Project 08 — American Option Pricing (LSM)

Longstaff–Schwartz Monte Carlo with early exercise

Quant Finance Portfolio (Karim)

December 19, 2025

Abstract

This report supports **Project 08**, a pricing engine for **American-style options** using the **Longstaff–Schwartz Method (LSM)**. The project implements risk-neutral Monte Carlo simulation, learns a continuation value via regression on basis functions, and applies an optimal early-exercise rule. It highlights recruiter-relevant aspects: correct dynamic programming logic, stable numerical design, and validation against benchmarks (European limit, binomial tree, monotonicity and bounds).

Contents

1 What you build in Project 08

American options allow exercise at multiple dates, which creates an **optimal stopping** problem. This project implements a practical Monte Carlo solution:

- Simulate risk-neutral paths for the underlying (GBM baseline).
- Price **European** options as a baseline Monte Carlo check.
- Implement **LSM**: backward induction + regression to approximate continuation values.
- Apply an **early exercise policy**: exercise if immediate payoff exceeds continuation value.
- Report stable outputs: price, exercise boundary intuition, sensitivity to basis choice, and diagnostics.
- Provide interactive visualization (exercise regions, value convergence vs paths/time steps).

Recruiter takeaway. This project shows you can implement a core sell-side quant technique for early-exercise derivatives: dynamic programming + regression, with correct risk-neutral discounting and robust validation.

2 Prerequisites (math and pricing concepts)

2.1 Risk-neutral pricing and discounting

Under a risk-neutral measure $\mathbb{Q}$ (no-arbitrage assumption), discounted prices are martingales. For a payoff H at maturity T,

$$V_0 = \mathbb{E}^{\mathbb{Q}}\left[e^{-rT} H\right].$$

2.2 Optimal stopping and American options

An American option value is the supremum over stopping times τ taking values in exercise dates:

$$V_0 = \sup_{\tau \in \mathcal{T}} \mathbb{E}^{\mathbb{Q}}\left[e^{-r\tau}\, \Phi(S_\tau)\right],$$

where Φ is the payoff (e.g. put: $(K-S)^+$).

2.3 Dynamic programming (Bellman principle)

With discrete exercise dates $t_0 < \cdots < t_N$,

$$V(t_i, S_{t_i}) = \max\left(\Phi(S_{t_i}),\ \text{Continuation}(t_i, S_{t_i})\right),$$

where continuation is the expected discounted value of holding the option:

$$\text{Continuation}(t_i, S_{t_i}) = \mathbb{E}^{\mathbb{Q}}\left[e^{-r\Delta t}\, V(t_{i+1}, S_{t_{i+1}}) \mid \mathcal{F}_{t_i}\right].$$

LSM approximates this conditional expectation by regression.

2.4 Regression basics (least squares)

Given data pairs (x_j, y_j), least squares fits coefficients β minimizing

$$\sum_j (y_j - \sum_k \beta_k \psi_k(x_j))^2,$$

where ψ_k are basis functions (polynomials, Laguerre functions, etc.).

3 Model setup: risk-neutral Monte Carlo

3.1 Underlying dynamics (GBM baseline)

Under $\mathbb{Q}$:

$$\frac{\mathrm{d}S_t}{S_t} = r\ \mathrm{d}t + \sigma\ \mathrm{d}W_t,$$

and in discrete time with step Δt:

$$S_{t_{i+1}} = S_{t_i} \exp\Big((r - \tfrac{1}{2}\sigma^2)\Delta t + \sigma\sqrt{\Delta t}\, Z_i\Big), \quad Z_i \sim \mathcal{N}(0,1).$$

3.2 Payoffs

Call: $\Phi(S) = (S-K)^+$, $\qquad$ Put: $\Phi(S) = (K-S)^+$.

Note. Early exercise is mainly relevant for **puts** (and for calls with dividends). For a non-dividend-paying stock, American call $\approx$ European call.

4 Longstaff–Schwartz Method (LSM)

4.1 Core idea

At each time t_i, for paths that are *in the money*, estimate the continuation value

$$C_i(S_{t_i}) \approx \mathbb{E}^{\mathbb{Q}}\Big[e^{-r\Delta t}V_{i+1} \mid S_{t_i}\Big]$$

by regressing discounted future cashflows on basis functions of S_{t_i}.

4.2 Algorithm (discrete exercise dates)

Assume we simulate M paths and store $S_i^{(m)}$.

1. Initialize cashflows at maturity: $V_N^{(m)} = \Phi(S_N^{(m)})$.
2. For $i = N-1, \dots, 1$ (backward):
   - Restrict to in-the-money paths: $\Phi(S_i^{(m)}) > 0$.
   - Define regression targets: $Y^{(m)} = e^{-r\Delta t}V_{i+1}^{(m)}$ (discounted continuation from next step).
   - Fit $\hat{C}_i(\cdot)$ via least squares on basis functions $\psi_k(S_i)$.
   - Exercise rule: set
   $$V_i^{(m)} = \begin{cases} \Phi(S_i^{(m)}) & \text{if } \Phi(S_i^{(m)}) \geq \hat{C}_i(S_i^{(m)}) \\ Y^{(m)} & \text{otherwise.} \end{cases}$$
   - For exercised paths, future cashflows are set to zero (exercise happens once).
3. Price is the average discounted cashflow at t_0:

$$V_0 \approx \frac{1}{M}\sum_{m=1}^{M} e^{-rt_{\tau(m)}}\, \Phi(S_{\tau(m)}^{(m)}).$$

4.3 Choice of basis functions

Common basis choices include:

- polynomials in S (e.g. $1, S, S^2$),
- polynomials in normalized S/K,
- Laguerre polynomials (classic LSM choice).

Too few basis functions underfit continuation; too many can overfit and destabilize exercise decisions.

5 Validation and benchmarks

Recruiter-grade pricing requires checks beyond "it runs".

5.1 Bounds and monotonicity

For a put:

$$\max(K - S_0, 0) \le V_0^A \le K,$$

and V_0^A increases with K and with σ, decreases with r.

5.2 European limit and binomial benchmark

- For calls without dividends: $V_0^A \approx V_0^E$.
- Compare an American put price from LSM with a **binomial tree** price for a sanity benchmark.

5.3 Convergence diagnostics

- Increase number of paths M and observe price stabilization.
- Increase time steps N (more exercise opportunities) and observe controlled changes.
- Track the exercise frequency and ensure it is economically plausible (not exercising deep OTM).

6 Implementation notes (what the notebook is doing)

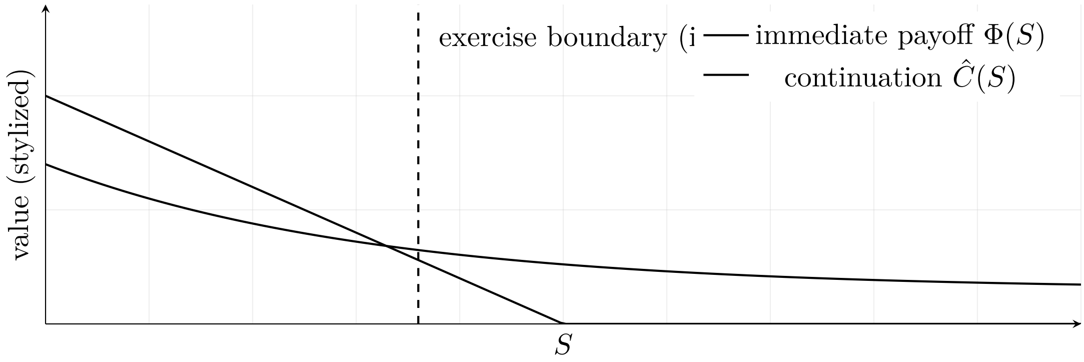
- Simulate GBM paths under $\mathbb{Q}$ with reproducible random seeds.
- Compute European MC prices as baseline.
- Implement LSM: backward loop, in-the-money filter, regression, exercise decision, cashflow bookkeeping.
- Provide interactive visuals: sample paths, exercise region heatmap, price vs #paths/#steps, basis sensitivity.

7 Sanity checks you should always do

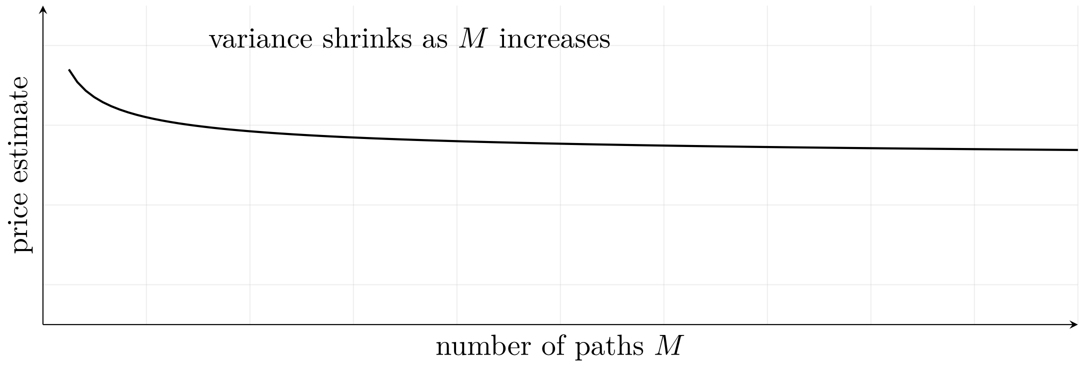
- **American call (no dividends):** LSM price should be close to European price.
- **Put exercise:** deep ITM puts may exercise early; deep OTM should almost never exercise.
- **Monotonicity:** increasing σ should not decrease the price.
- **Regression stability:** changing the basis degree should not cause wild, non-economic exercise behavior.
- **Reproducibility:** fixed seed gives stable results for the same parameters.

8 Overleaf plots (conceptual, fast to compile)

8.1 Early exercise rule (stylized)



8.2 Monte Carlo convergence intuition (stylized)



9 Interview pitch

I implemented American option pricing using the Longstaff–Schwartz Monte Carlo method. I simulated risk-neutral paths, performed backward induction, and approximated continuation values with least-squares regression on basis functions, then applied an optimal early-exercise rule. I validated the engine with bounds, European-call consistency, and comparisons against a binomial tree benchmark, and I added diagnostics on convergence and basis sensitivity.

LSM can be sensitive to basis choice and sample size; that's why I include stability checks and convergence diagnostics. For higher dimensions, the method is attractive because it scales better than trees, but regression design becomes even more important.