Algorithmic Trading with Portfolio Protection

by

Florent Fischer

September 1, 2022

A thesis submitted in fulfillment for the degree of Master of Science in Finance (MScF)

Supervisor: Thomas Cho

Department of Finance, HEC- Faculty of Business and Economics, University of Lausanne

Expert: Luis Ferreira

Financial product strategist - product analysis and selection, BCV Bank

MSc in Finance (Option: Asset and Risk Management)

HEC - Faculty of Business and Economics, University of Lausanne

UNIL | Université de Lausanne

HEC Lausanne

HEC - UNIVERSITY OF LAUSANNE

Declaration of Authorship

I, Florent Fischer, declare that this thesis entitled, “Algorithmic Trading with Portfolio Protection”, and the work presented in it are my own. I confirm that:

- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ______________________________

Date: August 19, 2022
____

Abstract

The purpose of this thesis is to develop and test a technical algorithmic trading system with a built-in portfolio protection mechanism. The most standard technical indicators, namely MACD, RSI, and EMV are first introduced. An approach to help the investment decision is then implemented thanks to the combined reading of these technical indicators. Among the decision tools, primarily, an SVR model is implemented, and then deep learning models with a simple ANN model and an RNN model based on LSTM units are also developed. Finally, various portfolio protection methods are applied and compared to the trading portfolios. We find that modern algorithmic trading using machine learning and deep learning improves the creation of trading signals and that the cost of protecting trading portfolios makes it possible to maintain substantial returns while efficiently controlling for the downside risk. In addition, a web app dashboard was deployed in parallel with this project for easy access to the study results.

Keywords: Trading, Machine Learning, Deep Learning, Portfolio Protection

JEL Classification : G11, G12, G17, G20

Dans ce papier, le sujet est le trading algorithmique avec des méthodes de protection de portefeuille. La forme de trading la plus élémentaire est d'abord présentée avec les indicateurs techniques MACD, RSI et EMV. Puis une approche d'aide à la décision d'investissement grâce à la lecture combinée de ces indicateurs technique est mise en œuvre. Parmi les outils décisionnels, un modèle de machine learning SVR est d'abord construit, puis des modèles de deep learning avec un modèle ANN basique et un modèle RNN composé d’unités LSTM. Ensuite, des solutions de protection sont adaptées à ces derniers avec la mise en place d'une OBPI, d'une CPPI, d'une TIPP, ainsi qu'une version hybride de l'OBPI et de la CPPI. Finalement, un Dashboard a été conçu pour permettre un accès ergonomique aux résultats de l'étude. Le papier conclut que le trading algorithmique moderne utilisant le machine learning et le deep learning apporte une amélioration bénéfique à la création de signaux de trading, et aussi, que le coût de protection des portefeuilles de trading permet de maintenir des rendements conséquents tout en contrôlant efficacement le risque du portefeuille.

Acknowledgement

Primarily, I would especially like to acknowledge Thomas Cho, Professor of Finance, Faculty of Business & Economics of the University of Lausanne (HEC – UNIL), who supervised me throughout this Master thesis. I wish to express my sincere gratitude for all his help and insights in the elaboration of this master thesis. Indeed, without his expert and informed point of view, the coherence of this work would certainly not have been the same. Moreover, I would like to thank him for his willingness, and his constantly encouraging discourse.

I would also like to indirectly thank all the professors of the Master of Finance at the University of Lausanne (HEC – UNIL), and more specifically for those in my "Risk and Asset Management" track. They allowed me to develop my curiosity and my knowledge of finance through the academic excellence of the courses given in this program.

Finally, I would like to express my deepest thanks to my parents, my sister, and my brother, who have always supported me in my choice and without whom I would never have written this academic thesis today. I would also thank them for the cultural and intellectual awareness they have given me.

Executive summary

The introduction aims to define the limits and issues related to the problem of this work. Indeed, the points addressed in this chapter motivate the choices of trading solutions and portfolio protection methods implemented. Afterward, given a few lines for a better understanding of the risks of the financial markets and the losses that trading can cause, is introduced the most basic trading tools used by many financial market participants. Then, to understand how modern finance can guide and help us improve trading accuracy, decision tools are described. For this purpose, some elements are given to understand the usefulness of machine learning and deep learning. Finally, since security trading is a risk-taking activity, methods for portfolio protection that could contain the risk incurred by the different trading portfolios are presented.

The topics mentioned above are described in the subsequent chapter, the literature review, by bringing the most recent publications for each subject. Indeed, several research papers on security trading and their statistical significance are cited, and more generally, papers on systematic and momentum trading. Some research papers on modern machine learning and deep learning methods in finance and security trading are also reviewed. Finally, publications on portfolio protection methods are highlighted to give the reader an idea of the range of existing methods.

In the Data chapter, some key points are highlighted for the rest of the work. The different indices used in this paper are described first, with the S&P500, the AGG, the VIX, and the 60/40 portfolio. A concise description of what they represent is provided, including how they are found (with DataStream® and Bloomberg®). Then, it is described in detail how the European option pricing model "Heston and Nandi" is used to construct the put option dataset. The risks associated with the indices are also discussed because the academic setting does not expose us to the equivalent risk as in a live market investment context, where risks other than asset volatility may appear in a practical trading setting. Finally, a fixed and general cost for each trading transaction is defined.

In the methodology chapter, the first part deals with the purpose of trading portfolio construction. The MACD, the RSI, and the EMV technical indicators are presented with a brief

description and their composition. Once these indicators are optimally configured, machine learning and deep learning models are used to predict market trends by combining all technical indicators. Among the models, the SVR machine learning model is studied first, and then the ANN deep learning model and the RNN model with LSTM units are studied. Each model uses an optimization of the choice of hyperparameters to increase the accuracy. The second central part of the methodology consists of portfolio protection methods applied to the trading portfolios. Several protection methods in the field of capital protection are detailed. First with the OBPI, then the CPPI, therefore the TIPP, and finally, a hybrid method combining the OBPI and the CPPI called the HOC. The same process is used for each portfolio protection method, i.e., the protection solution is introduced by giving a brief description. The general mathematical structure of this protection method is given, and finally, an adaptation of the protection solution to the case trading portfolio is proposed.

The results chapter follows the same path as in the methodology. Indeed, the results obtained within the construction framework of the various trading portfolios and the protection are primarily given. The results of optimizing the parameters for the different strategies are also provided for the trading portfolios. Then, we analyze the performance over the out-sample period of the trading portfolios by comparing them to the performance obtained over the in-sample period. Regarding the results of the protection methods adapted to the trading portfolios, an analysis of the four protections for each trading portfolio one after one is given with a quantitative approach. In this second analysis part, no visualization graphs are presented because of their number. However, thanks to the Web App deployed, all the results obtained in this work are accessible such as performance graphs. This Web App allows the user to search by himself for a trading portfolio and a protection method to obtain the results.

In the last chapter of this work, the conclusion of this paper is given. First, the whole of the work is recontextualized by stating the framework and the method established. In a second step, the main lines of the results obtained are summarized, emphasizing the different points covered in this work. Then, concerning the weak points of the methodology, the limitations are highlighted by explaining where these limitations could come from and how they could be partially solved. Finally, this work is concluded by giving some elements of research for further work on this topic. Among the proposed improvements, we were able to take up the beginnings of solutions proposed to face the limitations of this work.

Contents

List of tables

List of figures

Chapter 1 – Introduction

With the recent development of investment technics such as security trading, many brokers have emerged and deployed their trading platforms. Despite the strict regulations for these financial market access providers, it is evident that they make money from the transactions and requests of each investor. They have a primary interest in inciting transactions rather than advising or warning a particular investor, despite the regulation of the financial markets. These intermediaries between investors and the market base their communication on the potential for investors to earn money quickly. Often on their platform, they provide market analysis tools such as technical indicators, creating the illusion for the retail investor to be practically a professional. Even if some of these investors are very knowledgeable and apply good asset and risk management, many other beginner's investors are exposed to risk, reminding a well-known statistic that between 80% and 90% of retail investors are losing in financial markets[1] without mentioning the emotional and decisional biases that are also a potential source of this statistic.

The main reason for losses in the financial market is the lack of rigor of these investors. They are not sufficiently informed, on the one hand, by the incurred risks, and on the other hand, by the use of analysis tools and the existence of extension of these tools to improve their results. However, with the emergence of the free circulation of information, anyone with access to the financial markets now has access to sufficient information to train and understand the operation of the tools used to analyze the financial markets.

Many technical indicators have existed for many years to make investment decisions. Indeed, they make it possible to capture information emitted by the market. Furthermore, defining a bullish or bearish trend signal depends on a specific investment period. However, using a single technical indicator can lead to significant information losses. It is, therefore, more prudent for an investor to combine several technical indicators to confirm or reject a bullish or bearish market trend signal based on each indicator's information. Using a technical indicator alone is relatively easy for an investor who follows a trading rule related to that indicator. However, as

[1] https://www.quora.com/Why-do-90-of-traders-and-investors-lose-their-money

said earlier, it is helpful to use several technical indicators simultaneously to limit the loss of market information.

The objective of this paper is to be able to answer the problems exposed previously. First, different technical indicators are set up in their most basic form, such as an individual investor would have. Allowing to see how to calibrate these indicators and if they are efficient after a calibration. Then, based on the calibrated technical indicators, we implement machine learning and deep learning methods. These tools, which are relatively recent in the financial literature, allow, in this case, the simultaneous use of technical indicators. The machine learning and deep learning models are implemented as decision tools for the information that the different indicators send back. When well implemented and sufficiently calibrated, these innovative tools avoid a large number of psychological biases of an investor who does not know to discern the information sent by multiple indicators correctly.

Building an algorithmic trading portfolio is interesting, but it can be risky since it entrusts the trading decision to a tool, the final decision is not ours. In a worst-case scenario, a trading algorithm may make successive decisions leading to the complete liquidation of a trading portfolio. Even if this scenario is not very realistic since an investor would have indeed stopped it before, it may be that the market is unpredictable, exposing the investor to a permanent risk. To avoid these situations, many methods of protecting trading portfolios exist. The best-known method is taken from the famous quote, "do not put all your eggs in one basket" it is the so-called diversification and is done by buying several assets instead of just one to protect the overall portfolio.

It led to the second objective of this paper, which is the implementation of portfolio protection solutions. The portfolio protection solutions are based on existing dynamic protection methods but are adapted to the different trading portfolios. Their dynamic characteristic is a key since they bring permanent protection to the various trading portfolios according to the market evolution. These protection solutions involve additional and essential assets belonging to the classes of bonds and derivatives, such as European options. Finally, some portfolio protection solutions use a market volatility index to ensure their protective role.

In contrast to the implementation of trading portfolios and to analyze the portfolio protection solutions, we analyze them by giving protection performance results of portfolios according to different protections and parameter variations. These parameters define the profile of different possible investors. We highlight which protection methods are the most efficient and under what conditions they can provide a practical solution for risk management in the case of protection on an algorithmic trading portfolio.

Chapter 2 – Literature review

Due to their widespread use, market trend indicators are often the subject of research papers. Indeed, permanent verifications of the statistical significance of the trading results obtained with the indicators are necessary to update their optimal calibrations. Among the most recent research papers, Asness et al. (2013) proposed a research paper on Value and Momentum indicators due to their wide use in financial markets. This paper evaluates the statistical robustness of market trend indicators, such as the momentum and value factor, depending on several parameters for these indicators. For the momentum, they used a simple moving average as an indicator of market momentum. They based their study of performance significance on a multiple factor model highlighting several weaknesses in asset pricing models and the overperformance of indicators studied. Their work follows different research papers, such as Goyal and Wahal (2013), who also researched the significance of momentum factors in 36 countries based on different momentum parameters.

In the same study register but on different technical indicators, Chong et al. (2014) proposed a study evaluating the statistical significance that a technical indicator must beat the market. These technical indicators are the MACD and RSI oscillators, and they assumed that the MACD and RSI could beat the market. This research paper detailed these technical indicators and the construction of their trading signal. The paper concludes that the statistical significance of the performance of these technical indicators against the market is confirmed in rare cases, and most often, the market beats these technical indicators.

More recently, Li and Tam (2018) published a research paper in which they set up several trading strategies, i.e., momentum and reversal trading, and trading methods using machine learning and deep learning. It is indeed based on momentum and reversal trading strategies that they built their machine and deep learning model. This paper aimed to highlight the superiority of new decision-making models such as the Decision Tree model, Support Vector Machine model, Multilayer perceptron model, and Recurrent neural network using LSTM units. They see that these models can beat the market and, in some cases, the initial trading strategies.

Finally, still among trading portfolios, Lim et al. (2019) highlight the ability of machine learning and deep learning models to beat the market and improve classical trading strategies. In this paper, they emphasize that technical indicators are time series transformations of the market, making machine learning and deep learning models the most efficient way to read these indicators and provide practical solutions to beat the market. Indeed, these new decision models allow for a significant improvement in the results of trading strategies.

As already mentioned in this paper, different dynamic portfolio protection solutions are implemented based on existing solutions with an adaptation of them to the trading portfolios. These portfolio protection methods were created at the same period as the arrival of asset insurance derivatives, call and put options. For some of the protection strategies, their initial goal was to reproduce the performance of these famous derivatives products to guarantee returns when they reach maturity.

The first research paper on portfolio insurance was written by Perold (1986). This was the first paper in the finance literature to introduce the famous Constant Proportion Portfolio Insurance (CPPI). This method of portfolio protection, among other things, guarantees a portfolio value when maturity is reached. With CPPI, the weights invested between risky and riskless assets vary to guarantee a minimum value during the investment period. This solution also benefits from a bull market and protects the investment portfolio when the market is bearish.

Two years later, Estep and Kirtzman (1988) published an alternative very close to the CPPI, the Time Invariant Portfolio Protection (TIPP). This method is practically based on the same structure as the CPPI and benefits from bullish market trends by varying the protection threshold of the strategy when maximum portfolio levels are reached. The method avoids a scenario of a substantial decline in the portfolio's value following an extreme rise. Indeed, each time a maximum portfolio value is reached, this value is used as a reference threshold below which the portfolio can no longer fall.

As mentioned earlier, portfolio protection methods can also use derivatives such as options. The best-known method is Option Based Portfolio Protection (OBPI) which was first published in the paper of Leland and Rubinstein (1988). This paper implemented a static portfolio protection solution using the protective put method. This protection method guarantees a minimum portfolio value when maturity is reached, in a static framework, unlike the CPPI.

Because of the proximity and the pronounced similarity between the CPPI and the OBPI, many recent works propose stochastic comparisons of these two methods. Notably, the research paper of Bertrand and Prigent (2001) highlights these two methods' stochastic relevance. Following the results of the paper mentioned above, new methods of portfolio protection called hybrid could be imagined to improve the stochastic relevance of these methods to create new structured products. Di Persio et al. (2021) show this in their implementation of a hybrid version of the OBPI and the CPPI, which obtained encouraging results for the creation of new protection methods.

Finally, despite the evidence of these different methods to protect a portfolio, Nicholas McQuinn et al. (2021) show that the results obtained are often quite different. Indeed, in their paper, they compare a 60/40 portfolio with several protection strategies aimed at risk mitigation. They discovered that underestimating market volatility, especially for theoretical OBPI methods, does not always guarantee the healthy protection of a portfolio.

Chapter 3 – Data

3.1 Data description

Through this work, we base on the U.S. financial market. Indeed, the objective is focused on the Standard & Poor's 500 Index (S&P500). Among the index used, the first one is the S&P500. Then the second index is the CBOE Volatility Index (VIX). The third one is the Bloomberg US Aggregate Total Return (AGG). Finally, to compare the strategies implemented in the continuity of this work, a portfolio with an allocation of 60% in equity and 40% in bonds is created with daily rebalancing inspired by the 60/40 Bloomberg Multi-Asset Index (60/40). The data used come from two different sources. For the S&P500 and the VIX, the data comes from DataStream®. For the AGG, the data was extracted with a Bloomberg® terminal.

The S&P500 is a stock market index constituted of the 500 largest companies in the United States listed on the stock exchange. It was created by the Standard and Poor's Global Rating company on March 4, 1957. The S&P500 is used in every trading strategy implemented to conduct this research. Since we are dealing with trading a global index, we do not replicate the individual selection of each security that composes the S&P500. To exploit most of the qualities of this index, we have chosen a daily data set from 01-01-1960 and ending on 01-12-2021 consisting of opening, closing, and adjusted closing prices as well as the volume.

The CBOE Volatility Index (VIX) is an index introduced in 1992 by the Chicago Board of Options Exchange and aims to measure the expected volatility of S&P 500 call and put options with 30 days to maturity. It is commonly used to capture the expected volatility of the U.S. financial market. For this work, the VIX is used only for the out-sample period and not for the in-sample since it defines no parameters in the strategies. Indeed, the VIX is used in a second step to protect the portfolios created. Therefore, the VIX dataset dates from 01-01-2000 and ends on 01-12-2021, with a daily chronology.

The Bloomberg US Aggregate Total Return (AGG) is a general bond index melting various kinds of financial products in the US market related to the bond market. As mentioned in the product description provided by Bloomberg, "*the index includes Treasuries, government-related and corporate securities, MBS (agency fixed-rate pass-throughs), ABS, and CMBS*" [2]. In this work, it is used as a "risk-less" asset. The data set starts on 01-01-2000 and finish at the end of 2021, like the two previous indices. The data extracted is based on daily values. According to the VIX and the protection process, this index is used on the out-sample period since no parameter is estimated with it.

To compare the performance of the different strategies, a benchmark is used as a reference for the out-sample analysis. A portfolio index is used with an allocation of 60% equity and 40% fixed income. This index is inspired by the Bloomberg US Multi-Asset Index. This index is composed of 60% US Equity (S&P500) and 40% US Fixed income[3] (AGG), and the portfolio weights are rebalanced daily without any transaction costs. This index is composed of daily data starting from 01-01-2000 to 01-12-2021.

A final element of data to be added is the annual risk-free rate, sometimes used to estimate potential price variations and metrics. Based on US market rates from 2000 to 2021, a general estimate of 3% based on 10-year US Treasury bill interest rate data[4] is given as the risk-free rate.

Concerning the statistics of the dataset, the key elements of each index are given in Table 3.1. Including the VIX index, even if it cannot be compared to the three leading indices given its characteristic of measuring market volatility. In addition, a box plot allowing a better understanding of the variation's distribution of the three leading indices is also given in Figure 3.1, giving a better reading of the data. Among the three indices, the annualized returns vary between 4.8% and 8.2%, the S&P500 being the highest and the AGG being the lowest. This also partly explains the highest volatility for the S&P500 with 16.1% and the lowest for the AGG with 3.7%. The same order of repartition is also found for the minimum and maximum returns. Indeed, the S&P500 is the index that obtains the lowest minimum returns and the highest maximum returns. For the AGG, it has the highest minimum return and the lowest maximum return. The 60/40 index is in the middle, with a minimum of –6.8% and a maximum of 6.3% in returns.

---

[2] https://www.bloomberg.com/quote/LBUSTRUU:IND
[3] https://www.bloomberg.com/quote/BMA6040:IND
[4] https://www.cnbc.com/quotes/US10Y

The indices' Sharpe ratio allows seeing the performance of each index. The S&P500 and the 60/40 portfolio have a Sharpe ratio of around 0.5, while the AGG riskless index has a Sharpe ratio of 1.29. This very high ratio is explained by this index's constant returns and very low volatility. Finally, concerning the shape of the distribution of the returns, each index has a negative skewness, implying more frequent negative returns. The excess kurtosis allows observing the shape of the distribution, and the AGG index has an approximately normal distribution with excess kurtosis of about 3. The S&P500 and the 60/40 portfolios have a platykurtic shape, i.e., a high density of negative returns compared to the rest of the distribution.

Concerning the VIX, its statistics analysis is irrelevant since its comparison with other indices is meaningless. However, these statistics express more about the very volatile nature of this index, which allows understanding its particularity.

Indices	Number of observations	Mean	Std. Dev.	Min.	Median	Max.	Sharpe Ratio	Skewness	Excess Kurtosis
60/40 Index	5479	0,0602	0,1141	-0,0684	0,0004	0,0630	0,5273	-0,3425	10,2966
S&P500	15550	0,0823	0,1615	-0,2047	0,0005	0,1079	0,5096	-0,3156	9,8805
US Bond Agg.	5479	0,0481	0,0372	-0,0207	0,0003	0,0133	1,2930	-0,3193	3,1430
VIX Index	5479	0,5931	1,1182	-0,2957	-0,0037	1,1560	0,5304	1,9425	16,2239

Table 3.1- Assets statistics

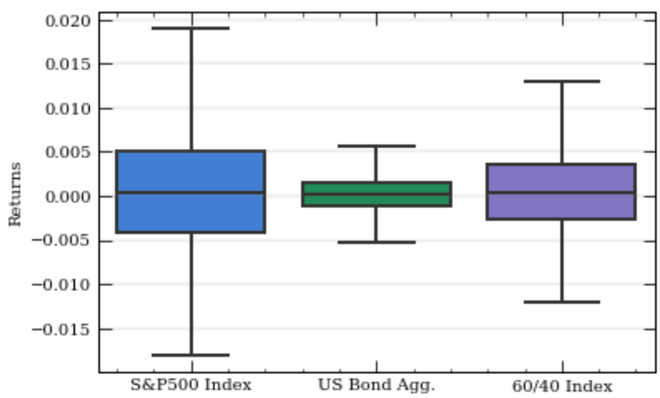

Figure 3.1 – Boxplot of daily index returns

3.2 Option pricing model

To conduct this study, options data of the S&P500 as the underlying are needed. Unfortunately, options products are difficult to find for sufficient historical length. Facing this lack of resources, an alternative is to use option pricing models. This allows flexibility regarding options maturities and moneyness, even if it can also bring mispricing errors. Generally, the methodology used to price stock options comprises two parts. The first one is the implementation of a model that aims to estimate the potential future returns of the concerned asset from statistical observation of it, which can be, for example, past returns and past volatility. Then, when the model is calibrated, many future returns are simulated to reduce mispricing errors, and finally, the price average of the options is calculated with the stock simulations. In a nutshell, options pricing models consist of model calibration, followed by a pricing simulation.

A wide range of pricing models exist for options, and each has its particularities. So, specific criteria are essential to isolate the most relevant one. In this case, the implied volatility is a crucial element in option pricing. Indeed, options are financial products that tend to be relatively sensitive to market volatility. This involved the use of a model that included the volatility in it. One of the most suitable models for this purpose is a GARCH model, and more precisely, the GARCH model created by Heston and Nandi (1997) in the case of option pricing. One of the strong points of the GARCH model is the inclusion of volatility that adapts over time, and according to market trends, the conditional variance captures the estimation of the market volatility. Moreover, the GARCH model considers the volatility persistence according to the evaluated asset, i.e., after a shock such as a financial crisis, the model has a performant adaptation in terms of a decrease of the estimated volatility according to the historical persistence of the asset. Finally, the Heston and Nandi model[5] is an adaptation built based on a GARCH model and allows for relating the volatility of the asset to its logarithmic price. Once the Heston and Nandi model is calibrated, a simulation with 10'000 iterations is launched to estimate the potential evolution of the estimated asset prices. Finally, when all the daily returns of the prices have been simulated, the average of options[6] prices is computed with the returns found.

[5] Appendix B.2: Description and details of the Heston and Nandi model implemented
[6] Appendix A.8: Visualization of 10 random options from the model implementation

3.3 Risk of the products

While this paper represents a theoretical investment framework, it is essential to understand the relationship between the investment strategies and the potential investment in a live context. Indeed, the theoretical implementation avoids many scenarios an investor can face in a live market. Firstly, the risk related to the types of products exchanged. And secondly, the risk associated with the imperfect reflection between theoretical and genuine prices.

First, regarding the risk related to the types of products traded, several markets are involved. Indeed, indices such as the S&P500 and AGG are not traded in the same market as European options. The S&P500 and AGG can be traded on the exchange markets as futures contracts and/or ETF products. The risks associated with trading these products are low since the exchange price is guaranteed on the expiration date.

However, European options are mainly traded on the Over-The-Counter market. This type of exchange is particular because the contract settlement is done only at maturity. Unlike futures contracts, OTC trades involve the risk of default. Indeed, since the exchange is done privately between two parties, if the other party goes bankrupt at the end of the contract or before, the exchange is canceled without reimbursement of potential losses, known as "counterparty risk."

Finally, the second risk to consider in a precise investment case is the poor reflection between the prices that constitute our data set and the execution of an order in live trading (market microstructure). In this paper, the strategies are based on daily variations of the adjusted asset prices. However, since the execution of a buy or a sell must occur during the market's opening, it is evident that a transaction rarely is the same as the values of the daily data. This risk linked to the execution of a transaction is an element that depends notably on the markets' liquidity and the markets' microstructure, which is not represented in this work but would necessarily impact the results in a real investment case.

3.4 Transaction costs applied

To be as close to an authentic investment case, transaction costs are considered in this work. These transaction costs are based on estimates of several elements. The transaction costs practiced by brokers are generally between 0.2% and 0.5% per transaction made. Even if it is not a generality for every broker, these transactions' costs are an average of the fees on the financial markets for our types of products.

Then, Engle et al. (2012) provide solutions to this question. It depends mainly on the liquidity, volatility, and volume of the trade realized. Concerning the NYSE and the NASDAQ, they found a maximum level between 0.2% and 0.3% of transaction cost if 1% of the daily volume is traded. They studied the range between 0% to 2% of the daily volume traded, so 1% is the middle range, and with the average of the cost, the global cost is fixed at 0.25% per trade executed.

Based on these two elements mentioned previously and the diversity of the products in this work, we can estimate that the overall transaction costs are about 0.25% per transaction made. These transaction costs depend on the volume traded, so that they may be slightly underestimated or overestimated in a real investment case.

Chapter 4 – Methodology

4.1 General framework

In this chapter, the methodological structure necessary for implementing this topic is set up, and two parts are derived from it. In the first part, the subject of systematic trading is approached by developing technical indicators and algorithmic trading based on Machine Learning and Deep Learning. This first part can be seen as the construction of different trading portfolios. Then, in the second part, portfolio protection solutions are implemented based on the trading portfolios established in the first part. Concerning the analysis of the different investment strategies resulting from the two parts mentioned above, each part has precise particularities to highlight, but globally, the same evaluation measures are used. Among these measures, the following points enable to see how each one is calculated and what they highlight.

The annualized return is the method of computation of the yearly returns in percentage, realized by an investment, and over a given period. The formula of the annualized return r is given by:

$$r = \frac{\sum_{i=1}^{n} r_i}{n} \cdot 252 \tag{1}$$

where r_i are the daily returns, n is the number of days of the sample and 252 is the number of trading day in a year.

The volatility is the method of computation of the volatility in percentage, realized by an investment, and over a given period. The formula of the volatility σ is given by:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} (r_{\mathrm{i}} - \bar{r})^2}{n-1}} \cdot \sqrt{252} \tag{2}$$

The Sharpe ratio is a risk-adjusted performance measure of investment created by Sharpe (1994). It involves the returns and the volatility of an investment strategy. The formula of the Sharpe ratio is given by:

$$\text{Sharpe ratio} = \frac{r - r_f}{\sigma} \tag{3}$$

where r_f is the risk-free rate, r is the annualized return, and σ is the volatility. As lecture guidance, an investment with Sharpe between 0.5 and 1 is a good investment, a Sharpe ratio over 1 being an efficient investment strategy in terms of profitability.

The Sortino ratio is a downside risk-adjusted performance measure of investment like the Sharpe ratio. It was first introduced in the paper of Sortino and Price (1994). It is given by the ratio between the returns of the investment and the downside risk. The formula of the Sortino ratio is given by:

$$\text{Sortino ratio} = \frac{r}{\text{SD}} \tag{4}$$

where SD is the semi-deviation and correspond to the squared root of the average negative returns over the returns sample, and r is the annualized return. Also, like the Sharpe ratio, a Sortino ratio between 0.5 and 1 is a good investment and over 1 is highly profitable.

The Value at Risk (VaR) is a risk measure that aims to provide the potential loss of an investment given a probability, as shown with the formula:

$$\text{VaR}_{95\%} = \min\{r_i : F_L(r_i) \geq 0.95\}, \tag{5}$$

where r_i are the daily returns and F_L is the cumulative distribution function of the loss L. For a 95% threshold, the VaR shows that in 5% of the case, the investment gives returns lower or equal to the VaR value.

The Conditional Value at Risk (CVaR) is a risk measure that aims to provide the expected loss of an investment given a probability, as shown with the formula:

$$\text{CVaR}_{95\%} = \lambda \cdot \text{VaR}_{95\%} + (1 - \lambda) \cdot \text{CVaR}^{+}_{95\%} \; ; \quad \lambda = \frac{F_L(\text{VaR}_{95\%}) - 0.95}{1 - 0.95} \tag{6}$$

where $\text{CVaR}^{+}_{95\%}$ is the Upper CVaR. For a 95% threshold probability, the CVaR shows that in the 5% of the case, the investment is expected to lose an average CVaR value in returns.

The Maximum Drawdown (MDD) aims to provide the maximum loss of the investment after the previous highest cumulative returns value, given by:

$$\text{MDD} = \min\left(\frac{C - P}{P}\right) \tag{7}$$

where C is the cumulative returns over the returns sample and P is the peak (or maximum) of cumulative over the returns sample. This measure shows if an investment strategy is well protected against consecutive losses or not.

For all the computational elements of this work, the Python programming language was used with the help of specific packages and libraries related to the subject. Except for the elementary packages, python libraries such as `Sklearn`, `TensorFlow` from the `Keras API`, and `Optuna` are the most important ones, and their use is mentioned in the specific part. A requirements file is available on the paper GitHub for the detail of libraries used, and the GitHub link is provided at the end of the Results chapter.

4.2 Trading portfolios

4.2.1 Technical indicators general framework:

Financial markets face fluctuations that seem to follow random patterns when taken independently. However, even if independent realizations taken one by one are useless, taking them together for a specific period can help capture some patterns called systematic patterns. Indeed, long-term tendencies can be observed when paying attention to them. It is in the interest of being able to anticipate market trends that systematic indicators have been designed. They are called "Technical Indicators." They allow for observing periods with a strong bull market trend and periods with a strong bear market trend. So, the technical indicators are tools that help to determine when to buy or sell an asset.

However, a common root for all technical indicators is based on their construction. First, an indicator is built for any security. Then, the configuration of the parameters of this technical indicator is determined to define a buy or sell signal of the asset. Therefore, despite the differences linked to the specificity of the analysis, the objective ultimately is to build a signal that defines a buy or sell position for the asset studied. In this paper, the sell possibility is eliminated. Indeed, selling an asset is something that can bring additional risk and then volatility to trading strategies. Thus, when an indicator recommends selling the asset, the position is closed, and cash is held instead of the security.

Every technical indicator has its particularities. Some are better to use in short-term trend analysis, and others in long-term trend analysis. However, they are customizable, so a short-term could be used for long-term analysis, and similar for the opposite. The relevance and significance of the results obtained are impacted when trying to change the original use of an indicator. Three indicators are implemented for three different types of analyses in this work. The first one, the “Moving Average Convergence Divergence” (MACD), is focused on long-term trend analysis, and the second one, the “Relative Strength Index” (RSI), is usually used for middle and long-term trends analysis. The last one, the “Ease of Movement Value” (EMV), is also used for middle-term analysis like the RSI. Despite their specific period application, a statistical significance parameter test is made for every technical indicator on the in-sample period to avoid any mistake of arbitrary period params selection.

The choice of these three indicators is coherent because they allow for observing three different cases of analysis. They also highlight individual particularities that each technical indicator can bring. Indeed, the three indicators presented above are rarely used alone in live trading scenarios, and they are typically used together to highlight a strong market trend. Therefore, it is essential in this analysis to observe each indicator's effectiveness and calibrate their parameters as much as possible. In the second step, these indicators are used together in machine learning and deep learning algorithms to get the highest accuracy. For the three indicators, the same method is used. First, a short description of the indicator is provided. Then the construction of the indicator and its signal is described. Finally, one or two parameters of the technical indicator that can affect the result of the overall strategy are chosen, and a few combinations of these parameters are tested to find optimal ones.

To evaluate the in-sample strategy performance and affirm that one strategy is significantly better than another, we follow the method proposed by Harvey and Liu (2014). This method adapts the Sharpe ratio of a strategy in a t-statistic. In that way, we find the p-value associated with the trading strategy and evaluate the statistical significance. However, an adaption of the p-value is treated case by case. Indeed, this method involves multiple testing, and this p-value adaptation depends on the number of tests realized. The Sharpe ratio to t-statistics is general for all indicators and is given by:

$$\mathrm{T} - \mathrm{Statistic} = \mathrm{Sharpe\ ratio}\ \times \sqrt{\mathrm{Number\ of\ years}} \tag{8}$$

For the three technical indicators studied, optimal parameters that seem statistically significant are found during the in-sample frame. Then with these optimal parameters found, we base the results on them for every technical indicator during the out-sample period. This out-sample result is used to compare all strategies implemented in this work.

4.2.2 MACD technical indicator

The MACD is one of the most popular technical indicators for trading. This indicator has known a vast evolution since its creation. Initially, it was built on a simple moving average (SMA), as shown by the recent work of Asness et al. (2013), where it simply allowed to observe if the sign of the market's average returns were positive or negative over a past period of observation. With this simple positive or negative value of past returns average, they were able to conclude if the market is in a positive or negative trend. This indicator is actually better structured and is much more elaborate than initially. The latest research about MACD was produced by Baz et al. (2015) and then was taken up in papers by Lim et al. (2020). These research papers found that the MACD indicator is constructed by crossing two exponential moving averages (EMA). The trend created by these two averages is normalized to produce an oscillator indicator and create the trading signal as consistent as possible with the market trend. The exponential moving averages have the particularity of weighting more strongly the values close to the current market value and weakly the values far from the current market value, providing better results than a simple moving average. Using the indicator proposed by Baz et al. (2015) and then taken up by Brian Lim et al. (2020), the MACD is constructed as follows:

Indicator construction:

$$\mathrm{MACD}_t(C,L) = \mathrm{EMA}_t\left(S_t, \frac{1}{C}\right) - \mathrm{EMA}_t\left(S_t, \frac{1}{L}\right) \tag{9}$$

$$\mathrm{EMA}_t(S,\alpha) = \begin{cases} S_0 & , \text{ if } t = 0 \\ \alpha \cdot S_t + (1-\alpha)\cdot \mathrm{EMA}_{t-1}(S,\alpha), & \text{if } t > 0 \end{cases} \tag{10}$$

$$q_t = \frac{\mathrm{MACD}_t(C,L)}{\mathrm{std}\left(S_{t,t-63}\right)} \tag{11}$$

$$X_t = \frac{q_t}{\mathrm{std}\left(q_{t,t-252}\right)} \tag{12}$$

$$Y_t = \phi(X_t) = X_t\, e^{-x_t^2/4}\, \frac{1}{0.89} \tag{13}$$

where S_t is the asset price at time t; C and L are respectively the short-term and long-term lookback windows for the EMA; α is the exponential smoothing ratio corresponding to $1/C$ or $1/L$ depending on the EMA concerned. The variable q_t is the first normalization of the MACD indicator and X_t is the second normalization and final version of the technical indicator of the MACD according to Baz et al. (2015). These MACD normalizations ultimately help avoid a trend distortion due to a sudden shock.

In the second step of the signal rule, the trend is transformed into a signal, as shown in formula (13). This function aims to convert the trend into a signal emphasizing an acceleration of change of trend, as shown by Lim et al. (2020) and as shown in Figure 4.1.

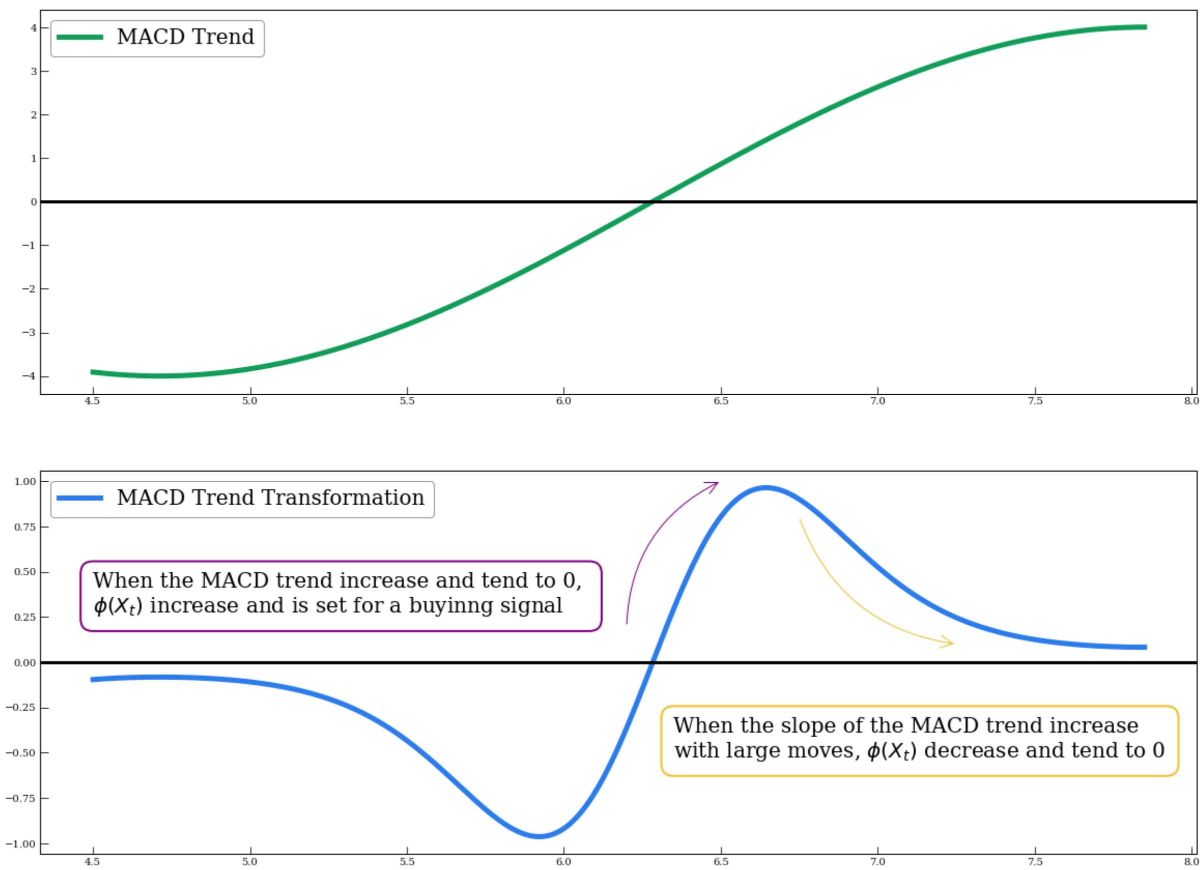

Figure 4.1 – MACD trend to signal transformation

This trend transformation makes it possible to avoid trend reversal and thus anticipate a market change more quickly. So, this signal trend transformation is essential for a coherent and further signal interpretation. Finally, with formula (13), the signal rule is implemented as follows:

Signal rule:

- $\text{signal}_t = 1, \text{ if } Y_{t-1} \geq 0$
- $\text{signal}_t = 0, \text{ if } Y_{t-1} < 0$

Once the trading strategy is implemented, the choice for the short and long period variable of the exponential moving average determines the performance results. This parameters calibration is essential to realize within the in-sample period. The objective is to plot a heatmap of all possible couples of params and observe which one produces the highest t-statistic with the methodology provided by Harvey and Liu (2014), where they propose to transform the strategy's Sharpe ratio in a t-statistics first and adjust the p-value of the t-statistic with a multiple test methodology. The range of params variations tested in this work is given in Table 4.1.

Parameter	**Parameter range of variation**
Long window (L)	[10, 12, 14, …, 118, 120]
Short window (C)	[10, 12, 14, …, 118, 120]

Table 4.1 - MACD In-sample parameters variations

The long and short parameters are tested to observe the strategy reaction. The computations to obtain the heatmap build a lower triangular matrix of the t-statistics. Therefore, as mentioned by Harvey and Liu (2014) and proposed by the Bonferroni test, a transformation of the t-statistics is mandatory in the case of multiple testing. When evaluating several trading strategies and in the context of multiple testing, the p-value depends on the number of tests performed, as shown by the formula (15). This transformation by the Bonferroni test is not wholly efficient since it does not consider the value of the other tests performed but did not impact the results.

$$p - \text{value} \cdot \text{number of test realized} < \text{threshold} \tag{14}$$

$$\Leftrightarrow \; p - \text{value} \cdot \frac{n(n+1)}{2} < \text{threshold} \tag{15}$$

where n is the number of parameters tested for one lookback rolling window.

Following the implementation of this test on the in-sample dataset, the ideal and statistically significant pair of parameters is found. These parameters are used when implementing the strategy during the out-sample period, which serves as a result for analysis in this work.

4.2.3 RSI technical indicator

The RSI is a well-known technical indicator created by Welles Wilder (1978). It belongs to the family of oscillator technical indicators. Unlike the MACD indicator, the RSI indicator is generally used in a smaller period range. This technical indicator observes overbought periods and oversells periods of an asset. An overbought period is defined as a period where the asset price goes up rapidly and is undoubtedly followed by a period where the asset price goes down. In the same logic, oversell period is defined as a period where the asset price tends to go down rapidly and is followed by a period where the asset price rise. The RSI indicator is commonly used in parallel with another indicator like MACD to confirm potential market trends. The RSI indicator takes values between 0 and 100, where 0 represents the maximum oversell level and 100 is the maximum overbought level. The neutral level between overbought and oversold is determined by the middle point of 50. In a classical RSI indicator setup, threshold levels are implemented. Indeed, even if 0 and 100 are maximum levels, they are rarely reached. In the literature, the oversell threshold level is 30, and the overbought threshold is 70. These threshold levels can be fixed because they illustrate well the live market context. To perfectly understand this oscillator indicator, his value permanently fluctuates between 0 and 100, allowing him to take advantage of the market by deciding optimal open and close positions. To construct the RSI indicator, we based on the original formula of Welles Wilder (1978) and the paper of Chong et al. (2014). More formally, the RSI indicator is given by (16) and uses the formula of the exponential moving average (10) from the MACD computation.

Indicator construction:

$$\mathrm{RSI}_t(N) = \frac{\mathrm{EMA}_t\left((S_{t-i} - S_{t-i-1}) \cdot \mathbb{I}\{S_{t-i} < S_{t-i-1}\}, \frac{1}{N}\right)}{\mathrm{EMA}_t\left(|S_{t-i} - S_{t-i-1}|, \frac{1}{N}\right)} \cdot 100 \tag{16}$$

where S_t is the asset price at time t, N is the lookback window associated with the exponential moving average, and $\mathbb{I}$ is the indicator function.

Concerning de signal rule, it differs from the MACD indicator. RSI indicator is already bounded between 0 and 100 thresholds. So, no additional indicator trend adaptation is necessary to normalize the technical indicator values. A classical setup is defined with an oversell threshold at 30, an overbought threshold at 70, and a neutral threshold at 50. Then, the rules of buying decisions are defined as follows.

Signal rule:

- $\text{signal}_t = 1, \quad \text{if } 50 < \text{RSI}_{t-1}(N) < 70$
- $\text{signal}_t = 1, \quad \text{if } \text{RSI}_{t-1}(N) \leq 30$ and till $\text{RSI}_{t-1}(N) \leq 50$
- $\text{signal}_t = 0, \quad \text{if } \text{RSI}_{t-1}(N) \geq 70$ and till $\text{RSI}_{t-1}(N) \geq 50$

Through these trading rules, the RSI oscillator follows mean-reversion patterns that help determine the market tendency for middle-term trading. More generally, we enter a buying period whenever the RSI value is between 50 and 70, except for an overbought period where the buying signal is blocked due to the mean reversion characteristic of the RSI. With the same argument, after an overbought period, when the RSI crosses the 30 thresholds, a buying signal is sent because of the mean reversion, meaning the market reverses to an intense buying period.

Therefore, the parameter choice is decisive in the optimal parameter selection. The exponential moving average parameter value corresponding to the window of the lookback period is the best candidate. The range of parameters tested is given in Table 4.2.

Parameter	Parameter range of variation
Lookback window (N)	[10,11,12, …, 119,120]

Table 4.2 - RSI In-sample parameters variations

As for the previous indicator, a decision for the optimal parameter choice is applied with the Sharpe ratio transformation in the t-statistic, following Harvey and Liu (2014). Then the p-value is adjusted for better adapted multiple tests and see how significant the result is after testing several trading strategies. Such as before, a Bonferroni test is used to adapt the necessary p-value for a relevant statistical significance. In this case, n different parameters possibilities are tested so the p-value can be adapted with the following formula:

$$p - \text{value} \cdot \text{n} < \text{threshold} \tag{17}$$

where n is the number of tests realized.

The testing period is implemented within the in-sample framework. Then, after finding which candidate constitutes an optimal parameter, we determine it as the optimal parameter to be used for the out-sample period. This out-sample period with the optimal parameter is used as a reference for the trading strategies analysis across all other strategies implemented in this work.

4.2.4 EMV technical indicator

The EMV is a less known indicator than the other two presented before. The EMV is a momentum and volume indicator created by Richard W. Arms that aims to react to sudden market variation. This indicator is typically used with a similar timeframe as the RSI indicator but with less often trading. Contrarily to MACD and RSI, the EMV detects fast market trends by using information brought by volumes traded on the market. The choice of the EMV is not random. Indeed, it adds another middle-term momentum indicator to the group, already composed of a middle-term momentum indicator with RSI and a long-term momentum indicator with MACD. Moreover, his particularity is the use of volumes to detect a market trend, so it gives a new way to capture market information about tendencies and avoid missed information from MACD and RSI indicators. Furthermore, EMV uses High and Low values of the trading day, giving additional information. EMV indicator is commonly used in combination with other indicators to confirm market tendency. This indicator is an oscillator that takes both positive and negative values, such as MACD indicators. A significant negative value of EMV means a large selling volume on the market. Inversely, a significant positive value of EMV shows a large buying volume on the market. In this work, the implementation of EMV was inspired by the Investopedia given formula[7], which is the general formula for EMV indicators.

Indicator construction:

$$\text{Distance moved}_t = \frac{\max(S_t) + \min(S_t)}{2} - \frac{\max(S_{t-1}) + \min(S_{t-1})}{2} \tag{18}$$

$$\text{Box Ratio}_t = \frac{\frac{\text{Volume}_t}{\text{Scale}_t}}{\max(S_t) + \min(S_t)} \tag{19}$$

$$\text{EMV}_t = \frac{\text{Distance moved}_t}{\text{Box Ratio}_t} \tag{20}$$

[7] https://www.investopedia.com/terms/e/easeofmovement.asp

$$\mathrm{EMV}_t(N) = \frac{\sum_{i=0}^{N-1} \mathrm{EMV}_{t-i}}{N} \tag{21}$$

As presented above, $\max(S_t)$ is the highest value of the asset on the trading day t, and $\min(S_t)$ is the lowest value of the asset on the trading day t. Volume is the overall traded volume on the trading day. $\mathrm{EMV}_t(N)$ is the simple moving average formula for EMV with N rolling days. Scale is a volume reduction constant. It depends on the asset traded and can range between 1e3 to 1e7. Because the S&P500 is an actively traded index, we fix the global daily volume value to 1e7.

Signal rule:

- $\text{signal}_t = 1, \quad \text{if } \mathrm{EMV}_{t-1}(N) \geq -1$
- $\text{signal}_t = 0, \quad \text{if } \mathrm{EMV}_{t-1}(N) < -1$

Trading rules are quite simple, when the EMV value is greater than -1 a buying position is emitted, and when the EMV value is lower than -1 a selling signal is emitted, and no additional rules are necessary.

Regarding optimal params selection, the only parameter variation is focused on the simple moving average rolling window value, similarly to the other two indicators. So, the selection is implemented with the same structure as the RSI one parameter selection, and details are given in Table 4.3.

Parameter	**Parameter range of variation**
Lookback window (N)	[10, 11, 12, …, 119, 120]

Table 4.3 - EMV In-sample parameters variations

A decision for the optimal parameters is applied by using the Sharpe ratio transformation in t-statistic, following Harvey and Liu (2014). Then, the p-value is adjusted for multiple tests and see the statistical significance by performing a Bonferroni test that adapts the necessary p-value for statistical significance. As for the MACD and the RSI, n different parameters possibilities are tested so the p-value can be adapted with the formula (17), already seen in the previous section.

The optimal parameter-seeking period is implemented during the in-sample period. Then, after finding which parameter is optimal, it is used for the out-sample period. This out-sample period with the optimal parameter is based as a reference for trading strategies analysis across all other strategies implemented in this paper.

4.2.5 Machine learning and deep learning general framework

Previously, we implemented trading with technical indicators and each of them has its particularities and strength in reading market trends. They help capture all market information and elements possible. Nevertheless, it remains a problematic issue when combining several indicators in the decision-making process. Indeed, it stays difficult, and the decision-maker can suffer from several biases that can unintentionally lead him to bad decisions and poor performance results.

Decision biases constitute a considerable hazard to individual investor performance. A solution to overcome them should be to entrust a professional advisor. However, even if they respect precise decision rules, they are also exposed to biases. By using an autonomous decision-making process and, more precisely, Machine Learning (ML) and Deep Learning (DL) processes, such bias can be enhanced. These decision models are recent, but they are widely accepted as they can, in some cases, improve the performance of those who use them. ML and DL models are not an exact science, but they give powerful results when they are well adjusted. Above all, they avoid decisional biases linked to human action. They can unfortunately be exposed to other biases, such as the overfitting bias, which is limited by applying model calibration solutions and partially avoiding them.

It is essential to know that the ML and DL models rely heavily on the amount of data on which they are trained. Undeniably, the more observations a model is trained on, the more it gives a precise decision for a given prediction. The size of the dataset, although relatively large between the three indicators, can be increased to improve the model's accuracy by proceeding to a data augmentation of the dataset. Many data augmentation solutions exist for ML and DL models, but the task is more complicated for financial data since not all methods are suitable. The most common data augmentation method in artificial intelligence is applied to convolutional neural networks, especially for image recognition applications. Unfortunately, in the data augmentation for convolutional networks, the task is simple since it is enough to transform the details of an image that we know entirely to train the network. Such an application with financial data would undoubtedly imply a forward-looking bias since the transformation is constructed with the knowledge of the complete dataset. Therefore, this type of method that includes a forward-looking bias is eliminated because the training results would be falsely improved. After reading the research paper by Fons et al. (2020), we can derive the most suitable method for this situation. To increase the data set, data jittering is performed. Although this method is not the most efficient, as shown by this research paper, it slightly improves the accuracy of the models.

Data jittering is a data augmentation method that consists of adding a small amount of white noise to the initial dataset, giving additional texture and perturbation to the initial dataset, and allowing the model better learning. The jittered data is then concatenated to the original dataset as an input variable. The formula for the data augmentation transformation is given by (22). It shows that augmentation data represent only minor stress of the initial input data, following a normal distribution $\mathcal{N}(\mu, \sigma)$, for data with a distribution mean μ and volatility σ. Five data augments are created for every indicator associated with the initial input data X^i.

$$X^i_{\text{augmentation}} = X^i_{\text{initial}} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\mu, \sigma) \tag{22}$$

However, it should be noted that for the Machine Learning model studied in this paper and in general, data augmentation processes are not particularly interesting and do not bring better results. It mainly concerns DL models that need a vast amount of data to provide correct accuracy, so data augmentation is used for DL models only.

Contrary to data augmentation, which only concerns deep learning models, a specific treatment of all the data used in the models is necessary, called data standardization. Data standardization is a data processing essential to facilitate the reading of input data for machine learning and deep learning models, as shown in the paper of Shanker et al. (1996). Without data standardization, the accuracy of the models can be fatally impacted. In this case, we opted for robust data standardization, as shown in the formula (23). More precisely, data standardization adjusts the input data of the vector X^i by putting them all on the same scale and thus can be compared.

$$X^i_{\text{standardized}} = \frac{X^i_{\text{initial}} - \min(X^i_{\text{initial}})}{\max(X^i_{\text{initial}}) - \min(X^i_{\text{initial}})} \tag{23}$$

Since a relatively significant randomness factor leads ML models and DL networks, once the networks and models are built, they are calibrated to reduce the randomness profile as much as possible. The calibration is done during the in-sample period to expect to obtain the best possible results during the out-sample period by optimizing the hyperparameters of the structure of the models. For the ML model, a calibration using a cross-validation method is implemented with an objective metric to improve the model accuracy. Several hyperparameters of the model are tested during this model calibration. For DL models, we calibrate the network hyperparameters using Bayesian optimization. This optimization method is ideally suited for expensive models in learning time. As an objective metric, a Sharpe ratio function is used. These calibration methods are discussed more in detail in the presentation of the models. Finally, when each model is optimally calibrated, we test them on the out-sample period. These tests are considered as the basis for the analysis of this research work and portfolio protection.

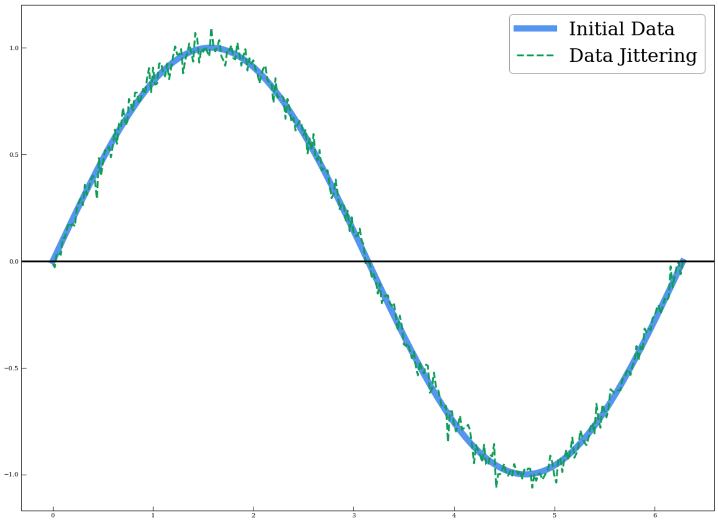

Figure 4.2 – Data jittering example

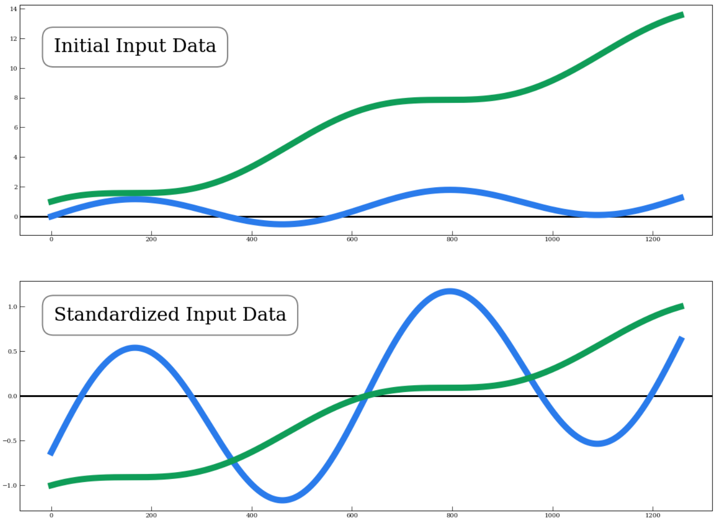

Figure 4.3 – Data standardization example

4.2.6 Support vector regression model

The "Support vector machine" (SVM) is a supervised machine learning model that can be used for regression and classification. The principle of the model was first introduced by Vapnik (1998) under the Vapnik-Chervonenkis theory. In general, the SVM model allows, among many other specificities, to perform classifications on non-linear data. In this case, we use the form of SVM applied for the regression realization, called "Support Vector Regression" (SVR). The SVR model was first introduced by Smola and Schölkopf (2004) and is based entirely on the SVM model. To understand the concept of SVR more efficiently, we should describe the construction of simple linear regression.

In a simple linear regression such as an OLS (Ordinary least squared), a vector of data inputs and a vector of data outputs are necessary. The objective is to find a vector of weights for the input data to reduce the error of the difference between the input value multiplied by the weight coefficient and the output values. More formally, the OLS can be written in formula (24).

$$\text{minimize:} \quad \sum_{i=0}^{n} (y_i - w_i x_i)^2 \tag{24}$$

where y_i is the output vector, x_i is the input vector and w_i is the coefficient vector.

With this simple case, the vector xi could be the data set corresponding to the RSI or MACD indicator, and the vector y_i would correspond to the buy or sell signal linked to the technical indicator or the sign of the daily returns.

However, this work aims to describe support vector regression. Even if, for the design, the SVR strongly appears like an OLS, the SVR allows to perform non-linear regressions on one side and to consider only the training data that respect a precise tolerance threshold on the other side. This tolerance threshold is accompanied by introducing a variable called "Slack" to cooperate with the infeasible constraints of the optimization problem. The SVR regression model, as presented in the research paper by Smola and Schölkopf (2004), is constructed as follows:

$$\text{minimize} \frac{1}{2} \|w\|^2 + \mathrm{C} \sum_{i=1}^{l} (\xi_i + \xi_i^*) \tag{25}$$

$$\text{subject to} \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon + \xi_i \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \quad \geq 0 \end{cases} \tag{26}$$

$$\text{and with,} \qquad w = \sum_i \alpha_i \cdot y_i \cdot k(x_i, x) \tag{27}$$

In the model presented above, w is the weights coefficient vector, x_i is the input values vector, y_i is the output values vector, b is the intercept of the regression. ε is the precision or error tolerance of the regression. $\xi_i$is the so-called "slack variable" of the model. ||•|| is the Euclidian norm l2, <•, •> is the scalar product. C is a constant that should be greater than 0 and determine the tradeoff between the flatness of the regression and the amount up to which the deviations larger than epsilon are tolerated. The kernel function is given by k. Finally, α_i is a dual variable**,** helping for the transformation of the space from a linear to quadratic optimization.

The particularity of the SVR model is its adaptation with the kernel function. This kernel function modulates the SVR model to adapt from linear to non-linear regression quickly. Each of them are included in the linear SVR model by changing the scalar product in the optimization problem by another kernel function. The different existing kernel functions are given later and described in Appendix B.4.

As for any regression model, two steps are essential for an SVR trading model, the training and the test sets. The training set corresponds to the in-sample period of the dataset, and the test set corresponds to the out-sample period of the dataset. During the in-sample training period, the model is fitted with vectors corresponding to the different technical indicators with their optimal parameters for each indicator (found in the technical indicators part) as input vectors. For the output vector, a data vector corresponding to the daily returns as a signal is given, i.e., if the returns are negative, the value is 0, and if the returns are positive, the value is 1. Once the model is fitted, we run the output prediction for the out-sample period.

However, as presented in the model, several hyperparameters are considered when designing an SVR model. Indeed, to calibrate the model as accurately as possible according to the hyperparameters, the cross-validation (CV) method is used with the ***GridSchearchCV*** functionality from the `Sklearn` python library. Therefore, a k-folds CV method is used. This method consists in splitting the training dataset into k subsets (also called "folds") of the same size. Then, on these k subsets, the different hyperparameters of the model are tried, and a score is given for each of these hyperparameters combination and subsets.

There are as many iterations in the k-folds CV process as hyperparameter combinations possible. Then for every hyperparameter combination, an average score of the splits is computed to determine the optimal hyperparameters for the model. These optimal hyperparameters are used for the out-sample test period as the retained model. The following diagrams allow a better understanding of a k-folds CV process.

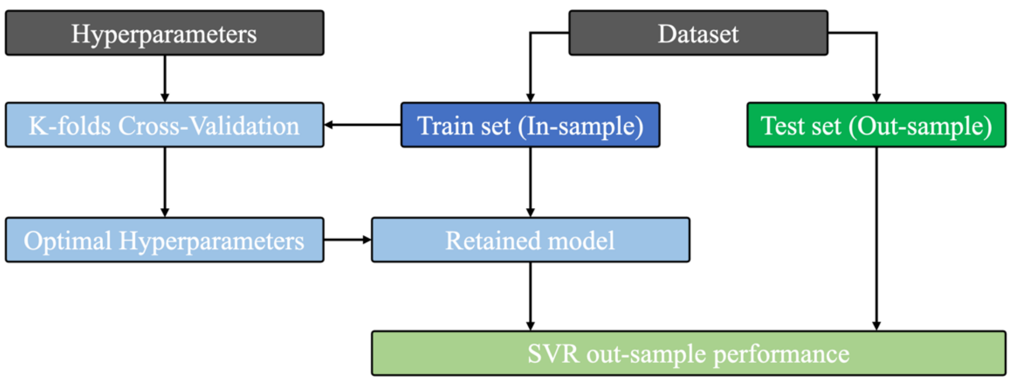

Figure 4.4 – K-fold cross validation entire process

With the SVR model, several hyperparameters that are summarized in Table 4.4 are tested.

Hyperparameter	Hyperparameter range of variation
Kernel function[8]	[Gaussian (RBF), Sigmoid]
Regularization parameter	[1, 10, 100]
Tolerance parameter	[0.001, 0.01, 0.1]

Table 4.4 - SVR In-sample hyperparameters variations

Ten-fold cross-validation is initiated, and use the model accuracy as an objective function between folds to find optimal params to fit the model. Once hyperparameters are found, they are used to fit the model within the in-sample dataset.

The SVR being a regressor model, signal predictions are found by using the fitted model according to the SVR model implemented. The result of the model predictions gives a value between 0 and 1, depending on the regression. So, the signal is equal to 1 if the prediction is equal to or over 0.5. The signal is equal to 0 if the output prediction is between 0 and 0.5. In that way, the results of the SVR strategy are obtained.

[8] Kernel function: Description of each kernel in Appendix B.3

4.2.7 Artificial neural network

An "Artificial Neural Network" (ANN) is a deep learning algorithm belonging to the family of deep learning algorithms. Indeed, they use several elements in common with machine learning algorithms. However, the ANN structure is different and more elaborate than machine learning algorithms such as SVM. First, it is essential to understand the forward propagation of a neural network. Such as a simple regression, a neural network is composed of an input layer and an output layer. Input and output layers are structured of only one dense. Between these two layers, the hidden layer is induced. The hidden layer can be composed of several dense. The forward propagation structure of the neural network means that the direction goes from the input layer and is then processed in the hidden layers of neurons, and finally, the result is processed in the output layer. This structure is equivalent to linear regressions with network configuration, as shown in Figure 4.5. The structure in layers also gives the name of "Multilayer perceptron" to ANN models.

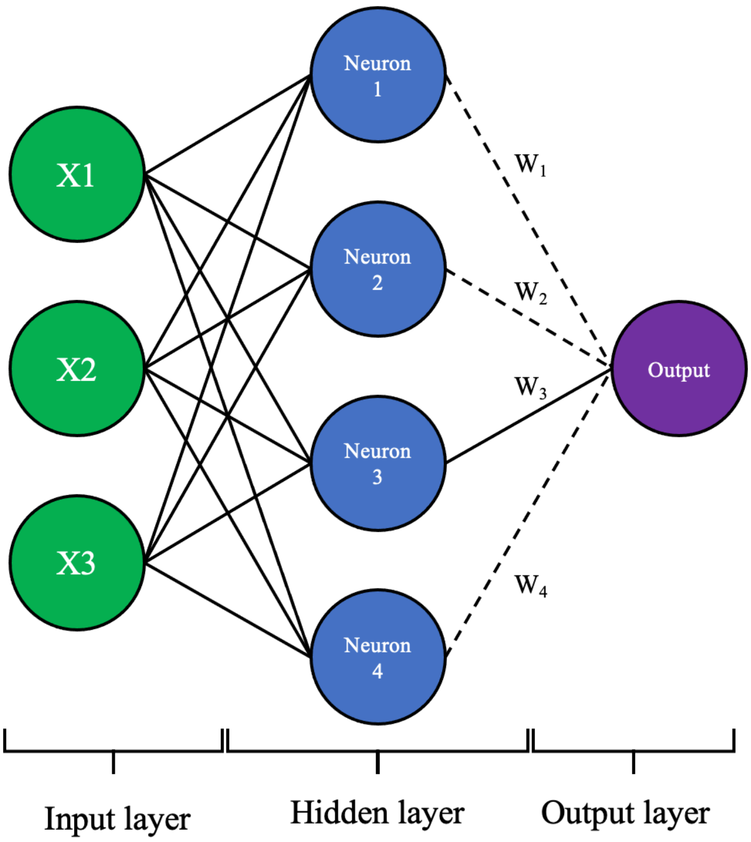

Figure 4.5 – Multilayer perceptron (ANN) structure

The neurons of the hidden layer are constructed like modified linear regressions. Each neuron (or "cell") represents a linear regression. Indeed, a neuron receives input values and returns an output value which is found using the weights assigned to the cell's input. Initially, the neural network weights are determined randomly in the input layer, and therefore, it can give fundamentally different results for two identical networks. From a theoretical point of view, a neuron can be summarized by the formula (28).

$$\sum_{i=1}^{n} x_i w_i + B = y \tag{28}$$

where x_i is the i-th input vector, w_i is the i-th input vector weight, B is the bias of the regression y is the output of the cell.

To find the optimal weights and thus allow the output layer to read the results produced by each neuron, the neurons are constructed with what is called an activation function. An activation function contains the results of the neuron's regression between limits of values, and it allows the outputs to have a distinct similarity in their reading. In the artificial neural network, activation functions, the Relu function (29), and the Linear function (30) are used. They depend on their place in the process of the ANN. The output produced by the activation function is then used to define whether a given neuron is accurate or not, causing activation or deactivation of a neuron in the concerned network. At the end of the hidden layer succession, the optimal weights are defined. Then, the model is fitted, and predictions can be made.

$$g(y) = \max\{0, y\} \tag{29}$$

$$g(y) = y \tag{30}$$

However, to adjust the choice of optimal weights, the neural network uses a loss function. This loss function is used in backpropagation, which means that the cost is estimated once each regression is determined. The choice of these weights is optimized thanks to gradient descent on this loss function. A "Mean Absolute Error" (MAE) loss function is used for this model. This function is chosen due to the neural network's structure and the activation functions implemented in the network. The main points for constructing an artificial neural network are defined, but there are still some essential hyperparameters that allow for obtaining better prediction results and reducing the randomness of the weight distribution.

Concerning the intern structure of the network, the number of neurons engaged for each hidden layer allows for specific cases to improve the network's learning. Indeed, the more neurons there are, the more the network observes different possibilities to adjust its learning. Always in the network structure, the choice of the optimizer that is decisive should be made for the network's accuracy. Then, the learning rate hyperparameter regulates the weights of the neural network with the gradient of the loss function, so this parameter is essential. Other hyperparameters external to the neural network structure exist. First, we have the number of epochs. It is the number of times the network is trained to converge towards coherence of the results obtained. Then, the other external hyperparameter is the batch size. It is the size of the dataset sent to the network for training. This last parameter is similar to the dataset split in the k-folds method.

A neural network is an expensive algorithm in terms of computation time. Indeed, optimizing its parameters is tedious and practically impossible if one had to test each hyperparameter. In this work, Bayesian optimization is used to find optimal hyperparameters. The Bayesian optimization was developed by Močkus (1975). It is an optimization that consists in testing the different parameters of the network without testing all the possibilities and using a black-box function that maps the results obtained at each iteration to detect which group of hyperparameters is optimal. To choose the best combination, Bayesian optimization uses a reward function as an objective function. In this case, it is used as reward function, a function computing the Sharpe ratio associated with the result of the model training. This optimization function is merely written as:

$$\max_{x \in A} f(x) \tag{31}$$

where A is the set of hyperparameters tested, and $f(x)$ is the Sharpe ratio objective function

As hyperparameters, the hyperparameters mentioned in Table 4.5 are chosen. Once the optimal hyperparameters are found after running *200 iterations* of Bayesian optimization, the final hyperparameter structure of the artificial neural network is obtained.

Hyperparameter	Hyperparameter range of variation
Optimizer	[Adam, SGD, RMSprop, Adadelta, Adagrad, Adamax, Nadam, Ftrl]
Neurons	[1, 2, 3, …, 98, 99, 100]
Batch size	[200, 201, 202, …, 998, 999, 1000]
Epochs	[20, 21, 22, …, 198, 199, 200]
Learning rate	[0.01, 0.02, 0.03, …, 0.98, 0.99, 1]

Table 4.5 - ANN In-sample hyperparameters variations

In this work, an artificial neural network is used to create a trading signal. As announced at the beginning of this section, the model is trained with the three technical indicators constructed previously, in coordination with their synthetic data built with the data augmentation method. This data set then constitutes the input data of the model. For the outputs to be predicted, it corresponds to the vector of daily returns transformed into 1 or 0 depending on their sign. Like the SVR model, we have a training frame of the network corresponding to the in-sample period and a test frame corresponding to the out-sample period. The signal produced by the ANN model is similarly found as the SVR model because of the regression output of the ANN model.

Despite the Bayesian optimization, the randomness of the ANN remains present. Therefore, 50 different ANN models are trained over the in-sample period. Then, the best network among those created during this in-sample period is chosen as a model for the out-sample period prediction. Concerning the specificities of the implementation, `TensorFlow`, `Keras API`, and `Optuna` python libraries are used to construct the ANN model and the Bayesian optimization.

4.2.8 Recurrent neural network model

After implementing an artificial neural network, it is easier to understand the intuition behind the construction of a "Recurrent Neural Network" (RNN). Indeed, similarly to the ANN, the RNN is composed of three layers essential to its network structure. First, the RNN is composed of an input layer, then the hidden layer, which is constituted of the neurons of the neural network, and finally, the end of the network is the same as for an ANN since it is the output layer. There are no significant elements to see the differences between an ANN and an RNN.

Indeed, the difference with the ANN is in the links of the network structure. In an RNN, the cells of the hidden layer are connected, as shown in Figure 4.6. In an RNN model, the previous neuron outputs are used for the following neuron input. Nevertheless, it is essential to notice that the information from the previous cell retains its hidden status, i.e., it is not formatted as an output.

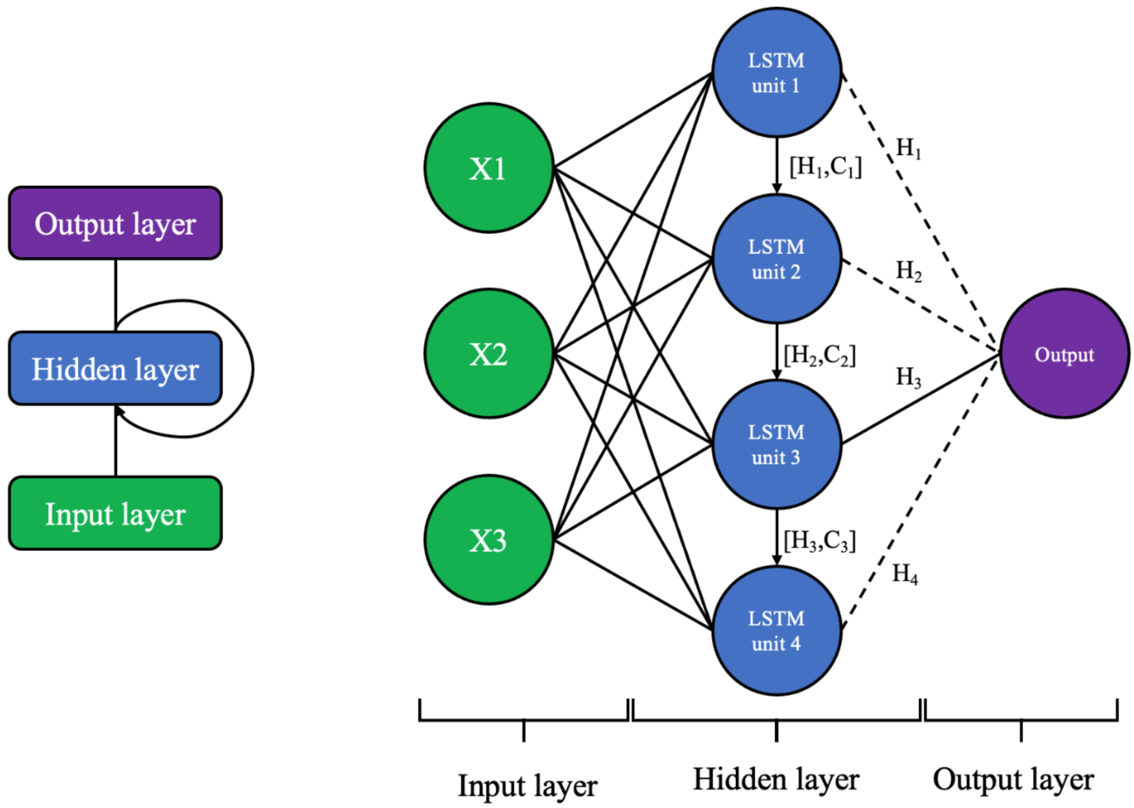

Figure 4.6 – Recurrent neural network simplified and general structure

To summarize, each cell of the hidden layer is connected cell by cell and layer by layer, which allows them to use the previous neuron information, so the main difference with an ANN is in the conception of the neurons that are not only simple regression with one input and one output. This architecture provides better accuracy in general. Because of this organization, the loss function of an RNN is calculated at all stages of the network. It uses backpropagation, which is done at each step, i.e., at each output of the cells of the hidden layer. Contrarily to the ANN, where the backpropagation is established at the end of the entire network, the RNN suggests the cost evaluation of the loss function at each step.

The global structure of the RNN having been defined, let describe the structure of a cell in the hidden layer. To implement the RNN, Long Short-Term Memory (LSTM) units are chosen for this work. This type of cell is used because the LSTM makes it possible to eliminate the issues of "gradient vanishing," generally present in the RNNs, mainly due to their interconnected structure and primarily because of the backpropagation carried out at each stage of the hidden layer. Moreover, contrary to other RNNs, the LSTM has a long-term memory, which gives a global vision of the results obtained during the process and thus activates or deactivates cells that are evaluated inefficiently to optimize the output. As shown in Figure 4.7, an LSTM unit has its activation functions allowing it to evaluate the allocation of weights to the input.

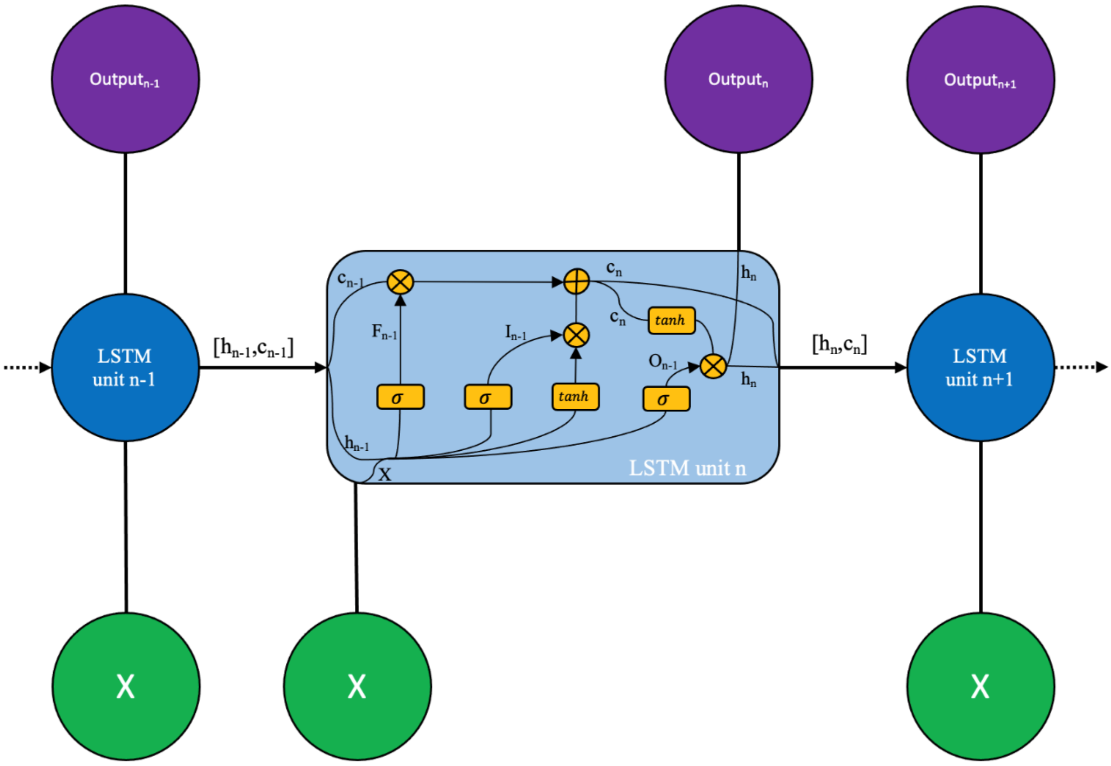

Figure 4.7 – LSTM unit intern structure

The architecture of an LSTM unit is composed of several formulas to produce the output from the input. The following formulas give the mathematical structure of an LSTM unit:

$$F_t = \sigma_g\left(W_f x_t + U_f h_{t-1} + b_f\right) \quad (31)$$

$$I_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \quad (32)$$

$$O_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \quad (33)$$

$$\tilde{c}_t = \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \quad (34)$$

$$c_t = F_t \circ c_{t-1} + I_t \circ \tilde{c}_t \quad (35)$$

$$h_t = o_t \circ \sigma_h(c_t) \quad (36)$$

where the different variables, operator domain, and name are described as follow:

$x_t \in \mathbb{R}^d$: input vector to the LSTM
$F_t \in (0,1)^h$: forget gate's activation vector
$I_t \in (0,1)^h$: input/update gate's activation vector
$O_t \in (0,1)^h$: output gate's activation vector
$h_t \in (-1,1)^h$: hidden state vector
$\tilde{c}_t \in (-1,1)^h$: cell input activation vector
$c_t \in \mathbb{R}^h$: cell state vector
$W \in \mathbb{R}^{h\times d}, U \in \mathbb{R}^{h\times h}, b \in \mathbb{R}^h$: weight matrices and bias vector parameter
σ_h, σ_g : hyperbolic tangent and sigmoid activation function
d, h : number of input features and hidden units
and where $c_0 = 0, h_0 = 0$ and $\circ$ is the Hadamard product

The different components of the LSTM cell are defined. However, there are still two elements to be added to finish the description of the RNN already sufficiently described. Indeed, the activation and the loss functions of the output layer depend on the structure of the network. For the activation function of the output layer, a **sigmoid** function (37) is chosen, which is the most adapted for the model. For the loss function, the **Binary Cross-entropy** function is chosen. It is a fundamental loss function adapted for binary classification models, being the most adapted in the RNN configuration composed of LSTM unit and following a sigmoid activation function:

$$g(x) = \frac{1}{1 + e^{-x}} \tag{37}$$

For the ANN, the Bayesian optimization (31) is performed on the same hyperparameters. The neurons are, in this case, the number of LSTM units present per hidden layer. The epochs and the batch size are the same as the ANN. Also, the optimizer and the learning rate are chosen identically to the ANN. As an objective function, the score given by the Sharpe ratio is used. Once the optimal hyperparameters are found using *100 iterations* of Bayesian optimization, the final structure of the recurrent neural network is obtained. Its structure is given in the results part.

Hyperparameter	Hyperparameter range of variation
Optimizer	[Adam, SGD, RMSprop, Adadelta, Adagrad, Adamax, Nadam, Ftrl]
Neurons	[10, 11, 12, …, 98, 99, 100]
Batch size	[500, 501, 502, …, 998, 999, 1000]
Epochs	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Learning rate	[0.01, 0.02, 0.03, …, 0.98, 0.99, 1]

Table 4.6 - RNN In-sample hyperparameters variations

Once again, and similarly to the ANN, the RNN implemented produces a trading signal resulting from the decision between several technical indicators. Indeed, the input of the RNN is the values of technical indicators, namely the MACD, the RSI, and the EMV. Synthetic values of these indicators being data augmentation also feed the RNN model. We have a dataset to train the model corresponding to the in-sample period. Moreover, a second part of the data set is used to test the model, corresponding to the out-sample period. Like the SVR and the ANN, the trading signal produced by the RNN model is found similarly because of the regression output of the model. Despite the Bayesian optimization, the randomness of the RNN remains present as in the ANN case. Therefore, 50 different RNN models are trained over the in-sample period. Then, the best network among those created during this in-sample period is chosen as the model for the out-sample period prediction. Finally, the implemented model gives buy or sell signals of the risky asset. Concerning the specificities of the implementation, `TensorFlow`, `Keras API`, and `Optuna` python libraries are used to construct the RNN model and the Bayesian optimization.

4.3 Portfolio protection

4.3.1 General framework

Until now, we have seen several algorithmic methods to develop trading portfolios. They aim to capture the bullish and bearish tendencies of the financial markets on different timeframes and define the asset's momentum. Even though these technical indicators are designed to observe the same phenomenon, they are constructed differently, leading to their different interpretation. Indeed, Technical indicators are often used together to confirm strong market trends. However, the investor's decision-making capacity can be confronted with cognitive biases linked to his psychology that can alter his results. In this work, we have used modern decision-making tools.

The first part of this work allowed to learn how to implement tools to help read the financial markets gradually. These tools aim at strongly limiting risk-taking by targeting the financial market trends. However, these tools are not above all and in no case neutralize the risks. Indeed, the different indicators and decision-making models are continually exposed to risks of sudden market variations. Therefore, in the second part of this work, we focus on portfolio protection, which is a dynamic way of limiting risk exposure. More generally, we see if implementing portfolio protection can improve the risk management of a trading portfolio.

Several dynamic asset allocation solutions are explored based on existing portfolio protection strategies. The objective is the adaptation of these protection methods to the case of trading portfolios. Indeed, trading portfolios have a particular non-continuous investment structure when the risky asset is no longer invested due to a bear market trend. The riskless index is mostly invested at that moment, creating allocation disproportions. This is followed by a recovery period when most risky assets are reinvested, creating a disproportion. This problem should therefore be seen as an adaptation of the protection to a trading portfolio with uncertain allocations. To solve this issue of protection for trading portfolio, several dynamic protection strategies are implemented. On the one hand, the objective is to adapt protection methods to a trading portfolio, and on the other hand, to allow these solutions to bring risk reduction compared to the essential portfolio without protection. To simplify the study of the risks, the same metrics to analyze the portfolios as for the trading portfolio's part above are used.

First, an "Option-Based Portfolio Insurance" (OBPI) method is implemented. An adaptive protective put method is implemented according to a targeted risk level. Then adaptations of the classical dynamic protection method are implemented, with a "Constant Portfolio Protection Insurance" (CPPI) first, then a "Time-Invariant Portfolio Protection" (TIPP). Finally, a hybrid strategy between option-based and constant portfolio protection insurance (HOC) is implemented.

4.3.2 OBPI

The OBPI is a static portfolio investment strategy introduced by Leland and Rubinstein (1988) that uses options to manage the risks linked to a given portfolio. In this strategy. The use of options is intended to protect from the risk exposure associated with purchasing a risky asset. Initially, the OBPI uses options to guarantee a return on the portfolio once the option's maturity is reached. In this context, the OBPI makes it possible to strongly limit the risks linked to market volatility since, at the time the OBPI is set up, the investor is fully aware of the entire risk of loss to which he is exposed (excluding the risk linked to the OTC market). The OBPI is a protection method that can be used with different options strategies. For example, it can be implemented by the simple purchase of call options or with the implementation of a protective put strategy. Nevertheless, what intrinsically defines an OBPI is using options to ensure the portfolio's protection. In this work, and as presented in the research paper of Leland and Rubinstein (1988), a protective put strategy is implemented. We consider the risky asset S_t that corresponds to the S&P500 at time t and a European put option relative to the risky asset. We also note the put option's maturity T and strike K.

At the time of purchase, the value of this portfolio is equivalent to:

$$V_0^{\text{OBPI}} = \left(S_0 + \text{Put}(0, S_0, K) - \text{Put}(0, S_0, K)\right) = S_0 \tag{38}$$

At maturity, the value of the portfolio is:

$$V_T^{\text{OBPI}} = \left(S_T + \text{Put}(T, S_T, K) - \text{Put}(0, S_0, K)\right) \tag{39}$$

From this, the payoff of this portfolio can be deduced as:

$$\text{Payoff} = \left(\max(S_T, K) - \text{Put}(0, S_0, K)\right) \tag{40}$$

As formula (40) shows, the OBPI is an investment strategy that guarantees a chosen level of returns under the condition of paying a risk premium (the price of the put option at the time of its purchase at time 0). Therefore, the OBPI portfolio's payoff depends on the option price. Indeed, although an option has many advantages, its main disadvantage is its cost. Many factors determine the price of an option, such as the option's maturity, the chosen strike, or the underlying asset's volatility. Beyond what directly defines the option price, one can eventually imagine an adaptive OBPI strategy that aims to hedge all or part of the trading portfolio according to market trends, to reduce the cost of protection.

Because of the additional cost involved in purchasing these insurance products, an adaptive OBPI strategy is implemented. The principle consists of partially hedging the portfolio's risky assets during periods of low or medium market volatility. And on the contrary, to fully hedge the portfolio during periods of high market volatility. To do this, a risk budgeting method introduced in the paper of Albeverio et al. (2013), is used to adjust the portfolio allocations according to the market risk. This method sets a target volatility value $\overline{\text{V}}$, representing the objective of volatility on the overall portfolio (for example, the historical volatility of the risky asset used). Then, depending on the market volatility determined by the VIX, the portfolio protection allocations vary, as shown in formula (41).

$$\alpha_t^1 = min\left\{\frac{\overline{\text{V}}}{\left(\frac{\text{VIX}_{t-1}}{100}\right)}; 1\right\} \tag{41}$$

$$\alpha_t^2 = 1 - \alpha_t^1 \tag{42}$$

Finally, the value of the investment strategy at each time t can be written as:

$$V_t^{\mathrm{OBPI}} = \alpha_t^1 \times S_t + \alpha_t^2 \times \left(\max(S_t, K) - \mathrm{Put}(0, S_0, K)\right) \tag{43}$$

In this case of adaptation to a trading portfolio, put options are used with a maturity of 1 day. Although options are very expensive with this maturity, it constitutes the solution that provides the most flexibility for the portfolio protection context. Usually, an OBPI consists of guaranteeing a given return over a long period. However, this solution is static, i.e., no asset flows take place between the period [0, T]. This condition is therefore problematic since in this case, we aim to protect a dynamic trading portfolio. Hence, an adaptation of formula (43) is essential for the strategy to work well. The control of the amount of protection limits the additional costs generated by the constant protection of the portfolio, which would reduce the profitability of the portfolio. Every day that the risky asset is traded, the level of protection is determined according to the level of volatility of the day. In other words, the protective put solution is determined daily, with a specific amount of protection allocation. Concerning the choice of the European option strike, the level of volatility as shown in formulas (41) and (42) is used. Indeed, the strike depends on the variable α_t^2 (the safe weight allocation), multiplied by a scalar determined by the number of the existing range of options in the list. The only parameter that can be explored in this strategy is the target volatility $\overline{\mathrm{V}}$ of the allocation formula. The range of parameters explored are summarized in Table 4.7.

Parameter	Range of variation of the parameter
Target volatility	$\overline{V} \in [0.20, 0.25, \ldots, 0.40, 0.45]$

Table 4.7 - OBPI parameters variations

All the elements that allow implementing the strategy in continuous time are defined. However, an adaptation keeping the best points of the trading portfolio while ensuring protection is necessary. For this purpose, specific trading rules are followed. The trading portfolios are built with technical indicators designed to avoid downward market trends. We, therefore, decide to suspend the purchase of the risky asset during periods when the signals indicate to stop buying the risky asset. Also, the protection is not active during these periods. To maintain a non-zero performance during these periods, the riskless index is partially invested when the risky index is not fully invested. More formally, the trading rule is set up as shown below.

$$\text{If signal}_t = 1: \begin{cases} \alpha_t^1 \times S_t \\ \alpha_t^2 \times \left(\max(S_t, K) - \text{Put}(0, S_0, K)\right) \\ 0 \times B_t \end{cases}$$

$$\text{If signal}_t = 0: \begin{cases} 0 \times S_t \\ 0 \times \left(\max(S_t, K) - \text{Put}(0, S_0, K)\right) \\ (\alpha_t^1 + \alpha_t^2) \times B_t \end{cases}$$

This protection trading method is relatively basic but provides excellent flexibility and cost containment in using European put options to protect the portfolio.

4.3.3 CPPI

The CPPI is a dynamic portfolio management strategy designed by Perold (1986). Indeed, CPPI is a dynamic asset allocation method that consists of allocating a portion of investment capital to a risky asset and the remaining part of the capital to a riskless asset. The allocation proportions between risky and riskless assets are determined according to several points. First, the CPPI considers a reference interval that corresponds to the entire investment period of the portfolio, which is noted [0, T]. This timeframe is essential for the CPPI since it defines the elements of allocation decisions. Among these elements, the most important is the Floor (F_t). This is the threshold below which the portfolio is fully invested in riskless asset until the end of the strategy. The Floor is itself defined by a protection threshold (γ_T) and a theoretical risk-free rate (r). It is the value of the Floor and more precisely the Cushion (C_t) that define the level of investment in the risky asset and the riskless asset. The Cushion corresponds to the excess of the difference between the value of the CPPI portfolio and the Floor. This investment strategy also depends on a parameter associated with the risk preferences of the investor, the Multiplier (m). The Multiplier and the Cushion help to define the exposure level to the risky asset (E_t) and the remaining part is allocated to the safe asset (B_t). So, the higher the multiplier is, the more the investor can take risks, and inversely. As presented in the research paper of Perold (1986), the CPPI is formally defined as $\forall t \in [0, T]$:

$$V_t^{\text{CPPI}} = \theta_{t-1}^1 \times S_t + \theta_{t-1}^2 \times B_t \tag{44}$$

$$F_t = \gamma_t \times V_0^{\text{CPPI}} \tag{45}$$

$$C_t = \max\left\{V_t^{\text{CPPI}} - F_t, 0\right\} \tag{46}$$

$$E_t = m \times C_t = \text{m} \times \max\left\{V_t^{\text{CPPI}} - F_t, 0\right\} \tag{47}$$

$$B_t = V_t^{\text{CPPI}} - E_t \tag{48}$$

$$\theta_t^1 = \frac{E_t}{V_t^{\text{CPPI}}} \tag{49}$$

$$\theta_t^2 = \frac{B_t}{V_t^{\text{CPPI}}} \tag{50}$$

where, $\gamma_t = \gamma_T \times e^{-r\times(T-t)}$, is the rate of protected capital, and $V_0^{\text{CPPI}} = \theta_0^1 \times S_0 + \theta_0^2 \times B_0$ is the value of the CPPI at time 0, θ_t^1 and θ_t^2 are the allocation in risky and riskless assets.

In the implementation of this investment strategy, two parameters define the behavior of the strategy. First, the level of protection of the invested capital represented by γ_T, and secondly, the level of the multiplier m. These two parameters are constant throughout the CPPI and depend on the preference of the investor and, therefore cannot be determined arbitrarily. As far as the level of protection is concerned, a fixed level of protection at 100% is considered. The reason is evident, the out-sample period over which the results are observed is defined as approximately 21 years, so over such a long period, recovering at least the initial investment, if not more, is consistent. Then for the multiplier, levels of the multiplier are tested to try to understand if risk-taking is necessary in the case of protection of trading portfolios. Even if the objective of implementing portfolio protection aims to reduce the risk and the equity exposure strongly, different values of the multiplier are tested.

The CPPI is not constraint as the OBPI. Indeed, an expensive product to protect the portfolio is not included in the trading portfolio anymore. Theoretically, a risk budgeting solution is not necessary since the CPPI is a risk budgeting solution in itself. What causes issues in this case is, on the one hand, the number of portfolio rebalancing and, on the other hand, the coherent follow-up of the CPPI strategy allocations with the trading portfolio, i.e., the adaptation between CPPI and trading portfolio. Indeed, for the first point, it is evident that daily rebalancing is unfeasible because it would generate high costs and would not allow stabilizing the portfolio performance. This problem also comes partly from the fact that intraday trading is not applied. If this had been the case, we could have implemented a take profit and stop loss solution that would have stabilized the portfolio performance more efficiently with a daily rebalancing of the CPPI. Therefore, optimizing the rebalancing by limiting their number is essential. To do this, based on the daily allocations of the CPPI, the daily rebalancing of the initial CPPI is reduced to rebalance over longer periods, namely: δ ∈ [1 month, 3 months, 6 months, 12 months]. To define the allocation values, to go from daily values to 1 month, for example, we make a weighted average of the daily allocations of the previous month, which correspond to the monthly allocation for the following month, as explained in allocations formulas (51) and (52):

$$\beta_t^1 = \sum_{i \in \delta_m} \frac{\theta_i^1}{\delta} \tag{51}$$

$$\beta_t^2 = \sum_{i \in \delta_m} \frac{\theta_i^2}{\delta} \tag{52}$$

where δ_m is the m-th interval of rebalancing of size δ, such as there are $\frac{n}{\delta}$ intervals over the investment period, n being the total number of trading day, and $t \in \delta_m$.

Then the second point is the coherence between CPPI and the trading portfolio. The CPPI is an investment strategy shaped for dynamic management of risk exposure, i.e., risky assets. However, the hypothesis of a CPPI that would follow the movements of the trading portfolio would be subject to unjustified increases in riskless allocations. Indeed, during a non-trading signal from the indicator, the floor that continuously increases is closer to the level of the trading portfolio due to the stagnation of returns when the portfolio is not traded. Similarly, including the riskless asset in the trading strategy, the CPPI strategy would consider an increase in the return during periods of non-trading, i.e., periods of a downward trend. The CPPI would then naturally increase its exposure to risky assets during these critical periods. Therefore, and to be consistent with the trading portfolios, the CPPI strategy is used between the risky index (S&P500) and the riskless index (AGG), allowing to extract the ideal allocations to limit the portfolio's exposure to risky assets and then apply the allocations found to the trading portfolio. The trading portfolio protection is coming after the CPPI weights computation. These daily allocation values are adjusted to monthly or more extended periods, following the method of the previous paragraph. Table 4.8 show the variable exploration of the strategy implementation.

Parameter	Range of variation of the parameter
Rebalancing (months)	$\delta \in [1, 3, 6, 12]$
Multiplier (scalar)	$m \in [1, 3, 10]$

Table 4.8 - CPPI parameters variations

As with the OBPI strategy, the CPPI investment strategy is adapted to best fit the trading portfolio. Indeed, the objective is to provide the most efficient protection solution to the portfolio based on a trading investment strategy. As shown in the previous paragraph, the CPPI method extracts allocations in risky and riskless indices respectively β_t^1 and β_t^2. These allocations are used for trading portfolio protection. However, just like the OBPI, during periods when the trading signals indicate 0, the portfolio is suspended, and the investment is 100% in the riskless index. More concretely, the investment rule based on the signal is described below.

$$\text{If signal}_t = 1: \begin{cases} \beta_t^1 \times S_t \\ \beta_t^2 \times B_t \end{cases}$$

$$\text{If signal}_t = 0: \begin{cases} 0 \times S_t \\ (\beta_t^1 + \beta_t^2) \times B_t \end{cases}$$

4.3.4 TIPP

The TIPP is a protection strategy that was first introduced by Estep and Kirtzman (1988). This strategy is an improvement of the CPPI, but where the Floor is moving with the upward market trend. Contrarily to the CPPI, it allows for locking the profits made with this trading strategy during downward market trends. Regarding the computation of the TIPP, the structure is the same as for the CPPI, except that the computation of the Floor is different. The TIPP depends on the maximum level the portfolio has reached since the beginning of the implementation. This strategy is efficient in bumpy markets, and in comparison to the CPPI, the circumstance in which the CPPI portfolio would have benefited from an extreme increase, moving it gradually away from the Floor of the strategy. However, if the CPPI portfolio, which is highly exposed to risky assets, falls without any increase after. Then the only limit of the portfolio is the floor level, which is very low compared to the level previously reached by the portfolio. The CPPI is then losing a significant part of its benefits, up to the minimum guaranteed level established at the beginning of the strategy. Instead, the TIPP allows the Floor to follow the portfolio's evolution and, therefore, maintain the returns of the strategy in case of a market downturn. Concretely, the TIPP sees the introduction of a new variable in the strategy, the Ratchet Capital (RC). Ratchet capital records the highest level reached by the portfolio during its implementation. It is based on the ratchet capital that the Floor is determined as shown in (53):

$$V_t^{\text{TIPP}} = \theta_{t-1}^1 \times S_t + \theta_{t-1}^2 \times B_t \tag{53}$$

$$RC_t = \max\{V_t^{\text{TIPP}}, RC_{t-1}\} \tag{54}$$

$$F_t = \gamma_t \times RC_t \tag{55}$$

$$C_t = \max\{V_t^{\mathrm{TIPP}} - F_t, 0\} \tag{56}$$

$$E_t = m \times C_t = \mathrm{m} \times \max\{V_t^{\mathrm{TIPP}} - F_t, 0\} \tag{57}$$

$$B_t = V_t^{\mathrm{TIPP}} - E_t \tag{58}$$

$$\theta_t^1 = \frac{E_t}{V_t^{\mathrm{TIPP}}} \tag{59}$$

$$\theta_t^2 = \frac{B_t}{V_t^{\mathrm{TIPP}}} \tag{60}$$

where $\gamma_t = \gamma_T \times e^{-r\times(T-t)}$ is the rate of protected capital, and $V_0^{\mathrm{TIPP}} = \theta_0^1 \times S_0 + \theta_0^2 \times B_0$ is the value of the TIPP at time 0; θ_t^1 and θ_t^2 are the allocation in risky and riskless assets.

As shown above, the difference between CPPI and TIPP is discreet since it does not seem to bring a significant change. We, therefore, try to compare its performance with that of the CPPI to see if these modifications bring, in this case, more efficient protection to the portfolio.

For the adaptation of TIPP to the trading portfolio, the same process as for CPPI is followed. Indeed, the TIPP strategy is applied using the S&P500 as a risky index and the AGG as a riskless index. In this way, and as for the CPPI, the ideal allocations of the two indices mentioned above are extracted to limit the risk exposure of the portfolio. To limit the costs caused by too frequent rebalancing, the method consisting in performing the weighted average of the allocations over the past time interval is used to define the allocations for the next time interval, as already described in formulas (51) and (52). The same rebalancing intervals as for the CPPI is studied, i.e., $\delta \in$ [1 month, 3 months, 6 months, 12 months]. Finally, like the CPPI, the behavior of the TIPP under several different multipliers is tested, i.e., m ∈ [1, 3, 10]. The TIPP is used as a comparison to see if improvements in the trading portfolio metrics are visible. Table 4.9 summarize the parameters exploration of the strategy.

Parameter	Range of variation of the parameter
Rebalancing (months)	δ ∈ [1, 3, 6, 12]
Multiplier (scalar)	m ∈ [1, 3, 10]

Table 4.9 - TIPP parameters variations

Again, the TIPP trading rule is identical to that of CPPI, depending on the trading portfolio and its buy or block buy signals as shown below:

$$\text{If signal}_t = 1\text{: } \begin{cases} \beta_t^1 \times S_t \\ \beta_t^2 \times B_t \end{cases}$$

$$\text{If signal}_t = 0\text{: } \begin{cases} 0 \times S_t \\ (\beta_t^1 + \beta_t^2) \times B_t \end{cases}$$

4.3.5 HOC

So far, several portfolio protection strategies have been built in dynamic settings. First, the OBPI was built with an initially static investment strategy but adapted to a daily trading portfolio framework. Indeed, we decided to protect a given percentage of the portfolio by buying several European put options according to a risk budgeting level and depending on the VIX level. This strategy allows, among other advantages, to efficiently protect the portfolio during a sudden negative market shock and control the protection costs. Then a CPPI was implemented, with a dynamic investment strategy consisting of rebalancing the weight of the portfolio allocations according to the portfolio value. To best fit the strategy to the trading portfolio, the ideal allocation weights were extracted from a CPPI between the S&P500 and the AGG and applied to the trading portfolio. Finally, a TIPP strategy that modifies the CPPI was implemented. This alternative to the CPPI allows blocking the performance of the portfolio. Indeed, with the TIPP, the floor is continuously revalued according to the maximum value the portfolio has reached, blocking possible losses at the time of fall of markets preceding a substantial rise.

Looking at these different portfolio protection strategies, we can easily see that they have points of compatibility that allow drawing on the strengths of each of them. To develop this innovative protection method, we inspired the strategy with the work of Di Persio et al. (2021). Indeed, they propose a hybrid strategy framework between the CPPI and the OBPI. The strategy presented in their paper consists in investing a given percentage in a zero-coupon bond to guarantee a terminal value. Then the remaining part is invested in a European option whose underlying is the risky asset, following CPPI allocations. However, the calculation of the CPPI includes a minimum level of exposure to the risky asset to avoid the possibility of cash-out in a classical CPPI strategy. Based on this work, a hybrid solution between the CPPI and the OBPI is proposed and adapted to the trading portfolio.

To construct this hybrid strategy, the precise compatibility between the CPPI and the OBPI must be defined. First, the OBPI that we have set up ensures continuous protection of the risky assets. The level of protection of the portfolio is determined daily using the risk budgeting method. The OBPI is adapted to the trading portfolio by investing totally in the OBPI when the trading signals are equal to 1 and totally in the riskless asset when the trading signals are equal to 0. From this point of view, it can be noted that the OBPI, i.e., the continuous protective put solution, can be considered as the risky asset. Then, the CPPI defined allocations between risky and riskless assets to limit the portfolio's risk exposure during bear market trends. In this case, the CPPI does not guarantee a terminal value because the riskless assets do not have a fixed return. Therefore, cash-out opportunities are not of primary use in this case.

By understanding these different elements that build the OBPI and CPPI, a hybrid version of these two strategies can be developed. Indeed, the OBPI can be used as risky assets in the adaptation between the CPPI and the trading portfolio. This modification considers the calculation method differently, impacting the allocations of the CPPI. Indeed, by taking up the idea of the paper written by Di Persio et al. (2021), a minimum exposure threshold β_{min}^1 to the risky asset is determined, preventing scenarios where the risky asset is totally invested as in (60).

$$E_t = \mathrm{m} \times C_t = \mathrm{m} \times \max\left\{V_t^{\mathrm{CPPI}} - F_t, \beta_{\min}^1 \times V_t^{\mathrm{CPPI}}\right\} \tag{60}$$

This modification of the minimum exposure to the risky asset in the CPPI calculation makes much sense since the portfolio benefits from the adaptive protection of the OBPI.

Finally, in the alternative solution proposed, which mixes OBPI and CPPI, the OBPI strategy is not modified and is used as a risky asset in the portfolio without modification in its computation. Then, for the CPPI, the same way as before is used, the only difference being its calculation method which includes a minimum exposure threshold to the risky asset in the allocation calculations.

The combination of these two strategies can create a problem for the analysis since it involves many parameters to be varied following the analysis proposed for both the CPPI and the OBPI. For the OBPI, we tried to analyze the level of protection applied to the portfolio as a function of market volatility. In this case, it is not possible to put aside this parameter for the analysis. Concerning the CPPI, we tried to vary two distinct parameters, the strategy multiplier and the occurrence of allocation rebalancing. Since the inclusion of the OBPI ensures a continuous level of protection for the portfolio, we can decide to put aside the study of rebalancing in this case of hybrid strategy. Indeed, the rebalancing implies a non-negligible increase in costs, which is why we set the rebalancing associated with the CPPI every 12 months. Finally, three distinct parameters are highlighted in the analysis. In the OBPI, the target volatility is evaluated. In the CPPI, the level of the multiplier as well as the new parameter of minimal exposure to risky assets are also evaluated, as shown in Table 4.10.

Parameter	**Range of variation of the parameter**
Target volatility (OBPI)	$\bar{\mathrm{V}} \in [0.25, 0.35, 0.45]$
Multiplier (CPPI)	$\mathrm{m} \in [1, 10]$
Minimum equity exposure (CPPI)	$\beta^1_{\min} \in [0.1, 0.3, 0.5]$

Table 4.10 - HOC parameters variations

As described above, the trading rules involve the combination of several elements seen before. The so-called risky asset is the protective put associated with the S&P500 index, and the riskless asset is always the AGG index. Then the investment proportions depend on both the OBPI with α^1_t and α^2_t in formulas (41) and (42). But also, with the proportions of the CPPI β^1_t et β^2_t introduced in formulas (51) and (52), as it is shown in the signal rules below.

$$\text{If signal}_t = 1: \begin{cases} \beta^1_t \left(\alpha^1_t \times S_t + \alpha^2_t \times \left(\max(S_t, K) - \mathrm{Put}(0, S_0, K)\right)\right) \\ \beta^2_t \times B_t \end{cases}$$

$$\text{If signal}_t = 0: \begin{cases} 0 \times \left(\alpha^1_t \times S_t + \alpha^2_t \times \left(\max(S_t, K) - \mathrm{Put}(0, S_0, K)\right)\right) \\ (\beta^1_t + \beta^2_t) \times B_t \end{cases}$$

Chapter 5 – Results

Following the methodology, the analysis of the results starts with the different trading methods implemented. Then in a second part, the results obtained during the implementation of the portfolio protection on these various trading strategies are analyzed. For the first chapter, i.e., the trading strategies, we always start with the optimal parameters found during the in-sample period, and we continue the analysis with the performance of these optimal parameters over the out-sample period. Finally, for the second part of the analysis of the results, a more quantitative approach is proposed, as it consists of testing several methods of portfolio protection on the optimal trading portfolios that we have found before. Several parameters are also tested to evaluate under which conditions an investor can reduce risk exposure and thus ensure the protection of his trading portfolio.

5.1 Trading portfolios results

5.1.1 MACD

For the MACD indicator, the variation of two parameters over the in-sample period was tested. These parameters correspond to the lookback rolling windows of the exponential moving averages of the MACD. In the structure of the MACD, these are the short-period and long-period windows. At the end of this analysis, Figure 5.1 shows that a number of these parameters obtain suitable results and belong to the same range of results. Indeed, as we can see on the heatmap of the combinations, the long-period window parameter should exceed 60 trading days, and the short-period window should not exceed 70 trading days, depending on the different cases. From this visualization, optimal parameters are obtained. The optimal short window is about 56 trading days, and the optimal long window is about 74. As shown in Table 5.2, over the in-sample period, the significance of these results is accepted with a p-value threshold of 10% and a Sharpe ratio of 0.685.

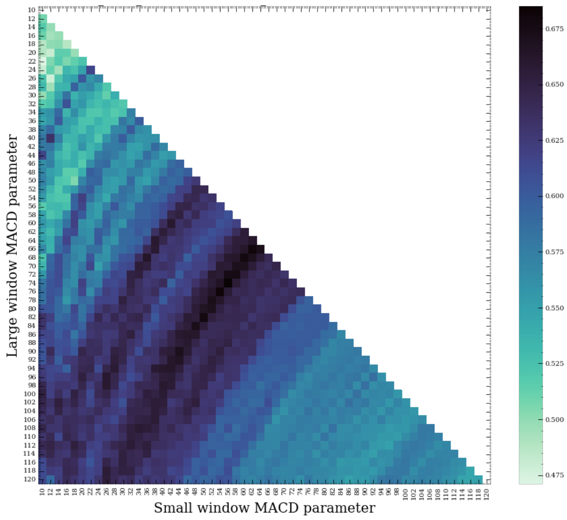

Figure 5.1 – In-sample parameters performance for MACD

Then, after testing these optimal parameters over the out-sample period, encouraging results were found. Indeed, as shown in Table 5.2, over the test period (out-sample), we were able to beat the market portfolio. Moreover, the risk-adjusted indicators also affirm that this strategy is very efficient as it is constructed. However, as shown in Figure 6.2 in Appendix A.7, we can still observe that in this configuration, the MACD does not control sudden bear market movements as in the period of the global crisis of the covid 19 pandemic in 2020. However, for the economic recession of the 2000s or the subprime crisis in 2008, MACD has proven to be very effective. More generally, the MACD is an efficient trading strategy that allows for excellent risk management.

5.1.2 RSI

For the RSI strategy, we varied a single parameter related to the lookback rolling window of the exponential moving average present in the structure of the RSI indicator. As we can see on the heatmap Figure 5.2, the results obtained over the in-sample period show that the optimal parameter is a window of 67 trading days, i.e., about three and a half months for the EMA parameter. Looking at this heatmap, we notice that this value of 67 days is in the center of a very efficient range of values among all those tested, suggesting a significant core of values over time.

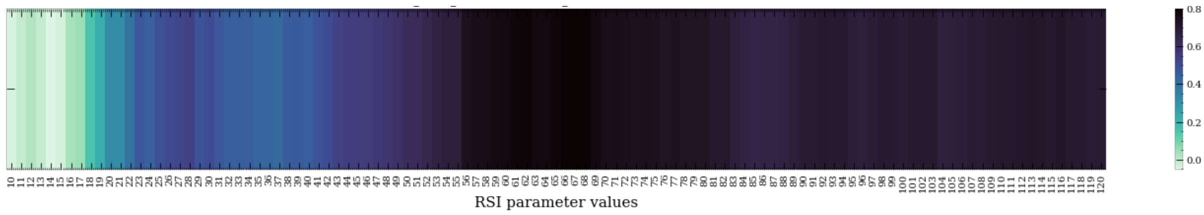

Figure 5.2 – In-sample parameters performance RSI

According to the t-statistic method obtained through the Sharpe ratio of the strategy, a significantly better result than for the MACD is obtained. The RSI strategy is statistically significant and is accepted at a p-value threshold lower than 1%. Indeed, with this strategy for an optimal parameter of 67 days for the exponential moving average, the Sharpe ratio is about 0.8, which is very high for a trading strategy like this.

When implementing the RSI over the out-sample period, the results were deceiving. Indeed, as shown in Table 5.2, over the out-sample period with the optimal parameter of 67 trading days in the EMA, the RSI does not beat the market portfolio over about 22 years of testing. The return barely exceeds 20%. On top of that, for most of this period, the return was negative (See Figure 6.4 Appendix A.7). The performance is disappointing and does not allow a suitable t-statistic. The risk performance is also deficient and shows that this strategy is unreliable. This strategy finally shows that past performance does not, in any way, reflect future performance. It also leads to the question of the importance of exploring protection methods to see if they can improve these results. Despite these poor results with the RSI, we can see later how a portfolio protection solution behaves and if it brings an improvement in the case of a non-performing portfolio.

5.1.3 EMV

To close the results obtained in the analysis of the technical indicators, let us see how the EMV behaves. Like the RSI, we varied a single parameter corresponding to the lookback rolling window period of the SMA included in the formula of the EMV. Over the in-sample period, we found 31 trading days for the rolling window as an optimal parameter. As we can see from the heatmap in Figure 5.3, this parameter does not appear to be part of a noticeable parameter group. Indeed, in this case, the parameters seem to be much more dispersed. For the parameter we found in the in-sample period, the Sharpe ratio of 0.667 is high, with a statistically significant p-value accepted at a threshold lower than 1% with the multiple testing methods.

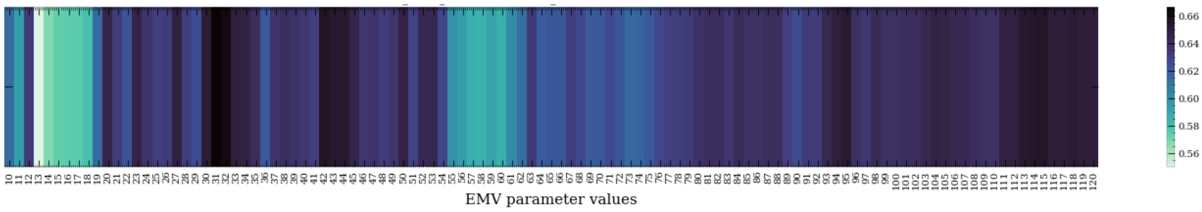

Figure 5.3 – In-sample parameters performance EMV

At the end of the test in the out-sample period, we could confirm that this strategy is indeed efficient. As Table 5.2 shows, the performance measures, as well as the risk measures, allow seeing that this strategy with this optimal parameter gives interesting results. The Sharpe ratio, as well as the Sortino ratio, are consistent, and the market volatility measures are well controlled. Contrary to the MACD, the EMV does not beat the market portfolio despite good results. Finally, the strong point of the EMV can be seen in Figure 6.6 (in Appendix A.7). During sudden market shocks, the EMV closed its buy positions effectively, for example, during the 2020 covid-19 pandemic period. However, during global bearish movements like in the 2000s with the 2000-2003 recession and the 2008 subprime crisis, the EMV does not consider large bearish market movements since buy signals are dominant. The EMV is, therefore, a relatively complementary technical indicator to the MACD.

	Ann. Returns	**Ann. Std. Dev.**	**Sharpe**	**Sortino**	**VaR**	**CVaR**	**MDD**	**t-stat**	**p-value**
S&P500	0.082	0.138	0.596	0.775	-0.013	-0.019	-0.356	–	–
MACD	0.080	0.116	0.685	0.742	-0.011	-0.017	-0.135	4.330	0.090
RSI	0.083	0.104	0.800	0.933	-0.010	-0.016	-0.107	5.060	0.000
EMV	0.089	0.134	0.667	0.874	-0.013	-0.019	-0.332	4.220	0.000

Table 5.1 - Technical indicators trading: In-sample results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD	t-stat	p-value
60/40 Portfolio	0.060	0.114	0.527	0.656	-0.010	-0.017	-0.352	–	–
MACD	0.069	0.136	0.506	0.538	-0.013	-0.021	-0.158	2.250	0.030
RSI	0.009	0.112	0.085	0.085	-0.012	-0.019	-0.664	0 .270	0.790
EMV	0.063	0.173	0.364	0.448	-0.018	-0.027	-0.587	1.340	0.180

Table 5.2 - Technical indicators trading: Out-sample results

5.1.4 SVR

Once we set up the three technical indicators and found their optimal parameters, we used them in the SVR Machine Learning algorithm. To calibrate the SVR model, a k-fold cross-validation method is used. This method highlights a selection of optimal hyperparameters for the SVR model during the in-sample period to obtain the best results in the out-sample period of application. As shown in Figure 5.4, the optimal parameters are ranked, one being the best and zero the worst, based on the accuracy of the predictions. Looking at this figure, a selection of optimal parameters stands out among all those tested. First, the optimal kernel function is the number 1, i.e., the Gaussian Kernel. Then the optimal epsilon error tolerance parameter is the highest, i.e., 0.1. Finally, the hyperplane deviation control parameter of regression C is optimal for the lowest value of 1.

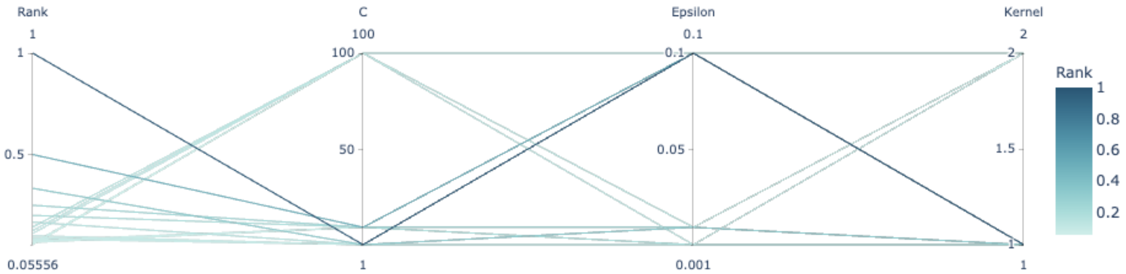

Figure 5.4 – Parallel coordinate plot of SVR k-fold cross-validation

Once these optimal hyperparameters were selected, we could test the implementation of this SVR model in the out-sample period of the dataset. The results found on this test were globally disappointing, as shown in Table 5.4. Indeed, despite the positive annualized returns, the risk measures of this trading portfolio in machine learning show many weaknesses. For example, a maximum drawdown of approximately -70% is obtained, far from that of the market portfolio. Then the Sharpe and Sortino ratios also show low values. However, the annualized volatility of the SVR portfolio is decent and close to that of the market portfolio, which is still encouraging.

From a more general point of view on this strategy, we notice that the SVR algorithm in this configuration does not capture well the information brought by the technical indicators. For example, and as shown in Figure 6.8 (in Appendix A.7), during the subprime crisis of 2008, several buy signals were issued but over periods that were too short, causing successive declines in the portfolio. The strong point of this algorithm is during periods of economic recession like the crisis of the early 2000s, where the blocks are effective on moderate and constant declines.

5.1.5 ANN

For the ANN, we also proceeded to a calibration of the hyperparameters of the network. This process is essential since it allows the ANN to be best adapted to the dataset that is used for training. However, the randomness of the ANN is in no way eliminated but is strongly attenuated since, at the end of the calibration, the hyperparameters selected are those with the best results. As shown in Figure 5.5, a plot of the optimal hyperparameters stands out. First, the most efficient optimizer is the 4, the Adagrad optimizer. Then the number of ideal neurons for each layer is around 20 and 50, and the learning rate is around 0.6. The number of epochs should be about 50, and finally, the batch size should also be close to the maximum, between 800 and 1000. With a selection of hyperparameters such as this, we could expect high results more often than with a random selection of hyperparameters.

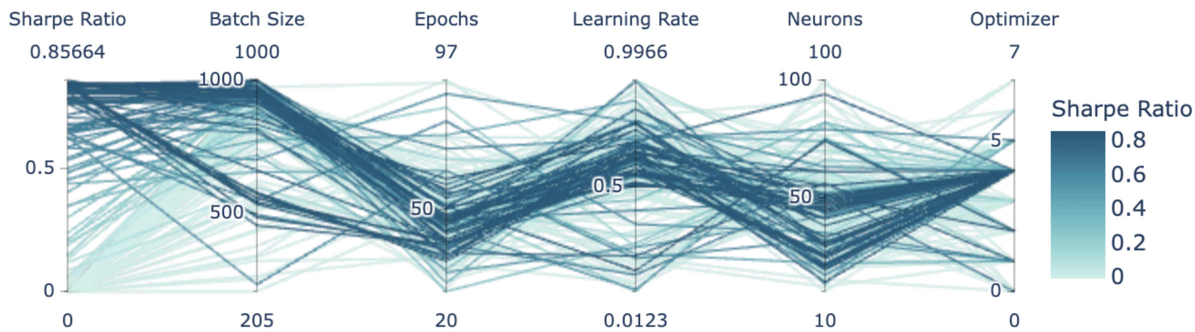

Figure 5.5 – Parallel coordinate plot of ANN hyperparameters Bayesian optimization

Once the optimal parameters were found using the Bayesian optimization, 50 networks were trained with the optimal parameters again, then selected the network giving the best Sharpe ratio in the in-sample period was considered the best-adapted network to the data set for the out-sample period. At the end of this process, and as shown in Table 5.4, the results obtained with this Deep Learning algorithm are satisfying. The annualized return is 8%, and the annualized volatility at 18% is good even if it is slightly above the market portfolio. For the risk measures concerned, they are globally interesting despite a high maximum drawdown of -63%.

We can see in Appendix A.7 with Figure 6.10 that ANN rarely makes extended periods of buy and hold decisions, but it still stands out from the market portfolio by often being above it and having the same sensitivity to economic market shocks.

5.1.6 RNN

For the RNN, the calibration, and the preparation of the network for the implementation, is similar to the ANN. Indeed, for the selection of the hyperparameters and as observed in Figure 5.6, the ideal optimizer is between the 3 and the 4, i.e., between Adadelta and Adagrad optimizers, the ideal number of LSTM units per layer is around 40. The learning rate should be between 0.4 and 0.6. Similarly, for the number of epochs, the higher, the better, i.e., 10 in this case. Finally, a high batch size is not recommended. A lot of different configurations among all the possibilities provide interesting results. In general, we notice that the selection of the optimal hyperparameter is like the ANN. However, the RNN model is different from the ANN for the results.

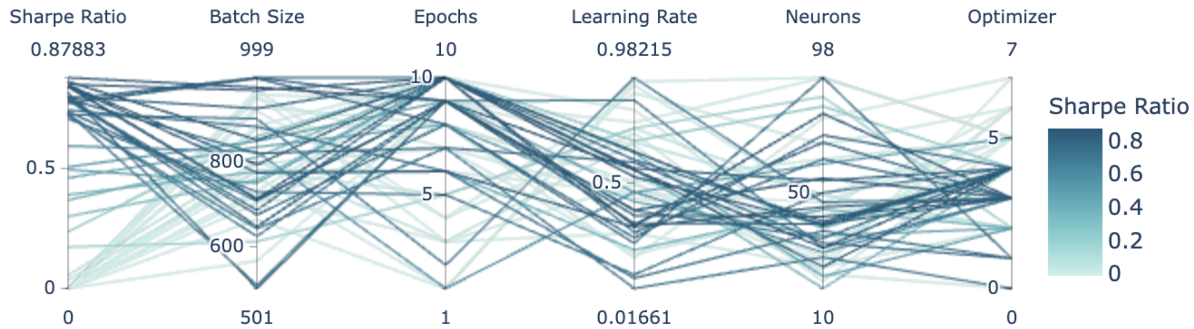

Figure 5.6 – Parallel coordinate plot of RNN hyperparameters Bayesian optimization

Indeed, as presented in Table 5.4, the annualized returns of the RNN are the second higher and close to the 60/40 portfolio with 6.6%. The volatility of the RNN is the lowest among the models of the trading portfolios, and the indicators of risk measures show that the RNN is well-performing. Indeed, the Sharpe and Sortino ratios are relevant and tend to be close to the 60/40 portfolio. We can, therefore, clearly conclude that the RNN Deep Learning model is the best of the three we implemented because it realizes the perfect trade-off between risk and returns. From a comparative point of view to the market portfolio, the RNN model follows the market trends but applies better-proportioned buy blocks than the ANN and the SVR, allowing it to obtain significant results and performances and should definitely beat the market in a condition of no transaction costs.

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD	t-stat	p-value
S&P500	0.082	0.138	0.596	0.775	-0.013	-0.019	-0.356	–	–
SVR	0.100	0.127	0.786	0.955	-0.012	-0.018	-0.194	6.175	0.000
ANN	0.088	0.136	0.649	0.828	-0.013	-0.019	-0.318	5.100	0.000
RNN	0.097	0.120	0.807	0.908	-0.011	-0.017	-0.152	6.341	0.000

Table 5.3 - Machine learning and deep learning trading: In-sample results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD	t-stat	p-value
60/40 Portfolio	0.060	0.114	0.527	0.656	-0.010	-0.017	-0.352	–	–
SVR	0.045	0.167	0.270	0.308	-0.016	-0.027	-0.684	2.123	0.034
ANN	0.079	0.179	0.440	0.541	-0.018	-0.028	-0.627	3.455	0.001
RNN	0.066	0.160	0.414	0.457	-0.015	-0.025	-0.284	3.251	0.001

Table 5.4 - Machine learning and deep learning trading: Out-sample results

5.2 Portfolio protection

The main objective of presenting the results for the protection part of the trading portfolio is to test the different trading portfolios for each protection method. In this way, we draw two conclusions. The first one is to know if the protections implemented are efficient and reduce the risk efficiently. The second is to highlight one method of portfolio protection that brings the best results. To best distinguish the different protection methods implemented, we use the different metrics presented previously in this work. All the results are available in Appendix A.1 to A.6 because of their number, and only the key metrics are given in the results part. Concerning visualization of the results, they are accessible on the Web App Dashboard presented at the end of the section.

5.2.1 MACD protection

First, for the OBPI protection applied to the MACD, we noticed right away that this is the protection strategy that maintains the highest levels of annualized returns for the targeted level of volatility. However, despite good overall volatility management, we notice that it turns out to be slightly higher than in the other protection methods. Nevertheless, the Sharpe and the

Sortino ratios are not impacted by it and provide for a target volatility of 25% parameter, better results than the 60/40 portfolio. Finally, we note that in terms of parameters, a target volatility level that is too low leads to poorer returns in general than when we set this variable at a higher level.

For the CPPI, we start to observe satisfying performance results with a Sortino ratio above 0.9 and a Sharpe ratio above 0.8. The returns are globally lower, but this can be explained by the cost of protection since the different volatility measures are low, showing good protection and, therefore, excellent risk management. However, even if they are lower, the returns are close to the 60/40 portfolio. The optimal parameters for the CPPI are a multiplier of 1 with an annual rebalancing.

Always for the MACD portfolio, the TIPP protection method gives the most impressive results. Indeed, as for the CPPI, with a multiplier of 1 and an annual rebalancing, we observe a Sharpe ratio approximately equal to 1 and a Sortino ratio of about 1.3. With the TIPP, risk management is slightly better than for the CPPI. However, the annualized returns are much lower for the TIPP. In this case of the TIPP, we note that the protection is efficient and brings to the trading portfolio a perfect trade-off between the returns, the risks, and the protection costs.

Finally, with the HOC, we notice that each method's positive points are put forward. Indeed, we see that we did not obtain the best results in terms of annual returns, but they are still globally acceptable and close to the initial MACD strategy. Similarly, the Sharpe and Sortino performance indicators are almost identical to those of the TIPP, and yet the HOC brings a much more interesting risk management, as shown by these risk indicators. However, one essential point is that the minimum equity exposure parameter must be greater than 0.3 to be effective. Indeed, between 0.1 and 0.3, the results of the strategies are the same. We can explain this phenomenon because the CPPI allocations never reach a value lower than 30% of the portfolio invested in the risky asset. The result could have been different for another portfolio with other assets, making this situation a particular case. Overall, the results are excellent for the whole strategy. We can say that among all the tested methods, the best one can indeed be the HOC for the MACD portfolio. The optimal parameters are a target volatility of 0.25, a multiplier of 1, and a minimum equity exposure of 0.3.

5.2.2 RSI protection

Of all the portfolios, the RSI trading strategy is the one that showed the most disappointing results. The basic RSI gave slightly less than 1% annual returns, proving to be problematic in ensuring portfolio protection that necessarily imposes a high cost. Indeed, in all the protection methods implemented, the RSI finally gave negative results for the Sharpe and Sortino ratios and the annualized returns. This result was close to zero but still disappointing. However, it is essential to mention that the protection methods work well because we notice a low volatility level and low VaR and C-VaR levels in every method. In this case, we cannot give a more efficient portfolio protection strategy than another because of these results. At the end of this case, it is essential to note that the trading methods implemented are not complete and should consider the possibility of a switch to negative performance and, therefore, a blocking of both the trading strategy and the protection that are too expensive. Thanks to the RSI, we can see that the protection of a portfolio combined with frequent trading is expensive and can lead to a fall in the performance of a strategy. It is worth noting that for the identical best parameters for the MACD protection, the RSI protection allows a stabilization with Sharpe and Sortino ratios very close to zero but negative. Another explanation for the poor protection is that trading is too frequent. Indeed, even if the protection mitigates the volatility of the portfolio in general, it is mainly the cost of trading that makes the performance of the portfolio fall, not allowing the protection to be correctly applied.

5.2.3 EMV protection

For the protection of the EMV technical indicator, we can remember that it was a moderately successful trading strategy, and the returns were globally efficient. However, the volatility and, therefore, the strategy's risk was relatively high, with a critical maximum drawdown.

With the implementation of the OBPI, we always obtain a varied profile depending on the parameters. Indeed, the higher the target volatility parameter is, the more the portfolio protection is reduced. However, in the case of the EMV, the Sharpe and Sortino ratios are the highest for high target volatility, i.e., 0.4 or 0.45 parameters. These parameter choices do not improve the performance ratios of the basic EMV, so we cannot say that the OBPI protection brings an improvement to the EMV portfolio.

The implementation of the CPPI improves the performance of the trading portfolio. Indeed, the Sharpe ratio is 0.39, and the Sortino ratio is 0.48 for the best parameter choice. This optimal choice in the CPPI for the EMV is a multiplier of 1 with an annual rebalancing. In addition to these performance improvements, we notice that with this specific configuration, the maximum drawdown achieved by the portfolio is about -35%, which is 20% lower than the initial EMV strategy. The annual returns are relatively low but reflect the cost of protecting the portfolio and remain positive.

For the TIPP, the optimal configuration is also with an annual rebalancing and a multiplier of 1. Compared to the basic strategy and the CPPI, the TIPP does not improve the portfolio's performance. Indeed, even if the volatility and the portfolio's global risk are significantly reduced, we notice that the returns are also strongly reduced with the TIPP. We can therefore conclude that TIPP is not a recommendable method of portfolio protection for the EMV and that CPPI is more efficient, even if the risk is well-carried by the protection method.

Finally, with the HOC strategy, we once again extract each previous method's strengths. Optimally, a multiplier of 1, target volatility of 0.35, and the minimum equity exposure between 0.1 and 0.3 give the same results because of the CPPI structure included in the HOC. With these parameters, the portfolio's performance is largely improved thanks to better risk management, i.e., the presence of low volatility. Even if the returns are annually lower than for the basic strategy, they are better than for the CPPI, and this decrease constitutes the cost of protection. Once again, this protection method is the best among all tested, and in general, all protection methods work and provide consistent protection of the trading portfolios.

5.2.4 SVR protection

The strategy based on the SVR Machine Learning model has presented correct results. We observe optimal parameters for the OBPI, with target volatility between 0.3 and 0.35. Thus, the performance of the SVR portfolio with the protection method is almost identical or even better for the Sortino ratio than the initial SVR portfolio. The Drawdown and the VaR or C-VaR metrics are relatively low but still better than the trading portfolio. For the SVR, we notice that a target volatility level that is too high for the OBPI does not guarantee better protection than the parameters found, and the same with lower target volatility parameters. Globally, the metrics of the protections are improved from the initial trading portfolio, but the results are not excellent.

For the CPPI, we still notice the same combination of optimal parameters, with a multiplier of 1 and an annual rebalancing of the CPPI weights. Despite this, the results are disappointing since the returns are largely reduced with barely 1% of annual returns. The same is true for the performance ratios, which are lower or equivalent to the basic SVR portfolio. However, the portfolio's volatility has been significantly improved, and the maximum drawdown has been decreased from the initial trading strategy for the best parameters found, which shows a slight efficiency.

As in all other strategies, the TIPP imposes a considerable cost of protection and less efficient allocations than the CPPI. Indeed, out of the 12 parameter combinations we tested, only the portfolio with a multiplier of 10 and an annual rebalancing allows for obtaining negative but the nearest to 0 performances. Therefore, we can understand the necessary compromise between the cost of protection and risk-taking that allows to obtain relevant results. This risk-taking implies an increase in volatility compared to the CPPI but is still lower than the initial SVR portfolio.

Finally, for the HOC, with a target volatility of 0.35, a multiplier of 1, and minimum equity exposure between 0.1 and 0.5, we obtain similar results as for the protection of other trading portfolios. Indeed, for a correct cost of protection, we obtain significantly improved performances with reduced volatility and lower returns than the basic strategy but positive in that case. Once again, this hybrid protection method positively improves the basic portfolio by making an efficient compromise between returns and protection costs.

5.2.5 ANN protection

With the protection of the portfolio built based on the ANN, we could confirm the various points previously notified. Indeed, for the OBPI, higher target volatility allowed to obtain better results. More precisely, with a target volatility of 0.35, the portfolio's overall performance is well-improved thanks to a slight decrease in the volatility of the portfolio and an increase in returns.

With the CPPI, we always obtain a clear improvement of the Sharpe and Sortino ratios even with the cost of the protection. This led to a slight decrease in returns but also implied a decrease in volatility from the initial ANN strategy. Here again, the optimal parameters are obtained with a multiplier of 1 with an annual rebalancing and provide satisfying results.
For the TIPP, we also note performances like the CPPI despite a decrease in annual returns but with a very low maximum drawdown, i.e., around 23%. We also get a low VaR and C-VaR compared to other protection methods. These positive results were found with a multiplier of 1, and a rebalancing triggered annually. So globally, the TIPP aims to significantly reduce the risks of the trading strategy and provide low but positive annual returns.

Finally, for the HOC, we found the strong points of each strategy again, allow for obtaining the best performances among all the implemented methods, i.e., a Sharpe ratio of 0.71 and a Sortino ratio of 0.94. And also, high returns at about 6% annually, which is very close to the basic ANN portfolio but with better management of the risks.

5.2.6 RNN protection

For the RNN, we obtained the same optimal protection parameters as the ANN trading portfolio and, generally, the same as the other portfolios. First, with the OBPI method, we also find that the target volatility of 0.35 improves the overall performance of the RNN trading portfolio while maintaining annualized returns close to the base portfolio, i.e., around 7%.

Then for the CPPI, we find a multiplier at 1 and a rebalancing of the CPPI weights annually as optimal protection parameters. This protection method strongly reduces the volatility of the trading portfolio while keeping annual returns above 4%, showing the robustness of this method at the end of all the trading portfolios we have tested. The performance of the portfolio is also improved mainly by this protection method. As for the TIPP, with a multiplier of 1 and a rebalancing every 12 months, we find a significant decrease in the portfolio's annual returns in exchange for a significant decrease in the strategy's risks.

Finally, for the HOC protection method, we also obtained the best results among all the methods tested previously, with a multiplier of 1, target volatility of 0.35, and a minimum equity exposure below 0.3. This way, we obtain a Sortino ratio of 0.91 and a Sharpe ratio of about 0.68. The risk-adjusted measure also shows the significant efficiency of this protection method.

The most important thing to remember about these different protection methods we have tested is the evidence that they reduce the risks incurred by the investment. The best strategies are the CPPI and the HOC. The most impressive is the implementation of the hybrid version. Indeed, it allows seeing that with the diversification of assets used to build a trading portfolio, we can both guarantee a level of protection by effectively reducing risks and at the same time maintain high returns allowing the strategy to beat the market. The results were consistent when looking at the Sharpe and Sortino ratios, which could, for some trading strategies, be higher than 1, which is impressive for a strategy including trading costs. Then it is also essential to notice that applying protection methods to trading portfolios is a complex task because, as we have seen in the case of the RSI when the trading portfolio is not efficient, and the cost of transactions occurs too often, it becomes impossible for the protection methods to catch up the loss.

5.3 Webapp dashboard

Implementing a web application dashboard proved to be the best solution to facilitate the communication of the results obtained in this work. Indeed, even in the absence of knowledge in algorithmic trading, any reader wishing to have the opportunity to access all the results obtained in this paper can be easily explored. The user can test investment strategies by himself within the limits of what the web application offers. Moreover, the format of the web application is an innovative solution allowing any user with an internet connection and support allowing access to a web browser to access the website. This support can be a computer, a tablet, or a cellphone.

The web interface is divided into two parts, as shown in the preview in Figure 5.7 and Figure 5.8. The first part consists of the elaboration of the trading portfolio. The user can select the different trading strategies in this paper. For the technical indicator trading strategies, the user can take control by testing his selections of parameters. This possibility of testing parameters is not available for machine learning and deep learning methods. The results of these trading strategies with a specific parameter selection over the in-sample period are observable in this part to guide the user on the optimal structure of the strategies. Metrics and technical indicators are given in addition to the performance of the strategies.

In the second part, the user can apply one of the four portfolio protections on the trading portfolio selected in the first part and test variations of protection parameters. It is important to note that for the trading portfolio based on technical indicators, their protection is realized on the optimal parameters found during the in-sample period of this research and not with the parameters selected in the first part of the dashboard.

The GitHub of this work is also available for any other information about using the web app. The technical implementation of the computational elements of this work paper are also available on it. Moreover, on this GitHub repository, the python notebook of this project hosted on Google Colab is available, and other elements such as the required libraries or the datasets used in this work are available on this repository.

Figure 5.7 - Web app dashboard overview (1)

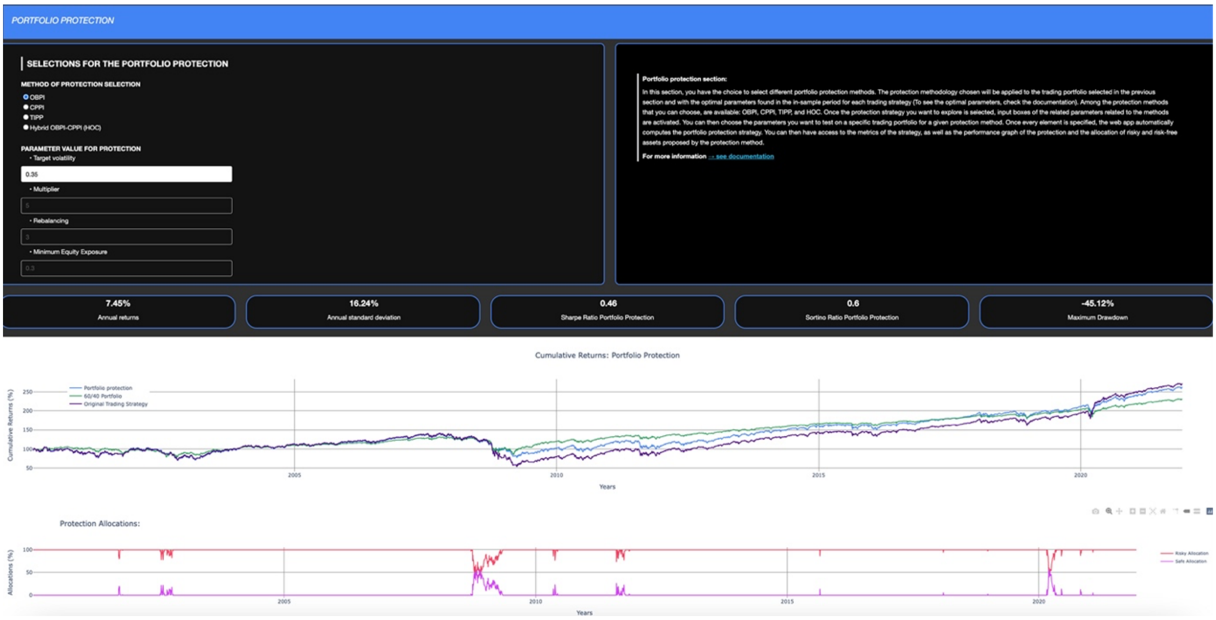

Figure 5.8 - Web app dashboard overview (2)

Dashboard link: http://algotradingportfolioprotection.pythonanywhere.com/

GitHub repository: https://github.com/FlorentFischer/TradingPortfolioProtection

Google Colaboraty Jupyter Notebook: Google_Colab

Chapter 6 – Conclusion

Security trading is one of the most well-known financial topics. It has been able to be expanded over the past 20 years by reaching a wider public than just financial professionals. However, risk management topics are less known despite their primary usefulness. Indeed, beyond the superficial aspect of trading securities with technical indicators, learning how to manage positions by applying effective risk management solutions is essential to avoid or limit heavy financial losses. With the example of the major crashes that took place during our investment period, we notice that for the recession of the 2000s or the subprime crisis of 2008, a well-calibrated technical indicator can limit the risks caused by these periods. However, dynamic risk management solutions for more sudden events such as the covid-19 pandemic in 2020 are practically the only ones that can provide risk mitigations.

Several findings allow learning more about algorithmic trading on the trading portfolio part. First, past performances do not reflect future performances in some cases, as with the RSI indicator. Indeed, despite a calibration including transaction costs and outstanding results during the in-sample period, the trading frequencies were too high in the out-sample period, and the strategies were ineffective, not allowing the RSI indicator to obtain satisfactory results. However, machine learning and deep learning models, trained based on these technical indicators, are well adapted to trading and can get impressive results thanks to their ability to work with time series. In any case, the different trading strategies can be impacted by high risk-taking in some cases, explaining the importance of trying to implement protection methods on trading portfolios.

For portfolio protection methods, we based on existing solutions such as OBPI, CPPI, TIPP, and HOC, and the most efficient modifications to adapt trading portfolios to ensure their protection was made. We started with the specific intuition that these solutions are costly and that different parameterizations can be studied depending on the profile of investors wishing to protect their portfolios. Indeed, such protection methods do not allow in any case to catch up with a loss-making trading portfolio and even decrease its performance. For already efficient portfolios, protection solutions slightly reduce performance but mitigate the risk remarkably.

Finally, these protection methods are effective for risk-averse investors and do not allow other types of investors to achieve higher performance than the risk-averse investor.

However, this study work is theoretical and makes specific assumptions that do not fully reflect an investment in real market situations. These limitations are mainly due to the quantity and quality of some data used in this work.

Regarding the quantity of data, the daily data is acceptable for this theoretical context. However, intraday data would be much better for a live trading environment since it would allow realizing trades closer to reality by including, for example, a stop loss or a trailing stop once the target return of the day is reached. This improvement does not consider the liquidity of the markets but should be considered.

For data quality, we modeled option prices using the Heston and Nandi model in this work. Despite the efficiency of this model and the satisfactory results we obtained, it would be preferable to use live market data. Access to this type of data is possible but difficult for long historical data length.

To conclude, algorithmic trading and portfolio protection are vast subjects that can be the purpose of numerous research papers based on the points discussed in this study. Indeed, the limitations of this work are easy to overcome approach these same elements from another point of view. In the same idea, we saw some of the existing trading portfolios and protections in this paper, but it should be noted that many other solutions and methods exist. For example, one could use dozens of other technical indicators to trade and select the most effective ones to improve the precision of predictions. Also, their optimal parameterization can be reviewed and improved. These same models can also be compared to others that may be more suitable for security trading. Finally, improvements in portfolio protection can be made. Diversification is possible by adding new securities for protection, notably by using other derivatives. Finally, hybrid protection solutions crossing the protection methods in another way is also achievable and would give other results. Globally, these few lines show how much this subject deserves to be reviewed from other points of view that could bring significant improvement to these topics.

References

Albeverio, S., V. Steblovskaya and K. Wallbaum (2013), "Investment instruments with volatility target mechanism", *Taylor & Francis, Quantitative Finance, 13*(10), 1519-1528.

Asness, C. S., T.J. Moskowitz and L.H. Pedersen (2013), "Value and momentum everywhere", *The Journal of Finance, 68*, 929-985.

Baz, J., N. Granger, C.R. Harvey, N. Le Roux and S. Rattray (2015), "Dissecting Investment Strategies in the Cross Section and Time Series."

Bertrand, P. and J.-L. Prigent (2001), "Portfolio insurance strategies: OBPI versus CPPI."

Byun, S. J. (2011), "A Study on Heston-Nandi GARCH Option Pricing Model", *International Conference on Information and Financial Engineering , 12*.

Chong, T.-L. T., W.-K. Ng, and K.-S. V. Liew (2014), "Revisiting the performance of MACD and RSI oscillators", *The Journal of Risk and Financial Management*(7), 1-12.

Di Persio, L., O. Immacolata and K. Wallbaum (2021), "Options on constant proportion portfolio insurance with guaranteed minimum equity exposure", *John Wiley & Sons, 37(1)*, 98-112.

Duan, J.-C. and J.-G. Simonato (1995), "Empirical Martingal Simulation for Asset Prices", *Centre interuniversitaire de recherche en analyse des organisation (CIRANO), 43*.

Engel, R., R. Ferstenberg and J. Rusell (2012), "Measuring and Modeling Execution Cost and Risk", *The Journal of Portfolio Management, 38*(2), 14-28.

Estep, T. and M. Kirtzman (1988), "TIPP: Insurance without complexity", *The Journal of Portfolio Management*(14(4)), 38-42.

Fons , E., P. Dawson, X.-J. Zeng, J. Keane and A. Iosifidis (2020), "Evaluating Data Augmentation for Financial Time Series Classification."

Goyal, A. and S. Wahal (2013), "Is Momentum an Echo?"

Harvey, C. R. and Y. Liu (2014), "Evaluating Trading Strategies", *The Journal of Portfolio Management, 40*(5), 108-118.

Heston, S. L. and S. Nandi (1997), "A closed-form GARCH option pricing model", *FRB Atlanta Working Paper, 97*(9).

Hochreiter, S. and J. Schmidhuber (1997), "*Long Short-Term Memory",* Massachusetts Institute of Technology.

Hull, J. C. (2014), "*Options, Futures, and Other Derivatives",* Pearson.

Leland, H. E. and M. Rubinstein (1988), "The evolution of portfolio insurance", *John Wiley and Sons*.

Li , Z. and V. Tam (2018), "A machine learning view on momentum and reversal trading", *MDPI Journal*(11), 170.

Lim, B., S. Zohren and S. Roberts (2019), "Enhancing time series momentum strategies using deep neural networks", *The Journal of Data Science* (1(4)), 19-38.

McQuinn, N., A. Thapar and D. Villalon (2021), "Portfolio Protection? It's a Long (Term) Story…", *The Journal of Portfolio Management*(47(3)), 35-50.

Močkus, J. (1975), "On Bayesian methods for seeking the extremum", *Optimization Techniques IFIP Technical Conference Novosibirsk*.

Perold, A. (1986), "A constant proportion portfolio insurance", *Harvard Business School (Unpublished manuscript)*.

Rohrbach, J., S. Suremann and J. Osterrider (2017), "Trend Following Strategies for Currencies and Bitcoin", *Zurich University of Applied Sciences*.

Shanker, M., M. Y. Yu and M. S. Hung (1996), "Effect of Data Standardization on Neural Network Training", *International Journal of Management Science, 24*(4), 385-397.

Sharpe, W. (1994), "The Sharpe Ratio", *The Journal of Portfolio Management, 21*, 49-58.

Smola , A. J. and B. Schölkopf (2004), "A Tutorial on Support Vector Regression", *Springer, Statistics and Computing, 14*, 199-222.

Sortino, F. A. and L. Price (1994), "Performance Measurement in a Downside Risk Framework", *The Journal of Investing, 3*, 59-64.

Vapnik, V. N. (2000), "*The Nature of Statistical Learning Theory",* Springer.

Welder, A. (1993), "*Trading for living",* New York: John Wiley & Sons Inc.

Welles Wilder, J. (1978), "*New Concept in Technical Trading Systems."*

APPENDIX A

A.1 – MACD Portfolio protection results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	0,060	0,114	0,527	0,656	**-0,011**	**-0,017**	-0,352
OBPI MACD(vtarget=0.2)	0,072	**0,112**	0,648	0,817	-0,012	-0,018	-0,124
OBPI MACD(vtarget=0.25)	0,086	0,120	**0,714**	**0,883**	-0,013	-0,019	**-0,120**
OBPI MACD(vtarget=0.3)	**0,088**	0,125	0,702	0,856	-0,013	-0,020	-0,121
OBPI MACD(vtarget=0.35)	**0,088**	0,127	0,689	0,835	-0,013	-0,020	-0,121
OBPI MACD(vtarget=0.4)	0,087	0,129	0,673	0,809	-0,013	-0,021	-0,121
OBPI MACD(vtarget=0.45)	0,086	0,131	0,656	0,780	-0,013	-0,021	-0,122

Table 6.1 - OBPI strategy on MACD trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	0,060	0,114	0,527	0,656	-0,011	-0,017	-0,352
CPPI MACD(m=1;reb=1)	0,004	0,077	0,053	0,063	-0,007	-0,012	-0,274
CPPI MACD(m=1;reb=3)	0,043	0,075	0,572	0,670	-0,007	**-0,011**	-0,130
CPPI MACD(m=1;reb=6)	0,053	0,074	0,719	0,840	-0,007	**-0,011**	-0,117
CPPI MACD(m=1;reb=12)	0,058	**0,073**	**0,801**	**0,935**	**-0,006**	**-0,011**	**-0,110**
CPPI MACD(m=3;reb=1)	0,021	0,118	0,178	0,202	-0,010	-0,019	-0,286
CPPI MACD(m=3;reb=3)	0,051	0,117	0,434	0,485	-0,010	-0,018	-0,192
CPPI MACD(m=3;reb=6)	0,057	0,115	0,498	0,551	-0,009	-0,018	-0,180
CPPI MACD(m=3;reb=12)	**0,062**	0,112	0,555	0,612	-0,009	-0,017	-0,171
CPPI MACD(m=10;reb=1)	0,004	0,119	0,033	0,037	-0,011	-0,019	-0,282
CPPI MACD(m=10;reb=3)	0,032	0,117	0,272	0,295	-0,010	-0,019	-0,203
CPPI MACD(m=10;reb=6)	0,038	0,115	0,334	0,359	-0,009	-0,018	-0,193
CPPI MACD(m=10;reb=12)	0,051	0,112	0,459	0,498	-0,009	-0,018	-0,178

Table 6.2 - CPPI strategy on MACD trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	0,060	0,114	0,527	0,656	-0,011	-0,017	-0,352
TIPP MACD(m=1;reb=1)	-0,007	0,048	-0,139	-0,183	**-0,005**	**-0,007**	-0,221
TIPP MACD(m=1;reb=3)	0,032	0,047	0,679	0,903	**-0,005**	**-0,007**	-0,061
TIPP MACD(m=1;reb=6)	0,042	**0,046**	0,903	1,206	**-0,005**	**-0,007**	**-0,053**
TIPP MACD(m=1;reb=12)	0,047	**0,046**	**1,008**	**1,350**	**-0,005**	**-0,007**	**-0,053**
TIPP MACD(m=3;reb=1)	0,008	0,091	0,084	0,103	-0,009	-0,014	-0,162
TIPP MACD(m=3;reb=3)	0,042	0,089	0,466	0,562	-0,008	-0,014	-0,124
TIPP MACD(m=3;reb=6)	0,051	0,088	0,581	0,693	-0,008	-0,014	-0,120
TIPP MACD(m=3;reb=12)	0,056	0,086	0,653	0,788	-0,008	-0,013	-0,120
TIPP MACD(m=10;reb=1)	0,047	0,117	0,400	0,485	-0,011	-0,019	-0,138
TIPP MACD(m=10;reb=3)	0,059	0,117	0,506	0,601	-0,011	-0,019	-0,124
TIPP MACD(m=10;reb=6)	**0,065**	0,115	0,563	0,665	-0,011	-0,018	-0,120
TIPP MACD(m=10;reb=12)	0,064	0,116	0,557	0,650	-0,011	-0,019	-0,120

Table 6.3 - TIPP strategy on MACD trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	0,060	0,114	0,527	0,656	-0,011	-0,017	-0,352
HOC MACD(vtarget=0.25;m=1;min_exp=0.1)	0,066	**0,063**	1,043	1,361	-0,006	-0,010	-0,055
HOC MACD(vtarget=0.35;m=1;min_exp=0.1)	0,066	0,067	0,991	1,251	-0,006	-0,010	-0,063
HOC MACD(vtarget=0.45;m=1;min_exp=0.1)	0,065	0,069	0,936	1,138	-0,007	-0,011	-0,090
HOC MACD(vtarget=0.25;m=1;min_exp=0.3)	0,066	**0,063**	**1,043**	**1,361**	**-0,006**	**-0,010**	**-0,055**
HOC MACD(vtarget=0.35;m=1;min_exp=0.3)	0,066	0,067	0,991	1,251	**-0,006**	**-0,010**	-0,063
HOC MACD(vtarget=0.45;m=1;min_exp=0.3)	0,065	0,069	0,936	1,138	-0,007	-0,011	-0,090
HOC MACD(vtarget=0.25;m=1;min_exp=0.5)	0,067	0,066	1,015	1,324	-0,007	-0,010	-0,057
HOC MACD(vtarget=0.35;m=1;min_exp=0.5)	0,067	0,070	0,966	1,220	-0,007	-0,011	-0,063
HOC MACD(vtarget=0.45;m=1;min_exp=0.5)	0,066	0,072	0,915	1,116	-0,007	-0,011	-0,090
HOC MACD(vtarget=0.25;m=10;min_exp=0.1)	0,065	0,098	0,669	0,791	-0,009	-0,016	-0,120
HOC MACD(vtarget=0.35;m=10;min_exp=0.1)	0,065	0,103	0,631	0,733	-0,009	-0,017	-0,121
HOC MACD(vtarget=0.45;m=10;min_exp=0.1)	0,063	0,108	0,584	0,658	-0,009	-0,017	-0,142
HOC MACD(vtarget=0.25;m=10;min_exp=0.3)	0,076	0,105	0,724	0,880	-0,010	-0,017	-0,120
HOC MACD(vtarget=0.35;m=10;min_exp=0.3)	0,077	0,111	0,692	0,826	-0,010	-0,018	-0,121
HOC MACD(vtarget=0.45;m=10;min_exp=0.3)	0,074	0,115	0,646	0,751	-0,011	-0,018	-0,133
HOC MACD(vtarget=0.25;m=10;min_exp=0.5)	0,079	0,110	0,723	0,891	-0,011	-0,018	-0,120
HOC MACD(vtarget=0.35;m=10;min_exp=0.5)	**0,080**	0,116	0,693	0,838	-0,011	-0,019	-0,121
HOC MACD(vtarget=0.45;m=10;min_exp=0.5)	0,078	0,119	0,652	0,768	-0,011	-0,019	-0,129

Table 6.4 - HOC strategy on MACD trading portfolio: results

A.2 RSI Portfolio protection results:

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	**-0,011**	**-0,017**	**-0,352**
OBPI RSI(vtarget=0.2)	-0,017	**0,108**	-0,160	-0,192	-0,012	-0,018	-0,776
OBPI RSI(vtarget=0.25)	-0,006	0,114	-0,049	-0,058	-0,012	-0,019	-0,694
OBPI RSI(vtarget=0.3)	-0,004	0,117	-0,031	-0,037	-0,012	-0,020	-0,712
OBPI RSI(vtarget=0.35)	-0,003	0,117	-0,024	-0,028	-0,012	-0,020	-0,711
OBPI RSI(vtarget=0.4)	-0,003	0,117	-0,022	-0,025	-0,012	-0,020	-0,711
OBPI RSI(vtarget=0.45)	-0,003	0,117	-0,022	-0,025	-0,012	-0,020	-0,711

Table 6.5 - OBPI strategy on RSI trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	**-0,352**
CPPI RSI(m=1;reb=1)	-0,065	0,069	-0,955	-1,158	-0,008	**-0,011**	-1,632
CPPI RSI(m=1;reb=3)	-0,029	0,067	-0,430	-0,514	**-0,007**	**-0,011**	-0,892
CPPI RSI(m=1;reb=6)	-0,019	0,067	-0,287	-0,340	**-0,007**	**-0,011**	-0,698
CPPI RSI(m=1;reb=12)	-0,015	**0,066**	-0,224	-0,266	**-0,007**	**-0,011**	-0,610
CPPI RSI(m=3;reb=1)	-0,057	0,097	-0,588	-0,679	-0,010	-0,016	-1,704
CPPI RSI(m=3;reb=3)	-0,030	0,096	-0,310	-0,353	-0,009	-0,016	-1,113
CPPI RSI(m=3;reb=6)	-0,025	0,095	-0,260	-0,291	-0,009	-0,016	-1,018
CPPI RSI(m=3;reb=12)	-0,020	0,094	-0,210	-0,236	-0,009	-0,016	-0,916
CPPI RSI(m=10;reb=1)	-0,074	0,092	-0,812	-0,907	-0,009	-0,016	-1,675
CPPI RSI(m=10;reb=3)	-0,049	0,089	-0,554	-0,609	-0,009	-0,016	-1,214
CPPI RSI(m=10;reb=6)	-0,046	0,091	-0,510	-0,548	-0,009	-0,016	-1,197
CPPI RSI(m=10;reb=12)	-0,033	0,092	-0,352	-0,384	-0,009	-0,016	-1,002

Table 6.6 - CPPI strategy on RSI trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	**-0,352**
TIPP RSI(m=1;reb=1)	-0,076	0,051	-1,501	-1,821	**-0,006**	-0,009	-1,685
TIPP RSI(m=1;reb=3)	-0,041	0,050	-0,821	-0,981	**-0,006**	**-0,008**	-0,910
TIPP RSI(m=1;reb=6)	-0,031	**0,049**	-0,636	-0,756	**-0,006**	**-0,008**	-0,722
TIPP RSI(m=1;reb=12)	-0,027	0,050	-0,548	-0,650	**-0,006**	**-0,008**	-0,646
TIPP RSI(m=3;reb=1)	-0,080	0,082	-0,971	-1,121	-0,009	-0,014	-1,748
TIPP RSI(m=3;reb=3)	-0,048	0,081	-0,588	-0,666	-0,008	-0,014	-1,073
TIPP RSI(m=3;reb=6)	-0,040	0,081	-0,498	-0,553	-0,008	-0,014	-0,935
TIPP RSI(m=3;reb=12)	-0,036	0,081	-0,438	-0,488	-0,008	-0,014	-0,865
TIPP RSI(m=10;reb=1)	-0,046	0,102	-0,448	-0,525	-0,011	-0,017	-1,005
TIPP RSI(m=10;reb=3)	-0,036	0,100	-0,361	-0,414	-0,011	-0,017	-0,897
TIPP RSI(m=10;reb=6)	-0,032	0,100	-0,322	-0,368	-0,011	-0,017	-0,920
TIPP RSI(m=10;reb=12)	-0,027	0,102	-0,268	-0,308	-0,011	-0,018	-0,858

Table 6.7 - TIPP strategy on RSI trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	**-0,352**
HOC RSI(vtarget=0.25;m=1;min_exp=0.1)	-0,010	**0,065**	-0,147	-0,176	**-0,007**	**-0,011**	-0,494
HOC RSI(vtarget=0.35;m=1;min_exp=0.1)	-0,008	0,066	-0,119	-0,141	**-0,007**	**-0,011**	-0,495
HOC RSI(vtarget=0.45;m=1;min_exp=0.1)	-0,008	0,066	-0,116	-0,138	**-0,007**	**-0,011**	-0,495
HOC RSI(vtarget=0.25;m=1;min_exp=0.3)	-0,010	**0,065**	-0,147	-0,176	**-0,007**	**-0,011**	-0,494
HOC RSI(vtarget=0.35;m=1;min_exp=0.3)	-0,008	0,066	-0,119	-0,141	**-0,007**	**-0,011**	-0,495
HOC RSI(vtarget=0.45;m=1;min_exp=0.3)	-0,008	0,066	-0,116	-0,138	**-0,007**	**-0,011**	-0,495
HOC RSI(vtarget=0.25;m=1;min_exp=0.5)	-0,009	0,067	-0,136	-0,163	**-0,007**	**-0,011**	-0,490
HOC RSI(vtarget=0.35;m=1;min_exp=0.5)	-0,007	0,069	-0,109	-0,129	**-0,007**	-0,012	-0,491
HOC RSI(vtarget=0.45;m=1;min_exp=0.5)	-0,007	0,069	-0,106	-0,126	**-0,007**	-0,012	-0,491
HOC RSI(vtarget=0.25;m=10;min_exp=0.1)	-0,022	0,093	-0,241	-0,269	-0,009	-0,016	-0,827
HOC RSI(vtarget=0.35;m=10;min_exp=0.1)	-0,020	0,095	-0,213	-0,236	-0,009	-0,016	-0,833
HOC RSI(vtarget=0.45;m=10;min_exp=0.1)	-0,020	0,095	-0,210	-0,233	-0,009	-0,016	-0,833
HOC RSI(vtarget=0.25;m=10;min_exp=0.3)	-0,013	0,101	-0,133	-0,153	-0,010	-0,017	-0,796
HOC RSI(vtarget=0.35;m=10;min_exp=0.3)	-0,011	0,103	-0,105	-0,120	-0,010	-0,018	-0,800
HOC RSI(vtarget=0.45;m=10;min_exp=0.3)	-0,011	0,103	-0,102	-0,117	-0,010	-0,018	-0,800
HOC RSI(vtarget=0.25;m=10;min_exp=0.5)	-0,011	0,105	-0,106	-0,124	-0,011	-0,018	-0,749
HOC RSI(vtarget=0.35;m=10;min_exp=0.5)	-0,009	0,107	-0,079	-0,092	-0,011	-0,018	-0,752
HOC RSI(vtarget=0.45;m=10;min_exp=0.5)	-0,008	0,107	-0,077	-0,089	-0,011	-0,018	-0,752

Table 6.8 - HOC strategy on RSI trading portfolio: results

A.3 EMV Portfolio protection results:

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	**0,114**	**0,527**	**0,656**	**-0,011**	**-0,017**	**-0,352**
OBPI EMV(vtarget=0.2)	-0,001	0,131	-0,011	-0,014	-0,014	-0,020	-1,009
OBPI EMV(vtarget=0.25)	0,028	0,148	0,192	0,253	-0,016	-0,023	-0,882
OBPI EMV(vtarget=0.3)	0,038	0,158	0,238	0,307	-0,017	-0,025	-0,856
OBPI EMV(vtarget=0.35)	0,042	0,164	0,256	0,326	-0,017	-0,026	-0,859
OBPI EMV(vtarget=0.4)	0,043	0,168	0,258	0,328	-0,017	-0,026	-0,854
OBPI EMV(vtarget=0.45)	0,044	0,170	0,256	0,322	-0,018	-0,027	-0,845

Table 6.9 - OBPI strategy on EMV trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	-0,352
CPPI EMV(m=1;reb=1)	-0,021	**0,083**	-0,250	-0,315	-0,009	**-0,013**	-0,737
CPPI EMV(m=1;reb=3)	0,018	**0,083**	0,219	0,273	**-0,008**	**-0,013**	-0,373
CPPI EMV(m=1;reb=6)	0,028	0,084	0,338	0,418	**-0,008**	**-0,013**	-0,349
CPPI EMV(m=1;reb=12)	0,033	0,084	0,394	0,486	**-0,008**	**-0,013**	**-0,344**
CPPI EMV(m=3;reb=1)	-0,021	0,123	-0,172	-0,208	-0,013	-0,020	-1,129
CPPI EMV(m=3;reb=3)	0,008	0,128	0,066	0,079	-0,013	-0,021	-0,840
CPPI EMV(m=3;reb=6)	0,015	0,133	0,114	0,136	-0,014	-0,022	-0,808
CPPI EMV(m=3;reb=12)	0,018	0,141	0,128	0,149	-0,014	-0,023	-0,846
CPPI EMV(m=10;reb=1)	-0,044	0,124	-0,354	-0,419	-0,013	-0,021	-1,041
CPPI EMV(m=10;reb=3)	-0,017	0,132	-0,133	-0,153	-0,014	-0,022	-0,906
CPPI EMV(m=10;reb=6)	-0,010	0,143	-0,072	-0,083	-0,015	-0,024	-0,896
CPPI EMV(m=10;reb=12)	0,005	0,145	0,034	0,040	-0,014	-0,024	-0,862

Table 6.10 - CPPI strategy on EMV trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	-0,352
TIPP EMV(m=1;reb=1)	-0,036	**0,058**	-0,627	-0,787	**-0,006**	**-0,009**	-0,805
TIPP EMV(m=1;reb=3)	0,002	**0,058**	0,035	0,044	**-0,006**	**-0,009**	-0,362
TIPP EMV(m=1;reb=6)	0,012	**0,058**	0,206	0,257	**-0,006**	**-0,009**	-0,338
TIPP EMV(m=1;reb=12)	0,017	0,060	0,280	0,347	**-0,006**	**-0,009**	**-0,333**
TIPP EMV(m=3;reb=1)	-0,046	0,110	-0,418	-0,502	-0,011	-0,018	-1,037
TIPP EMV(m=3;reb=3)	-0,012	0,116	-0,102	-0,120	-0,012	-0,019	-0,836
TIPP EMV(m=3;reb=6)	-0,002	0,122	-0,015	-0,017	-0,012	-0,020	-0,806
TIPP EMV(m=3;reb=12)	0,002	0,131	0,015	0,017	-0,013	-0,021	-0,835
TIPP EMV(m=10;reb=1)	-0,018	0,145	-0,126	-0,149	-0,015	-0,023	-0,951
TIPP EMV(m=10;reb=3)	0,001	0,150	0,005	0,006	-0,015	-0,024	-0,868
TIPP EMV(m=10;reb=6)	0,010	0,155	0,061	0,073	-0,016	-0,025	-0,894
TIPP EMV(m=10;reb=12)	0,016	0,158	0,103	0,123	-0,016	-0,025	-0,868

Table 6.11 - TIPP strategy on EMV trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	-0,352
HOC RSI(vtarget=0.25;m=1;min_exp=0.1)	0,031	**0,072**	0,434	0,571	**-0,008**	**-0,011**	**-0,339**
HOC RSI(vtarget=0.35;m=1;min_exp=0.1)	0,038	0,080	0,473	0,604	**-0,008**	-0,012	-0,340
HOC RSI(vtarget=0.45;m=1;min_exp=0.1)	0,038	0,083	0,462	0,581	**-0,008**	-0,013	-0,343
HOC RSI(vtarget=0.25;m=1;min_exp=0.3)	0,031	**0,072**	0,434	0,571	**-0,008**	**-0,011**	**-0,339**
HOC RSI(vtarget=0.35;m=1;min_exp=0.3)	0,038	0,080	0,473	0,604	**-0,008**	-0,012	-0,340
HOC RSI(vtarget=0.45;m=1;min_exp=0.3)	0,038	0,083	0,462	0,581	**-0,008**	-0,013	-0,343
HOC RSI(vtarget=0.25;m=1;min_exp=0.5)	0,032	0,076	0,418	0,554	**-0,008**	-0,012	-0,365
HOC RSI(vtarget=0.35;m=1;min_exp=0.5)	0,039	0,084	0,461	0,593	-0,009	-0,013	-0,367
HOC RSI(vtarget=0.45;m=1;min_exp=0.5)	0,040	0,087	0,453	0,574	-0,009	-0,014	-0,367
HOC RSI(vtarget=0.25;m=10;min_exp=0.1)	0,006	0,124	0,052	0,063	-0,013	-0,020	-0,862
HOC RSI(vtarget=0.35;m=10;min_exp=0.1)	0,015	0,138	0,110	0,132	-0,014	-0,023	-0,859
HOC RSI(vtarget=0.45;m=10;min_exp=0.1)	0,015	0,144	0,108	0,127	-0,015	-0,024	-0,847
HOC RSI(vtarget=0.25;m=10;min_exp=0.3)	0,018	0,132	0,140	0,178	-0,014	-0,021	-0,865
HOC RSI(vtarget=0.35;m=10;min_exp=0.3)	0,028	0,146	0,195	0,241	-0,015	-0,023	-0,859
HOC RSI(vtarget=0.45;m=10;min_exp=0.3)	0,029	0,151	0,189	0,231	-0,016	-0,024	-0,863
HOC RSI(vtarget=0.25;m=10;min_exp=0.5)	0,022	0,136	0,159	0,206	-0,015	-0,021	-0,870
HOC RSI(vtarget=0.35;m=10;min_exp=0.5)	0,033	0,151	0,216	0,272	-0,016	-0,024	-0,859
HOC RSI(vtarget=0.45;m=10;min_exp=0.5)	0,033	0,157	0,212	0,263	-0,016	-0,025	-0,864

Table 6.12 - HOC strategy on EMV trading portfolio: results

A.4 SVR Portfolio protection results:

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	**0,114**	**0,527**	**0,656**	**-0,011**	**-0,017**	**-0,352**
OBPI SVR(vtarget=0.2)	0,010	0,127	0,076	0,102	-0,014	-0,020	-0,771
OBPI SVR(vtarget=0.25)	0,035	0,141	0,249	0,323	-0,015	-0,022	-0,676
OBPI SVR(vtarget=0.3)	0,040	0,149	0,270	0,347	-0,016	-0,024	-0,635
OBPI SVR(vtarget=0.35)	0,041	0,154	0,265	0,337	-0,016	-0,024	-0,621
OBPI SVR(vtarget=0.4)	0,038	0,158	0,242	0,304	-0,016	-0,025	-0,682
OBPI SVR(vtarget=0.45)	0,035	0,161	0,219	0,271	-0,017	-0,026	-0,742

Table 6.13 - OBPI strategy on SVR trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	**-0,352**
CPPI SVR(m=1;reb=1)	-0,042	0,087	-0,490	-0,596	-0,009	-0,014	-1,163
CPPI SVR(m=1;reb=3)	-0,005	**0,086**	-0,063	-0,076	**-0,008**	-0,014	-0,659
CPPI SVR(m=1;reb=6)	0,004	**0,086**	0,042	0,049	**-0,008**	-0,014	-0,598
CPPI SVR(m=1;reb=12)	0,008	**0,086**	0,093	0,110	**-0,008**	**-0,013**	-0,561
CPPI SVR(m=3;reb=1)	-0,022	0,132	-0,170	-0,203	-0,013	-0,021	-1,075
CPPI SVR(m=3;reb=3)	0,003	0,134	0,021	0,025	-0,013	-0,022	-0,814
CPPI SVR(m=3;reb=6)	0,005	0,138	0,033	0,038	-0,013	-0,023	-0,872
CPPI SVR(m=3;reb=12)	0,005	0,141	0,033	0,037	-0,013	-0,023	-0,933
CPPI SVR(m=10;reb=1)	-0,047	0,130	-0,362	-0,419	-0,013	-0,021	-1,052
CPPI SVR(m=10;reb=3)	-0,021	0,136	-0,155	-0,176	-0,013	-0,022	-0,774
CPPI SVR(m=10;reb=6)	-0,024	0,144	-0,169	-0,189	-0,014	-0,024	-0,950
CPPI SVR(m=10;reb=12)	-0,011	0,143	-0,076	-0,084	-0,013	-0,024	-0,944

Table 6.14 - CPPI strategy on SVR trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	**-0,352**
TIPP SVR(m=1;reb=1)	-0,060	**0,056**	-1,075	-1,336	**-0,006**	**-0,009**	-1,323
TIPP SVR(m=1;reb=3)	-0,024	**0,056**	-0,434	-0,531	**-0,006**	**-0,009**	-0,554
TIPP SVR(m=1;reb=6)	-0,015	**0,056**	-0,271	-0,328	**-0,006**	**-0,009**	-0,494
TIPP SVR(m=1;reb=12)	-0,010	0,057	-0,183	-0,221	**-0,006**	**-0,009**	-0,463
TIPP SVR(m=3;reb=1)	-0,052	0,107	-0,481	-0,589	-0,011	-0,017	-1,130
TIPP SVR(m=3;reb=3)	-0,021	0,111	-0,194	-0,231	-0,011	-0,018	-0,755
TIPP SVR(m=3;reb=6)	-0,017	0,115	-0,150	-0,174	-0,011	-0,019	-0,835
TIPP SVR(m=3;reb=12)	-0,016	0,120	-0,131	-0,147	-0,011	-0,020	-0,920
TIPP SVR(m=10;reb=1)	-0,015	0,139	-0,111	-0,135	-0,014	-0,022	-0,735
TIPP SVR(m=10;reb=3)	-0,008	0,144	-0,052	-0,062	-0,015	-0,023	-0,843
TIPP SVR(m=10;reb=6)	-0,005	0,147	-0,032	-0,038	-0,015	-0,024	-0,965
TIPP SVR(m=10;reb=12)	0,001	0,149	0,007	0,008	-0,015	-0,024	-0,913

Table 6.15 - TIPP strategy on SVR trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	**-0,352**
HOC SVR(vtarget=0.25;m=1;min_exp=0.1)	0,018	**0,072**	0,247	0,324	**-0,008**	**-0,011**	-0,456
HOC SVR(vtarget=0.35;m=1;min_exp=0.1)	0,020	0,078	0,258	0,329	**-0,008**	-0,012	-0,429
HOC SVR(vtarget=0.45;m=1;min_exp=0.1)	0,017	0,082	0,208	0,255	**-0,008**	-0,013	-0,465
HOC SVR(vtarget=0.25;m=1;min_exp=0.3)	0,018	**0,072**	0,247	0,324	**-0,008**	**-0,011**	-0,456
HOC SVR(vtarget=0.35;m=1;min_exp=0.3)	0,020	0,078	0,258	0,329	**-0,008**	-0,012	-0,429
HOC SVR(vtarget=0.45;m=1;min_exp=0.3)	0,017	0,082	0,208	0,255	**-0,008**	-0,013	-0,465
HOC SVR(vtarget=0.25;m=1;min_exp=0.5)	0,019	0,075	0,252	0,333	**-0,008**	-0,012	-0,469
HOC SVR(vtarget=0.35;m=1;min_exp=0.5)	0,022	0,082	0,267	0,342	-0,009	-0,013	-0,441
HOC SVR(vtarget=0.45;m=1;min_exp=0.5)	0,019	0,086	0,220	0,272	-0,009	-0,013	-0,457
HOC SVR(vtarget=0.25;m=10;min_exp=0.1)	0,011	0,118	0,095	0,116	-0,012	-0,019	-0,663
HOC SVR(vtarget=0.35;m=10;min_exp=0.1)	0,013	0,129	0,103	0,124	-0,013	-0,021	-0,662
HOC SVR(vtarget=0.45;m=10;min_exp=0.1)	0,007	0,136	0,049	0,057	-0,014	-0,023	-0,798
HOC SVR(vtarget=0.25;m=10;min_exp=0.3)	0,025	0,125	0,198	0,250	-0,014	-0,020	-0,665
HOC SVR(vtarget=0.35;m=10;min_exp=0.3)	0,028	0,136	0,203	0,251	-0,014	-0,022	-0,649
HOC SVR(vtarget=0.45;m=10;min_exp=0.3)	0,021	0,143	0,147	0,176	-0,015	-0,023	-0,785
HOC SVR(vtarget=0.25;m=10;min_exp=0.5)	0,028	0,130	0,218	0,280	-0,014	-0,021	-0,668
HOC SVR(vtarget=0.35;m=10;min_exp=0.5)	0,032	0,141	0,227	0,285	-0,015	-0,023	-0,638
HOC SVR(vtarget=0.45;m=10;min_exp=0.5)	0,026	0,148	0,174	0,211	-0,015	-0,024	-0,774

Table 6.16 - HOC strategy on SVR trading portfolio: results

A.5 ANN Portfolio protection results:

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	0,060	**0,114**	**0,527**	**0,656**	**-0,011**	**-0,017**	**-0,352**
OBPI ANN(vtarget=0.2)	0,030	0,131	0,230	0,313	-0,014	-0,020	-0,595
OBPI ANN(vtarget=0.25)	0,063	0,147	0,432	0,573	-0,016	-0,022	-0,431
OBPI ANN(vtarget=0.3)	0,074	0,157	0,474	0,622	-0,017	-0,024	-0,366
OBPI ANN(vtarget=0.35)	**0,079**	0,163	0,483	0,631	-0,017	-0,025	-0,420
OBPI ANN(vtarget=0.4)	0,078	0,168	0,467	0,603	-0,017	-0,026	-0,495
OBPI ANN(vtarget=0.45)	0,077	0,171	0,451	0,574	-0,018	-0,027	-0,552

Table 6.17 - OBPI strategy on ANN trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	0,527	0,656	-0,011	-0,017	-0,352
CPPI ANN(m=1;reb=1)	-0,001	**0,088**	-0,010	-0,013	-0,009	**-0,013**	-0,603
CPPI ANN(m=1;reb=3)	0,038	**0,088**	0,427	0,546	**-0,008**	**-0,013**	-0,361
CPPI ANN(m=1;reb=6)	0,047	0,089	0,530	0,670	**-0,008**	-0,014	-0,347
CPPI ANN(m=1;reb=12)	0,051	0,089	**0,575**	**0,721**	**-0,008**	-0,014	**-0,345**
CPPI ANN(m=3;reb=1)	0,025	0,134	0,187	0,241	-0,013	-0,021	-0,646
CPPI ANN(m=3;reb=3)	0,052	0,138	0,378	0,476	-0,013	-0,022	-0,466
CPPI ANN(m=3;reb=6)	0,054	0,143	0,378	0,465	-0,014	-0,023	-0,529
CPPI ANN(m=3;reb=12)	0,053	0,150	0,349	0,417	-0,014	-0,024	-0,626
CPPI ANN(m=10;reb=1)	0,002	0,130	0,015	0,019	-0,013	-0,021	-0,486
CPPI ANN(m=10;reb=3)	0,031	0,139	0,219	0,267	-0,014	-0,022	-0,467
CPPI ANN(m=10;reb=6)	0,029	0,151	0,193	0,227	-0,015	-0,025	-0,576
CPPI ANN(m=10;reb=12)	0,039	0,153	0,255	0,299	-0,014	-0,025	-0,645

Table 6.18 - CPPI strategy on ANN trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	-0,352
TIPP ANN(m=1;reb=1)	-0,027	**0,057**	-0,466	-0,592	**-0,006**	**-0,009**	-0,598
TIPP ANN(m=1;reb=3)	0,011	**0,057**	0,196	0,245	**-0,006**	**-0,009**	-0,245
TIPP ANN(m=1;reb=6)	0,021	0,058	0,357	0,442	**-0,006**	**-0,009**	**-0,237**
TIPP ANN(m=1;reb=12)	0,025	0,059	0,423	0,521	**-0,006**	**-0,009**	-0,243
TIPP ANN(m=3;reb=1)	-0,017	0,111	-0,154	-0,191	-0,011	-0,018	-0,560
TIPP ANN(m=3;reb=3)	0,015	0,116	0,131	0,158	-0,011	-0,019	-0,448
TIPP ANN(m=3;reb=6)	0,020	0,122	0,165	0,194	-0,012	-0,020	-0,518
TIPP ANN(m=3;reb=12)	0,020	0,131	0,151	0,172	-0,012	-0,021	-0,629
TIPP ANN(m=10;reb=1)	0,020	0,145	0,140	0,172	-0,015	-0,023	-0,529
TIPP ANN(m=10;reb=3)	0,035	0,152	0,233	0,282	-0,015	-0,024	-0,561
TIPP ANN(m=10;reb=6)	0,038	0,157	0,240	0,288	-0,016	-0,025	-0,665
TIPP ANN(m=10;reb=12)	0,043	0,161	0,268	0,323	-0,016	-0,025	-0,666

Table 6.19 - TIPP strategy on ANN trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	0,060	0,114	0,527	0,656	-0,011	-0,017	-0,352
HOC ANN(vtarget=0.25;m=1;min_exp=0.1)	0,050	**0,073**	0,681	0,913	**-0,008**	**-0,011**	**-0,151**
HOC ANN(vtarget=0.35;m=1;min_exp=0.1)	0,057	0,080	**0,713**	**0,938**	**-0,008**	-0,012	-0,218
HOC ANN(vtarget=0.45;m=1;min_exp=0.1)	0,057	0,085	0,669	0,857	**-0,008**	-0,013	-0,289
HOC ANN(vtarget=0.25;m=1;min_exp=0.3)	0,050	**0,073**	0,681	0,913	**-0,008**	**-0,011**	**-0,151**
HOC ANN(vtarget=0.35;m=1;min_exp=0.3)	0,057	0,080	**0,713**	**0,938**	**-0,008**	-0,012	-0,218
HOC ANN(vtarget=0.45;m=1;min_exp=0.3)	0,057	0,085	0,669	0,857	**-0,008**	-0,013	-0,289
HOC ANN(vtarget=0.25;m=1;min_exp=0.5)	0,050	0,076	0,662	0,892	**-0,008**	**-0,011**	-0,167
HOC ANN(vtarget=0.35;m=1;min_exp=0.5)	0,059	0,085	0,697	0,921	-0,009	-0,013	-0,224
HOC ANN(vtarget=0.45;m=1;min_exp=0.5)	0,058	0,089	0,657	0,847	-0,009	-0,014	-0,296
HOC ANN(vtarget=0.25;m=10;min_exp=0.1)	0,042	0,124	0,336	0,419	-0,013	-0,020	-0,427
HOC ANN(vtarget=0.35;m=10;min_exp=0.1)	0,053	0,138	0,384	0,475	-0,014	-0,022	-0,410
HOC ANN(vtarget=0.45;m=10;min_exp=0.1)	0,050	0,146	0,344	0,415	-0,014	-0,024	-0,543
HOC ANN(vtarget=0.25;m=10;min_exp=0.3)	0,053	0,131	0,406	0,522	-0,014	-0,021	-0,428
HOC ANN(vtarget=0.35;m=10;min_exp=0.3)	0,065	0,145	0,449	0,571	-0,015	-0,023	-0,425
HOC ANN(vtarget=0.45;m=10;min_exp=0.3)	0,062	0,153	0,407	0,504	-0,016	-0,024	-0,561
HOC ANN(vtarget=0.25;m=10;min_exp=0.5)	0,057	0,136	0,416	0,546	-0,014	-0,021	-0,429
HOC ANN(vtarget=0.35;m=10;min_exp=0.5)	**0,069**	0,150	0,461	0,597	-0,016	-0,023	-0,430
HOC ANN(vtarget=0.45;m=10;min_exp=0.5)	0,067	0,158	0,422	0,531	-0,016	-0,025	-0,566

Table 6.20 - HOC strategy on ANN trading portfolio: results

A.6 RNN Portfolio protection results:

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	0,060	**0,114**	**0,527**	**0,656**	**-0,011**	**-0,017**	-0,352
OBPI RNN(vtarget=0.2)	0,041	0,121	0,337	0,444	-0,013	-0,019	-0,397
OBPI RNN(vtarget=0.25)	0,062	0,133	0,468	0,600	-0,014	-0,021	-0,288
OBPI RNN(vtarget=0.3)	0,067	0,141	0,476	0,604	-0,015	-0,022	-0,253
OBPI RNN(vtarget=0.35)	**0,069**	0,145	0,472	0,595	-0,015	-0,023	**-0,249**
OBPI RNN(vtarget=0.4)	0,067	0,149	0,451	0,561	-0,015	-0,023	-0,249
OBPI RNN(vtarget=0.45)	0,065	0,152	0,426	0,522	-0,015	-0,024	-0,249

Table 6.21 - OBPI strategy on RNN trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	-0,352
CPPI RNN(m=1;reb=1)	-0,010	0,085	-0,117	-0,144	**-0,008**	**-0,013**	-0,548
CPPI RNN(m=1;reb=3)	0,028	0,085	0,329	0,398	**-0,008**	**-0,013**	-0,174
CPPI RNN(m=1;reb=6)	0,037	0,084	0,439	0,527	**-0,008**	**-0,013**	-0,167
CPPI RNN(m=1;reb=12)	0,042	**0,083**	0,498	0,596	**-0,008**	**-0,013**	**-0,159**
CPPI RNN(m=3;reb=1)	0,013	0,130	0,097	0,116	-0,012	-0,020	-0,448
CPPI RNN(m=3;reb=3)	0,041	0,132	0,308	0,362	-0,012	-0,021	-0,251
CPPI RNN(m=3;reb=6)	0,043	0,135	0,321	0,370	-0,012	-0,021	-0,256
CPPI RNN(m=3;reb=12)	0,046	0,136	0,337	0,383	-0,012	-0,021	-0,275
CPPI RNN(m=10;reb=1)	-0,008	0,128	-0,061	-0,070	-0,012	-0,020	-0,419
CPPI RNN(m=10;reb=3)	0,023	0,132	0,178	0,205	-0,012	-0,021	-0,251
CPPI RNN(m=10;reb=6)	0,020	0,140	0,145	0,163	-0,012	-0,023	-0,276
CPPI RNN(m=10;reb=12)	0,034	0,138	0,243	0,273	-0,012	-0,022	-0,291

Table 6.22 - CPPI strategy on RNN trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	**0,060**	0,114	**0,527**	**0,656**	-0,011	-0,017	-0,352
TIPP RNN(m=1;reb=1)	-0,027	**0,052**	-0,521	-0,691	-0,006	**-0,008**	-0,617
TIPP RNN(m=1;reb=3)	0,010	**0,052**	0,194	0,254	**-0,005**	**-0,008**	-0,131
TIPP RNN(m=1;reb=6)	0,019	**0,052**	0,370	0,480	**-0,005**	**-0,008**	-0,126
TIPP RNN(m=1;reb=12)	0,024	0,053	0,455	0,586	**-0,005**	**-0,008**	**-0,124**
TIPP RNN(m=3;reb=1)	-0,015	0,100	-0,147	-0,185	-0,010	-0,016	-0,392
TIPP RNN(m=3;reb=3)	0,018	0,103	0,179	0,220	-0,010	-0,016	-0,251
TIPP RNN(m=3;reb=6)	0,024	0,107	0,221	0,263	-0,010	-0,017	-0,249
TIPP RNN(m=3;reb=12)	0,028	0,110	0,254	0,297	-0,010	-0,017	-0,284
TIPP RNN(m=10;reb=1)	0,024	0,128	0,185	0,231	-0,013	-0,020	-0,249
TIPP RNN(m=10;reb=3)	0,037	0,135	0,273	0,331	-0,014	-0,021	-0,251
TIPP RNN(m=10;reb=6)	0,039	0,138	0,279	0,330	-0,014	-0,022	-0,289
TIPP RNN(m=10;reb=12)	0,041	0,141	0,292	0,346	-0,014	-0,023	-0,279

Table 6.23 - TIPP strategy on RNN trading portfolio: results

	Ann. Returns	Ann. Std. Dev.	Sharpe	Sortino	VaR	CVaR	MDD
60/40 Portfolio	0,060	0,114	0,527	0,656	-0,011	-0,017	-0,352
HOC RNN(vtarget=0.25;m=1;min_exp=0.1)	0,047	**0,068**	**0,683**	**0,912**	**-0,007**	**-0,010**	-0,109
HOC RNN(vtarget=0.35;m=1;min_exp=0.1)	0,049	0,074	0,666	0,867	**-0,007**	-0,011	**-0,100**
HOC RNN(vtarget=0.45;m=1;min_exp=0.1)	0,047	0,079	0,601	0,750	-0,008	-0,012	-0,131
HOC RNN(vtarget=0.25;m=1;min_exp=0.3)	0,047	**0,068**	**0,683**	**0,912**	**-0,007**	**-0,010**	-0,109
HOC RNN(vtarget=0.35;m=1;min_exp=0.3)	0,049	0,074	0,666	0,867	**-0,007**	-0,011	**-0,100**
HOC RNN(vtarget=0.45;m=1;min_exp=0.3)	0,047	0,079	0,601	0,750	-0,008	-0,012	-0,131
HOC RNN(vtarget=0.25;m=1;min_exp=0.5)	0,047	0,071	0,664	0,887	**-0,007**	-0,011	-0,113
HOC RNN(vtarget=0.35;m=1;min_exp=0.5)	0,051	0,078	0,651	0,847	-0,008	-0,012	-0,103
HOC RNN(vtarget=0.45;m=1;min_exp=0.5)	0,048	0,082	0,591	0,740	-0,008	-0,013	-0,131
HOC RNN(vtarget=0.25;m=10;min_exp=0.1)	0,044	0,112	0,391	0,483	-0,011	-0,018	-0,288
HOC RNN(vtarget=0.35;m=10;min_exp=0.1)	0,048	0,122	0,391	0,477	-0,012	-0,020	-0,249
HOC RNN(vtarget=0.45;m=10;min_exp=0.1)	0,043	0,130	0,333	0,390	-0,012	-0,021	-0,249
HOC RNN(vtarget=0.25;m=10;min_exp=0.3)	0,056	0,118	0,470	0,593	-0,012	-0,019	-0,288
HOC RNN(vtarget=0.35;m=10;min_exp=0.3)	0,060	0,129	0,468	0,583	-0,013	-0,020	-0,249
HOC RNN(vtarget=0.45;m=10;min_exp=0.3)	0,055	0,136	0,408	0,489	-0,013	-0,022	-0,249
HOC RNN(vtarget=0.25;m=10;min_exp=0.5)	0,058	0,122	0,474	0,606	-0,013	-0,019	-0,288
HOC RNN(vtarget=0.35;m=10;min_exp=0.5)	**0,063**	0,133	0,474	0,597	-0,014	-0,021	-0,249
HOC RNN(vtarget=0.45;m=10;min_exp=0.5)	0,059	0,140	0,418	0,508	-0,014	-0,022	-0,249

Table 6.24 - HOC strategy on RNN trading portfolio: results

A.7 Trading portfolios strategies:

MACD trading strategy :

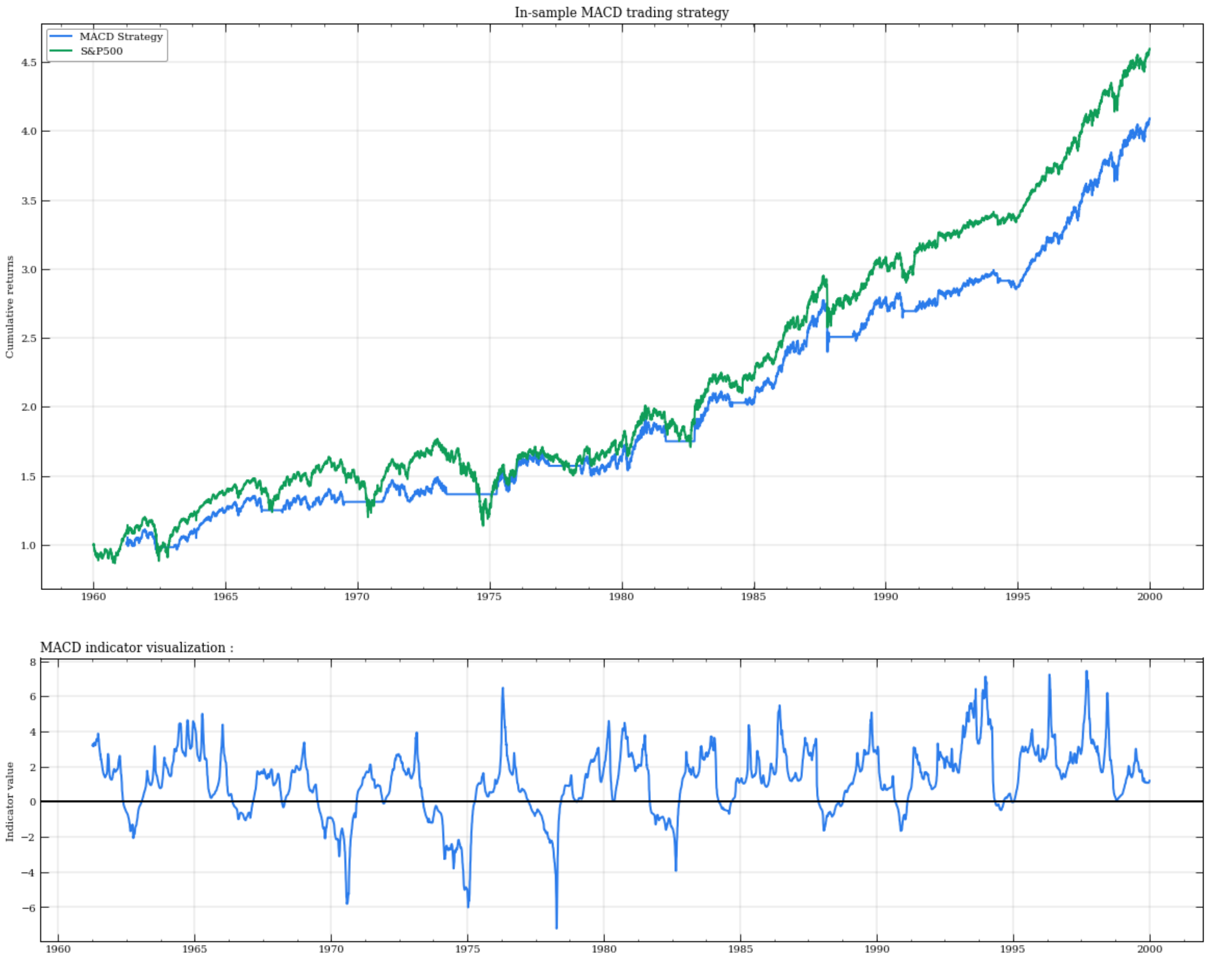

Figure 6.1 – MACD Trading Strategy: In-sample

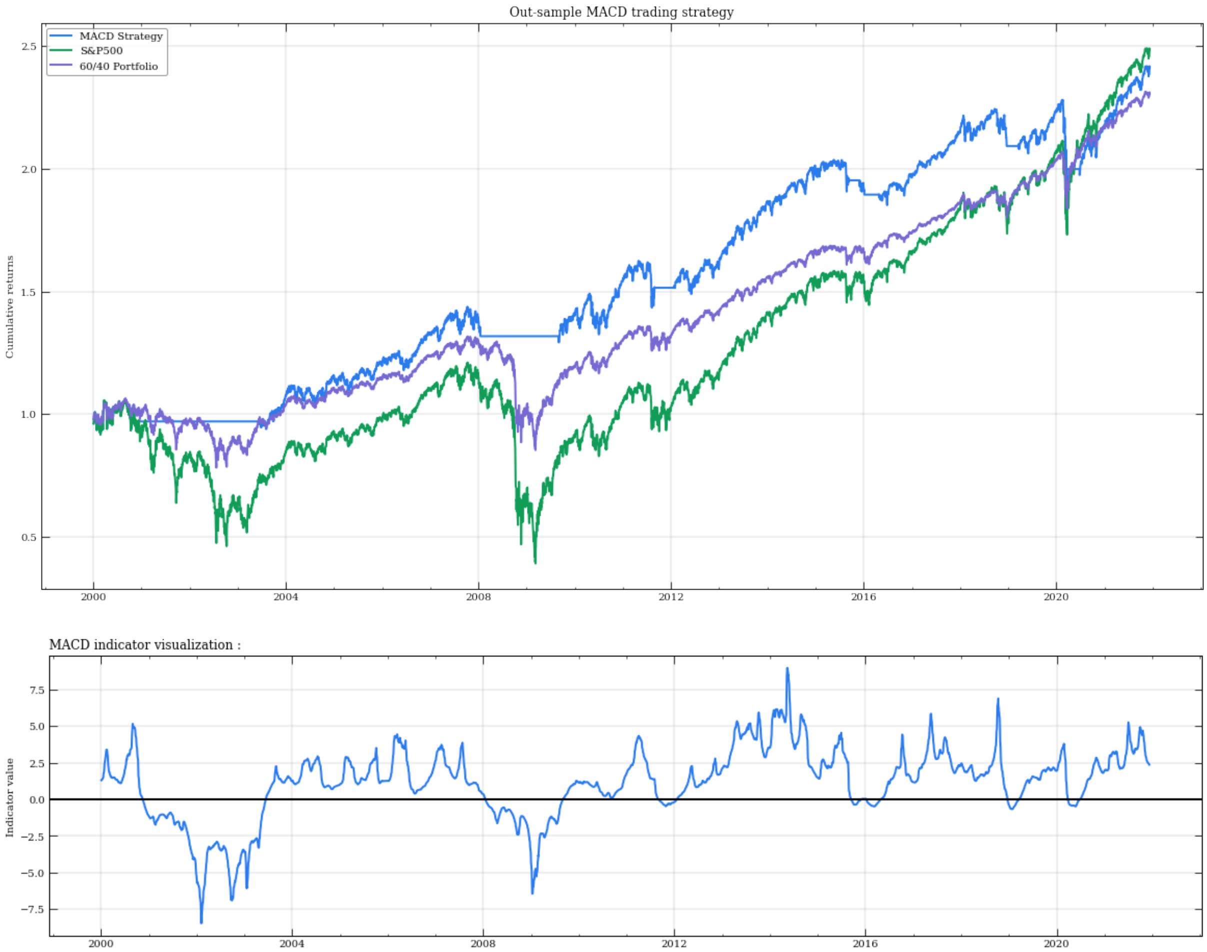

Figure 6.2 – MACD Trading Strategy: Out-sample

RSI trading strategy :

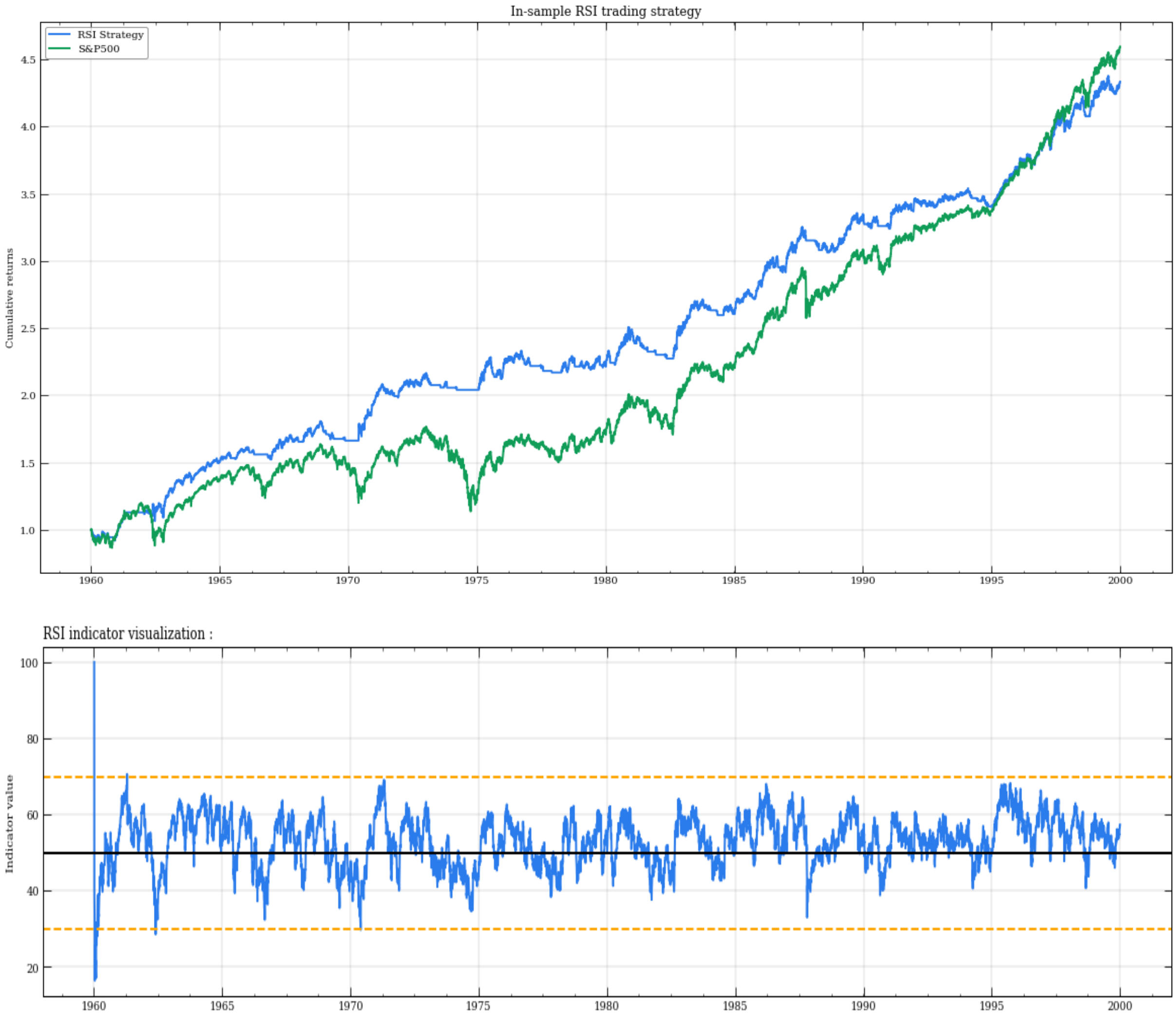

Figure 6.3 – RSI Trading Strategy: In-sample

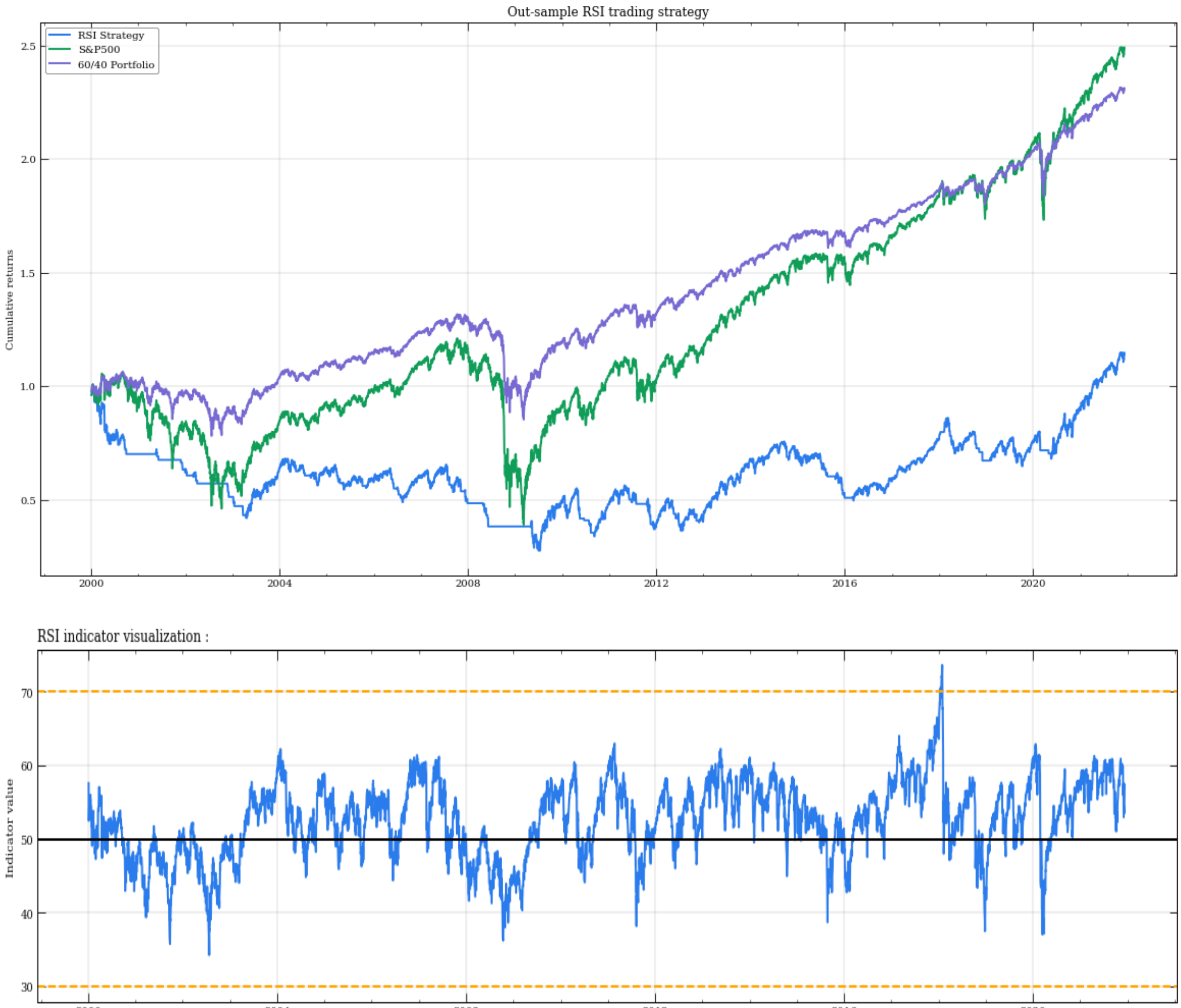

Figure 6.4 – RSI Trading Strategy: Out-sample

EMV trading strategy :

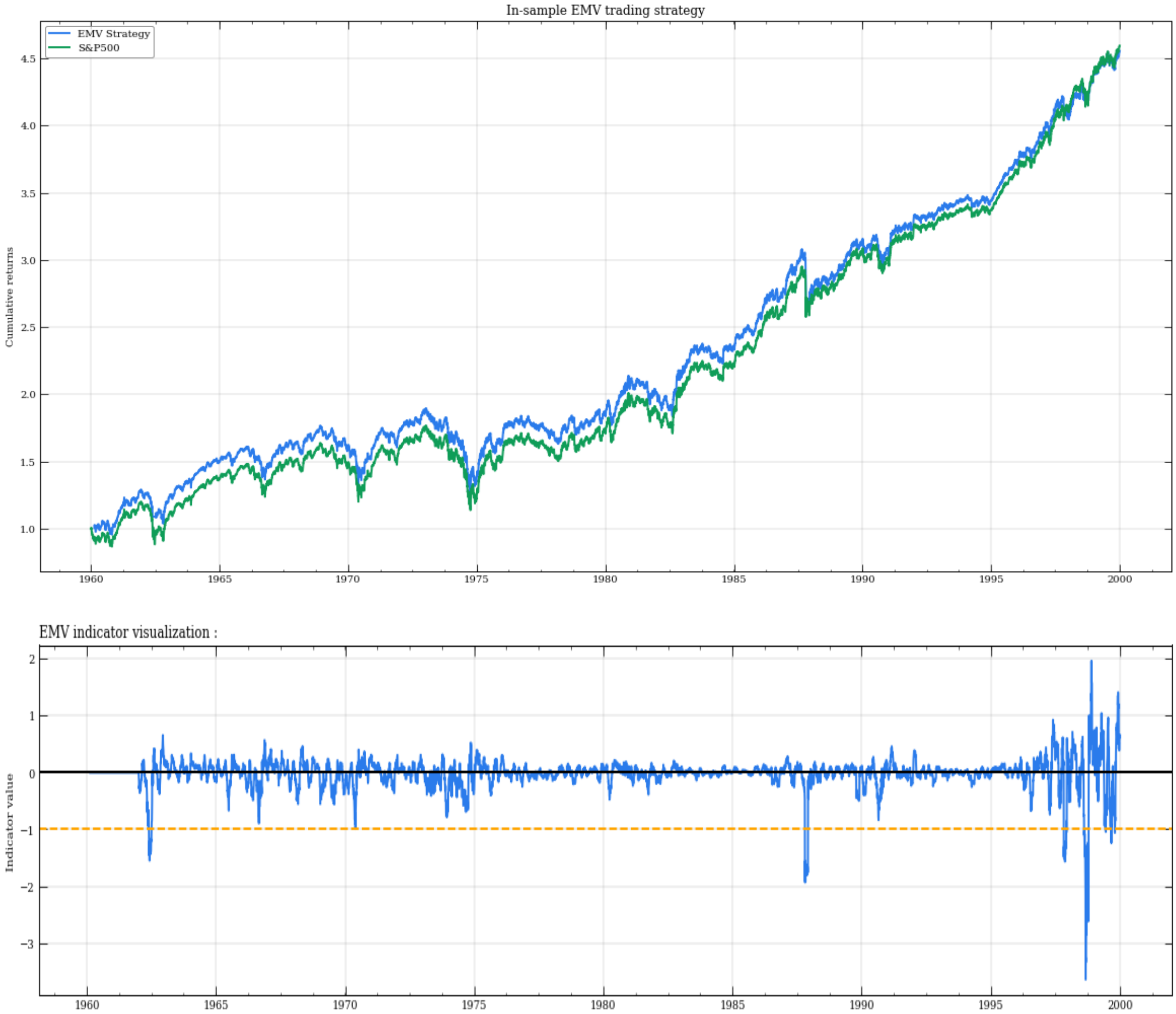

Figure 6.5 – EMV Trading Strategy: In-sample

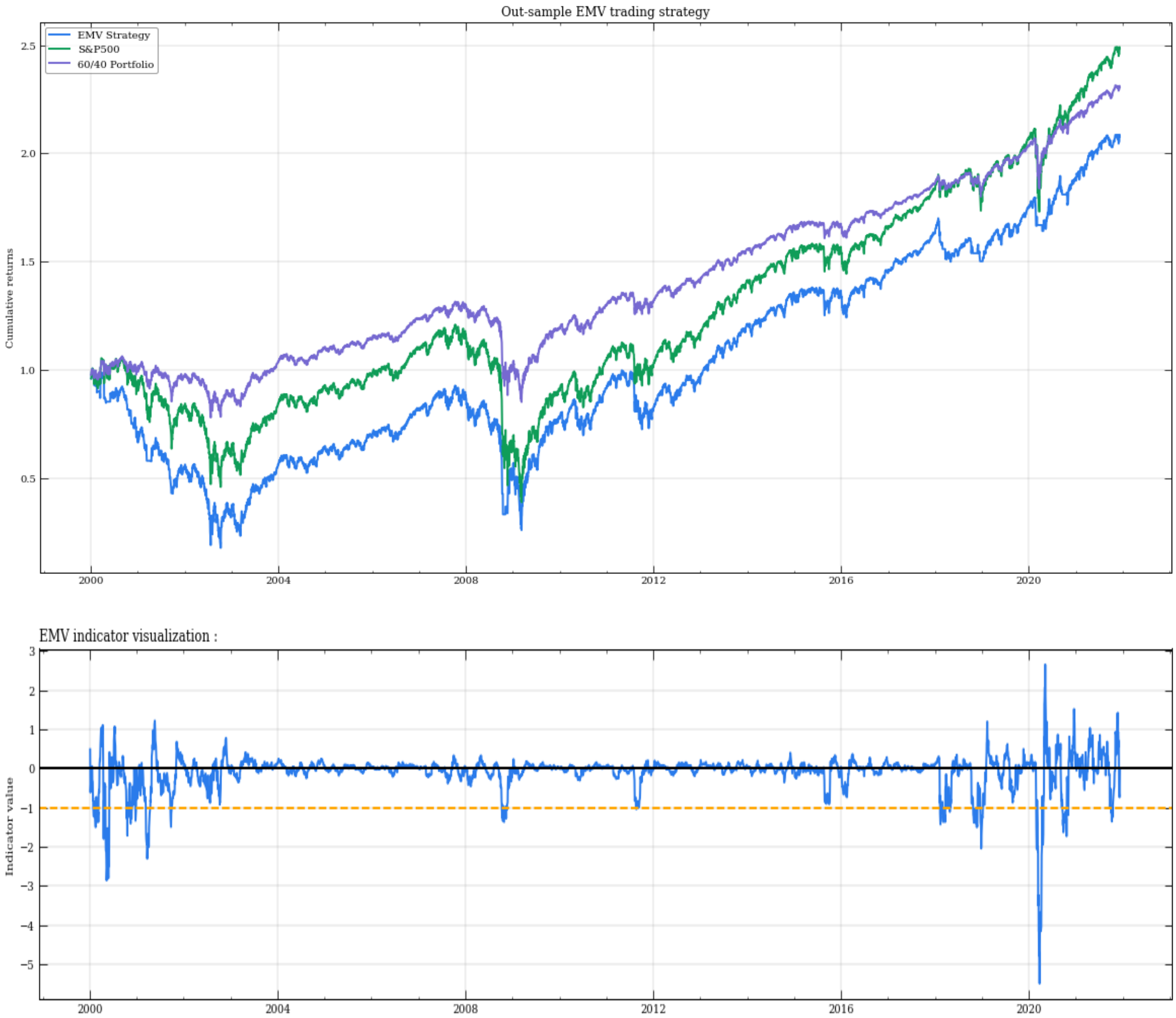

Figure 6.6 – EMV Trading Strategy: Out-sample

SVR trading strategy :

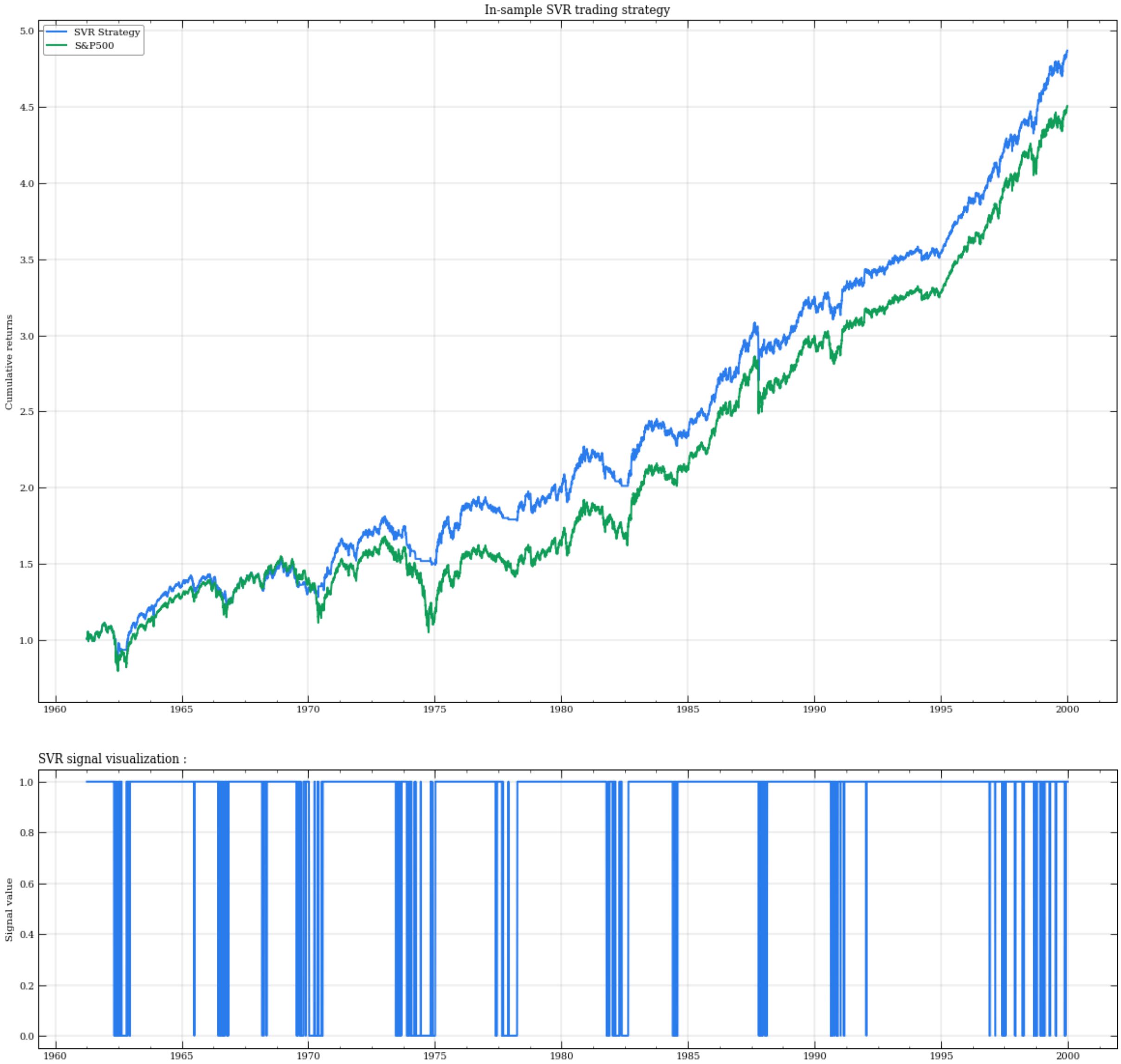

Figure 6.7 – SVR Trading Strategy: In-sample

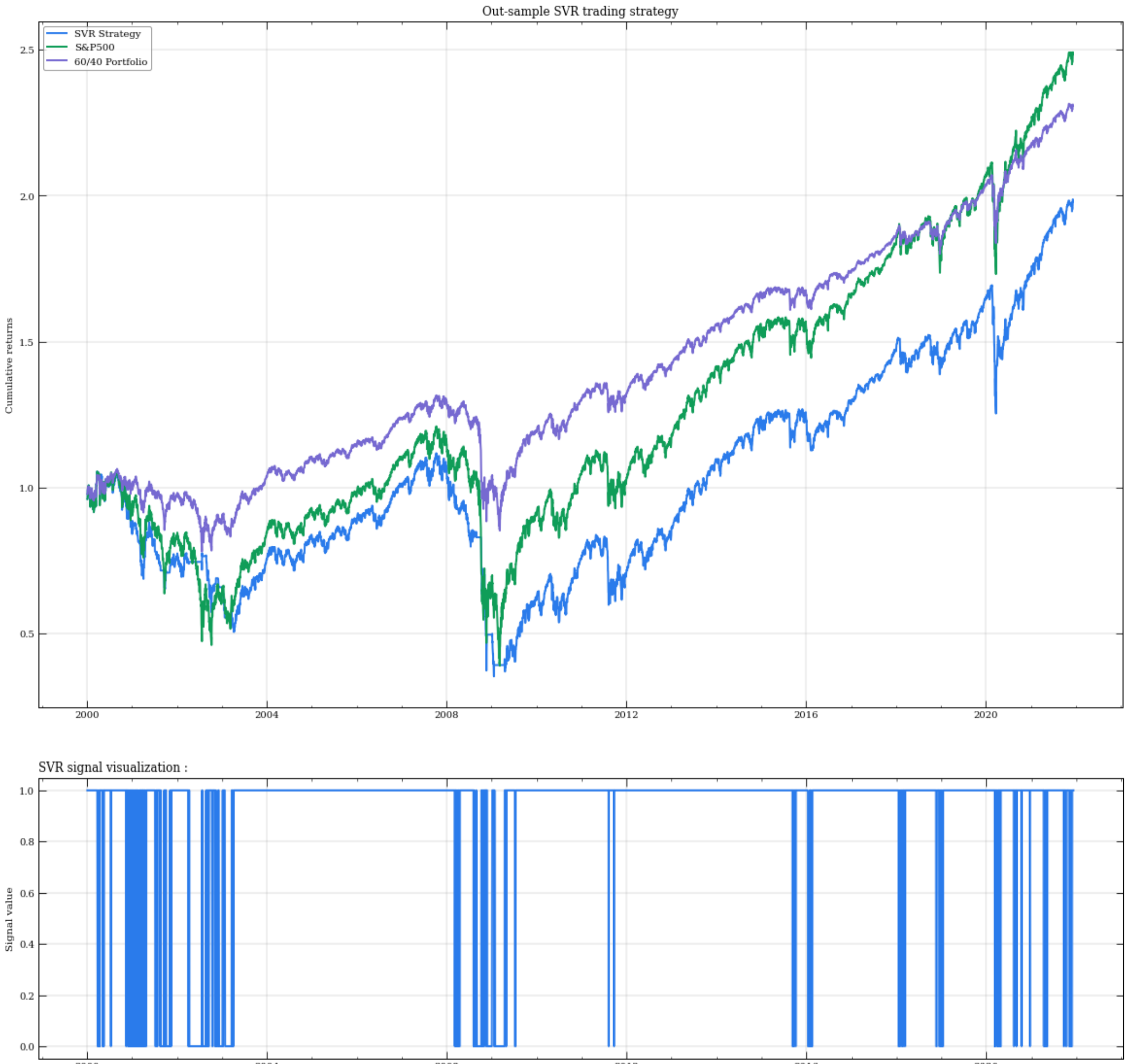

Figure 6.8 – SVR Trading Strategy: Out-sample

ANN trading strategy :

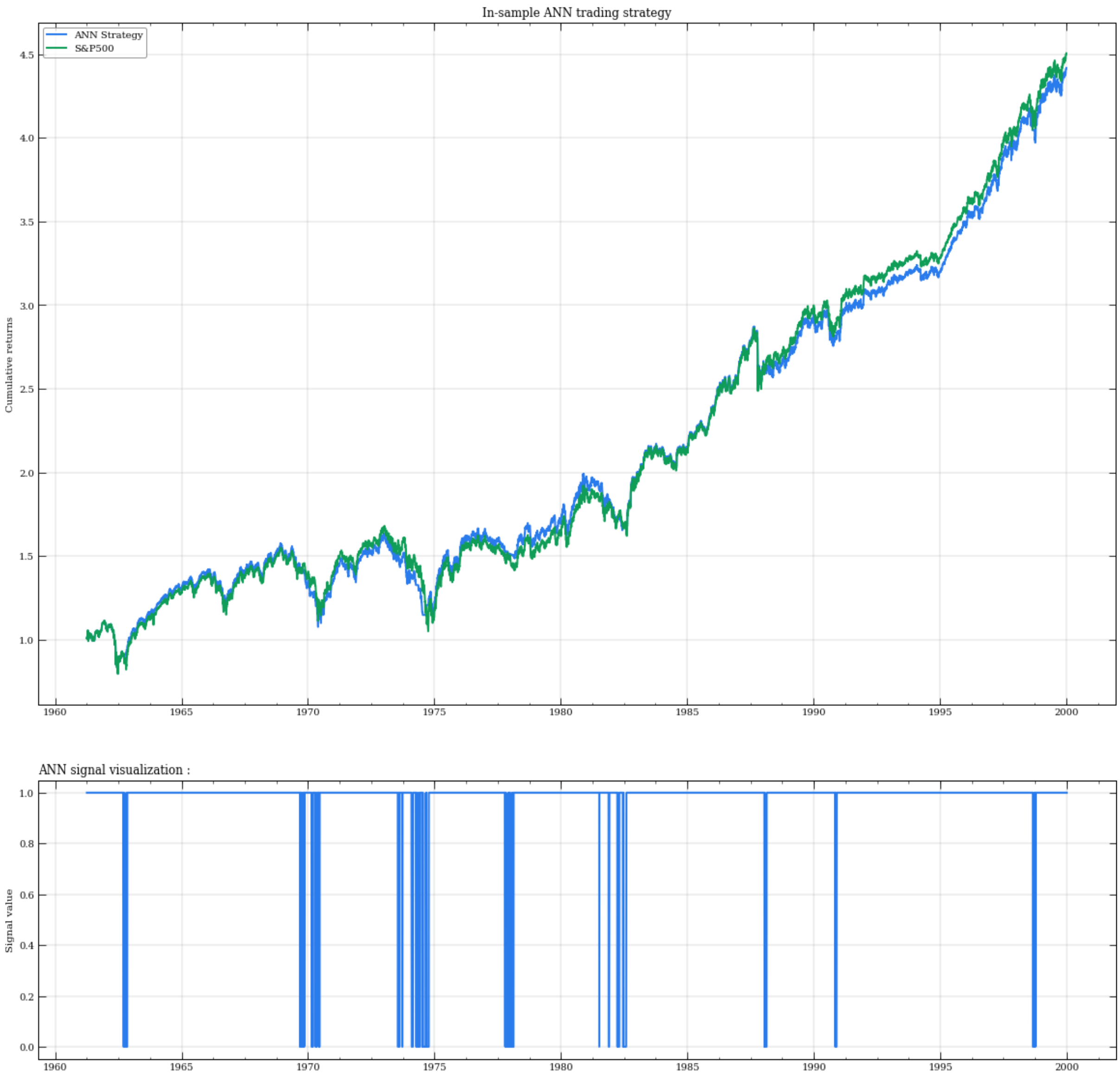

Figure 6.9 – ANN Trading Strategy: In-sample

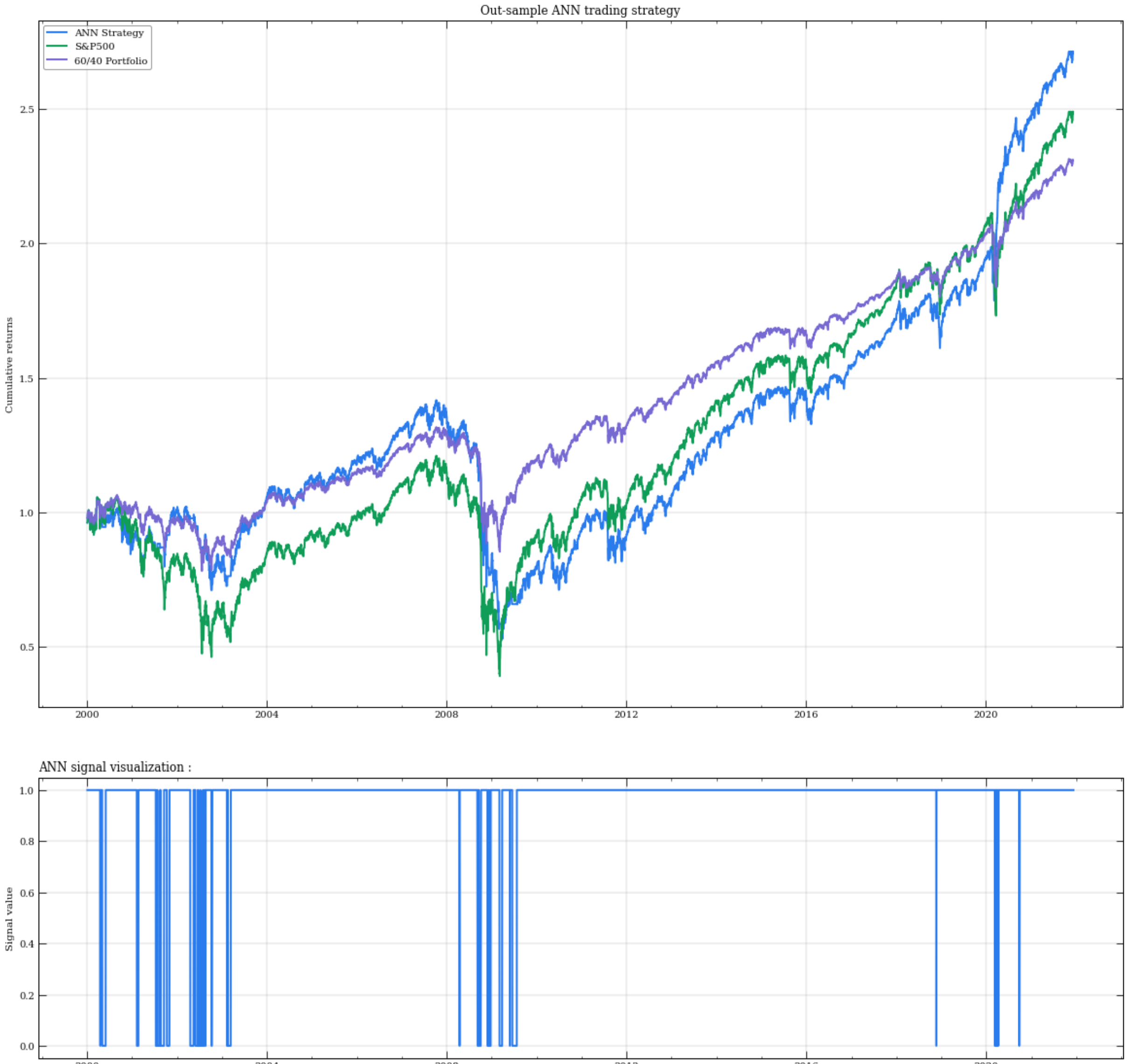

Figure 6.10 – ANN Trading Strategy: Out-sample

RNN trading strategy:

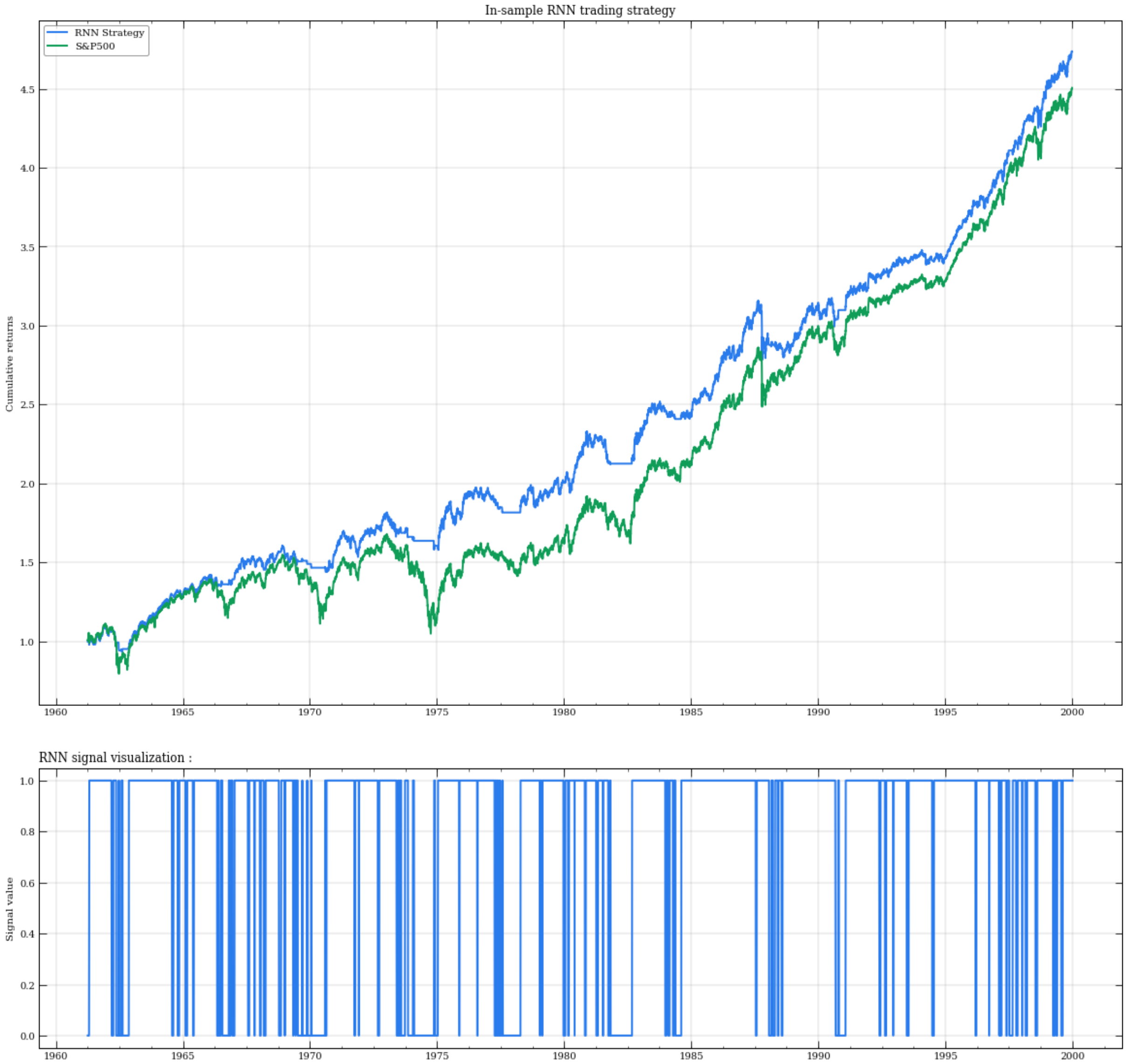

Figure 6.11 – RNN Trading Strategy: In-sample

Figure 6.12 – RNN Trading Strategy: Out-sample

A.8 Option pricing model

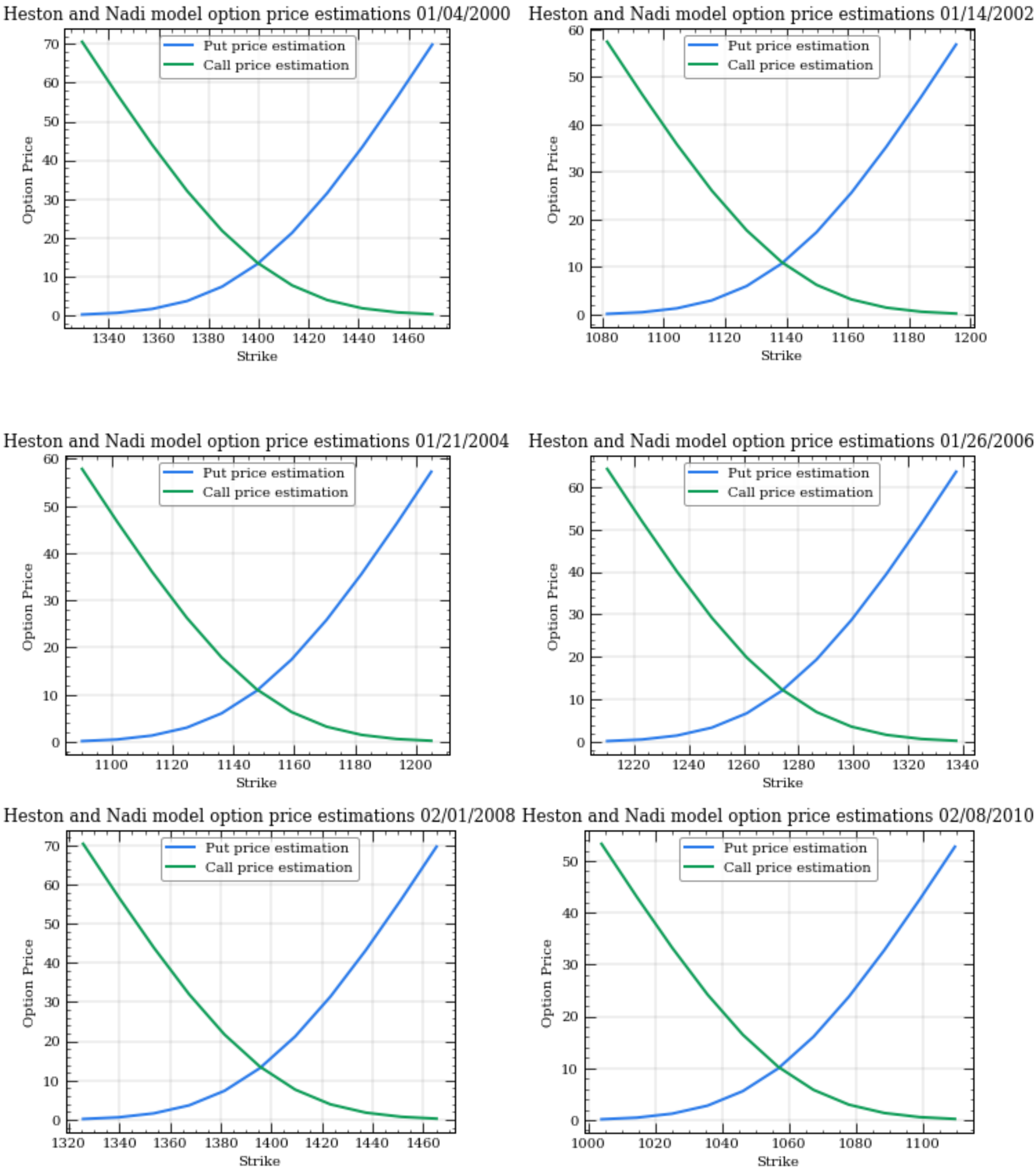

Figure 6.13 – Options price data estimations (1)

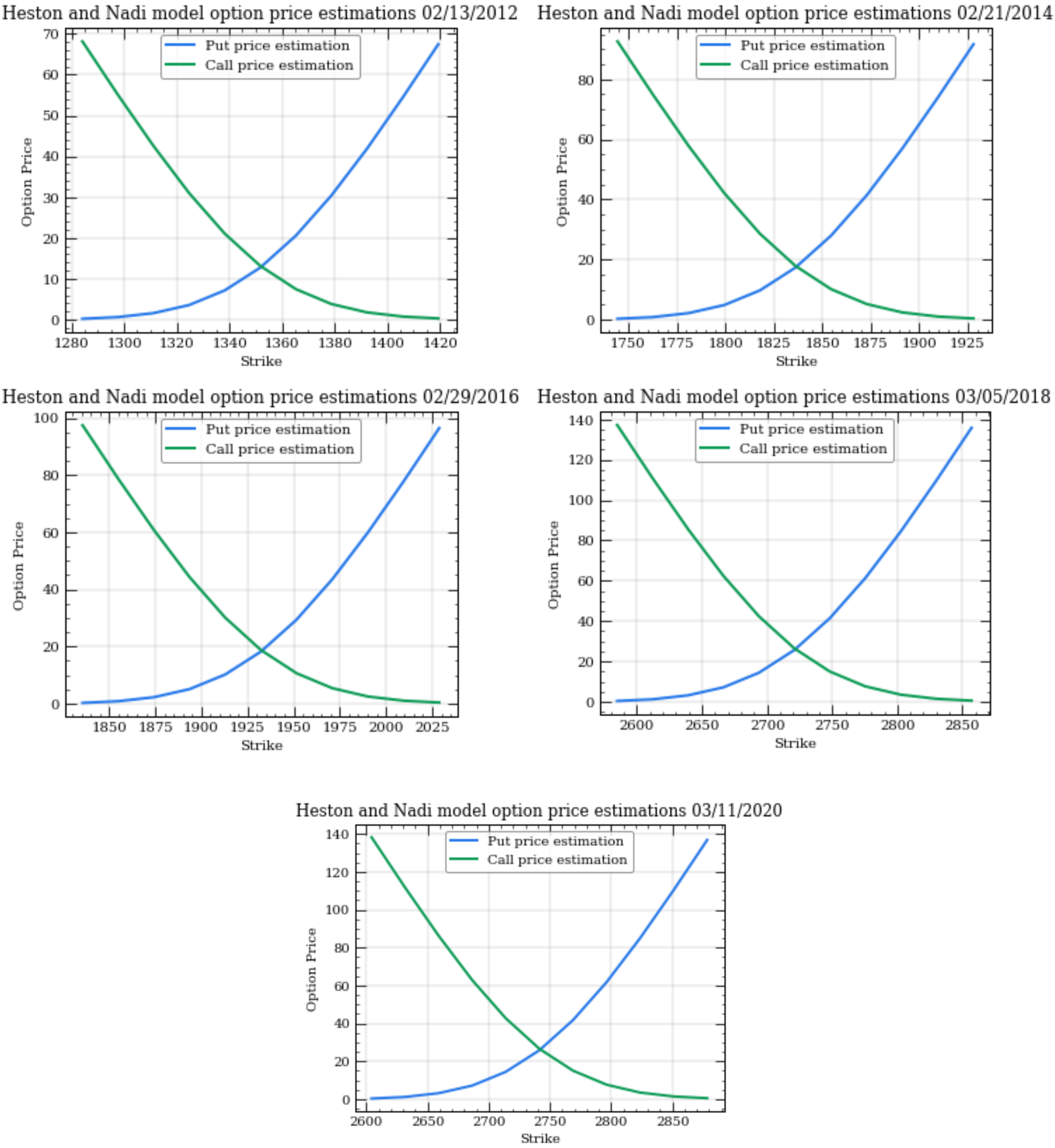

Figure 6.14 – Options price data estimations (2)

APPENDIX B

B.1 Assumptions of assets dynamic

In this work, we consider that the portfolios that we want to construct should normally be composed of two different kinds of assets, a ***risky asset***, and a ***riskless asset***. Indeed, as we have seen in this paper, trading portfolios are only composed of risky index (S&P500), and protected trading portfolios are composed of risky index (S&P500) and riskless index (AGG). The dynamic of the other assets involved are not relevant to mentioned because their dynamics are not important in strategies constructions and are slightly like the previously cited indices.

Riskless asset dynamic

As we have seen in several strategies we have implemented in this work, although the riskless index is not a deterministic interest rate, we consider it in the calculations to simplify its integration in the formulas. In this paper, we consider B to be the value of the riskless index, so that the dynamics followed by this index is defined by:

$$dB_t = B_t r dt \tag{62}$$

Where, r is the deterministic interest rate defined in the “Data” part of this work, and t belong to the investment interval [0, T].

Risky asset dynamic

In a similar way, we define the dynamics of the risky index, being more elaborate than for the riskless index. Indeed, the risky index follows a dynamic of Geometric Brownian motion or "diffusion process" such that, for a value of risky asset S, the dynamic followed is defined by:

$$dS_t = S_t[\mu_t dt + \sigma_t dW_t] \tag{63}$$

Where $(W_t)_t$ is a standard Wiener process at time t, μ_t is the drift at time t and σ_t is the volatility of the motion at time t.

However, the dynamic of the stochastic volatility of the model is determined by:

$$d\sigma_t = a(t, \sigma_t)dt + b(t, \sigma_t)d\widetilde{W}_t \tag{64}$$

Where $(\widetilde{W}_t)_t$ is another standard Wiener process, a and b are functions of σ_t.

B.2 Heston and Nandi model for option pricing

To construct the Heston and Nandi option pricing model, we mainly used as reference, the paper of Byun (2011). Indeed, this work paper present precisely how to construct an option pricing model under the Heston and Nandi model and with the help of a Monte-Carlo simulation.

Asset dynamic under Heston and Nandi model

With the assets dynamic defined before we can determine a new asset dynamic that fit better in the case of option pricing constraint. Indeed, it should focus more on the stochastic volatility of the indices to provide better results for the option pricing purpose. For Heston and Nandi model the asset dynamic is given by:

$$R_t = \ln\left(\frac{S_t}{S_{t-1}}\right) = r + \lambda h_t + \sqrt{h_t} z_t \tag{65}$$

Where R_t is the log return of the asset at time t, S_t is the asset value at time t, r is the deterministic interest rate. $h_t$is the conditional variance of the log-return. $\lambda h_t$is the equity risk premium. And finally, z_t is a standard normal random variable.

The dynamic of the stochastic volatility is the GARCH (1,1) and is given by:

$$h_t = w + bh_{t-1} + a\left(z_{t-1} - c\sqrt{h_{t-1}}\right)^2 \tag{66}$$

Where a, b, c, w, and λ are the GARCH (1,1) parameters that must be estimated.

The dynamic under the Heston and Nandi is completely defined and is the base for further computations.

Likelihood function of the model

As we have seen in the previous paragraph, the model is composed of five distinct parameters. These parameters are a, b, c, w, λ. The objective is to estimate them over the period of 2000-

01-01 to 2021-12-31 by solving the likelihood function associated to the model. The likelihood function is given by (67).

$$L = \prod_{t=1}^{T} \frac{1}{\sqrt{2\pi h_t}} exp\left\{-\frac{(R_t - r - \lambda h_t)^2}{2h_t}\right\} \tag{67}$$

And for ease of computation, we can consider the log-likelihood function by:

$$\log L = \sum_{t=1}^{T} -0.5\left\{\log(2\pi h_t) + \frac{(R_t - r - \lambda h_t)^2}{h_t}\right\} \tag{68}$$

The model parameters estimation is essential to determine the options price. Indeed, by solving the log likelihood function through an optimization problem, we are able to determine the optimal parameter for the model. These parameters solve the optimization problem by giving the right ponderation to every variable of the model and maximize the likelihood function.

Simulation of asset prices to estimate European options prices

After solving the optimization problem of the model and then finding every parameter we need, we are able to replace parameters in the model by the ones we found. Once done, potential future returns are determined with a coherent stochastic volatility according to the Heston and Nandi model. With a Monte-Carlo simulation, 10’000 returns are simulated with this model to evaluate a possible evolution of the market for all the period studied. Some adjustment on the returns is applicated to correct the results obtained with respect to the paper of Duan and Simonato (1995) on empirical martingale simulation for asset price.

The implementation of this model aims to determine “future” assets price with a huge importance on the stochastic volatility, and then determine option price with the variations found. In that way we can get coherent option prices with a volatility near to the real market implied volatility for the maturity chosen.

B.3 Statistical formulas:

Skewness

The skewness aims to measure the asymmetry of a random variable. A negative skewness value means that the mass of the distribution is centered to the left side of the distribution. A negative skewness is also called a left tailed distribution. A positive skewness value means that the mass of the distribution is centered to the right side of the distribution. A positive skewness is also called a right tailed distribution. In finance, a positive skewness is preferable even if it's common to find negative skewness for security returns distributions. Indeed, a negative skewness in security returns distribution means that negative returns are more often to occurs, and in contrary, a positive skewness in security returns distribution means that positive returns are more often to occurs.

Technically, the skewness of a random variable X corresponds to the third central moment and is given by:

$$S(X) = E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] \tag{69}$$

Where X is the asset daily total returns, μ is the asset returns mean, σ is the asset returns volatility.

Excess of Kurtosis

The excess of Kurtosis aims to measure the tail-fatness of a random variable. A negative excess of kurtosis for a distribution, also called platykurtic distribution, means that large extreme events can occurs less often. A positive excess of kurtosis for a distribution, also called leptokurtic distribution, means that large extreme events can occurs more often. In finance, we generally prefer platykurtic distribution than leptokurtic distribution, because leptokurtic distribution means that large negative event may appear more often than in a platykurtic distribution.

Technically, the excess of Kurtosis of a random variable X corresponds to the fourth central moment and is given by:

$$K(X) = E\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] - 3 \tag{70}$$

Where X is the asset returns sample, μ is the asset returns mean, σ is the asset returns volatility.

B.4 Kernel functions:

Let's first remind that an SVR model is given by:

$$minimize \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{l} (\xi_i + \xi_i^*) \tag{25}$$

$$\text{subject to} \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon + \xi_i \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \tag{26}$$

$$with, \qquad w = \sum_i \alpha_i \cdot y_i \cdot k(x_i, x) \tag{27}$$

However, the model above is a linear SVR because it is composed of a linear kernel. To adapt the model to a nonlinear kernel, one can use the kernel trick and then adapt the model to a nonlinear kernel Even if we demonstrate the kernel trick below, in application with python, the kernel trick to adapt the kernel of the SVR model is automatically put in place when it's necessary without manual operation. The kernel trick is given by:

$$k(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j) \tag{71}$$

Where $\varphi(\bullet)$ is a linear transform, and "·" is the dot product.

Gaussian (RBF) kernel

The Gaussian radial basis function kernel is given by:

$$k(x_i, x_j) = exp\left(-\gamma \|x_i - x_j\|^2\right) \tag{72}$$

With $\gamma > 0$.

Sigmoid kernel

The Sigmoid function kernel is given by:

$$k(x_i, x_j) = \tanh(\alpha {x_i}^T x_j + c) \tag{73}$$

With α > 0, and c < 0.