Monte Carlo Simulation and Copula Function

by
Gerardo Ferrara

Introduction

- A Monte Carlo method is a computational algorithm that relies on repeated random sampling to compute its results.
- In a nutshell, instead of performing long complex calculations, we perform a large number of "experiments" using a quasi random number generation and see what happens.
- Monte Carlo methods tend to be used when it is infeasible or impossible to compute an exact result with a deterministic algorithm.

Background/History

- “Monte Carlo” from the gambling town of the same name (no surprise).
- First applied in 1947 to model diffusion of neutrons through fissile materials.
- Limited use because time consuming.
- Much more common since late 80’.

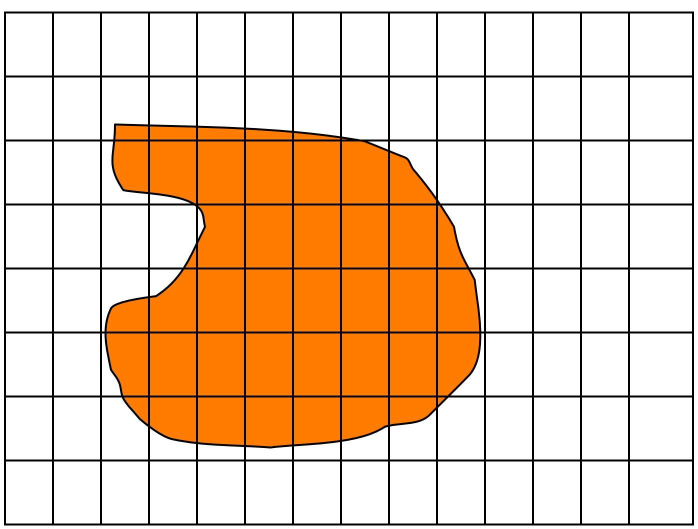
The steps in Monte Carlo simulation corresponding to the uncertainty propagation are relatively simple:

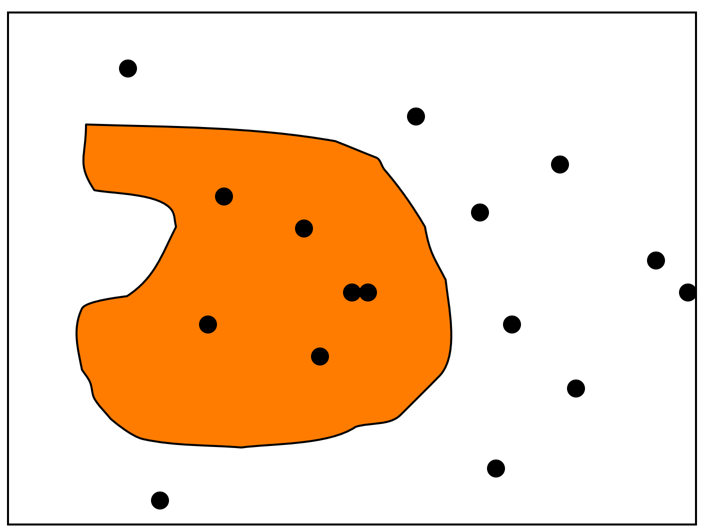
- **Step 1: Create a parametric model**, $y = f(x1, x2, ..., xq)$.
- **Step 2: Generate a set of random inputs**, $xi1, xi2, ..., xiq$.
- **Step 3: Evaluate the model** and store the results as yi.
- **Step 4: Repeat** steps 2 and 3 for $i = 1$ to n.
- **Step 5: Analyze the results** using histograms, summary statistics, confidence intervals, etc.

EXAMPLE – Area of a figure

- Cover the figure by a grid, calculate the number of grid cells which are inside and this gives you the area.
- Shoot at random at the figure. Count the bullets that hit it. The area of then figure is

  S=(Nhit/Ntotal)*S(rectangle)





Monte Carlo Methods

- A Monte Carlo simulation creates samples from a known distribution.

- For example, if you know that a coin is weighted so that heads will occur 90% of the time, then you might assign the following values:

X	0	1
$f_X(x)$	0.10	0.90

Monte Carlo Methods (cont.)

- If you tossed the coin, the expected value would be 0.9

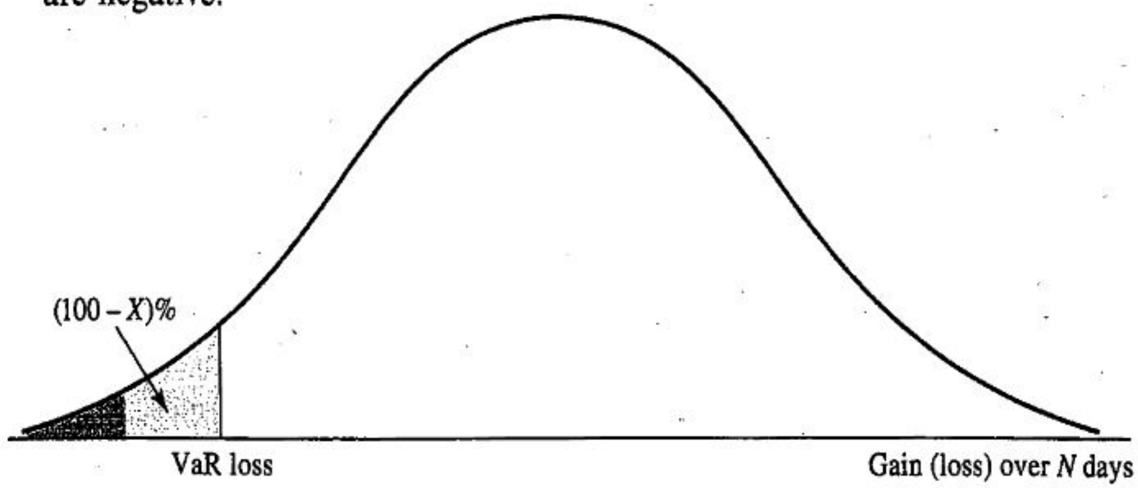
- However, a sample simulation might yield the results 1, 1, 1, 0, 1, 1, 0, 1, 0, 1.

- The average of the sample is 0.7 (close, but not the same as the expected average).

Value at Risk (VaR)

- "We are X percent certain that we will not lose more than V dollars in time T".
- Function of confidence level X and time T.

Figure 20.1 Calculation of VaR from the probability distribution of the change in the portfolio value; confidence level is X%. Gains in portfolio value are positive; losses are negative.



Pseudo Random Number Generators

- Monte Carlo simulations are based on computer generation of pseudo random numbers.
- Starting point is generation of sequence of independent, identically distributed uniform ($U(0,1)$) random variables:
  - $U(0,1)$ random numbers of direct interest in some applications;
  - More commonly, $U(0,1)$ numbers transformed to random numbers having ***other*** distributions.

Example Use of Simulation: Monte Carlo Integration

- Common problem is estimation of $\int_{\Omega} f(\boldsymbol{x})d\boldsymbol{x}$ where *f* is a function, ***x*** is vector and Ω is domain of integration
  - Monte Carlo integration popular for complex *f* and/or Ω.
- Special case: Estimate $\int_a^b f(x)dx$ for scalar *x*, and limits of integration *a*, *b.*
- *One* approach:
  - Let *p*(*u*) denote uniform density function over [*a*, *b*]
  - Let U_i denote *i* th uniform random variable generated by Monte Carlo according to the density *p*(*u*)
  - Then, for "large" *n*:

$$\int_a^b f(x)dx \approx \frac{b-a}{n}\sum_{i=1}^{n} f(U_i)$$

Numerical Example of Monte Carlo Integration

- Suppose interested in $\int_0^b \sin(x)dx$
  - Simple problem with known solution.
- Considerable variability in quality of solution for varying *b*
  - Accuracy of numerical integration *sensitive* to integrand and domain of integration.

Integral estimates for varying *n*			
	n = 20	*n* = 200	*n* = 2000
b = π (ans.=2)	2.296	2.069	2.000
b = 2π (ans.=0)	0.847	0.091	-0.0054

Homework Exercise 1

This problem uses the Monte Carlo integration technique to estimate

$$\int_a^b 2x^2 + 3x - 1\,dx$$

for varying *a*, *b*, and *n.* Specifically:

(a) To at least 3 post-decimal digits of accuracy, what is the *true* integral value when *a* = 0, *b* = 1? And for *a* = 0, *b* = 4?

(b) Using *n* = 20, 200, and 2000, estimate (via Monte Carlo) the integral for the two combinations of *a* and *b* in part (a).

(c) Comment on the relative accuracy of the two settings.

Copulas

Suppose you want to generate samples from some distribution with probability density f(x). All you need is a source of uniform random variables, because you can transform these random variables to have the distribution that you want (Sklar's Theorem).

General algorithm

- Generate (w1,w2) from a Multivariate Normal.
- Get u = F(w1), v = F(w2) where F(x) is normal cumulative distribution function (CDF).
- Generate x = G^{-1}(u), y = G^{-1}(v) where G^{-1} is empirical CDF from data.
- The distribution multivariate normal distribution is important; this is what controls dependence at the uniform density stage.

What is an empirical CDF?

- Given a vector S_t of observations (then you can use the "ecdf" function in R).
- The methodology assigns a 1/n probability to each observation, orders the data from smallest to largest in value, and calculates the sum of the assigned probabilities up to and including each observation. The result is a step function that increases by at each datum.
- $p = G(z) = \text{fraction}(S_t \leq z)$
- $G^{-1}(p) = \text{quantile}(S_t, p)$