

Financial Mathematics 32000

Lecture 7

Roger Lee

2025 May 12

MC for American options

UNIT 4: Numerical Integration

Numerical integration in (log-)price space

Can MC be used for Americans?

The 1993 edition of a leading derivatives textbook:

“One limitation of the Monte Carlo simulation approach is that it can be used only for European-style derivative securities”

In later editions, this statement was removed.

Francis Longstaff and Eduardo Schwartz



Approximate American by Bermudan

Consider an American option exercisable at any time in $[0, T]$.

- ▶ Let Y_t be the discounted (to time 0) payoff for a time- t exercise. Assume Y is some given function of a Markov state variable (or vector) X . Define Y_T to incorporate the correct time- T decision. Example: for American put on X , let $Y_T := e^{-rT}(K - X_T)^+$ and let $Y_t = e^{-rt}(K - X_t)$ for $t < T$.
- ▶ Approximate the American put by allowing exercise only at a finite number of times:

$$0 = t_0 < t_1 < \cdots < t_N = T$$

So we approximate the American by a *Bermudan*.

[0, T]

Bermuda

T



Source: Google Earth.

“The island has a distinctive blend of British and American culture”

Bermudan facts

1. The time-0 value of the Bermudan is

$$\max_{\tau} \mathbb{E}Y_{\tau}$$

where τ ranges over stopping times taking values in $\{t_0, t_1, \dots, t_N\}$. So τ is a random exercise time that can depend on X , but only in a non-anticipating way.

2. Define $\tau_N := T$. Idea is that τ_n is the optimal exercise time given no exercise prior to t_n ; thus τ_n is a stopping time taking values in $\{t_n, t_{n+1}, \dots, t_N\}$. Given τ_{n+1} , define τ_n by

$$\tau_n := \mathbf{1}\left(Y_{t_n} \geq \mathbb{E}(Y_{\tau_{n+1}} | X_{t_n})\right) + \tau_{n+1} \mathbf{1}\left(Y_{t_n} < \mathbb{E}(Y_{\tau_{n+1}} | X_{t_n})\right).$$

Then τ_0 attains the max in 1.

Continuation value function

But we need to estimate the *continuation value function*

$$f_n(x) := \mathbb{E}(Y_{\tau_{n+1}} | X_{t_n} = x)$$

Longstaff-Schwartz: Use simulation and least-squares regression.

- ▶ Simulate M paths of X . Each path includes t_0, t_1, \dots, t_N .
- ▶ Define $\hat{\tau}_N := T$. On each path $m = 1, \dots, M$, given $\hat{\tau}_{n+1}^m$, define

$$\hat{\tau}_n^m := t_n \mathbf{1}(Y_{t_n}^m \geq \hat{f}_n(X_{t_n}^m)) + \hat{\tau}_{n+1}^m \mathbf{1}(Y_{t_n}^m < \hat{f}_n(X_{t_n}^m))$$

where \hat{f}_n is a “least-squares” estimate of continuation value.

If f_n known to be ≥ 0 , can use $\hat{f}_n(X_{t_n}^m)^+$ instead of $\hat{f}_n(X_{t_n}^m)$.

- ▶ Estimate of the time-0 option price is

$$\hat{C}_M^{\text{LS}} := \frac{1}{M} \sum_m Y_{\hat{\tau}_0^m}^m$$

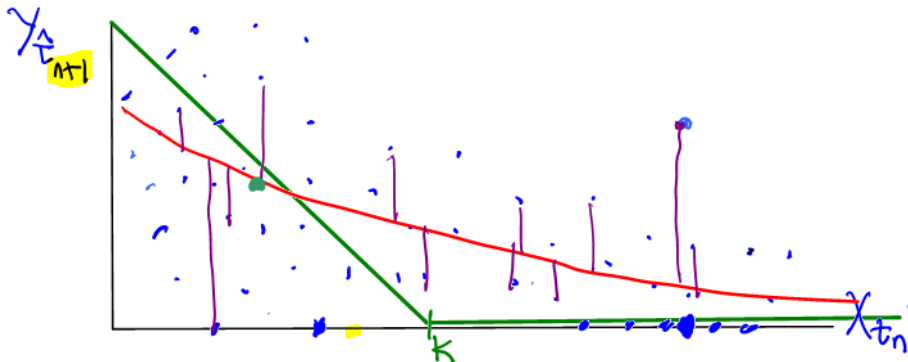
Estimated continuation value function

How to obtain the *estimated continuation value function* \hat{f}_n :

Each of the M paths of X generates a pair $(X_{t_n}^m, Y_{\hat{\tau}_{n+1}^m}^m)$.

Intuition: Consider a plot of all M pairs.

We want to extract the conditional expectation from the cloud.



Estimated continuation value function

- ▶ Choose $\hat{f}_n(x)$ as a linear combination of some set of B basis functions. (Example: with $B = 3$ and basis functions $1, x, x^2$ and coefficients $\hat{\beta}$, we have $\hat{f}_n(x) = \hat{\beta}_0 + \hat{\beta}_1 x + \hat{\beta}_2 x^2$)
- ▶ Choose coefficients $\hat{\beta}$ to minimize the sum of squared residuals

$$\sum_m \left(\hat{f}_n(X_{t_n}^m) - Y_{\hat{\tau}_{n+1}}^m \right)^2.$$

To implement this, run an ordinary least squares linear regression of the simulated $Y_{\hat{\tau}_{n+1}}$ on the basis functions (evaluated at the simulated X_{t_n}).

For $n = 0$, regression unnecessary; $\hat{f}_0(X_{t_0}^m)$ is just $\frac{1}{M} \sum_{m=1}^M Y_{\hat{\tau}_1}^m$.

- ▶ (Alternatives: Other basis functions? Non-OLS regression?)

Implementation

In Python: `sklearn.linear_model.LinearRegression` or

`statsmodels.api.OLS` or `scipy.linalg.lstsq(A,y)`

where A is the $M \times B$ matrix of regressors, and y is the length- M vector of simulations of continuation payoffs $Y_{\hat{\tau}_{n+1}}^m$. Example:

$$A = \begin{pmatrix} 1 & X_{t_n}^{(1)} & [X_{t_n}^{(1)}]^2 \\ 1 & X_{t_n}^{(2)} & [X_{t_n}^{(2)}]^2 \\ \vdots & \vdots & \vdots \\ 1 & X_{t_n}^{(M)} & [X_{t_n}^{(M)}]^2 \end{pmatrix} \quad y = \begin{pmatrix} Y_{\hat{\tau}_{n+1}}^{(1)} \\ Y_{\hat{\tau}_{n+1}}^{(2)} \\ \vdots \\ Y_{\hat{\tau}_{n+1}}^{(M)} \end{pmatrix}$$

In Excel: `LINEST(y,A)` but omit constant column from A

Also note that Excel outputs the coefficients in the *opposite* order.

Ignore at each time step the OTM data?

L-S: at each time step, include in the regression only the paths which are ITM at that time.

- ▶ We don't care about the continuation value function f_n at points X which are OTM, because we won't exercise there anyway.
- ▶ We care about getting a good estimate of f_n at points ITM.
- ▶ Discarding OTM points allows us to fit better the ITM points.

This can help especially if our parametric form does not capture well the behavior of f_n across the whole range of X .



But throwing away OTM observations also reduces sample size, magnifying the effect of the randomness in the ITM observations.

Bias

Let \hat{C}^{LS} be the MC estimate. Bias is by definition


$$\mathbb{E}\hat{C}^{\text{LS}} - C$$

where C be the true value of the American.

- ▶ Bias due to pricing Bermudan instead of American: 
- ▶ Bias in Bermudan price due to using the *same paths* to estimate optimal exercise policy and to price the Bermudan: 

Can eliminate by segregating paths: \hat{f} -estimation and pricing.

(Similar idea as *out-of-sample* testing and *cross-validation*)

- ▶ Bias in Bermudan price due to failure of basis functions to span the true continuation value function, and due to randomness in the \hat{f} -estimation simulations: 

MC for American options

UNIT 4: Numerical Integration

Numerical integration in (log-)price space

Numerical integration to calculate an expectation

Two approaches:

- ▶ Integrate discounted payoff \times density
- ▶ Integrate (Fourier transform of discounted payoff) \times
(Fourier transform of density)

MC for American options

UNIT 4: Numerical Integration

Numerical integration in (log-)price space

Integrate discounted payoff against density

Expectations of discounted payoffs are integrals

- ▶ If payoff is a function of some random X , then

$$\text{Price} = \int f(x)g(x)dx$$

where f is discounted payoff function and g is density of X .

- ▶ Example: if X is a stock price and payoff is a call option on stock price, then $f(x) = e^{-rT}(x - K)^+$, and g is density of stock price.
- ▶ Example: if X is log of a stock price and payoff is a call on stock price, then $f(x) = e^{-rT}(e^x - K)^+$, and g is density of $\log(\text{stock})$.

This is how we derived Black-Scholes in FINM 33000.

Integrate discounted payoff against density

The density function is known in special cases. For example if

$$dS_t = rS_t dt + \sigma S_t dW_t$$

then the time- t conditional density of $\log S_T$ is, for $-\infty < x < \infty$,

$$g_X(x) = \frac{1}{\sqrt{2\pi\sigma^2(T-t)}} \exp\left(-\frac{(x - (\log S_t + (r - \sigma^2/2)(T-t)))^2}{2\sigma^2(T-t)}\right)$$

and the the time- t conditional density of S_T is, for $0 < x < \infty$,

$$g_S(S) = \frac{1}{S\sqrt{2\pi\sigma^2(T-t)}} \exp\left(-\frac{(\log S - (\log S_t + (r - \sigma^2/2)(T-t)))^2}{2\sigma^2(T-t)}\right)$$

If fg can be calculated, then integral can too,

by numerical integration: `scipy.integrate` in Python.

When is this not recommended

When is numerical integration of payoff \times density not recommended:

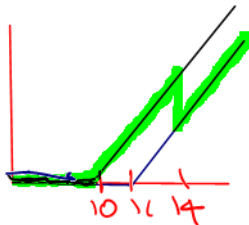
- ▶ When density function not known
- ▶ When easier methods available (for instance: g is lognormal and f is continuous piecewise linear in S)

Interview question (EY Complex Securities)

So given

Assume a 70% volatility and 2% risk-free rate. Find the time 0 price of an option which pays at time 1

$$\begin{cases} (S_1 - 11)^+ & \text{if } S_1 > 14 \text{ or } S_{0.5} > 12 \\ (S_1 - 10)^+ & \text{otherwise} \end{cases}$$



If $S_{0.5} > 12$ then payoff = $(S_1 - 11)^+$

$$e^{-r(1-0.5)} E_{0.5} \text{ payoff} = C^B(S=S_{0.5}, t=0.5, r=2\%, K=11, T=1)$$

If $S_{0.5} \leq 12$ then payoff = the green [redacted]

$$e^{-r(1-0.5)} E_{0.5}(\text{[redacted]}) = C^B(S=S_{0.5}, t=0.5, r=2\%, K=10, T=1) \\ = e^{-r(1-0.5)} N(d_2(K=11))$$

Payoff of
binary call

Interview question (EY Complex Securities)