Monte Carlo Methods in Option Pricing

UiO-STK4510
Autumn 2015

The Basics of Monte Carlo Method

- **Goal**: Estimate the expectation $\theta = \mathbb{E}[g(X)]$, where g is a measurable function and X is a random variable such that $g(X)$ is integrable.
- Let $\{X_i\}_{i=1,..,N}$ of i.i.d. random variables with law $\mathcal{L}(X)$. By the law of large numbers we have that

$$\tilde{\theta}_N \triangleq \frac{1}{N}\sum_{i=1}^{N} g(X_i) \underset{N\to\infty}{\longrightarrow} \theta,$$

  where the convergence may be a.s. (strong law of large numbers) or in probability (weak law of large numbers).
- If we assume in addition that $\mathbb{E}[|g(X)|^2] < \infty$ then by the central limit theorem we have that

$$\sqrt{N}\frac{\tilde{\theta}_N - \theta}{\mathrm{Var}[g(X)]} \underset{N\to\infty}{\overset{\mathcal{L}}{\longrightarrow}} \mathcal{N}(0,1).$$

The Basics of Monte Carlo Method

- Assume that we can generate $x_1, x_2, ..., x_N$ random numbers from the distribution X, then the Monte Carlo estimation of θ will be

$$\tilde{\theta}_N = \frac{1}{N} \sum_{i=1}^{N} g(x_i).$$

- From the central limit theorem we can construct the 95% confidence interval for θ

$$\left(\tilde{\theta}_N - 1.96 \frac{\mathrm{Var}[g(X)]}{\sqrt{N}}, \tilde{\theta}_N + 1.96 \frac{\mathrm{Var}[g(X)]}{\sqrt{N}} \right).$$

- $\mathrm{Var}[g(X)]$ is unknown, but can be estimated by

$$\hat{\sigma}^2_{N-1} = \frac{1}{N-1} \sum_{i=1}^{N} \left(g(x_i) - \tilde{\theta}_N \right)^2$$

The Basics of Monte Carlo Method

- Usually, the estimator $\hat{\sigma}^2_{N-1}$ converges fast to $\mathrm{Var}[g(X)]$.
- One can run a pilot simulation with less samples $N_p < N$ and use $\hat{\sigma}^2_{N_p-1}$ instead of $\mathrm{Var}[g(X)]$ to compute a confidence interval, i.e.,

$$\left(\tilde{\theta}_N - 1.96\frac{\hat{\sigma}^2_{N_p-1}}{\sqrt{N}}, \tilde{\theta}_N + 1.96\frac{\hat{\sigma}^2_{N_p-1}}{\sqrt{N}}\right).$$

- The important fact is that the rate of convergence of the method is $1/\sqrt{N}$.
- Variance reduction techniques: Note that

$$\mathrm{Var}[\tilde{\theta}_N] = \frac{1}{N}\mathrm{Var}[g(X)].$$

  There are modifications of the Monte Carlo estimator $\hat{\theta}_N$ that allow to reduce $\mathrm{Var}[\hat{\theta}_N]$ and get better confidence intervals using the same number of simulations.

The Basics of Monte Carlo Method

- However, these variance reduction techniques do not change the rate of convergence.
- Another important aspect is that the rate of convergence is independent of the dimension of the problem.
- As a rule of thumb when an expectation can be computed using numerical quadrature of integrals and this integrals are one dimensional, Monte Carlo methods perform worst than quadrature methods.
- If the dimension is high, Monte Carlo methods perform better than quadrature methods and it is usually simpler to implement.

Pricing Simple Contingent Claims

- Assume that we have a contingent claim of the form $H = h(S_T)$.
- By the risk-neutral pricing formula we get that

$$f(t,x) = e^{-r(T-t)}\mathbb{E}_Q[h(S_T^{t,x})],$$

where, under Q, $S^{t,x}$ is a geometric Brwonian motion with dirft $r - \frac{\sigma^2}{2}$, volatility σ and initial state $S_t^{t,x} = x$.

- Hence,

$$f(t,x) = e^{-r(T-t)}\mathbb{E}_Q\left[h\left(x\exp\left(\left(r - \frac{\sigma^2}{2}\right)(T-t) + \sigma(\tilde{W}_T - \tilde{W}_t)\right)\right)\right],$$

where $\tilde{W}$ is a Brownian motion under Q.

- Note that $\tilde{W}_T - \tilde{W}_t \sim \sqrt{T-t}Z$ where $Z \sim \mathcal{N}(0,1)$ under Q.

Pricing Simple Contingent Claims

- Therefore, the Monte Carlo algorithm for pricing the contingent claim is:

1. Draw N independent samples from a $Z \sim \mathcal{N}(0,1)$:

$$(z_1, ..., z_N).$$

2. Compute

$$e^{-r(T-t)} \frac{1}{N} \sum_{i=1}^{N} h\left(x \exp\left(\left(r - \frac{\sigma^2}{2}\right)(T-t) + \sigma\sqrt{T-t} z_i\right)\right)$$

- All statistical packages have implemented functions to generate random numbers from the most common distributions, in particular the normal distribution.
- If you use R or Matlab you can generate simultaneously vectors of samples from a standard normal distribution. This feature makes easy the *vectorization* of many simulation algorithms.
- Recall that these languages are interpreted and you must avoid the use of loops whenever possible.

Pricing Simple Contingent Claims

- Recall that using the density approach we can express the delta in the hedging strategy as an expectation

$$\frac{\partial f}{\partial x}(t,x) = e^{-r(T-t)}\mathbb{E}_Q[g(t,x,S_T^{t,x})],$$

where

$$g(t,x,s) = h(s)\frac{\log(s/x) - (r-\sigma^2/2)(T-t)}{x\sigma^2(T-t)}.$$

- Moreover,

$$g(t,x,S_T^{t,x}) = h(S_T^{t,x})\frac{\tilde{W}_T - \tilde{W}_t}{x\sigma^2(T-t)}$$

- Hence, to compute the delta we can use the Monte Carlo algorithm with a modified payoff.

Pricing Simple Contingent Claims

- An alternative approach is to use numerical differentiation.
- We can make the following approximation

$$\frac{\partial f}{\partial x}(t,x) \approx \frac{f(t,x+h)-f(t,x)}{h}.$$

- One can compute $f(t,x)$ and $f(t,x+h)$ using the Monte Carlo algorithm and then dividing the difference by h.
- Although it seems more work to run two times the Monte Carlo simulation, one can use the same random numbers to compute $f(t,x)$ and $f(t,x+h)$.
- This technique is called common random numbers and is one of the simplest methods to reduce the variance of the Monte Carlo estimate of $f(t,x+h)-f(t,x)$.
- Sometimes is used the symmetric difference

$$\frac{\partial f}{\partial x}(t,x) \approx \frac{f(t,x+h)-f(t,x-h)}{2h}.$$

Pricing of Path-Dependent Claims

- We consider the pricing of a knock-out call option, that is, a contingent claim with payoff

$$H = \max(0, S_T - K)\,\mathbf{1}_{\{S_t \le b : t \in [0,T]\}}.$$

- This contingent claim pays the same as a call option whenever the price process never exceeds the threshold b during the life of the claim. Note that $b > K$ for the contract to make sense.
- The price of this option depends on the whole path of the price process not only S_T.
- From the risk-neutral pricing formula we get that the price of a knock-out call option at time 0 is given by

$$\pi_0(H) = e^{-rT}\mathbb{E}_Q[\max(0, S_T - K)\,\mathbf{1}_{\{S_t \le b : t \in [0,T]\}}].$$

Pricing of Path-Dependent Claims

- In order to simulate a non-zero outcome from the payoff H we must check if $S_t \leq b$ for all $t \in [0, T]$.
- Of course this is impossible to check.
- What we do is to simulate the values of S_t is a fine partition $\{t_i\}_{i=0,\dots,M}$ of $[0, T]$ and check that $S_{t_i} \leq b$ for $i = 0, \dots, M$.
- This procedure introduces an error or bias that tends to zero as M tends to infinity.
- The idea is to simulate the discretized path recursively.
- Fix $M \in \mathbb{N}$ large and set $\delta = T/M$. Consider $\{t_j = j\delta\}_{j=0,\dots,M}$.
- Recall that

$$S_t = S_0 \exp\left(\left(r - \frac{\sigma^2}{2}\right) t + \sigma \tilde{W}_t\right),$$

  where $\tilde{W}$ is a Brownian motion under Q.

Pricing of Path-Dependent Claims

- We can write

$$
\begin{aligned}
S_{t_j} &= S_0 \exp\left(\left(r - \frac{\sigma^2}{2}\right) t_j + \sigma \tilde{W}_{t_j}\right) \\
&= S_0 \exp\left(\left(r - \frac{\sigma^2}{2}\right) (t_{j-1} + \delta) + \sigma \left(\tilde{W}_{t_{j-1}} + \tilde{W}_{t_j} - \tilde{W}_{t_{j-1}}\right)\right) \\
&= S_0 \exp\left(\left(r - \frac{\sigma^2}{2}\right) t_{j-1} + \sigma \tilde{W}_{t_{j-1}}\right) \\
&\quad \times \exp\left(\left(r - \frac{\sigma^2}{2}\right) \delta + \sigma \left(\tilde{W}_{t_j} - \tilde{W}_{t_{j-1}}\right)\right) \\
&= S_{t_{j-1}} \exp\left(\left(r - \frac{\sigma^2}{2}\right) \delta + \sigma \sqrt{\delta} Z_j\right),
\end{aligned}
$$

for $j = 1, \ldots, M$.

Pricing of Path-Dependent Claims

- The random variables $Z_j = \delta^{-1/2}\left(\tilde{W}_{t_j} - \tilde{W}_{t_{j-1}}\right)$ are distributed according to a $\mathcal{N}(0,1)$ and are independent of $S_{t_{j-1}}$.
- With this recursion formula is easy to use a Monte Carlo approach to simulate the path of S_t at the times $\{t_j\}_{j=0,\dots,M}$ in the partition.
- Of course it may happen that $S_t > b$ for some $t \in (t_j, t_{j+1})$ while $S_{t_j} \leq b$ and $S_{t_{j-1}} \leq b$. The probability that this happens tends to zero as we increase the points in the partion but there alway be a small bias.
- We simulate an outcome of H by simulating S_t at points $\{t_j\}_{j=0,\dots,M}$ while checking if the condition $S_{t_j} \leq b$ is fullfilled for all $j = 1, \dots, M$.. If this is the case the outcome is $\max(0, S_T - K)$, otherwise the outcome is zero.

Pricing of Path-Dependent Claims

The Monte Carlo algorithm for a Knock-Out call option.

1. For $k = 1, ..., N$
   1.1 For $j = 1, ..., M$
      - Draw one outcome z_j^k from $Z_j \sim \mathcal{N}(0, 1)$.
      - Compute
      $$s_j^k = s_{j-1}^k \exp\left(\left(r - \frac{\sigma^2}{2}\right)\delta + \sigma\sqrt{\delta} z_j^k\right).$$
      - If $s_j^k > b$, let $x^k = 0$ and return to 1.

   1.2 Let $x^k = \max\left(0, s_M^k - K\right)$.
2. Compute
$$e^{-rT}\frac{1}{N}\sum_{k=1}^{N} x^k.$$

References

- In Benth's book you will find:
  - Pricing contingent claims on many underlying stocks.
  - Pricing an Asian option

$$H = \max\left(0, \frac{1}{T}\int_0^T S_t dt - K\right).$$

- An excellent reference book for Monte Carlo methods in finance is
  - Glasserman, P. *Monte Carlo Methods in Financial Engineering*. Springer Verlag. 2004.