MCMC and ABC Methodologies in the context of Controlled Branching Processes

M. González, I. del Puerto

Department of Mathematics. University of Extremadura
Spanish Branching Processes Group

Workshop Métodos Bayesianos 11
Madrid, November 2011

Contents

Contents

Contents

Contents

Contents

Inside the general context concerning Stochastic Models, Branching Processes Theory provides appropriate mathematical models for description of the probabilistic evolution of systems whose components (cell, particles, individuals in general), after certain life period, reproduce and die. Therefore, it can be applied in several fields (Biology, Demography, Ecology, Epidemiology, Genetics, Algorithms,...).

Example

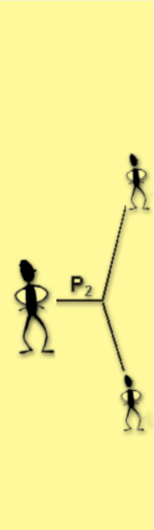
$Z_0 = 1$

$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$

Example

$Z_0 = 1$



$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$

Branching Processes

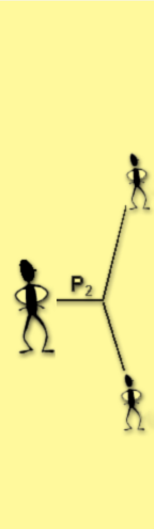
Example

$Z_0 = 1$

$Z_1 = 2$

$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$



Branching Processes

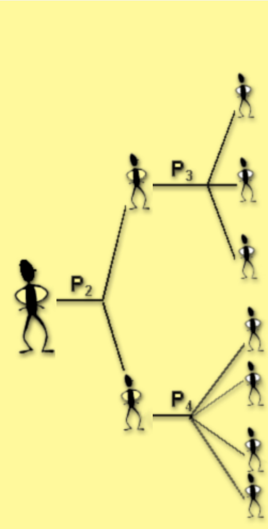
Example

$Z_0 = 1$

$Z_1 = 2$

$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$



Example

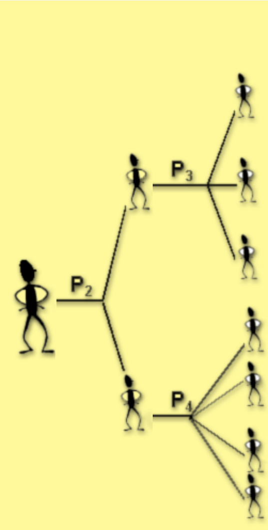
$Z_0 = 1$

$Z_1 = 2$

$Z_2 = 7$

$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$



Branching Processes

Example

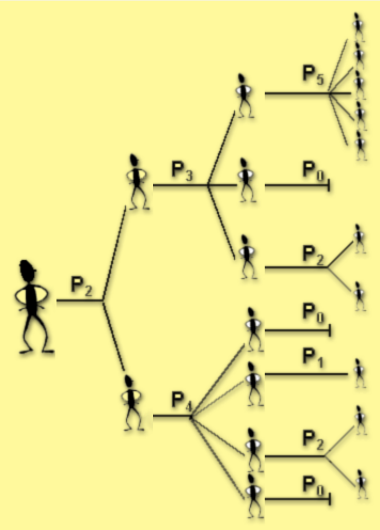
$Z_0 = 1$

$Z_1 = 2$

$Z_2 = 7$

$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$



Branching Processes

Example

$Z_0 = 1$

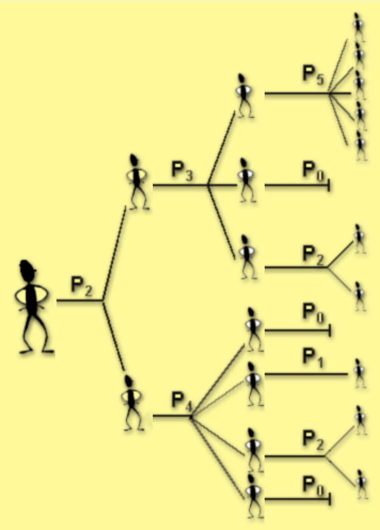
$Z_1 = 2$

$Z_2 = 7$

$Z_3 = 10$

$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$



Branching Processes

Example

$Z_0 = 1$

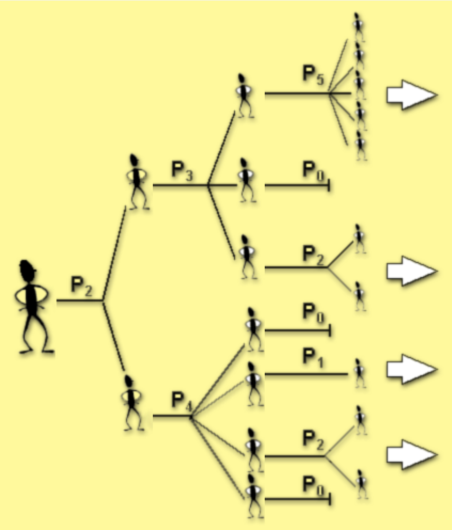
$Z_1 = 2$

$Z_2 = 7$

$Z_3 = 10$

$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$



Branching Processes

Example

$Z_0 = 1$

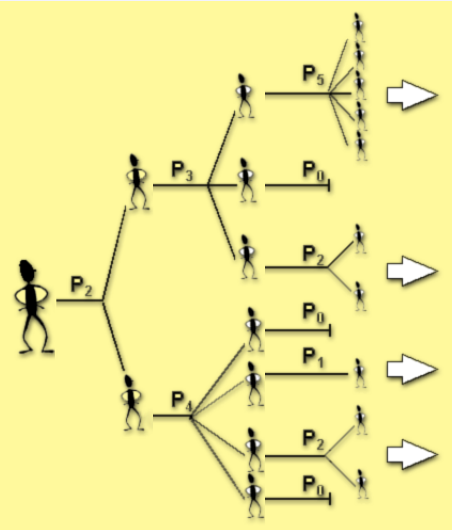
$Z_1 = 2$

$Z_2 = 7$

$Z_3 = 10$

$\vdots$

$$Z_{n+1} = \sum_{j=1}^{Z_n} X_{nj}$$



Main Results for Galton–Watson Branching Processes

Let $m = E[X_{01}]$ and $\sigma^2 = Var[X_{01}]$

- Extinction Problem
  - If $m \leq 1 \Rightarrow$ the process dies out with probability 1
  - If $m > 1 \Rightarrow$ there exists a positive probability of non-extinction
- Asymptotic behaviour
- Statistical Inference

Main Results for Galton–Watson Branching Processes

Let $m = E[X_{01}]$ and $\sigma^2 = Var[X_{01}]$

- Extinction Problem
  - If $m \leq 1 \Rightarrow$ the process dies out with probability 1
  - If $m > 1 \Rightarrow$ there exists a positive probability of non-extinction
- Asymptotic behaviour
- Statistical Inference

Main Results for Galton–Watson Branching Processes

Let $m = E[X_{01}]$ and $\sigma^2 = Var[X_{01}]$

- Extinction Problem
  - If $m \leq 1 \Rightarrow$ the process dies out with probability 1
  - If $m > 1 \Rightarrow$ there exists a positive probability of non-extinction
- Asymptotic behaviour
- Statistical Inference

Main Results for Galton–Watson Branching Processes

Let $m = E[X_{01}]$ and $\sigma^2 = Var[X_{01}]$

- Extinction Problem
  - If $m \leq 1 \Rightarrow$ the process dies out with probability 1
  - If $m > 1 \Rightarrow$ there exists a positive probability of non-extinction
- Asymptotic behaviour
- Statistical Inference

Branching Processes

Many monographs about the theory and applications about the branching processes have been published:

- Harris, T. (1963). The Theory of branching processes. Springer-Verlag.
- Jagers, P. (1975). Branching processes with Biological Applications, John Wiley and Sons, Inc.
- Asmussen, S. and Hering, H. (1983). Branching processes. Birkhäuser. Boston.
- Athreya, K.B. and Jagers, P. (1997). Classical and modern branching processes. Springer-Verlag.
- Kimmel, M. and Axelrod, D.E. (2002). Branching processes in Biology, Springer-Verlag New York, Inc.
- Haccou, P., Jagers, P., and Vatutin, V. (2005). Branching Processes: Variation, Growth, and Extinction of Populations. Cambridge University Press.
- González, M., del Puerto, I., Martínez, R., Molina, M., Mota, M., Ramos, A. (Editors) (2010). Workshop on Branching Processes and their Applications. Lecture Notes in Statistics, 197. Springer.

A Controlled Branching Process is a discrete-time stochastic growth population model in which the individuals with reproductive capacity in each generation are controlled by some function ϕ. This branching model is well-suited for describing the probabilistic evolution of populations in which, for various reasons of an environmental, social or other nature, there is a mechanism that establishes the number of progenitors who take part in each generation.

Mathematically: Controlled Branching Process $\{Z_n\}_{n\geq 0}$

$$Z_0 = N, \quad Z_{n+1} = \sum_{i=1}^{\phi_n(Z_n)} X_{ni}, \quad n = 0, 1, \ldots$$

Two independent sequences of random variables (r.v.):

- $\{X_{ni} : i = 1, 2, \ldots, n = 0, 1, \ldots\}$ are i.i.d. r.v.
  $p = \{p_k : k = 0, 1, \ldots\}$ Offspring Distribution
  $m = E[X_{01}], \sigma^2 = Var[X_{01}]$
- $\{\phi_n(k) : n = 0, 1, \ldots; \ k = 0, 1, \ldots\}$, where $\{\phi_n(k)\}_{k\geq 0}$ are independent stochastic processes with identical one-dimensional probability distributions, $n = 0, 1, \ldots$ Random Control Functions
  $\varepsilon(k) = E[\phi_n(k)], \sigma^2(k) = Var[\phi_n(k)]$.
- $\phi_n(k) = \phi(k), k = 0, 1, \ldots$ Deterministic Control Function

Branching Processes

Mathematically: Controlled Branching Process $\{Z_n\}_{n\geq 0}$

$$Z_0 = N, \quad Z_{n+1} = \sum_{i=1}^{\phi_n(Z_n)} X_{ni}, \quad n = 0, 1, \ldots$$

Two independent sequences of random variables (r.v.):

- $\{X_{ni} : i = 1, 2, \ldots, n = 0, 1, \ldots\}$ are i.i.d. r.v.
  $p = \{p_k : k = 0, 1, \ldots\}$ Offspring Distribution
  $m = E[X_{01}], \sigma^2 = Var[X_{01}]$
- $\{\phi_n(k) : n = 0, 1, \ldots; \; k = 0, 1, \ldots\}$, where $\{\phi_n(k)\}_{k\geq 0}$ are independent stochastic processes with identical one-dimensional probability distributions, $n = 0, 1, \ldots$ Random Control Functions
  $\varepsilon(k) = E[\phi_n(k)], \sigma^2(k) = Var[\phi_n(k)]$.
- $\phi_n(k) = \phi(k), k = 0, 1, \ldots$ Deterministic Control Function

Controlled Branching Processes

Properties

- $\{Z_n\}_{n\geq 0}$ is a Homogeneous Markov Chain
- Duality Extinction-Explosion: $P(Z_n \to 0) + P(Z_n \to \infty) = 1$

Main Topics Investigated

- Extinction Problem
  - Sevast'yanov and Zubkov (1974)
  - Zubkov (1974)
  - Molina, González and Mota (1998)
- Asymptotic Behaviour: Growth rates
  - Bagley (1986)
  - Molina, González and Mota (1998)
  - González, Molina, del Puerto (2002, 2003, 2004, 2005a,b)

Controlled Branching Processes

Properties

- $\{Z_n\}_{n\geq 0}$ is a Homogeneous Markov Chain
- Duality Extinction-Explosion: $P(Z_n \to 0) + P(Z_n \to \infty) = 1$

Main Topics Investigated

- Extinction Problem
  - Sevast'yanov and Zubkov (1974)
  - Zubkov (1974)
  - Molina, González and Mota (1998)
- Asymptotic Behaviour: Growth rates
  - Bagley (1986)
  - Molina, González and Mota (1998)
  - González, Molina, del Puerto (2002, 2003, 2004, 2005a,b)

Controlled Branching Processes

Main Topics Investigated

Statistical Inference

- Dion, J. P. and Essebbar, B. (1995). On the statistics of controlled branching processes. Lecture Notes in Statistics, 99:14-21.
- M. González, R. Martínez, I. Del Puerto (2004). Nonparametric estimation of the offspring distribution and the mean for a controlled branching process. Test, 13(2), 465-479.
- M. González, R. Martínez, I. Del Puerto (2005). Estimation of the variance for a controlled branching process. Test, 14(1), 199-213.
- T.N. Sriram, A. Bhattacharya, M. González, R. Martínez, I. Del Puerto (2007). Estimation of the offspring mean in a controlled branching process with a random control function. Stochastic Processes and their Applications, 117, 928-946.
- R. Martínez, I. del Puerto, M. Mota (2009). On asymptotic posterior normality for controlled branching processes. Statistics, 43, 367-378.

Non-Parametric Framework

Offspring Distribution: $p = \{p_k : k \in \mathcal{S}\}$ $\mathcal{S}$ finite.

Deterministic Control Function: $\phi(\cdot)$

Sample: The entire family tree up to the current generation

$$\{X_{ki} : i = 1, \ldots, \phi(Z_k), k = 0, 1, \ldots, n\}$$

or at least

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$$

where $Z_j(k) = \sum_{i=1}^{\phi(Z_j)} I_{\{X_{ji}=k\}}$ = number of parents in the jth-generation which generate exactly k offspring

Objective: Make inference on p

Non-Parametric Framework

Offspring Distribution: $p = \{p_k : k \in \mathcal{S}\}$ $\quad$ $\mathcal{S}$ finite.

Deterministic Control Function: $\phi(\cdot)$

Sample: The entire family tree up to the current generation

$$\{X_{ki} : i = 1, \ldots, \phi(Z_k), k = 0, 1, \ldots, n\}$$

or at least

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$$

where $Z_j(k) = \sum_{i=1}^{\phi(Z_j)} I_{\{X_{ji}=k\}}$ = number of parents in the jth-generation which generate exactly k offspring

Objective: Make inference on p

Non-Parametric Framework

Offspring Distribution: $p = \{p_k : k \in \mathcal{S}\}$ $\quad$ $\mathcal{S}$ finite.

Deterministic Control Function: $\phi(\cdot)$

Sample: The entire family tree up to the current generation

$$\{X_{ki} : i = 1, \ldots, \phi(Z_k), k = 0, 1, \ldots, n\}$$

or at least

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$$

where $Z_j(k) = \sum_{i=1}^{\phi(Z_j)} I_{\{X_{ji}=k\}}$ = number of parents in the jth-generation which generate exactly k offspring

Objective: Make inference on p

Non-Parametric Framework

Offspring Distribution: $p = \{p_k : k \in \mathcal{S}\}$ $\mathcal{S}$ finite.

Deterministic Control Function: $\phi(\cdot)$

Sample: The entire family tree up to the current generation

$$\{X_{ki} : i = 1, \ldots, \phi(Z_k), k = 0, 1, \ldots, n\}$$

or at least

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$$

where $Z_j(k) = \sum_{i=1}^{\phi(Z_j)} I_{\{X_{ji}=k\}}$ = number of parents in the jth-generation which generate exactly k offspring

Objective: Make inference on p

Likelihood Function

$$f(\mathcal{Z}_n|p) \propto \prod_{k\in\mathcal{S}} p_k^{\sum_{j=0}^{n} Z_j(k)}$$

Conjugate Class of Distributions: Dirichlet Family

- Prior Distribution: $p \sim D(\alpha_k : k \in \mathcal{S})$
- Posterior Distribution:

$$p|\mathcal{Z}_n \sim D(\alpha_k + \sum_{j=0}^{n} Z_j(k) : k \in \mathcal{S})$$

Bayesian Inference for Controlled Branching Processes

Likelihood Function

$$f(\mathcal{Z}_n|p) \propto \prod_{k\in\mathcal{S}} p_k^{\sum_{j=0}^{n} Z_j(k)}$$

Conjugate Class of Distributions: Dirichlet Family

- Prior Distribution: $p \sim D(\alpha_k : k \in \mathcal{S})$
- Posterior Distribution:

$$p|\mathcal{Z}_n \sim D(\alpha_k + \sum_{j=0}^{n} Z_j(k) : k \in \mathcal{S})$$

Setting out the Problem

In real problems it is difficult to observe the entire family tree $\{X_{ki}: i = 1, 2, \ldots, k = 0, 1, \ldots, n\}$ or even the random variables $\mathcal{Z}_n = \{Z_j(k): k \in \mathcal{S}, j = 0, \ldots, n\}$

Usual Sample Information

- $\mathcal{Z}_n^* = \{Z_j : j = 0, \ldots, n\}$

Solution

We introduce an algorithm to approximate the distribution

$$p|\mathcal{Z}_n^*$$

using Markov Chain Monte Carlo Methods

Setting out the Problem

In real problems it is difficult to observe the entire family tree $\{X_{ki} : i = 1, 2, \dots, k = 0, 1, \dots, n\}$ or even the random variables $\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \dots, n\}$

Usual Sample Information

- $\mathcal{Z}_n^* = \{Z_j : j = 0, \dots, n\}$

Solution

We introduce an algorithm to approximate the distribution

$$p|\mathcal{Z}_n^*$$

using Markov Chain Monte Carlo Methods

Setting out the Problem

In real problems it is difficult to observe the entire family tree $\{X_{ki} : i = 1, 2, \ldots, k = 0, 1, \ldots, n\}$ or even the random variables $\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$

Usual Sample Information

- $\mathcal{Z}_n^* = \{Z_j : j = 0, \ldots, n\}$

Solution

We introduce an algorithm to approximate the distribution

$$p|\mathcal{Z}_n^*$$

using Markov Chain Monte Carlo Methods

Gibbs Sampler: Introducing the Method

- Sample: $\mathcal{Z}_n^* = \{Z_j : j = 0, \ldots, n\}$

The Problem

$$p|\mathcal{Z}_n^*$$

- Latent Variables:

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$$

- Gibbs Sampler:

$$p|\mathcal{Z}_n, \mathcal{Z}_n^* \qquad \mathcal{Z}_n|\mathcal{Z}_n^*, p$$

Gibbs Sampler: Introducing the Method

- Sample: $\mathcal{Z}_n^* = \{Z_j : j = 0, \ldots, n\}$

The Problem

$$p|\mathcal{Z}_n^*$$

- Latent Variables:

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$$

- Gibbs Sampler:

$$p|\mathcal{Z}_n, \mathcal{Z}_n^* \qquad \mathcal{Z}_n|\mathcal{Z}_n^*, p$$

- Sample: $\mathcal{Z}_n^* = \{Z_j : j = 0, \dots, n\}$

The Problem

$$p|\mathcal{Z}_n^*$$

- Latent Variables:

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \dots, n\}$$

- Gibbs Sampler:

$$p|\mathcal{Z}_n, \mathcal{Z}_n^* \qquad \mathcal{Z}_n|\mathcal{Z}_n^*, p$$

Gibbs Sampler: Introducing the Method

- Sample: $\mathcal{Z}_n^* = \{Z_j : j = 0, \ldots, n\}$

The Problem

$$p \mid \mathcal{Z}_n^*$$

- Latent Variables:

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$$

- Gibbs Sampler:

$$p \mid \mathcal{Z}_n, \mathcal{Z}_n^* \qquad \mathcal{Z}_n \mid \mathcal{Z}_n^*, p$$

Gibbs Sampler: Introducing the Method

First Conditional Distribution: $p|\mathcal{Z}_n, \mathcal{Z}_n^*$

$$p|\mathcal{Z}_n, \mathcal{Z}_n^* \equiv p|\mathcal{Z}_n \sim D(\alpha_k + \sum_{j=0}^{n} Z_j(k) : k \in \mathcal{S})$$

- For $j = 0, \ldots, n$

$$\phi(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k)$$

$$Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

First Conditional Distribution: $p|\mathcal{Z}_n, \mathcal{Z}_n^*$

$$p|\mathcal{Z}_n, \mathcal{Z}_n^* \equiv p|\mathcal{Z}_n \sim D(\alpha_k + \sum_{j=0}^{n} Z_j(k) : k \in \mathcal{S})$$

- For $j = 0, \ldots, n$

$$\phi(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k)$$

$$Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

First Conditional Distribution: $p|\mathcal{Z}_n, \mathcal{Z}_n^*$

$$p|\mathcal{Z}_n, \mathcal{Z}_n^* \equiv p|\mathcal{Z}_n \sim D(\alpha_k + \sum_{j=0}^{n} Z_j(k) : k \in \mathcal{S})$$

- For $j = 0, \ldots, n$

$$\phi(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k)$$

$$Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n|\mathcal{Z}_n^*, p$

$$f(\mathcal{Z}_n|\mathcal{Z}_n^*, p) = \prod_{j=0}^{n} f(Z_j(k) : k \in \mathcal{S}|Z_j, Z_{j+1}, p)$$

$$(Z_j(k) : k \in \mathcal{S})|Z_j, Z_{j+1}, p$$

is obtained from a

$$Multinomial(\phi(Z_j), p)$$

normalized by considering the constraint

$$Z_{j+1} = \sum_{k \in \mathcal{S}} kZ_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p$

$$f(\mathcal{Z}_n|\mathcal{Z}_n^*, p) = \prod_{j=0}^{n} f(Z_j(k) : k \in \mathcal{S} | Z_j, Z_{j+1}, p)$$

$$(Z_j(k) : k \in \mathcal{S}) | Z_j, Z_{j+1}, p$$

is obtained from a

$$Multinomial(\phi(Z_j), p)$$

normalized by considering the constraint

$$Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p$

p

$\phi(Z_0)$

$Z_0(k), k \in \mathcal{S}$

Z_1 $\phi(Z_1)$

$Z_1(k), k \in \mathcal{S}$

Z_2 $\phi(Z_2)$

$\vdots$ $\vdots$ $\vdots$

Z_n $\phi(Z_n)$

$Z_n(k), k \in \mathcal{S}$

Z_{n+1}

$$\phi(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p$

$$p$$

$$\phi(Z_0)$$

$$Z_0(k), k \in \mathcal{S}$$

$$Z_1 \qquad \phi(Z_1)$$

$$Z_1(k), k \in \mathcal{S}$$

$$Z_2 \qquad \phi(Z_2)$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$Z_n \qquad \phi(Z_n)$$

$$Z_n(k), k \in \mathcal{S}$$

$$Z_{n+1}$$

$$\phi(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p$

p

$\phi(Z_0)$

$Z_0(k), k \in \mathcal{S}$

Z_1 $\phi(Z_1)$

$Z_1(k), k \in \mathcal{S}$

Z_2 $\phi(Z_2)$

$\vdots$ $\vdots$ $\vdots$

Z_n $\phi(Z_n)$

$Z_n(k), k \in \mathcal{S}$

Z_{n+1}

$$\phi(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p$

$$
\begin{array}{ccc}
 & p & \\
 & & \phi(Z_0) \\
 & Z_0(k), k \in \mathcal{S} & \\
Z_1 & & \phi(Z_1) \\
 & Z_1(k), k \in \mathcal{S} & \\
Z_2 & & \phi(Z_2) \\
\vdots & \vdots & \vdots \\
Z_n & & \phi(Z_n) \\
 & Z_n(k), k \in \mathcal{S} & \\
Z_{n+1} & &
\end{array}
$$

$$\phi(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Developing the Method

Algorithm

```
Fixed p^(0)
Do l = 1
   Generate Z_n^(l) ~ Z_n | Z_n^*, p^(l-1)
   Generate p^(l) ~ p | Z_n^(l)
Do l = l + 1
```

Fixed $p^{(0)}$
Do $l = 1$
  Generate $\mathcal{Z}_n^{(l)} \sim \mathcal{Z}_n | \mathcal{Z}_n^*, p^{(l-1)}$
  Generate $p^{(l)} \sim p | \mathcal{Z}_n^{(l)}$
Do $l = l + 1$

- For a run of the sequence $\{p^{(l)}\}_{l \geq 0}$, we choose $Q + 1$ vectors in the way $\{p^{(N)}, p^{(N+G)}, \dots, p^{(N+QG)}\}$, where N is the burn-in period and G is a batch size.
- The vectors $\{p^{(N)}, p^{(N+G)}, \dots, p^{(N+QG)}\}$ are considered independent samples from $p | \mathcal{Z}_n^*$ if G and N are large enough (Tierney (1994)).
- Since these vectors could be affected by the initial state $p^{(0)}$, we apply the algorithm T times, obtaining a final sample of length $T(Q + 1)$.

Gibbs Sampler: Developing the Method

Algorithm

```
Fixed p^(0)
Do l = 1
   Generate Z_n^(l) ~ Z_n | Z_n^*, p^(l-1)
   Generate p^(l) ~ p | Z_n^(l)
Do l = l + 1
```

Fixed $p^{(0)}$

Do $l = 1$

Generate $\mathcal{Z}_n^{(l)} \sim \mathcal{Z}_n | \mathcal{Z}_n^*, p^{(l-1)}$

Generate $p^{(l)} \sim p | \mathcal{Z}_n^{(l)}$

Do $l = l + 1$

- For a run of the sequence $\{p^{(l)}\}_{l \geq 0}$, we choose $Q+1$ vectors in the way $\{p^{(N)}, p^{(N+G)}, \ldots, p^{(N+QG)}\}$, where N is the burn-in period and G is a batch size.
- The vectors $\{p^{(N)}, p^{(N+G)}, \ldots, p^{(N+QG)}\}$ are considered independent samples from $p|\mathcal{Z}_n^*$ if G and N are large enough (Tierney (1994)).
- Since these vectors could be affected by the initial state $p^{(0)}$, we apply the algorithm T times, obtaining a final sample of length $T(Q+1)$.

Gibbs Sampler: Developing the Method

Algorithm

```
Fixed p^(0)
Do l = 1
   Generate Z_n^(l) ~ Z_n | Z_n^*, p^(l-1)
   Generate p^(l) ~ p | Z_n^(l)
Do l = l + 1
```

Fixed $p^{(0)}$
Do $l = 1$
  Generate $\mathcal{Z}_n^{(l)} \sim \mathcal{Z}_n | \mathcal{Z}_n^*, p^{(l-1)}$
  Generate $p^{(l)} \sim p | \mathcal{Z}_n^{(l)}$
Do $l = l + 1$

- For a run of the sequence $\{p^{(l)}\}_{l \geq 0}$, we choose $Q + 1$ vectors in the way $\{p^{(N)}, p^{(N+G)}, \ldots, p^{(N+QG)}\}$, where N is the burn-in period and G is a batch size.
- The vectors $\{p^{(N)}, p^{(N+G)}, \ldots, p^{(N+QG)}\}$ are considered independent samples from $p|\mathcal{Z}_n^*$ if G and N are large enough (Tierney (1994)).
- Since these vectors could be affected by the initial state $p^{(0)}$, we apply the algorithm T times, obtaining a final sample of length $T(Q + 1)$.

Gibbs Sampler: Developing the Method

Algorithm

```
Fixed p^(0)
Do l = 1
   Generate Z_n^(l) ~ Z_n | Z_n^*, p^(l-1)
   Generate p^(l) ~ p | Z_n^(l)
Do l = l + 1
```

- For a run of the sequence $\{p^{(l)}\}_{l\geq 0}$, we choose $Q+1$ vectors in the way $\{p^{(N)}, p^{(N+G)}, \ldots, p^{(N+QG)}\}$, where N is the burn-in period and G is a batch size.
- The vectors $\{p^{(N)}, p^{(N+G)}, \ldots, p^{(N+QG)}\}$ are considered independent samples from $p|\mathcal{Z}_n^*$ if G and N are large enough (Tierney (1994)).
- Since these vectors could be affected by the initial state $p^{(0)}$, we apply the algorithm T times, obtaining a final sample of length $T(Q+1)$.

Gibbs Sampler: Simulated Example

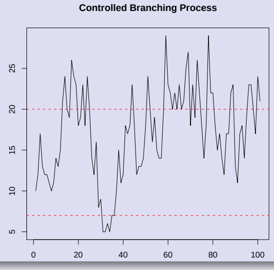
Offspring Distribution:

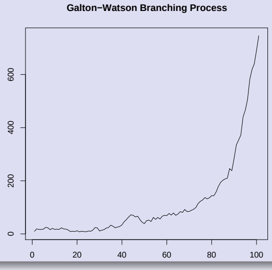
k	0	1	2	3	4
p_k	0.28398	0.42014	0.233090	0.05747	0.00531

Parameters: $m = 1.08$, $\sigma^2 = 0.7884$

Control function: $\phi(x) = 7 \text{ if } x \leq 7; \quad x \text{ if } 7 < x \leq 20; \quad 20 \text{ if } x > 20$

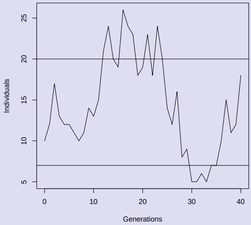
Simulated Data





Gibbs Sampler: Simulated Example

Observed Data: $n = 40$



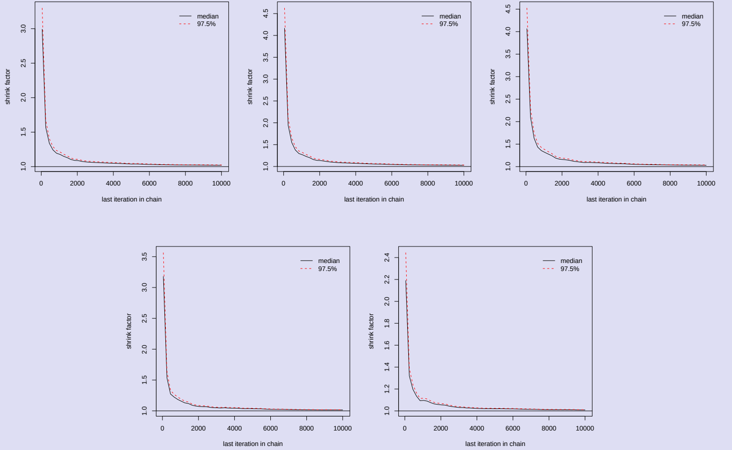
$p \sim D(1/2, \ldots, 1/2)$

Selection of N, G, Q and T

- Gelman-Rubin-Brooks diagnostic plots.
- Estimated potential scale reduction factor.
- Autocorrelation values.

Gibbs Sampler: Simulated Example

Gelman-Rubin-Brooks diagnostic plots (CODA package for **R**)



$p \sim D(1/2, \ldots, 1/2)$

Selection of N, G, Q and T

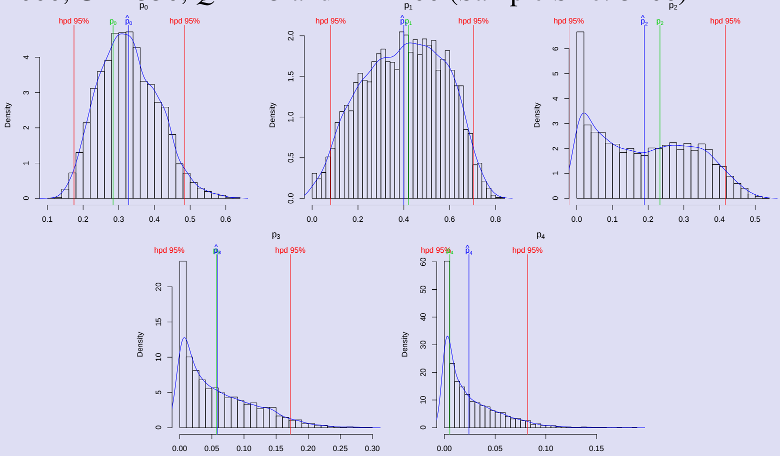
$N = 1000$, $G = 350$, $Q = 25$ and $T = 200$

- Gelman-Rubin-Brooks diagnostic plots.
- Estimated potential scale reduction factor.
- Autocorrelation values.

Gibbs Sampler: Simulated Example

Sample Information: $\mathcal{Z}_n^*$

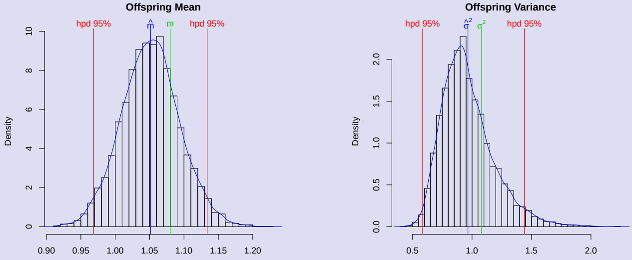
$N = 1000$, $G = 350$, $Q = 25$ and $T = 200$ (Sample Size: 5200)



Gibbs Sampler: Simulated Example

Sample Information: $\mathcal{Z}_n^*$

$N = 1000$, $G = 350$, $Q = 25$ and $T = 200$ (Sample Size: 5200)



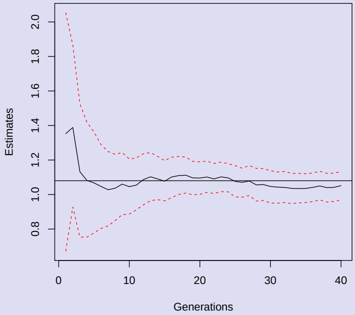
Algorithm's Efficiency

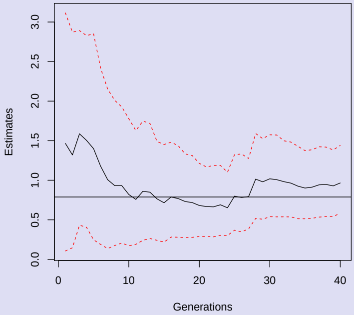
	MEAN	SD	MCSE	TSSE
m	1.051518	0.042049	0.000583	0.000551
σ^2	0.965560	0.219196	0.003040	0.002793

Gibbs Sampler: Simulated Example

Sample Information: $\mathcal{Z}_n^*$

$N = 1000$, $G = 350$, $Q = 25$ and $T = 200$ (Sample Size: 5200)





Non-Parametric/Parametric Framework

Offspring Distribution: $p = \{p_k : k \in \mathcal{S}\}$ $\quad \mathcal{S}$ finite.

Random Control Function: Power series family distributions, i.e.

$$P(\phi_n(k) = j) = a_k(j)\theta^j / A_k(\theta), \quad j = 0, 1, \ldots, \ \theta \in \Theta, \ k = 1, 2, \ldots$$

$a_k(j) \geq 0$ known values, $A_k(\theta) = \sum_{j=0}^{\infty} a_k(j)\theta^j$, $\Theta = \{\theta > 0 : A_k(\theta) < \infty\}$ open subset of $\mathbb{R}$.

- Regularity assumption:

$$\prod_{k \in B} A_k(\theta) = A_{\sum_{k \in B} k}(\theta), \text{ for every } B \subseteq \mathbb{N}, \ \theta \in \Theta.$$

Sample: The entire family tree up to the current generation, $\mathcal{Z}_n$.

Objective: Make inference on (p, θ)

Non-Parametric/Parametric Framework

Offspring Distribution: $p = \{p_k : k \in \mathcal{S}\}$ $\mathcal{S}$ finite.
Random Control Function: Power series family distributions, i.e.

$$P(\phi_n(k) = j) = a_k(j)\theta^j / A_k(\theta), \quad j = 0, 1, \ldots, \ \theta \in \Theta, \ k = 1, 2, \ldots$$

$a_k(j) \geq 0$ known values, $A_k(\theta) = \sum_{j=0}^{\infty} a_k(j)\theta^j$, $\Theta = \{\theta > 0 : A_k(\theta) < \infty\}$ open subset of $\mathbb{R}$.

- Regularity assumption:

$$\prod_{k \in B} A_k(\theta) = A_{\sum_{k \in B} k}(\theta), \text{ for every } B \subseteq \mathbb{N}, \ \theta \in \Theta.$$

Sample: The entire family tree up to the current generation, $\mathcal{Z}_n$.
Objective: Make inference on (p, θ)

Likelihood Function

$$f(\mathcal{Z}_n|p,\theta) \propto \prod_{k\in\mathcal{S}} p_k^{Z^*_{n,k}} \theta^{Y^*_n} / A_{Y_n}(\theta)$$

with $Z^*_{n,k} = \sum_{l=0}^{n-1} Z_l(k)$, $k \in \mathcal{S}$; $Y_n = \sum_{j=0}^{n-1} Z_j$ and $Y^*_n = \sum_{j=0}^{n-1} \phi_j(Z_j)$.

Conjugate Class of Distributions

- Prior Distribution: $(p,\theta) \sim p \otimes \theta$ with $p \sim D(\alpha_k : k \in \mathcal{S})$ and

$$\pi(\theta) = \varphi(a,b)^{-1}\theta^a / A_b(\theta), \text{ where } \varphi(a,b) = \int_\Theta \theta^a / A_b(\theta) d\theta.$$

- Posterior Distribution: $(p,\theta)|\mathcal{Z}_n \sim p|\mathcal{Z}_n \otimes \theta|\mathcal{Z}_n$ with $p|\mathcal{Z}_n \sim D(\alpha_k + Z^*_{n,k} : k \in \mathcal{S})$ and

$$\pi(\theta|\mathcal{Z}_n) = \varphi(a+Y^*_n, b+Y_n)^{-1}\theta^{a+Y^*_n} / A_{b+Y_n}(\theta)$$

Likelihood Function

$$f(\mathcal{Z}_n|p,\theta) \propto \prod_{k\in\mathcal{S}} p_k^{Z^*_{n,k}} \theta^{Y^*_n} / A_{Y_n}(\theta)$$

with $Z^*_{n,k} = \sum_{l=0}^{n-1} Z_l(k)$, $k \in \mathcal{S}$; $Y_n = \sum_{j=0}^{n-1} Z_j$ and $Y^*_n = \sum_{j=0}^{n-1} \phi_j(Z_j)$.

Conjugate Class of Distributions

- **Prior Distribution:** $(p,\theta) \sim p \otimes \theta$ with $p \sim D(\alpha_k : k \in \mathcal{S})$ and

$$\pi(\theta) = \varphi(a,b)^{-1}\theta^a/A_b(\theta), \text{ where } \varphi(a,b) = \int_\Theta \theta^a/A_b(\theta)d\theta.$$

- **Posterior Distribution:** $(p,\theta)|\mathcal{Z}_n \sim p|\mathcal{Z}_n \otimes \theta|\mathcal{Z}_n$ with $p|\mathcal{Z}_n \sim D(\alpha_k + Z^*_{n,k} : k \in \mathcal{S})$ and

$$\pi(\theta|\mathcal{Z}_n) = \varphi(a+Y^*_n, b+Y_n)^{-1}\theta^{a+Y^*_n}/A_{b+Y_n}(\theta)$$

Gibbs Sampler: Introducing the Method

- Usual Sample Information: $\mathcal{Z}_n^* = \{Z_j : j = 0, \ldots, n\}$

The Problem

$$(p, \theta) | \mathcal{Z}_n^*$$

- Latent Variables:

$$\mathcal{Z}_n = \{Z_j(k) : k \in \mathcal{S}, j = 0, \ldots, n\}$$

- Gibbs Sampler:

$$(p, \theta) | \mathcal{Z}_n, \mathcal{Z}_n^* \qquad \mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$$

Gibbs Sampler: Introducing the Method

First Conditional Distribution: $(p, \theta)|\mathcal{Z}_n, \mathcal{Z}_n^*$

$$(p, \theta)|\mathcal{Z}_n, \mathcal{Z}_n^* \equiv (p, \theta)|\mathcal{Z}_n = p|\mathcal{Z}_n \otimes \theta|\mathcal{Z}_n$$

$$p|\mathcal{Z}_n \sim D(\alpha_k + Z_{n,k}^* : k \in \mathcal{S})$$

$$\pi(\theta|\mathcal{Z}_n) = \varphi(a + Y_n^*, b + Y_n)^{-1}\theta^{a+Y_n^*}/A_{b+Y_n}(\theta)$$

- For $j = 0, \ldots, n$

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k) \quad Z_{j+1} = \sum_{k \in \mathcal{S}} kZ_j(k)$$

Gibbs Sampler: Introducing the Method

First Conditional Distribution: $(p,\theta)|\mathcal{Z}_n, \mathcal{Z}_n^*$

$$(p,\theta)|\mathcal{Z}_n, \mathcal{Z}_n^* \equiv (p,\theta)|\mathcal{Z}_n \equiv p|\mathcal{Z}_n \otimes \theta|\mathcal{Z}_n$$

$$p|\mathcal{Z}_n \sim D(\alpha_k + Z_{n,k}^* : k \in \mathcal{S})$$

$$\pi(\theta|\mathcal{Z}_n) = \varphi(a + Y_n^*, b + Y_n)^{-1}\theta^{a+Y_n^*}/A_{b+Y_n}(\theta)$$

- For $j = 0, \ldots, n$

$$\phi_j(Z_j) = \sum_{k\in\mathcal{S}} Z_j(k) \quad Z_{j+1} = \sum_{k\in\mathcal{S}} kZ_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n|\mathcal{Z}_n^*, p, \theta$

$$f(\mathcal{Z}_n|\mathcal{Z}_n^*, p, \theta) = \prod_{j=0}^{n} f(Z_j(k) : k \in \mathcal{S}|Z_j, Z_{j+1}, p, \theta)$$

$$P(Z_l(k) = z_l(k),\ k \in \mathcal{S} \mid Z_l = z_l, Z_{l+1} = z_{l+1}, p, \theta)$$
$$= \frac{1}{p_{z_l, z_{l+1}}} \frac{\phi_l^*!}{\prod_{k\in\mathcal{S}} z_l(k)!} \prod_{k\in\mathcal{S}} p_k^{z_l(k)} a_l(\phi_l^*)\, \theta^{\phi_l^*} / A_l(\theta)$$

$z_l = \sum_{k\in\mathcal{S}} z_l(k)$, $z_{l+1} = \sum_{k\in\mathcal{S}} k z_l(k)$, $\phi_l^* = \sum_{k\in\mathcal{S}} z_l(k)$ and $p_{z_l, z_{l+1}} = P(Z_{l+1} = z_{l+1} \mid Z_l = z_l, p, \theta)$

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

$$f(\mathcal{Z}_n|\mathcal{Z}_n^*, p, \theta) = \prod_{j=0}^{n} f(Z_j(k) : k \in \mathcal{S} | Z_j, Z_{j+1}, p, \theta)$$

$$P(Z_l(k) = z_l(k),\ k \in \mathcal{S} \mid Z_l = z_l, Z_{l+1} = z_{l+1}, p, \theta)$$
$$= \frac{1}{p_{z_l, z_{l+1}}} \frac{\phi_l^*!}{\prod_{k \in \mathcal{S}} z_l(k)!} \prod_{k \in \mathcal{S}} p_k^{z_l(k)} a_l\left(\phi_l^*\right) \theta^{\phi_l^*} / A_l(\theta)$$

$z_l = \sum_{k \in \mathcal{S}} z_l(k)$, $z_{l+1} = \sum_{k \in \mathcal{S}} k z_l(k)$, $\phi_l^* = \sum_{k \in \mathcal{S}} z_l(k)$ and $p_{z_l, z_{l+1}} = P(Z_{l+1} = z_{l+1} \mid Z_l = z_l, p, \theta)$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

$$(p, \theta)$$

Z_0 $\qquad$ $\phi_0(Z_0)$

$Z_0(k), k \in \mathcal{S}$

Z_1 $\qquad$ $\phi_1(Z_1)$

$Z_1(k), k \in \mathcal{S}$

Z_2 $\qquad$ $\phi_2(Z_2)$

$\vdots \qquad \vdots \qquad \vdots$

Z_n $\qquad$ $\phi_n(Z_n)$

$Z_n(k), k \in \mathcal{S}$

Z_{n+1}

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

(p, θ)

Z_0 $\qquad$ $\phi_0(Z_0)$

$Z_0(k), k \in \mathcal{S}$

Z_1 $\qquad$ $\phi_1(Z_1)$

$Z_1(k), k \in \mathcal{S}$

Z_2 $\qquad$ $\phi_2(Z_2)$

$\vdots \qquad \vdots \qquad \vdots$

Z_n $\qquad$ $\phi_n(Z_n)$

$Z_n(k), k \in \mathcal{S}$

Z_{n+1}

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

(p, θ)

Z_0 $\qquad$ $\phi_0(Z_0)$

$Z_0(k), k \in \mathcal{S}$

Z_1 $\qquad$ $\phi_1(Z_1)$

$Z_1(k), k \in \mathcal{S}$

Z_2 $\qquad$ $\phi_2(Z_2)$

$\vdots \qquad \vdots \qquad \vdots$

Z_n $\qquad$ $\phi_n(Z_n)$

$Z_n(k), k \in \mathcal{S}$

Z_{n+1}

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

$$(p, \theta)$$

$$\begin{array}{ccc}
Z_0 & & \phi_0(Z_0) \\
 & Z_0(k), k \in \mathcal{S} & \\
Z_1 & & \phi_1(Z_1) \\
 & Z_1(k), k \in \mathcal{S} & \\
Z_2 & & \phi_2(Z_2) \\
\vdots & \vdots & \vdots \\
Z_n & & \phi_n(Z_n) \\
 & Z_n(k), k \in \mathcal{S} & \\
Z_{n+1} & &
\end{array}$$

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

$$(p, \theta)$$

$$\begin{array}{ccc}
Z_0 & & \phi_0(Z_0) \\
& Z_0(k), k \in \mathcal{S} & \\
Z_1 & & \phi_1(Z_1) \\
& Z_1(k), k \in \mathcal{S} & \\
Z_2 & & \phi_2(Z_2) \\
\vdots & \vdots & \vdots \\
Z_n & & \phi_n(Z_n) \\
& Z_n(k), k \in \mathcal{S} & \\
Z_{n+1} & &
\end{array}$$

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

$$(p, \theta)$$

Z_0		$\phi_0(Z_0)$
	$Z_0(k), k \in \mathcal{S}$	
Z_1		$\phi_1(Z_1)$
	$Z_1(k), k \in \mathcal{S}$	
Z_2		$\phi_2(Z_2)$
$\vdots$	$\vdots$	$\vdots$
Z_n		$\phi_n(Z_n)$
	$Z_n(k), k \in \mathcal{S}$	
Z_{n+1}		

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

$$
\begin{array}{ccc}
 & (p, \theta) & \\
Z_0 & & \phi_0(Z_0) \\
 & Z_0(k), k \in \mathcal{S} & \\
Z_1 & & \phi_1(Z_1) \\
 & Z_1(k), k \in \mathcal{S} & \\
Z_2 & & \phi_2(Z_2) \\
\vdots & \vdots & \vdots \\
Z_n & & \phi_n(Z_n) \\
 & Z_n(k), k \in \mathcal{S} & \\
Z_{n+1} & &
\end{array}
$$

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Gibbs Sampler: Introducing the Method

Second Conditional Distribution: $\mathcal{Z}_n | \mathcal{Z}_n^*, p, \theta$

$$
\begin{array}{ccc}
 & (p, \theta) & \\
Z_0 & & \phi_0(Z_0) \\
 & Z_0(k), k \in \mathcal{S} & \\
Z_1 & & \phi_1(Z_1) \\
 & Z_1(k), k \in \mathcal{S} & \\
Z_2 & & \phi_2(Z_2) \\
\vdots & \vdots & \vdots \\
Z_n & & \phi_n(Z_n) \\
 & Z_n(k), k \in \mathcal{S} & \\
Z_{n+1} & &
\end{array}
$$

$$\phi_j(Z_j) = \sum_{k \in \mathcal{S}} Z_j(k), \quad Z_{j+1} = \sum_{k \in \mathcal{S}} k Z_j(k)$$

Algorithm

Initialize $l = 0$

Generate $p^{(0)} \sim$ Dirichlet(α)

Generate $\theta^{(0)}$ from $\pi(\theta) = \varphi(a,b)^{-1}\theta^{a}/A_b(\theta)$

Iterate

  $l = l + 1$

  Generate $\mathcal{Z}_n^{(l)} \sim f(\mathcal{Z}_n \mid \mathcal{Z}_n^*, p^{(l-1)}, \theta^{(l-1)})$

  Generate $(p^{(l)}, \theta^{(l)}) \sim \pi(p, \theta \mid \mathcal{Z}_n^{(l)})$

- For a run of the sequence $\{(\theta,p)^{(l)}\}_{l\geq 0}$, we choose $Q+1$ vectors in the way $\{(\theta,p)^{(N)}, (\theta,p)^{(N+G))}, \ldots, (\theta,p)^{(N+QG)}\}$, where N is a burning period and G is a batch size.
- The vectors $\{(\theta,p)^{(N)}, (\theta,p)^{(N+G)}, \ldots, (\theta,p)^{(N+QG)}\}$ are considered independent samples from $(\theta,p)|\mathcal{Z}_n^*$ if G and N are large enough.
- Since these vectors could be affected by the initial state $(\theta,p)^{(0)}$, we apply the algorithm T times, obtaining a final sample of length $T(Q+1)$.

Gibbs Sampler: Developing the Method

Algorithm

```
Initialize l = 0
Generate p^(0) ~ Dirichlet(α)
Generate θ^(0) from π(θ) = φ(a,b)^(-1) θ^a / A_b(θ)
Iterate
   l = l + 1
   Generate Z_n^(l) ~ f(Z_n | Z_n^*, p^(l-1), θ^(l-1))
   Generate (p^(l), θ^(l)) ~ π(p, θ | Z_n^(l))
```

- For a run of the sequence $\{(\theta,p)^{(l)}\}_{l\geq 0}$, we choose $Q+1$ vectors in the way $\{(\theta,p)^{(N)},(\theta,p)^{(N+G))},\ldots,(\theta,p)^{(N+QG)}\}$, where N is a burning period and G is a batch size.
- The vectors $\{(\theta,p)^{(N)},(\theta,p)^{(N+G)},\ldots,(\theta,p)^{(N+QG)}\}$ are considered independent samples from $(\theta,p)|\mathcal{Z}_n^*$ if G and N are large enough.
- Since these vectors could be affected by the initial state $(\theta,p)^{(0)}$, we apply the algorithm T times, obtaining a final sample of length $T(Q+1)$.

Gibbs Sampler: Simulated Example

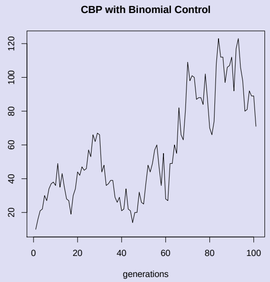
Offspring Distribution:

k	0	1	2	3	4
p_k	0.0081	0.0756	0.2646	0.4116	0.2401

Parameters: $m = 2.8$, $\sigma^2 = 0.84$

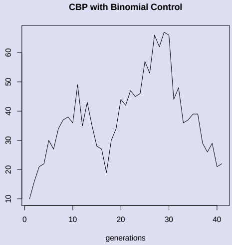
Random Control function: $\phi_n(k) \sim \text{Binom}(k, \theta)$, $k = 0, 1, \ldots; \theta = 0.35$

Simulated Data



Gibbs Sampler: Simulated Example

Observed Data: $n = 40$



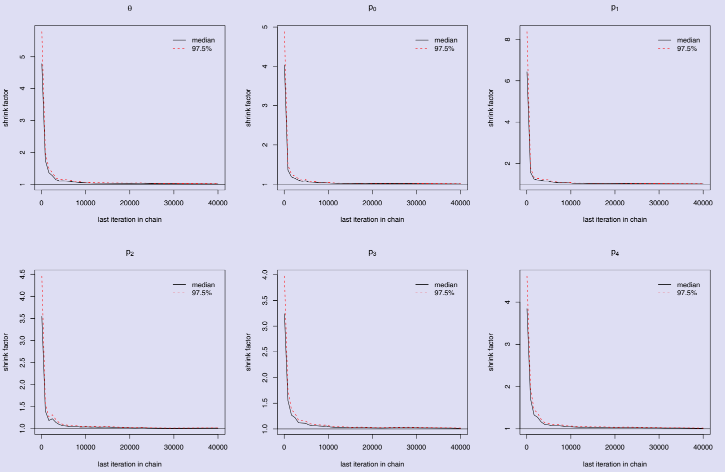
$p \sim D(1/2, \ldots, 1/2)$

Selection of N, G, Q and T

- Gelman-Rubin-Brooks diagnostic plots.
- Estimated potential scale reduction factor.
- Autocorrelation values.

Gibbs Sampler: Simulated Example

Gelman-Rubin-Brooks diagnostic plots (CODA package for **R**)



Gibbs Sampler: Simulated Examples

$p \sim D(1/2, \ldots, 1/2)$

Selection of N, G, Q and T

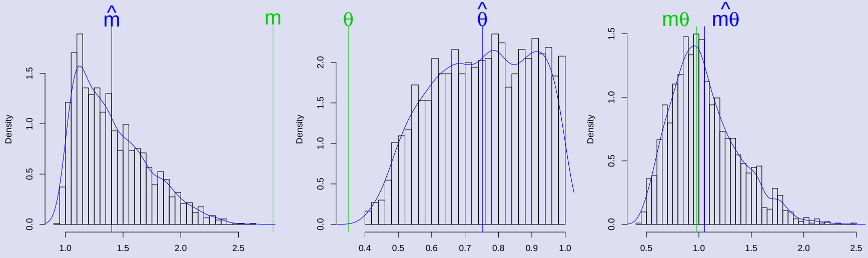
$N = 10000$, $G = 1000$, $Q = 30$ and $T = 59$

- Gelman-Rubin-Brooks diagnostic plots.
- Estimated potential scale reduction factor.
- Autocorrelation values.

Gibbs Sampler: Simulated Example

Sample Information: $\mathcal{Z}_n^*$

$N = 10000$, $G = 1000$, $Q = 30$ and $T = 59$ (Sample Size: 1770)



Algorithm's Efficiency

	MEAN	SD	MCSE	TSSE
m	1.4021	0.3100	0.0073	0.0077
θ	0.7518	0.1492	0.0035	0.0036
$m\theta$	1.0551	0.3203	0.0075	0.0074

Approximate Bayesian Computation

Marin, J.M., Pudlo,P., Robert,C.P., Ryder,R.J. (2011). Approximate Bayesian computational methods. Statistics and Computing.
DOI 10.1007/s11222-011-9288-2

Likelihood-free rejection sampler: $\pi_\varepsilon(p,\theta|\mathcal{Z}_n^*)$

```
for i = 1 to N do
  repeat
    Generate p' ~ Dirichlet(α)
    Generate θ' from  π(θ) = φ(a,b)^{-1} θ^a / A_b(θ)
    Generate Z'_n from the likelihood f(Z_n|p',θ')
  until ρ(S(Z'_n), S(Z_n)) ≤ ε
  set (p_i, θ_i) = (p', θ')
end for
```

- $\mathcal{S}(\cdot)$ a function on $\mathcal{Z}_n$ defining a summary statistic: $\mathcal{S}(\mathcal{Z}_n) = \mathcal{Z}_n^*$.
- ρ is a metric on $\mathcal{S}(\mathcal{Z}_n)$.
- ε a tolerance level.

Approximate Bayesian Computation

Marin, J.M., Pudlo,P., Robert,C.P., Ryder,R.J. (2011). Approximate Bayesian computational methods. Statistics and Computing.
DOI 10.1007/s11222-011-9288-2

Likelihood-free rejection sampler: $\pi_\varepsilon(p,\theta|\mathcal{Z}_n^*)$

for $i=1$ to N do
  repeat
    Generate $p' \sim$ Dirichlet(α)
    Generate θ' from $\pi(\theta)=\varphi(a,b)^{-1}\theta^a/A_b(\theta)$
    Generate $\mathcal{Z}'_n$ from the likelihood $f(\mathcal{Z}_n|p',\theta')$
  until $\rho(\mathcal{S}(\mathcal{Z}'_n),\mathcal{S}(\mathcal{Z}_n)) \le \varepsilon$
  set $(p_i,\theta_i)=(p',\theta')$
end for

- $\mathcal{S}(\cdot)$ a function on $\mathcal{Z}_n$ defining a summary statistic: $\mathcal{S}(\mathcal{Z}_n)=\mathcal{Z}_n^*$.
- ρ is a metric on $\mathcal{S}(\mathcal{Z}_n)$.
- ε a tolerance level.

Approximate Bayesian Computation

Marin, J.M., Pudlo,P., Robert,C.P., Ryder,R.J. (2011). Approximate Bayesian computational methods. Statistics and Computing.
DOI 10.1007/s11222-011-9288-2

Likelihood-free rejection sampler: $\pi_\varepsilon(p, \theta | \mathcal{Z}_n^*)$

```
for i = 1 to N do
  repeat
    Generate p' ~ Dirichlet(α)
    Generate θ' from  π(θ) = φ(a,b)^{-1} θ^a / A_b(θ)
    Generate Z'_n from the likelihood f(Z_n | p', θ')
  until ρ(S(Z'_n), S(Z_n)) ≤ ε
  set (p_i, θ_i) = (p', θ')
end for
```

- $\mathcal{S}(\cdot)$ a function on $\mathcal{Z}_n$ defining a summary statistic: $\mathcal{S}(\mathcal{Z}_n) = \mathcal{Z}_n^*$.
- ρ is a metric on $\mathcal{S}(\mathcal{Z}_n)$.
- ε a tolerance level.

Approximate Bayesian Computation

Likelihood-free rejection sampler: $\pi_\varepsilon(p, \theta | \mathcal{Z}_n^*)$

for $i = 1$ to N do
  repeat
    Generate $p' \sim$ Dirichlet(α)
    Generate θ' from $\pi(\theta) = \varphi(a,b)^{-1}\theta^a / A_b(\theta)$
    Generate $\mathcal{Z}'_n$ from the likelihood $f(\mathcal{Z}_n | p', \theta')$
  until $\rho(\mathcal{S}(\mathcal{Z}'_n), \mathcal{S}(\mathcal{Z}_n)) \leq \varepsilon$
  set $(p_i, \theta_i) = (p', \theta')$
end for

- ρ is a metric on $\mathcal{S}(\mathcal{Z}_n)$. Wilkinson (2008)

$$\rho(\mathcal{Z}_n^*, \mathcal{Z'}_n^*) = \left| \frac{\sum_{i=1}^n Z_i'}{\sum_{i=1}^n Z_i} - 1 \right| + \frac{1}{2} \sum_{j=1}^n \left| \frac{Z_j}{\sum_{i=1}^n Z_i} - \frac{Z_j'}{\sum_{i=1}^n Z_i'} \right|$$

ABC: Simulated Example

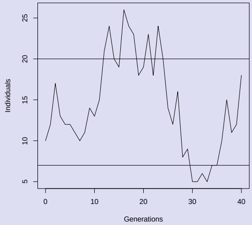
Offspring Distribution:

k	0	1	2	3	4
p_k	0.28398	0.42014	0.233090	0.05747	0.00531

Parameters: $m = 1.08$, $\sigma^2 = 0.7884$

Control function: $\phi(x) = 7$ if $x \leq 7$; x if $7 < x \leq 20$; 20 if $x > 20$

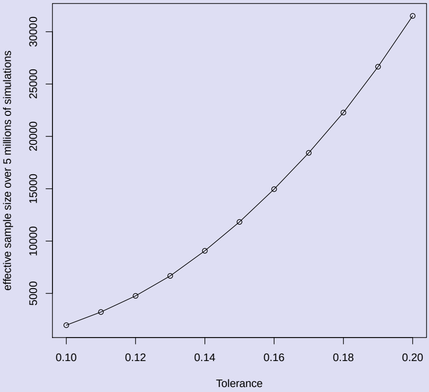
Observed Data: $n = 40$



ABC: Simulated Example

Sample Information $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

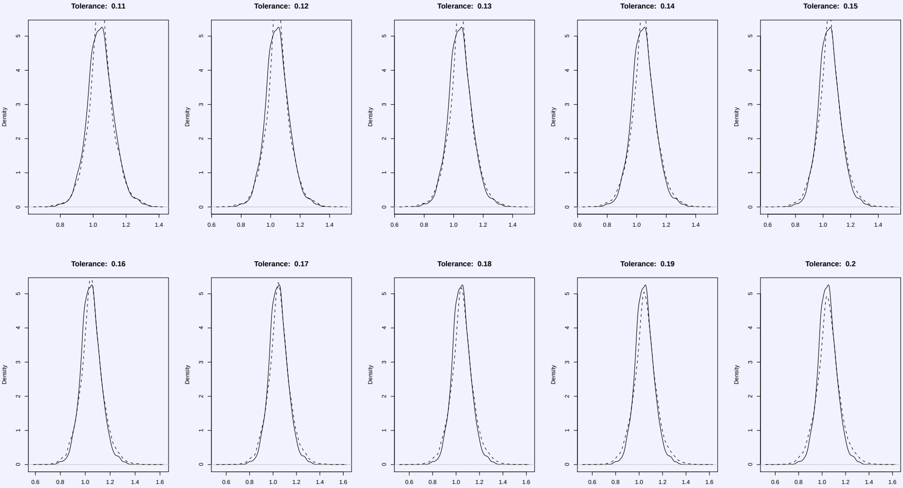
Generation 10



ABC: Simulated Example

Sample Information $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

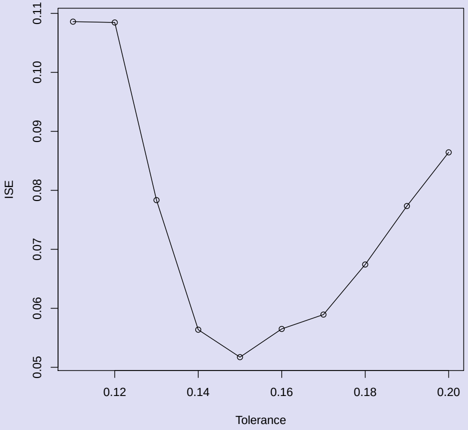
Generation 10



ABC: Simulated Example

Sample Information: $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

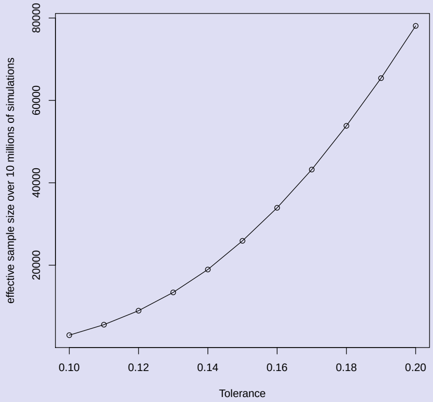
Generation 10



ABC: Simulated Example

Sample Information $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

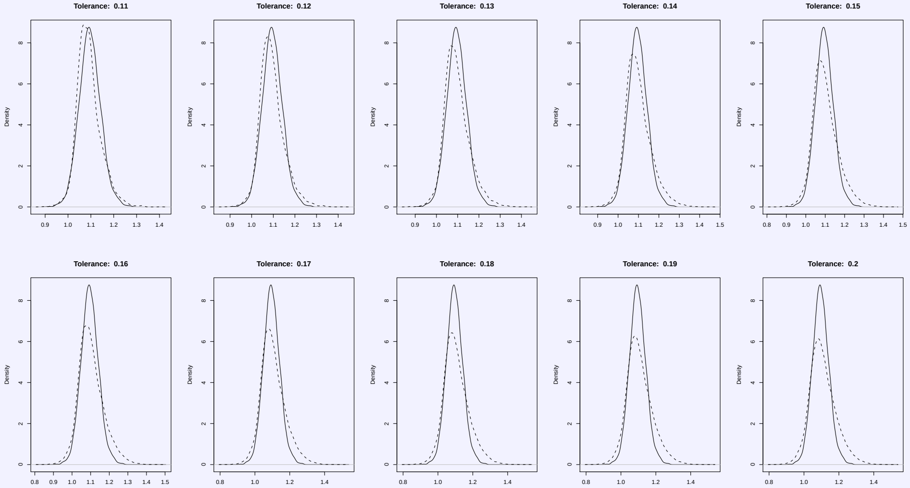
Generation 20



ABC: Simulated Example

Sample Information $\mathcal{Z}_n^*, p \sim D(1/2, \ldots, 1/2), N = 20$ millions

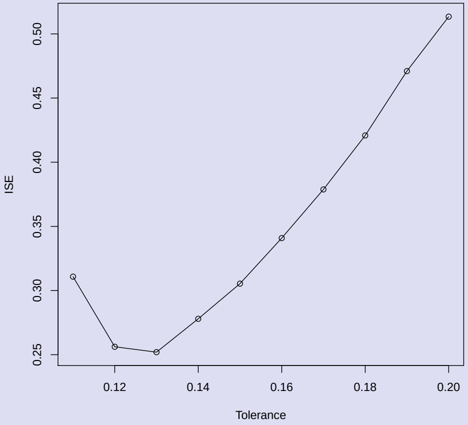
Generation 20



ABC: Simulated Example

Sample Information: $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

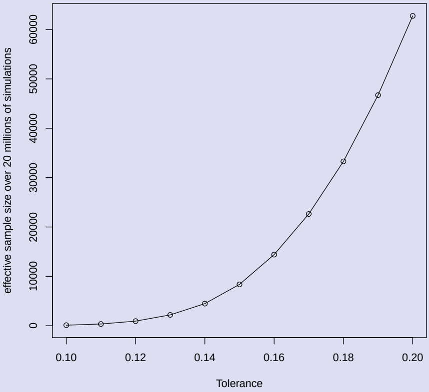
Generation 20



ABC: Simulated Example

Sample Information $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

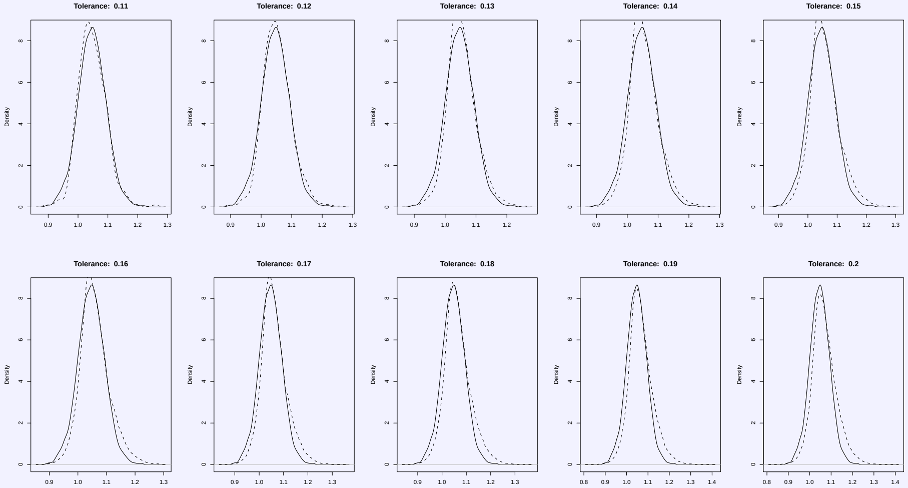
Generation 30



ABC: Simulated Example

Sample Information $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

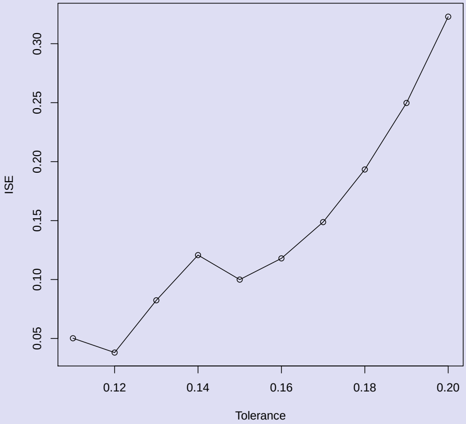
Generation 30



ABC: Simulated Example

Sample Information: $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

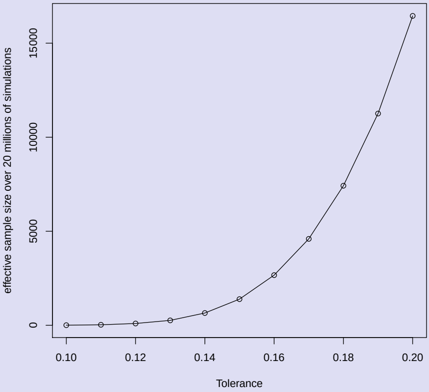
Generation 30



ABC: Simulated Example

Sample Information $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

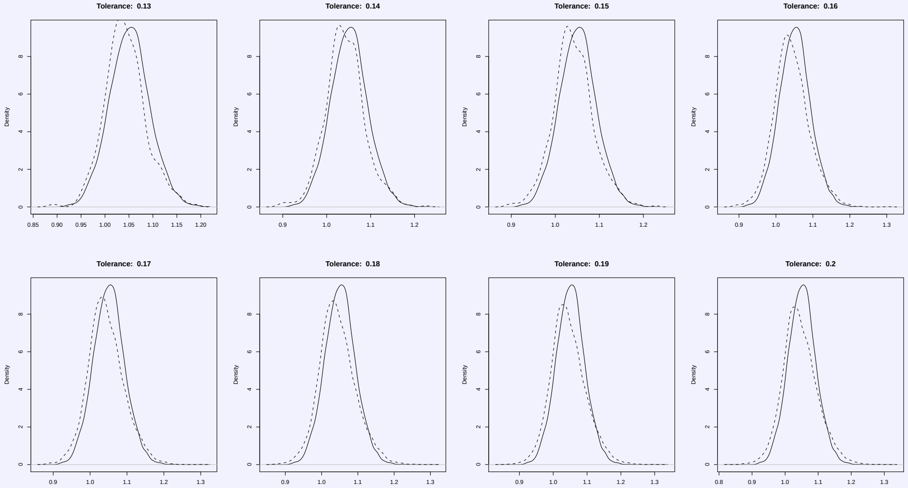
Generation 40



ABC: Simulated Example

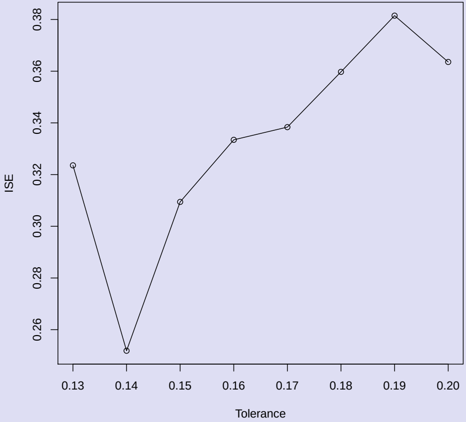
Sample Information $\mathcal{Z}_n^*$, $p \sim D(1/2, \ldots, 1/2)$, $N = 20$ millions

Generation 40



Sample Information: $\mathcal{Z}_n^*, p \sim D(1/2, \ldots, 1/2), N = 20$ millions

Generation 40



Concluding Remarks

- In a non-parametric Bayesian framework we can make inference on the offspring distribution of CBP, and consequently on the rest of offspring parameters, without observing the entire family tree, but only considering the total number of individuals in each generation.
- We use a MCMC method (Gibbs sampler) in order to give a "likely" approach to family trees, for both CBP with deterministic and with random control function.
- We take advantage of the ABC methodology to make inference on the main parameters of the model by simulating.
- The ABC approach shows a quite good behaviour, being a good alternative to the MCMC approach.
- We have developed the above methodologies using statistical software and programming environment R.

Concluding Remarks

- In a non-parametric Bayesian framework we can make inference on the offspring distribution of CBP, and consequently on the rest of offspring parameters, without observing the entire family tree, but only considering the total number of individuals in each generation.
- We use a MCMC method (Gibbs sampler) in order to give a "likely" approach to family trees, for both CBP with deterministic and with random control function.
- We take advantage of the ABC methodology to make inference on the main parameters of the model by simulating.
- The ABC approach shows a quite good behaviour, being a good alternative to the MCMC approach.
- We have developed the above methodologies using statistical software and programming environment R.

Concluding Remarks

- In a non-parametric Bayesian framework we can make inference on the offspring distribution of CBP, and consequently on the rest of offspring parameters, without observing the entire family tree, but only considering the total number of individuals in each generation.
- We use a MCMC method (Gibbs sampler) in order to give a "likely" approach to family trees, for both CBP with deterministic and with random control function.
- We take advantage of the ABC methodology to make inference on the main parameters of the model by simulating.
- The ABC approach shows a quite good behaviour, being a good alternative to the MCMC approach.
- We have developed the above methodologies using statistical software and programming environment R.

Concluding Remarks

- In a non-parametric Bayesian framework we can make inference on the offspring distribution of CBP, and consequently on the rest of offspring parameters, without observing the entire family tree, but only considering the total number of individuals in each generation.
- We use a MCMC method (Gibbs sampler) in order to give a "likely" approach to family trees, for both CBP with deterministic and with random control function.
- We take advantage of the ABC methodology to make inference on the main parameters of the model by simulating.
- The ABC approach shows a quite good behaviour, being a good alternative to the MCMC approach.
- We have developed the above methodologies using statistical software and programming environment R.

Concluding Remarks

- In a non-parametric Bayesian framework we can make inference on the offspring distribution of CBP, and consequently on the rest of offspring parameters, without observing the entire family tree, but only considering the total number of individuals in each generation.
- We use a MCMC method (Gibbs sampler) in order to give a "likely" approach to family trees, for both CBP with deterministic and with random control function.
- We take advantage of the ABC methodology to make inference on the main parameters of the model by simulating.
- The ABC approach shows a quite good behaviour, being a good alternative to the MCMC approach.
- We have developed the above methodologies using statistical software and programming environment R.

References

- Bagley, J. H. (1986). On the almost sure convergence of controlled branching processes. Journal of Applied Probability, 23:827-831.
- M. González, M. Molina, I. del Puerto (2002). On the class of controlled branching process with random control functions. Journal of Applied Probability, 39 (4), 804-815.
- M. González, M. Molina, I. del Puerto (2003). On the geometric growth in controlled branching processes with random control function. Journal of Applied Probability, 40(4), 995-1006.
- M. González, M. Molina, I. del Puerto (2004). Limiting distribution for subcritical controlled branching processes with random control function. Statistics and Probability Letters, 67(3), 277-284.
- M. González, M. Molina, I. del Puerto (2005a). Asymptotic behaviour of critical controlled branching process with random control function. Journal of Applied Probability, 42(2), 463-477.
- M. González, M. Molina, I. del Puerto (2005b). On the L2-convergence of controlled branching processes with random control function. Bernoulli, 11(1), 37-46.
- M. Molina, M. González, M. Mota (1998). Some theoretical results about superadditive controlled Galton-Watson branching processes. Proceedings of the International Conference Prague Stochastics98, 2, 413-418.
- Sevast'yanov, B. A. and Zubkov, A. (1974). Controlled branching processes. Theory of Probability and its Applications, 19:14-24.
- Tierney, L. (1994). Markov chains for exploring posterior distributions. Annals of Statistics, 22, 1701-1762.
- Wilkinson, R.D. (2008). Approximate Bayesian computation (ABC) gives exact results under the assumption of model error. Technical Report. arXiv:0811.3355.
- Zubkov, A. M. (1974). Analogies between Galton-Watson processes and ϕ-branching processes. Theory of Probability and its Applications,19:309-331.

Thank you very much!