Contents

1

Introduction

There must be, he thought, some key, some crack in this mystery he could use to achieve an answer.
—P.C. Doherty, *Crown in Darkness*

Until the advent of powerful and accessible computing methods, the experimenter was often confronted with a difficult choice. Either describe an accurate model of a phenomenon, which would usually preclude the computation of explicit answers, or choose a standard model which would allow this computation, but may not be a close representation of a realistic model. This dilemma is present in many branches of statistical applications, for example, in electrical engineering, aeronautics, biology, networks, and astronomy. To use realistic models, the researchers in these disciplines have often developed original approaches for model fitting that are customized for their own problems. (This is particularly true of physicists, the originators of Markov chain Monte Carlo methods.) Traditional methods of analysis, such as the usual numerical analysis techniques, are not well adapted for such settings.

In this introductory chapter, we examine some of the statistical models and procedures that contributed to the development of simulation-based inference. The first section of this chapter looks at some statistical models, and the remaining sections examine different statistical methods. Throughout these sections, we describe many of the computational difficulties associated with the methods. The final section of the chapter contains a discussion of deterministic numerical analysis techniques.

1.1 Statistical Models

In a purely statistical setup, computational difficulties occur at both the level of *probabilistic modeling* of the inferred phenomenon and at the level of *statistical inference* on this model (estimation, prediction, tests, variable selection,

etc.). In the first case, a detailed representation of the causes of the phenomenon, such as accounting for potential explanatory variables linked to the phenomenon, can lead to a probabilistic structure that is too complex to allow for a parametric representation of the model. Moreover, there may be no provision for getting closed-form estimates of quantities of interest. One setup with this type of complexity is *expert systems* (in medicine, physics, finance, etc.) or, more generally, *graph structures*. See Pearl (1988), Robert (1991),[1] Spiegelhalter et al. (1993), Lauritzen (1996) for examples of complex expert systems.[2]

Another situation where model complexity prohibits an explicit representation appears in econometrics (and in other areas) for structures of *latent* (or *missing*) variable models. Given a "simple" model, aggregation or removal of some components of this model may sometimes produce such involved structures that simulation is really the only way to draw an inference. In these situations, an often used method for estimation is the EM algorithm (Dempster et al. 1977), which is described in Chapter 3. In the following example, we illustrate a common missing data situation. The concept and use of missing data techniques and in particular of the two following examples will reoccur throughout the book.

Example 1.1. Censored data models. *Censored data models* are missing data models where densities are not sampled directly. To obtain estimates and make inferences in such models usually requires involved computations and precludes analytical answers.

In a typical simple statistical model, we would observe random variables[3] (rv's) $Y_1, \ldots, Y_n$, drawn independently from a population with distribution $f(y|\theta)$. The distribution of the sample would then be given by the product $\prod_{i=1}^n f(y_i|\theta)$. Inference about θ would be based on this distribution.

In many studies, particularly in medical statistics, we have to deal with *censored* random variables; that is, rather than observing Y_1, we may observe $\min\{Y_1, \overline{u}\}$, where $\overline{u}$ is a constant. For example, if Y_1 is the survival time of a patient receiving a particular treatment and $\overline{u}$ is the length of the study being done (say $\overline{u} = 5$ years), then if the patient survives longer than 5 years, we do not observe the survival time, but rather the censored value $\overline{u}$. This modification leads to a more difficult evaluation of the sample density.

Barring cases where the censoring phenomenon can be ignored, several types of censoring can be categorized by their relation with an underlying (unobserved) model, $Y_i \sim f(y_i|\theta)$:

[1] Claudine, not Christian!

[2] Section 10.6.4 also gives a brief introduction to graphical models in connection with Gibbs sampling.

[3] Throughout the book we will use uppercase Roman letters for random variables and lowercase Roman letters for their realized values. Thus, we would observe $X = x$, where the random variable X produces the observation x. (For esthetic purposes this distinction is sometimes lost with Greek letter random variables.)

(i) Given random variables Y_i, which are, for instance, times of observation or concentrations, the actual observations are $Y_i^* = \min\{Y_i, \overline{u}\}$, where $\overline{u}$ is the maximal observation duration, the smallest measurable concentration rate, or some other truncation point.

(ii) The original variables Y_i are kept in the sample with probability $\rho(y_i)$ and the number of censored variables is either known or unknown.

(iii) The variables Y_i are associated with auxiliary variables $X_i \sim g$ such that $y_i^* = h(y_i, x_i)$ is the observation. Typically, $h(y_i, x_i) = \min(y_i, x_i)$. The fact that truncation occurred, namely the variable $\mathbb{I}_{Y_i > X_i}$, may be either known or unknown.

As a particular example, if

$$X \sim \mathcal{N}(\theta, \sigma^2) \quad \text{and} \quad Y \sim \mathcal{N}(\mu, \tau^2),$$

the variable $Z = X \wedge Y = \min(X, Y)$ is distributed as

$$\left[1 - \Phi\left(\frac{z-\theta}{\sigma}\right)\right] \times \tau^{-1}\varphi\left(\frac{z-\mu}{\tau}\right) + \left[1 - \Phi\left(\frac{z-\mu}{\tau}\right)\right] \sigma^{-1}\varphi\left(\frac{z-\theta}{\sigma}\right), \tag{1.1}$$

where φ is the density of the normal $\mathcal{N}(0,1)$ distribution and Φ is the corresponding cdf, which is not easy to compute.

Similarly, if X has a Weibull distribution with two parameters, $\mathcal{W}e(\alpha, \beta)$, and density

$$f(x) = \alpha\beta x^{\alpha-1} \exp(-\beta x^\alpha)$$

on $\mathbb{R}^+$, the observation of the censored variable $Z = X \wedge \omega$, where ω is constant, has the density

$$f(z) = \alpha\beta z^{\alpha-1} e^{-\beta z^\alpha}\, \mathbb{I}_{z \le \omega} + \left(\int_\omega^\infty \alpha\beta x^{\alpha-1} e^{-\beta x^\alpha} dx\right) \delta_\omega(z)\,, \tag{1.2}$$

where $\delta_a(\cdot)$ is the Dirac mass at a. In this case, the weight of the Dirac mass, $P(X \ge \omega)$, can be explicitly computed (Problem 1.4).

The distributions (1.1) and (1.2) appear naturally in quality control applications. There, testing of a product may be of a duration ω, where the quantity of interest is time to failure. If the product is still functioning at the end of the experiment, the observation on failure time is censored. Similarly, in a longitudinal study of a disease, some patients may leave the study either due to other causes of death or by simply being lost to follow-up. ‖

In some cases, the additive form of a density, while formally explicit, prohibits the computation of the density of a sample $(X_1, \ldots, X_n)$ for n large. (Here, "explicit" has the restrictive meaning that "it can be computed in a reasonable time.")

Example 1.2. Mixture models. Models of *mixtures of distributions* are based on the assumption that the observations X_i are generated from one of k elementary distributions f_j with probability p_j, the overall density being

$$X \sim p_1 f_1(x) + \ldots + p_k f_k(x) \,. \tag{1.3}$$

If we observe a sample of independent random variables $(X_1, \ldots, X_n)$, the sample density is

$$\prod_{i=1}^{n} \{p_1 f_1(x_i) + \cdots + p_k f_k(x_i)\} \,.$$

When $f_j(x) = f(x|\theta_j)$, the evaluation of the likelihood at a given value of $(\theta_1, \ldots, \theta_n, p_1, \ldots, p_n)$ only requires on the order[4] of $\mathcal{O}(kn)$ computations, but we will see later that likelihood and Bayesian inferences both require the expansion of the above product, which involves $\mathcal{O}(k^n)$ computations, and is thus prohibitive to compute in large samples.[5]

While the computation of standard moments like the mean or the variance of these distributions is feasible in many setups (and thus so is the derivation of moment estimators, see Problem 1.6), the representation of the likelihood function (and therefore the analytical computation of maximum likelihood or Bayes estimates) is generally impossible for mixtures. ‖

Finally, we look at a particularly important example in the processing of temporal (or time series) data where the likelihood cannot be written explicitly.

Example 1.3. Moving average model. An MA(q) model describes variables (X_t) that can be modeled as $(t = 0, \ldots, n)$

$$X_t = \varepsilon_t + \sum_{j=1}^{q} \beta_j \varepsilon_{t-j} \,, \tag{1.4}$$

where for $i = -q, -(q-1), \ldots$, the ε_i's are iid random variables $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ and for $j = 1, \ldots, q$, the β_j's are unknown parameters. If the sample consists of the observation $(X_0, \ldots, X_n)$, where $n > q$, the sample density is (Problem 1.5)

[4] Recall that the notation $\mathcal{O}(n)$ denotes a function that satisfies
$0 < \limsup_{n\to\infty} \mathcal{O}(n)/n < \infty$.

[5] This class of models will be used extensively over the book. Although the example is self-contained, detailed comments about such models are provided in Note 9.7.1 and Titterington et al. (1985).

$$\int_{\mathbb{R}^q} \sigma^{-(n+q)} \prod_{i=1}^{q} \varphi\left(\frac{\varepsilon_{-i}}{\sigma}\right) \varphi\left(\frac{x_0 - \sum_{i=1}^{q} \beta_i \varepsilon_{-i}}{\sigma}\right)$$
$$\times \varphi\left(\frac{x_1 - \beta_1 \hat{\varepsilon}_o - \sum_{i=2}^{q} \beta_i \varepsilon_{1-i}}{\sigma}\right) \cdots \tag{1.5}$$
$$\times \varphi\left(\frac{x_n - \sum_{i=1}^{q} \beta_i \hat{\varepsilon}_{n-i}}{\sigma}\right) \, d\varepsilon_{-1} \cdots d\varepsilon_{-q} \,,$$

with

$$\hat{\varepsilon}_0 = x_0 - \sum_{i=1}^{q} \beta_i \varepsilon_{-i} \,,$$
$$\hat{\varepsilon}_1 = x_1 - \sum_{i=2}^{q} \beta_i \varepsilon_{1-i} - \beta_1 \hat{\varepsilon}_0 \,,$$
$$\vdots$$
$$\hat{\varepsilon}_n = x_n - \sum_{i=1}^{q} \beta_i \hat{\varepsilon}_{n-i} \,.$$

The iterative definition of the $\hat{\varepsilon}_i$'s is a real obstacle to an explicit integration in (1.5) and hinders statistical inference in these models. Note that for $i = -q, -(q-1), \ldots, -1$ the perturbations ε_{-i} can be interpreted as missing data (see Section 5.3.1). ‖

Before the introduction of simulation-based inference, computational difficulties encountered in the modeling of a problem often forced the use of "standard" models and "standard" distributions. One course would be to use models based on *exponential families*, defined below by (1.9) (see Brown 1986, Robert 2001, Lehmann and Casella 1998), which enjoy numerous regularity properties (see Note 1.6.1). Another course was to abandon parametric representations for nonparametric approaches which are, by definition, robust against modeling errors.

We also note that the reduction to simple, perhaps non-realistic, distributions (necessitated by computational limitations) does not necessarily eliminate the issue of nonexplicit expressions, whatever the statistical technique. Our major focus is the application of simulation-based techniques to provide solutions and inference for a more realistic set of models and, hence, circumvent the problems associated with the need for explicit or computationally simple answers.

1.2 Likelihood Methods

The statistical techniques that we will be most concerned with are *maximum likelihood* and *Bayesian* methods, and the inferences that can be drawn from

their use. In their implementation, these approaches are customarily associated with specific mathematical computations, the former with maximization problems—and thus to an *implicit* definition of estimators as solutions of maximization problems— the later with integration problems—and thus to a (formally) *explicit* representation of estimators as an integral. (See Berger 1985, Casella and Berger 2001, Robert 2001, Lehmann and Casella 1998, for an introduction to these techniques.)

The method of maximum likelihood estimation is quite a popular technique for deriving estimators. Starting from an iid sample $\mathbf{x} = (x_1, \ldots, x_n)$ from a population with density $f(x|\theta_1, \ldots, \theta_k)$, the *likelihood function* is

$$\begin{aligned} L(\theta|\mathbf{x}) &= L(\theta_1, \ldots, \theta_k|x_1, \ldots, x_n) \\ &= \prod_{i=1}^n f(x_i|\theta_1, \ldots, \theta_k). \end{aligned} \tag{1.6}$$

More generally, when the X_i's are not iid, the likelihood is defined as the joint density $f(x_1, \ldots, x_n|\theta)$ taken as a function of θ. The value of θ, say $\hat{\theta}$, which is the parameter value at which $L(\theta|\mathbf{x})$ attains its maximum as a function of θ, with $\mathbf{x}$ held fixed, is known as a *maximum likelihood estimator (MLE)*. Notice that, by its construction, the range of the MLE coincides with the range of the parameter. The justifications of the maximum likelihood method are primarily asymptotic, in the sense that the MLE is converging almost surely to the true value of the parameter, under fairly general conditions (see Lehmann and Casella 1998) although it can also be interpreted as being at the fringe of the Bayesian paradigm (see, e.g., Berger and Wolpert 1988).

Example 1.4. Gamma MLE. A maximum likelihood estimator is typically calculated by maximizing the logarithm of the likelihood function (1.6). Suppose $X_1, \ldots, X_n$ are iid observations from the gamma density

$$f(x|\alpha, \beta) = \frac{1}{\Gamma(\alpha)\beta^\alpha} x^{\alpha-1} e^{-x/\beta},$$

where we assume that α is known. The *log likelihood* is

$$\begin{aligned} \log L(\alpha, \beta|x_1, \ldots, x_n) &= \log \prod_{i=1}^n f(x_i|\alpha, \beta) \\ &= \log \prod_{i=1}^n \frac{1}{\Gamma(\alpha)\beta^\alpha} x_i^{\alpha-1} e^{-x_i/\beta} \\ &= -n \log \Gamma(\alpha) - n\alpha \log \beta + (\alpha - 1) \sum_{i=1}^n \log x_i - \sum_{i=1}^n x_i/\beta, \end{aligned}$$

where we use the fact that the log of the product is the sum of the logs, and have done some simplifying algebra. Solving $\frac{\partial}{\partial \beta} \log L(\alpha, \beta|x_1, \ldots, x_n) = 0$ is straightforward and yields the MLE of β, $\hat{\beta} = \sum_{i=1}^n x_i/(n\alpha)$.

Suppose now that α was also unknown, and we additionally had to solve

$$\frac{\partial}{\partial \alpha} \log L(\alpha, \beta|x_1, \ldots, x_n) = 0.$$

This results in a particularly nasty equation, involving some difficult computations (such as the derivative of the gamma function, the *digamma function*). An explicit solution is no longer possible. ‖

Calculation of maximum likelihood estimators can sometimes be implemented through the minimization of a sum of squared residuals, which is the basis of the *method of least squares*.

Example 1.5. Least squares estimators. Estimation by *least squares* can be traced back to Legendre (1805) and Gauss (1810) (see Stigler 1986). In the particular case of linear regression, we observe (x_i, y_i), $i = 1, \ldots, n$, where

$$Y_i = a + bx_i + \varepsilon_i, \quad i = 1, \ldots, n, \tag{1.7}$$

and the variables ε_i's represent errors. The parameter (a, b) is estimated by minimizing the distance

$$\sum_{i=1}^n (y_i - ax_i - b)^2 \tag{1.8}$$

in (a, b), yielding the least squares estimates. If we add more structure to the error term, in particular that $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, independent (equivalently, $Y_i|x_i \sim \mathcal{N}(ax_i + b, \sigma^2)$), the log-likelihood function for (a, b) is proportional to

$$\log(\sigma^{-n}) - \sum_{i=1}^n (y_i - ax_i - b)^2/2\sigma^2,$$

and it follows that the maximum likelihood estimates of a and b are identical to the least squares estimates.

If, in (1.8), we assume $\mathbb{E}(\varepsilon_i) = 0$, or, equivalently, that the linear relationship $\mathbb{E}[Y|x] = ax + b$ holds, minimization of (1.8) is equivalent, from a computational point of view, to imposing a normality assumption on Y conditionally on x and applying maximum likelihood. In this latter case, the additional estimator of σ^2 is consistent if the normal approximation is asymptotically valid. (See Gouriéroux and Monfort 1996, for the related theory of *pseudo-likelihood*.) ‖

Although somewhat obvious, this formal equivalence between the optimization of a function depending on the observations and the maximization of a likelihood associated with the observations has a nontrivial outcome and applies in many other cases. For example, in the case where the parameters

are constrained, Robertson et al. (1988) consider a $p \times q$ table of random variables Y_{ij} with means θ_{ij}, where the means are increasing in i and j. Estimation of the θ_{ij}'s by minimizing the sum of the $(y_{ij} - \theta_{ij})^2$'s is possible through the (numerical) algorithm called *"pool-adjacent-violators,"* developed by Robertson et al. (1988) to solve this specific problem. (See Problems 1.18 and 1.19.) An alternative is to use an algorithm based on simulation and a representation using a normal likelihood (see Section 5.2.4).

In the context of exponential families, that is, distributions with density

$$f(x) = h(x)\, e^{\theta \cdot x - \psi(\theta)}, \qquad \theta, x \in \mathbb{R}^k, \tag{1.9}$$

the approach by maximum likelihood is (formally) straightforward. The maximum likelihood estimator of θ is the solution of

$$x = \nabla\psi\{\hat{\theta}(x)\}\,, \tag{1.10}$$

which also is the equation yielding a method of moments estimator, since $\mathbb{E}_\theta[X] = \nabla\psi(\theta)$. The function ψ is the log-Laplace transform, or *cumulant generating function* of h; that is, $\psi(t) = \log \mathbb{E}[\exp\{th(X)\}]$, where we recognize the right side as the log *moment generating function* of h.

Example 1.6. Normal MLE. In the setup of the normal $\mathcal{N}(\mu, \sigma^2)$ distribution, the density can be written as in (1.9), since

$$\begin{aligned} f(y|\mu,\sigma) &\propto \sigma^{-1} e^{-(\mu-y)^2/2\sigma^2} \\ &= \sigma^{-1} e^{(\mu/\sigma^2)\,y - (1/2\sigma^2)\,y^2 - \mu^2/2\sigma^2}\,. \end{aligned}$$

The so-called *natural parameters* are then $\theta_1 = \mu/\sigma^2$ and $\theta_2 = -1/2\sigma^2$, with $\psi(\theta) = -\theta_1^2/4\theta_2 + \log(-\theta_2/2)/2$. While there is no MLE for a single observation from $\mathcal{N}(\mu, \sigma^2)$, equation (1.10) leads to

$$\begin{aligned} -n\tfrac{\theta_1}{2\theta_2} &= \textstyle\sum_{i_1}^n y_i = \quad n\bar{y}\,, \\ n\tfrac{\theta_1^2}{4\theta_2^2} - \tfrac{n}{2\theta_2} &= \textstyle\sum_{i_1}^n y_i^2 = n(s^2 + \bar{y}^2)\,, \end{aligned} \tag{1.11}$$

in the case of n iid observations $y_1, \ldots, y_n$, that is, to the regular MLE, $(\hat{\mu}, \hat{\sigma}^2) = (\bar{y}, s^2)$, where $s^2 = \sum(y_i - \bar{y})^2/n$. ‖

Unfortunately, there are many settings where ψ cannot be computed explicitly. Even if it could be done, it may still be the case that the solution of (1.10) is not explicit, or there are constraints on θ such that the maximum of (1.9) is not a solution of (1.10). This last situation occurs in the estimation of the table of θ_{ij}'s in the discussion above.

Example 1.7. Beta MLE. The Beta $\mathcal{B}e(\alpha, \beta)$ distribution is a particular case of exponential family since its density,

$$f(y|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\,\Gamma(\beta)}\, y^{\alpha-1}(1-y)^{\beta-1}\,, \qquad 0 \le y \le 1,$$

can be written as (1.9), with $\theta = (\alpha, \beta)$ and $x = (\log y, \log(1-y))$. Equation (1.10) becomes

$$\begin{aligned} \log y \quad &= \Psi(\alpha) - \Psi(\alpha+\beta)\,, \\ \log(1-y) &= \Psi(\beta) - \Psi(\alpha+\beta)\,, \end{aligned} \tag{1.12}$$

where $\Psi(z) = d\log\Gamma(z)/dz$ denotes the *digamma function* (see Abramowitz and Stegun 1964). There is no explicit solution to (1.12). As in Example 1.6, although it may seem absurd to estimate both parameters of the $\mathcal{B}e(\alpha, \beta)$ distribution from a single observation, Y, the formal computing problem at the core of this example remains valid for a sample $Y_1, \ldots, Y_n$ since (1.12) is then replaced by

$$\begin{aligned} \frac{1}{n}\sum_i \log y_i &= \Psi(\alpha) - \Psi(\alpha+\beta)\,, \\ \frac{1}{n}\sum_i \log(1-y_i) &= \Psi(\beta) - \Psi(\alpha+\beta)\,. \end{aligned}$$

‖

When the parameter of interest λ is not a one-to-one function of θ, that is, when there are *nuisance* parameters, the maximum likelihood estimator of λ is still well defined. If the parameter vector is of the form $\theta = (\lambda, \psi)$, where ψ is a nuisance parameter, a typical approach is to calculate the full MLE $\hat{\theta} = (\hat{\lambda}, \hat{\psi})$ and use the resulting $\hat{\lambda}$ to estimate λ. In principle, this does not require more complex calculations, although the distribution of the maximum likelihood estimator of λ, $\hat{\lambda}$, may be quite involved. Many other options exist, such as *conditional, marginal, or profile* likelihood (see, for example, Barndorff-Nielsen and Cox 1994).

Example 1.8. Noncentrality parameter. If $X \sim \mathcal{N}_p(\theta, I_p)$ and if $\lambda = \|\theta\|^2$ is the parameter of interest, the nuisance parameters are the angles $\boldsymbol{\Psi}$ in the polar representation of θ and the maximum likelihood estimator of λ is $\hat{\lambda}(x) = \|x\|^2$, which has a constant bias equal to p. Surprisingly, an observation $Y = \|X\|^2$ which has a noncentral chi squared distribution, $\chi^2_p(\lambda)$ (see Appendix A), leads to a maximum likelihood estimator of λ which differs[6] from Y, since it is the solution of the implicit equation

$$\sqrt{\lambda}\, I_{(p-1)/2}\left(\sqrt{\lambda y}\right) = \sqrt{y}\, I_{p/2}\left(\sqrt{\lambda y}\right)\,, \qquad y > p\,, \tag{1.13}$$

where I_ν is the *modified Bessel function*

[6] This phenomenon is not paradoxical, as $Y = \|X\|^2$ is not a sufficient statistic in the original problem.

$$I_\nu(t) = \frac{(z/2)^\nu}{\sqrt{\pi}\Gamma(\nu+\frac{1}{2})}\int_0^\pi e^{t\cos(\theta)}\sin^{2\nu}(\theta)d\theta$$
$$= \left(\frac{t}{2}\right)^\nu \sum_{k=0}^{\infty}\frac{(z/2)^{2k}}{k!\Gamma(\nu+k+1)}.$$

So even in the favorable context of exponential families, we are not necessarily free from computational problems, since the resolution of (1.13) requires us first to evaluate the special functions $I_{p/2}$ and $I_{(p-1)/2}$. Note also that the maximum likelihood estimator is not a solution of (1.13) when $y < p$ (see Problem 1.20). ‖

When we leave the exponential family setup, we face increasingly challenging difficulties in using maximum likelihood techniques. One reason for this is the lack of a *sufficient statistic of fixed dimension* outside exponential families, barring the exception of a few families such as uniform or Pareto distributions whose support depends on θ (Robert 2001, Section 3.2). This result, known as the *Pitman–Koopman Lemma* (see Lehmann and Casella 1998, Theorem 1.6.18), implies that, outside exponential families, the complexity of the likelihood increases quite rapidly with the number of observations, n and, thus, that its maximization is delicate, even in the simplest cases.

Example 1.9. Student's t distribution. Modeling random perturbations using normally distributed errors is often (correctly) criticized as being too restrictive and a reasonable alternative is the Student's t distribution, denoted by $\mathcal{T}(p,\theta,\sigma)$, which is often more "robust" against possible modeling errors (and others). The density of $\mathcal{T}(p,\theta,\sigma)$ is proportional to

$$\sigma^{-1}\left(1+\frac{(x-\theta)^2}{p\sigma^2}\right)^{-(p+1)/2}. \tag{1.14}$$

Typically, p is known and the parameters θ and σ are unknown. Based on an iid sample $(X_1,\ldots,X_n)$ from (1.14), the likelihood is proportional to a power of the product

$$\sigma^{n\frac{p+1}{2}}\prod_{i=1}^{n}\left(1+\frac{(x_i-\theta)^2}{p\sigma^2}\right).$$

When σ is known, for some configurations of the sample, this polynomial of degree $2n$ may have n local minima, each of which needs to be calculated to determine the global maximum, the maximum likelihood estimator (see also Problem 1.14). Figure 1.1 illustrates this multiplicity of modes of the likelihood from a Cauchy distribution $\mathcal{C}(\theta,1)$ ($p=1$) when $n=3$ and $X_1=0$, $X_2=5$, and $X_3=9$. ‖

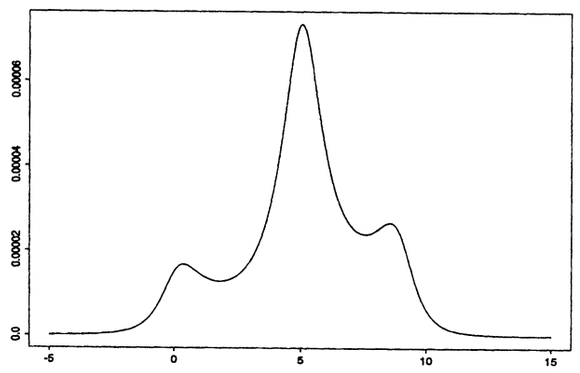


Fig. 1.1. Likelihood of the sample $(0,5,9)$ from the distribution $\mathcal{C}(\theta,1)$.

Example 1.10. (Continuation of Example 1.2) In the special case of a mixture of two normal distributions,

$$p\mathcal{N}(\mu,\tau^2)+(1-p)\mathcal{N}(\theta,\sigma^2),$$

an iid sample $(X_1,\ldots,X_n)$ results in a likelihood function proportional to

$$\prod_{i=1}^{n}\left[p\tau^{-1}\varphi\left(\frac{x_i-\mu}{\tau}\right)+(1-p)\,\sigma^{-1}\,\varphi\left(\frac{x_i-\theta}{\sigma}\right)\right] \tag{1.15}$$

containing 2^n terms if expanded. Standard maximization techniques often fail to find the global maximum because of multimodality of the likelihood function, and specific algorithms must be devised (to obtain the global maximum with high probability).

The problem is actually another order of magnitude more difficult, since the likelihood is unbounded here. The expansion of the product (1.15) contains the terms

$$p^n\tau^{-n}\prod_{i=1}^{n}\varphi\left(\frac{x_i-\mu}{\tau}\right)$$
$$+p^{n-1}(1-p)\,\tau^{-n+1}\sigma^{-1}\,\varphi\left(\frac{x_1-\theta}{\sigma}\right)\prod_{i=2}^{n}\varphi\left(\frac{x_i-\mu}{\tau}\right)+\ldots.$$

This expression is unbounded in σ (let σ go to 0 when $\theta=x_1$). However, this difficulty with the likelihood function does not preclude us from using the maximum likelihood approach in this context, since Redner and Walker (1984) have shown that there exist solutions to the *likelihood equations*, that is, local maxima of (1.15), which have acceptable properties. (Similar problems occur in the context of linear regression with "errors in variables." See Casella and Berger 2001, Chapter 12, for an introduction.) ‖

In addition to the difficulties associated with optimization problems, likelihood-related approaches may also face settings where the likelihood function is only expressible as an integral (for example, the censored data models of Example 1.1). Similar computational problems arise in the determination of the *power* of a testing procedure in the Neyman–Pearson approach (see Lehmann 1986, Casella and Berger 2001, Robert 2001)

For example, inference based on a likelihood ratio statistic requires computation of quantities such as

$$P_\theta(L(\theta|X)/L(\theta_0|X) \le k)\,,$$

with fixed θ_0 and k, where $L(\theta|x)$ represents the likelihood based on observing $X = x$. Outside of the more standard (simple) settings, this probability cannot be explicitly computed because dealing with the distribution of test statistics under the alternative hypothesis may be quite difficult. A particularly delicate case is the *Behrens–Fisher problem*, where the above probability is difficult to evaluate even under the null hypothesis (see Lehmann 1986, Lee 2004). (Note that likelihood ratio tests cannot be rigorously classified as a likelihood-related approach, since they violate the *Likelihood Principle*, see Berger and Wolpert 1988, but the latter does not provide a testing theory per se.)

1.3 Bayesian Methods

Whereas the difficulties related to maximum likelihood methods are mainly *optimization* problems (multiple modes, solution of likelihood equations, links between likelihood equations and global modes, etc.), the Bayesian approach more often results in *integration* problems. In the Bayesian paradigm, information brought by the data x, a realization of $X \sim f(x|\theta)$, is combined with prior information that is specified in a *prior distribution* with density $\pi(\theta)$ and summarized in a probability distribution, $\pi(\theta|x)$, called the *posterior distribution*. This is derived from the *joint* distribution $f(x|\theta)\pi(\theta)$, according to *Bayes formula*

$$\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int f(x|\theta)\pi(\theta)d\theta}, \tag{1.16}$$

where $m(x) = \int f(x|\theta)\pi(\theta)d\theta$ is the *marginal density* of X (see Berger 1985, Bernardo and Smith 1994, Robert 2001, for more details, in particular about the philosophical foundations of this inferential approach).

For the estimation of a particular parameter $h(\theta)$, the decision-theoretic approach to statistical inference (see, e.g. Berger 1985) requires the specification of a loss function $\mathrm{L}(\delta, \theta)$, which represents the loss incurred by estimating $h(\theta)$ with δ. The Bayesian version of this approach leads to the minimization of the *Bayes risk*,

$$\int\int \mathrm{L}(\delta,\theta) f(x|\theta)\pi(\theta)dxd\theta\,,$$

that is, the loss integrated against both X and θ. A straightforward inversion of the order of integration (Fubini's theorem) leads to choosing the estimator δ that minimizes (for each x) the *posterior loss*,

$$\mathbb{E}[\mathrm{L}(\delta,\theta)|x] = \int \mathrm{L}(\delta,\theta)\,\pi(\theta|x)\,d\theta\,. \tag{1.17}$$

In the particular case of the quadratic loss

$$\mathrm{L}(\delta,\theta) = \|h(\theta) - \delta\|^2\,,$$

the Bayes estimator of $h(\theta)$ is $\delta^\pi(x) = \mathbb{E}^\pi[h(\theta)|x]$. (See Problem 1.22.)

Some of the difficulties related to the computation of $\delta^\pi(x)$ are, first, that $\pi(\theta|x)$ is not generally available in closed form and, second, that in many cases the integration of $h(\theta)$ according to $\pi(\theta|x)$ cannot be done analytically. Loss functions $\mathrm{L}(\delta,\theta)$ other than the quadratic loss function are usually even more difficult to deal with.

The computational drawback of the Bayesian approach has been so great that, for a long time, the favored types of priors in a Bayesian modeling were those allowing explicit computations, namely *conjugate priors*. These are prior distributions for which the corresponding posterior distributions are themselves members of the original prior family, the Bayesian updating being accomplished through updating of parameters. (See Note 1.6.1 and Robert 2001, Chapter 3, for a discussion of the link between conjugate priors and exponential families.)

Example 1.11. Binomial Bayes estimator. For an observation X from the binomial distribution $\mathcal{B}(n,p)$, a family of conjugate priors is the family of Beta distributions $\mathcal{B}e(a,b)$. To find the Bayes estimator of p under squared error loss, we can find the minimizer of the Bayes risk, that is,

$$\min_\delta \int_0^1 \sum_{x=1}^{n} [p-\delta(x)]^2 \binom{n}{x} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} p^{x+a-1}(1-p)^{n-x+b-1}dp.$$

Equivalently, we can work with the posterior expected loss (1.17) and find the estimator that yields

$$\min_\delta \frac{\Gamma(a+b+n)}{\Gamma(a+x)\Gamma(n-x+b)} \int_0^1 [p-\delta(x)]^2 p^{x+a-1}(1-p)^{n-x+b-1}dp\,,$$

where we note that the posterior distribution of p (given x) is $\mathcal{B}e(x+a, n-x+b)$. The solution is easily obtained through differentiation, and the Bayes estimator δ^π is the posterior mean

$$\delta^\pi(x) = \frac{\Gamma(a+b+n)}{\Gamma(a+x)\Gamma(n-x+b)} \int_0^1 p\, p^{x+a-1}(1-p)^{n-x+b-1}dp = \frac{x+a}{a+b+n}.$$

The use of squared error loss results in the Bayes estimator being the mean of the posterior distribution, which usually simplifies calculations. If, instead, we had specified a *absolute error loss* $|p-\delta(x)|$ or had used a nonconjugate prior, the calculations could have become somewhat more involved (see Problem 1.22). ‖

The use of conjugate priors for computational reasons implies a restriction on the modeling of the available prior information and may be detrimental to the usefulness of the Bayesian approach as a method of statistical inference. This is because it perpetuates an impression both of subjective "manipulation" of the background (prior information) and of formal expansions unrelated to reality. The considerable advances of Bayesian decision theory have often highlighted the negative features of modeling using only conjugate priors. For example, Bayes estimators are the optimal estimators for the three main classes of optimality (admissibility, minimaxity, invariance), but those based on conjugate priors only partially enjoy these properties (see Berger 1985, Section 4.7, or Robert 2001, Chapter 8) .

Example 1.12. (Continuation of Example 1.8). For the estimation of $\lambda = \|\theta\|^2$, a *reference prior*[7] on θ is $\pi(\theta) = \|\theta\|^{-(p-1)}$ (see Berger et al. 1998), with corresponding posterior distribution

$$\pi(\theta|x) \propto \frac{e^{-\|x-\theta\|^2/2}}{\|\theta\|^{p-1}} . \tag{1.18}$$

The normalizing constant corresponding to $\pi(\theta|x)$ is not easily obtainable and the Bayes estimator of λ, the posterior mean

$$\frac{\int_{\mathbb{R}^p} \|\theta\|^{2-p}\, e^{-\|x-\theta\|^2/2}\, d\theta}{\int_{\mathbb{R}^p} \|\theta\|^{1-p}\, e^{-\|x-\theta\|^2/2}\, d\theta} , \tag{1.19}$$

cannot be explicitly computed. (See Problem 1.20.) ‖

The computation of the normalizing constant of $\pi(\theta|x)$ is not just a formality. Although the derivation of a posterior distribution is generally done through proportionality relations, that is, using *Bayes Theorem* in the form

$$\pi(\theta|x) \propto \pi(\theta)\, f(x|\theta) ,$$

it is sometimes necessary to know the posterior distribution or, equivalently, the marginal distribution, exactly. For example, this is the case in the Bayesian

[7] A reference prior is a prior distribution which is derived from maximizing a distance measure between the prior and the posterior distributions. When there is no nuisance parameter in the model, the standard reference prior is Jeffreys (1961) prior (see Bernardo and Smith 1994, Robert 2001, and Note 1.6.1.).

comparison of (statistical) models. If $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_k$ are possible models for the observation X, with densities $f_j(\cdot|\theta_j)$, if the associated parameters $\theta_1, \theta_2, \ldots, \theta_k$ are *a priori* distributed from $\pi_1, \pi_2, \ldots, \pi_k$, and if these models have the prior weights $p_1, p_2, \ldots, p_k$, the posterior probability that X originates from model $\mathcal{M}_j$ is (Problem 1.21)

$$\frac{p_j \int f_j(x|\theta_j)\, \pi_j(\theta_j)\, d\theta_j}{\sum_{i=1}^k p_i \int f_i(x|\theta_i)\, \pi_i(\theta_i)\, d\theta_i} . \tag{1.20}$$

In particular, the comparison of two models $\mathcal{M}_1$ and $\mathcal{M}_2$ is often implemented through the *Bayes factor*

$$B^\pi(x) = \frac{\int f_1(x|\theta_1)\, \pi_1(\theta_1)\, d\theta_1}{\int f_2(x|\theta_2)\, \pi_2(\theta_2)\, d\theta_2} ,$$

for which the proportionality constant is quite important (see Kass and Raftery 1995 and Goutis and Robert 1998 for different perspectives on Bayes factors). Unsurprisingly, there has been a lot of research in the computation of these normalizing constants (Gelman and Meng 1998, Kong et al. 2003).

Example 1.13. Logistic regression. A useful regression model for binary $(0-1)$ responses is the *logit model*, where the distribution of Y conditional on the explanatory (or dependent) variables $x \in \mathbb{R}^p$ is modeled by the relation

$$P(Y=1) = p = \frac{\exp(\alpha + x\beta)}{1+\exp(\alpha + x\beta)}. \tag{1.21}$$

Equivalently, the *logit* transform of p, $\text{logit}(p) = \log[p/(1-p)]$, satisfies the linear relationship $\text{logit}(p) = \alpha + x\beta$.

In 1986, the space shuttle Challenger exploded during take off, killing the seven astronauts aboard. The explosion was the result of an *O-ring* failure, a splitting of a ring of rubber that seals the parts of the ship together. The accident was believed to be caused by the unusually cold weather (31° F or 0° C) at the time of launch, as there is reason to believe that the O-ring failure probabilities increase as temperature decreases (Dalal et al. 1989).

Flight	14	9	23	10	1	5	13	15	4	3	8	17	2	11	6	7	16	21	19	22	12	20	18
Failure	1	1	1	1	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0
Temp.	53	57	58	63	66	67	67	67	68	69	70	70	70	70	72	73	75	75	76	76	78	79	81

Table 1.1. Temperature at flight time (degrees F) and failure of O-rings *(1 stands for failure, 0 for success).*

Data on previous space shuttle launches, and O-ring failures, is given in Table 1.1. It is reasonable to fit a logistic regression, as in (1.21) with $p =$

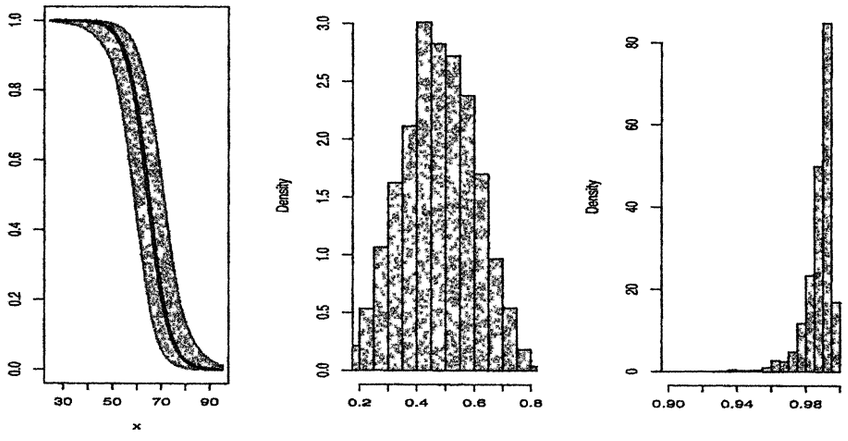


Fig. 1.2. The figure shows the result of 10,000 Monte Carlo simulations of the model (1.21). The left panel shows the average logistic function and variation, the middle panel shows predictions of failure probabilities at 65° Fahrenheit, and the right panel shows predictions of failure probabilities at 45° Fahrenheit.

probability of an O-ring failure and x = temperature. The results are shown in Figure 1.2 for an exponential prior on $\log \alpha$ and a flat prior on β.

The left panel in Figure 1.2 shows the logistic regression line, and the grey curves represent the results of a Monte Carlo simulation explained in Example 7.3 from the posterior distribution of the model showing the variability in the data. It is clear that the ends of the function have little variability, while there is some in the middle. However, the next two panels are most important, as they show the variability in failure probability predictions at two different temperatures. The middle panel, which gives the failure probability at 65° Fahrenheit, shows that, at this temperature, a failure is just about as likely as a success. However, at 45° Fahrenheit, the failures are strongly skewed toward 1. Given this trend, imagine what the failure probabilities look like at 31° Fahrenheit, the temperature at Challenger launch time: At that temperature, failure was almost a certainty.

The logistic regression Monte Carlo analysis of this data is quite straightforward, and gives easy-to-understand answers to the relevant questions. Non-Monte Carlo alternatives would typically be based on likelihood theory and asymptotics, and would be more difficult to implement and interpret. ‖

The computational problems encountered in the Bayesian approach are not limited to the computation of integrals or normalizing constants. For instance, the determination of confidence regions (also called *credible regions*) with *highest posterior density*,

$$C^\pi(x) = \{\theta; \pi(\theta|x) \geq k\} ,$$

requires the solution of the equation $\pi(\theta|x) = k$ for the value of k that satisfies

$$P(\theta \in C^\pi(x)|x) = P(\pi(\theta|x) \geq k|x) = \gamma ,$$

where γ is a predetermined confidence level.

Example 1.14. Bayes credible regions. For iid observations $X_1, \ldots, X_n$ from a normal distribution $\mathcal{N}(\theta, \sigma^2)$, and a prior distribution $\theta \sim \mathcal{N}(0, \tau^2)$, the posterior density of θ is normal with mean and variance

$$\delta^\pi = \frac{n\tau^2}{n\tau^2 + \sigma^2}\bar{x} \qquad \text{and} \qquad \frac{n\tau^2\sigma^2}{n\tau^2 + \sigma^2} ,$$

respectively. If we assume that σ^2 and τ^2 are known, the highest posterior density region is

$$\left\{\theta : \sqrt{\frac{n\tau^2 + \sigma^2}{2\pi n\tau^2}} \exp\left[-\frac{n\tau^2 + \sigma^2}{2n\tau^2}(\theta - \delta^\pi)^2\right] \geq k\right\} .$$

Since the posterior distribution is symmetric and unimodal, this set is equivalent to (Problem 1.24)

$$\{\theta;\ \delta^\pi - k' \leq \theta \leq \delta^\pi + k'\} ,$$

for some constant k' that is chosen to yield a specified posterior probability. Since the posterior distribution of θ is normal, this can be done by hand using a normal probability table.

For the situation of Example 1.11, the posterior distribution of p, $\pi(p|x, a, b)$, was found to be $\mathcal{B}e(x+a, n-x+b)$, which is not necessarily symmetric. To find the 90% highest posterior density region for p, we must find limits $l(x)$ and $u(x)$ that satisfy

$$\int_{l(x)}^{u(x)} \pi(p|x, a, b)dp = .9 \qquad \text{and} \qquad \pi(l(x)|x, a, b) = \pi(u(x)|x, a, b).$$

This cannot be solved analytically. ‖

Computation of a confidence region can be quite delicate when $\pi(\theta|x)$ is not explicit. In particular, when the confidence region involves only one component of a vector parameter, calculation of $\pi(\theta|x)$ requires the integration of the joint distribution over all the other parameters. Note that knowledge of the normalizing factor is of minor importance in this setup. (See Robert 2001, Chapter 6, for other examples.)

Example 1.15. Cauchy confidence regions. Consider $X_1, \ldots, X_n$, an iid sample from the Cauchy distribution $\mathcal{C}(\theta, \sigma)$, with associated prior distribution $\pi(\theta, \sigma) = \sigma^{-1}$. The confidence region on θ is then based on

$$\pi(\theta|x_1,\ldots,x_n) \propto \int_0^\infty \sigma^{-n-1} \prod_{i=1}^n \left[1+\left(\frac{x_i-\theta}{\sigma}\right)^2\right]^{-1} d\sigma\,,$$

an integral which cannot be evaluated explicitly. Similar computational problems occur with likelihood estimation in this model. One method for obtaining a likelihood confidence interval for θ is to use the *profile likelihood*

$$\ell^P(\theta|x_1,\ldots,x_n) = \max_\sigma \ell(\theta,\sigma|x_1,\ldots,x_n)$$

and consider the region $\{\theta : \ell^P(\theta|x_1,\ldots,x_n) \geq k\}$. Explicit computation is also difficult here. ‖

Example 1.16. Linear calibration. In a standard regression model, $Y = \alpha + \beta x + \varepsilon$, there is interest in estimating or predicting features of Y from knowledge of x. In *linear calibration models* (see Osborne 1991, for an introduction and review of these models), the interest is in determining values of x from observed responses y. For example, in a chemical experiment, one may want to relate the precise but expensive measure y to the less precise but inexpensive measure x. A simplified version of this problem can be put into the framework of observing the independent random variables

$$Y \sim \mathcal{N}_p(\beta,\sigma^2 I_p),\ Z \sim \mathcal{N}_p(x_0\beta,\sigma^2 I_p),\ S \sim \sigma^2\chi^2_q\,,$$

with $x_0 \in \mathbb{R}$, $\beta \in \mathbb{R}^p$. The parameter of interest is now x_0 and this problem is equivalent to Fieller (1954) problem (see, e.g Lehmann and Casella 1998).

A reference prior on (x_0,β,σ) is given in Kubokawa and Robert (1994), and yields the joint posterior distribution

$$\pi(x_0,\beta,\sigma^2|y,z,s) \propto \sigma^{-(3p+q)-\frac{1}{2}} \exp\{-(s+\|y-\beta\|^2 + \|z-x_0\beta\|^2)/2\sigma^2\}\,(1+x_0^2)^{-1/2}\,. \tag{1.22}$$

This can be analytically integrated to obtain the marginal posterior distribution of X_0 to be

$$\pi(x_0|y,z,s) \propto \frac{(1+x_0^2)^{(p+q-1)/2}}{\left\{\left(x_0-\frac{y^tz}{s+\|y\|^2}\right)^2+\frac{\|z\|^2+s}{\|y\|^2+s}-\frac{(y^tz)^2}{(s+\|y\|^2)^2}\right\}^{(2p+q)/2}}\,. \tag{1.23}$$

However, the computation of the posterior mean, the Bayes estimate of x_0, is not feasible analytically; neither is the determination of the confidence region $\{\pi(x_0|\mathcal{D}) \geq k\}$. Nonetheless, it is desirable to determine this confidence region since alternative solutions, for example the *Fieller–Creasy* interval, suffer from defects such as having infinite length with positive probability (see Gleser and Hwang 1987, Casella and Berger 2001, Ghosh et al. 1995, Philippe and Robert 1998b). ‖

1.4 Deterministic Numerical Methods

The previous examples illustrated the need for techniques, in both the construction of complex models and estimation of parameters, that go beyond the standard analytical approaches. However, before starting to describe simulation methods, which is the purpose of this book, we should recall that there exists a well-developed alternative approach for integration and optimization, based on *numerical* methods. We refer the reader to classical textbooks on numerical analysis (see, for instance, Fletcher 1980 or Evans 1993 for a description of these methods, which are generally efficient and can deal with most of the above examples (see also Lange 1999 or Gentle 2002 for presentations in statistical settings).

1.4.1 Optimization

We briefly recall here the more standard approaches to optimization and integration problems, both for comparison purposes and for future use. When the goal is to solve an equation of the form $f(x) = 0$, a common approach is to use a *Newton–Raphson* algorithm, which produces a sequence x_n such that

$$x_{n+1} = x_n - \left(\left.\frac{\partial f}{\partial x}\right|_{x=x_n}\right)^{-1} f(x_n) \tag{1.24}$$

until it stabilizes around a solution of $f(x) = 0$. (Note that $\frac{\partial f}{\partial x}$ is a matrix in multidimensional settings.) Optimization problems associated with smooth functions F are then based on this technique, using the equation $\nabla F(x) = 0$, where ∇F denotes the *gradient* of F, that is, the vector of derivatives of F. (When the optimization involves a constraint $G(x) = 0$, F is replaced by a Lagrangian form $F(x) - \lambda G(x)$, where λ is used to satisfy the constraint.) The corresponding techniques are then the *gradient methods*, where the sequence x_n is such that

$$x_{n+1} = x_n - \left(\nabla\nabla^t F\right)^{-1}(x_n)\nabla F(x_n)\,, \tag{1.25}$$

where $\nabla\nabla^t F$ denotes the matrix of second derivatives of F.

Example 1.17. A simple Newton–Raphson Algorithm. As a simple illustration, we show how the Newton–Raphson algorithm can be used to find the square root of a number. If we are interested in the square root of b, this is equivalent to solving the equation

$$f(x) = x^2 - b = 0.$$

Applying (1.24) results in the iterations

$$x^{(j+1)} = x^{(j)} - \frac{f(x^{(j)})}{f'(x^{(j)})} = x^{(j)} - \frac{x^{(j)2}-b}{2x^{(j)}} = \frac{1}{2}(x^{(j)} + \frac{b}{x^{(j)}}).$$

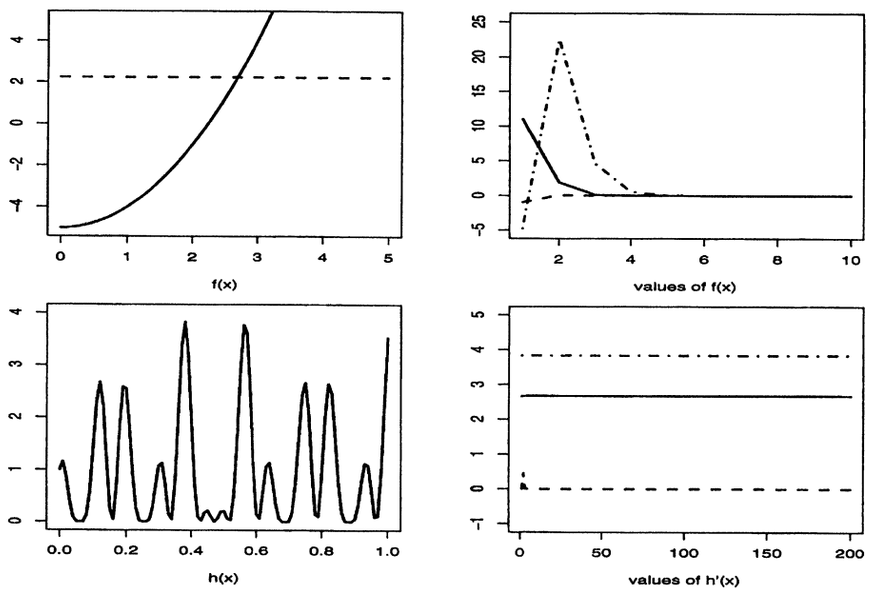


Fig. 1.3. Calculation of the root of $f(x) = 0$ and $h'(x) = 0$ for the functions defined in Example 1.17. The top left panel is $f(x)$, and the top right panel shows that from different starting points the Newton–Raphson algorithm converges rapidly to the square root. The bottom left panel is $h(x)$, and the bottom right panel shows that the Newton–Raphson algorithm cannot find the maximum of $h(x)$, but rather converges to whatever mode is closest to the starting point.

Figure 1.3 shows that the algorithm converges rapidly to the correct answer from different starting points (for $b = 2$, three runs are shown, starting at $x = .5, 2, 4$).

However, if we consider the function

$$h(x) = [\cos(50x) + \sin(20x)]^2, \tag{1.26}$$

and try to derive its maximum, we run into problems. The "greediness" of the Newton–Raphson algorithm, that is, the fact that it always goes toward the nearest mode, does not allow it to escape from local modes. The bottom two panels in Figure 1.3 show the function, and the convergence of the algorithm from three different starting points, $x = .25, .379, .75$ (the maximum occurs at $x = .379$). From the figure we see that wherever we start the algorithm, it goes to the closest mode, which is often not the maximum. In Chapter 5, we will compare this solution to those of Examples 5.2 and 5.5 where the (global) maximum is obtained. ‖

Numerous variants of Newton–Raphson-type techniques can be found in the literature, among which one can mention the *steepest descent* method,

where each iteration results in a unidimensional optimizing problem for $F(x_n + td_n)$ $(t \in \mathbb{R})$, d_n being an acceptable direction, namely such that

$$\left.\frac{d^2F}{dt^2}(x_n + td_n)\right|_{t=0}$$

is of the proper sign. The direction d_n is often chosen as ∇F or as the smoothed version of (1.25),

$$\left[\nabla\nabla^t F(x_n) + \lambda I\right]^{-1} \nabla F(x_n),$$

in the *Levenberg–Marquardt version*. Other versions are available that do not require differentiation of the function F.

1.4.2 Integration

Turning to integration, the numerical computation of an integral

$$\mathfrak{I} = \int_a^b h(x)dx$$

can be done by simple *Riemann integration* (see Section 4.3), or by improved techniques such as the *trapezoidal rule*

$$\hat{\mathfrak{I}} = \frac{1}{2}\sum_{i=1}^{n-1}(x_{i+1} - x_i)(h(x_i) + h(x_{i+1}))\,,$$

where the x_i's constitute an ordered partition of $[a, b]$, or yet *Simpson's rule*, whose formula is

$$\tilde{\mathfrak{I}} = \frac{\delta}{3}\left\{f(a) + 4\sum_{i=1}^{n} h(x_{2i-1}) + 2\sum_{i=1}^{n} h(x_{2i}) + f(b)\right\}$$

in the case of equally spaced samples with $(x_{i+1} - x_i) = \delta$. Other approaches involve orthogonal polynomials (Gram–Charlier, Legendre, etc.), as illustrated by Naylor and Smith (1982) for statistical problems, or splines (see Wahba 1981, for a statistical connection). See also Note 2.6.2 for the *quasi–Monte Carlo* methods that, despite their name, pertain more to numerical integration than to simulation, since they are totally deterministic. However, due to the *curse of dimensionality*, these methods may not work well in high dimensions, as stressed by Thisted (1988).

1.4.3 Comparison

Comparison between the approaches, simulation versus numerical analysis, is delicate because both approaches can provide well-suited tools for many problems (possibly needing a preliminary study) and the distinction between

these two groups of techniques can be vague. So, rather than addressing the issue of a general comparison, we will focus on the requirements of each approach and the objective conditions for their implementation in a statistical setup.

By nature, standard numerical methods do not take into account the probabilistic aspects of the problem; that is, the fact that many of the functions involved in the computations are related to probability densities. Therefore, a numerical integration method may consider regions of a space which have zero (or low) probability under the distribution of the model, a phenomenon which usually does not appear in a simulation experiment.[8] Similarly, the occurrence of local modes of a likelihood will often cause more problems for a deterministic gradient method than for a simulation method that explores high-density regions. (But multimodality must first be identified for these efficient methods to apply, as in Oh 1989 or Oh and Berger 1993.)

On the other hand, simulation methods very rarely take into account the specific analytical form of the functions involved in the integration or optimization, while numerical methods often use higher derivatives to provide bounds on the error of approximation. For instance, because of the randomness induced by the simulation, a gradient method yields a much faster determination of the mode of a unimodal density. For small dimensions, integration by Riemann sums or by quadrature converges faster than the mean of a simulated sample. Moreover, existing scientific software (for instance, `Gauss`, `Maple`, `Mathematica`, `Matlab`, `R`) and scientific libraries like `IMSL` often provide highly efficient numerical procedures, whereas simulation is, at best, implemented through pseudo-random generators for the more common distributions. However, software like `BUGS` (see Note 10.6.2) are progressively bridging the gap.

Therefore, it is often reasonable to use a numerical approach when dealing with regular functions in small dimensions and in a given single problem. On the other hand, when the statistician needs to study the details of a likelihood surface or posterior distribution, or needs to simultaneously estimate several features of these functions, or when the distributions are highly multimodal (see Examples 1.2 and 1.8), it is preferable to use a simulation-based approach. Such an approach captures (if only approximately through the generated sample) the different characteristics of the density and thus allows, at little cost, extensions of the inferential scope to, perhaps, another test or estimator.

However, given the dependence on specific problem characteristics, it is fruitless to advocate the superiority of one method over the other, say of the simulation-based approach over numerical methods. Rather, it seems more reasonable to justify the use of simulation-based methods by the statistician in terms of *expertise*. The intuition acquired by a statistician in his or her everyday processing of random models can be directly exploited in the implementation of simulation techniques (in particular in the evaluation of the variation of the proposed estimators or of the stationarity of the resulting output), while purely numerical techniques rely on less familiar branches of mathematics. Finally, note that many desirable approaches are those which efficiently combine both perspectives, as in the case of *simulated annealing* (see Section 5.2.3) or Riemann sums (see Section 4.3).

[8] Simulation methods using a distribution other than the distribution of interest, such as importance sampling (Section 3.3) or Metropolis–Hastings algorithms (Chapter 7), may suffer from such a drawback.

1.5 Problems

1.1 For both the censored data density (1.1) and the mixture of two normal distributions (1.15), plot the probability density function. Use various values for the parameters μ, θ, σ and τ.

1.2 In the situation of Example 1.1, establish that the densities are indeed (1.1) and (1.2).

1.3 In Example 1.1, the distribution of the random variable $Z = \min(X, Y)$ was of interest. Derive the distribution of Z in the following case of *informative censoring*, where $Y \sim \mathcal{N}(\theta, \sigma^2)$ and $X \sim \mathcal{N}(\theta, \theta^2\sigma^2)$. Pay attention to the identifiability issues.

1.4 In Example 1.1, show that the integral

$$\int_\omega^\infty \alpha\beta x^{\alpha-1} e^{-\beta x^\alpha} dx$$

can be explicitly calculated. (*Hint:* Use a change of variables.)

1.5 For the model (1.4), show that the density of $(X_0, \ldots, X_n)$ is given by (1.5).

1.6 In the setup of Example 1.2, derive the moment estimator of the weights $(p_1, \ldots, p_k)$ when the densities f_j are known.

1.7 In the setup of Example 1.6, show that the likelihood equations are given by (1.11) and that their solution is the standard $(\bar{y}, s^2)$ statistic.

1.8 (Titterington et al. 1985) In the case of a mixture of two exponential distributions with parameters 1 and 2,

$$\pi\mathcal{E}xp(1) + (1-\pi)\mathcal{E}xp(2)\,,$$

show that $\mathbb{E}[X^s] = \{\pi + (1-\pi)2^{-s}\}\Gamma(s+1)$. Deduce the best (in s) moment estimator based on $t_s(x) = x^s/\Gamma(s+1)$.

1.9 Give the moment estimator for a mixture of k Poisson distributions, based on $t_s(x) = x(x-1)\cdots(x-s+1)$. (*Note:* Pearson 1915 and Gumbel 1940 proposed partial solutions in this setup. See Titterington et al. 1985, pp. 80–81, for details.)

1.10 In the setting of Example 1.9, plot the likelihood based on observing $(x_1, x_2, x_3) = (0, 5, 9)$ from the Student's t density (1.14) with $p = 1$ and $\sigma = 1$ (which is the standard Cauchy density). Observe the effect on multimodality of adding a fourth observation x_4 when x_4 varies.

1.11 The *Weibull distribution* $\mathcal{W}e(\alpha, c)$ is widely used in engineering and reliability. Its density is given by

$$f(x|\alpha, c) = c\alpha^{-1}(x/\alpha)^{c-1} e^{-(x/\alpha)^c}.$$

2

Random Variable Generation

"Have you any thought," resumed Valentin, "of a tool with which it could be done?"
"Speaking within modern probabilities, I really haven't," said the doctor.
—G.K. Chesterton, *The Innocence of Father Brown*

The methods developed in this book mostly rely on the possibility of producing (with a computer) a supposedly endless flow of random variables (usually iid) for well-known distributions. Such a simulation is, in turn, based on the production of uniform random variables. Although we are not directly concerned with the *mechanics* of producing uniform random variables (see Note 2.6.1), we are concerned with the *statistics* of producing uniform and other random variables.

In this chapter we first consider what statistical properties we want a sequence of simulated uniform random variables to have. Then we look at some basic methodology that can, starting from these simulated uniform random variables, produce random variables from both standard and nonstandard distributions.

2.1 Introduction

Methods of simulation are based on the production of random variables, originally independent random variables, that are distributed according to a distribution f that is not necessarily explicitly known (see, for example, Examples 1.1, 1.2, and 1.3). The type of random variable production is formalized below in the definition of a *pseudo-random number generator*. We first concentrate on the generation of random variables that are uniform on the interval $[0, 1]$, because the uniform distribution $\mathcal{U}_{[0,1]}$ provides the basic probabilistic representation of randomness and also because all other distributions require a sequence of uniform variables to be simulated.

2.1.1 Uniform Simulation

The logical paradox[1] associated with the generation of "random numbers" is the problem of producing a *deterministic* sequence of values in $[0,1]$ which imitates a sequence of *iid* uniform random variables $\mathcal{U}_{[0,1]}$. (Techniques based on the physical imitation of a "random draw" using, for example, the internal clock of the machine have been ruled out. This is because, first, there is no guarantee on the *uniform* nature of numbers thus produced and, second, there is no reproducibility of such samples.) However, we really do not want to enter here into the philosophical debate on the notion of "random," and whether it is, indeed, possible to "reproduce randomness" (see, for example, Chaitin 1982, 1988).

For our purposes, there are methods that use a fully deterministic process to produce a random sequence in the following sense: Having generated $(X_1, \ldots, X_n)$, knowledge of X_n [or of $(X_1, \ldots, X_n)$] imparts no discernible knowledge of the value of X_{n+1} if the transformation function is not available. Of course, given the initial value X_0 and the transformation function, the sample $(X_1, \ldots, X_n)$ is always the same. Thus, the "pseudo-randomness" produced by these techniques is limited since two samples $(X_1, \ldots, X_n)$ and $(Y_1, \ldots, Y_n)$ produced by the algorithm will not be independent, nor identically distributed, nor comparable in any probabilistic sense. This limitation should not be forgotten: The validity of a random number generator is based on a single sample $X_1, \ldots, X_n$ when n tends to $+\infty$ and not on replications $(X_{11}, \ldots, X_{1n}), (X_{21}, \ldots, X_{2n}), \ldots (X_{k1}, \ldots, X_{kn})$, where n is fixed and k tends to infinity. In fact, the distribution of these n-tuples depends only on the manner in which the initial values X_{r1} $(1 \le r \le k)$ were generated.

With these limitations in mind, we can now introduce the following operational definition, which avoids the difficulties of the philosophical distinction between a deterministic algorithm and the reproduction of a random phenomenon.

Definition 2.1. *A uniform pseudo-random number generator* is an algorithm which, starting from an initial value u_0 and a transformation D, produces a sequence $(u_i) = (D^i(u_0))$ of values in $[0,1]$. For all n, the values $(u_1, \ldots, u_n)$ reproduce the behavior of an iid sample $(V_1, \ldots, V_n)$ of uniform random variables when compared through a usual set of tests.

This definition is clearly restricted to *testable* aspects of the random variable generation, which are connected through the deterministic transformation

[1] Von Neumann (1951) summarizes this problem very clearly by writing *"Any one who considers arithmetical methods of reproducing random digits is, of course, in a state of sin. As has been pointed out several times, there is no such thing as a random number—there are only methods of producing random numbers, and a strict arithmetic procedure of course is not such a method."*

$u_i = D(u_{i-1})$. Thus, the validity of the algorithm consists in the verification that the sequence $U_1, \ldots, U_n$ leads to acceptance of the hypothesis

$$\mathrm{H}_0 : U_1, \ldots, U_n \quad \text{are iid} \quad \mathcal{U}_{[0,1]}.$$

The set of tests used is generally of some consequence. There are classical tests of uniformity, such as the Kolmogorov–Smirnov test. Many generators will be deemed adequate under such examination. In addition, and perhaps more importantly, one can use methods of *time series* to determine the degree of correlation between between U_i and $(U_{i-1}, \ldots, U_{i-k})$, by using an ARMA$(p,q)$ model, for instance. One can use nonparametric tests, like those of Lehmann (1975) or Randles and Wolfe (1979), applying them on arbitrary decimals of U_i. Marsaglia[2] has assembled a set of tests called `Die Hard`.

Definition 2.1 is therefore *functional*: An algorithm that generates uniform numbers is acceptable if it is not rejected by a set of tests. This methodology is not without problems, however. Consider, for example, particular applications that might demand a large number of iterations, as the theory of large deviations (Bucklew 1990), or particle physics, where algorithms resistant to standard tests may exhibit fatal faults. In particular, algorithms having hidden periodicities (see below) or which are not uniform for the smaller digits may be difficult to detect. Ferrenberg et al. (1992) show, for instance, that an algorithm of Wolff (1989), reputed to be "good," results in systematic biases in the processing of Ising models (see Example 5.8), due to long-term correlations in the generated sequence.

The notion that a deterministic system can imitate a random phenomenon may also suggest the use of *chaotic* models to create random number generators. These models, which result in complex deterministic structures (see Bergé et al. 1984, Gleick 1987, Ruelle 1987) are based on dynamic systems of the form $X_{n+1} = D(X_n)$ which are very sensitive to the initial condition X_0.

Example 2.2. The logistic function. The logistic function $D_\alpha(x) = \alpha x(1-x)$ produces, for some values of $\alpha \in [3.57, 4.00]$, chaotic configurations. In particular, the value $\alpha = 4.00$ yields a sequence (X_n) in $[0,1]$ that, theoretically, has the same behavior as a sequence of random numbers (or random variables) distributed according to the *arcsine distribution* with density $1/\pi\sqrt{x(1-x)}$. (See Problem 2.4 for another random number generator based on the "tent" function.)

Although the limit distribution (also called the stationary distribution) associated with a dynamic system $X_{n+1} = D(X_n)$ is sometimes defined and known, the chaotic features of the system do not guarantee acceptable behavior (in the probabilistic sense) of the associated generator. Figure 2.1 illustrates the properties of the generator based on the logistic function D_α. The

[2] These tests are now available as a freeware on the site `http://stat.fsu.edu/~geo/diehard.html`

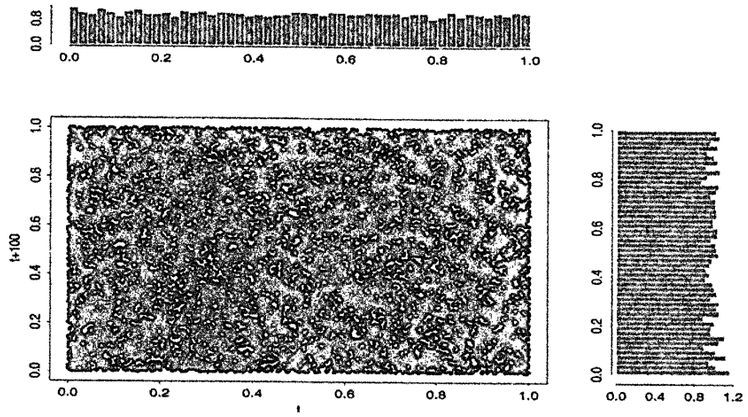


Fig. 2.1. Plot of the sample (y_n, y_{n+100}) $(n = 1, \ldots, 9899)$ for the sequence $x_{n+1} = 4x_n(1 - x_n)$ and $y_n = F(x_n)$, along with the (marginal) histograms of y_n *(on top)* and y_{n+100} *(right margin)*.

histogram of the transformed variables $Y_n = 0.5 + \arcsin(X_n)/\pi$, of a sample of successive values $X_{n+1} = D_\alpha(X_n)$ fits the uniform density extremely well. Moreover, while the plots of (Y_n, Y_{n+1}) and (Y_n, Y_{n+10}) do not display characteristics of uniformity, Figure 2.1 shows that the sample of (Y_n, Y_{n+100}) satisfactorily fills the unit square. However, even when these functions give a good approximation of randomness in the unit square $[0,1] \times [0,1]$, the hypothesis of randomness is rejected by many standard tests.

Classic examples from the theory of chaotic functions do not lead to acceptable pseudo-random number generators. Moreover, the 100 calls to D_α between two generations are excessive in terms of computing time. ‖

We have presented in this introduction some necessary basic notions to now understand a very good pseudo-random number generator, the algorithm Kiss[3] of Marsaglia and Zaman (1993). However, many of the details involve notions that are a bit tangential to the main topic of this text and, in addition, most computer packages now include a well-behaved uniform random generator. Thus, we leave the details of the Kiss generator to Note 2.6.1.

2.1.2 The Inverse Transform

In describing the structure of a space of random variables, it is always possible to represent the generic probability triple $(\Omega, \mathcal{F}, P)$ (where Ω represents the whole space, $\mathcal{F}$ represents a σ-algebra on Ω, and P is a probability measure) as

[3] The name is an acronym of the saying *Keep it simple, stupid!*, and not reflective of more romantic notions. After all, this is a Statistics text!

$([0,1], \mathcal{B}, \mathcal{U}_{[0,1]})$ (where $\mathcal{B}$ are the Borel sets on $[0,1]$) and therefore equate the variability of $\omega \in \Omega$ with that of a uniform variable in $[0,1]$ (see, for instance, Billingsley 1995, Section 2). The random variables X are then functions from $[0,1]$ to $\mathcal{X}$, that is, functions of uniform variates transformed by the *generalized inverse* function.

Definition 2.3. For a non-decreasing function F on $\mathbb{R}$, the *generalized inverse* of F, F^-, is the function defined by

$$F^-(u) = \inf\{x : \ F(x) \geq u\} \ . \tag{2.1}$$

We then have the following lemma, sometimes known as the *probability integral transform*, which gives us a representation of any random variable as a transform of a uniform random variable.

Lemma 2.4. *If $U \sim \mathcal{U}_{[0,1]}$, then the random variable $F^-(U)$ has the distribution F.*

Proof. For all $u \in [0,1]$ and for all $x \in F^-([0,1])$, the generalized inverse satisfies

$$F(F^-(u)) \geq u \quad \text{and} \quad F^-(F(x)) \leq x \ .$$

Therefore,

$$\{(u,x) : \ F^-(u) \leq x\} = \{(u,x) : \ F(x) \geq u\}$$

and

$$P(F^-(U) \leq x) = P(U \leq F(x)) = F(x) \ .$$

□

Thus, formally, in order to generate a random variable $X \sim F$, it suffices to generate U according to $\mathcal{U}_{[0,1]}$ and then make the transformation $x = F^-(u)$.

Example 2.5. Exponential variable generation. If $X \sim \mathcal{E}xp(1)$, so $F(x) = 1 - e^{-x}$, then solving for x in $u = 1 - e^{-x}$ gives $x = -\log(1-u)$. Therefore, if $U \sim \mathcal{U}_{[0,1]}$, the random variable $X = -\log U$ has the exponential distribution (as U and $1 - U$ are both uniform). ‖

The generation of uniform random variables is therefore a key determinant in the behavior of simulation methods for other probability distributions, since those distributions can be represented as a deterministic transformation of uniform random variables. (Although, in practice, we often use methods other than that of Lemma 2.4, this basic representation is usually a good way to think about things. Note also that Lemma 2.4 implies that a bad choice of a uniform random number generator can invalidate the resulting simulation procedure.)

As mentioned above, from a theoretical point of view an operational version of any probability space $(\Omega, \mathcal{A}, P)$ can be created from the uniform distribution $\mathcal{U}_{[0,1]}$ and Lemma 2.4. Thus, the generation of any sequence of random variables can be formally implemented through the uniform generator Kiss. In practice, however, this approach only applies when the cumulative distribution functions are "explicitly" available, in the sense that there exists an algorithm allowing the computation of $F^-(u)$ in acceptable time. In particular, for distributions with explicit forms of F^- (for instance, the exponential, double-exponential, or Weibull distributions; see Problem 2.5 for other examples), Lemma 2.4 does lead to a practical implementation. But this situation only covers a small number of cases, described in Section 2.2 and additional problems. Other methods, like the Accept–Reject method of Section 2.3, are more general and do not use any strong analytic property of the densities. Thus, they can handle more general cases as, for example, the simulation of distributions in dimensions greater than one.

2.1.3 Alternatives

Although computation by Monte Carlo methods can be thought of as an exact calculation (as the order of accuracy is only a function of computation time), it is probably more often thought of as an approximation. Thus, numerical approximation is an alternative to Monte Carlo, and should also be considered a candidate for solving any particular problem. The following example shows how numerical approximations can work in the calculation of normal probabilities (see Sections 1.4, 3.6.2 and 3.4 for other approaches).

Example 2.6. Normal probabilities. Although Φ, the cumulative distribution function of the normal distribution cannot be expressed explicitly, since

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\{-z^2/2\} dz,$$

there exist approximations of Φ and of Φ^{-1} up to an arbitrary precision. For instance, Abramowitz and Stegun (1964) give the approximation

$$\Phi(x) \simeq 1 - \varphi(x) \left[b_1 t + b_2 t^2 + b_3 t^3 + b_4 t^4 + b_5 t^5\right] \qquad (x > 0),$$

where φ denotes the normal density, $t = (1 + px)^{-1}$ and

$$\begin{array}{lll} p = 0.2316419, & b_1 = 0.31938, & b_2 = -0.35656, \\ b_3 = 1.78148, & b_4 = -1.82125, & b_5 = 1.33027. \end{array}$$

Similarly, we also have the approximation

$$\Phi^{-1}(\alpha) \simeq t - \frac{a_0 + a_1 t}{1 + b_1 t + b_2 t^2},$$

where $t^2 = \log(\alpha^{-2})$ and

$$a_0 = 2.30753, \quad a_1 = 0.27061, \quad b_1 = 0.99229, \quad b_2 = 0.04481.$$

These two approximations are exact up to an error of order 10^{-8}, the error being absolute. If no other fast simulation method was available, this approximation could be used in settings which do not require much precision in the tails of $\mathcal{N}(0,1)$. (However, as shown in Example 2.8, there exists an exact and much faster algorithm.) ‖

2.1.4 Optimal Algorithms

Devroye 1985 presents a more comprehensive (one could say almost exhaustive!) treatment of the methods of random variable generation than the one presented in this chapter, in particular looking at refinements of existing algorithms in order to achieve uniformly optimal performances. (We strongly urge the reader to consult this book[4] for a better insight on the implications of this goal in terms of probabilistic and algorithmic complexity.)

Some refinements of the simulation techniques introduced in this chapter will be explored in Chapter 4, where we consider ways to accelerate Monte Carlo methods. At this point, we note that the concepts of "optimal" and "efficient" algorithms are particularly difficult to formalize. We can naively compare two algorithms, $[B_1]$ and $[B_2]$ say, in terms of time of computation, for instance through the average generation time of one observation. However, such a comparison depends on many subjective factors like the quality of the programming, the particular programming language used to implement the method, and the particular machine on which the program runs. More importantly, it does not take into account the conception and programming (and debugging) times, nor does it incorporate the specific use of the sample produced, partly because a quantification of these factors is generally impossible. For instance, some algorithms have a decreasing efficiency when the sample size increases. The reduction of the efficiency of a given algorithm to its average computation time is therefore misleading and we only use this type of measurement in settings where $[B_1]$ and $[B_2]$ are already of the same complexity. Devroye (1985) also notes that the simplicity of algorithms should be accounted for in their evaluation, since complex algorithms facilitate programming errors and, therefore, may lead to important time losses.[5]

A last remark to bring this section to its end is that simulation of the standard distributions presented here is accomplished quite efficiently by many statistical programming packages (for instance, `Gauss`, `Mathematica`, `Matlab`, `R`, `Splus`). When the generators from these general-purpose packages are easily accessible (in terms of programming), it is probably preferable to use such

[4] The book is now out-of-print but available for free on the author's website, at McGill University, Montréal, Canada.

[5] In fact, in numerous settings, the time required by a simulation is overwhelmingly dedicated to programming. This is, at least, the case for the authors themselves!!!

a generator rather than to write one's own. However, if a generation technique will get extensive use or if there are particular features of a problem that can be exploited, the creation of a personal library of random variable generators can accelerate analyses and even improve results, especially if the setting involves "extreme" cases (sample size, parameter values, correlation structure, rare events) for which the usual generators are poorly adapted. The investment represented by the creation and validation of such a personal library must therefore be weighed against the potential benefits.

2.2 General Transformation Methods

When a distribution f is linked in a relatively simple way to another distribution that is easy to simulate, this relationship can often be exploited to construct an algorithm to simulate variables from f. In this section we present alternative (to Lemma 2.4) techniques for generating nonuniform random variables. Some of these methods are rather case-specific, and are difficult to generalize as they rely on properties of the distribution under consideration and its relation with other probability distributions.

We begin with an illustration of some distributions that are simple to generate.

Example 2.7. Building on exponential random variables. In Example 2.5 we saw how to generate an exponential random variable starting from a uniform. Now we illustrate some of the random variables that can be generated starting from an exponential distribution. If the X_i's are iid $\mathcal{E}xp(1)$ random variables, then

$$Y = 2\sum_{j=1}^{\nu} X_j \sim \chi^2_{2\nu}\,, \qquad \nu \in \mathbb{N}^*\,,$$

$$Y = \beta\sum_{j=1}^{a} X_j \sim \mathcal{G}a(a,\beta)\,, \qquad a \in \mathbb{N}^*\,, \tag{2.2}$$

$$Y = \frac{\sum_{j=1}^{a} X_j}{\sum_{j=1}^{a+b} X_j} \sim \mathcal{B}e(a,b)\,, \qquad a,b \in \mathbb{N}^*\,.$$

Other derivations are possible (see Problem 2.6). ‖

These transformations are quite simple to use and, hence, will often be a favorite. However, there are limits to their usefulness, both in scope of variables that can be generated and in efficiency of generation. For example, as we will see, there are more efficient algorithms for Gamma and Beta random variables. Also, we cannot use exponentials to generate Gamma random variables with a non-integer shape parameter. For instance, we cannot get a χ^2_1

variable, which would, in turn, get us a $\mathcal{N}(0,1)$ variable. For that, we look at the following example of the *Box–Muller* algorithm (1958) for the generation of $\mathcal{N}(0,1)$ variables.

Example 2.8. Normal variable generation. If r and θ are the polar coordinates of (X_1, X_2), then, since the distribution of (X_1, X_2) is rotation invariant (see Problem 2.7)

$$r^2 = X_1^2 + X_2^2 \sim \chi^2_2 = \mathcal{E}xp(1/2)\,,$$
$$\theta \sim \mathcal{U}_{[0,2\pi]}\,.$$

If U_1 and U_2 are iid $\mathcal{U}_{[0,1]}$, the variables X_1 and X_2 defined by

$$X_1 = \sqrt{-2\log(U_1)}\,\cos(2\pi U_2)\,, \qquad X_2 = \sqrt{-2\log(U_1)}\,\sin(2\pi U_2)\,,$$

are then iid $\mathcal{N}(0,1)$. The corresponding algorithm is

Algorithm A.3 –Box–Muller–

1 Generate U_1, U_2 iid $\mathcal{U}_{[0,1]}$

2 Define

$$\begin{cases} x_1 = \sqrt{-2\log(u_1)}\cos(2\pi u_2)\,, \\ x_2 = \sqrt{-2\log(u_1)}\sin(2\pi u_2)\,, \end{cases} \tag{A.3}$$

3 Take x_1 and x_2 as two independent draws from $\mathcal{N}(0,1)$.

In comparison with algorithms based on the Central Limit Theorem, this algorithm is exact, producing two normal random variables from two uniform random variables, the only drawback (in speed) being the necessity of calculating functions such as log, cos, and sin. If this is a concern, Devroye (1985) gives faster alternatives that avoid the use of these functions (see also Problems 2.8 and 2.9). ‖

Example 2.9. Poisson generation. The Poisson distribution is connected to the exponential distribution through the Poisson process; that is, if $N \sim \mathcal{P}(\lambda)$ and $X_i \sim \mathcal{E}xp(\lambda)$, $i \in \mathbb{N}^*$, then

$$P_\lambda(N = k) = P_\lambda(X_1 + \cdots + X_k \le 1 < X_1 + \cdots + X_{k+1})\,.$$

Thus, the Poisson distribution can be simulated by generating exponential random variables until their sum exceeds 1. This method is simple, but is really practical only for smaller values of λ. On average, the number of exponential variables required is λ, and this could be prohibitive for large values of λ. In these settings, Devroye (1981) proposed a method whose computation time

is uniformly bounded (in λ) and we will see another approach, suitable for large λ's, in Example 2.23. Note also that a generator of Poisson random variables can produce negative binomial random variables since, when $Y \sim \mathcal{G}a(n,(1-p)/p)$ and $X|y \sim \mathcal{P}(y)$, $X \sim \mathcal{N}eg(n,p)$. (See Problem 2.13.) ‖

Example 2.9 shows a specific algorithm for the generation of Poisson random variables. Based on an application of Lemma 2.4, we can also construct a generic algorithm that will work for any discrete distribution.

Example 2.10. Discrete random variables. To generate $X \sim P_\theta$, we can calculate (once for all) the probabilities

$$p_0 = P_\theta(X \le 0), \quad p_1 = P_\theta(X \le 1), \quad p_2 = P_\theta(X \le 2), \quad \dots$$

then generate $U \sim \mathcal{U}_{[0,1]}$ and take

$$X = k \text{ if } p_{k-1} < U < p_k.$$

For example, to generate $X \sim \mathcal{B}in(10,.3)$, the first values are

$$p_0 = 0.028, \quad p_1 = 0.149, \quad p_2 = 0.382, \dots, p_{10} = 1\,,$$

and to generate $X \sim \mathcal{P}(7)$, take

$$p_0 = 0.0009, \quad p_1 = 0.0073, \quad p_2 = 0.0296, \dots$$

the sequence being stopped when it reaches 1 with a given number of decimals. (For instance, $p_{20} = 0.999985$.) Specific algorithms, such as Example 2.9, are usually more efficient but it is mostly because of the storage problem. See Problem 2.12 and Devroye (1985). ‖

Example 2.11. Beta generation. Consider $U_1, \dots, U_n$, an iid sample from $\mathcal{U}_{[0,1]}$. If $U_{(1)} \le \cdots \le U_{(n)}$ denotes the ordered sample, that is, the *order statistics* of the original sample, $U_{(i)}$ is distributed as $\mathcal{B}e(i, n-i+1)$ and the vector of the differences $(U_{(i_1)}, U_{(i_2)} - U_{(i_1)}, \dots, U_{(i_k)} - U_{(i_{k-1})}, 1 - U_{(i_k)})$ has a Dirichlet distribution $\mathcal{D}(i_1, i_2 - i_1, \dots, n - i_k + 1)$ (see Problem 2.17). However, even though these probabilistic properties allow the direct generation of Beta and Dirichlet random variables from uniform random variables, they do not yield efficient algorithms. The calculation of the order statistics can, indeed, be quite time-consuming since it requires sorting the original sample. Moreover, it only applies for integer parameters in the Beta distribution.

The following result allows for an alternative generation of Beta random variables from uniform random variables: Jöhnk's Theorem (see Jöhnk 1964 or Devroye 1985) states that if U and V are iid $\mathcal{U}_{[0,1]}$, the distribution of

$$\frac{U^{1/\alpha}}{U^{1/\alpha} + V^{1/\beta}},$$

conditional on $U^{1/\alpha} + V^{1/\beta} \le 1$, is the $\mathcal{B}e(\alpha,\beta)$ distribution. However, given the constraint on $U^{1/\alpha} + V^{1/\beta}$, this result does not provide a good algorithm to generate $\mathcal{B}e(\alpha,\beta)$ random variables for large values of α and β, as shown by the fast decrease of the probability of accepting a pair (U,V) as a function of $\alpha = \beta$ in Figure 2.2. ‖

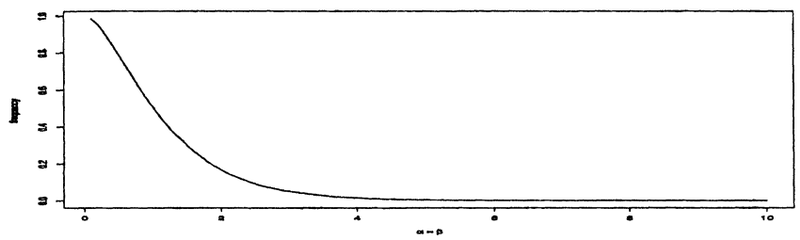


Fig. 2.2. Probability of accepting a pair (U,V) in Jöhnk (1964) algorithm as a function of α, when $\alpha = \beta$.

Example 2.12. Gamma generation. Given a generator of Beta random variables, we can derive a generator of Gamma random variables $\mathcal{G}a(\alpha,1)$ $(\alpha < 1)$ the following way: If $Y \sim \mathcal{B}e(\alpha, 1-\alpha)$ and $Z \sim \mathcal{E}xp(1)$, then $X = YZ \sim \mathcal{G}a(\alpha,1)$. Indeed, by making the transformation $x = yz, w = z$ and integrating the joint density, we find

$$\begin{aligned} f(x) &= \frac{\Gamma(1)}{\Gamma(\alpha)\Gamma(1-\alpha)} \int_x^\infty \left(\frac{x}{w}\right)^{\alpha-1} \left(1 - \frac{x}{w}\right)^{-\alpha} w^{-1} e^{-w} dw \\ &= \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x}. \end{aligned} \tag{2.3}$$

Alternatively, if we can start with a Gamma random variable, a more efficient generator for $\mathcal{G}a(\alpha,1)$ $(\alpha < 1)$ can be constructed: If $Y \sim \mathcal{G}a(\alpha+1,1)$ and $U \sim \mathcal{U}_{[0,1]}$, independent, then $X = YU^{1/\alpha}$ is distributed according to $\mathcal{G}a(\alpha,1)$, since

$$f(x) \propto \int_x^\infty w^\alpha e^{-w} \left(\frac{x}{w}\right)^{\alpha-1} w^{-1} dw = x^{\alpha-1}\, e^{-x}\,. \tag{2.4}$$

(See Stuart 1962 or Problem 2.14). ‖

The representation of a probability density as in (2.3) is a particular case of a *mixture of distributions*. Not only does such a representation induce relatively efficient simulation methods, but it is also related to methods in Chapters 9 and 10. The principle of a mixture representation is to write a density f as the marginal of another distribution, in the form

$$f(x) = \int_{\mathcal{Y}} g(x,y)\,dy \qquad \text{or} \qquad f(x) = \sum_{i\in\mathcal{Y}} p_i\, f_i(x)\,, \tag{2.5}$$

depending on whether $\mathcal{Y}$ is continuous or discrete. For instance, if the joint distribution $g(x,y)$ is simple to simulate, then the variable X can be obtained as a component of the generated (X,Y). Alternatively, if the component distributions $f_i(x)$ can be easily generated, X can be obtained by first choosing f_i with probability p_i and then generating an observation from f_i.

Example 2.13. Student's t generation. A useful form of (2.5) is

$$f(x) = \int_{\mathcal{Y}} g(x,y)\,dy = \int_{\mathcal{Y}} h_1(x|y)h_2(y)\,dy\,, \tag{2.6}$$

where h_1 and h_2 are the conditional and marginal densities of $X|Y=y$ and Y, respectively. For example, we can write Student's t density with ν degrees of freedom in this form, where

$$X|y \sim \mathcal{N}(0,\nu/y) \quad \text{and} \quad Y \sim \chi^2_\nu.$$

Such a representation is also useful for discrete distributions. In Example 2.9, we noted an alternate representation for the negative binomial distribution. If X is negative binomial, $X \sim \mathcal{N}eg(n,p)$, then $P(X=x)$ can be written as (2.6) with

$$X|y \sim \mathcal{P}(y) \quad \text{and} \quad Y \sim \mathcal{G}(n,\beta),$$

where $\beta = (1-p)/p$. Note that the discreteness of the negative binomial distribution does not result in a discrete mixture representation of the probability. The mixture is continuous, as the distribution of Y is itself continuous. ‖

Example 2.14. Noncentral chi squared generation. The noncentral chi squared distribution, $\chi^2_p(\lambda)$, also allows for a mixture representation, since it can be written as a sum of central chi squared densities. In fact, it is of the form (2.6) with h_1 the density of a χ^2_{p+2K} distribution and h_2 the density of $\mathcal{P}(\lambda/2)$. However, this representation is not as efficient as the algorithm obtained by generating $Z \sim \chi^2_{p-1}$ and $Y \sim \mathcal{N}(\sqrt{\lambda},1)$, and using the fact that $Z+Y^2 \sim \chi^2_p(\lambda)$. Note that the noncentral chi squared distribution does not have an explicit form for its density function. It is either represented as an infinite mixture (see (3.31)) or by using modified Bessel functions (see Problem 1.8). ‖

In addition to the above two examples, other distributions can be represented as mixtures (see, for instance, Gleser 1989). In many cases this representation can be exploited to produce algorithms for random variable generation (see Problems 2.24–2.26, and Note 2.6.3).

2.3 Accept–Reject Methods

There are many distributions from which it is difficult, or even impossible, to directly simulate by an inverse transform. Moreover, in some cases, we are not even able to represent the distribution in a usable form, such as a transformation or a mixture. In such settings, it is impossible to exploit direct probabilistic properties to derive a simulation method. We thus turn to another class of methods that only requires us to know the functional form of the density f of interest up to a multiplicative constant; no deep analytical study of f is necessary. The key to this method is to use a simpler (simulationwise) density g from which the simulation is actually done. For a given density g—called the *instrumental density*— there are thus many densities f—called the *target densities*—which can be simulated this way. The corresponding algorithm, called *Accept–Reject*, is based on a simple connection with the uniform distribution, discussed below.

2.3.1 The Fundamental Theorem of Simulation

There exists a fundamental (simple!) idea that underlies the Accept–Reject methodology, and also plays a key role in the construction of the slice sampler (Chapter 8). If f is the density of interest, on an arbitrary space, we can write

$$f(x) = \int_0^{f(x)} du\,. \tag{2.7}$$

Thus, f appears as the *marginal density* (in X) of the joint distribution,

$$(X,U) \sim \mathcal{U}\{(x,u) : 0 < u < f(x)\}\,. \tag{2.8}$$

Since U is not directly related to the original problem, it is called an *auxiliary variable*, a notion to be found again in later chapters like Chapters 8–10.

Although it seems like we have not gained much, the introduction of the auxiliary uniform variable in (2.7) has brought a considerably different perspective: Since (2.8) is the joint density of X and U, we can generate from this joint distribution by just generating uniform random variables on the constrained set $\{(x,u) : 0 < u < f(x)\}$. Moreover, since the marginal distribution of X *is* the original target distribution, f, by generating a uniform variable on $\{(x,u) : 0 < u < f(x)\}$, we have generated a random variable from f. And this generation was produced without using f other than through the calculation of $f(x)$! The importance of this equivalence is stressed in the following theorem:

Theorem 2.15 (Fundamental Theorem of Simulation). *Simulating*

$$X \sim f(x)$$

is equivalent to simulating

$$(X,U) \sim \mathcal{U}\{(x,u) : 0 < u < f(x)\}\,.$$

While this theorem is fundamental in many respects, it appears mostly as a formal representation at this stage because the simulation of the uniform pair (X, U) is often not straightforward. For example, we could simulate $X \sim f(x)$ and $U|X = x \sim \mathcal{U}(0, f(x))$, but then this makes the whole representation useless. And the symmetric approach, which is to simulate U from its marginal distribution, and then X from the distribution conditional on $U = u$, does not often result in a feasible calculation. The solution is to simulate the entire pair (X, U) at once in a bigger set, where simulation is easier, and then take the pair if the constraint is satisfied.

For example, in a one-dimensional setting, suppose that

$$\int_a^b f(x)dx = 1$$

and that f is bounded by m. We can then simulate the random pair $(Y, U) \sim \mathcal{U}(0 < u < m)$ by simulating $Y \sim \mathcal{U}(a, b)$ and $U|Y = y \sim \mathcal{U}(0, m)$, and take the pair only if the further constraint $0 < u < f(y)$ is satisfied. This results in the correct distribution of the accepted value of Y, call it X, because

$$\begin{aligned} P(X \le x) &= P(Y \le x | U < f(Y)) \\ &= \frac{\int_a^x \int_0^{f(y)} du\, dy}{\int_a^b \int_0^{f(y)} du\, dy} = \int_a^x f(y)\, dy. \end{aligned} \tag{2.9}$$

This amounts to saying that, if $A \subset B$ and if we generate a uniform sample on B, keeping only the terms of this sample that are in A will result in a uniform sample on A (with a random size that is independent of the values of the sample).

Example 2.16. Beta simulation. We have seen (Example 2.11) that direct simulation of Beta random variables can be difficult. However, we can easily use Theorem 2.15 for this simulation when $\alpha \ge 1$ and $\beta \ge 1$. Indeed, to generate $X \sim \mathcal{B}e(\alpha, \beta)$, we take $Y \sim \mathcal{U}_{[0,1]}$ and $U \sim \mathcal{U}_{[0,m]}$, where m is the maximum of the Beta density (Problem 2.15). For $\alpha = 2.7$ and $\beta = 6.3$ Figure 2.3 shows the results of generating 1000 pairs (Y, U). The pairs that fall under the density function are those for which we accept $X = Y$, and we reject those pairs that fall outside. ‖

In addition, it is easy to see that the probability of acceptance of a given simulation in the box $[a, b] \times [0, m]$ is given by

$$P(\text{Accept}) = P(U < f(Y)) = \frac{1}{m} \int_0^1 \int_0^{f(y)} du\, dy = \frac{1}{m}.$$

For Example 2.16, $m = 2.67$, so we accept approximately $1/2.67 = 37\%$ of the values.

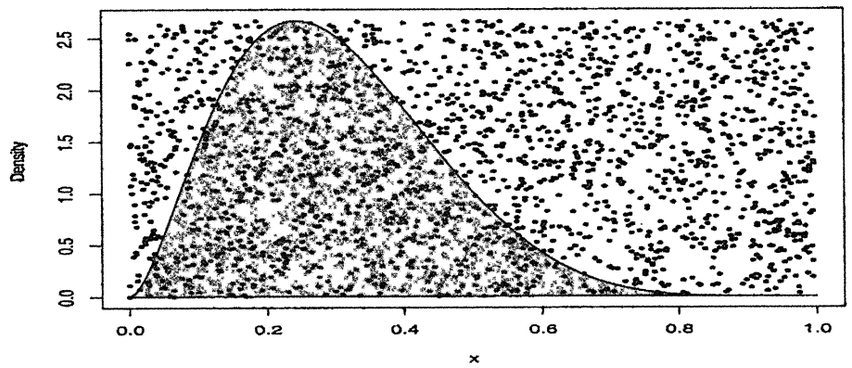


Fig. 2.3. Generation of Beta random variables: Using Theorem 2.15, 1000 (Y, U) pairs were generated, and 365 were accepted (the black circles under the Beta $\mathcal{B}e(2.7, 6.3)$ density function).

The argument leading to (2.9) can easily be generalized to the situation where the larger set is not a box any longer, as long as simulating uniformly over this larger set is feasible. This generalization may then allow for cases where either or both of the support of f and the maximum of f are unbounded. If the larger set is of the form

$$\mathcal{L} = \{(y, u) : 0 < u < m(y)\},$$

the constraints are thus that $m(x) \ge f(x)$ and that simulation of a uniform on $\mathcal{L}$ is feasible. Obviously, efficiency dictates that m be as close as possible to f in order to avoid wasting simulations. A remark of importance is that, because of the constraint $m(x) \ge f(x)$, m cannot be a probability density. We then write

$$m(x) = Mg(x) \text{ where } \int_{\mathcal{X}} m(x)\, dx = \int_{\mathcal{X}} Mg(x)\, dx = M,$$

since m is necessarily integrable (otherwise, $\mathcal{L}$ would not have finite mass and a uniform distribution would not exist on $\mathcal{L}$). As mentioned above, a natural way of simulating the uniform on $\mathcal{L}$ is then to use (2.7) backwards, that is, to simulate $Y \sim g$ and then $U|Y = y \sim \mathcal{U}(0, Mg(y))$. If we only accept the y's such that the constraint $u < f(y)$ is satisfied, we have

$$\begin{aligned} P(X \in \mathcal{A}) &= P(Y \in \mathcal{A} | U < f(Y)) \\ &= \frac{\int_{\mathcal{A}} \int_0^{f(y)} \frac{du}{Mg(y)}\, g(y) dy}{\int_{\mathcal{A}} \int_0^{f(y)} \frac{du}{Mg(y)}\, g(y) dy} = \int_{\mathcal{A}} f(y)\, dy \end{aligned}$$

for every measurable set $\mathcal{A}$ and the accepted X's are indeed distributed from f. We have thus derived a more general implementation of the fundamental theorem, as follows:

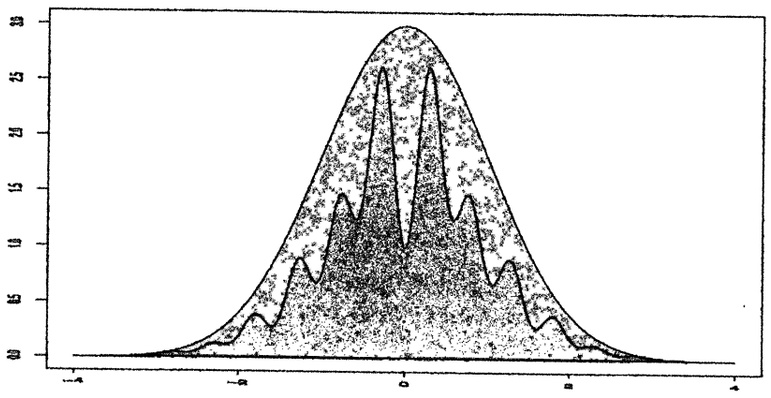


Fig. 2.4. Plot of a uniform sample over the set $\{(x,u) : 0 < u < f(x)\}$ for $f(x) \propto \exp(-x^2/2)(\sin(6x)^2 + 3\cos(x)^2\sin(4x)^2 + 1)$ and of the envelope function $g(x) = 5\exp(-x^2/2)$.

Corollary 2.17. *Let $X \sim f(x)$ and let $g(x)$ be a density function that satisfies $f(x) \leq Mg(x)$ for some constant $M \geq 1$. Then, to simulate $X \sim f$, it is sufficient to generate*

$$Y \sim g \qquad \text{and} \qquad U|Y = y \sim \mathcal{U}(0, Mg(y))\,,$$

until $0 < u < f(y)$.

Figure 2.4 illustrates Corollary 2.17 for the target density

$$f(x) \propto \exp(-x^2/2)(\sin(6x)^2 + 3\cos(x)^2\sin(4x)^2 + 1)$$

with upper bound (or, rather, dominating density) the normal density

$$g(x) = \exp(-x^2/2)/\sqrt{2\pi}\,,$$

which is obviously straightforward to generate.

Corollary 2.17 has two consequences. First, it provides a generic method to simulate from any density f that is known *up to a multiplicative factor*; that is, the normalizing constant of f need not be known, since the method only requires input of the ratio f/M, which does not depend on the normalizing constant. This is for instance the case of Figure 2.4, where the normalizing constant of f is unknown. This property is particularly important in Bayesian calculations. There, a quantity of interest is the posterior distribution, defined according to Bayes Theorem by

$$\pi(\theta|x) \propto \pi(\theta)\, f(x|\theta)\,. \tag{2.10}$$

Thus, the posterior density $\pi(\theta|x)$ is easily specified up to a normalizing constant and, to use Corollary 2.17, this constant need not be calculated. (See Problem 2.29.)

Of course, there remains the task of finding a density g satisfying $f \leq Mg$, a bound that need not be tight, in the sense that Corollary 2.17 remains valid when M is replaced with any larger constant. (See Problem 2.30.)

A second consequence of Corollary 2.17 is that the probability of acceptance is exactly $1/M$ (a geometric waiting time), when evaluated for the properly normalized densities, and the expected number of trials until a variable is accepted is M (see Problem 2.30). Thus, a comparison between different simulations based on different instrumental densities $g_1, g_2, \ldots$ can be undertaken through the comparison of the respective bounds $M_1, M_2, \ldots$ (as long as the corresponding densities $g_1, g_2, \ldots$ are correctly normalized). In particular, a first method of optimizing the choice of g in $g_1, g_2, \ldots$ is to find the smallest bound M_i. However, this first and rudimentary comparison technique has some limitations, which we will see later in this chapter.

2.3.2 The Accept–Reject Algorithm

The implementation of Corollary 2.17 is known as the *Accept–Reject method*, which is usually stated in the slightly modified, but equivalent form. (See Problem 2.28 for extensions.)

Algorithm A.4 –Accept–Reject Method–

1. Generate $X \sim g$, $U \sim \mathcal{U}_{[0,1]}$;
2. Accept $Y = X$ if $U \leq f(X)/Mg(X)$; [A.4]
3. Return to 1. otherwise.

In cases where f and g are normalized so they are both probability densities, the constant M is necessarily larger than 1. Therefore, the size of M, and thus the efficiency of [A.4], becomes a function of how closely g can imitate f, especially in the tails of the distribution. Note that for f/g to remain bounded, it is necessary for g to have tails thicker than those of f. It is therefore impossible for instance to use [A.4] to simulate a Cauchy distribution f using a normal distribution g; however, the reverse works quite well. (See Problem 2.34.) Interestingly enough, the opposite case when g/f is bounded can also be processed by a tailored Markov chain Monte Carlo algorithm derived from Doukhan et al. (1994) (see Problems 7.5 and 7.6).

A limited optimization of the Accept–Reject algorithm is possible by choosing the instrumental density g in a parametric family, and then determining the value of the parameter which minimizes the bound M. A similar comparison between two parametric families is much more delicate since it is then necessary to take into account the computation time of one generation from g in [A.4]. In fact, pushing the reasoning to the limit, if $g = f$

and if we simulate $X \sim f$ by numerical inversion of the distribution function, we formally achieve the minimal bound $M = 1$, but this does not guarantee that we have an efficient algorithm, as can be seen in the case of the normal distribution.

Example 2.18. Normals from double exponentials. Consider generating a $\mathcal{N}(0,1)$ by [A.4] using a double-exponential distribution $\mathcal{L}(\alpha)$, with density $g(x|\alpha) = (\alpha/2)\exp(-\alpha|x|)$. It is then straightforward to show that

$$\frac{f(x)}{g(x|\alpha)} \leq \sqrt{2/\pi}\alpha^{-1}e^{\alpha^2/2}$$

and that the minimum of this bound (in α) is attained for $\alpha = 1$. The probability of acceptance is then $\sqrt{\pi/2e} = .76$, which shows that to produce one normal random variable, this Accept–Reject algorithm requires on the average $1/.76 \approx 1.3$ uniform variables, to be compared with the fixed single uniform required by the Box–Muller algorithm. ‖

A real advantage of the Accept–Reject algorithm is illustrated in the following example.

Example 2.19. Gamma Accept–Reject We saw in Example 2.7 that if $\alpha \in \mathbb{N}$, the Gamma distribution $\mathcal{G}a(\alpha,\beta)$ can be represented as the sum of α exponential random variables $\epsilon_i \sim \mathcal{E}xp(\beta)$, which are very easy to simulate, since $\epsilon_i = -\log(U_i)/\beta$, with $U_i \sim \mathcal{U}([0,1])$. In more general cases (for example when $\alpha \notin \mathbb{N}$), this representation does not hold.

A possible approach is to use the Accept–Reject algorithm with instrumental distribution $\mathcal{G}a(a,b)$, with $a = [\alpha]$ ($\alpha \geq 1$). (Without loss of generality, suppose $\beta = 1$.) The ratio f/g is $b^{-a}x^{\alpha-a}\exp\{-(1-b)x\}$, up to a normalizing constant, yielding the bound

$$M = b^{-a}\left(\frac{\alpha-a}{(1-b)e}\right)^{\alpha-a}$$

for $b < 1$. Since the maximum of $b^{-a}(1-b)^{\alpha-a}$ is attained at $b = a/\alpha$, the optimal choice of b for simulating $\mathcal{G}a(\alpha,1)$ is $b = a/\alpha$, which gives the same mean for $\mathcal{G}a(\alpha,1)$ and $\mathcal{G}a(a,b)$. (See Problem 2.31.) ‖

It may also happen that the complexity of the optimization is very expensive in terms of analysis or of computing time. In the first case, the construction of the optimal algorithm should still be undertaken when the algorithm is to be subjected to intensive use. In the second case, it is most often preferable to explore the use of another family of instrumental distributions g.

Example 2.20. Truncated normal distributions. *Truncated normal distributions* appear in many contexts, such as in the discussion after Example 1.5. When constraints $x \geq \underline{\mu}$ produce densities proportional to

$$e^{-(x-\mu)^2/2\sigma^2}\,\mathbb{I}_{x\geq\underline{\mu}}$$

for a bound $\underline{\mu}$ large compared with μ, there are alternatives which are far superior to the naïve method in which a $\mathcal{N}(\mu,\sigma^2)$ distribution is simulated until the generated value is larger than $\underline{\mu}$. (This approach requires an average number of $1/\Phi((\mu-\underline{\mu})/\sigma)$ simulations from $\mathcal{N}(\mu,\sigma^2)$ for one acceptance.) Consider, without loss of generality, the case $\mu = 0$ and $\sigma = 1$. A potential instrumental distribution is the translated exponential distribution, $\mathcal{E}xp(\alpha,\underline{\mu})$, with density

$$g_\alpha(z) = \alpha e^{-\alpha(z-\underline{\mu})}\,\mathbb{I}_{z\geq\underline{\mu}}\,.$$

The ratio $f/g_\alpha(z) = e^{-\alpha(z-\underline{\mu})}\,e^{-z^2/2}$ is then bounded by $\exp(\alpha^2/2 - \alpha\underline{\mu})$ if $\alpha > \underline{\mu}$ and by $\exp(-\underline{\mu}^2/2)$ otherwise. The corresponding (upper) bound is

$$\begin{cases} 1/\alpha\,\exp(\alpha^2/2 - \alpha\underline{\mu}) & \text{if } \alpha > \underline{\mu}, \\ 1/\alpha\,\exp(-\underline{\mu}^2/2) & \text{otherwise.} \end{cases}$$

The first expression is minimized by

$$\alpha^* = \underline{\mu} + \frac{1}{2}\sqrt{\underline{\mu}^2+4}\,, \tag{2.11}$$

whereas $\tilde{\alpha} = \underline{\mu}$ minimizes the second bound. The optimal choice of α is therefore (2.11), which requires the computation of the square root of $\underline{\mu}^2+4$. Robert (1995b) proposes a similar algorithm for the case where the normal distribution is restricted to the interval $[\underline{\mu},\overline{\mu}]$. For some values of $[\underline{\mu},\overline{\mu}]$, the optimal algorithm is associated with a value of α, which is a solution to an implicit equation. (See also Geweke 1991 for a similar resolution of this simulation problem and Marsaglia 1964 for an earlier solution.) ‖

One criticism of the Accept–Reject algorithm is that it generates "useless" simulations when rejecting. We will see in Chapter 3 how the method of importance sampling (Section 3.3) can be used to bypass this problem and also how both methods can be compared.

2.4 Envelope Accept–Reject Methods

2.4.1 The Squeeze Principle

In numerous settings, the distribution associated with the density f is difficult to simulate because of the complexity of the function f itself, which may require substantial computing time at each evaluation. In the setup of Example 1.9 for instance, if a Bayesian approach is taken with θ distributed (a posteriori) as

$$\prod_{i=1}^{n}\left[1+\frac{(x_i-\theta)^2}{p\sigma^2}\right]^{-\frac{p+1}{2}}, \tag{2.12}$$

where σ is known, each single evaluation of $\pi(\theta|x)$ involves the computation of n terms in the product. It turns out that an acceleration of the simulation of densities such as (2.12) can be accomplished by an algorithm that is "one step beyond" the Accept–Reject algorithm. This algorithm is an *envelope* algorithm and relies on the evaluation of a simpler function g_l which bounds the target density f from below. The algorithm is based on the following extension of Corollary 2.17 (see Problems 2.35 and 2.36).

Lemma 2.21. *If there exist a density g_m, a function g_l and a constant M such that*

$$g_l(x) \le f(x) \le M g_m(x)\,,$$

then the algorithm

Algorithm A.5 –Envelope Accept–Reject–

1. Generate $X \sim g_m(x)$, $U \sim \mathcal{U}_{[0,1]}$;
2. Accept X if $U \le g_l(X)/Mg_m(X)$; [A.5]
3. otherwise, accept X if $U \le f(X)/Mg_m(X)$

produces random variables that are distributed according to f.

By the construction of a lower envelope on f, based on the function g_l, the number of evaluations of f is potentially decreased by a factor

$$\frac{1}{M}\int g_l(x)\,dx,$$

which is the probability that f is not evaluated. This method is called the *squeeze principle* by Marsaglia (1977) and the ARS algorithm [A.7] in Section 2.4 is based on it. A possible way of deriving the bounds g_l and Mg_m is to use a Taylor expansion of $f(x)$.

Example 2.22. Lower bound for normal generation. It follows from the Taylor series expansion of $\exp(-x^2/2)$ that $\exp(-x^2/2) \ge 1-(x^2/2)$, and hence

$$\left(1-\frac{x^2}{2}\right) \le f(x),$$

which can be interpreted as a lower bound for the simulation of $\mathcal{N}(0,1)$. This bound is obviously useless when $|X| < \sqrt{2}$, an event which occurs with probability 0.61 for $X \sim \mathcal{C}(0,1)$. ‖

Example 2.23. Poisson variables from logistic variables. As indicated in Example 2.9, the simulation of the Poisson distribution $\mathcal{P}(\lambda)$ using a Poisson process and exponential variables can be rather inefficient. Here, we describe a simpler alternative of Atkinson (1979), who uses the relationship between the Poisson $\mathcal{P}(\lambda)$ distribution and the *logistic distribution*. The logistic distribution has density and distribution function

$$f(x) = \frac{1}{\beta}\,\frac{\exp\{-(x-\alpha)/\beta\}}{[1+\exp\{-(x-\alpha)/\beta\}]^2} \quad \text{and} \quad F(x) = \frac{1}{1+\exp\{-(x-\alpha)/\beta\}}$$

and is therefore analytically invertible.

To better relate the continuous and discrete distributions, we consider $N = \lfloor x+0.5 \rfloor$, the integer part of $x+0.5$. Also, the range of the logistic distribution is $(-\infty,\infty)$, but to better match it with the Poisson, we restrict the range to $[-1/2,\infty)$. Thus, the random variable N has distribution function

$$P(N=n) = \frac{1}{1+e^{-(n+0.5-\alpha)/\beta}} - \frac{1}{1+e^{-(n-0.5-\alpha)/\beta}}$$

if $x > 1/2$ and

$$P(N=n) = \left(\frac{1}{1+e^{-(n+0.5-\alpha)/\beta}} - \frac{1}{1+e^{-(n-0.5-\alpha)/\beta}}\right)\frac{1+e^{-(0.5+\alpha)/\beta}}{e^{-(0.5+\alpha)/\beta}}$$

if $-1/2 < x \le 1/2$ and the ratio of the densities is

$$\lambda^n \big/ P(N=n)\, e^{\lambda}\, n!\,. \tag{2.13}$$

Although it is difficult to compute a bound on (2.13) and, hence, to optimize it in (α,β), Atkinson (1979) proposed the choice $\alpha = \lambda$ and $\beta = \pi/\sqrt{3\lambda}$. This identifies the two first moments of X with those of $\mathcal{P}(\lambda)$. For this choice of α and β, analytic optimization of the bound on (2.13) remains impossible, but numerical maximization and interpolation yields the bound $c = 0.767 - 3.36/\lambda$. The resulting algorithm is then

Algorithm A.6 –Atkinson's Poisson Simulation–

0. Define $\beta = \pi/\sqrt{3\lambda}$, $\alpha = \lambda\beta$ and $k = \log c - \lambda - \log\beta$.
1. Generate $U_1 \sim \mathcal{U}_{[0,1]}$ and calculate
$$x = \{\alpha - \log[(1-u_1)/u_1]\}/\beta$$
until $X > -0.5$.
2. Define $N = \lfloor X+0.5 \rfloor$ and generate $U_2 \sim \mathcal{U}_{[0,1]}$.
3. Accept $N \sim \mathcal{P}(\lambda)$ if [A.6]
$$\alpha - \beta u_1 + \log\left(u_2/\{1+\exp(\alpha-\beta u_1)\}^2\right) \le k + N\log\lambda - \log N!$$

Although the resulting simulation is exact, this algorithm is based on a number of approximations, both through the choice of (α, β) and in the computation of the majorization bounds and the density ratios. Moreover, note that it requires the computation of factorials, $N!$, which may be quite time-consuming. Therefore, although [A.6] usually has a reasonable efficiency, more complex algorithms such as those of Devroye (1985) may be preferable. ‖

2.4.2 Log-Concave Densities

The particular case of *log-concave densities* (that is, densities whose logarithm is concave) allows the construction of a generic algorithm that can be quite efficient.

Example 2.24. Log-concave densities. Recall the exponential family (1.9)

$$f(x) = h(x)\, e^{\theta \cdot x - \psi(\theta)}, \qquad \theta, x \in \mathbb{R}^k.$$

This density is log-concave if

$$\frac{\partial^2}{\partial x^2} \log f(x) = \frac{\partial^2}{\partial x^2} \log h(x) = \frac{h(x)h''(x) - [h'(x)]^2}{h^2(x)} < 0,$$

which will often be the case for the exponential family. For example, if $X \sim \mathcal{N}(\theta, 1)$, then $h(x) \propto \exp\{-x^2/2\}$ and $\partial^2 \log h(x)/\partial x^2 = -1$. See Problems 2.40–2.42 for properties and examples of log-concave densities. ‖

Devroye (1985) describes some algorithms that take advantage of the log-concavity of the density, but here we present a universal method. The algorithm, which was proposed by Gilks (1992) and Gilks and Wild (1992), is based on the construction of an envelope and the derivation of a corresponding Accept–Reject algorithm. The method is called *adaptive rejection sampling* (ARS) and it provides a sequential evaluation of lower and upper envelopes of the density f when $h = \log f$ is concave.

Let $\mathcal{S}_n$ be a set of points $x_i, i = 0, 1, \ldots, n+1$, in the support of f such that $h(x_i) = \log f(x_i)$ is known up to the same constant. Given the concavity of h, the line $L_{i,i+1}$ through $(x_i, h(x_i))$ and $(x_{i+1}, h(x_{i+1}))$ is below the graph of h in $[x_i, x_{i+1}]$ and is above this graph outside this interval (see Figure 2.5). For $x \in [x_i, x_{i+1}]$, if we define

$$\overline{h}_n(x) = \min\{L_{i-1,i}(x), L_{i+1,i+2}(x)\} \quad \text{and} \quad \underline{h}_n(x) = L_{i,i+1}(x)\,,$$

the envelopes are

$$\underline{h}_n(x) \le h(x) \le \overline{h}_n(x) \tag{2.14}$$

uniformly on the support of f. (We define

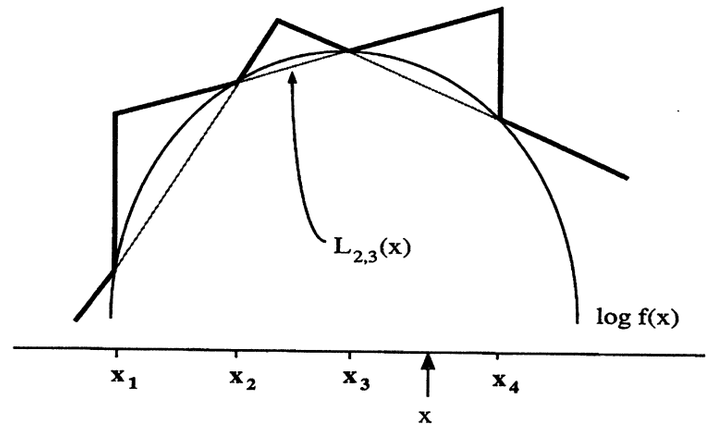


Fig. 2.5. Lower and upper envelopes of $h(x) = \log f(x)$, f a log-concave density (*Source:* Gilks et al. 1995).

$$\underline{h}_n(x) = -\infty \quad \text{and} \quad \overline{h}_n(x) = \min(L_{0,1}(x), L_{n,n+1}(x))$$

on $[x_0, x_{n+1}]^c$.) Therefore, for $\underline{f}_n(x) = \exp \underline{h}_n(x)$ and $\overline{f}_n(x) = \exp \overline{h}_n(x)$, (2.14) implies that

$$\underline{f}_n(x) \le f(x) \le \overline{f}_n(x) = \varpi_n\, g_n(x)\,,$$

where ϖ_n is the normalized constant of f_n; that is, g_n is a density. The ARS algorithm to generate an observation from f is thus

Algorithm A.7 –ARS Algorithm–

0. Initialize n and $\mathcal{S}_n$.
1. Generate $X \sim g_n(x)$, $U \sim \mathcal{U}_{[0,1]}$.
2. If $U \le \underline{f}_n(X)/\varpi_n\, g_n(X)$, accept X; [A.7]
   otherwise, if $U \le f(X)/\varpi_n\, g_n(X)$, accept X
   and update $\mathcal{S}_n$ to $\mathcal{S}_{n+1} = \mathcal{S}_n \cup \{X\}$.

An interesting feature of this algorithm is that the set S_n is only updated when $f(x)$ has been previously computed. As the algorithm produces variables $X \sim f(x)$, the two envelopes $\underline{f}_n$ and $\overline{f}_n$ become increasingly accurate and, therefore, we progressively reduce the number of evaluations of f. Note that in the initialization of S_n, a necessary condition is that $\varpi_n < +\infty$ (i.e., that g_n is actually a probability density). To achieve this requirement, $L_{0,1}$ needs to have positive slope if the support of f is not bounded on the left and $L_{n,n+1}$ needs to have a negative slope if the support of f is not bounded on the right. (See Problem 2.39 for more on simulation from g_n.)

The ARS algorithm is not optimal in the sense that it is often possible to devise a better specialized algorithm for a given log-concave density. However, although Gilks and Wild (1992) do not provide theoretical evaluations of simulation speeds, they mention reasonable performances in the cases they consider. Note that, in contrast to the previous algorithms, the function g_n is updated during the iterations and, therefore, the average computation time for one generation from f decreases with n. This feature makes the comparison with other approaches quite delicate.

The major advantage of [A.7] compared with alternatives is its *universality*. For densities f that are only known through their functional form, the ARS algorithm yields an automatic Accept–Reject algorithm that only requires checking f for log-concavity. Moreover, the set of log-concave densities is wide; see Problems 2.40 and 2.41. The ARS algorithm thus allows for the generation of samples from distributions that are rarely simulated, without requiring the development of case-specific Accept–Reject algorithms.

Example 2.25. Capture–recapture models. In a heterogeneous *capture–recapture model* (see Seber 1983, 1992 or Borchers et al. 2002), animals are captured at time i with probability p_i, the size N of the population being unknown. The corresponding likelihood is therefore

$$L(p_1,\ldots,p_I|N,n_1,\ldots,n_I) = \frac{N!}{(N-r)!}\prod_{i=1}^{I} p_i^{n_i}(1-p_i)^{N-n_i},$$

where I is the number of captures, n_i is the number of captured animals during the ith capture, and r is the total number of different captured animals. If N is a priori distributed as a $\mathcal{P}(\lambda)$ variable and the p_i's are from a *normal logistic* model,

$$\alpha_i = \log\left(\frac{p_i}{1-p_i}\right) \sim \mathcal{N}(\mu_i,\sigma^2),$$

as in George and Robert (1992), the posterior distribution satisfies

$$\pi(\alpha_i|N,n_1,\ldots,n_I) \propto \exp\left\{\alpha_i n_i - \frac{1}{2\sigma^2}(\alpha_i-\mu_i)^2\right\} \Big/ (1+e^{\alpha_i})^N .$$

If this conditional distribution must be simulated (for reasons which will be made clearer in Chapters 9 and 10), the ARS algorithm can be implemented. In fact, the log of the posterior distribution

$$\alpha_i n_i - \frac{1}{2\sigma^2}(\alpha_i-\mu_i)^2 - N\,\log(1+e^{\alpha_i}) \tag{2.15}$$

is concave in α_i, as can be shown by computing the second derivative (see also Problem 2.42).

As an illustration, consider the dataset $(n_1,\ldots,n_{11}) = (32,20,8,5,1,2,0,2,1,1,0)$ which describes the number of recoveries over the years 1957–1968 of

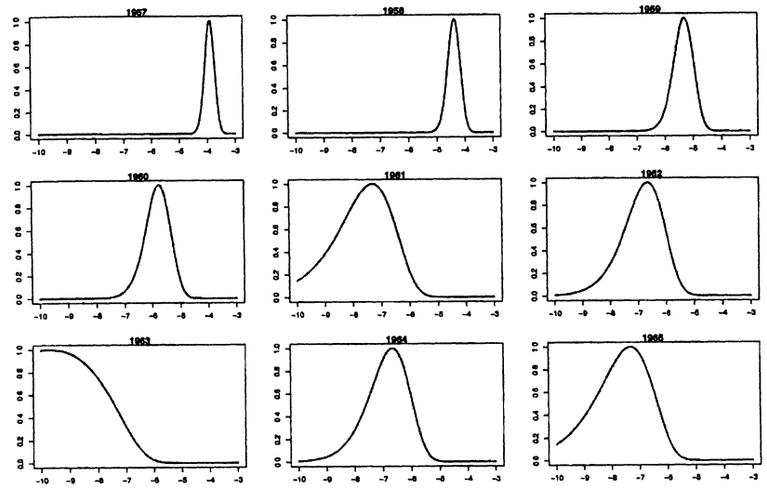


Fig. 2.6. Posterior distributions of the capture log-odds ratios for the Northern Pintail duck dataset of Johnson and Hoeting (2003) for the years 1957–1965.

$N = 1612$ Northern Pintail ducks banded in 1956, as reported in Johnson and Hoeting (2003). Figure 2.6 provides the corresponding posterior distributions for the first 9 α_i's. The ARS algorithm can then be used independently for each of these distributions. For instance, if we take the year 1960, the starting points in $\mathcal{S}$ can be $-10, -6$ and -3. The set $\mathcal{S}$ then gets updated along iterations as in Algorithm [A.7], which provides a correct simulation from the posterior distributions of the α_i's, as illustrated in Figure 2.7 for the year 1960. ‖

The above example also illustrates that checking for log-concavity of a Bayesian posterior distribution is straightforward, as $\log\,\pi(\theta|x) = \log\,\pi(\theta) + \log\,f(x|\theta) + c$, where c is a constant (in θ). This implies that the log-concavity of $\pi(\theta)$ and of $f(x|\theta)$ (in θ) are sufficient to conclude the log-concavity of $\pi(\theta|x)$.

Example 2.26. Poisson regression. Consider a sample $(Y_1,x_1),\ldots,(Y_n,x_n)$ of integer-valued data Y_i with explanatory variable x_i, where Y_i and x_i are connected via a Poisson distribution,

$$Y_i|x_i \sim \mathcal{P}(\exp\{a+bx_i\}) .$$

If the prior distribution of (a,b) is a normal distribution $\mathcal{N}(0,\sigma^2)\times\mathcal{N}(0,\tau^2)$, the posterior distribution of (a,b) is given by

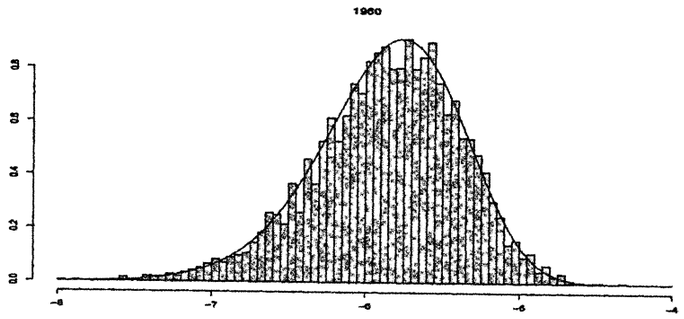


Fig. 2.7. Histogram of an ARS sample of 5000 points and corresponding posterior distribution of the log-odds ratio α_{1960}.

$$\pi(a, b|\mathbf{x}, \mathbf{y}) \propto \exp\left\{a\sum_i y_i + b\sum_i y_i x_i - e^a \sum_i e^{x_i b}\right\} e^{-a^2/2\sigma^2}\, e^{-b^2/2\tau^2}.$$

We will see in Chapter 9 that it is often of interest to simulate successively the (full) conditional distributions $\pi(a, |\mathbf{x}, \mathbf{y}, b)$ and $\pi(b|\mathbf{x}, \mathbf{y}, a)$. Since

$$\log \pi(a|\mathbf{x}, \mathbf{y}, b) = a\sum_i y_i - e^a \sum_i e^{x_i b} - a^2/2\sigma^2\,,$$
$$\log \pi(b|\mathbf{x}, \mathbf{y}, a) = b\sum_i y_i x_i - e^a \sum_i e^{x_i b} - b^2/2\tau^2\,,$$

and

$$\frac{\partial^2}{\partial a^2} \log \pi(a|\mathbf{x}, \mathbf{y}, b) = -\sum_i e^{x_i b}\, e^a - \sigma^{-2} < 0\,,$$
$$\frac{\partial^2}{\partial b^2} \log \pi(b|\mathbf{x}, \mathbf{y}, a) = -e^a \sum_i x_i^2\, e^{x_i b} - \tau^{-2} < 0\,,$$

the ARS algorithm directly applies for both conditional distributions.

As an illustration, consider the data in Table 2.1. This rather famous data set gives the deaths in the Prussian Army due to kicks from horses, gathered by von Bortkiewicz (1898). A question of interest is whether there is a trend in the deaths over time. For illustration here, we show how to generate the conditional distribution of the intercept, $\pi(a|\mathbf{x}, \mathbf{y}, b)$, since the generation of the other conditional is quite similar.

Before implementing the ARS algorithm, we note two simplifying things. One, if $f(x)$ is easy to compute (as in this example), there is really no need to construct $f_n(x)$, and we just skip that step in Algorithm [A.7]. Second, we do not need to construct the function g_n, we only need to know how to simulate

Table 2.1. Data from the 19^{th} century study by Bortkiewicz (1898) of deaths in the Prussian army due to horse kicks. The data are the number of deaths in fourteen army corps from 1875 to 1894.

Year	75	76	77	78	79	80	81	82	83	84
Deaths	3	5	7	9	10	18	6	14	11	9
Year	85	86	87	88	89	90	91	92	93	94
Deaths	5	11	15	6	11	17	12	15	8	4

from it. To do this, we only need to compute the area of each segment above the intervals $[x_i, x_{i+1}]$.

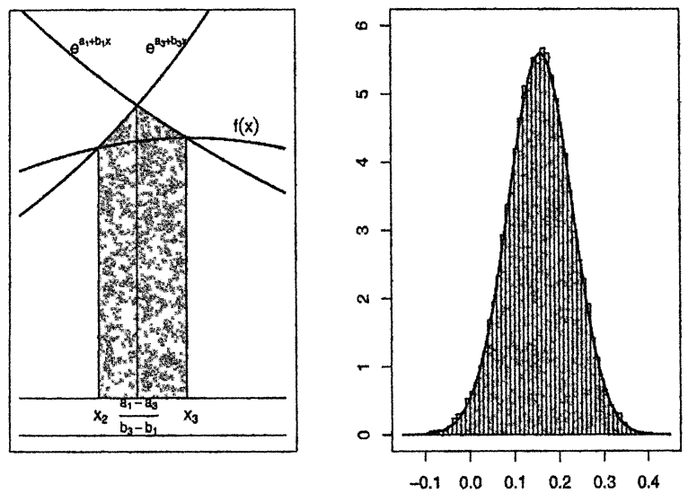


Fig. 2.8. Left panel is the area of integration for the weight of the interval $[x_2, x_3]$. The right panel is the histogram and density of the sample from g_n, with $b = .025$ and $\sigma^2 = 5$.

The left panel of Figure 2.8 shows the region of the support of $f(x)$ between x_2 and x_3, with the grey shaded area proportional to the probability of selecting the region $[x_2, x_3]$. If we denote by $a_i + b_i x$ the line through $(x_i, h(x_i))$ and $(x_{i+1}, h(x_{i+1}))$, then the area of the grey region of Figure 2.8 is

$$\begin{aligned}\omega_2 &= \int_{x_2}^{\frac{a_1-a_3}{b_3-b_1}} e^{a_1+b_1 x}\, dx + \int_{\frac{a_1-a_3}{b_3-b_1}}^{x_3} e^{a_3+b_3 x}\, dx \\ &= \frac{e^{a_1}}{b_1}\left[e^{\frac{a_1-a_3}{b_3-b_1}b1} - e^{b_1 x_2}\right] + \frac{e^{a_3}}{b_3}\left[e^{b_3 x_3} - e^{\frac{a_1-a_3}{b_3-b_1}b_3}\right].\end{aligned} \tag{2.16}$$

Thus, to sample from g_n we choose a region $[x_i, x_{i+1}]$ proportional to ω_i, generate $U \sim \mathcal{U}_{[0,1]}$ and then take

$$X = x_i + U(x_{i+1} - x_i.)$$

The right panel of Figure 2.8 shows the good agreement between the histogram and density of g_n. (See Problems 2.37 and 2.38 for generating the g_n corresponding to $\pi(b|\mathbf{x}, \mathbf{y}, a)$, and Problems 9.7 and 9.8 for full Gibbs samplers.) ||

2.5 Problems

2.1 Check the uniform random number generator on your computer:
(a) Generate 1,000 uniform random variables and make a histogram
(b) Generate uniform random variables $(X_1, \ldots, X_n)$ and plot the pairs (X_i, X_{i+1}) to check for autocorrelation.

2.2 (a) Generate a binomial $\mathcal{B}in(n, p)$ random variable with $n = 25$ and $p = .2$. Make a histogram and compare it to the binomial mass function, and to the R binomial generator.
(b) Generate 5,000 *logarithmic series* random variables with mass function

$$P(X = x) = \frac{-(1-p)^x}{x \log p}, \quad x = 1, 2, \ldots \quad 0 < p < 1.$$

Make a histogram and plot the mass function.

2.3 In each case generate the random variables and compare to the density function
(a) Normal random variables using a Cauchy candidate in Accept–Reject;
(b) Gamma $\mathcal{G}a(4.3, 6.2)$ random variables using a Gamma Ga(4, 7);
(c) Truncated normal: Standard normal truncated to $(2, \infty)$.

2.4 The arcsine distribution was discussed in Example 2.2.
(a) Show that the *arcsine distribution*, with density $f(x) = 1/\pi\sqrt{x(1-x)}$, is invariant under the transform $y = 1 - x$, that is, $f(x) = f(y)$.
(b) Show that the uniform distribution $\mathcal{U}_{[0,1]}$ is invariant under the "tent" transform,

$$D(x) = \begin{cases} 2x & \text{if } x \le 1/2 \\ 2(1-x) & \text{if } x > 1/2. \end{cases}$$

(c) As in Example 2.2, use both the arcsine and "tent" distributions in the dynamic system $X_{n+1} = D(X_n)$ to generate 100 uniform random variables. Check the properties with marginal histograms, and plots of the successive iterates.
(d) The tent distribution can have disastrous behavior. Given the finite representation of real numbers in the computer, show that the sequence (X_n) will converge to a fixed value, as the tent function progressively eliminates the last decimals of X_n. (For example, examine what happens when the sequence starts at a value of the form $1/2^n$.)

2.5 For each of the following distributions, calculate the explicit form of the distribution function and show how to implement its generation starting from a uniform random variable: (*a*) exponential; (*b*) double exponential; (*c*) Weibull; (*d*) Pareto; (*e*) Cauchy; (*f*) extreme value; (*g*) arcsine.

2.6 Referring to Example 2.7:
(a) Show that if $U \sim \mathcal{U}_{[0,1]}$, then $X = -\log U/\lambda \sim \mathcal{E}xp(\lambda)$.
(b) Verify the distributions in (2.2).
(c) Show how to generate an $\mathcal{F}_{m,n}$ random variable, where both m and n are even integers.
(d) Show that if $U \sim \mathcal{U}_{[0,1]}$, then $X = \log \frac{u}{1-u}$ is a Logistic(0, 1) random variable. Show also how to generate a Logistic(μ, β) random variable.

2.7 Establish the properties of the *Box–Muller algorithm* of Example 2.8. If U_1 and U_2 are iid $\mathcal{U}_{[0,1]}$, show that:
(a) The transforms

$$X_1 = \sqrt{-2\log(U_1)}\,\cos(2\pi U_2)\,, \qquad X_2 = \sqrt{-2\log(U_1)}\,\sin(2\pi U_2)\,,$$

are iid $\mathcal{N}(0, 1)$.
(b) The polar coordinates are distributed as

$$r^2 = X_1^2 + X_2^2 \sim \chi_2^2\,,$$
$$\theta = \arctan\frac{X_1}{X_2} \sim \mathcal{U}[0, 2\pi].$$

(c) Establish that $\exp(-r^2/2) \sim \mathcal{U}[0, 1]$, and so r^2 and θ can be simulated directly.

2.8 (Continuation of Problem 2.7)
(a) Show that an alternate version of the Box–Muller algorithm is
Algorithm A.8 –Box–Muller (2)–

1. Generate [A.8]
$$U_1, U_2 \sim \mathcal{U}([-1, 1])$$
until $S = U_1^2 + U_2^2 < 1$.
2. Define $Z = \sqrt{-2\log(S)/S}$ and take
$$X_1 = Z\,U_1, \qquad X_2 = Z\,U_2.$$

(*Hint:* Show that (U_1, U_2) is uniform on the unit sphere and that X_1 and X_2 are independent.)
(b) Give the average number of generations in 1. and compare with the original Box–Muller algorithm [*A*.3] on a small experiment.
(c) Examine the effect of not constraining (U_1, U_2) to the unit circle.

2.9 Show that the following version of the Box–Muller algorithm produces one normal variable and compare the execution time with both versions [*A*.3] and [*A*.8]:

3

Monte Carlo Integration

> Cadfael had heard the words without hearing them and enlightenment fell on him so dazzlingly that he stumbled on the threshold.
> —Ellis Peter, *The Heretic's Apprentice*

While Chapter 2 focussed on developing techniques to produce random variables by computer, this chapter introduces the central concept of Monte Carlo methods, that is, taking advantage of the availability of computer generated random variables to approximate univariate and multidimensional integrals. In Section 3.2, we introduce the basic notion of Monte Carlo approximations as a byproduct of the Law of Large Numbers, while Section 3.3 highlights the universality of the approach by stressing the versatility of the representation of an integral as an expectation.

3.1 Introduction

Two major classes of numerical problems that arise in statistical inference are *optimization* problems and *integration* problems. (An associated problem, that of *implicit equations*, can often be reformulated as an optimization problem.) Although optimization is generally associated with the likelihood approach, and integration with the Bayesian approach, these are not strict classifications, as shown by Examples 1.5 and 1.15, and Examples 3.1, 3.2 and 3.3, respectively.

Examples 1.1–1.15 have also shown that it is not always possible to derive explicit probabilistic models and that it is even less possible to analytically compute the estimators associated with a given paradigm (maximum likelihood, Bayes, method of moments, etc.). Moreover, other statistical methods, such as *bootstrap* methods (see Note 1.6.2), although unrelated to the Bayesian

approach, may involve the integration of the empirical cdf. Similarly, alternatives to standard likelihood, such as *marginal* likelihood, may require the integration of the nuisance parameters (Barndorff-Nielsen and Cox 1994).

Although many calculations in Bayesian inference require integration, this is not always the case. Integration is clearly needed when the Bayes estimators are posterior expectations (see Section 1.3 and Problem 1.22), however Bayes estimators are not always posterior expectations. In general, the Bayes estimate under the loss function $\mathrm{L}(\theta,\delta)$ and the prior π is the solution of the minimization program

$$\min_{\delta} \int_{\Theta} \mathrm{L}(\theta,\delta)\, \pi(\theta)\, f(x|\theta)\, d\theta\,. \tag{3.1}$$

Only when the loss function is the quadratic function $\|\theta-\delta\|^2$ will the Bayes estimator be a posterior expectation. While some other loss functions lead to general solutions $\delta^\pi(x)$ of (3.1) in terms of $\pi(\theta|x)$ (see, for instance, Robert 1996b, 2001 for the case of *intrinsic losses*), a specific setup where the loss function is constructed by the decision-maker almost always precludes analytical integration of (3.1). This necessitates an approximate solution of (3.1) either by numerical methods or by simulation.

Thus, whatever the type of statistical inference, we are led to consider numerical solutions. The previous chapter has illustrated a number of methods for the generation of random variables with any given distribution and, hence, provides a basis for the construction of solutions to our statistical problems. Thus, just as the search for a stationary state in a dynamical system in physics or in economics can require one or several simulations of successive states of the system, statistical inference on complex models will often require the use of simulation techniques. (See, for instance, Bauwens 1984, Bauwens and Richard 1985 and Gouriéroux and Monfort 1996 for illustrations in econometrics.) We now look at a number of examples illustrating these situations before embarking on a description of simulation-based integration methods.

Example 3.1. L_1 **loss.** For $\theta\in\mathbb{R}$ and $\mathrm{L}(\theta,\delta)=|\theta-\delta|$, the Bayes estimator associated with π is the posterior median of $\pi(\theta|x)$, $\delta^\pi(x)$, which is the solution to the equation

$$\int_{\theta\le\delta^\pi(x)} \pi(\theta)\, f(x|\theta)\, d\theta = \int_{\theta\ge\delta^\pi(x)} \pi(\theta)\, f(x|\theta)\, d\theta\,. \tag{3.2}$$

In the setup of Example 1.7, that is, when $\lambda=\|\theta\|^2$ and $X\sim\mathcal{N}_p(\theta,I_p)$, this equation is quite complex, since, when using the reference prior of Example 1.12,

$$\pi(\lambda|x)\propto\lambda^{p-1/2}\int e^{-\|x-\theta\|^2/2}\prod_{i=1}^{p-2}\sin(\varphi_i)^{p-i-1}\,d\varphi_1\ldots d\varphi_{p-1}\,,$$

where $\lambda,\varphi_1,\ldots,\varphi_{p-1}$ are the polar coordinates of θ, that is, $\theta_1=\lambda\cos(\varphi_1)$, $\theta_2=\lambda\sin(\varphi_1)\cos(\varphi_2)$, ... ‖

Example 3.2. Piecewise linear and quadratic loss functions. Consider a loss function which is piecewise quadratic,

$$\mathrm{L}(\theta,\delta)=w_i(\theta-\delta)^2 \quad \text{when} \quad \theta-\delta\in[a_i,a_{i+1}),\quad \omega_i>0. \tag{3.3}$$

Differentiating the posterior expectation (3.3) shows that the associated Bayes estimator satisfies

$$\sum_i w_i\int_{a_i}^{a_{i+1}} (\theta-\delta^\pi(x))\,\pi(\theta|x)\,d\theta=0\,,$$

that is,

$$\delta^\pi(x)=\frac{\sum_i w_i\int_{a_i}^{a_{i+1}}\theta\,\pi(\theta)\,f(x|\theta)\,d\theta}{\sum_i w_i\int_{a_i}^{a_{i+1}}\pi(\theta)\,f(x|\theta)\,d\theta}\,.$$

Although formally explicit, the computation of $\delta^\pi(x)$ requires the computation of the posterior means restricted to the intervals $[a_i,a_{i+1})$ and of the posterior probabilities of these intervals.

Similarly, consider a piecewise linear loss function,

$$\mathrm{L}(\theta,\delta)=w_i|\theta-\delta| \quad \text{if} \quad \theta-\delta\in[a_i,a_{i+1}),$$

or Huber's (1972) loss function,

$$\mathrm{L}(\theta,\delta)=\begin{cases}\rho(\theta-\delta)^2 & \text{if } |\theta-\delta|<c,\\ 2\rho c\{|\theta-\delta|-c/2\} & \text{otherwise,}\end{cases}$$

where ρ and c are specified constants. Although a specific type of prior distribution leads to explicit formulas, most priors result only in integral forms of δ^π. Some of these may be quite complex. ‖

Inference based on *classical decision theory* evaluates the performance of estimators (maximum likelihood estimator, best unbiased estimator, moment estimator, etc.) through the loss imposed by the decision-maker or by the setting. Estimators are then compared through their expected losses, also called risks. In most cases, it is impossible to obtain an analytical evaluation of the risk of a given estimator, or even to establish that a new estimator (uniformly) dominates a standard estimator.

It may seem that the topic of *James–Stein* estimation is an exception to this observation, given the abundant literature on the topic. In fact, for some families of distributions (such as exponential or spherically symmetric) and some types of loss functions (such as quadratic or concave), it is possible to analytically establish domination results over the maximum likelihood estimator or unbiased estimators (see Lehmann and Casella 1998, Chapter 5 or Robert 2001, Chapter 2). Nonetheless, in these situations, estimators such as *empirical Bayes estimators*, which are quite attractive in practice, will rarely

allow for analytic expressions. This makes their evaluation under a given loss problematic.

Given a sampling distribution $f(x|\theta)$ and a conjugate prior distribution $\pi(\theta|\lambda,\mu)$, the empirical Bayes method estimates the *hyperparameters* λ and μ from the *marginal distribution*

$$m(x|\lambda,\mu) = \int f(x|\theta)\,\pi(\theta|\lambda,\mu)\,d\theta$$

by maximum likelihood. The estimated distribution $\pi(\theta|\hat{\lambda},\hat{\mu})$ is often used as in a standard Bayesian approach (that is, without taking into account the effect of the substitution) to derive a point estimator. See Searle et al. (1992, Chapter 9) or Carlin and Louis (1996) for a more detailed discussion on this approach. (We note that this approach is sometimes called *parametric* empirical Bayes, as opposed to the *nonparametric* empirical Bayes approach developed by Herbert Robbins. See Robbins 1964, 1983 or Maritz and Lwin 1989 for details.) The following example illustrates some difficulties encountered in evaluating empirical Bayes estimators (see also Example 4.12).

Example 3.3. Empirical Bayes estimator. Let X have the distribution $X \sim \mathcal{N}_p(\theta, I_p)$ and let $\theta \sim \mathcal{N}_p(\mu, \lambda I_p)$, the corresponding conjugate prior. The hyperparameter μ is often specified, and here we take $\mu = 0$. In the empirical Bayes approach, the scale hyperparameter λ is replaced by the maximum likelihood estimator, $\hat{\lambda}$, based on the marginal distribution $X \sim \mathcal{N}_p(0, (\lambda+1)I_p)$. This leads to the maximum likelihood estimator $\hat{\lambda} = (\|x\|^2/p - 1)^+$. Since the posterior distribution of θ given λ is $\mathcal{N}_p(\lambda x/(\lambda+1), \lambda I_p/(\lambda+1))$, empirical Bayes inference may be based on the pseudo-posterior $\mathcal{N}_p(\hat{\lambda}x/(\hat{\lambda}+1), \hat{\lambda}I_p/(\hat{\lambda}+1))$. If, for instance, $\|\theta\|^2$ is the quantity of interest, and if it is evaluated under a quadratic loss, the empirical Bayes estimator is

$$\begin{aligned}\delta^{eb}(x) = \mathbb{E}(\|\theta\|^2|x) &= \left(\frac{\hat{\lambda}}{\hat{\lambda}+1}\right)^2 \|x\|^2 + \frac{\hat{\lambda}p}{\hat{\lambda}+1} \\ &= \left[\left(1-\frac{p}{\|x\|^2}\right)^+\right]^2 \|x\|^2 + p\left(1-\frac{p}{\|x\|^2}\right)^+ \\ &= (\|x\|^2 - p)^+ .\end{aligned}$$

This estimator dominates both the best unbiased estimator, $\|x\|^2 - p$, and the maximum likelihood estimator based on $\|x\|^2 \sim \chi^2_p(\|\theta\|^2)$ (see Saxena and Alam 1982 and Example 1.8). However, since the proof of this second domination result is quite involved, one might first check for domination through a simulation experiment that evaluates the risk function,

$$R(\theta,\delta) = \mathbb{E}_\theta[(\|\theta\|^2 - \delta)^2] ,$$

for the three estimators. This quadratic risk is often normalized by $1/(2\|\theta\|^2 + p)$ (which does not affect domination results but ensures the existence of a minimax estimator; see Robert 2001). Problem 3.8 contains a complete solution to the evaluation of risk. ||

A general solution to the different computational problems contained in the previous examples and in those of Section 1.1 is to use simulation, of either the true or approximate distributions to calculate the quantities of interest. In the setup of Decision Theory, whether it is classical or Bayesian, this solution is natural, since risks and Bayes estimators involve integrals with respect to probability distributions. We will see in Chapter 5 why this solution also applies in the case of maximum likelihood estimation. Note that the possibility of producing an almost infinite number of random variables distributed according to a given distribution gives us access to the use of *frequentist* and *asymptotic* results much more easily than in usual inferential settings (see Serfling 1980 or Lehmann and Casella 1998, Chapter 6) where the sample size is most often fixed. One can, therefore, apply probabilistic results such as the Law of Large Numbers or the Central Limit Theorem, since they allow for an assessment of the convergence of simulation methods (which is equivalent to the deterministic bounds used by numerical approaches.)

3.2 Classical Monte Carlo Integration

Before applying our simulation techniques to more practical problems, we first need to develop their properties in some detail. This is more easily accomplished by looking at the generic problem of evaluating the integral

$$\mathbb{E}_f[h(X)] = \int_{\mathcal{X}} h(x)\,f(x)\,dx\,. \tag{3.4}$$

Based on previous developments, it is natural to propose using a sample $(X_1,\ldots,X_m)$ generated from the density f to approximate (3.4) by the empirical average[1]

$$\overline{h}_m = \frac{1}{m}\sum_{j=1}^{m} h(x_j)\,,$$

since $\overline{h}_m$ converges almost surely to $\mathbb{E}_f[h(X)]$ by the Strong Law of Large Numbers. Moreover, when h^2 has a finite expectation under f, the speed of convergence of $\overline{h}_m$ can be assessed since the variance

$$\text{var}(\overline{h}_m) = \frac{1}{m}\int_{\mathcal{X}} (h(x) - \mathbb{E}_f[h(X)])^2\,f(x)dx$$

[1] This approach is often referred to as the *Monte Carlo method*, following Metropolis and Ulam (1949). We will meet Nicolas Metropolis (1915–1999) again in Chapters 5 and 7, with the simulated annealing and MCMC methods.

can also be estimated from the sample $(X_1, \ldots, X_m)$ through

$$v_m = \frac{1}{m^2} \sum_{j=1}^{m} [h(x_j) - \overline{h}_m]^2 .$$

For m large,

$$\frac{\overline{h}_m - \mathbb{E}_f[h(X)]}{\sqrt{v_m}}$$

is therefore approximately distributed as a $\mathcal{N}(0,1)$ variable, and this leads to the construction of a convergence test and of confidence bounds on the approximation of $\mathbb{E}_f[h(X)]$.

Example 3.4. A first Monte Carlo integration. Recall the function (1.26) that we saw in Example 1.17, $h(x) = [\cos(50x) + \sin(20x)]^2$. As a first example, we look at integrating this function, which is shown in Figure 3.1 *(left)*. Although it is possible to integrate this function analytically, it is a good first test case. To calculate the integral, we generate $U_1, U_2, \ldots, U_n$ iid $\mathcal{U}(0,1)$ random variables, and approximate $\int h(x)dx$ with $\sum h(U_i)/n$. The center panel in Figure 3.1 shows a histogram of the values of $h(U_i)$, and the last panel shows the running means and standard errors. It is clear that the Monte Carlo average is converging, with value of 0.963 after 10,000 iterations. This compares favorably with the exact value of 0.965. (See Example 4.1 for a more formal monitoring of convergence.) ‖

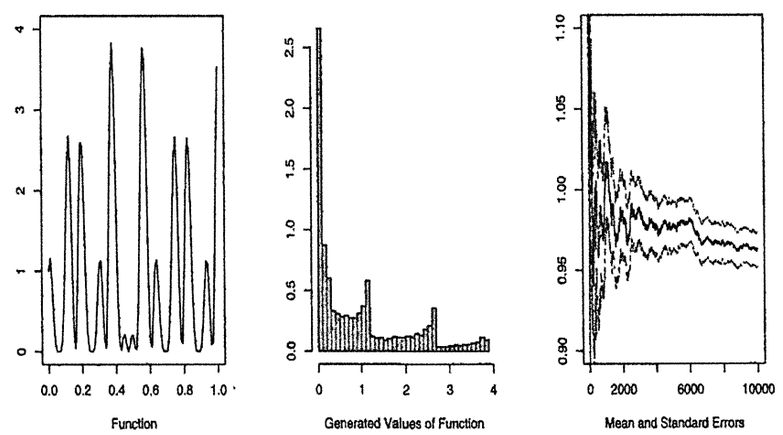


Fig. 3.1. Calculation of the integral of the function (1.26): *(left)* function (1.26), *(center)* histogram of 10,000 values $h(U_i)$, simulated using a uniform generation, and *(right)* mean ± one standard error.

n	0.0	0.67	0.84	1.28	1.65	2.32	2.58	3.09	3.72
10^2	0.485	0.74	0.77	0.9	0.945	0.985	0.995	1	1
10^3	0.4925	0.7455	0.801	0.902	0.9425	0.9885	0.9955	0.9985	1
10^4	0.4962	0.7425	0.7941	0.9	0.9498	0.9896	0.995	0.999	0.9999
10^5	0.4995	0.7489	0.7993	0.9003	0.9498	0.9898	0.995	0.9989	0.9999
10^6	0.5001	0.7497	0.8	0.9002	0.9502	0.99	0.995	0.999	0.9999
10^7	0.5002	0.7499	0.8	0.9001	0.9501	0.99	0.995	0.999	0.9999
10^8	0.5	0.75	0.8	0.9	0.95	0.99	0.995	0.999	0.9999

Table 3.1. Evaluation of some normal quantiles by a regular Monte Carlo experiment based on n replications of a normal generation. The last line gives the exact values.

The approach followed in the above example can be successfully utilized in many cases, even though it is often possible to achieve greater efficiency through numerical methods (Riemann quadrature, Simpson method, etc.) in dimension 1 or 2. The scope of application of this Monte Carlo integration method is obviously not limited to the Bayesian paradigm since, similar to Example 3.3, the performances of complex procedures can be measured in any setting where the distributions involved in the model can be simulated. For instance, we can use Monte Carlo sums to calculate a normal cumulative distribution function (even though the normal cdf can now be found in all software and many pocket calculators).

Example 3.5. Normal cdf. Since the normal cdf cannot be written in an explicit form, a possible way to construct normal distribution tables is to use simulation. Consider the generation of a sample of size n, $(x_1, \ldots, x_n)$, based on the Box–Muller algorithm $[A_4]$ of Example 2.2.2.

The approximation of

$$\Phi(t) = \int_{-\infty}^{t} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy$$

by the Monte Carlo method is thus

$$\hat{\Phi}(t) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}_{x_i \le t},$$

with (exact) variance $\Phi(t)(1-\Phi(t))/n$ (as the variables $\mathbb{I}_{x_i \le t}$ are independent Bernoulli with success probability $\Phi(t)$). For values of t around $t = 0$, the variance is thus approximately $1/4n$, and to achieve a precision of four decimals, the approximation requires on average $n = (\sqrt{2}\ 10^4)^2$ simulations, that is, 200 million iterations. Table 3.1 gives the evolution of this approximation for several values of t and shows an accurate evaluation for 100 million iterations. Note that greater (absolute) accuracy is achieved in the tails and that more efficient simulations methods could be used, as in Example 3.8 below. ‖

We mentioned in Section 3.1 the potential of this approach in evaluating estimators based on a decision-theoretic formulation. The same applies for testing, when the level of significance of a test, and its power function, cannot be easily computed, and simulation thus can provide a useful improvement over asymptotic approximations when explicit computations are impossible. The following example illustrates this somewhat different application of Monte Carlo integration.

Many tests are based on an asymptotic normality assumption as, for instance, the *likelihood ratio test*. Given H_0, a null hypothesis corresponding to r independent constraints on the parameter $\theta \in \mathbb{R}^k$, denote by $\hat{\theta}$ and $\hat{\theta}^0$ the unconstrained and constrained (under H_0) maximum likelihood estimators of θ, respectively. The likelihood ratio $\ell(\hat{\theta}|x)/\ell(\hat{\theta}^0|x)$ then satisfies

$$\log[\ell(\hat{\theta}|x)/\ell(\hat{\theta}^0|x)] = 2\,\{\log\,\ell(\hat{\theta}|x) - \log\,\ell(\hat{\theta}^0|x)\} \xrightarrow{\mathcal{L}} \chi^2_r\,, \tag{3.5}$$

when the number of observations goes to infinity (see Lehmann 1986, Section 8.8, or Gouriéroux and Monfort 1996). However, the χ^2_r approximation only holds asymptotically and, further, this convergence only holds under regularity constraints on the likelihood function (see Lehmann and Casella 1998, Chapter 6, for a full development); hence, the asymptotics may even not apply.

Example 3.6. Contingency Tables. Table 3.2 gives the results of a study comparing radiation therapy with surgery in treating cancer of the larynx.

	Cancer Controlled	Cancer not Controlled	
Surgery	21	2	23
Radiation	15	3	18
	36	5	41

Table 3.2. Comparison of cancer treatment success from surgery or radiation only (*Source:* Agresti 1996, p.50).

Typical sampling models for contingency tables may condition on both margins, one margin, or only the table total, and often the choice is based on philosophical reasons (see, for example, Agresti 1992). In this case we may argue for conditioning on the number of patients in each group, or we may just condition on the table total (there is little argument for conditioning on both margins). Happily, in many cases the resulting statistical conclusion is not dependent on this choice but, for definiteness, we will choose to condition only on the table total, $n = 41$.

Under this model, each observation X_i comes from a multinomial distribution with four cells and cell probabilities $\mathbf{p} = (p_{11}, p_{12}, p_{21}, p_{22})$, with $\sum_{ij} p_{ij} = 1$, that is,

$$X_i \sim \mathcal{M}_4(1, \mathbf{p}), \quad i = 1, \ldots, n\,.$$

If we denote by y_{ij} the number of x_i that are in cell ij, the likelihood function can be written

$$\ell(\mathbf{p}|\mathbf{y}) \propto \prod_{ij} p_{ij}^{yij}\,.$$

The null hypothesis to be tested is one of independence, which is to say that the treatment has no bearing on the control of cancer. To translate this into a parameter statement, we note that the full parameter space corresponding to Table 3.2 is

p_{11}	p_{12}	p_1
p_{21}	p_{22}	$1-p_1$
p_2	$1-p_2$	1

and the null hypothesis of independence is $H_0 : p_{11} = p_1 p_2$. The likelihood ratio statistic for testing this hypothesis is

$$\lambda(\mathbf{y}) = \frac{\max_{\mathbf{p}:p_{11}=p_1p_2}\,\ell(\mathbf{p}|\mathbf{y})}{\max_{\mathbf{p}}\,\ell(\mathbf{p}|\mathbf{y})}.$$

It is straightforward to show that the numerator maximum is attained at $\hat{p}_1 = (y_{11} + y_{12})/n$ and the denominator maximum at $\hat{p}_{ij} = y_{ij}/n$.

As mentioned above, under H_0, $-2\log\lambda$ is asymptotically distributed as χ^2_1. However, with only 41 observations, the asymptotics do not necessarily apply. One alternative is to use an exact permutation test (Mehta et al. 2000), and another alternative is to devise a Monte Carlo experiment to simulate the null distribution of $-2\log\lambda$ or equivalently of λ in order to obtain a cutoff point for a hypothesis test. If we denote this null distribution by $f_0(\lambda)$, and we are interested in an α level test, we specify α and solve for λ_α the integral equation

$$\int_0^{\lambda_\alpha} f_0(\lambda)d\lambda = 1 - \alpha\,. \tag{3.6}$$

The standard Monte Carlo approach to this problem is to generate random variables $\lambda^t \sim f_0(\lambda)$, $t = 1, \ldots, M$, then order the sample $\lambda^{(1)} \le \lambda^{(2)} \le \cdots \lambda^{(M)}$ and finally calculate the empirical $1-\alpha$ percentile $\lambda^{(\lfloor(1-\alpha)M\rfloor)}$. We then have

$$\lim_{M\to\infty} \lambda^{(\lfloor(1-\alpha)M\rfloor)} \to \lambda_\alpha\,.$$

(Note that this is a slightly unusual Monte Carlo experiment in that α is known and λ_α is not, but it is nonetheless based on the same convergence of empirical measures.)

Percentile	Monte Carlo	χ^2_1
.10	2.84	3.87
.05	3.93	4.68
.01	6.72	6.36

Table 3.3. Cutoff points for the null distribution f_0 compared to χ^2_1.

To run the Monte Carlo experiment, we need to generate values from $f_0(\lambda)$. Since this distribution is not completely specified (the parameters p_1 and p_2 can be any value in $(0,1)$), to generate a value from $f_0(\lambda)$ we generate

$$(3.7) \qquad p_i \sim \mathcal{U}(0,1), \quad i=1,2,$$
$$\mathbf{X} \sim \mathcal{M}_4(p_1p_2, p_1(1-p_2), (1-p_1)p_2, (1-p_1)(1-p_2)),$$

and calculate $\lambda(\mathbf{x})$. The results, given in Table 3.3 and Figure 3.2, show that the Monte Carlo null distribution has a slightly different shape than the χ^2_1 distribution, being slightly more concentrated around 0 but with longer tails.

The analysis of the given data is somewhat anticlimactic, as the observed value of $\lambda(\mathbf{y})$ is .594, which according to any calibration gives overwhelming support to H_0. ‖

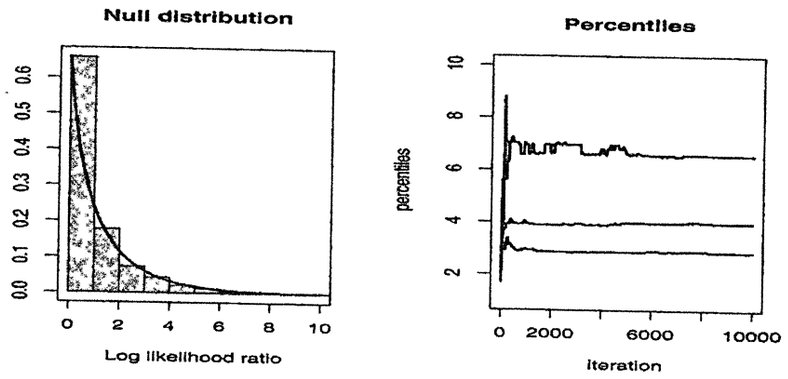


Fig. 3.2. For Example 3.6, histogram of null distribution and approximating χ^2_1 density(left panel). The right panel gives the running empirical percentiles (.90, .95, .99), from bottom to top. Notice the higher variability in the higher percentiles (10,000 simulations).

Example 3.7. Testing the number of components. A situation where the standard χ^2_r regularity conditions do not apply for the likelihood ratio test is that of the normal mixture (see Example 1.10)

$$p\,\mathcal{N}(\mu,1) + (1-p)\,\mathcal{N}(\mu+\theta,1)\,,$$

where the constraint $\theta > 0$ ensures *identifiability*. A test on the existence of a mixture cannot be easily represented in a hypothesis test since $H_0 : p = 0$ effectively eliminates the mixture and results in the identifiability problem related with $\mathcal{N}(\mu+\theta,1)$. (The inability to estimate the nuisance parameter p under H_0 results in the likelihood not satisfying the necessary regularity conditions; see Davies 1977. However, see Lehmann and Casella 1998, Section 6.6 for mixtures where it is possible to construct efficient estimators.)

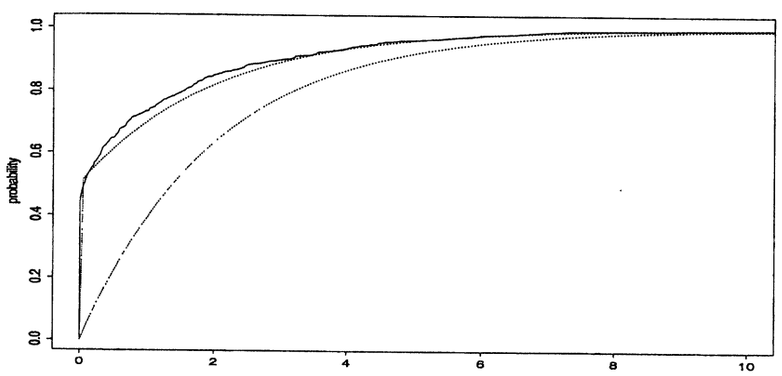


Fig. 3.3. Empirical cdf of a sample of log-likelihood ratios for the test of presence of a Gaussian mixture (solid lines) and comparison with the cdf of a χ^2_2 distribution (dotted lines, below) and with the cdf of a $.5 - .5$ mixture of a χ^2_2 distribution and of a Dirac mass at 0 (dotted lines, above) (based on 1000 simulations of a normal $\mathcal{N}(0,1)$ sample of size 100).

A slightly different formulation of the problem will allow a solution, however. If the identifiability constraint is taken to be $p \geq 1/2$ instead of $\theta > 0$, then H_0 can be represented as

$$H_0 : \quad p = 1 \quad \text{or} \quad \theta = 0\,.$$

We therefore want to determine the limiting distribution of (3.5) under this hypothesis and under a local alternative. Figure 3.3 represents the empirical cdf of $2\,\{\log\,\ell(\hat{p},\hat{\mu},\hat{\theta}|x) - \log\,\ell(\hat{\mu}^0|x)\}$ and compares it with the χ^2_2 cdf, where $\hat{p},\hat{\mu},\hat{\theta}$, and $\hat{\mu}^0$ are the respective MLEs for 1000 simulations of a normal $\mathcal{N}(0,1)$ sample of size 100. The poor agreement between the asymptotic approximation and the empirical cdf is quite obvious. Figure 3.3 also shows how the χ^2_2 approximation is improved if the limit (3.5) is replaced by an equally weighted mixture of a Dirac mass at 0 and a χ^2_2 distribution.

Note that the resulting sample of the log-likelihood ratios can also be used for inferential purposes, for instance to derive an exact test via the estimation

of the quantiles of the distribution of (3.5) under H_0 or to evaluate the power of a standard test. ‖

It may seem that the method proposed above is sufficient to approximate integrals like (3.4) in a controlled way. However, while the straightforward Monte Carlo method indeed provides good approximations of (3.4) in most regular cases, there exist more efficient alternatives which not only avoid a direct simulation from f but also can be used repeatedly for several integrals of the form (3.4). The repeated use can be for either a family of functions h or a family of densities f. In particular, the usefulness of this flexibility is quite evident in Bayesian analyses of *robustness*, of *sensitivity* (see Berger 1990, 1994), or for the computation of power functions of specific tests (see Lehmann 1986, or Gouriéroux and Monfort 1996).

3.3 Importance Sampling

3.3.1 Principles

The method we now study is called *importance sampling* because it is based on so-called *importance functions*, and although it would be more accurate to call it "weighted sampling," we will follow common usage. We start this section with a somewhat unusual example, borrowed from Ripley (1987), which shows that it may actually pay to generate from a distribution other than the distribution f of interest or, in other words, to modify the representation of an integral as an expectation against a given density. (See Note 3.6.1 for a global approach to the approximation of tail probabilities by *large deviation* techniques.)

Example 3.8. Cauchy tail probability. Suppose that the quantity of interest is the probability, p, that a Cauchy $\mathcal{C}(0,1)$ variable is larger than 2, that is,

$$p = \int_2^{+\infty} \frac{1}{\pi(1+x^2)}\, dx\,.$$

When p is evaluated through the empirical average

$$\hat{p}_1 = \frac{1}{m}\sum_{j=1}^{m} \mathbb{I}_{X_j>2}$$

of an iid sample $X_1,\ldots,X_m \sim \mathcal{C}(0,1)$, the variance of this estimator is $p(1-p)/m$ (equal to $0.127/m$ since $p = 0.15$). This variance can be reduced by taking into account the symmetric nature of $\mathcal{C}(0,1)$, since the average

$$\hat{p}_2 = \frac{1}{2m}\sum_{j=1}^{m} \mathbb{I}_{|x_j|>2}$$

has variance $p(1-2p)/2m$ equal to $0.052/m$.

The (relative) inefficiency of these methods is due to the generation of values outside the domain of interest, $[2,+\infty)$, which are, in some sense, irrelevant for the approximation of p. If p is written as

$$p = \frac{1}{2} - \int_0^2 \frac{1}{\pi(1+x^2)}\, dx\,,$$

the integral above can be considered to be the expectation of $h(X) = 2/\pi(1+X^2)$, where $X \sim \mathcal{U}_{[0,2]}$. An alternative method of evaluation for p is therefore

$$\hat{p}_3 = \frac{1}{2} - \frac{1}{m}\sum_{j=1}^{m} h(U_j)$$

for $U_j \sim \mathcal{U}_{[0,2]}$. The variance of $\hat{p}_3$ is $(\mathbb{E}[h^2] - \mathbb{E}[h]^2)/m$ and an integration by parts shows that it is equal to $0.0285/m$. Moreover, since p can be written as

$$p = \int_0^{1/2} \frac{y^{-2}}{\pi(1+y^{-2})}\, dy\,,$$

this integral can also be seen as the expectation of $\frac{1}{4}\, h(Y) = 1/2\pi(1+Y^2)$ against the uniform distribution on $[0,1/2]$ and another evaluation of p is

$$\hat{p}_4 = \frac{1}{4m}\sum_{j=1}^{m} h(Y_j)$$

when $Y_j \sim \mathcal{U}_{[0,1/2]}$. The same integration by parts shows that the variance of $\hat{p}_4$ is then $0.95\ 10^{-4}/m$.

Compared with $\hat{p}_1$, the reduction in variance brought by $\hat{p}_4$ is of order 10^{-3}, which implies, in particular, that this evaluation requires $\sqrt{1000} \approx 32$ times fewer simulations than $\hat{p}_1$ to achieve the same precision. ‖

The evaluation of (3.4) based on simulation from f is therefore not necessarily optimal and Theorem 3.12 shows that this choice is, in fact, always suboptimal. Note also that the integral (3.4) can be represented in an infinite number of ways by triplets $(\mathcal{X}, h, f)$. Therefore, the search for an optimal estimator should encompass all these possible representations (as in Example 3.8). As a side remark, we should stress that the very notion of "optimality" of a representation is quite difficult to define precisely. Indeed, as already noted in Chapter 2, the comparison of simulation methods cannot be equated with the comparison of the variances of the resulting estimators. Conception and computation times should also be taken into account. At another level, note that the optimal method proposed in Theorem 3.12 depends on the function h involved in (3.4). Therefore, it cannot be considered as optimal when several integrals related to f are simultaneously evaluated. In such cases, which often

occur in Bayesian analysis, only generic methods can be compared (that is to say, those which are independent of h).

The principal alternative to direct sampling from f for the evaluation of (3.4) is to use importance sampling, defined as follows:

Definition 3.9. The method of *importance sampling* is an evaluation of (3.4) based on generating a sample $X_1, \ldots, X_n$ from a given distribution g and approximating

$$\mathbb{E}_f[h(X)] \approx \frac{1}{m} \sum_{j=1}^{m} \frac{f(X_j)}{g(X_j)} h(X_j) \,. \tag{3.8}$$

This method is based on the alternative representation of (3.4):

$$\mathbb{E}_f[h(X)] = \int_{\mathcal{X}} h(x) \frac{f(x)}{g(x)} g(x)\, dx \,, \tag{3.9}$$

which is called the *importance sampling fundamental identity*, and the estimator (3.8) converges to (3.4) for the same reason the regular Monte Carlo estimator $\overline{h}_m$ converges, whatever the choice of the distribution g (as long as $\text{supp}(g) \supset \text{supp}(f)$).

Note that (3.9) is a very general representation that expresses the fact that a given integral is not intrinsically associated with a given distribution. Example 3.8 shows how much of an effect this choice of representation can have. Importance sampling is therefore of considerable interest since it puts very little restriction on the choice of the instrumental distribution g, which can be chosen from distributions that are easy to simulate. Moreover, the same sample (generated from g) can be used repeatedly, not only for different functions h but also for different densities f, a feature which is quite attractive for robustness and Bayesian sensitivity analyses.

Example 3.10. Exponential and log-normal comparison. Consider X as an estimator of λ, when $X \sim \mathcal{E}xp(1/\lambda)$ or when $X \sim \mathcal{LN}(0, \sigma^2)$ (with $e^{\sigma^2/2} = \lambda$, see Problem 3.11). If the goal is to compare the performances of this estimator under both distributions for the scaled squared error loss

$$L(\lambda, \delta) = (\delta - \lambda)^2/\lambda^2,$$

a *single* sample from $\mathcal{LN}(0, \sigma^2)$, $X_1, \ldots, X_T$, can be used for both purposes, the risks being evaluated by

$$\hat{R}_1 = \frac{1}{T\lambda^2} \sum_{t=1}^{T} X_t e^{-X_t/\lambda} \lambda^{-1} e^{\log(X_t)^2/2\sigma^2} \sqrt{2\pi\sigma} (X_t - \lambda)^2$$

in the exponential case and by

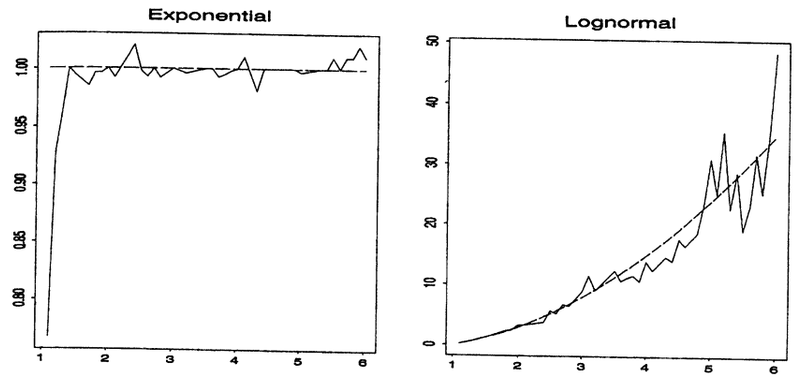


Fig. 3.4. Graph of approximate scaled squared error risks of X vs. λ for an exponential and a log-normal observation, compared with the theoretical values (dashes) for $\lambda \in [1, 6]$ (10,000 simulations).

$$\hat{R}_2 = \frac{1}{T\lambda^2} \sum_{t=1}^{T} (X_t - \lambda)^2$$

in the log-normal case. In addition, the scale nature of the parameterization allows a single sample $(Y_1^0, \ldots, Y_T^0)$ from $\mathcal{N}(0, 1)$ to be used for all σ's, with $X_t = \exp(\sigma Y_t^0)$.

The comparison of these evaluations is given in Figure 3.4 for $T = 10,000$, each point corresponding to a sample of size T simulated from $\mathcal{LN}(0, \sigma^2)$ by the above transformation. The exact values are given by 1 and $(\lambda+1)(\lambda-1)$, respectively. Note that implementing importance sampling in the opposite way offers little appeal since the weights $\exp\{-\log(X_t)^2/2\sigma^2\} \times \exp(\lambda X_t)/X_t$ have infinite variance (see below). The graph of the risk in the exponential case is then more stable than for the original sample from the log-normal distribution. ‖

We close this section by revisiting a previous example with a new twist.

Example 3.11. Small tail probabilities. In Example 3.5 we calculated normal tail probabilities with Monte Carlo sums, and found the method to work well. However, the method breaks down if we need to go too far into the tail. For example, if $Z \sim \mathcal{N}(0, 1)$, and we are interested in the probability $P(Z > 4.5)$ (which we know is very small), we could simulate $Z^{(i)} \sim \mathcal{N}(0, 1)$ for $i = 1, \ldots M$ and calculate

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

If we do this, a value of $M = 10,000$ usually produces all zeros of the indicator function.

Of course, the problem is that we are calculating the probability of a very rare event, and naïve simulation will need a lot of iterations to get a reasonable answer. However, with importance sampling we can greatly improve our accuracy.

Let $Y \sim \mathcal{TE}(4.5, 1)$, an exponential distribution (left) truncated at 4.5 with scale 1, with density

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

If we now simulate from f_Y and use importance sampling, we obtain (see Problem 3.16)

$$P(Z > 4.5) \approx \frac{1}{M} \sum_{i=1}^{M} \frac{\varphi(Y^{(i)})}{f_Y(Y^{(i)})} \mathbb{I}(Y^{(i)} > 4.5) = .000003377 \,. \quad \|$$

3.3.2 Finite Variance Estimators

Although the distribution g can be almost any density for the estimator (3.8) to converge, there are obviously some choices that are better than others, and it is natural to try to compare different distributions g for the evaluation of (3.4). First, note that, while (3.8) does converge almost surely to (3.4), its variance is finite only when the expectation

$$\mathbb{E}_g \left[h^2(X) \frac{f^2(X)}{g^2(X)} \right] = \mathbb{E}_f \left[h^2(X) \frac{f(X)}{g(X)} \right] = \int_{\mathcal{X}} h^2(x) \, \frac{f^2(x)}{g(x)} \, dx < \infty \,.$$

Thus, instrumental distributions with tails lighter than those of f (that is, those with unbounded ratios f/g) are not appropriate for importance sampling. In fact, in these cases, the variances of the corresponding estimators (3.8) will be infinite for many functions h. More generally, if the ratio f/g is unbounded, the weights $f(x_j)/g(x_j)$ will vary widely, giving too much importance to a few values x_j. This means that the estimator (3.8) may change abruptly from one iteration to the next one, even after many iterations. Conversely, distributions g with thicker tails than f ensure that the ratio f/g does not cause the divergence of $\mathbb{E}_f[h^2 f/g]$. In particular, Geweke (1989) mentions two types of sufficient conditions:

(a) $f(x)/g(x) < M \;\; \forall x \in \mathcal{X}$ and $\text{var}_f(h) < \infty$;
(b) $\mathcal{X}$ is compact, $f(x) < F$ and $g(x) > \varepsilon \;\; \forall x \in \mathcal{X}$.

These conditions are quite restrictive. In particular, $f/g < M$ implies that the Accept–Reject algorithm [A.4] also applies. (A comparison between the two approaches is given in Section 3.3.3.)

An alternative to (3.8) which addresses the finite variance issue, and generally yields a more stable estimator, is to use

$$\frac{\sum_{j=1}^{m} h(x_j) \, f(x_j)/g(x_j)}{\sum_{j=1}^{m} f(x_j)/g(x_j)} \,, \tag{3.10}$$

where we have replaced m with the sum of the weights. Since $(1/m) \sum_{j=1}^{m} f(x_j)/g(x_j)$ converges to 1 as $m \to \infty$, this estimator also converges to $\mathbb{E}_f h(X)$ by the Strong Law of Large Numbers.. Although this estimator is biased, the bias is small, and the improvement in variance makes it a preferred alternative to (3.8) (see also Lemma 4.3). In fact, Casella and Robert (1998) have shown that the weighted estimator (3.10) may perform better (when evaluated under squared error loss) in some settings. (See also Van Dijk and Kloeck 1984.) For instance, when h is nearly constant, (3.10) is close to this value, while (3.8) has a higher variation since the sum of the weights is different from one.

Among the distributions g leading to finite variances for the estimator (3.8), it is, in fact, possible to exhibit the optimal distribution corresponding to a given function h and a fixed distribution f, as stated by the following result of Rubinstein (1981); see also Geweke (1989).

Theorem 3.12. *The choice of g that minimizes the variance of the estimator (3.8) is*

$$g^*(x) = \frac{|h(x)| \, f(x)}{\int_{\mathcal{X}} |h(z)| \, f(z) \, dz} \,.$$

Proof. First note that

$$\text{var} \left[\frac{h(X)f(X)}{g(X)} \right] = \mathbb{E}_g \left[\frac{h^2(X)f^2(X)}{g^2(X)} \right] - \left(\mathbb{E}_g \left[\frac{h(X)f(X)}{g(X)} \right] \right)^2 ,$$

and the second term does not depend on g. So, to minimize variance, we only need minimize the first term. From Jensen's inequality it follows that

$$\mathbb{E}_g \left[\frac{h^2(X)f^2(X)}{g^2(X)} \right] \geq \left(\mathbb{E}_g \left[\frac{|h(X)|f(X)}{g(X)} \right] \right)^2 = \left(\int |h(x)| f(x) dx \right)^2 ,$$

which provides a lower bound that is independent of the choice of g. It is straightforward to verify that this lower bound is attained by choosing $g = g^*$. □

This optimality result is rather formal since, when $h(x) > 0$, the optimal choice $g^*(x)$ requires the knowledge of $\int h(x) f(x) dx$, the integral of interest! A practical alternative taking advantage of Theorem 3.12 is to use the estimator (3.10) as

$$\frac{\sum_{j=1}^{m} h(x_j) \, f(x_j)/g(x_j)}{\sum_{j=1}^{m} f(x_j)/g(x_j)} = \frac{\sum_{j=1}^{m} h(x_j)|h(x_j)|^{-1}}{\sum_{j=1}^{m} |h(x_j)|^{-1}} , \tag{3.11}$$

where $x_j \sim g \propto |h|f$. Note that the numerator is the number of times $h(x_j)$ is positive minus the number of times it is negative. In particular, when h is positive, (3.11) is the *harmonic mean*. Unfortunately, the optimality of Theorem 3.12 does not transfer to (3.11), which is biased and may exhibit severe instability.[2]

From a practical point of view, Theorem 3.12 suggests looking for distributions g for which $|h|f/g$ is almost constant with finite variance. It is important to note that although the finite variance constraint is not necessary for the convergence of (3.8) and of (3.11), importance sampling performs quite poorly when

$$\int \frac{f^2(x)}{g(x)}\, dx = +\infty\,, \tag{3.12}$$

whether in terms of behavior of the estimator (high-amplitude jumps, instability of the path of the average, slow convergence) or of comparison with direct Monte Carlo methods. Distributions g such that (3.12) occurs are therefore not recommended.

The next two examples show that importance sampling methods can bring considerable improvement over naïve Monte Carlo estimates when implemented with care. However, they can encounter disastrous performances and produce extremely poor estimates when the variance conditions are not met.

Example 3.13. Student's t distribution. Consider $X \sim \mathcal{T}(\nu, \theta, \sigma^2)$, with density

$$f(x) = \frac{\Gamma((\nu+1)/2)}{\sigma\sqrt{\nu\pi}\,\Gamma(\nu/2)} \left(1 + \frac{(x-\theta)^2}{\nu\sigma^2}\right)^{-(\nu+1)/2}$$

Without loss of generality, we take $\theta = 0$ and $\sigma = 1$. We choose the quantities of interest to be $\mathbb{E}_f[h_i(X)]$ $(i = 1, 2, 3)$, with

$$h_1(x) = \sqrt{\left|\frac{x}{1-x}\right|}, \quad h_2(x) = x^5 \mathbb{I}_{[2.1,\infty[}(x), \quad h_3(x) = \frac{x^5}{1+(x-3)^2}\, \mathbb{I}_{x\geq 0}\,.$$

Obviously, it is possible to generate directly from f. Importance sampling alternatives are associated here with a Cauchy $\mathcal{C}(0,1)$ distribution and a normal $\mathcal{N}(0, \nu/(\nu-2))$ distribution (scaled so that the variance is the same as $\mathcal{T}(\nu, \theta, \sigma^2)$). The choice of the normal distribution is not expected to be efficient, as the ratio

$$\frac{f^2(x)}{g(x)} \propto \frac{e^{x^2(\nu-2)/2\nu}}{[1+x^2/\nu]^{(\nu+1)}}$$

does not have a finite integral. However, this will give us an opportunity to study the performance of importance sampling in such a situation. On the other hand, the $\mathcal{C}(0,1)$ distribution has larger tails than f and ensures that the variance of f/g is finite.

[2] In fact, the optimality only applies to the numerator, while another sequence should be used to better approximate the denominator.

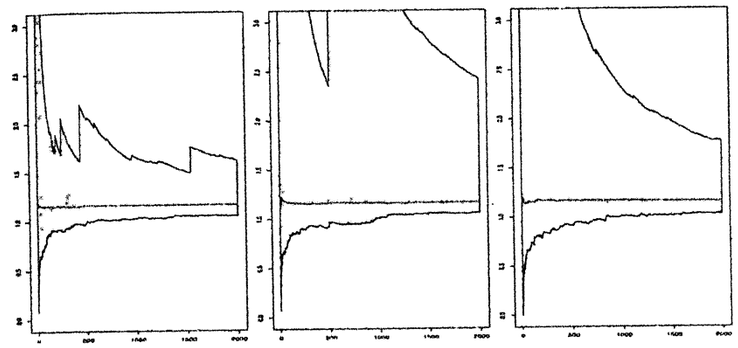


Fig. 3.5. Empirical range of three series of estimators of $\mathbb{E}_f[|X/(1-X)|^{1/2}]$ for $\nu = 12$ and 500 replications: sampling from f *(left)*, importance sampling with a Cauchy instrumental distribution *(center)* and importance sampling with normal importance distribution *(right)*. Average of the 500 series in overlay.

Figure 3.5 illustrates the performances of the three corresponding estimators for the function h_1 when $\nu = 12$ by representing the range of 500 series over 2000 iterations. The average of these series is quite stable over iterations and does not depend on the choice of the importance function, while the range exhibits wide jumps for all three. This phenomenon is due to the fact that the function h_1 has a singularity at $x = 1$ such that h_1^2 is not integrable under f but also such that none of the two other importance sampling estimators has a finite variance (Problem 3.20)! Were we to repeat this experiment with 5000 series rather than 500 series, we would then see larger ranges. There is thus no possible comparison between the three proposals in this case, since they all are inefficient. An alternative choice devised purposely for this function h_1 is to choose g such that $(1-x)g(x)$ is better behaved in $x = 1$. If we take for instance the double Gamma distribution folded at 1, that is, the distribution of X symmetric around 1 such that

$$|X - 1| \sim \mathcal{G}a(\alpha, 1)\,,$$

the ratio

$$h_1(x)\frac{f^2(x)}{g(x)} \propto \sqrt{x}\, f^2(x)\, |1-x|^{1-\alpha-1}\, exp|1-x|$$

is integrable around $x = 1$ when $\alpha < 1$. Obviously, the exponential part creates problems at ∞ and leads once more to an infinite variance, but it has much less influence on the stability of the estimator, as shown in Figure 3.6.

Since both h_2 and h_3 have restricted supports, we could benefit by having the instrumental distributions take this information into account. In the case

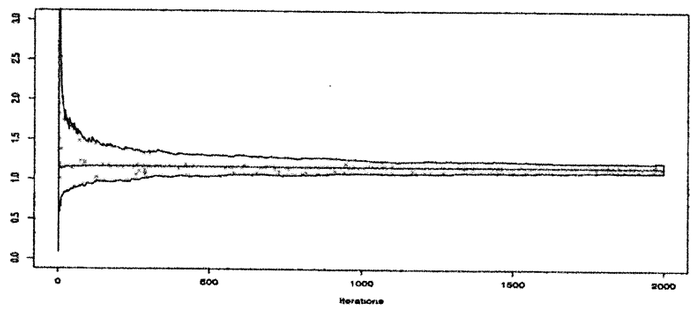


Fig. 3.6. Empirical range of the importance sampling estimator of $\mathbb{E}_f[|X/(1-X)|^{1/2}]$ for $\nu = 12$ and 500 replications based on the double Gamma $\mathcal{G}a(\alpha, 1)$ distribution folded at 1 when $\alpha = .5$. Average of the 500 series in overlay.

of h_2, a uniform distribution on $[0, 1/2.1]$ is reasonable, since the expectation $\mathbb{E}_f[h_2(X)]$ can be written as

$$\int_0^{1/2.1} u^{-7}\, f(1/u)\, du = \frac{1}{2.1} \int_0^{1/2.1} 2.1\, u^{-7}\, f(1/u)\, du\,,$$

as in Example 3.8. The corresponding importance sampling estimator is then

$$\delta_2 = \frac{1}{2.1m} \sum_{j=1}^{m} U_j^{-7} f(1/U_j)\,,$$

where the U_j's are iid $\mathcal{U}([0, 1/2.1])$. Figure 3.7 shows the improvement brought by this choice, with the estimator δ_2 converging to the true value after only a few hundred iterations. The importance sampling estimator associated with the Cauchy distribution is also quite stable, but it requires more iterations to achieve the same precision. Both of the other estimators (which are based on the true distribution and the normal distribution, respectively) fluctuate around the exact value with high-amplitude jumps, because their variance is infinite.

In the case of h_3, a reasonable candidate for the instrumental distribution is $g(x) = \exp(-x)\mathbb{I}_x \geq 0$, leading to the estimation of

$$\begin{aligned}\mathbb{E}_f[h_3(X)] &= \int_0^{\infty} \frac{x^5}{1+(x-3)^2}\, f(x)\, dx \\ &= \int_0^{\infty} \frac{x^5 e^x}{1+(x-3)^2}\, f(x)\, e^{-x}\, dx\end{aligned}$$

by

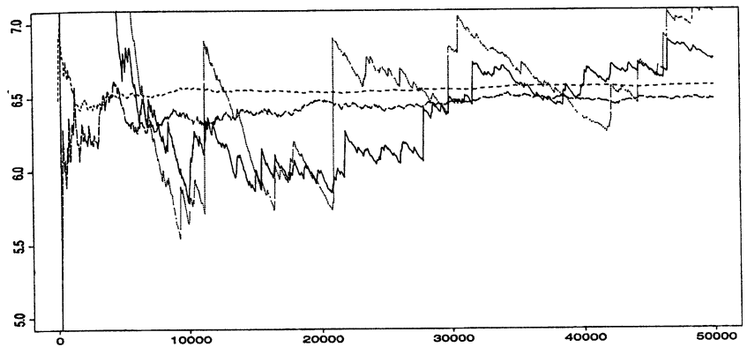


Fig. 3.7. Convergence of four estimators of $\mathbb{E}_f[X^5\mathbb{I}_{X\geq 2.1}]$ for $\nu = 12$: Sampling from f (solid lines), importance sampling with Cauchy instrumental distribution (short dashes), importance sampling with uniform $\mathcal{U}([0, 1/2.1])$ instrumental distribution (long dashes) and importance sampling with normal instrumental distribution (dots). The final values are respectively 6.75, 6.48, 6.57, and 7.06, for an exact value of 6.54.

$$\frac{1}{m} \sum_{j=1}^{m} h_3(X_j)\, w(X_j)\,, \tag{3.13}$$

where the X_j's are iid $\mathcal{E}xp(1)$ and $w(x) = f(x)\exp(x)$. Figure 3.8 shows that, although this weight does not have a finite expectation under $\mathcal{T}(\nu, 0, 1)$, meaning that the variance is infinite, the estimator (3.13) provides a good approximation of $\mathbb{E}_f[h_3(X)]$, having the same order of precision as the estimation provided by the exact simulation, and greater stability. The estimator based on the Cauchy distribution is, as in the other case, stable, but its bias is, again, slow to vanish, and the estimator associated with the normal distribution once more displays large fluctuations which considerably hinder its convergence. ‖

Example 3.14. Transition matrix estimation. Consider a Markov chain with two states, 1 and 2, whose transition matrix is

$$T = \begin{pmatrix} p_1 & 1-p_1 \\ 1-p_2 & p_2 \end{pmatrix}\,,$$

that is,

$$\begin{aligned}P(X_{t+1} = 1|X_t = 1) &= 1 - P(X_{t+1} = 2|X_t = 1) = p_1, \\ P(X_{t+1} = 2|X_t = 2) &= 1 - P(X_{t+1} = 1|X_t = 2) = p_2\,.\end{aligned}$$

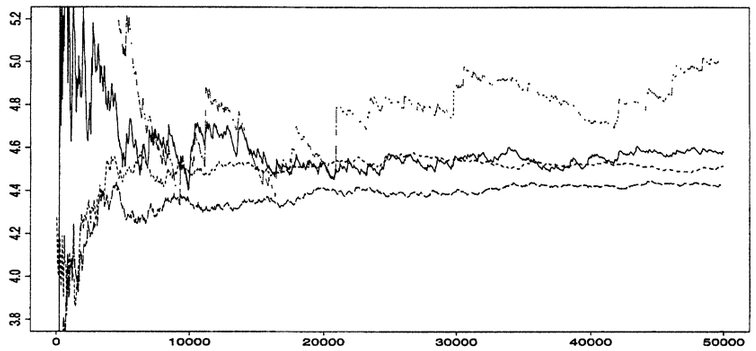


Fig. 3.8. Convergence of four estimators of $\mathbb{E}_f[h_3(X)]$: Sampling from f (solid lines), importance sampling with Cauchy instrumental distribution (short dashes), with normal instrumental distribution (dots), and with exponential instrumental distribution (long dashes). The final values after 50,000 iterations are respectively 4.58, 4.42, 4.99, and 4.52, for a true value of 4.64.

Assume, in addition, that the constraint $p_1 + p_2 < 1$ holds (see Geweke 1989 for a motivation related to continuous time processes). If the sample is $X_1, \ldots, X_m$ and the prior distribution is

$$\pi(p_1, p_2) = 2\, \mathbb{I}_{p_1+p_2<1}\ ,$$

the posterior distribution of (p_1, p_2) is

$$\pi(p_1, p_2|m_{11}, m_{12}, m_{21}, m_{22}) \propto p_1^{m_{11}}(1-p_1)^{m_{12}}(1-p_2)^{m_{21}}p_2^{m_{22}}\, \mathbb{I}_{p_1+p_2<1}\ ,$$

where m_{ij} is the number of passages from i to j, that is,

$$m_{ij} = \sum_{t=2}^{m} \mathbb{I}_{x_t=i}\mathbb{I}_{x_{t+1}=j},$$

and it follows that $\mathcal{D} = (m_{11}, \ldots, m_{22})$ is a sufficient statistic.

Suppose now that the quantities of interest are the posterior expectations of the probabilities and the associated odds:

$$h_1(p_1, p_2) = p_1,\ h_2(p_1, p_2) = p_2,\ h_3(p_1, p_2) = \frac{p_1}{1-p_1}$$

and

$$h_4(p_1, p_2) = \frac{p_2}{1-p_2}, \quad h_5(p_1, p_2) = \log\left(\frac{p_1(1-p_2)}{p_2(1-p_1)}\right)\ ,$$

respectively.

We now look at a number of ways in which to calculate these posterior expectations.

(i) The distribution $\pi(p_1, p_2|\mathcal{D})$ is the restriction of the product of two distributions $\mathcal{B}e(m_{11}+1, m_{12}+1)$ and $\mathcal{B}e(m_{22}+1, m_{21}+1)$ to the simplex $\{(p_1, p_2) :\ p_1+p_2 < 1\}$. So a reasonable first approach is to simulate these two distributions until the sum of two realizations is less than 1. Unfortunately, this naïve strategy is rather inefficient since, for the given data $(m_{11}, m_{12}, m_{21}, m_{22}) = (68, 28, 17, 4)$ we have $P^\pi(p_1 + p_2 < 1|\mathcal{D}) = 0.21$ (Geweke 1989). The importance sampling alternatives are to simulate distributions which are restricted to the simplex.

(ii) A solution inspired from the shape of $\pi(p_1, p_2|\mathcal{D})$ is a Dirichlet distribution $\mathcal{D}(m_{11}+1, m_{22}+1, m_{12}+m_{21}+1)$, with density

$$\pi_1(p_1, p_2|\mathcal{D}) \propto p_1^{m_{11}}p_2^{m_{22}}(1-p_1-p_2)^{m_{12}+m_{21}}\ .$$

However, the ratio $\pi(p_1, p_2|\mathcal{D})/\pi_1(p_1, p_2|\mathcal{D})$ is not bounded and the corresponding variance is infinite.

(iii) Geweke's (1989) proposal is to use the normal approximation to the binomial distribution, that is,

$$\begin{aligned}\pi_2(p_1, p_2|\mathcal{D}) \propto\ & \exp\{-(m_{11}+m_{12})(p_1-\hat{p}_1)^2/2\,\hat{p}_1(1-\hat{p}_1)\} \\ & \times \exp\{-(m_{21}+m_{22})(p_2-\hat{p}_2)^2/2\,\hat{p}_2(1-\hat{p}_2)\}\, \mathbb{I}_{p_1+p_2\le 1}\ ,\end{aligned}$$

where $\hat{p}_i$ is the maximum likelihood estimator of p_i, that is, $m_{ii}/(m_{ii} + m_{i(3-i)})$. An efficient way to simulate π_2 is then to simulate p_1 from the normal distribution $\mathcal{N}(\hat{p}_1, \hat{p}_1(1-\hat{p}_1)/(m_{12}+m_{11}))$ restricted to $[0, 1]$, then p_2 from the normal distribution $\mathcal{N}(\hat{p}_2, \hat{p}_2(1-\hat{p}_2)/(m_{21}+m_{22}))$ restricted to $[0, 1-p_1]$, using the method proposed by Geweke (1991) and Robert (1995b). The ratio π/π_2 then has a finite expectation under π, since (p_1, p_2) is restricted to $\{(p_1, p_2) :\ p_1 + p_2 < 1\}$.

(iv) Another possibility is to keep the distribution $\mathcal{B}(m_{11}+1, m_{12}+1)$ as the marginal distribution on p_1 and to modify the conditional distribution $p_2^{m_{22}}(1-p_2)^{m_{21}}\, \mathbb{I}_{p_2<1-p_1}$ into

$$\pi_3(p_2|p_1, \mathcal{D}) = \frac{2}{(1-p_1)^2}\, p_2\, \mathbb{I}_{p_2<1-p_1}\ .$$

The ratio $w(p_1, p_2) \propto p_2^{m_{22}-1}(1-p_2)^{m_{21}}(1-p_1)^2$ is then bounded in (p_1, p_2).

Table 3.4 provides the estimators of the posterior expectations of the functions h_j evaluated for the true distribution π (simulated the naïve way, that is, until $p_1 + p_2 < 1$) and for the three instrumental distributions π_1, π_2 and π_3. The distribution π_3 is clearly preferable to the two other instrumental distributions since it provides the same estimation as the true distribution, at a lower computational cost. Note that π_1 does worse in all cases.

Figure 3.9 describes the evolution of the estimators (3.10) of $\mathbb{E}[h_5]$ as m increases for the three instrumental distributions considered. Similarly to

Distribution	h_1	h_2	h_3	h_4	h_5
π_1	0.748	0.139	3.184	0.163	2.957
π_2	0.689	0.210	2.319	0.283	2.211
π_3	0.697	0.189	2.379	0.241	2.358
π	0.697	0.189	2.373	0.240	2.358

Table 3.4. Comparison of the evaluations of $\mathbb{E}_f[h_j]$ for the estimators (3.10) corresponding to three instrumental distributions π_i and to the true distribution π (10,000 simulations).

Table 3.4, it shows the improvement brought by the distribution π_3 upon the alternative distributions, since the precision is of the same order as the true distribution, for a significantly lower simulation cost. The jumps in the graphs of the estimators associated with π_2 and, especially, with π_1 are characteristic of importance sampling estimators with infinite variance. ‖

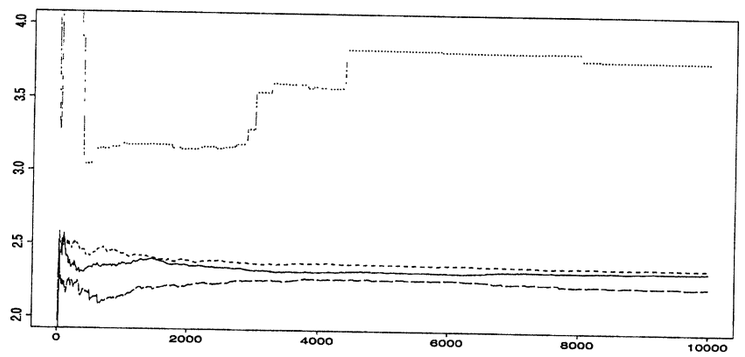


Fig. 3.9. Convergence of four estimators of $\mathbb{E}_f[h_5(X)]$ for the true distribution π (solid lines) and for the instrumental distributions π_1 (dots), π_2 (long dashes), and π_3 (short dashes). The final values after 10,000 iterations are 2.373, 3.184, 2.319, and 2.379, respectively.

We therefore see that importance sampling cannot be applied blindly. Rather, care must be taken in choosing an instrumental density as the almost sure convergence of (3.8) is only formal (in the sense that it may require an enormous number of simulations to produce an accurate approximation of the quantity of interest). These words of caution are meant to make the user aware of the problems that might be encountered if importance sampling is used when $\mathbb{E}_f[|f(X)/g(X)|]$ is infinite. (When $\mathbb{E}_f[f(X)/g(X)]$ is finite, the stakes are not so high, as convergence is more easily attained.) If the issue of

finiteness of the variance is ignored, and not detected, it may result in strong biases. For example, it can happen that the obvious divergence behavior of the previous examples does not occur. Thus, other measures, such as monitoring of the range of the weights $f(X_i)/g(X_i)$ (which are of mean 1 in all cases), can help to detect convergence problems. (See also Note 4.6.1.)

The finiteness of the ratio $\mathbb{E}_f[f(X)/g(X)]$ can be achieved by substituting a mixture distribution for the density g,

$$\rho g(x) + (1-\rho)\ell(x), \tag{3.14}$$

where ρ is close to 1 and ℓ is chosen for its heavy tails (for instance, a Cauchy or a Pareto distribution). From an operational point of view, this means that the observations are generated with probability ρ from g and with probability $1-\rho$ from ℓ. However, the mixture (g versus ℓ) does not play a role in the computation of the importance weights; that is, by construction, the estimator integrates out the uniform variable used to decide between g and ℓ. (We discuss in detail such a marginalization perspective in Section 4.2, where uniform variables involved in the simulation are integrated out in the estimator.) Obviously, (3.14) replaces $g(x)$ in the weights of (3.8) or (3.11), which can then ensure a finite variance for integrable functions h^2. Hesterberg (1998) studies the performances of this approach, called a *defensive mixture*.

3.3.3 Comparing Importance Sampling with Accept–Reject

Theorem[3] 3.12 formally solves the problem of comparing Accept–Reject and importance sampling methods, since with the exception of the constant functions $h(x) = h_0$, the optimal density g^* is always different from f. However, a more realistic comparison should also take account of the fact that Theorem 3.12 is of limited applicability in a practical setup, as it prescribes an instrumental density that depends on the function h of interest. This may not only result in a considerable increase of the computation time for every new function h (especially if the resulting instrumental density is not easy to generate from), but it also eliminates the possibility of reusing the generated sample to estimate a number of different quantities, as in Example 3.14. Now, when the Accept–Reject method is implemented with a density g satisfying $f(x) \leq Mg(x)$ for a constant $1 < M < \infty$, the density g can serve as the instrumental density for importance sampling. A positive feature is that f/g is bounded, thus ensuring finiteness of the variance for the corresponding importance sampling estimators. Bear in mind, though, that in the Accept–Reject method the resulting sample, $X_1, \ldots, X_n$, is a subsample of $Y_1, \ldots, Y_t$, where the Y_i's are simulated from g and where t is the (random) number of simulations from g required for produce the n variables from f.

[3] This section contains more specialized material and may be omitted on a first reading.

To undertake a comparison of estimation using Accept–Reject and estimation using importance sampling, it is reasonable to start with the two traditional estimators

$$\delta_1 = \frac{1}{n}\sum_{i=1}^{n} h(X_i) \qquad \text{and} \qquad \delta_2 = \frac{1}{t}\sum_{j=1}^{t} h(Y_j)\,\frac{f(Y_j)}{g(Y_j)}\,. \tag{3.15}$$

These estimators correspond to the straightforward utilization of the sample produced by Accept–Reject and to an importance sampling estimation derived from the overall sample, that is, to a recycling of the variables rejected by algorithm [A.4].[4] If the ratio f/g is only known up to a constant, δ_2 can be replaced by

$$\delta_3 = \sum_{j=1}^{t} h(Y_j)\,\frac{f(Y_j)}{g(Y_j)} \Bigg/ \sum_{j=1}^{t} \frac{f(Y_j)}{g(Y_j)}\,.$$

If we write δ_2 in the more explicit form

$$\delta_2 = \frac{n}{t}\left\{\frac{1}{n}\sum_{i=1}^{n} h(X_i)\,\frac{f(X_i)}{g(X_i)} + \frac{t-n}{n}\,\frac{1}{t-n}\sum_{i=1}^{t-n} h(Z_i)\,\frac{f(Z_i)}{g(Z_i)}\right\}\,,$$

where $\{Y_1,\ldots,Y_t\} = \{X_1,\ldots,X_n\}\cup\{Z_1,\ldots,Z_{t-n}\}$ (the Z_i's being the variables rejected by the Accept–Reject algorithm [A.4]), one might argue that, based on sample size, the variance of δ_2 is smaller than that of the estimator

$$\frac{1}{n}\sum_{i=1}^{n} h(X_i)\,\frac{f(X_i)}{g(X_i)}\,.$$

If we could apply Theorem 3.12, we could then conclude that this latter estimator dominates δ_1 (for an appropriate choice of g) and, hence, that it is better to recycle the Z_i's than to discard them. Unfortunately, this reasoning is flawed since t is a random variable, being the *stopping rule* of the Accept–Reject algorithm. The distribution of t is therefore a negative binomial distribution, $\mathcal{N}eg(n, 1/M)$ (see Problem 2.30) so $(Y_1,\ldots,Y_t)$ is not an iid sample from g. (Note that the Y_j's corresponding to the X_i's, including Y_t, have distribution f, whereas the others do not.)

The comparison between δ_1 and δ_2 can be reduced to comparing $\delta_1 = f(y_t)$ and δ_2 for $t \sim \mathcal{G}eo(1/M)$ and $n = 1$. However, even with this simplification, the comparison is quite involved (see Problem 3.34 for details), so a general comparison of the bias and variance of δ_2 with $\mathrm{var}_f(h(X))$ is difficult (Casella and Robert 1998).

While the estimator δ_2 is based on an incorrect representation of the distribution of $(Y_1,\ldots,Y_t)$, a reasonable alternative based on the correct distribution of the sample is

[4] This obviously assumes a relatively tight control on the simulation methods rather than the use of a (black box) pseudo-random generation software, which only delivers the accepted variables.

$$\delta_4 = \frac{n}{t}\,\delta_1 + \frac{1}{t}\sum_{j=1}^{t-n} h(Z_j)\,\frac{(M-1)f(Z_j)}{Mg(Z_j) - f(Z_j)}\,, \tag{3.16}$$

where the Z_j's are the elements of $(Y_1,\ldots,Y_t)$ that have been rejected. This estimator is also unbiased and the comparison with δ_1 can also be studied in the case $n = 1$; that is, through the comparison of the variances of $h(X_1)$ and of δ_4, which now can be written in the form

$$\delta_4 = \frac{1}{t}\,h(X_1) + (1-\rho)\,\frac{1}{t}\sum_{j=1}^{t-1} h(Z_j)\left(\frac{g(Z_j)}{f(Z_j)} - \rho\right)^{-1}\,.$$

Assuming again that $\mathbb{E}_f[h(X)] = 0$, the variance of δ_4 is

$$\mathrm{var}(\delta_4) = \mathbb{E}\left[\frac{t-1}{t^2}\int h^2(x)\frac{f^2(x)(M-1)}{Mg(x) - f(x)}\,dx + \frac{1}{t^2}\,\mathbb{E}_f[h^2(X)]\right]\,,$$

which is again too case-specific (that is, too dependent on f, g, and h) to allow for a general comparison.

The marginal distribution of the Z_i's from the Accept–Reject algorithm is $(Mg - f)/(M-1)$, and the importance sampling estimator δ_5 associated with this instrumental distribution is

$$\delta_5 = \frac{1}{t-n}\sum_{j=1}^{t-n}\frac{(M-1)f(Z_j)}{Mg(Z_j) - f(Z_j)}h(Z_j),$$

which allows us to write δ_4 as

$$\delta_4 = \frac{n}{t}\delta_1 + \frac{t-n}{t}\delta_5,$$

a weighted average of the usual Monte Carlo estimator and of δ_5.

According to Theorem 3.12, the instrumental distribution can be chosen such that the variance of δ_5 is lower than the variance of δ_1. Since this estimator is unbiased, δ_4 will dominate δ_1 for an appropriate choice of g. This domination result is of course as formal as Theorem 3.12, but it indicates that, for a fixed g, there exist functions h such that δ_4 improves on δ_1.

If f is only known up to the constant of integration (hence, f and M are not properly scaled), δ_4 can replaced by

$$\delta_6 = \frac{n}{t}\delta_1 + \frac{t-n}{t}\sum_{j=1}^{t-n}\frac{h(Z_j)f(Z_j)}{Mg(Z_j) - f(z_j)} \Bigg/ \sum_{j=1}^{t-n}\frac{f(Z_j)}{Mg(Z_j) - f(Z_j)}\,. \tag{3.17}$$

Although the above domination of δ_1 by δ_4 does not extend to δ_6, nonetheless, δ_6 correctly estimates constant functions while being asymptotically equivalent to δ_4. See Casella and Robert (1998) for additional domination results of δ_1 by weighted estimators.

Example 3.15. Gamma simulation. For illustrative purposes, consider the simulation of $\mathcal{G}a(\alpha,\beta)$ from the instrumental distribution $\mathcal{G}a(a,b)$, with $a=[\alpha]$ and $b=a\beta/\alpha$. (This choice of b is justified in Example 2.19 as maximizing the acceptance probability in an Accept–Reject scheme.) The ratio f/g is therefore

$$w(x)=\frac{\Gamma(a)}{\Gamma(\alpha)}\,\frac{\beta^\alpha}{b^a}\,x^{\alpha-a}e^{(b-\beta)x}\,,$$

which is bounded by

$$\begin{aligned} M &= \frac{\Gamma(a)}{\Gamma(\alpha)}\,\frac{\beta^\alpha}{b^a}\,\left(\frac{\alpha-a}{\beta-b}\right)^{\alpha-a}\,e^{-(\alpha-a)} \\ &= \frac{\Gamma(a)}{\Gamma(\alpha)}\,\exp\{\alpha(\log(\alpha)-1)-a(\log(a)-1)\}\,. \end{aligned} \tag{3.18}$$

Since the ratio $\Gamma(a)/\Gamma(\alpha)$ is bounded from above by 1, an approximate bound that can be used in the simulation is

$$M'=\exp\{a(\log(a)-1)-\alpha(\log(\alpha)-1)\}\,,$$

with $M'/M=1+\varepsilon=\Gamma(\alpha)/\Gamma([\alpha])$. In this particular setup, the estimator δ_4 is available since f/g and M are explicitly known. In order to assess the effect of the approximation (3.17), we also compute the estimator δ_6 for the following functions of interest:

$$h_1(x)=x^3,\quad h_2(x)=x\log x\,,\quad\text{and}\quad h_3(x)=\frac{x}{1+x}\,.$$

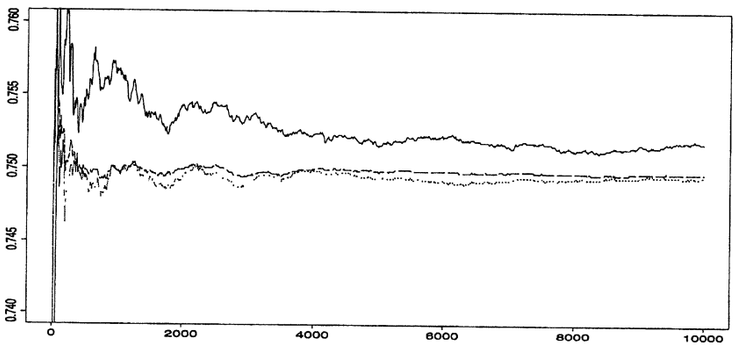


Fig. 3.10. Convergence of the estimators of $\mathbb{E}[X/(1+X)]$, δ_1 (solid lines), δ_4 (dots) and δ_6 (dashes), for $\alpha=3.7$ and $\beta=1$. The final values are respectively 0.7518, 0.7495, and 0.7497, for a true value of the expectation equal to 0.7497.

Figure 3.10 describes the convergence of the three estimators of h_3 in m for $\alpha=3.7$ and $\beta=1$ (which yields an Accept–Reject acceptance probability of $1/M=.10$). Both estimators δ_4 and δ_6 have more stable graphs than the empirical average δ_1 and they converge much faster to the theoretical expectation 0.7497, δ_6 then being equal to this value after 6,000 iterations. For $\alpha=3.08$ and $\beta=1$ (which yields an Accept–Reject acceptance probability of $1/M=.78$), Figure 3.11 illustrates the change of behavior of the three estimators of h_3 since they now converge at similar speeds. Note the proximity of δ_4 and δ_1, δ_6 again being the estimator closest to the theoretical expectation 0.7081 after 10,000 iterations.

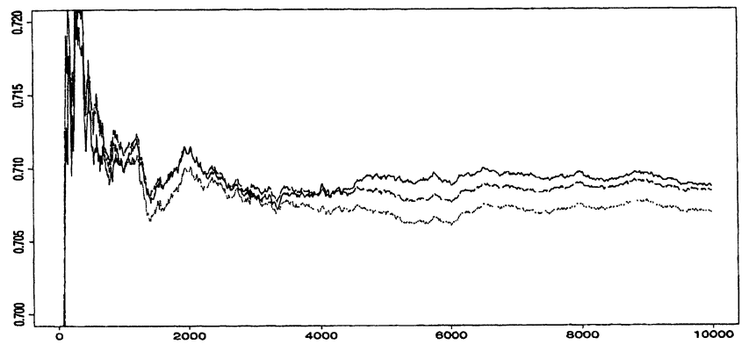


Fig. 3.11. Convergence of estimators of $\mathbb{E}[X/(1+X)]$, δ_1 (solid lines), δ_4 (dots) and δ_6 (dashes) for $\alpha=3.08$ and $\beta=1$. The final values are respectively 0.7087, 0.7069, and 0.7084, for a true value of the expectation equal to 0.7081.

Table 3.5 provides another evaluation of the three estimators in a case which is a priori very favorable to importance sampling, namely for $\alpha=3.7$. The table exhibits, in most cases, a strong domination of δ_4 and δ_6 over δ_1 and a moderate domination of δ_4 over δ_6. ‖

In contrast to the general setup of Section 3.3, δ_4 (or its approximation δ_6) can always be used in an Accept–Reject sampling setup since this estimator does not require additional simulations. It provides a second evaluation of $\mathbb{E}_f[h]$, which can be compared with the Monte Carlo estimator for the purpose of convergence assessment.

3.4 Laplace Approximations

As an alternative to simulation of integrals, we can also attempt analytic approximations. One of the oldest and most useful approximations is the integral

m	100			1000			5 000		
	δ_1	δ_4	δ_6	δ_1	δ_4	δ_6	δ_1	δ_4	δ_6
h_1	87.3	55.9	64.2	36.5	0.044	0.047	2.02	0.54	0.64
h_2	1.6	3.3	4.4	4.0	0.00	0.00	0.17	0.00	0.00
h_3	6.84	0.11	0.76	4.73	0.00	0.00	0.38	0.02	0.00

Table 3.5. Comparison of the performances of the Monte Carlo estimator (δ_1) with two importance sampling estimators (δ_4 and δ_6) under squared error loss after m iterations for $\alpha = 3.7$ and $\beta = 1$. The squared error loss is multiplied by 10^2 for the estimation of $\mathbb{E}[h_2(X)]$ and by 10^5 for the estimation of $\mathbb{E}[h_3(X)]$. The squared errors are actually the difference from the theoretical values (99.123, 5.3185, and 0.7497, respectively) and the three estimators are based on the same unique sample, which explains the lack of monotonicity (in m) of the errors. (*Source:* Casella and Robert 1998.)

Laplace approximation. It is based on the following argument: Suppose that we are interested in evaluating the integral

$$\int_A f(x|\theta)dx \tag{3.19}$$

for a fixed value of θ. (The function f needs to be non-negative and integrable; see Tierney and Kadane 1986 and Tierney et al. 1989 for extensions.). Write $f(x|\theta) = \exp\{nh(x|\theta)\}$, where n is the sample size or another parameter which can go to infinity, and use a Taylor series expansion of $h(x|\theta)$ about a point x_0 to obtain

$$\begin{aligned} h(x|\theta) \approx\ & h(x_0|\theta) + (x-x_0)h'(x_0|\theta) + \frac{(x-x_0)^2}{2!}h''(x_0|\theta) \\ & + \frac{(x-x_0)^3}{3!}h'''(x_0|\theta) + R_n(x)\,, \end{aligned} \tag{3.20}$$

where we write

$$h'(x_0|\theta) = \left.\frac{\partial h(x|\theta)}{\partial x}\right|_{x=x_0},$$

and similarly for the other terms, while the remainder $R_n(x)$ satisfies

$$\lim_{x\to x_0} R_n(x)/(x-x_0)^3 = 0.$$

Now choose $x_0 = \hat{x}_\theta$, the value that satisfies $h'(\hat{x}_\theta|\theta) = 0$ and maximizes $h(x|\theta)$ for the given value of θ. Then, the linear term in (3.20) is zero and we have the approximation

$$\int_A e^{nh(x|\theta)}dx \simeq e^{nh(\hat{x}_\theta|\theta)} \int_A e^{n\frac{(x-\hat{x}_\theta)^2}{2}h''(\hat{x}_\theta|\theta)} e^{n\frac{(x-\hat{x}_\theta)^3}{3!}h'''(\hat{x}_\theta|\theta)}dx,$$

which is valid within a neighborhood of $\hat{x}_\theta$. (See Schervish 1995, Section 7.4.3, for detailed conditions.) Note the importance of choosing the point x_0 to be a maximum.

The cubic term in the exponent is now expanded in a series around $\hat{x}_\theta$. Recall that the second order Taylor expansion of e^y around 0 is $e^y \approx 1 + y + y^2/2!$, and hence expanding $\exp\{n(x-\hat{x}_\theta)^3 h'''(\hat{x}_\theta|\theta)/3!\}$ around $\hat{x}_\theta$, we obtain the approximation

$$1 + n\frac{(x-\hat{x}_\theta)^3}{3!}h'''(\hat{x}_\theta|\theta) + n^2\frac{(x-\hat{x}_\theta)^6}{2!(3!)^2}[h'''(\hat{x}_\theta|\theta)]^2$$

and thus

$$\begin{aligned} \int_A e^{nh(x|\theta)}dx \simeq\ & e^{nh(\hat{x}_\theta|\theta)} \int_A e^{n\frac{(x-\hat{x}_\theta)^2}{2}h''(\hat{x}_\theta|\theta)} \\ & \times \left[1 + n\frac{(x-\hat{x}_\theta)^3}{3!}h'''(\hat{x}_\theta|\theta) + n^2\frac{(x-\hat{x}_\theta)^6}{2!(3!)^2}[h'''(\hat{x}_\theta|\theta)]^2 + R_n\right] dx, \end{aligned} \tag{3.21}$$

where R_n again denotes a remainder term.

Excluding R_n, we call the integral approximations in (3.21) a *first-order* approximation if it includes only the first term in the right-hand side, a *second-order* approximation if it includes the first two terms; and a *third-order* approximation if it includes all three terms.

Since the above integrand is the kernel of a normal density with mean $\hat{x}_\theta$ and variance $-1/n\,h''(\hat{x}_\theta|\theta)$, we can evaluate these expressions further. More precisely, letting $\Phi(\cdot)$ denote the standard normal cdf, and taking $A = [a,b]$, we can evaluate the integral in the first-order approximation to obtain (see Problem 3.25)

$$\begin{aligned} \int_a^b e^{nh(x|\theta)}dx \simeq\ & e^{nh(\hat{x}_\theta|\theta)}\sqrt{\frac{2\pi}{-nh''(\hat{x}_\theta|\theta)}} \\ & \times \left\{\Phi[\sqrt{-nh''(\hat{x}_\theta|\theta)}(b-\hat{x}_\theta)] - \Phi[\sqrt{-nh''(\hat{x}_\theta|\theta)}(a-\hat{x}_\theta)]\right\}. \end{aligned} \tag{3.22}$$

Example 3.16. Gamma approximation. As a simple illustration of the Laplace approximation, consider estimating a Gamma $\mathcal{G}a(\alpha, 1/\beta)$ integral, say

$$\int_a^b \frac{x^{\alpha-1}}{\Gamma(\alpha)\beta^\alpha}\, e^{-x/\beta}dx. \tag{3.23}$$

Here we have $h(x) = -\frac{x}{\beta} + (\alpha-1)\log(x)$ with second order Taylor expansion (around a point x_0)

$$\begin{aligned} h(x) &\approx h(x_0) + h'(x_0)(x-x_0) + h''(x_0)\frac{(x-x_0)^2}{2!} \\ &= -\frac{x_0}{\beta} + (\alpha-1)\log(x_0) + \left(\frac{\alpha-1}{x_0} - \frac{1}{\beta}\right)(x-x_0) - \frac{\alpha-1}{2x_0^2}(x-x_0)^2. \end{aligned}$$

Choosing $x_0 = \hat{x}_\theta = (\alpha-1)\beta$ (the mode of the density and maximizer of h) yields

$$h(x) \approx \frac{\hat{x}_\theta}{\beta} + (\alpha - 1)\log(\hat{x}_\theta) + \frac{\alpha - 1}{2\hat{x}_\theta^2}(x - \hat{x}_\theta)^2$$

Now substituting into (3.22) yields the Laplace approximation

$$\int_a^b \frac{x^{\alpha-1}}{\Gamma(\alpha)\beta^\alpha} e^{-x/\beta} dx = \hat{x}_\theta^{\alpha-1} e^{\hat{x}_\theta/\beta} \sqrt{\frac{2\pi\hat{x}_\theta^2}{\alpha - 1}} \times \left\{ \Phi\left(\sqrt{\frac{\alpha-1}{\hat{x}_\theta^2}}(b - \hat{x}_\theta) \right) - \Phi\left(\sqrt{\frac{\alpha-1}{\hat{x}_\theta^2}}(a - \hat{x}_\theta) \right) \right\}$$

For $\alpha = 5$ and $\beta = 2$, $\hat{x}_\theta = 8$, and the approximation will be best in that area. In Table 3.6 we see that although the approximation is reasonable in the central region of the density, it becomes quite unacceptable in the tails. ||

Interval	Approximation	Exact
$(7, 9)$	0.193351	0.193341
$(6, 10)$	0.375046	0.37477
$(2, 14)$	0.848559	0.823349
$(15.987, \infty)$	0.0224544	0.100005

Table 3.6. Laplace approximation of a Gamma integral for $\alpha = 5$ and $\beta = 2$.

Thus, we see both the usefulness and the limits of the Laplace approximation. In problems where Monte Carlo calculations are prohibitive because of computing time, the Laplace approximation can be useful as a guide to the solution of the problem. Also, the corresponding Taylor series can be used as a proposal density, which is particularly useful in problems where no obvious proposal exists. (See Example 7.12 for a similar situation.)

3.5 Problems

3.1 For the normal-Cauchy Bayes estimator

$$\delta(x) = \frac{\int_{-\infty}^{\infty} \frac{\theta}{1+\theta^2} e^{-(x-\theta)^2/2} d\theta}{\int_{-\infty}^{\infty} \frac{1}{1+\theta^2} e^{-(x-\theta)^2/2} d\theta}$$

(*a*) Plot the integrand and use Monte Carlo integration to calculate the integral.
(*b*) Monitor the convergence with the standard error of the estimate. Obtain three digits of accuracy with probability .95.

3.2 (Continuation of Problem 3.1)
(a) Use the Accept–Reject algorithm, with a Cauchy candidate, to generate a sample from the posterior distribution and calculate the estimator.

(b) Design a computer experiment to compare Monte Carlo error when using (*i*) the same random variables θ_i in numerator and denominator, or (*ii*) different random variables.

3.3 (a) For a standard normal random variable Z, calculate $P(Z > 2.5)$ using Monte Carlo sums based on indicator functions. How many simulated random variables are needed to obtain three digits of accuracy?
(b) Using Monte Carlo sums verify that if $X \sim \mathcal{G}(1, 1)$, $P(X > 5.3) \approx .005$. Find the exact .995 cutoff to three digits of accuracy.

3.4 (a) If $X \sim \mathcal{N}(0, \sigma^2)$, show that

$$\mathbb{E}[e^{-X^2}] = \frac{1}{\sqrt{2\sigma^2 + 1}}.$$

(b) Generalize to the case $X \sim \mathcal{N}(\mu, \sigma^2)$.

3.5 Referring to Example 3.6:
(a) Verify the maximum of the likelihood ratio statistic.
(b) Generate 5000 random variables according to (3.7), recreating the left panel of Figure 3.2. Compare this distribution to a null distribution where we *fix* null values of p_1 and p_2, for example, $(p_1, p_2) = (.25, .75)$. For a range of values of (p_1, p_2), compare the histograms both with the one from (3.7) and the χ_1^2 density. What can you conclude?

3.6 An alternate analysis to that of Example 3.6 is to treat the contingency table as two binomial distributions, one for the patients receiving surgery and one for those receiving radiation. Then the test of hypothesis becomes a test of equality of the two binomial parameters. Repeat the analysis of the data in Table 3.2 under the assumption of two binomials. Compare the results to those of Example 3.6.

3.7 A famous medical experiment was conducted by Joseph Lister in the late 1800s to examine the relationship between the use of a disinfectant, carbolic acid, and surgical success rates. The data are

	Disinfectant	
	Yes	No
Success	34	19
Failure	6	16

Using the techniques of Example 3.6, analyze these data to examine the association between disinfectant and surgical success rates. Use both the multinomial model and the two-binomial model.

3.8 Referring to Example 3.3, we calculate the expected value of $\delta^\pi(x)$ from the posterior distribution $\pi(\theta|x) \propto \|\theta\|^{-2} \exp\{-\|x - \theta\|^2/2\}$, arising from a normal likelihood and noninformative prior $\|\theta\|^{-2}$ (see Example 1.12).
(a) Show that if the quadratic loss of Example 3.3 is normalized by $1/(2\|\theta\|^2 + p)$, the resulting Bayes estimator is

$$\delta^\pi(x) = \mathbb{E}^\pi\left[\frac{\|\theta\|^2}{2\|\theta\|^2 + p} \,\middle|\, x, \lambda \right] \Big/ \mathbb{E}^\pi\left[\frac{1}{2\|\theta\|^2 + p} \,\middle|\, x, \lambda \right].$$

(b) Simulation of the posterior can be done by representing θ in polar coordinates $(\rho, \varphi_1, \varphi_2)$ $(\rho > 0, \varphi_1 \in [-\pi/2, \pi/2], \varphi_2 \in [-\pi/2, \pi/2])$, with $\theta = (\rho\cos\varphi_1,\ \rho\sin\varphi_1\cos\varphi_2,\ \rho\sin\varphi_1\sin\varphi_2)$. If we denote $\xi = \theta/\rho$, which

Solving the saddlepoint equation $\partial \log \phi_X(t)/\partial t = x$ yields the saddlepoint

$$\hat{t}(x) = \frac{-p + 2x - \sqrt{p^2 + 8\lambda x}}{4x} \tag{3.33}$$

and applying (3.30) yields the approximate density (see Hougaard 1988 or Problem 3.38). ‖

The saddlepoint can also be used to approximate the tail area of a distribution. From (3.30), we have the approximation

$$\begin{aligned} P(\bar{X} > a) &= \int_a^\infty \left(\frac{n}{2\pi K_X''(\hat{\tau}(x))}\right)^{1/2} \exp\left\{n\left[K_X\left(\hat{\tau}(x)\right) - \hat{\tau}(x)x\right]\right\} dx \\ &= \int_{\hat{\tau}(a)}^{1/2} \left(\frac{n}{2\pi}\right)^{1/2} \left[K_X''(t)\right]^{1/2} \exp\left\{n\left[K_X(t) - tK_X'(t)\right]\right\} dt, \end{aligned}$$

where we make the transformation $K_X'(t) = x$ and $\hat{\tau}(a)$ satisfies $K_X'(\hat{\tau}(a)) = a$. This transformation was noted by Daniels (1983, 1987) and allows the evaluation of the integral with only one saddlepoint evaluation.

Interval	Approximation	Renormalized approximation	Exact
$(36.225, \infty)$	0.1012	0.0996	0.10
$(40.542, \infty)$	0.0505	0.0497	0.05
$(49.333, \infty)$	0.0101	0.0099	0.01

Table 3.7. Saddlepoint approximation of a noncentral chi squared tail probability for $p = 6$ and $\lambda = 9$.

To examine the accuracy of the saddlepoint tail area, we return to the noncentral chi squared distribution of Example 3.18. Table 3.7 compares the tail areas calculated by integrating the exact density and using the regular and renormalized saddlepoints. As can be seen, the accuracy is quite impressive.

The discussion above shows only that the order of the approximation is $\mathcal{O}(1/n)$, not the $\mathcal{O}(n^{-3/2})$ that is often claimed. This better error rate is obtained by renormalizing (3.30) so that it integrates to 1.

Saddlepoint approximations for tail areas have seen much more development than given here. For example, the work of Lugannani and Rice (1980) produced a very accurate approximation that only requires the evaluation of one saddlepoint and *no* integration. There are other approaches to tail area approximations; for example, the work of Barndorff-Nielsen (1991) using ancillary statistics or the Bayes-based approximation of DiCiccio and Martin (1993). Wood et al. (1993) give generalizations of the Lugannani and Rice formula.

4

Controling Monte Carlo Variance

> The others regarded him uncertainly, none of them sure how he had arrived at such a conclusion or on how to refute it.
> —Susanna Gregory, *A Deadly Brew*

In Chapter 3, the Monte Carlo method was introduced (and discussed) as a simulation-based approach to the approximation of complex integrals. There has been a considerable body of work in this area and, while not all of it is completely relevant for this book, in this chapter we discuss the specifics of variance estimation and control. These are fundamental concepts, and we will see connections with similar developments in the realm of MCMC algorithms that are discussed in Chapters 7–12.

4.1 Monitoring Variation with the CLT

In Chapter 3, we mentioned the use of the Central Limit Theorem for assessing the convergence of a Monte Carlo estimate,

$$\overline{h}_m = \frac{1}{m}\sum_{j=1}^{m} h(X_j) \qquad X_j \sim f(x)\,,$$

to the integral of interest

$$\mathfrak{I} = \int h(x) f(x)\, dx\,. \tag{4.1}$$

Figure 3.1 *(right)* was, for example, an illustration of the use of a normal confidence interval for this assessment. It also shows the limitation of a straightforward application of the CLT to a sequence $(\overline{h}_m)$ of estimates that are *not* independent. Thus, while a given slice (that is, for a given m) in Figure 3.1 *(right)* indeed provides an asymptotically valid confidence interval, the envelope built over iterations and represented in this figure has no overall validity,

that is, another Monte Carlo sequence $(\overline{h}_m)$ will not stay in this envelope with probability 0.95. To gather a valid assessment of convergence of Monte Carlo estimators, we need to either derive the joint distribution of the sequence $(\overline{h}_m)$ or recover independence by running several sequences in parallel. The former is somewhat involved, but we will look at it in Section 4.1.2. The latter is easier to derive and more widely applicable, but greedy in computing time. However, this last "property" is a feature we will meet repeatedly in the book, namely that validation of the assessment of variation is of an higher order than convergence of the estimator itself. Namely, this requires much more computing time than validation of the pointwise convergence (except in very special cases like regeneration).

4.1.1 Univariate Monitoring

In this section we look at monitoring methods that are univariate in nature. That is, the bounds placed on the estimate at iteration k depend on the values at time k and essentially ignore any correlation structure in the iterates. We begin with an example.

Example 4.1. Monitoring with the CLT. When considering the evaluation of the integral of $h(x) = [\cos(50x) + \sin(20x)]^2$ over $[0,1]$, Figure 3.1 *(right)* provides one convergence path with a standard error evaluation. As can be seen there, the resulting confidence band is moving over iterations in a rather noncoherent fashion, that is, the band exhibits the same "wiggles" as the point estimate.[1]

If, instead, we produce parallel sequences of estimates, we get the output summarized in Figure 4.1. The main point of this illustration is that the range *and* the empirical 90% band (derived from the set of estimates at each iteration by taking the empirical 5% and 95% quantiles) are much wider than the 95% confidence interval predicted by the CLT, where the variance was computed by averaging the empirical variances over the parallel sequences. ‖

This simple example thus warns even further against the blind use of a normal approximation when repeatedly invoked over iterations with dependent estimators, simply because the normal confidence approximation only has a marginal and static validation. Using a band of estimators in parallel is obviously more costly but it provides the correct assessment on the variation of these estimators.

Example 4.2. Cauchy prior. For the problem of estimating a normal mean, it is sometimes the case that a *robust* prior is desired (see, for example, Berger 1985, Section 4.7). A degree of robustness can be achieved with a Cauchy prior, so we have the model

[1] This behavior is fairly natural when considering that, for each iteration, the confidence band is centered at the point estimate.

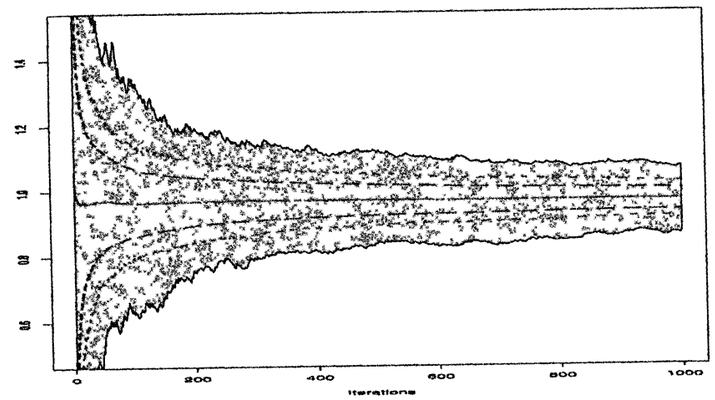


Fig. 4.1. Convergence of 1000 parallel sequences of Monte Carlo estimators of the integral of $h(x) = [\cos(50x) + \sin(20x)]^2$: The straight line is the running average of 1000 estimates, the dotted line is the empirical 90% band and the dashed line is the normal approximation 95% confidence interval. The grey shading represents the range of the entire set of estimates.

$$X \sim \mathcal{N}(\theta, 1), \quad \theta \sim \mathcal{C}(0, 1). \tag{4.2}$$

Under squared error loss, the posterior mean is

$$\delta^\pi(x) = \int_{-\infty}^{\infty} \frac{\theta}{1+\theta^2} e^{-(x-\theta)^2/2} d\theta \Big/ \int_{-\infty}^{\infty} \frac{1}{1+\theta^2} e^{-(x-\theta)^2/2} d\theta \,. \tag{4.3}$$

From the form of $\delta^\pi(x)$ we see that we can simulate iid variables $\theta_1, \cdots, \theta_m \sim \mathcal{N}(x, 1)$ and calculate

$$\hat{\delta}^\pi_m(x) = \sum_{i=1}^{m} \frac{\theta_i}{1+\theta_i^2} \Big/ \sum_{i=1}^{m} \frac{1}{1+\theta_i^2} \,. \tag{4.4}$$

The Law of Large Numbers implies that $\hat{\delta}^\pi_m(x)$ goes to $\delta^\pi(x)$ as m goes to ∞, since both the numerator and the denominator are convergent (in m). ‖

Note the "little problem" associated with this example: computing the variance of the estimator $\hat{\delta}^\pi_m(x)$ directly "on the run" gets fairly complicated because the estimator is a ratio of estimators and the variance of a ratio is not the ratio of the variances! This is actually a problem of some importance in that ratios of estimators are very common, from importance sampling (where the weights are most often unnormalized) to the computation of Bayes factors in model choice (see Problems 4.1 and 4.2).

Consider, thus, a ratio estimator, represented under the importance sampling form

$$\delta_h^n = \sum_{i=1}^n \omega_i\, h(x_i) \Big/ \sum_{i=1}^n \omega_i$$

without loss of generality. We assume that the x_i's are realizations of random variables $X_i \sim g(y)$, where g is a candidate distribution for target f. The ω_i's are realizations of random variables W_i such that $\mathbb{E}[W_i|X_i = x] = \kappa f(x)/g(x)$, κ being an arbitrary constant (that corresponds to the lack of normalizing constants in f and g). We denote

$$S_h^n = \sum_{i=1}^n W_i h(X_i)\,, \quad S_1^n = \sum_{i=1}^n W_i\,.$$

(Note that we do not assume independence between the X_i's as in regular importance sampling.) Then, as shown in Liu (1996a) and Gåsemyr (2002), the asymptotic variance of δ_h^n can be derived in general:

Lemma 4.3. *The asymptotic variance of δ_h^n is*

$$\text{var}(\delta_h^n) = \frac{1}{n^2\kappa^2}\left(\text{var}(S_h^n) - 2\mathbb{E}^\pi[h]\,\text{cov}(S_h^n, S_1^n) + \mathbb{E}^\pi[h]^2\,\text{var}(S_1^n)\right)\,.$$

Proof. As shown in Casella and Berger (2001), the variance of a ratio X/Y of random variables can be approximated by the *delta method* (also called *Cramér-Wold's theorem*) as

$$\text{var}\left(\frac{X}{Y}\right) \approx \frac{\text{var}(X)}{\mathbb{E}[Y^2]} - 2\,\frac{\mathbb{E}[X]}{\mathbb{E}[Y^3]}\,\text{cov}(X,Y) + \frac{\mathbb{E}[X^2]}{\mathbb{E}[Y^4]}\,\text{var}(Y)\,. \tag{4.5}$$

The result then follows from straightforward computation (Problem 4.7). □

As in Liu (1996a), we can then deduce that, for the regular importance sampling estimator and the right degree of approximation, we have

$$\text{var}\delta_h^n \approx \frac{1}{n}\,\text{var}^\pi(h(X))\,\{1 + \text{var}_g(W)\}\,,$$

which evaluates the additional degree of variability due to the denominator in the importance sampling ratio. (Of course, this is a rather crude approximation, as can be seen through the fact that this variance is always higher than $\text{var}_f(h(X))$, which is the variance for an iid sample with the same size, and there exist choices of g and h where this does not hold!)

Example 4.4 (Continuation of Example 4.2). If we generate normal $\mathcal{N}(x,1)$ samples for the importance sampling approximation[2] of δ^π in (4.3),

[2] The estimator (4.4) is, formally, an importance sampler because the target distribution is the Cauchy. However, the calculation is just a straightforward Monte Carlo sum, generating the variables from $\mathcal{N}(x,1)$. This illustrates that we can consider any Monte Carlo sum an importance sampling calculation.

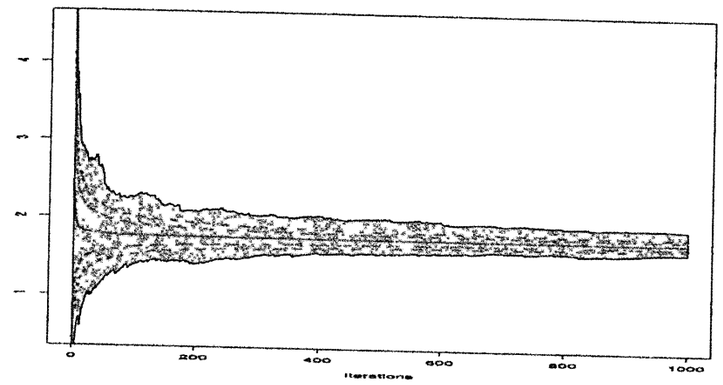


Fig. 4.2. Convergence of 1000 parallel sequences of Monte Carlo estimators for the posterior mean in the Cauchy-Normal problem when $x = 2.5$: The straight line is the running average of 1000 estimates, the dotted line is the empirical 90% band and the dashed line is the normal approximation 95% confidence interval, using the variance approximation of Lemma 4.3. The grey shading represents the range of the entire set of estimates at each iteration.

the variance approximation of Lemma 4.3 can be used to assess the variability of these estimates, but, again, the asymptotic nature of the approximation must be taken into account. Figure 4.2 compares the asymptotic variance (computed over 1,000 parallel sequences of estimators of $\delta^\pi(x)$ for $x = 2.5$) with the actual variation of the estimates, evaluated over the 1,000 parallel sequences. Even though the scales are not very different, there is once more a larger variability than predicted.

Figure 4.3 reproduces this evaluation in the case where the θ_i's are simulated from the prior $\mathcal{C}(0,1)$ distribution and associated with the importance sampling estimate[3]

$$\tilde{\delta}_h^n = \sum_{i=1}^n \theta_i\, \exp\left\{-(x-\theta_i)^2/2\right\} \Big/ \sum_{i=1}^n \exp\left\{-(x-\theta_i)^2/2\right\}\,.$$

In this case, since the corresponding functions h are bounded for both choices, the variabilities of the estimates are quite similar, with a slight advantage to the normal sampling. ‖

[3] The inversion of the roles of the $\mathcal{N}(x,1)$ and $\mathcal{C}(0,1)$ distributions illustrates once more both the ambiguity of the integral representation and the opportunities open by importance sampling.

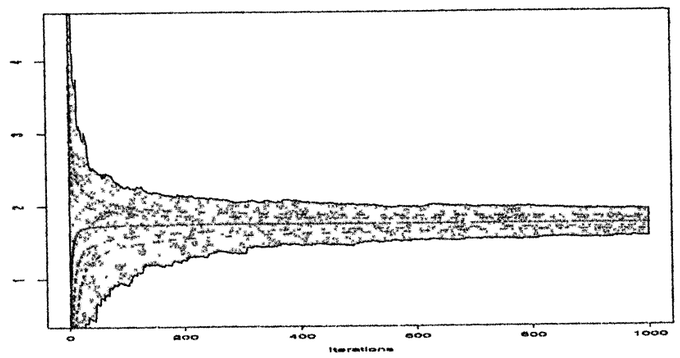


Fig. 4.3. Same plot as Figure 4.2 when the θ_i's are simulated from the prior $\mathcal{C}(0,1)$ distribution.

4.1.2 Multivariate Monitoring

As mentioned in the introduction to this chapter, one valid method for attaching variances to a mean plot, and of having a valid Central Limit Theorem, is to derive the bounds using a multivariate approach. Although the entire calculation is not difficult, and is even enlightening, it does get a bit notation-intensive, like many multivariate calculations.

Suppose that $X_1, X_2, \ldots$ is a sequence of independent (iid) random variables that are simulated with the goal of estimating $\mu = \mathbb{E}_f(X_1)$. (Without loss of generality we work with the X_i, when we are typically interested in $h(X_i)$ for some function h.) Define $\bar{X}_m = (1/m)\sum_{i=1}^m X_i$, for $m = 1, 2, \ldots, n$, where n is the number of random variables that we will simulate (typically a large number). A *running mean plot*, something that we have already seen, is a plot of $\bar{X}_m$ against m, and our goal is to put valid error bars on this plot.

For simplicity's sake, we assume that $X_i \sim \mathcal{N}(0, \sigma^2)$, independent, and want the distribution of the vector $\bar{\mathbf{X}} = (\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_n)$. Since each element of this random variable has mean μ, a simultaneous confidence interval, based on the multivariate normal distribution, will be a valid assessment of variance.

Let $\mathbf{1}$ denote the $n \times 1$ vector of ones. Then $\mathbb{E}[\bar{\mathbf{X}}] = \mathbf{1}\mu$. Moreover, it is straightforward to calculate

$$\operatorname{cov}(\bar{\mathbf{X}}_k, \bar{\mathbf{X}}_{k'}) = \frac{\sigma^2}{\max\{k, k'\}}.$$

It then follows that $\bar{\mathbf{X}} \sim \mathcal{N}_n(\mathbf{1}\mu, \Sigma)$, where

$$\Sigma = \sigma^2 \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots & \frac{1}{n} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots & \frac{1}{n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ \frac{1}{n} & \frac{1}{n} & \frac{1}{n} & \frac{1}{n} & \frac{1}{n} & \cdots & \frac{1}{n} \end{pmatrix}$$

and

$$(\bar{\mathbf{X}} - \mathbf{1}\mu)'\Sigma^{-1}(\bar{\mathbf{X}} - \mathbf{1}\mu) \sim \begin{cases} \chi^2_n & \text{if } \sigma^2 \text{ is known} \\ nF_{n,\nu} & \text{if } \sigma^2 \text{ is unknown,} \end{cases} \tag{4.6}$$

when we have an estimate $\hat{\sigma}^2 \sim \chi^2_\nu$, independent of $\bar{\mathbf{X}}$.

Our goal now is to make a running mean plot of $(\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_n)$ and, at each value of $\bar{X}_k$, attach error bars according to (4.6). At first, this seems like a calculational nightmare: Since n will typically be 5–20 *thousand* or more, we are in the position of inverting a huge matrix Σ a huge number of times. This could easily get prohibitive.

However, it turns out that the inverse of Σ is not only computable in closed form, the elements can be computed with a recursion relation *and* the matrix Σ^{-1} is tridiagonal (Problem 4.9) and is given by

$$\Sigma^{-1} = \frac{1}{\sigma^2} \begin{pmatrix} 2 & -2 & 0 & 0 & 0 & \cdots & 0 \\ -2 & 8 & -6 & 0 & 0 & \cdots & 0 \\ 0 & -6 & 18 & -12 & 0 & \cdots & 0 \\ 0 & 0 & -12 & 32 & -20 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & -n(n-1) \\ 0 & 0 & 0 & 0 & \cdots & -n(n-1) & n^2 \end{pmatrix}. \tag{4.7}$$

Finally, if we choose d_n to be the appropriate χ^2_n or $F_{n,\nu}$ cutoff point, the confidence limits on μ are given by

$$\left\{\mu : (\bar{\mathbf{x}} - \mathbf{1}\mu)'\Sigma^{-1}(\bar{\mathbf{x}} - \mathbf{1}\mu) \le d\right\},$$

and a bit of algebra will show that this is equivalent to

$$\left\{\mu : n\mu^2 - 2n\bar{x}_n\mu + \bar{\mathbf{x}}'\Sigma^{-1}\bar{\mathbf{x}} \le d\right\} = \left\{\mu : \mu \in \bar{x}_n \pm \sqrt{\bar{x}_n^2 - \frac{\mathbf{x}'\Sigma^{-1}\bar{\mathbf{x}} - d_n}{n}}\right\}. \tag{4.8}$$

To implement this procedure, we can plot the estimate of μ,

$$\bar{x}_k \pm \sqrt{\bar{x}_k^2 - \frac{\mathbf{x}'\Sigma^{-1}\bar{\mathbf{x}} - d_k}{k}} \quad \text{for } k = 1, 2, \ldots, n. \tag{4.9}$$

Figure 4.4 shows this plot, along with the univariate normal bands of Section 4.1.1. The difference is striking, and shows that the univariate bands are

extremely overoptimistic. The more conservative multivariate bands give a much more accurate picture of the confidence in the convergence.

Note that if when we implement (4.8) in a plot such as Figure 4.4, the width of the band is not dependent on $k = 1, 2, \ldots, n$, so will produce horizontal line with width equal to the final value. Instead, we suggest using (4.9), which gives the appropriate band for each value of k, showing the evolution of the band and equaling (4.8) at $k = n$.

This procedure, although dependent on the normality assumption, can be used as an approximation even when normality does not hold, with the gain from the more conservative procedure outweighing any loss due to the violation of normality.

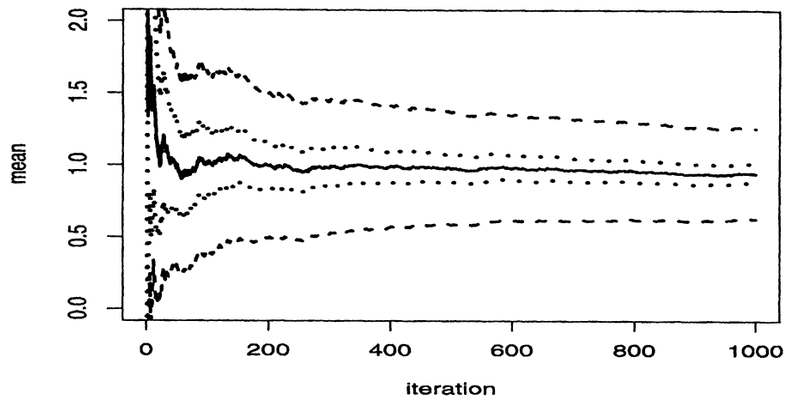


Fig. 4.4. Monte Carlo estimator of the integral of $h(x) = [\cos(50x) + \sin(20x)]^2$ (solid line). The narrow bands (grey shading) are the univariate normal approximate 95% confidence interval, and the wider bands (lighter grey) are the multivariate bands of (4.9).

4.2 Rao–Blackwellization

An approach to reduce the variance of an estimator is to use the conditioning inequality

$$\text{var}(\mathbb{E}[\delta(X)|Y]) \leq \text{var}(\delta(X)) \,,$$

sometimes called *Rao–Blackwellization* (Gelfand and Smith 1990; Liu et al. 1994; Casella and Robert 1996) because the inequality is associated with the Rao–Blackwell Theorem (Lehmann and Casella 1998), although the conditioning is not always in terms of sufficient statistics.

In a simulation context, if $\delta(X)$ is an estimator of $\mathfrak{I} = \mathbb{E}_f[h(X)]$ and if X can be simulated from the joint distribution $f(x, y)$ satisfying

$$\int f(x, y)dy = f(x),$$

the estimator $\delta^*(Y) = \mathbb{E}_f[\delta(X)|Y]$ dominates δ in terms of variance (and in squared error loss, since the bias is the same). Obviously, this result only applies in settings where $\delta^*(Y)$ can be explicitly computed.

Example 4.5. Student's t expectation. Consider the expectation of $h(x) = \exp(-x^2)$ when $X \sim \mathcal{T}(\nu, \mu, \sigma^2)$. The Student's t distribution can be simulated as a mixture of a normal distribution and of a gamma distribution by Dickey's decomposition (1968),

$$X|y \sim \mathcal{N}(\mu, \sigma^2 y) \text{ and } Y^{-1} \sim \mathcal{G}a(\nu/2, \nu/2) \Rightarrow X \sim \mathcal{T}(\nu, \mu, \sigma^2).$$

Therefore, the empirical average

$$\delta_m = \frac{1}{m} \sum_{j=1}^{m} \exp(-X_j^2)$$

can be improved upon when the X_j are parts of the sample $((X_1, Y_1), \ldots, (X_m, Y_m))$, since

$$\delta_m^* = \frac{1}{m} \sum_{j=1}^{m} \mathbb{E}[\exp(-X^2)|Y_j] = \frac{1}{m} \sum_{j=1}^{m} \frac{1}{\sqrt{2\sigma^2 Y_j + 1}} \tag{4.10}$$

is the conditional expectation when $\mu = 0$ (see Problem 4.4). Figure 4.5 provides an illustration of the difference of the convergences of δ_m and δ_m^* to $\mathbb{E}_g[\exp(-X^2)]$ for $(\nu, \mu, \sigma) = (4.6, 0, 1)$. For δ_m to have the same precision as δ_m^* requires 10 times as many simulations. ‖

Unfortunately, this conditioning method seems to enjoy a limited applicability since it involves a particular type of simulation (joint variables) and requires functions that are sufficiently regular for the conditional expectations to be explicit.

There exists, however, a specific situation where Rao–Blackwellization is always possible.[4] This is in the general setup of Accept–Reject methods, which are not always amenable to the other acceleration techniques mentioned later in Section 4.4.1 and Section 4.4.2. (Casella and Robert 1996 distinguish between *parametric* Rao–Blackwellization and *nonparametric* Rao–Blackwellization, the parametric version being more restrictive and only used in specific setups such as Gibbs sampling. See Section 7.6.2.)

[4] This part contains rather specialized material, which will not be used again in the book. It can be omitted at first reading.

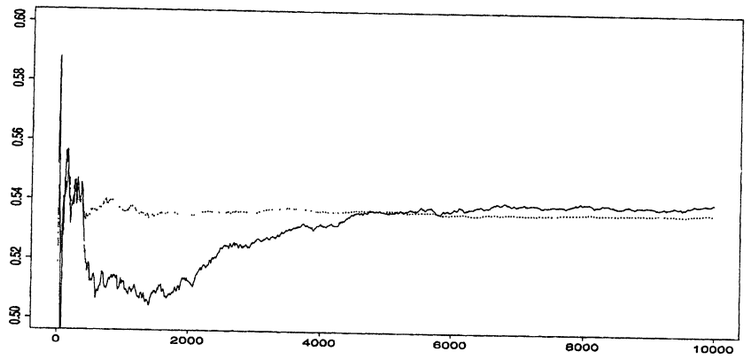


Fig. 4.5. Convergence of the estimators of $\mathbb{E}[\exp(-X^2)]$, δ_m (solid lines) and δ_m^* (dots) for $(\nu, \mu, \sigma) = (4.6, 0, 1)$. The final values are 0.5405 and 0.5369, respectively, for a true value equal to 0.5373.

Consider an Accept–Reject method based on the instrumental distribution g. If the original sample produced by the algorithm is $(X_1, \ldots, X_m)$, it can be associated with two iid samples, $(U_1, \ldots, U_N)$ and $(Y_1, \ldots, Y_N)$, with corresponding distributions $\mathcal{U}_{[0,1]}$ and g; N is then the stopping time associated with the acceptance of m variables Y_j. An estimator of $\mathbb{E}_f[h]$ based on $(X_1, \ldots, X_m)$ can therefore be written

$$\delta_1 = \frac{1}{m} \sum_{i=1}^{m} h(X_i) = \frac{1}{m} \sum_{j=1}^{N} h(Y_j)\, \mathbb{I}_{U_j \le w_j}\,,$$

with $w_j = f(Y_j)/Mg(Y_j)$. A reduction of the variance of δ_1 can be obtained by integrating out the U_i's, which leads to the estimator

$$\delta_2 = \frac{1}{m} \sum_{j=1}^{N} \mathbb{E}[\mathbb{I}_{U_j \le w_j} | N, Y_1, \ldots, Y_N]\; h(Y_j) = \frac{1}{m} \sum_{i=1}^{N} \rho_i h(Y_i),$$

where, for $i = 1, \ldots, n-1$, ρ_i satisfies

$$\begin{aligned} \rho_i &= \mathrm{P}(U_i \le w_i | N = n, Y_1, \ldots, Y_n) \\ &= w_i \frac{\sum_{(i_1,\ldots,i_{m-2})} \prod_{j=1}^{m-2} w_{i_j} \prod_{j=m-1}^{n-2} (1 - w_{i_j})}{\sum_{(i_1,\ldots,i_{m-1})} \prod_{j=1}^{m-1} w_{i_j} \prod_{j=m}^{n-1} (1 - w_{i_j})}, \end{aligned} \tag{4.11}$$

while $\rho_n = 1$. The numerator sum is over all subsets of $\{1, \ldots, i-1, i+1, \ldots, n-1\}$ of size $m-2$, and the denominator sum is over all subsets of size $m-1$. The resulting estimator δ_2 is an average over all the possible permutations of the realized sample, the permutations being weighted by

their probabilities. The Rao–Blackwellized estimator is then a function only of $(N, Y_{(1)}, \ldots, Y_{(N-1)}, Y_N)$, where $Y_{(1)}, \ldots, Y_{(N-1)}$ are the order statistics.

Although the computation of the ρ_i's may appear formidable, a recurrence relation of order n^2 can be used to calculate the estimator. Define, for $k \le m < n$,

$$S_k(m) = \sum_{(i_1,\ldots,i_k)} \prod_{j=1}^{k} w_{i_j} \prod_{j=k+1}^{m} (1 - w_{i_j}),$$

with $\{i_1, \ldots, i_m\} = \{1, \ldots, m\}$, $S_k(m) = 0$ for $k > m$, and $S_k^i(i) = S_k(i-1)$. Then we can recursively calculate

$$\begin{aligned} S_k(m) &= w_m S_{k-1}(m-1) + (1 - w_m) S_k(m-1), \\ S_k^i(m) &= w_m S_{k-1}^i(m-1) + (1 - w_m) S_k^i(m-1) \end{aligned} \tag{4.12}$$

and note that weight ρ_i of (4.11) is given by

$$\rho_i = w_i\; S_{t-2}^i(n-1)/S_{t-1}(n-1) \qquad (i < n).$$

Note that, if the random nature of N and its dependence on the sample are ignored when taking the conditional expectation, this leads to the importance sampling estimator,

$$\delta_3 = \sum_{j=1}^{N} w_j\; h(Y_j) \Big/ \sum_{j=1}^{N} w_j\;,$$

which does not necessarily improve upon δ_1 (see Section 3.3.3).

Casella and Robert (1996) establish the following proposition, showing that δ_2 can be computed and dominates δ_1. (The proof is left to Problem 4.6.)

Proposition 4.6. *The estimator $\delta_2 = \frac{1}{m} \sum_{j=1}^{N} \rho_j\; h(y_j)$ dominates the estimator δ_1 under quadratic loss.*

The computation of the weights ρ_i is obviously more costly than the derivation of the weights of the importance sampling estimator δ_3 or of the corrected estimator of Section 3.3.3. However, the recursive formula (4.12) leads to an overall simplification of the computation of the coefficients ρ_i. Nonetheless, the increase in computing time can go as high as seven times (Casella and Robert 1996), but the corresponding variance decrease is even greater (80%).

Example 4.7. (Continuation of Example 3.15) If we repeat the simulation of $\mathcal{G}a(\alpha, \beta)$ from the $\mathcal{G}a(a, b)$ distribution, with $a \in \mathbb{N}$, it is of interest to compare the estimator obtained by Rao–Blackwellization, δ_2, with the standard estimator based on the Accept–Reject sample and with the biased importance sampling estimator δ_3. Figure 4.6 illustrates the substantial improvement brought by conditioning, since δ_2 (uniformly) dominates both

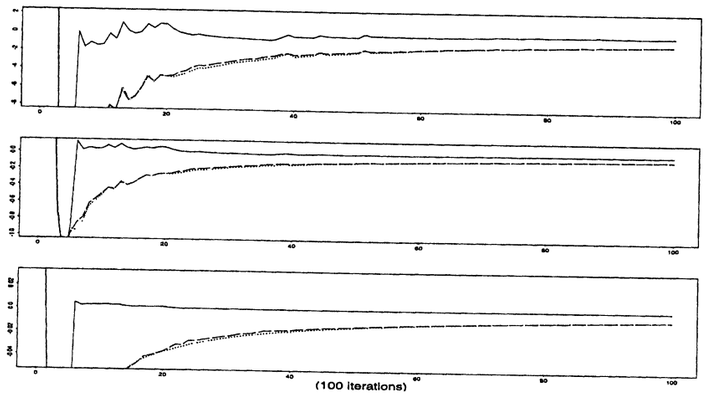


Fig. 4.6. Comparisons of the errors $\delta - \mathbb{E}[h_i(X)]$ of the Accept–Reject estimator δ_1 (long dashes), of the importance sampling estimator δ_3 (dots), and of the conditional version of δ_1, δ_2 (solid lines), for $h_1(x) = x^3$ (top), $h_2(x) = x\log(x)$ (middle), and $h_3(x) = x/(1+x)$ (bottom) and $\alpha = 3.7$, $\beta = 1$. The final errors are respectively -0.998, -0.982, and -0.077 (top), -0.053, -0.053, and -0.001 (middle), and -0.0075, -0.0074, and -0.00003 (bottom).

alternatives for the observed simulations. Note also the strong similarity between δ_1 and the importance sampling estimator, the latter failing to bring any noticeable improvement in this setup. ‖

For further discussion, estimators and examples see Casella (1996), Casella and Robert (1996). See also Perron (1999), who does a slightly different calculation, conditioning on N and the order statistics $Y_{(1)}, \ldots, Y_{(n)}$ in (4.11).

4.3 Riemann Approximations

In approximating an integral $\mathfrak{I}$ like (4.1), the simulation-based approach is justified by *probabilistic* convergence result for the empirical average

$$\frac{1}{m}\sum_{i=1}^{m} h(X_i)\,,$$

when the X_i's are simulated according to f. As briefly mentioned in Section 1.4, numerical integration (for one-dimensional integrals) is based on the *analytical* definition of the integral, namely as a limit of Riemann sums. In fact, for every sequence $(a_{i,n})_i$ $(0 \le i \le n)$ such that $a_{0,n} = a$, $a_{n,n} = b$, and $a_{i,n} - a_{i,n-1}$ converges to 0 (in n), the (Riemann) sum

$$\sum_{i=0}^{n-1} h(a_{i,n})\, f(a_{i,n})(a_{i+1,n} - a_{i,n})\,,$$

converges to (4.1) as n goes to infinity. When $\mathcal{X}$ has dimension greater than 1, the same approximation applies with a grid on the domain $\mathcal{X}$ (see Rudin 1976 for more details.)

When the two approaches are put together, the result is a Riemann sum with random steps, with the $a_{i,n}$'s simulated from f (or from an instrumental distribution g). This method was first introduced by Yakowitz et al. (1978) as *weighted Monte Carlo integration* for uniform distributions on $[0,1]^d$. In a more general setup, we call this approach *simulation by Riemann sums* or *Riemannian simulation*, following Philippe (1997a,b), although it is truly an integration method.

Definition 4.8. The method of *simulation by Riemann sums* approximates the integral $\mathfrak{I}$ by

$$\sum_{i=0}^{m-1} h(X_{(i)})\, f(X_{(i)})(X_{(i+1)} - X_{(i)})\,, \tag{4.13}$$

where $X_0, \ldots, X_m$ are iid random variables from f and $X_{(0)} \le \cdots \le X_{(m)}$ are the order statistics associated with this sample.

Suppose first that the integral $\mathfrak{I}$ can be written

$$\mathfrak{I} = \int_0^1 h(x)\, dx,$$

and that h is a differentiable function. We can then establish the following result about the validity of the Riemannian approximation. (See Problem 4.14 for the proof.)

Proposition 4.9. *Let $U = (U_0, U_1, \ldots, U_m)$ be an ordered sample from $\mathcal{U}_{[0,1]}$. If the derivative h' is bounded on $[0,1]$, the estimator*

$$\delta(U) = \sum_{i=0}^{m-1} h(U_i)(U_{i+1} - U_i) + h(0)U_0 + h(U_m)(1 - U_m)$$

has a variance of order $\mathcal{O}(m^{-2})$.

Yakowitz et al. (1978) improve on the order of the variance by symmetrizing δ into

$$\tilde{\delta} = \sum_{i=-1}^{m} (U_{i+1} - U_i)\, \frac{h(U_i) + h(U_{i+1})}{2}\,.$$

When the second derivative of h is bounded, the error of $\tilde{\delta}$ is then of order $\mathcal{O}(m^{-4})$.

Even when the additional assumption on the second derivative is not satisfied, the practical improvement brought by Riemann sums (when compared with regular Monte Carlo integration) is substantial since the magnitude of the variance decreases from m^{-1} to m^{-2}. Unfortunately, this dominance fails to extend to the case of multidimensional integrals, a phenomenon that is related to the so-called "curse of dimensionality"; that is, the subefficiency of numerical methods compared with simulation algorithms for dimensions d larger than 4 since the error is then of order $\mathcal{O}(m^{-4/d})$ (see Yakowitz et al. 1978). The intuitive reason behind this phenomenon is that a numerical approach like the Riemann sum method basically covers the entire space with a grid. When the dimension of the space increases, the number of points on the grid necessary to obtain a given precision increases too, which means, in practice, a much larger number of iterations for the same precision.

The result of Proposition 4.9 holds for an arbitrary density, due to the property that the integral $\mathfrak{I}$ can also be written as

$$\int_0^1 H(x)\,dx\,, \tag{4.14}$$

where $H(x) = h(F^-(x))$, and F^- is the generalized inverse of F, cdf of f (see Lemma 2.4). Although this is a formal representation when F^- is not available in closed form and cannot be used for simulation purposes in most cases (see Section 2.1.2), (4.14) is central to this extension of Proposition 4.9. Indeed, since

$$X_{(i+1)} - X_{(i)} = F^-(U_{i+1}) - F^-(U_i),$$

where the $X_{(i)}$'s are the order statistics of a sample from F and the U_i's are the order statistics of a sample from $\mathcal{U}_{[0,1]}$,

$$\begin{aligned}\sum_{i=0}^{m-1} & h(X_{(i)}) f(X_{(i)})(X_{(i+1)} - X_{(i)}) \\ &= \sum_{i=0}^{m-1} H(U_i) f(F^-(U_i))(F^-(U_{i+1}) - F^-(U_i)) \\ &\simeq \sum_{i=0}^{m-1} H(U_i)(U_{i+1} - U_i)\,,\end{aligned}$$

given that $(F^-)'(u) = 1/f(F^-(u))$. Since the remainder is negligible in the first-order expansion of $F^-(U_{i+1}) - F^-(U_i)$, the above Riemann sum can then be expressed in terms of uniform variables and Proposition 4.9 does apply in this setup, since the extreme terms $h(0)U_0$ and $h(U_m)(1 - U_m)$ are again of order m^{-2} (variancewise). (See Philippe 1997a,b, for more details on the convergence of (4.13) to $\mathfrak{I}$ under some conditions on the function h.)

The above results imply that the Riemann sums integration method will perform well in unidimensional setups when the density f is known. It thus provides an efficient alternative to standard Monte Carlo integration in this setting, since it does not require additional computations (although it requires keeping track of and storing all the $X_{(i)}$'s). Also, as the convergence is of a higher order, there is no difficulty in implementing the method. When f is known only up to a constant (that is, $f_0(x) \propto f(x)$), (4.13) can be replaced by

$$\frac{\sum_{i=0}^{m-1} h(X_{(i)})\, f_0(X_{(i)})(X_{(i+1)} - X_{(i)})}{\sum_{i=0}^{m-1} f_0(X_{(i)})(X_{(i+1)} - X_{(i)})}\,, \tag{4.15}$$

since both the numerator and the denominator almost surely converge. This approach thus provides, in addition, an efficient estimation method for the normalizing constant via the denominator in (4.15). Note also that when f is entirely known, the denominator converges to 1, which can be used as a convergence assessment device (see Philippe and Robert 2001, and Section 12.2.4).

Example 4.10. (Continuation of Example 3.15) When $X \sim \mathcal{G}a(3.7, 1)$, assume that $h_2(x) = x\log(x)$ is the function of interest. A sample $X_1, \ldots, X_m$ from $\mathcal{G}a(3.7, 1)$ can easily be produced by the algorithms [A.14] or [A.15] of Chapter 2 and we compare the empirical mean, δ_{1m}, with the Riemann sum estimator

$$\delta_{2m} = \frac{1}{\Gamma(3.7)} \sum_{i=1}^{m-1} h_2(X_{(i)})\, X_{(i)}^{2.7} e^{-X_{(i)}}\, (X_{(i+1)} - X_{(i)})\,,$$

which uses the known normalizing constant. Figure 4.7 clearly illustrates the difference in convergence speed between the two estimators and the much greater stability of δ_{2m}, which is close to the theoretical value 5.3185 after 3000 iterations. ‖

If the original simulation is done by importance sampling (that is, if the sample $X_1, \ldots, X_m$ is generated from an instrumental distribution g), since the integral $\mathfrak{I}$ can also be written

$$\mathfrak{I} = \int h(x)\frac{f(x)}{g(x)} g(x)dx,$$

the Riemann sum estimator (4.13) remains unchanged. Although it has similar convergence properties, the boundedness conditions on h are less explicit and, thus, more difficult to check. As in the original case, it is possible to establish an equivalent to Theorem 3.12, namely to show that $g(x) \propto |h(x)|f(x)$ is optimal (in terms of variance) (see Philippe 1997c), with the additional advantage that the normalizing constant does not need to be known, since g does not appear in (4.13).

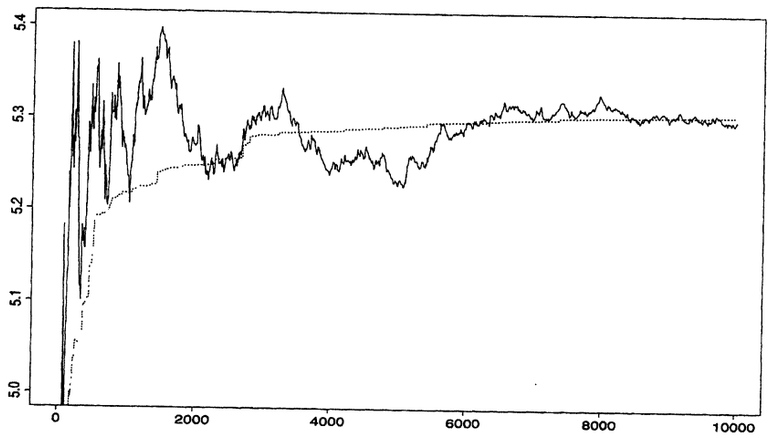


Fig. 4.7. Convergence of estimators of $\mathbb{E}[X \log(X)]$, the Riemann sum δ_{1m} (solid lines and smooth curve) and the empirical average δ_{2m} (dots and wiggly curve) for $\alpha = 3.7$ and $\beta = 1$. The final values are 5.3007 and 5.3057, respectively, for a true value of 5.31847.

Example 4.11. (Continuation of Example 3.13) If $\mathcal{T}(\nu, 0, 1)$ is simulated by importance sampling from the normal instrumental distribution $\mathcal{N}(0, \nu/(\nu-2))$, the difference between the two distributions is mainly visible in the tails. This makes the importance sampling estimator δ_{1m} very unstable (see Figures 3.5, 3.7, and 3.8). Figure 4.8 compares this estimator to the Riemann sum estimator

$$\delta_{2m} = \sum_{i=1}^{m-1} h_4(X_{(i)}) \frac{\Gamma((\nu+1)/2)}{\sqrt{\nu\pi}\,\Gamma(\nu/2)} \left[1 + X_{(i)}^2/\nu\right]^{-\frac{\nu+1}{2}} (X_{(i+1)} - X_{(i)})$$

and its normalized version

$$\delta_{3m} = \frac{\sum_{i=0}^{m-1} h_4(X_{(i)}) \left[1 + X_{(i)}^2/\nu\right]^{-\frac{\nu+1}{2}} (X_{(i+1)} - X_{(i)})}{\sum_{i=0}^{m-1} \left[1 + X_{(i)}^2/\nu\right]^{-\frac{\nu+1}{2}} (X_{(i+1)} - X_{(i)})},$$

for $h_4(X) = (1 + e^X)\, \mathbb{I}_{X \le 0}$ and $\nu = 2.3$.

We can again note the stability of the approximations by Riemann sums, the difference between δ_{2m} and δ_{3m} mainly due to the bias introduced by the approximation of the normalizing constant in δ_{3m}. For the given sample, note that δ_{2m} dominates the other estimators.

If, instead, the instrumental distribution is chosen to be the Cauchy distribution $\mathcal{C}(0, 1)$, the importance sampling estimator is much better behaved. Figure 4.9 shows that the speed of convergence of the associated estimator

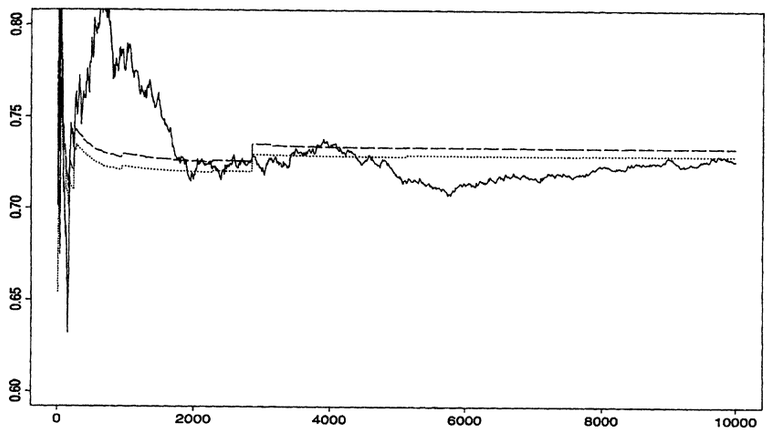


Fig. 4.8. Convergence of estimators of $\mathbb{E}_\nu[(1 + e^X)\mathbb{I}_{X \le 0}]$, δ_{1m} (solid lines), δ_{2m} (dots), and δ_{3m} (dashes), for a normal instrumental distribution and $\nu = 2.3$. The final values are respectively 0.7262, 0.7287, and 0.7329, for a true value of 0.7307.

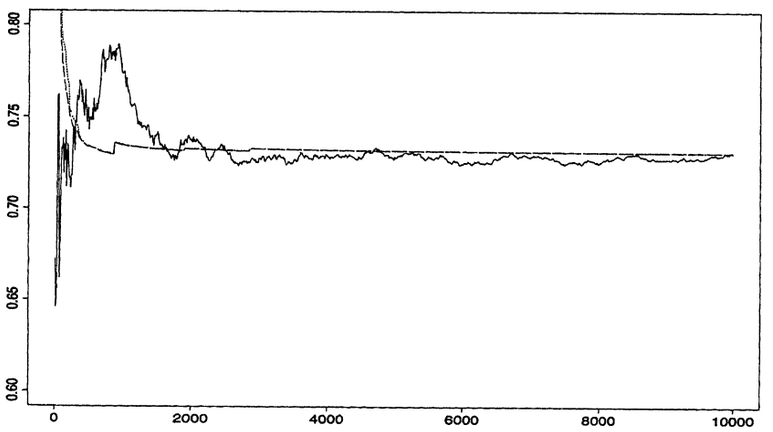


Fig. 4.9. Convergence of estimators of $\mathbb{E}_\nu[(1 + e^X)\mathbb{I}_{X \le 0}]$, δ_{1m} (solid lines), δ_{2m} (dots), and δ_{3m} (dashes) for a Cauchy instrumental distribution and $\nu = 2.3$. The two Riemann sum approximations are virtually equal except for the beginning simulations. The final values are respectively 0.7325, 0.7314, and 0.7314, and the true value is 0.7307.

is much faster than with the normal instrumental distribution. Although δ_{2m} and δ_{3m} have the same representation for $\mathcal{N}(0,\nu/(\nu-2))$ and $\mathcal{C}(0,1)$, the corresponding samples differ, and these estimators exhibit an even higher stability in this case, giving a good approximation of $\mathbb{E}[h_4(X)]$ after only a few hundred iterations. The two estimators are actually identical almost from the start, a fact which indicates how fast the denominator of δ_{3m} converges to the normalizing constant. ‖

4.4 Acceleration Methods

While the different methods proposed in Chapter 3 and in this chapter seem to require a comparison, we do not expect there to be any clear-cut domination (as was the case with the comparison between Accept–Reject and importance sampling in Section 3.3.3). Instead, we look at more global *acceleration* strategies, which are more or less independent of the simulation setup but try to exploit the output of the simulation in more efficient ways.

The acceleration methods described below can be used not only in a single implementation but also as a control device to assess the convergence of a simulation algorithm, following the argument of *parallel estimators*. For example, if $\delta_{1m},\ldots,\delta_{pm}$ are p convergent estimators of the same quantity $\mathfrak{I}$, a stopping rule for convergence is that $\delta_{1m},\ldots,\delta_{pm}$ are identical or, given a minimum precision requirement ε, that

$$\max_{1\le i<j\le p} |\delta_{im}-\delta_{jm}| < \varepsilon\,,$$

as in Section 4.1.

4.4.1 Antithetic Variables

Although the usual simulation methods lead to iid samples (or quasi-iid, see Section 2.6.2), it may actually be preferable to generate samples of correlated variables when estimating an integral $\mathfrak{I}$, as they may reduce the variance of the corresponding estimator.

A first setting where the generation of independent samples is less desirable corresponds to the comparison of two quantities which are close in value. If

$$\mathfrak{I}_1 = \int g_1(x)f_1(x)dx \quad \text{and} \quad \mathfrak{I}_2 = \int g_2(x)f_2(x)dx \tag{4.16}$$

are two such quantities, where δ_1 estimates $\mathfrak{I}_1$ and δ_2 estimates $\mathfrak{I}_2$, independently of δ_1, the variance of $(\delta_1-\delta_2)$, is then $\text{var}(\delta_1)+\text{var}(\delta_2)$, which may be too large to support a fine enough analysis on the difference $\mathfrak{I}_1-\mathfrak{I}_2$. However, if δ_1 and δ_2 are positively correlated, the variance is reduced by a factor $-2\,\text{cov}(\delta_1,\delta_2)$, which may greatly improve the analysis of the difference.

A convincing illustration of the improvement brought by correlated samples is the comparison of (regular) statistical estimators via simulation. Given a density $f(x|\theta)$ and a loss function $L(\delta,\theta)$, two estimators δ_1 and δ_2 are evaluated through their risk functions, $R(\delta_1,\theta)=\mathbb{E}[L(\delta_1,\theta)]$ and $R(\delta_2,\theta)$. In general, these risk functions are not available analytically, but they may be approximated, for instance, by a regular Monte Carlo method,

$$\hat{R}(\delta_1,\theta)=\frac{1}{m}\sum_{i=1}^{m} L(\delta_1(X_i),\theta), \qquad \hat{R}(\delta_2,\theta)=\frac{1}{m}\sum_{i=1}^{m} L(\delta_2(Y_i),\theta),$$

the X_i's and Y_i's being simulated from $f(\cdot|\theta)$. Positive correlation between $L(\delta_1(X_i),\theta)$ and $L(\delta_2(Y_i),\theta)$ then reduces the variability of the approximation of $R(\delta_1,\theta)-R(\delta_2,\theta)$.

Before we continue with the development in this section, we pause to make two elementary remarks that should be observed in any simulation comparison.

(i) First, the *same sample* $(X_1,\ldots,X_m)$ should be used in the evaluation of $R(\delta_1,\theta)$ and of $R(\delta_2,\theta)$. This repeated use of a single sample greatly improves the precision of the estimated difference $R(\delta_1,\theta)-R(\delta_2,\theta)$, as shown by the comparison of the variances of $\hat{R}(\delta_1,\theta)-\hat{R}(\delta_2,\theta)$ and of

$$\frac{1}{m}\sum_{i=1}^{m}\{L(\delta_1(X_i),\theta)-L(\delta_2(X_i),\theta)\}\,.$$

(ii) Second, *the same sample* should be used for the comparison of risks *for every value of* θ. Although this sounds like an absurd recommendation since the sample $(X_1,\ldots,X_m)$ is usually generated from a distribution depending on θ, it is often the case that the same uniform sample can be used for the generation of the X_i's for every value of θ. Also, in many cases, there exists a transformation M_θ on $\mathcal{X}$ such that if $X^0\sim f(X|\theta_0)$, $M_\theta X^0\sim f(X|\theta)$. A single sample $(X_1^0,\ldots,X_m^0)$ from $f(X|\theta_0)$ is then sufficient to produce a sample from $f(X|\theta)$ by the transform M_θ. (This second remark is somewhat tangential for the theme of this section; however, it brings significant improvement in the practical implementation of Monte Carlo methods.)

The variance reduction associated with the conservation of the underlying uniform sample is obvious in the graphs of the resulting risk functions, which then miss the irregular peaks of graphs obtained with independent samples and allow for an easier comparison of estimators. See, for instance, the graphs in Figure 3.4, which are based on samples generated independently for each value of λ. By comparison, an evaluation based on a single sample corresponding to $\lambda=1$ would give a constant risk in the exponential case.

Example 4.12. James–Stein estimation. In the case $X \sim \mathcal{N}_p(\theta, I_p)$, the transform is the location shift $M_\theta X = X + \theta - \theta_0$. When studying *positive-part James–Stein estimators*

$$\delta_a(x) = \left(1 - \frac{a}{\|x\|^2}\right)^+ x, \qquad 0 < a < 2(p-2)$$

(see Robert 2001, Chapter 2, for a motivation), the squared error risk of δ_a can be computed "explicitly," but the resulting expression involves several special functions (Robert 1988) and the approximation of the risks by simulation is much more helpful in comparing these estimators. Figure 4.10 illustrates this comparison in the case $p = 5$ and exhibits a crossing phenomenon for the risk functions in the same region; however, as shown by the inset, the crossing point for the risks of δ_a and δ_c depends on (a, c). ‖

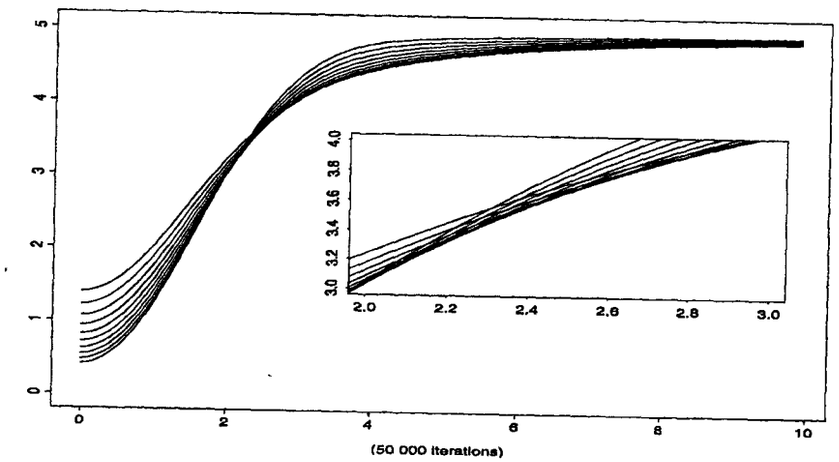


Fig. 4.10. Approximate squared error risks of truncated James–Stein estimators for a normal distribution $\mathcal{N}_5(\theta, I_5)$, as a function of $\|\theta\|$. The inset gives a magnification of the intersection zone for the risk functions.

In a more general setup, creating a strong enough correlation between δ_1 and δ_2 is rarely so simple, and the quest for correlation can result in an increase in the conception and simulation burdens, which may even have a negative overall effect on the efficiency of the analysis. Indeed, to use the same uniform sample for the generation of variables distributed from f_1 and f_2 in (4.16) is only possible when there exists a simple transformation from f_1 to f_2. For instance, if f_1 or f_2 must be simulated by Accept–Reject methods, the use of a random number of uniform variables prevents the use of a common sample.

The method of *antithetic variables* is based on the same idea that higher efficiency can be brought about by correlation. Given two samples $(X_1, \ldots, X_m)$ and $(Y_1, \ldots, Y_m)$ from f used for the estimation of

$$\mathfrak{I} = \int_{\mathbb{R}} h(x) f(x) dx \,,$$

the estimator

$$\frac{1}{2m} \sum_{i=1}^{m} [h(X_i) + h(Y_i)] \tag{4.17}$$

is more efficient than an estimator based on an iid sample of size $2m$ if the variables $h(X_i)$ and $h(Y_i)$ are *negatively correlated*. In this setting, the Y_i's are the *antithetic variables*, and it remains to develop a method for generating these variables in an optimal (or, at least, useful) way. However, the correlation between $h(X_i)$ and $h(Y_i)$ depends both on the pair (X_i, Y_i) and on the function h. (For instance, if h is even, X_i has mean 0, and $X_i = -Y_i$, X_i and Y_i are negatively correlated, but $h(X_i) = h(Y_i)$.) A solution proposed in Rubinstein (1981) is to use the uniform variables U_i to generate the X_i's and the variables $1-U_i$ to generate the Y_i's. The argument goes as follows: If $H = h \circ F^-$, $X_i = F^-(U_i)$, and $Y_i = F^-(1-U_i)$, then $h(X_i)$ and $h(Y_i)$ are negatively correlated *when H is a monotone function*. Again, such a constraint is often difficult to verify and, moreover, the technique only applies for direct transforms of uniform variables, thus excluding the Accept–Reject methods.

Geweke (1988) proposed the implementation of an inversion at the level of the X_i's by taking $Y_i = 2\mu - X_i$ when f *is symmetric around* μ. With some additional conditions on the function h, the improvement brought by

$$\frac{1}{2m} \sum_{i=1}^{m} [h(X_i) + h(2\mu - X_i)]$$

upon

$$\frac{1}{2m} \sum_{i=1}^{2m} h(X_i)$$

is quite substantial for large sample sizes m. Empirical extensions of this approach can then be used in cases where f is not symmetric, by replacing μ with the mode of f or the median of the associated distribution. Moreover, if f is unknown or, more importantly, if μ is unknown, μ can be estimated from a first sample (but caution is advised!). More general group actions can also be considered, as in Kong et al. (2003), where the authors replace the standard average by an average (over i) of the average of the $h(gx_i)$ (over the transformations g).

Example 4.13. (Continuation of Example 3.3) Assume, for the sake of illustration, that the noncentral chi squared variables $\|X_i\|^2$ are simulated

from normal random variables $X_i \sim \mathcal{N}_p(\theta, I_p)$. We can create negative correlation by using $Y_i = 2\theta - X_i$, which has a correlation of -1 with X_i, to produce a second sample, $||Y_i||^2$. However, the negative correlation does not necessarily transfer to the pairs $(h(||X_i||^2), h(||Y_i||^2))$. Figure 4.11 illustrates the behavior of (4.17) for

$$h_1(||x||^2) = ||x||^2 \qquad \text{and} \qquad h_2(||x||^2) = \mathbb{I}_{||x||^2 < ||\theta||^2 + p},$$

when $m = 500$ and $p = 4$, compared to an estimator based on an iid sample of size $2m$. As shown by the graphs in Figure 4.11, although the correlation between $h(||X_i||^2)$ and $h(||Y_i||^2 i)$ is actually positive for small values of $||\theta||^2$, the improvement brought by (4.17) over the standard average is quite impressive in the case of h_1. The setting is less clear for h_2, but the variance of the terms of (4.17) is much smaller than its independent counterpart. ‖

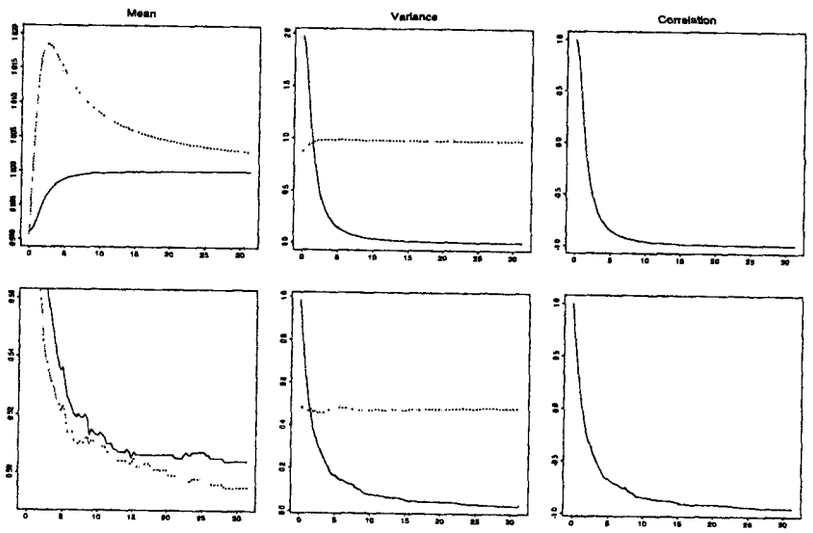


Fig. 4.11. Average of the antithetic estimator (4.17) (solid lines) against the average of an standard iid estimate (dots) for the estimation of $\mathbb{E}[h_1(||X||^2)]$ *(upper left)* and $\mathbb{E}[h_2(||X||^2)]$ *(lower left)*, along with the empirical variance of $h_1(X_i) + h_1(Y_i)$ *(upper center)* and $h_2(X_i) + h_2(Y_i)$ *(lower center)*, and the correlation between $h_1(||X_i||^2)$ and $h_1(||Y_i||^2)$ *(upper right)* and between $h_2(||X_i||^2)$ and $h_2(||Y_i||^2)$ *(lower right)*, for $m = 500$ and $p = 4$. The horizontal axis is scaled in terms of $||\theta||$ and the values in the upper left graph are divided by the true expectation, $||\theta||^2 + p$, and the values in the upper central graph are divided by $8||\theta||^2 + 4p$.

4.4.2 Control Variates

In some settings, there exist functions h_0 whose mean under f is known. For instance, if f is symmetric around μ, the mean of $h_0(X) = \mathbb{I}_{X \geq \mu}$ is $1/2$. We also saw a more general example in the case of Riemann sums with known density f, with a convergent estimator of 1. This additional information can reduce the variance of an estimator of $\mathfrak{I} = \int h(x)f(x)dx$ in the following way. If δ_1 is an estimator of $\mathfrak{I}$ and δ_3 an unbiased estimator of $\mathbb{E}_f[h_0(X)]$, consider the weighted estimator $\delta_2 = \delta_1 + \beta(\delta_3 - \mathbb{E}_f[h_0(X)])$. The estimators δ_1 and δ_2 have the same mean and

$$\text{var}(\delta_2) = \text{var}(\delta_1) + \beta^2 \,\text{var}(\delta_3) + 2\beta \,\text{cov}(\delta_1, \delta_3) \,.$$

For the optimal choice

$$\beta^* = -\frac{\text{cov}(\delta_1, \delta_3)}{\text{var}(\delta_3)} \,,$$

we have

$$\text{var}(\delta_2) = (1 - \rho_{13}^2)\,\text{var}(\delta_1) \,,$$

ρ_{13}^2 being the correlation coefficient between δ_1 and δ_3, so the control variate strategy will result in decreased variance. In particular, if

$$\delta_1 = \frac{1}{m}\sum_{i=1}^{m} h(X_i) \qquad \text{and} \qquad \delta_3 = \frac{1}{m}\sum_{i=1}^{m} h_0(X_i) \,,$$

the control variate estimator is

$$\delta_2 = \frac{1}{m}\sum_{i=1}^{m} (h(X_i) + \beta^* h_0(X_i)) - \beta^* \mathbb{E}_f[h_0(X)] \,,$$

with $\beta^* = -\text{cov}(h(X), h_0(X))/\text{var}(h_0(X))$. Note that this construction is only formal since it requires the computation of β^*. An incorrect choice of β may lead to an increased variance; that is, $\text{var}(\delta_2) > \text{var}(\delta_1)$. (However, in practice, the sign of β^* can be evaluated by a regression of the $h(x_i)$'s over the $h_0(x_i)$'s. More generally, functions with known expectations can be used as side controls in convergence diagnoses.)

Example 4.14. Control variate integration. Let $X \sim f$, and suppose that we want to evaluate

$$P(X > a) = \int_a^\infty f(x)dx.$$

The natural place to start is with $\delta_1 = \frac{1}{n}\sum_{i=1}^n \mathbb{I}(X_i > a)$, where the X_i's are iid from f.

Suppose now that f is symmetric or, more, generally, that for some parameter μ we know the value of $P(X > \mu)$ (where we assume that $a > \mu$). We can then take $\delta_3 = \frac{1}{n}\sum_{i=1}^n \mathbb{I}(X_i > \mu)$ and form the control variate estimator

$$\delta_2 = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}(X_i > a) + \beta\left(\frac{1}{n}\sum_{i=1}^{n}\mathbb{I}(X_i > \mu) - P(X > \mu)\right).$$

Since $\text{var}(\delta_2) = \text{var}(\delta_1) + \beta^2\text{var}(\delta_3) + 2\beta\text{cov}(\delta_1, \delta_3)$ and

$$\begin{aligned}\text{cov}(\delta_1, \delta_3) &= \frac{1}{n}P(X > a)[1 - P(X > \mu)]\,,\\ \text{var}(\delta_3) &= \frac{1}{n}P(X > \mu)[1 - P(X > \mu)]\,,\end{aligned} \tag{4.18}$$

it follows that δ_2 will be an improvement over δ_1 if

$$\beta < 0 \quad \text{and} \quad |\beta| < 2\frac{\text{cov}(\delta_1, \delta_3)}{\text{var}(\delta_3)} = 2\frac{P(X > a)}{P(X > \mu)}.$$

If $P(X > \mu) = 1/2$ and we have some idea of the value of $P(X > a)$, we can choose an appropriate value for β (see Problems 4.15 and 4.18). ‖

Example 4.15. Logistic regression. Consider the logistic regression model introduced in Example 1.13,

$$P(Y_i = 1) = \exp(x_i^t\theta)/\{1 + \exp(x_i^t\theta)\}.$$

The likelihood associated with a sample $((x_1, Y_1), \ldots, (x_n, Y_n))$ can be written

$$\exp\left(\theta^t\sum_i Y_i x_i\right)\prod_{i=1}^{n}\{1 + \exp(x_i^t\theta)\}^{-1}\,.$$

When $\pi(\theta)$ is a *conjugate prior* (see Note 1.6.1),

$$\pi(\theta|\zeta, \lambda) \propto \exp(\theta^t\zeta)\prod_{i=1}^{n}\{1 + \exp(x_i^t\theta)\}^{-\lambda}, \qquad \lambda > 0, \tag{4.19}$$

the posterior distribution of θ is of the same form, with $(\sum_i Y_i x_i + \zeta, \lambda + 1)$ replacing (ζ, λ).

The expectation $\mathbb{E}^\pi[\theta|\sum_i Y_i x_i + \zeta, \lambda + 1]$ is derived from variables θ_j $(1 \le j \le m)$ generated from (4.19). Since the logistic distribution is in an exponential family, the following holds (see Brown 1986 or Robert 2001, Section 3.3):

$$\mathbb{E}_\theta\left[\sum_i Y_i x_i\right] = n\nabla\psi(\theta)$$

and

$$\mathbb{E}^\pi\left[\nabla\psi(\theta)\middle|\sum_i Y_i x_i + \zeta, \lambda + 1\right] = \frac{\sum_i Y_i x_i + \zeta}{n(\lambda + 1)}\,.$$

Therefore, the posterior expectation of the function

$$n\nabla\psi(\theta) = \sum_{i=1}^{n}\frac{\exp(x_i^t\theta)}{1 + \exp(x_i^t\theta)}\,x_i$$

is known and equal to $(\sum_i Y_i x_i + \zeta)/(\lambda + 1)$ under the prior distribution $\pi(\theta|\zeta, \lambda)$. Unfortunately, a control variate version of

$$\delta_1 = \frac{1}{m}\sum_{j=1}^{m}\theta_j$$

is not available since the optimal constant β^* (or even its sign) cannot be evaluated, except by the regression of the θ_j's upon the

$$\sum_i \frac{\exp x_i^t\theta_j}{1 + \exp x_i^t\theta_j}\,x_i\,.$$

Thus the fact that the posterior mean of $\nabla\psi(\theta)$ is known does not help us to establish a control variate estimator. This information can be used in a more informal way to study convergence of δ_1 (see, for instance, Robert 1993). ‖

In conclusion, the technique of control variates is manageable only in very specific cases: the control function h must be available, as well as the optimal weight β^*. See, however, Brooks and Gelman (1998b) for a general approach based on the score function (whose expectation is null under general regularity conditions).

4.5 Problems

4.1 (Chen and Shao 1997) As mentioned, normalizing constants are superfluous in Bayesian inference except in the case when several models are considered at once (as in the computation of Bayes factors). In such cases, where $\pi_1(\theta) = \tilde{\pi}_1(\theta)/c_1$ and $\pi_2(\theta) = \tilde{\pi}_2(\theta)/c_2$, and only $\tilde{\pi}_1$ and $\tilde{\pi}_2$ are known, the quantity to approximate is $\varrho = c_1/c_2$ or $\xi = \log(c_1/c_2)$.

(a) Show that the ratio ϱ can be approximated by

$$\frac{1}{n}\sum_{i=1}^{n}\frac{\tilde{\pi}_1(\theta_i)}{\tilde{\pi}_2(\theta_i)}, \qquad \theta_1, \ldots, \theta_n \sim \pi_2.$$

(*Hint:* Use an importance sampling argument.)

(b) Show that

$$\frac{\int\tilde{\pi}_1(\theta)\alpha(\theta)\pi_2(\theta)d\theta}{\int\tilde{\pi}_2(\theta)\alpha(\theta)\pi_1(\theta)d\theta} = \frac{c_1}{c_2}$$

holds for every function $\alpha(\theta)$ such that both integrals are finite.

While this approach may seem far-fetched, because of its requirements on the distribution f, note that it can be combined with importance sampling, where the importance function g may be chosen in such a way that its quantiles are well known. Also, it can be iterated, with each step producing an evaluation of the ϱ_i's and of the corresponding variance factors, which may help in selecting the partition and the n_i's in the next step (see, however, Problem 4.22).

An extension proposed by McKay et al. (1979) introduces stratification on all input dimensions. More precisely, if the domain is represented as the unit hypercube $\mathcal{H}$ in $\mathbb{R}^d$ and the integral (3.5) is written as $\int_{\mathcal{H}} h(x)dx$, *Latin hypercube sampling* relies on the simulation of (a) d random permutations π_j of $\{1,\ldots,n\}$ and (b) uniform $\mathcal{U}([0,1])$ rv's $U_{i_1,\ldots,i_d,j}$ $(j = 1,\ldots,d,\ 1 \le i_1,\ldots,i_d \le n)$. A sample of n vectors $\mathbf{X}_i$ in the unit hypercube is then produced as $\mathbf{X}(\pi_1(i),\ldots,\pi_d(i))$ with

$$\mathbf{X}(i_1,\ldots,i_d) = (X_1(i_1,\ldots,i_d),\ldots,X_d(i_1,\ldots,i_d)),$$
$$X_j(i_1,\ldots,i_d) = \frac{i_j - U_{i_1,\ldots,i_d,j}}{n}, \qquad 1 \le j \le d.$$

The component $X_j(i_1,\ldots,i_d)$ is therefore a point taken at random on the interval $[i_{j-1}/n, i_j/n]$ and the permutations ensure that no uniform variable is taken twice from the same interval, for every dimension. Note that we also need only generate $n \times d$ uniform random variables. (Latin hypercubes are also used in agricultural experiments to ensure that all parcels and all varieties are used, at a minimal cost. See Mead 1988 or Kuehl 1994.) McKay et al. (1979) show that when h is a real-valued function, the variance of the resulting estimator is substantially reduced, compared with the regular Monte Carlo estimator based on the same number of samples. Asymptotic results about this technique can be found in Stein (1987) and Loh (1996). (It is, however, quite likely that the curse of dimensionality, see Section 4.3, occurs for this technique.)

5

Monte Carlo Optimization

> "Remember, boy," Sam Nakai would sometimes tell Chee, "when you're tired of walking up a long hill you think about how easy it's going to be walking down."
> —Tony Hillerman, *A Thief of Time*

This chapter is the equivalent for optimization problems of what Chapter 3 is for integration problems. Here we distinguish between two separate uses of computer generated random variables. The first use, as seen in Section 5.2, is to produce stochastic techniques to reach the maximum (or minimum) of a function, devising random explorations techniques on the surface of this function that avoid being trapped in a local maximum (or minimum) but also that are sufficiently attracted by the global maximum (or minimum). The second use, described in Section 5.3, is closer to Chapter 3 in that it approximates the function to be optimized. The most popular algorithm in this perspective is the EM (Expectation–Maximization) algorithm.

5.1 Introduction

Similar to the problem of integration, differences between the numerical approach and the simulation approach to the problem

$$\max_{\theta\in\Theta} h(\theta) \tag{5.1}$$

lie in the treatment of the function[1] h. (Note that (5.1) also covers minimization problems by considering $-h$.) In approaching an optimization problem

[1] Although we use θ as the running parameter and h typically corresponds to a (possibly penalized) transform of the likelihood function, this setup applies to inferential problems other than likelihood or posterior maximization. As noted in the introduction to Chapter 3, problems concerned with complex loss functions or confidence regions also require optimization procedures.

using deterministic numerical methods, the analytical properties of the target function (convexity, boundedness, smoothness) are often paramount. For the simulation approach, we are more concerned with h from a probabilistic (rather than analytical) point of view. Obviously, this dichotomy is somewhat artificial, as there exist simulation approaches where the probabilistic interpretation of h is not used. Nonetheless, the use of the analytical properties of h plays a lesser role in the simulation approach.

Numerical methods enjoy a longer history than simulation methods (see, for instance, Kennedy and Gentle 1980 or Thisted 1988), but simulation methods have gained in appeal due to the relaxation of constraints both on the regularity of the domain Θ and on the function h. Of course, there may exist an alternative numerical approach which provides an exact solution to (5.1), a property rarely achieved by a stochastic algorithm, but simulation has the advantage of bypassing the preliminary steps of devising an algorithm and studying whether some regularity conditions on h hold. This is particularly true when the function h is very costly to compute.

Example 5.1. Signal processing. Ó Ruanaidh and Fitzgerald (1996) study signal processing data, of which a simple model is $(i = 1, \ldots, N)$

$$x_i = \alpha_1 \cos(\omega t_i) + \alpha_2 \sin(\omega t_i) + \epsilon_i, \qquad \epsilon_i \sim \mathcal{N}(0, \sigma^2),$$

with unknown parameters $\alpha = (\alpha_1, \alpha_2), \omega$, and σ and observation times $t_1, \ldots, t_N$. The likelihood function is then of the form

$$\sigma^{-N} \exp\left(-\frac{(\mathbf{x} - G\alpha^t)^t(\mathbf{x} - G\alpha^t)}{2\sigma^2}\right),$$

with $\mathbf{x} = (x_1, \ldots, x_N)$ and

$$G = \begin{pmatrix} \cos(\omega t_1) & \sin(\omega t_1) \\ \vdots & \vdots \\ \cos(\omega t_N) & \sin(\omega t_N) \end{pmatrix}.$$

The prior $\pi(\alpha, \omega, \sigma) = \sigma^{-1}$ leads to the marginal distribution

$$\pi(\omega|\mathbf{x}) \propto \left(\mathbf{x}^t\mathbf{x} - \mathbf{x}^t G(G^tG)^{-1}G^t\mathbf{x}\right)^{(2-N)/2} (\det G^tG)^{-1/2}, \tag{5.2}$$

which, although explicit in ω, is not particularly simple to compute. This setup is also illustrative of functions with many modes, as shown by Ó Ruanaidh and Fitzgerald (1996). ‖

Following Geyer (1996), we want to distinguish between two approaches to Monte Carlo optimization. The first is an *exploratory* approach, in which the goal is to optimize the function h by describing its entire range. The actual properties of the function play a lesser role here, with the Monte Carlo

aspect more closely tied to the exploration of the entire space Θ, even though, for instance, the slope of h can be used to speed up the exploration. (Such a technique can be useful in describing functions with multiple modes, for example.) The second approach is based on a probabilistic *approximation* of the objective function h and is somewhat of a preliminary step to the actual optimization. Here, the Monte Carlo aspect exploits the probabilistic properties of the function h to come up with an acceptable approximation and is less concerned with exploring Θ. We will see that this approach can be tied to *missing data methods*, such as the EM algorithm. We note also that Geyer (1996) only considers the second approach to be "Monte Carlo optimization." Obviously, even though we are considering these two different approaches separately, they might be combined in a given problem. In fact, methods like the EM algorithm (Section 5.3.2) or the Robbins–Monro algorithm (Section 5.5.3) take advantage of the Monte Carlo approximation to enhance their particular optimization technique.

5.2 Stochastic Exploration

5.2.1 A Basic Solution

There are a number of cases where the exploration method is particularly well suited. First, if Θ is bounded, which may sometimes be achieved by a reparameterization, a first approach to the resolution of (5.1) is to simulate from a uniform distribution on Θ, $u_1, \ldots, u_m \sim \mathcal{U}_\Theta$, and to use the approximation $h_m^* = \max(h(u_1), \ldots, h(u_m))$. This method converges (as m goes to ∞), but it may be very slow since it does not take into account any specific feature of h. Distributions other than the uniform, which can possibly be related to h, may then do better. In particular, in setups where the likelihood function is extremely costly to compute, the number of evaluations of the function h is best kept to a minimum.

Example 5.2. A first Monte Carlo maximization. Recall the function that we looked at in Example 3.4, $h(x) = [\cos(50x) + \sin(20x)]^2$. Since the function is defined on a bounded interval, we try our naïve strategy and simulate $u_1, \ldots, u_m \sim \mathcal{U}(0, 1)$, and use the approximation $h_m^* = \max(h(u_1), \ldots, h(u_m))$. The results are shown in Figure 5.1. There we see that the random search has done a fine job of mimicking the function. The Monte Carlo maximum is 3.832, which agrees perfectly with the "true" maximum, obtained by an exhaustive evaluation.

Of course, this is a small example, and as mentioned above, this naïve method can be costly in many situations. However, the example illustrates the fact that in low-dimensional problems, if function evaluation is rapid, this method is a reasonable choice. ‖

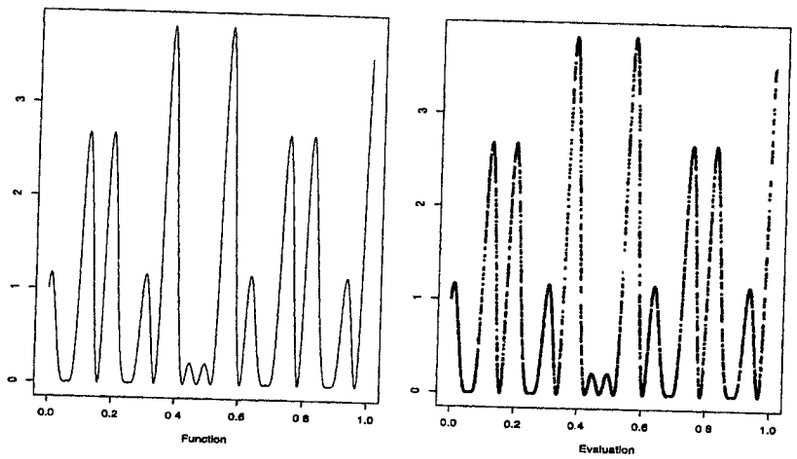


Fig. 5.1. Calculation of the maximum of the function (1.26), given in Example 3.4. The left panel is a graph of the function, and the right panel is a scatterplot of 5000 random $\mathcal{U}(0,1)$ and the function evaluated at those points.

This leads to a second, and often more fruitful, direction, which relates h to a probability distribution. For instance, if h is positive and if

$$\int_\Theta h(\theta)\, d\theta < +\infty\,,$$

the resolution of (5.1) amounts to finding the *modes* of the density h. More generally, if these conditions are not satisfied, then we may be able to transform the function $h(\theta)$ into another function $H(\theta)$ that satisfies the following:

(i) The function H is non-negative and satisfies $\int H < \infty$.
(ii) The solutions to (5.1) are those which maximize $H(\theta)$ on Θ.

For example, we can take

$$H(\theta) = \exp(h(\theta)/T) \quad \text{or} \quad H(\theta) = \exp\{h(\theta)/T\}/(1+\exp\{h(\theta)/T\})$$

and choose T to accelerate convergence or to avoid local maxima (as in simulated annealing; see Section 5.2.3). When the problem is expressed in statistical terms, it becomes natural to then generate a sample $(\theta_1,\ldots,\theta_m)$ from h (or H) and to apply a standard mode estimation method (or to simply compare the $h(\theta_i)$'s). (In some cases, it may be more useful to decompose $h(\theta)$ into $h(\theta) = h_1(\theta)h_2(\theta)$ and to simulate from h_1.)

Example 5.3. Minimization of a complex function. Consider minimizing the (artificially constructed) function in $\mathbb{R}^2$

$$h(x,y) = (x\sin(20y) + y\sin(20x))^2\cosh(\sin(10x)x) + (x\cos(10y) - y\sin(10x))^2\cosh(\cos(20y)y)\,,$$

whose global minimum is 0, attained at $(x,y) = (0,0)$. (This is the big brother of the function in Example 5.2.) Since this function has many local minima, as shown by Figure 5.2, it does not satisfy the conditions under which standard minimization methods are guaranteed to provide the global minimum. On the other hand, the distribution on $\mathbb{R}^2$ with density proportional to $\exp(-h(x,y))$ can be simulated, even though this is not a standard distribution, by using, for instance, Markov chain Monte Carlo techniques (introduced in Chapters 7 and 8), and a convergent approximation of the minimum of $h(x,y)$ can be derived from the minimum of the resulting $h(x_i,y_i)$'s. An alternative is to simulate from the density proportional to

$$h_1(x,y) = \exp\{-(x\sin(20y) + y\sin(20x))^2 - (x\cos(10y) - y\sin(10x))^2\},$$

which eliminates the computation of both cosh and sinh in the simulation step. ‖

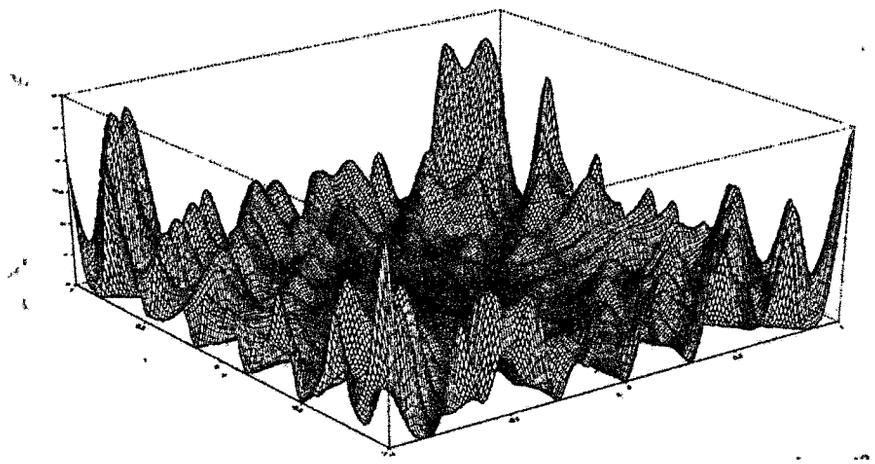


Fig. 5.2. Grid representation of the function $h(x,y)$ of Example 5.3 on $[-1,1]^2$.

Exploration may be particularly difficult when the space Θ is not convex (or perhaps not even connected). In such cases, the simulation of a sample $(\theta_1,\ldots,\theta_m)$ may be much faster than a numerical method applied to (5.1). The appeal of simulation is even clearer in the case when h can be represented as

$$h(\theta) = \int H(x,\theta)dx\,.$$

In particular, if $H(x,\theta)$ is a density and if it is possible to simulate from this density, the solution of (5.1) is the mode of the marginal distribution of θ. (Although this setting may appear contrived or even artificial, we will see in Section 5.3.1 that it includes the case of missing data models.)

We now look at several methods to find maxima that can be classified as *exploratory methods*.

5.2.2 Gradient Methods

As mentioned in Section 1.4, the *gradient method* is a deterministic numerical approach to the problem (5.1). It produces a sequence (θ_j) that converges to the exact solution of (5.1), θ^*, when the domain $\Theta \subset \mathbb{R}^d$ and the function $(-h)$ are both convex. The sequence (θ_j) is constructed in a recursive manner through

$$\theta_{j+1} = \theta_j + \alpha_j \nabla h(\theta_j)\,, \qquad \alpha_j > 0\,, \tag{5.3}$$

where ∇h is the gradient of h. For various choices of the sequence (α_j) (see Thisted 1988), the algorithm converges to the (unique) maximum.

In more general setups (that is, when the function or the space is less regular), equation (5.3) can be modified by stochastic perturbations to again achieve convergence, as described in detail in Rubinstein (1981) or Duflo (1996, pp. 61–63). One of these stochastic modifications is to choose a second sequence (β_j) to define the chain (θ_j) by

$$\theta_{j+1} = \theta_j + \frac{\alpha_j}{2\beta_j}\,\Delta h(\theta_j, \beta_j\zeta_j)\,\zeta_j\,. \tag{5.4}$$

The variables ζ_j are uniformly distributed on the unit sphere $||\zeta|| = 1$ and $\Delta h(x,y) = h(x+y) - h(x-y)$ approximates $2||y||\nabla h(x)$. In contrast to the deterministic approach, this method does not necessarily proceed along the steepest slope in θ_j, but this property is sometimes a *plus* in the sense that it may avoid being trapped in local maxima or in saddlepoints of h.

The convergence of (θ_j) to the solution θ^* again depends on the choice of (α_j) and (β_j). We note in passing that (θ_j) can be seen as a *nonhomogeneous Markov chain* (see Definition 6.4) which almost surely converges to a given value. The study of these chains is quite complicated given their ever-changing transition kernel (see Winkler 1995 for some results in this direction). However, sufficiently strong conditions such as the decrease of α_j toward 0 and of α_j/β_j to a nonzero constant are enough to guarantee the convergence of the sequence (θ_j).

Example 5.4. (Continuation of Example 5.3) We can apply the iterative construction (5.4) to the multimodal function $h(x,y)$ with different sequences of α_j's and β_j's. Figure 5.3 and Table 5.1 illustrate that, depending on the starting value, the algorithm converges to different local minima of the function h. Although there are occurrences when the sequence $h(\theta_j)$ increases

and avoids some local minima, the solutions are quite distinct for the three different sequences, both in location and values.

As shown by Table 5.1, the number of iterations needed to achieve stability of θ_T also varies with the choice of (α_j, β_j). Note that Case 1 results in a very poor evaluation of the minimum, as the fast decrease of (α_j) is associated with big jumps in the first iterations. Case 2 converges to the closest local minima, and Case 3 illustrates a general feature of the stochastic gradient method, namely that slower decrease rates of the sequence (α_j) tend to achieve better minima. The final convergence along a valley of h after some initial big jumps is also noteworthy. ||

α_j	β_j	θ_T	$h(\theta_T)$	$\min_t h(\theta_t)$	Iteration T
$1/10j$	$1/10j$	$(-0.166, 1.02)$	1.287	0.115	50
$1/100j$	$1/100j$	$(0.629, 0.786)$	0.00013	0.00013	93
$1/10\log(1+j)$	$1/j$	$(0.0004, 0.245)$	4.24×10^{-6}	2.163×10^{-7}	58

Table 5.1. Results of three stochastic gradient runs for the minimization of the function h in Example 5.3 with different values of (α_j, β_j) and starting point $(0.65, 0.8)$. The iteration T is obtained by the stopping rule $||\theta_T - \theta_{T-1}|| < 10^{-5}$.

This approach is still quite close to numerical methods in that it requires a precise knowledge on the function h, which may not necessarily be available.

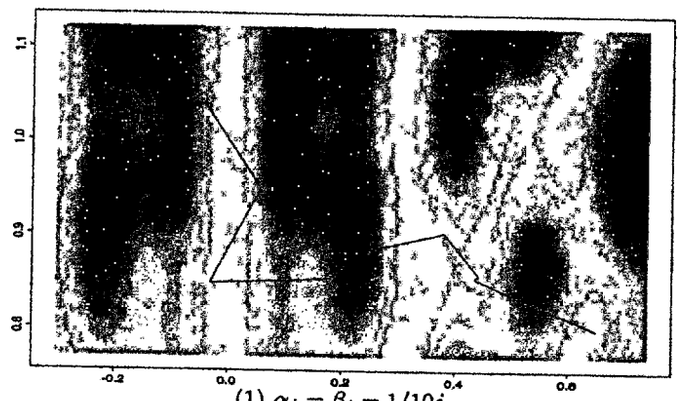
5.2.3 Simulated Annealing

The simulated annealing algorithm[2] was introduced by Metropolis et al. (1953) to minimize a criterion function on a finite set with very large size[3], but it also applies to optimization on a continuous set and to simulation (see Kirkpatrick et al. 1983, Ackley et al. 1985, and Neal 1993, 1995).

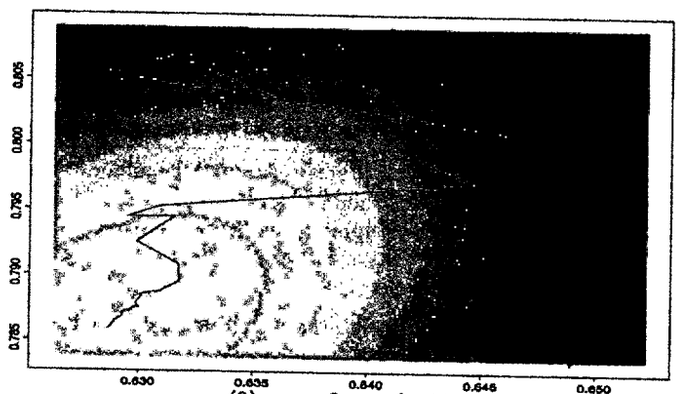
The fundamental idea of simulated annealing methods is that a change of scale, called *temperature*, allows for faster moves on the surface of the function h to maximize, whose negative is called *energy*. Therefore, rescaling partially avoids the trapping attraction of local maxima. Given a temperature parameter $T > 0$, a sample $\theta_1^T, \theta_2^T, \ldots$ is generated from the distribution

[2] This name is borrowed from wetallurgy: a metal manufactured by a slow decrease of temperature (*annealing*) is stronger than a metal manufactured by a fast decrease of temperature. There is also input from physics, as the function to be minimized is called *energy* and the variance factor T, which controls convergence, is called *temperature*. We will try to keep these idiosyncrasies to a minimal level, but they are quite common in the literature.

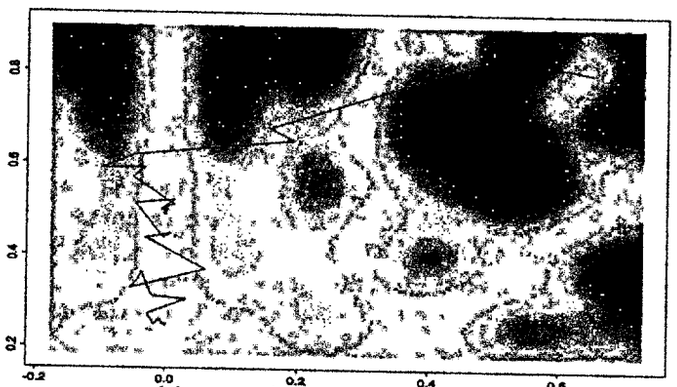
[3] This paper is also the originator of the *Markov chain Monte Carlo methods* developed in the following chapters.



(1) $\alpha_j = \beta_j = 1/10j$



(2) $\alpha_j = \beta_j = 1/100j$



(3) $\alpha_j = 1/10\log(1+j)$, $\beta_j = 1/j$

Fig. 5.3. Stochastic gradient paths for three different choices of the sequences $(\alpha_j)$and (β_j) and starting point $(0.65, 0.8)$ for the same sequence (ζ_j) in (5.4). The gray levels are such that darker shades mean higher elevations. The function h to be minimized is defined in Example 5.3.

$$\pi(\theta) \propto \exp(h(\theta)/T)$$

and can be used as in Section 5.2.1 to come up with an approximate maximum of h. As T decreases toward 0, the values simulated from this distribution become concentrated in a narrower and narrower neighborhood of the local maxima of h (see Theorem 5.10 and Winkler 1995).

The fact that this approach has a moderating effect on the attraction of the local maxima of h becomes more apparent when we consider the simulation method proposed by Metropolis et al. (1953). Starting from θ_0, ζ is generated from a uniform distribution on a neighborhood $\mathcal{V}(\theta_0)$ of θ_0 or, more generally, from a distribution with density $g(|\zeta - \theta_0|)$, and the new value of θ is generated as follows:

$$\theta_1 = \begin{cases} \zeta & \text{with probability } \rho = \exp(\Delta h/T) \wedge 1 \\ \theta_0 & \text{with probability } 1-\rho, \end{cases}$$

where $\Delta h = h(\zeta) - h(\theta_0)$. Therefore, if $h(\zeta) \geq h(\theta_0)$, ζ is accepted with probability 1; that is, θ_0 is always changed into ζ. On the other hand, if $h(\zeta) < h(\theta_0)$, ζ may still be accepted with probability $\rho \neq 0$ and θ_0 is then changed into ζ. This property allows the algorithm to escape the attraction of θ_0 if θ_0 is a local maximum of h, with a probability which depends on the choice of the scale T, compared with the range of the density g. (This method is in fact the *Metropolis algorithm*, which simulates the density proportional to $\exp\{h(\theta)/T\}$, as the limiting distribution of the chain $\theta_0, \theta_1, \ldots$, as described and justified in Chapter 7.)

In its most usual implementation, the simulated annealing algorithm modifies the temperature T at each iteration; it is then of the form

Algorithm A.19 –Simulated Annealing–

1. Simulate ζ from an instrumental distribution with density $g(|\zeta - \theta_i|)$.
2. Accept $\theta_{i+1} = \zeta$ with probability $\rho_i = \exp\{\Delta h_i/T_i\} \wedge 1$; [A.19] take $\theta_{i+1} = \theta_i$ otherwise.
3. Update T_i to T_{i+1}.

Example 5.5. A first simulated annealing maximization. We again look at the function from Example 5.2,

$$h(x) = [\cos(50x) + \sin(20x)]^2,$$

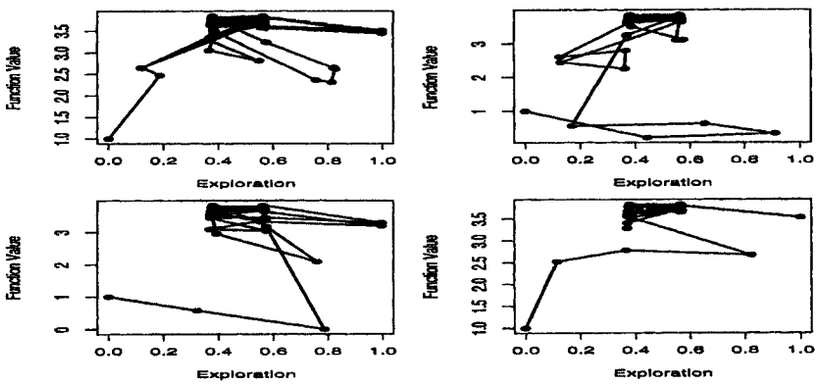


Fig. 5.4. Calculation of the maximum of the function (1.26), given in Example 5.5. The four different panels show the trajectory of 2500 pairs $(x^{(t)}, h(x^{(t)}))$ for each of four different runs.

and apply a simulated annealing algorithm to find the maximum. The specific algorithm we use is

At iteration t the algorithm is at $(x^{(t)}, h(x^{(t)}))$:

1. Simulate $a \sim \mathcal{U}(a_t, b_t)$ where $a_t = \max(x^{(t)} - r, 0)$ and $b_t = \min(x^{(t)} + r, 1)$.
2. Accept $x^{(t+1)} = a$ with probability
$$\rho^{(t)} = \min\left\{\exp\left(\frac{h(a) - h(x^{(t)})}{T_t}\right), 1\right\};$$
take $x^{(t+1)} = x^{(t)}$ otherwise.
3. Update T_t to T_{t+1}.

For $r = .5$ and $T_t = 1/\log(t)$, the results of the algorithm are shown in Figure 5.4. The four panels show different trajectories of the points $(x^{(t)}, h(x^{(t)}))$. It is interesting to see how the path moves toward the maximum fairly rapidly, and then remains there, oscillating between the two maxima (remember that h is symmetric around $1/2$.

The value of r controls the size of the interval around the current point (we truncate to stay in $(0, 1)$ and the value of T_t controls the cooling. For different values of r and T the path will display different properties. See Problem 5.4 and the more complex Example 5.9. ‖

An important feature of the simulated annealing algorithm is that there exist convergence results in the case of finite spaces, as Theorem 5.7 below, which was proposed by Hàjek (1988). (See Winkler 1995, for extensions.)

Consider the following notions, which are used to impose restrictions on the decrease rate of the temperature:

Definition 5.6. Given a finite state-space $\mathcal{E}$ and a function h to be maximized:

(i) a state $e_j \in \mathcal{E}$ can be *reached at altitude* $\underline{h}$ from state $e_i \in \mathcal{E}$ if there exists a sequence of states $e_1, \ldots, e_n$ linking e_i and e_j, such that $h(e_k) \geq \underline{h}$ for $k = 1, \ldots, n$;

(ii) the *height of a maximum* e_i is the largest value d_i such that there exists a state e_j such that $h(e_j) > h(e_i)$ which can be reached at altitude $h(e_i) + d_i$ from e_i.

Thus, $h(e_i) + d_i$ is the altitude of the highest pass linking e_i and e_j through an optimal sequence. (In particular, $h(e_i) + d_i$ can be larger than the altitude of the closest pass relating e_i and e_j.) By convention, we take $d_i = -\infty$ if e_i is a global maximum. If $\mathcal{O}$ denotes the set of local maxima of E and $\underline{\mathcal{O}}$ is the subset of $\mathcal{O}$ of global maxima, Hàjek (1988) establishes the following result:

Theorem 5.7. *Consider a system in which it is possible to link two arbitrary states by a finite sequence of states. If, for every* $\underline{h} > 0$ *and every pair* (e_i, e_j), e_i *can be reached at altitude* $\underline{h}$ *from* e_j *if and only if* e_j *can be reached at altitude* $\underline{h}$ *from* e_i, *and if* (T_i) *decreases toward* 0, *the sequence* (θ_i) *defined by Algorithm* [A.19] *satisfies*

$$\lim_{i \to \infty} P(\theta_i \in \underline{\mathcal{O}}) = 1$$

if and only if

$$\sum_{i=1}^{\infty} \exp(-D/T_i) = +\infty\,,$$

with $D = \min\{d_i : e_i \in \mathcal{O} - \underline{\mathcal{O}}\}$.

This theorem therefore gives a necessary and sufficient condition, on the rate of decrease of the temperature, so that the simulated annealing algorithm converges to the set of global maxima. This remains a relatively formal result since D is, in practice, unknown. For example, if $T_i = \Gamma/\log i$, there is convergence to a global maximum if and only if $\Gamma \geq D$. Numerous papers and books have considered the practical determination of the sequence (T_n) (see Geman and Geman 1984, Mitra et al. 1986, Van Laarhoven and Aarts 1987, Aarts and Kors 1989, Winkler 1995, and references therein). Instead of the above logarithmic rate, a geometric rate, $T_i = \alpha^i T_0$ $(0 < \alpha < 1)$, is also often adopted in practice, with the constant α calibrated at the beginning of the algorithm so that the acceptance rate is high enough in the Metropolis algorithm (see Section 7.6).

The fact that approximate methods are necessary for optimization problems in finite state-spaces may sound rather artificial and unnecessary, but the spaces involved in some modeling can be huge. For instance, a black-and-white TV image with 256×256 pixels corresponds to a state-space with cardinality $2^{256\times256} \simeq 10^{20,000}$. Similarly, the analysis of DNA sequences may involve 600 thousand bases (A, C, G, or T), which corresponds to state-spaces of size $4^{600,000}$ (see Churchill 1989, 1995).

Example 5.8. Ising model. The *Ising model* can be applied in electromagnetism (Cipra 1987) and in image processing (Geman and Geman 1984). It models two-dimensional tables s, of size $D \times D$, where each term of s takes the value $+1$ or -1. The distribution of the entire table is related to the (so-called energy) function

$$h(s) = -J \sum_{(i,j)\in\mathcal{N}} s_i s_j - H \sum_i s_i \,, \tag{5.5}$$

where i denotes the index of a term of the table and $\mathcal{N}$ is an equivalence neighborhood relation, for instance, when i and j are neighbors either vertically or horizontally. (The scale factors J and H are supposedly known.) The model (5.5) is a particular case of models used in spatial statistics (Cressie 1993) to describe multidimensional correlated structures.

Note that the conditional representation of (5.5) is equivalent to a *logit* model on $\tilde{s}_i = (s_i + 1)/2$,

$$P(\tilde{S}_i = 1 | s_j, j \neq i) = \frac{e^g}{1 + e^g} \,, \tag{5.6}$$

with $g = g(s_j) = 2(H + J\sum_j s_j)$, the sum being taken on the neighbors of i. For known parameters H and J, the inferential question may be to obtain the most likely configuration of the system; that is, the minimum of $h(s)$. The implementation of the Metropolis et al. (1953) approach in this setup, starting from an initial value $s^{(0)}$, is to modify the sites of the table s one at a time using the conditional distributions (5.6), with probability $\exp(-\Delta h/T)$, ending up with a modified table $s^{(1)}$, and to iterate this method by decreasing the temperature T at each step. The reader can consult Swendson and Wang (1987) and Swendson et al. (1992) for their derivation of efficient simulation algorithms in these models and accelerating methods for the Gibbs sampler (see Problem 7.43). ‖

Duflo (1996, pp. 264–271) also proposed an extension of these simulated annealing methods to the general (continuous) case. Andrieu and Doucet (2000) give a detailed proof of convergence of the simulated annealing algorithm, as well as sufficient conditions on the cooling schedule, in the setup of hidden Markov models (see Section 14.6.3). Their proof, which is beyond our scope, is based on the developments of Haario and Sacksman (1991).

Case	T_i	θ_T	$h(\theta_T)$	$\min_t h(\theta_t)$	Accept. rate
1	$1/10i$	$(-1.94, -0.480)$	0.198	$4.02\,10^{-7}$	0.9998
2	$1/\log(1+i)$	$(-1.99, -0.133)$	3.408	3.823×10^{-7}	0.96
3	$100/\log(1+i)$	$(-0.575, 0.430)$	0.0017	4.708×10^{-9}	0.6888
4	$1/10\log(1+i)$	$(0.121, -0.150)$	0.0359	2.382×10^{-7}	0.71

Table 5.2. Results of simulated annealing runs for different values of T_i and starting point $(0.5, 0.4)$.

Example 5.9. (Continuation of Example 5.3) We can apply the algorithm [*A*.19] to find a local minimum of the function h of Example 5.3, or equivalently a maximum of the function $\exp(-h(x,y)/T_i)$. We choose a uniform distribution on $[-0.1, 0.1]$ for g, and different rates of decrease of the temperature sequence (T_i). As illustrated by Figure 5.5 and Table 5.2, the results change with the rate of decrease of the temperature T_i. Case 3 leads to a very interesting exploration of the valleys of h on both sides of the central zone. Since the theory (Duflo 1996) states that rates of the form $\Gamma/\log(i+1)$ are satisfactory for Γ large enough, this shows that $\Gamma = 100$ should be acceptable. Note also the behavior of the acceptance rate in Table 5.2 for Step 2 in algorithm [*A*.19]. This is indicative of a rule we will discuss further in Chapter 7 with Metropolis–Hastings algorithms, namely that superior performances are not always associated with higher acceptance rates. ‖

5.2.4 Prior Feedback

Another approach to the maximization problem (5.1) is based on the result of Hwang (1980) of convergence (in T) of the so-called *Gibbs measure* $\exp(h(\theta)/T)$ (see Section 5.5.3) to the uniform distribution on the set of global maxima of h. This approach, called *recursive integration* or *prior feedback* in Robert (1993) (see also Robert and Soubiran 1993), is based on the following convergence result.

Theorem 5.10. *Consider h a real-valued function defined on a closed and bounded set, Θ, of $\mathbb{R}^p$. If there exists a unique solution θ^* satisfying*

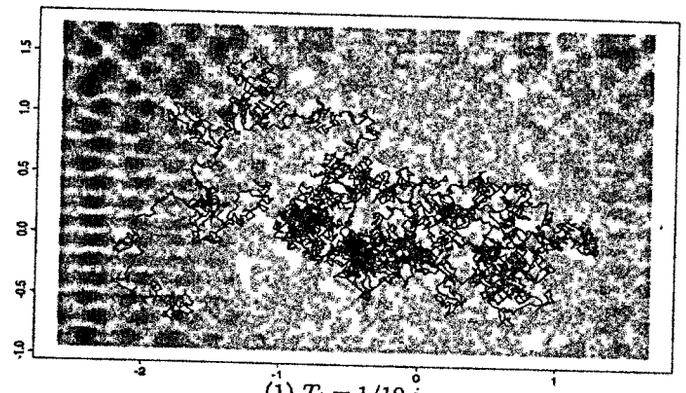
$$\theta^* = \arg\max_{\theta\in\Theta} h(\theta) \,,$$

then

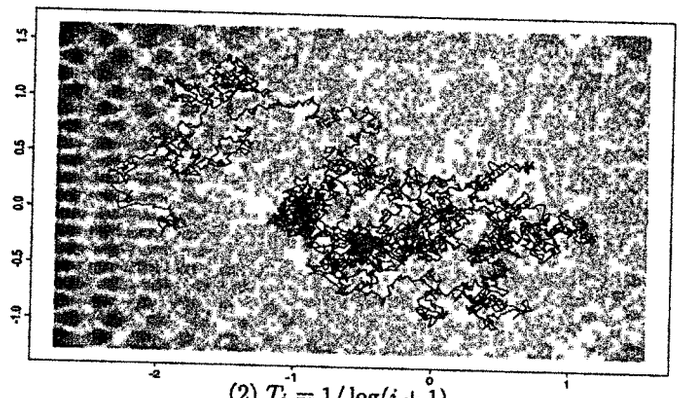
$$\lim_{\lambda\to\infty} \frac{\int_\Theta \theta\, e^{\lambda h(\theta)}\, d\theta}{\int_\Theta e^{\lambda h(\theta)}\, d\theta} = \theta^* \,,$$

provided h is continuous at θ^*.

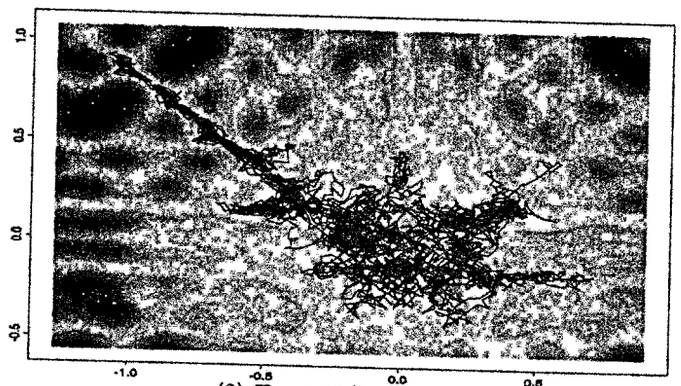
See Problem 5.6 for a proof. More details can be found in Pincus (1968) (see also Robert 1993 for the case of exponential families and Duflo 1996, pp.



(1) $T_i = 1/10\,i$



(2) $T_i = 1/\log(i+1)$



(3) $T_i = 100/\log(i+1)$

Fig. 5.5. Simulated annealing sequence of 5000 points for three different choices of the temperature T_i in [A.19] and starting point $(0.5, 0.4)$, aimed at minimizing the function h of Example 5.3.

244–245, for a sketch of a proof). A related result can be found in D'Epifanio (1989, 1996). A direct corollary to Theorem 5.10 then justifies the recursive integration method which results in a Bayesian approach to maximizing the log-likelihood, $\ell(\theta|x)$.

Corollary 5.11. *Let π be a positive density on Θ. If there exists a unique maximum likelihood estimator θ^*, it satisfies*

$$\lim_{\lambda\to\infty} \frac{\int \theta e^{\lambda\ell(\theta|x)}\,\pi(\theta)d\theta}{\int e^{\lambda\ell(\theta|x)}\,\pi(\theta)d\theta} = \theta^* .$$

This result uses the same technique as in Theorem 5.10, namely the Laplace approximation of the numerator and denominator integrals (see also Tierney et al. 1989). It mainly expresses the fact that the maximum likelihood estimator can be written as a limit of Bayes estimators associated with an *arbitrary* distribution π and with *virtual* observations corresponding to the λth power of the likelihood, $\exp\{\lambda\ell(\theta|x)\}$. When $\lambda \in \mathbb{N}$,

$$\delta^\pi_\lambda(x) = \frac{\int \theta e^{\lambda\ell(\theta|x)}\,\pi(\theta)d\theta}{\int e^{\lambda\ell(\theta|x)}\,\pi(\theta)d\theta}$$

is simply the Bayes estimator associated with the prior distribution π and a corresponding sample which consists of λ replications of the initial sample x. The intuition behind these results is that as the size of the sample goes to infinity, the influence of the prior distribution vanishes and the distribution associated with $\exp(\lambda\ell(\theta|x))\pi(\theta)$ gets more and more concentrated around the global maxima of $\ell(\theta|x)$ when λ increases (see, e.g., Schervish 1995).

From a practical point of view, the recursive integration method can be implemented by computing the Bayes estimators $\delta^\pi_{\lambda_i}(x)$ for $i = 1, 2, \ldots$ until they stabilize.

Obviously, it is only interesting to maximize the likelihood by this method when more standard methods like the ones above are difficult or impossible to implement and the computation of Bayes estimators is straightforward. (Chapters 7 and 9 show that this second condition is actually very mild.) It is, indeed, necessary to compute the Bayes estimators $\delta^\pi_\lambda(x)$ corresponding to a sequence of λ's until they stabilize. Note that when iterative algorithms are used to compute $\delta^\pi_\lambda(x)$, the previous solution (in λ) of $\delta^\pi_\lambda(x)$ can serve as the new initial value for the computation of $\delta^\pi_\lambda(x)$ for a larger value of λ. This feature increases the analogy with simulated annealing. The differences with simulated annealing are:

(i) for a fixed temperature $(1/\lambda)$, the algorithm converges to a fixed value, δ^π_λ;
(ii) a continuous decrease of $1/\lambda$ is statistically meaningless;

λ	5	10	100	1000	5000	10^4
δ^π_λ	2.02	2.04	1.89	1.98	1.94	2.00

Table 5.3. Sequence of Bayes estimators of δ^π_λ for the estimation of α when $X \sim \mathcal{G}(\alpha, 1)$ and $x = 1.5$.

(iii) the speed of convergence of λ to $+\infty$ does not formally matter[4] for the convergence of $\delta^\pi_\lambda(x)$ to θ^*;
(iv) the statistical motivation of this method is obviously stronger, in particular because of the meaning of the parameter λ;
(v) the only analytical constraint on $\ell(\theta|x)$ is the existence of a global maximum, θ^* (see Robert and Titterington 1998 for extensions).

Example 5.12. Gamma shape estimation. Consider the estimation of the shape parameter, α, of a $\mathcal{G}(\alpha, \beta)$ distribution with β known. Without loss of generality, take $\beta = 1$. For a constant (improper) prior distribution on α, the posterior distribution satisfies

$$\pi_\lambda(\alpha|x) \propto x^{\lambda(\alpha-1)} e^{-\lambda x} \Gamma(\alpha)^{-\lambda} .$$

For a fixed λ, the computation of $\mathbb{E}[\alpha|x, \lambda]$ can be obtained by simulation with the Metropolis–Hastings algorithm (see Chapter 7 for details) based on the instrumental distribution $\mathcal{E}xp(1/\alpha^{(n-1)})$, where $\alpha^{(n-1)}$ denotes the previous value of the associated Markov chain. Table 5.3 presents the evolution of $\delta^\pi_\lambda(x) = \mathbb{E}[\alpha|x, \lambda]$ against λ, for $x = 1.5$.

An analytical verification (using a numerical package like `Mathematica`) shows that the maximum of $x^\alpha/\Gamma(\alpha)$ is, indeed, close to 2.0 for $x = 1.5$. ‖

The appeal of recursive integration is also clear in the case of constrained parameter estimation.

Example 5.13. Isotonic regression. Consider a table of normal observations $X_{i,j} \sim \mathcal{N}(\theta_{i,j}, 1)$ with means that satisfy

$$\theta_{i-1,j} \vee \theta_{i,j-1} \leq \theta_{i,j} \leq \theta_{i+1,j} \wedge \theta_{i,j+1}.$$

Dykstra and Robertson (1982) have developed an efficient *deterministic* algorithm which maximizes the likelihood under these restrictions (see Problems 1.18 and 1.19.) However, a direct application of recursive integration also provides the maximum of likelihood estimator of $\theta = (\theta_{ij})$, requiring neither an extensive theoretical study nor high programming skills.

[4] However, we note that if λ increases too quickly, the performance is affected in that there may be convergence to a local mode (see Robert and Titterington 1998 for an illustration).

Table 5.4 presents the data of Robertson et al. (1988), which relates the notes at the end of first year with two entrance exams at the University of Iowa. Although these values are bounded, it is possible to use a normal model if the function to minimize is the least squares criterion (1.8), as already pointed out in Example 1.5. Table 5.5 provides the solution obtained by recursive integration in Robert and Hwang (1996); it coincides with the result of Robertson et al. (1988). ‖

ACT	1–12	13–15	16–18	19–21	22–24	25–27	28–30	31–33	34–36
91- 99	1.57 (4)	2.11 (5)	2.73 (18)	2.96 (39)	2.97 (126)	3.13 (219)	3.41 (232)	3.45 (47)	3.51 (4)
81- 90	1.80 (6)	1.94 (15)	2.52 (30)	2.68 (65)	2.69 (117)	2.82 (143)	2.75 (70)	2.74 (8)	– (0)
71- 80	1.88 (10)	2.32 (13)	2.32 (51)	2.53 (83)	2.58 (115)	2.55 (107)	2.72 (24)	2.76 (4)	– (0)
61- 70	2.11 (6)	2.23 (32)	2.29 (59)	2.29 (84)	2.50 (75)	2.42 (44)	2.41 (19)	– (0)	– (0)
51- 60	1.60 (11)	2.06 (16)	2.12 (49)	2.11 (63)	2.31 (57)	2.10 (40)	1.58 (4)	2.13 (1)	– (0)
41- 50	1.75 (6)	1.98 (12)	2.05 (31)	2.16 (42)	2.35 (34)	2.48 (21)	1.36 (4)	– (0)	– (0)
31- 40	1.92 (7)	1.84 (6)	2.15 (5)	1.95 (27)	2.02 (13)	2.10 (13)	1.49 (2)	– (0)	– (0)
21- 30	1.62 (1)	2.26 (2)	1.91 (5)	1.86 (14)	1.88 (11)	3.78 (1)	1.40 (2)	– (0)	– (0)
00- 20	1.38 (1)	1.57 (2)	2.49 (5)	2.01 (7)	2.07 (7)	– (0)	0.75 (1)	– (0)	– (0)

Table 5.4. Average grades of first-year students at the University of Iowa given their rank at the end of high school (HSR) and at the ACT exam. Numbers in parentheses indicate the number of students in each category. (*Source:* Robertson et al. 1988.)

ACT	1 – 12	13 – 15	16 – 18	19 – 21	22 – 24	25 – 27	28 – 39	31 – 32	34 – 36
91 – 99	1.87	2.18	2.73	2.96	2.97	3.13	3.41	3.45	3.51
81 – 89	1.87	2.17	2.52	2.68	2.69	2.79	2.79	2.80	–
71 – 79	1.86	2.17	2.32	2.53	2.56	2.57	2.72	2.76	–
61 – 69	1.86	2.17	2.29	2.29	2.46	2.46	2.47	–	–
51 – 59	1.74	2.06	2.12	2.13	2.24	2.24	2.24	2.27	–
41 – 49	1.74	1.98	2.05	2.13	2.24	2.24	2.24	–	–
31 – 39	1.74	1.94	1.99	1.99	2.02	2.06	2.06	–	–
21 – 29	1.62	1.93	1.97	1.97	1.98	2.05	2.06	–	–
00 – 20	1.38	1.57	1.97	1.97	1.97	–	1.97	–	–

Table 5.5. Maximum likelihood estimates of the mean grades under lexicographical constraint. (*Source:* Robert and Hwang 1996).

5.3 Stochastic Approximation

We next turn to methods that work more directly with the objective function rather than being concerned with fast explorations of the space. Informally speaking, these methods are somewhat preliminary to the true optimization step, in the sense that they utilize approximations of the objective function h. We note that these approximations have a different purpose than those we have previously encountered (for example, Laplace and saddlepoint approximations in Section 3.4 and Section 3.6.2). In particular, the methods described here may sometimes result in an additional level of error by looking at the maximum of an approximation to h.

Since most of these approximation methods only work in so-called *missing data models*, we start this section with a brief introduction to these models. We return to the assumption that the objective function h satisfies $h(x) = \mathbb{E}[H(x, Z)]$ and (as promised) show that this assumption arises in many realistic setups. Moreover, note that artificial extensions (or *demarginalization*), which use this representation, are only computational devices and do not invalidate the overall inference.

5.3.1 Missing Data Models and Demarginalization

In the previous chapters, we have already met structures where some missing (or latent) element greatly complicates the observed model. Examples include the obvious censored data models (Example 1.1), mixture models (Example 1.2), where we do not observe the indicator of the component generating the observation, or logistic regression (Example 1.13), where the observation Y_i can be interpreted as an indicator that a continuous variable with logistic distribution is less than $X_i^t\beta$.

Missing data models are best thought of as models where the likelihood can be expressed as

$$g(x|\theta) = \int_{\mathcal{Z}} f(x, z|\theta)\, dz \tag{5.7}$$

or, more generally, where the function $h(x)$ to be optimized can be expressed as the expectation

$$h(x) = \mathbb{E}[H(x, Z)]\,. \tag{5.8}$$

This assumption is relevant, and useful, in the setup of censoring models:

Example 5.14. Censored data likelihood. Suppose that we observe Y_1, $\cdots$, Y_n, iid, from $f(y - \theta)$ and we have ordered the observations so that $\mathbf{y} = (y_1, \cdots, y_m)$ are uncensored and $(y_{m+1}, \ldots, y_n)$ are censored (and equal to a). The likelihood function is

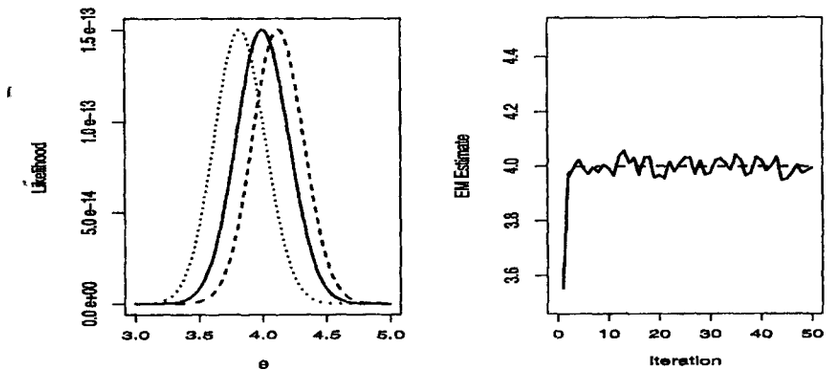


Fig. 5.6. The left panel shows three "likelihoods" of a sample of size 25 from a $\mathcal{N}(4, 1)$. The leftmost is the likelihood of the sample where values greater than 4.5 are replaced by the value 4.5 (dotted), the center (solid) is the observed-data likelihood (5.9), and the rightmost (dashed) is the likelihood using the actual data. The right panel shows EM (dashed) and MCEM (solid) estimates; see Examples 5.17 and 5.20.

$$L(\theta|\mathbf{y}) = [1 - F(a - \theta)]^{n-m} \prod_{i=1}^{m} f(y_i - \theta)\,, \tag{5.9}$$

where F is the cdf associated with f. If we had observed the last $n - m$ values, say $\mathbf{z} = (z_{m+1}, \ldots, z_n)$, with $z_i > a$ $(i = m + 1, \ldots, n)$, we could have constructed the (complete data) likelihood

$$L^c(\theta|\mathbf{y}, \mathbf{z}) = \prod_{i=1}^{m} f(y_i - \theta) \prod_{i=m+1}^{n} f(z_i - \theta),$$

with which it often is easier to work. Note that

$$L(\theta|\mathbf{y}) = \mathbb{E}[L^c(\theta|\mathbf{y}, \mathbf{Z})] = \int_{\mathcal{Z}} L^c(\theta|\mathbf{y}, \mathbf{z}) f(\mathbf{z}|\mathbf{y}, \theta)\, d\mathbf{z},$$

where $f(\mathbf{z}|\mathbf{y}, \theta)$ is the density of the missing data conditional on the observed data. For $f(y - \theta) = \mathcal{N}(\theta, 1)$ three likelihoods are shown in Figure 5.6. Note how the observed-data likelihood is biased down from the true value of θ. ‖

When (5.7) holds, the Z vector merely serves to simplify calculations, and the way Z is selected to satisfy (5.8) should not affect the value of the estimator. This is a *missing data model*, and we refer to the function $L^c(\theta|x, z)) = f(x, z|\theta)$ as the "complete-model" or "complete-data" likelihood, which corresponds to the observation of the *complete data* (x, z). This complete model is often within the exponential family framework, making it much easier to work with (see Problem 5.14).

More generally, we refer to the representation (5.7) as *demarginalization*, a setting where a function (or a density) of interest can be expressed as an integral of a more manageable quantity. We will meet such setups again in Chapters 8–9. They cover models such as missing data models (censoring, grouping, mixing, etc.), latent variable models (tobit, probit, arch, stochastic volatility, etc.) and also artificial embedding, where the variable Z in (5.8) has no meaning for the inferential or optimization problem, as illustrated by *slice sampling* (Chapter 8).

5.3.2 The EM Algorithm

The EM (*Expectation–Maximization*) algorithm was originally introduced by Dempster et al. (1977) to overcome the difficulties in maximizing likelihoods by taking advantage of the representation (5.7) and solving a sequence of easier maximization problems whose limit is the answer to the original problem. It thus fits naturally in this demarginalization section, even though it is not a stochastic algorithm in its original version. Monte Carlo versions are examined in Section 5.3.3 and in Note 5.5.1. Moreover, the EM algorithm relates to MCMC algorithms in the sense that it can be seen as a forerunner of the Gibbs sampler in its Data Augmentation version (Section 10.1.2), replacing simulation by maximization.

Suppose that we observe $X_1, \ldots, X_n$, iid from $g(x|\theta)$ and want to compute $\hat{\theta} = \arg\max L(\theta|\mathbf{x}) = \prod_{i=1}^n g(x_i|\theta)$. We augment the data with $\mathbf{z}$, where $\mathbf{X}, \mathbf{Z} \sim f(\mathbf{x}, \mathbf{z}|\theta)$ and note the identity (which is a basic identity for the EM algorithm)

$$k(\mathbf{z}|\theta, \mathbf{x}) = \frac{f(\mathbf{x}, \mathbf{z}|\theta)}{g(\mathbf{x}|\theta)}, \tag{5.10}$$

where $k(\mathbf{z}|\theta, \mathbf{x})$ is the conditional distribution of the missing data $\mathbf{Z}$ given the observed data $\mathbf{x}$. The identity (5.10) leads to the following relationship between the complete-data likelihood $L^c(\theta|\mathbf{x}, \mathbf{z})$ and the observed-data likelihood $L(\theta|\mathbf{x})$. For any value θ_0,

$$\log L(\theta|\mathbf{x}) = \mathbb{E}_{\theta_0}[\log L^c(\theta|\mathbf{x}, \mathbf{z})] - \mathbb{E}_{\theta_0}[\log k(\mathbf{z}|\theta, \mathbf{x})], \tag{5.11}$$

where the expectation is with respect to $k(\mathbf{z}|\theta_0, \mathbf{x})$. We now see the EM algorithm as a demarginalization model. However, the strength of the EM algorithm is that it can go further. In particular, to maximize $\log L(\theta|\mathbf{x})$, we only have to deal with the first term on the right side of (5.11), as the other term can be ignored.

Common EM notation is to denote the expected log-likelihood by

$$Q(\theta|\theta_0, \mathbf{x}) = \mathbb{E}_{\theta_0}[\log L^c(\theta|\mathbf{x}, \mathbf{z})]. \tag{5.12}$$

We then maximize $Q(\theta|\theta_0, \mathbf{x})$, and if $\hat{\theta}_{(1)}$ is the value of θ maximizing $Q(\theta|\theta_0, \mathbf{x})$, the process can then be repeated with θ_0 replaced by the updated value $\hat{\theta}_{(1)}$. In this manner, a sequence of estimators $\hat{\theta}_{(j)}$, $j = 1, 2, \ldots$,

is obtained iteratively where $\hat{\theta}_{(j)}$ is defined as the value of θ maximizing $Q(\theta|\hat{\theta}_{(j-1)}, \mathbf{x})$; that is,

$$Q(\hat{\theta}_{(j)}|\hat{\theta}_{(j-1)}, \mathbf{x}) = \max_{\theta} Q(\theta|\hat{\theta}_{(j-1)}, \mathbf{x}). \tag{5.13}$$

The iteration described above contains both an expectation step and a maximization step, giving the algorithm its name. At the jth step of the iteration, we calculate the expectation (5.12), with θ_0 replaced by $\hat{\theta}_{(j-1)}$ (*the E-step*), and then maximize it (*the M-step*).

Algorithm A.20 –The EM Algorithm–

1. Compute
$$Q(\theta|\hat{\theta}_{(m)}, \mathbf{x}) = \mathbb{E}_{\hat{\theta}_{(m)}}[\log L^c(\theta|\mathbf{x}, \mathbf{z})],$$
where the expectation is with respect to $k(\mathbf{z}|\hat{\theta}_m, \mathbf{x})$. (*the E-step*)
2. Maximize $Q(\theta|\hat{\theta}_{(m)}, \mathbf{x})$ in θ and take (*the M-step*) [A.20]
$$\hat{\theta}_{(m+1)} = \arg\max_{\theta} Q(\theta|\hat{\theta}_{(m)}, \mathbf{x}).$$

The iterations are conducted until a fixed point of Q is obtained.

The theoretical core of the EM Algorithm is based on the fact that by maximizing $Q(\theta|\hat{\theta}_{(m)}, \mathbf{x})$ at each step, the likelihood on the left side of (5.11) is increased at each step. The following theorem was established by Dempster et al. (1977).

Theorem 5.15. *The sequence $(\hat{\theta}_{(j)})$ defined by (5.13) satisfies*

$$L(\hat{\theta}_{(j+1)}|\mathbf{x}) \geq L(\hat{\theta}_{(j)}|\mathbf{x}),$$

with equality holding if and only if $Q(\hat{\theta}_{(j+1)}|\hat{\theta}_{(j)}, \mathbf{x}) = Q(\hat{\theta}_{(j)}|\hat{\theta}_{(j)}, \mathbf{x})$.

Proof. On successive iterations, it follows from the definition of $\hat{\theta}_{(j+1)}$ that

$$Q(\hat{\theta}_{(j+1)}|\hat{\theta}_{(j)}, \mathbf{x}) \geq Q(\hat{\theta}_{(j)}|\hat{\theta}_{(j)}, \mathbf{x}).$$

Thus, if we can show that

$$\mathbb{E}_{\hat{\theta}_{(j)}}[\log k(\mathbf{Z}|\hat{\theta}_{(j+1)}, \mathbf{x})] \leq \mathbb{E}_{\hat{\theta}_{(j)}}[\log k(\mathbf{Z}|\hat{\theta}_{(j)}, \mathbf{x})], \tag{5.14}$$

it will follow from (5.11) that the value of the likelihood is increased at each iteration.

Since the difference of the logarithms is the logarithm of the ratio, (5.14) can be written as

$$\text{(5.15)}\qquad \mathbb{E}_{\hat\theta_{(j)}}\left[\log\left(\frac{k(\mathbf{Z}|\hat\theta_{(j+1)},\mathbf{x})}{k(\mathbf{Z}|\hat\theta_{(j)},\mathbf{x})}\right)\right] \le \log \mathbb{E}_{\hat\theta_{(j)}}\left[\frac{k(\mathbf{Z}|\hat\theta_{(j+1)},\mathbf{x})}{k(\mathbf{Z}|\hat\theta_{(j)},\mathbf{x})}\right] = 0,$$

where the inequality follows from Jensen's inequality (see Problem 5.15). The theorem is therefore established. □

Although Theorem 5.15 guarantees that the likelihood will increase at each iteration, we still may not be able to conclude that the sequence $(\hat\theta_{(j)})$ converges to a maximum likelihood estimator. To ensure convergence we require further conditions on the mapping $\hat\theta_{(j)} \to \hat\theta_{(j+1)}$. These conditions are investigated by Boyles (1983) and Wu (1983). The following theorem is, perhaps, the most easily applicable condition to guarantee convergence to a *stationary point*, a zero of the first derivative that may be a local maximum or saddlepoint.

Theorem 5.16. *If the expected complete-data likelihood $Q(\theta|\theta_0,\mathbf{x})$ is continuous in both θ and θ_0, then every limit point of an EM sequence $(\hat\theta_{(j)})$ is a stationary point of $L(\theta|\mathbf{x})$, and $L(\hat\theta_{(j)}|\mathbf{x})$ converges monotonically to $L(\hat\theta|\mathbf{x})$ for some stationary point $\hat\theta$.*

Note that convergence is only guaranteed to a stationary point. Techniques such as running the EM algorithm a number of times with different, random starting points, or algorithms such as *simulated annealing* (see, for example, Finch et al. 1989) attempt to give some assurance that the global maximum is found. Wu (1983) states another theorem that guarantees convergence to a local maximum, but its assumptions are difficult to check. It is usually better, in practice, to use empirical methods (graphical or multiple starting values) to check that a maximum has been reached.

As a first example, we look at the censored data likelihood of Example 5.14.

Example 5.17. EM for censored data. For $Y_i \sim \mathcal{N}(\theta,1)$, with censoring at a, the complete-data likelihood is

$$L^c(\theta|\mathbf{y},\mathbf{z}) \propto \prod_{i=1}^{m}\exp\{-(y_i-\theta)^2/2)\}\prod_{i=m+1}^{n}\exp\{(z_i-\theta)^2/2\}.$$

The density of of the missing data $\mathbf{z} = (z_{n-m+1},\dots,z_n)$ is a truncated normal

$$\text{(5.16)}\qquad \mathbf{Z} \sim k(\mathbf{z}|\theta,by) = \frac{1}{(2\pi)^{(n-m)/2}}\exp\left\{\sum_{i=m+1}^{n}(z_i-\theta)^2/2\right\},$$

resulting in the expected complete-data log likekihood

$$-\frac{1}{2}\sum_{i=1}^{m}(y_i-\theta)^2 - \frac{1}{2}\sum_{i=n-m+1}^{n}\mathbb{E}_{\theta'}[(Z_i-\theta)^2].$$

Before evaluating the expectation we differentiate and set equal to zero, solving for the EM estimate

$$\hat\theta = \frac{m\bar y + (n-m)\mathbb{E}_{\theta'}(Z_1)}{n}.$$

This leads to the EM sequence

$$\text{(5.17)}\qquad \hat\theta^{(j+1)} = \frac{m}{n}\bar y + \frac{n-m}{n}\hat\theta^{(j)} + \frac{1}{n}\frac{\phi(a-\hat\theta^{(j)})}{1-\Phi(a-\hat\theta^{(j)})},$$

where ϕ and Φ are the normal pdf and cdf, respectively. (See Problem 5.16 for details of the calculations.)

The EM sequence is shown in the right panel of Figure 5.6. The convergence in this problem is quite rapid, giving an MLE of 3.99, in contrast to the observed data mean of 3.55. ‖

The following example has a somewhat more complex model.

Example 5.18. Cellular phone plans. A clear case of missing data occurs in the following estimation problem. It is typical for cellular phone companies to offer "plans" of options, bundling together four or five options (such as messaging, caller id, etc.) for one price, or, alternatively, selling them separately. One cellular company had offered a four-option plan in some areas, and a five-option plan (which included the four, plus one more) in another area.

In each area, customers were ask to choose their favorite plan, and the results were tabulated. In some areas they choose their favorite from four plans, and in some areas from five plans. The phone company is interested in knowing which are the popular plans, to help them set future prices. A portion of the data are given in Table 5.6. We can model the complete data as follows. In area i, there are n_i customers, each of whom choses their favorite plan from Plans 1 – 5. The observation for customer i is $Z_i = (Z_{i1},\dots,Z_{i5})$, where Z_i is $\mathcal{M}(1,(p_1,p_2,\dots,p_5))$. If we assume the customers are independent, in area i the data are $T_i = (T_{i1},\dots,T_{i5}) = \sum_{j=1}^{n_i} Z_i \sim \mathcal{M}(n_i,(p_1,p_2,\dots,p_5))$ (see Problem 5.23). If the first m observations have the Z_{i5} missing, denote the missing data by x_i and then we have the complete-data likelihood

(5.18)

$$L(\mathbf{p}|\mathbf{T},\mathbf{x}) = \prod_{i=1}^{m}\binom{n_i+x_i}{T_{i1},\dots,T_{i4},x_i}p_1^{T_{i1}}\cdots p_4^{T_{i4}}p_5^{x_i}\times\prod_{i=m+1}^{n}\binom{n_i}{T_{i1},\dots,T_{i5}}\prod_{j=1}^{5}p_j^{T_{ij}},$$

where $\mathbf{p} = (p_1,p_2,\dots,p_5)$, $\mathbf{T} = (T_1,T_2,\dots,T_5)$, $\mathbf{x} = (x_1,x_2,\dots,x_m)$, and $\binom{n}{n_1,n_2,\dots,n_k}$ is the multinomial coefficient $\frac{n!}{n_1!n_2!\cdots n_k!}$. The observed-data likelihood can be calculated as $L(\mathbf{p}|\mathbf{T}) = \sum_{\mathbf{x}} L(\mathbf{p}|\mathbf{T},\mathbf{x})$ leading to the missing data distribution

	Plan						Plan				
	1	2	3	4	5		1	2	3	4	5
1	26	63	20	0	–	20	56	18	29	5	–
2	31	14	16	51	–	21	27	53	10	0	–
3	41	28	34	10	–	22	47	29	4	11	–
4	27	29	19	25	–	23	43	66	6	1	–
5	26	48	41	0	–	24	14	30	23	23	6
6	30	45	12	14	–	25	4	24	24	32	7
7	53	39	12	11	–	26	11	30	22	23	8
8	40	25	26	6	–	27	0	55	0	28	10
9	37	26	15	0	–	28	1	20	25	33	12
10	27	59	17	0	–	29	29	25	5	34	13
11	32	39	12	0	–	30	23	71	10	0	13
12	28	46	17	12	–	31	53	29	1	5	18
13	27	5	44	3	–	32	9	31	22	20	19
14	43	51	11	5	–	33	0	49	6	32	20
15	52	30	17	0	–	34	21	25	12	37	20
16	31	55	21	0	–	35	35	26	18	0	25
17	54	51	0	0	–	36	8	15	4	54	32
18	34	29	32	10	–	37	39	19	3	8	38
19	42	33	7	13	–						

Table 5.6. Cellular phone plan preferences in 37 areas: Data are number of customers who choose the particular plan as their favorite. Some areas ranked 4 plans (with the 5^{th} plan denoted by –) and some ranked 5 plans.

$$k(\mathbf{x}|\mathbf{T},\mathbf{p}) = \prod_{i=1}^{m} \binom{n_i + x_i}{x_i} p_5^{x_i} (1-p_5)^{n_i+1}, \tag{5.19}$$

a product of negative binomial distributions.

The rest of the EM analysis follows. Define $W_j = \sum_{i=1}^{n} T_{ij}$ for $j = 1, \ldots, 4$, and $W_5 = \sum_{i=1}^{m} T_{i5}$ for $j = 5$. The expected complete-data log likelihood is

$$\sum_{j=1}^{4} W_j \log p_j + [W_5 + \sum_{i=1}^{m} \mathbb{E}(X_i|\mathbf{p}')] \log(1 - p_1 - p_2 - p_3 - p_4),$$

leading to the EM iterations

$$\mathbb{E}(X_i|\mathbf{p}^{(t)}) = (n_i + 1)\frac{\hat{p}_5^{(t)}}{1 - \hat{p}_5^{(t)}}, \quad \hat{p}_j^{(t+1)} = \frac{W_j}{\sum_{i=1}^{m} \mathbb{E}(X_i|\mathbf{p}^{(t)}) + \sum_{j'=1}^{5} W_{j'}}$$

for $j = 1, \ldots, 4$. The MLE of $\mathbf{p}$ is $(0.273, 0.329, 0.148, 0.125, 0.125)$, with convergence being very rapid. Convergence of the estimators is shown in Figure 5.7, and further details are given in Problem 5.23. (See also Example 9.22 for a Gibbs sampling treatment of this problem.) ‖

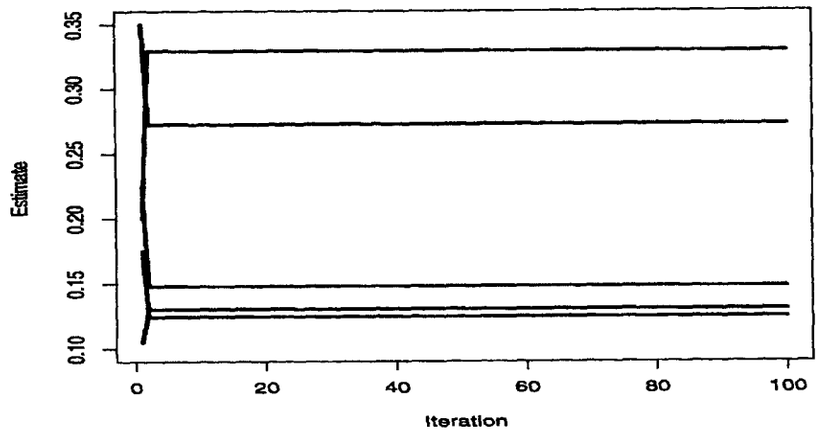


Fig. 5.7. EM sequence for cellular phone data, 25 iterations

One weakness of EM is that it is "greedy"; it always moves upward to a higher value of the likelihood. This means that it cannot escape from a local model. The following example illustrates the possible convergence of EM to the "wrong" mode, even in a well-behaved case.

Example 5.19. EM for mean mixtures of normal distributions Consider the mixture of two normal distributions already introduced in Example 1.10,

$$p\mathcal{N}(\mu_1, \sigma^2) + (1-p)\mathcal{N}(\mu_2, \sigma^2),$$

in the special case where all parameters but (μ_1, μ_2) are known. Figure 5.8 *(bottom)* shows the log-likelihood surface associated with this model and 500 observations simulated with $p = 0.7$, $\sigma = 1$ and $(\mu_1, \mu_2) = (0, 3.1)$. As easily seen from this surface, the likelihood is bimodal,[5] with one mode located near the true value of the parameters, and another located at $(2, -.5)$. Running EM five times with various starting points chosen at random, we represent the corresponding occurences: three out of five sequences are attracted by the higher mode, while the other two go to the lower mode (even though the likelihood is considerably smaller). This is because the starting points happened to be in the domain of attraction of the lower mode. We also represented in Figure 5.8 *(top)* the corresponding (increasing) sequence of log-likelihood values taken by the sequence. Note that in a very few iterations the value

[5] Note that this is not a special occurrence associated with a particular sample: there are two modes of the likelihood for every simulation of a sample from this mixture, even though the model is completely identifiable.

is close to the modal value, with improvement brought by further iterations being incremental. ‖

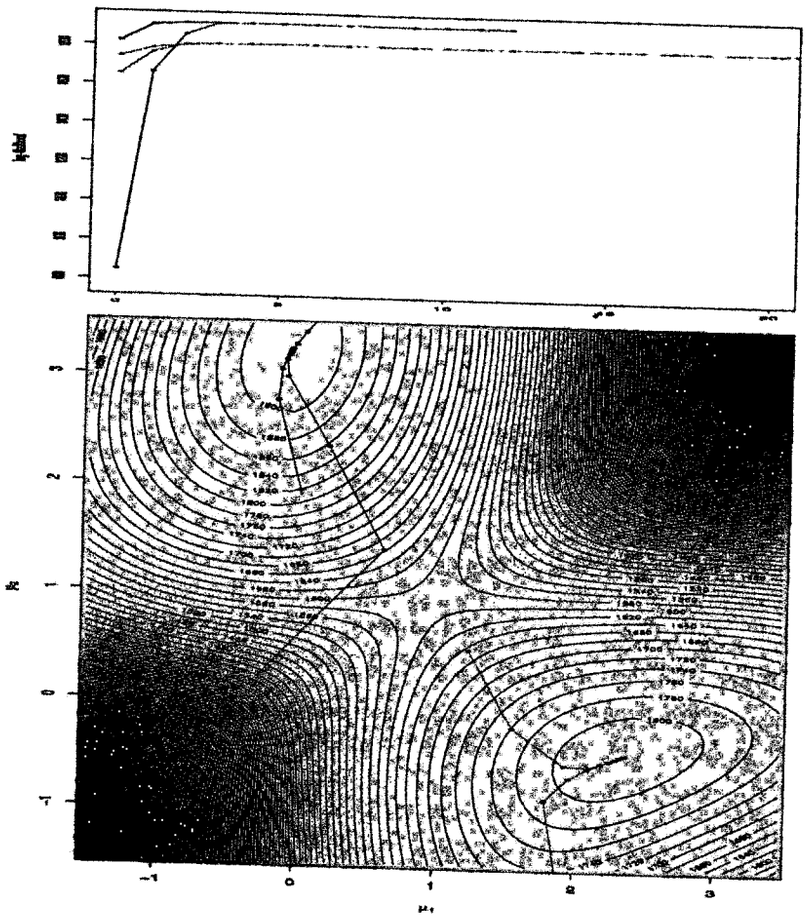


Fig. 5.8. Trajectories of five runs of the EM algorithm for Example 5.19 with their likelihood values *(top)* and their position on the likelihood surface *(bottom)*.

This example reinforces the case of the need for rerunning the algorithm a number of times, each time starting from a different initial value.

The books by Little and Rubin (1987) and Tanner (1996) provide good overviews of the EM literature. Other references include Louis (1982), Little and Rubin (1983), Laird et al. (1987), Meng and Rubin (1991), Qian and Titterington (1992), Liu and Rubin (1994), MacLachlan and Krishnan (1997), and Meng and van Dyk (1997).

5.3.3 Monte Carlo EM

A difficulty with the implementation of the EM algorithm is that each "E-step" requires the computation of the expected log likelihood $Q(\theta|\theta_0, \mathbf{x})$. Wei and Tanner (1990a,b) propose to use a Monte Carlo approach (MCEM) to overcome this difficulty, by simulating $Z_1, \ldots, Z_m$ from the conditional distribution $k(\mathbf{z}|\mathbf{x}, \theta)$ and then maximizing the approximate complete-data log-likelihood

$$\hat{Q}(\theta|\theta_0, \mathbf{x}) = \frac{1}{m}\sum_{i=1}^{m} \log L^c(\theta|\mathbf{x}, \mathbf{z}) \,. \tag{5.20}$$

When m goes to infinity, this quantity indeed converges to $Q(\theta|\theta_0, \mathbf{x})$, and the limiting form of the *Monte Carlo EM* algorithm is thus the regular EM algorithm. The authors suggest that m should be increased along with the iterations. Although the maximization of a sum like (5.20) is, in general, rather involved, exponential family settings often allow for closed-form solutions.

Example 5.20. MCEM for censored data. The EM solution of Example 5.17 can easily become an MCEM solution. For the EM sequence

$$\hat{\theta}^{(j+1)} = \frac{m\bar{y} + (n-m)\mathbb{E}_{\hat{\theta}^{(j)}}(Z_1)}{n},$$

the MCEM solution replaces $\mathbb{E}_{\theta^{(j)}}(Z_1)$ with

$$\frac{1}{M}\sum_{i=1}^{M} Z_i, \quad Z_i \sim k(z|\hat{\theta}^{(j)}, \mathbf{y}).$$

The MCEM sequence is shown in the right panel of Figure 5.6. The convergence is not quite so rapid as EM. The variability is controlled by the choice of M, and a larger value would bring the sequences closer together. ‖

Example 5.21. Genetic linkage. A classic (perhaps overused) example of the EM algorithm is the genetics problem (see Rao 1973, Dempster et al. 1977, or Tanner 1996), where observations (x_1, x_2, x_3, x_4) are gathered from the multinomial distribution

$$\mathcal{M}\left(n; \frac{1}{2} + \frac{\theta}{4}, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{\theta}{4}\right).$$

Estimation is easier if the x_1 cell is split into two cells, so we create the augmented model

$$(z_1, z_2, x_2, x_3, x_4) \sim \mathcal{M}\left(n; \frac{1}{2}, \frac{\theta}{4}, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{\theta}{4}\right),$$

with $x_1 = z_1 + z_2$. The complete-data likelihood function is then simply $\theta^{x_2+x_4}(1-\theta)^{x_2+x_3}$, as opposed to the observed-data likelihood function $(2+\theta)^{x_1}\theta^{x_4}(1-\theta)^{x_2+x_3}$. The expected complete log-likelihood function is

$$\begin{aligned}&\mathbb{E}_{\theta_0}[(Z_2 + x_4)\log\theta + (x_2 + x_3)\log(1-\theta)] \\ &\quad = \left(\frac{\theta_0}{2+\theta_0}x_1 + x_4\right)\log\theta + (x_2 + x_3)\log(1-\theta),\end{aligned}$$

which can easily be maximized in θ, leading to

$$\hat{\theta}_1 = \frac{\dfrac{\theta_0\, x_1}{2+\theta_0} + x_4}{\dfrac{\theta_0\, x_1}{2+\theta_0} + x_2 + x_3 + x_4}.$$

If we instead use the Monte Carlo EM algorithm, $\theta_0\, x_1/(2+\theta_0)$ is replaced with the average

$$\overline{z}_m = \frac{1}{m}\sum_{i=1}^{m} z_i,$$

where the z_i's are simulated from a binomial distribution $\mathcal{B}(x_1, \theta_0/(2+\theta_0))$. The maximum in θ is then

$$\theta_1 = \frac{\overline{z}_m + x_4}{\overline{z}_m + x_2 + x_3 + x_4}.$$

‖

This example is merely an illustration of the Monte Carlo EM algorithm since EM also applies. The next example, however, details a situation in which the expectation is quite complicated and the Monte Carlo EM algorithm works quite nicely.

Example 5.22. Capture–recapture models revisited. A generalization of a capture–recapture model (see Example 2.25) is to assume that an animal i, $i = 1, 2, \ldots, n$ can be captured at time j, $j = 1, 2, \ldots, t$, in one of m locations, where the location is a multinomial random variable

$$H \sim \mathcal{M}_m(\theta_1, \ldots, \theta_m).$$

Of course, the animal may not be captured (it may not be seen, or it may have died). As we track each animal through time, we can model this process with two random variables. The random variable H can take values in the set $\{1, 2, \ldots, m\}$ with probabilities $\{\theta_1, \ldots, \theta_m\}$. Given $H = k$, the random variable is $X \sim \mathcal{B}(p_k)$, where p_k is the probability of capturing the animal in location k. (See Dupuis 1995 for details.)

As an example, for $t = 6$, a typical realization for an animal might be

$$\mathbf{h} = (4, 1, -, 8, 3, -), \quad \mathbf{x} = (1, 1, 0, 1, 1, 0),$$

where $\mathbf{h}$ denotes the sequence of observed h_j's and of non-captures. Thus, we have a missing data problem. If we had observed all of $\mathbf{h}$, the maximum likelihood estimation would be trivial, as the MLEs would simply be the cell means. For animal i, we define the random variables $X_{ijk} = 1$ if animal i is captured at time j in location k, and 0 otherwise,

$$Y_{ijk} = \mathbb{I}(H_{ij} = k)\mathbb{I}(X_{ijk} = 1)$$

(which is the observed data), and

$$Z_{ijk} = \mathbb{I}(H_{ij} = k)\mathbb{I}(X_{ijk} = 0)$$

(which is the missing data). The likelihood function is

$$\begin{aligned}&L(\theta_1, \ldots, \theta_m, p_1, \ldots, p_m | \mathbf{y}, \mathbf{x}) \\ &\quad = \sum_{\mathbf{z}} L(\theta_1, \ldots, \theta_m, p_1, \ldots, p_m | \mathbf{y}, \mathbf{x}, \mathbf{z}) \\ &\quad = \sum_{\mathbf{z}} \prod_{k=1}^{m} p_k^{\sum_{i=1}^{n}\sum_{j=1}^{t} x_{ijk}} \\ &\qquad \times (1-p_k)^{nt - \sum_{i=1}^{n}\sum_{j=1}^{t} x_{ijk}} \theta_k^{\sum_{i=1}^{n}\sum_{j=1}^{t}(y_{ijk} + z_{ijk})},\end{aligned}$$

where the sum over $\mathbf{z}$ represents the expectation over all the states that could have been visited. This can be a complicated expectation, but the likelihood can be calculated by first using an EM strategy and working with the complete data likelihood $L(\theta_1, \ldots, \theta_k, p_1, \ldots, p_k | \mathbf{y}, \mathbf{x}, \mathbf{z})$, then using MCEM for the calculation of the expectation. Note that calculation of the MLEs of $p_1, \ldots, p_k$ is straightforward, and for $\theta_1, \ldots, \theta_k$, we use

Algorithm A.21 –Capture–recapture MCEM Algorithm–

1. *(M-step)* Take $\hat{\theta}_k = \frac{1}{nt}\sum_{i=1}^{n}\sum_{j=1}^{t} y_{ijk} + z_{ijk}$.
2. *(Monte Carlo E-step)* If $z_{ijk} = 0$, for $l = 1, \ldots, L$, generate

$$z_{ijkl} \sim \mathcal{M}_k(\hat{\theta}_1, \ldots, \hat{\theta}_m),$$

and calculate

$$z_{ijk} = \sum_{l} z_{ijkl}/L.$$

Scherrer (1997) examines the performance of more general versions of this algorithm and shows, in particular, that they outperform the conditional likelihood approach of Brownie et al. (1993). ‖

Note that the MCEM approach does not enjoy the EM monotonicity any longer and may even face some smoothness difficulties when the sample used in

(5.20) is different for each new value θ_j. In more involved likelihoods, Markov chain Monte Carlo methods can also be used to generate the missing data sample, usually creating additional dependence structures between the successive values produced by the algorithm.

5.3.4 EM Standard Errors

There are many algorithms and formulas available for obtaining standard errors from the EM algorithm (see Tanner 1996 for a selection). However, the formulation by Oakes (1999), and its Monte Carlo version, seem both simple and useful.

Recall that the variance of the MLE, is approximated by

$$\text{Var}\,\hat{\theta} \approx \left[\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x})\right]^{-1}.$$

Oakes (1999) shows that this second derivative can be expressed in terms of the complete-data likelihood

$$\begin{aligned}(5.21)\qquad &\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x})\\ &= \left\{\frac{\partial^2}{\partial\theta'^2}\mathbb{E}[\log L(\theta'|\mathbf{x},\mathbf{z})] + \frac{\partial^2}{\partial\theta'\partial\theta}\mathbb{E}[\log L(\theta'|\mathbf{x},\mathbf{z})]\right\}\Big|_{\theta'=\theta},\end{aligned}$$

where the expectation is taken with respect to the distribution of the missing data. Thus, for the EM algorithm, we have a formula for the variance of the MLE.

The advantage of this expression is that it only involves the distribution of the complete data, which is often a reasonable distribution to work with. The disadvantage is that the mixed derivative may be difficult to compute. However, in complexity of implementation it compares favorably with other methods.

For the Monte Carlo EM algorithm, (5.21) cannot be used in its current form, as we would want all expectations to be on the outside. Then we could calculate the expression using a Monte Carlo sum. However, if we now take the derivative inside the expectation, we can rewrite Oakes' identity as

$$\begin{aligned}(5.22)\qquad &\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x})\\ &= \mathbb{E}\left(\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x},\mathbf{z})\right)\\ &\quad +\mathbb{E}\left[\left(\frac{\partial}{\partial\theta}\log L(\theta|\mathbf{x},\mathbf{z})\right)^2\right] - \left[\mathbb{E}\left(\frac{\partial}{\partial\theta}\log L(\theta|\mathbf{x},\mathbf{z})\right)\right]^2,\end{aligned}$$

which is better suited for simulation, as all expectations are on the outside. Equation (5.22) can be expressed in the rather pleasing form

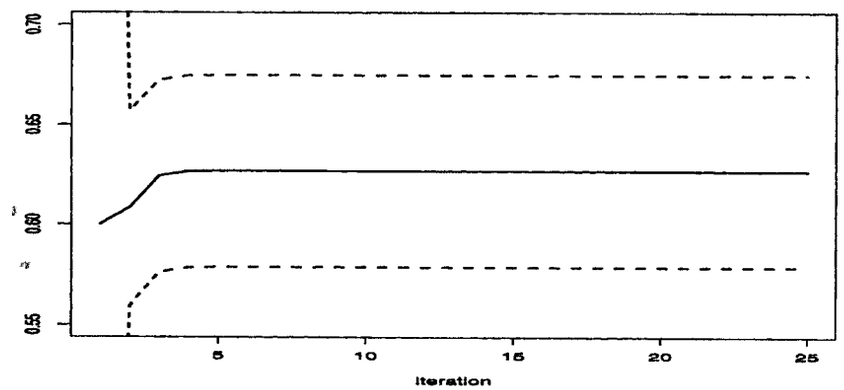


Fig. 5.9. EM sequence ± one standard deviation and for genetic linkage data, 25 iterations

$$\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x}) = \mathbb{E}\left(\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x},\mathbf{z})\right) + \text{var}\left(\frac{\partial}{\partial\theta}\log L(\theta|\mathbf{x},\mathbf{z})\right),$$

which allows the Monte Carlo evaluation

$$\begin{aligned}&\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x})\\ &= \frac{1}{M}\sum_{j=1}^{M}\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x},\mathbf{z}^{(j)})\\ &\quad +\frac{1}{M}\sum_{j=1}^{M}\left(\frac{\partial}{\partial\theta}\log L(\theta|\mathbf{x},\mathbf{z}^{(j)}) - \frac{1}{M}\sum_{j'=1}^{M}\frac{\partial}{\partial\theta}\log L(\theta|\mathbf{x},\mathbf{z}^{(j')})\right)^2,\end{aligned}$$

where $(\mathbf{z}^{(j)})$, $j = 1,\ldots,M$ are generated from the missing data distribution (and have already been generated to do MCEM).

Example 5.23. Genetic linkage standard errors. For Example 5.21, the complete-data likelihood is $\theta^{z_2+x_4}(1-\theta)^{x_2+x_3}$, and applying (5.22) yields

$$\frac{\partial^2}{\partial\theta^2}\log L(\theta|\mathbf{x}) = \frac{\mathbb{E}Z_2(1-\theta)^2 + x_4(1-\theta)^2 + (x_2+x_3)\theta^2}{\theta^2(1-\theta)^2},$$

where $\mathbb{E}Z_2 = x_1\theta/(2+\theta)$. In practice, we evaluate the expectation at the converged value of the likelihood estimator. For these data we obtain $\hat{\theta} = .627$ with standard deviation .048. Figure 5.9 shows the evolution of the EM estimate and the standard error. ‖