## CHAPTER 12

# Monte Carlo Methods

Deus ex machina.

Wall Street is often compared to a casino. The analogy is appropriate in one respect: Securities firms commonly use simulation techniques, known as *Monte Carlo methods*, to value complex derivatives and to measure risk. Simulation methods approximate the behavior of financial prices by using computer simulations to generate random price paths.

These methods are used to simulate a variety of different scenarios for the portfolio value on the target date. These scenarios can be generated in a random fashion (as in Monte Carlo simulation) or from historical data (as in historical simulation) or in other, more systematic ways. The portfolio value at risk (VAR) then can be read off directly from the distribution of simulated portfolio values.

Because of its flexibility, the simulation method is by far the most powerful approach to VAR. It potentially can account for a wide range of risks, including price risk, volatility risk, and complex interactions such as described by copulas in Chapter 8. Simulations can account for nonlinear exposures and complex pricing patterns. In principle, simulations can be extended to longer horizons, which is important for credit risk measurement and to more complex models of expected returns. Also, it can be used for operational risk measurement, as well as integrated risk management.

This approach, however, involves costly investments in intellectual and systems development. It also requires substantially more computing power than simpler methods. VAR measures using Monte Carlo methods often require hours to run. Time requirements, however, are being whittled down by advances in computers and faster valuation methods.

This chapter shows how simulation methods can be used to uncover VAR. The first section presents the rationale for Monte Carlo simulations. Section 12.2 introduces a simple case with just one random variable. Section 12.3 then discusses the tradeoff between speed and accuracy. The case with many sources of risk is discussed in Section 12.4. Next, Sections 12.5 and 12.6 turn to newer methods, such as deterministic simulations. The choice of models is reviewed in Section 12.7.

## 12.1 WHY MONTE CARLO SIMULATIONS?

The basic concept behind the Monte Carlo approach is to simulate repeatedly a random process for the financial variable of interest covering a wide range of possible situations. These variables are drawn from prespecified probability distributions that are assumed to be known, including the analytical function and its parameters. Thus simulations recreate the entire distribution of portfolio values, from which VAR can be derived.

*Monte Carlo simulations* were developed initially as a technique of statistical sampling to find solutions to integration problems, as shown in Box 12-1. For instance, take the problem of numerical integration of a

**BOX 12-1**

### MONTE CARLO SIMULATIONS

Numerical simulations were first used by atom bomb scientists at Los Alamos in 1942 to crack problems that could not be solved by conventional means. Stanislaw Ulam, a Polish mathematician, is usually credited with inventing the Monte Carlo method while working at the Los Alamos laboratory.

While there, Ulam suggested that numerical simulations could be used to evaluate complicated mathematical integrals that arise in the theory of nuclear chain reactions. This suggestion led to the more formal development of Monte Carlo methods by John Von Neumann, Nicholas Metropolis, and others.

In his autobiography, *Adventures of a Mathematician*, Ulam recollects that the method was named in honor of his uncle, who was a gambler. The name *Monte Carlo* was derived from the name of a famous casino established in 1862 in the south of France (actually, in Monaco). What better way to evoke random draws, roulette, and games of chance?

function with many variables. A straightforward method is to perform the integration by computing the area under the curve using a number of evenly spaced samples from the function. In general, this works very well for functions of one variable. For functions with many variables, however, this method quickly becomes inefficient. With two variables, a $10 \times 10$ grid requires 100 points. With 100 variables, the grid requires $10^{100}$ points, which is too many to compute. This problem is called the *curse of dimensionality.*

Monte Carlo simulation instead provides an approximate solution to the problem that is much faster. Instead of systematically covering all values in the multidimensional space, it generates $K$ random samples for the vector of variables. By the *central limit theorem,* this method generates estimates whose standard error decreases at the rate of $1/\sqrt{K}$, which does not depend on the size of the sample space. Thus the method does not suffer from the curse of dimensionality.

## 12.2 SIMULATIONS WITH ONE RANDOM VARIABLE

### 12.2.1 Simulating a Price Path

We first concentrate on a simple case with just one random variable. The first, and most crucial, step in the simulation consists of choosing a particular stochastic model for the behavior of prices. A commonly used model is the *geometric brownian motion* (GBM) *model,* which underlies much of options pricing theory. The model assumes that innovations in the asset price are uncorrelated over time and that small movements in prices can be described by

$$dS_t = \mu_t S_t dt + \sigma_t S_t dz \tag{12.1}$$

where $dz$ is a random variable distributed normally with mean zero and variance $dt$. This variable drives the random shocks to the price and does not depend on past information. It is *brownian* in the sense that its variance decreases continuously with the time interval, $V(dz) = dt$. This rules out processes with sudden jumps, for instance. The process is also *geometric* because all parameters are scaled by the current price $S_t$.

The parameters $\mu_t$ and $\sigma_t$ represent the instantaneous drift and volatility at time $t$, which can evolve over time. For simplicity, we will assume in what follows that these parameters are constant over time. But since $\mu_t$

and $\sigma_t$ can be functions of past variables, it would be easy to simulate time variation in the variances as in a GARCH process, for example.

In practice, the process with infinitesimally small increments $dt$ can be approximated by discrete moves of size $\Delta t$. Define $t$ as the present time, $T$ as the target time, and $\tau = T - t$ as the (VAR) horizon. To generate a series of random variables $S_{t+i}$ over the interval $\tau$, we first chop up $\tau$ into $n$ increments, with $\Delta t = \tau/n$.[1]

Integrating $dS/S$ over a finite interval, we have approximately

$$\Delta S_t = S_{t-1}\,(\mu\Delta t + \sigma\epsilon\sqrt{\Delta t}) \qquad (12.2)$$

where $\epsilon$ is now a standard normal random variable, that is, with mean zero and unit variance. We can verify that this process generates a mean $E(\Delta S/S) = \mu\Delta t$, which grows with time, as does the variance $V(\Delta S/S) = \sigma^2\Delta t$.

To simulate the price path for $S$, we start from $S_t$ and generate a sequence of epsilons ($\epsilon$'s) *for* $i = 1, 2, \ldots, n$. Then $S_{t+1}$ is set at $S_{t+1} = S_t + S_t(\mu\Delta t + \sigma\epsilon_1\sqrt{\Delta t})$, $S_{t+2}$ is similarly computed from $S_{t+1} + S_{t+1}(\mu\Delta t + \sigma\,\epsilon_2\sqrt{\Delta t})$, and so on for future values, until the target horizon is reached, at which point the price is $S_{t+n} = S_T$.

Table 12-1 illustrates a simulation of a process with a drift $\mu$ of zero and volatility $\sigma$ of 10 percent over the total interval. The initial price is \$100, and the interval is cut into 100 steps. Therefore, the local volatility is $0.10 \times \sqrt{1/100} = 0.01$.

**TABLE 12-1**

Simulating a Price Path

| Step $i$ | Previous Price $S_{t+i-1}$ | Random Variable $\epsilon_i$ | Increment $\Delta S$ | Current Price $S_{t+i}$ |
|---|---|---|---|---|
| 1 | 100.00 | 0.199 | 0.00199 | 100.20 |
| 2 | 100.20 | 1.665 | 0.01665 | 101.87 |
| 3 | 101.87 | −0.445 | −0.00446 | 101.41 |
| 4 | 101.41 | −0.667 | −0.00668 | 100.74 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 100 | 92.47 | 1.153 | −0.0153 | 91.06 |

[1] The choice of the number of steps should depend on the VAR horizon and the required accuracy. A smaller number of steps will be faster to implement but may not provide a good approximation of the stochastic process.

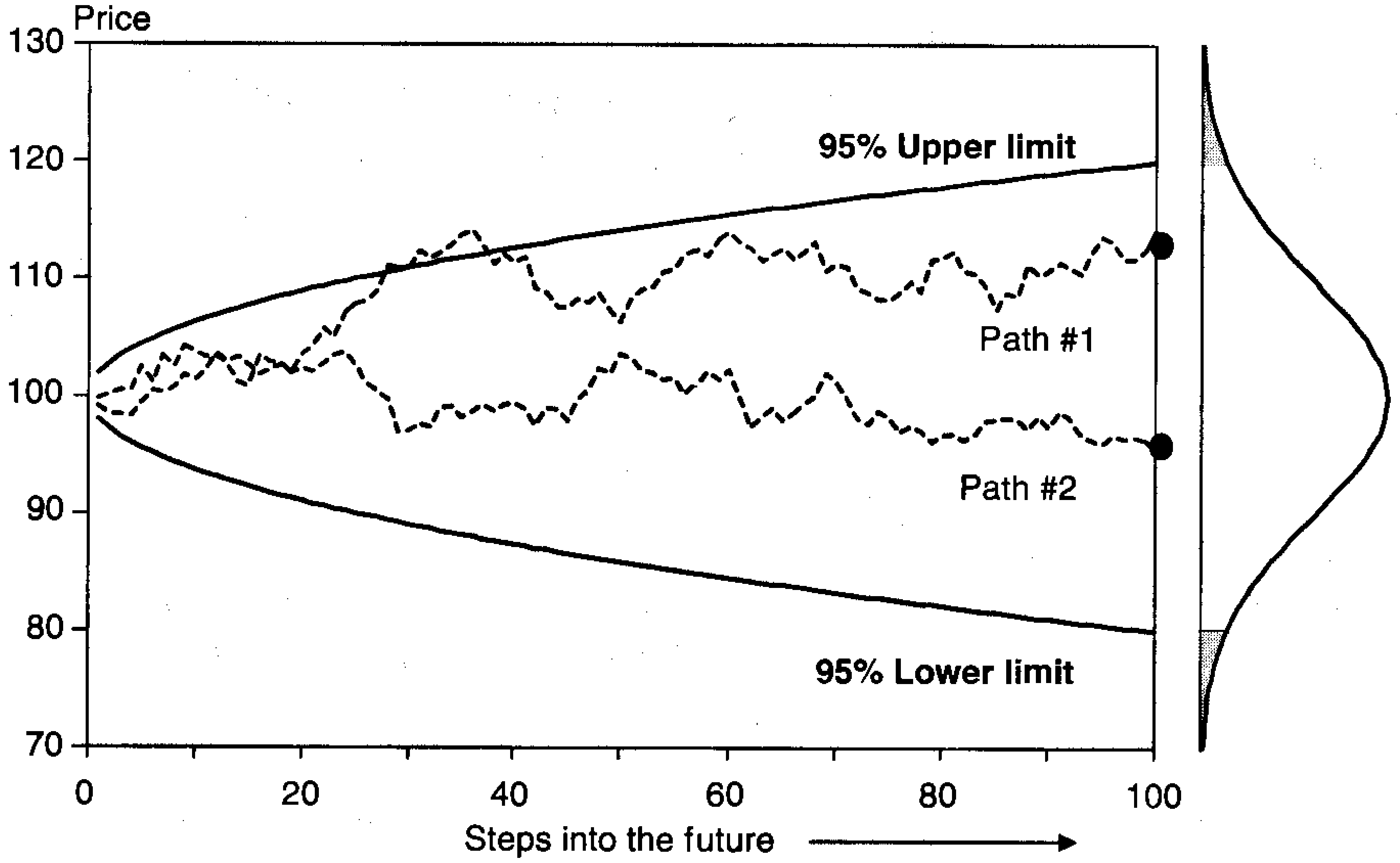
The second column starts with the initial price. The next column displays the realization of a standard normal variable. With no drift, the increment in the following column is simply ($\epsilon \times 0.01$). Finally, the last column computes the current price from the previous price and the increment. The values at each point are conditional on the simulated values at the previous point. The process is repeated until the final price of \$91.06 is reached at the 100th step.

Figure 12-1 presents two price paths, each leading to a different ending price. Given these assumptions, the ending price must follow a normal distribution with mean of \$100 and standard deviation of \$10.[2] This distribution is illustrated on the right side of the figure, along with 95 percent confidence bands, corresponding to two standard deviation intervals.

But the distribution also is known at any intermediate point. The figure displays 95 percent confidence bands that increase with the square root of time until they reach ±2 × 10 percent. In this simple model, risk can be computed at any point up to the target horizon.

**FIGURE 12-1**

Simulating price paths.



[2] In fact, the ending distribution is lognormal because the price can never fall below 0.

### 12.2.2 Creating Random Numbers

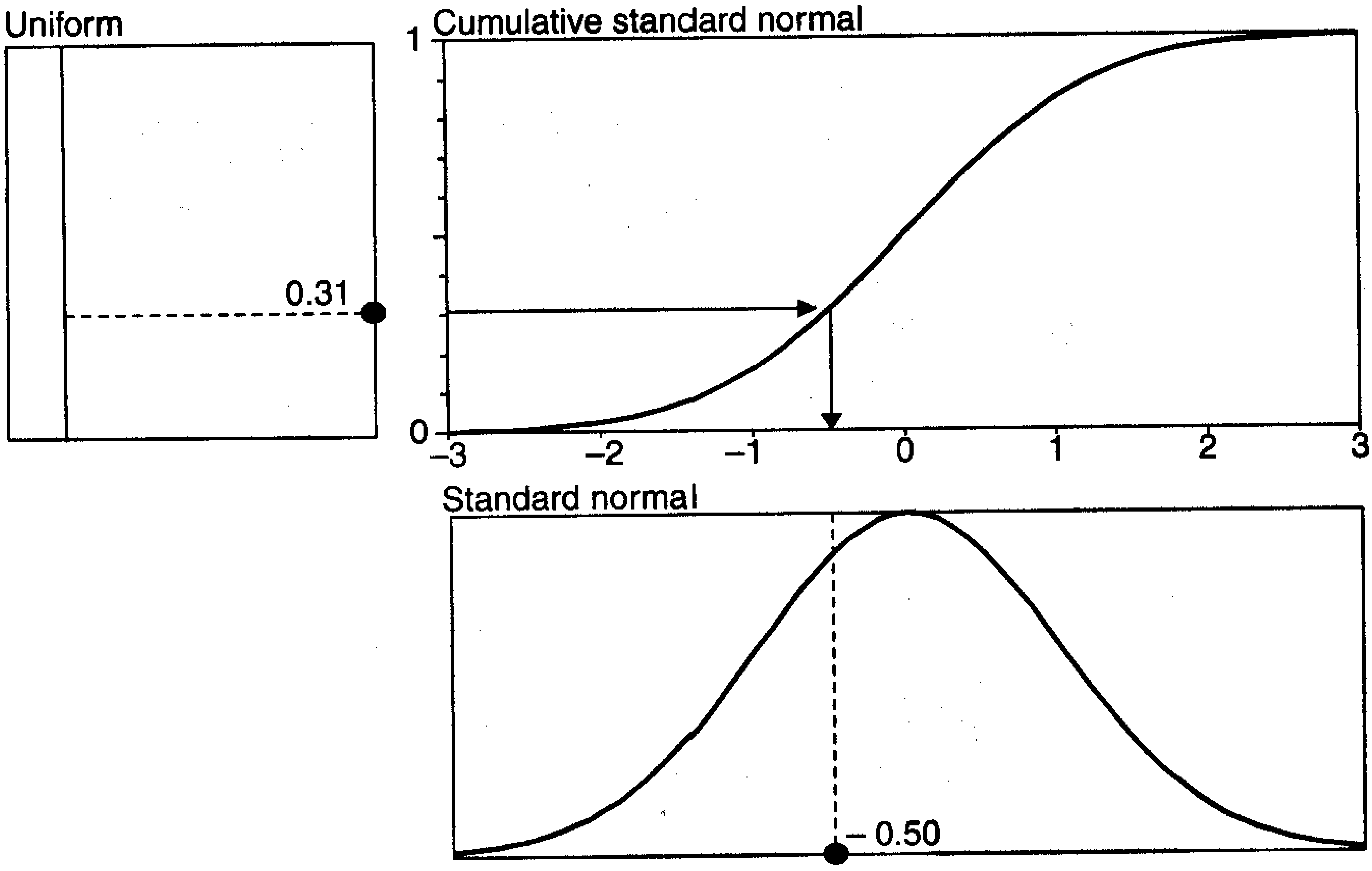
Monte Carlo simulations are based on random draws $\epsilon$ from a variable with the desired probability distribution. The numerical analysis usually proceeds in two steps.

The first building block for a random-number generator is a uniform distribution over the interval [0,1] that produces a random variable $x$. More properly speaking, these numbers are "pseudo" random because they are generated from an algorithm using a predefined rule. Starting from the same "seed" number, the sequence can be repeated at will.

The next step is to transform the uniform random number $x$ into the desired distribution through the inverse cumulative probability distribution function (pdf). Take the normal distribution. By definition, the cumulative pdf $N(y)$ is always between 0 and 1. Therefore, to generate a normally distributed random variable, we compute $y$ such that $x = N(y)$ or $y = N^{-1}(x)$.[3] More generally, any distribution function can be generated as long as the function $N(y)$ can be inverted. Figure 12-2 illustrates this procedure, called the *inverse transform method.*

**FIGURE 12-2**

Transformation from uniform to normal.



[3] Moro (1995) shows how to use approximations to the function $N^{-1}$ to accelerate the speed of computation.

At this point, an important caveat is in order. It seems easy to generate variables that are purely random, but in practice, it is quite difficult. A well-designed algorithm should generate draws that "appear" independent over time. Whether this sequence is truly random is a philosophical issue that we will not address. Good random-number generators must create series that pass all conventional tests of independence. Otherwise, the characteristics of the simulated price process will not obey the underlying model.

Most operating systems, unfortunately, provide a random-number generator that is simple but inaccurate. All algorithms "cycle" after some iterations; that is, they repeat the same sequence of pseudorandom numbers. Good algorithms cycle after billions of draws; bad ones may cycle after a few thousand only.

If the cycle is too short, dependencies will be introduced in the price process solely because of the random-number generator. As a result, the range of possible portfolio values may be incomplete, thus leading to incorrect measures of VAR. This is why it is important to use a good-quality algorithm, such as those found in numerical libraries.

### 12.2.3 The Bootstrap

An alternative to generating random numbers from a hypothetical distribution is to sample from historical data. Thus we are agnostic about the distribution. For example, suppose that we observe a series of $M$ returns $R = \Delta S/S$, $\{\mathrm{R}\} = (R_1 \cdots R_M)$, which can be assumed to be i.i.d. random variables drawn from an unknown distribution. The historical simulation method consists of using this series once to generate pseudoreturns. But this can be extended much further.

The bootstrap estimates this distribution by the empirical distribution of $R$, assigning equal probability to each realization. The method was proposed initially by Efron (1979) as a nonparametric randomization technique that draws from the observed distribution of the data to model the distribution of a statistic of interest.[4]

The procedure is carried out by sampling from $\{R\}$, with replacement, as many observations as necessary. For instance, assume that we want to generate 100 returns into the future, but we do not want to impose any assumption on the distribution of daily returns. We could project returns by randomly picking one return at a time from the sample over the

[4] The asymptotic properties of the bootstrap for commonly used statistics such as the mean, median, variance, and distribution quantiles have been studied by Bickel and Freedman (1981).

past $M = 500$ days, with replacement. Define the index choice as $m(1)$, a number between 1 and 500. The selected return then is $R_{m(1)}$, and the simulated next-day return is $S_{t+1} = S_t(1 + R_{m(1)})$. Repeating the operation for a total of 100 draws yields a total of 100 pseudovalues $S_{t+1}, \ldots, S_{t+n}$.

An essential advantage of the bootstrap is that it can include fat tails, jumps, or any departure from the normal distribution. For instance, one could include the return for the crash of October 19, 1987, which would never (or nearly never) occur under a normal distribution. The method also accounts for correlations across series because one draw consists of the simultaneous returns for $N$ series, such as stock, bonds, and currency prices.

The bootstrap approach, it should be noted, has limitations. For small sample sizes $M$, the bootstrapped distribution may be a poor approximation of the actual one. Therefore, it is important to have access to sufficient data points. The other drawback of the bootstrap is that is relies heavily on the assumption that returns are independent. By resampling at random, any pattern of time variation is broken.

The bootstrap, however, can accommodate some time variation in parameters as long as we are willing to take a stand on the model. For instance, the bootstrap can be applied to the normalized residuals of a GARCH process, that is,

$$\epsilon_t = \frac{r_t}{\sigma_t}$$

where $r_t$ is the actual return, and $\sigma_t$ is the conditional standard deviation from the estimated GARCH process. To recreate pseudoreturns, one then would first sample from the historical distribution of $\epsilon$ and then reconstruct the conditional variance and pseudoreturns.

### 12.2.4 Computing VAR

Once a price path has been simulated, we can build the portfolio distribution at the end of the selected horizon. The simulation is carried out by the following steps:

1. Choose a stochastic process and parameters.
2. Generate a pseudosequence of variables $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$, from which prices are computed as $S_{t+1}, S_{t+2}, \ldots, S_{t+n}$.
3. Calculate the value of the asset (or portfolio) $F_{t+n} = F_T$ under this particular sequence of prices at the target horizon.
4. Repeat steps 2 and 3 as many times as necessary, say, $K = 10{,}000$.

This process creates a distribution of values $F_T^1, \ldots, F_T^{10,000}$. We can sort the observations and tabulate the expected value $E(F_T)$ and the quantile $Q(F_T, c)$, which is the value exceeded in $c$ times 10,000 replications. VAR relative to the mean then is

$$\text{VAR}(c,T) = E(F_T) - Q(F_T,c) \tag{12.3}$$

### 12.2.5 Risk Management and Pricing Methods

It is interesting to note that Monte Carlo methods in finance were proposed originally in the context of options valuation.[5] Simulations are particularly useful to evaluate options that have no closed-form solution. Under the risk-neutral valuation method, Monte Carlo simulation consists of the following steps:

1. Choose a process with a drift equal to the risk-free rate, that is, with $\mu = r$ in Equation (12.1).
2. Simulate prices to the horizon $S_T$.
3. Calculate the payoff of the derivative at maturity $T$, $F(S_T)$.
4. Repeat these steps as often as needed.

The current value of the derivative is obtained from discounting at the risk-free rate and averaging across all experiments, that is,

$$f_t = E^*[e^{-r\tau}F(S_T)] \tag{12.4}$$

where the expectation indicates averaging, and the asterisk is a reminder that the price paths are under *risk neutrality*, that is, both changing the expected return and the discount rate to the risk-free rate.

This method is quite general and can be applied to options that have price-dependent paths (such as look-back options or average-rate options) or strange payoffs at expiration (such as nonlinear functions of the ending price). Its main drawback is that it cannot price options accurately when the holder can exercise early. Also, the distribution of prices must be finely measured to price options with sharp discontinuities, such as binary options, which pay a fixed amount if the price ends up above or below the strike price. With large "holes" in the price distributions, the payoffs on combinations of binary options simply could not appear in the

[5] See Boyle (1977) and, more recently, a review by Boyle et al.(1997).

final portfolio distribution. Thus highly complex payoffs can be handled with increased precision.

Monte Carlo methods also allow users to measure vega risk, or exposure to changes in volatility. All that is required is to repeat the simulation with the same sequence of $\epsilon$ values but with another value for $\sigma$. The change in the value of the asset is owing solely to the change in the volatility measures vega risk.

Overall, the methods for pricing derivatives and measuring risk have much in common. Pricing, however, requires *risk-neutral* distributions, whereas risk measurement requires *physical*, or *objective*, distributions.

## 12.3 SPEED VERSUS ACCURACY

The main drawback of Monte Carlo (MC) methods is their computational time requirements. Consider, for instance, a portfolio exposed to one risk factor only. Say that we require 10,000 replications of this risk factor for acceptable accuracy. If the portfolio contains 1000 assets to be priced using full valuations, we will need 10 million valuations.

If, in addition, the portfolio contains complex instruments, such as mortgages or exotic options, whose valuation itself requires a simulation, measuring risk at a target date then requires "a simulation within a simulation":

- For valuation (i.e., from the VAR horizon to the maturity of the instrument)
- For risk management (i.e., from the present time to the VAR horizon)

Without shortcuts, the number of required simulations soon can reach astronomical values. This is why the industry is busily developing methods to cut down the number of simulations without too much loss in accuracy.

### 12.3.1 Accuracy

Simulations inevitably generate *sampling variability*, or variations in estimator values, owing to the limited number of replications. More replications lead to more precise estimates but take longer to estimate. Define $K$ as the number of *replications*, or pseudorandom trials. To choose $K$, it is useful to assess the tradeoff between precision and the number of replications.

FIGURE 12-3

Convergence to true distribution.

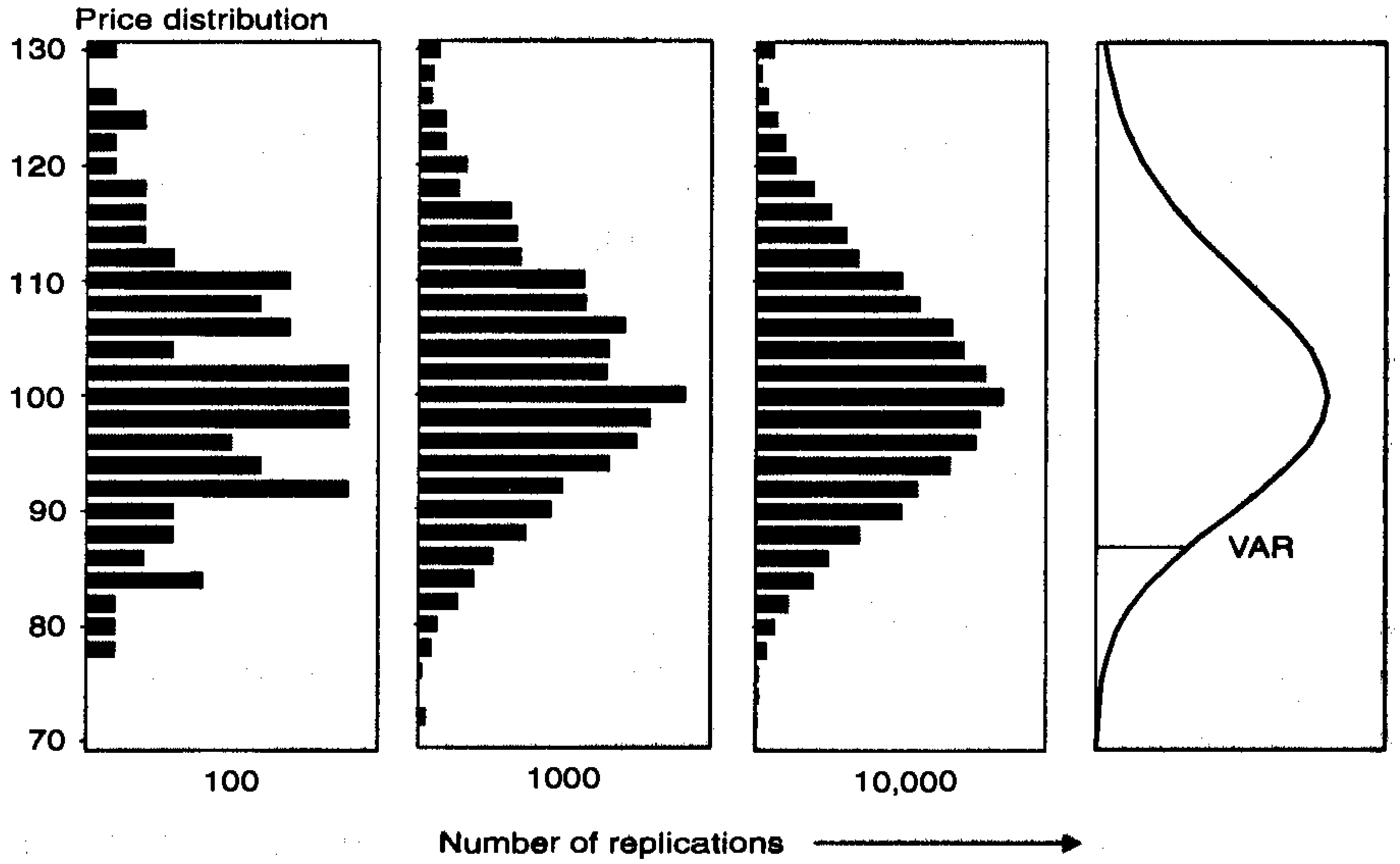


Figure 12-3 illustrates the convergence of the empirical distribution in Figure 12-1 toward the true one. With $K = 100$, the histogram representing the distribution of the ending price is quite irregular. The histogram becomes smoother with 1000 replications, even more so with 10,000 replications, and eventually should converge to the continuous distribution in the right panel. One advantage of the Monte Carlo method is that the user can evaluate the increase in accuracy directly as the number of replications increases.

If the underlying process is normal, the empirical distribution must converge to a normal distribution. In this situation, Monte Carlo analysis should yield exactly the same result as the delta-normal method: The VAR estimated from the sample quantile must converge to the value of $\alpha\sigma$.

Any deviation must be due to sampling variation. Assuming no other sonrce of error, this effect can be measured by the asymptotic standard er-
for the sample quantile reported in Chapter 5, using $K$ as the sample
:. A simple method to assess accuracy is to repeat the simulations mul-
e times, say, $M = 1000$, and take the standard error of the estimated
ntiles across the $M$ experiments. This is illustrated in Table 12-2, which

**TABLE 12-2**

Convergence Statistics for Risk Measures

| | | Standard Error | | | |
|---|---|---|---|---|---|
| | | Replications | | | |
| Left Tail | Expected Quantile | 100 | 500 | 1000 | 10,000 |
| 1% | −2.326 | 0.409 | 0.170 | 0.119 | 0.037 |
| 5% | −1.645 | 0.216 | 0.092 | 0.066 | 0.021 |
| 10% | −1.282 | 0.170 | 0.075 | 0.052 | 0.017 |
| Std.dev. | 1.000 | 0.069 | 0.032 | 0.022 | 0.007 |

describes the results of 1000 simulation runs on a standard normal distribution with an increasing number of replications.

The table shows that for a 99 percent VAR with 100 replications, the standard error of the estimate around −2.326 is 0.409, rather high. In our sample of 1000 runs, the VAR estimate ranged from −4.17 to −1.53. This dispersion is rather disturbing. To increase VAR precision by a factor of 10, we need to increase the number of replications by a factor of 100, for a total of 10,000. Indeed, the first line shows that this decreases the standard error to 0.037, which is approximately 0.409 divided by 10. Note that, in contrast, the standard deviation is estimated much more precisely because it uses data from the entire distribution.

We also could report the standard error in relative terms, defined as the ratio of the standard error to the expected value of the risk measure. For example, banks typically report their 99 percent VAR using about 500 days. From Table 12-2, this leads to a relative error in VAR of around 0.170/2.326 = 7.3 percent.

The relative error depends on the number of replications, as well as on the shape of the distribution, as shown in Table 12-3.[6] The table shows that the error is higher for left-skewed distributions and conversely lower for right-skewed distributions. This is so because the longer the left tail, the less precise is the VAR estimate.

[6] The skewed distributions were generated by long and short positions in 5-day options with a risk horizon of 1 day.

**TABLE 12-3**

Relative Error in 99 Percent VAR for Various Distributions

| | | Relative Error (Percent) | | | |
|---|---|---|---|---|---|
| | | Replications | | | |
| Distribution | Skewness | 100 | 500 | 1000 | 10,000 |
| Normal | 0.00 | 17.6 | 7.3 | 5.1 | 1.5 |
| Right skew | 0.76 | 9.3 | 4.2 | 3.0 | 0.9 |
| Left skew | −0.76 | 23.4 | 9.2 | 6.3 | 1.9 |

Alternatively, we could search for the number of replications required to measure VAR with a relative error of 1 percent. For the normal distribution, we need more than 20,000 replications to make sure that the relative error in the first row is below 1 percent.

### 12.3.2 Acceleration Methods

This led to a search for methods to accelerate computations. One of the earliest, and easiest, is the *antithetic variable technique*, which consists of changing the sign of all the random samples $\epsilon$. This method, which is appropriate when the original distribution is symmetric, creates twice the number of replications for the risk factors at little additional cost. We still need, however, twice the original number of full valuations on the target date.

This approach can be applied to the historical simulation method, where we can add a vector of historical price changes with the sign reversed. This is also useful to eliminate the effect of trends in the recent historical data.

Another useful tool is the *control variates technique*. We are trying to estimate VAR, a function of the data sample. Call this $V(X)$. Assume now that the function can be approximated by another function, such as a quadratic approximation $V^Q(X)$, for which we have a closed-form solution $v^Q$.[7]

[7] This can be found using the anaytical approximations described in Appendix 10.A. Note that $v^Q$ does not depend on the random sample $X$.

For any sample, the error then is known to be $V^Q(X) - v^Q$ for the quadratic approximation. If this error is highly correlated with the sampling error in $V(X)$, the control variate estimator can be taken as

$$V_{CV} = V(X) - [V^Q(X) - v^Q] \tag{12.5}$$

This estimator has much lower variance than the original one when the quadratic function provides a good approximation of the true function.

The most effective acceleration method is the *importance sampling technique*, which attempts to sample along the paths that are most important to the problem at hand. The idea is that if our goal is to measure a tail quantile accurately, there is no point in doing simulations that will generate observations in the center of the distribution. The method involves shifts in the distribution of random variables. Glasserman et al. (2000) show that relative to the usual Monte Carlo method, the variance of VAR estimators can be reduced by a factor of at least 10.

A related application is the *stratified sampling technique*, which can be explained intuitively as follows: assume that we require VAR for a long position in a call option.[8] We are trying to keep the number of replications at $K = 1000$. To increase the accuracy of the VAR estimator, we could partition the simulation region into two zones. As before, we start from a uniform distribution, which then is transformed into a normal distribution for the underlying asset price using the *inverse transform method.*

Define these two zones, or strata, for the uniform distribution as [0.0, 0.1] and [0.1, 1.0]. Thus *stratification* is the process of grouping the data into mutually exclusive and collectively exhaustive regions. Usually, the probabilities of the random number falling in both zones are selected as $p_1 = 10$ percent and $p_2 = 90$ percent, respectively. Now we change these probabilities to 50 percent for both regions. The number of observations now is $K_1 = 500$ for the first region and $K_2 = 500$ for the second. This increases the number of samples for the risk factor in the first, left-tail region.

Estimators for the mean need to be adjusted for the stratification. We weight the estimator for each region by its probability, that is,

$$E(F_T) = p_1 \frac{\sum_{i=1}^{K_1} F_i}{K_1} + p_2 \frac{\sum_{i=1}^{K_2} F_i}{K_2} \tag{12.6}$$

[8] More generally, the classification can be based on a quadratic approximation. See Cardenas et al. (1999) for more details.

To compute VAR, we simply examine the first region. The 50 percent quantile for the first region, for example, provides an estimator of a 10 × 0.5 = 5 percent left-tail VAR. Because VAR only uses the number of observations in the right region, we do not even need to compute their value, which economizes on the time required for full valuation.

This reflects the general principle that using more information about the portfolio distribution results in more efficient simulations. In general, unfortunately, the payoff function is not known. All is not lost, however. Instead, the simulation can proceed in two passes. The first pass runs a traditional Monte Carlo. The risk manager then examines the region of the risk factors that causes losses around VAR. A second pass then is performed with many more samples from this region.

## 12.4 SIMULATIONS WITH MULTIPLE VARIABLES

Modern risk measurement applications are large-scale problems because they apply at the highest level of the financial institution. This requires simulations over multiple sources of risk.

### 12.4.1 From Independent to Correlated Variables

Simulations generate independent random variables that need to be transformed to account for correlations. Define $N$ as the number of sources of risk. If the variables are uncorrelated, the randomization can be performed independently for each variable, that is,

$$\Delta S_{j,t} = S_{j,t-1}\,(\mu_j \Delta t + \sigma_j \epsilon_{j,t}\sqrt{\Delta t}) \tag{12.7}$$

where the $\epsilon$ values are independent across time period and series $j = 1, \ldots, N$.

To account for correlations between variables, we start with a set of independent variables $\eta$, which then are transformed into the $\epsilon$. In a two-variable setting, we construct

$$\begin{aligned} \epsilon_1 &= \eta_1 \\ \epsilon_2 &= \rho\eta_1 + (1-\rho^2)^{1/2}\eta \end{aligned} \tag{12.8}$$

where $\rho$ is the correlation coefficient between the variables $\epsilon$. First, we verify that the variance of $\epsilon_2$ is unity, that is,

$$V(\epsilon_2) = \rho^2 V(\eta_1) + [(1 - \rho^2)^{1/2}]^2 V(\eta_2) = \rho^2 + (1 - \rho^2) = 1$$

Then we compute the covariance of the $\epsilon$ *as*

$$\mathrm{cov}(\epsilon_1,\epsilon_2) = \mathrm{cov}[\eta_1, \rho\eta_1 + (1 - \rho^2)^{1/2}\eta_2] = \rho\, \mathrm{cov}(\eta_1, \eta_1) = \rho$$

This confirms that the $\epsilon$ variables have correlation of $\rho$. The question is, how was the transformation in Equation (12.8) chosen?

More generally, suppose that we have a vector of $N$ values of $\epsilon$ for which we would like to display some correlation structure $V(\epsilon) = E(\epsilon\epsilon')$ = R. We will use *Cholesky factorization,* named after the French mathematician André-Louis Cholesky, to generate correlated variables. Since the matrix $R$ is a symmetric real matrix, it can be decomposed into its *Cholesky factors*, this is,

$$R = TT' \tag{12.9}$$

where $T$ is a lower triangular matrix with zeros in the upper right corners.

We start with an $N$ vector $\eta$ that is composed of independent variables all with unit variances. In other words, $V(\eta) = I$, where $I$ is the identity matrix with zeroes everywhere except on the diagonal. Next, construct the variable $\epsilon = T\eta$. Its covariance matrix is $V(\epsilon) = \mathrm{E}(\epsilon\epsilon') = E(T\eta\eta'T') = TE(\eta\eta')T' = TIT' = TT' = R$. Thus we have confirmed that the values of $\epsilon$ have the desired correlations.

As an example, consider the two-variable case. The matrix can be decomposed into

$$\begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} = \begin{bmatrix} a_{11} & 0 \\ a_{12} & a_{22} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} \\ 0 & a_{22} \end{bmatrix} = \begin{bmatrix} a_{11}^2 & a_{11}a_{12} \\ a_{11}a_{12} & a_{12}^2 + a_{22}^2 \end{bmatrix}$$

The entries in the right-hand-side matrix must match exactly each entry in the correlation matrix. Because the Cholesky matrix is triangular, the factors can be found by successive substitution by setting

$$a_{11}^2 = 1$$
$$a_{11}a_{12} = \rho$$
$$a_{12}^2 + a_{22}^2 = 1$$

which yields $$\begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \rho & (1-\rho^2)^{1/2} \end{bmatrix} \begin{bmatrix} 1 & \rho \\ 0 & (1-\rho^2)^{1/2} \end{bmatrix}$$

And indeed, this is how Equation (12.8) was obtained:

$$\begin{bmatrix}\epsilon_1\\ \epsilon_2\end{bmatrix}=\begin{bmatrix}1 & 0\\ \rho & (1-\rho^2)^{1/2}\end{bmatrix}\begin{bmatrix}\eta_1\\ \eta_2\end{bmatrix}$$

This explains how a multivariate set of random variables can be created from simple building blocks consisting of i.i.d. variables. In addition to providing a method to generate correlated variables, however, this approach generates valuable insight into the random number generation process.

### 12.4.2 Number of Risk Factors

For the decomposition to work, the matrix $R$ must be *positive definite.* Otherwise, there is no way to transform $N$ independent source of risks into $N$ correlated variables of $\epsilon$.

As discussed in Chapter 8, this condition can be verified with the *singular value decomposition.* This decomposition of the covariance matrix provides a check that the matrix is well behaved. If any of the eigenvalues is zero or less than zero, the Cholesky decomposition will fail.

When the matrix $R$ is not positive definite, its *determinant* is zero. Intuitively speaking, the determinant $d$ is a measure of the "volume" of a matrix. If $d$ is zero, the dimension of the matrix is less than $N$. The determinant can be computed easily from the Cholesky decomposition. Since the matrix $T$ has zeros above its diagonal, its determinant reduces to the product of all diagonal coefficients $d_T = \Pi_{i=1}^{N} a_{ii}$. The determinant of the covariance matrix $R$ then is $d = d_T^2$.

In our two-factor example, the matrix is not positive definite if $\rho = 1$. In practice, this implies that the two factors are really the same. The Cholesky decomposition then yields $a_{11} = 1$, $a_{12} = 1$, and $a_{22} = 0$, and the determinant $d = (a_{11}a_{22})^2$ is 0. As a result, the second factor $\eta_2$ is never used, and $\epsilon_1$ is always the same as $\epsilon_2$. The second random variable is totally superfluous. In this case, the covariance matrix is not positive definite. Its true dimension, or *rank*, is 1, which means that it has only one meaningful risk factor.

These conditions may seem academic, but unfortunately, they soon become very real with simulations based on a large number of factors. The RiskMetrics covariance matrix, for instance, is routinely nonpositive definite owing to the large number of assets. These problems can arise for a number of reasons. Perhaps this is simply due to the large number of correlations. With $N = 450$, for instance, we have about 100,000 correlations with rounding errors. This also could happen when the effective number of observations $T$ is less than the number of factors $N$. One drawback of time-varying models of variances is that they put less weight on older observations, thereby reducing the effective sample size. Or the correlations may have been measured over different periods, which may produce inconsistent correlations.[9] Another reason would be that the series are naturally highly correlated (such as the 9-year zero-coupon bond with the adjoining maturities) or that some series were constructed as a linear combination of others (such as a currency basket).

For simulations, this may be a blessing in disguise because fewer number of variables are sufficient. In Chapter 8 we gave the example of 11 bonds for which the covariance matrix could be reduced without much loss of information to two, or perhaps three, principal components. Thus the problem can be solved using a matrix of smaller dimensions, which speeds up the computation considerably. This illustrates that the design of simulation experiments, including the number of risk factors, is critical. As we have seen, however, the choice of the number of risk factors should be related to the trading strategy.

## 12.5 DETERMINISTIC SIMULATION

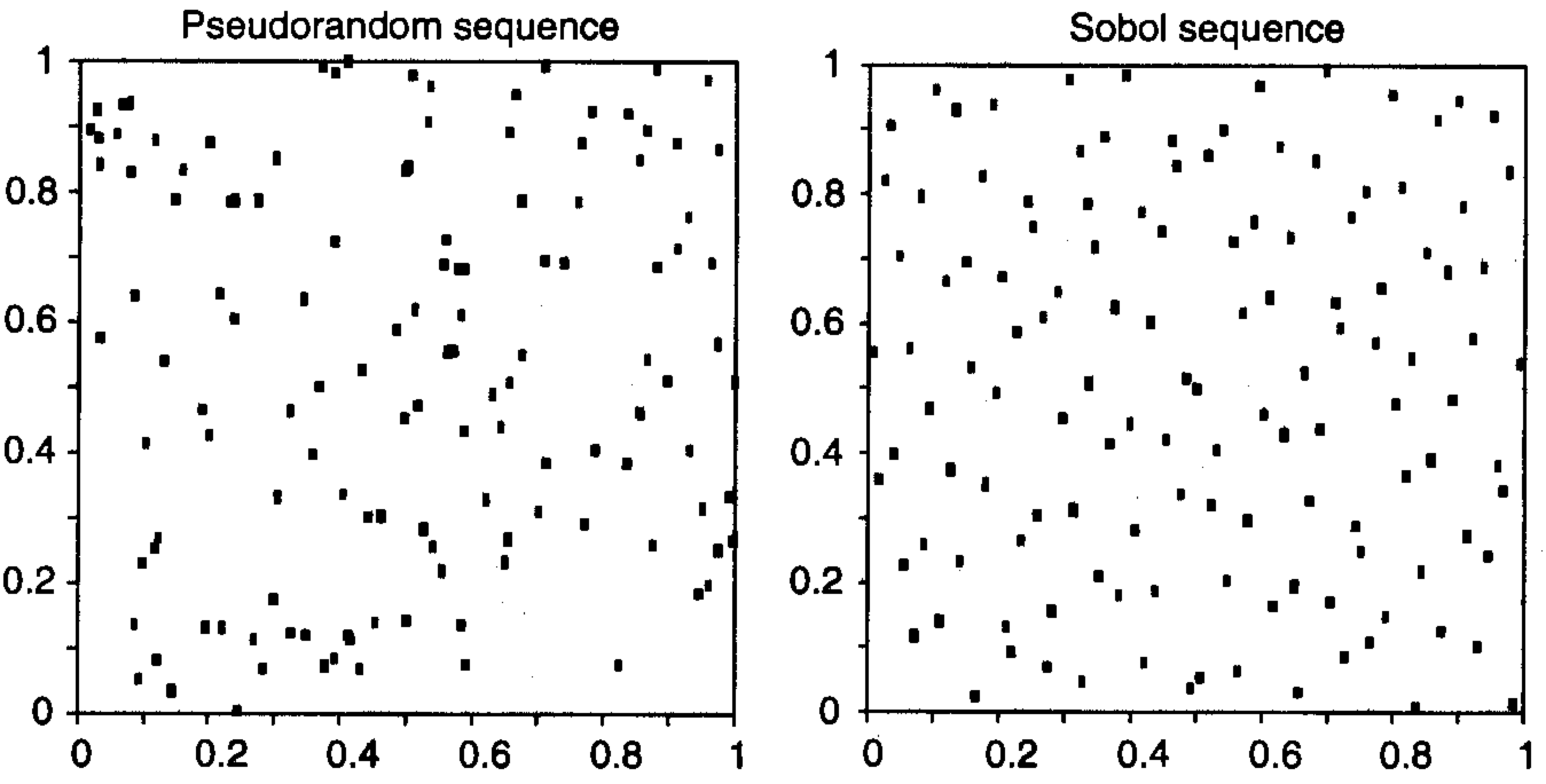
Monte Carlo simulation methods generate independent, pseudorandom points that attempt to "fill" an $N$-dimensional space, where $N$ is the number of risk factors. The sequence of points does not have to be chosen randomly, however.

Indeed, it is possible to use a *deterministic* scheme that is constructed to provide a more consistent fill to the $N$-space. The choice must account for the sample size, dimensionality of the problem, and possibly the shape of the function being integrated. These deterministic schemes are sometimes called *quasi–Monte Carlo* (QMC), although this is a misnomer because there is nothing random about them. The numbers are not independent but rather are constructed as an ordered sequence of points.

[9] As an example, consider three assets. We have 1 year of daily data for assets A and B, which have a high correlation, say, 0.9. For asset C, we only have 1 month of data. If, over the shorter period, asset C has a high observed correlation with A and a low enough correlation with B, the correlation matrix will be inconsistent. For instance, if the correlation between C and A is 0.9, the lowest possible correlation between C and B is 0.62.

FIGURE 12-4

Comparison of distributions.



To illustrate, Figure 12-4 compares a distribution for two variables only (after all, this is the number of dimensions of a page). The figure shows, on the left, pseudorandom points and, on the right, a deterministic, *low-discrepancy* sequence obtained from a so-called Sobol procedure.[10]

The left graph shows that the points often "clump" in some regions and leave out large areas. These clumps are a waste because they do not contribute more information. The right panel, in contrast, has more uniform coverage. Instead of drawing independent samples, the deterministic scheme systematically fills the space left by the previous numbers in the series.

Quasirandom methods have the desirable property that the standard error shrinks at a faster rate, proportional to close to $1/K$ rather than $1/\sqrt{K}$ for standard simulations. Indeed, a number of authors have shown that deterministic methods provide a noticeable improvement in speed.[11] Papageorgiou and Paskov (1999) compare the computation of VAR for a portfolio exposed to 34 risk factors using 1000 points. They find that the deterministic sequence can be 10 times more accurate than the Monte Carlo method.

[10] This algorithm is described in Press et al. (1992).

[11] See, for example, Boyle et al. (1997) for call options and Paskov and Traub (1995) for mortgage securities.

One drawback of these methods is that since the draws are not independent, accuracy cannot be assessed easily. For the Monte Carlo method, in contrast, we can construct confidence bands around the estimates. Another issue is that for high-dimensionality problems, some QMC sequences tend to cycle, which leads to decreases in performance. Overall, however, suitably selected QMC methods can provide substantial accelerations in the computations.

## 12.6 CHOOSING THE MODEL

Simulation methods are most prone to model risk. If the stochastic process chosen for the price is unrealistic, so will be the estimate of VAR. This is why the choice of the underlying process is particularly important.

For example, the geometric brownian motion model in Equation (12.1) adequately describes the behavior of some financial variables, such as stock prices and exchange rates, but certainly not that of fixed-income securities. In the brownian motion models, shocks to the price are never reversed, and prices move as a random walk. This cannot represent the price process for default-free bond prices, which must converge to their face value at expiration.

Another approach is to model the dynamics of interest rates as

$$dr_t = \kappa(\theta - r_t)dt + \sigma r_t^{\gamma} dz_t \qquad (12.10)$$

This class of model includes the Vasicek (1977) model when $\gamma = 0$; changes in yields then are normally distributed, which is particularly convenient because this leads to many closed-form solutions. With $\gamma = 0.5$, this is also the Cox, Ingersoll, and Ross (1985) model of the term structure (CIR). With $\gamma = 1$, the model is lognormal.[12]

This process is important because it provides a simple description of the stochastic nature of interest rates that is consistent with the empirical observation that interest rates tend to be mean-reverting. Here, the parameter $\kappa < 1$ defines the speed of mean reversion toward the long-run value $\theta$. Situations where current interest rates are high, such as $r_t > \theta$, imply a negative drift $\kappa(\theta - r_t)$ until rates revert to $\theta$. Conversely, low current rates are associated with positive expected drift. Also note that with $\gamma = 0.5$, the variance of this process is proportional to the level of

[12] Bliss and Smith (1998) show that $\gamma = 0.5$ provides a good fit to U.S. short-term interest rates.

interest rates; as the interest rate moves toward 0, the variance decreases, so $r$ can never fall below 0. If the horizon is short, however, the trend or mean reversion term will not be important.

Equation (12.10) describes a one-factor model of interest rates that is driven by movements in short-term rates $dr_t$. In this model, movements in longer-term rates are perfectly correlated with movements in this short-term rate through $dz$. Therefore, the correlation matrix of zero-coupon bonds consists of ones only. This may be useful to describe the risks of some simple portfolios but certainly not for the leveraged fixed-income portfolios of financial institutions.

For more precision, additional factors can be added. Brennan and Schwartz (1979), for example, proposed a two-factor model with a short and long interest rate modeled as

$$dr_t = \kappa_1(\theta_1 - r_t)dt + \sigma_1 dz_{1t} \tag{12.11}$$

$$dl_t = \kappa_2(\theta_2 - l_t)dt + \sigma_2 dz_{2t} \tag{12.12}$$

where $l$ is the long rate and the errors have some correlation. Generalizing, one could use the full covariance matrix along some 14 points on the yield curve, as provided by RiskMetrics. In theory, *finer granularity* should result in better risk measures, albeit at the expense of computational time. In all these cases, the Monte Carlo experiment consists of first simulating movements in the driving risk factors and then using the simulated term structure to price the securities at the target date.

Here is where risk management differs from valuation methods. For short horizons (say, 1 day to 1 month), we could assume that changes in yields are jointly normally distributed. This assumption may be quite sufficient for risk management purposes. Admittedly, it would produce inconsistencies over long horizons (say, beyond a year) because yields could drift in different directions, creating term structures that look unrealistic.[13]

With longer horizons, the drift term in Equation (12.11), for example, becomes increasingly important. To ensure that the two rates cannot move too far away from each other, one could incorporate into the drift of the short rate an *error-correction term* that pushes the short rate down when it is higher than the long rate. For instance, one could set

[13] This explains why the Black model is used often to price short-term options on long-term bonds. Although theoretically inconsistent, it produces good results because the maturity of the option is so short relative to that of the underlying.

$$dr_t = \kappa_1[\theta_1 - (r_t - l_t)]dt + \sigma_1 dz_{1t} \qquad (12.13)$$

Indeed, much work has been devoted to the analysis of time series that are *cointegrated*, that is, that share a common random component.[14] These error-correction mechanisms can be applied to larger-scale problems, thus making sure that our 14 yields move in a realistic fashion.

But again, over short horizons, the modeling of expected returns is not too important. This also applies to the choice of term structure models, equilibrium models versus arbitrage models. *Equilibrium models* postulate a stochastic process for some risk factors, which generates a term structure. This term structure, however, will not fit exactly the current term structure, which is not satisfactory for fixed-income option traders. They argue that if the model does not even fit current bond prices, it cannot possibly be useful to describe options. This is why *arbitrage models* take today's term structure as an input (instead of output for the equilibrium models) and fit the stochastic process accordingly.

For instance, a one-factor no-arbitrage model is

$$dr_t = \theta(t)dt + \sigma dz_t \qquad (12.14)$$

where the function $\theta(t)$ is chosen so that today's bond prices fit the current term structure. This approach has been extended to two-factor Heath-Jarrow-Morton (1992) models, but their estimation is computer-intensive and has been described "at the very boundaries of feasibility."[15] These *arbitrage* models are less useful for risk management, however.

For risk management purposes, what matters is to capture the richness in movements in the term structure, not necessarily to price today's instruments to the last decimal point. Thus the "art" of risk management lies in deciding what elements of the model are important.

## 12.7 CONCLUSIONS

Simulation methods are now used widely for risk management purposes. Interestingly, these methods can be traced back to the valuation of complex options, except that there is no discounting or risk-neutrality assumption. Thus the investment in intellectual and systems development for derivatives

[14] Much of the groundbreaking work in cointegration was done by Engle and Granger while at the University of California at San Diego. See Engle and Granger (1991), a good review book.

[15] See Rebonato (1996).

TABLE 12-4

Comparison of VAR Methods

| | Valuation Method | |
|---|---|---|
| Risk-Factor Distribution | Local Valuation | Full Valuation |
| Variance-covariance | Delta-normal | |
| Historical simulation | | Historical path |
| Deterministic simulation | | Full simulation |
| Monte Carlo simulation | Delta-gamma-MC | Full Monte Carlo |

trading can be used readily for computing VAR. No doubt this is why officials at the Fed have stated that derivatives "have had favorable spillover effects on institutions' abilities to manage their total portfolios."

Simulation methods are quite flexible. They can either postulate a stochastic process or resample from historical data. They allow full valuation on the target date. On the downside, they are more prone to model risk owing to the need to prespecify the distribution and are much slower and less transparent than analytical methods. In addition, simulation methods create sampling variation in the measurement of VAR. Greater precision comes at the expense of vastly increasing the number of replications, which slows the process down.

VAR methods are listed in Table 12-4 in order of increasing time requirement. At one extreme is the Monte Carlo method, which requires the most computing time. For the same accuracy, deterministic simulations are faster because they create more systematic coverage of the risk factors. Next is the historical simulation method, which uses recent history in a limited number of simulations. At other extreme is the delta-normal method, which requires no simulation and is very fast. With the ever-decreasing cost of computing power and advances in scientific methods, however, we should expect greater use of simulation methods.

## QUESTIONS

1. What is the main assumption for the risk factors underlying the Monte Carlo simulation method?
2. What is the main assumption for the risk factors underlying the historical simulation method?

3. Explain why numerical integration is plagued by the curse of dimensionality and why this is avoided by the Monte Carlo simulation method.
4. Define $K$ as the number of Monte Carlo replications. At what rate does the standard error of estimates decrease?
5. What are the major drawbacks of the Monte Carlo simulation method?
6. Consider an operating system that has a random-number generator with a short cycle, that is, that repeats the same sequence of numbers after a few thousand iterations. Will this lead to inaccuracy in the calculation of VAR? Why?
7. Explain how the inverse transform method could generate draws from a student $t$ distribution.
8. What is an advantage of the bootstrap approach compared with a Monte Carlo simulation based on the normal distribution?
9. If the movements in the risk factors have positive autocorrelation from one day to the next, can we bootstrap on the changes in the risk factors?
10. To compute VAR using a simulation method, which two statistics are required?
11. Can Monte Carlo simulation be adapted to changing volatility?
12. Explain why pricing methods use risk-neutral distributions. Does risk measurement need risk-neutral or physical distributions?
13. A Monte Carlo simulation creates a 99 percent VAR estimate of $10 million with a standard error of $4 million using 1000 replications. How many replications are needed to shrink this standard error to less than $1 million?
14. The relative error in the previous question was 4/10. Would you expect this ratio to be higher or lower for a 95 percent VAR?
15. How many years of daily data do we need to estimate a 99 percent VAR with a precision of 1 percent or better? Would we need more/ fewer years for distributions with negative skewness, and why?
16. Explain how stratified sampling could generate more precise estimates of VAR.
17. A risk manager needs to generate two variables with a correlation of 0.6. Explain how this could be done starting from independent variables. Verify that the final variables have unit volatility.
18. Assume now that the correlation between the two variables is 1. What does this imply in terms of independent risk factors for the simulation?

19. Are sequences of variables in the quasi–Monte Carlo methods independent?
20. Is the geometric brownian motion model a good description of the behavior of fixed-income securities?
21. Explain how equilibrium and no-arbitrage models use the current term structure.