STOCHASTIC SIMULATION FOR BLOCKED DATA

- **Monte Carlo simulation**
- **Rejection sampling**
- **Importance sampling**
- **Markov chain Monte Carlo**

Monte Carlo simulation

Introduction:

If we know how to directly sample from $f(x)$ to obtain i.i.d. samples $\left\{\hat{X}^{(i)} : i = 1, ..., N\right\}$, according to the Law of Large Number:

$$E_{f(x)}\left(g(X)\right) \approx \frac{1}{N}\sum_{i=1}^{N} g\left(\hat{X}^{(i)}\right) \equiv \hat{g}^{MCS}$$

Procedure: skipped since it is trivial

Analysis:

1. The MCS estimator $\hat{g}^{MCS}$ is unbiased and the variance decays with the 1/N rate.

$$E_{f(x)}\left[\hat{g}^{MCS}\right] = \frac{1}{N} E_{f(x)}\left[\sum_{i=1}^{N} g\left(X^{(i)}\right)\right] = E_{f(x)}\left[g(X)\right]$$

$$Var_{f(x)}\left[\hat{g}^{MCS}\right] = \frac{1}{N^2}\sum_{i=1}^{N} Var_{f(x)}\left[g\left(X^{(i)}\right)\right] = \frac{1}{N} Var_{f(x)}\left[g(X)\right]$$

2. Confidence interval

MCS estimator $\hat{g}^{MCS} \equiv \frac{1}{N}\sum_{i=1}^{N} g\left(X^{(i)}\right)$

$$\xrightarrow[N\to\infty]{\text{Central Limit Theorem}} \text{Normal}\left(E_{f(x)}\left(g(X)\right), \frac{Var_{f(x)}\left(g(X)\right)}{N}\right)$$

To build up confidence interval, we need $Var_{f(x)}\left(g(X)\right)$, which can be estimated simply as the sample variance of $\left\{g\left(\hat{X}^{(i)}\right) : i = 1, ..., N\right\}$, i.e.

$$\hat{V}ar_{f(x)}\left(g(X)\right) \equiv \frac{1}{N}\sum_{i=1}^{N}\left[g\left(\hat{X}^{(i)}\right) - \hat{g}^{MCS}\right]^2$$

Therefore, the following 95.4% confidence interval for Gaussian can be built:

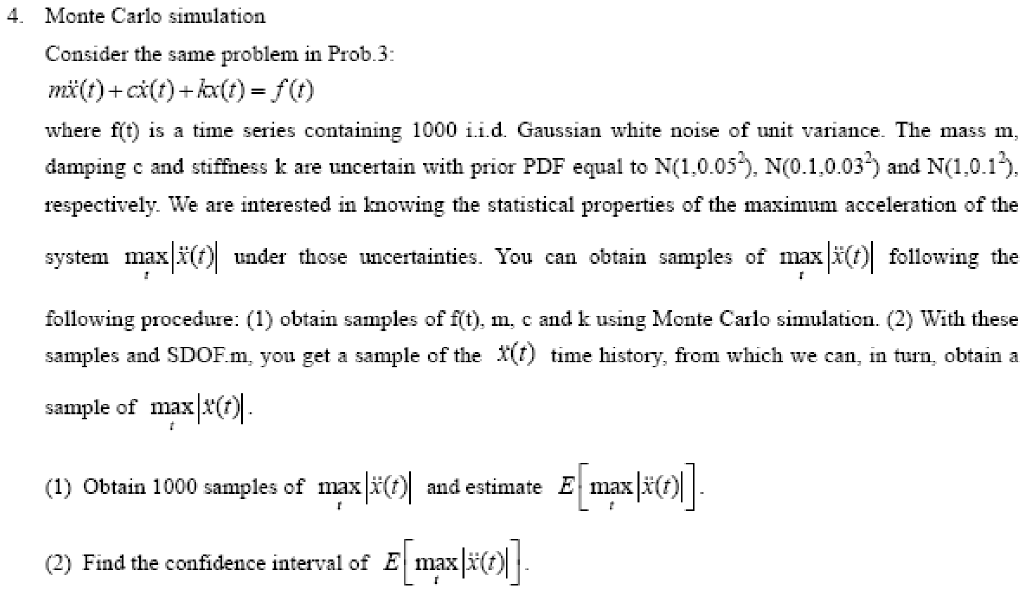
$$\hat{g}^{MCS}-2\sqrt{\frac{\hat{V}ar_{f(x)}\left(g\left(X\right)\right)}{N}}\le E_{f(x)}\left[g\left(X\right)\right]\le\hat{g}^{MCS}+2\sqrt{\frac{\hat{V}ar_{f(x)}\left(g\left(X\right)\right)}{N}}$$

Remarks:

1. "If we know how to directly sample from $f(x)$" is a very strict premise since it usually requires the knowledge of $F^{-1}(\cdot)$ (the inverse of the cumulative density function), which is available to us only for certain standard probability density functions, e.g. uniform, Gaussian, Gamma, etc. In the case that we know $F^{-1}(\cdot)$, we can sample from $f(x)$ using the following steps: Draw $U\sim unif[0,1]$ and let $X=F^{-1}(U)$. It can be shown that $X\sim f(x)$.

   Proof: $P(X\le x)=P\left(F^{-1}(U)\le x\right)=P\left(U\le F(x)\right)=F(x)$

2. The accuracy of $\hat{g}^{MCS}$ is robust against the dimension of X since $Var\left(\hat{g}^{MCS}\right)$ does not depend on the dimension.



4. Monte Carlo simulation

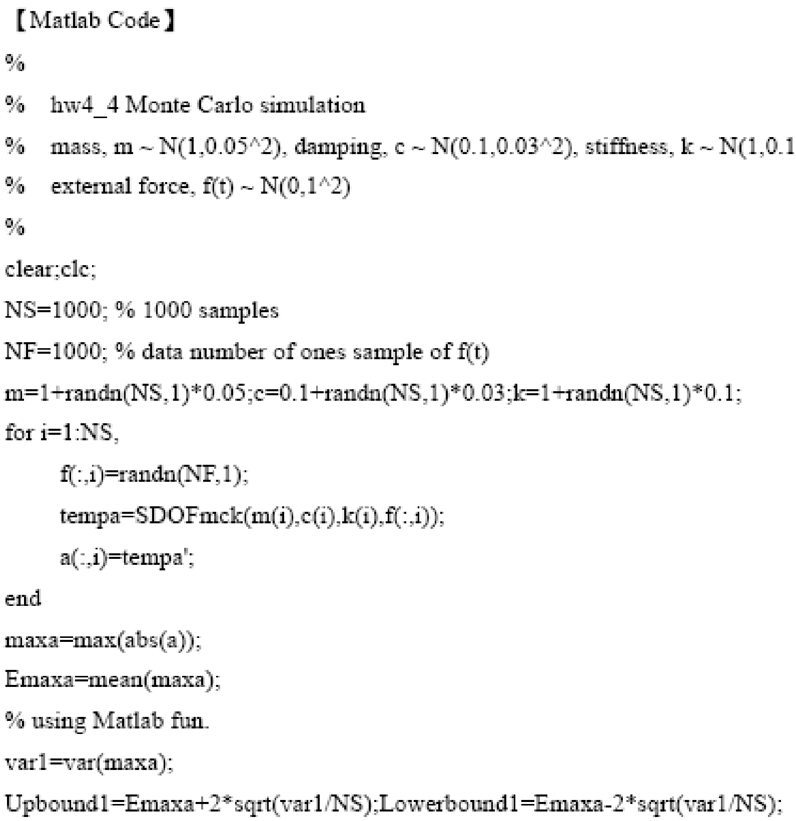
Consider the same problem in Prob.3:

$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=f(t)$

where f(t) is a time series containing 1000 i.i.d. Gaussian white noise of unit variance. The mass m, damping c and stiffness k are uncertain with prior PDF equal to N(1,0.05^2), N(0.1,0.03^2) and N(1,0.1^2), respectively. We are interested in knowing the statistical properties of the maximum acceleration of the system $\max_t|\ddot{x}(t)|$ under those uncertainties. You can obtain samples of $\max_t|\ddot{x}(t)|$ following the following procedure: (1) obtain samples of f(t), m, c and k using Monte Carlo simulation. (2) With these samples and SDOF.m, you get a sample of the $\ddot{x}(t)$ time history, from which we can, in turn, obtain a sample of $\max_t|\ddot{x}(t)|$.

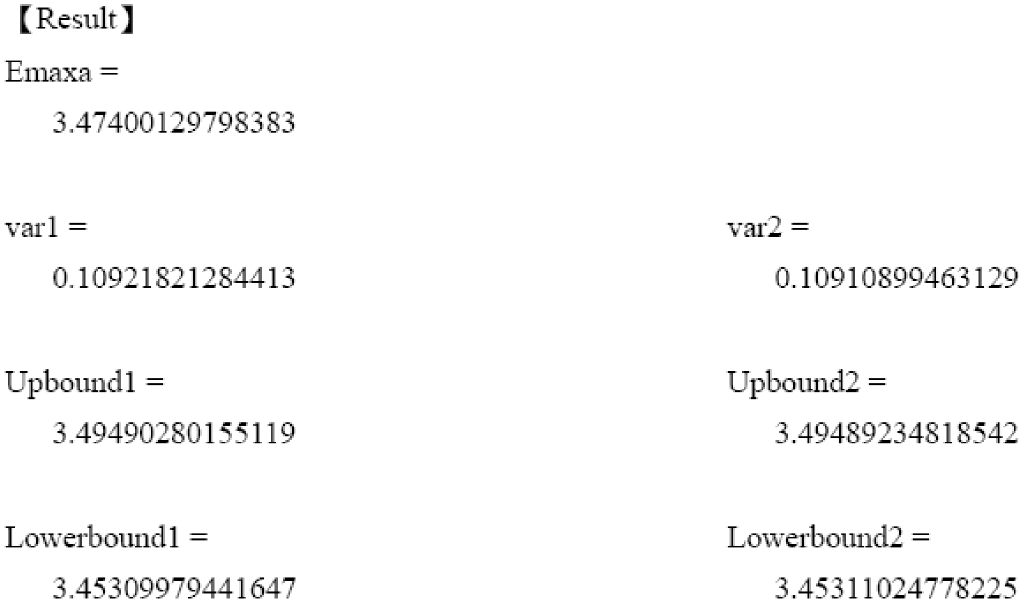
(1) Obtain 1000 samples of $\max_t|\ddot{x}(t)|$ and estimate $E\left[\max_t|\ddot{x}(t)|\right]$.

(2) Find the confidence interval of $E\left[\max_t|\ddot{x}(t)|\right]$.



【Matlab Code】

```
%
%    hw4_4 Monte Carlo simulation
%    mass, m ~ N(1,0.05^2), damping, c ~ N(0.1,0.03^2), stiffness, k ~ N(1,0.1^2),
%    external force, f(t) ~ N(0,1^2)
%
clear;clc;
NS=1000; % 1000 samples
NF=1000; % data number of ones sample of f(t)
m=1+randn(NS,1)*0.05;c=0.1+randn(NS,1)*0.03;k=1+randn(NS,1)*0.1;
for i=1:NS,
        f(:,i)=randn(NF,1);
        tempa=SDOFmck(m(i),c(i),k(i),f(:,i));
        a(:,i)=tempa';
end
maxa=max(abs(a));
Emaxa=mean(maxa);
% using Matlab fun.
var1=var(maxa);
Upbound1=Emaxa+2*sqrt(var1/NS);Lowerbound1=Emaxa-2*sqrt(var1/NS);
```



【Result】

```
Emaxa =
    3.47400129798383


var1 =                                              var2 =
    0.10921821284413                                    0.10910899463129


Upbound1 =                                          Upbound2 =
    3.49490280155119                                    3.49489234818542


Lowerbound1 =                                       Lowerbound2 =
    3.45309979441647                                    3.45311024778225
```

✦ Rejection sampling

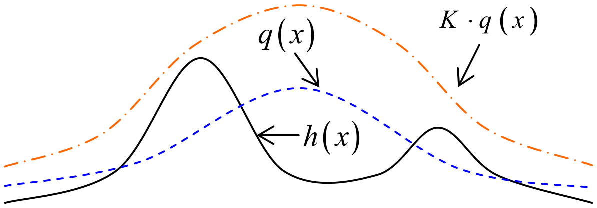
Introduction:

Suppose we don't know how to sample $f(x)$ directly, but we do know how to evaluate $f(x)$ up to a constant. $\rightarrow f(x) = a \cdot h(x)$

where a: unknown constant, $h(x)$: we know how to calculate it. Note that this is the usual situation for Bayesian analysis, where the posterior PDF has the following form:

$$f\left(x \mid \hat{Y}\right) = \frac{f\left(\hat{Y} \mid x\right) f(x)}{f\left(\hat{Y}\right)} = \underbrace{\frac{1}{f\left(\hat{Y}\right)}}_{a} \underbrace{\left[f\left(\hat{Y} \mid x\right) f(x)\right]}_{h(x)}$$

Assume that we know how to sample and evaluate a chosen PDF $q(x)$



Procedure:

1. Let K be a number such that $K \cdot q(x) \geq h(x)$, $\forall x$.
2. Let $\hat{X}^C \sim q(x)$, C: candidate.
3. Accept $\hat{X} = \hat{X}^C$ w.p. $\dfrac{h\left(\hat{X}^C\right)}{Kq\left(\hat{X}^C\right)}$.
4. Cycle 2 → 3 until we get enough accepted samples.

Analysis:

Same as the Monte Carlo simulation except that the number of samples should be the number of the accepted samples.

Remarks:

(1) Different from MCMC, when a sample is rejected, we do not repeat the previous sample.

(2) The number K can be solved as the following: Let $x^* = \arg\max_x \dfrac{h(x)}{q(x)} \Rightarrow K = \dfrac{h(x^*)}{q(x^*)}$, i.e. solving K usually requires optimization.

(3) How do we do Step 3? Let $\hat{U} \sim unif[0,1] \Rightarrow$ Let $\hat{X} = \hat{X}^C$, if $\hat{U} \leq \dfrac{h\left(\hat{X}^C\right)}{Kq\left(\hat{X}^C\right)}$

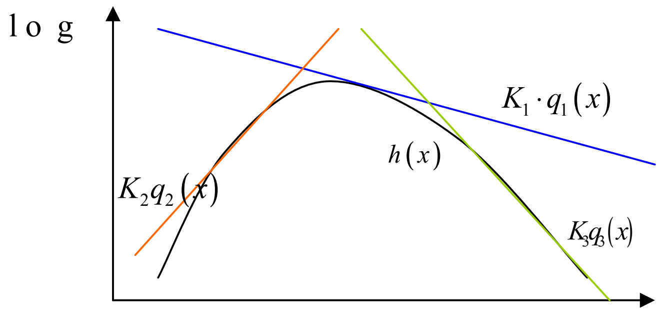
(4) If the shapes of $q(x)$ and $h(x)$ are very different, the efficiency of rejection

sampling will be poor. However, it is often the case that the shape of $h(x)$ is unknown to us. It's hard to choose an efficient $q(x)$.

(5) When the X dimension is high, rejection sampling can be extremely inefficient. This is because it can be much harder to find a good $q(x)$ in high dimensional space.

Adaptive rejection sampling

Suppose $\log h(x)$ is concave.



Procedure:

(1) Let $q_1(x)$ be an exponential PDF and is used as the initial rejection sampling PDF. Note that $K_1q_1(x)$ is tangent to $h(x)$.

(2) Draw $\hat{X}^C \sim q_1(x)$. If $\hat{X}^C$ accepted, keep on using $q_1(x)$ to generate the next sample. However, if $\hat{X}^C$ rejected, let $q_2(x)$ be the second exponential PDF so that $K_2q_2(x)$ is tangent to $h(x)$ at $\hat{X}^C$ in the log space. Generate the next sample using the envelope PDF formed by $K_1q_1(x)$ and $K_2q_2(x)$. Continue doing so until we get enough number of accepted samples.

Remarks:

1. Not applicable to non-log-concave PDF.
2. Still not good for high dimensional X.
3. There are variants of adaptive rejection sampling techniques that do not require evaluating the gradients [1] and do not require log-concaveness [2].

⌖ Importance sampling

Introduction:
Suppose we don’t know how to directly sample from $f(x)$, but we know how to evaluate it. We would like to compute the expected value of some function $g(X)$ with respect to $f(x)$, i.e. to compute

$$E_{f(x)}[g(X)] = \int g(x)f(x)dx$$

Let $q(x)$, called the importance sampling PDF, be a PDF that we know how to sample and evaluate and whose support region contains the support region of $f(x)$, then

$$E_{f(x)}\left[g(X)\right] = \int g(x)f(x)dx = \int g(x)\frac{f(x)}{q(x)}q(x)dx = E_{q(x)}\left[g(x)\frac{f(x)}{q(x)}\right]$$

Procedure:

(1) Draw $\{\hat{X}^{(i)}, i = 1,\cdots,N\}$ i.i.d. from $q(x)$.

(2) According to the Law of Large Number, we have

$$E_{f(x)}\left[g(X)\right] = E_{q(x)}\left[g(X)\frac{f(X)}{q(X)}\right] \approx \frac{1}{N}\sum_{i=1}^{N} g\left(\hat{X}^{(i)}\right)\frac{f\left(\hat{X}^{(i)}\right)}{q\left(\hat{X}^{(i)}\right)} \equiv \hat{g}^{IS}$$

Analysis:

1. The IS estimator $\hat{g}^{IS}$ is unbiased and the variance of $\hat{g}^{IS}$ decays with the 1/N rate.

$$\begin{aligned}
E_{q(x)}\left[\hat{g}^{IS}\right] &= \frac{1}{N}E_{q(x)}\left[\sum_{i=1}^{N} g\left(X^{(i)}\right)\frac{f\left(X^{(i)}\right)}{q\left(X^{(i)}\right)}\right] \\
&= \frac{1}{N}E_{q(x)}\left[N\times\left(g(X)\frac{f(X)}{q(X)}\right)\right] \\
&= E_{q(x)}\left[g(X)\frac{f(X)}{q(X)}\right] \\
&= \int g(x)\frac{f(x)}{q(x)}q(x)dx = E_{f(x)}\left[g(X)\right]
\end{aligned}$$

$$Var_{q(x)}\left[\hat{g}^{IS}\right]=\frac{1}{N^2}\sum_{i=1}^{N}Var_{q(x)}\left[g\left(X^{(i)}\right)\frac{f\left(X^{(i)}\right)}{q\left(X^{(i)}\right)}\right]=\frac{1}{N}Var_{q(x)}\left[g(X)\frac{f(X)}{q(X)}\right]$$

2. Confidence interval

IS estimator $\hat{g}^{IS}\equiv\frac{1}{N}\sum_{i=1}^{N}g\left(X^{(i)}\right)\frac{f\left(X^{(i)}\right)}{q\left(X^{(i)}\right)}$

$$\xrightarrow[N\to\infty]{\text{Central Limit Theorem}}\text{Normal}\left(E_{q(x)}\left(g(X)\frac{f(X)}{q(X)}\right),\frac{Var_{q(x)}\left(g(X)\frac{f(X)}{q(X)}\right)}{N}\right)$$

$$\xrightarrow[N\to\infty]{\text{Central Limit Theorem}}\text{Normal}\left(E_{f(x)}\left(g(X)\right),\frac{Var_{q(x)}\left(g(X)\frac{f(X)}{q(X)}\right)}{N}\right)$$

To build up confidence interval, we need $Var_{q(x)}\left(g(X)\frac{f(X)}{q(X)}\right)$, which can be estimated simply as the sample variance of $\left\{g\left(\hat{X}^{(i)}\right)\frac{f\left(\hat{X}^{(i)}\right)}{q\left(\hat{X}^{(i)}\right)}:i=1,...,N\right\}$, i.e.

$$\hat{V}ar_{q(x)}\left(g(X)\frac{f(X)}{q(X)}\right)\equiv\frac{1}{N}\sum_{i=1}^{N}\left[g\left(\hat{X}^{(i)}\right)\frac{f\left(\hat{X}^{(i)}\right)}{q\left(\hat{X}^{(i)}\right)}-\hat{g}^{IS}\right]^2$$

Therefore, the following 95.4% confidence interval for Gaussian can be built:

$$\hat{g}^{IS}-2\sqrt{\frac{\hat{V}ar_{q(x)}\left(g(X)\frac{f(X)}{q(X)}\right)}{N}}\le E_{f(x)}\left[g(X)\right]\le\hat{g}^{IS}+2\sqrt{\frac{\hat{V}ar_{q(x)}\left(g(X)\frac{f(X)}{q(X)}\right)}{N}}$$

Remarks:

1. Recall that the variance of the MCS estimator is $\frac{1}{N}Var_{f(x)}\left(g(X)\right)$. Here we have seen that the variance of the IS estimator is $\frac{1}{N}Var_{q(x)}\left[g(X)\frac{f(X)}{q(X)}\right]$. It can be shown that if the importance sampling PDF $q(x)\propto g(x)f(x)$, the variance of

the IS estimator is 0. We call this the optimal importance sampling PDF. However, this PDF usually cannot be directly sampled and evaluated. It is also likely to choose an importance sampling PDF so that the variance of the resulting IS estimator is larger than that of MCS estimator.

2. Importance weights $\left\{w^{(i)} : i=1,\ldots,N\right\}$

   Let us define $w^{(i)} \equiv \frac{1}{N}\frac{f\left(\hat{X}^{(i)}\right)}{q\left(\hat{X}^{(i)}\right)}$ as the importance weight of the i^{th} sample. One can see that the IS estimator is simply $\sum_{i=1}^{N} w^{(i)} g\left(\hat{X}^{(i)}\right)$. Note that

$$E_{q(x)}\left[w^{(i)}\right]=\frac{1}{N}E_{q(x)}\left[\frac{f(X)}{q(X)}\right]=\frac{1}{N}$$

   Also,

$$\sum_{i=1}^{N} w^{(i)}=\frac{1}{N}\sum_{i=1}^{N}\frac{f\left(X^{(i)}\right)}{q\left(X^{(i)}\right)}\xrightarrow[N\to\infty]{\text{Central Limit Theorem}}\text{Normal}\left(1,\frac{Var_{q(x)}\left(\frac{f(X)}{q(X)}\right)}{N}\right)$$

$$\Rightarrow\sum_{i=1}^{N} w^{(i)}\xrightarrow{N\to\infty}1$$

3. Importance sampling is inefficient when X dimension is high. This is because when the dimension is high, the importance weights may become highly non-uniform. The consequence is that the number of effective samples is little.

Modified importance sampling

Introduction:

For Bayesian analysis, we usually only know how to evaluate the posterior PDF $f\left(x\,|\,\hat{Y}\right)$ up to a constant:

$$f\left(x\,|\,\hat{Y}\right)=\frac{f\left(\hat{Y}\,|\,x\right)f(x)}{f\left(\hat{Y}\right)}$$

Note that we usually don’t know the normalizing constant $f\left(\hat{Y}\right)$. So the importance sampling technique cannot be directly applied since it requires full evaluation of $f\left(x\mid\hat{Y}\right)$. Modified importance sampling only requires the evaluation of $f\left(x\mid\hat{Y}\right)$ up to a constant. → $f\left(x\mid\hat{Y}\right)=a\cdot h(x)$, where a: we don't know it, $h(x)$: we know how to evaluate it. Now let the normalized importance weight be

$$w^{(i)}\equiv\frac{h\left(\hat{X}^{(i)}\right)}{q\left(\hat{X}^{(i)}\right)}\Bigg/\sum_{j=1}^{N}\frac{h\left(\hat{X}^{(j)}\right)}{q\left(\hat{X}^{(j)}\right)}$$

Note that $\sum_{i=1}^{N}w^{(i)}=1$. The MIS estimator for $E_{f\left(x\mid\hat{Y}\right)}\left[g(X)\right]\equiv E\left[g(X)\mid\hat{Y}\right]$ is simply $\sum_{i=1}^{N}w^{(i)}g\left(\hat{X}^{(i)}\right)$.

Observation: when $N\to\infty$, MIS is the same as IS.

Proof:

When $N\to\infty$, $\frac{1}{N}\sum_{j=1}^{N}\frac{h\left(\hat{X}^{(j)}\right)}{q\left(\hat{X}^{(j)}\right)}\to E_{q(x)}\left[\frac{h(X)}{q(X)}\right]=\frac{1}{a}$, so $\sum_{j=1}^{N}\frac{h\left(\hat{X}^{(j)}\right)}{q\left(\hat{X}^{(j)}\right)}\to\frac{N}{a}$.

Therefore, the normalized importance weight in MIS and the importance weight in IS become identical. The consequence is that MIS is only asymptotically unbiased.

Procedure:

1. Draw $\left\{\hat{X}^{(i)}:i=1,...,N\right\}\sim q(x)$ and compute $w^{(i)}\equiv\frac{h\left(\hat{X}^{(i)}\right)}{q\left(\hat{X}^{(i)}\right)}\Bigg/\sum_{j=1}^{N}\frac{h\left(\hat{X}^{(j)}\right)}{q\left(\hat{X}^{(j)}\right)}$.
2. $E\left[g(X)\mid\hat{Y}\right]\approx\sum_{i=1}^{N}w^{(i)}g\left(\hat{X}^{(i)}\right)$

Remarks:

1. When N is large, the statistical properties of the MIS estimator is similar to those for the IS estimator. In fact, MIS is asymptotically unbiased.
2. MIS can be used to estimate the normalizing constant $f\left(\hat{Y}\right)$ in Bayesian analysis.

   Let $\alpha^{(i)}=\frac{h(\hat{X}_i)}{q(\hat{X}_i)}$ be the non-normalized importance weight. It can be shown that $\frac{1}{N}\sum_{i=1}^{N}\alpha^{(i)}$ is an unbiased estimator of $f\left(\hat{Y}\right)$.

Sample-importance resampling

Introduction:

Both importance sampling and modified importance sampling are mainly for estimating $E\left[g\left(X\right)|\hat{Y}\right]$. If we would like to obtain samples from $f\left(x|\hat{Y}\right)$, we can adopt SIR, which is just one step ahead of MIS. The idea is to resample the MIS samples according to their normalized weights.

Procedure: Recall IIS, we have $\{w^{(i)}:i=1,\cdots,N\}$, $\sum_{i=1}^{N}w^{(i)}=1$

1. Do MIS to obtain $\left\{\left(\hat{X}^{(i)},w^{(i)}\right):i=1,...,N\right\}$. Recall that $\sum_{i=1}^{N}w^{(i)}=1$.
2. Resampling: Let $X_{new}^{(j)}=X^{(i)}$ w.p. $w^{(i)}$ for $j=1,\cdots,M$, then

   $\left\{X_{new}^{(j)}:j=1,\cdots,M\right\}\sim f\left(x|\hat{Y}\right)$

Remarks:

1. There may be repeated samples in $\left\{X_{new}^{(j)}:j=1,\cdots,M\right\}$.
2. M is better to be no larger than N.
3. If N is small, $\left\{X_{new}^{(j)}:j=1,\cdots,M\right\}\nsim f\left(x|\hat{Y}\right)$. This is because MIS is only asymptotically unbiased.

1. Improved importance sampling (IIS)

   Consider the following SDOF system:

   $m\ddot{x}(t)+c\dot{x}(t)+kx(t)=f(t)$

   where x(t) is the displacement; f(t) is the input force; m = 1 is the mass; c = 0.1 is the damping; k is the uncertain stiffness, for which we assign a prior PDF N(1,0.1^2); we observe the acceleration of the system $\ddot{x}(t)$ plus some noise, i.e.

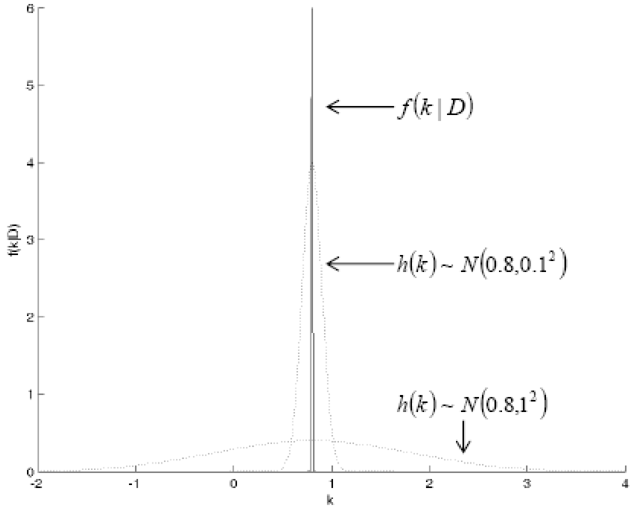
   $Y(t)=\ddot{x}(t)+e(t) \qquad e(t)\sim i.i.d. \quad N(0,1^2)$

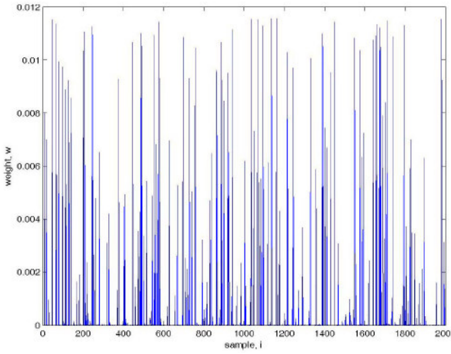
   Please use the same data as HW4.3, i.e. download HW4_3.mat from the website, that contains the data D, including f(t) and Y(t). Also download SDOF2.m.

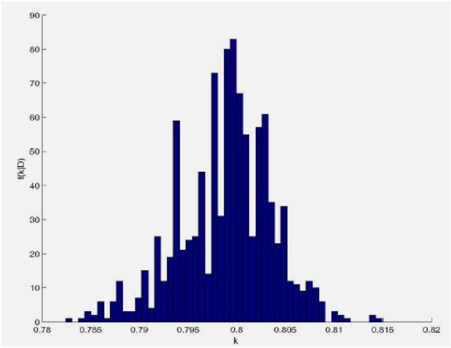
   (1) Estimate E[k|D] and Var[k|D] using IIS, where the importance sampling PDF is chosen to be N(0.8,0.1^2) and N(0.8,1^2). Also obtain the 95.4% confidence for E[k|D]. Which importance PDF is better? Why?

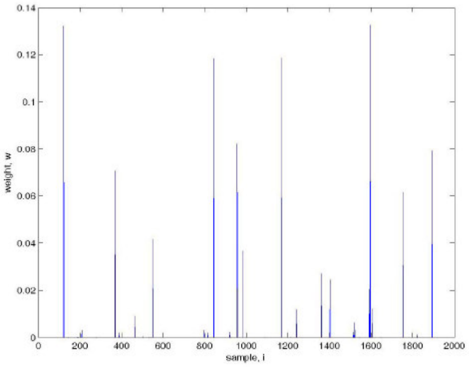
   (2) Obtain samples from f(k|D) using SIR (Sample-importance resampling), where the importance sampling PDF is chosen to be N(0.8,0.1^2) and N(0.8,1^2). Which importance PDF is better? Why?

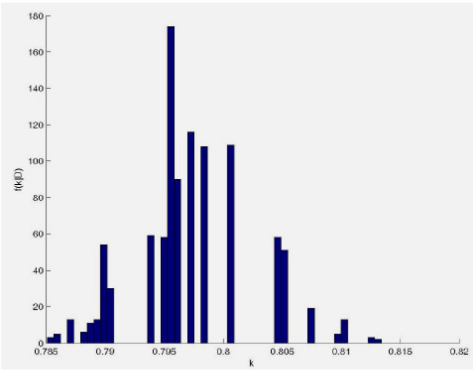
   (3) Note that now SDOF2.m also outputs the maximum absolute displacement umax of the SDOF due to the excitation f(t). Estimate E[umax|D] using IIS, where the importance sampling PDF is chosen to be N(0.8,0.1^2), and also obtain the 95.4% confidence for E[umax|D].



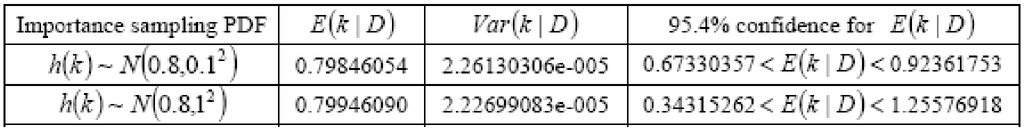






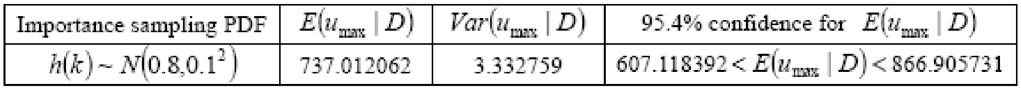


【Results】



Importance sampling PDF	$E(k \mid D)$	$Var(k \mid D)$	95.4% confidence for $E(k \mid D)$
$h(k) \sim N(0.8, 0.1^2)$	0.79846054	2.26130306e-005	$0.67330357 < E(k \mid D) < 0.92361753$
$h(k) \sim N(0.8, 1^2)$	0.79946090	2.22699083e-005	$0.34315262 < E(k \mid D) < 1.25576918$

【Result】



Importance sampling PDF	$E(u_{\max} \mid D)$	$Var(u_{\max} \mid D)$	95.4% confidence for $E(u_{\max} \mid D)$
$h(k) \sim N(0.8, 0.1^2)$	737.012062	3.332759	$607.118392 < E(u_{\max} \mid D) < 866.905731$

✦ **Markov Chain Monte Carlo**

1. Metropolis-Hasting algorithm
2. Gibbs sampler
3. Hybrid Monte Carlo

Metropolis-Hasting algorithm

Introduction:

Suppose the target PDF is the posterior PDF in a Bayesian analysis:

$$f\left(x \mid \hat{Y}\right) = \frac{f\left(\hat{Y} \mid x\right) f(x)}{f\left(\hat{Y}\right)} = a \cdot h(x)$$

The idea of MH is to create a Markov chain whose stationary distribution is the same as the target PDF.

Procedure:

1. Let $H\left(x^c \mid x^{(0)}\right)$ be the chosen proposal PDF.

2. Initialize $\hat{X}^{(0)}$ = any place.

3. Let the candidate $\hat{X}^C \sim H\left(x^c \mid \hat{X}^{(0)}\right)$ and compute $r = \dfrac{h\left(\hat{X}^C\right) H\left(\hat{X}^{(0)} \mid \hat{X}^C\right)}{h\left(\hat{X}^{(0)}\right) H\left(\hat{X}^C \mid \hat{X}^{(0)}\right)}$.

4. Let $\hat{X}^{(1)} = \begin{cases} \hat{X}^C & w.p \quad \min(1,r) \\ \hat{X}^{(0)} & w.p \ 1-\min(1,r) \end{cases}$

5. Cycle 3 → 4 to obtain Markov chain samples $\left\{\hat{X}^{(t)} : t = 0,...,T\right\}$. These samples will be asymptotically distributed as $f\left(x \mid \hat{Y}\right)$ if the Markov chain is ergodic.

First note that the MH algorithm creates a Markov chain that satisfies detailed balance: $F(z \mid x)h(x) = B(x \mid z)h(z) \quad \forall x,z$, $F(z \mid x)$ and $B(x \mid z)$ are the forward and backward probability transition kernels of the Markov chain. The forward and backward kernels in the MH algorithm are

$$F(z \mid x) = \left[1-\min(1,r)\right]\delta(z-x) + \min(1,r)H(z \mid x)$$

$$B(x \mid z) = \left[1-\min(1,r_B)\right]\delta(x-z) + \min(1,r_B)H(x \mid z)$$

where $r = \dfrac{h(z)H(x \mid z)}{h(x)H(z \mid x)} = \dfrac{f\left(z \mid \hat{Y}\right)H(x \mid z)}{f\left(x \mid \hat{Y}\right)H(z \mid x)}$, $r_B = \dfrac{h(x)H(z \mid x)}{h(z)H(x \mid z)} = \dfrac{f\left(x \mid \hat{Y}\right)H(z \mid x)}{f\left(z \mid \hat{Y}\right)H(x \mid z)}$.

$$\begin{aligned} LHS &= F(z \mid x)h(x) \\ &= \left[1-\min\left(1,\frac{h(z)H(x \mid z)}{h(x)H(z \mid x)}\right)\right]\delta(z-x)h(x) + \min\left(1,\frac{h(z)H(x \mid z)}{h(x)H(z \mid x)}\right)H(z \mid x)h(x) \\ &= \left[1-\min\left(1,\frac{h(x)H(z \mid x)}{h(z)H(x \mid z)}\right)\right]\delta(x-z)h(z) + \min\left(H(z \mid x)h(x), h(z)H(x \mid z)\right) \\ &= \left[1-\min\left(1,\frac{h(x)H(z \mid x)}{h(z)H(x \mid z)}\right)\right]\delta(x-z)h(z) + \min\left(\frac{h(x)H(z \mid x)}{h(z)H(x \mid z)},1\right)h(z)H(x \mid z) \\ &= B(x \mid z)h(z) = RHS \end{aligned}$$

Therefore, as long as the Markov chain is ergodic, the stationary distribution will be unique and equal to the target $f\left(x \mid \hat{Y}\right)$.

Analysis:

1. The MCMC estimator for $E\left[g(X)|\hat{Y}\right]$ is simply $\hat{g}^{MCMC}=\frac{1}{T}\sum_{t=1}^{T}g\left(\hat{X}^{(t)}\right)$. The MCMC estimator is asymptotically unbiased because

   $$E\left[\hat{g}_{MCMC}\right]=E\left[\frac{1}{T}\sum_{t=1}^{T}g\left(\hat{X}^{(t)}\right)|\hat{Y}\right]\underset{T\to\infty}{\to}E\left[g(X)|\hat{Y}\right]$$

   Note that if we ignore the MC samples in the burn-in period, the estimator is unbiased.

   $$Var\left[\hat{g}_{MCMC}\right]=Var\left[\frac{1}{T}\sum_{t=1}^{T}g\left(\hat{X}^{(t)}\right)|\hat{Y}\right]=\frac{1}{T^2}Var\left[\sum_{t=1}^{T}g\left(\hat{X}^{(t)}\right)|\hat{Y}\right]$$

   $$=\frac{1}{T^2}\left[\sum_{t=1}^{T}Var\left(g\left(\hat{X}^{(t)}\right)|\hat{Y}\right)+\sum_{s=1}^{T}\sum_{t=1}^{T}\operatorname{cov}\left(g\left(\hat{X}^{(s)}\right),g\left(\hat{X}^{(t)}\right)|\hat{Y}\right)\right]$$

   Note that after the MC reaches its stationary state, the time origin is forgotten, so

   $$\operatorname{cov}\left(g\left(\hat{X}^{(t)}\right),g\left(\hat{X}^{(t+k)}\right)|\hat{Y}\right)=R(k)$$

   Note that $R(k)=R(-k)$ and $R(0)=Var\left(g(X)|\hat{Y}\right)$

   $$Var\left[\hat{g}_{MCMC}\right]=\frac{1}{T^2}\left[T\cdot R(0)+\sum_{s=1}^{T}\sum_{t=1}^{T}R(s-t)\right]=\frac{1}{T^2}\left[T\cdot R(0)+2\sum_{s=1}^{T-1}\left[R(s)(T-s)\right]\right]$$

   $$=\frac{R(0)}{T}\left[1+2\sum_{s=1}^{T-1}\left(\frac{(T-s)\cdot R(s)}{T\cdot R(0)}\right)\right]=\frac{R(0)}{T}[1+\gamma]$$

   where $\gamma=2\sum_{s=1}^{T-1}\left(\frac{(T-s)\cdot R(s)}{T\cdot R(0)}\right)$ quantifies the degree of dependence of the MC samples. Note that $\frac{R(0)}{T}$ is the variance of the estimator if the MC samples are all independent. However, the MC samples are dependent so the actual variance is larger than $\frac{R(0)}{T}$ for a factor of $[1+\gamma]$. One can say that the equivalent number of independent samples is $\frac{T}{1+\gamma}$. Note that $R(k)$ can be estimated from the MC samples as the following:

   $$R(k)=\operatorname{cov}\left(g\left(X^{(t)}\right),g\left(X^{(t+k)}\right)\right)$$

   $$\approx\frac{1}{T-k}\sum_{t=1}^{T-k}\left(\left[g\left(\hat{X}^{(t)}\right)-\hat{g}_{MCMC}\right]\cdot\left[g\left(\hat{X}^{(t+k)}\right)-\hat{g}_{MCMC}\right]\right)\equiv\hat{R}(k)$$

2. Confidence interval

   Note that the Central Limit Theorem and the Law of Large Number still hold for weakly dependent samples, i.e.

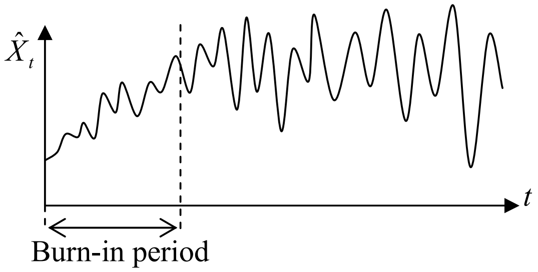
$$\hat{g}_{MCMC} = \frac{1}{T}\sum_{t=1}^{T} g\left(\hat{X}^{(t)}\right) \xrightarrow[T\to\infty]{} N\left(E\left[g(X)\,|\,\hat{Y}\right],\ \frac{Var\left(g(X)\,|\,\hat{Y}\right)}{T}[1+\gamma] \right)$$

So the following 95.4% confidence interval can be built:

$$\hat{g}^{MCMC} - 2\sqrt{\frac{\hat{R}(0)}{T}[1+\hat{\gamma}]} \le E\left[g(X)\,|\,\hat{Y}\right] \le \hat{g}^{MCMC} + 2\sqrt{\frac{\hat{R}(0)}{T}[1+\hat{\gamma}]}$$

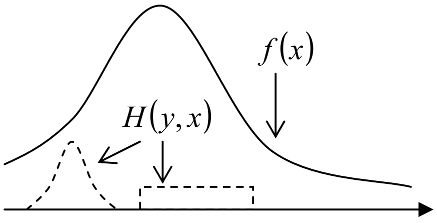
Remarks:

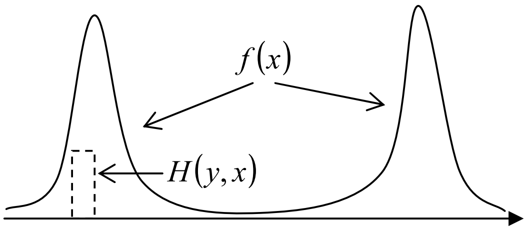
1. Recall that $f\left(x\,|\,\hat{Y}\right) = a\cdot h(x)$. We don't need to know how to evaluate the normalizing constant a since it is cancelled out as we compute *r*.
2. When the initial state of the Markov chain is chosen arbitrarily, the chain may need some time to reach its stationary state. We call this period the burn-in period. The MC samples in the burn-in period are not distributed as $f\left(x\,|\,\hat{Y}\right)$. We can choose to discard these samples. The usual way of identifying the burn-in period is to plot the sample value time history (or the time history of certain chosen statistics) and identify it visually.



Samples in the burn-in period can be discarded

3. How do we know the resulting MC is ergodic? In many cases, we don't know it for sure. But if we conduct several MCMCs to obtain independent MCs and found their behavior is similar, we can argue that the MC may be ergodic. There are cases where we can immediately see the MC is ergodic or not:

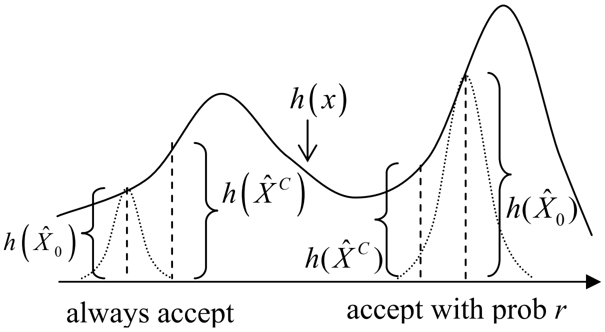




4. After reaching the stationary state, the MC samples become identically distributed but dependent. The dependency between two samples decays with the time

difference.

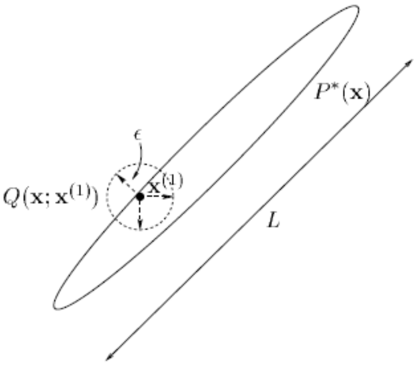
5. When $H(z \mid x) = H(x \mid z)$, the resulting algorithm is called the Metropolis algorithm. Note $r = \dfrac{h(\hat{X}^C)}{h(\hat{X}^{(0)})}$.



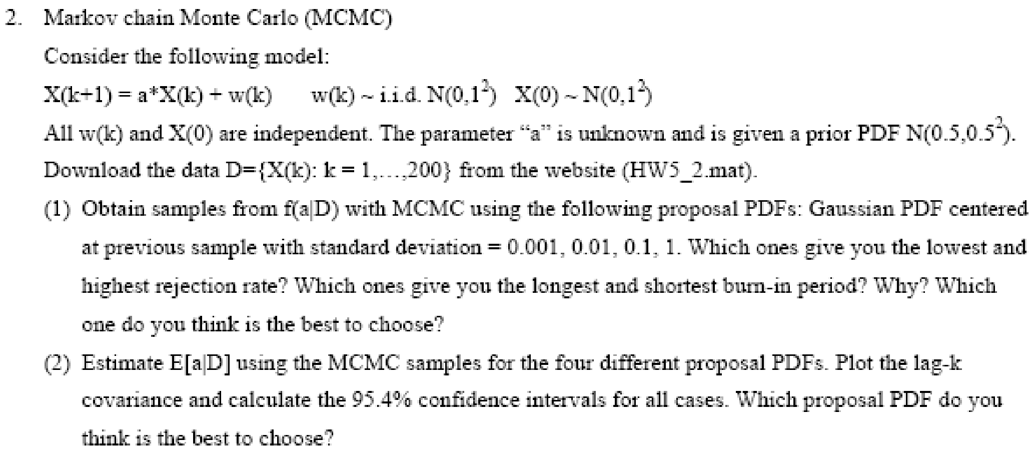
6. The choice of the proposal PDF $H(z \mid x)$ can significantly affect the efficiency of MCMC. From our experience, the type of $H(z \mid x)$ is not critical but the width of $H(z \mid x)$ matters. If $H(z \mid x)$ is very narrow, $r \approx 1$ and no rejection, but adjacent samples are highly correlated. If $H(z \mid x)$ is very wide, r is often small and lots of rejection and repeating samples, so the samples can be highly correlated. Therefore, $H(z \mid x)$ should not be too narrow or too wide. The rule of thumb is to take the width of $H(z \mid x)$ to be the same order of the width of $f(x \mid \hat{Y})$. Usually, the performance is the best when the rejection rate is around 50%.
7. In principle, the MH algorithm should work even when the X dimension is high. In practice, an efficient proposal PDF can be difficult to build because we usually lack of knowledge of the geometry of the target PDF $f(x \mid \hat{Y})$. There are also issues for the local-random-walk MH, e.g. MH with a Gaussian proposal PDF. Consider the following Gaussian PDF. Let $\sigma_{\max}^2$ and $\sigma_{\min}^2$ be the maximum and minimum eigenvalues of the covariance matrix. Suppose we choose a Gaussian proposal PDF $H(z \mid x)$ whose standard deviation in all directions is identical and equal to $\sigma_{\min}$. One can see that in order to have the MC samples travel throughout the significant region of the target PDF, we need at least $\sigma_{\max}/\sigma_{\min}$ MC

time steps.

This difficulty is due to high asperity ratio may occur regardless the dimension of *X*. However, from my experience, it occurs more frequently when *X* dimension is high. This is definitely not an issue when *X* is a scalar.



8. MH may not work for multi-modal PDFs since the resulting MC may be non-ergodic.



2. Markov chain Monte Carlo (MCMC)

   Consider the following model:

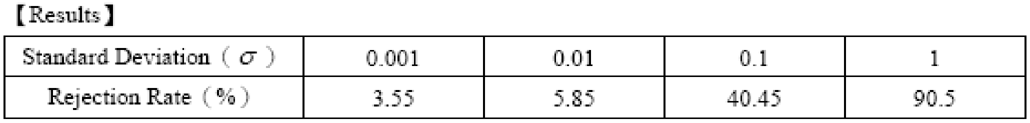
   X(k+1) = a*X(k) + w(k)   w(k) ~ i.i.d. N(0,1^2)   X(0) ~ N(0,1^2)

   All w(k) and X(0) are independent. The parameter "a" is unknown and is given a prior PDF N(0.5,0.5^2).

   Download the data D={X(k): k = 1,…,200} from the website (HW5_2.mat).

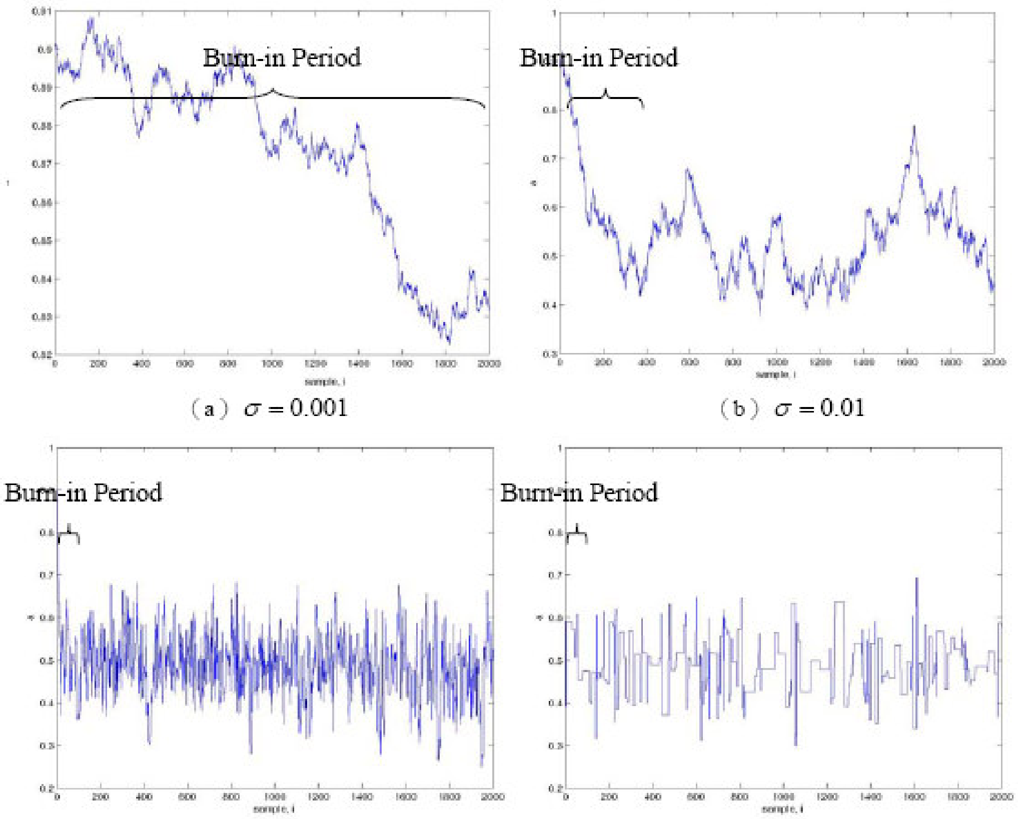
   (1) Obtain samples from f(a|D) with MCMC using the following proposal PDFs: Gaussian PDF centered at previous sample with standard deviation = 0.001, 0.01, 0.1, 1. Which ones give you the lowest and highest rejection rate? Which ones give you the longest and shortest burn-in period? Why? Which one do you think is the best to choose?

   (2) Estimate E[a|D] using the MCMC samples for the four different proposal PDFs. Plot the lag-k covariance and calculate the 95.4% confidence intervals for all cases. Which proposal PDF do you think is the best to choose?



【Results】

Standard Deviation（σ）	0.001	0.01	0.1	1
Rejection Rate（%）	3.55	5.85	40.45	90.5



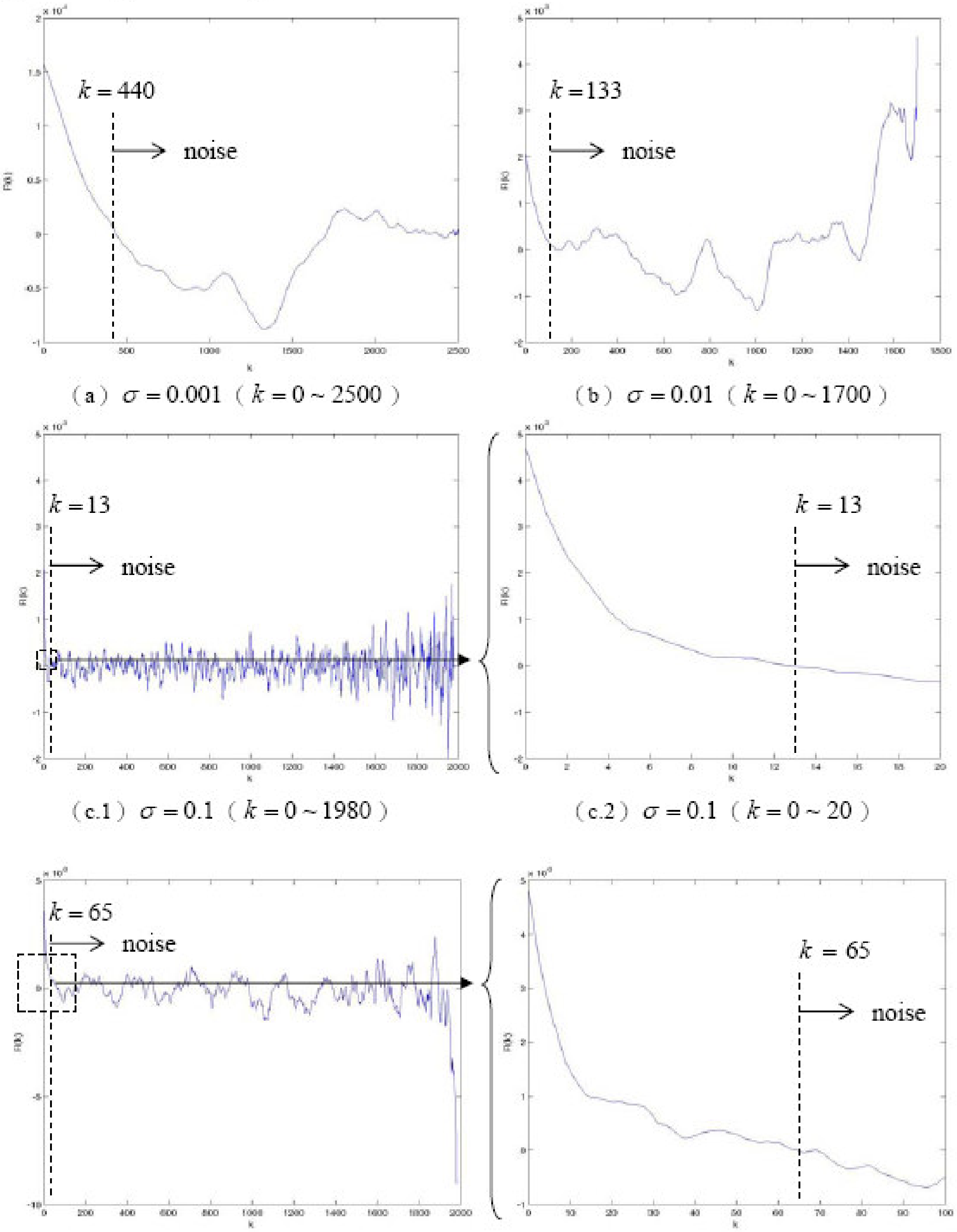
（a） $\sigma = 0.001$

（b） $\sigma = 0.01$

（c） $\sigma = 0.1$

（d） $\sigma = 1$

【Plot the lag-k covariance】



（a） $\sigma = 0.001$ （ $k = 0 \sim 2500$ ）

（b） $\sigma = 0.01$ （ $k = 0 \sim 1700$ ）

（c.1） $\sigma = 0.1$ （ $k = 0 \sim 1980$ ）

（c.2） $\sigma = 0.1$ （ $k = 0 \sim 20$ ）

（d.1） $\sigma = 1$ （ $k = 0 \sim 1980$ ）

（d.2） $\sigma = 1$ （ $k = 0 \sim 100$ ）

【Results】

Standard Deviation	$E(a \mid D)$	95.4% confidence for $E(a \mid D)$	Gamma, γ
$\sigma = 0.001$	0.46448930	$0.45499924 < E(a \mid D) < 0.47397936$	356.37133969
$\sigma = 0.01$	0.48923176	$0.46872213 < E(a \mid D) < 0.50974139$	82.83202645
$\sigma = 0.1$	0.49083634	$0.48336158 < E(a \mid D) < 0.49831110$	4.86750536
$\sigma = 1$	0.48869452	$0.47363000 < E(a \mid D) < 0.50375902$	22.11833874

Gibbs sampler

Introduction:

Is it possible to conduct MCMC without rejecting samples? It turns out to be possible. This can happen when the uncertain variable X can be divided into m groups $\{X_i : i=1,...,m\}$ and, moreover, when it is feasible to directly sample from $f\left(x_i \mid \{x \setminus x_i\}, \hat{Y}\right)$ $\forall i$, where $\{x \setminus x_i\}$ denotes $\{x_1,...,x_{i-1},x_{i+1},...,x_m\}$.

Procedure:

1. Initialize $\hat{X}_0 = \left[\hat{X}_1^{(0)}, \hat{X}_2^{(0)}, \cdots, \hat{X}_m^{(0)}\right]$ at any place.
2. Sample $\hat{X}^{(1)} = \begin{cases} \hat{X}_1^{(1)} \sim f\left(x_1 \mid x_2^{(0)},\cdots,x_m^{(0)},\hat{Y}\right) \\ \hat{X}_2^{(1)} \sim f\left(x_2 \mid x_1^{(1)},x_3^{(0)},\cdots,x_m^{(0)},\hat{Y}\right) \\ \hat{X}_3^{(1)} \sim f\left(x_3 \mid x_1^{(1)},x_2^{(1)},x_4^{(0)},\cdots,x_m^{(0)},\hat{Y}\right) \\ \vdots \\ \hat{X}_m^{(1)} \sim f\left(x_m \mid x_1^{(1)},\cdots,x_{m-1}^{(1)},\hat{Y}\right) \end{cases}$
3. Repeat step 2 to get $\left\{\hat{X}^{(t)} : t=0,\cdots,T\right\}$. These samples will be asymptotically distributed as $f\left(x \mid \hat{Y}\right)$ if the Markov chain is ergodic.

Note that the Gibbs sampler algorithm creates a Markov chain that satisfies detailed balance: $F(z \mid x) f\left(x \mid \hat{Y}\right) = B(x \mid z) f\left(z \mid \hat{Y}\right)$ $\quad \forall x,z$, $F(z \mid x)$ and $B(x \mid z)$ are the forward and backward probability transition kernels of the Markov chain. Using the following transition as an example:

$$\cdots \to \underset{x}{\underline{\underline{\begin{pmatrix} x_1^{(t+1)} \\ x_2^{(t+1)} \\ x_3^{(t)} \\ x_4^{(t)} \\ \vdots \\ x_m^{(t)} \end{pmatrix}}}} \to \underset{z}{\underline{\underline{\begin{pmatrix} x_1^{(t+1)} \\ x_2^{(t+1)} \\ x_3^{(t+1)} \\ x_4^{(t)} \\ \vdots \\ x_m^{(t)} \end{pmatrix}}}} \to \cdots$$

The forward and backward kernels in the Gibbs sampler algorithm are:

$$F(z\,|\,x)=\delta(z_1-x_1)\delta(z_2-x_2)\delta(z_4-x_4)\cdots\delta(z_m-x_m)\cdot f\left(z_3\,|\,x_1,x_2,x_4,\cdots,x_m,\hat{Y}\right)$$

$$B(x\,|\,z)=\delta(x_1-z_1)\delta(x_2-z_2)\delta(x_4-z_4)\cdots\delta(x_m-z_m)\cdot f\left(x_3\,|\,z_1,z_2,z_4,\cdots,z_m,\hat{Y}\right)$$

$$\begin{aligned}
&F(z\,|\,x)f\left(x\,|\,\hat{Y}\right)\\
&=\underbrace{\delta(z_1-x_1)\cdots\delta(z_m-x_m)}_{No\quad \delta(z_3-x_3)}\cdot f\left(z_3\,|\,\underbrace{x_1,\cdots,x_m}_{No\quad x_3},\hat{Y}\right)f\left(x_1,\cdots,x_m\,|\,\hat{Y}\right)\\
&=\underbrace{\delta(x_1-z_1)\cdots\delta(x_m-z_m)}_{No\quad \delta(x_3-z_3)}\cdot f\left(z_3\,|\,\underbrace{z_1,\cdots,z_m}_{No\quad z_3},\hat{Y}\right)\cdot f\left(z_1,z_2,x_3,z_4\cdots,z_m\,|\,\hat{Y}\right)\\
&=\underbrace{\delta(x_1-z_1)\cdots\delta(x_m-z_m)}_{No\quad \delta(x_3-z_3)}\cdot \frac{f\left(z_1,\cdots,z_m\,|\,\hat{Y}\right)}{f\left(\underbrace{z_1,\cdots,z_m}_{No\quad z_3}\,|\,\hat{Y}\right)}\cdot f\left(z_1,z_2,x_3,z_4\cdots,z_m\,|\,\hat{Y}\right)\\
&=\underbrace{\delta(x_1-z_1)\cdots\delta(x_m-z_m)}_{No\quad \delta(x_3-z_3)}\cdot \frac{f\left(z_1,z_2,x_3,z_4\cdots,z_m\,|\,\hat{Y}\right)}{f\left(\underbrace{z_1,\cdots,z_m}_{No\quad z_3}\,|\,\hat{Y}\right)}\cdot f\left(z_1,\cdots,z_m\,|\,\hat{Y}\right)\\
&=\underbrace{\delta(x_1-z_1)\cdots\delta(x_m-z_m)}_{No\quad \delta(x_3-z_3)}\cdot f\left(x_3\,|\,\underbrace{z_1,\cdots,z_m}_{No\quad z_3},\hat{Y}\right)\cdot f\left(z_1,\cdots,z_m\,|\,\hat{Y}\right)=B(x\,|\,z)f\left(z\,|\,\hat{Y}\right)
\end{aligned}$$

Remarks:

1. Unlike MH, Gibbs sampler needs no proposal PDF, so no rejection. But the tradeoff is that we need to know how to sample from $f(x_i\,|\,x\setminus x_i)$.
2. In the case that we don't know how to directly sample from $f(x_i\,|\,x\setminus x_i)$, we can sample it using sample-importance resampling, MCMC, rejection sampling, etc. A popular choice is to use adaptive rejection sampling to sample from $f(x_i\,|\,x\setminus x_i)$ [3].
3. The advantages and disadvantages of Gibbs sampler are similar to those for MH. The analysis is also the same.
4. Gibbs sampler will not work well when some uncertain parameters are highly correlated conditioning on the data. The consequence is that the MC samples move very slowly when sample these highly correlated variables. There are variants of Gibbs sampler that enjoy larger jumps in the MC samples even when such high correlation exists, e.g. Gibbs sampler with overrelaxation [4,5].

Hybrid Monte Carlo

Introduction:

Both MH and Gibbs sampler create MCs with local random walk behavior. However, this behavior is not desirable. With this behavior, it may take a long time to have the resulting MC travel throughout the significant region of $f\left(x \mid \hat{Y}\right)$, especially when X dimension is high. Hybrid Monte Carlo is a MCMC algorithm that does not use local random walk.

The basic idea is to add an auxiliary uncertain variable Z to the process, where the new target PDF is proportional to

$$f(x,z) \propto h(x) \cdot e^{-\frac{1}{2}\sum_{i=1}^{n} z_i^2}$$

where $f\left(x \mid \hat{Y}\right) = a \cdot h(x)$ and n is the dimension of X. If we are able to sample from $f(x,z)$, it is clear that the X part of the samples will be distributed as $f\left(x \mid \hat{Y}\right)$. How do we sample from $f(x,z)$? We employ MCMC.

The main trick for Hybrid Monte Carlo is to give $h(x)$ and z some physical meaning: $-\log\left[h(x)\right]$ is considered to be the potential energy of a ball with unit mass (x is the location of the ball; think of $-\log\left[h(x)\right]$ to be the profile of a valley) and z is the velocity of the ball. So the total energy of the ball is

$$-\log h(x) + \frac{1}{2}\sum_{i=1}^{n} z_i^2 = -\log f(x,z) \equiv H(x,z)$$

If there is no friction when the ball rolls on the valley, the total energy $H(x,z)$ is conservative, i.e. $f(x,z)$ is conservative, and the ball rolls according to the following equations:

$$\frac{dx_i}{dt} = z_i \qquad \frac{dz_i}{dt} = -\frac{\partial H(x,z)}{\partial x_i} = \frac{\partial \log h(x)}{\partial x_i} \qquad i = 1,\ldots,n$$

Procedure:

1. Initialize $\hat{X}^{(0)}$, $\hat{Z}^{(0)}$
2. Solve $x(t)$, $z(t)$ according to the following governing equation

$$\frac{dx_i}{dt} = z_i \qquad \frac{dz_i}{dt} = -\frac{\partial H(x,z)}{\partial x_i} = \frac{\partial \log h(x)}{\partial x_i} \qquad i = 1,\ldots,n$$

with initial condition $x(0)=\hat{X}^{(0)}$ and $z(0)=\hat{Z}^{(0)}$. Evolve the solution for randomized duration of $\hat{R}^{(0)}$. Let $\hat{X}^{(1)}=x\left(\hat{R}^{(0)}\right)$.

3. Resample $\hat{Z}^{(1)} \sim e^{-\frac{1}{2}\sum_{i=1}^{n} z_i^2}$
4. Cycle 2-3 to get $\left\{\hat{X}^{(t)}:t=0,...,T\right\}$. These samples will be asymptotically

   distributed as $f\left(x\,|\,\hat{Y}\right)$ if the Markov chain is ergodic.

Analysis: same as MH

Remarks:

1. The HMC algorithm indeed samples from $f\left(x,z\right)$. To see this, let's view the end product of Step 2 in the algorithm as a candidate, i.e. $\hat{X}^{C}=x\left(\hat{R}^{(0)}\right),\hat{Z}^{C}=z\left(\hat{R}^{(0)}\right)$ is the candidate of the MCMC algorithm. One can verify that

   $$r=\frac{f\left(\hat{X}^{C},\hat{Z}^{C}\right)}{f\left(\hat{X}^{(0)},\hat{Z}^{(0)}\right)}=1$$

   This is because the total energy $H\left(x,z\right)$, and hence $f(x,z)$, remains constant in the solution process of the Hamiltonian equations. Therefore, the candidate $\hat{X}^{C},\hat{Z}^{C}$ is always accepted as the next MC sample.
2. The duration $\hat{R}^{(0)}$ in Step2 is better to be randomized to avoid a periodic MC, although the chance of being so is very small.
3. Step 3 is necessary since without it, $f(x,z)$ will be always constant, hence the MC will not explore the entire phase space.
4. In practice, Step 2 cannot be solved analytically and must be solved approximately. Doing so, the resulting *r* ratio will be only approximately equal to 1. To handle this issue, we can simply add an accept/reject step, i.e. change Step 2 to the following:

   Solve $x\left(t\right)$, $z\left(t\right)$ **approximately** according to the following governing equation

$$\frac{dx_i}{dt} = z_i \qquad \frac{dz_i}{dt} = -\frac{\partial H(x,z)}{\partial x_i} = \frac{\partial \log h(x)}{\partial x_i} \qquad i = 1,...,n$$

with initial condition $x(0) = \hat{X}^{(0)}$ and $z(0) = \hat{Z}^{(0)}$. Evolve the solution for randomized duration of $\hat{R}^{(0)}$. Let $\hat{X}^{C} = x\left(\hat{R}^{(0)}\right)$ and $\hat{Z}^{C} = z\left(\hat{R}^{(0)}\right)$. Accept the candidate, i.e. take $\hat{X}^{(1)} = \hat{X}^{C}$, with probability *r*, where

$$r = \frac{f\left(\hat{X}^{C}, \hat{Z}^{C}\right)}{f\left(\hat{X}^{(0)}, \hat{Z}^{(0)}\right)}$$

If not accepted, repeat the previous sample, i.e. take $\hat{X}^{(1)} = \hat{X}^{(0)}$.

The approximate solution algorithm to the Hamiltonian equations can not be arbitrary. In fact, the chosen algorithm must be reversible in time, i.e. if starting from initial condition, we obtain the candidate solution; a reversible algorithm must have the property that starting from that candidate and solve the equation backward in time, we will obtain the same initial condition. This is so because the resulting HMC algorithm must satisfy the so called detailed balance (or reversibility) condition. The following "leapfrog" (finite difference) algorithm is reversible and is accurate to the 2nd order in the Taylor series expansion:

$$z_i\left(t + \frac{\Delta t}{2}\right) = z_i(t) + \frac{\Delta t}{2} \frac{\partial h(x)/\partial x_i\big|_{x=x(t)}}{h(x(t))}$$

$$x_i(t + \Delta t) = x_i(t) + \Delta t \cdot z_i\left(t + \frac{\Delta t}{2}\right)$$

$$z_i(t + \Delta t) = z_i\left(t + \frac{\Delta t}{2}\right) + \frac{\Delta t}{2} \frac{\partial h(x)/\partial x_i\big|_{x=x(t+\Delta t)}}{h(x(t+\Delta t))}$$

5. One can see that HMC does not do local random walk and it's possible for HMC to make large jumps. Of course the tradeoff is that HMC requires solving the Hamiltonian equations, which may requires much computation. According to the experience from other researchers, it seems that the cost is usually worthwhile, compared to MH and Gibbs sampler.

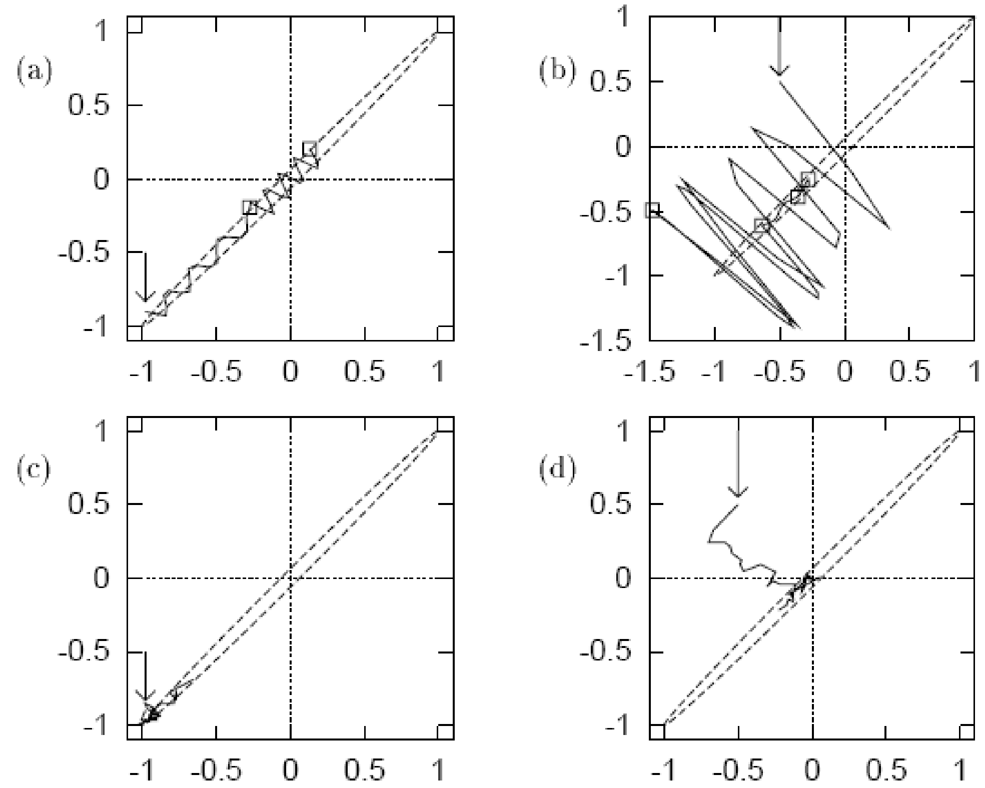


Figure 11. (a,b) Hybrid Monte Carlo used to generate samples from a bivariate Gaussian with correlation $\rho = 0.998$. (c,d) Random–walk Metropolis method for comparison. (a) Starting from the state indicated by the arrow, the continuous line represents two successive trajectories generated by the Hamiltonian dynamics. The squares show the endpoints of these two trajectories. Each trajectory consists of `Tau` = 19 'leapfrog' steps with `epsilon` = 0.055. After each trajectory, the momentum is randomized. Here, both trajectories are accepted; the errors in the Hamiltonian were +0.016 and −0.06 respectively. (b) The second figure shows how a sequence of four trajectories converges from an initial condition, indicated by the arrow, that is not close to the typical set of the target distribution. The trajectory parameters `Tau` and `epsilon` were randomized for each trajectory using uniform distributions with means 19 and 0.055 respectively. The first trajectory takes us to a new state, $(-1.5, -0.5)$, similar in energy to the first state. The second trajectory happens to end in a state nearer the bottom of the energy landscape. Here, since the potential energy E is smaller, the kinetic energy $K = \mathbf{p}^2/2$ is necessarily larger than it was at the start. When the momentum is randomized for the third trajectory, its magnitude becomes much smaller. After the fourth trajectory has been simulated, the state appears to have become typical of the target density. (c) A random–walk Metropolis method using a Gaussian proposal density with radius such that the acceptance rate was 58% in this simulation. The number of proposals was 38 so the total amount of computer time used was similar to that in (a). The distance moved is small because of random walk behaviour. (d) A random–walk Metropolis method given a similar amount of computer time to (b).

References:

[1] Gilks, W. R. (1992) Derivative-free adaptive rejection sampling for Gibbs sampling. Bayesian Statistics 4, (eds. Bernardo, J., Berger, J., Dawid, A. P., and Smith, A. F. M.) Oxford University Press.

[2] Gilks, W. R., Best, N. G. and Tan, K. K. C. (1995) Adaptive rejection Metropolis sampling. *Applied Statistics*, **44**, 455-472.

[3] Gilks, W.R., and Wild, P. (1992). Adaptive rejection sampling for Gibbs sampling. *Applied Statistics*, **41**, 337--348.

[4] Adler, S.L. (1981). Over-relaxation method for the Monte Carlo evaluation of the partition function for multiquadratic actions. *Physical Review D* - Particles and Fields, **23**(12), 2901-2904.

[5] Neal, R.M. (1995). Suppressing random walks in Markov chain Monte Carlo using ordered overrelaxation. *Technical Report 9508*, Dept. of Statistics, University of Toronto.

In general:

Mackay, J.D.C. "Introduction to Monte Carlo methods."