ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
(ICAI)

Máster en Big Data: Tecnología y Analítica Avanzada

Interpretable Neural Networks for Option Pricing

A Comparative Study with Black-Scholes

Autor
Juan Sánchez Fernández

Dirigido por
Alejandro Polo Molina
Jaime Pizarroso Gonzalo

Madrid
June 17, 2025

Juan Sánchez Fernández, declara bajo su responsabilidad, que el Proyecto con título **Interpretable Neural Networks for Option Pricing: A Comparative Study with Black-Scholes** presentado en la ETS de Ingeniería (ICAI) de la Universidad Pontificia Comillas en el curso académico 2024/25 es de su autoría, original e inédito y no ha sido presentado con anterioridad a otros efectos. El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.

Fdo.: .... Fecha: 17 / 06 / 2025

Autorizan la entrega:

Directores del Proyecto

Alejandro Polo Molina

Fdo.: Alejandro Polo Fecha: 17 / 06 / 2025

Jaime Pizarroso Gonzalo

Fdo.: .... Fecha: 17 / 06 / 2025

V. B. del Coordinador de Proyectos

Carlos Morrás Ruiz-Falcó

Fdo.: .... Fecha: ...... / ...... / ....

AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESINAS O MEMORIAS DE BACHILLERATO

1º. Declaración de la autoría y acreditación de la misma.

El autor D. Juan Sánchez Fernández **DECLARA** ser el titular de los derechos de propiedad intelectual de la obra: Interpretable Neural Networks: A Comparative Study with Black-Scholes, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

2º. Objeto y fines de la cesión.

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, los derechos de digitalización, de archivo, de reproducción, de distribución y de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

3º. Condiciones de la cesión y acceso

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

(a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar "marcas de agua" o cualquier otro sistema de seguridad o de protección.

(b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.

(c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.

(d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.

(e) Asignar por defecto a estos trabajos una licencia Creative Commons.

(f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente*).

4º. Derechos del autor.

El autor, en tanto que titular de una obra tiene derecho a:

(a) Que la Universidad identifique claramente su nombre como autor de la misma

(b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.

(c) Solicitar la retirada de la obra del repositorio por causa justificada.

(d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

5º. Deberes del autor.

(a) El autor se compromete a:

(b) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.

(c) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.

(d) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.

(e) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción de derechos derivada de las obras objeto de la cesión.

6º. Fines y funcionamiento del Repositorio Institucional.

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusive del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a . . . . . . de . . . . de . . . . . .

ACEPTA

Fdo.: . . . .

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

Resumen

En este trabajo se aborda el problema de la valoración de opciones financieras mediante técnicas de aprendizaje automático, y en particular, mediante redes neuronales multicapa (MLP). Partiendo de las limitaciones conocidas del modelo clásico de Black-Scholes, se propone el uso de modelos data-driven capaces de capturar relaciones no lineales más representativas de los mercados reales.

Se desarrolla un pipeline completo de recolección, limpieza y preprocesamiento de datos sobre opciones europeas del S&P 500, utilizando librerías de código abierto como `yfinance` y `BeautifulSoup`. Sobre este conjunto de datos, se implementan dos modelos principales: un modelo *Baseline MLP* con las mismas variables de entrada que Black-Scholes (precio spot, strike, tiempo hasta vencimiento e implied volatility), y un modelo *Extended MLP* que incorpora variables adicionales de mercado (open interest, volumen y moneyness).

El análisis comparativo de resultados muestra que ambos modelos MLP son capaces de mejorar sustancialmente la capacidad predictiva frente al modelo de Black-Scholes, especialmente en términos de error cuadrático medio (RMSE) y coeficiente de determinación (R^2). El modelo extendido alcanza un R^2 superior al 92% sobre el conjunto de test.

Además, se lleva a cabo un análisis exhaustivo de interpretabilidad mediante el uso de derivadas parciales del modelo (sensibilidades) y curvas α implementadas con la librería `NeuralSens`. Este análisis permite comparar las sensibilidades aprendidas por la red neuronal con los clásicos *Greeks* de Black-Scholes, mostrando cómo el modelo MLP captura patrones no lineales y dependencias locales especialmente relevantes en escenarios extremos de mercado (por ejemplo, volatilidad próxima a cero, vencimientos muy cortos o opciones muy ITM).

Los resultados obtenidos ponen de manifiesto tanto las ventajas predictivas como los retos interpretativos de los modelos neuronales en finanzas, y refuerzan el potencial de enfoques híbridos que combinen modelos data-driven con herramientas de explicabilidad financiera.

Abstract

This work addresses the problem of financial option pricing through machine learning techniques, specifically using multilayer perceptrons (MLPs). Starting from the well-known limitations of the classical Black-Scholes model, we propose data-driven models capable of capturing nonlinear relationships that better reflect real market dynamics.

A complete pipeline for collecting, cleaning, and preprocessing European option data from the S&P 500 is developed, using open-source libraries such as `yfinance` and `BeautifulSoup`. Based on this dataset, two main models are implemented: a *Baseline MLP* with the same input variables as Black-Scholes (spot price, strike price, time to maturity, and implied volatility), and an *Extended MLP* that incorporates additional market variables (open interest, volume, and moneyness).

The comparative analysis shows that both MLP models substantially improve predictive performance over the Black-Scholes model, particularly in terms of Root Mean Squared Error (RMSE) and coefficient of determination (R^2). The extended model achieves an R^2 exceeding 92% on the test set.

Additionally, an interpretability analysis is conducted using the model's partial derivatives (sensitivities) and α curves implemented with the `NeuralSens` library. This analysis enables a comparison between the sensitivities learned by the neural network and the classical Black-Scholes Greeks, revealing how the MLP captures nonlinear patterns and local dependencies that become especially relevant under extreme market scenarios (e.g., near-zero volatility, very short maturities, or deep ITM options).

The results highlight both the predictive advantages and interpretability challenges of neural network models in finance, reinforcing the potential of hybrid approaches that combine data-driven models with financial explainability tools.

PARA MI ABUELO,
que, aunque no llegó a verme terminar esta etapa,
nunca tuvo la menor duda de que lo conseguiría.

"All models are wrong, but some are useful."
GEORGE E. P. BOX

Agradecimientos

Me gustaría darle las gracias, primero a mi familia, ya que sin su apoyo incondicional este año habría sido muchísimo más duro, ojalá haber podido estar más cerca precisamente este año.

A mis amigos, a los nuevos y a los de siempre, por contrubuir, a su manera, a conformar la persona que soy a dia de hoy.

También a mis tutores, Alejandro y Jaime, por haberme guiado en este trabajo y haberme ayudado tanto en tan poco tiempo. Gracias por haberme dado la oportunidad de conocer el mundo de la investigación más de cerca y por vuestra paciencia y dedicación durante todo el proceso. Vuestro apoyo ha sido fundamental para la realización de este trabajo y para mi crecimiento académico y personal.

Y por último a Pedro que, aun estando más lejos este año que el anterior, me ha apoyado aún más si cabe y me ha hecho sentir que la distancia Madrid-Bruselas es solo un paseo. Estoy seguro de que pronto podremos celebrarlo juntos como nos merecemos.

Juan Sánchez Fernández

Madrid

15 junio, 2025

Contents

Chapter 1

Introduction

Since its conception, financial markets have evolved into ever more complex systems in which derivative products, such as options, play a key role. Financial options are widely used by investors for hedging, speculation, and portfolio diversification.

By definition, a financial option is a contract that gives the holder the right, but not the obligation, to buy or sell an underlying asset at a predetermined price (the strike price) on or before a specified expiration date. The two main types of options are call options, which give the holder the right to buy the underlying asset, and put options, which give the holder the right to sell it [1].

The use of contracts with option features is not a modern invention. The basis for such features emerges from the fundamental process of exchange, and there is evidence that even in ancient times, merchants and traders used similar contracts to hedge against price fluctuations [2].

The evolution of options revolved around two main elements: security of the transactions; and the emergence of speculative trading. Both things were closely related to the concentration of commercial activity, initially at medieval markets and fairs and later on the bourse. One example of this is the Antwerp Bourse, which was established in the 16th century [3], where, due to the rapid expansion of seaborne trade, speculative transactions on 'to arrive' grain gained popularity among merchants.

In the 19th century, Louis Bachelier set the foundations for modern financial theory by introducing the concept of Brownian motion to derivative pricing in his 1900 thesis [4]. Although his work had little impact at the time, it laid the groundwork for future developments in the theory of financial derivatives. Almost a century later, in 1973, Fischer Black and Myron Scholes, with the earlier contribution of Robert Merton, published the Black-Scholes (BS) model [1], which led to an enormous growth in options trading and provided mathematical legitimacy to the field of financial derivatives.

Options trading has grown exponentially since the introduction of the BS model, reaching in 2024, according to the *Options Clearing Corporation* (OCC), a value of $1.4 trillion in traded options [5]. This explosive growth in options trading volume has underscored the need for more robust and accurate pricing methods. While the BS model marked a groundbreaking milestone in financial mathematics, its simplifying assumptions, such as constant volatility and log-normal price distributions, often fail to capture the complex dynamics observed in modern markets. As a result, there has been a growing interest in developing alternative models that can better reflect market realities.

In recent years, the increasing availability of large-scale financial datasets combined with relevant progress in computing power has opened the possibility of data-driven approaches. In particular, machine learning (ML) techniques have entered the scene as powerful tools capable of learning complex relationships from data, making them very promising candidates for addressing the problem of option pricing in a more versatile and adaptive way. Among them, Multilayer Perceptrons (MLPs) have been proposed as a simple yet powerful architecture for modeling the nonlinear relationships inherent in option pricing [6].

Therefore, this thesis aims to empirically evaluate the effectiveness of MLPs in approximating option prices, taking advantage of their capacity to model complex and nonlinear patterns and adapt to empirical data without relying on rigid assumptions. In particular, we focus on two core goals: to demonstrate, through concrete metrics such as RMSE and R^2, that an MLP can outperform BS in fitting observed market prices; and to enhance the interpretability of these models through a sensitivity analysis of the partial derivatives of the model output with respect to the input variables. The structure of this work will be further explained in Subsection 1.3.

1.1 Motivation

While classical models like BS continue to play a foundational role in the pricing of financial options, their simplifying assumptions—such as constant volatility and log-normal asset returns—often lead to systematic pricing errors in real market conditions [7, 8, 9]. These limitations have motivated the exploration of data-driven alternatives capable of capturing more complex relationships observed in market data.

Another fundamental limitation of current ML models lies in their lack of interpretability. Neural Networks (NN), in particular, are often perceived as black boxes, providing little transparency about how inputs are transformed into outputs. This opaqueness poses a barrier to adoption in the financial industry, where trust, regulatory scrutiny, and risk management require not only accurate but also explainable predictions [10].

In this context, this work aims to explore the use of modern ML models for option pricing with a dual focus: evaluating predictive performance and enhancing interpretability. By doing so, it contributes to the ongoing effort to integrate advanced ML into financial decision-making in a robust and explainable way.

1.2 Objectives

The main objective of this thesis is to empirically evaluate the effectiveness of MLPs in option pricing, comparing their performance to the classical BS model and analyzing the interpretability of their predictions through sensitivity analysis. To this end, the specific goals of the thesis are as follows:

- Review the existing literature on option pricing models, with a focus on the limitations of traditional approaches and the potential of ML techniques.
- Develop and implement MLP-based NNs for option pricing capable of capturing the complex dynamics of implied volatility surfaces.
- Benchmark the performance of the proposed MLPs against the BS model using quantitative metrics such as Root Mean Squared Error (RMSE) and coefficient of determination (R^2).
- Analyze the interpretability of the trained MLP models by computing and visualizing the partial derivatives of the model output with respect to its input variables, interpreting these partial derivatives as analogs to the Greeks, thus enabling comparisons between classical sensitivities and neural network behavior.

1.3 Methodology

This thesis will follow a structured approach to achieve the objectives outlined above and will be divided into the following chapters:

- **Chapter 1: Introduction** – Introduces the motivation behind the study, outlines the research objectives, and describes the overall methodology followed throughout the thesis.
- **Chapter 2: Classical Option Pricing Models** – Presents the theoretical foundations of option pricing, including the BS model, the concept of moneyness, and the main sensitivity measures known as Greeks.
- **Chapter 3: Literature Review** – Reviews previous research related to data-driven approaches for option pricing, with a particular focus on studies that apply ML and NN to improve pricing accuracy and model interpretability.
- **Chapter 4: Data Collection and Preprocessing** – Describes the data sources used in this work, the option chain extraction pipeline, and the preprocessing steps applied to prepare the dataset for model training and evaluation.
- **Chapter 5: Model Development and Evaluation** – Presents the design and implementation of the ML models, covering architectural choices, training procedures, and evaluation metrics. It also includes a comparative analysis with traditional models.
- **Chapter 6: Interpretability Analysis** – Analyzes the interpretability of the proposed models by examining their sensitivity to input variables changes through derivative-based methods and visual sensitivity measures.
- **Chapter 7: Discussion, Conclusions, and Future Work** – Summarizes the experimental results, discusses the key insights derived from the models' behavior, and proposes directions for future research.

1.4 Resources

To develop and evaluate this thesis, we will require access to a large and representative dataset of option prices and their associated market information.

We will focus on S&P 500 index options, which are widely used in financial research and practice due to their liquidity and well-documented characteristics. Importantly, S&P 500 options are European-style, meaning they can only be exercised at expiration. This simplifies the modeling process and aligns well with many theoretical pricing frameworks.

The data will be collected using the `yfinance` Python library [11], which provides access to both historical asset prices and the full option chain (including bid, ask, implied volatility, and open interest) for each security. This approach enables full control over the data acquisition process and allows for transparent, reproducible experiments.

This dataset will serve as the common data source from which training and testing subsets will be extracted for our ML models, allowing for a fair comparison with the BS benchmark. Additionally, the use of a custom-built dataset supports the goal of making the entire research pipeline accessible and replicable for future work.

The full process of data collection, cleaning, and feature engineering is described in detail in Chapter 4, where we present the methodology used to construct the final dataset employed in the experiments.

Chapter 2

Classical Option Pricing Models

As introduced in the previous chapter, option pricing has long been one of the central problems in quantitative finance. This chapter presents the theoretical foundations of classical option pricing models, which serve as a baseline and point of comparison for the data-driven approaches developed later in this work.

2.1 The Black-Scholes Model

As stated in Chapter 1, the BS model, introduced in 1973 by Fischer Black and Myron Scholes, and later extended by Robert Merton, was the first widely adopted framework for pricing European-style options [12]. The model is built upon several simplifying assumptions: the underlying asset price follows a geometric Brownian motion; markets are frictionless and free of transaction costs or taxes; volatility is constant; the short-term risk-free interest rate is known and constant over time; no arbitrage opportunities exist; and the underlying asset pays no dividends [7, 13].

Under these assumptions, the price of a European call option can be expressed in closed form as follows:

$$
\begin{aligned}
C(S) &= S \cdot \Phi(d_1) - Ke^{-rT} \cdot \Phi(d_2), \\
d_1 &= \frac{\ln(S/K) + (r + \sigma^2/2)T}{\sigma\sqrt{T}}, \\
d_2 &= d_1 - \sigma\sqrt{T}
\end{aligned}
\tag{2.1}
$$

where:

- $C(S)$ is the price of the call option at time t,
- S is the current price of the underlying asset,
- K is the strike price,
- T is the time to maturity (in years),
- r is the risk-free interest rate,
- σ is the volatility of the underlying asset,
- $\Phi(\cdot)$ denotes the cumulative distribution function of the standard normal distribution.

This model allows practitioners to compute theoretical option prices based on observable market inputs. However, the restrictive assumptions of the model often limit its accuracy when applied to real-world financial markets.

2.2 Moneyness

A key concept in option pricing is *moneyness*, which refers to the relationship between the current price of the underlying asset and the strike price of the option. Moneyness helps categorize options based on their intrinsic value and their likelihood of being exercised profitably.

For call options:

- **In-the-money (ITM)**: $S > K$
- **At-the-money (ATM)**: $S \approx K$
- **Out-of-the-money (OTM)**: $S < K$

For put options, the definitions are inverted:

- **ITM**: $S < K$
- **ATM**: $S \approx K$
- **OTM**: $S > K$

Moneyness plays a critical role in determining the option's intrinsic value, which is the difference between the current price of the underlying asset and the strike price. The intrinsic value is zero for OTM options, while ITM options have a positive intrinsic value [14].

2.3 Option Greeks

Beyond pricing itself, a key aspect of option trading and risk management is understanding how option prices respond to changes in market conditions. For this purpose, traders rely heavily on a set of sensitivity measures known as *Greeks*. These quantify how the price of an option reacts to infinitesimal changes in key variables such as asset price, volatility, time, and interest rates.

Greeks can be analytically derived from option pricing models such as BS, and the most commonly used for options include:

- **Delta** (Δ): Quantifies how much the option price changes as the underlying asset price varies:
$$\Delta = \frac{\partial C}{\partial S}$$

- **Gamma** (Γ): Captures the curvature of the option price with respect to the underlying price, i.e., how Delta itself changes as S moves:
$$\Gamma = \frac{\partial^2 C}{\partial S^2}$$

- **Vega** (ν): Reflects the sensitivity of the option price to variations in the implied volatility of the underlying:
$$\nu = \frac{\partial C}{\partial \sigma}$$

- **Theta** (Θ): Represents the time decay of the option, indicating how its price change as the expiration date approaches:
$$\Theta = \frac{\partial C}{\partial T}$$

- **Rho** (ρ): Expresses the sensitivity of the option price to changes in the risk-free interest rate:
$$\rho = \frac{\partial C}{\partial r}$$

These sensitivities provide valuable information for hedging, risk management, and trading decisions under various market conditions [15].

For example, if a trader were to choose between two options with identical strike prices and expiration dates, but different Delta values, the option with the higher Delta would be more sensitive to changes in the underlying asset's price. This sensitivity would directly affect the trader's decision depending on their risk preferences and market outlook.

2.4 Limitations of Classical Models

While the BS model and its associated Greeks have become foundational tools in modern financial markets [16], their simplifying assumptions often fail to capture real-world complexities. In particular, assumptions of constant volatility, log-normal price dynamics, and frictionless markets are frequently violated. As a result, systematic pricing errors may arise, especially during periods of market stress or when pricing exotic instruments.

These limitations have motivated extensive research into alternative pricing methodologies. In recent years, data-driven approaches such as ANN have gained increasing attention for their ability to model nonlinear dependencies directly from observed data [17]. The following chapter provides a detailed review of previous research in this direction.

Chapter 3

Literature Review

This chapter provides a review of the most relevant research efforts that have attempted to improve option pricing beyond classical analytical models. In particular, we focus on the growing body of work exploring data-driven approaches, with an emphasis on the application of ML and NN to address the limitations of traditional models and capture the complex dynamics observed in real financial markets.

3.1 Early Applications of Machine Learning in Finance

One of the first relevant contributions towards non-parametric modeling was made by Hutchinson et al. [18], who applied NN and kernel regressions to learn option prices directly from market data, pioneering a line of research that would later become central to modern data-driven finance. Their work started by identifying the inherent limitations of parametric pricing models such as BS, which rely on strong assumptions regarding volatility, return distributions, and market efficiency. In contrast, they proposed a non-parametric alternative based on learning networks capable of estimating pricing functions directly from historical data, without requiring a fixed functional form.

Hutchinson et al. evaluated several network architectures, including radial basis function networks (RBF), MLPs, and projection pursuit regression (PPR), and demonstrated through Monte Carlo simulations that these models could successfully replicate the BS pricing formula with high out-of-sample accuracy. Further-more, by applying their methodology to real data on S&P 500 futures options, they showed that learning networks not only matched but often outperformed BS, particularly in pricing and hedging effectiveness across different moneyness and maturities. Their study concluded by highlighting the promise of learning networks as viable alternatives to parametric pricing models and outlined several directions for further research, such as expanding input features, optimizing network architectures, and improving evaluation metrics for nonlinear models.

Building on these early findings, the following decade saw machine learning applications to option pricing become increasingly widespread, driven by advances in computational power and algorithmic sophistication. Studies such as Mitra [19] and Can & Fadda [20] applied MLP incorporating standard financial features like spot price, strike price, time to maturity, interest rates, and historical volatility, demonstrating the ability of machine learning models to capture complex relationships beyond the reach of traditional formulas. Ruf and Wang [21] later provided a comprehensive review of machine learning applications to derivative pricing up to 2019, marking a transition point towards more sophisticated deep learning architectures that now form the basis of ongoing research in this area.

3.2 Recent Advancements

Following the foundational contributions Hutchinson et al.[18] and the subsequent stream of studies that demonstrated the viability of NN for option pricing[19, 20, 21], recent years have witnessed a significant acceleration in the use of more advanced deep learning techniques. The increasing availability of high-frequency financial data, coupled with advances in computing power and algorithmic design, has enabled researchers to explore architectures capable of capturing more complex temporal dependencies and nonlinear relationships that are characteristic of real financial markets.

A representative example of this evolution is the study conducted in 2019 by Stanford University students Alexander Ke and Andrew Yang, who investigated the application of Long Short-Term Memory (LSTM) networks to option pricing [7]. In their work, they compared three neural network architectures: a simple MLP trained with 20-day historical volatility to predict option prices; an alternative MLP architecture estimating bid and ask prices separately; and finally, an LSTM network designed to learn volatility dynamics directly from historical price sequences. All three models demonstrated superior predictive accuracy compared to the BS model, with the bid-ask MLP variant achieving the most favorable results.

Subsequent studies have further refined these approaches by systematically analyzing model robustness across different market conditions. For instance, İltüzer [8] examined how NN compare with BS under various volatility regimes. Her findings suggest that NN tend to outperform BS for call options during tranquil market periods, while BS may retain an advantage during turbulent phases, particularly for calls, with an inverse pattern observed for puts.

Beyond recurrent architectures, the application of alternative machine learning paradigms has also gained momentum. In this context, Berger [9] conducted a comprehensive empirical comparison between deep learning models implemented in TensorFlow, ensemble methods such as XGBoost, and automated pipelines generated by Google Cloud's AutoML. His results demonstrate that while all machine learning models achieved lower mean absolute error (MAE) than BS, the XGBoost model with maximum depth 10 delivered the best performance, outperforming even AutoML despite significantly lower computational cost.

3.3 Interpretability and explainability

A critical challenge of ML models in finance is their interpretability compared to traditional parametric models, which offer explicit financial meaning to different parameters. In contrast, NN and other ML algorithms often operate as black-box models, making it difficult to assess whether their predictions adhere to fundamental financial principles such as no-arbitrage conditions, monotonicity with respect to key variables, or consistency with the structure of implied volatility surfaces. As a result, explainable AI (XAI) methods have been increasingly applied to provide insights into ML pricing models and to ensure their economic plausibility [22].

For example, Pimentel et al. [23] applied SHAP (SHapley Additive exPlanations) values to decompose LSTM predictions into the marginal contributions of each input feature, including spot price, strike price, time to maturity, volatility, and risk-free rate. Their analysis confirmed that the LSTM model learned economically meaningful relationships, such as the positive sensitivity of call option prices to underlying price and volatility, as well as the expected decay in option value as time to maturity decreases, consistent with classical option pricing theory.

Similarly, Liang and Cai [24] employed Accumulated Local Effects (ALE) plots to analyze the response of the model to variations in key inputs such as implied volatility and moneyness. The ALE analysis provided a global visualization of how these inputs affected predicted option prices, revealing that the model successfully captured the shape of empirical implied volatility surfaces observed in real markets.

In this context, this thesis will further explore sensitivity-based explainability, with a focus on partial derivatives and alpha curves, as an alternative to global XAI techniques.

3.4 Remaining Challenges and Future Directions

Despite the substantial progress made in recent years, several open challenges remain in the application of machine learning to option pricing, offering margin for continued research.

First, while NN have repeatedly demonstrated superior predictive accuracy relative to classical approaches, they often lack financial interpretability. As highlighted in recent studies [23, 24, 22], understanding how they internally process financial inputs remains a central issue. Without proper interpretability tools, there is a risk that models may learn spurious correlations or exhibit behaviors inconsistent with fundamental financial principles. This thesis addresses this issue by applying sensitivity-based interpretability techniques, such as partial derivatives and alpha curves, enabling a direct comparison with classical Greeks and offering financial insights into model behavior

Second, like most existing studies, this work relies on supervised learning frameworks that directly map inputs to observed prices, without explicitly enforcing financial principles such as no-arbitrage or risk-neutral valuation. Future research may explore hybrid models that combine analytical structures with neural networks to improve financial consistency.

Finally, while this thesis focuses on static option snapshots, real markets are dynamic, and adapting models to shifting volatility regimes remains challenging. Incorporating temporal information or regime-switching mechanisms may enhance robustness in changing market environments.

Overall, this thesis contributes to ongoing efforts by combining predictive performance with interpretable neural network models, providing a framework that balances flexibility and financial reasoning.

Chapter 4

Data collection and Preprocessing

This chapter provides a comprehensive overview of the data collection and preprocessing steps undertaken to prepare the dataset for training and evaluating ML models for option pricing. The process involves several key stages, including the selection of data sources, scraping option chain data, cleaning and feature engineering, and summarizing the dataset's characteristics. Each section will detail the methodologies employed, the challenges encountered, and the solutions implemented to ensure a robust and reliable dataset.

Unlike many existing studies that rely on pre-processed, and quite often proprietary, datasets, in this work, we tried to focus on open source data collection alternatives in order to ensure the reproducibility of our results.

4.1 Data Source and Ticker Selection

The primary data source for this study is the `yfinance` Python library, which provides access to historical stock prices and option chain data for a wide range of securities. This open-source library utilize Yahoo's publicly available APIs for accessing financial data, making it a suitable choice for our research. The library allows us to retrieve not only the historical prices of underlying assets but also the full option chain, including bid, ask, implied volatility, and open interest for each.

The list of tickers used in this study consists of companies listed in the S&P 500 index. This index was chosen due to its high liquidity, market representativeness, and the widespread availability of options data across its members. The list of S&P 500 tickers is retrieved dynamically from the corresponding Wikipedia page using the `BeautifulSoup` library [25]. This approach ensures that the list is always up to date with the latest index composition, avoiding reliance on static files.

4.2 Option Chain Scraping Pipeline

Once the list of S&P 500 tickers was obtained from Wikipedia, a custom scraping pipeline was developed to retrieve option chain data for each ticker through the `yfinance` interface.

For every company in the index, the pipeline iterates over all available expiration dates, downloading both call and put options. Each option contract includes metadata such as:

- Ticker symbol
- Strike price
- Expiration date
- Option type (call or put)
- Bid and ask prices
- Implied volatility
- Open interest
- Last trade date

Each option is enriched with two additional fields:

- **price** (the spot price of the underlying asset at the time of download),
- **remaining** (the number of days remaining until expiration, calculated as the difference between the expiration date and the current date).

To ensure scalability and reproducibility, the pipeline is implemented as a standalone Python script, designed to be executed daily. The output of this step consists of two raw CSV files, one for call options and one for puts, saved in the `data/raw/` directory with the current date in the filename.

4.3 Data Cleaning and Feature Engineering

After the raw option data is collected, a preprocessing phase is applied to clean the dataset and engineer relevant features for modeling.

First, the following filters are applied to ensure data consistency:

- Rows with missing values in critical fields such as `bid`, `ask`, or `openInterest` are removed.

- Contracts with non-positive implied volatility (`impliedVolatility` ≤ 0) are excluded, since implied volatility values equal to zero or negative are not meaningful from a financial perspective and may be result of data errors or market anomalies.

Then, several derived features are computed to enhance the dataset:

- **Time to maturity** (T), calculated as the number of remaining days to expiration divided by 365.
- **Mid-price**, computed as the average between bid and ask prices.

These features are crucial for training models that generalize well, as they encode time and pricing dynamics that are not directly available in the raw data.

4.4 Dataset Summary and Exploratory Statistics

After completing the scraping and preprocessing steps, the resulting dataset contains a rich and diverse collection of option contracts across multiple tickers, expiration dates, and strike prices. The final dataset used for model training consists of approximately **263,000** option contracts, covering **500** different tickers. The exact number of observations may vary slightly depending on the execution date and the availability of option chain data on Yahoo Finance.

In this section, we present a statistical summary of the dataset and explore its key characteristics.

4.4.1 Summary Statistics

Table 4.1 shows a general overview of the cleaned dataset, including key descriptive statistics for relevant numerical features.

Variable	**Mean**	**Std. Dev.**	**Min**	**Max**
Spot Price (`price`)	317.85	619.69	8.59	5475.26
Strike Price (`strike`)	330.79	599.48	0.5	7900
Time to Maturity (T)	0.53	0.56	0.003	2.55
Mid Price (`midPrice`)	55.53	159.72	0.00	3986
Implied Volatility	0.52	0.59	0.00001	37.22
Open Interest	642.6	2854.98	0.00	231208

Table 4.1: Summary of the cleaned dataset.

4.4.2 Distribution of Contracts

The dataset includes both call and put options, with a wide variety of maturities and strike prices. The following characteristics were observed:

- Most options have a time to maturity (T) of less than 100 days, consistent with the high trading volume of near-term contracts.
- Approximately 58% of the options in the dataset are OTM, indicating a slight skew in favor of contracts with strike prices above (for calls) or below (for puts) the spot price of the underlying asset.
- The distribution of implied volatility is right-skewed, with most values clustered between 0% and 10%, but with some extreme values reaching above 20%.
- The mid-price also exhibits a right-skewed distribution, with a significant number of contracts priced below \$1000, but some high-value contracts exceeding \$3000.

4.5 Limitations and Considerations

While the data collection and preprocessing pipeline was designed to ensure transparency, reproducibility, and scalability, it is important to acknowledge several limitations and methodological choices that may impact the generalizability of the results.

4.5.1 Scraping Limitations

- **Snapshot-only data**: The dataset reflects a single snapshot in time, as all option chain data was scraped on a specific execution date. As a result, the dataset does not capture historical trends or temporal dynamics such as volatility clustering, term structure shifts, or changes in market sentiment over time.
- **Limited coverage for some tickers**: Although the S&P 500 contains highly liquid stocks, not all tickers have the same depth in their option chains. Some companies, especially those with lower trading volume or lower investor interest, may only offer a few expiration dates or a sparse set of strike prices. This can reduce the representativeness or richness of the data for those assets.

4.5.2 Modeling Decisions

- **No imputation of missing values**: Instead of applying interpolation or statistical imputation techniques, all rows containing missing values in key fields (e.g., bid, ask, open interest, implied volatility) were removed during preprocessing. While this ensures data quality, it may bias the dataset toward more liquid contracts.
- **Bias toward short-term, liquid contracts**: Due to filtering steps and data availability, the final dataset contains a majority of short-term options with fewer than 100 days to maturity and relatively high open interest. This focus on near-term contracts enhances model reliability but may limit its extrapolation capacity to longer-dated or less actively traded options.

These limitations do not compromise the validity of the results for the stated objectives, comparing neural network pricing performance to the BS model and exploring model explainability. However, they should be considered when extending this work to production-grade systems or other financial contexts.

4.6 Code Availability

All scripts, data collection pipelines, and preprocessing steps described in this chapter have been implemented in Python and are fully available for reproducibility purposes. The complete source code repository can be accessed at:

github.com/juansanchezf/optionGreeks

This repository includes data scraping scripts, feature engineering routines, model training code, and all experiments performed throughout this thesis.

Chapter 5

Model Development and Evaluation

In this chapter, we describe the methodology followed for training and evaluating neural network models for option price prediction. The primary objective is to assess the capability of deep learning techniques, particularly MLPs, to improve upon the classical BS formula even when provided with the same set of inputs. Subsequently, we extend the architecture to include additional inputs and evaluate its performance gain.

To this end, two MLP architectures are implemented:

- A **baseline MLP**: Designed with the same inputs as the BS model, to facilitate direct comparison under equivalent assumptions.
- An **extended MLP**: Which incorporates additional features to exploit more of the available data and further improve predictive accuracy

In addition to evaluating predictive performance, the baseline model will also be used in Chapter 6 to perform a sensitivity analysis of the model's predictions. This will allow us to understand how changes in input variables affect the predicted option prices and how these new metrics can be compared to classical financial instruments like the Greeks.

5.1 Baseline MLP: Reproducing BS Inputs

We begin by establishing a fair benchmark model. Specifically, we first design a simple MLP that replicates the input space of the BS model: spot price (S), strike price (K), time to maturity (T), and implied volatility (σ). This design isolates the predictive power of the neural architecture itself under equivalent conditions.

5.1.1 Model Architecture and Training

The baseline model takes the price, strike, time to maturity, and implied volatility as inputs and predicts the mid-price of the option, defined as the average of the bid and ask prices. The dataset was split into training and test sets using an 80/20 proportion.

Next, a hyperparameter tuning phase was conducted using a grid search with 10-fold cross-validation over the following parameter grid:

```
param = {
    'MLP__activation': ['relu', 'tanh'],
    'MLP__alpha': [0.01, 0.001],
    'MLP__hidden_layer_sizes': [(30,), (40,), (60,)],
    'MLP__learning_rate_init': [0.001, 0.01, 0.05, 0.1, 0.2]
}
```

This tuning procedure explored different combinations of the regularization factor, number of neurons for the hidden layer, initial learning rate and activation function. The best combination, shown in Figure 5.1, corresponds to a hidden layer of 60 neurons and $\alpha = 0.001$, with a Tahn activation function and a learning rate of 0.01. The model was trained using the Adam optimizer with a mean squared error (MSE) loss function.

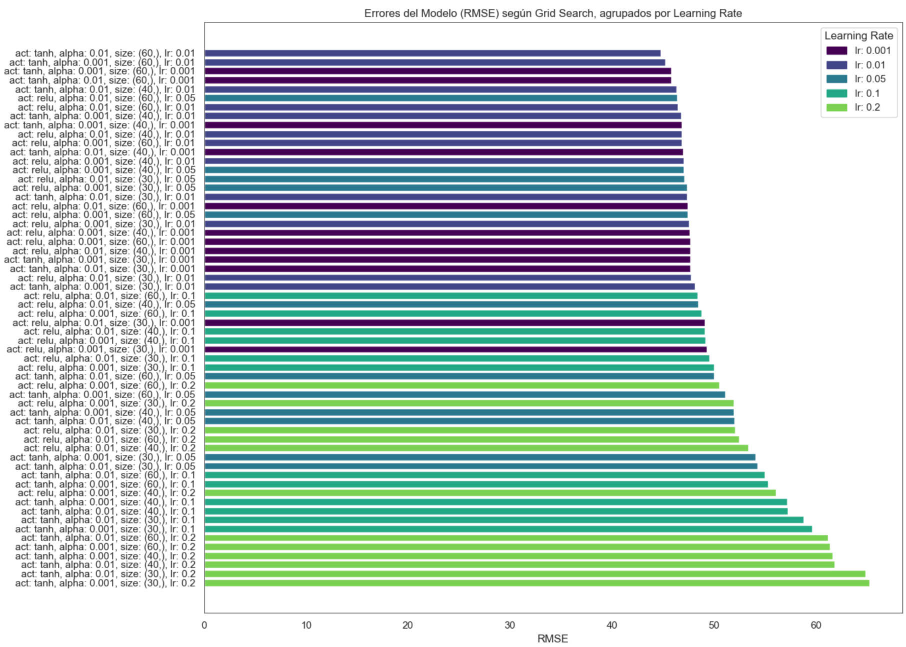

Figure 5.1: Grid Search results for baseline MLP hyperparameters.

5.1.2 Performance Evaluation

With the model fully trained, we proceed to evaluate its performance relative to the BS analytical pricing formula. Predictions were obtained for both training and test sets using each method. The BS predictions were computed with a constant risk-free interest rate of $r = 4.5\%$ [26].

The overall performance metrics obtained from both models are summarized in Table 5.1:

	Training			Test		
Model	**MAE**	**RMSE**	$\mathbf{R^2}$	**MAE**	**RMSE**	$\mathbf{R^2}$
Black-Scholes	**10.72**	80.39	0.75	**10.75**	87.17	0.71
Baseline MLP	12.26	**52.78**	**0.89**	12.51	**57.39**	**0.87**

Table 5.1: Performance comparison between BS and Baseline MLP.

Beyond these aggregated metrics, we conducted a more granular analysis to compare how often each model outperformed the other on individual predictions. Specifically, we computed the number of test cases in which the BS formula yielded a prediction closer to the observed mid-price than the MLP, and vice versa.

The results of this pairwise comparison show that the BS model outperformed the MLP in **19,434** instances, with an average improvement margin of **8.91** and a median of **3.49** in absolute error. Conversely, the MLP provided a better approximation in **7,927** cases, achieving a substantially higher average margin of improvement of **15.83** and a median of **2.24**.

5.1.3 Discussion

As shown in Table 5.1, while the Mean Absolute Error (MAE) is slightly lower in the BS model, the baseline MLP drastically reduces the RMSE and improves the coefficient of determination (R^2), especially in capturing large deviations in price. This confirms the ability of neural networks to learn nonlinear patterns beyond the assumptions embedded in the analytical formula.

Moreover, although the BS model yields more accurate predictions in a greater number of individual cases, the neural network achieves considerably larger improvements in the subset of cases where it performs better. This asymmetric behavior highlights the complementary nature of both approaches and reinforces the potential of ML techniques in modeling financial instruments.

These results motivate a more detailed sensitivity analysis, particularly centered on those extreme cases where the MLP substantially outperforms the BS model. Such analysis can reveal which input configurations drive the largest improvements and help interpret the internal mechanisms behind the MLP's predictions. This will be thoroughly explored in Chapter 6.

5.2 Extended MLP: Adding Informative Inputs

Following the encouraging results obtained with the baseline model, we next investigate whether expanding the input space with additional market variables can further enhance the model's predictive capabilities. While the baseline model demonstrated that a neural network can approximate the BS formula with competitive accuracy, its input vector was constrained to the same variables that the analytical model uses. However, real-world market dynamics often involve more complex patterns and additional factors that influence option pricing. To capture such information, we develop a more expressive model by enriching the input vector with auxiliary features such as market volume, open interest, and the moneyness indicator.

5.2.1 Model Architecture and Training

To incorporate this additional information, we extend the input space to include three new variables: `openInterest`, `volume`, and a binary indicator `inTheMoney`, which signals whether the option is currently profitable to exercise.

As in the baseline experiment, hyperparameter optimization was performed through grid search using a cross-validated pipeline. This time, however, a wider architecture space was explored to account for the increased input dimensionality:

- Hidden layer sizes: $(80,)$, $(40, 40)$, and $(80, 40, 20)$
- Activation functions: `relu` and `tanh`
- Learning rates: 0.01 and 0.001
- Regularization parameter `alpha`: 0.001

The results of the grid search, presented in Figure 5.2, indicate that the best performing configuration corresponds to a two-layer network $(40, 40)$ with ReLU activations, learning rate 0.001 and $\alpha = 0.001$.

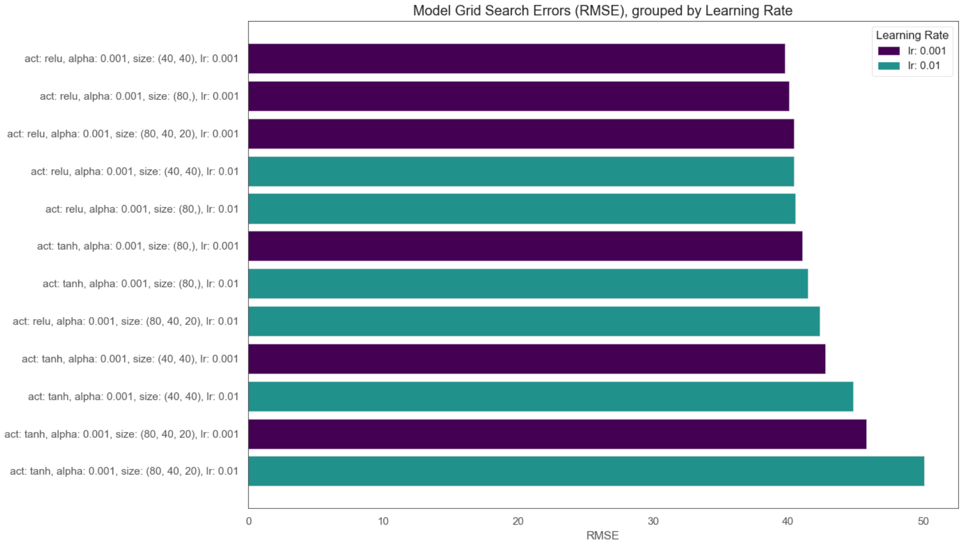

Figure 5.2: Grid Search RMSE for the extended MLP model.

5.2.2 Performance Evaluation

Once the optimal configuration was determined, the model was retrained and evaluated on both the training and test sets. As before, we benchmark its performance against both the BS model and the baseline MLP. The following table summarizes the obtained performance metrics:

Model	Training			Test		
	MAE	**RMSE**	**R^2**	**MAE**	**RMSE**	**R^2**
Black-Scholes	10.72	80.39	0.75	10.75	87.17	0.71
Baseline MLP	12.26	52.78	0.89	12.51	57.39	0.87
Extended MLP	**7.22**	**38.64**	**0.932**	**7.20**	**41.92**	**0.921**

Table 5.2: Performance comparison between BS, Baseline MLP and Extended MLP.

In addition to these global metrics, we again analyzed the models' performance at the individual contract level to evaluate how frequently each approach produced better predictions. Specifically, we computed the number of test cases in which the BS formula yielded a prediction closer to the observed mid-price than the extended MLP, and vice versa.

The analysis shows that the BS model outperformed the MLP in **15,742** instances, with an average improvement margin of **4.24** and a median of **1.66** in absolute error. Conversely, the MLP produced more accurate predictions in **9,557** cases, achieving a substantially larger average improvement of **12.13** and a median of **2.09**.

5.2.3 Discussion

The results in Table 5.2 demonstrate a clear performance gain from incorporating additional market information into the model. The extended MLP significantly outperforms both the baseline MLP and the BS formula across all evaluation metrics. Notably, the RMSE is reduced by nearly 27% compared to the baseline, and the coefficient of determination (R^2) increases to over 0.92 on the test set, suggesting a superior ability to capture the complex dependencies inherent in real-world option prices.

Moreover, the MAE is also substantially reduced, indicating that the improvements extend beyond isolated outlier scenarios and affect the overall distribution of predictions.

When analyzing the individual prediction errors, we observe a similar pattern to that observed with the baseline model: while BS yields a lower absolute error in a greater number of cases (15,742 vs. 9,557), the extended MLP achieves much larger improvements in the subset of cases where it performs better. This asymmetric behavior suggests that neural networks are particularly effective at capturing pricing regimes where the simplifying assumptions of the BS formula, such as constant volatility or frictionless markets, break down.

In summary, these results reinforce the added value of incorporating auxiliary features beyond the standard inputs and highlight the expressive capacity of neural networks when provided with richer information. This motivates the next stage of the analysis, where we investigate in detail the sensitivity of the models to each input variable through interpretability techniques.

Chapter 6

Interpretability Analysis

This chapter explores the interpretability of the Baseline Model developed in Chapter 5. In particular, we aim to understand how variations in input features affect the predicted option prices. We leverage techniques such as partial derivative analysis and *alpha curves* [27], with a special focus on comparing their financial significance with classical instruments like the Greeks.

In order to accomplish this task effectively, we will utilize the novel *Neuralsens* Python library, presented in the paper "Neuralsens: Sensitivity Analysis of Neural Networks". This library provides a framework for analyzing NN models, enabling us to evaluate variable importance based on sensitivity measures and characterize relationships between input and output variables.

6.1 Motivation and Background

The motivation behind this chapter is to enhance our understanding of how NNs, particularly those used in option pricing, can be interpreted. While these models often achieve high predictive accuracy, their complexity can obscure the underlying mechanisms driving their predictions. By employing sensitivity analysis techniques, we aim to shed light on the contributions of individual input features to the model's output. This interpretability is crucial for several reasons:

1. **Risk Management**: Understanding how input features influence option prices can help traders and risk managers make informed decisions.

2. **Model Validation**: Sensitivity analysis can serve as a tool for validating the model's behavior, ensuring that it aligns with financial intuition and theoretical expectations.

3. **Regulatory Compliance**: In some jurisdictions, financial institutions are required to provide explanations for their models' predictions, making interpretability a regulatory necessity [22].

While classical sensitivity analysis techniques such as the Greeks are well-established in the context of the BS model and widely used by investors and risk managers, they tend to capture the average influence of input variations under model assumptions. However, these global measures often fall short in providing insights into local behavior, particularly in extreme scenarios where the relevance of certain inputs may change drastically. As NNs are capable of capturing highly nonlinear relationships, it becomes increasingly important to develop localized interpretability tools that reflect how the model behaves in specific regions of the input space [16].

6.2 Methodology

To begin our interpretability analysis, we will first load the Baseline Model and the dataset used for training in Chapter 5. The model is a MLP trained to predict option prices based on various input features, including underlying asset price, strike price, time to maturity, risk-free interest rate, and volatility. Its architecture consists of one hidden layer with 60 neurons and uses the Tanh activation function. The model was trained using the Adam optimizer with a learning rate of 0.001 and a regularization parameter of 0.01.

6.2.1 Partial Derivatives as Sensitivity Measures

Partial derivatives quantify how much the model's prediction changes with an infinitesimal change in each input feature. Formally, the sensitivity of the output y_k of the k^{th} neuron in the output layer with respect to the input x_i of the i^{th} neuron in the input layer, evaluated at a specific input sample $\mathbf{x}_n$, is defined as:

$$s_{ik}\big|_{\mathbf{x}_n} = \frac{\partial y_k}{\partial x_i}(\mathbf{x}_n)$$

This expression quantifies how much the predicted option price y_k would change given a marginal variation in input feature x_i, when evaluated at the specific instance $\mathbf{x}_n$. To compute this sensitivity, the method `jacobianmlp` applies the chain rule to propagate gradients through the network's internal layers, recursively combining weight connections and activation function derivatives at each step.

Then we can access the returned object to obtain a summary of the sensitivities for each input feature across all samples in the dataset. This summary includes the mean, the standard deviation, and the squared mean of the sensitivities, providing insights into both the average influence of each input feature and its variability across different instances.

Input Variable	**Mean**	**Std**	**Mean Squared**
Spot Price (S)	325.35	179.19	371.43
Strike Price (K)	-251.83	169.08	303.33
Time to Maturity (T)	37.01	74.64	83.31
Implied Volatility (σ)	-35.79	349.43	351.26

Table 6.1: Summary of first-order sensitivity measures with respect to the predicted call price. Values represent average partial derivatives across the test dataset.

6.2.2 Alpha Curves as Visual Sensitivity Measures

To complement the numerical summary of partial derivatives, we employ a visual interpretability technique known as **alpha curves**. Introduced in the NeuralSens framework [28], alpha curves provide a systematic way to analyze the global sensitivity of a NN model with respect to each input variable by capturing not only the average effect but also the distribution of local variations across the input space.

Formally, for each input variable x_j, the alpha curve is defined as the sequence of α-means of the absolute value of the partial derivatives:

$$S_j^\alpha = \left(\frac{1}{N} \sum_{i=1}^{N} \left| \frac{\partial f(\mathbf{x}^{(i)})}{\partial x_j} \right|^\alpha \right)^{1/\alpha}$$

where N is the number of samples, $f(\mathbf{x})$ is the model's prediction, and $\alpha \in \mathbb{R}^+$ controls the aggregation behavior.

By varying the value of α, the curve reveals different aspects of the derivative distribution:

- For $\alpha = 1$, the curve reflects the mean absolute sensitivity (robust to outliers).
- For $\alpha = 2$, it corresponds to the root mean square sensitivity (more sensitive to large values).

- As $\alpha \to \infty$, the curve converges to the maximum observed absolute sensitivity.

Thus, alpha curves allow us to go beyond a single scalar importance score and instead characterize the distributional shape of the model's sensitivity to each input. A relatively flat alpha curve indicates that the variable has a stable, linear effect across all samples. In contrast, a steeply increasing alpha curve suggests that the variable has localized, high-impact regions where its influence spikes, indicative of nonlinearity or potential instability.

These curves provide an aggregated view of how sensitivity varies across different regions of the input space. In the following section, we analyze their variation across α values to uncover stability patterns and localized behaviors.

6.3 Global Analysis of Sensitivities

Having introduced the concept of alpha curves as a visual tool to summarize model sensitivity, we now turn to a more detailed analysis of how these sensitivities behave globally. This involves not only examining how sensitivity changes across different alpha values, but also looking at the raw distributions of partial derivatives to uncover irregularities, dominant patterns, and localized effects that may not be apparent from summary metrics alone.

6.3.1 Stability Across Alpha Values

While alpha curves already provide an overview of how sensitivity evolves with increasing α, it is useful to examine the rate of variation between successive α values. Variables whose sensitivity increases rapidly with α are indicative of strong local effects, whereas stable or flat alpha curves reflect globally consistent contributions.

In Figure 6.1, we observed that the curve for implied volatility (σ) not only reaches the highest asymptotic value but also shows the most pronounced curvature. This suggests that while σ may not always dominate in average sensitivity, it becomes highly influential in localized regions of the input space, consistent with its known financial behavior in ATM scenarios.

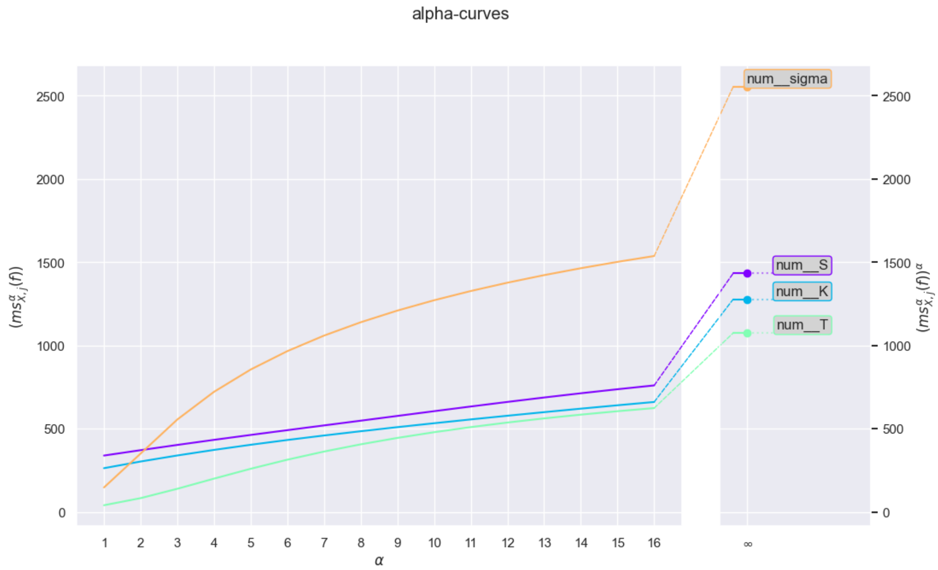

Figure 6.1: Alpha curves for each input variable in the Baseline MLP model.

In contrast, the curves for spot price (S) and strike price (K) grow more slowly and consistently, indicating that their impact is less dependent on specific configurations of input variables. The time to maturity (T) curve is the flattest for low α but begins to rise more noticeably for higher values, hinting at conditional relevance near expiration.

This analysis highlights the value of exploring the full spectrum of α rather than relying solely on mean sensitivities (e.g., $\alpha = 1$), as relevant variables may only reveal their influence under certain conditions.

6.3.2 Distribution of Partial Derivatives

To further analyze the distribution of partial derivatives, we can visualize the distribution of values for each input variable across the test dataset. This allows us to identify the range and variability of sensitivities, as well as potential outliers that may indicate regions of high sensitivity or instability in the model's predictions.

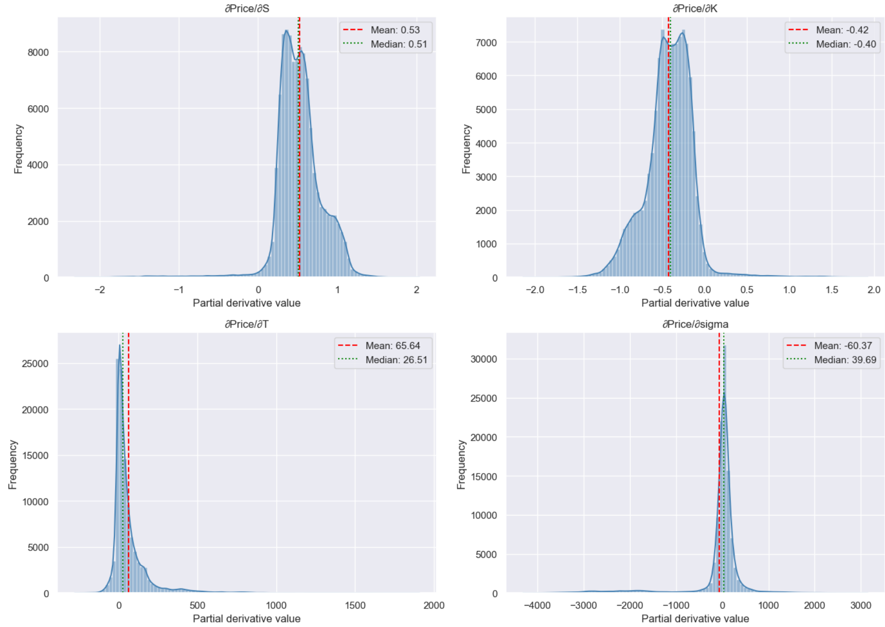

Figure 6.2: Distribution of partial derivatives for each input variable in the Baseline MLP model.

Figure 6.2 presents the histograms of the partial derivatives with respect to each input variable. As expected, most values concentrate around their respective means and medians, indicating that the model's sensitivity remains relatively stable for the majority of input samples.

However, the distribution of the partial derivative with respect to implied volatility ($sigma$) stands out for its wider spread along the x-axis, reflecting the existence of a non-negligible number of cases where the model exhibits a high degree of local non-linearity in response to changes in volatility.

Similarly, the partial derivative with respect to the spot price (S) displays a noticeable spread, although narrower than that of σ, suggesting that this variable also contributes to significant variation in model sensitivity across the input space and may warrant further investigation into its non-linear impact.

6.3.3 Global Summary Statistics

In addition to the distributional and visual sensitivity analyses, we compute global summary statistics to quantify the differences between the NN sensitivities and the classical Greeks derived from the BS model.

Sensitivity	BS Greeks Median	BS Greeks Std	NeuralSens Median	NeuralSens Std
$\Delta \mid \frac{\partial Price}{\partial S}$	0.592	0.35	0.505	0.289
ν (Vega) $\mid \frac{\partial Price}{\partial \sigma}$	15.04	137.18	39.69	589.434
$\Theta \mid \frac{\partial Price}{\partial T}$	14.97	111.68	26.51	132.39

Table 6.2: Median and standard deviations of Greeks and NN partial derivatives.

From these aggregated statistics, shown in Table 6.2, we observe that the NN displays similar median sensitivity to the underlying price (S) compared to the BS model, but shows higher median senstivities to both implied volatility (σ) and time-to-maturity (T), as well as substantially larger standard deviations. This indicates that the neural network not only assigns greater importance to these variables on average, but also captures a wider variability in their influence across different regions of the input space. Such behavior suggests that the NN has learned to model complex, non-linear interactions that are not fully captured by the parametric structure of the Black-Scholes formula.

6.4 Local Analysis of Sensitivities

To complement the global analysis, we will now examine individual input samples that exhibit atypical sensitivity patterns. The goal is to gain deeper insight into specific regions of the input space where the model exhibits strong nonlinear behavior or localized instability, phenomena that may be obscured when averaging over the full dataset.

As observed in Figure 6.2, most input samples cluster around the mean and median values of the partial derivatives, but some cases emerge as clear outliers, particularly for the input variables σ, T, and S. These outliers can provide valuable information about how the model behaves in edge scenarios, potentially revealing cases of financial or numerical significance.

To formally identify outliers, we adopt a standard statistical rule based on standard deviation: a sample is classified as an outlier for a given sensitivity if its partial derivative exceeds the mean by more than three standard deviations, i.e.,

$$\left| \frac{\partial C}{\partial x_i} - \mu_i \right| > 3\sigma_i,$$

where μ_i and σ_i denote the sample mean and standard deviation of the sensitivity for input variable x_i. This criterion allows us to focus on those instances where the model's sensitivity deviates most strongly from its typical behavior.

The empirical distributions from which these thresholds are derived are illustrated in Figure 6.2. As can be observed, while most partial derivatives concentrate around their respective means, a small number of extreme values emerge in the tails of the distributions, motivating the identification of these samples as local sensitivity outliers.

Our main goal in exploring these edge cases is twofold: first, to understand whether the MLP model exhibits consistent and interpretable behavior in regions of high local sensitivity; and second, to assess whether its performance in these scenarios surpasses or falls short of that of the classical BS model. This comparison is especially relevant in stress conditions, where pricing accuracy becomes critical.

To this end, we examine the local input context and the resulting partial derivatives in each case, and we compare the predicted prices generated by both models against the true market prices. Visualizations of local sensitivity profiles and selected input configurations are provided for each case, along with a brief financial interpretation of the observed model behavior.

6.4.1 Case 1: Outlier in Sigma Sensitivity

In this selected case, the partial derivative of the predicted option price with respect to implied volatility (σ) was identified as an outlier. The input values and model predictions for this sample are presented in Table 6.3.

Metric	S	K	T	σ	Mid Price	BS	MLP
Value	167.10	87.50	0.532	0.00001	33.80	81.67	48.07

Table 6.3: Input features and model predictions for the σ-sensitivity outlier.

This input configuration corresponds to a scenario where the implied volatility is nearly zero. From a financial standpoint, such a low value of σ is highly atypical, as it implies an almost riskless market environment. Under this condition, the BS formula tends to simplify drastically: if the option is **ITM**, its price approaches the intrinsic value, i.e., $C \approx S - K$ for a call; if the option is **OTM**, its value tends to zero.

In the present case, the option is clearly ITM ($S \gg K$), so BS estimates the price very close to the intrinsic value, disregarding any potential variation due to volatility. This leads to a significant overestimation compared to the observed market price. The MLP model, however, appears to handle this extreme input more robustly, yielding a prediction closer to the market mid price and demonstrating reduced sensitivity to an implausibly low volatility input.

This suggests that the NN may have implicitly learned to discount extreme input configurations when they do not align with realistic market behavior, offering a form of regularization or robustness not present in the analytical BS formula.

Moreover, to analyze how the predicted sensitivities compare with the BS Greeks, we compute the partial derivatives of the predicted option price with respect to each input variable. The results are summarized in Table 6.4.

Sensitivity	**BS Greeks**	**NeuralSens Partial Derivatives**
Δ | $\frac{\partial Price}{\partial S}$	1.0000	0.6809
ν (Vega) | $\frac{\partial Price}{\partial \sigma}$	0.0000	-2762.37
Θ | $\frac{\partial Price}{\partial T}$	3.8444	29.3248

Table 6.4: Comparison between BS Greeks and MLP partial derivatives for the σ-sensitivity outlier.

As expected, the BS Vega converges to zero when volatility approaches zero and assigns full exposure to the underlying asset price ($\Delta = 1$). This behavior can be further explained by considering the analytical limit of the Black-Scholes formula when $\sigma \to 0$. In this extreme regime, the option price simplifies as:

$$C(S) \approx \max\left(S - Ke^{-rT}, 0\right),$$

which implies that the pricing function essentially becomes a deterministic linear combination of S and K discounted at the risk-free rate. Under this approximation, the relationship between the option price and the underlying asset becomes almost perfectly linear, leading to a Delta close to 1.

However, the MLP is able to capture that even in low-volatility regimes, small nonlinearities still exist due to market effects, discrete trading, or non-continuous adjustments in implied volatilities for extreme moneyness. As a result, the MLP assigns a slightly lower Delta ($\Delta \approx 0.68$), indicating that it does not fully adhere to the rigid linear assumption of BS but instead learns a smoother sensitivity profile that may better reflect empirical market behavior under such edge cases.

6.4.2 Case 2: Outlier in Time to Maturity Sensitivity

In this case, we analyze a data point where the partial derivative of the predicted option price with respect to time to maturity (T) is identified as an outlier. The input configuration and resulting predictions are shown in Table 6.5.

Metric	S	K	T	σ	Mid Price	BS	MLP
Value	576.22	490.00	0.033	1.4137	111.50	108.42	111.40

Table 6.5: Input features and model predictions for the T-sensitivity outlier.

This configuration corresponds to a scenario where the time to maturity is very short ($T \approx 12$ days), implying that the option is close to expiration. In financial theory, such options typically carry limited time value, and their price converges rapidly toward their intrinsic value $(S - K)$ [29]. In this case, both the Black-Scholes model and the MLP provide fairly accurate predictions, differing by less than 3 units with respect to the observed mid price.

However, when analyzing the sensitivities, noticeable differences emerge between the classical Greeks and the sensitivities learned by the neural network, as summarized in Table 6.6.

Sensitivity	**BS Greeks**	**NeuralSens Partial Derivatives**
$\Delta \mid \frac{\partial Price}{\partial S}$	0.7782	0.6860
ν (Vega) $\mid \frac{\partial Price}{\partial \sigma}$	31.0773	19.2093
$\Theta \mid \frac{\partial Price}{\partial T}$	683.4634	463.4519

Table 6.6: Comparison between BS Greeks and Neural Network partial derivatives for the T-sensitivity outlier.

In this scenario, the MLP exhibits the same qualitative behavior as BS, correctly identifying the dominant role of time-to-maturity near expiration. However, its sensitivities are systematically smoother and less extreme than those produced by the analytical formula. This suggests that the neural network is implicitly regularizing the highly nonlinear interactions between time-to-maturity and volatility, likely as a consequence of the finite sample regime and the regularization applied during training. Such behavior may be advantageous in practice, as it prevents the model from overreacting to marginal changes in T that may lead to instability in risk management applications.

6.4.3 Case 3: Outlier in Spot Price Sensitivity

This final case corresponds to an outlier in the partial derivative of the predicted option price with respect to the spot price of the underlying asset (S). The input configuration and corresponding model predictions are summarized in Table 6.7.

Metric	S	K	T	σ	Mid Price	BS	MLP
Value	1241.47	780.00	2.526	0.5267	604.50	652.76	639.20

Table 6.7: Input features and model predictions for the S-sensitivity outlier.

This configuration represents a long-dated, deep ITM option, where $S \gg K$ and T is over two years. In such cases, the option price is strongly driven by the intrinsic value, but the sensitivity to the spot price remains important due to the potential for further price movement over time.

In this case, the NN outperforms the BS model, providing a prediction significantly closer to the observed market mid price. This suggests that the MLP may be capturing pricing effects that are not fully reflected in the BS formula, potentially incorporating complex interactions between volatility, time, and moneyness that deviate from the classical assumptions made by the BS model.

To further analyze difference between sensitivities, we compute the partial derivatives of the predicted option price with respect to each input variable. The results are summarized in Table 6.8.

Sensitivity	**BS Greeks**	**NeuralSens Partial Derivatives**
$\Delta \mid \frac{\partial Price}{\partial S}$	0.8664	1.3980
ν (Vega) $\mid \frac{\partial Price}{\partial \sigma}$	425.3361	261.4783
$\Theta \mid \frac{\partial Price}{\partial T}$	63.3687	212.9347

Table 6.8: Comparison between BS Greeks and Neural Network partial derivatives for the S-sensitivity outlier.

The most prominent difference appears in the sensitivity to the underlying asset, where the NN assigns a marginal sensitivity $\Delta > 1$, while BS constrains this value below one due to its linear payoff assumption and constant volatility framework. This suggests that the NN may implicitly incorporate nonlinear payoff effects or market frictions that are not modeled in classical arbitrage-free theory.

Similarly, discrepancies are observed in the sensitivities to volatility and time-to-maturity, further indicating that the MLP captures a distinct sensitivity structure that may better reflect market behavior for long-dated options with significant intrinsic value.

These local case studies highlight how MLP may exhibit distinct sensitivity structures when compared to classical Greeks. In the following section, we extend this analysis to a global perspective, using alpha curves to explore how sensitivities behave across broader regions of the input space

6.5 Discussion and Insights

The local sensitivity analysis performed in this chapter provides valuable insights into how neural networks, specifically the trained MLP model, behave in comparison to the classical Black-Scholes model when evaluated under extreme or challenging input conditions. Several important patterns emerge from these case studies:

First, the MLP demonstrates remarkable flexibility in adapting to edge cases that often challenge parametric models like BS. For instance, in the σ-outlier case, where the implied volatility approaches zero (an extremely rare market condition), Black-Scholes converges mechanically to the intrinsic value, while the MLP still corrects its output and maintains a more reasonable pricing closer to market observations. This suggests that the NN has implicitly learned to discount unrealistic volatility inputs, incorporating a form of data-driven regularization not present in the analytical formula.

Second, the distributional analysis of partial derivatives exposed substantial heterogeneity in the NN's sensitivities across different regions of the input space. In particular, the wider dispersion of sensitivities with respect to σ and T highlights that the network is capable of detecting local regimes where volatility and time have a disproportionate influence on option prices, a behavior that may better reflect certain real-world market conditions.

Third, in the T-outlier scenario, where time-to-maturity is extremely short (approximately 12 days), both models achieve high pricing accuracy. However, the sensitivities reveal important differences. While BS produces extreme Θ values reflecting the rapid time decay near expiration, the MLP assigns lower but still dominant sensitivity to time, suggesting a smoother treatment of near-maturity risk. The same regularization is observed across Delta and Vega, with the MLP consistently assigning slightly more conservative sensitivities. This behavior may be advantageous in practice, as it prevents overreaction to marginal changes in T, reducing the likelihood of unstable hedging decisions near expiry.

Fourth, the introduction of alpha curves proved more useful than point estimates of sensitivities, offering a richer and more stable representation of how variable importance changes under different data regions. The growth of the σ alpha curve with increasing α strongly suggests that volatility plays an outsized role in driving extreme sensitivity behaviors in certain subregions of the input space.

Importantly, these results suggest that, precisely in those regions of the input space where classical models such as BS become less reliable, the partial derivatives extracted from the MLP may serve as more informative sensitivity indicators than the standard Greeks. This opens the door to leveraging NN sensitivities not only for pricing, but also as potentially superior risk management metrics in extreme market conditions where the limitations of parametric models become more pronounced.

In summary, while the BS model continues to offer interpretable, stable, and well-behaved local sensitivities, NNs provide an extended capacity to capture nonlinear interactions that may be more aligned with empirical option pricing behavior. However, this added flexibility also introduces interpretability challenges that make sensitivity analysis tools, such as partial derivatives and alpha curves, crucial for model validation and practical deployment.

Overall, these findings reinforce the value of hybrid approaches that combine data-driven models with domain knowledge and interpretability frameworks, paving the way for more flexible yet transparent option pricing methodologies.

Chapter 7

Conclusions and Future Work

In this final chapter, we synthesize the key contributions and findings of this thesis, emphasizing both the predictive improvements achieved through MLPs and the novel interpretability insights derived from sensitivity analysis. Additionally, we discuss the practical relevance of these results, acknowledge limitations of the current work, and outline promising avenues for future research.

7.1 Summary of Results

The objectives of this thesis were two: (1) to evaluate whether MLPs can approximate and improve upon traditional analytical models for option pricing; and (2) to explore how sensitivity analysis techniques may provide interpretability tools that help explain the predictions of MLPs in a financially meaningful way.

The empirical evaluation carried out on a large-scale dataset of S&P 500 index options confirmed that:

- **Predictive Performance:** Both the baseline and extended MLP architectures significantly improved predictive accuracy compared to the BS model. While BS outperformed the MLP on a larger number of individual predictions, the MLP achieved substantially larger improvements in the subset of cases where it performed better, leading to lower global RMSE and higher R^2.
- **Sensitivity Behavior:** The MLP successfully captured more complex, non-linear relationships across the input space that were not modeled by BS.

Sensitivity analysis revealed that the MLP assigns higher median sensitivities to implied volatility (σ) and time-to-maturity (T), suggesting that it internalizes additional dependencies present in real market data.

- **Interpretability:** The use of partial derivatives and alpha curves enabled us to obtain interpretable sensitivity measures from the trained MLP, providing a direct comparison with classical Greeks.

- **Robustness in Edge Cases:** The local analysis of extreme scenarios (e.g., near-zero volatility, short time-to-maturity, deep ITM contracts) demonstrated that the MLP exhibited flexible and adaptive behaviors, sometimes outperforming BS precisely where the analytical model's assumptions fail.

7.2 Key Insights and Interpretation

The findings of this thesis underline that MLPs, despite their black-box nature, can capture rich and complex pricing dynamics that are not accessible through traditional analytical models. The partial derivatives extracted from the MLP provided a first-order interpretability layer, allowing us to analyze sensitivities in a way that is analogous, but not identical, to classical Greeks.

In particular, the analysis of global sensitivity measures revealed that while both models exhibit broadly similar exposure to the spot price, the MLP consistently assigns greater importance to volatility and time-to-maturity. This reflects its ability to internalize more complex interactions present in real markets, especially in regions where volatility surfaces or time decay behave in a nonlinear way. The use of alpha curves proved especially valuable to capture these nonlinearities, highlighting how sensitivity magnitudes evolve as we consider higher-order aggregation levels.

At the local level, the study of specific outlier cases confirmed that the MLP exhibits adaptive behaviors. In some regions, such as deep ITM options with extreme volatility configurations, the MLP provided more realistic price estimations than BS. However, the analysis also revealed that in certain edge cases, particularly for short-term maturities combined with high volatility, the MLP may exhibit overly strong sensitivity reactions, pointing to the importance of carefully monitoring the model's behavior in underrepresented regions of the data.

7.3 Practical Implications for Finance

From a practical standpoint, the results obtained suggest that MLP models, when combined with proper interpretability tools, can serve as valuable complements to classical pricing frameworks. Their ability to capture non-linearities makes them particularly well-suited for markets or products where the assumptions of constant volatility, frictionless trading, or log-normal price dynamics are violated.

The partial derivatives extracted from MLPs could also serve as alternative sensitivity measures, particularly in regions where classical Greeks may provide misleading or unstable estimates. This opens the possibility of using data-driven models not only for pricing but also for risk management, hedging, and stress testing purposes, provided that their interpretability and stability are carefully monitored.

Moreover, the combination of MLPs with sensitivity analysis frameworks such as NeuralSens allows practitioners to bridge the gap between predictive accuracy and financial explainability, addressing one of the main obstacles preventing the widespread adoption of ML models in regulated financial environments.

7.4 Limitations

Despite the promising results, several limitations of this study should be acknowledged. The dataset used consisted of a single snapshot of the option market, limiting the capacity to capture temporal dynamics such as volatility clustering. Additionally, the MLP was trained without explicit enforcement of financial constraints such as no-arbitrage conditions, monotonicity, or convexity properties, which could help ensure consistency with fundamental pricing principles.

The interpretability analysis was primarily applied to the baseline MLP architecture. Extending these techniques to more complex models incorporating additional features may require further methodological developments. Finally, the empirical study focused exclusively on European call options; future extensions should consider different option types and exotic derivatives to validate the generalizability of the approach.

7.5 Future Research Directions

Building on the insights obtained in this work, several directions emerge for future research. Incorporating historical data and temporal dependencies would allow the models to capture dynamic volatility patterns and regime shifts, providing more realistic pricing under evolving market conditions. Exploring hybrid modeling approaches that combine the flexibility of MLPs with embedded financial knowledge may further enhance both predictive accuracy and theoretical consistency.

Moreover, extending sensitivity-based interpretability frameworks to deeper or more complex neural architectures could improve the financial explainability of increasingly powerful models. Applying these methods to other asset classes or derivative products would also help validate the robustness of NN approaches across a broader financial landscape.

Finally, translating MLP sensitivities into actionable risk management metrics remains an important avenue for transforming these models into production-ready tools that can complement or even enhance the classical Greek framework used by practitioners today.

Bibliography

[1] F. Black and M. Scholes, "The princing of options and corporate liabilities," *Journal of political economy*, vol. 81, no. 3, pp. 637–654, 1973.

[2] G. Poitras, "Vinzenz bronzin's option pricing models," in *Vinzenz Bronzin's Option Pricing Models*, W. Hafner and H. Zimmermann, Eds. Springer, 2009, ch. 18, pp. 489–518.

[3] R. H. Tawney, "Introduction," in *A Discourse upon Usury by T. Wilson (1572).* London: Frank Cass, 1962, originally published in 1925.

[4] L. Bachelier, "The theory of speculation," Société Mathématique de France, Tech. Rep., 1900.

[5] Options Clearing Corporation, "Monthly & weekly volume statistics," https://www.theocc.com/market-data/market-data-reports/volume-and-open-interest/monthly-weekly-volume-statistics, 2024, accessed: 2025-05-04. [Online]. Available: https://www.theocc.com/market-data/market-data-reports/volume-and-open-interest/monthly-weekly-volume-statistics

[6] G. Rigopoulos, "A multilayer perceptron learning architecture for black-scholes call options pricing," *International Journal of Computer Science and Mobile Computing*, vol. 12, no. 8, 2023.

[7] A. Ke and A. Yang, "Option pricing with deep learning," *Department of Computer Science, Standford University, In CS230: Deep learning*, vol. 8, pp. 1–8, 2019.

[8] Z. İltüzer, "Option pricing with neural networks vs. black-scholes under different volatility forecasting approaches for bist 30 index options," *Borsa Istanbul Review*, vol. 22, pp. 725–742, 7 2022.

[9] J. E. Berger, "Pricing european options with google automl, tensorflow, and xgboost," *arXiv preprint arXiv:2307.00476*, 2023.

[10] J. M. Benítez, J. L. Castro, and I. Requena, "Are artificial neural networks black boxes?" *IEEE Transactions on neural networks*, vol. 8, no. 5, pp. 1156–1164, 1997.

[11] R. Aroussi, "yfinance: Download market data from yahoo! finance's api," 2019, accessed: 2025-04-14. [Online]. Available: https://github.com/ranaroussi/yfinance

[12] M. Del Giudice, F. Evangelista, and M. Palmaccio, "Defining the black and scholes approach: a first systematic literature review," *Journal of Innovation and Entrepreneurship*, vol. 5, pp. 1–13, 2015.

[13] Z. Janková, "Drawbacks and limitations of black-scholes model for options pricing," *Journal of Financial Studies and Research*, vol. 2018, pp. 1–7, 2018.

[14] S. N. Neftci, *Principles of Financial Engineering*. Academic Press, 2008, see Section 11.2 "How Can We Define Moneyness?", pp. 458–460.

[15] J. Paunović, "Options, greeks, and risk management," *The European Journal of Applied Economics*, vol. 11, no. 1, 2014.

[16] A. Kumar, "A study on risk hedging strategy: efficacy of option greeks," *Abhinav National Monthly Refereed Journal of Research in Commerce & Management*, vol. 7, no. 4, pp. 77–85, 2018.

[17] W. Li, "Application of machine learning in option pricing: a review," in *2022 7th International Conference on Social Sciences and Economic Development (ICSSED 2022)*. Atlantis Press, 2022, pp. 209–214.

[18] J. M. Hutchinson, A. W. Lo, and T. Poggio, "A nonparametric approach to pricing and hedging derivative securities via learning networks," *The journal of Finance*, vol. 49, no. 3, pp. 851–889, 1994.

[19] S. K. Mitra, "An option pricing model that combines neural network approach and black scholes formula," *Global Journal of Computer Science and Technology*, vol. 12, no. 4, pp. 6–16, 2012.

[20] M. Can and S. Fadda, "A nonparametric approach to pricing options learning networks," *Southeast Europe Journal of Soft Computing*, vol. 3, no. 1, pp. 45–57, 2014.

[21] J. Ruf and W. Wang, "Neural networks for option pricing and hedging: a literature review," 2020. [Online]. Available: https://arxiv.org/abs/1911.05620

[22] O. Sarioguz and E. Miser, "Integrating ai in financial risk management: Evaluating the effects of machine learning algorithms on predictive accuracy and regulatory compliance," *International Journal of Science and Research*, 2024.

[23] R. Pimentel, M. Risstad, S. Rogde, E. S. Rygg, J. Vinje, S. Westgaard, and C. Wu, "Option pricing with deep learning: a long short-term memory approach," *Decisions in Economics and Finance*, pp. 1–32, 2025.

[24] L. Liang and X. Cai, "Time-sequencing european options and pricing with deep learning–analyzing based on interpretable ale method," *Expert Systems with Applications*, vol. 187, p. 115951, 2022.

[25] L. Richardson, "Beautiful soup: Html parsing library for python," 2004, accessed: 2025-04-14. [Online]. Available: https://www.crummy.com/software/BeautifulSoup/

[26] Trading Economics, "United states interest rate," https://tradingeconomics.com/united-states/interest-rate, 2025, accessed: 2025-05-08. [Online]. Available: https://tradingeconomics.com/united-states/interest-rate

[27] J. Pizarroso, "Neuralsens: sensitivity analysis of neural networks," *Journal of Statistical Software*, vol. 102, pp. 1–36, 2022.

[28] J. Pizarroso, D. Alfaya, J. Portela, and A. Muñoz, "Metric tools for sensitivity analysis with applications to neural networks," *Applied Soft Computing*, p. 113300, 2025.

[29] P. Falloon, "Options: Time value," http://demonstrations.wolfram.com/OptionsTimeValue/, 2011, wolfram Demonstrations Project.