EURANDOM PREPRINT SERIES
2013-016
June 24, 2013

Examples in Monte Carlo Simulation

Søren Asmussen

ISSN 1389-2355

Examples in Monte Carlo Simulation

Søren Asmussen

June 24, 2013

Preface

The present document was developed for the lectures ($3 \times 2 \times 45$ minutes) I gave at the EURANDOM Activity Month in Risk and Queueing in March 2013. The main purpose was to show some more practically oriented aspects of the Monte Carlo method. Parts are solutions of exercises in S. Asmussen & P.W. Glynn, *Stochastic Simulation. Algorithms and Analysis*, Springer 2007 (henceforth referred to as AG), resulting from my teaching over the years. Other parts are extracted from AG and one (Ch. 5) is related to new research results.

Another purpose of the lectures was to to give an impression of how I have done my teaching in the area. The Aarhus form (the one in Lund was closely related) was a 2 quarter = 14 weeks course. Each week would have a topic like Importance Sampling, Gaussian Processes etc. with 2×45 min lectures and a project done in a 3h computer lab (some preparation or extra time would usually be needed). The lectures would go through the basic theory with a special view towards the exercise. The project would typically be one of the exercises marked Assignment in AG. This form has worked very well and been favorably received by the students. I personally thinks one need some practical work to get the flavor and an understanding of the Monte Carlo area — doing a course as lectures alone does not achieve that. Unfortunately, it is my impression that very few courses at this level, even worldwide, are organized this way. I highly recommend others to follow up on this!

My students have been half Statistics students and half MathFinance students, who were given a set of exercises only in that area. This is reflected in some bias towards MathFinance in the material incorporated here [the story would have been different if I have had many physics student, say!].

AG is deliberately written without reference to specific software. However, in my own teaching I have been using Matlab. Some code is incorporated at a few places and some general issues discussed in the Appendix. Matlab programs for the 2012 course for statistics students have been supplied by Ólöf Thórisdóttir and will be posted at my web page `http://home.imf.au/asmus` under the heading *Papers and programs for downloading.*

Thanks goes to Anders Alexander Vedel Helweg-Mikkelsen, Leonardo Rojas-Nandyapa, Anders Rønn Nielsen and Ólöf Thórisdóttir for supplying most of the exercise solutions. Another purpose of this document has been to collect them even if some do not refer to topics discussed in the EURANDOM minicourse. They are often inserted in the form given to me or close to, without attempting a perfectly uniform style for which much additional work would have been required. One should certainly take all material presented here with a grain of salt!

Lars Madsen deserves a special thanks for, as on many other occasion, helping with the set-up, improving the typography and solving problems that were beyond my LaTeX ability

Søren Asmussen
Aarhus and Eindhoven, March 2013

Contents

1 Random Variate Generation

1.1 Uniform random numbers

The point of view of AG is that the average user should keep clear of uniform random number generation and use available software (commercial or public domain). Modern algorithms are much better than some decades ago, pass all goodness-of-fit tests and there is little chance of competing neither in terms of accuracy or speed.

An example of such an algorithm is the following developed by L'Ecuyer and used in many software packages (e.g., Arena, or SAS)

1. $x_n \longleftarrow \bigl(A_1 x_{n-2} - A_2 x_{n-3}\bigr) \bmod M_1$,
2. $y_n \longleftarrow \bigl(B_1 y_{n-1} - B_2 y_{n-3}\bigr) \bmod M_2$,
3. $z_n \longleftarrow (x_n - y_n) \bmod M_1$,
4. If $z_n > 0$, return $u_n = z_n/(M_1+1)$; else return $u_n = M_1/(M_1+1)$,

where M_1 = 4,294,967,087, M_2 = 4,294,944,443, A_1 = 1,403,580, A_2 = 810,728, B_1 = 527,612, B_2 = 1,370,589 (the seed is the first three x's and the first three y's).

I have seen the claim that this is what Matlab uses but also that it is a Mersenne twister.

A recent algorithm worth mentioning is L'Ecuyer's `mrg32k3a`.

1.2 Non-uniform random numbers

Again, the point of view of AG is that modern software does an excellent job. In Matlab, there are lots of routines producing standard distributions like the gamma, normal etc., and there is little point in trying to improve these or compete.

However, in a few situations the need arises to generate r.v.'s from non-standard distribution. Two such cases, from which we will give examples, are MCMC (Markov chain Monte Carlo, Example 1) and Lévy processes (Section 1.2).

Assume we want to generate a real-valued r.v. X from the distribution F. There are two standard methods around, *inversion* and *acceptance-rejection*.

Inversion assumes the c.d.f. $F(x) = \mathbb{P}(X \le x)$ to be known and the inverse F^{-1} to be computable. X can then be generated as $F^{-1}(U)$ where U is uniform. Standard example: the exponential distribution, where $F(x) = 1-\mathrm{e}^{-x}$, $F^{-1}(x) = -\log(1-x)$.

Main limitation: F^{-1} may not be computable. Even F needs not be so, or at least of a complicated form, say involving special functions.

In acceptance-rejection, the density $f(x)$ (the target) of X needs to be available. One then finds a density $g(x)$ (the proposal), such that easily simulated and has the property $f(x) \leq Cg(x)$, for some known constant $C < \infty$; see Fig. 1.1.

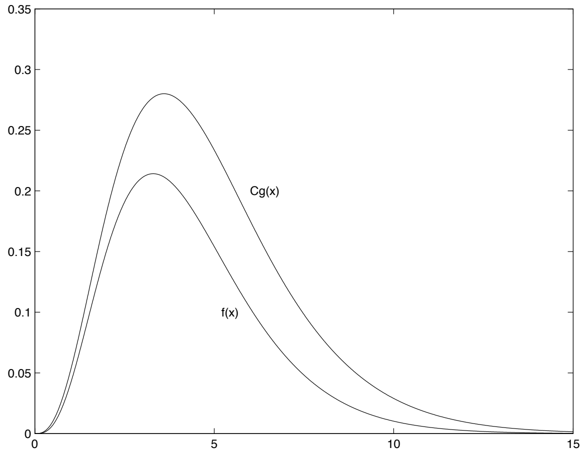


Figure 1.1

The A-R algorithm is

```
r=0;
while r
    y= a r.v. from g; u=rand;
    r=u>f(y)/(Cg(y));
end;
x=y;
```

In earlier years, I have been using the following exercise as introduction to non-uniform r.v. generation:

Exercise AG II.2.9.

Produce 100,000 standard normal r.v.'s using each of the following methods: (a) inversion using the approximation AG.II.(2.2) of Φ^{-1}, (b) Box-Muller, (c) Marsaglia polar, (d) A–R as in Example AG.II.2.7, (e) ratio-of-uniforms and (as comparison) (f) a routine of a standard package, say Matlab. The desired results are the CPU time needed for each of the methods.

The weakness of the exercise is that the results are highly implementation dependent. In particular when comparing of speeds, Matlab will always beat anything produced by the student and methods (like (b)) that can be programmed with Matlab's matrix facilities will come out favorably compared to those that don't (like (d) where the A-R step must use something like **`while`**. In the 2012 class, I used instead:

Exercise Nov 1, 2012.

Generate r.v.'s from the density $f(x) = 2x\mathrm{e}^{-x^2}$, $x > 0$, using (a) inversion, (b) acceptance-rejection with the standard exponential density $g(x) = \mathrm{e}^{-x}$ as proposal. Report your results as histograms compared to $f(x)$.

This makes the student try both inversion and A-R, but is deceivingly simple: For inversion, F^{-1} is explicitly available ($F(x) = 1 - \mathrm{e}^{-x^2}$, $F^{-1}(u) = \sqrt{-\log(1-u)}$) and for A-R, the dominating constant C can just be computed by noting that $f(x)/g(x) = 2x\mathrm{e}^{-x^2+1}$ is maximized for $x = 1$ with maximum $C = 2$.

Solution.

The implementation is straightforward. Fig. 1.2 summarizes the results of using inversion and acceptance-rejection in terms of an empirical histogram plotted along with the true density.

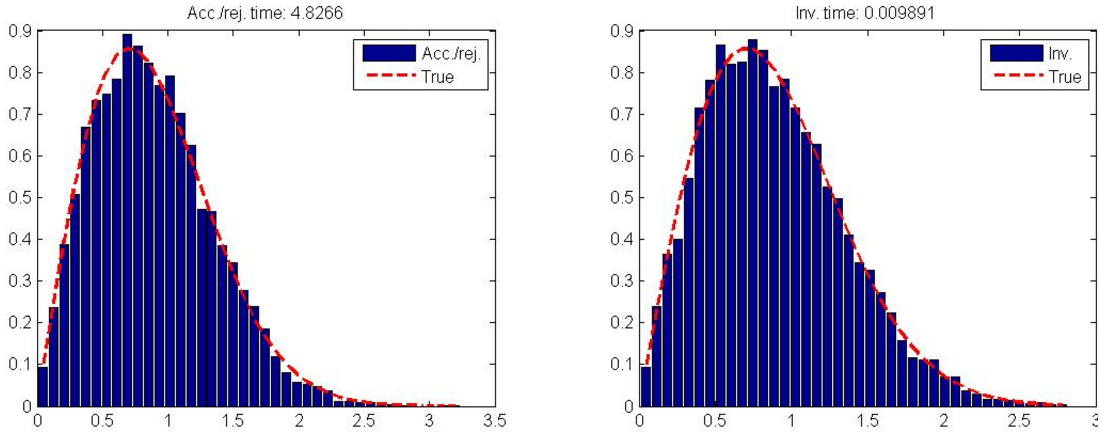


Figure 1.2

Example 1.

Consider a stochastic volatility model

$$X_{k+1} = \phi X_k + \sigma U_k, \qquad Y_k = \theta \mathrm{e}^{X_k/2} V_k + \omega V_k$$

where U, V are standard normal. We think of the Y as the asset prices and of the $\theta \mathrm{e}^{X_k/2}$ as the stochastic volatilities given in terms of the autoregressive process X. One may be interested in recovering $X_1, \ldots, X_N$ and thereby the volatilities from an observed sequence $Y_1, \ldots, Y_N$. One approach is MCMC, where one generates a Markov chain $\xi = (\xi_{n1}, \ldots, \xi_{nN})_{n=1,2,\ldots}$ with state space $\mathbb{R}^N$ such that the stationary distribution is the distribution of $X_1, \ldots, X_N$ given $Y_1, \ldots, Y_N$. For large values of n one can then think of $(\xi_{n1}, \ldots, \xi_{nN})$ as a typical value in this conditional distribution.

The Markov chain ξ is conveniently simulated by Gibbs sampling, where $(\xi_{(n+1)1}, \ldots, \xi_{(n+1)N})$ is generated from $(\xi_{n1}, \ldots, \xi_{nN})$ by updating one component at a time.

Component k is then generated from the conditional distribution of X_k given X_{k-1}, X_{k+1}, Y_k. After some algebra, the density of this distribution comes out as proportional to

$$\exp\{-\alpha(x-\mu)^2-\beta e^{x-\mu}\}$$

for some constants α,β,μ which in addition to X_{k-1},X_{k+1},Y_k involve also $\phi,\theta,\sigma^2,\omega^2$ (which are assumed known). R.v. generation from this density is of course non-standard.

For more detail, see references in footnote.[1].

Lévy processes via the Lévy density

A Lévy process X has the form

$$X(t)=mt+\sigma B(t)+J(t)\,,$$

where B is standard Brownian motion and J an independent pure jump process specified by its Lévy measure ν (which may be infinite); the intuitive description is that jumps of size x occur at Poisson intensity $\nu(\mathrm{d}x)$. Examples of Lévy measures (usually specified by their Lebesgue density) are in the following table.

Process	Density of ν	
Gamma	$\alpha e^{-\lambda x}/x$	$x>0$
	0	$x<0$
Inverse Gaussian	$\dfrac{1}{\sqrt{2\pi}\,x^{3/2}}e^{-x\gamma^2/2}$	$x>0$
	0	$x<0$
Stable	$C_+/y^{\alpha+1}$	$y>0$
	$C_-/\lvert y\rvert^{\alpha+1}$	$y<0$
Tempered stable (CGMY)	C_+e^{-Mx}/x^{1+Y}	$x>0$
	$C_-e^{Gx}/\lvert x\rvert^{1+Y}$	$x<0,$
NIG (normal inverse Gaussian)	$\alpha\delta K_1\bigl(\alpha\lvert x\rvert\bigr)e^{\beta x}/\pi\lvert x\rvert$	$0<x<\infty$
Generalized hyperbolic	$\dfrac{e^{\beta x}}{\lvert x\rvert}\displaystyle\int_0^\infty\frac{\exp\{-\lvert x\rvert\sqrt{2y+\alpha^2}}{\pi^2y\bigl[J^2_{-\lambda}\bigl(\delta\sqrt{2y}\bigr)+Y^2_{-\lambda}\bigl(\delta\sqrt{2y}\bigr)\bigr]}\,\mathrm{d}y$	$0<x<\infty$

Here K_1,J_μ,Y_μ are certain Bessel functions (for the generalized hyperbolic case, there is a similar expression with an added exponential term if the parameter λ is non-negative).

[1] p. 184 in O. Cappé, E. Moulines, & T. Rydén (2005) *Inference in Hidden Markov Models.* Springer-Verlag.
N. Shephard & M. Pitt (1997) Likelihood analysis og non-Gaussian measurement time series. *Biometrika* **84**, 653–667. Erratum 2004 in **91**, 249–250.

For simulation of the paths, the simplest case is that the distribution of $X(h)$ is known for any h in a form that allows for simulation; then one can just simulate discrete skeletons as for Brownian motion. Examples are the Gamma and inverse Gaussian Lévy processes. There are also some Lévy processes that can be simulated by *subordination*, i.e. as $X(t) = Y\bigl(T(t)\bigr)$, where Y is a Lévy process (often Brownian motion) and T a subordinator (an increasing Lévy process). An example is the NIG process where Y is Brownian motion and T is the inverse Gaussian Lévy process. However, in many examples (e.g. GGMY and hyperbolic) none such simplification is available and one has to proceed via the Lévy measure ν.

The easy case when the total mass $\lambda = \int \nu(\mathrm{d}x)$ is finite. Then $X(t) = \sum_1^{N(t)} Y_i$ where N is Poisson with rate λ and the Y_i are i.i.d. with distribution ν/λ, and simulation is straightforward (provided, of course, that simulation from F is within reach). If, as most often, $\lambda = \infty$ one instead truncates. This means defining $\nu_{\varepsilon,+}, \nu_{\varepsilon,+}, \nu_{\varepsilon,0}$ as the restrictions of ν to (ε, ∞), $(-\infty, -\varepsilon)$, resp. $[-\varepsilon, \varepsilon]$ allowing to write $X = X_{\varepsilon,+} + X_{\varepsilon,+} + X_{\varepsilon,0}$ for Lévy processes corresponding to these Lévy measures. The advantage is that $X_{\varepsilon,+}, X_{\varepsilon,+}$ are compound Poisson. $X_{\varepsilon,0}$ is either put to 0 or approximated by a Brownian motion with the same drift and variance (for accuracy, ε needs to be sufficiently small for both procedures).

As the examples above show, many Lévy densities are sufficiently complicated that generating the jumps of $X_{\varepsilon,+}, X_{\varepsilon,+}$ is a non-standard problem in r.v. generation. Another such non-standard problem arises when generating the minimal entropy process X^* associated with a given X (this comes up in risk-neutral pricing). Namely, $\nu^*(\mathrm{d}y) = \exp\{\lambda(\mathrm{e}^y - 1)\}\,\nu(\mathrm{d}y)$, and the additional factor $\exp\{\lambda(\mathrm{e}^y - 1)\}$ invariably creates difficulties no matter how easy it is to simulate from X itself

Exercise Nov 1, 2012.

Generate paths of a CGMY Lévy process with suitably chosen parameters by a compound Poisson approximation (forgetting about small jump approximations).

Solution.

We took $C_+ = C_- = 1.5$, $G = M = 1$, $Y = 1$ and $\varepsilon = 0.5$ [this value of ε may be too large for use in realistic settings; however, we are here only interested in demonstrating the methodology]. $X_{\varepsilon,+}$ and $X_{\varepsilon,=}$ are simulated separately. The method for generating jumps of $X_{\varepsilon,+}$ (the case of $X_{\varepsilon,=}$ is similar) is to first generate the number of jumps as Poisson(λ_ε). To generate the jump sizes, one can split the interval (ε, ∞) up into $K+1$ subintervals ($K = 3$ on Fig. 1.3) and compute the masses $p_1, \ldots, p_{K+1}$ over each by numerical integration. One then first selects an interval w.p. $p_k/(p_1 + \cdots + p_{K+1})$ for k, and has to simulate from the distribution with density proportional to the Lévy density restricted to this interval. For $k \le K$, this is done by A-R with a uniform proposal. For $k = K+1$, one uses that the Lévy measure is bounded by e^{-x} and applies A-R with $g(x) = \mathrm{e}^{-(x-t_N)}$, $x > t_N$ as proposal and x^{-1-Y} as acceptance probability [this requires $t_N > 1$].

Carrying out this procedure gives the results depicted in Fig. 1.4.

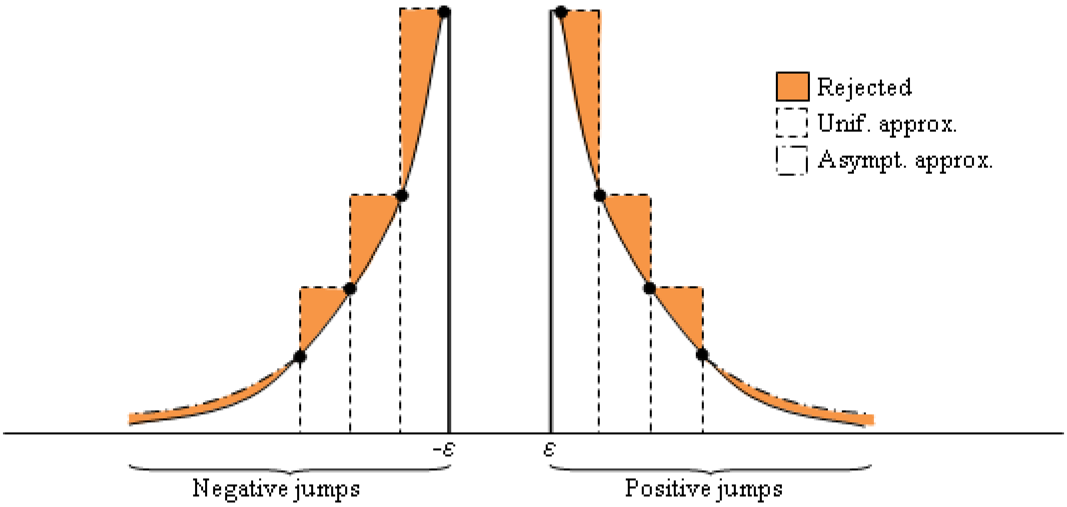


Figure 1.3

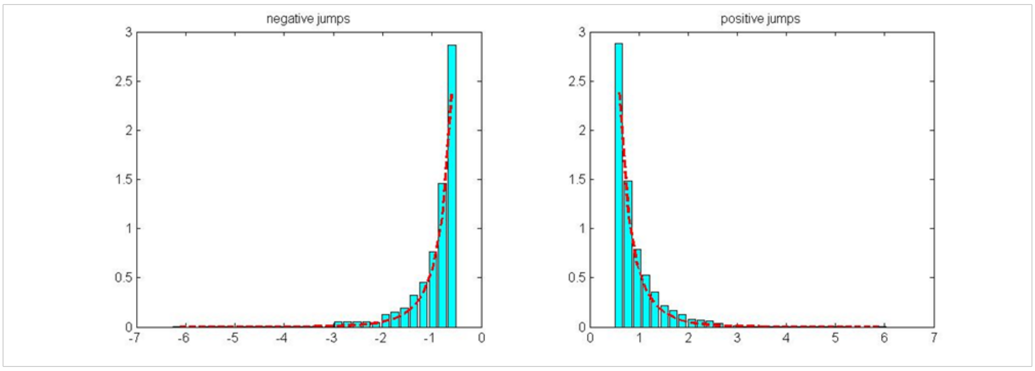


Figure 1.4

An example of a sample path of the compound Poisson corresponding to a given set of jumps is shown in Fig. 1.5 (the positions may be generated as iid uniforms on $(0, 1)$)

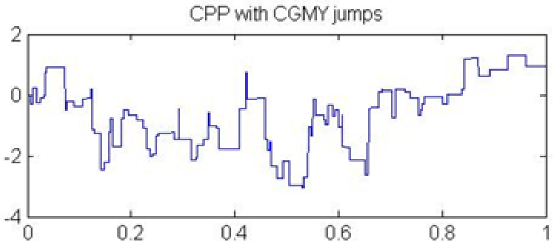


Figure 1.5

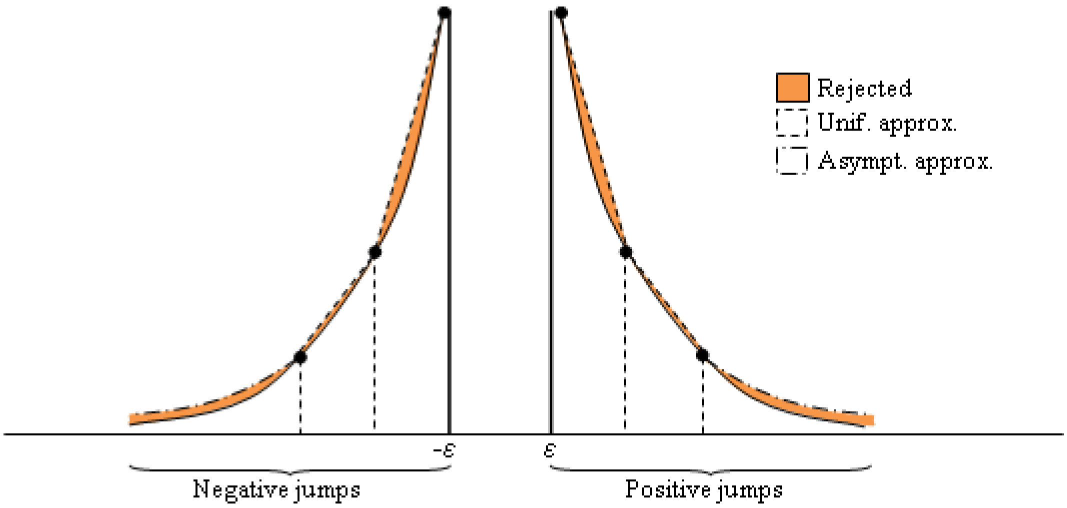


Figure 1.6

It is tempting instead of the approximation with boxes to use one with trapezoids, cf. Fig. 1.6. However, the belief that this may lead to fewer rejections is false at least if one uses the most obvious method to generate from the distribution with density proportional to the the dominating chord on interval k, namely $A - R$: in the end the number of rejections will be the same!

Remark 1.1. A large number of Lévy densities are close to a power $x^{-\alpha+1}$ for small x and close to an exponential $\mathrm{e}^{-\mu x}$ for large x. One may ask whether this observation can be used to build a general procedure for simulating a jump $\geq \varepsilon$ by using a Pareto proposal for small x, an exponential proposal for large x and somehow interpolate in between. I don't believe this is possible because there is no general handle on the behavior of the Lévy density in the intermediate range. One will most often have to use the box procedure above.

The box procedure may be a resort for r.v. generation from other complicated densities than truncated Lévy densities.

Example 2.

Simulation from a density truncated to (a, b) also comes up in other context, in particular stratification. Even if simulation from the original density f may be easy, so needs not be the case when truncating. If inversion applies to f, so it does to f truncated to (a, b), cf. AG p. 39. But assume for example that f is the density of a NIG r.v., i.e.

$$f(x) = \frac{\alpha\delta}{\pi} \exp\bigl\{\delta\sqrt{\alpha^2 - \beta^2} - \beta\mu\bigr\} \frac{K_1\bigl(\alpha\sqrt{\delta^2 + (x-\mu)^2}\bigr)}{\sqrt{\delta^2 + (x-\mu)^2}}\, \mathrm{e}^{\beta x},$$

where K_1 is a Bessel function. A r.v. from f can be generated in a simple way as $Z^{1/2}V$ where Z is inverse Gaussian and V standard normal. When truncating, there is no adaptation of this procedure, and one has to use for example the box procedure. ■

2 Variance Reduction

In a vast number of applications of Monte Carlo simulation, the aim is to produce an estimate of a number z that can be written as an expected value $\mathbb{E}\,Z$ for some r.v. Z. For example, z can be a probability, in which case $Z = \mathbb{1}_A$, or an option price (e.g. of the form $\mathrm{e}^{-rT}\,\mathbb{E}[\mathrm{e}^Y - K]^+$ so that $Z = \mathrm{e}^{-rT}[\mathrm{e}^Y - K]^+$. The Monte Carlo method then proceeds by simulating R i.i.d. replications $Z_1, \ldots, Z_R$ of Z, estimate z by the empirical mean $\widehat{z} = (Z_1 + \cdots + Z_R)/R$, and report the final result of the simulation experiment as a (say) 95% confidence interval $\widehat{z} \pm 1.96s/\sqrt{R}$ where $s^2 = \sum_1^R (Z_r - \widehat{z})^2/(R-1)$ is the empirical variance. We also talk about the *Crude Monte Carlo* (CMC) method.

Variance reduction methods aim at finding an alternative algorithm, which estimates z with smaller variance within the same amount of computer time. Often this means finding a different Z_{VR} with $\mathbb{E}\,Z_{\mathrm{VR}} = z$ and proceeding as in the CMC for a confidence interval.

Variance reduction is typically most readily available in well-structured problems. Also, variance reduction typically involves a fair amount of both theoretical study of the problem in question and added programming effort. For this reason, variance reduction is most often worthwhile only if it is substantial. Assume, for example, that a sophisticated method reduces the variance by 25 % compared to the CMC method, i.e., $\sigma^2_{\mathrm{VR}} = 0.75\sigma^2_{\mathrm{CMC}}$ in obvious notation, and consider the numbers R_{CMC}, R_{VR} of replications to obtain a given precision ε (say in terms of half-width of the 95 % confidence interval). Then

$$\varepsilon = \frac{1.96\sigma_{\mathrm{CMC}}}{\sqrt{R_{\mathrm{CMC}}}} = \frac{1.96\sigma_{\mathrm{VR}}}{\sqrt{R_{\mathrm{VR}}}}, \quad R_{\mathrm{VR}} = \frac{\sigma^2_{\mathrm{VR}}}{\sigma^2_{\mathrm{CMC}}} R_{\mathrm{CMC}} = 0.75\,R_{\mathrm{CMC}}\,,$$

so that at best (assuming that the expected CPU times $T_{\mathrm{CMC}}, T_{\mathrm{VR}}$ for one replication are about equal), one can reduce the computer time by only 25 %. This is in most cases of no relevance compared to the additional effort to develop and implement the variance reduction method. If $T_{\mathrm{VR}} > T_{\mathrm{CMC}}/0.75$, as may easily be the case, there is no gain at all.

2.1 Importance sampling

The idea behind importance sampling is changing the basic distribution underlying the simulation experience. For a simple example, assume $Z = \varphi(\boldsymbol{X}, \boldsymbol{Y})$ where $\boldsymbol{X}, \boldsymbol{Y}$

are random vectors with densities $f_{\boldsymbol{X}}(\boldsymbol{x})$, $f_{\boldsymbol{Y}}(\boldsymbol{y})$. If $\widetilde{f}(\boldsymbol{x})$ is a different density for $\boldsymbol{X}$, one then has

$$\begin{aligned} z = \mathbb{E}\,\varphi(\boldsymbol{X},\boldsymbol{Y}) &= \iint \varphi(\boldsymbol{x},\boldsymbol{y}) f_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{Y}}(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}\,\mathrm{d}\boldsymbol{x} \\ &= \iint \varphi(\boldsymbol{x},\boldsymbol{y}) \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{\widetilde{f}_{\boldsymbol{X}}(\boldsymbol{x})} \widetilde{f}_{\boldsymbol{X}}(\boldsymbol{x}) f_{\boldsymbol{Y}}(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y}\,\mathrm{d}\boldsymbol{x} = \widetilde{\mathbb{E}}[ZL]\,, \end{aligned}$$

where $L = f_{\boldsymbol{X}}(\boldsymbol{x})(\boldsymbol{X})/\widetilde{f}_{\boldsymbol{X}}(\boldsymbol{X})$ is the likelihood ratio and $\widetilde{\mathbb{E}}$ refers to a probability measure $\widetilde{\mathbb{P}}$ under which the density of $\boldsymbol{X}$ is $f_{\boldsymbol{X}}(\boldsymbol{x})(\boldsymbol{x})$, not $f_{\boldsymbol{X}}(\boldsymbol{x})$, whereas the density of $\boldsymbol{Y}$ remains $f_{\boldsymbol{Y}}(\boldsymbol{x})(\boldsymbol{y})$. Thus one can let $Z_{\mathrm{IS}} = ZL = \varphi(\boldsymbol{X},\boldsymbol{Y})L(\boldsymbol{X})$ where $\boldsymbol{X}$ is generated from $\widetilde{f}_{\boldsymbol{X}}(\boldsymbol{x})$, not from $f_{\boldsymbol{X}}(\boldsymbol{x})$ as for CMC.

Virtually all implementations of IS follows this pattern or minor extensions. The abstract formulation is that the likelihood ratios (Radon-Nikodym derivatives) are connected by $\widetilde{L} = \mathrm{d}\widetilde{\mathbb{P}}/\mathrm{d}\,\mathbb{P} = 1/L$ where $L = \mathrm{d}\,\mathbb{P}\,/\mathrm{d}\widetilde{\mathbb{P}}$.

Example 3.

Let $z = \mathbb{P}(\boldsymbol{X} \in \boldsymbol{A})$ where $\boldsymbol{A} = \{(x,y) : x \geq a,\, y \geq a\}$ and let $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0},\boldsymbol{C})$ be $n = 2$-dimensional normal (Gaussian) where

$$\boldsymbol{C} = \begin{pmatrix} 4 & -1 \\ -1 & 4 \end{pmatrix}.$$

We try to estimate z by changing the distribution of $\boldsymbol{X}$ to $\mathcal{N}(\boldsymbol{\mu},\widetilde{\boldsymbol{C}})$. Then

$$\begin{aligned} f_{\boldsymbol{X}}(\boldsymbol{x}) &= \frac{1}{(\det \boldsymbol{C})^{1/2}} \exp\bigl\{-\tfrac{1}{2}\boldsymbol{x}^{\mathsf{T}}\boldsymbol{C}^{-1}\boldsymbol{x}\bigr\}\,, \\ \widetilde{f}_{\boldsymbol{X}}(\boldsymbol{x}) &= \frac{1}{\bigl(\det \widetilde{\boldsymbol{C}}\bigr)^{1/2}} \exp\bigl\{-\tfrac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\mathsf{T}}\widetilde{\boldsymbol{C}}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\bigr\}\,. \end{aligned}$$

We took $a = 3$, $\widetilde{\boldsymbol{C}} = \delta\boldsymbol{C}$ and chose by trial-and-error $\delta = 1/2, 1, 2$, $\boldsymbol{\mu} = (2,2)$, $(2,0)$, $(0,0)$, $(-2,0)$ $(-2,-2)$. Performing $R = 10{,}1000$ replications gave the following values of the estimated variance ratio $s^2_{\mathrm{IS}}/s^2_{\mathrm{CMC}}$:

	$\delta = 1/2$	$\delta = 1$	$\delta = 2$
$\boldsymbol{\mu} = (2,2)$	0.00135±0.00004	0.00139±0.00004	0.00137±0.00005
$\boldsymbol{\mu} = (2,0)$	0.00150±0.00031	0.00130±0.00009	0.00132±0.00007
$\boldsymbol{\mu} = (0,0)$	0.00096±0.00125	0.00131±0.00022	0.00142±0.00011
$\boldsymbol{\mu} = (-2,0)$	0.00000±0.00000	0.00082±0.00072	0.00130±0.00021
$\boldsymbol{\mu} = (-2,-2)$	0.00000±0.00000	0.00000±0.00000	0.00143±0.00050

We note that $\boldsymbol{\mu} = (0,0)$, $\delta = 1$ corresponds to CMC and observe that IS may give variance reduction, i.e. $s^2_{\mathrm{IS}}/s^2_{\mathrm{CMC}} < 1$, but not always. Choosing $\boldsymbol{\mu}$ wtih negative components and $\delta > 1$ appears disadvantageous.

The common general rule of thumb for chosing the importance distribution is that "$\widetilde{\mathbb{P}}$ should give more weight than $\mathbb{P}$ in areas of the sample space where Z is large". This together with the preliminary findings in Example 3 motivates to proceed as in the following two exercises.

Exercise AG.V.1.4.

Let $z, \boldsymbol{X}, \boldsymbol{A}$ be as in Example 3. For $a = 1, 3, 10$:

(i) Try first to give simulation estimates of $z = \mathbb{P}(\boldsymbol{X} \in \boldsymbol{A})$ and associated 95% confidence intervals using the CMC method.

(ii) Find next the point $\boldsymbol{b} \in \boldsymbol{A}$ that maximizes the $\mathcal{N}(\boldsymbol{0}, \boldsymbol{C})$ density and repeat (i), with the CMC method replaced by importance sampling, where the importance distribution is $\mathcal{N}(\boldsymbol{b}, \boldsymbol{C})$.

(iii) In (ii), experiment with importance distributions of the form $\mathcal{N}(\boldsymbol{b}, \delta\boldsymbol{C})$

Solution.

(i) The results shown in Table 2.1 are Crude Monte Carlo simulation estimates corresponding to $R =$ 1,000,000 replications.

a	$\widehat{z}$	$\widehat{\sigma}^2$
1	$6.510 \cdot 10^{-2}$	$6.09 \cdot 10^{-2}$
3	$1.354 \cdot 10^{-3}$	$1.35 \cdot 10^{-2}$
10	—	—

Table 2.1: CMC.

In the last line where $a = 10$, $\mathbb{P}(\boldsymbol{X} \in \boldsymbol{A})$ is so low that $\boldsymbol{X} \in \boldsymbol{A}$ was never observed. Therefore both mean and variance came out as zero. The values of the empirical variance are in the good agreement with the theoretical variance $z(1-z)$.

CMC requires large sample sizes for even moderately small probabilities. For example, for $a = 3$ the number R of replicates required to get the second significant digit in $\widehat{z}$ correct is given by

$$\frac{1.96\sqrt{1.35e-2}}{\sqrt{R}} = e-4$$

which gives $R = 700,130$. For the third digit, 100 times as many are required!

(ii) Observe that the point $\boldsymbol{a} = (a, a)^T$ maximizes the $\mathcal{N}(\boldsymbol{0}, \boldsymbol{C})$ density in the set $\boldsymbol{A}$. Importance sampling simulation estimates are given in Table 2.2, using $\mathcal{N}(\boldsymbol{a}, \boldsymbol{C})$ as importance distribution (again, $R =$ 1,000,000).

a	$\widehat{z}_{\text{IS}}(a)$	$\widehat{\sigma}^2_{\text{IS}}(a)$
1	$6.50 \cdot 10^{-2}$	$1.99 \cdot 10^{-2}$
3	$1.38 \cdot 10^{-3}$	$2.04 \cdot 10^{-5}$
10	$1.17 \cdot 10^{-17}$	$1.00 \cdot 10^{-32}$

Table 2.2: IS as in (ii).

(iii) Finally, we experiment with importance sampling distributions of the form $\mathcal{N}(\mathbf{a}, \delta\mathbf{C})$ for different $\delta > 0$. The Radon-Nikodym derivative is given by

$$L(\boldsymbol{X}, a, \delta) = \sqrt{\delta^{n/2}} \exp\left\{ -\frac{1}{2}\left(\boldsymbol{X}^{\mathsf{T}}\boldsymbol{C}^{-1}\boldsymbol{X} - \frac{(\boldsymbol{X}-\boldsymbol{a})^{\mathsf{T}}\boldsymbol{C}^{-1}(\boldsymbol{X}-\boldsymbol{a})}{\delta}\right)\right\}.$$

where n (here 2) is the dimension of $\boldsymbol{X}$. In the numerical estimates shown in Table 2.3 the value of δ was chosen in such way that the variance of the estimator was minimized.

a	δ	$\widehat{z}_{\text{IS}}(a,\delta)$	$\widehat{\sigma}^2_{\text{IS}}(a,\delta)$
1	0.644	$6.53 \cdot 10^{-2}$	$1.64 \cdot 10^{-2}$
3	0.308	$1.39 \cdot 10^{-3}$	$8.28 \cdot 10^{-6}$
10	0.057	$1.18 \cdot 10^{-17}$	$7.25 \cdot 10^{-34}$

Table 2.3: IS as in (iii).

Since $\boldsymbol{C}$ corresponds to negative correlation, it seems reasonable to think that variance reduction could be obtained by letting $\widetilde{\boldsymbol{C}}$ have positive off-diagonal elements, but this has not been implemented.

Exercise Exercise AG.V.2.8.

Consider a European call basket option with payout $\mathrm{e}^{-rT}\big[S(T)-K\big]^+$ and $S(t) = S_1(t)+\cdots+S_{10}(t)$, where the log-returns of the 10 spot prices are geometric Brownian motions and we for simplicity assume independence and that the yearly volatilities σ_i all equal $\sigma = 0.25$. The initial spot prices are $6, \ldots, 15$, we take $T = 2$ years, $r = 4\,\%$, and $K = 300$. Since $S(0) = 105$, we are thus in the “out-of-the-money” case. The assignment is to illustrate the efficiency of importance sampling by comparing the half-width of the confidence interval for the price Π to that of the crude Monte Carlo method. The importance distribution (this is only one possibility) is obtained by adding the same μ to the drifts $r - \sigma_i^2$ under the risk-neutral measure, with μ determined from pivotal runs such that $\widetilde{\mathbb{E}}S(T) \approx K$.

Solution.

$K = 300$ is unrealistically out-of-the-money so we took $K = 150$ instead. With $K = 150$ one finds through pilot runs that adding $\mu = 0.142$ to the drifts of all ten assets ensures approximate fulfillment of the condition $\widetilde{\mathbb{E}}S\,(T) \approx K$. [1] A comparison of the first 50 sample paths under the crude Monte Carlo method and under the Monte Carlo method with importance sampling is given in Fig. 2.1.

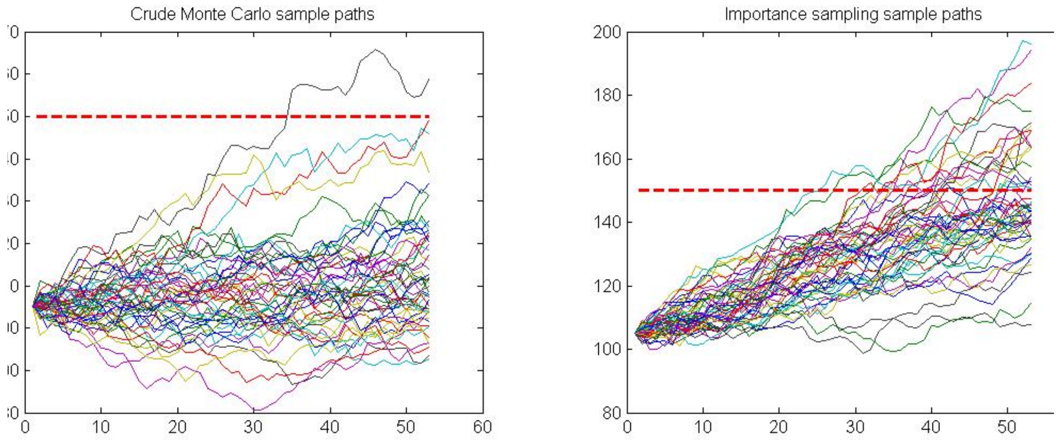


Figure 2.1

One sees that $Z_{\mathrm{CMC}} = 0$ (corresponding to $S(T) \le K$) in many replications, which one therefore somehow feel are vasted, and that IS resolves this problem.

The resulting gain in efficiency is apparent from the confidence intervals and corresponding half-widths reported in Table 2.4.

	$\hat{q}_{0.025}$	$\hat{z}$	$\hat{q}_{0.975}$	$\widetilde{\mathbb{P}}\,[S\,(T) > K]$	$\widetilde{\mathbb{E}}S\,(T)$	*Half-width*
CMC	$4.96 \cdot 10^{-2}$	$5.44 \cdot 10^{-2}$	$5.92 \cdot 10^{-2}$	0.0071	113.3	0.0132
IS	$5.30 \cdot 10^{-2}$	$5.40 \cdot 10^{-2}$	$5.51 \cdot 10^{-2}$	0.4685	149.7	0.0028

Table 2.4

There is in fact an optimal choice of $\widetilde{\mathbb{P}}$ or, equivalently, $\widetilde{L} = \mathrm{d}\widetilde{\mathbb{P}}/\mathrm{d}\,\mathbb{P}$: Assuming $Z \ge 0$ for simplicity, let $\mathbb{P}^*$ be defined by

$$\frac{\mathrm{d}\,\mathbb{P}^*}{\mathrm{d}\,\mathbb{P}} = \frac{Z}{z}, \qquad \text{i.e., } \mathbb{P}^*(\mathrm{d}\omega) = \frac{Z}{z}\,\mathbb{P}(\mathrm{d}\omega)$$

or $L^* = \mathbb{E}\,|Z|/|Z|$. Then ZL has variance 0 under $\widetilde{\mathbb{P}}$. The optimal choice $\widetilde{\mathbb{P}} = \mathbb{P}^*$ can, however, never be implemented in practice, since the evaluation of the estimator involves knowledge of the unknown z. Nevertheless, it is suggested that large variance reduction can be achieved by sampling outcomes $\omega \in \Omega$ in rough proportion to $Z(\omega)$. If $z = \mathbb{P}(A)$, $Z = \mathbb{1}_A$, then

$$\mathbb{P}^*(\mathrm{d}\omega) = \frac{\mathbb{1}\{\omega \in A\}}{\mathbb{P}(A)}\,\mathbb{P}(\mathrm{d}\omega)\,,$$

[1] The equation $\widetilde{\mathbb{E}}S\,(T) \approx K$ can in fact be solved explicitly for μ, but in many other examples, one has to resort to pivotal runs.

so that $\mathbb{P}^*(\cdot) = \mathbb{P}(\cdot \mid A)$. Thus, when computing probabilities, we wish to use a sampling distribution $\widetilde{\mathbb{P}}$ that resembles as closely as possible the conditional distribution of $\mathbb{P}$ given A. In Exercise AG.V.1.4 on Gaussian probabilities, we are not aware of any reasonably precise description of the distribution of $\boldsymbol{X} = (X_1\ X_2)$ given $X_1 \geq a, X_2 \geq a$. The proposed IS schemes aim at coming somewhat in that direction, with a modest level of ambition. Similar remarks apply in the option pricing exercise AG II.2.9, where one would ideally try to describe the conditional distribution of $S_1(T), \ldots, S_{10}(T)$ given $S_1(T) + \cdots + S_{10}(T) > K$.

2.2 Control variates

The idea is to look for an r.v. W that has a strong correlation (positive or negative) with Z and a *known* mean w, generate $(Z_1, W_1), \ldots, (Z_R, W_R)$ rather than $Z_1, \ldots, Z_R$, and combine the empirical means $\widehat{z}, \widehat{w}$ to an estimator with lower variance than the CMC estimator $\widehat{z}$ of $z = \mathbb{E}\, Z$.

The naive method is to choose some arbitrary constant α and consider the estimator $\widehat{z} + \alpha(\widehat{w} - w)$. The point is that since w is known, we are free just to add a term $\alpha(\widehat{w} - w)$ with mean zero to the CMC estimator $\widehat{z}$, so that unbiasedness is preserved. The variance is

$$\sigma_Z^2 + \alpha^2\sigma_W^2 + 2\alpha\sigma_{ZW}^2, \tag{2.1}$$

where

$$\sigma_Z^2 \overset{\text{def}}{=} \mathbb{V}\text{ar}\, Z, \qquad \sigma_W^2 \overset{\text{def}}{=} \mathbb{V}\text{ar}\, W, \qquad \sigma_{ZW}^2 \overset{\text{def}}{=} \mathbb{C}\text{ov}(Z, W).$$

In general, nothing can be said about how (2.1) compares to the variance σ_Z^2 of the CMC estimator $\widehat{z}$ (though sometimes a naive choice such as $\alpha = 1$ works to produce a lower variance). However, it is easily seen that (2.1) is minimized for $\alpha = -\sigma_{ZW}^2/\sigma_W^2$, and that the minimum value is

$$\sigma_Z^2(1 - \rho^2)\,, \quad \text{where} \quad \rho \overset{\text{def}}{=} \mathbb{C}\text{orr}(Z, W) = \frac{\sigma_{ZW}^2}{\sqrt{\sigma_Z^2\sigma_W^2}}\,. \tag{2.2}$$

One then simply estimates the optimal α by replacing σ_{ZW}^2, σ_W^2 by their empirical values,

$$\widehat{\alpha} \overset{\text{def}}{=} -\frac{s_{ZW}^2}{s_W^2},$$

where

$$s_Z^2 \overset{\text{def}}{=} s^2 \overset{\text{def}}{=} \frac{1}{R-1}\sum_{r=1}^{R}(Z_r - \widehat{z})^2, \qquad s_W^2 \overset{\text{def}}{=} \frac{1}{R-1}\sum_{r=1}^{R}(W_r - \widehat{w})^2\,,$$

$$s_{ZW}^2 \overset{\text{def}}{=} \frac{1}{R-1}\sum_{r=1}^{R}(Z_r - \widehat{z})(W_r - \widehat{w})\,,$$

and uses the estimator $\widehat{z}_{\mathrm{CV}} = \widehat{z}+\widehat{\alpha}(\widehat{w}-w)$, which has the same asymptotic properties as $\widehat{z}+\alpha(\widehat{w}-w)$; in particular, the asymptotic variance is $\sigma_Z^2(1-\rho^2)/R$, and a confidence interval is constructed by replacing σ_Z^2, ρ^2 by their empirical values s_Z^2, $s_{ZW}^4/s_Z^2 s_W^2$.

The procedure reduces the variance by a factor $1-\rho^2$. Thus, one needs to look for a control variate W with $|\rho|$ as close to 1 as possible. The exact value of ρ will be difficult to asses a priori, so that in practice one would just try to make W and Z as dependent as possible (in some vague sense). It is, however, an appealing feature that even if one is not very successful, the resulting variance is never increased.

Example 4.

A famous example of control variates occurs in Asian options, where the key step in estimating the price is evaluating the expected value of $[S(0)A-K]^+$, where $A \overset{\text{def}}{=} \sum_1^p \mathrm{e}^{B(iT/p)})/p$ is the average of a discretely sampled geometric Brownian motion $\{B(t)\}$, with drift say μ and variance σ^2 ($S(0)>0, K, T$ are constants). The idea is that whereas the distribution of A is intractable, such is not the case for the geometric average

$$A^* \overset{\text{def}}{=} \Bigl(\prod_{i=1}^p \mathrm{e}^{B(iT/p)}\Bigr)^{1/p} = \prod_{i=1}^p \mathrm{e}^{(p-i+1)Y_i/p}\,,$$

where $Y_i \overset{\text{def}}{=} B(iT/p)-B\bigl((i-1)T/p\bigr)$. Namely, since the Y_i are i.i.d. $\mathcal{N}(\mu T/p, \sigma^2 T/p)$, we have that $\log A^*$ is normal with mean and variance

$$\theta \overset{\text{def}}{=} \frac{\mu T}{p}\sum_{i=1}^p (p-i+1), \qquad \text{respectively} \qquad \omega^2 \overset{\text{def}}{=} \frac{\sigma^2 T}{p^2}\sum_{i=1}^p (p-i+1)^2$$

(θ, ω^2 can be reduced but we omit the details). Thus, we can take $W \overset{\text{def}}{=} \bigl[S(0)A^*-K\bigr]^+$ as control variate, since the expectation

$$\int_{\log\left(K/S(0)\right)}^{\infty} (s_0\mathrm{e}^z - K)\frac{1}{\sqrt{2\pi\omega^2}}\mathrm{e}^{-(z-\theta)^2/2\omega^2}\,\mathrm{d}z$$

is explicitly available by the Black–Scholes formula (Appendix A2).

Exercise Sept. 20, 2012.

Redo the basket option, Ex. AG.V.1.8, with the following modifications:

(1) Take K so that the option is in-the-money (say $K=100$), and forget about importance sampling.

(2) Use both uncorrelated logreturns as in V.1.8 and a correlation between all logreturns of 0.38.

(3) Use $W = \mathrm{e}^{-rT}[A^*-K]^+$ as control variate, where A^* is the geometric average G of the 10 asset prices.

(4) Supplement with W^2 as control.

Solution.

The CMC estimator can be written as

$$Z = \mathrm{e}^{-rT}[A-K]^+, \qquad A = \tfrac{1}{10}10\, S_i(0)\mathrm{e}^{Y_i}\,.$$

The corresponding geometric average is

$$A^* \stackrel{\text{def}}{=} \Bigl(\prod_{i=1}^{10} \bigl(10\, S_i(0)\mathrm{e}^{Y_i}\Bigr)^{1/10} = x\mathrm{e}^{Y} \qquad \text{where } x = 10\Bigl(\prod_{i=1}^{10} \bigl(10\, S_i(0)\bigr)\Bigr)^{1/10}$$

and $Y = \sum_1^{10} Y_i/10$. Thus the proposed control W becomes $\mathrm{e}^{-rT}[x\mathrm{e}^Y - K]^+$. Under the risk-neutral measure P^*, $Y \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$ where

$$\mu_Y = \mathbb{E}^* Y = (r-\sigma^2/2)T\,, \qquad \sigma_Y^2 = \mathbb{V}\mathrm{ar}^* Y = \tfrac{1}{10}(1+9\rho)\sigma^2 T\,,$$

and the required expression for $\mathbb{E}\, W$ comes out from the Black-Scholes formula in the form given in Appendix A2.

The details for W^2 are rather similar.

The outcome of implementing the geometric mean as control is summarized in Table 2.5.

	$\hat{\alpha}$	$\hat{\rho}$	$1-\hat{\rho}^2$	w	$\hat{w}$	$w-\hat{w}$
Control variate	-2.9502	0.38481	0.85192	0.0014161	0.0014868	$7.0746 \cdot 10^{-5}$

Table 2.5

The resulting variance reduction is seen from Table 2.6.

	$\hat{q}_{0.025}$	$\hat{z}$	$\hat{q}_{0.975}$
Crude Monte Carlo	0.050644	0.055584	0.060524
Control variate	0.050816	0.055375	0.059935

Table 2.6

Exercise AG.V.7.1.

Suggest some variance reduction methods for evaluating

$$\int_0^\infty (x+0.02x^2)\exp\{0.1\sqrt{1+\cos x} - x\}\,\mathrm{d}x$$

by Monte Carlo integration.

Solution.

The integral can be simulated either as $\mathbb{E}\,h_1(X_1)$ or $\mathbb{E}\,h_2(X_2)$ where X_1 has density e^{-x} and X_2 has density $x\mathrm{e}^{-x}$, and

$$h_1(x) = (x + 0.02x^2)\mathrm{e}^{0.01\sqrt{1-\cos x}}\,, \qquad h_2(x) = (1 + 0.02x)\mathrm{e}^{0.01\sqrt{1-\cos x}}\,.$$

The second way is much better: the ratio of standard deviations came out as $1.15 : 0.04$.

The contribution from the 0.02 term is much smaller than the one from the preceeding one. When using $\cos X_2$ as control variate, one obtained $\rho^2 = 0.97$. One then needs the formula

$$\int_0^\infty \cos x\, x\mathrm{e}^{-x}\,\mathrm{d}x = 0\,.$$

There should be many more ideas for variance reduction!

Exercise AG.V.2.3, New exercise.

Consider Exercise AG.V.1.4 and let $S = X_1 + X_2$ and $D = X_1 - X_2$. Experiment with

$$\mathbb{1}_{\{X_1 \geq a\}}, \quad \mathbb{1}_{\{S \geq 2a\}}, \quad \mathbb{1}_{\{X_1 \geq a\}} + \mathbb{1}_{\{X_2 \geq a\}}, \quad \mathbb{1}_{\{|D| < c\}}$$

as control variates.

Solution.

Table 2.7 show simulation estimates of $\mathbb{P}(\mathbf{X} \in \mathbf{A})$ using the indicated W as control variate. As $\widehat{z}$ we employed Crude Monte Carlo.

W	$(1-\rho)^2$	$\widehat{z}_{CV}$	$\widehat{\sigma}^2_{\mathrm{CV}}$
0 (CMC)		$1.375000 \cdot 10^{-3}$	$1.373110 \cdot 10^{-3}$
$\mathbb{1}_{\{X_1 \geq a\}}$	0.981357	$1.332530 \cdot 10^{-3}$	$1.301513 \cdot 10^{-3}$
$\mathbb{1}_{\{S \geq 2a\}}$	0.981416	$1.328600 \cdot 10^{-3}$	$1.301591 \cdot 10^{-3}$
$\mathbb{1}_{\{X_1 \geq a\}} + \mathbb{1}_{\{X_2 \geq a\}}$	0.814486	$1.334705 \cdot 10^{-3}$	$1.080202 \cdot 10^{-3}$
$\mathbb{1}_{\{\lvert D\rvert < c\}}$	0.997934	$1.325516 \cdot 10^{-3}$	$1.323498 \cdot 10^{-3}$

Table 2.7

2.3 Stratification

My experience from teaching is that the general formulation of stratification as given in AG is difficult for the students to grasp. In the 2012 course and the EURANDOM minicourse, I instead started with presenting following exercise for afterwards to proceed to the general theory.

Poll exercise.

A survey sampling company is to make a poll for the 2011 election in a country like Denmark. It counts votes for Blue or Red block (the goal is to give an estimate of the precentage p of R votes together with an associated confidence interval), and it uses a sample of $2,000$. There is a total of 4 million voters, and we will imagine that 2.05 m (i.e., 51.25 %) would vote for R at the time of the poll (of course, this number is not known to the company!).

(1) The crudest method is to select $2,000$ people at random from the whole population and ask for their vote. Explain that this would give a confidence interval of order 2.1% [you don't need to simulate; this is a simple calculation in the binomial distribution].

(2) In practice polls, are often based on more detailed information. Let us assume that the company divides the communities in the country into three types, a: countryside, b: suburb or city with an annual income $>$ 300.000 DKK, c: suburb or city with an annual income $\leq$ 300.000 DKK, with 1.5, 1.0, resp. 1.5 m inhabitants, respectively. An alternative to 1) is then stratification with strata $\mathrm{a, b, c}$ and proportional allocation. Give a point estimate for p and an associated confidence interval, assuming that (at the time of the poll) 0.5 m in stratum a would be R-voters, 0.3 m in stratum b and 1.25 m in stratum c.

(3) Finally consider a poststratification scheme aB,aR,bB,bR,cB,cR where, e.g., cR stands for the group of voters in c who voted R at the 2007 election. Note that the number of voters is known, but the poll cannot select the cR-number R_{cR} in exact proportionality because for any c voter, it is not public whether he/she is cB or cR. Carry out the simulation assuming the following distribution of the voters (here cRB is the number of voters in c who voted R in 2007 and will vote B in 2011 etc.):

aBB	aBR	aRB	aRR	bBB	bBR	bRB	bRR	cBB	cBR	cRB	cRR
0.95	0.15	0.05	0.35	0.65	0.15	0.05	0.15	0.10	0.20	0.15	1.05

Solution.

(1) is just binomial sampling, so the half-width of the confidence interval is $1.96\sqrt{p(1-p)}/\sqrt{2000} = 0.22$. Proportional allocation in (2) gave 0.19, and post-stratification 0.16.

This reduction may not appear much, and is certainly not so either in a Monte Carlo context. However, in survey sampling it is worthwhile: the sample size N needed to get 0.16 in the simple scheme (1) is given by $0.22/0.16 = \sqrt{N/2000}$ which gives $N = 3438$, implying an added cost of 75% for the company.

Exercise (new).

Consider again the Gaussian problem $\mathbb{P}(\boldsymbol{X} \in \boldsymbol{X})$ in Exercise AG.V.1.4 and the quantities

$$\mathbb{1}_{\{X_1 \geq a\}}, \quad \mathbb{1}_{\{S \geq 2a\}}, \quad \mathbb{1}_{\{X_1 \geq a\}} + \mathbb{1}_{\{X_2 \geq a\}}, \quad \mathbb{1}_{\{|D| < c\}}$$

that we earlier tried as controls. Use them instead for stratification.

Solution.

Consider the following sets:

$$\begin{aligned}
\Omega_0 &= \{S \leq 2a\} \\
\Omega_{11} &= \{6 < S \leq 7, |D| \leq 3\} \\
\Omega_{12} &= \{6 < S \leq 7, 3 < |D| \leq 5\} \\
\Omega_{13} &= \{6 < S \leq 7, 5 < |D|\} \\
\Omega_{21} &= \{7 < S \leq 8, |D| \leq 3\} \\
\Omega_{22} &= \{7 < S \leq 8, 3 < |D| \leq 5\} \\
\Omega_{23} &= \{7 < S \leq 8, 5 < |D|\} \\
\Omega_{31} &= \{8 < S, |D| \leq 3\} \\
\Omega_{32} &= \{8 < S, 3 < |D| \leq 5\} \\
\Omega_{33} &= \{8 < S, 5 < |D|\}
\end{aligned}$$

We observe that some of above sets are disjoint with respect to A. That is,

$$\mathbb{P}((\Omega_0 \cup \Omega_{12} \cup \Omega_{13} \cup \Omega_{22} \cup \Omega_{23}) \cap A) = 0.$$

and therefore a further variance reduction can be obtained by noting that

$$\begin{aligned}
\mathbb{P}(\boldsymbol{A}) &= \mathbb{P}((\Omega_{11} \cup \Omega_{21} \cup \Omega_{31} \cup \Omega_{32} \cup \Omega_{33}) \cap \boldsymbol{A}) \\
&= \mathbb{P}(\boldsymbol{A}|\Omega_{11})p_{11} + \mathbb{P}(\boldsymbol{A}|\Omega_{21})p_{21} + \mathbb{P}(\boldsymbol{A}|\Omega_{31})p_{31} \\
&\quad + \mathbb{P}(\boldsymbol{A}|\Omega_{32})p_{32} + \mathbb{P}(\boldsymbol{A}|\Omega_{33})p_{33}
\end{aligned}$$

where $p_{ij} = \mathbb{P}(\Omega_{ij})$. So, the last expression can be used to build a estimator via stratification provided that we count with a method to simulate from every set Ω_{ij} (a description on how to simulate from these r.v.'s is found at the end of the exercise). The estimator is as follows

$$\widehat{z}_{\mathrm{ST}} = \widehat{z}_{11}p_{11} + \widehat{z}_{21}p_{21} + \widehat{z}_{31}p_{31} + \widehat{z}_{32}p_{32} + \widehat{z}_{33}p_{33}$$

where $\widehat{z}_{ij}$ is the Crude Monte Carlo estimate of $\mathbb{P}(\boldsymbol{A}|\Omega_{ij})$. However, since we have removed some of the original strata, the variance of the estimator takes the following shape

$$\widehat{\sigma}^2_{\mathrm{ST}} = \frac{(p_{11}/p)^2\widehat{\sigma}^2_{11}}{R_{11}/R} + \frac{(p_{21}/p)^2\widehat{\sigma}^2_{21}}{R_{21}/R} + \frac{(p_{31}/p)^2\widehat{\sigma}^2_{31}}{R_{31}/R} + \frac{(p_{32}/p)^2\widehat{\sigma}^2_{32}}{R_{32}/R} + \frac{(p_{33}/p)^2\widehat{\sigma}^2_{33}}{R_{33}/R}$$

where R_{ij} is the number of replications used for each estimator $\widehat{z}_{ij}$ and

$$p = p_{11} + p_{21} + p_{31} + p_{32} + p_{33}.$$

The number of replications R_{ij} using proportional allocation is given by

$$R_{ij} = \frac{p_{ij}\, R}{p_{11} + p_{21} + p_{31} + p_{32} + p_{33}}.$$

The results for this estimator are given in Table 2.8.

a	$\widehat{z}_{\mathrm{ST}}$	$\widehat{\sigma}^2_{\mathrm{ST}}$
3	$1.382303 \cdot 10^{-3}$	$1.733475 \cdot 10^{-6}$

Table 2.8

Exercise Dec 6, 2012.

Consider an European call option with payout $\left[e^Y - K\right]^+$ where Y has a NIG distribution with parameters $\mu = 0$,$\delta = 1$, $\alpha = 2$, $\beta = 1$ and K is chosen such that the option is in-the-money. Compute the expected payout using (a) crude Monte Carlo, (b) stratification of Y with proportional allocation, and report on the variance reduction. For simulation of Y as well as for the stratification, a chord algorithm may be relevant.

Solution.

The NIG draws can be generated in exactly the same way that the CGMY draws were generated in a previous exercise on random variate generation. Since the approach used in that exercise relied on splitting the support of the distribution into several segments, one can use these very segments for the stratification as well. Setting $K = 1$ and using 20 segments one achieves a considerable variance reduction with proportional proportional allocation as seen in Table 2.9.

	$\hat{q}_{0.025}$	$\hat{z}$	$\hat{q}_{0.975}$
Crude Monte Carlo	0.9861	1.0793	1.1725
Stratification	1.0215	1.0459	1.0703

Table 2.9

Brownian bisection and stratification

When considering stratification applied to BM, a difficulty is that when generating BM in (say) $[0, 1]$ from (say) $N = 1000$, it is infeasible to stratify all N increments since even just 2 strata for each would give a total of 2^N strata. If, as in many other contexts, one would concentrate on the most important variables, what comes to mind is to take first $B(1)$ as the most important, then $B(1/2)$, next $B(1/4), B(3/4)$, and so on. Values between the binary grid points can then be filled out with Brownian bridges or by continuing the bisection construction. See Fig. 2.2.[2]

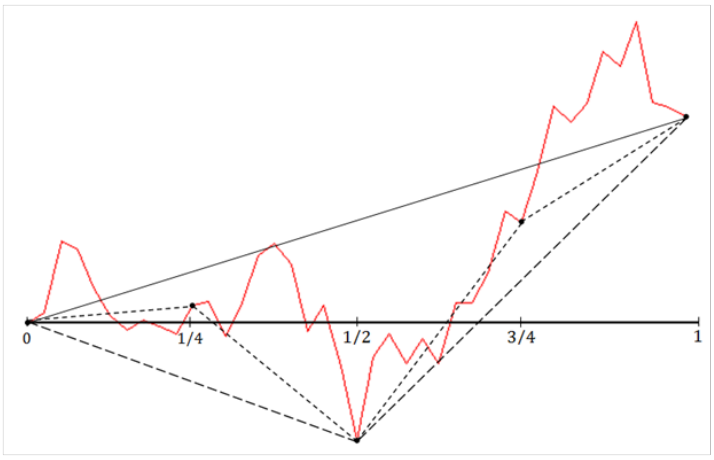


Figure 2.2

In each step of the bisection, one needs to fill in the midpoint between two binary grid points, which is easily done from the formula

$$B(t+h) \,\big|\, B(t)=a,\ B(t+2h)=b \sim \mathcal{N}((a+b)/2, h/2)\,.$$

Exercise AG.X.2.1.

Redo the Asian option in Exercise AG.IX.6.2 for $N = 6$ sampling points, using bisection and stratification. The simplest way may be to generate the Brownian motion at $N' = 8$ half-yearly sampling points and ignore the last two. The stratification can be done, for example, by taking eight strata for the r.v. generating $B(8)$, four for the one

Solution.

Stratification reduces the half-width of the confidence interval with a factor of about 2 (not very impressive!).

Exercise AG.X.2.1.

Redo Exercise AG.III.4.2 (the Kolmogorov–Smirnov test) using bisection and stratification. The stratification can be done, for example, by taking eight strata for the r.v. generating, $B(1/2)$, and four for each of the ones for $B(1/4), B(3/4)$. What about $B(1)$?

[2] That the minimum of B occurs at 1/2 as on the Figure is of course an event of probability zero!

Solution.

$\mathbb{E}\,M$ was estimated to about 0.83. Stratification reduces the variance with a factor of about 2 (not very impressive!).

2.4 Conditional Monte Carlo

Here $Z = Z_{\mathrm{CMC}}$ is replaced by $Z_{\mathit{Cond}} \overset{\mathrm{def}}{=} \mathbb{E}\big[Z_{\mathrm{CMC}} \,\big|\, W\big]$ for some r.v. W (more generally, one could consider $\mathbb{E}\big[Z_{\mathrm{CMC}} \,\big|\, \mathscr{G}\big]$ for some σ-field $\mathscr{G}$). Clearly, $\mathbb{E}\,Z_{\mathrm{Cond}} = \mathbb{E}\,Z_{\mathrm{CMC}} = z$. Since

$$\begin{aligned}\sigma^2_{\mathrm{CMC}} &= \mathbb{V}\mathrm{ar}(Z_{\mathrm{CMC}}) = \mathbb{V}\mathrm{ar}\big(\mathbb{E}\big[Z_{\mathrm{CMC}} \,\big|\, W\big]\big) + \mathbb{E}\big(\mathbb{V}\mathrm{ar}\big[Z_{\mathrm{CMC}} \,\big|\, W\big]\big)\\ &= \sigma^2_{\mathit{Cond}} + \mathbb{E}\big(\mathbb{V}\mathrm{ar}\big[Z_{\mathrm{CMC}} \,\big|\, W\big]\big) \geq \sigma^2_{\mathit{Cond}},\end{aligned}$$

conditional Monte Carlo always provides variance reduction, which is appealing. The difficulty is to find W such that the conditional expectation is computable.

Example 5.

Consider an option with expected pay-out $z = \mathbb{E}[S(0)\exp\{\mu T + \sigma B(T) - \alpha N(T)\} - K]^+$ for some suitable probability distribution $\mathbb{P}$, where B is standard Brownian motion and N an independent Poisson(λ) process. This can be seen as the Black-Scholes model with independent disasters at the epoch of N, such that a disaster decreases the asset price by a factor $\mathrm{e}^{-\alpha}$. We use conditional Monte Carlo with $W = N(T)$. Then

$$\begin{aligned}&\mathbb{E}\Big[\big[S(0)\exp\{\mu T + \sigma B(T) - \alpha N(T)\} - K\big]^+ \,\Big|\, N(T)\Big]\\ &\qquad = \mathbb{E}\big[\big[x\exp\{\mu T + \sigma B(T))\} - K\big]^+\end{aligned}$$

where $x = S(0)\exp\{-\alpha N(T)\}$ should be treated as a constant. Here the r.h. expectation can indeed be evaluated explicitly and is given by the Black-Scholes formula as in the Appendix. Thus, we don't need to simulate both $B(T)$ and $N(T)$, but can simulate only $N(T)$.

Example 6.

Consider an option with expected pay-out $\mathbb{E}\big[S(0)\mathrm{e}^{x(T)} - K\big]$ where $x(0) = 0$,

$$\mathrm{d}X(t) = \mu\,\mathrm{d}t + \mathrm{e}^{V(t)}\,\mathrm{d}B(t)$$

with V independent of B. This is a stochastic volatility model. One can use conditional Monte Carlo, conditioning on the whole path of V, noting that the conditional distribution of $\int_0^T \mathrm{e}^{V(t)}\,\mathrm{d}B(t)$ given V is normal with mean zero and variance $\int_0^T \mathrm{e}^{2V(t)}\,\mathrm{d}t$ and using again Black-Scholes.

Exercise V.4.4, new.

Consider Exercise V.1.4 and $S := X_1 + X_2$ and $D := X_1 - X_2$.

(i) Show that S and D are independent.

(ii) Show that the region $\boldsymbol{A}$ can be rewritten in terms of the variables S and D as

$$\boldsymbol{A} = \{(S, D) : 2a - S \leq D \leq S - 2a\}$$

(note that the condition $S \geq 2a$ is implicit), so we have

$$\begin{aligned}\mathbb{P}(\boldsymbol{X} \in \boldsymbol{A}) &= \mathbb{P}(2a - S \leq D \leq S - 2a) \\ &= \int_{2a}^{\infty} \mathbb{P}(2a - S \leq D \leq S - 2a | S = s) f_S(s) ds\end{aligned}$$

(iii) Implement conditional Monte Carlo, conditioning on S.

Solution.

1. It is well-known that S and D have a multivariate normal distribution. Therefore, in order to prove that those r.v.'s are actually independent it only remains to prove that their correlation is 0. Hence

$$\mathbb{C}\mathrm{orr}(S, D) = \mathbb{C}\mathrm{orr}(X_1 + X_2, X_1 - X_2) = \mathbb{V}\mathrm{ar}(X_1) - \mathbb{V}\mathrm{ar}(X_2) = 0$$

From a similar computation we obtain that $S \sim \mathcal{N}(0, 6)$ and $D \sim \mathcal{N}(0, 10)$.

2. It can be easily proved that with a change of variables the region $\boldsymbol{A}$ can be rewritten in terms of the variables S and D as

$$\boldsymbol{A} = \{(S, D) : 2a - S \leq D \leq S - 2a\}$$

(note that the condition $S \geq 2a$ is implicit), so we have

$$\begin{aligned}\mathbb{P}(\boldsymbol{X} \in \boldsymbol{A}) &= \mathbb{P}(2a - S \leq D \leq S - 2a) \\ &= \int_{2a}^{\infty} \mathbb{P}(2a - S \leq D \leq S - 2a | S = s) f_S(s) ds\end{aligned}$$

3. Easy.

4. Then, the conditional Monte Carlo algorithm is as follows

   (a) Simulate $S \sim \mathcal{N}(0, 6)$.

   (b) If $S \leq 2a$ return 0. Else return

$$\Phi\left(\frac{S - 2a}{\sqrt{10}}\right) - \Phi\left(\frac{2a - S}{\sqrt{10}}\right)$$

We obtained a point estimate of $1.385e - 03$ (as should be) and a variance of $4.583e - 04$, to be compared with the binomial variance of $1.384e - 03$.

3 Stochastic Differential Equations

3.1 Euler and Milstein

A stochastic differential equation (SDE) in one dimension has the form $X(0) = x_0$,

$$\mathrm{d}X(t) = a\bigl(t, X(t)\bigr)\,\mathrm{d}t + b\bigl(t, X(t)\bigr)\,\mathrm{d}B(t)\,, \qquad t \geq 0, \tag{3.1}$$

where $\{B(t)\}_{t\geq 0}$ is standard Brownian motion. The precise mathematical meaning is

$$X(t) = x_0 + \int_0^t a\bigl(s, X(s)\bigr)\,\mathrm{d}s + \int_0^t b\bigl(s, X(s)\bigr)\,\mathrm{d}B(s)\,, \qquad t \geq 0, \tag{3.2}$$

where the first integral is an ordinary integral and the second has to be interpreted in the Itô sense.

The numerical methods for SDEs are modeled after those for ODEs. The Euler method uses a tangential approximation

$$\int_0^h a\bigl(s, X(s)\bigr)\,\mathrm{d}s \approx h\,a\bigl(0, X(0)\bigr)\,, \quad \int_0^h b\bigl(s, X(s)\bigr)\,\mathrm{d}B(s) \approx B(h)\,b\bigl(0, X(0)\bigr) \tag{3.3}$$

for small h. This leads to The Euler scheme $X^h(0) = x_0$,

$$X_n^h = X_{n-1}^h + a\bigl(t_{n-1}^h, X_{n-1}^h\bigr)\,h + b\bigl(t_{n-1}^h, X_{n-1}^h\bigr)\,\Delta_n^h B\,,$$

where the $\Delta_n^h B$ are i.i.d. $\mathcal{N}(0, h)$ for fixed h and $X_n^h \overset{\text{def}}{=} X^h(t_n^h)$. When considering the time horizon $0 \leq t \leq 1$, we take $h = 1/N$ with $N \in \mathbb{N}$.

The proof that $X^h \to X$ as $h \downarrow 0$ is contained in the standard proof of the existence of a strong solution to (3.1) (regularity conditions are required). Thus, taking h small enough one is on safe grounds. Students often ask what 'small enough' means. The answer is that this depends on the time horizon and other features under study. A large value of h will produce an inaccurate approximation while a small one requires larger amounts of simulations. Fig. 3.1 illustrates some features of this balance. The aim in this particular problem [the submarine exercise below] is to estimate a certain probability p and the time horizon is of order 4.

The upper plot shows estimates of p as function of h, with the conclusion that step sizes smaller than e-03 will produce significative bias. The left plot in the bottom gives the corresponding variances, exhibiting similar bias behavior. Finally the right plot in the bottom shows the CPU time. We see that that this grows linearly with respect to the inverse size step size h, as was to be expected.

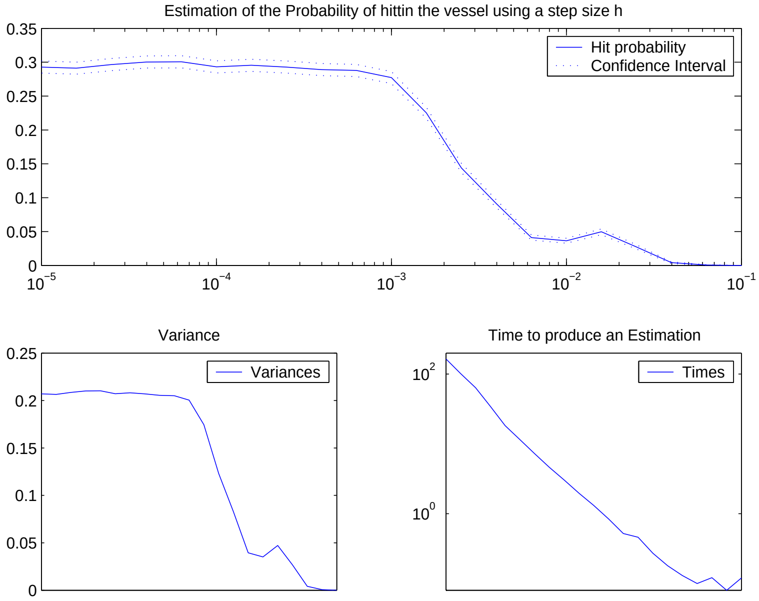


Figure 3.1

A frequently used refinement of Euler is the Milstein scheme. The idea is that the approximation

$$\int_0^h b\big(t, X(t)\big)\,\mathrm{d}B(t) \approx b\big(0, X(0)\big)B(h)$$

in (3.3) is the main source of error for the Euler scheme. To improve it, estimate the error by Itô's formula for $b\big(t, X(t)\big)$:[1]

$$\begin{aligned}
&\int_0^h b\big(t, X(t)\big)\,dB(t) - b\big(0, X(0)\big)B(h)\\
&\quad= \int_0^h \big\{b\big(t, X(t)\big) - b\big(0, X(0)\big)\big\}\,\mathrm{d}B(t)\\
&\quad= \int_0^h \Big\{\int_0^t \Big[b_t\big(s, X(s)\big) + a\big(s, X(s)\big)b_x\big(s, X(s)\big)\\
&\qquad+ \tfrac{1}{2}b^2\big(s, X(s)\big)b_{xx}\big(s, X(s)\big)\Big] + \int_0^t b\big(s, X(s)\big)b_x\big(s, X(s)\big)\,\mathrm{d}B(s)\Big\}\,\mathrm{d}B(t)
\end{aligned}$$

[1] The $\mathrm{O}(h^{3/2})$ term comes by noting that a Lebesgue integral $\int_0^t C(s)\,\mathrm{d}s$ is of order $\mathrm{O}(h)$ for $t \le h$ and an Itô integral $\int_0^t D(s)\,B(\mathrm{d}s)$ of order $\mathrm{O}(h^{1/2})$ since $B(t)$ has mean 0 and standard deviation $h^{1/2}$. Thus a double Lebesgue integral is of order $\mathrm{O}(h^2)$, a double Itô integral of order $\mathrm{O}(h)$ (dominating both $\mathrm{O}(h)$ and $\mathrm{O}(h^{3/2})$ for h small) and a mixed integral of order $\mathrm{O}(h^{3/2})$ (dominating $\mathrm{O}(h)$).

$$\begin{aligned}
&\sim \mathrm{O}(h^{3/2}) + b(0,x_0)b_x(0,x_0)\int_0^h\int_0^t \mathrm{d}B(s)\,\mathrm{d}B(t)\\
&\sim b(0,x_0)b_x(0,x_0)\int_0^h B(t)\,\mathrm{d}B(t) = b(0,x_0)b_x(0,x_0)\bigl\{\tfrac{1}{2}B(h)^2 - \tfrac{1}{2}h\bigr\}\,.
\end{aligned}$$

This leads to the Milstein scheme $X_0^h = x_0$,

$$X_n^h = X_{n-1}^h + ah + b\Delta_n^h B + \tfrac{1}{2}bb_x\{\Delta_n^h B^2 - h\}\,, \tag{3.4}$$

where $a = a(t_{n-1}^h, x_{n-1}^h)$ and similarly for b, b_x.

It seems intuitively obvious that Milstein must be better than Euler but we will see that there are in fact some twists to this. Discussions of properties and the comparison Milstein vs. Euler is usually carried out via the concepts of strong and weak error. The motivation and the precise definitions are as follows:

(s) X^h should be a good approximation of the sample path of X, that is, a good coupling, as measured by the strong error

$$e_{\mathrm{s}}(h) \overset{\mathrm{def}}{=} \mathbb{E}\bigl|X(1) - X^h(1)\bigr| = \mathbb{E}\bigl|X(1) - X_N^h\bigr|\,.$$

(w) $X^h(1) = X_N^h$ should give a good approximation of the distribution of $X(1)$. That is, $e_{\mathrm{w}}^g(h) = \bigl|\mathbb{E}\,g(X(1)) - \mathbb{E}\,g(X_N^h)\bigr|$ should be small for g smooth.

We say that X^h converges strongly to X at time 1 with order $\beta > 0$ if $e_{\mathrm{s}}(h) = \mathrm{O}(h^\beta)$, and weakly if for all g in a suitable class of smooth functions. It can then be proved that the Euler scheme converges strongly with order $\beta = 1/2$ and weakly with order $\beta = 1$, whereas the convergence order of Milstein is $\beta = 1/2$ in both the strong and weak sense.

Example 7.

Starting from the same $512 = 2^9$ i.i.d. $\mathcal{N}(0, 2^{-9})$ r.v.'s $V_1, \ldots, V_{512}$, we simulated geometric BM with $\mu = 2$, $\sigma^2 = 4$ in $[0,1]$ using the V_i as common random numbers for the updating. We took $h = 1/n$ with $n = 4, 8, \ldots, 512$ and implemented both Euler (dashed line) and Milstein (dot-dashed line); the solid line is interpolation between the exact value of GBM at the grid point (e.g. $\exp\{(\mu - \sigma^2/2)/4 + V_1 + \cdots + V_{128}\}$ at $t = 1/4$); the normal r.v.'s used in the updating for, for example, $h = 2^{-6}$ are $V_1 + \cdots + V_8$, $V_9 + \cdots + V_{16}$, etc.). The results are given in Fig. 3.2 and illustrate the better strong convergence properties of the Milstein scheme.

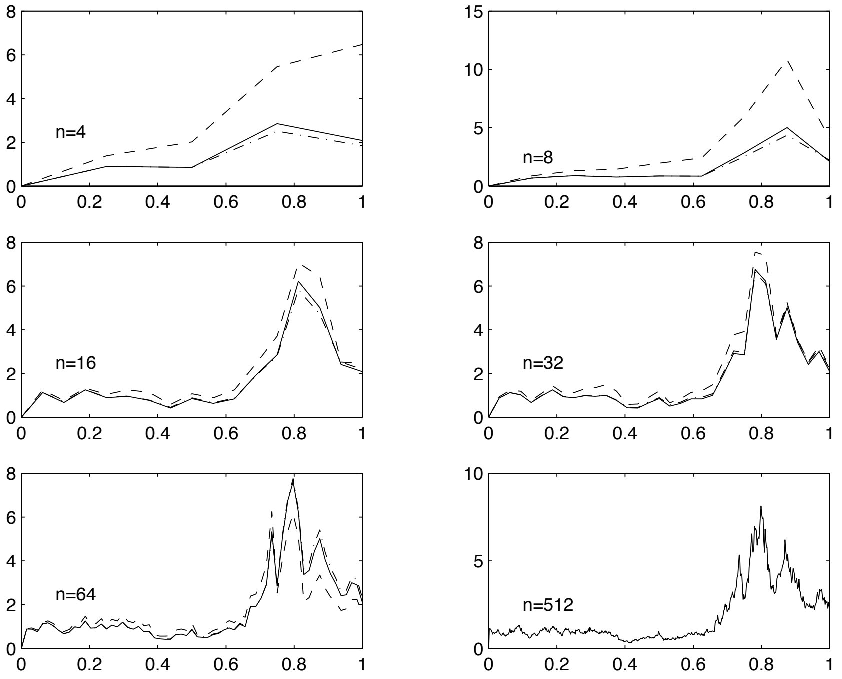


Figure 3.2

The example and the better strong rate of Milstein are, however, somewhat deceiving because most applications call for good weak properties and not strong. What is measured by the strong error is how well a given scheme performs in terms of reconstructing the exact solution as function of the Brownian motion driving the SDE. But in all examples we can think of, it is the approximation of the distribution of the SDE that matters rather than this coupling, and here the cited theoretical results on error rates do not give any argument favoring Milstein over Euler (maybe it is surprising that the weak order is not improved!)

Nevertheless, we recommend using Milstein whenever possible. The additional effort is certainly negligible – one needs an expression for the partial derivative b_x, but given this, all other quantities needed in the Milstein correction have already been computed for the use in Euler.

One important case where Milstein is not possible is multidimensions. For multidimensional coupled SDE's of the form

$$\mathrm{d}X_i(t) = a_i\bigl(t, \boldsymbol{X}(t)\bigr)\,\mathrm{d}t + \sum_{j=1}^{q} b_{ij}\bigl(t, \boldsymbol{X}(t)\bigr)\,\mathrm{d}B_j(t)\,, \qquad i = 1, \ldots, q\,, \tag{3.5}$$

the generalization of Euler is straightforward. However, Milstein gets into difficulties because the multidimensional Itô formula has a form that makes the correction term to Euler contain r.v.'s of the form

$$I_{jk} \;=\; \int_0^h B_k(s)\,\mathrm{d}B_j(s)\,,$$

whose density cannot be found in closed form when $j \neq k$ and for which there is no straightforward r.v. generation.

3.2 Three exercises

Exercise AG.X.4.2.

At time $t = 0$, a submarine located at $(0, 0)$ fires a torpedo against an enemy vessel whose midpoint is currently at $(0, 4)$ (the unit is km). The vessel is 0.14 km long, its speed measured in km/h at time t is $Z_1(t)$, a Cox–Ingersoll–Ross process with parameters $\alpha_1 = 6, c_1 = 30, \beta_1 = 1$, and the direction is given by the angle 30° NW. The information available to the submarine commander is a $\mathcal{N}(c_1, \sigma^2)$ estimate $\widehat{c}_1$ of c_1, where $\sigma^2 = 4$. The speed of the torpedo is another Cox–Ingersoll–Ross process $Z_2(t)$ with parameters $\alpha_2 = 60, c_2 = 60, \beta_2 = 7$, the angle (in radians!) giving the direction is $\theta(t) = \bigl(\theta(0) + \omega B(t)\bigr) \mod 2\pi$, where B is standard Brownian motion and $\omega^2 = 0.04$, and $\theta(0)$ is chosen by the submarine commander such that the torpedo would hit the midpoint of the vessel in the absence of stochastic fluctuations, that is, if the vessel moved with speed $\widehat{c}_1$, and the torpedo with constant direction $\theta(0)$ and speed c_2. See Figure 3.3.

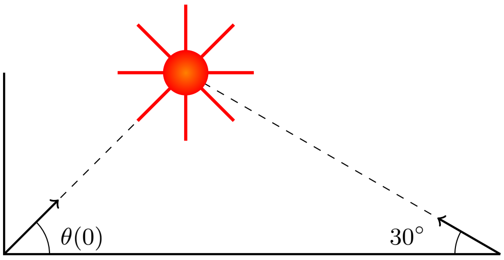


Figure 3.3

Compute the probability p that the torpedo hits the vessel, taking $Z_1(0) = c_1$, $Z_2(0) = c_2$.

Hint: Verify that (except in the extreme tails of $\widehat{z}$), $\theta(0)$ is the arcsin of $(\widehat{c}_1/2c_1)\sin 30^{\circ}$.

Solution.

Motivated from Fig. 3.1, we used step size $h = e - 04$ (that such a small h is needed may come as some of a surprise), giving a point estimate of order $\widehat{z} = 0.3$. Since we just are dealing with simple binomial sampling, the variance is of order $0.21/R$.

Fig. 3.4 shows plots of 50 different sample paths of the torpedo.

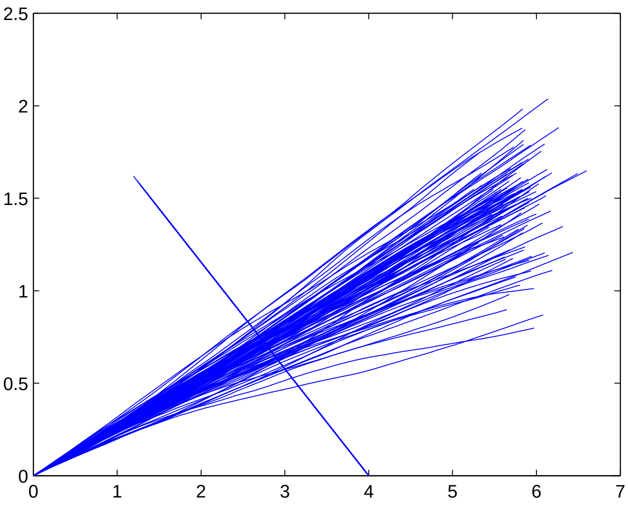


Figure 3.4

Exercise AG.X.4.1.

A bank wants to price its 5-year annuity loans in a market in which the short rate $r(t)$ at time t is stochastic. A loan is paid off continuously at a constant rate, say p, and thus the amount paid back is determined by

$$s(0) = 0, \qquad \mathrm{d}s(t) \;=\; \bigl(p + s(t)r(t)\bigr)\,\mathrm{d}t\,,$$

whereas an amount $q(0)$ kept in the bank will develop according to

$$\mathrm{d}q(t) = q(t)r(t)\,\mathrm{d}t.$$

Thus, for a loan of size $q(0)$ the payment rate p should be determined such that $Ec(5) = Eq(5)$ (ignoring profit and administration costs). To determine this, it suffices by an obvious proportionality argument to give estimates of the two expectations when $p = 1$, $q(0) = 1$.

Note that a short rate $r(t)$ corresponds to an interest per year of $\varepsilon = \mathrm{e}^{r(t)} - 1$.

The bank employs the Cox–Ingersoll–Ross process as model for $\{r(t)\}$. This means that we have a drift toward c, which we thus can interpret as the typical long-term interest rate and which the bank estimates corresponds to $\varepsilon = 6\,\%$; the interest rate at time 0 corresponds to $\varepsilon = 6.5\,\%$.

For your simulations of $\{r(t)\}$, use the Milstein scheme. Do first some pilot runs to determine (by sample path inspection) some values of the remaining two parameters α, β that appear to give reasonable fluctuations of $r(t)$. Compare finally your results with the deterministic values corresponding to $r(t) \equiv c$.

Solution.

Figure 3.5 gives a sample path of $\{r(t)\}_{0 \le t \le 5}$ corresponding to $\alpha = 1$, $\beta = 0.045$; the shape is what we consider realistic (fluctuations of reasonable size and speed).

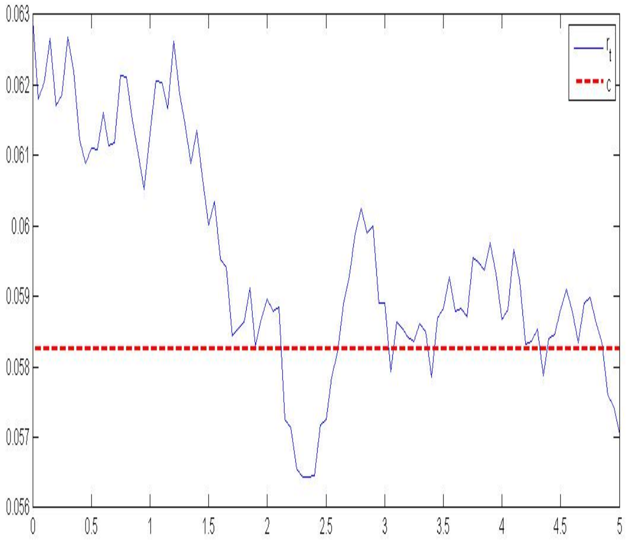


Figure 3.5

The 'obvious proportionality argument' goes a follows. Write $s_p(t)$ for $s(t)$ corresponding to a certain p and $q_b(t)$ for $q(t)$ corresponding to $q(0) = b$. For a given b, we are then looking for the solution $p^*(b)$ to $\mathbb{E}\, s_{p^*(b)}(5) = \mathbb{E}\, q_b(5)$. However, it is clear that $s_p(t) = ps_1(t)$, $q_b(t) = bq_1(t)$. Thus, the desired $p^*(b)$ is given by $b\,\mathbb{E}\, q_1(5)/\,\mathbb{E}\, s_1(5)$, so that it suffices to estimate $\mathbb{E}\, q_1(5)/\,\mathbb{E}\, s_1(5)$. To give a confidence interval, one must then use the delta method with $f(z_1, z_2) = z_1/z_2$, cf. the book pp. 75–76

Simulation is straightforward; since $b(x) = \beta\sqrt{x}$, the Milstein correction in X.(4.1) is just

$$\frac{1}{2}\beta\sqrt{x} \cdot \frac{\beta}{2\sqrt{x}}\{\Delta_n^h B^2 - h\} = \frac{\beta^2}{4}\{\Delta_n^h B^2 - h\}\,.$$

After $r(t)$ has been computed by Milstein, one just uses Euler for $s_1(t), q_1(t)$. With $h = 0.05$, the estimate of $\mathbb{E}\, q_1(5)/\,\mathbb{E}\, s_1(t)$ with $R = 10.000$ was 0.23 with a half-width of the confidence interval of order $e - 05$ [surprisingly narrow!].

With a constant r, we have $s_1(t) = (\mathrm{e}^{rt} - 1)/r$, $q_1(t) = \mathrm{e}^{rt}$, and with $r = 0.06$, this comes out very close to 0.23. That is, the stochastic interest makes little difference, at least with the chosen parameters.

Exercise AG.X.3.1.

Let $p(t, T)$ be the price at time t of a zero-coupon bond expiring at time $T > t$. The return on such a bond corresponds to a continuous interest rate of

$$r(t, T) \stackrel{\text{def}}{=} -\frac{1}{T - t}\log p(t, T)\,.$$

Typically, $r(t, T)$ depends not only on the short rate $r(t) = r(t, t+)$ at time t but also on T, and the curve $\{r(t, t + u)\}_{u \geq 0}$ is the *term structure* at time t.

Defining the instantaneous forward rate $f(t,T)$ as

$$f(t,T) \overset{\text{def}}{=} -\frac{\partial}{\partial T}\log p(t,T)\,, \quad \text{we have} \quad p(t,T) = \exp\Bigl\{-\int_t^T f(t,s)\,\mathrm{d}s\Bigr\}\,.$$

The (one-factor) *Heath–Jarrow–Morton model* postulates that for any fixed T,

$$\mathrm{d}f(t,T) = \alpha(t,T)\,\mathrm{d}t + v(s,T)\,\mathrm{d}B(t)\,, \tag{3.6}$$

where the driving BM is the same for all T.

To identify a risk-neutral measure $\mathbb{P}^*$, one combines the nonarbitrage argument with the identity

$$\exp\Bigl\{-\int_0^T f(0,s)\,\mathrm{d}s\Bigr\} = \mathbb{E}^*\exp\Bigl\{-\int_0^T r(s)\,\mathrm{d}s\Bigr\}\,,$$

which holds because both sides must equal $p(0,T)$. After some calculations, this gives that under $\mathbb{P}^*$, the $f(t,T)$ evolve as in (3.6) with $\alpha(t,T)$ replaced by

$$\alpha^*(t,T) \overset{\text{def}}{=} v(t,T)\int_t^T v(t,s)\,\mathrm{d}s\,.$$

For these facts and further discussion, see, e.g., the book by Björk.

Your assignment is to give projections (some typical sample paths) of the risk-neutral term structure $\{r(5,5+u)\}_{0\le u\le 10}$ after $t=5$ years, using the Vasicek volatility structure $v(t,T) = \beta\mathrm{e}^{-\alpha(T-t)}$ and the initial term structure $r(0,T) = \bigl(6+T/30-\mathrm{e}^{-T}\bigr)/100$, which has roughly the shape of the data in Jarrow's book p. 3. The parameters α,β should be calibrated so that sample paths of the short rate in $[0,5]$ look reasonable.

Generate, for example, the $r(5,5+u)$ at a quarter-yearly grid and use 10 yearly grid points for the $f(s,T)$. Thus, you will need to calculate the $f(i/10,5+j/4)$ for $i=1,\dots,50$, $j=1,\dots,40$. Note that the initial values $f(0,T)$ are analytically available from the expression for $r(0,T)$. For calibration of a,b, use $f(t-1/10,t)$ as approximation for $r(t)$.

Solution.

To obtain the risk-neutral term structure, we simulate $f(t,s)$ along the grid (different from the one suggested in the exercise text but more convenient) spanned by $t\in(0,0.1,\dots,5)$ and $T\in(5,5.25,\dots,15)$:

(t,T)	5	5.25	$\cdots$	15
0	○	○	$\cdots$	○
0.1				
$\vdots$				
5	●	●	$\cdots$	●

The risk-neutral term structure at time $t = 5$, that is $\{r(5, 5+u)\}_{0 \le u \le 10}$, is then found using the sequence $\{f(5, 5+u)\}_{0 \le u \le 10}$ (indicated with $\bullet$ in the grid) and the fact that yields and forward rates are related by the equation

$$r(t,T) = \frac{1}{T-t} \int_t^T f(t,s)\, \mathrm{d}s$$

which we approximate with

$$r(t,T) = \frac{1}{T-t} \sum_{t \le s \le T} f(t,s)\, \Delta s$$

where $\Delta s = 0.25$ is the time increment in the maturity (T) direction.

To initialize the construction of the grid we need the sequence $\{f(0,u)\}_{0 \le u \le 10}$ (indicated with $\circ$ in the grid). This is found by combining the inverse of the above relation, $f(t,T) = \frac{\partial}{\partial T}[r(t,T)(T-t)]$, with our knowledge of the initial term structure, $r(0,T) = \frac{1}{100}\left(6 + \frac{T}{30} - \mathrm{e}^{-T}\right)$, such that the time 0 forward rates can be found as

$$f(0,T) = \frac{1}{100}\left[6 + \frac{T}{30} - \mathrm{e}^{-T} + \left(\frac{1}{30} + \mathrm{e}^{-T}\right)(T-t)\right]$$

The remaining rows are found by iteratively using the Euler discretization of the SDE for the forward rate under the risk-neutral measure $\mathbb{P}^\star$. Here, the actual drift $\alpha(t,T)$ is replaced with the risk-neutral drift $\alpha^\star(t,T)$ such that (with time step h)

$$f(t+h,T) = \alpha^\star(t,T)\, h + v(t,T)\, \Delta_h B(t)\ , \qquad \Delta_h B(t) \sim N(0,h)$$

Note that the *same* Brownian increment is used across all maturities, T, since for a given t the forward rate curve is described by a deterministic function. With the Vasicek volatility structure the volatility is $v(t,T) = \beta \mathrm{e}^{-\alpha(T-t)}$ and the drift $\alpha^\star(t,T)$ is found from the HJM restriction

$$\alpha^\star(t,T) = v(t,T) \int_t^T v(t,s)\, ds = \frac{\beta^2}{\alpha} \mathrm{e}^{-\alpha(T-t)} \left(1 - \mathrm{e}^{-\alpha(T-t)}\right).$$

Finally, in order to assess the choice of the parameters, α and β, we consider the short rate path $r(t,t) = f(t,t)$. To obtain this path we extend the above grid to a trapez and find the short rate path as the sequence indicated with ×

$$\begin{array}{cccccccc}
(t,T) & 0 & 0.25 & 0.5 & \cdots & 5 & \cdots & 15 \\
0 & \times & & & & \circ & \cdots & \circ \\
0.1 & & & & & & & \\
\vdots & & \ddots & & & & & \\
0.5 & & & \times & & & & \\
\vdots & & & & \ddots & & & \\
5 & & & & & \times\bullet & \cdots & \bullet
\end{array}$$

Note that the forward rate is not defined in the triangle below the line of ×'s.

Choosing $\alpha = 1$ and $\beta = 0.03$ one obtains the series depicted in Figure 3.6 [of course, this is just one example because of randomness].

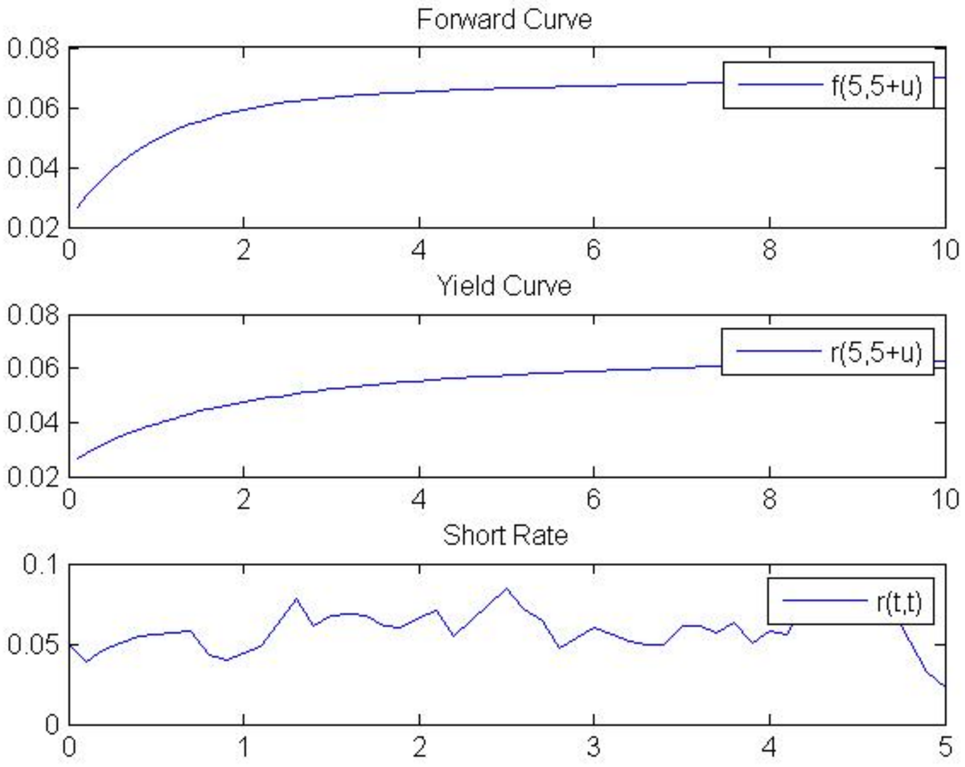


Figure 3.6

What about Euler vs Milstein and strong vs weak error in these three exercises? In the bank loan exercise, we are certainly asking for the distribution of smooth functionals so weak error is the relevant concept and there is no strong argument for Milstein. In the submarine exercise, we are again asking for distributional properties (a certain probability p), but an indicator function is not smooth and overall, the setting is more complex than looking at a smooth function at a fixed time as is done when considering weak error. In the HJM exercise, we are interested in $r(5, 5 + u)$ in a whole range of u-values, and since the driving BM is the same for different u, couplings and accordingly Milstein are potentially relevant.

4 Gaussian Processes

4.1 Cholesky factorization and fBM examples

A stochastic process $X = \bigl(X(t)\bigr)$ is *Gaussian* if for any $t_1, \ldots, t_n$ the joint distribution of $X(t_1), \ldots, X(t_n)$ is n-dimensional Gaussian. Time may be discrete, $t \in \mathbb{N}$ or $t \in \mathbb{Z}$, or continuous, $t \in [0, \infty)$ or $-\infty < t < \infty$.[1]

A stochastic process is determined by its finite-dimensional distributions, and multivariate Gaussian distributions are determined by the means and covariances. A Gaussian process is therefore specified by the $\mu(t) = \mathbb{E}\,X(t)$ and the $r(s,t) = \mathbb{C}\mathrm{ov}\bigl(X(s), X(t)\bigr)$.

A main example of a Gaussian process receiving much current attention in various application areas is *fractional Brownian motion*, where $\mu(t) = 0$,

$$r(t,s) = \frac{\sigma^2}{2}\bigl(|t|^{2H} + |s|^{2H} - |t-s|^{2H}\bigr) \tag{4.1}$$

The case $H = 1/2$ is standard Brownian motion, and H is called the *Hurst parameter*.

In this generality, the problem of simulating a Gaussian process (or rather a discrete skeleton on a finite time segment) is essentially equivalent to simulating from a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. This is available as the routine `mvn` in Matlab. The main method is Cholesky factorization, which is is an algorithm for writing a given symmetric $p \times p$ matrix $\boldsymbol{\Sigma} = (\Sigma_{ij})_{i,j=1,\ldots,p}$ as $\boldsymbol{\Sigma} = \boldsymbol{C}\boldsymbol{C}^{\mathsf{T}}$, where $\boldsymbol{C} = (c_{ij})_{i,j=1,\ldots,p}$ is (square) lower triangular ($c_{ij} = 0$ for $j > i$). One can then generate $\boldsymbol{X}$ as $\boldsymbol{\mu} + \boldsymbol{C}\boldsymbol{Y}$ where $\boldsymbol{Y} = (Y_1 \ \ldots \ Y_p)$ with $Y_1, \ldots, Y_p$ i.i.d. standard normal. Also Cholesky factorization is available in Matlab.

The main difficulty with Cholesky factorization is speed. The complexity is $\mathrm{O}(p^3)$ which quickly sets a limit for the dimension p with which one can deal in reasonable time (in the exercises, $p = 1000$ was feasible but $p = 10{,}000$ not).

Exercise Nov 22, 2012.

Consider an Asian option with payout $Z = [\mathrm{e}^{X(1)} + \cdots + \mathrm{e}^{X(12)} - K]^+$ where the X-process is fBM with the variance constant is chosen s.t. $\mathbb{V}\mathrm{ar}\bigl[X(12)\bigr] = 1$ and $K = 2$ [the value of K may be changed later]. Compute the expected payout for the Hurst parameter H taking values $0.1, 0.2, \ldots, 0.9$, using Cholesky factorizartion.

[1] Also the case of multidimensional time, for example $t \in \mathbb{R}^2$, is important for applications, and one then speaks of *Gaussian random fields*. We do not cover this here.

Solution.

Using (4.1), one can find the variance constant σ^2 by solving the equation

$$\mathbb{V}\mathrm{ar}[X(12)] = r(12,12) = \sigma^2 12^{2H} = 1.$$

That is, $\sigma^2 = 12^{-2H}$. Table 4.1 provides the expected payoffs for the nine different values of H

H	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$\mathbb{E}\,Z$	0.264	0.218	0.186	0.162	0.143	0.127	0.114	0.103	0.094

Table 4.1

Fig. 4.1 shows the first three sample paths for each value of H. Here it is clear that higher values of H are associated with considerably smoother paths, as is seen also from a figure in AG.

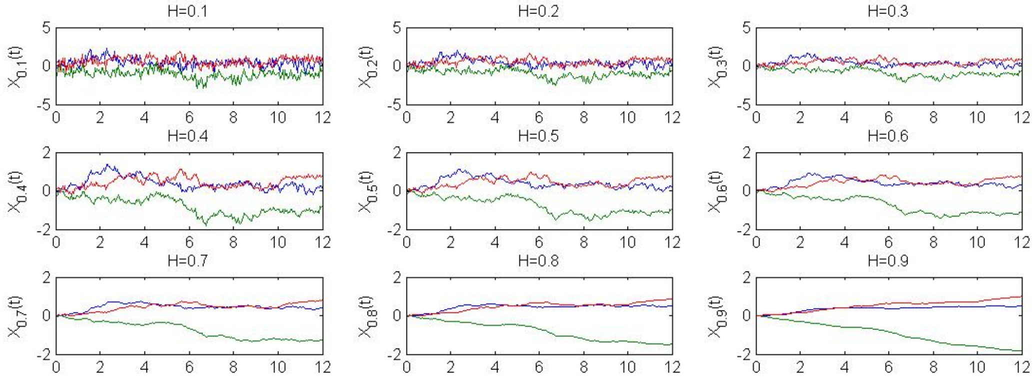


Figure 4.1

4.2 The stationary case

An important class of Gaussian processes is the stationary ones, where $\mu(t)$ does not depend on t and $r(s,t)$ only on $|t-s|$. Here methods faster than Cholesky factorization are available.

One such merthod is spectral simulation. Consider for simplicity the case of a discrete-time process $X_0, X_1, X_2, \ldots$ and write $r_k = r(k)$. Then the sequence $\{r_k\}$ is positive definite, and so by Herglotz's theorem, it can be represented as

$$r_k = \int_0^{2\pi} \mathrm{e}^{\mathrm{i}k\lambda}\,\nu(\mathrm{d}\lambda) \tag{4.2}$$

for some finite real measure ν on $[0, 2\pi)$, the *spectral measure.* The *spectral representation* of the process is

$$X_n = \int_0^{2\pi} \mathrm{e}^{\mathrm{i}n\lambda}\, Z(\mathrm{d}\lambda)\,, \tag{4.3}$$

where $\{Z(\lambda)\}_{\lambda\in[0,2\pi)}$ is a complex Gaussian process that is traditionally described as having increments satisfying

$$\begin{aligned} \mathbb{E}\bigl[\bigl(Z(\lambda_2) - Z(\lambda_1)\bigr)\overline{\bigl(Z(\lambda_4) - Z(\lambda_3)\bigr)}\bigr] &= 0, \\ \mathbb{E}\bigl|Z(\lambda_2) - Z(\lambda_1)\bigr|^2 &= \nu(\lambda_1, \lambda_2] \end{aligned} \tag{4.4}$$

for $\lambda_1 \le \lambda_2 \le \lambda_3 \le \lambda_4$. After some manipulations with complex numbers, this leads to the following procedure, where we assume the existence of a spectral density s: define Z_1, Z_2 by

$$\mathrm{d}Z_i(\lambda) = \sqrt{\tfrac{1}{2}s\bigl(B_i(\lambda)\bigr)}\,\mathrm{d}B_i(\lambda)\,,$$

where B_1, B_2 are independent standard Brownian motions. Then X can be constructed by

$$X_n = 2\int_0^{\pi} \cos(n\lambda)Z_1(\mathrm{d}\lambda) - 2\int_0^{\pi} \sin(n\lambda)Z_2(\mathrm{d}\lambda). \tag{4.5}$$

When simulating using (4.5), discretization is needed and for this reason spectral simulation is not exact (in particular, it destroyes the long-range dependence of fBM).

Another fast method is the stationary case is *circulant embeddings*, which has the advantage over Cholesky factorization of having a far better complexity, $\mathrm{O}(N\log N)$ compared to $\mathrm{O}(N^3)$ due to the fact that the needed matrix manipulations can be done via the FFT.

A *circulant* of dimension n is a $n \times n$ matrix of the form

$$\boldsymbol{C} = \begin{pmatrix} c_0 & c_{n-1} & \cdot & c_2 & c_1 \\ c_1 & c_0 & c_{n-1} & \cdot & c_2 \\ \cdot & c_1 & c_0 & \cdot & \cdot \\ c_{n-2} & \cdot & \cdot & \cdot & c_{n-1} \\ c_{n-1} & c_{n-2} & \cdot & c_1 & c_0 \end{pmatrix};$$

note the pattern of equal entries $c_{ij} \overset{\mathrm{def}}{=} c_k$ on $\{ij : i-j = k \bmod n\}$. Again, we label the rows and columns $0, 1, \dots, n-1$.

The first step is to embed the covariance matrix $\boldsymbol{\Sigma}$ of $X_0, \dots, X_N$ as the upper left corner of a circulant of order $2M$. It is easy to see that this is possible if and only if $M \ge N$. If $M = N$, the circulant $\boldsymbol{C}$ is unique and equals

$$\begin{pmatrix} r_0 & r_1 & \cdot & r_{N-1} & r_N & r_{N-1} & r_{N-2} & \cdot & r_2 & r_1 \\ r_1 & r_0 & \cdot & r_{N-2} & r_{N-1} & r_N & r_{N-1} & \cdot & r_3 & r_2 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ r_N & r_{N-1} & \cdot & r_1 & r_0 & r_1 & r_2 & \cdot & r_{N-2} & r_{N-1} \\ r_{N-1} & r_N & \cdot & r_2 & r_1 & r_0 & r_1 & \cdot & r_{N-3} & r_{N-2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ r_1 & r_2 & \cdot & r_N & r_{N-1} & r_{N-2} & r_{N-3} & \cdot & r_1 & r_0 \end{pmatrix}.$$

For example, if $N = 3$, $\boldsymbol{r} = (8\ 4\ 2\ 1)^{\mathsf{T}}$, then

$$\boldsymbol{\Sigma} = \begin{pmatrix} 8 & 4 & 2 & 1 \\ 4 & 8 & 4 & 2 \\ 2 & 4 & 8 & 4 \\ 1 & 2 & 4 & 8 \end{pmatrix} \quad \text{and} \quad \boldsymbol{C} = \begin{pmatrix} 8 & 4 & 2 & 1 & 2 & 4 \\ 4 & 8 & 4 & 2 & 1 & 2 \\ 2 & 4 & 8 & 4 & 2 & 1 \\ 1 & 2 & 4 & 8 & 4 & 2 \\ 2 & 1 & 2 & 4 & 8 & 4 \\ 4 & 2 & 1 & 2 & 4 & 8 \end{pmatrix} .$$

The idea of this circulant embedding is that the eigendecomposition $\boldsymbol{C} = \boldsymbol{F}\boldsymbol{\Lambda}\overline{\boldsymbol{F}}/n$ of $\boldsymbol{C}$, where $\boldsymbol{\Lambda}$ is the diagonal matrix with the eigenvalues on the diagonal, is easy to evaluate and that one thereby also obtains a square root $\boldsymbol{D}$ (in the sense that $\boldsymbol{D}\overline{\boldsymbol{D}}^{\mathsf{T}} = \boldsymbol{C}$), namely $\boldsymbol{D} \overset{\text{def}}{=} (1/n^{1/2})\boldsymbol{F}\boldsymbol{\Lambda}^{1/2}$, *provided all entries of* $\boldsymbol{\Lambda}$ *are nonnegative*, that is, provided the vector $\boldsymbol{F}\boldsymbol{c}$ is nonnegative. Of course, $\boldsymbol{D}$ is typically complex, but this is no problem: we may choose $\boldsymbol{\varepsilon} = (\varepsilon_0\,, \ldots, \varepsilon_{2N-1})$ with components that are i.i.d. and standard complex Gaussian (i.e., the real and the imaginary parts are independent $\mathcal{N}(0,1)$), and then the real and the imaginary parts, say $X_0, \ldots, X_{2N}$ and $X'_0, \ldots, X'_{2N}$, of $\boldsymbol{D}\boldsymbol{\varepsilon}$ will be (dependent) $2N$-dimensional Gaussian with covariance matrix $\boldsymbol{C}$. Therefore $X_0, \ldots, X_N$ and $X'_0, \ldots, X'_N$ have the desired distribution.

Exercise AG.XI.2.1 (extended).

Let X be a stationary Gaussian process with mean zero and covariance function

$$\gamma(s) = \gamma(t, t+s) = \begin{cases} (2+|s|)(1-|s|)^2 & -1 < s < 1, \\ 0 & |s| \geq 1. \end{cases}$$

Simulate a discrete skeleton of X using Cholesky factorization as well as circulant embeddings to get a Monte Carlo estimate of

$$z = \mathbb{P}\Big(\sup_{0 \leq t \leq 2} X(t) > 2\Big) .$$

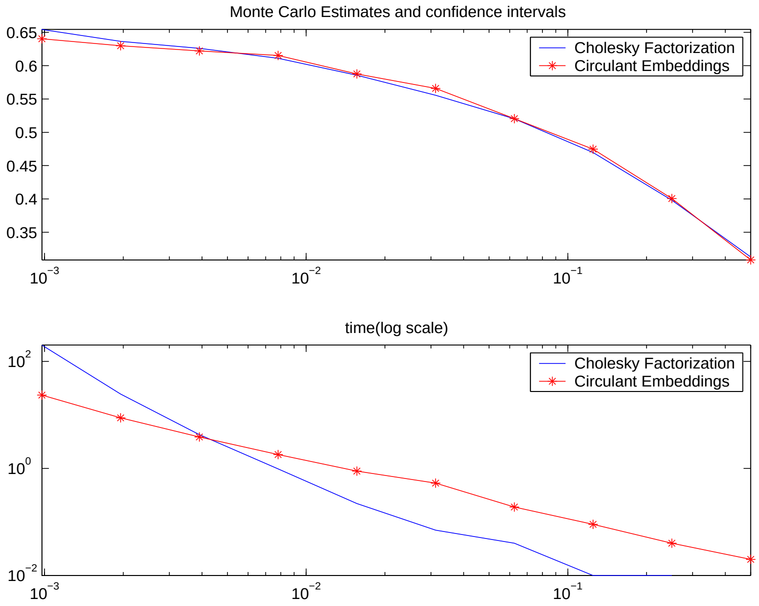
Solution.

Since we are using a discrete skeleton to produce an approximation of the real probability the accuracy depends largely on the size step h used for this purpose. The upper panel in Fig. 4.2 shows estimates of z as function of the step size h for both methods. It is seen that there could be some significative bias if we use step sizes smaller than 2^{-10}.

The second graph shows the time required to produce an estimate and illustrates the superiority of the circulant embeddings method over Cholesky factorization

The results in Table 4.2 are estimates using $h = 2^{-10}$.

Method	$\widehat{z}$	$\widehat{\mathbb{V}\mathrm{ar}}(\widehat{z})$
Cholesky Factorization	0.654	0.226
Circulant-Embeddings	0.640	0.230



Monte Carlo Estimates and confidence intervals

time(log scale)

Figure 4.2

Table 4.2

5 Conditional Monte Carlo and Heavy Tails

We consider the problem of estimating $z = \mathbb{P}(S_n > x)$ for large x, where $S_n = Y_1 + \cdots + Y_n$ with $Y_1, \ldots, Y_n$ i.i.d. with common distribution F and x is large so that z is small.

With light tails, there is a standard algorithm: consider the exponentially twisted distribution $F_\theta(\mathrm{d}y) = \mathrm{e}^{\theta y} F(\mathrm{d}y) / \mathbb{E}\, \mathrm{e}^{\theta Y}$, choose θ as solution of the equation $\mathbb{E}_\theta\, S_n = x$ and use importance sampling where the Y_k are simulated as i.i.d. $\sim F_\theta$.

In the heavy-tailed case, $\mathbb{E}\, \mathrm{e}^{\theta Y} = \infty$ for all $\theta > 0$, and exponential twisting is therefore impossible. Various importance sampling algorithm have been suggested, but we shall here concentrate on some conditional Monte Carlo ideas which are at the same time extremely efficient and easy to implement.

Assume for simplicity the existence of densities to avoid multiple ties and define $S_k = \sum_{j \le k} Y_k$, $M_k = \max_{j \le k} Y_k$. Using exchangeability in the first step,

$$\begin{aligned} z = \mathbb{P}(S_n > x) &= n\ \mathbb{P}(S_n > x, M_n = Y_n) \\ &= n\ \mathbb{E}\big[\mathbb{P}(S_n > x, M_n = Y_n \,|\, Y_1, \ldots, Y_{n-1})\big] \\ &= n\ \mathbb{E}\,\overline{F}\big(M_{n-1} \vee (x - S_{n-1})\big) \end{aligned}$$

where in the last step we noted that for $S_n > x, M_n = Y_n$ to occur, we must have $S_{n-1} + Y_n > x$, $Y_n > M_{n-1}$. We thereby arrive at the so-called *Asmussen-Kroese estimator*[1]

$$Z_{\mathrm{AK}}(x) = n\overline{F}\big(M_{n-1} \vee (x - S_{n-1})\big).$$

Before proceeding with the discussion of $Z_{\mathrm{AK}}(x)$, we need to survey some background material. F is said to belong to the *subexponential class* if

$$\frac{\overline{F}^{*2}(x)}{\overline{F}(x)} = \frac{\mathbb{P}(Y_1 + Y_2 > x)}{\mathbb{P}(Y > x)} \ \to\ 2\,.$$

The main cases are:

Regular variation: $\overline{F}(x) = L(x)/x^\alpha$ with L slowly varying (for example asymptotically constant or a power of the logarithm).

[1] S. Asmussen & D.P. Kroese (2006) Improved algorithms for rare event simulation with heavy tails. *Adv. Appl. Probab.* **38**, 545–558.

The lognormal case: $Y = \mathrm{e}^{\mu+\sigma V}$ with $V \sim \mathcal{N}(0,1)$.

Heavy Weibull tails: $\overline{F}(x) = \mathrm{e}^{-x^\beta}$ with $0 < \beta < 1$.

The definition of subexponentiality extends to $n > 2$: if F is subexponential, then

$$\frac{\overline{F^{*n}}(x)}{\overline{F}(x)} = \frac{\mathbb{P}(S_n > x)}{\mathbb{P}(Y > x)} \to n$$

The implication is that

$$z = z(x) = \mathbb{P}(S_n > x) \sim n\overline{F}(x)\,, \qquad x \to \infty\,. \tag{5.1}$$

The relative error $e(x)$ of a simulation estimator $Z = Z(x)$ for $z = z(x)$ is defined by $e^2(x) = \mathbb{V}\mathrm{ar}\, Z/z^2$, and Z is said to have *bounded relative error* if $\limsup_{x\to\infty} e(x) < \infty$. If instead only $\limsup_{x\to\infty} z(x)^\varepsilon e(x) < \infty$ for all $\varepsilon > 0$, one speaks of *logarithmic efficiency.*

Theorem 5.1. *The Asmussen-Kroese estimator has bounded relative error in the regularly varying case.*

Proof. If $M_{n-1} \le x/2(n-1)$, then $x - S_{n-1} > x/2$, implying $Z_{\mathrm{AK}} \le n\overline{F}(x/2)$. If instead $M_{n-1} > x/2(n-1)$ then $Z_{\mathrm{AK}}(x) \le n\overline{F}\big(x/2(n-1)\big)$. The result follows by combining (5.1) and the consequence $\limsup \overline{F}(ax)/\overline{F}(x) < \infty$ for all a of regular variation. □

AK also proved that there is logarithmic efficiency in the Weibull case, but the poof of this is much more difficult, and bounded relative error will in fact be established below.

Theorem 5.1 was improved by Hartinger & Kortschak (2009) who proved that there is in fact vanishing relative error, i.e. $e(x) \to 0$ as $x \to \infty$

We proceed to some recent analysis,[2] which gives precise estimates of $e(x)$ as well as improvements of Z_{AK}.

The analysis draws on methods from second order regular variation asymptotics which states that

$$\mathbb{P}(S_n > x) \;\sim\; n\overline{F}(x) + n(n-1)\,\mathbb{E}\,Y f(x)\,. \tag{5.2}$$

To illustrate the idea we sketch the proof of (5.2) for $n = 2$. One divides according to which of Y_1, Y_2 is the largest and whether the smaller one exceeds $x/2$ or not, and gets

[2] S. Asmussen & D. Kortschak (2012) On error rates in rare event simulation with heavy tails. *Proceedings of the Winter Simulation Conference.*
S. Asmussen & D. Kortschak (2013/14) Error rates and improved algorithms for rare event simulation with heavy Weibull tails. *Methodology and Computing in Applied Probability* (accepted). Available from `www.thiele.au.dk`.

$$
\begin{aligned}
&\mathbb{P}(Y_1+Y_2>x, Y_1>Y_2, Y_2\le x/2)\\
&\qquad = \int_0^{x/2} \overline{F}(x-y)\,F(\mathrm{d}y) \;=\; \int_0^{x/2} [\overline{F}(x)+yf(x)+\cdots]\,F(\mathrm{d}y)\,,\\
&\mathbb{P}(Y_1+Y_2>x, Y_1>Y_2, Y_2> x/2)\\
&\qquad = \mathrm{O}(\overline{F}(x/2)^2) = \mathrm{o}(f(x))\,,
\end{aligned}
$$

and so easy estimates give

$$
\begin{aligned}
&\mathbb{P}(Y_1+Y_2>x)\\
&\quad = 2\overline{F}(x) + 2\int_0^{x/2} [yf(x)+\cdots]\,F(\mathrm{d}y)\\
&\quad \sim 2\overline{F}(x) + 2\,\mathbb{E}\,Y\cdot f(x)
\end{aligned}
$$

as desired. Note that this Taylor expansion is not useful for light tails: here $\overline{F}(x), f(x)$ (and the higher order derivatives) are of the same order [just think of the exponential!].

We return to the AK estimator $Z(_{\mathrm{AK}}(x) = n\overline{F}\big(M_{n-1}\vee(x-S_{n-1})\big)$.

Theorem 5.2. *Assume $f(x)=\alpha L(x)/x^{\alpha+1}$, where f is the density of F. If $\alpha>2$ or, more generally, $\mathbb{E}[Y^2]<\infty$ then*

$$\mathbb{V}\mathrm{ar}\ Z_{\mathrm{AK}} \sim n^2\,\mathbb{V}\mathrm{ar}[S_{n-1}]f(x)^2 = n^2(n-1)\,\mathbb{V}\mathrm{ar}[Y_1]f(x)^2.$$

If $\alpha<2$ then $\mathbb{V}\mathrm{ar}\,Z_{\mathrm{AK}}\sim n^2(n-1)k_\alpha\overline{F}(x)^3$ where

$$k_\alpha = \Big(2^\alpha + \tfrac{1}{3}2^{3\alpha} - 2^{2\alpha} + \alpha\int_0^{1/2}\big((1-y)^{-\alpha}-1\big)^2 y^{-\alpha-1}dy\Big)\,.$$

The rates for $\mathbb{V}\mathrm{ar}\,Z_{\mathrm{AK}}$ in Theorem 5.2 have to be compared with the bounded relative error rate $L(x)^2/x^{2\alpha}$. For $\alpha>2$, one sees an improvement to $L(x)^2/x^{2\alpha+2}$, for $\alpha<2$ to $L(x)^3/x^{3\alpha}$. We next exhibit an estimator improving this rate for $1<\alpha<2$. The estimator is

$$Z = Z_{\mathrm{AK}} + n\,(\mathbb{E}\,S_{n-1}-S_{n-1})\,f(x) \tag{5.3}$$

Theorem 5.3. *Assume $f'(x)=-\alpha(\alpha-1)L(x)/x^{\alpha+2}$. If $\alpha>4$ or, more generally, $\mathbb{E}[Y^4]<\infty$, then the estimator in* (5.3) *satisfies*

$$\mathbb{V}\mathrm{ar}\,Z \sim \tfrac{1}{4}n^2\,\mathbb{V}\mathrm{ar}[S_{n-1}^2]f'(x)^2.$$

If $2<\alpha<4$ then $\mathbb{V}\mathrm{ar}\ Z\sim n^2(n-1)k_\alpha\overline{F}(x)^3$ where

$$k_\alpha = \alpha\int_0^\infty \big(((1-z)\vee z)^{-\alpha}-1-\alpha z\big)^2 z^{-\alpha-1}\mathrm{d}z\,.$$

The reduction of variance from bounded relative error to AK to (5.3) becomes

$$\frac{L(x)^2}{x^{2\alpha}} \to \frac{L(x)^3}{x^{3\alpha}} \to \frac{L(x)^4}{x^{4\alpha}}$$

We finally give a brief survey of the results for the Weibull case $\overline{F}(x) = \mathrm{e}^{-x^\beta}$ with $0 < \beta < 1$ (related distributions, say modified by a power, are easily included, but for simplicity, we refrain from this). The density is $f(x) = \beta x^{\beta-1}\mathrm{e}^{-x^\beta}$ and $f'(x) = -p(x)\overline{F}(x)$ where $p(x) = \beta^2 x^{2(\beta-1)} + \beta(1-\beta)x^{\beta-2}$.

Theorem 5.4. *If $0 < \beta < \log(3/2)/\log(2)$, then the Asmussen-Kroese estimator's variance is asymptotically given by*

$$\mathbb{V}\mathrm{ar}(Z_{\mathrm{AK}}) \sim n^2\,\mathbb{V}\mathrm{ar}(S_{n-1})f(x)^2.$$

Note that $\log(3/2)/\log(2) \approx 0.585$ was also found to be critical by Asmussen and Kroese to be the threshold for logarithmic efficiency to hold.

Theorem 5.5. *Denote with $f^{(k)}$ the k-th derivative of the density f. Define the estimator*

$$Z_m = Z_{\mathrm{AK}} + n\sum_{k=1}^{m}\frac{(-1)^{k-1}}{k!}\left(\mathbb{E}\,S_{n-1}^k - S_{n-1}^k\right)f^{(k-1)}(x). \tag{5.4}$$

If $0 < \beta < \beta_0$, then the estimator Z_m in (5.4) *has vanishing relative error. More precisely,*

$$\mathbb{V}\mathrm{ar}(Z_m(x)) \sim \frac{n^2}{(m+1)!^2}\,\mathbb{V}\mathrm{ar}((S_{n-1})^{m+1})f^{(m)}(x)^2.$$

Remark 5.6. The rates for the variances in Theorems 5.4 and 5.5 have to be compared with the bounded relative error rate $\mathrm{e}^{-2\beta}$. Note that $f(x) = \beta x^{\beta-1}\mathrm{e}^{-x^\beta}$ and $f^{(k)}(x) = (-1)^k p_k(x)\overline{F}(x)$ where p_k is regularly varying with index $(k+1)(\beta-1)$. Thus Z_{AK} improves the bounded relative error rate by a factor of $x^{1-\beta}$ and (5.4) by a factor of $x^{(k+1)(1-\beta)}$.

When $m = 1$, the estimator Z in (5.4) has the form $Z_{\mathrm{AK}} + \alpha(S_{n-1} - \mathbb{E}\,S_{n-1})$, so it is a control variate estimator, using S_{n-1} as control for Z_{AK}. It is natural to ask whether the $\alpha = -nf(x)$ at least asymptotically coincides with the optimal $\alpha^* = -\,\mathbb{C}\mathrm{ov}(Z_{\mathrm{AK}}, S_{n-1})/\,\mathbb{V}\mathrm{ar}(S_{n-1})$. This can indeed be checked to be the case.

Remark 5.7. In applications to ruin theory and the M/G/1 queue, the number n of terms in S_n is an independent r.v. With some effort, the theory can be refined to this case, but we will not present the details here.

We finally give a numerical example. In Table 5.1, the relative squared error of various of the estimators for $z = \mathbb{P}(S_n > x)$ is given for a Weibull Y with $\beta = 0.25$ and $n = 10$. The column $Z_{0,N}$ corresponds to the simple AK estimator Z_{AK} and $Z_{m,N}$ to the higher order estimator of order m in (5.4).

x	z	$Z_{0,N}$	$Z_{1,N}$	$Z_{2,N}$	$Z_{3,N}$	$Z_{4,N}$
666	0.077	0.11	1.08	936	8480000	$3.08 \cdot 10^{11}$
2493	0.0099	0.084	0.159	13.1	187000	$1.51 \cdot 10^{8}$
7412	0.001	0.0485	0.0318	1.44	164	33500
17,785	$1 \cdot 10^{-4}$	0.0225	0.00943	0.041	1.62	26.9
36,647	$1 \cdot 10^{-5}$	0.00944	0.00329	0.0053	0.0654	0.699
67,708	$1 \cdot 10^{-6}$	0.00396	0.0012	0.00151	0.00446	0.0886
115,355	$1 \cdot 10^{-7}$	0.00158	0.000506	0.00049	0.000729	0.000247
184,647	$1 \cdot 10^{-8}$	0.000598	0.000216	0.00015	$4.62 \cdot 10^{-5}$	$2.94 \cdot 10^{-5}$
281,317	$1 \cdot 10^{-9}$	0.000286	$9.41 \cdot 10^{-5}$	$5.77 \cdot 10^{-5}$	$2.29 \cdot 10^{-5}$	$7.02 \cdot 10^{-6}$
411,776	$1 \cdot 10^{-10}$	0.000116	$4.94 \cdot 10^{-6}$	$1.09 \cdot 10^{-6}$	$4.67 \cdot 10^{-7}$	$8.14 \cdot 10^{-7}$
583,108	$1 \cdot 10^{-11}$	$5.82 \cdot 10^{-5}$	$1.84 \cdot 10^{-6}$	$1.63 \cdot 10^{-5}$	$1.59 \cdot 10^{-8}$	$4.40 \cdot 10^{-8}$
803,069	$1 \cdot 10^{-12}$	$3.95 \cdot 10^{-5}$	$6.26 \cdot 10^{-7}$	$4.72 \cdot 10^{-7}$	$1.25 \cdot 10^{-9}$	$2.59 \cdot 10^{-10}$

Table 5.1: Relative squared error for Weibull Y with $\beta = 0.25$ and $n = 10$

The picture is that already $Z_{0,N}$ has excellent precision. Indeed, in the case of 10,000 replications the halfwidth of the confidence interval is

$$\frac{1.96\sqrt{0.084}}{\sqrt{10000}}z \;=\; 5.7 \cdot 10^{-6}\,, \quad \text{resp. } \frac{1.96\sqrt{3.95 \cdot 10^{-5}}}{\sqrt{10000}}z \;=\; 1.2 \cdot 10^{-16}$$

for $z = \mathbb{P}(S_n > x) = 10^{-3}$, resp. 10^{-12}, corresponding to 3, resp. 4 correct significant digits. For $z = 10^{-12}$, $Z_{1,N}$ gives an extra digit and $Z_{3,N}$ another one. Note also that in view of the computational simplicity of the estimators, 10,000 replications is very modest.

It is also seen that whereas the relative precision of $Z_{m,N} = Z_{m,N}(x)$ increases in x when m is fixed, there is some degradation as m increases with x fixed.

6 Example from Stochastic Optimization

The material on stochastic optimization was prepared by Leonardo Rojas-Nandyapa.

Exercise (new).

Assume that

$$X_1 \sim \text{Gamma}(2,2), \qquad X_2 \sim \text{Gamma}(2,1), \qquad X_3 \sim \text{Gamma}(2,2/3)$$

Find the θ^* optimizing the expression

$$z(\theta) = \mathbb{P}(\max\{\theta X_1 + X_2, (1-\theta)X_3\} \leq 3)$$

where $\theta \in [0;1]$.

Solution.

We use 3 different algorithms to find an estimate $\hat{\theta}$ of θ^*. In each algorithm an estimate of $z'(\theta)$ is used. Hence two derivative estimators – -Likelihood Ratio and Finite Differences — will be applied.

Robbins-Monro algorithm

The algorithm is

$$\theta_{n+1} = \theta_n - K\tfrac{1}{n}Y_{n+1}$$

where K is close to $1/z''(\theta^*)$ and Y_{n+1} is an estimator of $z'(\theta_n)$. Then it holds that

$$\sqrt{n}(\theta_n - \theta^*) \xrightarrow{\mathcal{D}} \mathcal{N}(0,\sigma^2)$$

where

$$\sigma^2 = \frac{K^2\omega^2}{2Kz''(\theta^*) - 1}$$

with ω^2 the variance of the derivative estimator Y at the optimizer θ^*. Hence the (θ_n)–sequence gives an estimate of θ.

The LR estimator of $z'(\theta)$ and $z''(\theta)$

Let f_θ denote the joint density of $(Z_1, Z_2, Z_3) = (\theta X_1, X_2, (1-\theta)X_3)$. Since

$$Z_1 \sim \text{erlang}(2, \tfrac{2}{\theta}), \qquad Z_2 \sim \text{erlang}(2, 1), \qquad Z_3 \sim \text{erlang}(2, \tfrac{2}{3(1-\theta)})$$

we have

$$f_\theta(z_1, z_2, z_3) = \left(\frac{2}{\theta}\right)^2 z_1 \exp\left(-\tfrac{2}{\theta} z_1\right) z_2 \exp(-z_2) \left(\frac{2}{3(1-\theta)}\right)^2 \exp\left(-\frac{2}{3(1-\theta)} z_3\right)$$

Furthermore

$$z'(\theta_0) = \mathbb{E}_{\theta_0}[ZS]$$

where

$$Z = \mathbb{1}(\max\{Z_1 + Z_2, Z_3\} \leq 3) \quad \text{and} \quad S = \frac{f'_{\theta_0}(Z_1, Z_2, Z_3)}{f_{\theta_0}(Z_1, Z_2, Z_3)}$$

From plain differentiation we have

$$\begin{aligned} f'_\theta(z_1, z_2, z_3) &:= \frac{\partial}{\partial \theta} f_\theta(z_1, z_2, z_3) \\ &= \left(-\frac{2}{\theta} + \frac{2}{\theta^2} z_1 - \frac{2}{1-\theta} + \frac{2}{3(1-\theta)^2} z_3\right) f_\theta(z_1, z_2, z_3). \end{aligned}$$

Simulating m iid copies $(Z_1, S_1), \ldots, (Z_m, S_m)$ under $\mathbb{P}_{\theta_0}$ yields an estimate Y of $z'(\theta_0)$:

$$Y = \frac{1}{m} \sum_{i=1}^{m} Z_i S_i \,.$$

Similarly an estimator

$$\hat{z}''(\theta_0) = \frac{1}{m} \sum_{i=1}^{m} Z_i T_i$$

of $z''(\theta_0)$ is obtained where $(Z_1, T_1), \ldots, (Z_m, T_m)$ are iid copies of

$$Z = \mathbb{1}(\max\{Z_1 + Z_2, Z_3\} \leq 3) \quad \text{and} \quad T = \frac{f''_{\theta_0}(Z_1, Z_2, Z_3)}{f_{\theta_0}(Z_1, Z_2, Z_3)}.$$

Here we have

$$\begin{aligned} \frac{f''_\theta(z_1, z_2, z_3)}{f_\theta(z_1, z_2, z_3)} = &\left(-\frac{2}{\theta} + \frac{2}{\theta^2} z_1 - \frac{2}{1-\theta} + \frac{2}{3(1-\theta)^2} z_3\right)^2 \\ &+ \left(\frac{2}{\theta^2} - \frac{4}{\theta^3} z_1 + \frac{1}{(1-\theta)^2} - \frac{4}{3(1-\theta)^3} z_3\right). \end{aligned}$$

Polyak–Ruppert algorithm

This algorithm is a development of the Robbins–Monro algorithm. Like before a sequence (θ_n) is derrived succesively by

$$\theta_{n+1} = \theta_n - K\frac{1}{n^\gamma}Y_{n+1}$$

with some constants K and $0 < \gamma \leq 1$. Again each Y_{n+1} is an independent reproduction of the LR estimator of $z'(\theta_n)$. This algorithm differs from the Robbins–Monro as the final estimate, $\hat{\theta}_n$, of θ^* is given as the average of the (θ_n)–sequence.

$$\hat{\theta}_n = \frac{1}{n}\sum_{i=1}^{n}\theta_i.$$

Kiefer–Wolfowitz algorithm

As before a sequence (θ_n) is defined successively

$$\theta_{n+1} = \theta_n - K\frac{1}{n^\gamma}Y_{n+1}.$$

But now each Y_{n+1} is a central finite difference (FD) estimator of $z'(\theta_n)$.

The FD estimator of $z'(\theta)$

Define for each θ

$$Z(\theta) = \mathbb{1}(\max\{\theta X_1 + X_2, (1-\theta)X_3\} \leq 3).$$

From m independent copies of (X_1, X_2, X_3) the estimator Y_{n+1} is expressed by

$$Y_{n+1} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{c_i}\left(Z(\theta + c_i/2) - Z(\theta - c_i/2)\right)$$

where (c_n) is a suitable sequence of constants with $c_n \to 0$ and for each i the same (X_1, X_2, X_3)–vector is used to produce both $Z(\theta - c_i/2)$ and $Z(\theta + c_i/2)$.

The results

In the Robbins–Monro algorithm we take $\theta_0 = 0.5$ and $K = 1/\hat{z}''(\theta_0)$. The likelihood ratio derivative estimator has $m = 10$ replications in each step.

Moreover in Polyak–Ruppert γ is taken to be 0.8.

In Fig. 6.1 the sequence (θ_n) is sketched for both the RM and the PR algorithms. We see that the Polyak–Rupert algorithm stabilizes faster than the Robbins–Monro. In the Kiefer–Wolfowitz algorithm we choose $\gamma = 0.8$ and $c_n = n^{-0.8/6}$. But here et seems necessary to let $m = 100$ – the number of replications in each step. In Fig. 6.2the (θ_n)–sequence is sketched with both $m = 10$ and $m = 100$ replications

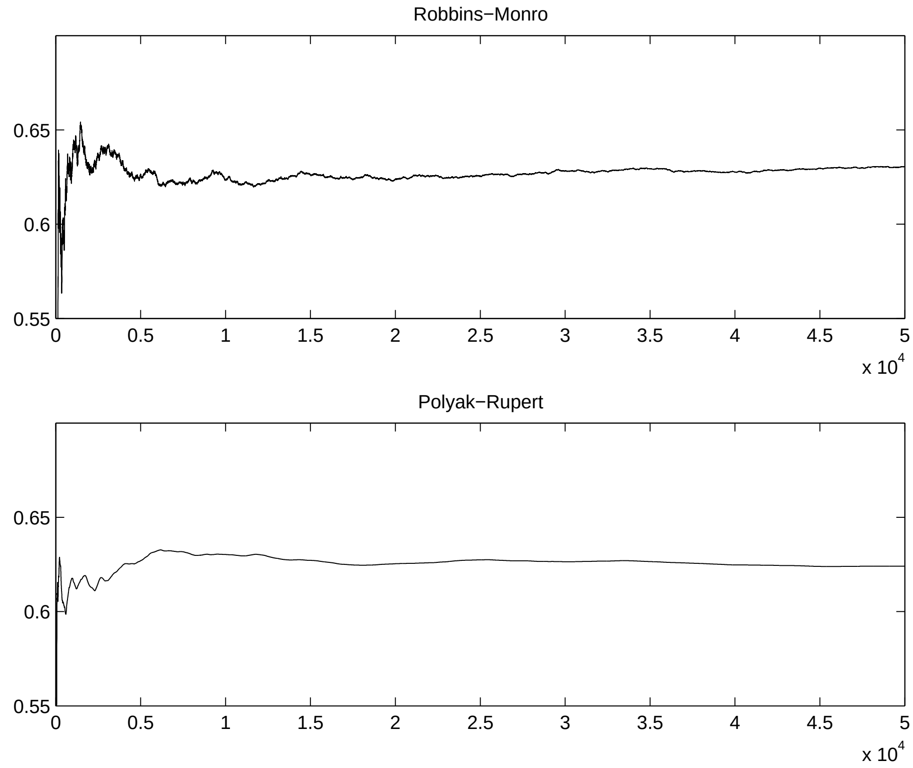


Figure 6.1: Shows the (θ_n)–sequence for the RM and the PR algortihms

Algorithm	Lower	Estimate	Upper
Robbins–Monro	$6.1897 \cdot 10^{-1}$	$6.2434 \cdot 10^{-1}$	$6.2971 \cdot 10^{-1}$
Polyak–Rupert	$6.2434 \cdot 10^{-1}$	$6.2592 \cdot 10^{-1}$	$6.2749 \cdot 10^{-1}$
Kiefer–Wolfowotz	$6.2301 \cdot 10^{-1}$	$6.2579 \cdot 10^{-1}$	$6.2857 \cdot 10^{-1}$

Table 6.1

in each step. With $m = 10$ the sequence hasn't stabilised after 50,000 (and experiments show that 150,000 won't suffice either). Since the number of steps affects the computation time much more than the number of replications in each step $m = 100$ seems reasonable.

Table 6.1 shows the estimates for the optimal θ together with confidence intervals. For the the RM algorithm the interval is constructed using the asymptotic normal distribution. In the PR and the KW case both the estimate and the interval is constructed using sectioning and student t confidence intervals. That is running the algorithm $N = 5$ times and then utilise that each of the 5 estimates asymptotically follow a normal distribution. Since the PR and the KW intervals takes 5 runs of the algorithm and the RM only uses a single run they should not be compared directly. Similarly the bigger computational effort caused by the larger amount of replications in each step should be taken into account in the comparison of the KW

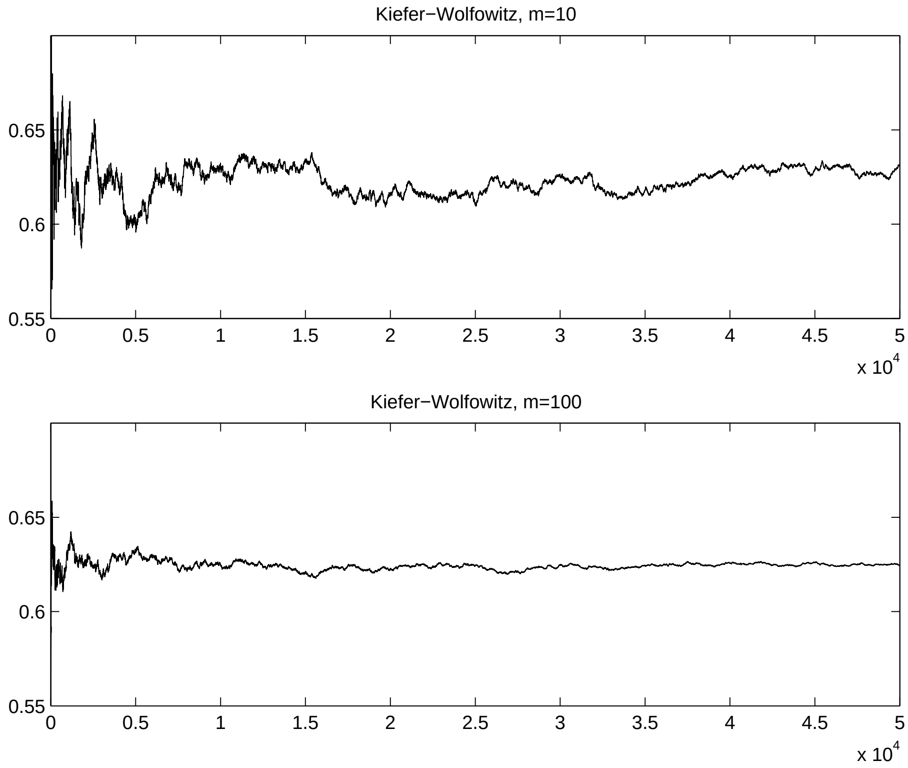


Figure 6.2: Shows the (θ_n)–sequence for KW algorithm both in the $m = 10$ case and the $m = 100$ case.

interval with the PR interval.

7 Miscellaneous

7.1 Introductory exercises from AG Ch. 1

Exercise AG.I.6.1.

In a soccer tournament with 16 teams, the first 8 games are 1-2 (i.e., team 1 versus team 2), 3-4, ..., 15-16, the next four (the quarterfinals) the winner of 1-2 versus the winner of 3-4 and so on until the final. Ties are decided by sudden death (the winner is the first to score). Team i is characterized by three parameters λ_i, p_i, γ_i, such that normally the number of major possibilities for a goal it creates is Poisson(λ_i) and the probability of preventing the opponent from scoring on one of their major possibilities is p_i. However, a bad day may occur w.p. γ_i and then λ_i, p_i are reduced to $2/3$ of their values. The parameters are as follows:

	1	2	3	4	5	6	7	8
λ	4.2	6.2	6.4	4.9	6.2	3.2	6.6	6.2
p	0.65	0.80	0.82	0.66	0.78	0.82	0.47	0.53
γ	0.36	0.23	0.23	0.32	0.42	0.19	0.37	0.41
	9	10	11	12	13	14	15	16
λ	4.2	4.1	8.7	3.3	6.8	0.7	4.1	4.9
p	0.65	0.60	0.88	0.55	0.72	0.50	0.74	0.69
γ	0.36	0.30	0.23	0.19	0.30	0.38	0.32	0.29

Explain that in a match between teams i and j, the score is N_{ij} goals to N_{ji}, where N_{ij} is Poisson(μ_{ij}) given two independent r.v.'s Y_i, Y_j that are $2/3$ w.p. γ_i, γ_j and 1 otherwise, and $\mu_{ij} \overset{\text{def}}{=} \lambda_i Y_i (1 - p_j Y_j)$, and similarly for N_{ji}. Show also that the conditional probability that team i wins a possible draw is $\mu_{ij} / \big(\mu_{ij} + \mu_{ji}\big)$. Give next a table over estimated values of the probabilities of the different teams to win the tournament.

Solution.

Table 7.1 shows the estimated probabilities[1] of the different teams to win the tournament using $R = 1 \cdot 10^6$ replications.

[1] In general, it is bad style to give as many digits as here. An excuse is that the probabilities vary from approx. $1/2$ to $1\text{e} - 05$.

i	$\widehat{z}_i$	i	$\widehat{z}_i$
1	0.001638	9	0.005361
2	0.078196	10	0.005712
3	0.102803	11	0.493091
4	0.010395	12	0.003386
5	0.039142	13	0.119012
6	0.035517	14	0.000011
7	0.010613	15	0.029786
8	0.015461	16	0.049876

Table 7.1: $\widehat{z}_i = \mathbb{P}(\text{Team } i \text{ wins the tournament})$.

Confidence intervals were not asked for, but are trivial to give since the method is just binomial sampling.

An interesting twist of the exercise is to study the importance of the drawing of matches, i.e. to order the teams at random in each of the R replications. Not implemented here.

Exercise AG.I.6.3.

A company holding a European call option with maturity $T = 1$ year and strike price K delta hedges every week. Assuming the log-returns to follow the GBM Black–Scholes model with (yearly) volatility $\sigma = 0.25$, $S(0) = K = 1$, and $r = 4\,\%$ per year, investigate how good the delta hedge replicates the payout $[S(T) - K]^+$, say by plotting 25 values of the pairs $([S(T) - K]^+, w(T))$, where $w(t)$ is the value of the hedging portfolio at time t.

Explanation. Let $\Pi(x, t, T)$ be the price at time t of the option given $S(t) = x$ and $\Delta(t) = (\partial/\partial x)\Pi(x, t, T)\big|_{x=S(t)}$ (which can be computed by straightforward differentiation in the Black-Scholes formula). The portfolio that is delta hedging the option at times $0 = t_0 < t_1 < \cdots < t_n < T$ invests in $a_1(t_i) = \Delta(t_i)$ units of the underlying asset, whereas the amount $a_2(t_i)$ put in the bank is chosen so as to make the portfolio self-financing, i.e., one should have

$$a_1(t_{i-1})S(t_i) + a_2(t_{i-1})\mathrm{e}^{r(t_i - t_{i-1})} = a_1(t_i)S(t_i) + a_2(t_i)\,.$$

The initial weights $a_1(0), a_2(0)$ are chosen such that $w(0-) = \Pi\big(S(0), 0, T\big)$, and the terminal value of the hedge becomes $w(T) = a_1(t_n)S(T) + a_2(t_n)\mathrm{e}^{T - t_n}$. See further Björk's book.

Solution (I).

The desired plot is in Fig. 7.1. The line is the diagonal $x = y$, where the continuous-time hedge would lie. The conclusion is that the discrete hedge time works very good (maybe surprisingly good!).

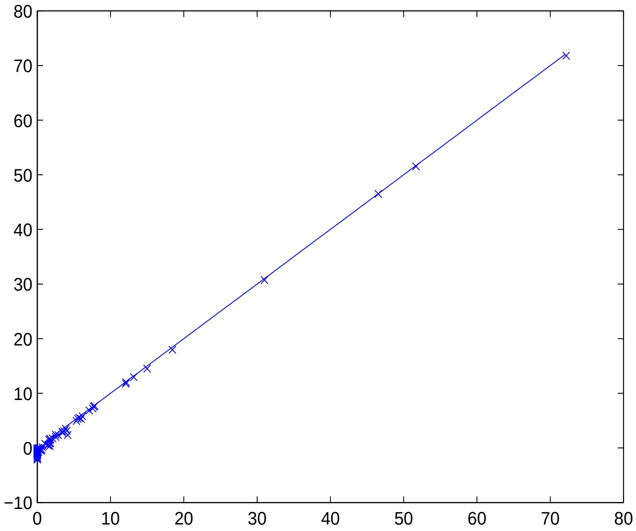


Figure 7.1

Solution (Anders Alexander Helweg-Mikkelsen).

The condition $a_1(t_i) = \Delta(t_i)$ determines the entire sequence $\{a_1(t)\}_{t=0}^{T}$. Also, the fact that the initial value of the hedging portfolio should equal that of the call option provides an initial condtion for $a_2(0)$ while the self-financing condition gives an equation relating $a_2(t_i)$ to $a_2(t_{i-1})$ such that the sequence $\{a_2(t)\}_{t=0}^{T}$ can be recovered as well. Using these sequences to construct the hedging portfolio gives rise to the results shown in Fig. 7.2. While the hedging portfolio appears to replicate the option well, Fig. 7.2 reveals that the hedging portfolio can attain negative values whereas the option value remains positive. Fig. 7.3 performs the same comparison in terms of a scatter plot of the value of the hedging portolio against the option value where we obviously expect a to see a line close to the diagonal $x = y$.

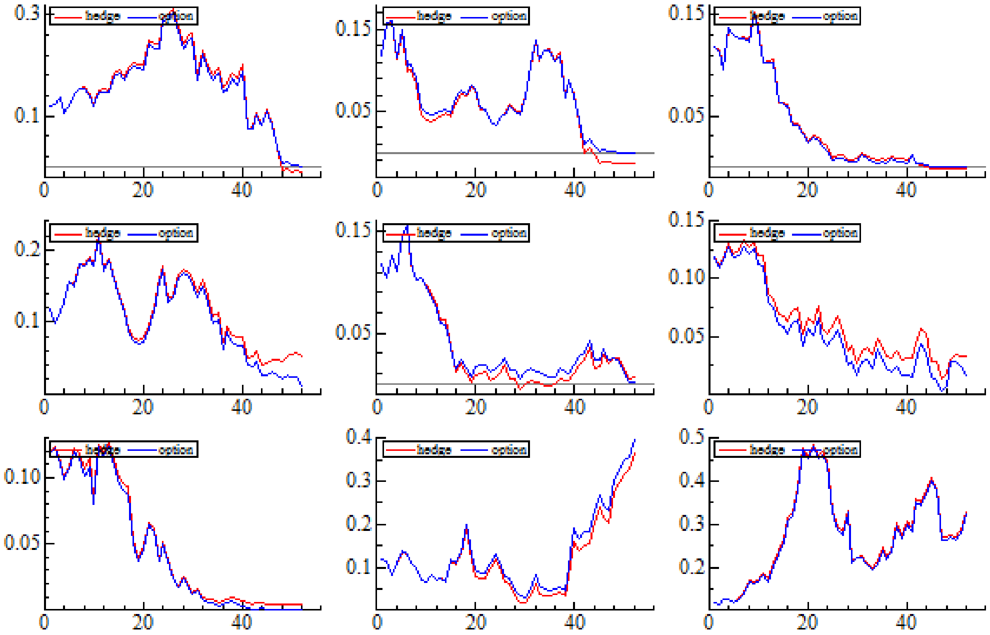


Figure 7.2

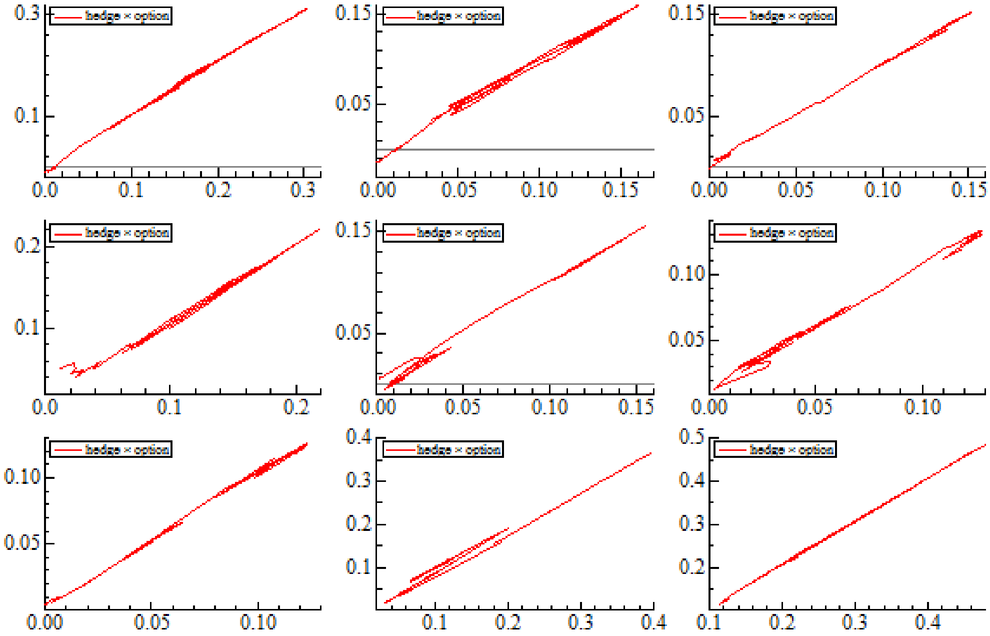


Figure 7.3

7.2 VaR and copulas

Exercise AG II.3.8 .

A portfolio has 10 assets of current values

$$\begin{pmatrix} x_1 & x_2 & \dots & x_{10} \end{pmatrix} = \begin{pmatrix} 5 & 2 & 10 & 8 & 8 & 3 & 15 & 4 & 1 & 2 \end{pmatrix}.$$

The value of the holdings of asset i next week is $x_i \mathrm{e}^{Y_i}$, where $Y_1, \dots, Y_{10}$ are (possibly dependent) r.v.'s. Thus, the loss is

$$L \stackrel{\text{def}}{=} \sum_{i=1}^{10} x_i (1 - \mathrm{e}^{Y_i}),$$

and the VaR (value at risk) is the 99% quantile in the distribution of L.

There is good statistical evidence that the marginal distributions of the Y_i are normal with mean zero (as is often the case for a short time horizon) and volatilities

$$4 \ 7 \ 1 \ 5 \ 3 \ 5 \ 2 \ 6 \ 4 \ 4$$

(in %), respectively (thus, for example, $Y_2 \sim \mathcal{N}(0, 0049)$). However, the dependence structure is less well understood. Your assignment is to estimate the VaR for the given normal marginals and the following six dependence structures: the three $N_{10}(0, \mathbf{\Sigma})$ distributions where $\mathbf{\Sigma}$ corresponds to symmetric correlation 0, 0.3, 0.6, and the three Student $t_{10}(\mathbf{\Sigma}, f)$ copulas with the same $\mathbf{\Sigma}$'s and $f = 1$.

Note that the c.d.f. of the t-distribution with $f = 1$ is $1/2 + \arctan(x)/\pi$. The α-quantile of an r.v. L is estimated by simulating R replications $L_1, \dots, L_R$, forming the order statistics $L_{(1)} < \dots < L_{R)}$, and using the estimate $L_{(\alpha(R+1))}$ (thus, it is convenient to choose R such that $\alpha(R+1)$ is an integer).

Solution.

Ten repetitions with $R = 9{,}000$ replications in each produced the following table, with the three Gaussian values followed by the three t-values in each column:

1.49	2.62	3.40	1.70	2.78	3.60
1.52	2.64	3.44	1.99	3.21	3.94
1.51	2.64	3.32	1.69	2.89	3.48
1.53	2.56	3.46	1.75	2.94	3.53
1.52	2.61	3.42	1.72	2.89	3.55
1.51	2.62	3.43	1.81	2.95	3.59
1.56	2.70	3.50	1.83	2.91	3.59
1.48	2.60	3.39	1.89	3.10	3.54
1.53	2.65	3.30	1.72	2.84	3.62
1.52	2.75	3.39	1.74	2.80	3.60

One observes, as expected, the values to be increasing in the correlation and somewhat larger for the Gauss copulas than for the t-copulas. Comparing the values for different repetitions show quite some variation, confirming the folklore that quantile estimation is difficult. Note (AG.III.4a) that there is a normal confidence interval procedure, but it requires density estimation, a topic difficult in itself. In the class, we have therefore been using bootstrap (AG.III.5c) to get confidence intervals, see the next exercise.

Exercise AG.III.5.1.

Complement your solution of Exercise AG.II.3.8 by giving an upper 95% bootstrap confidence limit for the VaR. For simplicity, you may consider the multivariate Gaussian copula only.

Solution.

Tables 7.2–7.3 provide an upper 95 % Bootstrap confidence limit for the VaR corresponding to the Gaussian copula and the t-copula respectively. We have used $R = 10{,}000$ and $b = 1000$.

ρ	Upper Limit
0	1.550479
0.3	2.697105
0.6	3.478171

Table 7.2: Gaussian copula

ρ	Upper Limit
0	3.498314
0.3	4.954051
0.6	5.527842

Table 7.3: t-copula

7.3 Steady-state simulation

Exercise AG.IV.8.1.

Perform perfect sampling of the Moran dam (Example AG.IV.8.5) for the case in which V is geometric with mean 2 and $m = 3$, and p is variable. Use both independent updating and monotone updating, and compare the two methods in terms of the size of values of p for which you are able to produce Z within reasonable time.

Solution.

Table 7.4 below shows the estimated stationary probabilities using $p = 30$ and $R = 1 \cdot 10^5$.

i	$\widehat{p}_i$	i	$\widehat{p}_i$	i	$\widehat{p}_i$
0	0.56912				
1	0.08150	11	0.00956	21	0.00107
2	0.06514	12	0.00858	22	0.00097
3	0.05456	13	0.00635	23	0.00087
4	0.04365	14	0.00521	24	0.00068
5	0.03565	15	0.00433	25	0.00048
6	0.02814	16	0.00368	26	0.00046
7	0.02385	17	0.00311	27	0.00048
8	0.01828	18	0.00226	28	0.00027
9	0.01522	19	0.00178	29	0.00018
10	0.01252	20	0.00151	30	0.00036

Table 7.4: Stationary Probabilities

The comparison between independent updating and monotone updating is given in Fig. 7.4 below.

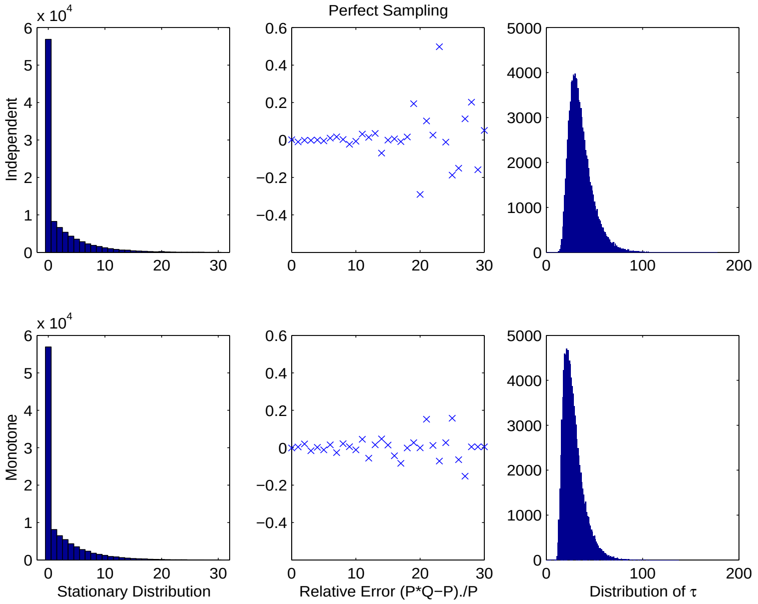


Figure 7.4

The graphs in the first row correspond to the results with independent updating while the second to monotone updating. The approximated stationary distributions are displayed in the first column. The second column shows a measure of the error the approximations. Finally, the third column corresponds to histogram that depict the distribution of the *backward coupling time* τ.

From the figure above it is seen that the monotone updating method provides more accurate results than independent updating. Moreover, the monotone updating method is more efficient from a practical point of view since it requires a smaller amount of simulations and it is more likely to finish with less updates than the independent updating (this is easily seen from the distribution of τ). In fact, in our examples, monotone updating was around 2.5 times faster than independent updating.

Exercise AG.IV.6.3.

Consider an (s, S) inventory system in which the number of items stored at time t is $V(t)$. Demands arrive (one at a time) according to a Poisson process with intensity λ. When $V(t-) = s + 1$, $V(t) = s$, an order of size $S - s$ is placed and arrives after a random time Z (the lead time). Demands arriving while $V(t) = 0$ are lost. It is assumed that $S - s > s$.

Write a program for regenerative simulation of p, the long-run probability that a demand is lost. Use whatever values of s, S, λ you like and whatever distribution of Z you prefer.

Solution.

In the results for Table 7.5 we have chosen $s = 15$, $S = 100$ and Gamma$(2, 5)$ as the distribution of Z where we have used $R = 100,000$ replications.

$\widehat{p}$	Lower Limit	Upper Limit	$\mathbb{V}\text{ar}(\widehat{p})$
$1.677632\,e-02$	1.666819e-02	1.688446e-02	3.043777e-03

Table 7.5: Confidence Interval

7.4 Derivative estimation

If $z = z(\theta)$ is a number depending on a parameter θ, the problem of computing the derivative $z'(\theta)$ of $z(\theta)$ comes up in a number of applications, including the Greeks in finance. Methods are discussed in AG Ch. VII and examples given. At my Aarhus course, we have most often treated the topic but did not do so at the EURANDOM minicourse. Here is an exercise:

Exercise AG.VII.3.1.

For the compound Poisson sum $C = \sum_1^N V_i$ in the insurance risk setting, assume that $\lambda = 1000$ and that the V_i are lognormal with $\mu = 0$, $\sigma^2 = 1$. Let $z(\lambda) = f\,\mathbb{P}_\lambda(C > 5\,\mathbb{E}\,C)$. Give an estimate of $z'(\lambda)$ and an associated confidence interval by means of the LR method.

Solution.

$5\,\mathbb{E}\,C$ war replaced by $x = 1.1\,\mathbb{E}\,C$. Then $z(\lambda) = \mathbb{P}_\lambda(C > x) \approx 0.3$. The LR estimator of $z'(\lambda)$ is calculated just as in Example VII.3.3 and with $R = 100,000$ replications, the confidence interval came out as 0.0013±5.6e-5.

Exercise AG.VII.2.4.

Consider the delta $z'(\theta)$ of a weekly sampled Asian option with price $z(\theta)$ where θ is the initial price of the underlying asset. Assume the GBM Black–Scholes model, maturity $T = 1$ years, strike price $K = 120$, $\theta_0 = 100$, $r = 5\%$ per year, and $\sigma = 0.25$.

1. Compute $z(\theta)$, $z'(\theta)$, and associated confidence intervals.
2. Improve the efficiency of your method by antithetic sampling of the Brownian increments.
3. If time permits, do some experiments on variance reduction by changing the drift of the Brownian motion. At least outline what the changed estimates are!

Solution.

(i) With R = 100,000 replications, the confidence interval for $z(\theta)$ came out as 1.19±0.05, and for $z'(\theta)$ as 0.154±0.005

(ii) The improvement by using antithetic sampling is minor: Remember from AG p. 144 that the variance is reduced by a factor ρ compared to CMC. We obtained $\rho = -0.075$ in the case of $z(\theta)$ and $\rho = -0.148$ in the case of $z'(\theta)$.

(iii): not implemented here.

7.5 The stochastic counterpart method

Assume we want to estimate the minimum or maximum of a smooth function $w(\theta)$ of $\theta \in \Theta \subseteq \mathbb{R}$ or, equivalently (under some regularity conditions), the solution of $w'(\theta)$. In the stochastic counterpart method, one then provides a smooth simulation estimate $\widehat{w}(\theta)$ of $w(\theta)$ and finds the zero θ^* by deterministic algorithms such as Newton-Raphson. How to give confidence intervals is discussed in AG.III.4.

Exercise (new).

Consider the PERT net in AG Example VIII.5.1 and let

$$M(\theta) = \max(\theta X_1 + X_2, (1-\theta)X_3)$$

where

$$X_1 \sim \text{Erlang}(2, 1/2) \qquad X_2 \sim \text{Erlang}(2, 1) \qquad X_3 \sim \text{Erlang}(2, 3/2)$$

and $g(\theta) = \mathbf{1}\{M(\theta) > d\}$ (with $d = 3$). Your task is to find the value $\theta = \theta^*$ that minimizes

$$w(\theta) := \mathbb{E}[g(\theta)] = \mathbb{P}(M(\theta) > 3).$$

Solution.

To start with, we use Crude Monte Carlo to generate the graph in Figure 7.5 which depicts $w(\theta)$ as a function of θ in the interval $[-1, 2]$. It is clear that $w(\theta)$ has a global minima and we can safely restrict our search to the open interval $(0, 1)$. To do so, observe that with $\theta \in (0, 1)$

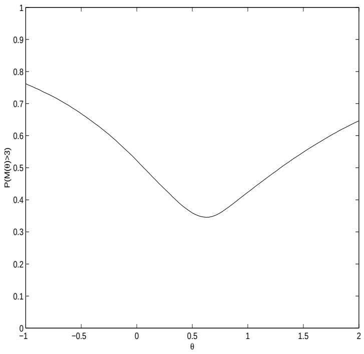


Figure 7.5: $w(\theta)$.

$$\theta X_1 \sim \text{Erlang}(2, \theta/2) \qquad \text{and} \qquad (1-\theta)X_3 \sim \text{Erlang}(2, 3(1-\theta)/2)$$

and hence we can write $w(\theta)$ as

$$\begin{aligned}&\int_{\{M(\theta)>3\}} f(x_1; 2, 1/2) f(x_2; 2, 1) f(x_3; 2, 3/2)\, dx_1\, dx_2\, dx_3 \\ &\quad = \int_{\{\max(x_1+x_2, x_3)>3\}} f(x_1; 2, \theta/2) f(x_2; 2, 1) f(x_3; 2, 3(1-\theta)/2)\, dx_1\, dx_2\, dx_3\end{aligned}$$

where $f(\cdot\,; k, \lambda)$ is the density of an Erlang distribution with parameters k and λ. So, using importance sampling with importance distribution corresponding to $\widetilde{\theta} = 1/2$ we can rewrite $w(\theta)$ as

$$\begin{aligned}&\mathbb{E}_\theta[\mathbb{1}(\max(X_1 + X_2, X_3) > 3)] \\ &\quad = \mathbb{E}_{1/2}\left[\frac{f(X_1; 2, \theta/2) f(X_3; 2, 3(1-\theta)/2)}{f(X_1; 2, 1/4) f(X_3; 2, 3/4)} \mathbb{1}(\max(X_1 + X_2, X_3) > 3)\right].\end{aligned}$$

Define $W(\theta, X_1, X_2, X_3)$ as

$$\frac{f(X_1; 2, \theta/2) f(X_3; 2, 3(1-\theta)/2)}{f(X_1; 2, 1/4) f(X_3; 2, 3/4)} \mathbb{1}(\max(X_1 + X_2, X_3) > 3)$$

which after some simplifications becomes

$$W(\cdot) = \frac{1}{16}\frac{\exp\left\{\frac{2}{3}\frac{(2\theta-1)(3(\theta-1)X_1+\theta X_3)}{\theta(\theta-1)}\right\}}{\theta^2\left(\theta-1\right)^2}\mathbb{1}(\max(X_1+X_2,X_3)>3)$$

It can be proved that $W(\theta)$ satisfy the necessary conditions required by the stochastic counterpart method (valid derivative interchange). Hence, the Monte Carlo estimator $\widehat{\theta}^*$ of θ^* is given by the root of

$$\widehat{z}(\theta) := \frac{1}{R}\sum_{i=1}^{R} Z_i(\theta) \qquad \text{where } Z_i(\theta) = \frac{\partial}{\partial\theta}W(\theta, X_{1i}, X_{2i}, X_{3i})$$

Using $R = 10^7$ replications gave the 95% confidence interval 0.6258 ± 0.0014. The following are expressions for the partial derivatives of W

$$\begin{aligned}\frac{\partial}{\partial\theta}W(\cdot) &= 2\,W(\cdot)\left(\frac{X_1}{\theta^2} - \frac{X_3}{3(1-\theta)^2} - \frac{1-2\theta}{\theta(1-\theta)}\right)\\ \frac{\partial^2}{\partial\theta^2}W(\cdot) &= 4\,W(\cdot)\left[\left(\frac{X_1}{\theta^2} - \frac{X_3}{3(1-\theta)^2} - \frac{1-2\theta}{\theta(1-\theta)}\right)^2\right.\\ &\qquad\qquad\left. - \frac{X_1}{\theta^3} - \frac{X_3}{3(1-\theta)^3} + \frac{\theta^2-\theta+\frac{1}{2}}{\theta^2(1-\theta^2)}\right].\end{aligned}$$

Exercise (new).

Consider the coalescent in Example AG.XIII.4.4 with $K = k = 31$ and $n = 42$. Give a simulation estimate θ^* of the maximum likelihood estimator $\widehat{\theta}$ using the stochastic counterpart method.

Solution.

Point estimate 7.56 with (estimated) variance 153 per replication.

7.6 Markov Chain Monte Carlo

Exercise (new).

Consider the coalescent in Example AG.XIII.4.4 with $K = k = 31$ and $n = 42$ and assume that the the value $\theta^* = 7.56$ found in the exercise at the end of Section 7.5 is the correct one. Give a histogram of the posterior of W_{42}

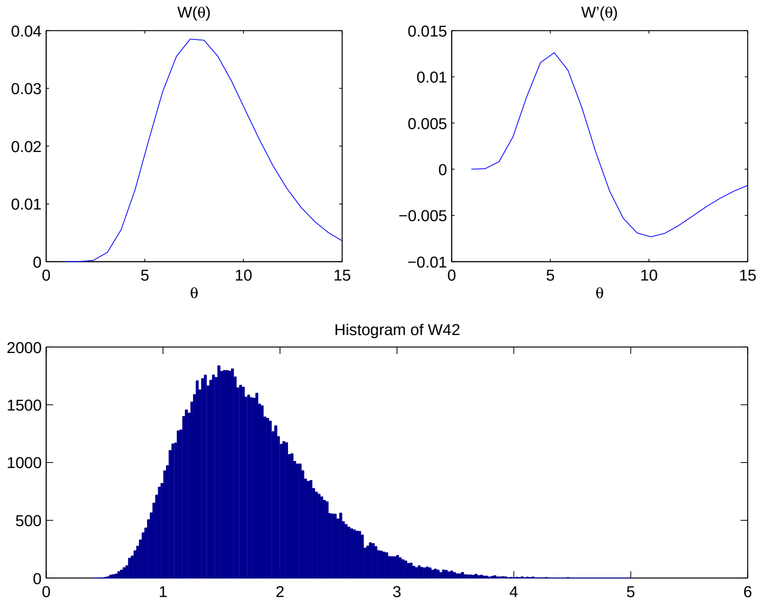


Figure 7.6

Solution.

We assume that the value θ^* given in Exercise III.4.3 is the correct one. The figure 7.6 shows an histogram that depicts the posterior of W_{42} given $K = 31$.

Moreover, the Table 7.6 shows an equitailed 95 % confidence interval based on the posterior.

2.5 % Percentile	97.5 % Percentile
0.863947	3.017179

Table 7.6

Finally, the value 1.924317 is the Markov chain value giving the highest likelihood.

Exercise AG XIII.5.2.

A classsical data set contains the numbers of cases of mastitis (an inflammatory disease) in 127 herds of dairy cattle. With n_i the number of herds having i cases, the data (adjusted for herd size) are as follows:

i	0	1	2	3	4	5	6	7	8	9	10	11	12
n_i	7	12	8	9	7	8	9	6	5	3	4	7	4
i	13	14	15	16	17	18	19	20	21	22	23	24	25
n_i	5	2	1	4	3	3	4	2	2	4	1	0	5

Assume that for a given herd j, the number of cases is Poisson(λ_j), where the λ_j are Gamma(α, β_j) and the β_j are themselves Gamma(a, b). Use Gibbs sampling with $\alpha = 0.1$, $a = b = 1$ to give histograms of the posterior distribution of λ_j for a herd with 0, 10, or 20 cases and to report the posterior means of α, a, b.

Solution.

The figure 7.7 shows the histograms of the posterior distribution of λ_i for cases of herds with 0, 10 and 20 cases using the Gibbs sampler.

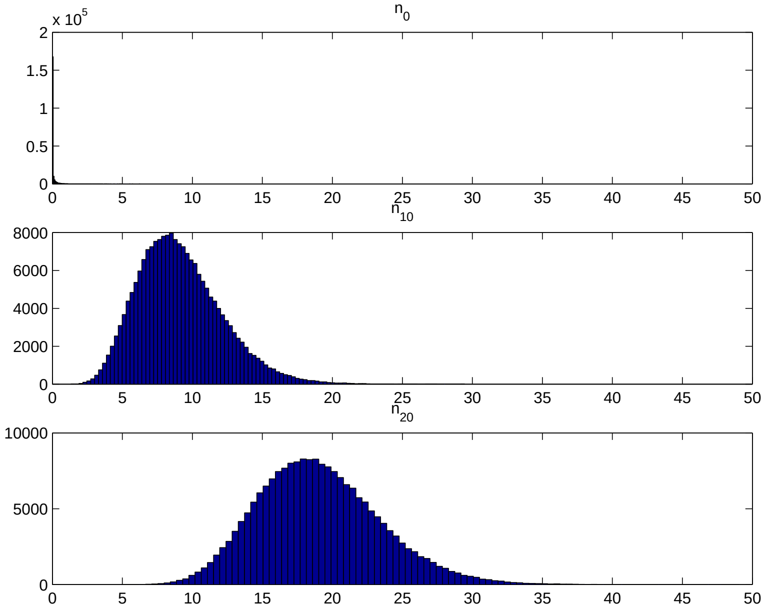


Figure 7.7

7.7 An example on neutron cascades

Exercise AG.XIV.10.4.

A rod of fuel in a nuclear reactor is represented as a line segment of length x. Spontaneous emissions of neutrons occur according to a Poisson process on $[0, x] \times [0, \infty)$ with unit rate, with the second coordinate representing time. Each split creates one new neutron, and the old and the new neutron choose independently to move left or right w.p. 1/2 for each direction. Give simulation estimates of the temperature $T = T(X)$ of the rod as a function of x, where $T(x)$ is defined as the expected number of splits within one time unit. Take $\alpha = 1$. Hint: Reduce to the study of characteristics of cascades as in the book.

Solution.

Consider the neutron transport process in AG.XIV.10c. Fig. 7.8 shows simulation estimates of the average number of collision per cascade as a function of the size of the reactor (height) using $\mu = 1$, $a = 5$ and $\lambda = 0$ (no spontaneous emissions).

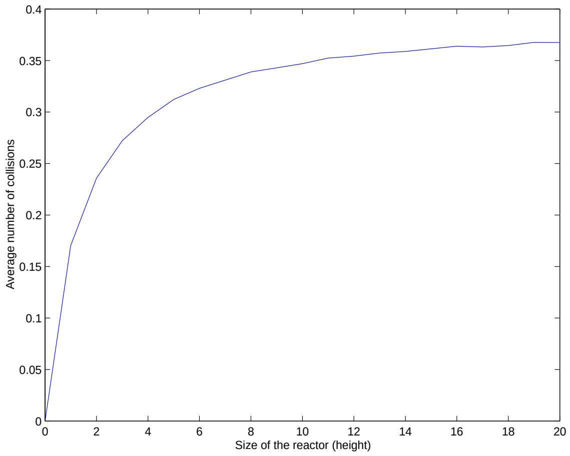

Average number of collisions
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0
2
4
6
8
10
12
14
16
18
20
Size of the reactor (height)

Figure 7.8

Appendix

A1 Matlab issues

AG is written without focus on any specific software. For my teaching, I have invariably been using Matlab and this is also chosen for the present document. The choice of Matlab was motivated by the language being easy to learn, the many built-in facilities and its widespread use on a global basis (but note that **R** is popular among statisticians). For Monte Carlo use, the Statistics Toolbox is indispensable, and for special applications other toolboxes are useful, say the Finance Toolbox.

To get started with Matlab, just search Google for “Matlab, tutorial”. A vast number of choices will come up. Later, the `Help` facility will prove most useful.

Matlab is matrix-based so that many operations need not be done element by element but can be done for matrices. For a simple example, if `x` is a $n \times n$ matrix, the block `for i=1:n, for j=1:m, x(i,j)=rand; end; end;` and the command `x=rand(n,m);` both fill `x` with pseudo-random numbers. Similarly,

```
for i=1:n, for j=1:m
    if y(i,j)<z(i,j), x(i,j)=1; else x(i,j)=0; end;
end; end;
```

and `x=(y<z);` are equivalent. Matrix commands as in the second implementations are much faster and can condens programs quite a lot. The cost is that readability as in the first may be lost. For this reason, I myself am not using the matrix facilities in full and to do also requires more of programming skill. Two further examples follow.

Example 8.

When constructing a covariance matrix $\boldsymbol{V}$ from a correlation matrix $\boldsymbol{P}$ and a vector $\boldsymbol{v}$ of volatilities, the routines

```
for i=1:length(v), for j=1:length(v)
    V(i,j)=P(i,j){*}v(i){*}v(j);
end; end;
```

and `V=diag(v){*}P{*}diag(v);` are equivalent.

Example 9.

Generating R paths, each with T time steps, from a Brownian motion with drift a, volatility b, and starting value s can be done by combining the initialization

```
    mS=zeros(T+1,R); mS(1,:)=s{*}ones(1,R); mEps=rannorm(0,1,T,R);
```

with either

```
for i=1:R, for j=1:T
    mS(j+1,i)=mS(j,i)+a{*}h+b{*}sqrt(h){*}mEps(j,i);
end; end;
```

or

```
mS(2:Steps+1,:)=a{*}h+b{*}sqrt(h){*}mEps;
mS = cumsum(mS);
```

Complicacted subroutines are most conveniently incorporated as separate `m`-files. In simpler cases, one can use the `@` structure. For example, $\int_1^2 \mathrm{e}^x \,\mathrm{d}x$ is computed via `f=@(x) exp(x); quad(f,1,2)` (note that `quad(exp,1,2)` won't work!).

Speed is seldom a concern in the examples considered here. However, one should note that programming structures such as **`for␣to`**, **`while`** etc. are slow in Matlab. The execution time of a block may be obtained by writing **`tic`** at the start and **`toc`** at the end. Comparisons of execution times of different algorithms are highly implementation dependent.

For plotting, note the **`subplot`** structure that enables a number of different figures to be nicely collected in one) for an example, see Fig. 7.4). Graphics can be exported in `.pdf` format as needed for Mac LaTeX or in `.eps` format in most other LaTeX implementations.

A2 A variant of the Black-Scholes formula

Lemma A.1. *If $Z \sim \mathcal{N}(\mu, \sigma^2)$, then*

$$\mathbb{E}[x\mathrm{e}^Z - K]^+ = x\mathrm{e}^{\mu+\sigma^2/2}\overline{\Phi}(y_0 - \sigma) \;-\; K\overline{\Phi}(y_0)$$

where $y_0 = (\log K - \log x - \mu)/\sigma$, $\overline{\Phi}(z) = 1 - \Phi(z) = \displaystyle\int_z^\infty \frac{1}{\sqrt{2\pi}}\mathrm{e}^{-y^2/2}\,\mathrm{d}y.$

Proof.

$$\begin{aligned}
\mathbb{E}[x\mathrm{e}^Z - K]^+ &= \int_{-\infty}^{\infty} [x\mathrm{e}^{\mu+\sigma y} - K]^+ \frac{1}{\sqrt{2\pi}}\mathrm{e}^{-y^2/2}\,\mathrm{d}y \\
&= \int_{y_0}^{\infty} [x\mathrm{e}^{\mu+\sigma y} - K] \frac{1}{\sqrt{2\pi}}\mathrm{e}^{-y^2/2}\,\mathrm{d}y \\
&= x\mathrm{e}^{\mu}\int_{y_0}^{\infty} \mathrm{e}^{\sigma y}\frac{1}{\sqrt{2\pi}}\mathrm{e}^{-y^2/2}\,\mathrm{d}y - K\overline{\Phi}(y_0) \\
&= x\mathrm{e}^{\mu+\sigma^2/2}\int_{y_0}^{\infty} \frac{1}{\sqrt{2\pi}}\mathrm{e}^{-(y-\sigma)^2/2}\,\mathrm{d}y - K\overline{\Phi}(y_0) \\
&= x\mathrm{e}^{\mu+\sigma^2/2}\overline{\Phi}(y_0 - \sigma) - K\overline{\Phi}(y_0)
\end{aligned}$$

□

The Black-Scholes formula is an expression for $\mathrm{e}^{-rT}\,\mathbb{E}[x\mathrm{e}^Z - K]^+$ where $Z \sim \mathcal{N}((r - \sigma^2)T, \sigma^2 T)$. Of course, this comes out as a special case of the lemma.