Louvain School of Management

Building and backtesting a systematic trading strategy based on technical analysis

Can a trading strategy based on technical analysis indicators outperform the cryptocurrency market in terms of risk-adjusted returns ?

Author(s): Nathan Quiévreux & Julien Adam
Supervisor(s): Leonardo Iania
Academic year 2023-2024
Dissertation for the master of 120 credits in business engineering
Finance (FIN): Financial markets
Daytime schedule

Abstract :

This thesis investigates the potential of a systematic trading strategy, based on technical analysis indicators, to outperform the cryptocurrency market in terms of risk-adjusted returns. The research focuses on the development and enhancement of a trading system applied to the cryptocurrency market. Employing a semi-systematic approach, the study begins with generating trading strategy ideas derived from personal market experience and knowledge. These ideas are then systematized into clear, objective entry and exit rules, followed by rigorous backtesting against historical data.

The methodology uses the Relative Strength Index (RSI) for entry signals and the Average True Range (ATR) to set flexible stop-losses based on market conditions. The strategy is tested on ten major cryptocurrencies, with results analyzed based on various system parameters and market conditions. Empirical and statistical validation demonstrates that the improved trading strategy significantly outperforms a passive "Buy and Hold" approach.

The system achieves higher average returns with nearly half the volatility, leading to a superior Sharpe ratio compared to the benchmark. These findings indicate the possibility of outperforming the cryptocurrency market through active management using technical analysis. The implications of this study challenge the Efficient Market Hypothesis (EMH), as it demonstrates the potential for achieving higher risk-adjusted returns through systematic trading strategies. This study provides valuable insights for traders and investors seeking to optimize their strategies in the volatile cryptocurrency market, highlighting the viability of systematic trading based on technical analysis.

Keywords:

Technical analysis, trading strategy, trading system, quantitative finance, Investment Strategies, Trading Patterns, financial market, cryptocurrency, algorithmic trading, systematic trading, graphical analysis, Relative Strength Index, Average True Range, Backtesting, EMH, Efficient market, market anomalies, Trend Following, Mean Reversion, Equity Curves

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
Louvain School of Management
Place des Doyens, 1 bte L2.01.01, 1348 Louvain-la-Neuve
Boulevard Emile Devreux 6, 6000 Charleroi, Belgique
Chaussée de Binche 151, 7000 Mons, Belgique
www.uclouvain.be/lsm

During the preparation of this master's thesis, the authors utilized Chat GPT (Version 4o) for the following purpose:

1. **Clarifying Expression**: The AI tool was employed to refine and articulate our ideas clearly and professionally. It assisted in rewriting sections of the thesis to ensure precision and coherence in the presentation of our concepts.

2. **Literature Review Assistance**: The AI tool was also used to support the literature review process. With a predefined structure in place, the AI helped suggest relevant articles and sources that could bolster our arguments and provide additional context to our research.

After using chat GPT, the authors diligently reviewed and edited the content produced by the tool. We take full responsibility for the final content presented in this thesis.

By signing this declaration, we affirm that the content of this master's thesis reflects our original work, augmented by the responsible use of AI.

Nathan Quiévreux:

Julien Adam:

Signed the 1st August 2024

Acknowledgment & Dedication

Acknowledgment

We would like to express our appreciation to our primary supervisor, Professor Leonardo Iania, who guided us throughout this project.

Dedication

We would like to dedicate this thesis to our respective families and friends for their devoted and undeviating support.

Contributions

Nathan Quiévreux was responsible for developing the theories and designing the trading system based on his market experience. He also authored the thesis, conducted the result analyses, and proposed improvements for the trading system. Julien Adam provided essential technical support, including programming and backtesting the trading system and the data analysis graphs used to evaluate the system's performance.

Contents

List of Figures

List of Tables

1 Introduction

Is it possible to consistently outperform the market using technical analysis, especially in the highly volatile cryptocurrency market? This study explores the application of technical analysis as a tool for developing trading systems that aim to achieve superior risk-adjusted returns compared to traditional market benchmarks. By challenging the Efficient Market Hypothesis (EMH)—which asserts that all available information is already reflected in asset prices, making it impossible to consistently achieve above-market returns—this research delves into new frontiers in trading strategy development.

Despite its widespread use, technical analysis often faces skepticism, particularly from proponents of the EMH, who argue that consistent outperformance is unachievable. Traditional financial theories like the EMH have long dominated investment strategy development. However, anomalies and market inefficiencies, especially evident in volatile environments such as cryptocurrency markets, offer compelling opportunities for exploitation. Existing research has extensively examined technical analysis and various trading strategies, yet there remains a considerable gap in effectively synthesizing systematic and discretionary approaches within a single cohesive framework.

Our approach challenges the EMH by employing a semi-systematic methodology. This method combines discretionary insights derived from personal experience with a systematic framework that establishes clear entry and exit signals. Such a hybrid approach aims to bridge the gap between subjective analysis and objective, data-driven decision-making. It addresses a significant research gap by integrating the precision of systematic methods with the adaptability of discretionary trading, which is particularly suited for the dynamic and often unpredictable nature of cryptocurrency markets.

While substantial research has examined technical analysis and various trading strategies, a considerable gap remains in the effective synthesis of systematic and discretionary approaches within a single framework. Our research contributes to this area by demonstrating how a balanced, semi-systematic approach can mitigate the inherent limitations of each method. This approach not only challenges the core tenets of the EMH by suggesting that not all market inefficiencies are immediately and fully absorbed into asset prices but also offers a robust framework capable of capturing transient opportunities in highly volatile markets.

Research indicates that while systematic trading reduces human bias and enhances the precision of backtesting (Fischer, 2018; Man Institute, 2017)[16], it often struggles to adapt to sudden market shifts and may overfit to historical data (MDPI, 2020)[1]. Conversely, discretionary trading capitalizes on traders' intuition and flexibility, essential in responding to unexpected market changes (Man Institute, 2017; CME Group, 2024)[18][17][2], but is often vulnerable to emotional biases which may impair decision-making (CME Group, 2024)[17].

This backdrop underscores a significant research gap: the potential synergy of a semi-systematic approach that leverages the strengths of both systematic precision and discretionary flexibility has not been thoroughly explored. Addressing this gap could lead to the development

[1] `https://www.mdpi.com/2227-7099/8/1/20`

[2] `https://www.cmegroup.com/education/courses/trade-and-risk-management/system-based-vs-discretionary-trading.html`

of more robust trading strategies, especially suited for the complex dynamics of cryptocurrency markets.

Recognizing the significant potential of technical analysis in the highly volatile cryptocurrency market, this research aims to provide empirical evidence that systematic technical analysis can identify and exploit market inefficiencies. The core research problem we seek to address is:

Can a trading strategy based on technical analysis indicators outperform the cryptocurrency market in terms of risk-adjusted returns?

The importance of this research lies in its potential to challenge the EMH by providing empirical evidence that systematic technical analysis can identify and exploit market inefficiencies. If effective, technical analysis could offer traders a significant competitive advantage, demonstrating that markets are not always perfectly efficient and that opportunities for superior risk-adjusted returns exist.

Our primary objective is to develop a trading strategy that achieves better risk-adjusted returns compared to the benchmark, demonstrating robustness and consistency across various market conditions. This involves not only generating higher returns but also managing risk effectively to ensure the strategy can withstand market volatility and drawdowns while maintaining profitability.

We hypothesize that the cryptocurrency market exhibits anomalies that can be systematically exploited through the identification and application of technical patterns. These patterns, derived from historical price data, are expected to provide reliable signals that enhance the performance of a trading strategy beyond what is achievable through a passive "buy and hold" approach. Specifically, we anticipate that our strategy will yield a higher Sharpe ratio on average than the benchmark.

Our methodology involves a two-phase approach:

Discretionary Phase: Drawing on our personal trading experience, we develop an initial trading strategy based on identified technical patterns and market behavior. This phase emphasizes subjective analysis and intuition to craft a strategy tailored to observed market dynamics.

Systematic Phase: The discretionary strategy is then translated into a mechanical trading system with clearly defined entry and exit rules. This systematic approach allows for rigorous statistical testing and validation, ensuring the strategy's robustness and adaptability to different market conditions. By quantifying the strategy's performance metrics, we aim to fine-tune and optimize the system, enhancing its overall effectiveness and reliability.

The research will be structured as follows:

- Literature Review: An examination of existing theories and methodologies in technical analysis and market efficiency.

- Development of the Trading System: This phase includes generating trading pattern ideas based on personal experience and systematizing them through clear entry and exit rules.

- Backtesting and Validation: Application of the trading system to historical data, with a comprehensive analysis of performance metrics such as returns, volatility, Sharpe ratio, and drawdown.

- Data analysis: Analysis of results and categorization according to different system parameters, market conditions, and potential technical signals tracked during the backtest.

- System improvement points: Based on the findings from the data analysis, establish potential improvements and optimizations to the trading system. This phase rigorously considers trade-offs between metrics (Risk-reward ratio vs. winning rate and number of occurrences vs. winning rate) impacting the system's overall profitability.

- Backtest and final results: Application of the improved trading system to a new, diversified dataset encompassing various market conditions. Present performance metrics of the trading system and benchmark comparisons.

- Conclusion and Future Research: Summarization of the study's contributions, implications for the EMH, and potential directions for further investigation, including dynamic position sizing, leveraging, and expansion to other markets and time frames.

By following this structured approach, we aim to provide a comprehensive analysis of technical analysis as a tool for developing a competitive trading edge, contributing valuable insights to the ongoing discourse in financial trading and market efficiency.

2 Methodology & Data

In this section, we outline the methodology and data sources utilized for developing and testing our trading strategy. Our primary goal is to construct a robust and systematic trading system capable of outperforming the benchmark, specifically within the cryptocurrency market. The benchmark selected is a passive "Buy and Hold" strategy applied individually across the ten leading cryptocurrencies, as detailed in Appendix A.1.2.

We compare our trading strategy against a "Buy and Hold" benchmark for each of the ten cryptocurrencies. The key performance metrics employed are the Sharpe ratio, return, and maximum drawdown. These metrics provide a comprehensive assessment of risk-adjusted returns, overall profitability, and potential risk exposure, respectively. In addition to these primary metrics, we also consider other fundamental metrics such as volatility, the correlation and market exposure of the strategy, the number of trades executed, etc. All these metrics are thoroughly developed and explained in Appendix A.3.

The backtesting of the trading strategy is conducted using the R programming language for quantitative analysis, with data analysis and visualization facilitated through Excel. This dual approach allows for rigorous testing and detailed observation of results. The statistical tests are performed to confirm the validity of our observations, ensuring that the findings are statistically significant and not merely a result of random market fluctuations.

The development of the trading strategy follows a structured flowchart (see figure 1), guiding the transition from a conceptual idea to a fully operational trading system. The core indicators employed in the strategy are the Relative Strength Index (RSI) and the Average True Range (ATR). The RSI serves as the foundation for identifying potential entry points, while the ATR is crucial for the money management aspect, particularly in setting stop-loss levels. Additionally, a trend detection indicator based directly on price action is integrated into the system to enhance the robustness of trend-following mechanisms.

The framework is structured into eight main steps, each encompassing specific sub-steps that guide the process from the initial hypothesis formulation to the final review and improvement. The flowchart serves as a roadmap, illustrating the sequence of actions, decision points, and feedback loops involved in creating and refining a trading strategy.

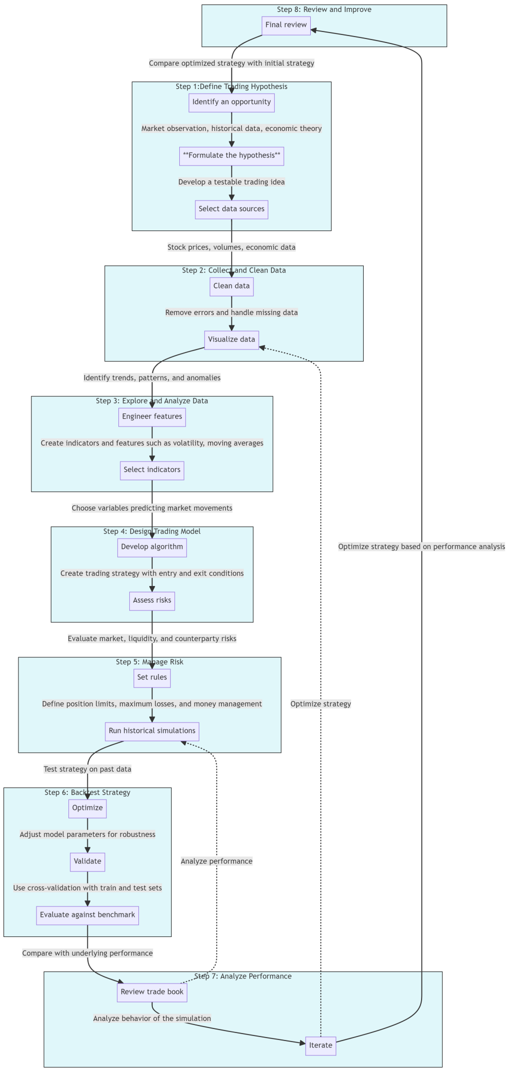


Figure 1: Trading Strategy Development Methodology Flowchart

In section 4, we detail more how to get our trading system ready for backtesting.

In this thesis, a semi-systematic approach was adopted for the development of the trading strategy. This approach combines intuition and personal experience with systematic and objective rules, allowing for flexibility in strategy design while maintaining scientific rigor. The process begins with the generation of trading strategy ideas, based on personal experience and market knowledge. This initial phase aims to formulate hypotheses that may reveal exploitable market patterns. These ideas are then systematized into clear and programmable rules, precisely defining the conditions for entering and exiting trades.

Once the rules are established, the trading system undergoes backtesting using a sample dataset, which enables the assessment of its initial performance and viability. The results obtained are thoroughly analyzed, categorizing them according to the system parameters and various market conditions. This analysis provides a detailed view of the system's performance, employing rigorous statistical methods. During this step, categorization can also be extended to other potential technical signals tracked during the backtest.

Based on this analysis, the strategy is refined by adjusting the entry and exit conditions according to the best parameters and observed market conditions or technical signals. Special attention is given to the system's architecture and the trade-offs between different performance metrics, such as the winning rate versus the risk-reward ratio, and the number of opportunities versus the winning rate, which directly influence the system's expected profitability. The improved trading system is then retested on a new, diversified dataset that includes various market conditions to minimize the risk of overfitting.

This process can be iteratively repeated, with phases of analysis and backtesting occurring in succession until the final version of the trading system is achieved. This continuous feedback loop ensures constant improvement and optimization of the trading strategy.

The dataset utilized in this study includes market data captured at 15-minute and 1-hour intervals, covering the period from September 2020 to February 2024. The dataset comprises essential variables such as open, high, low, and close prices, as well as trading volumes and the number of trades. These data points are crucial for conducting thorough technical analysis.

To facilitate rigorous testing and optimization, the dataset was divided into two halves. The first half served as a training set for initial strategy development and preliminary analysis. This phase allowed us to refine our trading strategy by implementing various optimizations and adjustments in the section 5.4 by analyzing in depth the behavior of our trading strategy under different market conditions and parameters. The second half of the dataset was reserved as a testing set, enabling us to validate the improved trading strategy on a fresh set of data. This division ensures that our findings are not only accurate but also generalizable, preventing overfitting and confirming the robustness of the trading strategy across different market conditions.

The data was sourced from Binance[3], a leading global cryptocurrency exchange known for its high trading volume and wide range of supported digital assets. Binance's comprehensive data coverage, transparency, and focus on spot market transactions ensure the reliability and accuracy of the data used in this analysis. Furthermore, to enhance the realism of our findings, we accounted for Binance's transaction fees in our calculations. This inclusion ensures that the reported performance metrics reflect the potential real-world outcomes, considering the costs associated with trading.

[3]`https://www.binance.com/en/landing/data`

3 Literature review

3.1 Human behaviour creates patterns on markets

Human behavior is inherently psychological and emotional, leading to the creation of patterns in financial markets. Behavioral finance studies suggest that investors are not always rational and are influenced by biases, heuristics, and market sentiments. These psychological factors can create predictable patterns that technical analysis aims to exploit.

Behavioral finance, a field that blends psychology with financial theory, examines how cognitive biases and emotional responses influence investment decisions and market outcomes. Notable contributions to this field include the work of Kahneman and Tversky (1979)[22] on prospect theory, which explains how people make decisions involving risk and uncertainty. They found that investors often overreact or underreact to market events, leading to mispricings that can be exploited through technical analysis (Kahneman & Tversky, 1979)[22].

Additionally, Shefrin (2000)[39] discusses the psychological underpinnings of investor behavior, illustrating how emotions and cognitive biases create market patterns. He emphasizes that understanding these behavioral patterns is crucial for developing effective trading strategies (Shefrin, 2000)[39].

Barberis and Thaler (2003)[5] provide a comprehensive overview of behavioral finance, explaining how investor psychology affects market outcomes. They highlight various biases, such as overconfidence and loss aversion, that can lead to predictable market anomalies (Barberis & Thaler, 2003)[5].

Concrete examples of how human behavior creates market patterns can be seen in phenomena such as herding behavior, where investors follow the actions of others, leading to trend formations. For instance, during a market rally, the initial rise in asset prices may attract more investors, further driving prices up in a self-reinforcing cycle (Shiller, 2000)[40]. Conversely, panic selling can lead to sharp declines as investors collectively rush to exit their positions.

Another example is the impact of overconfidence on trading behavior. Barber and Odean (2001)[4] found that overconfident investors tend to trade more frequently, often resulting in lower net returns due to increased transaction costs and poor timing. This trading pattern can create short-term price movements that technical analysts may exploit (Barber & Odean, 2001)[4].

Market anomalies such as the January effect, where stock prices tend to rise more in January than in other months, also illustrate how investor behavior affects market patterns. Thaler (1987)[46] attributes this anomaly to tax-related selling at year-end followed by repurchases in January, driven by individual investors' psychological and financial decision-making processes (Thaler, 1987)[46].

These examples demonstrate the link between human behavior and market patterns, providing a foundation for the use of technical analysis as a tool to predict and capitalize on these patterns.

Can Behaviour patterns be Captured ?

The ability of traders and investors to capture market patterns has been the subject of extensive research. Studies have shown that certain trading strategies based on these patterns can indeed be profitable, though the success often depends on various factors including the strategy's implementation and the market conditions.

For instance, Lyle and Yohn (2017)[30] analyzed the profitability of trading strategies based on academic findings about earnings announcements. They found that investors could profit from these patterns, particularly the post-earnings-announcement drift, where stock prices tend to move in the direction of an earnings surprise for a few days to weeks after the announcement. Their research highlights that these strategies can be effective when properly implemented, accounting for real-world trading constraints (Lyle & Yohn, 2017)[30].

Machine learning and artificial intelligence have also been applied to capture and exploit market patterns. Studies have shown that models like logistic regression, neural networks, and support vector machines can effectively predict stock market movements by recognizing patterns in historical data. For example, Fischer et al. (2018)[16] found that Long Short-Term Memory (LSTM) networks outperformed other models in predicting directional movements of SP 500 stocks, demonstrating the potential of advanced algorithms in trading (Fischer et al., 2018)[16].

Sentiment analysis has emerged as another powerful tool for predicting market volatility and returns. Research by Tetlock (2007)[45] and subsequent studies have shown that sentiment extracted from news articles, financial reports, and social media can predict stock market behavior. For instance, Bollen et al. (2011)[8] demonstrated that Twitter sentiment could significantly improve the accuracy of predictions for the Dow Jones Industrial Average (Bollen et al., 2011)[8].

In conclusion, while capturing market patterns through trading strategies is challenging, evidence from academic research supports the notion that it is possible. The success of these strategies often hinges on the sophistication of the models used and the ability to adapt to changing market conditions.

3.2 Is technical analysis a relevant tool?

Technical analysis and fundamental analysis have long been subjects of scrutiny, with mixed evidence concerning their profitability. Unlike fundamental analysis, which has been somewhat more accepted in academic circles, technical analysis has faced significant skepticism since its critique by Fama (1970)[14]. However, its continuous evolution has drawn massive attention, particularly in the 2000s, although earlier empirical studies often faced methodological challenges.

Another aspect of the ongoing debate between these two schools of thought is influenced by investor beliefs, which are not necessarily derived from a lack of knowledge, education, or social factors, as discussed by Menkhoff (2010)[32]. Menkhoff's findings also suggest that practitioners of technical analysis often reject the efficient market hypothesis, attributing market price deviations to emotional and psychological biases, rather than informational efficiency (Carver R, Systematic trading)[11].

Our study does not aim to conclude this debate about technical and fundamental analysis but rather to empirically assess if technical strategies can outperform, provide an alternative, or offer diversification to more conventional strategies such as "Buy and Hold". This research first aims to establish a proper framework for comparing some technical strategies against "Buy and Hold" and, secondly, to determine if strategies built from technical factors can withstand varying market conditions and sustain over the long-term.

The technical approach to investment reflects the belief that prices move in trends influenced by various economic, monetary, political, and psychological forces. The art of technical analysis, as defined by Pring (2002)[37], lies in identifying trend reversals at an early stage and riding these trends until evidence suggests they have reversed.

Historically, technical analysis dates back to at least the eighteenth century with the development of candlestick charting in Japan, a technique introduced to the Western world in the 1970s by Nison (1991)[33]. Park and Irwin (2007)[35] conducted a comprehensive survey of the literature on technical analysis from the 1970s to 2007, providing significant insights into studies and practices. They found that 56 studies identified trading profits from technical trading strategies, while 20 studies reported losses, highlighting the methodological deficiencies prevalent in earlier studies.

Further advancements in the field were noted by Lukac, Brorsen, and Irwin (1988)[29], who demonstrated that technical analysis strategies, such as moving averages and channel systems, yielded significant returns even after accounting for transaction costs on futures markets from 1975 to 1984. Subsequent studies by Lukac and Brorsen (1990)[28], and Brock, Lakonishok, and LeBaron (1992)[9], supported these findings over broader asset classes and longer timeframes, confirming the enduring predictive power of technical strategies.

Nevertheless, the challenge of establishing a definitive framework for evaluating the profitability of these strategies remains, as highlighted by Park and Irwin (2007)[35]. They advocate for a rigorous, unbiased methodological framework that accounts for risk and transaction costs to make meaningful comparisons.

In summary, while there is historical profitability associated with technical analysis, the need for a robust, well-defined framework is imperative. The potential for technical strategies remains vast, as noted by Pring (2002)[37], reinforcing the need for methodological rigor in their evaluation.

3.3 Systematic VS Discretionary approach

In the realm of financial markets, the necessity for a clear and structured approach to trading cannot be overstated. This need arises from the inherent complexity and volatility of markets, driven by a multitude of economic, monetary, political, and psychological factors. The discipline of technical analysis, which aims to predict future market movements based on historical price patterns, requires rigorous frameworks to ensure its efficacy and reliability (Park Irwin, 2007)[35].

Systematic Trading

Systematic trading, also known as algorithmic trading, involves using predefined rules and algorithms to execute trades. This method reduces human bias and allows for consistent, data-driven decisions. Studies like those by Fischer and Krauss (2018)[16] highlight the effectiveness of advanced algorithms, such as Long Short-Term Memory (LSTM) networks, in predicting market trends.

Despite its advantages, systematic trading is not without challenges. The development of robust algorithms demands significant technical expertise, and these systems can be prone to overfitting, where models perform well on historical data but fail to generalize to new data (Man Institute, 2017).

Discretionary Trading

Discretionary trading, on the other hand, relies on the trader's judgment, experience, and intuition to make decisions. This approach allows for flexibility and adaptability, enabling traders to respond to unexpected market events and changing conditions. Discretionary traders can incorporate qualitative factors, such as news events and market sentiment, which may not be easily quantifiable but can have a significant impact on market movements (Man Institute, 2017)[18].

Studies comparing the performance of discretionary and systematic trading have shown mixed results. According to a study by the Man Institute (2017)[18], discretionary hedge funds sometimes outperform systematic ones, particularly in volatile and uncertain market environments where human judgment can be advantageous. However, the same study also found that systematic funds tend to perform better in stable, trend-following markets due to their ability to consistently apply their trading rules without being influenced by emotions (Man Institute, 2017)[18].

Discretionary trading allows for more nuanced decision-making, but it also has its drawbacks. Human traders are susceptible to cognitive biases and emotional reactions, which can lead to inconsistent decision-making and suboptimal performance. Moreover, discretionary trading requires continuous monitoring of the markets, which can be time-consuming and mentally demanding (Fischer & Krauss, 2018)[16].

Comparative Analysis

Choosing between systematic and discretionary trading depends on various factors including the trader's objectives, resources, and the specific market conditions. Systematic trading offers scalability and consistency, making it suitable for high-frequency and quantitative strategies. Discretionary trading provides the advantage of human insight and adaptability, which can be crucial in certain market scenarios (Basanisi & Torresetti, 2023; Fischer & Krauss, 2018; Man Institute, 2017)[6][16].

The research by Basanisi and Torresetti (2023)[6] underscores the importance of a comprehensive framework for systematic trading. Their findings highlight the need for methodologies that can adapt to varying market conditions and ensure the reliability of trading strategies. This research is crucial in balancing the strengths and mitigating the weaknesses of both systematic and discretionary approaches.

Systematic strategies ensure consistency and can be scaled to handle large volumes of trades. This makes them particularly effective for high-frequency trading and quantitative strategies where speed and precision are crucial (CME Group, 2024)[17]. On the other hand, discretionary trading shines in complex, rapidly changing market conditions where human intuition and experience can provide a competitive edge (CME Group, 2024)[17].

Research indicates that systematic trading has generally performed well in stable market environments due to its rule-based nature. However, it may struggle in periods of significant market shifts unless the algorithms are updated to adapt to new conditions (CME Group, 2024)[17]. Discretionary trading, while potentially more profitable in volatile markets, can suffer from human errors and emotional biases, making consistency a challenge (CME Group, 2024)[17].

Recent studies, such as those by Basanisi and Torresetti (2023)[6], emphasize the importance of a robust framework for systematic trading. Their findings highlight the need for comprehensive methodologies that can adapt to varying market conditions and ensure the reliability of algorithmic strategies. This research is crucial in balancing the strengths and mitigating the weaknesses of both systematic and discretionary approaches.

3.4 Research gap and our positioning

The final chapter of this section provides an ideal transition to introducing our methodology, the semi-systematic approach. This approach combines the strengths of both systematic and discretionary trading methods, aiming to capitalize on their respective advantages while mitigating their drawbacks.

Semi-Systematic Approach

The semi-systematic approach involves using discretionary techniques to identify potential trading opportunities, which are then rigorously analyzed and executed using systematic, rule-based methods. This hybrid approach leverages the flexibility and intuitive strengths of discretionary trading while maintaining the discipline and consistency of systematic trading.

Literature Review and Research Gap

While the literature on purely systematic and discretionary trading is extensive, studies specifically examining the semi-systematic approach are relatively sparse. Most research has focused on the advantages and disadvantages of each method independently.

Research suggests that systematic trading offers a structured and disciplined approach, minimizing human error and emotional bias. It enables backtesting strategies on historical data, which helps refine and optimize them before live trading (Fischer, 2018 ; ManInstitute, 2017)[16]. However, the main drawbacks include a potential lack of adaptability to sudden market changes and the risk of overfitting models to historical data, leading to poor performance in real-time trading (MDPI, 2020)[4].

[4] https://www.mdpi.com/2227-7099/8/1/20

In the other hand, Studies suggest that discretionary trading benefits from the trader's experience and intuition, allowing for quick adaptation to unexpected market conditions. This flexibility can be crucial in volatile markets (ManInstitute,2017; CMEGroup, 2024)[18][17][5]. However, it is prone to emotional biases, such as overconfidence and fear, which can lead to inconsistent decision-making and performance (CMEGroup,2024)[17]

Current Research and Our Position

The semi-systematic approach, combining elements from both strategies, is relatively underexplored in academic literature. A few studies and industry reports highlight its potential. Technology and data analysis play crucial roles in semi-systematic trading, utilizing advanced technology for data collection and analysis, enabling traders to develop and test strategies based on historical data while incorporating real-time adjustments (MDPI, 2020)[6]. This approach allows for the integration of human judgment in strategy development and execution, providing flexibility while maintaining systematic rigor (CitadelSecurities, 2024)[7].

Given the limited academic exploration of this hybrid method, our research aims to fill this gap. We will systematically investigate the efficacy of semi-systematic trading strategies, focusing on their ability to adapt to varying market conditions and optimize performance by combining discretionary insights with a systematic framework.

Adding to this, our approach aims to be as realistic as possible by including spot data from brokers and incorporating real transaction fees to ensure the realism of our presented results challenging the conventional omission of real-market condition constraints in many studies.

Moreover, our personal research endeavor seeks to identify exploitable patterns and construct a competitive edge: developing a system based on more advanced signals rather than basic ones like buying when the RSI is oversold or crossing moving averages. The intent is to design a unique system not reliant on widely known signals, thereby preserving effectiveness as popular strategies often degrade when widely adopted. This insight aligns with observations of successful trading strategies that have diminished in profitability due to market evolution; strategies with a limited lifespan tend to be simplistic.

Furthermore, our adaptive approach continually adjusts to the dynamic market environment, analyzing the efficacy of our system and refining it in response to market feedback, in contrast to many studies that do not modify popular signals to suit changing market conditions.

Initially, a discretionary search for market patterns informs our semi-systematic approach, later codified for consistent, rational execution. It ensures the development of novel, non-mainstream strategies, enhancing potential success due to lower predictability and competition.

Finally, a rigorous and scientific application of technical analysis serves not merely as a standalone tool but as part of a broader systematic approach to detect, codify, and validate market patterns through statistical testing, ensuring robustness and reliability of our signals.

---

[5] `https://www.cmegroup.com/education/courses/trade-and-risk-management/system-based-vs-discretionary-trading.html`

[6] `https://www.mdpi.com/2227-7099/8/1/20`

[7] `https://www.cmegroup.com/education/courses/managed-futures/comparing-cta-strategies.html`

4 Framework

In this comprehensive section, we will look at the entire process we follow to achieve the results of our trading strategy.

Firstly, we will discuss in the section "4.1 Pattern identification" why we chose the cryptocurrency market and the use of technical analysis, as well as the different technical indicators we will use to build our trading system. This section will also cover the steps required to transform a conceptual idea into a practical trading system ready to be tested.

Next, we will discuss in the "4.2 Trading system building" key considerations in the development of the trading system by exploring crucial concepts such as the trade-off between metrics such as the win rate and the risk-reward ratio, as well as the trade-off between the win rate and the number of opportunities within a system. We will then develop our trading system in concrete terms, focusing on the central concept underlying our strategy, namely the flexible neutrality zone of the RSI indicator.

Finally, in the "4.3 Algorithm structure" we will look at the more technical aspects of implementing the system in code and the overall backtesting infrastructure. In particular, we will discuss the programming language chosen for the quantitative tests, the key coding functions and the machine learning techniques envisaged for our analysis.

4.1 Pattern identification

In this section, we will explore the preliminary components essential for the composition of our trading system.

First, we will share our rationale for using technical analysis as a differentiation strategy against large institutions and hedge funds, and we will explain our decision to focus our tests on the cryptocurrency market.

Next, we will develop the technical analysis tools that we will use in the creation of our trading system. Specifically, we will examine how the RSI and price action can aid in creating a pattern-based strategy, and how the ATR can be a formidable tool for developing robust money management within the system.

Finally, we will detail the steps involved in our methodology for developing a trading system from an initial idea. This includes using technical analysis tools to identify patterns that align with the idea, systematizing these patterns for better implementation and statistical analysis. Additionally, we will present a series of examples applied to our methodology, which could serve as a basis for further research.

In summary, this section aims to provide a clear and comprehensive understanding of our methodological approach, setting the stage for the construction of our trading system, which will be presented in the next section.

4.1.1 Technical Analysis and the Cryptocurrency Market: Developing Our Edge

In this thesis, we will focus exclusively on the use of technical analysis (TA) to develop trading strategies. Although TA is often considered subjective, this characteristic can actually create a competitive edge in financial markets. Unlike price modeling based on interdependent variables, which is largely dominated by financial institutions such as hedge funds and banks, TA offers individual traders a unique opportunity to compete.

It may seem paradoxical, but the advantage of TA is that it is a highly subjective practice and you can therefore really develop a competitive advantage on the financial markets. When modeling prices with interdependent variables, it is often thought that competing with large players such as hedge funds and banks (institutional investors) is difficult, if not impossible, due to their substantial human, financial, technology and data resources.

The objective and methodical nature of the modeling employed by these institutions makes the process easily replicable and scalable. However, according to the theory of market efficiency, once a flaw is observed in the markets, it will be quickly exploited and corrected by these institutions. Therefore, it is crucial to differentiate and find exploitable flaws in a manner that these large players do not employ, as it is practically impossible to compete without similar financial, technological, and human resources.

In this context, it is necessary for individual traders to seek their edge by venturing into different territories than those occupied by institutional investors. Algorithmic trading, such as arbitrage, is reserved for an elite group of institutions that possess the required technological, financial, and human resources. In this case, no individual trader can realistically make money using the traditional arbitrage techniques employed by hedge funds.

In this thesis, we will strive to think outside the box because we firmly believe that this approach will allow us to create profitable trading strategies and develop an edge in the markets. The goal is to observe what others (institutional investors) do not and to use different techniques from those traditionally employed by large institutional players.

Technical analysis is ultimately a tool for modeling price movements, similar to regression, which is commonly used as a modeling technique. Like any modeling technique, it must be used rigorous to yield significant results.

The way we seek patterns is entirely subjective and not derived from machine learning techniques or scientifically/rigorously approved processes. We believe that it is through this initial subjectivity that it is possible to differentiate and build an edge in financial markets. Our aim is to observe what has not been observed by large institutional players. We will seek exploitable patterns in the markets differently from institutions.

Some funds find their edge through privileged data that few have access to, others through cutting-edge technological means (notably arbitrage funds with fast market connections and advanced technological capabilities). Other funds find an edge in the market through their expertise in modeling or machine learning. Aware that it is difficult, if not impossible, to compete with well-established major players on these fronts, we choose to seek an edge by applying entirely different modeling techniques using technical analysis.

Far from being an innovative tool, we believe that technical analysis is underutilized by major players. In this thesis, we will explore how technical analysis can be used to create profitable trading strategies. We believe that by observing what institutions overlook and adopting a different approach, we can develop a unique edge. The goal is to demonstrate that, through rigorous and initially subjective analysis, it is possible to succeed in financial markets.

We will focus our pattern research on price action and the RSI indicator. This decision to concentrate on specific tools allows us to limit the scope and increase the depth of our analysis. By reducing the number of variables and possibilities, we can better understand and exploit the identified patterns.

Our tests will primarily be oriented towards the cryptocurrency market. This choice is motivated by several reasons:

1. **Market Youth**: The cryptocurrency market is relatively young and still underinvested by institutional players, offering unexploited arbitrage opportunities.
2. **Volatility**: Cryptocurrencies are known for their high volatility, which is essential for trading as it creates frequent and potentially lucrative opportunities.
3. **Psychology and Emotions**: This market is largely dominated by individual investors, making it more sensitive to human psychology and emotions. The abrupt movements of euphoria and panic (FUD) create ideal conditions for identifying exploitable patterns based on human psychology.

By focusing on a specific niche of assets (cryptocurrencies) and a particular tool (technical analysis), we adopt an initially very subjective process. This approach allows us to differentiate ourselves, much like a startup competing against well-established oligopoly players through a product differentiation strategy.

The central idea is that the market results from human interpretations of available information at a given moment. These interpretations, although not always accurate, guide the market direction. Thus, modeling and anticipating market movements is partly about modeling human psychology rather than seeking an absolute truth.

Financial movements often originate in human psychology, and this is especially true for cryptocurrency markets, which have no fundamental value and are primarily speculative. In this thesis, we will mathematically model exploitable fragments of human psychology using the vast amounts of data available in financial markets through the tool of technical analysis.

Our approach opposes the market efficiency hypothesis and the idea that all market participants are rational. We believe that subjectivity and emotions play a crucial role in market movements, and it is precisely by exploiting these aspects that we can develop profitable trading strategies. By focusing on technical analysis and cryptocurrency markets, we aim to identify and exploit patterns that escape traditional institutions, thus creating a genuine edge.

4.1.2 Technical analysis tools

In this section we will develop the various technical analysis tools that we will use to design our trading system. First, we will develop the RSI indicator, which will be the central pillar of our strategy. We will then see how we can improve the strategy with a confluence of signals by adding filters on the price action. Finally, we will develop the ATR indicator for adaptive and flexible money management in our trading system.

Relative Strength Index (RSI)

The Relative Strength Index (RSI) was developed by J. Welles Wilder Jr. and introduced in his book "New Concepts in Technical Trading Systems" in 1978. This publication marked a significant milestone for technical traders, providing several new tools and indicators for market analysis. Wilder designed the RSI to measure the speed and magnitude of price movements, thereby offering a way to identify overbought and oversold conditions. Since its introduction, the RSI has become a standard tool in technical analysis, used by traders to anticipate trend reversals and optimize their trading decisions. Essentially, it is an indicator that can be used to detect market excesses, both to the upside and downside.

The RSI is a momentum oscillator that quantifies the strength and speed of price movements of an asset by comparing the magnitude of recent gains to recent losses. It oscillates between 0 and 100, with reference levels commonly set at 70 and 30. An RSI above 70 suggests that the asset may be overbought and could be poised for a downward correction, while an RSI below 30 indicates that the asset may be oversold and could be ready for an upward rebound.

Traders also use the RSI to identify divergences between the indicator and asset prices, which can signal an imminent price reversal. For example, if the price of an asset reaches a new high but the RSI does not confirm this high with a new peak, it may indicate weakening momentum and a potential trend reversal. Additionally, some traders apply more advanced techniques, such as analyzing adjusted RSI levels for different market environments or using the RSI's midline (level 50) to identify the general direction of market momentum.

The calculation of the RSI begins with determining the Relative Strength (RS), which is the ratio of the average closing prices of up periods to the average closing prices of down periods over a given time frame. Formally, the RSI is calculated as follows:

$$\mathrm{RSI} = 100 - \left(\frac{100}{1+\mathrm{RS}}\right) \tag{1}$$

where:

$$\mathrm{RS} = \frac{\text{Average Gain}}{\text{Average Loss}} \tag{2}$$

Gains and losses are first calculated as the positive and negative differences between successive closing prices. Then, the average gains and losses are typically computed using an exponential moving average for greater accuracy, giving more weight to recent data.

The calculation period of the RSI is a critical parameter that affects the indicator's sensitivity to price movements. Wilder recommended a 14-day period, which strikes a balance between sensitivity and the reliability of generated signals. A shorter period makes the RSI more responsive to recent price changes, which can be beneficial in volatile markets or for short-term trading strategies. However, this also increases noise and the likelihood of false signals. Conversely, a longer period smooths the indicator, reducing false signals but potentially delaying the response to new market movements, which might be more suitable for long-term strategies or investors preferring a less aggressive approach.

This concept will be explored in more detail in the Tradeoff sectionof this thesis, but it is essential to understand the trade-off between the winning rate and the risk/reward ratio influenced by the RSI period setting. A shorter RSI period allows for more optimal entries, capturing a greater amplitude of market movements, which tends to improve the risk/reward ratio. However, this setup increases signal volatility, potentially reducing the winning rate due to lower reliability. On the other hand, increasing the RSI period, although resulting in less optimal entries and capturing less price movement, improves the winning rate due to more stable signals and better market noise filtration. This trade-off necessitates careful adjustment of RSI parameters to align trading strategies with the trader's specific goals and risk tolerance.

Statistical properties:

The first relevant property is that the RSI is a bounded indicator, oscillating between the fixed values of 0 and 100. This characteristic means that the RSI can never exceed these limits, regardless of the magnitude of the underlying asset's price movements. This intrinsic property of the RSI provides several significant advantages:

Firstly, the bounded range of the RSI allows for clear and consistent interpretation of trading signals. Standard levels of overbought (70) and oversold (30) provide universally accepted reference points, facilitating quick decision-making and coherent trading strategies. The RSI's consistency also makes it easy to compare overbought and oversold levels across different assets and time periods, simplifying comparative analysis and helping traders identify cross-asset opportunities.

Additionally, the RSI smooths out erratic price fluctuations and reduces market noise, allowing traders to focus on significant trends and improving the quality of trading signals. Its bounded nature excels at identifying extreme market conditions, providing clear indications of potential price reversals and enabling traders to strategically position their entries and exits for optimal profit. Therefore, it is a useful indicator for exploiting market excesses both to the downside and upside, particularly when included in a mean reversion strategy.

The RSI is particularly suitable for automated trading strategies due to its fixed levels, which simplify the programming of trading rules. Algorithms can easily execute buy or sell actions based on RSI thresholds (e.g., buying when the RSI drops below 30 and selling when it exceeds 70), ensuring reliable implementation of automated strategies.

Using the RSI as a bounded indicator also allows for more effective risk management. Traders can set clear thresholds for overbought and oversold levels, helping to limit potential losses and maximize gains. For example, by setting stop-loss orders based on RSI levels, traders can better manage their positions and minimize the impact of adverse price movements.

In summary, the RSI's bounded nature provides clear signal interpretation, easy comparability, effective noise filtering, identification of extreme market conditions, suitability for automation, and improved risk management. These advantages make the RSI a valuable tool for creating and optimizing trading strategies. We will see in section 4.2.2, during the design of our own trading system, that the RSI indicator will be the central pillar of our strategy.

The RSI is a lagging indicator because it is based on past prices to calculate its current value. This means the RSI reacts to price changes with a delay, which can impact its use in trading strategies. The lagging nature of the RSI can influence the speed of trading decisions, as signals may appear after a price movement has already begun, potentially causing a delay in trade execution. To compensate for this, some traders combine the RSI with "leading indicators" to obtain a more comprehensive market view.

As a lagging indicator, the RSI offers distinct advantages and challenges for traders. Its ability to confirm trends, filter out false signals, and identify potential reversal points makes it valuable for long-term trading strategies. However, traders must be aware of its limitations and adjust their strategies accordingly. By combining the RSI with other indicators and adjusting its parameters, traders can maximize its utility and improve the accuracy of their trading decisions. Additionally, integrating RSI with price action analysis can further enhance its effectiveness.

The combined use of the RSI and price action analysis significantly enhances the reliability of trading signals by leveraging the strengths of both methodologies. Price action analysis, which focuses on the study of raw price movements without relying on technical indicators, aims to identify patterns, support and resistance levels, and market trends. By integrating the RSI, traders can obtain additional confirmations for their price action-based decisions, thereby reducing the risk of false signals.

For instance, when price action suggests a potential trend reversal or continuation, the RSI can offer valuable confirmation. If price action indicates a bullish candlestick pattern at a key support level, an RSI reading in the oversold zone (below 30) can reinforce this bullish reversal signal. This dual confirmation increases the probability that the signal is valid. However, we will see that there is another trade-off between the winning rate and the total number of market opportunities when adding signals to a strategy, as discussed in Tradeoff section

In summary, the combination of RSI and price action analysis improves the reliability of trading signals by providing additional confirmations, filtering out false signals, and identifying divergences and key market levels. These complementary approaches enhance the accuracy of entry and exit points, enabling traders to make more informed decisions and optimize their trading strategies. In the section 4.2.2, we will discuss how the RSI will serve as a central component in the development of our trading system, particularly in establishing our entry conditions according to our strategy.

Price action

We have seen that the RSI is a momentum indicator with properties that are highly relevant for developing a trading strategy. However, being a lagging indicator, it can be advantageous to use it in conjunction with price action to identify patterns with a higher probability of success. In this section, we will explain what price action is and provide various examples of its application.

Price Action is a market analysis method that relies exclusively on historical and current price movements, without using additional technical indicators like moving averages or oscillators.

This approach prioritizes direct observation of prices, operating on the premise that all necessary information for trading decisions is contained within the price data itself. It assumes that prices reflect all external variables influencing a market, making price fluctuations sufficient for predicting future trends.

Trend detection

A fundamental aspect of price action analysis is the ability to identify market trends. A trend can be bullish, bearish, or neutral (sideways). Price action typically identifies an uptrend by a series of higher highs and higher lows, or a downtrend by a series of lower highs and lower lows. Correctly recognizing and interpreting these trends allows traders to align with the overall market movement, optimizing their position-taking.

Support and resistance levels are critical components in price action analysis. A support level is a lower price point where the market tends to find enough buyers to prevent the price from falling further, whereas resistance is an upper price point where increased seller presence prevents the price from rising. These levels are identified by recurring price turning points around specific values. Price action traders monitor these levels as they can indicate potential reversal points or areas where the market might consolidate.

Consolidation zones, often referred to as ranges or congestion areas, form when prices move sideways between clearly defined support and resistance levels, without a notable uptrend or downtrend. These zones are important for price action traders as they may signal accumulation or distribution before a breakout and a new trend.

In essence, price action is a powerful and straightforward method of analysis that allows traders to decipher market sentiment through price movements alone. This approach requires a deep understanding of price dynamics and an ability to accurately interpret the signals the market provides through its natural fluctuations.

Candlestick pattern

Candlesticks are essential tools in price action trading. Each candlestick provides visual information about the opening, high, low, and closing prices for a given period, allowing traders to quickly grasp market dynamics for that period. Common candlestick types and their interpretations include the Doji, characterized by a small body with longer upper and lower shadows, indicating market indecision, often signaling a potential reversal in a trending context; Marubozu, a candlestick with no shadows, indicating that prices closed very near the high or low, representing strong buyer or seller dominance; and candlesticks with long wicks, signaling price rejection at certain levels, such as a long lower wick indicating support.

As shown by T.-H. Lu et al.[27], Figure 2 illustrates some patterns of two-day candlestick reversal patterns.

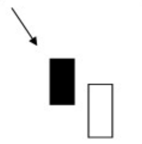
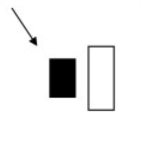
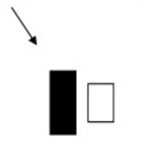
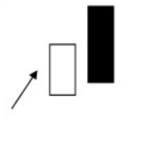
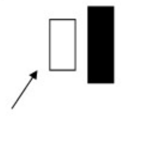
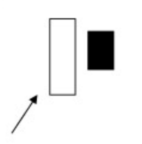
Name	Statement	Chart
Piercing	In a downtrend, following a black line the market opens lower, but closes above the mid-point of the prior candlestick's real body.	
Bullish Engulfing	In a downtrend, following a black line the market opens lower, but closes above the open of the prior candlestick's real body.	
Bullish Harami	In a downtrend, following a long black line the market opens higher than the prior close, and closes below the prior open. The second day's small real body holds within the prior long real body.	
Dark-Cloud Cover	In an uptrend, following a white line the market opens higher, but closes blow the mid-point of the prior candlestick's real body.	
Bearish Engulfing	In an uptrend, following a white line the market opens higher, but closes below the open of the prior candlestick's real body.	
Bearish Harami	In an uptrend, following a long white line the market opens lower than the prior close, and closes above the prior open. The second day's small real body holds within the prior long real body.	

Figure 2: Two-Day Reversal Patterns of Candlesticks

Beyond individual candlesticks, configurations of multiple bars can provide deeper market insights. Pin Bars, characterized by a long wick extending from a relatively small body, indicate clear price rejection and can signal imminent reversals, especially near key support or resistance levels. Engulfing Bars occur when the body of one candlestick completely engulfs the body of the previous one, suggesting a strong shift in market sentiment, with bullish engulfing bars indicating rising buyer optimism and bearish engulfing bars indicating increasing selling pressure. Inside Bars, where a candlestick's body is entirely contained within the range of the previous bar, often signal consolidation and potential pauses before a significant move in the current trend direction.

Properly interpreting these price bar configurations can provide valuable market signals and help traders position their trades more strategically. Using this information within the framework of price action allows for decisions based on clear market behavior indications, which can be particularly useful in volatile or uncertain trading environments.

However, Bernard Prats-Desclaux, in his book "Stratégies de marchés," notes through quantitative analysis that candlestick patterns alone do not provide a sufficient edge in financial markets. Nonetheless, when combined with other signals in a trading system, these patterns can be formidable, enhancing entry timing and overall trading performance. For example, a Doji, symbolizing market indecision, can effectively confirm entry points within a trading strategy when other pre-established criteria are met.

Considering a mean reversion strategy, incorporating a confirmation signal like the Doji can be highly beneficial. This not only refines trade entry timing but also increases trade success probabilities by adding an extra validation signal. Moreover, such signals can be crucial in effective risk and capital management. The ability to precisely determine the risk associated with each trade is significantly improved when money management is based on price action rather than indicators like the RSI.

Using price action to inform money management decisions enables traders to set stop-losses and profit targets based on key price levels identified by candlesticks, leading to more direct and precise risk management. This practice not only integrates technical analysis but also enriches the strategic approach by making trading decisions more robust and less dependent on the random fluctuations of indicators.

Chart patterns

Continuation and reversal patterns are essential elements of price action analysis, providing insights into probable pauses or changes in current market trends. A precise understanding of these patterns helps traders anticipate future price movements and adjust their strategies accordingly.

Continuation patterns include triangles, which can be ascending, descending, or symmetrical. Ascending triangles, characterized by a horizontal resistance line and rising support, suggest a possible bullish continuation. Descending triangles, with descending resistance and horizontal support, often indicate a bearish continuation. Symmetrical triangles, where resistance and support converge symmetrically, can break in either direction depending on the prior trend. Traders look for decreasing trading volume as the triangle forms and an increase in volume upon breakout to confirm the anticipated move. Flags and pennants are other continuation patterns, representing brief pauses or consolidations before the market resumes its previous trend. Flags show parallel channels of consolidation, while pennants are small triangular consolidations.

Reversal patterns include the head and shoulders pattern, both classic and inverse. The classic head and shoulders pattern features three peaks, with the central peak (head) being the highest and the two side peaks (shoulders) lower and roughly equal in height. The inverse head and shoulders pattern indicates a bullish reversal. The neckline, formed by the lows between the head and shoulders, is crucial; a break below the neckline confirms a bearish reversal, while a break above confirms a bullish reversal. Double tops and double bottoms are also key reversal patterns. Double tops indicate that the price has tested a resistance level twice without breaking through, suggesting a probable reversal to the downside. Conversely, double bottoms occur when the price tests a support level twice, indicating a potential reversal to the upside.

These analyses fall under the domain of chart patterns, which involves studying the figures and patterns formed by price action. Chart pattern is a fundamental component of traditional technical analysis, used to predict future market movements by interpreting various graphical configurations. However, despite its apparent utility and popularity among traders, we have decided not to incorporate specific chart patterns into our algorithmic trading systems.

This decision is based on several considerations. First, chart patterns is widely regarded as highly subjective. The identification and interpretation of chart patterns can vary significantly between analysts, introducing uncertainty and imprecision into trading strategies based solely on these observations. Second, the inherent subjectivity of chart patterns makes it difficult to replicate in a systematic algorithm. Algorithms require clear, unvarying rules, whereas interpreting chart patterns often requires adjustments and judgments based on contextual factors or trader experience.

Finally, we aim to build trading systems that can be backtested with clear, measurable criteria. Chart patterns, due to their subjective nature, pose particular challenges for precise backtesting, as confirming a pattern may depend on personal judgments rather than clear quantitative signals. Therefore, while we acknowledge the value of chart techniques in some trading approaches, we choose to focus on strategies that can be systematized and rigorously tested, enhancing the reliability and reproducibility of our trading models.

This pragmatic approach allows us to apply strict quantitative methods while avoiding the pitfalls of excessive subjectivity, ensuring that our systems are based on solid empirical data and testable methodologies.

Average True Range (ATR)

Having detailed price action and the RSI as tools to primarily develop our entry conditions in a trading strategy, we will now focus on the Average True Range (ATR) indicator. In this section, we will develop the ATR as a relevant indicator for defining exit conditions, particularly for setting stop-loss levels, to complete our trading system with robust money management.

The ATR is a technical indicator developed by J. Welles Wilder Jr. like the RSI presented earlier, but this time to measure price volatility. The ATR calculates the average of the true ranges over a defined period, typically 14 days. The true range is defined as the greatest of the following three ranges:

1. The difference between the highest and lowest prices of the current day.
2. The difference between the highest price of the current day and the previous day's closing price.
3. The difference between the lowest price of the current day and the previous day's closing price.

The formula for ATR is as follows:

$$\text{TR} = \max(\text{High} - \text{Low}, |\text{High} - \text{Previous Close}|, |\text{Low} - \text{Previous Close}|) \tag{3}$$

where:

- High is the highest price of the current period.
- Low is the lowest price of the current period.
- Previous Close is the closing price of the previous period.

$$\text{ATR} = \frac{1}{n}\sum_{i=1}^{n} \text{TR}_i \tag{4}$$

where:

- TR_i represents the True Range for each period i.
- n is the number of periods used to calculate the average.

The ATR is used in technical analysis to assess market volatility. Unlike other indicators that signal trends or reversals, the ATR focuses solely on measuring price fluctuations. A high ATR value indicates strong volatility, while a low ATR value indicates low volatility. This allows traders to adapt their strategies according to current market conditions, such as adjusting stop-loss levels to account for larger or smaller price movements.

Wilder designed the ATR to overcome the limitations of other volatility measures available at the time. Before the ATR, traders often used the simple price range (difference between the day's high and low) to measure volatility. However, this method did not account for gaps between consecutive days' closing and opening prices, which could lead to underestimating actual volatility. By including the differences between closing prices and the extremes of the current period, the ATR provides a more comprehensive and accurate measure of volatility.

Since its creation, the ATR has become an essential tool for traders and technical analysts. It is commonly used in various trading strategies, including setting stop-loss and take-profit levels and adjusting position sizes based on market volatility. The ATR is particularly valued for its versatility, applicable to all types of markets and assets, whether stocks, currencies, commodities, or futures.

Relevance of the indicator in a strategy

The Average True Range (ATR) is widely used by traders to make informed trading decisions and adjust their strategies according to market conditions. Here are some common uses of the ATR in trading strategies:

Traders use the ATR to determine appropriate stop-loss levels. An ATR-based stop-loss can be set at a distance equal to a multiple of the ATR below the purchase price (for long positions) or above the selling price (for short positions). This approach allows the stop-loss to be adjusted based on current market volatility, reducing the risk of being prematurely stopped out by normal market fluctuations. For example, if the ATR of a stock is 2 points, a trader might place a stop-loss 2 or 3 times the ATR below the purchase price to allow sufficient leeway.

Volatility measured by the ATR can be used to adjust position sizes. In highly volatile market conditions (high ATR), traders can reduce position sizes to limit their risk exposure. Conversely, in low volatility conditions (low ATR), they can increase position sizes. This approach allows for dynamic risk management based on market conditions.

The ATR can be used to filter potential trades. For example, a trader might decide to enter positions only when the ATR exceeds a certain threshold, indicating that volatility is high enough to justify the risk. Similarly, a trader might avoid trading when the ATR is very low, suggesting that the market is too calm and profit opportunities are limited.

The ATR is a powerful tool for measuring market volatility and informing trading decisions. We can use the ATR to determine appropriate stop-loss levels, adjust position sizes, and identify volatile or calm market conditions. The ATR is particularly useful for risk management strategies, as it allows protection levels to be calibrated based on current volatility. This is exactly how we will use it in our trading system. The ATR will be a tool within the system's money management framework, useful for setting coherent and flexible exit conditions according to the asset's volatility over a given period.

In conclusion, we have developed the technical analysis tools that will be used in the design of our trading system in section 4.2.2. We have examined the RSI, which will serve as our entry condition in confluence with price action. We have also explored how the ATR can be implemented in our system for optimal risk management. In the next section, we will develop our own methodology for identifying patterns in financial markets, forming the basis for a comprehensive trading system.

4.1.3 Steps to turn an idea into a trading system

In the world of trading, the development of effective strategies often relies on a deep understanding of human behaviors and market dynamics. Our methodology for generating trading strategy ideas is a structured approach that combines elements of human psychology and technical analysis. It allows for the transformation of subjective observations into systematic and robust trading systems.

Our methodology unfolds in five key steps:

1. **Formulate a Fundamental Hypothesis Based on Human Psychology**: The first step involves formulating a hypothesis based on typical investor behaviors. By understanding concepts such as fear, greed, and panic, we can anticipate market reactions and formulate initial hypotheses.

2. **Identify the Psychological Hypothesis as a Pattern in the Markets**: Next, we search for occurrences of this behavior in the markets using technical indicators like price action or RSI. This step allows us to concretely visualize how our hypothesis manifests in financial markets.

3. **Centralize Similar Patterns**: Once the patterns are identified, we group them to analyze their common characteristics. This centralization helps reinforce our initial hypothesis by showing concrete and repeated examples.

4. **Analyze in Detail How the Pattern is Constructed**: We use technical analysis tools to deeply understand the construction of the patterns. This involves using various tools like Fibonacci ratios and RSI to identify common points and recurrences.

5. **Systematize and Objectify the Pattern**: Finally, we transform these observations into a systematic trading strategy. This includes defining clear rules, conducting rigorous backtests, and optimizing parameters. The goal is to minimize emotional biases and make the strategy replicable and scalable. Automation also allows for quick execution without human intervention.

Thus, this is primarily an analytical task: visualization, identification, and analysis. The initial subjectivity of observation is gradually transformed into an objective and systematic approach.

Systematizing the trading strategy is crucial for several reasons. Transforming a subjective observation into a systematic trading strategy brings numerous benefits that enhance the consistency, reliability, and performance of trading operations. Here are the detailed reasons why systematization is essential:

- **Reduction of Subjectivity and Emotional Biases**

Trading decisions based on emotions and intuitions can often lead to errors and inconsistent performance. Emotions such as fear and greed can drive traders to make irrational decisions. By systematizing the strategy, we define clear and objective rules for entering and exiting positions. This eliminates the impact of emotions on trading decisions, ensuring that each trade is based on rigorous analysis rather than impulses. For example, a trader might panic and sell a winning position too early due to the fear of a potential loss. A systematic strategy, on the other hand, would maintain the position until the objective exit criteria are met.

- **Replicability and Scalability**

A subjective strategy heavily depends on the skills and emotions of an individual trader, which limits its replication and scalability. A systematic strategy can be easily reproduced and scaled. The codified rules allow other traders or algorithms to follow the same strategy with the expected results. This is particularly important for financial institutions that need to manage large volumes of transactions consistently and efficiently.

- **Ability to Conduct Rigorous Backtests**

It is crucial to test a strategy on historical data to evaluate its viability before applying it to live markets. By systematizing a strategy, we can code it and conduct rigorous backtests. Backtests simulate the past performance of the strategy using historical data, which helps identify its strengths and weaknesses. For example, we can test a strategy over several years of data to see how it would have performed under different market conditions.

- **Data Analysis and Statistical Testing**

Data analysis and statistical testing are essential to assess the robustness of a strategy. A systematic strategy allows for large-scale data collection, facilitating analysis and statistical testing. We can calculate metrics such as the Sharpe ratio, win rate, and drawdown to evaluate the performance and robustness of the strategy. For example, we can analyze whether a strategy produces consistent returns with acceptable volatility.

- **Potential for Automation**

Automating the trading strategy enables fast execution without human error. By coding a systematic strategy, we can automate it, allowing us to quickly seize market opportunities and manage transactions 24/7 without human intervention. This is particularly useful in volatile or high-frequency markets, where quick decisions are crucial. For example, a trading algorithm can execute thousands of transactions in a fraction of a second, exploiting market inefficiencies that human traders could not capture, although this is not the type of strategy we develop in this thesis.

In conclusion systematizing a trading strategy offers numerous advantages, including reducing emotional biases, replicability, scalability, the ability to conduct rigorous backtests, and the potential to automate transactions. By transforming subjective observations into objective rules, we can develop robust and effective trading systems that deliver consistent and reliable performance under various market conditions. This is essential for maximizing gains, minimizing risks, and ensuring a disciplined and rigorous trading approach.

Finally, to understand but also to see the unlimited potential of possibilities of trading strategies, we applied the methodology to some examples that can be retrieve in the appendices section A.5

4.2 Trading system building

In this section, we will first thoroughly develop two new essential concepts to consider when designing and improving a trading system. Subsequently, we will develop our own trading system based on technical signals.

4.2.1 Key concepts impacting system design

In this subsection, we introduce and explore three principal metrics crucial to the efficacy of a trading system: the winning rate, the risk/reward ratio, and the number of opportunities available in the market. This exploration is grounded in proprietary concepts developed specifically for this thesis. These include the trade-off between the winning rate and the risk/reward ratio, as well as the trade-off between the winning rate and the number of available market opportunities, which are contingent upon the specific configurations of a trading system. We will rigorously detail these innovative concepts through mathematical formulations, underscoring their vital importance in the strategic framework of trading system construction. Each concept reflects a novel approach, designed to enhance understanding and implementation of effective trading strategies.

Tradeoff between Winning rate & Risk/Reward ratio metrics

In any trading system, there is a fundamental trade-off between the risk/reward ratio (RRR) and the winning rate (WR). This trade-off is crucial for the overall performance of the trading system and must be carefully considered when designing and adjusting trading strategies.

The Risk/Reward Ratio (RRR) is the ratio between the average amount gained on a winning trade (Reward, R) and the average amount lost on a losing trade (Risk, L):

$$\text{RRR} = \frac{R}{L}$$

The Winning Rate (WR) is the proportion of trades that are profitable:

$$\text{WR} = \frac{\text{Number of winning trades}}{\text{Total number of trades}}$$

The Losing Rate (LR) is the proportion of trades that are losing:

$$\mathrm{LR} = 1 - \mathrm{WR}$$

So the expectancy per trade (E) of a trading strategy can be expressed as:

$$E = \mathrm{WR} \times R - \mathrm{LR} \times L$$

$$E = \mathrm{WR} \times R - (1 - \mathrm{WR}) \times L$$

For a strategy to be profitable, the expectancy must be positive:

$$E > 0$$

$$\mathrm{WR} \times R > (1 - \mathrm{WR}) \times L$$

Rearranging this inequality, we get:

$$\mathrm{WR} \times R > L - \mathrm{WR} \times L$$

$$\mathrm{WR} \times (R + L) > L$$

$$\mathrm{WR} > \frac{L}{R + L}$$

This inequality shows that the winning rate must be greater than the proportion of losses to the sum of gains and losses for the strategy to be profitable.

Using the definition of the RRR:

$$R = \mathrm{RRR} \times L$$

We can substitute this into the previous inequality:

$$\mathrm{WR} > \frac{L}{(\mathrm{RRR} \times L) + L}$$

$$\mathrm{WR} > \frac{1}{\mathrm{RRR} + 1}$$

This equation represents the condition for a trading system to be profitable. This equation shows that as the RRR increases, the WR required for the strategy to be profitable decreases. A higher RRR allows for a lower WR, but in practice, achieving a consistently high RRR is more challenging, which lowers the WR. Therefore, traders must adjust these parameters based on their objectives and risk tolerance to find an optimal balance between RRR and WR.

We will now try to explain why there is a trade-off between the winning rate (WR) metric and the risk/reward ratio (RRR) metric for a trading system. In fact, this trade-off is due to the relationship between stop-loss level and take-profit level with the entry level of a trade. To demonstrate this trade-off, we can examine how adjustments to stop-loss and take-profit levels affect the WR and RRR. We'll explore the different scenarios for which this trade-off applies.

As a reminder:

- Stop-Loss (SL) is the price level at which a losing trade is closed to limit losses.
- Take-Profit (TP) is the price level at which a winning trade is closed to secure profits.

A tighter stop-loss means that the trade will be closed with a small loss if the market moves unfavorably. This increases the potential risk/reward ratio because losses are minimized while the potential reward remains the same. But in terms of Winning Rate a tighter stop-loss increases the likelihood of the trade being stopped out prematurely due to market noise or minor fluctuations, reducing the winning rate. Since the average loss (L) is reduced with a tighter stop-loss, the RRR (R/L) increases.

A more distant take-profit means that the trade will aim for a larger profit before closing. A more distant take-profit level makes it less likely that the trade will reach the take-profit target, reducing the winning rate. Since the average reward (R) increases with a more distant take-profit, the RRR (R/L) increases.

To illustrate the trade-off, let's consider a scenario where the stop-loss and take-profit levels are adjusted to see their impact on WR and RRR.

Scenario 1: Tight Stop-Loss, Close Take-Profit

- We assume SL=1 unit, TP=2 units.
- Let the winning rate be relatively high because the take-profit is easily achievable.

The RRR is calculated as:

$$\text{RRR}_1 = \frac{TP}{SL} = \frac{2}{1} = 2$$

Scenario 2: Tight Stop-Loss, Distant Take-Profit

- We assume SL=1 unit, TP=4 units.
- The winning rate decreases because the take-profit is harder to reach.

The RRR is calculated as:

$$\text{RRR}_2 = \frac{TP}{SL} = \frac{4}{1} = 4$$

Scenario 3: Wide Stop-Loss, Close Take-Profit

- We assume SL=2 units, TP=2 units.
- The winning rate increases because the trade has more room to move before hitting the stop-loss.

The RRR is calculated as:

$$\text{RRR}_3 = \frac{TP}{SL} = \frac{2}{2} = 1$$

Scenario 4: Wide Stop-Loss, Distant Take-Profit

- We assume SL=2 units, TP=4 units.
- The winning rate is lower than Scenario 3 but higher than Scenario 2 because the stop-loss is wider.

The RRR is calculated as:

$$\text{RRR}_4 = \frac{TP}{SL} = \frac{4}{2} = 2$$

From the scenarios above, we see the following patterns:

- Tight Stop-Loss increases the RRR but decreases the WR.
- Wide Stop-Loss increases the WR but decreases the RRR.
- Distant Take-Profit increases the RRR but decreases the WR.
- Close Take-Profit increases the WR but decreases the RRR.

Balancing this trade-off requires careful adjustment of stop-loss and take-profit levels to align with the trader's goals and risk tolerance. If the goal is to maximize the RRR, the strategy will likely involve tight stop-losses and distant take-profits, accepting a lower winning rate. Conversely, if the goal is to maximize the winning rate, the strategy will likely involve wider stop-losses and closer take-profits, accepting a lower risk/reward ratio.

In conclusion the mathematical demonstration of the trade-off between the risk/reward ratio and the winning rate shows that the adjustments to stop-loss and take-profit levels directly influence both metrics. A higher RRR is achievable with tighter stop-losses and more distant take-profits, but at the cost of a lower winning rate. Conversely, increasing the winning rate involves wider stop-losses and closer take-profits, which reduces the RRR. Understanding and managing this trade-off is crucial for developing a robust and effective trading system.

We will demonstrate in our data analysis and through a simulation varying the take-profit placement in section 5.4.1that it is indeed possible to find an optimal combination of the trade-off between the Risk/Reward ratio and the winning rate to maximize the expected gain of a trading system.

Tradeoff between Winning rate & number of occurrences metrics

Incorporating additional signals into a trading strategy, aiming for signal confluence, can enhance the winning rate of trades. This improvement in winning rate results from the strategy becoming more selective, requiring the validation of multiple conditions before initiating a market position. However, this increased selectivity also leads to a reduction in the total number of trading opportunities available. In simpler terms, the more conditions a strategy imposes for entering a trade, the fewer market setups it will detect.

The rise in complexity and the integration of additional signals into a trading system often lead to a system that, although more accurate, offers fewer opportunities. This reduction in opportunities is the downside of the improved win rate: while the executed trades are more likely to succeed, the total number of trades carried out decreases. Therefore, complicating a trading system is not necessarily beneficial, especially in the context of algorithmic trading where the ability to capture a large number of opportunities is crucial for maximizing gains.

In algorithmic trading, a simple and effective system is often preferable because it allows for the execution of many trades across various market setups. The capability of algorithms to process and execute trades rapidly means that capturing numerous small opportunities can lead to robust overall performance. Conversely, a discretionary trader, who cannot manually execute an unlimited number of trades, would benefit more from a more complex strategy. For a discretionary trader, continuously improving the win rate through strategy complexity can help filter trades, thus reducing the number of setups to manage manually and increasing the quality of the executed trades.

In the end, although adding extra signals can improve the winning rate, it is crucial to consider the type of trading practiced. For trading algorithms, a less complex and more opportunistic system may prove more effective. For discretionary traders, however, a more complex strategy focused on the quality of trades may be more suitable and manageable. The key lies in balancing system complexity and the ability to efficiently handle opportunities according to the specific resources and constraints of each type of trading.

To mathematically demonstrate the trade-off between the winning rate (WR) and the number of opportunities (N) in a trading strategy, we can examine how the addition of multiple signals affects these metrics. The key concept is that adding more conditions (signals) increases the selectivity of the strategy, improving the winning rate but reducing the number of opportunities.

The Winning Rate (WR) is the proportion of trades that are profitable:

$$\text{WR} = \frac{\text{Number of winning trades}}{\text{Total number of trades}} \tag{5}$$

The Number of Opportunities (N) is the total number of trades generated by the strategy:

$$N = \text{Total number of trades} \tag{6}$$

The Probability of Signal Detection (P) is the probability that a single signal validates an opportunity:

$$P = \text{Probability of single signal detection} \tag{7}$$

The Probability of Confluence of Signals (P_C) is the probability that an opportunity is validated by multiple independent signals:

$$P_C = P_1 \times P_2 \times \cdots \times P_k \tag{8}$$

We consider two strategies:

- **Simple Strategy (S)**: Uses one signal with a probability of detection P.
- **Complex Strategy (C)**: Uses k independent signals with each having a probability of detection $P_i \approx P$.

For the Simple Strategy:

$$N_S \propto P \tag{9}$$

$$WR_S = \text{Winning rate of the simple strategy} \tag{10}$$

For the Complex Strategy, with k independent signals:

$$P_C = P^k \tag{11}$$

$$N_C \propto P^k \tag{12}$$

$$WR_C = f(WR_S, k) \tag{13}$$

where f is an increasing function representing the improved winning rate due to the confluence of multiple signals.

We assume $P = 0.5$ for the probability of detection for each signal.

For $k = 1$ (Simple Strategy):

$$P_C = 0.5 \tag{14}$$

$$N_S \propto 0.5 \tag{15}$$

For $k = 2$ (Two signals):

$$P_C = 0.5 \times 0.5 = 0.25 \tag{16}$$

$$N_C \propto 0.25 \tag{17}$$

For $k = 3$ (Three signals):

$$P_C = 0.5 \times 0.5 \times 0.5 = 0.125 \tag{18}$$

$$N_C \propto 0.125 \tag{19}$$

In general, for k signals:

$$P_C = P^k \tag{20}$$

$$N_C \propto P^k \tag{21}$$

As k increases, N_C decreases exponentially:

$$N_C = N_S \times P^{(k-1)} \tag{22}$$

The winning rate WR_C for the complex strategy improves as k increases due to the increased selectivity of trades. Let $f(WR_S, k)$ denote this relationship.

We assume $WR_S = 0.4$ (Simple Strategy Winning Rate).

For $k = 1$:

$$WR_C = WR_S = 0.4 \tag{23}$$

For $k = 2$:

$$WR_C = f(0.4, 2) \approx 0.6 \tag{24}$$

For $k = 3$:

$$WR_C = f(0.4, 3) \approx 0.7 \tag{25}$$

In general, as k increases, WR_C asymptotically approaches 1 (100%), but in practical terms, the increase is incremental and may depend on the nature of the signals used.

Mathematically, the trade-off between the winning rate and the number of opportunities in a trading strategy can be expressed as follows:

$$\text{Number of Opportunities: } N_C \propto P^k \tag{26}$$

$$\text{Winning Rate: } WR_C = f(WR_S, k) \tag{27}$$

As the number of signals k increases, the number of opportunities N_C decreases exponentially, while the winning rate WR_C increases due to higher selectivity. This trade-off highlights the importance of balancing complexity and opportunity in the design of trading strategies.

In conclusion, we have identified two critical trade-offs in designing a trading system. Firstly, we noted that adjusting the take-profit and stop-loss levels involves a trade-off between the winning rate and the risk/reward ratio. Secondly, we observed that adding filters can improve the winning rate but will reduce the total number of available opportunities in the market that meet our pattern configuration. Ultimately, it is essential to consider both of these trade-offs when constructing a trading system to achieve a balance that maximizes the expected gain while maintaining an adequate number of trading opportunities.

Since the winning rate partly determines the expected gain (in addition to the risk/reward ratio), we can extend the concept by saying that there is a trade-off between the total number of opportunities available in the market and the expected gain per trade. If we delve deeper, considering that the number of opportunities determines the overall profit expectancy of the trading system, we can conclude that there is a trade-off between the profit expectancy per trade and the overall profit expectancy of the system. Therefore, it is crucial to consider the winning rate, the risk/reward ratio, and the number of opportunities available for a trading system to find the optimal combination of these three metrics to maximize the overall expectation of the trading system.

The ultimate goal is to find an optimal combination of winning rate and risk/reward ratio to maximize the expected gain (E) while maintaining a manageable number of opportunities (N). It is essential to recognize that the number of opportunities is a critical variable. Although it does not directly determine the expected gain per trade, it significantly influences the overall expected gain of the system. The more opportunities available, the more profitable trades can be executed, thereby enhancing the system's overall profitability. This delicate balance ensures that the trading system is not only profitable on a per-trade basis but also scalable and robust over a large number of trades, ultimately leading to consistent and sustainable returns.

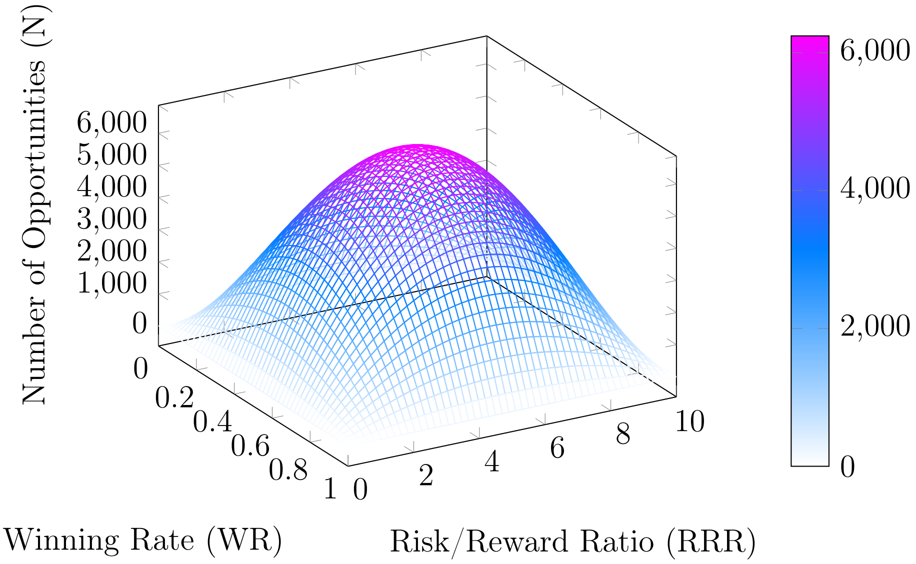


Figure 3: Trade-off between Winning Rate, Risk/Reward Ratio, and Number of Opportunities

This 3D plot is not realistic, it is intended to show that there is an optimal combination of number of opportunities, winning rate and risk/reward ratio to maximise the total expected gain from a trading system.

In section 5.4.1, we will explore finding an optimal balance between the risk/reward ratio and the winning rate to maximize the expected gain per trade. However, incorporating a filter to enhance the winning rate can paradoxically diminish the system's overall expected gain. This reduction stems from a decrease in the number of available trading opportunities, a loss not always sufficiently offset by the improved winning rate. The challenge in algorithmic trading is to implement a filter that significantly boosts the winning rate without disproportionately curtailing the number of trade opportunities, thereby genuinely enhancing the system's overall expected gain.

4.2.2 Development of the trading system

In this section, we will detail the development of our trading system, which we will subsequently backtest within this thesis. We will follow the steps outlined in our methodology in section 4.1.3, starting from establishing a hypothesis to constructing a fully operational trading system.

1. Formulate a Fundamental Hypothesis:

In the study of financial markets, it is observed that when an asset is trending upwards, it typically forms a pattern of higher highs and higher lows. These higher lows signify the end of short-term corrections within the broader upward trend. Understanding and identifying these correction lows can be crucial for traders aiming to capitalize on the continuation of the trend.

Our hypothesis posits that these correction lows, or pullbacks, share certain common characteristics that can be systematically identified and exploited. By recognizing these traits, we can time our entries more effectively, positioning ourselves at the lows of these corrections to maximize our participation in the ensuing trend continuation. This strategy could offer significant risk/reward ratios.

2. Identify the Hypothesis as a Pattern in the Markets:

We will now attempt to identify the common characteristics of corrections within an uptrend. It is observed that when an uptrend is well-defined through price action analysis, there is a specific zone on the RSI that corresponds to most of the correction lows. Our analysis shows that, indeed, during a defined uptrend, the RSI often enters a particular range at the correction lows.

We will now develop the main concept of our pattern, which is the Flexible Neutral RSI Zone. This zone varies depending on the trend, the asset and the time-frame, hence the term "flexible." We identify a neutrality zone on the RSI that serves as the correction low point in the price action during an uptrend.

The Flexible Neutral RSI Zone is detected as a range on the RSI that corresponds to the correction lows in an uptrend. This zone is flexible because it is defined for each market trend individually. When a trend concludes, the associated neutrality zone becomes obsolete. A new neutrality zone on the RSI will only emerge once a new trend is established.

This concept allows us to dynamically adjust our trading strategy based on the prevailing market conditions. By attributing a specific neutrality zone to each market trend, we can more accurately identify potential correction lows, thus optimizing our entry points and enhancing the overall effectiveness of our trading system.

The Flexible Neutral RSI Zone provides a robust framework for analyzing and responding to market trends, ensuring that our strategy remains adaptive and relevant across different assets and market conditions.

3. & 4. Centralize and analyze in Detail How the Pattern is Constructed:

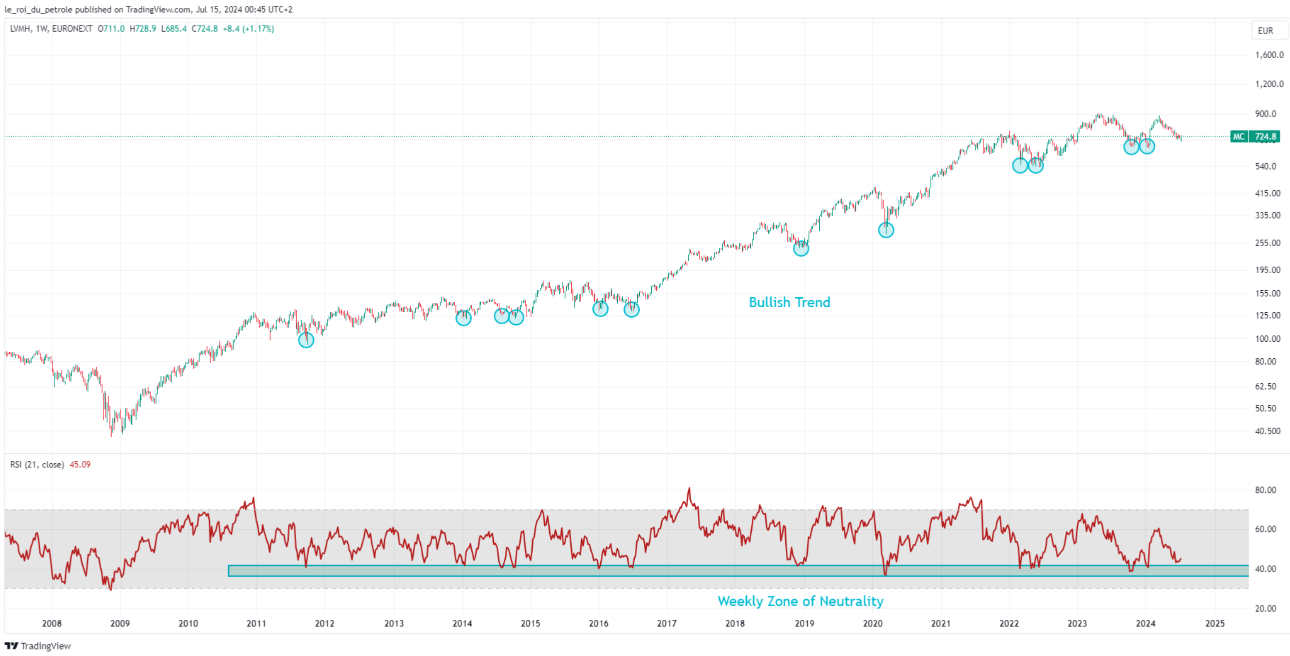


Figure 4: Weekly time-frame pattern example on LVMH stock

In this example, examining the weekly time frame for LVMH from 2010 to 2024, we can see that the Neutral Zone successfully captures 12 correction lows. This allows for purchasing at the correction lows within the major upward trend characterizing the price movement over this period.

The strength of this pattern lies in its ability to offer substantial risk/reward ratios. This is achieved through its mean-reversion characteristic, which aims to capture the correction lows, and its trend-following aspect, which seeks to ride the upward waves of the major trend. Consequently, the pattern allows for a relatively tight stop-loss due to precise timing, while providing significant upside potential by following the trend.

Figure 5: Daily time-frame pattern example on Netflix stock

In this new example, illustrated by Netflix's daily time frame, we observe an upward trend that starts in July 2022 and continues into 2024. This uptrend is marked by several corrections, and we notice that the lows of these corrections correspond to lows on the RSI indicator within the same zone. This zone, therefore, represents our Neutral Zone for this trend on this time frame for this asset.

According to our strategy, this zone will serve as a relevant buying area to continue riding the upward trend until proven otherwise. As long as this zone is not broken on the RSI (allowing for a margin of error), we will consider it an excellent buying opportunity.

The advantage of this pattern is its versatility; it can be applied to any asset class as long as a trend is present (stocks, crypto, commodities, FOREX, etc.). Markets are generally composed of fractals, consisting of impulses and corrections across various time frames, whether on a large scale (monthly) or a very small scale (m5). This inherent fractal nature ensures a vast number of opportunities across all types of markets and time frames. Ultimately, this versatility and scalability make the pattern a valuable tool to capitalize on trends in diverse market environments.

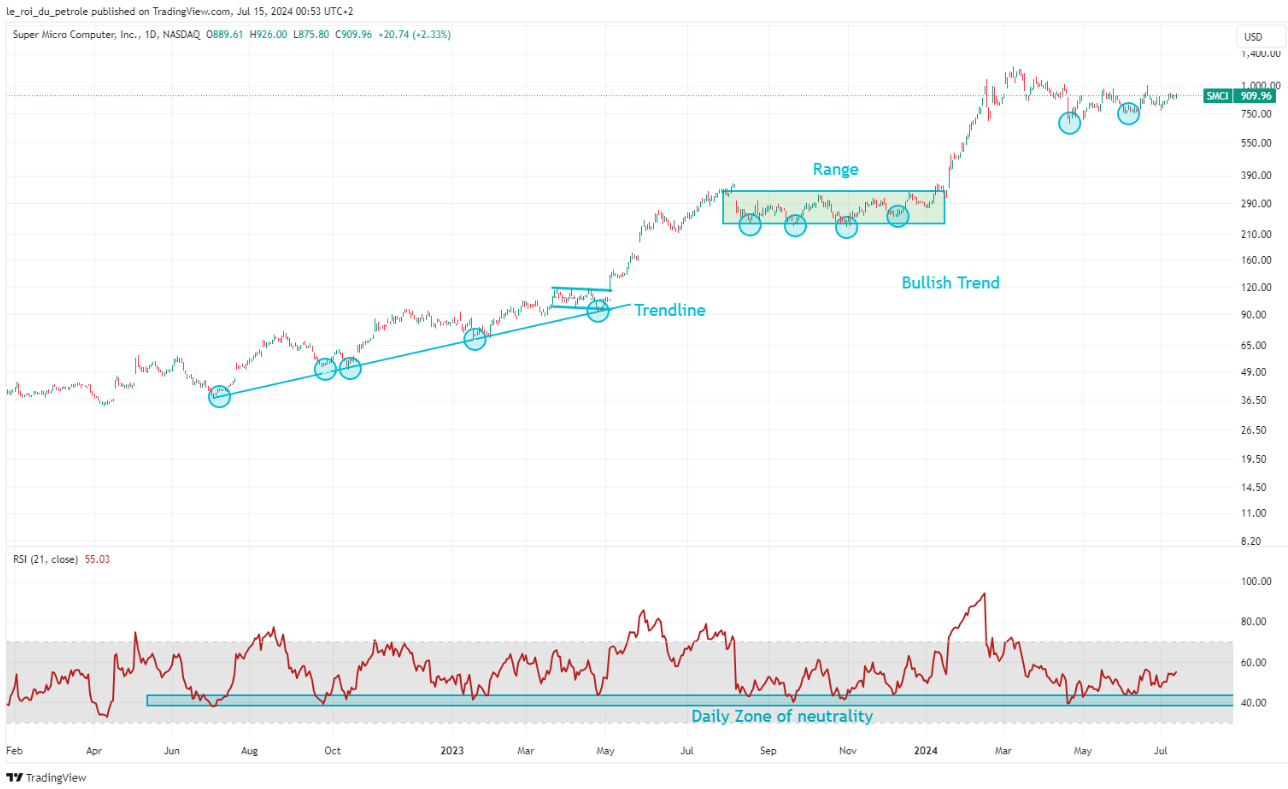


Figure 6: Daily time-frame pattern example on SMCI stock

In this example, it becomes evident that incorporating additional signals into our pattern could potentially improve its success rate. By analyzing multiple iterations of our pattern based on the Flexible Neutral RSI Zone, we observe that the pattern tends to perform better, with higher success rates and improved risk/reward ratios, in low-volatility price action environments. Specifically, the pattern is more effective when the correction takes the form of a range rather than a sharp, spiky correction.

This observation brings us back to the trade-off between win rate and the number of available market opportunities, as discussed in Tradeoff section By adding an entry condition related to such a price action signal to our pattern, we could achieve a confluence of positive signals, thereby enhancing the probability of success. This approach might be suitable for discretionary trading. However, our research indicates that incorporating such a signal drastically reduces the number of trading opportunities.

By adding the price action signal to the system, we would exclude a number of patterns from our trading strategy. This would result in missing out on potentially profitable patterns that could have compensated for the losses from unsuccessful ones. When comparing the overall profitability between a simpler trading system and a more complex one with the added price action signal, it becomes clear that maintaining a simpler system is preferable. The added complexity would lead to a significant loss of successful patterns that would have offset the failures avoided by the more complex system.

In conclusion, while adding signals to our pattern could theoretically improve the success rate, the reduction in trading opportunities would negatively impact the overall profitability. Therefore, it is more beneficial to retain the simpler trading system, as it captures a higher number of profitable patterns, ultimately leading to greater overall returns.

5. Systematize and Objectify the Pattern:

We will now establish the entry and exit conditions of our trading system with a focus on systematic implementation, as discussed in section 4.3.1

Entry Conditions:

- An uptrend observed on the trading time frame

  AND

- A localized RSI neutrality zone with a maximum width of 10 and at least 2 local bottoms on the RSI

Exit Conditions:

- Break of the neutrality zone (with margin of error) (stop-loss based on RSI)

  OR

- Stop-loss based on price action and ATR

Conclusion

In conclusion, this section has addressed the crucial considerations involved in constructing a trading system. We developed the concept of trade-offs between the win rate and the risk/reward ratio, as well as the trade-off between the win rate and the number of available market opportunities. Following our methodology, we have successfully built and systematized our trading system. This will enable us, in the next section, to code the system and conduct backtests on real cryptocurrency price data.

4.3 Algorithm structure

In this section, we explore the rationale behind our initial selection of R as the programming language for our trading strategy project, despite the existence of other robust alternatives like Python and MQL5 and how this approach enhanced our understanding and control over the process. The subsequent parts detail our code structure, the optimization and impact of our parameters, and the incorporation of machine learning methods to refine and validate our trading strategies. This comprehensive approach ensures a robust framework for backtesting and evaluating the performance of our trading strategy developed during this thesis.

4.3.1 Programming language

The initial choice of programming language for our project was R. This decision was driven by our familiarity and extensive experience with R, making it a natural starting point. While exploring various trading strategies and conducting research, we encountered numerous other coding languages and tools that are arguably better suited for trading, such as Python, MQL5 or

Pinescript. Python, in particular, boasts a large financial community, trading-oriented packages, and easier access to data. Despite these alternatives, R has its own set of advantages:

- Statistical Power: R is renowned for its strong statistical capabilities, which are crucial for analyzing financial data and implementing sophisticated trading algorithms.
- Rich Ecosystem: R offers a rich ecosystem of packages, such as quantmod, TTR, and dplyr, which provide robust tools for financial analysis and data manipulation.
- Data Visualization: R excels in data visualization with packages like ggplot2, enabling clear and comprehensive graphical representations of trading data and results.
- Ease of Use: For those already proficient in R, it offers a high level of ease and efficiency, allowing for rapid development and iteration of trading strategies.

One of the greatest challenges in our coding journey was translating subjective technical strategies into objective, executable code. This required a deep understanding of both the trading strategies and the R programming language. By using R, we had to manually incorporate all the steps of our process, ensuring there were no shadow zones in the work presented. This meticulous approach, although time-consuming, enhanced our understanding and allowed us to fully control each step of the strategy implementation.

It is clear that other tools, particularly Python, offer more automation and convenience, potentially saving significant amounts of time. These tools often have built-in functions for trading operations, seamless integration with various data sources, and broader community support for financial applications. However, by using R, we ensured a transparent and thorough development process, with each step explicitly defined and executed, providing a solid foundation for the strategies developed in this thesis.

4.3.2 Code Structure

The code structure of our project is designed to support the backtesting and evaluation of various trading strategies. This section outlines the key functions and their purposes.

The primary objective of this code is to backtest trading strategies by computing the indicators needed for the strategy, simulating the trading following entry and exit signals and returning output for analysis. The outputs are further describe. In a second time and for optimization purpose, another function iterate through diverse setups. The results are subsequently transferred to an Excel workbook for detailed analysis between setups.

Sequential diagram

The sequence diagram depicted in Figure 7 outlines the step-by-step process flow of the trading strategy simulation. This diagram provides a clear visualization of the interactions between different components, demonstrating how the strategy is executed from start to finish.

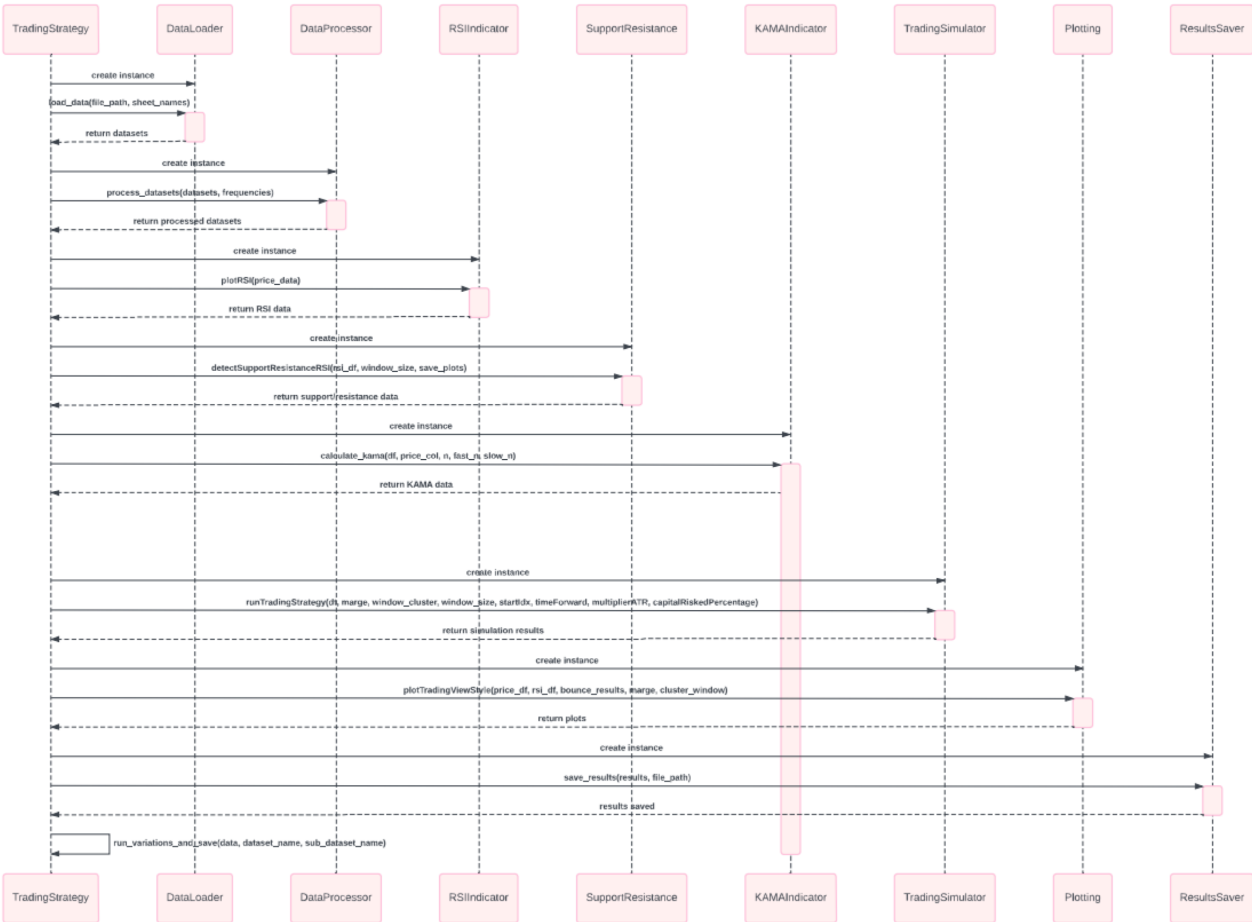


Figure 7: Sequential diagram of the code

Outputs

Performance Metrics

The function calculates various performance metrics to evaluate the strategy performance. The key performance metrics used in this analysis include Sharpe ratio (4), drawdown (1), and annualized return (2) as the most important for optimization. Other metrics include winning rate (3), number of trades (5), generated alpha (6), beta (7), market exposure time (8), profit factor (9), maximum adverse excursion (MAE) (10), maximum favorable excursion (MFE) (10), expectancy (11), and recovery factor (12).

Trade Book

The trade book is a detailed record of all trades executed during the simulation. It includes information such as entry and exit dates, prices, ATR-based stop-loss levels, amounts traded, transaction costs, profits, and returns. This trade book allows for in-depth analysis of each trade's performance and the overall strategy.

Optimization

A key aspect of our code structure is the optimization of strategy parameters. We implemented a systematic approach to vary the parameters of indicators, resulting in different variations per frequency.

Our database containing 10 different assets, optimizing with limited data presents a significant challenge, primarily due to the risk of overfitting. Overfitting occurs when a model is too finely tuned to the specific characteristics of the training data, resulting in poor generalization to new, unseen data.

To mitigate this risk, we carefully selected a limited number of parameter variations, balancing thorough optimization with the potential for overfitting. This approach was guided by the principle of parsimony, which advocates for simplicity in model design. By avoiding excessive complexity, we aimed to enhance the robustness and generalizability of our strategies. Additionally, the limited number of variations helped manage computational resources effectively, ensuring that our optimization process was feasible given our constraints. The optimization process focused on the following performance metrics, listed in order of importance:

- **Maximize the Sharpe ratio**
- **Minimize the drawdown**
- **Maximize the return**

To ensure robustness and avoid overfitting, we averaged the performance metrics over 10 different assets for each parameter setup. This averaging process helps to generalize the strategy's effectiveness across various market conditions and asset behaviors. These metrics were optimized over two different frequency units: 15 minutes and 1 hour, each with its specific parameters.

Parameters

Margin The main indicator of this trading strategy is the Relative Strength Index (RSI), specifically focusing on detecting support levels where the RSI is expected to bounce. The Margin (Marge) defines the allowable error around these support levels, effectively representing the thickness of the support zone. It influences two key aspects of the entry and exit conditions:

- **Entry Condition:** A cross down below a support level is a trigger for entering a trade. A larger margin means that more observations will be considered as crossing down below the support, potentially leading to more frequent trade entries.
- **Exit Condition:** A break below the support level is a trigger for exiting a trade. A larger margin makes the strategy more lenient, allowing more room for the RSI to reverse before exiting, which can result in larger potential losses if the reversal does not occur.

Window Size To identify significant support levels, we need to define a window over which we will search for these supports. The window size determines the number of periods (data points) considered in the calculation.

Fraction/Clusters The `detectSupportResistanceRSI` function identifies significant support and resistance levels within a given window of RSI data through a process that involves clustering. First, it calculates rolling maximum and minimum RSI values over a specified period, referred to as the window size. These rolling values are combined and filtered to ensure distinct levels, which are then subjected to hierarchical clustering to group similar RSI values. The clustering process helps in identifying the most significant support and resistance levels by averaging the values within each cluster.

The fraction parameter, referred to as `window_cluster`, plays a crucial role in this process. It defines the size of the sub-period, expressed as a fraction of the total window size, over which the rolling calculations are performed. For example, if the window size is 100 periods, setting `window_cluster` to 0.25 means that rolling maximum and minimum values are computed over 25-period sub-windows within the 100-period window.

Multiplier ATR (multiplierATR) This parameter is used to calculate the stop-loss level for trades. The Average True Range (ATR) is a measure of volatility, and the stop-loss is set based on a multiple of the ATR. This helps in setting a dynamic stop-loss that adjusts according to market volatility.

$$\text{ATRprice} = \text{entryPrice} - (\text{multiplierATR} \times \text{ATR})$$

4.3.3 Machine learning Methods

Initially, we planned to employ three Machine Learning (ML) methods: feature engineering, cross-validation, and Monte Carlo simulation. However, our final implementation evolved based on the specific needs and limitations of our project.

- **Feature Engineering:** Feature engineering was initially intended to create and test various features to enhance our strategy's predictive power. In practice, we tailored feature engineering to align with our specific strategies, selecting features based on domain knowledge rather than through trial and error. This approach ensured that our features were relevant and impactful, streamlining the development process. Additionally, we adhered to the principle of parsimony, focusing on simplicity and relevance to prevent overfitting given our limited computational resources.

- **Cross-Validation:** Our dataset extends from November 2019 up to the end of February 2024. We structured the data by considering the first half as the training set and the second half as the test set, effectively mimicking traditional cross-validation with a distinct training and test set. Limiting the parameter variations to 36 per asset and time unit further mitigated the risk of overfitting, ensuring a balanced approach to model validation.

- **Monte Carlo Simulation:** Initially, we considered using Monte Carlo simulations to generate synthetic data and test our strategies under various scenarios. However, we realized that creating data with realistic behavioral factors was challenging and could lead to misleading results. Given that our strategies rely heavily on behavioral patterns, we decided against using Monte Carlo simulations for price data to avoid potential inaccuracies. Instead, we applied Monte Carlo simulations to non-correlated data, such as equity curves and risk of ruin distribution functions, where behavioral patterns are less critical.

Conclusion

Our journey through these ML methods underscores the importance of adapting techniques to the project's specific needs and limitations. The behavioral aspect of our strategies is a fundamental component of technical analysis and could not be overlooked. By focusing on tailored feature engineering, optimizing strategies, and carefully managing the trade-off between optimization and overfitting, we ensured that our backtesting framework was robust despite limitations and aligned with our goals. The code structure reflects this adaptability, providing a clear and effective approach to strategy development and analysis. The detailed list of functions used, the code, inputs, and outputs are fully described in the appendix section: A.2 Code Structure.

5 Analysis of results and optimization

5.1 Introduction

In this section, we will present the results of our backtest for the trading system. We will then conduct statistical analyses to propose improvements for an enhanced and optimized version of the system. Our primary challenge throughout this process will be to avoid overfitting or "cherry-picking" and to consider the concept of trade-offs between the number of occurrences and the win rate, as we have previously developed. By carefully balancing these elements, we aim to refine our system to achieve robust and reliable performance in various market conditions.

5.2 Train dataset & correlation

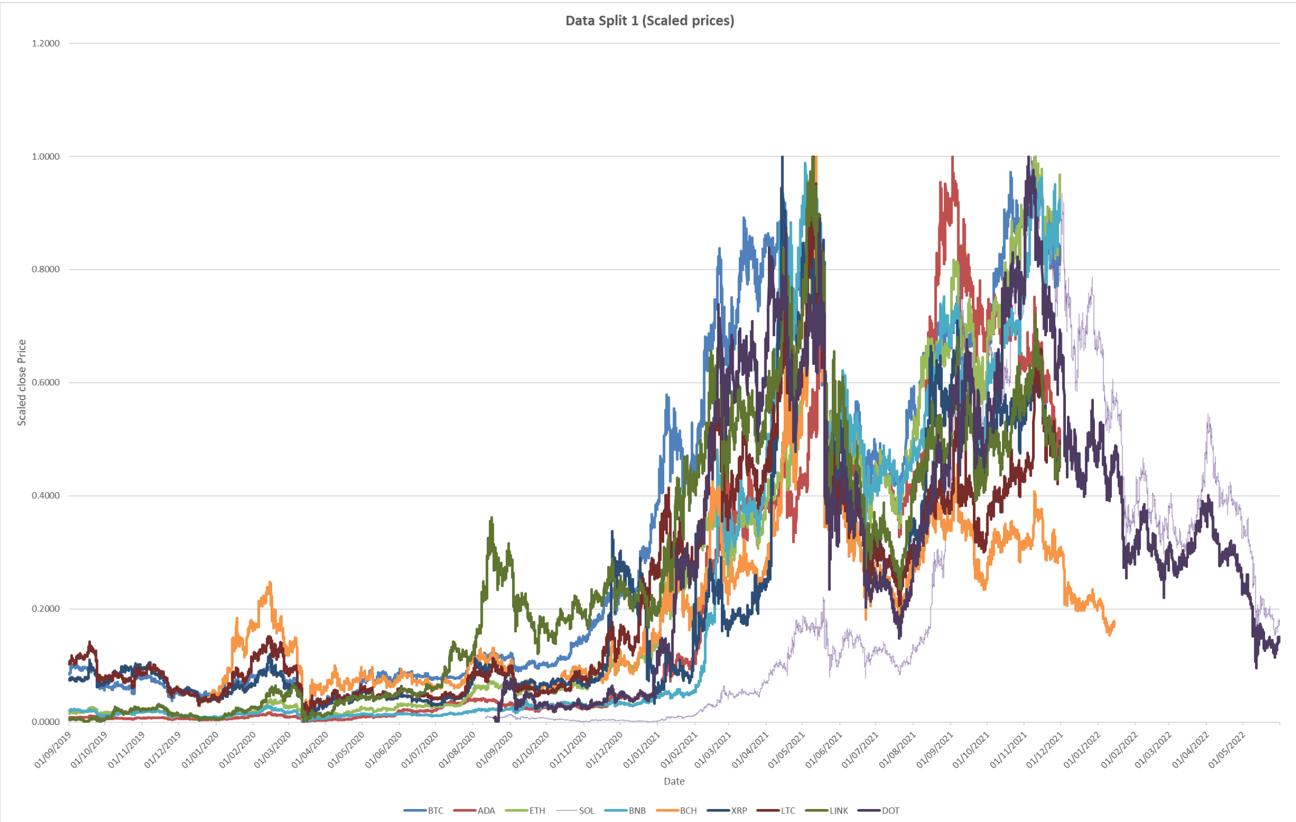


Figure 8: Data Split 1 (Scaled closed prices)

The price curves do not all stop on the same date, which is expected, as we have simply divided the price data we had access to for each asset into two parts. These data sets do not necessarily have the same historical range. The first half is represented in this graph, while the second half will be used for the backtest of Version 2 of our trading system after analyzing the results and implementing improvements.

It is clear that the data is valuable as it encompasses various market conditions. We observe strong bullish trends as well as equally strong bearish trends. This diversity will be interesting to analyze to see how our trading system reacts to these different conditions, thereby evaluating its robustness across market changes and understanding the conditions that significantly impact the system's performance.

Similarly, we can see that volatility is relatively low in 2019 and 2020 compared to later periods. This will also allow us to examine how the system's results vary under more or less volatile market conditions. It should be noted that the curve is arithmetic, so 2019 and 2020 appear relatively flat, although there was still some volatility during these years.

Correlation	BTC	ADA	ETH	SOL	BNB	BCH	XRP	LTC	LINK	DOT
BTC	1.00	0.78	0.85	0.62	0.84	0.76	0.74	0.90	0.87	0.95
ADA		1.00	0.92	0.75	0.88	0.68	0.83	0.69	0.71	0.78
ETH			1.00	0.85	0.94	0.74	0.85	0.77	0.77	0.85
SOL				1.00	0.72	0.36	0.63	0.44	0.45	0.68
BNB					1.00	0.82	0.92	0.84	0.83	0.86
BCH						1.00	0.84	0.81	0.93	0.75
XRP							1.00	0.81	0.81	0.76
LTC								1.00	0.95	0.88
LINK									1.00	0.90
DOT										1.00

Table 1: Correlation matrix of cryptocurrency closing prices.

It is important to note that the different assets we are testing are highly correlated, as shown in the correlation matrix. This high correlation reduces the robustness of the backtest due to the strong correlations, but it reflects the reality of the cryptocurrency market, which is heavily correlated with the main cryptocurrency, Bitcoin. Opening a trade on a cryptocurrency other than Bitcoin (Altcoins) also means being exposed to Bitcoin's fluctuations to a certain degree. Thus, in the reality of crypto trading, it is essential to consider this strong market correlation to Bitcoin, which tends to guide the market.

Having price variations on other cryptocurrencies besides Bitcoin also allows us to analyze the system's results while minimizing the risk of overfitting. Similar to a Monte Carlo simulation of multiple possible price paths, we can observe how the system's results vary with different price path variants represented by the prices of our 10 cryptocurrencies. Unlike Monte Carlo simulations, which lack the capacity to simulate the underlying human psychology that influences price movement dynamics, using real data ensures greater realism.

5.3 Backtest results

In this section[8] , we will discuss the results of our trading system. First, we will compare the performance of our trading system to a benchmark, which corresponds to a simple "Buy and Hold" strategy of the asset. Next, we will analyze the distribution of returns based on the frequency of trades executed by our trading system.

[8]Additional results such as histograms and optional tables are available in the Appendix A.4

5.3.1 Strategy VS Benchmark

In this subsection, we have created a graph comparing the Sharpe ratio of our trading system against the benchmark. The Sharpe ratio considers the average performance of the 10 assets for a comprehensive comparison. Additionally, we have plotted a graph detailing the drawdowns of the benchmark and another graph showing the returns, both compared to our trading system, with a detailed breakdown for each asset.

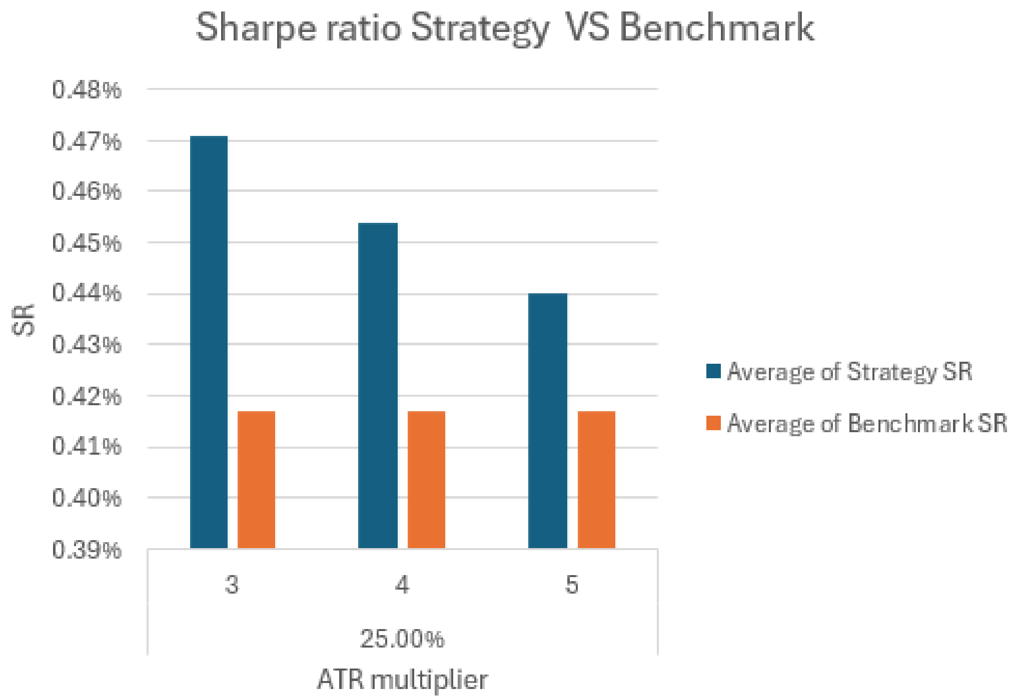


Figure 9: Sharpe ratio Strategy V1 vs Benchmark

We can observe that when comparing the Sharpe ratio of our trading system to the benchmark for the three possible stop-loss configurations we implemented, the average Sharpe ratio is better in all three cases. Unsurprisingly, the Sharpe ratio of our system tends to be higher when the multiplier in front of the ATR decreases towards 3, demonstrating that a tighter stop-loss in our system has a positive impact.

In more detail, if the Sharpe ratio tends to increase as the stop-loss becomes tighter, there are two possible reasons.
The first reason is that the average expected gain increases, thereby enhancing the Risk-Reward ratio more than the reduction in the Win rate. This implies that while the Win rate decreases, it is more than compensated for by an improved Risk-Reward ratio, resulting in better risk-adjusted returns. As a reminder, we demonstrated this trade-off concept between the Win rate and the Risk-Reward ratio in Tradeoff section.
The second possible reason is that the volatility of the returns decreases proportionally more than the average return when the stop-loss becomes tighter. This reduction in volatility results in a higher Sharpe ratio.

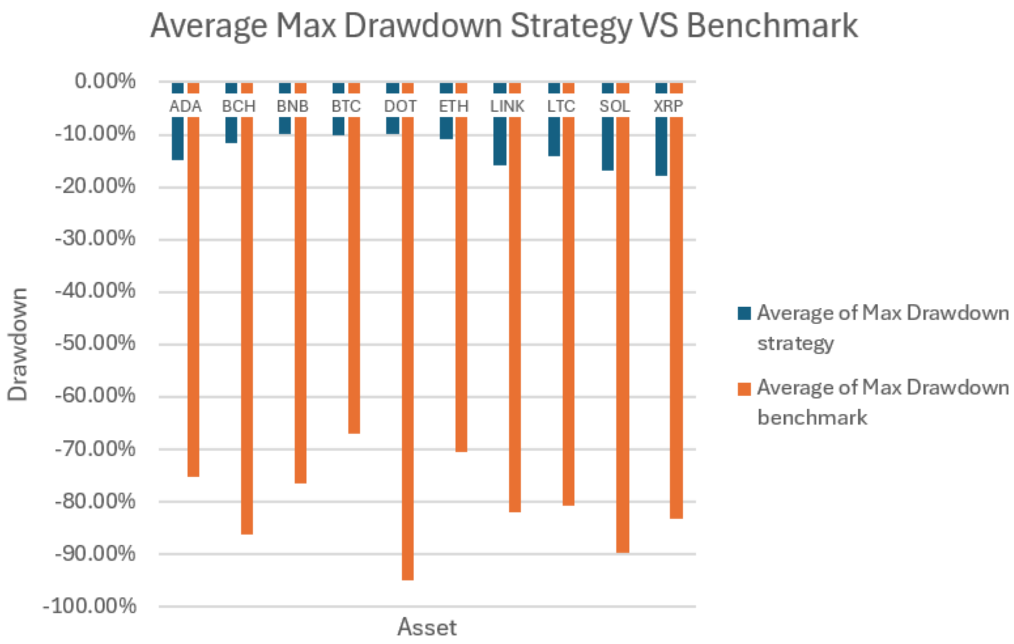


Figure 10: Max Drawdown Strategy V1 vs Benchmark by asset

Similarly, when we observe the drawdowns of our trading system compared to the traditional "Buy and Hold" strategy, we can see that our trading system consistently and significantly exhibits a much lower drawdown for each of the 10 assets. Thus, the drawdown of our trading system ranges from -9% to -17% depending on the assets, while the benchmark experiences substantial drawdowns ranging from -67% to -94%.

Thus, we can acknowledge that our trading system has the advantage of limiting losses, thereby avoiding difficult drawdown periods in a highly volatile cryptocurrency market. This is an interesting feature of our system, especially when we consider that this market is characterized by significant upward and downward movements with substantial drawdowns.

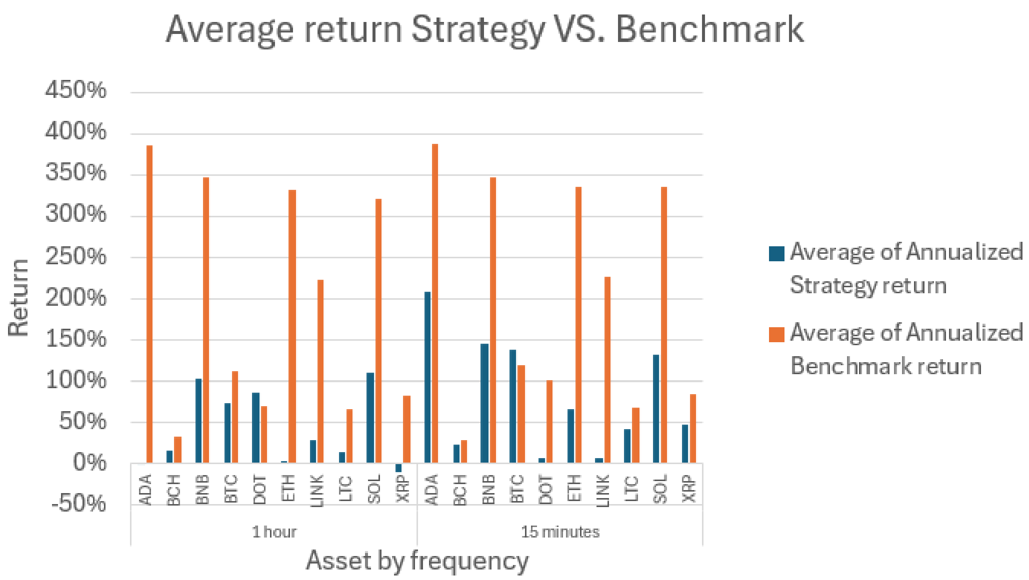


Figure 11: Average anuualized return Strategy V1 vs Benchmark by asset

However, when we observe the returns of our trading system by asset compared to those of the "Buy and Hold" strategy, we can see that the system does not outperform the benchmark. This is not a significant issue considering that our system's Sharpe ratio is better. With a higher Sharpe ratio, it is possible to apply leverage to the system, which would yield a higher return for the same level of risk as the benchmark.

Therefore, knowing this, our priority for version 2 of the system is to improve the Sharpe ratio, as we believe it is a more comprehensive measure of the system's actual performance. Furthermore, it is important to understand that our system is not invested 100% of the time in the market, unlike the "Buy and Hold" strategy. This is advantageous since, when the system is not invested, the capital can be allocated to other opportunities to maximize capital growth. This variable is not taken into account when calculating returns on capital.

In conclusion, we can observe that our trading system outperforms the benchmark in terms of drawdowns, resulting in a better Sharpe ratio. This initial result challenges the Efficient Market Hypothesis, which posits that it is impossible to consistently outperform the market. It also provides a preliminary indication of the relevance and effectiveness of technical analysis as a tool for creating a profitable trading system. In section 5.4, we will analyse the results in greater depth in order to refine our edge on the markets and present an improved version of the trading system in section 5.5.

5.3.2 Distribution of returns

In this other subsection, we will analyze in detail the distribution of returns of our trading system based on the frequency of trades executed in our backtest. This will provide us with a more comprehensive understanding of the system and its performance.

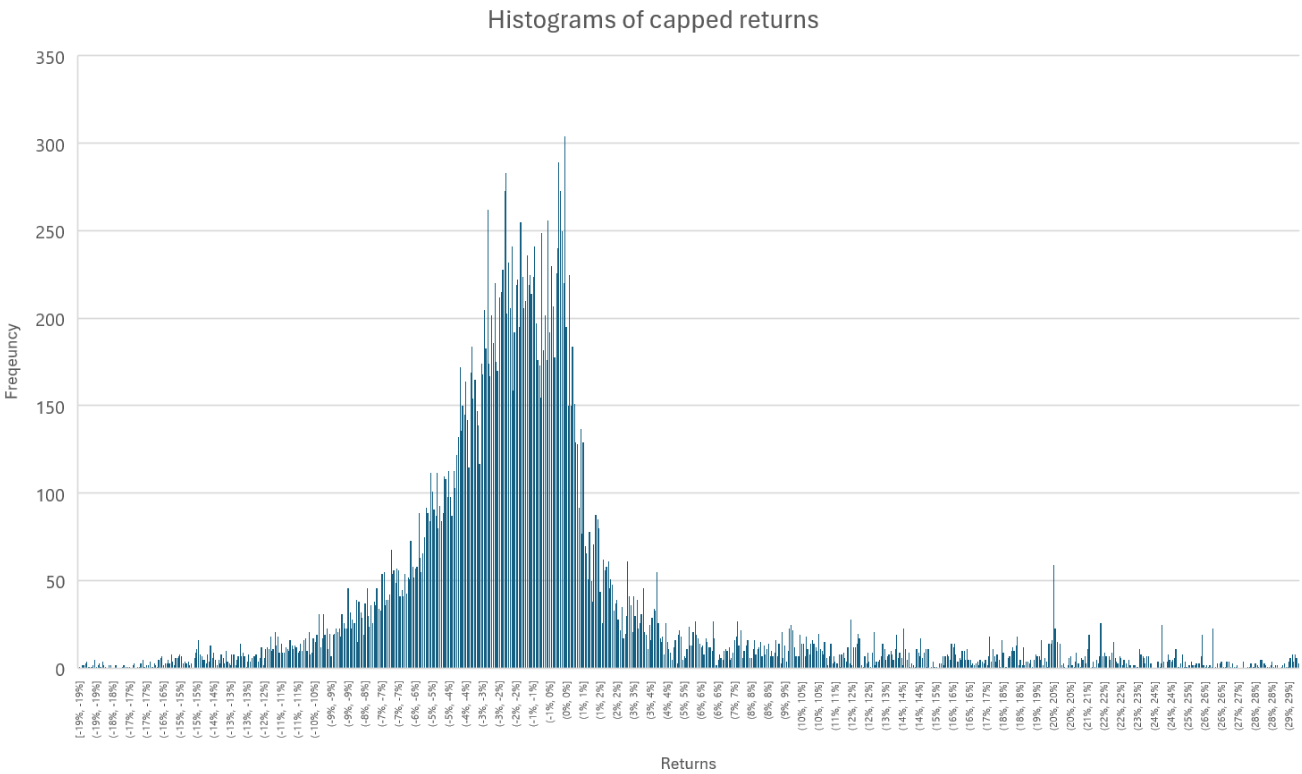


Figure 12: Return distributions

Here, we can see the detailed returns generated by the trading system based on trade frequency. We observe a normal distribution with an average return per trade of 8.33%, though the returns exhibit significant volatility, represented by a high skewness of 15.26.

The total number of transactions is 28,590, which includes various iterations since we simulated the system with different margin levels, stop-loss multipliers (ATR), fractions (clusters), and window sizes.

We note a non-negligible trade frequency with significantly higher returns, aligning with our trading system designed to capture future major bullish impulses in a trend-following approach. This explains why a few trades capture substantial movements, generating most of the system's performance. Specifically, 1.65% of our trades generate 84% of the performance. The system exhibits a relatively low win rate but aims for extraordinary risk-reward ratios by capturing significant price movements. Hence, it is logical that a few trades produce the majority of the system's returns. The strong correlation of 74.6% between the returns generated and the time in position, indicating the system's intent to capture strong trending movements that persist over time.

Capping the histogram by removing the top 1.65% of trades that generate the best performance drastically reduces the average return from 8.33% to 1.36%. Given this context, we question whether it is necessary to add a take-profit in Version 2 of the system or if it might degrade results by limiting the size of the significant movements we aim to capture with this system. We will explore this question in greater depth when we analyse the results in detail below.

Welch's t-test for Difference in Means of capped and uncapped returns

We performed a hypothesis test to verify the difference between the means of the uncapped and capped returns of the strategy. The uncapped group includes all the returns, while the capped group includes returns capped under 100%.

	Uncapped	**Capped**
Mean	8.33%	1.36%
Variance	0.65	0.02
Sample Size (N)	28,590	28,119

Results

- Test Statistic (t): 14.39
- P-Value: 8.00×10^{-47}

Since the p-value is extremely small (much smaller than any common significance level such as 0.05 or 0.01), we reject the null hypothesis. This indicates that there is a statistically significant difference between the means of the uncapped and capped groups.

These results show that the top trades have a disproportionate influence on the overall performance of the trading system. By removing just 1.65% of the top trades, there is a significant drop in average returns, reduced variance, and lower skewness and kurtosis measures. This highlights the importance of these exceptional trades for the overall profitability and volatility of the system. In practice, this means that the success of the trading system relies heavily on a small number of very high-performing trades.

5.4 Data analysis: Improving and optimising the system for version 2

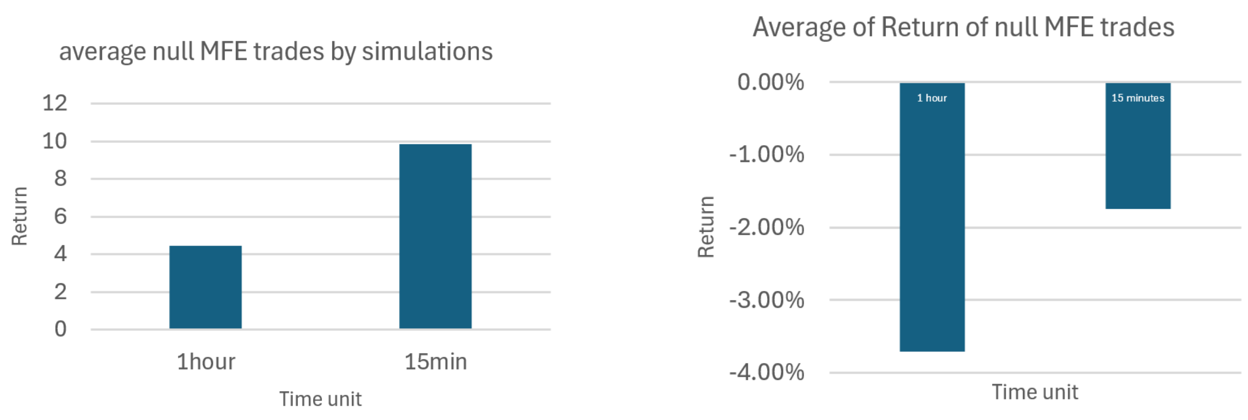
In this section, we will conduct an in-depth analysis of the system's results. The purpose of this analysis is to better understand the behavior of our trading system, as well as to identify areas for improvement and optimization to create a version 2 of the trading system in the next section.

5.4.1 Risk management and exit conditions

1. Takeprofit optimisation

In this part, we will analyze the distribution of the *Maximum Favorable Excursion (MFE)* metric in detail to determine an optimal take-profit level for version 2 of the trading system. It is important to note that we deliberately omitted a take-profit in this version of the system to allow for an unbiased analysis of the MFE. Including a take-profit would have restricted our ability to observe the maximum potential of the trades.

Trades with Zero MFE:



(a) Average number of null MFE per simulation

(b) Average returns of null MFE

Figure 13: Average null MFE and returns of null MFE

In our analysis, we observed that 17.8% of trades have a zero Maximum Favorable Excursion (MFE), which is a significant finding. This indicates that nearly one-fifth of the trades never turned positive at any point, suggesting premature entries. These trades, often referred to as "catching a falling knife," occur when a buy is executed as soon as the RSI closes in the neutral zone but the price continues to decline. This premature buying reflects inadequate timing and contributes to the overall inefficiency of the trading system.

To address this issue, we propose adding an additional entry condition based on price action. This supplementary condition aims to filter out poor signals and prevent premature entries. By incorporating this refinement, we anticipate improving the system's performance and reducing the number of trades with zero MFE. This adjustment will enhance the precision of our entry points, leading to more favorable trade outcomes and a more robust trading strategy overall.

Trades with High MFE:

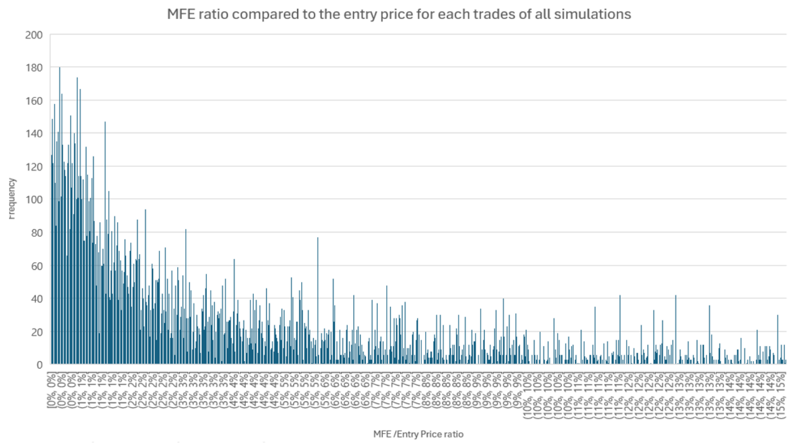


Figure 14: Histogram of the MFE ratio over the entry price

Moreover, we found that approximately 3% of the trades, equating to around 800 transactions, represent the highest and most extreme Maximum Favorable Excursions (MFE), boosting the average MFE by over 15%. This observation highlights that a very small fraction of trades is responsible for almost the entire performance of the trading system.

The implication of this finding is significant. Introducing a take-profit level would mean accepting the potential loss of a portion of the gains from these extreme trades. However, this loss needs to be weighed against the benefits of adding a take-profit level.

To address this, the action plan involves finding a balance by incorporating a take-profit level that captures sufficient gains while also stabilizing the equity curve. The goal is to create a more consistent and less volatile performance profile for the trading system. By carefully setting the take-profit level, we aim to preserve the gains from the majority of trades while minimizing the impact of missing out on those few highly profitable trades. This adjustment is expected to lead to a more stable and reliable trading strategy.

Percentile	Trades %	MFE Ratio
99%	1%	282.72%
95%	5%	78.02%
90%	10%	40.56%
80%	20%	19.13%
70%	30%	9.72%
60%	40%	5.83%
50%	50%	3.68%
40%	60%	2.20%
30%	70%	1.33%
20%	80%	0.75%
10%	90%	0.37%
0%	100%	0.01%

Table 2: MFE by percentile

Determine optimal Takeprofit for V2

It is highly pertinent to analyze the distribution of Maximum Favorable Excursion (MFE) to approximate key metrics such as the win rate, the risk-reward ratio, and consequently, the expected gain per trade by simulating take profit placements at different levels. For example, if we observe that X% of the trades in our system fall below a certain MFE value, this indicates that if we had set a take profit at this MFE value, the system's win rate would have been X%. To calculate the associated risk-reward ratio, we divide this MFE value by the average distance between the entry and the stop loss.

Given that it is possible to approximate the metrics of a variant of the trading system by simulating different levels of take profit with MFE, we aim to understand how varying take profit placements influence the win rate, the risk-reward ratio, and consequently, the expected gain. In other words, is there an optimal level that would maximize the expected gain?

Thus, we conducted a simulation to illustrate how setting the take profit at different levels impacts the expected gain. The results are presented in the graph below.

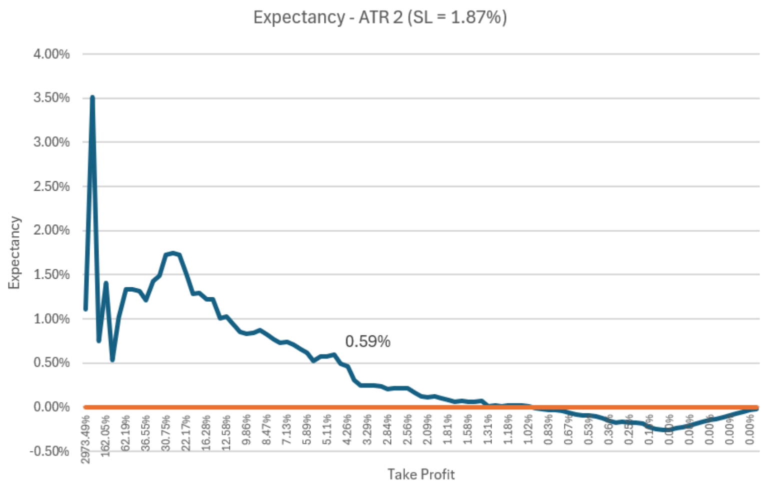


Figure 15: Expectancy of the strategy by take Profit (frequency of 15min)

We can observe that the expected gain per trade tends to increase as the simulated take profit level is set higher. This increase is steady up to a certain point, beyond which it becomes highly variable. At this stage, we see a drastic drop in expected gain, followed by a sharp rise if the take profit is set even higher, tending towards infinity or essentially becoming non-existent.

Take Profit	RR	Win rate	Expectancy
6.70%	3.59	30%	0.70%
6.26%	3.35	31%	0.65%
5.89%	3.16	32%	0.62%
5.39%	2.89	33%	0.53%
5.30%	2.84	34%	0.57%
5.11%	2.74	35%	0.58%
4.96%	2.66	36%	0.59%
4.50%	2.41	37%	0.49%
4.26%	2.28	38%	0.46%
3.71%	1.98	39%	0.31%
3.42%	1.83	40%	0.25%

Table 3: Metrics variation by takeprofit placement

In this table, we can see how increasing the simulated take profit level pushes the expected gain higher due to an improving risk-reward ratio, which more than compensates for the decrease in win rate. There indeed exists an optimal zone for Takeprofit that maximizes the expected gain. According to the graph, this zone is around +25% before the expected gain starts to decline, excluding the upward spike caused by extreme profit outliers. Thus, setting a take profit at +25% provides an optimal combination that balances the trade-off between win rate and risk-reward ratio (see Tradeoff section), thereby maximizing the expected gain in this system and with these backtest data.

Currently, our system, which does not have a take profit, has a win rate of 23.8% with a risk-reward ratio of 5.79. This results in an expected gain per trade of +0.14%. But our simulation provides a framework for predicting the performance of the trading system in V2, assuming that all other factors remain constant. Thus, in version 2, we could set a take profit to increase the expected gain and reduce the variance in the system's returns, thereby improving our Sharpe ratio.

2. Stoploss optimisation

In this part, we will take a closer look at the stop-loss exit condition. Since a stop-loss has already been implemented in this version of the system, we will not be analyzing the Maximum Adverse Excursion (MAE) metric, as it is inherently limited by the current stop-loss. Instead, we will analyze the different stop-loss variants and their impact on the trading system's performance. As a reminder, the system's stop-loss is based on the Average True Range (ATR) indicator, which varies according to market volatility and is calculated using the previous 21 candles.

Stoploss	**ATR 3**	**ATR 4**	**ATR 5**
Mean	-4.12%	-5.45%	-6.72%
Var	0.00	0.00	0.00
Skew	4.50	5.02	4.98
Kurtosis	39.10	45.12	47.16
N	10,582	9,374	8,634
Max	-48.19%	-64.26%	-80.32%
VaR 95%	-10.03%	-12.93%	-15.95%
CVaR	-14.98%	-20.12%	-24.18%

Table 4: Stoploss distribution by ATR Levels

	Return ATR 3	**Return ATR 4**	**Return ATR 5**
Mean	7.24%	8.53%	9%
Variance	0.54	0.69	0.76
Sample Size (N)	10,582	9,374	8,634

Table 5: Return distribution by stoploss

Upon analyzing the table, we observe that the average returns increase as the stop-loss distance increases. This contradicts one of the previously mentioned possibilities in the 7.3.1. In fact, it is the increase in the Win rate that compensates for the decrease in the Risk-Reward ratio, thereby increasing the expected gains when the stop-loss is placed further from the entry point.

However, it is important to note that a greater stop-loss distance indeed leads to a decrease in the Risk-Reward ratio and an increase in the Win rate. Additionally, it results in a decrease in the average position size per trade to maintain the same level of risk per trade. This is crucial to understand how the variation of a single metric within the system can impact multiple central metrics.

Furthermore, we observe that the variance decreases as the stop-loss becomes tighter, confirming the second possible reason for the increase in the Sharpe ratio when the stop-loss is tightened. Thus, a tighter stop-loss results in lower volatility in the returns generated by the system.

We will now conduct statistical tests to determine if there is a significant difference in the returns associated with each stop-loss level, as well as if there is a difference in the variance of returns for each stop-loss level.

T-Test for Difference in Means:

- **Test Statistic (T-value)**: 20.71
- **P-Value**: 3.15×10^{-16}

Since the p-value is less than 0.05, we reject the null hypothesis. This indicates that there is a statistically significant difference in the means of the returns for ATR 3 and ATR 4.

Levene's Test for Difference in Variances:

- **Test Statistic**: 139.04
- **P-Value**: 8.11×10^{-61}

Since the p-value is less than 0.05, we reject the null hypothesis. This indicates that there is a statistically significant difference in the variances of the returns for ATR 3 and ATR 4.

In conclusion the T-test indicates a significant difference in the means of the returns for ATR 3 and ATR 4. Therefore, the ATR level with the higher mean return (ATR 4) is considered better in terms of return. However, Levene's test also indicates a significant difference in variances, with ATR 3 having a lower variance. As observed insection 5.3.1 , despite the decrease in returns when the stop-loss is tightened, the volatility also decreases, resulting in an improved Sharpe ratio with a closer stop-loss. Therefore, in Version 2 of the system, we aim to achieve a better Sharpe ratio by setting a tighter stop-loss.

3. Position size and risk per trade

In this part, we will aim to determine the optimal risk per trade based on the system's winning rate and risk-reward ratio. The risk per trade represents the amount the system is willing to lose in the event of a losing trade. This metric inevitably affects the position size associated with each trade and, consequently, has a significant impact on the trading system's performance. Our objective in this part is to conduct simulations to find the risk per trade that minimizes the risk of ruin while maximizing capital growth.

In the realm of trading, the Kelly formula is a well-known method for calculating the optimal allocation per trade, given the Win rate and the Risk-Reward ratio. The Kelly formula is designed to maximize the growth rate of capital by determining the optimal fraction of the total capital to allocate for each trade. The formula is expressed as follows:

$$f^* = W - \frac{(1 - W)}{R}$$

where:

- f^* is the optimal fraction of the total capital to allocate per trade,
- W is the Win rate, representing the probability of a winning trade,
- R is the Risk-Reward ratio, defined as the ratio of the average win to the average loss.

This formula calculates the fraction of capital that should be risked on each trade to maximize the logarithm of wealth. However, it is important to note that we do not know the actual Win rate and Risk-Reward ratio but only approximations based on our backtest results. The inherent uncertainty in these approximations means that the real-world performance might differ significantly from the backtested results.

Additionally, one of the main drawbacks of the Kelly criterion is that it can lead to very large drawdowns, which can be psychologically challenging for traders and potentially lead to significant capital loss during extended losing streaks. Therefore, while the Kelly formula provides a theoretical framework for optimal capital allocation, its practical application must be carefully considered, and often a fraction of the Kelly value is used to mitigate the risks associated with large drawdowns.

In our case, we will conduct a simulation to understand the impact of different levels of risk per transaction on returns and drawdown. In this simulation, we have considered the following components: the simulation operates on a 15-minute trading time frame, the stop loss is set at 2 ATR with an average distance of 1.87% from the entry, the win rate is 36.7%, and the risk-reward ratio is 2.74 . We will run the simulation for various levels of risk per trade to analyze their effects.

Capital Risked	**Equity traded**	**Annualized Return**	**Max Drawdown**	**Ratio R/DD**
1%	77%	112%	-17%	6.6
2%	154%	303%	-31%	9.7
3%	231%	602%	-44%	13.7
4%	308%	1034%	-54%	19.1
5%	385%	1612%	-63%	25.6
6%	462%	2329%	-70%	33.1
7%	539%	3152%	-76%	41.4
8%	616%	4021%	-81%	49.6
9%	693%	4856%	-85%	57.2
10%	770%	5570%	-88%	63.2
11%	835%	6082%	-91%	67.0
12%	911%	6337%	-93%	68.3
13%	987%	6312%	-94%	66.9
14%	1063%	6022%	-96%	63.0
15%	1139%	5513%	-97%	57.1

Table 6: Simulation of multiple risk levels per trade

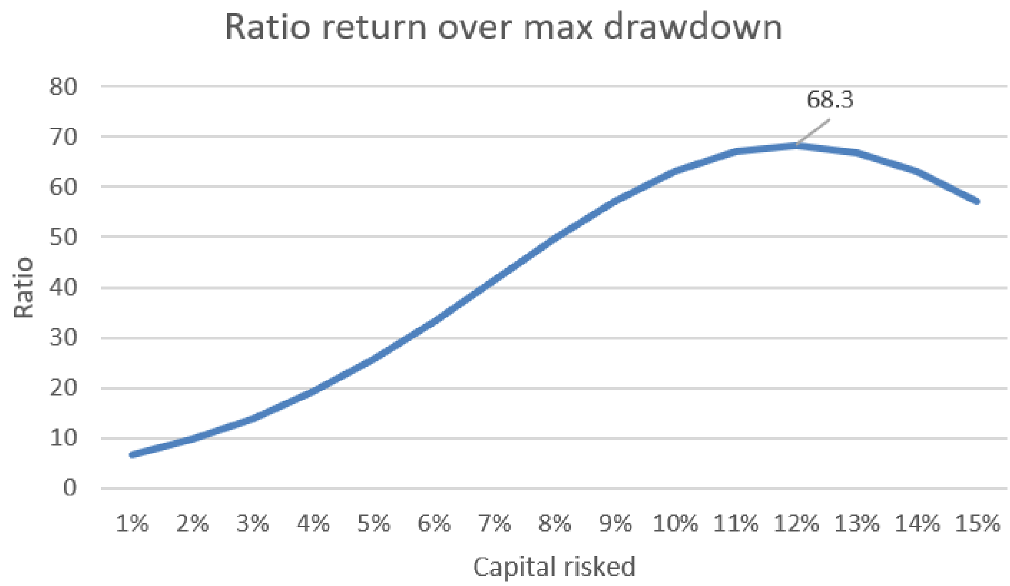


Figure 16: Ratio return over max drawdown

We observe that at the optimal Kelly allocation, the Return/Drawdown ratio is the highest. However, as previously explained, the max drawdown resulting from this allocation is very high, which may not align with most risk profiles. If we aim for a lower drawdown to match our risk profile constraints, the Return/Max Drawdown ratio decreases as we lower the allocation and capital risked per trade. This means that for each unit of drawdown, we should expect fewer units of associated return. In other words, the return decreases more proportionally than the drawdown when we reduce the capital risked per trade.

Ultimately, aiming for a drawdown that matches our risk profile means accepting a trade-off where we lose a portion of the return associated with each unit of drawdown. It is evident that selecting a lower allocation than the optimal Kelly criterion is prudent, as the Kelly allocation is based on estimated win rates and risk-reward ratios derived from backtesting. Markets are constantly evolving, and this calls for caution. Therefore, it is advisable to adopt a fraction of the optimal Kelly allocation to ensure against bankruptcy in the event of sudden market changes. On the other hand, choosing a risk per trade that is too low is also suboptimal, as it results in a proportional decrease in returns relative to the drawdown. Ultimately, there is a reasonable zone between these two extremes that seems appropriate.

In other words, being overly cautious can be detrimental to performance, while being too aggressive by strictly applying the optimal Kelly allocation without considering and adapting to the ever-evolving financial markets can clearly lead to bankruptcy. Therefore, it is necessary to make an arbitrary choice that balances excessive caution and excessive aggressiveness. In our case, we believe that a risk per trade between 2% and 3% is appropriate and aligns well with our risk profile.

5.4.2 Strategy and entry conditions

In this section, we will first analyze the market conditions that can influence the trading system's results. In the second part, we will examine the various parameters of our system and the impact of their variations on the trading system's performance. The goal of this in-depth analysis is to identify potential improvements and optimizations in the strategy that makes up our trading system. For example, we may consider adding more precise entry conditions in the next version of the trading system.

1. Impacting market conditions

In this part, we will analyze various categories of market conditions to determine their impact on the trading system's results. Specifically, we will examine the direction of the market trend and market volatility.

First, we will analyze our system's performance according to whether the market is in an uptrend or a downtrend, in order to understand the impact of this market condition.

Trend	**Uptrend**	**Downtrend**
Mean	0.32	0.04
Variance	2.86	0.07
Skew	7.49	7.86
Kurtosis	60.90	107.59
N	5836	4260
Win rate	39%	37%
Return over 100%	6.0%	1.1%
VaR (95%)	-9%	-13%
CVaR (95%)	-12%	-19%

Table 7: Return distribution by market trends

The first thing we notice is that there is only a 27% difference in occurrences when moving from a bullish market to a bearish market. Given that our system is "long only" and need to be in a bullish market regime, we expected a more significant decrease in the number of opportunities when the system is in a bearish market. This result can likely be explained by the fact that the system operates on 15-minute and 1-hour time frames, where temporary bullish trends can frequently occur even within a predominantly bearish market. Therefore, the impact of a longer-term bearish market is less significant on a trading system that operates on shorter time frames.

On the other hand, we observe that the average return is significantly lower when the system operates in a bearish market compared to a bullish one. More precisely, the number of trades with a return above 100% is nearly six times higher in a bullish market, confirming that it is indeed the extremely positive trades that strongly boost the average return. This phenomenon is further corroborated by the fact that the variance is much higher in a bullish market.

However, when we focus on the win rate metric, it remains very similar regardless of the market trend. The fundamental difference in results between the two types of markets lies in the number of trades with extremely positive returns, which is significantly higher in a bullish market, thus boosting the average return. This makes sense, as a bullish market trend allows trades to capitalize on strong upward momentum, which is precisely what the trading system was designed to capture.

T-Test for Mean Difference:

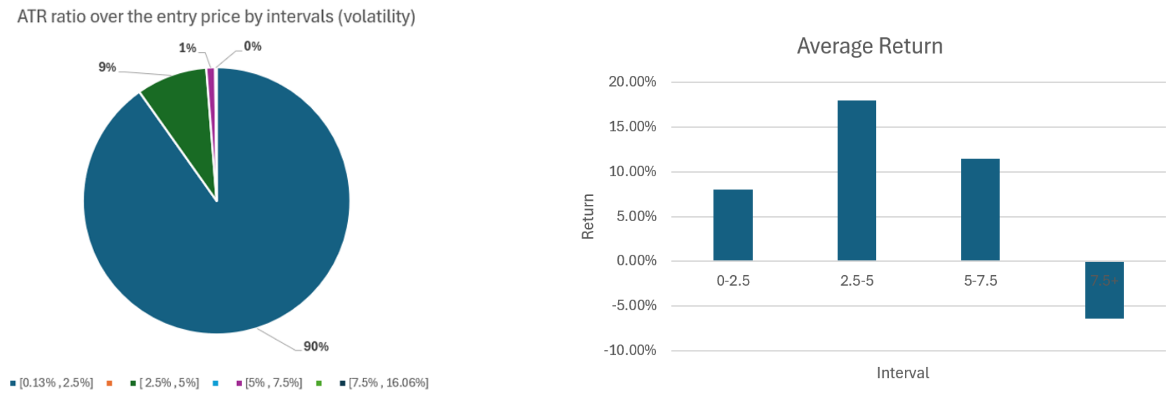
- Uptrend vs Downtrend: $t = 15.02$, $p < 0.001$

Levene's Test for Variance Difference:

- Levene's Statistic: $F = 395.22$, $p < 0.001$

Based on the t-test, there is a significant difference in the mean returns between the uptrend and downtrend ($p < 0.001$). The Levene's test indicates that there are significant differences in the variances between the uptrend and downtrend ($p < 0.001$) which confirms our analysis.

Impact of the volatility (ATR based):



(a) Entry Repartition by ATR Intervals (b) Average Return by ATR intervals

Figure 17: Entry repartition and average return by ATR intervals

The goal here is to understand how the system behaves depending on market volatility. We have used the ATR (Average True Range) indicator to measure this volatility, where higher values indicate a more volatile market.

Intervals	**0-2.5**	**2.5-5**	**5-7.5**	**7.5+**
Mean	8.01%	18.01%	11.44%	-6.43%
Var	0.60	1.61	1.29	0.16
Skew	15.82	10.33	6.45	1.46
Kurt	288.67	118.42	41.01	2.94
N	24323	2309	289	46
Winrate	31.94%	40.28%	26.99%	41.30%
Return over 100%	1.40%	5.07%	3.46%	6.52%

Table 8: Return distribution by ATR Intervals

We observe that 90% of the system's opportunities are found in a relatively low volatility market, where the ATR represents between 0 and 2.5% of the price. This is beneficial given that the stop-loss is based on the ATR, meaning a less volatile market results in a tighter stop-loss and thus a better risk-reward ratio.

However, we observe that the best average returns are found when the ATR value is between 2.5% and 5.0%, with an average return of +18% compared to +8% when the ATR is between 0% and 2.5% of the price. Notably, the number of trades with returns over 100% is almost four times higher in the ATR range of 2.5-5.0%. We have a hypothesis concerning the reason for this better performance when the ATR is high.

We believe that a relatively high ATR, indicating a very volatile market, leads to sharp movements that cause the RSI indicator to reach extreme zones, such as the oversold area. Due to the system's exit condition linked to the breakout of the neutral zone, this condition becomes harder to execute as the neutral zone is positioned at an extreme RSI level. It therefore takes longer for the market to reach this extreme zone again and execute the breakout condition on the RSI and this gives the trade more freedom to perform over time. This hypothesis will be explored when we analyze returns by RSI intervals in the next part.

T-Tests for Mean Differences:

- 0-2.5 vs 2.5-5: $t = -24.94$, $p < 0.001$
- 0-2.5 vs 5-7.5: $t = -6.72$, $p < 0.001$
- 2.5-5 vs 5-7.5: $t = 14.50$, $p < 0.001$

Levene's Test for Variance Differences:

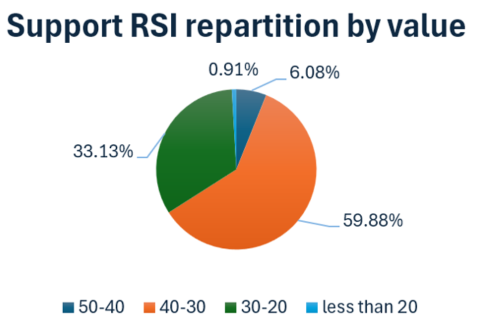
- Levene's Statistic: $F = 112.84$, $p < 0.001$

In conclusion based on the t-tests, there are significant differences in the mean returns between the intervals (0-2.5 vs 2.5-5, 0-2.5 vs 5-7.5, and 2.5-5 vs 5-7.5, all with $p < 0.001$). The Levene's test indicates that there are significant differences in the variances across these intervals ($p < 0.001$). This confirms our analysis of the significant difference in returns per ATR interval due to a higher number of extremely profitable trades, which corresponds to the increase in variance.

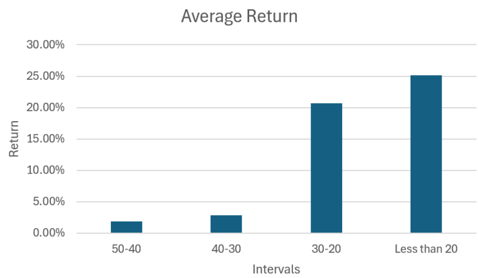
2. Impacting system parameters

In this part, we will analyze the various parameters that constitute our trading system to identify their impact on the system's performance. We will examine how the placement of the RSI neutral zone affects the results, the influence of transaction costs implemented in our simulations, and the impact of different time frames in which trades are executed. Additionally, we will explore the effects of different simulated parameters, such as margin level, window size, cluster, and the ATR multiplier for the stop-loss. This detailed analysis will help us understand the influence of each parameter on the trading system and provide insights for optimizing and improving the system in its next version.

Impact of RSI support value:



(a) Percentage of RSI support by intervals



(b) Average returns by intervals

Figure 18: Support RSI repartition and average return by intervals

The neutral zone, which is the cornerstone of our trading strategy, is calculated based on the RSI. In this part, we will analyze the placement of this neutral zone by RSI interval.

Intervals	**50-40**	**40-30**	**30-20**	**Less than 20**
Mean	0.02	0.03	0.21	0.25
Variance	0.09	0.11	1.82	1.06
Skew	15.88	26.88	9.41	3.34
Kurtosis	263.68	931.49	98.26	10.23
N	1640	16138	8928	246
Win rate	29%	34%	31%	13%
Return over 100%	0.4%	0.6%	3.8%	8.5%
VaR (95%)	-5%	-8%	-10%	-14%
CVaR (95%)	-8%	-12%	-14%	-17%

Table 9: Return distribution by RSI intervals

When analyzing the distribution of occurrences by RSI interval, we observe that over 90% of the opportunities fall within the RSI range of 20 to 40, with a significant majority specifically in the 30 to 40 interval. There are very few occurrences between 40 and 50, and even fewer below the level of 20.

In terms of returns, we notice a significant increase as the RSI interval decreases. The average returns per trade exceed 20% when the buy zone is below the 30 level on the RSI. This supports our earlier hypothesis that the lower the neutral zone is on the RSI, the harder it becomes to execute the exit condition related to breaking out of the neutral zone. Indeed, the more extreme this zone, the harder it will be to reach it again. This prolongs the duration of holding positions and requires another extreme movement for the breakout exit condition to be met and executed.

In version 2, it could be beneficial to focus the trading system on lower RSI intervals to maximize the average return. However, adding this additional entry condition would reduce the number of opportunities by approximately 70%. This addition would result in higher-quality opportunities but at the cost of losing about two-thirds of the total opportunities. This example aptly illustrates the trade-off concept discussed in Tradeoff section, where adding a filter improves the quality of opportunities but decreases the total number of opportunities in the market.

We will now conduct statistical tests to confirm if there is a significant difference between the average returns generated across different RSI intervals:

T-Tests for Mean Differences:

- 50-40 vs 40-30: $t = 0.70$, $p = 0.483$
- 50-40 vs 30-20: $t = -5.16$, $p < 0.001$
- 40-30 vs 30-20: $t = -7.02$, $p < 0.001$

Levene's Test for Variance Differences:

- Levene's Statistic: $F = 13.94$, $p < 0.001$

Based on the t-tests, the mean returns for the intervals 50-40 and 40-30 are not significantly different ($p > 0.05$). However, there are significant differences in the mean returns between the other intervals (50-40 vs 30-20 and 40-30 vs 30-20, both with $p < 0.001$). In addition, the Levene's test indicates that there are significant differences in the variances across the different return intervals ($p < 0.001$).

Impact of Transaction costs:

Here we will analyse the impact of transaction costs on the performance of our trading system. The transactions costs on Binance for spot trading is the following: 0.1% to the entry capital and 0.1% to the exit capital. This two-step fee has the following equation:

$$\text{Transaction costs} = \text{Amount traded} \times (0.001999 + 0.000999r) \tag{28}$$

$$\frac{dTC}{dAT} = 0.001999 + 0.000999r \tag{29}$$

$$\frac{dTC}{dr} = 0.000999 \times AT \tag{30}$$

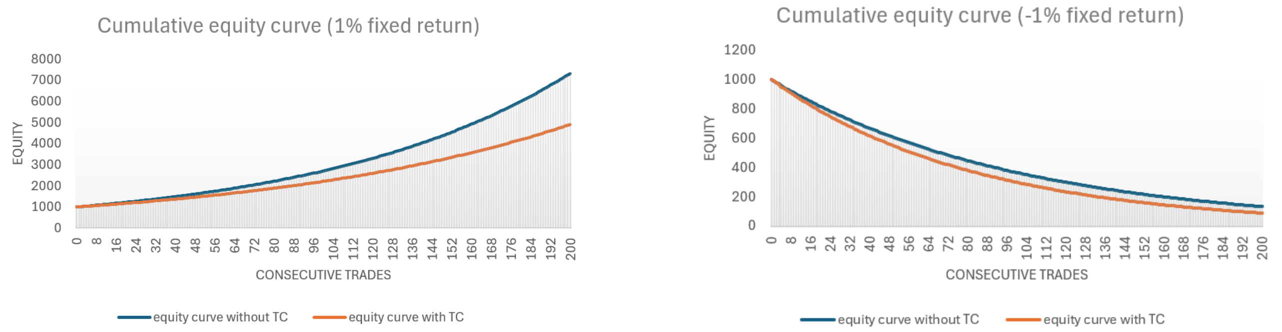
where:

- **AT** is the amount traded
- **r** is the trader's return at the end of the transaction

This equation represents the transaction costs per trade. It has two variables: the amount traded, which represents the entry capital, and the market return of the specific asset, r.

The partial differentiation shows a positive correlation for the two variables, which makes total sense from the brokerage perspective.

Note that since we subtract 0.01% in transaction costs when the trade is committed, the position size will be equal to 99.9% of the capital actually committed, which explains the formula developed above.



(a) Cumulative impact on TC for positive fixed returns

(b) Cumulative impact on TC for negative fixed returns

Figure 19: Cumulative impact on transaction costs for fixed returns

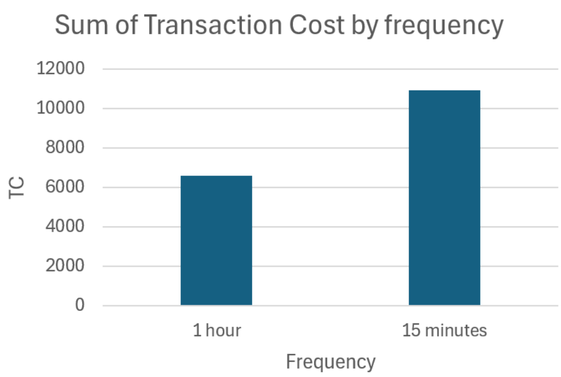


Figure 20: Sum of transaction costs by frequency

The graph with the equity curve simulation clearly demonstrates that the impact of transaction costs becomes exponential as the number of trades increases, especially when the equity curve is in an upward trend. Additionally, considering the opportunity costs, the money spent on transaction costs is money that has not been reinvested into the trading system, thereby dampening the exponential effect due to the phenomenon of compounding interest when a portion of the profits is reinvested after each transaction. Ultimately, the gap between the equity curve with and without transaction costs becomes exponentially larger as the number of trades increases.

We can see that as trades multiply in increasingly smaller time units, the impact of transaction costs inevitably becomes even greater. Therefore, for scalping trading, which involves a large number of transactions lasting only a few seconds or minutes, it becomes essential to have the lowest possible transaction costs because the final performance will be significantly impacted.

Impact of the trading time frame:

The results presented in the table show the performance statistics of a trading system applied to two different time frames: a 1-hour (H1) time frame and a 15-minute (M15) time frame.

	Returns 1H	Returns 15min
Mean	11.00%	7.35%
Variance	0.74	0.62
Sample Size (N)	7,673	20,917

Table 10: Return distribution by frequency

First, we can observe that the average returns of trades are significantly higher for the 1-hour time frame (11.00%) compared to the 15-minute time frame (7.35%). This indicates that trades taken over a longer period tend to be more profitable on average. This is quite logical, as the longer the trading time frame, the more likely the system is to capture major upward movements.

On the other hand, the number of trades is significantly higher in the M15 time frame (20,917) compared to H1 (7,673), which makes sense. Therefore, in H1, we have almost three times fewer trading opportunities than in M15. In discretionary trading, one might prefer trades in H1 due to their seemingly higher quality. However, in algorithmic trading, where capturing a large number of opportunities is not a constraint, the M15 time frame would be preferred as it presents more opportunities. Nevertheless, as mentioned earlier, while the M15 time frame offers more opportunities, traders must also consider the higher transaction costs due to the increased number of trades.

We will now perform a statistical test to confirm that there is indeed a significant difference in returns between the two time-frames:

- Test Statistic (t): **34.72**
- Degrees of Freedom (df): **11,585.89**
- P-Value: 2.14×10^{-256}

Since the p-value is extremely small (much smaller than any common significance level such as 0.05 or 0.01), we reject the null hypothesis. This indicates that there is a statistically significant difference between the means of the returns for 1-hour and 15-minute Time unit.

Impact of the margin parameter:

This time, the results presented in the table show the performance statistics of our system applied to two different margin levels representing the width of the neutrality zone in our trading system.

	Returns marge3	Returns marge5
Mean	5.15%	13.78%
Variance	0.32	1.21
Sample Size (N)	18,042	10,548

Table 11: Return distribution by margin level

We observe that in terms of returns, when the margin parameter is set to 5, the returns are significantly better at 13.78% compared to 5.15% with the margin set to 3. Although a higher margin leads to a more distant exit condition due to the stop loss based on the breakout of the neutral zone, the returns are better. This is likely because a higher margin results in a better win rate, as it allows more "margin" for market noise.

On the other hand, we observe that the number of occurrences is almost twice as low when the margin is set to 5. We believe this is due to the fact that trades last longer when the margin is higher, requiring more time to close the trade and open a new one. In contrast, with a margin of 3, more trades are executed as they are stopped more quickly by the breakout exit condition.

We will now perform a statistical test to confirm that there is indeed a significant difference in returns between the two margin level:

- Test Statistic (t): **42.96**
- Degrees of Freedom (df): **18,246.70**
- P-Value: 0.0 **(practically zero)**

Since the p-value is extremely small too, we reject the null hypothesis. So there is a statistically significant difference between the means of the margin returns and this confirms our analysis .

Impact of the window size parameter:

In this case, the results presented in the table show the performance statistics of our system applied to three different window sizes representing the time and number of periods to calculate the neutrality zone in our trading system.

	Returns 100	**Returns 200**	**Returns 400**
Mean	8.10%	8.25%	8.70%
Variance	0.68	0.59	0.69
Sample Size (N)	10,381	9,541	8,668

Table 12: Return distribution by window size

We notice that as the window size increases, the average return slightly increases as well. However, this comes at the cost of a decrease in the number of occurrences. This makes sense because if it takes more time to calculate and consider the neutral zone, there will be less time available to trade within this neutral zone.

We will now perform a statistical test to check if there is indeed a significant difference in returns between the different window sizes:

- Test Statistic (ANOVA F-value): **2.27**
- Degrees of Freedom (Between Groups): **2**
- Degrees of Freedom (Within Groups): **28,587**

- P-Value: **0.104**

Since the p-value is greater than the common significance level of 0.05, we fail to reject the null hypothesis. This indicates that there is no statistically significant difference between the means of the three groups.

We will now perform a statistical test to check if there is a significant difference in variance this time between the different window sizes:

We choose the Levene's test which is more robust to deviations from normality compared to other tests for equality of variances, such as Bartlett's test. This is important because financial return data often do not follow a normal distribution. Furthermore Levene's test is designed to handle comparisons across multiple groups, making it appropriate for comparing the three window sizes.

- Test Statistic: **139.04**
- P-Value: 8.11×10^{-61}

While the ANOVA test indicates no statistically significant difference in the means of the three groups, Levene's test shows a significant difference in variances. The comparison of variances reveals that the Returns 200 group has the smallest variance, making it the preferable choice for stability.

Impact of the cluster parameter:

Here the results presented in the table show the performance statistics of our system applied to two different level of cluster representing the clusters of local extremes on the RSI which are used to calculate the strategy's neutrality zone.

	Returns 25%	**Returns 33%**
Mean	7.17%	9.73%
Variance	0.53	0.80
Sample Size (N)	15,607	12,983

Table 13: Return distribution by cluster

The same as before, we notice that as the cluster level increases, the average return slightly increases as well. However, this comes at the cost of a decrease in the number of occurrences. This makes sense because as the cluster parameter decreases, creating a higher fraction of the window size for clustering, the rolling window used to calculate maximum and minimum values becomes smaller.

This results in more granular data points and potentially more clusters, each representing more specific local maxima and minima within the data. As clusters become more defined and precise, they naturally capture fewer occurrences each but potentially more significant ones, reflecting crucial turning points in the RSI that are more likely to indicate meaningful support and resistance levels. Consequently, while the average return per identified opportunity may increase due to the specificity of these levels, the total number of trading signals generated decreases.

This trade-off highlights the importance of optimizing the cluster to balance the frequency of trade opportunities with the quality and potential return of each.

We will now perform a statistical test to check if there is a significant difference in returns between the two different level of cluster:

- Test Statistic (t): **24.27**
- Degrees of Freedom (df): **25,421.36**
- P-Value: 1.4×10^{-129}

Since the p-value is extremely small (much smaller than any common significance level such as 0.05 or 0.01), we reject the null hypothesis. This indicates that there is a statistically significant difference between the means of the returns for the 25% and 33% cluster fractions.

5.5 Improved Trading System (Version 2)

In this section, we will present the final results of our trading system for this thesis. First, we will discuss the objectives that guided the changes and improvements made in Version 2 (the final version for this thesis). We will then review the results of our final trading system, applied to a new data sample, by examining the main metrics that reflect the detailed performance of our trading system.

5.5.1 Objectives

In this subsection, we will first discuss the objectives pursued with Version 2 of our trading system. Next, we will outline the various changes and improvements made in this second and final version of our trading system for this thesis.

System Improvements:

When designing a trading system, the objective is not solely to maximize returns. The process is more complex, involving various goals that must be considered during the development phase. It is crucial to define these objectives clearly before proceeding, as there are inherent trade-offs between the different goals pursued when enhancing a trading system.

Maximizing returns and generating alpha is one of the most popular objectives when developing a trading system. However, there are numerous other goals that can be pursued. Here are a few non-exhaustive examples: Minimizing beta, or correlation with the benchmark, aims to generate returns that are partially or entirely uncorrelated with market conditions, ensuring income even when the markets are down. Another possible objective is to maximize the return adjusted for time in position, considering the opportunity cost of investing money elsewhere. Additionally, minimizing drawdown while ensuring a reasonable return is crucial for risk-averse investors who still want to grow their capital. In this case, analyzing the return/drawdown ratio is pertinent.

To illustrate, consider the formula for alpha:

$$Generated\ Alpha = R_p - (R_f + \beta \times (R_m - R_f)) \tag{31}$$

Where:

- R_p is the average return of the portfolio (trading system) over the backtesting period.
- R_f is the risk-free rate of return.
- β is the portfolio's sensitivity to movements in the benchmark.
- R_m is the return of the benchmark index.

Alpha represents the excess return of an investment relative to the return of a benchmark index. A positive alpha indicates that the investment has outperformed the market after adjusting for risk, while a negative alpha suggests underperformance. In our context, maximizing alpha involves increasing the system's returns and minimizing its beta (correlation with the benchmark). This aims to generate additional returns that are uncorrelated with the benchmark.

For the objectives we have decided to pursue, we aim to maximize the Sharpe ratio, as it is a comprehensive measure reflecting the performance of a trading system. The Sharpe ratio takes into account both return and risk through the variance of returns. Simultaneously, we will focus on maintaining a high number of occurrences to maximize the overall expected return of the trading system. This involves a trade-off, as detailed in Tradeoff section, where increasing the system's complexity by adding entry conditions can improve the win rate and consequently the Sharpe ratio. However, by deliberately including the maximization of the system's overall expected return, we must keep the number of occurrences high, thereby potentially sacrificing an even higher Sharpe ratio.

The formula for the Sharpe ratio is:

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p} \tag{32}$$

Where:

- R_p is the average return of the portfolio over the backtesting period.
- R_f is the risk-free rate of return, which is the return of an investment with no risk of financial loss.
- σ_p is the standard deviation of the portfolio returns, representing the volatility or risk of the portfolio.

Ultimately, our objectives can be summarized as maximizing the trade-off between the Sharpe ratio and the number of occurrences. We will not focus on beta or time in position for this thesis, as our objective is to demonstrate that technical analysis is a viable tool for outperforming the markets. Therefore, the most important metrics for improving Version 2 of the system are return, volatility, and max drawdown.

In conclusion, our success will be measured by consistently outperforming the benchmark in both the return/drawdown ratio and the Sharpe ratio. This approach ensures that we consider both the returns and the risks involved, providing a robust evaluation of our trading system's performance.

Changes from Version 1 to Version 2:

In this section, we will discuss all the changes made to the trading system to develop Version 2.

First, we adjusted the risk management parameters. While the risk per trade in the previous version was set at 2%, we initially aimed to increase this to 4% for the 1-hour (H1) time frame and 3% for the 15-minute (M15) time frame. However, due to the constraint that our position size is capped at a maximum of 100% of our capital, as we do not use leverage in this thesis, the actual avarage risk per trade remains very close to 2%.

For instance, if our stop-loss is set at a distance of 2% from the entry price and we intend to risk 4% of our capital on this trade, we would need to allocate a position size equivalent to 200% of our capital, which is impossible without leverage. Therefore, despite our intention to increase the risk per trade, the practical implementation keeps the risk per trade near the 2% threshold. This adjustment was based on our simulation results in section 5.4.1, which helped us identify an appropriate risk per trade aligned with the win rate and average risk/reward ratio (RR) for both H1 and M15 time frames. Additionally, we considered our risk profile, opting for a slightly sub-optimal risk per trade according to the Kelly criterion to avoid enduring excessively high maximum drawdowns.

Next, we addressed changes to money management. We adjusted our stop-loss distance to 2x the Average True Range (ATR). This decision aims to minimize the downside variance of returns and thus maximize the Sharpe ratio, aligning with our objectives. Additionally, this adjustment increases the average RR, compensating for the potential decrease in the win rate and resulting in a better expected return per trade due to a more optimal trade-off between RR and win rate, as discussed in Tradeoff section.

Regarding the take-profit condition, we decided not to add a take-profit level, maintaining the same structure as in Version 1. The strategy's goal is to ride bullish trends, and setting a take-profit would counteract this objective. Furthermore, placing a take-profit based on previously simulated optimal levels (see section 5.4.2) could lead to overfitting, reducing the strategy's effectiveness on new data samples.

Now, let's discuss the entry conditions of our system. We adjusted our margin parameters for each frequency to 5, and adjusted the cluster to 33% of the window size. Additionally, we optimize the window size to 100 and 200, respectively for the 15min and 1H strategy. While we considered adding new entry conditions, such as only executing trades when the RSI is below 30, we chose not to add any entry conditions. This decision was made to maintain a high number of occurrences, ensuring both the robustness of our results and maximizing the system's overall expected return, even though the expected return per trade could be improved with additional entry conditions. This concept was explored in depth in Tradeoff section, where we concluded that maximizing the number of occurrences generally leads to better overall expected returns in algorithmic trading.

Despite the clear evidence that our results are better on the 15-minute time frame compared to the 1-hour time frame, we decided to conduct backtesting on both time frames. This approach allows us to confirm or refute our analysis. Additionally, we retained the exit condition related to the breakout of the neutrality zone, which also indirectly serves as a trailing stop, allowing profits to be realized without setting a direct take-profit level.

Moreover, we update the conditions to our support computation by limiting the duration of our support. Indeed, if no breaks have been detected after a certain amount of time (same as the window size), we recompute a new support based on the last periods. The memory on prices to decrease the more we move away from the initial period (De Prado Lopez, 2018)[25], which means that older data points have less influence on the newly computed support. This adaptive approach ensures that our support levels remain relevant and reflect the most recent price action, improving the responsiveness and accuracy of our trading strategy. By dynamically adjusting support levels based on recent price data, we can better capture and respond to changes in market conditions, enhancing the overall performance of our system.

5.5.2 Analysis of results

In this subsection, we have backtested our improved trading system according to the various points previously outlined, using a new dataset. We will conduct a comparison with the benchmark and analyze the main metrics of the system to evaluate its performance when applied to both a 1-hour time frame and a 15-minute time frame.

Strategy Returns VS benchmark:

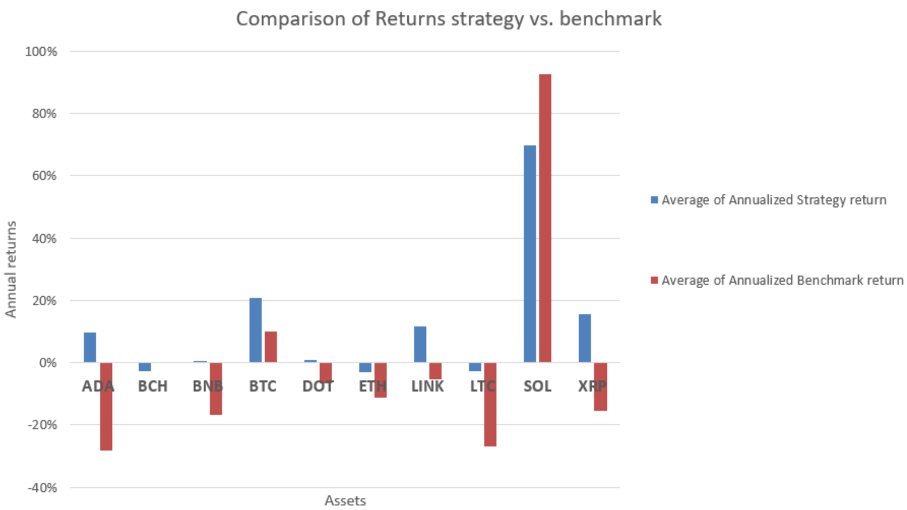


Figure 21: Average Annualized Return strategy V2 VS benchmark by asset (1H)

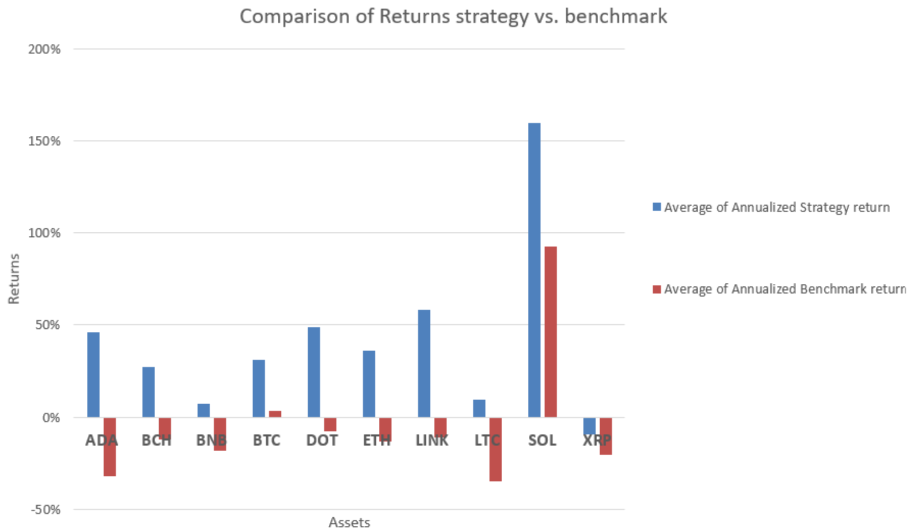


Figure 22: Average Annualized Return strategy V2 VS benchmark by asset (15min)

Above, we have the return charts of our trading system compared to the benchmark for each of the 10 cryptocurrencies on which we applied the backtest. The results are presented for the system applied to both a 1-hour time frame and a 15-minute time frame.

We consistently observe that our system outperforms the benchmark across all 10 cryptocurrencies when applied to a 15-minute time frame. However, this is not the case for the 1-hour time frame, where our system outperforms the benchmark 8 out of 10 times (8 out of 10 isn't too shabby either!). This confirms our analysis in the previous section 5.4.2, where we concluded that our system indeed performs better on a 15-minute time frame.

We can analyze further by noting that despite the negative returns of the benchmark, our trading system manages to perform well even in a bearish trend, a market condition identified as unfavorable in our analysis in section 5.4.2 The system achieves positive returns 9 out of 10 times in the 15-minute time frame and 5 out of 10 times in the 1-hour time frame. The highest return is observed with the cryptocurrency Solana, which is not surprising given its performance reflecting a bullish trend. This market condition is favorable to our system, allowing it to effectively capitalize on upward impulses, as it was designed to do.

To conclude on the returns, we observe significantly better performance when the system is applied to a 15-minute time frame compared to the 1-hour time frame, which confirms our expectations. Additionally, the trading system demonstrates resilience with commendable performance, outperforming the benchmark most of the time despite the bearish trend indicated by the benchmark's negative performance. Lastly, we note an exceptionally high return when market conditions are favorable, as seen with the asset Solana, which achieved a return close to +160% compared to the benchmark's +90% in m15 time frame.

Strategy Drawdowns VS benchmark:

This time we have the max drawdowns charts of our trading system compared to the benchmark for each of the 10 cryptocurrencies on which we applied the backtest. The results are presented for the system applied to both a 1-hour time frame and a 15-minute time frame.

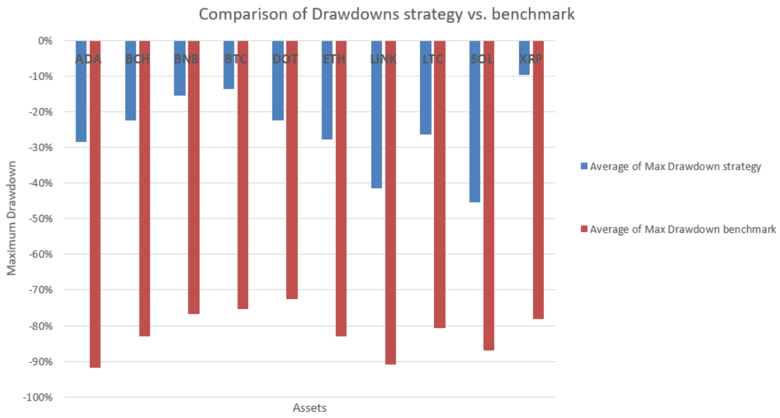


Figure 23: Max Drawdown strategy V2 VS benchmark by asset (1H)

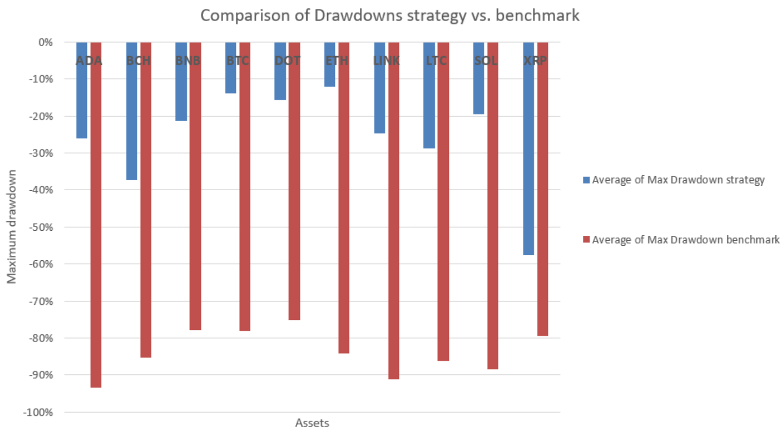


Figure 24: Max Drawdown strategy V2 VS benchmark by asset (15min)

We can observe that, whether applied to a 1-hour time frame or a 15-minute time frame, our trading system consistently outperforms the benchmark with a significantly lower maximum drawdown. In all 20 instances, our system achieves a better max drawdown compared to the benchmark, demonstrating effective and controlled risk management within our trading system. This consistent performance highlights the robustness of our system in mitigating risk across different time frames.

If we analyze the results in detail, we can observe that the main differences between the 1-hour and 15-minute time frames occur with Solana and XRP. Our trading system exhibits a significantly lower max drawdown on Solana in the 15-minute time frame compared to the 1-hour time frame. This can be explained by the fact that Solana's bullish trend is very abrupt and powerful, with minor corrections that the system applied to the 15-minute time frame can capture, making the system more effective in this shorter time frame. We will illustrate this further with the profit curve.

Conversely, for XRP, the max drawdown is significantly lower in the 1-hour time frame compared to the 15-minute time frame. This can be attributed to XRP's less powerful trend, making the 1-hour time frame more suitable for better performance in such situations. Our hypothesis is that the stronger the trend, the more relevant the trading system applied to a shorter time frame becomes for capturing the impulses and corrections of that trend. The previous return graphs support this analysis.

Strategy Sharpe ratio VS benchmark:

This time, we are going to analyse the sharpe ratios of our trading system compared to the benchmark for each of the 10 crypto-currencies on which we carried out the backtest. The results are presented for the system applied to both a one-hour time frame and a 15-minute time frame.

Since the Sharpe ratio provides a more comprehensive measure of system performance than just return or max drawdown, we will analyze the return adjusted for each unit of risk that our trading system is capable of generating. This will give us a clearer understanding of its ability to truly outperform the benchmark

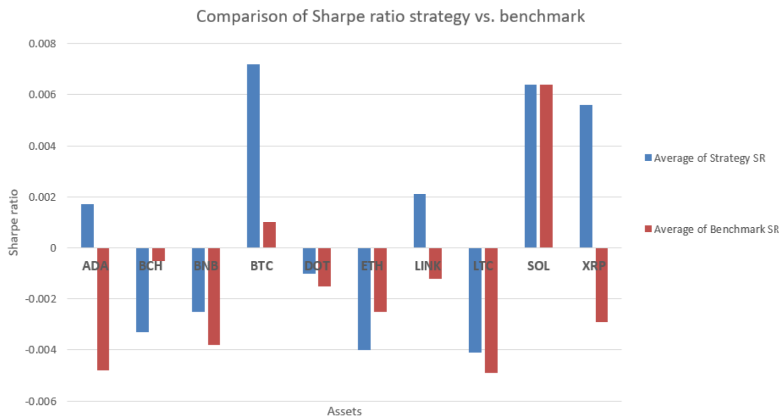


Figure 25: Sharpe ratio strategy V2 VS benchmark by asset (1H)

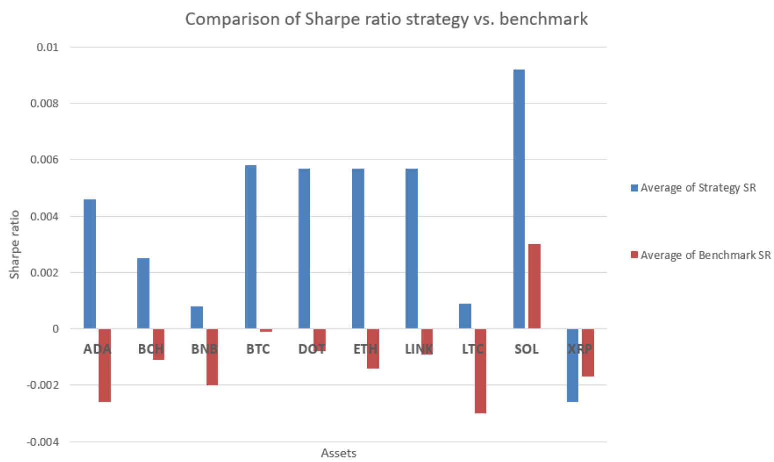


Figure 26: Sharpe ratio strategy V2 VS benchmark by asset (15min)

Regarding the Sharpe ratio of our trading system applied on a 1-hour time frame, we observe that our system outperforms its benchmark 8 out of 10 times. When the system is applied on a 15-minute time frame, we see outperformance in 9 out of the 10 cryptocurrencies. This outcome is not particularly surprising given our system's outperformance in terms of returns. It logically follows that the Sharpe ratios would also outperform, considering the Sharpe ratio formula, which incorporates both returns and volatility.

Upon closer analysis, we observe that despite the benchmark's Sharpe ratio being negative 9 out of 10 times on the 15-minute time frame, our system successfully achieves a significantly positive Sharpe ratio. This demonstrates our system's ability to generate positive returns while maintaining reasonable volatility, even when the markets are declining. This is particularly noteworthy given that our trading system is designed to capitalize on upward trends.

Additionally, it is worth noting the size of the bars in our graph on a 15-minute time frame, which are larger for our trading system. This indicates that our Sharpe ratio is not only superior in relative terms but also better in absolute value. This demonstrates that, besides generating positive returns when the benchmark is negative, the volatility associated with each unit of return is lower for our trading system.

Strategy Equity curve VS benchmark:

We have chosen to create equity curve charts to visualize the behavior of our trading system compared to the benchmark. We present the equity curves of our trading system applied on the 15-minute time frame against the benchmark for three assets: Bitcoin, Solana, and XRP. We selected Bitcoin as it is the leading cryptocurrency in the market, and we chose Solana and XRP to demonstrate our previously developed hypothesis. This hypothesis suggests that the stronger the benchmark trend, the more effective the trading system will be on a shorter time frame.

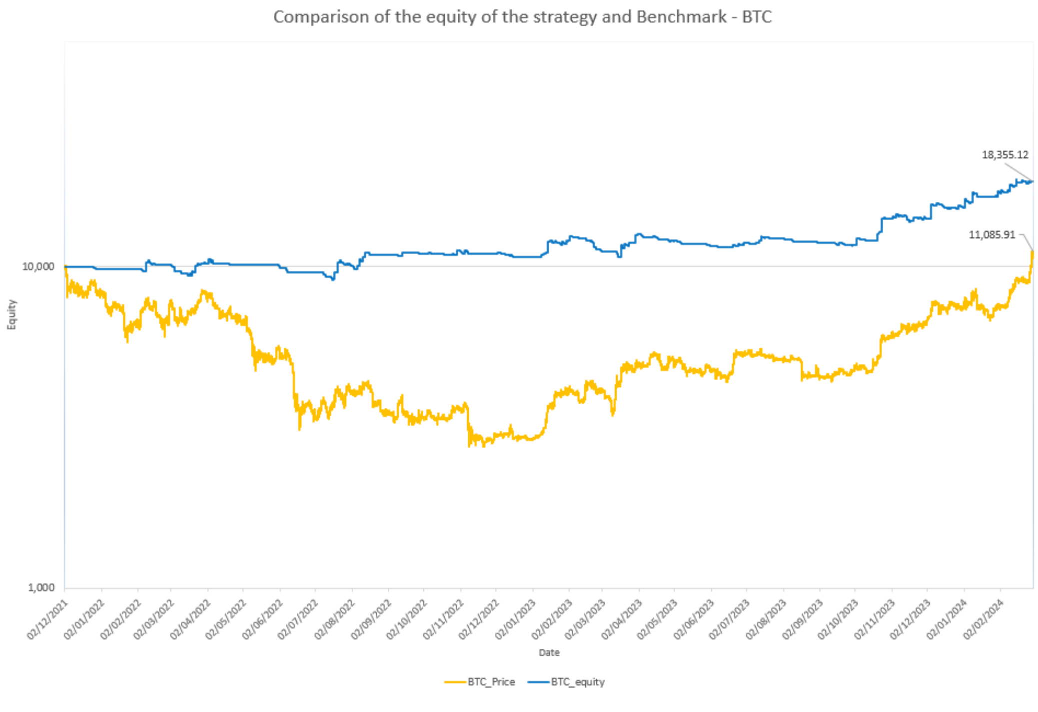


Figure 27: Bitcoin Equity curve strategy VS benchmark in 15m time unit

The profit curve for Bitcoin is particularly interesting to analyze, as it allows us to observe the behavior of our trading system under two different market conditions. The first half of the data, from 01/12/2021 to 01/01/2023, shows a bearish phase for the benchmark, Bitcoin, represented in yellow on the graph. The second half of the data, from 01/01/2023 to 01/02/2024, presents a bullish trend.

We can see that our trading system is very resilient during the bearish trend phase, maintaining a very stable profit curve, which reflects the limited opportunities for our "long only" strategy. In the bullish trend phase, the trading system capitalizes on the emerging upward trend, resulting in excellent performance with more executed trades. This is evidenced by the decrease in the number of "plateau" phases on the profit curve, indicating more dynamic and profitable trading activity during the bullish period.

Ultimately, we can note our trading system's ability to maintain itself when market conditions are unfavorable and present no interesting opportunities. The system effectively accomplishes its mission of riding bullish impulses when the benchmark trend is upward. Thus, our trading system has the capability to truly activate only when a bullish trend emerges, while remaining stable during less favorable conditions. This makes it a robust trading system across different market environments.

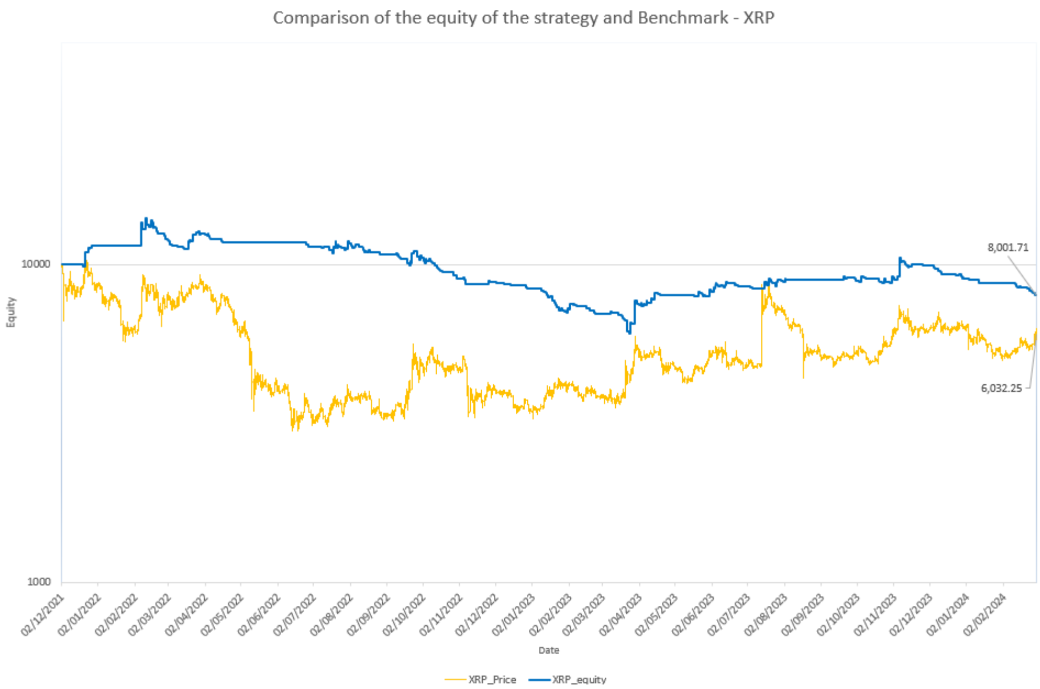


Figure 28: XRP Equity curve strategy VS benchmark in 15m time unit

Regarding the profit curve for XRP, we can corroborate our analysis by noting that, once again, the trading system maintains a very stable profit curve despite the strong bearish trend of XRP from 12/01/2021 to 07/01/2022. This demonstrates the system's ability to withstand unfavorable market conditions with no interesting opportunities.

On the other hand, the trend for XRP from 07/01/2022 to 2024 is bullish, but our trading system fails to be profitable, showing a very irregular profit curve. We believe, as explained earlier, that because the bullish trend for XRP is relatively weak, the trading system applied to a short 15-minute time frame does not perform well due to a lack of clarity in the signals on such a short time frame.

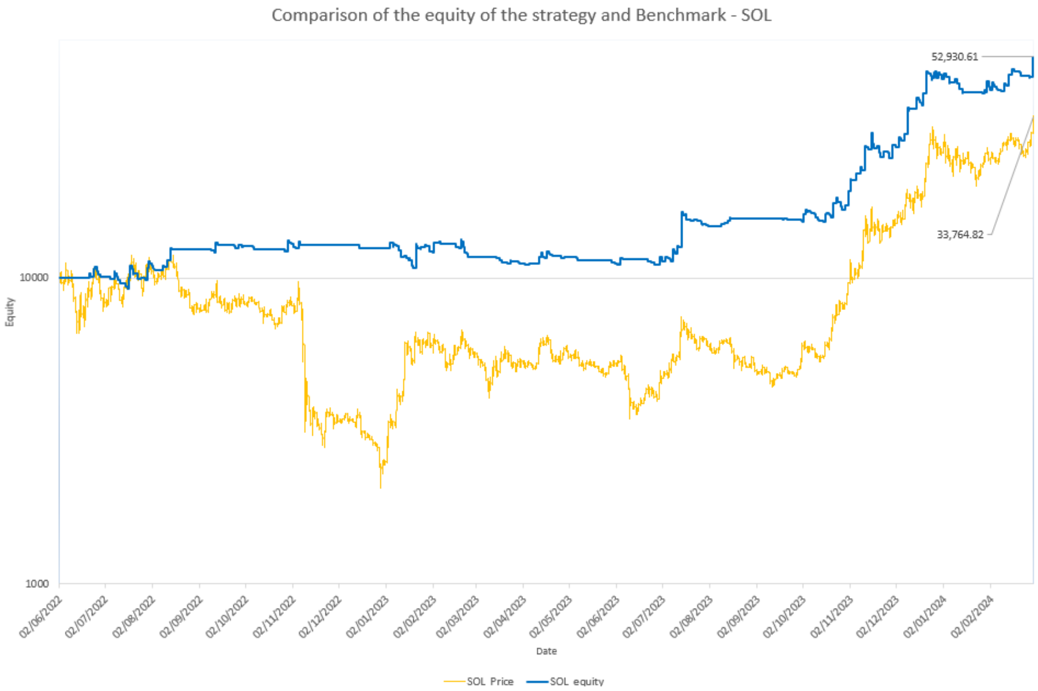


Figure 29: SOLANA Equity curve strategy VS benchmark in 15m time unit

Analyzing the profit curve structure for Solana, we can again observe the resilience of our trading system during the sharp decline of Solana from 11/01/2022 to 01/01/2023, where the price dropped by over 50% in just two months. Despite this dramatic drop, our trading system maintains impressive stability in its profit curve.

Moreover, we see that as soon as a strong and clear trend begins on 10/01/2023, the trading system successfully capitalizes on it, with its profit curve running parallel to Solana's during this phase. This demonstrates the system's ability to effectively align with strong upward trends, reinforcing its robustness and adaptability to varying market conditions.

Ultimately, we can conclude that the stronger the emerging bullish trend, the more relevant it becomes to apply the trading system to a shorter time frame, such as 15 minutes. Conversely, when the trend is relatively weak, as observed with XRP, it is more appropriate to apply the trading system to a higher time frame, such as 1 hour. This insight highlights the importance of selecting the right time frame based on the strength of the market trend to maximize the system's effectiveness.

5.6 Summary of our trading system metrics:

Finally, we compiled the average Sharpe ratio, maximum drawdown, and returns of our trading system applied to the 10 cryptocurrencies, using both the 1-hour and 15-minute time frames, and compared these metrics to the average benchmarks.

	Strategy	Benchmark
Average Sharpe Ratio	0.00081	-0.00147
Average Max Drawdown	-25%	-82%
Average Annualized Return	12%	-1%

Table 14: 1 Hour Frequency Strategy VS benchmark Statistics

	Strategy	Benchmark
Average Sharpe Ratio	0.00383	-0.00106
Average Max Drawdown	-26%	-84%
Average Annualized Return	42%	-5%

Table 15: 15 Minutes Frequency Strategy VS benchmark Statistics

The average Sharpe ratio of our trading system significantly outperformed the benchmark in both time frames. The 15-minute time frame showed a particularly strong Sharpe ratio, indicating that the system generates higher risk-adjusted returns compared to the benchmark, especially when trends are strong and clear.

Our trading system consistently exhibited a lower maximum drawdown compared to the benchmark in both time frames. This indicates that our system is more effective at managing risk and limiting losses, providing greater stability during market downturns, and reinforcing the robustness of our approach.

The returns of our trading system, on average, outperformed the benchmark in both the 1-hour and 15-minute time frames. However, the returns were notably higher in the 15-minute time frame, aligning with our hypothesis that a shorter time frame is more effective for capturing strong bullish trends. This confirms that the system is better suited for shorter time frames when the market exhibits strong directional movements.

Below, we present a summary table that details the key metrics used to evaluate the performance of our trading system. This table allows for a more in-depth analysis of the system's effectiveness. The table is divided into two parts: one for the metrics associated with the 1-hour time frame and the other for the 15-minute time frame. The values in the table represent the average performance of our trading system applied to the 10 cryptocurrencies on which we conducted the backtests.

Frequency	15 minutes	1 hour
Total Period Strategy SR	3.04	0.20
Total Period Benchmark SR	-0.74	-0.54
Average Strategy SR	0.00383	0.00081
Average Benchmark SR	-0.00106	-0.00147
Average Annualized Strategy Return	41.51%	12.03%
Average Annualized Benchmark Return	-5.34%	-0.74%
Average Volatility Strategy	26.42%	15.44%
Average Volatility Benchmark	39.46%	38.79%
Average 95% ES Strategy	-0.89%	-1.95%
Average 95% ES Benchmark	-1.01%	-2.01%
Average Max Drawdown Strategy	-25.69%	-25.32%
Average Max Drawdown Benchmark	-83.92%	-81.94%
Average Alpha	39.81%	10.08%
Average Beta	24.68%	43.34%
Average Market Exposure	21.85%	18.13%
Total Number of Trades	4073	587
Average Number of Trades	407	59
Average Winning Rate (%)	32.49%	30.52%
Average expectancy	0.20	0.34
Average Profit Factor	1.28	1.28

Table 16: Recapitulative table of Strategy Performance Metrics V2 by frequency

Firstly, it is evident that our trading system significantly outperforms the benchmark in terms of returns across both time frames. This is achieved with considerably lower volatility, almost three times lower on the 1-hour time frame compared to the benchmark. Consequently, our Sharpe ratio surpasses that of the benchmark, with an impressive average Sharpe ratio of 3 when the system is applied to the 15-minute time frame, even under occasionally unfavorable market conditions for certain assets like XRP.

Furthermore, our trading system exhibits a relatively low beta, which is noteworthy given that it is designed to ride upward trends. A beta of 0.24 on the 15-minute time frame demonstrates that our system maintains a low correlation with the benchmark. This indicates that during unfavorable market conditions for our "long only" strategy—when the benchmark is declining—our system often remains in a holding pattern. This behavior is evident in the equity curves, where periods of market decline result in fewer trades and a more cautious approach.

Additionally, our system generates a substantial alpha, demonstrating its ability to outperform the benchmark independently of market movements. Specifically, we observe an alpha of nearly 40% when the system is applied to the 15-minute time frame, and 10% on the 1-hour time frame. This significant alpha indicates that our system not only capitalizes on market trends but also adds considerable value through its unique strategy and effective risk management.

On the other hand, it is noteworthy that our system is invested in the market approximately

20% of the time, compared to 100% of the time for the benchmark in a "Buy and Hold" strategy. This indicates that not only does our system outperform the benchmark in terms of returns, drawdowns, volatility, Sharpe ratio, and alpha, but it also achieves this outperformance while being invested only 20% of the time. This allows for the possibility of executing other investments with our capital during the remaining 80% of the time, potentially amplifying our overall performance.

In conclusion, we have achieved and significantly exceeded our objectives. We aimed to develop a trading system with a better risk-adjusted return than the benchmark in a robust and consistent manner (across 10 assets in different types of possible market conditions). Our results demonstrate that we have successfully met this goal.

6 Conclusion & discussion

Summary of Findings and Answer to Research Question

In this thesis, we have demonstrated that it is not only possible to outperform cryptocurrency market but also that technical analysis is a powerful tool for constructing a robust trading system. Our research provides compelling evidence that, when applied with rigor and objectivity, technical analysis can offer a significant competitive edge.

To answer our research question: yes, it is possible to outperform the cryptocurrency market in terms of risk-adjusted returns with a trading strategy based on technical analysis. Our results show that our strategy achieved a significantly higher Sharpe ratio and return than the benchmark, with a lower maximum drawdown and volatility.

Our findings challenge the Efficient Market Hypothesis (EMH), which posits that it is impossible to consistently achieve higher returns than the overall market because all available information is already reflected in asset prices. While the EMH suggests that any anomalies would be quickly corrected by the market, our research shows that systematic technical analysis can identify and exploit these inefficiencies, leading to sustained outperformance.

Our approach was twofold. First, we utilized a discretionary method based on personal experience, allowing us to develop a custom-tailored strategy that aligns with specific market conditions and nuances. This personalized approach enabled us to craft a strategy that fits well with our understanding and interpretation of market dynamics.

Second, our systematic approach involved developing a fully mechanical trading system based on clear entry and exit signals. This method minimized the potential for cognitive biases and ensured that our trading decisions were consistently objective and data-driven. By relying on a systematic framework, we established a replicable and quantifiable process that enhances the reliability and robustness of our trading strategy.

Furthermore, our rigorous statistical approach allowed us to optimize our system by adapting it to varying market conditions. Through empirical testing and validation, we fine-tuned our strategy to maximize its effectiveness across different market environments, thereby reinforcing our edge in financial markets.

In conclusion, the integration of both discretionary insights and systematic rigor has proven to be a potent combination. This hybrid approach not only underscores the versatility of technical analysis as a tool but also highlights the importance of a disciplined, statistical methodology in developing a sustainable and profitable trading system. Our research thus provides a comprehensive answer to our initial research question, affirming the potential of technical analysis to achieve consistent market outperformance through several types of market conditions.

The Relevance of Technical Analysis

Technical analysis (TA) is a valuable tool among the various quantifiable methods available for developing a trading system aimed at achieving a competitive edge in the markets. Just like fundamental analysis, TA requires rigorous and objective application, along with proper quantification and measurement to validate or invalidate trading system hypotheses. Unfortunately, TA is often misused, contributing to its poor reputation among the general public and even among finance professionals.

Based on our perspective, for any tool to be considered effective in developing a trading system, it must meet several criteria:

- **Measurable:** The tool must allow for empirical testing and the application of statistical rigor. For instance, backtesting a trading strategy on historical data to evaluate its performance metrics like returns, volatility, and Sharpe ratio.

- **Objective Application:** Its application must be as objective as possible to avoid cognitive biases that can distort measurements. This is why we emphasize the creation of systematic (mechanical) strategies, which use algorithms to automate trading decisions based on predefined rules.

- **Replicability:** The tool must produce results that are replicable to ensure their significance. This involves repeatedly testing the tool under various market conditions to confirm that the outcomes are consistent and reliable.

- **Reliable Data Sources:** The tool should rely on accurate and reliable data sources, making the analysis fact-based. Ensuring data quality from reputable providers and performing regular quality checks are crucial for maintaining the integrity of the analysis.

Therefore, we believe that technical analysis, when used with the same rigor as fundamental analysis, is an undeniably effective tool in a trader's toolkit. However, one must always remain aware that markets are inherently uncertain, and predicting the future with certainty is impossible. This necessitates the incorporation of probabilistic thinking to support the robustness of TA.

Similarly, fundamental analysis lacks value if not accompanied by statistical measures to verify the relevance of a strategy based on fundamental insights. Statistical rigor is ultimately the essential ingredient that must accompany all tools used in developing a trading system. Technical analysis is no exception. It must be quantified, objective, replicable and reliable to allow for the application of statistical rigor, thereby solidifying the system's foundation and enhancing its overall reliability.

Future Research Directions

In this part, we explore potential avenues for future research and improvements to our trading system. These suggestions aim to enhance the system's performance, robustness, and adaptability across various market conditions.

One potential improvement is the implementation of dynamic position sizing, which involves adjusting the size of positions based on the probability of success. This can be achieved by analyzing trade series to establish robust conditional probabilities or by adapting according to different configurations that yield varying results. The Kelly criterion suggests increasing allocation if a configuration shows a higher winning rate or risk-reward ratio. For instance, if data indicates a better performance on the 15-minute chart compared to the 1-hour chart, the allocation can be adjusted accordingly. Similarly, if a consecutive loss streak leads to a significantly lower winning rate, the allocation for subsequent trades can be reduced based on the Kelly criterion. Additionally, if the confluence of signals, such as support from the RSI, significantly increases the winning rate, a larger average allocation can be applied to these positions. The goal is to gather extensive data, categorize it, and optimize allocation to maximize system performance.

Another potential improvement involves the strategic use of leverage to achieve optimal position sizes according to the Kelly criterion and maximizing the return/Max Drawdown ratio. The current 2% risk per trade might be suboptimal, and leveraging positions could help in optimizing returns.

Extending the backtesting period to 10 years, especially for assets like Bitcoin, could provide insights into the strategy's long-term performance. Consistent outperformance over such a period would strengthen the argument for the strategy's robustness and reliability.

Given the better results observed on the 15-minute chart compared to the 1-hour chart, expanding the backtesting to 5-minute or 1-minute intervals could increase the number of trade occurrences and potentially improve the system's expected gain. This approach, however, requires significant computational power.

Optimizing parameters for each individual asset could yield more optimal and tailored results while ensuring robustness. This involves fine-tuning the system to adapt to the unique characteristics of different assets.

Expanding the system to include short positions could take advantage of bear markets, significantly improving performance relative to the benchmark. This would allow the profit curve to increase even when the benchmark's performance declines.

Creating a portfolio of algorithms with different trading systems could aggregate returns and reduce drawdowns, thereby multiplying the overall Sharpe ratio. This approach could boost capital performance while simultaneously mitigating risk.

Implementing take-profit levels could further improve results by increasing the number of occurrences, as trades would be shorter and more frequent. This should lead to a higher winning rate and reduced probability of consecutive losing trades, thereby minimizing drawdowns. The realized risk-reward ratio per trade might be smaller, but the variance in returns would decrease, leading to a better expected gain per trade. Judiciously placed take-profit levels could significantly enhance the Sharpe ratio.

By incorporating these potential improvements, the trading system's performance could be significantly enhanced, creating a substantial gap against the benchmark. These refinements include dynamic position sizing, strategic leverage usage, extended backtesting, higher frequency data analysis, asset-specific optimization, the inclusion of short positions, the creation of an algorithm portfolio, and the implementation of take-profit levels. Together, these enhancements have the potential to exponentially increase the trading system's performance and solidify its competitive edge.

7 References

References

[1] M. Akbıyık et al. "A New Approach to Technical Analysis of Oil Prices". In: *Turkish Journal of Mathematics & Computer Science* 15.1 (2023). This study compares the Fibonacci golden ratio with a new ratio (nickel) and examines their significance in analyzing oil prices across different time frames., pp. 145–156. DOI: 10.47000/tjmcs.1117784.

[2] Doron Avramov, Guy Kaplanski, and Avanidhar Subrahmanyam. "Moving Average Distance as a Predictor of Equity Returns". In: *Review of Financial Economics, Forthcoming* (2018). DOI: 10.2139/ssrn.3111334. URL: https://ssrn.com/abstract=3111334.

[3] David H. Bailey et al. "The Probability of Backtest Overfitting". In: *Journal of Computational Finance* (2015). Forthcoming. DOI: 10.2139/ssrn.2326253.

[4] Brad M. Barber and Terrance Odean. "The Internet and the Investor". In: *Journal of Economic Perspectives* 15.1 (2001), pp. 41–54.

[5] Nicholas Barberis and Richard Thaler. "A Survey of Behavioral Finance". In: *Handbook of the Economics of Finance*. 2003.

[6] M. Basanisi and R. Torresetti. "An Overview of Technical Analysis in Systematic Trading Strategies Returns and a Novel Systematic Strategy Yielding Positive Significant Returns". In: *Journal of Contemporary Research in Business, Economics and Finance* 5.1 (2023), pp. 12–24. DOI: 10.55214/jcrbef.v5i1.204.

[7] Marco Basanisi and Roberto Torresetti. "A Framework for Systematic Trading: Methodologies and Applications". In: *Journal of Financial Markets* (2023).

[8] Johan Bollen, Huina Mao, and Xiaojun Zeng. "Twitter mood predicts the stock market". In: *Journal of Computational Science* 2.1 (2011), pp. 1–8.

[9] W. Brock, J. Lakonishok, and B. LeBaron. "Simple Technical Trading Rules and the Stochastic Properties of Stock Returns". In: *The Journal of Finance* 47.5 (1992), pp. 1731–1764. DOI: 10.1111/j.1540-6261.1992.tb04681.x.

[10] T. N. Bulkowski. *Encyclopedia of Chart Patterns*. 2000. URL: http://ci.nii.ac.jp/ncid/BA51404510.

[11] R. Carver. *Systematic Trading: A Unique New Method for Designing Trading and Investing Systems*. Harriman House Limited, 2015.

[12] T. S. Chande. *Beyond Technical Analysis: How to Develop and Implement a Winning Trading System*. 1996. URL: http://ci.nii.ac.jp/ncid/BA2978631X.

[13] J. R. Dahlquist and R. J. Bauer. *Technical Analysis of Gaps: Identifying Profitable Gaps for Trading*. FT Press, 2012.

[14] E. F. Fama. "Efficient Capital Markets: A Review of Theory and Empirical Work". In: *The Journal of Finance* 25.2 (1970), p. 383. DOI: 10.2307/2325486.

[15] E. F. Fama and M. E. Blume. "Filter Rules and Stock-Market Trading". In: *The Journal of Business* 39.1 (1966), pp. 226–241. DOI: 10.1086/294849.

[16] Tobias Fischer and Christopher Krauss. "Deep learning with long short-term memory networks for financial market predictions". In: *European Journal of Operational Research* 270.2 (2018), pp. 654–669.

[17] CME Group. *System-Based vs. Discretionary Trading*. 2024.

[18] Man Institute. "Man vs. Machine: Comparing Discretionary and Systematic Hedge Fund Performance". In: *The Journal of Portfolio Management* (2017).

[19] M. C. Jensen. "Random Walks: Reality or Myth—Comment". In: *Financial Analysts Journal* 23.6 (1967), pp. 77–85. DOI: 10.2469/faj.v23.n6.77.

[20] M. C. Jensen and G. A. Benington. "Random Walks and Technical Theories: Some Additional Evidence". In: *The Journal of Finance* 25.2 (1970), pp. 469–482. DOI: 10.2307/2325498.

[21] Thami Kabbaj. *L'art du trading*. Editions Organisation, 2015. ISBN: 9782212558593.

[22] Daniel Kahneman and Amos Tversky. "Prospect Theory: An Analysis of Decision under Risk". In: *Econometrica* 47.2 (1979), pp. 263–291.

[23] Marcos López de Prado. *Advances in Financial Machine Learning*. Wiley, 2018. ISBN: 9781119482086.

[24] Marcos López de Prado and Marcos López de Prado. *The 7 Reasons Most Machine Learning Funds Fail*. Presentation Slides, Available at SSRN: https://ssrn.com/abstract=3031282. 2017. DOI: 10.2139/ssrn.3031282.

[25] Marcos López de Prado and Marcos López de Prado. *Advances in Financial Machine Learning: Lecture 1/10*. Seminar slides, Available at SSRN: https://ssrn.com/abstract=3270329. 2018. DOI: 10.2139/ssrn.3270329.

[26] Marcos López de Prado and Marcos López de Prado. "The 10 Reasons Most Machine Learning Funds Fail". In: *Journal of Portfolio Management* (2018). Forthcoming, Available at SSRN: https://ssrn.com/abstract=3104816. DOI: 10.2139/ssrn.3104816.

[27] Tsung-Hsun Lu, Yung-Ming Shiu, and Tsung-Chi Liu. "Profitable candlestick trading strategies—The evidence from a new perspective". In: *Review of Financial Economics* 21.2 (2012), pp. 63–68. ISSN: 1058-3300. DOI: https://doi.org/10.1016/j.rfe.2012.02.001. URL: https://www.sciencedirect.com/science/article/pii/S1058330012000092.

[28] L. P. Lukac and B. W. Brorsen. "A Comprehensive Test of Futures Market Disequilibrium". In: *Financial Review* 25.4 (1990), pp. 593–622. DOI: 10.1111/j.1540-6288.1990.tb01300.x.

[29] L. P. Lukac, B. W. Brorsen, and S. H. Irwin. "A Test of Futures Market Disequilibrium Using Twelve Different Technical Trading Systems". In: *Applied Economics* 20.5 (1988), pp. 623–639. DOI: 10.1080/00036848800000113.

[30] Matthew R. Lyle and Teri L. Yohn. "Can Investors Profit from the Post-Earnings-Announcement Drift?" In: *The Journal of Financial Economics* 124.1 (2017), pp. 85–101.

[31] *Machine Learning Financial Laboratory*. https://www.mlfinlab.com/. Accessed: 2024-07-16.

[32] L. Menkhoff. "The Use of Technical Analysis by Fund Managers: International Evidence". In: *Journal of Banking and Finance* 34.11 (2010), pp. 2573–2586. DOI: 10.1016/j.jbankfin.2010.04.014.

[33] S. Nison. *Japanese Candlestick Charting Techniques*. New York: New York Institute of Finance, 1991.

[34] C. Park and S. H. Irwin. *The Profitability of Technical Analysis: A Review*. Social Science Research Network. 2004. DOI: 10.2139/ssrn.603481.

[35] C. H. Park and S. H. Irwin. "What Do We Know About the Profitability of Technical Analysis?" In: *Journal of Economic Surveys* 21.4 (2007), pp. 786–826. DOI: 10.1111/j.1467-6419.2007.00519.x.

[36] Bernard Prats-Desclaux. *Stratégies de marchés: Futures, CFD, actions*. Eyrolles, 2016. ISBN: 9782212564396.

[37] M. J. Pring. *Technical Analysis Explained*. New York: McGraw-Hill, 2002.

[38] K. Rink. "The Predictive Ability of Technical Trading Rules: An Empirical Analysis of Developed and Emerging Equity Markets". In: *Financ Mark Portf Manag* 37 (2023), pp. 403–456. DOI: 10.1007/s11408-023-00433-2.

[39] Hersh Shefrin. *Beyond Greed and Fear: Understanding Behavioral Finance and the Psychology of Investing*. Harvard Business School Press, 2000.

[40] Robert J. Shiller. *Irrational Exuberance*. Princeton University Press, 2000.

[41] R. Ślepaczuk, G. Zakrzewski, and P. Sakowski. *Investment Strategies Beating the Market: What Can We Squeeze from the Market?* Social Science Research Network. 2012. DOI: 10.2139/ssrn.2508647.

[42] A. Sokolovsky et al. "Interpretable Trading Pattern Designed for Machine Learning Applications". In: *Machine Learning with Applications* 11 (2023), p. 100448. DOI: 10.1016/j.mlwa.2023.100448.

[43] S. Srivastava and R. Bhattacharyya. *Evaluating the Building Blocks of a Dynamically Adaptive Systematic Trading Strategy*. Social Science Research Network. 2018. DOI: 10.2139/ssrn.3144169.

[44] M. Svoboda. "Stochastic Model of Short-Term Prediction of Stock Prices and Its Profitability in the Czech Stock Market". In: *E+M. Ekonomie a Management* 19.2 (2016), pp. 188–200. DOI: 10.15240/tul/001/2016-2-013.

[45] Paul C. Tetlock. "Giving Content to Investor Sentiment: The Role of Media in the Stock Market". In: *The Journal of Finance* 62.3 (2007), pp. 1139–1168. DOI: 10.1111/j.1540-6261.2007.01232.x.

[46] Richard H. Thaler. "Anomalies: The January Effect". In: *Journal of Economic Perspectives* 1.1 (1987), pp. 197–201.

[47] J. C. Van Horne and G. G. Parker. "Technical Trading Rules: A Comment". In: *Financial Analysts Journal* 24.4 (1968), pp. 128–132. DOI: 10.2469/faj.v24.n4.128.

A Appendix

A.1 Detailed Data sample

A.1.1 Market data variables

- **Date** - The calendar date for the data entry.
- **Open Time** - The timestamp when the trading period begins.
- **Open** - The price of the asset at the beginning of the trading period.
- **High** - The highest price of the asset during the trading period.
- **Low** - The lowest price of the asset during the trading period.
- **Close** - The price of the asset at the end of the trading period.
- **Volume** - The total trading volume of the asset during the trading period.
- **Close Time** - The timestamp when the trading period ends.
- **Quote Volume** - The total volume of the asset traded in quote currency.
- **Count** - The number of trades that took place during the trading period.
- **Taker Buy Volume** - The volume of the asset bought by takers during the trading period.
- **Taker Buy Quote Volume** - The quote volume of the asset bought by takers during the trading period.

A.1.2 Assets

The analysis covers market data from the following 10 cryptocurrencies:

- **BTC (Bitcoin):** From 2019-09-01 00:00 to 2024-02-29 23:45
- **ADA (Cardano):** From 2019-09-01 00:00 to 2024-02-29 23:45
- **ETH (Ethereum):** From 2019-09-01 00:00 to 2024-02-29 23:45
- **SOL (Solana):** From 2020-08-11 06:00 to 2024-02-29 23:45
- **BNB (Binance Coin):** From 2019-09-01 00:00 to 2024-02-29 23:45
- **BCH (Bitcoin Cash):** From 2019-11-28 10:00 to 2024-02-29 23:45
- **XRP (Ripple):** From 2019-09-01 00:00 to 2024-02-29 23:45
- **LTC (Litecoin):** From 2019-09-01 00:00 to 2024-02-29 23:45
- **LINK (Chainlink):** From 2019-09-01 00:00 to 2024-02-29 23:45
- **DOT (Polkadot):** From 2020-08-18 23:00 to 2024-02-29 23:45

A.2 Code Structure

The code structure contains the functions used, their inputs and outputs, and their goals. The function used for a specific strategy has an asterisk (*) in contrast to the core functions.

The main purpose of this code is to backtest the strategies (`RunTradingSimulation`). It takes in input datasets and sorts the output metrics, trades, and plots for analysis. A number of variations are also applied to the strategy indicators' parameters to perceive the performance under different setups (`run_variations_and_save`). The outputs are transferred to an Excel workbook for analysis purposes.

As the core function `RunTradingSimulation` uses all the indicator functions (marked with an asterisk) to define entry/exit conditions, simulate trades, and compute performance metrics for any dataset in input.

Some functions only have a visualization goal (`plotTradingViewStyle`), but many plots have been implemented in the functions to improve visuality and ensure that the strategy behaves as expected.

A.2.1 Class Diagram

The following class diagram (Figure 30) provides a visual representation of the structure and relationships between the various classes used in the trading strategy simulation. Each class represents a distinct component of the system, outlining its responsibilities and interactions with other classes.

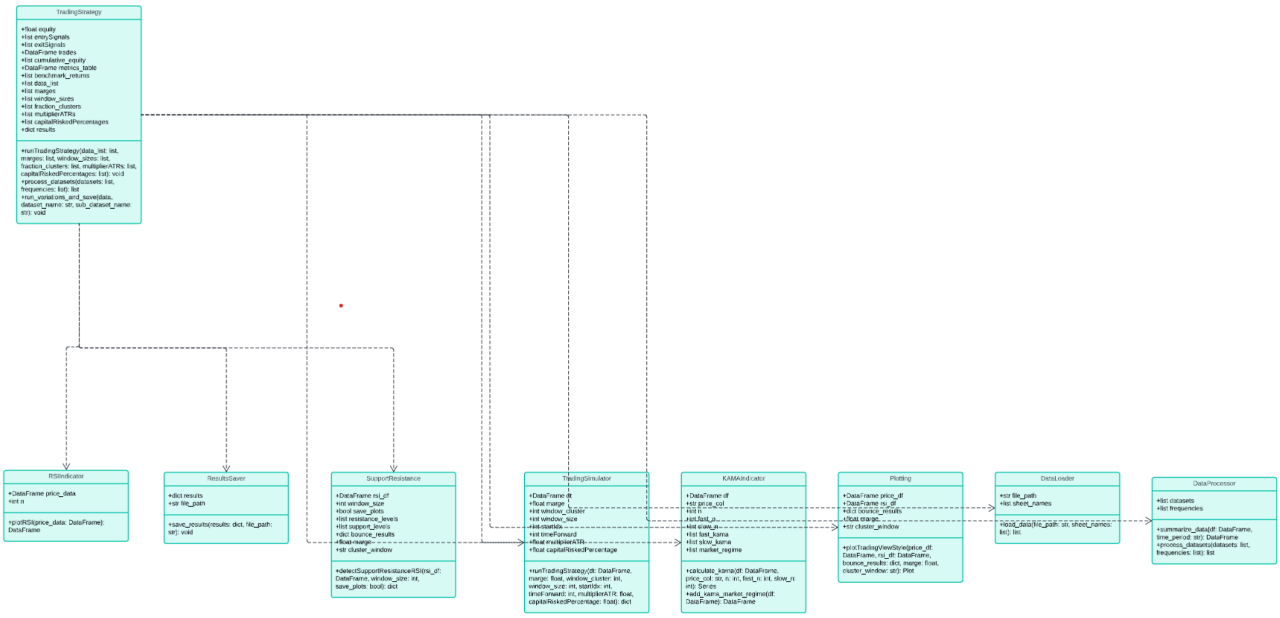


Figure 30: Class Diagram of Trading Strategy Simulation

A.2.2 Detailed process & functions

1. Package Loading and Installation

This section ensures that required packages are installed and loaded into the R session.

```r
if (!require("quantmod")) install.packages("quantmod")
library(quantmod)
if (!require("TTR")) install.packages("TTR")
library(TTR)
if (!require("readxl")) install.packages("readxl")
if (!require("dplyr")) install.packages("dplyr")
if (!require("lubridate")) install.packages("lubridate")
if (!require("writexl")) install.packages("writexl")
```

2. Data Loading

The data is composed of 10 of the most correlated assets of Bitcoin, our reference asset. The data selection is further explained in the “methodology and data” section.

```r
datasets <- list(
  BTC = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'BTC'),
  ADA = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'ADA'),
  ETH = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'ETH'),
  SOL = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'SOL'),
  BNB = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'BNB'),
  BCH = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'BCH'),
  XRP = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'XRP'),
  LTC = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'LTC'),
  LINK = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'LINK'),
  DOT = read_excel("~/Master LSM/Master2/Mémoire/Thesis_data.xlsx", sheet = 'DOT')
)
```

3. Data Summarization

This function aggregates the data into specified time periods and ensures the test of different frequencies.

Function Name: `summarize_data`
Inputs:

- `df`: A data frame with columns Date, open, high, low, close, volume, quote_volume, count, taker_buy_volume, taker_buy_quote_volume.
- `time_period`: A string specifying the time period for aggregation (e.g., "15 minutes", "1 hour").

Outputs:

- A summarized data frame with aggregated OHLCV (Open, High, Low, Close, Volume) and other metrics for each time period.

```r
summarize_data <- function(df, time_period) {
  df %>%
    mutate(time = floor_date(Date, time_period)) %>%
    group_by(time) %>%
    summarise(
```

```
      open = first(open),
      high = max(high),
      low = min(low),
      close = last(close),
      volume = sum(`volume`),
      quote_volume = sum(`quote_volume`),
      count = mean(`count`),
      taker_buy_volume = mean(`taker_buy_volume`),
      taker_buy_quote_volume = mean(`taker_buy_quote_volume`)
    )
}
```

4. Dataset Processing

This function applies the `summarize_data` function to each dataset for various time frequencies in view of the backtest.

Function Name: `process_datasets`
Inputs:

- `datasets`: A list of data frames.
- `frequencies`: A vector of strings specifying different time frequencies.

Outputs:

- A list of summarized data frames for each frequency.

```
process_datasets <- function(datasets, frequencies) {
  results <- list()
  for (freq in frequencies) {
    results[[freq]] <- list()
    for (name in names(datasets)) {
      df <- datasets[[name]]
      freq_data <- summarize_data(df, freq)
      results[[freq]][[name]] <- freq_data
    }
  }
  return(results)
}
```

5. RSI Plotting*

This function calculates and plots the Relative Strength Index (RSI).

Function Name: `plotRSI`
Inputs:

- `price_data`: A data frame with close prices and time.

Outputs:

- A plot for visualization and a data frame of the RSI.

```
plotRSI <- function(price_data) {
  if (!inherits(price_data$time, "POSIXct")) {
    price_data$time <- as.POSIXct(price_data$time, format="%Y-%m-%d %H:%M:%S", tz="UTC")
  }
  price_xts <- xts(price_data$close, order.by=price_data$time)
  rsi_data <- RSI(price_xts, n=21)
  rsi_df <- data.frame(date = index(rsi_data), RSI = coredata(rsi_data))
  ggplot(rsi_df, aes(x = date, y = RSI)) +
    geom_line() +
    geom_hline(yintercept = 70, linetype = "dashed", color = "red") +
    geom_hline(yintercept = 30, linetype = "dashed", color = "blue") +
    labs(title = "RSI Plot", x = "Date", y = "RSI") +
    theme_minimal()
  return(rsi_df)
}
```

6. Support and Resistance Detection*

This function detects support and resistance levels based on RSI values. It is a function depending on the strategy used.

Function Name: `detectSupportResistanceRSI`
Inputs:

- `rsi_df`: A data frame with date and RSI columns.
- `window_size`: The size of the window for rolling calculations.
- `save_plots`: A boolean indicating whether to save the plots.

Outputs:

- A list of support and resistance levels.

```
detectSupportResistanceRSI <- function(rsi_df, window_size, save_plots = FALSE) {
  results <- list()
  for (rsi_column in names(rsi_df)[-1]) {
    rsi_data <- na.omit(rsi_df[, c("date", rsi_column)])
    rsi_max <- rollapply(rsi_data[[rsi_column]], window_size, max, fill = NA, align =
    ↪  "right")
    rsi_min <- rollapply(rsi_data[[rsi_column]], window_size, min, fill = NA, align =
    ↪  "right")
    max_points <- data.frame(date = rsi_data$date, rsi = rsi_max, type = "Resistance")
    min_points <- data.frame(date = rsi_data$date, rsi = rsi_min, type = "Support")
    combined <- rbind(max_points, min_points) %>% filter(!is.na(rsi))
    combined <- combined %>% distinct(rsi, .keep_all = TRUE)
    d <- dist(combined$rsi)
    fit <- hclust(d, method = "ward.D2")
    k <- ceiling(sqrt(nrow(combined)))
    clusters <- cutree(fit, k = k)
    combined$cluster <- clusters
    significant_levels <- combined %>%
      group_by(cluster, type) %>%
      summarize(rsi = mean(rsi), .groups = 'drop') %>%
```

```
      arrange(desc(rsi))
    resistance_levels <- significant_levels %>% filter(type == "Resistance")
    support_levels <- significant_levels %>% filter(type == "Support")
    results[[rsi_column]] <- list(resistance = resistance_levels$rsi, support =
    ↪  support_levels$rsi)
    plot_object <- ggplot(rsi_data, aes(x = date, y = .data[[rsi_column]])) +
      geom_line() +
      geom_hline(data = resistance_levels, aes(yintercept = rsi), color = "red", linetype =
      ↪  "dashed") +
      geom_hline(data = support_levels, aes(yintercept = rsi), color = "blue", linetype =
      ↪  "dashed") +
      labs(title = paste("RSI Support and Resistance for", rsi_column), x = "Date", y =
      ↪  "RSI") +
      theme_minimal()
    if (save_plots) {
      ggsave(filename = paste0("RSI_Support_Resistance_", rsi_column, ".png"), plot =
      ↪  plot_object, width = 10, height = 6)
    }
  }
  return(results)
}
```

7. Trading View Style Plot

This function plots price data and RSI with detected support and resistance levels.

Function Name: `plotTradingViewStyle`
Inputs:

- `price_df`: Data frame with time and close prices.
- `rsi_df`: Data frame with date and RSI values.
- `bounce_results`: Support and resistance levels detected by `detectSupportResistanceRSI`.
- `marge`: Margin value for plotting.
- `cluster_window`: Window size for clustering.

Outputs:

- A list containing ggplot objects for the plots (only for visualization).

```
plotTradingViewStyle <- function(price_df, rsi_df, bounce_results, marge, cluster_window =
↪  "2 days") {
  plots <- list()
  all_crossings_detailed <- list()
  price_data <- data.frame(date = price_df$time, price = price_df$close)
  rsi_data <- data.frame(date = rsi_df$date, rsi = rsi_df$rsi)
  resistance_levels <- sort(unlist(bounce_results$rsi$resistance), decreasing = TRUE)
  support_levels <- sort(unlist(bounce_results$rsi$support), decreasing = FALSE)
  cross_data <- tidyr::crossing(date = rsi_data$date, levels = c(resistance_levels,
  ↪  support_levels))
  cross_data <- left_join(cross_data, rsi_data, by = "date")
```

```
cross_data <- cross_data %>%
  mutate(is_cross = abs(rsi - levels) < marge) %>%
  filter(is_cross)
cross_data <- merge(cross_data, price_data, by = "date")
detailed_crossings <- data.frame(Date = rsi_data$date)
detailed_crossings[paste("Resistance", seq_along(resistance_levels))] <- FALSE
detailed_crossings[paste("Support", seq_along(support_levels))] <- FALSE
for (lvl in seq_along(resistance_levels)) {
  crosses <- cross_data[cross_data$levels == resistance_levels[lvl] & cross_data$is_cross,
  ↪  "date"]
  if (length(crosses) > 1) {
    clusters <- cut(crosses, breaks = "2 days", labels = FALSE)
    significant_crosses <- tapply(crosses, clusters, function(x)
    ↪  x[which.max(rsi_data$rsi[rsi_data$date %in% x])])
    detailed_crossings[detailed_crossings$Date %in% significant_crosses,
    ↪  paste("Resistance", lvl)] <- TRUE
  } else {
    detailed_crossings[detailed_crossings$Date %in% crosses, paste("Resistance", lvl)] <-
    ↪  TRUE
  }
}
for (lvl in seq_along(support_levels)) {
  crosses <- cross_data[cross_data$levels == support_levels[lvl] & cross_data$is_cross,
  ↪  "date"]
  if (length(crosses) > 1) {
    clusters <- cut(crosses, breaks = "2 days", labels = FALSE)
    significant_crosses <- tapply(crosses, clusters, function(x)
    ↪  x[which.min(rsi_data$rsi[rsi_data$date %in% x])])
    detailed_crossings[detailed_crossings$Date %in% significant_crosses, paste("Support",
    ↪  lvl)] <- TRUE
  } else {
    detailed_crossings[detailed_crossings$Date %in% crosses, paste("Support", lvl)] <-
    ↪  TRUE
  }
}
all_crossings_detailed <- detailed_crossings
p1 <- ggplot(price_data, aes(x = date, y = price)) +
  geom_line() +
  geom_point(data = cross_data[cross_data$is_cross & cross_data$levels %in%
  ↪  resistance_levels, ],
             aes(x = date, y = price), color = "red", size = 2) +
  geom_point(data = cross_data[cross_data$is_cross & cross_data$levels %in%
  ↪  support_levels, ],
             aes(x = date, y = price), color = "blue", size = 2) +
  labs(title = paste("Price and RSI Analysis"), y = "Price", x = NULL) +
  theme_minimal()
p2 <- ggplot(rsi_data, aes(x = date, y = rsi)) +
  geom_line(color = "blue") +
  geom_hline(data = data.frame(rsi = resistance_levels), aes(yintercept = rsi), color =
  ↪  "red", linetype = "dashed") +
  geom_hline(data = data.frame(rsi = support_levels), aes(yintercept = rsi), color =
  ↪  "blue", linetype = "dashed") +
  geom_point(data = cross_data[cross_data$is_cross & cross_data$levels %in%
  ↪  resistance_levels, ],
             aes(x = date, y = rsi), color = "red", size = 2) +
  geom_point(data = cross_data[cross_data$is_cross & cross_data$levels %in%
  ↪  support_levels, ],
```

```
                aes(x = date, y = rsi), color = "blue", size = 2) +
    labs(title = "RSI with Support and Resistance", y = "RSI", x = NULL) +
    theme_minimal()
  combined_plot <- p1 / p2
  plots <- combined_plot
  return(list(plots = plots, detailed_crossings = all_crossings_detailed))
}
```

8. Process and Cluster Crossings*

This function processes and clusters RSI crossings to determine significant support and resistance levels.

Function Name: `processAndClusterCrossings`
Inputs:

- `detailed_crossings`: Data frame with detailed crossings information.
- `height`: Clustering height parameter.

Outputs:

- A data frame with clustered crossings labeled.

```
processAndClusterCrossings <- function(detailed_crossings, height = 0.1) {
  aggregated_results <- list()
  crossings <- detailed_crossings
  crossings$Date <- as.POSIXct(crossings$Date)
  numeric_crossings <- data.frame(Date = crossings$Date)
  crossing_labels <- names(crossings)[-1]
  for (col in crossing_labels) {
    numeric_crossings[[col]] <- as.numeric(crossings[[col]])
  }
  transposed_crossings <- t(numeric_crossings[-1])
  dissimilarity_matrix <- dist(transposed_crossings)
  hc <- hclust(dissimilarity_matrix, method = "ward.D2")
  clusters <- cutree(hc, h = height)
  clustered_data <- data.frame(Date = crossings$Date)
  unique_clusters <- sort(unique(clusters))
  cluster_labels <- sapply(unique_clusters, function(cluster_id) {
    labels_in_cluster <- crossing_labels[clusters == cluster_id]
    labels_in_cluster[1]
  }, USE.NAMES = FALSE)
  for (i in seq_along(unique_clusters)) {
    cluster_id <- unique_clusters[i]
    label <- cluster_labels[i]
    cluster_columns <- crossing_labels[clusters == cluster_id]
    clustered_data[[label]] <- apply(numeric_crossings[cluster_columns], 1, max)
  }
  aggregated_results <- clustered_data
  return(aggregated_results)
}
\end{verbatim}
```

```
\subsubsection*{9. KAMA Calculation*}
This function calculates the Kaufman Adaptive Moving Average (KAMA) for market regime
↪ analysis. It is also based on the specific strategy used.\

\textbf{Function Name:} \texttt{calculate\_kama}

\textbf{Inputs:}
\begin{itemize}
\item \texttt{df}: Data frame with price data.
\item \texttt{price\_col}: Column name of the price data.
\item \texttt{n}: Base period for KAMA calculation.
\item \texttt{fast\_n}: Fast period for KAMA calculation.
\item \texttt{slow\_n}: Slow period for KAMA calculation.
\end{itemize}

\textbf{Outputs:}
\begin{itemize}
\item A vector of KAMA values.
\end{itemize}
\begin{verbatim}
calculate_kama <- function(df, price_col, n, fast_n, slow_n) {
  kama <- rep(NA, nrow(df))
  change <- abs(df[[price_col]] - lag(df[[price_col]], n))
  volatility <- rollapplyr(df[[price_col]], width = n, FUN = function(x) sum(abs(diff(x))),
  ↪ fill = NA, align = 'right')
  ER <- change / volatility
  ER[is.na(ER)] <- 0
  fastSC <- 2 / (fast_n + 1)
  slowSC <- 2 / (slow_n + 1)
  SC <- ((ER * (fastSC - slowSC)) + slowSC) ^ 2
  for (i in (n + 1):nrow(df)) {
    if (!is.na(kama[i-1])) {
      kama[i] <- kama[i - 1] + SC[i] * (df[[price_col]][i] - kama[i - 1])
    } else {
      kama[i] <- mean(df[[price_col]][1:n], na.rm = TRUE)
    }
  }
  return(kama)
}
```

10. Add KAMA Market Regime*

This function adds market regime information based on KAMA to the data frame.

Function Name: `add_kama_market_regime`
Inputs:

- `df`: Data frame with price data.

Outputs:

- Data frame with KAMA and market regime columns added.

11. Trading Strategy

This function runs a trading strategy based on specific indicators, simulating trades, and calculating performance metrics. The inputs can change from a specific strategy to another as it is for all apart from `dt`, indicator parameters.

Function Name: `runTradingStrategy`
Inputs:

- `dt`: Data frame with price data.
- `marge`: Margin for detecting support levels.
- `window_cluster`: Window size for clustering support levels.
- `window_size`: Window size for support calculation.
- `startIdx`: Start index for the strategy.
- `timeForward`: Time forward for recalculating support.
- `multiplierATR`: Multiplier for ATR-based stop-loss calculation.

Outputs:

- A list with performance metrics, trades data, and combined data for plotting.

```
runTradingStrategy <- function(dt, marge, window_cluster, window_size, startIdx,
↪  timeForward, multiplierATR) {
  library(quantmod)
  Date <- as.POSIXct(dt$time, format="%Y-%m-%d %H:%M:%S", tz="UTC")
  rsi_df <- plotRSI(dt)
  support_values <- rep(NA, nrow(dt))
  levels_results <- detectSupportResistanceRSI(rsi_df[1:startIdx,], window_size =
  ↪  window_cluster)
  if (length(levels_results$rsi$support) == 1) {
    current_support <- levels_results$rsi$support[1]
  } else {
    current_support <- levels_results$rsi$support[length(levels_results$rsi$support)]
  }
  support_values[1:startIdx] <- current_support
  for (t in (startIdx + 1):nrow(dt)) {
    if (!is.na(current_support) && rsi_df[t-1, "rsi"] <= current_support - marge) {
      next_calculation <- t + timeForward
      support_values[t:min(next_calculation - 1, nrow(dt))] <- 0
      t <- next_calculation
      if (next_calculation < nrow(dt)) {
        new_start_idx <- max(1, next_calculation - window_size)
        new_end_idx <- next_calculation - 1
        levels_results <- detectSupportResistanceRSI(rsi_df[new_start_idx:new_end_idx,],
        ↪  window_size = window_cluster)
        if (length(levels_results$rsi$support) == 1) {
          current_support <- levels_results$rsi$support[1]
        } else {
          current_support <-
          ↪  levels_results$rsi$support[length(levels_results$rsi$support)-1]
        }
      }
    } else {
      if (is.na(support_values[t])) {
```

```
      support_values[t] <- current_support
    }
  }
}
if (!is.na(current_support) && max(which(!is.na(support_values))) < nrow(dt)) {
  support_values[(max(which(!is.na(support_values))) + 1):nrow(dt)] <- current_support
}
support <- support_values
crossings_down <- data.frame(Time = integer(), RSI_Value = numeric(), Support_Value =
↪  numeric(), stringsAsFactors = FALSE)
plot_data <- data.frame(Time = 1:nrow(dt), Close = dt$close, Support = support_values, RSI
↪  = rsi_df$rsi)
for (i in 2:nrow(plot_data)) {
  current_rsi <- plot_data$RSI[i]
  current_support <- plot_data$Support[i]
  previous_rsi <- plot_data$RSI[i-1]
  previous_support <- plot_data$Support[i-1]
  if (!is.na(previous_rsi) && !is.na(current_rsi) && !is.na(previous_support) &&
  ↪  !is.na(current_support)) {
    if (previous_rsi > previous_support && current_rsi-marge < current_support) {
      crossings_down <- rbind(crossings_down, data.frame(Time = plot_data$Time[i],
      ↪  RSI_Value = current_rsi, Support_Value = current_support))
    }
  }
}
crossings_down <- crossings_down[order(crossings_down$Time), ]
row.names(crossings_down) <- NULL
crossing_down_flags <- data.frame(
  Time = startIdx:NROW(dt),
  CrossingDown = integer(NROW(dt) - startIdx + 1)
)
crossing_down_flags$CrossingDown <- 0
for (i in (startIdx + 1):NROW(dt)) {
  current_rsi <- plot_data$RSI[i]
  current_support <- plot_data$Support[i]
  previous_rsi <- plot_data$RSI[i - 1]
  previous_support <- plot_data$Support[i - 1]
  if (!is.na(previous_rsi) && !is.na(current_rsi) && !is.na(previous_support) &&
  ↪  !is.na(current_support)) {
    if (previous_rsi > previous_support && current_rsi - marge < current_support) {
      crossing_down_flags$CrossingDown[i - startIdx + 1] <- 1
    }
  }
}
dt <- add_kama_market_regime(dt)
MR <- dt$market_regime
library(TTR)
atr_values <- ATR(HLC = dt[, c("high", "low", "close")], n = 21, maType = "EMA")
dt$ATR_21 <- atr_values[,"atr"]
entrySignals <- rep(FALSE, length(dt))
exitSignals <- rep(FALSE, length(dt))
equity <- dt$close[startIdx]
inPosition <- FALSE
positionSize <- numeric()
entryPrice <- numeric()
equity_values <- numeric()
ATR <- numeric()
```

```
ATRprice <- numeric()
TC <- 0.0001
tradeProfit <- numeric()
entryConditions <- NA
exitConditions <- NA
trades <- data.frame(EntryDate = as.POSIXct(character()),
                     ExitDate = as.POSIXct(character()),
                     EntryPrice = numeric(),
                     ExitPrice = numeric(),
                     ATR_Price = numeric(),
                     StopLoss = numeric(),
                     AmountTraded = numeric(),
                     TransactionCost = numeric(),
                     Profit = numeric(),
                     Return = numeric(),
                     stringsAsFactors = FALSE)
for (t in startIdx + 1:NROW(dt)) {
  entryConditions <- (crossing_down_flags[t-1-startIdx,2] == 1 && MR[t-1] == 1)
  exitConditions <- (support[t-1] == 0)
  entrySignals[t] <- entryConditions
  exitSignals[t] <- exitConditions
}
entryDate <- NA
capitalRiskedPercentage <- 0.02
drawdowns <- rep(0, NROW(dt))
for (t in startIdx:NROW(dt)) {
  if (!is.na(entrySignals[t]) && !inPosition && entrySignals[t]) {
    inPosition <- TRUE
    entryPrice <- dt$close[t]
    entryDate <- Date[t]
    entryIndex <- which(dt$time == entryDate)
    ATR <- dt$ATR_21[entryIndex]
    ATRprice <- entryPrice - (multiplierATR * ATR)
    Implied_SL <- -((ATRprice - entryPrice) / entryPrice)
    equitytotrade <- equity
    equityRisked <- equity * capitalRiskedPercentage
    if (equityRisked / Implied_SL >= equity) {
      equityTrade <- equitytotrade
    } else {
      equityTrade <- equityRisked / Implied_SL
    }
    ratio <- equityTrade / equity
    positionSize <- equityTrade / entryPrice
  } else if (inPosition) {
    currentPrice <- dt$close[t]
    if (inPosition && exitSignals[t] || inPosition && currentPrice <= ATRprice) {
      exitPrice <- currentPrice
      exitIndex <- which(dt$time == Date[t])
      tradeProfit <- (exitPrice - entryPrice) * positionSize - TC * entryPrice - TC *
      ↪  exitPrice
      if (tradeProfit < 0) {
        drawdowns[t] <- tradeProfit / equity
      }
      equity <- equity + tradeProfit
      return <- (tradeProfit / (equityTrade))
      TCs <- (-TC * entryPrice - TC * exitPrice)
      trades <- rbind(trades, data.frame(EntryDate = entryDate,
```

```
                                             ExitDate = Date[t],
                                             EntryPrice = entryPrice,
                                             ExitPrice = exitPrice,
                                             ATR_Price = ATRprice,
                                             StopLoss = Implied_SL,
                                             AmountTraded = equityTrade,
                                             TransactionCost = TCs,
                                             Profit = tradeProfit,
                                             Return = return))
      inPosition <- FALSE
      positionSize <- 0
    }
  }
  equity_values[t] <- equity
}
combined_data <- data.frame(
  Date = Date,
  Stock_Price = as.numeric(dt$close),
  ATR = as.numeric(dt$ATR_21),
  RSI = as.numeric(rsi_df$rsi),
  Support = as.numeric(support),
  equity = equity_values
)
combined_data_filtered <- combined_data[combined_data$Date >= Date[startIdx], ]
combined_data_filtered$crossing <- as.numeric(crossing_down_flags$CrossingDown)
combined_data_filtered$SignalType <- ifelse(!is.na(entrySignals[(startIdx):NROW(dt)]) &
↪  entrySignals[startIdx:NROW(dt)], 'Entry',
                                            ifelse(!is.na(exitSignals[startIdx:NROW(dt)])
                                            ↪  & exitSignals[startIdx:NROW(dt)], 'Exit',
                                            ↪  'None'))
combined_data_filtered$Date <- as.POSIXct(combined_data_filtered$Date, format="%Y-%m-%d
↪  %H:%M:%S")
entry_dates <- as.character(trades$EntryDate)
exit_dates <- as.character(trades$ExitDate)
combined_data_filtered$ActualSignalType <-
↪  ifelse(as.character(combined_data_filtered$Date) %in% entry_dates, 'Entry',

                                                  ↪  ifelse(as.character(combined_data_filtered$Date
                                                  ↪  %in% exit_dates, 'Exit', NA))
SignalsPlot <- ggplot(combined_data_filtered, aes(x = Date, y = Stock_Price)) +
  geom_line(color = "blue", linewidth = 1) +
  geom_point(data = subset(combined_data_filtered, SignalType == 'Entry'), aes(color =
  ↪  SignalType), size = 2) +
  geom_point(data = subset(combined_data_filtered, SignalType == 'Exit'), aes(color =
  ↪  SignalType), size = 2) +
  scale_color_manual(values = c(Entry = 'green', Exit = 'red')) +
  theme_minimal() +
  labs(title = "Stock Price with Entry and Exit Signals", y = "Stock Price", x = "") +
  theme(legend.title = element_blank())
SignalsTradedPlot <- ggplot() +
  geom_line(data = combined_data_filtered, aes(x = Date, y = Stock_Price), color = "blue",
  ↪  linewidth = 1) +
  geom_point(data = subset(combined_data_filtered, ActualSignalType == 'Entry'), aes(x =
  ↪  Date, y = Stock_Price, color = ActualSignalType), size = 3, shape = 17) +
  geom_point(data = subset(combined_data_filtered, ActualSignalType == 'Exit'), aes(x =
  ↪  Date, y = Stock_Price, color = ActualSignalType), size = 3, shape = 18) +
  scale_color_manual(values = c(Entry = 'green', Exit = 'red')) +
```

```
    theme_minimal() +
    labs(title = "Stock Price with Traded Entry and Exit Signals", y = "Stock Price", x =
     ↪  "Stock") +
    theme(legend.title = element_blank())
  equityPlot <- ggplot(combined_data_filtered, aes(x = Date, y = equity)) +
    geom_line(color = "darkgreen", linewidth = 1) +
    theme_minimal() +
    labs(title = "Equity Curve Over Time", y = "Equity", x = "stock") +
    theme(legend.title = element_blank())
  tot_benchmark_returns <- ((dt$close[NROW(dt)] - dt$close[startIdx + 1]) /
  ↪  dt$close[startIdx + 1])
  benchmark_equity <- dt$close[(startIdx):NROW(dt)]
  benchmark_returns <- diff(log((dt$close[(startIdx):NROW(dt)])))
  annual_rf_rate <- 0.04
  granularity <- as.numeric(difftime(dt$time[2], dt$time[1], units = "mins"))
  total_time_available_year <- (nrow(combined_data_filtered) * granularity) / (60 * 24 *
  ↪  365.25)
  cumulative_equity <- combined_data_filtered$equity
  initial_equity <- cumulative_equity[1]
  final_equity <- tail(cumulative_equity, 1)
  strategy_total_return <- ((final_equity - initial_equity) / initial_equity)
  intervals_per_year <- 365.25 * 24 * 60 / granularity
  periodic_rf_rate <- exp(annual_rf_rate / intervals_per_year) - 1
  cumulative_return <- diff(log(cumulative_equity))
  sharpe_ratio_strat <- round(x = (mean(cumulative_return) - periodic_rf_rate) /
  ↪  sd(cumulative_return), digits = 4)
  adj_sharpe_ratio_strat <- round(x = mean(cumulative_return) / sd(cumulative_return),
  ↪  digits = 4)
  sharpe_ratio_benchmark <- round(x = (mean(benchmark_returns) - periodic_rf_rate) /
  ↪  sd(benchmark_returns), digits = 4)
  adj_sharpe_ratio_benchmark <- round(x = mean(benchmark_returns) / sd(benchmark_returns),
  ↪  digits = 4)
  strategy_annual_return <- ((1 + strategy_total_return) ^ (1 / total_time_available_year))
  ↪  - 1
  benchmark_annual_return <- ((1 + tot_benchmark_returns) ^ (1 / total_time_available_year))
  ↪  - 1
  beta <- round(x = cor(cumulative_equity, benchmark_equity), digits = 4)
  alpha <- round(x = (strategy_annual_return - (annual_rf_rate + (cov(cumulative_equity,
  ↪  benchmark_equity) / var(benchmark_equity)) * (benchmark_annual_return -
  ↪  annual_rf_rate))), digits = 4)
  winning_trades <- trades[trades$Profit > 0, ]
  win <- winning_trades$Profit
  losing_trades <- trades[trades$Profit <= 0, ]
  loss <- losing_trades$Profit
  total_trades <- length(trades$Profit)
  winning_rate <- (length(winning_trades$Profit) / total_trades)
  total_profit <- sum(win)
  total_loss <- sum(loss)
  profit_factor <- round(x = total_profit / abs(total_loss), digits = 2)
  VaR_95_strat <- round(quantile(cumulative_return, 0.05), 4)
  below_threshold_s <- cumulative_return[cumulative_return < VaR_95_strat]
  if (length(below_threshold_s) > 0) {
    ES_95_strat <- round(mean(below_threshold_s), 4)
  } else {
    ES_95_strat <- VaR_95_strat
  }
  VaR_95_benchmark <- round(quantile(benchmark_returns, 0.05), 4)
```

```
below_threshold_b <- benchmark_returns[benchmark_returns < VaR_95_benchmark]
if (length(below_threshold_b) > 0) {
  ES_95_benchmark <- round(mean(below_threshold_b), 4)
} else {
  ES_95_benchmark <- VaR_95_benchmark
}
max_drawdown_percentage <- round(x = (min(drawdowns)), digits = 4) * 100
max_drawdown_percentage_benchmark <- min(benchmark_returns) * 100
trades$TimeInPosition <- as.numeric(difftime(trades$ExitDate, trades$EntryDate, units =
↪  "hours"))
total_time_in_position_minutes <- sum(trades$TimeInPosition, na.rm = TRUE)
total_time_in_position_year <- total_time_in_position_minutes / (60 * 24 * 365.25)
time_exposure <- round(x = ((total_time_in_position_year / total_time_available_year) *
↪  100), digits = 2)
if (nrow(trades) > 0) {
  trades$MAE <- NA
  trades$MFE <- NA
  trades$RR <- NA
  trades$ExitCondition <- NA
  for (i in 1:nrow(trades)) {
    entry_date <- trades$EntryDate[i]
    exit_date <- trades$ExitDate[i]
    entryIndex <- which(dt$time == entry_date)
    exitIndex <- which(dt$time == exit_date)
    averageMr <- mean(MR[entryIndex:exitIndex], na.rm = TRUE)
    averageMr <- ifelse(averageMr > 0, 1, -1)
    trades$MR[i] <- averageMr
    trade_data <- dt[dt$time >= entry_date & dt$time <= exit_date, ]
    mae_price <- min(trade_data$close)
    trades$MAE[i] <- trades$EntryPrice[i] - mae_price
    mfe_price <- max(trade_data$close)
    trades$MFE[i] <- mfe_price - trades$EntryPrice[i]
    returns <- (trades$ExitPrice[i] - trades$EntryPrice[i])
    SL <- -(trades$ATR_Price[i] - trades$EntryPrice[i])
    trades$RR[i] <- (returns / SL)
    trades$RR[i] <- ifelse(trades$RR[i] <= -1, -1, trades$RR[i])
    trades$ExitCondition[i] <- ifelse(trades$RR[i] <= -1, "Stoploss", "Break")
  }
  averageMAE <- round(mean(trades$MAE, na.rm = TRUE), digits = 2)
  averageMFE <- round(mean(trades$MFE, na.rm = TRUE), digits = 2)
  RR <- mean(trades$RR)
} else {
  averageMAE <- NA
  averageMFE <- NA
}
PosReturns <- trades$Return[trades$Return > 0]
NegReturns <- trades$Return[trades$Return <= 0]
expectancy <- round(x = ((winning_rate * mean(PosReturns)) - ((1 - winning_rate) *
↪  mean(NegReturns))), digits = 4)
recovery_factor <- round(x = (strategy_total_return / abs(min(drawdowns))), digits = 2)
risk_reward_ratio <- RR
metrics_table <- data.frame(
  Metric = c("Annualized Strategy return", "Annualized Benchmark return", "Strategy SR",
             "Strategy Adjusted-SR", "Benchmark SR", "Benchmark Adjusted-SR", "Alpha",
             "Beta", "Number of trades", "Winning Rate (%)",
             "Risk-Reward ratio", "Profit Factor", "95% VaR strategy", "95% ES strategy",
             ↪  "Max Drawdown strategy",
```

```
              "95% VaR benchmark", "95% ES benchmark", "Max Drawdown benchmark",
              "Time Exposure", "Average MAE", "Average MFE", "Expectancy", "Recovery
               ↪  Factor"),
    Value = c(sprintf("%.2f%%", strategy_annual_return * 100), sprintf("%.2f%%",
    ↪  benchmark_annual_return * 100),
              sharpe_ratio_strat, adj_sharpe_ratio_strat, sharpe_ratio_benchmark,
              ↪  adj_sharpe_ratio_benchmark, alpha,
              beta, total_trades, sprintf("%.2f%%", winning_rate * 100),
              risk_reward_ratio, profit_factor, sprintf("%.2f%%", VaR_95_strat * 100),
              ↪  sprintf("%.2f%%", ES_95_strat * 100), sprintf("%.2f%%",
              ↪  max_drawdown_percentage),
              sprintf("%.2f%%", VaR_95_benchmark * 100), sprintf("%.2f%%", ES_95_benchmark *
              ↪  100), sprintf("%.2f%%", max_drawdown_percentage_benchmark),
              sprintf("%.2f%%", time_exposure), averageMAE, averageMFE, sprintf("%.2f%%",
              ↪  expectancy), recovery_factor)
  )
  MetricsPlot <- gt(metrics_table) %>%
    tab_header(title = "Trading Performance Metrics") %>%
    cols_label(Metric = "Metric", Value = "Value") %>%
    fmt_number(columns = c(Value), decimals = 4) %>%
    tab_options(
      heading.title.font.size = px(20),
      column_labels.font.size = px(15)
    )
  return(list(metrics_table = metrics_table,
              trades = trades,
              combined_data_filtered = combined_data_filtered))
}
```

12. Run Variations and Save Results

This function runs multiple variations of the trading strategy and saves the results to an Excel file.

Function Name: `run_variations_and_save`
Inputs:

- `data`: Data frame with price data.
- `dataset_name`: Name of the dataset.
- `sub_dataset_name`: Name of the sub-dataset.

Outputs:

- Excel file with metrics and trades data.

```
run_variations_and_save <- function(data, dataset_name, sub_dataset_name) {
  all_results <- list()
  result_idx <- 1
  all_trades_data <- NULL
  for (marge in marges) {
    for (window_size in window_sizes) {
      for (fraction in fraction_clusters) {
```

```
      for (multiplierATR in multiplierATRs) {
        window_cluster <- floor(window_size * fraction)
        timeForward <- window_cluster
        startIdx <- window_size
        print(paste("Processing", dataset_name, sub_dataset_name,
                    "- Marge:", marge, "Window Size:", window_size,
                    "Fraction:", fraction, "MultiplierATR:", multiplierATR))
        result <- runTradingStrategy(data, marge, window_cluster, window_size, startIdx,
        ↪  timeForward, multiplierATR)
        config_name <- sprintf("Marge%d-WinSize%d-Clust%d-Mult%d", marge, window_size,
        ↪  window_cluster, multiplierATR)
        all_results[[result_idx]] <- list(
          config_name = config_name,
          data = result,
          marge = marge,
          window_size = window_size,
          fraction = fraction,
          multiplierATR = multiplierATR,
          frequency = dataset_name,
          sub_dataset_name = sub_dataset_name
        )
        result_idx <- result_idx + 1
      }
    }
  }
}
file_path <- sprintf("~/Master LSM/Master2/Mémoire/Strategy1_results_%s_%s.xlsx",
↪  dataset_name, sub_dataset_name)
wb <- createWorkbook()
all_metrics <- NULL
original_order <- NULL
for (i in seq_along(all_results)) {
  config_name <- all_results[[i]]$config_name
  metrics_data <- all_results[[i]]$data$metrics_table
  metrics_data <- data.frame(Metric = metrics_data$Metric, Value = metrics_data$Value)
  additional_rows <- data.frame(
    Metric = c("Marge", "Window Size", "Fraction", "MultiplierATR", " "),
    Value = c(all_results[[i]]$marge, all_results[[i]]$window_size,
    ↪  all_results[[i]]$fraction, all_results[[i]]$multiplierATR, " ")
  )
  metrics_data <- rbind(additional_rows, metrics_data)
  if (is.null(all_metrics)) {
    all_metrics <- metrics_data
    original_order <- metrics_data$Metric
  } else {
    all_metrics <- merge(all_metrics, metrics_data, by = "Metric", all = TRUE, sort =
    ↪  FALSE)
  }
}
all_metrics <- all_metrics[match(original_order, all_metrics$Metric), ]
addWorksheet(wb, "Metrics")
writeData(wb, "Metrics", all_metrics, startRow = 1, rowNames = FALSE)
all_trades_data <- NULL
for (i in seq_along(all_results)) {
  config_name <- all_results[[i]]$config_name
  trades_data <- as.data.frame(all_results[[i]]$data$trades)
  trades_data$Marge <- all_results[[i]]$marge
```

```
    trades_data$WindowSize <- all_results[[i]]$window_size
    trades_data$Fraction <- all_results[[i]]$fraction
    trades_data$MultiplierATR <- all_results[[i]]$multiplierATR
    trades_data$Frequency <- all_results[[i]]$frequency
    trades_data$SubDatasetName <- all_results[[i]]$sub_dataset_name
    all_trades_data <- rbind(all_trades_data, trades_data)
  }
  addWorksheet(wb, "Tradebook")
  writeData(wb, "Tradebook", all_trades_data, startRow = 1, rowNames = FALSE)
  saveWorkbook(wb, file_path, overwrite = TRUE)
  print(paste("Workbook saved to", file_path))
}
```

A.2.3 Main Execution

This section runs the variations of the trading strategy on different datasets and saves the results.

```
data_list <- list("15 minutes" = data_15min, "1 hour" = data_1hour)

marges <- c(3, 5)
window_sizes <- c(100, 200, 400)
fraction_clusters <- c(1/3, 1/4)
multiplierATRs <- c(3, 4, 5)

for (dataset_name in names(data_list)) {
  dataset <- data_list[[dataset_name]]
  for (sub_dataset_name in names(dataset)) {
    data <- dataset[[sub_dataset_name]]
    run_variations_and_save(data, dataset_name, sub_dataset_name)
  }
}
```

A.3 Performance metrics

1. **Max Drawdown** The Maximum Drawdown is a measure of the largest single drop from peak to trough in the value of the trading system, before a new peak is achieved. It is expressed as a percentage of the peak value.

   Maximum Drawdown is used to assess the risk of a trading strategy. It provides investors with an idea of the possible losses that might be incurred from the peak performance level of a portfolio. A smaller Max Drawdown is preferred as it indicates that the strategy has lower risk. This metric is particularly important for investors who need to manage their drawdowns and understand the volatility and potential losses in their investment strategies.What's more, this measure will enable us later, in a portfolio of trading algorithms, to have a coherent allocation according to risk.

$$Maximum\ Drawdown = \max_{\tau\in[0,T]} \left(\max_{t\in[0,\tau]} V(t) - V(\tau) \right) \tag{33}$$

   Where:

- $V(t)$ is the portfolio value at time t.
- τ is any given time during the observed period T such that $0 \leq \tau \leq T$.
- T is the total length of the observed period.

The Maximum Drawdown is a crucial metric for determining the worst-case scenario for an investment and for evaluating how much an account could shrink during a particularly bad period in the market.

2. **Return annualized** The Annualized Return for a trading system is calculated to standardize the performance of the system across different time frames. It represents the geometric average amount of money earned by an investment each year over a given time period. In the context of backtesting, it is used to normalize the performance of a trading system, allowing for a comparison with other trading strategies or benchmark indices.

   In backtesting, the Annualized Return provides a way to compare the performance of a trading system against other systems or benchmarks on an annual basis, irrespective of the actual period the backtest covers. This standardization is crucial when the backtesting periods vary or when the results need to be compared to annual market performance metrics.

$$Annualized\ Return = \left(\frac{Ending\ Portfolio\ Value}{Beginning\ Portfolio\ Value} \right)^{\frac{1}{Number\ of\ Years}} - 1 \qquad (34)$$

   Where:

   - $Ending\ Portfolio\ Value$ is the final equity of the trading account at the end of the backtesting period.
   - $Beginning\ Portfolio\ Value$ is the initial equity of the trading account at the beginning of the backtesting period.
   - $Number\ of\ Years$ is the total length of the backtesting period expressed in years. If the backtest period is not a whole number of years, this figure is adjusted to represent the fraction of the year.

   While the Annualized Return is useful for comparing the raw growth rates of different trading strategies, it does not account for risk, making it essential to use it in conjunction with risk-adjusted return measures such as the Sharpe Ratio or Sortino Ratio.

3. **Winning rate** The Winning Rate, also known as the Win Ratio, is a key performance metric for a trading system, representing the percentage of trades that are profitable out of the total number of trades executed during the backtesting period.

   The Winning Rate is crucial for understanding the frequency of successful trades generated by a trading strategy. It provides insight into the system's effectiveness at capitalizing on market opportunities. However, while a high Winning Rate is desirable, it should be evaluated in the context of risk-reward ratios, as a strategy can have a high Winning Rate but still be unprofitable if the losses from losing trades significantly outweigh the gains from winning trades.

$$Winning\ Rate = \left(\frac{Number\ of\ Winning\ Trades}{Total\ Number\ of\ Trades} \right) \times 100\% \qquad (35)$$

Where:

- *Number of Winning Trades* is the count of trades that resulted in a profit.
- *Total Number of Trades* is the overall count of trades executed during the backtesting period.

This metric is expressed as a percentage to easily compare the effectiveness of different trading systems or modifications within the same system. A higher Winning Rate suggests a more consistent ability to generate profit, but it must be balanced with an understanding of the profit and loss from each trade to fully assess a strategy's viability.

4. **Sharpe ratio** The Sharpe Ratio is a measure used to assess the performance of a trading system by adjusting its returns for risk. It represents the excess return per unit of risk, with risk measured as the standard deviation of the portfolio returns.

   The Sharpe Ratio is particularly valuable in the context of backtesting trading systems as it provides a risk-adjusted measure of return. It allows for the comparison of trading strategies on a level playing field by factoring in the volatility of returns. A higher Sharpe Ratio indicates a more efficient trading strategy with a higher return per unit of risk. This makes it an essential metric for evaluating the overall effectiveness of a trading system, especially when comparing systems or modifications to a system.

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p} \tag{36}$$

   Where:

   - R_p is the average return of the portfolio over the backtesting period.
   - R_f is the risk-free rate of return, which is the return of an investment with no risk of financial loss.
   - σ_p is the standard deviation of the portfolio returns, representing the volatility or risk of the portfolio.

   The Sharpe Ratio adjusts the returns of a portfolio for the risk taken to achieve those returns. By comparing the excess return over the risk-free rate to the standard deviation of returns, investors can evaluate how much additional risk is associated with additional return. This metric is instrumental in distinguishing between strategies that achieve high returns through high risk and those that are genuinely efficient.

5. **Number of trades** The Number of Trades is a metric that quantifies the total count of trades executed by a trading system during the backtesting period. This metric includes both winning and losing trades, providing a comprehensive overview of the trading activity.

   The Number of Trades is fundamental for evaluating the activity level and the operational frequency of a trading system. It helps investors understand the system's market engagement and its potential impact on trading costs. A higher number of trades might indicate a more aggressive strategy, which could entail higher transaction costs and potentially higher profits or losses. Conversely, a lower number of trades often suggests a more selective strategy, possibly indicating a focus on quality over quantity. This metric is crucial for assessing the balance between potential profit opportunities and associated costs, including the impact of market exposure on the overall performance.

$$Number\ of\ Trades = Total\ Winning\ Trades + Total\ Losing\ Trades \tag{37}$$

Where:

- $Total\ Winning\ Trades$ is the count of all trades that resulted in a profit.
- $Total\ Losing\ Trades$ is the count of all trades that resulted in a loss.

The Number of Trades provides a direct measure of how active a trading strategy is within the market. This metric, when analyzed in conjunction with other performance metrics such as Win Rate, Expectancy, and Profit Factor, offers a more nuanced view of the strategy's effectiveness and operational efficiency.

On the other hand, the results of a backtest will be statistically more significant with a high number of trades. We will see later how to set the number of trades threshold in order to obtain reliable results.

6. **Generated Alpha** Generated Alpha is a measure of the performance of a trading system relative to a benchmark index. It represents the excess return of the trading system over the expected performance based on the benchmark's returns.

   In the context of backtesting trading systems, Generated Alpha is crucial for assessing the value added by a trading strategy beyond a simple market return. It helps to identify the portion of returns attributable to the strategy's specific decisions and market timing, rather than the general market movement. A positive alpha indicates that the strategy has outperformed the benchmark, adjusting for the market's performance, which is particularly valuable for investors looking for strategies that can beat the market.

$$Generated\ Alpha = R_p - (R_f + \beta \times (R_m - R_f)) \tag{38}$$

   Where:

   - R_p is the average return of the portfolio (trading system) over the backtesting period.
   - R_f is the risk-free rate of return.
   - β is the portfolio's sensitivity to movements in the benchmark.
   - R_m is the return of the benchmark index.

   Generated Alpha quantifies the ability of the trading system to generate returns that exceed those expected based on the market's risk (as measured by the benchmark's returns). A significant positive alpha is the hallmark of a superior trading strategy that provides value beyond the broader market's performance.

7. **Beta** Beta is a measure of a trading system's sensitivity to movements in a benchmark index. It quantifies the expected change in the system's returns relative to a 1% change in the benchmark. A beta greater than 1 indicates that the system is more volatile than the market, while a beta less than 1 suggests it is less volatile.

   In the realm of backtesting trading systems, Beta provides insight into the risk profile of a trading strategy compared to the market. It is an essential metric for understanding how market movements are likely to impact the trading system's performance. A high beta implies that the system's returns are expected to increase significantly in a rising market and decrease significantly in a declining market. Conversely, a low beta indicates that the system is less affected by market movements, which could be preferable for risk-averse investors.

$$Beta = \frac{Cov(R_p, R_m)}{Var(R_m)} \tag{39}$$

Where:

- $Cov(R_p, R_m)$ is the covariance between the portfolio (trading system) returns and the market (benchmark) returns.
- $Var(R_m)$ is the variance of the market (benchmark) returns.
- R_p is the return of the portfolio (trading system) over the backtesting period.
- R_m is the return of the benchmark index.

Beta is a critical component in the Capital Asset Pricing Model (CAPM), which is used to calculate the expected return of an asset based on its beta and the expected market returns. In the context of trading system backtesting, analyzing beta helps in tailoring the strategy's market exposure to align with an investor's risk tolerance and market outlook.

The aim later on when building a portfolio of algorithms is to be uncorrelated with market movements and to neutralise this risk with a beta as close as possible to 0 in order to be profitable whatever the market conditions. Beta is therefore also important as a measure of consistent allocation within a portfolio of trading algorithms, with the aim of neutralising market risk.

8. **Market exposure time** Market Exposure Time quantifies the proportion of the total backtesting period during which a trading system is actively engaged in the market. It is measured as the percentage of time the system holds an open position, either long or short, relative to the total time available for trading.

   Understanding the Market time Exposure of a trading system is vital for assessing the strategy's market risk exposure. A higher Time Exposure indicates that the system is more frequently active, potentially increasing both the opportunity for gains and the exposure to market volatility. Conversely, a lower Time Exposure suggests a more conservative strategy that engages in the market less frequently, possibly reducing risk but also potentially limiting opportunities for profit. This metric is crucial for investors who wish to balance their desire for returns with their tolerance for risk.

$$Time\ Exposure = \left(\frac{Total\ Time\ in\ Market}{Total\ Time\ Available\ for\ Trading}\right) \times 100\% \tag{40}$$

Where:

- $Total\ Time\ in\ Market$ is the cumulative time the system holds any position (long or short).
- $Total\ Time\ Available\ for\ Trading$ represents the total time span of the backtesting period.

Market Time Exposure is a critical measure for gauging how actively a strategy participates in the market. It provides insights into the operational dynamics of the trading system and helps in evaluating the balance between potential returns and market risk exposure.

Furthermore, in the context of a portfolio of algorithms with limited funds, it is preferable to have the lowest exposure to the market, since this is an opportunity cost. If, for example, the system gives us a return similar to the benchmark and a much lower exposure to the market, then we can use the remaining funds elsewhere and outperform the market.

9. **Profit factor** The Profit Factor is a metric that quantifies the efficiency of a trading system by comparing the total gross profits to the total gross losses. It is calculated by dividing the sum of all winning trade returns by the sum of all losing trade returns.

   The Profit Factor provides a straightforward assessment of a trading system's profitability. A Profit Factor greater than 1 indicates that the system generates more in winnings than it loses, highlighting its effectiveness in capitalizing on market opportunities. Conversely, a Profit Factor less than 1 suggests that the system's losses outweigh its gains, signaling a potential need for strategy adjustment. This metric is invaluable for traders and investors seeking to evaluate the financial robustness of a trading strategy, particularly in comparison to other strategies or benchmarks.

$$Profit\ Factor = \frac{Total\ Gross\ Profits}{Total\ Gross\ Losses} \tag{41}$$

   Where:

   - $Total\ Gross\ Profits$ is the sum of the profits from all winning trades.
   - $Total\ Gross\ Losses$ is the sum of the losses from all losing trades.

   It's important to note that the Profit Factor does not differentiate between trading strategies that achieve high profitability through a few large wins or through many small wins. Thus, while it is a critical measure of profitability, it should be considered alongside other performance metrics to obtain a comprehensive understanding of a trading system's effectiveness.

10. **Maximum Adverse Excursion (MAE) and Maximum Favorable Excursion (MFE)**: (check le livre stratégie de marché par Bernard Prat-Desclaux) Maximum Adverse Excursion (MAE) and Maximum Favorable Excursion (MFE) are metrics used to assess the extent of potential losses and gains for trades during their lifetime, before they are closed. MAE measures the largest unrealized loss in a trade, whereas MFE measures the largest unrealized gain.

    MAE and MFE provide insights into the risk and reward characteristics of a trading strategy that are not apparent from final outcomes alone. MAE helps in understanding the potential drawdown risk in a trade, indicating the level of risk management needed. MFE, on the other hand, can indicate the strategy's potential profitability and the effectiveness of exit strategies. Analyzing the patterns of MAE and MFE across trades can guide adjustments to stop-loss and take-profit levels, improving the overall efficiency of the trading system.

$$MAE = \max(Entry\ Price - Lowest\ Price\ Before\ Closing) \tag{42}$$

$$MFE = \max(Highest\ Price\ Before\ Closing - Entry\ Price) \tag{43}$$

Where:

- $Entry\ Price$ is the price at which the trade is entered.
- $Lowest\ Price\ Before\ Closing$ is the lowest price reached by the asset before the trade is closed.
- $Highest\ Price\ Before\ Closing$ is the highest price reached by the asset before the trade is closed.

MAE and MFE are critical for evaluating the risk-reward environment of trades within a trading system. By understanding these metrics, traders can fine-tune entry and exit strategies, set appropriate risk levels, and optimize the performance of the trading system.

11. **Expectancy** Expectancy is a trading metric that represents the average amount a trader can expect to win (or lose) per trade. It combines the probability of winning, the average win size, the probability of losing, and the average loss size into a single value, providing a comprehensive measure of a trading system's profitability per trade.

    Expectancy is crucial for evaluating the overall effectiveness of a trading strategy. It goes beyond simple win rates to incorporate the size of wins and losses, offering a more nuanced view of a trading system's performance. A positive expectancy indicates a strategy that is likely to be profitable over time, while a negative expectancy suggests a strategy that will lose money. This metric allows traders to understand the potential profitability of a trading system in the long run, making it a vital tool for strategy selection and refinement.

$$Expectancy = (Win\ Rate \times Average\ Win) - (Loss\ Rate \times Average\ Loss) \tag{44}$$

    Where:

    - $Win\ Rate$ is the proportion of winning trades to total trades.
    - $Average\ Win$ is the average profit from winning trades.
    - $Loss\ Rate$ is the proportion of losing trades to total trades.
    - $Average\ Loss$ is the average loss from losing trades.

    Expectancy provides a dollar-figure estimate of the average profit or loss per trade, integrating both the frequency and magnitude of wins and losses. This comprehensive overview is indispensable for traders aiming to optimize their trading strategies for maximum profitability.

12. **Recovery Factor** The Recovery Factor measures the ability of a trading system to recover from its largest drawdown. It is calculated as the ratio of the total net profit to the absolute value of the maximum drawdown experienced during the backtesting period.

    The Recovery Factor is an essential metric for evaluating the risk-adjusted return of a trading system. It helps in assessing how efficiently a trading strategy can overcome losses and return to profitability. A higher Recovery Factor indicates a system that can recover from losses more effectively, suggesting a potentially lower risk for investors. This metric is particularly useful for comparing the resilience of different trading strategies under adverse market conditions.

$$Recovery\ Factor = \frac{Total\ Net\ Profit}{|Maximum\ Drawdown|} \tag{45}$$

Where:

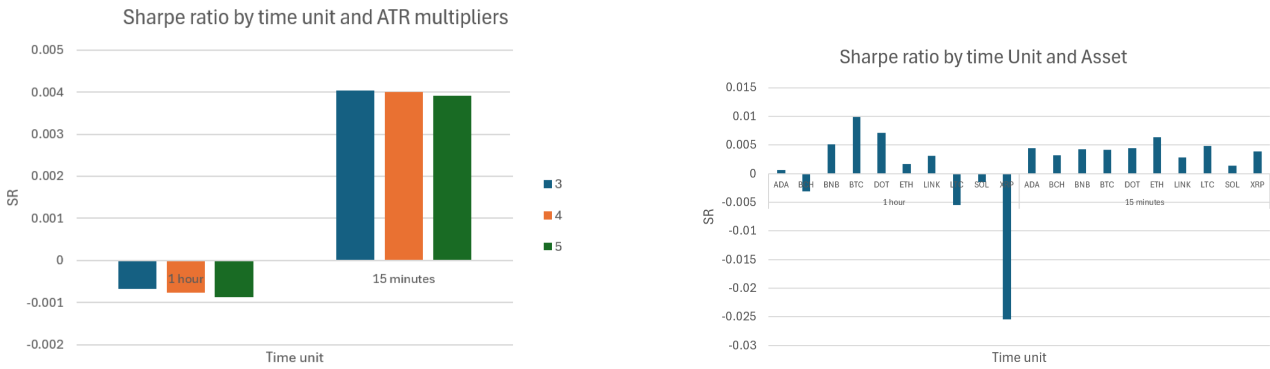
- $Total\ Net\ Profit$ is the sum of all profits minus the sum of all losses over the back-testing period.
- $Maximum\ Drawdown$ is the largest peak-to-trough decline in the value of the port-folio during the backtesting period. The absolute value is used to ensure the metric is always positive.

A Recovery Factor greater than 1.0 is desirable as it indicates that the trading system has generated more profit than the magnitude of its largest loss. This metric provides insight into the durability and stability of a trading strategy, offering a comprehensive view of its performance under stress.

A.4 Additional analysis

A.4.1 Risk management and exit conditions

Sharpe ratio by asset and time Unit



(a) Sharpe ratio by time unit and ATR multipliers

(b) Sharpe ratio by asset

Figure 31: Sharpe ratio by asset and time unit

SR Values and Best Parameter Setup

	Average SR	Variance SR
Margin		
3	0.002260556	6.16218E-05
5	0.000960556	0.000108915
Window size		
100	0.001983333	7.33054E-05
200	0.00184375	7.97982E-05
400	0.001004583	0.000103769
Fraction cluster		
25%	0.001835556	7.41895E-05
33%	0.001385556	9.70936E-05
ATR multipliers		
3	0.0016825	9.75998E-05
4	0.001624583	8.2872E-05
5	0.001524583	7.6951E-05

Based on the provided SR values, the best parameter setup can be determined as follows:

- **Margin**: The best margin is 3, as it has the highest average SR (0.002260556) and a lower variance (6.16218E-05).
- **Window Size**: The best window size is 100, with the highest average SR (0.001983333) and a moderate variance (7.33054E-05).
- **Fraction Cluster**: The best fraction cluster is 25%, with a higher average SR (0.001835556) and a lower variance (7.41895E-05).
- **ATR Multipliers**: The best ATR multiplier is 3, with a higher average SR (0.0016825) and a moderate variance (9.75998E-05).

In conclusion, the optimal parameters for the highest average SR and reasonable variance are Margin 3, Window Size 100, Fraction Cluster 25%, and ATR Multiplier 3.

Sharpe ratio by asset with the best parameter setup

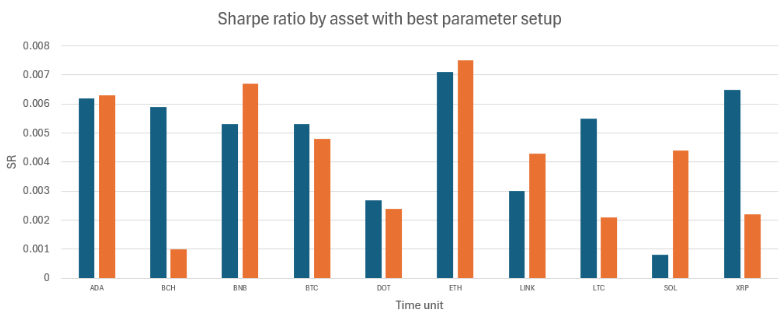


Figure 32: Sharpe ratio by asset

Time In Position according different time Units

Data

Frequency	1 hour	15 minutes
Mean (hour)	248	115
Var	319891	175767
Skew	6	13
Kurtosis	47	211
N	7673	20917

Independent Samples t-test

- **Test Statistic (t-value)**: 47.48
- **P-Value**: 3.15×10^{-16}

Since the p-value is less than 0.05, we reject the null hypothesis. This indicates that there is a statistically significant difference in the means of the time in position (hours) for 1 hour and 15 minutes frequencies.

Levene's Test for Difference in Variances

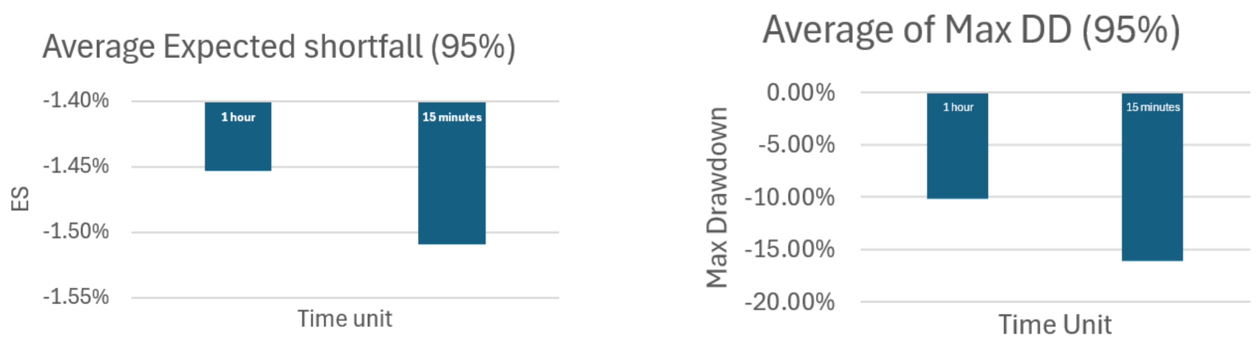
- **Test Statistic**: 0.46
- **P-Value**: 0.5

Since the p-value is greater than 0.05, we fail to reject the null hypothesis. This indicates that there is no statistically significant difference in the variances of the time in position (hours) for 1 hour and 15 minutes frequencies.

Conclusion

Based on the t-test, we conclude that there is a significant difference in the means of the time in position (hours) between the 1 hour and 15 minutes frequencies.

Risk management - Drawdown and Expected shortfall



(a) Average Expected shortfall(95%) (b) Average Max Drawdown

Figure 33: Expected shortfall and Max drawdown by time unit

Expected Shortfall (95%) and Max Drawdown

	Mean	Variance
Margin		
3	-1.32%	0.00056%
5	-1.65%	0.00064%
Window size		
100	-1.43%	0.00082%
200	-1.48%	0.00087%
400	-1.53%	0.00087%
Fraction cluster		
25%	-1.48%	0.00078%
33%	-1.49%	0.00096%
ATR multiplier		
3	-1.57%	0.00069%
4	-1.50%	0.00079%
5	-1.38%	0.00094%

Table 17: Average and Variance of 95% ES strategy according to different parameters

	Mean	Variance
Margin		
3	-13.53%	0.52%
5	-12.81%	0.38%
Window size		
100	-12.99%	0.40%
200	-12.84%	0.43%
400	-13.67%	0.52%
Fraction cluster		
25%	-13.63%	0.43%
33%	-12.70%	0.47%
ATR multiplier		
3	-15.52%	0.57%
4	-13.05%	0.40%
5	-10.94%	0.28%

Table 18: Average and Variance of Max Drawdown strategy according to different parameters

Winning Rate and Risk-Reward Ratio

Parameters	**Value**	**Number of Trades**				**Winning Rate**				**RR Ratio**			
ATR Multiplier		**3**	**4**	**5**	**Mean**	**3**	**4**	**5**	**Mean**	**3**	**4**	**5**	**Me**
Margin	3	55	49	46	50	0.31	0.33	0.34	0.33	1.14	1.02	1.00	1.0
	5	33	29	26	29	0.26	0.29	0.31	0.29	3.02	2.88	2.57	2.8
Window Size	100	48	43	39	44	0.29	0.31	0.32	0.31	2.11	2.03	1.85	2.0
	200	44	39	36	40	0.29	0.31	0.32	0.31	2.12	1.95	1.79	1.9
	400	40	35	32	36	0.28	0.30	0.31	0.30	2.08	1.95	1.79	1.9
Fraction Cluster	25%	48	43	39	44	0.29	0.31	0.32	0.31	1.80	1.67	1.50	1.6
	33%	48	35	33	36	0.29	0.31	0.32	0.31	2.37	2.22	2.07	2.2

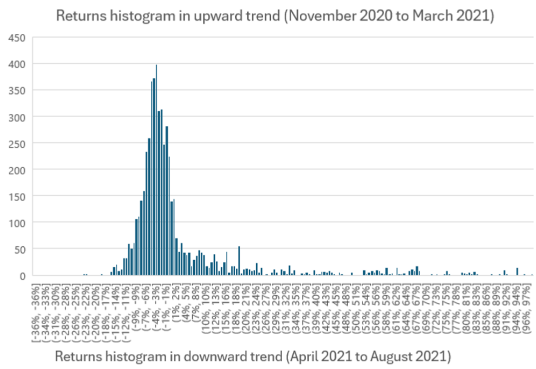
Table 19: Heatmap of the average number of trades, winning rate, and RR ratio according to parameters

Conclusion

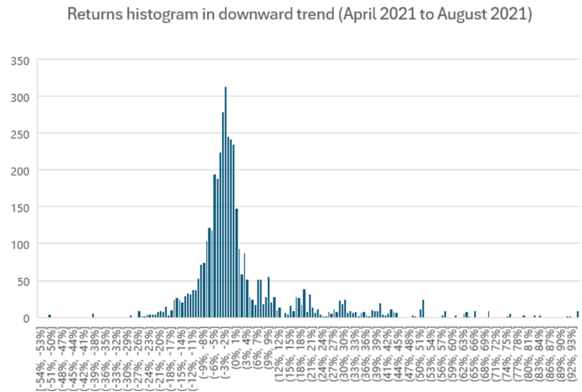
- **Margin**: Margin of 3 is better for winning rate (0.34) while Margin of 5 is better for RR ratio (3.02).
- **Window Size**: Window size of 100 has the highest winning rate (0.32) and RR ratio (2.11).
- **Fraction Cluster**: Fraction cluster of 33% has the highest RR ratio (2.37), while both clusters have the same winning rate (0.32).
- **ATR Multiplier**: ATR multiplier of 3 is generally better across different parameters.

A.4.2 Strategy and entry conditions

Impact of the trend



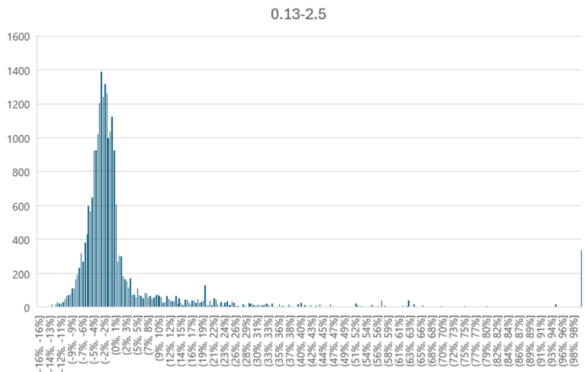
(a) Histogram in Uptrend



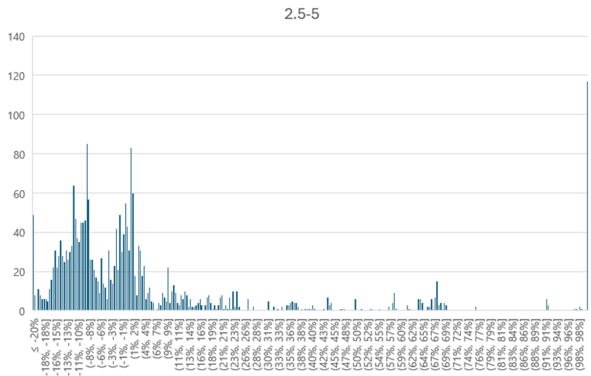
(b) Histogram in Downtrend

Figure 34: Histograms based on the trend

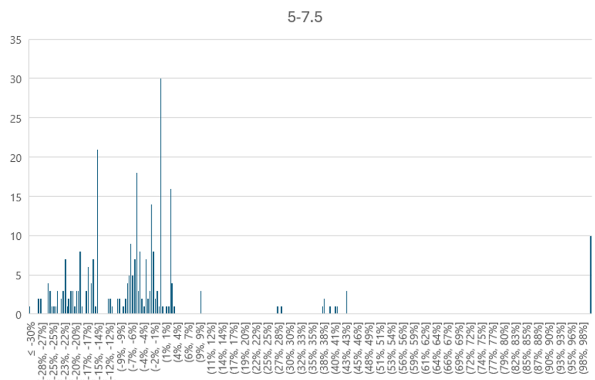
Impact of the volatility (ATR based)



(a) Histogram for Lower Interval



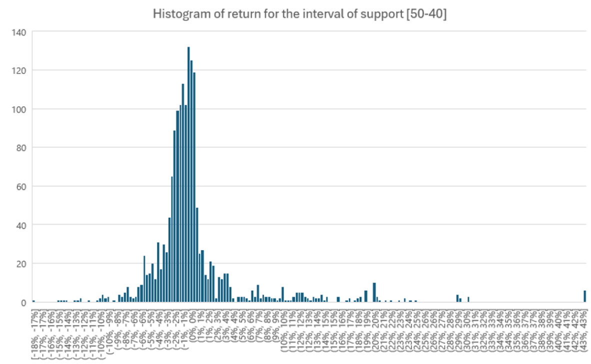
(b) Histogram for Middle Interval



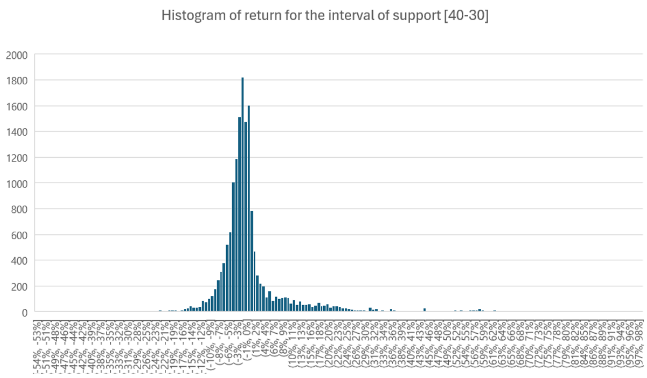
(c) Histogram for Upper Interval

Figure 35: Histograms for different volatility intervals

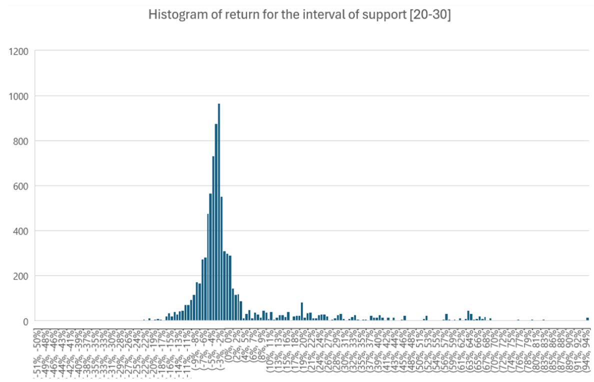
Impact of RSI support value



(a) Histogram for upper Interval



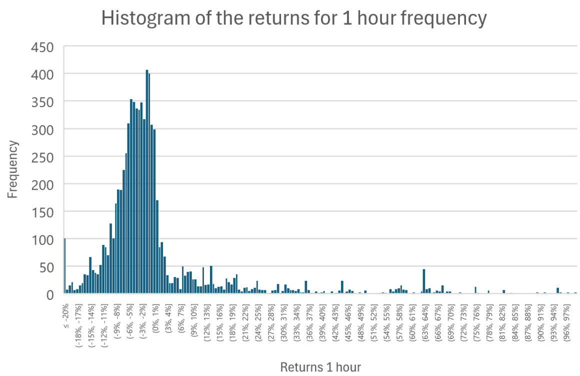
(b) Histogram for middle Interval



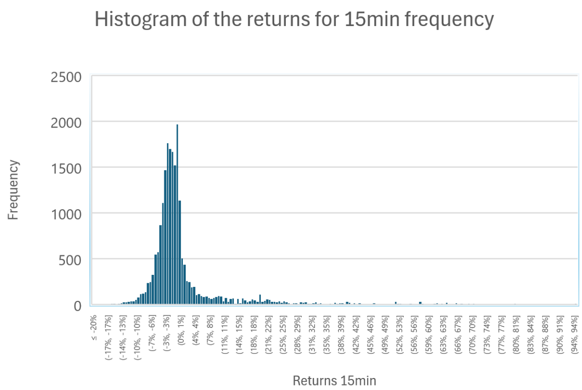
(c) Histogram for lower Interval

Figure 36: Histograms based on the intervals

Impact of the trading time frame:



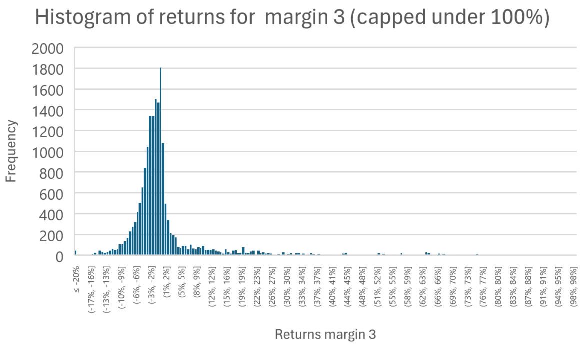
(a) Histogram of 1 hour frequency returns



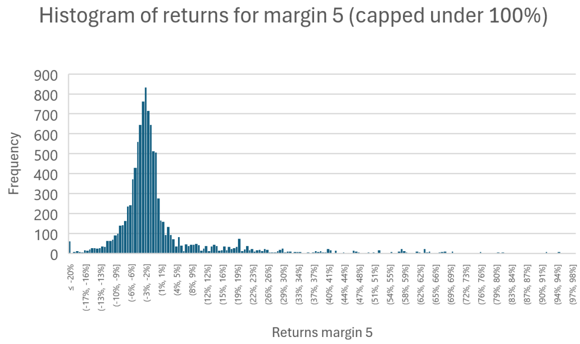
(b) Histogram of 15 minutes frequency returns

Figure 37: Comparison of Returns Frequency Histograms

Impact of the margin parameter:



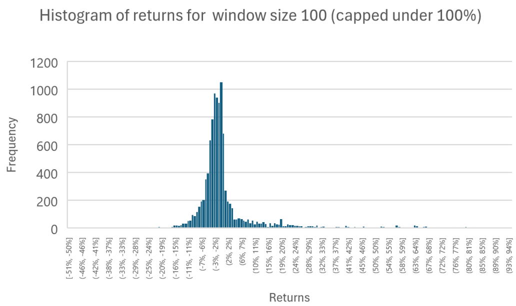
(a) Histogram of returns for margin = 3



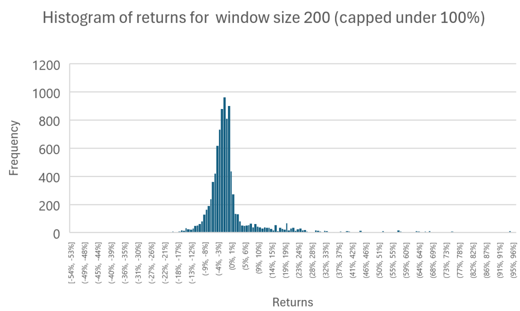
(b) Histogram of returns for margin = 5

Figure 38: Comparison of Returns Histograms for margins

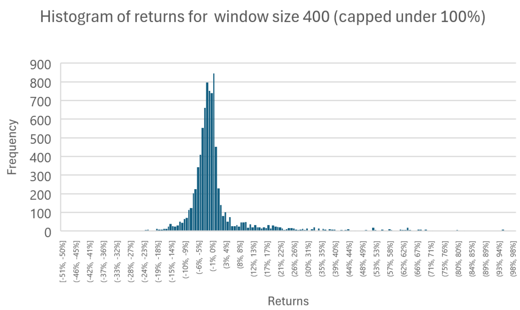
Impact of the window size parameter:



(a) Histogram of returns for window size = 100



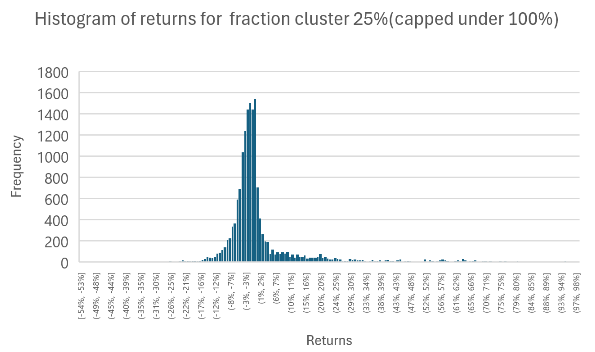
(b) Histogram of returns for window size = 200



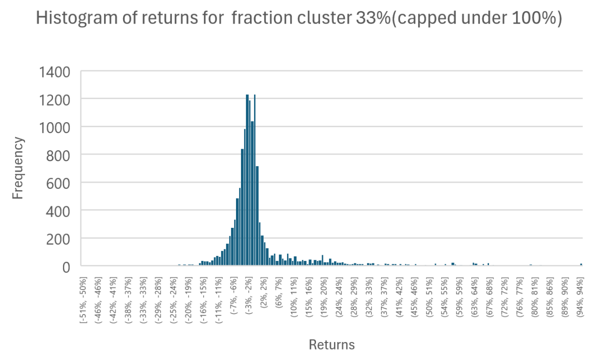
(c) Histogram of returns for window size = 400

Figure 39: Comparison of Returns Histograms for Different Window Sizes

Impact of the cluster parameter:



(a) Histogram of returns for fraction cluster = 25%



(b) Histogram of returns for fraction cluster = 33%

Figure 40: Comparison of Returns Histograms for Fraction Clusters

A.4.3 Final version additional analysis

Returns distributions

	Return 15 minutes	Return 1 hour
Mean	0.18%	0.60%
Variance	0.07%	0.71%
Skew	5.03	8.96
Kurtosis	42.84	116.54
N	4073	587

Table 20: Returns Distribution by time unit

A.5 Examples applied to the methodology

We will now illustrate our five-step methodology with several examples aimed at generating trading strategy ideas. This preliminary step allows us to evaluate the relevance of these ideas before considering the construction of a complete system, which will then undergo more extensive backtesting and statistical testing.

EXAMPLE 1 - HUMAN PSYCHOLOGY FOMO AND HERD BEHAVIOR BIAS

1. *Formulating a Fundamental Hypothesis Based on Human Psychology:*

The first step involves formulating a fundamental hypothesis based on an assumption such as human psychology in this example. Financial markets are often influenced by the emotions and behaviors of participants. Concepts such as fear, greed, euphoria, and panic play a crucial role in price formation and market trends.

Examples of psychological hypotheses:

- **Fear and Greed Theory**: Traders tend to sell quickly during price declines (fear) and buy rapidly during price increases (greed).

- **Crowd Behavior Theory**: Investors often follow the actions of the majority, which can lead to speculative bubbles or selling panics.

2. *Identifying This Psychological Hypothesis in the Form of a Market Pattern Based on Price Action or RSI:*

The second step involves identifying market patterns that reflect these psychological hypotheses. These patterns can manifest through price action or technical indicators such as the RSI (Relative Strength Index).

Examples of patterns:

- **Japanese Candlestick Patterns**: Hammers, shooting stars, dojis, etc., which indicate trend reversals based on trader emotions.
- **RSI Divergences**: When the price reaches a new high or low, but the RSI does not confirm this movement, indicating potential weakness in the current trend.

3. *Centralizing Similar Patterns:*

After identifying individual patterns, the third step is to centralize and group similar patterns. This centralization helps to better understand the conditions under which these patterns form and to identify common characteristics.

Methods of centralization:

- **Using Graphical Analysis Software**: For example, TradingView or MetaTrader to visualize and annotate patterns.
- **Compiling Historical Data**: Collect and organize examples of similar patterns in a database to facilitate comparative analysis.

4. *Analyzing the Detailed Construction of the Pattern:*

Once the patterns are centralized, the next step is to analyze their construction in detail. This involves breaking down each pattern to understand the elements that comprise it and the specific market conditions that favor it.

Aspects to analyze:

- **Price Formation**: Examine how prices evolve before, during, and after the pattern formation.
- **Transaction Volume**: Analyze volumes to identify periods of high or low activity, often associated with trend changes.
- **Technical Indicators**: Study indicators such as RSI to see how they interact with the pattern.

5. *Systematizing and Objectifying the Set-up, the Pattern:*

The final step is to systematize and objectify the set-up based on the identified patterns. The goal is to transform subjective observation into an objective and reproducible trading strategy.

The systematization process involves defining clear rules to establish entry and exit criteria based on technical indicators and specific price levels.

This is a very simplistic example based on human psychology. However, it is entirely possible to propose more advanced and complex hypotheses. For instance, the following example is based on a more sophisticated concept, demonstrating how intricate patterns and theories can be applied to develop robust trading strategies.

EXAMPLE 2: SELF-FULFILLING PROPHECY CONCEPT

Context: Financial markets are often influenced by the beliefs and expectations of investors. Self-fulfilling prophecies play a crucial role in market movements, as the actions of investors based on their anticipations can cause these anticipations to come true.

Hypothesis: If a large number of investors observe a support level in prices, there is a high likelihood that the self-fulfilling prophecy phenomenon will lead to a price rebound at this support level. The collective belief that prices will rebound at this level encourages investors to buy, which indeed causes the expected rebound.

Detailed Methodology applied

1. *Formulate a Psychological Hypothesis*

Support levels are zones where prices have historically struggled to fall below. The shared belief that an asset will rebound from a support level prompts investors to place buy orders at this level, creating increased demand when prices reach the support, thus causing a rebound. For example, if many traders observe that the price of a stock tends to rebound around \$100, they will place buy orders at this support level. When the price reaches \$100, the buy orders are triggered, increasing demand and causing a price rebound.

2. *Identify Patterns in the Markets*

To verify this hypothesis, we look for occurrences of this behavior using technical indicators and graphical analysis tools. We analyze periods where prices reach major support levels to identify if there are significant rebounds from identified support levels. We can analyze volumes to identify periods of high buying activity around support levels.

3. *Centralize Similar Patterns*

We centralize several instances of these patterns to analyze their common characteristics. We can examine the amplitude of the rebounds, the duration of the movements, the market conditions etc.

4. *Analyze the Detailed Construction of the Patterns*

We use technical analysis tools to understand the construction of the patterns:

Fibonacci Ratios: We identify common retracement levels after a rebound from support. Specific retracements, such as a 50% or 61.8% rebound, are well-known levels that can be subject to self-fulfilling prophecy

RSI (Relative Strength Index): We can check if we observe oversold levels before the price rebound.

Price Structure: We examine price movements before, during, and after the rebound.

Volumes: We analyze transaction volumes to identify periods of high liquidity around support levels.

5. *Systematize and Objectify the Pattern*

We transform the identified pattern into systematic trading strategy. We define clear entry and exit condition rules based on technical indicators and price levels. In this example, if we observe that most of the time the RSI is in the oversold zone when prices are on major support and there is a significant rebound, these will be our entry conditions. For exit conditions, we can set a stop-loss below the price support level and a take-profit based on our observations of the Fibonacci ratios, for example, at 50% of the previous downward impulse.

After demonstrating how we can transition from a financial market concept to a systematic trading strategy based on technical analysis, we can apply a similar approach to a well-known market adage. The adage "Buy the rumors, sell the news" provides an excellent example. We will explore how this adage can be transformed into a preliminary trading strategy.

EXAMPLE 3 - "BUY THE RUMOR, SELL THE NEWS"

“Buy the rumor, sell the news” is a common adage in the trading world, especially in volatile markets where information and speculation drive price movements. This strategy is based on the concept of acting on market expectations and speculation rather than waiting for actual events or announcements.

1. *Formulate a Psychological Hypothesis*

Investors tend to act on rumors or anticipated news, buying assets in expectation of favorable outcomes. This collective behavior drives prices up before the actual news release. Once the news is confirmed, the initial buying pressure diminishes, and profit-taking ensues, causing a price drop.
For example if traders hear a rumor that a company is about to announce excellent quarterly earnings. Anticipating a positive reaction, they buy the company’s stock, driving the price up. When the earnings are officially announced, traders begin to sell their shares to lock in profits, leading to a price decline.

2. *Identify Patterns in the Markets*

To verify this hypothesis, we look for occurrences where significant rumors precede news releases and observe the corresponding price movements. We analyze the price increases during the rumor period and the subsequent declines post-news release. We track price changes from the rumor start to the news release and shortly after. We can observe volume spikes during the rumor period and post-news release.

3. *Centralize Similar Patterns*

We centralize several instances of these patterns to analyze their common characteristics. This involves examining the magnitude of price changes, the duration of the rumor effect, and the volume patterns.

An example of this pattern would be a price that rises during the rumour period and a volume that rises before the news is published and falls afterwards.

4. *Analyze the Detailed Construction of the Patterns*

We use technical analysis tools to understand the construction of these patterns:

Price Structure: Examine the price trajectory from the rumor onset to post-news release.

Volume Analysis: Assess trading volumes to identify high activity periods.

Technical Indicators: Utilize RSI to spot overbought conditions during the rumor period.

5. *Systematize and Objectify the Pattern*

We transform the identified pattern into a systematic trading strategy. So we define clear Rules by setting entry conditions based on the onset of the rumor and technical indicators showing positive momentum. Exit conditions could be set just before the anticipated news release or based on specific technical signals indicating overbought conditions.

Through these various examples, we demonstrate how we use technical analysis as a tool to analyze and quantify a hypothesis about financial markets. Technical analysis serves as a means to mathematically observe the construction of patterns that reflect underlying behavior (our initial hypothesis). In the next example, we will show how targeting specific technical indicators can be beneficial based on the initial hypothesis.

EXAMPLE 4 - PRICE MANIPULATION BY LARGE PLAYERS

Context: The cryptocurrency markets are young, lightly regulated, and often characterized by relatively low transaction volumes. In this context, large players (referred to as "whales") and cryptocurrency brokers have the ability to significantly influence prices due to their size and purchasing power.

1. *Hypothesis*

Manipulation by Large Players:
Large market players, or "whales," have an incentive to exert pressure on prices when they plan to make significant purchases. The goal is to push prices towards liquidity zones, where many stop-loss orders are placed by small investors (retail). By triggering these orders, they can buy back positions at lower prices.

For example a whale planning to purchase $1.5 billion worth of Bitcoin may sell heavily to drive prices down towards a stop-loss zone. Once prices reach this zone and trigger stop-loss orders, the whale can buy back the positions at a lower price, thereby maximizing profits. The stop-loss zone thus acts as a liquidity pocket that serves as a counterparty to absorb the whale's massive purchases.

Manipulation by Cryptocurrency Brokers:
Cryptocurrency brokers, such as exchanges, may also manipulate prices to generate additional brokerage fees. These brokers benefit from pushing prices towards zones where numerous buy or sell orders are placed (liquidation zones) to collect transaction fees.

To understand the viability of this manipulation, it is crucial to consider several factors:

1. Proximity to the Liquidity Zone:

The distance between current prices and the target liquidity zone is critical. The further the liquidity zone, the harder it is for manipulators to push prices towards it.

2. Market Conditions:

Market conditions, such as transaction volumes and prevailing trends, influence the difficulty of manipulation. For example, if the market is strongly bullish with high buying volumes, it will be more resource-intensive for manipulators to drive prices down.

3. Necessary Resources:

The amount of capital needed to influence prices is also a determining factor. Manipulators must evaluate whether the capital investment required to manipulate prices will be profitable compared to the potential gains from the operation.

4. Size of the Liquidity Pocket:

The size of the liquidity pocket in the target zone will determine the potential gains for manipulators. A large liquidity pocket means more orders will be executed, increasing potential profits.

Considering these factors, it becomes possible to analyze the viability and profitability of market manipulations by large players and cryptocurrency brokers. Manipulation patterns can be identified and used to develop systematic trading strategies. By monitoring order flow indicators and volume profiles, it is possible to spot liquidity zones and predict price movements induced by market manipulations.

Identifying such manipulation allows for exploitation within a trading system, and technical analysis will help in identifying these patterns by analyzing their construction in detail and the market conditions conducive to observing such patterns if they exist.

2. *Identify Patterns in the Markets*

To verify this hypothesis, we look for occurrences of this behavior using order flow indicators and volume profile tools. These tools help us identify areas of high liquidity and determine significant price movements caused by large players.

- Order Flow: Analyzing transactions to identify areas of high buying or selling activity.
- Volume Profile: Using traded volume at different price levels to spot areas of support, resistance, and liquidity.

We can search for occurrences of significant bearish movements accompanied by liquidation spikes, followed by high buying volumes.

3. *Centralize Similar Patterns*

We centralize multiple instances of these patterns to analyze their common characteristics. We examine the amplitude of price movements, the duration of movements, and the surrounding market conditions.

4. *Analyze the Detailed Construction of the Patterns*

We use technical analysis tools to understand the construction of these patterns. In this example, we will use the volume profile and order flow indicators to detect areas of liquidity.

5. *Systematize and Objectify the Pattern*

We transform the identified patterns into systematic trading strategies. For instance, consider the entry conditions where we require the trading volume to be higher than the average, indicating a liquidity zone, and a significant area detected by the order flow indicator. The exit conditions might involve setting a stop-loss just below the liquidity zone defined by the order flow. Additionally, we can establish a take-profit level based on resistance identified in the price structure.

Conclusion

In conclusion, these examples demonstrate how our methodology allows us to progressively transform a hypothesis about financial markets into a discernible pattern using technical analysis tools. By thoroughly studying these patterns, we can systematically establish precise entry and exit conditions, ultimately creating a robust trading strategy. Having a clear strategy will subsequently enable us to program and backtest the strategy on historical data, as well as apply rigorous statistical tests.