Neural Networks for extracting implied information from American options

Shuaiqiang Liu, Álvaro Leitao, Anastasia Borovykh and Cornelis W. Oosterlee

26th CEDYA / 16th CMA

June 14, 2021

Motivation and proposal

- Extracting implied information, like volatility and/or dividend, from observed option prices is a challenging task when dealing with early-exercise options.
- Computing American-style option prices is generally more expensive than pricing European-style options.
- The optimization process to address the inverse problem traditionally requires solving the pricing model many thousands of times.
- Other complicating factors, such as a negative interest rate or dividend yield may lead to complex-shaped early-exercise regions.
- To determine the implied volatility, the inverse function is approximated by an artificial neural network.
- This decouples the offline (training) and online (prediction) phases and thus eliminates the need for an iterative process.
- For the implied dividend yield, we formulate the inverse problem as a calibration problem and determine simultaneously the implied volatility and dividend yield.

Outline

Problem formulation

- Although other pricing models would easily fit into our framework, we consider the Geometric Brownian Motion (GBM) process,

$$dS_t = (r - q)S_t \mathrm{d}t + \sigma S_t \mathrm{d}W_t, \tag{1}$$

  where S_t is the underlying spot price, and σ is the volatility parameter and W_t a Wiener process.

- The arbitrage-free value of an American option at time t is given by

$$V_{am}(S_t, t) = \sup_{u \in [0,T]} \mathbb{E}_t^{\mathbb{Q}}[e^{-r(T-t)} H(K, S_u) | S_u], \tag{2}$$

  where $H(\cdot)$ is the payoff function with strike price K and expiration time is T.

Problem formulation

- An optimal exercise boundary $S_t^* \equiv S^*(t)$, which depends on the time to maturity $T-t$, divides the domain into early-exercise (stopping) regions Ω_s and continuation (or holding) regions Ω_h.
- As an American option can be exercised anytime before the expiration time, a corresponding early-exercise premium should be added to the European option counterpart,

$$\begin{aligned} V_{am}^P(S_t, t) = \mathbb{E}_t^{\mathbb{Q}}[e^{-r(T-t)} \max(K - S_T, 0)] \\ + \int_t^T \mathbb{E}_u^{\mathbb{Q}}[(rK - qS_u)\mathbf{1}_{\{S_u \in \Omega_s\}}]du, \end{aligned} \tag{3}$$

  where Ω_s represents the stopping region, and the whole domain is $\Omega = \Omega_s + \Omega_h$ with Ω_h being the holding region.
- At the free boundary, we have,

$$V_{am}(S^*, t) = H(K, S_t^*), \quad \frac{\partial V_{am}}{\partial S} = -1. \tag{4}$$

Negative interest rates and dividends

- Two continuation regions may arise when both the interest rate and dividend yield become negative.

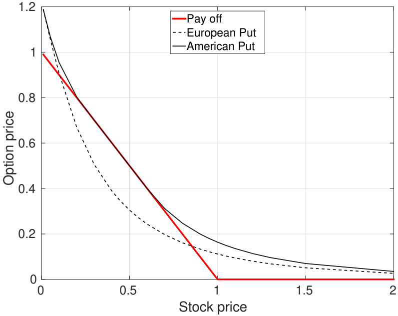


Figure: The setting: $r = -0.01$, $q = -0.06$, $\sigma = 0.2$, $T = 20$, $K = 1.0$. The option value in the solid black line hits the payoff function twice. The stopping region is between the two early-exercise points.

Implied volatility

- The implied volatility is the level of volatility which, inserted in the pricing model, makes the market and model prices match.
- Implied volatility is an indication for the future uncertainty of the underlying asset prices as estimated by market participants.
- Computing the implied volatility can be seen as an inverse problem:

$$\sigma^* = BS_{am}^{-1}(V_{am}^{mkt}; S, K, \tau, r, q, \alpha), \tag{5}$$

  where $BS_{am}^{-1}(\cdot)$ denotes the inversion of the Black-Scholes formula, and V_{am}^{mkt} is an American option price observed in the market.
- The implied volatility inverse problem is often solved by a nonlinear root-finding method, following an iterative algorithm.
- Given an American option price observed in the market, the implied volatility σ^* is determined by

$$V_{am}^{mkt} - BS_{am}(\sigma^*; S_0, K, \tau, r, q, \alpha) = 0. \tag{6}$$

Issues in computing implied volatility (I)

- Existence of σ^* can be guaranteed by the monotonicity of the Black-Scholes equation w.r.t to the volatility in the holding region.
- However, a closed-from expression for the derivative of the American option value with respect to the volatility is not available.
- Gradient-free methods, like bisection, may converge slowly because of the stopping regions.
- The option's Vega, becomes zero in the stopping region for American call and put options. It is well-known that,

$$|\Delta| = |\frac{\partial V_{am}}{\partial S}| = 1, \quad \text{Vega} = \frac{\partial V_{am}}{\partial \sigma} = 0. \tag{7}$$

- In other words, the American option prices do not depend on the volatility in the stopping regions. Consequently,

$$\frac{\partial \sigma}{\partial V_{am}} = \frac{1}{\text{Vega}} \rightarrow \infty.$$

Issues in computing implied volatility (II)

- When we invert the American Black-Scholes pricing problem in the stopping regions, there is no unique solution for the implied volatility.
- Therefore, the definition domain of Formula (5) should be the continuation region.

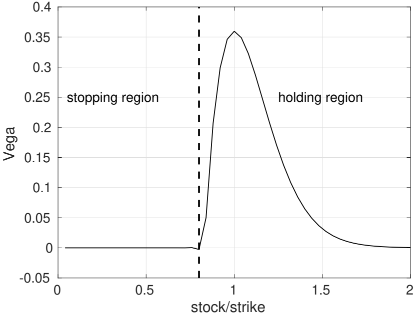


Figure: The Vega for American puts in different regions.

Implied dividend

- Many companies pay a share of the stock value to the share holder on the ex-dividend date, which causes the stock price to drop.
- This quantity is often called the actual dividend.
- The *Implied dividend* reflects how the market anticipates future dividend payments of stocks.
- The difference between actual dividends and implied dividends is similar to that between historical and implied volatility. The two parameters reflect different market aspects.
- Some companies do not pay dividends, but the corresponding options may imply a non-zero dividend, which may reflect the borrowing level of the stock.
- The borrowing costs are seen as a factor that influences the implied dividend as a function of the time or the strike price.

Issues computing implied dividend (I)

- Our approach is to estimate implied dividend and implied volatility at once assuming the implied dividend is not constant over strike prices.
- In the case of European stock options, the implied dividend can be estimated by the put-call parity,

$$V_{eu}^{C}(S,t) - V_{eu}^{P}(S,t) = S_t e^{-q\tau} - Ke^{-r\tau}, \tag{8}$$

so that,

$$q = -\frac{1}{\tau}\log\left(\frac{V_{eu}^{C} - V_{eu}^{P} + Ke^{-r\tau}}{S_t}\right). \tag{9}$$

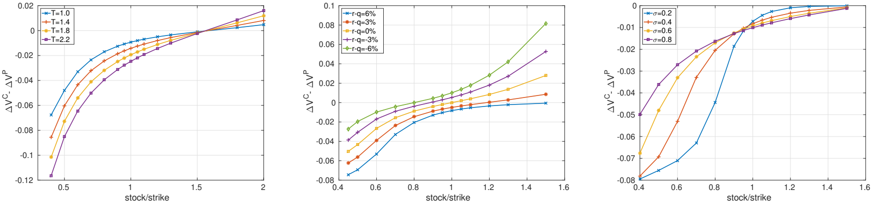
- For American-style options, the put-call parity does not hold.
- Writting in terms of the early-exercise premium, denoted by ΔV^C and ΔV^P,

$$V_{am}^{C} = V_{eu}^{C} + \Delta V^{C} \text{ and } V_{am}^{P} = V_{eu}^{P} + \Delta V^{P}.$$

Issues computing implied dividend (II)

- The “deviation” from the European put-call parity can be written as

$$\text{EED} := \Delta V^C - \Delta V^P = V^C_{am} - V^P_{am} - S_t e^{-q\tau} + K e^{-r\tau}. \tag{10}$$



(a) EED over maturity T (b) EED against $r-q$ (c) EED against volatility

Figure: Left: EED over maturity time T, with $\sigma = 0.4$, $r = 0.1$, $q = 0.05$, $K = 1.0$. Middle: EED against $r-q$, with $T = 1.0$, $r = 0.1$, $\sigma = 0.4$, $K = 1.0$. Right: EEP against volatility σ, with $T = 1.0$, $r = 0.1$, $q = 0.04$, $K = 1.0$.

Computing the implied dividend

- Again, we will employ a model-based approach: Black-Scholes model
- The dividend yield is inverted in order to extract the implied dividend.
- Point of departure is the American call and put model prices:

$$\begin{cases} V_{am}^{C} = BS_{am}(\sigma^*, q^*; S_0, K, \tau, r, \alpha = 1), \\ V_{am}^{P} = BS_{am}(\sigma^*, q^*; S_0, K, \tau, r, \alpha = -1), \end{cases} \tag{11}$$

- Assuming the implied volatility and the implied dividend are the same for calls and puts with the same parameter set K, S_0, τ and r, there appears to be a unique solution (two equations and two unknowns).
- This is not always the case, as option prices do not depend on the volatility in the stopping regions.
- The system of equations is usually formulated as a minimization problem and a numerical optimization algorithm is employed to search the solution space.
- Note that a local search based optimization method will most likely not converge when traversing those early-exercise regions.

Artificial Neural Networks (ANN)

- ANNs are powerful function approximators.
- An ANN can be described as a composite function,

$$\mathrm{F}(\mathrm{x}|\boldsymbol{\theta}) = f^{(\ell)}(\ldots f^{(2)}(f^{(1)}(\mathrm{x}; \boldsymbol{\theta}^{(1)}); \boldsymbol{\theta}^{(2)}); \ldots \boldsymbol{\theta}^{(\ell)}), \tag{12}$$

  where x stands for the input variables, $\boldsymbol{\theta}$ for the hidden parameters (i.e. the weights and the biases in artificial neurons), ℓ for the total number of hidden layers, and $f^{(\ell)}(\cdot)$ represents a hidden-layer function.
- The composite function, $\mathrm{F}(\cdot)$, depends on these hidden parameters and activation/transfer functions.
- Once the structure is determined, an ANN becomes a deterministic function.
- And once trained, the the evaluation of F is super fast.

Training an ANN

- Determining the values of the hidden parameters which will minimize a loss function.
- A popular approach for training neural networks is to employ first-order optimization algorithms.
- Gradient-based algorithms are often fast, but it may be difficult to calculate the gradients for a large test set.
- Stochastic gradient descent algorithms (SGD) randomly select a portion of the data set (saving memory).
- SGD and its variants (like Adam) are thus preferable to train the ANNs on big data sets.
- In supervised learning, the objective function is

$$\arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}|(\mathrm{X}, \mathrm{Y})), \tag{13}$$

given the input-output pairs (X, Y) and a user-defined loss, $L(\boldsymbol{\theta})$.

ANN for implied volatility

- The idea is to use the ANN to approximate the inverse function in Ω_h,

$$\begin{aligned} \sigma^* &= BS_{am}^{-1}(V_{am}^{mkt}; S, K, \tau, r, q, \alpha) \\ &\approx \mathrm{NN}(V_{am}^{mkt}; S, K, \tau, r, q, \alpha), \quad [V, S, K, \tau, r, q] \in \Omega_h. \end{aligned} \tag{14}$$

- The ANN must be trained based on the known market variables to approximate the unique target variable σ^*.
- The effective definition domain Ω_h corresponds to the continuation regions.
- The continuation regions are not known initially or are so complicated that there is no analytic formula to describe them.
- However, the counterpart, the early-exercise regions, can be found implicitly from the data.

Computing the continuation region

- Then, we obtain the approximate continuation region by $\Omega_h = \Omega - \Omega_s$.
- Two indicators to detect the samples in the early-exercise region, Ω_s:
  - The difference between the option value and the payoff.
  - The option's sensitivity Vega.
- To control numerical errors, threshold values, ϵ_1 and ϵ_2, are prescribed for the two indicators.

$$|V_{am}(S_t, K, \tau, r, q, \sigma) - H(K, S_t)| > \epsilon_1. \tag{15}$$

$$\text{Vega} > \epsilon_2. \tag{16}$$

- In principle, Criterion (15) should cover Criterion (16), but for robustness reasons, both will be enforced.
- The procedure is done off-line, prior the training phase, by filtering out the samples according to the criteria.

Continuation region

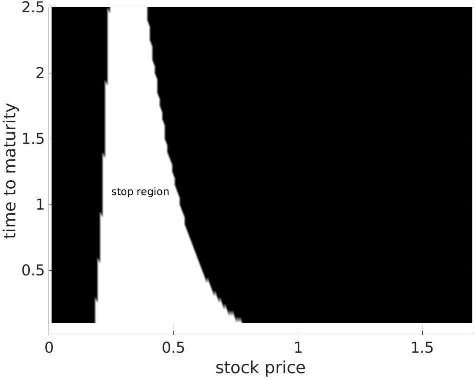


Figure: Schematic diagram: An example of two continuation regions for an American put. The shaded area represents the holding region, while the white area represents the stopping region. There are two isolated continuation regions. Here the strike price is fixed $K = 1$.

Gradient-squashing of the option prices

- ANNs can be inaccurate when functions with steep gradients.
- We employ the gradient-squashing technique for the option prices.
- We subtract the intrinsic value from the American option price to obtain the corresponding *time value*.
- Recall the put-call parity for European options,

$$V_{eu}^{C} - S_t e^{-q\tau} = V_{eu}^{P} - K e^{-r\tau}.$$

- The following lower bound can be deduced,

$$V_{eu}^{P}(S_t, t) = V_{eu}^{C}(S_t, t) + K e^{-r\tau} - S_t e^{-q\tau} \geq K e^{-r\tau} - S_t e^{-q\tau}, \quad (17)$$

- As an American option is at least as expensive as European option

$$V_{am}^{P}(S_t, t) \geq V_{eu}^{P}(S_t, t) \geq K e^{-r\tau} - S_t e^{-q\tau}, \quad (18)$$

- An it should not be worth less than the pay-off function at any time,

$$\hat{V}_{am}^{P} = V_{am}^{P}(S_t, t) - \max(K - S_t, K e^{-r\tau} - S_t e^{-q\tau}, 0). \quad (19)$$

- The gradient squashing consist of taking the logarithm of the time value, i.e. $\log(\hat{V}_{am}^{P})$.

ANN for implied information

- Determining both implied volatility and implied dividend simultaneously.
- We assume the implied volatility and the implied dividend are identical for American calls and puts with the same K, S_0, T, t and r:

$$\begin{cases} V_{am}^{C,mkt} - BS_{am}(\sigma^*, q^*; S_0, K, \tau, r, \alpha = 1) = 0, \\ V_{am}^{P,mkt} - BS_{am}(\sigma^*, q^*; S_0, K, \tau, r, \alpha = -1) = 0, \end{cases} \tag{20}$$

- There are two unknown parameters to calibrate, implied volatility σ^* and the implied dividend yield q^*, given a pair of American option prices, $V_{am}^{C,mkt}$ and $V_{am}^{P,mkt}$.
- The above system is reformulated as a minimization problem,

$$\arg \min_{\sigma^* \in R^+, q^* \in R} (BS_{am}(\sigma^*, q^*; \alpha = 1) - V_{am}^{C,mkt})^2 + (BS_{am}(\sigma^*, q^*; \alpha = -1) - V_{am}^{P,mkt})^2.$$

- The challenges include the American option pricing method and the region with Vega=0 (where the optimization convergence may be hampered).

CaNN for implied information

- We adapt a fast, generic and robust calibration framework, the CaNN (Calibration Neural Networks) developed in [1].
- The basic idea of the methodology is to convert the calibration of model parameters into an estimation of a neural network's hidden units.
- The model calibration and training ANNs can be reduced to solving an optimization problem.
- It enables parallel GPU computing to speed up the computations, which makes feasible to employ a global optimization technique to search the solution space.
- Here, we employ the gradient-free optimization algorithm, Differential Evolution (DE) which does not get stuck in local minima or in the stopping region.
- And DE is an inherently parallel technique.

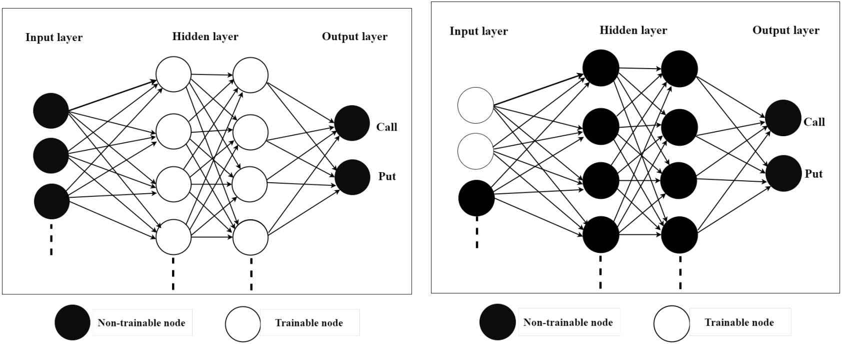
CaNN for implied information

- The CaNN consists of two stages:
  - The forward pass, including the training and testing phase, approximates the American-style Black-Scholes prices.
  - The backward pass, on the other hand, aims to find the two parameters, (σ^*, q^*), to match the two observed American option prices, $V_{am}^{P,mkt}$ and $V_{am}^{C,mkt}$, with strike price K, maturity time T, spot price S_0, interest rate r.
- We have developed one neural network providing two output values, the American call and put prices.
- Then, the objective function of model calibration can be approximated by

$$\arg \min_{\sigma^* \in R^+, q^* \in R} (\mathrm{NN}(\sigma^*, q^*; \alpha = 1) - V_{am}^{C,mkt})^2 + (\mathrm{NN}(\sigma^*, q^*; \alpha = -1) - V_{am}^{P,mkt})^2,$$

which is used as the loss function for the backward pass in the CaNN.

CaNN structure



(a) Training phase (b) Calibration phase

Figure: Left: In the forward pass of the CaNN, the output layer produces two option prices. Right: In the calibration phase, the CaNN estimates the two parameters, implied volatility and implied dividend, in the original input layer.

The ANN design and training set

- We find a balance between representation power and efficiency with the following configuration:

Table: The ANN configuration.

Hyper-parameter	**Value**
Hidden layers	4
Neurons (each layer)	200
Activation	Softplus
Initialization	Glorot_uniform
Optimizer	Adam
Batch size	1024

- Other useful operations for deep NN (dropout or batch normalization) do not bring any significant benefits in our "shallow" ANN.
- Samples for the training/test set are generated by the COS method.

Settings for computing implied volatility

- Without loss of generality, we use a fixed spot price $S_0 = 1.0$.
- The two thresholds are set to $\epsilon_1 = 0.0001$ and $\epsilon_2 = 0.001$.

Table: Train dataset for American options under the Black-Scholes model; The spot price $S_0 = 1$ is fixed. The upper bound of American put price is 1.2. LHS stands for Latin Hypercube Sampling.

ANN	Parameters	Value range	Employed method
ANN Input	Strike, K	[0.6, 1.4]	LHS
	Time value, $\log(\hat{V}^P_{am})$	$(-11.51, -0.24)$	COS
	Time to maturity, τ	[0.05, 3.0]	LHS
	Interest rate, r	[-0.05, 0.1]	LHS
	Dividend yield, q	[-0.05, 0.1]	LHS
ANN output	Implied volatility, σ^*	$(0.01, 1.05)$	LHS

- We consider the measures

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \quad \text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|, \quad \text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i}.$$

Numerical results: implied volatility

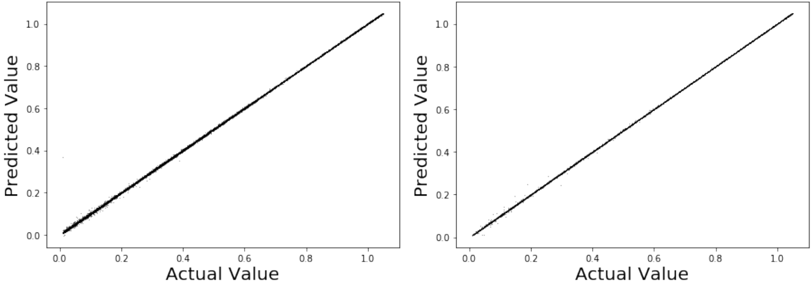
- The training preformace:

Table: Multiple measures are used to evaluate the performance.

-	MSE	MAE	MAPE	R^2
Training	$4.33 \cdot 10^{-7}$	$2.44 \cdot 10^{-4}$	$1.11 \cdot 10^{-3}$	0.999994
Testing	$4.60 \cdot 10^{-7}$	$2.51 \cdot 10^{-4}$	$1.15 \cdot 10^{-3}$	0.999993

- The test performance is close to the train performance, suggesting that the trained ANN generalizes well for unseen data
- The ANN predicted implied volatility values approximate the true values accurately for both the train and test datasets, as is indicated by the R^2 measure.

- It is observed that the trained model performance tends to decrease when the pricing model parameters gets close to the upper or lower bounds
- Thus the training data set is recommended to have a wider parameter range than the test range of interest.



(a) Training

(b) Testing

Figure: Left: R^2=0.999994; Right: R^2=0.999993

Settings for computing implied information

- The loss function includes two components:

$$\text{MSE} = \frac{1}{2n}\sum_{i=1}^{n}\{(\tilde{V}^{P}_{am,i} - V^{P,mod}_{am,i})^2 + (\tilde{V}^{C}_{am,i} - V^{C,mod}_{am,i})^2\}. \qquad (21)$$

Table: Training data set for the forward pass. We fix $S_0 = 1$, and sample strike prices K to generate different moneyness levels. The total number of the data samples is nearly one million, with 80% training, 10% validation, 10% test samples.

ANN	Parameters	Value range	Method
Forward input	Strike, K	[0.45, 1.55]	LHS
	Time to maturity, τ	[0.08, 3.05]	LHS
	Risk-free rate, r	[-0.1, 0.25]	LHS
	Dividend yield, q	[-0.1, 0.25]	LHS
	Implied volatility, σ	$(0.01, 1.05)$	LHS
Forward output	American put, V^P_{am}	$(0, 1.8)$	COS
	American call, V^C_{am}	$(0, 1.2)$	COS

CaNN: performance of the forward pass:

- Performance of the forward pass:

Table: The performance of the CaNN forward pass with two outputs.

–	Option	MSE	MAE	MAPE	R^2
Training	Call	1.40×10^{-7}	3.00×10^{-4}	1.25×10^{-3}	0.9999965
	Put	2.54×10^{-7}	4.24×10^{-4}	1.64×10^{-3}	0.9999959
Testing	Call	1.43×10^{-7}	3.02×10^{-4}	1.27×10^{-3}	0.9999964
	Put	2.55×10^{-7}	4.26×10^{-4}	1.64×10^{-3}	0.9999959

- The results, for both Calls and Puts, are highly satisfactory, achieving very good levels of precision in all the considered measures.

CaNN: performance of the backward pass:

- The results suggest that the CaNN can accurately recover the implied volatility and implied dividend from "artificial market option data".
- Even in complex scenarios (negative interest rates and/or dividend yields), CaNN recovers the true values without stalling convergence.
- The method's robustness may be attributed to the robust numerical solver and the gradient-free optimizer.

Table: Using CaNN to extract implied volatility and implied dividend. † indicates the prescribed values, $*$ indicates the calibrated values.

K/S_0	T	r	$\sigma^\dagger$	$q^\dagger$	C_{am}^{mkt}	P_{am}^{mkt}	σ^*	q^*
1.0	0.5	-0.04	0.1	0.06	0.0146	0.0597	0.099	0.059
1.1	0.5	-0.04	0.2	-0.06	0.0255	0.1181	0.198	-0.061
1.0	0.75	0.0	0.3	-0.02	0.1119	0.0976	0.300	-0.020
1.2	1.0	-0.04	0.4	0.08	0.0603	0.3810	0.40	0.080
0.8	1.0	0.02	0.3	0.02	0.2322	0.03472	0.299	0.020
0.7	1.25	0.0	0.4	-0.04	0.3886	0.0378	0.399	-0.040

Conclusions

- We studied a data-driven method to extract the implied volatility and/or implied dividend yield from observed market American option prices in a fast and robust way.
- For computing the American implied volatility, we propose a sophisticated ANN to approximate the inverse problem.
- The problem domain is extracted from the data, preserving the ANN offline-online decoupling advantage.
- We also propose a method for finding simultaneously implied dividend and implied volatility from American options using a calibration approach.
- The numerical experiments demonstrate that the CaNN is able to accurately extract multiple pieces of implied information from American options.
- It should be feasible to extend the approach to deal with time-dependent or discrete dividends.

References

Shuaiqiang Liu, Anastasia Borovykh, Lech A. Grzelak, and Cornelis W. Oosterlee.
A neural network-based framework for financial model calibration.
Journal of Mathematics in Industry, 9(1):9, 2019.

Shuaiqiang Liu, Álvaro Leitao, Anastasia Borovykh, and Cornelis W. Oosterlee.
On Calibration Neural Networks for extracting implied information from American options, 2020.
Available at arXiv: https://arxiv.org/abs/2001.11786.

Bowen Zhang and Cornelis W. Oosterlee.
Efficient pricing of European-style Asian options under exponential Lévy processes based on Fourier cosine expansions.
SIAM Journal on Financial Mathematics, 4(1):399–426, 2013.

Acknowledgements & Questions

Thanks to the support received from the Centro de Investigación de Galicia "CITIC", funded by Xunta de Galicia and the European Union (European Regional Development Fund- Galicia 2014-2020 Program), by grant ED431G 2019/01 and support from the Spanish Ministry of Science, Innovation and Universities, through the Juan de la Cierva-formación 2017 (FJC17).

More: `alvaro.leitao@udc.gal` and `alvaroleitao.github.io`

Thank you for your attention