Deep Learning Algorithms for Solving the Black-Scholes Equation: A Comparative Study

ELBAYED Zakaria* and Abdelmjid Qadi el Idrissi [1].

Laboratory for Analysis and Modeling of Systems and Decision Support, Center for Doctoral Studies in Science and Technology and Medical Sciences, Hassan First University, National School of Applied Sciences Berrchid , Morocco

Z.elbayed.doc@uhp.ac.ma , qadi.majid@gmail.com

Abstract. The Black-Scholes model is widely used for pricing European options, but its simplifying assumptions, such as constant volatility and the absence of transaction costs, limit its accuracy in real-world market conditions. This paper explores alternative deep learning methods for solving the Black-Scholes partial differential equation (PDE). Specifically, we compare the performance of Physics-Informed Neural Networks (PINNs), Fourier Neural Operators (FNOs), Long Short-Term Memory (LSTM) networks, Deep Galerkin Method (DGM), and Extreme Learning Machine (ELM). Each method is evaluated based on accuracy, convergence speed, computational complexity, scalability, memory requirements, and robustness to noisy data. The results demonstrate that deep learning approaches can significantly improve option pricing accuracy in volatile markets, while maintaining computational efficiency. This comprehensive comparison highlights the strengths and weaknesses of each algorithm, providing insight into their practical applications in financial modeling.

Keywords: Black-Scholes model, deep learning, Physics-Informed Neural Networks (PINNs), Fourier Neural Operator (FNO), Long Short-Term Memory (LSTM), Deep Galerkin Method (DGM), Extreme Learning Machine (ELM), option pricing, partial differential equations (PDEs), computational finance, scalability, robustness.

1 Introduction

Partial differential equations (PDEs) are fundamental in financial modeling, particularly for pricing derivatives such as options. The Black-Scholes model, introduced in 1973, remains one of the most widely used frameworks for pricing European options. It is based on several simplifying assumptions, including constant volatility, no transaction costs, and a fixed risk-free interest rate. While these assumptions allow for an analytical solution to the Black-Scholes equation, they often fail to reflect the complexities of real financial markets, especially in the presence of fluctuating volatility.

Traditional numerical methods, such as finite differences and finite elements, have been extensively used to solve the Black-Scholes equation. However, these methods struggle with scalability and computational complexity, particularly when addressing high-dimensional problems or rapidly changing market conditions. This has led to the exploration of more advanced techniques, notably those based on deep learning.

Deep learning models, such as Physics-Informed Neural Networks (PINNs), Fourier Neural Operators (FNOs), and Long Short-Term Memory (LSTM) networks, have emerged as promising alternatives for solving PDEs in financial contexts. These models offer the ability to capture complex nonlinear dynamics and provide flexible frameworks for modeling option prices under varying market conditions. Recent developments in deep learning, such as the Deep Galerkin Method (DGM) and Extreme Learning Machine (ELM), further expand the range of available tools for solving the Black-Scholes equation, offering faster convergence and simpler implementation.

In this paper, we perform a comparative study of five deep learning algorithms—PINNs, FNOs, LSTMs, DGMs, and ELMs—for solving the Black-Scholes equation. We evaluate these models based on several criteria, including accuracy, convergence speed, computational complexity, scalability, memory requirements, and robustness to noisy data. Additionally, we explore each model's ability to handle more complex financial products, such as exotic options, which present additional challenges in pricing due to their path-dependent features.

The goal of this study is to provide a comprehensive analysis of the strengths and weaknesses of each algorithm, offering insights into their practical applications for option pricing and risk management in financial markets.

2 Literature review

Traditional methods such as finite differences and finite elements have long been used to solve the Black-Scholes partial differential equation (PDE). These approaches are effective for low-dimensional problems but face significant challenges when applied to high-dimensional scenarios or rapidly changing market conditions. Specifically, they often suffer from scalability issues, high computational costs, and limited flexibility in handling complex financial products like exotic options. These limitations have motivated the exploration of alternative approaches based on deep learning.

Physics-Informed Neural Networks (PINNs) incorporate the underlying physical laws, expressed as PDEs, directly into the loss function of the neural network. This allows the network to learn solutions that respect the PDE constraints, without the need for large amounts of labeled data. Raissi et al. (2019) demonstrated the effectiveness of PINNs for solving various forward and inverse PDE problems, including those in finance. While PINNs offer good accuracy for low-dimensional problems, they can be sensitive to hyperparameter tuning and may require long training times for complex scenarios.

The Fourier Neural Operator (FNO), introduced by Li et al. (2020), utilizes Fourier transformations to capture complex relationships in high-dimensional data. FNOs are particularly suitable for high-dimensional PDEs due to their scalability and computational efficiency. Unlike traditional neural networks, which rely on grid-based methods, FNOs are grid-independent, allowing for faster convergence in high-dimensional settings. While FNOs have been applied successfully in physics and engineering problems, their application in finance remains relatively new, but promising.

Long Short-Term Memory (LSTM) networks, a type of recurrent neural network (RNN), are widely used for modeling sequential data and time series. In finance, LSTMs are particularly effective for capturing the temporal dynamics of asset prices, interest rates, and volatility. However, LSTMs face challenges with long sequences, where performance can degrade due to vanishing gradients and computational complexity (Zhang et al., 2020). Their application in option pricing is limited by the need for large amounts of data to properly train the model over long time horizons.

The Deep Galerkin Method (DGM), developed by Sirignano and Spiliopoulos (2018), offers a deep learning-based approach for solving high-dimensional PDEs without the need for mesh discretization. DGMs use neural networks to approximate the solutions of PDEs and are particularly well-suited for problems in finance, where dimensionality and market conditions can vary widely. DGMs have demonstrated strong performance in pricing complex financial derivatives, such as path-dependent options, and are noted for their ability to handle high-dimensional problems more efficiently than traditional methods.

Extreme Learning Machines (ELMs), introduced by Huang et al. (2006), are a type of feedforward neural network with a single hidden layer. ELMs are known for their fast training times and simplicity in implementation. However, they tend to struggle with nonlinear and complex dynamic problems, making them less suitable for modeling financial derivatives under volatile market conditions. Despite their limitations, ELMs remain attractive for problems where computational speed is a priority, albeit at the cost of accuracy.

The literature reveals that while deep learning approaches offer significant improvements over traditional numerical methods, their performance depends on the specific characteristics of the problem being addressed. PINNs, FNOs, and DGMs are particularly well-suited for high-dimensional problems, whereas LSTMs excel in time-series forecasting. ELMs, although less accurate, provide fast and efficient solutions for simple problems. Each of these methods presents unique strengths and weaknesses that make them suitable for different financial modeling scenarios.

3 Methodology

This section outlines the approach taken to compare five deep learning algorithms used for solving the Black-Scholes equation. The algorithms compared are Physics-Informed Neural Networks (PINNs), Fourier Neural Operators (FNOs), Long Short-Term Memory (LSTM) networks, Deep Galerkin Method (DGM), and Extreme Learning Machine (ELM). These models are evaluated based on their performance as reported in existing literature, focusing on several key criteria, such as accuracy, convergence speed, computational complexity, scalability, and robustness to noisy data.

3.1 Sources of Data and Results

The comparison is based on the results and findings from previous studies in the field of financial modeling and deep learning. The performance of each algorithm—accuracy, convergence speed, and robustness—is derived from well-known works, such as Raissi et al. (2019) for PINNs, Li et al. (2020) for FNOs, and Sirignano and Spiliopoulos (2018) for DGM. Similarly, Huang et al. (2006) provide insights into the behavior of Extreme Learning Machines (ELM), while Zhang et al. (2020) address the application of LSTM networks in finance.

The results of these studies are synthesized into a qualitative comparison to highlight the strengths and limitations of each method in solving the Black-Scholes equation and pricing financial derivatives.

3.2 Deep Learning Algorithms Compared

We examine the performance of the following five deep learning algorithms:

Physics-Informed Neural Networks (PINNs) incorporate the governing physical equations directly into the loss function, enabling the model to learn solutions that satisfy the PDE constraints. While they show good results for low-dimensional problems, they can face challenges in handling more complex, high-dimensional PDEs.

Fourier Neural Operator (FNO) leverage Fourier transformations to model complex relationships in high-dimensional data without the need for grid-based discretization. They are notable for their rapid convergence and scalability, making them a promising choice for high-dimensional problems.

Long Short-Term Memory (LSTM), a type of recurrent neural network, are well-suited for modeling time-series data. They are commonly used to capture the temporal dynamics of asset prices and volatility. However, their computational complexity increases with sequence length, and performance can degrade when modeling long-term dependencies.

Deep Galerkin Method (DGM) are designed to solve high-dimensional PDEs without the need for mesh discretization. They use neural networks to approximate PDE solutions and are particularly effective for complex financial derivatives, such as exotic options.

Extreme Learning Machine (ELM) are feedforward neural networks with a single hidden layer, known for their fast training times and simplicity. While ELMs are quick to implement, they struggle with highly nonlinear dynamics, making them less suitable for complex financial problems.

3.3 Evaluation Criteria

The comparison of these algorithms is based on the following performance criteria, derived from the literature:

Accuracy: The ability of the model to predict option prices with minimal error.

Convergence Speed: The number of iterations or time required for the model to reach a stable solution.

Computational Complexity: The computational cost in terms of processing time and resources required to train each model.

Scalability: The model's capacity to scale effectively with increasing data or problem dimensionality.

Memory Requirements: The amount of memory needed to process large financial datasets.

Robustness to Noisy Data: The model's ability to handle noisy or incomplete data, a common issue in financial markets.

This study relies entirely on a qualitative review of existing literature. No direct experimentation or dataset generation was conducted. Instead, the strengths and weaknesses of each method are synthesized based on published results, providing a clear comparison of their suitability for solving the Black-Scholes equation under varying market conditions.

4 Results

This section presents a qualitative comparison of the performance of the five deep learning algorithms PINNs, FNO, LSTM, DGM, and ELM based on the key criteria identified in the methodology: accuracy, convergence speed, computational complexity, scalability, memory requirements, and robustness to noisy data. The results are synthesized from previous studies to highlight the strengths and weaknesses of each method in solving the Black-Scholes equation for option pricing.

4.1 Qualitative Comparison of Performance

The table below summarizes the performance of the five algorithms across multiple criteria based on findings from the literature:

Criterion	PINNs [8]	FNO [9]	LSTM [10]	DGM [11]	ELM [6]
Accuracy	High for low-dimensional PDEs but sensitive to hyperparameter tuning	Strong performance for high-dimensional problems	Effective for capturing temporal dynamics but degrades with long sequences	Highly accurate for solving complex financial PDEs	Fast but less accurate for nonlinear, complex problems
Convergence Speed	Slower convergence, requires significant tuning.	Fast convergence, particularly for high-dimensional problems.	Slower convergence for long sequences; can suffer from vanishing gradients.	Fast convergence for high-dimensional PDEs	Extremely fast due to simple architecture but lacks precision.
Computational Complexity	Moderate; training requires significant resources, especially for large problems.	Low complexity; highly scalable and efficient for large datasets.	High complexity due to the sequential nature of LSTMs, especially for long sequences.	Moderate complexity; computational resources scale with problem size.	Low complexity, but at the cost of reduced model flexibility.
Scalability	Limited scalability; performance degrades for higher-dimensional problems.	Highly scalable, well-suited for large-scale, high-dimensional problems	Scalability limited by sequence length and memory requirements.	Strong scalability; designed to handle high-dimensional PDEs	Scalable, but better suited for simpler, low-dimensional problems.
Memory Requirements	Moderate memory usage, but increases with problem dimensionality.	Low memory requirements, efficient for large datasets.	High memory usage for long temporal sequences	Moderate; memory requirements increase with problem complexity.	Low memory usage due to the simplicity of the architecture
Robustness to Noisy Data	Sensitive to noisy or incomplete data, requiring careful data preprocessing.	Robust to noisy data, performs well even with imperfect data.	Can be unstable when trained on noisy data.	Highly robust to noisy or uncertain data	Less robust; sensitive to data noise and outliers.

4.2 Analysis of Results

Based on the qualitative comparison, several key observations can be made:

PINNs offer strong accuracy for solving low-dimensional PDEs and provide a physically-informed approach, but they are sensitive to hyperparameters and require careful tuning for convergence. Their scalability is limited, making them less suitable for high-dimensional financial problems.

FNOs stand out for their ability to efficiently handle high-dimensional data. They offer fast convergence and low computational complexity, making them a promising candidate for large-scale problems in finance.

LSTM networks perform well when modeling time-series data, capturing temporal dynamics of asset prices effectively. However, their performance degrades with long sequences, and their complexity makes them resource-intensive, limiting their scalability in certain contexts.

DGM demonstrates excellent performance in solving high-dimensional financial PDEs, particularly in complex scenarios such as the pricing of exotic options. It offers both fast convergence and strong robustness to noisy data, making it highly suitable for real-world financial applications.

ELM, while exceptionally fast due to its simple architecture, struggles with the complex, nonlinear dynamics often present in financial markets. Its lack of robustness to noisy data and reduced accuracy make it less suitable for complex financial derivatives, though it remains an option when computational speed is a priority.

5 Discussion

5.1 Comparative Analysis

The results from the literature highlight that each deep learning algorithm—PINNs, FNO, LSTM, DGM, and ELM—offers distinct advantages depending on the specific financial modeling problem being addressed. When comparing their strengths and weaknesses, several key patterns emerge:

- Physics-Informed Neural Networks (PINNs) excel in capturing the physical laws governing PDEs, particularly for low-dimensional problems. However, their sensitivity to hyperparameter tuning and slower convergence make them less ideal for high-dimensional scenarios, which are common in financial modeling. Their robustness to noisy data also leaves room for improvement, suggesting that PINNs are more suitable for controlled, structured environments where data quality is high.
- Fourier Neural Operators (FNOs) demonstrate superior performance in high-dimensional settings, where traditional methods often struggle. Their ability to scale efficiently and converge rapidly makes them highly suitable for large-scale financial applications, such as multi-asset options pricing. In addition, FNOs are less sensitive to noisy or incomplete data, which further enhances their practical utility in real-world market conditions where data imperfections are frequent.
- Long Short-Term Memory (LSTM) networks, designed for time-series modeling, show particular promise in capturing the sequential nature of asset prices and volatility. However, the computational burden and complexity of training LSTMs increase with sequence length, which can limit their scalability and make them less appropriate for real-time or high-frequency trading environments. Additionally, LSTMs are less robust to noisy data, which may result in unstable predictions under volatile market conditions.
- Deep Galerkin Method (DGM) stands out as one of the most versatile approaches for solving high-dimensional PDEs, especially in the context of complex financial derivatives such as exotic options. DGMs combine fast convergence with strong robustness to noisy data, making them an excellent choice for pricing complex derivatives in uncertain or fluctuating market environments. Their performance across multiple criteria suggests that DGMs are highly adaptable to a wide range of financial products, from simple European options to path-dependent derivatives.
- Extreme Learning Machine (ELM), while providing rapid training times, is limited in its ability to model the nonlinear dynamics that characterize financial markets. Its lack of accuracy and robustness, especially when dealing with complex financial instruments or noisy data, suggests that ELM is best suited for simple, low-dimensional problems where computational speed is the primary concern.

5.2 Practical Implications

The findings of this comparative study have several important implications for financial practitioners and researchers:

- Algorithm Selection Based on Problem Complexity: For high-dimensional problems, such as multi-asset options or derivatives with path-dependent features, FNOs and DGMs emerge as the most promising solutions due to their scalability, efficiency, and robustness. These methods provide a practical framework for handling large datasets while maintaining computational efficiency, making them suitable for real-time financial applications.
- Trade-offs Between Accuracy and Speed: While ELM offers unmatched speed in terms of training, it comes at the cost of accuracy, particularly for complex financial models. This trade-off must be carefully considered when selecting a method for pricing financial derivatives. In contrast, PINNs and LSTMs offer better accuracy but are computationally more expensive, which may limit their applicability in fast-paced trading environments.
- Handling Noisy Data: Financial markets are inherently noisy, with data often subject to inaccuracies or incomplete information. FNO and DGM demonstrate strong robustness to such noisy data, which enhances their reliability for real-world applications. In contrast, models like LSTM and ELM may require more extensive data preprocessing and tuning to mitigate the effects of data noise.
- Applications in Financial Risk Management: The adaptability of DGMs to high-dimensional problems and their ability to model complex derivatives make them a strong candidate for use in risk management, particularly in assessing the risk associated with exotic options or structured products. On the other hand, PINNs may be better suited for more traditional financial instruments where the underlying dynamics are well understood and where accuracy takes precedence over speed.

5.3 Limitations and Future Work

Despite the significant advantages of deep learning methods for solving the Black-Scholes equation, several limitations remain. The performance of each model is highly dependent on the quality and quantity of the data used for training. Moreover, the interpretability of deep learning models, particularly FNOs and LSTMs, remains a challenge, as these models are often viewed as "black-box" methods. Addressing these interpretability concerns will be crucial for gaining wider acceptance of these models in regulated financial environments.

Future research should explore hybrid approaches that combine the strengths of multiple algorithms. For instance, integrating the speed of ELM with the robustness of DGM could lead to more efficient models for real-time pricing. Additionally, the application of these models to real-world financial datasets would provide further validation of their practical utility and help refine their performance in dynamic market conditions.

6 Conclusion

This paper has presented a qualitative comparison of five deep learning algorithms—Physics-Informed Neural Networks (PINNs), Fourier Neural Operators (FNOs), Long Short-Term Memory (LSTM) networks, Deep Galerkin Method (DGM), and Extreme Learning Machine (ELM)—in the context of solving the Black-Scholes equation for option pricing. Each of these algorithms offers distinct advantages and disadvantages, making them more or less suitable depending on the specific financial problem being addressed.

PINNs provide a robust framework for solving low-dimensional PDEs by incorporating physical constraints into the learning process. However, their sensitivity to hyperparameters and slower convergence make them less practical for high-dimensional, real-time financial applications. **FNOs**, by contrast, excel in high-dimensional settings, offering fast convergence and scalability, making them highly applicable for complex, large-scale financial derivatives. **LSTMs** are well-suited for time-series modeling, but their computational complexity and sensitivity to sequence length limit their scalability for long-term financial forecasting.

DGM stands out as the most versatile algorithm for pricing complex financial derivatives, such as exotic options, due to its fast convergence and robustness to noisy data. **ELM**, while less accurate and robust than the other models, offers unparalleled speed, making it an attractive choice for simple, low-dimensional financial problems where computational efficiency is paramount.

In practice, the selection of an appropriate deep learning algorithm for option pricing will depend on the specific financial instrument, the complexity of the problem, and the available computational resources. **FNOs** and **DGMs** are well-suited for high-dimensional problems requiring scalability and robustness, while **PINNs** and **LSTMs** may be preferred for smaller-scale or time-sensitive problems where accuracy is crucial. **ELM** remains a viable option for cases where speed takes precedence over precision.

Although deep learning methods have demonstrated significant potential for improving option pricing models, further research is needed to refine their applicability in real-world financial markets. Future work should focus on:

- Testing these algorithms on real market data to validate their performance in dynamic, high-volatility environments.
- Developing hybrid models that leverage the strengths of multiple deep learning techniques, combining the speed of **ELM** with the robustness and scalability of **FNOs** or **DGMs**.
- Addressing the challenge of interpretability to ensure that these models can be trusted in regulated financial contexts.

Ultimately, the integration of deep learning methods into financial modeling holds great promise for advancing the accuracy, efficiency, and scalability of option pricing models, particularly in the face of increasingly complex financial instruments and market conditions.

References

1. Cox, J. C., Ross, S. A., & Rubinstein, M. (1979). Option pricing: A simplified approach. Journal of Financial Economics, 7(3), 229–263. DOI: 10.1016/0304-405X(79)90015-1.

2. Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. Journal of Political Economy, 81(3), 637–654. DOI: 10.1086/260062.

3. Kadalbajoo, M. K., & Gupta, V. K. (2012). A collocation method using cubic B-splines for solving singularly perturbed parabolic differential equations. Applied Mathematics and Computation, 219(8), 3735-3745. DOI: 10.1016/j.amc.2012.10.018.

4. Roul, R. K., & De, S. (2020). A high order finite difference method for the Black-Scholes equation governing option pricing in finance. Mathematics and Computers in Simulation, 172, 27-44. DOI: 10.1016/j.matcom.2020.01.002.

5. Lagaris, I. E., Likas, A., & Fotiadis, D. I. (1998). Artificial neural networks for solving ordinary and partial differential equations. IEEE Transactions on Neural Networks, 9(5), 987-1000. DOI: 10.1109/72.712178.

6. Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: Theory and applications. Neurocomputing, 70(1-3), 489-501. DOI: 10.1016/j.neucom.2005.12.126.

7. Yang, X., Zhang, S., & Yang, X. (2018). A Legendre neural network method for solving high-dimensional partial differential equations. Neurocomputing, 314, 460-472. DOI: 10.1016/j.neucom.2018.06.063.

8. Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving partial differential equations. Journal of Computational Physics, 378, 686-707. DOI: 10.1016/j.jcp.2018.10.045.

9. Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., & Anandkumar, A. (2020). Fourier Neural Operator for Parametric Partial Differential Equations. arXiv preprint arXiv:2010.08895. DOI: 10.48550/arXiv.2010.08895.

10. Ma, X., Wu, J., & Zhang, Y. (2021). Structure automatic-determined Fourier extreme learning machine and its application. Neural Computing and Applications, 33(11), 5845-5860. DOI: 10.1007/s00521-020-05545-7.

11. Sirignano, J., & Spiliopoulos, K. (2018). DGM: A deep learning algorithm for solving partial differential equations. Journal of Computational Physics, 375, 1339-1364. DOI: 10.1016/j.jcp.2018.08.029.

12. Han, J., Jentzen, A., & E, W. (2018). Solving high-dimensional partial differential equations using deep learning. Proceedings of the National Academy of Sciences, 115(34), 8505-8510. DOI: 10.1073/pnas.1718942115.

13. Lu, L., Jin, P., Pang, G., Zhang, Z., & Karniadakis, G. E. (2021). Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. Nature Machine Intelligence, 3(3), 218-229. DOI: 10.1038/s42256-021-00302-5.

14. E, W., & Yu, B. (2018). The Deep Ritz Method: A deep learning-based numerical algorithm for solving variational problems. Communications in Mathematics and Statistics, 6(1), 1-12. DOI: 10.1007/s40304-018-0127-z.

15. Jacot, A., Gabriel, F., & Hongler, C. (2018). Neural tangent kernel: Convergence and generalization in neural networks. Advances in Neural Information Processing Systems, 31, 8571-8