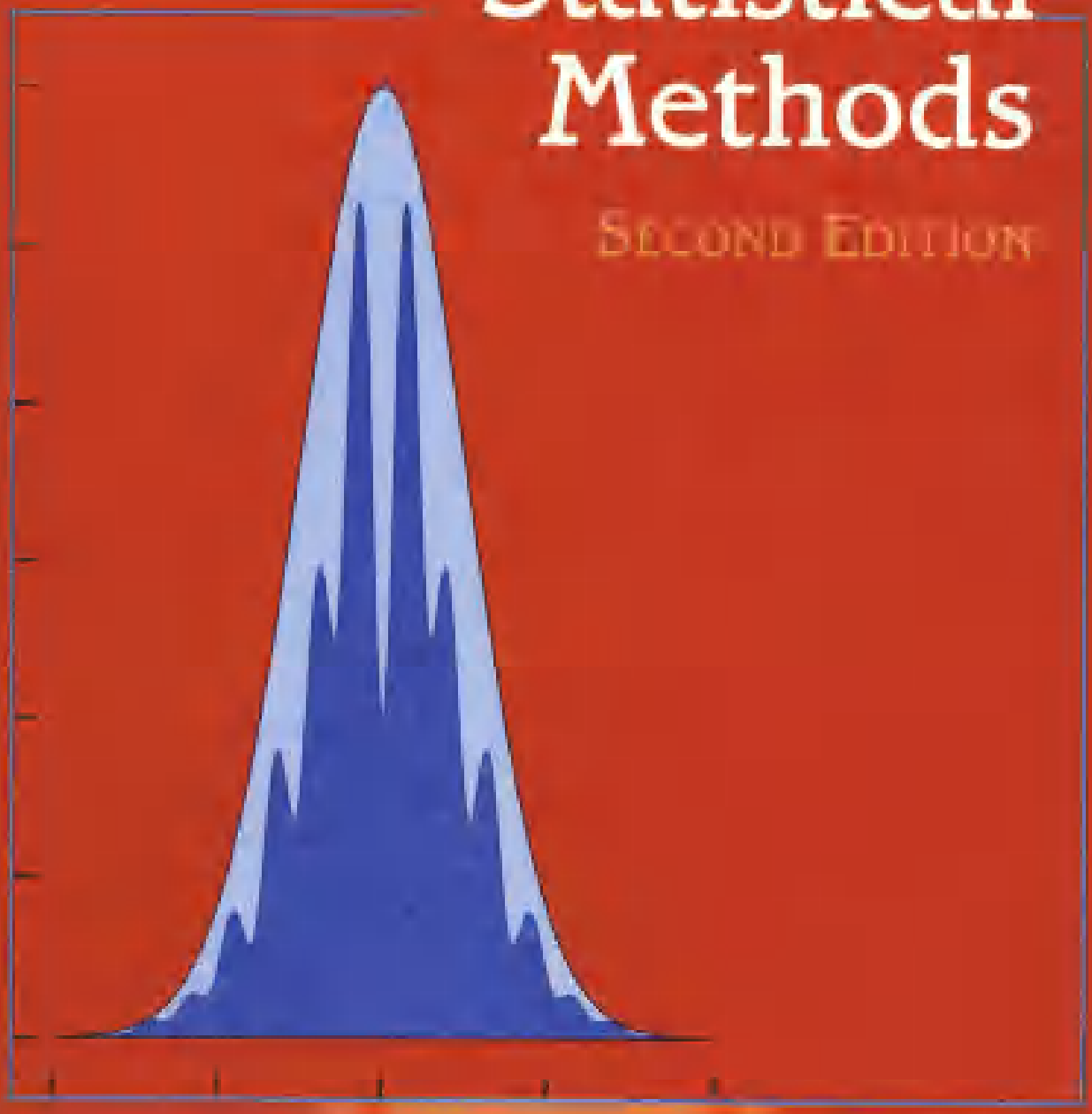
SPRINGER TEXTS IN STATISTICS

Monte Carlo Statistical Methods

SECOND EDITION



Christian P. Robert
George Casella

Springer

Springer Texts in Statistics

Advisors:
George Casella · Stephen Fienberg · Ingram Olkin

Springer Texts in Statistics

Alfred: Elements of Statistics for the Life and Social Sciences

Berger: An Introduction to Probability and Stochastic Processes

Bilodeau and Brenner: Theory of Multivariate Statistics

Blom: Probability and Statistics: Theory and Applications

Brockwell and Davis: Introduction to Times Series and Forecasting, Second Edition

Chow and Teicher: Probability Theory: Independence, Interchangeability, Martingales, Third Edition

Christensen: Advanced Linear Modeling: Multivariate, Time Series, and Spatial Data; Nonparametric Regression and Response Surface Maximization, Second Edition

Christensen: Log-Linear Models and Logistic Regression, Second Edition

Christensen: Plane Answers to Complex Questions: The Theory of Linear Models, Third Edition

Creighton: A First Course in Probability Models and Statistical Inference

Dean and Voss: Design and Analysis of Experiments

du Toit, Steyn, and Stumpf: Graphical Exploratory Data Analysis

Durrett: Essentials of Stochastic Processes

Edwards: Introduction to Graphical Modelling, Second Edition

Finkelstein and Levin: Statistics for Lawyers

Flury: A First Course in Multivariate Statistics

Heiberger and Holland: Statistical Analysis and Data Display: An Intermediate Course with Examples in S-PLUS, R, and SAS

Jobson: Applied Multivariate Data Analysis, Volume I: Regression and Experimental Design

Jobson: Applied Multivariate Data Analysis, Volume II: Categorical and Multivariate Methods

Karr: Probability, Springer Texts in Statistics, Second Edition

Kulkarni and Harman: An Elementary Introduction to Statistical Learning Theory

Lange: Applied Probability, Second Edition

Lehmann: Elements of Large-Sample Theory

Lehmann and Romano: Testing Statistical Hypotheses, Third Edition

Lehmann and Casella: Theory of Point Estimation, Second Edition

Lindman: Analysis of Variance in Experimental Design

Lindsey: Models for Repeated Measurements

Madansky: Prescriptions for Working Statisticians

McPherson: Statistics: An Introduction, Second Edition

Nguyen and Rogers: Fundamentals of Mathematical Statistics: Volume I: Probability, Second Edition

Nguyen and Rogers: Fundamentals of Mathematical Statistics: Volume II: Statistical Inference, Second Edition

Nolan and Speed: Stat Labs: Mathematical Statistics through Applications

Santner and Duffy: The Statistical Analysis of Discrete Data

Saville and Wood: Statistical Methods: The Geometric Approach, Second Edition

Sen and Srivastava: Regression Analysis: Theory, Methods, and Applications, Second Edition

Terrell: Mathematical Statistics: A Unified Introduction

Timerman: Introduction to Harmonic Analysis and Generalized Functions

Whittle: Probability via Expectation, Fourth Edition

Zacks: Introduction to Reliability Analysis: Probability Models and Statistical Methods

Christian P. Robert George Casella

Monte Carlo Statistical Methods

Second Edition

With 150 Illustrations

Springer

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

[illegible]
Department of Statistics
[illegible]
[illegible]
[illegible]
USA
[illegible]

[illegible]

[illegible]
Department of Statistics
[illegible]
[illegible]
USA

[illegible]
Department of Statistics
[illegible]
[illegible]
USA

[illegible]
Department of Statistics
[illegible]
[illegible]
USA

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Preface to the Second Edition

Preface to the First Edition

[illegible]

[illegible]

Contents

Preface to the Second Edition IX

Preface to the First Edition XIII

1 Introduction 1

2.6 Notes [illegible]
2.6.1 The Kiss Problem [illegible]
2.6.2 [illegible] [illegible]
2.6.3 Notes [illegible]

3 Monte Carlo Integration [illegible]
3.1 Introduction [illegible]
3.2 Classical Monte Carlo Integration [illegible]
3.3 Importance Sampling [illegible]
3.3.1 Theory [illegible]
3.3.2 Optimal Variance Reduction [illegible]
3.3.3 Importance Sampling Applied to Monte Carlo [illegible]
3.3.4 Applications [illegible]
3.4 Conclusion [illegible]
3.5 Notes [illegible]
3.5.1 [illegible] [illegible]
3.5.2 [illegible] [illegible]

4 Controlling Monte Carlo Variance [illegible]
4.1 Monte Carlo Variance [illegible]
4.1.1 Theory [illegible]
4.1.2 Monte Carlo Models [illegible]
4.2 The Weak Law [illegible]
4.3 Monte Carlo Approximations [illegible]
4.4 Antithetic Variates [illegible]
4.4.1 Antithetic Variates [illegible]
4.4.2 Control Variates [illegible]
4.5 Conclusion [illegible]
4.6 Notes [illegible]
4.6.1 Monte Carlo [illegible] Variance [illegible]
4.6.2 Antithetic Variates [illegible]
4.6.3 Notes [illegible]

5 Monte Carlo Optimization [illegible]
5.1 Introduction [illegible]
5.2 Simulated Annealing [illegible]
5.2.1 Algorithm [illegible]
5.2.2 Cooling Schedule [illegible]
5.2.3 Simulation [illegible]
5.2.4 Implementation [illegible]
5.3 Simulated Annealing [illegible]
5.3.1 Monte Carlo [illegible] [illegible]
5.3.2 [illegible] Variance [illegible]
5.3.3 Monte Carlo [illegible]
5.3.4 [illegible] [illegible]

[illegible]

6 Markov Chains [illegible]

[illegible]

7 The Metropolis Hastings Algorithm [illegible]

[illegible]

[illegible]
[illegible] Bibliography [illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

8 The Slice Sampler [illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

9 The Two Stage Gibbs Sampler [illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

10 The Multi-Stage Gibbs Sampler [illegible]
[illegible]
[illegible]
[illegible]

[illegible]

11 Variable Dimension Models and Reversible Jump Algorithms [illegible]

[illegible]

12 Diagnosing Convergence [illegible]

[illegible]

[illegible]

13 Perfect Sampling [illegible]

14 Iterated and Sequential Importance Sampling [illegible]

[illegible]

A Probability Distributions [illegible]

B Notation [illegible]

References [illegible]

Index of Notation [illegible]

Index of Subjects [illegible]

List of Tables

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

List of Figures

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

1

Introduction

1.1 Statistical Models

[illegible]

[illegible]

Example [illegible]. Consumed data models. [illegible]

[illegible]

[illegible]

- [illegible]
- [illegible]
- [illegible]

Example 1.2. Mixture models. [illegible]

$$[illegible]$$

[illegible]

$$\prod [illegible]$$

[illegible]

Example 1.3. Moving average models. [illegible]

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[

[illegible]

[illegible]

$$[illegible] = \prod [illegible]$$

[illegible]

Example 1.1. Gerstner [illegible]. [illegible]

$$[illegible]$$

[illegible]

$$[illegible] \prod [illegible]$$

$$[illegible] \prod [illegible]$$

$$[illegible] \sum [illegible] \sum [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 1.5. Least squares estimators. [illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 1.6. Normal MLE. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 1.7. Beta MLE. [illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

Example [illegible]. Non-normality parameters. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Example 1.9. Student's t-distribution. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Fig. [illegible] [illegible]

Example 7.10. (Continuation of Example 7.2) [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

1.3 Bayesian Methods

[illegible]

1.1 [illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 1.11. Binomial Bayes estimator. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Example [illegible]. (Continuation of Example 1.8). [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 1.13. Logistic regression. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Table 1.1. [illegible]

[illegible]

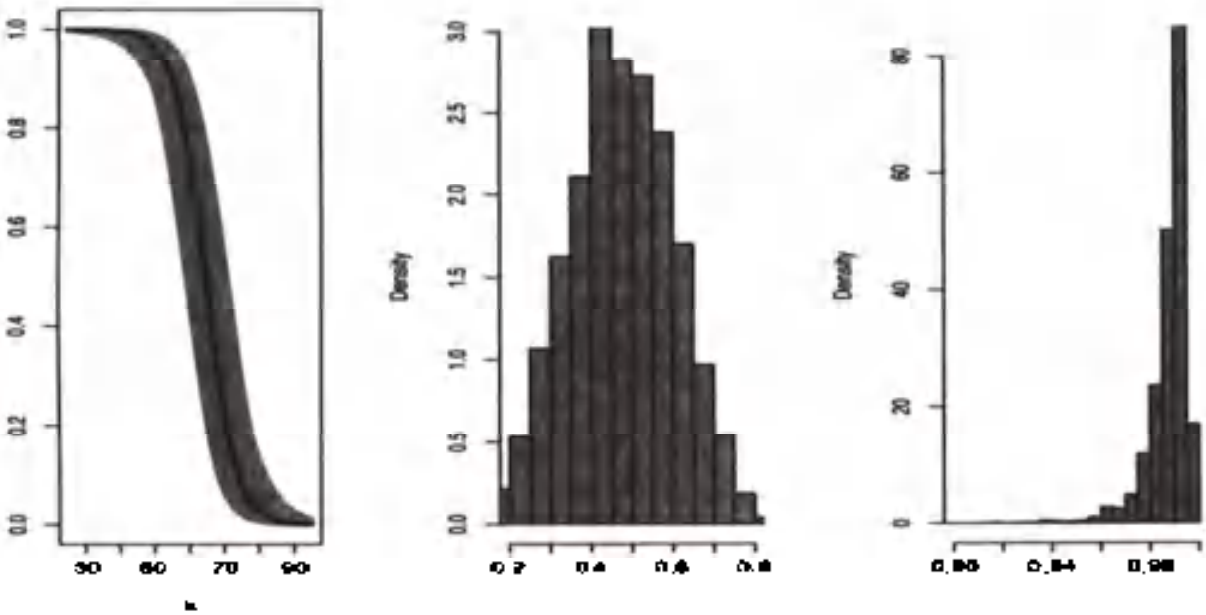


Fig. 1.2. The figure shows the result of 10,000 Monte Carlo simulations of the model (1.21). The left panel shows the average logistic function and variation, the middle panel shows predictions of failure probabilities at 65° Fahrenheit, and the right panel shows predictions of failure probabilities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

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

[illegible]

[illegible]

Example 1.14. Bayes credible regions. [illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 1.15. Chernoff confidence regions. [illegible]

II

[illegible]

[illegible]

[illegible]

Example 1.1b. Linear calibration. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

1.4 Deterministic Numerical Methods

[illegible]

1.4.1 Optimization

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

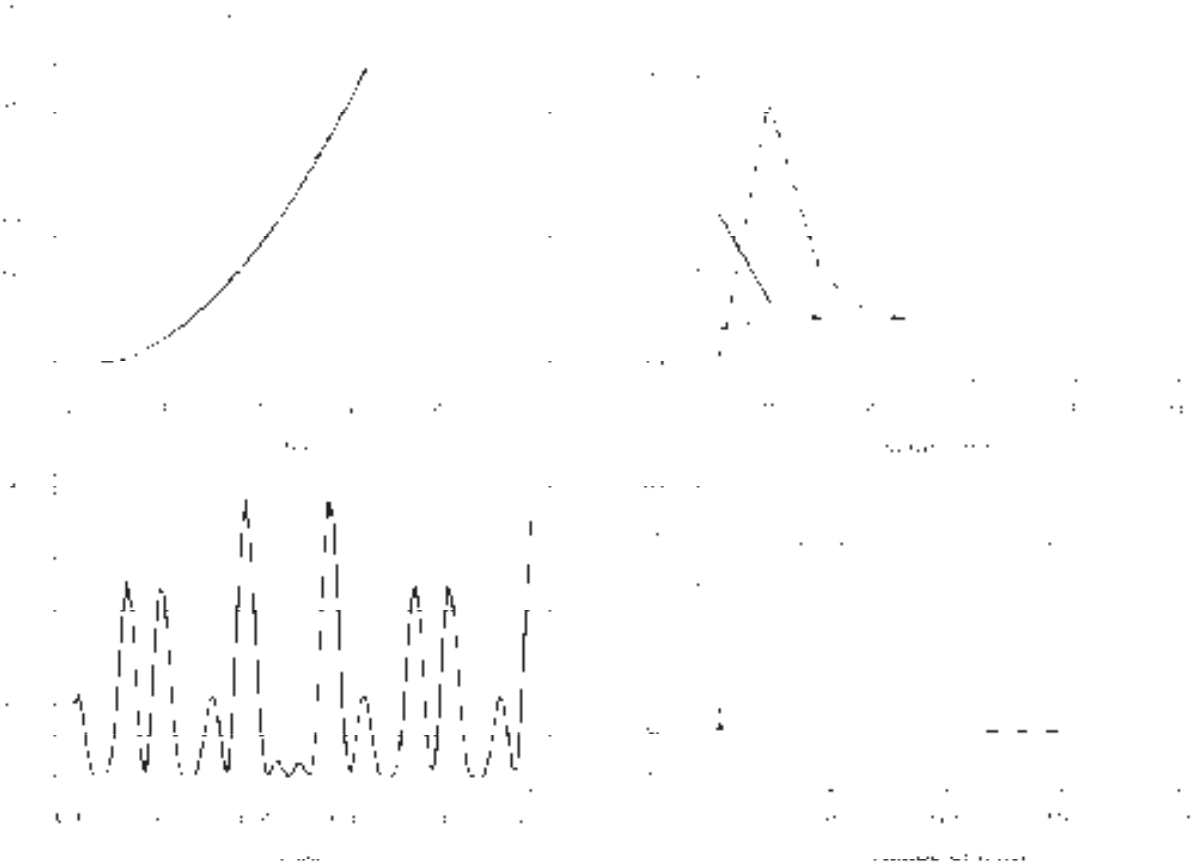
$$[illegible]$$

Example 1.17. A Simple Newton–Raphson Algorithm. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

1.1.2 Integration

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

1.1.3 Comparison

[illegible]

[illegible]

1.5 Problems

1.1 [illegible]

1.2 [illegible]

1.3 [illegible]

1.4 [illegible]

$$[illegible]$$

[illegible]

1.5 [illegible]

1.6 [illegible]

1.7 [illegible]

1.8 [illegible]

$$[illegible]$$

[illegible]

1.9 [illegible]

1.10 [illegible]

1.11 [illegible]

[illegible]

[illegible]

1.12 [illegible]

1.13 [illegible]

[illegible]

[illegible]

1.14 [illegible]

[illegible]

1.15 [illegible]

1.16 [illegible]

1.17 [illegible]

[illegible]

1. [illegible]

(*Note:* Olkin et al. 1981 were one of the first to investigate the stability of the MLE for the binomial parameter n; see also Carroll and Lombard 1985, Casella 1986, and Hall 1994.)

1.18 (Robertson et al. 1988) For a sample $X_1, \ldots, X_n$, and a function f on $\mathcal{X}$, the isotonic regression of f with weights ω_i is the solution of the minimization in g of

$$\sum_{i=1}^{n} \omega_i (g(x_i) - f(x_i))^2,$$

under the constraint $g(x_1) \leq \cdots \leq g(x_n)$.

(a) Show that a solution to this problem is obtained by the *pool-adjacent-violators* algorithm:

Algorithm A.1 –Pool-adjacent-violators–

```
If f is not isotonic, find i such that
f(x_{i-1}) > f(x_i), replace f(x_{i-1}) and f(x_i) by
```

$$f^*(x_i) = f^*(x_{i-1}) = \frac{\omega_i f(x_i) + \omega_{i-1} f(x_{i-1})}{\omega_i + \omega_{i-1}},$$

```
and repeat until the constraint is satisfied. Take g = f*.
```

(b) Apply this algorithm to the case $n = 4$, $f(x_1) = 23$, $f(x_2) = 27$, $f(x_3) = 25$, and $f(x_4) = 28$, when the weights are all equal.

1.19 (Continuation of Problem 1.18) The simple *tree ordering* is obtained when one compares treatment effects with a control state. The isotonic regression is then obtained under the constraint $g(x_i) \geq g(x_1)$ for $i = 2, \ldots, n$.

(a) Show that the following provides the isotonic regression g^*:

Algorithm A.2 –Tree ordering–

```
If f is not isotonic, assume w.l.o.g.
that the f(x_i)'s are in increasing order (i ≥ 2).
Find the smallest j such that
```

$$A_j = \frac{\omega_1 f(x_1) + \cdots + \omega_j f(x_j)}{\omega_1 + \cdots \omega_j} < f(x_{j+1}),$$

```
take g*(x_1) = A_j = g*(x_2) = ··· = g*(x_j), g*(x_{j+1}) = f(x_{j+1}), ...
```

(b) Apply this algorithm to the case where $n = 5$, $f(x_1) = 18$, $f(x_2) = 17$, $f(x_3) = 12$, $f(x_4) = 21$, $f(x_5) = 16$, with $\omega_1 = \omega_2 = \omega_5 = 1$ and $\omega_3 = \omega_4 = 3$.

1.20 For the setup of Example 1.8, where $X \sim \mathcal{N}_p(\theta, I_x)$:

(a) Show that the maximum likelihood estimator of $\lambda = \|\theta\|^2$ is $\hat{\lambda}(x) = \|x\|^2$ and that it has a constant bias equal to p.

(b) If we observe $Y = \|X\|^2$, distributed as a noncentral chi squared random variable (see Appendix A), show that the MLE of λ is the solution in (1.13). Discuss what happens if $y < p$.

(c) In part (a), if the reference prior $\pi(\theta) = \|\theta\|^{-(p-1)}$ is used, show that the posterior distribution is given by (1.18), with posterior mean (1.19).

1.21 Show that under the specification $\theta_j \sim \pi_j$ and $X \sim f_j(x|\theta_j)$ for model $\mathcal{M}_j$, the posterior probability of model $\mathcal{M}_j$ is given by (1.20).

1.22 Suppose that $X \sim f(x|\theta)$, with prior distribution $\pi(\theta)$, an interest is in the estimation of the parameter $h(\theta)$.

[illegible]

[illegible]

[illegible]

[illegible]

1.25 [illegible]

1.26 [illegible]

1.26 [illegible]

1.20 [illegible]

1.27 [illegible]

1.28 [illegible]

1.29 [illegible]

[illegible]

[illegible]

[illegible]

1.36 [illegible]

1.37 [illegible]

1.38 [illegible]

[illegible]

[illegible]

[illegible]

1.39 [illegible]

[illegible]

[illegible]

[illegible]

1.20 [illegible]

[illegible]

1.6 Notes

1.6.1 Prior Distributions

[illegible]

[illegible]

[illegible]

[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]

Table [illegible]. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

1.6.2 Sequential Methods

[illegible]

$$[illegible] = \sum_{[illegible]} [illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

2

Random Variable Generation

[illegible]

[illegible]

[illegible]

2.1 Introduction

[illegible]

[illegible]

2.1.1 Uniform Simulation

[illegible]

Definition 2.1 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 2.2. [illegible] **logistic function.** [illegible]

[illegible]

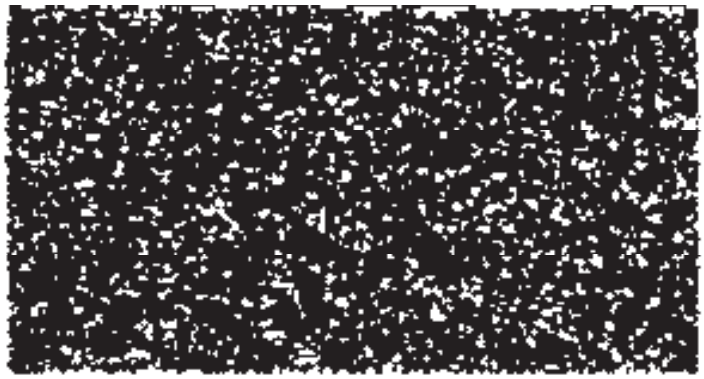


Fig. 2.1.

2.1.2 The Inverse Transform

[illegible]

Definition 2.3. [illegible]

$$[illegible]$$

[illegible]

Lemma 2.1. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

and

$$[illegible]$$

[illegible]

Example 2.5. Exponential variable generation. [illegible]

[illegible]

[illegible]

[illegible]

2.1.3 Alternatives

[illegible]

Example 2.6. Normal probabilities. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

2.1.4 Optimal Algorithms

[illegible]

2.2 General Transformation Methods

[illegible]

Example 2.7. Building an exponential random variable. [illegible]

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

[illegible]

[illegible]

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 [A.3]

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

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

[illegible]

[illegible]

Example 2.10. Discrete random variables. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 2.11. Bernoulli distribution. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Fig. 2.2. [illegible]

Example 2.12. Gaussian generation. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 2.13. Student's generation. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 2.14. Nonlinear chi-squared generation. [illegible]

[illegible]

2.3 Accept–Reject Methods

[illegible]

2.3.1 The Fundamental Theorem of Simulation

[illegible]

$$[illegible]$$

[illegible]

Theorem 2.15 (Fundamental Theorem of Simulation) [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

Example 2.16. Beta simulation. [illegible]

[illegible]

[illegible]

[illegible]

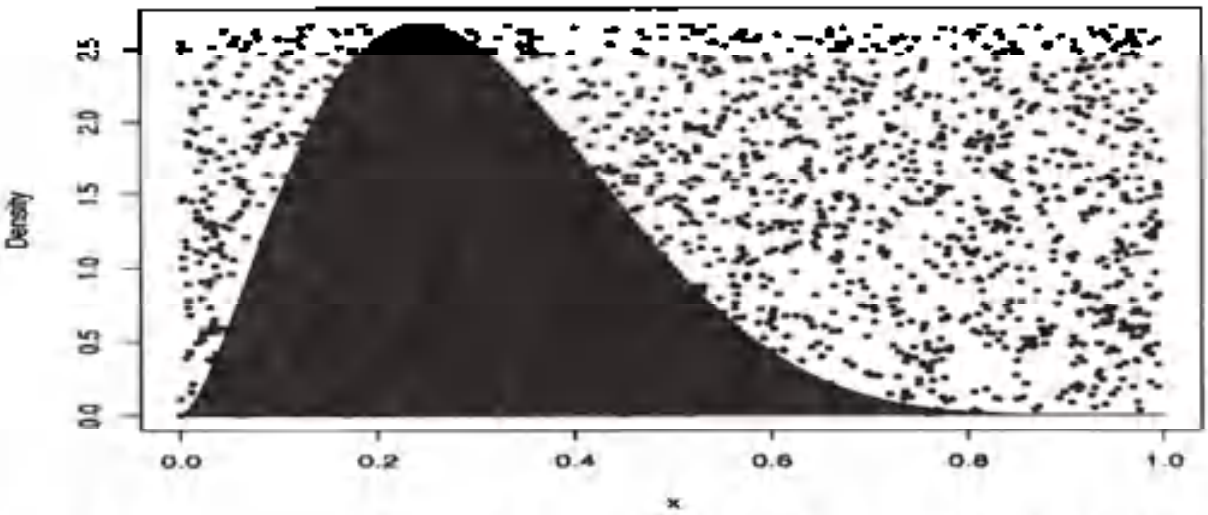


Fig. 2.3. Generation of Beta random variables: Using Theorem 2.15, 1000 (Y, U) pairs were generated, and 365 were accepted (the black circles under the Beta $Be(2.7, 6.3)$ density function).

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

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

the constraints are thus that $m(x) \geq f(x)$ and that simulation of a uniform on $\mathscr{L}$ 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 $m(x) \geq f(x)$, 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, $\mathscr{L}$ would not have finite mass and a uniform distribution would not exist on $\mathscr{L}$). As mentioned above, a natural way of simulating the uniform on $\mathscr{L}$ 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

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

for every measurable set 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:

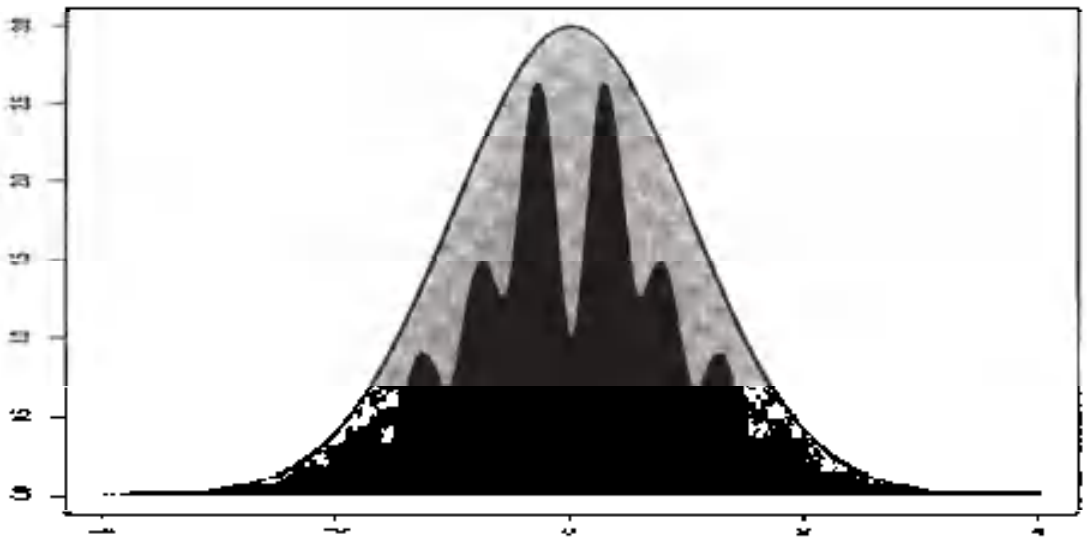


Fig. 2.4. Plot of a uniform sample over the set $\{(x,u) : 0 < u < f(x)\}$ for $f(x) \propto \exp(-x^2/2)(\sin(6x)^2 + 3\cos(x)^2\sin(4x)^2 + 1)$ and of the envelope function $g(x) = 5\exp(-x^2/2)$.

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 $g = f$

[illegible]

[illegible]

Example 2.18. **Normals from truncated exponentials.** [illegible]

$$[illegible]$$

[illegible]

[illegible]

Example 2.19. **Gamma: Accept–Reject.** [illegible]

$$[illegible]$$

[illegible]

Example 2.20. **Truncated normal distributions.** [illegible]

[illegible]

[illegible]

2.4 Envelope Accept–Reject Methods

2.4.1 The Squeeze Principle

[illegible]

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

where σ is known, each single evaluation of $\pi(\theta|\underline{x})$ 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*

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

then the algorithm

Algorithm A.5 –Envelope Accept–Reject–

```
1. Generate X ~ g_m(x),  U ~ U_[0,1];
2. Accept X if U ≤ g_l(X)/M g_m(X);                    [A.5]
3. otherwise, accept X if U ≤ f(X)/M g_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 $\exp(-x^2/2) \geq 1 - (x^2/2)$, and hence

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

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

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

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]

$$\alpha - \beta x + \log\left(u_2/\{1+\exp(\alpha-\beta x)\}^2\right) \leq k + N\log\lambda - \log N! \,.$$

[illegible]

2.1.2 Log-Concave Densities

[illegible]

Example 2.21. Log-concave densities. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

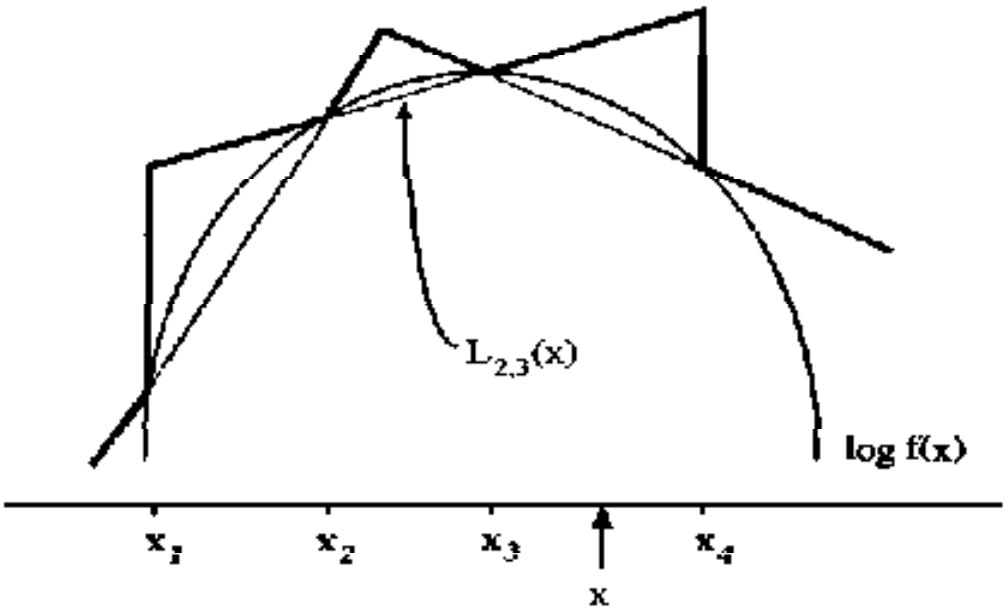


Fig. 2.5. Lower and upper envelopes of $h(x) = \log f(x)$, f a log-concave density (*Source:* Gilks et al. 1995).

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

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

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

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 $\mathcal{S}_n$ 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 $\mathcal{S}_n$, 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.)

[illegible]

[illegible]

Example 2.25. Capture-recapture models. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Fig. 2.6. [illegible]

[illegible]

[illegible]

Example 2.26. Poisson regression. [illegible]

[illegible]

[illegible]

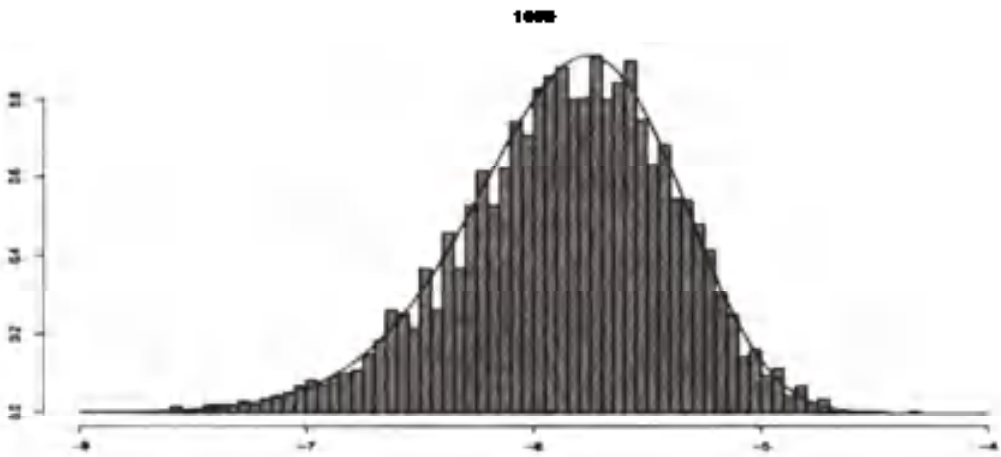


Fig. 2.7. Histogram of an ARS sample of 5000 points and corresponding posterior distribution of the log-odds ratio α_{1960}.

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

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 $\underline{f}_n(x)$, 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 19th century study by Bortkiewicz (1898) of deaths in the Prussian army due to horse kicks. The dat a 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 $[x_i, x_{i+1}]$.

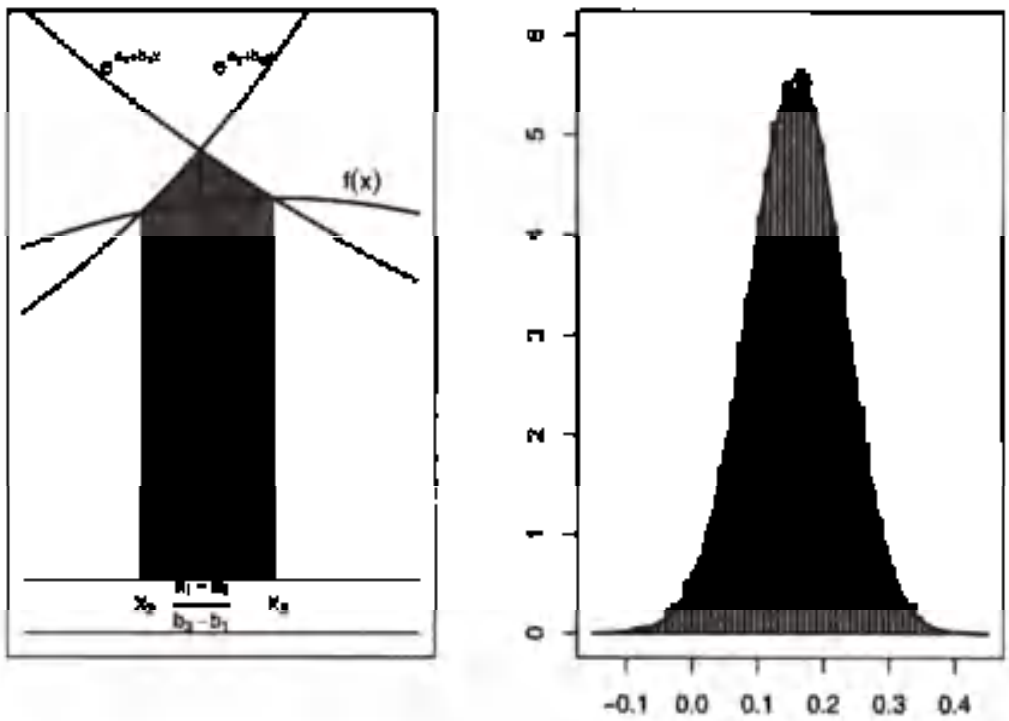


Fig. 2.8. Left panel is the area of integration for the weight of the interval $[x_2, x_3]$. The right panel is the histogram and density of the sample from g_n, with $b = .025$ and $\sigma^2 = 5$.

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

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

[illegible]

[illegible]

2.5 Problems

2.1 [illegible]

2.2 [illegible]

2.3 [illegible]

2.4 [illegible]

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 $S = U_1^2 + U_2^2 \le 1$.

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]:

Algorithm A.9 –Box–Muller (3)–

1. **Generate**

$$Y_1, Y_2 \sim \mathcal{E}xp(1)$$

until $Y_2 > (1 - Y_1)^2/2$. [A.9]

2. **Generate** $U \sim \mathcal{U}([0,1])$ **and take**

$$X = \begin{cases} Y_1 & \text{if } U < 0.5 \\ -Y_1 & \text{if } U > 0.5. \end{cases}$$

2.10 *Examine the properties of an algorithm to simulate $\mathcal{N}(0,1)$ random variables* based on the Central Limit Theorem, which takes the appropriately adjusted mean of a sequence of uniforms $U_1, \ldots, U_n$ for $n = 12$, $n = 48$ and $n = 96$. Consider, in particular, the moments, ranges, and tail probability calculations based on the generated variables.

2.11 For the generation of a Cauchy random variable, compare the inversion method with one based on the generation of the normal pair of the polar Box–Muller method (Problem 2.8).

(a) Show that, if X_1 and X_2 are iid normal, $Y = X_1/X_2$ is distributed as a Cauchy random variable.

(b) Show that the Cauchy distribution function is $F(x) = \tan^{-1}(x)/\pi$, so the inversion method is easily implemented.

(c) Is one of the two algorithms superior?

2.12 *Use the algorithm of Example 2.10 to generate the following random variables.* In each case make a histogram and compare it to the mass function, and to the generator in your computer.

(a) Binomials and Poisson distributions;

(b) The hypergeometric distribution;

(c) The *logarithmic series* distribution. A random variable X has a logarithmic series distribution with parameter p if

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

(d) Referring to part (a), for different parameter values, compare the algorithms there with those of Problems 2.13 and 2.16.

2.13 Referring to Example 2.9.

(a) Show that if $N \sim \mathcal{P}(\lambda)$ and $X_i \sim \mathcal{E}xp(\lambda)$, $i \in \mathbb{N}^*$, independent, then

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

(b) Use the results of part (a) to justify that the following algorithm simulates a Poisson $\mathcal{P}(\lambda)$ random variable:

Algorithm A.10 –Poisson simulation–

$p = 1$, $N = 0$, $c = e^{-\lambda}$.

1. **Repeat** [A.10]

   $N = N + 1$

   generate U_i

   update $p = pU_i$

   until $p < c$.

2. **Take** $X = N - 1$.

(*Hint:* For part (a), integrate the Gamma density by parts.)

2.14 There are (at least) two ways to establish (2.4) of Example 2.12:

(a) Make the transformation $x = yw^{1/\alpha}$, $w = y$, and integrate out w.

(b) Make the transformation $x = yz$, $w = z$, and integrate out w.

2.15 In connection with Example 2.16, for a Beta distribution $\mathcal{B}e(\alpha,\beta)$, find the maximum of the $\mathcal{B}e(\alpha,\beta)$ density.

2.16 Establish the validity of Knuth (1981) $\mathcal{B}(n,p)$ generator:

Algorithm A.11 –Binomial–

Define $k = n$, $\theta = p$ and $x = 0$.
1. Repeat $i = \lceil 1 + k\theta \rceil$
   $V \sim \mathcal{B}e(i, k+1-i)$ [A.11]
   if $\theta > V$, $\theta = \theta/V$ and $k = i - 1$;
   otherwise, $x = x + i$, $\theta = (\theta - V)/(1 - V)$ and $k = k - i$
   until $k \leq K$.
2. For $i = 1, 2, \ldots, k$,
   generate U_i
   if $U_i < p$, $x = x + 1$.
3. Take x.

2.17 Establish the claims of Example 2.11: If $U_1, \ldots, U_n$ is an iid sample from $\mathcal{U}_{[0,1]}$ and $U_{(1)} \leq \cdots \leq U_{(n)}$ are the corresponding order statistics, show that

(a) $U_{(i)} \sim \mathcal{B}e(i, n-i+1)$;

(b) $(U_{(i_1)}, U_{(i_2)} - U_{(i_1)}, \ldots, U_{(i_k)} - U_{(i_{k-1})}, 1 - U_{(i_k)}) \sim \mathcal{D}(i_1, i_2 - i_1, \ldots, n - i_k + 1)$;

(c) 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 $U^{1/\alpha} + V^{1/\beta} \leq 1$, is the $\mathcal{B}e(\alpha,\beta)$ distribution.

(d) Show that the order statistics can be directly generated via the *Renyi representation* $u_{(i)} = \sum_{j=1}^{i} \nu_j / \sum_{j=1}^{n} \nu_j$, where the ν_j's are iid $\mathcal{E}xp(1)$.

2.18 For the generation of a Student's t distribution, $\mathcal{T}(\nu, 0, 1)$, Kinderman et al. (1977) provide an alternative to the generation of a normal random variable and a chi squared random variable.

Algorithm A.12 –Student's t–

1. Generate $U_1, U_2 \sim \mathcal{U}([0,1])$.
2. If $U_1 < 0.5$, $X = 1/(4U_1 - 1)$ and $V = X^{-2}U_2$;
   otherwise, $X = 4U_1 - 3$ and $V = U_2$. [A.12]
3. If $V < 1 - (|X|/2)$ or $V < (1 + (X^2/\nu))^{-(\nu+1)/2}$, take X;
   otherwise, repeat.

Validate this algorithm and compare it with the algorithm of Example 2.13.

2.19 For $\alpha \in [0,1]$, show that the algorithm

Algorithm A.13

Generate
$$U \sim \mathcal{U}([0,1]) \qquad [A.13]$$
until $U < \alpha$.

produces a simulation from $\mathcal{U}([0,\alpha])$. Compare it with the transform αU, $U \sim \mathcal{U}(0,1)$ for values of α close to 0 and close to 1.

2.20 In each case generate the random variables and compare the histogram to the density function

(a) Normal random variables using a Cauchy candidate in Accept–Reject

(b) Gamma(4.3, 6.2) random variables using a Gamma(4, 7)

(c) Truncated normal: Standard normal truncated to $(2,\infty)$

2.21 *An efficient algorithm for the simulation of Gamma $\mathcal{G}a(\alpha,1)$ distributions is* based on *Burr's distribution*, a distribution with density

$$g(x) = \lambda\mu \frac{x^{\lambda-1}}{(\mu+x^{\lambda})^2}, \qquad x > 0 .$$

It has been developed by Cheng (1977) and Cheng and Feast (1979). (See Devroye 1985.) For $\alpha > 1$, it is

Algorithm A.14 –Cheng and Feast's Gamma–

```
Define c1 = α−1, c2 = (α−(1/6α))/c1, c3 = 2/c1, c4 = 1+c3,
and c5 = 1/√α.
1. Repeat
   generate U1, U2
   take U1 = U2 + c5(1 − 1.86U1) if α > 2.5
   until 0 < U1 < 1.                                  [A.14]
2. W = c2U2/U1.
3. If c3U1 + W + W^−1 ≤ c4 or c3 log U1 − log W + W ≤ 1,
   take c1W;
   otherwise, repeat.
```

(a) Show g is a density.

(b) Show that this algorithm produces variables generated from $\mathcal{G}a(\alpha,1)$

2.22 Ahrens and Dieter (1974) propose the following algorithm to generate a Gamma $\mathcal{G}a(\alpha,1)$ distribution:

Algorithm A.15 –Ahrens and Dieter's Gamma–

```
1. Generate U0, U1.
2. If U0 > e/(e+α), x = −log((α+e)(1−U0)/αe) and y = x^(α−1);
   otherwise, x = {(α+e)U0/e}^(1/α) and y = e^−x.     [A.15]
3. If U1 < y, take x;
   otherwise, repeat.
```

Show that this algorithm produces variables generated from $\mathcal{G}a(\alpha,1)$. Compare with Problem 2.21.

2.23 To generate the Beta distribution $\mathcal{B}e(\alpha,\beta)$ we can use the following representation:

(a) Show that, if $Y_1 \sim \mathcal{G}a(\alpha,1)$, $Y_2 \sim \mathcal{G}a(\beta,1)$, then

$$X = \frac{Y_1}{Y_1+Y_2} \sim \mathcal{B}e(\alpha,\beta) .$$

(b) Use part (a) to construct an algorithm to generate a Beta random variable.

(c) Compare this algorithm with the method given in Problem 2.17 for different values of (α, β).

(d) Compare this algorithm with an Accept–Reject algorithm based on (i) the uniform distribution; (ii) the truncated normal distribution (when $\alpha \geq 1$ and $\beta \geq 1$).

(*Note:* See Schmeiser and Shalaby 1980 for an alternative Accept–Reject algorithm to generate Beta rv's.)

2.24 (a) Show that Student's t density can be written in the form (2.6), where $h_1(x|y)$ is the density of $\mathcal{N}(0, \nu/y)$ and $h_2(y)$ is the density of χ^2_ν.

(b) Show that Fisher's $\mathcal{F}_{m,\nu}$ density can be written in the form (2.6), with $h_1(x|y)$ the density of $\mathcal{G}a(m/2, \nu/m)$ and $h_2(y)$ the density of χ^2_ν.

2.25 The noncentral chi squared distribution, $\chi^2_p(\lambda)$, can be defined by a mixture representation (2.6), where $h_1(x|K)$ is the density of χ^2_{p+2K} and $h_2(k)$ is the density of $\mathcal{P}(\lambda/2)$.

(a) Show that it can also be expressed as the sum of a χ^2_{p-1} random variable and of the square of a standard normal variable.

(b) Compare the two algorithms which can be derived from these representations.

(c) Discuss whether a direct approach via an Accept–Reject algorithm is at all feasible.

2.26 (Walker 1997) Show that the Weibull distribution, $\mathcal{W}e(\alpha, \beta)$, with density

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

can be represented as a mixture of $X \sim \mathcal{B}e(\alpha, \omega^{1/\alpha})$ by $\omega \sim \mathcal{G}a(2, \beta)$. Examine whether this representation is helpful from a simulation point of view.

2.27 An application of the mixture representation can be used to establish the following result (see Note 2.6.3):

Lemma 2.27. *If*

$$f(x) = \frac{f_1(x) - \varepsilon f_2(x)}{1 - \varepsilon},$$

where f_1 and f_2 are probability densities such that $f_1(x) \geq \varepsilon f_2(x)$, the algorithm

Generate

$$(X, U) \sim f_1(x)\, \mathbf{I}_{[0,1]}(u)$$

until $U \geq \varepsilon f_2(X)/f_1(X)$,

produces a variable X distributed according to f.

(a) Show that the distribution of X satisfies

$$P(X \leq x_0) = \int_{-\infty}^{x_0} \left(1 - \frac{\varepsilon f_2(x)}{f_1(x)}\right) f_1(x)\, dx \sum_{i=0}^{\infty} \varepsilon^i.$$

(b) Evaluate the integral in (a) to complete the proof.

2.28 (a) Demonstrate the equivalence of Corollary 2.17 and the Accept–Reject algorithm.

[illegible]

2.29 [illegible]

$$[illegible]$$

[illegible]

2.30 [illegible]

$$[illegible]$$

[illegible]

2.31 [illegible]

2.32 [illegible]

$$[illegible]$$

[illegible]

2.33 The right-truncated Gamma distribution $\mathcal{TG}(a,b,t)$ is defined as the restriction of the Gamma distribution $\mathcal{G}a(a,b)$ to the interval $(0,t)$.

(a) Show that we can consider $t=1$ without loss of generality.

(b) Give the density f of $\mathcal{TG}(a,b,1)$ and show that it can be expressed as the following mixture of Beta $\mathcal{B}e(a,k+1)$ densities:

$$f(x) = \sum_{k=0}^{\infty} \frac{b^a e^{-b}}{\gamma(a,b)} \frac{b^k}{k!} x^{a-1}(1-x)^k,$$

where $\gamma(a,b) = \int_0^1 x^{a-1}e^{-x}$.

(c) If f is replaced with g_n, which is the series truncated at term $k=n$, show that the acceptance probability of the Accept–Reject algorithm based on (g_n, f) is

$$\frac{1 - \dfrac{\gamma(n+1,b)}{n!}}{1 - \dfrac{\gamma(a+n+1,b)\Gamma(a)}{\Gamma(a+n+1)\gamma(a,b)}}.$$

(d) Evaluate this probability for different values of (a,b).

(e) Give an Accept–Reject algorithm based on the pair (g_n, f) and a computable bound. (*Note:* See Philippe 1997c for a complete resolution of the problem.)

2.34 Let $f(x) = \exp(-x^2/2)$ and $g(x) = 1/(1+x^2)$, densities of the normal and Cauchy distributions, respectively (ignoring the normalization constants).

(a) Show that the ratio

$$\frac{f(x)}{g(x)} = (1+x^2)\, e^{-x^2/2} \le 2/\sqrt{e},$$

which is attained at $x = \pm 1$.

(b) Show that for the normalized densities, the probability of acceptance is $\sqrt{e/2\pi} = 0.66$., which implies that, on the average, one out of every three simulated Cauchy variables is rejected. Show that the mean number of trials to success is $1/.66 = 1.52$.

(c) Replacing g by a Cauchy density with scale parameter σ,

$$g_\sigma(x) = 1/\{\pi\sigma(1 + x^2/\sigma^2)\},$$

show that the bound on f/g_σ is $2\sigma^{-1}\exp\{\sigma^2/2 - 1\}$ and is minimized by $\sigma^2 = 1$. (This shows that $\mathcal{C}(0,1)$ is the best choice among the Cauchy distributions for simulating a $\mathcal{N}(0,1)$ distribution.)

2.35 There is a direct generalization of Corollary 2.17 that allows the proposal density to change at each iteration.

Algorithm A.16 –Generalized Accept–Reject–

At iteration i ($i \ge 1$)
1. Generate $X_i \sim g_i$ and $U_i \sim \mathcal{U}([0,1])$, independently.
2. If $U_i \le c_i f(X_i)/g_i(X_i)$, accept $X_i \sim f$;
3. otherwise, move to iteration $i+1$.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

2.36 [illegible]

[illegible]

[illegible]

[illegible]

2.37 [illegible]

2.38 [illegible]

2.39 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Algorithm A.17 –Supplemental ARS Algorithm–

1. Select the interval $[x_i, x_{i+1}]$ with probability

$$e^{\beta_i} \frac{e^{\alpha_i x_{i+1}} - e^{\alpha_i x_i}}{\varpi_n \alpha_i} \qquad [A.17]$$

2. Generate $U \sim \mathcal{U}_{[0,1]}$ and take

$$X = \alpha_i^{-1} \log[e^{\alpha_i x_i} + U(e^{\alpha_i x_{i+1}} - e^{\alpha_i x_i})].$$

Note that the segment $\alpha_i + \beta_i x$ is not the same as the line $a_i + b_i x$ used in (2.16)

2.40 As mentioned in Section 2.4, many densities are log-concave.

(a) Show that the so-called *natural* exponential family,

$$dP_\theta(x) = \exp\{x \cdot \theta - \psi(\theta)\} d\nu(x)$$

is log-concave.

(b) Show that the logistic distribution of (2.23) is log-concave.

(c) Show that the *Gumbel* distribution

$$f(x) = \frac{k^k}{(k-1)!} \exp\left\{-kx - ke^{-x}\right\}, \quad k \in \mathbb{N}^*,$$

is log-concave (Gumbel 1958).

(d) Show that the *generalized inverse Gaussian* distribution,

$$f(x) \propto x^\alpha e^{-\beta x - \alpha/x}, \qquad x > 0,\ \alpha > 0,\ \beta > 0,$$

is log-concave.

2.41 (George et al. 1993) For the natural exponential family, the conjugate prior measure is defined as

$$d\pi(\theta|x_0, n_0) \propto \exp\{x_0 \cdot \theta - n_0\psi(\theta)\} d\theta,$$

with $n_0 > 0$. (See Brown 1986, Chapter 1, for properties of exponential families.)

(a) Show that

$$\varphi(x_0, n_0) = \log \int_\Theta \exp\{x_0 \cdot \theta - n_0\psi(\theta)\} d\theta$$

is convex.

(b) Show that the so-called *conjugate likelihood distribution*

$$L(x_0, n_0|\theta_1, \ldots, \theta_p) \propto \exp\left\{x_0 \cdot \sum_{i=1}^p \theta_i - n_0 \sum_{i=1}^p \psi(\theta) - p\varphi(x_0, n_0)\right\}$$

is log-concave in (x_0, n_0).

(c) Deduce that the ARS algorithm applies in hierarchical Bayesian models with conjugate priors on the natural parameters and log-concave hyperpriors on (x_0, n_0).

(d) Apply the ARS algorithm to the case

$$X_i|\theta_i \sim \mathcal{P}(\theta_i t_i), \qquad \theta_i \sim \mathcal{G}a(\alpha, \beta), \qquad i = 1, \ldots, n,$$

with fixed α and $\beta \sim \mathcal{G}a(0.1, 1)$.

[illegible]

2.2 [illegible]

2.3 [illegible]

2.4 [illegible]

2.6 Notes

2.6.1 The Kiss Generator

[illegible]

Definition 2.28 [illegible]

[illegible]

Definition 2.29 [illegible]

[illegible]

Definition 2.30. [illegible]

[illegible]

[illegible]

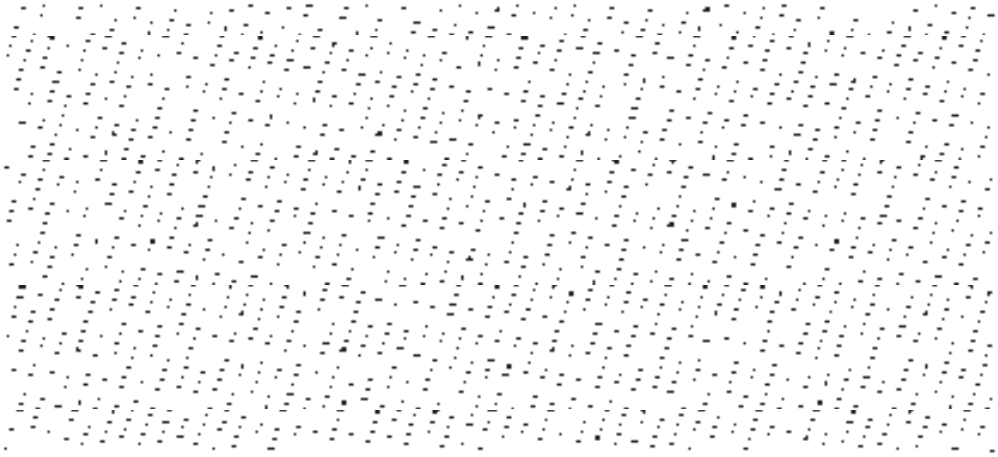


Fig. 2.9 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

To generate a sequence of integers $X_1, X_2, \ldots$, the Kiss algorithm generates three sequences of integers. First, the algorithm uses a congruential generator to obtain

$$I_{n+1} = (69069 \times I_n + 23606797) \pmod{2^{32}} ,$$

and then two shift register generators of the form,

$$J_{n+1} = (I + L^{15})(I + R^{17})\, J_n \pmod{2^{32}} ,$$
$$K_{n+1} = (I + L^{13})(I + R^{18})\, K_n \pmod{2^{31}} .$$

These are then combined to produce

$$X_{n+1} = (I_{n+1} + J_{n+1} + K_{n+1}) \mod 2^{32} .$$

Formally, this algorithm is not of the type specified in Definition 2.1, since it uses three parallel chains of integers. However, this feature yields advantages over algorithms based on a single dynamic system $X_{n+1} = f(X_n)$ since the period of Kiss is of order 2^{95}, which is almost $(2^{32})^3$. In fact, the (usual) congruential generator I_n has a maximal period of 2^{32}, the generator of K_n has a period of $2^{31} - 1$ and that of J_n a period of $2^{32} - 2^{21} - 2^{11} + 1$ for almost all initial values J_0 (see Marsaglia and Zaman 1993 for more details). The Kiss generator has been successfully tested on the different criteria of Die Hard, including tests on random subsets of *bits*. Figure 2.10 presents plots of (X_n, X_{n+1}), (X_n, X_{n+2}), (X_n, X_{n+5}) and (X_n, X_{n+10}) for $n = 1, \ldots, 5000$, where the sequence (X_n) has been generated by Kiss, without exhibiting any nonuniform feature on the square $[0,1]^2$. A version of this algorithm in the programming language C is given below.

Algorithm A.18 –The Kiss Algorithm–

```
long int kiss (i,j,k)
unsigned long *i,*j,*k
{
*j = *j ∧ (*j << 17);                                  [A.18]
*k = (*k ∧ (*k << 18)) & 0X7FFFFFFF ;
return  ((*i = 69069 * (*i) + 23606797) +
(*j ∧ = (*j >> 15)) + (*k ∧ = (*k >> 13)) ;
}
```

(See Marsaglia and Zaman 1993 for a Fortran version of Kiss). Note that some care must be exercised in the use of this program as a generator on $[0,1]$, since it implies *dividing by the largest integer available on the computer and may sometimes result* in uniform generation on $[-1,1]$.

2.6.2 Quasi-Monte Carlo Methods

Quasi-Monte Carlo methods were proposed in the 1950s to overcome some drawbacks of regular Monte Carlo methods by replacing probabilistic bounds on the errors with deterministic bounds. The idea at the core of quasi-Monte Carlo methods is to substitute the randomly (or pseudo-randomly) generated (uniform $[0,1]$) sequences

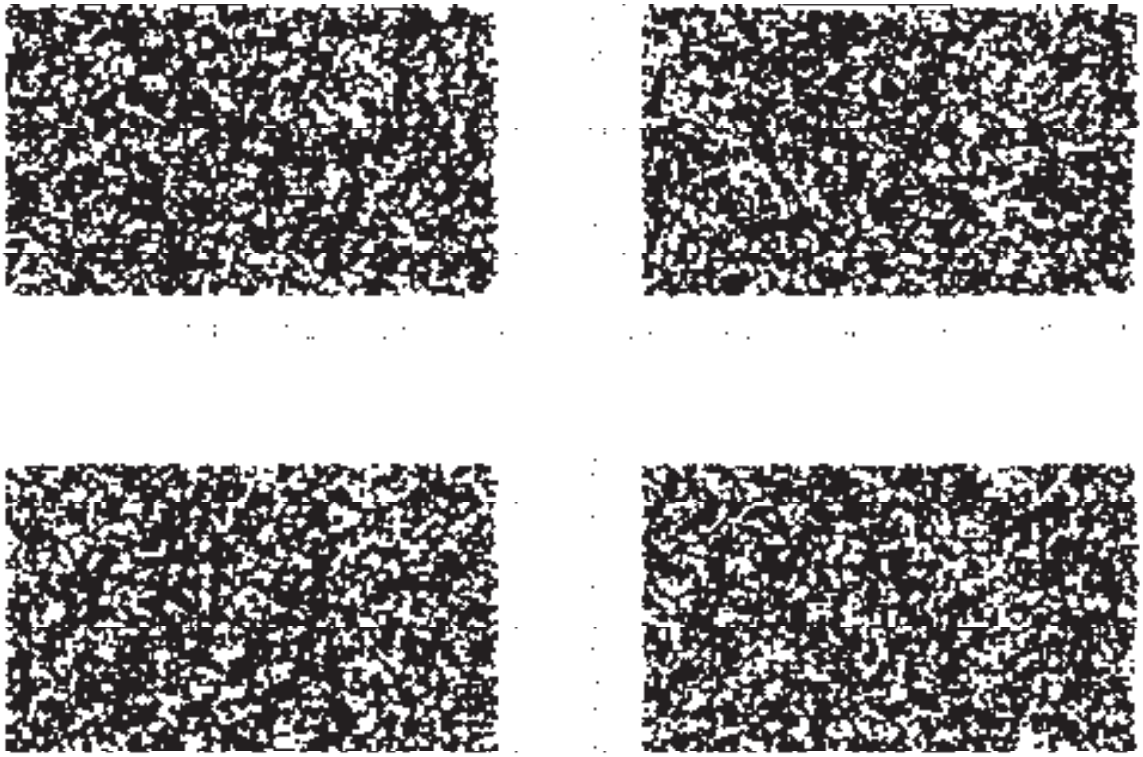


Fig. 2.10 [illegible]

[illegible]

$$[illegible] \frac{[illegible]}{[illegible]}$$

[illegible]

$$\sum [illegible]$$

[illegible]

[illegible]

[illegible]

2.8.3 Mixture Representations

[illegible]

[illegible]

[illegible]

3

Monte Carlo Integration

3.1 Introduction

[illegible]

[illegible]

[illegible]

Example 3.1. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 3.2. Piecewise linear and quadratic loss functions. [illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 3.3. Empirical Bayes estimator. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

3.2 Classical Monte Carlo Integration

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

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.) ‖

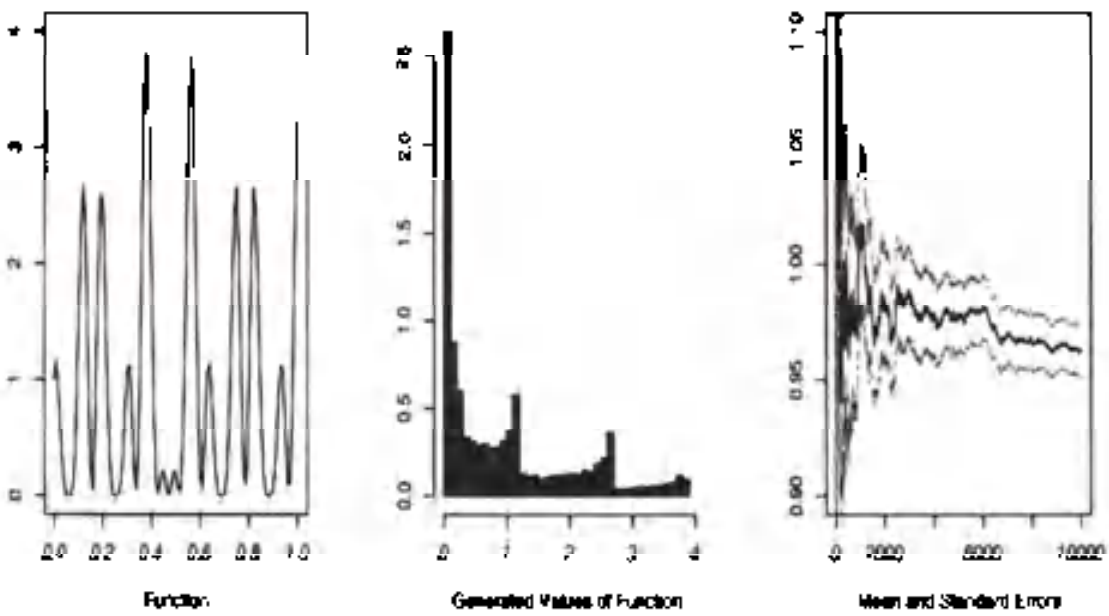


Fig. 3.1. Calculation of the integral of the function (1.26): *(left)* function (1.26), *(center)* histogram of 10,000 values $h(U_i)$, simulated using a uniform generation, and *(right)* mean ± one standard error.

[illegible]

[illegible]

Lemma 3.1. [illegible]

[illegible]

Example 3.5. Normal coff[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 3.6. (Contingency Tables) [illegible]

[illegible]			
[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]

Table 3.2. [illegible]

[illegible]

[illegible]

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

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

and calculate $\lambda(x)$. 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. ‖

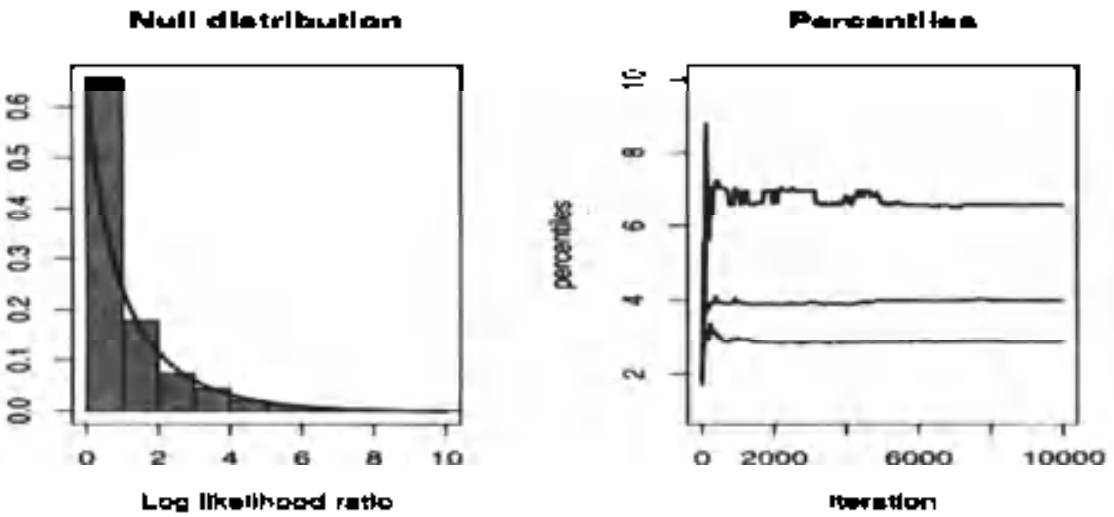


Fig. 3.2. For Example 3.6, histogram of null distribution and approximating χ^2_1 density(left panel). The right panel gives the running empirical percentiles (.90, .95, .99), from bottom to top. Notice the higher variability in the higher percentiles (10,000 simulations).

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

[illegible]

[illegible]

[illegible]

Fig. 3.3. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

3.3 Importance Sampling

3.3.1 Principles

[illegible]

Example 3.8. Cauchy tail probability. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Definition 3.9. [illegible]

$$[illegible] \sum [illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 3.10. Exponential and logarithmic comparison. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Fig. 3.1. [illegible]

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

We [illegible]

Example 3.1. Small [illegible] probabilities [illegible]

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

3.3.2 Finite Variance Estimators

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 9.12. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

3.12 [illegible]

[illegible]

Example 3.13. Student's t distribution. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

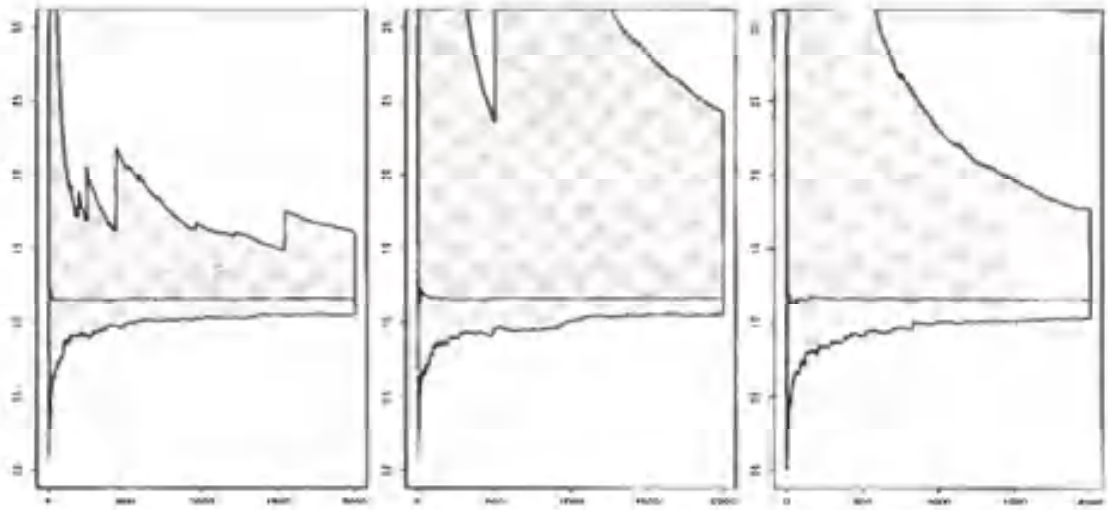


Fig. 3.5. *Empirical range of three series of estimators of* $\mathbb{E}_f[|X/(1-X)|^{1/2}]$ *for* $\nu = 12$ and 500 replications: sampling from f *(left)*, importance sampling with a Cauchy instrumental distribution *(center)* and importance sampling with normal importance distribution *(right)*. Average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

other hand, the $\mathcal{C}(0,1)$ distribution has larger tails than f and ensures that the variance of f/g is finite.

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

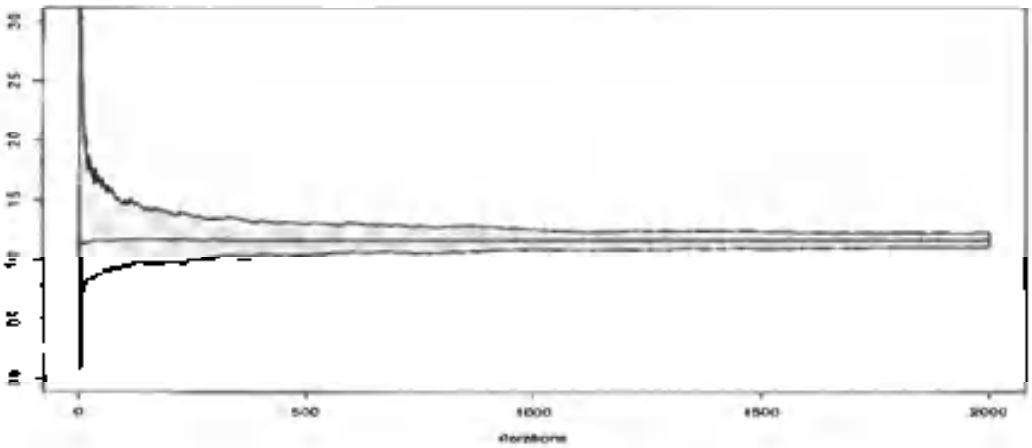


Fig. 3.6. Empirical range of the importance sampling estimator of $\mathbb{E}_f[|X/(1-X)|^{1/2}]$ for $\nu = 12$ and 500 replications based on the double Gamma $\mathcal{G}a(\alpha, 1)$ distribution folded at 1 when $\alpha = .5$. Average of the 500 series in overlay.

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

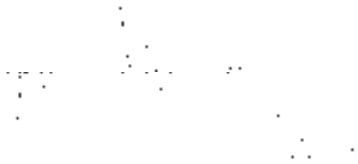


Fig. 3.7. [illegible]

3.11 [illegible]

[illegible]

Example 3.14. Transition matrix estimation. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

3.3.3 Comparing Importance Sampling with Accept–Reject

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible] = \sum [illegible] \frac{[illegible]}{[illegible]}$$

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible] = [illegible] \sum \frac{[illegible]}{[illegible]}$$

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

Example 3.15. Gamma simulation. [illegible]

[illegible]

[illegible]

[

Fig. 3 [illegible]

3.1 Laplace Approximations

[illegible]

[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]

Table [illegible] [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

3.3 [illegible]

3.4 [illegible]

3.5 [illegible]

3.6 [illegible]

3.7 [illegible]

3.8 [illegible]

[illegible]

[illegible]

[illegible]

3.31 [illegible]

3.32 [illegible]

3.33 [illegible]

3.34 [illegible]

3.35 [illegible]

3.36 [illegible]

3.37 [illegible]

3.38 [illegible]

3.39 [illegible]

3.40 [illegible]

3.41 [illegible]

3.42 [illegible]

3.43 [illegible]

3.44 [illegible]

3.45 [illegible]

3.46 [illegible]

3.47 [illegible]

[illegible]

[illegible]

3.17 [illegible]

[illegible]

3.18 [illegible]

[illegible]

3.19 [illegible]

[illegible]

3.20 [illegible]

[illegible]

3.20 [illegible]

[illegible]

3.22 [illegible]

[illegible]

3.23 [illegible]

[illegible]

3.24 [illegible]

[illegible]

3.25 [illegible]

3.26 [illegible]

3.27 [illegible]

3.28 [illegible]

3.29 [illegible]

[illegible]

[illegible]

[illegible]

3.32 [illegible]

3.33 [illegible]

3.34 [illegible]

[illegible]

[illegible]

3.35 [illegible]

[illegible]

3.36 [illegible]

3.37 [illegible]

[illegible]

[illegible]

3.38 [illegible]

[illegible]

3.6 Notes

3.6.1 Large Deviations Techniques

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 3.3.1. Laplace distribution [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

3.3.2 The Saddle-point Approximation

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Table 8.7. [illegible]

[illegible]

4

Controling Monte Carlo Variance

[illegible]

[illegible]

4.1 Monitoring Variation with the CLT

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

4.1.4 Univariate Measurements

[illegible]

Example 4.1. Monitoring with the CEL. [illegible]

[illegible]

Example 4.2. Cauchy priors. [illegible]

[illegible]

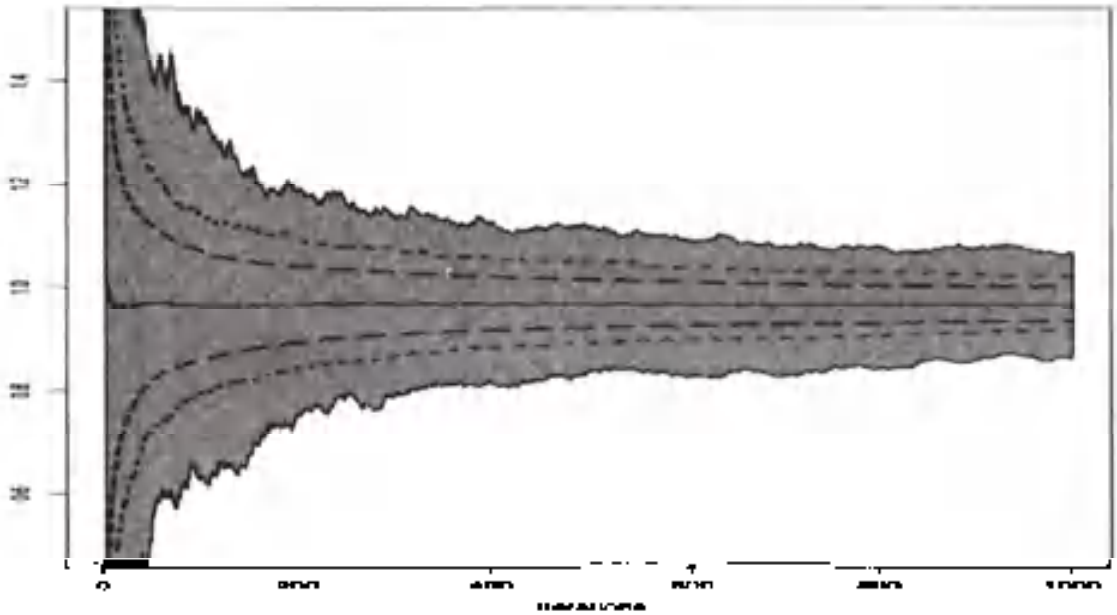


Fig. 4.1. Convergence of 1000 parallel sequences of Monte Carlo estimators of the integral of $h(x) = [\cos(50x) + \sin(20x)]^2$: The straight line is the running average of 1000 estimates, the dotted line is the empirical 90% band and the dashed line is the normal approximation 95% confidence interval. 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 $\theta_1, \ldots, \theta_m \sim \mathcal{N}(x, 1)$ 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

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Lemma 1.6. [illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 1.8 (Continuation of Example 1.2)**.** [illegible]

[illegible]

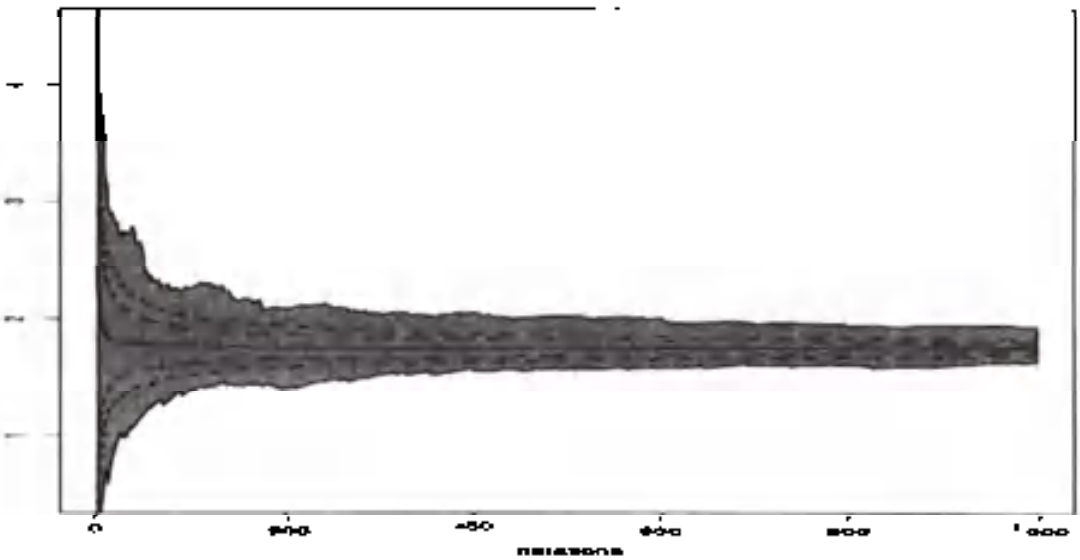


Fig. 4.2. Convergence of 1000 parallel sequences of Monte Carlo estimators for the posterior mean in the Cauchy–Normal problem when $x = 2.5$. The straight line is the running average of 1000 estimates, the dotted line is the empirical 90% band and the dashed line is the normal approximation 95% confidence 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]

$$\delta_h^n = \sum_{i=1}^n \theta_i \exp\left\{-(x-\theta_i)^2/2\right\} \Big/ \sum_{i=1}^n \exp\left\{-(x-\theta_i)^2/2\right\} .$$

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.

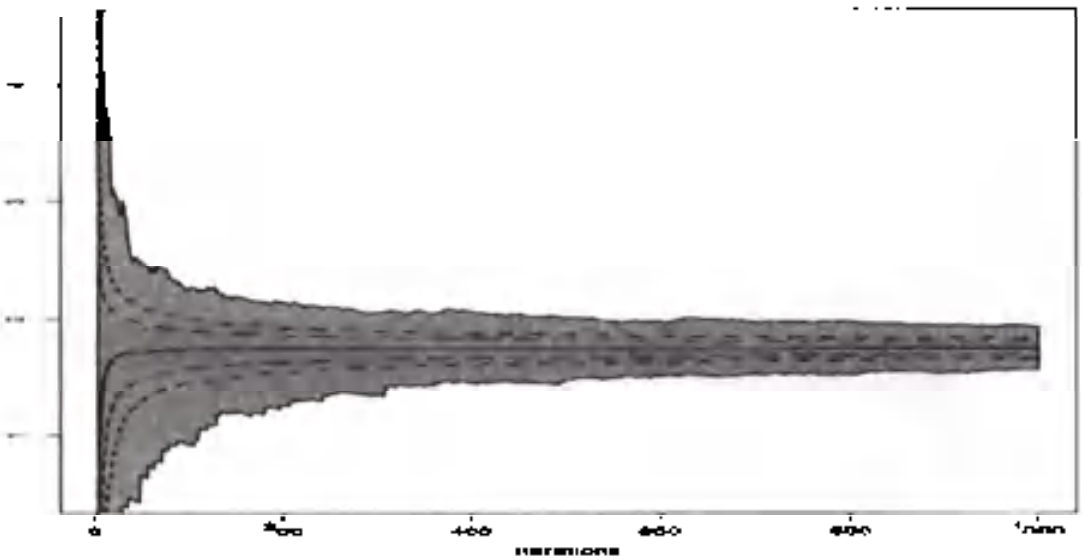


Fig. 4.3. Same plot as Figure 4.2 when the θ_i's are simulated from the prior $C(0,1)$ distribution.

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 $\mu = \mathbf{E}_f(X_1)$. (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 $\mathbf{E}(\bar{\mathbf{X}}) = \mathbf{1}\mu$. Moreover, it is straightforward to calculate

$$\mathrm{cov}(\bar{X}_k, \bar{X}_{k'}) = \frac{\sigma^2}{\max\{k, k'\}}.$$

It then follows that $\bar{\mathbf{X}} \sim \mathcal{N}_n(\mathbf{1}\mu, \Sigma)$, where

[illegible]

[illegible]

[illegible]

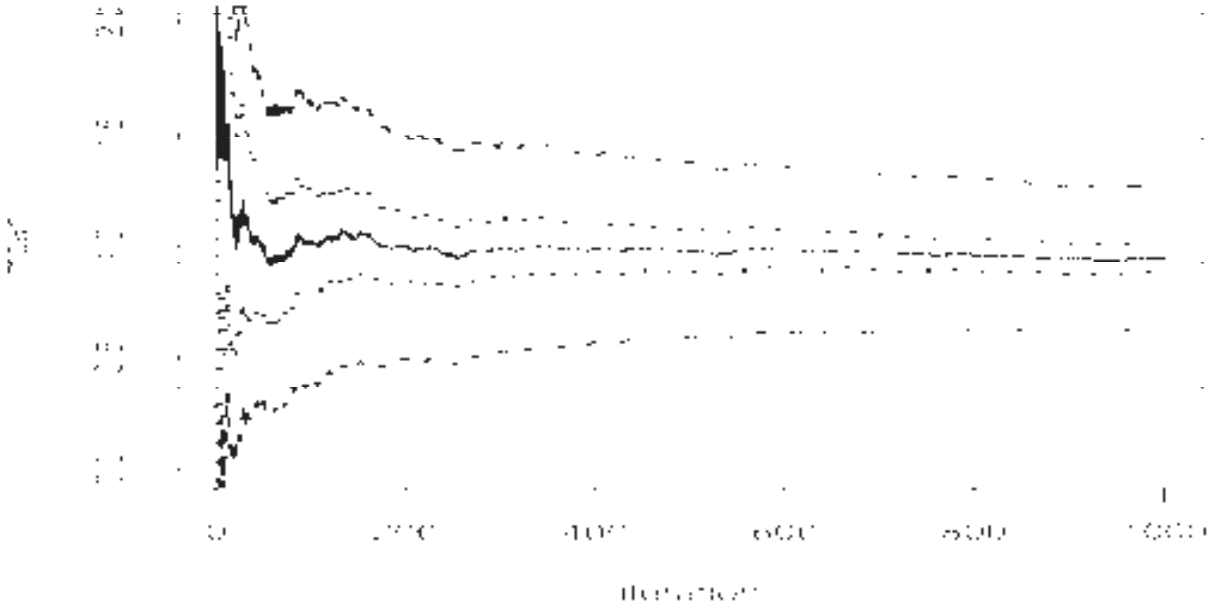


Fig. 1.

1.2 Rao-Blackwellization

[illegible]

[illegible]

$$[illegible]$$

[illegible]

Example 4.5. Stopping's exponentiation. [illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Fig. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Fig. 1.6. [illegible]

[illegible]

[illegible]

1.3 Riemann Approximations

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

Definition 1.8. [illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

Proposition 1.9. [illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 1.10. (Continuation of Example 1.15). [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Fig. 3.7. [illegible]

Example 3.14. (Continuation of Example 3.13): [illegible]

[illegible]

[illegible]

[illegible]

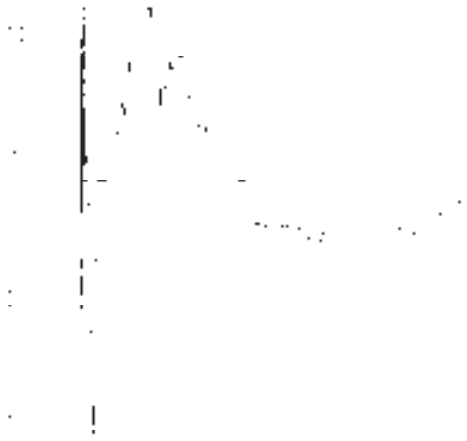


Fig. 1.8.

Fig. 1.9.

[illegible]

1.4 Acceleration Methods

[illegible]

[illegible]

1.4.1 Anderson Variations

[illegible]

$$[illegible] = \int [illegible] \qquad [illegible] = \int [illegible]$$

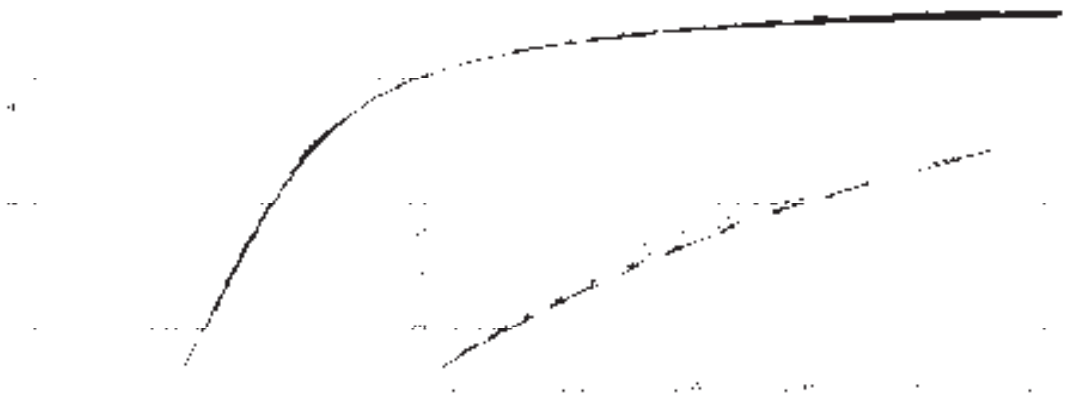
[illegible]

[illegible]

Example 4.12. James–Stein estimation. [illegible]

$$[illegible]$$

[illegible]



[illegible]

Fig. 4.10. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

4.4.2 General Variates

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 4.44. *Change variable integration.* [illegible]

$$[illegible]$$

[illegible]

Suppose [illegible]

[illegible]

$$[illegible] \sum_{[illegible]} [illegible] \left\{ -\sum_{[illegible]} [illegible] \right\}$$

[illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 4.15. Logistic regression. [illegible]

$$[illegible]$$

[illegible]

$$[illegible] \sum [illegible] \Big] \quad [illegible]$$

[illegible]

$$[illegible] \Big] \quad [illegible]$$

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

4.5 Problems

4.1 [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

1.2 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

1.10 [illegible]

[illegible]

1.11 [illegible]

[illegible]

[illegible]

1.12 [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

1.13 [illegible]

1.14 [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

1.21 [illegible]

$$[illegible]$$

[illegible]

1.22 [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

1.6 Notes

1.6.1 Monitoring Importance Sampling Convergence

[illegible]

[illegible]

[illegible]

16.2 Accept–Reject with Loose Bounds

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

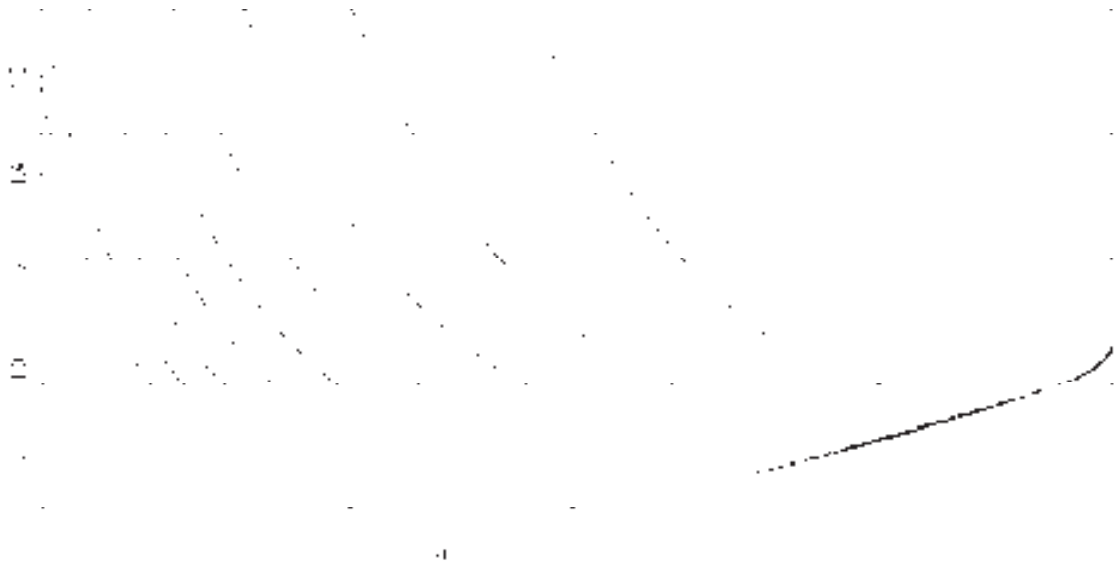


Fig. 1.12. [illegible]

1.6.3 Partitioning

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

5

Monte Carlo Optimization

[illegible]

[illegible]

5.1 Introduction

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 5.1. Signal processing. [illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

5.2 Stochastic Exploration

5.2.1 A Basic Solution

[illegible]

Example 5.2. A tiny Monte Carlo maximization. [illegible]

[illegible]

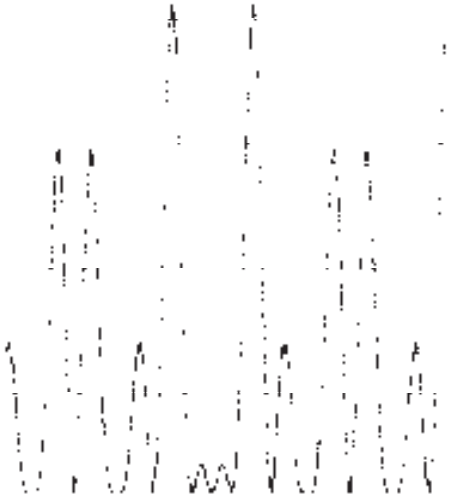


Fig. 5. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 5.1. Minimization of a complex function. [illegible]

$$h(x,y) = (x\sin(20y) + y\sin(20x))^2\cosh(\sin(10x)x) + (x\cos(10y) - y\sin(10x))^2\cosh(\cos(20y)y)\ ,$$

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. ‖

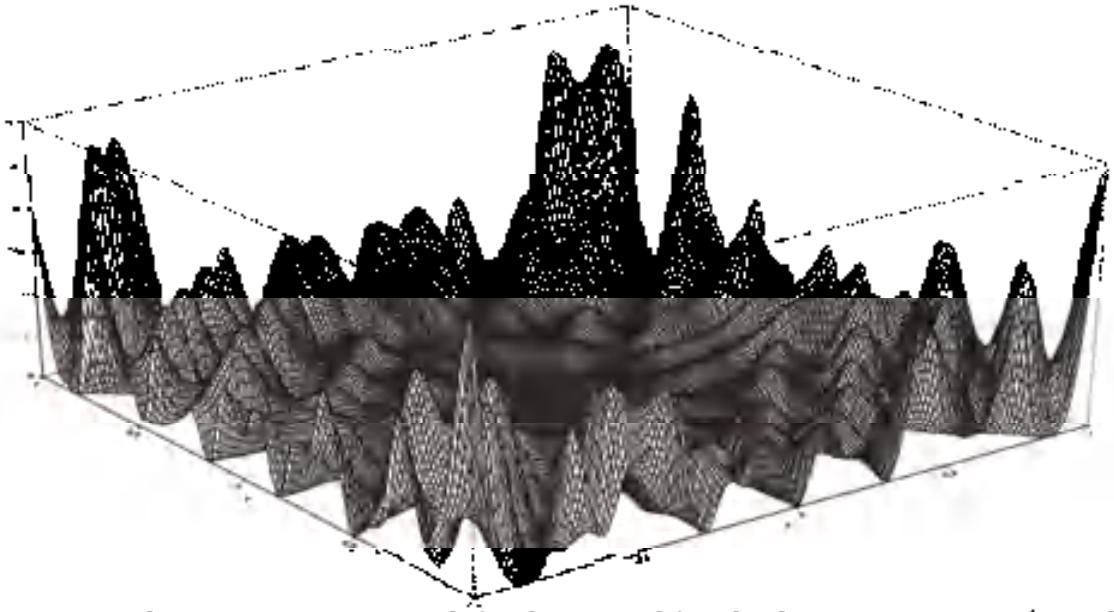


Fig. 5.2. Grid representation of the function $h(x,y)$ of Example 5.3 on $[-1,1]^2$.

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

$$h(\theta) = \int H(x,\theta)dx$$

[illegible]

[illegible]

5.2.2 Gradient Methods

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 5.1. (Continuation of Example 5.3): [illegible]

[illegible]

[illegible]

[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]

Table 5. [illegible]

[illegible]

5.2.3 Simulation Annealing

[illegible]

[illegible]

[illegible]

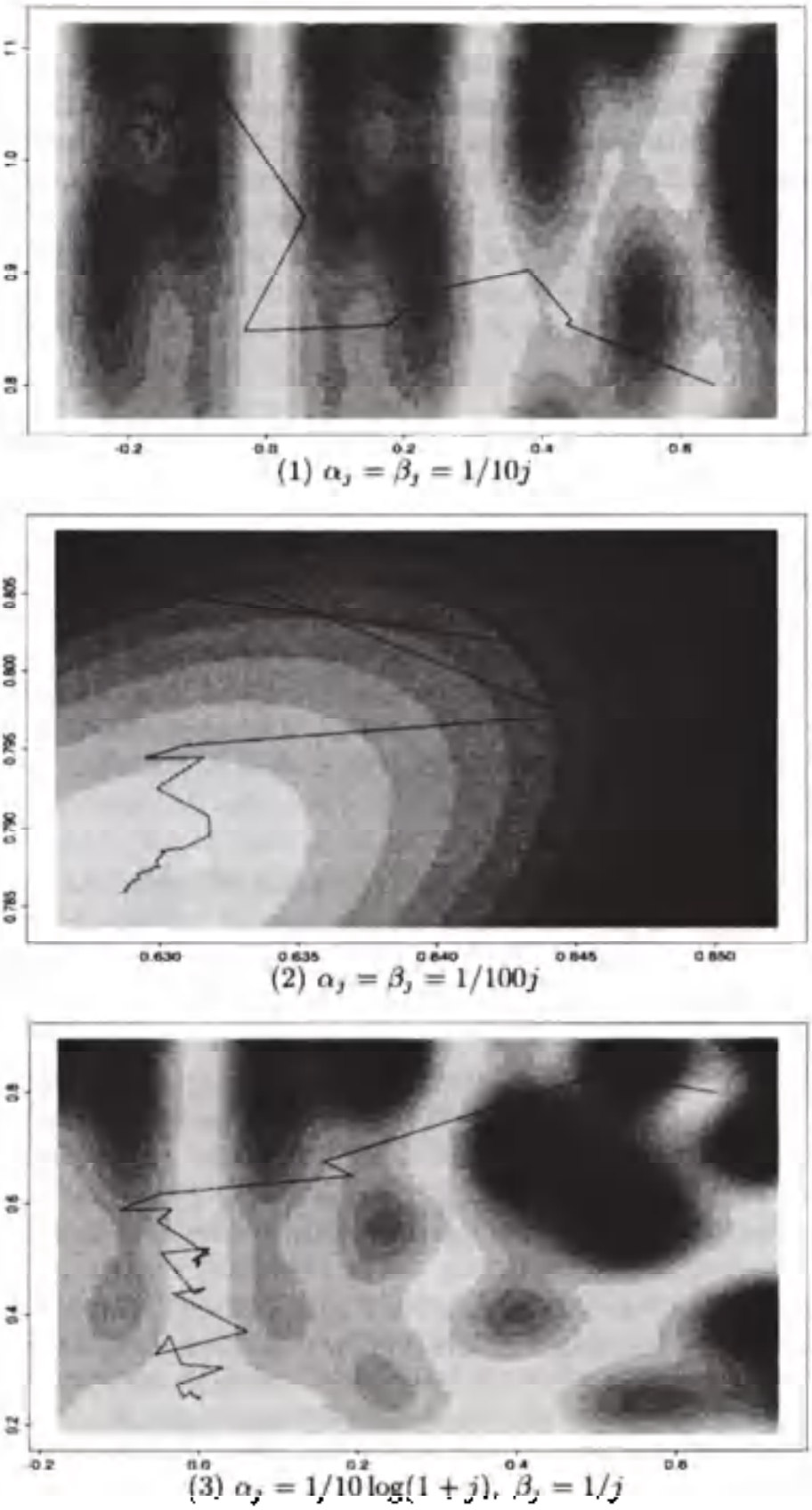


Fig. 5.3. Stochastic gradient paths for three different choices of the sequences (α_j) and (β_j) and starting point $(0.65, 0.8)$ for the same sequence (ζ_j) in (5.4). The gray levels are such that darker shades mean higher elevations. The function h to be minimized is defined in Example 5.3.

$$\pi(\theta) \propto \exp(h(\theta)/T)$$

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,

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

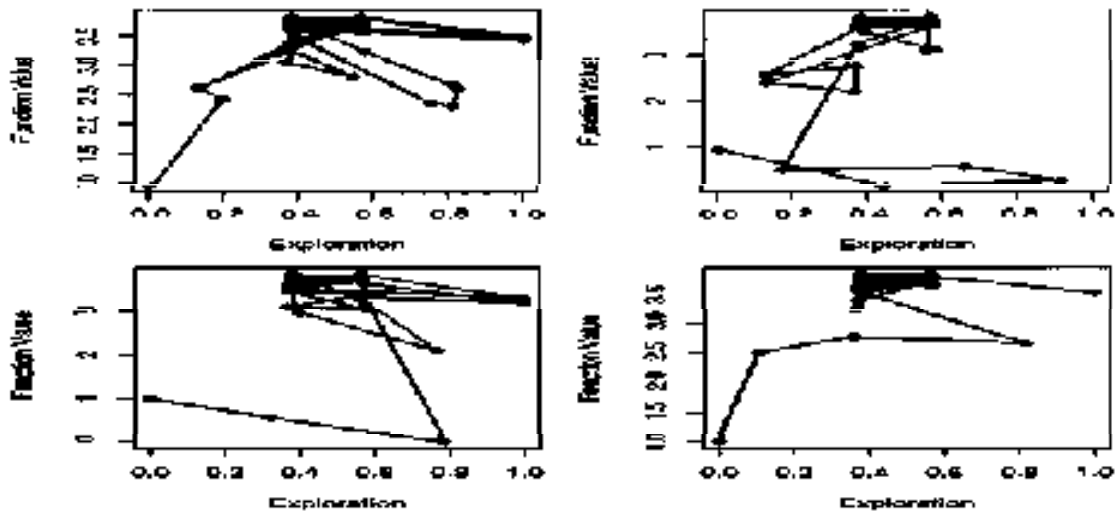


Fig. 5.4. Calculation of the maximum of the function (1.26), given in Example 5.5. The four different panels show the trajectory of 2500 pairs $(x^{(t)}, h(x^{(t)})$ for each of four different runs.

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 $u \sim \mathcal{U}(a_t, b_t)$ where $a_t = \max(x^{(t)} - r, 0)$ and $b_t = \min(x^{(t)} + r, 1)$
2. Accept $x^{(t+1)} = u$ with probability

$$\rho^{(t)} = \min\left\{\exp\left(\frac{h(u) - h(x^{(t)})}{T_t}\right), 1\right\},$$

   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.)

[illegible]

[illegible]

Definition 5.6. [illegible]

[illegible]

[illegible]

Theorem 5.7. [illegible]

$$[illegible]$$

[illegible]

$$\sum_{[illegible]}^{[illegible]} [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

Example 6.6. Being modeled [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

Table 5.2: [illegible] page 5.7

Example 5.31: (Continuation of Example 5.30) [illegible]

5.2.1 Price Feedback

[illegible]

Theorem 5.10: [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

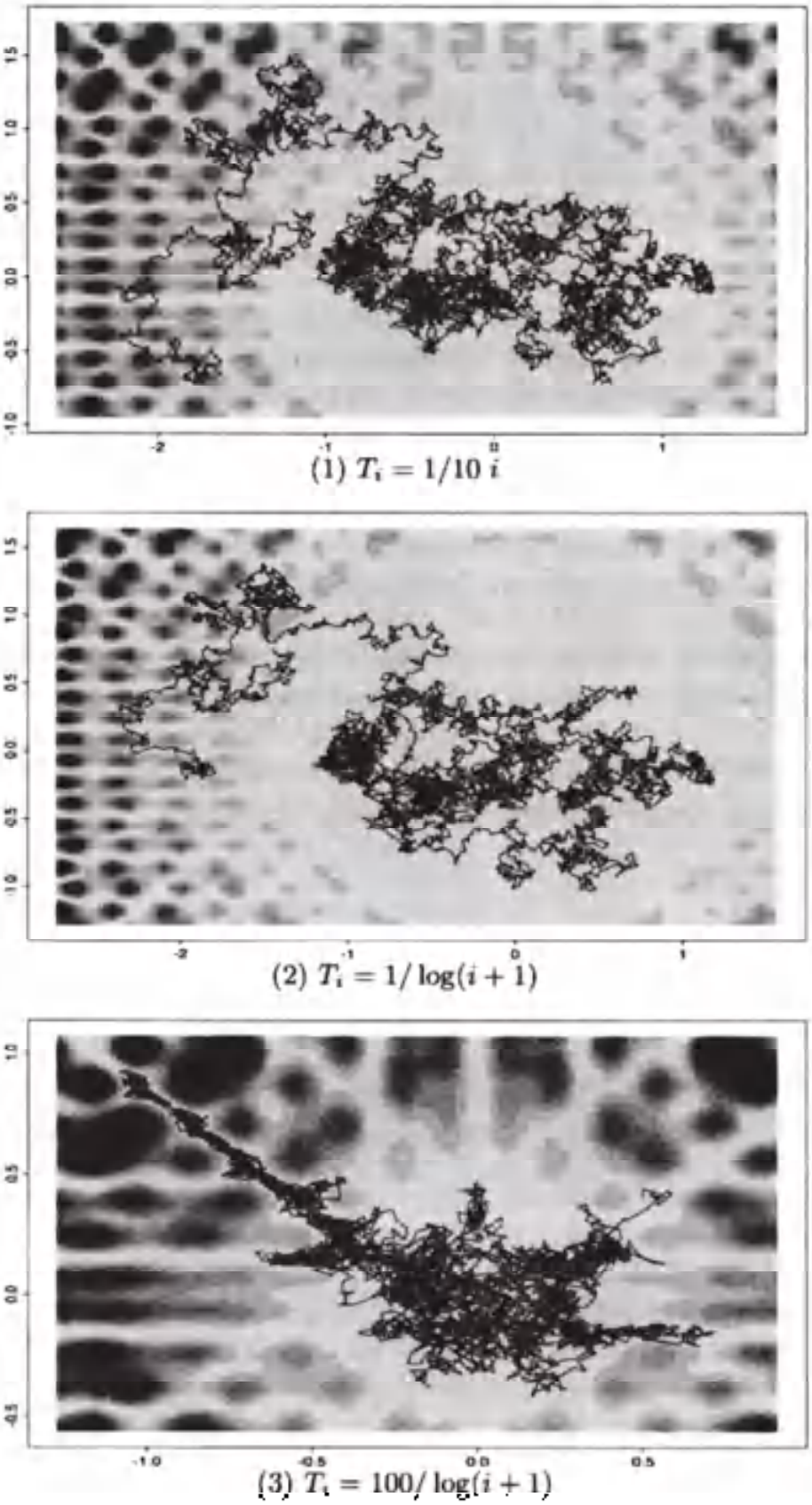


Fig. 5.5. Simulated annealing sequence of 5000 points for three different choices of the temperature T_i in [4.19] and starting point $(0.5, 0.4)$, aimed at minimizing the function h of Example 5.3.

[illegible]

[illegible]

Corollary 5.11. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Table 3.3. [illegible]

[illegible]

Example 3.12. Gamma shape estimation. [illegible]

[illegible]

[illegible]

[illegible]

Example 3.13. Second regression. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Table 5.1 [illegible]

[illegible]

Table 5.2. [illegible]

[illegible]

5.3 Stochastic Approximation

[illegible]

5.3.1 Missing Data Models and Demarginalization

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 5.14. Censored data likelihood. [illegible]

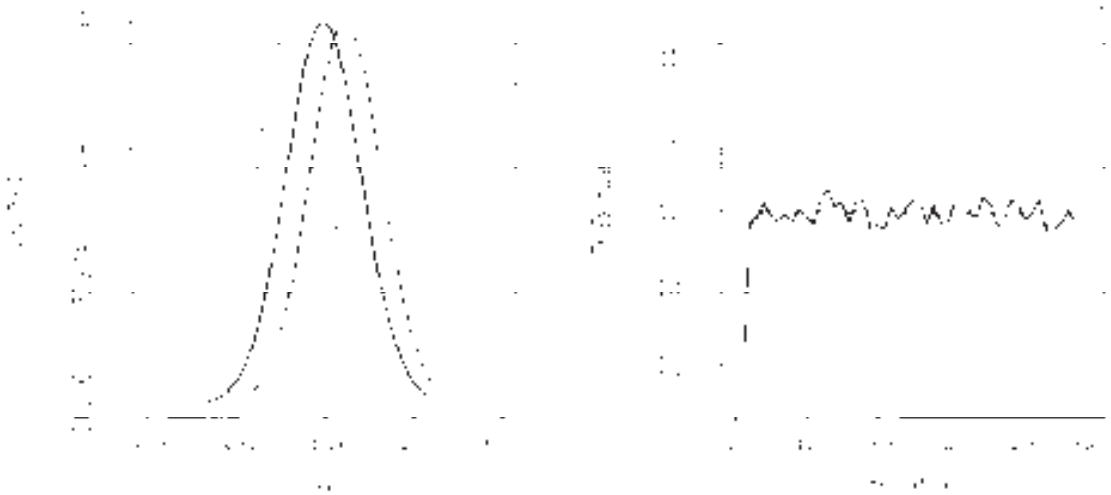


Fig. 5.6 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

6.3.2 The EM Algorithm

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

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

$$Q(\theta|\hat{\theta}_{(m)}, \mathbf{x}) = \mathbf{E}_{\hat{\theta}_{(m)}}[\log L^c(\theta|\mathbf{x}, \mathbf{z})],$$

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]

$$\theta_{(m+1)} = \arg\max_{\theta} Q(\theta|\hat{\theta}_{(m)}, \mathbf{x}).$$

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

$$\mathbf{E}_{\hat{\theta}_{(j)}}[\log k(\mathbf{Z}|\hat{\theta}_{(j+1)}, \mathbf{x})] \leq \mathbf{E}_{\hat{\theta}_{(j)}}[\log k(\mathbf{Z}|\hat{\theta}_{(j)}, \mathbf{x})] , \tag{5.14}$$

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

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Theorem 5.[illegible]. [illegible]

[illegible]

Example 5.17. PAC [illegible] consonant data. [illegible]

$$[illegible] \prod [illegible] \prod [illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

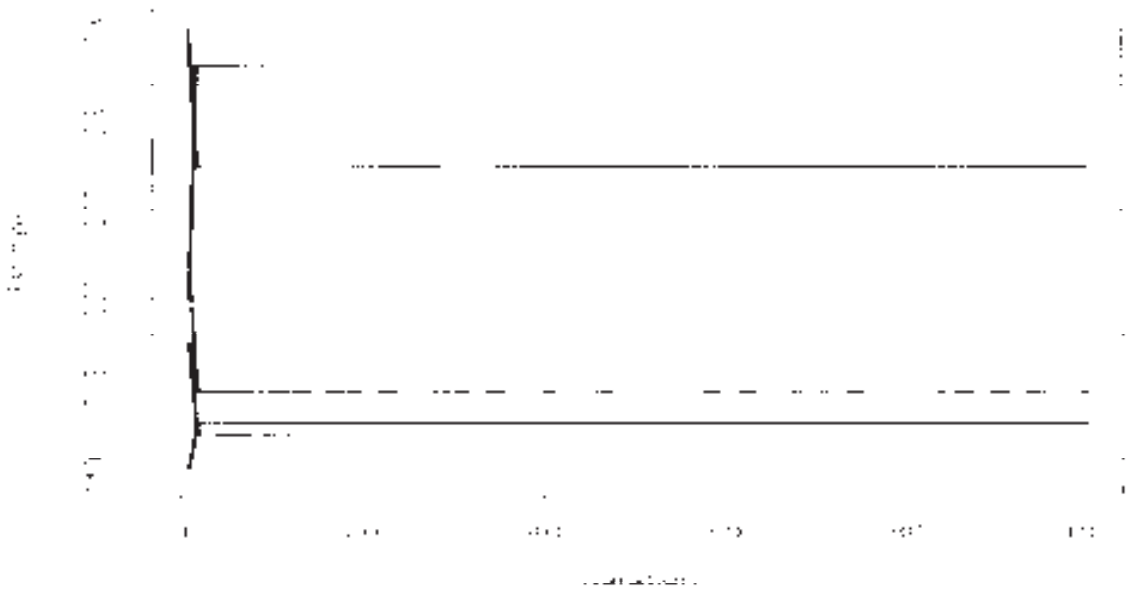


Fig. 6.7 [illegible]

[illegible]

Example 6.19. EM for mean mixtures of normal distributions. [illegible]

[illegible]

[illegible]

is close to the modal value, with improvement brought by further iterations being incremental. ‖

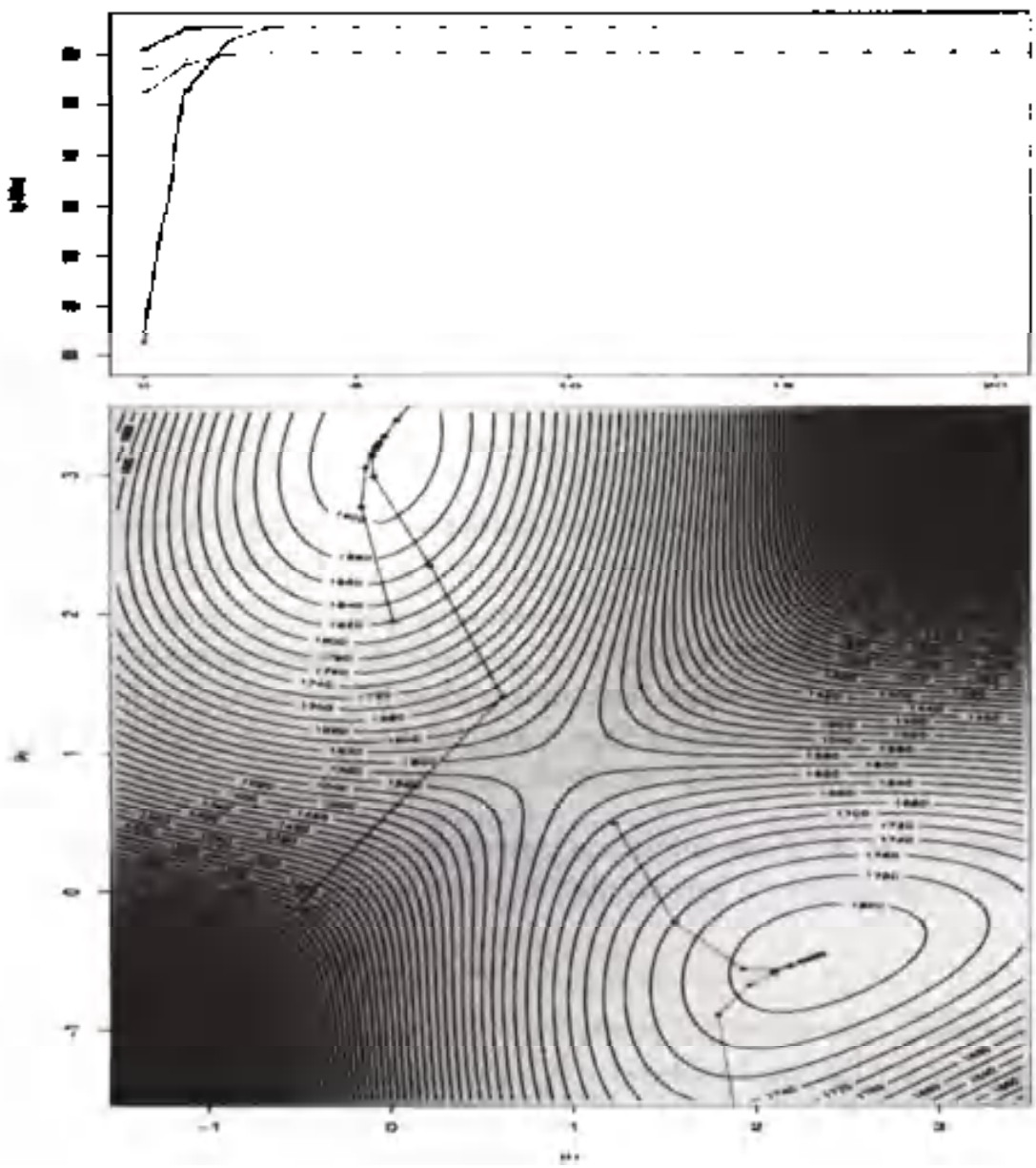


Fig. 5.8. Trajectories of five runs of the EM algorithm for Example 5.19 with their likelihood values *(top)* and their position on the likelihood surface *(bottom)*.

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).

5.3.4 Monte Carlo EM

[illegible]

$$[illegible]$$

[illegible]

Example 5.20. MCEM for censored data. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 5.21. Genetic Linkage. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 5.32 Capture-recapture models revisited. [illegible]

[illegible]

[illegible]

[illegible]

where $\mathbf{h}$ denotes the sequence of observed h_{ij}'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 $\hat{\theta}_k = \frac{1}{nt} \sum_{i=1}^{n} \sum_{j=1}^{t} y_{ijk} + \hat{z}_{ijk}$.
2. *(Monte Carlo E-step)* If $x_{ijk} = 0$, for $\ell = 1, \ldots, L$, generate

$$\hat{z}_{ijk\ell} \sim \mathcal{M}_k(\hat{\theta}_1, \ldots, \hat{\theta}_m)$$

and calculate

$$\hat{z}_{ijk} = \sum_{\ell} \hat{z}_{ijk\ell} / L.$$

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

[illegible]

[illegible]

5.8.1 SM Standard Errors

[illegible]

[illegible]

[illegible]

Fig. 5.8. [illegible]

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

[illegible]

Example 5.23. Genetic Energy [illegible]

$$[illegible]$$

[illegible]

5.4 Problems

5.1 [illegible]

5.2 [illegible]

5.3 [illegible]

5.4 [illegible]

5.5 [illegible]

5.6 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

5.16 [illegible]

[illegible]

5.17 [illegible]

[illegible]

5.18 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

5.20 [illegible]

[illegible]

[illegible]

5.21 [illegible]

[illegible]

5.22 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

5.24 [illegible]

5.25 [illegible]

[illegible]

[illegible]

5.26 [illegible]

5.27 [illegible]

5.28 [illegible]

5.29 [illegible]

[illegible]

[illegible]

Table [illegible]: [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

5.5 Notes

5.5.1 Variations on EM

[illegible]

$$\sum_{i=1}^{n} \frac{\partial \log p(x_i \mid \theta)}{\partial \theta} = 0$$

[illegible]

[illegible]

[illegible]

5.5.2 Neural Networks

[illegible]

[illegible]

[illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

5.5.3 The Robbins–Monro procedure

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 7.24 [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

5.7.4 Monte Carlo Approximation

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

(iii) The number of points z_i used in the approximation should vary with x to achieve the same precision in the approximation of $h(x)$, but this is usually impossible to assess in advance.

In the case $g(x) = f(x|x_0)$, Geyer's (1996) solution to (ii) is to use a recursive process in which x_0 is updated by the solution of the last optimization at each step. The Monte Carlo maximization algorithm then looks like the following:

Algorithm A.22 –Monte Carlo Maximization–

At step i

1. **Generate** $z_1, \ldots, z_m \sim f(z|x_i)$ **and compute** $\hat{h}_{g_i}$ **with** $g_i = f(\cdot|x_i)$.
2. **Find** $x^* = \arg\max \hat{h}_{g_i}(x)$.
3. **Update** x_i **to** $x_{i+1} = x^*$. [A.22]

Repeat until $x_i = x_{i+1}$.

Example 5.25. Maximum likelihood estimation for exponential families. Geyer and Thompson (1992) take advantage of this technique to derive maximum likelihood estimators in exponential families; that is, for functions

$$h(x|\theta) = c(\theta)e^{\theta x - \nu(x)} = c(\theta)\tilde{h}(x|\theta)\,.$$

However, $c(\theta)$ may be unknown or difficult to compute. Geyer and Thompson (1992, 1995) establish that maximization of $h(x|\theta)$ is equivalent to maximizing

$$\log \frac{\tilde{h}(x|\theta)}{\tilde{h}(x|\eta)} - \log \frac{1}{m} \sum_{i=1}^{m} \frac{\tilde{h}(x_i|\theta)}{\tilde{h}(x_i|\eta)}\,,$$

where the X_i's are generated from $h(x|\eta)$ (see Problem 5.11, and Geyer 1993, 1994).

This representation also extends to setups where the likelihood $L(\theta|x)$ is known up to a multiplicative constant (that is, where $L(\theta|x) \propto h(\theta|x)$). ∎

6

Markov Chains

[illegible]

6.1 Essentials for MCMC

[illegible]

Definition 6.1 [illegible]

$$\mathcal{X} \qquad \mathcal{Y}$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

6.2 Basic Notions

[illegible]

Definition 6.2. [illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

Example 6.3. **Bernoulli Lapse Model.** [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Definition 6.4. [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

Example 6.5. *Simplified Normalizing.* [illegible]

[illegible]

[illegible]

[illegible]

Lemma 6.7. (Chapman–Kolmogorov equations) [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Definition 6.8. [illegible]

$$[illegible] = \sum [illegible]$$

[illegible]

[illegible]

Proposition 6.9. (Weak Markov property) [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

Definition 6.10. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Proposition 6.11. Strong Markov property. [illegible]

[illegible]

[illegible]

Example 6.12 Code tossing. [illegible]

[illegible]

6.3 Irreducibility, Atoms, and Small Sets

6.3.1 Irreducibility

[illegible]

$$[illegible]$$

[illegible]

Definition 6.13. [illegible]

Example 6.14. [illegible]

$$[illegible]$$

[illegible]

Theorem 6.15. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.16. [illegible]

[illegible]

$$[illegible]$$

[illegible]

Example 6.17 (Continuation of Example 6.6)**.** [illegible]

$$[illegible]$$

[illegible]

6.3.2 Atoms and Small Sets

[illegible]

Definition 6.18. [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

Definition 6.19. [illegible]

[illegible]

Example 6.20 (Continuation of Example 6.17). [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.21. [illegible]

[illegible]

[illegible]

[illegible]

Definition 6.22. [illegible]

[illegible]

$$[illegible] \begin{cases} [illegible] \\ [illegible] \\ [illegible] \end{cases}$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Definition 6.23. [illegible]

[illegible]

Example 6.24 (Continuation of Example 6.11). [illegible]

[illegible]

6.4 Transience and Recurrence

6.4.1 Classification of Irreducible Chains

[illegible]

[illegible]

Definition 6.25. [illegible]

[illegible]

Example 6.26. Branching process. [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

Definition 6.27. [illegible]

$$[illegible]$$

Theorem 6.28. [illegible]

[illegible]

[illegible]

Definition 6.29. [illegible]

[illegible]

[illegible]

[illegible]

$$[illegible] \qquad [illegible]$$

[illegible]

$$[illegible] \qquad [illegible]$$

[illegible]

Theorem 6.30. [illegible]

6.4.2 Criteria for Recurrence

[illegible]

Proposition 6.31. [illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

6.4.3 Harris Recurrence

[illegible]

Definition 6.32. [illegible]

[illegible]

Proposition 6.33. [illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Theorem 6.34 [illegible]

[illegible]

[illegible]

6.5 Invariant Measures

6.5.1 Stationary Chains

[illegible]

Definition 6.35. [illegible]

$$[illegible]$$

[illegible]

[illegible]

Proposition 6.36. [illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

6.5.2 Kac's Theorem

[illegible]

$$[illegible]$$

[illegible]

Theorem 6.37. [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 6.40 (Random walk on [illegible]).

[illegible]

[illegible]

[illegible]

Example 6.41 (Continuation of Example 6.24). [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 6.12 (Continuation of Example 6.26) [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 6.13. (Continuation of Example 6.28) [illegible]

[illegible]

[illegible]

a representation of π that is associated with Kac's Theorem, under the form

$$\pi(A) = \sum_{t=1}^{\infty} \Pr(N_t \in A) \Pr(T^* = t), \tag{6.20}$$

where T^* is an integer valued random variable and, for each $t \in \mathbb{N}$, N_t is a random variable whose distribution depends on t. This representation is very closely related to Kac's (6.17), but offers a wider range of applicability since it does not require the chain to have an atom. We simply assume that the minorizing condition (6.10) is satisfied (and for simplicity's sake we take $m = 1$ in (6.10)). The random variables N_t and T^* can then be defined in terms of the split chain of Section 6.3.2. If $\tau_C \geq 1$ denotes the renewal time associated with the small set C in (6.10), then $\mathbb{E}_\nu[\tau_C] < \infty$ by recurrence (Problem 6.32), and T^* is given by the tail probabilities of τ_C as

$$P(T^* = t) = \frac{\Pr_\nu(\tau_C \geq t)}{\mathbb{E}_\nu(\tau_C)}. \tag{6.21}$$

The random variable N_t is then logically distributed from ν if $t = 1$ and as X_t conditional on no renewal before time t otherwise, following from (6.10). Breyer and Roberts (2000b) derive the representation (6.20) by the mean of a functional equation (see Problem 6.33).

Simulating from π thus amounts to simulating T^* according to (6.21) and then, for $T^* = t$, to simulating N_t. Simulating the latter starts from the minorizing measure ν and then runs $t-1$ steps of the residual distribution

$$\tilde{K}(x, \cdot) = \frac{K(x, \cdot) - \epsilon \mathbb{I}_C(x)\nu(\cdot)}{1 - \epsilon \mathbb{I}_C(x)}.$$

In cases when simulating from the residual is too complex, a brute force Accept–Reject implementation is to run the split chain t iterations until $\tau_C \geq t$, but this may be too time-consuming in many situations. Hobert and Robert (2004) also propose more advanced approaches to the simulation of T^*.

Note that, when the state space $\mathcal{X}$ is small, the chain is said to be *uniformly ergodic* (see Definition 6.58 below), $\tilde{K}(x, y) = (K(x, y) - \epsilon\nu(y))/(1 - \epsilon)$ and the mixture representation (6.20) translates into the following algorithm.

Algorithm A.23 –Kac's Mixture Implementation–

1. Simulate $X_0 \sim \nu$, $\omega \sim \mathcal{G}eo(\epsilon)$.
2. Run the transition $X_{t+1} \sim \tilde{K}(x_t, \cdot)$ $t = 0, \cdots, \omega - 1$, and take X_ω.

6.5.3 Reversibility and the Detailed Balance Condition

The stability property inherent to stationary chains can be related to another stability property called *reversibility*, which states that the direction of time

[illegible]

[illegible]

Definition 6.14. [illegible]

[illegible]

Definition 6.15. [illegible]

[illegible]

[illegible]

Theorem 6.16. [illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

6.6 Ergodicity and Convergence

6.6.1 Ergodicity

[illegible]

Definition 6.47. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

Proposition 6.48. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

$$[illegible] \sum [illegible] \sum [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.50. [illegible]

$$[illegible]$$

[illegible]

[illegible]

Theorem 6.51. [illegible]

$$[illegible]$$

[illegible]

[illegible]

Proposition 6.52. [illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

[illegible]

[illegible]

[illegible]

6.6.2 Geometric Convergence

[illegible]

$$[illegible]$$

Definition 6.54. [illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Definition 6.55. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Theorem 6.56. [illegible]

$$[illegible]$$

Example 6.57. Nonparametric regression [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

6.6.3 Ergodicity

[illegible]

Definition 6.38. [illegible]

$$[illegible]$$

[illegible]

Theorem 6.39. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

6.7 Finite Theorems

[illegible]

6.7.1 Logodic Uncertainty

[illegible]

$$[illegible]$$

[illegible]

Definition 6.6.6. [illegible]

$$[illegible]$$

[illegible]

Proposition 6.6.1. [illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

Lemma 6.62. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.63. Ergodic Theorem: [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

6.7.2 Control Flow Linearization

[illegible]

6.7.2.1 The Discrete Case

[illegible]

Theorem 6.64. [illegible]

[illegible]

[illegible]

[

6.7.2.2 Reversibility

[illegible]

[illegible]

Theorem 6.65. [illegible]

$$[illegible] \sum_{[illegible]} [illegible]$$

[illegible]

Example 6.66 (Continuation of Example 6.13)**.** [illegible]

$$[illegible]$$

[illegible]

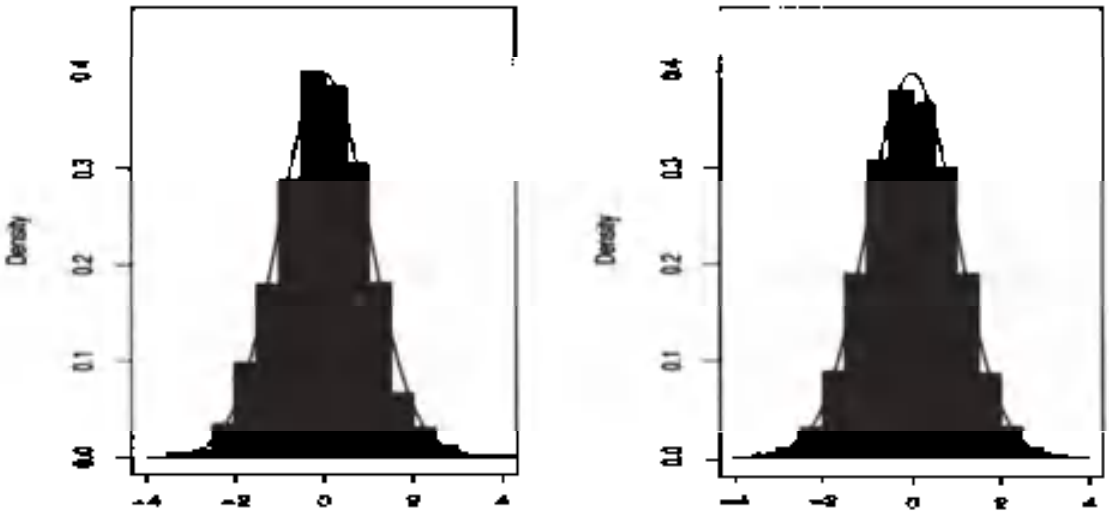


Fig. 6.2. *Histogram of 2500 means (each based on 50 observations) from an AR(1) chain.* The left panel corresponds to $\theta = .5$, which results in an ergodic chain. The right panel corresponds to $\theta = 2$, which corresponds to a transient chain.

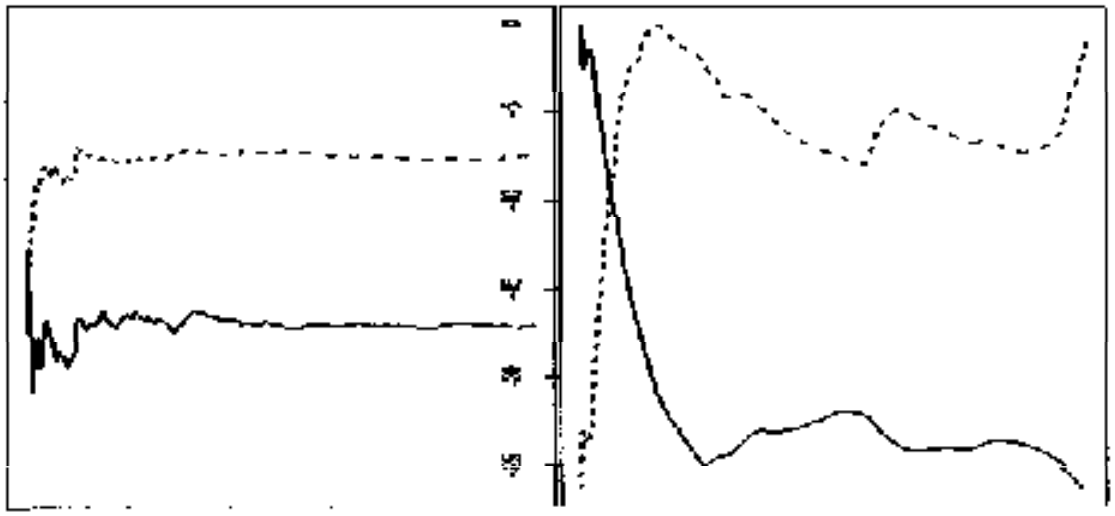


Fig. 6.3. Trajectories of mean (solid line) and standard deviation (dashed line) from the AR(1) process of Example 6.66. The left panel has $\theta = .5$, resulting in an ergodic Markov chain, and displays convergence of the mean and standard deviation. The right panel has $\theta = 1.0001$, resulting in a transient Markov chain and no convergence.

6.7.2.3 Geometric Ergodicity and Regeneration

There is yet another approach to the Central Limit Theorem for Markov chains. It relies on geometric ergodicity, a Liapounov-type moment condition on the function h, and a regeneration argument. Hobert et al. (2002), extending work of Chan and Geyer (1994) (see Problem 6.66), give specific conditions for Theorem 6.67 to apply, namely for Liapounov condition to apply and a *consistent estimate of* γ_h^2 *to be found*.

Theorem 6.67. *If* (X_n) *is aperiodic, irreducible, positive Harris recurrent with invariant distribution* π *and geometrically ergodic, and if, in addition,*

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

6.8 Problems

6.1 [illegible]

6.2 [illegible]

6.3 [illegible]

[illegible]

[illegible]

[illegible]

6.4 [illegible]

6.5 [illegible]

6.6 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

6.7 [illegible]

[illegible]
[illegible]
[illegible]
[illegible]

[illegible]

6.8 [illegible]

6.9 [illegible]

6.10 [illegible]

6.11 [illegible]

6.12 [illegible]

[illegible]

[illegible]

6.13 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

6.13 [illegible]

6.16 [illegible]

6.17 [illegible]

6.18 [illegible]

6.19 [illegible]

6.20 [illegible]

6.21 [illegible]

[illegible]

[illegible]

6.22 [illegible]

[illegible]

[illegible]

[illegible]

6.23 [illegible]

[illegible]

6.24 [illegible]

[illegible]

[illegible]

6.25 [illegible]

6.26 [illegible]

6.27 [illegible]

6.28 [illegible]

6.29 [illegible]

6.30 [illegible]

6.31 [illegible]

[illegible]

6.32 [illegible]

6.33 [illegible]

[illegible]

[illegible]

[illegible]

6.34 [illegible]

6.35 [illegible]

6.36 [illegible]

6.37 [illegible]

6.38 [illegible]

[illegible]

[illegible]

6.39 [illegible]

[illegible]

[illegible]

6.11 [illegible]

6.12 [illegible]

6.13 [illegible]

[illegible]

6.14 [illegible]

6.15 [illegible]

6.16 [illegible]

[illegible]

6.17 [illegible]

6.18 [illegible]

6.19 [illegible]

[illegible]

[illegible]

6.50 [illegible]

[illegible]

6.51 [illegible]

[illegible]

6.52 [illegible]

6.53 [illegible]

[illegible]

6.54 [illegible]

[illegible]

[illegible]

6.15 [illegible]

6.16 [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

6.17 [illegible]

$$[illegible]$$

[illegible]

6.18 [illegible]

[illegible]

6.19 [illegible]

[illegible]

6.20 [illegible]

6.61 [illegible]

[illegible]

[illegible]

6.62 [illegible]

6.63 [illegible]

6.64 [illegible]

6.65 [illegible]

[illegible]

[illegible]

6.66 [illegible]

[illegible]

6.67 [illegible]

6.68 [illegible]

6.69 [illegible]

[illegible]

[illegible]

[illegible]

6.9 Notes

6.9. [illegible] Conditions

[illegible]

[illegible]

[illegible]

Lemma 6.70. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.71. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.72. [illegible]

[illegible]

[illegible]

[illegible]

Example 6.73. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.74. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

6.9.1 Lotoh's Admissibility Condition

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.78. [illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.79. [illegible]

[illegible]

[illegible]

[illegible]

6.11.4 Alternative Convergence Conditions

[illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

Theorem 6.8.1 [illegible]

$$[illegible]$$

[illegible]

6.11.4 Mixing Conditions and Central Limit Theorems

[illegible]

[illegible]

[illegible]

Definition 6.81. [illegible]

[illegible]

[illegible]

[illegible]

Proposition 6.82. [illegible]

[illegible]

[illegible]

[illegible]

Theorem 6.83. [illegible]

[illegible]

[illegible]

6.9.5 Convergence in Markov Chains

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 6.8.1. Consider the [illegible] Example 6.6.1 [illegible] M:

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

7

The Metropolis–Hastings Algorithm

[illegible]

[illegible]

7.1 The MCMC Principle

[illegible]

[illegible]

[illegible]

[illegible]

Definition 7.1. [illegible]

[illegible]

[illegible]

7.2 Monte Carlo Methods Based on Markov Chains

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

requires some knowledge of the target density to derive some conditional densities, but it can also be more effective than a generic Metropolis Hastings algorithm.

7.3 The Metropolis–Hastings algorithm

Before illustrating the universality of Metropolis–Hastings algorithms and demonstrating their straightforward implementation, we first address the (important) issue of theoretical validity. Since the results presented below are valid for all types of Metropolis–Hastings algorithms, we do not include examples in this section, but rather wait for Sections 7.4 and 7.5, which present a collection of specific algorithms.

7.3.1 Definition

The Metropolis–Hastings algorithm starts with the objective *(target)* density f. A conditional density $q(y|x)$, defined with respect to the dominating measure for the model, is then chosen. The Metropolis Hastings algorithm can be implemented in practice when $q(\cdot|x)$ is easy to simulate from and is either explicitly available (up to a multiplicative constant *independent of* x) or *symmetric*; that is, such that $q(x|y) = q(y|x)$. The target density f must be available to some extent: a general requirement is that the ratio

$$f(y)/q(y|x)$$

is known up to a constant *independent* of x.

The Metropolis–Hastings algorithm associated with the objective (target) density f and the conditional density q produces a Markov chain $(X^{(t)})$ through the following transition.

Algorithm A.24 –Metropolis–Hastings–

Given $x^{(t)}$,

1. Generate $Y_t \sim q(y|x^{(t)})$,
2. Take

$$X^{(t+1)} = \begin{cases} Y_t & \text{with probability } \rho(x^{(t)}, Y_t), \\ x^{(t)} & \text{with probability } 1 - \rho(x^{(t)}, Y_t), \end{cases}$$

where [A.24]

$$\rho(x,y) = \min\left\{\frac{f(y)}{f(x)}\,\frac{q(x|y)}{q(y|x)}, 1\right\}.$$

[illegible]

[illegible]

[illegible]

Theorem 7.2. [illegible]

[illegible]

[illegible]

7.2 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

7.3.2 Convergence Properties

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Lemma 7.3. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 7.4. [illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Corollary 7.5. [illegible]

[illegible]

Lemma 7.6. [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Corollary 7.7. [illegible]

[illegible]

where most of the mass of the density f is located. This issue of selecting a good proposal distribution q for a given f is detailed in Section 7.6.

Since we have provided no examples so far, we now proceed to describe two particular approaches used in the literature, with some probabilistic properties and corresponding examples. Note that a complete classification of the Metropolis–Hastings algorithms is impossible, given the versatility of the method and the possibility of creating even more hybrid methods (see, for instance, Roberts and Tweedie 1995, 2004 and Stramer and Tweedie 1999b).

7.4 The Independent Metropolis–Hastings Algorithm

7.4.1 Fixed Proposals

This method appears as a straightforward generalization of the Accept–Reject method in the sense that the instrumental distribution q is independent of $X^{(t)}$ and is denoted g by analogy. The algorithm [A.24] will then produce the following transition from $x^{(t)}$ to $X^{(t+1)}$.

Algorithm A.25 –Independent Metropolis–Hastings–

Given $x^{(t)}$

1 Generate $Y_t \sim g(y)$.

2 Take [A.25]

$$X^{(t+1)} = \begin{cases} Y_t & \text{with probability } \min\left\{\dfrac{f(Y_t)\,g(x^{(t)})}{f(x^{(t)})\,g(Y_t)}, 1\right\} \\ x^{(t)} & \text{otherwise.} \end{cases}$$

Although the Y_t's are generated independently, the resulting sample is not iid: for instance, the probability of acceptance of Y_t depends on $X^{(t)}$ (except in the trivial case when $f = g$).

The convergence properties of the chain $(X^{(t)})$ follow from properties of the density g in the sense that $(X^{(t)})$ is irreducible and aperiodic (thus, ergodic according to Corollary 7.5) if and only if g is almost everywhere positive on the support of f. Stronger properties of convergence like geometric and uniform ergodicity are also clearly described by the following result of Mengersen and Tweedie (1996).

Theorem 7.8. *The algorithm [A.25] produces a uniformly ergodic chain if there exists a constant M such that*

$$f(x) \leq Mg(x)\,, \quad \forall x \in \operatorname{supp} f. \tag{7.7}$$

In this case,

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

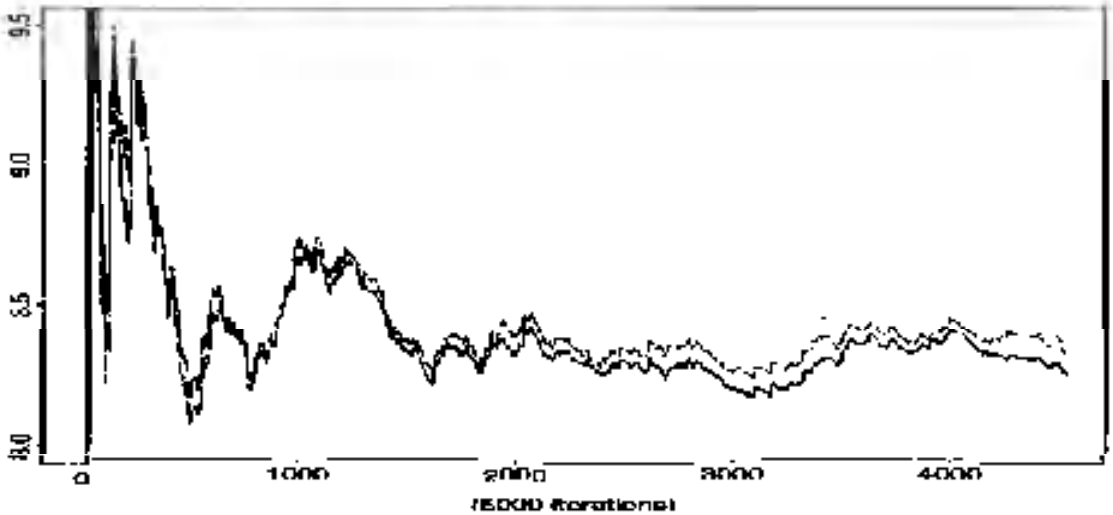


Fig. 7.1. Convergence of Accept–Reject (solid line) and Metropolis Hastings (dashed line) estimators to $\mathbb{E}_f[X^2] = 8.33$, for $\alpha = 2.43$ based on the same sequence $y_1, \ldots, y_{5000}$ simulated from $\mathcal{G}a(2, 2/2.43)$. The number of acceptances in [A.27] is then random. The final values of the estimators are 8.25 for [A.27] and 8.32 for [A.26].

Algorithm A.26 –Gamma Metropolis–Hastings–

1. Generate $Y_t \sim \mathcal{G}a(\lfloor\alpha\rfloor, \lfloor\alpha\rfloor/\alpha)$.
2. Take [A.26]

$$X^{(t+1)} = \begin{cases} Y_t & \text{with probability } \varrho_t \\ x^{(t)} & \text{otherwise,} \end{cases}$$

where

$$\varrho_t = \min\left[\left(\frac{Y_t}{x^{(t)}} \exp\left\{\frac{x^{(t)} - Y_t}{\alpha}\right\}\right)^{\alpha - \lfloor\alpha\rfloor}, 1\right].$$

and

Algorithm A.27 –Gamma Accept–Reject–

1. Generate $Y \sim \mathcal{G}a(\lfloor\alpha\rfloor, \lfloor\alpha\rfloor/\alpha)$.
2. Accept $X = Y$ with probability [A.27]

$$\left(\frac{ey\exp(-y/\alpha)}{\alpha}\right)^{\alpha - \lfloor\alpha\rfloor}$$

Note that (7.7) does apply in this particular case with $\exp(x/\alpha)/x > e/\alpha$.

A first comparison is based on a sample $(y_1, \ldots, y_n)$, of fixed size n, generated from $\mathcal{G}a(\lfloor\alpha\rfloor, \lfloor\alpha\rfloor/\alpha)$ with $x^{(0)}$ generated from $\mathcal{G}a(\alpha, 1)$. The number

Fig. 7.2. [illegible]

[illegible]

[illegible]

Example 7.1. Logistic Regression. [illegible]

[illegible]

[illegible]

$$[illegible] = \frac{[illegible]}{[illegible]} = \frac{[illegible]}{[illegible]}$$

[illegible]

[illegible]

[illegible]

$$[illegible] = \frac{[illegible]}{[illegible]} = \frac{[illegible]}{[illegible]} = [illegible]$$

[illegible]

[illegible]

[illegible]

$$[illegible] = \frac{[illegible]}{[illegible]}$$

[illegible]

Example 7.12 Solid epsilon tail area approximation. [illegible]

[illegible]

[illegible]

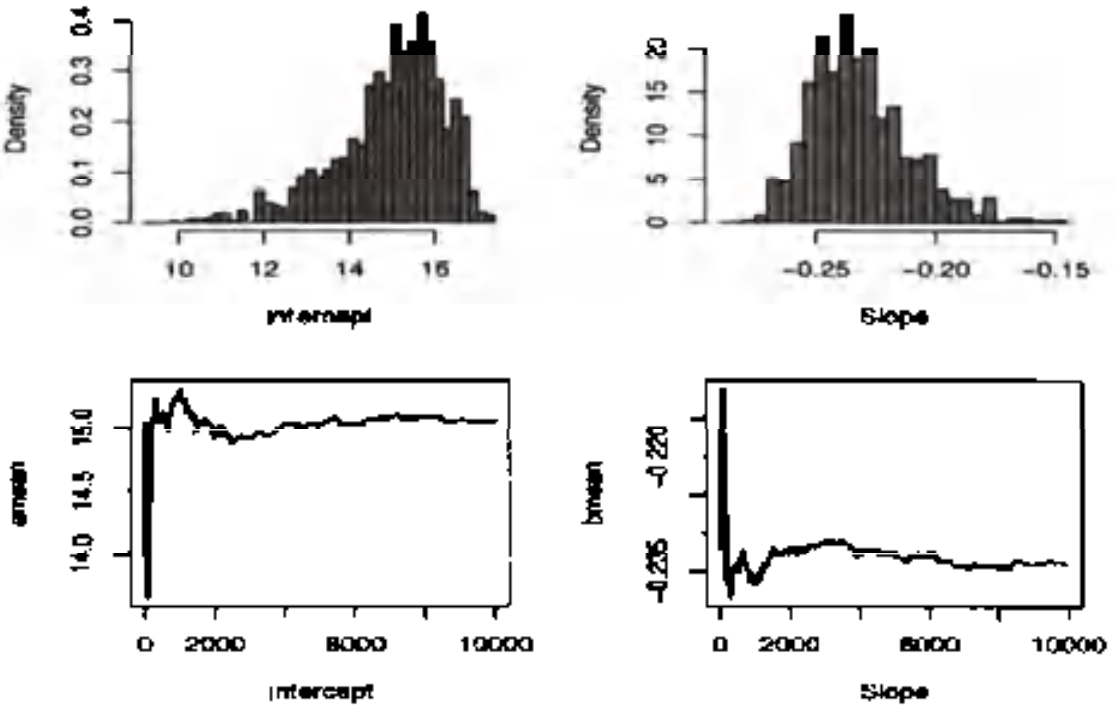


Fig. 7.3. Estimation of the slope and intercept from the Challenger logistic regression. The top panels show histograms of the distribution of the coefficients, while the bottom panels show the convergence of the means.

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

$$\approx \frac{1}{m}\sum_{i=1}^{m} \mathbb{I}[Z_i > \hat{\tau}(a)] \,, \tag{7.13}$$

where $K_X(\tau)$ is the cumulant generating function of X and $\hat{\tau}(x)$ is the solution of the saddlepoint equation $K'(\hat{\tau}(x)) = x$ (see Section 3.6.2).

Note that we are simulating from the transformed density. It is interesting (and useful) to note that we can easily derive an instrumental density to use in a Metropolis–Hastings algorithm. Using a Taylor series approximation, we find that

$$\exp\{n[K_X(t) - tK_X'(t)]\} \approx \exp\left\{-nK_X''(0)\frac{t^2}{2}\right\} \,, \tag{7.14}$$

so a first choice for an instrumental density is the $\mathcal{N}(0, 1/nK_X''(0))$ distribution (see Problem 7.26 for details). Booth et al. (1999) use a Student's t approximation instead.

We can now simulate the noncentral chi squared tail areas using a normal instrumental density with $K_X''(t) = 2[p(1-2t) + 4\lambda]/(1-2t)^3$. The results are presented in Table 7.1, where we see that the approximations are quite good. Note that the same set of simulated random variables can be used for all the tail area probability calculations. Moreover, by using the Metropolis–

[illegible]

[illegible]

[illegible]		
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]

Tabela 7.1. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

7.4.2 A Metropolis–Hastings Version of ARS

[illegible]

[illegible]

[illegible]

Algorithm A.28 –ARMS Metropolis–Hastings–

1. **Simulate** Y **from** $g_n(y)$ **and** $U \sim \mathcal{U}_{[0,1]}$ **until**
$$U \le f_1(Y)/\exp\{\bar{h}_n(Y)\}.$$
2. **Generate** $V \sim \mathcal{U}_{[0,1]}$ **and take** [A.28]

$$X^{(t+1)} = \begin{cases} Y & \text{if} \quad V < \dfrac{f_1(Y)\,\psi_n(x^{(t)})}{f_1(x^{(t)})\,\psi_n(Y)} \wedge 1 \\ x^{(t)} & \text{otherwise.} \end{cases} \tag{7.15}$$

The Accept–Reject step indeed produces a variable distributed from $\psi_n(x)$ and this justifies the expression of the acceptance probability in the Metropolis Hastings step. Note that [A.28] is a particular case of the approximate Accept–Reject algorithms considered by Tierney (1994) (see Problem 7.9). The probability (7.15) can also be written

$$\begin{cases} \min\left[1, \dfrac{f_1(Y)\exp\{\bar{h}_n(x^{(t)})\}}{f_1(x^{(t)})\exp\{\bar{h}_n(Y)\}}\right] & \text{if} \quad f_1(Y) > \exp\{\bar{h}_n(Y)\}, \\ \min\left[1, \dfrac{\exp\{\bar{h}_n(x^{(t)})\}}{f_1(x^{(t)})}\right] & \text{otherwise,} \end{cases}$$

which implies a sure acceptance of Y when $f_1(x^{(t)}) < \exp\{\bar{h}_n(x^{(t)})\}$; that is, when the bound is correct.

Each simulation of $Y \sim g_n$ in Step 1 of [A.28] provides, in addition, an update of S_n in $S_{n+1} = S_n \cup \{y\}$, and therefore of g_n, when Y is rejected. As in the case of the ARS algorithm, the initial S_n set must be chosen so that g_n is truly a probability density. If the support of f is not bounded from below, $L_{0,1}$ must be increasing and, similarly, if the support of f is not bounded from above, $L_{n,n+1}$ must be decreasing. Note also that the simulation of g_n detailed in Section 2.4.2 is valid in this setting.

Since the algorithm [A.28] appears to be a particular case of independent Metropolis Hastings algorithm, the convergence and ergodicity results obtained in Section 7.4 should apply for [A.28]. This is not the case, however, because of the lack of time homogeneity of the chain (see Definition 6.4) produced by [A.28]. The transition kernel, based on g_n, can change at each step with a positive probability. Since the study of nonhomogeneous chains is quite delicate, the algorithm [A.28] can be justified only by reverting to the homogeneous case; that is, by fixing the function g_n and the set S_n after a warm-up period of length n_0. The constant n_0 need not be fixed in advance as this warm-up period can conclude when the approximation of f_1 by g_n is satisfactory, for instance when the rejection rate in Step 1 of [A.28] is sufficiently small. The algorithm [A.28] must then start with an initializing (or

[illegible]

Example 7.13. Poisson logistic model. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

7.5 Random Walks

[illegible]

$$[illegible]$$

[illegible]

δ	0.1	0.5	1.0
Mean	0.399	0.111	0.10
Variance	0.698	1.11	1.06

Table 7.2. Estimators of the mean and the variance of a normal distribution $\mathcal{N}(0,1)$ based on a sample obtained by a Metropolis–Hastings algorithm using a random walk on $[-\delta,\delta]$ (15,000 simulations).

The convergence results of Section 7.3.2 naturally apply in this particular case. Following Lemma 7.6, if g is positive in a neighborhood of 0, the chain $(X^{(t)})$ is f-irreducible and aperiodic, therefore ergodic. The most common distributions g in this setup are the uniform distributions on spheres centered at the origin or standard distributions like the normal and the Student's t distributions. All these distributions usually need to be scaled; we discuss this problem in Section 7.6. At this point, we note that the choice of a *symmetric function* g (that is, such that $g(-t) = g(t)$), leads to the following original expression of [A.24], as proposed by Metropolis et al. (1953).

Algorithm A.29 –Random walk Metropolis–Hastings–

Given $x^{(t)}$,

1. Generate $Y_t \sim g(|y - x^{(t)}|)$.
2. Take [A.29]

$$X^{(t+1)} = \begin{cases} Y_t & \text{with probability } \min\left\{1, \dfrac{f(Y_t)}{f(x^{(t)})}\right\} \\ x^{(t)} & \text{otherwise.} \end{cases}$$

Example 7.14. A random walk normal generator. Hastings (1970) considers the generation of the normal distribution $\mathcal{N}(0,1)$ based on the uniform distribution on $[-\delta,\delta]$. The probability of acceptance is then $\rho(x^{(t)}, y_t) = \exp\{((x^{(t)})^2 - y_t^2)/2\} \wedge 1$. Figure 7.4 describes three samples of 15,000 points produced by this method for $\delta = 0.1, 0.5$, and 1. The corresponding estimates of the mean and variance are provided in Table 7.2. Figure 7.4 clearly shows the different speeds of convergence of the averages associated with these three values of δ, with an increasing regularity (in δ) of the corresponding histograms and a faster exploration of the support of f. ‖

Despite its simplicity and its natural features, the random walk Metropolis–Hastings algorithm does not enjoy uniform ergodicity properties. Mengersen and Tweedie (1996) have shown that in the case where supp $f = \mathbb{R}$, this algorithm cannot produce a uniformly ergodic Markov chain on $\mathbb{R}$ (Problem 7.16). This is a rather unsurprising feature when considering the *local* character of

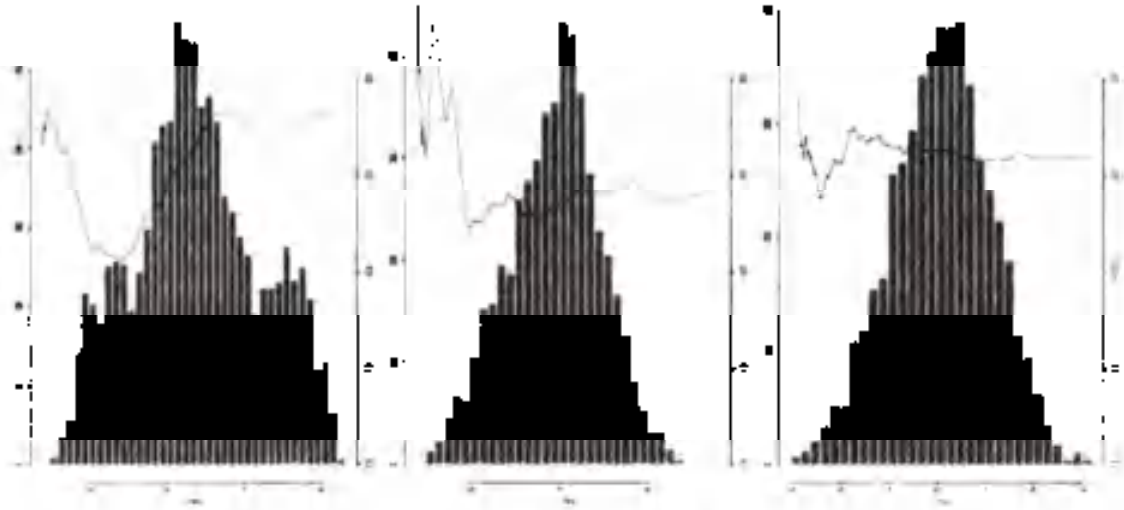


Fig. 7.4. Histograms of three samples produced by the algorithm [A.29] for a random walk on $[-\delta, \delta]$ with (a) $\delta = 0.1$, (b) $\delta = 0.5$, and (c) $\delta = 1.0$, with the convergence of the means (7.1), superimposed with scales on the right of the graphs (15,000 simulations).

the random walk proposal, centered at the current value of the Markov chain.

Although uniform ergodicity cannot be obtained with random walk Metropolis–Hastings algorithms, it is possible to derive necessary and sufficient conditions for geometric ergodicity. Mengersen and Tweedie (1996) have proposed a condition based on the *log-concavity of f* in the tails; that is, if there exist $\alpha > 0$ and x_1 such that

$$\log f(x) - \log f(y) \geq \alpha|y - x| \tag{7.16}$$

for $y < x < x_1$ or $x_1 < x < y$.

Theorem 7.15. *Consider a symmetric density f which is log-concave with associated constant α in (7.16) for $|x|$ large enough. If the density g is positive and symmetric, the chain $(X^{(t)})$ of [A.29] is geometrically ergodic. If f is not symmetric, a sufficient condition for geometric ergodicity is that $g(t)$ be bounded by $b\exp\{-\alpha|t|\}$ for a sufficiently large constant b.*

The proof of this result is based on the use of the drift function $V(x) = \exp\{\alpha|x|/2\}$ (see Note 6.9.1) and the verification of a geometric drift condition of the form

$$\Delta V(x) \leq -\lambda V(x) + b\mathbb{I}_{[-x^*,x^*]}(x)\,, \tag{7.17}$$

for a suitable bound x^*. Mengersen and Tweedie (1996) have shown, in addition, that this condition on g is also necessary in the sense that if $(X^{(t)})$ is geometrically ergodic, there exists $s > 0$ such that

$$\int e^{s|x|}\, f(x)dx < \infty\,. \tag{7.18}$$

Fig. 7.5. [illegible]

Example 7.16. A comparison of LCP efficiency. [illegible]

$$[illegible]$$

[illegible]

Example 7.17. Random walk geometric generation. [illegible]

[illegible]

7.6 Optimization and Control

[illegible]

7.6.1 Optimizing the Acceptance Rate

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

7.6.2 Conditioning and Accelerations

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]

Table 7.4. [illegible]

[illegible]

[illegible]

Theorem 7.19. [illegible]

[illegible]

[illegible]

[illegible]

Example 7.20. Rao-Blackwellization improvement for a [illegible] simulation. [illegible]

[illegible]

[illegible]

Figure 7.21. [illegible]

[illegible]

[illegible]

Example 7.22 (Continuation of Example 7.20). [illegible]

[illegible]

Table 7.5. [illegible]

[illegible]

7.6.3 Adaptive Schemes

[illegible]

[illegible]

[illegible]

Example 7.23. [illegible]

[illegible]

[illegible]

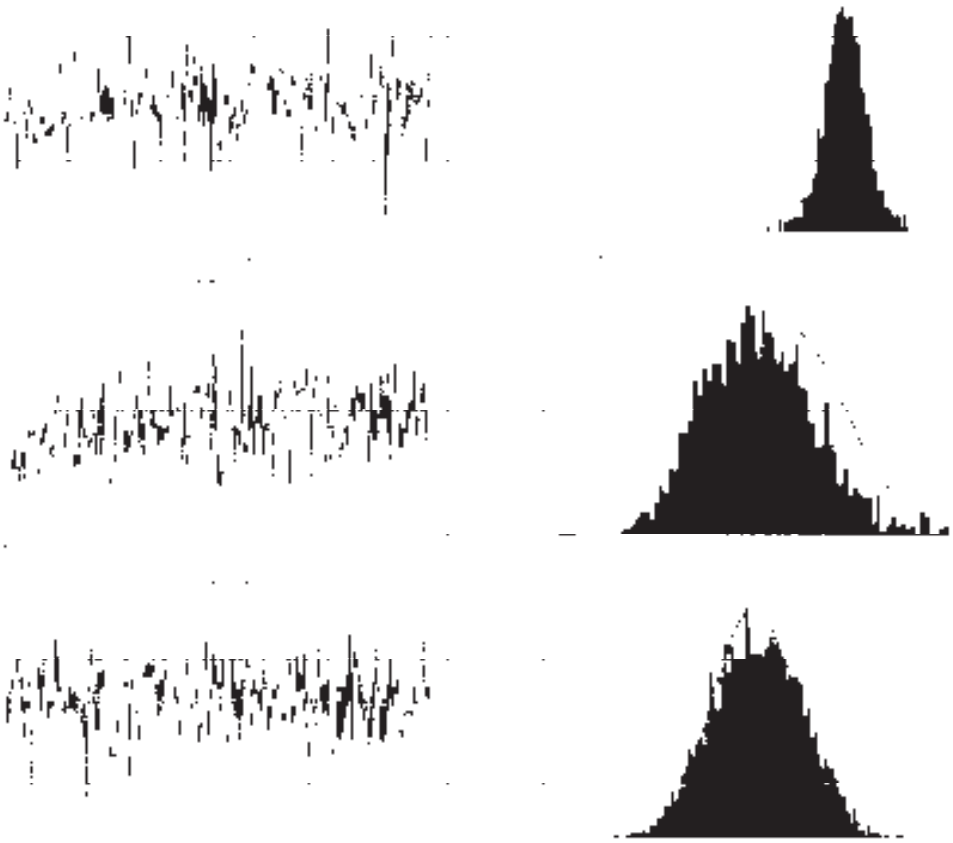
[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



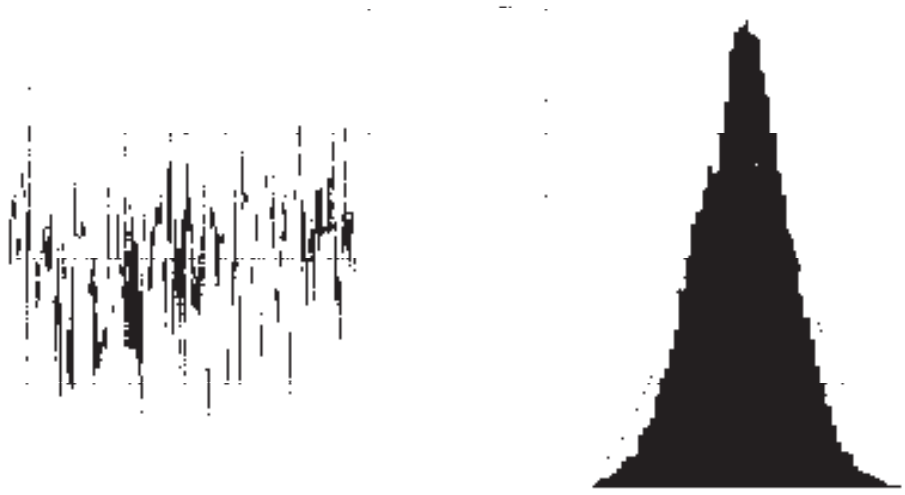


Fig. 7.8. [illegible]

[illegible]

[illegible]

7.7 Problems

7.1 [illegible]

[illegible]

In each case monitor the convergence.

7.2 Student's $\mathcal{T}_\nu$ density with ν degrees of freedom is given by

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

Calculate the mean of a t distribution with 4 degrees of freedom using a Metropolis–Hastings algorithm with candidate density

(a) $\mathrm{N}(0,1)$

(b) t with 2 degrees of freedom.

Monitor the convergence of each.

7.3 Complete some details of Theorem 7.2.

(a) To establish (7.2), show that

$$\begin{aligned} K(x,A) &= P(X_{t+1} \in A | X_t = x) \\ &= P(Y \in A \text{ and } X_{t+1} = Y | X_t = x) + P(x \in A \text{ and } X_{t+1} = x | X_t = x) \\ &= \int_A q(y|x)\varrho(x,y)dy + \int_{\mathcal{Y}} \mathbb{1}(x \in A)(1 - \varrho(x,y))q(y|x)dy, \end{aligned}$$

where $q(y|x)$ is the instrumental density and $\varrho(x,y) = P(X_{t+1} = y | X_t = x)$. Take the limiting case $A = \{y\}$ to establish (7.2).

(b) Establish (7.3). Notice that $\delta_y(x)f(y) = \delta_x(y)f(x)$.

7.4 For the transition kernel,

$$X^{(t+1)}|x^{(t)} \sim \mathcal{N}(\rho x^{(t)}, \tau^2)$$

gives sufficient conditions on ρ and τ for the stationary distribution π to exist. Show that, in this case, π is a normal distribution and that (7.4) occurs.

7.5 (Doukhan et al. 1994) The algorithm presented in this problem is used in Chapter 12 as a benchmark for slow convergence.

(a) Prove the following result:

Lemma 7.24. *Consider a probability density g on $[0,1]$ and a function $0 < \rho < 1$ such that*

$$\int_0^1 \frac{g(x)}{1-\rho(x)}\, dx < \infty .$$

The Markov chain with transition kernel

$$K(x,x') = \rho(x)\, \delta_x(x') + (1-\rho(x))\, g(x') ,$$

where δ_x is the Dirac mass at x, has stationary distribution

$$f(x) \propto g(x)/(1-\rho(x)).$$

(b) Show that an algorithm for generating the Markov chain associated with Lemma 7.24 is given by

Algorithm A.30 –Repeat or Simulate–

```
1. Take X^(t+1) = x^(t) with probability ρ(x^(t))
2. Else, generate X^(t+1) ~ g(y).                    [A.30]
```

(c) Highlight the similarity with the Accept–Reject algorithm and discuss in which sense they are complementary.

7.6 (Continuation of Problem 7.5) Implement the algorithm of Problem 7.5 when g is the density of the $\mathcal{B}e(\alpha+1,1)$ distribution and $\rho(x) = 1 - x$. Give the expression of the stationary distribution f. Study the acceptance rate as α varies around 1. (*Note:* Doukhan et al. 1994 use this example to derive β-mixing chains which do not satisfy the Central Limit Theorem.)

7.7 (Continuation of Problem 7.5) Compare the algorithm [A.30] with the corresponding Metropolis–Hastings algorithm; that is, the algorithm [A.25] associated with the same pair (f, g). (*Hint:* Take into account the fact that [A.30] simulates only the y_t's which are not discarded and compare the computing times when a recycling version as in Section 7.6.2 is implemented.)

7.8 Determine the distribution of Y_t given $y_{t-1}, \ldots$ in [A.25].

7.9 (Tierney 1994) Consider a version of [A.25] based on a "bound" M on f/g that is not a uniform bound; that is, $f(x)/g(x) > M$ for some x.

(a) If an Accept–Reject algorithm uses the density g with acceptance probability $f(y)/Mg(y)$, show that the resulting variables are generated from

$$\tilde{f}(x) \propto \min\{f(x), Mg(x)\},$$

instead of f.

(b) Show that this error can be corrected, for instance by using the Metropolis–Hastings algorithm:

1. Generate $Y_t \sim \tilde{f}$.
2. Accept with probability

$$P(X^{(t+1)} = y_t | x^{(t)}, y_t) = \begin{cases} \min\left\{1, \dfrac{f(y_t)g(x^{(t)})}{g(y_t)f(x^{(t)})}\right\} & \text{if } \dfrac{f(y_t)}{g(y_t)} > M \\ \min\left\{1, \dfrac{Mg(x^{(t)})}{f(x^{(t)})}\right\} & \text{otherwise.} \end{cases}$$

to produce a sample from f.

7.10 The inequality (7.8) can also be established using *Orey's inequality* (See Problem 6.42 for a slightly different formulation) For two transitions P and Q,

$$\|P^n - Q^n\|_{TV} \leq 2P(X_n \neq Y_n), \qquad X_n \sim P^n, \quad Y_n \sim Q^n.$$

Deduce that when P is associated with the stationary distribution f and when X_n is generated by [A.25], under the condition (7.7),

$$\|P^n - f\|_{TV} \leq \left(1 - \frac{1}{M}\right)^n .$$

Hint: Use a coupling argument based on

$$X^n = \begin{cases} Y^n & \text{with probability } 1/M \\ Z^n \sim \dfrac{g(z) - f(z)/M}{1 - 1/M} & \text{otherwise.} \end{cases}$$

7.11 Complete the proof of Theorem 7.8:

[illegible]

7.12 [illegible]

7.13 [illegible]

7.14 [illegible]

[illegible]

7.15 [illegible]

7.16 [illegible]

Theorem 7.25 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

7.17 [illegible]

Theorem 7.26. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

7.18 [illegible]

[illegible]

[illegible]

7.19 [illegible]

7.20 [illegible]

[illegible]

Table 7.16. [illegible]

7.21 [illegible]

[illegible]

[illegible]

7.24 [illegible]

$$[illegible]$$

[illegible]

7.26 [illegible]

7.27 [illegible]

7.26 [illegible]

7.27 [illegible]

7.28 [illegible]

7.29 [illegible]

7.30 [illegible]

7.31 [illegible]

[illegible]

[illegible]

[illegible]

7.33 [illegible]

7.34 [illegible]

7.35 [illegible]

7.36 [illegible]

7.37 [illegible]

7.38 [illegible]

7.39 [illegible]

[illegible]

[illegible]

7.40 (Continuation of Problem 7.39) Deduce from Problem 7.39 that for a given instrumental matrix Q in a Metropolis–Hastings algorithm, the choice

$$p_{ij}^* = q_{ij}\left(\frac{\pi_j q_{ji}}{\pi_i q_{ij}} \wedge 1\right)$$

is optimal among the transitions such that

$$p_{ij} = \frac{q_{ij}s_{ij}}{1 + \dfrac{\pi_i q_{ij}}{\pi_j q_{ji}}} = q_{ij}\alpha_{ij}\,,$$

where $s_{ij} = s_{ji}$ and $0 \le \alpha_{ij} \le 1$. (*Hint:* Give the corresponding α_{ij} for the Metropolis–Hastings algorithm and show that it is maximal for $i \neq j$. Tierney 1998, Mira and Geyer 1998, and Tierney and Mira 1998 propose extensions to the continuous case.)

7.41 Show that f is the stationary density associated with the acceptance probability (7.20).

7.42 In the setting of Example 7.28, implement the simulated annealing algorithm to find the maximum of the likelihood $L(\theta|x_1, x_2, x_3)$. Compare with the performances based on $\log L(\theta|x_1, x_2, x_3)$.

7.43 (Winkler 1995) A *Potts model* is defined on a set S of "sites" and a finite set G of "colors" by its energy

$$H(x) = -\sum_{(s,t)} \alpha_{st} \mathbb{I}_{x_s = x_t}, \qquad x \in G^S,$$

where $\alpha_{st} = \alpha_{ts}$, the corresponding distribution being $\pi(x) \propto \exp(H(x))$. An additional structure is introduced as follows: "Bonds" b are associated with each pair (s,t) such that $\alpha_{st} > 0$. These bonds are either active ($b = 1$) or inactive ($b = 0$).

(a) Defining the joint distribution

$$\mu(x,b) \propto \prod_{b_{st}=0} q_{st} \prod_{b_{st}=1} (1 - q_{st}) \mathbb{I}_{x_s = x_t},$$

with $q_{st} = \exp(\alpha_{st})$, show that the marginal of μ in x is π. Show that the marginal of μ in b is

$$\mu(b) \propto |G|^{c(b)} \prod_{b_{st}=0} q_{st} \prod_{b_{st}=1} (1 - q_{st}),$$

where $c(b)$ denotes the number of *clusters* (the number of sites connected by active bonds).

(b) Show that the *Swendson–Wang* (1987) algorithm

1. Take $b_{st} = 0$ if $x_s \neq x_t$ and, for $x_s = x_t$,

$$b_{st} = \begin{cases} 1 & \text{with probability } 1 - q_{st} \\ 0 & \text{otherwise.} \end{cases}$$

2. For every cluster, choose a color at random on G.

leads to simulations from π (*Note:* This algorithm is acknowledged as accelerating convergence in image processing.)

7.44 (McCulloch 1997) In a generalized linear mixed model, assume the link function is $h(\xi_i) = x_i'\beta + z_i'\mathbf{b}$, and further assume that $\mathbf{b} = (b_1, \ldots, b_l)$ where $\mathbf{b} \sim f_{\mathbf{b}}(\mathbf{b}|D)$. (Here, we assume φ to be unknown.)

(a) Show that the usual (incomplete-data) likelihood is

$$L(\theta, \varphi, D|y) = \int \prod_{i=1}^{n} f(y_i|\theta_i) f_{\mathbf{b}}(\mathbf{b}|D) d\mathbf{b} .$$

(b) Denote the complete data by $\mathbf{w} = (\mathbf{y}, \mathbf{b})$, and show that

$$\log L_{\mathbf{W}} = \sum_{i=1}^{n} \log f(y_i|\theta_i) + \log f_{\mathbf{b}}(b_i|D) .$$

(c) Show that the EM algorithm, given by

1. Choose starting values $\beta^{(0)}$, $\varphi^{(0)}$, and $D^{(0)}$.
2. Calculate (expectations evaluated under $\beta^{(m)}$, $\varphi^{(m)}$, and $D^{(m)}$) $\beta^{(m+1)}$ and $\varphi^{(m+1)}$, which maximize $\mathbf{E}[\log f(y_i|\theta_i, u, \beta, \varphi)|y]$.
3. $D^{(m+1)}$ maximizes $\mathbf{E}[f_{\mathbf{b}}(\mathbf{b}|D)|y]$.
4. Set m to $m+1$.

converges to the MLE.

The next problems (7.45–7.49) deal with Langevin diffusions, as introduced in Section 7.8.5.

7.45 Show that the naïve discretization of (7.22) as $dt = \sigma^2$, $dL_t = X^{[t+\sigma^2]} - X^{[t]}$, and $dB_t = B_{t+\sigma^2} - B_t$ does lead to the representation (7.23).

7.46 Consider f to be the density of $\mathcal{N}(0,1)$. Show that when $\sigma = 2$ in (7.23), the limiting distribution of the chain is $\mathcal{N}(0,2)$.

7.47 Show that (7.27) can be directly simulated as

$$\theta_2 \sim \mathcal{N}\left(\frac{y}{3}, \frac{5}{6}\right) , \qquad \theta_1|\theta_2 \sim \mathcal{N}\left(\frac{2y - \theta_2}{5}, \frac{4}{5}\right) .$$

7.48 Show that when (7.24) exists and is larger (smaller) than 1 (−1) at −∞ (∞), the random walk (7.23) is transient.

7.49 (Stramer and Tweedie 1999a) Show that the following stochastic differential equation still produces f as the stationary distribution of the associated process:

$$dL_t = \sigma(L_t)\nabla \log f(L_t) + b(L_t)dt,$$

when

$$b(x) = \frac{1}{2}\nabla \log f(x)\sigma^2(x) + \sigma(x)\nabla\sigma(x).$$

Give a discretized version of this differential equation to derive a Metropolis–Hastings algorithm and apply to the case $\sigma(x) = \exp(\omega|x|)$.

7.8 Notes

7.8.1 Background of the Metropolis Algorithm

The original Metropolis algorithm was introduced by Metropolis et al. (1953) in a setup of optimization on a discrete state-space, in connection with particle physics:

7.8.2 Geometric Convergence of Metropolis-Hastings Algorithms

[illegible]

$$[illegible]$$

[illegible]

Theorem 7.27. [illegible]

$$[illegible]$$

[illegible]

[illegible]

7.8.3 A Reinterpretation of Simulated Annealing

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

7.8.4 Reference Acceptance Rates

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Table 7.7. [illegible]

[illegible]

Example 7.28. Cauchy posterior distribution. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

7.8.5 Interpretation Algorithm

[illegible]

7.9 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 7.23 Nonlinearizable normal model. [illegible]

[illegible]

[illegible]

[

8

The Slice Sampler

[illegible]

8.1 Another Look at the Fundamental Theorem

[illegible]

$$\mathscr{S}(f) = \{(x, u); 0 \leq u \leq f(x)\} ,$$

whatever the dimension of x, and f need only be known up to a normalizing constant.

From the development at the beginning of Chapter 7, we can consider the possibility of using a Markov chain with stationary distribution equal to this uniform distribution on $\mathscr{S}(f)$ as an approximate way to simulate from f. A natural solution is to use a *random walk* on $\mathscr{S}(f)$, since a random walk on a set $\mathscr{A}$ usually results in a stationary distribution that is the uniform distribution on $\mathscr{A}$ (see Examples 6.39, 6.40, and 6.73).

There are many ways of implementing a random walk on this set, but a natural solution is to go one direction at a time, that is, to move iteratively along the u-axis and then along the x-axis. Furthermore, we can use uniform moves on both directions, since, as formally shown below, the associated Markov chain on $\mathscr{S}(f)$ does not require a Metropolis–Hastings correction to have the uniform distribution on $\mathscr{S}(f)$ as stationary distribution. Starting from a point (x, u) in $\{(x, u) : 0 < u < f(x)\}$, the move along the u-axis will correspond to the conditional distribution

$$U|X = x \sim \mathcal{U}(\{u : u < f(x)\}), \tag{8.1}$$

resulting in a change from point (x, u) to point (x, u'), still in $\mathscr{S}(f)$, and then the move along the x-axis to the conditional distribution

$$X|U = u' \sim \mathcal{U}(\{x : u' \leq f(x)\}), \tag{8.2}$$

resulting in a change from point (x, u') to point (x', u').

This set of proposals is the basis chosen for the original *slice sampler* of Neal (1997) (published as Neal 2003) and Damien et al. (1999), which thus uses a 2-step uniform random walk over the subgraph. We inaccurately call it the *2D slice sampler* to distinguish it from the general slice sampler defined in Section 8.2, even though the dimension of x is arbitrary.

Algorithm A.31 –2D slice sampler–

At iteration t, simulate

1. $u^{(t+1)} \sim \mathcal{U}_{[0, f(x^{(t)})]}$; [A.31]
2. $x^{(t+1)} \sim \mathcal{U}_{A^{(t+1)}}$, **with**

$$A^{(t+1)} = \{x : f(x) \geq u^{(t+1)}\}.$$

From (8.1) it is also clear that $x^{(t)}$ is always part of the set $A^{(t+1)}$, which is thus nonempty. Moreover, the algorithm remains valid if $f(x) = Cf_1(x)$, and we use f_1 instead of f (see Problem 8.1). This is quite advantageous in settings where f is an unnormalized density like a posterior density.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example S.1. Simple slice sampler. [illegible]

[illegible]

[illegible]

Example S.2. Improper uniform distribution. [illegible]

[illegible]

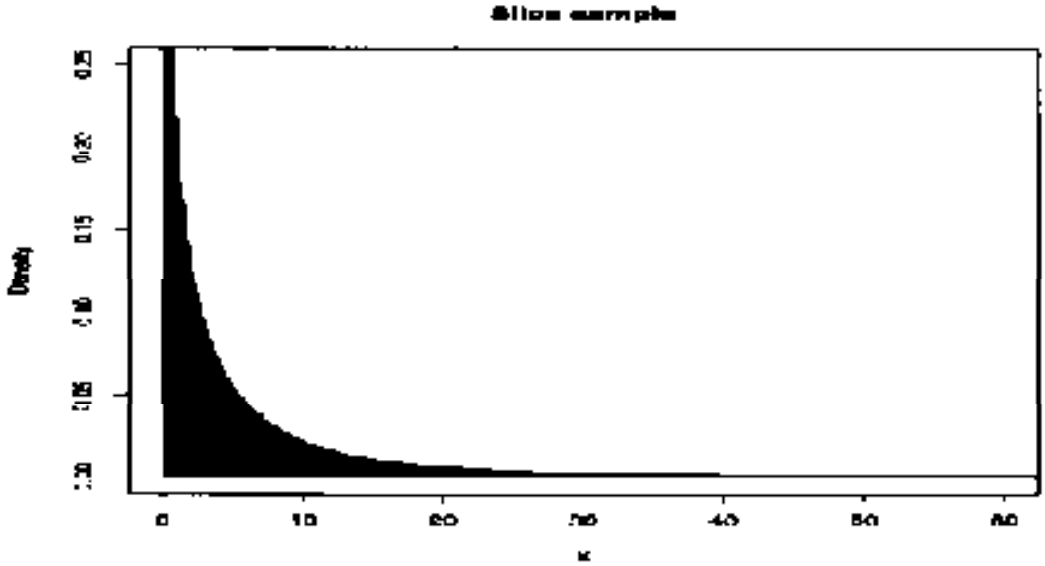


Fig. 8.1. The slice sampler histogram and density for Example 8.1.

Accept–Reject algorithm of Example 2.20 can be costly if the algorithm is to be used only a few times. The slice sampler [A.31] applied to this problem is then associated with the horizontal slice

$$A^{(t-1)} = \{y;\ \exp\{-(y-3)^2/2\} \geq u\, f_1(x^{(t)})\}\,.$$

If we denote $\omega^{(t)}$ the value $u\, f_1(x^{(t)})$, the slice is also given by

$$A^{(t+1)} = \{y \in [0,1];\ (y-3)^2 \leq -2\log(\omega^{(t)})\}\,,$$

which is an interval of the form $[0, \gamma^{(t)}]$. Figure 8.2 shows the first ten steps of [A.31] started from $x^{(0)} = 0.25$. ||

This algorithm will work well only if the exploration of the subgraph of f_1 by the corresponding random walk is fast enough; if we take the above example of the truncated normal, this is the case. Whatever the value of $x^{(t)}$, the next value $x^{(t+1)}$ can be anything in $[0,1]$. This property actually holds for all f's: given that, when $\omega^{(t+1)}$ is close to 0, the set $A^{(t+1)}$ is close to the entire support of f and, formally, every value $x^{(t+1)}$ in the support of f can be simulated in one step. (We detail below some more advanced results about convergence properties of the slice sampler.)

Figure 8.3 illustrates the very limited dependence on the starting value for the truncated normal example: the three graphs represent a sample of values of $x^{(10)}$ when $(x^{(0)}, \omega^{(1)})$ is in the upper left hand corner, the lower right hand corner and the middle of the subgraph, respectively. To study the effect of the starting value, we always used the same sequence of uniforms for the three starting points. As can be seen from the plot, the samples are uniformly spread out over the subgraph of f and, more importantly, almost identical!

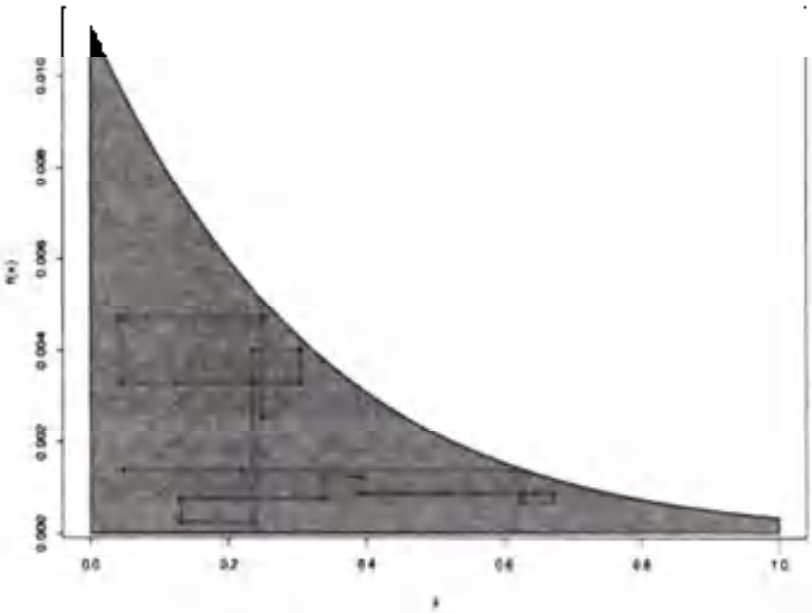


Fig. 8.2. First ten steps of the slice sampler for the truncated normal distribution $\mathcal{N}(3,1)$, restricted to the interval $[0,1]$.

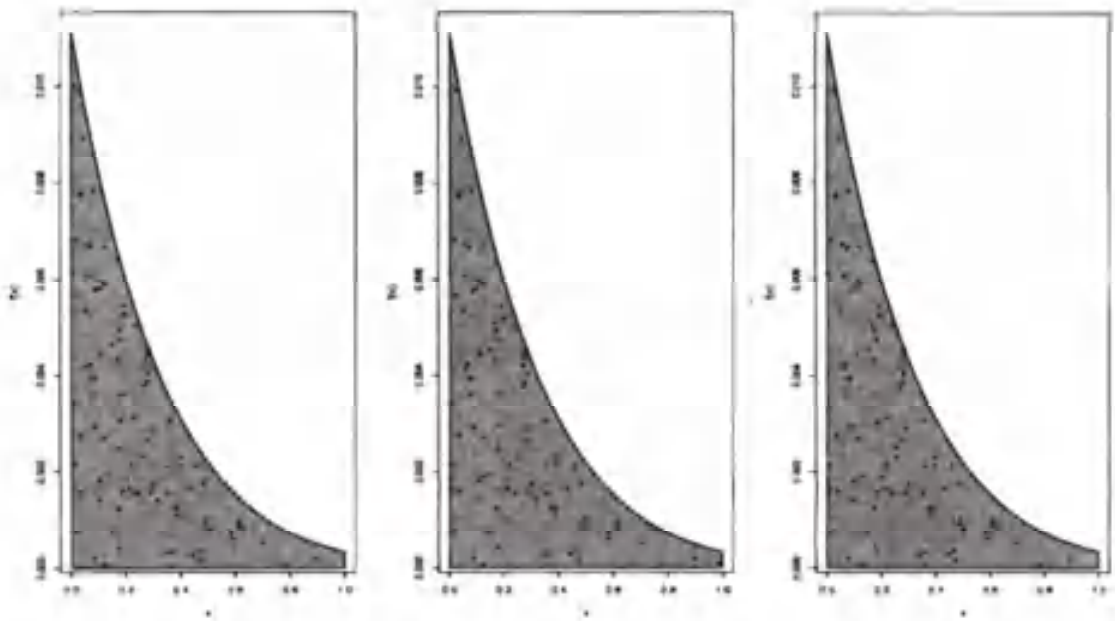


Fig. 8.3. Comparison of three samples obtained by ten iterations of the slice sampler starting from $(.01, .01)$ *(left)*, $(.99, .001)$ *(center)* and $(.25, .025)$ *(right)* and based on the same pool of uniform variates.

In this simple example, ten iterations are thus enough to cancel the effect of the starting value.

The major (practical) difficulty with the slice sampler is with the simulation of the uniform distribution $\mathcal{U}_{A(\omega)}$, since the determination of the set of y's such that $f_1(y) > \omega$ can be intractable if f_1 is complex enough. We thus consider an extension to [A.31] in the next section, but call attention to an extension by Neal (2003) covered in Note 8.5.1.

8.2 The General Slice Sampler

As indicated above, sampling uniformly from the slice $A^{(t)} = \{x; f_1(x) \geq \omega^{(t)}\}$ may be completely intractable, even with the extension of Note 8.5.1. This difficulty persists as the dimension of x gets larger. However, there exists a generalization of the 2D slice sampler [A.31] that partially alleviates this difficulty by introducing multiple slices.

This general slice sampler can be traced back to the auxiliary variable algorithm of Edwards and Sokal (1988) applied to the Ising model of Example 5.8 (see also Swendsen and Wang 1987, Wakefield et al. 1991, Besag and Green 1993, Damien and Walker 1996, Higdon 1996, Neal 1997, Damien et al. 1999, and Tierney and Mira 1999). It relies upon the decomposition of the density $f(x)$ as

$$f(x) \propto \prod_{i=1}^{k} f_i(x),$$

where the f_i's are positive functions, but *not necessarily densities*. For instance, in a Bayesian framework with a flat prior, the $f_i(x)$ may be chosen as the individual likelihoods.

This decomposition can then be associated with k auxiliary variables ω_i, rather than one as in the fundamental theorem, in the sense that each $f_i(x)$ can be written as an integral

$$f_i(x) = \int \mathbb{I}_{0 \leq \omega_i \leq f_i(x)} \, d\omega_i,$$

and that f is the marginal distribution of the joint distribution

$$(x, \omega_1, \ldots, \omega_k) \sim p(x, \omega_1, \ldots, \omega_k) \propto \prod_{i=1}^{k} \mathbb{I}_{0 < \omega_i < f_i(x)}. \tag{8.3}$$

This particular *demarginalization of f* (Section 5.3.1) introduces a larger dimensionality to the problem and induces a generalization of the random walk of Section 8.1 which is to have uniform proposals one direction at a time. The corresponding generalization of [A.31] is thus as follows.

Algorithm A.32 –Slice Sampler–

At iteration $t+1$, simulate

1. $\omega_1^{(t+1)} \sim \mathcal{U}_{[0, f_1(x^{(t)})]}$;

⋮ [A.32]

k. $\omega_k^{(t+1)} \sim \mathcal{U}_{[0, f_k(x^{(t)})]}$;

k+1. $x^{(t+1)} \sim \mathcal{U}_{A^{(t+1)}}$, **with**

$$A^{(t+1)} = \{y;\ f_i(y) \geq \omega_i^{(t+1)},\ i = 1, \ldots, k\}.$$

Example 8.3. A 3D slice sampler. Consider the density proportional to

$$(1+\sin^2(3x))\,(1+\cos^4(5x))\,\exp\{-x^2/2\}.$$

The corresponding functions are, for instance, $f_1(x) = (1+\sin^2(3x))$, $f_2(x) = (1+\cos^2(5x))$, and $f_3(x) = \exp\{-x^2/2\}$. In an iteration of the slice sampler, three uniform $\mathcal{U}([0,1])$ u_1, u_2, u_3 are generated and the new value of x is uniformly distributed over the set

$$\left\{x : |x| \le \sqrt{-2\log\omega_3}\right\} \cap \left\{x : \sin^2(3x) \ge 1-\omega_1\right\} \cap \left\{x : \cos^4(5x) \ge 1-\omega_2\right\},$$

which is made of one or several intervals depending on whether $\omega_1 = u_1 f_1(x)$ and $\omega_2 = u_2 f_2(x)$ are larger than 1. Figure 8.4 shows how the sample produced by the algorithm fits the target density. ‖

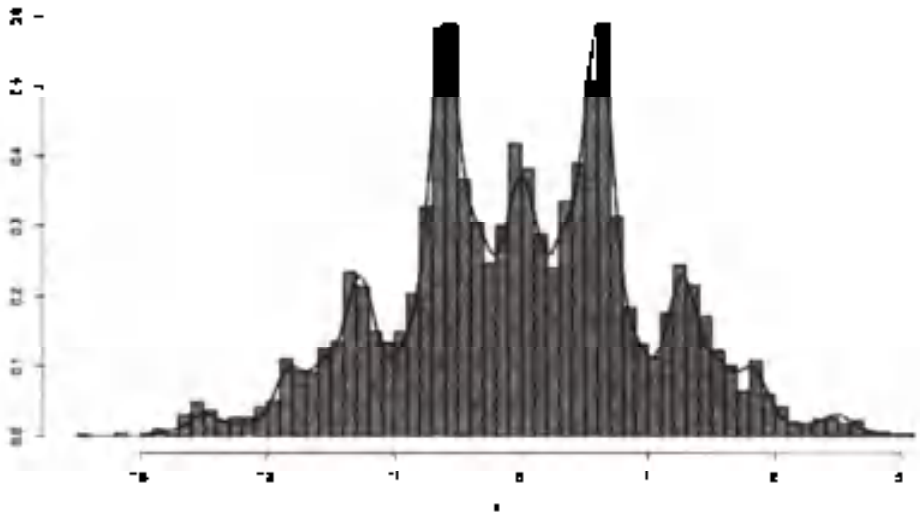


Fig. 8.4. Histogram of a sample produced by 5,000 iterations of a 3D slice sampler and superposition of the target density proportional to $(1+\sin^2(3x))\,(1+\cos^4(5x))\,\exp\{-x^2/2\}$.

In Section 5.3.1, the completion was justified by a *latent variable* representation of the model. Here, the auxiliary variables are more artificial, but there often is an obvious connection in (natural) latent variable models.

Example 8.4. Censored data models. As noted in Section 5.3.1, censored data models (Example 5.14) can be associated with a missing data structure. Consider

$$y^* = (y_1^*, y_2^*) = (y \wedge r, \mathbf{I}_{y<r}), \quad \text{with } y \sim f(y|\theta) \text{ and } r \sim h(r),$$

so the observation y is censored by the random variable r. If we observe $(y_1^*, \ldots, y_n^*)$, the density of $y_i^* = (y_{1i}^*, y_{2i}^*)$ is then

[illegible]

[illegible]

[illegible]

[illegible]

8.3 Convergence Properties of the Slice Sampler

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Lemma 8.5. [illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Lemma 8.6. [illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Lemma 8.7. [illegible]

[illegible]

[illegible]

[illegible]

Example 8.8. Exponential [illegible] distribution for [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 8.9. A poor slice sampler. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

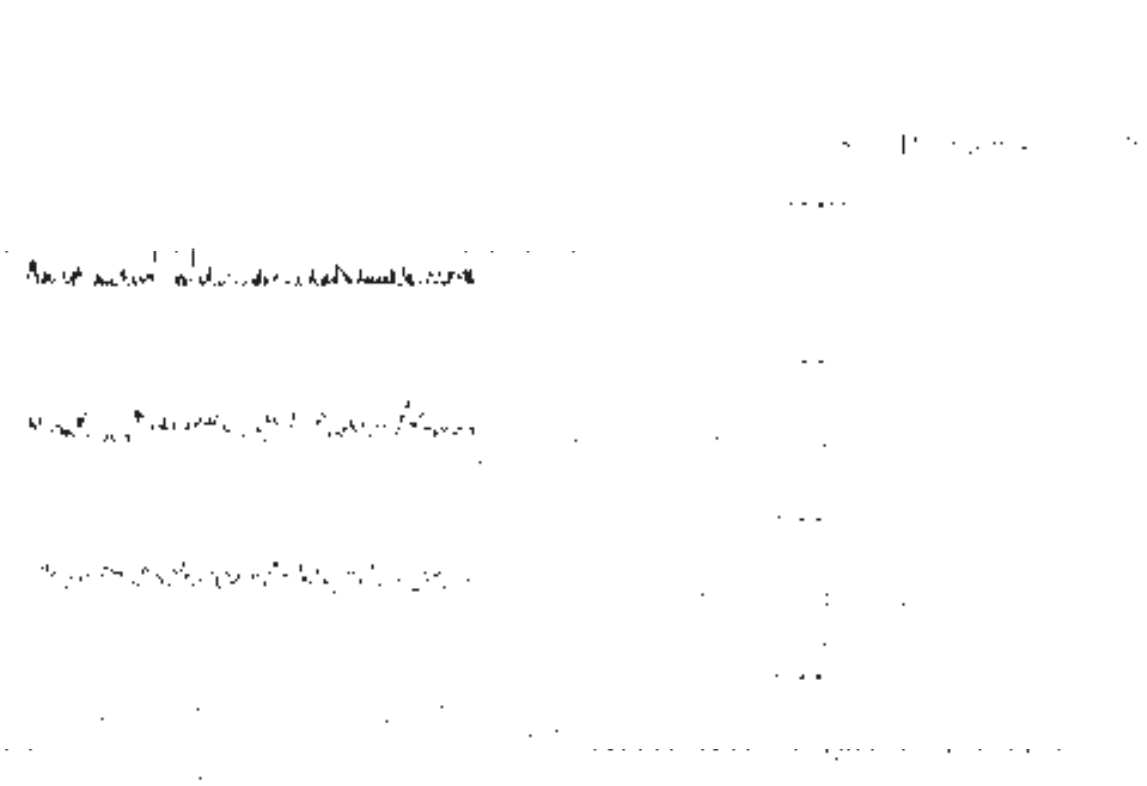


Fig. 8.5 [illegible]

8.4 Problems

8.1 [illegible]

8.2 [illegible]

8.3 [illegible]

8.4 [illegible]

8.5 [illegible]

8.6 [illegible]

[illegible]

8.7 [illegible]

8.8 [illegible]

[illegible]

8.9 [illegible]

8.10 [illegible]

8.11 [illegible]

[illegible]

8.12 [illegible]

[illegible]

[illegible]

8.13 [illegible]

8.14 [illegible]

8.15 [illegible]

8.16 [illegible]

8.17 [illegible]

8.18 [illegible]

8.19 [illegible]

8.20 [illegible]

8.5 Notes

8.5.1 Dealing with Difficult Sites

[illegible]

9

The Two-Stage Gibbs Sampler

[illegible]

[illegible]

9.1 A General Class of Two-Stage Algorithms

9.1.1 From Slice Sampling to Gibbs Sampling

[illegible]

[illegible]

[illegible]

[illegible]

U's gets somehow superfluous since we are really interested in the generation from $f(x,y)$, rather than from the uniform on $\mathcal{S}(f)$.

Although this is *not* the way it was originally derived, this introduction points out the strong link between the slice sampler and the two-stage Gibbs sampler, sometimes called *Data Augmentation* (Tanner and Wong 1987). It also stresses how the two-stage Gibbs sampler takes further advantage of the knowledge on the distribution f, compared with the slice sampler that uses only the numerical values of $f(x)$. This is reflected in the fact that each step of the two-stage Gibbs sampler amounts to an infinity of steps of a special slice sampler. (Note that this does not mean that the two-stage Gibbs sampler always does better than any slice sampler.)

9.1.2 Definition

The algorithmic implementation of the two-stage Gibbs sampler is thus straightforward. If the random variables X and Y have joint density $f(x,y)$, the two-stage Gibbs sampler generates a Markov chain (X_t, Y_t) according to the following steps:

Algorithm A.33 –Two-stage Gibbs sampler–

Take $X_0 = x_0$
For $t = 1, 2, \ldots$, generate

1. $Y_t \sim f_{Y|X}(\cdot|x_{t-1})$; [A.33]
2. $X_t \sim f_{X|Y}(\cdot|y_t)$.

where $f_{Y|X}$ and $f_{X|Y}$ are the conditional distributions associated with f, as in (9.1).

We note here that not only is the sequence (X_t, Y_t) a Markov chain, but also each subsequence (X_t) and (Y_t) is a Markov chain. For example, the chain (X_t) has transition density

$$K(x,x^*) = \int f_{Y|X}(y|x) f_{X|Y}(x^*|y)dy,$$

which indeed depends on the past only through the last value of (X_t). (Note the similarity to Eaton's transition (6.46).) In addition, it is also easy to show that f_X is the stationary distribution associated with this (sub)chain, since

$$\begin{aligned} f_X(x') &= \int f_{X|Y}(x'|y) f_Y(y)dy \\ &= \int f_{X|Y}(x'|y) \int f_{Y|X}(y|x) f_X(x)dxdy \qquad (9.2) \\ &= \int \left[\int f_{X|Y}(x'|y) f_{Y|X}(y|x)dy\right] f_X(x)dx\,. \\ &= \int K(x,x') f_X(x)dx\,. \end{aligned}$$

[illegible]

Example 9.[illegible]. Normal univariate Gibbs [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 9.[illegible]. Gibbs sampling for mixture gaussians. [illegible]

$$\sum_{j=1}^{k} p_j \, f(x|\xi_j) \,, \tag{9.5}$$

where $f(\cdot|\xi)$ belongs to an exponential family

$$f(x|\xi) = h(x) \, \exp\{\xi \cdot x - \psi(\xi)\}$$

and ξ is distributed from the associated conjugate prior

$$\pi(\xi|\alpha_0, \lambda) \propto \exp\{\lambda(\xi \cdot \alpha_0 - \psi(\xi))\} \,, \qquad \lambda > 0, \quad \alpha_0 \in \mathcal{X} \,,$$

while

$$(p_1, \ldots, p_k) \sim \mathcal{D}_k(\gamma_1, \ldots, \gamma_k) \,.$$

Given a sample $(x_1, \ldots, x_n)$ from (9.5), we can associate with every observation an indicator variable $z_i \in \{1, \ldots, k\}$ that indicates which component of the mixture is associated with x_i (see Problems 5.8–5.10). The demarginalization (or *completion*) of model (9.5) is then

$$Z_i \sim \mathcal{M}_k(1; p_1, \ldots, p_k), \qquad x_i|z_i \sim f(x|\xi_{z_i}) \,.$$

Thus, considering $x_i^* = (x_i, z_i)$ (instead of x_i) entirely eliminates the mixture structure since the likelihood of the completed model is

$$\begin{aligned} \ell(p, \xi|x_1^*, \ldots, x_n^*) &\propto \prod_{i=1}^{n} p_{z_i} \, f(x_i|\xi_{z_i}) \\ &= \prod_{j=1}^{k} \prod_{i;z_i=j} p_j \, f(x_i|\xi_j) \,. \end{aligned}$$

(This latent structure is also exploited in the original implementation of the EM algorithm; see Section 5.3.2.) The two steps of the Gibbs sampler are

Algorithm A.34 –Mixture Posterior Simulation–

1. Simulate Z_i $(i = 1, \ldots, n)$ from

$$P(Z_i = j) \propto p_j \, f(x_i|\xi_j) \qquad (j = 1, \ldots, k)$$

and compute the statistics [A.34]

$$n_j = \sum_{i=1}^{n} \mathbb{I}_{z_i=j} \,, \qquad n_j \overline{x}_j = \sum_{i=1}^{n} \mathbb{I}_{z_i=j} x_i \,.$$

2. Generate $(j = 1, \ldots, k)$

$$\xi \sim \pi\left(\xi \middle| \frac{\lambda_j \alpha_j + n_j \overline{x}_j}{\lambda_j + n_j}, \lambda_j + n_j\right),$$

$$p \sim \mathcal{D}_k(\gamma_1 + n_1, \ldots, \gamma_k + n_k) \,.$$

[illegible]

[illegible]

[illegible]

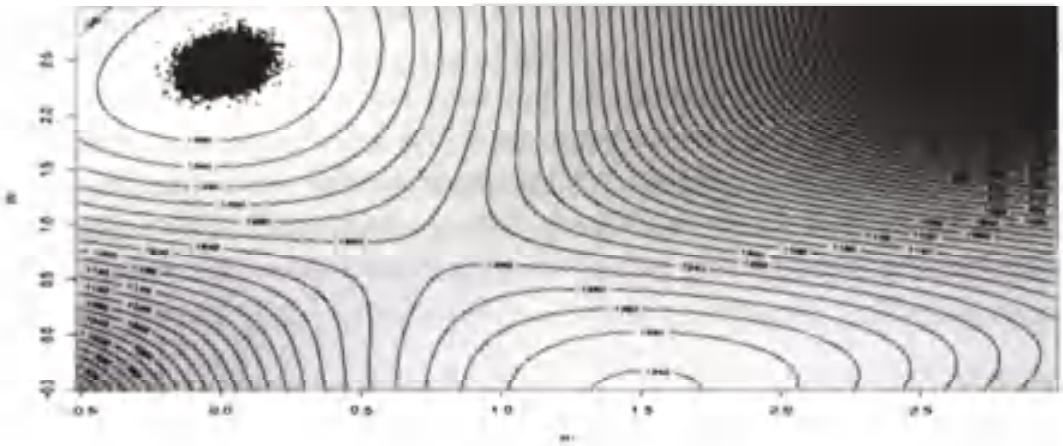


Fig. 9.1. *Gibbs sample of 5,000 points for the mixture posterior against the posterior surface.*

9.1.3 Back to the Slice Sampler

Just as the two-stage Gibbs sampler can be seen as a limiting case of the slice sampler in a three-coordinate problem, the slice sampler can be interpreted as a special case of two-stage Gibbs sampler when the joint distribution is the uniform distribution on the subgraph $\mathcal{S}(f)$. From this point of view, the slice sampler starts with $f_X(x)$ and creates a joint density $f(x,u) = \mathbb{I}(0 < u < f_X(x))$. The associated conditional densities are

$$f_{X|U}(x|u) = \frac{\mathbb{I}(0 < u < f_X(x))}{\int \mathbb{I}(0 < u < f_X(x))dx} \text{ and } f_{U|X}(u|x) = \frac{\mathbb{I}(0 < u < f_X(x))}{\int \mathbb{I}(0 < u < f_X(x))du},$$

which are exactly those used in the slice sampler. Therefore, the X sequence is also a Markov chain with transition kernel

$$K(x,x') = \int f_{X|U}(x'|u) f_{U|X}(u|x)du$$

and stationary density $f_X(x)$.

What the slice sampler tells us is that we can induce a Gibbs sampler for any marginal distribution $f_X(x)$ by creating a joint distribution that is, formally, arbitrary. Starting from $f_X(x)$, we can take any conditional density $g(y|x)$ and create a Gibbs sampler with

$$f_{X|Y}(x|y) = \frac{g(y|x)f_X(x)}{\int g(y|x)f_X(x)dx} \quad \text{and} \quad f_{Y|X}(y|x) = \frac{g(y|x)f_X(x)}{\int g(y|x)f_X(x)dy}.$$

9.1.4 The Hammersley–Clifford Theorem

A most surprising feature of the Gibbs sampler is that the conditional distributions contain sufficient information to produce a sample from the joint distribution. (This is the case for both two-stage and multi-stage Gibbs; see also

[illegible]

[illegible]

Theorem 9.3. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

9.2 Fundamental Properties

[illegible]

9.2. Probabilistic Structures

[illegible]

Definition 9.4 [illegible]

[illegible]

Lemma 9.5 [illegible]

$$[illegible]$$

[illegible]

[illegible]

Theorem 9.6 [illegible]

$$[illegible]$$

[illegible]

[illegible]

Example 9.7. Grouped counting data. For 360 consecutive time units, consider recording the number of passages of individuals, per unit time, past some sensor. This can be, for instance, the number of cars observed at a crossroad or the number of leucocytes in a region of a blood sample. Hypothetical results are given in Table 9.1. This table, therefore, involves a grouping of the

Number of passages	0	1	2	3	4 or more
Number of observations	139	128	55	25	13

Table 9.1. Frequencies of passage for 360 consecutive observations.

observations with four passages and more. If we assume that every observation is a Poisson $\mathcal{P}(\lambda)$ random variable, the likelihood of the model corresponding to Table 9.1 is

$$\ell(\lambda|x_1,\ldots,x_5) \propto e^{-347\lambda}\lambda^{128+55\times 2+25\times 3}\left(1-e^{-\lambda}\sum_{i=0}^{3}\frac{\lambda^i}{i!}\right)^{13},$$

for $x_1 = 139,\ldots,x_5 = 13$.

For $\pi(\lambda) = 1/\lambda$ and $\mathbf{y} = (y_1,\ldots,y_{13})$, vector of the 13 units larger than 4, it is possible to complete the posterior distribution $\pi(\lambda|x_1,\ldots,x_5)$ in $\pi(\lambda,y_1,\ldots,y_{13}|x_1,\ldots,x_5)$ and to propose the Gibbs sampling algorithm associated with the two components λ and $\mathbf{y}$.

Algorithm A.35 –Poisson–Gamma Gibbs Sampler–

Given $\lambda^{(t-1)}$,

1. Simulate $Y_i^{(t)} \sim \mathcal{P}(\lambda^{(t-1)})\,\mathbf{I}_{y\geq 4}$ $(i=1,\ldots,13)$
2. Simulate [A.35]

$$\lambda^{(t)} \sim \mathcal{G}a\left(313+\sum_{i=1}^{13} y_i^{(t)},\ 360\right).$$

Figure 9.2 describes the convergence of the Rao–Blackwellized estimator (see Section 4.2 for motivation and Sections 9.3 and 10.4.2 for justification)

$$\begin{aligned}\delta_{rb} &= \frac{1}{T}\sum_{t=1}^{T}\mathbf{E}\left[\lambda|x_1,\ldots,x_5,y_1^{(t)},\ldots,y_{13}^{(t)}\right]\\ &= \frac{1}{360T}\sum_{t=1}^{T}\left(313+\sum_{i=1}^{13}y_i^{(t)}\right),\end{aligned}$$

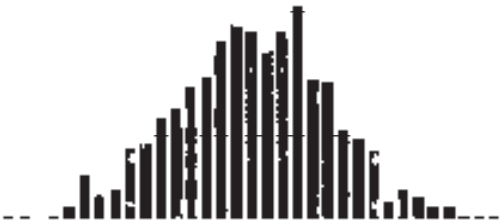


Fig. 9.2. [illegible]

[illegible]

Example 9.5. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 9.9. Capture recapture uniform model. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

9.2.2 Reversible and Irreducibility Chains

[illegible]

Definition 9.10. [illegible]

[illegible]

Lemma 9.11. [illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

Algorithm A.36 –Reversible Two-Stage Gibbs–

Given $y_2^{(t)}$,

1. Simulate $W \sim g_1(w|y_2^{(t)})$;
2. Simulate $Y_2^{(t+1)} \sim g_2(y_2|w)$; [A.36]
3. Simulate $Y_1^{(t+1)} \sim g_1(y_1|y_2^{(t+1)})$.

In this case,

$$\begin{aligned}(Y_1, Y_2, Y_1', Y_2') &\sim g(y_1, y_2) \int g_1(w|y_2)\, g_2(y_2'|w)\, dw\; g_1(y_1'|y_2') \\ &= g(y_1', y_2') \int \frac{g_2(y_2'|w)}{g^1(y_2')}\, g(w, y_2)\, dw\; g_1(y_1|y_2) \\ &\quad g(y_1', y_2') \int g_1(w|y_2')\, g_2(y_2|w)\, dw\; g_1(y_1|y_2)\end{aligned}$$

is distributed as (Y_1', Y_2', Y_1, Y_2).

9.2.3 The Duality Principle

The well-known concept of Rao Blackwellization (Sections 4.2, 7.6.2, and to come in Section 9.3) is based on the fact that a conditioning argument, using variables other than those directly of interest, can result in an improved procedure. In the case of interleaving Markov chains, this phenomenon goes deeper. We will see in this section that this phenomenon is more fundamental than a mere improvement of the variance of some estimators, as it provides a general technique to establish convergence properties for the chain of interest $(X^{(t)})$ based on the instrumental chain $(Y^{(t)})$, even when the latter is unrelated with the inferential problem. Diebolt and Robert (1993, 1994) have called this use of the dual chain the *Duality Principle* when they used it in the setup of mixtures of distributions. While it has been introduced in a two-stage Gibbs sampling setup, this principle extends to other Gibbs sampling methods since it is sufficient that the chain of interest $(X^{(t)})$ be generated conditionally on another chain $(Y^{(t)})$, which supplies the probabilistic properties.

Theorem 9.12. *Consider a Markov chain $(X^{(t)})$ and a sequence $(Y^{(t)})$ of random variables generated from the conditional distributions*

$$X^{(t)}|y^{(t)} \sim \pi(x|y^{(t)}), \qquad Y^{(t+1)}|x^{(t)}, y^{(t)} \sim f(y|x^{(t)}, y^{(t)}).$$

If the chain $(Y^{(t)})$ is ergodic (geometrically or uniformly ergodic) and if $\pi^t_{y^{(0)}}$ denotes the distribution of $(X^{(t)})$ associated with the initial value $y^{(0)}$, the norm $\|\pi^t_{y^{(0)}} - \pi\|_{TV}$ goes to 0 when t goes to infinity (goes to 0 at a geometric or uniformly bounded rate).

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible] \int [illegible] \quad [illegible]$$

[illegible]

$$[illegible] \quad [illegible] \sum [illegible] \quad [illegible]$$

[illegible]

$$\sum [illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Corollary 9.14. [illegible]

Example 9.15. (Continuation of Example 9.8) [illegible]

[illegible]

Proposition 9.16. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

$$\sum [illegible]$$

$$\sum [illegible]$$

$$\sum [illegible]$$

$$\sum [illegible]$$

[illegible]

[illegible]

11.3 Monotone Covariance and Rao–Blackwellization

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Lemma 9.17. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Proposition 9.18. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 9.19 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 9.20. (Continuation of Example 9.[illegible]) [illegible]

$$[illegible]$$

[illegible]

9.4 The EM Gibbs-Connection:

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 9.21 Censored data (Wolfe [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

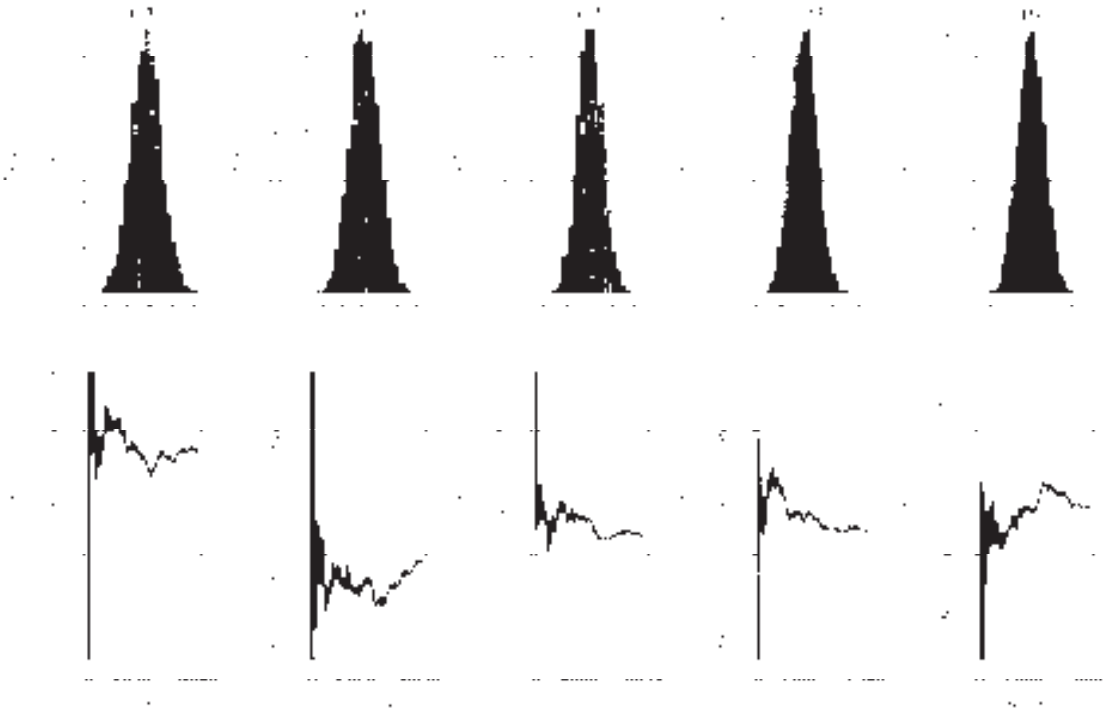


Fig. 9.6 [illegible]

[illegible]

[illegible]

Example 9.22. Cellular phone GSM. [illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

[illegible]

[illegible]

9.5 Transition

[illegible]

9.6 Problems

9.1 [illegible]

9.2 [illegible]

9.3 [illegible]

9.4 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

9.14 [illegible]

[illegible]

9.15 [illegible]

[illegible]

9.16 [illegible]

[illegible]

9.17 [illegible]

[illegible]

9.18 [illegible]

[illegible]

9.19 In the setup of Example 9.23, show that the likelihood function is not bounded and deduce that, formally, there is no maximum likelihood estimator. (*Hint:* Take $\mu_{j_0} = x_{i_1}$ and let τ_{j_0} go to 0.)

9.20 A model consists in the partial observation of normal vectors $Z = (X, Y) \sim \mathcal{N}_2(0, \Sigma)$ according to a mechanism of random censoring. The corresponding data are given in Table 9.2.

x	1.17	−0.98	0.18	0.57	0.21	−	−	−
y	0.34	−1.24	−0.13	−	−	−0.12	−0.83	1.64

Table 9.2. Independent observations of $Z = (X, Y) \sim \mathcal{N}_2(0, \Sigma)$ with missing data (denoted −).

(a) Show that inference can formally be based on the likelihood

$$\prod_{i=1}^{3} \left\{ |\Sigma|^{-1/2} e^{-z_i^t \Sigma^{-1} z_i/2} \right\} \sigma_1^{-2} e^{-(x_4^2+x_5^2)/2\sigma_1^2} \sigma_2^{-3} e^{-(y_6^2+y_7^2+y_8^2)/2\sigma_2^2}.$$

(b) Show that the choice of the prior distribution $\pi(\Sigma) \propto |\Sigma|^{-4}$ leads to difficulties given that σ_1 and σ_2 are isolated in the likelihood

(c) Show that the missing components can be simulated through the following algorithm.

Algorithm A.37 –Normal Completion–

1. Simulate

$$X_i^* \sim \mathcal{N}\left(\rho \frac{\sigma_1}{\sigma_2} y_i, \sigma_1^2(1-\rho^2)\right) \qquad (i = 6, 7, 8),$$

$$Y_i^* \sim \mathcal{N}\left(\rho \frac{\sigma_2}{\sigma_1} x_i, \sigma_2^2(1-\rho^2)\right) \qquad (I = 4, 5).$$

2. Generate

$$\Sigma^{-1} \sim \mathcal{W}_2(8, X^{-1}),$$

with $X = \sum_{i=1}^{8} z_i^* z_i^{*t}$, the dispersion matrix of the completed data.

to derive the posterior distribution of the quantity of interest, ρ.

(d) Propose a Metropolis–Hastings alternative based on a slice sampler.

9.21 Roberts and Rosenthal (2003) derive the *polar slice sampler* from the decomposition $\pi(x) \propto f_0(x) f_1(x)$ of a target distribution $\pi(x)$.

(a) Show that the Gibbs sampler whose two steps are (i) to simulate U uniformly on $(0, f_1(x)$ and (ii) to simulate X from $f_0(x)$ restricted to the set of x's such that $f_1(x) > u$ is a valid MCMC algorithm with target distribution π.

(b) Show that, if f_0 is constant, this algorithm is simply the slice sampler of Algorithm [A.31] in Chapter 8. (It is also called the *uniform slice sampler* in Roberts and Rosenthal 2003.)

(c) When $x \in \mathbb{R}^d$, using the case $f_0(x) = |x|^{-(d-1)}$ and $f_1(x) = |x|^{d-1}\pi(x)$ is called the *polar slice sampler*. Show that using the polar slice sampler on a d-dimensional log-concave target density is equivalent to a uniform slice sampler on a corresponding one-dimensional log-concave density.

(d) Illustrate this property in the case when $\pi(x) = \exp -|x|$.

9.22 Consider the case of a mixture of normal distributions,

$$f(x) = \sum_{j=1}^{k} p_j \frac{e^{-(x-\mu_j)^2/(2\tau_j^2)}}{\sqrt{2\pi}\,\tau_j},$$

(a) Show that the conjugate distribution on (μ_j, τ_j) is

$$\mu_j|\tau_j \sim \mathcal{N}\left(\alpha_j, \tau_j^2/\lambda_j\right), \qquad \tau_j^2 \sim \mathcal{IG}\left(\frac{\lambda_j+3}{2}, \frac{\beta_j}{2}\right).$$

(b) Show that two valid steps of the Gibbs sampler are as follows.

Algorithm A.38 –Normal Mixture Posterior Simulation–

1. Simulate $(i = 1, \ldots, n)$

$$Z_i \sim P(Z_i = j) \propto p_j \exp\left\{-(x_i-\mu_j)^2/(2\tau_j^2)\right\} \tau_j^{-1}$$

and compute the statistics $(j = 1, \ldots, k)$ [A.38]

$$n_j = \sum_{i=1}^{n} \mathbb{I}_{z_i=j}, \quad n_j\overline{x}_j = \sum_{i=1}^{n} \mathbb{I}_{z_i=j}x_i, \quad s_j^2 = \sum_{i=1}^{n} \mathbb{I}_{z_i=j}(x_i-\overline{x}_j)^2.$$

2. Generate

$$\mu_j|\tau_j \sim \mathcal{N}\left(\frac{\lambda_j\alpha_j + n_j\overline{x}_j}{\lambda_j+n_j}, \frac{\tau_j^2}{\lambda_j+n_j}\right),$$

$$\tau_j^2 \sim \mathcal{IG}\left(\frac{\lambda_j+n_j+3}{2}, \frac{\lambda_j\alpha_j^2+\beta_j+s_j^2-(\lambda_j+n_j)^{-1}(\lambda_j\alpha_j+n_j\overline{x}_j)^2}{2}\right),$$

$$p \sim \mathcal{D}_k(\gamma_1+n_1, \ldots, \gamma_k+n_k).$$

(*Note:* Robert and Soubiran 1993 use this algorithm to derive the maximum likelihood estimators by recursive integration (see Section 5.2.4), showing that the Bayes estimators converge to the local maximum, which is the closest to the initial Bayes estimator.)

9.23 Referring to Section 9.7.2, for the factor ARCH model of (9.18):

(a) Propose a noninformative prior distribution on the parameter $\theta = (\alpha, \beta, a, \Sigma)$ that leads to a proper posterior distribution.

(b) Propose a completion step for the latent variables based on $f(y_t^*|y_t, y_{t-1}^*, \theta)$.

(*Note:* See Diebold and Nerlove 1989, Gouriéroux et al. 1993, Kim et al. 1998, and Billio et al. 1998 for different estimation approaches to this model.)

9.24 Check whether a negative coefficient b in the random walk $Y_t = a + b(X^{(t)} - a) + Z_t$ induces a negative correlation between the $X^{(t)}$'s. Extend to the case where the random walk has an ARCH structure,

$$Y_t = a + b(X^{(t)} - a) + \exp(c + d(X^{(t)} - a)^2)Z_t.$$

9.25 [illegible]

9.26 [illegible]

9.27 [illegible]

9.28 [illegible]

9.7 Notes

9.7.1 Inference for Mixtures

[illegible]

[illegible]

[illegible]

Example 9.23 (Continuation of Example 9.2) [illegible]

[illegible]

9.7.2 ARCH Models

[illegible]

[illegible]

[illegible]

[illegible]

10

The Multi-Stage Gibbs Sampler

[illegible]

10.1 Basic Derivations

10.1.1 Definition

[illegible]

the corresponding univariate conditional densities $f_1, \ldots, f_p$, that is, we can simulate

$$X_i|x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p \sim f_i(x_i|x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p)$$

for $i = 1, 2, \ldots, p$. The associated *Gibbs sampling* algorithm (or *Gibbs sampler*) is given by the following transition from $X^{(t)}$ to $X^{(t+1)}$:

Algorithm A.39 –The Gibbs Sampler–

Given $\mathbf{x}^{(t)} = (x_1^{(t)}, \ldots, x_p^{(t)})$, **generate**

1. $X_1^{(t+1)} \sim f_1(x_1|x_2^{(t)}, \ldots, x_p^{(t)})$;
2. $X_2^{(t+1)} \sim f_2(x_2|x_1^{(t+1)}, x_3^{(t)}, \ldots, x_p^{(t)})$, [A.39]

$\vdots$

p. $X_p^{(t+1)} \sim f_p(x_p|x_1^{(t+1)}, \ldots, x_{p-1}^{(t+1)})$.

The densities $f_1, \ldots, f_p$ are called the *full conditionals*, and a particular feature of the Gibbs sampler is that these are the only densities used for simulation. Thus, even in a high-dimensional problem, *all of the simulations may be univariate*, which is usually an advantage.

Example 10.1. Autoexponential model. The *autoexponential model* of Besag (1974) has been found useful in some aspects of spatial modeling. When $y \in \mathbb{R}_+^3$, the corresponding density is

$$f(y_1, y_2, y_3) \propto \exp\{-(y_1 + y_2 + y_3 + \theta_{12}y_1y_2 + \theta_{23}y_2y_3 + \theta_{31}y_3y_1)\},$$

with known $\theta_{ij} > 0$. The full conditional densities are exponential. For example,

$$Y_3|y_1, y_2 \sim \mathcal{E}xp(1 + \theta_{23}y_2 + \theta_{31}y_1).$$

They are thus very easy to simulate from. In contrast, the other conditionals and the marginal distributions have forms such as

$$f(y_2|y_1) \propto \frac{\exp\{-(y_1 + y_2 + \theta_{12}y_1y_2)\}}{1 + \theta_{23}y_2 + \theta_{31}y_1},$$

$$f(y_1) \propto e^{-y_1}\int_0^{+\infty} \frac{\exp\{-y_2 - \theta_{12}y_1y_2\}}{1 + \theta_{23}y_2 + \theta_{31}y_1}\, dy_2,$$

which cannot be simulated easily. ||

Example 10.2. Integrations. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

Definition 10.3. Given a probability density f, a density g that satisfies

$$\int_{\mathcal{Z}} g(x, z)\, dz = f(x)$$

is called a *completion* of f.

The density g is chosen so that the full conditionals of g are easy to simulate from and the Gibbs algorithm [A.39] is implemented on g instead of f. For $p > 1$, write $y = (x, z)$ and denote the conditional densities of $g(y) = g(y_1, \ldots, y_p)$ by

$$\begin{aligned}
&Y_1|y_2, \ldots, y_p \sim g_1(y_1|y_2, \ldots, y_p),\\
&Y_2|y_1, y_3, \ldots, y_p \sim g_2(y_2|y_1, y_3, \ldots, y_p),\\
&\vdots\\
&Y_p|y_1, \ldots, y_{p-1} \sim g_p(y_p|y_1, \ldots, y_{p-1}).
\end{aligned}$$

The move from $Y^{(t)}$ to $Y^{(t+1)}$ is then defined as follows.

Algorithm A.40 –Completion Gibbs Sampler–

Given $(y_1^{(t)}, \ldots, y_p^{(t)})$, simulate

1. $Y_1^{(t+1)} \sim g_1(y_1|y_2^{(t)}, \ldots, y_p^{(t)})$,
2. $Y_2^{(t+1)} \sim g_2(y_2|y_1^{(t+1)}, y_3^{(t)}, \ldots, y_p^{(t)})$, [A.40]

   $\vdots$

p. $Y_p^{(t+1)} \sim g_p(y_p|y_1^{(t+1)}, \ldots, y_{p-1}^{(t+1)})$.

The two-stage Gibbs sampler of Chapter 9 obviously corresponds to the particular case of [A.40], where f is completed in g and x in $y = (y_1, y_2)$ such that both conditional distributions $g_1(y_1|y_2)$ and $g_2(y_2|y_1)$ are available. (Again, note that both y_1 and y_2 can be either scalars or vectors.)

In cases when such completions seem necessary (for instance, when every conditional distribution associated with f is not explicit, or when f is unidimensional), there exist an infinite number of densities for which f is a marginal density. We will not discuss this choice in terms of optimality, because, first, there are few practical results on this topic (as it is similar to finding an optimal density g in the Metropolis–Hastings algorithm) and, second, because there exists, in general, a *natural* completion of f in g. As already pointed out in Chapter 9, *missing data models* (Section 5.3.1) provide a series of examples of natural completion.

In principle, the Gibbs sampler does not require that the completion of f into g and of x in $y = (x, z)$ should be related to the problem of interest. Indeed, there are settings where the vector z has no meaning from a statistical point of view and is only a useful device, that is, an *auxiliary variable* as in (2.8).

Example 10.4. Cauchy–Normal posterior distribution. [illegible]

[illegible]

[illegible]

[

Algorithm A.41 –Reversible Gibbs Sampler–

Given $(y_2^{(t)}, \ldots, y_p^{(t)})$, generate

1. $Y_1^* \sim g_1(y_1|y_2^{(t)}, \ldots, y_p^{(t)})$
2. $Y_2^* \sim g_2(y_2|y_1^*, y_3^{(t)}, \ldots, y_p^{(t)})$

$\vdots$

p-1. $Y_{p-1}^* \sim g_{p-1}(y_{p-1}|y_1^*, \ldots, y_{p-2}^*, y_p^{(t)})$

p. $Y_p^{(t+1)} \sim g_p(y_p|y_1^*, \ldots, y_{p-1}^*)$ [A.41]

p+1. $Y_{p-1}^{(t+1)} \sim g_{p-1}(y_{p-1}|y_1^*, \ldots, y_{p-2}^*, y_p^{(t+1)})$

$\vdots$

2p-1. $Y_1^{(t+1)} \sim g_1(y_1|y_2^{(t+1)}, \ldots, y_p^{(t+1)})$

An[1] alternative to [A.41] has been proposed by Liu et al. (1995) and is called Gibbs sampling with *random scan*, as the simulation of the components of y is done in a random order following each transition. For the setup of [A.40], the modification [A.42] produces a reversible chain with stationary distribution g, and every simulated value is used.

Algorithm A.42 –Random Sweep Gibbs Sampler–

1. Generate a permutation $\sigma \in \mathcal{G}_p$;
2. Simulate $Y_{\sigma_1}^{(t+1)} \sim g_{\sigma_1}(y_{\sigma_1}|y_j^{(t)}, j \neq \sigma_1)$; [A.42]

$\vdots$

p+1. Simulate $Y_{\sigma_p}^{(t+1)} \sim g_{\sigma_p}(y_{\sigma_p}|y_j^{(t+1)}, j \neq \sigma_p)$.

This algorithm improves upon [A.41], which only uses one simulation out of two.[2] (Recall that, following Theorem 6.65, the reversibility of $(Y^{(t)})$ allows application of the Central Limit Theorem.)

10.1.3 The General Hammersley–Clifford Theorem

We have already seen the Hammersley–Clifford Theorem in the special case of the two-stage Gibbs sampler (Section 9.1.4). The theorem also holds in the general case and follows from similar manipulations of the conditional distri-

[1] *Step p is only called once, since repeating it would result in a complete waste of time!*

[2] Obviously, nothing prevents the use of all the simulations in integral approximations.

[illegible]

[illegible]

Theorem 12.5. Hamiltonian by Clifford. [illegible]

$$[illegible] \quad \coprod [illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible] \quad \coprod \frac{[illegible]}{[illegible]} [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

10.2 Theoretical Justifications

10.2.1 Markov Properties of the Gibbs Sampler

[illegible]

Theorem 10.6. [illegible]

Proof. [illegible]

[illegible]

$$[illegible] = \int [illegible]$$

$$= \int [illegible]$$

$$[illegible]$$

$$= \int [illegible]$$

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 2.5.7. Non-connected support. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Theorem 2.5.8. [illegible]

[illegible]

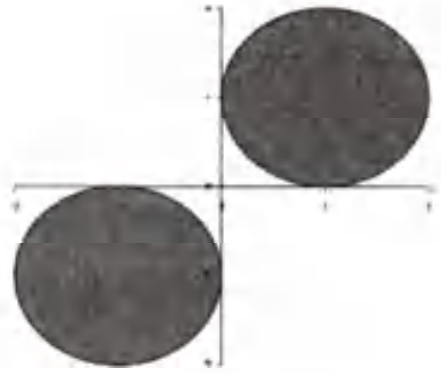


Fig. 10.1. Support of the function $f(x_1, x_2)$ of (10.3)

Lemma 10.9. *If the transition kernel associated with* [A.40] *is absolutely continuous with respect to the dominating measure, the resulting chain is Harris recurrent.*

The proof of Lemma 10.9 is similar to that of Lemma 7.3, where Harris recurrence was shown to follow from irreducibility. Here, the condition on the Gibbs transition kernel yields an irreducible chain, and Harris recurrence follows. Once the irreducibility of the chain associated with [A.40] is established, more advanced convergence properties can be established.

This condition of absolute continuity on the kernel (10.2) is satisfied by most decompositions. However, if one of the i steps $(1 \leq i \leq p)$ is replaced by a simulation from an Metropolis–Hastings algorithm, as in the hybrid algorithms described in Section 10.3, absolute continuity is lost and it will be necessary to either study the recursion properties of the chain or to introduce an additional simulation step to guarantee Harris recurrence.

From the property of Harris recurrence, we can now establish a result similar to Theorem 7.4.

Theorem 10.10. *If the transition kernel of the Gibbs chain* $(Y^{(t)})$ *is absolutely continuous with respect to the measure* μ,

(i) If $h_1, h_2 \in L^1(g)$ *with* $\int h_2(y)dg(y) \neq 0$, *then*

$$\lim_{n\to\infty} \frac{\sum_{t=1}^{T} h_1(Y^{(t)})}{\sum_{t=1}^{T} h_2(Y^{(t)})} - \frac{\int h_1(y)dg(y)}{\int h_2(y)dg(y)} \quad a.e.\ g.$$

(ii) If, in addition, $(Y^{(t)})$ *is aperiodic, then, for every initial distribution* μ,

$$\lim_{n\to\infty} \left\| \int K^{m}(y, \ )\mu(dx) - g \right\|_{TV} = 0 .$$

[illegible]

[illegible]

Lemma 10.11. [illegible]

$$[illegible]$$

[illegible]

[illegible]

Corollary 10.12. [illegible]

10.2.2 Gibbs Sampling as Metropolis–Hastings

[illegible]

Theorem 10.13. [illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

Example [illegible] (Continuation of Example [illegible]) [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

10.2.3 Hierarchical Structures

[illegible]

[illegible]

Example 10.17. Animal epidemiology. [illegible]

[illegible]

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

$$[illegible]$$

[illegible]

Figure 10.2 shows selected estimates of λ_i and β_i. For more details see Problem 10.10 or Eberly and Casella (2003). ||

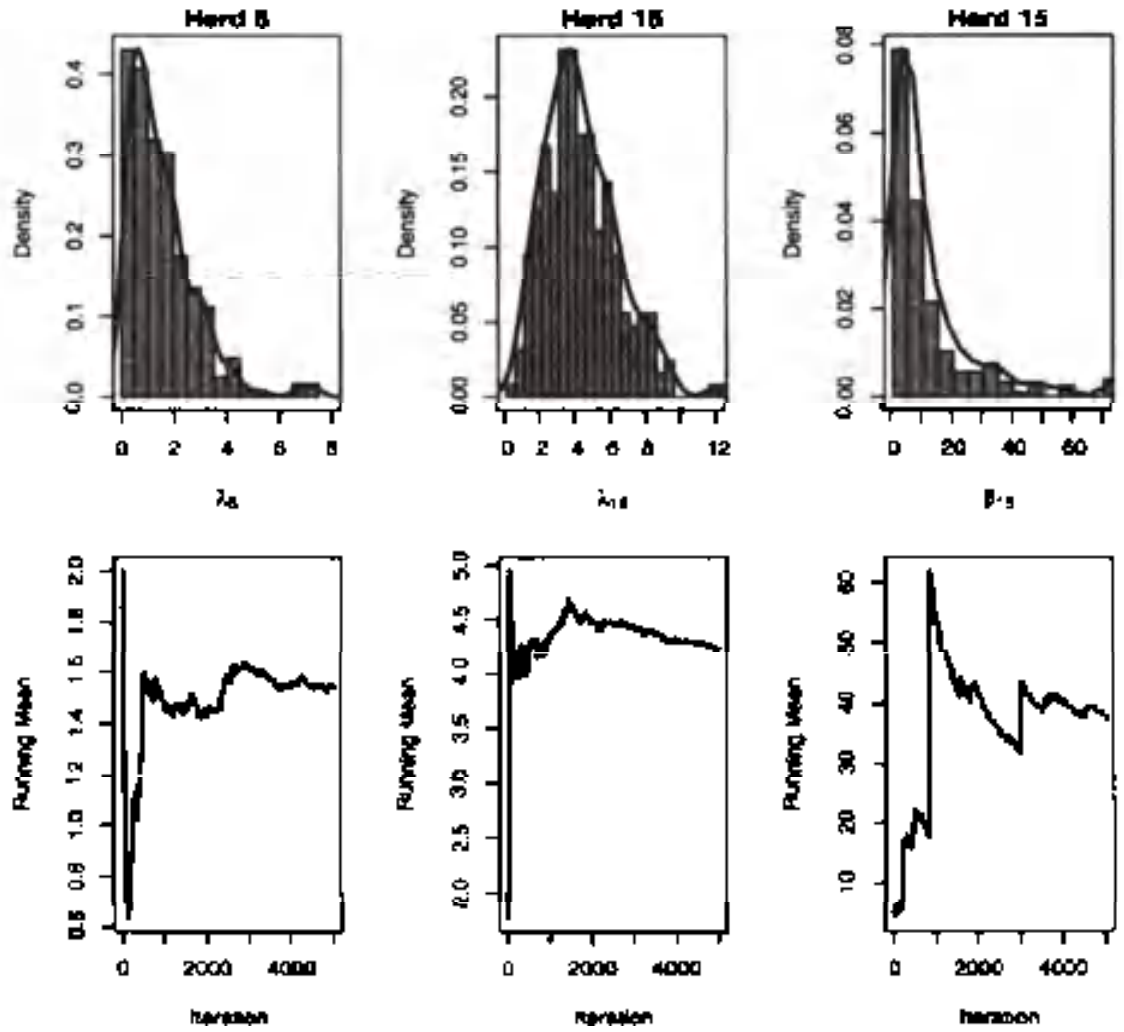


Fig. 10.2. Histograms, density estimates and running mean estimates for three selected parameters of the mastitis data (Example 10.15).

Example 10.16. Medical models. Part of the concern of the study of *pharmacokinetics* is the modeling of the relationship between the dosage of a drug and the resulting concentration in the blood. (More generally, pharmacokinetics studies the different interactions of a drug and the body.) Gilks et al. (1993) introduce an approach for estimating pharmacokinetic parameters that uses the traditional mixed-effects model and nonlinear structure, but which is also robust to the outliers common to clinical trials. For a given dose d_i administered at time 0 to patient i, the measured log concentration in the blood at time t_{ij}, X_{ij}, is assumed to follow a Student's t distribution

$$\frac{X_{ij} - \log g_{ij}(\lambda_i)}{\sigma\sqrt{n/(n-2)}} \sim \mathcal{T}_n,$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

10.3 Hybrid Gibbs Samplers

10.3.1 Comparison with Metropolis-Hastings Algorithms

[illegible]

[illegible]

[illegible]

[illegible]

Example 10.18. Two-dimensional mixture. [illegible]

[illegible]

[illegible]

10.3.2 Mixture-mag Cycles

[illegible]

[illegible]

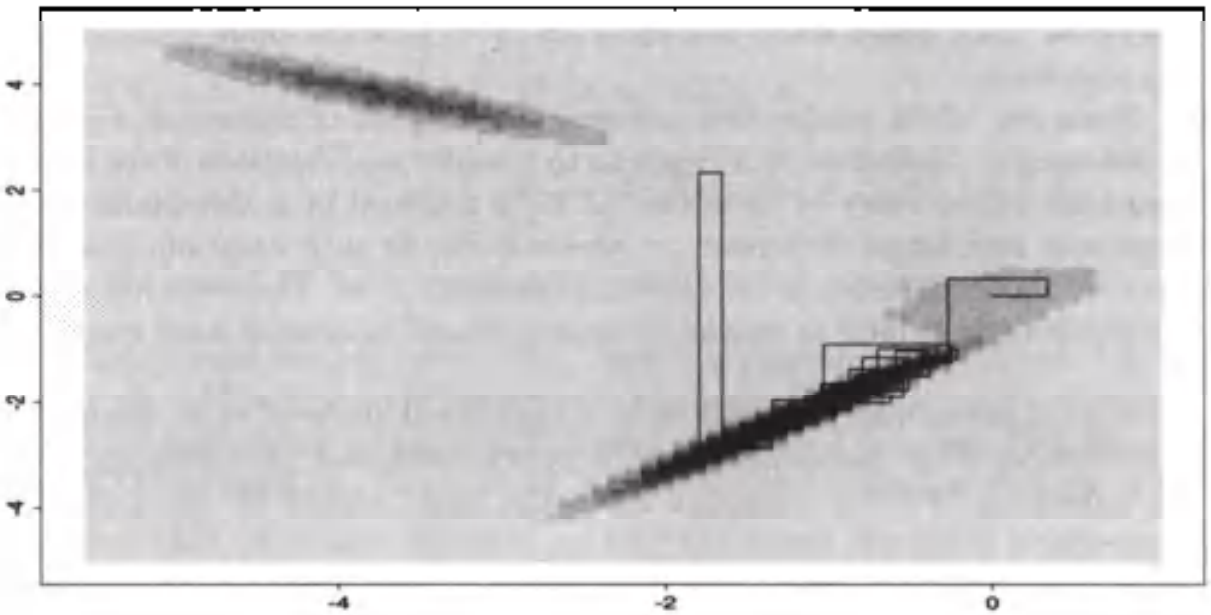


Fig. 10.3. Successive (full conditional) moves of the Gibbs chain on the surface of the stationary distribution (10.7), represented by gray levels (darker shades mean higher elevations).

f and the instrumental distribution. Moreover, the freedom brought by Metropolis–Hastings methods sometimes allows for remedies to these drawbacks through the modification of some scale (parameters or hyperparameters are particularly useful).

Compared to the Gibbs sampler, a failing of Metropolis–Hastings algorithms is to miss the finer details of the distribution f, if the simulation is "too coarse." However, following Tierney (1994), a way to take advantage of both algorithms is to implement a *hybrid* approach which uses both Gibbs sampling and Metropolis–Hastings algorithms.

Definition 10.19. An *hybrid MCMC algorithm* is a Markov chain Monte Carlo method which simultaneously utilizes both Gibbs sampling steps and Metropolis–Hastings steps. If $K_1, \ldots, K_n$ are kernels which correspond to these different steps and if $(\alpha_1, \ldots, \alpha_n)$ is a probability distribution, a *mixture* of $K_1, K_2, \ldots, K_n$ is an algorithm associated with the kernel

$$\tilde{K} = \alpha_1 K_1 + \cdots + \alpha_n K_n$$

and a *cycle* of $K_1, K_2, \ldots, K_n$ is the algorithm with kernel

$$K^* = K_1 \circ \cdots \circ K_n,$$

where "o" denotes the composition of functions.

Of course, this definition is somewhat ambiguous since Theorem 10.13 states that the Gibbs sampling is already a composition of Metropolis–Hastings kernels; that is, a cycle according to the above definition. Definition 10.19

[illegible]

[illegible]

[illegible]

Proposition. 10.21. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 10.21. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

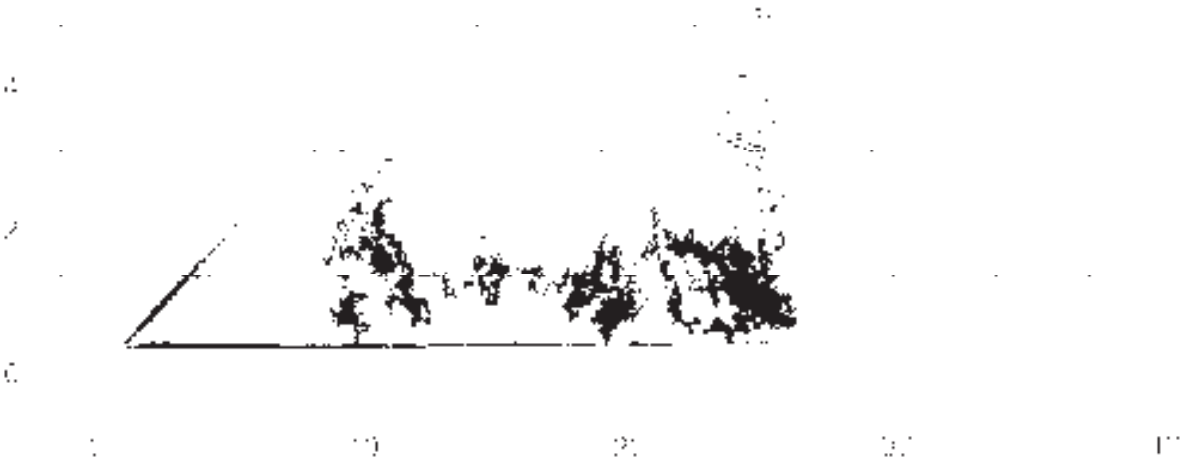


Fig. 10.4 [illegible]

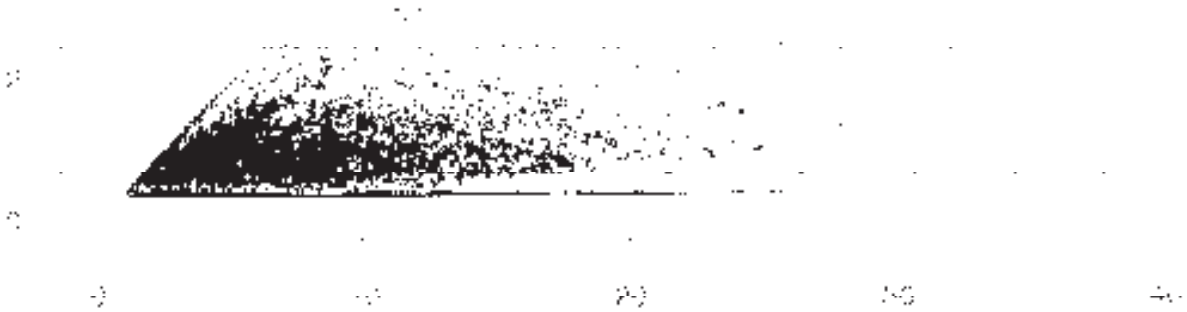


Fig. 10.5 [illegible]

10.3.3 Metropolizing the Gibbs Sampler

[illegible]

compromise suggested by Müller (1991, 1993) (sometimes called "Metropolis-within-Gibbs"). In any step i of the algorithm [A.40] with a difficult simulation from $g_i(y_i|y_j, j \neq i)$, substitute a simulation from an instrumental distribution q_i. In the setup of [A.40], Müller's (1991) modification is

Algorithm A.43 –Hybrid MCMC–

For $i = 1, \ldots, p$, given $(y_1^{(t+1)}, \ldots, y_{i-1}^{(t+1)}, y_i^{(t)}, \ldots, y_p^{(t)})$:

1. Simulate

$$\tilde{y}_i \sim q_i(y_i|y_1^{(t+1)}, \ldots, y_i^{(t)}, y_{i+1}^{(t)}, \ldots, y_p^{(t)})$$

2. Take [A.43]

$$y_i^{(t+1)} = \begin{cases} y_i^{(t)} & \text{with probability } 1-\rho, \\ \tilde{y}_i & \text{with probability } \rho, \end{cases}$$

where

$$\rho = 1 \wedge \left\{ \frac{\left(\dfrac{g_i(\tilde{y}_i|y_1^{(t+1)}, \ldots, y_{i-1}^{(t+1)}, y_{i+1}^{(t)}, \ldots, y_p^{(t)})}{q_i(\tilde{y}_i|y_1^{(t+1)}, \ldots, y_{i-1}^{(t+1)}, y_i^{(t)}, y_{i+1}^{(t)}, \ldots, y_p^{(t)})} \right)}{\left(\dfrac{g_i(y_i^{(t)}|y_1^{(t+1)}, \ldots, y_{i-1}^{(t+1)}, y_{i+1}^{(t)}, \ldots, y_p^{(t)})}{q_i(y_i^{(t)}|y_1^{(t+1)}, \ldots, y_{i-1}^{(t+1)}, \tilde{y}_i, y_{i+1}^{(t)}, \ldots, y_p^{(t)})} \right)} \right\}.$$

An important point about this substitution is that the above Metropolis–Hastings step is only used *once* in an iteration from [A.40]. The modified step thus produces a *single* simulation $\tilde{y}_i$ instead of trying to approximate $g_i(y_i|y_j, j \neq i)$ more accurately by producing T simulations from q_i. The reasons for this choice are twofold: First, the resulting hybrid algorithm is valid since g is its stationary distribution (see Section 10.3.2 or Problems 10.13 and 10.15.). Second, Gibbs sampling also leads to an approximation of g. To provide a more "precise" approximation of $g_i(y_i|y_j, j \neq i)$ in [A.43] does not necessarily lead to a better approximation of g and the replacement of g_i by q_i may even be beneficial for the speed of excursion of the chain on the surface of g.[7] (See also Chen and Schmeiser 1998.)

When several Metropolis–Hastings steps appear in a Gibbs algorithm, either because of some complex conditional distributions (see Besag et al. 1995) or because of convergence contingencies, a method proposed by Müller (1993) is to run a single acceptance step after the p conditional simulations. This approach is more time-consuming in terms of simulation (since it may result in the rejection of the p simulated components), but the resulting algorithm

[7] The multi-stage Gibbs sampler is itself an illustration of this, as announced in Section 9.5.

can be written as a simple Metropolis–Hastings method instead of a (hybrid) combination of Metropolis–Hastings algorithms. Moreover, it produces an approximation $q(y)$ of the distribution $g(y)$ rather than *local* approximations of the conditional distributions of the subvectors y_i.

We conclude this section with a surprising result of Liu (1995), who shows that, in a discrete state setting, Gibbs sampling can be improved by Metropolis–Hastings steps. The improvement is expressed in terms of a reduction of the variance of the empirical mean of the $h(y^{(t)})$'s for the Metropolis–Hastings approach. This modification, called *Metropolization* by Liu (1995), is based on the following result, established by Peskun (1973), where $T_1 \ll T_2$ means that the non-diagonal elements of T_2 are larger that those of T_1 (see Problem 7.39 for a proof).

Lemma 10.22. *Consider two reversible Markov chains on a countable state-space, with transition matrices T_1 and T_2 such that $T_1 \ll T_2$. The chain associated with T_2 dominates the chain associated with T_1 in terms of variances.*

Given a conditional distribution $g_i(y_i|y_j, j \neq i)$ on a discrete space, the modification proposed by Liu (1995) is to use an additional Metropolis–Hastings step.

Algorithm A.44 –Metropolization of the Gibbs Sampler–

Given $y^{(t)}$,

1. Simulate $z_i \neq y_i^{(t)}$ with probability

$$\frac{g_i(z_i|y_j^{(t)}, j \neq i)}{1 - g_i(y_i^{(t)}|y_j^{(t)}, j \neq i)};$$

2. Accept $y_i^{(t+1)} = z_i$ with probability [A.44]

$$\frac{1 - g_i(y_i^{(t)}|y_j^{(t)}, j \neq i)}{1 - g_i(z_i|y_j^{(t)}, j \neq i)} \wedge 1.$$

The probability of moving from $y_i^{(t)}$ to a different value is then necessarily higher in [A.44] than in the original Gibbs sampling algorithm and Lemma 10.22 implies the following domination result.

Theorem 10.23. *The modification* [A.44] *of the Gibbs sampler is more efficient in terms of variances.*

Example 10.24. (Continuation of Example 9.8) For the aggregated multinomial model of Tanner and Wong (1987), the completed variables z_i

[illegible]

[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]

Table 10.2. [illegible]

[illegible]

10.4 Statistical Considerations

10.4.1 Representativeness

[illegible]

[illegible]

[illegible]

[illegible]

Example 10.25. Random effects model. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 10.25. (Continuation of Example 10.2) [illegible]

$$[illegible] \sum_{[illegible]} [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 10.27. Activity [illegible]

[illegible]

[illegible]

[

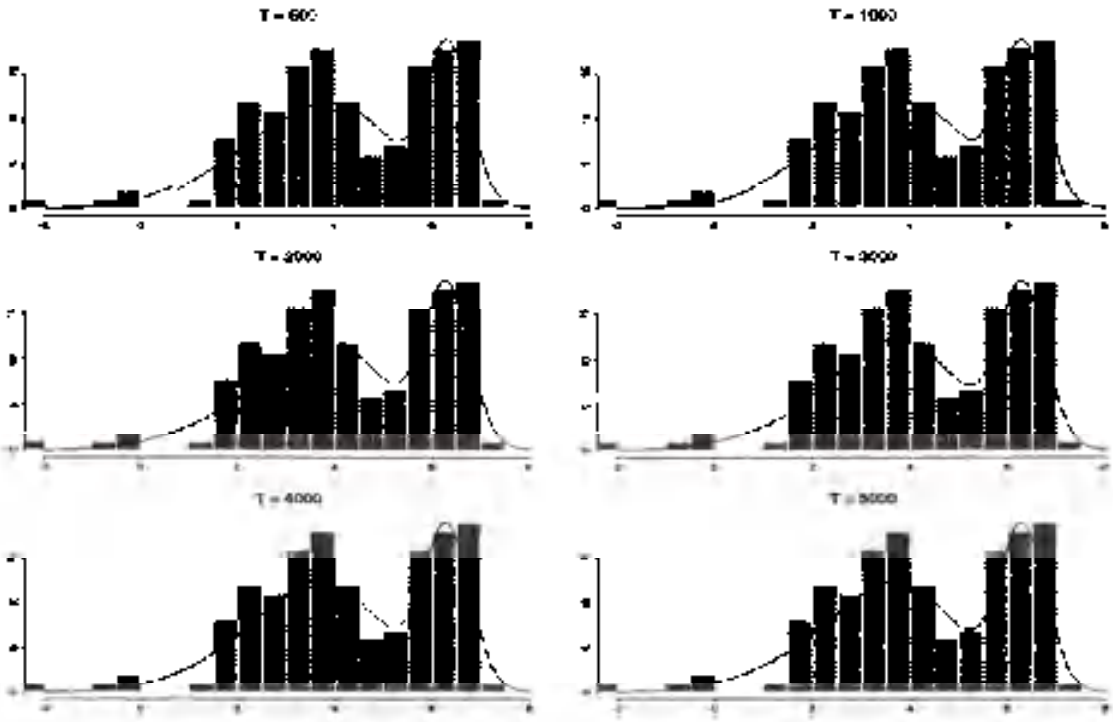


Fig. 10.7. Evolution of the estimation of the density (10.13) for three components and T iterations of a Gibbs sampling algorithm. The estimations are based on 149 observations of acidity levels for lakes in the American Northeast, used in Crawford et al. (1992) and represented by the histograms.

σ_j^{-1} is distributed from a $\mathcal{U}_{[0,1]}$ distribution. The simulation steps for σ_j are modified, but there exists a direct Accept–Reject algorithm to simulate from the conditional posterior distribution of the σ_j's. An alternative solution is to keep the identifiability constraints and to use a hybrid Markov chain Monte Carlo algorithm, where, in every U iterations, a random permutation of the values of z_i is generated via a Metropolis–Hastings scheme:

Algorithm A.45 –Hybrid Allocation Mixture Estimation–

0. Simulate Z_i $(i = 1, \ldots, n)$.
1. Generate a random permutation ψ on $\{1, 2, \ldots, k\}$ and derive

$$\tilde{z} = (\psi(z_i))_i \qquad \text{and} \qquad \tilde{\xi} = (\tilde{p}, \tilde{\mu}, \tilde{w}) = (p_{\psi(j)}, \mu_{\psi(j)}, w_{\psi(j)})_j. \qquad [A.45]$$

2. Generate ξ' conditionally on $(\tilde{\xi}, \tilde{z})$ by a standard Gibbs sampler iteration.
3. Accept $(\xi', \tilde{z})$ with probability

$$(10.14) \qquad \frac{\pi(\xi', \tilde{z})\pi_1(\psi^{-1}(z^-)|\xi')\,\pi_2(\xi|z^-, \psi(\xi'))}{\pi(\xi, z^-)\pi_1(\psi^{-1}(\tilde{z})|\xi)\,\pi_2(\xi'|\tilde{z}, \tilde{\xi})} \wedge 1\,,$$

where $\pi(\xi, z)$ denotes the distribution of (ξ, z),

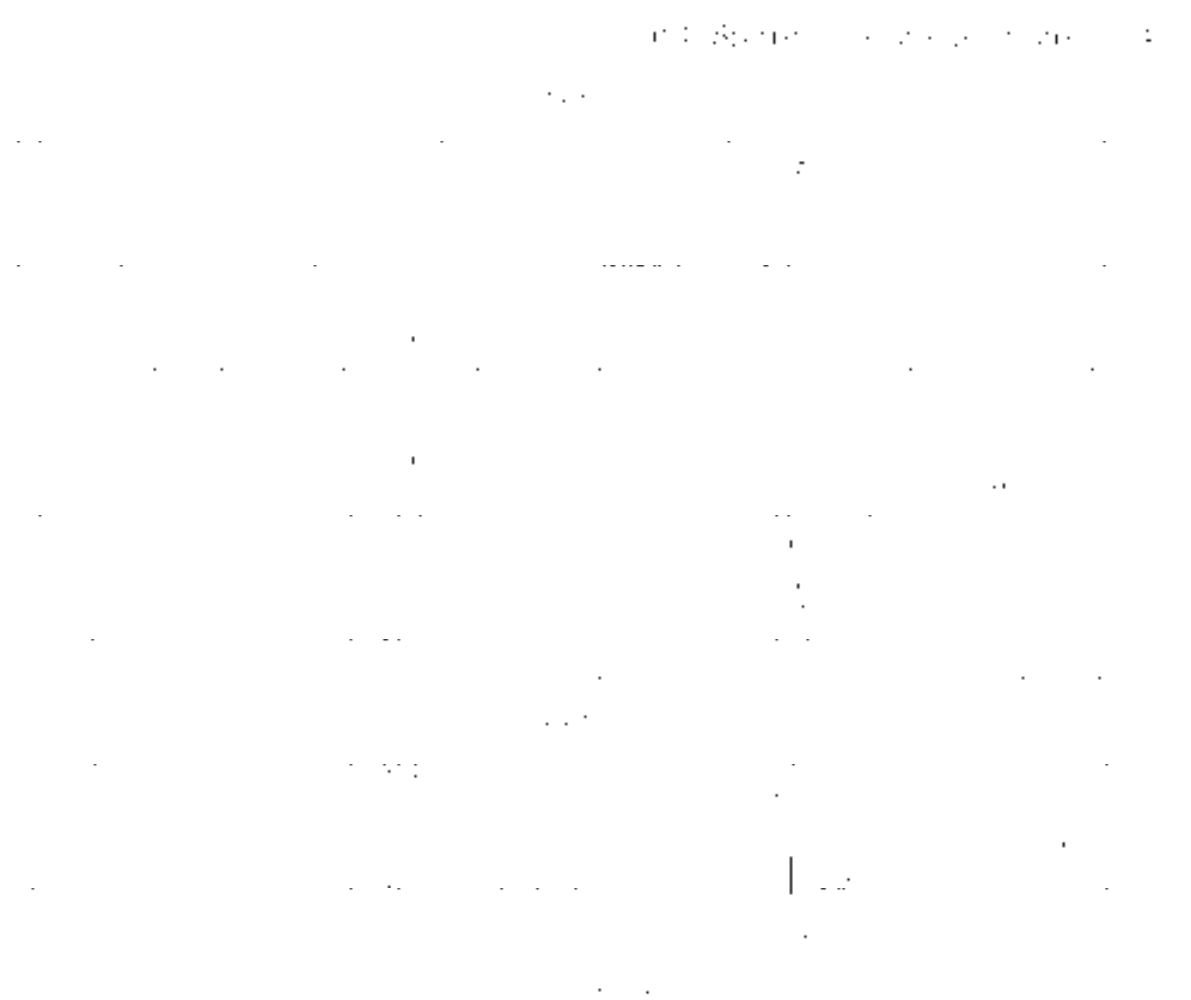


Fig. [illegible]

II [illegible]

[illegible]

10.4.2 Rao-Blackwellization

Some of the Rao-Blackwellization results of Section 9.3 carry over to the multi-stage Gibbs sampler. For example, Liu et al. (1995) are able to extend Proposition 9.18 to the multi-stage Gibbs sampler in the case of the *random Gibbs sampler*, where every step only updates a single component of y. In the setup of [A.40], define a multinomial distribution $\sigma = (\sigma_1, \ldots, \sigma_p)$.

Algorithm A.46 –Random Gibbs Sampler–

Given $y^{(t)}$,

1. Select a component $\nu \sim \sigma$. [A.46]
2. Generate $Y_\nu^{(t+1)} \sim g_\nu(y_\nu | y_j^{(t)}, j \neq \nu)$ and take

$$y_j^{(t+1)} = y_j^{(t)} \qquad \text{for} \qquad j \neq \nu.$$

Note that, although [A.46] only generates one component of y at each iteration, the resulting chain is strongly irreducible (Definition 6.13) because of the random choice of ν. It also satisfies the following property.

Proposition 10.28. *The chain $\{Y^{(t)}\}$ generated by [A.46] has the property that for every function $h \in \mathcal{L}_2(g)$, the covariance $\mathrm{cov}(h(Y^{(0)}), h(Y^{(t)}))$ is positive and decreasing in t.*

Proof. Assume again $\mathbb{E}_g[h(Y)] = 0$; then

$$\begin{aligned}
\mathbf{E}[h(Y^{(0)})h(Y^{(t)})] &= \sum_{v=1}^{p} \sigma_v \, \mathrm{E}[\mathbb{E}[h(Y^{(0)})h(Y^{(1)}) | \nu = v, (Y_j^{(0)}, j \neq v)]] \\
&= \sum_{v=1}^{p} \sigma_v \, \mathbb{E}[\mathbb{E}[h(Y^{(0)}) | (Y_j^{(0)}, j \neq v)]^2] \\
&= \mathbf{E}[\mathrm{E}[h(Y) | \nu, (Y_j, j \neq \nu)]^2],
\end{aligned}$$

due to the reversibility of the chain and the independence between $Y^{(0)}$ and $Y^{(1)}$, conditionally on ν and $(y_j^{(0)}, j \neq v)$. A simple recursion implies

$$\mathbf{E}[h(Y^{(0)})h(Y^{(t)})] = \mathrm{var}(\mathrm{E}[\cdots \mathbf{E}[\mathrm{E}[h(Y) | \nu, (Y_j, j \neq \nu)] | Y] \cdots]),$$

where the second term involves t conditional expectations, successively in $(\nu, (y_j, j \neq \nu))$ and in Y. □

This proof suggests choosing a distribution σ that more heavily weights components with small $\mathbb{E}[h(Y) | (y_j, j \neq v)]^2$, so the chain will typically visit components where $g_v(y_v | y_j, j \neq v)$ is not too variable. However, the opposite choice seems more logical when considering the *speed* of convergence to

[illegible]

[illegible]

[illegible]

[illegible]

[illegible] Tropical Forests

[illegible]

[illegible]

[illegible]

Example 10.29. Conditional exponential distributions. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 10.30. Normal inverse posterior. [illegible]

[illegible]

[illegible]

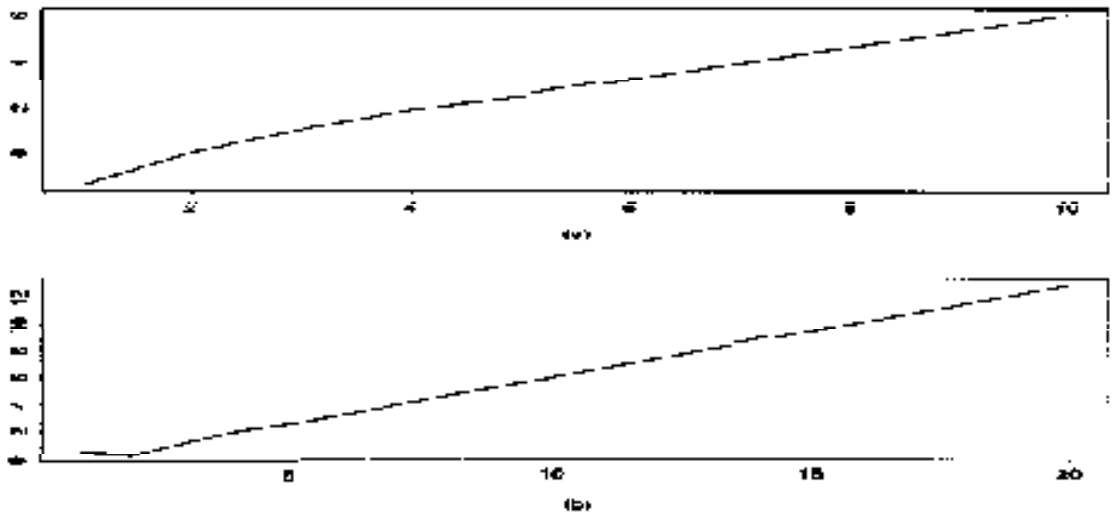


Fig. 10.9. First iterations of the chains (a) $\log(|\theta^{(t)}|)$ and (b) $\log(\sigma^{(t)})$ for a divergent Gibbs sampling algorithm.

$$\int f(x|\zeta)\,\pi(\zeta)\,d\zeta = \int \frac{1}{\sqrt{2\pi}\sigma}\, e^{-(x-\theta)^2/(2\sigma^2)}\,\frac{d\theta\,d\sigma}{\sigma^2} = \infty,$$

and, hence, $\pi(\theta,\sigma|x)$ is not defined.

However, this impropriety may never be detected as it is common Bayesian practice to consider only the proportionality relation between $\pi(\theta,\sigma|x)$ and $f(x|\theta,\sigma)\,\pi(\theta,\sigma)$. Similarly, a derivation of a Gibbs sampler for this model is also based on the proportionality relation since the conditional distributions $\pi(\theta|\sigma,x)$ and $\pi(\sigma|\theta,x)$ can be obtained directly as

$$\pi(\theta|\sigma,x) \propto e^{-(\theta-x)^2/(2\sigma^2)}, \qquad \pi(\sigma|\theta,x) \propto \sigma^{-3}\, e^{-(\theta-x)^2/(2\sigma^2)},$$

which lead to the full conditional distributions $\mathcal{N}(x,\sigma^2)$ and $\mathcal{E}xp((\theta-x)^2/2)$ for θ and σ^{-2}, respectively.

This practice of omitting the computation of normalizing constants is justified as long as the joint distribution does exist and is particularly useful because these constants often cannot be obtained in closed form. However, the use of improper prior distributions implies that the derivation of conditional distributions from the proportionality relation is not always justified. In the present case, the Gibbs sampler associated with these two conditional distributions is

1. Simulate $\theta^{(t+1)} \sim \mathcal{N}(x,\sigma^{(t)2})$;
2. Simulate $\varepsilon \sim \mathcal{E}xp\left((\theta^{(t+1)}-x)^2/2\right)$ and take $\sigma^{(t+1)} = 1/\sqrt{\varepsilon}$.

[illegible]

[illegible]

[illegible]

Example 10.41. Improper random effects posterior. [illegible]

[illegible]

[illegible]

[

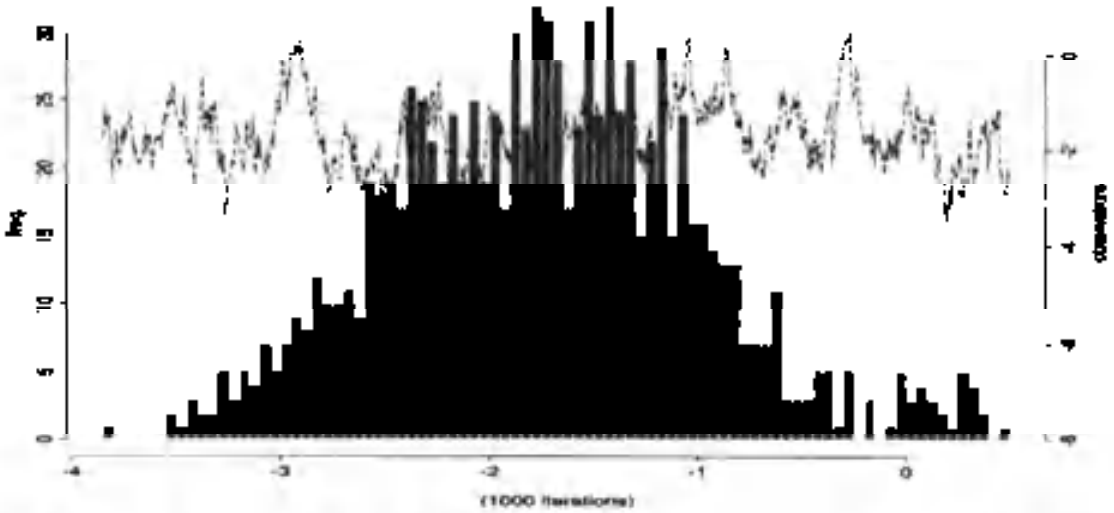


Fig. 10.10. *Sequence $(\beta^{(t)})$ and corresponding histogram for the random effects model.* The measurement scale for the $\beta^{(t)}$'s is on the right and the scale of the histogram is on the left.

are well defined and a Gibbs sampling can be easily implemented in this setting. Figure 10.10 provides the sequence of the $\beta^{(t)}$ produced by [A.40] and the corresponding histogram for 1000 iterations. The trend of the sequence and the histogram do not indicate that the corresponding "joint distribution" does not exist (Problem 10.25). ‖

Under some regularity conditions on the transition kernel, Hobert and Casella (1996) have shown that if there exist a positive function b, $\varepsilon > 0$ and a compact set C such that $b(x) < \varepsilon$ for $x \in C^c$, the chain $(y^{(t)})$ satisfies

$$\liminf_{t \to +\infty} \frac{1}{t} \sum_{s=1}^{t} b(y^{(s)}) = 0. \tag{10.15}$$

A drawback of the condition (10.15) is the derivation of the function b, whereas the monitoring of a lim inf is delicate in a simulation experiment. Other references on the analysis of improper Gibbs samplers can be found in Besag et al. (1995) and in the subsequent discussion (see, in particular, Roberts and Sahu 1997), as well as in Natarajan and McCulloch (1998).

10.5 Problems

10.1 Propose a direct Accept–Reject algorithm for the conditional distribution of Y_2 given y_1 in Example 10.1. Compare with the Gibbs sampling implementation in a Monte Carlo experiment. (*Hint:* Run parallel chains and compute 90% confidence intervals for both methods.)

[illegible]

10.2 [illegible]

10.3 [illegible]

10.4 [illegible]

10.5 [illegible]

[illegible]

[illegible]

10.6 [illegible]

10.7 [illegible]

$$[illegible]$$

$$[illegible]$$

10.8 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

0.11 [illegible]

0.12 [illegible]

[illegible]

[illegible]

0.13 [illegible]

[illegible]

[illegible]

[

Show that the following algorithm provides a valid approximation of the posterior distribution of (β, σ^2).

1. Simulate $y_i^* \sim \mathcal{N}_-(x_i^t\beta, \sigma^2, 0)$ if $y_i = 0$.
2. Simulate $(\beta, \sigma) \sim \pi(\beta, \sigma | y^*, x)$ with [A_{31}]

$$\pi(\beta, \sigma | y^*, x) \propto \sigma^{-n} \exp\left\{-\sum_i (y_i^* - x_i^t\beta)^2/2\sigma^2\right\} \pi(\beta, \sigma) .$$

10.19 Let (X_n) be a reversible stationary Markov chain (see Definition 6.44) with $\mathbb{E}[X_i] = 0$.

(a) Show that the distribution of $X_k|X_1$ is the same as the distribution of $X_1|X_k$.

(b) Show that the covariance between alternate random variables is positive. More precisely, show that $\text{cov}(X_0, X_{2\nu}) = \mathbb{E}[\mathbb{E}(X_0|X_\nu)^2] > 0$.

(c) Show that the covariance of alternate random variables is decreasing; that is, show that $\text{cov}(X_0, X_{2\nu}) \geq \text{cov}(X_0, X_{2(\nu+1)})$. (*Hint*: Use the fact that

$$\mathbb{E}[\mathbb{E}(X_0|X_\nu)^2] = \text{var}[\mathbb{E}(X_0|X_\nu)] \geq \text{var}[\mathbb{E}\{\mathbb{E}(X_0|X_\nu)|X_{\nu+1}\}],$$

and show that this latter quantity is $\text{cov}(X_0, X_{2(\nu+1)})$.)

10.20 Show that a Gibbs sampling kernel with more than two full conditional steps cannot produce interleaving chains.

10.21 In the setup of Example 10.2:

(a) Consider a grid in $\mathbb{R}^2$ with the simple nearest-neighbor relation $\mathcal{N}$ (that is, $(i, j) \in \mathcal{N}$ if and only if $\min(|i_1 - j_1|, |i_2 - j_2|) = 1$). Show that a Gibbs sampling algorithm with only two conditional steps can be implemented in this case.

(b) Implement the Metropolization scheme of Liu (1995) discussed in Section 10.3.3 and compare the results with those of part (a).

10.22 (Gelfand et al. 1995) In the setup of Example 10.25, show that the original parameterization leads to the following correlations:

$$\rho_{\mu_i, \mu_j} = \left(1 + \frac{I\sigma_y^2}{J\sigma_\alpha^2}\right)^{-1/2}, \qquad \rho_{\alpha_i, \alpha_j} = \left(1 + \frac{I\sigma_y^2}{J\sigma_\alpha^2}\right)^{-1},$$

for $i \neq j$.

10.23 (Continuation of Problem 10.22) For the hierarchical parameterization, show that the correlations are

$$\rho_{\mu_i, \mu_j} = \left(1 + \frac{IJ\sigma_\alpha^2}{\sigma_y^2}\right)^{-1/2}, \qquad \rho_{\alpha_i, \alpha_j} = \left(1 + \frac{IJ\sigma_\alpha^2}{\sigma_y^2}\right)^{-1},$$

for $i \neq j$.

10.24 In Example 10.29, we noted that a pair of conditional densities is *functionally compatible* if the ratio $f_1(x|y)/f_2(y|x) = h_1(x)/h_2(y)$, for some functions h_1 and h_2. This is a necessary condition for a joint density to exist, but not a sufficient condition. If such a joint density does exist, the pair f_1 and f_2 would be *compatible*.

(a) Formulate the definitions of compatible and functionally compatible for a set of densities $f_1, \ldots, f_m$.

[illegible]

[illegible]

[illegible]

10.27 [illegible]

[illegible]

[illegible]

10.28 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

10.27 [illegible]

[illegible]

[illegible]

10.28 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

10.31 [illegible]

[illegible]

[illegible]

10.32 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

10.33 [illegible]

10.34 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

10.35 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

10.36 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

10.37 [illegible]

(a) Use the identifiability constraint, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k$, to write the mixture with the "cascade" parameterization

$$p\,\mathcal{E}xp(\lambda) + \sum_{j=1}^{k-1}(1-p)\cdots q_j\,\mathcal{E}xp(\lambda\sigma_1\cdots\sigma_j),$$

with $q_{k-1} = 1$, $\sigma_1 \leq 1, \ldots, \sigma_{k-1} \leq 1$.

(b) For the prior distribution $\pi(\lambda) = \frac{1}{\lambda}$, $p, q_j \sim \mathcal{U}_{[0,1]}$, $\sigma_j \sim \mathcal{U}_{[0,1]}$, show that the corresponding posterior distribution is always proper.

(c) For the prior of part (b), show that [A.34] leads to the following algorithm:

Algorithm A.47 –Exponential Mixtures–

2. **Generate**

$$\begin{aligned}
\lambda &\sim \mathcal{G}a\left(n, n_0\overline{x}_0 + \sigma_1 n_1\overline{x}_1 + \cdots + \sigma_1\cdots\sigma_{k-1}n_{k-1}\overline{x}_{k-1}\right),\\
\sigma_1 &\sim \mathcal{G}a\left(n_1 + \cdots + n_{k-1}, \lambda\{n_1\overline{x}_1 + \sigma_2 n_2\overline{x}_2 + \cdots + \right.\\
&\qquad \left.\sigma_2\cdots\sigma_{k-1}n_{k-1}\overline{x}_{k-1}\}\right)\mathbb{I}_{\sigma_1\leq 1},\\
&\vdots\\
\sigma_{k-1} &\sim \mathcal{G}a\left(n_{k-1}, \lambda n_{k-1}\overline{x}_{k-1}\right)\mathbb{I}_{\sigma_{k-1}\leq 1},\\
p &\sim \mathcal{B}e(n_0+1, n-n_0+1),\\
&\vdots\\
q_{k-2} &\sim \mathcal{B}e(n_{k-2}+1, n_{k-1}+1),
\end{aligned}$$

where $n_0, n_1, \ldots, n_{k-1}$ denote the size of subsamples allocated to the components $\mathcal{E}xp(\lambda)$, $\mathcal{E}xp(\lambda\sigma_1)$, ..., $\mathcal{E}xp(\lambda\sigma_1\ldots\sigma_{k-1})$ and $n_0\overline{x}_0, n_1\overline{x}_1, \ldots, n_{k-1}\overline{x}_{k-1}$ are the sums of the observations allocated to these components.

10.6 Notes

10.6.1 A Bit of Background

Although somewhat removed from statistical inference in the classical sense and based on earlier techniques used in statistical Physics, the landmark paper by Geman and Geman (1984) brought Gibbs sampling into the arena of statistical application. This paper is also responsible for the name *Gibbs sampling*, because it implemented this method for the Bayesian study of *Gibbs random fields* which, in turn, derive their name from the physicist Josiah Willard Gibbs (1839–1903). This original implementation of the Gibbs sampler was applied to a discrete image processing problem and did not involve completion.

The work of Geman and Geman (1984) built on that of Metropolis et al. (1953) and Hastings (1970) and his student Peskun (1973), influenced Gelfand and Smith (1990) to write a paper that sparked new interest in Bayesian methods, statistical computing, algorithms, and stochastic processes through the use of computing algorithms such as the Gibbs sampler and the Metropolis–Hastings algorithm. It is interesting to see, in retrospect, that earlier papers had proposed similar solutions but did not find the same response from the statistical community. Among these, one may quote Besag (1974, 1986), Besag and Clifford (1989), Broniatowski et al. (1984), Qian and Titterington (1990), and Tanner and Wong (1987).

10.6.2 The BUGS Software

The acronym BUGS stands for *Bayesian inference using Gibbs sampling*. This software was developed by Spiegelhalter et al. (1995a,b,c) at the MRC Biostatistics Unit in Cambridge, England. As shown by its name, it has been designed to take advantage of the possibilities of the Gibbs sampler in Bayesian analysis. BUGS includes a language which is C or R like and involves declarations about the model, the data, and the prior specifications, for single or multiple levels in the prior modeling. For instance, for the benchmark nuclear pump failures dataset of Example 10.17, the model and priors are defined by

```
for (i in 1:N) {
    theta[i] ~ dgamma(alpha,beta)
    lambda[i]~ theta[i] * t[i]
    x[i]     ~ dpois(lambda[i])
}
alpha ~ dexp(1.0)
beta  ~ dgamma(0.1,1.0)
```

(see Spiegelhalter et al. 1995b, p.9). Most standard distributions are recognized by BUGS (21 are listed in Spiegelhalter et al. 1995a), which also allows for a large range of transforms. BUGS also recognizes a series of commands like **compile**, **data**, **out**, and **stat**. The output of BUGS is a table of the simulated values of the parameters after an open number of warmup iterations, the batch size being also open.

A major restriction of this software is the use of the conjugate priors or, at least, log-concave distributions for the Gibbs sampler to apply. However, more complex distributions can be handled by discretization of their support and assessment of the sensitivity to the discretization step. In addition, improper priors are not accepted and must be replaced by proper priors with small precision, like **dnorm(0,0.0001)**, which represents a normal modeling with mean 0 and *precision* (inverse variance) 0.0001.

The BUGS manual (Spiegelhalter et al. 1995a) is quite informative and well written.[12] In addition, the authors have compiled a most helpful example manual (Spiegelhalter et al. 1995b,1996), which exhibits the ability of BUGS to deal with an amazing number of models, including meta-analysis, latent variable, survival analysis, nonparametric smoothing, model selection and geometric modeling, to name a few. (Some of these models are presented in Problems 10.29–10.36.) The BUGS software is also compatible with the convergence diagnosis software CODA presented in Note 12.6.2.

10.6.3 Nonparametric Mixtures

Consider $X_1, \ldots, X_n$ distributed from a mixture of geometric distributions,

[12] At the present time, that is, Spring 2004, the BUGS software is available as freeware on the Web site **http://www.mrc-bsu.cam.ac.uk/bugs** for a wide variety of platforms.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

11.6.4 Graphics Modes

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Exercise 10.34 Graphical Gaussian model [illegible]

[illegible]

[illegible]

[illegible]

11

Variable Dimension Models and Reversible Jump Algorithms

[illegible]

[illegible]

11.1 Variable Dimension Models

[illegible]

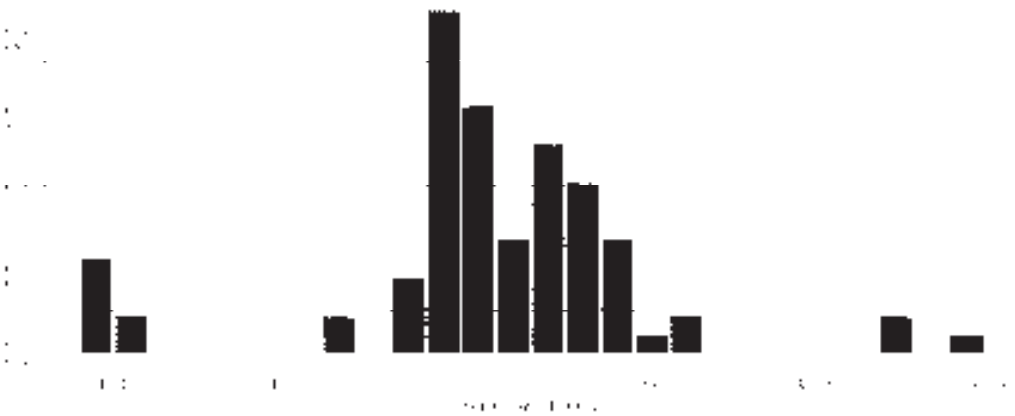


Fig. 11.1. [illegible]

[illegible]

Example 11.1. Mixture modeling. [illegible]

$$[illegible] \quad \underline{\Sigma} \quad [illegible]$$

[illegible]

11.1.1. Bayesian Model Choice

[illegible]

Definition 11.2. [illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 11.3. Order of an ARM model. [illegible]

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

11.2 Reversible Jump Algorithms

[illegible]

11.2.1 Green's Algorithm

[illegible]

$$[illegible] \qquad [illegible]$$

[illegible]

[illegible]

$$[illegible] \sum \int [illegible]$$

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$\int [illegible] \qquad \int [illegible]$$

$$\int [illegible]$$

$$\int [illegible]$$

[illegible]

[illegible]

(1995) is to supplement each of the spaces Θ_{k_1} and Θ_{k_2} with adequate artificial spaces in order to create a *bijection* between them.

For instance, if $\dim(\Theta_{k_1}) > \dim(\Theta_{k_2})$ and if the move from Θ_{k_1} to Θ_{k_2} can be represented by a deterministic transformation of $\theta^{(k_1)}$

$$\theta^{(k_2)} = T(\theta^{(k_1)}),$$

Green (1995) imposes a *dimension matching* condition which is that the opposite move from Θ_{k_2} to Θ_{k_1} is concentrated on the curve

$$\{\theta^{(k_1)} \,:\, \theta^{(k_2)} = T(\theta^{(k_1)})\}\,.$$

In the general case, if $\theta^{(k_1)}$ is completed by a simulation $u_1 \sim g_1(u_1)$ into $(\theta^{(k_1)}, u_1)$ and $\theta^{(k_2)}$ by $u_2 \sim g_2(u_2)$ into $(\theta^{(k_2)}, u_2)$ so that the mapping between $(\theta^{(k_1)}, u_1)$ and $(\theta^{(k_2)}, u_2)$ is a bijection,

$$(\theta^{(k_2)}, u_2) = T(\theta^{(k_1)}, u_1), \tag{11.3}$$

the probability of acceptance for the move from $\mathcal{M}_{k_1}$ to $\mathcal{M}_{k_2}$ is then

$$\min\left(\frac{\pi(k_2, \theta^{(k_2)})}{\pi(k_1, \theta^{(k_1)})}\,\frac{\pi_{21}g_2(u_2)}{\pi_{12}g_1(u_1)}\,\left|\frac{\partial T(\theta^{(k_1)}, u_1)}{\partial(\theta^{(k_1)}, u_1)}\right|, 1\right), \tag{11.4}$$

involving the Jacobian of the transform (11.3), the probability π_{ij} of choosing a jump to $\mathcal{M}_{k_j}$ while in $\mathcal{M}_{k_i}$, and g_i, the density of u_i. This proposal satisfies the detailed balance condition and the symmetry assumption of Green (1995) if the move from $\mathcal{M}_{k_2}$ to $\mathcal{M}_{k_1}$ also satisfies (11.3) with $u_2 \sim g_2(u_2)$.

The pseudo-code representation of Green's (1995) algorithm is thus as follows.

Algorithm A.48 —Green's Algorithm—

At iteration t, if $x^{(t)} = (m, \theta_m^{(t)})$,

1 Select model $\mathcal{M}_n$ with probability π_{mn}
2 Generate $u_{mn} \sim \varphi_{mn}(u)$
3 Set $(\theta_n, v_{nm}) = T_{mn}(\theta_m^{(t)}, u_{mn})$
4 Take $\theta_n^{(t)} = \theta_n$ with probability [A.48]

$$\min\left(\frac{\pi(n, \theta_n)}{\pi(m, \theta_m^{(t)})}\,\frac{\pi_{nm}\varphi_{nm}(v_{nm})}{\pi_{mn}\varphi_{mn}(u_{mn})}\,\left|\frac{\partial T_{mn}(\theta_m^{(t)}, u_{mn})}{\partial(\theta_n^{(t)}, u_{mn})}\right|, 1\right),$$

and take $\theta_m^{(t+1)} = \theta_m^{(t)}$ otherwise.

As pointed out by Green (1995), the density π does not need to be normalized, but the different component densities $\pi_k(\theta_k)$ must be known up to the same constant.

Example 11.1. A Linear Discriminant. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

11.2.2 A Fixed Dimension Representation

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

11.2.3 The Practice of Reversible Jump MCMC

[illegible]

Example 11.5. [illegible] versus quadratic regression. [illegible]

[illegible]

[illegible]

[illegible]

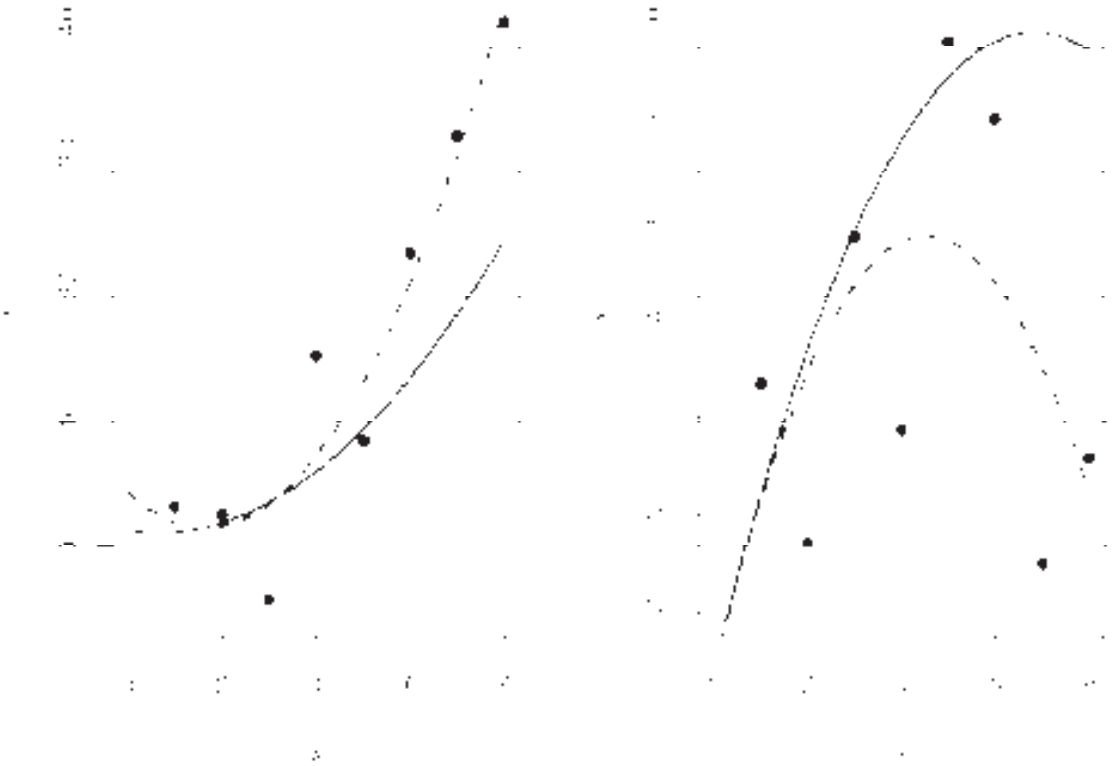


Fig. 1.2.

$$\sum$$

$$\frac{\quad}{\quad}$$

$$\sqrt{\quad}$$

[illegible]

Example [illegible] (Continuation of Example [illegible]) [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

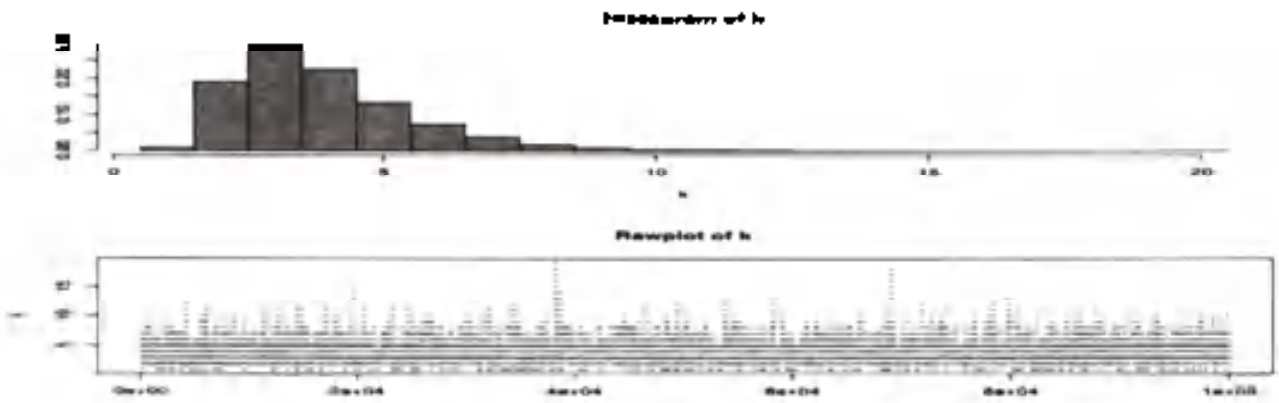


Fig. 11.3. Histogram and raw plot of 100,000 k's produced by a reversible jump MCMC algorithm for the Galaxy dataset of Figure 11.1.

gorithms, as a case of *model averaging*: this density is obtained as the average over iterations t of

$$\sum_{j=1}^{k^{(t)}} p_{jk}^{(t)} \mathcal{N}(\mu_{jk}^{(t)}, (\sigma_{jk}^{(t)})^2),$$

which approximates the posterior expectation $\mathrm{E}[f(y|\theta)|\mathbf{x}]$, where $\mathbf{x}$ denotes the data $x_1, \ldots, x_{82}$. (This can also be seen as a Bayesian equivalent of a kernel estimator.) ‖

Example 11.8. (Continuation of Example 11.3) An AR(p) model

$$X_t = \sum_{i=1}^{p} \theta_{ip} X_{t-i} + \sigma_p \epsilon_t \tag{11.7}$$

is often restricted by a stationarity condition on the process (X_t) (see, e.g., Robert, 2001, Section 4.5). While this constraint can be expressed either on the coefficients θ_{ip} of the model or on the partial auto-correlations, a convenient (from a prior point of view) representation is to use the lag-polynomial associated with the model $\mathcal{M}_p$,

$$\prod_{i=1}^{p} (1 - \lambda_i B)\, X_t = \epsilon_t\,, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2)\,,$$

and to constrain the inverse roots[3] λ_i, to stay within the unit circle if complex and within $[-1, 1]$ if real (see Barnett et al. 1996, Huerta and West 1999, and Robert, 2001, Section 4.5.2). A natural prior in this setting is then to use *uniform priors for the real and complex roots* λ_i, *that is*,

[3] In order to simplify notation, we have refrained from using a double index on the λ_i's, which should be, strictly speaking, the λ_{ip}'s.

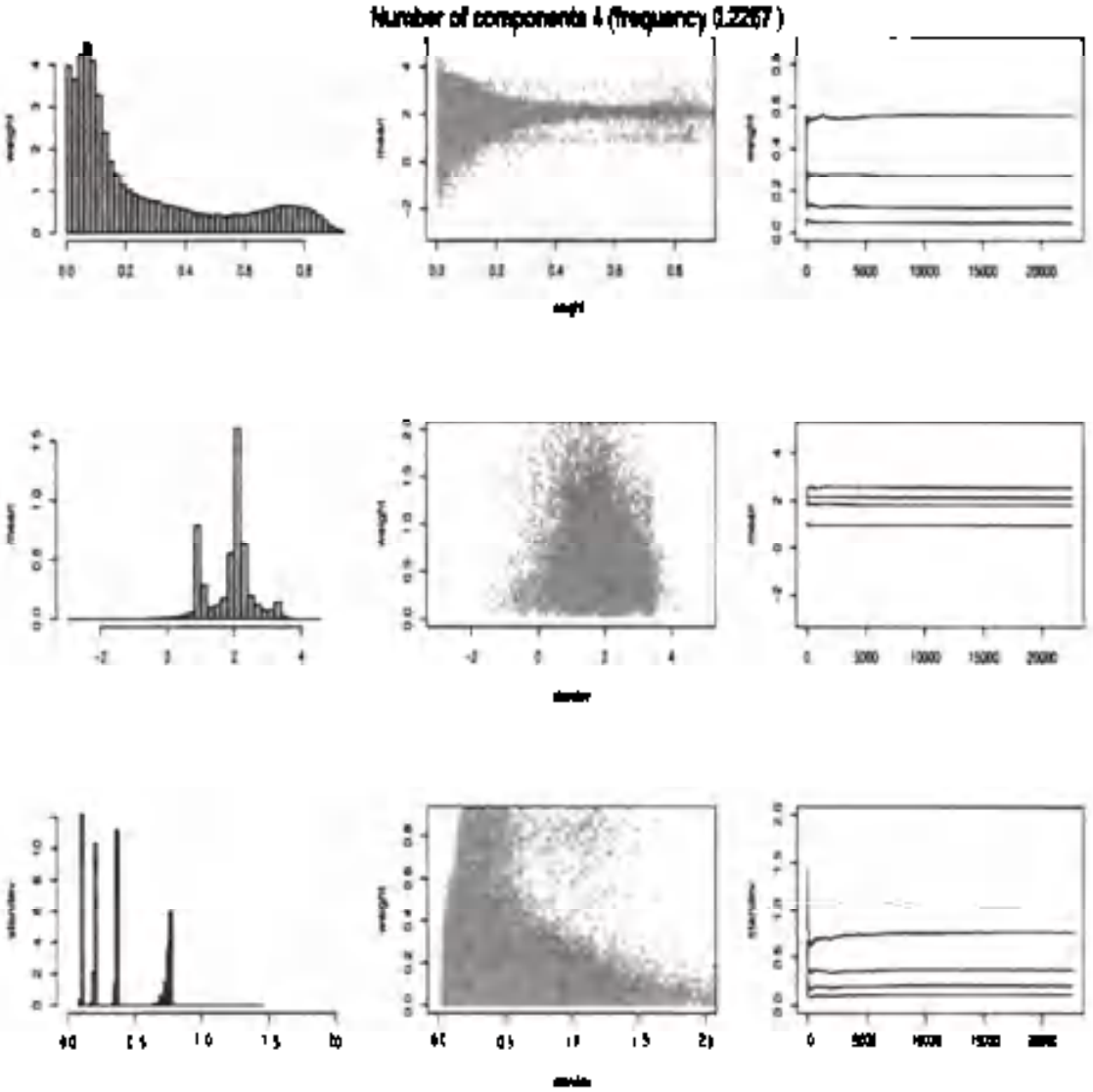


Fig. 11.4. Reversible jump MCMC output on the parameters of the model $\mathcal{M}_3$ for the Galaxy dataset, obtained by conditioning on $k = 3$. *The left column gives the histogram of the weights, means, and variances; the middle column the scatterplot of the pairs weights–means, means–variances, and variances–weights; the right column plots the cumulated averages (over iterations) for the weights, means, and variances.*

$$\pi_p(\boldsymbol{\lambda}) = \frac{1}{\lfloor p/2 \rfloor + 1} \prod_{\lambda_i \in \mathbb{R}} \frac{1}{2} \mathbb{I}_{|\lambda_i|<1} \prod_{\lambda_i \notin \mathbb{R}} \frac{1}{\pi} \mathbb{I}_{|\lambda_i|<1} ;$$

where $\lfloor p/2 \rfloor + 1$ is the number of different values of r_p.

Note that this factor, while unimportant for a fixed p setting, must be included within the posterior distribution when using reversible jump since it does not vanish in the acceptance probability of a move between models $\mathcal{M}_p$ and $\mathcal{M}_q$. If it is omitted in the acceptance probability of the reversible jump move, this results in a modification of the prior probability of each model $\mathcal{M}_p$, from $\varrho(p)$ to $\varrho(p)(\lfloor p/2 \rfloor + 1)$. See also Vermaak et al. (2003) for a similar phenomenon.

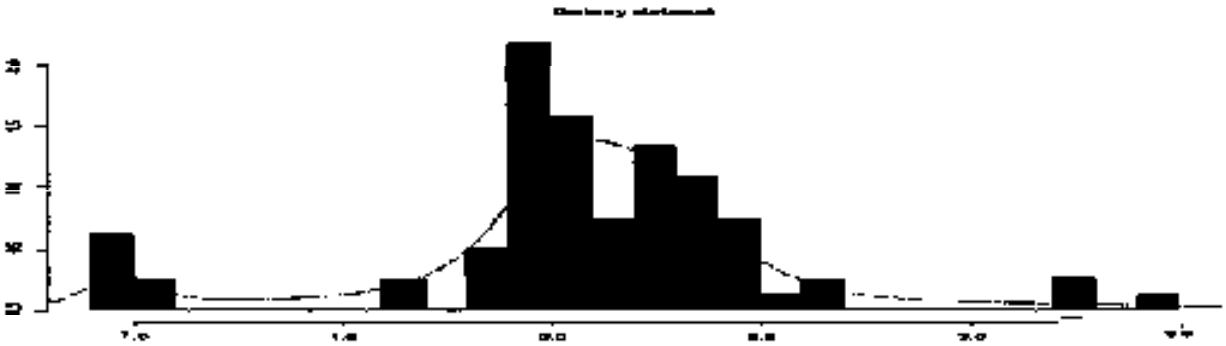


Fig. 11.5. Fit of the dataset by the averaged density, $\mathbb{E}[f(y|\theta)|x]$

The most basic choice for a reversible jump algorithm in this framework is to use the uniform priors as proposals for a birth-and-death scheme where the birth moves are either from $\mathcal{M}_p$ to $\mathcal{M}_{p+1}$, for the creation of a real root λ_{p+1}, or from $\mathcal{M}_p$ to $\mathcal{M}_{p+2}$, for the creation of two conjugate complex roots λ_{p+1} and $\bar{\lambda}_{p+1}$. The corresponding death moves are then completely determined: from $\mathcal{M}_{p+1}$ to $\mathcal{M}_p$ with the deletion of a real root (if any) and from $\mathcal{M}_{p+2}$ to $\mathcal{M}_p$ with the deletion of two conjugate complex roots (if any).

As in the birth-and-death proposal for Example 11.7, the acceptance probability simplifies quite dramatically since it is, for example,

$$\min\left(\frac{\pi_{(p+1)p}}{\pi_{p(p+1)}}\,\frac{(r_p+1)!}{r_p!}\,\frac{\lfloor p/2\rfloor+1}{\lfloor (p+1)/2\rfloor+1}\,\frac{\ell_{p+1}(\theta_{p+1})}{\ell_p(\theta_p)},1\right)$$

in the case of a move from $\mathcal{M}_p$ to $\mathcal{M}_{p+1}$. (As for the above mixture example, the factorials are related to the possible choices of the created and the deleted roots.)

Figure 11.6 presents some views of the corresponding reversible jump MCMC algorithm. Besides the ability of the algorithm to explore a range of values of k, it also shows that Bayesian inference using these tools is much richer, since it can, for instance, condition on or average over the order k, mix the parameters of different models and run various tests of these parameters. A last remark on this graph is that both the order and the value of the parameters are well estimated, with a characteristic trimodality on the histograms of the θ_i's, even when conditioning on k different from 3, the value used for the simulation. We refer the reader to Vermaak et al. (2003) and Ehlers and Brooks (2003) for different analyses of this problem using either the partial autocorrelation representation (and the Durbin-Levinson recursive algorithm) or the usual AR(p) parameters without stationarity constraint. ||

For any given variable dimension model, there exist an infinity of possible reversible jump algorithms. Compared with the fixed dimension case of, for instance, Chapter 7, there is more freedom and less structure, because the between model moves cannot rely on an Euclidean structure common to both models (unless they are embedded). Brooks et al. (2003b) tried to provide

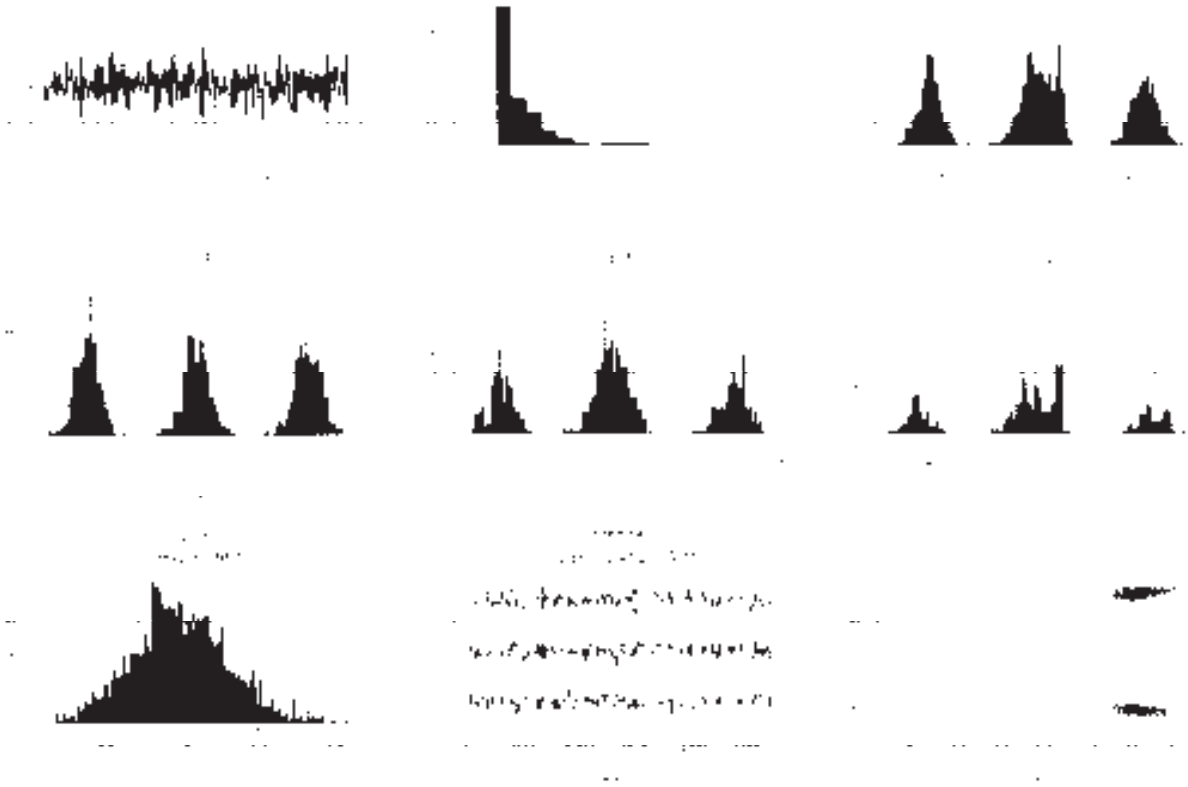


Fig. 12.6. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

11.3 Alternatives to Reversible Jump MCMC

[illegible]

11.3.1 Saturation

[illegible]

[illegible]

$$\pi(k,\theta_k)\, q_k(u_k) \prod_{i=1}^{n_{max}} \psi(\omega_i).$$

Within this augmented (or *saturated*) framework, there is no varying dimension anymore since, for all models, the entire vector (θ_k, u_k, ω) is of fixed dimension. Therefore, moves between models can be defined just as freely as moves between points of each model. See also Besag (2000) and Godsill (2003) for a similar development, which differs from the full saturation scheme of Carlin and Chib (1995) (Problem 11.9).

Brooks et al. (2003b) propose a three-stage MCMC update as follows.

Algorithm A.49 —Saturation Algorithm—

At iteration t

a. Update the current value of the parameter, θ_k, within model $\mathcal{M}_k$
b. Update u_k and ω conditional on θ_k
c. Update the model from $\mathcal{M}_k$ into $\mathcal{M}_{k'}$ using the bijection

$$(\theta_{k'}, u_{k'}) = T_{kk'}(\theta_k, u_k|\omega).$$

Note that, for specific models, saturation schemes appear rather naturally. For instance, Green (1995) considered a time series model with change-points, $Y_t \sim p(y_k - \theta_k)$, where θ_t changes values according to a geometric jump scheme (see Problems 11.15, 11.17 and 11.18). Whatever the number of change-points in the series, a reparameterization of the model by the missing data, composed of the indicators X_t of whether or not a change occurs at time t, creates a fixed dimension model. (See Cappé et al., 2003, for a similar representation of the semi-Markov jump process introduced in Note 14.6.3.)

Example 11.9. (Continuation of Example 11.8) Within model $\mathcal{M}_k$, for the AR(k) parameters $\theta_1, \ldots, \theta_k$ and σ_k, the move in step a can be any Metropolis–Hastings step that preserves the posterior distribution (conditional on k), like a random walk proposal on the real and complex roots of the lag-polynomial. The remaining u_k can then be updated via an AR(1) move $u'_k = \lambda u_k + \sqrt{1-\lambda^2}\,\varepsilon$ if the proper stationary distribution $q_k(u_k)$ is a $\mathcal{N}(0,1)$ distribution. As noted in Brooks et al. (2003b), the ω_k's are not really necessary when the u_k's are independent. In the move from $\mathcal{M}_k$ to $\mathcal{M}_{k+1}$, Brooks et al. (2003b) suggest using (11.8), or a combination of the non-saturated approach with an auxiliary variable where $\theta_{k+1} = \sigma u_{k,k+1}$. If u_k is updated symmetrically, that is, if $u_{k,k+1} = \theta_{k+1}/\sigma$ when moving from $\mathcal{M}_{k+1}$ to $\mathcal{M}_k$, the algorithm provides the chain with some memory of previous values *within* each model and should thus facilitate moves *between* models. ||

The motivation of the saturation scheme by Brooks et al. (2003b) is not a simplification of the reversible jump algorithm, since the acceptance probabilities remain fundamentally the same in step c of Algorithm [A.49], but

[illegible]

[illegible]

1.3.2 Condensation Cross-Linking Processes

[illegible]

[illegible]

[illegible]

Algorithm A.50 —Continuous-Time MCMC—

At time t, for $\theta^{(t)} = (k, \theta_k)$

1. For $\ell \in \{k_1, \ldots, k_m\}$, generate $u_{k\ell} \sim \varphi_{k\ell}(u_{k\ell})$ so that

$$(\theta_\ell, u_{\ell k}) = T_{k\ell}(\theta_k, u_{k\ell})$$

and compute the intensity

$$\lambda_{k\ell} = \frac{L_\ell(\theta_\ell)\pi(\ell, \theta_\ell)}{L_k(\theta_k)\pi(k, \theta_k)} \frac{\varphi_{\ell k}(u_{\ell k})}{\varphi_{\ell k}(u_{k\ell})} \left| \frac{\partial T_{k\ell}(\theta_k, u_{k\ell})}{\partial(\theta_k, u_{k\ell})} \right| . \tag{11.10}$$

2. Compute the full intensity

$$\lambda = \lambda_{kk_1} + \cdots + \lambda_{kk_m} \qquad [A.50].$$

3. Generate the jumping time as $t + V$, with $V \sim \mathcal{E}xp(\lambda)$.
4. Select the jump move as k_i with probability λ_{kk_i}/λ.
5. Set time to $t = t + V$ and take $\theta^{(t)} = (k_i, \theta_{k_i})$.

Moves that update model parameters (or hyperparameters) without changing its dimension may also be incorporated as additional jump processes, that is, by introducing corresponding (fixed) intensities for these moves in the sum of Step 2. above. Note the strong similarity of (11.10) with the reversible jump acceptance probability (11.4). Recall that the intensity of an exponential distribution is the inverse expectation: this means that (11.10) is inversely proportional to the waiting time until the jump to (ℓ, θ_ℓ) and that this waiting time will be small if either $L_\ell(\theta_\ell)$ is large or if $L_k(\theta_k)$ is small. The latter case corresponds to a previous move to an unlikely value of θ.

The continuous time structure of [A.50] is rather artificial. Rather than being simulated as in Step 3., using a regular Rao–Blackwellization argument, it can just as well be implemented as a sequence of iterations, where each value (k, θ_k) is weighted by the average waiting time λ^{-1}.

The following example illustrates this generalization of Stephens (2000) in the setting of mixtures.

Example 11.10. (Continuation of Example 11.1.) For the following representation of the normal mixture model,

$$Y_t \sim \sum_{i=1}^{k} \omega_i \mathcal{N}(\mu_i, \sigma_i^2) \Big/ \sum_{i=1}^{k} \omega_i ,$$

Cappé et al. (2003) have developed split-and-combine moves for continuous time schemes, reproducing Richardson and Green (1997) within this context. The *split* move for a given component θ_ℓ of the k component vector θ_k is to split this component as to give rise to a new parameter vector with $k+1$ components, defined as

11.4 Problems

11.1 [illegible]

11.2 [illegible]

11.3 [illegible]

11.4 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Decade/Year	0	1	2	3	4	5	6	7	8	9
1850	–	4	5	4	1	0	4	3	4	0
1860	6	3	3	4	0	2	6	3	3	5
1870	4	5	3	1	4	4	1	5	5	3
1880	4	2	5	2	2	3	4	2	1	3
1890	2	2	1	1	1	1	3	0	0	1
1900	0	1	1	0	0	3	1	0	3	2
1910	2	0	1	1	1	0	1	0	1	0
1920	0	0	2	1	0	0	0	1	1	0
1930	2	3	3	1	1	2	1	1	1	1
1940	2	4	2	0	0	0	1	4	0	0
1950	0	1	0	0	0	0	0	1	0	0
1960	1	0	1	–	–	–	–	–	–	–

Table 11.2. Yearly number of mining accidents in England for the years 1851 to 1962, from Maguire et al. (1952) and Jarrett (1979) datasets. (*Source:* Carlin et al. 1992.)

Algorithm A.51 –Changepoint Poisson Model–

1. **Generate** $(1 \le k \le n)$

$$\tau \sim P(\tau = k) \propto \exp\left\{(\lambda - \theta)\sum_{i=1}^{k} t_i\right\} \exp\left\{\log(\theta/\lambda)\sum_{i=1}^{k} y_i\right\}.$$

2. **Generate** [A.51]

$$\theta \sim \mathcal{G}a\left(\alpha_1 + \sum_{i=1}^{\tau} y_i, \beta_1 + \sum_{i=1}^{\tau} t_i\right),$$

$$\lambda \sim \mathcal{G}a\left(\alpha_2 + \sum_{i=\tau+1}^{n} y_i, \beta_2 + \sum_{i=\tau+1}^{n} t_i\right).$$

3. **Generate**

$$\beta_1 \sim \mathcal{G}a(\delta_1 + \alpha_1, \theta + \varepsilon_1), \quad \beta_2 \sim \mathcal{G}a(\delta_2 + \alpha_2, \lambda + \varepsilon_2).$$

is a valid Gibbs sampler.

(c) Show that the model can also be represented by associating with each observation y_i a latent variable $Z_i \in \{0, 1\}$ which is the indicator of the change-point. Deduce that this representation does not modify the above algorithm.

(d) For the data in Table 11.2, which summarizes the number of mining accidents in England from 1851 to 1962, from 1851 to 1962, and for the hyperparameters $\alpha_1 = \alpha_2 = 0.5$, $\delta_1 = \delta_2 = 0$, and $\varepsilon_1 = \varepsilon_2 = 1$, apply the above algorithm to estimate the change-point. (*Note:* Raftery and Akman 1986 obtain estimates of k located between 1889 and 1892.)

11.18 Barry and Hartigan (1992, 1993) study another change-point model where the number r of change-points is random. They introduce the notion of a

11.21 A typical setup of *convolution* consists in the observation of

$$Y_t = \sum_{j=0}^{k} h_j X_{t-j} + \varepsilon_t\,, \qquad t = 1, \ldots, n,$$

where $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$, the X_t's are not observed but have finite support, $\mathcal{X} = \{s_1, \ldots, s_m\}$, and where the parameters of interest h_j are unknown. The discrete variables X_t are iid with distribution $\mathcal{M}_m(\theta_1, \ldots, \theta_m)$ on $\mathcal{X}$, the probabilities θ_i are distributed from a Dirichlet prior distribution, $\mathcal{D}_m(\gamma_1, \ldots, \gamma_m)$, and the coefficients h_j are from a conjugate distribution $h = (h_0, \ldots, h_k) \sim \mathcal{N}_{k+1}(\mu, \Sigma)$.

(a) Show that the completed likelihood is

$$L(x, h, \sigma) \propto \sigma^{-n} \exp\left\{ \frac{1}{2\sigma^2} \sum_{t=1}^{n} \left(y_t - \sum_{j=0}^{k} h_j x_{t-j} \right)^2 \right\}$$

and examine whether or not the EM algorithm can be implemented in this setting.

(b) Verify that the completion of Y_t in (Y_t, X_t) can be done as follows:

Algorithm A.52 –Deconvolution Completion–

1. **Generate** $(t = 1-k, \ldots, n)$

$$X_t \sim P(x_t = s_\ell) \propto \frac{\theta_\ell s_\ell}{\sigma^2} \exp\left\{ -s_\ell^2 \sum_{j=0}^{k} h_j^2 / 2\sigma^2 + \sum_{j=0}^{k} h_j \left(y_{t+j} - \sum_{\substack{i=0 \\ i \neq j}}^{k} h_i x_{t+j-i} \right) \right\}$$

and derive [A.52]

$$X = \begin{pmatrix} x_1 & x_0 & \ldots & x_{1-k} \\ & & \ldots & \\ x_n & x_{n-1} & \ldots & x_{n-k} \end{pmatrix}, \qquad \hat{h} = (X^t X)^{-1} X^t y\,,$$

where $y = (y_1, \ldots, y_n)$.

(c) Verify that the update of the parameters (h, σ^2) is given by

2. **Generate**

$$h \sim \mathcal{N}_{k+1}\left((\Sigma^{-1} + \sigma^{-2} X^t X)^{-1} (\Sigma^{-1}\mu + \sigma^{-2} X^t X \hat{h}), (\Sigma^{-1} + \sigma^{-2} X^t X)^{-1} \right)$$

$$\sigma^2 \sim \mathcal{IG}\left(\frac{n}{2}, \frac{1}{2} \sum_{t=1}^{n} \left(y_t - \sum_{j=0}^{k} h_j x_{t-j} \right)^2 \right),$$

while the step for hyperparameters θ_ℓ $(\ell = 1, \ldots, m)$ corresponds to

3. **Generate**

$$(\theta_1, \ldots, \theta_m) \sim \mathcal{D}\left(\gamma_1 + \sum_{t=1-k}^{n} \mathbb{I}_{x_t = s_1}, \ldots, \gamma_m + \sum_{t=1-k}^{n} \mathbb{I}_{x_t = s_m} \right)$$

[illegible]

[illegible]

11.22 [illegible]

[illegible]

11.5 Notes

11.5.1 Occam's Razor

[illegible]

[illegible]

[illegible]

12

Diagnosing Convergence

[illegible]

12.1 Stopping the Chain

[illegible]

Example 12.1. Probit [illegible] revisited. [illegible]

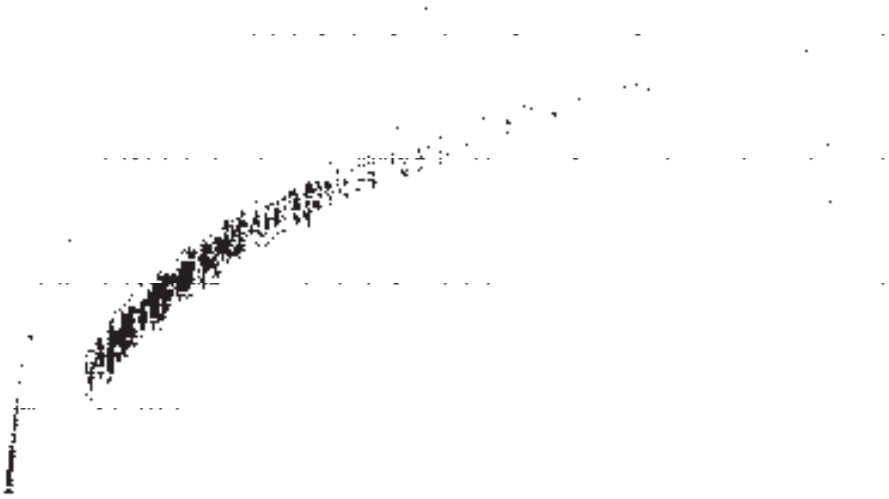


Fig. 12.1 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

12.1.1 Convergence Criteria

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Lemma 12.2. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

12.1.2 Markov Chains

12.1.3 Monitoring Reversibility

[illegible]

12.2 Monitoring Convergence to the Stationary Distribution

12.2.1 A First Illustration

[illegible]

[illegible]

[illegible]

Example 12.3. (Continuation of Example 12.1) [illegible]

12.2.2 Nonparametric Tests of Stationarity

[illegible]

Fig. 12.2.

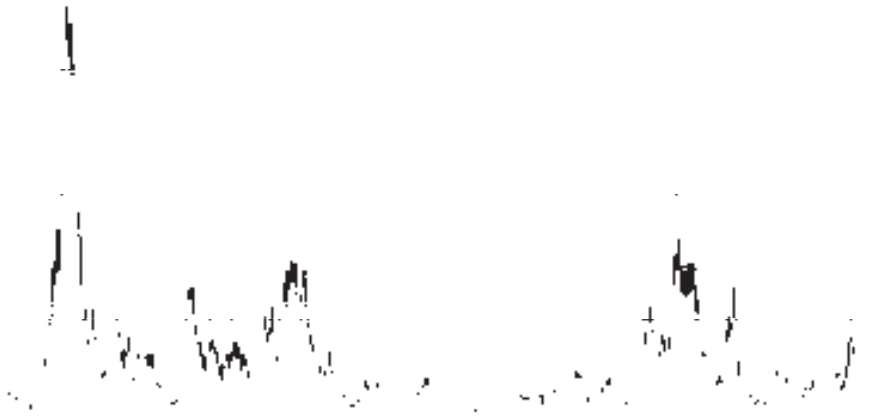


Fig. 12.3.

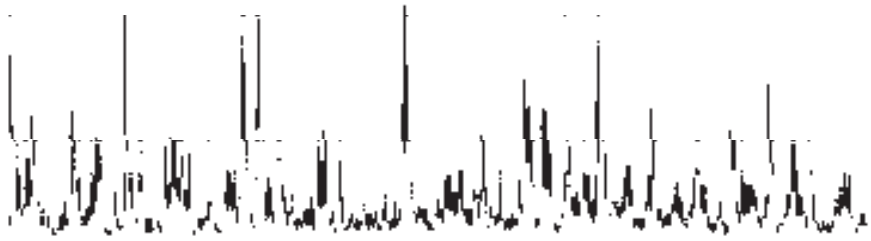
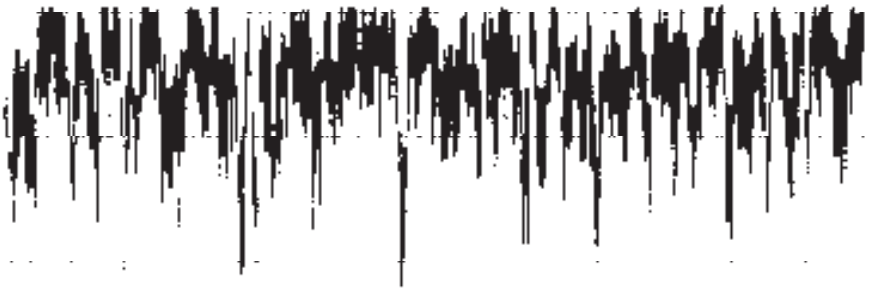


Fig. 12.4.



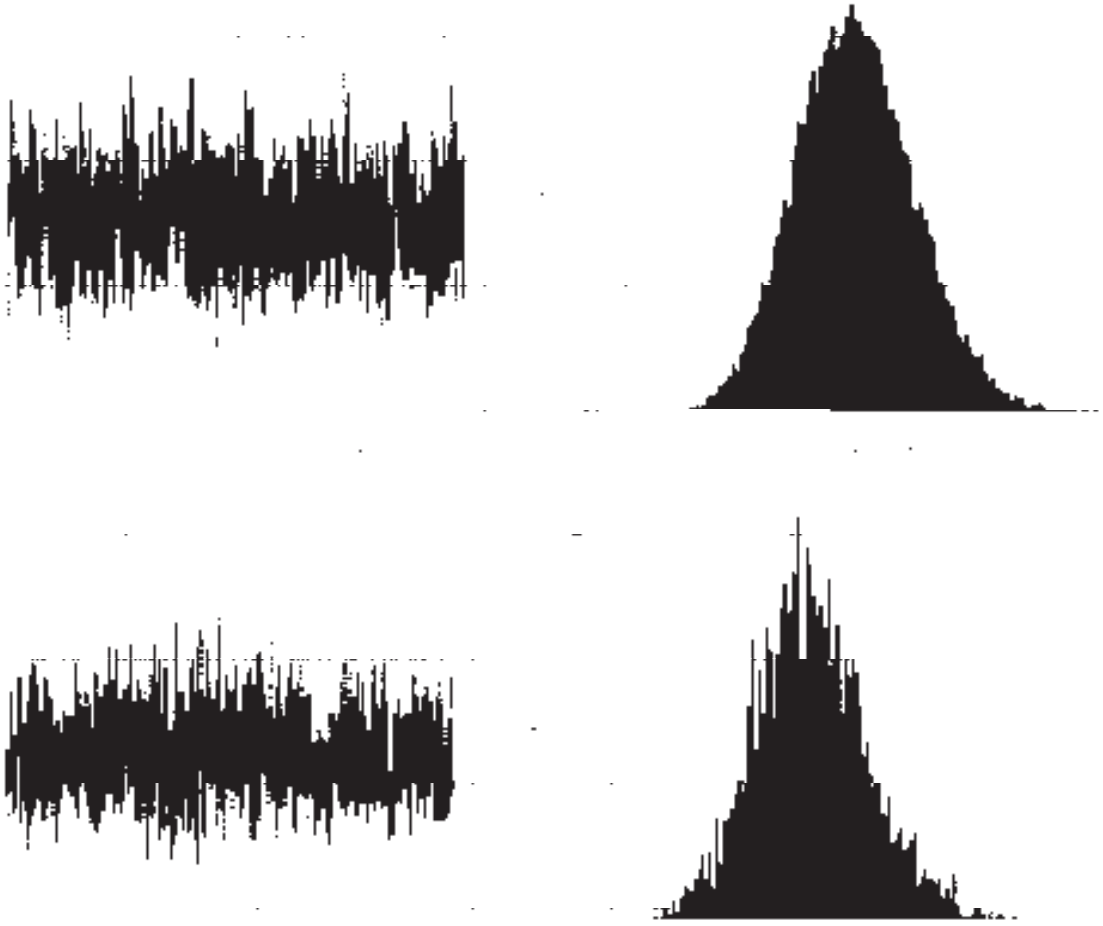


Fig. 12.7. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Fig. 12.8. [illegible]

[illegible]

Example 12.1. Nuclear power [illegible]

12.2.3 Regional Methods

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Lemma 12.5. [illegible]

[illegible]

[illegible]

[illegible]

Proof. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 12.6 (Continuation of Example 12.4) [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

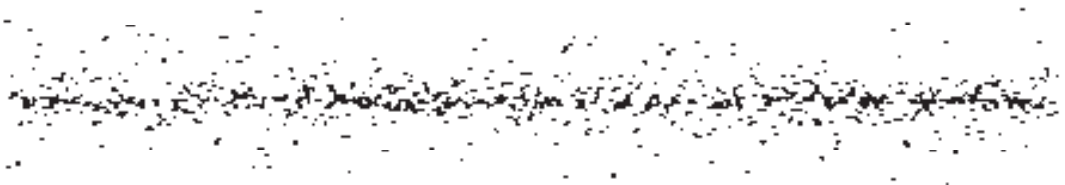


Fig. 12.4. [illegible]

[illegible]

[illegible]

12.2.1 Missing Mass

[illegible]

$$[illegible] \qquad / \qquad [illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 12.7. Binnsdle target. [illegible]

[illegible]

[illegible]

Fig. 2.10. [illegible]

[illegible]

[illegible]

[illegible]

Fig. 12.1. [illegible]

Example 12.8. Discounted mixture revisited. [illegible]

[illegible]

[illegible]

[

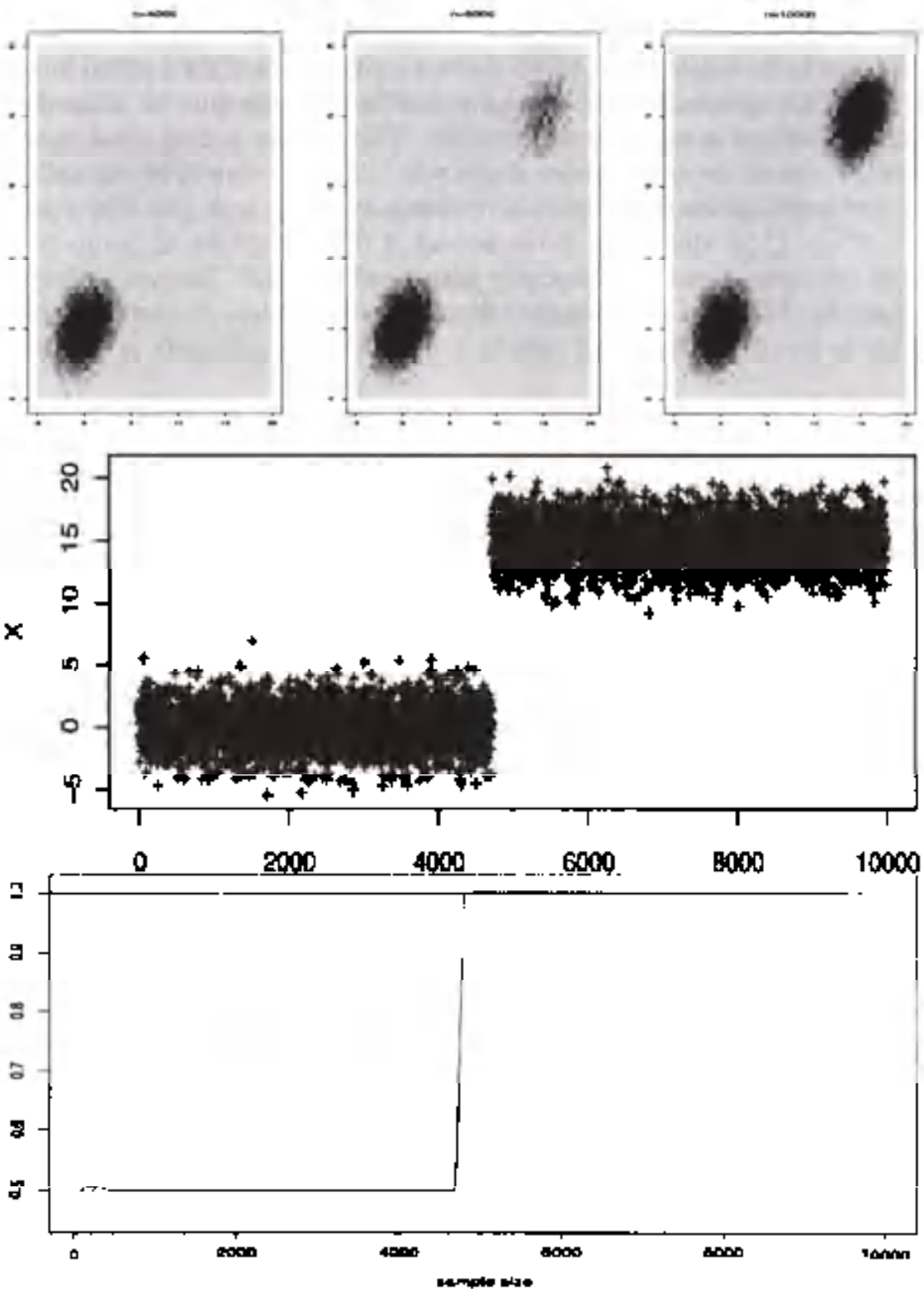


Fig. 12.12. *(top)* 2D histogram of the Markov chain of Example 12.8 after 4000, 6000 and 10,000 iterations; *(middle)* Path of the Markov chain for the first coordinate x; *(bottom)* Control curves for the bivariate mixture model, for the parameters $\mu = (0,0)$, $\nu = (15,15)$, $p = 0.5$, $\Sigma = \Sigma' = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}$. (*Source:* Philippe and Robert 2001.)

[illegible]

[illegible]

[illegible]

[illegible]

12.2.5 Distance Evaluations

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

[illegible]

A

12.3 Monitoring Convergence of Averages

12.3.1 A First Observation

[illegible]

Example [2.10]. Pathological Data Generator. [illegible]

$$[illegible] \tag{12.1}$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Fig. 12.13 [illegible]

Fig. 2. [illegible]

[illegible]

[illegible]

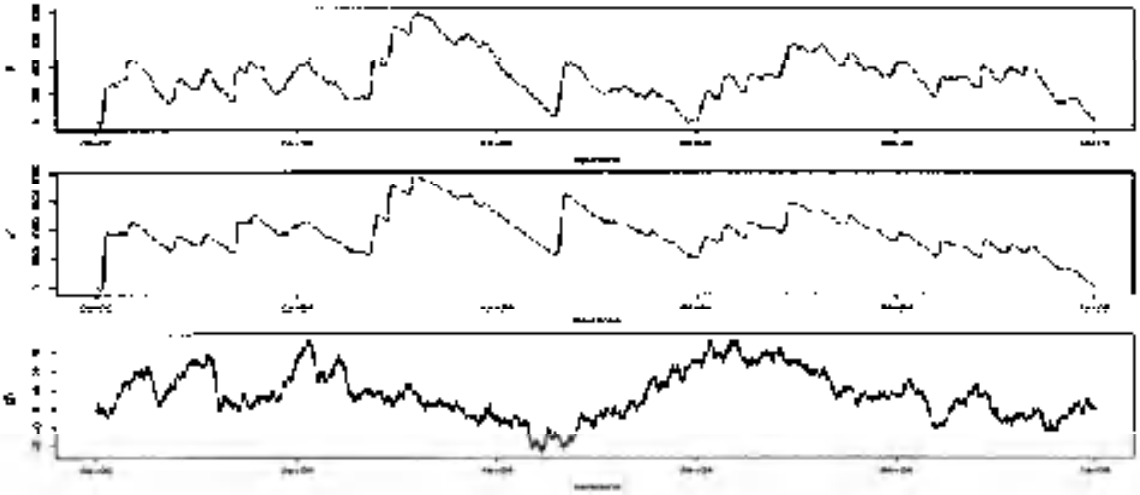


Fig. 12.15. Evolution of the CUSUM criterion D_t^i for the chains $\beta^{(t)}$ (top), $\sigma^{(t)}$ (middle) and $\beta^{(t)}/\sigma^{(t)}$ (bottom) of Example 12.1.

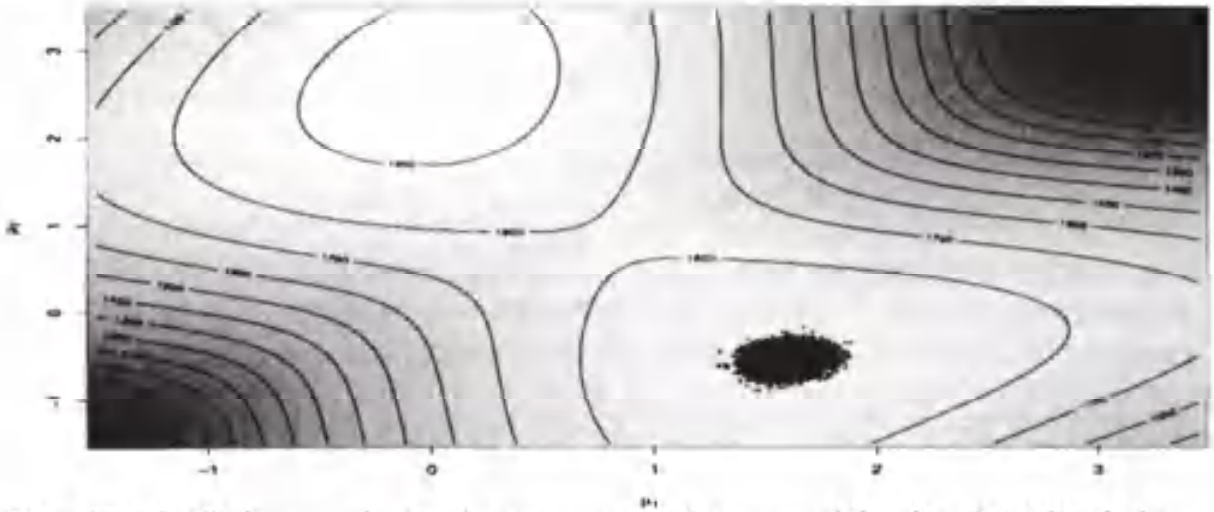


Fig. 12.16. Gibbs sample for the two mean mixture model, when initialized close to the second and lower mode, for true values $\mu_1 = 0$ and $\mu_2 = 2.5$, over the log-likelihood surface.

sum indicator (12.9) which "estimates" 1 and can detect probability losses in the region covered by the sample, and the related method developed in Brooks 1998a.)

12.3.2 Multiple Estimates

In most cases, the graph of the raw sequence $(\theta^{(t)})$ is unhelpful in the detection of stationarity or convergence. Indeed, it is only when the chain has explored different regions of the state-space during the observation time that lack of stationarity can be detected (Gelman and Rubin 1992 also illustrate the use of the raw sequence graphs when using chains in parallel with quite distinct supports.)

Given some quantities of interest $\mathrm{E}_f[h(\theta)]$, a more helpful indicator is the behavior of the averages (12.13) in terms of T. A necessary condition

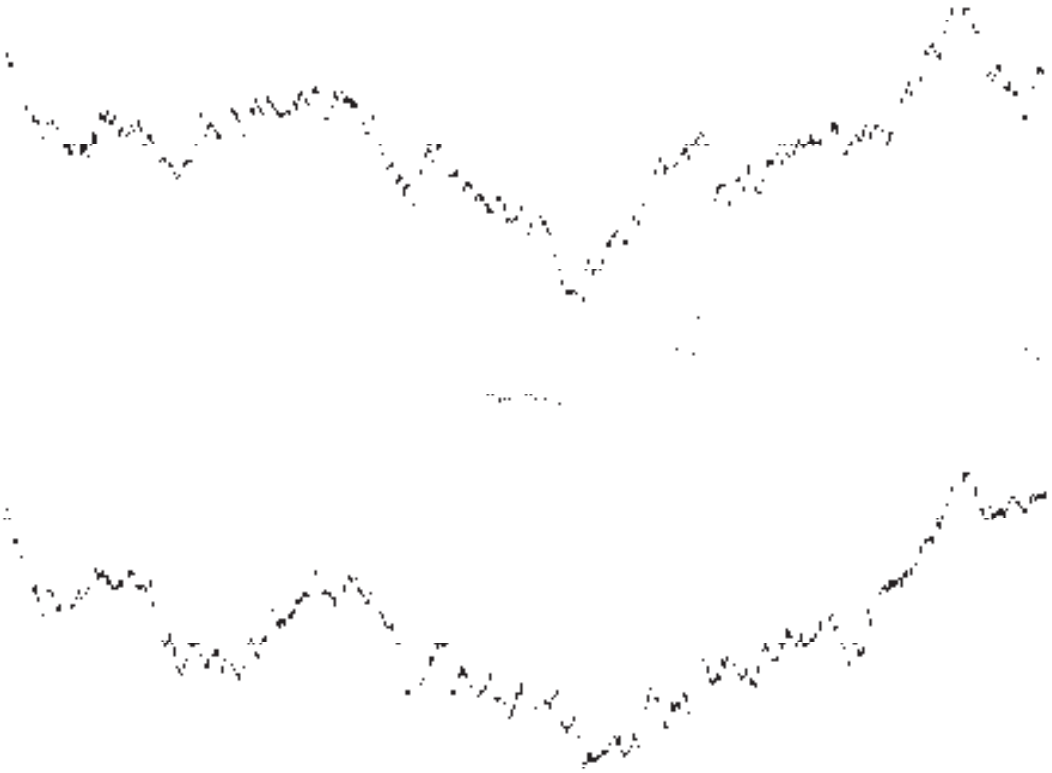


Fig. 12.17 [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Lemma 12.1. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

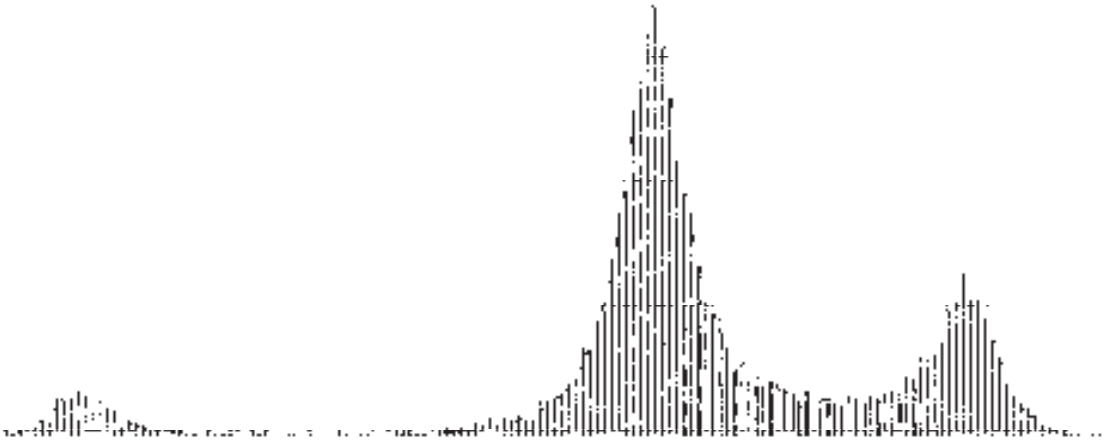
$$[illegible] \sum [illegible]$$

[illegible]

$$[illegible] \sum_{[illegible]}^{[illegible]} [illegible]$$

[illegible]

Example 12.12 (Gaussian posterior distribution). [illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 12.13. (Continuation of Example 12.10) [illegible]

[illegible]

Figure 2.19. [illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Fig. 12.201 [illegible]

[illegible]

12.3.3 Renewal Theory

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$\frac{1}{n}\sum_{[illegible]} [illegible]$$

[illegible]

$$[illegible] \sum_{[illegible]} [illegible] \sum_{[illegible]} [illegible]$$

[illegible]

Proposition 12.14 [illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Example 12.15. (Continuation of Example 12.12.) [illegible]

$x_2 \in [r_1, r_2]$, $x_1 < r_1$, and $x_3 > r_2$ (if $x_1 < x_2 < x_3$). This choice leads to the following bounds:

$$\rho_{11} = r_1 - x_1 < |\theta - x_1| < \rho_{12} = r_1 - x_1,$$

$$0 < |\theta - x_2| < \rho_{22} = \max(r_2 - x_2, x_2 - r_1),$$

$$\rho_{31} = x_3 - r_2 < |\theta - x_3| < \rho_{32} = x_3 - r_1,$$

for $\theta \in C$, which give ε and ν. In fact,

$$\begin{aligned} K(\theta,\theta') \geq \int_{\mathbb{R}^3_+} & \frac{1}{\sqrt{2\pi}}\, \tau(\eta_1,\eta_2,\eta_3)^{-1} \\ & \times \exp\left\{-(\theta' - \mu(\eta_1,\eta_2,\eta_3))^2 / (2\tau^2(\eta_1,\eta_2,\eta_3))\right\} \\ & \times \frac{1+\rho_{11}^2}{2} \exp\{-(1+\rho_{12}^2)\eta_1/2\} \frac{1}{2} \exp\{-(1+\rho_{22}^2)\eta_2/2\} \\ & \times \frac{1+\rho_{31}^2}{2} \exp\{-(1+\rho_{32}^2)\eta_3/2\}\, d\eta_1 d\eta_2 d\eta_3 \\ - \frac{1+\rho_{11}^2}{1+\rho_{12}^2} & \frac{1}{1+\rho_{22}^2} \frac{1+\rho_{31}^2}{1+\rho_{32}^2}\, \nu(\theta') - \varepsilon\nu(\theta')\,, \end{aligned}$$

where ν is the marginal density of θ for

$$\begin{aligned} (\theta,\eta_1,\eta_2,\eta_3) \sim & \mathcal{N}\left(\mu(\eta_1,\eta_2,\eta_3), \tau^2(\eta_1,\eta_2,\eta_3)\right) \\ & \times \mathcal{E}xp\left(\frac{1+\rho_{12}^2}{2}\right) \mathcal{E}xp\left(\frac{1+\rho_{22}^2}{2}\right) \mathcal{E}xp\left(\frac{1+\rho_{32}^2}{2}\right)\,. \end{aligned}$$

One can thus evaluate the frequency $1/\mu_C$ of visits to C and calibrate r_1 and r_2 to obtain optimal renewal probabilities. ||

Another difficulty with this method is that the splitting technique requires the generation of variables distributed as

$$R(\theta^{(t)},\xi) = \frac{K(\theta^{(t)},\xi) - \varepsilon\nu(\xi)}{1-\varepsilon}\,, \tag{12.21}$$

called the *residual kernel*. It is actually enough to know the ratio $K(\theta^{(t)},\xi)/\varepsilon\nu(\xi)$ since, following from Lemma 2.27, the algorithm

Algorithm A.53 –Negative Weight Mixture Simulation–

1. Simulate $\xi \sim K(\theta^{(t)},\xi)$.
2. Accept ξ with probability $1 - \dfrac{\varepsilon\nu(\xi)}{K(\theta^{(t)},\xi)}$, [A.53]
   otherwise go to 1.

[illegible]

[illegible]

Example 12.16. (Generalization of Example 12.15) [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Table 12.1. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 12.17. A finite Markov chain. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Table 12.2. [illegible]

[illegible]

Example 12.18. Grouped [illegible] [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]

Table 12.3. [illegible]

12.3.4 Within and Between Variances

[illegible]

$$[illegible]$$

$$[illegible]$$

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

Fig. 12.21. [illegible]

[illegible]

[illegible]

[illegible]

Example 12.13. Continuation of Example 12.12: [illegible]

[illegible]

[illegible]

[illegible]

12.3.5 Effective Sample Size

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

12.4 Simultaneous Monitoring

12.4.1 Binary Control

[illegible]

$$[illegible]$$

[illegible]

[illegible]

Table 12.4: [illegible]

[illegible]				
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

[illegible]

[illegible]

Example 12.203. (Continuation of Example 12.13) [illegible]

[illegible]

[illegible]

Fig. 12.22. [illegible]

[illegible]

12.1.2 Void Discretization

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

Fig. 12.23. [illegible]

[illegible]

Example 12.21. (Continuation of Example 12.12): [illegible]

[illegible]

12.5 Problems

12. [illegible]

[illegible]

12.2

12.3

12.4

12.5

12.7

[illegible]

[illegible]

12.8 [illegible]

[illegible]

12.9 [illegible]

[illegible]

12.10 [illegible]

12.11 [illegible]

[illegible]

9.12 [illegible]

[illegible]

12.13 [illegible]

12.14 [illegible]

[illegible]

12.15 [illegible]

12.17

12.18

12.19

12.20

12.21

12.22

12.23

[illegible]

12.24 [illegible]

[illegible]

12.6 Notes

12.6.1 Spectral Analysis

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

13

Perfect Sampling

[illegible]

13.1 Introduction

[illegible]

[illegible]

[illegible]

[illegible]

13.2 Coupling, from the Past

[illegible]

[illegible]

[illegible]

13.2.1 Random Mappings and Coupling

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 2.3.1 Consider [illegible]

[illegible]

[illegible]

[illegible]

Although [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Fig. [illegible] [illegible]

[illegible]

[illegible]

[illegible]

Theorem [illegible] [illegible]

(i) $\theta_0^{(j)} = j$;
(ii) $\theta_{t+1}^{(j)} = \Psi(x_t^{(j)}, u_{t+1})$, *where the* U_t*'s are mutually independent.*

Then the time τ *to coalescence is a random variable that depends only on* $U_1, U_2, \ldots$.

An interpretation of the coalescence time τ is that it is the time at which the initial state of the chain has "worn off": the distribution of $\theta_\tau^{(j)}$ is obviously the same for all j's. A natural subsequent question is whether or not $\theta_\tau^{(j)}$ is a draw from the stationary distribution π. While it is the case that, for a *fixed* time $T^* > \tau$, $\theta_{T^*}^{(j)}$ is distributed from the stationary distribution, the fact that τ is a stopping time implies that $\theta_\tau^{(j)}$ is usually *not* distributed from π, as shown in Example 13.4 (see also Problems 13.1 and 13.2).

13.2.2 Propp and Wilson's Algorithm

The implementation of the above principle thus reduces to a search for a constant composition $\Psi_0 \circ \ldots \circ \Psi_T$: once one such $T < 0$ is found, coalescence occurs between time T and time 0, and the common value of the chains at time 0 is distributed from π. The algorithm, called *coupling from the past*, thus reads as follows:

Algorithm A.54 –Coupling from the Past–

1. Generate random mappings $\Psi_{-1}, \Psi_{-2}, \ldots$.
2. Define compositions ($t = -1, -2, \ldots$)

$$\Phi_t(\theta) = \Psi_0 \circ \Psi_{-1} \circ \ldots \circ \Psi_{-t}(\theta).$$

3. Determine T such that Φ_T is constant by looking successively at $\Phi_{-1}, \Phi_{-2}, \Phi_{-4}, \ldots$
4. For an arbitrary value θ_0, take $\Phi_T(\theta_0)$ as a realization from π.

In Step 3 of this algorithm, the backward excursion stops[3] at one time T when Φ_T is a constant function; we are thus spared the task of running the algorithm until $-\infty$! The same applies to the simulation of the random mappings Ψ_t, which only need to be simulated till time T.

Note that, by virtue of the separate definitions of the maps Ψ_t, for a given θ_T, the value of $\theta^{(0)}$ is also given: not only does this save simulation time but, more importantly, it ensures the validity of the method.

Theorem 13.3. *The algorithm [A.54] produces a random variable distributed exactly according to the stationary distribution of the Markov chain.*

[3] Note that to establish that Φ_T is constant is formally equivalent to consider all possible starting values at time T and to run the corresponding coupled chains till time 0.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 13.5. (Continuation of Example 13.1). Recall that the transition kernel on the (sub-)chain X_t is

$$K(x_t, x_{t+1}) \propto \binom{n}{x_{t+1}} \frac{\Gamma(\alpha + x_t + x_{t+1})\Gamma(\beta + n - x_t - x_{t+1})}{\Gamma(\alpha + x_t)\Gamma(\beta + n - x_t)}.$$

For $x_1 > x_2$,

$$\begin{aligned}\frac{K(x_1, y)}{K(x_2, y)} &= \frac{\Gamma(\alpha + x_1 + y)\Gamma(\beta + n - x_1 - y)\Gamma(\alpha + x_2)\Gamma(\beta + n - x_2)}{\Gamma(\alpha + x_2 + y)\Gamma(\beta + n - x_2 - y)\Gamma(\alpha + x_1)\Gamma(\beta + n - x_1)} \\ &= \frac{\Gamma(\alpha + x_2)\Gamma(\beta + n - x_2)}{\Gamma(\alpha + x_1)\Gamma(\beta + n - x_1)} \frac{(\alpha + x_1 + y - 1)\cdots(\alpha + x_2 + y)}{(\beta + n - x_2 - y - 1)\cdots(\beta + n - x_1 - y)}\end{aligned}$$

is increasing in y. This monotone ratio property implies that the transition is monotone for the usual ordering on $\mathbb{N}$ and therefore that $\bar{0} = 0$ and that $\bar{1} = n$. It is thus sufficient to monitor the chains started from 0 and from n to check for coalescence, for any value of n. This property can be observed in Figure 13.1. ‖

Example 13.6. Mixtures of distributions. Consider the simplest possible mixture structure

$$pf_0(x) + (1-p)f_1(x), \tag{13.2}$$

where both f_0 and f_1 are known, with a uniform prior on $0 < p < 1$. For a given sample $x_1, \ldots, x_n$, the corresponding Gibbs sampler is based on the following steps at iteration t.

1. Generate n iid $\mathcal{U}(0,1)$ random variables $u_1^{(t)}, \ldots, u_n^{(t)}$.
2. Derive the indicator variables $z_i^{(t)}$ as $z_i^{(t)} = 0$ iff

$$u_i^{(t)} \leq \frac{p^{(t-1)} f_0(x_i)}{p^{(t-1)} f_0(x_i) + (1 - p^{(t-1)}) f_1(x_i)}$$

and compute

$$m^{(t)} = \sum_{i=1}^{n} z_i^{(t)}.$$

3. Simulate $p^{(t)} \sim Be(n + 1 - m^{(t)}, 1 + m^{(t)})$.

To design a coupling strategy and a corresponding CFTP algorithm, we first notice that $m^{(t)}$ can only take $n+1$ possible values, namely, $0, 1, \ldots, n$. We are thus de facto in a *finite* setting.

If we recall that a $Be(m+1, n-m+1)$ random variable can be represented as the ratio

[illegible]

Fig. [illegible] [illegible]

[illegible]

[illegible]

[illegible]

it nonetheless remains valid: the value produced at time 0 is distributed from π. Besides, as we will see below in Example 13.7, it may also save computing time. Note also that the upper and lower bounding sequences $\underline{\theta}_t$ and $\overline{\theta}_t$ need not necessarily evolve within $\mathcal{X}$ at every step. This is quite a useful feature when working on complex constrained spaces.

Example 13.7. (Continuation of Example 13.6.) Consider the extension of (13.2) to the 3-component case

$$p_1 f_1(x) + p_2 f_2(x) + p_3 f_3(x)\,, \qquad p_1 + p_2 + p_3 = 1\,, \tag{13.4}$$

where f_1, f_2, f_3 are known densities. If we use a (flat) Dirichlet $\mathcal{D}(1,1,1)$ prior on the parameter (p_1, p_2, p_3), an associated Gibbs sampler for a sample $x_1, \ldots, x_n$ from (13.4) is as follows.

1. Generate $u_1, \ldots, u_n \sim \mathcal{U}(0,1)$.
2. Take

$$n_1 = \sum_{i=1}^{n} \mathbf{I}\left(u_i \le \frac{p_1 f_1(x_i)}{p_1 f_1(x_i) + p_2 f_2(x_i) + p_3 f_3(x_i)}\right),$$

$$n_2 = \sum_{i=1}^{n} \left\{ \mathbf{I}\left(u_i > \frac{p_1 f_1(x_i)}{p_1 f_1(x_i) + p_2 f_2(x_i) + p_3 f_3(x_i)}\right) \times \mathbf{I}\left(u_i \le \frac{p_1 f_1(x_i) + p_2 f_2(x_i)}{p_1 f_1(x_i) + p_2 f_2(x_i) + p_3 f_3(x_i)}\right)\right\},$$

and $n_3 = n - n_1 - n_2$.

3. Generate $(p_1, p_2, p_3) \sim \mathcal{D}(n_1+1, n_2+1, n_3+1)$.

Once again, since (n_1, n_2) can take only a finite number of values, the setting is *finite* and we can monitor all possible starting values of (n_1, n_2) to run our CFTP algorithm. However, there are $(n+2)(n+1)/2$ such values. The computational cost is thus in $\mathrm{O}(n^2)$, rather than $\mathrm{O}(n)$ for the two component case. And there is no obvious monotonicity structure, even when considering the exponential representation of the Dirichlet distribution:

$$\left(\frac{\sum_{i=1}^{n_1+1} \omega_{1i}}{\sum_{j=1}^{3}\sum_{i=1}^{n_j+1} \omega_{ji}}, \frac{\sum_{i=1}^{n_2+1} \omega_{2i}}{\sum_{j=1}^{3}\sum_{i=1}^{n_j+1} \omega_{ji}}, \frac{\sum_{i=1}^{n_3+1} \omega_{3i}}{\sum_{j=1}^{3}\sum_{i=1}^{n_j+1} \omega_{ji}}\right) \sim \mathcal{D}(n_1+1, n_2+1, n_3+1)$$

with

$$\omega_{11}, \ldots, \omega_{1(n+1)}, \omega_{21}, \ldots, \omega_{2(n+1)}, \omega_{31}, \ldots, \omega_{3(n+1)} \sim \mathcal{E}xp(1)\,.$$

While looking only at the boundary cases $n_i = 0$ $(i = 1,2,3)$ seemed to be the natural generalization of the case with 2 components, Hobert et al. (1999) exhibited occurrences where coalescence of the entire range of chains had not occurred at coalescence of the boundary chains.

[illegible]

[illegible]

[illegible]

Fig. 13.3. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

13.2.1 Continuous State Spaces

[illegible]

First, as seen in Examples 13.1 and 13.6, it is possible to devise perfect sampling algorithms in data augmentation settings (see Section 9.2.1), where one of the two chains has a finite support. In Example 13.1, once X is produced by the perfect sampling algorithm, the continuous variable θ can be generated from the conditional $\mathcal{B}e(\alpha + x, \beta + n - x)$. (This is another application of the duality principle given in Section 9.2.3.)

A more general approach to perfect sampling in the continuous case relies on the *Kac's representation of stationary distributions* (Section 6.5.2)

$$\pi(A) = \sum_{t=1}^{\infty} \Pr(N_t \in A) \Pr(T^* = t),$$

where the kernel satisfies a minorizing condition $K(x,y) \geq \varepsilon\nu(y)\mathbb{I}_C(x)$, τ_α is the associated renewal stopping rule, T^* is the *tail renewal time*

$$\Pr(T^* = t) = \frac{\Pr_\alpha(\tau_\alpha \geq t)}{\mathbb{E}_\alpha(\tau_\alpha)},$$

and N_t has the same distribution as X_t, given $X_1 \sim \nu(\cdot)$ and *no regeneration before time* t.

This was first applied by Murdoch and Green (1998) to uniformly ergodic chains (for example, Metropolis–Hastings algorithms with atoms, including the independent case (see Section 12.2.3)). When the kernel is uniformly ergodic, *Doeblin's condition* holds (Section 6.59) and the kernel satisfies $K(x,y) \geq \varepsilon\nu(y)$ for all $x \in \mathcal{X}$. Kac's representation then simplifies to

$$\pi(x) = \sum_{i=0}^{+\infty} \varepsilon(1-\varepsilon)^i (\tilde{K}^i \nu)(x),$$

where $\tilde{K}$ denotes the residual kernel

$$\tilde{K}(x,y) = \frac{K(x,y) - \varepsilon\nu(x)}{1-\varepsilon}.$$

The application of this representation to perfect sampling is then obvious: the backward excursion time T is distributed from a geometric distribution $\mathcal{G}eo(\varepsilon)$ and a *single* chain needs to be simulated forward from the residual kernel T times. The algorithm thus looks as follows.

Algorithm A.55 –Kac's Perfect Sampling–

1. Simulate $x_0 \sim \nu$, $\omega \sim \mathcal{G}eo(\varepsilon)$.
2. Run the transition $x_{t+1} \sim \tilde{K}(x_t, y)$ $t = 0, \cdots, \omega - 1$, and take x_ω as a realization from π.

[illegible]

[illegible]

Lemma 13.8. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Lemma 13.9. [illegible]

[illegible]

[illegible]

[illegible]

13.2.5 Proof: Step Stamping

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

13.3 [illegible]

[illegible]

[illegible]

[illegible]

Lemma 13.10. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 3.11. Truncated normal distributions. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

$$\pi(z|\mu,\Sigma) \propto \exp\left\{-(h^{-1}(z)-\mu)^{\mathrm{T}}\Sigma^{-1}(h^{-1}(z)-\mu)/2\right\} \prod_{i=1}^{p} \frac{1}{(1-z_i)^2}\,.$$

With this new parameterization, $\bar{\mathbf{1}}$ has to be found, analytically or numerically (because of the additional Jacobian term in the density), while $\bar{\mathbf{0}} = (1,\ldots,1)$. In the special case when $\mu = (-2.4, 1.8)$ and

$$\Sigma = \begin{pmatrix} 1.0 & -1.2 \\ -1.2 & 4.4 \end{pmatrix}.$$

$\bar{\mathbf{1}} = (0, 1.8/2.8)$. See Philippe and Robert (2003) for the implementation details of perfect sampling in this setting. ‖

When π is bounded and the maximum value $\bar{\mathbf{1}} = \arg\max \pi(\omega)$ is known (see Problem 13.14 for an extension), an associated perfect sampling algorithm thus reads as follows.

Algorithm A.56 –Perfect Monotone Slice Sampler–

At time $-T$

1. Start from $\bar{\mathbf{0}}$.
2. Generate $u_{-T} \sim \mathcal{U}([0,1])$.
3. Generate $\omega_0^{(-T)}$ uniformly over the parameter set.
4. Generate the images $\omega_0^{(-t)}$ as $t = -T+1,\ldots,0$

$$\mathcal{U}\left(\{\omega;\, \pi(\omega) \geq u_{-t}\pi(\omega_0^{(-t-1)})\}\right).$$

5. Set $\omega_1^{(-T-1)} = \bar{\mathbf{1}}$.
6. For $t = -T,\ldots,0$,
   if $\omega_0^{(-t)}$ satisfies

$$\pi(\omega_0^{(-t)}) \geq u_{-t}\pi(\omega_1^{(-t-1)}),$$

   there is coalescence at time 0;
   if not, generate $\omega_1^{(-t)}$ as

$$\mathcal{U}\left(\{\omega;\, \pi(\omega) \geq u_{-t}\pi(\omega_1^{(-t-1)})\}\right).$$

7. If there is no coalescence at time 0, increment T.

Example 13.12. Exponential mixtures For an exponential mixture

$$p\mathcal{E}xp(\lambda_0) + (1-p)\mathcal{E}xp(\lambda_1),$$

where $0 < p < 1$ is known, a direct implementation of slice sampling is quite difficult, since simulation from

[illegible]

[illegible]

[illegible]

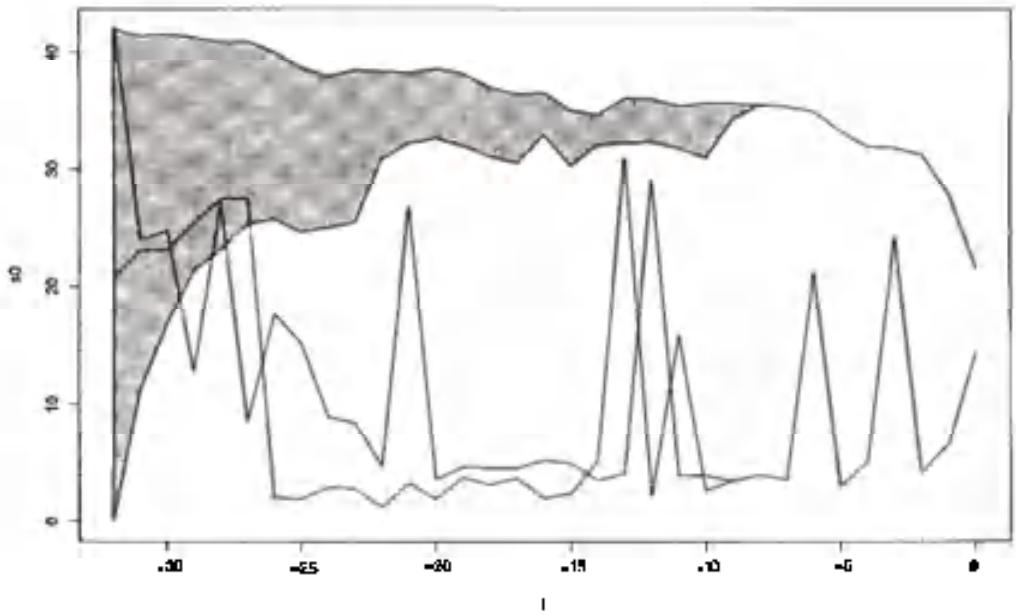


Fig. 13.4. Coupling from the past on the distribution (13.7) for a fixed $n_0 = 35$, for $n = 40$ simulated observations, with the sequence of $s_0^{(t)}$'s for both the upper and the lower chains, and the range of the marginal posterior in the background. (*Source:* Casella et al. 2002.)

13.2.6 Perfect Sampling via Automatic Coupling

In addition to slice sampling, there is another approach to perfect sampling, which is interesting from a theoretical point of view, as it highlights the connection with Accept–Reject algorithms. It is based on Breyer and Roberts (2000a) automatic coupling.

We have two chains $(\omega_0^{(t)})$ and $(\omega_1^{(t)})$, where the first chain, $(\omega_0^{(t)})$, is run following a standard Metropolis–Hastings scheme with proposal density $q(\omega'|\omega)$ and corresponding acceptance probability

$$\frac{\pi(\omega')\,q(\omega_0^{(t)}|\omega')}{\pi(\omega_0^{(t)})\,q(\omega'|\omega_0^{(t)})} \wedge 1\,.$$

The transition for the second chain, $(\omega_1^{(t)})$, is based on the same proposed value ω', with the difference being that the proposal distribution is *independent* of the second chain and is associated with the acceptance probability

$$\frac{\pi(\omega')\,q(\omega_1^{(t)}|\omega_0^{(t)})}{\pi(\omega_1^{(t)})\,q(\omega'|\omega_0^{(t)})} \wedge 1\,.$$

If ω' is *not accepted* by the second chain, it then stays the same, $\omega_1^{(t+1)} = \omega_1^{(t)}$. Note that, keeping in mind the perfect sampling perspective, we can use the same uniform variable u_t for both transitions, thus ensuring that the chains coalesce once they both accept a proposed value ω'.

[illegible]

[illegible]

[illegible]

Fig. 13.5. [illegible]

[illegible]

[illegible]

[illegible]

13.3 Forward Coupling

[illegible]

[illegible]

Markov chain *backward* to time 0 to see if this value is acceptable. It relies on the availability of the *reverse kernel* $\tilde{K}$, namely,

$$\tilde{K}(\omega,\omega') = \frac{\pi(\omega')K(\omega',\omega)}{\pi(\omega)}, \tag{13.9}$$

which corresponds to running the associated Markov chain backward in time.[7] It also requires simulation of the variable U in the random mapping Ψ conditional on its image, that is, conditional on $\omega' = \Psi(\omega, u)$ for fixed ω and ω'.

Let $g_\Psi(\cdot|\omega,\omega')$ denote the conditional distribution of U. A first version of this algorithm for a finite state space is the following.

Algorithm A.57 –Fill's Perfect Sampling Algorithm–

1. Choose a time T and a state $x_T = z$.
2. Generate $X_{T-1}|x_T$, $X_{T-2}|x_{T-1}$, ..., $X_0|x_1$ from the reverse kernel $\tilde{K}$.
3. Generate $U_1 \sim g_\Psi(u|x_0,x_1)$, $U_2 \sim g_\Psi(u|x_1,x_2)$, ..., $U_T \sim g_\Psi(u|x_{T-1},x_T)$ and derive the random mappings $\Psi_1 = \Psi(u_1,\cdot),\ldots,\Psi_T = \Psi(u_T,\cdot)$.
4. Check whether the composition $\Psi_1 \circ \cdots \circ \Psi_T$ is constant
5. If so, then accept x_0 as a draw from π.
6. Otherwise begin again, possibly with new values of T and z.

Lemma 13.14. *The algorithm [A.57] returns an exact simulation from the stationary distribution π.*

There are several ways to prove that [A.57] is correct. Problem 13.17 gives one proof, but perhaps the most intuitive approach is to establish that [A.57] is, in essence, a rejection algorithm.

Proof. Let $C_T(z)$ denote the event that all chains have coalesced and are in state z at time T. The algorithm [A.57] is a rejection algorithm in that it generates $X_0 = x$, then accepts x as a draw from π if $C_T(z)$ has occurred. The associated proposal distribution is the T-step transition density $\tilde{K}^T(z,\cdot)$. The validity of the algorithm is established if x is accepted with probability

$$\frac{1}{M}\frac{\pi(x)}{\tilde{K}^T(z,x)}, \text{ where } M \geq \sup_x \frac{\pi(x)}{\tilde{K}^T(z,x)}.$$

From (13.9), $\pi(x)/\tilde{K}^T(z,x) = \pi(z)/K^T(x,z)$. In addition, $P[C_T(z)] \leq K^T(x',z)$ for any x', since the probability of coalescence is smaller than the probability that the chain reaches z starting in x'. Hence $P[C_T(z)] \leq \min_{x'} K^T(x',z)$ and we have the bound

[7] If detailed balance (Definition 6.45) holds, $\tilde{K} = K$.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Example 13.15. Continuation of Example 13.11. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

13.1 Perfect Sampling in Practice

13.5 Problems

13.1 [illegible]

13.2 [illegible]

$$[illegible]$$

[illegible]

13.3 [illegible]

[illegible]

13.4 [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

13.5 [illegible]

$$[illegible]$$

[illegible]

[illegible]

13.12 [illegible]

13.13 [illegible]

[illegible]

13.16 [illegible]

[illegible]

13.17 [illegible]

[illegible]

[illegible]

13.6 Notes

13.6.1 History

[illegible]

13.6.2 Perfect Sampling and Tempering

[illegible]

$$[illegible] \sum [illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Example 13.16. Power tempering. [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

Algorithm A.58 –Tempered Perfect Sampling–

1. Go backward far enough to find $\tau^* < 0$ such that $D_{\tau^*} = N^*$ and $D_t = 0$ for at least one $\tau^* < t \leq 0$.
2. Simulate the coupled chain (θ_t, N_t) τ^* steps, with
$$(\theta_{\tau^*}, N_{\tau^*}) = (0,0)\,.$$
3. Take
$$(\theta_0, N_0) \sim \mu\,.$$

The validity of this algorithm follows from the monotonicity lemma above, in that the (virtual) chains starting from all possible values in $\{0, 1, \ldots, N^*\}$ have coalesced by the time the chain starting from the largest value N^* takes the lowest value 0. Note also that $(0,0)$ is a particular renewal event for the chain (θ_t, N_t).

Example 13.18. Flour beetle mortality. As described in Carlin and Louis (2001), a logit model is constructed for a flour beetle mortality dataset, corresponding to 8 levels of dosage ω_i of an insecticide, with n_i exposed beetles and $y_i n_i$ killed beetles. The target density is

$$\pi(\mu, \sigma, \nu) \propto \pi(\mu, \sigma, \nu) \prod_{i=1}^{8} \left\{ \frac{\exp y_i \frac{\omega_i - \mu}{\sigma}}{1 + \exp \frac{\omega_i - \mu}{\sigma}} \right\}^{n_i \nu}\,,$$

with the prior density $\pi(\mu, \sigma, \nu)$ corresponding to a normal distribution on μ, an inverted gamma distribution on σ^2 and a gamma distribution on ν.

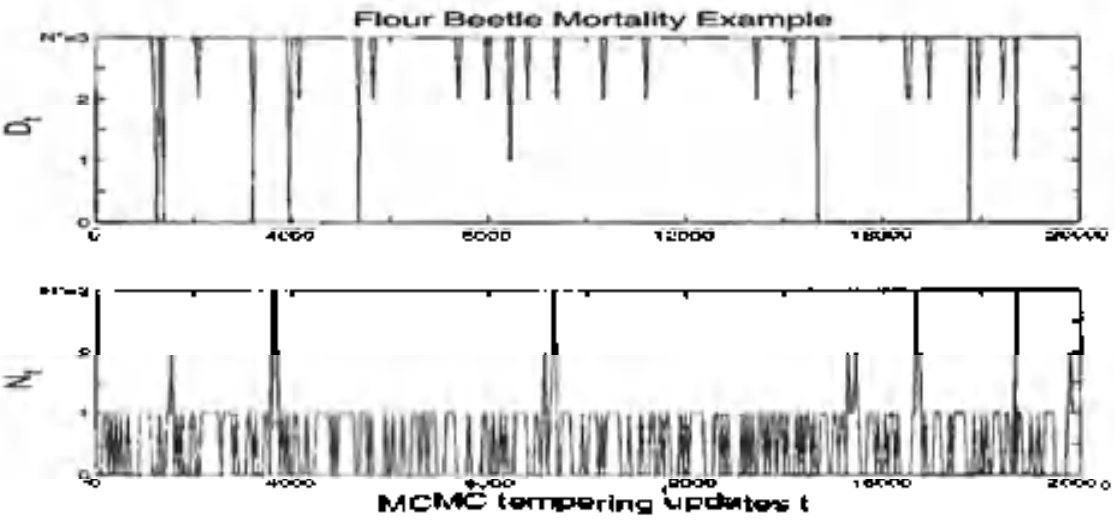


Fig. 13.6. Forward simulation of the dominating process D_t *(above)*, along with the tempering process N_t *(below)*. *(Source: Møller and Nicholls 2004.)*

The tempering sequence chosen in Møller and Nicholls (2004) is

$$\varpi_n(\mu, \sigma, \nu) \propto \pi(\mu, \sigma, \nu) \prod_{i=1}^{8} \left\{ \frac{\exp y_i \frac{\omega_i - \mu}{\sigma}}{1 + \exp \frac{\omega_i - \mu}{\sigma}} \Big/ f_i^{\sigma} \right\}^{\beta_n n_i \nu}\,,$$

[illegible]

14

Iterated and Sequential Importance Sampling

14.1 Introduction

[illegible]

[illegible]

[illegible]

[illegible]

14.2 Generalized Importance Sampling

[illegible]

Lemma 14.[illegible] [illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible] \int [illegible]$$

[illegible]

$$[illegible] \int [illegible]$$

[illegible]

1.4.3 Particle Systems

1.4.3.1 Sequential Monte Carlo

[illegible]

[illegible]

Example 11.2. *Target tracking.* [illegible]

$$[illegible]$$

[illegible]

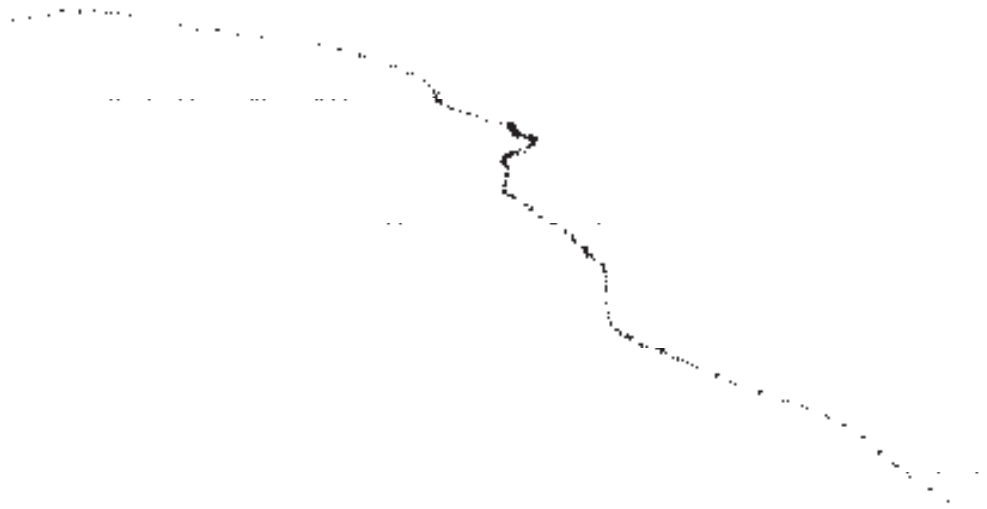


Fig. 11.1. [illegible]

[illegible]

[illegible]

1.3.2 Middle Alerkov Mounds

[illegible]

[illegible]

[illegible]

[illegible]

Fig. 1.1.2. [illegible]

[illegible]

Example 1.1.3. Stochastic [illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

14.3.3 Weight Degeneracy

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

FIG. 14.3. [illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible]

[illegible]

Example 14.4.1 (Continuation of Example 14.4.2) [illegible]

[illegible]

[illegible]

14.3.4 Particle Filters

[illegible]

[illegible]

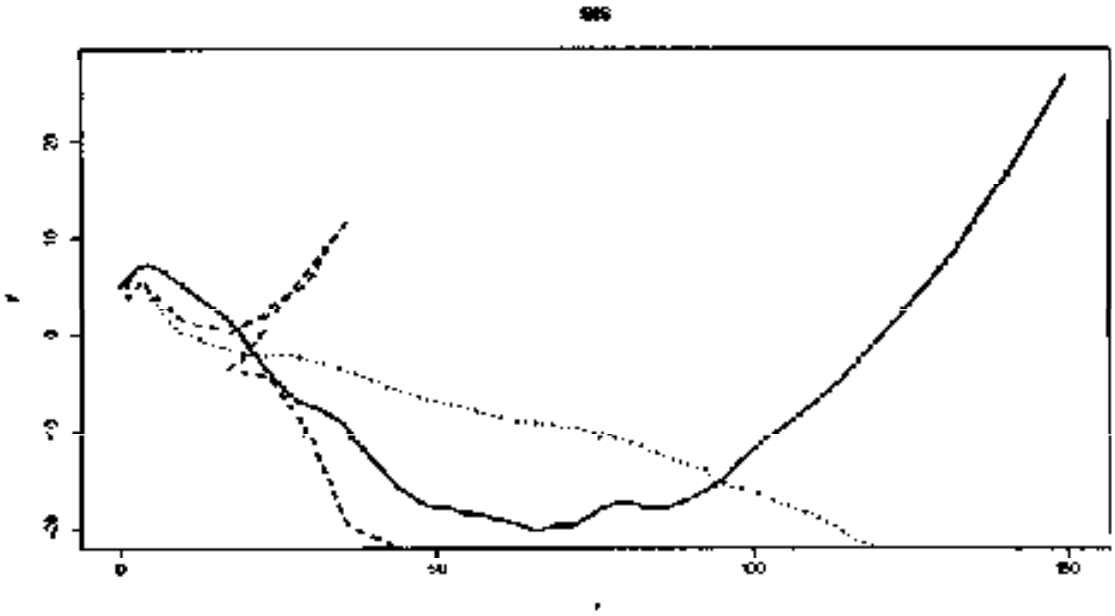


Fig. 14.4. Simulated sequence of 50 target moves (full line), for $\tau = 0.05$ and $\eta = .01$, along with the stepwise importance sampling (dashes) and the final importance sampling estimates (dots) for importance samples of size 1000.

In the particular case $\mathbf{Z}_t \sim \pi_t(\mathbf{z}_t)$, where $\mathbf{Z}_t = (\mathbf{Z}_{t-1}, Z_t)$ (as in Section 14.3.2), with importance/proposal distribution $q_t(z|\mathbf{z}_{t-1})$ that simulates only the last component of $\mathbf{Z}_t$ at time t, the algorithm of Gordon et al. (1993) reads as follows.[4]

Algorithm A.59 –Bootstrap Filter–

At time t,

1 Generate
$$\tilde{Z}_t^{(i)} \sim q_t(z|\mathbf{z}_{t-1}^{(i)}) \qquad i = 1, \ldots, n$$
and set
$$\tilde{\mathbf{z}}_t^{(i)} = (\mathbf{z}_{t-1}^{(i)}, \tilde{Z}_t^{(i)}).$$

2 Compute the importance weight
$$\omega_t^{(i)} \propto \pi_t(\tilde{\mathbf{z}}_t^{(i)})/q_t(\tilde{z}_t^{(i)}|\mathbf{z}_{t-1}^{(i)})\pi_{t-1}(\mathbf{z}_{t-1}^{(i)}).$$

3 Resample, with replacement, n particles $\mathbf{Z}_t^{(i)}$ from the $\tilde{\mathbf{z}}_t^{(i)}$'s according to the importance weights $\omega_t^{(i)}$ and set all weights to $1/n$.

Therefore, there is no direct degeneracy phenomenon, since the weights are set back to $1/n$ after each resampling step in [A.59]. The most relevant

[4] In the original version of the algorithm, the sampling distribution of $\tilde{Z}_t^{(i)}$ in Step 1. was restricted to the prior.

[illegible]

[illegible]

$$[illegible]$$

[illegible]

Example 14.5. (Continuation of Example 14.2) [illegible]

14.3.5 Sampling Strategies

[illegible]

[illegible]

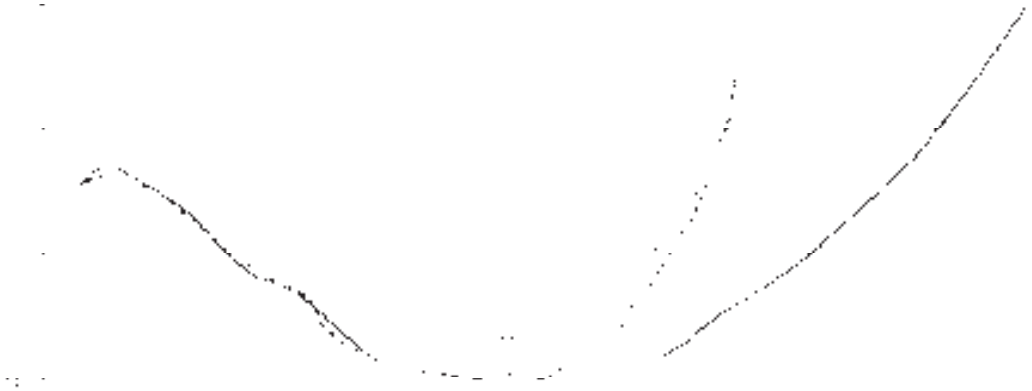


Fig. [illegible]. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

and Titterington (2001), where the empirical distribution (14.8) is replaced with a (nonparametric) kernel approximation which smoothes the proposal into a continuous distribution, as we will encounter a similar proposal in Section 14.4.

14.3.6 Fighting the Degeneracy

When considering an entire path $\tilde{\mathbf{z}}_t$ simulated by the sole addition of only the current state z_t at time t, as in Algorithm [A.59], it is not difficult to realize that as t becomes large, the algorithm must degenerate. As in repeated applications of the bootstrap (Section 1.6.2), iterated multinomial sampling can deplete the sample all the way down to a single value. That is, since the first values of the path are *never re-simulated*, the *multinomial selection step* at each iteration reduces the number of different values for $t < t_0$, at a rate that is faster than exponential (Chopin 2002). Therefore, if we are interested *in a long path*, $\tilde{\mathbf{z}}_t$, *the basic particle filter algorithm simply does not work* well. Note, however, that this is rarely the focus. Instead, the main interest in sequential Monte Carlo is to produce good approximations to the distribution *of the process* z_t *at time* t.

Example 14.6. (Continuation of Example 14.2) Figure 14.6 represents the range of 100 paths simulated over $T = 50$ observations of the tracking model (14.2), that is, at the end of the observation sequence, after reweighting has taken place at each observational instant. While the range of these paths *is never very wide, it is much thinner at the beginning, a result of the repeated* resampling at each stage of the algorithm. ||

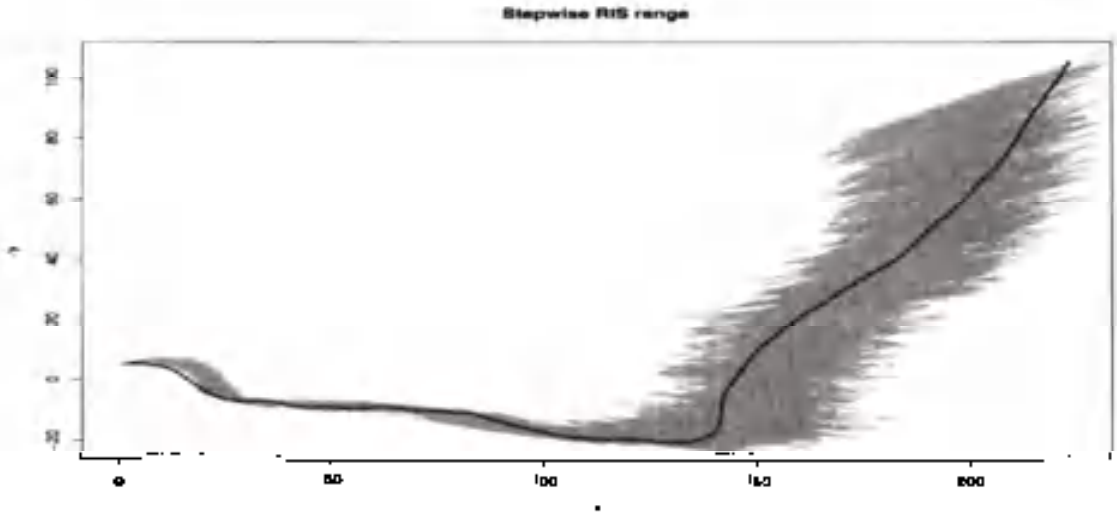


Fig. 14.6. Range of the particle filter path at the end of the observation sequence, along with the true (simulated) sequence of 50 target moves, for $\tau = 0.05$, $\eta = .01$ and 1000 particles.

If one wants to preserve the entire sample path with proper accuracy, the price to pay is, at each step, to increase the number of particles in compensation for the thinning (or *depletion*) of the sample. In general, this is not a small price, unfortunately, as the size must increase exponentially, as detailed in Section 14.3.7! In a specific setting, namely *state space models*, Doucet et al. (2000) establish an optimality result on the best proposal distribution in terms of the variance of the importance weights, conditional on the past iterations. But this optimum is not necessarily achievable, as with the regular importance optimal solution of Theorem 3.12, and can at best slow down the depletion rate.

Note that degeneracy can occur in a case that is a priori more favorable, that is, when the π_t's all have the same support and where particles associated with the previous target π_{t-1} are resampled with weights π_t/π_{t-1}. For instance, if the targets move too quickly in terms of shape or support, very few particles will survive from one iteration to the next, leading to a fast depletion of the sample.

Gilks and Berzuini (2001) have devised another solution to this problem, based on the introduction of additional MCMC moves. Their method is made of two parts, *augmentation* if the parameter space is different for π_t and π_{t-1} and *evolution*, which is the MCMC step. The corresponding algorithm reads as follows.

Algorithm A.60 –Resample–Move Particle Filter–

At iteration t,

1. **Augmentation:** For $i = 1, \ldots, n$, augment $\mathbf{z}_i^{(t-1)}$ with its missing part, as $\tilde{Z}_i^{(t)} \sim \alpha_t(\tilde{z}|\mathbf{z}_i^{(t-1)})$ and $\tilde{\mathbf{z}}_i^{(t)} = (\mathbf{z}_i^{(t-1)}, \tilde{z}_i^{(t)})$.
2. **Weighting:** Compute the corresponding importance weights for the target distribution π_t, $\omega_1^{(t)}, \ldots, \omega_n^{(t)}$, as

$$\omega_i^{(t)} = \pi_t(\tilde{\mathbf{z}}_i^{(t)}) \Big/ \alpha_t(\tilde{z}_i^{(t)}|\mathbf{z}_i^{(t-1)})\pi_{t-1}(\mathbf{z}_i^{(t-1)}).$$

3. **Evolution:** Generate a new iid sample $\mathbf{z}_i^{(t)}$, $i = 1, \ldots, n$, using a proposal proportional to

$$\sum_{i=1}^{n} \omega_i^{(t-1)} q_t(\mathbf{z}|\tilde{\mathbf{z}}_i^{(t-1)}),$$

where q_t is a Markov transition kernel with stationary distribution π_t.

The basis of Algorithm [A.60] is thus to pick up a point $\tilde{\mathbf{z}}_i^{(t-1)}$ according to its weight $\omega_i^{(t-1)}$, that is, via multinomial sampling, and then to implement one iteration of a Markov transition that is stable with respect to π_t. Since the regular multinomial importance sampling produces a sample that is (em-

[illegible]

[illegible] Augmentation [illegible]

[illegible]

14.3.7 Convergence of Particle Systems

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

14.4 Population Monte Carlo

[illegible]

[illegible]

[illegible]

11.1.1 Sample Simulation

[illegible]

$$[illegible] \quad][$$

[illegible]

11.1.2 General Iterative Importance Sampling

[illegible]

[illegible]

$$\varpi_t = \frac{1}{tn} \sum_{\tau=1}^{t} \sum_{i=1}^{n} \frac{\pi(z_i^{(\tau)})}{q_{i\tau}(z_i^{(\tau)})} \tag{14.11}$$

is a convergent estimator of the inverse of the normalizing constant. Therefore, as t increases, ϖ_t contributes less and less to the bias and variability of $\mathcal{I}_t$ and the above properties can be considered as holding for t large enough. In addition, if ϖ_{t-1} in (14.11) is used instead of ϖ_t, that is, if

$$\varrho_i^{(t)} = \frac{\pi(z_i^{(t)})}{\varpi_{t-1}\, q_{it}(z_i^{(t)})}, \tag{14.12}$$

the variance decomposition (14.10) can be approximately recovered, via the *same conditioning argument* (Problem 14.16).

14.4.3 Population Monte Carlo

Following Iba (2000), Cappé et al. (2004) called their iterative approach *population Monte Carlo*, to stress the idea of *repeatedly* simulating an entire sample rather than *iteratively* simulating the points of an approximate sample.

Since the above section establishes that an iterated importance sampling scheme based on sample dependent proposals is fundamentally a specific kind of importance sampling, we can propose the following algorithm, which is validated by the same principles as regular importance sampling.

Algorithm A.61 –Population Monte Carlo–

For $t = 1, \ldots, T$

1. For $i = 1, \ldots, n$,
   (i) Select the generating distribution $q_{it}(\cdot)$.
   (ii) Generate $\tilde{Z}_i^{(t)} \sim q_{it}(z)$.
   (iii) Compute $\varrho_i^{(t)} = \pi(z_i^{(t)})/q_{it}(\tilde{z}_i^{(t)})$.
2. Normalize the $\varrho_i^{(t)}$'s to sum to 1.
3. Resample n values from the $\tilde{z}_i^{(t)}$'s with replacement, using the weights $\varrho_i^{(t)}$, to create the sample $(Z_1^{(t)}, \ldots, Z_n^{(t)})$.

Step 1.(i) is singled out because it is an essential feature of the PMC algorithm: as demonstrated in the previous Section, the proposal distributions can be individually tailored at each step of the algorithm without jeopardizing the validity of the method. The proposals q_{it} can therefore be picked according to the performances of the previous $q_{i(t-1)}$'s (possibly in terms of survival of the values generated with a particular proposal, or a variance criterion, or even on all the previously simulated samples, if storage allows). For instance, the q_{it}'s can include, with low probability, large tails proposals as in the *defensive sampling* strategy of Hesterberg (1998) (Section 3.3.2), to ensure finite variance.

[illegible]

14.4.4 An Illustration for the Mixture Model

[illegible]

$$[illegible]$$

[illegible]

factors σ_{it} can be chosen at random from a set of K scales v_k $(1 \leq k \leq K)$ ranging from, e.g., 10^3 down to 10^{-3} if this range is compatible with the range of the observations. At each iteration t of the PMC algorithm, the probability of choosing a particular scale v_k can be calibrated according to the performance of the different scales over the previous iterations. For instance, possible criterion is to select a scale proportional to its *non-degeneracy rate* on the previous iterations, that is, the percentage of points associated with v_k that survived past the resampling step 3. The reasoning behind this scheme is that, if most μ_j's associated with a given scale v_k are not resampled, the scale is not appropriate and thus should not be much used in the next iterations. However, when the survival rate is null, in order to avoid a definitive removal of the corresponding scale, the next probability ζ_k is set to a positive value ε.

In order to smooth the selection of the scales, Rao-Blackwellization should also be used in the computation of the importance weights, using as the denominator

$$\sum_k \zeta_k \varphi\left(\mu_i^{(t)}; \mu_i^{(t-1)}, v_k\right),$$

where $\varphi(\mu; \xi, v)$ here denotes the density of the two-dimensional normal distribution with mean ξ and variance vI_2 at the vector μ.

The corresponding PMC algorithm thus looks as follows.

Algorithm A.62 Mixture PMC algorithm–

Step 0: Initialization

For $i = 1, \ldots, n$, generate $\mu_i^{(0)}$ from an arbitrary distribution
For $k = 1, \ldots, K$, set v_k and $\zeta_k = 1/K$.

Step t: Update

For $i = 1, \ldots, n$,

a. with probability ζ_k, take $\sigma_{it} = v_k$

b. generate

$$\mu_i^{(t)} \sim \mathcal{N}_2\left(\mu_i^{(t-1)}, \sigma_{it} I_2\right)$$

c. compute the weights

$$\varrho_i \propto \frac{\pi\left(\mu_i^{(t)}|\mathbf{x}\right)}{\sum_k \varrho_k \varphi\left(\mu_i^{(t)}; \mu_i^{(t-1)}, v_k\right)}.$$

Resample the $\mu_i^{(t)}$'s using the weights ϱ_i.
Update the ζ_k's as $\zeta_k \propto \varepsilon + r_k$ where r_k is the number of μ_i's generated with variance v_k that have been resampled in the previous step.

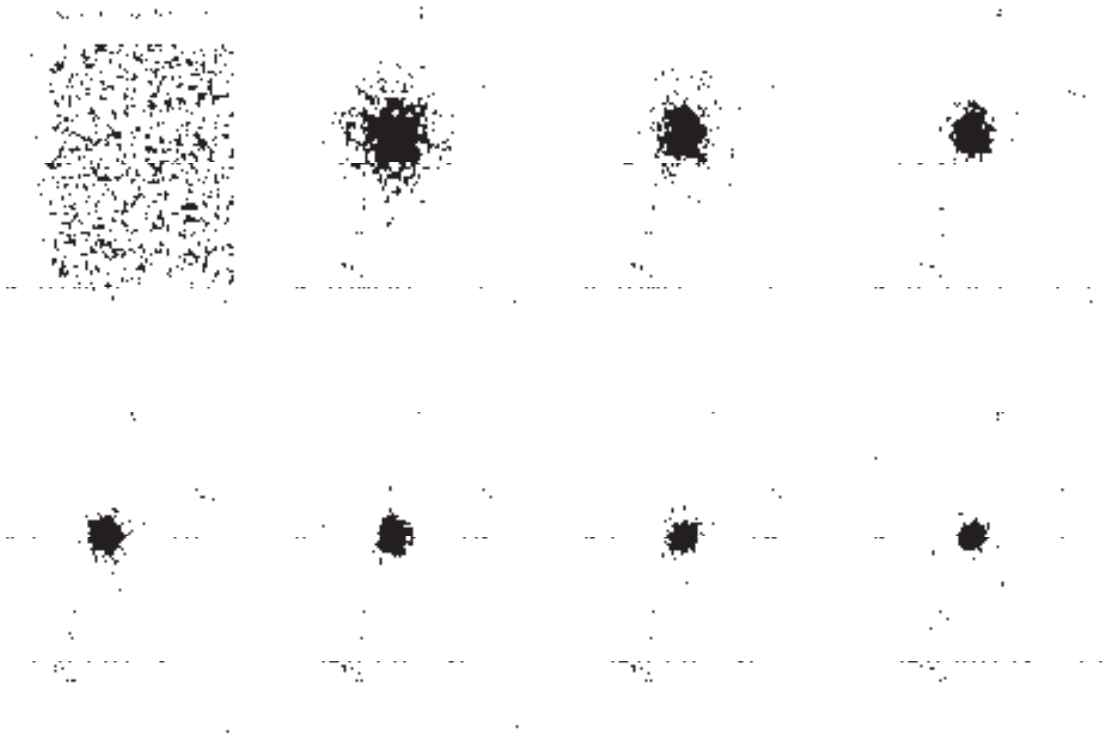


Fig. 14.7 [illegible]

14.4.5 Adaptability to Sequential Algorithms

[illegible]

[illegible]

Fig. 14.5. [illegible]

[illegible]

Example 14.7. (Continuation of Example 14.3) [illegible]

$$\sum_{\quad} \qquad \sum_{\quad} \qquad \sum_{\quad}$$

[illegible]

[illegible]

[illegible]

[illegible]

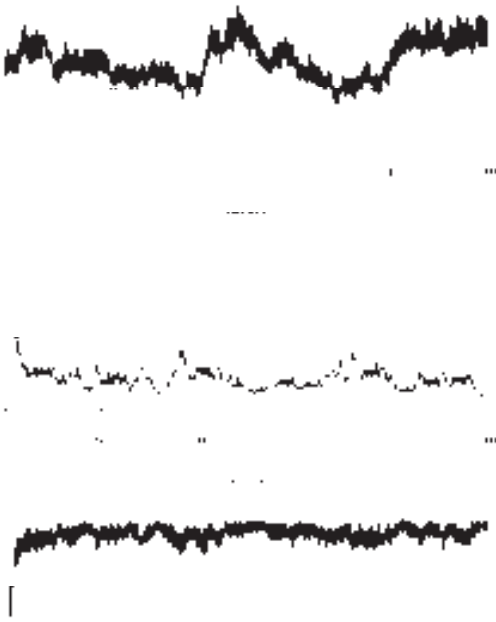


Fig. 11.30. [illegible]

[illegible]

Fig. 11.31. [illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

11.21 [illegible]

$$[illegible]$$

11.6 Notes

11.6.1 A Brief History of Particle Systems

[illegible]

11.6.2 Dynamic Importance Sampling

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

$$\mathfrak{I} = \int_{\mathcal{X}} h(x) f(x) dx .$$

Obviously, regular importance sampling is a special case where X_t is marginally distributed from g and, conditional on x_t, ω is deterministic, with $\omega \propto f(x)/g(x)$. A more general scheme provided by Liang (2002) is based on a transition kernel K as follows.

Algorithm A.63 –Liang's Dynamic Importance Sampling–

At iteration t, given (x_t, ω_t),

1 Generate $y \sim K(x_t, y)$ and compute

$$\varrho = \omega \frac{f(y)K(y, x_t)}{f(x_t)K(x_t, y)} .$$

2 Generate $u \sim \mathcal{U}(0,1)$ and take

$$(x_{t+1}, \omega_{t+1}) = \begin{cases} (y, (1+\delta)\varrho/a) & \text{if } u < a , \\ (x_t, (1+\delta)\omega_t/(1-a)) & \text{otherwise,} \end{cases}$$

where $a = \varrho/(\varrho + \theta(x_t, \omega_t))$, and θ and $\delta > 0$ are both either constant or independent random variables.

The intuition behind this construction is that the increases in the weights allow for moves that would not be allowed by regular Metropolis–Hastings algorithms. But the drawback of this construction is that the importance weights diverge (in t), as shown in Problem 14.2. (Note that the ratio $\varrho/(\varrho+\theta)$ is Boltzman's ratio (7.20).)

In comparison, Liu et al. (2001) use a more Metropolis-like ratio, in the so-called *Q-move* alternative, where

$$(x_{t+1}, \omega_{t+1}) = \begin{cases} (y, \theta \vee \varrho) & \text{if } u < 1 \wedge \varrho/\theta , \\ (x_t, a\omega_t) & \text{otherwise,} \end{cases}$$

with $a \geq 1$ either a constant or an independent random variable. In the spirit of Peskun (1973) (Lemma 10.22) Liang (2002) also establishes an inequality. That is, the acceptance probability is lower for his algorithm compared with a Metropolis–Hastings move.

Liang (2002) distinguishes between the *R-move*, where $\delta = 0$ and $\theta \equiv 1$, yielding

$$(x_{t+1}, \omega_{t+1}) = \begin{cases} (y, \varrho + 1) & \text{if } u < \varrho/(\varrho+1) \\ (x_t, \omega_t(\varrho+1)) & \text{otherwise} \end{cases}$$

and the *W-move*, where $\theta \equiv 0$, $a = 1$ and

$$(x_{t+1}, \omega_{t+1}) = (y, \varrho) .$$

Wong and Liang (1997) also mention the *M-move*, which is essentially the result of Muchlachern et al. (1999): in the above algorithm, when f is the stationary distribution of the kernel K, $\varrho = \omega_t$ and $(x_{t+1}, \omega_{t+1}) = (y, \omega_t)$. Note that all moves are identical when $\theta \equiv 0$.

[illegible]

Proposition 1.8 [illegible]

[illegible]

$$[illegible]$$

[illegible]

1.1.6.3 Hidden Markov Models

[illegible]

A

Probability Distributions

[illegible]

A.1 Normal Distribution [illegible]

[illegible]

[illegible]

[illegible]

A.2 Gamma Distributions [illegible]

[illegible]

[illegible]

A.3 Beta Distributions [illegible]

[illegible]

[illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible] Student's t-Distribution [illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible] Fisher's F-Distribution [illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible] Inverse Gamma Distribution [illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible] Noncentral Chi-Squared Distribution [illegible]

[illegible]

$$[illegible]$$

[illegible]

[illegible]

[illegible] Dirichlet Distribution, [illegible]

[illegible]

[illegible]

[illegible]

[illegible] Pareto Distribution, [illegible]

[illegible]

[illegible]

[illegible] Binomial Distribution, [illegible]

[illegible]

[illegible]

[illegible] Multinomial Distribution, [illegible]

[illegible]

[illegible]

[illegible] Poisson Distribution, [illegible]

[illegible]

[illegible]

[illegible] Negative Binomial Distribution, [illegible]

[illegible]

[illegible]

[illegible]

[illegible] Hypergeometric Distribution [illegible]

[illegible]

[illegible]

B

Notation

B.1 Mathematical

[illegible]

B.2 Probability

[illegible]

B.3 Distributions

[illegible]

[illegible]

18.1 Markov Chains

[illegible]

[illegible]

[illegible]

B.5 Statistics

[illegible]

B.6 Algorithms

[illegible]

References

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]
[illegible]
[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Index of Names

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

Index of Subjects

[illegible]
[illegible]
[illegible]

[illegible]

baseball, [illegible]

[illegible]

cancer treatment success, [illegible]

DNA sequence

[illegible]
[illegible]
[illegible]

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
mastiffs [illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[

[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
Northern Pintail ducks [illegible]
nuclear pump failures [illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[

[illegible]

[illegible]

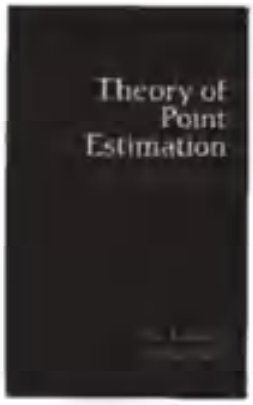
[illegible]

[illegible]

[illegible]
[illegible]
[illegible]

Lehmann: Testing Statistical Hypotheses: Second Edition
Lehmann and Casella: Theory of Point Estimation: Second Edition
Lehmann: Analysis of Variance: Experimental Design
Lehmann: Applying Generalized Linear Models
Madansky: Prescriptions for Working Statisticians
McPherson: Applying and Interpreting Statistics: A Comprehensive Guide: Second Edition
Mueller: Basic Principles of Structural Equation Modeling: An Introduction to LISREL and EQS
Nguyen and Rogers: Fundamentals of Mathematical Statistics: Volume I: Probability for Statistics
Nguyen and Rogers: Fundamentals of Mathematical Statistics: Volume II: Statistical Inference
Nguyen: Introduction to Statistics: An Integrated Way
Nolan and Speed: Stat Labs: Mathematical Statistics through Applications
Peters: Counting for Something: Statistical Principles and Personalities
Pitman: Probability for Applications
Pitman: Probability
Rencher: Methods and Theory: Applied Regression Analysis
Robert: The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation: Second Edition
Robert and Casella: Monte Carlo Statistical Methods: Second Edition
Rose and Smith: Mathematical Statistics with Mathematica
Ruppert: Statistics and Finance: An Introduction
Santner and Duffy: The Statistical Analysis of Discrete Data
Santner and Tamhane: Statistical Methods: The Geometric Approach
Santner and Tamhane: Regression Analysis: Theory, Methods, and Applications
Shao: Mathematical Statistics: Second Edition
Shao: Probability for Statisticians
Shumway and Stoffer: Time Series Analysis and Its Applications
Wasserman: Analyzing Categorical Data
Zacks: Mathematical Statistics: A Unified Introduction
Zacks: Applied Multivariate Analysis
Zimmerman: Statistical Analysis of Ecological Experiments: Second Edition
Ross and Pekoz: A First Course in Statistical Inference
Ross: Probability and Expectation: Fourth Edition
Zacks: Introduction to Reliability Analysis: Probability Models and Statistical Methods

ALSO AVAILABLE FROM SPRINGER!



THE CROSS ENTROPY METHOD

A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning

REUVEN Y. RUBINSTEIN and DIRK P. KROESE

The cross-entropy (CE) method is one of the most significant developments in randomized optimization and simulation in recent years. This book explains in detail how and why the CE method works. The simplicity and versatility of the method is illustrated via a diverse collection of optimization and estimation problems. The book is aimed at a broad audience of engineers, computer scientists, mathematicians, statisticians, and in general anyone, theorist and practitioner, who is interested in smart simulation, fast optimization, learning algorithms, and image processing.

2004/320 PP./HARDCOVER/ISBN 0-387-21240-X
INFORMATION SCIENCE, STATISTICS

THE BAYESIAN CHOICE

Second Edition

CHRISTIAN P. ROBERT

This graduate-level textbook presents an introduction to Bayesian statistics and decision theory. Its scope covers both the basic ideas of statistical theory, and also some of the more modern and advanced topics of Bayesian statistics such as complete class theorems, the Stein effect, Bayesian model choice, hierarchical and empirical Bayes modeling, Monte Carlo integration, including Gibbs sampling, and other MCMC techniques. The second edition includes a new chapter on model choice (Chapter 7), and the chapter on Bayesian calculations (6) has been extensively revised. Chapter 4 includes a new section on dynamic models. In Chapter 3, the material on noninformative priors has been expanded, and Chapter 10 has been supplemented with more examples.

2001/624 PP./HARDCOVER/ISBN 0-387-95231-4
SPRINGER TEXTS IN STATISTICS

THEORY OF POINT ESTIMATION

Second Edition

E.L. LEHMANN and GEORGE CASELLA

This second-much enlarged edition by Lehmann and Casella of Lehmann's classic text on point estimation maintains the outlook and general style of the first edition. All of the topics are updated. An entirely new chapter on Bayesian and hierarchical Bayesian approaches is provided, and there is much new material on simultaneous estimation. Each chapter concludes with a Notes section, which contains suggestions for further study. The book is a companion volume to the second edition of Lehmann's *Testing Statistical Hypotheses*.

1998/615 PP./HARDCOVER/ISBN 0-387-96604-6
SPRINGER TEXTS IN STATISTICS

To Order or for Information:

In the Americas: **CALL:** 1-800-SPRINGER or **FAX:** (201) 348-4505 • **WRITE:** Springer-Verlag New York, Inc., Dept. S7926, PO Box 2485, Secaucus, NJ 07096-2485 • **VISIT:** Your local technical bookstore • **E-MAIL:** orders@springer-ny.com

Outside the Americas: **CALL:** +49 (0) 6221 345-217/8 • **FAX:** +49 (0) 6221 345-229 • **WRITE:** Springer Customer Service, Haberstrasse 7, 69126 Heidelberg, Germany • **E-MAIL:** orders@springer.de

PROMOTION S7926

Springer

Monte Carlo statistical methods, particularly those based on Markov chains, are now an essential component of the standard set of techniques used by statisticians. This new edition has been revised towards a coherent and flowing coverage of these simulation techniques, with incorporation of the most recent developments in the field. In particular, the introductory coverage of random variable generation has been totally revised, with many concepts being unified through a fundamental theorem of simulation

There are five completely new chapters that cover Monte Carlo control, reversible jump, slice sampling, sequential Monte Carlo, and perfect sampling. There is a more in-depth coverage of Gibbs sampling, which is now contained in three consecutive chapters. The development of Gibbs sampling starts with slice sampling and its connection with the fundamental theorem of simulation, and builds up to two-stage Gibbs sampling and its theoretical properties. A third chapter covers the multi-stage Gibbs sampler and its variety of applications. Lastly, chapters from the previous edition have been revised towards easier access, with the examples getting more detailed coverage.

This textbook is intended for a second year graduate course, but will also be useful to someone who either wants to apply simulation techniques for the resolution of practical problems or wishes to grasp the fundamental principles behind those methods. The authors do not assume familiarity with Monte Carlo techniques (such as random variable generation), with computer programming, or with any Markov chain theory (the necessary concepts are developed in Chapter 6). A solutions manual, which covers approximately 40% of the problems, is available for instructors who require the book for a course.

Christian P. Robert is Professor of Statistics in the Applied Mathematics Department at Université Paris Dauphine, France. He is also Head of the Statistics Laboratory at the Center for Research in Economics and Statistics (CREST) of the National Institute for Statistics and Economic Studies (INSEE) in Paris, and Adjunct Professor at Ecole Polytechnique. He has written three other books, including *The Bayesian Choice*, Second Edition, Springer 2001. He also edited *Discretization and MCMC Convergence Assessment*, Springer 1998. He has served as associate editor for the *Annals of Statistics* and the *Journal of the American Statistical Association*. He is a fellow of the Institute of Mathematical Statistics, and a winner of the Young Statistician Award of the Société de Statistique de Paris in 1995.

George Casella is Distinguished Professor and Chair, Department of Statistics, University of Florida. He has served as the Theory and Methods Editor of the *Journal of the American Statistical Association* and Executive Editor of *Statistical Science*. He has authored three other textbooks: *Statistical Inference*, Second Edition, 2001, with Roger L. Berger; *Theory of Point Estimation*, 1998, with Erich Lehmann; and *Variance Components*, 1992, with Shayle R. Searle and Charles E. McCulloch. He is a fellow of the Institute of Mathematical Statistics and the American Statistical Association, and an elected fellow of the International Statistical Institute.

ISBN 0-387-21239-6

EAN

9 780387 212395 >

springeronline.com