Machine learning regression for American options: neural networks and random forests

Jérôme Lelong (joint work with Zineb El Filali Ech-chafiq, Pierre Henry Labordère and Bernard Lapeyre)

Univ. Grenoble Alpes, LJK

LMM, Le Mans Université
15 mars 2022

A gambling problem

Consider a gambling game such that at each step n you are offered a gain Z_n.

- you can either take it and leave the game,
- or turn down your potential gain and keep playing.

Question: How to maximize your gain if you know the distribution of $(Z_n)_n$?

Our framework

- Finite time horizon N
- For all $n \in \{0, \dots, N\}$, $Z_n = \phi_n(X_n)$ where X is Markov chain in $\mathbb{R}^d$ and $\phi_n : \mathbb{R}^d \to \mathbb{R}_+$.

The above question is equivalent to finding $\tau^\star$ maximizing

$$\sup_{\tau \in \mathcal{T}_0} \mathbb{E}[Z_\tau | X_0].$$

A gambling problem

For $k \in \{0, \dots, N\}$, define

$$U_k = \sup_{\tau \in \mathcal{T}_k} \mathbb{E}[Z_\tau | X_k]$$

where $\mathcal{T}_k$ is the set of all stopping times with values in $\{k, k+1, \dots N\}$.
The Snell enveloppe theory yields that U solves

$$\begin{cases} U_N = Z_N \\ U_k = \max\left(Z_k, \mathbb{E}[U_{k+1} | X_k]\right) \end{cases} \tag{1}$$

We are more interested in the stopping rule.

A gambling problem

Let τ_k be the smallest optimal stopping time after k.

$$\begin{cases} \tau_N = N \\ \tau_k = k\mathbf{1}_{\{Z_k \geq \mathbb{E}[U_{k+1}|X_k]\}} + \tau_{k+1}\mathbf{1}_{\{Z_k < \mathbb{E}[U_{k+1}|X_k]\}}. \end{cases}$$

Note that $\mathbb{E}[U_{k+1}|X_k] = \mathbb{E}[\mathbb{E}[Z_{k+1}|X_{k+1}]|X_k] = \mathbb{E}[Z_{\tau_{k+1}}|X_k]$.

$$\begin{cases} \tau_N = N \\ \tau_k = k\mathbf{1}_{\left\{Z_k \geq \mathbb{E}[Z_{\tau_{k+1}}|X_k]\right\}} + \tau_{k+1}\mathbf{1}_{\left\{Z_k < \mathbb{E}[Z_{\tau_{k+1}}|X_k]\right\}}. \end{cases} \tag{2}$$

Question: How to compute the conditional expectation?
Assume square integrability, any solution to

$$\inf_{\psi \in L^2(\mathcal{L}(X_k))} \mathbb{E}\left[\left|Z_{\tau_{k+1}} - \psi(X_k)\right|^2\right]$$

satisfies $\mathbb{E}[U_{k+1}|X_k] = \psi(X_k)$.

American options

- The above gambling game is an American option.
- The process Z is the discounted payoff process, X is typically the underlying asset.
- We are equally interested in τ_0 and U_0.
- U_0 can be deduced from τ_0 by using Monte Carlo.
- The least square problem

$$\inf_{\psi \in L^2(\mathcal{L}(X_k))} \mathbb{E}\left[\left|Z_{\tau_{k+1}} - \psi(X_k)\right|^2\right]$$

  is solved by approximating the space L^2 by a finite dimensional vector space (polynomials, local functions, ...)
- We investigate two alternative approximations: neural networks and random forest

Some related works

- Some previous works using NN for optimal stopping
  - Michael Kohler, Adam Krzyżak, and Nebojsa Todorovic. Pricing of high-dimensional american options by neural networks. *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics*, 20(3):383–410, 2010
  - S. Becker, P. Cheridito, and A. Jentzen. Deep optimal stopping. *Journal of Machine Learning Research*, 20(74):1–25, 2019a
  - Sebastian Becker, Patrick Cheridito, Arnulf Jentzen, and Timo Welti. Solving high-dimensional optimal stopping problems using deep learning, 2019b

LS: truncation step

Longstaff-Schwartz type algorithms rely on direct approximation of *stopping times* and use of *the same sample paths* for all time steps (large computational savings).

- $(g_k, k \geq 1)$ is an $L^2(\mathcal{L}(X))$ basis and $\Phi_p(X,\theta) = \sum_{k=1}^{p} \theta_k \, g_k(X)$.
- Backward approximation of iteration policy using (2),

$$\begin{cases} \widehat{\tau}_N^{p,} = N \\ \widehat{\tau}_n^p = n \, \mathbf{1}_{\left\{Z_n \geq \Phi_p(X_n;\widehat{\theta}_n^p)\right\}} + \widehat{\tau}_{n+1}^p \, \mathbf{1}_{\left\{Z_n < \Phi_p(X_n;\widehat{\theta}_n^p)\right\}} \end{cases}$$

- with conditional expectation computed using a Monte Carlo minimization problem: $\widehat{\theta}_n^p$ is a minimizer of

$$\inf_{\theta} \mathbb{E}\left(\left|\Phi_p(X_n;\theta) - Z_{\widehat{\tau}_{n+1}^p}\right|^2\right).$$

- Price approximation: $U_0^p = \max\left(Z_0, \mathbb{E}\left(Z_{\widehat{\tau}_1^p}\right)\right)$.

The LS algorithm

- $(g_k, k \geq 1)$ is an $L^2(\mathcal{L}(X))$ basis and $\Phi_p(X,\theta) = \sum_{k=1}^p \theta_k \, g_k(X)$.
- Paths $X_0^{(m)}, X_1^{(m)}, \ldots, X_N^{(m)}$ and payoff paths $Z_0^{(m)}, Z_1^{(m)}, \ldots, Z_N^{(m)}$, $m = 1, \ldots, M$.
- Backward approximation of iteration policy,

$$\begin{cases} \widehat{\tau}_N^{p,(m)} = N \\ \widehat{\tau}_n^{p,(m)} = n\mathbf{1}_{\left\{Z_n^{(m)} \geq \Phi_p(X_n^{(m)};\widehat{\theta}_n^{p,M})\right\}} + \widehat{\tau}_{n+1}^{p,(m)}\mathbf{1}_{\left\{Z_n^{(m)} < \Phi_p(X_n^{(m)};\widehat{\theta}_n^{p,M})\right\}} \end{cases}$$

- with conditional expectation computed using a Monte Carlo minimization problem: $\widehat{\theta}_n^{p,M}$ is a minimizer of

$$\inf_\theta \frac{1}{M} \sum_{m=1}^M \left| \Phi_p(X_n^{(m)};\theta) - Z^{(m)}_{\tau_{n+1}^{p,(m)}} \right|^2 .$$

- Price approximation: $U_0^{p,M} = \frac{1}{M} \sum_{m=1}^M Z^{(m)}_{\widehat{\tau}_0^{p,(m)}}$.

Some related works

- Description of the algorithm: F.A. Longstaff and R.S. Schwartz.

  Valuing American options by simulation : A simple least-square approach.
  Review of Financial Studies, 14:113–147, 2001.

- Rigorous approach: Emmanuelle Clément, Damien Lamberton, and

  Philip Protter. An analysis of a least squares regression method for american option pricing.
  Finance and Stochastics, 6(4):449–471, 2002.

  - U_0^p converge to U_0, $p \to +\infty$
  - $U_0^{p,M}$ converge to U_0^p, $M \to +\infty$ a.s.
  - A central limit theorem but the limiting variance is unknown.

The modified algorithm

- In LS algorithm replace the approximation on a Hilbert basis $\Phi_p(.;\theta)$ by a neural network or a random forest. This is not a vector space approximation (non linear).
- The linear structure played a key role in Clément et al. [2002]'s analysis.
- The neural network is obtained as the solution of a non convex optimization problem.
- A random tree does not even write as the solution of a minimization problem.
- Our aim: extending the proof of (a.s.) convergence results and performing some numerical tests.

The result

Theorem 1

Under good assumptions

- *Convergence of the NN or RF approximation*

$$\lim_{p\to\infty} \mathbb{E}[Z_{\tau_n^p}|\mathcal{F}_n] = \mathbb{E}[Z_{\tau_n}|\mathcal{F}_n] \text{ in } L^2(\Omega) \quad (i.e.\ U_0^p \to U_0).$$

- *SLLN: for every $k = 1, \dots, N$,*

$$\lim_{M\to\infty} \frac{1}{M} \sum_{m=1}^{M} Z^{(m)}_{\widehat{\tau}_k^{p,(m)}} = \mathbb{E}\left[Z_{\tau_k^p}\right] \quad a.s. \quad (i.e.\ U_0^{p,M} \to U_0^p)$$

A quick view of Neural Networks

- In short, a NN is a non-linear mapping $x \to \Phi_p(x, \theta) \in \mathbb{R}$, with $\theta \in \mathbb{R}^d$, d large
- $\Phi_p = A_L \circ \sigma_a \circ A_{L-1} \circ \cdots \circ \sigma_a \circ A_1$, $L \geq 2$
- $A_l(x_l) = w_l x_l + \beta_l$ (affine functions)
- $L - 2$ "number of hidden layers"
- p "maximum number of neurons per layer" (i.e. sizes of the w_l matrix)
- σ_a a fixed non linear (called *activation function*) applied component wise
- $\theta := (w_l, \beta_l)_{l=1,\ldots,L}$ parameters of all the layers
- Restrict to a compact set $\Theta_p = \{\theta : |\theta| \leq \gamma_p\}$ and assume $\lim_{p\to\infty} \gamma_p = \infty$. $\to$ use the USLLN.
- $\mathcal{NN}_p = \{\Phi_p(\cdot, \theta) : \theta \in \Theta_p\}$ and $\mathcal{NN}_\infty = \cup_{p\in\mathbb{N}} \mathcal{NN}_p$

Hypotheses

- For every p, there exists $q \geq 1$

$$\forall \theta \in \Theta_p, \quad |\Phi_p(x,\theta)| \leq \kappa_q(1+|x|^q)$$

  a.s. the random function $\theta \in \Theta_p \longmapsto \Phi_p(X_n,\theta)$ are continuous.
- $\mathbb{E}[|X_n|^{2q}] < \infty$ for all $0 \leq n \leq N$.
- For all p, $n < N$, $\mathbb{P}\left(Z_n = \Phi_p(X_n;\theta_n^p)\right) = 0$.
- If θ_1 and θ_2 solve

$$\inf_{\theta \in \Theta_p} \mathbb{E}\left(\left|\Phi_p(X_n;\theta) - Z_{\widehat{\tau}_{n+1}^p}\right|^2\right),$$

  then $\Phi_p(x,\theta_1) = \Phi_p(x,\theta_2)$ for almost all x
  No need for a unique minimizer but only of the represented function.

Convergence of the NN approximation

A simple consequence of Hornik [1991], also known as the "Universal Approximation Theorem".

Theorem 2 (Hornik)

Assume that the function σ_a is non constant and bounded. Let μ denote a probability measure on $\mathbb{R}^r$, then $\mathcal{NN}_\infty$ is dense in $L^2(\mathbb{R}^r, \mu)$.

Corollary 3

If for every p, $\theta_p^* \in \Theta_p$ is a minimizer of

$$\inf_{\theta \in \Theta_p} \mathbb{E}[|\Phi_p(X;\theta) - Y|^2],$$

$(\Phi_p(X;\theta_p^*))_p$ converges to $\mathbb{E}[Y|X]$ in $L^2(\Omega)$ when $p \to \infty$.

Convergence of the Monte-Carlo approximation

- p is fixed, $N \to +\infty$
- Now, minimisation problems are non linear, need more abstract arguments to prove convergence
- Two ingredients (quite "old" results)
- First result: approximation of minimization problems

Lemma 4 (Rubinstein and Shapiro [1993])

- *$(f_n)_n$ defined on a compact set $K \subset \mathbb{R}^d$. $v_n = \inf_{x \in K} f_n(x)$*
- *x_n a sequence of minimizers $f_n(x_n) = \inf_{x \in K} f_n(x)$.*
- *$v^\star = \inf_{x \in K} f(x)$ and $\mathcal{S}^\star = \{x \in K : f(x) = v^\star\}$.*

If $(f_n)_n$ converges uniformly on K to a continuous function f, then $v_n \to v^\star$ and $d(x_n, \mathcal{S}^\star) \to 0$ a.s.

Convergence of the Monte-Carlo approximation

- Second result: SLLN in Banach spaces (Ledoux and Talagrand [1991], goes back to Mourier [1953]).

Lemma 5

Let $(\xi_i)_{i\geq 1}$ *i.i.d.* $\mathbb{R}^m$*-valued,* $h : \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}$*. If*

- *a.s.,* $\theta \in \mathbb{R}^d \mapsto h(\theta, \xi_1)$ *is continuous,*
- $\forall K > 0,\ \mathbb{E}\left[\sup_{|\theta|\leq K} |h(\theta, \xi_1)|\right] < +\infty.$

Then

$$\lim_{n\to\infty} \sup_{|\theta|\leq K} \left| \frac{1}{n} \sum_{i=1}^{n} h(\theta, \xi_i) - \mathbb{E}[h(\theta, \xi_1)] \right| = 0 \quad a.s.$$

Implementation details

- Python code with TensorFlow.
- We use ADAM algorithm to fit the neural network at each time step.
- We use the same NN through all time steps: take $\widehat{\theta}^{p,M}_{n+1}$, as the starting point of the training algorithm at time time n.
- No use of setting *epochs*> 1 for $n < N - 1$. This allows for huge computational time savings.

Regression trees

Consider some samples $(Z_m, X_m)_{1 \leq m \leq M}$ values in $\mathbb{R} \times [0,1]$. We want to approximate $\mathbb{E}[Z|X]$. The iterative procedure writes as

$$\inf_{z,z',x^*} \sum_{m=1}^{M} \left(Z_m - \left(z \mathbf{1}_{\{X_m > x^*\}} + z' \mathbf{1}_{\{X_m \leq x^*\}} \right) \right)^2 .$$

For a fixed x^*,

$$z = \frac{\sum_{i=1}^{M} Z_i \mathbf{1}_{\{X_i > x^*\}}}{\sum_{i=1}^{M} \mathbf{1}_{\{X_i > x^*\}}}; \quad z' = \frac{\sum_{i=1}^{M} Z_i \mathbf{1}_{\{X_i \leq x^*\}}}{\sum_{i=1}^{M} \mathbf{1}_{\{X_i \leq x^*\}}}.$$

Repeat this procedure on each subset $\{(Z_m, X_m)_m : X_m \leq x^*\}$ and $\{(Z_m, X_m)_m : X_m > x^*\}$ up to a given depth p or as long as there are enough samples left.

Regression trees

- It is sufficient to determine x^* at each level, known as the split point.
- To ensure convergence of the tree, we modify the procedure
  - With a small probability q, choose x^* as the midpoint;
  - With probability $1-q$, choose the optimal $x*$.
- When M goes to infinity,

$$z = \mathbb{E}[Z|\mathbf{1}_{\{X>x^*\}}]; \quad z' = \mathbb{E}[Z|\mathbf{1}_{\{X\leq x^*\}}].$$

- When X is multidimensional, we first choose (at random) the direction of the new split and then apply the one dimensional procedure.
- If for every p, $\alpha_p \in \Theta_p$ is a minimizer of

$$\inf_{\theta\in\Theta_p} \mathbb{E}[|\Phi_p(X;\theta) - Y|^2],$$

$(\Phi_p(X;\alpha_p))_p$ converges to $\mathbb{E}[Y|X]$ in $L^2(\Omega)$ when $p \to \infty$.

Regression trees

A regression tree $\mathcal{T}_p(X)$ can be written as follows

$$\mathcal{T}_p(X) = \sum_{i=1}^{2^p} \alpha_p^i \mathbf{1}_{\left\{X \in \prod_{j=1}^d [a_p^i(j) - a_p^{i-1}(j))\right\}}$$

with $\left([a_p^{i-1}, a_p^i)\right)_{1 \leq i \leq 2^p}$ forming a partition of $[0,1]^d$ and

$$\alpha_p^i = \mathbb{E}\left[Z | X \in [a_p^{i-1}, a_p^i)\right].$$

It fits in our framework by considering $\theta^p = (a_\ell^i)_{0 \leq i \leq 2^\ell, 1 \leq \ell \leq p}$.
Beware that the value of θ^p produced by the regression tree does come from an optimization problem.

Convergence of the regression tree approximation

Proposition

$$\lim_{p\to\infty} \mathbb{E}\left[\left|\mathcal{T}_p(X) - \mathbb{E}[Y|X]\right|^2\right] = 0.$$

Proposition

For p fixed, $\lim_{M\to\infty} T_p^M(X_1;\theta^{p,M}) \to T_p(X_1;\theta^p)$ *a.s.*

In practice, use random forests

Consider a fraction of the initial data and build a regression tree $T^M_{p,k}$. Repeat this procedure, to obtain a collection of B trees: $(T^M_{p,k})_{1\leq k\leq B}$. The random forest approximation is defined by

$$\frac{1}{B}\sum_{k=1}^{B} T^M_{p,k}(X).$$

It is an ensemble method so it looses explainability but provides a more accurate approximation.

Put option in the Heston model

L	d_l	epochs=1	epochs=5	epochs=10
2	32	1.69 ($\pm$ 0.017)	1.7 ($\pm$ 0.017)	1.7 ($\pm$ 0.016)
2	128	1.69 ($\pm$ 0.017)	1.7 ($\pm$ 0.019)	1.7 ($\pm$ 0.019)
2	512	1.69 ($\pm$ 0.019)	1.69 ($\pm$ 0.019)	1.69 ($\pm$ 0.018)
4	32	1.69 ($\pm$ 0.022)	1.69 ($\pm$ 0.017)	1.7 ($\pm$ 0.018)
4	128	1.69 ($\pm$ 0.024)	1.69 ($\pm$ 0.02)	1.7 ($\pm$ 0.016)
4	512	1.68 ($\pm$ 0.025)	1.69 ($\pm$ 0.022)	1.69 ($\pm$ 0.022)
8	32	1.69 ($\pm$ 0.023)	1.69 ($\pm$ 0.02)	1.69 ($\pm$ 0.019)
8	128	1.68 ($\pm$ 0.03)	1.69 ($\pm$ 0.022)	1.69 ($\pm$ 0.02)
8	512	1.68 ($\pm$ 0.03)	1.68 ($\pm$ 0.041)	1.68 ($\pm$ 0.053)

Table: Prices using NN for put option in the Heston model with parameters the geometric basket put option with parameters with $S_0 = K = 100$, $T = 1$, $\sigma_0 = 0.01$, $\xi = 0.2$, $\theta = 0.01$, $\kappa = 2$, $\rho = -0.3$, $r = 0.1$, $N = 10$ and $M = 10^5$. The standard Longstaff Schwartz algorithm yields 1.70 ± 0.008 (resp. 1.675 ± 0.005) for an order 6 (resp. 1) polynomial regression.

Put option in the Heston model

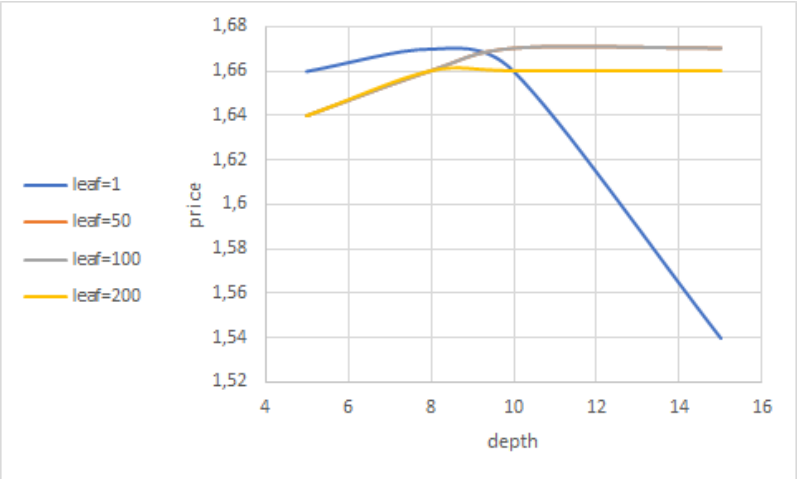

Figure: Regression trees

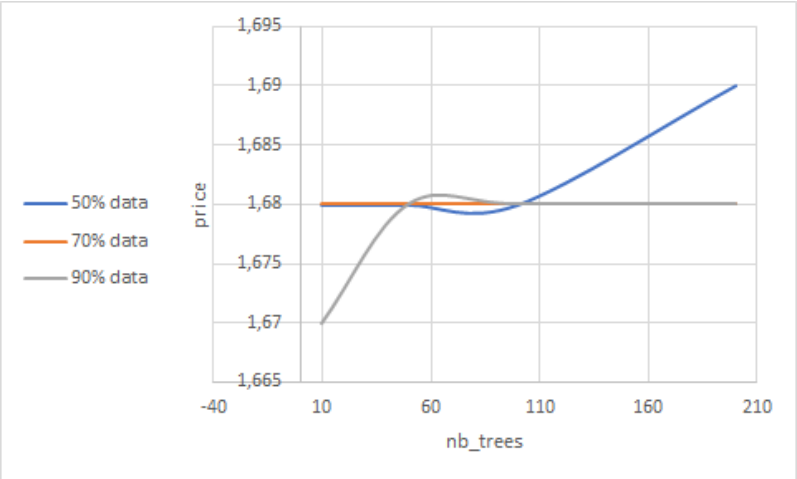

Figure: Random forests

A call on maximum in the Black Scholes model

L	d_l	epochs=1	epochs=5	epochs=10
2	128	68.99 ($\pm$ 0.179)	69.26 ($\pm$ 0.164)	69.42 ($\pm$ 0.169)
2	256	69.07 ($\pm$ 0.149)	69.42 ($\pm$ 0.125)	69.45 ($\pm$ 0.138)
2	512	69.11 ($\pm$ 0.194)	69.43 ($\pm$ 0.18)	69.51 ($\pm$ 0.167)
4	128	68.91 ($\pm$ 0.365)	69.29 ($\pm$ 0.334)	69.55 ($\pm$ 0.339)
4	256	68.72 ($\pm$ 0.358)	69.24 ($\pm$ 0.341)	69.5 ($\pm$ 0.369)
4	512	68.54 ($\pm$ 0.548)	69.17 ($\pm$ 0.356)	69.34 ($\pm$ 0.359)
8	128	68.59 ($\pm$ 0.613)	69.32 ($\pm$ 0.348)	69.71 ($\pm$ 0.497)
8	256	68.57 ($\pm$ 0.797)	69.25 ($\pm$ 0.564)	69.4 ($\pm$ 0.484)
8	512	68.32 ($\pm$ 1.444)	69.01 ($\pm$ 0.738)	69.49 ($\pm$ 0.487)

Table: Prices using NN for the call option on the maximum of 50 assets with parameters $S_0^j = 100$, $T = 3$, $r = 0.05$, $K = 100$, $\rho = 0$, $\sigma^j = 0.2$, $\delta^j = 0.1$, $N = 9$ and $M = 10^5$.

A call on maximum in the Black Scholes model

L	d_l	epochs=1	epochs=5	epochs=10
2	128	68.85 ($\pm$ 0.074)	68.96 ($\pm$ 0.095)	69.01 ($\pm$ 0.119)
2	256	68.87 ($\pm$ 0.1)	69.0 ($\pm$ 0.143)	69.07 ($\pm$ 0.146)
2	512	68.82 ($\pm$ 0.082)	69.05 ($\pm$ 0.128)	69.19 ($\pm$ 0.136)
4	128	68.84 ($\pm$ 0.221)	69.28 ($\pm$ 0.153)	69.41 ($\pm$ 0.211)
4	256	68.75 ($\pm$ 0.342)	69.14 ($\pm$ 0.296)	69.38 ($\pm$ 0.342)
4	512	68.7 ($\pm$ 0.426)	69.05 ($\pm$ 0.317)	69.35 ($\pm$ 0.254)
8	128	68.81 ($\pm$ 0.277)	69.28 ($\pm$ 0.291)	69.64 ($\pm$ 0.22)
8	256	68.57 ($\pm$ 0.512)	69.34 ($\pm$ 0.378)	69.65 ($\pm$ 0.414)

Table: Same parameters but $M = 10^6$.

A call on maximum in the Black Scholes model

depth	samples per leaf	price
50	50	66,89
50	100	66.88
100	50	67.13
100	100	67.31
200	50	67.16
200	100	67.28

Table: A call option on the maximum of 50 asset with regression trees

nb trees	sample fraction	price
10	50%	68,32
10	70%	68,32
10	90%	68,29

Table: A call option on the maximum of 50 asset with random forests

Conclusion

- Learn the continuation value using a NN or a RF instead of a polynomial regression.
- NN and RF do not help much for low dimensional problems but do scale better.
- Relatively small NN provide very accurate results (a few hundred neurons with 1 or 2 hidden layers)
- Setting $epochs = 1$ is fine for all dates but the last one.
- NN have proved to be a very versatile and efficient tool to compute Bermudan option prices...
- ...but keep in mind that using large NN is not green!

Bibliography (1)

S. Becker, P. Cheridito, and A. Jentzen. Deep optimal stopping. *Journal of Machine Learning Research*, 20(74):1–25, 2019a.

Sebastian Becker, Patrick Cheridito, Arnulf Jentzen, and Timo Welti. Solving high-dimensional optimal stopping problems using deep learning, 2019b.

Emmanuelle Clément, Damien Lamberton, and Philip Protter. An analysis of a least squares regression method for american option pricing. *Finance and Stochastics*, 6(4):449–471, 2002.

Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251 – 257, 1991.

Côme Huré, Huyên Pham, Achref Bachouch, and Nicolas Langrené. Deep neural networks algorithms for stochastic control problems on finite horizon, part i: convergence analysis. *arXiv preprint arXiv:1812.04300*, 2018.

Michael Kohler, Adam Krzyżak, and Nebojsa Todorovic. Pricing of high-dimensional american options by neural networks. *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics*, 20(3):383–410, 2010.

Michel Ledoux and Michel Talagrand. *Probability in Banach spaces*, volume 23 of *Ergebnisse der Mathematik und ihrer Grenzgebiete (3) [Results in Mathematics and Related Areas (3)]*. Springer-Verlag, Berlin, 1991. ISBN 3-540-52013-9. Isoperimetry and processes.

Bibliography (2)

F.A. Longstaff and R.S. Schwartz. Valuing American options by simulation : A simple least-square approach. *Review of Financial Studies*, 14:113–147, 2001.

Edith Mourier. Eléments aléatoires dans un espace de Banach. *Ann. Inst. H. Poincaré*, 13: 161–244, 1953.

Reuven Y. Rubinstein and Alexander Shapiro. *Discrete event systems*. Wiley Series in Probability and Mathematical Statistics: Probability and Mathematical Statistics. John Wiley & Sons Ltd., Chichester, 1993. ISBN 0-471-93419-4. Sensitivity analysis and stochastic optimization by the score function method.

J.N. Tsitsiklis and B. Van Roy. Regression methods for pricing complex American-style options. *IEEE Trans. Neural Netw.*, 12(4):694–703, 2001.