2

Basic Principles: Rejection, Weighting, and Others

2.1 Generating Simple Random Variables

To generate random variables that follow a general probability distribution function π, we need first to generate random variables *uniformly* distributed in [0,1]. These random variables are often called *random numbers* for simplicity. However, this "simple-sounding" task is not easily achievable on a computer. But even if it were possible, it might not be desirable to use authentic random numbers because of the need to debug computer programs. In debugging a program, we often have to repeat the same computation many times; this require us to reproduce the same sequence of random numbers repeatedly. What becomes an accepted alternative in the community of scientific computing is to generate *pseudo-random* numbers. More formally, we can define a *uniform pseudo-random number generator* as an algorithm which, starting from an initial value u_0 (i.e., the *seed*), produces a sequence $(u_i) = (D^i(u_0))$ of values in [0,1]. For all n, the values $(u_1, \dots, u_n)$ should reproduce the behavior of an i.i.d. sample $(V_1, \dots, V_n)$ of uniform random variables. A few very good pseudo-random number generators are available; we refer the reader to Marsaglia and Zaman (1993) and Knuth (1997) for further reference. Consequently, we *assume* from now on that uniform random variables can be satisfactorily produced on the computer. The following simple lemma enables us to produce nonuniform random variables. Its proof is left as an exercise for the reader.

J. S. Liu, *Monte Carlo Strategies in Scientific Computing*

Lemma 2.1.1 *Suppose $U \sim$ Uniform[0,1] and F is a one-dimensional cumulative distribution function (cdf). Then, $X = F^{-1}(U)$ has the distribution F. Here we define $F^{-1}(u) = \inf\{x;\ F(x) \geq u\}$.*

This lemma suggests to us an explicit way (i.e., the inversion method) of generating a one-dimensional random variable when its cdf is available. However, because many distributions (e.g., Gaussian) do not have a closed-form cdf, it is often difficult to directly apply the above inversion procedure. For distributions with nice mathematical properties, special methods are often available for drawing random samples from them. For example, a fast way of generating standard Gaussian random variables is to use the property that a standard *bivariate* Gaussian random vector (X, Y) (with zero mean and identity covariance matrix) can be generated by first uniformly choosing an angle in $\mathbb{R}^2$ (two-dimensional Euclidean space) and then generating the square distance from an Exponential distribution (Devroye 1986). A Beta random variable can be constructed as the ratio $X_1/(X_1 + X_2)$, where X_1 and X_2 are independent Gamma random variables. For mathematically less convenient distributions, von Neumann (1951) proposed a very general algorithm, the *rejection method*, which can — at least in principle — be applied to draw from any probability distribution with a density function given up to a normalizing constant, regardless of dimensions.

2.2 The Rejection Method

Suppose $l(\mathbf{x}) = c\pi(\mathbf{x})$ is computable, where π is a probability distribution function or density function and c is unknown. If we can find a sampling distribution $g(\mathbf{x})$ and "covering constant" M so that the envelope property [i.e., $Mg(\mathbf{x}) \geq l(\mathbf{x})$] is satisfied for all $\mathbf{x}$, then we can apply the following procedure.

Rejection sampling [von Neumann (1951)]:

(a) Draw a sample $\mathbf{x}$ from $g(\ )$ and compute the ratio

$$r = \frac{l(\mathbf{x})}{Mg(\mathbf{x})} \quad (\leq 1).$$

(b) Flip a coin with success probability r;

- if the head turns up, we accept and return the $\mathbf{x}$;
- otherwise, we reject the $\mathbf{x}$ and go back to (a).

The accepted sample follows the target distribution π.

To show that the foregoing procedure is correct, we let I be the indicator function so that $I = 1$ if the sample $\boldsymbol{X}$ drawn from $g(\ )$ is accepted, and

$I = 0$, otherwise. Then, we observe that

$$P(I=1) = \int P(I=1 \mid X = \mathbf{x}) g(\mathbf{x}) d\mathbf{x} = \int \frac{c\pi(\mathbf{x})}{Mg(\mathbf{x})} g(\mathbf{x}) d\mathbf{x} = \frac{c}{M}.$$

Hence,

$$p(\mathbf{x} \mid I=1) = \frac{c\pi(\mathbf{x})}{Mg(\mathbf{x})} g(\mathbf{x}) / P(I=1) = \pi(\mathbf{x}).$$

Because the expected number of "operations" for obtaining one accepted sample is M, The key to a successful application of the algorithm is to find a good trial distribution $g(\mathbf{x})$ which gives rise to a small M. It is usually very difficult to apply the simple rejection method for a high-dimensional Monte Carlo simulation problem.

Example: *Truncated Gaussian distribution.* Suppose we want to draw random samples from $\pi(x) \propto \phi(x) I_{\{x>c\}}$, where $\phi(x)$ is the standard normal density and I is the indicator function. A simple strategy can be applied when $c < 0$: We continue to generate random samples from a standard Gaussian distribution until a sample satisfying $X > c$ is obtained. In the worst case, the efficiency of this method is 50%.

For $c > 0$, especially when c is large, the above strategy is very inefficient. we can use the rejection method with an exponential distribution as the envelope function. Suppose the density of this exponential distribution has the form $\lambda_0 e^{-\lambda_0 x}$. We want to find the smallest constant b such that

$$\frac{\phi(x+c)}{1-\Phi(c)} \leq b\lambda_0 e^{-\lambda_0 x}, \quad \forall\, x \geq 0.$$

The optimal choice of b is

$$b = \frac{\exp\{(\lambda_0^2 - 2\lambda_0 c)/2\}}{\sqrt{2\pi}\lambda_0(1-\Phi(c))}.$$

The acceptance rate for using the posited exponential distribution as the envelope function is then $1/b$. To achieve the minimum rejection rate, we further find that the best choice for λ_0 is

$$\lambda_0 = (c + \sqrt{c^2+4})/2.$$

With this choice of λ_0 and b, we can implement the rejection method. The rejection rate for this scheme decreases as c increases, and this rate becomes very small for moderate to large c. For example, for c =0, 1, and 2, the rejection rates are 0.24, 0.12, and 0.07, respectively.

2.3 Variance Reduction Methods

Here we briefly describe a few techniques commonly used for variance reduction in Monte Carlo computations. More detailed descriptions can be found in standard Monte Carlo books (Hammersley and Handscomb 1964, Rubinstein 1981).

Stratified Sampling. It is a powerful and commonly used technique in population survey and is also very useful in Monte Carlo computations. Mathematically, this method can be viewed as a special importance sampling method with its trial density $g(x)$ constructed as a piecewise constant function. Suppose we are interested in estimating $\int_{\mathcal{X}} f(x)dx$. If possible, we want to break the region $\mathcal{X}$ into the union of k disjoint subregions, $D_1, \ldots, D_k$, so that within each subregion, the function $f(x)$ is relatively "homogeneous" (e.g., close to being a constant). Then, we can spend m_i random samples, $X^{(i,1)}, \ldots, X^{(i,m_i)}$, in the subregion D_i, and approximate the subregional integral $\int_{D_i} f(x)dx$ by

$$\hat{\mu}_i = \frac{1}{m_i}[f(X^{(i,1)}) + \cdots + f(X^{(i,m_i)})].$$

The overall integral μ can be approximated by

$$\hat{\mu} = \hat{\mu}_1 + \cdots + \hat{\mu}_k,$$

whose variance is easily calculated as

$$\text{var}(\hat{\mu}) = \frac{\sigma_1^2}{m_1} + \cdots + \frac{\sigma_m^2}{m_k},$$

where σ_i^2 is the variation of $f(x)$ in region D_i. In contrast, if we use all of the $m = m_1 + \cdots + m_k$ samples to do a plain uniform sampling in the region $\mathcal{X}$, the variance of the estimate would be σ^2/n, with σ^2 being the overall variation of $f(x)$ in $\mathcal{X}$.

Clearly, if we fail to have relatively homogeneous $f(x)$ in each region D_i (in other words, if σ_i^2 is not much different from σ^2), stratified sampling actually makes the computation less accurate than a plain Monte Carlo. The moral is this: There is no free lunch, and one needs to think carefully before adopting any advanced techniques.

Control Variates Method. In this method, one uses a control variate C, which is correlated with the sample X, to produce a better estimate. Suppose the estimation of $\mu = E(X)$ is of interest and $\mu_C = E(C)$ is known. Then, we can construct Monte Carlo samples of the form

$$X(b) = X + b(C - \mu_C),$$

which have the same mean as X, but a new variance

$$\text{var}\{X(b)\} = \text{var}(X) - 2b\text{cov}(X, C) + b^2\text{var}(C).$$

If the computation of $\text{cov}(X, C)$ and $\text{var}(C)$ is easy, then we can let $b = \text{cov}(X, C)/\text{var}(C)$, in which case

$$\text{var}\{X(b)\} = (1 - \rho_{XC}^2)\text{var}(X) < \text{var}(X).$$

Another situation is when we know only that $E(C)$ is equal to μ. Then, we can form $X(b) = bX + (1-b)C$. It is easy to show that if C is correlated with X, we can always choose a proper b so that $X(b)$ has a smaller variance than X. Extensions to more than one control variate are also useful in Monte Carlo computations, but are omitted in this book.

Antithetic Variates Method. This method is due to Hammersley and Morton (1956), where they describe a way of producing negatively correlated samples. Suppose U is the random number used in the production of a sample X that follows a distribution with cdf F [i.e., $X = F^{-1}(U)$ according to Lemma2.1.1]. Then, $X' = F^{-1}(1-U)$ also follows distribution F. More generally, if g is a monotonic function, then

$$\{g(u_1) - g(u_2)\}\{g(1-u_1) - g(1-u_2)\} \leq 0$$

for any $u_1, u_2 \in [0,1]$. For two independent uniform random variables U_1 and U_2 (in fact, it is only required that the two are i.i.d. a with symmetric density in [0,1]), we have

$$E[\{g(U_1) - g(U_2)\}\{g(1-U_1) - g(1-U_2)\}] = \text{cov}(X, X') \leq 0,$$

where $X = g(U)$ and $X' = g(1-U)$. Thus, $\text{var}[(X + X')/2] \leq \text{var}(X)/2$, implying that using the pair X and X' is better than using two independent Monte Carlo draws for estimating $E(X)$.

Rao-Blackwellization. This method reflects a basic principle (or rule of thumb) in Monte Carlo computation: One should carry out analytical computation as much as possible. The problem can be formulated as follows: Suppose we have drawn independent samples $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$ from the target distribution $\pi(\mathbf{x})$ and are interested in evaluating $I = E_\pi h(\mathbf{x})$. A straightforward estimator is

$$\hat{I} = \frac{1}{m}\left\{h(\mathbf{x}^{(1)}) + \cdots + h(\mathbf{x}^{(m)})\right\}.$$

Suppose, in addition, that $\mathbf{x}$ can be decomposed into two parts (x_1, x_2) and that the conditional expectation $E[h(\mathbf{x}) \mid x_2]$ can be carried out analytically. An alternative estimator of I is

$$\tilde{I} = \frac{1}{m}\left\{E[h(\mathbf{x}) \mid x_2^{(1)}] + \cdots + E[h(\mathbf{x}) \mid x_2^{(m)}]\right\}.$$

Clearly, both $\hat{I}$ and $\tilde{I}$ are unbiased[1] because of the simple fact that

$$E_\pi h(\mathbf{x}) = E_\pi[E\{h(\mathbf{x}) \mid x_2\}].$$

[1] An estimator $\hat{\mu}$ is called an unbiased estimator of μ if $E_\mu \hat{\mu} = \mu$. In words, this means that the average behavior of $\hat{\mu}$ is "on target."

If the computational effort for obtaining the two estimates are the same, then $\tilde{I}$ should be preferred because

$$\text{var}\{h(\mathbf{x})\} = \text{var}\{E[h(\mathbf{x}) \mid x_2]\} + E\{\text{var}[h(\mathbf{x}) \mid x_2]\},$$

which implies that

$$\text{var}(\hat{I}) = \frac{\text{var}\{h(\mathbf{x})\}}{m} \geq \frac{\text{var}\{E[h(\mathbf{x}) \mid x_2]\}}{m} = \text{var}(\tilde{I}).$$

In statistics, $\hat{I}$ is often called the "histogram estimator" or the empirical estimator and $\tilde{I}$ the "mixture estimator." Statisticians find that by conditioning an inferior estimator on the value of sufficient statistics, one can obtain the optimal estimator. This procedure is often referred to as *Rao-Blackwellization* (Bickel and Doksum 2000). Some other uses of Rao-Blackwellization in Monte Carlo estimations can be found in Casella and Robert (1996) More discussions on this issue can be found in Section 2.5.5 and Chapter 6.

2.4 Exact Methods for Chain-Structured Models

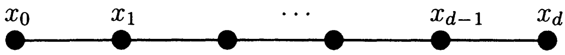
An important probability distribution used in many applications has the following form:

$$\pi(\mathbf{x}) \propto \exp\left\{-\sum_{i=1}^{d} h_i(x_{i-1}, x_i)\right\}, \tag{2.1}$$

where $\mathbf{x} = (x_0, x_1, \ldots, x_d)$. This type of model can be seen as having a "Markovian structure" because the conditional distribution $\pi(x_i \mid \mathbf{x}_{[-i]})$, where $\mathbf{x}_{[-i]} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_d)$, depends only on the two neighboring variables x_{i-1} and x_{i+1}; that is,

$$\pi(x_i \mid \mathbf{x}_{[-i]}) \propto \exp\left\{-h(x_{i-1}, x_i) - h(x_i, x_{i+1})\right\}.$$

The unobserved state variables in a state-space model (Section 1.6) can clearly be represented in this form, which can also be depicted by the following graph:



When $x_0, \ldots, x_d$ are discrete random variables taking values in a finite set $\mathcal{S} = \{s_1, \ldots, s_k\}$, this structure is often referred to as the *hidden Markov model* (HMM) and we can do many things with it. First, the "dynamic programming" (DP) method (Bellman 1957) can be used to find the global maximum of $\pi(\mathbf{x})$ and its maximizer $\hat{\mathbf{x}}$ with $O(dk^2)$ operations. Second, an algorithm of the same order as the DP exists for finding the marginal distribution of each x_i and drawing "exact" random samples from $\pi(\mathbf{x})$. Clearly, these exact algorithms are only practical when k, the number of distinctive values that x_i can take, is not too large.

2.4.1 Dynamic programming

Suppose each x_i in $\mathbf{x}$ only takes values in set $\mathcal{S} = \{s_1, \ldots, s_k\}$. Maximizing the distribution $\pi(\mathbf{x})$ in (2.1) is equivalent to minimizing its exponent

$$H(\mathbf{x}) = h_1(x_0, x_1) + \cdots + h_d(x_{d-1}, x_d).$$

A recursive procedure can be carried out:

- Define
$$m_1(x) = \min_{s_i \in \mathcal{S}} h_1(s_i, x), \text{ for } x = s_1, \ldots, s_k.$$
- Recursively compute the function
$$m_t(x) = \min_{s_i \in \mathcal{S}} \{m_{t-1}(s_i) + h_t(s_i, x)\}, \text{ for } x = s_1, \ldots, s_k.$$
- The optimal value $H(\mathbf{x})$ is attained by $\min_{s \in \mathcal{S}} m_d(s)$.

It is not difficult to see that the minimum of $m_1(x)$ is equal to the minimum of $h_1(x_0, x_1)$. By induction, one can easily argue that

$$\min_{x \in \mathcal{S}} m_t(x) = \min_{x_0, \ldots, x_t \in \mathcal{S}} [h_1(x_0, x_1) + \cdots + h_t(x_{t-1}, x_t)].$$

Thus, the above procedure indeed minimizes the target function $H(\mathbf{x})$.

To find out which $\mathbf{x}$ gives rise to the global minimum of $H(\mathbf{x})$, we can trace backward as follows:

- Let $\hat{x}_d$ be the minimizer of $m_d(x)$; that is,
$$\hat{x}_d = \arg \min_{s_i \in \mathcal{S}} m_d(s_i).$$
Break ties arbitrarily.
- For $t = d-1, d-2, \ldots, 1$, we let
$$\hat{x}_t = \arg \min_{s_i \in \mathcal{S}} \{m_t(s_i) + h_{t+1}(s_i, \hat{x}_{t+1})\}.$$
Break ties arbitrarily.

Configuration $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_d)$ obtained by this method is the minimizer of $H(\mathbf{x})$.

2.4.2 Exact simulation

The first step for simulating from $\pi(\mathbf{x})$ in (2.1) is to draw x_d from its marginal distribution. This requires us to marginalize $x_1, \ldots, x_{d-1}$ in the joint distribution $\pi(\mathbf{x})$. After we have drawn x_d, we can work our way backward recursively; that is, sampling x_{d-1} from $\pi(x_{d-1}|x_d)$; x_{d-2} from $\pi(x_{d-2}|x_{d-1})$; and so on. The principle behind the marginalization step is based on the observation that the overall summation can be decomposed into recursive steps; that is,

$$Z \equiv \sum_{\mathbf{x}} \exp\{-H(\mathbf{x})\} = \sum_{x_d} \left[\cdots \left[\sum_{x_1} \left\{ \sum_{x_0} e^{-h_1(x_0,x_1)} \right\} e^{-h_2(x_1,x_2)} \right] \cdots \right].$$

More precisely, the following recursions similar to those in DP can be carried out by a computer:

- Define $V_1(x) = \sum_{x_0 \in \mathcal{S}} e^{-h_1(x_0,x)}$.
- Compute recursively for $t = 2, \ldots, d$:

$$V_t(x) = \sum_{y \in \mathcal{S}} V_{t-1}(y) e^{-h_t(y,x)}. \tag{2.2}$$

Then, the partition function is $Z = \sum_{x \in \mathcal{S}} V_d(x)$ and the marginal distribution of x_d is $\pi(x_d) = V_d(x_d)/Z$. To simulate $\mathbf{x}$ from π, we can do the following:

- Draw x_d from $\mathcal{S}$ with probability $V_d(x_d)/Z$;
- For $t = d-1, d-2, \ldots, 1$, we draw x_t from distribution

$$p_t(x) = \frac{V_t(x) e^{-h_{t+1}(x,x_{t+1})}}{\sum_{y \in \mathcal{S}} V_t(y) e^{-h_{t+1}(y,x_{t+1})}}.$$

The random sample $\mathbf{x} = (x_1, \ldots, x_d)$ obtained in this way follows the distribution $\pi(\mathbf{x})$.

As an example, we can use the forward-recursion formula (2.2) to compute the partition function for a one-dimensional Ising model

$$\pi(\mathbf{x}) = Z^{-1} \exp\{\beta(x_0 x_1 + \cdots + x_{d-1} x_d)\},$$

where x_i takes value in $\mathcal{S} = \{-1, 1\}$. First, we have

$$V_1(x) = e^{\beta x} + e^{-\beta x} \equiv e^{\beta} + e^{-\beta},$$

which is a constant. Applying the recursion, we easily obtain that

$$V_t(x) = (e^{-\beta} + e^{\beta})^t$$

and $Z = 2(e^{-\beta}+e^{\beta})^d$. Details for an exact simulation from this distribution are left to the reader.

An important feature of the target distribution $\pi(\mathbf{x})$ treated in this section is that it can be written as

$$\pi(\mathbf{x}) \propto \exp\left\{-\sum_{C\in\mathcal{C}} h(\mathbf{x}_C)\right\},$$

where $\mathcal{C}$ is the set of some subsets of $\{1,\ldots,d\}$ and $\mathbf{x}_C = (x_i,\ i \in C)$. In Lauritzen and Spiegelhalter (1988), each subset C in $\mathcal{C}$ is called a "clique." Any two subsets C_1 and C_2 are said to "connected" if they share at least one common component. Any probability distribution that possesses this dependency structure is termed a *graphical model.* Our model in this section can be seen as a special graphical model in which the set of cliques is $\mathcal{C} = \{C_1,\ldots,C_d\}$, where $C_i = \{i-1,i\}$. When $\mathcal{C}$ forms a "tree" and each clique does not have too many vertices (components), one can derive efficient algorithms for optimization and exact simulation similar to the algorithms described in this section (Lauritzen and Spiegelhalter 1988). There does not seem to be a common name for this sampling method. Some people call it the *peeling algorithm* because this method was first developed for a genetic linkage problem (Cannings, Thompson and Skolnick 1978). Some others refer it as the *forward-summation-backward-sampling* method, a rather awkward name. We prefer to use the name *propagation method* for the reason that both the forward and the backward steps can be thought of as propagating information along the chain graph.

2.5 Importance Sampling and Weighted Sample

2.5.1 An example

Suppose we wish to evaluate the quantity

$$\theta = \int_{\mathcal{X}} h(x)\pi(x)dx = E_\pi[h(X)],$$

where the support of the random variable X is denoted as $\mathcal{X}$ and $h(x) \geq 0$. In standard numerical methods, we discretize the domain $\mathcal{X}$ by regular grids, evaluate $h(x)\pi(x)$ on each of the grid points, and then use the Riemann sum as an approximation. Consider the target function given by Figure 2.1(a):

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

where $(x,y) \in [-1,1]\times[-1,1]$. More than two-thirds of computing time are wasted on evaluating those grid points on which the function is virtually

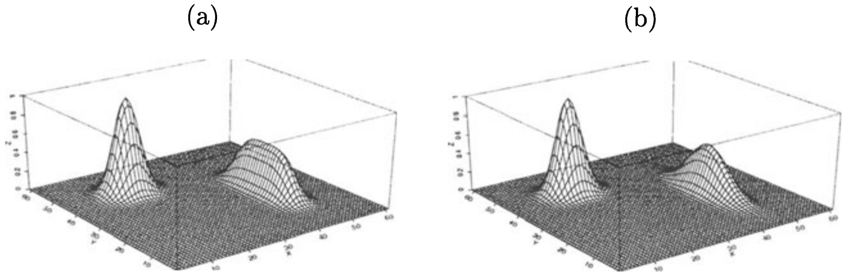


FIGURE 2.1. (a) The target function whose integral is of interest. (b) A possible trial distribution $g(x, y)$ for applying importance sampling.

zero. It is easy to imagine that the situation deteriorates very rapidly as the dimension of space $\mathcal{X}$ increases.

By taking m random samples, $(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})$, uniformly in $[-1, 1] \times [-1, 1]$, we implemented a vanilla Monte Carlo algorithm to estimate the integral $\mu = \int\int f(x, y)dxdy$. Because the density for the sampling distribution is a constant, 1/4, in the region, the estimate of the integral was produced as

$$\hat{\mu} = \frac{4}{m}\left\{f^{(1)} + \cdots + f^{(m)}\right\},$$

where $f^{(i)} = f(x^{(i)}, y^{(i)})$. With m=2,500, we obtained $\hat{\mu}$= 0.1307, with a standard deviation 0.009, which was estimated by

$$\text{std}(\hat{\mu}) = \sqrt{\frac{1}{n(n-1)}\sum_{i=1}^{m}(f_i - \hat{\mu})^2}.$$

Clearly, this vanilla Monte Carlo algorithm suffers a similar problem as its deterministic counterpart: It wastes a lot of effort in evaluating random samples located in regions where the function value is almost zero. Although the theoretical convergence rate is $m^{-1/2}$ for essentially *all* Monte Carlo methods, it is the constant in front of this rate that makes a huge difference in a real problem.

2.5.2 The basic idea

The *importance sampling* idea (Marshall 1956) suggests that one should focus on the region(s) of "importance" so as to save computational resources. Although it may not seem so important in the toy example shown earlier, the idea of biasing toward "importance" regions of the sample space becomes essential for Monte Carlo computation with high-dimensional models such as those in statistical physics, molecular simulation, and Bayesian statistics. In high-dimensional problems, the region in which the target function is meaningfully nonzero compared with the whole space $\mathcal{X}$ is just like a needle compared with a haystack. Vanilla Monte Carlo schemes (e.g., sampling uniformly from a regular region) are bound to fail in these problems.

Suppose one is interested in evaluating

$$\mu = E_\pi\{h(\mathbf{x})\} = \int h(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}.$$

The following procedure is a simple form of the *importance sampling algorithm*:

(a) Draw $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$ from a *trial distribution* $g(\cdot)$.

(b) Calculate the *importance weight*

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

(c) Approximate μ by

$$\hat{\mu} = \frac{w^{(1)}h(\mathbf{x}^{(1)}) + \cdots + w^{(m)}h(\mathbf{x}^{(m)})}{w_1 + \cdots + w_m}. \tag{2.3}$$

Thus, in order to make the estimation error small, one wants to choose $g(\mathbf{x})$ as "close" in shape to $\pi(\mathbf{x})h(\mathbf{x})$ as possible. A major advantage of using (2.3) instead of the unbiased estimate,

$$\tilde{\mu} = \frac{1}{m}\left\{w^{(1)}h(\mathbf{x}^{(1)}) + \cdots + w^{(m)}h(\mathbf{x}^{(m)})\right\}, \tag{2.4}$$

is that in using the former, we need *only* to know the ratio $\pi(\mathbf{x})/g(\mathbf{x})$ up to a multiplicative constant; whereas in the latter, the ratio needs to be known exactly. Additionally, although inducing a small bias, (2.3) often has a smaller mean squared error than the unbiased one $\tilde{\mu}$.

Another scenario for resorting to importance sampling is when we want to generate i.i.d. random samples from π but doing so directly is infeasible. In this case, we may generate random samples from a different, but similar, trivial distribution $g(\ )$, and then correct the bias by using the importance sampling procedure. Similar to the rejection method, a successful application of importance sampling in this case requires that the sampling distribution g is reasonably close to π; in particular, that g has a longer tail than π. Note that finding a good trial distribution g can be a major — and sometimes impossible — undertaking in high-dimensional problems.

Alternatively, we can opt for *correlated* samples produced by running a Markov chain whose stationary distribution is π. This methodology is referred to as *Markov chain Monte Carlo* (MCMC) throughout the book. A very general recipe for designing a proper Markov chain was first proposed by Metropolis et al. (1953) and has been subject to active research in the past few decades. We will discuss this class of methods in the latter part of this book (Chapters 5–11).

Let us illustrate the *importance sampling* method with the toy example shown in Figure 2.1. After a visual examination of function $f(x,y)$, we decided to use a distribution $g(x,y)$, which is of the form

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

with $(x,y) \in [-1,1] \times [-1,1]$. This is a truncated mixture of Gaussian distributions:

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

We can sample from this mixture distribution by a two-step procedure: (a) a biased coin (with probability 0.464 of showing heads) is first tossed; (b) if the head turns up, we draw a random vector from the first Gaussian distribution, otherwise, we draw from the second Gaussian distribution. The integral can then be estimated by averaging the ratios $w(x,y) = f(x,y)/g(x,y)$, with $w = 0$ when (x,y) falls out of the region $\mathcal{X}$. With m=2500, our estimate of μ is 0.1259, with a standard error 0.0005.

2.5.3 The "rule of thumb" for importance sampling

Importance sampling suggests estimating $\mu = E_\pi\{h(\mathbf{x})\}$ by first generating independent samples $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$ from an easy-to-sample trial distribution, $g(\ )$, and then correcting the bias by incorporating the importance weight $w^{(j)} \propto \pi(\mathbf{x}^{(j)})/g(\mathbf{x}^{(j)})$ in estimation using either (2.3) or 2.4. By properly choosing $g(\cdot)$, one can reduce the variance of the estimate substantially. A good candidate for $g(\cdot)$ is one that is close to the shape of $h(\mathbf{x})\pi(\mathbf{x})$. Therefore, the importance sampling method can be super-efficient; that is, the resulting variance of $\hat{\mu}$ can be smaller than that obtained using independent samples from π. The method is generalized to the case of, say, evaluating $E_\pi\{h(\mathbf{x})\}$ when sampling from $\pi(\cdot)$ directly is difficult but generating from $g(\cdot)$ and computing the importance ratio $w(\mathbf{x}) = \pi(\mathbf{x})/g(\mathbf{x})$ (up to a multiplicative constant) are easy. The efficiency of such a method is then difficult to measure. A useful "rule of thumb" is to use the *effective sample size* (ESS) to measure how different the trial distribution is from the target distribution. Suppose m independent samples are generated from $g(\mathbf{x})$; then, the ESS of this method is defined as

$$\text{ESS}(m) = \frac{m}{1 + \text{var}_g[w(\mathbf{x})]}.$$

Since the target distribution π is known only up to a normalizing constant in many problems, the variance of the *normalized* weight needs to be estimated by the *coefficient of variation* of the unnormalized weight:

$$\mathrm{cv}^2(w) = \frac{\sum_{j=1}^m (w^{(j)} - \bar{w})^2}{(m-1)\bar{w}^2}, \tag{2.5}$$

where $\bar{w}$ is the sample average of the $w^{(j)}$. The ESS measure of efficiency can be partially justified by the delta method as follows (Kong et al. 1994, Liu 1996a).

Note that $E_p\{w(\mathbf{x})\} = 1$; hence, both (2.3) and (2.4) are proper estimates of μ. In particular, the two estimates are related to each other in the following form:

$$\hat{\mu} = \frac{\frac{1}{m}\sum_{j=1}^m h(\mathbf{x}^{(j)})w(\mathbf{x}^{(j)})}{\frac{1}{m}\sum_{j=1}^m w(\mathbf{x}^{(j)})} \equiv \frac{\tilde{\mu}}{\bar{W}}. \tag{2.6}$$

Let $Z = h(\mathbf{x})w(\mathbf{x})$, $W = w(\mathbf{x})$, and let $\bar{Z}$ and $\bar{W}$ be the corresponding sample averages. As we have mentioned in Section 2.5.2, there are two advantages for choosing $\hat{\mu}$ over $\bar{Z}$ for estimation: The importance sampling ratios need only to be evaluated up to an unknown constant; and $\hat{\mu}$ may have smaller mean squared error than $\tilde{\mu} \equiv \bar{Z}$. By the delta method, we see that

$$\begin{aligned} E_g(\hat{\mu}) &\approx E_g\{\bar{Z}[1 - (\bar{W} - 1) + (\bar{W} - 1)^2 + \cdots]\} \\ &\approx \mu - \frac{\mathrm{cov}_g(W, Z)}{m} + \frac{\mu \mathrm{var}_g W}{m} \end{aligned}$$

The variance of $\hat{\mu}$ can be explored by using the standard delta method for ratio statistics:

$$\mathrm{var}_g(\hat{\mu}) \approx \frac{1}{m}[\mu^2 \mathrm{var}_g(W) + \mathrm{var}_g(Z) - 2\mu \mathrm{cov}_g(W, Z)]. \tag{2.7}$$

In contrast, $E_g(\tilde{\mu}) = \mu$ and $\mathrm{var}_g(\tilde{\mu}) = \mathrm{var}_g(Z)/m$. Hence, the mean squared error (MSE) of $\tilde{\mu}$ is

$$\mathrm{MSE}(\tilde{\mu}) = E_g(\tilde{\mu} - \mu)^2 = \mathrm{var}_g(Z)/m,$$

and that for $\hat{\mu}$ is

$$\begin{aligned} \mathrm{MSE}(\hat{\mu}) &= [E_g(\hat{\mu}) - \mu]^2 + \mathrm{var}_g(\hat{\mu}) \\ &= \frac{1}{m}\mathrm{MSE}(\tilde{\mu}) + \frac{1}{m}[\mu^2 \mathrm{var}_g(W) - 2\mu \mathrm{cov}_g(W, Z)] + O(m^{-2}) \end{aligned}$$

Without loss of generality, we let $\mu > 0$. Then, $\mathrm{MSE}(\hat{\mu})$ is smaller in comparison with $\mathrm{MSE}(\tilde{\mu})$ when $\mu^2 - 2\mu \mathrm{cov}_g(W, Z) < 0$ (i.e., when W and Z are strongly correlated).

Denoting $H = h(\mathbf{x})$, we observe that $Z = WH$, $\mu = E_g(WH)$, and

$$\mathrm{cov}_g(W, Z) = E_\pi(HW) - \mu = \mathrm{cov}_\pi(W, H) + \mu E_\pi(W) - \mu.$$

Similarly

$$\begin{aligned} \mathrm{var}_g(Z) &= E_\pi(WH^2) - \mu^2 \\ &\approx E_\pi(W)E_\pi^2(H) + \mathrm{var}_\pi(H)E_\pi(W) + 2\mu\mathrm{cov}_\pi(W, H) - \mu^2, \end{aligned}$$

where the approximation is made based on the delta method involving the first two moments of W and H. It is easy to show that the remainder term in the above approximation is

$$E_\pi[\{W - E_\pi(W)\}(H - \mu)^2], \tag{2.8}$$

which is not necessarily small. By reformulating (2.7), we find that

$$\mathrm{var}_g(\tilde{\mu}) \approx \mathrm{var}_\pi(H)\{1 + \mathrm{var}_p(W)\}/m.$$

Roughly speaking, if μ were estimated by $\hat{\hat{\mu}} = \sum_{j=1}^m h(\mathbf{y}^{(j)})/m$ where the $\mathbf{y}^{(j)}$ are i.i.d. draws from π, then the efficiency of $\hat{\mu}$ relative to $\hat{\hat{\mu}}$ is

$$\frac{\mathrm{var}_\pi\{h(\mathbf{y})\}}{\mathrm{var}_g\{h(\mathbf{x})w(\mathbf{x})\}} \approx \frac{1}{1 + \mathrm{var}_g\{w(\mathbf{x})\}}.$$

This can be interpreted as that the m weighted samples is worth of $m/\{1 + \mathrm{var}_g[w(\mathbf{x})]\}$ i.i.d. samples drawn from the target distribution. Obviously, the "rule of thumb" approximation can be substantially off if the remainder term (2.8) is large. The advantage of the "rule" is that it does not involve $h(\mathbf{x})$, which makes it particularly useful as a measure of the relative efficiency of the method when many different h's are of potential interest.

2.5.4 Concept of the weighted sample

The concept of a *properly weighted sample* is a useful generalization to the foregoing importance sampling procedure. Suppose we are interested in Monte Carlo estimation of $\mu = E_\pi h(\mathbf{x})$ for some arbitrary function h. The importance sampling principle suggests that the random sample $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$ used to estimate μ need not be drawn from π — they can be drawn from almost any distribution provided that a proper set of weights are associated with the sample and the weights are not too skewed.

Definition 2.5.1 *A set of weighted random samples* $\{(\mathbf{x}^{(j)}, w^{(j)})\}_{j=1}^m$ *is called proper with respect to* π *if for any square integrable function* $h(\cdot)$,

$$E[h(\mathbf{x}^{(j)})w^{(j)}] = cE_\pi h(\mathbf{x}), \quad \textit{for} \;\; j = 1, \ldots, m,$$

where c *is a normalizing constant common to all the* m *samples.*

With this set of weighted samples, we can estimate μ as

$$\hat{\mu} = \frac{1}{W}\sum_{j=1}^{m} w^{(j)} h(\mathbf{x}^{(j)}), \tag{2.9}$$

where $W = \sum_{j=1}^{m} w^{(j)}$. Mathematically, this says that the joint distribution $g(w, \mathbf{x})$ for both the weight and the sample satisfies the relationship: For any square integrable $h(\cdot)$, $E_g\{h(\mathbf{x})w\}/E_g(w) = E_\pi\{h(\mathbf{x})\}$. This equality also implies that

$$\frac{E_g(w \mid \mathbf{x})}{E_g(w)} g(\mathbf{x}) = \pi(\mathbf{x}), \tag{2.10}$$

where $g(\mathbf{x})$ is the marginal distribution of $\mathbf{x}$ under $g(\mathbf{x}, w)$. Thus, a *necessary and sufficient* condition for $\mathbf{x}$ to be properly weighted by w with respect to π is (2.10).

The whole point of this generalization is to emphasize that there are many possible choices of the weighting function w for any given $\mathbf{x}$. In the context of importance sampling, the importance weight w is a deterministic function of the corresponding sample $\mathbf{x}$ [i.e., $w = \pi(\mathbf{x})/g(\mathbf{x})$]. Thus, in this case the joint distribution of $(w, \mathbf{x})$ is a degenerate one. It is also possible that conditional on $\mathbf{x}$, the weight variable w has a proper distribution and this flexibility can be useful for combining importance sampling with MCMC algorithms (more details in later chapters).

2.5.5 Marginalization in importance sampling

As we explained in Section 2.3, the method of Rao-Blackwellization is useful for improving estimation in a vanilla Monte Carlo scheme. Here, we show that the same technique takes the form of *marginalization* in importance sampling and is useful for reducing the variance of the importance weight.

Theorem 2.5.1 *Let $f(\mathbf{z}_1, \mathbf{z}_2)$ and $g(\mathbf{z}_1, \mathbf{z}_2)$ be two probability densities, where the support of f is a subset of the support of g. Then,*

$$var_g\{\frac{f(Z_1, Z_2)}{g(Z_1, Z_2)}\} \geq var_g\{\frac{f_1(Z_1)}{g_1(Z_1)}\},$$

where $f_1(\mathbf{z}_1) = \int f(\mathbf{z}_1, \mathbf{z}_2) d\mathbf{z}_2$ and $g_1(\mathbf{z}_1) = \int g(\mathbf{z}_1, \mathbf{z}_2) d\mathbf{z}_2$ are marginal densities. The variances are taken with respect to g.

Proof: It is easy to see that

$$\begin{aligned} \frac{f_1(\mathbf{z}_1)}{g_1(\mathbf{z}_1)} &= \int \frac{f(\mathbf{z}_1, \mathbf{z}_2)}{g_1(\mathbf{z}_1) g_{2|1}(\mathbf{z}_2|\mathbf{z}_1)} g_{2|1}(\mathbf{z}_2|\mathbf{z}_1) d\mathbf{z}_2 \\ &= E_g\left\{ \frac{f(Z_1, Z_2)}{g(Z_1, Z_2)} \,\middle|\, Z_1 = \mathbf{z}_1 \right\}. \end{aligned}$$

Hence,

$$\mathrm{var}_g\left\{\frac{f(Z_1,Z_2)}{g(Z_1,Z_2)}\right\} \geq \mathrm{var}_g\left\{E_g\left[\frac{f(Z_1,Z_2)}{g(Z_1,Z_2)} \;\middle|\; Z_1\right]\right\} = \mathrm{var}_g\left\{\frac{f_1(Z_1)}{g_1(Z_1)}\right\}.$$

We can also obtain an explicit expression of the variance reduction:

$$\mathrm{var}_g\left\{\frac{f(Z_1,Z_2)}{g(Z_1,Z_2)}\right\} - \mathrm{var}_g\left\{\frac{f_1(Z_1)}{g_1(Z_1)}\right\} = E_g\left\{\mathrm{var}_g\left[\frac{f(Z_1,Z_2)}{g(Z_1,Z_2)} \;\middle|\; Z_1\right]\right\},$$

which, in the Analysis of Variance (ANOVA) terminology, is the average "within-group" variation with the group indexed by Z_1. ◇

The moral of the theorem is, again, that in Monte Carlo computations, one is encouraged to do as much analytical work as possible. Bringing down dimensionality is almost surely a good practice, although some examples exist in which one actually wants to *increase* the dimension of the space to improve the efficiency of the Monte Carlo algorithms (Section 7). This theorem was used in MacEachern, Clyde and Liu (1999) to justify a new importance sampling algorithm for a nonparametric Bayesian inference problem.

Another place to use Rao-Blackwellization is in estimation. For example, if the sample $\mathbf{x}^{(j)}$ can be decomposed as $(x_1^{(j)}, x_2^{(j)})$ and if $E_\pi[h(\mathbf{x}) \mid x_2]$ is available in closed-form, then an often more efficient estimator of $\mu = E_\pi h(\mathbf{x})$ is

$$\breve{\mu} = \frac{1}{W}\sum_{j=1}^{m} w^{(j)} E_\pi[h(\mathbf{x}) \mid x_2^{(j)}], \quad W = \sum_{j=1}^{m} w^{(j)},$$

whose asymptotic unbiasedness is easily shown. However, it is no longer as trivial as in Section 2.3 to prove its optimality — it in fact can not be proved that the new estimator $\breve{\mu}$ is always better than the plain estimator in (2.3).

2.5.6 Example: Solving a linear system

It has been noted that many deterministic systems can be solved by Monte Carlo methods (Hammersley and Handscomb 1964, Ripley 1987). Such systems include the boundary problems of partial differential equations, general high-dimensional linear equations, and other fixed-point problems. We here follow the general formulation in Section 4 of Griffiths and Tavare (1994). Suppose a system can be written in a recursive form as

$$q(\mathbf{x}) = \sum_{\mathbf{y}\in\mathcal{A}} r(\mathbf{x},\mathbf{y})q(\mathbf{y}) + \sum_{\mathbf{z}\in\mathcal{B}} r(\mathbf{x},\mathbf{z})q(\mathbf{z}), \quad \text{for } \mathbf{x}\in\mathcal{B}, \tag{2.11}$$

where $q(\mathbf{x})$ is known for $\mathbf{x} \in \mathcal{A}$ and is unknown for $\mathbf{x} \in \mathcal{B}$ (this happens in solving a difference equation with a given boundary condition). By repeatedly substituting the unknown $q(\mathbf{z})$ in the right-hand side of (2.11) by relationship (2.11), we have

$$\begin{aligned} q(\mathbf{x}) &= \sum_{\mathbf{y}\in\mathcal{A}} r(\mathbf{x},\mathbf{y})q(\mathbf{y}) + \sum_{\mathbf{y}_1\in\mathcal{B}}\sum_{\mathbf{y}\in\mathcal{A}} r(\mathbf{x},\mathbf{y}_1)r(\mathbf{y}_1,\mathbf{y})q(\mathbf{y}) \\ &\quad + \sum_{\mathbf{y}_1\in\mathcal{B}}\sum_{\mathbf{y}_2\in\mathcal{B}}\sum_{\mathbf{y}\in\mathcal{A}} r(\mathbf{x},\mathbf{y}_1)r(\mathbf{y}_1,\mathbf{y}_2)r(\mathbf{y}_2,\mathbf{y})q(\mathbf{y}) + \cdots \\ &= \sum_{k=0}^{\infty}\left\{\sum_{\mathbf{y}_1\in\mathcal{B}}\cdots\sum_{\mathbf{y}_k\in\mathcal{B}}\sum_{\mathbf{y}\in\mathcal{A}} r(\mathbf{y}_1,\mathbf{y}_2)\cdots r(\mathbf{y}_k,\mathbf{y})q(\mathbf{y})\right\}. \quad (2.12) \end{aligned}$$

Suppose we can construct a Markov transition function $A(\mathbf{x},\mathbf{y})$ that satisfies the following conditions: (a) $A(\mathbf{x},\mathbf{y}) > 0$ whenever $r(\mathbf{x},\mathbf{y}) > 0$ and (b) the chain visits $\mathcal{A}$ with probability 1 starting from any $\mathbf{x} \in \mathcal{B}$. Then, for any given $\mathbf{x}_0 \in \mathcal{B}$, we can simulate this Markov chain (some basics of Markov chain theory is given in Chapter 12) with $\mathbf{x}_0$ as its initial state (i.e. $\boldsymbol{X}_0 = \mathbf{x}_0$) and run the chain until it hits $\mathcal{A}$ for the first time. With this construction, expression (2.12) can be rewritten probabilistically as

$$q(\mathbf{x}_0) = E_{\mathbf{x}_0}\left\{q(\boldsymbol{X}_\tau)\prod_{k=1}^{\tau}\frac{r(\boldsymbol{X}_{k-1},\boldsymbol{X}_k)}{A(\boldsymbol{X}_{k-1},\boldsymbol{X}_k)}\right\},$$

where τ is the first time the chain visits $\mathcal{A}$ (the hitting time). Thus, $q(\mathbf{x})$ can be estimated as follows. We run m independent Markov chains with the transition function $A(\cdot,\cdot)$ and the starting value $\mathbf{x}_0$ until hitting $\mathcal{A}$. Let these chains be $\boldsymbol{X}_1^{(j)},\ldots,\boldsymbol{X}_{\tau_j}^{(j)}$, where τ_j is the hitting time of the jth chain; then,

$$\hat{q} = \frac{1}{m}\left\{\sum_{j=1}^{m} q(\boldsymbol{X}_{\tau_j}^{(j)})\prod_{k=1}^{\tau_j}\frac{r(\boldsymbol{X}_{k-1}^{(j)},\boldsymbol{X}_k^{(j)})}{A(\boldsymbol{X}_{k-1}^{(j)},\boldsymbol{X}_k^{(j)})}\right\}.$$

However, this method is usually inferior to its deterministic counterpart except for a few special cases (Ripley 1987). For example, this Monte Carlo approach might be attractive when one is only interested in estimating a few values of $q(\mathbf{x})$. We will discuss in Section 4.1.2 and Chapter 3 several techniques (e.g., resampling, rejection control, etc.) for improving efficiencies of the importance sampling method [see also Liu and Chen (1998)]. A proper implementation of these new techniques might lead to a Monte Carlo method that is more appealing than the corresponding deterministic approach (Chen and Liu 2000b).

2.5.7 Example: A Bayesian missing data problem

In statistics, it is often the case that part of the data is missing which renders the standard likelihood computation difficult (see the Appendix). The current example is constructed by Murray (1977) in the discussion of Dempster et al. (1977). Table 1 contains 12 observations assumed to be drawn from a bivariate normal distribution with known mean vector (0,0) and unknown covariance matrix.

1	1	−1	−1	2	2	−2	−2	*	*	*	*
1	−1	1	−1	*	*	*	*	2	2	−2	−2

TABLE 2.1. An artificial dataset of bivariate Gaussian observations with missing parts (Murray, 1977). The symbol * indicates that the value is missing.

Let ρ denote the correlation coefficient and let σ_1 and σ_2 denote the marginal variances. The complete data are $y_1, \ldots, y_{12}$, where $y_t = (y_{t,1}, y_{t,2})$ for $t = 1, \ldots, 12$. So the $y_{t,2}$ are missing for $t = 5, \ldots, 8$, whereas $y_{t,1}$ are missing for the $t = 9, \ldots, 12$. We are interested in the posterior distribution of ρ given the incomplete data. Note that the information on σ_1 and σ_2 provided by the eight incomplete observations cannot be ignored in drawing likelihood-based inference about ρ.

For simplicity, the covariance matrix of the bivariate normal is assigned the Jeffreys' noninformative prior distribution (Box and Tiao 1973)

$$\pi(\Sigma) \propto \mid \Sigma \mid^{-\frac{d+1}{2}}, \tag{2.13}$$

where d is the dimensionality and is equal to 2 in this example. The posterior distribution of Σ given t i.i.d. observed complete data $y_1, \ldots, y_t$ is

$$p(\Sigma \mid y_1, \ldots, y_t) \propto |\Sigma|^{-\frac{t+d+1}{2}} \exp\left\{-\frac{1}{2} \operatorname{tr}[\Sigma \cdot S]\right\},$$

where $S = (s_{ij})_{2\times 2}$ is the uncorrected sum of squares matrix (i.e., $s_{ij} = \sum_{s=1}^{t} y_{s,i} y_{s,j}$). This distribution is called the *inverse Wishart* distribution. Box and Tiao (1973) and Gelman, Carlin, Stern and Rubin (1995) give more details on the standard Bayes inference with multivariate Gaussian observations. An introduction on the general Bayesian inference can be found in the Appendix.

By letting $\mathcal{Z} = \Sigma^{-1}$, we see that $\mathcal{Z}$ follows a Wishart(t, S) distribution:

$$p(\mathcal{Z} | \text{complete data}) \propto \mathcal{Z}^{\frac{t-d-1}{2}} \exp\left\{-\frac{1}{2} \operatorname{tr}[\mathcal{Z} \cdot S]\right\}. \tag{2.14}$$

See Johnson and Kotz (1972) for a detailed derivation. In this distribution, n is often referred as its *degree of freedom* and S its scale matrix (required to be positive definite). Sampling from this Wishart distribution when $t \geq d+1$ can be accomplished as follows: Simulate t independent samples $\epsilon_1, \ldots, \epsilon_t$ from a d-dimensional multivariate Gaussian distribution, $N(0, S)$, and then let $Z = \sum_{i=1}^{t} \epsilon_i \epsilon_i^T$. Suppose S is decomposed as $S = CC^T$, a more efficient algorithm proposed by Odell and Feiveson (1966) is as follows:

(a) Simulate independent random samples $V_j \sim \chi^2(t-j)$, $j = 1, \ldots, d$.

(b) Simulate independent random variables $N_{ij} \sim N(0,1)$, for $i < j \leq d$.

(c) Construct a symmetric matrix $B = (b_{ij})_{d \times d}$ as follows:

$$\begin{aligned} b_{11} &= V_1, \quad b_{1j} = N_{1j}\sqrt{V_1}; \\ b_{jj} &= V_j + \sum_{i=1}^{j-1} N_{ij}^2, \quad j = 2, \ldots, d; \\ b_{ij} &= N_{ij}\sqrt{V_i} + \sum_{k=1}^{i-1} N_{ki}N_{kj}, \quad i < j \leq d. \end{aligned}$$

(d) Then, $Z = CBC^T$ follows the Wishart(t,S) distribution in (2.14).

Now let us go back to the original problem of making inference on ρ with incomplete data. In this case, we can write down the joint posterior distribution of Σ and missing data $\mathbf{y}_{\text{mis}} = (y_{5,2}, \ldots, y_{8,2}, y_{9,1}, \ldots, y_{12,1})$ as

$$\begin{aligned} p(\Sigma, \mathbf{y}_{\text{mis}} \mid \mathbf{y}_{\text{obs}}) &\propto p(\mathbf{y}_{\text{mis}}, \mathbf{y}_{\text{obs}} \mid \Sigma) p(\Sigma) \\ &\propto |\Sigma|^{-\frac{12+3}{2}} \exp\left\{ -\frac{1}{2} \text{tr}[\Sigma^{-1} \cdot S(\mathbf{y}_{\text{mis}})] \right\}, \end{aligned}$$

where $S(\mathbf{y}_{\text{mis}})$ emphasizes that its value depends on the value of $\mathbf{y}_{\text{mis}}$. Thus, the posterior distribution of Σ can be derived from the above joint distribution with $\mathbf{y}_{\text{mis}}$ integrated out. An importance sampling algorithm for achieving this goal can be implemented as follows:

- Sample Σ from some trial distribution $g_0(\Sigma)$,
- Draw $\mathbf{y}_{\text{mis}}$ from $p(\mathbf{y}_{\text{mis}} \mid \mathbf{y}_{\text{obs}}, \Sigma)$.

It should be noted that given Σ, the predictive distribution of $\mathbf{y}_{\text{mis}}$ is simply the Gaussian distribution. For example,

$$[y_{5,2} \mid \Sigma, \mathbf{y}_{\text{obs}}] = [y_{5,2} \mid \Sigma, y_{5,1}] \sim N(\mu_*, \sigma_*^2)$$

where $\mu_* = \rho y_{5,1}\sqrt{\sigma_{11}/\sigma_{22}}$ and $\sigma_*^2 = (1-\rho^2)\sigma_{11}$. Thus, the key question is how to draw Σ. A simple idea is to draw Σ from its posterior distribution conditional only on the first four complete observations:

$$g_0(\Sigma) \propto |\Sigma|^{-7/2} \exp\{-\text{tr}[\Sigma^{-1}S_4]/2\},$$

where S_4 refers to the sum of the square matrix computed from the first four observations. With g_0 so chosen, the estimated coefficient of variation of the importance weights is 2.25 with 5000 Monte Carlo samples. Of course, other choices of Σ are also possible.

For a comparison, we obtained the analytical form of the observed-data posterior distribution of ρ up to a normalizing constant:

$$p(\rho \mid \text{data in Table 2.1}) \propto \frac{[(1-\rho^2)^{4.5}]}{[(1.25-\rho^2)^8]}.$$

Figure 2.2 displays the Monte Carlo estimates of the density versus the true posterior density of ρ.

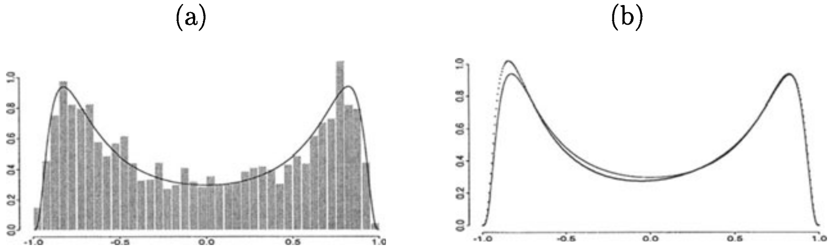


FIGURE 2.2. Importance sampling estimate of the posterior density of the correlation coefficient ρ (with 5000 iterations) for a bivariate Gaussian model with Murray's (1977) data. (a) the usual estimate with m=5000 overlaid by the "true" density; (b) the estimate resulting from Rao-Blackwellization (Section 2.5.5) with m=1000 (by courtesy of Mr. Yuguo Chen).

2.6 Advanced Importance Sampling Techniques

2.6.1 Adaptive importance sampling

It is often a good idea to "learn" about the target distribution of interest along with Monte Carlo sampling. A simple way of achieving this is to start with a trial density, say $g_0(\mathbf{x}) = t_\alpha(\mathbf{x}; \mu_0, \Sigma_0)$, where t_α represents a t-distribution with α degrees of freedom. With weighted Monte Carlo samples, one can estimate the mean and covariance matrix, denoted as μ_1 and Σ_1, respectively, of the target distribution. Then, a new trial density can be constructed as $g_1(\mathbf{x}) = t_\alpha(\mathbf{x}, \mu_1, \Sigma_1)$ (Oh and Berger 1992). This procedure can be iterated until a certain measure of discrepancy between the trial distribution and the target distribution, such as the coefficient of variation of the importance weights, does not improve any more.

Another way of doing an adaptation (Oh and Berger 1993) is to assume a parametric form for the new trial density $g_1(\mathbf{x})$ [e.g., suppose it is of the form $g(\mathbf{x};\lambda)$]. Then, we try to find the optimal choice of λ, defined as the one that minimizes, say, the estimated coefficient of variation of the importance weights, based on the current sample. Let $w(\mathbf{x};\lambda)=\pi(\mathbf{x})/g_1(\mathbf{x})$. We note that

$$\mathrm{var}_{g_1}(w)=\int \frac{\pi^2(\mathbf{x})}{g_1(\mathbf{x})g_0(\mathbf{x})}g_0(\mathbf{x})d\mathbf{x}-1.$$

Suppose we have drawn $\mathbf{x}^{(1)},\ldots,\mathbf{x}^{(m)}$ from the trial distribution $g_0(\mathbf{x})$. The coefficient of variation of $\pi(\mathbf{x})/g_1(\mathbf{x})$ can be approximated as

$$\widehat{cv}^2(\lambda)=\bar{H}(\lambda)-1,$$

where $\bar{H}(\lambda)=\sum_{j=1}^m H^{(j)}(\lambda)/m$ and

$$H^{(j)}(\lambda)=\frac{\pi^2(\mathbf{x}^{(j)})}{g_1(\mathbf{x}^{(j)})g_0(\mathbf{x}^{(j)})}=\frac{\{\pi(\mathbf{x}^{(j)})/g_0(\mathbf{x}^{(j)})\}^2}{g(\mathbf{x}^{(j)};\lambda)/g_0(\mathbf{x}^{(j)})}.$$

When $\pi(\mathbf{x})$ can only be evaluated up to a normalizing constant, we need to use the estimate

$$\widehat{cv}^2(\lambda)=\frac{\bar{H}(\lambda)}{\bar{W}_0^2}-1,$$

where $\bar{W}_0$ is the sample average of the un-normalized importance ratio $\pi(\mathbf{x}^{(j)})/g_0(\mathbf{x}^{(j)})$. Then, we can implement a Newton-Raphson method to find the optimal λ. Of particular interest is that for $\lambda=(\epsilon,\mu,\Sigma)$,

$$g(\mathbf{x};\lambda)=\epsilon g_0(\mathbf{x})+(1-\epsilon)t_\nu(\mathbf{x};\mu,\Sigma);$$

that is, the "improved version" is the mixture of the previous trial distribution g_0 with a new parametric component.

The reader should be cautious in using these adaptive methods since they are typically unstable. Perhaps a less greedy but more robust approach is to minimize a more robust distance measure between the trial and the target densities (e.g., the Hellinger or the Kullback-Leibler distance).

2.6.2 Rejection and weighting

When implementing the rejection method, one needs to find a trial density $g(\ )$ and an envelope constant M such that $\pi(\mathbf{x})\le Mg(\mathbf{x})$ for all $\mathbf{x}$. Its efficiency is determined as $1/M$; that is, on the average, M random samples have to be generated in order to produce an accepted one. Thus, finding a good M is crucial, but is nontrivial. Suppose one uses a reasonable M but is unsure whether the envelope inequality holds in the entire support of $\pi(\cdot)$; one can, in fact, accept those $\mathbf{x}$'s that lie in the region $\{\mathbf{x}:\ \pi(\mathbf{x})>Mg(\mathbf{x})\}$, and adjust the bias by giving these samples appropriate weights. In this way, we may achieve faster computation and better efficiency.

When applying importance sampling, one often produces random samples with very small importance weights because of a less than ideal trial density. Suppose we are interested in estimating $E_\pi[h(\mathbf{x})]$, but the evaluation of $h(\mathbf{x})$ is expensive. In this case, we would like to evaluate as few samples as possible but without losing much information or creating a bias. The following simple technique for combining rejection and importance weighting can be used.

Suppose we have drawn samples $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$ from $g(\mathbf{x})$. Let $w^{(j)} = \pi(\mathbf{x}^{(j)})/g(\mathbf{x}^{(j)})$. We can conduct the the following operation for any given threshold value $c > 0$:

Rejection Control (RC)

- For $j = 1, \ldots, m$, accept $\mathbf{x}^{(j)}$ with probability
$$r^{(j)} = \min\left\{1, \frac{w^{(j)}}{c}\right\}.$$
- If the jth sample $\mathbf{x}^{(j)}$ is accepted, its weight is updated to $w^{(*j)} = q_c w^{(j)}/r^{(j)}$, where
$$q_c = \int \min\left\{1, \frac{w(\mathbf{x})}{c}\right\} g(\mathbf{x}) d\mathbf{x}.$$

Constant q_c is maintained only for conceptual clarity instead of computational need in estimating a expectation with respect to π. This is because q_c, the same for all the accepted samples, is not needed for the evaluation of the ratio estimate in (2.3). But in cases where one is interested in estimating the normalizing constant of the target distribution (also called the *partition function*), one may need a good estimate of q_c.

The above RC scheme can be viewed as a technique for adjusting the trial density g in light of current importance weights. The new trial density $g^*(\mathbf{x})$ resulting from this adjustment is expected to be closer to the target function $\pi(\mathbf{x})$. In fact, it can be seen that

$$g^*(\mathbf{x}) = q_c^{-1} \min\{g(\mathbf{x}), \pi(\mathbf{x})/c\}. \tag{2.15}$$

Because of the relationship

$$q_c = \int \min\{g(\mathbf{x}), \pi(\mathbf{x})/c\} dx = E_g \min\left\{1, \frac{w(\mathbf{x})}{c}\right\} d\mathbf{x},$$

the normalizing constant q_c can be unbiasedly estimated from the sample as

$$\hat{p}_c = \frac{1}{m} \sum_{j=1}^{m} \min\left\{1, \frac{w^{(j)}}{c}\right\}.$$

After applying rejection control, we will typically have fewer than N samples. More samples can be drawn from either $g(x)$ or $g^*(x)$ (via rejection control) to make up for the rejected samples. The usefulness of the method in sequential importance sampling as shown by Liu, Chen and Wong (1998) will be discussed in the next chapter. Theoretically, one can show the following:

Theorem 2.6.1 *The rejection control method indeed reduces the χ^2 distance between the target distribution and the modified trial distribution; that is,*

$$var_{g*}[\pi(\mathbf{x})/g^*(\mathbf{x})] \leq var_g[\pi(\mathbf{x})/g(\mathbf{x})]. \tag{2.16}$$

Proof: With $w(\mathbf{x}) = \pi(\mathbf{x})/g(\mathbf{x})$, the rejection probability q_c in (2.15) can be expressed as

$$q_c = \int \min\left\{g(\mathbf{x}), \frac{\pi(\mathbf{x})}{c}\right\} d\mathbf{x} = \frac{1}{c}E_g[\min\{w(\mathbf{x}), c\}]. \tag{2.17}$$

On the other hand, we have

$$\begin{aligned} 1 + \text{var}_{g*}\left\{\frac{\pi(\mathbf{x})}{g^*(\mathbf{x})}\right\} &= \int \frac{\pi^2(\mathbf{x})}{g^*(\mathbf{x})}d\mathbf{x} = q_c \int \frac{\pi(\mathbf{x})}{\min\{g(\mathbf{x}), \pi(\mathbf{x})/c\}}\pi(\mathbf{x})d\mathbf{x} \\ &= \int q_c \max\{w(\mathbf{x}), c\}\pi(\mathbf{x})d\mathbf{x} \qquad (2.18) \\ &= q_c E_g[\max\{w(\mathbf{x}), c\}w(\mathbf{x})]. \qquad (2.19) \end{aligned}$$

Now we show that for any $w_1 > 0, w_2 > 0$,

$$h(w_1, w_2) = [\min\{w_1, c\} - \min\{w_2, c\}][w_1 \max\{w_1, c\} - w_2 \max\{w_2, c\}] \geq 0.$$

There are three scenarios for value c: (i) $\min(w_1, w_2) > c$, then $h(w_1, w_2) = 0$; (ii) $c > \max(w_1, w_2)$, then $h(w_1, w_2) = c(w_1 - w_2)^2 \geq 0$; and (iii) c is between w_1 and w_2, in which case we assume without loss of generality that $w_1 \leq c \leq w_2$, then

$$h(w_1, w_2) = (c - w_1)(w_2^2 - cw_1) \geq 0.$$

Hence, the two random variables $\min\{w(\mathbf{x}), c\}$ and $w(\mathbf{x})\max\{w(\mathbf{x}), c\}$ are positively correlated. Together with the fact that

$$\min\{w(\mathbf{x}), c\}\max\{w(x), c\} = cw(x),$$

and formulas (2.17) and (2.19), we have

$$\begin{aligned} c\left[1 + \text{var}_{g*}\left\{\frac{\pi(\mathbf{x})}{g^*(\mathbf{x})}\right\}\right] &= E_g[\min\{w(\mathbf{x}), c\}]E_g[\max\{w(\mathbf{x}), c\}w(\mathbf{x})] \\ &\leq E_g[\min\{w(\mathbf{x}), c\}\max\{w(\mathbf{x}), c\}w(\mathbf{x})] \\ &= cE_g[w^2(\mathbf{x})] = c\left[1 + \text{var}_g\left\{\frac{\pi(\mathbf{x})}{g(\mathbf{x})}\right\}\right]. \end{aligned}$$

Hence, we have proved the result (2.16). ◇

2.6.3 Sequential importance sampling

It is nontrivial to design a good trial distribution for doing importance sampling in high-dimensional problems. One of the most useful strategies in these problems is to build up the trial density sequentially. Suppose we can decompose $\mathbf{x}$ as $\mathbf{x} = (x_1, \ldots, x_d)$ where each of the x_j may be multidimensional. Then, our trial density can be constructed as

$$g(\mathbf{x}) = g_1(x_1) g_2(x_2 \mid x_1) \cdots g_d(x_d \mid x_1, \ldots, x_{d-1}), \tag{2.20}$$

by which we hope to obtain some guidance from the target density while building up the trial density. Corresponding to the decomposition of $\mathbf{x}$, we can rewrite the target density as

$$\pi(\mathbf{x}) = \pi(x_1)\pi(x_2 \mid x_1) \cdots \pi(x_d \mid x_1, \ldots, x_{d-1}) \tag{2.21}$$

and the importance weight as

$$w(\mathbf{x}) = \frac{\pi(x_1)\pi(x_2 \mid x_1) \cdots \pi(x_d \mid x_1, \ldots, x_{d-1})}{g_1(x_1) g_2(x_2 \mid x_1) \cdots g_d(x_d \mid x_1, \ldots, x_{d-1})}. \tag{2.22}$$

Equation (2.22) suggests a recursive way of computing and monitoring the importance weight; that is, by denoting $\mathbf{x}_t = (x_1, \ldots, x_t)$ (thus, $\mathbf{x}_d \equiv \mathbf{x}$), we have

$$w_t(\mathbf{x}_t) = w_{t-1}(\mathbf{x}_{t-1}) \frac{\pi(x_t \mid \mathbf{x}_{t-1})}{g_t(x_t \mid \mathbf{x}_{t-1})}.$$

At the end, w_d is equal to $w(\mathbf{x})$ in (2.22). Potential advantages of this recursion and (2.21) are the following: (a) We can stop generating further components of $\mathbf{x}$ if the *partial weight* derived from the sequentially generated *partial sample* is too small and (b) we can take advantage of $\pi(x_t \mid \mathbf{x}_{t-1})$ in designing $g_t(x_t \mid \mathbf{x}_{t-1})$. In other words, the marginal distribution $\pi(\mathbf{x}_t)$ can be used to guide the generation of $\mathbf{x}$.

Although the above "idea" sounds interesting, the trouble is that the decomposition of π as in (2.21) and that of w as in (2.22) are impractical at all! The reason is that in order to get (2.21), one needs to have the marginal distribution

$$\pi(\mathbf{x}_t) = \int \pi(x_1, \ldots, x_d) dx_{t+1} \cdots dx_d,$$

whose computation involves integrating out components $x_{t+1}, \ldots, x_d$ in $\pi(\mathbf{x})$ and is as difficult as — or even more difficult than — the original problem.

In order to carry out the sequential sampling idea, we need to introduce another layer of complexity. Suppose we can find a sequence of "auxiliary distributions," $\pi_1(x_1), \pi_2(\mathbf{x}_2), \ldots, \pi_d(\mathbf{x})$, so that $\pi_t(\mathbf{x}_t)$ is a reasonable approximation to the marginal distribution $\pi(\mathbf{x}_t)$, for $t = 1, \ldots, d-1$ and

$\pi_d = \pi$. We want to emphasize that the π_t are only required to be known up to a normalizing constant and they *only* serve as "guides" to our construction of the whole sample $\mathbf{x} = (x_1, \ldots, x_d)$. The *sequential importance sampling* (SIS) method can then be defined as the following recursive procedure (for $t = 2, \ldots, d$).

SIS Step:

(A) Draw $X_t = x_t$ from $g_t(x_t | \mathbf{x}_{t-1})$, and let $\mathbf{x}_t = (\mathbf{x}_{t-1}, x_t)$.

(B) Compute

$$u_t = \frac{\pi_t(\mathbf{x}_t)}{\pi_{t-1}(\mathbf{x}_{t-1}) g_t(x_t \mid \mathbf{x}_{t-1})}, \tag{2.23}$$

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

In the SIS step, we call u_t an "incremental weight." It is easy to show that $\mathbf{x}_t$ is properly weighted by w_t with respect to π_t provided that $\mathbf{x}_{t-1}$ is properly weighted by w_{t-1} with respect to π_{t-1}. Thus, the whole sample $\mathbf{x}$ obtained in this sequential fashion is properly weighted by the final importance weight, w_d, with respect to the target density $\pi(\mathbf{x})$. One reason for the sequential buildup of the trial density is that it breaks a difficult task into manageable pieces. The SIS framework is particularly attractive, as it can use the sequence of "auxiliary distributions" $\pi_1, \pi_2, \ldots, \pi_d$ to help construct more efficient trial distribution:

- We can build g_t in light of π_t. For example, one can choose (if possible)

$$g_t(x_t \mid \mathbf{x}_{t-1}) = \pi_t(x_t \mid \mathbf{x}_{t-1}).$$

  Then, the incremental weight becomes

$$u_t = \pi_t(\mathbf{x}_t) / \pi_{t-1}(\mathbf{x}_{t-1}).$$

- When we observe that w_t is getting too small, we can choose to reject the sample halfway and restart again. In this way, we avoid wasting time on generating samples that are doomed to have little effect in the final estimation. However, as an outright rejection incurs bias, the rejection control technique described in Section 2.6.2 can be used to correct such bias (Section 2.6.4)

In configurational bias Monte Carlo (Siepmann and Frenkel 1992), the SIS is used as a proposal (independent) transition in a Metropolis-Hastings algorithm (see Section 5.4.3).

The most important unanswered question in the SIS framework is how to find a reasonable set of "auxiliary distributions." This issue will be illustrated through several practical examples in Chapters 3 and 4. For example, in a nonlinear filtering problem, the "auxiliary distributions" often correspond to the "current" posterior distributions of the true signals.

2.6.4 *Rejection control in sequential importance sampling*

Although the rejection control method (Section 2.6.2) was described in a "static" form, it can be applied dynamically to improve an SIS scheme. Suppose a sequence of "check points," $0 < t_1 < t_2 < \cdots < t_k \leq d$, and a sequence of threshold values $c_1, \ldots, c_k$, are given in advance. The following procedure can be implemented:

1. At each check point t_j, start RC(t_k) as described in Section 2.6.2 with the threshold value $c = c_j$. If the partial sample $(x_1, \ldots, x_{t_j})$ has a weight w_{t_j}, then we accept this partial sample with probability $\min\{1, w_{t_j}/c_j\}$ and, if accepted, replace its weight by $w^*_{t_j} = \max\{w_{t_j}, c_j\}$.

2. For each rejected partial sample, restart from the beginning again and let it pass through all the check points at $t_1, \ldots, t_j$, with threshold values $c_1, \ldots, c_j$, respectively. If rejected in any middle check point, start again.

Note that after the first rejection control at stage t_1, the sampling distribution $g^*_t(\mathbf{x}_t)$ for $\boldsymbol{X}_t$ is no longer the same as the one described in (2.20). It is shown by Liu, Chen and Wong (1998) that for any time t, partial sample $\mathbf{x}_t$ resulting from the above procedure is properly weighted with respect to π_t by their modified weights w^*_t. To retain a proper estimate of the normalizing constant for π, one has to estimate p_c, the probability of acceptance, and adjust the weight to $p_c w^*_t$. Since this method requires that each rejected sample be restarted from stage 0, it tends to be impractical when the number of components d is large. An interesting way to combine the RC operation with resampling is described in Section 3.4.5.

2.7 Application of SIS in Population Genetics

Evolutionary theory holds that stochastic mutational events may alter the genome of an individual and that these changes may be passed to its progeny. Thus, comparing homologous DNA regions (segments) of a random sample of individuals taken from a certain population can shed light on the evolutionary process of this population. This comparison can also yield important information for locating genes that are responsible for genetic diseases. Recent advances in the biotechnology revolution have provided a wealth of DNA sequence data for which meaningful studies on the evolution process can be made and biologically verified.

Following Griffiths and Tavare (1994) and Stephens and Donnelly (2000), we consider the simplest demographic model focusing on populations of constant size N which evolve in non-overlapping generations. Each individual in a population is a sufficiently small chromosomal region in which no recombination is allowed (in reality, recombination can happen with a very small probability). Thus, each chromosomal segment seen in the dataset can be treated as a descendent of a single parental segment in the previous generation — and it is sufficient to consider the haploid model (i.e., each "individual" only has one parent). Each segment has a genetic type and the set of all possible types is denoted as E. If a parental segment is of type $\alpha \in E$, then its progeny is of type $\alpha \in E$ with probability $1 - \mu$ and of type $\beta \in E$ with probability $\mu P_{\alpha\beta}$. Thus, μ can be seen as the mutation rate per chromosome per generation. The mutation transition matrix $\boldsymbol{P} = (P_{\alpha\beta})$ is assumed to have a unique stationary distribution.

Suppose we observe a random sample from the current population assumed to be at stationarity. The ancestral relationships among the individuals in the sample — when being traced back to their most recent common ancestor (MRCA) — can be described by a tree. Figure 2.3 shows a genealogical tree for an example when the segment has only two genetic types, C and T (i.e., $E = \{C, T\}$), and the sample consists of five observations.

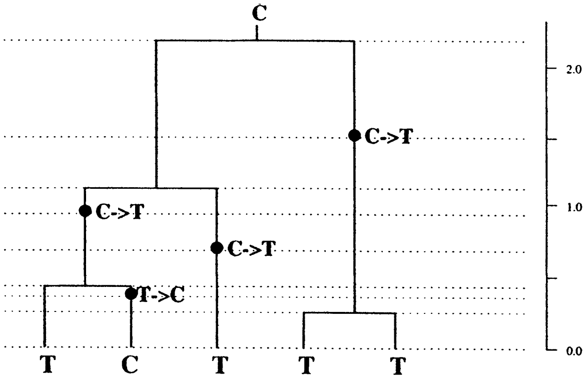


FIGURE 2.3. Illustration of a genealogical tree. The set of five observed individuals at the current time is $\{T, C, T, T, T\}$ (the bottom of the tree). The plotted tree illustrates a possible route for these five individuals to descend from a common ancestor of type C. Ancestral lineages are jointed by horizontal lines (and are said to coalesce) when they share a common ancestor. The dots represent mutations and the horizontal dotted lines indicate the times at which events (coalescence and mutations) occur. The history $\mathcal{H} = (H_{-k}, H_{-(k-1)}, \ldots, H_{-1}, H_0)$ in this case is $(\{C\}, \{C, C\}, \{C, T\}, \{C, C, T\}, \{C, T, T\}, \{T, T, T\}, \{T, T, T, T\}, \{C, T, T, T\}, \{C, T, T, T, T\})$. Reproduced from Stephens and Donnelly (2000).

Stephens and Donnelly (2000) used $\mathcal{H} = (H_{-m}, \ldots, H_{-1}, H_0)$ to denote the whole ancestral history (unobserved) of the observed individuals at the present time, where k is the first time when all the individuals in the sample coalesce (i.e., the first time they have a common ancestor). Each H_{-i} is an unordered list of genetic types of the ancestors i generations ago. Thus, the history $\mathcal{H}$ has a one-to-one correspondence with the tree topology (evolution time is not reflected in $\mathcal{H}$). Note that only H_0 is observable. For any given $\mathcal{H}$ that is compatible with H_0, however, we can compute the likelihood function $p_\theta(\mathcal{H})$ as

$$p_\theta(\mathcal{H}) \propto p_\theta(H_{-k})p_\theta(H_{-k+1} \mid H_{-k}) \cdots p_\theta(H_0 \mid H_{-1})p_\theta(\text{stop} \mid H_0).$$

Here, $p_\theta(H_{-k}) = \pi_0(H_{-k})$, with π_0 being the stationary distribution of $\boldsymbol{P}$. The coalescence theory (Griffiths and Tavare 1994, Stephens and Donnelly 2000) tells us that

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

for $i = -(k-1), \ldots, 0$ and the process is stopped just before a new genetic type is produced:

$$p_\theta(\text{stop}|H_0) = \sum_\alpha \frac{n_\alpha}{n}\frac{n-1}{n-1+\theta}. \tag{2.24}$$

Here, n is the sample size at generation H_{i-1} and n_α is the number of chromosomes of type α in the sample. The notation $H_i = H_{i-1}+\alpha$ indicates that the new generation H_i is obtained from H_{i-1} by a split of a line of type α, and the notation $H_i = H_{i-1} - \alpha + \beta$ means that H_i is obtained from H_{i-1} by a mutation from a type α to a type β. The parameter $\theta = 2N\mu/\nu$, with N being the population size (assumed to be constant throughout the history) and ν^2 being the variance of the number of progeny of a random chromosome.

To infer the value of θ, one can use the MLE method (see the Appendix for more descriptions), which requires us to compute for each given θ the likelihood value

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

This computation cannot be solved analytically and we have to resort to some approximation methods — Monte Carlo appears to be a natural choice. In a naive Monte Carlo, one may randomly choose the generation number k and then simulate forward from H_{-k}, which only has a single individual, to H_0. However, except for trivial dataset, such simulated history $\mathcal{H}$ has little chance to be compatible with the observed H_0. An alternative strategy is to simulate $\mathcal{H}$ backward starting from H_0 and then use weight to correct bias. In a sequential importance sampling strategy (equivalent to the method of Griffiths and Tavare (1994)), we can simulate $H_{-1}, H_{-2}, \ldots$

from a trial distribution built up sequentially by reversing the forward sampling probability at a fixed θ_0; that is, for $t = 1, \ldots, k$, we have

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

and the final trial distribution

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

In other words, each g_t is the "local" posterior distribution of H_{-t}, under a uniform prior, conditional on H_{-t+1}. By simulating from $g(\ )$ multiple copies of the history, $\mathcal{H}^{(j)}$, $j = 1, \ldots, m$, we can approximate the likelihood function as

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

In this approach, the choice of θ_0 can influence the final result. We tested this importance sampling method on a small test dataset in Stephens and Donnelly (2000), {8, 11, 11, 11, 11, 12, 12, 12, 12, 13}, with $E = \{0, 1, \ldots, 19\}$ and a simple random walk mutation transition on E. Figure 2.4 shows the likelihood curve of θ estimated from m=1,000,000 Monte Carlo samples and $\theta_0 = 10$. Stephens and Donnelly (2000) recently proposed a new SIS construction of the trial distribution and is significantly better than the simple construction described in this section. We will discuss a resampling method in Section 4.1.2 that can improve both algorithms.

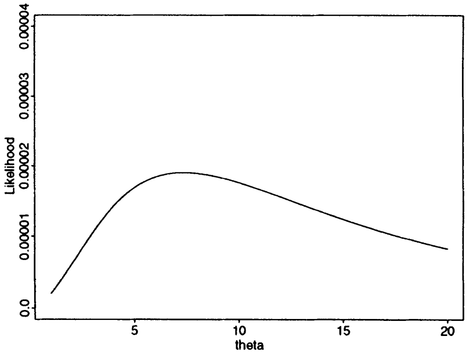


FIGURE 2.4. The estimation of the likelihood function for a dataset in Stephens and Donnelly (2000), with $\theta_0 = 10$ and m=1,000,000 iterations.

2.8 Problems

1. Evaluate integral $\int_0^1 \sin^2(1/x)dx$ by both a deterministic method and a Monte Carlo method. Comment on relative advantages and disadvantages.

2. Prove that the rejection method does produce random variables that follow the target distribution π. Show that the expected acceptance rate is $1/c$, where c is the "envelope constant."

3. Implement an adaptive importance sampling algorithm to evaluate mean and variance of a density

$$\pi(\mathbf{x}) \propto N(\mathbf{x}; \mathbf{0}, 2I_4) + 2N(\mathbf{x}; 3\boldsymbol{e}, I_4) + 1.5N(\mathbf{x}; -3\boldsymbol{e}, D_4),$$

   where $\boldsymbol{e} = (1,1,1,1)$, $I_4 = \text{diag}(1,1,1,1)$, and $D_4 = \text{diag}(2,1,1,.5)$. A possible procedure is as follows:

   - Start with a trial density $g_0 = t_\nu(0, \Sigma)$;
   - Recursively, we build

$$g_k(\mathbf{x}) = (1-\epsilon)g_{k-1}(\mathbf{x}) + \epsilon t_\nu(\mu, \Sigma),$$

     in which one chooses (ϵ, μ, Σ) to minimize the variation of coefficient of the importance weights.

4. Describe the process of simulating from a Wishart distribution and prove that the proposed method is correct.

5. Formulate the method described in Section 2.5.6 for the continuous state space and use it to solve a differential equation.