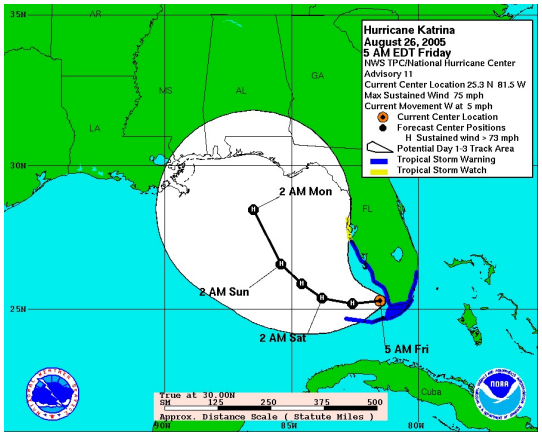
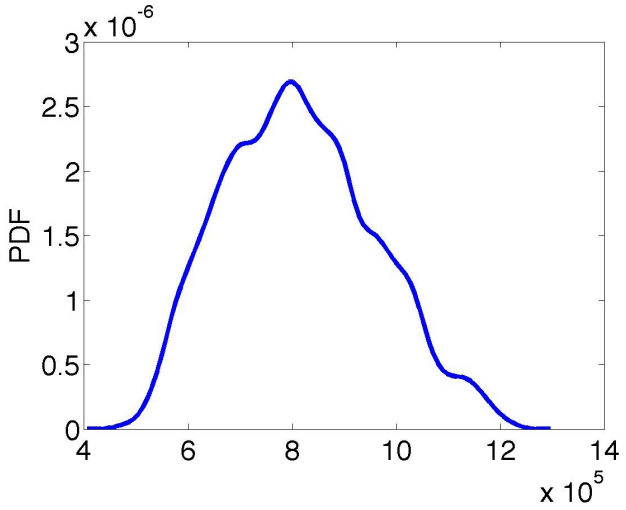
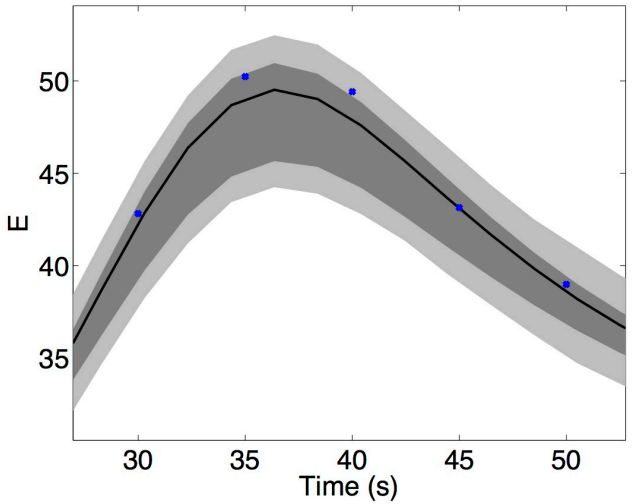
Introduction to Bayesian Inference and Uncertainty Propagation

Ralph C. Smith
Department of Mathematics
North Carolina State University







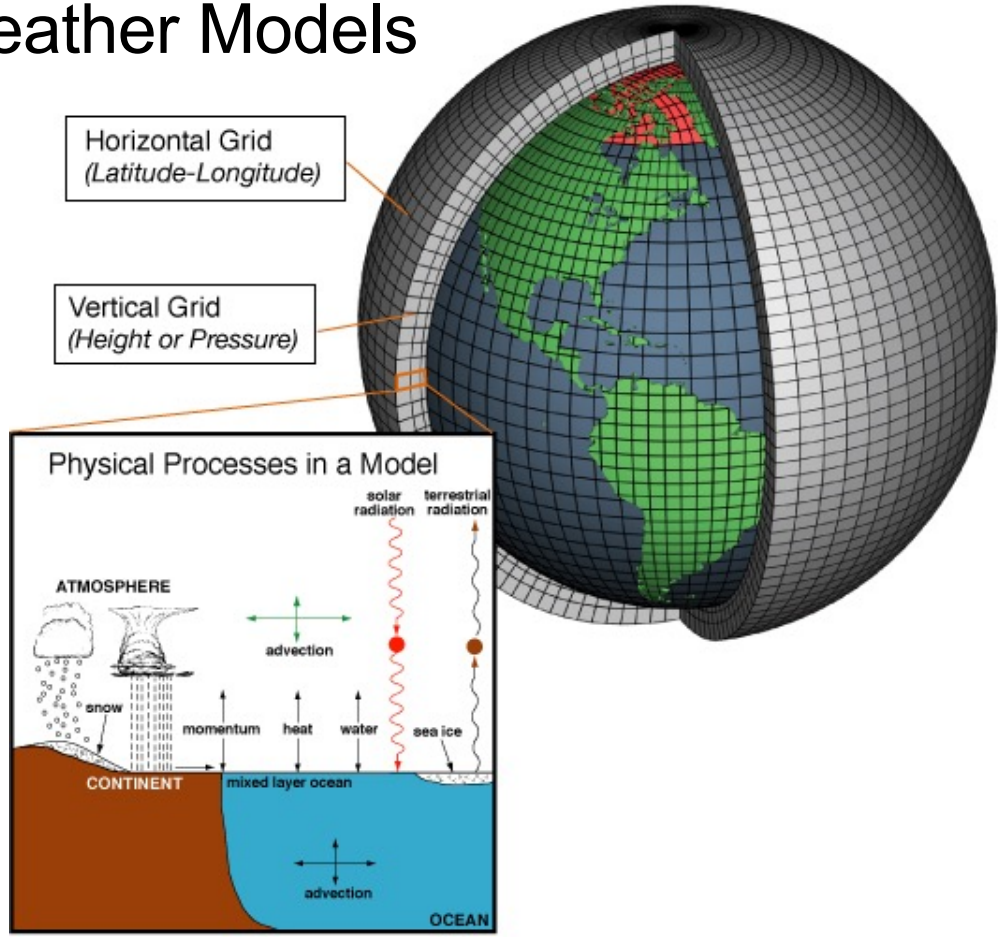
Essentially, all models are wrong, but some are useful, George E.P. Box, Industrial Statistician.

Support: DOE Consortium for Advanced Simulation of LWR (CASL)
NNSA Consortium for Nonproliferation Enabling Capabilities (CNEC)
National Science Foundation (NSF)
Air Force Office of Scientific Research (AFOSR)

Example 1: Weather Models

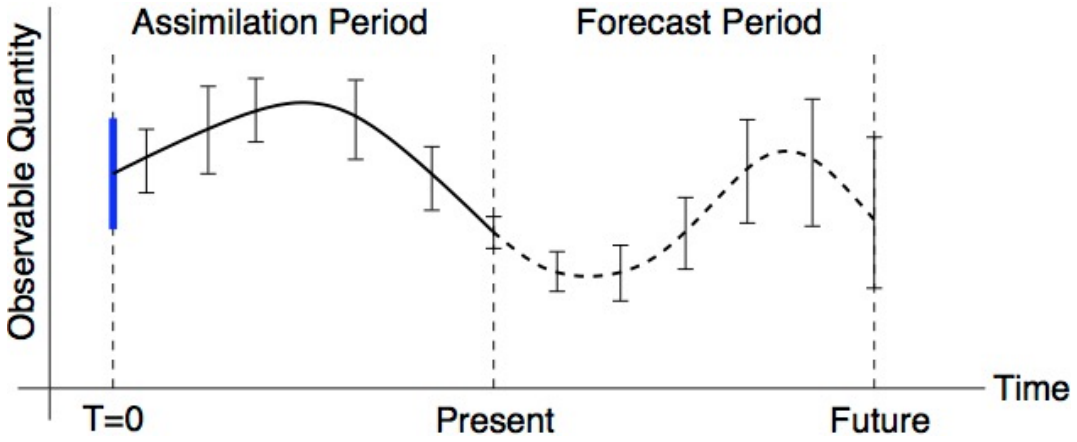
Challenges:

- Coupling between temperature, pressure gradients, precipitation, aerosol, etc.;
- Models and inputs contain uncertainties;
- Numerical grids necessarily larger than many phenomena; e.g., clouds
- Sensors positions may be uncertain; e.g., weather balloons, ocean buoys.



Goal:

- Assimilate data to quantify uncertain initial conditions and parameters;
- Make predictions with quantified uncertainties.



Equations of Atmospheric Physics

Conservation Relations:

Mass
$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) = 0$$

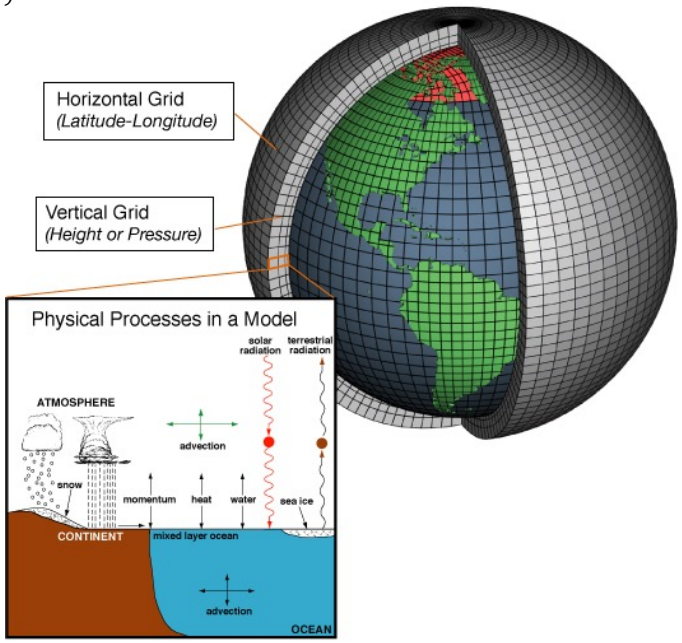
Momentum
$$\frac{\partial v}{\partial t} = -v \cdot \nabla v - \frac{1}{\rho}\nabla p - g\hat{k} - 2\Omega \times v$$

Energy
$$\rho c_V \frac{\partial T}{\partial t} + p\nabla \cdot v = -\nabla \cdot F + \nabla \cdot (k\nabla T) + \rho \dot{q}(T, p, \rho)$$

$$p = \rho RT$$

Water
$$\frac{\partial m_j}{\partial t} = -v \cdot \nabla m_j + S_{m_j}(T, m_j, \chi_j, \rho)\ ,\ j = 1, 2, 3,$$

Aerosol
$$\frac{\partial \chi_j}{\partial t} = -v \cdot \nabla \chi_j + S_{\chi_j}(T, \chi_j, \rho)\ ,\ j = 1, \cdots, J,$$



Constitutive Closure Relations: e.g.,

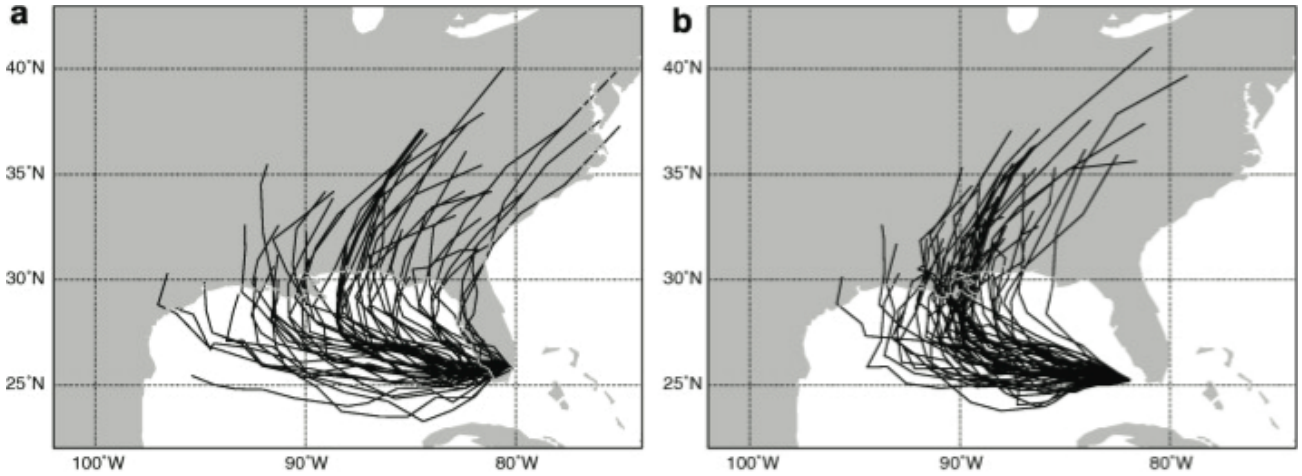
$$S_{m_2} = S_1 + S_2 + S_3 - S_4$$

where

$$S_1 = \bar{\rho}(m_2 - m_2^*)^2 \left[1.2 \times 10^{-4} + \left(1.569 \times 10^{-12} \frac{n_r}{d_0(m_2 - m_2^*)}\right)\right]^{-1}$$

Ensemble Predictions

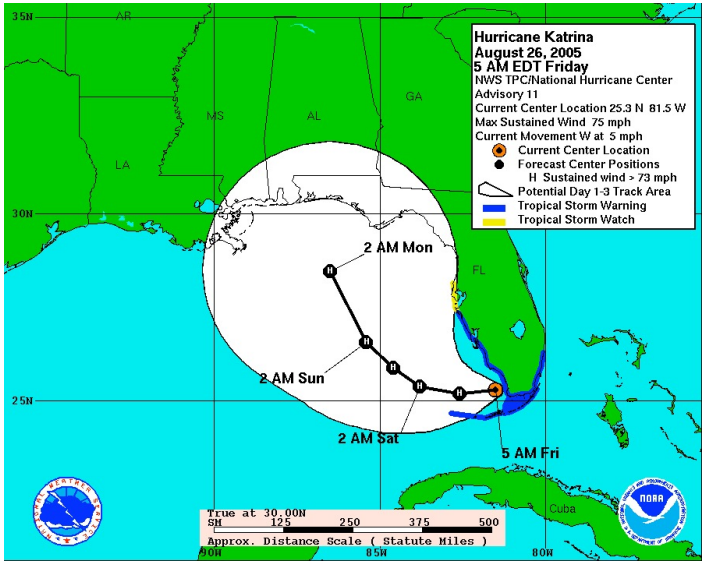
Ensemble Predictions:



00 UTC on August 26, 2005

12 UTC on August 26, 2005

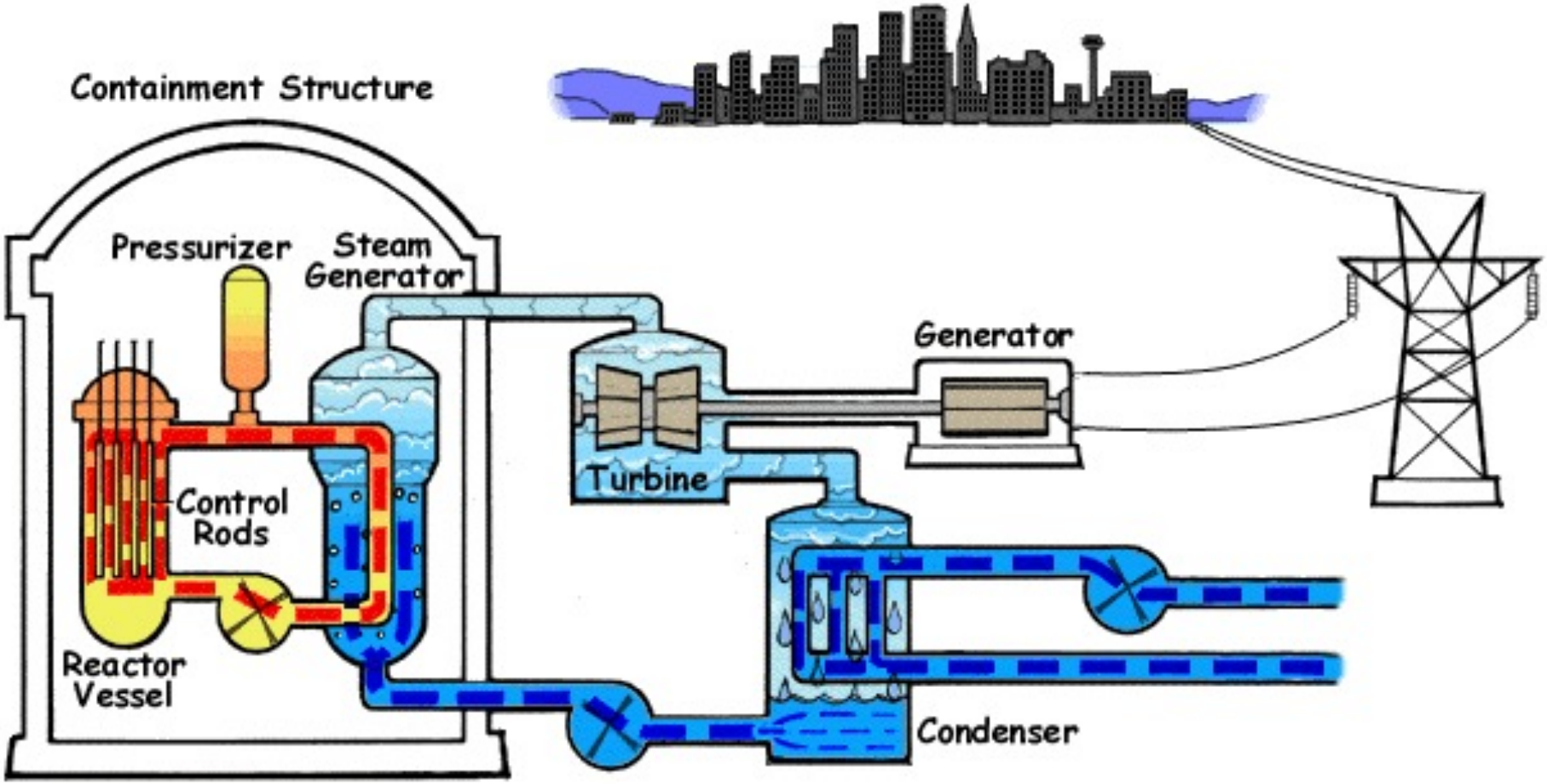
Cone of Uncertainty:



General Questions:

- What is expected rainfall in Research Triangle on July 12?
- What are average high and low temperatures?
- What is predicted average snow fall?
- Note: Quantities are statistical in nature.

Example 2: Pressurized Water Reactors (PWR)



Models:

- Involve neutron transport, thermal-hydraulics, chemistry, fuels
- Inherently multi-scale, multi-physics -- Must be incorporated in surrogate models

Objective: Develop Virtual Environment for Reactor Applications (VERA)

Example: Pressurized Water Reactors (PWR)

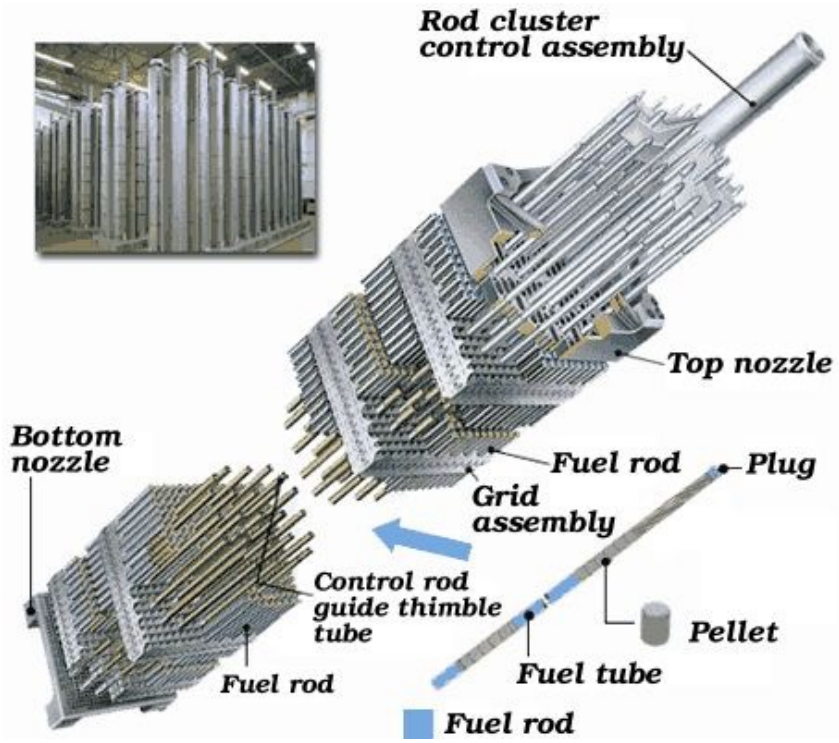
Challenges:

• Models linear in the state but function of 7 independent variables:

$$r = x, y, z; E; \Omega = \theta, \phi; t$$

• Very large number of inputs or parameters; e.g., 100,000. Parameter selection critical.

• Codes can take hours to days to run.



Example: Shearon Harris outside Raleigh

UQ Questions:

• What is peak operating temperature?

• What is expected level of CRUD buildup?

• What is associated risk?

• What is expected profit for new design?

Example 3: HIV Model for Characterization and Control Regimes

HIV Model:

$$\dot{T}_1 = \lambda_1 - d_1 T_1 - (1-\varepsilon) k_1 V T_1$$

$$\dot{T}_2 = \lambda_2 - d_2 T_2 - (1-f\varepsilon) k_2 V T_2$$

$$\dot{T}_1^* = (1-\varepsilon) k_1 V T_1 - \delta T_1^* - m_1 E T_1^*$$

$$\dot{T}_2^* = (1-f\varepsilon) k_2 V T_2 - \delta T_2^* - m_2 E T_2^*$$

$$\dot{V} = N_T \delta (T_1^* + T_2^*) - cV - [(1-\varepsilon)\rho_1 k_1 T_1 + (1-f\varepsilon)\rho_2 k_2 T_2] V$$

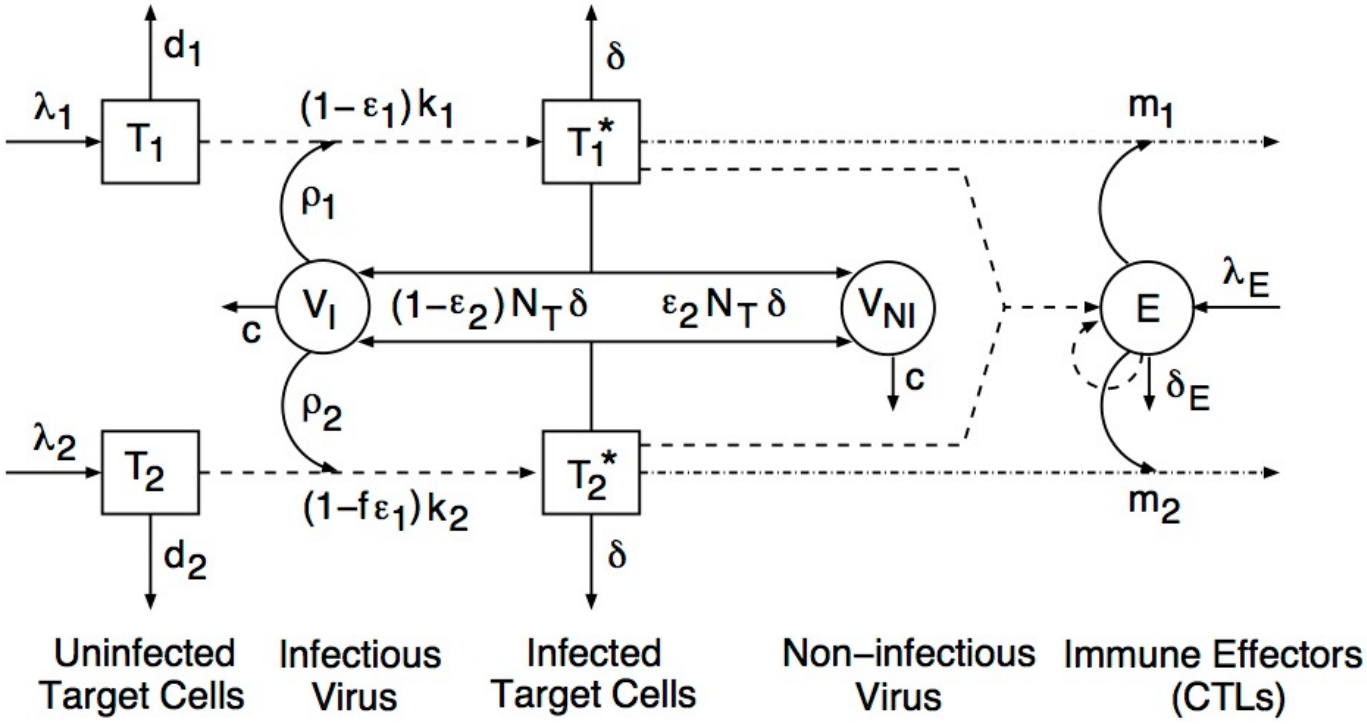
$$\dot{E} = \lambda_E + \frac{b_E (T_1^* + T_2^*)}{T_1^* + T_2^* + K_b} E - \frac{d_E (T_1^* + T_2^*)}{T_1^* + T_2^* + K_d} E - \delta_E E$$

Notes: 21 parameters

[Adams, Banks et al., 2005, 2007]

Notation: $\dot{E} \equiv \frac{dE}{dt}$

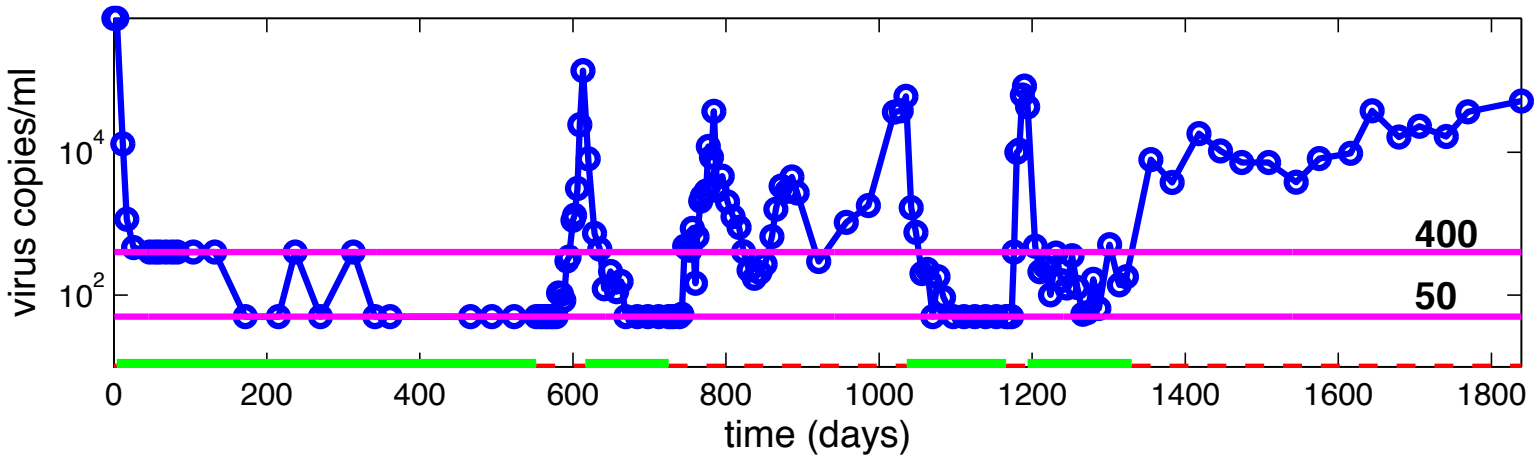
Compartments:



Example: HIV Model for Characterization and Treatment Regimes

HIV Model: Several sources of uncertainty including viral measurement techniques

Example: Upper and lower limits to assay sensitivity



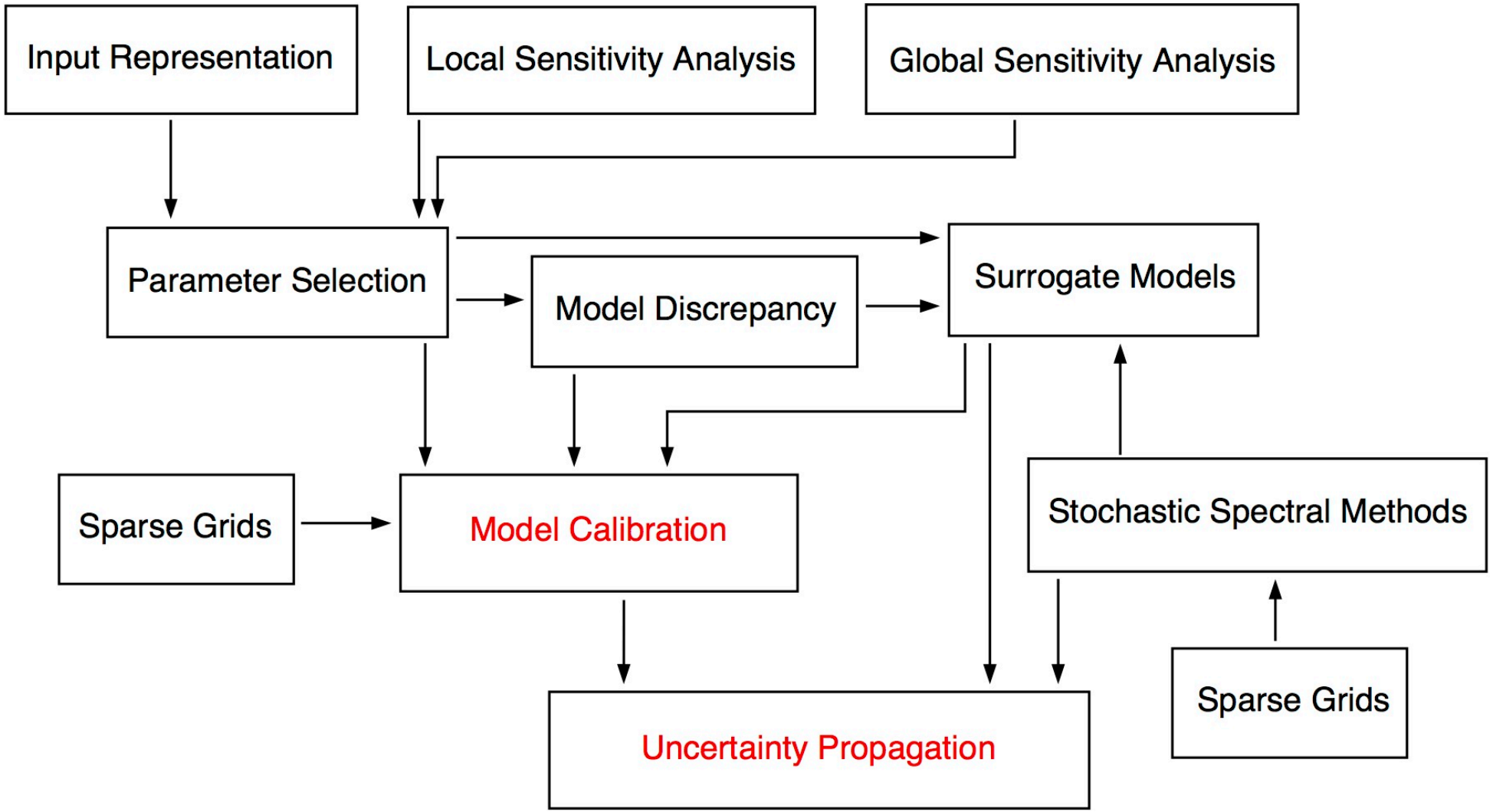
UQ Questions:

- What are the uncertainties in parameters that cannot be directly measured?
- What is optimal treatment regime that is "safe" for patient?
- What is expected viral load? Issue: very often requires high-dimensional integration!
- e.g., $\mathbb{E}[V(t)] = \int_{\mathbb{R}^{21}} V(t,q)\rho(q)dq$

Experimental results are believed by everyone, except for the person who ran the experiment, source anonymous, quoted by Max Gunzburger, Florida State University.

Steps in Uncertainty Quantification

Note: Uncertainty quantification requires synergy between statistics, mathematics and application area.



Model Calibration and Uncertainty Propagation

Sources of Uncertainty:

- Model
- Parameters
- Sensor measurements
- Initial conditions

Strategy:

- Quantify uncertainty in parameters
- Propagate uncertainty through model

Example: HIV model

$$\dot{T}_1 = \lambda_1 - d_1 T_1 - (1-\varepsilon) k_1 V T_1$$

$$\dot{T}_2 = \lambda_2 - d_2 T_2 - (1-f\varepsilon) k_2 V T_2$$

$$\dot{T}_1^* = (1-\varepsilon) k_1 V T_1 - \delta T_1^* - m_1 E T_1^*$$

$$\dot{T}_2^* = (1-f\varepsilon) k_2 V T_2 - \delta T_2^* - m_2 E T_2^*$$

$$\dot{V} = N_T \delta (T_1^* + T_2^*) - cV - [(1-\varepsilon)\rho_1 k_1 T_1 + (1-f\varepsilon)\rho_2 k_2 T_2] V$$

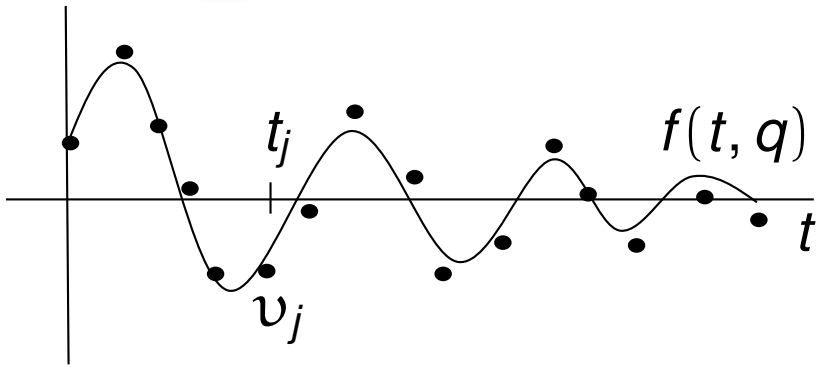
$$\dot{E} = \lambda_E + \frac{b_E (T_1^* + T_2^*)}{T_1^* + T_2^* + K_b} E - \frac{d_E (T_1^* + T_2^*)}{T_1^* + T_2^* + K_d} E - \delta_E E$$

Parameters: Reduced set

$$q = [b_E, \delta, d_1, k_2, \lambda_1, K_b]$$

Point Estimates: Ordinary least squares

$$q^0 = \arg\min_q \frac{1}{2} \sum_{j=1}^{N} [\upsilon_j - f(t_j, q)]^2$$



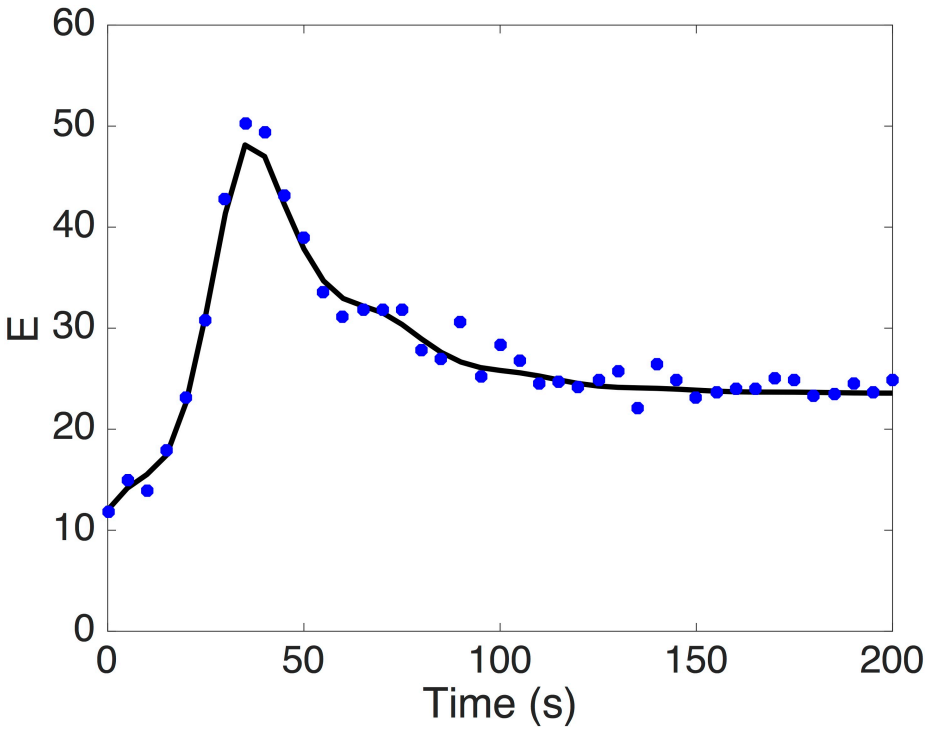
Note: Scaling critical since parameter values vary by 8 orders of magnitude.

Model Calibration and Predictions

Optimization Results:

b_E	δ	d_1	k_2	λ_1	K_b
0.30	0.68	9.1×10^{-3}	1.22×10^{-4}	9.95×10^{3}	88.5

Data and Prediction of Immune Effector Response E:



Note: Point estimates but no quantification of uncertainty in:

- Model
- Parameters
- Data

Goals:

- Replace point estimates with distributions.
- Construct credible and prediction intervals.
- Natural in a Bayesian framework

Statistical Inference

Goal: The goal in statistical inference is to make conclusions about a phenomenon based on observed data.

Frequentist: Observations made in the past are analyzed with a specified model. Result is regarded as confidence about state of real world.

- Probabilities defined as frequencies with which an event occurs if experiment is repeated several times.
- Parameter Estimation:
  - Relies on estimators derived from different data sets and a specific sampling distribution.
  - Parameters may be unknown but are fixed and deterministic.

Bayesian: Interpretation of probability is subjective and can be updated with new data.

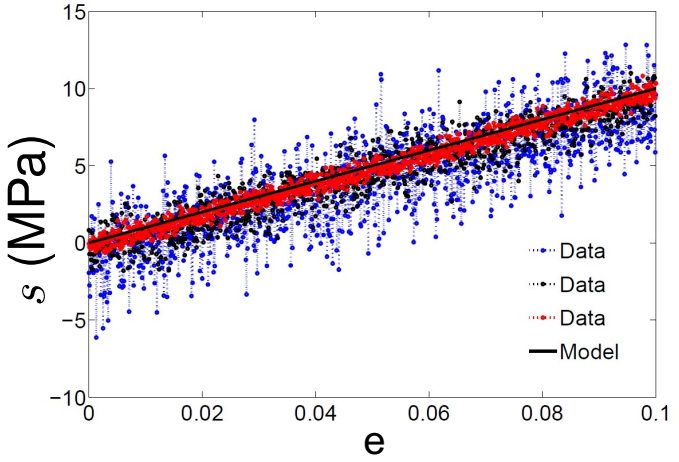
- Parameter Estimation: Parameters are considered to be random variables having associated densities.

Bayesian Inference: Motivation

Example: Displacement-force relation (Hooke's Law)

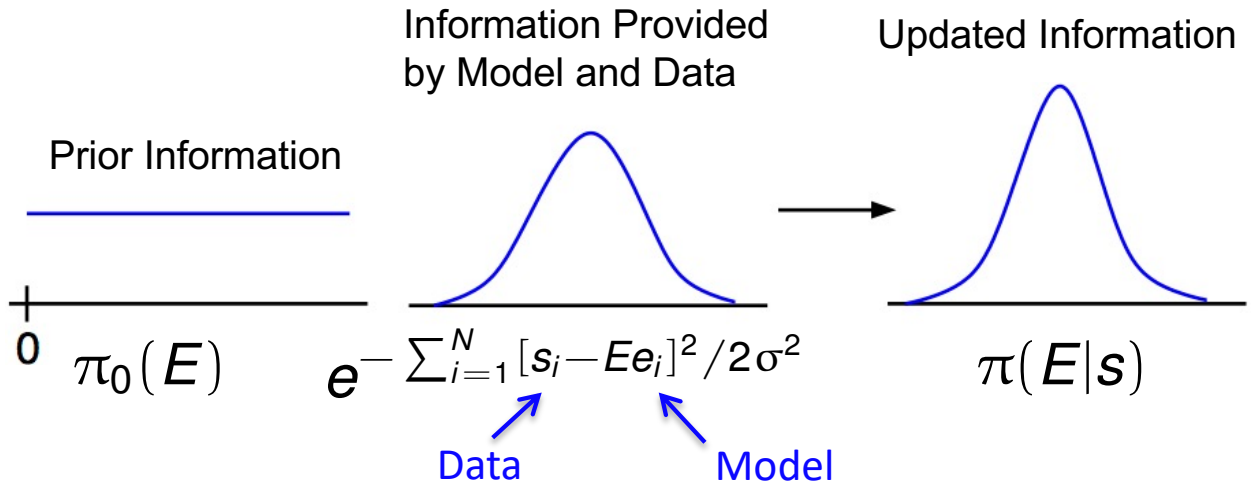
$$s_i = Ee_i + \varepsilon_i \ , \ i = 1, \ldots, N$$

$$\varepsilon_i \sim N(0, \sigma^2)$$



Parameter: Stiffness E

Strategy: Use model fit to data to update prior information

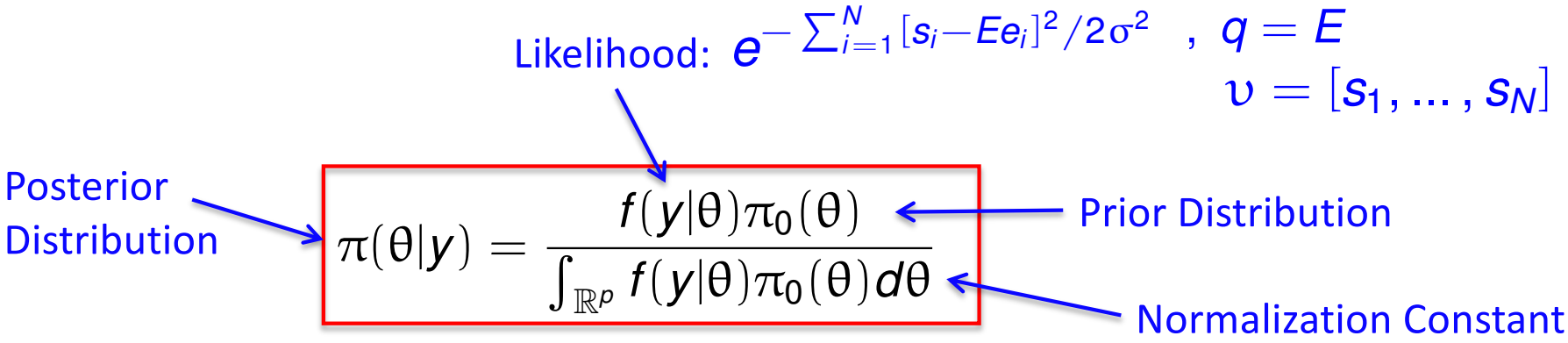


Non-normalized Bayes' Relation:

$$\pi(E|s) = e^{-\sum_{i=1}^{N}[s_i - Ee_i]^2/2\sigma^2}\pi_0(E)$$

Bayesian Inference

Bayes' Relation: Specifies posterior in terms of likelihood and prior



- Prior Distribution: Quantifies prior knowledge of parameter values
- Likelihood: Probability of observing a data given set of parameter values.
- Posterior Distribution: Conditional distribution of parameters given observed data.

Problem: Can require high-dimensional integration

- e.g., Many applications: p = 10-50!
- Solution: Sampling-based Markov Chain Monte Carlo (MCMC) algorithms.
- Metropolis algorithms first used by nuclear physicists during Manhattan Project in 1940's to understand particle movement underlying first atomic bomb.

Parameter Estimation Problem

Observation Model:

$$y_i = f_i(\theta) + \varepsilon_i \ , \ i = 1, \ldots, n$$

Assumption: Assume that measurement errors are iid and $\varepsilon_i \sim N(0, \sigma^2)$

Likelihood:

$$f(y|\theta) = L(\theta, \sigma|y) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-SS_\theta/2\sigma^2}$$

where

$$SS_\theta = \sum_{j=1}^{n} [y_j - f_i(\theta)]^2$$

is the sum of squares error.

Parameter Estimation: Example

Example: Consider the spring model

$$\ddot{z} + C\dot{z} + Kz = 0$$

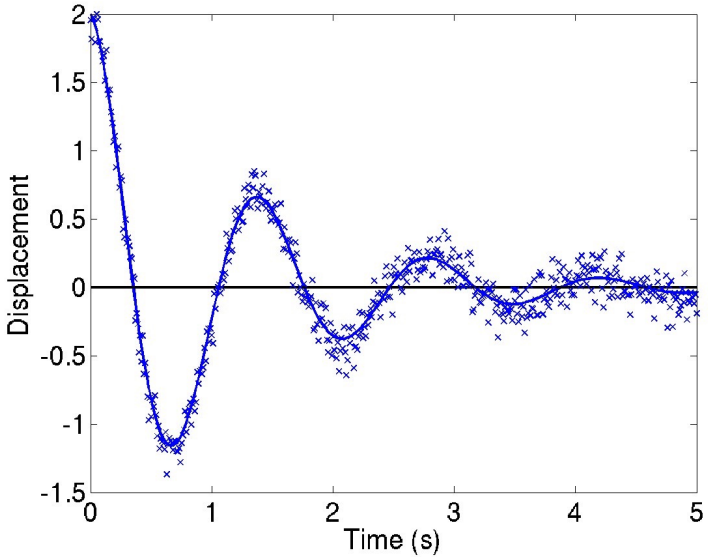
$$z(0) = 2\,,\ \dot{z}(0) = -C$$

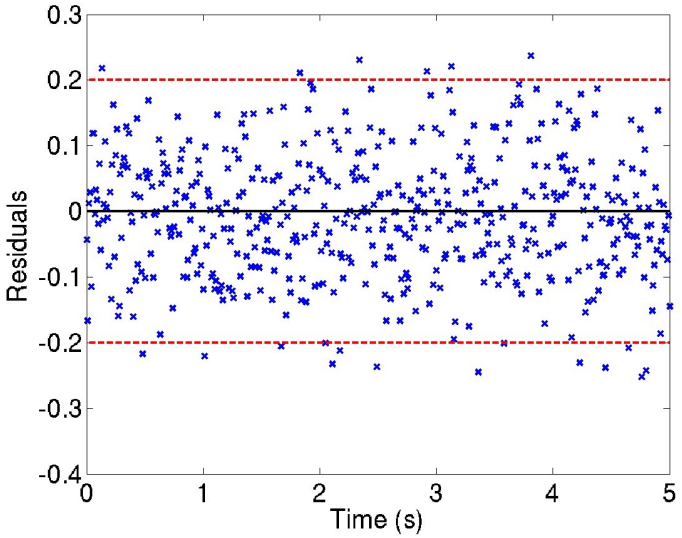
which has the solution

$$z(t) = 2e^{-Ct/2}\cos(\sqrt{K - C^2/4}\cdot t)$$

Note: Take $K = 20.5$, $C^0 = 1.5$

Take K to be known and $\theta = C$. Assume that $\varepsilon_i \sim N(0, \sigma_0^2)$ where $\sigma_0 = 0.1$





Parameter Estimation: Example

Example: The sensitivity matrix is

$$\mathcal{X}(\theta) = \left[\frac{\partial y}{\partial C}(t_1, \theta), \cdots, \frac{\partial y}{\partial C}(t_n, \theta)\right]^T$$

where

$$\frac{\partial y}{\partial C} = e^{-Ct/2}\left[\frac{Ct}{\sqrt{4K - C^2}}\sin\left(\sqrt{K - C^2/4}\cdot t\right) - t\cos\left(\sqrt{K - C^2/4}\cdot t\right)\right]$$

Here

$$V = \sigma_c^2 = \sigma_0^2\left[\mathcal{X}^T(\theta)\mathcal{X}(\theta)\right]^{-1} = 3.35 \times 10^{-4}$$

so that

$$\widehat{C} \sim N\left(C_0, \sigma_c^2\right)\ ,\ \sigma_c = 0.0183$$

Note: In 10,000 simulations, 9455 of confidence intervals contained true parameter value.

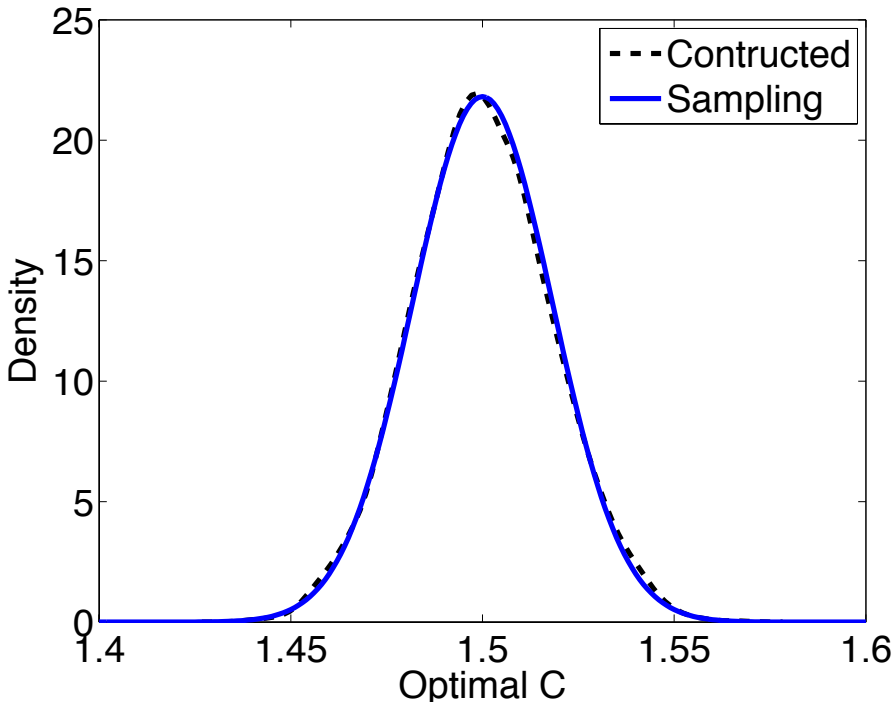


Figure: Sampling distribution compared with that constructed using 10,000 estimated values of C.

Parameter Estimation: Example

Bayesian Inference: Employ the flat prior

$$\pi_0(\theta) = \chi_{[0,\infty)}(\theta)$$

Posterior Distribution:

$$\pi(\theta|y) = \frac{e^{-SS_\theta/2\sigma_0^2}}{\int_0^\infty e^{-SS_\zeta/2\sigma_0^2} d\zeta} = \frac{1}{\int_0^\infty e^{-(SS_\zeta - SS_\theta)/2\sigma_0^2} d\zeta}$$

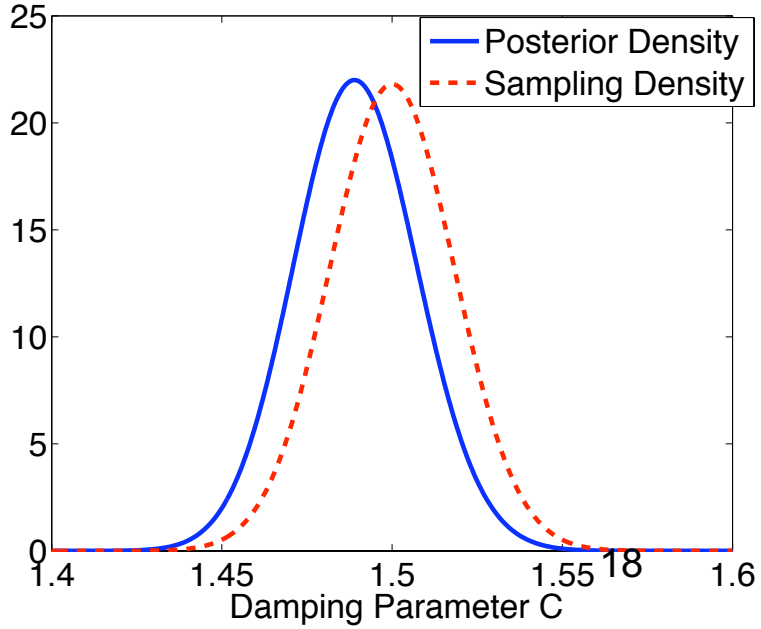
Issue: $e^{-SS_{\theta\,MAP}} \approx 3 \times 10^{-113}$

Midpoint formula:

$$\pi(\theta|y) \approx \frac{1}{\sum_{i=1}^{k} e^{-(SS_{\zeta_i} - SS_\theta)/2\sigma_0^2} w_i}$$

Note:

- Slow even for one parameter.
- Strategy: create Markov chain using random sampling so that created chain has the posterior distribution as its limiting (stationary) distribution.



Markov Chain Monte Carlo Methods

General Strategy:

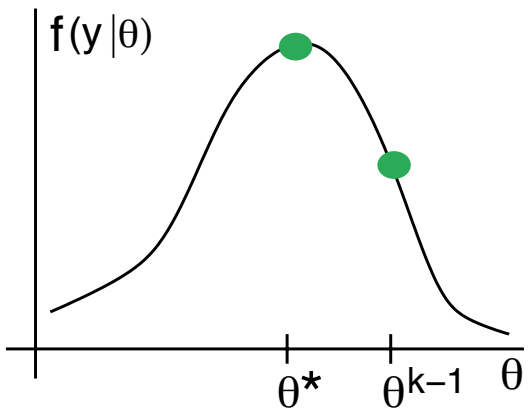
- Current value: $X_{k-1} = \theta^{k-1}$
- Propose candidate $\theta^* \sim J(\theta^*|\theta^{k-1})$ from proposal (jumping) distribution
- With probability $\alpha(\theta^*, \theta^{k-1})$, accept θ^*; i.e., $X_k = \theta^*$
- Otherwise, stay where you are: $X_k = \theta^{k-1}$

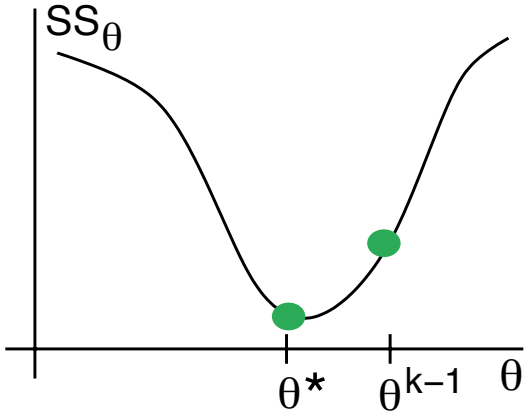
Intuition: Recall that

$$\pi(\theta|y) = \frac{f(y|\theta)\pi_0(\theta)}{\int_{\mathbb{R}^p} f(y|\theta)\pi_0(\theta)d\theta}$$

where

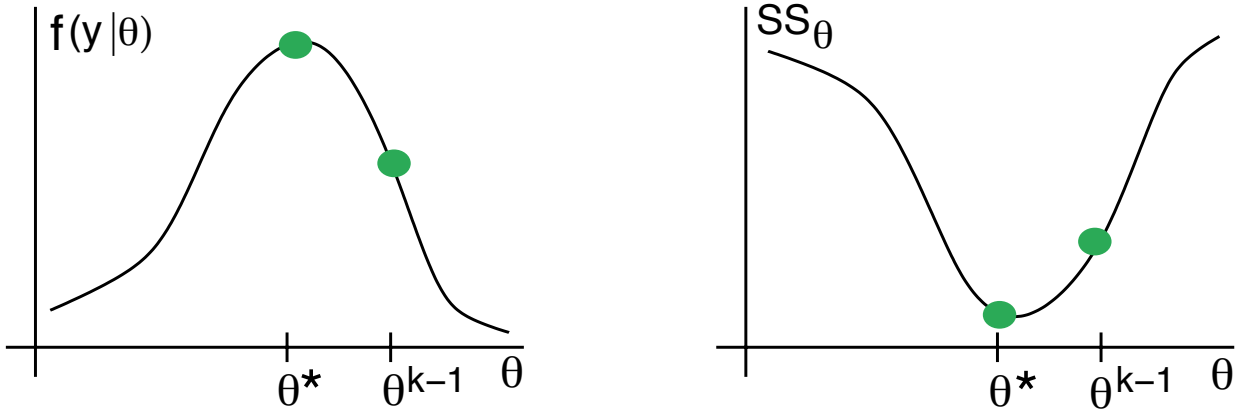
$$f(y|\theta) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\sum_{i=1}^{n}[y_i - f_i(\theta)]^2/2\sigma^2} = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-SS_\theta/2\sigma^2}$$





Markov Chain Monte Carlo Methods

Intuition:

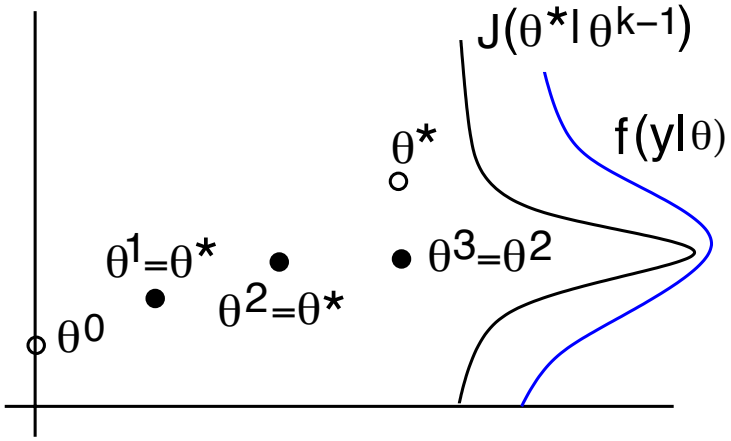


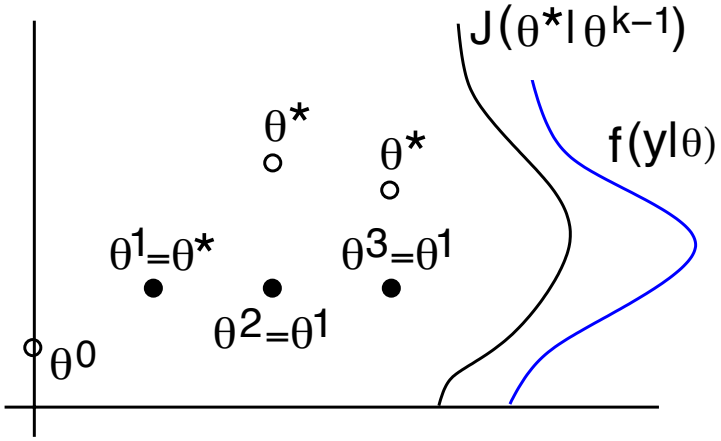
- Consider $r(\theta^*|\theta^{k-1}) = \frac{\pi(\theta^*|y)}{\pi(\theta^{k-1}|y)} = \frac{f(y|\theta^*)\pi_0(\theta^*)}{f(y|\theta^{k-1})\pi_0(\theta^{k-1})}$
  - If $r < 1 \Rightarrow \ f(y|\theta^*) < f(y|\theta^{k-1})$, accept with probability $\alpha = r$
  - If $r > 1$, accept with probability $\alpha = 1$

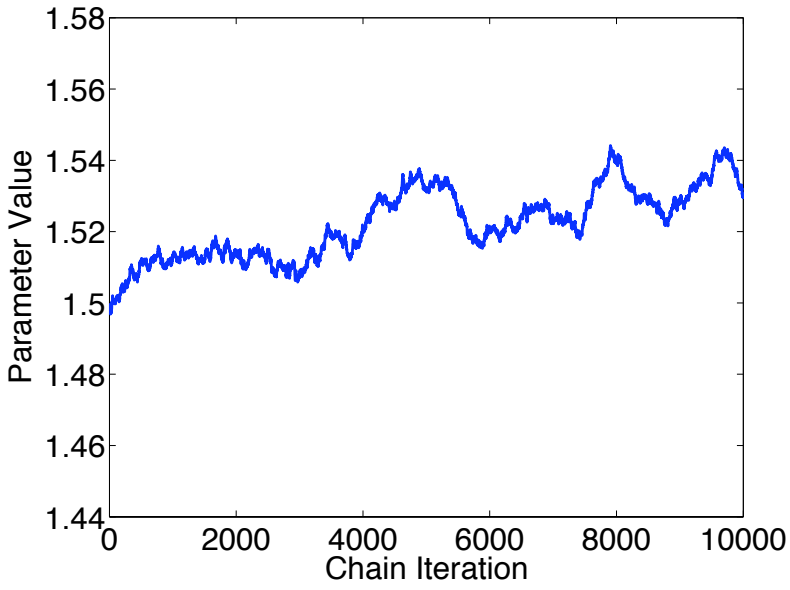
Note: Narrower proposal distribution yields higher probability of acceptance.

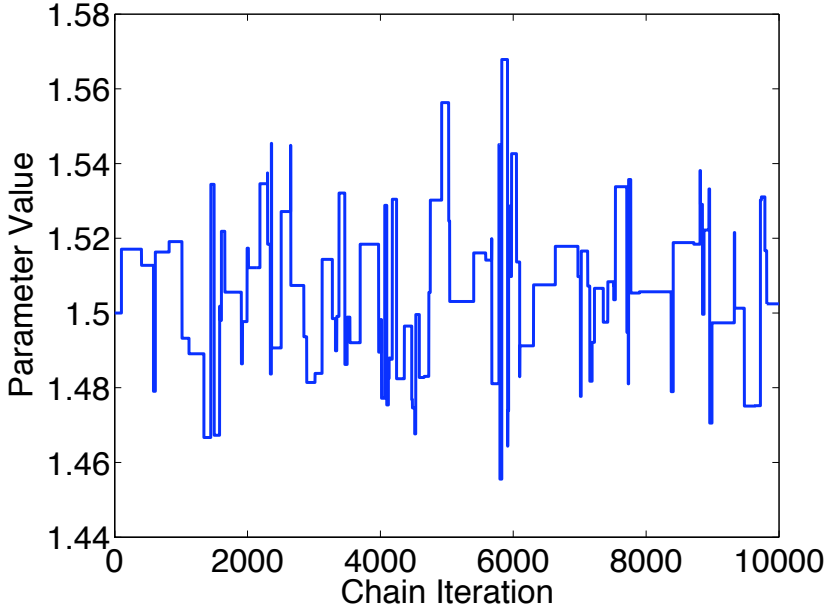
Markov Chain Monte Carlo Methods

Note: Narrower proposal distribution yields higher probability of acceptance.









Proposal Distribution

Proposal Distribution: Two basic approaches

- Choose a fixed proposal function
  - o Independent Metropolis
- Random walk (local Metropolis)

$$\theta^* = \theta^{k-1} + Rz$$

o Two (of several) choices: $Z \sim N(0, 1)$

(i) $R = cI \Rightarrow \theta^* \sim N(\theta^{k-1}, cI)$

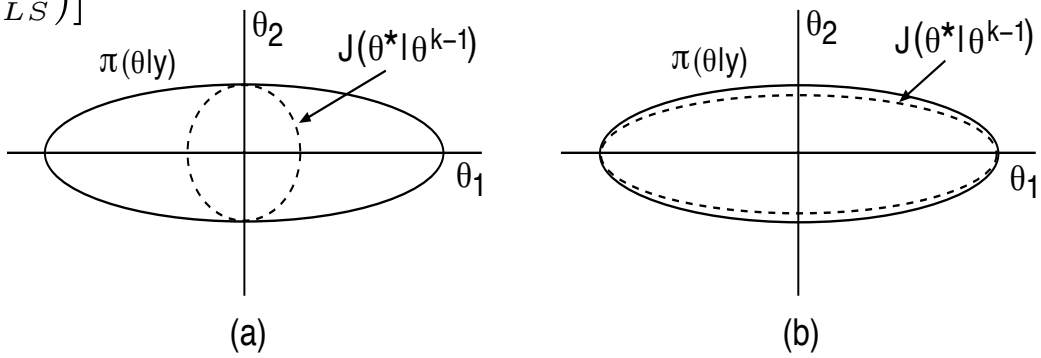
(ii) $R = \text{chol}(V) \Rightarrow \theta^* \sim N(\theta^{k-1}, V)$

where

Sensitivity Matrix

$$V = \sigma^2_{OLS} \left[\mathcal{X}^T(\theta_{OLS})\mathcal{X}(\theta_{OLS})\right]^{-1}$$

$$\sigma^2_{OLS} = \frac{1}{n-p}\sum_{i=1}^{n}\left[y_i - f_i(\theta_{OLS})\right]^2$$



Metropolis Algorithm

Metropolis Algorithm: [Metropolis and Ulam, 1949]

1. Initialization: Choose an initial parameter value θ^0 that satisfies $\pi(\theta^0|y) > 0$.
2. For $k = 1, \cdots, M$

   (a) For $z \sim N(0, 1)$, construct the candidate

   $$\theta^* = \theta^{k-1} + Rz$$

   where R is the Cholesky decomposition of V or D. This ensures that

   $$\theta^* \sim N(\theta^{k-1}, V) \text{ or } \theta^* \sim N(\theta^{k-1}, D).$$

   (b) Compute the ratio

   $$r(\theta^*|\theta^{k-1}) = \frac{\pi(\theta^*|y)}{\pi(\theta^{k-1}|y)} = \frac{\pi(y|\theta^*)\pi_0(\theta^*)}{\pi(y|\theta^{k-1})\pi_0(\theta^{k-1})}.$$

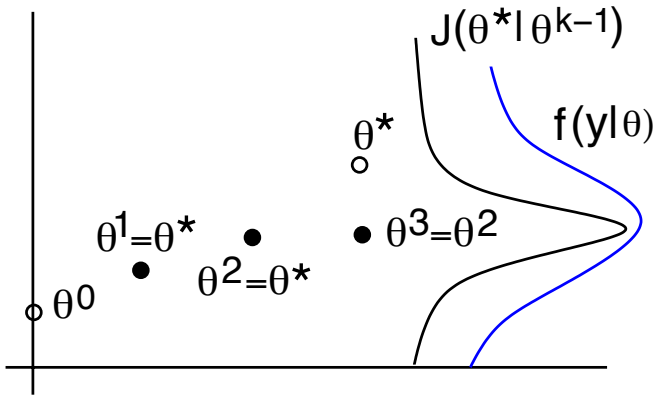
   (c) Set

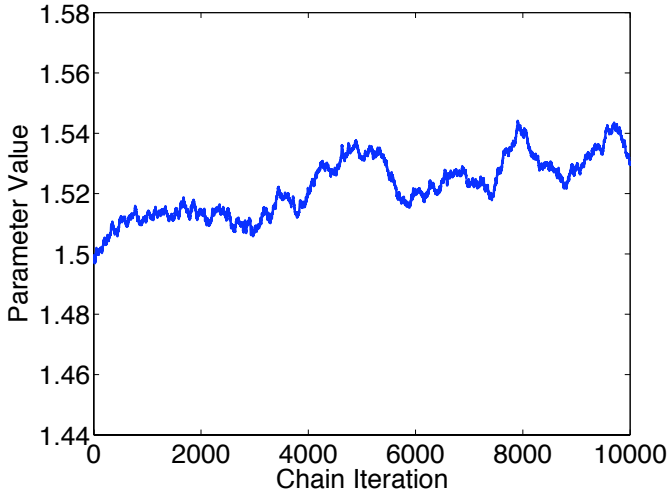
   $$\theta^k = \begin{cases} \theta^* & , \text{ with probability } \alpha = \min(1, r) \\ \theta^{k-1} & , \text{ else.} \end{cases}$$

   That is, we accept θ^* with probability 1 if $r \geqslant 1$ and we accept it with probability r if $r < 1$.

Random Walk Metropolis Algorithm for Parameter Estimation

1. Set number of chain elements M and design parameters n_s, σ_s
2. Determine $\theta^0 = \arg\min_\theta \sum_{i=1}^{N} [y_i - f_i(\theta)]^2$
3. Set $SS_{\theta^0} = \sum_{i=1}^{N} [y_i - f_i(\theta^0)]^2$
4. Compute initial variance estimate: $s_0^2 = \frac{SS_{\theta^0}}{n-p}$
5. Construct covariance estimate $V = s_0^2 [\mathcal{X}^T(\theta^0)\mathcal{X}(\theta^0)]^{-1}$ and $R = \text{chol}(V)$
6. For $k = 1, \cdots, M$
   - (a) Sample $z_k \sim N(0, 1)$
   - (b) Construct candidate $\theta^* = \theta^{k-1} + Rz_k$
   - (c) Sample $u_\alpha \sim \mathcal{U}(0, 1)$
   - (d) Compute $SS_{\theta^*} = \sum_{i=1}^{N} [y_i - f_i(\theta^*)]^2$
   - (e) Compute
     $$\alpha(\theta^* | \theta^{k-1}) = \min\left(1, e^{-[SS_{\theta^*} - SS_{\theta^{k-1}}]/2s_{k-1}^2}\right)$$
   - (f) If $u_\alpha < \alpha$,
     Set $\theta^k = \theta^*$, $SS_{\theta^k} = SS_{\theta^*}$
     else
     Set $\theta^k = \theta^{k-1}$, $SS_{\theta^k} = SS_{\theta^{k-1}}$
     endif
   - (g) Update $s_k \sim \text{Inv-gamma}(a_{val}, b_{val})$ where
     $$a_{val} = 0.5(n_s + n) \ , \ b_{val} = 0.5(n_s \sigma_s^2 + SS_{\theta^k})$$





Random Walk Metropolis

Example: We revisit the spring model

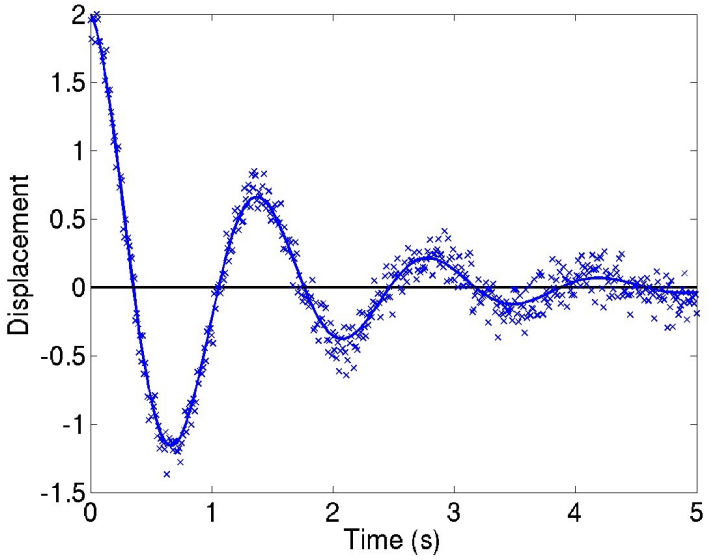
$$\ddot{z} + C\dot{z} + Kz = 0$$

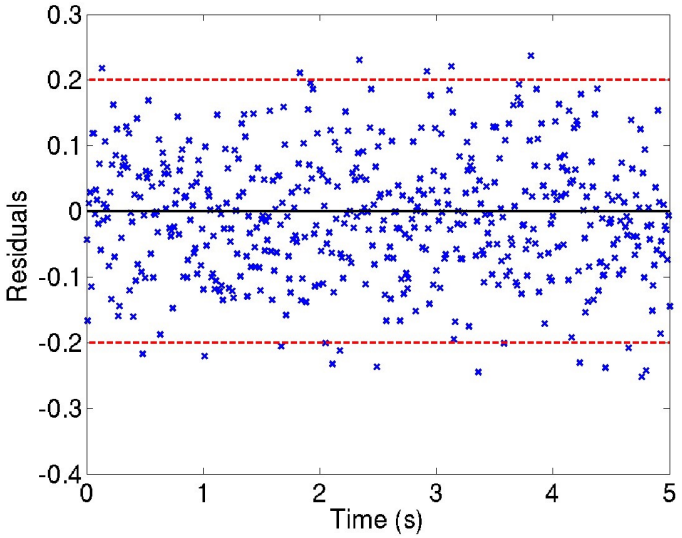
$$z(0) = 2\,,\ \dot{z}(0) = -C$$

which has the solution

$$z(t) = 2e^{-Ct/2}\cos(\sqrt{K - C^2/4}\cdot t)$$

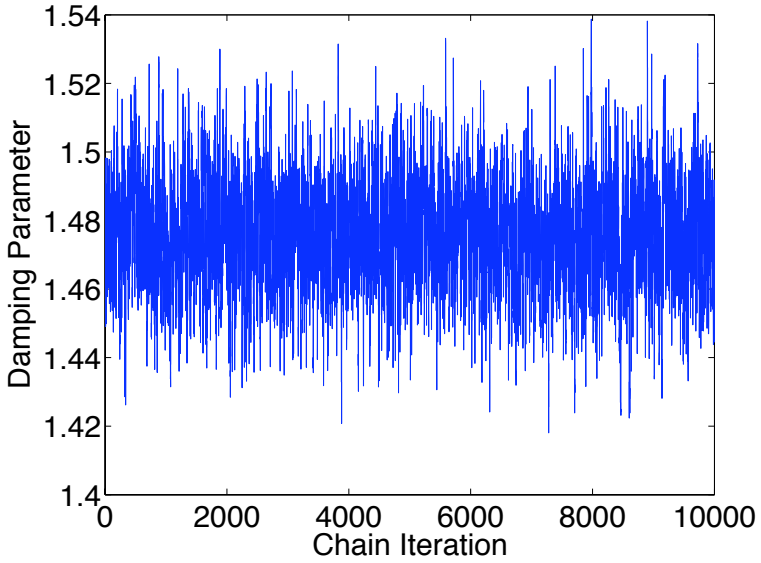
We assume that $\varepsilon_i \sim N(0, \sigma_0^2)$ where $\sigma_0 = 0.1$.

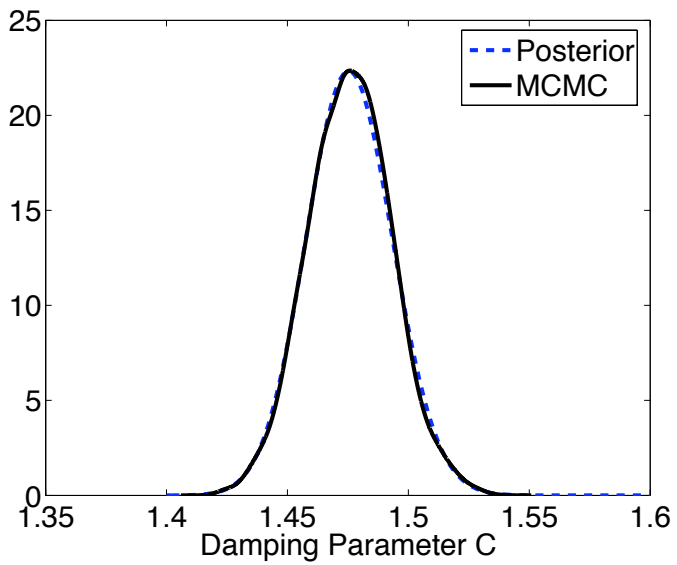




Random Walk Metropolis

Case i: Take K =20.5 and $\theta = [C, \sigma^2]$





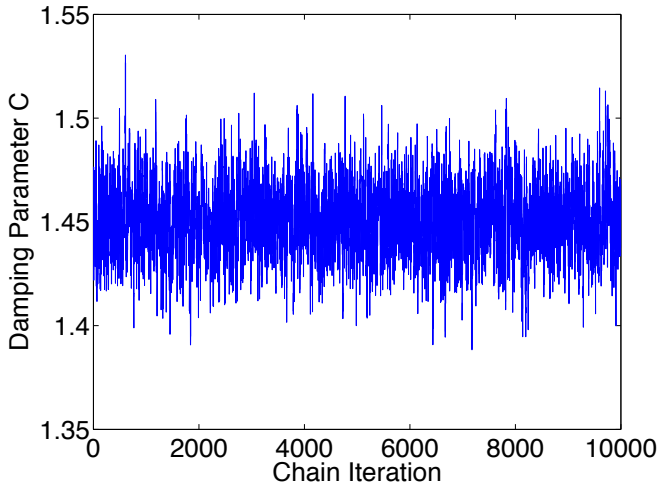
Note: Kernel density estimator (KDE) used to construct density.

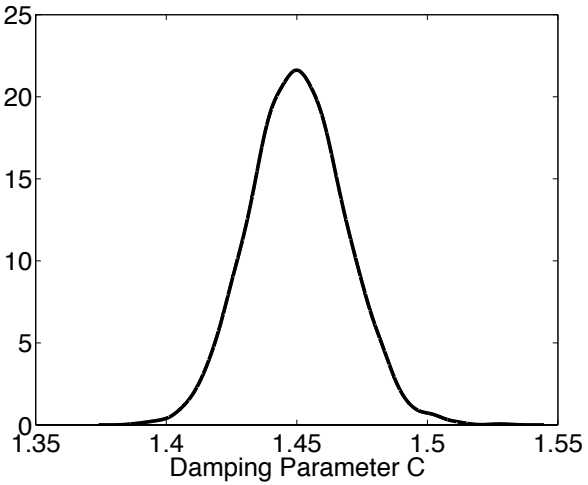
Posted MATLAB Code: spring_mcmc_C.m

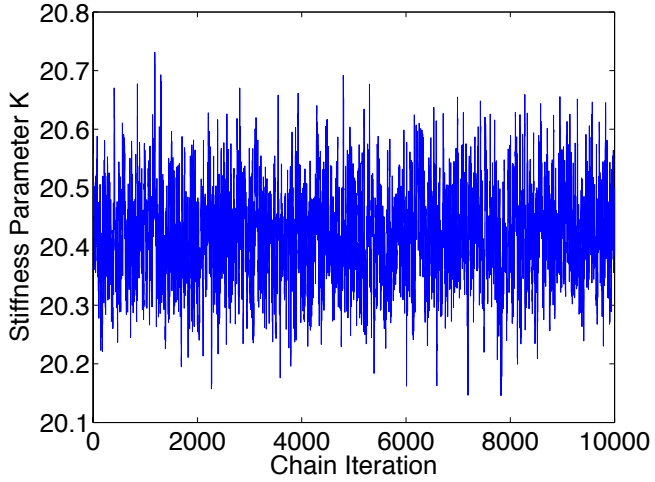
Random Walk Metropolis

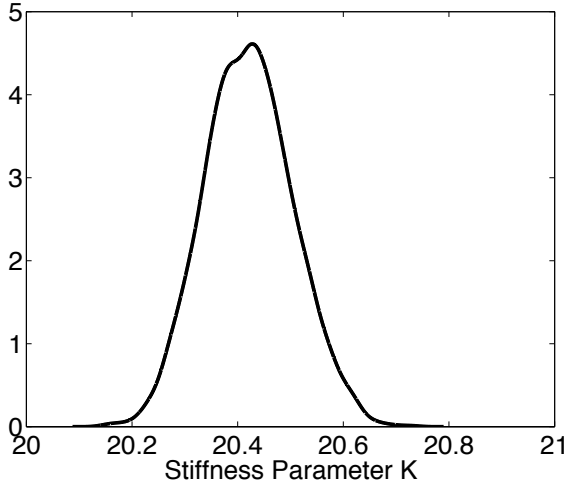
Case ii: Take $\theta = [C, K, \sigma^2]$ with $J(\theta^*|\theta^{k-1}) = N(\theta^{k-1}, V)$ and

$$V = \begin{bmatrix} 0.000345 & 0.000268 \\ 0.000268 & 0.007071 \end{bmatrix}$$









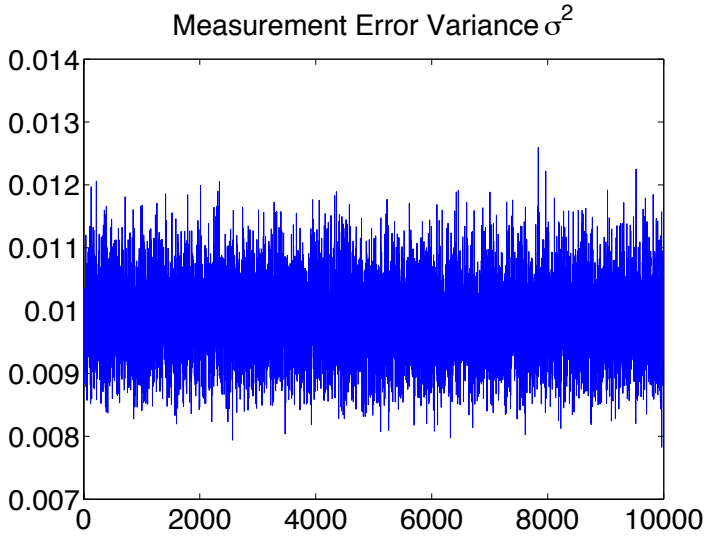
Note:

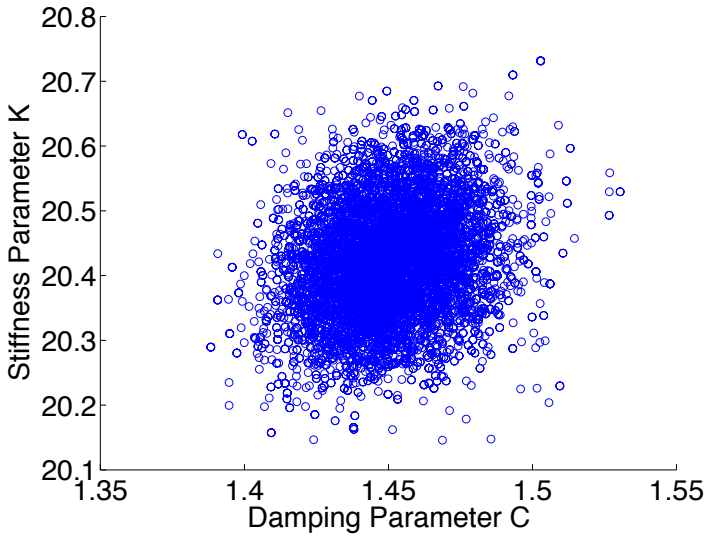
$$2\sigma_C \approx 0.04$$
$$\Rightarrow \sigma_C^2 \approx 0.4 \times 10^{-3}$$

$$2\sigma_K \approx 0.18$$
$$\Rightarrow \sigma_K^2 \approx 0.0081$$

Random Walk Metropolis

Case ii: Measurement error variance and joint samples





Posted MATLAB Code: spring_mcmc_C_K_sigma.m

Delayed Rejection Adaptive Metropolis (DRAM)

Example: HIV model

$$\dot{T}_1 = \lambda_1 - d_1 T_1 - (1-\varepsilon) k_1 V T_1$$

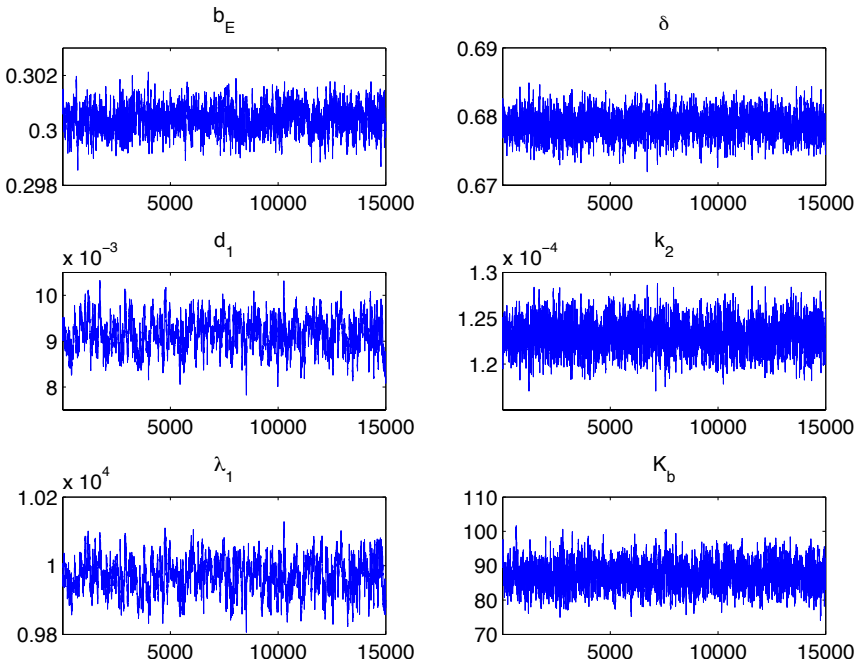
$$\dot{T}_2 = \lambda_2 - d_2 T_2 - (1-f\varepsilon) k_2 V T_2$$

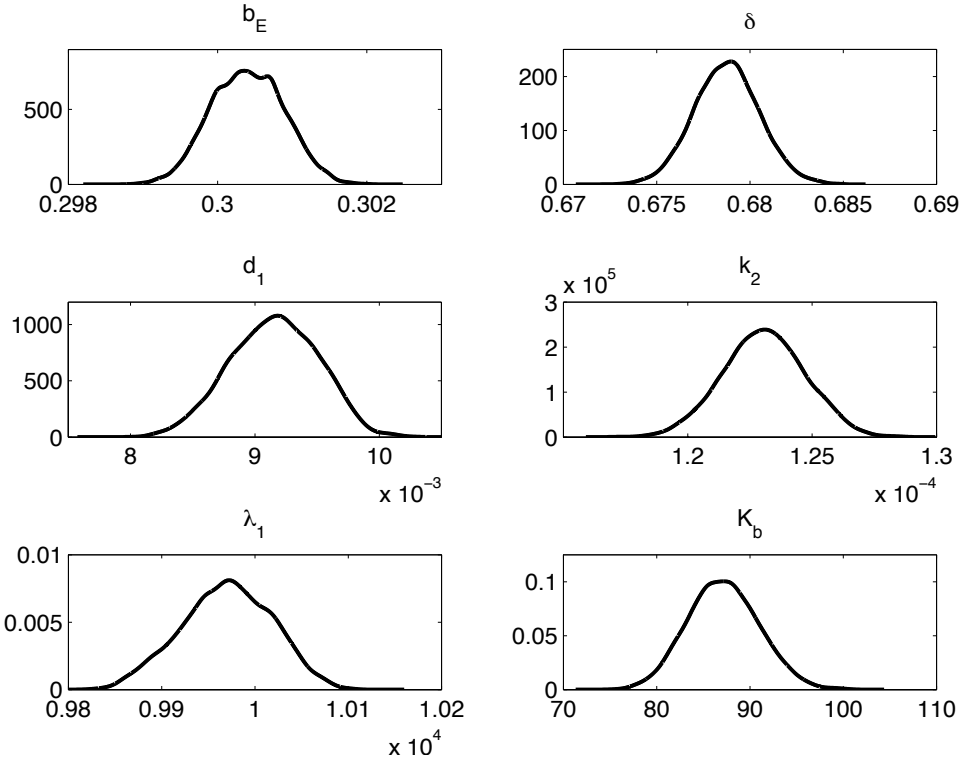
$$\dot{T}_1^* = (1-\varepsilon) k_1 V T_1 - \delta T_1^* - m_1 E T_1^*$$

$$\dot{T}_2^* = (1-f\varepsilon) k_2 V T_2 - \delta T_2^* - m_2 E T_2^*$$

$$\dot{V} = N_T \delta (T_1^* + T_2^*) - cV - [(1-\varepsilon)\rho_1 k_1 T_1 + (1-f\varepsilon)\rho_2 k_2 T_2] V$$

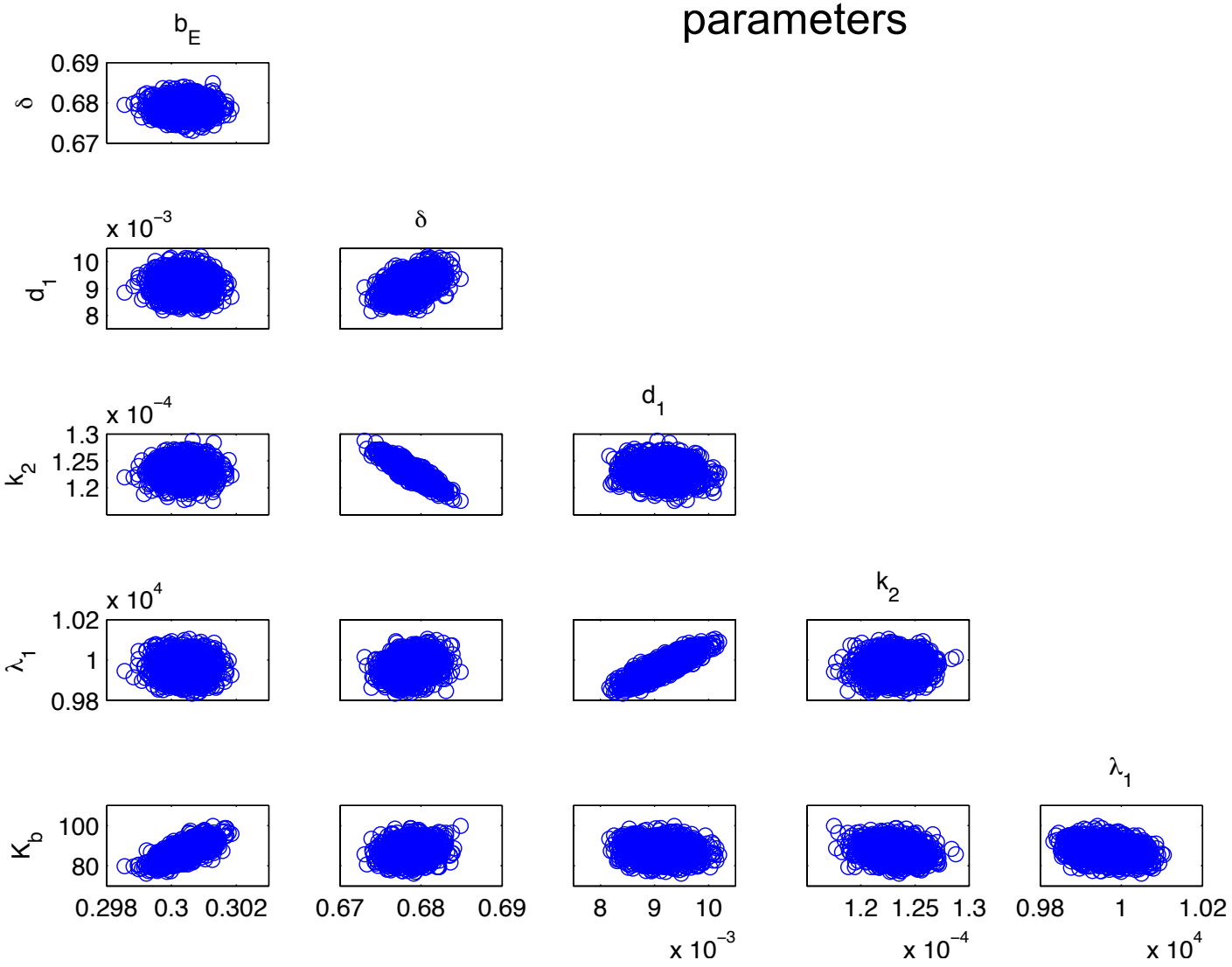
$$\dot{E} = \lambda_E + \frac{b_E (T_1^* + T_2^*)}{T_1^* + T_2^* + K_b} E - \frac{d_E (T_1^* + T_2^*)}{T_1^* + T_2^* + K_d} E - \delta_E E.$$





Delayed Rejection Adaptive Metropolis (DRAM)

Example: HIV model

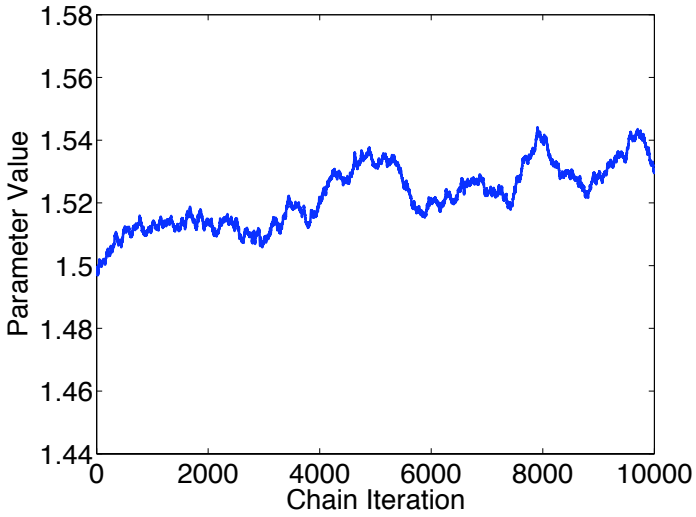


Note: Correlated versus nonidentifable parameters

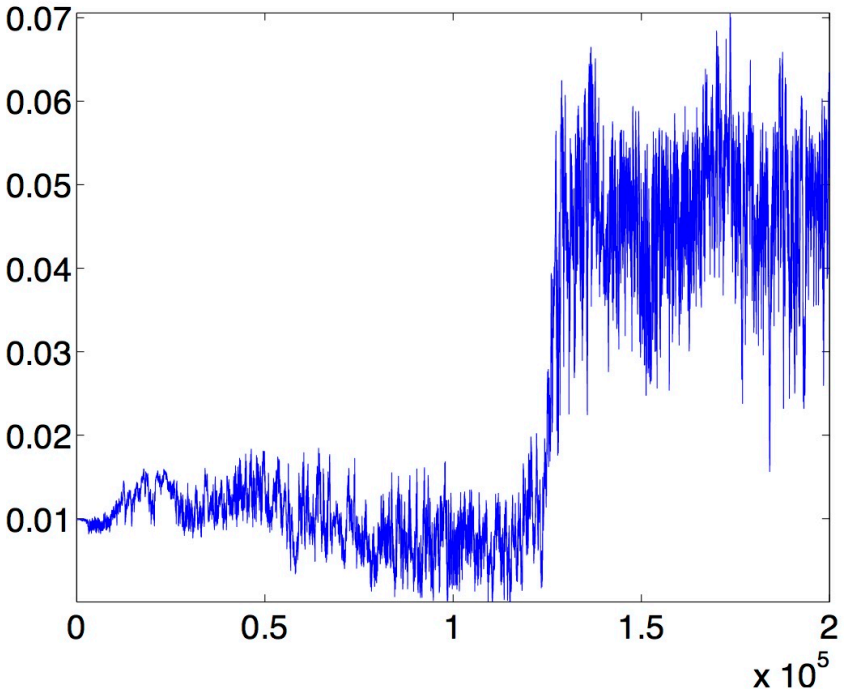
Chain Convergence (Burn-In)

Techniques:

- Visually check chains
- Statistical tests
- Often abused in the literature



Chain not converged



Chain for nonidentifiable parameter

Effects of Parameter Non-identifiability

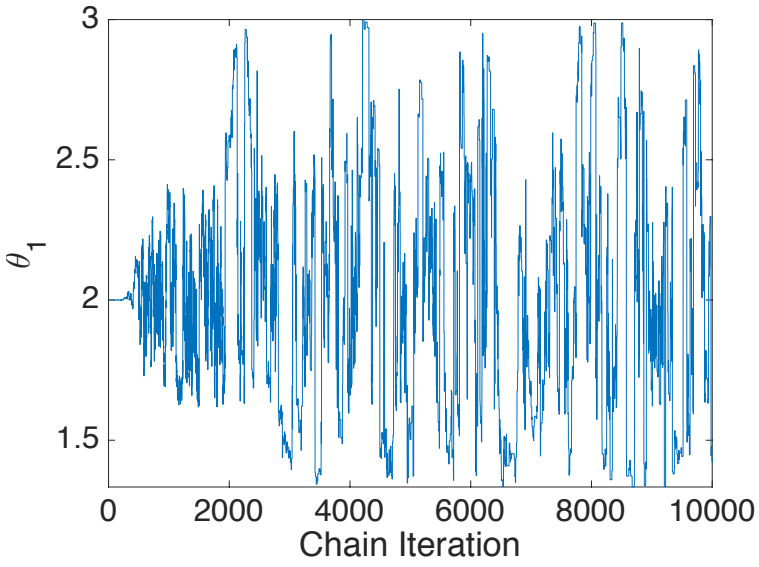
Example:

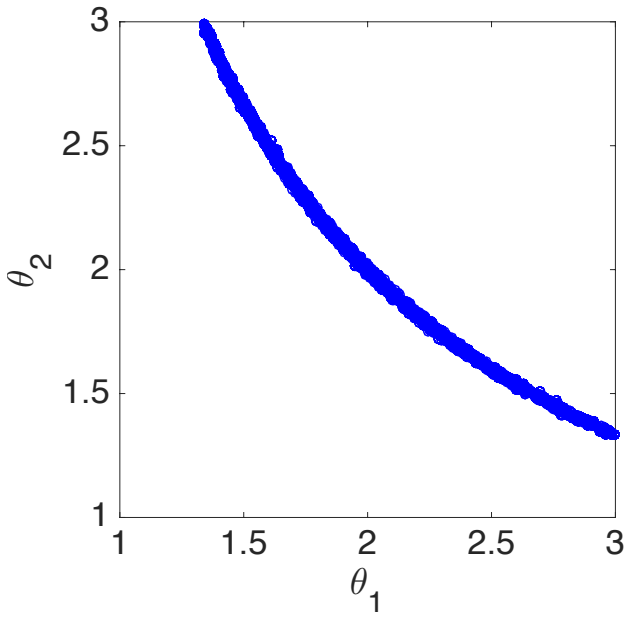
$$y_i = \theta_1 \theta_2 t_i + \varepsilon_i \ , \ i = 1, \ldots, n$$

Parameter values: $\theta_1 = \theta_2 = 2$

Times: $t_i \in [0, 1]$

Prior: $\mathcal{U}^2(4/3, 3)$





Note: Non-identifiable on manifold $h(\theta_{sub}) = 4 - \theta_1 \theta_2$

Uncertainty Propagation

Setting:

- We assume that we have determined distributions for parameters
  - e.g., Bayesian inference, experiments

$$\dot{T}_1 = \lambda_1 - d_1 T_1 - (1-\varepsilon) k_1 V T_1$$

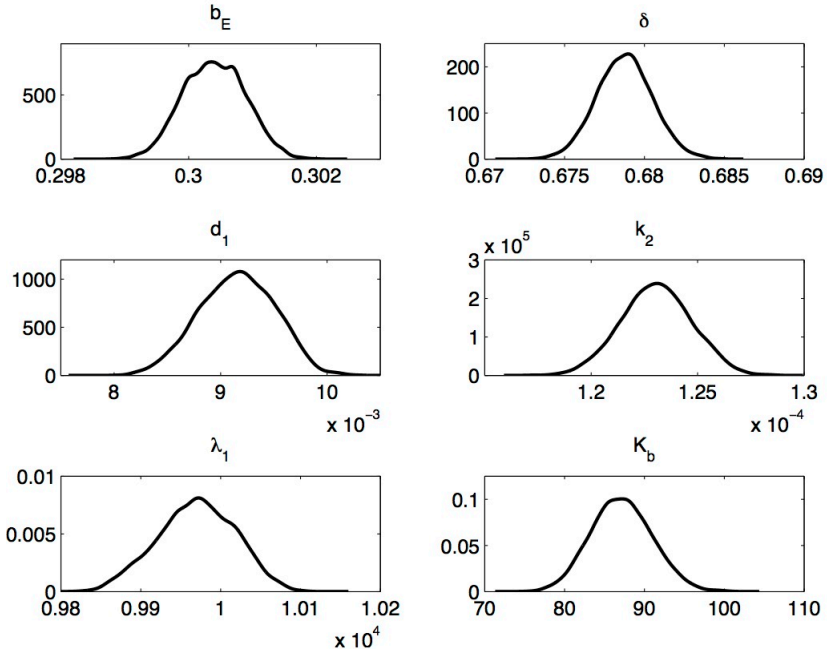
$$\dot{T}_2 = \lambda_2 - d_2 T_2 - (1-f\varepsilon) k_2 V T_2$$

$$\dot{T}_1^* = (1-\varepsilon) k_1 V T_1 - \delta T_1^* - m_1 E T_1^*$$

$$\dot{T}_2^* = (1-f\varepsilon) k_2 V T_2 - \delta T_2^* - m_2 E T_2^*$$

$$\dot{V} = N_T \delta (T_1^* + T_2^*) - cV - [(1-\varepsilon)\rho_1 k_1 T_1 + (1-f\varepsilon)\rho_2 k_2 T_2] V$$

$$\dot{E} = \lambda_E + \frac{b_E (T_1^* + T_2^*)}{T_1^* + T_2^* + K_b} E - \frac{d_E (T_1^* + T_2^*)}{T_1^* + T_2^* + K_d} E - \delta_E E$$



Goal: Construct statistics for QoI

- e.g., Expected viral load in HIV patient with appropriate uncertainty intervals
- Note: Often involves moderate to high-dimensional integration

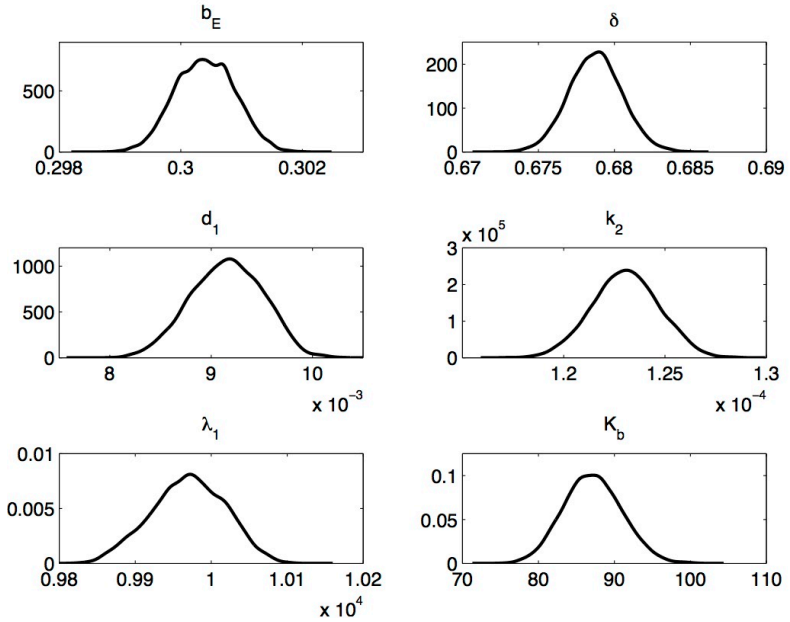
$$\mathbb{E}[V(t)] = \int_{\mathbb{R}^6} V(t,q)\rho(q)dq$$

Issues:

- How do we efficiently propagate input uncertainties through models? Surrogate models.
- How do we approximately integrate in moderate to high dimensions; e.g., p = 50-100?

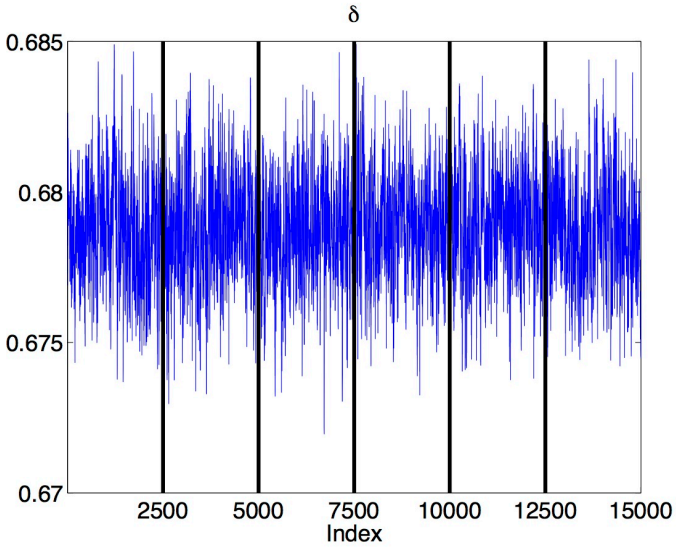
Propagation of Uncertainty in Models – HIV Example

Parameter Densities:

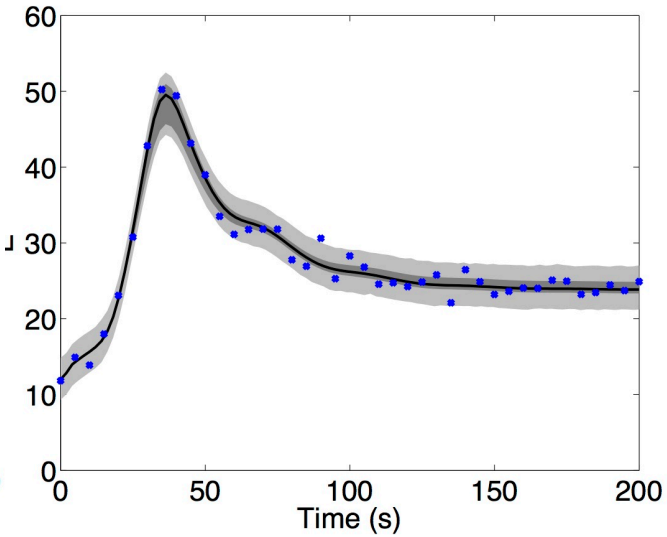


Techniques:

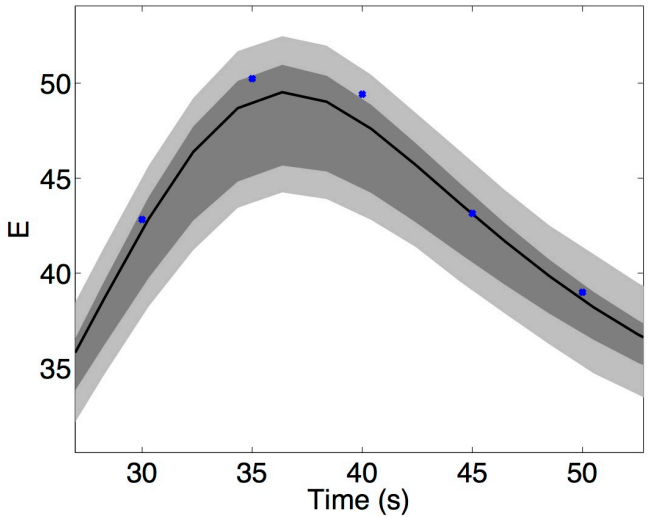
- Sample from parameter densities to construct prediction intervals for QoI.
- Slow convergence rate $O(1/\sqrt{M})$
- 100-fold more evaluations required to gain additional place of accuracy.
- Significant numerical analysis used to efficiently propagate densities.



Samples from Chain



Data, Credible Intervals and Prediction Intervals



Non-Gaussian Credible and Prediction Intervals

Forward Uncertainty Propagation: Sampling Methods

Strategy: Randomly sample from parameter and measurement error distributions and propagate through model to quantify response uncertainty.

Advantages:

- Applicable to nonlinear models.
- Parameters can be correlated and non-Gaussian.
- Straight-forward to apply and convergence rate is independent of number of parameters.
- Can directly incorporate both parameter and measurement uncertainties.

Disadvantages:

- Very slow convergence rate: $\mathcal{O}(1/\sqrt{M})$ where M is the number of samples.
- 100-fold more evaluations required to gain additional place of accuracy.

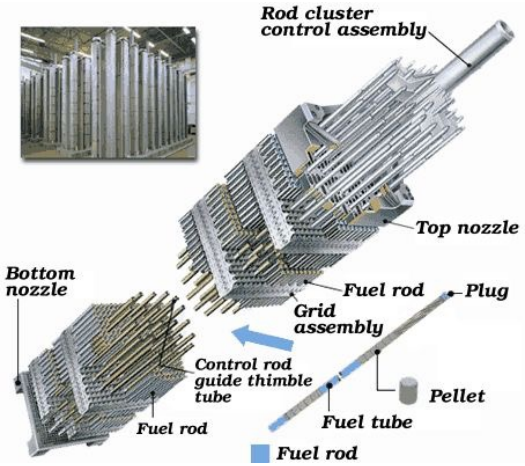
Use of Prediction Intervals: Nuclear Power Plant Design

Subchannel Code (COBRA-TF): numerous closure relations, ~70 parameters

e.g., Dittus—Boelter Relation

$$Nu = 0.023 Re^{0.8} Pr^{0.4}$$

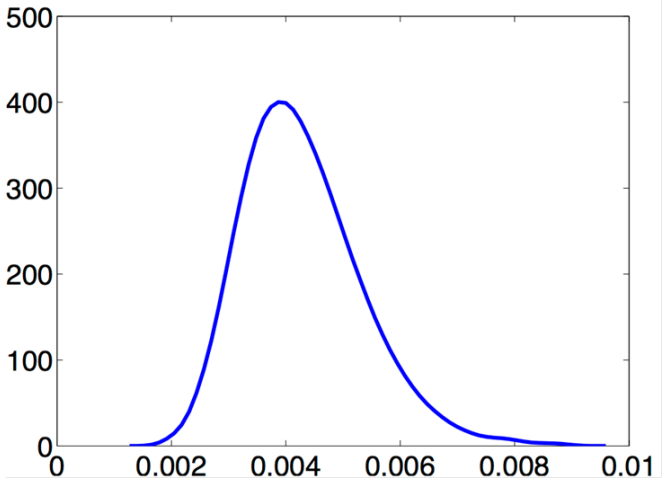
Nu: Nusselt number
Re: Reynolds number
Pr: Prandtl number



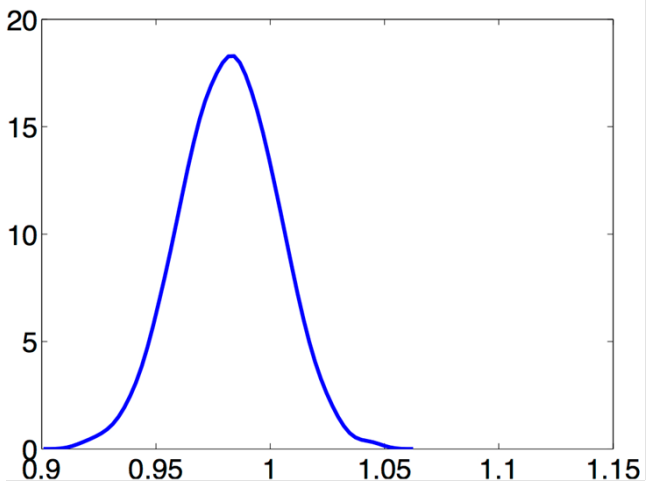
Industry Standard: Employ conservative, uniform, bounds

i.e., [0, 0.046], [0, 1.6], [0,0.8]

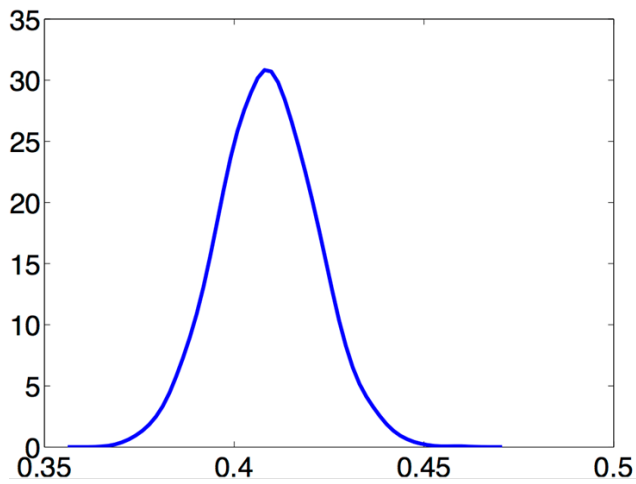
Bayesian Analysis: Employ conservative bounds as priors



$2\sigma \approx 0.0035$



$2\sigma \approx 0.06$

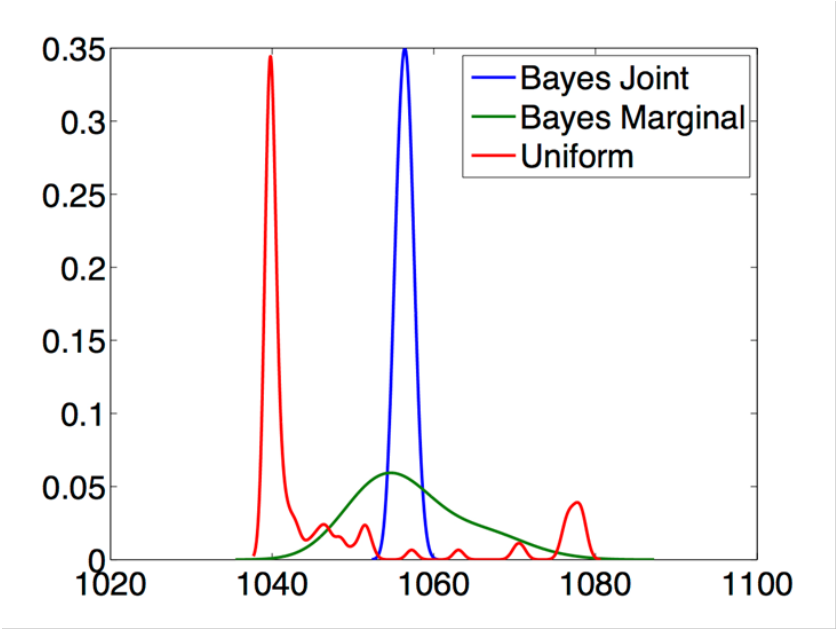


$2\sigma \approx 0.03$

Note: Substantial reduction in parameter uncertainty

Use of Prediction Intervals: Nuclear Power Plant Design

Strategy: Propagate parameter uncertainties through COBRA-TF to determine uncertainty in maximum fuel temperature



Notes:

- Temperature uncertainty reduced from 40 degrees to 5 degrees
- Can run plant 20 degrees hotter, which significantly improves efficiency

Ramification: Savings of 10 billion dollars per year for US power plants

Issues:

- We considered only one of many physical relations
- Nuclear regulatory commission takes years to change requirements and codes

Good News: We are now working with Westinghouse to reduce uncertainties.

Delayed Rejection Adaptive Metropolis (DRAM)

Websites

- https://rsmith.math.ncsu.edu/UQ_TIA/CHAPTER8/index_chapter8.html
- https://mjlaine.github.io/mcmcstat/

Delayed Rejection Adaptive Metropolis (DRAM)

We fit the Monod model

$$y = \theta_1 \frac{t}{\theta_2 + t} + \varepsilon \ , \ \varepsilon \sim \mathcal{N}(0, I\sigma^2)$$

to observations

x (mg / L COD): 28 55 83 110 138 225 375

y (1 / h): 0.053 0.060 0.112 0.105 0.099 0.122 0.125

First clear some variables from possible previous runs.

```
clear data model options
```

Next, create a data structure for the observations and control variables. Typically, one could make a structure data that contains fields xdata and ydata.

```
data.xdata = [28 55 83 110 138 225 375]';  % x (mg / L COD)
data.ydata = [0.053 0.060 0.112 0.105 0.099 0.122 0.125]'; % y (1 / h)
```

Construct model

```
modelfun = @(x,theta) theta(1)*x./(theta(2)+x);
ssfun = @(theta,data) sum((data.ydata-modelfun(data.xdata,theta)).^2);
model.ssfun = ssfun;
model.sigma2 = 0.01^2;
```

Delayed Rejection Adaptive Metropolis (DRAM)

Input parameters

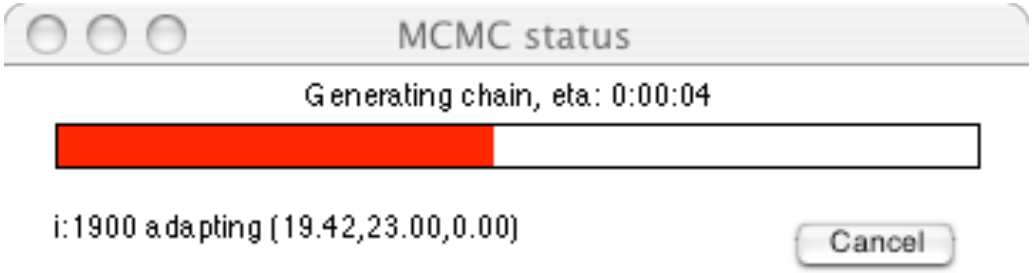
```
params = {
    {'theta1', tmin(1), 0}
    {'theta2', tmin(2), 0} };
```

and set options

```
options.nsimu = 4000;
options.updatesigma = 1;
options.qcov = tcov;
```

Run code

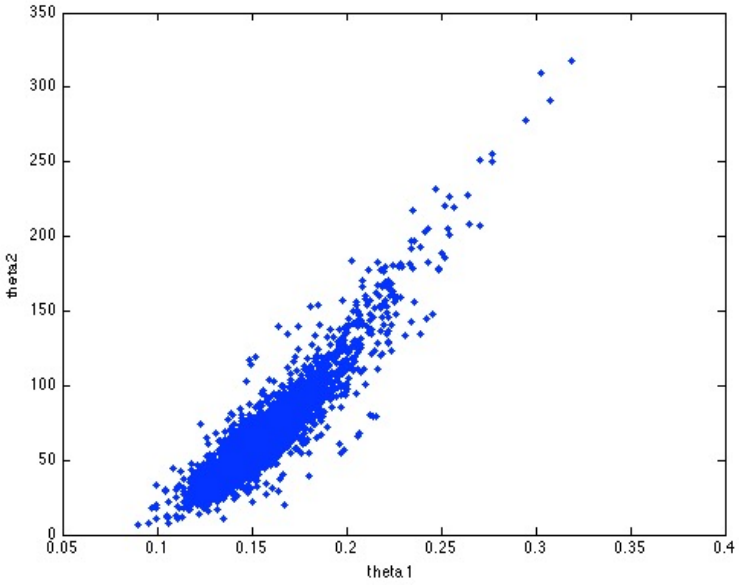
```
[res,chain,s2chain] = mcmcrun(model,data,params,options);
```

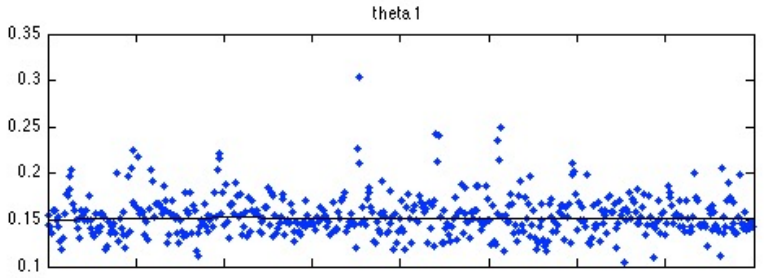


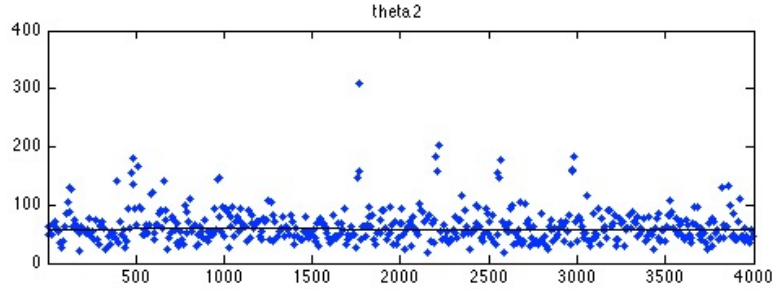
Delayed Rejection Adaptive Metropolis (DRAM)

Plot results

```
figure(2); clf
mcmcplot(chain,[],res,'chainpanel');
figure(3); clf
mcmcplot(chain,[],res,'pairs');
```



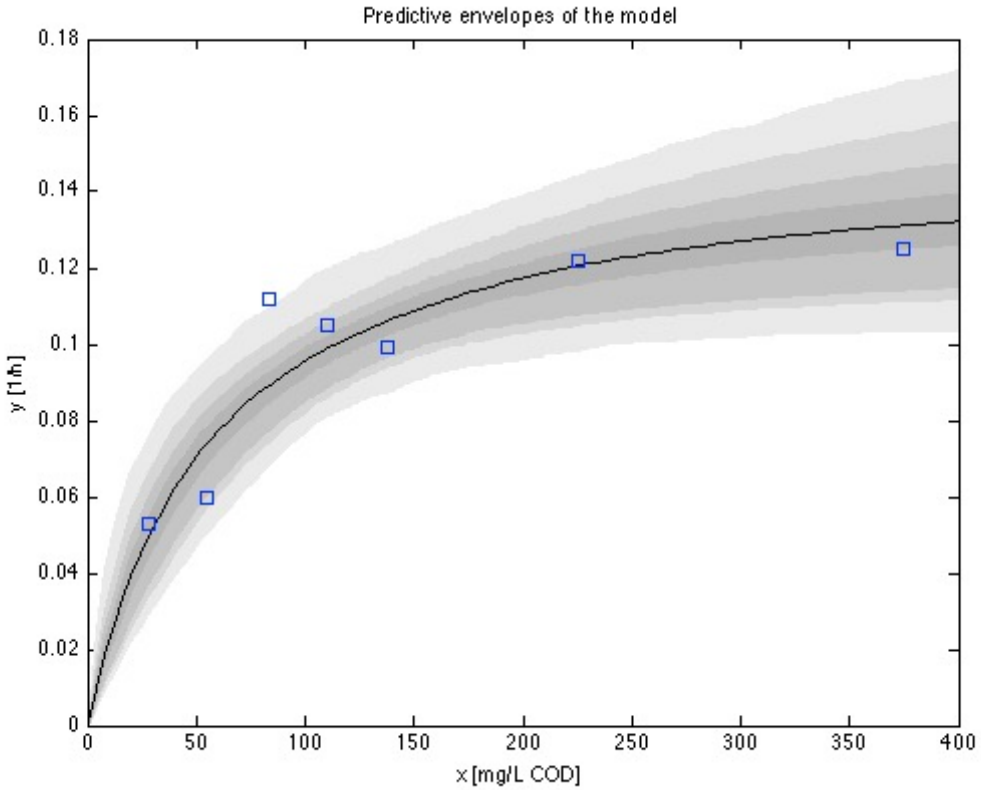




Examples:

- Several available in MCMC_EXAMPLES
- ODE solver illustrated in algae example

Delayed Rejection Adaptive Metropolis (DRAM)

Construct credible and prediction intervals

```
figure(5); clf
out = mcmcpred(res,chain,[],x,modelfun);
mcmcpredplot(out);
hold on
plot(data.xdata,data.ydata,'s'); % add data points to the plot
xlabel('x [mg/L COD]');
ylabel('y [1/h]');
hold off
title('Predictive envelopes of the model')
```