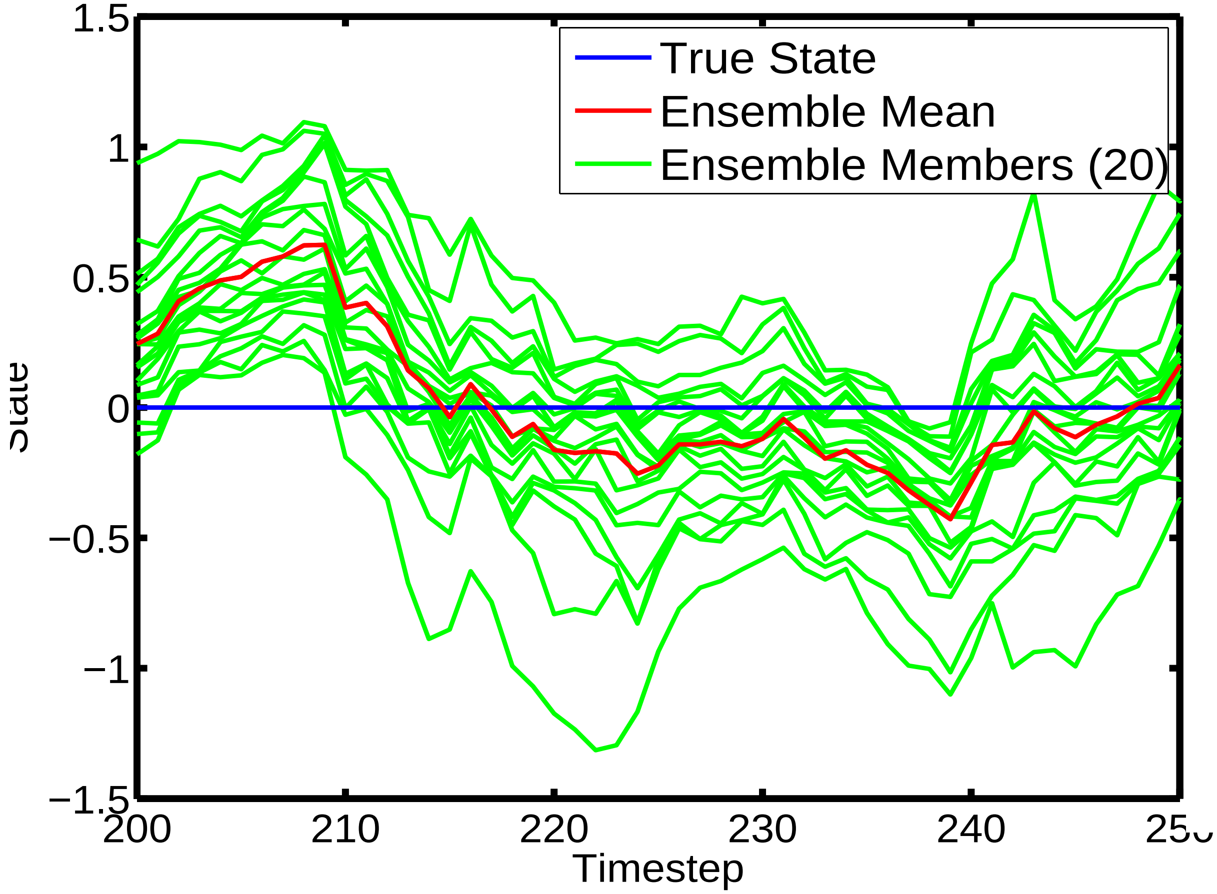
Linear Model Results ($\alpha = 0$): EAKF

(All results throughout are for prior estimates)

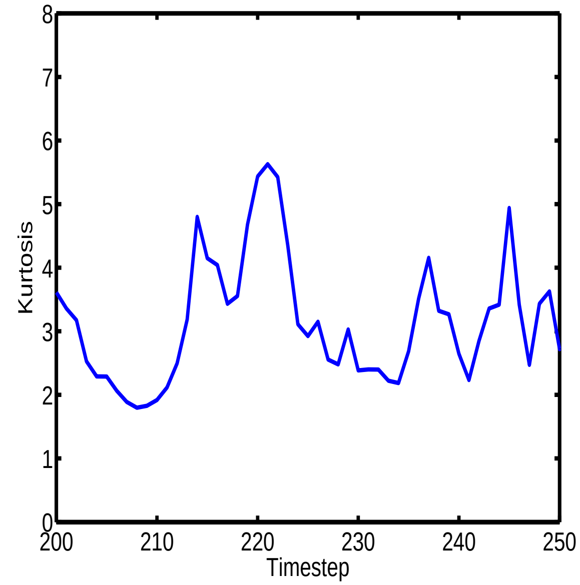


EAKF is just an algorithm for computing Kalman Filter.
Ensemble members don't cross, keep identical spacing.

Linear Model Results (α = 0): EnKF

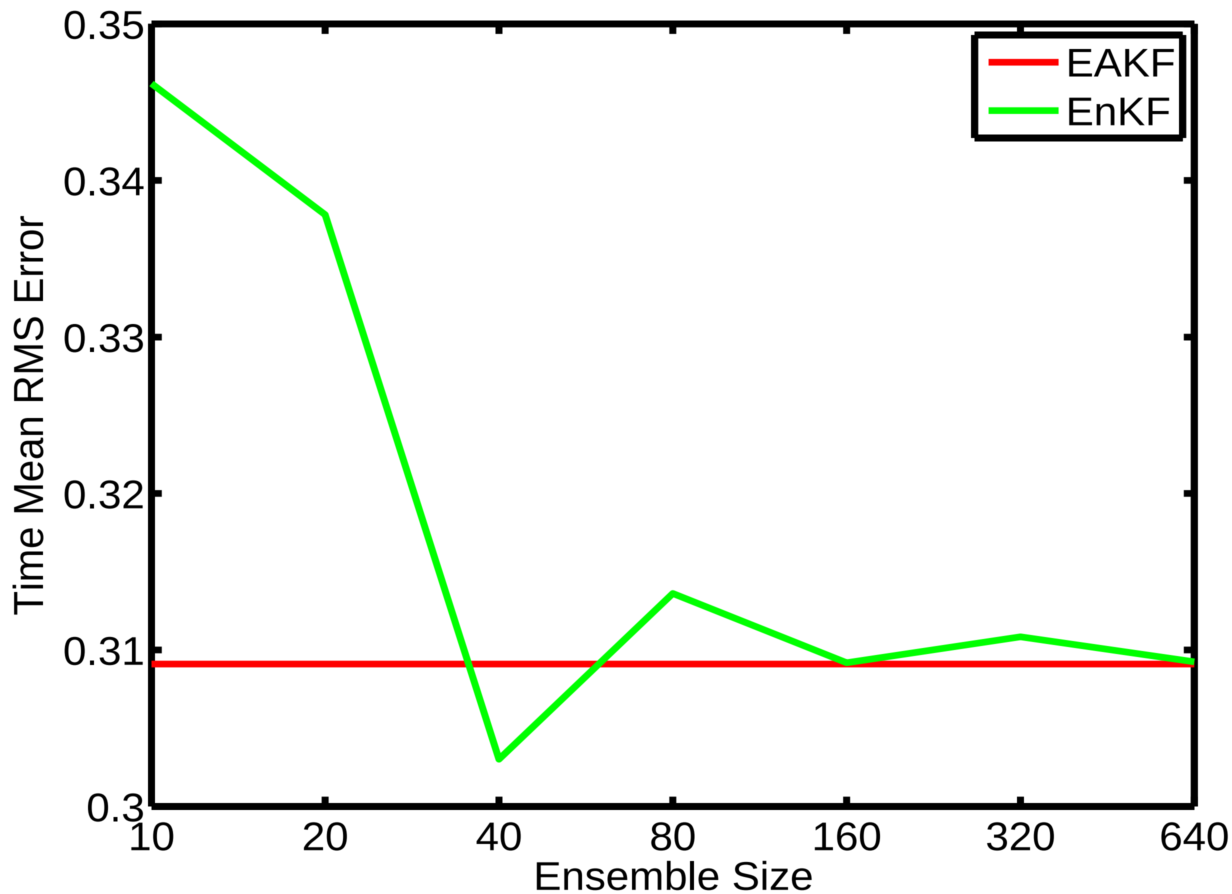


Kurtosis



EnKF is a Monte Carlo algorithm approximating Kalman Filter.
Ensemble members cross, moments (like kurtosis) vary with time.
Sampling error due to small ensembles is an issue.

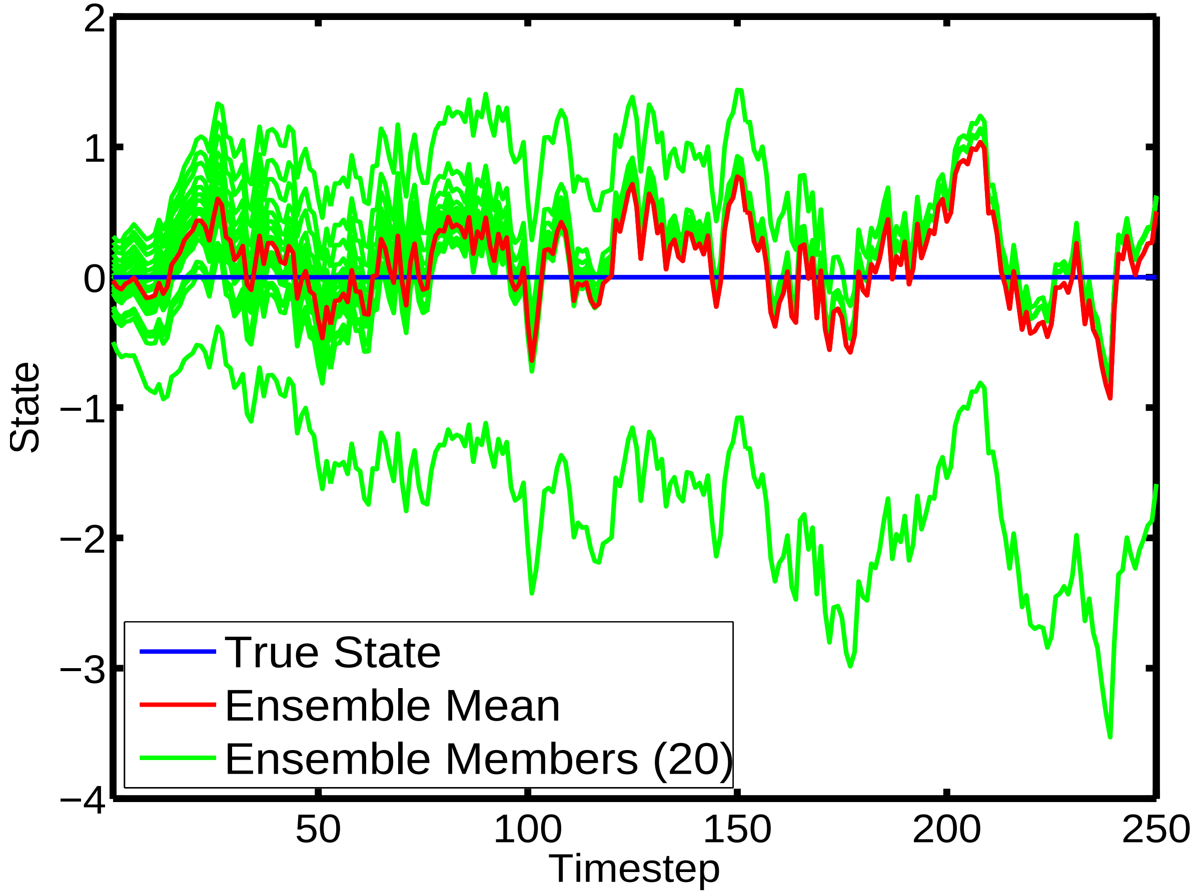
Linear Model RMS Error as Function of Ensemble Size



EAKF exact for any ensemble size (>1).

EnKF has sampling error (smaller value at 40 is an ‘accident’).

Nonlinear Model Results (α = 0.8): EAKF

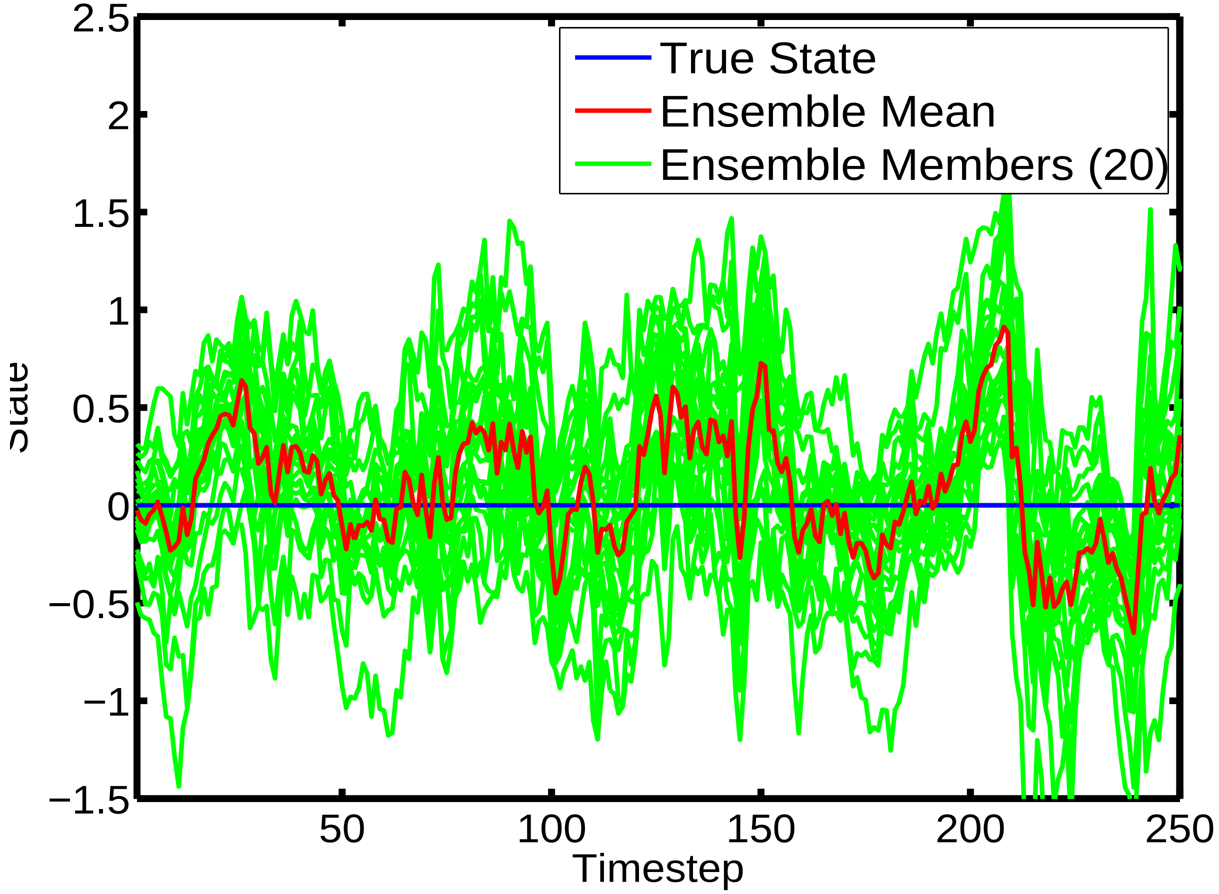


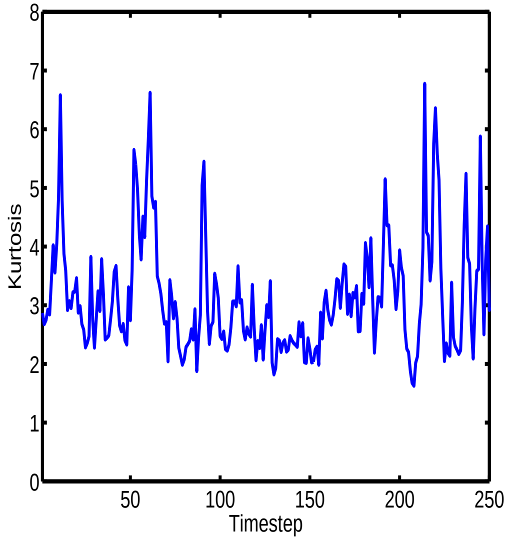
Model advance: furthest outlier pushed out fastest.

All members pulled in linearly by assimilation.

All members but outlier clump together; get huge kurtosis.

Nonlinear Model Results (α = 0.8): EnKF



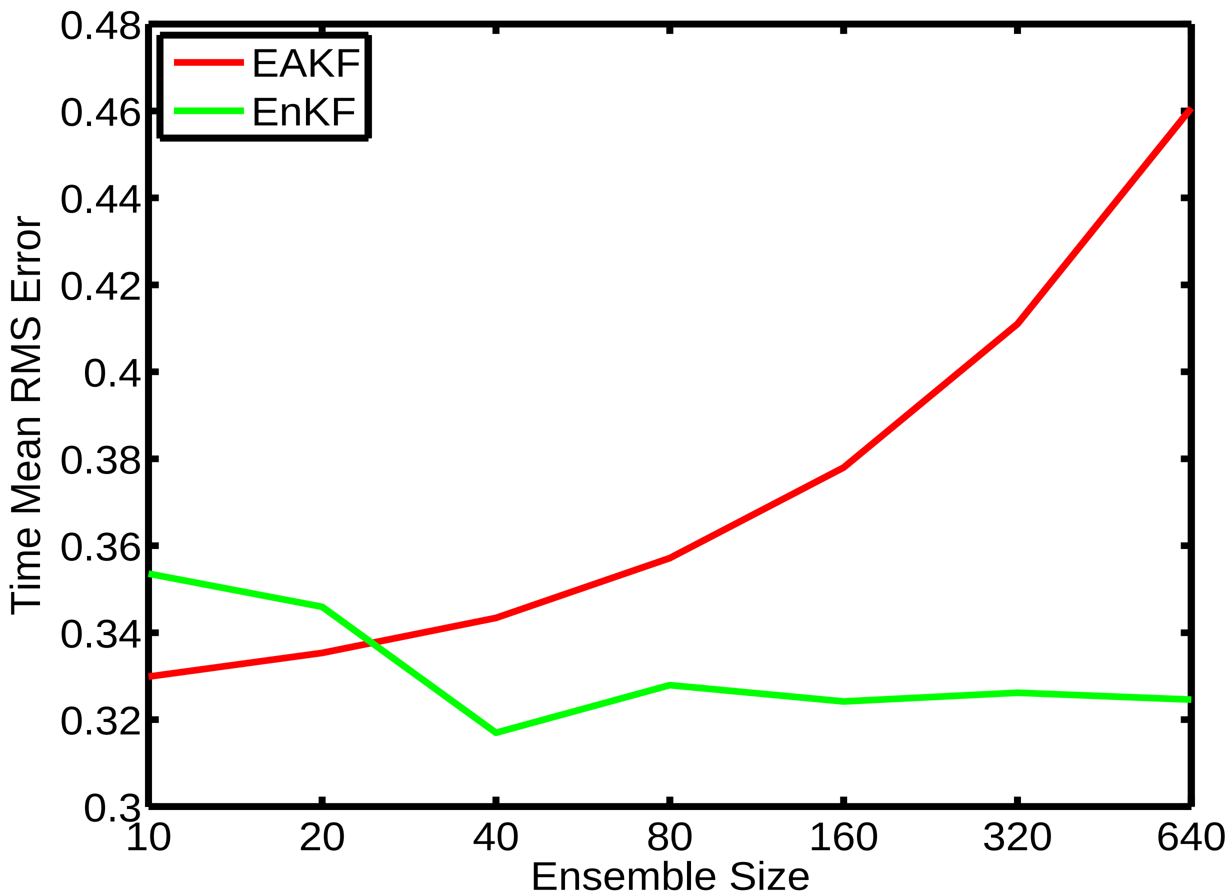


Model advance: furthest outlier pushed out fastest.

Assimilation mixes members some.

Still get high kurtosis sometimes.

EAKF Problem Gets Worse as Ensemble Size Increases (α = 0.2).



RMS error as function of ensemble size.

A Deterministic Non-Gaussian Observation Space Update.

1. Most ensemble filters assume prior and likelihood are ~gaussian.

2. Particle filters do full non-gaussian, but don’t scale.

3. Assuming non-gaussian in observation space is possible.

4. Gaussian kernel filters have been proposed but work poorly.

Requirements for an observation space update:

1. Low information content obs. can’t lead to large increments;

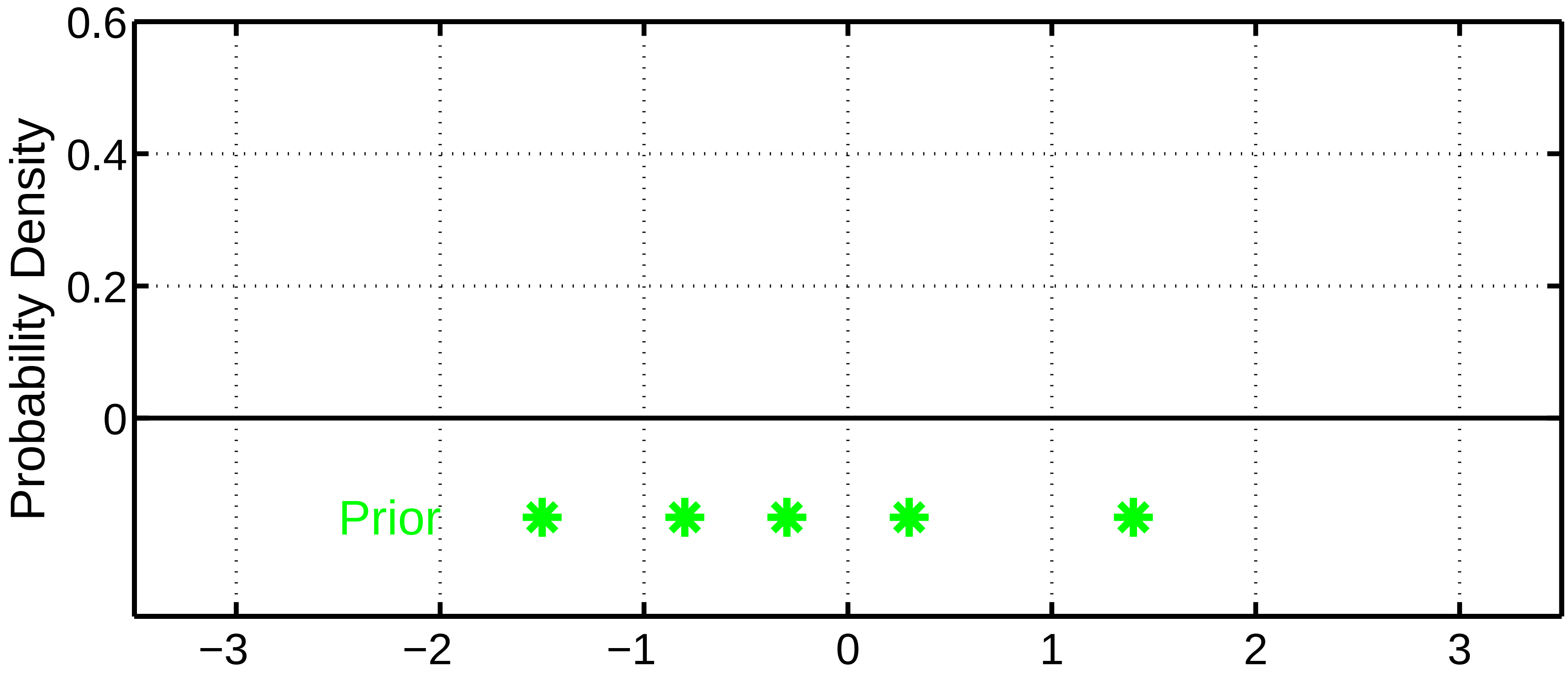
2. Want smallest possible increments for all cases;

3. Comparable to gaussian filters for ~gaussian cases;

4. Better than gaussian in non-gaussian cases;

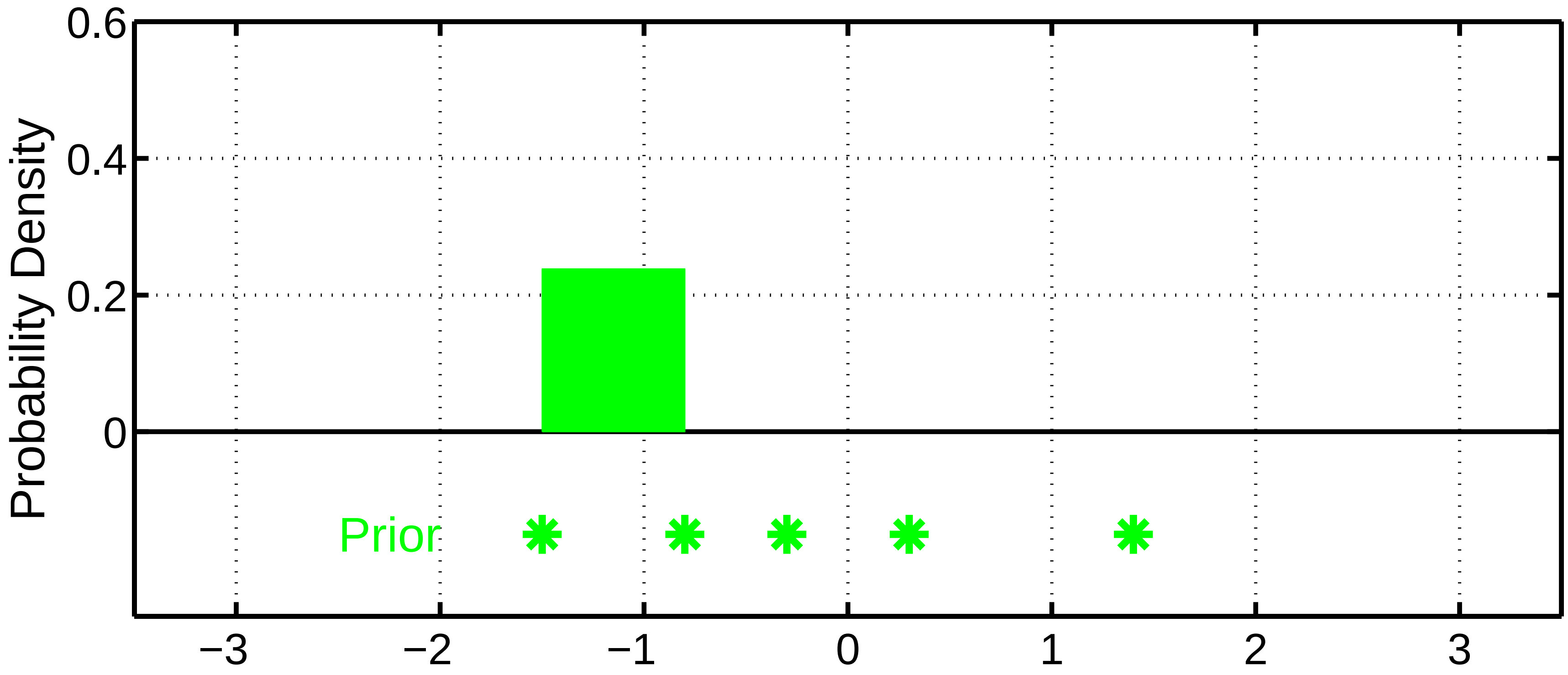
5. Computationally cheap.

Observation Space Rank Histogram Filter



Apply forward operator to each ensemble member.
Get prior ensemble in observation space.

Observation Space Rank Histogram Filter

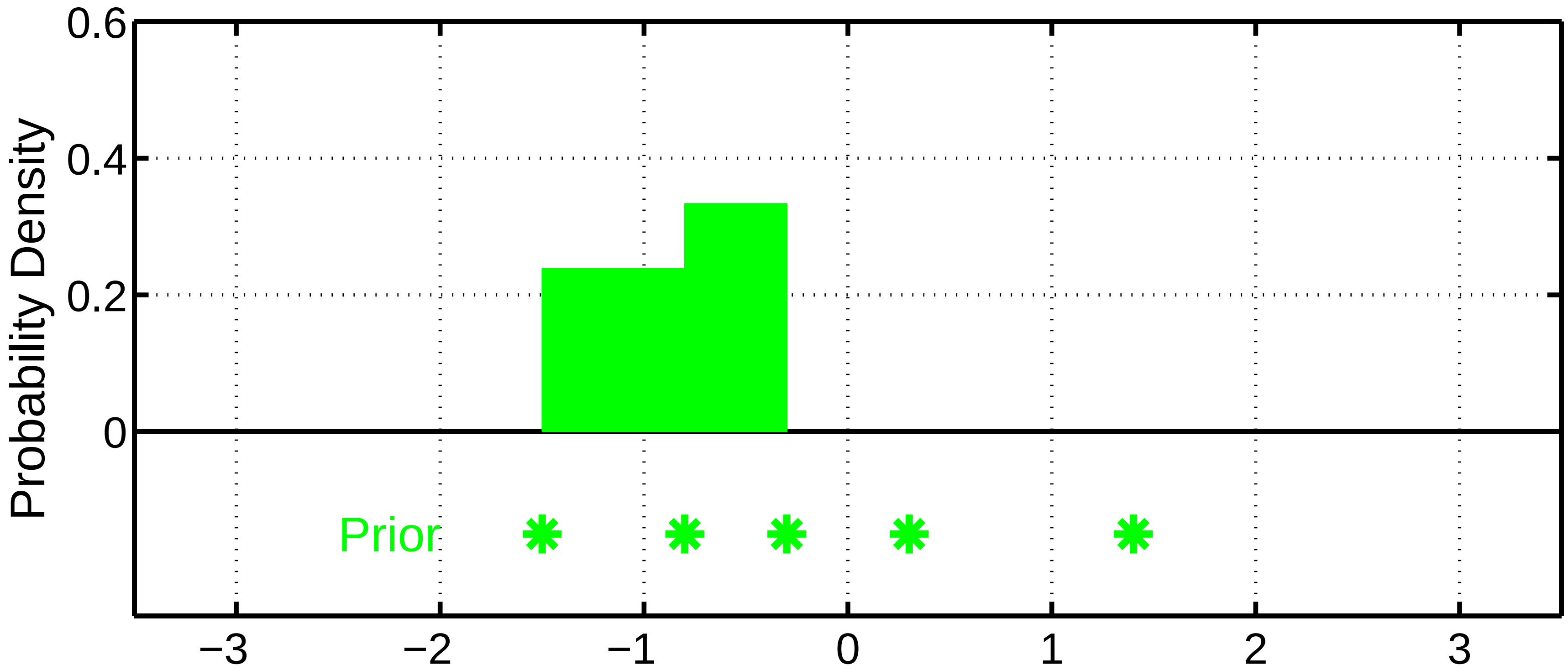


Step 1: Get continuous prior distribution density.

Place $(\text{ens\_size} + 1)^{-1}$ mass between adjacent ensemble members.

Reminiscent of rank histogram evaluation method.

Observation Space Rank Histogram Filter

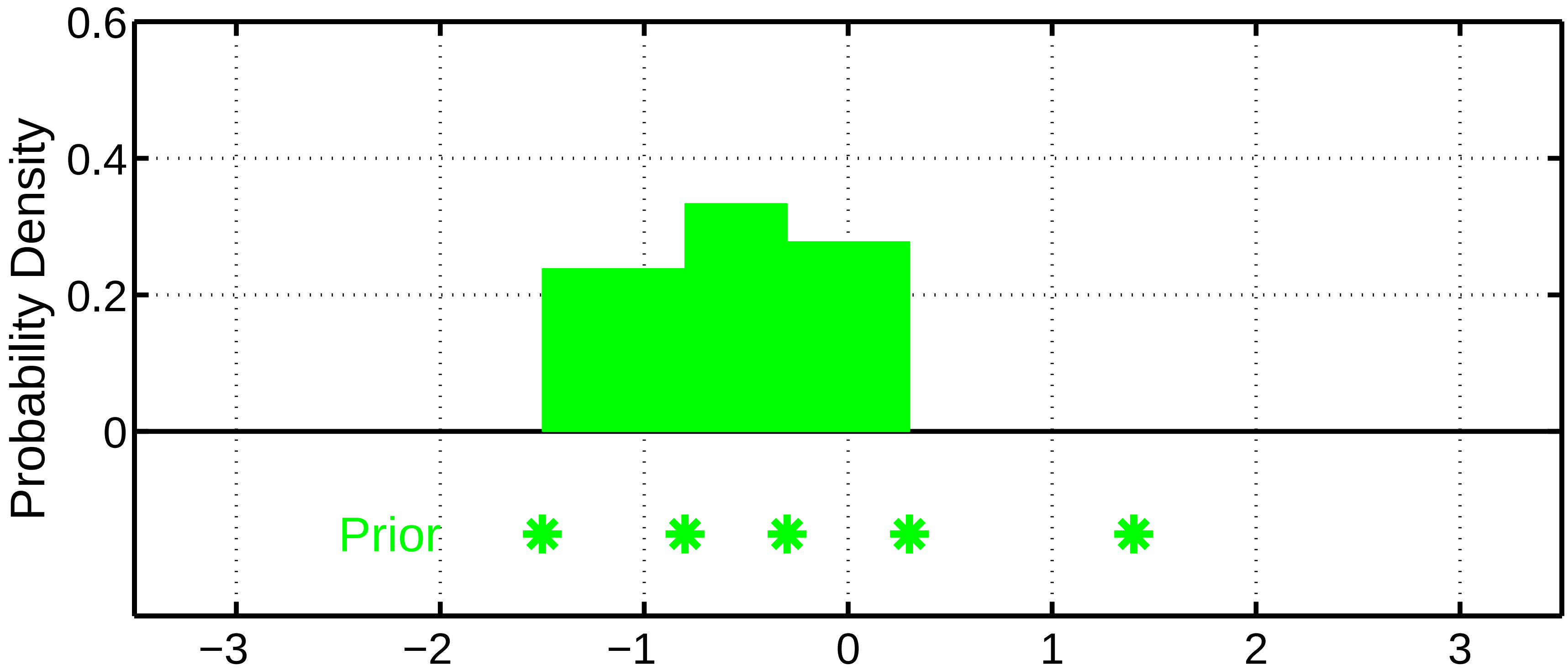


Step 1: Get continuous prior distribution density.

Place (ens_size + 1)$^{-1}$ mass between adjacent ensemble members.

Reminiscent of rank histogram evaluation method.

Observation Space Rank Histogram Filter

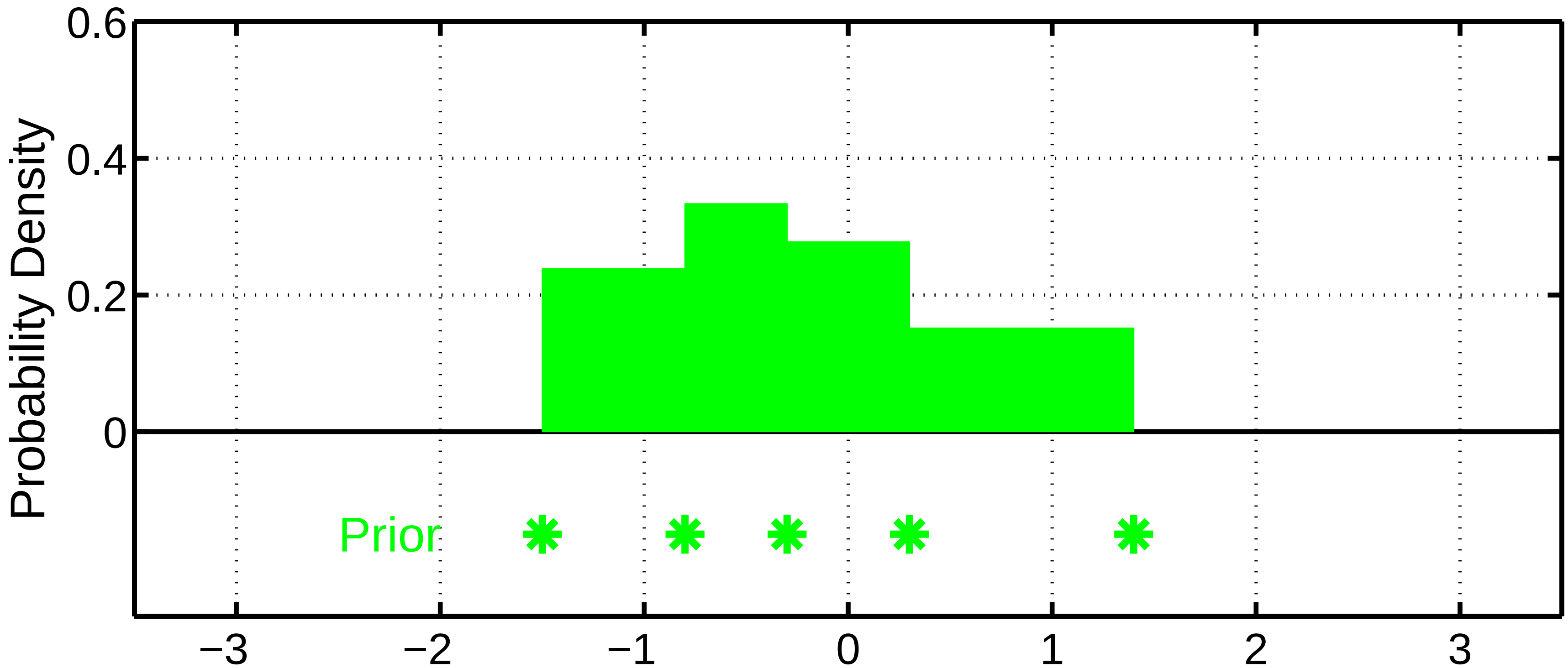


Step 1: Get continuous prior distribution density.

Place $(\text{ens\_size} + 1)^{-1}$ mass between adjacent ensemble members.

Reminiscent of rank histogram evaluation method.

Observation Space Rank Histogram Filter

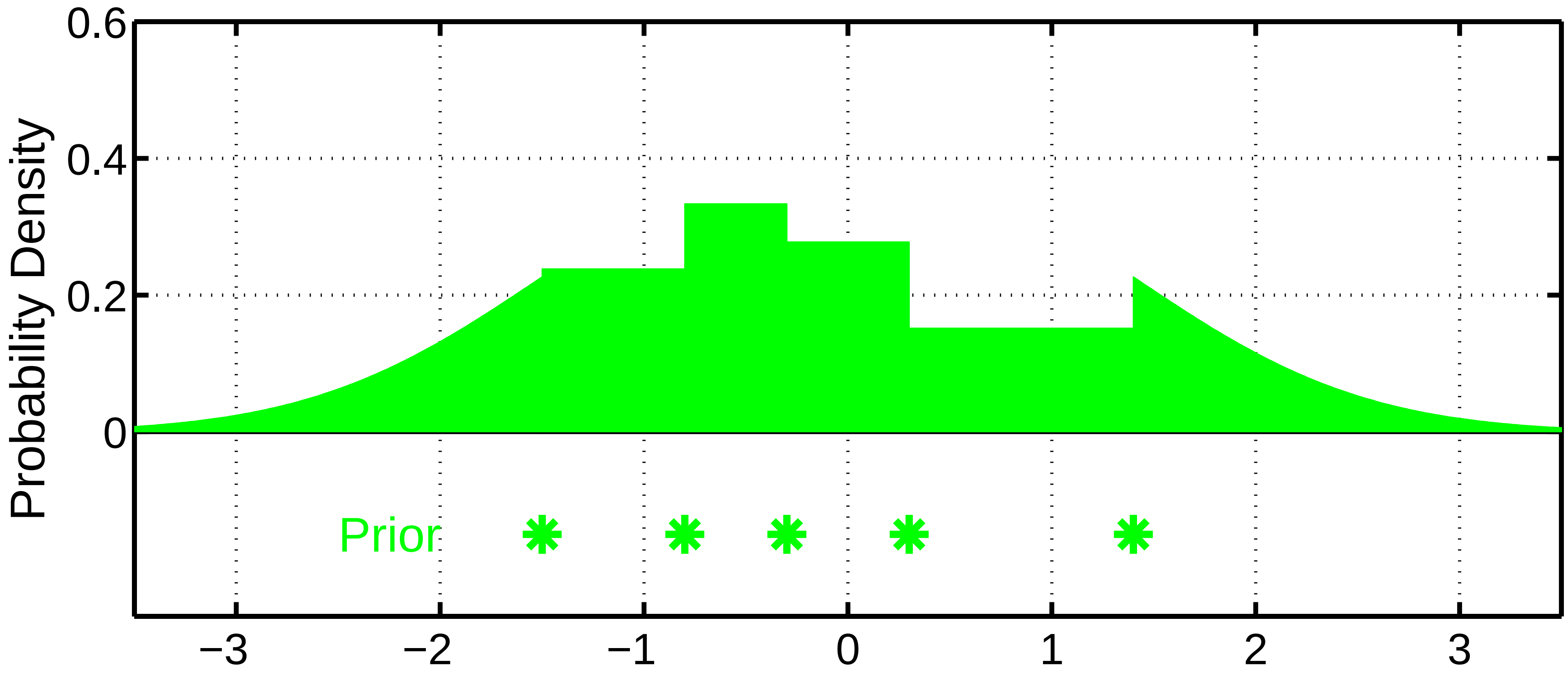


Step 1: Get continuous prior distribution density.

Place $(\text{ens\_size} + 1)^{-1}$ mass between adjacent ensemble members.

Reminiscent of rank histogram evaluation method.

Observation Space Rank Histogram Filter



Step 1: Get continuous prior distribution density.

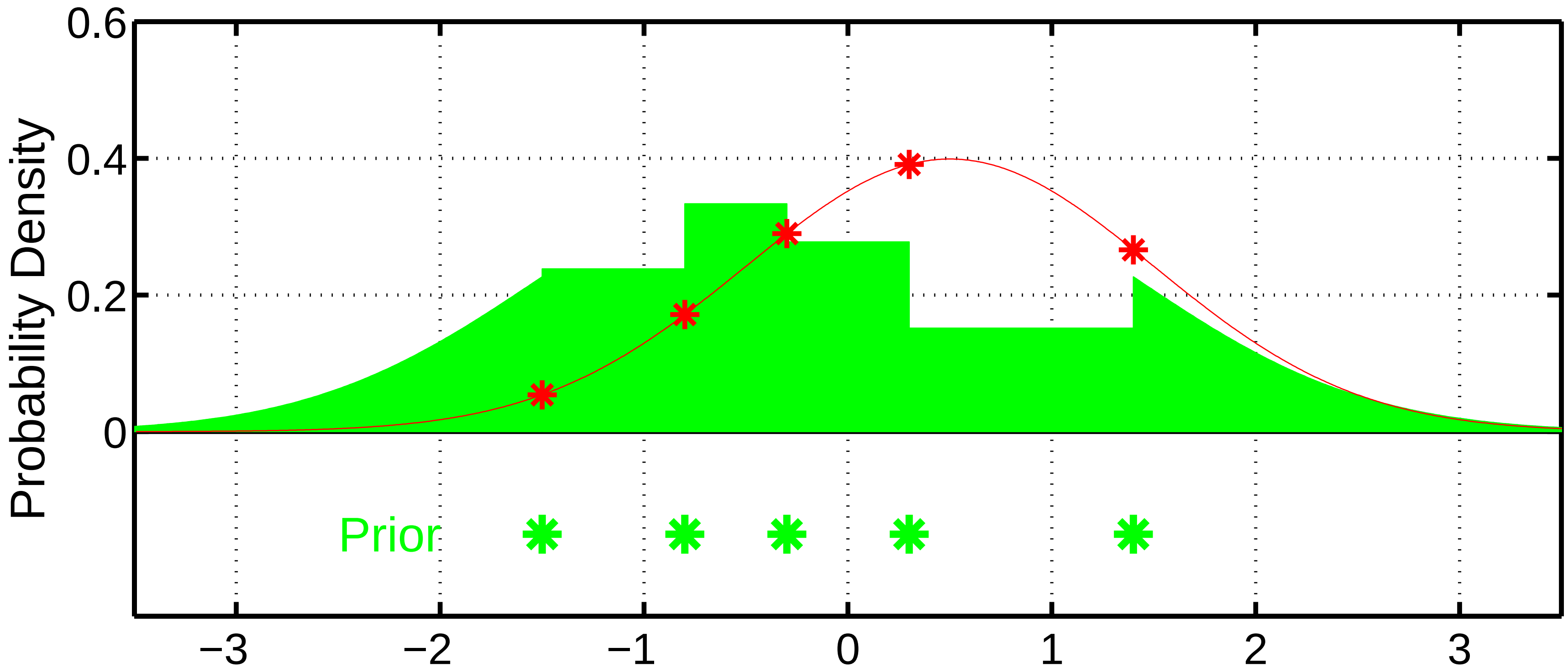
Place $(\text{ens\_size} + 1)^{-1}$ mass between adjacent ensemble members.

Partial gaussian kernels on tails, N(*tail_mean*, σ_{ens}).

tail_mean selected so that $(\text{ens\_size} + 1)^{-1}$ mass is in tail.

Performance is sensitive to the tail structure.

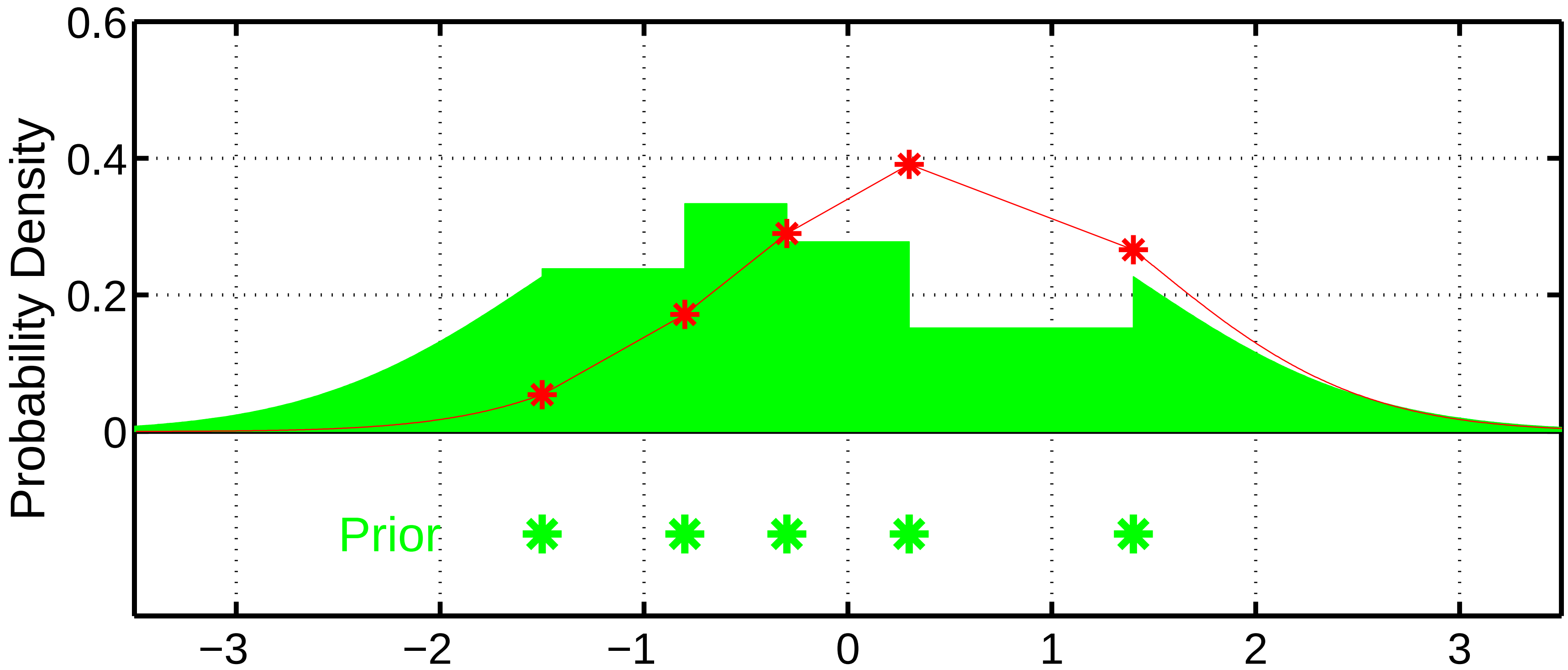
Observation Space Rank Histogram Filter



Step 2: Use likelihood to compute weight for each ensemble member.

Analogous to classical particle filter.
Can be extended to non-gaussian obs. likelihoods.

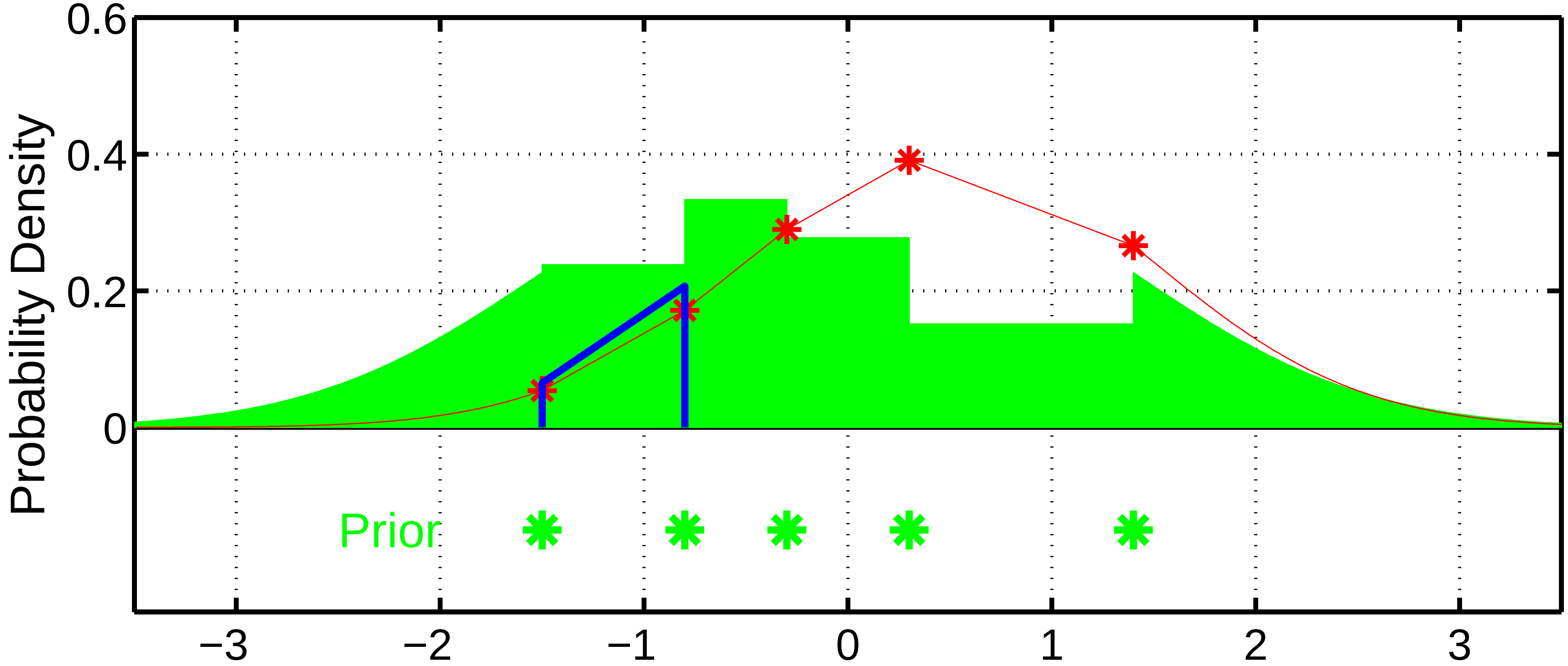
Observation Space Rank Histogram Filter



Step 2: Use likelihood to compute weight for each ensemble member.

Approximate interior likelihood with linear fit; for efficiency.
Can be extended to non-gaussian obs. likelihoods.

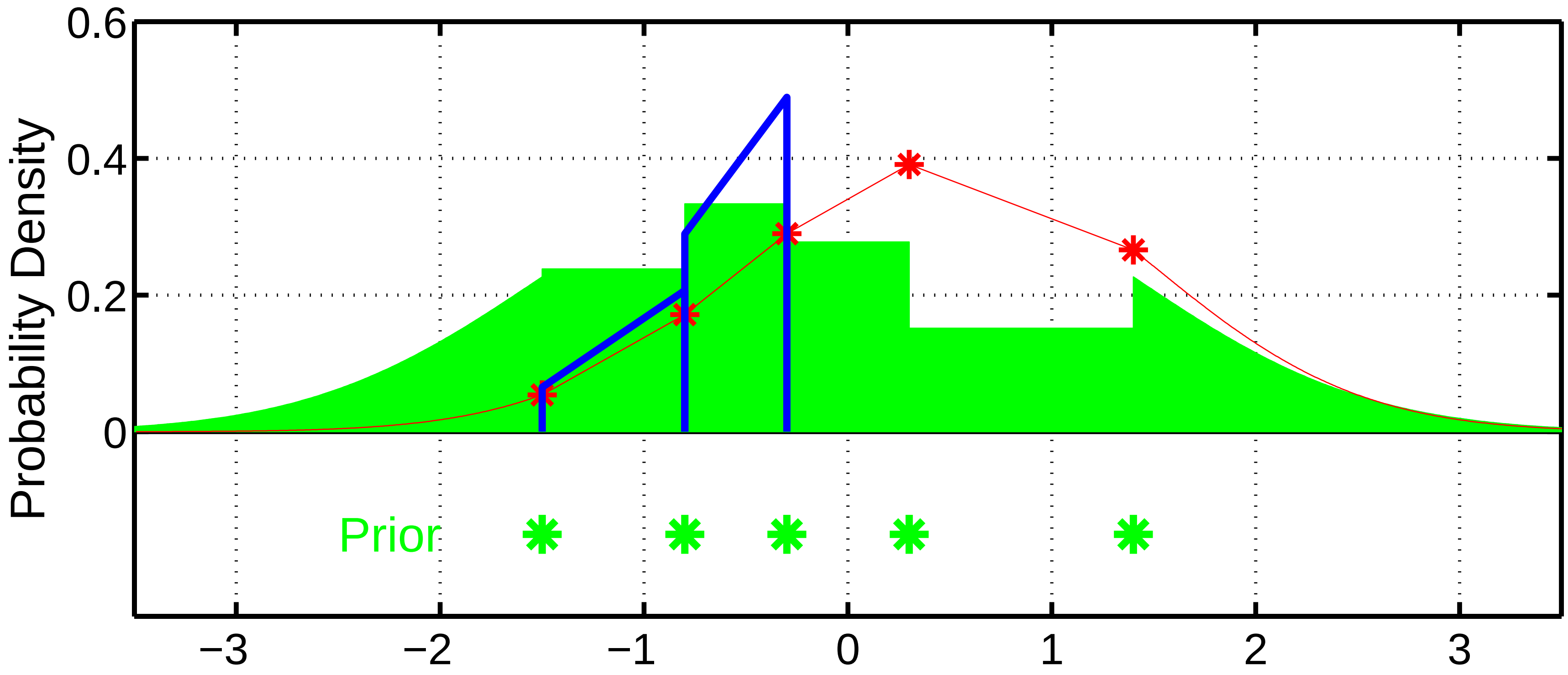
Observation Space Rank Histogram Filter



Step 3: Compute continuous posterior distribution.

1. Approximate likelihood with trapezoidal quadrature, take product.
   (Displayed product normalized to make posterior a PDF).

Observation Space Rank Histogram Filter

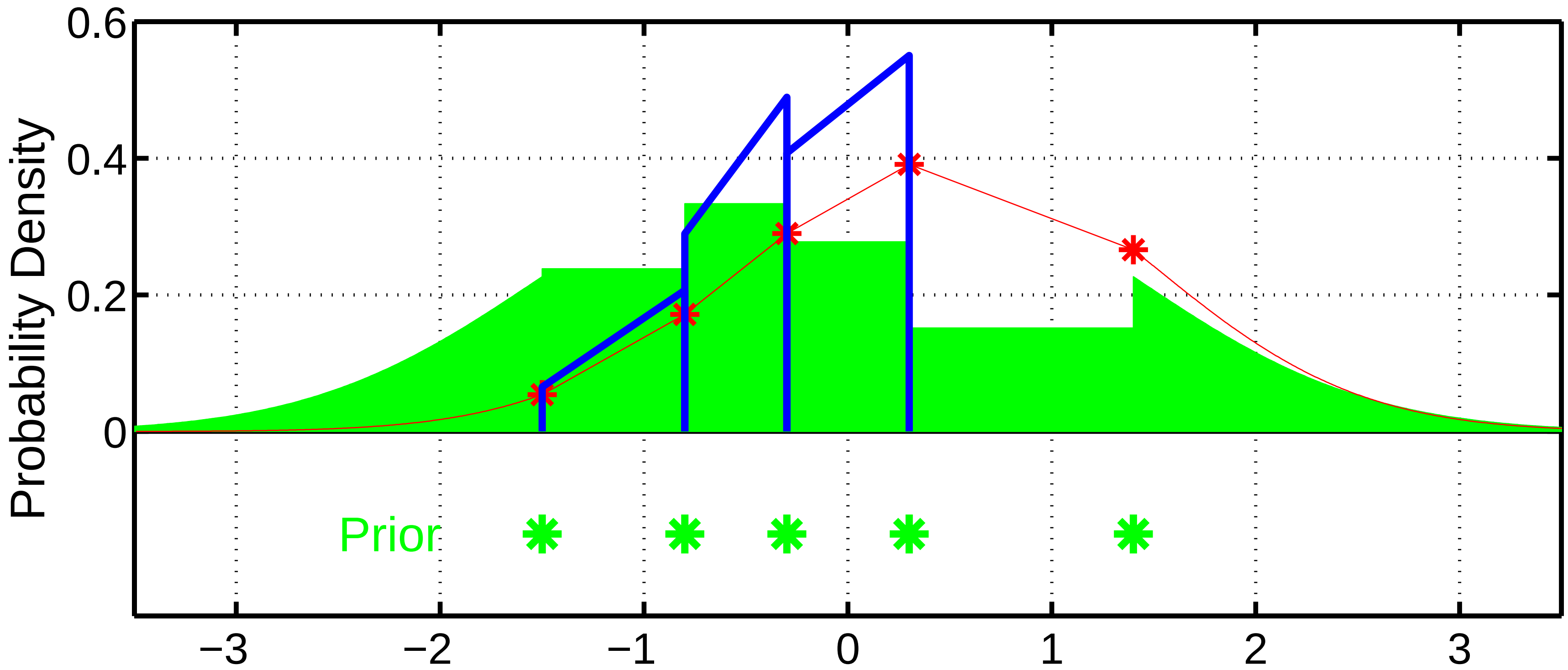


Step 3: Compute continuous posterior distribution.

1. Approximate likelihood with trapezoidal quadrature, take product.

   (Displayed product normalized to make posterior a PDF).

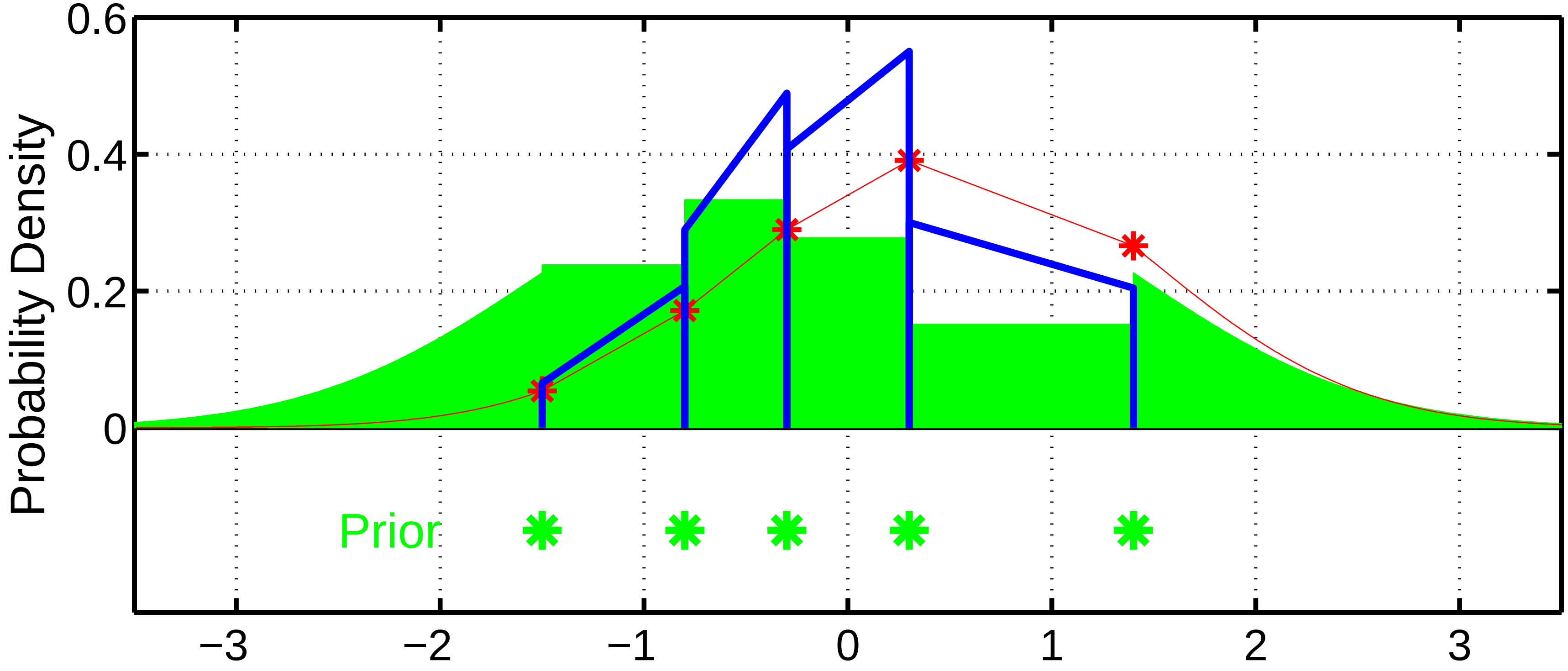
Observation Space Rank Histogram Filter



Step 3: Compute continuous posterior distribution.

1. Approximate likelihood with trapezoidal quadrature, take product.
   (Displayed product normalized to make posterior a PDF).

Observation Space Rank Histogram Filter

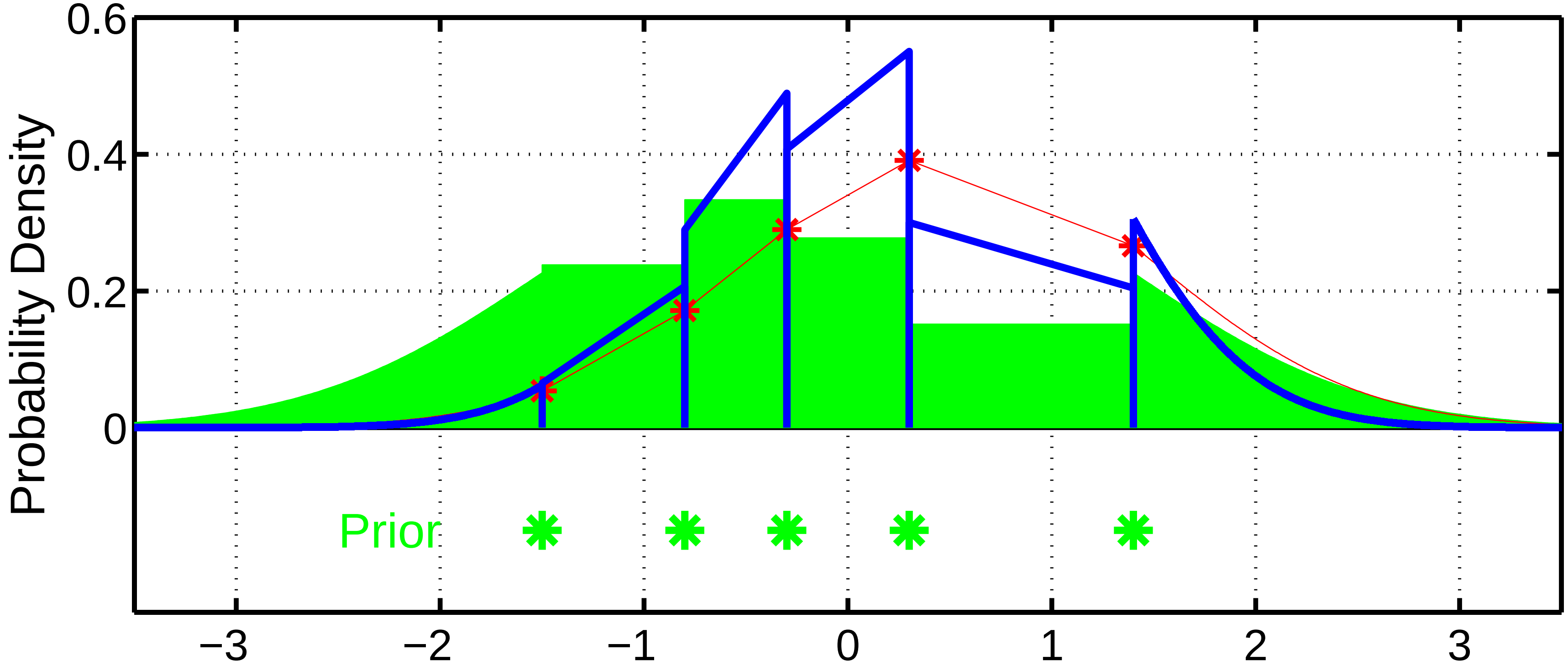


Step 3: Compute continuous posterior distribution.

1. Approximate likelihood with trapezoidal quadrature, take product.

(Displayed product normalized to make posterior a PDF).

Observation Space Rank Histogram Filter



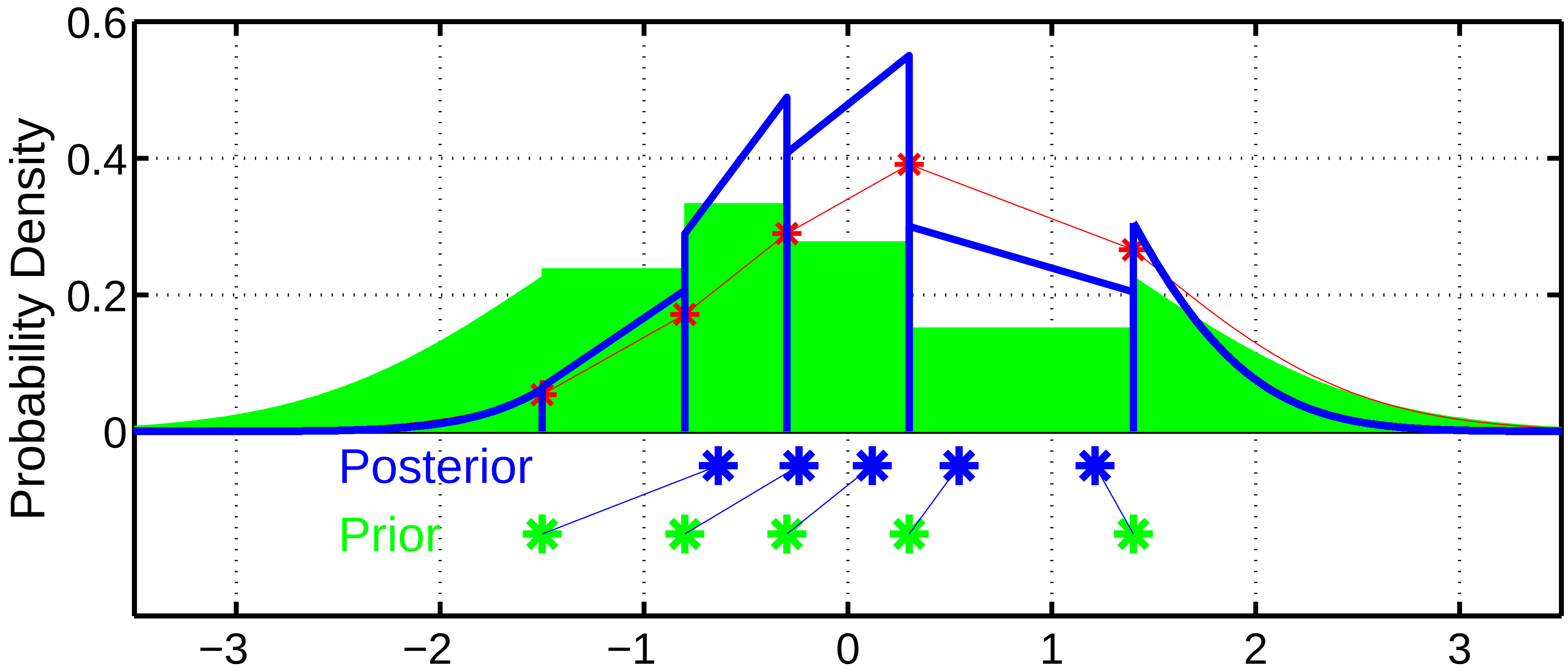
Step 3: Compute continuous posterior distribution.

1. Approximate likelihood with trapezoidal quadrature, take product.
2. Product of prior gaussian kernel with likelihood for tails.

    Easy for gaussian likelihood.

    More quadrature if non-Gaussian likelihood.

Observation Space Rank Histogram Filter



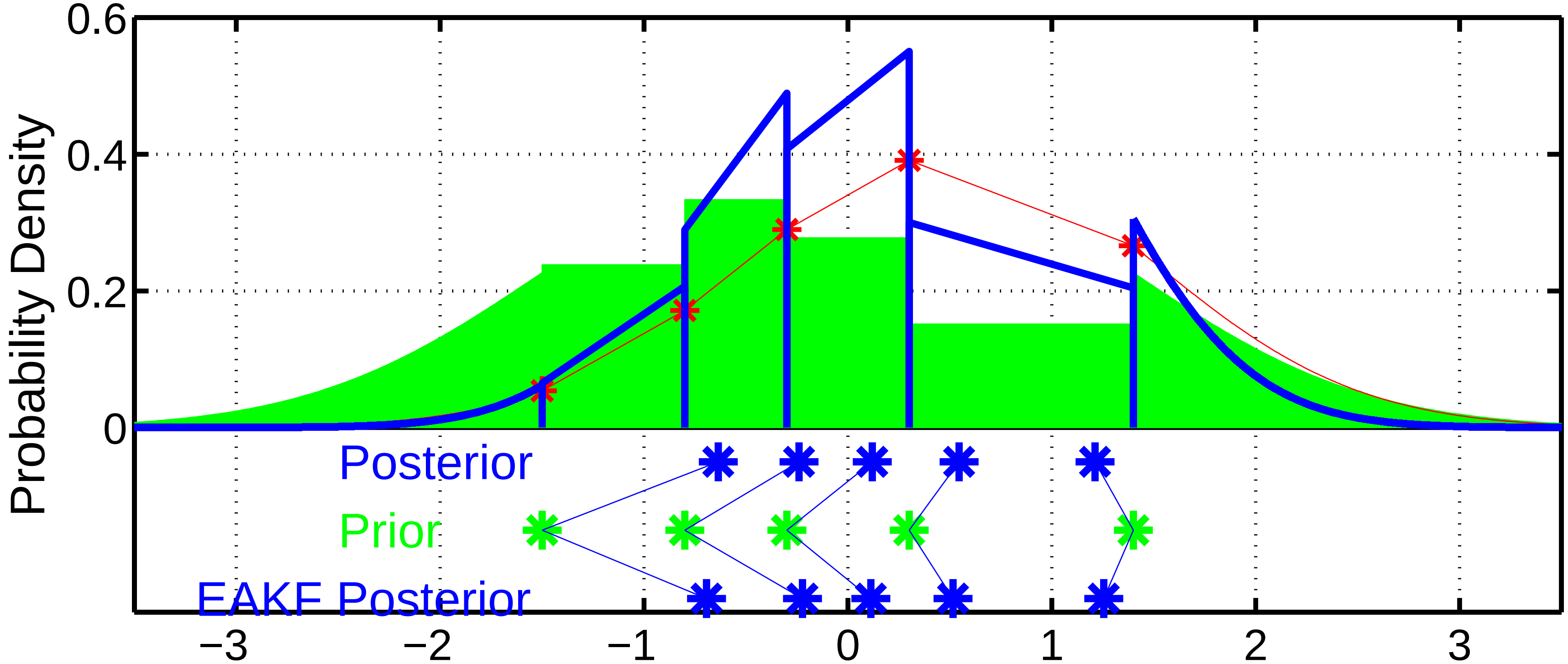
Step 4: Compute updated ensemble members:

(ens_size + 1)$^{-1}$ of posterior mass between each ensemble pair.

(ens_size + 1)$^{-1}$ in each tail.

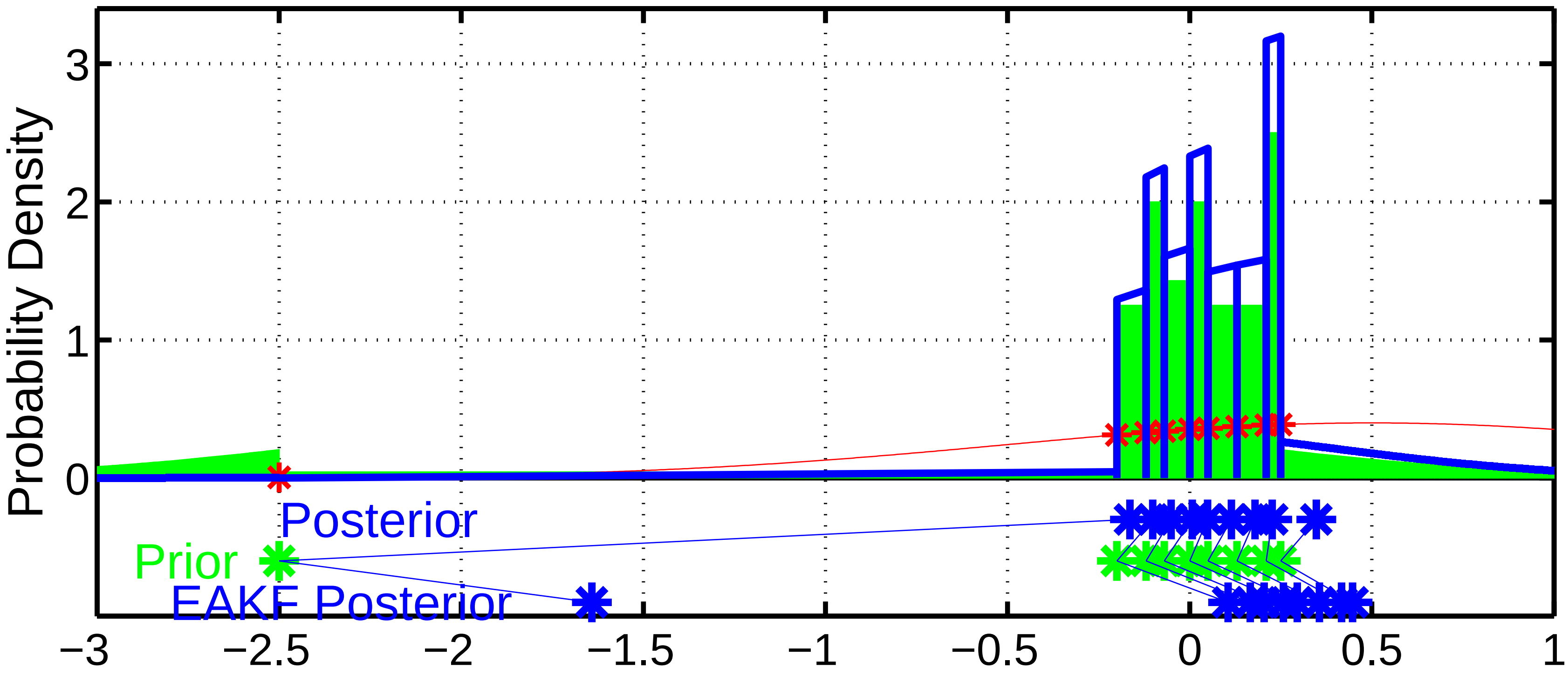
Uninformative observation has no impact.

Observation Space Rank Histogram Filter



Compare to standard Ensemble Adjustment Filter (EAKF).
Nearly gaussian case, differences are small.

Outliers are a Challenge for Gaussian Filters

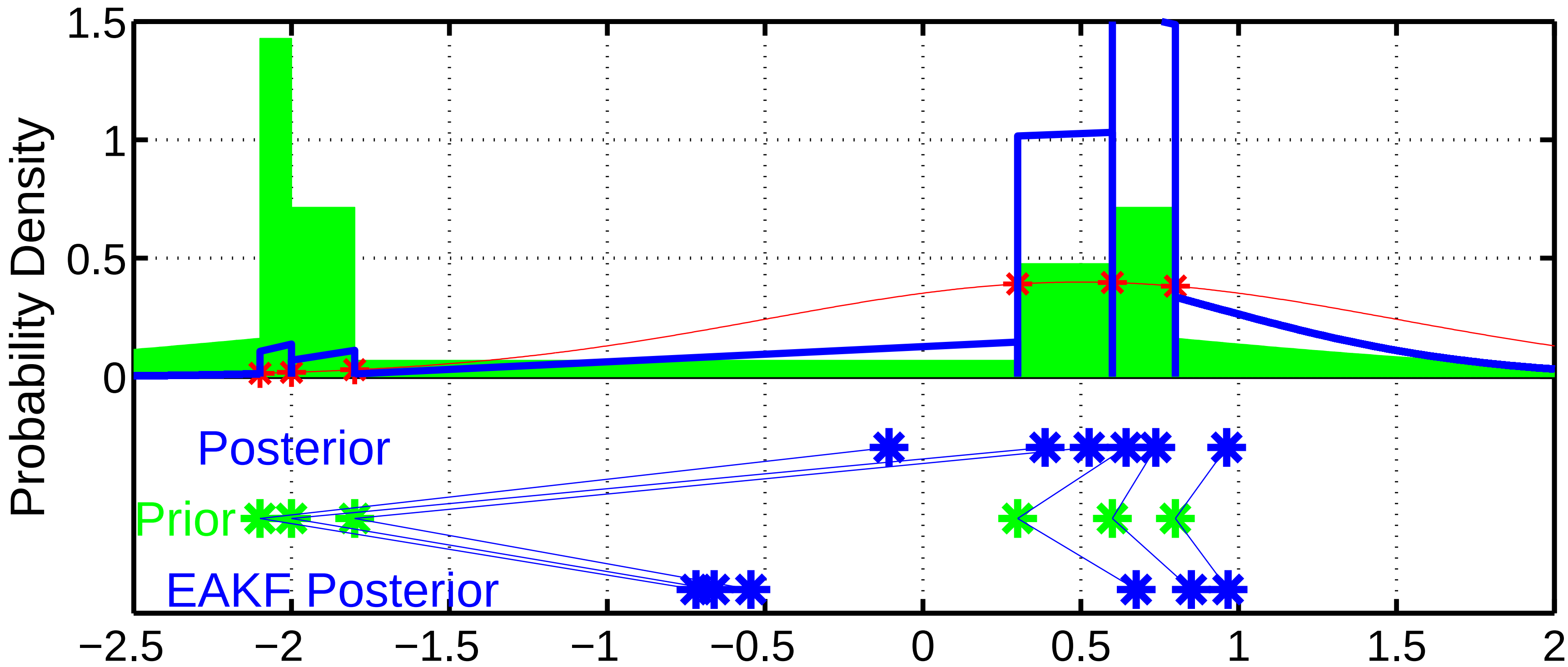


Rank Histogram gets rid of outlier that is clearly inconsistent with obs.

EAKF can't get rid of outlier.

Large prior variance from outlier causes EAKF to shift all members too much towards observation.

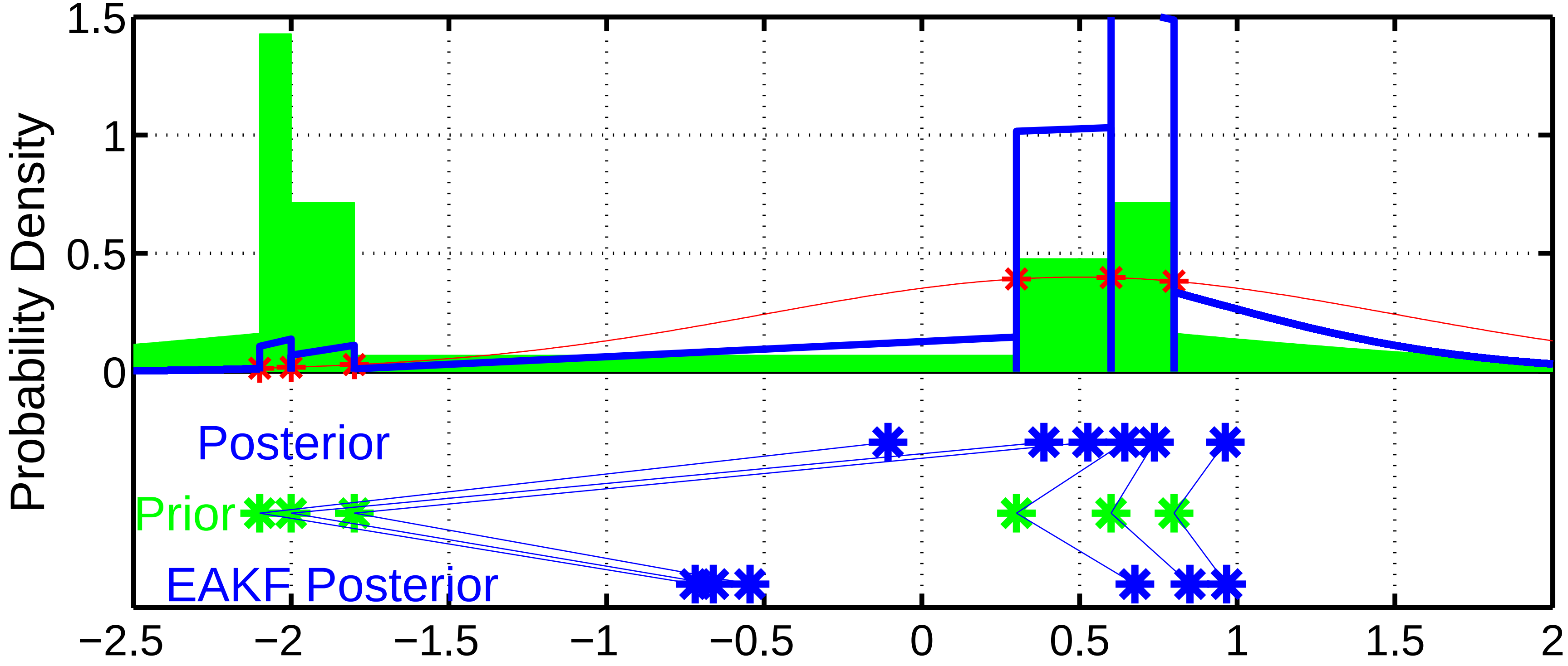
Multimodal Prior Distributions



Rank Histogram handles multimodal prior and compelling observation.
EAKF still bimodal; left mode is inconsistent with everything.

Lorenz_63 can have priors like this.

Multimodal Prior Distributions



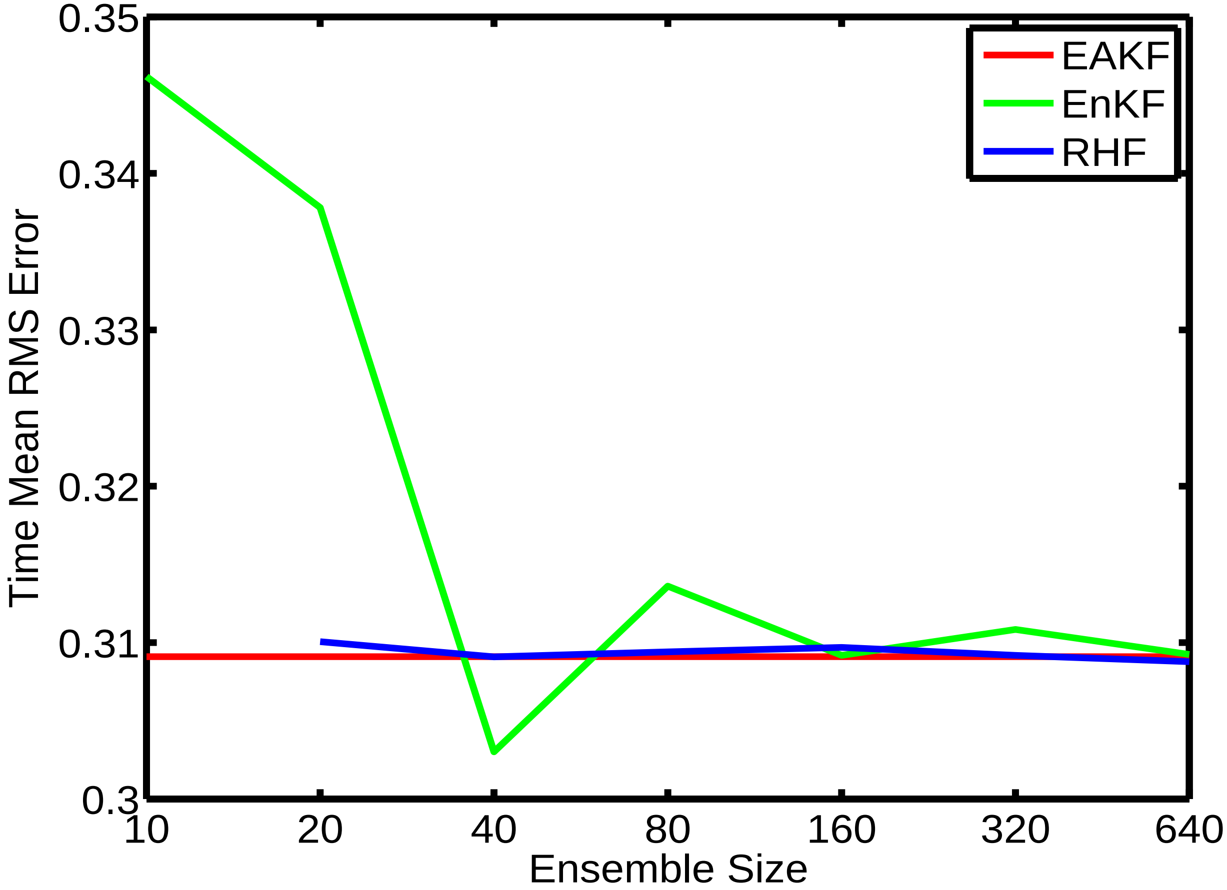
Convective scale models have analogous behavior.

Convection may fire at ‘random’ locations.

Subset of ensembles will be in right place, rest in wrong place.

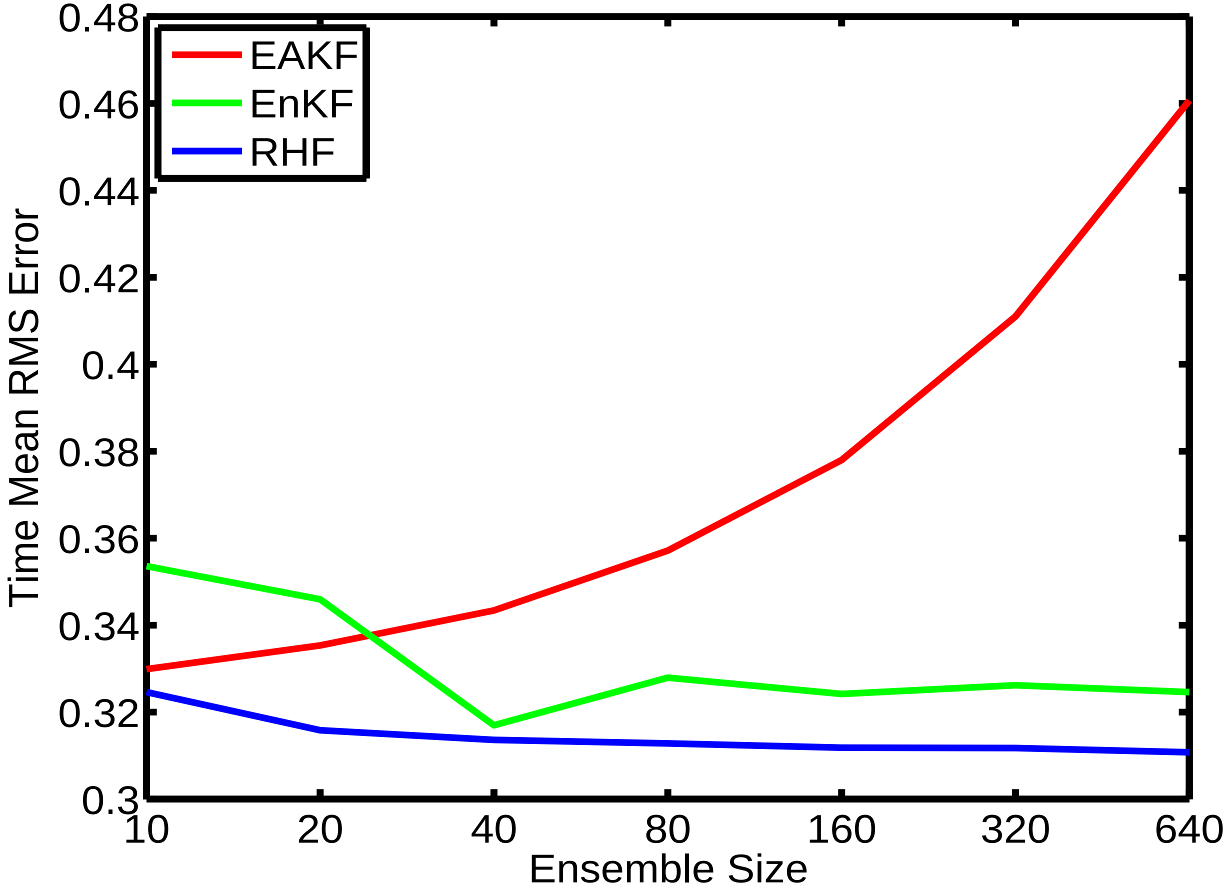
Want to aggressively eliminate convection in wrong place.

Results: Linear Model ($\alpha = 0$)



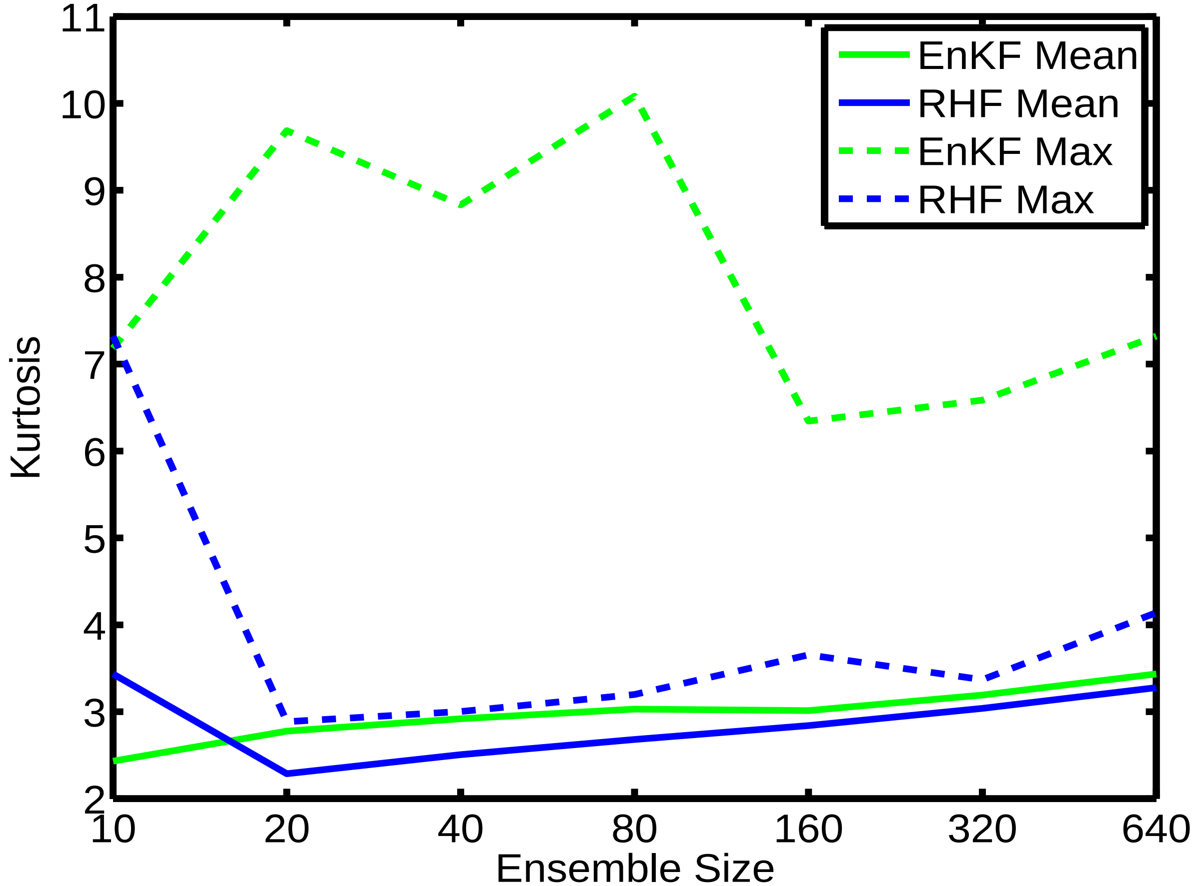
Rank Histogram Filter (RHF) fails for 10 members.
Competitive for >20 members.

Results: Nonlinear Model RMS ($\alpha = 0.2$)



RHF best for all ensemble sizes.

Results: Nonlinear Model Kurtosis (α = 0.2)



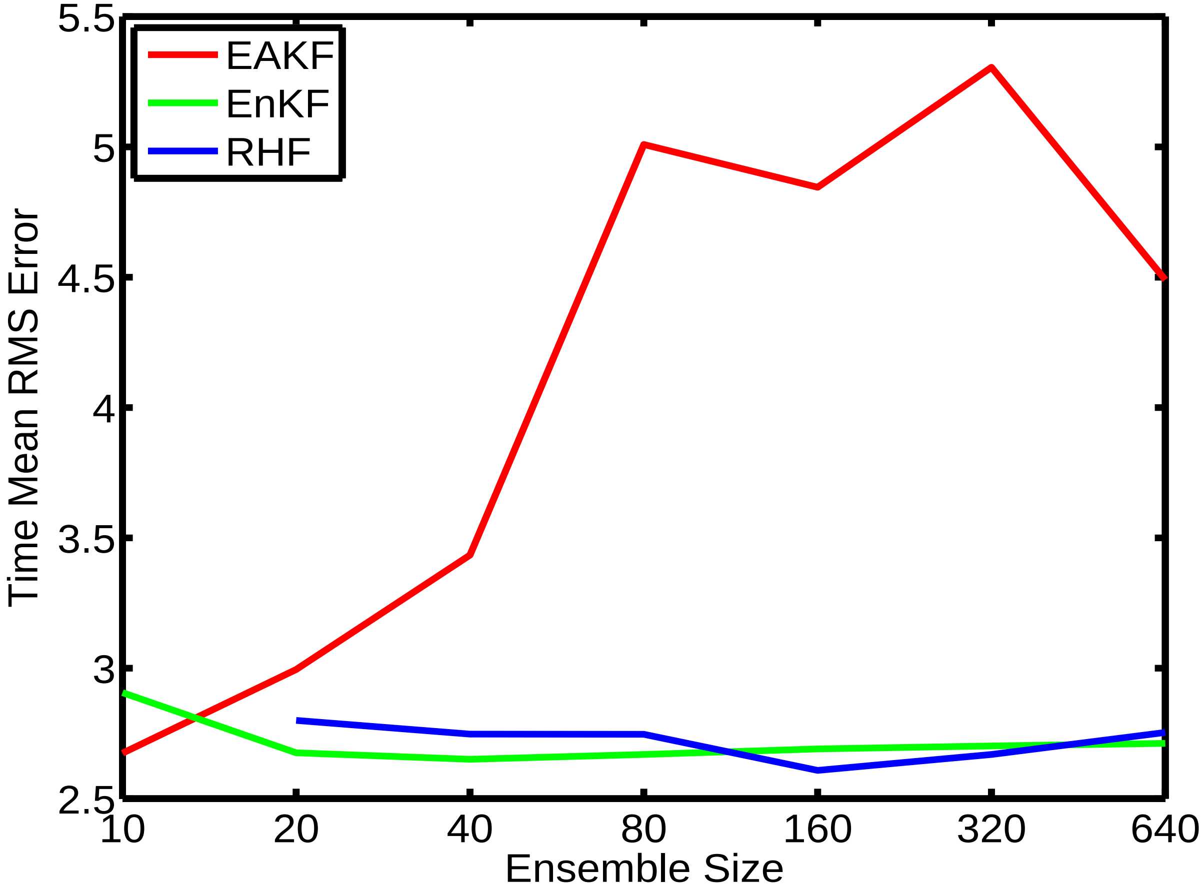
RHF smallest for ensemble sizes > 10.

Doesn’t have outlier excursions (max is small).

EAKF has HUGE kurtosis (off the plot).

Results: Lorenz63 RMS

All 3 state variables observed, error variance 1.0

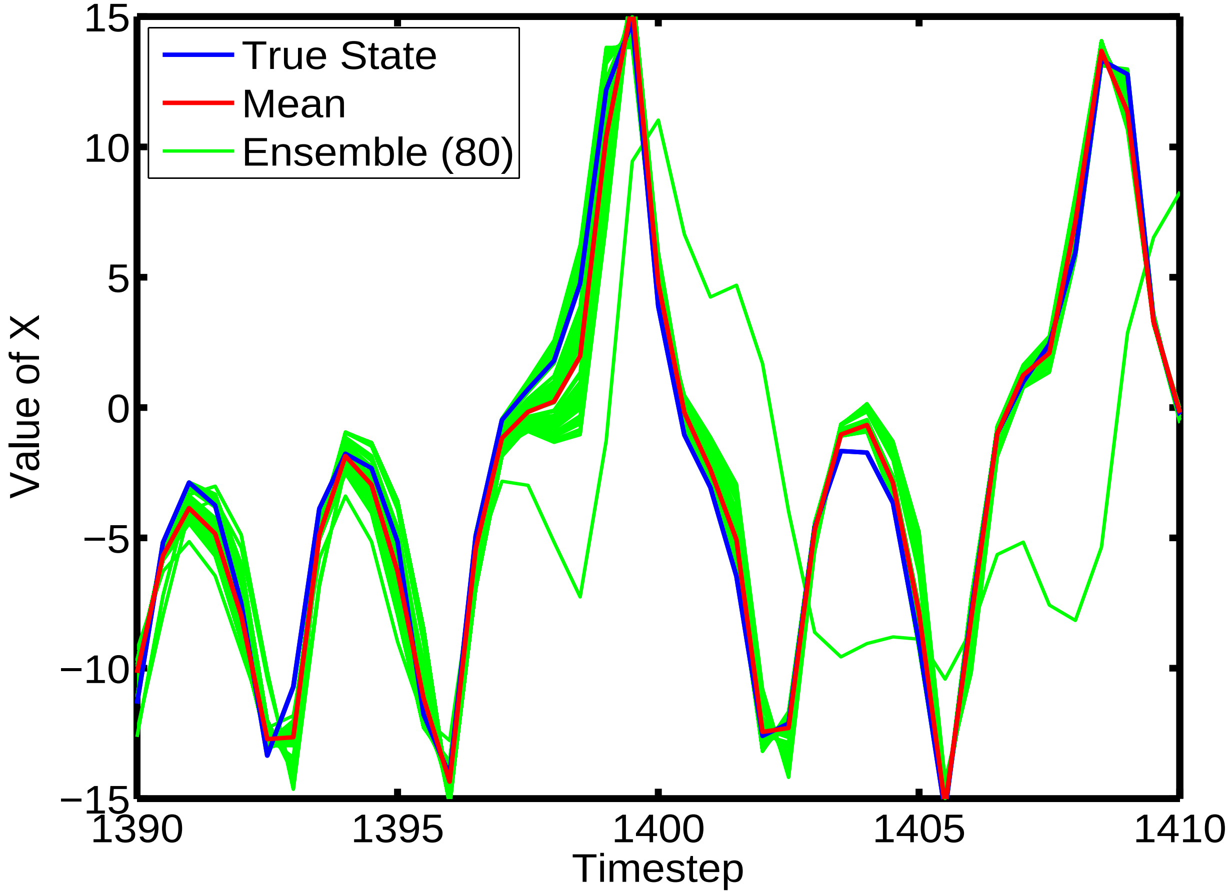


RHF and EnKF comparable.

EAKF gets progressively worse (but pretty good for 10 members).

Results: Lorenz63 EAKF

All 3 state variables observed, error variance 1.0



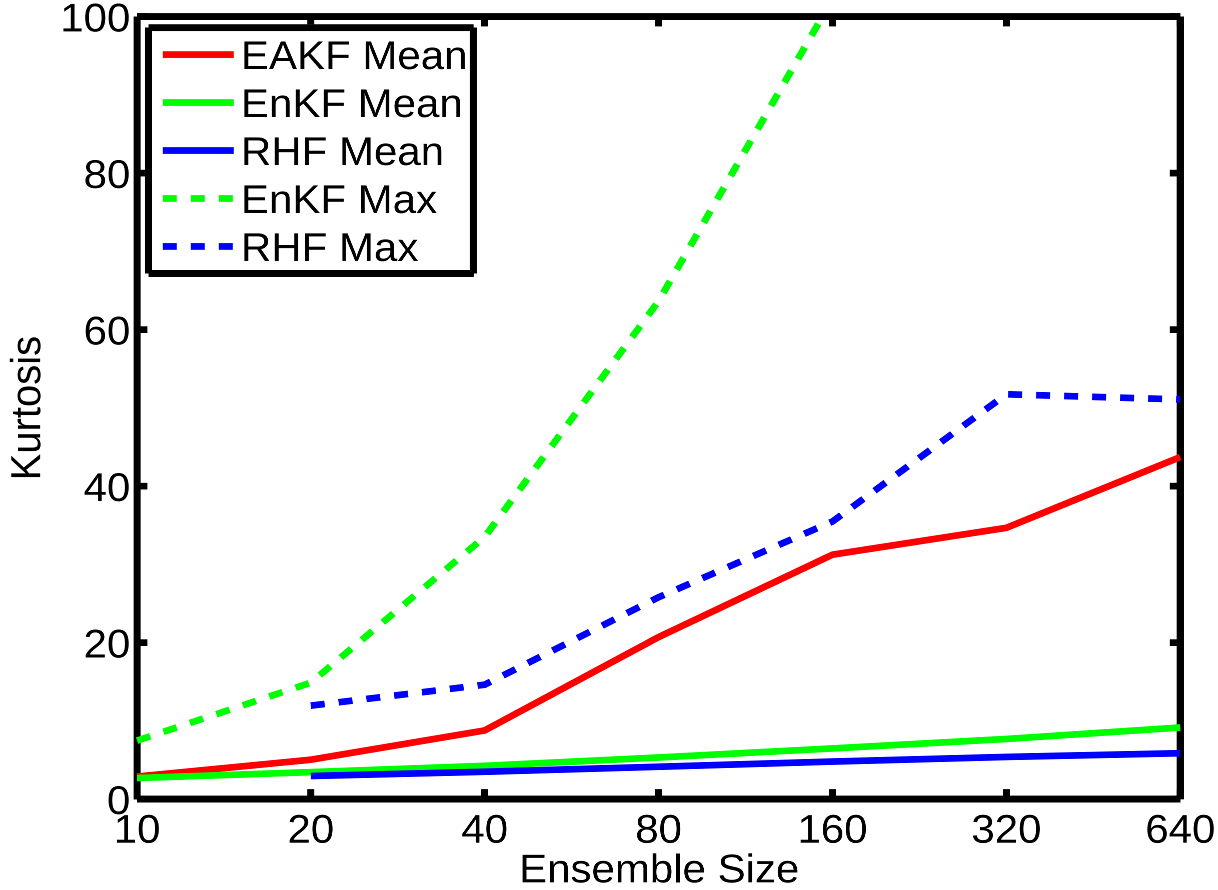
Wandering ensemble member can detach, head into wrong lobe.

Happens less frequently and severely in EnKF.

Can reattach due to mixing from other variables.

Results: Lorenz63 Kurtosis

All 3 state variables observed, error variance 1.0



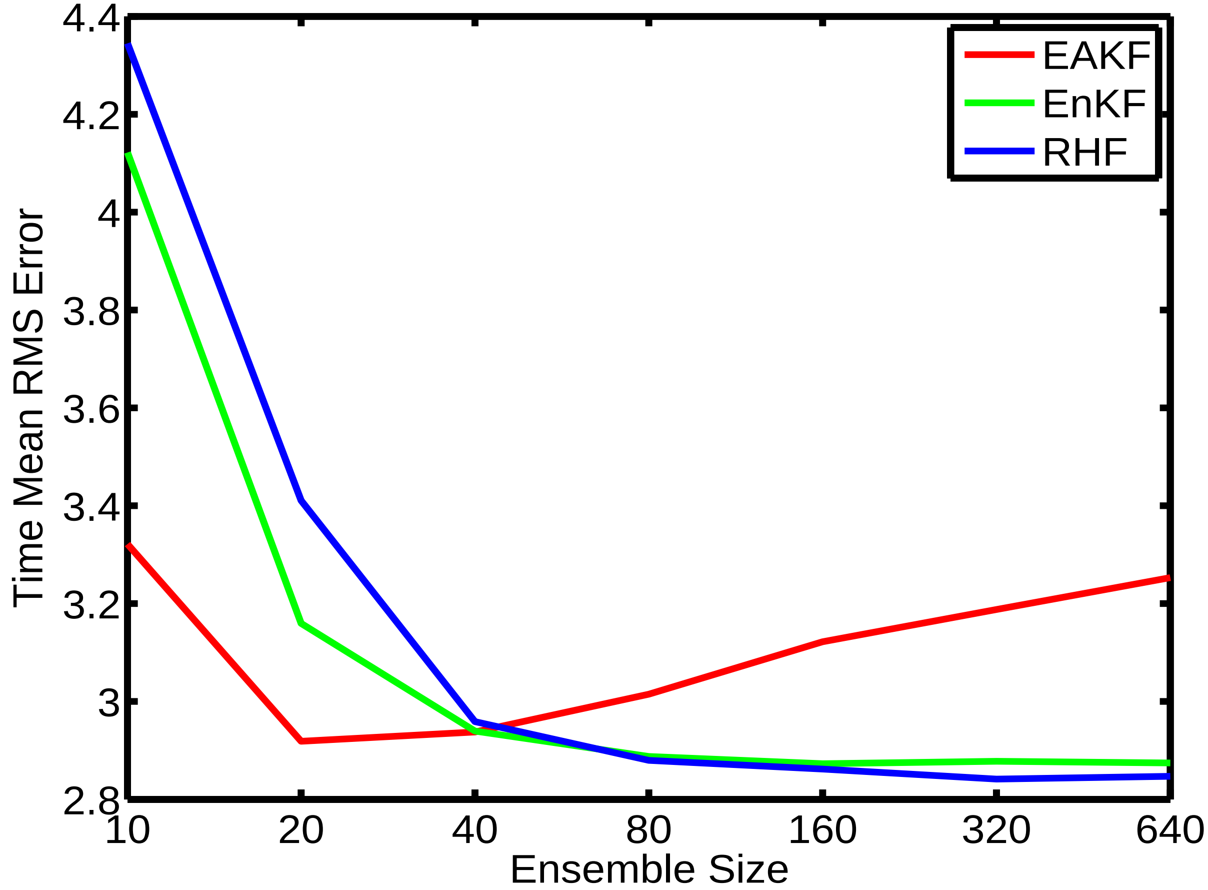
Wandering ensemble member can detach, head into wrong lobe.

Happens less frequently and severely in EnKF.

Can reattach due to mixing from other variables.

Results: Lorenz96 RMS

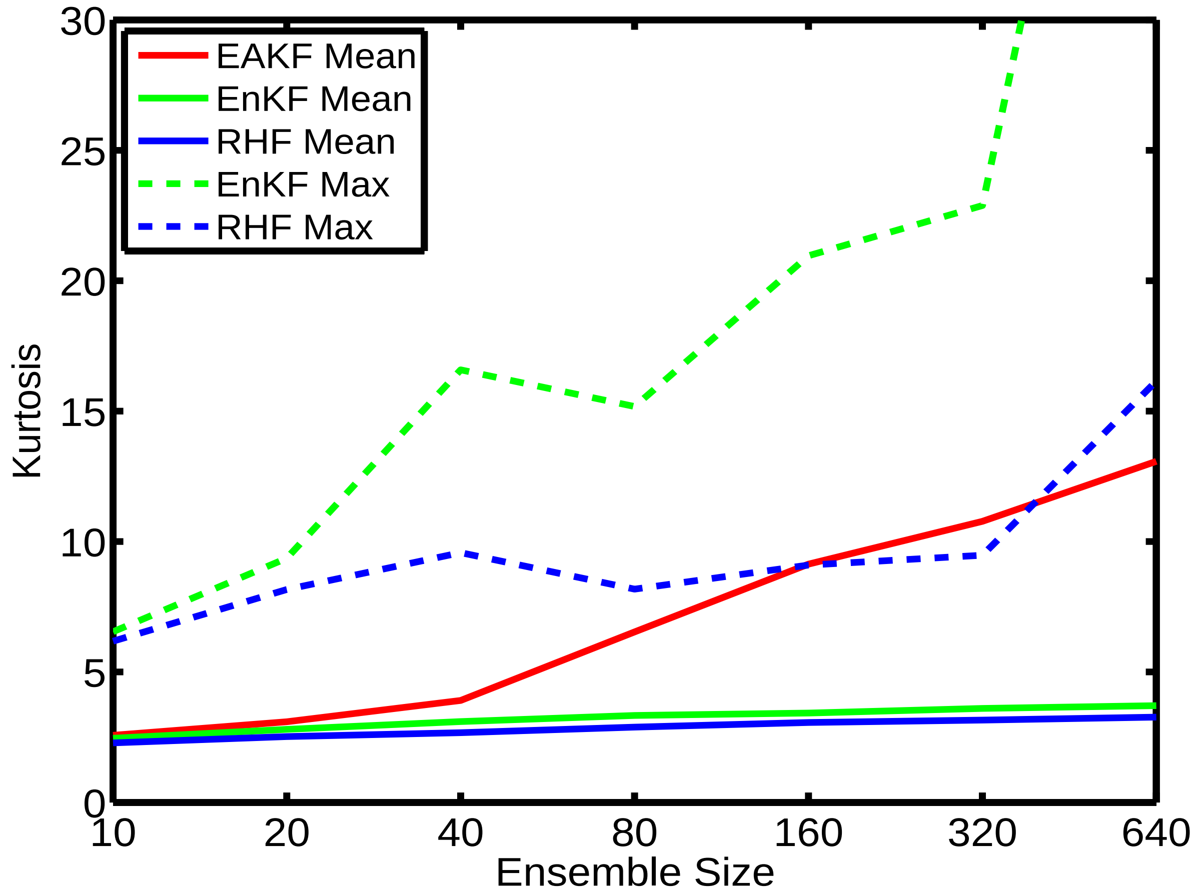
40 Observations, average of adjacent state variables, Error var = 4.
Localization halfwidth 0.3 of domain, adaptive inflation.



EAKF RMS increases moderately with ensemble size.

Results: Lorenz96 Kurtosis

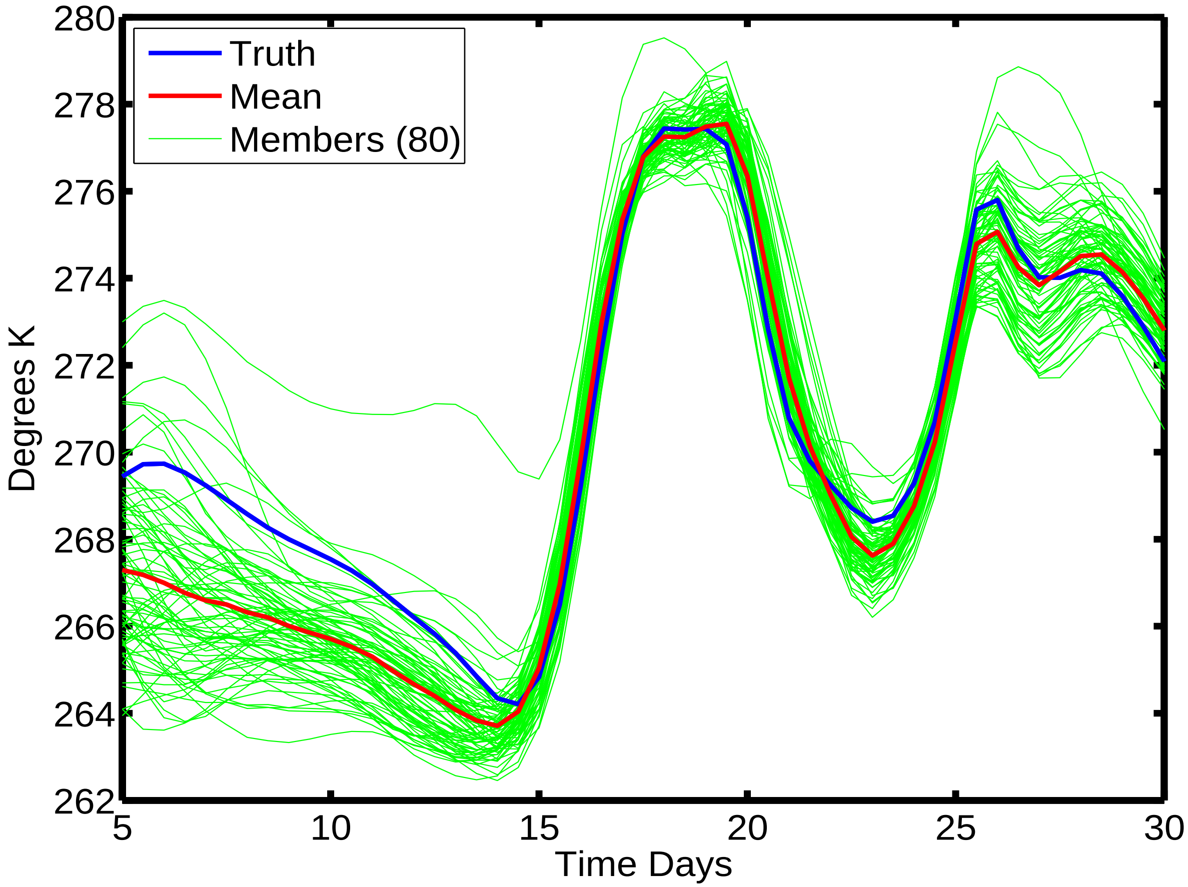
40 Observations, average of adjacent state variables, Error var = 4.
Localization halfwidth 0.3 of domain, adaptive inflation.



EnKF has sporadic large kurtosis, increases with ensemble size.
EAKF max kurtosis very large (off plot).

Results: Dynamical Core of GFDL AM2 GCM

1. EAKF: sporadic spatially localized outlier behavior.



Lowest level T 30W, 50N.

300 radiosonde profiles every 12 hours.

No inflation.

0.2 radian localization.

2. EnKF: similar behavior less frequently.

3. RHF: no evidence this occurs.

Results: Global NWP in Finite Volume CAM

1. Limited evidence of outlier events in any filter.

2. Prior fit to observations:
   80-member EAKF and RHF virtually indistinguishable.
      (Comparable to NCEP operational, better in tropics, near sfc.).
   80-member EnKF significantly worse.

Additional Capabilities of RHF

1. Observations with highly non-gaussian observation likelihoods:
   Bounded quantities like RH, precip., or reflectivity,
   Just need to evaluate likelihood at prior locations (caveat tails).

2. Priors that are highly non-gaussian:
   Non-linear forward operators like radiances.

3. Ability to deal with discrete structure priors:
   Example: Convective scale.
      Subset of priors may have convection in a given location.
      Posterior should be either yes or no, not maybe.

RHF and particle filters

1. The RHF is nearly a particle filter with resampling.
   Given likelihood weights can do everything except tails.

2. Is there a way to make an effective filter that ONLY uses weights?

3. If so, could get rid of regression and just use weights for state vars.

4. It's known that particle filters don't scale to high dimensions.

5. With localization, this idea might scale.

Want to try it out?

The Rank Histogram Filter and 7 other ensemble update variants are in DART.

www.image.ucar.edu/DAReS/DART.