Least Squares Approach in Valuing American Options

Project Report

Sai Prateek K	A Anish	Adha Satwik
ES20BTECH11024	ES20BTECH11001	ES19BTECH11020

March 5, 2023

As we are valuing American Options, we have the ability to exercise it at any point in time before the expiration date. So the optimal strategy would be to compare the immediate value if we exercise the option and the expected value of the stock from continuing, and then exercising the option if the immediate value is greater.

This approach uses the previous information to identify the conditional expectation function. We do this by regressing the future cash flows from continuation on a set of basis functions of the values of the relevant state variables . The fitted value of this regression is an efficient unbiased estimate of the conditional expectation function and allows us to estimate when to stop accurately.

There are four major parts to the paper:

1. Giving a numerical example
2. Explaining the Algorithm
3. Applying the algorithm
4. Problems and Implementation Issues in the Algorithm

There are three sections to explain the algorithm:

1. The valuation framework
2. The LSM algorithm
3. Convergence results

The paper then discusses the implementation of the algorithm in 5 situations:

1. Valuing American Put Options
2. Valuing an American-Bermuda-Asian Option
3. Valuing Cancelable Index Amortizing Swaps
4. Jump-Diffusions and American Option Valuation
5. Valuing Swaptions in a String Model

Finally, the paper discusses the numerical and implementation issues in the algorithm:

1. Higher-dimensional problems
2. Types of Least squares
3. Choice of basis Functions
4. Computational Speed of the algorithm

In this project, we will look at the implementation of the algorithm in valuing American put options.

1 A Numerical Example

Before the expiration date, the optimal strategy is to compare the immediate exercise value with the expected cash flows from continuing. At the final exercise date, the optimal exercise strategy for an American option is to exercise the option if we have a net gain. Thus, the key to optimally exercising an American option is identifying the conditional expected value of continuation. So, in this approach, we try to identify the conditional expectation function. We do this by regressing the subsequent realised cash flows from continuation on a set of basis functions of the values of the relevant state variables. The fitted value of this regression is an efficient unbiased estimate of the conditional expectation function and allows us to estimate the optimal stopping rule for the option accurately estimate the optimal stopping rule for the option.

Let us take a numerical example to explain this. Take an American put option with a strike price at times 1,2,3 and the risk-free interest rate is 6%. Let the stock prices at the different times be represented by the below matrix. We have $\mathrm{e}^{-0.06} = 0.94176$

Stock price paths

Path	$t = 0$	$t = 1$	$t = 2$	$t = 3$
1	1.00	1.09	1.08	1.34
2	1.00	1.16	1.26	1.54
3	1.00	1.22	1.07	1.03
4	1.00	.93	.97	.92
5	1.00	1.11	1.56	1.52
6	1.00	.76	.77	.90
7	1.00	.92	.84	1.01
8	1.00	.88	1.22	1.34

Conditional on not exercising the option before the final expiration date at time 3, the cash flows realized by the option holder from following the optimal strategy at time 3 are given below. We calculate this using the Put option payoff $= max(K - S_T, 0)$

Cash-flow matrix at time 3

Path	$t = 1$	$t = 2$	$t = 3$
1	—	—	.00
2	—	—	.00
3	—	—	.07
4	—	—	.18
5	—	—	.00
6	—	—	.20
7	—	—	.09
8	—	—	.00

We have five paths in which the put option is in the money at time t=2. Let X denote the stock prices at time 2 for these five paths and Y denote the corresponding discounted cash flows received at time 3 if the put is not exercised at time 2.

Regression at time 2

Path	Y	X
1	$.00 \times .94176$	1.08
2	—	—
3	$.07 \times .94176$	1.07
4	$.18 \times .94176$	.97
5	—	—
6	$.20 \times .94176$	.77
7	$.09 \times .94176$	.84
8	—	—

The resulting conditional expectation function is

$$E\left[\frac{Y}{X}\right] = -1.070 + 2.983X - 1.813X^2. \tag{1}$$

Then we take a matrix comparing the exercise value with the value that we get from the conditional expectation function

Optimal early exercise decision at time 2

Path	Exercise	Continuation
1	.02	.0369
2	—	—
3	.03	.0461
4	.13	.1176
5	—	—
6	.33	.1520
7	.26	.1565
8	—	—

So we can see that it is optimal to exercise the option at time 2 for the fourth, sixth, and seventh paths. This leads to the following matrix, which shows the cash flows received by the option holder conditional on not exercising prior to time 2.

Cash-flow matrix at time 2

Path	$t = 1$	$t = 2$	$t = 3$
1	—	.00	.00
2	—	.00	.00
3	—	.00	.07
4	—	.13	.00
5	—	.00	.00
6	—	.33	.00
7	—	.26	.00
8	—	.00	.00

When the option is exercised at time 2, the cash flow in the final column becomes zero. This is because once the option is exercised, there are no further cash flows since the option can only be exercised once. We continue recursively and do the same for time t = 1, and we get the following matrix.

Regression at time 1

Path	Y	X
1	$.00 \times .94176$	1.09
2	—	—
3	—	—
4	$.13 \times .94176$	.93
5	—	—
6	$.33 \times .94176$	.76
7	$.26 \times .94176$	.92
8	$.00 \times .94176$	.88

This time, we get the estimated conditional expectation function is

$$E\left[\frac{Y}{X}\right] = 2.038 - 3.335X - 1.356X^2. \tag{2}$$

By similarly comparing continuing value vs value when immediately exercised, we get

Optimal early exercise decision at time 1

Path	Exercise	Continuation
1	.01	.0139
2	—	—
3	—	—
4	.17	.1092
5	—	—
6	.34	.2866
7	.18	.1175
8	.22	.1533

Finally, we get the optimal stopping rule for each path as given below by the following matrix:

Stopping rule

Path	$t = 1$	$t = 2$	$t = 3$
1	0	0	0
2	0	0	0
3	0	0	1
4	1	0	0
5	0	0	0
6	1	0	0
7	1	0	0
8	1	0	0

The final cash flow matrix is given by

Option cash flow matrix

Path	$t = 1$	$t = 2$	$t = 3$
1	.00	.00	.00
2	.00	.00	.00
3	.00	.00	.07
4	.17	.00	.00
5	.00	.00	.00
6	.34	.00	.00
7	.18	.00	.00
8	.22	.00	.00

As shown by this example, the LSM approach is easily implemented since nothing more than simple regression is involved. After identifying the cash flows generated by the American put at each date along each path, the option can now be valued by discounting each cash flow in the option cash flow matrix back to time zero, and averaging over all paths.

2 Explaining the Algorithm

2.1 Valuation Framework

The objective of the LSM algorithm is to provide a pathwise approximation to the optimal stopping rule that maximizes the value of the American option. We use the algorithm by taking that the option can only be exercised at K discrete time intervals

$$0 < t_1 < t_2 < ..... < t_k = T$$

where T is the expiration date. As we know that the American Option can be exercised at any time, we take K to be a very large value. The value of the option is maximized pathwise, and hence unconditionally, if the investor exercises as soon as the immediate exercise value is greater than or equal to the value of continuation.

According to the No-arbitrage valuation theory, the value of the option assuming that it cannot be exercised until after t_k, is given by taking the expectation of the remaining discounted cash flows *C(w,s;*t_k*, T)* with respect to the risk-neutral pricing measure Q. At the time t_k, we express the value of continuation *F(w;*t_k*)* using the information in the set $\mathcal{F}_{t_k}$ at the time t_k. So now we know when to exercise by comparing the immediate exercise value with this conditional expectation, and then exercising as soon as the immediate exercise value is positive and greater than or equal to the conditional expectation.

2.2 The LSM Algorithm

The LSM approach uses least squares to approximate the conditional expectation function at $t_{k-1}, t_{k-2}, ..., t_1$. We work backwards because *C(w,s;*t_k*, T)* is defined recursively. Specifically, at time t_{k-1}, we assume that the unknown functional form of *F(w;*t_k*)* can be represented as a linear combination of a countable set of $\mathcal{F}_{t_k}$ - measurable basis functions.

To implement the LSM approach, we approximate *F(w;*t_k*)* using the first $M < \infty$ basis functions, and denote this approximation F_M*(w;*t_{k-1}*)* as the number N of (in-the-money) paths in the simulation goes to infinity.

Once the conditional expectation function at time t_{k-1}, is estimated, we can determine whether early exercise at time t_{k-1}, is optimal for an in-the-money A path w by comparing the immediate exercise value with $\dot{F}$*(w;*t_{k-1}*)* and repeating for each in-the-money path. Once the exercise decision is identified, the option cash flow paths *C(w,s;*t_{k-1}*, T)* can then be approximated. This process is recursed for t_{k-2} and is repeated until the exercise decisions at each exercise time along each path have been determined.

$$dS/S : asset\ return$$

$$dt : time\ unit\ (assume\ year)$$

$$\mu : expected\ return\ per\ year$$

$$\sigma : return\ volatility\ per\ year$$

$$S : asset\ price\ per\ unit$$

$$dz \sim N(0, dt) \sim \epsilon\sqrt{dt}$$

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

$$Assume\text{: } dS_t/S_t = \mu_t dt + \sigma_t dz_t$$

By Ito's lemma:

$$d \log S_t = (\mu_t - \sigma_t^2/2)dt + \sigma_t dz_t$$

$$\int_0^T d \log S_t = \int_0^T (\mu - \sigma^2/2)dt + \sigma dz_t$$

$$\log S_T - \log S_0 = (\mu_t - \sigma_t^2/2)T + \sigma\epsilon\sqrt{T}$$

$$S_T = S_0 \exp((\mu_t - \sigma_t^2/2)T + \sigma_t \epsilon\sqrt{T})$$

If we assume risk-neutral measure,

$$\mu_t = r_t - q_t$$

r is risk-free rate, and q is dividend yield per year

The American option is then valued by starting at time zero, moving forward along each path until the first stopping time occurs, discounting the resulting cash flow from exercise back to time zero, and then taking the average over all paths ω. Our goal is to simulate a lot of call option payoff, and then average it out, and then discount the average back to the present. This will give us the derivative price.The code snippets are given below.

```python
# Longstaff & Schwartz algorithm
import math
import numpy as np
from math import exp, log, sqrt

np.random.seed(123456)

# Inputs
S0 = 100.        # underlying asset price (S_0)
sigma = 0.20978 # volatility per year
r = 0.01         # risk free rate per year
div = 0.001      # dividend yield per year
T = 1.           # time to maturity
K = 110.         # exercise price

# simulation parameters input
I = 10**3        # Number of simulation (The larger, the more accurate)
M = 12           # number of discrete time intervals (The larger, the more accurate)
```

```python
dt = T/M               # length of time step
df = math.exp(-r*dt)   # discount factor
mu = r - div           # expected under risk neutral measure
return_series = (mu - sigma**2/2.)*dt + sigma*sqrt(dt)*np.random.standard_normal((M,I))
#print(return_series.shape)
S = S0*np.exp(np.cumsum(return_series,axis=0))
#print(S)
```

```python
# Value of option if it is excercised
h = np.maximum(S - K,0.)          # for put (K-S) and for call (S-K) should be done here
V = h[-1]                         # terminal value

# Let us compute the value of American option with backward induction
for t in range(M-1,0,-1):         # For all the paths we had created using return_series
  f  = np.polyfit(S[t],V*df,10)   # estimate continuation value using S[t] with order of 10,S[t] is the current stock price, V*df is discounted value of the option
  C = np.polyval(f,S[t])          # evaluate continuation value with stock price S[t](this gives us the value of american option if we don't excercise the option)
  V = np.where(h[t]>C,h[t],V*df)  # new value of the american option if h[t]>C then we excercise the option or else we'll hold on to it

V0 = df*np.average(V)
print("American call option value is", round(V0,4))
```

```python
V0 = df*np.average(V)
print("American call option value is", round(V0,4))
```

2.3 Convergence Results

We examine the theoretical convergence of the algorithm to the actual value *V(X)* of the American Option.
We give two propositions to discuss this:

1. For any finite choice of M, K, and vector $\theta \in R^{M*(K-1)}$ representing the coefficients for the M basis functions at each of the K -1 early exercise dates, let *LSM(ω; M, K)* denote the discounted cash flow resulting from following the LSM rule of exercising when the immediate exercise value is positive and greater than or equal to $F_M(\omega_t; t_k)$ as defined by θ. Then the following inequality holds almost surely:

$$V(X) \geq \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} LSM(\omega; M, K) \tag{3}$$

As The LSM algorithm results in a stopping rule for an American-style option, if we consider the most optimal stopping rule, it is greater than or equal to all other stopping rules, including the stopping rule implied by the LSM algorithm. This result is useful because it provides an objective criterion for convergence.

2. Assume that the value of an American option depends on a single state variable X with support on (0,∞)which follows a Markov process. Assume further that the option can only be exercised at times t, and t, and that the conditional expectation function F($\omega; t_1$) is continuous and

$$\begin{aligned} &\int_0^\infty e^{-X} F^2(\omega; t_1) dX < \infty, \\ &\int_0^\infty e^{-X} F_X^2(\omega; t_1) dX < \infty \end{aligned} \tag{4}$$

Then for any $\epsilon > 0$, there exists an $M < \infty$ such that

$$\lim_{N\to\infty} Pr\left[|V(X) - \frac{1}{N} \sum_{i=1}^{N} LSM(\omega_i; M, K)| > \epsilon \right] = 0 \tag{5}$$

Intuitively this result means that by selecting M large enough and letting N → ∞, the LSM algorithm results in a value for the American option within ϵ of the true value. The key to this result is that the convergence of $F_M(\omega;\ \mathrm{t}_1) \to F(\omega;\ \mathrm{t}_1)$ is uniform on (0, ∞) when the indicated integrability conditions are met. This bounds the maximum error in estimating the conditional expectation, which in turn, bounds the maximum pricing error.

3 Valuing American Stock Options

Let us assume that we are interested in pricing an American-style put option on a share of stock, where the risk-neutral stock price process follows the stochastic differential equation

$$dS = rSdt + \sigma sdZ, \tag{6}$$

and where r and G are constants, Z is a standard Brownian motion, and the stock does not pay dividends. Furthermore, assume that the option is exercisable 50 times per year at a strike price of K up to and including the final expiration date T of the option. (Bermuda exercise feature). Thus we regress discounted realized cash flows on a constant and three nonlinear functions of the stock price. Since we use linear regression to estimate the conditional expectation function, it is straightforward to add additional basis functions as explanatory variables in the regression if needed.

4 Numerical and Implementation Issues

4.1 Higher-Dimensional Problems

The numerical examples in the paper benchmark the performance of the LSM algorithm for several low-dimensional problems which can be solved by standard finite difference techniques. As an additional benchmark, we also investigate the performance of the algorithm for a higher dimensional problem studied by Broadie and Glasserman (1997c). In their article, they applied a stochastic mesh approach, and their algorithm took 20 hours to run. To get a similar confidence band, using the LSM approach, it took just 1 to 2 minutes to run using a similar processor.

4.2 Least squares

In this algorithm, we use ordinary least squares to estimate the conditional expectation function. In some cases, however, using other techniques, such as weighted least squares, maybe more efficient. generalized least squares, or even generalized method of moments (GMM) in estimating the conditional expectation function. In estimating the least squares regressions, it may be noted that the R2s from the regressions are often somewhat low. Low R2 are to be expected when unexpected cash flows are highly volatile. In general, since the LSM algorithm is based on conditional first moments rather than second moments, the R2s from the regression should have little impact on the quality of the LSM approximation to the American option value.

4.3 Choice of Basis Function

We see that the LSM algorithm's results change with a change in basis functions. Few basis functions are needed to closely approximate the conditional expectation function over the relevant range where early exercise may be optimal. While the results change based on the choice of basis functions, it is important to be aware of the numerical implications of choice. This choice of basis functions also has a statistical significance for each basis function in the regression. Some of the basis functions have a high correlation with each other, which leads to difficulties in estimating the individual regression coefficient(which is used in econometrics). This does not affect the LSM algorithm since LSM is focused on the fitted value of regression(which is unaffected by the correlation).

4.4 Computational Speed

For the LSM algorithm, there's only one constraint. This constraint is on parallel computation. That is the regression needs to use the cross-sectional information in the simulation. Regression takes a lot of time, but the bottleneck due to this involves little loss in computational efficiency. There are many ways in which regressions could be estimated using individual CPUs and then aggregated across CPUs to form a composite estimate of the conditional expectation function. The use of quasi-Monte-Carlo techniques in conjunction with the LSM algorithm may lead to significant improvements in computational speed and efficiency.

5 Conclusion

This approach presents a simple new technique for approximating the value of American-style options by Monte Carlo simulation. This approach is intuitive, accurate, easy to apply, and computationally efficient.