Numerical Methods

Monte Carlo Method

Oleg Kozlovski

1 Central Limit Theorem

Theorem 1.1. *Let $\xi_i \colon \Omega \to \mathbb{R}$ be i.i.d. (independent random variables with the* **same** *distribution and with $\mu = E\xi_i < \infty$ and $\sigma^2 = \operatorname{var} \xi_i < \infty$) then*

$$\lim_{n\to\infty} \; P\left\{ \left(\frac{(\xi_1 - \mu) + \cdots + (\xi_n - \mu)}{n} \right) \in [A\frac{\sigma}{\sqrt{n}}, B\frac{\sigma}{\sqrt{n}}] \right\} =$$

$$= \frac{1}{\sqrt{2\pi}} \int_A^B e^{-x^2/2}\, dx.$$

A consequence of the central limit theorem is that

$$P\left(\bar{\xi}_n - 1.96\frac{\sigma}{\sqrt{n}} \leq \mu \leq \bar{\xi}_n + 1.96\frac{\sigma}{\sqrt{n}} \right) \approx 0.95$$

So

$$\left[\bar{\xi}_n - 1.96\frac{\sigma}{\sqrt{n}}, \bar{\xi}_n + 1.96\frac{\sigma}{\sqrt{n}} \right]$$

is a 95% *confidence interval* of μ. Or more generally, if z_α is the numbers so that $P(|Z| \leq z) = \alpha$ for some $\alpha > 0$ then with $1 - \alpha$ confidence one gets the above inequality with 1.96 replaced by z. (For example, $\alpha = 0.01$ corresponds to $z_\alpha \approx 2.58$.)

2 Monte Carlo Methods for integration

One way of integrating a function is to take averages of a randomly chosen set of function values:

Let y_i, $i = 0, 1, \ldots$ be i.i.d. with $y_i \sim U(0, 1)$. Moreover, let $f \colon [0, 1] \to \mathbb{R}$ be a continuous function and write $\xi_i = f(y_i)$. By the strong law of large numbers for almost all ω,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} f(y_i(\omega)) = \int_0^1 f(t)\, dt.$$

The central limit theorem gives the following error bound, There is a 95% probability that

$$\int_0^1 f(t)\,dt$$

lies between

$$\left[\frac{(\xi_1+\cdots+\xi_n)}{n}-1.96\frac{\sigma}{\sqrt{n}},\frac{(\xi_1+\cdots+\xi_n)}{n}+1.96\frac{\sigma}{\sqrt{n}}\right]$$

where σ^2 is equal to $\int_0^1[f(t)]^2\,dt\ -I^2$, and could again be estimated from the sample. So in order to increase the precision (but again with the same confidence) by a factor 100 one needs to increase the sample size by a factor 10000.

It is good to compare this with most simple classical numerical integration scheme (so not using a Monte Carlo approach). Let $f\colon[0,1]\to\mathbb{R}$ be differentiable and such that $|f'|\leq M$, and take y_i be equally spaced: $y_i=i/n$, $i=0,\ldots,n$. Then by the Mean-Value Theorem, $f(x)\in[f(y_i)-M/n,f(y_i)+M/n]$ for each $x\in[y_i,y_{i+1}]$. So

$$\int_{y_i}^{y_{i+1}} f(t)\,dt\in\frac{f(y_i)}{n}+[-\frac{M}{n^2},\frac{M}{n^2}].$$

In fact, if f is twice differentiable then there exists a constant M so that

$$\int_{y_{i-1}}^{y_i} f(t)\,dt\in\frac{1}{n}f(z_i)+[-\frac{M}{n^3},\frac{M}{n^3}],$$

where $z_i=(i-\frac{1}{2})/n$. Hence

$$\int_0^1 f(t)\,dt\in\frac{1}{n}\sum_{i=1}^{n}f(z_i)+[-Mn^{-2},Mn^{-2}].$$

This implies that using the non-probabilistic method, one increases the precision by a factor 100 by merely increasing the sample size by a factor 10. So for such a nice function defined on an interval it does not really make sense to use Monte Carlo methods for estimating the integral.

But all these methods get less efficient in higher dimensions: if $f\colon[0,1]^d\to\mathbb{R}$ the error bounds get worse for large d. For example, the analogous calculation, gives then for the non-probabilistic method an error bound of

$$\int_0^1\cdots\int_0^1 f(t_1,\ldots,t_d)\,dt_1\ldots dt_d\in$$

$$\frac{1}{n^d}\sum_{i_1=1,\ldots,i_d=1}^{n,\ldots,n}f(z^1_{i_1},\ldots,z^d_{i_d})+[-Mn^{-2},Mn^{-2}].$$

This means that if we perform $N=n^d$ operations, our error is $MN^{-\frac{2}{d}}(=Mn^{-2})$. For large values of d these bounds become bad: the *curse of dimensionality*. Compare this with the error of $n^{-1/2}$ of the Monte-Carlo method. So if d is large, the Monte-Carlo method wins.

2.1 Error term

Let us be given a probability space $(\Omega, \mathcal{F}, P)$. In general the Monte-Carlo method evaluates $E(f)$, where f is some random variable on Ω, by

$$E(f) \approx E_n(f),$$

where

$$E_n(f) = \frac{1}{n}\sum_{i=1}^{n} f(x_i)$$

and x_i is a sequence of independent identically distributed with respect to P random samples.

The error term of the MC method is

$$\epsilon_n(f) = E(f) - E_n(f).$$

The Central Limit theorem tells us that the MC method is unbiased, i.e.

$$E(\epsilon_n(f)) = 0.$$

Moreover, for large values of n

$$\epsilon_n(f) \approx \frac{1}{\sqrt{n}}\sigma\nu,$$

where ν is $N(0,1)$ normally distributed random variable and σ^2 is the variance of f.

Given n samples, the empirical variance is

$$\begin{aligned} \tilde{\sigma}_n^2(f) &= E_n(f^2) - E_n(f)^2 & (1)\\ &= \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - E_n(f))^2. & (2) \end{aligned}$$

This is slightly biased estimator for $\sigma^2(f)$. The unbiased estimator is

$$\begin{aligned} \hat{\sigma}_n^2(f) &= \frac{1}{n-1}\sum_{i=1}^{n}(f(x_i) - E_n(f))^2 & (3)\\ &= \frac{n}{n-1}\tilde{\sigma}_n^2(f). & (4) \end{aligned}$$

3 Examples

3.1 European options

The most obvious way to compute the option price by Monte Carlo methods is to compute the distribution for the underlying price at the expiry date:

$$S(T) = S(0)\exp((r - \frac{1}{2}\sigma^2)T + \sigma\sqrt{T}\epsilon)$$

with $\epsilon \sim N(0,1)$.

The European call option price is the discounted payoff:

$$f(S) = \exp(-rT)\max(0, S - X),$$

where X is the strike price.

Thus, the European call option price is $E(f)$ where the expectation is taken with respect to the distribution of the underlying.

So one could simply compute the expected value of this expression as follows:

1. Compute n pseudo random $N(0,1)$ numbers. (Matlab function `randn` will do it for you.)
2. Compute $S(T)$ for each sample by formula above.
3. Apply the discounted payoff to $S(T)$.
4. Average the result.

3.2 Correlated Normal Random Variables

Let x is a vector of independent $N(0,1)$ variables, and define a new vector $y = Lx$ where L is a matrix.

Each element of y is Normally distributed,

$$E[y] = LE[x] = 0,$$

and

$$E[yy^T] = E[Lxx^T L^T] = LE[xx^T]L^T = LL^T$$

since $E[xx^T] = Id$ because elements of x are independent ($E[x_i x_j] = 0$ for $i \neq j$) and have unit variance ($E[x_i^2] = 1$).

Usually one has to solve opposite problem: suppose y is a vector of Normally distributed random variables with the given covariance matrix Σ; how one can generate pseudo-random variables y?

If we can decompose $\Sigma = LL^T$, the problem is solved because we know how to generate independent Normally distributed random variables.

The decomposition $\Sigma = LL^T$ is not unique but always exists because Σ is symmetric positive definite matrix. Simplest choice is to use a Cholesky factorization in which L is lower-triangular, with a positive diagonal.

3.3 Basket options

Basket option is a derivative exposed to two or more sources of uncertainty. The payoff of the European basket call option is given by

$$\max\left(0, \frac{1}{M}\sum_{i=1}^{M} S_i - X\right).$$

Assume that the assets prices S_i perform the usual geometrical Brownian motion:

$$dS_i = rS_i dt + \sigma_i S_i dW_i,$$

where different W_i might be highly correlated.

As before, at time T S_i is distributed as

$$S_i(T) = S_i(0) \exp((r - \frac{1}{2}\sigma_i^2)T + \sigma_i \sqrt{T}\epsilon_i)$$

with $\epsilon_i \sim N(0, 1)$.

If $\sigma_i W_i(T)$ have covariance matrix Σ, then use Cholesky factorization $LL^T = \Sigma$ to get

$$S_i(T) = S_i(0) \exp((r - \frac{1}{2}\sigma_i^2)T + \sum_{j=1}^{M} L_{ij}\epsilon'_j)$$

where ϵ'_j are independent $N(0, 1)$ random variables.

As before, the price of European basket call option is the expectation of the discounted payoff

$$f(S) = \exp(-rT) \max(0, \frac{1}{M}\sum_{i=1}^{M} S_i - X)$$

which can be written as the M–dimensional integral.

This is a good example for Monte Carlo simulation: the cost scales linearly with the number of stocks, whereas it would be exponential for grid-based numerical integration.