POLITECNICO DI MILANO

Markov Chain Monte Carlo (MCMC) for model and parameter identification

N. Pedroni, nicolapedroni@gmail.com

Multidisciplinary Course: **Monte Carlo Simulation Methods for the Quantitative Analysis of Stochastic and Uncertain Systems**

Problem statement: statistical inference

(e.g., a mechanical component)

Real system

(e.g., failure times)

Experimental data

$$\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$$

System model

(parameters)

$$\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_n\}$$

(e.g., Exponential)

(e.g., failure rate)

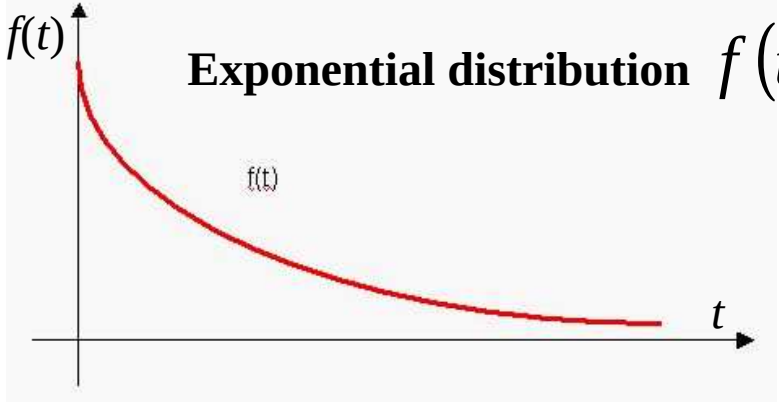
On the basis of the experimental data *x*, estimate the (unknown) parameters *θ* of the (known) model

$$\hat{\boldsymbol{\theta}}$$

Statistical inference – Example: component with constant failure rate

SYSTEM MODEL (hypothesis)

Component failure time



Exponential distribution $f(t) = \lambda e^{-\lambda t} = \theta e^{-\theta t}$

EXPERIMENTAL DATA (from **real** components)

$\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\} = \boldsymbol{t} = \{t_1, t_2, \ldots, t_N\} = \{1.71\text{h}, 48.8\text{h}, \ldots, 0.39\text{h}\}$

(N = 100 failure times from N = 100 'identical' components)

MODEL PARAMETER θ? (= FAILURE RATE λ?)

Statistical inference: classical (frequentist) approach

- **Only** the (random) **data** are used to estimate the parameter

(e.g.) MAXIMUM LIKELIHOOD ESTIMATION

$$\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\} = \boldsymbol{t} = \{t_1, t_2, \ldots, t_N\} \ (N = 100 \text{ failure times})$$

$$\hat{\theta} = \hat{\lambda} = \left[\frac{\text{failures}}{\text{time}}\right] = \frac{\text{Number of collected failures}}{\text{Total observed time}} = \frac{N}{\sum_{i=1}^{N} t_i} = \frac{100}{3.05 \cdot 10^3 \text{ h}} = 3.28 \cdot 10^{-2} \text{ h}^{-1}$$

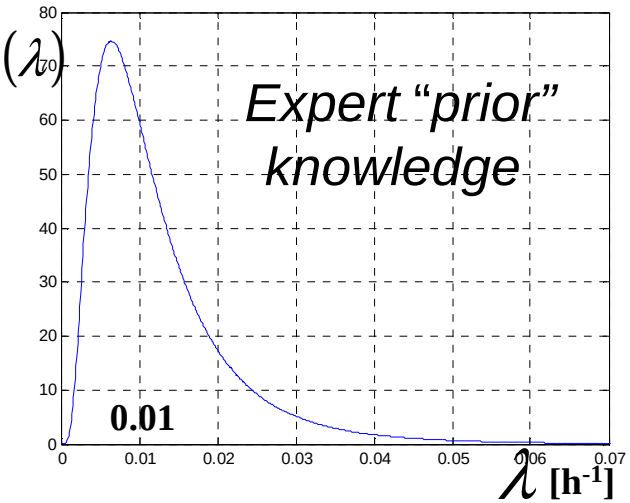
(NB: also confidence intervals can be computed, reflecting the variability in the random data)

Statistical inference: the Bayesian approach

- In addition to the (random) data, it uses the analyst's **degree of belief** (**state of knowledge**) about the true value of the parameter

- The analyst's **state of knowledge** about the parameter **before** obtaining the data is represented by a **(prior) probability distribution**
(*Bayesian interpretation of probability*)

$p(\boldsymbol{\theta}) = p(\lambda)$



+

$\boldsymbol{x} = \boldsymbol{t} = \{t_1, t_2, \ldots, t_N\}$

(N = 100 failure times)

COMBINATION TO OBTAIN THE "POSTERIOR" DISTRIBUTION $p(\boldsymbol{\theta} \mid \boldsymbol{x}) = p(\lambda \mid \boldsymbol{t})$

Bayesian statistical inference: definitions

$$p(\boldsymbol{\theta} \text{ and } \boldsymbol{x}) = p(\boldsymbol{\theta} \mid \boldsymbol{x})p(\boldsymbol{x}) = p(\boldsymbol{x} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})$$

Definition of 'and'

$$p(\boldsymbol{\theta} \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{x})} = \frac{p(\boldsymbol{x} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\boldsymbol{x} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}}$$

Bayes theorem

$p(\boldsymbol{\theta} \mid \boldsymbol{x}) = \pi(\boldsymbol{\theta} \mid \boldsymbol{x})$ **Posterior distribution of the parameters given the data**
(objective of the analysis)

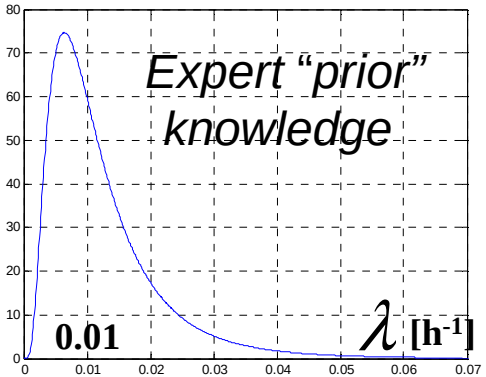
$p(\boldsymbol{x} \mid \boldsymbol{\theta}) = L(\boldsymbol{\theta} \mid \boldsymbol{x})$ **Likelihood of the parameters given dataset**
(dependent on the functional form of the *chosen* model)

$p(\boldsymbol{\theta})$ **Prior distribution of the parameters**
(dependent on the *a priori knowledge* on the values of the parameters)

$p(\boldsymbol{x})$ **Probability of the experimental dataset**
("normalization" factor: requires the evaluation of a complex integral)

Bayesian statistical inference – Example: component with constant failure rate

$p(\boldsymbol{\theta}) = p(\lambda)$



+

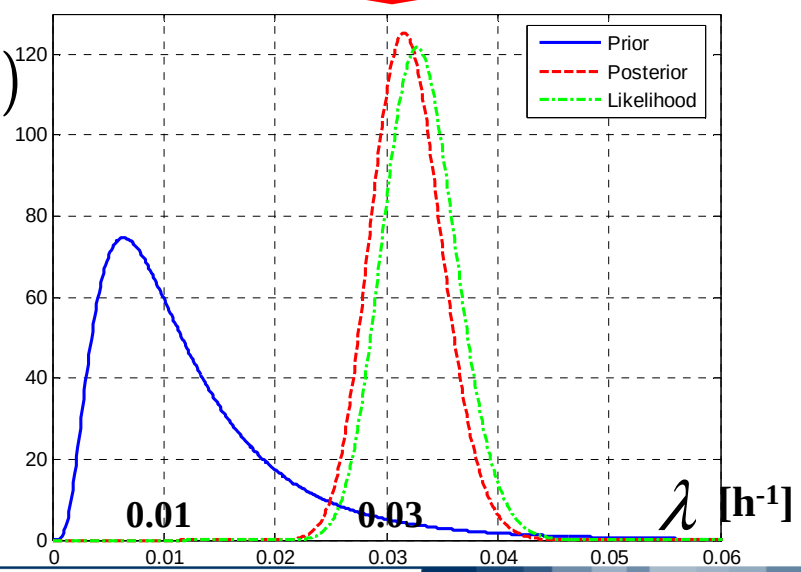
$\boldsymbol{x} = \boldsymbol{t} = \{t_1, t_2, \ldots, t_N\}$

(N = 100 failure times)

Exponential model for the failure times $f(t) = \lambda e^{-\lambda t}$ → "Exponential likelihood"

$$L(\theta \mid \boldsymbol{x}) = L(\lambda \mid \boldsymbol{t}) = (\lambda \exp[-\lambda t_1]) \cdot (\lambda \exp[-\lambda t_2]) \cdot \ldots \cdot (\lambda \exp[-\lambda t_N]) = \lambda^N \exp\left[-\lambda \sum_{i=1}^{N} t_i\right]$$

$\pi(\lambda \mid \boldsymbol{t})$



Bayesian statistical inference: Markov Chain Monte Carlo (MCMC) simulation approach

It generates samples according to <u>any desired probability distribution</u>

Randomly chosen value $\boldsymbol{\theta}^0$

$\Downarrow$

MCMC rules + **Data, x**

$\Downarrow$

$$\left\{\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \ldots, \boldsymbol{\theta}^j, \ldots, \boldsymbol{\theta}^{N_s}\right\}$$

If N_s is large enough, then $\left\{\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \ldots, \boldsymbol{\theta}^j, \ldots, \boldsymbol{\theta}^{N_s}\right\} \sim \pi\left(\boldsymbol{\theta} \mid \mathbf{x}\right)$

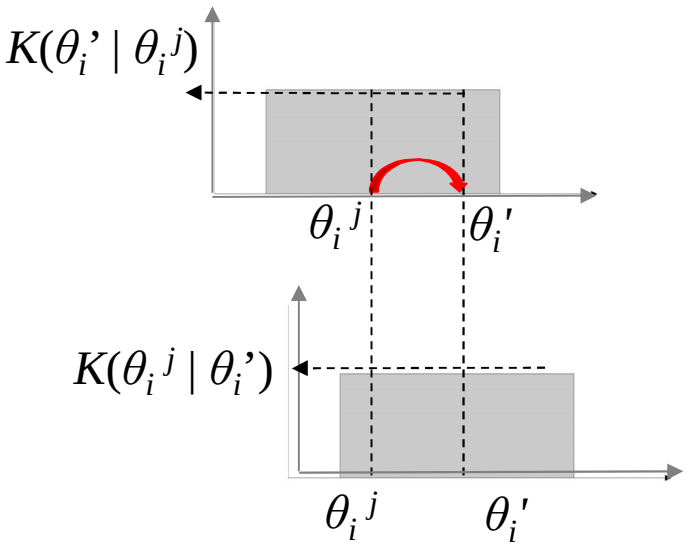
independently on the initial point $\boldsymbol{\theta}^0$

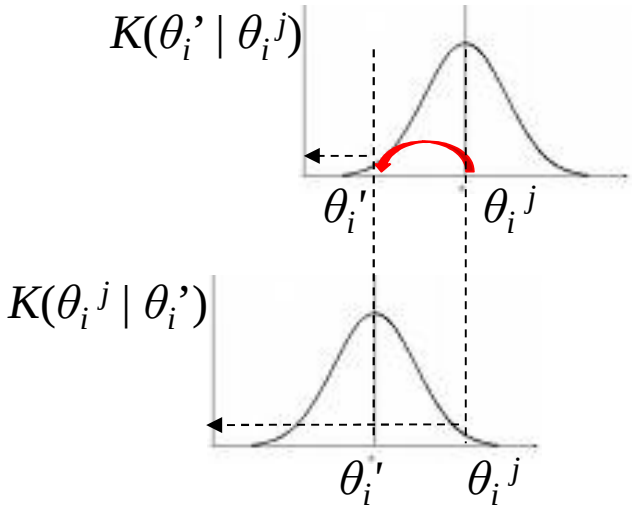
MCMC simulation: the Metropolis-Hastings algorithm (1)

- Randomly choose an initial value $\boldsymbol{\theta}^0 = \{\theta_1^0, \theta_2^0, \ldots, \theta_i^0, \ldots, \theta_n^0\}$ for the Markov chain
- For $j = 0, 1, \ldots, N_s$ (chain length)

1. Sample a proposal value $\boldsymbol{\theta}' = \{\theta_1', \theta_2', \ldots, \theta_i', \ldots, \theta_n'\}$ from a proposal distribution $K(\boldsymbol{\theta}'|\boldsymbol{\theta}^j)$

   Example for variable θ_i, $i = 1, 2, \ldots, n$





$K(\theta_i' \mid \theta_i^j)$

$=$

$K(\theta_i^j \mid \theta_i')$

SYMMETRIC PROPOSAL FOR CONVENIENCE …

MCMC simulation: the Metropolis-Hastings algorithm (2)

2. Calculate the acceptance probability α for proposal value $\boldsymbol{\theta}$'

$$\alpha = \min\left\{1, \frac{\pi(\boldsymbol{\theta}' \mid \mathbf{x}) K(\boldsymbol{\theta}^j \mid \boldsymbol{\theta}')}{\pi(\boldsymbol{\theta}^j \mid \mathbf{x}) K(\boldsymbol{\theta}' \mid \boldsymbol{\theta}^j)}\right\}$$

Since $\pi(\boldsymbol{\theta} \mid \mathbf{x}) = \dfrac{L(\boldsymbol{\theta} \mid \mathbf{x}) p(\boldsymbol{\theta})}{p(\mathbf{x})}$

$$\alpha = \min\left\{1, \frac{L(\boldsymbol{\theta}' \mid \mathbf{x}) p(\boldsymbol{\theta}') K(\boldsymbol{\theta}^j \mid \boldsymbol{\theta}')}{L(\boldsymbol{\theta}^j \mid \mathbf{x}) p(\boldsymbol{\theta}^j) K(\boldsymbol{\theta}' \mid \boldsymbol{\theta}^j)}\right\}$$

NO NEED FOR NORMALIZATION FACTOR CALCULATION

3. Accept/reject proposal value $\boldsymbol{\theta}$'

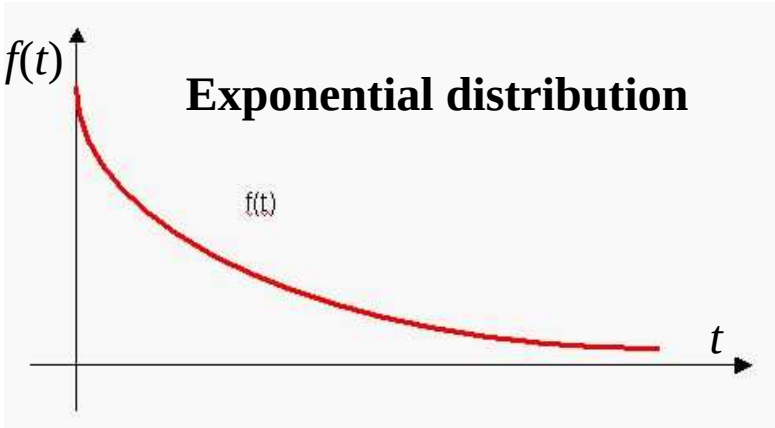
- Draw a uniform random number $r \sim U[0, 1)$
- If $r \leq \alpha$, then accept the proposal value $\boldsymbol{\theta}$ ', i.e. set $\boldsymbol{\theta}^{j+1} = \boldsymbol{\theta}$ '

  else reject the proposal value $\boldsymbol{\theta}$ ', i.e. set $\boldsymbol{\theta}^{j+1} = \boldsymbol{\theta}^j$

- End For ($j = 0, 1, \ldots, N_s$)

Application: component with constant failure rate

Application: component with constant failure rate



Exponential distribution $f(t) = \lambda e^{-\lambda t}$

Failure rate λ = constant

Experimental dataset:

$\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\} =$

$= \boldsymbol{t} = \{t_1, t_2, \ldots, t_N\}$ (N = 100 failure times)

Likelihood function (given by the *chosen* model)

$$L(\theta \mid \boldsymbol{x}) = L(\lambda \mid \boldsymbol{t}) = (\lambda \exp[-\lambda t_1]) \cdot (\lambda \exp[-\lambda t_2]) \cdot \ldots \cdot (\lambda \exp[-\lambda t_N]) = \lambda^N \exp\left[-\lambda \sum_{i=1}^{N} t_i\right]$$

Prior distribution

$$p(\theta) = p(\lambda) = U[0, \lambda_{\max}] = 1/\lambda_{\max}$$ ("uninformative")

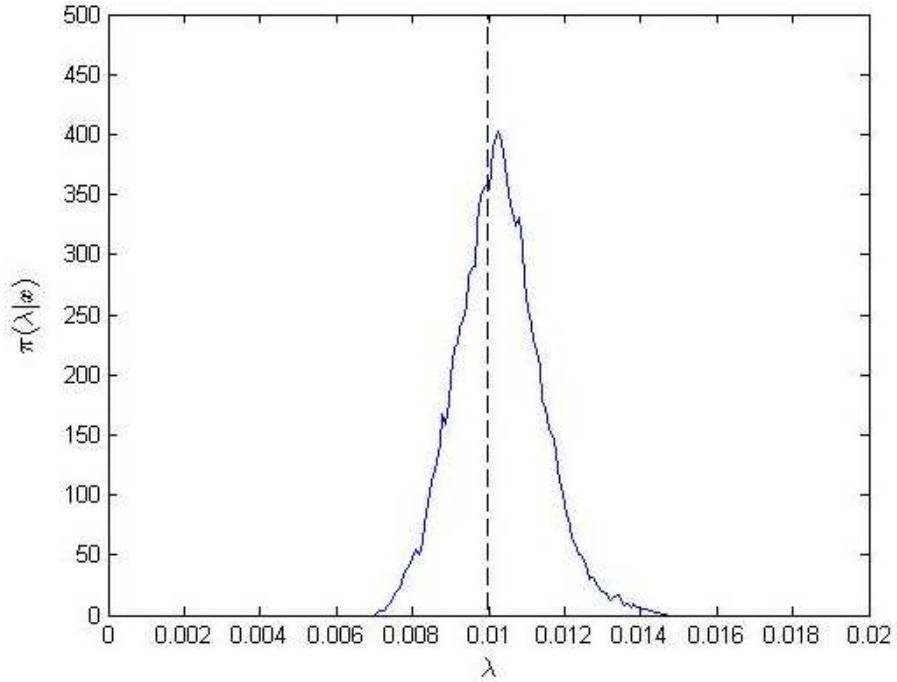
Proposal distribution

$$K(\lambda' \mid \lambda) = N(\lambda, \sigma); K(\lambda \mid \lambda') = N(\lambda', \sigma)$$ ("symmetric")

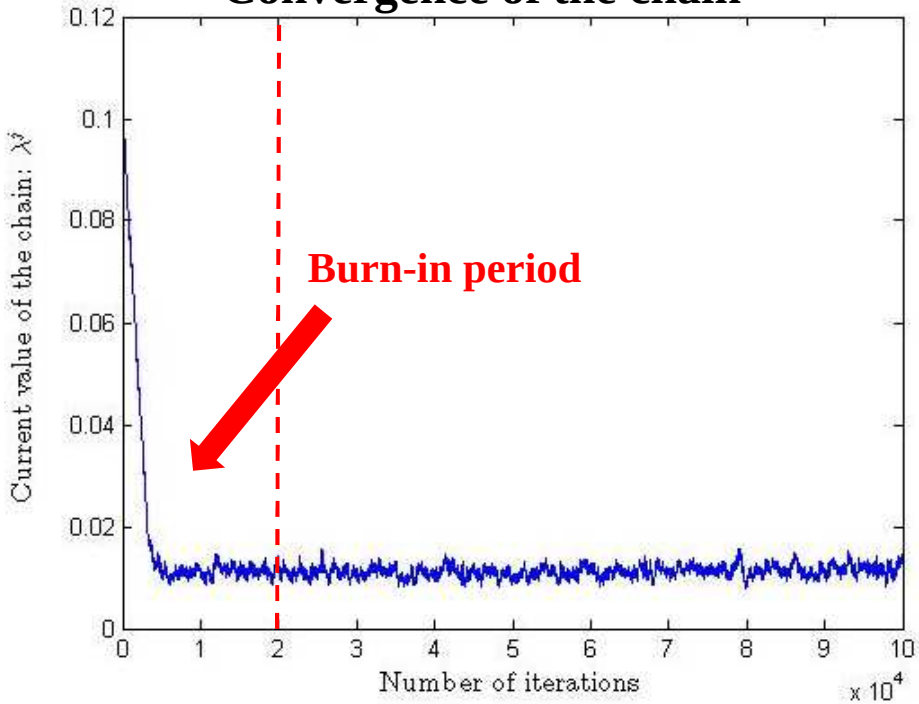
$$\alpha = \min\left\{1, \frac{L(\lambda' \mid \boldsymbol{t}) p(\lambda') K(\lambda \mid \lambda')}{L(\lambda \mid \boldsymbol{t}) p(\lambda) K(\lambda' \mid \lambda)}\right\} = \min\left\{1, \frac{L(\lambda' \mid \boldsymbol{t})}{L(\lambda \mid \boldsymbol{t})}\right\} = \min\left\{1, \frac{(\lambda')^N \exp\left(-\lambda' \sum_{i=1}^{N} t_i\right)}{(\lambda)^N \exp\left(-\lambda \sum_{i=1}^{N} t_i\right)}\right\}$$

Application: results

Posterior distribution



Convergence of the chain

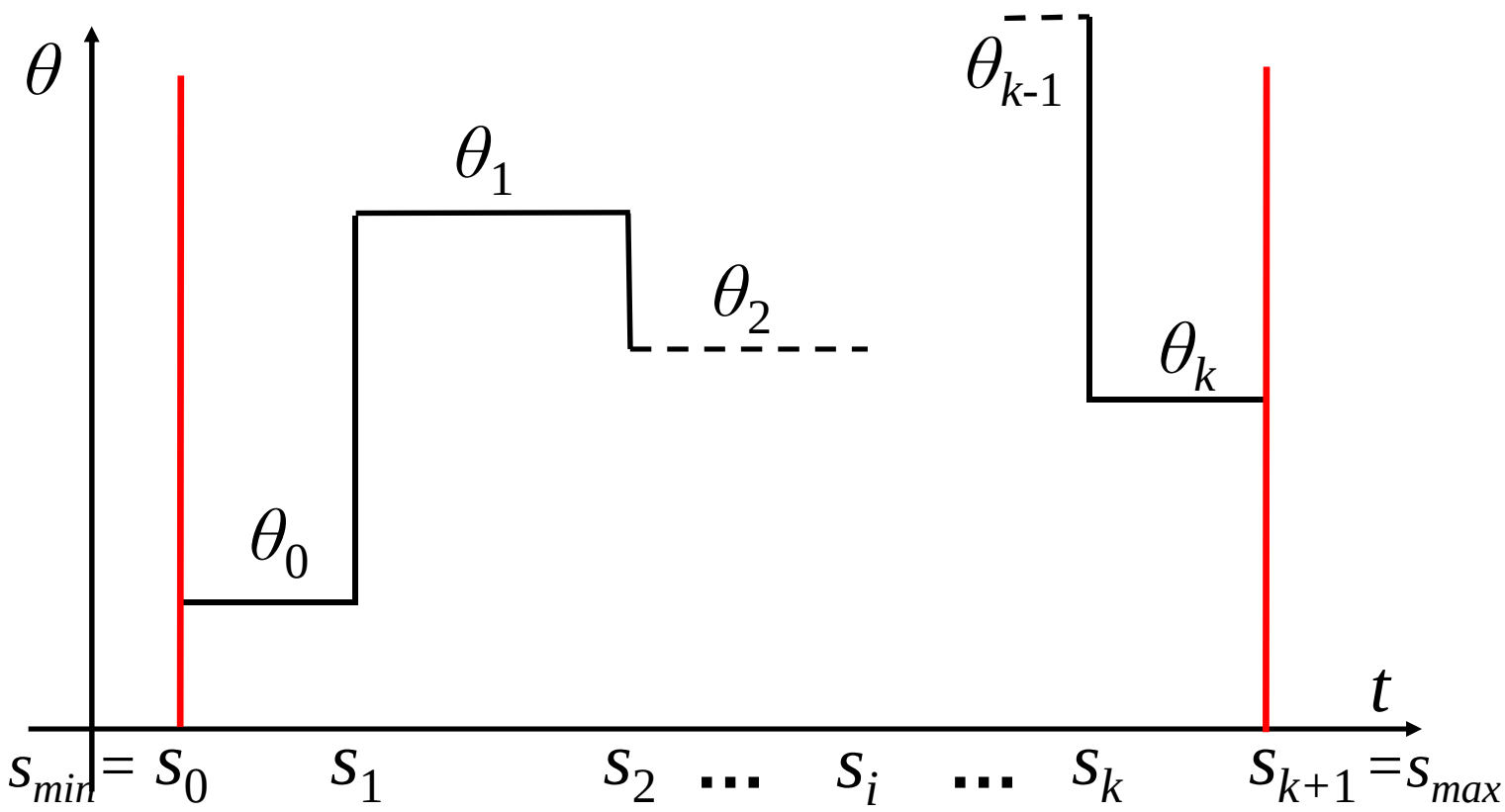


GOOD AGREEMENT WITH "TRUE" VALUE, λ = 0.01

The reversible-jump MCMC algorithm

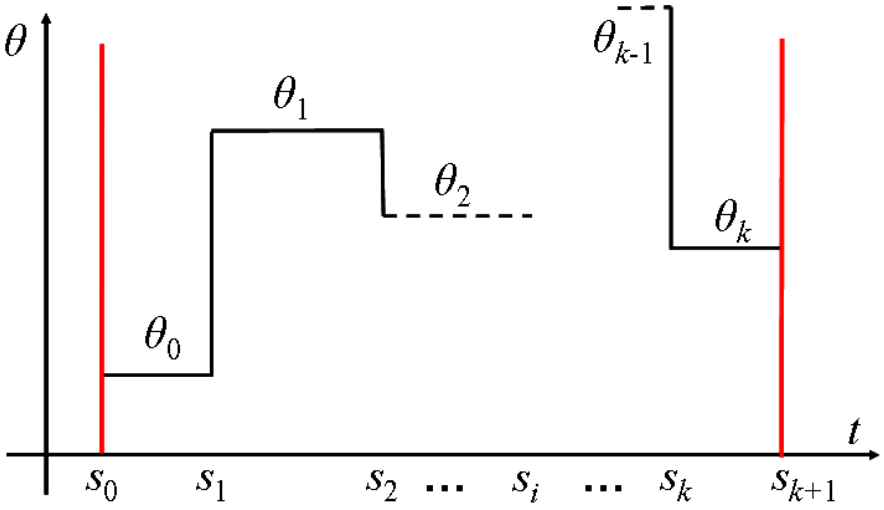
Problem statement

Presence of changepoints in model parameter θ



Assumption: step changes in model parameter θ

Objectives of the Bayesian inference



- Location(s) s_i, $i = 1, 2, \ldots, k$, of the changepoints

$$\pi(s_i \mid \boldsymbol{x}), i = 1, 2, \ldots, k$$

- Values θ_i, $i = 0, 1, \ldots, k$, of the parameter(s) before and after the transitions

$$\pi(\theta_i \mid \boldsymbol{x}), i = 0, 1, \ldots, k$$

- **Number *k* of changepoints**

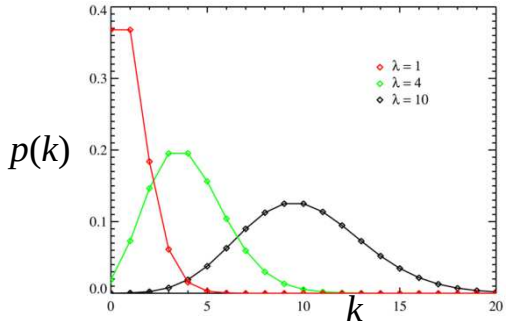
$$\pi(k \mid \boldsymbol{x})$$

THE NUMBER OF CHANGEPOINTS IS NOT KNOWN → IT IS AN UNCERTAIN VARIABLE!

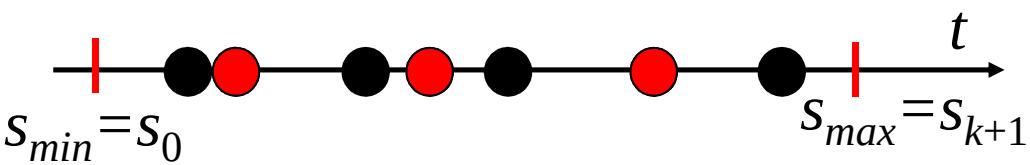
Reversible-jump MCMC: prior distributions

Prior distribution $p(k)$ for the number of changepoints k:

$$p(k) = \frac{e^{-\lambda}\lambda^k}{k!}$$ **(Truncated) Poisson distribution**



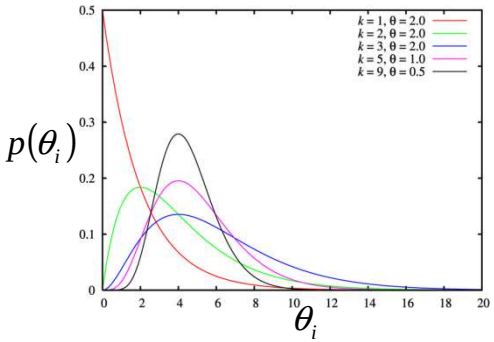
Given k, prior distribution for step positions $s_1, s_2, \ldots, s_k$:



Even numbered order statistics from $(2k + 1)$ points uniform in $[s_0, s_{k+1}]$

Given the step positions $s_1, s_2, \ldots, s_k$, prior distributions for parameters $\theta_0, \theta_1, \ldots, \theta_k$:

$$p(\theta_i) = \frac{\beta^\alpha}{\Gamma(\alpha)}\theta^{\alpha-1}e^{-\beta\theta}$$ **Gamma distribution**

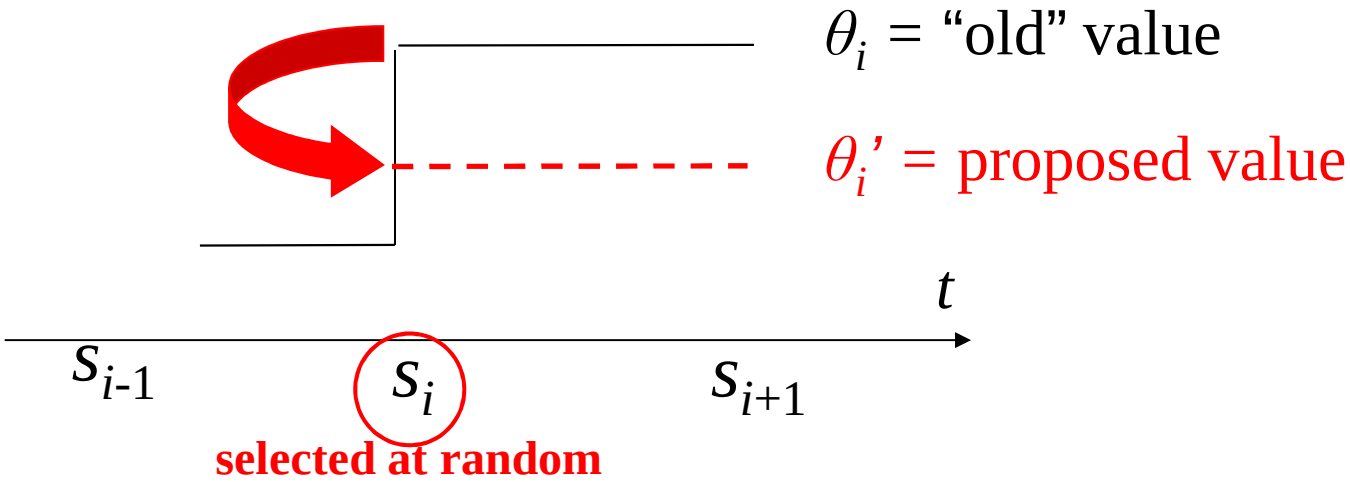


Reversible-jump MCMC: possible random moves

Four possible random moves to explore the uncertain parameter space:

1. **The parameter value θ is varied at a random location s_i**
2. **A randomly chosen step location s_i is moved**
3. **A new step location is created at random in the interval $[s_0, s_{k+1}]$ (*birth move*)**
4. **A randomly chosen location s_i is eliminated (*death move*)**

Reversible-jump MCMC - Random move 1: variation of the parameter value

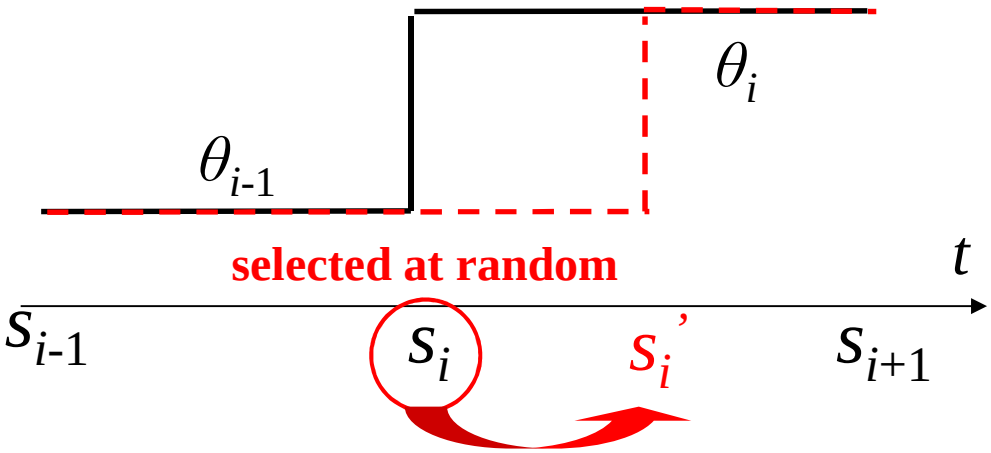


Proposed value: $$\log\left(\frac{\theta_i'}{\theta_i}\right) \approx U[-0.5,+0.5] \rightarrow \theta_i' = \theta_i \cdot e^U$$

Acceptance probability: $$\alpha_\eta = \min\left\{1, l \cdot \left(\frac{\theta_i'}{\theta_i}\right)^\alpha e^{-\beta(\theta_i'-\theta_i)}\right\}$$

where $l = \dfrac{L(\theta_i' \mid \mathbf{x})}{L(\theta_i \mid \mathbf{x})}$ = likelihood ratio (dependent on the problem)

Reversible-jump MCMC - Random move 2: variation of a step location

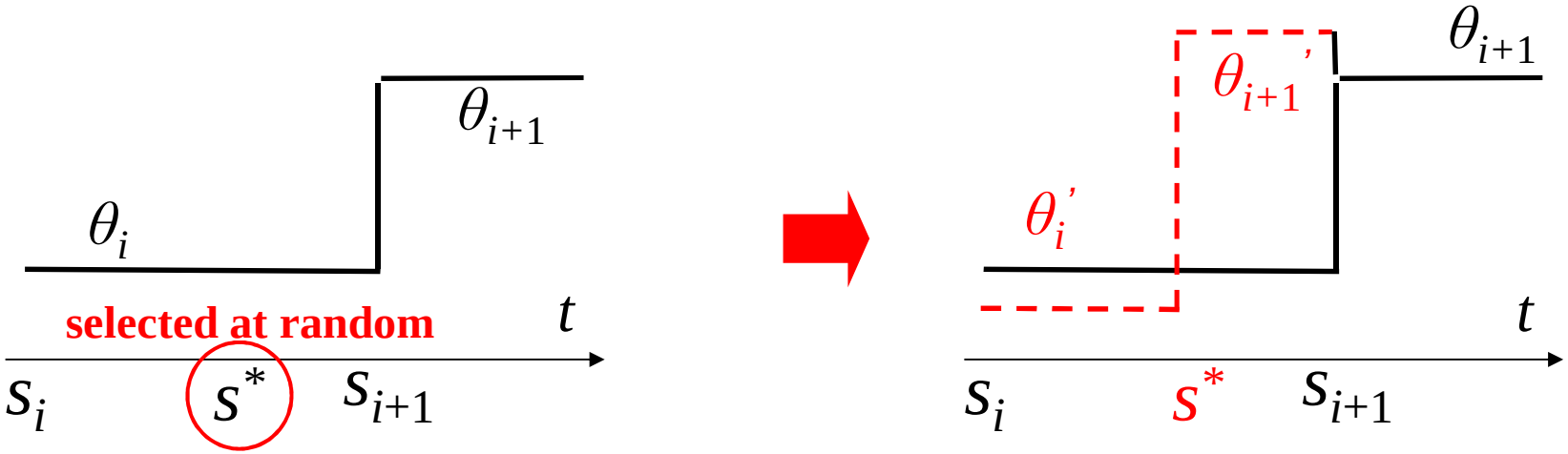


Proposed value: $s_i' \approx U\left[s_{i-1}, s_{i+1}\right]$

Acceptance probability: $\alpha_\chi = \min\left\{1, l \cdot \frac{\left(s_{i+1} - s_i'\right)\left(s_i' - s_{i-1}\right)}{\left(s_{i+1} - s_i\right)\left(s_i - s_{i-1}\right)}\right\}$

where $l = \frac{L\left(s_i' \mid \boldsymbol{x}\right)}{L\left(s_i \mid \boldsymbol{x}\right)}$ = likelihood ratio (dependent on the problem)

Reversible-jump MCMC - Random move 3: "birth" of a new location

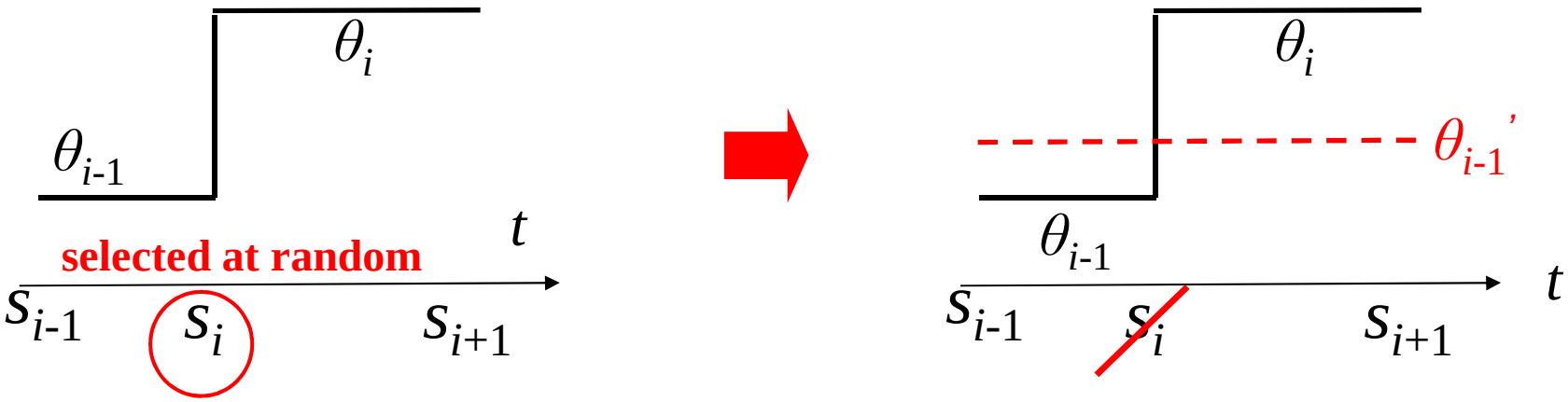


Proposed values: $s^* \approx U[s_0, s_{k+1}]$ **(and identify the interval where it lies)**

$$\begin{cases} (s^* - s_i)\theta_i' + (s_{i+1} - s^*)\theta_{i+1}' = (s_{i+1} - s_i)\theta_i & \text{(perturbation of } \theta_i\text{)} \\ \dfrac{\theta_{i+1}'}{\theta_i'} = \dfrac{1-r}{r}, r \approx U[0,1] & \text{(imposed)} \end{cases}$$

Acceptance probability, α_b: very complex structure! → see (Green, 1995)

Reversible-jump MCMC - Random move 4: "death" of a new location



"Reverse" the "birth" move:

Proposed values: $(s_i - s_{i-1})\theta_{i-1} + (s_{i+1} - s_i)\theta_i = (s_{i+1} - s_{i-1})\theta'_{i-1}$

Acceptance probability, α_d: very complex structure! → see (Green, 1995)

Reversible-jump MCMC: sequential steps of the algorithm (1)

1. **Select prior values of the Markov chain**

$$p(k) = \frac{e^{-\lambda}\lambda^k}{k!}$$

→ **Prior value for *k* (number of changepoints)**

→ **Prior positions of the changepoints: $s_1, s_2, \ldots, s_k$**

$$p(\theta_i) = \frac{\beta^\alpha}{\Gamma(\alpha)}\theta^{\alpha-1}e^{-\beta\theta}$$

→ **Prior values of the parameter: $\theta_0, \theta_1, \ldots, \theta_k$**

2. **Calculate the probabilities for each possible move**

$$\eta_k + \chi_k + b_k + d_k = 1$$ **Normalization**

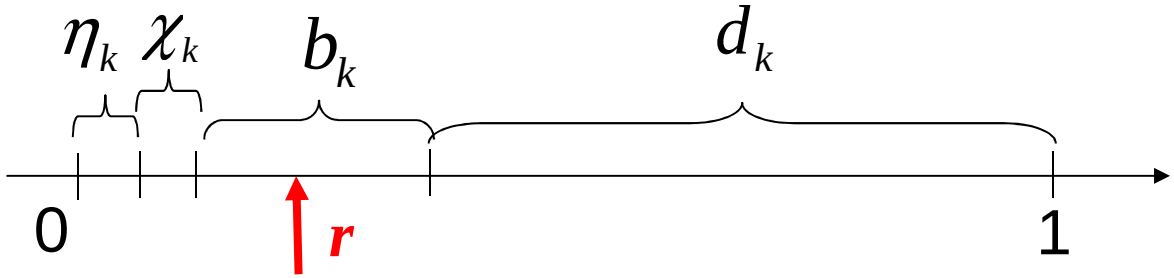
$$\chi_0 = b_{k_{\max}} = d_0 = 0$$ **"Physical" considerations**

$$\left.\begin{array}{l} \eta_k = \chi_k \quad \textbf{if } k \neq 0 \\ b_k = c\min\left\{1, \dfrac{p(k+1)}{p(k)}\right\}, d_k = c\min\left\{1, \dfrac{p(k-1)}{p(k)}\right\} \\ b_k + d_k \leq 0.9 \end{array}\right\}$$

Suggested by (Green, 1995) to obtain "good" chains (i.e., fast convergence and exploration of the parameter space)

Reversible-jump MCMC: sequential steps of the algorithm (2)

3. **Sample a uniform random number *r* in [0, 1) and select one of the possible moves according to the probabilities computed in step 2.**



4. **Generate proposal values according to the selected move**

   If move 1: variation of parameter value (θ_i')

   If move 2: variation of step location (s_i')

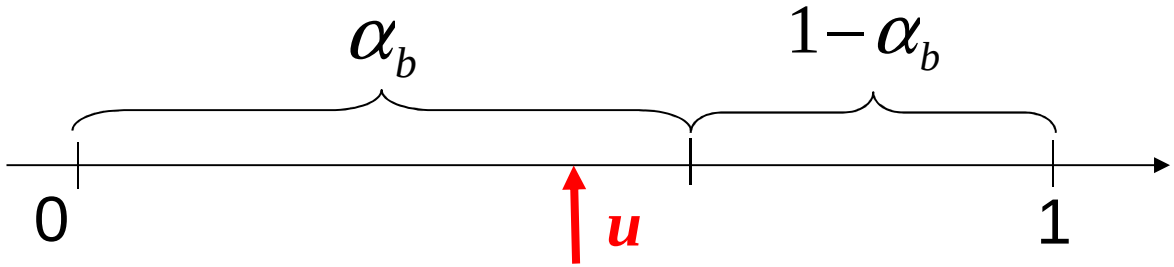
   If move 3: "birth" of a new location (s^*, θ_i' and θ_{i+1}')

   If move 4: "death" of a location (s_i removed and θ_i' proposed)

5. **Calculate the acceptance probability for the move(s) proposed in step 4. (i.e., α_η, α_χ, α_b or α_d)**

Reversible-jump MCMC: sequential steps of the algorithm (3)

5. **Sample a uniform random number *u* in [0, 1) and accept/reject the proposed move(s) according to the acceptance probabilities computed in step 4.**



6. **Update the values of the uncertain parameters in the chain:**
   New number *k* of changepoints,
   New locations s_i, $i = 1, 2, \ldots, k$, of the changepoints,
   New values θ_i, $i = 0, 1, \ldots, k$, of the parameter in each interval

7. **Return to step 2. above (i.e., update the values of the probabilities for each possible move η_k, χ_k, b_k, d_k and so on …)**

8. **Stop the algorithm when the number of iterations (i.e., the number of samples in the chain) reaches a predefined (large) value**

Reversible-jump MCMC: example (1)

- Iteration 0 (i.e., sample the prior values)

$$p(k) = \frac{e^{-\lambda}\lambda^k}{k!}, \lambda = 1, k_{\max} = 3$$ → $k^0 = 2$ (prior number of changepoints)

$$[s_{\min}, s_{\max}] = [0,100] \rightarrow \{61,35,10,43,87\} \rightarrow \{10,35,43,61,87\} \rightarrow \begin{cases} s_1^0 = 35 \\ s_2^0 = 61 \end{cases}$$

(uniform random sampling of $2k^0+1$ values) (Sort)

$$p(\theta_i) = \frac{\beta^\alpha}{\Gamma(\alpha)}\theta^{\alpha-1}e^{-\beta\theta}, \alpha = 3, \beta = 2 \rightarrow \{\theta_1^0, \theta_2^0, \theta_3^0\} = \begin{cases} 4.04, \text{if } s \in [s_{\min}, s_1^0] \\ 1.51, \text{ if } s \in [s_1^0, s_2^0] \\ 5.75, \text{ if } s \in [s_2^0, s_{\max}] \end{cases}$$

	k^0	s_1^0	s_2^0	θ_1^0	θ_2^0	θ_3^0
Sample 0	2	35	61	4.04	1.51	5.75

Reversible-jump MCMC: example (2)

- Iteration 1:
  - Calculate the probabilities for each possible move given $k = 2$

$$\eta_k + \chi_k + b_k + d_k = 1$$

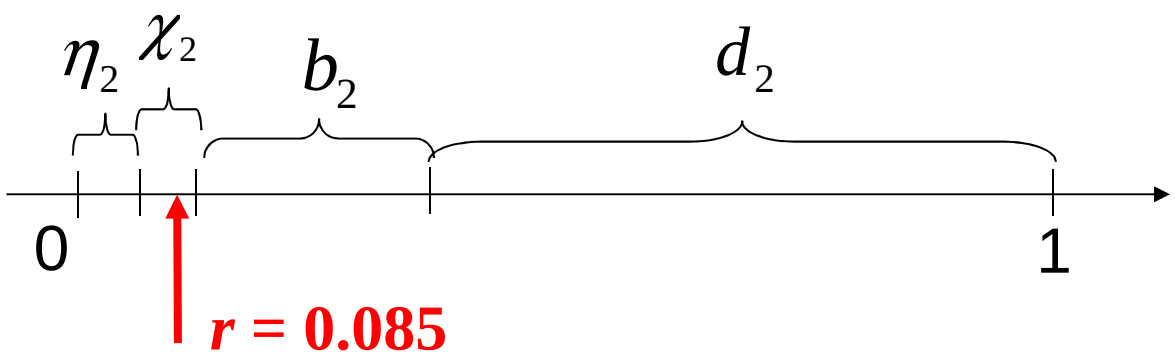
$$b_k = c\min\left\{1, \frac{p(k+1)}{p(k)}\right\}, d_k = c\min\left\{1, \frac{p(k-1)}{p(k)}\right\}$$

$$b_k + d_k \le 0.9$$

$$\eta_k = \chi_k$$

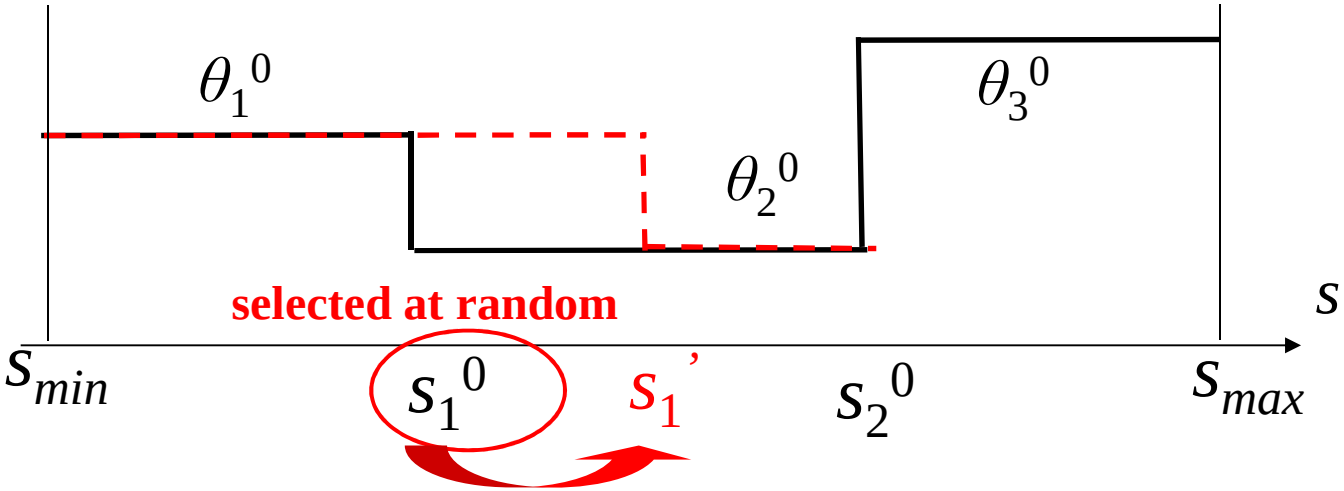
$$\rightarrow \begin{cases} \eta_2 = 0.0500 \\ \chi_2 = 0.0500 \\ b_2 = 0.2250 \\ d_2 = 0.6750 \end{cases}$$

  - Sample one of the possible moves



VARIATION OF A STEP LOCATION

Reversible-jump MCMC: example (3)



- Select randomly one of the changepoints $\rightarrow s_1^0 = 35$

- Sample a new proposal value s_1' from a uniform distribution:

$$U\left[s_{\min}, s_2^0\right) = U[0, 61) \rightarrow s_1^{'} = 51$$

- Accept/reject the candidate according to α_χ → e.g., accept: $s_1^1 = s_1$'

	k^1	s_1^1	s_2^1	θ_1^1	θ_2^1	θ_3^1
Sample 1	2	**51**	61	4.04	1.51	5.75

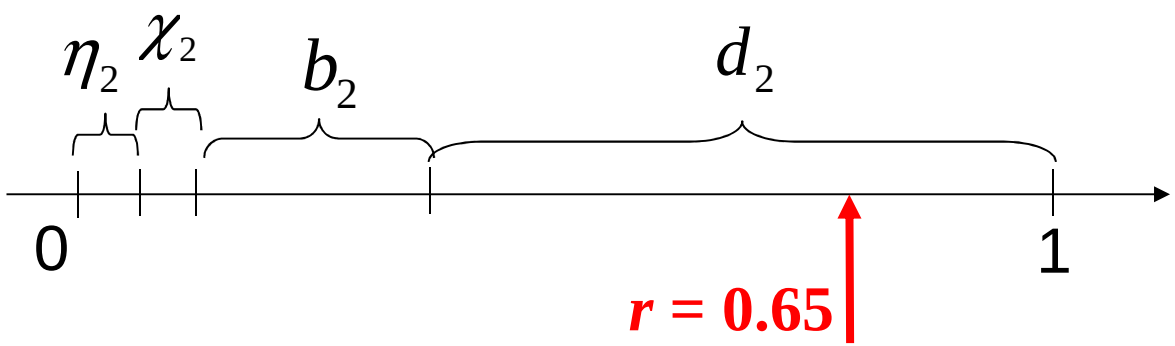
Reversible-jump MCMC: example (4)

- Iteration 2:

  - Calculate the probabilities for each possible move: since k still = 2 the probabilities are the same as before!

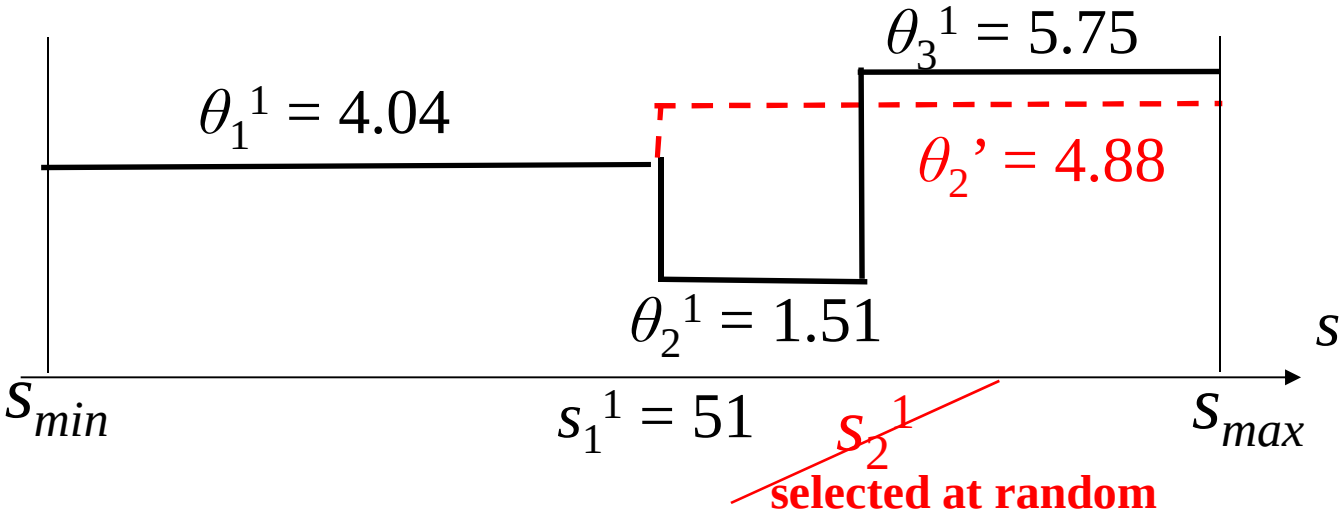
$$\rightarrow \begin{cases} \eta_k = \eta_2 = 0.0500 \\ \chi_k = \chi_2 = 0.0500 \\ b_k = b_2 = 0.2250 \\ d_k = d_2 = 0.6750 \end{cases}$$

  - Sample one of the possible moves



"DEATH" OF A LOCATION

Reversible-jump MCMC: example (5)



- Select randomly one of the changepoints $\rightarrow s_2^1 = 61$

- Proposed value: $\left(s_2^1 - s_1^1\right)\theta_2^1 + \left(s_{\max} - s_2^1\right)\theta_3^1 = \left(s_{\max} - s_1^1\right)\theta_2^{'} \rightarrow \theta_2^{'} = 4.88$

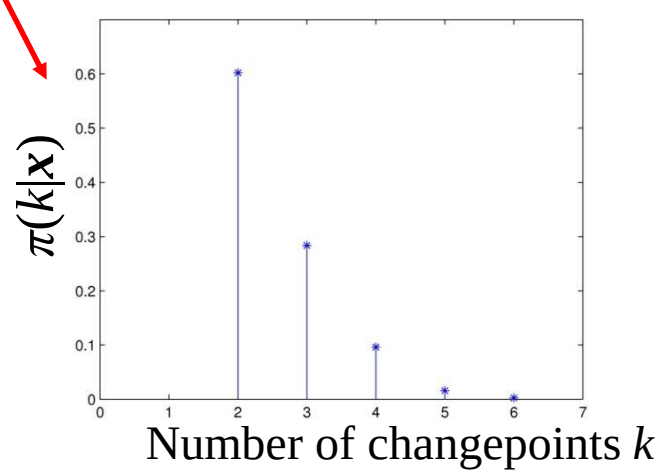
- Accept/reject the candidate according to α_d → e.g., accept: $\theta_2^2 = \theta_2'$

	k^2	s_1^2	s_2^2	θ_1^2	θ_2^2	θ_3^2
Sample 2	**1**	51	/	4.04	**4.88**	/

Reversible-jump MCMC: example (6)

	k	s_1	s_2	s_3	θ_1	θ_2	θ_3	θ_4
Sample 0	2	35	61		4.04	1.51	5.75	
Sample 1	2	51	61		4.04	1.51	5.75	
Sample 2	1	51			4.04	4.88		
		…				…		
Sample u	3	s_1^u	s_2^u	s_3^u	θ_1^u	θ_2^u	θ_3^u	θ_4^u
Sample $u+1$	3	s_1^{u+1}	s_2^{u+1}	s_3^{u+1}	θ_1^{u+1}	θ_2^{u+1}	θ_3^{u+1}	θ_4^{u+1}
Sample Ns	1	s_1^{Ns}			θ_1^{Ns}	θ_2^{Ns}		

Posterior distribution of the number of changepoints



Reversible-jump MCMC: example (7)

	k	s_1	s_2	s_3	θ_1	θ_2	θ_3	θ_4
Sample 0	2	35	61		4.04	1.51	5.75	
Sample 1	2	51	61		4.04	1.51	5.75	
Sample 2	1	51			4.04	4.88		
		...				...		
Sample u	3	s_1^u	s_2^u	s_3^u	θ_1^u	θ_2^u	θ_3^u	θ_4^u
Sample $u+1$	3	s_1^{u+1}	s_2^{u+1}	s_3^{u+1}	θ_1^{u+1}	θ_2^{u+1}	θ_3^{u+1}	θ_4^{u+1}
Sample Ns	1	s_1^{Ns}			θ_1^{Ns}	θ_2^{Ns}		

$$\pi(s \mid \mathbf{x}, k=2)$$

$$\pi(s \mid \mathbf{x}, k=1)$$

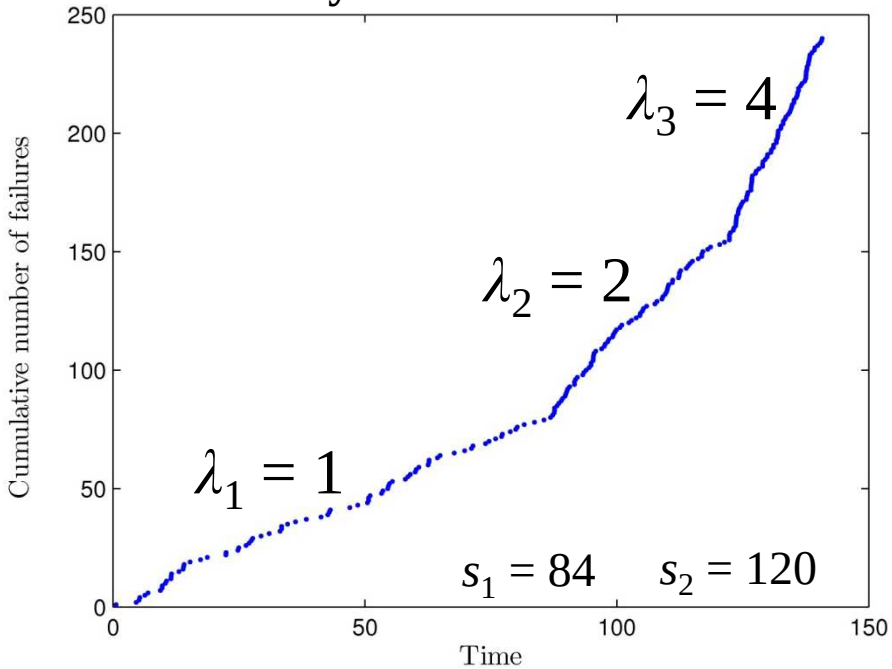
$$\pi(s \mid \mathbf{x}, k=3)$$

$$\pi(s \mid \mathbf{x}) = \sum_k \pi(s \mid \mathbf{x}, k)\pi(k \mid \mathbf{x})$$

Application 1: Component degradation due to reparations or aging

Application 1: inhomogeneous Poisson process

Synthetic dataset



Inhomogeneous Poisson process: $\lambda = \lambda(t)$

Experimental dataset:

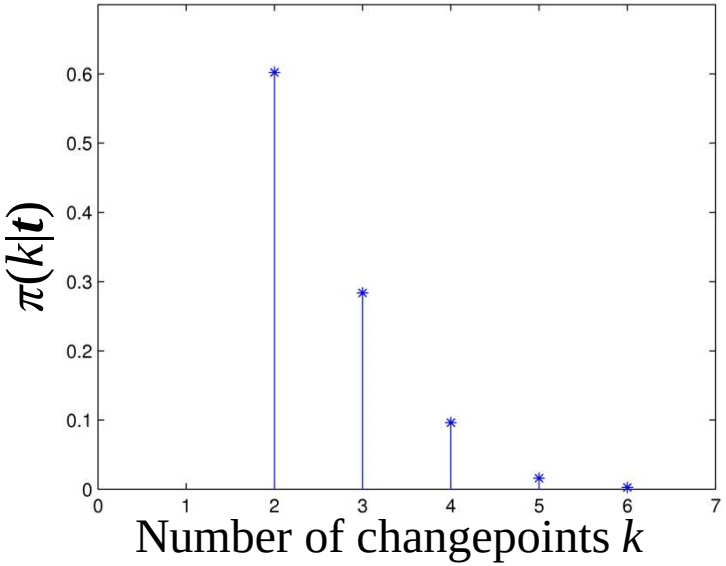
$\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$

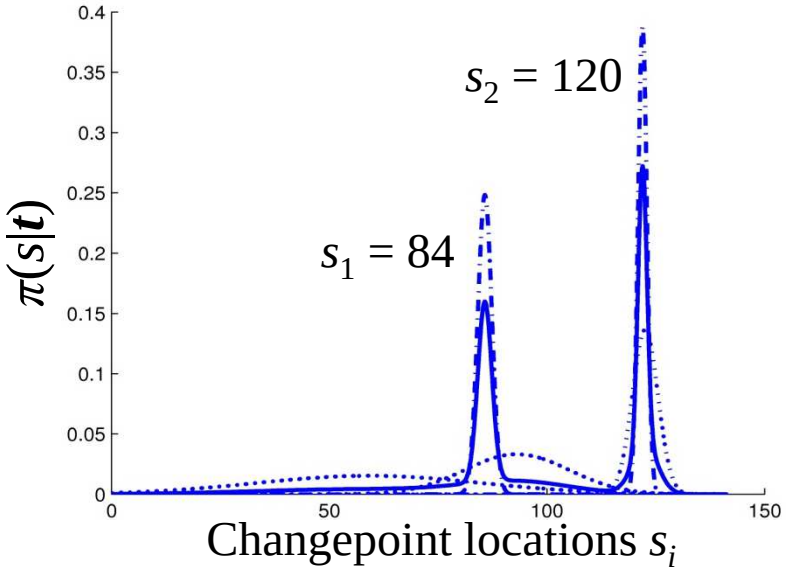
$\boldsymbol{t} = \{t_1, t_2, \ldots, t_N\}$ (N = 240 failure times)

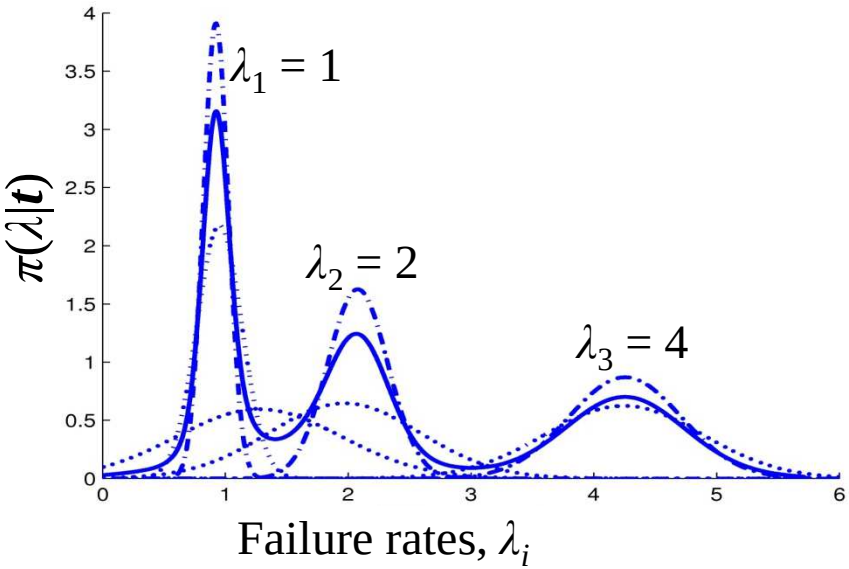
Likelihood function

$$L(\theta \mid \boldsymbol{x}) = L(\lambda \mid \boldsymbol{t}) = \prod_{i=1}^{N} \lambda(t_i) \exp\left[-\int_{s_0=0}^{s_{k+1}=T} \lambda(t)dt \right]$$

Application 1 – Results: posterior distributions



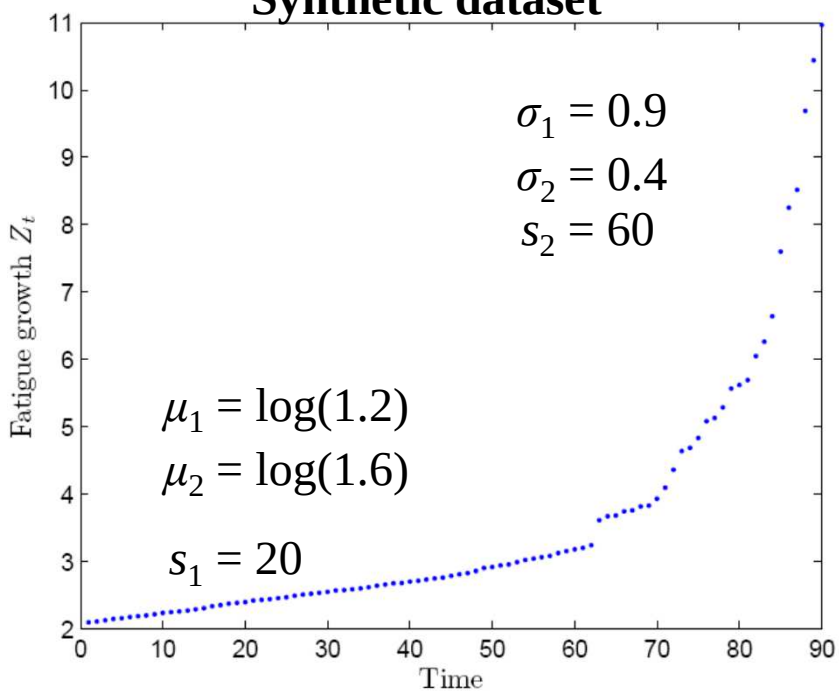




Application 2: Deterioration due to fatigue

Application: crack growth due to fatigue

Synthetic dataset



Paris-Erdogan law:

$$\frac{dZ_t}{dt} = a_0 Z_t^b X_t$$

a_0, b = constant

Z_t = crack size

X_t = multiplicative noise ~ $\exp[N(\mu, \sigma)]$

Experimental dataset:

$\boldsymbol{x} = \{x_1, x_2, \ldots, x_N\}$

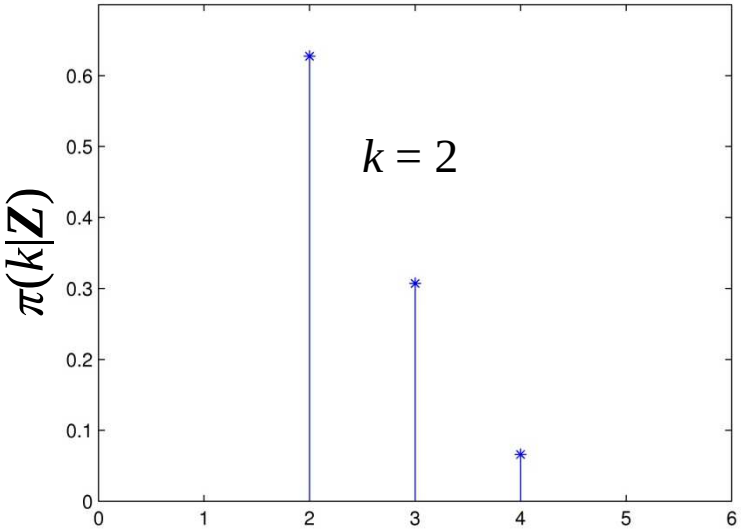
$\boldsymbol{Z} = \{Z_1, Z_2, \ldots, Z_N\}$ (N = 90 observations)

Likelihood function

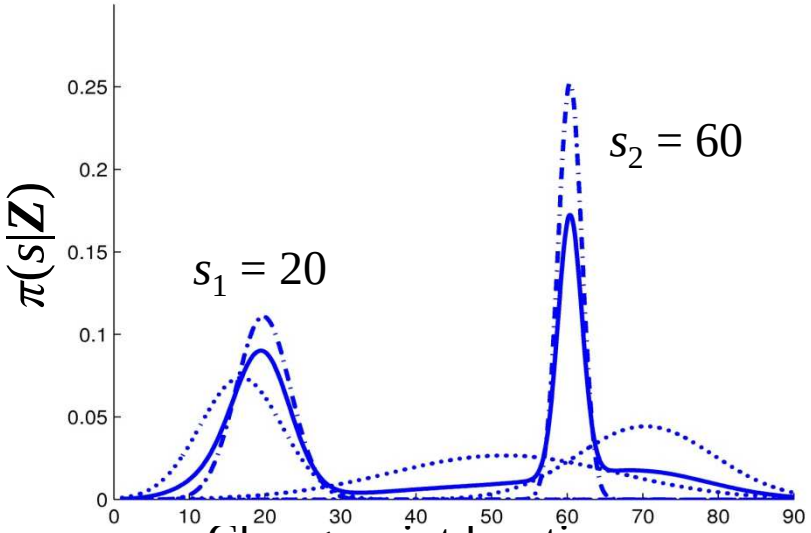
Setting $Y_t(X_t) = \log\left(\frac{1}{a_0}\frac{dZ_t}{dt} Z_t^{-b}\right) = \log(X_t) \approx N(\mu, \sigma)$ then

$$L(Y_t(Z_t) \mid \mu, \sigma) \propto \frac{1}{\sigma^N} \exp\left[-\sum_{i=1}^{N} \frac{(Y_i - \mu)^2}{2\sigma^2}\right]$$

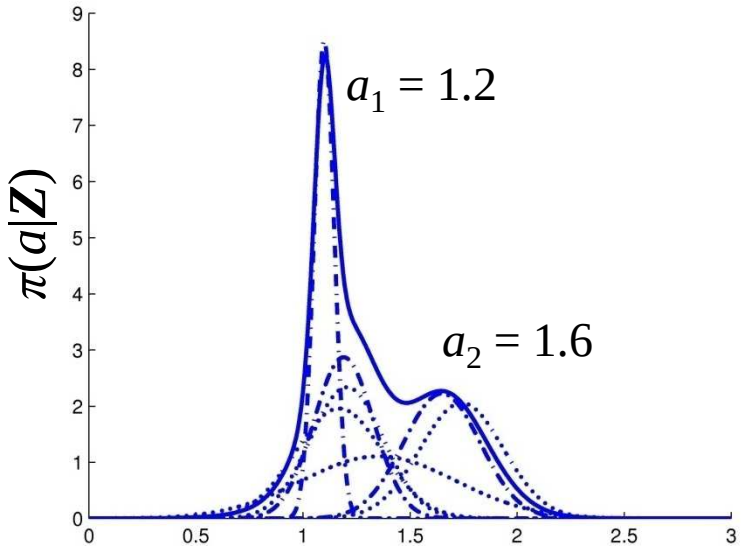
Application 2 – Results: posterior distributions



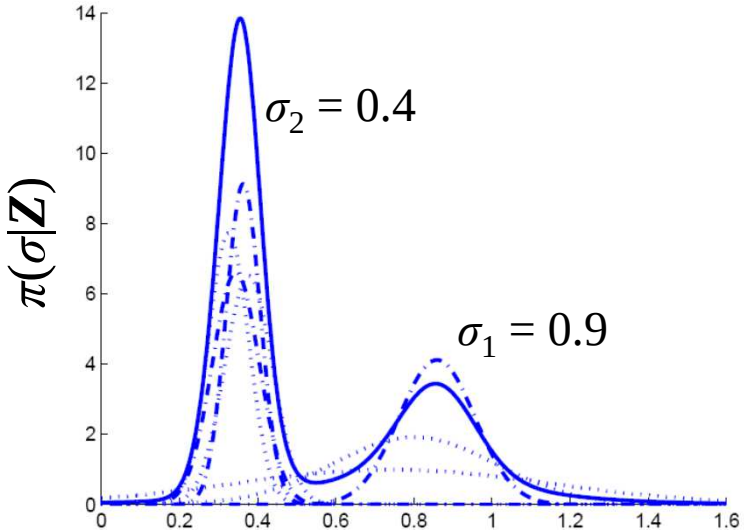
Number of changepoints k



Changepoint locations s_i



Paramater $a_i = exp(\mu_i)$



Standard deviation, σ_i