

# Introduction to Approximate Bayesian Computation

John Haman

Bowling Green State University

*jthaman@bgsu.edu*

December 8, 2015

# Implicit models

Two types of statistical models:

- Prescribed models - Likelihood function is available, common for us
- Implicit models - mechanisms for simulating data, common for biologists, psychologist

# Implicit models

Two types of statistical models:

- Prescribed models - Likelihood function is available, common for us
- Implicit models - mechanisms for simulating data, common for biologists, psychologist

Implicit models give scientists more freedom to model phenomena, complex processes are favored over simple ones

# Monte Carlo inference

Aim to sample from a posterior distribution:

$$\pi(\theta|\mathcal{D}) \propto \mathbb{P}(\mathcal{D}|\theta) * \pi(\theta)$$

# Monte Carlo inference

Aim to sample from a posterior distribution:

$$\pi(\theta|\mathcal{D}) \propto \mathbb{P}(\mathcal{D}|\theta) * \pi(\theta)$$

MCMC is great for performing Bayesian inference in complex models, and it's exact up to a monte carlo error. It's not great when....

# Monte Carlo inference

Aim to sample from a posterior distribution:

$$\pi(\theta|\mathcal{D}) \propto \mathbb{P}(\mathcal{D}|\theta) * \pi(\theta)$$

MCMC is great for performing Bayesian inference in complex models, and it's exact up to a monte carlo error. It's not great when....

- $\mathbb{P}(\mathcal{D}|\theta)$  is unknown or intractable.

# Monte Carlo inference

Aim to sample from a posterior distribution:

$$\pi(\theta|\mathcal{D}) \propto \mathbb{P}(\mathcal{D}|\theta) * \pi(\theta)$$

MCMC is great for performing Bayesian inference in complex models, and it's exact up to a monte carlo error. It's not great when....

- $\mathbb{P}(\mathcal{D}|\theta)$  is unknown or intractable.
- independence cannot be assumed for many model parameters

So there is a need for generating posterior data without a likelihood function

## Rejection Algorithm (old way)

- Draw  $\theta$  from  $\pi(\ast)$
- Accept  $\theta$  with probability  $\mathbb{P}(\mathcal{D}|\theta)$



## Rejection Algorithm (old way)

- Draw  $\theta$  from  $\pi(\ast)$
- Accept  $\theta$  with probability  $\mathbb{P}(\mathcal{D}|\theta)$

# Likelihood free inference

## Rejection Algorithm (old way)

- Draw  $\theta$  from  $\pi(*)$
- Accept  $\theta$  with probability  $\mathbb{P}(\mathcal{D}|\theta)$

## Mechanical Rejection Algorithm (Without Likelihood)

- Draw  $\theta$  from  $\pi(*)$
- Simulate  $\mathcal{D}' \sim \mathbb{P}(*|\theta)$
- Accept  $\theta$  if  $\mathcal{D} = \mathcal{D}'$

The last requirement is probably too restrictive, e.g. if the data are continuous

## ABC Rejection

- Draw  $\theta$  from  $\pi(\theta)$
- Simulate  $\mathcal{D}' \sim \mathbb{P}(*|\theta)$
- Accept  $\theta$  if  $d(\mathcal{D}, \mathcal{D}') < \epsilon$ , or if  $d(S(\mathcal{D}), S(\mathcal{D}')) < \epsilon$ ,  $S$  is a sufficient statistic.

## ABC Rejection

- Draw  $\theta$  from  $\pi(\theta)$
- Simulate  $\mathcal{D}' \sim \mathbb{P}(*|\theta)$
- Accept  $\theta$  if  $d(\mathcal{D}, \mathcal{D}') < \epsilon$ , or if  $d(S(\mathcal{D}), S(\mathcal{D}')) < \epsilon$ ,  $S$  is a sufficient statistic.

This generates data from is distribution  $\pi(\theta|d(\mathcal{D}, \mathcal{D}') < \epsilon)$

- If  $\epsilon \rightarrow 0$ , we obtain data from the target density,  $\pi(\theta|\mathcal{D})$ .
- If  $\epsilon \rightarrow \infty$  we obtain data from the prior,  $\pi(\theta)$ .

## ABC Rejection

- Draw  $\theta$  from  $\pi(\theta)$
- Simulate  $\mathcal{D}' \sim \mathbb{P}(*|\theta)$
- Accept  $\theta$  if  $d(\mathcal{D}, \mathcal{D}') < \epsilon$ , or if  $d(S(\mathcal{D}), S(\mathcal{D}')) < \epsilon$ ,  $S$  is a sufficient statistic.

This generates data from is distribution  $\pi(\theta|d(\mathcal{D}, \mathcal{D}') < \epsilon)$

- If  $\epsilon \rightarrow 0$ , we obtain data from the target density,  $\pi(\theta|\mathcal{D})$ .
- If  $\epsilon \rightarrow \infty$  we obtain data from the prior,  $\pi(\theta)$ .

Many types of ABC algorithms now... M-H, regression based, neural networks, all implemented by the R package abc.

# Metropolis Hastings

Rejection sampling is very inefficient, because  $\theta$  is being sampled from the prior at each step

# Metropolis Hastings

Rejection sampling is very inefficient, because  $\theta$  is being sampled from the prior at each step

We can speed things up by correlating observations to spend more time in regions of high likelihood...

## Metropolis-Hastings with ABC

- Suppose we are at  $\theta$ . Propose  $\theta'$  from density  $q(\theta, \theta')$
- Simulate  $\mathcal{D}'$  from  $\mathbb{P}(*|\theta')$
- If  $d(\mathcal{D}, \mathcal{D}') \leq \epsilon$  calculate

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

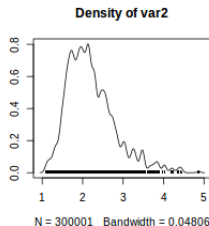
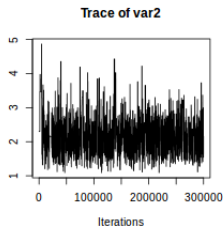
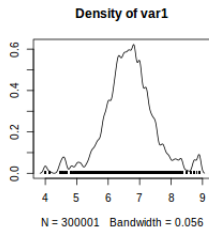
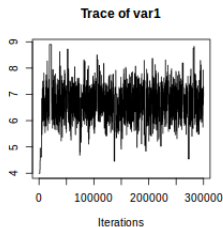
- Accept the move to  $\theta'$  with probability  $h(\theta, \theta')$ , else stay at  $\theta$ .

# Example

## Normal data

- Generate  $\mathcal{D} \sim N(5.3, 2.7^2)$  Save  $\bar{\mathcal{D}}$  and  $sd(\mathcal{D})$ .
- Accept  $\mathcal{D}'$  if  $d(\bar{\mathcal{D}}', \bar{\mathcal{D}}) < 0.1$  and  $d(sd(\mathcal{D}'), sd(\mathcal{D})) < 0.2$ .
- Initialize MH algorithm
- Add parameters that generated  $\mathcal{D}'$  to chain with probability  $h$  if  $\mathcal{D}'$  is accepted.





Trace and marginal distributions of the posterior sample