Introduction to Monte Carlo

Monte Carlo

Suppose that we want to estimate $\int_0^1 f(x)dx$ where $f(x)$ is a continuous function on $[0, 1]$

Let X be a random variable uniformly distributed on $[0, 1]$ and let us draw $x_1, .x_2, ...$ from X.

Can we approximate $\int_0^1 f(x)dx$ by using $x_1, x_2, ...$?

$$\int_0^1 f(x)dx = \int_0^1 f(x)g(x)dx$$

where $g(x)$ is the uniform density on $[0, 1]$.
And, therefore

$$\int_0^1 f(x)dx = E(f(X))$$

Monte Carlo

How can we estimate $E(f(X))$? Exactly by drawing $x_1, .x_2, ...$ and averaging the $f(x_i)s$.

So, this gives us a probabilistic method to approximate integrals.

How fast does it converge?

The variance of $\frac{1}{n}\sum_1^n f(x_i)$ is $\frac{\sigma_f^2}{n}$ so the standard deviation is $\frac{\sigma_f}{\sqrt{n}}$.

So, if we want to cut the error by two we need to increase n by a factor of 4.

If we were to use a numerical scheme, like a trapezoidal rule the convergence would be much faster.

However, convergence in numerical schemes slows down in more dimensions whereas in Monte Carlo it stays at the $\sqrt{n}$ rate.

Monte Carlo

When studying binomial trees we saw that the price of a call option can be found by taking an expected value:

$$C(S_0, K, T) = e^{-rT} \hat{E}(\max(S_T - K, 0))$$

where $\hat{E}$ represents the fact that we are taking probabilities with respect to the q's instead of the p's.

In that equation the random variable is S_T. In the next couple of weeks we will justify the fact that, under some conditions,

$$log(S_T) \sim N(log(S_0) + (r - \frac{\sigma^2}{2})T, \sigma\sqrt{T})$$

Therefore:

$$C(S_0, K, T) = \int_0^{\infty} \max(s - K, 0) f_{S_T}(s) ds$$

where f_{S_T} is the density of the random variable S_T.

So, as before, we can get a sample from S_T $(s_1, s_2, ..., s_n)$ and approximate:

$$C(S_0, K, T) \cong e^{-rT} (\frac{1}{n} \sum_{i=1}^{n} \max(s_i - K, 0))$$

Monte Carlo

To do this we need to be able to sample from a Normal variable.