Advanced Statistical Computing

Fall 2018

Steve Qin

Metropolis-Hastings Algorithm

- Start with any $X^{(0)}=x_0$, and a "*proposal chain*" T(x,y)
- Suppose $X^{(t)}=x_t$. At time t+1,
  - Draw y~T$(x_t,$y$)$ (i.e., propose a move for the next step)
  - Compute ***"goodness ratio"***

$$r = \frac{\pi(y)T(y,x_t)}{\pi(x_t)T(x_t,y)}$$

  - **Acceptance/Rejection decision:** Let

$$X^{(t+1)} = \begin{cases} y, & \text{with } p = \min\{1,r\} \\ x_t, & \text{with } 1-p \end{cases}$$

Illustration of Metropolis-Hastings

- At each step, calculate $r = \dfrac{\pi(x_{t+1}, y_{t+1})}{\pi(x_t, y_t)}$

$$T((x_t, y_t), (x_{t+1}, y_{t+1})) = T((x_{t+1}, y_{t+1}), (x_t, y_t)) = 1/\pi^2$$

since

$$r = \frac{\frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)} \left(x_{t+1}^2 - 2\rho x_{t+1} y_{t+1} + y_{t+1}^2 \right) \right)}{\frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)} \left(x_t^2 - 2\rho x_t y_t + y_t^2 \right) \right)}$$

$$= \exp\left(-\frac{1}{2(1-\rho^2)} \left(\left(x_{t+1}^2 - 2\rho x_{t+1} y_{t+1} + y_{t+1}^2 \right) - \left(x_t^2 - 2\rho x_t y_t + y_t^2 \right) \right) \right)$$

Summary on M-H algorithm

- Needs to know the density function of the target distribution (not necessarily to be complete)
- Up to a normalizing constant
- Easy to implement, ideal for homogeneous set of parameters (not too many).
- Require burn-in
- Monitor convergence

Gibbs Sampler

- Purpose: Draw random samples form a joint distribution (high dimensional)

$$x = (x_1, x_2, ..., x_n) \quad \text{Target} \quad \pi(x)$$

- Method: Iterative conditional sampling

$$\forall i, \quad \text{draw } x_i \sim \pi(x_i \mid x_{[-i]})$$

Illustration of Gibbs Sampler

- Suppose the target distribution is:

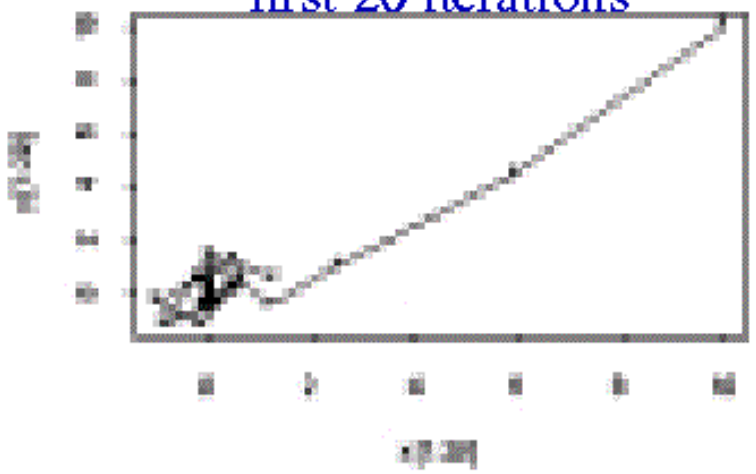
$$(X,Y) \sim N\left(\begin{pmatrix}0\\0\end{pmatrix},\begin{pmatrix}1 & \rho\\ \rho & 1\end{pmatrix}\right)$$

- Gibbs sampler:

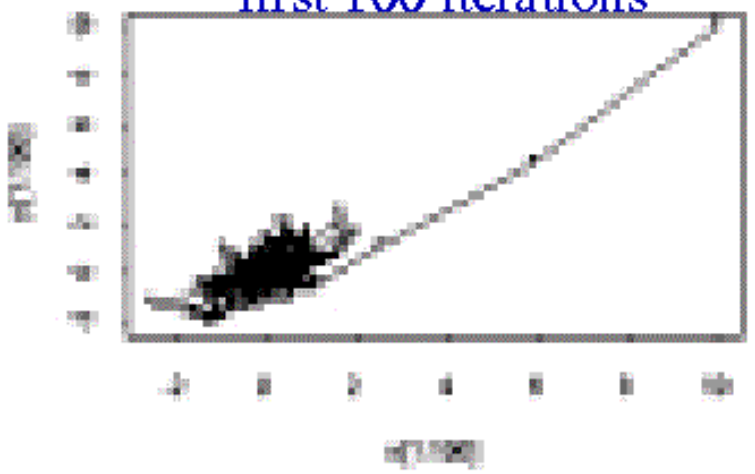
$$[X|Y=y] \sim N(\rho y, 1-\rho^2)$$
$$[Y|X=x] \sim N(\rho x, 1-\rho^2)$$

Start from, say, (X,Y)=(10,10), we can take a look at the trajectories. We took ρ=0.6.

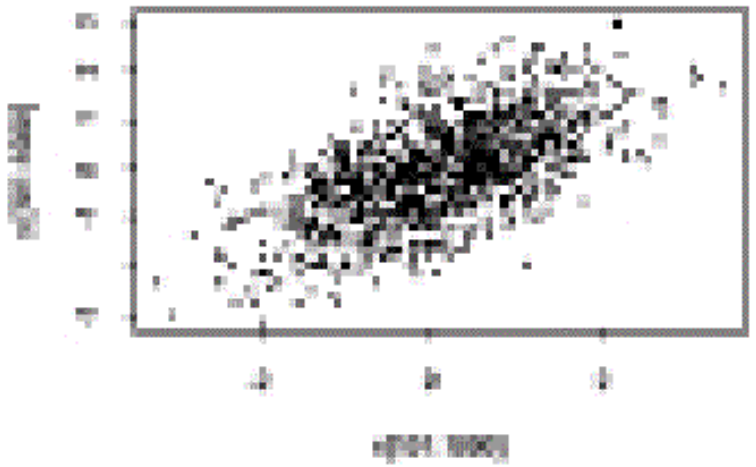
first 20 iterations



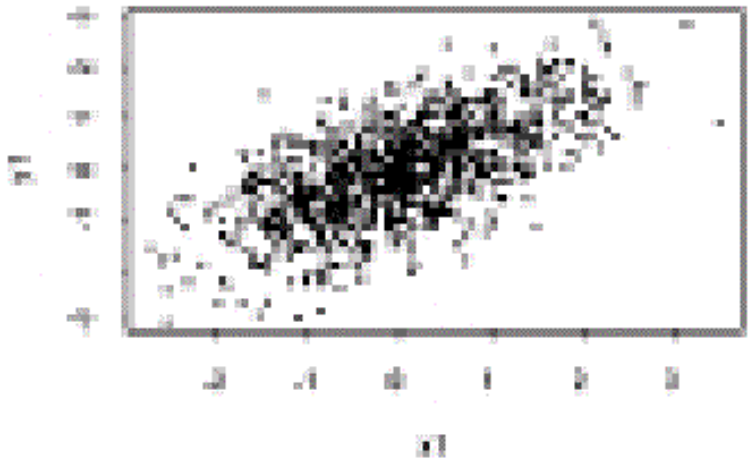
first 100 iterations



101-1000 iterations



900 iid samples



References

- Geman and Geman 1984,
- Gelfand and Smith 1990,
- Tutorial paper:

  Casella and George (1992). Explaining the Gibbs sampler. *The American Statistician*, **46**, 167-174.

Summary of Gibbs sampler

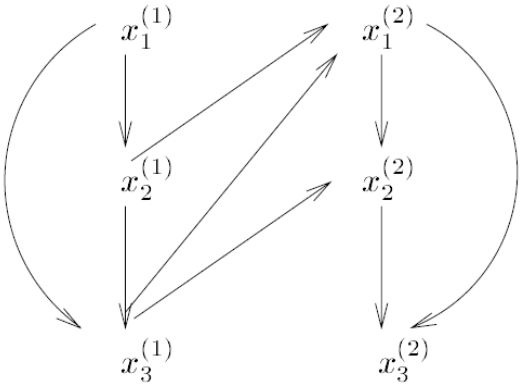
- Special case of MH algorithm
- The most popular MCMC method, generally more efficient than MH.
- Require some mathematical derivation.
- Verify convergence of the sequence
- Use multiple chains
- Be careful of *pseudo* Gibbs sampler!

Collapsing, predictive updating

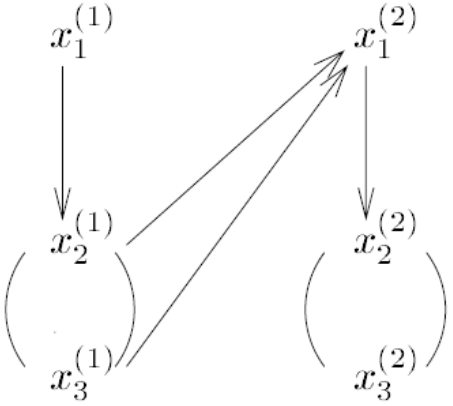
Collapsing and grouping

- Want to sample from $\mathbf{x} = (x_1, x_2, ..., x_d)$
- Regular Gibbs sampler:
  - Sample $x_1^{(t+1)}$ from $\pi(x_1^{(t+1)} \mid x_2^{(t)}, x_3^{(t)}, ..., x_d^{(t)})$,
  - Sample $x_2^{(t+1)}$ from $\pi(x_2^{(t+1)} \mid x_1^{(t)}, x_3^{(t)}, ..., x_d^{(t)})$,
  - …
  - Sample $x_d^{(t+1)}$ from $\pi(x_d^{(t+1)} \mid x_2^{(t)}, x_3^{(t)}, ..., x_{d-1}^{(t)})$,
- Alternatively:
  - Grouping: $\mathbf{x}_{d-1}' = (x_{d-1}, x_d)$.
  - Collapsing, i.e., integrate out x_d: $\mathbf{x}^{-} = (x_1, x_2, ..., x_{d-1})$

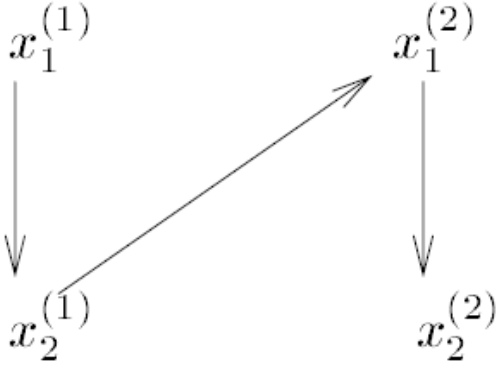
The three-schemes



standard



grouping



collapsing

Some theory

- Hilbert space $L_2(\pi)$ of functions $h()$.
- Define $\langle h, g \rangle = E_\pi \{h(x)g(x)\}$, thus $\|h\| = \mathrm{var}_\pi(h)$.
- Define forward operator $\boldsymbol{F}$ as

$$Fh(x) = \int K(x,y)h(y)dy = E_\pi \left\{ h\left(x^{(t+1)}\right) \middle| x^{(t)} = x \right\}.$$

$$\|F\| = \sup_h \|Fh(x)\| \text{ for all fucntions with } E(h^2) = 1.$$

- The convergence of Markov chains is tied to the norms of the corresponding forward operators.

Three-scheme theorem

- Standard $\boldsymbol{F_s}$: $x_1 \to x_2 \to \cdots \to x_d$;
- Grouping $\boldsymbol{F_g}$: $x_1 \to x_2 \to \cdots \to \{x_{d-1}, x_d\}$;
- Collapsing $\boldsymbol{F_c}$: $x_1 \to x_2 \to \cdots \to x_{d-1}$.

Theorem The norms of the three forward operators are ordered as

$$\|F_c\| \le \|F_g\| \le \|F_s\|$$

Examples

- Murray's data
- Bivariate Gaussian with mean 0 and unknown covariance matrix Σ

1	1	−1	−1	2	2	−2	−2	∗	∗	∗	∗
1	−1	1	−1	∗	∗	∗	∗	2	2	−2	−2

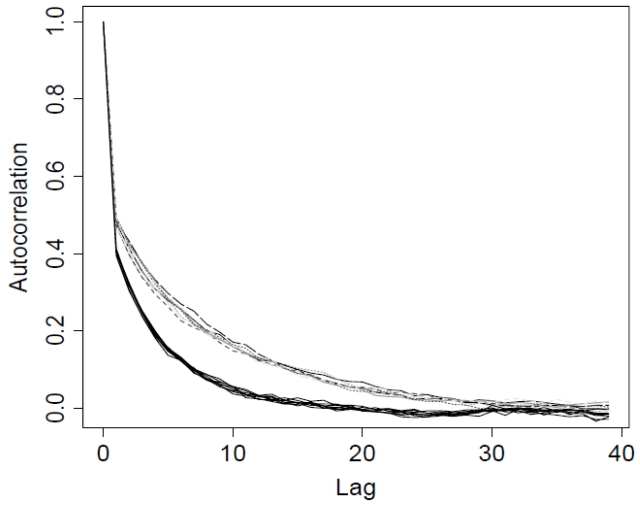
standard

$\Sigma \mid y_{obs}, y_{mis},$

$y_{mis} \mid y_{obs}, \Sigma.$

collapsing

$y_{mis,i} \mid y_{obs}, y_{mis,[-i]}.$



Remarks

- Avoid introducing unnecessary parameters into a Gibbs sampler,
- Do as much analytical work as possible,
- However, introducing some clever auxiliary variables can greatly improve computation efficiency.

Convergence Diagnostics

Patrick Lam

Within Chain Variance

$$W = \frac{1}{m} \sum_{j=1}^{m} s_j^2$$

where

$$s_j^2 = \frac{1}{n-1} \sum_{i=1}^{n} (\theta_{ij} - \bar{\theta}_j)^2$$

s_j^2 is just the formula for the variance of the jth chain. W is then just the mean of the variances of each chain.

W likely underestimates the true variance of the stationary distribution since our chains have probably not reached all the points of the stationary distribution.

Between Chain Variance

$$B = \frac{n}{m-1} \sum_{j=1}^{m} (\bar{\theta}_j - \bar{\bar{\theta}})^2$$

where

$$\bar{\bar{\theta}} = \frac{1}{m} \sum_{j=1}^{m} \bar{\theta}_j$$

This is the variance of the chain means multiplied by n because each chain is based on n draws.

Estimated Variance

We can then estimate the variance of the stationary distribution as a weighted average of W and B.

$$\hat{\mathrm{Var}}(\theta) = (1 - \frac{1}{n})W + \frac{1}{n}B$$

Because of overdispersion of the starting values, this overestimates the true variance, but is unbiased if the starting distribution equals the stationary distribution (if starting values were not overdispersed).

Gelman and Rubin Multiple Sequence Diagnostic

Steps (for each parameter):

1. Run $m \geq 2$ chains of length $2n$ from overdispersed starting values.
2. Discard the first n draws in each chain.
3. Calculate the within-chain and between-chain variance.
4. Calculate the estimated variance of the parameter as a weighted sum of the within-chain and between-chain variance.
5. Calculate the potential scale reduction factor.

Potential Scale Reduction Factor

The potential scale reduction factor is

$$\hat{R} = \sqrt{\frac{\hat{\mathrm{Var}}(\theta)}{W}}$$

When $\hat{R}$ is high (perhaps greater than 1.1 or 1.2), then we should run our chains out longer to improve convergence to the stationary distribution.

If we have more than one parameter, then we need to calculate the potential scale reduction factor for each parameter.

We should run our chains out long enough so that all the potential scale reduction factors are small enough.

We can then combine the mn total draws from our chains to produce one chain from the stationary distribution.

Sequential Monte Carlo

Accept-reject method

- The accept-reject method
  1. Generate $X \sim g$, $U \sim Uniform(0,1)$,
  2. Accept $Y = X$ *if* $U \leq f(X)/Mg(X)$,
  3. Repeat.

Accept-reject method example

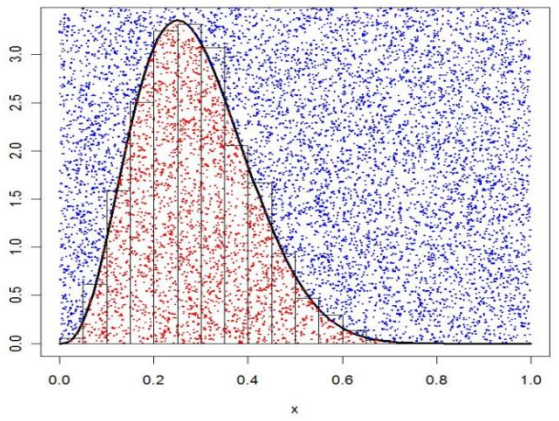
- *Beta* (α,β), $\alpha \geq 1$, $\beta \geq 1$,

  simulate $Y \sim Uniform(0,1)$ and

  $$U \sim Uniform(0,m),$$

  m is the max of the Beta density.

  select $X = Y$ if under curve



what is the acceptance rate?

Importance sampling

- A variance reduction technique.
- Certain values of the input random variables have more impact on the parameter being estimated than others.
- If these "important" values are emphasized by sampling more frequently, then the estimator variance can be reduced.
- Marshall (1956) suggested that one should focus on the region(s) of “importance” so as to save computational resource.
- Essential in high-dimensional models.

Importance sampling

- *Importance sampling:*

  to evaluate $E_f[h(X)] = \int_{\aleph} h(x) f(x) dx$

  based on generating a sample $X_1, \cdots, X_n$ from a given distribution g and approximating

$$E_f[h(X)] \approx \frac{1}{m} \sum_{j=1}^{m} \frac{f(X_j)}{g(X_j)} h(X_j)$$

  which is based on

$$E_f[h(X)] = \int_{\aleph} h(x) \frac{f(x)}{g(x)} g(x) dx$$

The algorithm

- To evaluate $\mu = E_\pi[h(X)] = \int_{\mathcal{X}} h(x)\pi(x)dx.$
  - Draw $x^{(1)},\ldots, x^{(m)}$ from a trial distribution $g()$.
  - Calculate the *importance weight*

$$w^{(j)} = \pi\left(x^{(j)} / g(x^{(j)})\right), \quad \text{for } j = 1,\ldots, m.$$

  - Approximate μ by $\hat{\mu} = \dfrac{w^{(1)}h(x^{(1)}) + \cdots + w^{(m)}h(x^{(m)})}{w^{(1)} + \cdots + w^{(m)}}.$

- Remark: $\hat{\mu}$ is better than the unbiased estimator $\tilde{\mu} = \dfrac{1}{m}\{w^{(1)}h(x^{(1)}) + \cdots + w^{(m)}h(x^{(m)})\}.$ why?

The basic methodology of importance sampling

- To choose a distribution which "encourages" the important values.
- This use of "biased" distributions will result in a biased estimator if it is applied directly.
- Weight to correct for the use of the biased distribution to ensure unbiasness.
- The weight is given by the likelihood ratio,

Importance sampling example

- Small tail probabilities:

  $Z \sim N(0,1)$, $P(Z > 4.5)$

  naïve: simulate $Z_i \sim N(0,1)$, $i=1,\ldots,M$.

  calculate

$$P(Z > 4.5) \approx \frac{1}{M} \sum_{i=1}^{M} I(Z_i > 4.5)$$

Importance sampling example

Let $Y \sim TExp(4.5,1)$ with density

$$f_Y(y) = e^{-(x-4.5)} / \int_{4.5}^{\infty} e^{-x} dx.$$

Now simulated from f_Y and use importance sampling, we obtain

$$P(Z > 4.5) \approx \frac{1}{M} \sum_{i=1}^{M} \frac{\varphi(Y_i)}{f_Y(Y_i)} I(Y_i > 4.5) = .000003377.$$

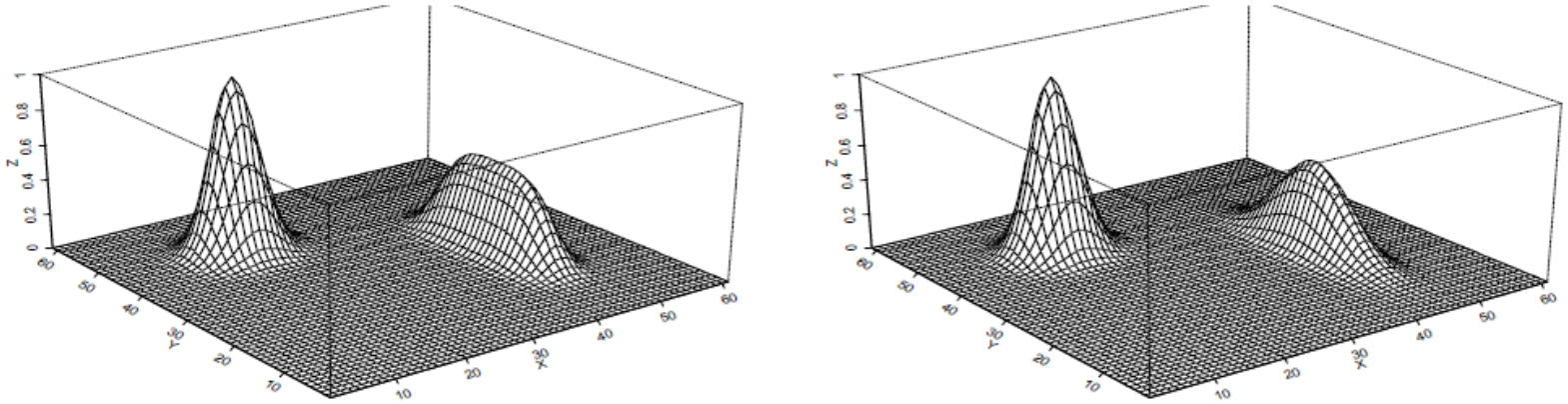
Importance sampling example

```
## theoretical value
p0=1-pnorm(4.5)
## sample directly from normal distribution
## this needs large number of samples
z=rnorm(10000000)
p1=mean(z>4.5)

## importance sampling
n0=10000
Y=rexp(n0, 1)+4.5
a=dnorm(Y)/dexp(Y-4.5)
p2=mean(a[Y>4.5])

c(p0, p1, p2) ##
[1] 3.397673e-06 2.600000e-06 3.418534e-06
```

Another example

$$f(x,y) = 0.5e^{-90(x-0.5)^2-45(y+0.1)^4} + e^{-45(x+0.4)^2-60(y-0.5)^2}$$

(a) (b)



- Both grid-point method and vanilla Monte Carlo methods wasted resources on “boring” desert area.

Another example

- Use proposal function

$$g(x,y) \propto 0.5e^{-90(x-0.5)^2-10(y+0.1)^2} + e^{-45(x+0.4)^2-60(y-0.5)^2},$$

with $(x,y) \in [-1,1] \times [-1,1]$, a truncated mixture of bivariate Gaussian

$$0.46\mathcal{N}\left[\begin{pmatrix} 0.5 \\ -0.1 \end{pmatrix}, \begin{pmatrix} \frac{1}{180} & 0 \\ 0 & \frac{1}{20} \end{pmatrix}\right] + 0.54\mathcal{N}\left[\begin{pmatrix} -0.4 \\ 0.5 \end{pmatrix}, \begin{pmatrix} \frac{1}{90} & 0 \\ 0 & \frac{1}{120} \end{pmatrix}\right]$$

Vanilla Monte Carlo

$\hat{\mu} = 0.1307$

$std(\hat{\mu}) = 0.009$

Importance Sampling

$\hat{\mu} = 0.1259$

$std(\hat{\mu}) = 0.0005$

Sequential importance sampling

- For high dimensional problem, how to design trial distribution is challenging.
- Suppose the target density of $\mathbf{x} = (x_1, x_2, ..., x_d)$ can be decomposed as

$$\pi(\mathbf{x}) = \pi(x_1)\pi(x_2 \mid x_1) \cdots \pi(x_d \mid x_1, .., x_{d-1})$$

then constructed trial density as

$$g(\mathbf{x}) = g_1(x_1)g_2(x_2 \mid x_1) \cdots g_d(x_d \mid x_1, .., x_{d-1})$$

Sequential importance sampling

$$w(\mathbf{x}) = \frac{\pi(x_1)\pi(x_2 \mid x_1)\cdots\pi(x_d \mid x_1,.., x_{d-1})}{g_1(x_1)g_2(x_2 \mid x_1)\cdots g_d(x_d \mid x_1,.., x_{d-1})}$$

Suggest a recursive way of computing and monitoring importance weight. Denote

$$\mathbf{x_t} = (x_1, x_2,..., x_t)$$

then we have

$$w_t(\mathbf{x_t}) = w_{t-1}(\mathbf{x_{t\text{-}1}})\frac{\pi(x_t \mid \mathbf{x_{t\text{-}1}})}{g_t(x_t \mid \mathbf{x_{t\text{-}1}})}$$

Sequential importance sampling

- Advantages of the recursion scheme
  - Can stop generating further components of x if the partial weight is too small.
  - Can take advantage of $\pi(x_t \mid \mathbf{x_{t-1}})$ in designing $g_t(x_t \mid \mathbf{x_{t-1}})$
- However, the scheme is impractical since requires the knowledge of marginal distribution $\pi(\boldsymbol{x}_t)$.

Sequential importance sampling

- Add another layer of complexity:
- Introduce a sequence of "auxiliary distributions" $\pi_1(x_1)\pi_2(\mathbf{x_2})\pi_d(\mathbf{x})$ such that $\pi_t(\mathbf{x_t})$ is a reasonable approximation of the marginal distribution $\pi(\mathbf{x_t})$, for $t = 1,\ldots,d$ -1 and $\pi_d = \pi$.
- Note the π_d are only required to be known up to a normalizing constant.

The SIS procedure

For $t = 2,\ldots,d$,

- Draw $X_t = x_t$ from $g_t(x_t \mid x_{t-1})$, and let

  $\mathbf{x_t} = (\mathbf{x_{t-1}}, x_t)$

- Compute $u_t = \dfrac{\pi_t(\mathbf{x_t})}{\pi_{t-1}(\mathbf{x_{t-1}})\, g_t(x_t \mid \mathbf{x_{t-1}})}$

  and let $w_t = w_{t-1}\, u_t$

- u_t : incremental weight.
- The key idea is to breaks a difficult task into manageable pieces.
- If w_t is getting too small, reject.

An application example of SIS

- Assume
  - Constant population size N,
  - Evolve in non-overlapping generation,
  - The chromosomal region is sufficiently small,
  - No recombination,
  - “haplotype”: each chromosome only has one parent.

Population genetics example

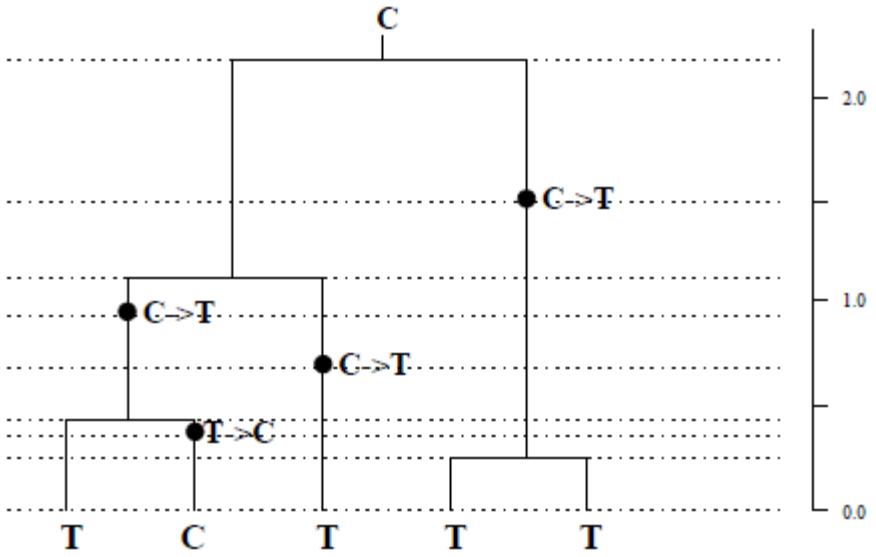
- Notation:
  - E: set of all possible genetic types,
  - μ: mutation rate per chromosome per generation,
  - $P = (P_{\alpha\beta})$: the mutation transition matrix,
  - If a parental segment of type $\alpha \in E$,

its progeny is $\begin{cases} \alpha & \text{with prob.} 1-\mu, \\ \beta & \text{with prob. } \mu P_{\alpha\beta}. \end{cases}$

Example data

- From Stephens and Donnelly (2000)
- $E=\{C,T\}$
- The history $H = (H_{-k}, H_{-(k-1)},\ldots, H_{-1}, H_0)$

$= (\{C\}, \{C,C\}, \{C,T\}, \{C,C,T\}, \{C,T,T\}, \{T,T,T\},$
$\{T,T,T,T\}, \{C,T,T,T,T\}, \{C,T,T,T,T\})$



Coalescence example

- Use $H = (H_{-m}, \ldots, H_{-1}, H_0)$ to denote the whole ancestral history (unobserved) of the 5 individuals.
- Compute the likelihood function

$$p_\theta(\boldsymbol{H}) = p_\theta(H_{-k})\, p_\theta(H_{-k+1} \mid H_{-k}) \cdots p_\theta(H_0 \mid H_{-1})\, p_\theta(\text{stop} \mid H_0)$$

$p_\theta(H_{-k}) = \pi_0(H_{-k})$, π_0 is the stationary distribution of P.

$$p_\theta(H_i \mid H_{i-1}) = \begin{cases} \dfrac{n_\alpha}{n} \dfrac{\theta}{n-1+\theta} P_{\alpha\beta} & \text{if } H_i = H_{i-1} - \alpha + \beta \\ \dfrac{n_\alpha}{n} \dfrac{n-1}{n-1+\theta} & \text{if } H_i = H_{i-1} + \alpha \\ 0 & \text{otherwise,} \end{cases}$$

Coalescence calculation

- For i = -(k-1),…,0

$$p_\theta(H_i \mid H_{i-1}) = \begin{cases} \dfrac{n_\alpha}{n}\dfrac{\theta}{n-1+\theta}P_{\alpha\beta} & \text{if } H_i = H_{i-1} - \alpha + \beta \\ \dfrac{n_\alpha}{n}\dfrac{n-1}{n-1+\theta} & \text{if } H_i = H_{i-1} + \alpha \\ 0 & \text{otherwise,} \end{cases}$$

$$p_\theta(\text{stop}|H_0) = \sum_\alpha \frac{n_\alpha}{n}\frac{n-1}{n-1+\theta}.$$

Notations

- n is the sample size at generation $H_{i\text{-}1}$
- n_α is the number of chromosome of type α in the sample.
- $\theta=2N\mu/v$.
- N population size.
- v^2 is the variance of the number of progeny of a random chromosome.

Strategies to estimate θ

- To get MLE, we need to compute likelihood

$$p_\theta(H_0) = \sum_{\mathcal{H}:\text{compatible with } H_0} p_\theta(\mathcal{H}).$$

- Naïve Monte Carlo won't work because of compatibility issue.
- An alternative is to simulate $\boldsymbol{H}$ backward starting from H_0 and use weight to correct bias.

An SIS approach

- Simulate H_{-1}, H_{-1},…, from a trial distribution built up sequentially by revering the forward sampling probability at a fixed θ_0. That is, for $i = 1,\ldots,k$, we have

$$g_t(H_{-t} \mid H_{-t+1}) = \frac{p_{\theta_0}(H_{-t} \mid H_{-t+1})}{\sum_{\text{all } H'_{-t}} p_{\theta_0}(H_{-t} \mid H'_{-t+1})},$$

the final trial distribution

$$g(\boldsymbol{H}) = g_1(H_{-1} \mid H_0) \cdots g_k(H_{-k} \mid H_{-k+1})$$

An SIS approach

- By simulating from $g()$ multiple copies of the history, $H^{(j)}$, j=1,…,m, we can approximate the likelihood function as

$$\hat{p}_\theta(H_0) = \frac{1}{m}\sum_{j=1}^{m}\frac{p_\theta(H^{(j)})}{g(H^{(j)})}.$$

- Note the choice of θ_0 can influence the final result.

Other examples of SIS

- Growing a polymer
  - Self avoid walk
- Sequential imputation for statistical missing data problem.
- More and details of these examples, see Liu 2001.