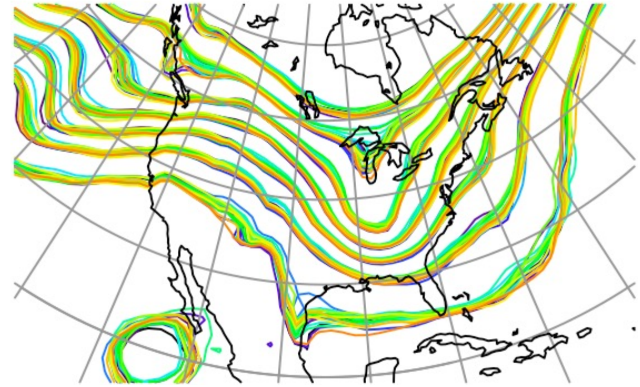
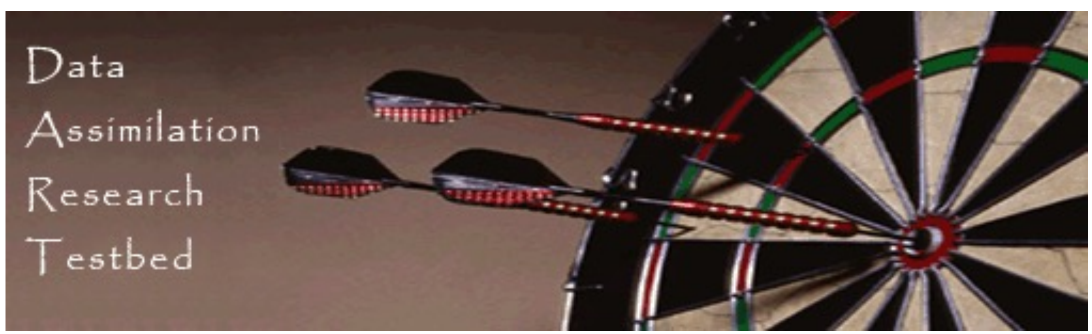


Non-Gaussian, Nonlinear Extensions for Ensemble Filter Data Assimilation with a Marginal Adjustment Rank Histogram Filter

Jeff Anderson, NCAR Data Assimilation Research Section



Schematic of a Sequential Ensemble Filter

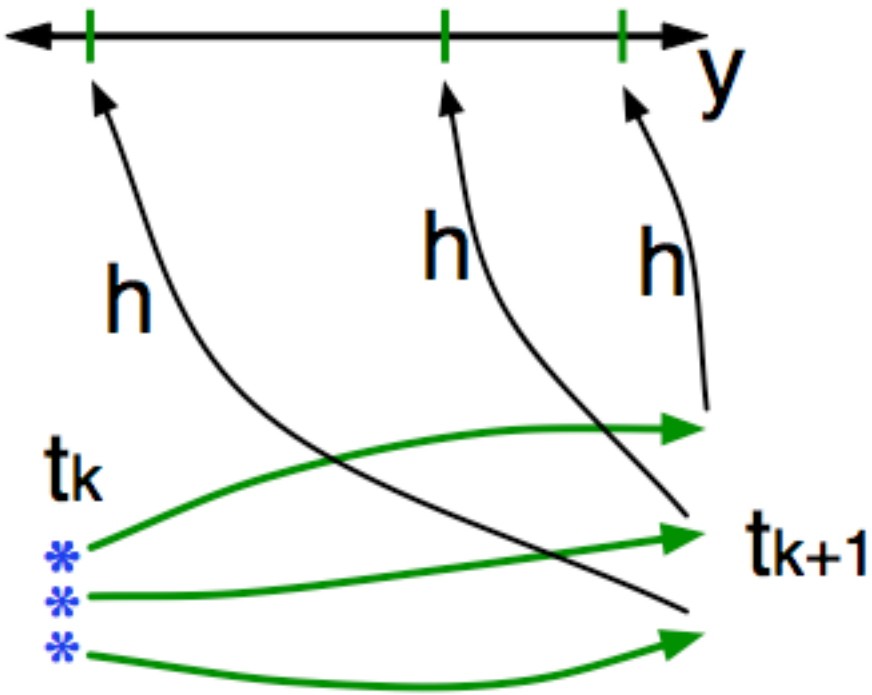
1. Use model to advance ensemble (3 members here) to time at which next observation becomes available.

Ensemble state estimate after using previous observation (analysis)

Ensemble state at time of next observation (prior)

Schematic of a Sequential Ensemble Filter

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

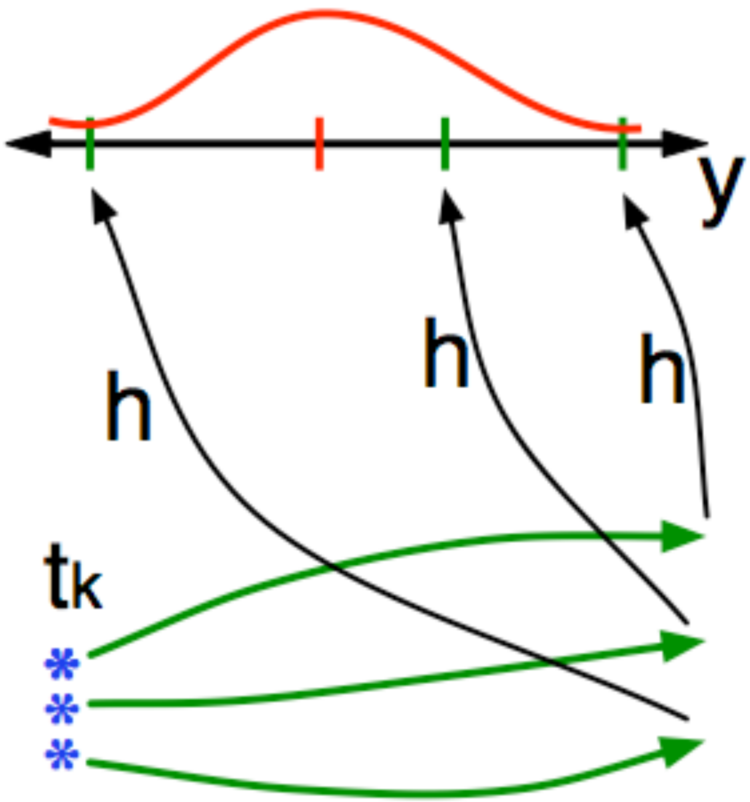


Theory: observations from instruments with uncorrelated errors can be done sequentially.

Can think about single observation without (too much) loss of generality.

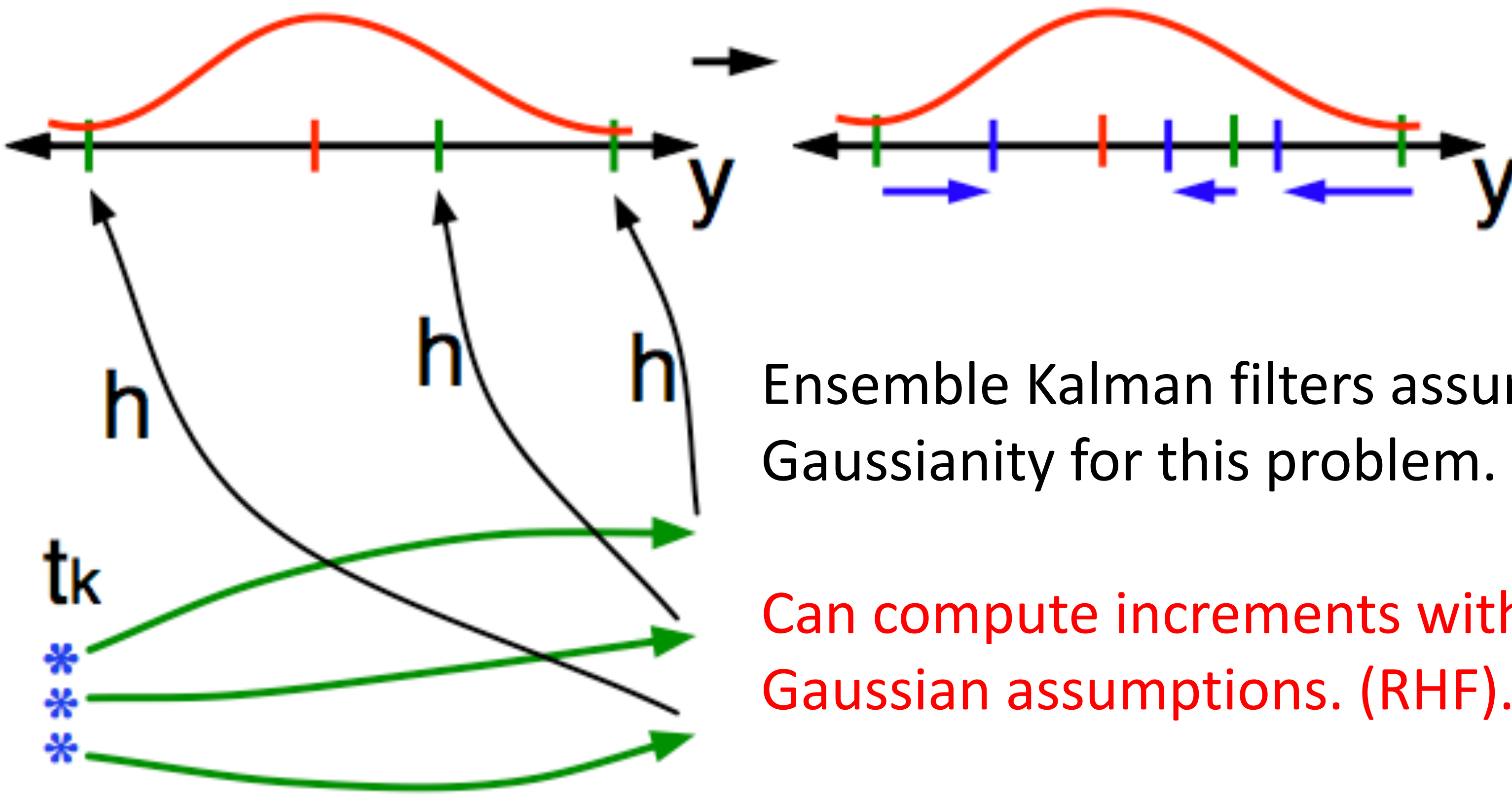
Schematic of a Sequential Ensemble Filter

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



Schematic of a Sequential Ensemble Filter

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

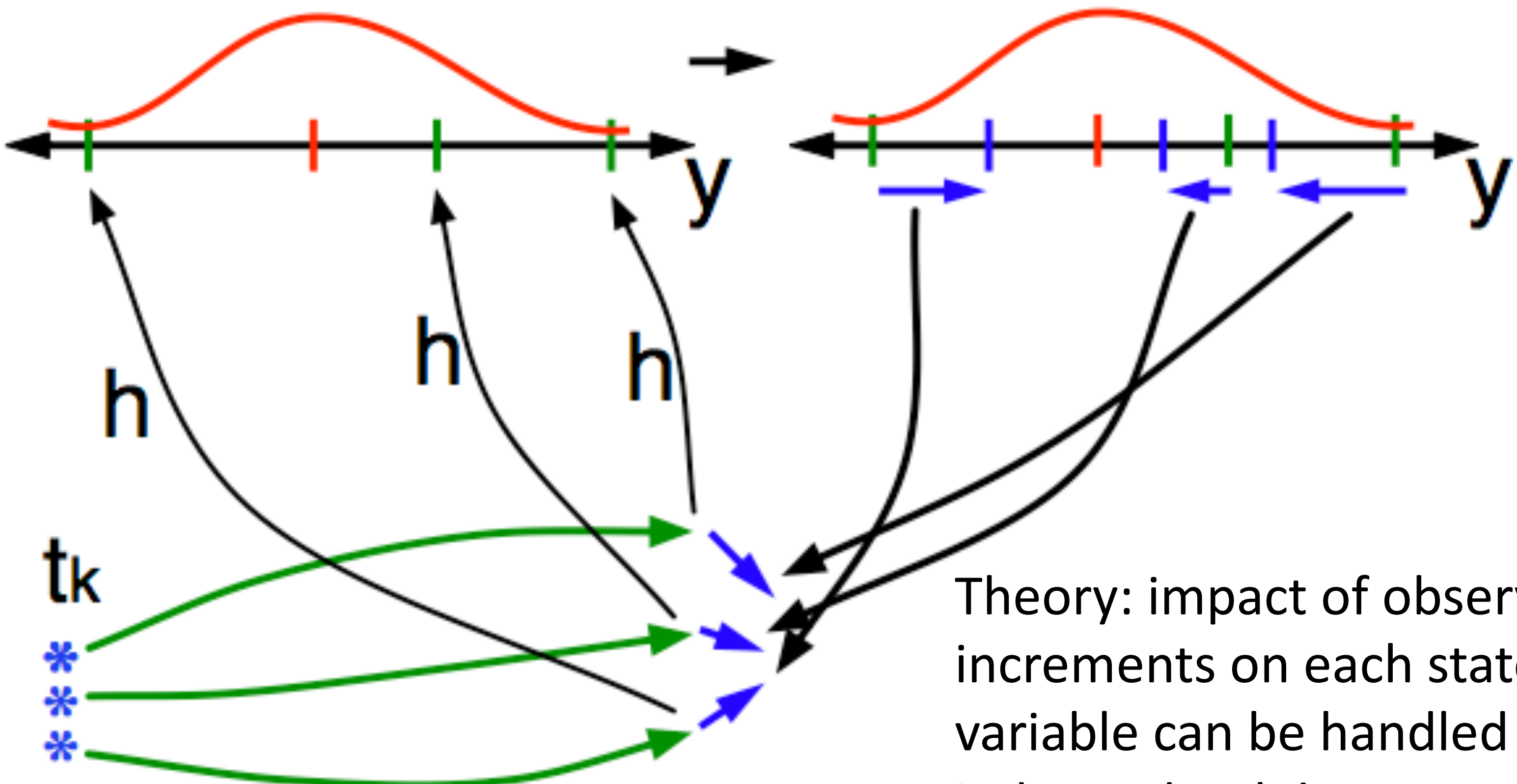


Ensemble Kalman filters assume Gaussianity for this problem.

Can compute increments without Gaussian assumptions. (RHF).

Schematic of a Sequential Ensemble Filter

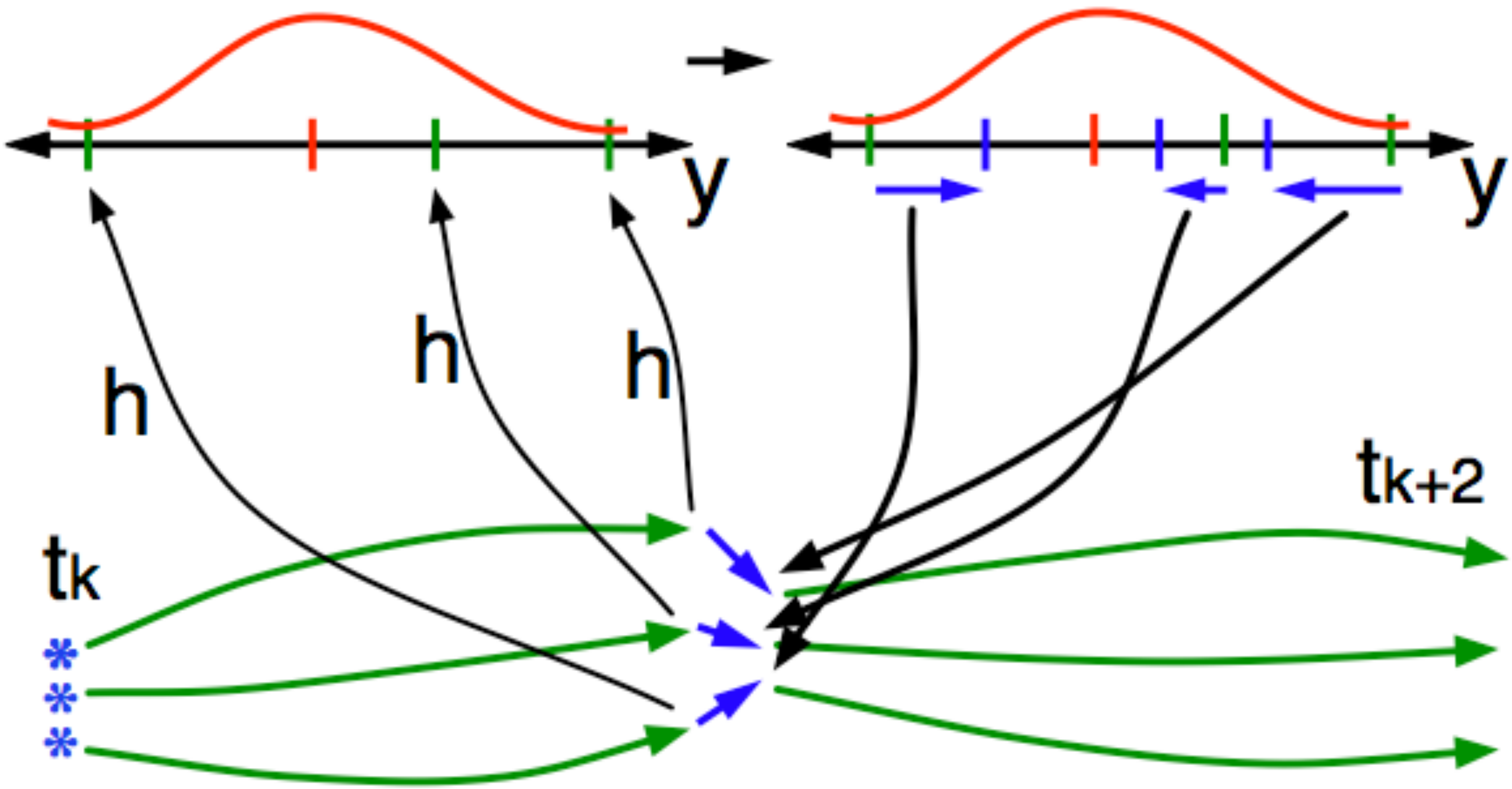
5. Use ensemble samples of y and each state variable to linearly regress observation increments onto state variable increments.



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

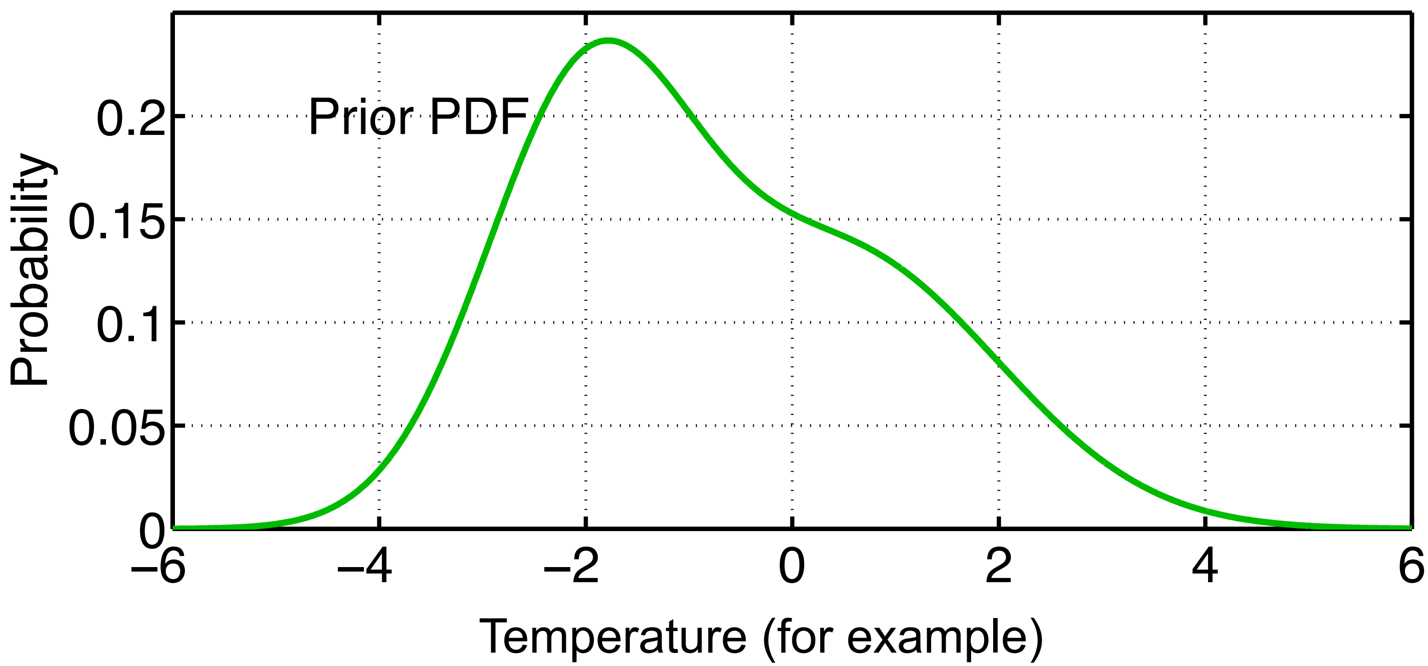
Schematic of a Sequential Ensemble Filter

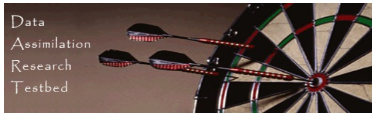
6. When all ensemble members for each state variable are updated, there is a new analysis. Integrate to time of next observation …



Bayes rule is the key to ensemble data assimilation.

$$P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_k}\right)=\frac{P(\mathbf{y}_k|\mathbf{x})P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_{k-1}}\right)}{Normalization}$$

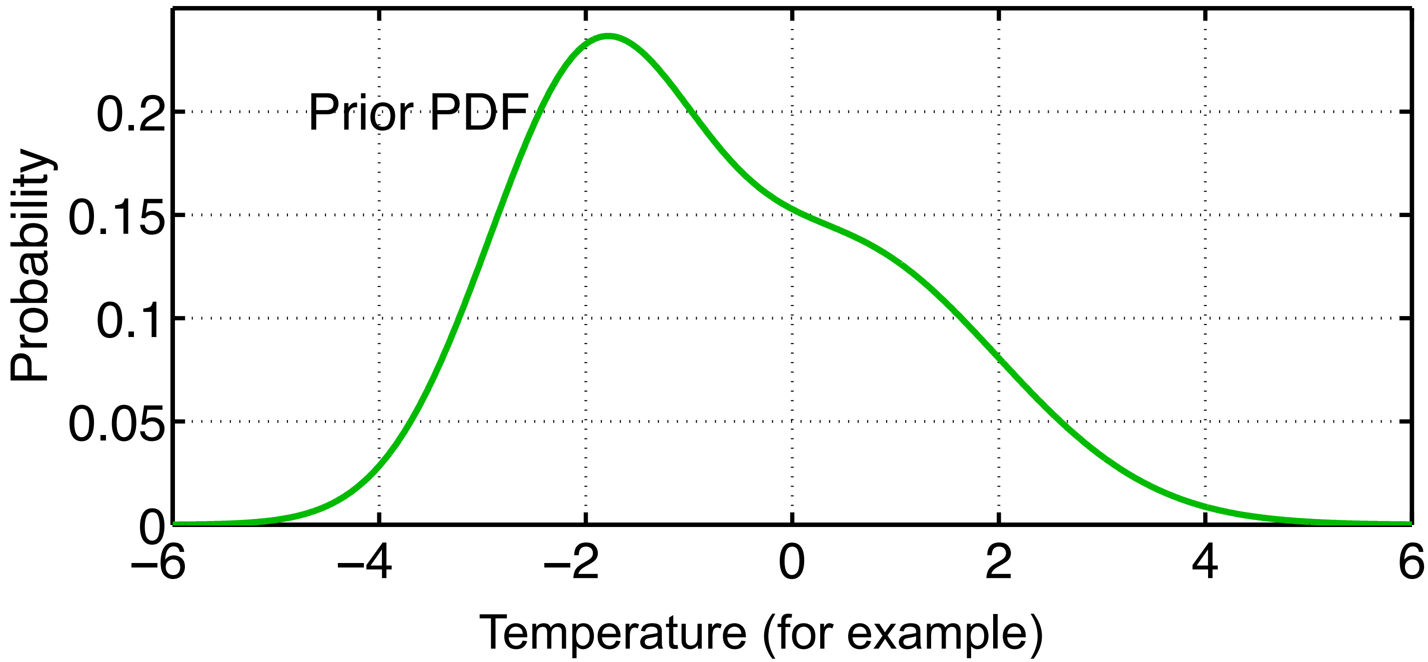




Bayes Rule (1D example)

Prior: from model forecast.

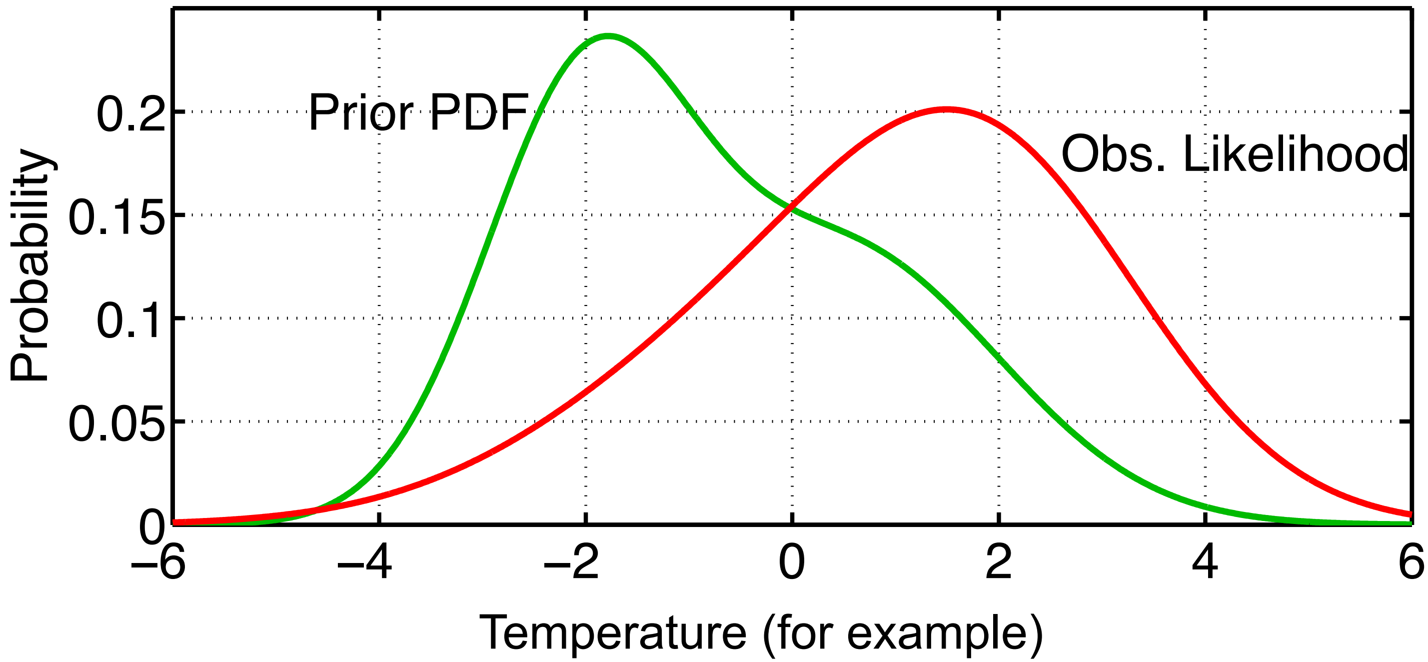
$$P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_k}\right) = \frac{P(\mathbf{y}_k|\mathbf{x})P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_{k-1}}\right)}{Normalization}$$



Bayes Rule (1D example)

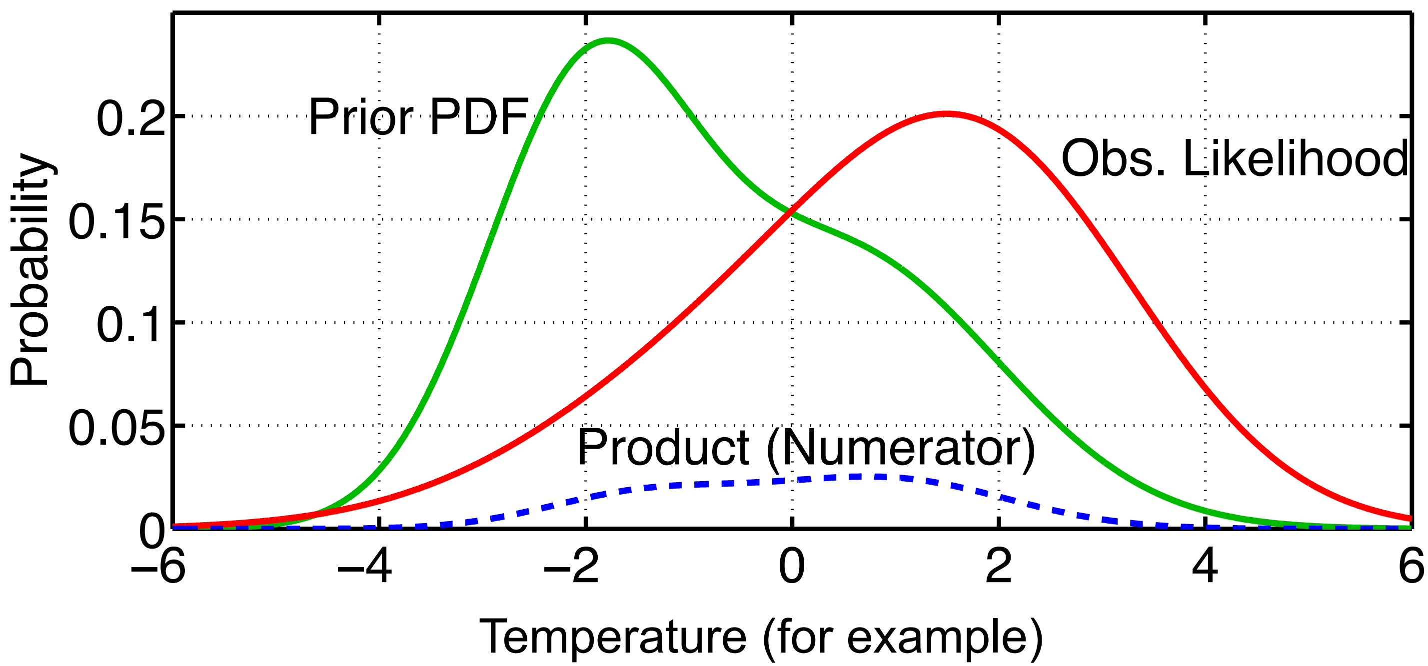
Likelihood:
from instrument.

$$P(\mathbf{x}_{t_k}|\mathbf{Y}_{t_k}) = \frac{P(\mathbf{y}_k|\mathbf{x})P(\mathbf{x}_{t_k}|\mathbf{Y}_{t_{k-1}})}{Normalization}$$



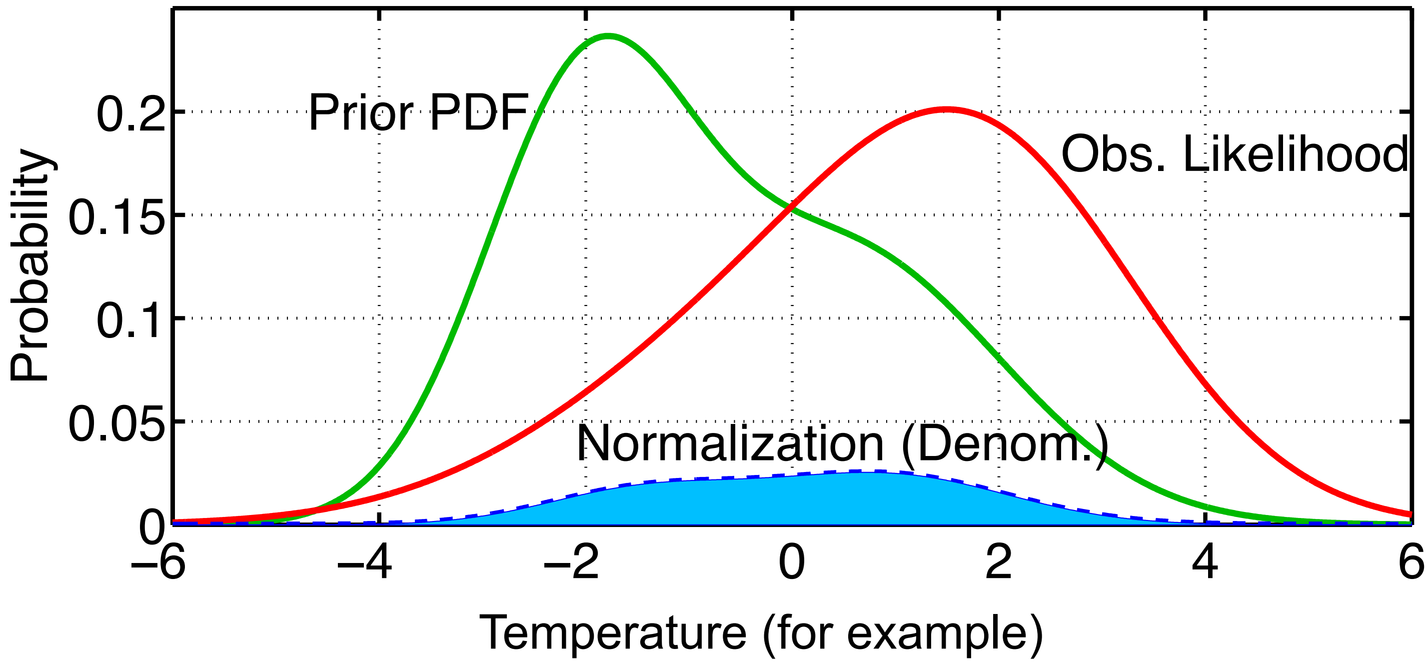
Bayes Rule (1D example)

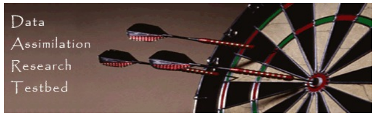
$$P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_k}\right) = \frac{P(\mathbf{y}_k|\mathbf{x})P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_{k-1}}\right)}{Normalization}$$



Bayes Rule (1D example)

$$P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_k}\right) = \frac{P(\mathbf{y}_k|\mathbf{x})P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_{k-1}}\right)}{Normalization}$$

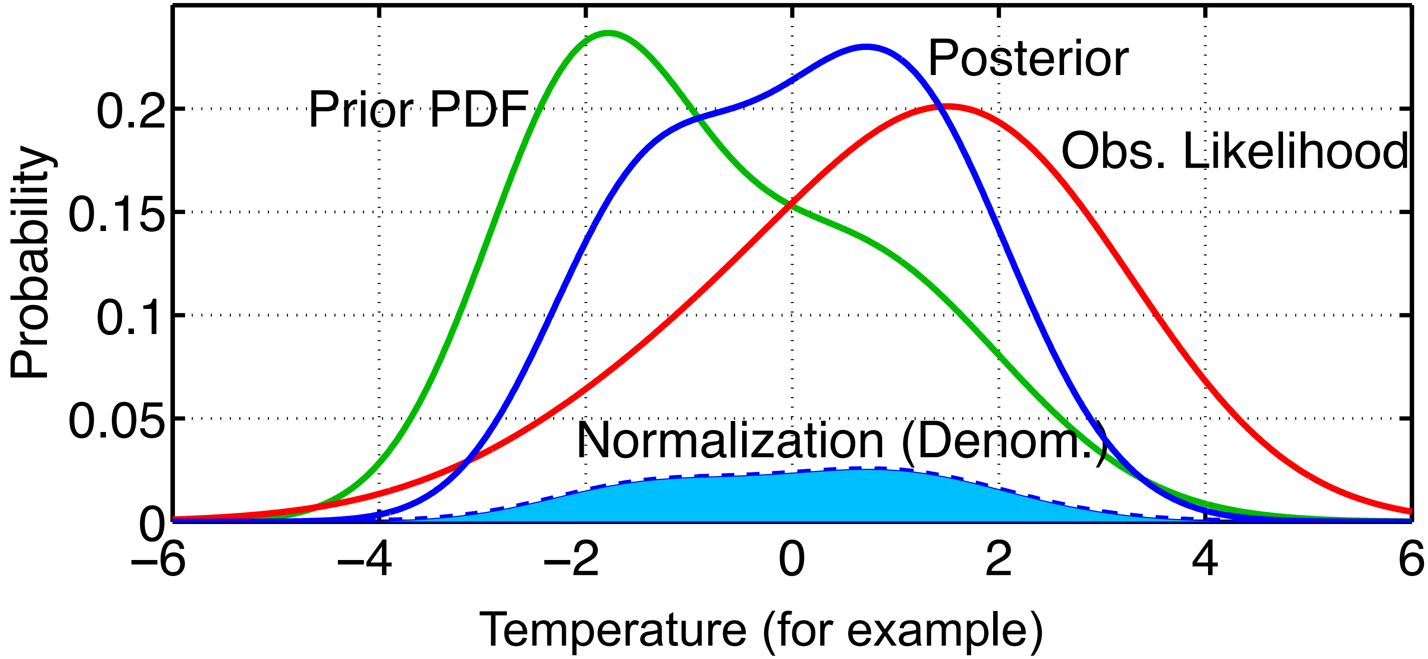




Bayes Rule (1D example)

Posterior:
(analysis).

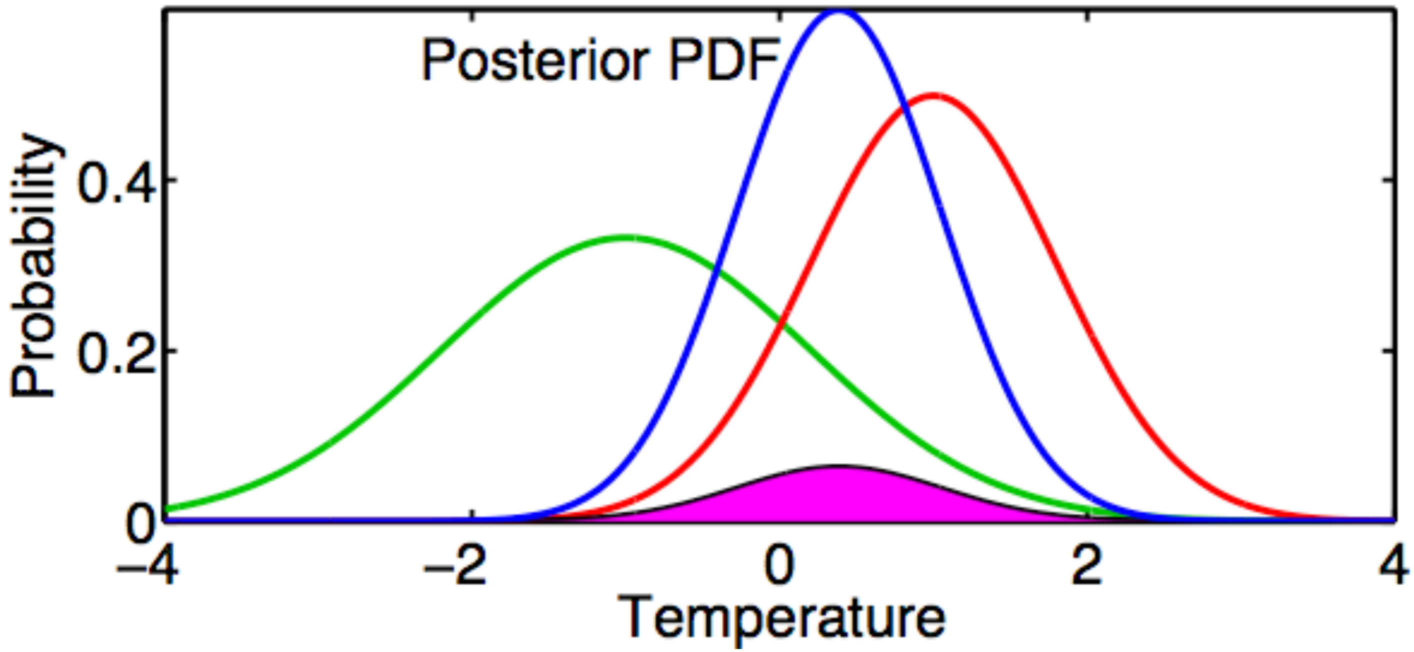
$$P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_k}\right) = \frac{P(\mathbf{y}_k|\mathbf{x})P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_{k-1}}\right)}{Normalization}$$



Bayes Rule (1D example)

Most ensemble assimilation algorithms assume Gaussians.
May be okay for quantity like temperature.

$$P(\mathbf{x}_{t_k}|\mathbf{Y}_{t_k}) = \frac{P(\mathbf{y}_k|\mathbf{x})P(\mathbf{x}_{t_k}|\mathbf{Y}_{t_{k-1}})}{Normalization}$$

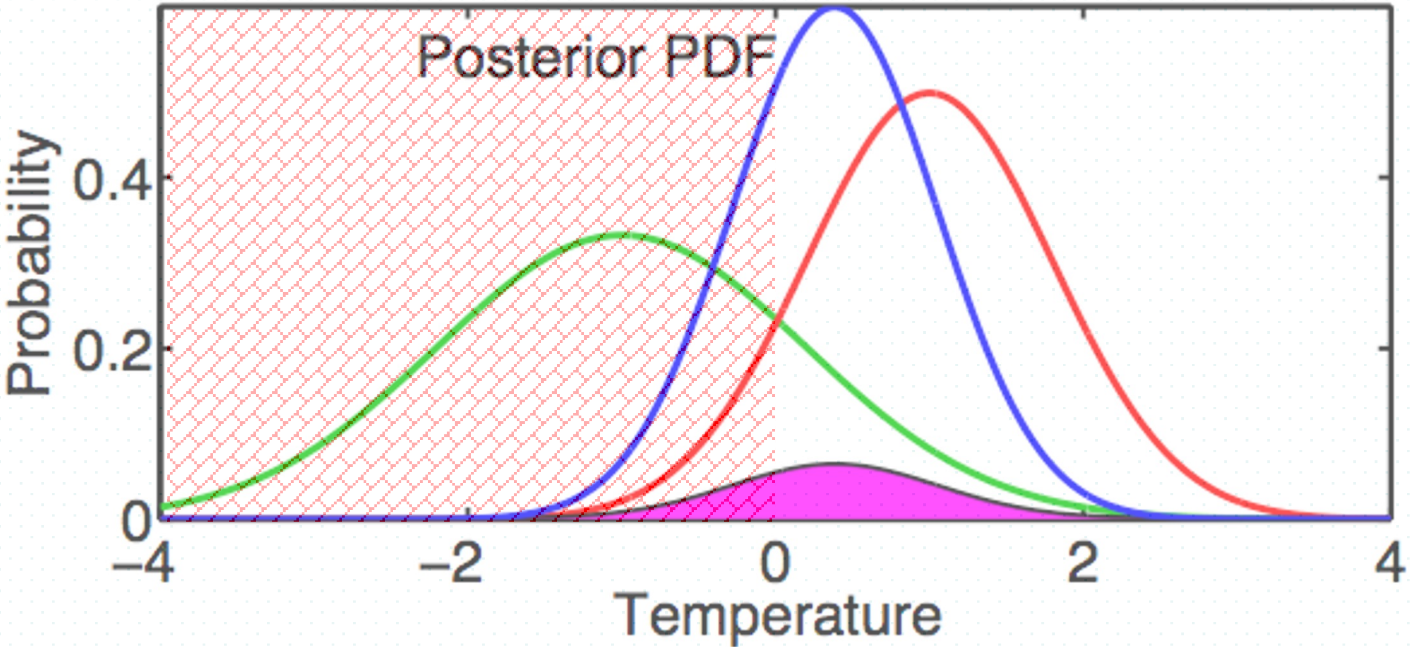


Bayes Rule (1D example)

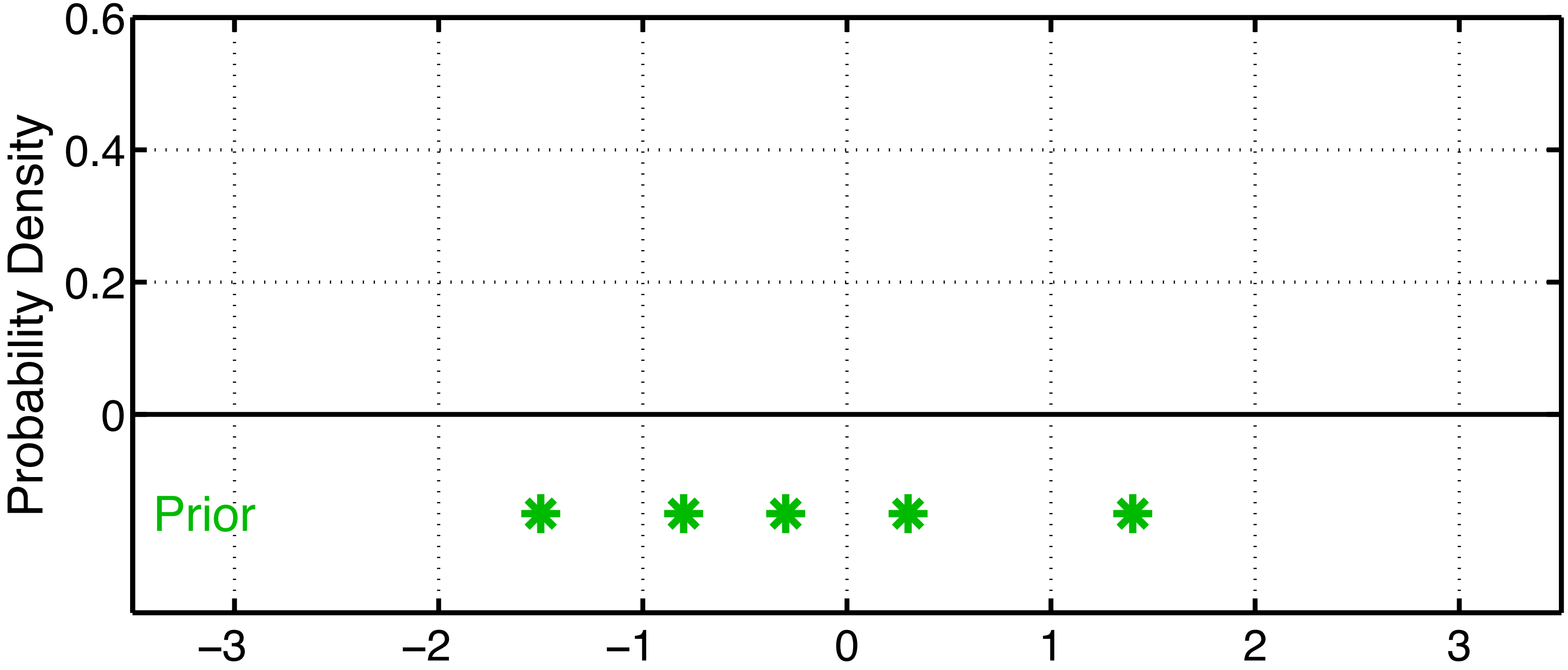
Most ensemble assimilation algorithms assume Gaussians.

Tracer concentration is bounded. Gaussian a poor choice.

$$P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_k}\right) = \frac{P(\mathbf{y}_k|\mathbf{x})P\left(\mathbf{x}_{t_k}|\mathbf{Y}_{t_{k-1}}\right)}{Normalization}$$



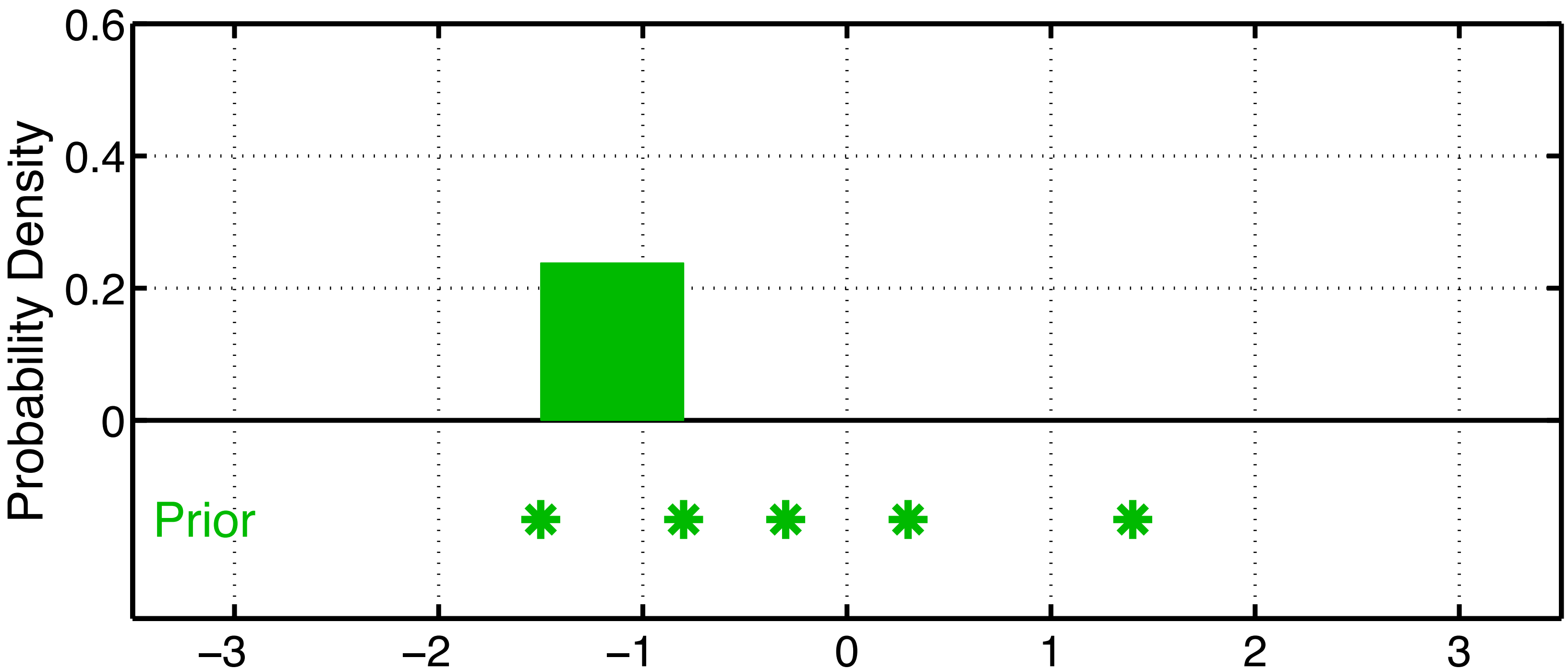
Observation-Space Rank Histogram Filter



Apply forward operator to each ensemble member.

Get prior ensemble in observation space.

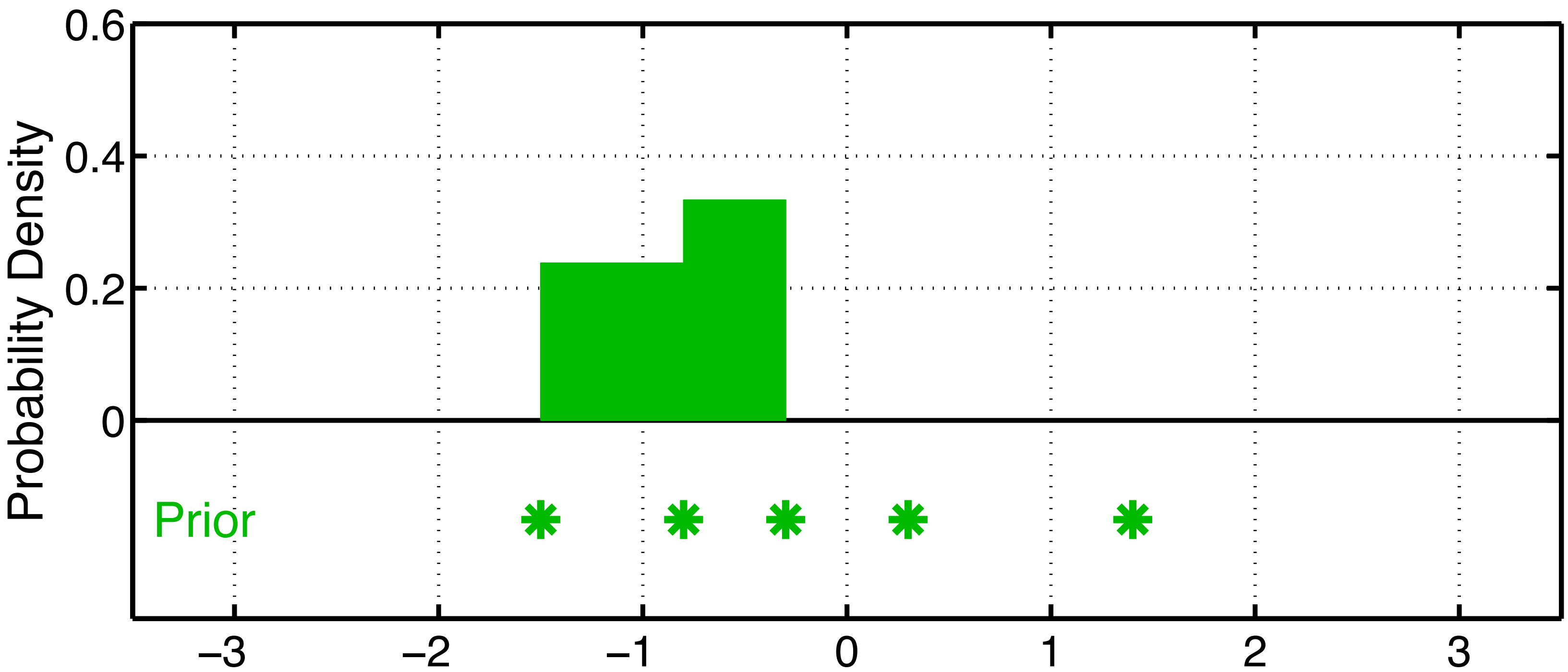
Observation-Space Rank Histogram Filter



Step 1: Get continuous prior distribution density.

- Place (ens_size + 1)$^{-1}$ mass between adjacent ensemble members.
- Reminiscent of rank histogram evaluation method.

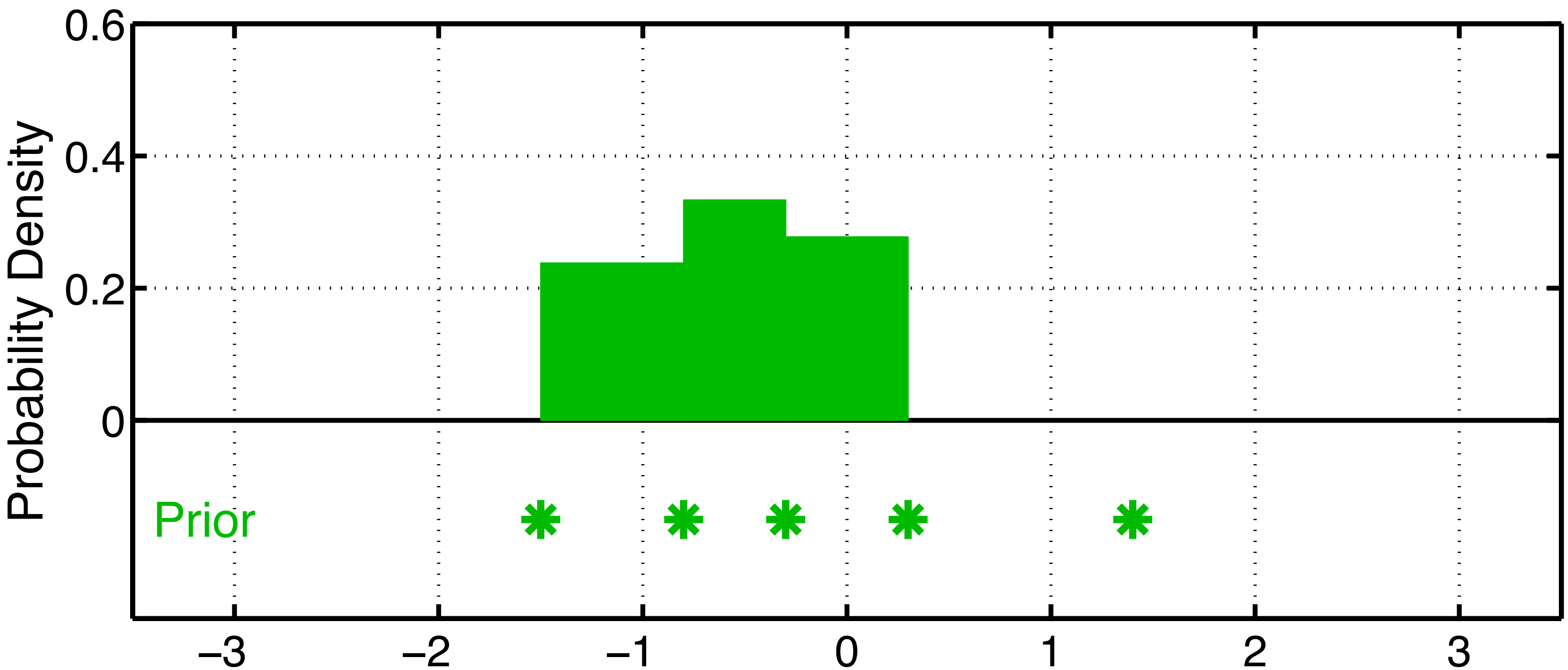
Observation-Space Rank Histogram Filter



Step 1: Get continuous prior distribution density.

- Place (ens_size + 1)$^{-1}$ mass between adjacent ensemble members.
- Reminiscent of rank histogram evaluation method.

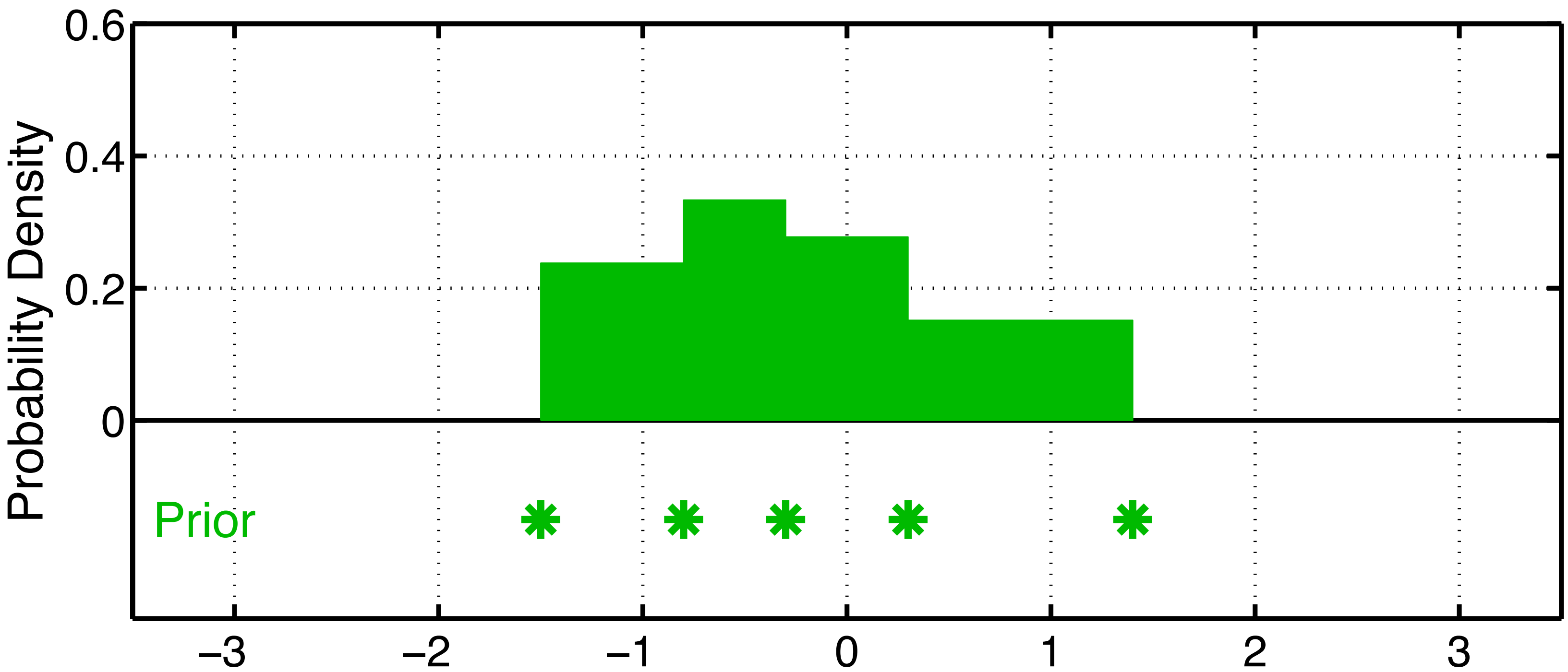
Observation-Space Rank Histogram Filter



Step 1: Get continuous prior distribution density.

- Place (ens_size + 1)$^{-1}$ mass between adjacent ensemble members.
- Reminiscent of rank histogram evaluation method.

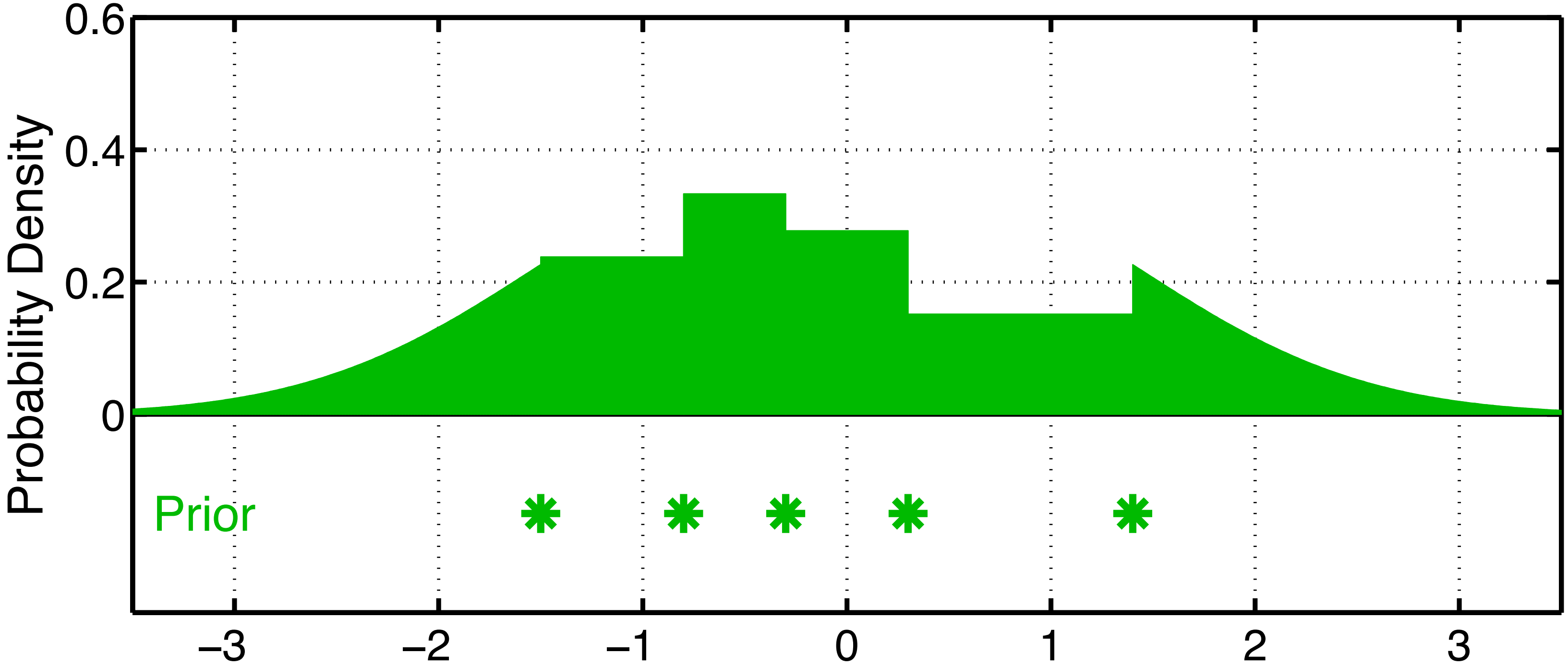
Observation-Space Rank Histogram Filter



Step 1: Get continuous prior distribution density.

- Place (ens_size + 1)$^{-1}$ mass between adjacent ensemble members.
- Reminiscent of rank histogram evaluation method.

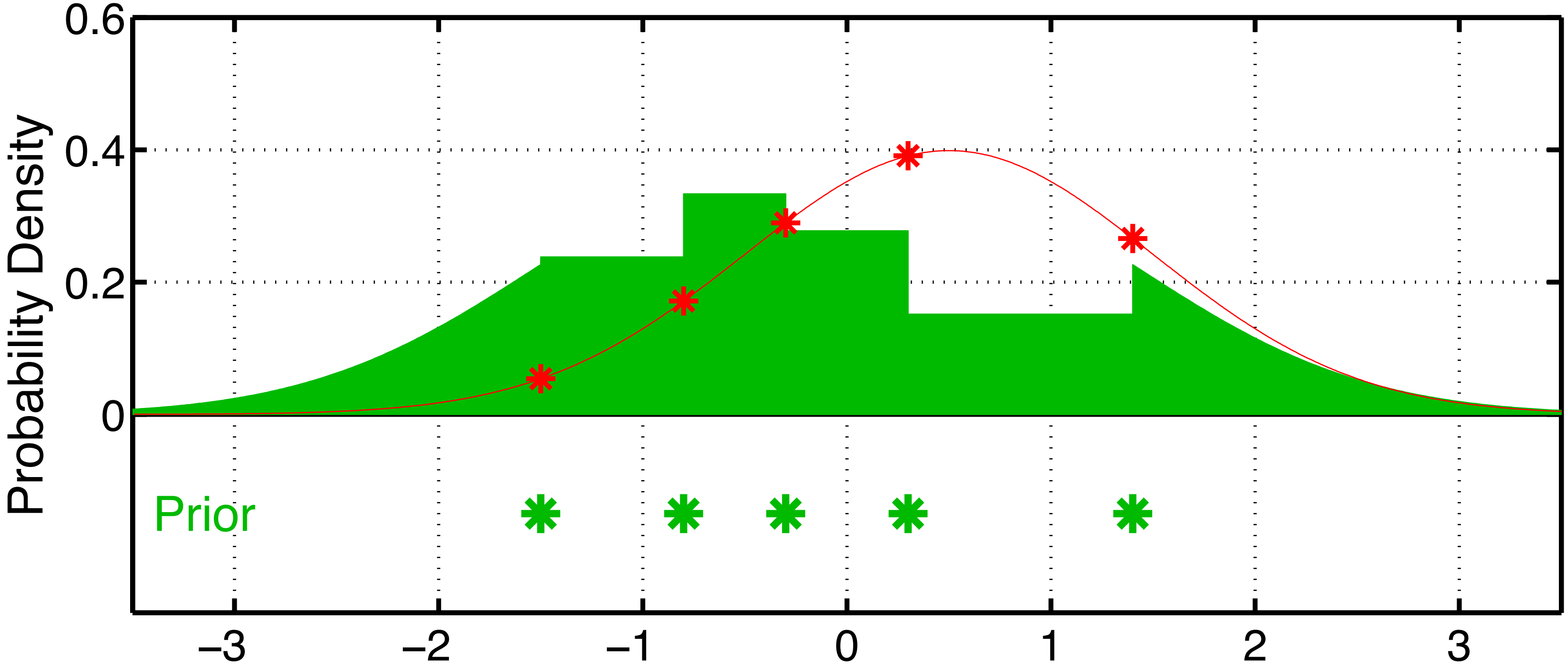
Observation-Space Rank Histogram Filter



Step 1: Get continuous prior distribution density.

- Partial gaussian kernels on tails, N(*tail_mean*, *ens_sd*).
- *tail_mean selected so that (ens_size + 1)$^{-1}$ mass is in tail.*

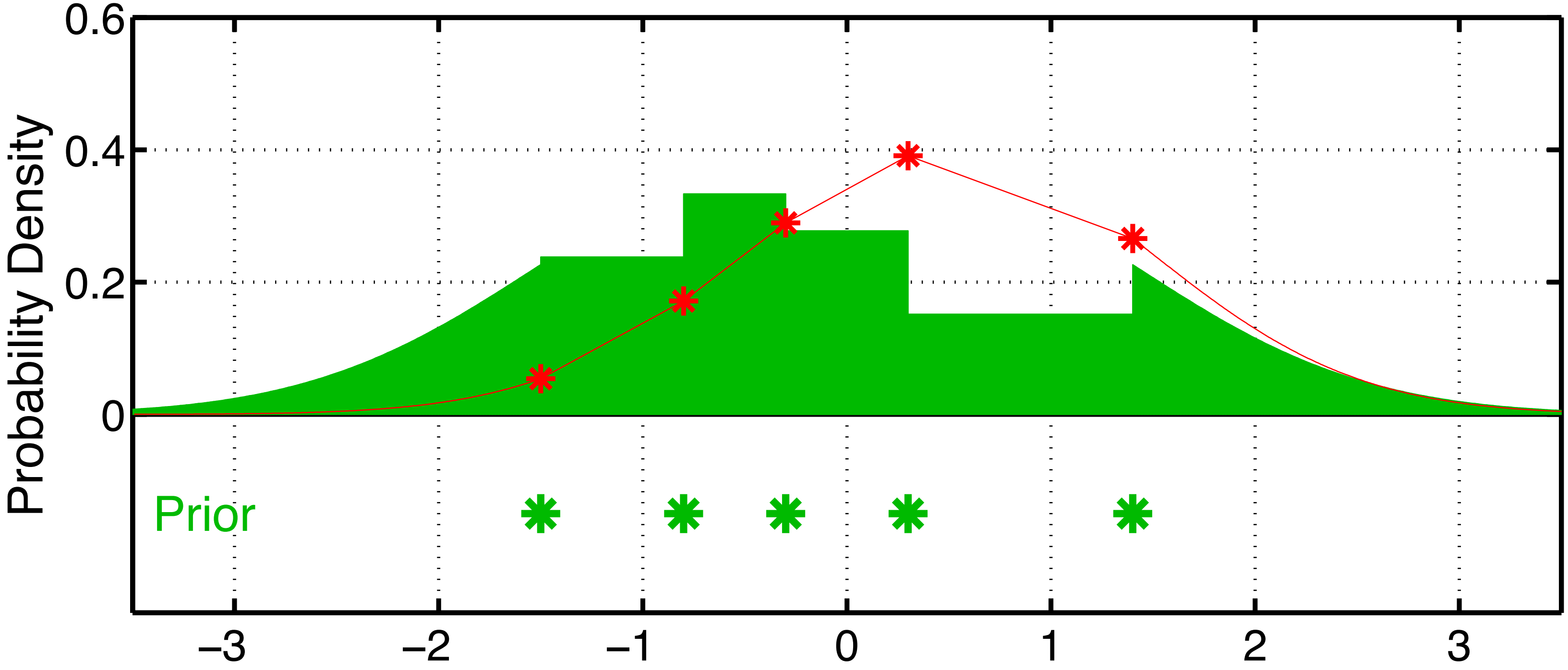
Observation-Space Rank Histogram Filter



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

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

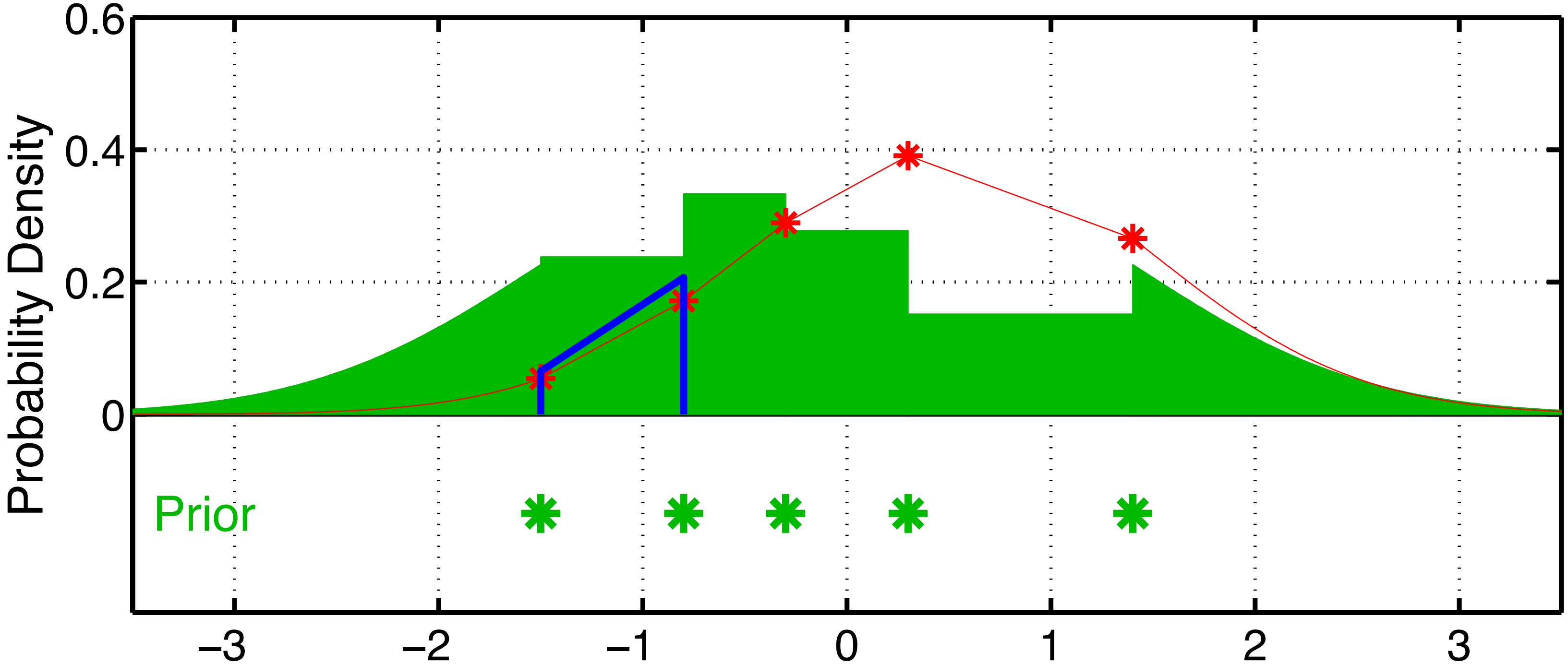
Observation-Space Rank Histogram Filter



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

- Can approximate interior likelihood with linear fit; for efficiency.

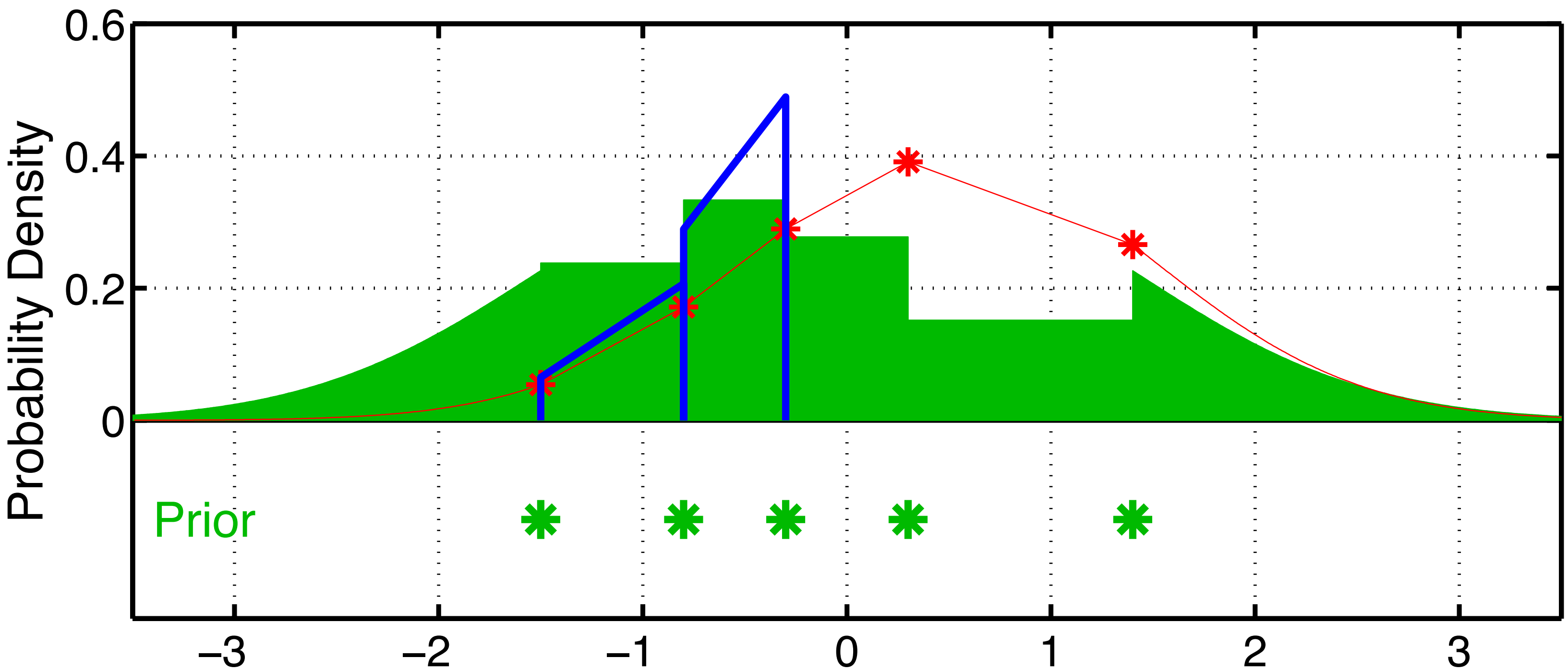
Observation-Space Rank Histogram Filter



Step 3: Compute continuous posterior distribution.

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

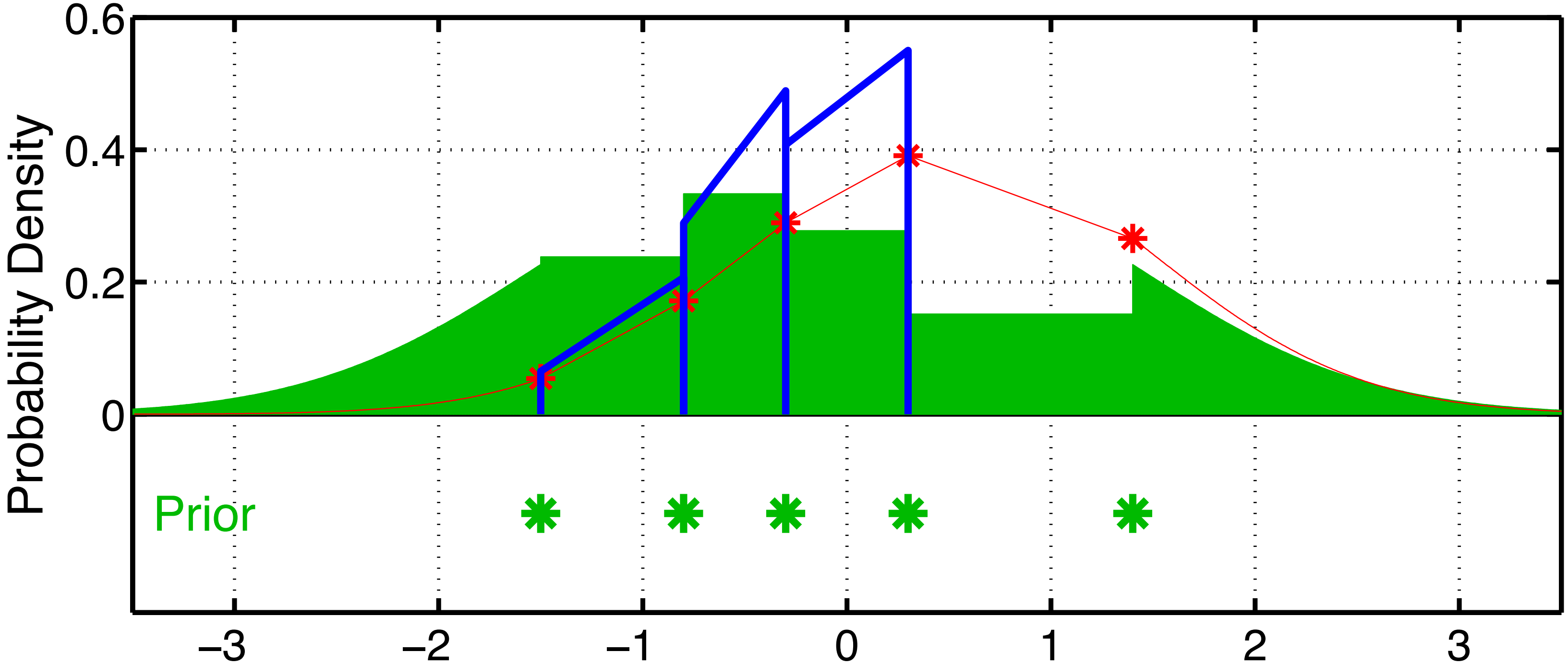
Observation-Space Rank Histogram Filter



Step 3: Compute continuous posterior distribution.

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

Observation-Space Rank Histogram Filter

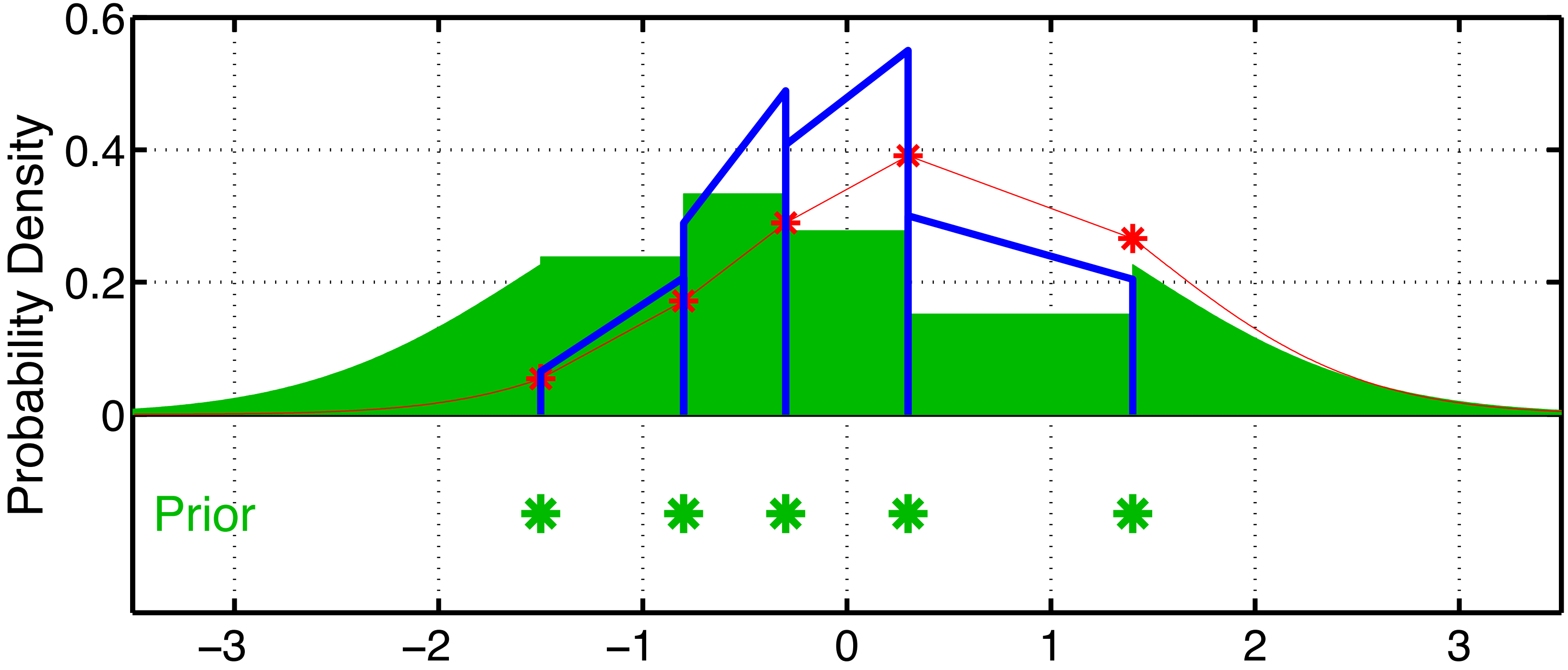


Step 3: Compute continuous posterior distribution.

- Approximate likelihood with trapezoidal quadrature, take product.

  (Displayed product normalized to make posterior a PDF).

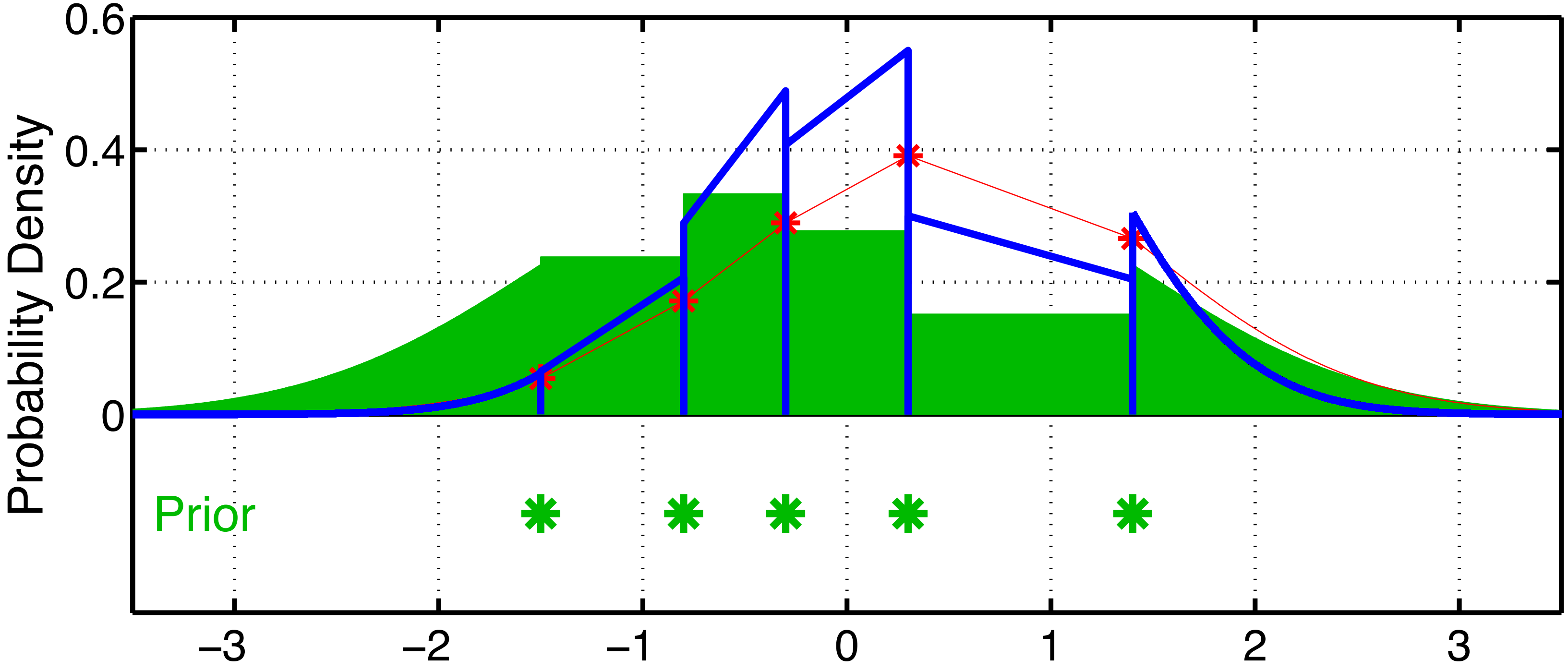
Observation-Space Rank Histogram Filter



Step 3: Compute continuous posterior distribution.

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

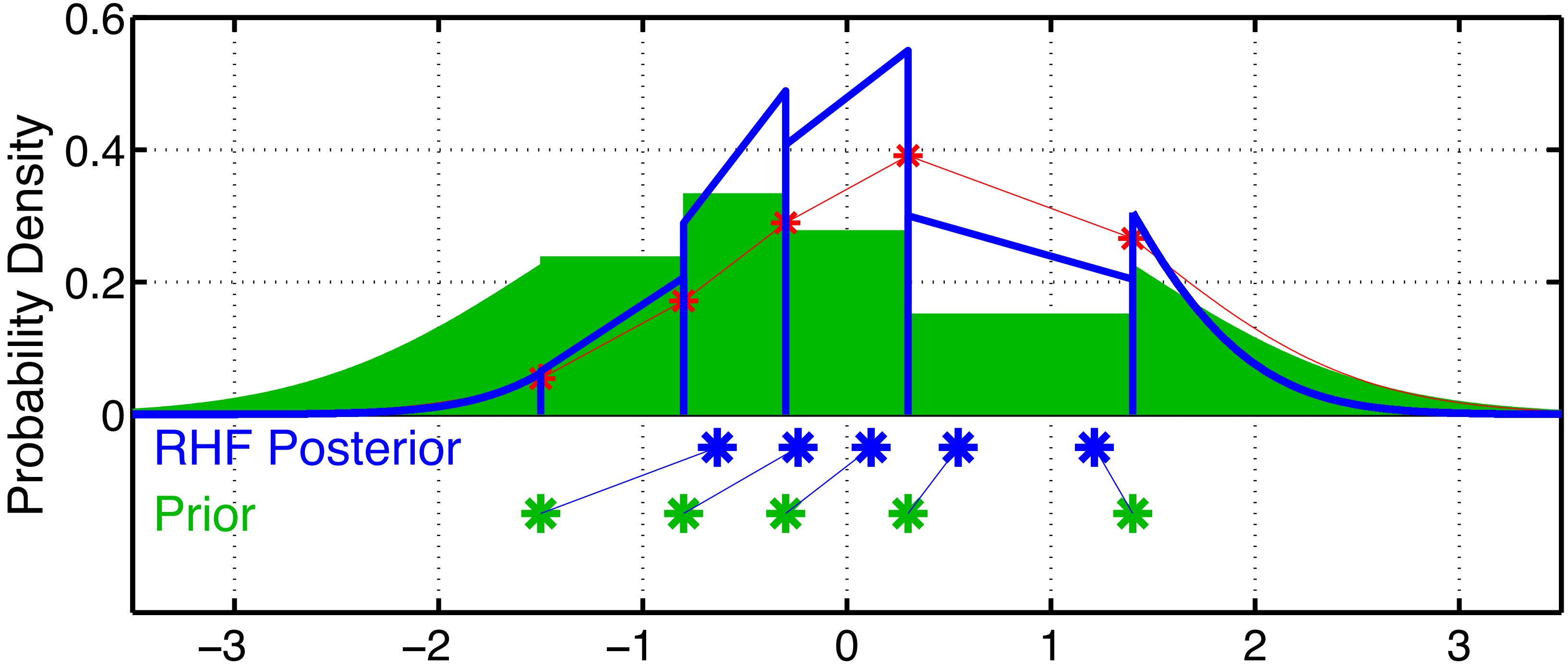
Observation-Space Rank Histogram Filter



Step 3: Compute continuous posterior distribution.

- Product of prior gaussian kernel with likelihood for tails.
- Easy for gaussian likelihood.

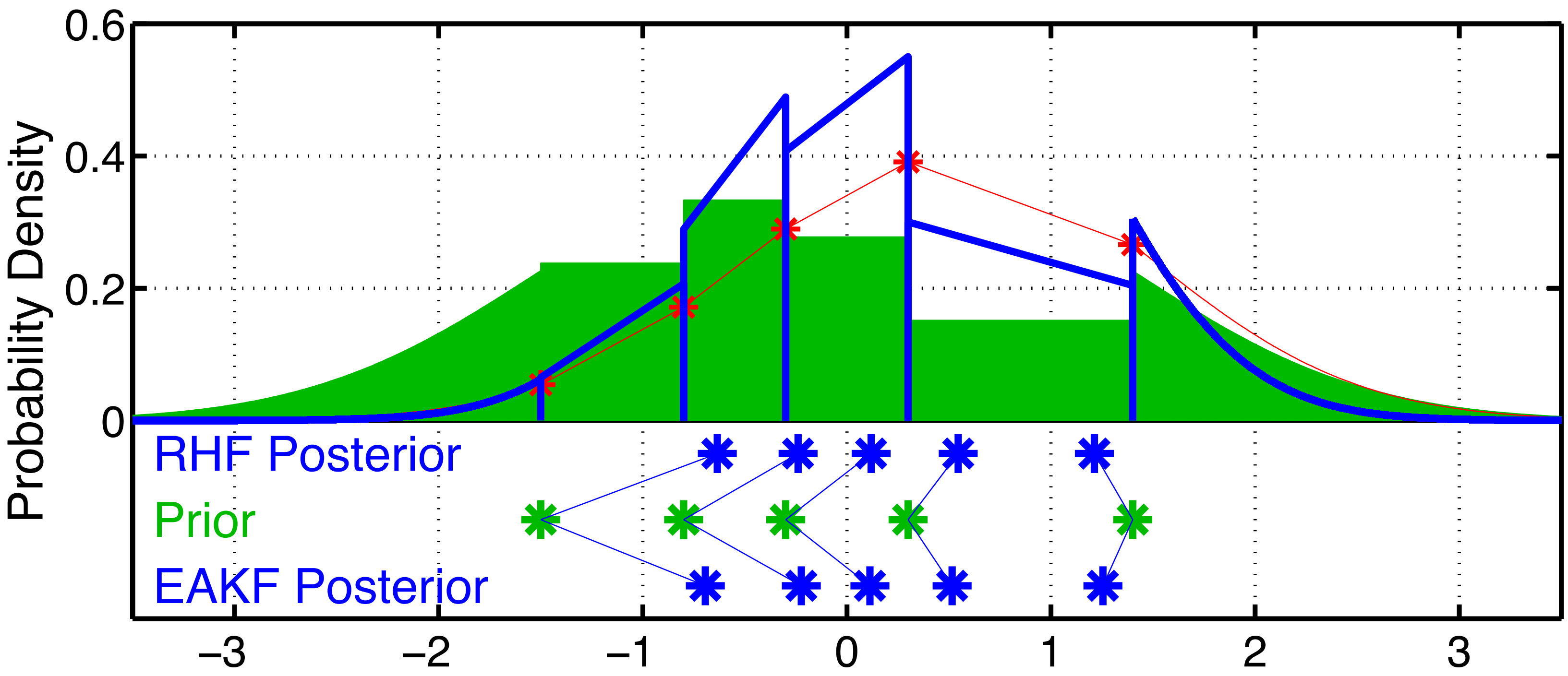
Observation-Space Rank Histogram Filter



Step 4: Compute posterior ensemble members:

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

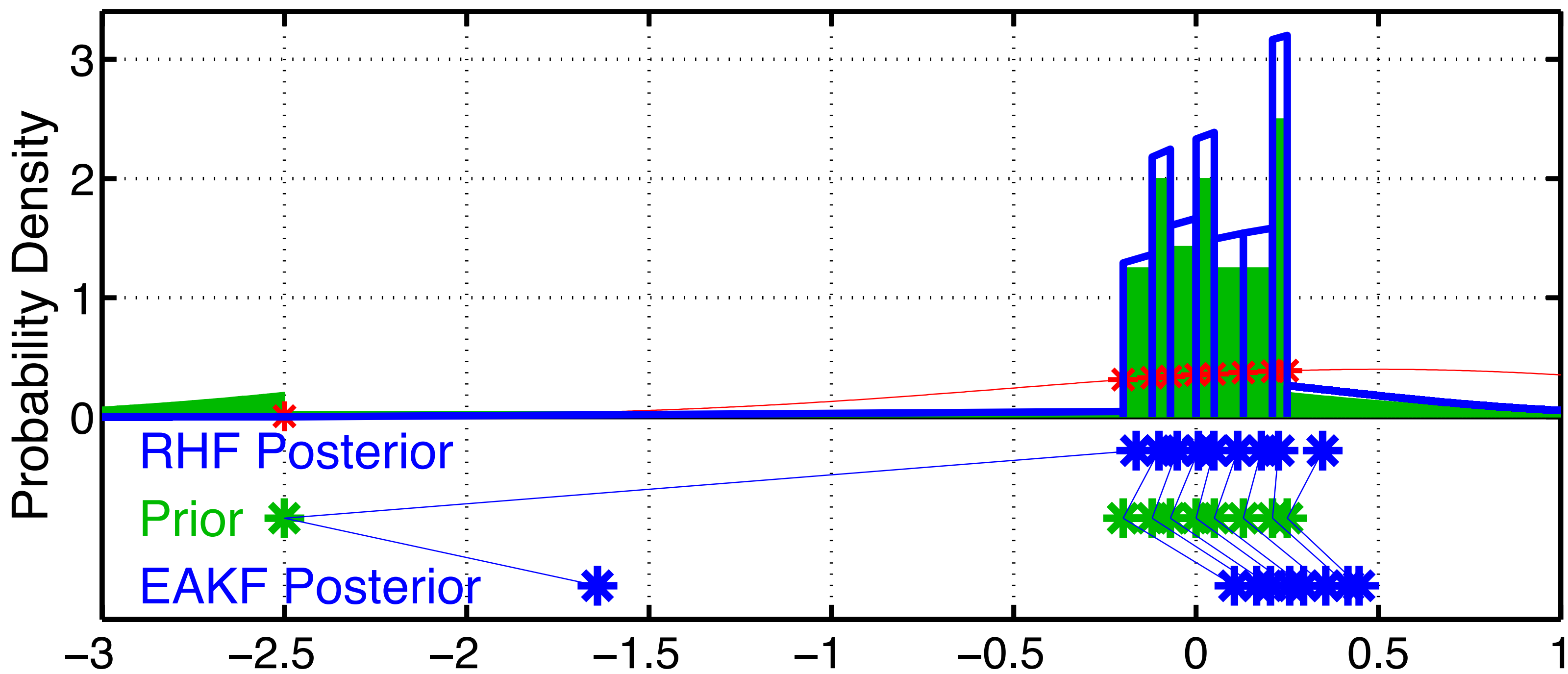
Observation-Space Rank Histogram Filter



Compare to standard Ensemble Adjustment Filter (EAKF).

Nearly gaussian case, differences are small.

Observation-Space Rank Histogram Filter

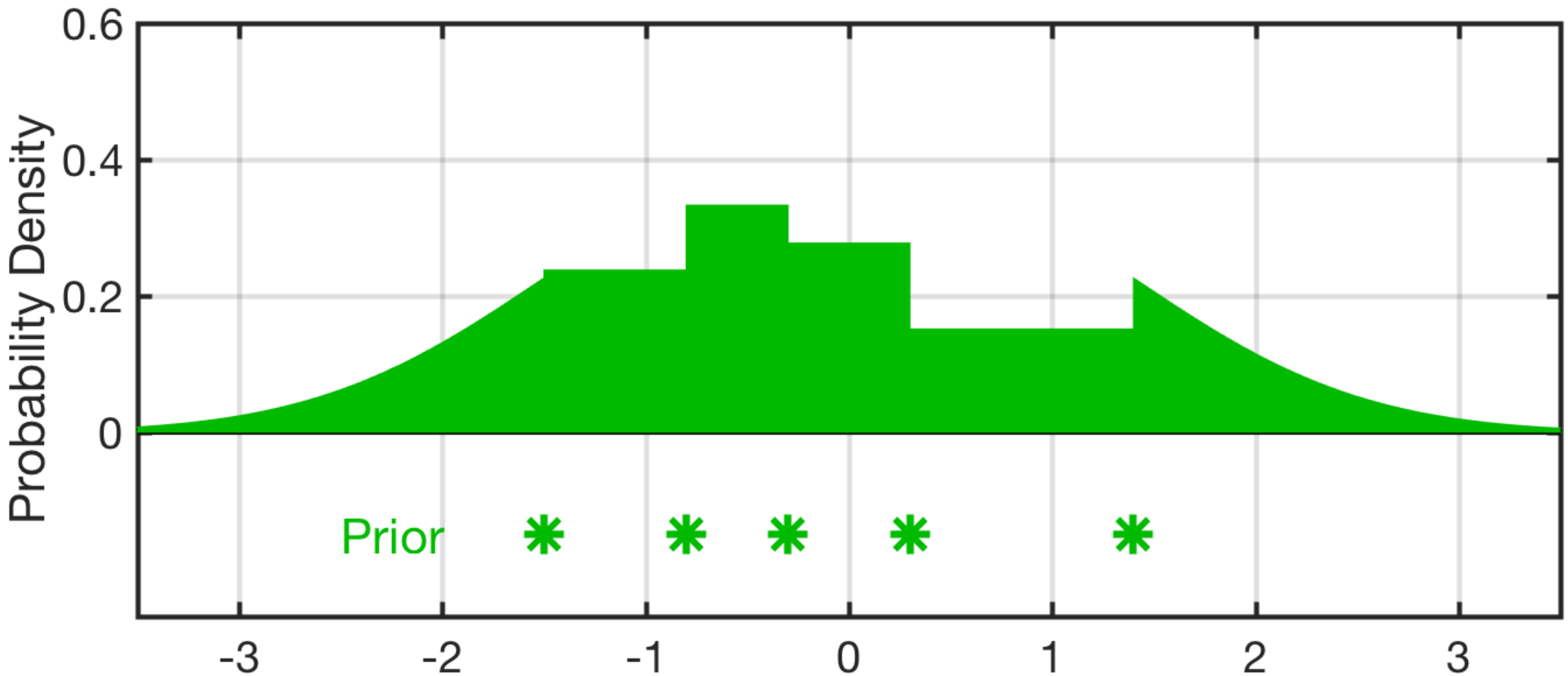


Rank Histogram gets rid of outlier that is clearly inconsistent with obs.

EAKF can't get rid of outlier.

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

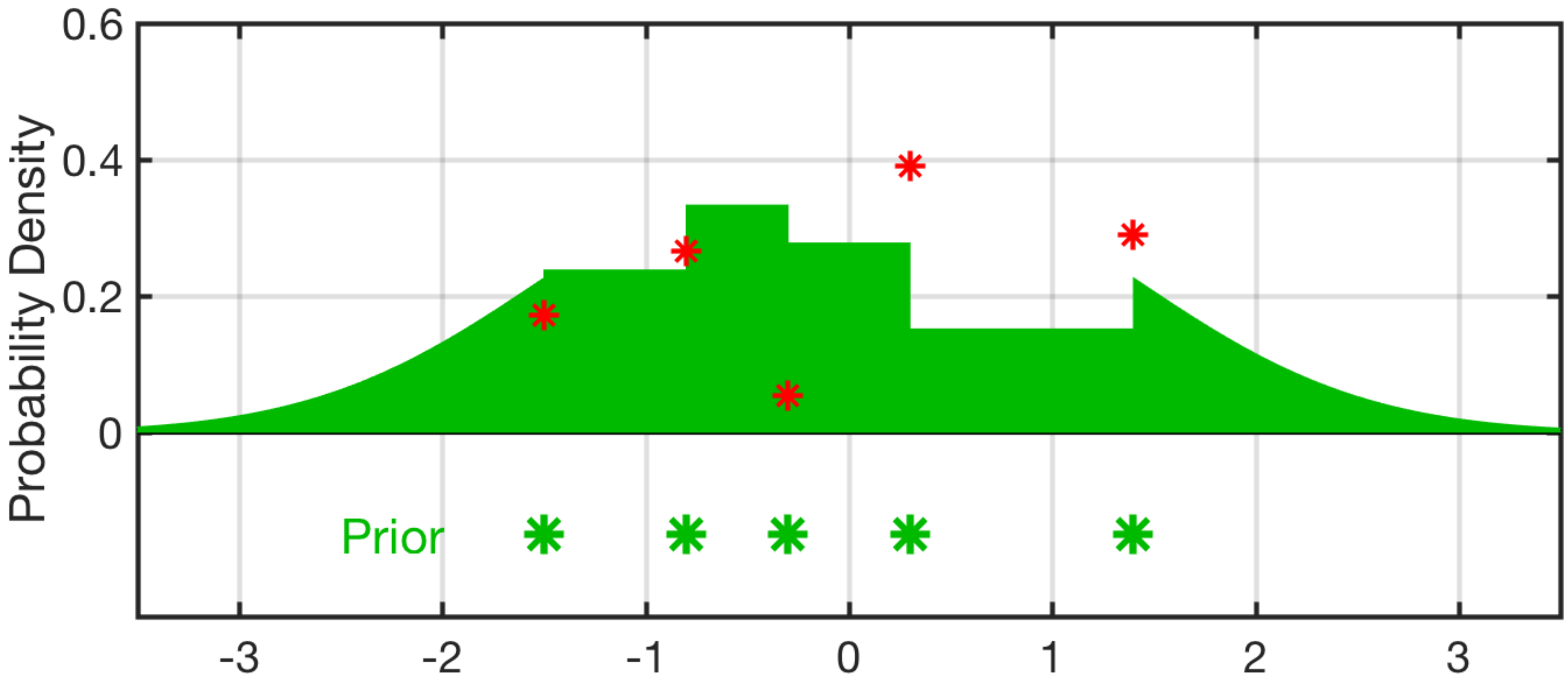
Rank Histogram Filter for State Marginals



Step 1: Get continuous prior distribution density (same).

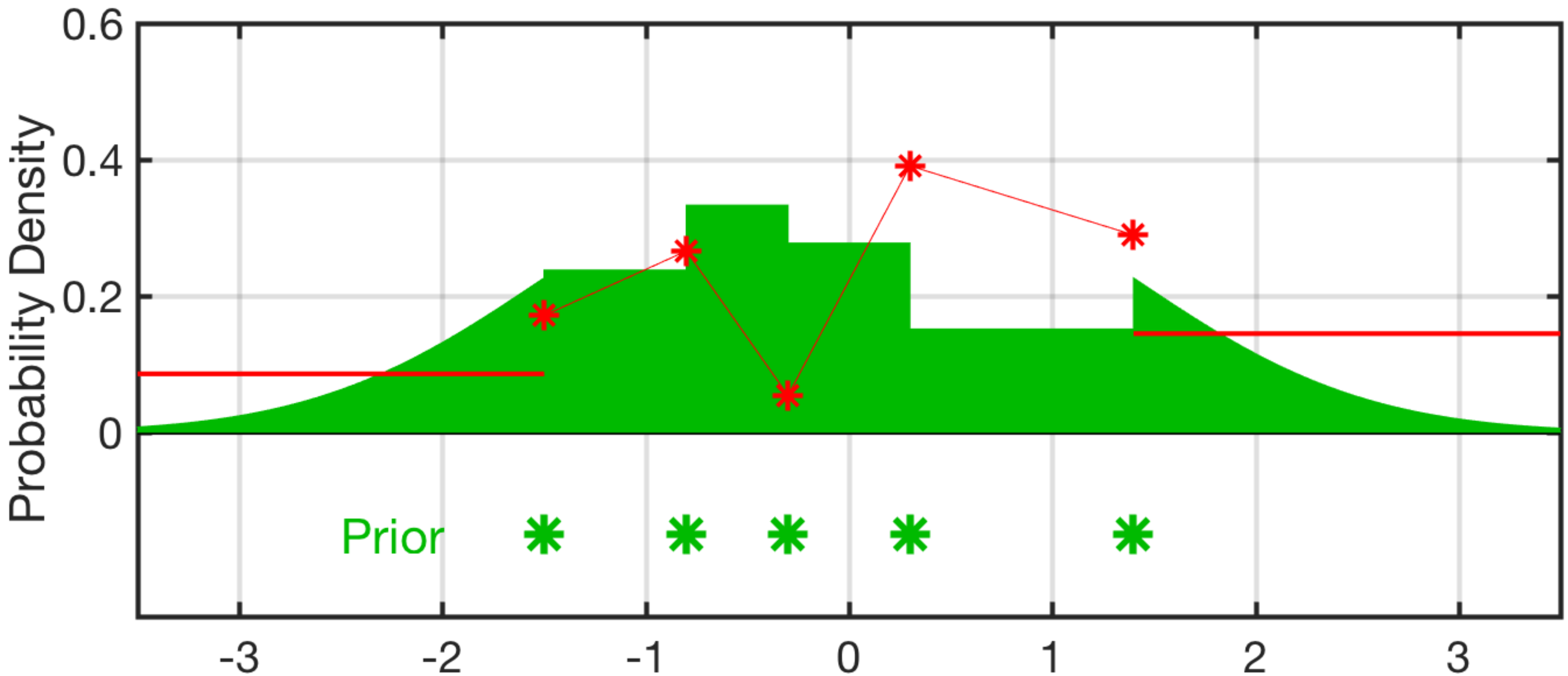
- Partial gaussian kernels on tails, N(*tail_mean*, *ens_sd*).
- *tail_mean selected so that (ens_size + 1)$^{-1}$ mass is in tail.*

Rank Histogram Filter for State Marginals



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

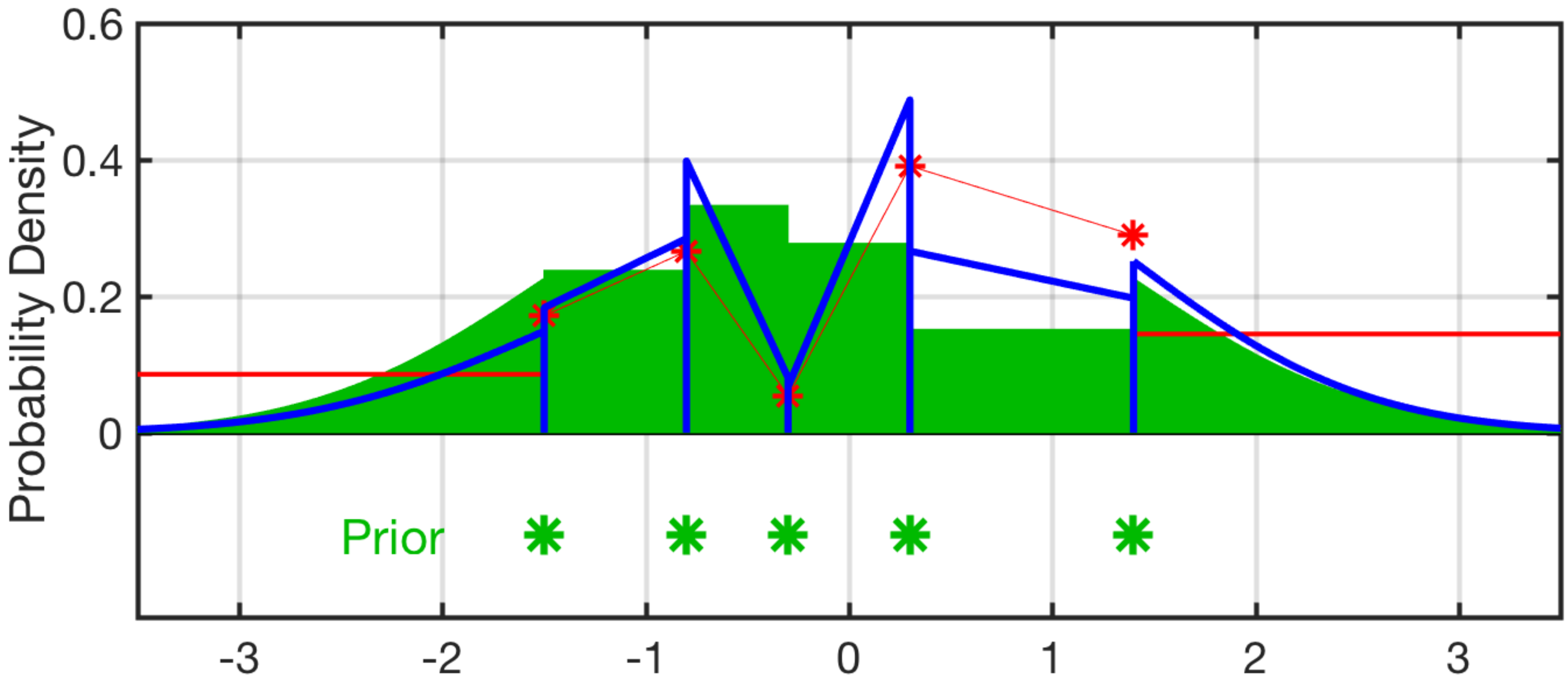
Rank Histogram Filter for State Marginals



Step 3: Compute continuous posterior distribution.

- Approximate likelihood with trapezoidal quadrature.
- Uniform tails! (Different).

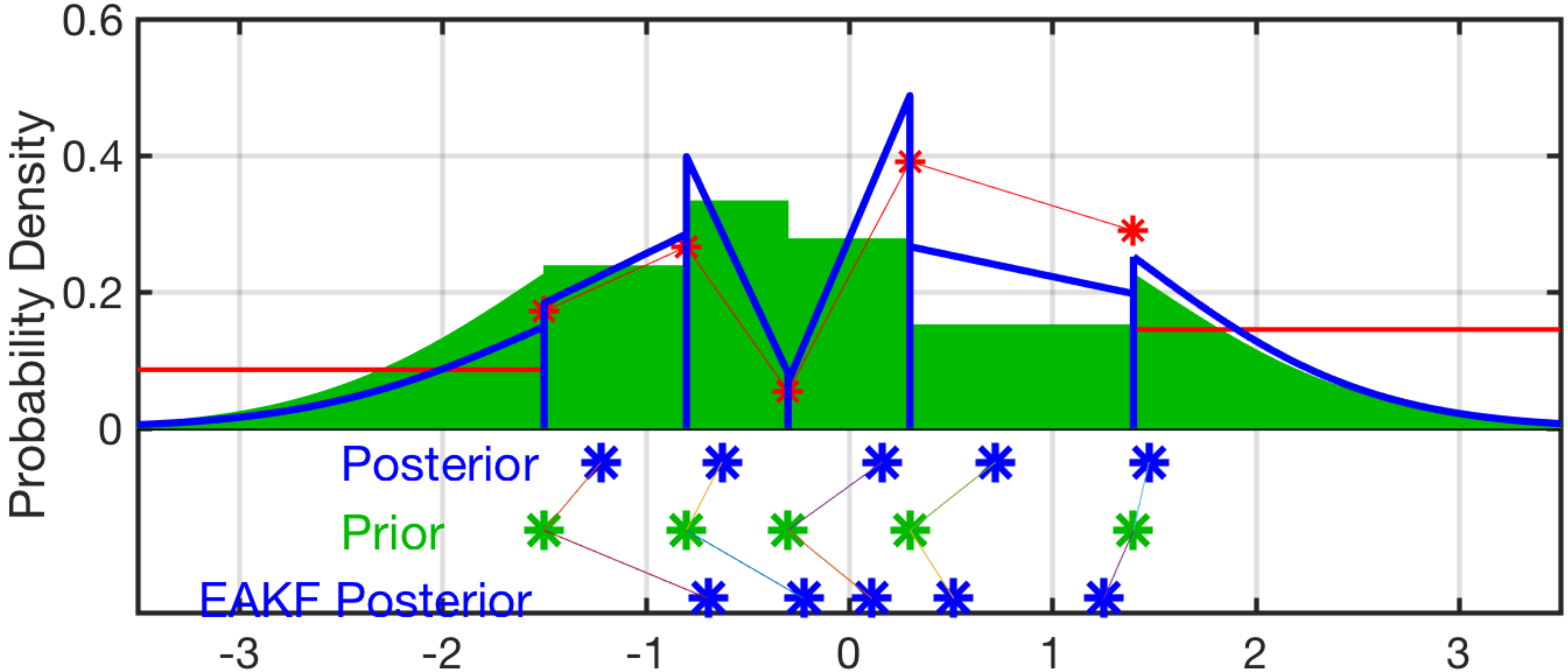
Rank Histogram Filter for State Marginals



Step 3: Compute continuous posterior distribution (same).

- Really simple with uniform likelihood tails.

Rank Histogram Filter for State Marginals



Step 4: Compute updated ensemble members (same):

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

Schematic of a Sequential Rank Histogram Filter for State Marginals

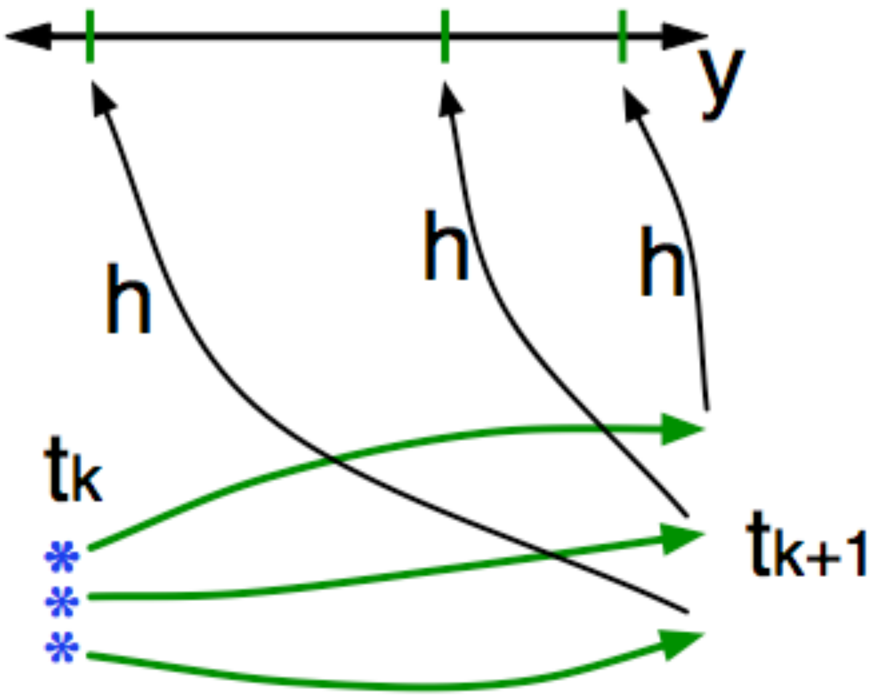
1. Use model to advance ensemble (3 members here) to time at which next observation becomes available.

Ensemble state estimate after using previous observation (analysis)

Ensemble state at time of next observation (prior)

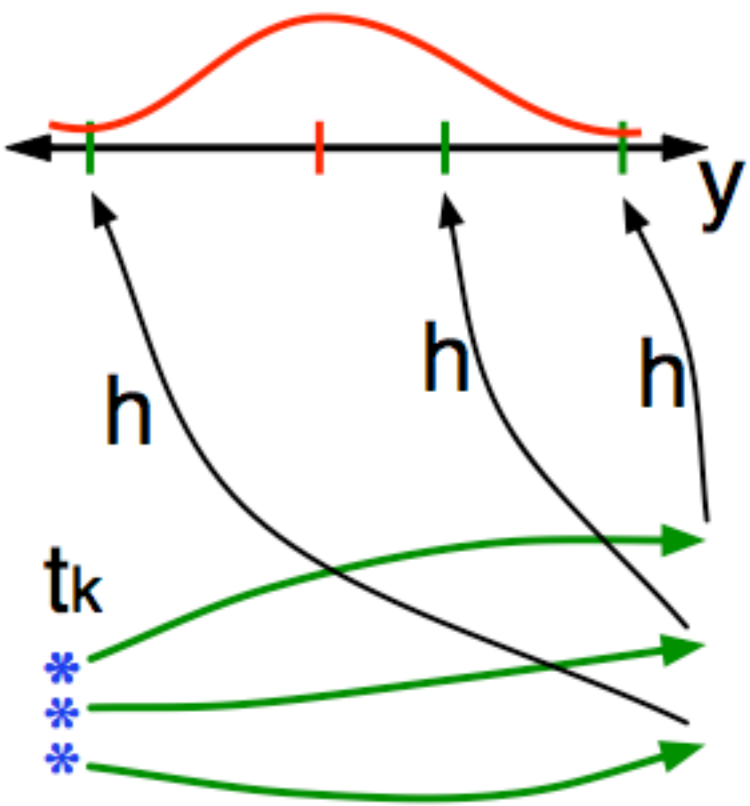
Schematic of a Sequential Rank Histogram Filter for State Marginals

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



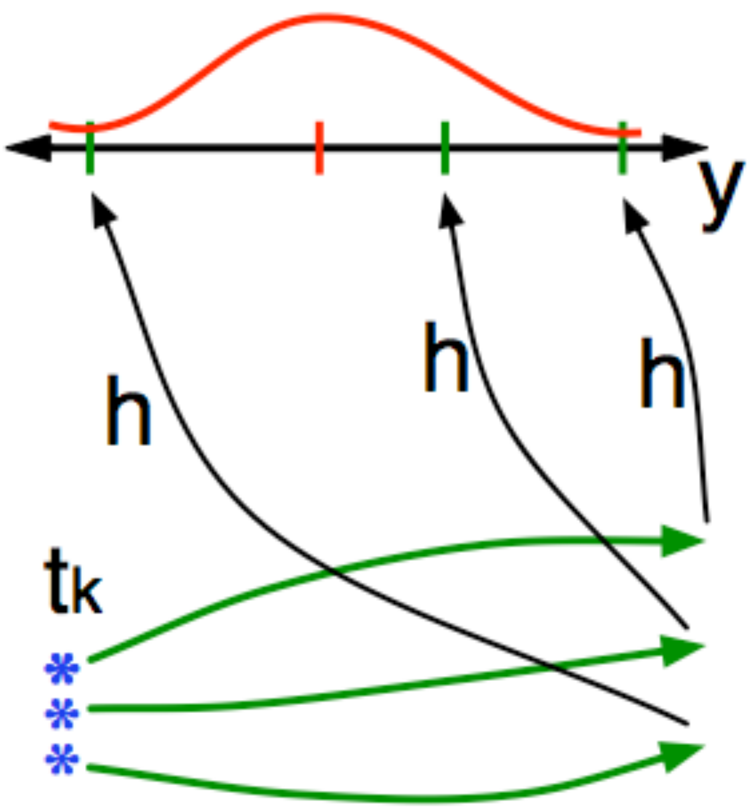
Schematic of a Sequential Rank Histogram Filter for State Marginals

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



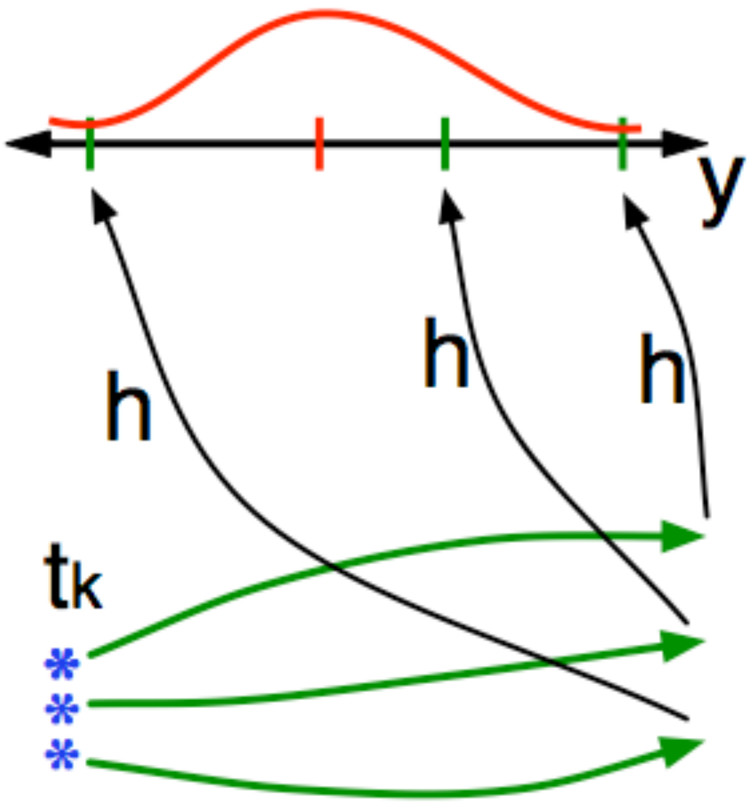
Schematic of a Sequential Rank Histogram Filter for State Marginals

4. Compute likelihood for each ensemble member. No need for gaussian error distribution or observation increments.



Schematic of a Sequential Rank Histogram Filter for State Marginals

5. Use RHF to update each state variable. Can be done in parallel.



But this just gives marginal for states.
Can 'pair' values to ensembles in many ways.

Pairing RHF Marginals Values to Ensemble Members

Naïve method:

Rank statistics of posterior ensemble same as prior.
Ensemble member with smallest prior value gets smallest posterior value.

This works fairly well for some applications.

Pairing RHF Marginals Values to Ensemble Members

Marginal Adjustment RHF method (MARHF):

Do standard RHF with regression, get preliminary posterior.

Get RHF State Marginal.

Rank statistics of posterior same as preliminary posterior.
Ensemble member with smallest preliminary posterior value gets smallest posterior value from RHF State Marginal.

Works well for many applications (but more expensive).

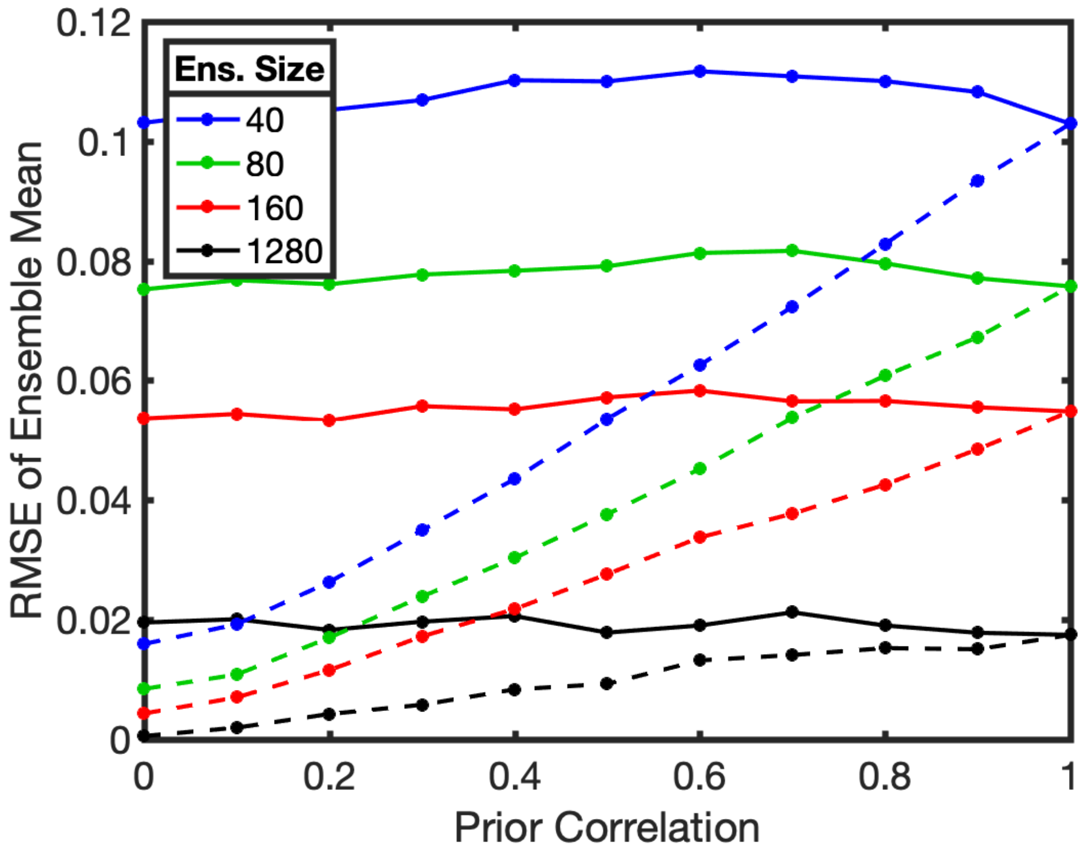
Gaussian Bivariate Results

Prior is random draw from bivariate gaussian.
Mean 0.
Variance 1.
Specified covariance.

One variable is observed, error variance 1.

Look at posterior statistics averaged over many cases.

Gaussian Bivariate Results

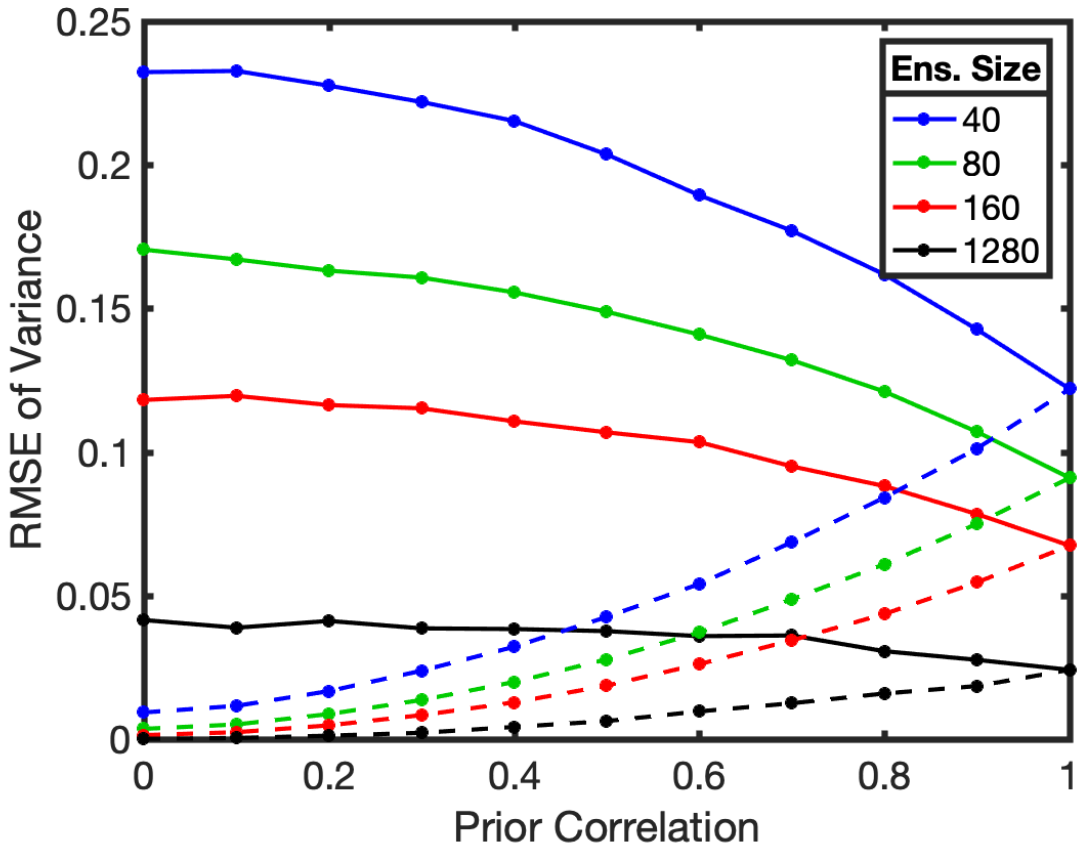


Dashed: EAKF.

Solid: State marginal RHF.

RMSE of posterior ensemble mean, unobserved variable. Marginal statistic, same for any pairing.

Gaussian Bivariate Results

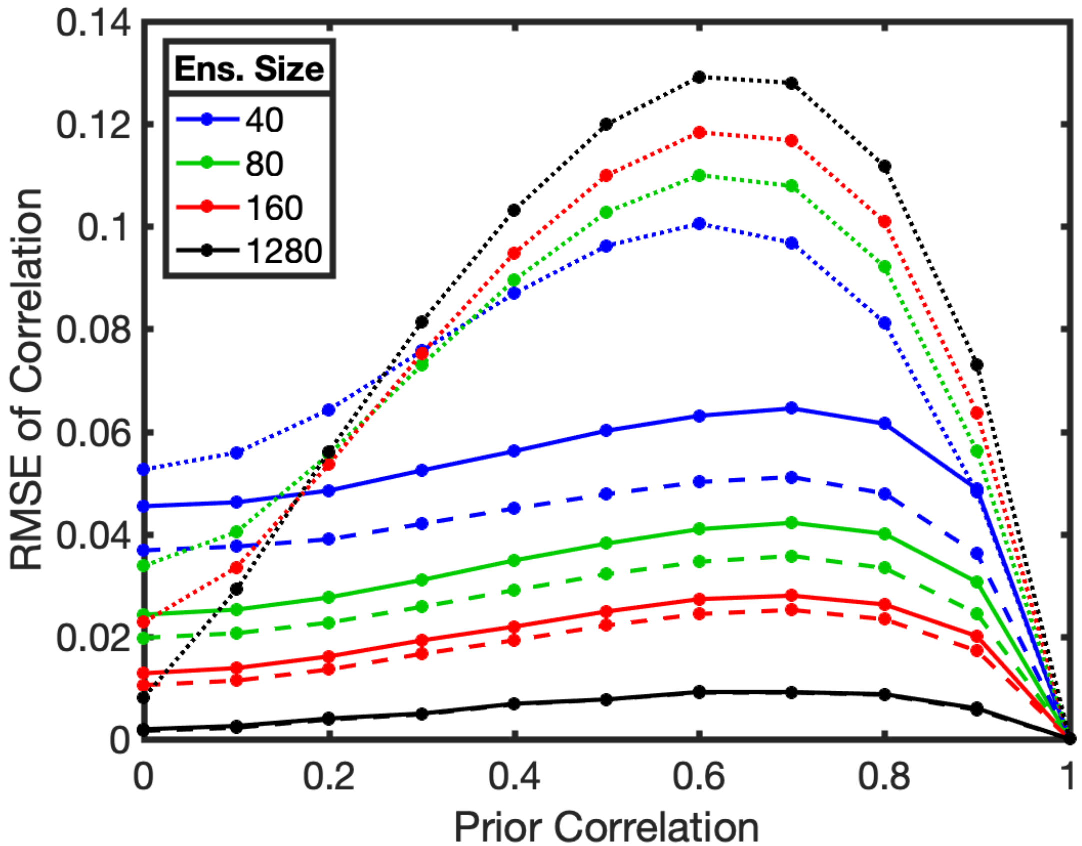


Dashed: EAKF.

Solid: State marginal RHF.

RMSE of posterior variance, unobserved variable.
Marginal statistic, same for any pairing.

Gaussian Bivariate Results



Dashed: EAKF.

Dotted: State marginal RHF.

Solid: MARHF.

RMSE of posterior correlation.
Bivariate statistic, pairing matters.

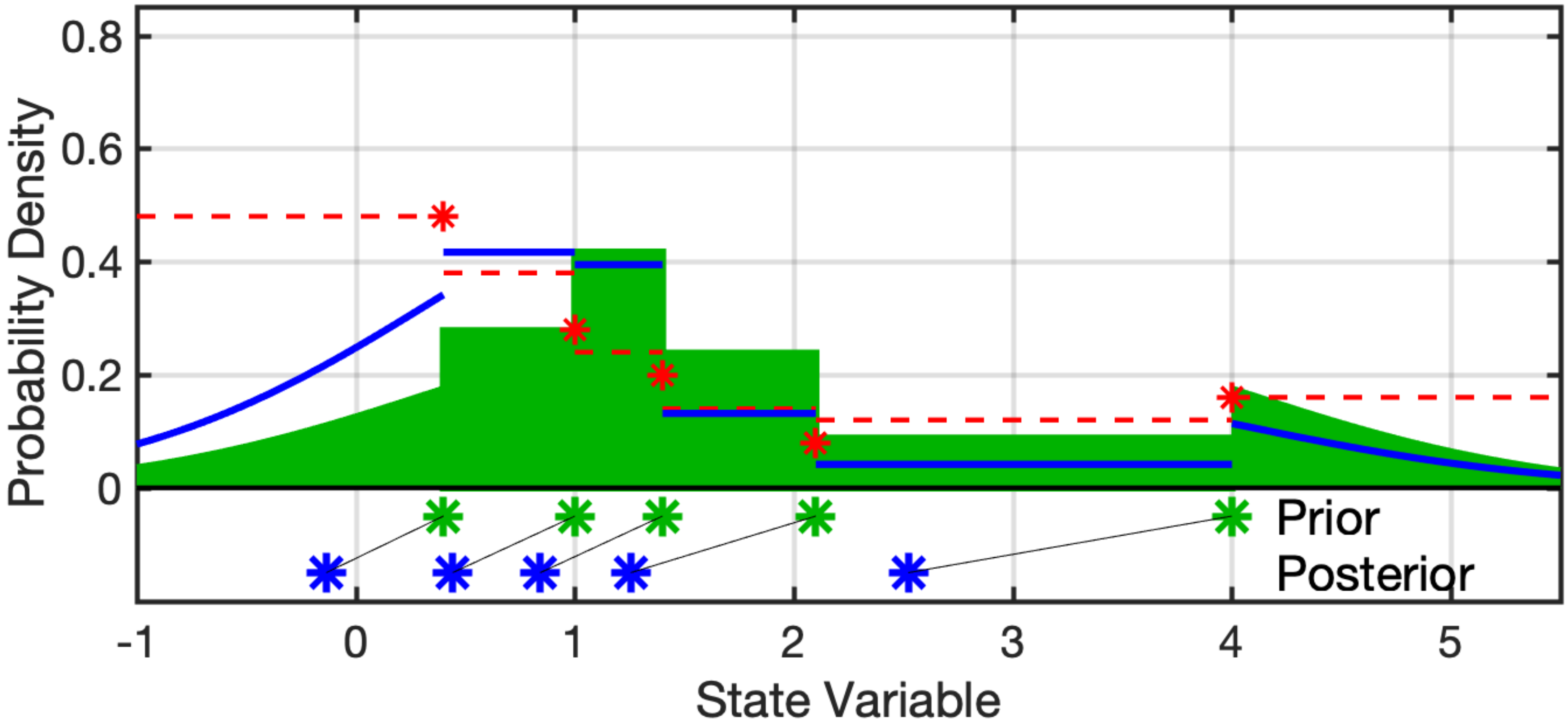
Additional RHF/MARHF Capabilities

Enforce additional prior constraints.

For instance, boundedness.

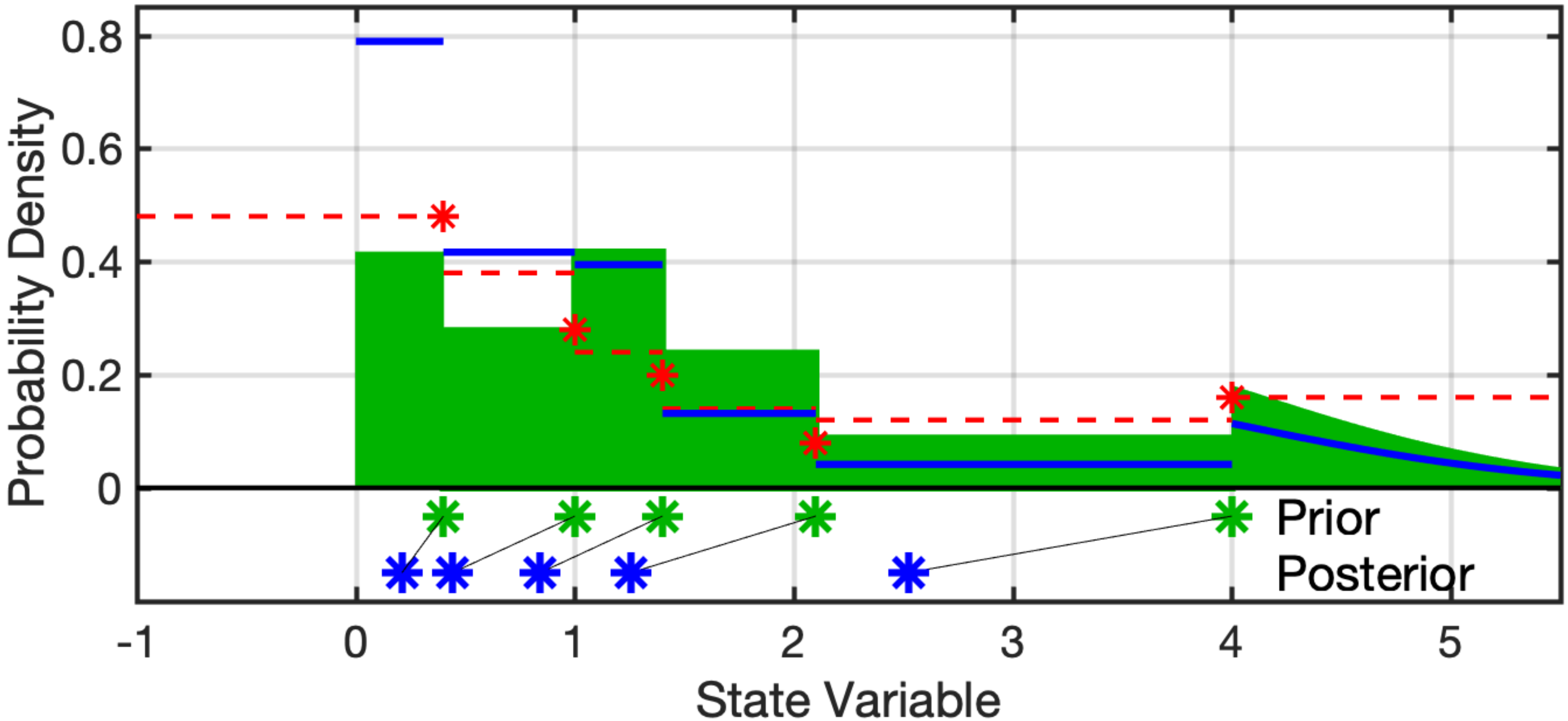
Use arbitrary likelihoods.

Standard RHF State Marginal.



RHF/MARHF with Bounded Prior

RHF State Marginal, same ensemble but positive prior.



Bounded State, Non-Gaussian Likelihoods

Bivariate example.

Prior of log is bivariate Gaussian, so prior is non-negative.

One variable observed.

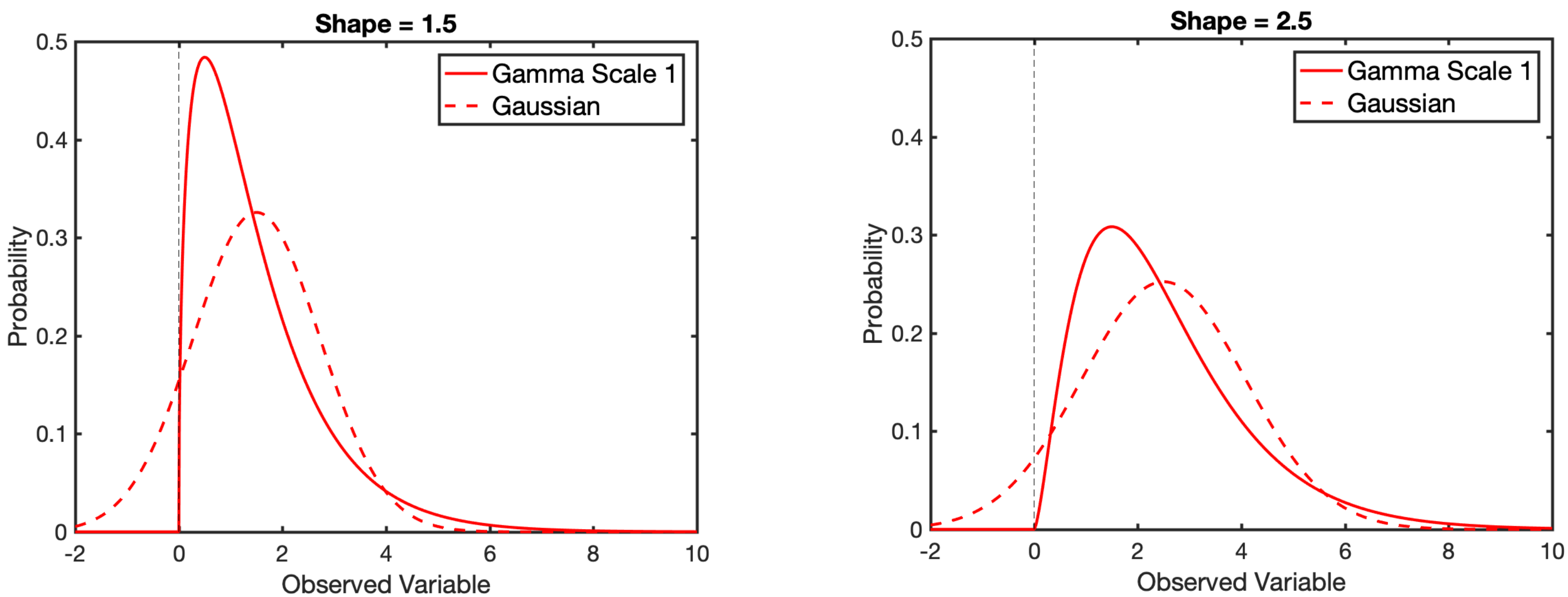
Likelihood is Gamma.

Shape parameter is same as first prior ensemble.

Scale parameter is 1.

Assimilate single observation for many random priors.

Compare Gamma likelihood to Gaussian approximation.



Data Assimilation Research Testbed

Bounded State, Non-Gaussian Likelihoods

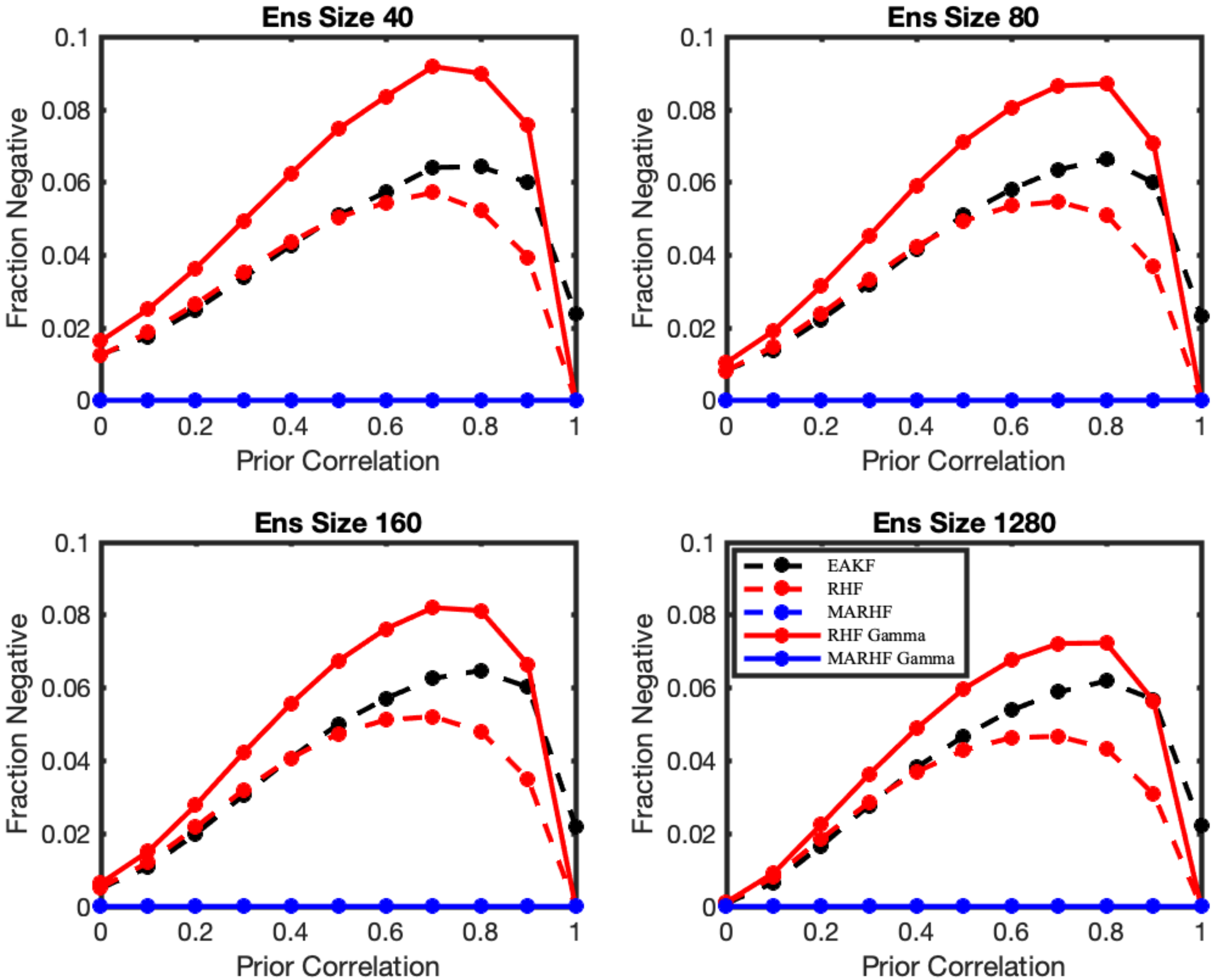
Compare 5 Methods

Observed Var.	Unobserved Var.	Likelihood
EAKF	Regression	Gaussian
RHF	Regression	Gaussian
RHF	MARHF	Gaussian
RHF	Regression	Gamma
RHF	MARHF	Gamma

Percent Negative Posterior Members

Blue = MARHF

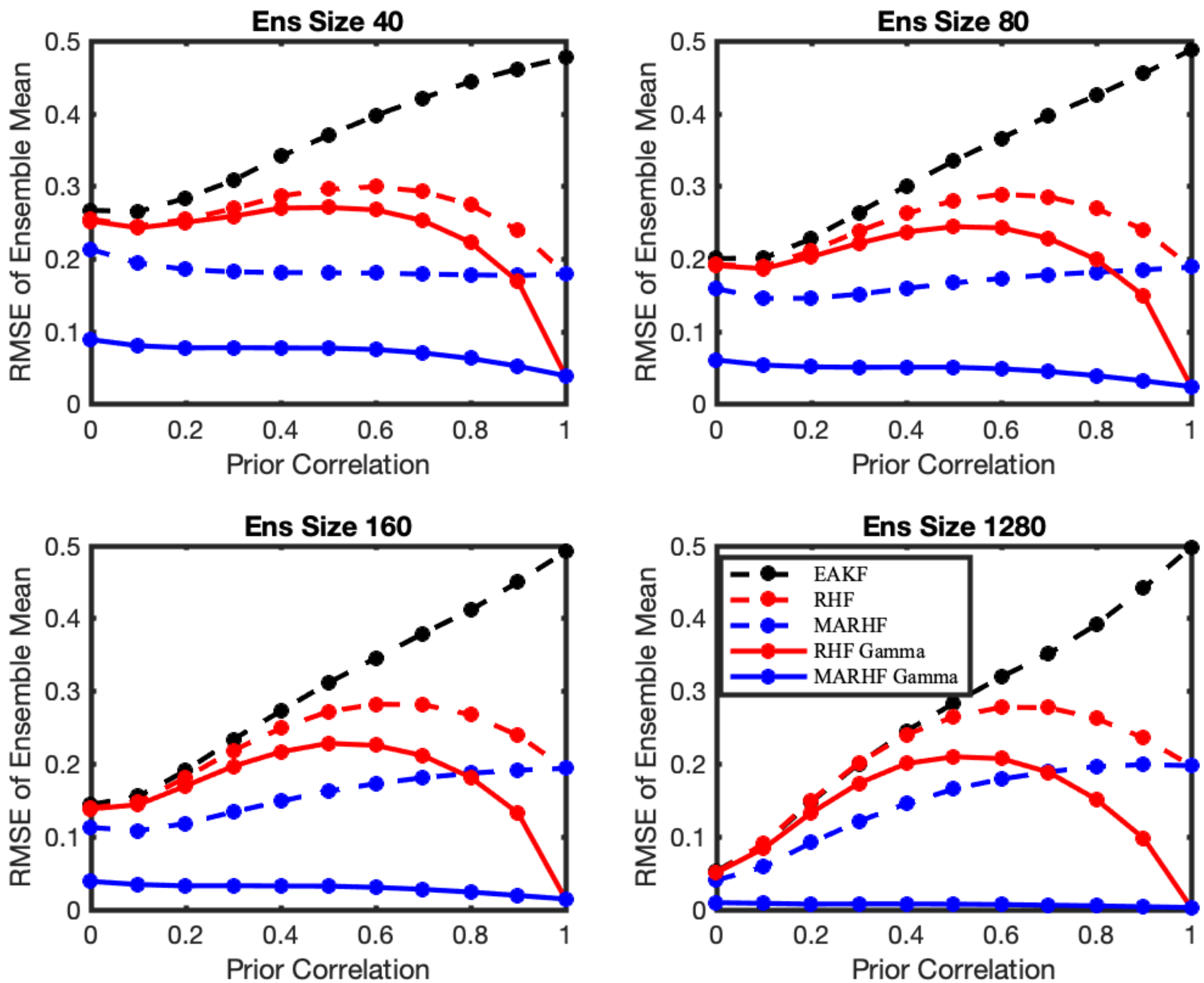
Solid = Gamma Likelihood



RMSE of Posterior Ensemble Mean

Blue = MARHF

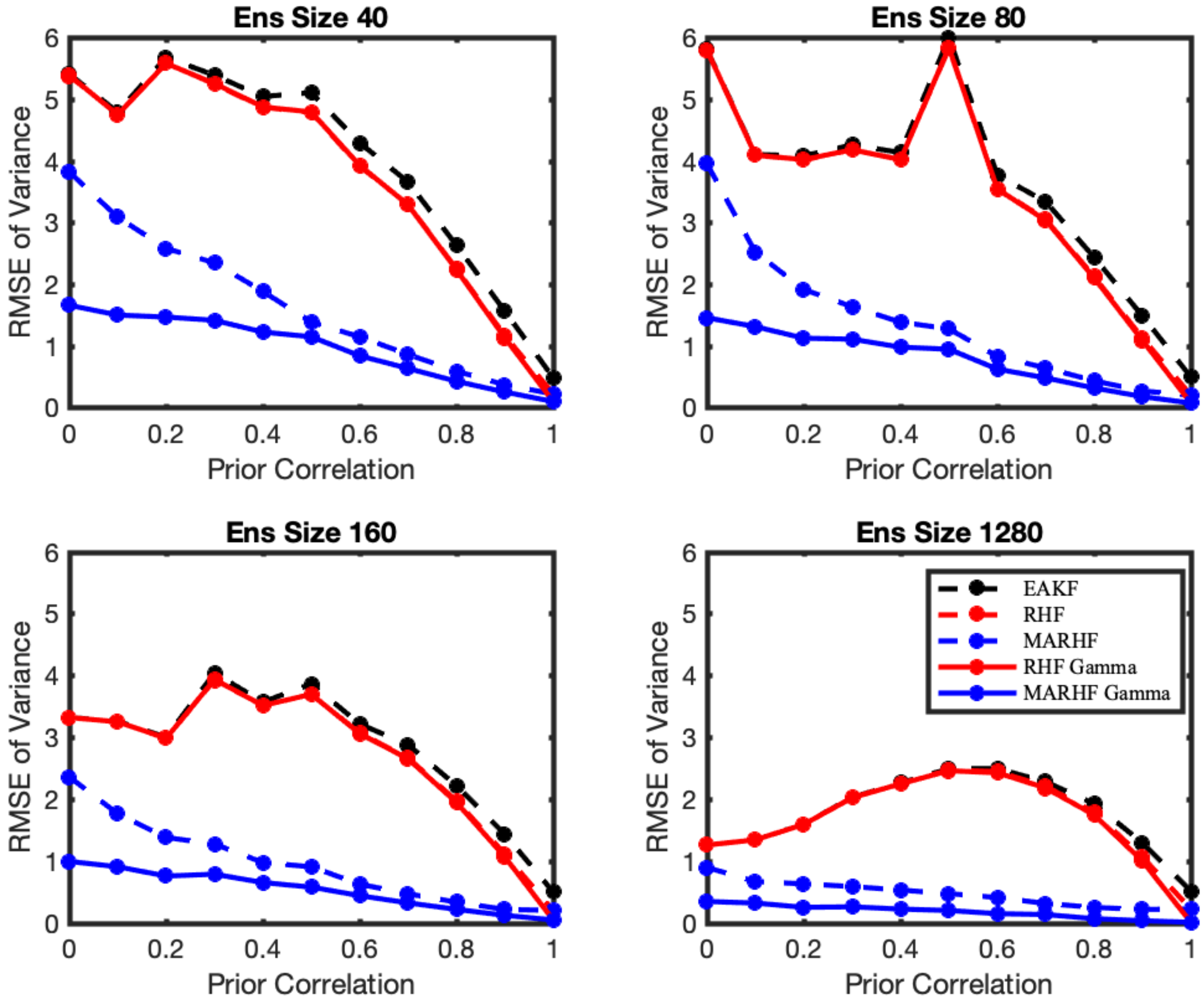
Solid = Gamma Likelihood



RMSE of Posterior Variance

Blue = MARHF

Solid = Gamma Likelihood



Summary

RHF filters represent non-Gaussian priors, posteriors.

MARHF allows limited non-linearity.

Particularly applicable to bounded quantities.

MARHF more expensive, but less than factor of 2.

General data assimilation theme:

Find algorithms with power of particle filters,
but much reduced cost of ensemble methods.

All results here with DARTLAB tools freely available in DART.

www.image.ucar.edu/DAReS/DART

Anderson, J., Hoar, T., Raeder, K., Liu, H., Collins, N., Torn, R., Arellano, A., 2009: *The Data Assimilation Research Testbed: A community facility.* BAMS, **90**, 1283—1296, doi: 10.1175/2009BAMS2618.1