

ABC estimation of a substitution rate from sequence data

Geoff Nicholls

24th of March 2010

24th March 2010:

ABC estimation of a substitution rate

9:30 ABC

10:30 Practical on ABC

11:30 Tree models and substitution models

14:30 ABC for Bayesian inference

15:00 Practical on ABC for substitution rate
from sequence data.

ABC

(A version of the) Rejection algorithm

Aim: draw $X \sim \pi(x|D) \propto p(D|x)p(x)$

1. draw $z \sim p(z)$ and $u \sim U(0, 1)$
 2. while $u > p(D|z)$ do
 - $z \sim p(z)$
 - $u \sim U(0, 1)$
- end
- return $X = z$

Note can do much better: at 1. $Z \sim q(z)$ and at 2. the test is $u > p(D|z)p(z)/cq(z)$ with $c = \max_z p(D|z)p(z)/q(z)$.

Example $D_i \sim \text{Poisson}(\lambda), i = 1, 2, \dots, N,$

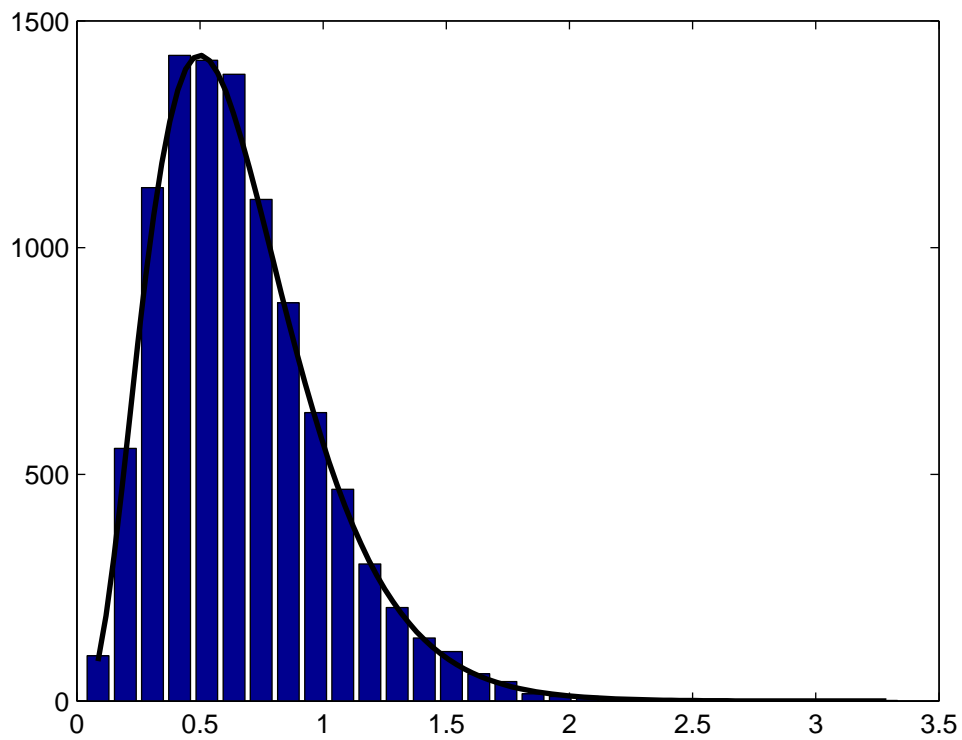
Prior $p(\lambda) = \exp(-\lambda/\mu)/\mu$

Likelihood $p(D|\lambda) = \exp(-N\lambda) \prod_i \lambda^{D_i}/D_i!$

Posterior $\pi(\lambda|D) \propto p(\lambda)p(D|\lambda).$

```
%synthetic data
lambda=0.5; N=5;
D=Poisson(lambda,N);
FP=prod(factorial(D));

%rejection sampling pi(lambda|D)
mu=1;
M=10000;
for k=1:M
    Z=-mu*log(rand);
    U=rand;
    while U>prod(Z.^D)*exp(-N*Z)/FP
        Z=-mu*log(rand);
        U=rand;
    end
    x(k)=Z;
end
```



A histogram of sampled x from the previous algorithm, with the exact posterior superimposed.

True λ was 0.5 and prior mean $\mu = 1$.

(Another version of the) Rejection algorithm

Aim: draw $X \sim \pi(x|D) \propto p(D|x)p(x)$

1. draw $z \sim p(z)$ and $D' \sim p(D'|z)$
 2. while $D' \neq D$ do
 - $z \sim p(z)$
 - $D' \sim p(D'|z)$end
- return $X = z$

The point is $D' = D$ at 2. is an event which occurs with probability $p(D|z)$ so this is algorithm is the same as the last.

```

lambda=0.5; N=5;
D=Poisson(lambda,N);           %D=sort(D);

mu=1;
M=10000;
x=zeros(1,M);
for k=1:M
    Z=-mu*log(rand);
    Dp=Poisson(Z,N);           %Dp=sort(Dp);
    while any(Dp~=D)
        Z=-mu*log(rand);
        Dp=Poisson(Z,N);
    end
    x(k)=Z;
end

```

This would be *very* slow.

Why sort? It is more efficient. Clearly we have a better chance of getting an accept.

Why is it OK to sort? The synthetic data is sorted. We are effectively defining a new observation process, still parameterized by λ .

(A version of) ABC Rejection algorithm

Aim: draw $X \sim \pi(x|D) \propto p(D|x)p(x)$ (approximately)

Fix $\epsilon > 0$ and some measure of distance $d(D', D)$ between 'data sets'.

1. draw $z \sim p(z)$ and $D' \sim p(D'|z)$
 2. while $d(D', D) > \epsilon$ do
 - $z \sim p(z)$
 - $D' \sim p(D'|z)$
- end
return $X = z$

Why do we like this? We need only simulate the data in order to compute the acceptance probability. We don't need actually to evaluate the likelihood. This is the crucial simplification. It is only approximate but perhaps the model of nature was not so hot we want to pay a big price to enforce it exactly. The choice of distance definition is crucial. We choose ϵ so that we generate samples at a useful rate, and otherwise as small as possible. Tavaré et al. (1997, Genetics) got the ball rolling.

(AVO) MCMC ABC

Aim: $X_t = x \rightarrow \pi(x|D) \propto p(x)p(D|x)$

Suppose $X_t = x$, and proposal $q(z|x)$ is given.

Then X_{t+1} is determined in the following way

1. set $X_{t+1} = x$
2. draw $z \sim q(z|x)$ and $D' \sim p(D'|z)$
3. set $\alpha = 1 \wedge \frac{p(z)q(x|z)}{p(x)q(z|x)}$
and draw $U \sim U(0, 1)$
4. if $d(D', D) < \epsilon$ and $U < \alpha$ set $X_{t+1} = z$

Verify transition matrix $P_{x,z}$ satisfies $\pi = \pi P$.

Check $\pi(x|D)P_{x,z} = \pi(z|D)P_{z,x}$

Much more efficient than rejection-style ABC, but, some of the disadvantages of regular MCMC (no likelihood evaluation, but must build updates $q(z|x)$ which may necessitate complex supporting data structures. See coalescent example later.)

1) Implement Rejection-ABC for exp-poisson.

Define the distance as $|\text{mean}(D) - \text{mean}(D')|$.

Accept if that distance is sufficiently small.