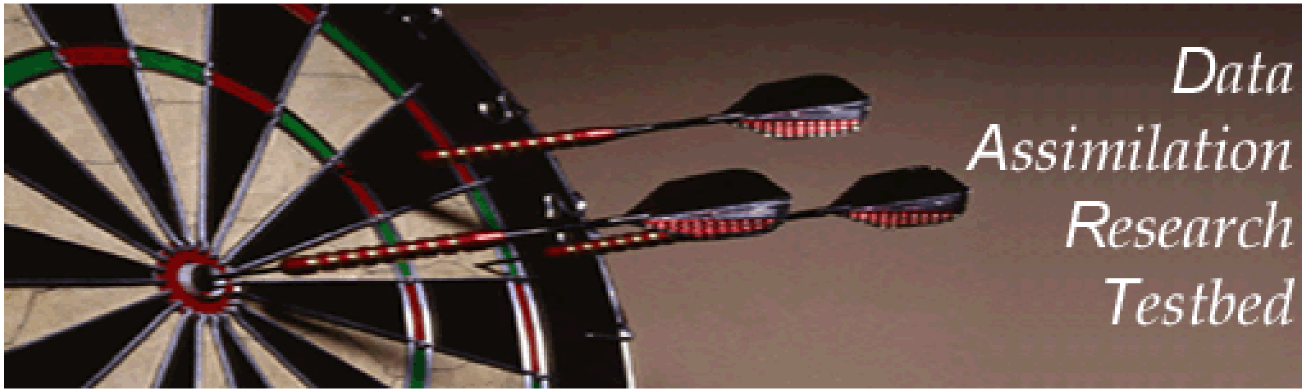
Data Assimilation Research Testbed Tutorial

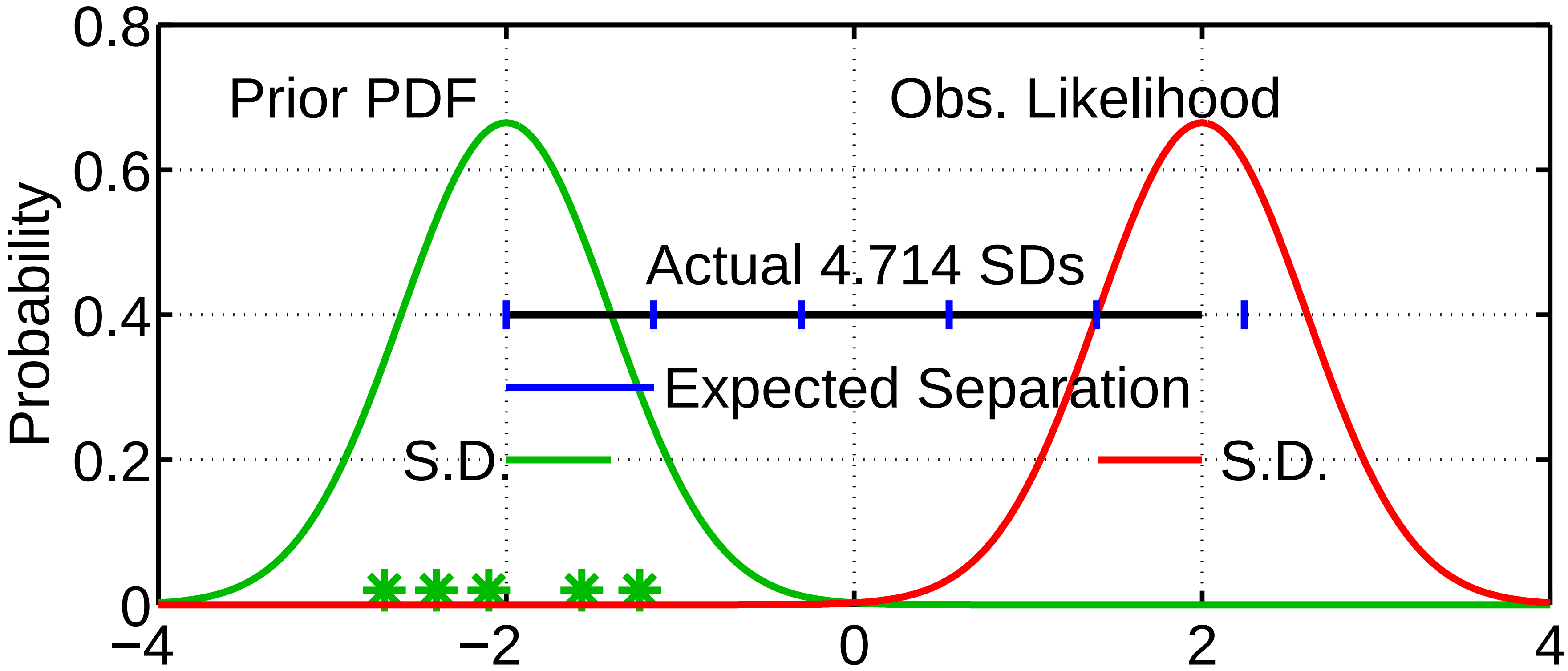


Section 12: Adaptive Inflation in Observation Space

Version 1.0: June, 2005

Variance inflation for Observations: An Adaptive Error Tolerant Filter

1. For observed variable, have estimate of prior-observed inconsistency



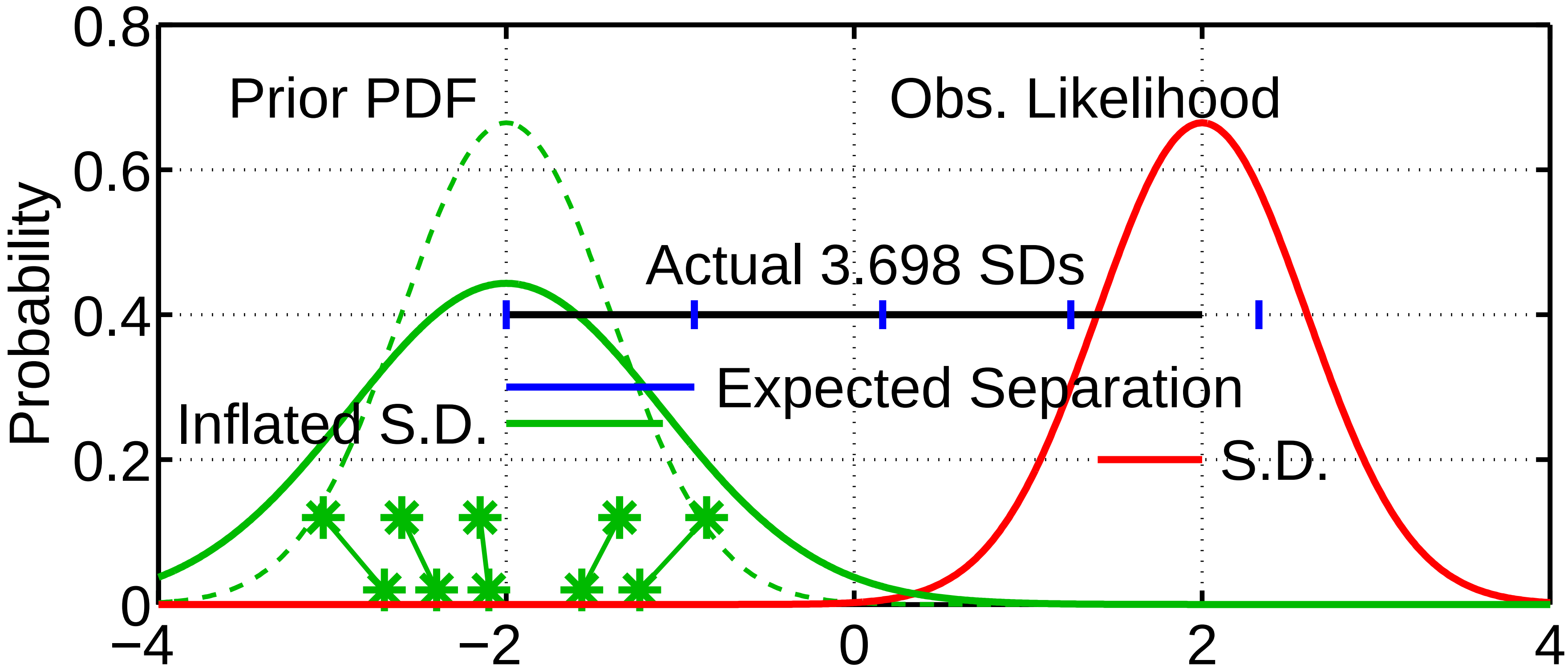
2. Expected(prior mean - observation) = $\sqrt{\sigma^2_{prior} + \sigma^2_{obs}}$.

Assumes that prior and observation are supposed to be unbiased.

Is it model error or random chance?

Variance inflation for Observations: An Adaptive Error Tolerant Filter

1. For observed variable, have estimate of prior-observed inconsistency



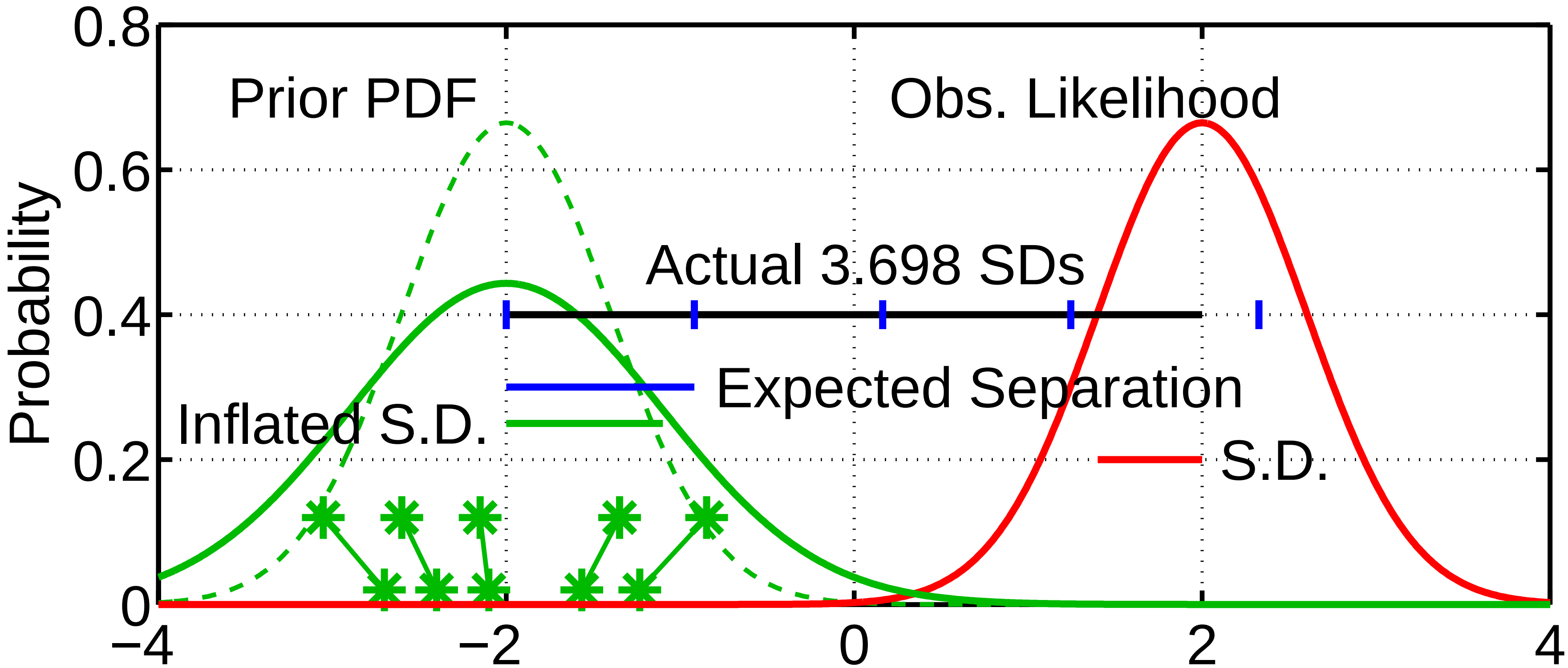
2. Expected(prior mean - observation) = $\sqrt{\sigma^2_{prior} + \sigma^2_{obs}}$.

3. Inflating increases expected separation.

Increases 'apparent' consistency between prior and observation.

Variance inflation for Observations: An Adaptive Error Tolerant Filter

1. For observed variable, have estimate of prior-observed inconsistency

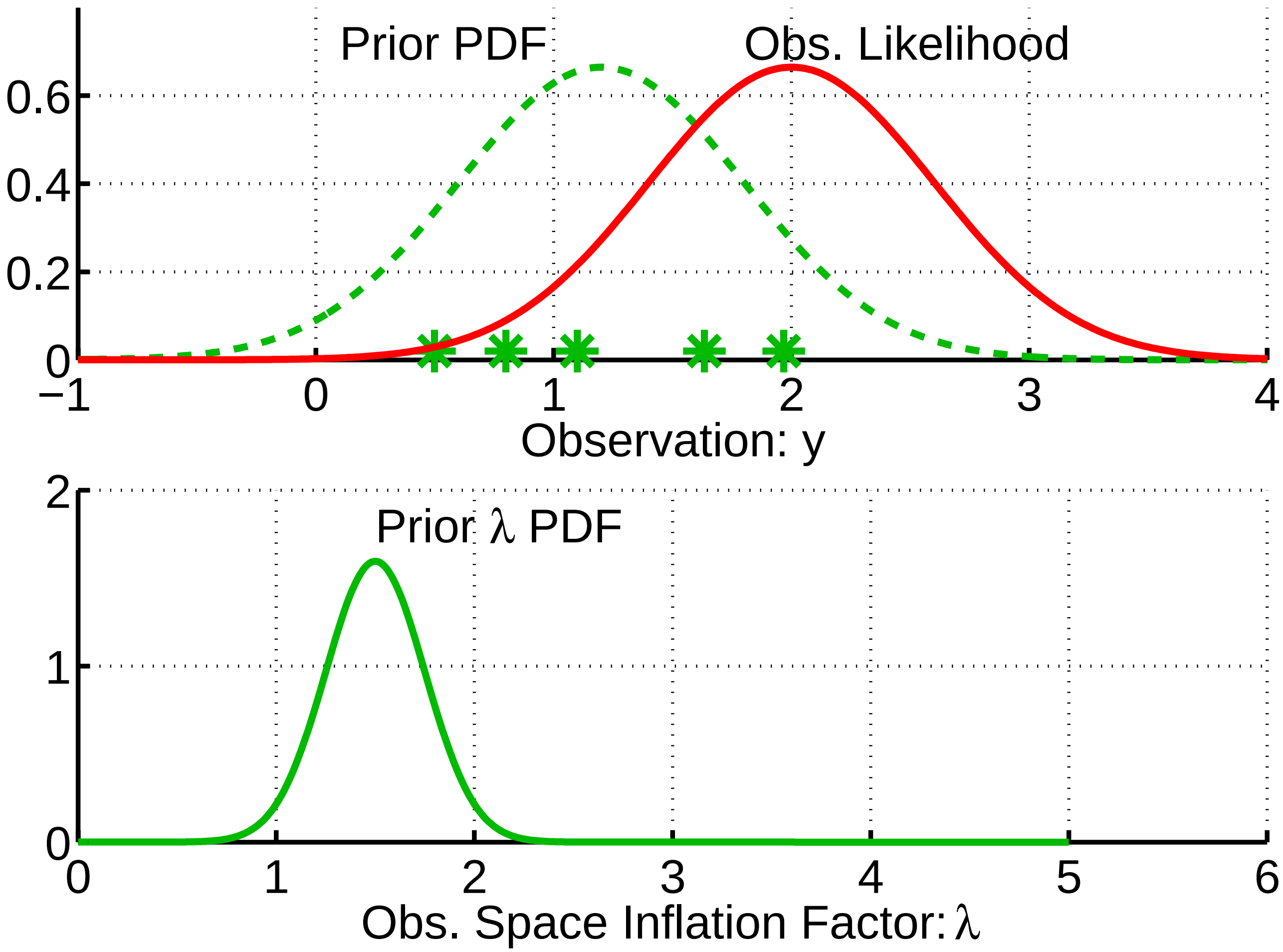


Distance, D, from prior mean y to obs. is $N\left(0, \sqrt{\lambda\sigma_{prior}^2 + \sigma_{obs}^2}\right) = N(0, \theta)$

Prob. y_o is observed given λ: $p(y_o | \lambda) = (2\Pi\theta)^{-1/2}\exp(-D^2/2\theta^2)$

Variance inflation for Observations: An Adaptive Error Tolerant Filter

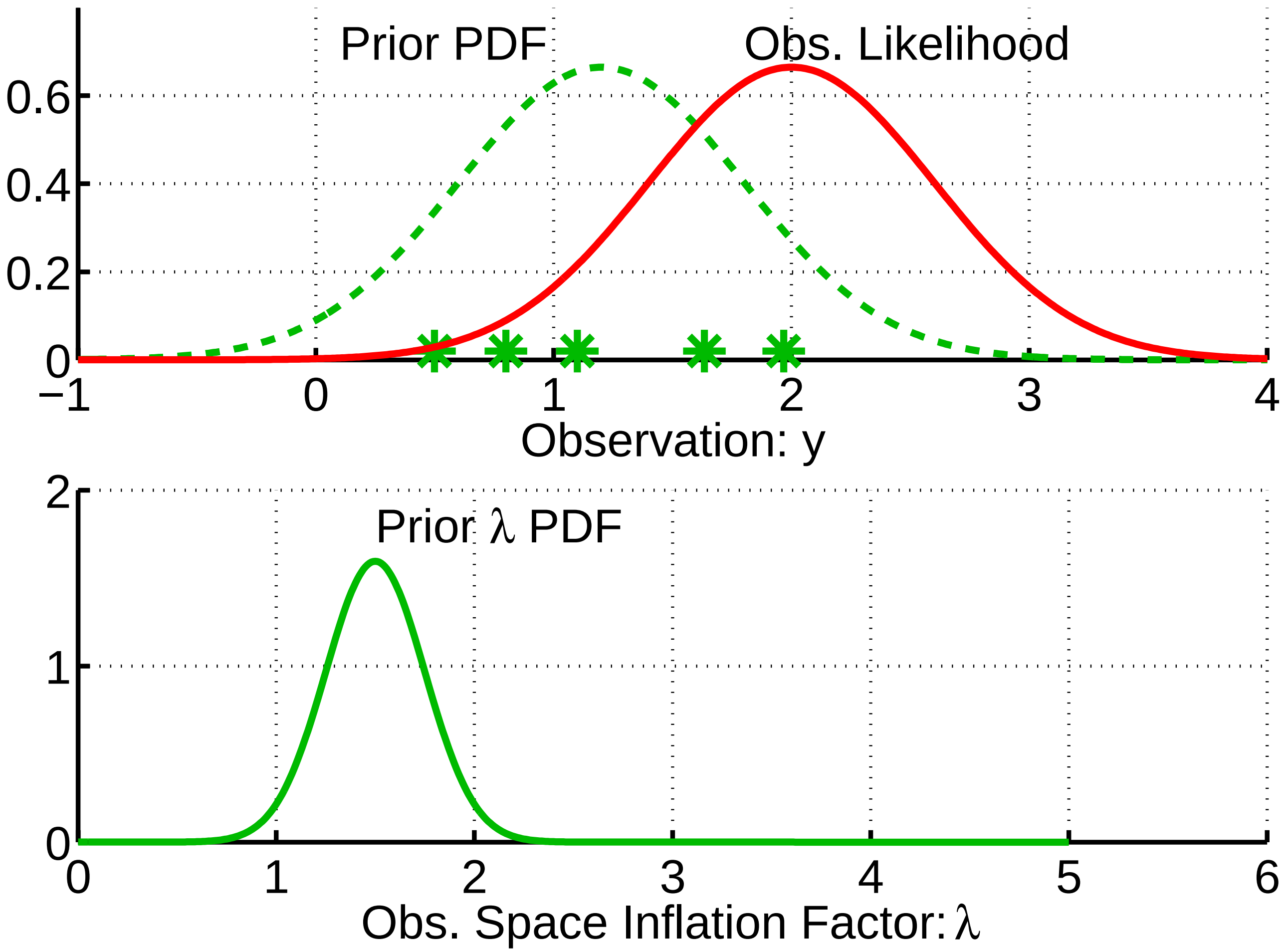
Use Bayesian statistics to get estimate of inflation factor, λ.



Assume some form for prior distribution for λ (Gaussian, gamma).
(Could assume other type of distribution or even use ensemble).

Variance inflation for Observations: An Adaptive Error Tolerant Filter

Use Bayesian statistics to get estimate of inflation factor, λ.



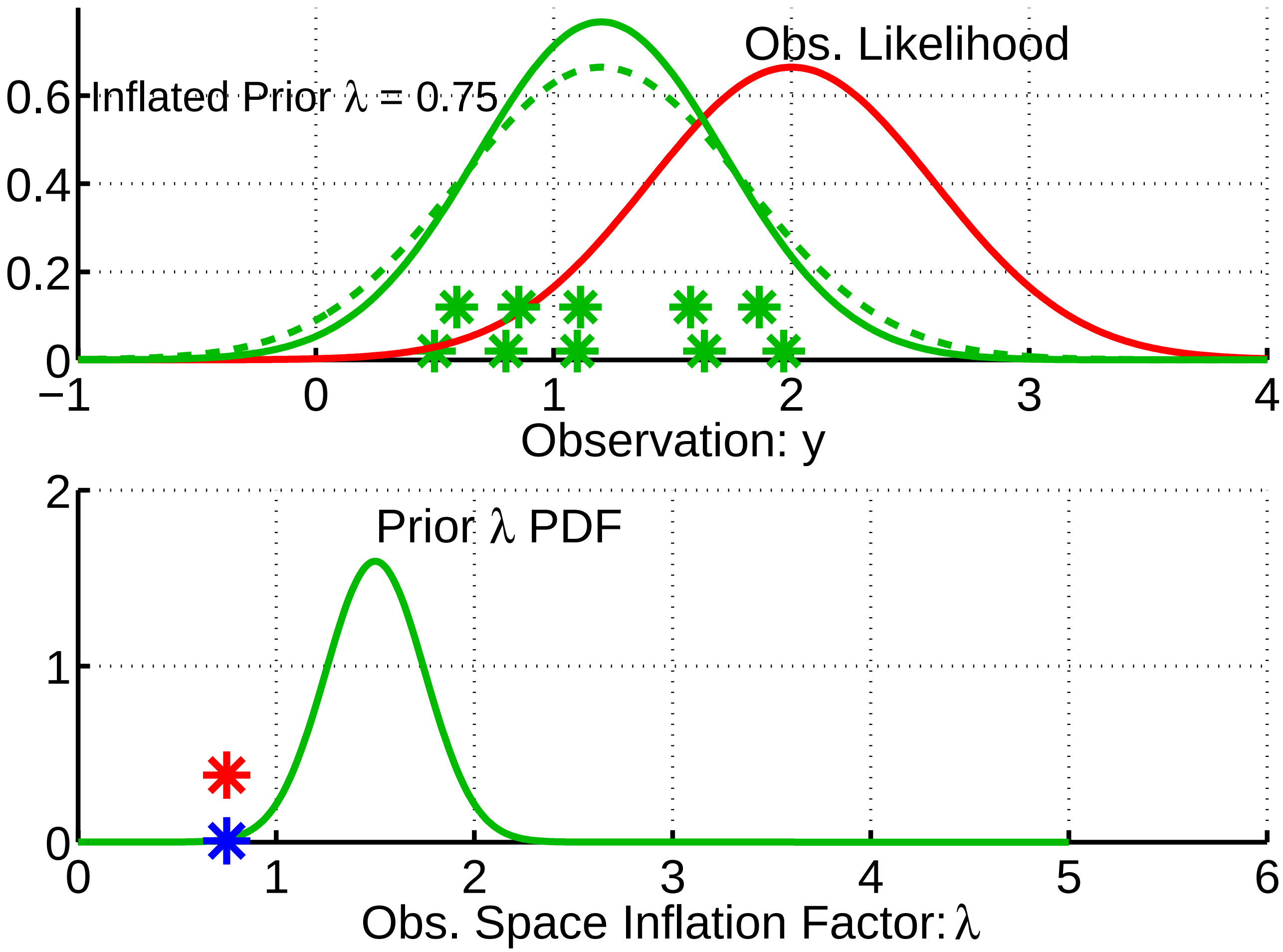
We've assumed a form for prior PDF $p(\lambda, t_k | Y_{t_{k-1}})$.

Recall that $p(y_k | \lambda)$ can be evaluated from normal PDF.

$$p(\lambda, t_k | Y_{t_k}) = p(y_k | \lambda) p(\lambda, t_k | Y_{t_{k-1}}) / normalization.$$

Variance inflation for Observations: An Adaptive Error Tolerant Filter

Use Bayesian statistics to get estimate of inflation factor, λ.



Get $p(y_k | \lambda = 0.75)$

from normal PDF.

Multiply by

$p(\lambda = 0.75, t_k | Y_{t_{k-1}})$

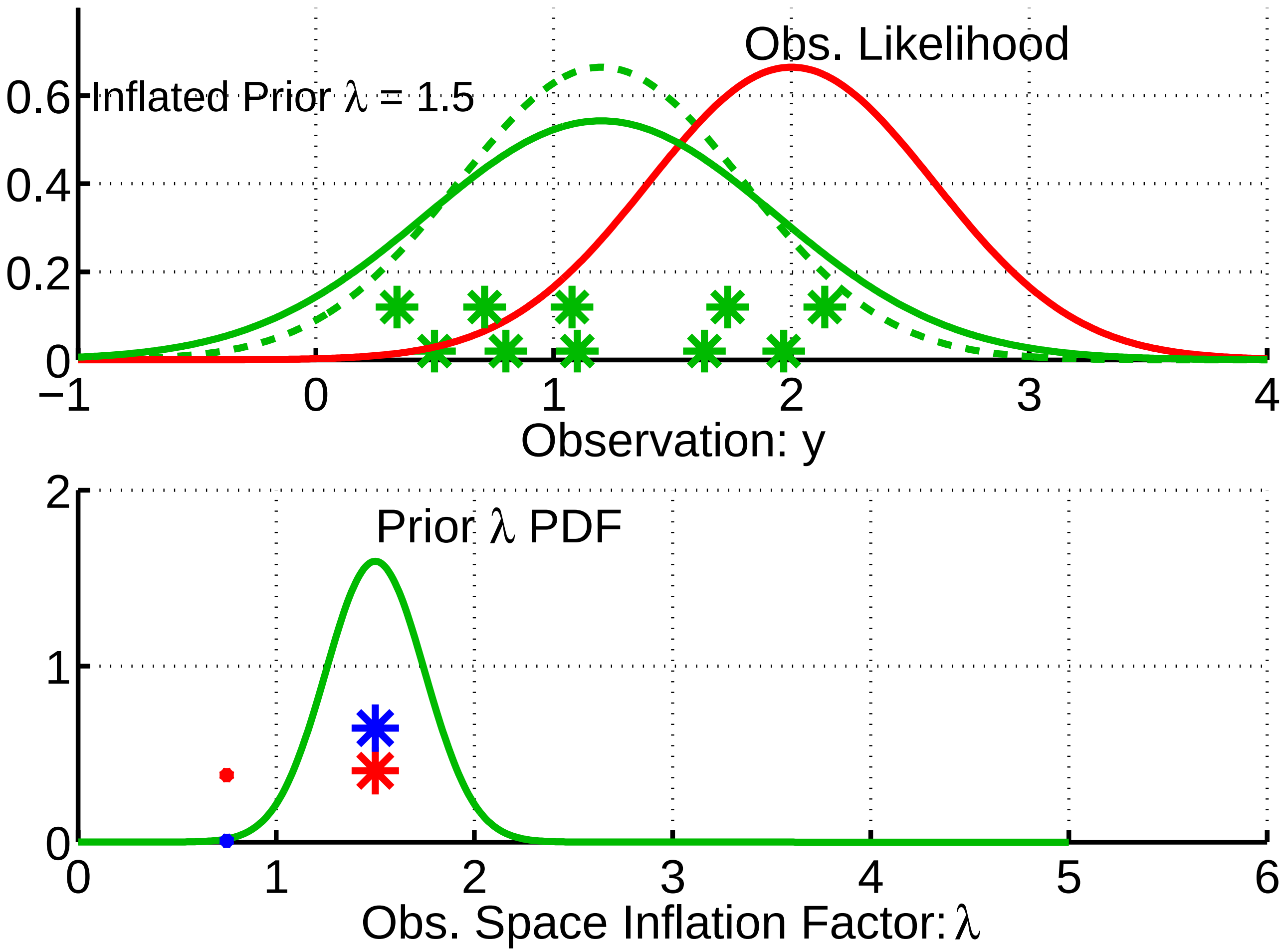
to get

$p(\lambda = 0.75, t_k | Y_{t_k})$

$$p(\lambda, t_k | Y_{t_k}) = p(y_k | \lambda) p(\lambda, t_k | Y_{t_{k-1}}) / normalization.$$

Variance inflation for Observations: An Adaptive Error Tolerant Filter

Use Bayesian statistics to get estimate of inflation factor, λ.



Get $p(y_k|\lambda = 1.50)$ from normal PDF.

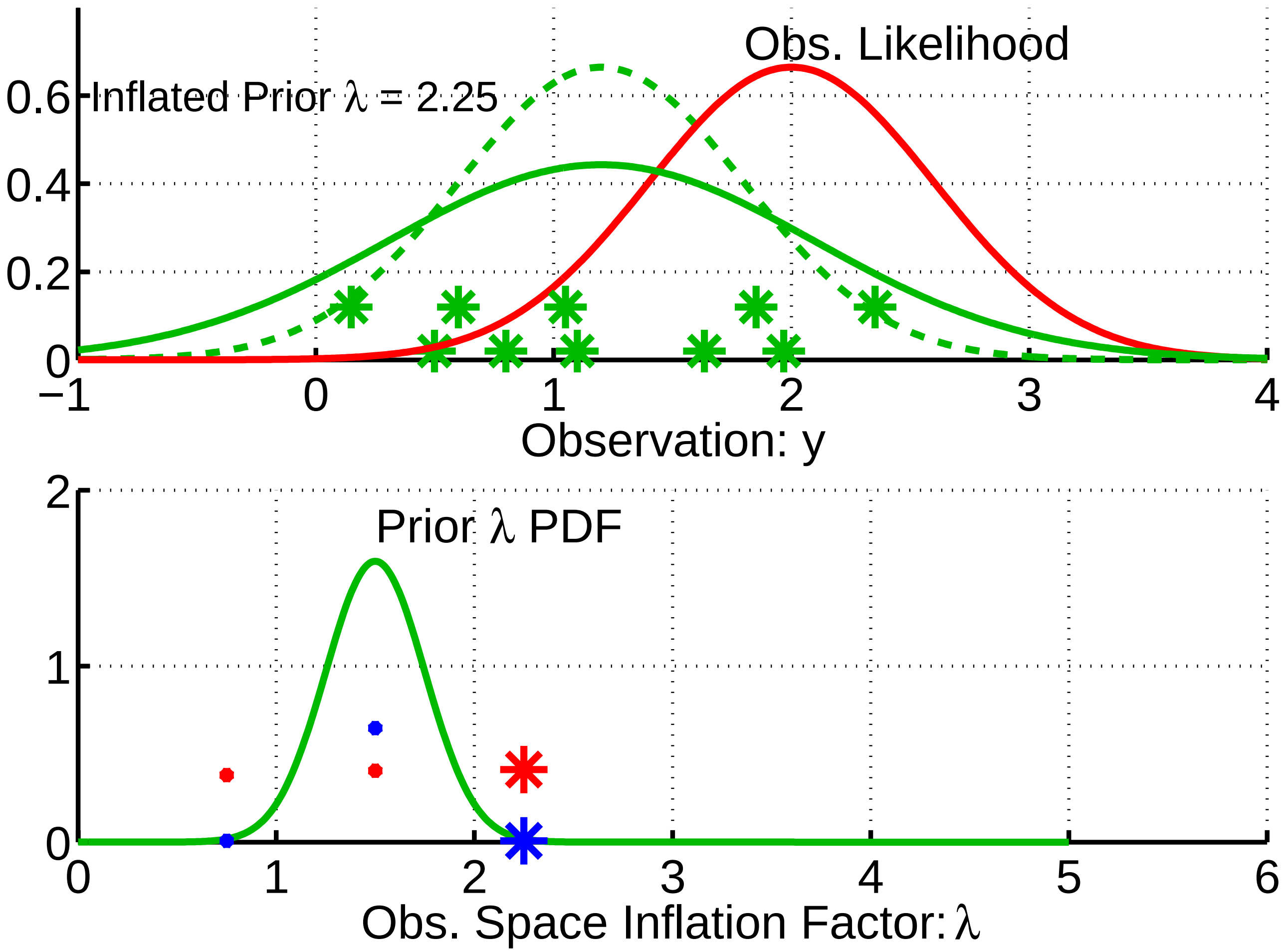
Multiply by $p(\lambda = 1.50, t_k|Y_{t_{k-1}})$

to get $p(\lambda = 1.50, t_k|Y_{t_k})$

$$p(\lambda, t_k|Y_{t_k}) = p(y_k|\lambda)p(\lambda, t_k|Y_{t_{k-1}})/normalization.$$

Variance inflation for Observations: An Adaptive Error Tolerant Filter

Use Bayesian statistics to get estimate of inflation factor, λ.



Get $p(y_k|\lambda = 2.25)$

from normal PDF.

Multiply by

$p(\lambda = 2.25, t_k|Y_{t_{k-1}})$

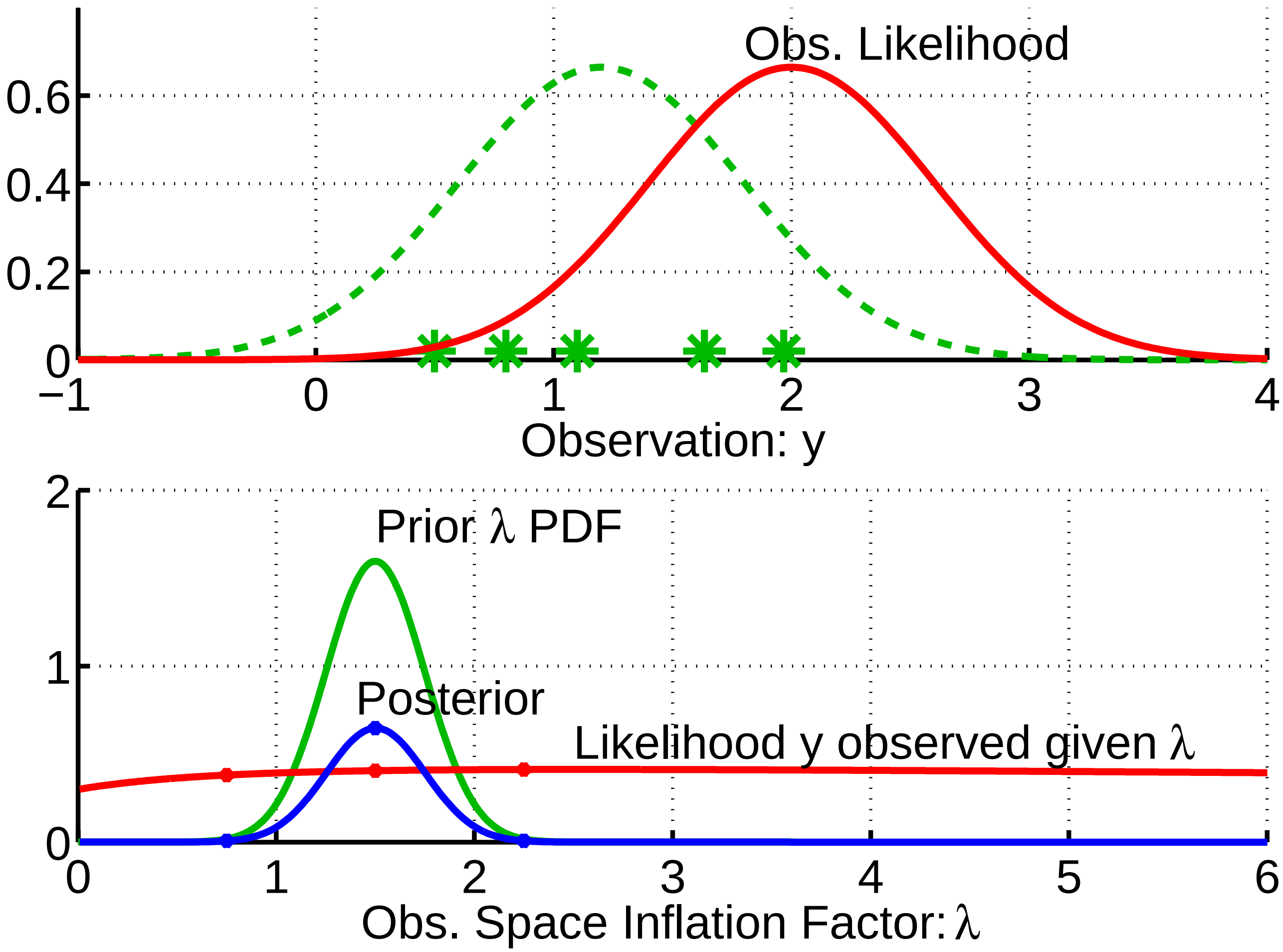
to get

$p(\lambda = 2.25, t_k|Y_{t_k})$

$$p(\lambda, t_k|Y_{t_k}) = p(y_k|\lambda)p(\lambda, t_k|Y_{t_{k-1}})/normalization.$$

Variance inflation for Observations: An Adaptive Error Tolerant Filter

Use Bayesian statistics to get estimate of inflation factor, λ.



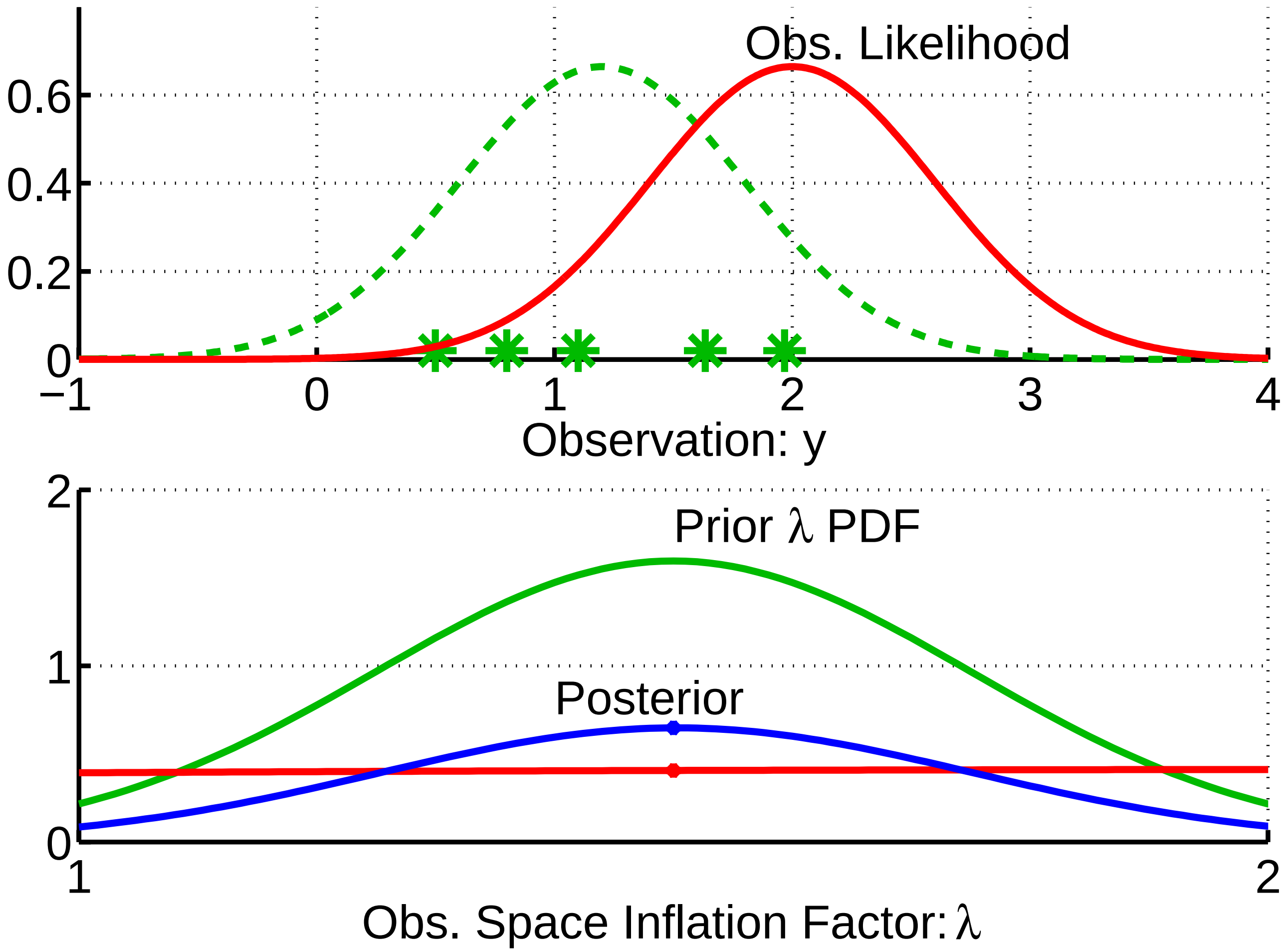
Repeat for a range of values of λ.

Now must get posterior in same form as prior.

$$p(\lambda, t_k | Y_{t_k}) = p(y_k | \lambda) p(\lambda, t_k | Y_{t_{k-1}}) / normalization.$$

Variance inflation for Observations: An Adaptive Error Tolerant Filter

Use Bayesian statistics to get estimate of inflation factor, λ.



Very little information about λ in a single observation.

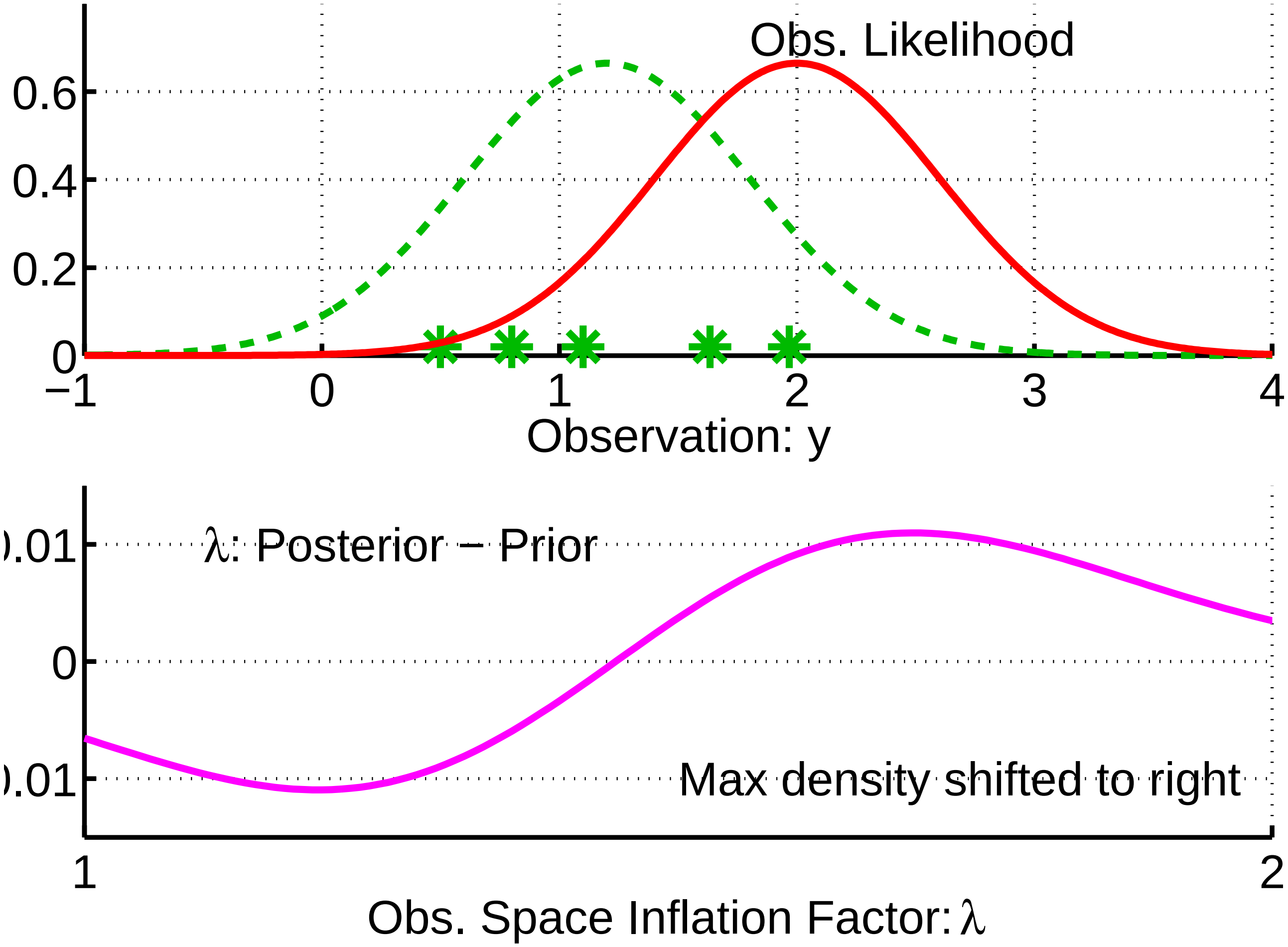
Posterior and prior are very similar.

Normalized posterior indistiguishable from prior.

$$p(\lambda, t_k | Y_{t_k}) = p(y_k | \lambda) p(\lambda, t_k | Y_{t_{k-1}}) / normalization.$$

Variance inflation for Observations: An Adaptive Error Tolerant Filter

Use Bayesian statistics to get estimate of inflation factor, λ.



Very little information about λ in a single observation.

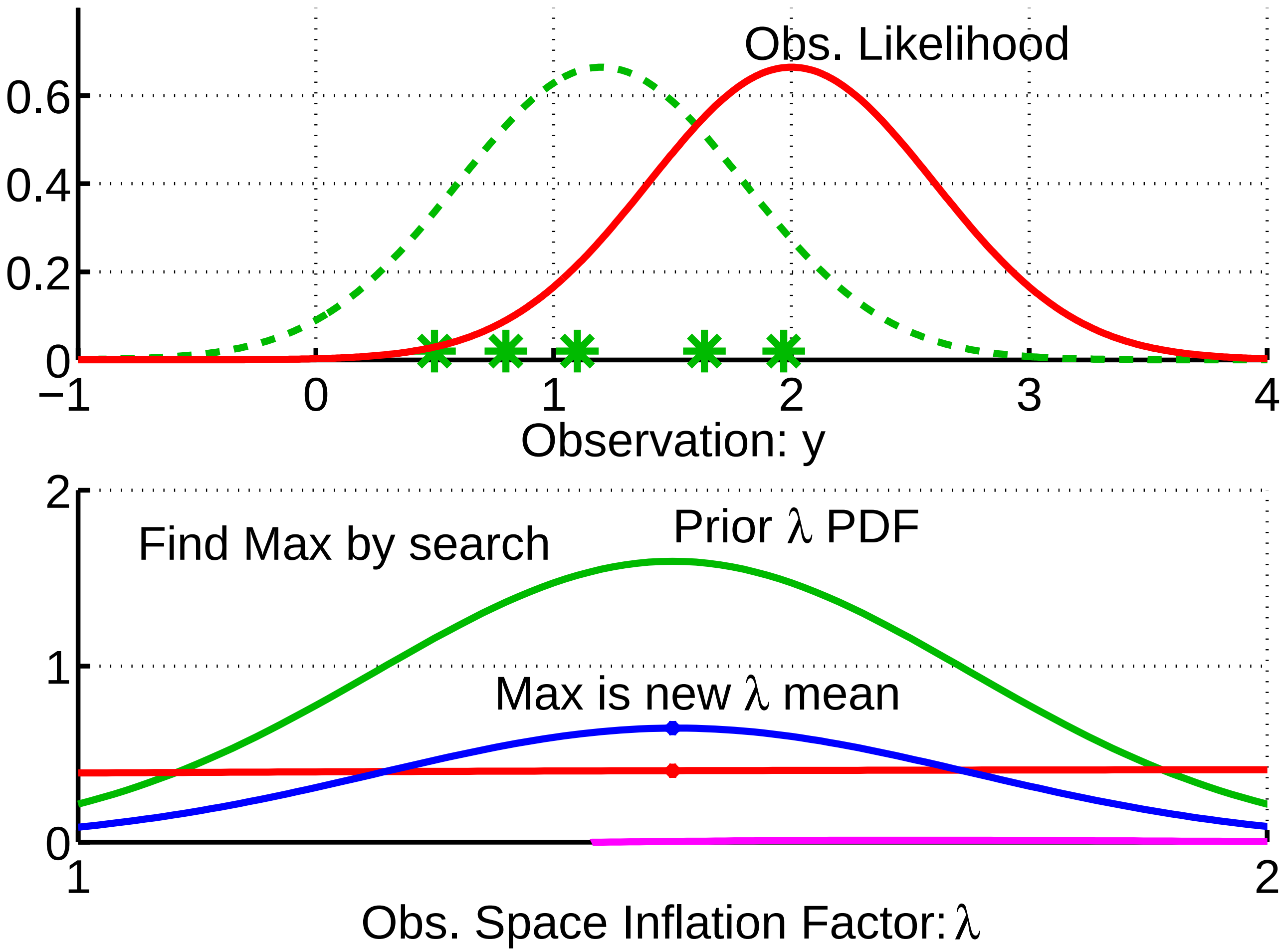
Posterior and prior are very similar.

Difference shows slight shift to larger values of λ.

$$p(\lambda, t_k | Y_{t_k}) = p(y_k | \lambda) p(\lambda, t_k | Y_{t_{k-1}}) / normalization.$$

Variance inflation for Observations: An Adaptive Error Tolerant Filter

Use Bayesian statistics to get estimate of inflation factor, λ.



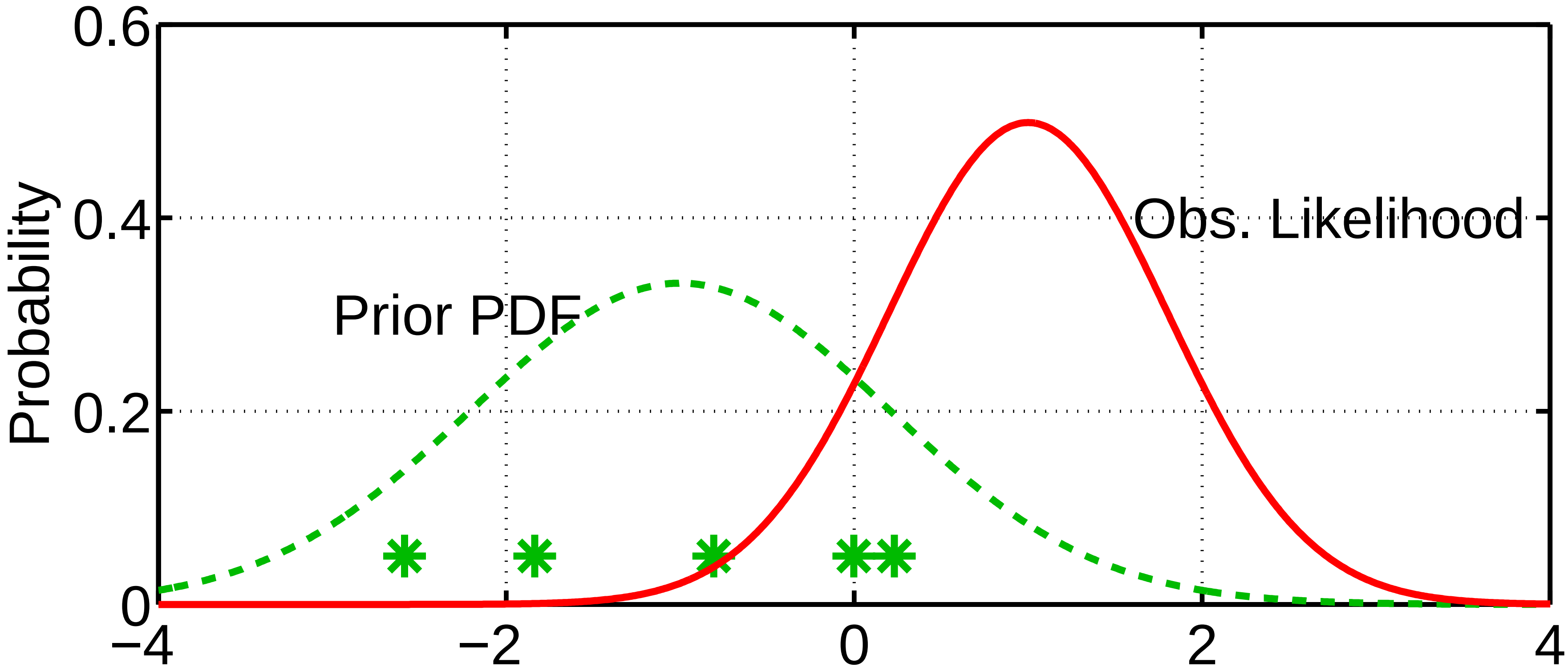
One option is to use Gaussian prior for λ.

Select max of posterior as mean of updated Gaussian.

Do a fit for updated standard deviation.

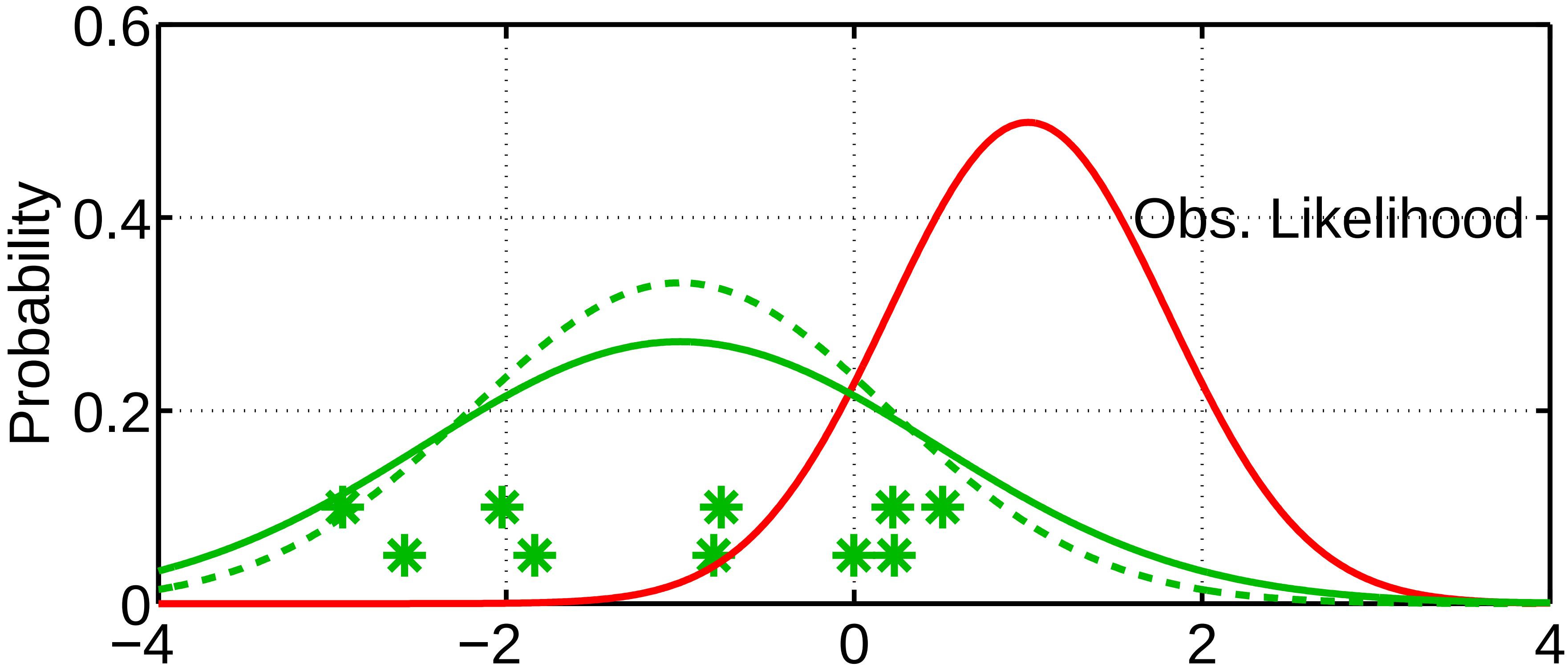
$$p(\lambda, t_k | Y_{t_k}) = p(y_k | \lambda) p(\lambda, t_k | Y_{t_{k-1}}) / normalization.$$

Observation Space Computations with Adaptive Error Correction



1. Compute updated inflation distribution, $p(\lambda, t_k | Y_{t_k})$.

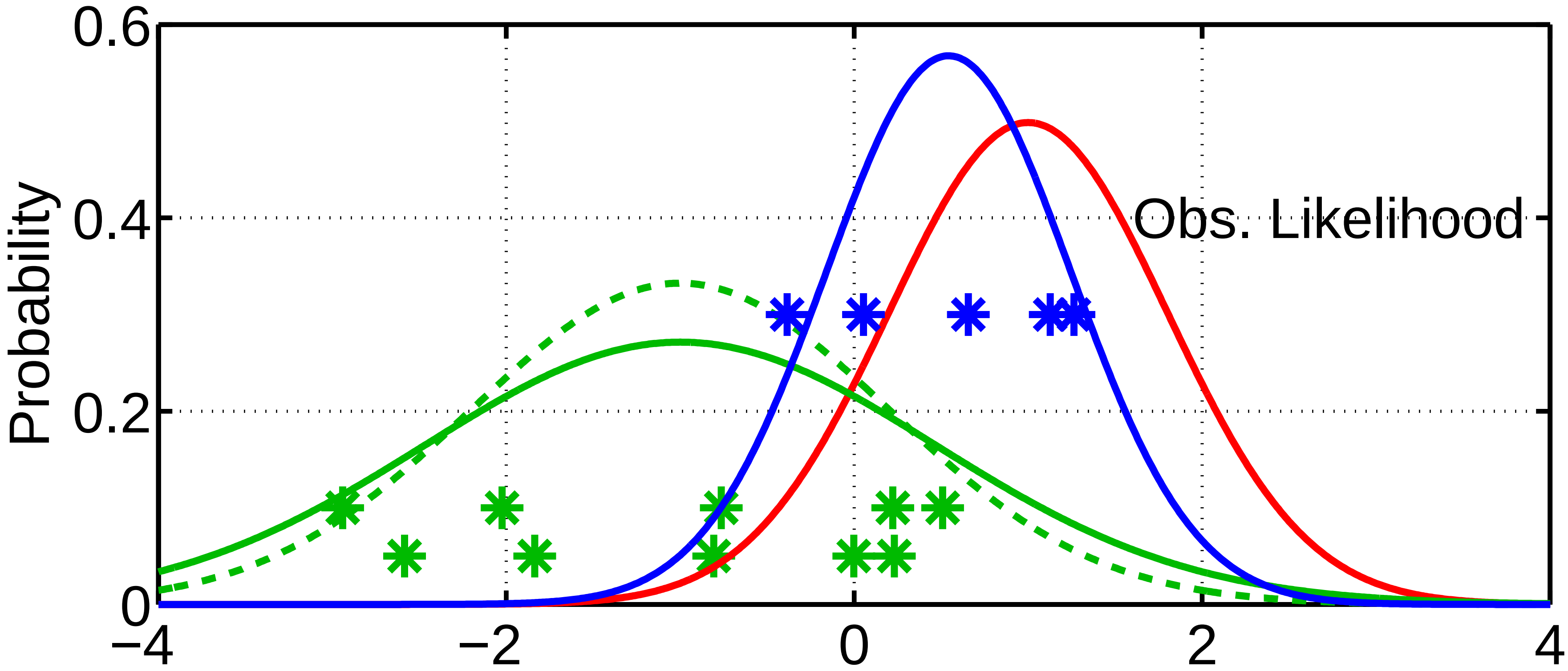
Observation Space Computations with Adaptive Error Correction



1. Compute updated inflation distribution, $p(\lambda, t_k | Y_{t_k})$.

2. Inflate ensemble using mean of updated λ distribution.

Observation Space Computations with Adaptive Error Correction

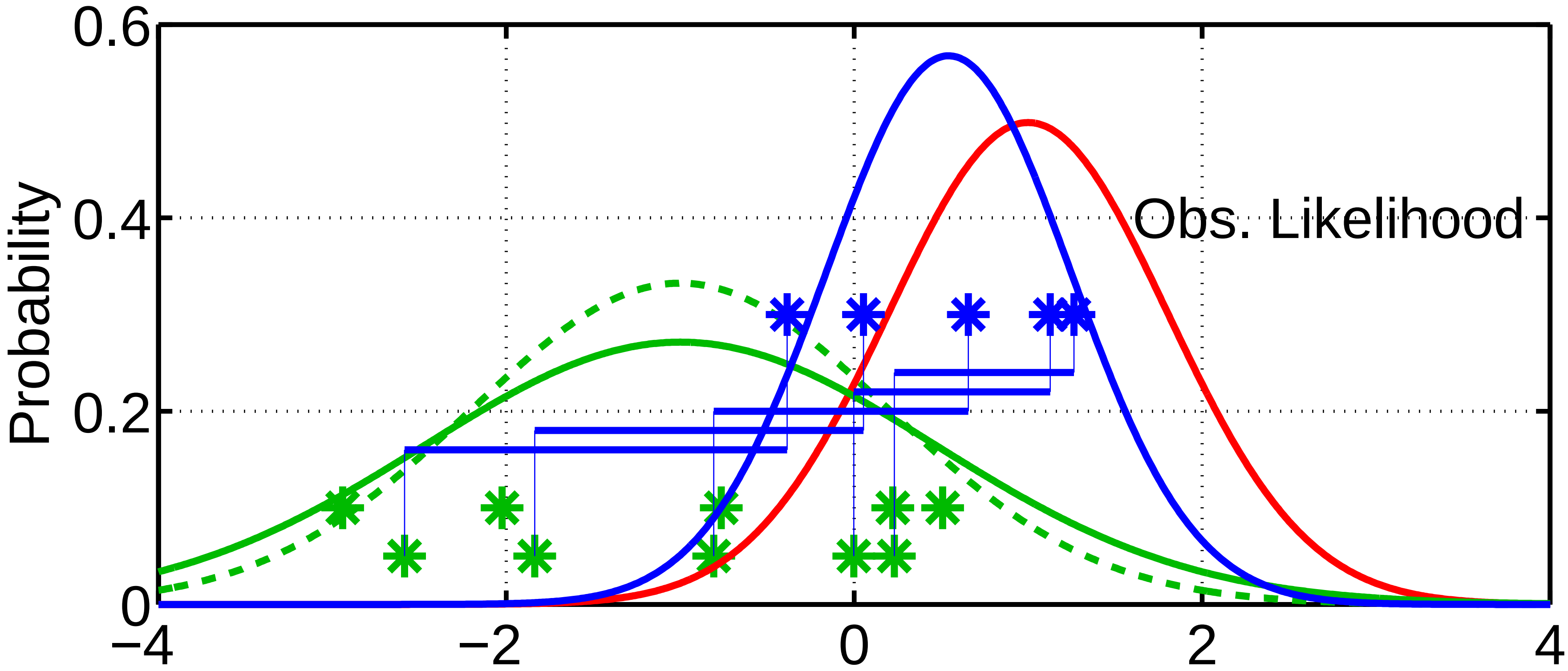


1. Compute updated inflation distribution, $p(\lambda, t_k | Y_{t_k})$.

2. Inflate ensemble using mean of updated λ distribution.

3. Compute posterior for y using inflated prior.

Observation Space Computations with Adaptive Error Correction



1. Compute updated inflation distribution, $p(\lambda, t_k | Y_{t_k})$.
2. Inflate ensemble using mean of updated λ distribution.
3. Compute posterior for y using inflated prior.
4. Compute increments from ORIGINAL prior ensemble.

Adaptive Observation Space Inflation in DART

Controlled by *cov_inflate*, *cov_inflate_sd*, *sd_lower_bound*, and *deterministic_cov_inflate* in *assim_tools_nml*.

Full implementation:

Set *cov_inflate* to positive initial value, for instance 1.0,

Set *cov_inflate_sd* to intitial value, for instance 0.20,

Set *sd_lower_bound* to 0.0, no limit on how small it can get.

Try this in Lorenz-96 (verify other aspects of input.nml).

To facilitate model error experiments, use 80 member ensemble.

(set *ens_size = 80* in *filter_nml*).

This is a very expensive algorithm.

Algorithmic variants:

1. Increase prior y variance by adding random gaussian noise.

   As opposed to ‘deterministics’ linear inflating.

   This is controlled by *deterministic_cov_inflate* in *assim_tools_nml*.

   True => inflate, False => random noise.

2. Just have a fixed value for obs. space λ

   Cheap, handles blow up of state vars unconstrained by obs.

   We already tried this in section 9.

Algorithmic variants:

3. Fix value of λ standard deviation.
   - Greatly reduces cost.
   - Avoids σ_λ getting small (error model filter divergence, Yikes!).
   - Have to have some intuition about the value for σ_λ.
   - This appears to be most viable option for large models.
   - Value of σ_λ = 0.05 works for very broad range of problems.
   - This is a sampling error closure problem (akin to turbulence).

To fix σ_λ, Set *cov_inflate* to positive initial value, for instance 1.0,
- Set *cov_inflate_sd* to fixed value, for instance 0.05,
- Set *sd_lower_bound* to same value as cov_inflate_sd.
  - (Can't get any smaller).

Try this in lorenz-96. Look at how the inflation varies.

Potential problems

1. Very heuristic.

2. Error model filter divergence (pretty hard to think about).

3. Equilibration problems, oscillations in λ with time.

4. Amplifying unwanted model resonances (gravity waves)

   Try turning this on in 9var model.

   Fixed 0.05 for *cov_inflate_sd*, *sd_lower_bound*.

   Behavior set by value of *cov_inflate* in assim_tools_nml.

Simulating Model Error in 40-Variable Lorenz-96 Model

Inflation can deal with all sorts of errors, including model error.

Can simulate model error in lorenz-96 by changing forcing.

Synthetic observations are from model with forcing = 8.0.

Use forcing in model_nml to introduce model error.

Try forcing values of 7, 6, 5, 3 with and without adaptive inflation.

The F = 3 model is periodic, looks very little like F = 8.

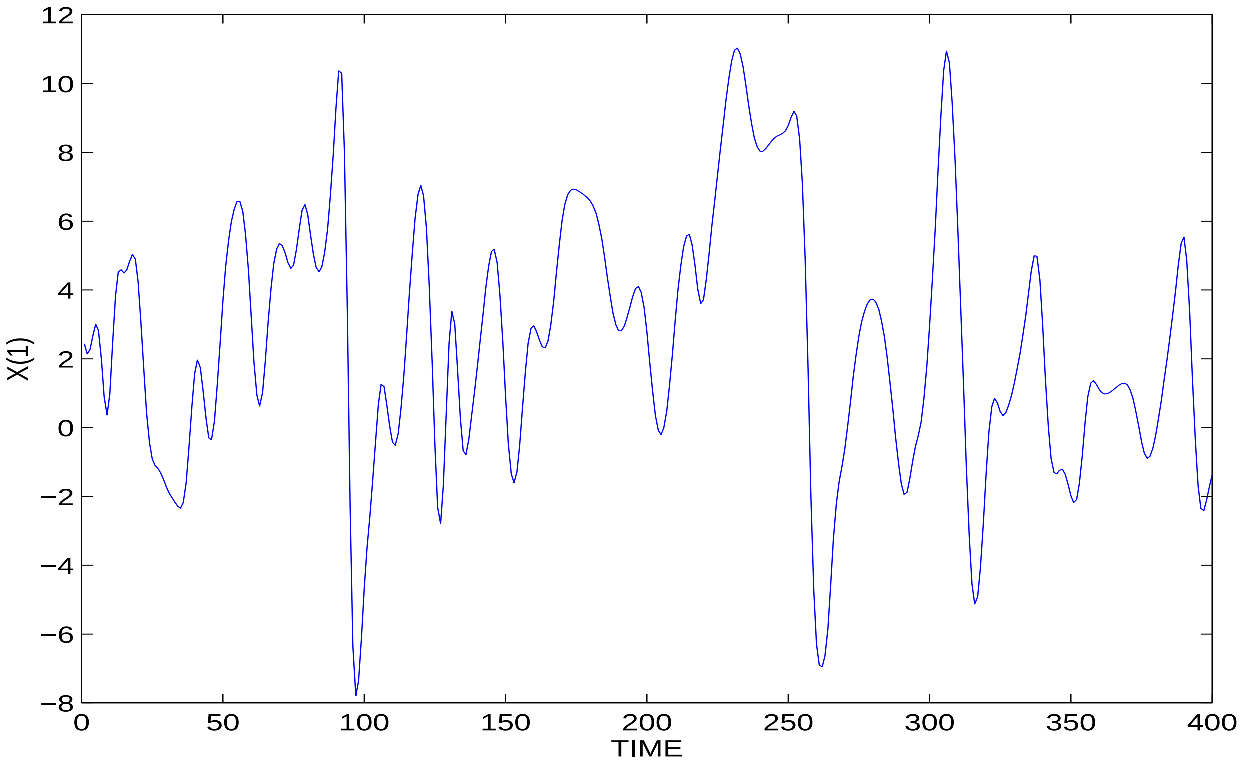
Simulating Model Error in 40-Variable Lorenz-96 Model

40 state variables: $X_1, X_2, \ldots, X_N$

$dX_i / dt = (X_{i+1} - X_{i-2})X_{i-1} - X_i + F;$

$i = 1, \ldots, 40$ with cyclic indices

Use $F = 8.0$, 4th-order Runge-Kutta with $dt=0.05$



Time series of state variable from free L96 integration

Experimental design: Lorenz-96 Model Error Simulation

Truth and observations comes from long run with F=8

200 randomly located (fixed in time) ‘observing locations’

Independent 1.0 observation error variance

Observations every hour

σ_λ is 0.05, mean of λ adjusts but variance is fixed

4 groups of 20 members each (80 ensemble members total)

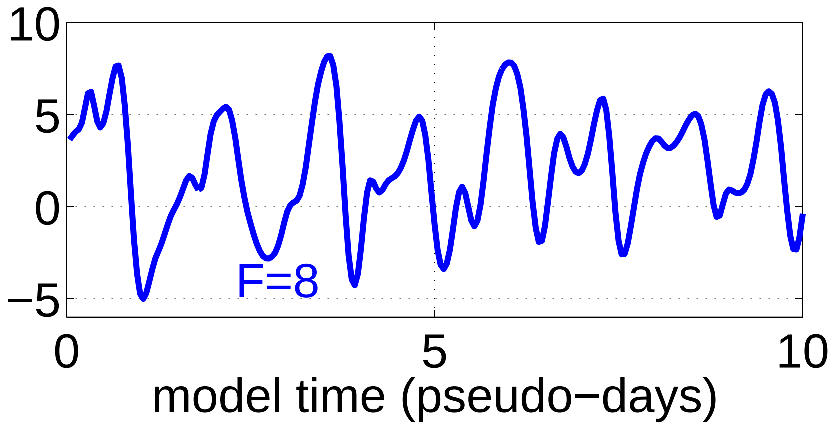
Results from 10 days after 40 day spin-up

Vary assimilating model forcing: F=8, 6, 3, 0

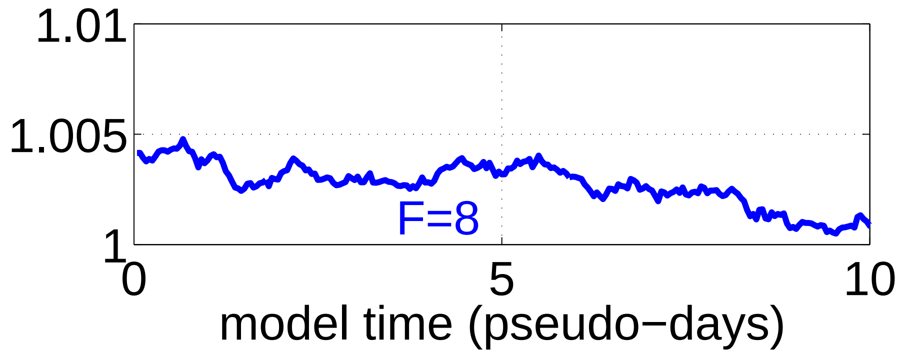
Simulates increasing model error

Assimilating F=8 Truth with F=8 Ensemble

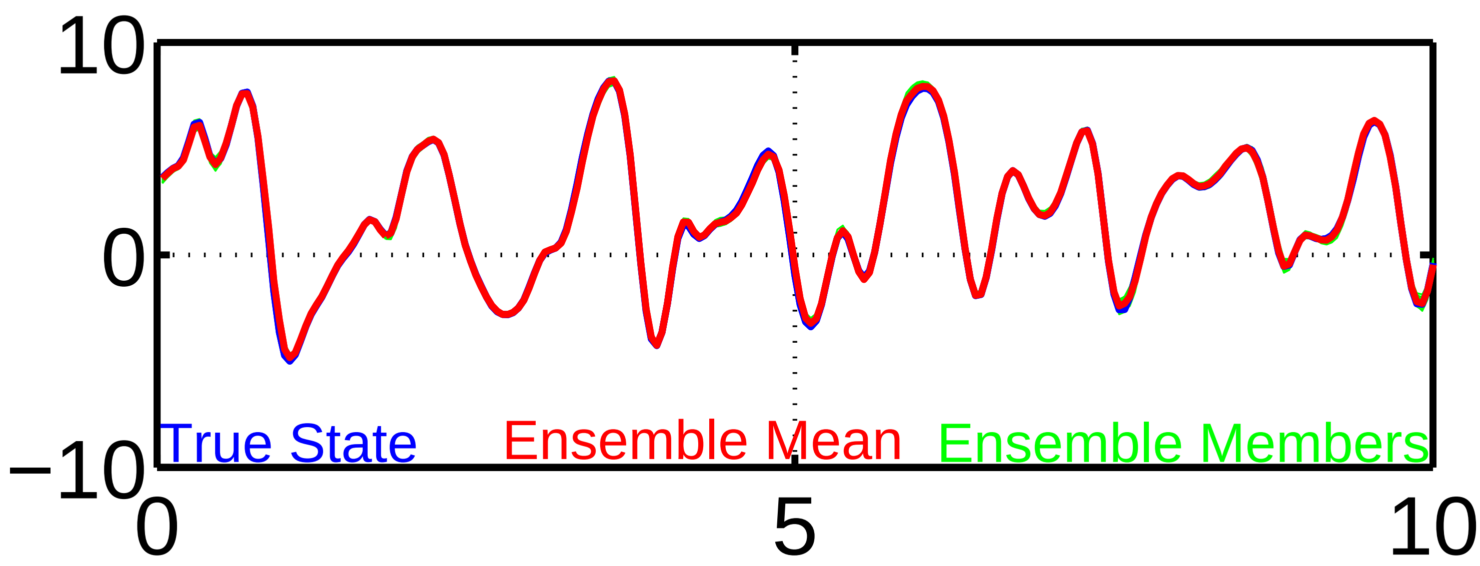
Model time series



Mean value of λ

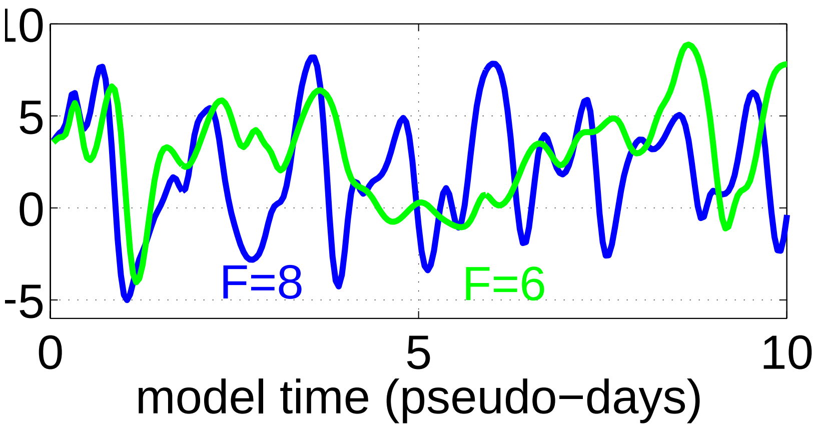


Assimilation Results

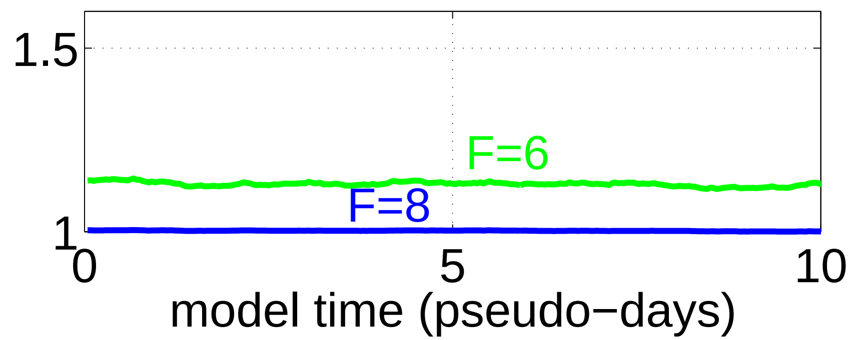


Assimilating F=8 Truth with F=6 Ensemble

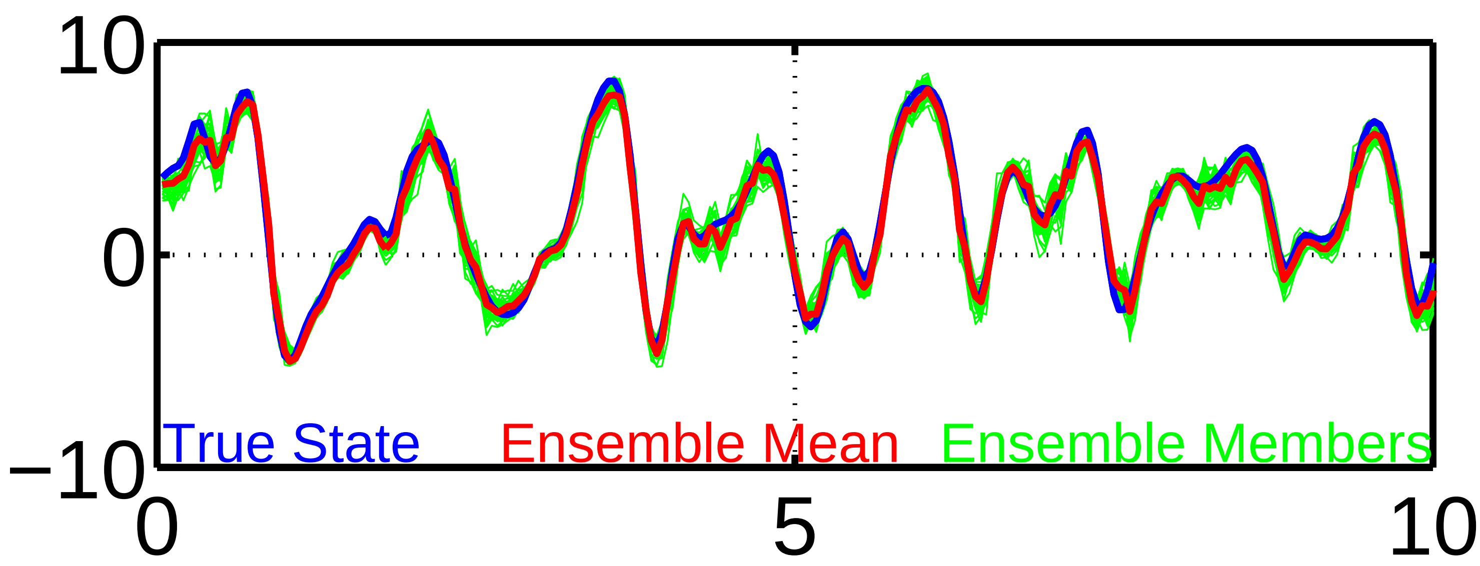
Model time series



Mean value of λ

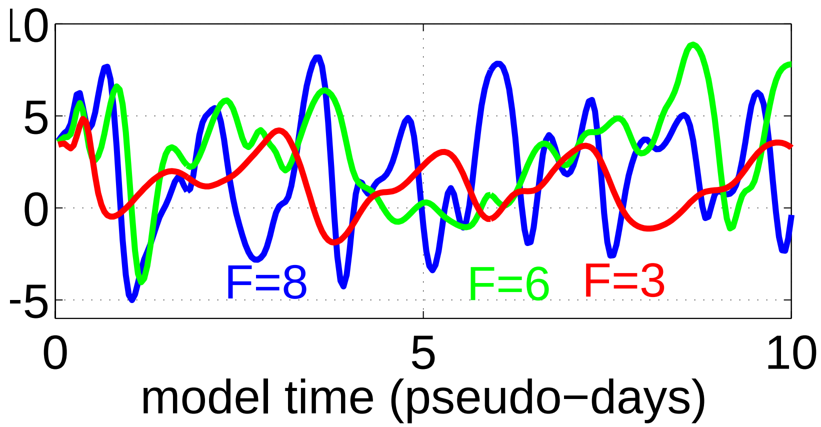


Assimilation Results

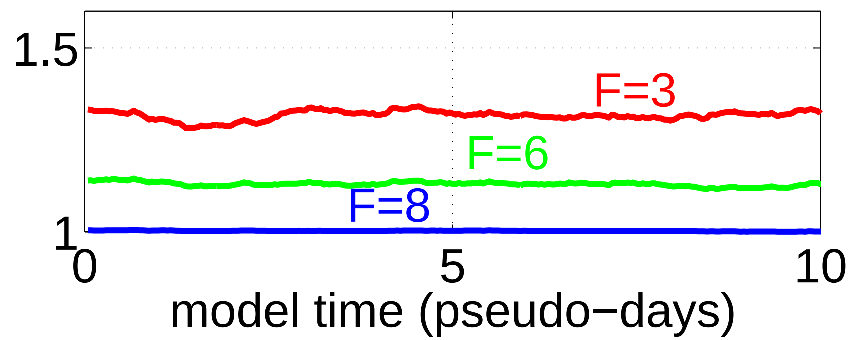


Assimilating F=8 Truth with F=3 Ensemble

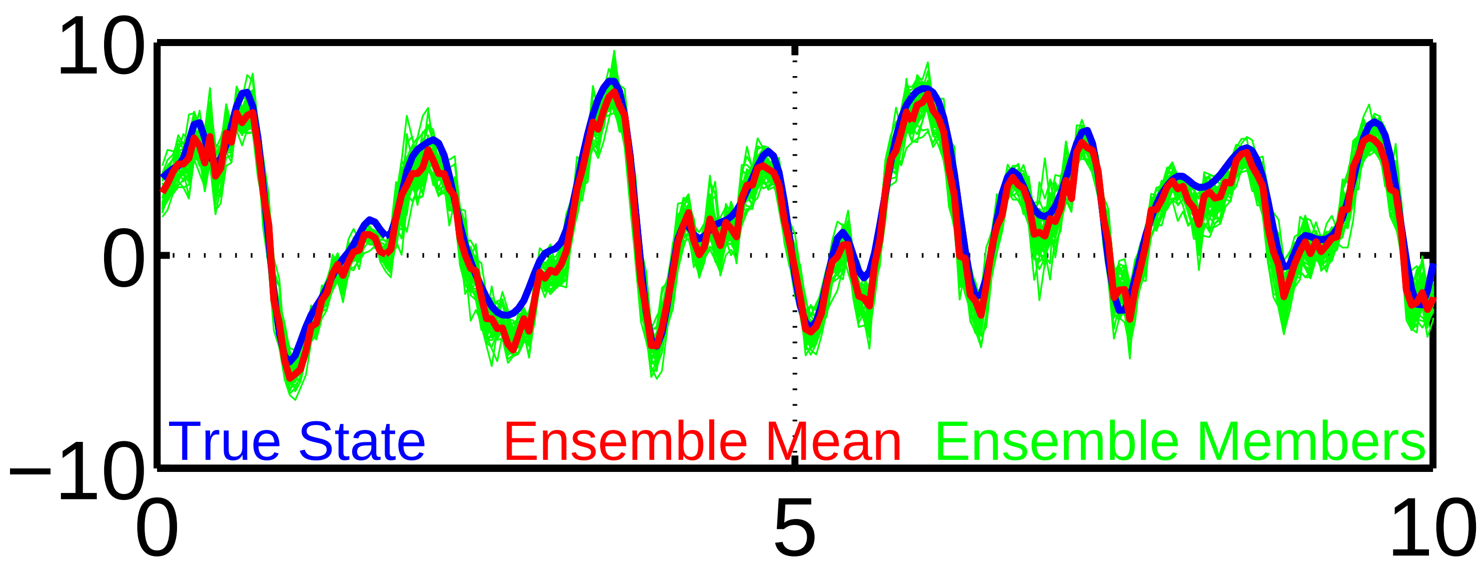
Model time series



Mean value of λ

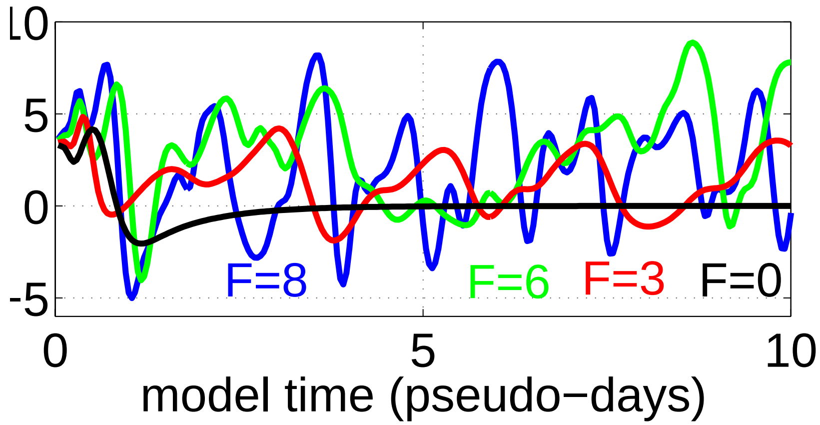


Assimilation Results

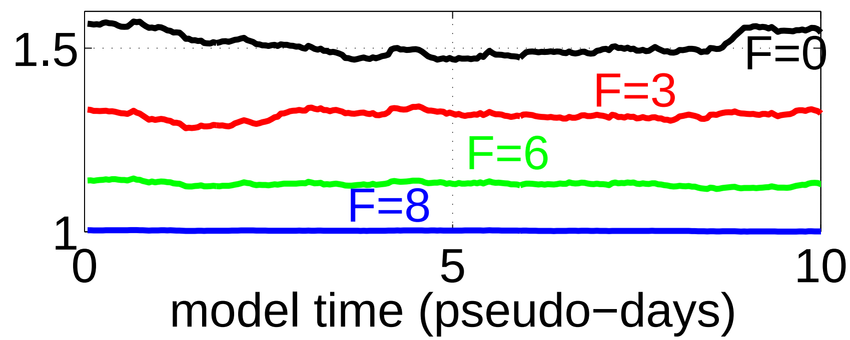


Assimilating F=8 Truth with F=0 Ensemble

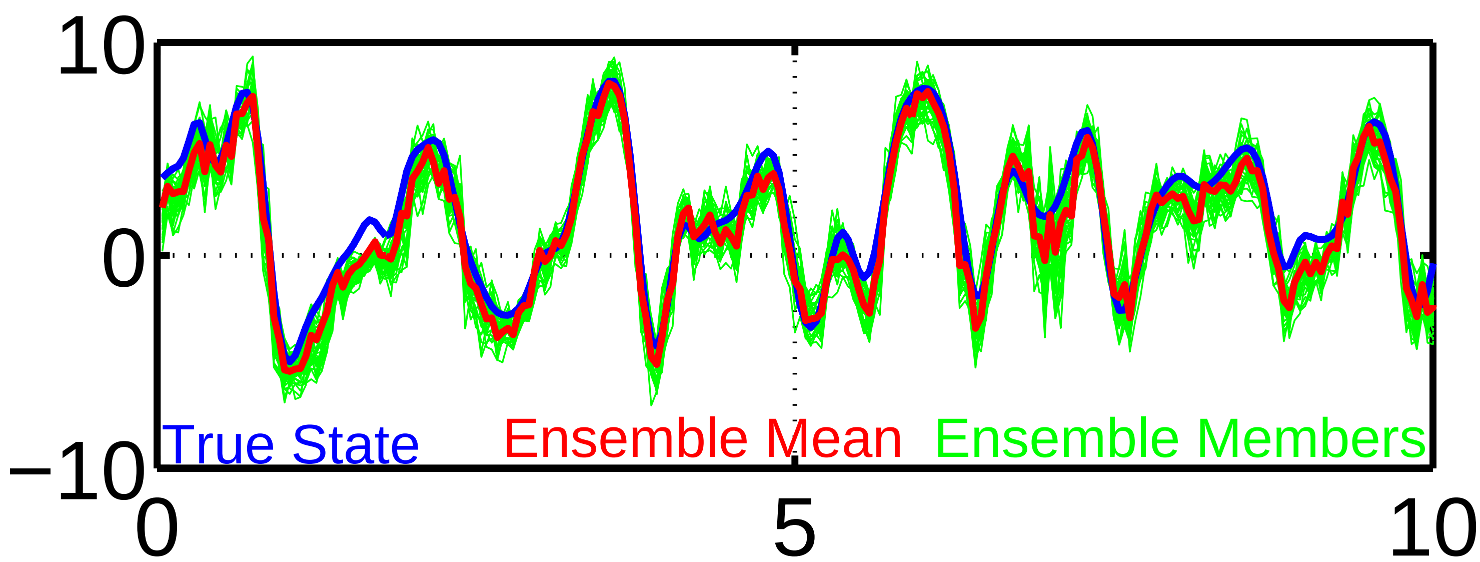
Model time series



Mean value of λ

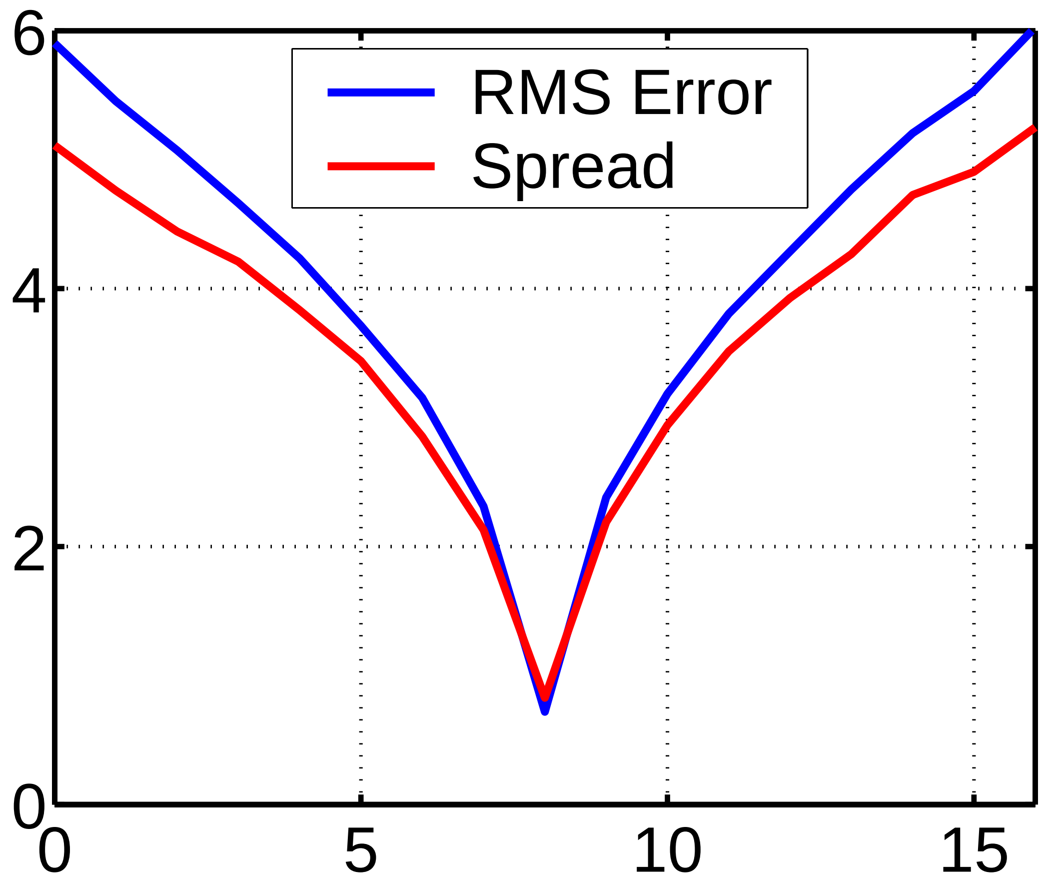


Assimilation Results

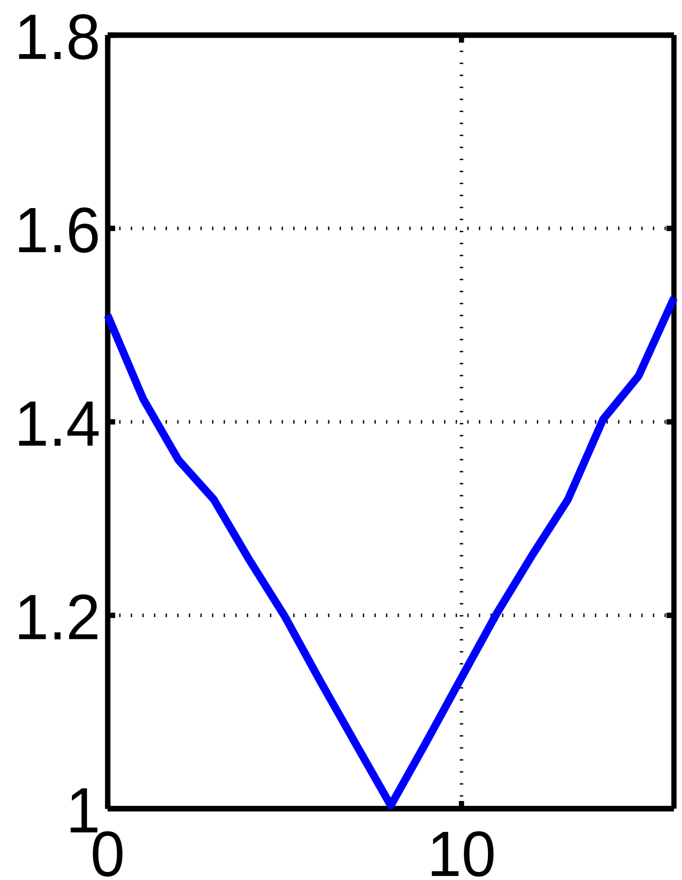


Prior RMS Error, Spread, and λ Grow as Model Error Grows

Base case: 200 randomly located observations per time



Assimilating Model Forcing, F



Assim. Forcing, F

(Error saturation is approximately 30.0)

Prior RMS Error, Spread, and λ Grow as Model Error Grows

Less well observed case, 40 randomly located observations per time