Microsoft Research

Each year Microsoft Research hosts hundreds of influential speakers from around the world including leading scientists, renowned experts in technology, book authors, and leading academics, and makes videos of these lectures freely available.

2013 © Microsoft Corporation. All rights reserved.

Approximate Bayesian computation (ABC)

NIPS Tutorial

Richard Wilkinson
r.d.wilkinson@nottingham.ac.uk

School of Mathematical Sciences
University of Nottingham

December 5 2013

Computer experiments

Rohrlich (1991): Computer simulation is

> *'a key milestone somewhat comparable to the milestone that started the empirical approach (Galileo) and the deterministic mathematical approach to dynamics (Newton and Laplace)'*

Challenges for statistics:
How do we make inferences about the world from a simulation of it?

Computer experiments

Rohrlich (1991): Computer simulation is

> *'a key milestone somewhat comparable to the milestone that started the empirical approach (Galileo) and the deterministic mathematical approach to dynamics (Newton and Laplace)'*

Challenges for statistics:
How do we make inferences about the world from a simulation of it?

- how do we relate simulators to reality? (model error)
- how do we estimate tunable parameters? (calibration)
- how do we deal with computational constraints? (stat. comp.)
- how do we make uncertainty statements about the world that combine models, data and their corresponding errors? (UQ)

Calibration

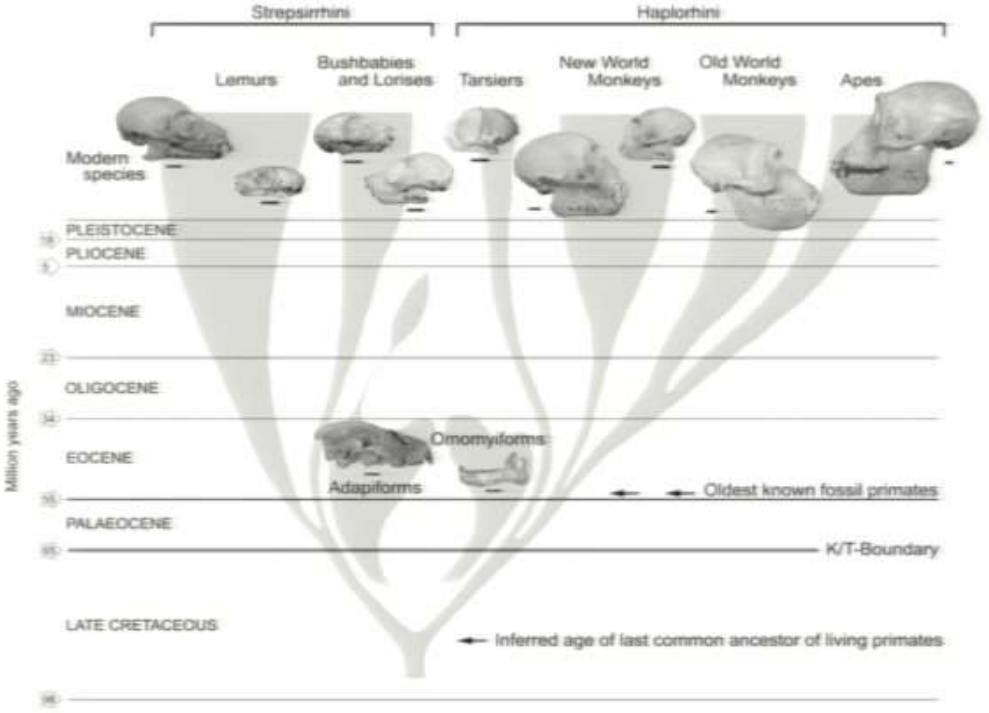
- For most simulators we specify parameters θ and i.c.s and the simulator, $f(\theta)$, generates output X.
- We are interested in the inverse-problem, i.e., observe data D, want to estimate parameter values θ which explain this data.

For Bayesians, this is a question of finding the posterior distribution

$$\pi(\theta|D) \propto \pi(\theta)\pi(D|\theta)$$

posterior $\propto$

prior $\times$ likelihood



Intractability

$$\pi(\theta|D) = \frac{\pi(D|\theta)\pi(\theta)}{\pi(D)}$$

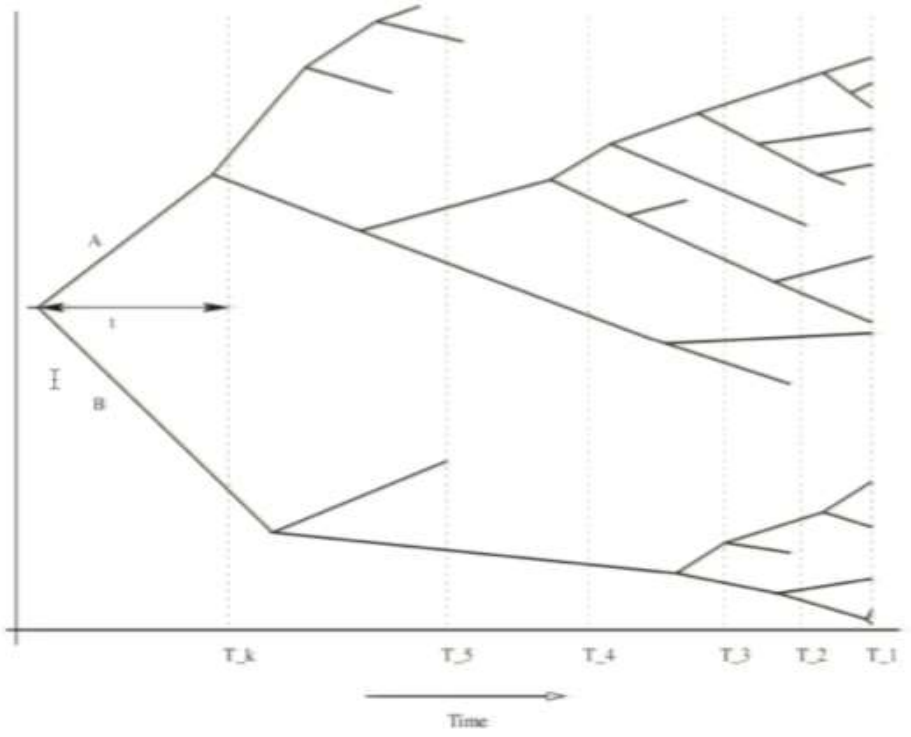
- usual intractability in Bayesian inference is not knowing $\pi(D)$.
- a problem is doubly intractable if $\pi(D|\theta) = c_\theta p(D|\theta)$ with c_θ unknown (cf Murray, Ghahramani and MacKay 2006)
- a problem is completely intractable if $\pi(D|\theta)$ is unknown and can't be evaluated (unknown is subjective). I.e., if the analytic distribution of the simulator, $f(\theta)$, run at θ is unknown.

Completely intractable models are where we need to resort to ABC methods

Common example

Tanaka *et al.* 2006, Wilkinson *et al.* 2009, Neal and Huang 2013 etc

Many models have unobserved branching processes that lead to the data making calculation difficult. For example, the density of the cumulative process is unknown in general.



Approximate Bayesian Computation (ABC)

Given a complex simulator for which we can't calculate the likelihood function - how do we do inference?

Approximate Bayesian Computation (ABC)

Given a complex simulator for which we can't calculate the likelihood function - how do we do inference?

If its cheap to simulate, then ABC (approximate Bayesian computation)is one of the few approaches we can use.

ABC algorithms are a collection of Monte Carlo methods used for calibrating simulators

- they do not require explicit knowledge of the likelihood function
- inference is done using simulation from the model (they are 'likelihood-free').

Approximate Bayesian computation (ABC)

ABC methods are primarily popular in biological disciplines, particularly genetics and epidemiology, and this looks set to continue growing.

- Simple to implement
- Intuitive
- Embarrassingly parallelizable
- Can usually be applied

ABC methods can be crude but they have an important role to play.

Approximate Bayesian computation (ABC)

ABC methods are primarily popular in biological disciplines, particularly genetics and epidemiology, and this looks set to continue growing.

- Simple to implement
- Intuitive
- Embarrassingly parallelizable
- Can usually be applied

ABC methods can be crude but they have an important role to play.

First ABC paper candidates

- Beaumont *et al.* 2002
- Tavaré *et al.* 1997 or Pritchard *et al.* 1999
- Or Diggle and Gratton 1984 or Rubin 1984
- ...

Tutorial Plan

Part I

i. Basics

ii. Efficient algorithms

iii. Links to other approaches

Part II

iv. Regression adjustments/ post-hoc corrections

v. Summary statistics

vi. Accelerating ABC using Gaussian processes

'Likelihood-Free' Inference

Rejection Algorithm

- Draw θ from prior $\pi(\cdot)$
- Accept θ with probability $\pi(D \mid \theta)$

Accepted θ are independent draws from the posterior distribution, $\pi(\theta \mid D)$.

'Likelihood-Free' Inference

Rejection Algorithm

- Draw θ from prior $\pi(\cdot)$
- Accept θ with probability $\pi(D \mid \theta)$

Accepted θ are independent draws from the posterior distribution, $\pi(\theta \mid D)$.
If the likelihood, $\pi(D|\theta)$, is unknown:

'Mechanical' Rejection Algorithm

- Draw θ from $\pi(\cdot)$
- Simulate $X \sim f(\theta)$ from the computer model
- Accept θ if $D = X$, i.e., if computer output equals observation

The acceptance rate is $\int \mathbb{P}(D|\theta)\pi(\theta)\mathrm{d}\theta = \mathbb{P}(D)$.
The number of runs to get n observations is negative binomial, with mean $\frac{n}{\mathbb{P}(D)}$: $\Rightarrow$ Bayes Factors!

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

'Likelihood-Free' Inference

Rejection Algorithm

- Draw θ from prior $\pi(\cdot)$
- Accept θ with probability $\pi(D \mid \theta)$

Accepted θ are independent draws from the posterior distribution, $\pi(\theta \mid D)$.
If the likelihood, $\pi(D|\theta)$, is unknown:

'Mechanical' Rejection Algorithm

- Draw θ from $\pi(\cdot)$
- Simulate $X \sim f(\theta)$ from the computer model
- Accept θ if $D = X$, i.e., if computer output equals observation

The acceptance rate is $\int \mathbb{P}(D|\theta)\pi(\theta)\mathrm{d}\theta = \mathbb{P}(D)$.
The number of runs to get n observations is negative binomial, with mean $\frac{n}{\mathbb{P}(D)}$: $\Rightarrow$ Bayes Factors!

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

'Likelihood-Free' Inference

Rejection Algorithm

- Draw θ from prior $\pi(\cdot)$
- Accept θ with probability $\pi(D \mid \theta)$

Accepted θ are independent draws from the posterior distribution, $\pi(\theta \mid D)$.
If the likelihood, $\pi(D|\theta)$, is unknown:

'Mechanical' Rejection Algorithm

- Draw θ from $\pi(\cdot)$
- Simulate $X \sim f(\theta)$ from the computer model
- Accept θ if $D = X$, i.e., if computer output equals observation

The acceptance rate is $\int \mathbb{P}(D|\theta)\pi(\theta)\mathrm{d}\theta = \mathbb{P}(D)$.
The number of runs to get n observations is negative binomial, with mean $\frac{n}{\mathbb{P}(D)}$: $\Rightarrow$ Bayes Factors!

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

This generates observations from $\pi(\theta \mid \rho(D, X) < \epsilon)$:

- As $\epsilon \to \infty$, we get observations from the prior, $\pi(\theta)$.
- If $\epsilon = 0$, we generate observations from $\pi(\theta \mid D)$.

ϵ reflects the tension between computability and accuracy.

For reasons that will become clear later, we call this *uniform-ABC*.

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

'Likelihood-Free' Inference

Rejection Algorithm

- Draw θ from prior $\pi(\cdot)$
- Accept θ with probability $\pi(D \mid \theta)$

Accepted θ are independent draws from the posterior distribution, $\pi(\theta \mid D)$.
If the likelihood, $\pi(D|\theta)$, is unknown:

'Mechanical' Rejection Algorithm

- Draw θ from $\pi(\cdot)$
- Simulate $X \sim f(\theta)$ from the computer model
- Accept θ if $D = X$, i.e., if computer output equals observation

The acceptance rate is $\int \mathbb{P}(D|\theta)\pi(\theta)\mathrm{d}\theta = \mathbb{P}(D)$.
The number of runs to get n observations is negative binomial, with mean $\frac{n}{\mathbb{P}(D)}$: $\Rightarrow$ Bayes Factors!

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

This generates observations from $\pi(\theta \mid \rho(D, X) < \epsilon)$:

- As $\epsilon \to \infty$, we get observations from the prior, $\pi(\theta)$.
- If $\epsilon = 0$, we generate observations from $\pi(\theta \mid D)$.

ϵ reflects the tension between computability and accuracy.

For reasons that will become clear later, we call this *uniform-ABC*.

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

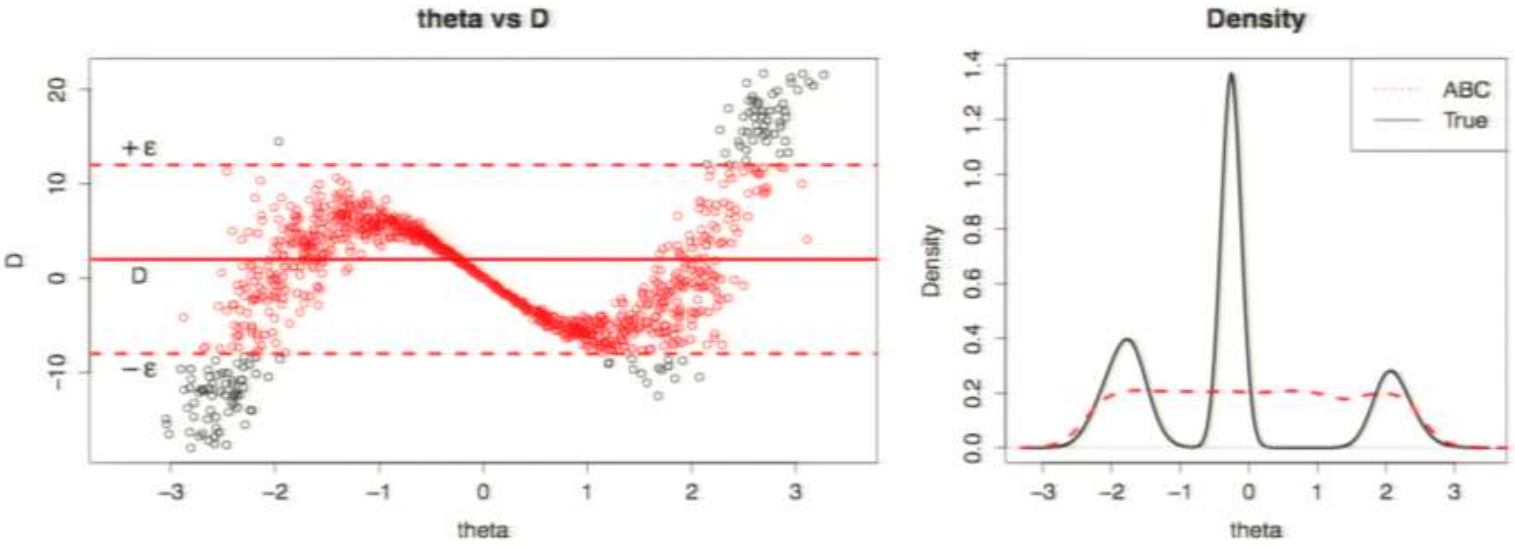
This generates observations from $\pi(\theta \mid \rho(D, X) < \epsilon)$:

- As $\epsilon \to \infty$, we get observations from the prior, $\pi(\theta)$.
- If $\epsilon = 0$, we generate observations from $\pi(\theta \mid D)$.

ϵ reflects the tension between computability and accuracy.

For reasons that will become clear later, we call this *uniform-ABC*.

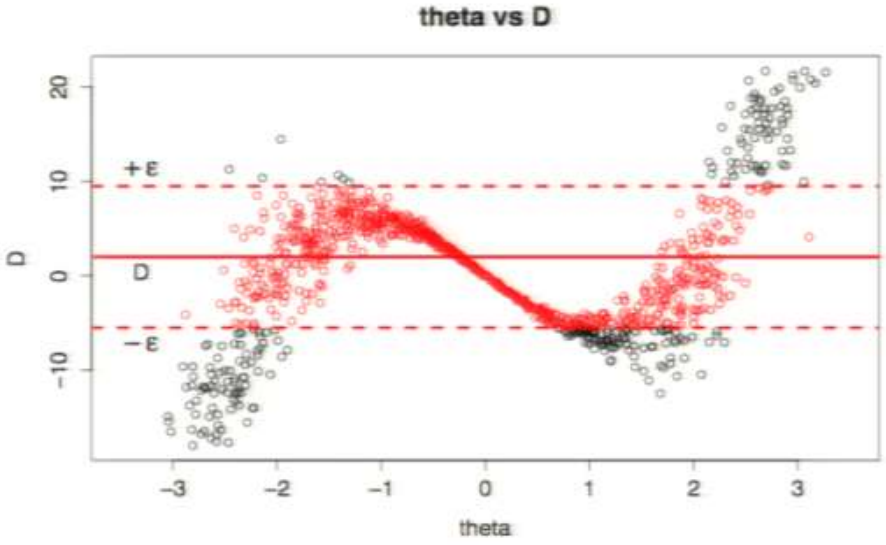
$\epsilon = 10$

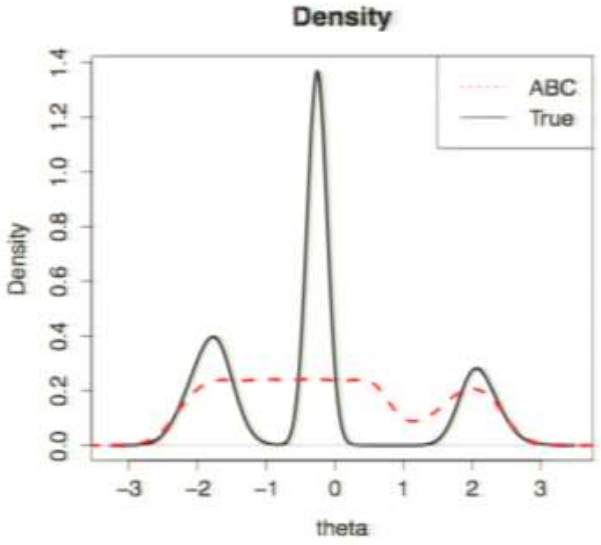


$$\theta \sim U[-10, 10], \qquad X \sim N(2(\theta + 2)\theta(\theta - 2), 0.1 + \theta^2)$$

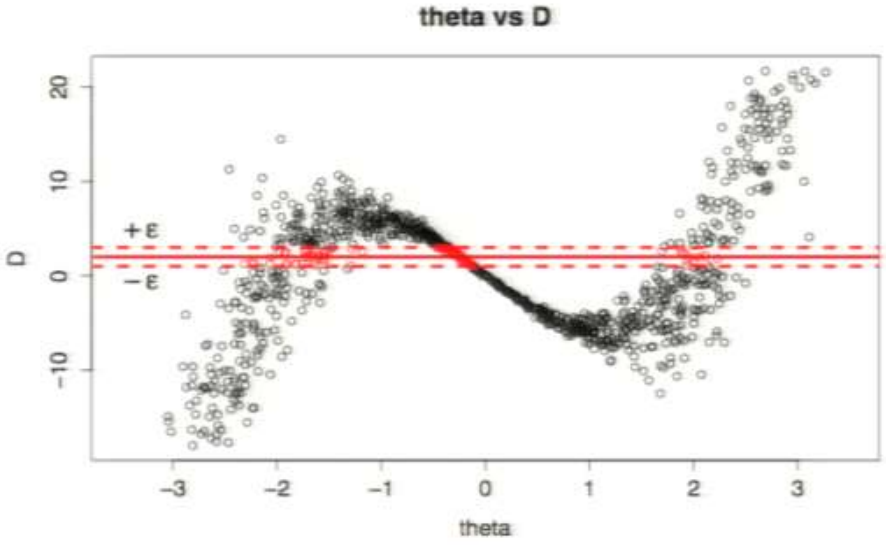
$$\rho(D, X) = |D - X|, \qquad D = 2$$

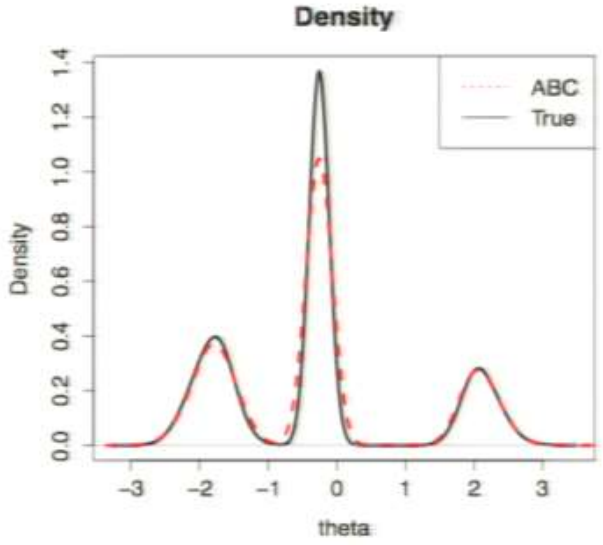
$\epsilon = 7.5$


theta vs D
D
theta
+ε
D
−ε
20
10
0
−10
−3
−2
−1
0
1
2
3


Density
Density
theta
ABC
True
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
−3
−2
−1
0
1
2
3

$\epsilon = 1$


theta vs D
D
theta
+ε
−ε
20
10
0
−10
−3
−2
−1
0
1
2
3


Density
Density
theta
ABC
True
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
−3
−2
−1
0
1
2
3

Rejection ABC

If the data are too high dimensional we never observe simulations that are 'close' to the field data - curse of dimensionality
Reduce the dimension using summary statistics, $S(D)$.

Approximate Rejection Algorithm With Summaries

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(S(D), S(X)) < \epsilon$

If S is sufficient this is equivalent to the previous algorithm.

Two ways of thinking

We think about linear regression in two ways

- Algorithmic: find the straight line that minimizes the sum of squared errors
- Probabilistic: a linear model with Gaussian errors fit using MAP estimates.

Two ways of thinking

We think about linear regression in two ways

- Algorithmic: find the straight line that minimizes the sum of squared errors
- Probabilistic: a linear model with Gaussian errors fit using MAP estimates.

Kalman filter:

- Algorithmic: linear quadratic estimation - find the best guess at the trajectory using linear dynamics and a quadratic penalty function
- Probabilistic: the (Bayesian) solution to the linear Gaussian filtering problem.

Two ways of thinking

We think about linear regression in two ways

- Algorithmic: find the straight line that minimizes the sum of squared errors
- Probabilistic: a linear model with Gaussian errors fit using MAP estimates.

Kalman filter:

- Algorithmic: linear quadratic estimation - find the best guess at the trajectory using linear dynamics and a quadratic penalty function
- Probabilistic: the (Bayesian) solution to the linear Gaussian filtering problem.

The same dichotomy exists for ABC.

- Algorithmic: find a good metric, tolerance and summary etc
- Probabilistic: What model does ABC correspond to, and how should this inform our choices?

Modelling interpretation - Calibration framework

Wilkinson 2008/2013

We can show that ABC is "exact", but for a different model to that intended.

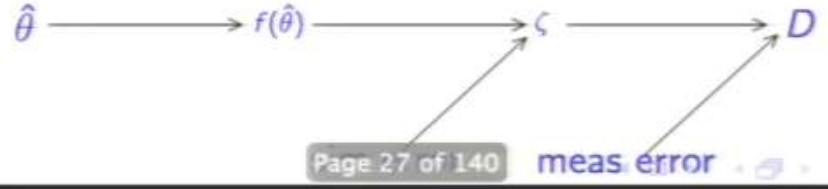
$\pi_{ABC}(D|\theta)$ is not just the simulator likelihood function:

$$\pi_{ABC}(D|\theta) = \int \pi_\epsilon(D|x)\pi(x|\theta)\mathrm{dx}$$

- $\pi_\epsilon(D|x)$ is a pdf relating the simulator output to reality - call it the *acceptance kernel*.
- $\pi(x|\theta)$ is the likelihood function of the simulator (ie not relating to reality)

Common way of thinking (Kennedy and O'Hagan 2001):

- Relate the best-simulator run ($X = f(\hat{\theta})$) to reality ζ
- Relate reality ζ to the observations D.



Calibration framework

The posterior is

$$\pi_{ABC}(\theta|D) = \frac{1}{Z}\int \pi_\epsilon(D|x)\pi(x|\theta)\mathrm{dx}.\ \pi(\theta)$$

where $Z = \iint \pi_\epsilon(D|x)\pi(x|\theta)\mathrm{dx}\pi(\theta)\mathrm{d}\theta$

Calibration framework

The posterior is

$$\pi_{ABC}(\theta|D) = \frac{1}{Z}\int \pi_\epsilon(D|x)\pi(x|\theta)\mathrm{dx}.\ \pi(\theta)$$

where $Z = \int\int \pi_\epsilon(D|x)\pi(x|\theta)\mathrm{dx}\pi(\theta)\mathrm{d}\theta$

To simplify matters, we can work in joint (θ, x) space

$$\pi_{ABC}(\theta, x|D) = \frac{\pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)}{Z}$$

NB: we can allow $\pi_\epsilon(D|X)$ to depend on θ.

How does ABC relate to calibration?

Consider how this relates to ABC:

$$\pi_{ABC}(\theta, x) := \pi(\theta, x|D) = \frac{\pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)}{Z}$$

How does ABC relate to calibration?

Consider how this relates to ABC:

$$\pi_{ABC}(\theta, x) := \pi(\theta, x|D) = \frac{\pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)}{Z}$$

Lets sample from this using the rejection algorithm with instrumental distribution

$$g(\theta, x) = \pi(x|\theta)\pi(\theta)$$

- Note: $supp(\pi_{ABC}) \subseteq supp(g)$ and that there exists a constant $M = \frac{\max_x \pi(D|X)}{Z}$ such that

$$\pi_{ABC}(\theta, x) \leq Mg(\theta, x) \qquad \forall\, (\theta, x)$$

Generalized ABC (GABC)

Wilkinson 2008, Fearnhead and Prangle 2012

The rejection algorithm then becomes

Generalized rejection ABC (Rej-GABC)

1. $\theta \sim \pi(\theta)$ and $X \sim \pi(x|\theta)$ (ie $(\theta, X) \sim g(\cdot)$)
2. Accept (θ, X) if

$$U \sim U[0,1] \leq \frac{\pi_{ABC}(\theta, x)}{Mg(\theta, x)} = \frac{\pi_\epsilon(D|X)}{\max_x \pi_\epsilon(D|x)}$$

How does ABC relate to calibration?

Consider how this relates to ABC:

$$\pi_{ABC}(\theta, x) := \pi(\theta, x|D) = \frac{\pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)}{Z}$$

Lets sample from this using the rejection algorithm with instrumental distribution

$$g(\theta, x) = \pi(x|\theta)\pi(\theta)$$

- Note: $supp(\pi_{ABC}) \subseteq supp(g)$ and that there exists a constant $M = \frac{\max_x \pi(D|X)}{Z}$ such that

$$\pi_{ABC}(\theta, x) \leq Mg(\theta, x) \qquad \forall\, (\theta, x)$$

Generalized ABC (GABC)

Wilkinson 2008, Fearnhead and Prangle 2012

The rejection algorithm then becomes

Generalized rejection ABC (Rej-GABC)

1. $\theta \sim \pi(\theta)$ and $X \sim \pi(x|\theta)$ (ie $(\theta, X) \sim g(\cdot)$)
2. Accept (θ, X) if

$$U \sim U[0,1] \leq \frac{\pi_{ABC}(\theta, x)}{Mg(\theta, x)} = \frac{\pi_\epsilon(D|X)}{\max_x \pi_\epsilon(D|x)}$$

Generalized ABC (GABC)

Wilkinson 2008, Fearnhead and Prangle 2012

The rejection algorithm then becomes

Generalized rejection ABC (Rej-GABC)

1 $\theta \sim \pi(\theta)$ and $X \sim \pi(x|\theta)$ (ie $(\theta, X) \sim g(\cdot)$)

2 Accept (θ, X) if

$$U \sim U[0,1] \leq \frac{\pi_{ABC}(\theta, x)}{Mg(\theta, x)} = \frac{\pi_\epsilon(D|X)}{\max_x \pi_\epsilon(D|x)}$$

In uniform ABC we take

$$\pi_\epsilon(D|X) = \begin{cases} 1 & \text{if } \rho(D, X) \leq \epsilon \\ 0 & \text{otherwise} \end{cases}$$

this reduces the algorithm to

2' Accept θ ifF $\rho(D, X) \leq \epsilon$

ie, we recover the *uniform* ABC algorithm.

Uniform ABC algorithm

This allows us to interpret uniform ABC. Suppose $X, D \in \mathcal{R}$

Proposition

Accepted θ from the uniform ABC algorithm (with $\rho(D, X) = |D - X|$) are samples from the posterior distribution of θ given D where we assume $D = f(\theta) + e$ and that

$$e \sim U[-\epsilon, \epsilon]$$

In general, uniform ABC assumes that

$$D|x \sim U\{d : \rho(d, x) \leq \epsilon\}$$

i.e., D is generated by adding noise uniformly chosen from a ball of radius ϵ around the best simulator output $f(\hat{\theta})$.

Generalized ABC (GABC)

Wilkinson 2008, Fearnhead and Prangle 2012

The rejection algorithm then becomes

Generalized rejection ABC (Rej-GABC)

1. $\theta \sim \pi(\theta)$ and $X \sim \pi(x|\theta)$ (ie $(\theta, X) \sim g(\cdot)$)
2. Accept (θ, X) if

$$U \sim U[0,1] \leq \frac{\pi_{ABC}(\theta, x)}{Mg(\theta, x)} = \frac{\pi_\epsilon(D|X)}{\max_x \pi_\epsilon(D|x)}$$

In uniform ABC we take

$$\pi_\epsilon(D|X) = \begin{cases} 1 & \text{if } \rho(D, X) \leq \epsilon \\ 0 & \text{otherwise} \end{cases}$$

this reduces the algorithm to

2' Accept θ ifF $\rho(D, X) \leq \epsilon$

ie, we recover the *uniform* ABC algorithm.

Uniform ABC algorithm

This allows us to interpret uniform ABC. Suppose $X, D \in \mathcal{R}$

Proposition

Accepted θ from the uniform ABC algorithm (with $\rho(D, X) = |D - X|$) are samples from the posterior distribution of θ given D where we assume $D = f(\theta) + e$ and that

$$e \sim U[-\epsilon, \epsilon]$$

In general, uniform ABC assumes that

$$D|x \sim U\{d : \rho(d, x) \leq \epsilon\}$$

i.e., D is generated by adding noise uniformly chosen from a ball of radius ϵ around the best simulator output $f(\hat{\theta})$.

Generalized ABC (GABC)

Wilkinson 2008, Fearnhead and Prangle 2012

The rejection algorithm then becomes

Generalized rejection ABC (Rej-GABC)

1. $\theta \sim \pi(\theta)$ and $X \sim \pi(x|\theta)$ (ie $(\theta, X) \sim g(\cdot)$)
2. Accept (θ, X) if

$$U \sim U[0,1] \leq \frac{\pi_{ABC}(\theta, x)}{Mg(\theta, x)} = \frac{\pi_\epsilon(D|X)}{\max_x \pi_\epsilon(D|x)}$$

In uniform ABC we take

$$\pi_\epsilon(D|X) = \begin{cases} 1 & \text{if } \rho(D, X) \leq \epsilon \\ 0 & \text{otherwise} \end{cases}$$

this reduces the algorithm to

2' Accept θ ifF $\rho(D, X) \leq \epsilon$

ie, we recover the *uniform* ABC algorithm.

Uniform ABC algorithm

This allows us to interpret uniform ABC. Suppose $X, D \in \mathcal{R}$

Proposition

Accepted θ from the uniform ABC algorithm (with $\rho(D, X) = |D - X|$) are samples from the posterior distribution of θ given D where we assume $D = f(\theta) + e$ and that

$$e \sim U[-\epsilon, \epsilon]$$

In general, uniform ABC assumes that

$$D|x \sim U\{d : \rho(d, x) \leq \epsilon\}$$

i.e., D is generated by adding noise uniformly chosen from a ball of radius ϵ around the best simulator output $f(\hat{\theta})$.

Uniform ABC algorithm

This allows us to interpret uniform ABC. Suppose $X, D \in \mathcal{R}$

Proposition

Accepted θ from the uniform ABC algorithm (with $\rho(D, X) = |D - X|$) are samples from the posterior distribution of θ given D where we assume $D = f(\theta) + e$ and that

$$e \sim U[-\epsilon, \epsilon]$$

In general, uniform ABC assumes that

$$D|x \sim U\{d : \rho(d, x) \leq \epsilon\}$$

i.e., D is generated by adding noise uniformly chosen from a ball of radius ϵ around the best simulator output $f(\hat{\theta})$.

ABC gives 'exact' inference under a different model!

Acceptance Kernel - $\pi(D|x)$

Kennedy and O'Hagan 2001, Goldstein and Rougier 2009

How do we relate the simulator to reality?

1. Measurement error - $D = \zeta + e$ - let $\pi_\epsilon(D|X)$ be the distribution e.
2. Model error - $\zeta = f(\theta) + \delta$ - let $\pi_\epsilon(D|X)$ be the distribution ϵ.

   Or both: $\pi_\epsilon(D|x)$ a convolution of the two distributions
3. Sampling a hidden space - often the data D are noisy observations of some latent feature (call it X), which is generated by a stochastic process. By removing the stochastic sampling from the simulator we can let $\pi(D|x)$ do the sampling for us (Rao-Blackwellisation).

Kernel Smoothing

Blum 2010, Fearnhead and Prangle 2012

Viewing ABC as an extension of modelling isn't commonly done.

- allows us to do the inference we want (and to interpret)
  - - makes explicit the relationship between simulator and observations.
- allows for the possibility of more efficient ABC algorithms

Kernel Smoothing

Blum 2010, Fearnhead and Prangle 2012

Viewing ABC as an extension of modelling isn't commonly done.

- allows us to do the inference we want (and to interpret)
  - - makes explicit the relationship between simulator and observations.
- allows for the possibility of more efficient ABC algorithms

A different but equivalent view of ABC is as kernel smoothing

$$\pi_{ABC}(\theta|D) \propto \int K_\epsilon(D-x)\pi(x|\theta)\pi(\theta)\mathrm{d}x$$

where $K_\epsilon(x) = 1/\epsilon K(x/\epsilon)$ and K is a standard kernel and ϵ is the bandwidth.

Efficient Algorithms

References:

- Marjoram *et al.* 2003
- Sisson *et al.* 2007
- Beaumont *et al.* 2008
- Toni *et al.* 2009
- Del Moral *et al.* 2011
- Drovandi *et al.* 2011

ABCifying Monte Carlo methods

Rejection ABC is the basic ABC algorithm.

- Inefficient as it repeatedly samples from prior

A large number of papers have been published turning other MC algorithms into ABC type algorithms for when we don't know the likelihood: IS, MCMC, SMC, EM, EP etc

Focus on MCMC and SMC

- presented for GABC with acceptance kernels, but most the algorithms were written down for uniform ABC, i.e.,

$$\pi_\epsilon(D|X) = \mathbb{I}_{\rho(D,X)\leq\epsilon}$$

and we can make this choice in most cases if desired.

MCMC-ABC

Marjoram *et al.* 2003

We are targeting the joint distribution

$$\pi_{ABC}(\theta, x|D) \propto \pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)$$

To explore the (θ, x) space, proposals of the form

$$Q((\theta, x), (\theta', x')) = q(\theta, \theta')\pi(x'|\theta')$$

seem to be inevitable (q arbitrary).

The Metropolis-Hastings (MH) acceptance probability is then

$$r = \frac{\pi_{ABC}(\theta'|D)Q((\theta', x'), (\theta, x))}{\pi_{ABC}(\theta|D)Q((\theta, x), (\theta', x'))}$$

ABCifying Monte Carlo methods

Rejection ABC is the basic ABC algorithm.

- Inefficient as it repeatedly samples from prior

A large number of papers have been published turning other MC algorithms into ABC type algorithms for when we don't know the likelihood: IS, MCMC, SMC, EM, EP etc

Focus on MCMC and SMC

- presented for GABC with acceptance kernels, but most the algorithms were written down for uniform ABC, i.e.,

$$\pi_\epsilon(D|X) = \mathbb{I}_{\rho(D,X)\leq\epsilon}$$

and we can make this choice in most cases if desired.

MCMC-ABC

Marjoram *et al.* 2003

We are targeting the joint distribution

$$\pi_{ABC}(\theta, x|D) \propto \pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)$$

To explore the (θ, x) space, proposals of the form

$$Q((\theta, x), (\theta', x')) = q(\theta, \theta')\pi(x'|\theta')$$

seem to be inevitable (q arbitrary).

The Metropolis-Hastings (MH) acceptance probability is then

$$r = \frac{\pi_{ABC}(\theta'|D)Q((\theta', x'), (\theta, x))}{\pi_{ABC}(\theta|D)Q((\theta, x), (\theta', x'))}$$

This gives the following MCMC kernel

MH-ABC - $P_{Marj}(\theta_0, \cdot)$

1. Propose a move from $z_t = (\theta, x)$ to (θ', x') using proposal Q above.
2. Accept move with probability

$$r((\theta, x), (\theta', x')) = \min\left(1, \frac{\pi_\epsilon(D|x')q(\theta', \theta)\pi(\theta')}{\pi_\epsilon(D|x)q(\theta, \theta')\pi(\theta)}\right),$$

otherwise set $z_{t+1} = z_t$.

This gives the following MCMC kernel

MH-ABC - $P_{Marj}(\theta_0, \cdot)$

1. Propose a move from $z_t = (\theta, x)$ to (θ', x') using proposal Q above.
2. Accept move with probability

$$r((\theta, x), (\theta', x')) = \min\left(1, \frac{\pi_\epsilon(D|x')q(\theta', \theta)\pi(\theta')}{\pi_\epsilon(D|x)q(\theta, \theta')\pi(\theta)}\right),$$

otherwise set $z_{t+1} = z_t$.

In practice, we find this algorithm often gets stuck at a given θ, as the probability of generating x' near D can be tiny if ϵ is small.

Note that this is a special case of a "pseudo marginal" Metropolis-Hastings algorithm, and can be modified to use multiple simulations at each θ, i.e.

$$r = \min\left(1, \frac{\sum_{i=1}^N \pi_\epsilon(D|x_i')q(\theta', \theta)\pi(\theta')}{\sum_{i=1}^N \pi_\epsilon(D|x_i)q(\theta, \theta')\pi(\theta)}\right)$$

to better approximate the likelih

Recent developments - Lee 2012

1-hit MCMC kernel - $P_{1hit}(\theta_0, \cdot)$

1. Propose $\theta' \sim q(\theta_t, \cdot)$
2. With probability

$$1 - \min\left(1, \frac{q(\theta', \theta_t)\pi(\theta')}{q(\theta_t, \theta')\pi(\theta_t)}\right)$$

   set $\theta_{t+1} = \theta_t$
3. Sample $x' \sim \pi(\cdot|\theta')$ and $x \sim \pi(\cdot|\theta_t)$ until $\rho(x', D) \leq \epsilon$ or $\rho(x, D) \leq \epsilon$.
4. If $\rho(x', D) \leq \epsilon$ set $\theta_{t+1} = \theta'$ otherwise set $\theta_{t+1} = \theta_t$

This gives the following MCMC kernel

MH-ABC - $P_{Marj}(\theta_0, \cdot)$

1. Propose a move from $z_t = (\theta, x)$ to (θ', x') using proposal Q above.
2. Accept move with probability

$$r((\theta, x), (\theta', x')) = \min\left(1, \frac{\pi_\epsilon(D|x')q(\theta', \theta)\pi(\theta')}{\pi_\epsilon(D|x)q(\theta, \theta')\pi(\theta)}\right),$$

otherwise set $z_{t+1} = z_t$.

MCMC-ABC

Marjoram *et al.* 2003

We are targeting the joint distribution

$$\pi_{ABC}(\theta, x|D) \propto \pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)$$

To explore the (θ, x) space, proposals of the form

$$Q((\theta, x), (\theta', x')) = q(\theta, \theta')\pi(x'|\theta')$$

seem to be inevitable (q arbitrary).

The Metropolis-Hastings (MH) acceptance probability is then

$$\begin{aligned} r &= \frac{\pi_{ABC}(\theta'|D)Q((\theta', x'), (\theta, x))}{\pi_{ABC}(\theta|D)Q((\theta, x), (\theta', x'))} \\ &= \frac{\pi_\epsilon(D|x')\pi(x'|\theta')\pi(\theta')q(\theta', \theta)\pi(x|\theta)}{\pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)q(\theta, \theta')\pi(x'|\theta')} \\ &= \frac{\pi_\epsilon(D|x')q(\theta', \theta)\pi(\theta')}{\pi_\epsilon(D|x)q(\theta, \theta')\pi(\theta)} \end{aligned}$$

This gives the following MCMC kernel

MH-ABC - $P_{Marj}(\theta_0, \cdot)$

1. Propose a move from $z_t = (\theta, x)$ to (θ', x') using proposal Q above.
2. Accept move with probability

$$r((\theta, x), (\theta', x')) = \min\left(1, \frac{\pi_\epsilon(D|x')q(\theta', \theta)\pi(\theta')}{\pi_\epsilon(D|x)q(\theta, \theta')\pi(\theta)}\right),$$

otherwise set $z_{t+1} = z_t$.

MCMC-ABC

Marjoram *et al.* 2003

We are targeting the joint distribution

$$\pi_{ABC}(\theta, x|D) \propto \pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)$$

To explore the (θ, x) space, proposals of the form

$$Q((\theta, x), (\theta', x')) = q(\theta, \theta')\pi(x'|\theta')$$

seem to be inevitable (q arbitrary).

The Metropolis-Hastings (MH) acceptance probability is then

$$\begin{aligned} r &= \frac{\pi_{ABC}(\theta'|D)Q((\theta', x'), (\theta, x))}{\pi_{ABC}(\theta|D)Q((\theta, x), (\theta', x'))} \\ &= \frac{\pi_\epsilon(D|x')\pi(x'|\theta')\pi(\theta')q(\theta', \theta)\pi(x|\theta)}{\pi_\epsilon(D|x)\pi(x|\theta)\pi(\theta)q(\theta, \theta')\pi(x'|\theta')} \end{aligned}$$

This gives the following MCMC kernel

MH-ABC - $P_{Marj}(\theta_0, \cdot)$

1. Propose a move from $z_t = (\theta, x)$ to (θ', x') using proposal Q above.
2. Accept move with probability

$$r((\theta, x), (\theta', x')) = \min\left(1, \frac{\pi_\epsilon(D|x')q(\theta', \theta)\pi(\theta')}{\pi_\epsilon(D|x)q(\theta, \theta')\pi(\theta)}\right),$$

otherwise set $z_{t+1} = z_t$.

Recent developments - Lee 2012

1-hit MCMC kernel - $P_{1hit}(\theta_0, \cdot)$

1. Propose $\theta' \sim q(\theta_t, \cdot)$
2. With probability

$$1 - \min\left(1, \frac{q(\theta', \theta_t)\pi(\theta')}{q(\theta_t, \theta')\pi(\theta_t)}\right)$$

   set $\theta_{t+1} = \theta_t$
3. Sample $x' \sim \pi(\cdot|\theta')$ and $x \sim \pi(\cdot|\theta_t)$ until $\rho(x', D) \leq \epsilon$ or $\rho(x, D) \leq \epsilon$.
4. If $\rho(x', D) \leq \epsilon$ set $\theta_{t+1} = \theta'$ otherwise set $\theta_{t+1} = \theta_t$

Recent developments

Lee *et al.* 2013 showed P_{Marj} is neither

- variance bounding
  - Let $\widehat{\mathbb{E}h(\theta)} = \frac{1}{m}\sum h(\theta_i)$ - Markov kernel P is variance bounding if $\mathbb{V}\text{ar}_P(\widehat{\mathbb{E}h(\theta)})$ is "reasonably small"
- nor geometrically ergodic (GE) i.e $||P^m(\theta_0, \cdot) - \pi_{ABC}(\cdot)||_{TV} \leq C\rho^m$ where $\rho < 1$. Markov kernels that are not GE may convergence extremely slowly.

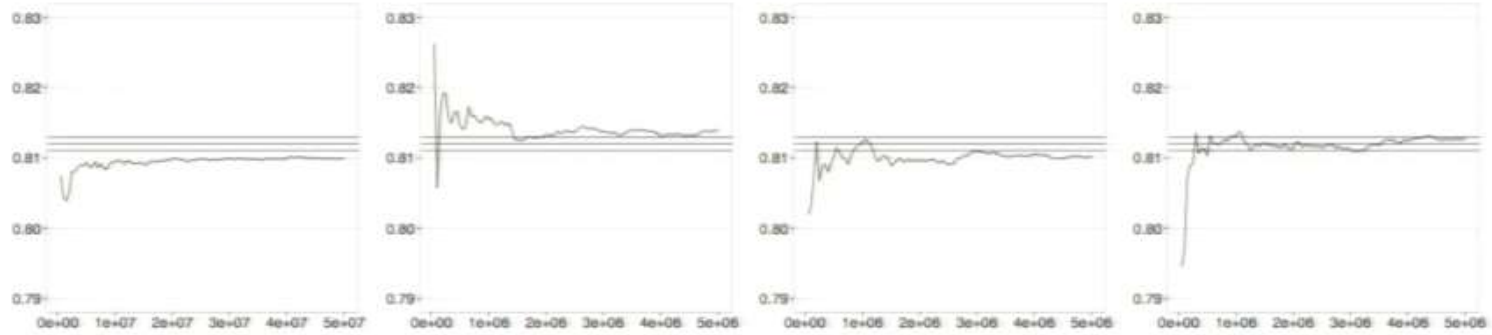
whereas P_{1hit} is (subject to conditions).

Recent developments

Lee *et al.* 2013 showed P_{Marj} is neither

- variance bounding
  - Let $\widehat{\mathbb{E}h(\theta)} = \frac{1}{m}\sum h(\theta_i)$ - Markov kernel P is variance bounding if $\mathbb{V}\text{ar}_P(\widehat{\mathbb{E}h(\theta)})$ is "reasonably small"
- nor geometrically ergodic (GE) i.e $||P^m(\theta_0, \cdot) - \pi_{ABC}(\cdot)||_{TV} \leq C\rho^m$ where $\rho < 1$. Markov kernels that are not GE may convergence extremely slowly.

whereas P_{1hit} is (subject to conditions).



(a) $P_{1,1}$ (b) $P_{1,15}$ (c) $P_{2,15}$ (d) P_3

Note that P_{1hit} requires significantly more computation per iteration (but this may be worth it)

Recent developments - Lee 2012

1-hit MCMC kernel - $P_{1hit}(\theta_0, \cdot)$

1. Propose $\theta' \sim q(\theta_t, \cdot)$
2. With probability

$$1 - \min\left(1, \frac{q(\theta', \theta_t)\pi(\theta')}{q(\theta_t, \theta')\pi(\theta_t)}\right)$$

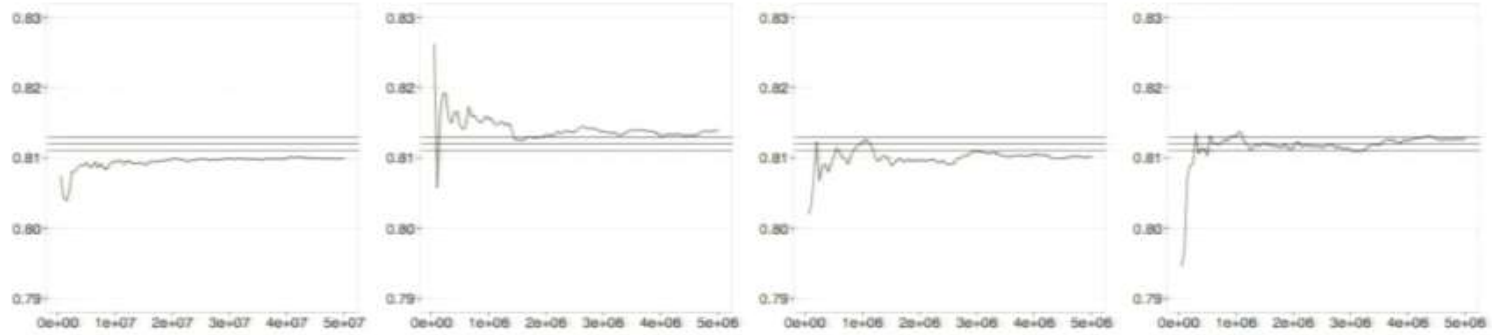
   set $\theta_{t+1} = \theta_t$
3. Sample $x' \sim \pi(\cdot|\theta')$ and $x \sim \pi(\cdot|\theta_t)$ until $\rho(x', D) \leq \epsilon$ or $\rho(x, D) \leq \epsilon$.
4. If $\rho(x', D) \leq \epsilon$ set $\theta_{t+1} = \theta'$ otherwise set $\theta_{t+1} = \theta_t$

Recent developments

Lee *et al.* 2013 showed P_{Marj} is neither

- variance bounding
  - Let $\widehat{\mathbb{E}h(\theta)} = \frac{1}{m}\sum h(\theta_i)$ - Markov kernel P is variance bounding if $\mathbb{V}\text{ar}_P(\widehat{\mathbb{E}h(\theta)})$ is "reasonably small"
- nor geometrically ergodic (GE) i.e $||P^m(\theta_0, \cdot) - \pi_{ABC}(\cdot)||_{TV} \leq C\rho^m$ where $\rho < 1$. Markov kernels that are not GE may convergence extremely slowly.

whereas P_{1hit} is (subject to conditions).



(a) $P_{1,1}$ (b) $P_{1,15}$ (c) $P_{2,15}$ (d) P_3

Note that P_{1hit} requires significantly more computation per iteration (but this may be worth it)

Importance sampling GABC

In uniform ABC, importance sampling simply reduces to the rejection algorithm with a fixed budget for the number of simulator runs.

But for GABC it opens new algorithms:

GABC - Importance sampling

1. $\theta_i \sim \pi(\theta)$ and $X_i \sim \pi(x|\theta_i)$.
2. Give (θ_i, x_i) weight $w_i = \pi_\epsilon(D|x_i)$.

Importance sampling GABC

In uniform ABC, importance sampling simply reduces to the rejection algorithm with a fixed budget for the number of simulator runs.

But for GABC it opens new algorithms:

GABC - Importance sampling

1. $\theta_i \sim \pi(\theta)$ and $X_i \sim \pi(x|\theta_i)$.
2. Give (θ_i, x_i) weight $w_i = \pi_\epsilon(D|x_i)$.

Which is more efficient - IS-GABC or Rej-GABC?

Proposition 2

IS-GABC has a larger effective sample size than Rej-GABC, or equivalently

$$\mathbb{V}\mathrm{ar}_{\mathrm{Rej}}(w) \geq \mathbb{V}\mathrm{ar}_{\mathrm{IS}}(w)$$

This can be seen as a Rao-Blackwell type result.

Rejection Control (RC)

A difficulty with IS algorithms is that they can require the storage of a large number of particles with small weights.

- thin particles with small weights using rejection control:

Rejection Control in IS-GABC

1. $\theta_i \sim \pi(\theta)$ and $X_i \sim \pi(X|\theta_i)$
2. Accept (θ_i, X_i) with probability

$$r(X_i) = \min\left(1, \frac{\pi_\epsilon(D|X_i)}{C}\right)$$

   for any threshold constant $C \geq 0$.
3. Give accepted particles weights

$$w_i = \max(\pi_\epsilon(D|X_i), C)$$

IS is more efficient than RC, unless we have memory constraints (relative to processor time).

Importance sampling GABC

In uniform ABC, importance sampling simply reduces to the rejection algorithm with a fixed budget for the number of simulator runs.

But for GABC it opens new algorithms:

GABC - Importance sampling

1. $\theta_i \sim \pi(\theta)$ and $X_i \sim \pi(x|\theta_i)$.
2. Give (θ_i, x_i) weight $w_i = \pi_\epsilon(D|x_i)$.

Which is more efficient - IS-GABC or Rej-GABC?

Proposition 2

IS-GABC has a larger effective sample size than Rej-GABC, or equivalently

$$\mathbb{V}\mathrm{ar}_{\mathrm{Rej}}(w) \geq \mathbb{V}\mathrm{ar}_{\mathrm{IS}}(w)$$

This can be seen as a Rao-Blackwell type result.

Rejection Control (RC)

A difficulty with IS algorithms is that they can require the storage of a large number of particles with small weights.

- thin particles with small weights using rejection control:

Rejection Control in IS-GABC

1. $\theta_i \sim \pi(\theta)$ and $X_i \sim \pi(X|\theta_i)$
2. Accept (θ_i, X_i) with probability

$$r(X_i) = \min\left(1, \frac{\pi_\epsilon(D|X_i)}{C}\right)$$

   for any threshold constant $C \geq 0$.
3. Give accepted particles weights

$$w_i = \max(\pi_\epsilon(D|X_i), C)$$

IS is more efficient than RC, unless we have memory constraints (relative to processor time).

Sequential ABC algorithms

The most popular efficient ABC algorithms are those based on sequential methods (Sisson *et al.* 2007, Toni *et al.* 2008, Beaumont *et al.* 2009, ...).

We aim to sample N particles successively from a sequence of distributions

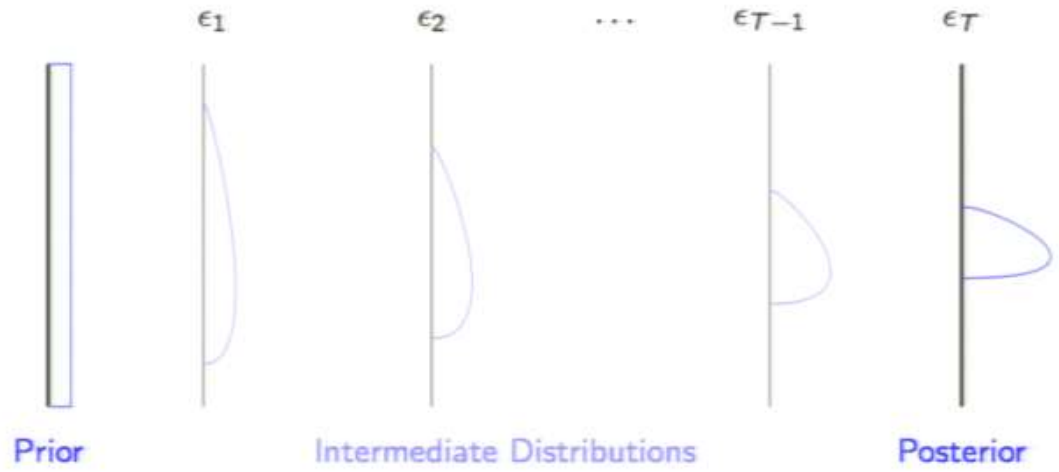
$$\pi_1(\theta), \ldots, \pi_T(\theta) = \text{target}$$

For ABC we decide upon a sequence of tolerances $\epsilon_1 > \epsilon_2 > \ldots > \epsilon_T$ and let π_t be the ABC distribution found by the ABC algorithm when we use tolerance ϵ_t.

Specifically, define a sequence of target distributions

$$\pi_t(\theta, x) = \frac{\pi_t(D|x)\pi(x|\theta)\pi(\theta)}{C_t} = \frac{\gamma_t(\theta, x)}{C_t}$$

with $\pi_t(D|X) = \pi_{\epsilon_t}(D|X)$

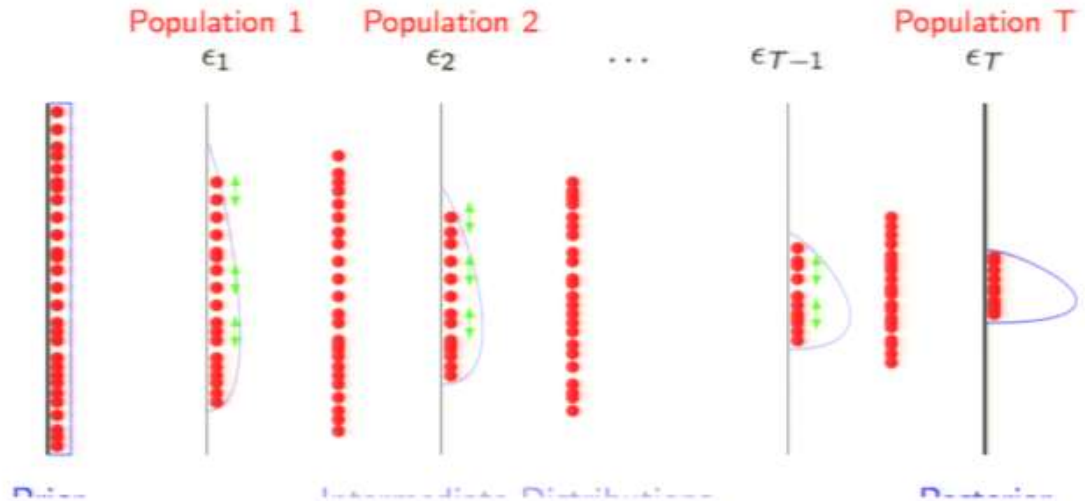


Picture from Toni and Stumpf 2010 tutorial

At each stage t, we aim to construct a weighted sample of particles that approximates $\pi_t(\theta, x)$.

$$\left\{\left(z_t^{(i)}, W_t^{(i)}\right)\right\}_{i=1}^{N} \text{ such that } \pi_t(z) \approx \sum W_t^{(i)} \delta_{z_t^{(i)}}(\mathrm{dz})$$

where $z_t^{(i)} = (\theta_t^{(i)}, x_t^{(i)})$.

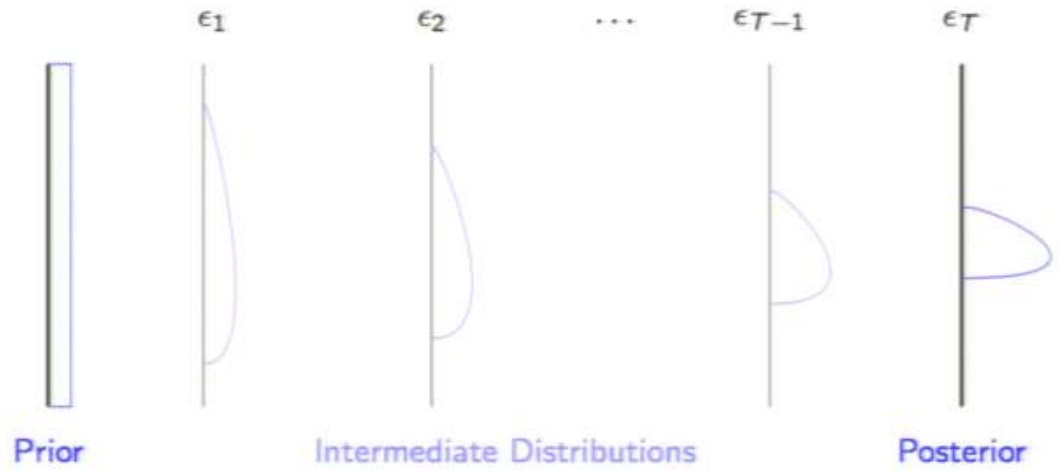


Picture from Toni and Stumpf 2010 tutorial

Specifically, define a sequence of target distributions

$$\pi_t(\theta, x) = \frac{\pi_t(D|x)\pi(x|\theta)\pi(\theta)}{C_t} = \frac{\gamma_t(\theta, x)}{C_t}$$

with $\pi_t(D|X) = \pi_{\epsilon_t}(D|X)$

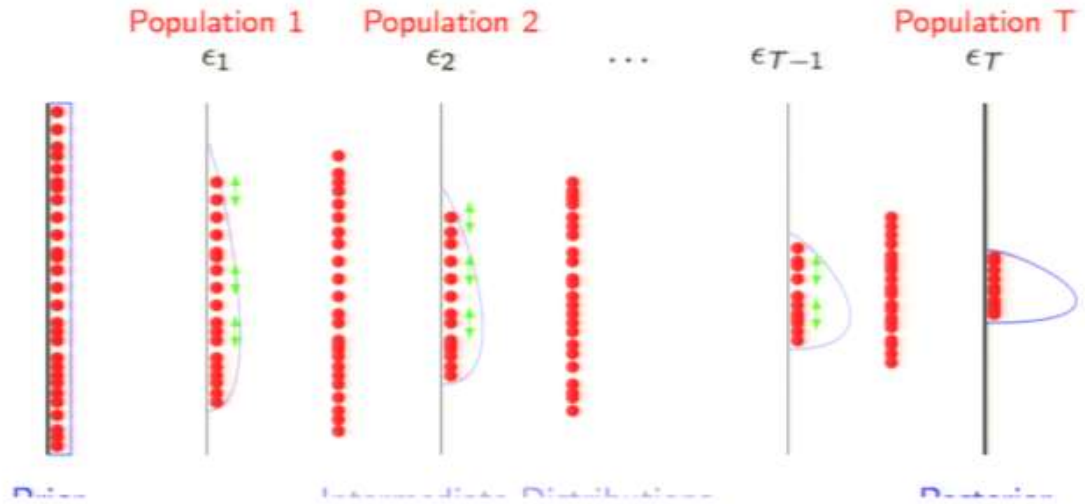


Picture from Toni and Stumpf 2010 tutorial

At each stage t, we aim to construct a weighted sample of particles that approximates $\pi_t(\theta, x)$.

$$\left\{\left(z_t^{(i)}, W_t^{(i)}\right)\right\}_{i=1}^{N} \text{ such that } \pi_t(z) \approx \sum W_t^{(i)} \delta_{z_t^{(i)}}(\mathrm{dz})$$

where $z_t^{(i)} = (\theta_t^{(i)}, x_t^{(i)})$.



Picture from Toni and Stumpf 2010 tutorial

Toni *et al.* (2008)

Assume we have a cloud of weighted particles $\{(\theta_i, w_i)\}_{i=1}^{N}$ that were accepted at step $t-1$.

1. Sample θ from the previous population according to the weights.
2. Perturb the particles according to perturbation kernel q_t. I.e.,

$$\tilde{\theta} \sim q_t(\theta, \cdot)$$

3. Reject particle immediately if $\tilde{\theta}$ has zero prior density, i.e., if

$$\pi(\tilde{\theta}) = 0$$

4. Otherwise simulate $X \sim f(\tilde{\theta})$ from the simulator. If $\rho(S(X), S(D)) \leq \epsilon_t$ accept the particle, otherwise reject.
5. Give the accepted particle weight

$$w_i = \frac{\pi(\tilde{\theta})}{\sum_{\theta_i} q_t(\theta_i, \tilde{\theta})}$$

6. Repeat steps 1-5 until we have N accepted particles at step t.

Sequential Monte Carlo (SMC)

All the SMC-ABC algorithms can be understood as special cases of Del Moral *et al.* 2006.

If at stage t we use proposal distribution $\eta_t(z)$ for the particles, then we create the weighted sample as follows:

Generic Sequential Monte Carlo - stage n

(i) For $i = 1, \ldots, N$

$$Z_t^{(i)} \sim \eta_t(z)$$

and correct between η_t and π_t

$$w_t(Z_t^{(i)}) = \frac{\gamma_t(Z_t^{(i)})}{\eta_t(Z_t^{(i)})}$$

(ii) Normalize to find weights $\{W_t^{(i)}\}$.

(iii) If effective sample size (ESS) is less than some threshold T, resample the particles and set $W_t^{(i)} = 1/N$. Set $t = t + 1$.

Toni *et al.* (2008)

Assume we have a cloud of weighted particles $\{(\theta_i, w_i)\}_{i=1}^{N}$ that were accepted at step $t-1$.

1. Sample θ from the previous population according to the weights.
2. Perturb the particles according to perturbation kernel q_t. I.e.,

$$\tilde{\theta} \sim q_t(\theta, \cdot)$$

3. Reject particle immediately if $\tilde{\theta}$ has zero prior density, i.e., if

$$\pi(\tilde{\theta}) = 0$$

4. Otherwise simulate $X \sim f(\tilde{\theta})$ from the simulator. If $\rho(S(X), S(D)) \leq \epsilon_t$ accept the particle, otherwise reject.
5. Give the accepted particle weight

$$w_i = \frac{\pi(\tilde{\theta})}{\sum_{\theta_i} q_t(\theta_i, \tilde{\theta})}$$

6. Repeat steps 1-5 until we have N accepted particles at step t.

Sequential Monte Carlo (SMC)

All the SMC-ABC algorithms can be understood as special cases of Del Moral *et al.* 2006.

If at stage t we use proposal distribution $\eta_t(z)$ for the particles, then we create the weighted sample as follows:

Generic Sequential Monte Carlo - stage n

(i) For $i = 1, \ldots, N$

$$Z_t^{(i)} \sim \eta_t(z)$$

and correct between η_t and π_t

$$w_t(Z_t^{(i)}) = \frac{\gamma_t(Z_t^{(i)})}{\eta_t(Z_t^{(i)})}$$

(ii) Normalize to find weights $\{W_t^{(i)}\}$.

(iii) If effective sample size (ESS) is less than some threshold T, resample the particles and set $W_t^{(i)} = 1/N$. Set $t = t + 1$.

Del Moral *et al.* SMC algorithm

We can build the proposal distribution $\eta_t(z)$, from the particles available at time $t-1$.

One way to do this is to propose new particles by passing the old particles through a Markov kernel $q_t(z, z')$.

- For $i = 1, \ldots, N$

$$z_n^{(i)} \sim q_t(z_{t-1}^{(i)}, \cdot)$$

This makes $\eta_t(z) = \int \eta_{t-1}(z') q_t(z', z) \mathrm{d}z'$ – which is unknown in general.

Del Moral *et al.* 2006 showed how to avoid this problem by introducing a sequence of backward kernels, L_{t-1}.

GABC versions of SMC

We need to choose

- Sequence of targets π_t
- Forward perturbation kernels K_t
- Backward kernels L_t
- Thresholds c_t.

By making particular choices for these quantities we can recover many of the published SMC-ABC samplers.

Other sequential GABC algorithms

We can combine SMC with MCMC type moves, by using

$$L_{t-1}(z_t, z_{t-1}) = \frac{\pi_{t-1}(z_{t-1})Q_t(z_{t-1}, z_t)}{\pi_{t-1}Q_t(z_t)}$$

If we then use a π_t invariant Metropolis-Hastings kernel Q_t and let

$$L_{t-1}(z_t, z_{t-1}) = \frac{\pi_t(z_{t-1})Q_t(z_{t-1}, z_t)}{\pi_t(z_t)}$$

then we get an ABC resample-move algorithm.

Approximate Resample-Move (with PRC)

RM-GABC

(i) While $ESS < N$

(a) Sample $z^* = (\theta^*, X^*)$ from $\{z_{t-1}^{(i)}\}$ according to weights $W_{t-1}^{(i)}$.

(b) Weight:

$$w^* = \tilde{w}_t(X^*) = \frac{\pi_t(D|X^*)}{\pi_{t-1}(D|X^*)}$$

(c) PRC: With probability $\min(1, \frac{w^*}{c_t})$, sample

$$z_t^{(i)} \sim Q_t(z^*, \cdot)$$

where Q_t is an MCMC kernel with invariant distribution π_t. Set $i = i + 1$.
Otherwise, return to (i)(a).

(ii) Normalise the weights to get $W_t^{(i)}$. Set $n = n + 1$

Note that because the incremental weights are independent of z_t we are able to swap the perturbation and PRC steps.

Conclusions

- The tolerance ϵ controls the accuracy of ABC algorithms, and so we desire to take ϵ as small as possible in many problems (although not always).
- By using efficient sampling algorithms, we can hope to better use the available computation resource to spend more time simulating in regions of parameter space likely to lead to accepted values
- MCMC and SMC versions of ABC have been developed, along with ABC versions of most other algorithms.

Links to other approaches

History-matching

e.g. Craig *et al.* 2001, Vernon *et al.* 2010

ABC can be seen as a probabilistic version of history matching. History matching is used in the analysis of computer experiments to rule out regions of space as implausible.

1. Relate the simulator to the system

$$\zeta = f(\theta) + \epsilon$$

where ϵ is our simulator discrepancy

2. Relate the system to the data (e represents measurement error)

$$D = \zeta + e$$

3. Declare θ implausible if, e.g.,

$$\| D - \mathbb{E}f(\theta) \| > 3\sigma$$

where σ^2 is the combined variance implied by the emulator, discrepancy and measurement error.

History-matching

If θ is not implausible we don't discard it. The result is a region of space that we can't rule out at this stage of the history-match.

Usual to go through several stages of history matching.
Notes:

- History matching can be seen as a principled version of ABC - lots of thought goes into the link between simulator and reality.
- The result of history-matching may be that there is no not-implausible region of parameter space
  - Go away and think harder - something is misspecified
  - This can also happen in rejection ABC.
  - In contrast, MCMC will always give an answer, even if the model is terrible.

History-matching

e.g. Craig *et al.* 2001, Vernon *et al.* 2010

ABC can be seen as a probabilistic version of history matching. History matching is used in the analysis of computer experiments to rule out regions of space as implausible.

1. Relate the simulator to the system

$$\zeta = f(\theta) + \epsilon$$

where ϵ is our simulator discrepancy

2. Relate the system to the data (e represents measurement error)

$$D = \zeta + e$$

3. Declare θ implausible if, e.g.,

$$\| D - \mathbb{E}f(\theta) \| > 3\sigma$$

where σ^2 is the combined variance implied by the emulator, discrepancy and measurement error.

History-matching

If θ is not implausible we don't discard it. The result is a region of space that we can't rule out at this stage of the history-match.

Usual to go through several stages of history matching.
Notes:

- History matching can be seen as a principled version of ABC - lots of thought goes into the link between simulator and reality.
- The result of history-matching may be that there is no not-implausible region of parameter space
  - Go away and think harder - something is misspecified
  - This can also happen in rejection ABC.
  - In contrast, MCMC will always give an answer, even if the model is terrible.

Noisy-ABC

Fearnhead and Prangle (2012) proposed the noisy-ABC algorithm:

Noisy-ABC

Initialise: Let $D' = D + e$ where $e \sim K(e)$ from some kernel $K(\cdot)$.

1. $\theta_i \sim \pi(\theta)$ and $X_i \sim \pi(x|\theta_i)$.
2. Give (θ_i, x_i) weight $w_i = K(X_i - D')$.

In our notation, replace the observed data D, with D' drawn from the acceptance kernel - $D' \sim \pi(D'|D)$

Noisy ABC

Noisy ABC is well calibrated. However, this is a frequency property, and so it only becomes relevant if we repeat the analysis with different D' many times

- highly relevant to filtering problems

Note that noisy ABC and GABC are trying to do different things:

- Noisy ABC moves the data so that it comes from the model we are assuming when we do inference.
  - Assumes the model $\pi(D|\theta)$ is true and tries to find the true posterior given the noisy data.
- GABC accepts the model is incorrect, and tries to account for this in the inference.

Other algorithms

- Wood 2010 is an ABC algorithm, but using sample mean μ_θ and covariance Σ_θ of the summary of $f(\theta)$ run n times at θ, and assuming

$$\pi(D|S) = \mathcal{N}(D; \mu_\theta, \Sigma_\theta)$$

- (Generalized Likelihood Uncertainty Estimation) GLUE approach of Keith Beven in hydrology - see Nott and Marshall 2012
- Kalman filtering, see Nott *et al.* 2012.

The dangers of ABC - H.L. Mencken

For every complex problem, there is an answer that is short, simple and wrong

Why use ABC? J. Galsworthy

Idealism increases in direct proportion to ones distance from the problem

Recap I

Uniform Rejection ABC

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

We've looked at a variety of more efficient sampling algorithms

- e.g. ABC-MCMC, ABC-IS, ABC-SMC
- The higher the efficiency the smaller the tolerance we can use for a given computational expense.

Recap II

Alternative approaches focus on avoiding the curse of dimensionality:

- If the data are too high dimensional we never observe simulations that are 'close' to the field data

Recap II

Alternative approaches focus on avoiding the curse of dimensionality:

- If the data are too high dimensional we never observe simulations that are 'close' to the field data

Approaches include

- Using summary statistics $S(D)$ to reduce the dimension

Uniform rejection ABC with summaries

Draw θ from $\pi(\theta)$

Simulate $X \sim f(\theta)$

Accept θ if $\rho(S(D), S(X)) < \epsilon$

If S is sufficient this is equivalent to the previous algorithm.

- Regression adjustment - model and account for the discrepancy between $S = S(X)$ and $S_{obs} = S(D)$.

Regression Adjustment

References:

- Beaumont *et al.* 2003
- Blum and Francois 2010
- Blum 2010
- Leuenberger and Wegmann 2010

Regression Adjustment

An alternative to rejection-ABC, proposed by Beaumont *et al.* 2002, uses post-hoc adjustment of the parameter values to try to weaken the effect of the discrepancy between s and s_{obs}.

Two key ideas

- use non-parametric kernel density estimation to emphasise the best simulations
- learn a non-linear model for the conditional expectation $\mathbb{E}(\theta|s)$ as a function of s and use this to learn the posterior at s_{obs}.

Idea 1 - kernel regression

Suppose we want to estimate

$$\mathbb{E}(\theta|s_{obs}) = \int \frac{\theta\pi(\theta, s_{obs})}{\pi(s_{obs})}\mathrm{d}\theta$$

using pairs $\{\theta_i, s_i\}$ from $\pi(\theta, s)$

Idea 1 - kernel regression

Suppose we want to estimate

$$\mathbb{E}(\theta|s_{obs}) = \int \frac{\theta\pi(\theta, s_{obs})}{\pi(s_{obs})}\mathrm{d}\theta$$

using pairs $\{\theta_i, s_i\}$ from $\pi(\theta, s)$

Approximating the two densities using a kernel density estimate

$$\hat{\pi}(\theta, s) = \frac{1}{n}\sum_i K(s - s_i)K(\theta - \theta_i) \qquad \hat{\pi}(s) = \frac{1}{n}\sum_i K(s - s_i)$$

and substituting gives the Naradaya-Watson estimator:

$$\mathbb{E}(\theta|s_{obs}) \approx \frac{\sum_i K(s_{obs} - s_i)\theta_i}{\sum_i K(s_{obs} - s_i)}$$

as $\int yK(y - a)\mathrm{d}y = a$.

- Beaumont *et al.* 2002 suggested using the Epanechnikov kernel

$$K_\epsilon(x) = \frac{c}{\epsilon}\left[1 - \left(\frac{x}{\epsilon}\right)^2\right]\mathbb{I}_{x \leq \epsilon}$$

as it has finite support - we discard the majority of simulations. They recommend ϵ be set by deciding on the proportion of simulations to keep e.g. best 5%

- This expression also arises if we view

$$\{\theta_i, W_i\}, \quad \text{with } W_i = K_\epsilon(s_{obs} - s_i) \equiv \pi_\epsilon(s_{obs}|s_i)$$

as a weighted particle approximation to the posterior

$$\pi(\theta|s_{obs}) = \sum w_i \delta_{\theta_i}(\theta)$$

where $w_i = W_i / \sum W_j$ are normalised weights

- The Naradaya-Watson estimator suffers from the curse of dimensionality - its rate of convergence drops rapidly as the dimension of s increases.

Idea 1 - kernel regression

Suppose we want to estimate

$$\mathbb{E}(\theta|s_{obs}) = \int \frac{\theta\pi(\theta, s_{obs})}{\pi(s_{obs})}\mathrm{d}\theta$$

using pairs $\{\theta_i, s_i\}$ from $\pi(\theta, s)$

Approximating the two densities using a kernel density estimate

$$\hat{\pi}(\theta, s) = \frac{1}{n}\sum_i K(s - s_i)K(\theta - \theta_i) \qquad \hat{\pi}(s) = \frac{1}{n}\sum_i K(s - s_i)$$

and substituting gives the Naradaya-Watson estimator:

$$\mathbb{E}(\theta|s_{obs}) \approx \frac{\sum_i K(s_{obs} - s_i)\theta_i}{\sum_i K(s_{obs} - s_i)}$$

as $\int yK(y - a)\mathrm{d}y = a$.

- Beaumont *et al.* 2002 suggested using the Epanechnikov kernel

$$K_\epsilon(x) = \frac{c}{\epsilon}\left[1 - \left(\frac{x}{\epsilon}\right)^2\right]\mathbb{I}_{x \leq \epsilon}$$

as it has finite support - we discard the majority of simulations. They recommend ϵ be set by deciding on the proportion of simulations to keep e.g. best 5%

- This expression also arises if we view

$$\{\theta_i, W_i\}, \quad \text{with } W_i = K_\epsilon(s_{obs} - s_i) \equiv \pi_\epsilon(s_{obs}|s_i)$$

as a weighted particle approximation to the posterior

$$\pi(\theta|s_{obs}) = \sum w_i \delta_{\theta_i}(\theta)$$

where $w_i = W_i / \sum W_j$ are normalised weights

- The Naradaya-Watson estimator suffers from the curse of dimensionality - its rate of convergence drops rapidly as the dimension of s increases.

Idea 2 - regression adjustments

Consider the relationship between the conditional expectation of θ and s:

$$\mathbb{E}(\theta|s) = m(s)$$

Think of this as a model for the conditional density $\pi(\theta|s)$: for fixed s

$$\theta_i = m(s) + e_i$$

where $\theta_i \sim \pi(\theta|s)$ and e_i are zero-mean and uncorrelated

Idea 2 - regression adjustments

Consider the relationship between the conditional expectation of θ and s:

$$\mathbb{E}(\theta|s) = m(s)$$

Think of this as a model for the conditional density $\pi(\theta|s)$: for fixed s

$$\theta_i = m(s) + e_i$$

where $\theta_i \sim \pi(\theta|s)$ and e_i are zero-mean and uncorrelated

Suppose we've estimated $m(s)$ by $\widehat{m}(s)$ from samples $\{\theta_i, s_i\}$.
Estimate the posterior mean by

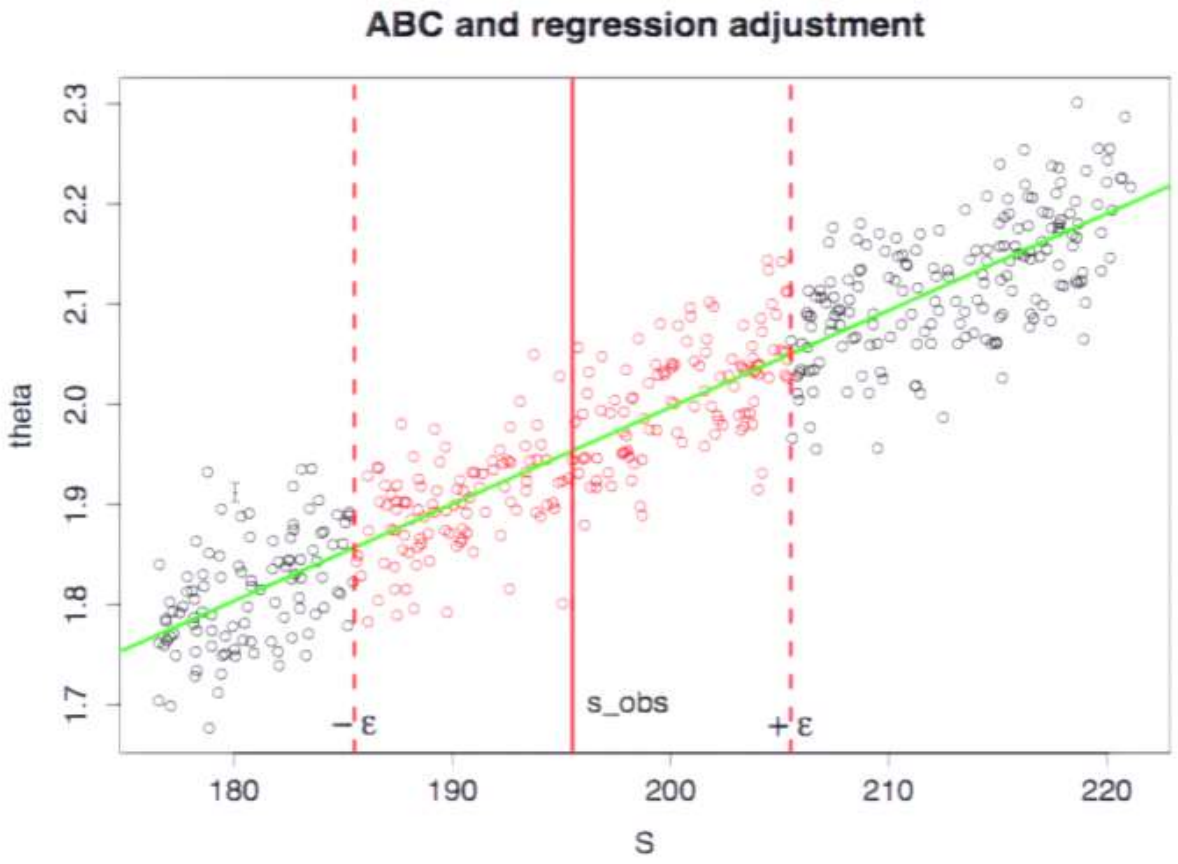
$$\mathbb{E}(\theta|s_{obs}) \approx \widehat{m}(s_{obs})$$

and assuming constant variance (wrt s), we can form the empirical residuals

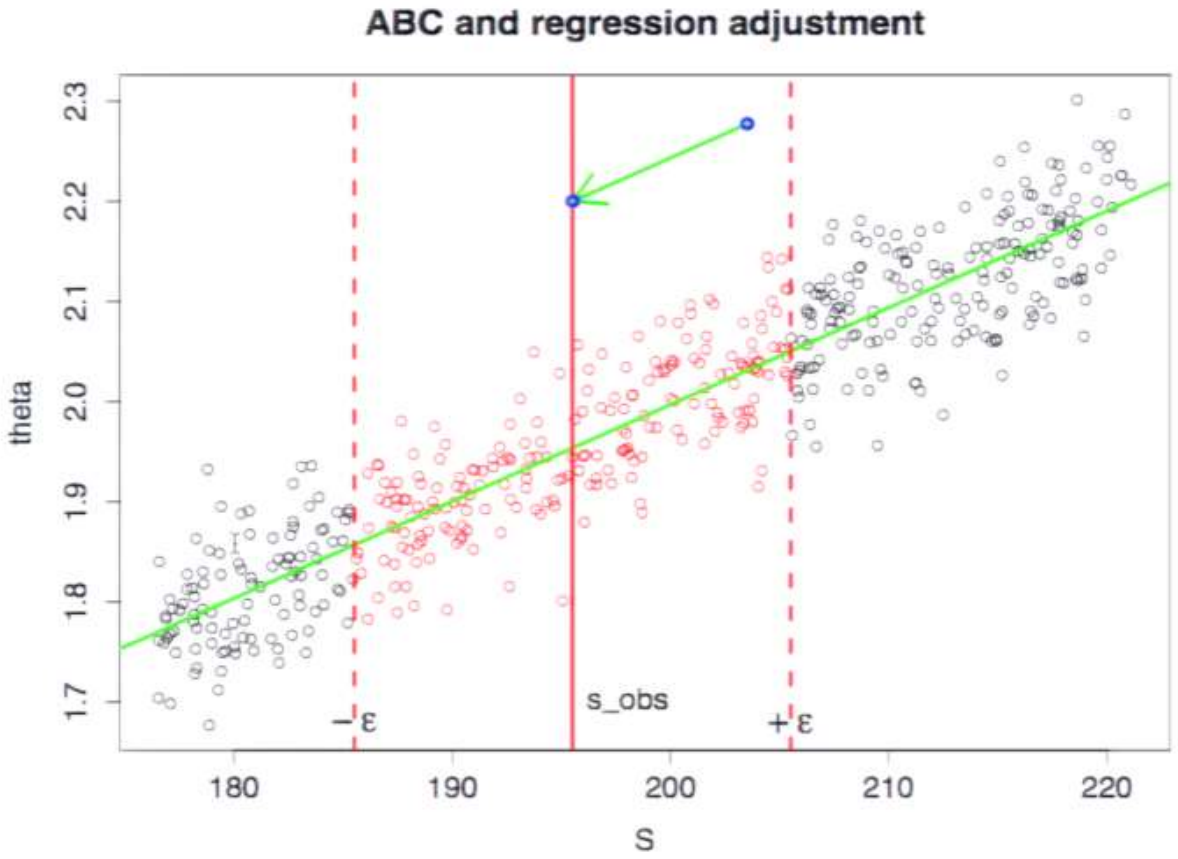
$$\hat{e}_i = \theta_i - \widehat{m}(s_i)$$

and approximate the posterior $\pi(\theta|s_{obs})$ by adjusting the parameters

$$\theta_i^* = \widehat{m}(s_{obs}) + \hat{e}_i = \theta_i + (\widehat{m}(s_{obs}) - \widehat{m}(s_i))$$



In rejection ABC, the red points are used to approximate the histogram.



In rejection ABC, the red points are used to approximate the histogram. Using regression-adjustment, we use the estimate of the posterior mean at s_{obs} and the residuals from the fitted line to form the posterior.

Models

Beaumont *et al.* 2003 used a local linear model for $m(s)$ in the vicinity of s_{obs}

$$m(s_i) = \alpha + \beta^T s_i$$

fit by minimising

$$\sum (\theta_i - m(s_i))^2 K_\epsilon (s_i - s_{obs})$$

so that observations nearest to s_{obs} are given more weight in the fit.

Models

Beaumont *et al.* 2003 used a local linear model for $m(s)$ in the vicinity of s_{obs}

$$m(s_i) = \alpha + \beta^T s_i$$

fit by minimising

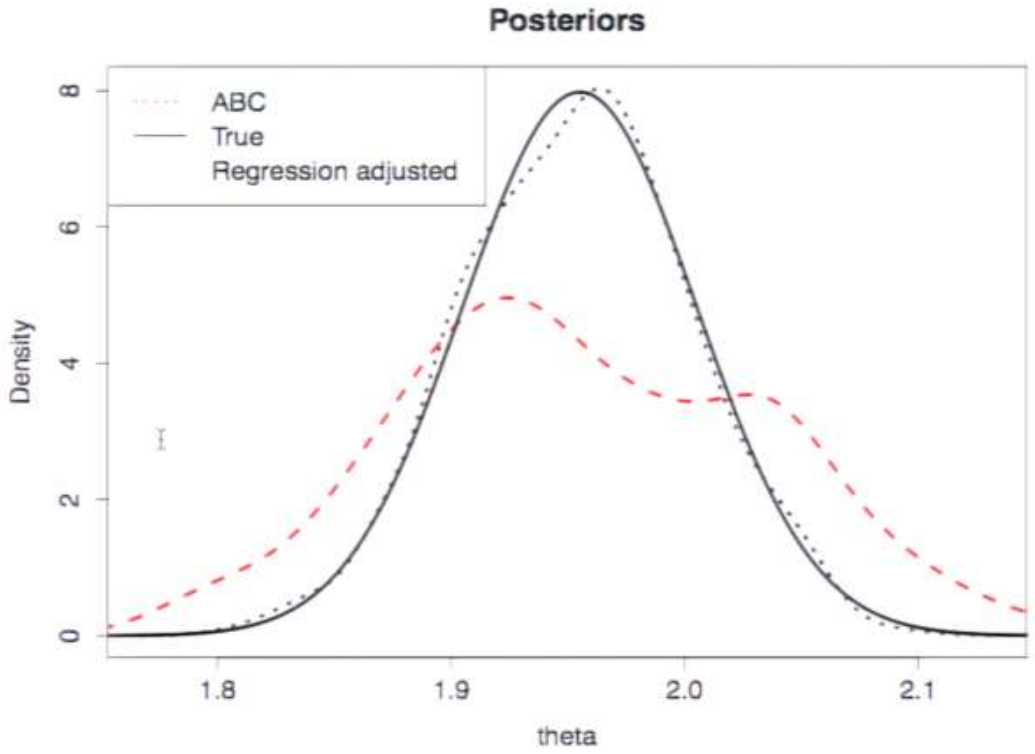
$$\sum (\theta_i - m(s_i))^2 K_\epsilon(s_i - s_{obs})$$

so that observations nearest to s_{obs} are given more weight in the fit.

The empirical residuals are then weighted so that the approximation to the posterior is a weighted particle set

$$\{\theta_i^*, W_i = K_\epsilon(s_i - s_{obs})\}$$
$$\pi(\theta|s_{obs}) = \widehat{m}(s_{obs}) + \sum w_i \delta_{\theta_i^*}(\theta)$$

Normal-normal conjugate model, linear regression



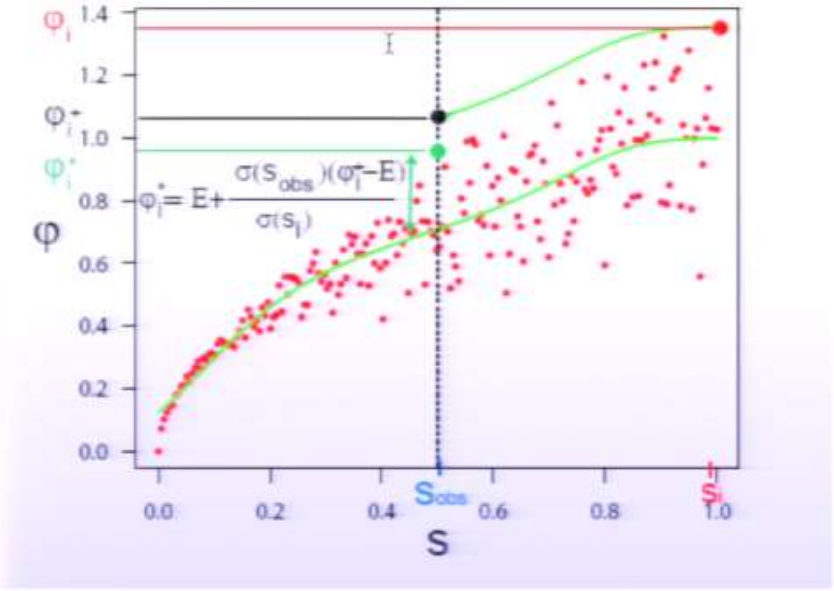
200 data points in both approximations. The regression-adjusted ABC gives a more confident posterior, as the θ_i have been adjusted to account for the discrepancy between s_i and s_{obs}

Extensions: Non-linear models

Blum and Francois 2010 proposed a nonlinear heteroscedastic model

$$\theta_i = m(s_i) + \sigma(s_u)e_i$$

where $m(s) = \mathbb{E}(\theta|s)$ and $\sigma^2(s) = \mathbb{V}\text{ar}(\theta|s)$. They used feed-forward neural networks for both the conditional mean and variance.



$$\theta_i^* = m(s_{obs}) + (\theta_i - \hat{m}(s_i))\,\frac{\hat{\sigma}(s_{obs})}{\hat{\sigma}(s_i)}$$

Picture from Michael Blum, www.ceremade.dauphine.fr/...pdf

Models

Beaumont *et al.* 2003 used a local linear model for $m(s)$ in the vicinity of s_{obs}

$$m(s_i) = \alpha + \beta^T s_i$$

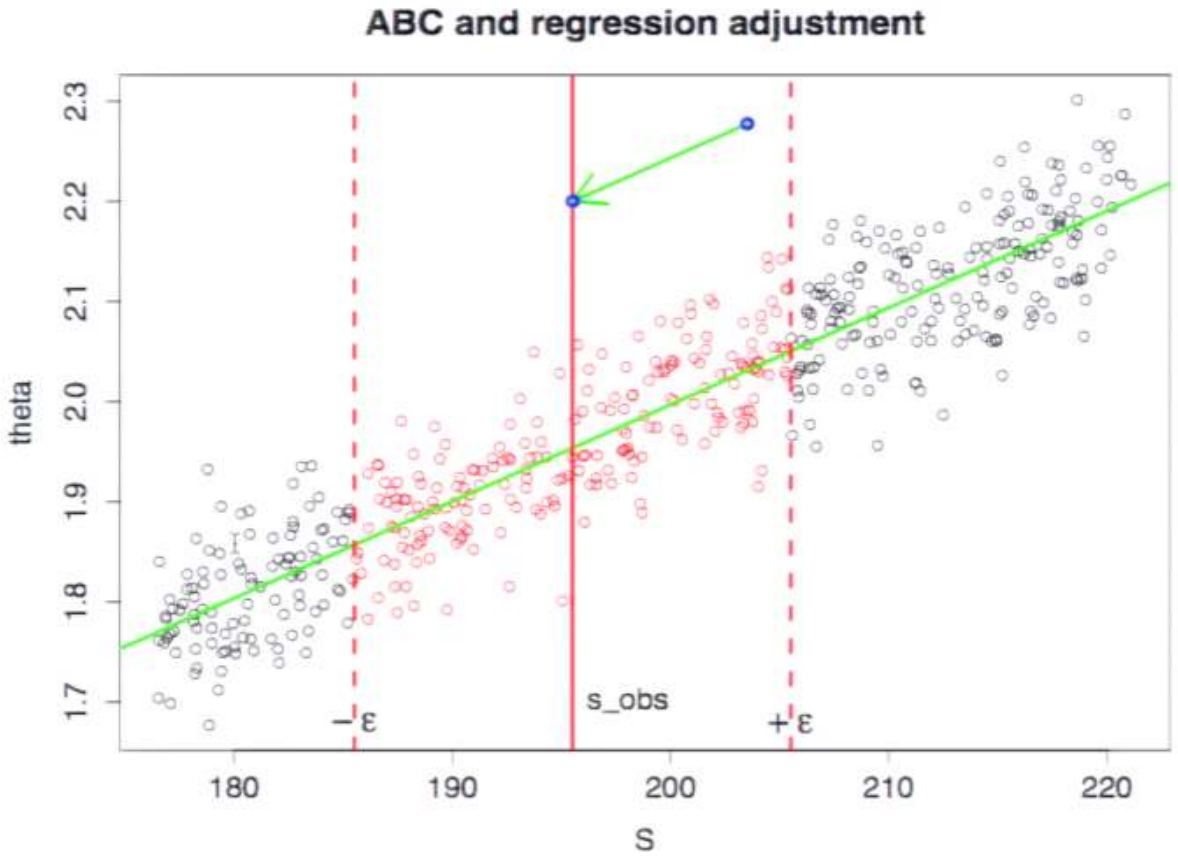
fit by minimising

$$\sum (\theta_i - m(s_i))^2 K_\epsilon(s_i - s_{obs})$$

so that observations nearest to s_{obs} are given more weight in the fit.

The empirical residuals are then weighted so that the approximation to the posterior is a weighted particle set

$$\{\theta_i^*, W_i = K_\epsilon(s_i - s_{obs})\}$$
$$\pi(\theta|s_{obs}) = \widehat{m}(s_{obs}) + \sum w_i \delta_{\theta_i^*}(\theta)$$



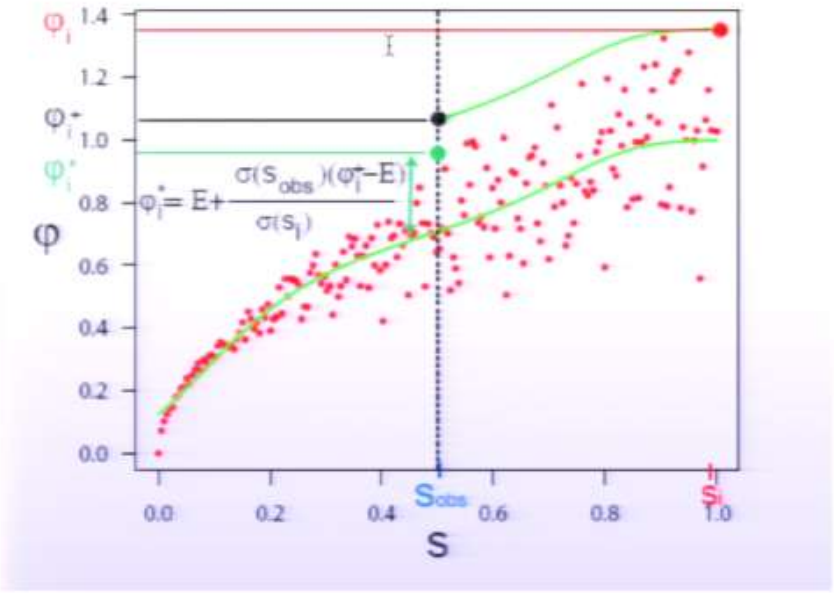
In rejection ABC, the red points are used to approximate the histogram. Using regression-adjustment, we use the estimate of the posterior mean at s_{obs} and the residuals from the fitted line to form the posterior.

Extensions: Non-linear models

Blum and Francois 2010 proposed a nonlinear heteroscedastic model

$$\theta_i = m(s_i) + \sigma(s_u)e_i$$

where $m(s) = \mathbb{E}(\theta|s)$ and $\sigma^2(s) = \mathbb{V}\text{ar}(\theta|s)$. They used feed-forward neural networks for both the conditional mean and variance.



$$\theta_i^* = m(s_{obs}) + (\theta_i - \hat{m}(s_i)) \frac{\hat{\sigma}(s_{obs})}{\hat{\sigma}(s_i)}$$

Picture from Michael Blum, www.ceremade.dauphine.fr/...pdf

Discussion

- These methods allow us to use a larger tolerance values and can substantially improve posterior accuracy with less computation. However, sequential algorithms can not easily be adapted, and so these methods tend to be used with simple rejection sampling.
- Many people choose not to use these methods, as they can give poor results if the model is badly chosen.
- Modelling variance is hard, so transformations to make the $\theta = m(s)$ as homoscedastic as possible (such as Box-Cox transformations) are usually applied
- Blum 2010 contains estimates of the bias and variance of these estimators. They show the properties of the ABC estimators may seriously deteriorate as $\dim(s)$ increases ...

Summary Statistics

References:

- Blum, Nunes, Prangle and Sisson 2012
- Joyce and Marjoram 2008
- Nunes and Balding 2010
- Fearnhead and Prangle 2012
- Robert *et al.* 2011

Error trade-off

Blum, Nunes, Prangle, Fearnhead 2012

The error in the ABC approximation can be broken into two parts

1. Choice of summary:

$$\pi(\theta|D) \stackrel{?}{\approx} \pi(\theta|S(D))$$

Error trade-off

Blum, Nunes, Prangle, Fearnhead 2012

The error in the ABC approximation can be broken into two parts

1. Choice of summary:

$$\pi(\theta|D) \stackrel{?}{\approx} \pi(\theta|S(D))$$

2. Use of ABC acceptance kernel:

$$\pi(\theta|s_{obs}) \stackrel{?}{\approx} \pi_{ABC}(\theta|s_{obs}) = \int \pi(\theta, s|s_{obs})\mathrm{d}s$$

$$\propto \int \pi_{\epsilon}(s_{obs}|S(x))\pi(x|\theta)\pi(\theta)\mathrm{d}x$$

Error trade-off

Blum, Nunes, Prangle, Fearnhead 2012

The error in the ABC approximation can be broken into two parts

1. Choice of summary:

$$\pi(\theta|D) \stackrel{?}{\approx} \pi(\theta|S(D))$$

2. Use of ABC acceptance kernel:

$$\pi(\theta|s_{obs}) \stackrel{?}{\approx} \pi_{ABC}(\theta|s_{obs}) = \int \pi(\theta, s|s_{obs})\mathrm{d}s$$
$$\propto \int \pi_{\epsilon}(s_{obs}|S(x))\pi(x|\theta)\pi(\theta)\mathrm{d}x$$

The first approximation allows the matching between $S(D)$ and $S(X)$ to be done in a lower dimension. There is a trade-off

- $\dim(S)$ small: $\pi(\theta|s_{obs}) \approx \pi_{ABC}(\theta|s_{obs})$, but $\pi(\theta|s_{obs}) \not\approx \pi(\theta|D)$
- $\dim(S)$ large: $\pi(\theta|s_{obs}) \approx \pi(\theta|D)$ but $\pi(\theta|s_{obs}) \not\approx \pi_{ABC}(\theta|s_{obs})$ as curse of dimensionality forces us to use larger ϵ

Choosing summary statistics

If $S(D) = s_{obs}$ is sufficient for θ, i.e., s_{obs} contains all the information contained in D about θ

$$\pi(\theta|s_{obs}) = \pi(\theta|D),$$

then using summaries has no detrimental effect

Choosing summary statistics

If $S(D) = s_{obs}$ is sufficient for θ, i.e., s_{obs} contains all the information contained in D about θ

$$\pi(\theta|s_{obs}) = \pi(\theta|D),$$

then using summaries has no detrimental effect

However, low-dimensional sufficient statistics are rarely available. How do we choose good low dimensional summaries?

Automated summary selection

Blum, Nunes, Prangle and Fearnhead 2012

Suppose we are given a candidate set $\mathcal{S} = (s_1, \ldots, s_p)$ of summaries from which to choose.

Methods break down into groups.

- Best subset selection
  - Joyce and Marjoram 2008
  - Nunes and Balding 2010
- Projection
  - Blum and Francois 2010
  - Fearnhead and Prangle 2012
- Regularisation techniques
  - Blum, Nunes, Prangle and Fearnhead 2012

Best subset selection

Introduce a criterion, e.g,

- τ-sufficiency (Joyce and Marjoram 2008): $s_{1:k-1}$ are τ-sufficient relative to s_k if

$$\begin{aligned}\delta_k &= \sup_\theta \log \pi(s_k|s_{1:k-1},\theta) - \inf_\theta \log \pi(s_k|s_{1:k-1},\theta) \\ &= \text{range}_\theta(\pi(s_{1:k}|\theta) - \pi(s_{1:k-1}|\theta)) \leq \tau\end{aligned}$$

  i.e. adding s_k changes posterior sufficiently.
- Entropy (Nunes and Balding 2010)

Implement within a search algorithm such as forward selection.
Problems:

- assumes every change to posterior is beneficial (see below)
- considerable computational effort required to compute δ_k

Projection

Several statistics from $\mathcal{S}$ may be required to get same info content as a single informative summary.

- project $\mathcal{S}$ onto a lower dimensional highly informative summary vector

Best subset selection

Introduce a criterion, e.g,

- τ-sufficiency (Joyce and Marjoram 2008): $s_{1:k-1}$ are τ-sufficient relative to s_k if

$$\begin{aligned}\delta_k &= \sup_\theta \log \pi(s_k|s_{1:k-1},\theta) - \inf_\theta \log \pi(s_k|s_{1:k-1},\theta) \\ &= \text{range}_\theta(\pi(s_{1:k}|\theta) - \pi(s_{1:k-1}|\theta)) \leq \tau\end{aligned}$$

  i.e. adding s_k changes posterior sufficiently.
- Entropy (Nunes and Balding 2010)

Implement within a search algorithm such as forward selection.
Problems:

- assumes every change to posterior is beneficial (see below)
- considerable computational effort required to compute δ_k

Projection

Several statistics from $\mathcal{S}$ may be required to get same info content as a single informative summary.

- project $\mathcal{S}$ onto a lower dimensional highly informative summary vector

Projection

Several statistics from $\mathcal{S}$ may be required to get same info content as a single informative summary.

- project $\mathcal{S}$ onto a lower dimensional highly informative summary vector

Most authors aim to find summaries so that

$$\pi_{ABC}(\theta|s) \approx \pi(\theta|D)$$

Fearnhead and Prangle 2012 weaken this requirement and instead aim to find summaries that lead to good parameter estimates.

Projection

Several statistics from $\mathcal{S}$ may be required to get same info content as a single informative summary.

- project $\mathcal{S}$ onto a lower dimensional highly informative summary vector

Most authors aim to find summaries so that

$$\pi_{ABC}(\theta|s) \approx \pi(\theta|D)$$

Fearnhead and Prangle 2012 weaken this requirement and instead aim to find summaries that lead to good parameter estimates.

They seek to minimise the expected posterior loss

$$\mathbb{E}((\theta_{true} - \hat{\theta})^2|D) \implies \hat{\theta} = \mathbb{E}(\theta|D)$$

They show that the optimal summary statistic is

$$s = \mathbb{E}(\theta|D)$$

Fearnhead and Prangle 2012

However, $\mathbb{E}(\theta|D)$ will not usually be known.

Instead, we can estimate it using the model

$$\theta_i = \mathbb{E}(\theta|D) + e_i = \beta^T f(s_i) + e_i$$

where $f(s)$ is a vector of functions of $\mathcal{S}$ and (θ_i, s_i) are output from a pilot ABC simulation. They choose the set of regressors using, e.g., BIC.

Fearnhead and Prangle 2012

However, $\mathbb{E}(\theta|D)$ will not usually be known.

Instead, we can estimate it using the model

$$\theta_i = \mathbb{E}(\theta|D) + e_i = \beta^T f(s_i) + e_i$$

where $f(s)$ is a vector of functions of $\mathcal{S}$ and (θ_i, s_i) are output from a pilot ABC simulation. They choose the set of regressors using, e.g., BIC.

They then use the single summary statistic

$$s = \hat{\beta}^T f(s)$$

for θ.

Advantages

- Scales well with large p and gives good point estimates.

Disadvantages

- Summaries usually lack interpretability and method gives no guarantees about the approximation of the posterior.

Summary warning:

Automated methods are a poor replacement for expert knowledge.

- Instead of automation, ask what aspects of the data do we expect our model to be able to reproduce? $S(D)$ may be highly informative about θ, but if the model was not built to reproduce $S(D)$ then why should we calibrate to it?
  - For example, many dynamical systems models are designed to model periods and amplitudes. Summaries that are not phase invariant may be informative about θ, but this information is uninformative.

  In the case where models and/or priors are mis-specified, this problem can be particularly acute.

Model selection

Wilkinson 2007, Grelaud *et al.* 2009

Ratmann *et al.* 2009 proposed methodology for testing the fit of a model without reference to other models.

But often we want to compare models $\rightarrow$ Bayes factors

$$B_{12} = \frac{\pi(D|M_1)}{\pi(D|M_2)}$$

where $\pi(D|M_i) = \int \pi_\epsilon(D|x)\pi(x|\theta, M_i)\pi(\theta)\mathrm{d}x\mathrm{d}\theta$

Model selection

Wilkinson 2007, Grelaud *et al.* 2009

Ratmann *et al.* 2009 proposed methodology for testing the fit of a model without reference to other models.

But often we want to compare models $\rightarrow$ Bayes factors

$$B_{12} = \frac{\pi(D|M_1)}{\pi(D|M_2)}$$

where $\pi(D|M_i) = \int \pi_\epsilon(D|x)\pi(x|\theta, M_i)\pi(\theta)\mathrm{d}x\mathrm{d}\theta$
For rejection ABC

$$\pi(D) \approx \frac{1}{N}\sum \pi_\epsilon(D|x_i)$$

which reduces to the acceptance rate for uniform ABC (Wilkinson 2007).

Or add an initial step into the rejection algorithm where we first pick a model - compare the ratio of acceptance rates to directly target the BF.

See Toni *et al.* 2009 for an SMC-ABC approach.

Summary statistics for model selection

Didelot *et al.* 2011, Robert *et al.* 2011

Care needs to be taken with regard summary statistics for model selection. Everything is okay if we target

$$B_S = \frac{\pi(S(D)|M_1)}{\pi(S(D)|M_2)}$$

Then the ABC estimator $\hat{B}_S^\epsilon \to B_S$ as $\epsilon \to 0, N \to \infty$ (Didelot *et al.* 2011).

Summary statistics for model selection

Didelot *et al.* 2011, Robert *et al.* 2011

Care needs to be taken with regard summary statistics for model selection.
Everything is okay if we target

$$B_S = \frac{\pi(S(D)|M_1)}{\pi(S(D)|M_2)}$$

Then the ABC estimator $\hat{B}_S^\epsilon \to B_S$ as $\epsilon \to 0, N \to \infty$ (Didelot *et al.* 2011).
However,

$$\frac{\pi(S(D)|M_1)}{\pi(S(D)|M_2)} \neq \frac{\pi(D|M_1)}{\pi(D|M_2)} = B_D$$

even if S is a sufficient statistic! S sufficient for $f_1(D|\theta_1)$ and $f_2(D|\theta_2)$ does not imply sufficiency for $\{m, f_m(D|\theta_m)\}$. Hence $\hat{B}_S^\epsilon \not\to B_D$

Summary statistics for model selection

Didelot *et al.* 2011, Robert *et al.* 2011

Care needs to be taken with regard summary statistics for model selection.
Everything is okay if we target

$$B_S = \frac{\pi(S(D)|M_1)}{\pi(S(D)|M_2)}$$

Then the ABC estimator $\hat{B}_S^\epsilon \to B_S$ as $\epsilon \to 0, N \to \infty$ (Didelot *et al.* 2011).
However,

$$\frac{\pi(S(D)|M_1)}{\pi(S(D)|M_2)} \neq \frac{\pi(D|M_1)}{\pi(D|M_2)} = B_D$$

even if S is a sufficient statistic! S sufficient for $f_1(D|\theta_1)$ and $f_2(D|\theta_2)$ does not imply sufficiency for $\{m, f_m(D|\theta_m)\}$. Hence $\hat{B}_S^\epsilon \not\to B_D$
Note - no problem if we view inference as conditional on a carefully chosen S.
See Prangle *et al.* 2013 for automatic selection of summaries for model selection.

Choice of metric ρ

Consider the following system

$$X_{t+1} = f(X_t) + N(0, \sigma^2) \tag{4}$$
$$Y_t = g(X_t) + N(0, \tau^2) \tag{5}$$

where we want to estimate measurement error τ and model error σ.
Default choice of metric (or similar)

$$\rho(Y, y^{obs}) = \sum (y_t^{obs} - Y_t)^2$$

or CRPS (a proper scoring rule)

$$\rho(y^{obs}, F(\cdot)) = \sum crps(y_t^{obs}, F_t(\cdot)) = \sum_t \int (F_t(u) - \mathbb{I}_{y_t \leq u})^2 \mathrm{du}$$

where $F_t(\cdot)$ is the distribution function of $Y_t | y_{1:t-1}$.

Summary statistics for model selection

Didelot *et al.* 2011, Robert *et al.* 2011

Care needs to be taken with regard summary statistics for model selection.
Everything is okay if we target

$$B_S = \frac{\pi(S(D)|M_1)}{\pi(S(D)|M_2)}$$

Then the ABC estimator $\hat{B}_S^\epsilon \to B_S$ as $\epsilon \to 0, N \to \infty$ (Didelot *et al.* 2011).
However,

$$\frac{\pi(S(D)|M_1)}{\pi(S(D)|M_2)} \neq \frac{\pi(D|M_1)}{\pi(D|M_2)} = B_D$$

even if S is a sufficient statistic! S sufficient for $f_1(D|\theta_1)$ and $f_2(D|\theta_2)$ does not imply sufficiency for $\{m, f_m(D|\theta_m)\}$. Hence $\hat{B}_S^\epsilon \not\to B_D$
Note - no problem if we view inference as conditional on a carefully chosen S.
See Prangle *et al.* 2013 for automatic selection of summaries for model selection.

Choice of metric ρ

Consider the following system

$$X_{t+1} = f(X_t) + N(0, \sigma^2) \tag{4}$$
$$Y_t = g(X_t) + N(0, \tau^2) \tag{5}$$

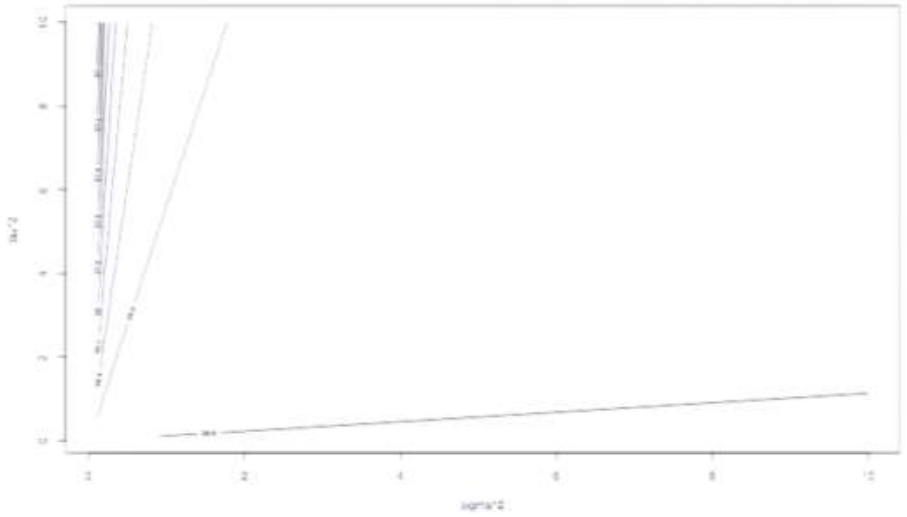
where we want to estimate measurement error τ and model error σ.
Default choice of metric (or similar)

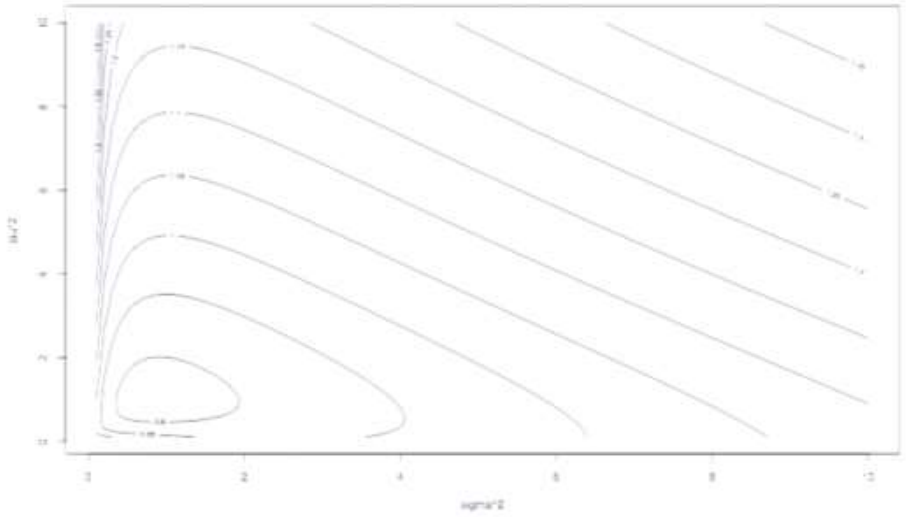
$$\rho(Y, y^{obs}) = \sum (y_t^{obs} - Y_t)^2$$

or CRPS (a proper scoring rule)

$$\rho(y^{obs}, F(\cdot)) = \sum crps(y_t^{obs}, F_t(\cdot)) = \sum_t \int (F_t(u) - \mathbb{I}_{y_t \leq u})^2 \mathrm{du}$$

where $F_t(\cdot)$ is the distribution function of $Y_t | y_{1:t-1}$.





Choice of metric ρ

Consider the following system

$$X_{t+1} = f(X_t) + N(0, \sigma^2) \tag{4}$$
$$Y_t = g(X_t) + N(0, \tau^2) \tag{5}$$

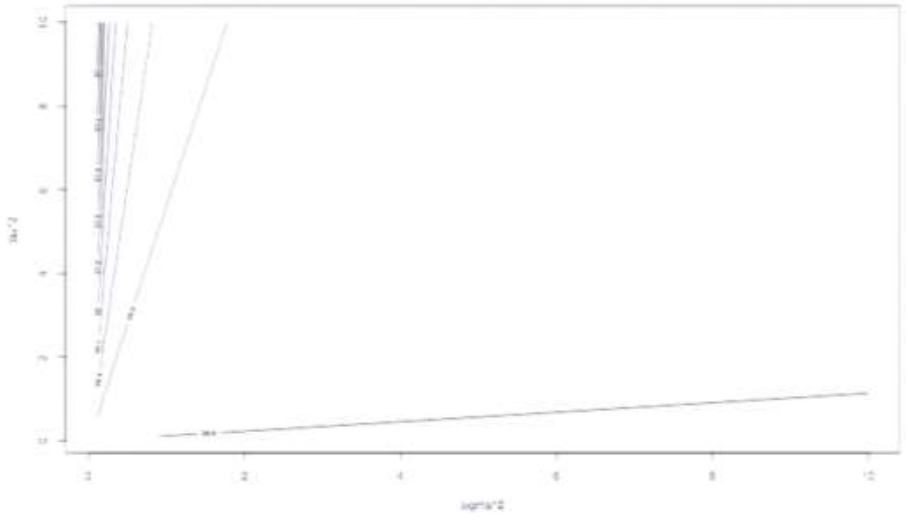
where we want to estimate measurement error τ and model error σ.
Default choice of metric (or similar)

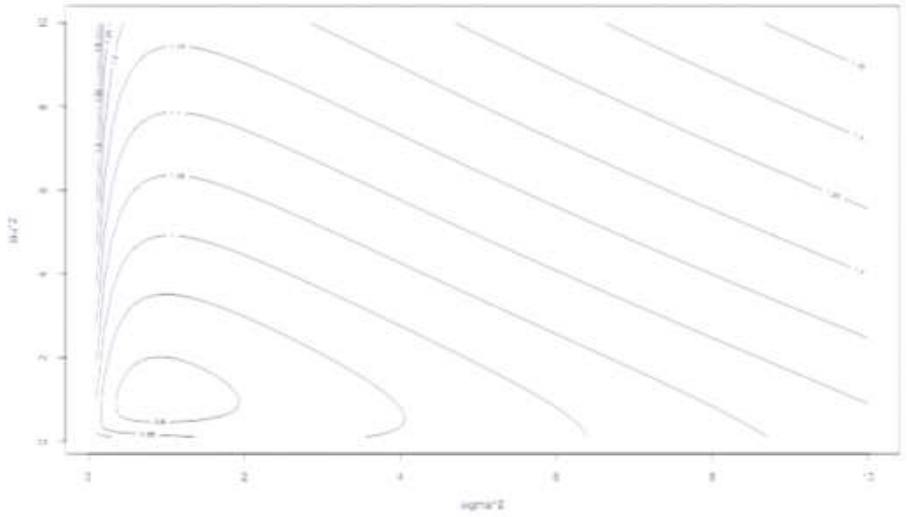
$$\rho(Y, y^{obs}) = \sum (y_t^{obs} - Y_t)^2$$

or CRPS (a proper scoring rule)

$$\rho(y^{obs}, F(\cdot)) = \sum crps(y_t^{obs}, F_t(\cdot)) = \sum_t \int (F_t(u) - \mathbb{I}_{y_t \leq u})^2 \mathrm{du}$$

where $F_t(\cdot)$ is the distribution function of $Y_t | y_{1:t-1}$.





Choice of metric ρ

Consider the following system

$$X_{t+1} = f(X_t) + N(0, \sigma^2) \tag{4}$$
$$Y_t = g(X_t) + N(0, \tau^2) \tag{5}$$

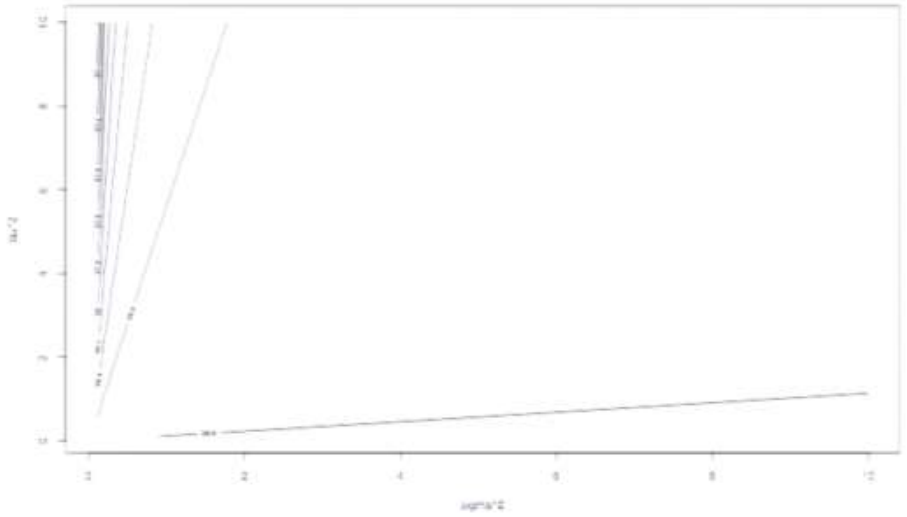
where we want to estimate measurement error τ and model error σ.
Default choice of metric (or similar)

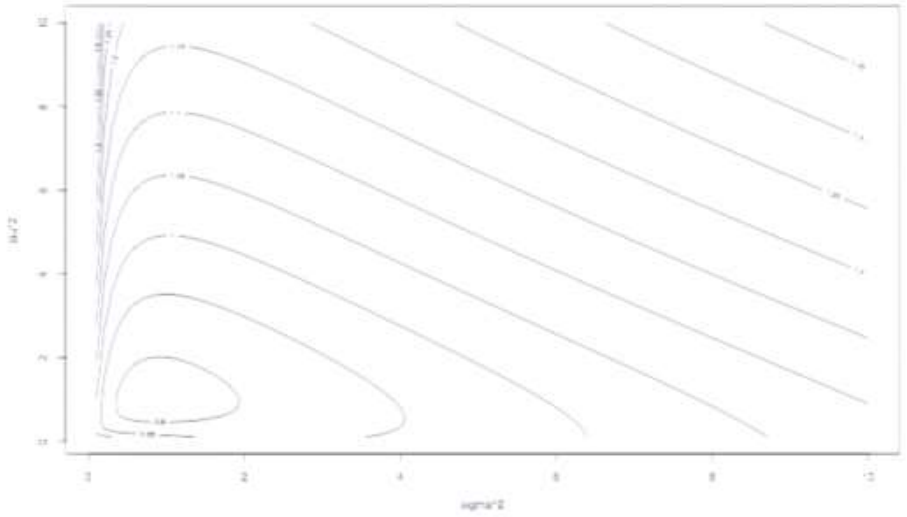
$$\rho(Y, y^{obs}) = \sum (y_t^{obs} - Y_t)^2$$

or CRPS (a proper scoring rule)

$$\rho(y^{obs}, F(\cdot)) = \sum crps(y_t^{obs}, F_t(\cdot)) = \sum_t \int (F_t(u) - \mathbb{I}_{y_t \leq u})^2 \mathrm{du}$$

where $F_t(\cdot)$ is the distribution function of $Y_t | y_{1:t-1}$.





Choice of metric ρ

Consider the following system

$$X_{t+1} = f(X_t) + N(0, \sigma^2) \tag{4}$$
$$Y_t = g(X_t) + N(0, \tau^2) \tag{5}$$

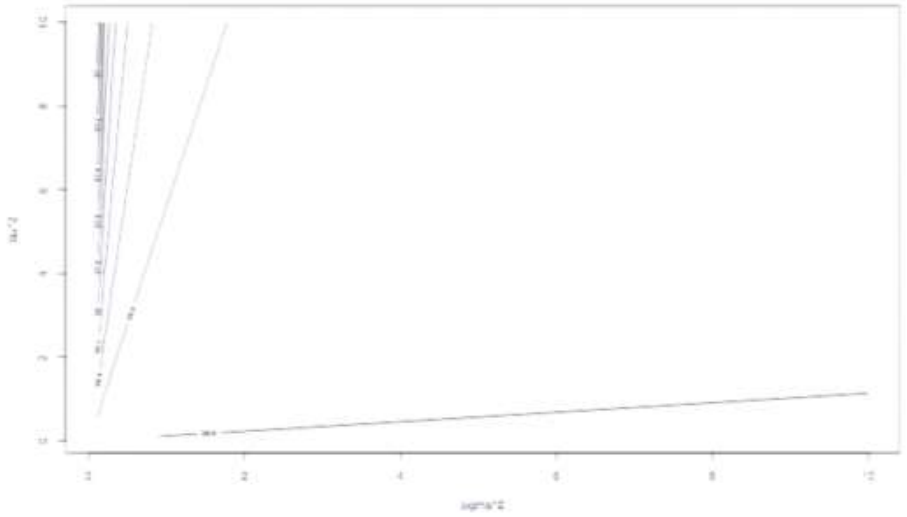
where we want to estimate measurement error τ and model error σ.
Default choice of metric (or similar)

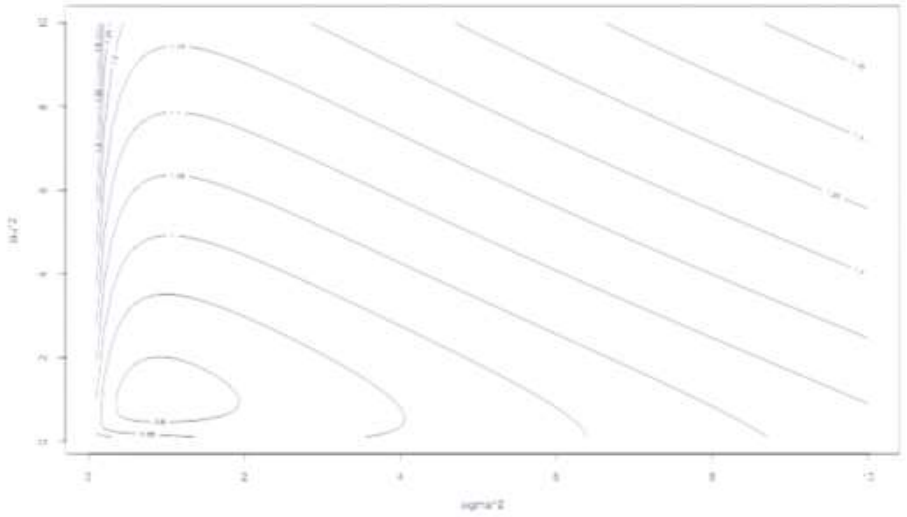
$$\rho(Y, y^{obs}) = \sum (y_t^{obs} - Y_t)^2$$

or CRPS (a proper scoring rule)

$$\rho(y^{obs}, F(\cdot)) = \sum crps(y_t^{obs}, F_t(\cdot)) = \sum_t \int (F_t(u) - \mathbb{I}_{y_t \leq u})^2 \mathrm{du}$$

where $F_t(\cdot)$ is the distribution function of $Y_t | y_{1:t-1}$.





GP-accelerated ABC

Problems with Monte Carlo methods

Monte Carlo methods are generally guaranteed to succeed if we run them for long enough.

This guarantee comes at a cost.

- Most methods sample naively - they don't learn from previous simulations.
- They don't exploit known properties of the likelihood function, such as continuity
- They sample randomly, rather than using space filling designs.

This naivety can make a full analysis infeasible without access to a large amount of computational resource.

Likelihood estimation

The GABC framework assumes

$$\pi(D|\theta) = \int \pi(D|X)\pi(X|\theta)\mathrm{dX}$$
$$\approx \frac{1}{N}\sum \pi(D|X_i)$$

where $X_i \sim \pi(X|\theta)$.

Likelihood estimation

The GABC framework assumes

$$\pi(D|\theta) = \int \pi(D|X)\pi(X|\theta)\mathrm{dX}$$
$$\approx \frac{1}{N}\sum \pi(D|X_i)$$

where $X_i \sim \pi(X|\theta)$.

For many problems, we believe the likelihood is continuous and smooth, so that $\pi(D|\theta)$ is similar to $\pi(D|\theta')$ when $\theta - \theta'$ is small

We can model $L(\theta) = \pi(D|\theta)$ and use the model to find the posterior in place of running the simulator.

History matching waves

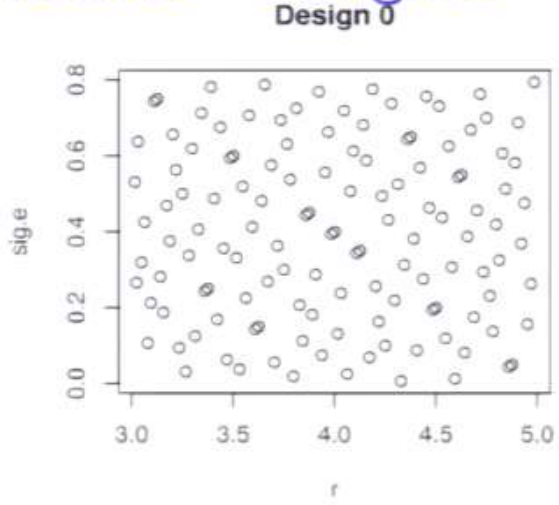
The likelihood is too difficult to model, so we model the log-likelihood instead.

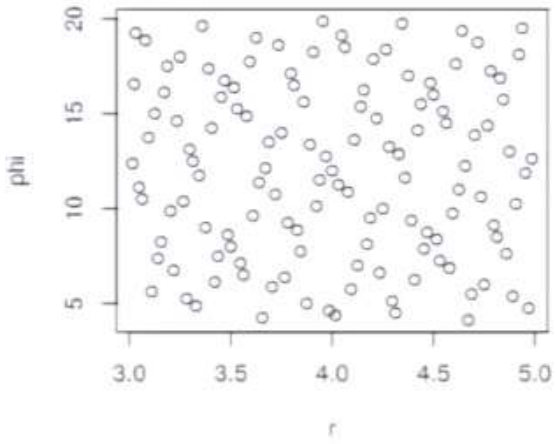
$$G(\theta) = \log L(\theta), \qquad \hat{L}(\theta_i) = \frac{1}{N}\sum \pi(D|X_i),\ X_i \sim \pi(X|\theta_i)$$

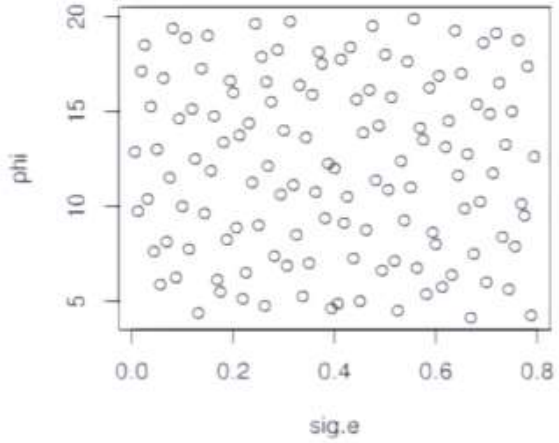
However, the log-likelihood for a typical problem ranges across too wide a range of values.

Consequently, any Gaussian process model will struggle to model the log-likelihood across the entire input range.

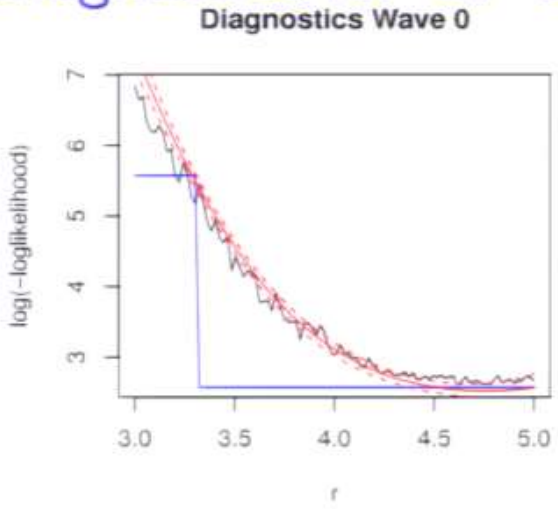
Results - Design 1 - 128 pts

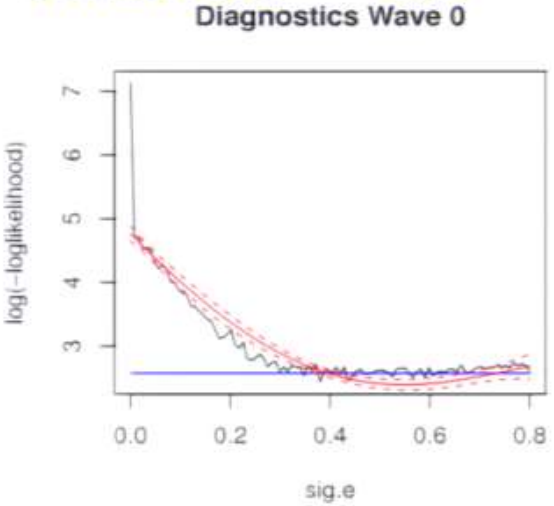


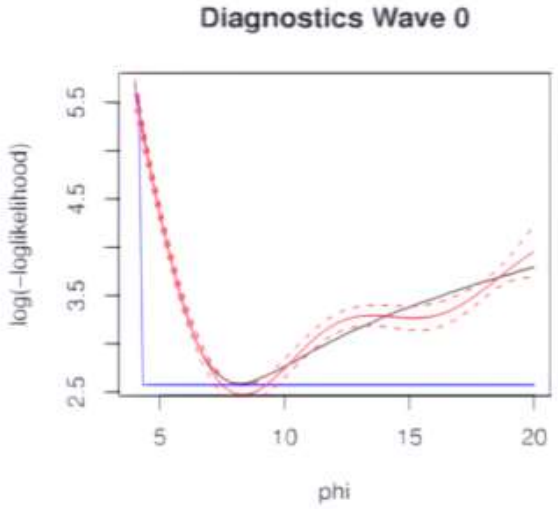




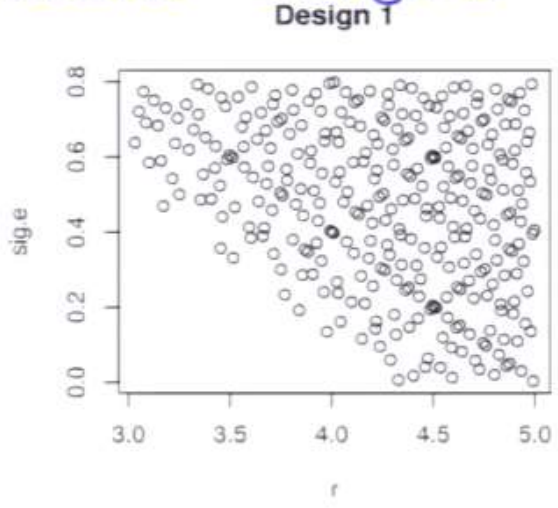
Diagnostics for GP 1 - threshold = 5.6

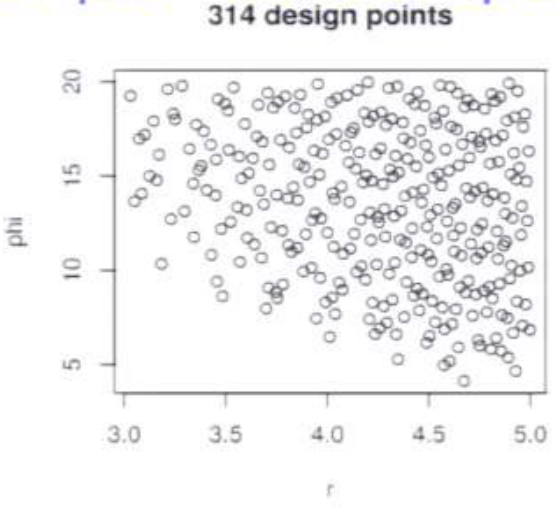


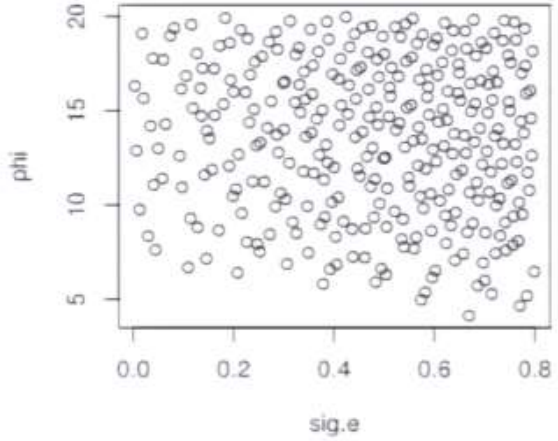




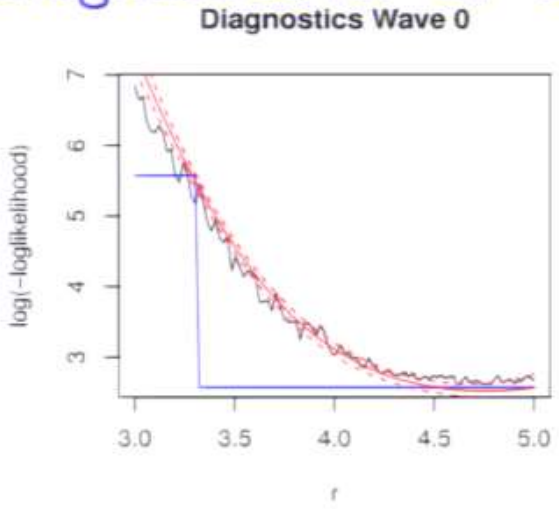
Results - Design 2 - 314 pts - 38% of space implausible

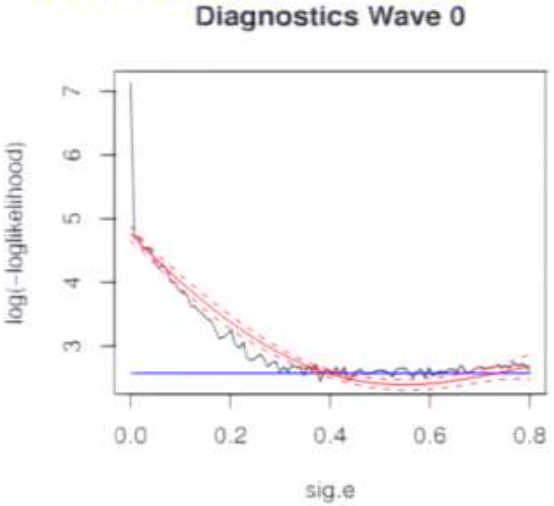


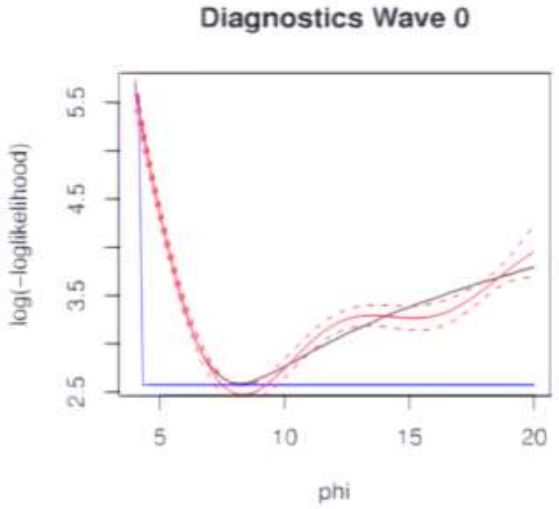




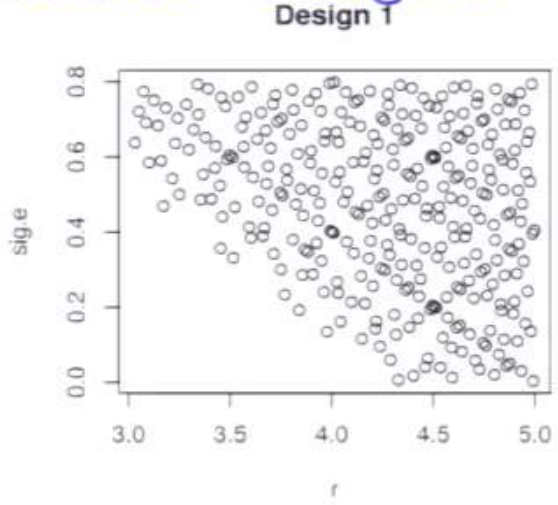
Diagnostics for GP 1 - threshold = 5.6

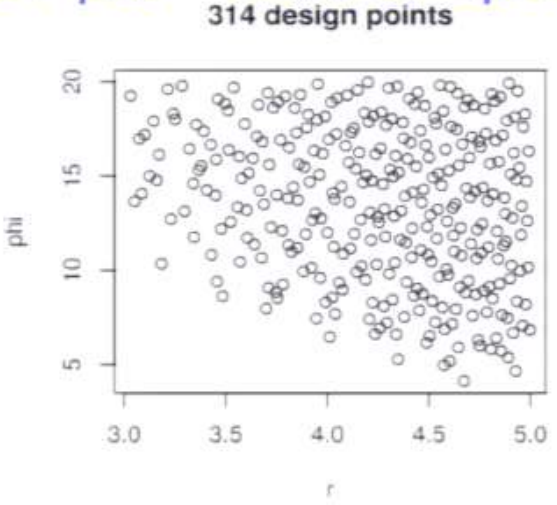


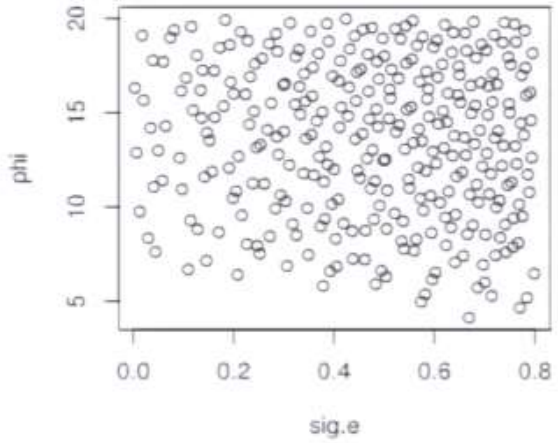




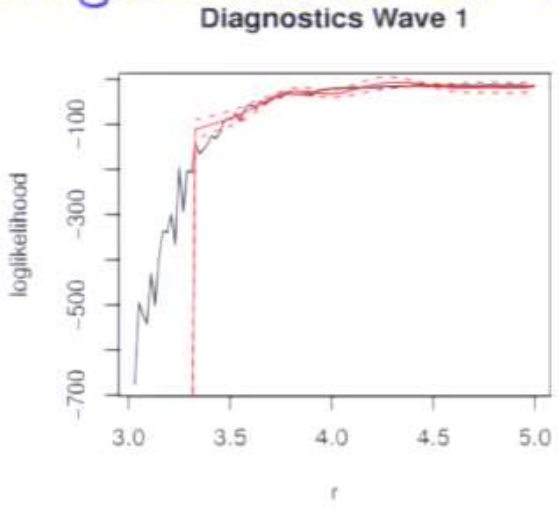
Results - Design 2 - 314 pts - 38% of space implausible

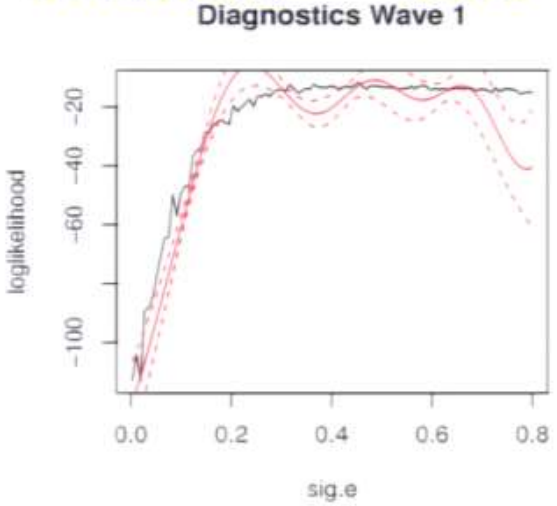


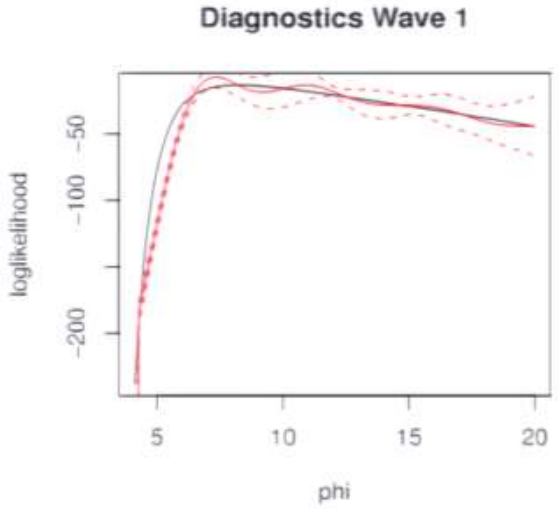




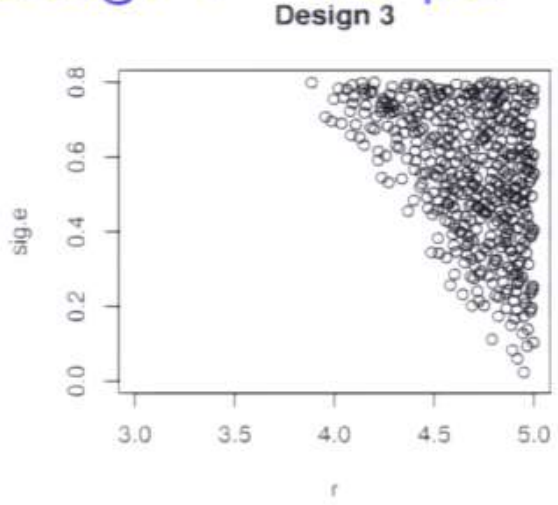
Diagnostics for GP 2 - threshold = -21.8

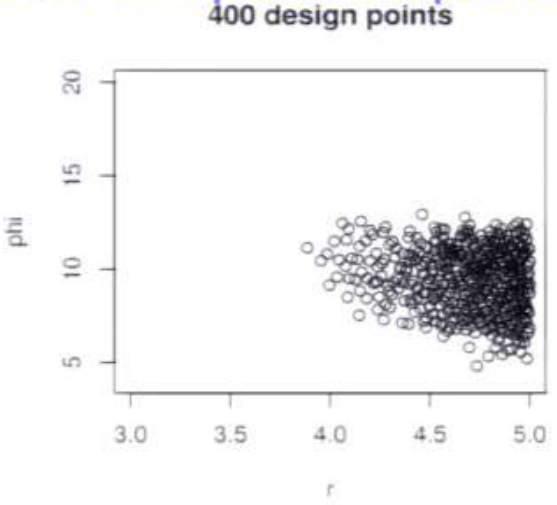


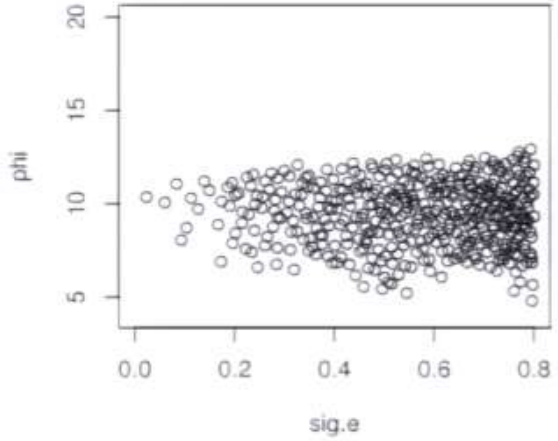




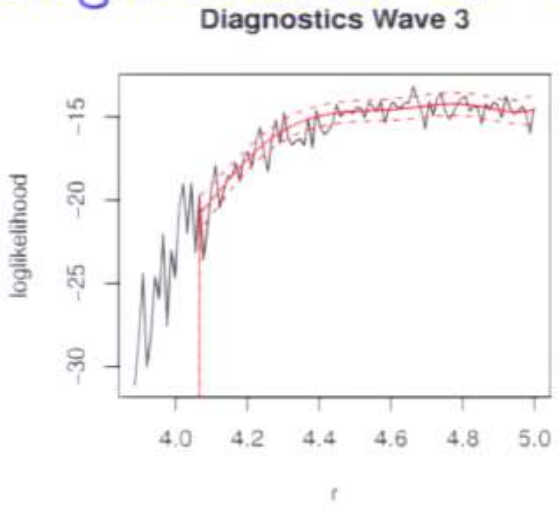
Design 4 - 400 pts - 95% of space implausible

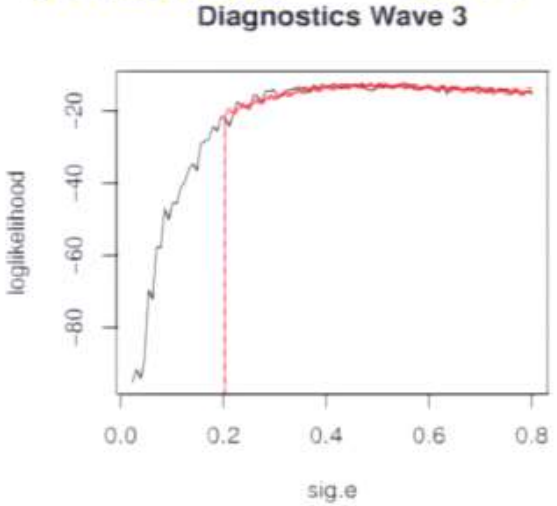


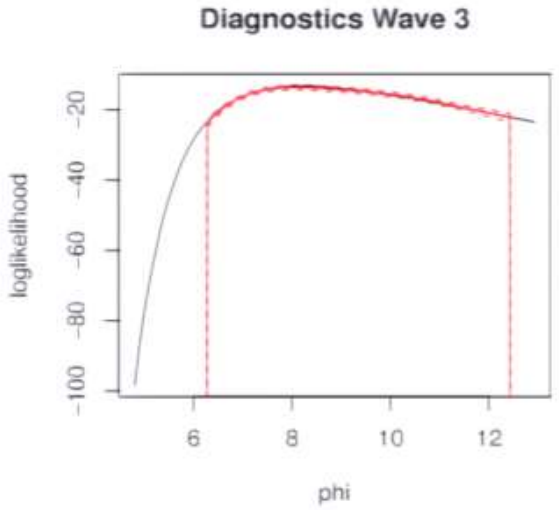




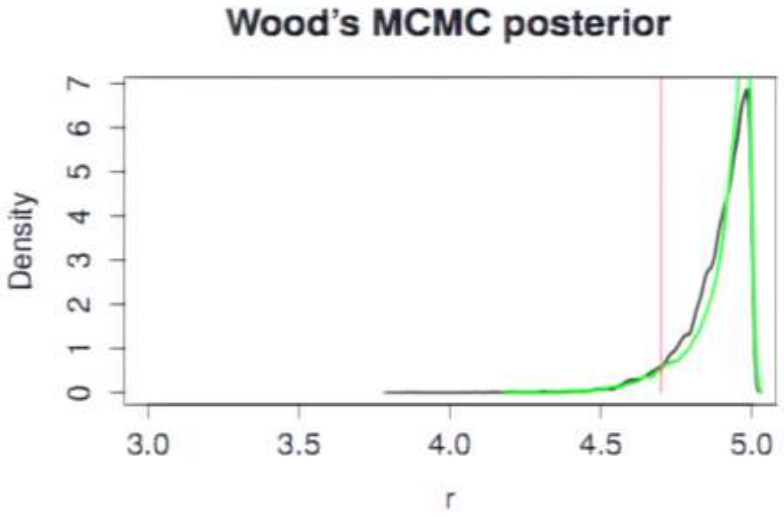
Diagnostics for GP 4 - threshold = -16.4

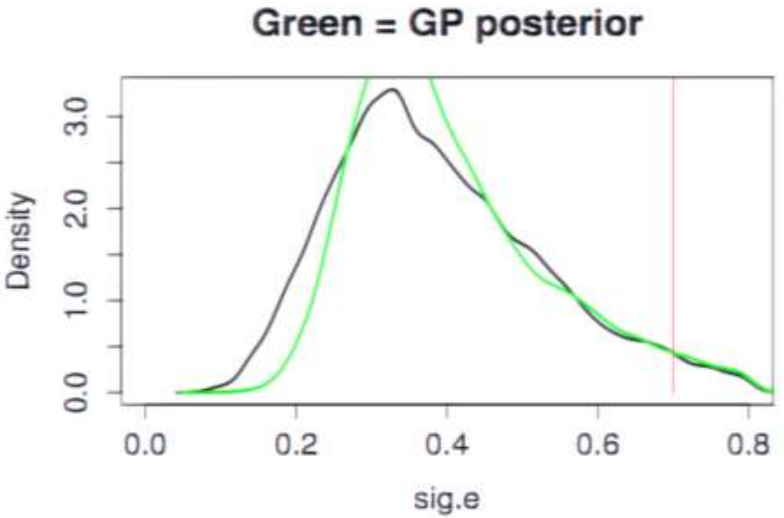


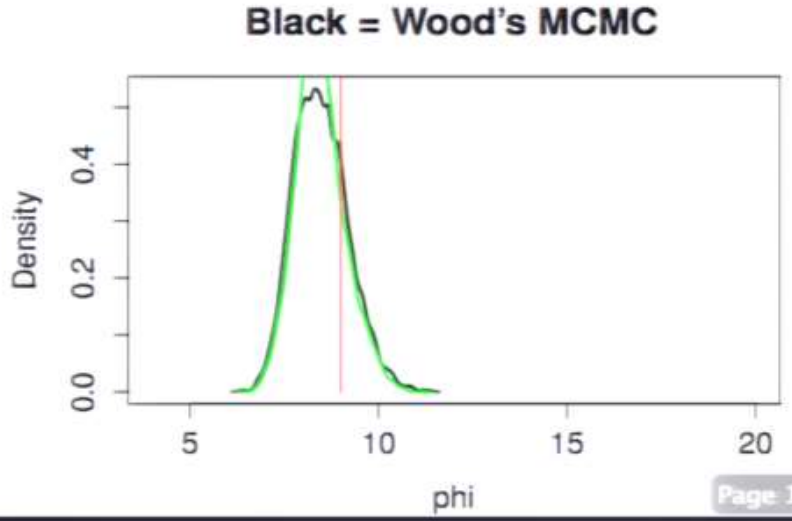




MCMC Results







Computational details

- The Wood MCMC method used $10^5 \times 500$ simulator runs
- The GP code used $(128 + 314 + 149 + 400) = 991 \times 500$ simulator runs
  - 1/100th of the number used by Wood's method.

By the final iteration, the Gaussian processes had ruled out over 98% of the original input space as implausible,

- the MCMC sampler did not need to waste time exploring those regions.

Conclusions

ABC allows inference in models for which it would otherwise be impossible.

- not a silver bullet - if likelihood methods possible, use them instead.

Algorithms and post-hoc regression can greatly improve computational efficiency, but computation is still usually the limiting factor.

- Challenge is to develop more efficient methods to allow inference in more expensive models.

Conclusions

ABC allows inference in models for which it would otherwise be impossible.

- not a silver bullet - if likelihood methods possible, use them instead.

Algorithms and post-hoc regression can greatly improve computational efficiency, but computation is still usually the limiting factor.

- Challenge is to develop more efficient methods to allow inference in more expensive models.

Areas for improvement (particularly those relevant to ML)?

- Automatic summary selection and dimension reduction
- Improved modelling in regression adjustments
- Learning of model error $\pi_\epsilon(D|X)$
- Accelerated inference via likelihood modelling
- Use of variational methods
- ...

Conclusions

ABC allows inference in models for which it would otherwise be impossible.

- not a silver bullet - if likelihood methods possible, use them instead.

Algorithms and post-hoc regression can greatly improve computational efficiency, but computation is still usually the limiting factor.

- Challenge is to develop more efficient methods to allow inference in more expensive models.

Areas for improvement (particularly those relevant to ML)?

- Automatic summary selection and dimension reduction
- Improved modelling in regression adjustments
- Learning of model error $\pi_\epsilon(D|X)$
- Accelerated inference via likelihood modelling
- Use of variational methods
- ...

Thank you for listening!

r.d.wilkinson@nottingham.ac.uk, www.maths.nottingham.ac.uk/personal/pmzrdw/

Conclusions

ABC allows inference in models for which it would otherwise be impossible.

- not a silver bullet - if likelihood methods possible, use them instead.

Algorithms and post-hoc regression can greatly improve computational efficiency, but computation is still usually the limiting factor.

- Challenge is to develop more efficient methods to allow inference in more expensive models.

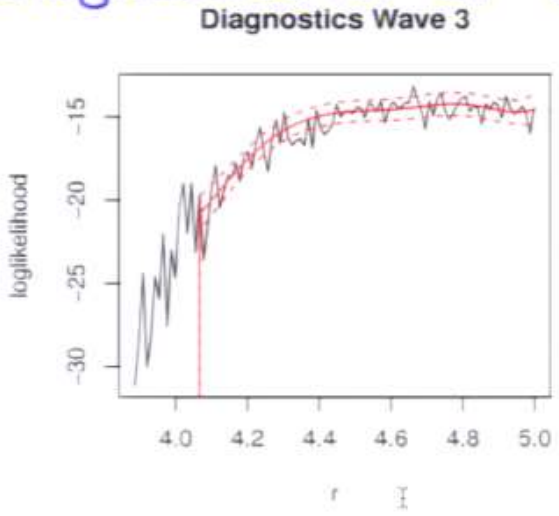
Computational details

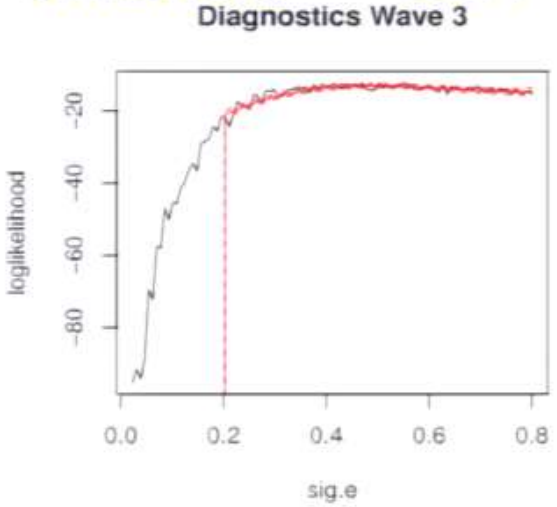
- The Wood MCMC method used $10^5 \times 500$ simulator runs
- The GP code used $(128 + 314 + 149 + 400) = 991 \times 500$ simulator runs
  - 1/100th of the number used by Wood's method.

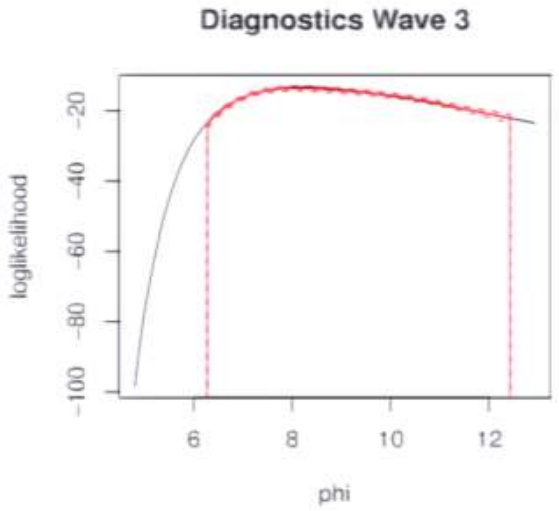
By the final iteration, the Gaussian processes had ruled out over 98% of the original input space as implausible,

- the MCMC sampler did not need to waste time exploring those regions.

Diagnostics for GP 4 - threshold = -16.4







Computational details

- The Wood MCMC method used $10^5 \times 500$ simulator runs
- The GP code used $(128 + 314 + 149 + 400) = 991 \times 500$ simulator runs
  - 1/100th of the number used by Wood's method.

By the final iteration, the Gaussian processes had ruled out over 98% of the original input space as implausible,

- the MCMC sampler did not need to waste time exploring those regions.

Conclusions

ABC allows inference in models for which it would otherwise be impossible.

- not a silver bullet - if likelihood methods possible, use them instead.

Algorithms and post-hoc regression can greatly improve computational efficiency, but computation is still usually the limiting factor.

- Challenge is to develop more efficient methods to allow inference in more expensive models.

Computational details

- The Wood MCMC method used $10^5 \times 500$ simulator runs
- The GP code used $(128 + 314 + 149 + 400) = 991 \times 500$ simulator runs
  - 1/100th of the number used by Wood's method.

By the final iteration, the Gaussian processes had ruled out over 98% of the original input space as implausible,

- the MCMC sampler did not need to waste time exploring those regions.

Conclusions

ABC allows inference in models for which it would otherwise be impossible.

- not a silver bullet - if likelihood methods possible, use them instead.

Algorithms and post-hoc regression can greatly improve computational efficiency, but computation is still usually the limiting factor.

- Challenge is to develop more efficient methods to allow inference in more expensive models.

Areas for improvement (particularly those relevant to ML)?

- Automatic summary selection and dimension reduction
- Improved modelling in regression adjustments
- Learning of model error $\pi_\epsilon(D|X)$
- Accelerated inference via likelihood modelling
- Use of variational methods
- ...

History matching waves

The likelihood is too difficult to model, so we model the log-likelihood instead.

$$G(\theta) = \log L(\theta), \qquad \hat{L}(\theta_i) = \frac{1}{N} \sum \pi(D|X_i),\ X_i \sim \pi(X|\theta_i)$$

History matching waves

The likelihood is too difficult to model, so we model the log-likelihood instead.

$$G(\theta) = \log L(\theta), \qquad \hat{L}(\theta_i) = \frac{1}{N} \sum \pi(D|X_i),\ X_i \sim \pi(X|\theta_i)$$

However, the log-likelihood for a typical problem ranges across too wide a range of values.

Consequently, any Gaussian process model will struggle to model the log-likelihood across the entire input range.

- Introduce waves of history matching, similar to those used in Michael Goldstein's work.
- In each wave, build a GP model that can rule out regions of space as *implausible*.

Likelihood estimation

The GABC framework assumes

$$\pi(D|\theta) = \int \pi(D|X)\pi(X|\theta)\mathrm{dX}$$
$$\approx \frac{1}{N}\sum \pi(D|X_i)$$

where $X_i \sim \pi(X|\theta)$.

For many problems, we believe the likelihood is continuous and smooth, so that $\pi(D|\theta)$ is similar to $\pi(D|\theta')$ when $\theta - \theta'$ is small

We can model $L(\theta) = \pi(D|\theta)$ and use the model to find the posterior in place of running the simulator.

History matching waves

The likelihood is too difficult to model, so we model the log-likelihood instead.

$$G(\theta) = \log L(\theta), \qquad \hat{L}(\theta_i) = \frac{1}{N}\sum \pi(D|X_i),\ X_i \sim \pi(X|\theta_i)$$

However, the log-likelihood for a typical problem ranges across too wide a range of values.

Consequently, any Gaussian process model will struggle to model the log-likelihood across the entire input range.

History matching waves

The likelihood is too difficult to model, so we model the log-likelihood instead.

$$G(\theta) = \log L(\theta), \qquad \hat{L}(\theta_i) = \frac{1}{N} \sum \pi(D|X_i),\ X_i \sim \pi(X|\theta_i)$$

However, the log-likelihood for a typical problem ranges across too wide a range of values.

Consequently, any Gaussian process model will struggle to model the log-likelihood across the entire input range.

- Introduce waves of history matching, similar to those used in Michael Goldstein's work.
- In each wave, build a GP model that can rule out regions of space as *implausible*.

Problems with Monte Carlo methods

Monte Carlo methods are generally guaranteed to succeed if we run them for long enough.

This guarantee comes at a cost.

- Most methods sample naively - they don't learn from previous simulations.
- They don't exploit known properties of the likelihood function, such as continuity
- They sample randomly, rather than using space filling designs.

This naivety can make a full analysis infeasible without access to a large amount of computational resource.

Likelihood estimation

The GABC framework assumes

$$\pi(D|\theta) = \int \pi(D|X)\pi(X|\theta)\mathrm{dX}$$
$$\approx \frac{1}{N}\sum \pi(D|X_i)$$

where $X_i \sim \pi(X|\theta)$.