UCLouvain

LOUVAIN
School of Management

Louvain School of Management

# Implied volatility modelling and non-linear machine learning estimation of payoff replication performances in the Black-Scholes framework.

Auteur : Adrien Konopek
Promoteur : Frédéric Vrins
Année académique 2019 - 2020

"*The most important investment you can make is in yourself*"

Warren Buffet

**SUMMARY**

In this thesis, we provided a thorough work on the modelling of the implied volatility of the Black-Scholes model and the estimation of the hedging error through non linear machine learning techniques.

We will compute the implied volatility and make some adjustments to moneyness as depicted in the work of MacBeth and Merville followed by estimating a simultaneous implied volatility and implied risk-free rate as in the work of Bianconi and al. Afterwards, based on the constant volatility hypothesis of the Black-Scholes model, we will build the replicating portfolio and compute the hedging error at the option's maturity. We wish to know if there is a difference in using the MacBeth and Merville adjustment or the Bianconi and al. adjustment. Furthermore, we will train a non-linear machine learning algorithm to predict hedging errors. In this part, we wish to know if we can precisely predict hedging errors as well as the hedging type.

The results that emerge from this thesis are that there is a difference in the hedging performance using one implied volatility adjustment versus another one where the sole moneyness adjustment as MacBeth and Merville tends to have a higher hedging error compared to the adjustments of Bianconi and al. The results regarding the prediction of the machine learning algorithm are that it can predict whether the replicating portfolio under or over hedges with an accuracy of 84.28% and predicting the exact value of the hedging error leads to an MSE of 26.83 and a RMSE of 5.18.

The implications coming forward in this thesis are that there is no perfect method to estimate the implied volatility as it depends on the goal of the investor and that hedging error prediction is possible to a certain extent.

## ACKNOWLEDGEMENTS

I would like to thank first and foremost my supervisor Pr. Frédéric Vrins for his expertise and experience in the field and without whom this thesis would not have been possible, steering me to the right direction when digressing from the main goal. I would also like to thank him for the classes taught and the growing interest he passed on me.

Secondly, I would also like to thank Dr. Nathan Lassance for his availability and his expertise. I would like to thank him for the data he sent me and without which this thesis would not have been possible either.

Last but not least, I would like to thank my family for the support they have given me throughout these years at university. My mother for her constant encouragements and interest in the subject and my father for the countless discussions we had. I would not be the person I am today and achieved what I have achieved without them.

# CONTENTS

**FIGURES**

## TABLES

# 1 INTRODUCTION

## 1.1 *Context & Research question*

### 1.1.1 Context

The financial markets are essential to the development and the growth of the global economy and its complexity has grown ever since the inception of the first stock market in Amsterdam in the $17^{th}$ century. This complexity has grown exponentially since the introduction of derivative instruments. In this thesis we will focus on one type of derivative instrument, the options. Options are traded both on two types of markets: the regulated market and the OTC (over the counter) market. The most common and hence the most traded options on the market are the call and put options. A call option is defined as a financial instrument that "gives the holder the right to buy an asset by a certain date for a certain price"[1] [17].
Amongst other factors, the misunderstanding of complex financial derivative instruments led to the disastrous 2008 financial crisis, making today's risk management even more important. In term of numbers, the worldwide GDP is as high as $91.98 trillion [39] where as the global notional amount of OTC derivatives rose up to $ 640 trillion [6]. In light of these numbers, the professionals of the field cannot allow themselves not to be thorough with their understanding of these instruments.

Today, options are used by professionals for many purposes. Speculating, arbitrage or hedging portfolios and in this thesis, we will focus on the latter. Indeed, hedging portfolios has increased in complexity and performance since the option's introduction and has risen many questions in the field ever since. The most basic questions in finance would be to know how to price those options. Nobel prizewinners Fisher Black, Myron Scholes and Robert Merton were the first to price options based on no-arbitrage and continuous stochastic processes. They were the first to publish an article developing the mathematical solution for option pricing, Robert Merton in *"Theory of Rational Option Pricing"* and Fisher Black alongside Myron Scholes in *"The Pricing of Options and Corporate Liabilities"*. The success of the model speaks for itself as numerous investment banks, hedge funds and mutual funds are still using their model. As all paradigm, the model has been challenged by its peers and criticized for its simplicity. Indeed, the

[1]Hull, J. (2016). *Fundamentals of Futures and Options Markets* (8th ed.) Pearson.

model sets a lot of unrealistic hypothesis still being challenged. The BSM (Black-Scholes-Merton) model takes in a set of parameters and outputs the option's price. One of the most important and most difficult parameter to estimate is the volatility. In this thesis, we will focus on the many ways to estimate this volatility.

### 1.1.2 Research question

This section will be divided in two subsections. The first section will discuss the different methods implemented to model the implied volatility. The second section will discuss the second part of the thesis namely the machine learning part that will try to predict over or under-hedging performances in a payoff replication strategy.

In the literature, the volatility can be retrieved and computed in many ways. The most common and easy way is to force the BSM price to be equal to the market price and to change the value of the input volatility parameter until the equation reaches equality. The volatility is called the implied volatility and is a concept that will be tackled in this thesis. We will focus on 2 methods retrieved from the literature and apply both methods on our data. Our data consists of call option prices from one of two of the most traded securities in the world, the S&P 500 and the Apple stock. The two methods we will apply are:

- The classical method used to retrieve the implied volatility by minimization of the residual square and applying the McBeth and Merville regression accounting for moneyness.
- The simultaneous estimation of the implied volatility and the implied risk-free rate developed by Bianconi & al where the implied volatility will be estimated through a classical regression (OLS) and a seemingly unrelated regression on a regularized and unregularized loss function.

Given those 2 methods, the first sub question to our research question is: "*Does a specific implied volatility model change the hedging error in terms of payoff replication?*

The main goal of this section will be to build the data set that we will have to use in the second section. We will hence compute several sets of implied volatility given different maturities and strike prices.

After performing those tests, we will be focusing on the machine learning part to see if there are any patterns in order to predict the range of the hedging errors. As for all machine learning projects, one of the most important steps for a successful project is the data base construction and its quality. Given that it has been performed in the first section, the second section will focus on the ML (machine learning) techniques used. We will use in this thesis, the famous neural network and not use a standard linear regression. The advantage of a neural network is that it is non-linear with respect to the linear regression. Given a first draft of methodology, we can already outline the second sub question: "*Can non linear machine learning techniques predict out-of-sample hedging errors in terms of payoff replication?*

The goal of the first research question is to be able, after all the analysis, to know if a method tends to over-hedge a portfolio or on the other hand tends to under-hedge it. The second research question's goal will be to know if we can predict the scale of the hedging errors given the inputs available at inception such as the strike price, underlying price or implied volatility just to name a few. This could become handy to financial institutions in terms of competitiveness and to lower prices given a specific set of parameters.

## 1.2 *Organization*

This thesis will be split alongside the following chapters.

- **Chapter 2**: We will start by describing the basics needed for the understanding of the thesis such as the Black-Scholes model and its most basic assumptions, the delta hedging, the implied volatility modelling and the problems of volatility skew and smile.
- **Chapter 3**: In this chapter, we will expose theoretically the methods used to compute the implied parameters.
- **Chapter 4 & 5**: We will implement the methods to our data and we will analyze and discuss the results. We will propose an answer to the first sub research question.
- **Chapter 6**: We will here outline the machine learning technique used and discuss the results. We

will also propose an answer to the second sub research question.

- **Chapter 7**: We will here digress from our main topic and discuss the role as well as the importance of ethics in quantitative finance.
- **Chapter 8**: We will end our thesis by wrapping up our results, outlining the limitations of the research and we will end by making some suggestions for future research.

# 2 BLACK-SCHOLES MERTON MODEL: CONCEPTS

## 2.1 *Option's properties*

In this section, we will discuss the properties of the options as described in [17]. Options are a type of financial instruments and can also be divided into subsections: plain vanilla options also known as *European options* and exotic options such as Asian options, American options or even the challenging cliquet option. We will here focus on plain European vanilla options as they are the easiest to price. We ought to clarify several parameters from the definition in the introduction. In a European option, the option holder has the right to purchase the asset at a certain price at a certain date. The price at which the holder buys the underlying is called a strike price that we denote $K$ and the date at which the option is exercised is the maturity that we will denote $T$. The time at which the option is priced will be denoted $t$, hence the time to maturity will be denoted $\tau = T - t$. The underlying price at inception (or at the time of pricing) will be noted $S_t$. The fact that an option is European, means it can only be exercised at maturity compared to an American option that can be exercised all the way through maturity. Just like a traditional sale where there is a buyer and a seller, a European option can be divided between a call and a put option. The call option holder has the right to buy the underlying asset whereas the put option gives the right to sell the underlying at the given strike price. The important side of every sale is the payoff and the payoff of a European option is the price of the option that can be written as follow:

$$
\begin{aligned}
C_T &= max(0, S_T - K) \\
P_T &= max(0, K - S_T)
\end{aligned}
$$

**Equation 2.1.**

We can already make an assumption about an option holder's position about the market based on what they hold. On one hand, call option holders can be described as bullish investors whereas on the other hand a put option holder is described as a bearish investor.

In this thesis, we will only focus on call options. One very convenient way to account put options given call options is the Put-Call parity. This equation relates the put and call option price with the following equation:

$$P_t + S_t = C_t + Ke^{-r(T-t)}$$

**Equation 2.2.**

where $r$ is the risk-free rate.

### 2.2 *Model's assumptions*

One of the greatest financial models created has been developed by Fisher Black and Myron Scholes with the help of Robert Merton. This model prices European options with a set of parameters. The BSM model makes a set of assumptions and we will develop them in the following section. From [17] and from [7] we can outline the following assumptions:

- The underlying stock price follows a geometric brownian motion.
- We assume constant volatility $\sigma$ and constant risk-free rate, $r$.
- No transaction costs.
- The underlying stock hands no dividend over.
- The option can only be exercised at maturity.
- It is possible to borrow or to hold cash at the short-term risk free rate, $r$.
- No penalty for short selling and continuous trading.

### 2.3 *Geometric Brownian Motion*

As defined in the first assumption of the model, the underlying stock supposedly follows a geometric Brownian motion. A stochastic process can be defined from a stochastic differential equation, i.e. by an equation of the form:

$$dS_t = \mu(t, S_t)dt + \sigma(t, S_t)dW_t \qquad \textbf{Equation 2.3.}$$

where $\mu(t, S_t)$ is the deterministic part called the *drift*, $dW_t$, the Wiener process under the probability measure $\mathbb{P}$, withholds the randomness of the process and $\sigma(t, S_t)$ is the diffusion coefficient. It is interesting to note that in economical terms, the drift is often depicted as the instantaneous expected return. Given Equation 2.3, we can adapt a little bit the formula to what is needed in the BSM model and we get to the following equation, being the geometric Brownian motion. In this Brownian Motion the drift and diffusion coefficient are given by $\mu(t, S_t) = \mu S_t$ and $\sigma(t, S_t) = \sigma S_t$.

$$dS_t = \mu S_t dt + \sigma S_t dW_t \qquad \textbf{Equation 2.4.}$$

Equation 2.4 is handy as in this case we know the distribution of $S_t$ and $dt$ which is a function of $W$. The distribution of the underlying $S_t$ follows a lognormal distribution that can be written as follow [17]:

$$ln(S_T) \sim \phi[ln(S_0) + (\mu - \frac{\sigma^2}{2})T, \sigma^2 T] \quad \textbf{Equation 2.5.}$$

## 2.4 *Risk-neutral measure*

Based on Girsanov's theorem's change of probability measure, an important feature that needs to be understood is the concept of risk-neutral measure or risk-neutral probability. As defined in [15] (adapted for a constant risk-free rate), "a risk neutral measure is a measure $\mathbb{P}$ that is equivalent to $\mathbb{Q}$ under which the stock price discounted at the rate $r$ is a martingale"[2]. The existence of a risk-neutral measure implies that there is no arbitrage opportunity, *i.e.* there is no "free lunch" meaning that one cannot have a higher rate of return than the risk-free rate without bearing any additional risk. This concept is important for the rest of the thesis as it allows us to replace the expected return of a stock price (the risk premium) by the risk-free rate. A hint could have been given through the Girsanov's theorem as "in a SDE, only the drift changes when changing the measures, not the diffusion coefficient"[3]. We will hence be able to replace the drift $\mu$ in equation 2.4 and in equation 2.5 by $r$ in the $\mathbb{Q}$ probability measure. In order to distinguish the upcoming equation from equation 2.4, we will need to write the Wiener process $dWt$ in the $\mathbb{Q}$ probability measure denoted as $d\tilde{W}_t$.
We will hence rewrite equation 2.4 as follow:

$$dS_t = rS_t dt + \sigma S_t d\tilde{W}_t \quad \textbf{Equation 2.6.}$$

## 2.5 *The Model*

Now that we have all the tools necessary for the development of the BSM model, let's dig into it. All the elements for the demonstration of the BSM rely on Hull, 2016[4], Black and Scholes, 1973[5] and van der Wijst, 2013[6]. We will not focus on the whole demonstration beginning from the Ito lemma whose

[2] Gautam, L. (2016). *Risk Neutral Measures*. Carnegie Mellon University, Mellon College of Science. Retrieved from `http://www.math.cmu.edu/~gautam/sj/teaching/2016-17/944-scalc-finance1/pdfs/ch4-rnm.pdf`

[3] Vrins, F. (2019). *Derivatives Pricing*. Louvain School of Management, Course Material.

[4] Hull, J. (2016). *Fundamentals of Futures and Options Markets* (8th ed.) Pearson.

[5] Black, F., & Scholes, M. (1973). The Pricing of Options and Corporate Liabilities. *The Journal of Political Economy*

[6] van der Wijst, N. (2013). *Finance: a Quantitative Introduction*. Cambridge: Cambridge University Press.

formulae can be found in [46].

Given a stock price $S_t$, following a geometric Brownian motion at a given time $t$, for a given maturity $T$ (hence a time to maturity $\tau = T - t$), a strike $K$ a risk-free rate $r$ and a given volatility $\sigma$, the price of a call and a put option is given by the following formula, known as the Black-Scholes-Merton formula where the payoff is given by equation 2.1:

$$C_t = S_t N(d_1) - Ke^{-r\tau} N(d_2)$$
$$P_t = Ke^{-r\tau} N(-d_2) - S_t N(-d1) \qquad \textbf{Equation 2.7.}$$

Where $d_1$ and $d_2$ are given by:

$$d_1 = \frac{ln(\frac{S_t}{K}) + (r + \frac{\sigma^2}{2})\tau}{\sigma\sqrt{\tau}}$$
$$d_2 = \frac{ln(\frac{S_t}{K}) + (r - \frac{\sigma^2}{2})\tau}{\sigma\sqrt{\tau}} \qquad \textbf{Equation 2.8.}$$

According to [28], $N(d_2)$ can be explained as the probability that the option will be exercised or not under the risk-adjusted probability. $N(d_1)$ is seen as the probability that an option will end in or out of the money. An option is said to be *in the money* (ITM) when the price of the underlying at maturity is higher than the exercise price for a call option and lower for a put option. An option is said to be *out of the money* (OTM) when the price of the underlying at maturity is lower than the exercise price for a call option and higher for a put option [33]. Usually, options are quoted at inception *at the money* (ATM) where the strike or exercise price is equal to the underlying's price.

## 2.6 *Delta and replicating strategy*

Now that we have defined the BSM model, we can tackle the strategy that a lot of professionals use. In order to understand that strategy it is important to define the first derivative of the BSM model, the delta $\Delta$. The delta is one of the Greeks that measures the sensitivity of the option's price with respect to an input of the model. Here the delta measures the sensitivity of the option's price with respect to the underlying's price. It is given by the following formula for a call option:

$$\Delta_t = \frac{\partial C_t}{\partial S_t} = N(d_1) \qquad \textbf{Equation 2.9.}$$

This delta is very important in the payoff replication strategy as it indicates how much of the underlying stock an option writer has to buy in order to hedge himself against the stock's inherent risk. Before digging into this strategy let's first assume a few things. According to [44], we assume a market with a single risky asset, $S_t$ and a risk less cash account that pays a return equal to the risk-free rate $r$. We can also borrow money at this same rate. This strategy also assumes to be *self financing*, meaning that no additional cash from an individual investor can be added to the portfolio and that the portfolio's value changes only due to the change in value of its assets. [44] Defines the replicating portfolio and strategy as following: "A portfolio Π (resp. a trading strategy $\vec{\Delta}$) is replicating a derivative security V if Π (resp. $\vec{\Delta}$) delivers the same stream of cash flows as V $\mathbb{P}$-almost surely"[7]. In equation terms, this replicating strategy can be written as follow:

$$d\Pi_t = \Delta_t dS_t + r(\Pi_t - \Delta_t S_t)dt \quad \textbf{Equation 2.10.}$$

Where $\Pi_0 = C_0$ is the option's premium and where the hedging error or the hedging performance is given by the following equation:

$$\varepsilon_T = \Pi_T - C_T = \Pi_T - max(0, (S_T - K)) \quad \textbf{Equation 2.11.}$$

This equation describes the position of a call option writer. In order to know the position of an option buyer, we just need to flip the equation. Under the BSM framework, this strategy ends up with a payoff equal to the portfolio, meaning that $\varepsilon_T = 0$. We will also need to compute the mean of the hedging error through the following equation:

$$\mathbb{E}[\varepsilon_T] = \frac{1}{n}\sum_{i=1}^{n}\varepsilon_{Ti} = \frac{1}{n}\sum_{i=1}^{n}\left[\Pi_{Ti} - max(0, S_{Ti} - K_i)\right] \quad \textbf{Equation 2.12.}$$

Where $n$ is the number of observed call options on which the dynamic strategy has been performed.

The fact that $\varepsilon_T = 0$ can be shown in the following figure where the initial inputs are as follow: $K = S_0 = 100$, $dt = 0.01$, $\sigma = 0.2$, $r = 0.01$ and the maturity $T$ is 5 years. One problem can already be outlined here being the fact that one of the assumptions of the BSM model was that trading needs to be made continuously. On figure 2.1, the time step is small enough to get close to a null hedging error. This phenomenon can be defined as the *rebalancing frequency* being the frequency at which we rebalance our portfolio.

[7] Vrins, F. (2019). *Credit & Interest Rates Risk*. Louvain School of Management, Course Material.

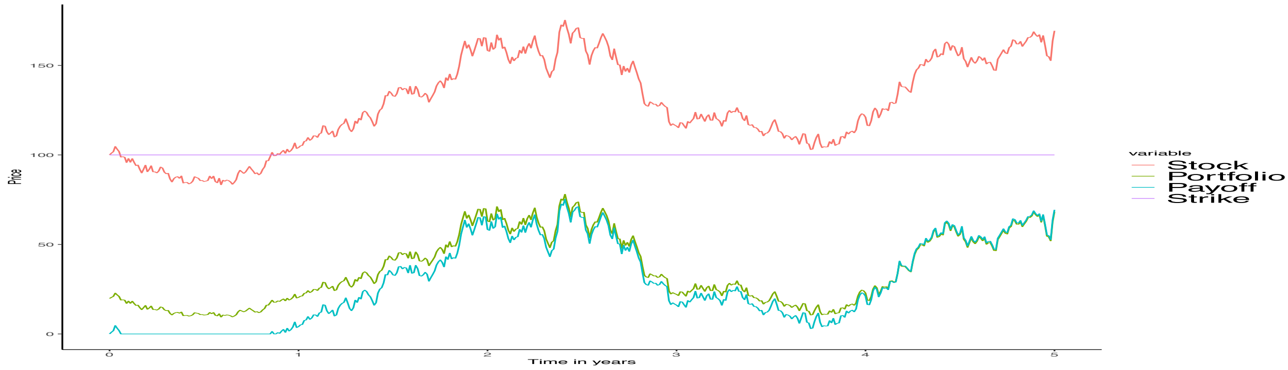
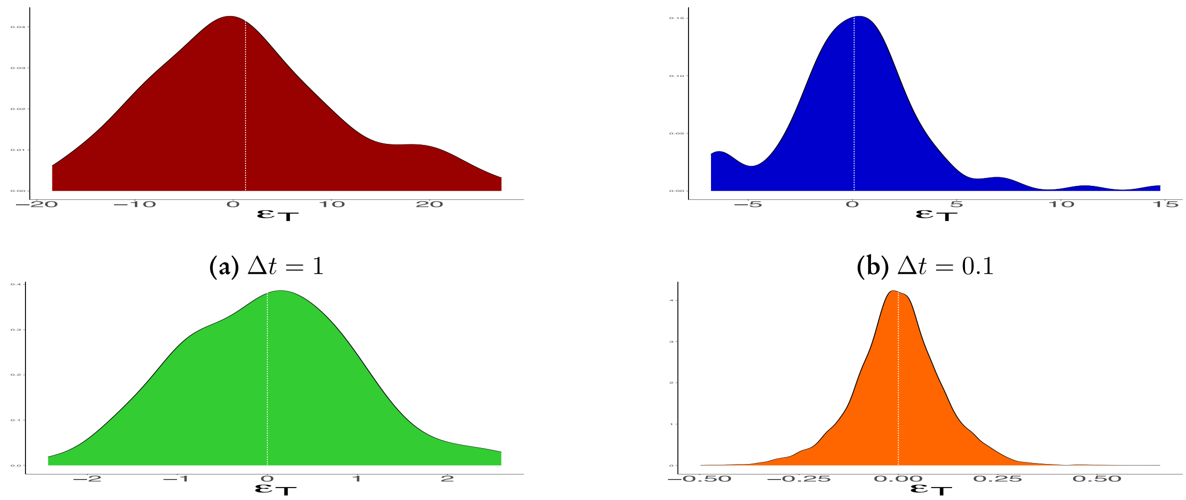



**Figure 2.1:** Payoff replication strategy under a GBM

The above figure also depicts the phenomenon where the price of the underlying goes below the strike price and the option holder has no interest in exercising the option. The payoff at these moments is always 0. As [36, p.17] said and when we need to discretize equation 2.10, the expected value of $\varepsilon_T$, $\mathbb{E}[\varepsilon_T]$ when the rebalancing frequency is higher tends to be 0. This is shown in the following figures depicting several density plots for lower to higher rebalancing frequencies. These graphs have been computed thanks to the simulation of 100 underlying stock paths. Ideally, a financial company would want the hedging error to be as close to 0 as possible, being as predictable as possible. According to Black-Scholes, the only way to succeed doing that is to rebalance the portfolio as often as possible and doing that for as many stocks as possible as depicted in plot 2.2d where the white dotted line is the mean of the plotted density.




**(a)** $\Delta t = 1$

**(b)** $\Delta t = 0.1$

**(c)** $\Delta t = 0.01$

**(d)** $\Delta t = 10^{-3}$

**Figure 2.2:** Density of $\varepsilon_T$ for different $\Delta t$

Still according to [36], this problem led by the rebalancing frequency does not take into consideration the transaction costs and other related fees. One way to overcome this problem would be to decrease the gamma value of the portfolio. Gamma is a second order derivative greek that measures the sensitivity of the delta with respect to the stock price. In light of these graphs we can define the concepts of under hedging and over hedging a portfolio. Over-hedging ($OH$) is defined as a financial position strategy that involves the opening of an offsetting position worth more than the underlying protected position [20]. An under-hedged ($UH$) position, the opposite of the over-hedged position, is worth less than the underlying protected position. In equation terms, this would lead to the following equations:

$$OH : \Pi_T > max(S_T - K, 0) \Leftrightarrow \varepsilon_T > 0 \qquad UH : max(S_T - K, 0) > \Pi_T \Leftrightarrow 0 > \varepsilon_T \qquad \textbf{Equation 2.13.}$$

Knowing those terms, it is handy to have a flavour of why some replication strategies tend to over-hedge a position or on the flip side, tend to under-hedge them. In order to understand that, we will need to dig deeper in the understanding of the "Greeks" that we will cover in the next section.

## 2.7 *The Greeks*

In this section we will propose an explanation about why some strategies are over-hedged or under-hedged. This will be done by introducing the concepts of Gamma and Theta. Before analyzing those terms, we ought to remember that the goal of the dynamic delta strategy is to replicate the price of the option[8]. Bearing that in mind, we first need to define the infinitesimal change in the value of the option, $dC_t$ whose equation is given by applying Ito's lemma (as in [14]).

$$dC_t = \Theta_t dt + \Delta_t dS_t + \frac{1}{2}\Gamma_t dS_t^2 \qquad \textbf{Equation 2.14.}$$

This equation needs some explanations regarding the absence of a volatility term, $\sigma$ or a risk-free rate term, $r$. This is due to the fact that this equation outlines a change in the option's price. For this reason and because of the fact that $\sigma$ and $r$ are held constant, $d\sigma$ and $dr$ are null. This does not mean that $r$ and $\sigma$ do not impact the price of the call option, they do (Equation 2.7). They will not impact the **change** in the option's price.

In Equation 2.14, $\Theta$ and $\Gamma$ are given by the following equation:

$$\Gamma_t = \frac{n(d_1)}{S_t \sigma \sqrt{\tau}} \qquad \& \qquad \Theta_t = -\frac{S_t n(d_1)\sigma}{2\sqrt{\tau}} - rKe^{-r\tau}N(d_2) \qquad \textbf{Equation 2.15.}$$

[8]Note that we apply all equations and all observations for a call option

Where $n(\bullet)$ is the cumulative distribution function of a normal distribution and $d_1$ and $d_2$ are defined as in equation 2.8. Based on those equations, we know that $\Theta$ is always negative and $\Gamma$ always positive. This makes sense for $\Theta$ as the value of the option is dropping as time passes by. For $\Gamma$, we can see that all terms are positive, a cumulative distribution function of the normal distribution being between 0 and 1, a stock's price and time being positive by their nature and volatility being positive by definition. According to [31] the positive convexity of the option has to be paid for which is paid through time decay, $\Theta$. According to [44], the portfolio that mimics the derivative but is not the derivative satisfies the following equation under the BSM framework:

$$r(\Pi_t - \Delta_t S_t) = \Theta_t + \frac{\sigma^2 S_t^2}{2}\Gamma_t \qquad \& \qquad r\Pi_t = \Theta_t + \frac{\sigma^2 S_t^2}{2}\Gamma_t \qquad \textbf{Equation 2.16.}$$

Where the equation on the right is the equation where the replicating portfolio is delta neutral. The following equation shows the change in portfolio in a delta neutral portfolio:

$$d\Pi_t = \Theta_t dt + \frac{1}{2}\Gamma_t dS_t^2 \qquad \textbf{Equation 2.17.}$$

This gives us a nice understanding on how $\Theta$ and $\Gamma$ impact the change in value of an option. Nevertheless, the above mentioned equations needs to be approximated as in real life, it is impossible to rebalance the portfolio continuously and hence compute the change in the option's value at an infinitesimal range. Indeed, we will need to adapt $dC_t = C_{t+dt} - C_t$ to $\Delta C_t = C_{t+\Delta t} - C_t$. We will here set the delta given by equation 2.9 as $\Delta^{BS}$ to avoid any confusion in terms. Now that we are looking to define $\Delta C_t$ we can approximate it through a Taylor expansion showing that the change in value of $C_t$ is related to its Greeks:

$$\Delta C_t \approx \frac{\partial C_t}{\partial S_t}\Delta S_t + \frac{\partial C_t}{\partial T}\Delta T + \frac{\partial C_t}{\partial r}\Delta r + \frac{\partial C_t}{\partial \sigma}\Delta\sigma + \frac{1}{2}\frac{\partial^2 C_t}{\partial S_t^2}(\Delta S_t)^2 + ... \qquad \textbf{Equation 2.18.}$$

Given the fact that in the BSM framework, the volatility and the risk-free rate are held constant, $\Delta\sigma$ and $\Delta r$ are worth 0. We need to outline that this equation is an approximation of the change in the option's price and the equation would hold exactly true only with higher order derivatives. We will hence also rewrite equations 2.10 and 2.14. This would lead to the following equations:

$$\Delta\Pi_t = \Delta_t^{BS}\Delta S_t + r(\Pi_t - \Delta_t^{BS} S_t)\Delta t \qquad \textbf{Equation 2.19a.}$$

$$\Delta C_t \approx \Theta_t \Delta t + \Delta_t^{BS}\Delta S_t + \frac{1}{2}\Gamma_t \Delta S_t^2 \qquad \textbf{Equation 2.19b.}$$

Where equation 2.19a is the discretization of equation 2.10 and equation 2.19b is the approximation of equation 2.14. This would lead to approximate the hedging portfolio through higher order derivatives and gives us the following equation:

$$\Delta\Pi_t = \Theta_t \Delta t + \Delta_t^{BS}\Delta S_t + \frac{1}{2}\Gamma_t \Delta S_t^2 + r(\Pi_t - \Delta_t^{BS} S_t)\Delta t \qquad \textbf{Equation 2.20.}$$

Because we approximate the change in value of the option, $dC_t$ to $\Delta C_t$, we have that $\varepsilon_T$ will not exactly be equal to 0.

In section 4 and 5, we will use equation 2.19a to compute our hedging error $\varepsilon_T$ as in equation 2.11. In section 5.2, we will compare the hedging error $\varepsilon_T(\Delta^{BS})$ computed thanks to equation 2.19a (where the only computed greek is the $\Delta^{BS}$) and the hedging error $\varepsilon_T(\Delta^{BS}, \Theta, \Gamma)$ computed thanks to equation 2.20 (where the computed greeks are $\Delta^{BS}$, $\Theta$ and $\Gamma$).

The observation made about the signs of $\Theta$ and $\Gamma$ as well as the above mentioned equations lead us to understand the over and under-hedging phenomenon. Indeed, if a call option is only defined by its $\Delta^{BS}$, *i.e.* the first term on the right hand side of the Taylor expansion (equation 2.18), we saw in section 2.6 that $\varepsilon_T$ tends to have a 0 mean with a small time step $\Delta t$, meaning that the $\Theta$ effect is equal to the $\Gamma$ effect and hence that the change in the portfolio at maturity in equation 2.17 is null. Based on equation 2.17 and equation 2.20, if on one hand, $\varepsilon_T$ has a positive mean, *i.e.* the portfolio is over-hedged, the $\Gamma$ effect is larger than the $\Theta$ effect whereas on the other hand, if $\varepsilon_T$ has a negative mean, *i.e.* the portfolio is under-hedged, the $\Theta$ effect has a bigger impact than the $\Gamma$ effect. This observation could come in handy if at the point where we compute $\varepsilon_T$, its mean is positive we know it could come from a positive $\Gamma$ effect.

Should one be interested in greeks and how it behaves with different input parameters, an application can be found on the following web site: `https://akonopek.shinyapps.io/delta_app/?fbclid=IwAR2bsJgl3L96MCJJ1d5cmu_UiXQ4I1mtdxP9lRnOjE37o-sBOircH9QH614`. On this website, it can be shown that $\Theta$ is always negative or equal to 0 and $\Gamma$ always positive or equal to 0.

## 2.8 Limitations

The limitations of the BSM model are numerous and one of its most controversial assumptions is that the model assumes a constant volatility. In this thesis, we will not try to compute a deterministic volatility as the local volatility developed by Bruno Dupire [13] nor a stochastic volatility model such as the one developed by Steven Heston [16]. In the first part of the thesis we will focus on the modelization of the implied volatility. The implied volatility is "the measure of how the options are quoted by the market participants"[9]. In other words, on the market, we can find the prices of the options and several of its parameters. The others need to be estimated or calibrated. The implied volatility is estimated by implementing the market's price in the BSM model and the volatility that equates both prices is the implied volatility. To quote Riccardo Rebonato in [30], "the implied volatility is the wrong number to put in the wrong formula to get the right price"[10].

In a BSM framework, the implied volatility is constant and draws a flat line with respect to a range of strike prices. In real life cases, as expected, it is not the case and this phenomenon is known as the volatility skew or volatility smile. The volatility skew depicts the fact that the implied volatility is higher for out of the money options and lower for at the money options. The implied volatility skew has been explained by [12]: regarding equities specifically, they are one of the most highest-returning asset and this is the main reason why people want long exposure in this asset class. This tends to increase the demand for ITM call options meaning that at the same time, stocks are more volatile and the higher the demand, the higher the volatility. Plots for the S&P500 volatility skews can be found in the Appendix on figure A.1. The volatility smile phenomenon depicts the fact that out of the money and in the money options have higher implied volatilities and at the money options have lower implied volatilities. According to [17], volatility smiles are more observable for currencies options.

[9] Ursone, P. (2015). How to Calculate Options Prices and Their Greeks: Exploring the Black Scholes Model from Delta to Vega. [ebook] Wiley.

[10] Rebonato, R. (1999). *Volatility and correlation in the pricing of equity, FX and interest-rate options.* Wiley

# 3 IMPLIED VOLATILITY MODELIZATION: THEORY

## 3.1 *Implied Volatility under moneyness adjustment*

First of all, we will compute the implied volatility the classical way, solving an optimization problem by minimizing the sum of square between the given call price by the market and the BSM call price. For a given set of strikes $K_i$ and a corresponding set of maturities $T_i$ where $i$ is the $i^{th}$ observed call option in a set $I$ of observed options, the implied volatility is computed as follow:

$$\sigma_i^* \equiv Arg \min_{\sigma} [C_i^{mkt} - C_i^{BSM}(\sigma_i, K_i, T_i)]^2 \qquad \forall i \in I$$ **Equation 3.1.**

We will run this minimization by using the Brent algorithm developed by Richard Brent (1973) [9]. This algorithm actually withholds 3 other different algorithms to choose from based upon the speed of computation. It is based upon the root-finding algorithm, the secant method and the inverse quadratic interpolation.
Because the next step is to run a regression on the implied volatility, we need to run an exploratory analysis and specifically handle outliers. This will allow the regression to be run without extreme values. In order to do so, we will successively remove data above the $3^{rd}$ quartile and below the $1^{st}$ quartile.

Now that we have found the implied volatility, we will need to adjust it to the moneyness of the option. This will allow us to take into consideration the irregularities in the implied volatility surface and better handle the skew and smile effects. This will also allow us to price at the money options that are almost never observable on the market and will allow institutions to sell correctly priced options.
Following MacBeth & Merville [23], we will need to run a regression on the implied volatility given the moneyness of the option. The moneyness $m_{ijt}$ for option $j$ on the underlying security $i$ at day $t$ is computed as follow, where $\tau = T - t$ is the time to maturity in years:

$$m_{ijt} = \frac{S_{it} - X_{ij}e^{-r\tau}}{X_{ij}e^{-r\tau}}$$ **Equation 3.2.**

The equation of the regression is written as follow:

$$\sigma_{ijt}^* = \theta_{0it} + \theta_{1it}m_{ijt} + \epsilon_{ijt}$$ **Equation 3.3.**

This regression will be computed through the classical ordinary least square error optimization on the R software. Given the regression parameters, we can already say that $\theta_{0it}$ is the implied volatility for at the money traded options as $m_{ijt} = 0$.

After daily estimating the parameters of the regression, we will estimate the implied volatility upon the strike price's present value and the underlying's current price. We will hence be able to estimate the price of at the money options through the regression of equation 3.3. Once the regression has been run and the implied volatility been estimated, we will proceed to the replicating strategy and compute the hedging errors. With this final data, the hedging errors will be used as out-of-sample data to the machine learning algorithm.

## 3.2 *Implied Volatility and risk-free rate*

In this section we will rely on the work of Bianconi et al. [3] and the way they computed the implied volatility differs from the traditional one as they perform a simultaneous optimization to retrieve 2 parameters from the BSM model: the implied volatility and the implied risk-free rate. Solving 2 equations with these 2 unknowns has already been performed in the literature, namely Krausz (1985)[21] or Swilder (1986)[35]. The goal here for the system of equation is to solve for both $C_1^{mkt} = C_1^{BSM}(\sigma^*, r^*)$ and $C_2^{mkt} = C_2^{BSM}(\sigma^*, r^*)$. This papers sets itself apart from the others as it offers a way to compute this system through another optimization problem in one single minimization equation that takes the following form.

$$(\sigma^*, r^*) = Arg \min_{(\sigma,r)} F(\sigma, r) \equiv Arg \min_{(\sigma,r)} \frac{[C_1^{mkt} - C_1^{BSM}(\sigma, r)]^2}{[C_1^{mkt}]^2} + \frac{[C_2^{mkt} - C_2^{BSM}(\sigma, r)]^2}{[C_2^{mkt}]^2} \quad \textbf{Equation 3.4.}$$

This method has its advantages compared to the previous one as it does not rely on the assumption that a unique solution for $F(\sigma, r) = 0$ exists. The algorithm used to find the minimum of this function is the Nelder-Mead algorithm developed by John Nelder and Roger Mead in [26]. This algorithm is a simplex non linear algorithm and needs estimated inputs to start. This comes in handy as we will run the algorithm 3 times with the output of the previous minimization that will be used as the input for the following minimization. This allows us to avoid finding the local minimum of the quadratic function. One of the drawbacks of this algorithm is that it is unconstrained and we will end up with extreme values for the implied volatility and risk-free rate. Based upon data observation this is mainly due to high trading volume in that day and hence a big spread between the ask and bid prices. This will be handled by removing all the outliers from the data for the next step.

Just like in the first method, outliers will be sorted and deleted from the estimated parameters to make economical sense. Indeed, having a risk-free rate as high as 50% does not make any sense for an option's price retrieved in 2016.

Previously, a moneyness adjustment has been made to re-estimate the implied volatility. For the simultaneous implied volatility and risk-free rate, this adjustment has been brought forward by Krausz [21] as the following equations:

$$\sigma^*_{ijt} = \theta_{0t} + \theta_{1t} m_{ijt} + \epsilon_{ijt} \qquad \textbf{Equation 3.5a.}$$

$$r^*_{ijt} = \rho_{0t} + \rho_{1t} m_{ijt} + \varepsilon_{ijt} \qquad \textbf{Equation 3.5b.}$$

where $m$ is defined as in equation 3.2 and the risk-free rate in that moneyness equation is defined as the average model implied risk-free rate. As Bianconi and al. outlined, this causes and endogeneity problem as $r$ is on both sides of the regression. The regression proposed in [3] takes the following form:

$$\sigma^*_{ijt} = \alpha_{0t} + \alpha_{1t}[ln(S_{it}) - ln(X_{ij}) + r_{ijt}\tau] + \epsilon''_{ijt} \qquad \textbf{Equation 3.6a.}$$

$$r^*_{ijt} = \beta_{0t} + \beta_{1t}\tau + \beta_{2t}[ln(S_{it}) - ln(X_{ij})] + \beta_{3t}[ln(S_{it}) - ln(X_{ij})\tau] + \epsilon''_{ijt} \qquad \textbf{Equation 3.6b.}$$

The first regression can be interpreted just as in the first method where the first estimated parameter, $\alpha_0$ is the implied volatility for at the money options.

The next step would be to estimate theses parameters through a regression. We decided to run a regular OLS regression to estimate the risk-free rate regression and then implement the estimated parameters into the implied volatility regression. We will compare the OLS estimation with a seemingly unrelated regression. This regression makes a lot of sense as it needs the independent variables of both regressions to be different and that the errors of both regression need to be correlated which is the case here. Bliss and Panigirtzoglou (2004) said that "risks are volatility dependent"[11]. Bianconi and al. outlined the advantages of using a SUR model: "The implications of using the SUR model are more evident when considering shocks that enter the system. Without using SUR, a shock in the error term for the risk-free rate regression has no impact on the volatility regression. With SUR, a shock in the error term for the risk-free rate regression also shocks the error term for implied volatility regression and vice versa."[12].

[11] Bliss, R., & Panigirtzoglou, N. (2004). Option–Implied Risk Aversion Estimates. *The Journal of Finance*, 59(1), 407-446.

[12] Bianconi, M., MacLachlan, S., & Sammon, M. (2015). Implied volatility and the risk-free rate of return in options markets. *North American Journal of Economics and Finance*, 31, 1-26.

Once these estimation have been performed through both methods, we will be able to go on with the out-of-sample machine learning predictions.
Before moving on with the method's comparison, Bianconi and al. outlined another interesting feature that can and will be performed. They introduced a regularization term in the optimization function being the L2 regularization developed by Tikhonov (1977) [37]. This regularization term can be seen as a penalty term in the optimization function. This regularization takes the general form:

$$||Ax - B||^2 + ||\Gamma x||^2 \quad \textbf{Equation 3.7.}$$

In our case, the penalty term is applied to the model implied risk-free rate and takes the following minimization:

$$(\sigma^*, r^*) \equiv Arg \min_{\sigma,r} \frac{[C_1^{mkt} - C_1^{BSM}(\sigma, r)]^2}{[C_1^{mkt}]^2} + \frac{[C_2^{mkt} - C_2^{BSM}(\sigma, r)]^2}{[C_2^{mkt}]^2} + P(r - R)^2 \quad \textbf{Equation 3.8.}$$

This penalty term measures by which degree the model implied risk-free rate diverges from the estimated risk-free rate $R$. This type of regularization is often used in the ridge regression. The following figure shows very well how the ridge regression uses the L2 regularization compared to the LASSO regression (Least Absolute Shrinkage and Selection Operator) to find its global minimum. On Figure 3.1 the left

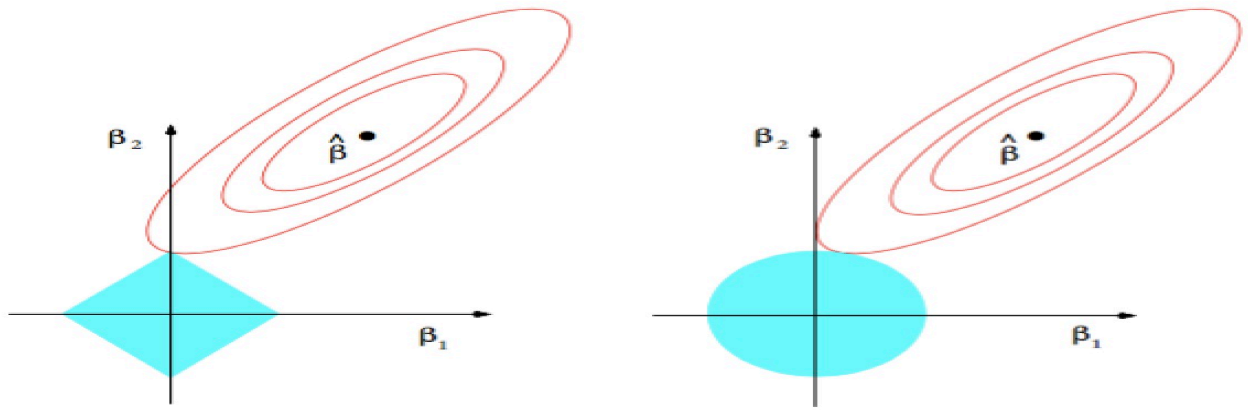



**Figure 3.1:** L1 & L2 regularization comparison[13]

hand side is the L1 regularization for a 2 parameter regression and the red function is the loss function of the regression. The right hand side is the one we will compute which is the L2 regularization. The only difference with our case is that we will perform a regularized optimization on a higher dimension.

[13] Wikimedia (n.a). *File:Regularization.jpg* retrieved from `https://commons.wikimedia.org/wiki/File:Regularization.jpg`

### 3.3 *Volatility Surface*

Comparing the different techniques will rely upon how the implied volatility has been computed. At this stage, we have 5 methods and 1 "raw" implied volatility vector. We have the single vector implied volatility adjusted et non-adjusted to moneyness, the simultaneous penalty-free implied volatility and implied risk-free rate adjusted on moneyness through OLS and SUR and we have the optimization problem with penalty term adjusted on moneyness through OLS and SUR. One way to compare all these implied volatilities is through the implied volatility surface, which is defined as a 3D plot of implied volatility given the log-moneyness and the time to maturity [1] which will be computed through linear interpolation. Another way to compare the methods we used is to apply the replicating strategy and to retrieve the 4 first moments of the error's density function. We will be able to see which method over-hedges and under-hedges as well as the shape of the density. This will enable us to build our data base and move forward to the machine learning section. Note here that the volatility surface is only computed for comparison purposes, the goal here is not to calibrate more exotic options based on the surface.

### 3.4 *Hedging Performance*

In order to stick with the research question, we need to compute the replicating portfolio and then compute the hedging performance of the methods and see whether they over or under hedge. In order to do so, we will need to discretize equation 2.10 which will be done through to the following formula:

$$\Pi_{t+\Delta t} = \Pi_t + \Delta_t(S_{t+\Delta t} - S_t) + r\Delta t(\Pi_t - \Delta_t S_t) \qquad \textbf{Equation 3.9.}$$

Once the replicating portfolio has been computed, we will need to compute the hedging performance through equation 2.11 and the final comparison of all models will be made by comparing the first 4 moments of the computed density of the performance.

### 3.5 *Normality tests*

One way to quantify the accuracy of a method would be to perform a test to see if the data follows a normal distribution and at which confidence level. Two well known tests for this are the Shapiro-Wilk test developed by Samuel Shapiro and Martin Wilk [32] and the Jarque-Bera test developed by Carlos Jarque and Anil Bera [19]. We will use both method to confirm our tests and not to rely only on the results of a single test.

### 3.5.1 Shapiro-Wilk

The Shapiro-Wilk test measures the null hypothesis that a sample follows a normal distribution. Like any test, we need to define the test statistic which is defined as follow according to [32].

$$W = \frac{(\sum_{i=1}^{n} a_i x_{(i)})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \qquad \textbf{Equation 3.10.}$$

Where $x_{(i)}$ is the $i^{th}$ smallest number in the sample and $\bar{x}$ is the sample mean. We will not dig into the details to determine the coefficient $a$ as the formula is just to give us a flavour of how this test works. We are not going into the details as the *shapiro.test* function from the *stats* package in R will test the sample for us. If the p-value of the test is below our confidence level, let's say 5%, it means that we have to reject the normal distribution whereas if the p-value is higher than 5%, the test is not significant and the sample follows a normal distribution at a 5% confidence level.

### 3.5.2 Jarque-Bera

The Jarque-Bera test measures the same thing as the Shapiro-Wilk test but uses other terms to define its test statistic. The latter is defined as follow [19]:

$$JB = \frac{n}{6}(S^2 + \frac{(K-3)^2}{4}) \qquad \textbf{Equation 3.11.}$$

Where $n$ is the number of observations in the sample, $S$ is the sample skewness, which is the 3rd moment of a distribution and $K$ is the sample kurtosis, which is the 4th moment of a distribution. In a normal distribution, the skewness and kurtosis are equal to respectively 0 and 3. Shape-wise, the skewness depicts the assymetry of a distribution, being negative if the density shifts slightly to the right, with a fat left-tail and positive if the density shifts to the left with a fat right-tail. Concerning the kurtosis, it describes the presence of extreme values. A high kurtosis means that there are a lot of outliers and that the data has heavy tails. Lower kurtosis means that there are few outliers and that the data has light tails. Shape-wise, a high kurtosis density would look thinner compared to a normal distribution where as the density of a low kurtosis looks larger, with a lower peak than the normal distribution. We will not linger on the equations of the skewness and the kurtosis as the R function *jarque.bera.test* from the *tseries* package will compute the test for us. The interpretation of the results is the same as the Shapiro-Wilk test, if the p-value is above a certain confidence level, the distribution follows a normal distribution where as if the p-value is below a certain confidence level, the distribution does not follow a normal distribution.

## 3.6 *Variance Analysis (ANOVA) tests*

Another way to see whether a method performs better than another one is to conduct an ANOVA test or an analysis of variance test. This test attempts to make an analysis of the variance within a given set of responses. The main objective of the ANOVA test is to identify the importance of independent variables and to determine how they will impact the dependent variable [24]. The classical ANOVA test compares the mean of only 2 populations of equal variance. In our case, we will need to use the ANOVA for a one-way layout, meaning that we will compare the means of all calibration methods $\mu_1, \mu_2, ..., \mu_k$ where we have $k$ methods to compare. Following [24], we ought to define some parameters. The total number of observations is $n = n_1 + n_2 + ... + n_k$ and $Y_{ij}$ is the response for the $j^{th}$ observation in the $i^{th}$ sample with $i = 1, 2, ..., k$. We will simplify the notations to better understand the following equations. We will define the following:

$$Y_{i\bullet} = \sum_{j=1}^{n_i} Y_{ij} \qquad \overline{Y}_{i\bullet} = \frac{1}{n_i}\sum_{j=1}^{n_i} Y_{ij}$$ **Equation 3.12.**

The null hypothesis $H_0 : \mu_1 = \mu_2 = ... = \mu_k$ will be rejected if:

$$F = \frac{MST}{MSE} > F_\alpha$$ **Equation 3.13.**

Where $F_\alpha$ is the critical value of F for a level set at $\alpha$ and $MST$ and $MSE$ are defined in Equation D.1a and D.1b in the appendix.

This test will come in handy to have a first draft of the answer of the first research question which is to know if there is a difference in the hedging performance in using one method versus another to compute the implied volatility.

# 4 IMPLIED VOLATILITY MODELIZATION: S&P 500

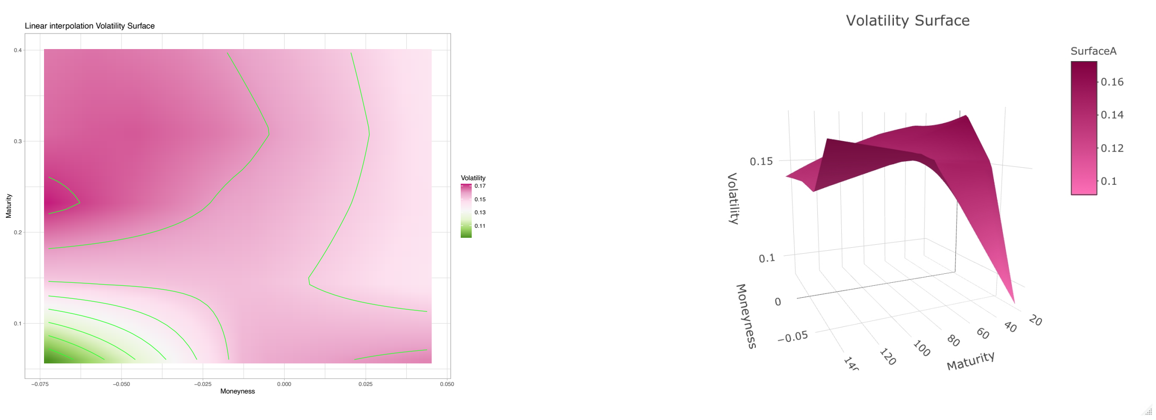
In this chapter, we will apply the methods explained here above on an S&P 500 call option database. The data has been collected on Bloomberg for every trading day from Janurary 04th until March 31st 2016. This data base contains 3050 observations for 61 observation days with 5 different maturities where the longest maturity is May 31st 2016. The S&P 500 does not give dividends and we will hence not rely on any dividend adjustments.

We ought to underline that a longer time span (3 years) is generally required but again, given the circumstances we weren't able to collect further data. We also ought to keep in mind that the objective of this chapter is to compute and visually compare the transformations the implied volatility is going through. This will enable us to have different implied volatilities for the same set of observed options and to see if there is a difference by using one method compared to the other.

## 4.1 *MacBeth & Merville*

In order to complete our dataset to have all required parameters for the BSM model, we needed to retrieve a risk-free rate. The usual risk-free rate that is taken by most of the literature is the 3 months Treasury Bill. We collected the data for each observation day on the St-Louis Federal Reserve web site [34].

Applying the optimization function of equation 3.1 will give us a set of implied volatility for the required time span. Based that, a first volatility surface can be drawn and some observations can be made.

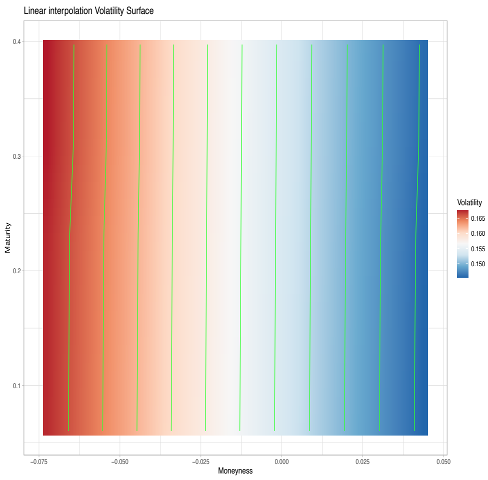



(a) Volatility Surface Contour plot

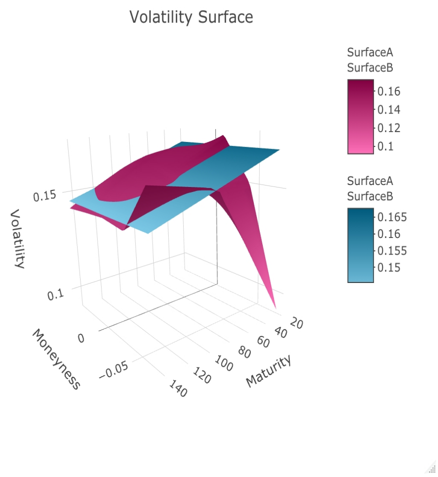
(b) Volatility Surface 3D plot

**Figure 4.1:** Volatility Surface: No moneyness adjustment

Based on these figures, the unconstrained volatility surface is relatively bumpy and the volatility skew is even more observable (see Figures A.1 in the appendix). The point of the other methods will be to adjust this volatility surface and to remove all noisy parts from the raw data.
Our next step is to remove all outliers that could make our data noisier to further predictions. Indeed, before handling those outliers, we ended up with a volatility value of 100% which is almost impossible. After handling the outliers, the range of volatility has shrunk to [0;0.25]. In the second section of this chapter we will see that the regularized optimization deals with that kind of phenomenon. This can be seen on the graphs in the appendix (Figure A.2).
Now that our data has been cleaned, we need to apply equation 3.3 to adjust for moneyness according to MacBeth & Merville. The volatility surfaces look like the following graphs:




**(a)** Volatility Surface: Moneyness Adjusted




**(b)** Volatility Surface 3D Comparison

**Figure 4.2:** Moneyness adjusted volatility surface comparison

The moneyness adjustment has a clear impact on the volatility surface as we can see on figure 4.2b. Indeed, the moneyness adjustment flattens the curve as expected. This will have an important impact on the replicating strategy as it is a rough approximation of the previous volatility surface. As in any regression problem, the point is to find the right balance between accuracy and variance which is known as the bias-variance trade off.

## 4.2 *Bianconi & al.*

In this section we will compare all 4 methods, the unregularized optimization adjusted to moneyness with a classical OLS regression, the unregularized optimization adjusted to moneyness with a seemingly unrelated regression, the regularized optimization adjusted to moneyness with a classical OLS regression and finally, the regularized optimization adjusted to moneyness with a seemingly unrelated regression. We will select, as input parameters for the optimization problems 3.4 and 3.8 a volatility of 50% and a risk-free rate of 0.286% which is the average of the 3 months Treasury Bill rates for January, February and March 2016. This value is also chosen for $R$ in our regularized problem.

For the 2 last methods, we will need to run the optimization problem with an L2 regularization term. In this problem, a question arises as for every regularization problem which is to determine the value of the penalty term $P$. In order to do so, we simulated some random values for $P$ and we computed the related loss function. As expected, the higher the penalty term, the higher the loss function, we will here need to choose a trade off between regularization and minimization of the loss function. To follow Bianconi and al. [3], we chose to plot a graph with different levels of penalty terms given the level of data fit, meaning how well the optimization function fits the data where a penalty level of 0 means that the data fits perfectly and, given the level of regularization, meaning here that a 0 penalty term would lead to the highest regularization. This leads to a trade off selection between data fit and the penalty from drifting away from $R$. We did so by computing the inflection point of the data and this lead to selecting a penalty term equal to $2.75$ as shown in the figure below where the intersection between the green and blue line is the inflection point computed through the *findiplist* function of the *inflection* package in the R software.

We now know the penalty term that we need to use in order to have the best trade off between precision and regularization. With that penalty term we can go on computing the implied volatilities and compare the volatility surfaces of each method. Before comparing those volatility surfaces, we can observe the effect of the penalty term on the outliers of the implied risk-free rate. The regularized term has less extreme outliers, higher than 100% risk-free rate for the unregularized method versus 59% risk-free rate for the regularized method. Of course, having a risk-free rate of 59% is way too high hence the point of handling outliers. The following plots show the above described phenomenon.

Now that all outliers have been handled, we can now compare the volatility surfaces on the following plots. This will have a further impact on the hedging performance of each method. On these graphs we

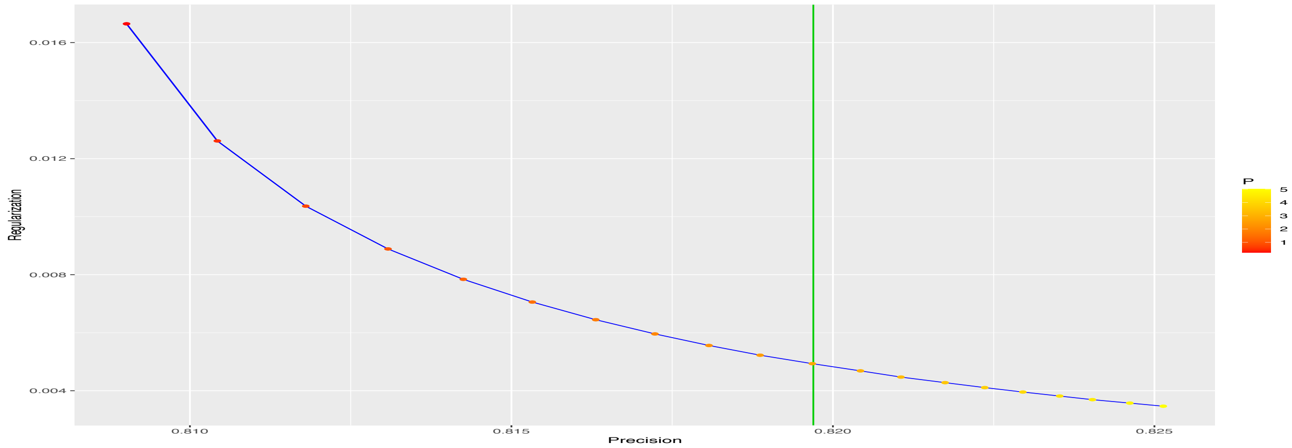
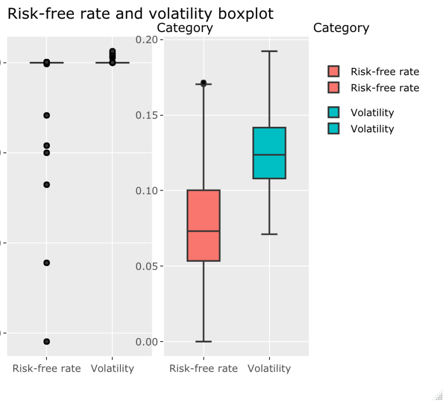
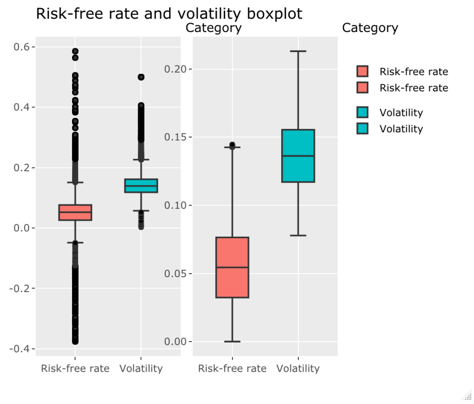



**Figure 4.3:** Penalty term selection



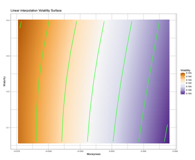

**(a)** Penalty term = 0



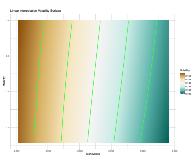

**(b)** Penalty term = 2.75

**Figure 4.4:** Comparison of outliers between penalty-free optimization and regularized optimization

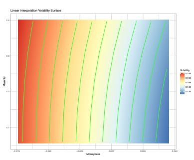
can see that the contour lines are closer to each other for the regularized optimization, meaning that the volatility skew is more important for regularized terms.



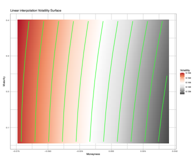
**(a)** OLS: P=0



**(b)** SUR: P=0



**(c)** OLS: P=2.75



**(d)** SUR: P=2.75

**Figure 4.5:** Comparison of the volatility surfaces

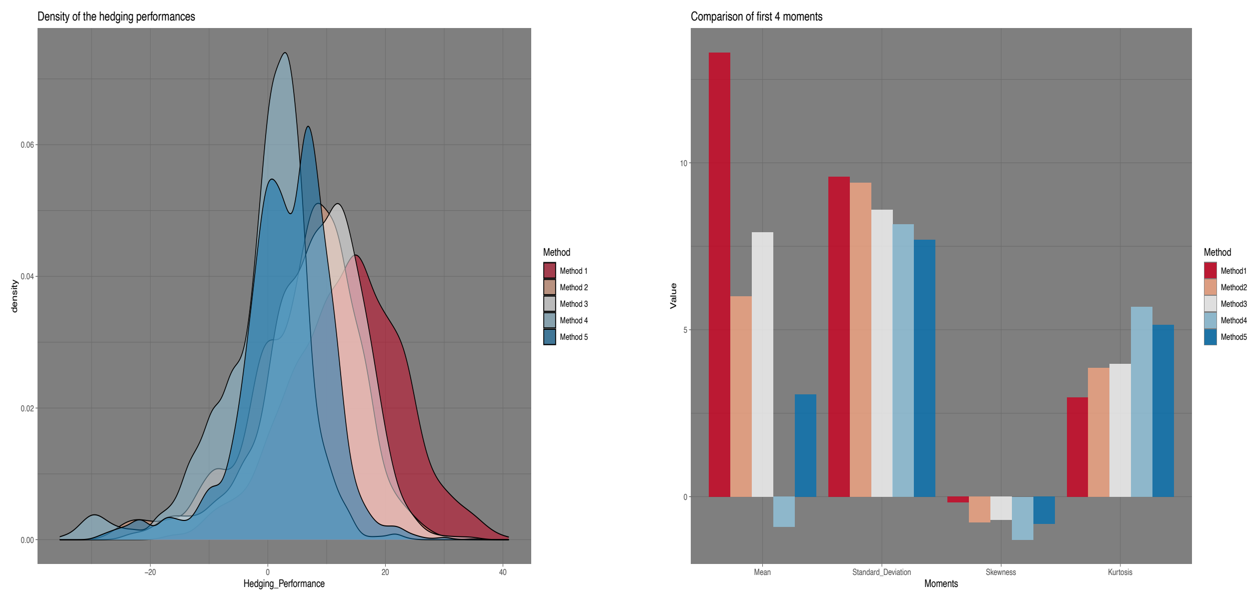
Given those figures, we can already see that the method used for the regression does not really impact the volatility. We will see in the next section that it changes for the hedging performance.

## 4.3 *Hedging performance*

In this section we will mainly apply equation 3.9 and equation 2.19a and rebalance our hedging portfolio daily. We will retrieve the pdf (probability density function) and the moments of the hedging performance, the difference between the payoff of the written option at maturity and the portfolio's value built throughout the option's life also computed at maturity denoted as $\varepsilon_T$ and the vector of hedging performance for all observed options will be denoted $\vec{\varepsilon_T}$. We will have 5 methods to compare: the first one is the method with the adjusted moneyness according to MacBeth and Merville [23]. The following 4 are the methods computed with OLS with no penalty and with penalty (method 2 & 3) and the methods computed with SUR without and with penalty term (method 4 & 5).

We can see that all the methods tend to over-hedge the portfolio as all densities seem to have a positive mean. To quantify these values, we ought to compute the first 4 moments of these densities. The less predictable method would be the one with the highest standard deviation, skewness and kurtosis.

The left figure shows the density comparison of all methods. The right figure shows a bar plot comparing each moment between the methods.




**(a)** S&P500: Density of hedging performance **(b)** S&P500: Moments comparison

**Figure 4.6:** S&P500: Hedging Performance comparison

At a first glance, we can confirm that all methods over-hedge our portfolio except method 4 with a negative mean. A clear question arises from these observations: Why do all methods have a tendency to over hedge the portfolio? This could be answered through higher order derivative of the call option and other greeks, namely the gamma and theta effect. The theta effect is known as the time decay effect and its impacts are that the evolution of a call option's price is decreasing with respect to the time to maturity. This means that an option today will always be worth less than an option yesterday. Regarding the gamma of the call option, it represents the speed at which the delta is increasing or decreasing. It can be seen as the acceleration of the underlying's price. [38] We will dig into the relationship between the gamma effect or the theta effect and the hedging performance in section 5.2.

The values of the moments can be found in the following table.

| **Method** | **Moments** | | | |
|---|---|---|---|---|
| | Mean | Standard Deviation | Skewness | Kurtosis |
| Method 1 | 13.301 | 9.585 | -0.173 | 2.970 |
| Method 2 | 5.999 | 9.413 | -0.766 | 3.854 |
| Method 3 | 7.927 | 8.6 | -0.698 | 3.982 |
| Method 4 | -0.9 | 8.166 | -1.288 | 5.681 |
| Method 5 | 3.062 | 7.692 | -0.816 | 5.152 |

**Table 4.1:** S&P500: 4 moments of $\vec{\varepsilon_T}$ for each method

The following table depicts the Shapiro-Wilk and Jarque-Bera tests for normality. The higher the p-value, the closer the sample is to a normal distribution. According to the following table, no method leads to a normally distributed hedging performance. We will hence use the first 4 moments as factors in the machine learning part as these tests are not conclusive.

| **Test** | **Method** | | | | |
|---|---|---|---|---|---|
| | Method 1 | Method 2 | Method 3 | Method 4 | Method 5 |
| Shapiro-Wilk | 8.226498e-06 | 1.666539e-26 | 1.804637e-24 | 3.491919e-39 | 4.004472e-30 |
| Jarque-Bera | 0.0005247184 | 0 | 0 | 0 | 0 |

**Table 4.2:** S&P500: Normality tests for $\vec{\varepsilon_T}$ for each method

Based on these graphs and these numbers, we observe that 4 methods out of the 5 tend to over-hedge the portfolio. This is mainly due to a larger $\Gamma$ effect over the $\Theta$ effect like we described in section 2.7. Going on with our methodology, we will apply section 3.6 on our data set. This enables us to see if there is a significant difference in using a method or another one to compute the hedging performance. We will perform an ANOVA test to see if there is a statistical difference between the means of each method. We will reject the null hypothesis according to which the means between the groups are identical if the p-value of the test falls below a certain threshold. We performed the ANOVA test using the *aov* function from the *stats* package and we ended up with a p-value smaller than $2e^{-16}$ meaning that we can reject with a high level of confidence $(1 - 2e^{-16})$ the null hypothesis and say that there is a significant difference by using a method versus another one. The following figure depicts the boxplots of the hedging performance given a certain method.

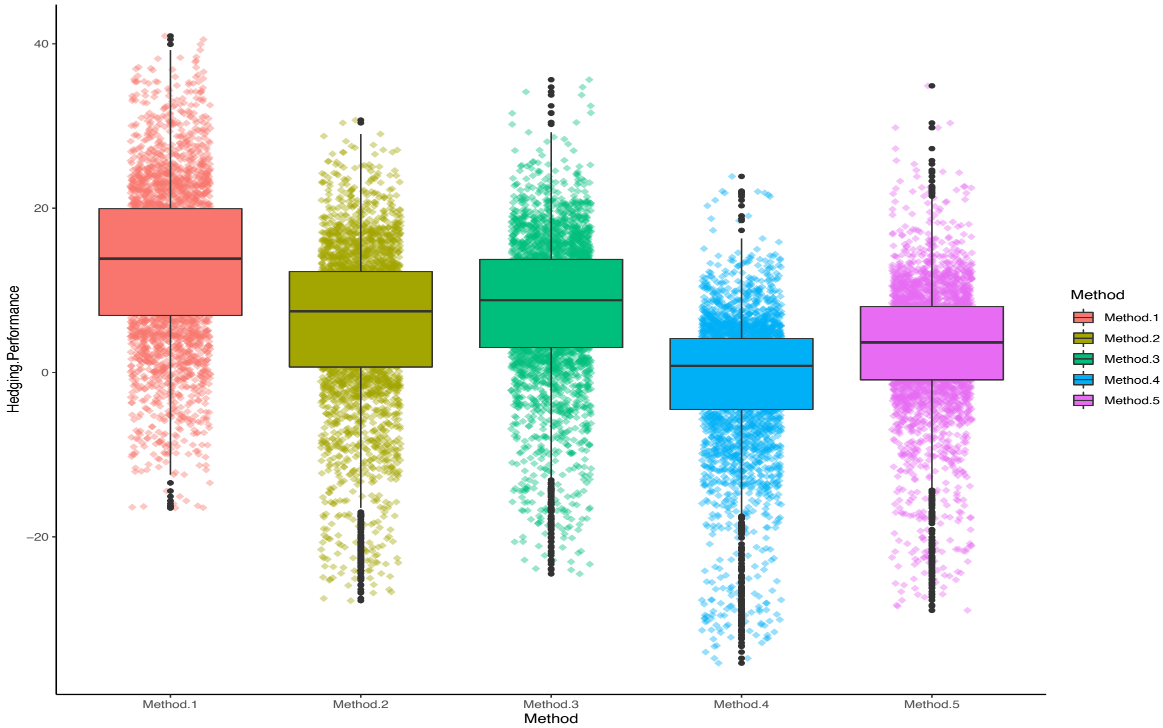



**Figure 4.7:** Anova test for each method on the S&P500 data set

In light of these numbers and graphs, we can already propose a first answer to the sub question in the introduction. *Does a specific implied volatility model change the hedging performance in terms of payoff replication?* The answer to that would be yes according to the last test we performed. The answer is not entirely complete though. Indeed, we need to adjust for other concepts and parameters to have a full answer to that question but that goes beyond the scope of this thesis.

# 5 IMPLIED VOLATILITY MODELIZATION: APPLE

In this chapter, we will perform the same tests as in chapter 4 but for the Apple stock whose data base has been built the same way as the S&P 500 data base with 3050 observations from January 4th until March 31st, 2016. The difference with the S&P500 here is that the longest maturity goes through October 22nd 2016. Another difference is that Apple gives out dividends but we chose to ignore dividends payout as it would break one of BSM's assumptions and that adapting all formulas to dividends has been done many times in the literature.

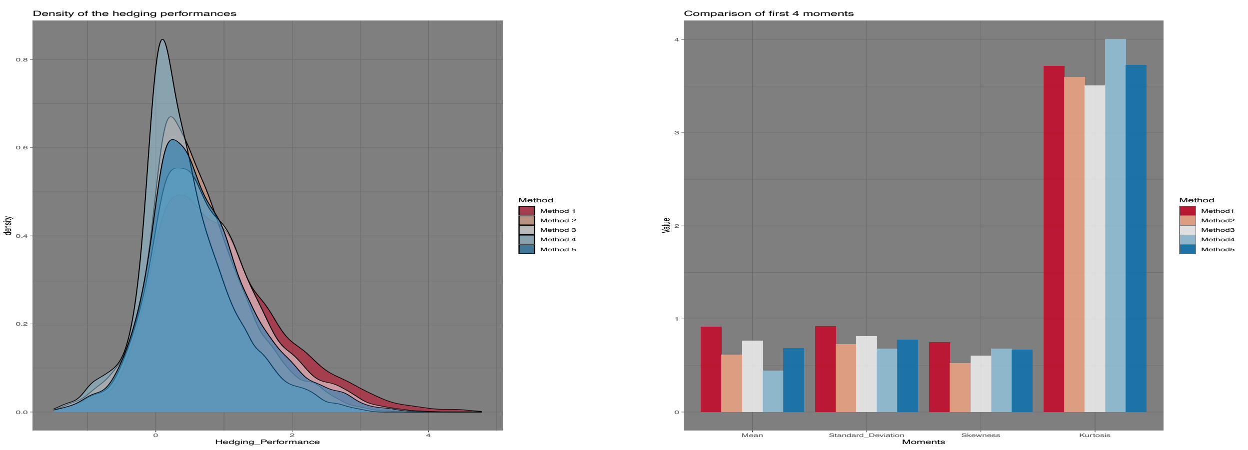
Given the fact that the methodology in this section is the same as in the previous one, we will skip it and jump right to the results. The graphs of the previous section adapted to the Apple data base can be found in the appendix. Nevertheless, we need to outline that we also used the same penalty term in the regularization optimization of 2.75 for a matter of comparison.

Based on the graphs of the appendix, the same observations as in the previous section can be made.

## 5.1 *Hedging performance*

This section will show us if the hedging performance is better than in the S&P 500 and why. The main reason on why the hedging performance could be better than the previous data set is because of the $\Gamma$ and $\Theta$ effects that were described in section 2.7.

We can see on the following graphs that all methods tend to over-hedge the portfolio as all densities have a positive mean. Compared to the S&P500's density functions, Apple has a lower mean, certainly due to a bigger $\Theta$ effect but this will be tackled in the following subsection.




**(a)** Apple: Density of hedging performance

**(b)** Apple: Moments comparison

**Figure 5.1:** Apple: Hedging Performance comparison

The values of the first 4 moments can be found in the following table:

| Method | Moments | | | |
|---|---|---|---|---|
| | Mean | Standard Deviation | Skewness | Kurtosis |
| Method 1 | 0.913 | 0.922 | 0.751 | 3.718 |
| Method 2 | 0.612 | 0.727 | 0.526 | 3.599 |
| Method 3 | 0.765 | 0.810 | 0.603 | 3.508 |
| Method 4 | 0.440 | 0.679 | 0.681 | 4.004 |
| Method 5 | 0.684 | 0.776 | 0.671 | 3.728 |

**Table 5.1:** Apple: 4 moments of $\vec{\varepsilon_T}$ for each method

Based on these observations, we can say that for the time span considered, the delta-hedging strategy has been more accurate for Apple than for the S&P 500 as the means of the density functions are closer to 0.

We will now run the normality tests to see if the data follows a normal distribution. The p-values can be found in the following table for the Shapiro-Wilk test and the Jarque-Bera test.

| Test | Method | | | | |
|---|---|---|---|---|---|
| | Method 1 | Method 2 | Method 3 | Method 4 | Method 5 |
| Shapiro-Wilk | 3.726781e-26 | 1.374877e-20 | 6.670895e-22 | 6.068037e-27 | 3.688523e-24 |
| Jarque-Bera | 0 | 0 | 0 | 0 | 0 |

**Table 5.2:** S&P500: Normality tests for $\vec{\varepsilon_T}$ for each method

Based on these numbers, we can say with confidence that no method would lead to a normal distribution of $\vec{\varepsilon_T}$.

We now need to perform an ANOVA test to see if a difference between the means of each method exists. Performing this test leads us to a p-value lower than $2e^{-16}$ meaning that we can reject with a high level of confidence the null hypothesis stating that there is a significant difference by using a method versus another one. The following figure depicts the boxplots of the hedging performance given a certain method for the Apple data set.

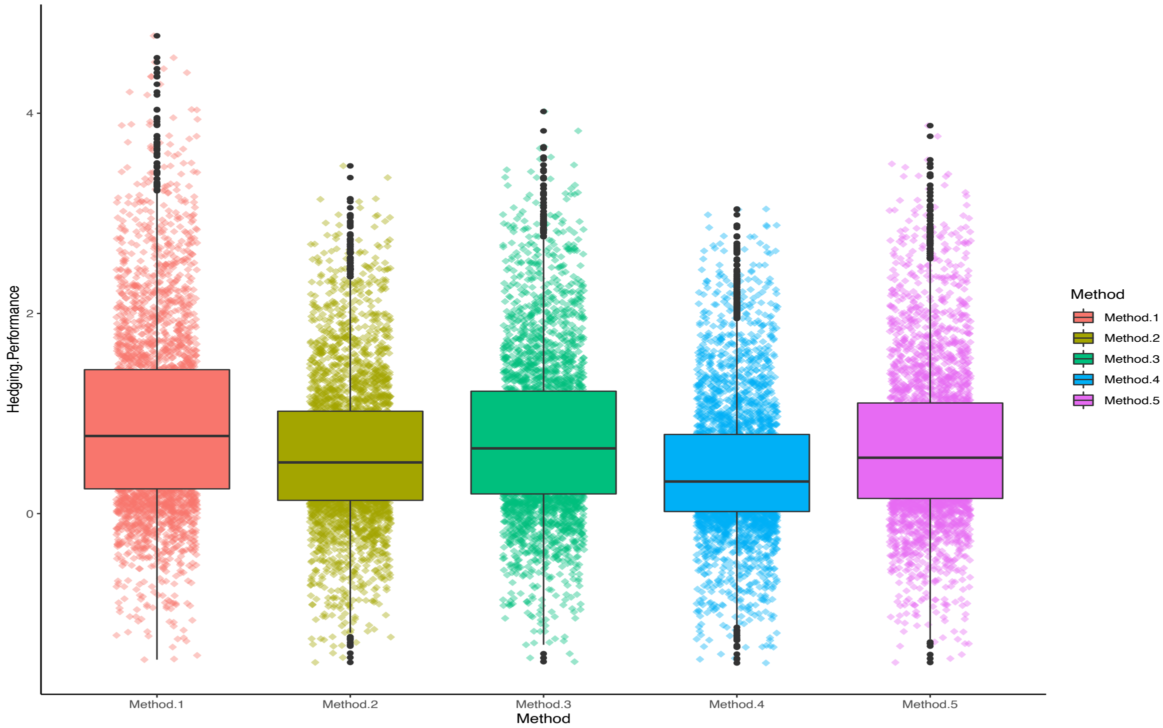


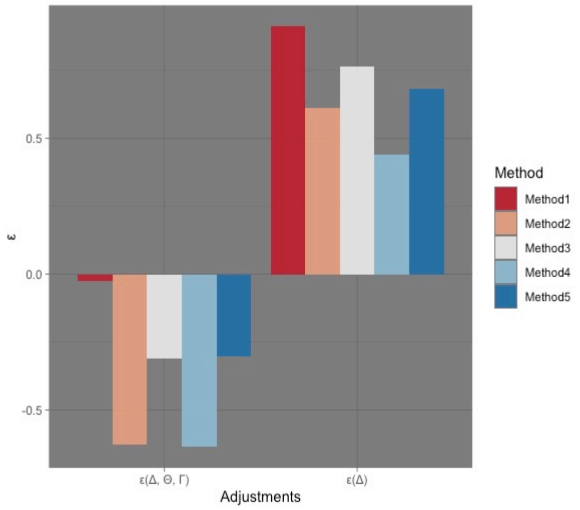

**Figure 5.2:** Anova test for each method on the Apple data set

## 5.2 *Apple vs S&P 500*

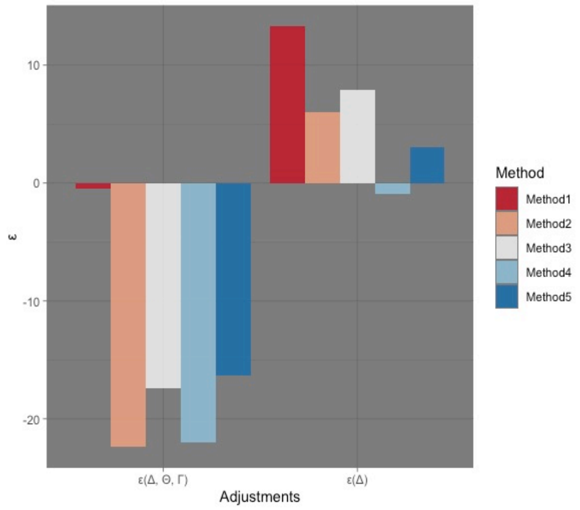
Now that all tests have been run and all graphs drawn, it is interesting to compare both results before moving on with the machine learning section. Indeed, in section 2.7 we outlined the fact that the replicating portfolio could be driven by something else than the only $\Delta$ of the BSM model. We saw that this portfolio also depends on the $\Gamma$ and the $\Theta$ of the BSM model. It is clear that the hedging performance of the replicating portfolio on S&P 500 quoted call options endures a stronger $\Gamma$ effect than the replicating portfolio on Apple quoted call options. In order to quantify these effects we decided to run the replicating portfolio by applying equation 2.20 accounting for $\Delta$, $\Gamma$ and $\Theta$. This leads us to compare two measures of $\mathbb{E}[\varepsilon_T]$ that we will denote as $\mathbb{E}[\varepsilon_T(\Delta)]$ which is the hedging error computed through the replicating portfolio accounting only for $\Delta$ as in equation 2.19a and $\mathbb{E}[\varepsilon_T(\Delta, \Theta, \Gamma)]$ which is the hedging error computed through the replicating portfolio accounting for $\Delta$, $\Theta$ and $\Gamma$ as in equation 2.20.

To clear everything up, for the $\Delta$ (resp. $\Delta$, $\Theta$, $\Gamma$) hedging strategy we first computed the replicating portfolio using equations 2.19a and 3.9 (resp. equation 2.20). Once the different hedging portfolios have been computed, we will afterwards need to compute the hedging errors for each observed option at the corresponding maturity $\varepsilon_{Ti}(\Delta)$ (resp. $\varepsilon_{Ti}(\Delta, \Theta, \Gamma)$) using equation 2.11. Once all hedging errors have been computed, we will need to compute $\mathbb{E}[\varepsilon_T(\Delta)]$ (resp. $\mathbb{E}[\varepsilon_T(\Delta, \Theta, \Gamma)]$) through equation 2.12.

By doing so for all 5 methods and for both securities we ended up with the following graph:




(a) Apple




(b) S&P500

**Figure 5.3:** $\mathbb{E}[\varepsilon_T(\Delta)]$ & $\mathbb{E}[\varepsilon_T(\Delta, \Theta, \Gamma)]$ for each method.

Based on both graphs we can see that the effect of $\Gamma$ and $\Theta$ is larger on the replicating strategy on the S&P500 than on Apple. Indeed, the 5 left bars of the plot being averaged hedging error $\mathbb{E}[\varepsilon_T(\Delta, \Theta, \Gamma)]$ we see that all bars are negative meaning that $\mathbb{E}[\varepsilon_T(\Delta)]$ is over estimating the $\Gamma$ effect and underestimating the $\Theta$ effect (we remind that a higher hedging error is mainly due to a large $\Gamma$ and that a lower hedging error is mainly due to a large negative $\Theta$). For the right graph, the S&P 500 plot also seems to overestimate the $\Gamma$ effect as the left bars are all negative. Nevertheless, for the S&P500, on one hand, method 1 seems to handle well the errors due to the $\Gamma$ and the $\Theta$ as $\mathbb{E}[\varepsilon_T(\Delta, \Theta, \Gamma)]$ is close to 0. On the other hand, method 4 estimates well the errors due to $\Gamma$ and $\Theta$ as the $\mathbb{E}[\varepsilon_T(\Delta)]$ is close to 0.

Now that our tests and observations have been made, we will move on with the machine learning section. In this section, we will first run some tests such as correlation tests and we will manipulate our data base. We will build the latter based on all methods and all parameters/inputs we used in the previous section.

# 6 MACHINE LEARNING

In this section, we will use the data we computed and analyzed in chapters 4 and 5 and we will run a machine learning algorithm on it to finally predict the hedging error $\varepsilon_{Ti}(\Delta)$. We will divide this section in 2 main sections starting with the methodology and explaining how we will run this algorithm and the second section will be to actually apply the methodology on the data set.

## 6.1 *Methodology*

In this section, we will need to outline the steps we will follow in order to run a successful machine learning algorithm. Before any machine learning algorithm, we will need to explain the data set and perform an exploratory analysis by running some tests on the data set. The data set is built as follows:

| Column label | Description | Type | Levels (if factor) |
|---|---|---|---|
| *Observation.Date* | Date of data collection | Date | NA |
| *Maturity* | Maturity of observed call | Date | NA |
| *TtM* | Difference between maturity and observed date | numeric | NA |
| *Bid* | Bid price of the call option | numeric | NA |
| *Ask* | Ask price of the call option | numeric | NA |
| *Mean.Bid.Ask* | Average between bid and ask prices | numeric | NA |
| *Strike* | Agreed upon strike price ($K$) | numeric | NA |
| *Price* | Underlying's price ($S_t$) | numeric | NA |
| *RF.Rate* | Risk free rate ($r$ or $r^*$) | numeric | NA |
| *Moneyness* | Moneyness as in equation 3.2 | numeric | NA |
| *Volume* | Trading volume on that day | numeric | NA |
| *Volatility* | Implied volatility ($\sigma^*$) | numeric | NA |
| *BSM* | BSM computed call price ($C_t$) | numeric | NA |
| *Method* | Method used to compute the implied volatility | factor | Method 1 to 5 |
| *Underlying* | Underlying used | factor | AAPL & SPY |
| *Hedging.Performance* | Hedging error, $\varepsilon_{Ti}(\Delta)$ | numeric | NA |

**Table 6.1:** Data Base description

We divided this data set to distinguish the independent variables being columns 1 through 15 and the last column being the dependent variable we are trying to predict. In this section we will refer to the column labels which are described the same way as in the attached R code.

### 6.1.1 Descriptive analysis

We will here describe the steps taken before running the algorithm. Indeed, in order to avoid over fitting, we ought to take out some data. In order to do so we firstly computed the correlations between all numerical variables (including the independent variable) and we secondly ran an ANOVA test to see if there is any difference in the average of the factor variables.

We will compute the pearson correlation coefficient as in [27] in the appendix (equation D.2). We will remove all independent but one variable with a high correlation (0.95 threshold). This means that if the $\rho(X_1, X_2) = 0.97$, that $\rho(X_1, X_3) = 0.98$ and that $\rho(X_2, X_3) = 0.96$, we will remove one of the following independent variable set: ($X_1$, $X_2$), ($X_1$, $X_3$) or ($X_2$, $X_3$). We will set a threshold of 0.95 for our correlation test selection.

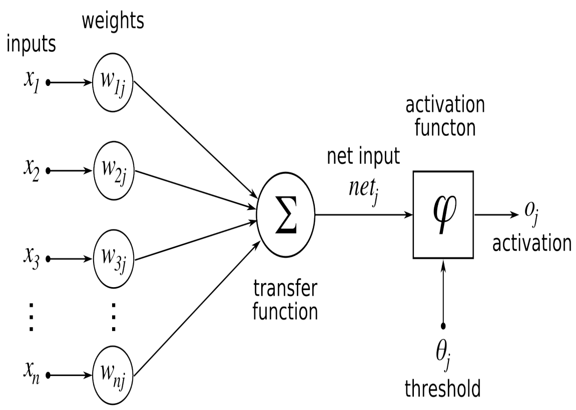
We will also run an ANOVA test as described in section 3.6 and we will remove the categorical variables if the ANOVA test cannot be rejected at a 5% confidence level. These tests will be done on the *Underlying* and the *Method* independent variables.

Now that we have removed all variables that could be prone to over fit our dependent variable, we need to scale all numerical variables in order to have variables at the same level. This is absolutely mandatory because it enables the algorithm to compare apples with apples and not apples with pears.

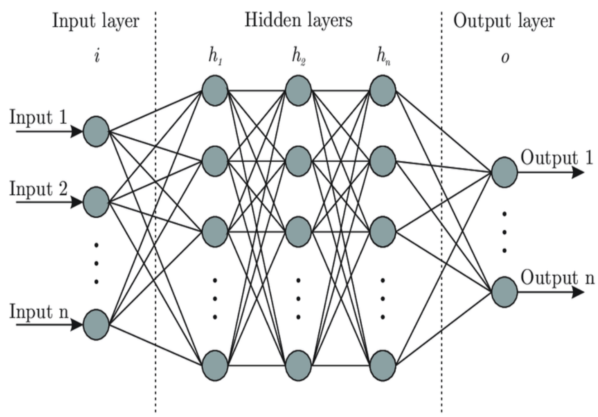
Analogously to the correlation test for numerical variables and ANOVA test for categorical variables, we now need to handle the values of categorical variables. In order to avoid values in the level of our categorical variables and because of the fact that the algorithm only takes binary variables as categorical input, we need to perform a one-hot encoding process as explained in [10]. This means that we will convert them all to binary variables.

### 6.1.2 Learning & Predictions

The first section being all about data preparation, we will here run the algorithm and predict our data. In any machine learning project we have to choose which algorithm to run and the choice panel is rather large. Choosing from random forests, to support vector machines or logistic regressions, we finally chose the artificial neural network (ANN), an algorithm whose process mimics that of the human brain. We chose this algorithm compared to others because it is extremely powerful as it can learn very complex relationships which is the case in our data. Moreover, it can also reduce our need for feature engineering. The ANN algorithm is described by [11] as following: " [...] an artificial neural network, in which each neuron has a set of inputs, each of which is given a specific weight. The neuron computes a function on these weighted inputs. A linear neuron takes a linear combination of weighted inputs and applies an activation function (sigmoid, tanh, and so on) on the aggregated sum. [...] The network feeds the weighted sum of the input into the logistic function (in case of a sigmoid function). The logistic function returns a value between 0 and 1 based on the set threshold."[14]
An ANN can be represented by the following figures:




**(a)** ANN: Activation Function[15]




**(b)** ANN: Hidden Layers[16]

**Figure 6.1:** ANN representations

[14]Dangeti, P. (2017). *Statistics for Machine Learning*. [ebook] Packt Publishing.
[15]Bre, F. Gimenez, J. Fachinotti, V. (2017). Prediction of wind pressure coefficients on building surfaces using Artificial Neural Networks. *Energy and Buildings*. 158.
[16]Wikibooks (n.a.), *Artificial Neural Networks/Activation Functions* retrieved from `https://en.wikibooks.org/wiki/Artificial_Neural_Networks/Activation_Functions#Activation_Functions`

By comparison with a 2 dimensional linear regression, where we have to select the intercept and the slope that fits best our data, we need to find the right weights in our neurons that will fit best our data. In order to find the weights of the neurons, the neural network needs a set of input parameters that can be selected arbitrary. These inputs are, for the *neuralnet* function from the package *neuralnet*, the number of hidden layers and the number of nodes in each layer, the threshold of the activation function and the number of maximum steps the network can take. Given the nature of the algorithm, the higher the threshold, the lower the accuracy of prediction but the lower the time consumption. These input parameters are called hyper parameters described as follow:

- Hidden Layers: the *neuralnet* function in the R software can take as input a vector of hidden layers where the length of the vector is the number of layers in our network (*e.g.* 3 in figure 6.1b) and each values within the vector is the number of nodes in the layer. We will then use a maximum of 1 layer (because of high time consumption of more layers) ranging from 1 to 5 nodes. This set will be denoted $H_i$.
- Threshold: we will here use a range of values going from 0.05 to 0.1 with 0.01 steps. This set will be denoted $T_j$.
- Hyper parameters set: The sets that will be tested here take the following form: $HT_{ij} = (H_i, T_j)$.

In order to find the right values for these hyper parameters, we will run a hyper parameter optimization where we will run $i * j$ times the algorithm with different hyper parameters and select the parameters for which the algorithm is the most accurate. According to [18] to start our optimization, we will need a set of hyper parameters from which we will have to choose the right combination to find the right minimum in a loss function or maximum for an accuracy function. As all machine learning project requires, we will need to divide our data in a training set, on which we will train our algorithm and in a testing set on which our algorithm will predict the dependent variable on unseen independent variables and on which we will compute our accuracy/loss function. Given an artificial neural network algorithm, $ANN$ in which the inputs are a set of hyper parameters $H_i$ and $T_j$, and given a loss function

$\mathcal{L}$ and an accuracy function $\mathcal{A}$, the hyper parameters $(H_i^*, T_j^*)$ that will optimize $\mathcal{A}$ or $\mathcal{L}$ are given by:

$$(H_i^*, T_j^*) = Arg \min_{H_i, T_j} \mathcal{L}\left[ANN(H_i, T_j)\right] \quad \forall i, j \qquad \textbf{Equation 6.1a.}$$

$$(H_i^*, T_j^*) = Arg \max_{H_i, T_j} \mathcal{A}\left[ANN(H_i, T_j)\right] \quad \forall i, j \qquad \textbf{Equation 6.1b.}$$

Where $\mathcal{A}$ is described in the following section (equation 6.2) and $\mathcal{L}$ is either the RMSE or the MSE whose equations are given in the appendix (Equations D.3).

Once we found our hyper parameters, we need to compute $\mathcal{A}$ and $\mathcal{L}$, by training our data set and then predict the dependent variable on our testing set. We will then have an accuracy/loss for our trained algorithm but this is not really accurate as we always used the same training and testing set. One way to overcome this lack of robustness is to use a *k-fold* cross validation which is explained by [11] where "trained data was divided into equal parts and training performed on all the other parts of the data except one part, on which performance would be evaluated. This process repeated as many parts user has chosen".[17] Here we will use a 10-fold cross validation method, meaning that we will divide the training data in 10 equal parts and that we will train our algorithm 10 times over 90% of the training data and predict on the 10% left. In a 10-fold cross validation, we will end up with 10 values for $\mathcal{A}$ and 10 values for $\mathcal{L}$. We will finally compare the average of all 10 $\mathcal{A}$ and $\mathcal{L}$, $\mathbb{E}[\mathcal{A}]$ and $\mathbb{E}[\mathcal{L}]$. This will improve the robustness of our algorithm but comes with a high computational price. Figure C.1 in the appendix gives us a better understanding of a 10-fold cross validation.

Now that we know how to run our algorithm, we will show how we are going to compute the accuracy of our data. This will depend on whether we will use a loss function like the RMSE or the MSE or if we will use an accuracy function. This will also depend on what type of independent variable we will predict, numerical or categorical. If on one hand, we wish to predict continuous values, we will use a loss function, the Root Mean Square Error (RMSE) and the Mean Square Error (MSE) whose equations can be found in the appendix (equation D.3). We will hence select the algorithm with the smallest RMSE and MSE. If on the other hand, we wish to predict categorical values, we will build a confusion matrix which is defined by [25] as a "performance measurement for machine learning classification". Let's use an example. Imagine that the goal of a machine learning algorithm is to classify the output into

[17] Dangeti, P. (2017). *Statistics for Machine Learning.* [ebook] Packt Publishing.

2 categories, $\alpha$ and $\beta$. 4 outcomes could here be possible: the output actually belongs to $\alpha$ ($a$ times) or $\beta$ ($b$ times), the output has been predicted belonging to $\alpha$ but actually belongs to $\beta$ ($c$ times) and the other way around, the output has been predicted belonging to $\beta$ but actually belongs to $\alpha$ ($d$ times). This can be represented as a matrix where the diagonal are the true predictions.

| Actual | Predicted | |
|---|---|---|
| | $\boldsymbol{\alpha}$ | $\boldsymbol{\beta}$ |
| $\boldsymbol{\alpha}$ | $a$ | $d$ |
| $\boldsymbol{\beta}$ | $c$ | $b$ |

**Table 6.2:** Confusion matrix

In that case, the precision will be computed the following way:

$$\mathcal{A} = \frac{a+b}{a+b+c+d} \qquad \textbf{Equation 6.2.}$$

Where the denominator is the size of the sample $n$.

We have now explained how we will handle and predict our hedging error based on a set of independent variables. The next step will be to apply this methodology on our data set.

## 6.2 *Hedging performance prediction*

We will now apply the methodology developed in the first section of this chapter to our data set. We will hence first conduct a descriptive analysis followed by the learning part and ending with the predictions and accuracy.

First, we computed all correlations from our numerical data base and found out that a few independent variables were correlated with each other. Indeed, the variables *Bid*, *Ask* and *Mean.Bid.Ask* are highly correlated which makes sense as the latter is a function of the 2 first ones. We hence decided to remove the 2 first variables. By using common sense, we also removed the variables *Observation.Date* and *Maturity* as they are used to compute the variable *TtM*. Another point that might be worth outlining is the fact that the variable *Strike* and the variable *Price* were also highly correlated but given the nature of the variables, we chose nevertheless to keep them in our data base. This was also true for the variable

*BSM* and the variables *Bid* and *Ask* where the nature of the variable made us keep it in our data base. All these decisions were made based on a correlation matrix that can be found in the appendix (Table E.1). We can also have a flavour of our correlation matrix through our correlogram in the appendix (Figure C.2)

Regarding the ANOVA tests, both tests had a p-value below the 5% confidence interval meaning that there is a statistical difference between the means of each group in the *Hedging.Performance* variable. Boxplots of those tests can be found in the appendix (Figure C.3a & C.3b). Once the ANOVA tests were performed we scaled the data using equation D.4 and we one-hot encoded our 2 categorical variables (*Underlying* and *Method*), expanding the number of variables from 2 to 7 (2 levels for the first variable and 5 levels for the second one).

Our data is now ready to be trained as all tests have been performed as well as all necessary data manipulation. We will now train our algorithm and run our hyper parameter optimization. To do so, we were supposed to run a 10-fold cross validation but we encountered a slight problem. The problem is that it is highly computational expensive to train the algorithm 10 times for only 1 set of hyper parameters and it is hence highly time consuming. We decided then to train the algorithm once per hyper parameter set and then select the best hyper parameters for our loss/accuracy function. Once we found those hyper parameters, we then performed a 10-fold cross validation. Because of the time consumption of the algorithm, we tried to reduce the computation time by categorizing the dependent variable *Hedging.Performance* and transforming it into a new dependent variable *Hedge.Type* which is now a 2 leveled categorical variable: Over.Hedged & Under.Hedged. "Over.Hedged" for all *Hedging.Performance* positive values and "Under.Hedged" for all negative ones.

For an artificial neural network, the hyper parameters that need to be found are the number of hidden layers of the network, the threshold of the error function and the number of maximal steps the network can take to train. Obviously, the last parameter needs to be higher if the algorithm does not converge as the training is not complete. This parameter is more a limit than a real hyper parameter. We will then focus on the 2 first for our optimization. Knowing which hyper parameters sets to use, we will now run the ANN network 30 times in order to find $(H_i^*, T_j^*)$. Given the fact that we are trying to predict a categorical variable, we will need to use 6.1b as an optimization problem and use 6.2 for $\mathcal{A}$. The main challenge here was that this optimization took a lot of time and this is why we needed our 2nd selection criteria to be time besides the accuracy of the prediction. We hence needed to select the hyper parameter that had the best trade off between the accuracy of the model and its time consumption. This led to build the tables that can be found in the appendix (Table E.2 and Table E.3). Based on both tables, we selected the pair $(2, 0.1)$ as it led to the best time-precision trade off. Getting these 2 hyper parameters was no easy task. Indeed, we faced 2 additional challenges, the first being the fact that the algorithm would not converge to a maximum given a high number of nodes in the hidden layer and a low threshold. This has been denoted as "NC" in Tables E.2 and E.3. The second challenge was that training the algorithm on the whole data set was way too time expensive which is why we narrowed the training set down to 25% only for the hyper parameter optimization.

Now that we have our hyper parameters, we are going to perform a 10-fold cross validation to compute $\mathbb{E}[\mathcal{A}]$ and $\mathbb{E}[\mathcal{L}]$. First of all, we will focus on $\mathbb{E}[\mathcal{A}]$ trying to predict categorical variables. We will secondly focus on $\mathbb{E}[\mathcal{L}]$ trying to predict the true value of the hedging performance. Running a first 10-fold cross validation, we ended up with $\mathbb{E}[\mathcal{A}] = 84.18\%$. This means that our neural networks predicts correctly the *Hedge.Type* 84.18% of the time. A table of the accuracy of this cross-validation can be found in the appendix. (Table E.4)

We will now go on with the prediction of the numerical value of the hedging error. We will run again a 10-fold cross validation but here we will compute $\mathbb{E}[\mathcal{L}]$ where $\mathcal{L}$ is the MSE and the RMSE. Doing so we ended up with a MSE of 26.16 and a RMSE of 5.11. A table of the RMSE and MSE can be found in the appendix. (Table E.5)

Now that we have trained and tested our data in an in-sample data set and that we know which inputs for our algorithm to use, we will perform an out-of-sample prediction. We will run our algorithm on unseen data and compute first the categorical variable *Hedge.Type*. We will afterwards compute the continuous variable *Hedging.Performance*. The fact that the prediction is out-of-sample just means that the prediction has been performed on unseen data and not that we don't know the actual value of the prediction. Because we know the value of the prediction, we can compute our accuracy, MSE and RMSE to see how the algorithm performed. We can see on the following table that the algorithm did perform as expected.

| Predictor Type | Features | | | |
|---|---|---|---|---|
| | **Accuracy** | **MSE** | **RMSE** | **TtR** |
| **Categorical** | 84.28% | / | / | 2:25'12" |
| **Continuous** | / | 26.83 | 5.18 | 2:03'00" |

**Table 6.3:** Out-of-sample prediction

Where *TtR* is the time to run the algorithm. A representation of the 2 neural networks with their weights can be found in the appendix (figure C.4).

In light of these numbers, we can answer our second research question. *Can non linear machine learning techniques predict out-of-sample hedging errors in terms of payoff replication?*. The answer to that is yes but to a certain extent. Indeed, because both algorithms are not 100% accurate, the answer to the research question is not yes to 100%. There are also a couple of limitations to this question, which will be discussed in the conclusion.

# 7 ETHICS IN QUANTITATIVE FINANCE

In this section, we will discuss about the role and the importance of ethics in quantitative finance. This section digresses from the main subject of this thesis but the importance of ethics by future financial engineers cannot be ignored. We will develop this section relying on the work of [45].

The role of quantitative finance has grown since the growth in complexity of financial instruments and alongside it, the need of a strong mathematical approach to price those instruments. "The role of a quantitative financial analyst or "quant" is to use mathematical techniques, computing technology and data manipulation to solve complex problems associated with asset pricing, trading and risk control in financial services."[18] Because they specialize in complex mathematical models and deep understanding of the financial market, whose complexity is beyond the understanding of company executives, the ethical education for those specialists is of paramount importance. Most of the time, the responsibility of the quants is to communicate the inherent risk of the pricing model to the executive and leave all ethical decision to them. This leaves the quants to work in a quasi ethical-free environment. The latter is the reason why we will dig a little bit deeper in the ethical environment and awareness of quantitative analysts.

The roles of ethics in quantitative finance are numerous, and the first role [45] has outlined is a fiduciary role. He shows that quantitative analysts often focus on the legal aspect of a financial product and the risk associated to it and often overlook the suitability of the product for the client. This led to numerous litigation such as Procter & Gamble VS Bankers Trust because the quantitative analysts of Bankers Trust modelled a complex and unsuited floating-rate swap structure yielding to ethical and trust issues.

Another role of ethics in quantitative finance is that quantitative analysts need to be aware of the ethical interpretation of their mathematical models. This concerns mainly derivative portfolio valuations for earnings reporting. The quant team values the portfolio of derivatives and the accountants book the value of that portfolio. The role of the quant team here is very important and subtle as it allows the company to meet earnings requirement. The pricing of the portfolio goes beyond the scope of the

[18]West, J. (2012). Money Mathematics: Examining Ethics Education in Quantitative Finance. *Journal of Business Ethics*, 9, 25-40.

accountants and subtleties of the portfolio valuation would lead to over or under stating earnings. The gap in the quant team's integrity and their models leads to misinterpreting model subtleties as earning manipulation and the literature as shown that earning manipulation leads to investor's distrust in the company and often a resulting drop in the stock price. In light of these observations, the quant team would want to build an ethically superior model but would trade ethics for accuracy. This has been shown by [5] as the trade-off between *fairness* and *efficiency*.

Having a team of highly qualified quantitative analysts has been and will always be a competitive advantage for financial institutions but today, the lack of ethical awareness of these analysts is still too high. This section has solely been written for awareness purposes and broader understanding of the implication of quantitative finance in real life ethics, which is in my opinion as important as excellence and thoroughness in the education of a quantitative analyst.

## 8 CONCLUSION

In this thesis, we provided a thorough work on the implied volatility modelling under the Black-Scholes framework and on an out-of-sample estimation of the hedging performance through non-linear machine learning techniques.

To do so, we needed to follow a research question, without which, we wouldn't have a clear eye on our goal. We phrased two of them in the introduction where, in short, we wanted first to know whether there is a difference in using one method to compute the implied volatility versus another one. For the second sub question, we wanted to know whether it was possible to predict out-of-sample hedging errors in terms of payoff replication or not. This was all presented in the introduction of the thesis.

In the second section we developed theoretically the pricing of an option as well as the methodology for the replicating portfolio. In the third section we outlined our methodology on how we would compute and adjust our implied volatility. Chapter 4 and 5 were the direct applications of the methodology to respectively the S&P 500 and Apple time series. Thanks to chapter 4 and 5 we were able to have our full data set we needed for our machine learning part. In the following section, section 6, we divided it in 2 subsections where the first one outlined the methodology for the second subsection.

In this conclusion, we will hence discuss the results of chapter 4 and 5 and chapter 6. We will also discuss the limitations that were encountered during this thesis and we will end up with some suggestions for future research.

### 8.1 *Implication of the results*

In this section, we will outline the results and implications we found for our 2 questions. We will start with the implication of our first research question. This work has shown that there is no perfect way to compute the implied volatility. Indeed, the ANOVA tests we performed showed us that there is a significant difference in the means of the hedging errors between the methods used. The best method to use here would depend on why the investor is computing this implied volatility. If the investor wants to lower the prices of the call options he's selling for competitive purposes, it is better for him to use the fourth method as the hedging error is really close to 0. If the investor wants to over-hedge its portfolio because of some reason, he would want to use another method leading to a higher hedging error. This is why investors need to pay attention to what they are willing to achieve, to think outside the box and to see the bigger picture.

The implication of the second part of this thesis is that hedging errors are predictable to a certain extent and again, we need to bear in mind what the investor's goal is. If the only goal of the investor is to know if his dynamically hedged portfolio will be over or under hedged, the algorithm will be more accurate compared to an investor who wishes to know the exact end value of the hedging error. This is as intuitive as to say that predicting a categorical variable is more accurate than predicting an continuous variable.

Because of all the hypothesis we had to do and along any results, we need to outline some limitations. The following sub section will outline the latter.

## 8.2 *Limitations of the research*

This section will focus on the limitations of this thesis and the problems that were faced during the writing of this thesis. First of all, we were living a very special moment due to this worldwide pandemic and data collection was really limited. I hence contacted Dr. Lassance to provide me with his data base of his thesis back in 2016.

A first limitation that can be outlined is the fact that all volatility models have not been harnessed. Indeed, limiting ourselves to the implied volatility is a matter of parsimony as we could have been digging into local volatilities or stochastic volatilities.

A second limitation is the fact that we did not relax any of the basic hypothesis of the Black-Scholes model. Indeed, accounting for a constant volatility in real life situation is indeed not realistic. This is a direct cause of our first limitation outlining the fact that we did not harness all volatility models. Ignoring dividend distribution was not a limitation in itself as we knew it could have been easily adapted to all formulas. Another Black-Scholes limitation is the fact that we ignored transaction costs. This might be one of the biggest limitation of this section. Ignoring transaction costs would lead to rethink if rebalancing our portfolio is worth the transaction cost or not. Indeed, we saw that the more often we rebalance our portfolio, the closer the hedging error gets to 0 but on the other hand, the more often we rebalance our portfolio, the more often we will need to pay these transaction costs and that could have a huge impact on our portfolio at the end, going to an over-hedged portfolio to an under-hedged one and hence generating losses.

A third big limitation would be the that we only built our algorithm on 2 time series on a small time span. This could lead to question the robustness of our work on a longer time span and on a higher volatility of the underlying. To improve the robustness of our work we would have needed to take other underlying assets such as currencies, small capitalization stocks or futures and whose implied volatility's behavior is different than the implied volatility of common stocks. Ideally, if we wish to know what the hedging performance would be for one specific stock, we would need to train the algorithm on a past time span for this stock and doing so for all stock we want to consider. However, we can expect our implications to hold for similar companies (Apple and Microsoft, Coca-Cola and Pepsi, Total and Shell, ...).

A fourth limitation comes from the machine learning techniques used. Because of the limited computational power of the computer used, we were not able to dig deep into heavy computational algorithms and hence had to limit ourselves to lower computational ones. Because of that, we limited ourselves to only one algorithm, the artificial neural network. To have more robust results, we could have selected other algorithms to confirm the results of our neural network.

Following [22], our last limitation comes from the data collection. Indeed, taking the closing prices for the option as well as for the underlying asset would lead to a non-synchronous bias. This means that we compare closing prices of options at 3:15 p.m. as the CBOE (Chicago Board Option Exchange) closes at that time and closing prices of underlying assets as the NASDAQ on which Apple is traded and the S&P 500 both close at 4 p.m. According to [22] and [2], the non-synchronous bias has a minor effect meaning that this should not affect this work's implications.

### 8.3 *Suggestions for future research*

Outlining the main limitations of this work leads inevitably to some improvements and suggestions for future research. We will here outline 3 suggestions, 2 regarding the implied volatility modelling and the replicating portfolio construction and 1 for the machine learning section.

Our first suggestion would be to relax the biggest hypothesis of the Black-Scholes model, the constant volatility. This means that besides computing the implied volatility, it is worth computing a deterministic volatility as the local volatility developed by [13] and/or a stochastic volatility as developed by [16].

A second suggestion would be to consider other replicating strategies than the simple Delta hedging strategy. We could indeed, develop a replicating strategy based on a Gamma neutral portfolio. We would here need to take into consideration other assets than the underlying asset to reach neutrality. Indeed, because the underlying asset has a 0 Gamma, we would need to buy or short sell other call or put options depending on how the Gamma's portfolio changes in value. Because of the first suggestion, the volatility is not constant anymore and a $\Delta, \nu$ (where $\nu$ is the vega of an option being the derivative of the option with respect to its volatility) replicating strategy could also be developed. We could also dig into a Greek dynamic hedging strategy, meaning that we would take into consideration, in our replicating strategy, $\Theta$, $\rho$, $\nu$, $\Gamma$, Volga and Vanna, where Volga is the derivative of $\nu$ with respect to the volatility and Vanna is the derivative of $\nu$ with respect to the stock price or alternatively the derivative of $\Delta$ with respect to the volatility.

A third and last suggestion would be to consider other machine learning techniques besides the artificial neural network such as support vector machines, decision trees, random forests or logistic regressions. This could be an entire thesis of it self as we would switch from quantitative finance to data analysis and predictive modelling.

## REFERENCES

[1] Anthony, C. (2018). *The Volatility Surface Explained* retrieved from https://www.investopedia.com/articles/stock-analysis/081916/volatility-surface-explained.asp

[2] Bates, D. (2000). Post-'87 Crash Fears in the S&P 500 Futures Option Market. *Journal of Econometrics*, 94, 181-238.

[3] Bianconi, M., MacLachlan, S., & Sammon, M. (2015). Implied volatility and the risk-free rate of return in options markets. *North American Journal of Economics and Finance*, 31, 1-26.

[4] Bliss, R., & Panigirtzoglou, N. (2004). Option Implied Risk Aversion Estimates. *The Journal of Finance*, 59(1), 407-446.

[5] Boatright, J. R. (2010), *Finance Ethics: Critical Issues in Theory and Practice*, John Wiley Sons, New Jersey.

[6] BIS (n.a.) *"OTC Derivatives statistics at end-June 2019"* retrieved from https://www.bis.org/publ/otc_hy1911.htm

[7] Black, F., & Scholes, M. (1973). The Pricing of Options and Corporate Liabilities. *The Journal of Political Economy*, 81(3), 637-654.

[8] Bre, F. Gimenez, J. & Fachinotti, V. (2017). Prediction of wind pressure coefficients on building surfaces using Artificial Neural Networks. *Energy and Buildings*. 158. 10.1016/j.enbuild.2017.11.045.

[9] Brent, R. (1973) *Algorithms for Minimization without Derivatives.* Prentice-Hall.

[10] Brownlee, J. (2017) *Why One-Hot Encode Data in Machine Learning?* retrieved from https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/

[11] Dangeti, P. (2017). *Statistics for Machine Learning.* [ebook] Packt Publishing.

[12] DayTrading (2019). *Volatility Skew* retrieved from https://www.daytrading.com/volatility-skew

[13] Dupire, B. (1994). *Pricing With a Smile*. Risk 7.

[14] Fays, B., HÃijbner, G., Lambert, M. (2019). *Gamma Trading Skills in Hedge Funds*. Universite de Liege.

[15] Gautam, L. (2016). *Risk Neutral Measures*. Carnegie Mellon University, Mellon College of Science. Retrieved from http://www.math.cmu.edu/~gautam/sj/teaching/2016-17/944-scalc-finance1/pdfs/ch4-rnm.pdf

[16] Heston, S. (1993). A Closed-Form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options. *The Review of Financial Studies*.

[17] Hull, J. (2016). *Fundamentals of Futures and Options Markets* (8th ed.) Pearson.

[18] Ippolito, P. (2019). *Hyperparameters Optimization* retrieved from https://towardsdatascience.com/hyperparameters-optimization-526348bb8e2d.

[19] Jarque, C. & Bera, A. (1987). A Test for Normality of Observations and Regression Residuals. *International Statistical Review*, 55(2), 163-172.

[20] Kantox (n.a.) *Over-Hedging* retrieved from https://www.kantox.com/en/glossary/over-hedging/

[21] Krausz, J. (1985). Option parameter analysis and market efficiency tests: A simultaneous solution approach. *Applied Economics*, 17, 885-896.

[22] Lassance, N. (2016). *Volatility Modelling in Option Pricing and its Impact on Payoff Replication Performance* , Louvain School of Management, Université Catholique de Louvain. Prom. : Vrins, Frédéric.

[23] MacBeth, J., & Merville, L. (1979). An Empirical Examination of the Black-Scholes Call Option Pricing Model. *Journal of Finance*, 5, 1173-1186.

[24] Mendenhall, W., Scheaffer, R., Wackerly, D. (2007). *Mathematical Statistics with Applications* (7th ed.) Brooks/Cole.

[25] Narkhede, S. (2018). *Understanding Confusion Matrix*, retrieved from https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62.

[26] Nelder, J., & Mead, R. (1965). A Simplex Method for Function Minimization. *The Computer Journal*, 7, 308-313.

[27] Newbold, P., Carlson, W. and Thorne, B. (2013). *Statistics for Business and Economics: Global Edition.* (8th ed.) [ebook] Pearson Education Limited.

[28] Nielsen, L.T. (1992). *Understanding* $N(d_1)$ *and* $N(d_2)$*: Risk-Adjusted Probabilities in the Black Scholes Model*

[29] Niu, M., Li, Y., Wang, C. & Han, K. (2018). RFAmyloid: A Web Server for Predicting Amyloid Proteins. *International journal of molecular sciences.* 19.

[30] Rebonato, R. (1999). *Volatility and correlation in the pricing of equity, FX and interest-rate options.* Wiley

[31] SP-Finance. *Theta - the cost of Gamma* retrieved from http://sp-finance.e-monsite.com/pages/variance-swaps/replication-and-hedging/theta-the-cost-of-gamma.html.

[32] Shapiro, S. & Wilk, M. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 55(3-4), 591-611.

[33] Simon, H. (2018). *What is Option Moneyness?* retrieved from https://www.investopedia.com/articles/optioninvestor/08/option-moneyness.asp

[34] St-Louis Federal Reserve. *3-Month Treasury Constant Maturity Rate*, retrieved from https://fred.stlouisfed.org/series/DGS3MO#.

[35] Swilder, S. (1986). Simultaneous option prices and an implied risk-free rate of interest: A test of the Black-Scholes Models. *Journal of Economics and Business*, 38(2), 155-164

[36] Tankov, P. (2006). *Calibration de modeles et couverture de produits derives*. HAL: Universite Paris-Diderot, pp.143.

[37] Tikhonov, A. & Vasiliy, A. (1977). *Solutions of Ill-Posed Problems*, V.H. Winston & Sons, Washington

[38] Ursone, P. (2015). How to Calculate Options Prices and Their Greeks: Exploring the Black Scholes Model from Delta to Vega. [ebook] Wiley.

[39] World population review (2020) *"GDP Ranked by Country 2020"* retrieved from https://worldpopulationreview.com/countries/countries-by-gdp/

[40] Wikibooks (n.a.), *Artificial Neural Networks/Activation Functions* retrieved from https://en.wikibooks.org/wiki/Artificial_Neural_Networks/Activation_Functions#Activation_Functions.

[41] Wikimedia (n.a). *File:Regularization.jpg* retrieved from https://commons.wikimedia.org/wiki/File:Regularization.jpg

[42] van der Wijst, N. (2013). *Finance: a Quantitative Introduction*. Cambridge: Cambridge University Press.

[43] Vrins, F. (2019). *Derivatives Pricing*. Louvain School of Management, Course Material.

[44] Vrins, F. (2019). *Credit & Interest Rates Risk*. Louvain School of Management, Course Material.

[45] West, J. (2012). Money Mathematics: Examining Ethics Education in Quantitative Finance. *Journal of Business Ethics*, 9, 25-40.

[46] Zhang, W. (2015). *Introduction to Ito's Lemma*. Cornell University, Department of Statistics. Retrieved from http://pi.math.cornell.edu/~web6720/Wendy_slides.pdf.

**LOUVAIN-LA-NEUVE** | BRUXELLES | MONS | TOURNAI | CHARLEROI | NAMUR
Place des Doyens, 1 bte L2.01.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/lsm**