Journal of Mathematical Modeling
Vol. 11, No. 2, 2022, pp. 229–244. Research Article

JMM

Pricing American option under exponential Levy Jump-diffusion model using Random Forest instead of least square regression

Maidoumi Mohamed[*], **Zahid Mehdi**, **Daafi Boubker**

LAMAI, Cadi Ayyad University, Marrakech, Morocco
Email(s): maidoumi-22@hotmail.fr, zahidmehdi10@gmail.com, boubkerdaafi@gmail.com

Abstract. In this paper, we aim to propose a new hybrid version of the Longstaff and Schwartz algorithm under the exponential Levy Jump-diffusion model using Random Forest regression. For this purpose, we will build the evolution of the option price according to the number of paths. Further, we will show how this approach numerically depicts the convergence of the option price towards an equilibrium price when the number of simulated trajectories tends to a large number. In the second stage, we will compare this hybrid model with the classical model of the Longstaff and Schwartz algorithm (as a benchmark widely used by practitioners in pricing American options) in terms of computation time, numerical stability and accuracy. At the end of this paper, we will test both approaches on the Microsoft share MSFT as an example of a real market.

Keywords: Monte Carlo simulation, Levy jump-diffusion model, Longstaff and Schwartz algorithm, American option, Random Forest RI regression, Microsoft "MSFT" put option, dynamic programming.
AMS Subject Classification 2010: 11K60, 11K62, 11K68.

1 Introduction

A derivative can be described as a financial instrument contracted between two or more parties whose value depends on the values of one or more agreed-upon underlying assets. Such as securities, indexes, fixed income, currency (forex option), etc. Hedgers might use derivatives to minimize the risk of future market movements while speculators may use derivatives to make profits. The valuation of derivatives has been the subject of extensive research in both academia and industry. Except for simple derivatives

[*] Corresponding author.

Received: 11 February 2022/ Revised: 21 August 2022/ Accepted: 09 September 2022
DOI: 10.22124/jmm.2022.21756.1909

such as futures, forwards and swaps. A large range of complex derivatives are valued by using numerical methods as the three main approaches to value complex derivatives; the tree [6], PDE [14] and Monte Carlo. However, due to the complexity of the implementation and numerical constraints (curse of dimensionality), the tree approach and the classical finite difference PDE approach are not practical for the pricing of high-dimensional derivatives. Therefore, the Monte Carlo method is commonly used in the valuation of high-dimensional derivatives. The Monte Carlo method must include some additional numerical techniques to value early-exercising instruments, such as American options, Bermuda options, structured callable bonds, etc. Hence, in the order to find the appropriate exercise strategy to price the above derivatives, various strategies have been proposed. As far as these strategies are concerned, it is not computationally feasible to perform a Monte Carlo simulation at the beginning of the exercise period to calculate the expected profit from continuation value. Barraquand and Martineau [3] devised a stratified state technique that sorts stock price paths by a state variable (rather than the stock price). However, an estimate of the error in the results cannot be derived using Barraquand and Martineaus method. Broadie and Glasserman [10] suggested a simulated tree method for pricing American options as well as upper and lower bounds for these options. Longstaff and Schwartz [20] suggested a least-squares Monte Carlo algorithm to price American options which involves a least-squares regression within the early exercise time to calculate the expected return from continuation value. Stentoft [26] developed a least-squares Monte Carlo method widely used by practitioners because it is computationally efficient with straightforward construction. The main advantage of the Monte Carlo simulation is its fast convergence rate which is independent of the number of underlying variables as well as its easiness in handling a wide range of models.

On the other hand, Random Forest (RF) combines basic predictors or estimators that are trees giving rise to what is now called the decision tree method. Generally, this method is the ensemble technique whose general principle (Dietterich [12]) is to build a collection of predictors. Then, aggregate them all together. In regression, aggregating the predictions of q predictors is, for example, the same as averaging them. Each predictor provides a prediction $\hat{y}_l$, and the final prediction is then $\frac{1}{q}\sum_{l=1}^{q}\hat{y}_l$. In classification, aggregation consists in making a majority vote among the class labels provided by predictors. Historically, Kwok and Carter [18] are the first to mention ensemble decision trees in literature. The authors show that averaging many decision trees with different structures consistently outperforms any other ensemble technique. Later, Breiman [7] was one of the first to emphasize, both theoretically and practically, the combination of numerous versions of an estimator into an ensemble technique might result in significant improvements in accuracy. Dietterich and Kong [16] proposed to randomize the choice of the best split at a given node by uniformly selecting one of the 20 best splits of node t at random. Amit and Brunel [1] presented a randomized form of the tree induction approach that involves searching for the best split at each node over a random sub-sample of the variables. Later on, Ho [13], inspired by the principles of bagging [7] and random subsets of variables [1], proposed to use the random subspace method to construct a decision forest whose trees are grown on random subsets of the input variables rather than all variables. Breiman [7, 9] combined Bagging with the random variable selection at each node in his seminal paper on RF. Using both methodologies, he generated one of the most effective machine learning algorithm which works unexpectedly well for almost any task. The author empirically shows that RF can compete with boosting [2] and arcing algorithm [8] which are both designed to reduce bias, whereas forests are focused on reducing variance in addition to bias. While the original foundations are the result of the contributions of many authors, Breiman is often considered the father of the randomized tree forest. Part of his success can be attributed to his groundbreaking theoretical analysis which always complemented his empirical analysis of algorithms.

This paper is devoted to the interesting issue of modelling the American option price with the sharp change in asset prices using RF with random inputs. In Section 2, we are presenting the theoretical framework of the least squares and RF regression in the Longstaff and Schwartz algorithm. In Section 3, we are focusing on the numerical implementation of the classical Longstaff and Schwartz algorithm and compare it to the RF regression approach. Furthermore, we are highlighting the theoretical and asymptotic convergence towards an equilibrium price. At the end of this paper, we are comparing between the two approaches based on real market data such as Microsoft's MSFT put option as an example from Yahoo Finance.

2 A conceptual model for pricing American option

2.1 Levy Jump-diffusion for forward simulation of asset price

The source of asset price jumps, according to Merton [23], can be generally produced by a specific event in the firm or its industry which might have a limited impact on the rest of the market. The jump component will be then uncorrelated with the market and constitute an unsystematic risk. Assume that $\left(\Omega, \mathscr{F}, (\mathscr{F}_t)_{0\leq t\leq T}, \mathbb{P}\right)$ is a filtered probability space, where $T \geq 0$ and $W_t = \left(W_t^1, \ldots, W_t^d\right)$ a d-dimensional Brownian motion. N is the Poisson random measure on $\mathscr{B}([0,T]) \otimes \mathscr{E}$, where $\mathscr{B}([0,T])$ is the Borel σ-algebra on $[0,T]$, and $(E, \mathscr{E})$ is a measurable space, where $E = \mathbb{R}^q$. We define $\mathscr{E}$ as the Borel σ-algebra on E. $\mathbb{P}$ is the probability measure on $\mathscr{F}$. The filtration $(\mathscr{F}_t)_{0\leq t\leq T}$ is completed with all $\mathbb{P}$-null sets, right continuous and $\mathscr{F}_t = \mathscr{F}_t^{W,N}$ is generated by $(W_t, N(\cdot, [0,t], \cdot))$ for $t \in [0,T]$. Assume that $\mathscr{F} = \mathscr{F}_T^{W,N}$, W and N are mutually independent under $\mathbb{P}$. Suppose that the compensating measure of N is $v(\,\mathrm{d}t,\,\mathrm{d}e) := v(\,\mathrm{d}e)\mathrm{d}t$, where v is a σ-finite measure on $(E, \mathscr{E})$ satisfying $\int_E \left(1 \wedge |e|^2\right) v(de) < \infty$. The corresponding compensated Poisson random measure is defined by $\tilde{N}(\omega, \mathrm{d}t,\,\mathrm{d}e) := N(\omega, \mathrm{d}t,\,\mathrm{d}e) - v(\,\mathrm{d}e)\mathrm{d}t$.

Definition 1 ([15] (A Quick reminder of the Levy process))**.** *A stochastic process $(X_t)_{t\geq 0}$, defined on $(\Omega, \mathscr{F}, \mathbf{P})$, and has its values in $\mathbf{R}^d$ with RCLL trajectories, is a Levy process if*

- *Its increments are independent: for each increasing sequence $t_0, \ldots, t_n$ the random variables $X_1 - X_0, \ldots, X_{t_n} - X_{t_{n_1}}$ are independent.*
- *Its increments are stationary: $X_{t+h} - X_t$ do not depend on t.*
- *It verifies the property of stochastic continuity: for all $\varepsilon > 0$, $\lim\limits_{h\longrightarrow 0} \mathbf{P}(\mid X_{t+h} - X_t \mid \geq \varepsilon) = 0$, i.e., the probability to have two jumps at the same time is 0.*

Definition 2 ([17] (Levy Jump diffusion process))**.** *A stochastic process $(X_t)_{t\geq 0}$, defined on $(\Omega, \mathscr{F}, \mathbf{P})$, and has its values in $\mathbf{R}^d$ with RCLL trajectories, is a Levy Jump diffusion process if*

$$dX_t = rdt + \sigma dW_t + J_t d\tilde{N}_t, \tag{1}$$

where r is the risk-free rate, σ is a constant volatility and J_t is the jump amplitude of $\tilde{N}_t$ the compound Poisson process at time t. We note $(O_t)_{t\in[0,T]}$ as the payoff process of $(S_t)_{t\in[0,T]}$ at time t. In the following, we assume that the underlying $(S_t)_{t=t_0,\ldots,t_N}$ is a Levy Jump diffusion process.

Definition 3 ([11] (Snell Envelope)). *We define the Snell Envelope $(U_t)_{t\geq 0}$ as the value of the American option at t, and it is given as $U_t = \operatorname{ess\,sup}_{\tau\in\Gamma_t} E[O_\tau|\mathscr{F}_t]$, where Γ_t is the set of all $\mathscr{F}_t$-stooping time in $[t,T]$.*

Theorem 1 ([24] (First Fundamental Theorem of Asset Pricing)). *There must be at least one risk-neutral probability measure Q is the equivalent of the initial probability measure P (i.e., $\mathbb{Q}(A) = 0 \Leftrightarrow P(A) = 0\ \forall A \in F$) for a discrete market on a discrete probability space $\left(\Omega, \mathscr{F}, (\mathscr{F}_t)_{0\leq t\leq T}, \mathbb{P}\right)$ to be arbitrage-free.*

In a complete market, the existence and uniqueness are satisfied. For further explanations, see the second fundamental theorem of asset pricing in [5].

Theorem 2 ([24]). *Following the same notations as above, in a free arbitrary market, and under an appropriate equivalent measure $\mathbb{Q}$, the price at time t is the conditional expected payout at T, i.e.,*

$$U_t = \mathbf{E}[e^{(T-t)r}U_T \mid \mathscr{F}_t].$$

Hence, $e^{-r(T-t)}U_t$ is a martingale under $\mathbb{Q}$. Such an equivalent martingale measure is also called a measure risk-neutral because the return on the risk-free investment is the same as the expected return on the asset.

2.2 Dynamic programming of the Longstaff and Schwartz algorithm

To emphasize the effectiveness of the Longstaff and Schwartz algorithm, we compare it to the approach of Tsitsiklis and Roy [27]. In the following, we assume that $\pi = \{t_0 = 0, \ldots, t_N = T\}$ is the set of possible exercise times (Bermudian option) with the following dynamic algorithm

$$\begin{cases} U_{t_N} = O_{t_N}, \\ U_{t_k} = \max(O_{t_k}, E[U_{t_{k+1}}|F_{t_k}]). \end{cases} \tag{2}$$

This dynamic programming algorithm is based on the re-simulation of paths at each time step and is obtained from the Snell Envelope. On the other hand, the Longstaff and Schwartz algorithm uses the same trajectory for each time step due to its dynamic programming that depends on the stopping time (see system 3) instead of the value function as in the case of Tsitsiklis and Roy [27]. This approach is very advantageous because of its robustness, low computational burden and time-saving compilations.

$$\begin{cases} \tau_N = t_N, \\ \tau_k = t_k \mathbf{1}_{O_{t_k}\geq E[O_{t_{k+1}}|\mathscr{F}_{t_k}]} + \tau_{k+1}\mathbf{1}_{O_{t_k}\leq E[O_{t_{k+1}}|\mathscr{F}_{t_k}]}, \end{cases} \tag{3}$$

where τ_k is the smallest optimal stopping time after t_k. The first line of system (3) highlights the fact that the Snell Envelope is equal to the payoff at time T. The second line is a comparison between the payoff at time t_k and the conditional expectation of the payoff at time t_{k+1} on $\mathscr{F}_{t_k}$ that we design by continuation value. Now, the main question that arises is how to approximate the conditional expectation.

Theorem 3 ([22]). *Assume that $(X_t)_{t=0,\ldots,T}$ is a Markovian process. There is ϕ_k which is a $\mathscr{F}_t$ measurable function where*

$$E[O_{t_{k+1}}|\mathscr{F}_{t_k}] = E[O_{t_{k+1}}|X_{t_k}] = \phi_k(X_{t_k}).$$

Because of the orthogonality of conditional expectation in L^2 we can compute ϕ_k by the least squares regression $\phi_k(X_{t_k}) = \inf_{\phi\in\Phi} E[|O_{t_{k+1}} - \phi(X_{t_k})|^2]$, where Φ is the set of $L^2(\sigma(X_{t_k}))$ functions. The classical numerical Longstaff and Schwartz algorithm is based on the approximation of the conditional expectation by a finite p-dimensional vector (e.g., polynomials basis, logistic function, etc.) in L^2 space. Let $\varphi = (\varphi_1, \varphi_2, \dots, \varphi_p)$ be a finite p-dimensional basis of functions in L^2, subsequently, ϕ_k is written as follows $\phi_k^p(X_{t_k}, \theta_k) = \sum_{n=1}^{p} \theta_n \varphi_n(X_{t_k})$.

2.3 Convergence of Monte Carlo simulation

Let $\{S_{t_0}^{(m)}, \dots, S_{t_N}^{(m)}\}$ be the price paths and $\{O_{t_0}^{(m)}, \dots, O_{t_N}^{(m)}\}$ the payoff paths, where $m \in \{1, \dots, M\}$. We present the least squares algorithm with Monte Carlo approximation policy as

$$\begin{cases} \hat{\tau}_N^{p,(m)} = t_N, \\ \hat{\tau}_k^{p,(m)} = t_k \mathbf{1}_{O_{t_k}^{(m)} \geq \phi_k^p(S_{t_k}^{(m)}, \hat{\theta}_k^{p,M})} + \tau_{k+1} \mathbf{1}_{O_{t_k}^{(m)} \leq \phi_k^p(S_{t_k}^{(m)}, \hat{\theta}_k^{p,M})}, \end{cases} \tag{4}$$

where $\hat{\tau}_k^{p,(m)}$ is the smallest optimal stopping time after t_k on the mth path for the p-dimensional approximation. Hence, the conditional expectation $\mathbf{E}[O_{t_{k+1}} | \mathscr{F}_{t_k}]$ is computed by the following minimization problem

$$\theta_k^p = \arg\inf_{\theta\in\Theta} \mathbf{E}(| \, O_{\hat{\tau}_{k+1}^p} - \phi_k^p(S_{t_k}, \theta) \, |^2),$$

or, in another way, by its Monte Carlo approximation

$$\hat{\theta}_k^{p,M} = \arg\inf_{\theta\in\Theta} \frac{1}{M} \sum_{m=1}^{M} | \, \phi_k^p(S_{t_k}^{(m)}, \theta) - O_{\hat{\tau}_{k+1}^{p,(m)}}^{(m)} \, |^2,$$

where Θ is a finite set in $\mathbb{R}$. The coefficients $(\hat{\theta}_k^{p,M})_{k=1,\dots,N}$ give us convenient multipliers of basis vectors in L^2 for determining the optimal policy of system (4). Thus, the option price at time 0 is obtained as $U_0^p = \mathbf{E}(O_{\hat{\tau}_1^p})$, where $\tau_1 = \inf\{k \in \{1, \dots, N\} \mid O_k = U_k\}$. Therefore, the Monte Carlo approximation is

$$U_0^{p,M} = \sum_{m=1}^{M} O_{\hat{\tau}_1^{p,(m)}}^{(m)}.$$

Theorem 4. *Let* $0 \leq j \leq N$ *and we assume that* $\varphi^p(S_{t_j}) = (\varphi_1(S_{t_j}), \dots, \varphi_p(S_{t_j}))$ *be a total basis in* $L^2(\sigma(S_{t_j}))$. *Then*

$$\lim_{p\to+\infty} \mathbf{E}(O_{\tau_j^p} \mid \mathscr{F}_{t_j}) = \mathbf{E}(O_{\tau_j} \mid \mathscr{F}_{t_j}) \text{ in } L^2.$$

Proof. See [11]. □

This result can be proved theoretically but not numerically. Unlike the following convergence result.

Theorem 5. *Assume that for* $0 \leq j \leq N$, $P(\beta_j \varphi(S_{t_j}) = O_{t_j}) = 0$, *and the simulated M paths are independent, then* $\lim_{M\longrightarrow+\infty} U_0^{p,M} = U_0^p$ *almost surely.*

Proof. First of all, we prove that for $0 \leq j \leq N$, $\frac{1}{M}\sum_{k=1}^{M} O_{\tau_j^{k,p,M}}^k \longrightarrow \mathbf{O}_{\tau_\mathbf{j}^\mathbf{p}}$, when $M \longrightarrow +\infty$ almost surely. Then, we conclude our result (see [11]). The numerical illustration of this theorem is shown in Figures 3-6. □

2.4 Least squares regression algorithm

Let p be the number of regressors (dimension of L^2 basis), M the number of paths. We aim to calculate the least squares coefficients $\beta = (\alpha, \beta_1, \beta_2, \dots, \beta_p)$. The orthogonal projection of $O(S_{t_N})$ on $L^2(\sigma(S_{t_{N-1}}))$ is the vector $\alpha + \beta_1\varphi_1(S_{t_{N-1}}) + \cdots + \beta_p\varphi_p(S_{t_{N-1}})$. At time t_{N-1} we use an ordinary least squares backward to regress $O(S_N)$ (on the space generated by these regressors $\varphi_1(S_{t_{N-1}}), \dots, \varphi_p(S_{t_{N-1}})$) by minimizing $O(S_{t_N}) - \alpha - \beta_1\varphi_1(S_{t_{N-1}}) - \cdots - \beta_p\varphi_p(S_{t_{N-1}})$. In this case the continuation value becomes

$$\phi^p_{t_{N-1}} = \mathbb{E}[e^{-r\Delta t} O(S^i_{t_N}) | S_{t_{N-1}}] \approx e^{-r\Delta t}(\alpha + \beta_1\varphi_1(S_{t_{N-1}}) + \beta_2\varphi_2(S_{t_{N-1}}) + \cdots + \beta_p\varphi_p(S_{t_{N-1}})). \tag{5}$$

On each trajectory from M' in the money paths at time t_{N-1} we obtain the least squares coefficients $\hat{\beta}_{t_{N-1}}$ by the least squares estimators

$$\hat{\beta}_{t_{N-1}} = \begin{pmatrix} \alpha_{t_{N-1}} \\ \beta_{1,t_{N-1}} \\ \vdots \\ \beta_{p,t_{N-1}} \end{pmatrix} = \left(S^T S\right)^{-1} S^T \left(O(S^1_T) \quad \cdots \quad O(S^{M'}_T)\right),$$

where

$$S = \begin{pmatrix} \varphi_1(S^1_{t_{N-1}}) & \cdots & \varphi_1(S^{M'}_{t_{N-1}}) \\ \vdots & \ddots & \vdots \\ \varphi_p(S^1_{t_{N-1}}) & \cdots & \varphi_p(S^{M'}_{t_{N-1}}) \end{pmatrix}.$$

Thus, we apply Eq. (5) backward for $t_{t_{N-2}}, t_{N-3}, \dots, t_0 = 0$.

3 Random Forest RI algorithm

3.1 Methodology

Random Forest-RI stands for Random Forest with Random Inputs which is a predictor obtained by aggregating a set of predictors by each of the following trees $\{h(\cdot, \theta_k), k = 1, \dots, B\}$. Their construction principle is first to generate several bootstrap samples $(\mathscr{L}^{\Theta_1}_n, \dots, \mathscr{L}^{\Theta_q}_n)$. Then, on each sample $\mathscr{L}^{\Theta_i}_n$, a variant of the Cart algorithm (see the Figure 1) is applied. More precisely, a tree is constructed in the following way: to cut a node, we randomly draw a number m' of variables and look for the best cut only according to the m' selected variables (e.g., the inter-class variance as a criterion). Moreover, the constructed tree is completely developed (i.e., maximal tree) and it is not pruned (see [9]). The collection of trees obtained is finally aggregated (e.g., average in regression $h(\cdot) = \sum_{b=1}^{B} h(\cdot, \theta_k)$). The drawing of m variables at each node represents an additional hazard compared to the Bagging method [7]. For RF-RI, there are therefore two sources of randomness for generating the collection of individual predictors: the randomness due to the bootstrap and the randomness of the choice of variables to cut each node of the tree. Thus, we disturb both the sample on which we launch the Cart algorithm and the core of the Cart construction. This random drawing of variables to cut out nodes had already been used by Amit and Geman [1] in image recognition problems. Their method greatly influenced Leo Breiman [9] in his development of RF-RI.

Algorithm 1 RF-RI for pricing American option

```
Let $S$ be a multi-dimensional underlying and $(O_{t_k})_{k=0,\dots,N}$ the discounted payoff at time $t_k$.
Generate $\big(S^{(m)}|_{t_0,\dots,t_N},\ m = 1,\dots,M\big)$ which are $M$ independent copies of $(S_{t_1},\dots,S_{t_N})$.
Let $\hat{\tau}_N^{(m)} = t_N\ m = 1,\dots,M$.
for $k = N-1$ to 1 do
    Inputs RF-RI:
        $d_k^L = \{(S_{t_k}^{(1)}, O_{t_{k+1}}^{(1)}),\dots,(S_{t_k}^{(L)}, O_{t_{k+1}}^{(L)})\}$, $L$ in the money path samples from $M$ samples.
        $B$ number of trees.
        $m' \in \mathbf{N}^*$ the number of underlyings to cut a node.
    for $i = 1$ to $B$ do
        $\theta_i$ draws bootstrap sample from $d_k^L$.
        Build a tree by the Cart algorithm. Each split is selected to minimize the inter-variance into the
        resulting node. This tree gives rise to the predictor $h(\cdot,\theta_i)$.
    end for
    Output:
    for $m = 1$ to $M$ do
        $\hat{\tau}_k^{(m)} = t_k \mathbf{1}_{O_{t_k}^{(m)} \geq h(S_{t_k}^{(m)})} + \tau_{k+1} \mathbf{1}_{O_{t_k}^{(m)} \leq h(S_{t_k}^{(m)})}$, where the continuation value is

            $\phi_k(S_{t_k}^{(m)}) = h(S_{t_k}^{(m)}) = \sum_{i=1}^{B} h(S_{t_k}^{(m)}, \theta_i),\ m = 1,\dots,L.$

        if $O(S_{\hat{\tau}_k^{(m)}}) \geq \phi_k(t_k, S_{\hat{\tau}_k^{(m)}})$ then
            The early exercise time (i.e., the optimal stopping time on each trajectory) on each trajectory
            is $\hat{\tau}_k^{(m)} = \tau_i,\ i = 1,\dots,M$, if it has not taken place previously.
        end if
    end for
end for
The value of the American option at time $t = 0$ is

    $U_0^M = \max(\phi_0(S_{t_0}), \sum_{i=1}^{M} \exp(-r\tau_i) O(S_{\tau_i}^i)).$
```

Remark 1. *When we have a small number of variables selected in the Cart algorithm, this comes closer to a random partition of the selected variables, e.g., if $m' = 1$ the selection of variable is chosen randomly, only the cutting axes d_i define the distribution. Hence, the random selection of variables i.i.d decrease the correlation between trees, and subsequently, the variance of the aggregated estimator decreases. On the other hand, the random choice generates a bad fit on the learning sample and later increases the bias of our model. We can also emphasize the influence of the chosen parameters including the number of observations in the final nodes (before stopping growing trees) on the choice of m' and d. Indeed, in the instance where a small number of observations in the final nodes is chosen. Then, we have a large variance and a weak bias. In this situation, we tend to decrease m'. In the instance where a large number*

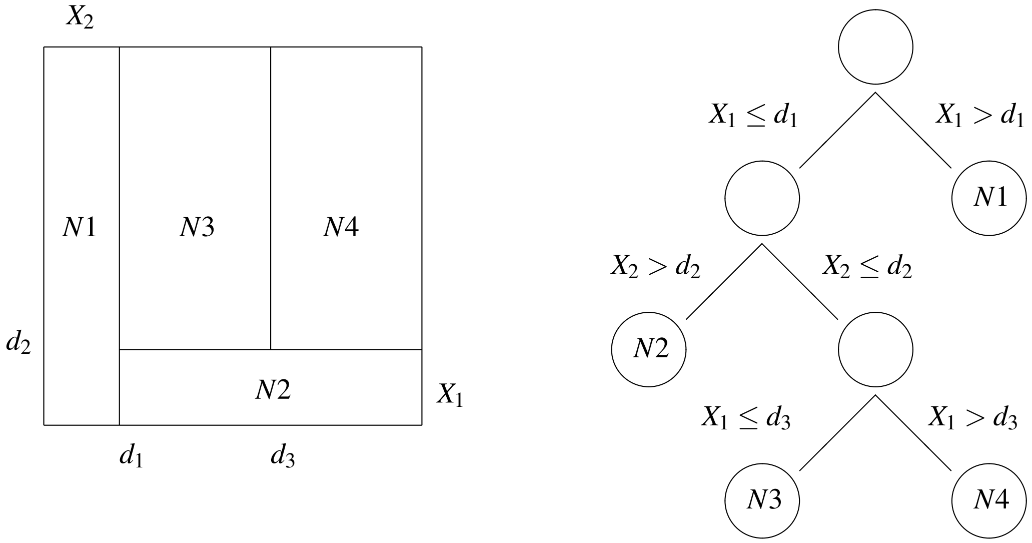

Figure 1: Cart algorithm illustration of one grown tree $m = 2$.

of observations is chosen in the final nodes, we will have a higher bias and a lower variance. Then, we try to increase the depth of trees (numbers of d, i.e., card(d)) or improve the criterion of the choice d_i. In many programming languages, the Default choice of the final observation is $m' = \frac{p}{3}$ for regression and $m' = \sqrt{p}$ for classification. Other discussions can be developed about evaluating this model (Out Bag Error and Importance sampling) but we are not going any further on these points.

4 Numerical implementation

According to the same logic in the work [21] in which the study revolves around the comparison of the asymptotic behaviour of Black and Sholes and Levy Jump-diffusion in the classical algorithm of Longstaff and Schwartz. We will apply the same methodology under Levy jump-diffusion to build the equilibrium price for our new hybrid algorithm using RF-RI regression instead of the least squares regression. In the simple case of a single underlying where the random subspace [13] is excluded, we limit ourselves to the bagging method [7]. Moreover, we will compare the new hybrid algorithm to the classical Longstaff and Schwartz algorithm as a benchmark.

4.1 Forward simulation of trajectories by Levy Jump-diffusion

Let N_T be the number of jumps between 0 and $T = 1$. T_i, $i = 1, \dots, N_T$ are the jump times. We break down the interval $[0, T]$ in N equal time periods $\Delta t = T/N$ to build the discretization $t_0 = 0, \dots, t_N = T$, where $N = 1000$ and therefore we use Euler scheme to simulate S by discretizing Eq. (1) as

$$S(t_{j+1}) - S(t_j) = r\Delta t + \sigma W_{\Delta t} + \sum_{i=0}^{N_T} J_i, \tag{6}$$

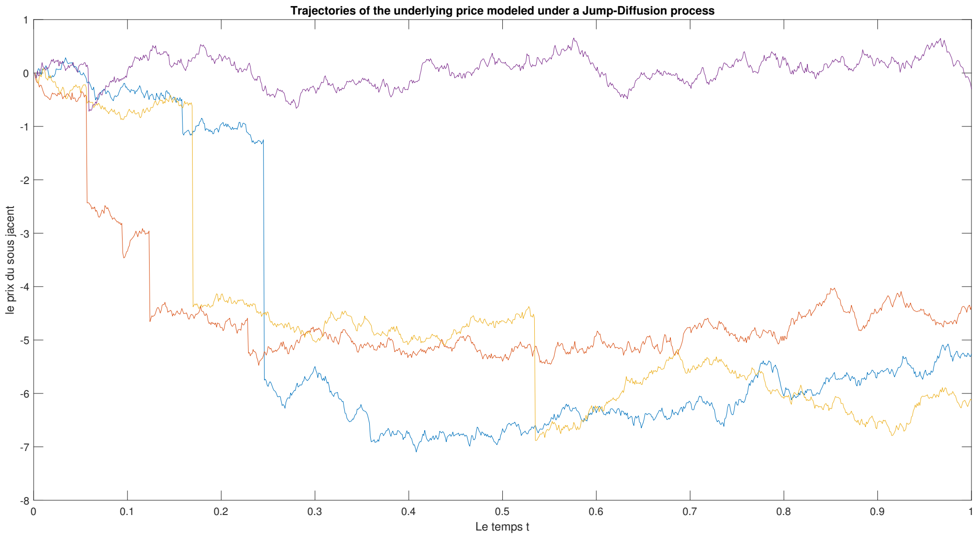

Figure 2: Levy Jump-diffusion asset paths with $\lambda_1 = 1$, $\lambda_2 = 1/4$ and $S_0 = 0$.

where $W_{\Delta t} \sim N(0,\Delta t)$, hence trajectories $S_t|_{t=t_0,\dots,t=t_N}$, where $S_0 = 80\$$ are calculated from the application step-by-step of Eq. (6) as follows: first of all we generate a Wiener process with volatility $\sigma = 4\%$ and free risk rate $r = 2\%$, $w_i = T_{i+1} - T_i$ as the process of duration between jumps which follows exponential law with the parameter $\lambda_2 = \frac{1}{4}$ (i.e. on the average we have $\frac{1}{\lambda_2}$ jumps in the meantime $[0,T]$), subsequently, we simulate time of jumps as $T_i = \sum_{j=1}^{i} w_j$ up to T_{N_T}, where $N_T = \sum_{i=1}^{\infty} 1_{\sum_{j=1}^{i} w_j < T}$ is the total number of jumps. As the case may be, we can generate negative exponential law for the jump size J_i with its parameter $\lambda_1 = 1$, i.e., losses which follow exponential laws (see the example of Figure 2).

4.2 Backward simulation of least squares regression

We consider the case of $p = 2$ (L^2 basis with three polynomial vectors) to highlight the ordinary least squares regression algorithm at time t_{N-1}. For example, we choose the following form of the regression function $\varphi = (\varphi_0(S_{t_{N-1}}) = 1, \varphi_1(S_{t_{N-1}}) = S_{t_{N-1}}, \varphi_2(S_{t_{N-1}}) = S^2_{t_{N-1}})$. Let $\hat{\beta} = (\hat{\alpha}_{t_{N-1}}, \hat{\beta}_{1,t_{N-1}}, \hat{\beta}_{2,t_{N-1}})$ be the estimator of the least squares (see Subsection 2.4) which minimizes the following error

$$\sum_{i=1}^{M'} [O(S^i_{t_N}) - \alpha - \beta_1 S^i_{t_{N-1}} - \beta_2 S^{i2}_{t_{N-1}}]^2.$$

At each trajectory, at the time t_N the value of the option is equal to the terminal payoff $O(S^i_{t_N})$, at the time t_{N-1} we estimate the continuation value if the trajectory is in-the-money and we reject out-of-money trajectories. So, on each trajectory, we have M' couples $(S^i_{t_{N-1}}, O(S^i_{t_N}))$ among M trajectories which explain $O(S^i_{t_{N-1}})$, using regressors $1, \varphi_1(S_{t_{N-1}}) = S_{t_{N-1}}, \varphi_2(S_{t_{N-1}}) = (S_{t_{N-1}})^2$. According to Eq. (5) at time t_{N-1} on trajectory i the continuation value is

$$\phi_k^2(t_{N-1}, S^i_{t_{N-1}}) = \mathbb{E}[e^{-r\Delta t} O(S^i_{t_N}) | S^i_{t_{N-1}} = S_{t_{N-1}}] \cong e^{-r\Delta t}(\hat{\alpha}_{t_{N-1}} + \hat{\beta}_{1,t_{N-1}} S^i_{t_{N-1}} + \hat{\beta}_{2,t_{N-1}} (S^i_{t_{N-1}})^2).$$

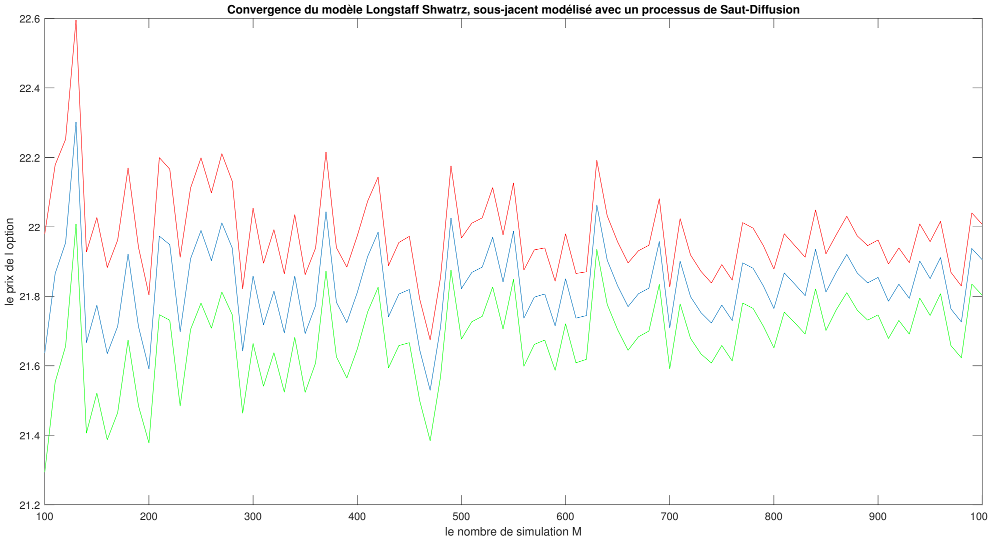

Figure 3: Numerical convergence of least squares Monte Carlo. Confidence level is 95% (the estimated value of the American option in blue, upper bound in red and lower bound in green). M from 100 to 1000 by 10 steps.

We notice that the intrinsic value $e^{-r\Delta t}O(S^i_{t_N})$ represents the discounted cash flows earned at time t_N in the case that the option is not exercised at time t_{N-1}. Thus, we make a comparison between the payoff $O(S^i_{t_N})$ and $\phi^p_k(t_{N-1}, S^i_{t_{N-1}})$ on each trajectory if $O(S^i_{t_N}) \geq \phi^p_k(t_{N-1}, S^i_{t_{N-1}})$. The early exercise of the option has therefore taken place at time t_{N-1} if it has not taken place previously. Hence, the value of option at time t_{N-1} on trajectory i is

$$U^{2,i}_{t_{N-1}} = \max(O(S^i_{t_{N-1}})), \phi^p_k(S^i_{t_{N-1}})).$$

We do the same for all times $t_{N-2}, \ldots, t_0$. Finally, an updated payoff is assigned to each trajectory i $\exp(-r\tau_i)O^i_\tau$ or is equal to 0 in the event of no exercise. Note that $\tau \in \{t_0 = 0, \ldots, t_N = T\}$ is the first time when $O(S^i_{t_N}) \geq \phi^p_k(t_{N-1}, S^i_{t_{N-1}})$ and as previously proved this is an optimal exercise time. Lastly, we estimate the final price of the option at time t_0 by

$$U^{2,M}_0 = \sum_{i=1}^{M} \exp(-r\tau_i)O(S^i_{\tau_i}).$$

Figures 3, 4 depict the backward least squares in Longstaff and Schwartz algorithm (blue line) based on the Levy Jump-diffusion implemented in the previous Subsection 4.1.

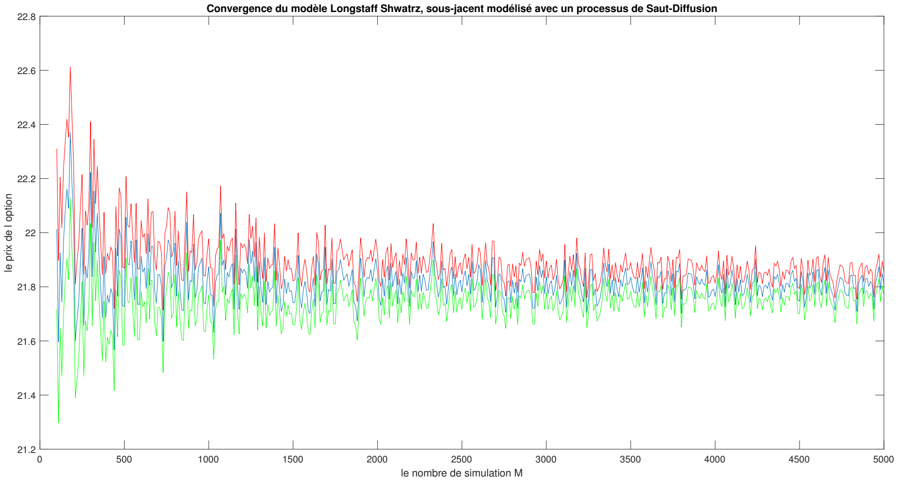

Figure 4: Numerical convergence of the least squares Monte Carlo Longstaff and Schwartz. Confidence level is 95% (the estimated value of the American option in blue, upper bound in red and lower bound in green). M from 100 to 5000 by 10 steps.

4.3 Numerical stability and performance of the hybrid RF algorithm compared to the least squares regression

Similarly, as the Algorithm 1 (RF implemented with `fitrensemble`[1] function in MATLAB) and based on the Levy Jump-diffusion process implemented in Subsection 4.1 we implement the RF hybrid algorithm according to the number of simulated paths, where the strike is $K = 100\$$. Hence, we illustrate the numerical convergence of the option price toward an equilibrium price ($U_0^{2,M} = 21.8270\$$ for the RF algorithm and $U_0^{2,M} = 21.8270\$$ for the least squares algorithm by using $M = 10000$ paths). On the other hand, we compare the empirical stability of both algorithms in Figures 4, 6, especially, in terms of accuracy in Table 1 and computational time in Table 2. We notice in this example that the least squares algorithm stabilizes the size of the confidence interval numerically when the M tends to $M = 5000$ better than the hybrid RF.

Table 1: Equilibrium price and confidence interval according to the number of samples for the least squares (LS) and RF algorithms with a confidence level 95%.

Trajectories	mean RF	min RF	max RF	mean LS	min LS	max LS
M=1000	22.41	22.2234	22.6142	21.7427	21.6420	21.8433
M=5000	22.0345	21.8945	22.145	21.8000	21.7546	21.8454
M=8000	21.8100	21.7509	21.8778	21.8000	21.77746	21.8401
M=10000	21.8270	21.8080	21.7830	21.8270	21.7950	21.8390

[1] For more details about the RF implementation on MATLAB fitrensemble visit: https://fr.mathworks.com/help/stats/fitrensemble.html

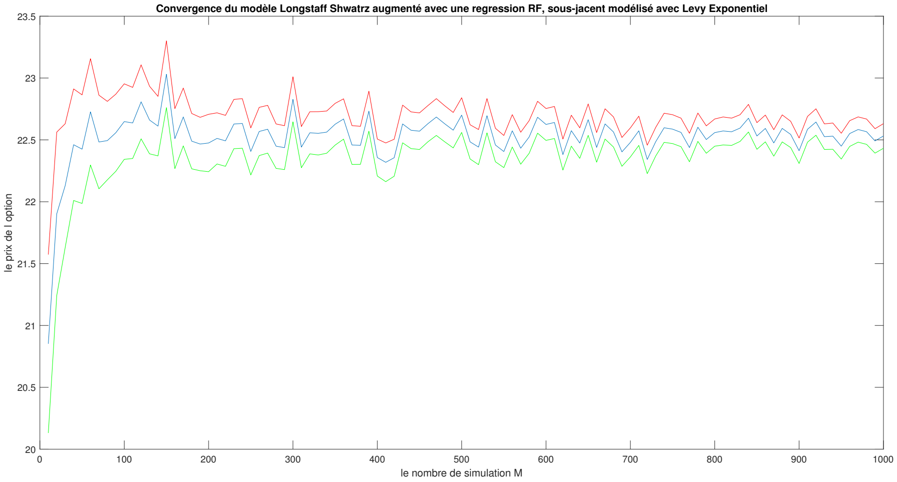

Figure 5: Numerical convergence of the RF hybrid algorithm. Confidence level is 95% (the estimated value of the American option in blue, upper bound in red and lower bound in green). M from 100 to 1000 by 10 steps.

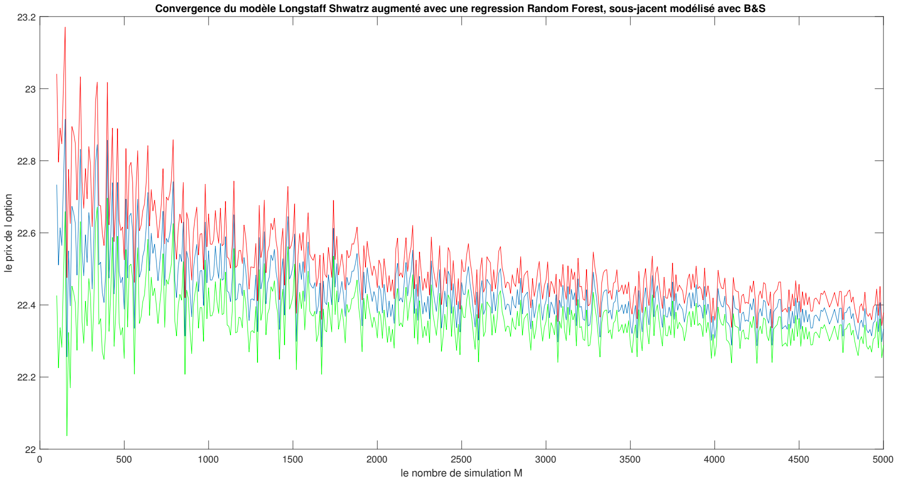

Figure 6: Numerical convergence of the RF hybrid algorithm. Confidence level is 95% (the estimated value of the American option in blue, upper bound in red and lower bound in green), M From 100 to 5000 by 10 steps.

The Central Limit theorem states that in case of paths are independent, the sum of the option price on each path in the Monte Carlo approximation can be estimated by a normal distribution $\mathscr{N}(0,\frac{\sigma}{\sqrt{M}})$ (unbiased Monte Carlo estimator). Hence, $\frac{U_0^M}{\sigma/\sqrt{M}} \sim \mathscr{N}(0,1)$. In the same pattern of [21], we can propose an estimator of the theoretical tolerance range with a confidence level according to its upper and lower bounds

$$\hat{I}_{model} = \frac{upper^{M}_{model} - lower^{M}_{model}}{2} = 1.96 * \frac{\sigma}{\sqrt{M}}.$$

For example, for $M = 10000$, $\hat{I}_{levy} = \frac{21.8678-21.7970}{2} = 0.0354$, which is a very small range of error. Therefore, the option price theoretically belongs to $[21.8678 - 0.0354, 21.7970 + 0.0354]$ with a confidence level of 95%.

Table 2: Time consuming (MATLAB R2019a program/Processor: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz (8 CPUs), 2.0GHz/Memory: 8192MB RAM.

number of sampling	RF	least square
100 to 1000 by 10 steps	400.2 s	40.544 s
100 to 5000 by 10 steps	11634.54 s	1163.004 s
100 to 10000 by 10 steps	64800s	7200 s

Now the question that arises is whether RF-RI or least squares is more effective in predicting American option prices in a real market context.

4.4 Example of an American put option on Microsoft stock

Table 3: The least square algorithm compared to RF regression on the MSFT put option traded on 19/03/2021 at price $S_0 = 231.02$. Expired on 18/03/2021 at table prices, $r = 4\%$ from Yahoo Finance.

Strike	Price	Vol	RF	RF error	LS	LS error	λ_1	λ_2
185	9.81	6.25%	5.4672	4.3428	5.2048	4.6052	5	1/8
210	17.8	1.56%	15.3825	2.4175	15.2053	2.5947	5	1/7
215	19.8	1.56%	19.4275	0.3725	19.3315	0.4685	5	1/7
220	21.95	0.78%	18.8531	3.0969	19.4438	2.5062	5	1/6
225	24.25	0.78%	23.2182	1.0318	23.5296	0.7204	5	1/6
230	26.6	0.05%	23.2814	3.3186	23.4150	3.185	5	1/5
255	41.48	0%	42.4939	1.0139	42.4125	0.9325	5	1/4
270	51.45	0%	47.6928	3.7572	47.3619	4.0881	5	1/2
275	55.2	0%	52.6691	2.5309	52.3785	2.8215	5	1/2
305	80.53	0%	79.6319	0.8981	81.2268	0.6968	5	1/1.5
355	126.68	0%	128.1305	1.4505	126.9668	0.2868	5	1/1.5
			Mean error	2.202790909	Mean error	2.082336364		

According to Table 3, we will compare the least squares regression and the RF hybrid algorithm by considering an example of actual data. For this purpose, we extracted market information ($r = 4\%$, implied volatility) and stock information (the MSFT put option traded on 19/03/2021 at price $S_0 =$ 231.02$, expired on 18/03/2021) from Yahoo Finance. On the other hand, the parameters of the Levy Jump-diffusion are calibrated as shown in Table 3 (λ_1 is the negative jump size approximated by the weekly mean losses of Microsoft stock between 2020 and 2021 and λ_2 approximated by the occurrence of the biggest losses which are higher than the half-price average in one year). Then, similarly to Subsection 4.1 we simulate the forward price of the underlying under Levy Jump-diffusion and the backward least squares model as presented in Subsection 4.2, as well as the RF hybrid algorithm simulated as explained in Subsection 4.3. Hence, we compare the predicted price for each model to the price of the put options quoted in the market.

As we notice in Table 3, for all strikes, the RF option price is quite similar to the least squares option price but slightly higher in terms of the final mean error which equals $\frac{ME_{RF}-ME_{LS}}{\max(ME_{RF}, ME_{LS})} = 5\%$ (absolute variation in terms of mean error). Indeed, in this specific example, the least squares algorithm predicts slightly better than the RF regression by using the calibration of parameters, as shown in Table 3. The Gauss-Markov theory states that in a linear model in which the errors have zero expectations, uncorrelated and whose variances are equal. Then, the least squares estimator is the best and most unbiased linear estimator of the coefficients. This theorem in addition to the previous assumptions explain our results in the context of linear projection in L^2 space. It is commonly believed that the least squares algorithm is a convenient benchmark widely used by practitioners to value American options under normal market conditions (no big crash, without a high-multidimensional and correlated portfolios). However, our results show (for this specific example) that the RF regression is close to the efficiency of this benchmark estimator with a 5% gap. Furthermore, RF generally performs better in the context of nonlinear and highly-correlated multidimensional models due to its randomized tree structure. As a result, we are motivated to extend our application to nonlinear estimations of the continuation value in a future study e.g., multi-factor systematic risk and multidimensional underlying.

Remark 2. *The use of the improved Levy Jump-diffusion model with hybrid jumps (loss and profit jumps), like Kou's model [17] (i.e. jumps follow a generalized Laplace non-symmetric law) can slightly improve our results.*

5 Conclusion

In this paper, we have presented a new approach for pricing the American option under the exponential Levy Jump-diffusion model. We also compared our RF hybrid algorithm to the classical Longstaff and Schwartz algorithm with the least squares regression as a benchmark in terms of time and numerical stability of its convergence according to the number of simulated trajectories. We have also proposed a numerical method to emphasize the numerical stability of the proposed hybrid RF algorithm by estimating the theoretical confidence interval according to the confidence level in Section 4.3. Furthermore, we noted that by varying the number of paths, we numerically highlight the equilibrium option price. Note that in our algorithm, the trees are built independently and can therefore be built in parallel. this is a huge advantage for modern computing methods such as parallelization on GPU/Graphics Clusters, henceforth, we can get better exercise policies without paying for execution. More calibrations can perform better with the exponential Levy Jump-diffusion model as it has been done in the context of S&P100 European options in [25]. Recent studies on machine learning and parallelism methods for solving Deep BSDEs open new horizons for pricing American option as in [4] or [19].

References

[1] D.J. Amit, N. Brunel, *Dynamics of a recurrent network of spiking neurons before and following learning*, Netw. Comput. Neural Syst. **8** (2021) 373-404.

[2] P. Auer, N. Cesa-Bianchi, Y. Freund, R.E. Schapire, *Gambling in a rigged casino: The adversarial multi-armed bandit problem*, in Proceedings of IEEE 36th annual foundations of computer science, IEEE, (1995) 322-331.

[3] J. Barraquand, D. Martineau, *Numerical valuation of high dimensional multivariate american securities*, J. Financial Quant. Anal. **30** (1995) 383-405.

[4] S. Becker, A. Jentzen, M.S. Muller, P.V. Wurstemberger, *Learning the random variables in monte carlo simulations with stochastic gradient descent: Machine learning for parametric pdes and financial derivative pricing*, 2022, arXiv preprint arXiv:2202.02717.

[5] T. Bjork, *Arbitrage Theory in Continuous Time*, Oxford University Press, 2009.

[6] R. Breen, *The accelerated binomial option pricing model*, J. Financial Quant. Anal. **29** (1991) 153-164.

[7] L. Breiman, *Bagging predictors*, Mach. Learn. **24** (1996) 123-140.

[8] L. Breiman, *Bias, variance, and arcing classifiers*, Tech. Rep. 460, University of California, Berkeley, CA, 1996.

[9] L. Breiman, *Random forests*, Mach. Learn. **45** (2001) 5-32.

[10] M. Broadie, P. Glasserman, *Pricing american-style securities using simulation*, J. Econ. Dyn. Control. **21** (1997) 1323-1352.

[11] E. Clement, D. Lamberton, P. Protter, *An analysis of a least squares regression method for american option pricing*, Finance Stoch. **6** (2002) 449-471.

[12] T.G. Dietterich, *Ensemble methods in machine learning*, in International workshop on multiple classifier systems, Springer, (2000) 1-15.

[13] T.K. Ho, *The random subspace method for constructing decision forests*, IEEE Trans. Pattern Anal. Mach. Intell. **20** (1998) 832-844.

[14] J. Hull, A. White, *Valuing derivative securities using the explicit finite difference method*, J. Financial Quant. Anal. (1990) 87-100.

[15] S. Ken-Iti, *Levy Processes and Infinitely Divisible Distributions*, Cambridge University Press, 1999.

[16] E.B. Kong, T.G. Dietterich, *Error-correcting output coding corrects bias and variance*, in Machine learning proceedings 1995, Elsevier, (1995) 313-321.

[17] S.G. Kou, *A jump-diffusion model for option pricing*, Manage. Sci. **48** (2002) 1086-1101.

[18] S.W. Kwok, C. Carter, *Multiple decision trees*, Mach. Intell. Patt. Rec. **9** (1990) 327—335.

[19] J. Liang, Z. Xu, P. Li, *Deep learning-based least squares forward-backward stochastic differential equation solver for high-dimensional derivative pricing*, Quant. Finance **21** (2021) 1309-1323.

[20] F.A. Longstaff, E.S. Schwartz, *Valuing american options by simulation: a simple least-squares approach*, Rev. Financ Stud. **14** (2001) 113-147.

[21] M. Maidoumi, B. Daafi, M. Zahid, *An analysis of asymptotic properties and error control under the exponential jump-diffusion model for American option pricing*, J. Appl. Math. **2021** (2021).

[22] J.L. Marchand, *Conditionnement de processus markoviens*, PhD thesis, Université Rennes 1, Rennes, 2012.

[23] R.C. Merton, *Option pricing when underlying stock returns are discontinuous*, J. Financ. Econ. **3** (1976) 125-144.

[24] A. Pascucci, *PDE and Martingale Methods in Option Pricing*, Springer Science & Business Media, 2011.

[25] F.M. Quittard-Pinon, R. Randrianarivony, *Calibrage d'options pour trois modles mixtes diffusions et sauts*, Finance **29** (2008) 103–130.

[26] L. Stentoft, *Convergence of the least squares Monte Carlo approach to American option valuation*, Manage. Sci. **50** (2004) 1193–1203.

[27] J.N. Tsitsiklis, B.V. Roy, *Regression methods for pricing complex American-style options*, IEEE Trans. Neural. Netw. Learn. Syst. **12** (2001) 694-703.