Deep Optimal Stopping & Pricing of American-Style Options

Applied Quantitative Finance Seminar

Sabina Georgescu, Konrad Müller

Overview

1. Introduction to American Options
2. Least-Squares Monte Carlo
3. Deep LSM
4. Deep Optimal Stopping
5. Results

1. Introduction to American Options

European, Bermudan, and American Options

- European Option: Can be exercised at a **single** date
- Bermudan Option: Can be exercised at **several** dates
- American Option: Can be exercised at **any** point in time before expiration

Bermudan Max-Call

- **Bermudan**: finite number of exercise dates $0 = t_0, t_1, \dots, t_N = T$
- **Max-Call**: Payoff at time t_n given by the maximum payoff of call options on d assets:

$$\max_{i=1,\dots,d} \left(S^i_{t_n} - K\right)^+$$

Pricing & Dynamics

- To price a Bermudan Max-Call must make assumption about the risk-neutral dynamics of the d underlying assets
- Simple choice: d uncorrelated exponential Brownian motions's with identical parameters, i.e., $\forall i = 1, ..., d$

$$dS_t^i = S_0^i \exp\left(\left[r - \delta - \frac{\sigma^2}{2}\right] dt + \sigma dW_t^i\right)$$

2. Least-Squares Monte Carlo

Recursive Formulation (Dynamic Programming Principle) / Continuation Values / ... SABINA

Approximating the Conditional Expectation SABINA

- Cond expecatation minimizes MSE over all Borel measureble functions
- Cannot optimize over that function space -> instead need set of basis functions ...

Choice of Basis Functions KONRAD

- What features to use as a regression basis?
- For Bermudan Max-Call ($d = 5$):
  - Longstaff-Schwartz (2001):
    - first five Hermite polynomials in the value of the most expensive asset
    - the value and its square of the other 4 assets
    - selected products between the individual asset prices
  - Broadie, Cao (2008): 12 - 18 polynonmial basis functions up to 5^{th} order

LSM Algorithm Pseudo-Code KONRAD

```
def lsm(paths):

    payoff_at_stop = payoff_fn(N, paths[:, -1])

    for n in np.arange(start=n_steps, stop=0, step=-1):

        x_n = paths[:, n]
        payoff_now = payoff_fn(n, x_n)

        features = feature_map(x_n, payoff_now)
        model = LinearRegression()
        model.fit(features, payoff_at_stop)
        continuation_values = model.predict(features)

        idx = payoff_now >= continuation_values
        payoff_at_stop[idx] = payoff_now[idx]

    biased_price = payoff_at_stop.mean()
```

Issues with Feature Engineering KONRAD

- Feature maps not expressive enough for precise pricing
- Potentially over-engineered to the specifics of the payoff / market dynamics
- Feature maps grow "only" linearly in d, but quickly enough to make them infeasible for large d or N

Solution: Learn the feature maps $\longrightarrow$ Deep LSM

3. Deep LSM

Lower Bound, Upper Bound, Point Estimate ... SABINA

- I would say we only talk here about the upper bound and leave it out in the LSM part

4. Deep Optimal Stopping

Motivation

- DLSM: Uses neural networks to approximate continuation values that are then compared to the immediate payoff of the option
- The immediate payoff is already a feature for the neural network

$\longrightarrow$ Let the neural network make this comparison step inherently, i.e., parametrize the stopping decision directly (**Deep Optimal Stopping**)

Parametrizing the stopping decision

$$\tau_{n+1} = \sum_{m=n+1}^{N} \left[m \cdot f^{\theta_m}(X_m) \cdot \prod_{j=n+1}^{m-1} \left(1 - f^{\theta_j}(X_j)\right) \right]$$

Parameter Optimization

- Reward (Loss) calculated using the **soft** stopping decision

$$r_n^k(\theta) = g(n, x_n^k) \cdot F^\theta(x_n^k) + g(l_{n+1}^k, x_{l_{n+1}^k}^k) \cdot (1 - F^\theta(x_n^k))$$

5. Results

Results LSM

Features	d	S_0	P_{biased}	s.e.	L	U	Point Est.	95 % CI
base	5	90	16.33	0.03	16.54	16.89	16.72	[16.30, 16.93]
ls	5	90	16.56	0.04	16.41	16.57	16.49	[16.17, 16.59]
r-NN	5	90	16.63	0.04	16.56	16.78	16.67	[16.33, 16.81]
base	5	100	25.71	0.06	25.72	26.28	26.00	[25.43, 26.32]
ls	5	100	26.01	0.02	26.03	26.21	26.12	[25.75, 26.24]
r-NN	5	100	25.94	0.04	26.00	26.32	26.16	[25.72, 26.35]

Results DLSM

d	S_0	L	U	Point Est.	95 % CI
5	90	16.62	16.66	16.64	[16.60, 16.66]
5	100	26.10	26.18	26.14	[26.09, 26.20]
5	110	36.69	36.79	36.74	[36.67, 36.81]
10	90	26.05	26.33	26.19	[26.03, 26.37]
10	100	38.09	38.40	38.24	[38.06, 38.43]
10	110	50.60	50.99	50.80	[50.57, 51.03]

Results DOS

d	S_0	L	U	Point Est.	95 % CI
5	90	16.60	16.65	16.62	[16.59, 16.66]
5	100	26.11	26.18	26.15	[26.10, 26.19]
5	110	37.78	37.86	37.82	[37.76, 37.88]
10	90	26.14	26.30	26.22	[26.13, 26.33]
10	100	38.23	38.45	38.34	[38.21, 38.49]
10	110	50.95	51.17	51.06	[50.93, 51.22]

Comparison to the Literature

d	S_0	DOS	DLSM	Literature
5	90	16.62	16.64	16.64
5	100	26.15	26.14	26.15
5	110	37.82	36.74	36.78
10	90	26.22	26.19	26.28
10	100	38.34	38.24	38.37
10	110	51.06	50.80	50.90

References

Literature

- Valuing American Options by Simulation:
  A Simple Least-Squares Approach
- Deep optimal stopping
- Pricing and Hedging American-Style Options with Deep Learning
- Optimal Stopping via Randomized Neural Networks

Code

https://github.com/konmue/american_options

Q&A