UNIVERSITÀ CATTOLICA DEL SACRO CUORE

Faculty of: Scienze Matematiche, Fisiche e Naturali;

Scienze Bancarie, Finanziarie e Assicurative

Graduate degree in Applied Data Science for Banking and Finance

From Black-Scholes to Machine Learning: A Journey Through Option Pricing

Candidate: Maria Pia Bellini

Student ID: 5310781

Supervisor: Prof. Alessandro Musesti

Co-supervisor: Prof. Giovanni Guastella

Academic Year 2024–2025

Abstract

From Black-Scholes to Machine Learning: A Journey Through Option Pricing

Maria Pia Bellini

Prof. Alessandro Musesti

Prof. Giovanni Guastella

This thesis addresses the limitations of traditional option pricing models, such as the Black-Scholes framework, in accurately valuing options in modern financial markets. The study posits that the restrictive assumptions of these classical models fail to capture the real-world complexities of market dynamics.

To overcome this, the research investigates the efficacy of machine learning (ML) models as a superior alternative. The methodology involves applying various ML techniques, including neural networks and regression trees, to historical market data and comparing their performance against traditional models.

The results consistently show that ML models provide greater pricing accuracy and adaptability. In conclusion, this research confirms that machine learning is a powerful tool for pricing financial derivatives, representing a significant advancement over conventional methods and pointing toward a new frontier in quantitative finance.

Acknowledgements

Vorrei ringraziare il prof. Musesti e il prof. Guastella, che mi hanno seguita con saggezza nella stesura di questo lavoro.
Ringrazio poi la mia famiglia, che mi ha supportata e mi ha permesso di intraprendere questo percorso.
Papà, che mi ha aiutata a credere in me quando non riesco a farlo, e che mi ha spronato ad affrontare le situazioni con la giusta visione, anche quando volevo mollare tutto e tornare a casa (e i bonifici istantanei!).
Mamma, per i pacchi di cibo prima di ogni partenza e per i consigli sulle pulizie che mi hanno consentito di sopravvivere da sola.
Agnese, vittima dei miei (frequenti) esaurimenti nervosi, che mi sopporta sempre e cerca di aiutarmi anche quando non collaboro.
Stefania, per le nostre telefonate infinite e le risate che mi hanno regalato compagnia nei giorni di solitudine lontana da casa. Somi, per l'aiuto nei momenti difficili e il supporto nei momenti di panico, facendomi ridere e stare meglio, e tutti i meravigliosi momenti che abbiamo passato. Ma soprattutto vorrei ringraziare me stessa e dedicare questo traguardo a tutti i momenti che ho vissuto in questi ultimi due anni, che la mia testardaggine mi ha portato a trascorrere lontana da casa, a Brescia, una città in cui non conoscevo nessuno. Alle lotte con le suore, ai pasti della mensa divisi tra pranzo e cena, alle estati passate a lavorare per mettere da parte qualcosa, alla spesa all'Aldi con lo zaino, alle sere di solitudine, allo studio "matto e disperatissimo" e ai 29 rifiutati, alla frustrazione perché pretendevo sempre di più da me stessa.
Ma anche agli amici di Brescia: le colazioni prima degli esami, le storie in uni, i bro della mensa, le giornate passate in aula studio tra mille heets fumate, le serate in Carmine e le nottate da Alessio, gli accenti bresciani poco comprensibili e le gite al lago.

Contents

Chapter 1

Introduction

In a modern context, a financial system is made up of financial institutions that act as intermediaries between policymakers, companies, and consumers that alternate between their roles as savers and investors, channeling savings into investments in financial markets through the purchase and sale of financial products (Petters and Dong 2016). Financial markets come in many forms, but the ones that are most significant to us are the following.

- Stock markets, a public market where buyers and sellers trade shares of publicly-listed companies (Investopedia Staff 2024);
- Bond markets, dealing with governments and other bonds;
- Currency markets, where currencies are bought and sold;
- Commodity markets, on which the trading of physical assets such as oil, gold, or electricity takes place;
- Futures and options markets, where the derivative products are traded (Wilmott et al. 1995).

Derivative instruments play a crucial role, both as tools for risk hedging and as instruments for speculation and financial engineering (Eales and Choudhry 2003).

Options, in particular, allow an investor to define complex strategies by exploiting asymmetries in price movements and volatility expectations.

An accurate option price is crucial to ensure efficiency and stability in markets, while an inexact evaluation could generate arbitrages, portfolio distortions, and unexpected risks, influencing both individual operators and institutions.

For this reason, academic and applied research have always dedicated considerable efforts in the development of models capable of estimating the value of these instruments in a reliable way (C. W. Smith 1976).

The first significant turning point in studying option pricing was the introduction of the Black-Scholes model (1973), which has provided an elegant analytical solution based on the hypothesis of the perfect market, absence of arbitrage and lognormal dynamic of the underlying asset price. This model, as well as being widely adopted in practice, had represented the starting point for many extensions and adaptations (Henderson 2014).

In the next years, more approaches have been developed, such as the binomial model, Monte Carlo simulations, and more complex stochastic models (i.e. Heston and Bates), designed to relax too restrictive hypothesis. However, these models also present some limitations, in particular in the capacity to capture empiric phenomena observed in markets, such as volatility smile and skew.

In recent years, the availability of large amounts of data and progress in the machine learning field have opened up new perspectives. Nonparametric and data-driven models, such as neural networks and advanced regression tech-

niques, offer the possibility of learning complex relations between variables without imposing constraints too strict on the functional form (Ivașcu 2021).
This thesis aims to compare traditional option pricing models with approaches based on machine learning techniques, analyzing strengths, limitations, and potential applications.
The main objective is to evaluate if, and under which conditions, data-driven models may result in more accurate or more robust estimations with respect to classic models, especially in contexts of markets characterized by a strong volatility and non-linear relationships.
The thesis is divided into two main parts.
The first part presents the theoretical framework of reference, outlining the priority models of option pricing, from classic formulation such as Black-Scholes to more advanced models of stochastic modeling, and then introducing the most relevant machine learning techniques for the financial context.
The second part has an empirical approach and involves carrying out a comparative analysis using the software R. Both traditional and machine learning models will be implemented using a dataset of real options, and performances will be compared through accuracy and robustness metrics.
The thesis concludes with a summary of the results, a reflection on the contributions of the work, and some indications for possible future developments.

CHAPTER 2

Theoretical Foundations of Option Pricing

2.1 Call and Put Options

An option contract can be defined as a contract with the following conditions: the owner of the option can buy (call) or sell (put) a prescribed asset (**underlying asset**) for a prescribed amount (**exercise or strike price**), at a prescribed time in the future (**expiration date**).

That means that the holder of the option acquires the *right* to buy or sell the underlying asset and is not *obligated* to do so. The other party to the contract (**writer**) has the *obligation* to sell or buy the asset in case the holder decides to exercise the option. Moreover, the option contract has a price (**premium**) that must be paid at the beginning of the contract. Since it gives the holder the right to buy or sell without obligation, it has some value, and the risk taken by the writer must be compensated (Wilmott et al. 1995).

Table 2.1: Call and Put Options

	Buyer	Seller
Call option	Right to buy asset	Obligation to sell asset if option is exercised
Put option	Right to sell asset	Obligation to buy asset if option is exercised

Options are useful for investors mainly for two reasons: speculation and hedging.

- If the investor expects that the price of a particular stock will rise, he will buy some shares in that company. If he was right, he makes money, otherwise he loses money. That means that the investor is **speculating**.

  In that case, he can buy a call option whose value is equal to the current price. Conversely, if the investor expects the stock to fall, he can buy puts instead of selling the asset. If the investor already owns a long-term investment, he may want to protect against temporary falls in the share price, rather than liquidating his holdings and buying them later (Wilmott et al. 1995). Options can represent a less expensive tool to expose a portfolio to high risk.

- The term **hedging** refers to the reduction of risk by taking advantage of the correlations between the price movements of assets and options (Stulz 1984).

  A common hedging strategy is to use a *protective put*. An investor who owns a stock can buy a put option on that stock to protect against a potential drop in its price. This strategy establishes a "floor" for the portfolio's value as the put option's value increases as the stock price falls, offsetting the loss on the stock itself. The cost of this protection is the premium paid for the put option, which is often less expensive than selling the stock and risking missing out on a potential price increase.

Another hedging strategy is the *covered call*. In this case, an investor who owns a stock sells a call option on it. This generates income from the premium received, which can offset some of the potential loss if the stock price drops (Misra and Dalmia 2007).

There are essentially two types of options when it comes to exercising the right to choose: European and American options. The difference between the American and European options is about early exercise, not geography. A European option can be exercised only on the expiration date, and an American option allows holders to exercise the option rights at any time before and at the expiration date.

2.2 Put-call parity

Put-call parity is a fundamental principle in option pricing that establishes a relationship between the prices of a European call option and a European put option, when they share the same underlying asset, strike price, and expiration date (Klemkosky and Resnick 1979).

This principle states that a portfolio consisting of a long call and a short put should produce the same returns as owning a single share of the underlying stock, provided specific conditions are satisfied.

2.2.1 Risk-neutral probability

The concept of risk-neutral probabilities is widely used in option pricing and can be the underlying assumption in different option pricing models.

Risk-neutral probability is a theoretical probability measure of future outcomes that has been adjusted for risk. It is a concept used for asset pricing, particularly for derivatives, and is based on two key assumptions:
The current value of an asset is equal to its expected future pay-off, discounted at the **risk-free rate**. The market has **no arbitrage opportunities**. This means that it is impossible to make a risk-free profit.
Under this probability measure, all investors are assumed to be risk-neutral, meaning they do not require a risk premium to hold a risky asset. This allows for the simplification of complex pricing problems.
Another interpretation of put-call parity involves **forward contracts**, a forward contract is an over-the-counter (OTC) instrument between two parties that allows the buyer to buy or sell an asset at a specific price on a future date (Jarrow and Oldfield 1981). This type of contract can be used to lock in a specified price to avoid volatility. If the price of the underlying asset increases, the long position benefits, if the underlying asset price decreases, the short position benefits (CFI Staff 2025).
Put-call parity states that holding a short European put and a long European call of the same class is equivalent to holding a single forward contract on the same underlying asset. This holds true as long as the options and the forward contract share the same expiration date and the forward price is identical to the option strike price (Klemkosky and Resnick 1979).
The put-call parity can be expressed as the following equation.

$$\mathrm{C} + \mathrm{PV}(x) = \mathrm{P} + \mathrm{S} \tag{2.1}$$

where:

C = Price of the European call option

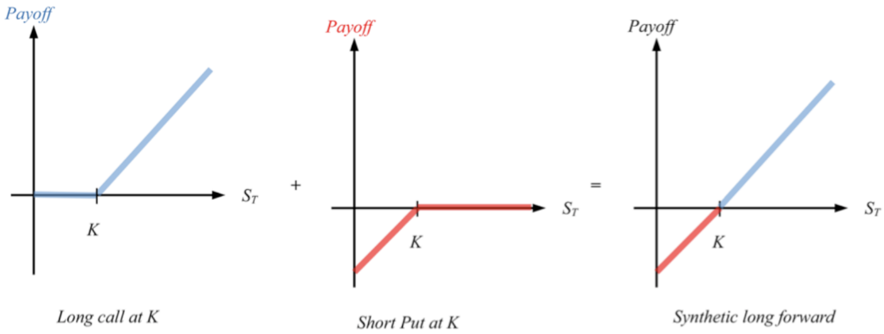

Figure 2.1: Put-call parity, (Debellefroid 2025)

PV(x) = Present value of the strike price (x), discounted from the value on the expiration date at the risk-free rate

P = Price of the European put

S = spot price or the current market value of the underlying asset

When the put-call parity equation holds, it indicates that the option market is in equilibrium and there are no opportunities for arbitrage. However, if the market prices of the put and call options deviate from this parity, an arbitrage opportunity arises. Traders can take advantage of this discrepancy by simultaneously buying the undervalued option and selling the overvalued one, thereby securing a risk-free profit.

2.3 Option value

The value of the option on the expiration date can be expressed as a function of the stock price and the exercise price (V. K. Smith 1983). To better understand this, consider the option value for the buyer of a call and put option X with an exercise price of €18:

When the stock price is at the exercise price of €18, both the call and put options have a value of zero.

Table 2.2: Options value, (Gupta 2017)

Stock price	€15	€16	€17	€18	€19	€20	€21
Call value	0	0	0	0	1	2	3
Put value	3	2	1	0	0	0	0

- Call option: The value of a call option for the buyer increases as the stock price rises above the exercise price. For example, if the stock price is €19, €20, or €21, the option's value is €1, €2, and €3 respectively, which is the difference between the stock price and the €18 exercise price. The buyer would not exercise the option if the stock price is below €18, so its value would be zero (V. K. Smith 1983).

- Put option: The value to the buyer of a put option is the opposite. It increases as the stock price falls below the exercise price. For example, if the stock price drops to €17, the option's value is €1, which is the difference between the €18 exercise price and the stock price. The buyer would not exercise if the price of the stock is greater than €18, so its value would be zero (V. K. Smith 1983).

This relationship between the option value, the price of the stock, and the exercise price can be visually represented in a position diagram:
The total pay-off of an option is the sum of the initial price and the value of the option when exercised.

2.4 American options

An **American option** has the additional feature that exercise is allowed at any time during the lifetime of the option.

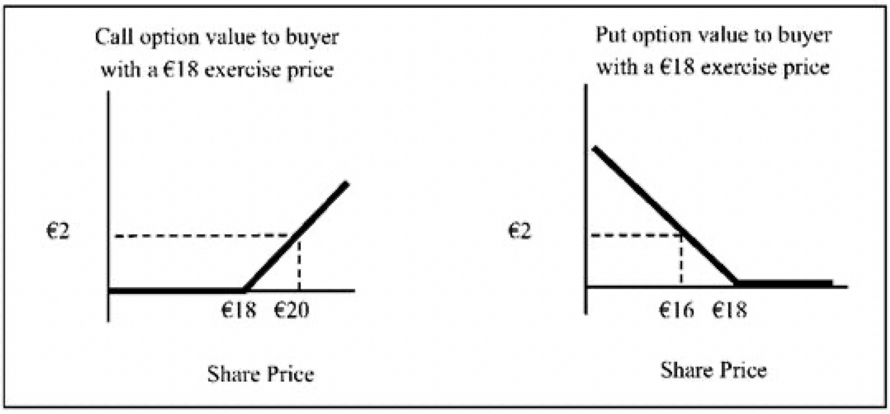

Figure 2.2: Option value for the holder, (Gupta 2017)

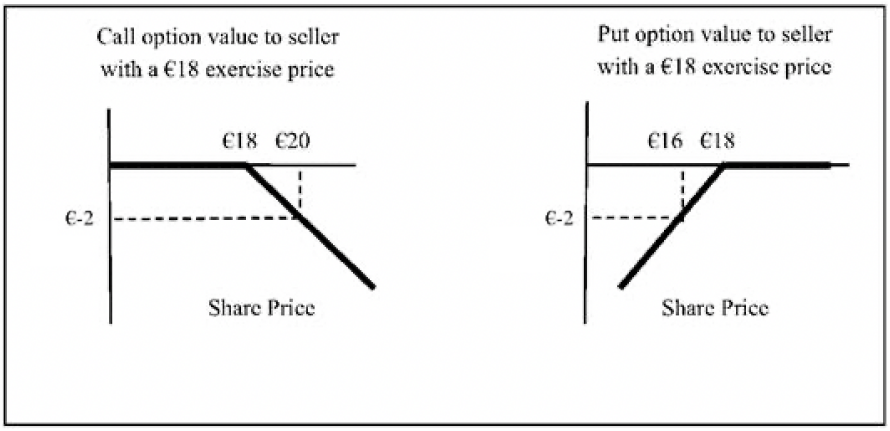

Figure 2.3: Option value for the writer, (Gupta 2017)

That implies that with an American option the investor can make profit as soon as the price is favorable, as well as taking advantage of dividend announcements. For that reason, American options are usually more valuable than Europeans, with an added premium or cost (James Chen 2022).

2.4.1 American Option value

Deciding when to exercise an American option depends on a simple comparison between the immediate pay-off and the potential future value of the option. At any point before expiration, the holder has a choice. They can exercise the option immediately to secure the current payoff, which is the difference between the stock price and the strike price. Alternatively, they can choose to wait, keeping the option alive in the hope that its value will increase further. The optimal strategy is to exercise the option only when the immediate pay-off is greater than the expected value of holding the option for a future period. By waiting, the investor foregoes the current gain, but retains the opportunity for a much larger profit if the stock price moves more favorably (Benninga and Wiener 1997).

We don't have a simple expression as for European options, but the value of American options can be easily calculated with backward recursion (Ju and Zhong 1999).

Consider an American option with price V, underlying asset S, and payoff of the option $\phi(S)$. Then we can write:

At t_n:

$$V_{t_n} = \phi(S_{t_n}) \tag{2.2}$$

At t_{n-1}, the option holder decides to exercise the option if the value of exercising the option is higher than the value of keeping it.

$$V_{t_{n-1}} = \max(\phi(S_{t_{n-1}}), e^{-r(t_n - t_{n-1})} \cdot E_{t_{n-1}}(\phi(S_{t_n})) \tag{2.3}$$

Where $\phi(S_{t_{n-1}})$ is the value of exercising the option and $e^{-r(t_n - tn-1)} \cdot E_{t_{n-1}}(\phi(S_{t_n}))$ is the value of keeping the option.

2.5 Option pricing models

Option pricing models are mathematical frameworks that are used to determine the theoretical value of a financial option. This "fair value" is an estimate of what an option should be worth based on a range of known inputs.

These models are essential tools for finance professionals because they allow for:

- **Valuation**: By comparing the model's output to the market price, professionals can identify if an option is over- or under-valued, informing their trading strategies.
- **Risk Management**: Models are integral to managing risk exposure, as they help measure how the price of an option will change in response to various market factors (CFI Staff 2025).

Chapter 3

Traditional Option Pricing Models

3.1 The Black-Scholes Model

The Black-Scholes model is used to price European call or put options where the underlying asset does not pay dividends or make other distributions (Shinde and Takale 2012).

3.1.1 The model assumptions

Before describing this model, it is fundamental to make some assumptions:

1. The asset price follows the lognormal random walk.

   A **random walk** is a process in which the change in the value of a variable over a given period of time is independent of all previous changes. The size and direction of these changes are unpredictable and can be considered random. The application of the random walk model to stock prices is based on the hypothesis of an efficient market. Therefore, future price movements are driven exclusively by new and unforeseen information, making them inherently unpredictable.

The term **lognormal** refers the fact that the natural logarithm of the price change of an asset over time follows a normal distribution. Consequently, the distribution of the asset price itself at a future point in time is considered lognormal. In simple terms, a lognormal distribution is one in which the logarithms of the values are normally distributed. This characteristic ensures that the price of the asset can never fall below zero, a property that aligns with the reality of the stock prices (SO Staff 2012).

2. The risk-free interest rate r and the volatility σ of the asset are known functions of time throughout the life of the option contract.

   Later we will relax the assumptions of the deterministic behavior of r modeling interest rates using a stochastic differential equation.

3. There are no transaction costs.

4. As we briefly mentioned before, the underlying asset does not pay dividends during the life of the option contract. This assumption can be relaxed if the dividend is known beforehand.

5. There are no arbitrage possibilities, which means that the option price is set to match the value of its replicating portfolio, a strategy that prevents any trader from making a risk-free profit by simultaneously buying one and selling the other (Wilmott et al. 1995).

6. The underlying asset can be traded continuously.

7. Short-selling and asset division are permitted. That means that any amount of the underlying asset, integer or not, can be brought or sold, and that non-owned assets can be sold.

3.1.2 Black-Scholes partial differential equation

To describe the Black-Scholes equation, we first need to understand some concepts, including the Wiener process and Itô's lemma.
In real life, asset prices are quoted at discrete intervals of time, which implies the presence of a practical lower bound for the basic time step dt of the random walk. The goal is to work towards a continuous-time random walk, which will provide a more sophisticated model for the time-varying price of assets. We can achieve this by increasing the number of time steps using Itô's lemma. A **Wiener process** W_t is a continuous-time stochastic process with real value, characterized by the following properties:

1. $W_0 = 0$

2. W has independent (Markov's property) and Gaussian increments:

   for every $t > 0$, the future increments $W_{t+u} - W_t$, $u \geq 0$, are independent on the past values W_s, $s < t$;

   $W_{t+u} - W_t$ is normally distributed with mean 0 and variance u.

3. W_t is continuous in t (Malliaris 1990).

Itô's lemma is a foundational concept in stochastic calculus, acting as the equivalent of the chain rule for functions of a stochastic process. It provides a way to calculate the differential of a time-dependent function.
The lemma can be understood by starting with a Taylor series expansion of the function. By expanding up to the second derivatives and keeping terms of

the first order in the time increment and second order in the Wiener process increment, one can derive a heuristic version of the lemma (Hassler 2016).

Consider an option whose value $V(S,t)$ depends only on S, the price of the underlying asset, and t, the time. Consider the return of the asset price, computed by the change in price, divided by the price:

$$dS = S(t + dt) - S(t) \tag{3.1}$$

Return can be expressed as a geometric Brownian motion with drift μ:

$$\frac{dS}{S} = \mu S dt + \sigma S dX \tag{3.2}$$

Where $\mu S dt$ is the deterministic part, depending on μ, the drift, and $\sigma S dX$ represents the random part of the process, characterized by the Wiener process $dX \sim N(0, dt)$.

Suppose that $dX^2 \to dt$ as $dt \to 0$, and that $f(S)$ is a smooth function of S. If we vary S by a small amount dS, then also f varies by a small amount, so from the Taylor series expansion we can write:

$$df = \frac{df}{dS}dS + \frac{1}{2}\frac{d^2f}{dS^2}S^2 + ... \tag{3.3}$$

Recall that dS is given by (3.2), squaring it we find:

$$dS^2 = (\sigma S dX + \mu S dt)^2 = \sigma^2 S^2 dX^2 + 2\sigma\mu S^2 dt dX + \mu^2 S^2 dt^2 \tag{3.4}$$

We now examine the order of magnitude of each term of the equation, since $dX = O(\sqrt{dt})$, the first term is the largest for small dt and dominates the other two, so we can write:

$$dS^2 = \sigma^2 S^2 dX^2 + ... \tag{3.5}$$

Since $dX^2 \to dt$, then:

$$dS^2 \to \sigma^2 S^2 dt \tag{3.6}$$

Substituting the latter result and the definition of dS given by (3.2) into (3.3), and retaining only those terms that are at least large as $O(dt)$, we get the **Itô's lemma**:

$$df = \frac{df}{dS}(\sigma S dX + \mu S dt) + \frac{1}{2}\sigma^2 S^2 \frac{d^2 f}{dS^2} dt = \sigma S \frac{df}{dS} dX + \left(\mu S \frac{df}{dS} + \frac{1}{2}\sigma^2 S^2 \frac{d^2 f}{dS^2}\right) dt. \tag{3.7}$$

This result can be further generalized by considering a function of the random variable S and of time $f(S,t)$, getting:

$$df = \sigma S \frac{\partial f}{\partial S} dX + \left(\mu S \frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} + \frac{\partial f}{\partial t}\right) dt. \tag{3.8}$$

Now that we derived Itô's lemma, we are in the position to derive the Black-Scholes equation.

Consider a general option whose value $V(S,t)$ depends only on the asset price S and the time t. Applying Itô's formula, we get the random walk followed by V:

$$dV = \sigma S \frac{\partial V}{\partial S} dX + \left(\mu S \frac{\partial V}{\partial S} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + \frac{\partial V}{\partial t}\right) dt. \tag{3.9}$$

We now construct a portfolio consisting of an option and an unspecified number Δ of the underlying asset, whose value is:

$$\Pi = V - \Delta S. \tag{3.10}$$

The jump in the value of this portfolio in one time step is the following:

$$d\Pi = dV - \Delta dS \tag{3.11}$$

Putting (3.11) together with (3.2), and (3.9) we find that Π follows a random walk:

$$d\Pi = \sigma S\left(\frac{\partial V}{\partial S} - \Delta\right)dX + \left(\mu S\frac{\partial V}{\partial S} + \frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2} + \frac{\partial V}{\partial t} - \mu\Delta S\right)dt. \tag{3.12}$$

We can eliminate the random component by choosing $\Delta = \frac{\partial V}{\partial S}$, with Δ being the value of $\frac{\partial V}{\partial S}$ at the start of the time step dt. This results in a portfolio whose increment is wholly deterministic:

$$d\Pi = \left(\frac{\partial V}{\partial S} + \frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2}\right)dt. \tag{3.13}$$

Now we recall the assumptions of the Black-Scholes model that there are no arbitrage opportunities and no transaction costs.

The return on an amount Π invested in a risk-free asset over time dt is $r\Pi dt$. In the absence of arbitrage, the return from a self-financing portfolio must be equal to this risk-free return.

If the portfolio return exceeded $r\Pi dt$, an arbitrager could exploit this by borrowing at the risk-free rate and investing in the portfolio, thus earning a guaranteed profit. In contrast, if the portfolio return was less than $r\Pi dt$, an arbitrager would short the portfolio and invest the proceeds at the risk-free rate, again securing a risk-free profit (Wilmott et al. 1995).

This mechanism, enabled by the ability to trade with minimal costs, ensures that the returns of the portfolio and the risk-free asset remain in equilibrium; thus, we have the following:

$$r\Pi dt = \left(\frac{\partial V}{\partial S} + \frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2}\right)dt. \tag{3.14}$$

Substituting (3.10) and (3.11) into (3.14), and dividing throughout by dt we can derive the **Black-Scholes partial differential equation**:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0. \tag{3.15}$$

This equation is a second-order linear partial differential equation (PDE) and without boundary conditions we will not be able to solve it to find the value V (Shinde and Takale 2012).
Consider the boundary conditions:

1. $V(0, t) = 0$, for all t
2. $V(S, t) \sim S - Ke^{-r(T-t)}$ as $S \to \infty$
3. $V(S, T) = max(S - K, 0)$

   Where T is the expiration date of the option. This last condition gives the value of the option at the time that the option matures.

The solution of the PDE returns the value of the option at a previous time, $E[max(S-K, 0)]$. The equation can be solved by recognizing that it is a Chauchy-Euler equation which can be transformed into a diffusion equation, by inducing the change of variable transformation (Wilmott et al. 1995).
With:
$\tau = T - t$

$u = Ce^{r\tau}$

$x = \ln(\frac{S}{K}) + (r - \frac{1}{2}\sigma^2)\tau$

Then the Black-Scholes PDE becomes the diffusion equation:

$$\frac{\partial u}{\partial \tau} = \frac{1}{2}\sigma^2 \frac{\partial^2 u}{\partial x^2}. \tag{3.16}$$

The terminal condition $V(S,T) = max(S - K, 0)$ now becomes an initial condition:

$$u(x, 0) = u_0(x) := K(e^{max(x,0)} - 1) = K(e^x - 1)H(x), \tag{3.17}$$

where $H(x)$ is the Heaviside step function. The **Heaviside step function** corresponds to the enforcement of the boundary data in the S, t coordinate system that is required when $t = T$,

$$V(S,T) = 0, \quad \forall S < K; \quad S, K > 0 \tag{3.18}$$

Using the standard convolution method to solve a diffusion equation given an initial value function, $u(x, 0)$, we have the following.

$$u(x,\tau) = \frac{1}{\sigma\sqrt{2\pi\tau}} \int_{-\infty}^{\infty} u_0(y) \exp\left[-\frac{(x-y)^2}{2\sigma^2\tau}\right] dy \tag{3.19}$$

After some manipulation:

$$u(x,\tau) = Ke^{x+\frac{1}{2}\sigma^2\tau} N(d_1) - KN(d_2), \tag{3.20}$$

where $N(\cdot)$ is the standard normal cumulative distribution function and d_1,d_2 are coefficients such that:

$$d_1 := \frac{\ln \frac{S}{E} + (r + \frac{\sigma^2}{2})(T - t)}{\sigma \sqrt{T - t}} \tag{3.21}$$

$$d_2 := d_1 - \sigma \sqrt{T - t} = \frac{\ln \frac{S}{E} - (r + \frac{\sigma^2}{2})(T - t)}{\sigma \sqrt{T - t}}, \tag{3.22}$$

from which we can deduce the semi-explicit (because of the integral) formulas for the value of either a call or a put option:

$$C(s, t) = S \cdot N(d_1) - Ee^{-r(T-t)} \cdot N(d_2), \tag{3.23}$$

$$P(s, t) = Ee^{-r(T-t)} \cdot N(-d_2) - S \cdot N(-d_1). \tag{3.24}$$

3.1.3 Limitations of Black-Scholes model

The evaluation of options with the Black-Scholes model requires strong assumptions that can often be difficult to encounter in the real world.

The simplifying characteristics required to use the differential equation (3.15), such as a continuous process or normal distribution, are difficult to find in real market data (Janková 2018).

Another crucial requirement for the equation to work is the nonexistence of arbitrage opportunities, which in practice is violated and often results in anomalies (Ambrož 2002).

In addition, the perfect derivative replication by the share and a risk-free instrument cannot be achieved without transaction costs. Volatility correlates not only with the price of the share, but also with the volume of trade and transaction costs (Gong et al. 2010). Moreover, the model does not account for the payment of dividends for the underlying asset, while the majority of stocks pay dividends.

For these reasons, many models have been designed in an attempt to improve the original model by eliminating its drawbacks and extending some of the conditions, in order to better reflect real market conditions (Janková 2018).

3.2 The binomial model

The binomial option pricing model (BOPM) can be considered the simplest approach to pricing options; in contrast to the Black-Scholes model and other option pricing models, which require solutions to stochastic differential equations, BOPM is mathematically simple (Fadugba et al. 2014).

The assumption underlying this model is that markets are perfectly efficient, allowing to model the movements of the underlying asset price over *discrete time periods*. Usually, these intervals are equal in length and the number of periods chosen can influence the level of accuracy and depend on the characteristics of the underlying asset (CQF Staff 2025).

3.2.1 Simple stock pricing

The idea is that an asset current price, S_0, can increase to S_1^u or decrease to S_1^d at maturity. The movement of the asset price from S_0 to S_1^u or S_1^d can be seen as the result of a Bernoulli trial.

We denote by q the probability of an increase in the asset price, in this case the output will be:

$$R^u = \frac{S_1^u}{S_0} > 1 \tag{3.25}$$

Conversely, we denote by $(1-q)$ the probability of a decrease in the asset price, getting:

$$R^d = \frac{S_1^d}{S_0} < 1. \tag{3.26}$$

The difference between R^u and R^d is the range of possible future price outcomes (Van der Hoek and Elliott 2006).

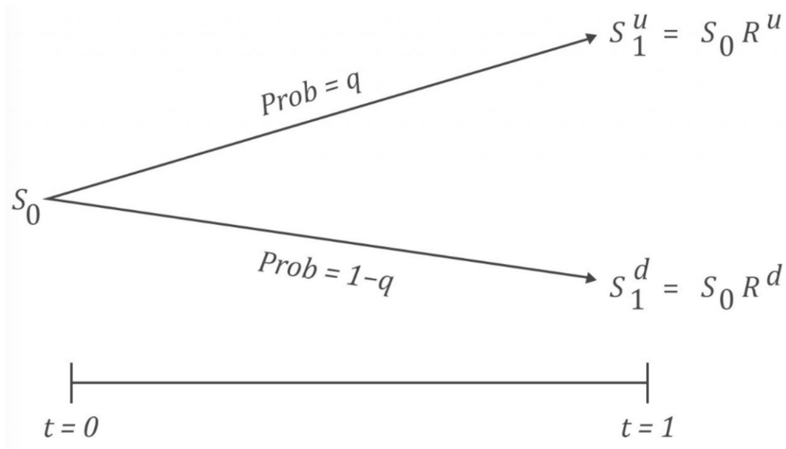

Figure 3.1: One-period Binomial Model - Stock, (AnalystPrep Staff 2025)

3.2.2 Pricing European Options

Similarly, BOPM can be used to price complex derivative instruments, such as European option contracts.

Consider a one-year call option with an underlying price of S_0 and an exercise price of X. Assume that $S_1^d < X < S_1^u$ and that the one-period binomial model is equivalent to the time of expiration of one year (Van der Hoek and Elliott 2006). The value of the option at time $t = 0$ is c_0 (unknown, has to be computed). At time $t = 1$, the option expires and the value of the option will be $c_1^u = max(0, S_1^u - X)$ if the underlying asset price increases to S_1^u or $c_1^d = max(0, S_1^d - X)$ if the underlying asset price falls to S_1^d (Benninga and Wiener 1997).

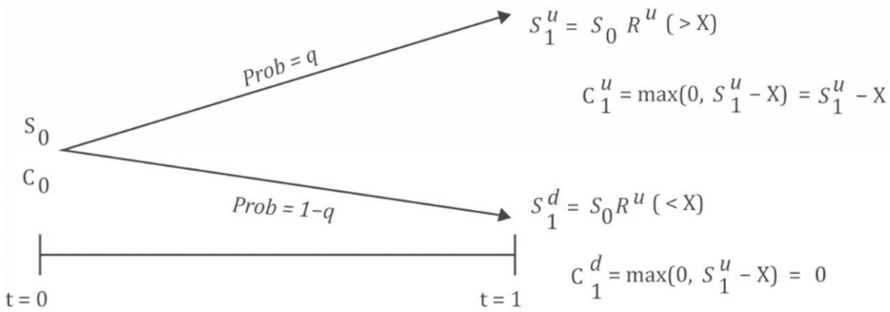

Figure 3.2: One-period Binomial Model - Option, (AnalystPrep Staff 2025)

Then, the **value of** c_0 can be computed by applying replication and no-arbitrage pricing. Assume that at time $t = 0$, we sell a call option for a price of c_0 and buy h units of the underlying asset. Let V be the value of the portfolio such that:

$$V_0 = hS_0 - c_0. \tag{3.27}$$

The value of the portfolio at time $t = 1$ is V_1^u or V_1^d, depending on whether the underlying asset price moves up or down:

$$V_1^u = hS_1^u - c_1^u = h \cdot R^u \cdot S_0 - max(0, S_1^u - X) \tag{3.28}$$

$$V_1^d = hS_1^d - c_1^d = h \cdot R^d \cdot S_0 - max(0, S_1^d - X) \tag{3.29}$$

Under the assumption that there is no arbitrage in the market, we have established two portfolios with identical payoff profiles at time $t = 1$. That means we need to find the value of h such that: $V_1^u = V_1^d$ (Benninga and Wiener 1997).

Therefore:

$$hS_1^u - c_1^u = hS_1^d - c_1^d \tag{3.30}$$

$$\Rightarrow h = \frac{c_1^u - c_1^d}{S_1^u - S_1^d}. \tag{3.31}$$

The value of h is defined as the **hedge ratio**, which is a proportion of the underlying that will offset the risk associated with an option (Benninga and Wiener 1997).

Since $V_1^u = V_1^d$, we can use either of the portfolios to value the option, which has an important implication for the return: $\frac{V_1^u}{V_0} = \frac{V_1^d}{V_0} = 1 + r$ (Benninga and Wiener 1997).

To respect the hypothesis of no-arbitrage, the portfolio value at $t = 1$, ($V_1 = V_1^u = V_1^d$) must be discounted at the risk-free rate, so that:

$$V_0 = V_1(1 + r)^{-1} \tag{3.32}$$

Recalling that $V_0 = hS_0 - c_0$:

$$hS_0 - c_0 = V_1(1 + r)^{-1} \tag{3.33}$$

$$\Rightarrow c_0 = hS_0 - V_1(1 + r)^{-1}. \tag{3.34}$$

The same process can be used to derive the current price of a put option:

$$p_0 = V_0 - hS_0 \tag{3.35}$$

Multiperiod BOP Model on an European Option

Equations (3.34) and (3.35) form the fundamental method to construct the binomial tree. Using the same principle, the binomial approximation for a European

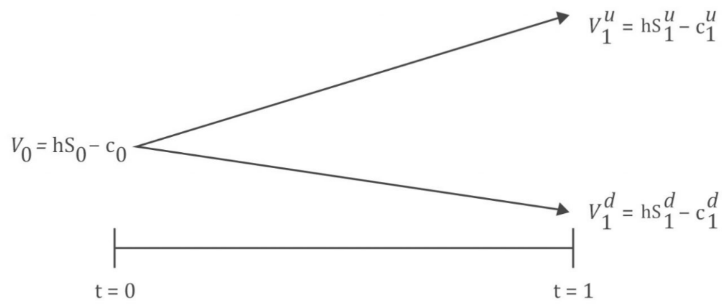

Figure 3.3: Call option pricing, (AnalystPrep Staff 2025)

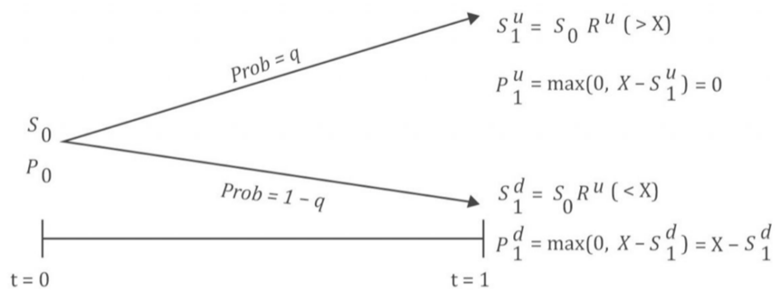

Figure 3.4: Put option pricing, (AnalystPrep Staff 2025)

call or put option can be evaluated for any predetermined number of periods (Hon 2013).

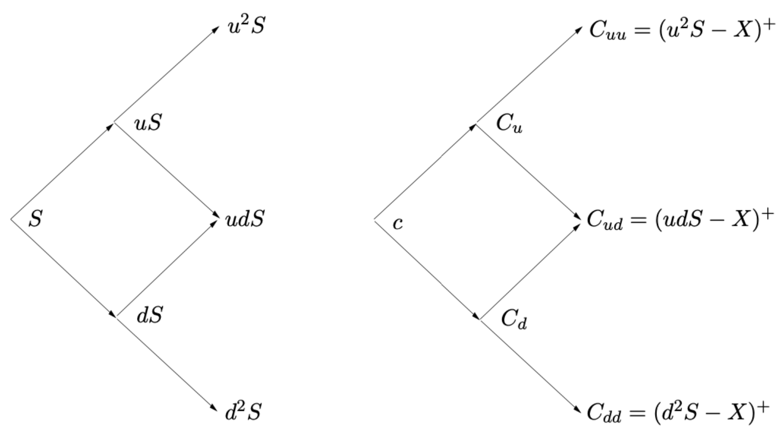

Figure 3.5: Two-period Binomial Model - Call Option, (Hon 2013)

We are considering a European call option. Under the two-period binomial tree, there are three possible prices in the second period: u^2S,udS,d^2S, let the respective option values at the three prices be C_{uu},C_{ud}, and C_{dd}. Thus,

$$C_{uu} = (u^2S - X)^+, \quad C_{ud} = (udS - X)^+, \quad C_{dd} = (d^2S - X)^+ \tag{3.36}$$

Given that the two potential stock prices at any future point depend solely on the current price and not on any prior prices, we can determine the option's value at each step using a one-period binomial model. This allows the valuation process to be broken down into a series of independent one-step calculations (Odegbile 2005).

So, we obtain the option values for the first period by applying the same logic as in the BOP model for one period.

$$C_u = \frac{pC_{uu} + (1-p)C_{ud}}{e^{r\delta t}}, \quad C_u = \frac{pC_{ud} + (1-p)C_{dd}}{e^{r\delta t}} \tag{3.37}$$

where $p = \frac{(e^{r\delta t}-d)}{(u-d)}$. Similarly, a portfolio can be set up for the call option that costs $C_u(C_d)$ if the stock price goes to uS (dS, respectively). Thus,

$$\begin{aligned} c = \frac{pC_u + (1-p)C_d}{e^{r\delta t}} &= \frac{p^2C_{uu} + 2p(1-p)C_{ud} + (1-p)^2C_{ud}}{e^{2r\delta t}} \\ &= \frac{p^2(u^2S - X)^+ + 2p(1-p)(udS - X)^+ + (1-p)^2(d^2S - X)^+}{e^{2r\delta t}} \end{aligned} \tag{3.38}$$

The hedging ratios Δ and B at each node can be derived using the same argument. The hedging ratio Δ tells how many units of the underlying asset to hold per option sold to replicate the option payoff. In other words, it measures how sensitive the option value is to changes in the price of the underlying asset. The hedging ratio B is the amount invested in the risk-free asset in the replicating portfolio. Once Δ is known, B is calculated to ensure that the value of the portfolio is matched to the payoff of the option at each node.

Δ changes from one node to another. For example, Δ at C_u and C_d are, respectively:

$$\frac{C_{uu} - C_{ud}}{S_{uu} - S_{ud}} \quad \text{and} \quad \frac{C_{ud} - C_{dd}}{S_{ud} - S_{dd}} \tag{3.39}$$

Now, we consider the n-period case. By carrying out the calculation at every node while moving backward in time, we have:

$$c = \frac{1}{e^{nr\delta t}} \sum_{j=0}^{n} \binom{n}{j} p^j (1-p)^{n-j} (u^j d^{n-j} S - X)^+. \tag{3.40}$$

Similarly, the price of a European put option is as follows:

$$p = \frac{1}{e^{nr\delta t}} \sum_{j=0}^{n} \binom{n}{j} p^j (1-p)^{n-j} (X - u^j d^{n-j} S)^+ \tag{3.41}$$

and, from the Cox-Ross-Rubistein formula:

$$c = S\phi(a; n, p, p^* u e^{-r\delta t}) - X e^{-nr\delta t} \phi(a; n, p*) \tag{3.42}$$

$$p = X e^{-nr\delta t} \phi(n - a + 1; n, 1 - p^*) - S\phi(n - a + 1; n, 1 - p^* u e^{-r\delta t}) \tag{3.43}$$

Where ϕ refers to the cumulative probability of getting at least a successes (i.e., upward price movements) in n binomial trials, with probability p of success.,

$p^* \equiv \frac{e^{r\delta t}-d}{u-d}$, a is the minimum number of price moves to the top of the market for the option to pay off, and r is the risk-free interest rate.

Convergence of the BOP Model to the Black-Scholes Model

The BOP model converges to the Black-Scholes model as the number of time steps approaches infinity. This essentially means that as the time between each step becomes infinitesimally small, the two models produce the same result (Cox et al. 1979).

The binomial model's results are only accurate when its up-and-down factors are calculated using specific formulas. These formulas ensure that as the number of time steps increases, the distribution of stock returns converges to the same lognormal distribution assumed by models such as Black-Scholes (Odegbile 2005).

A more general approach does not impose restrictions on the choice of up and down parameters (Hsia 1983). We start with our ultimate goal, the Black-Scholes formula for a call option. Suppose that S is the current stock price, X is the strike price, and r is the continuously compounded risk-free interest rate. Suppose further that t is the time to expiration and σ is the volatility. Then, the Black-Scholes model of a call option on a non-dividend-paying stock is:

$$
\begin{aligned}
c &= SN(d_1) - Xe^{-rt}N(d_2) \\
d_1 &= \frac{\ln\left(\frac{S}{X}\right) + \left(r + \frac{1}{2}\sigma^2\right)t}{\sigma\sqrt{t}} \\
d_2 &= \frac{\ln\left(\frac{S}{X}\right) + \left(r - \frac{1}{2}\sigma^2\right)t}{\sigma\sqrt{t}} = d_1 - \sigma\sqrt{t}
\end{aligned}
\tag{3.44}
$$

(Recall equations (3.23), (3.21) and (3.22)).

As $n \to \infty$, $\phi(a; n, pue^{-r\delta t}) \to N(d_1)$ and $\phi(a; n, p) \to N(d_2)$ (Odegbile 2005).

3.2.3 Pricing American Options

BOPM can be used to price American options using a random binary tree (Xiaoping and Jie 2014).

Randomized Binomial Tree

In the traditional binary tree model, the price of the stock moves upward u with a certain probability or downward d with a certain probability at each time node. (u, d) is named as the environment of the binary tree and is the same at any time.

In the random binary tree model, U_n and D_n are independent and identically distributed random variables. U_n can only take two values u_1, u_2 and D_n can only take two values d_1, d_2. Random market environment describes two market environments (u_1, d_1) and (u_2, d_2), where the probability of the market environment (u_1, d_1) is α $(0 \leq \alpha \leq 1)$ and the other is $1 - \alpha$.

To arrive at the computation of the option value, we need some results:

1. If the following condition is satisfied, the random binary tree market must be a market with no arbitrage:

$$max(d_1, d_2) < R < min(u_1, u_2) \tag{3.45}$$

   where R is the risk-free discount factor and the probability of the market environment α is given.

2. When the random binary tree is in the N^{th} period, the total number of nodes is:

$$\frac{1}{6}N^3 + N^2 + \frac{11}{6}N + 1 \tag{3.46}$$

which can also be written as $\frac{(n+1)(n+2)(n+3)}{6}$ or also as $\frac{n+3}{3}$.

3. Random binary tree nodes can be described by the four-tuple as follows: (N, n, i, j) where N is the period, n is the number of occurrences of the market environment (u_1, d_1), the occurrence number of market environment (u_2, d_2) is $N - n$, i represents that the frequency of u_1 is $n - i$ under the market environment (u_1, d_1), which of d_1 is i, and j represents that the frequency of u_2 is $N - n - j$ under the market environment (u_2, d_2), which of d_2 is j.

   There is a one-to-one relationship between the four-tuple (N, n, i, j) and the stock price formula $Su_1^{n-i}d_1^i u_2^{N-n-j}d_2^j$, so we can use the four-tuple instead of the stock price formula.

4. In the random binary tree, the unconditional probability of four child nodes $(N+1, n+1, i, j)$, $(N+1, n+1, i+1, j)$, $(N+1, n, i, j)$, and $(N+1, n, i, j+1)$ of any node (N, n, i, j) are, respectively, αp_1, $\alpha(1-p_1)$, $(1-\alpha)p_2$ and $(1-\alpha)(p_2)$.

Pricing American options based on Randomized Binomial Tree

Suppose that the risk-free rate is r, the risk-free discount factor is $R = e^{r\Delta T}$, T is the maturity of the option, $\Delta T = \frac{T}{N}$ is the time of each period of the random binary tree, and N are the parameters of the random binary tree.

According to the principle of risk-neutral pricing, we get:

$$p_1 = \frac{R - d_1}{u_1 - d_1}, \quad p_2 = \frac{R - d_2}{u_2 - d_2} \tag{3.47}$$

The maturity of the set American option is T, the payoff function is $f((N,n,i,j),K)$, and K is the strike price. We can calculate the price of the American option with the randomized binomial tree using the backward induction method:

1. *Step 1* On the expiration date T (the N^{th} period of the random binary tree), calculating the immediate strike price $f((N,n,i,j),K)$ at each node, this time the option price is $V((N,n,i,j)) = f((N,n,i,j),K)$.

2. *Step 2* When the period $m = 1,2,...,N-1$, for each node (m,n,i,j) in this period, calculate the immediate strike price $f((N,n,i,j),K)$ of the option of this node and then calculate the value of the option to continue to hold:

$$
\begin{aligned}
c((m,n,i,j)) = \frac{1}{R}\{ap_1 V((m+1,n,i,j \\
+ \alpha(1-p_1)V((m+1,n,i+1,j)) \\
+ (1-\alpha)p_2 V((m+1,n+1,i,j))+ \\
(1-\alpha)(1-p_2)V((m+1,n+1,i,j+1))\}
\end{aligned}
\tag{3.48}
$$

The value of the American option at node (m,n,i,j) is:

$$
V((m,n,i,j)) = max(f(m,n,i,j),K), c((m,n,i,j)) \tag{3.49}
$$

$$
\begin{aligned}
V = V((0,0,0,0)) = \frac{1}{R}\Big\{&\alpha p_1 V((1,0,0,0)) + \alpha(1-p_1)V((1,0,1,0)) \\
&+(1-\alpha)p_2 V((1,1,0,0)) \\
&+(1-\alpha)(1-p_2)V((1,1,0,1))\Big\}.
\end{aligned}
\tag{3.50}
$$

3.2.4 Limitations of the BOPM

Due to the BOPM simplifying assumptions, this model is quite hard to adapt to complex situations. Sometimes, the model does not account for the value of managerial flexibility inherent in many types of projects.
Although the binomial model can be modified to estimate the up and down movements from a stock's variance, it still cannot precisely predict the actual stock prices at the end of each period. The model can only determine the possible price paths and their probabilities (Odegbile 2005).
Another major limitation of the binomial option pricing model is its slow speed (Emmanuel et al. 2014).

3.3 Monte Carlo model

The Monte Carlo option model (MC) is based on Monte Carlo methods to evaluate an option contract under complicated and complex circumstances of uncertainty.
As other traditional methods, Monte Carlo valuation requires risk-neutral valuation, the price of the option being its discounted expected value (Boyle 1977). Essentially, the applied technique is as follows.

1. Generate a large number of possible, but random, paths for the underlying asset via simulation.

2. Calculate the associated exercise value (payoff) of the option for each path.

3. Get the value of the option by averaging and discounting the payoffs to today.

3.3.1 Valuation of European Call Option by MC

An option's value is the risk-neutral expected value of its future pay-off, discounted back to the present. We can estimate this by averaging a large number of these discounted payoffs (Bolia and Juneja 2005). Consider a European call option in the Black-Scholes world that pays:

$$\phi(S_T) = max(0, S_T - K) \quad \text{at maturity date T} \tag{3.51}$$

We know from the Feynman-Kac theorem that the solution to certain partial differential equations (PDEs), can be expressed as the expected value of a stochastic process under the risk-neutral measure. Specifically:

$$F(t, x) = E_{t,x}[e^{-r(T-t)}\phi(S_T)], \tag{3.52}$$

in other words, the option price at time 0 is

$$C_0 = F(0, S_0) = E_Q(e^{-rT}\phi(S_T)) \tag{3.53}$$

To implement a Monte Carlo simulation, it is first necessary to simulate the Geometric Brownian Motion (GBM) process of the underlying asset:

$$dS_t = (r - \delta)S_t dt + \sigma S_t dW_t. \tag{3.54}$$

Let $X_t = ln(S_t)$. We have the following:

$$dX_t = vdt + \sigma dW_t \quad \text{, where} \quad v = r - \sigma - \frac{1}{2}\sigma^2 \tag{3.55}$$

That can be discredited as:

$$X_{t+\Delta t} = X_t + v\Delta t + \sigma(W_{t+\Delta t} - W_t) \quad \text{, where} \quad \Delta t \quad \text{is a time increment} \tag{3.56}$$

In terms of the asset price S_t, we have:

$$\begin{aligned} S_{t+\Delta t} &= e^{X(t+\Delta t)} \\ &= e^{X_t + v\Delta t + \sigma(W_{t+\Delta t} - W_t)} \\ &= e^{X_t} e^{v\Delta t + \sigma(W_{t+\Delta t} - W_t)} \\ &= S_t e^{v\Delta t + \sigma(W_{t+\Delta t} - W_t)} \end{aligned} \tag{3.57}$$

The random increment $W_{t+\Delta t} - W_t \sim N(0, \Delta t)$ can be simulated by random samples of $\sqrt{\Delta t}\epsilon$, where $\epsilon \sim N(0, 1)$. Equation (3.57) provides a way of simulating values of S_t.

The Monte Carlo Euler scheme uses the Euler(or Euler-Maruyama) approximation on a time grid. In general,the time grid is $0 = t_0 < t_1 < \cdots < t_m$ and for a diffusion of type:

$$dX(t) = a(X(t))dt + b(X(t))dW(t). \tag{3.58}$$

The approximation $\hat{X}$ is given by:

$$\hat{X}(t_{i+1}) = \hat{X}(t_i) + a(\hat{X}(t_{i+1}))[t_{i+1} - t_i] + b(\hat{X}(t_{i+1}))\sqrt{t_{i+1} - t_i \epsilon_{i+1}} \tag{3.59}$$

with $\epsilon_1, \epsilon_2, \ldots$ independent standard normal random variables.

With a fixed time step, the implementation method is straightforward: we divide the time period $[0, T]$ over which we wish to simulate S_t into subintervals N of equal width $\Delta t = \frac{T}{N}$. We can then generate values of S_t at the end of these intervals, $t_i = i\Delta t$, $i = 1, \ldots, N$ using equation (3.56) as follows:

$$S_{t_i} = e^{X_{t_{i-1}}}, \tag{3.60}$$

where:

$$X_{t_i} = X_{t_{i-1}} + v\Delta t + \sigma \sqrt{\Delta t}\epsilon_i. \tag{3.61}$$

For each simulated path, we compute the payoff of the call option (3.51). To estimate the value of the call option, we simply take the discounted average of the simulated payoffs.

$$\hat{C}_0 = e^{-rT} \frac{1}{M} \sum_{j=1}^{M} max(0, S_T - K) \tag{3.62}$$

Note: C_0 is the true value of the option, and $\hat{C}_0$ is an estimate of that value. Since $\hat{C}_0$ is derived from a random sample average, it is itself a random variable with an error. The standard error is a measure of this estimation error:

$$SE(\hat{C}_0) = \frac{SD(\hat{C}_0)}{\sqrt{M}}, \quad SD(\hat{C}_0) = \sqrt{\frac{1}{M-1} \sum_{j=1}^{M} (C_{0,j} - \hat{C_0}^2)}. \tag{3.63}$$

3.3.2 Valuation of American Call Options by MC

Recent papers introduced a Monte Carlo simulation-based method for pricing American options with a finite number of exercise dates. These approaches involve creating two estimators from a simulated tree that are consistent and unbiased in the limit: one that is consistently too high and the other that is consistently too low. This allows us to calculate a conservative confidence interval for the true price of the option (Broadie and Glasserman 1995).

- Let there be d exercise opportunities at times $0 = t_0 < t_1 < \cdots < t_{d-1} = T$, T being the expiration time of the option initiated at time t_0 or today.
- Consider a Markov chain with vector values $\{S_t : t = 0, 1, \ldots, T\}$ consisting of all the information required to determine the return of exercising an option.
- e^{-Rt} is the discount factor from $t-1$ to t. We take R_t to be a component of the vector S_t and assume $R_t \geq 0$ for all t. Also, let $R_{0t} = \sum_{i=1}^{t} R_i$.
- $h_t(s)$ is the reward for exercise, $g_t(s) = E[e^{-R_{t+1}} f_{t+1}(S_{t+1}) | S_t = s]$ is the continuation value and $f_t(s) = max\{h_t(s), g_t(s)\}$ is the option value at time t in state s, $t = 0, \ldots, T-1$. With $f_T(s) = g_T(s) = h_T(s)$.

It is known that the price of an American option is the solution to an optimal stopping problem (Gautam 1997):

$$f_0(S_0) = \max_{\tau} E[e^{-R_{0\tau}} h_\tau(S_\tau)] \tag{3.64}$$

The optimal policy stops at:

$$\tau^* = inf\{t = 0, \ldots, T : h_t(S_t) \geq g_t(S_t)\} \tag{3.65}$$

i.e., the first time the immediate exercise value is at least as great as the continuation value.

There is no unbiased estimator of (3.64), so two estimators can be introduced, one biased high and one biased low, both consistent and asymptotically unbiased (Broadie and Glasserman 1995).

Instead of using sample paths, the Monte Carlo method for pricing American options simulates the evolution of the underlying asset's price S_t using random trees. This is different from the approach for European options, which only requires simulating simple price paths.

Given a value of the branching parameter b, the evolution of the tree can be described recursively as follows.

- *Step 1*: From the (fixed) initial state S_0, we generate b independent samples $S_1^1, \ldots, S_2^b$ of the state at time $t = 1$;
- *Step 2*: From each of these b samples, $S_1^{i_1}$, $i_1 = 1, \ldots, b$, in turn, generate b new samples, $S_2^{i_1 1}, \ldots, S_2^{i_1 b}$ at time $t = 2$; hence there are a total of b^2 nodes at $t = 2$;
- Repeat this procedure for every possible exercise time t

The high estimator is obtained by applying dynamic programming to the random tree. This is done by working backward through the tree, using recursive calculations to find the option's value at each step:

$$\Theta_T^{\alpha_t} = h_T(S_T^{\alpha_t}), \quad \Theta_t^{\alpha_t} = max\{h_t(S_t^{\alpha_t}), \frac{1}{b}\sum_{j=1}^{b} e^{-R_{t+1}^{\alpha_t j}} \Theta_{t+1}^{\alpha_t j}\} \tag{3.66}$$

we compute the high estimator $\Theta = \Theta_0$. At any given node, the **high estimate** is the greater of two values: the immediate exercise value or the discounted average of the high estimates of the next set of nodes. This high estimator uses all available branches to approximate the best course of action (to exercise or continue) and its corresponding value.

In contrast, the **low estimator** works differently. It separates the branches used to decide on an action from the branches used to determine the value of continuing to hold the option. This separation results in a different estimate of the option price (Broadie and Glasserman 1995).

In summary, this method involves creating random trees, where each node has a set number of branches, represented by the parameter b. Once these trees are generated, you work backward from the end to compute the option price at each node. This process effectively incorporates the early exercise feature into the simulation. The final option price is determined using two estimates: one that is biased high and another that is biased low. Both estimates become unbiased as the number of branches, b, approaches infinity (Gautam 1997).

The table below summarizes the results on options evaluation that have been calculated in this section.

	European options	American options
Black-Scholes model	$C(s,t) = S \cdot N(d_1)$ $- Ee^{-r(T-t)} \cdot N(d_2)$ $P(s,t) = Ee^{-r(T-t)} \cdot N(-d_2)$ $- S \cdot N(-d_1)$	-
Binomial model	$C = S\phi(a;n,p,p^{*}ue^{-r\delta t})$ $- Xe^{-nr\delta t}\phi(a;n,p^{*})$ $P = Xe^{-nr\delta t}\phi(n-a+1;$ $n, 1-p^{*}) - S\phi(n-a+1;$ $n, 1-p^{*}ue^{-r\delta t})$	$V = \frac{1}{R}\{ap_1 V((1,0,0,0)) +$ $\alpha(1-p_1)V((1,0,1,0)) +$ $(1-\alpha)p_2 V((1,1,0,0)) +$ $(1-\alpha)(1-p_2)V((1,1,0,1))\}$
Monte Carlo approach	$\hat{C}_0 = e^{-rT}\frac{1}{M}\sum_{j=1}^{M} \max(0, S_T -$ $K)$	$f_0(S_0) = \max\{E[e^{-R_{0\tau}} h_\tau(S_\tau)]\}$

Chapter 4

Machine Learning Methods

Machine learning (ML), a subfield of artificial intelligence (AI), focuses on giving computers the ability to learn and adapt without being explicitly programmed. ML algorithms improve their performance and accuracy by autonomously learning from experience and processing more data, much like humans (Alpaydin 2021).

An usual ML algorithm is composed of three components (Berkeley 2020):

1. A *decision process*: the general steps a ML algorithm takes to make a prediction.

2. An *error function*: the way to quantify the difference between predicted outputs and actual target values.

3. An *optimization process*: the process of adjusting the parameters of the model to minimize the error function (or maximize an objective function).

There are several types of ML algorithms, depending on the type of data needed, the learning methods, and the labels used (Ayodele 2010).
In **Supervised learning**, an algorithm learns a function that maps an input to a desired output. A common example is classification, where the model learns to assign an input vector to one of several predefined classes by studying a set of labeled input-output examples.
Unsupervised learning involves modeling a dataset of inputs without any corresponding labeled outputs. The goal is to discover hidden patterns or structures within the data on its own.
Semi-supervised learning combines a small amount of labeled data with a large amount of unlabeled data during training. This allows the algorithm to improve its learning function or classifier by using a mixture of both types of examples.
Reinforcement learning involves an algorithm that learns an optimal policy for making decisions based on observations of its environment. Each action it takes has an impact, and the environment provides feedback in the form of rewards or penalties, which guides the algorithm's learning process.

4.1 Regression Trees and Random Forests

Classification and regression trees are modern statistical methods that are well suited for exploring and modeling data (Bryzgalova et al. 2019). A Decision Tree for regression is a model that uses a tree-like structure to provide predicted numerical values. Data are split into key features, starting from a root question and branching out. Then each node asks for a characteristic, and the data is further divided until the leaf nodes reach the final predictions (Coqueret and

Guida 2020).

To begin with, the predictor space J is divided into non-overlapping regions, denoted $R_1, R_2, \ldots, R_J$. For any observation x_i falling into a specific region R_j, we estimate its value $f(x)$, taking the average of all training observations $\hat{y}_i$ within that same region.

Next, to create new partitions from an existing one, we select a predictor variable, j, and a splitting value, s. These define two new regions, $R_1(j, s)$ and $R_2(j, s)$, by splitting the data based on whether the predictor value x_j is greater than s.

$$R_1(j, s) = \{x|x_j < s\} \text{and} R_2(j, s) = \{x|x_j \geq s\} \tag{4.1}$$

The optimal pair for the predictor variable (j) and the split point (s) is chosen by minimizing the Residual Sum of Squares (RSS).

$$\sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y_{R_1}})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y_{R_2}})^2 \tag{4.2}$$

This calculation finds the split that results in the most homogeneous new regions, thereby reducing the overall error.

4.1.1 Random Forests

Instead of using a regularization parameter, trees are defined by hyperparameters that determine their structure, such as the number of branches or the minimum number of observations required in each final partition.

A single tree can have high predictive variance, which is addressed by random forests. This method improves predictive performance and stability by averaging predictions from multiple decision trees. Essentially, a random forest

involves building many regression trees and then averaging their results (Sutton 2005). To ensure that these individual trees are not identical, a bootstrap sampling method is used to introduce randomness. We build B decision trees $T_1, \ldots, T_B$ using the training sample.
For each observation in the test set, a final prediction is made by averaging the predictions from all the trees in the forest.

$$\hat{y} = \frac{1}{B} \sum_{i=1}^{B} \hat{y_{T_i}} \tag{4.3}$$

4.2 Neural Networks

In machine learning, Neural Networks (NNs) are computational models inspired by the structure and functions of biological neural networks.

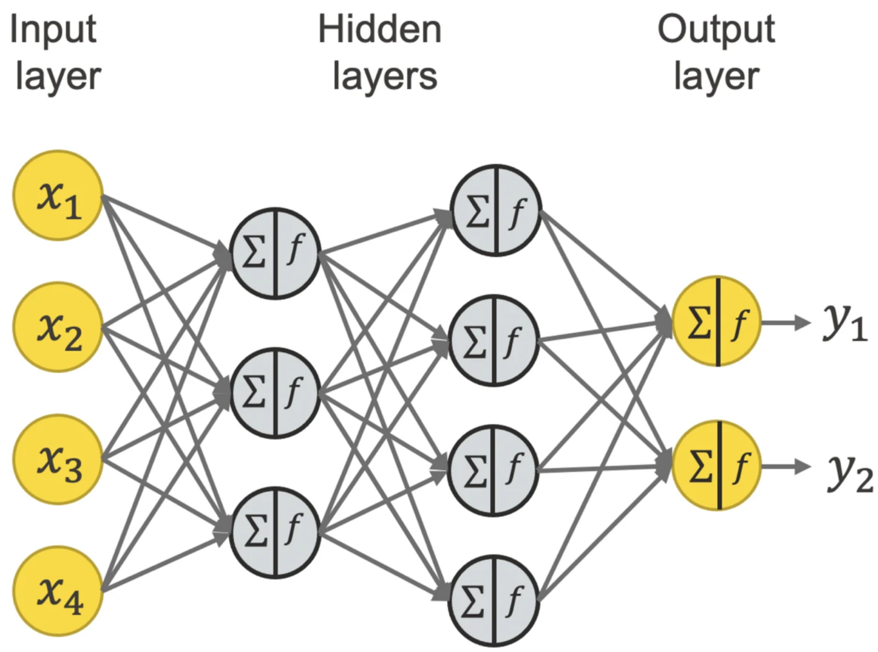

Figure 4.1: Example of a Neural Network, (Melcher 2021)

A NN is composed by units, called *neurons*, mimicking biological neurons. The neurons are connected to each other by *peaks*.
A "signal", which is a real number, is received and processed by the artificial connected neurons. The "importance" of each partial output is determined by the *weights*, which are adjusted during the learning process (Nielsen 2015).
The output of each neuron is computed by the *activation function*, a non-linear function of the totality of its input.

$$x_k^l = f(\theta_0^k + \sum_{j=1}^{N^l} z_j \theta_{l,j}^k) \tag{4.4}$$

Where, θ are the parameters to fit, N^l is the number of units (hyperparameters), and z_j are the input variables.
Selecting the most appropriate activation function is one of the key design decisions in building a neural network. The **sigmoid function** is frequently used in the output layer for binary classification problems. This is because its output, which is always between 0 and 1, can be directly interpreted as the probability that an observation belongs to a specific class.
A **deep neural network** is a neural network that contains multiple hidden layers between the input and output layers. This deep structure allows the network to represent more complex data relationships using fewer parameters compared to a simple, shallow (one-layer) network.

4.3 ML for Option Pricing

Machine learning algorithms can be extremely useful for the field of finance, since they offer a powerful and adaptive way to overcome the limitations of traditional models.

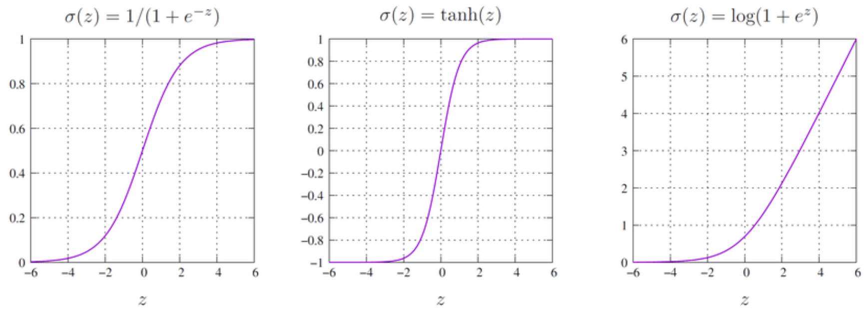

Figure 4.2: Most common activation functions: logistic, hyperbolic tangent and softplus, (Göküzüm et al. 2019)

Traditional option pricing models, while effective in certain contexts and still widely used throughout the world, share several key limitations that make them challenging to apply in today's complex markets. A primary issue is their reliance on restrictive assumptions (Yadav 2025). For example, the Black-Scholes model assumes that volatility is constant and that the market has no transaction costs, which is rarely true in the real world (D'Uggento 2025).

Similarly, the Binomial model, while more flexible, can become computationally intensive for options with long maturities, as the number of required steps grows exponentially (Hsia 1983).

The Monte Carlo method, while highly versatile, is also computationally demanding and can be slow to converge, making it unsuitable for real-time pricing of certain options, particularly American options without additional techniques (Wenda 2022).

Unlike traditional option pricing models, ML-based option pricing models do not require strong assumptions about market behavior. ML algorithms have the ability to directly analyze large datasets, identifying hidden patterns and relationships that are too complex for traditional methods to capture (Reaz et al. 2020). For example, an ML model can be trained to recognize how the historical price, the trading volume, and the sentiment of the news collectively influence its future volatility, leading to more accurate pricing (Gan et al. 2020).

Chapter 5

Methodological Framework of the Empirical Analysis

5.1 Data Collection

The current stock prices for the MiB30 tickers are retrieved from Yahoo Finance, along with the historical time series used to estimate the volatility of each stock. Option data, such as strike prices, maturities, and market quotes, are sourced from Refinitiv. Finally, the Euro short-term rate (€STR), published on the official website of the ECB, is adopted as the risk-free rate.

5.2 Traditional Option Pricing Models

Three traditional option pricing models are implemented as a benchmark for our comparison.

1. Black-Scholes Model

2. Monte Carlo Simulation

3. Binomial Tree Model

5.3 ML Option Pricing Models

Three Machine Learning models that can learn to solve the Black-Scholes equation by analyzing various option specifications and their corresponding prices, all without being explicitly given the underlying formula, are implemented.

1. Random Forest

2. Single Layer Neural Network

3. Deep Neural Network

The input features for the models include:

1. Spot price of the underlying asset

2. Strike price

3. Time to maturity

4. Risk-free interest rate

5. Volatility

The models are trained using historical option prices and market data from selected stock tickers. Hyperparameter tuning is performed using cross-validation, and the model is optimized using Adam optimizer.

5.4 Evaluation metrics

The accuracy of the models is evaluated using the following metrics:

1. Root Mean Squared Error (RMSE): calculates the standard deviation of the prediction errors, effectively indicating how concentrated the data is around the line of best fit. It is particularly sensitive to large errors, since the errors are squared before averaged.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{5.1}$$

2. R-squared (R^2): measures the proportion of variance in the dependent variable that can be predicted from the independent variables. It essentially indicates how well the model's predictions align with the actual data.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \tag{5.2}$$

CHAPTER 6

Implementation in R

6.1 Data

As introduced in the previous chapter, the stocks in the MIB30 index are selected as a representative sample of European equities. The stocks included in this index are summarized in the table below.

A2A.MI	A2A Energia	RACE.MI	Ferrari	PRY.MI	Prysmian
BMED.MI	Banca Mediolanum	FBK.MI	Fineco Bank	REC.MI	Recordati
BAMI.MI	Banco BPM	G.MI	Generali	REY.MI	Reply
BMPS.MI	Banca Monte dei Paschi di Siena	ISP.MI	Intesa San Paolo	SRG.MI	Snam
BPE.MI	BPER Banca	INW.MI	Inwit	STLAM.MI	Stellantis
BC.MI	Brunello Cucinelli	LDO.MI	Leonardo	STMMI.MI	STMicroelectronics
BZU.MI	Buzzi	MB.MI	Mediobanca	TEN.MI	Tenaris
CPR.MI	Campari	MONC.MI	Moncler	TRN.MI	Terna
ENEL.MI	Enel	NEXI.MI	Nexi	UCG.MI	UniCredit
ENI.MI	Eni	PST.MI	Poste Italiane	UNI.MI	Unipol

Table 6.1: List of MiB30 tickers with corresponding firms

The data required to implement an option pricing model consist of the following elements: S, the current stock price; *sigma*, the volatility of the stock; r, the risk-free interest rate; K, the strike price of the option; and T, the time to maturity of the option.

For evaluation, the last market price of the option will be used as a reference.

- S is collected from Yahoo Finance and stored in a dataframe that contains the ticker and the most recent price. To mitigate the risk of missing values, prices from the last five trading days are downloaded and the last available observation is retained.
- Sigma is estimated from historical returns as the standard deviation and is stored in a list.
- The risk-free rate r is set as the Euro short-term rate (€STR) from ECB official site, at the 3^{rd} of September 2025.
- Option data, including K, the historical last price and the expiration date, are stored in a dataframe for each stock.
- T is computed as the difference in years between the expiration date and the issue date (the 3^{3d} of September 2025).

6.2 Traditional models

The Black–Scholes formula is defined as follows (TidyFinance 2023):

```r
black_scholes_price <- function(S, K, r, T, sigma) {
  d1 <- (log(S / K) + (r + sigma^2 / 2) * T) / (sigma * sqrt(T))
  d2 <- d1 - sigma * sqrt(T)
  price <- S * pnorm(d1) - K * exp(-r * T) * pnorm(d2)
  return(price)
}
```

The binomial model is implemented via the `derivmkts` package, with the function `binomopt`. The parameters are set as follows:

```
binomopt(s, k, v=sigma, r, tt=T, d=0, nstep = 500,
         american = FALSE, putopt=FALSE)
```

for call options. Regarding put options, the parameter `putopt` is set on `TRUE`.

For the Monte Carlo simulation, the following function is defined using the `NMOF` package, specifically the function `gbm`:

```
set.seed(123)
npaths <- 20000
timesteps <- 100
mc_price_call <- function(S0, K, r, sigma, T) {
  v <- sigma^2
  paths <- gbm(npaths, timesteps, r, v, tau = T, S0 = S0)
  payoff <- pmax(paths[nrow(paths), ] - K, 0)
  price <- mean(payoff) * exp(-r * T)
  return(price)
}
```

For call options, while for put options, the payoff is computed as `payoff <- pmax(K - paths[nrow(paths), ], 0)`.

These traditional models are applied to estimate the option price of each stock.

	ExpiryDate	Last_C	Strike	Last_P	T_days	T_years	BS_price	Binom_price	MC_price
	<date>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	2025-09-19	6.09	91	1.49	16	0.04383562	7.075156	7.074728	7.083937
2	2025-09-19	5.34	92	1.74	16	0.04383562	6.226648	6.226113	6.175701
3	2025-09-19	4.66	93	2.05	16	0.04383562	5.422947	5.422140	5.412259
4	2025-09-19	3.00	94	2.51	16	0.04383562	4.670726	4.669854	4.696497
5	2025-09-19	3.41	95	2.80	16	0.04383562	3.975819	3.975535	3.937353
6	2025-09-19	2.86	96	3.26	16	0.04383562	3.342807	3.343189	3.357414

Figure 6.1: Estimated prices of six options using traditional models

Performance is evaluated using RMSE and R^2, using the observed last market price as a benchmark:

```
calc_rmse <- function(actual, predicted)
  sqrt(mean((predicted - actual)^2))
calc_r2   <- function(actual, predicted) {
  ss_res <- sum((actual - predicted)^2)
  ss_tot <- sum((actual - mean(actual))^2)
  1 - ss_res / ss_tot
}
```

The following table reports the mean RMSE and R^2 across all options:

Model	RMSE	R^2
Black-Scholes	0.913	0.735
Binomial	0.913	0.735
Monte Carlo Simulation	0.903	0.777

Table 6.2: Performance metrics of traditional models

The results indicate satisfactory performance: RMSE values remain below one, while R^2 is consistently above 0.7. Among these approaches, the Monte Carlo method yields slightly superior accuracy, with a lower RMSE and higher R^2.

These results serve as a benchmark for subsequent comparison with machine learning models.

6.3 ML-based models

The data set is divided into a training set (80% of the options) and a test set used for the final evaluation.

```
split_data <- initial_split(option_data, prop = 0.8)
train_data <- training(split_data)
test_data  <- testing(split_data)
```

A preprocessing recipe is then defined to standardize predictors and introduce derived features:

```
rec <- recipe(Last_Price ~ S + K + r + T + sigma,
data = train_data) %>%
step_mutate(
  moneyness = S / K,
  log_moneyness = log(S / K),
  sqrt_T = sqrt(T)
  ) %>%
  step_zv(all_predictors()) %>%
  step_normalize(all_predictors())
```

A neural network with one hidden layer of 50 neurons is trained via the `brulee` package:

```
nn_fit <- workflow() %>%
  add_recipe(rec) %>%
```

```
  add_model(
  mlp(epochs = 1000, hidden_units = 50, activation = "relu",
  penalty = 1e-4) %>%
  set_mode("regression") %>%
  set_engine("brulee", early_stopping = TRUE, patience = 50)
  ) %>%
  fit(data = train_data)
```

A deeper neural network with three hidden layers (50, 30, and 20 neurons, respectively) is fitted:

```
deep_nn_fit <- workflow() %>%
  add_recipe(rec) %>%
  add_model(
  mlp(epochs = 1000, hidden_units = c(50,30,20),
  activation = "relu", penalty = 1e-4) %>%
  set_mode("regression") %>%
  set_engine("brulee", early_stopping = TRUE, patience = 50)
  ) %>%
  fit(data = train_data)
```

Finally, a Random Forest model with 1000 trees is tuned and trained. The hyperparameters *mtry* (predictors considered in each split) and min_n (minimum samples per leaf) are optimized by grid search and cross-validation.

```
rf_model <- rand_forest(
  trees = 1000,
  min_n = tune(),
  mtry = tune()
```

```
  ) %>%
  set_mode("regression") %>%
  set_engine("ranger", importance = "impurity")

rf_wf <- workflow() %>%
  add_recipe(rec) %>%
  add_model(rf_model)

rf_grid <- grid_regular(
  min_n(range = c(1, 5)),
  mtry(range = c(2, 5)),
  levels = 5
)

rf_tune <- tune_grid(
  rf_wf,
  resamples = vfold_cv(train_data, v = 5),
  grid = rf_grid,
  metrics = metric_set(rmse),
  control = control_grid(save_pred = TRUE)
)

best_rf <- select_best(rf_tune, metric = "rmse")
rf_fit <- finalize_workflow(rf_wf, best_rf) %>%
  fit(data = train_data)
```

The final models are evaluated on the test set, and the average metrics are reported in Table 6.3.

Model	RMSE	R^2
Single-layer NN	0.406	0.954
Deep NN	0.427	0.961
Random Forest	0.310	0.967

Table 6.3: Performance metrics of ML-based models

The results demonstrate a substantial improvement compared to traditional models. Neural networks achieve higher accuracy, with values of R^2 greater than 0.95, while the random forest produces the lowest RMSE and the highest R^2, confirming its superior predictive performance.

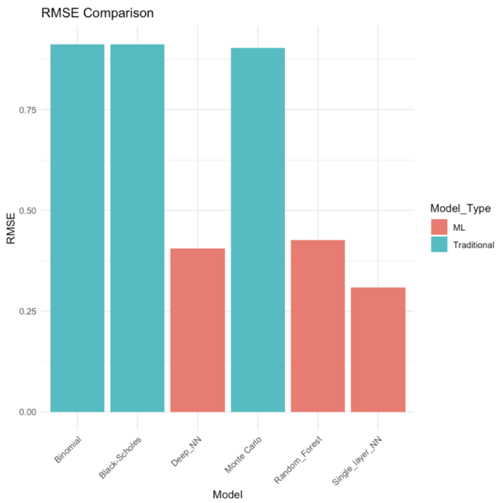

Figure 6.2: RMSE comparison

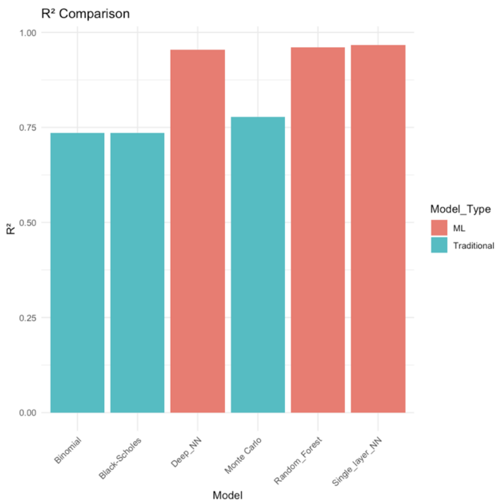

Figure 6.3: R^2 comparison

Chapter 7

Conclusion

The primary goal of this thesis was to compare traditional pricing models with Machine Learning techniques to evaluate whether data-driven models provide more accurate or robust estimations in financial markets characterized by volatility and non-linear relationships.
This research was structured in two parts: a theoretical framework outlining foundational models like Black-Scholes and the binomial model, and an empirical analysis comparing their performance against machine learning models.
Empirical analysis successfully demonstrated that machine learning models significantly outperformed traditional models in valuing options. That can be seen from the performance metrics that were calculated: the random forest achieved the lowest root mean square error (RMSE) of 0.310 and the highest R^2 of 0.967 (see 6.3). However, neural network models also performed well with R^2 values greater than 0.95.
Although traditional models are valuable for providing a theoretical foundation and analytical solutions , they often struggle to capture real-world market complexities such as volatility smile and skew due to their strict assumptions.

By providing a direct, empirical comparison, the thesis validates the use of machine learning as a powerful alternative for financial professionals, especially in situations where traditional models fall short. The findings highlight the potential for integrating advanced data science techniques into financial markets to obtain more precise valuation, risk management, and trading strategies. The results of this study open several avenues for future research. Further work could explore the application of more advanced or novel machine learning and deep learning architectures to see if they can further improve predictive accuracy.

Bibliography

Alpaydin, Ethem (2021). *Machine learning*. MIT press.

Ambrož, L (2002). "Option prices". *Praha*.

AnalystPrep Staff (2025). *The One-Period Binomial Model*. `https://analystprep.com/cfa-level-1-exam/derivatives/one-period-binomial-model-2/`. Accessed on August 13, 2025.

Ayodele, Taiwo Oladipupo (2010). "Types of machine learning algorithms". *New advances in machine learning*.

Benninga, Simon and Zvi Wiener (1997). "The binomial option pricing model". *Mathematica in Education and Research 6*.

Berkeley, UC (2020). "What is Machine Learning?" *UC Berkeley School of Information*.

Bolia, N. and S. Juneja (2005). "Monte Carlo methods for pricing financial options". *Sadhana 20*.

Boyle, Phelim P. (1977). "Options: A monte carlo approach". *Journal of financial economics 4.3*.

Broadie, M. and P. Glasserman (1995). "A Pruned and Bootstrapped American Option Simulator". *Pro- ceedings of the Winter Simulation Conference, Society of Computer Simulation, San Diego*.

Bryzgalova, Svetlana, Markus Pelger, and Jason Zhu (2019). "Forest through the trees: Building cross-sections of stock return". *SSRN 3493458*.

CFI Staff (2025). *What Is a Forward Contract?* `https://corporatefinanceinstitute.com/resources/derivatives/forward-contract/`. Accessed on August 13, 2025.

Coqueret, Guillaume and Tony Guida (2020). *Machine Learning for Factor Investing*. Chapman & Hall.

CQF Staff (2025). *What is the Binomial Model?* `https://www.cqf.com/blog/quant-finance-101/what-is-the-binomial-model`. Accessed on August 13, 2025.

D'Uggento, Angela Maria (2025). "Predicting option prices: From the Black-Scholes model to machine learning methods". *Big Data Research*.

Debellefroid, Maxime (2025). *A Deeper Understanding of Options*. `https://bookdown.org/maxime_debellefroid/MyBook/a-deeper-understanding-of-options.html`. Accessed on August 13, 2025.

Eales, Brian A. and Moorad Choudhry (2003). *Derivative Instruments, a guide to theory and practice*. Butterworth-Heinemann.

Emmanuel, F., Olayinka Adedoyin Ajayi, and Okedele Olanrewaju Hammed (2014). "Performance measure of binomial model for pricing American and European options". *Applied and Computational Mathematics*.

Fadugba, Sunday Emmanuel, Olayinka Adedoyin Ajayi, and Olanrewaju Hammed Okedele (2014). "Performance Measure of Binomial Model for Pricing American and European Options". *Applied and Computational Mathematics*.

Gan, Lirong, Wang Huamao, and Zhaojun Yang (2020). "Machine learning solutions to challenges in finance: An application to the pricing of financial products". *Technological Forecasting and Social Change*.

Gautam, Jain (1997). "Enhanced Monte Carlo Estimates for American Option Prices". *The Journal of Derivatives*.

Göküzüm, Felix, Lu Trong Khiem Nguyen, and Marc-André Keip (2019). "An Artificial Neural Network Based Solution Scheme for Periodic Computational Homogenization of Electrostatic Problems". *Mathematical and Computational Applications* 24. DOI: 10.3390/mca24020040.

Gong, H., A. Thavaneswaran, and J. Sing (2010). "A Black-Scholes Model with GARCH Volatility". *The Mathematical Scientist*.

Gupta, S. P. (2017). *Corporate Finance*. https://ebrary.net/721/business_finance/corporate_finance.

Hassler, Uwe (2016). "Ito's lemma. Stochastic Processes and Calculus: An Elementary Introduction with Applications". *Springer International Publishing*.

Henderson, Vicky (2014). "Black–scholes model". *Wiley StatsRef: Statistics Reference Online*.

Hon, Robert (2013). *Monte Carlo and Binomial Simulations for European Option Pricing*. https://downloads.dxfeed.com/specifications/dxLibOptions/Hon.pdf. Accessed on August 13, 2025.

Hsia, C-C. (1983). "On the Binomial Option Pricing". *The Journal of Financial Research*.

Investopedia Staff (2024). "What Is the Stock Market and How Does It Work?" *Investopedia*.

Ivașcu, Codruț-Florin (2021). "Option pricing using machine learning". *Expert Systems with Applications*.

James Chen (2022). "American Option Definition, Pros & Cons, Examples". *Investopedia*.

Janková, Zuzana (2018). "Drawbacks and limitations of Black-Scholes model for options pricing". *Journal of Financial Studies and Research*.

Jarrow, Robert A. and George S. Oldfield (1981). "Forward contracts and futures contracts". *Journal of Financial Economics 9.4*.

Ju, Nengjiu and Rui Zhong (1999). "An approximate formula for pricing American options". *Journal of Derivatives 7.2*.

Klemkosky, Robert C. and Bruce G. Resnick (1979). “Put-call parity and market efficiency”. *The Journal of Finance 34.5*.

Malliaris, A. G. (1990). “Wiener process”. *Time Series and Statistics*.

Melcher, Kathrin (2021). *A Friendly Introduction to [Deep] Neural Networks*. KNIME. URL: `https://www.knime.com/blog/a-friendly-introduction-to-deep-neural-networks` (visited on 08/21/2025).

Misra, Dheeraj and Rajesh Dalmia (2007). “Hedging Performance of Protective Puts and Covered Calls Portfolio: A Study of NSE Nifty Options”. *Management Dynamics*.

Nielsen, Michael A. (2015). *Neural networks and deep learning*. USA: Determination press.

Petters, Arlie O. and Xiaoying Dong (2016). *An Introduction to Mathematical Finance with Applications*. Springer.

Reaz, Chowdhury et al. (2020). “Predicting the Stock Price of Frontier Markets Using Modified Black-Scholes Option Pricing Model and Machine Learning”. *Physica A: Statistical Mechanics and its Applications*.

Shinde, Akank Sha and Kalyanrao Chimaji Takale (2012). “Study of Black-Scholes model and its applications”. *Procedia Engineering 38*.

Smith, Clifford W. (1976). “Option Pricing: A Review”. *Journal of Financial Economics*.

Smith, V. Kerry (1983). “Option value: a conceptual overview”. *outhern economic journal*.

SO Staff (2012). *Lognormal Random Walk Model for Stock Prices Part 1*. `https://stockopter.com/lognormal-random-walk-model-for-stock-prices-part-1/`. Accessed on August 14, 2025.

Stulz, René M. (1984). “Optimal hedging policies”. *ournal of Financial and Quantitative analysis 19.2*.

Sutton, Clifton D. (2005). “Classification and regression trees, bagging, and boosting”. *Handbook of statistics 24*.

TidyFinance (2023). *Option Pricing via Machine Learning*. Accessed: 2025-09-14.

Van der Hoek, John and Robert J. Elliott (2006). *Binomial models in Finance*. Springer New York.

Wenda, Li. (2022). "Application of Machine Learning in Option Pricing: A Review". *Atlantic Press*.

Wilmott, Paul, Sam Howison, and Jeff Dewynne (1995). *The Mathematics of Financial Derivatives: A Student Introduction*. Cambridge University Press.

Xiaoping, Hu and Cao Jie (2014). "Randomized Binomial Tree and Pricing of American-Style Options". *Mathematical Problems in Engineering*.

Yadav, Mandar Pandurang (2025). "Machine Learning in Option Pricing: A Comparative Analysis of Traditional and ML-Based Models". *Indian Institute of Technology Madras*.