Stock Market Prediction and Portfolio Composition Using a Hybrid Approach Combined with Self-Adaptive Evolutionary Algorithms

Rodrigo Manuel Oliveira de Gouvêa Lopes de Almeida

Thesis to obtain the Master of Science Degree in Information Systems and Computer Engineering

Supervisor: Professor Rui Fuentecilla Maia Ferreira Neves

Examination Committee

Chairperson: Professor João António Madeiras Pereira
Supervisor: Professor Rui Fuentecilla Maia Ferreira Neves
Member of the Committee: Professor Alexandre Paulo Lourenço Francisco

December 2019
Declaration

I declare that this document is an original work of my authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.
Acknowledgments

I want to express my gratitude to Professor Rui Neves for his patience, encouragement, guidance, support, and understanding through the entire development of this work. His valuable financial market knowledge, a constant presence, and availability made this work possible.

I am also very grateful to my friends and colleagues who helped through this great journey and made hard work more enjoyable. A special thanks to Sofia Gonçalves who was by my side each time I needed more support and also to Rodrigo Lousada, Renato Henriques and Marco Montez for their help and advice when requested.

Finally, I would like to thank my family for their support and wise pieces of advice.

To each and every one of you - Thank you.
Abstract

This work presents a new approach to maximize financial market investment returns. It incorporates two Evolutionary Algorithms (EAs) combined with fundamental and technical investment strategies. The first EA (simple) maintains its evolutionary parameters static during evolution. The second (self-adaptive) introduces the variation operators’ parameters’ values in the representation for them to evolve. The EA is responsible for optimizing the weight that financial ratios from the F-Score have on composing static/dynamic portfolios. Furthermore, it is also responsible for defining the importance that selected technical indicators have on revealing the best timing for market positions placement. A fundamental and a technical case study was created employing companies from the Standard & Poor’s 500 (SP500). These were trained/tested in a sliding window scheme between 01/2012 and 12/2018. Results showed that both case studies surpassed the SP500 returns, performing their best results using a self-adaptive EA combined with a static portfolio and a sliding window of 2 years of train/test. On the one hand, the technical case study showed better results in “bear markets” since it predicted some market declines. Its best subtest achieved returns on average 2.2x and in its best 3.5x higher than the benchmark. Its Sharpe Ratio achieved, on average, 4.9x and in its best 9x higher results than the benchmark. On the other hand, the fundamental case study displayed better performance in the “bull market”, achieving high market prices. Its best subtest achieved returns on average 2.4x and in its best 3.2x higher than the benchmark. Its Sharpe Ratio achieved, on average, 4.4x and in its best 6.5x higher results than the benchmark.

Keywords

Evolutionary Algorithms; Self-adaptive Evolutionary Algorithms; Technical Analysis; Fundamental Analysis; Technical Indicators; F-Score; S&P500.
Resumo

Este trabalho apresenta uma nova abordagem para maximizar retornos de investimento no mercado financeiro. Para tal, incorpora dois EAs combinados com estratégias de investimento técnicas e fundamentais. O primeiro EA (simples) mantém os seus parâmetros evolucionários estáticos durante a evolução. O segundo (auto-adaptável) introduz os valores dos parâmetros dos operadores de variação na representação para que eles possam evoluir. O EA é responsável por otimizar o peso que os rácios financeiros do F-Score têm durante a composição de portfólios estáticos/dinâmicos. Além disso, são também responsáveis por definir a importância que os indicadores técnicos selecionados têm quando revelam a melhor altura para fazer posições no mercado. Neste trabalho foi criado um caso de estudo fundamental e outro técnico, utilizando empresas do SP500. Estes foram treinados/testados num esquema de janela deslizante entre 01/2012 e 12/2018. Os resultados mostraram que ambos os casos de estudo superaram os retornos do SP500, obtendo os melhores resultados usando um EA auto-adaptável combinado com um portfólio estático e uma janela deslizante de 2 anos de treino/teste. Por um lado, o caso de estudo técnico mostrou melhores resultados nos bear markets, pois conseguiu prever algumas quedas no mercado. O seu melhor subteste obteve retornos em média 2.2x e no seu melhor 3.5x maior que o benchmark. O seu Sharpe Ratio alcançou, em média, 4.9x e nos seus melhores resultados 9x maiores que o benchmark. Por outro lado, o caso de estudo fundamental apresentou melhor desempenho em bull markets, atingindo valores altos. O seu melhor subteste obteve retornos em média 2.4x e no seu melhor 3.2x superior à referência. O seu Sharpe Ratio alcançou, em média, 4.4x e nos seus melhores resultados 6.5x mais altos que o benchmark.

Palavras Chave

Algoritmos Evolucionários; Algoritmos Evolucionários Auto-adaptáveis; Análise Técnica; Análise Fundamental; Indicadores Técnicos; F-Score; S&P500.
Contents

1 Introduction .............................................. 1
   1.1 Motivation ........................................ 4
   1.2 Work Goals ...................................... 4
   1.3 Contributions ................................... 4
   1.4 Document Structure .......................... 5

2 Related Work ........................................... 6
   2.1 Background ...................................... 7
      2.1.1 Financial Markets ........................... 7
      2.1.2 Types of Markets ............................ 7
      2.1.3 Market Forecasting ......................... 10
      2.1.4 Artificial Intelligence ..................... 18
   2.2 State Of the Art .................................. 26
      2.2.1 Works on Financial Markets Prediction .... 26
      2.2.2 Works on Evolutionary Algorithms ........... 28
      2.2.3 Analysis of the Related Work ............... 29

3 Architecture ............................................ 33
   3.1 System Overview .................................. 35
   3.2 Architecture .................................... 35
   3.3 Flow Description .................................. 36
   3.4 User Layer ....................................... 38
   3.5 Data Preparation .................................. 39
      3.5.1 Data Downloading ............................ 39
      3.5.2 Data Processing .............................. 41
      3.5.3 Data Manipulation ........................... 43
   3.6 Evolutionary Algorithm .......................... 51
      3.6.1 Representation ............................... 51
      3.6.2 Fitness Function ............................. 53
3.6.3 Selection .................................................. 53
3.6.4 Variation Operators ....................................... 54
3.6.5 Replacement .............................................. 55
3.6.6 Initialization .............................................. 55
3.6.7 Termination Condition .................................... 55
3.7 Portfolio Creator ........................................... 55
3.8 Trading Simulator ........................................... 57
3.8.1 Dynamic F-Score ......................................... 57
3.8.2 Investment Strategies ..................................... 57
3.9 Statistics .................................................. 59
3.9.1 Return Performance Metrics ............................. 59
3.9.2 Performance Metrics ..................................... 60

4 Results ................................................. 62
4.1 Results Overview ........................................... 63
4.2 Sliding Window ............................................ 63
4.3 Case Studies ............................................. 64
4.3.1 Technical Investment .................................... 65
4.3.2 Fundamental Investment ................................. 71
4.3.3 Results Comparison ..................................... 78

5 Conclusion ............................................... 84
5.1 Conclusions ............................................... 85
5.2 Future Work .............................................. 86

A Appendix .................................................. 95
List of Figures

2.1 Example of a Cash Flow Statement. ......................................................... 13
2.2 Support and Resistance. ................................................................. 16
2.3 Upward Trendline. ................................................................. 16
2.4 Leading Indicator. ................................................................. 17
2.5 On Balance Volume as example of a Momentum Indicator. ................. 18
2.6 Two Examples of Representations. ................................................... 22
2.7 Example of One-point crossover. ...................................................... 24
2.8 Example of N-point crossover with $n = 2$. ....................................... 24
2.9 Example of Uniform Crossover. The array $[0.4, 0.6, 0.3, 0.8, 0.9, 0.1]$ of random numbers and $p = 0.5$ were used to decide inheritance for this example. ....................... 24
2.10 Example of mutation operator. ...................................................... 25
2.11 Architecture of an Evolutionary Algorithm. ....................................... 26

3.1 System Architecture. ................................................................. 36
3.2 System Data Flow. ................................................................. 37
3.3 Relative Strength Index Example. ................................................... 46
3.4 Exponential Moving Average Example. ........................................... 47
3.5 Moving Average Crossovers Example. ............................................ 47
3.6 Rate of Change Example. ............................................................ 48
3.7 Moving Average Convergence Divergence Example. ......................... 50
3.8 Volatility Index Example. ............................................................. 51
3.9 Fundamental Chromosome composed of the F-Score (2.8) ratios. .......... 52
3.10 Technical Chromosome composed of the selected technical indicators. .... 52
3.11 Self-Adaptive Chromosome composed of the probability of mutation $P_m$ and crossover $P_c$ for the variation operators. ............................................ 53

4.1 Sliding Window Scheme. ............................................................. 64
4.2 Technical Investment With Dynamic F-Score Simulations. . . . . . . . . . . . . . . . . . . . . 67
4.3 Technical Investment Without Dynamic F-Score Simulations. . . . . . . . . . . . . . . . . 67
4.4 Technical Investment With Dynamic F-Score and Self-Improvement Simulations. . . . . . 69
4.5 Technical Investment Without Dynamic F-Score but With Self-Improvement Simulations. . 70
4.6 Fundamental Investment With Dynamic F-Score Simulations. . . . . . . . . . . . . . . . . 74
4.7 Fundamental Investment Without Dynamic F-Score Simulations. . . . . . . . . . . . . . . 74
4.8 Fundamental Investment With Dynamic F-Score and Self-Improvement Simulations. . . . 75
4.9 Fundamental Investment Without Dynamic F-Score but With Self-Improvement Simulations. 77
4.10 Average Training of the Evolutionary Algorithm. . . . . . . . . . . . . . . . . . . . . . . . 79
# List of Tables

<table>
<thead>
<tr>
<th>Section</th>
<th>Table Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Related Work Table</td>
<td>32</td>
</tr>
<tr>
<td>3.1</td>
<td>Quarter Dataset Sample</td>
<td>40</td>
</tr>
<tr>
<td>3.2</td>
<td>Annual Dataset Sample</td>
<td>40</td>
</tr>
<tr>
<td>3.3</td>
<td>SP500 Dataset Sample</td>
<td>41</td>
</tr>
<tr>
<td>3.4</td>
<td>Stock Prices Dataset Sample</td>
<td>41</td>
</tr>
<tr>
<td>3.5</td>
<td>VIX Dataset Sample</td>
<td>42</td>
</tr>
<tr>
<td>3.6</td>
<td>Annual Dataset Sample After Processement</td>
<td>44</td>
</tr>
<tr>
<td>3.7</td>
<td>RSI Scoring Rules</td>
<td>45</td>
</tr>
<tr>
<td>3.8</td>
<td>EMA Scoring Rules</td>
<td>46</td>
</tr>
<tr>
<td>3.9</td>
<td>MAC Scoring Rules</td>
<td>47</td>
</tr>
<tr>
<td>3.10</td>
<td>ROC Scoring Rules</td>
<td>48</td>
</tr>
<tr>
<td>3.11</td>
<td>MACD Scoring Rules</td>
<td>49</td>
</tr>
<tr>
<td>3.12</td>
<td>VIX Scoring Rules</td>
<td>50</td>
</tr>
<tr>
<td>3.13</td>
<td>Fictitious Example of the Fundamental Dataset</td>
<td>56</td>
</tr>
<tr>
<td>3.14</td>
<td>Fictitious Example of the Stock Prices Dataset</td>
<td>58</td>
</tr>
<tr>
<td>3.15</td>
<td>Trading Dataframe Example</td>
<td>58</td>
</tr>
<tr>
<td>4.1</td>
<td>Technical Investment Results</td>
<td>66</td>
</tr>
<tr>
<td>4.2</td>
<td>Weights Used in the Best Technical Results</td>
<td>71</td>
</tr>
<tr>
<td>4.3</td>
<td>Fundamental Investment Results</td>
<td>73</td>
</tr>
<tr>
<td>4.4</td>
<td>Weights Used in the Best Fundamental Results</td>
<td>78</td>
</tr>
<tr>
<td>4.5</td>
<td>Technical Case Study Correlation Matrix</td>
<td>80</td>
</tr>
<tr>
<td>4.6</td>
<td>Fundamental Case Study Correlation Matrix</td>
<td>80</td>
</tr>
<tr>
<td>A.1</td>
<td>Stock Prices Dataset With Calculated Technical Scores Sample</td>
<td>96</td>
</tr>
</tbody>
</table>
Acronyms & Abbreviations

**A-KNN-E**  
Adaptative Classification Nearest Neighbor using Euclidean distance

**A-KNN-M**  
Adaptative Classification Nearest Neighbor using Manhattan distance

**A-KNN**  
Adaptative Classification Nearest Neighbor

**A/D**  
Advance/Decline Line

**ABC**  
Artificial Bee Colony Algorithm

**AI**  
Artificial Intelligence

**ANN**  
Artificial Neural Network

**AvgP**  
Average Profit

**AvgR**  
Average Return

**BMFBOVESPA**  
São Paulo Stock Market

**CBOE**  
Chicago Board Options Exchange

**CFO**  
Cash Flow Operations

**CPU**  
Central Process Unit

**CRSP**  
Center for Research in Security Prices

**CR**  
Current Year’s Ratio

**EABC**  
Evolutionary Artificial Bee Colony Algorithm

**EA**  
Evolutionary Algorithm

**EBITDA**  
Earnings Before Interest, Taxes, Depreciation and Amortization

**EC**  
Evolutionary Computation
EGARCH  Exponential Generalized Autoregressive Conditional Heteroscedasticity
EMA    Exponential Moving Average
EMT    Theory of Efficient Markets
EQ OFFER Equity Offered
ESN    Echo State Network
FRPCA  Fuzzy Robust Principal Component Analysis
GAAP   Generally Accepted Accounting Principles
GARCH  Generalized Autoregressive Conditional Heteroscedasticity
GA     Genetic Algorithm
GLARE  Generalized Analytic Rule Extraction
IPO    Initial Public Offering
KNN-E  Nearest Neighbor Classification using Euclidean distance
KNN-M  Nearest Neighbor Classification using Manhattan distance
KNN    Nearest Neighbor Classification
KPCA   Kernel-Based Principal Component Analysis
LEV    Leverage
MACD   Moving Average Convergence Divergence
MAC    Moving Average Crossover
MARGIN Gross Margin
MA     Moving Average
MDD    Maximum Drawdown
MLP    Multilayer Feed-Forward Neural Network
MOEA   Multi-Objective Genetic Algorithm
MaxP   Maximum Profit
MinP   Minimum Profit
NASDAQ National Association of Securities Dealers Automated Quotations
NC Net Change
NN Neural Networks
NYSE New York Stock Exchange
OBV On-Balance Volume
PCA Principal Component Analysis
PETR4 Petróleo Brasileiro S.A.
PGSVM Penalty Guided Support Vector Machines
PT Profitable Transactions
ROA Return on Assets
ROC Rate of Change
ROI Return on Investment
ROR Rate of Return
RRR Risk-Return Ratio
RSI Relative Strength Index
SMA Simple Moving Average
SP500 Standard & Poor’s 500
SPY Standard & Poor’s 500 Exchange-Traded Fund
TCFFA Total Cash Flow from Financing Activities
TCFIO Total Cash Flow from Investing Operations
TCFOA Total Cash Flow from Operating Activities
TI Technical Indicators
TURN Turnover
UT Unprofitable Transactions
VALE5 Vale S.A.
VIX Volatility Index
1

Introduction

Contents

1.1 Motivation .................................................. 4
1.2 Work Goals ................................................. 4
1.3 Contributions .............................................. 4
1.4 Document Structure ....................................... 5
Financial markets have been around for more than four centuries. In 1602 the first “modern” securities market was founded in the Netherlands not long after the establishment of the Dutch East India Company. This creation was the result of the necessity of exchanging commodities between traders. Since the creation of the Amsterdam Stock Exchange many have followed its steps, and today a financial market is available in most nations of the modern world.

For a long time, markets have been a territory of interest not only for traders but also for researchers. The trends of the market, the price unpredictability, and the amount of money that generates created such interest that multiple theories and studies to comprehend the markets have been developed.

In the Theory of Efficient Markets (EMT), Fama et al. (1970) state that market prices reflect all the available information and that the market reacts rapidly to variations in the world [1]. This hypothesis additionally affirms that stocks trade at fair value. With that in mind, it is inconceivable for investors to purchase undervalued stocks and make a profit out of them (beat the market). Even though this hypothesis has substantial valid premises, there is also the opposite side of supporters who do not agree and have proposed numerous strategies to vanquish the market.

Two main strategies attempt to beat the market: the Fundamental and the Technical. On the one hand, the fundamental strategy bases itself on financial statements and applies this economic data to build portfolios [2]. On the other hand, the technical strategy uses past data (prices/indicators) and tries to mine patterns like trends that may prompt a superior sign to where the market is moving [3].

With the expansion and study of the financial markets, numerous models have been developed, and much data generated. Many brokers also started offering more technical information, and with the power of computers, many saw the possibility of processing all this information in a short time to help with trading decisions. From that point forward, multiple researchers from the most different areas have developed systems and hypotheses that enable the traders to have more power in time to make a trade. A particular area that has brought much innovation is the Artificial Intelligence (AI) domain that, in models such as Evolutionary Computation (EC) and Neural Networks (NN), has demonstrated potential on the research for forecasting models [4–8].

In this work, a new approach to portfolio composition is proposed, combining Evolutionary Computation with multiple indicators, ratios and proposed theories. To solve the problem of which stocks to choose for a portfolio composition, fundamental indicators and ratios are suggested in combination with an Evolutionary Algorithm (EA). The time and type of position also have incredible significance, so the Evolutionary Algorithm helps, with the support of technical indicators, to perceive the best time to make the position in the markets.

The Evolutionary Algorithm relies on creating populations that, based on their fitness, evolves the best individuals and removes the ones who got the worst results. With some evolution time, the population winds up showing signs of improvement and can advance the procedures of portfolio arrangement and
market exchanging.

1.1 Motivation

Since the conception of financial markets, multiple systems have surged with the purpose of making a more accurate market forecast. As increased computational power is available to researchers, numerous fields get good advancements and are able to test new theorems. Artificial Intelligence is a very robust instrument that, when applied to markets, can yield promising results. As a consequence, the motivation stands in developing a system that is able to use this technology, combining two distinct areas in order to beat the stock market and provide better profits to the investors.

1.2 Work Goals

The goals of this work are to:

- Maximize financial investment returns using Evolutionary Algorithms, combined with fundamental and technical information, to compose strong portfolios and make the appropriate positions in the market.

- Study the impact of holding a dynamic portfolio versus a static portfolio.

- Understand which ratios/indicators have a more positive/negative impact on the trading returns.

- Study the influence of adopting a self-adaptive EA to evolve the candidate solutions, versus an EA with preselected parameters.

- Understand the impact of using a fundamental/technical investment strategy.

- Attempt to perform better than the Standard & Poor’s 500 (SP500).

1.3 Contributions

The main contributions of this dissertation are:

- The implementation of a portfolio composition system that using financial information combined with the F-Score ratio ranks companies. Furthermore, the developed system allows the portfolios to be dynamic, updating the portfolio each year, or static, using the same portfolio throughout the whole investment period.
• The development of Evolutionary Algorithms that, combined with a fundamental and technical investment strategy, choose the best weights for portfolio composition variables and technical analysis indicators. Not only that, but one of the Evolutionary Algorithms was developed with self-adaptive capabilities. These features allow it to include its evolution parameters in the evolution, enabling them to evolve and adapt.

• The creation of a sliding window scheme to understand the best combination of years for training and testing trading algorithms.

1.4 Document Structure

This work is structured as follows:

• Chapter 2 addresses, in the first section, the fundamental knowledge and concepts both in the markets field as in the Artificial Intelligence area. In the second part, it describes the related work and analyzes what ideas and concepts can be taken to use for further development.

• Chapter 3 presents the proposed architecture, its functionalities and an in-depth analysis is performed for each developed module.

• Chapter 4 presents and examines the case studies developed in this work according to selected validation metrics. In the first section, the case studies are presented and their structure explained. After that, an in-depth look is taken at each case study and their respective subtests. At the end of the section, the results are compared and a summary is presented.

• Chapter 5 presents the conclusions of this thesis and suggestions for future work.
Related Work

Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Background</td>
<td>7</td>
</tr>
<tr>
<td>2.2 State Of the Art</td>
<td>26</td>
</tr>
</tbody>
</table>
This chapter provides, in a first section, fundamental concepts on Financial Markets and its different types, Market Forecasting as well as the tools used to perform it and Artificial Intelligence with a particular focus on Evolutionary Algorithms. In the second part, the related work is presented in Market Forecasting and Evolutionary Computation. The purpose of this section remains on providing information on what has already been developed in the area, and what conclusions can be drawn to implement in the future.

2.1 Background

This sub-section describes the concepts related to Financial Markets (its multiple derivations) and Market Forecasting with regards to the fundamental and technical tools used to implement it. The Artificial Intelligence topic is also approached, along with its sub-types with particular attention to Evolutionary Algorithms, and its structure.

2.1.1 Financial Markets

The concept of the financial market covers multiple types of marketplaces where several sorts of trading occur. The financial market is the name given to a place where buyers and sellers meet to trade assets. Until the XX\(^\text{TH}\) Century, the financial market was a physical place like the New York Stock Exchange (NYSE). However, after the technological revolution (1970), new markets have appeared such as the National Association of Securities Dealers Automated Quotations (NASDAQ) where most of the trading is done using an electronic system. The markets act in several ways, but all of them serve the same essential functions such as price setting, asset valuation, arbitrage, raising capital, commercial transactions, investing and risk management [9].

The investors or “floor traders” buy and sell assets in the exchanges to make the most profit they can. These decisions of buying and selling are based on expectations about future prices, and these, in turn, are conditional on present buy and sell decisions [2].

“The market is a pendulum that forever swings between unsustainable optimism (which makes stocks too expensive) and unjustified pessimism (which makes them too cheap).” - Jason Zweig [10]

2.1.2 Types of Markets

The Financial Market subdivides into multiple categories. In this section, each category is described with particular attention to the Equity Markets.
(A) Foreign Exchange Markets

Typically, every nation has its currency to use in each country. However, trade between countries occur and sometimes involves the mutual exchange of different currencies. This trade of currencies takes place in the foreign exchange market [11]. These form the most extensive financial markets by operating in every corner of the world in every single currency, underpinning all other markets [9].

(B) Money Markets

The money market is traditionally described as the market for financial assets that have maturities of one year or less. Fundamentally, it is the market for short-term debt instruments. One of the most critical roles of the money market is to provide an outlet for big companies with temporary excess cash to invest that capital in short-term money market instruments [11, 12].

(C) Bond Markets

A bond is a loan by one part (the investor or holder) to another part (the issuer) that works as a contract, agreement or guarantee. From all the financial instruments, bonds are the most widely used of all. An investor who acquires a bond is lending money. The bond represents the issuer's contractual agreement to pay interest and repay the principal according to specified terms. The bond market is the financial market where participants trade this type of instrument [9, 13, 14].

(D) International Fixed-Income Markets

During an extended period, financial market activity occurred just within the limits of a single country and on that country's currency. Nonetheless, a growing percentage is currently crossing national borders as individuals move capital into currencies that offer more extended returns, and borrowers scan the globe for cash at the most reduced cost. The international market is not an exchange nor even a particular group of products. This sort of exchange refers to a decentralized system in which currencies held outside their nations of origin are re-loaned without being changed over to currency [9].

(E) Futures and Commodities Markets

Commodities are a very particular type of physical goods that can be stored for long periods (or unlimited periods). Its value depends heavily on its physical location and measurable physical attributes. Commodities markets have existed for centuries and served the essential functions of setting prices for commodities, helping to protect producers and consumers against the risks in a world where prices have plenty of volatility. Likewise, these kinds of markets allow commodities producers to trade them for other sorts of goods [3, 9].

(F) Options and Derivatives Markets

To an average person, volatility may seem unpredictable and at a dimension, random. However, over time, average volatility can be assessed, the probability of price tendencies can be predicted and
investors can determine how much they are willing to pay to decrease volatility. The derivatives market is where this occurs. Since a derivative is a financial instrument with a value defined by prices of other things, an unlimited variety of derivative products might exist. Derivatives exchanges transact products based on a wide variety of interest rates, stock indexes, commodity prices, exchange rates and even non-financial items like the weather [9, 15].

(G) Equity Markets

Equities are usually known as shares or stocks that represent partial ownership of a company and also work as financial instruments. Shares are formed initially when a corporation is established. At this point, the corporation is a private organization as a close group of investors owns all the shares. The owners can pick the number of shares that they believe it is appropriate for the enterprise plans and valuation. As the company grows, some may choose to convert to a public enterprise, which lists on a public stock exchange, and where public investors can buy or sell shares. This process is identified as listing, where already existing or additional shares can be created and sold to the public during an Initial Public Offering (IPO). These shares qualify their holders to a share of the dividends declared by the board of directors to distribute from the corporation’s profits. Other than that, they also entitle owners to vote on important decisions at annual general meetings.

Shares come into distinct classes: Common and Preferred. These two broad classes have differing rights. On the one hand, preferred shares regularly possess a higher claim on dividends and on the assets of a firm in the case of liquidation, but usually, have no rights to vote and have a fixed dividend that will not increase with earnings. On the other hand, in the event of liquidation, common shareholders only have right to a company’s assets after bondholders, preferred shareholders and other debt-holders get paid in full.

The capital market is a general term that includes the stock market and other venues for trading financial products and long-term debt. The stock market allows banking institutions and investors to exchange stocks, either publicly or privately, to match capital savers with capital needers and to raise capital [11, 16]. In a broader definition, capital markets accommodate the trading of physical assets besides currencies and derivatives. This type of market offers a diverse range of investment possibilities, helping investors achieve enhanced portfolio diversification. This helps to support a better matching of the demands of investors with those of issuers.

Following an IPO, the shares trade on stock exchanges, and their valuation is influenced by supply and demand, which is determined by the underlying fundamentals of the macroeconomic business factors like the interest rates and market sentiment. The profit of shareholders is a result of both the dividends paid to them from the companies profits and of any actions in the share price (capital growth) [9, 16].
2.1.3 Market Forecasting

Market Forecasting is defined as the process of determining the direction in which the price is likely to move. Forecasting can be approached in two ways: using Technical Analysis or Fundamental Analysis. While technical analysis concentrates on the examination of market activity, fundamental analysis centers around the financial powers of supply and demand that cause the price to vary [17].

In this section, tools such as fundamental and technical analysis, financial statements, ratios and other indicators used to implement market forecasting get introduced and explained.

(A) Fundamental Analysis

It is believed that stocks have a real “fundamental” value, different from their current market price. The fundamental value of a stock should be defined concerning the earning power of the assets or concerning the fundamental value of other stocks. Throughout time, the market price of a stock should tend towards its fundamental value. When it happens, the analysis of fundamental values provide a useful guide to investors.

The fundamental analyst attempts to (in an emotionless process) authenticate how underlying values are reflected in stock prices. To do this, he uses economic data (e.g., production, consumption, exports) to forecast prices [2, 18]. This process allows the investor to uncover trading opportunities by identifying potential transitions to significantly more ample or tighter supply-demand balances [3].

(i) Financial Statements

Financial Statements are provided by organizations that present their financial activities, status, and performance, trimestrial, or annually in the most transparent way possible (usually following the Generally Accepted Accounting Principles (GAAP)). These records are meant for the general public with regards to fundamental investors that attempt to mine value information for their investments. Financial Statements for business, as a rule, incorporate three documents: Balance Sheet, Income Statements and Cash Flows.

• Balance Sheet

In the process of conducting a fundamental analysis of a company, it is essential to understand how much the company has in assets, cash, property, how much it owes to vendors, banks, and the bondholders. This type of information is on a document called balance sheet [19]. The balance sheet is an accountant’s snapshot of the company accounting value on a particular date [20].

The balance sheet divides itself into two parts: on the left are the assets and on the right stand the liabilities and stockholders’ equity. There are many different kinds of assets. They include cash, receivables, inventory, property, plant and equipment. The accounting definition
that expresses the balance sheet and describes the balance is:

\[ \text{Assets} \equiv \text{Liabilities} + \text{Stockholders' Equity} \quad (2.1) \]

The assets in the balance sheet are arranged in order by the length of time it usually would take an ongoing firm to convert them into cash. The liabilities and the stockholders’ equity are arranged in the order in which they must be paid [20].

When examining a balance sheet, the financial manager should be familiarized with three concepts: Accounting Liquidity, Debt versus Equity and Value versus Cost [20].

- **Accounting Liquidity**
  Accounting Liquidity alludes to the facility and quickness with which assets can be converted into cash.
  Current assets are the most liquid and incorporate funds and assets that will be transformed into cash within a year from the creation of the balance sheet. The vital part about current assets is the quick availability to be transformed into cash. All the other assets are those that will not or cannot be converted into cash in the year ahead.
  Accounts receivables are amounts to be collected (owed to the company) from clients for goods or services sold to them. It is essential to notice that a certain percentage whose goods were sold to will not pay, thus an expected sum for bad debts is deducted from the receivables.
  Inventory is composed of the product that has been warehoused with the purpose of selling to its vendors.
  Fixed assets are the least liquid type of assets. Tangible fixed assets include property, plant, and equipment. Assets like these do not convert to cash from regular business activity, and they are not commonly used to pay costs such as payroll [19, 20].

- **Debt Versus Equity**
  Liabilities are the company’s debt that arises during the progression of business operations. These obligations require a payout of cash within a stipulated time. Stockholders’ equity is the difference between assets and liabilities:

\[ \text{Assets} - \text{Liabilities} \equiv \text{Stockholders' Equity} \quad (2.2) \]

This represents in (accounting terms) the stockholder’s share in the company [19, 20].

- **Value Versus Cost**
  Usually, the word “value”, is related to an increase in positive balance, but in this particular case “carrying value” and “book value” is based on cost. This might be misleading to
many readers of financial statements, as they carry on thinking that the firm’s assets are registered at the values at which buyers and sellers are willing to trade assets (true market value). However, it is a good indicator of the investor’s perception of the business prospects [19, 20].

• Income Statement
The income statement is a document that provides the analyst with the results of the operations from the company in a specific time. This report divides itself into three significant parts: Revenue, Expenses and Profits. From this, it is possible to mine if a company is making money or not, what kind of margins it had, if it needed to spend a lot on research and development and if it needed to use much leverage to generate money.
The Earnings Before Interest, Taxes, Depreciation and Amortization (EBITDA) is a critical metric in this report for compiling all those economic definitions. Besides that, the non-operating section includes all the financial costs and variables like the interest expense.
In a separated part is also reported the amount of taxes levied on income and, finally, the last item that shows up in the report is the net income [19, 20]. The definition of income is the equivalent:

\[ Revenue - Expenses \equiv NetIncome \]  

(2.3)

• Cash Flow
The cash flow statement is the document that provides the analyst with information on how much money is arriving and leaving the company. There are two types of cash flow: the Positive and the Negative. The positive is when the company is bringing more money than it is spending, the negative is when the company is spending more than it is bringing. The cash flow statement can be divided into three segments that can be summed up in equations: Total Cash Flow from Operating Activities (TCFOA), Total Cash Flow from Investing Operations (TCFIO) and Total Cash Flow from Financing Activities (TCFFA) [19].
In the image below 2.1, an example of a Cash Flow Statement is presented with the several described parts.
Cash Flow from Operating Activities:

\[ TCFOA = NetIncome + Deprecation + Amortization \]  

(2.4)

Cash flow from Investing Operations:

\[ TCFIO = CapitalExpenditures + OtherInvestingCashFlowItems \]  

(2.5)
Cash Flow Statement

Cash Flow from Operations

<table>
<thead>
<tr>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cash receipts from customers</td>
<td>86,772</td>
</tr>
<tr>
<td>Cash paid for Inventory</td>
<td>-7,400</td>
</tr>
<tr>
<td>Cash paid for wages</td>
<td>-53,000</td>
</tr>
<tr>
<td>Net Cash Flow from Operations</td>
<td>26,372</td>
</tr>
</tbody>
</table>

Cash Flow from Investing

<table>
<thead>
<tr>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cash receipts from sale of property and equipment</td>
<td>13,500</td>
</tr>
<tr>
<td>Cash paid for purchase of equipment</td>
<td>-17,500</td>
</tr>
<tr>
<td>Net Cash Flow from Investing</td>
<td>-4,000</td>
</tr>
</tbody>
</table>

Cash Flow from Financing

<table>
<thead>
<tr>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cash paid for loan repayment</td>
<td>-5,000</td>
</tr>
<tr>
<td>Net Cash Flow from Financing</td>
<td>-5,000</td>
</tr>
</tbody>
</table>

Net Increase in Cash 17,372

**Figure 2.1**: Example of a Cash Flow Statement.

Cash flow from Financing Activities:

\[
TCFFA = CashDividendsPaid + IssuanceOfStock + IssuanceOfDebt \tag{2.6}
\]

Now if we add all these values into an equation then we get the company's Net Change (NC):

\[
NC = TCFOA + TCFIO + TCFFA \tag{2.7}
\]

(ii) **Fundamental Indicators**

Fundamental Indicators are a crucial part of the fundamental analysis. These tools provide the analyst with better methods for understanding the financial statements and formulate the appropriate conclusions.

- **Piotroski F-Score**

The F-Score is a fundamental scoring system that ranges from 0-9 (9 being the best), aiming to classify how strong the financial position of a company is. To make this classification, Piotroski et al. (2012) proposed an aggregate signal (2.8) that results from the addition of specific individual binary signals. These signals are financial ratios inferred from reports like the financial reports above explained. It is important to notice that the result of the Piotroski F-Score may be different in distinct industries [21].


\[ F_{\text{SCORE}} = F_{\text{ROA}} + F_{\Delta \text{ROA}} + F_{\text{CFO}} + F_{\text{ACCRUAL}} + F_{\Delta \text{MARGIN}} + \]
\[ F_{\Delta \text{TURN}} + F_{\Delta \text{LEVER}} + F_{\Delta \text{LIQUID}} + Eq\_\text{OFFER} \quad (2.8) \]

To understand each ratio, it is essential to understand not only the equations behind the value but also the meaning of the ratios.

The binary values are always given taking into account the result of the ratios. Usually, if the ratio value is positive, then receives a value of 1, on the contrary, gets the value of 0. In change ratio values, usually, are compared to the previous year. If the value is more significant than the previous year, than it is given a 1, else, it is given a 0. Besides this being considered a rule there may be exceptions and those will be pointed below.

Return on Assets (ROA) is a ratio that allows the analyst to understand how profitable a company is.

\[ ReturnOnAssets = \frac{\text{NetIncome}}{\text{AverageTotalAssets}} \quad (2.9) \]

Cash Flow Operations (CFO) is a good ratio for the analyst to understand the easiness that a company has to transform sales into cash.

\[ CashFlowOperations = \frac{\text{OperatingCashFlow}}{\text{TotalAssets}} \quad (2.10) \]

Accrual is the records of the revenues and expenses regarding a company during a specific time period. In this specific case, a value of 1 is given if CFO (2.10) > ROA (2.9) and a 0 in contrary.

Gross Margin (MARGIN) is a percentage that expresses how profitable the company's operations are. It measures how much revenue from sales the organization holds, after all the costs associated with providing a service or making a product are deducted.

\[ GrossMargin = \frac{\text{Revenues} - \text{CostOfGoodsSold}}{\text{Revenues}} \times 100 \quad (2.11) \]

Turnover (TURN) compares the sales value of a company to its average sales value, and is a clear way to measure the efficiency of how the company is managing its assets to create value.

\[ AssetTurnover = \frac{\text{TotalSales}}{\text{AverageTotalAssets}} \quad (2.12) \]

The change in Leverage (LEV) allows the analyst to have oversight over the changes in the
company debt related issues. In this ratio, a value of 1 is given if the result is lower in the current year (compared to the last year) and 0 otherwise.

\[
\text{Leverage} = \frac{\text{TotalLongTermDebt}}{\text{AverageTotalAssets}}
\]  

(2.13)

The Liquid variable measures the Current Year’s Ratio (CR) in comparison to the previous year’s current ratio. This result is a good way for the analyst to examine the capacity that the company has on paying debts.

\[
\text{CurrentRatio} = \frac{\text{TotalCurrentAssets}}{\text{TotalCurrentLiabilities}}
\]  

(2.14)

Finally, the last variable presented by Piotroski is Equity Offered (EQ OFFER). This variable tells the analyst if there is a change in the number of shares since the previous year. If the company issued new shares, then the value should be 0 otherwise should be 1 [21].

(B) Technical Analysis

While fundamental analysis uses the information that's available in the financial statements to forecast prices, technical analysis is based on finding patterns in data (as volume, open interest, sentiment measures, or others) available from databases [3]. During the technical analysis, the analyst looks at past data and searches for patterns like trends or regular cycles that could lead to correct market predictions. After the patterns extraction, it is possible to apply them and choose when to buy and sell stocks accordingly to the rules deduced from the data [11].

A description of the multiple instruments used in the technical analysis is presented below.

(i) Support and Resistance

The prices of stocks are a constant battle between bulls (buyers) and bears (sellers). Bulls are traders who plead for the price to go up. Bears are traders who praise for the price to go down. This constant battle creates what is called volatility.

On the one side, a value of support happens when the buyers take control of the price rate and prevent it from going lower than a specific threshold. On the other side, a resistance level happens when the sellers prevent the price from going above a certain value. These two terms used often in market trading refer to barriers that allow the traders to talk about price levels [18].

In 2.2 an example of Support and Resistance is presented.

(ii) Trends

While support and resistance are thresholds that cause the stock price to settle in a specific value, trends are a consistent fluctuation in price that makes the price value follow a specific direction. This direction is named a trendline. One of the most significant aspects of a trend is how much
volume is involved in it. If the trend presents low volume values, then the trend is weak. However, a high volume of transactions should accompany a strong trend [18].

In 2.3, an upward trendline is presented as an example of a trend type.

(iii) Technical Indicators

During trading, the bulls and bears try to predict when it's the best time to make an order. This order should be as precise as they can calculate so that their profits are maximized. To know when to do the orders on time, traders use what is called an indicator. These are mathematical calculations on stock price and volume that enable the trader to anticipate price changes [18].

The Technical Indicators (TI) can be divided into two different groups: the Lagging and the Leading Indicators:

- **Lagging Indicators**

  Lagging Indicators are usually associated with recognizing how strong a trend is. These types of indicators usually know how the market is moving (i.e., rising or falling) and are typically applied to long trends.

  Good examples of lagging indicators are the Moving Average (MA) and the Moving Average Convergence Divergence (MACD). The first helps to smooth the price action, whereas the
second represents the relation between the convergence moving average and the divergence moving average.

- **Leading Indicators**
  Leading Indicators help the investor know if the security is “overbought” or “oversold.” With this type of information, the trader can try to predict the next price of the stock and profit from it. Good examples of leading indicators are the Relative Strength Index (RSI) and the Volatility Index (VIX). The first measures the significance of a price change to evaluate if it is overbought or oversold. The second describes the market expectation of the volatility in a 30-day expectancy [18].

In 2.4, an example of a Leading Indicator is presented.

![Leading Indicator](image)

**Figure 2.4:** Leading Indicator.

(iv) **Market Indicators**

While Technical Indicators provide a glimpse of how a stock is behaving, market indicators present information on how a specific market is performing. Market Indicators are an essential part of the technical analysis because although the technical indicators let us know when we should buy an individual security, market indicators give us a glimpse of what direction the market is going, and that adds a depth layer to the prediction [18].

Market indicators can be divided into three different categories: Monetary, Sentiment, and Momentum:

- **Monetary Indicators**
  The monetary indicators are related to economic data. These help the analyst understand the type of financial surroundings and industry of the analyzed company. Examples of the monetary indicators are the interest rates or the money supply [18].

- **Sentiment Indicators**
  The sentiment indicators are related to the expectations of the traders. Although the stock market is much based on supply and demand, there is a big part of speculation. The expectations
of the traders are much correlated with that, making it possible to understand, based only on the sentiments of the market, if it is going bullish or bearish. Examples of sentiment indicators are odd lot sales (what the smallest investors are doing), the ratio of bullish investors versus bearish investors, or even the Advance/Decline Line (A/D), which outlines the variations in the advance-decline index value [18].

- **Momentum Indicators**
  The momentum indicators tell the analyst what prices are doing currently. This indicator does this by looking further than just the price. Examples of this type of market indicator are the number of stocks that achieved an all-time high or all-time low, and the relation between the number of stocks that declined in price versus the number of stocks that increased in price [18].
  Other examples of these indicators are the RSI and the On-Balance Volume (OBV). While the first was already presented in the Leading Indicators section (iii), the second is concerned with employing the volume flow to predict variations in the stock price. In figure 2.5 the OBV indicator is presented as an example of a Momentum Indicator.

  ![Figure 2.5: On Balance Volume as example of a Momentum Indicator.](image)

(v) **Time Element**

Time has a significant role in technical analysis. It enables the investors to see a pattern happen and it is the fundamental element of a trend or a cycle. The manipulation of time in market visualizations is a crucial instrument to comprehend what has happened in months, days, hours, or minutes in stocks. This element is such a vital part of an investment that can reveal different conclusions only by zooming in or zooming out, from the same stock [18].

2.1.4 **Artificial Intelligence**

Artificial Intelligence is a broad scientific field of study. To best define it, let us start by dividing it into four categories: thinking humanly, thinking rationally, acting humanly and acting rationally. The first two
are connected with the thought process and reasoning, whereas the other two address behavior. The first and third measure success in terms of closeness to human performance, whereas the second and fourth measure toward an ideal performance measure denominated rationality.\[22\].

(A) Thinking Humanly

To comprehend if a computer thinks as human, we first need to understand how humans think. There are three different approaches to do this:

1) Attempting to get the thoughts as they pass by, using introspection.
2) Following the behavior of an individual while operating psychological experiments.
3) Observing the brain working with brain imaging.

Once we have sufficiently precise information regarding the thought process, it becomes easier and feasible to try to express its process as a computer program. If the program’s input-output behavior matches the human behavior, and the traces of its reasoning match the steps used by humans to achieve such a solution, then such serves as evidence that some of the program’s mechanisms could also be operating in humans.\[22\].

(B) Thinking Rationally

Aristotle, the Greek philosopher, was one of the first to attempt studying “right-thinking,” which represents an irrefutable reasoning process. To do it, he created “syllogisms” that, when given the correct premises, would yield correct conclusions. A good example is the following: “Pigeons are birds; all birds can fly; therefore, pigeons can fly.” These laws work as a representation of the thought process and led to the creation of a new field called logic. Thousands of years later, logic is still a field of substantial interest and advances in it led to the creation of programs (based on the logical notion) that could solve any solvable problem.\[22\].

(C) Acting Humanly

During World War II, the British secret services hired multiple cryptographers to crack the code used by Nazi Germany in military communications. One of the cryptographers was called Alan Turing. Along with his team, he was able to break the code and save millions of lives. Years later, Turing proposed a test designed to represent a definition of intelligence. The test was called the “Turing Test.” A computer and a human interrogator compose the test. During the test, the human writes questions to the computer, and if the human cannot tell whether the written answers come from a computer or another person, it determines that the machine passes the intelligence test.

To this extent, the machine needs some capabilities:

\[1\] Given what it knows, a system is rational if it can reason and does “the right thing.”
• **Natural Language Processing** to communicate with the human.

• **Knowledge Representation** to store what it “learns” and what it “knows”.

• **Automated Reasoning** to put into practice the information to answer the questions asked and to draw conclusions.

• **Machine Learning** to adjust to new conditions and to find patterns.

Years later, Steve Harnad proposed the “Total Turing Test” which is an upgraded version of the original test with two new capabilities [23]:

• **Computer Vision** to perceive objects.

• **Robotics** for object manipulation.

With this improved version, the interrogator can experiment perceptual capabilities as it can pass objects, giving it more resemblance to a human.

The capabilities outlined in this subsection comprise most of the foundations of Artificial Intelligence [22].

(D) Acting Rationally

Computer agents are a particular sort of programs that work toward achieving specific goals. To fulfill these goals, the agents need to sense, adapt and act upon their environment as it changes. However, to act upon the environment, the agents require some reasoning process that allows them to understand what is the most logical move to perform. For that, the agents incorporate some of the abilities mentioned above in the Turing Tests. With these elements, the agents contain enough expertise to produce decisions and ultimately accomplish their goals [22].

Now that the foundations of AI were introduced, it is possible to get to know the different branches in which it divides itself. AI separates into very distinct categories: Symbolic, Deep Structured Learning, Bayesian Networks and Evolutionary Algorithms.

In this subchapter, all these categories are introduced and described, with particular attention to Evolutionary Algorithms.

(A) Symbolic Approach

The symbolic approach, based on the creation of systems, relies on production rules. These rules are arrangements of symbols that create “if-then” statements that allow outputting some answers to the user. Examples of these types of machines are prevalent in medical areas, where expert systems are an excellent way to help physicians in their job. To conceive this type of systems, a knowledge base of patients must be created in order to engineer these rules, so that future patients can have a better and faster diagnose [24].
(B) Deep Learning

Deep Learning is part of a big family of machine learning systems. These types of systems rely on learning representations or patterns during their training time so that in their test time, they can recognize the same structures in other datasets.

When we were younger, our parents taught us to recognize what a chair was, but for us to understand, they did not give us the specific measurements of it. They showed us one and called its name each time we saw one. After multiple times sitting in chairs, we would start saying the word “chair”, and we would get complimented when we did recognize one. However, when we said that a chair was a table, our parents would correct us so that we would think again if our reasoning was indeed correct. This reinforcement is one example of what happens in this type of algorithms. They learn from experience and train with models that tell them if their guesses are correct, or incorrect, in order to create a more in-depth understanding of the trained matter [25].

(C) Bayesian Networks

Bayesian Networks are probabilistic models that try to classify patterns. These types of networks are capable of performing classification by assigning labels to instances represented by a set of properties. The Bayesian Networks compute the probabilities of attributes of a given instance and then try to predict their respective class based on the highest probability. These networks usually use graphic models to represent their variables, allowing the researcher to understand what was the reasoning used by the network easily [26].

(D) Evolutionary Algorithms

Evolutionary Algorithms’ underlying idea is to follow the concept of evolution. In these algorithms, a population is generated in a particular environment with a limited amount of supplies. The competition to get those resources creates a natural selection that, in turn, produces an increase in population fitness. Given an optimization problem, multiple candidate solutions are generated and evaluated under a specific quality function. This function’s job (fitness function) is to evaluate the candidates.

After the evaluation, the algorithm ranks the solutions from best to worst so that the best-ranked individuals get to breed and create the next generation. In order to create the new generation, variation operators (reproduction/mutation) are applied to the best candidates. After breeding, the offspring is evaluated with the fitness function and subsequently competes with the other solutions for selection. This type of mechanisms allows the species to evolve in the direction of the best individual and also to have some randomness associated that enables new kinds of features to be tested in the environment. This process repeats itself until the best solution is attained or until a stop criteria set previously is reached [27,28].
The goal of Evolutionary Algorithms is to achieve the best solution possible. For that, the algorithm uses variation operators to create diversity within the population and a selection process to act as pressure as well as to enhance solutions quality. These two processes combined lead to the progression of species and consequently, to an increase in the fitness of the population on each generation (optimization) [27,28].

To better understand how each element of the algorithm works, let us now take an in-depth look at each one of them in detail.

(i) **Representation**

The first step in designing an Evolutionary Algorithm is deciding how to represent the solutions to the problem. In several cases, there is a variety of possibilities, but getting the right representation might be one of the most challenging parts of designing an Evolutionary Algorithm.

Each solution is composed of multiple genes which form a chromosome (candidate solution/individual). A population is an aggregate number of chromosomes that may be encoded in various ways: integers, binary, real-valued or others [27,28]. Figure 2.6 contains examples of a binary 2.6(a) and real-valued 2.6(b) representations.

![Example of Binary Representation](image1.png)

(a) Example of Binary Representation

![Example of Real-Valued Representation](image2.png)

(b) Example of Real-Valued Representation

**Figure 2.6:** Two Examples of Representations.

(ii) **Fitness Function**

The fitness function operates by presenting the requirements that the population needs to adapt, to achieve the best solution. Besides, this function evaluates individuals according to their performance in the proposed problem, allowing to sort skilled individuals from not so skilled. It is this function that defines improvement, assigns the quality measure to the genes, and also determines the problem to solve [27,28].

(iii) **Selection**

The selection step relies on choosing, based on fitness, the best individuals to become parents of the following generation. An individual is regarded as a parent if it is selected to undergo some variation to create the new offspring.

In Evolutionary Algorithms, the parent selection is typically a stochastic process. Meaning this that although the best individuals have more probability of being chosen as a parent, the candidates with worse fitness may also be selected.
The selection process has multiple variants. Below some of the most common are described:

- **Tournament Selection**
  A random sample from the population is chosen. From this sample, the individual with the highest fitness gets selected. The process runs until the pretended number of individuals is achieved.

- **Roulette Wheel Selection**
  Each individual has the probability of becoming a parent proportional to its fitness and the sum of the fitness of all the candidates. Meaning this that if the sum of the fitness of all individuals is 500, then an individual with the fitness of 50 will have a probability to become a parent of 50/500 = 10%. The process runs until the pretended number of individuals is achieved.

- **Truncation Selection**
  Select a subset of the population with the highest fitness individuals and elect them to be the parents.

- **Stochastic Universal Sampling**
  Very similar to the roulette wheel, however, in this case, instead of only one point of selection, there are $N$ equally spaced marks (where $N$ is the number of parents to select). A random number is generated, and each value in the mark position gets to be selected as a parent. Moreover, in this case, just one iteration selects all the parents pretended [27–29].

(iv) **Variation Operators**

In order to create new individuals, parents need to undergo some procedures. Those are called the variation operators and divide themselves into two categories: recombination and mutation. Below the two operators are described in detail:

- **Crossover/Recombination**
  This operator bases itself on how sexual reproduction works. Two parents merge their genes and create one or two offsprings. The theory behind this operator relies on the idea that by mating two parents with desirable characteristics, their offspring may combine both features and create even better individuals.

  Recombination has multiple variations. The most commonly used are described below:

  - **One-Point Crossover**
    This operator starts by generating a random number between 1 and the length of the individuals to crossover. The random value generated acts as a “cutting” point to slice the chromosomes and exchange the genetic material [27,28]. Figure 2.7 presents an example of this type of crossover.
– N-Point Crossover
This operator behaves like the One-Point Crossover, but in this case, there can be more than one point of crossover to select in order to exchange genetic material [27, 28]. Below an example of this type of crossover is presented in figure 2.8.

Figure 2.8: Example of N-point crossover with $n = 2$.

– Uniform Crossover
This operator works by creating an empty individual with the same size as the parents. This individual has a random value in each gene that, if below or above a specific threshold $p$, will inherit the original gene from the first parent or the second parent [27, 28]. Figure 2.9 presents an example of this type of crossover.

Figure 2.9: Example of Uniform Crossover. The array $[0.4, 0.6, 0.3, 0.8, 0.9, 0.1]$ of random numbers and $p = 0.5$ were used to decide inheritance for this example.

• Mutation
This operator implements a slight random change to the genes of the offspring. The mutation is the result of a random process that is applied according to the probability of mutation. The probability of mutation is the parameter that decides if a specific gene can be mutated or not. In some cases, this parameter is set before the program starts while in other cases it changes during the evolution.
Although there are multiple designs of mutation, the most well-known involves generating a random value to each gene of the individual, taking into account the probability of mutation [27, 28]. Below an example of mutation is presented in figure 2.10.
(v) **Replacement**

The role of replacement is to introduce the offspring to the old population. The choice may vary from choosing the worst candidates to be replaced with the offspring or to replace the oldest elements with the offspring.

A commonly used method of replacement is elitism. This scheme has the intent of keeping the best solutions from previous iterations. Thus the fittest element is always kept for the next generation. When elitism is in place, the next generation will have, in the worst case the same fitness as the previous generation [27, 28].

(vi) **Initialization**

Several randomly generated individuals commonly characterize the beginning of the evolutionary process. By generating the right amount of candidates, the algorithm “realizes” the best path to follow in order to reach the solution [27, 28].

(vii) **Termination Condition**

There are two ways of completing the evolutionary process. The first takes into account the problem possessing a known solution. In that case, the Evolutionary Algorithm should stop only when it finds that solution.

The second method occurs when the problem has an unknown solution or the goal of the evolution is some maximization/minimization. In those cases, there is no solution to achieve, so the process would continue running forever. In order to stop the program, several conditions may serve as termination criteria [27, 28]. Some of those conditions could be:

- **Total Elapsed Time**
  The program stops after a specified amount of Central Process Unit (CPU) time.

- **Number of Generations**
  The evolution stops after a specific number of generations.

- **Population Diversity**
  The program stops after the population diversity falls under a particular threshold.

- **Fitness Stagnation**
  The evolution stops after the fitness has reached a low volatility point and does not improve for a specific number of generations.
Now that Evolutionary Algorithms have been introduced, it is possible to understand each element that takes part in them. Figure 2.11 displays an example for an EA architecture showing how each element is connected to the others and its flow.

![Architecture of an Evolutionary Algorithm](image)

**Figure 2.11:** Architecture of an Evolutionary Algorithm.

### 2.2 State Of the Art

In this section, multiple works in the financial market's prediction and Evolutionary Computation area are presented and explained. The same papers are then analyzed to retrieve the best ideas for future implementations. At the end of the section, a table 2.1 is presented, featuring the different algorithms and results obtained in the works introduced.

#### 2.2.1 Works on Financial Markets Prediction

Based on the premise that a reasonable investor should sell stock near the top of a trend and buy it close to the bottom, an intelligent stock trading system is developed [6]. This system, based on technical analysis, uses parameters that are enhanced using genetic algorithms and a novel neural network (Echo State Network (ESN)) that attempt to predict, not only, the future stock price but also when a price trend will hit the peak or the bottom. Results show that the system has a margin of profit in the “bull market”, but it is in the “bear market” that it achieves the best results, contradicting SP500 and buy-and-hold on the losses by profiting in a down-trending market.

On another work, a data mining process to forecast the daily direction of the Standard & Poor’s 500 Exchange-Traded Fund (SPY) return, based on 60 financial and economic features, is created [30]. Principal Component Analysis (PCA), Fuzzy Robust Principal Component Analysis (FRPCA) and Kernel-Based Principal Component Analysis (KPCA) are used to rearrange the original data structure and as a technique to reduce dimensional data while trying to minimize information loss. An Artificial Neural Network (ANN) manages to classify the datasets and forecast the daily direction of future market returns. The results show that combining ANN with PCA produces slightly better results in price pre-
diction than combining them with FRPCA and KPCA. It is also possible to conclude that data collection and preprocessing is an important process that can help improve performance while decreasing the complexity of the mining procedures.

As a topic with much research in the most recent years, a new system with multiple experiments to investigate the forecasting power of NN in financial markets is designed [31]. Five types of experiments are developed in this work. The first three based on, respectively, the previous one, two and three years of data to predict the classification of the following year. The fourth experiment is similar to the third but also includes three years of financial and macroeconomic data. Finally, the fifth experiment is created to compare the performance of the neural networks and also the rules extracted using the Generalized Analytic Rule Extraction (GLARE) algorithm. The results show that the average returns from NN are above the overall market average. Furthermore, are able to outperform the recommended strategy of divided and balanced portfolios but are not effective outperforming the top one-third returns. The fifth experiment was able to present some potential as it achieved higher average returns than NN (53% better) and showed that it could approximate itself with the best performers in the market.

In order to predict the highest and lowest stock prices of each day, an ANN using a Multilayer Feed-Forward Neural Network (MLP) trained by back-propagation is implemented [32]. It is additionally built a day-trading framework capable of guiding the user into buying and selling stocks according to the output of the ANN. The results show that the trading system can double the initial capital of the investor in the tested period, showing that it is a system with potential within the dataset presented.

Based on the assumption that historical data from stocks and markets gives enough information about the future market performance, a new approach is developed to automatically manage a portfolio using a Genetic Algorithm (GA) conjugated with technical analysis [4]. The system implements a set of technical indicators to analyze the data prices present in the dataset and, based on that analysis, decides the type of approach (long, short, closing) that the stock should adopt. The results prove that this method is successful, and the combination between the technical indicators and the GA can be very fruitful.

On another study, a Multi-Objective Genetic Algorithm (MOEA) to create and manage a stock portfolio by implementing a fundamental and technical approach is proposed [5]. In this work, fundamental ratios are utilized to compare companies inside the same industry and to draw conclusions about the best corporations to invest. The technical indicators are used to help the algorithm select the markets to invest, the entry and market timing, the closing of a winning position, position size, and the number of stocks to compose the portfolio. The results show that the increase in the number of fundamental indicators make a positive impact on the precision of the simulation results, and also in the MOEA.

A technical strategy optimizer that combines the VIX indicator with a MOEA to predict the future tendency of assets price is developed in a different paper [7]. In this work, it is also conceived a multi-
objective system as in Silva et al. (2015) in which technical indicators are optimized, with return and risk as a metric, to find the best investment strategy [5]. The results show that the system outperforms market indexes and reduce investment risk. It is also observable that the returns increase when the risk rises, and that the VIX indicator avoids several falls in the stock market by reducing the negative turns.

In a 2013 essay, an intelligent system to predict market stock prices, applying an Artificial Bee Colony Algorithm (ABC), a selection of past values (lags), Nearest Neighbor Classification (KNN), and the Adaptive Classification Nearest Neighbor (A-KNN) was proposed [33]. On the one hand, the ABC algorithm is adopted to discover the best set of lags (k-value in KNN), the percentage of stop-loss, stop gain and to determine the best configuration for the search vector. On the other hand, the KNN and A-KNN indicate the operation to perform (BUY, SELL, KEEP). During the search component process, the ABC algorithm uses the KNN or A-KNN to evaluate the fitness function, which is set to maximize the profit. The experiments employ and test, both the Nearest Neighbor Classification using Manhattan distance (KNN-M) and Adaptive Classification Nearest Neighbor using Manhattan distance (A-KNN-M) as the Nearest Neighbor Classification using Euclidean distance (KNN-E) and Adaptive Classification Nearest Neighbor using Euclidean distance (A-KNN-E), using fifteen stocks from the São Paulo Stock Market (BMFBOVESPA). The results show that the A-KNN-M algorithm is the one that offers better results, achieving good profit results and getting better returns in 11 of the 15 stocks tested.

In another paper Generalized Autoregressive Conditional Heteroscedasticity (GARCH) models with the intent of predicting market volatility in the SP500 index are elaborated [34]. In order to enhance the forecasting capability of the models, a hybrid approach is proposed combining the classic models with NN. The results show that both hybrid systems developed, outperform the classic Exponential Generalized Autoregressive Conditional Heteroscedasticity (EGARCH) models in volatility prediction.

Since the beginning of stock trading that investors look for a method to get higher and persistent returns. From that point forward, technical and fundamental indicators have been the most widely used tools to predict the markets. A hybrid approach is proposed using F-Score (2.8) and G-Score along with a momentum strategy incorporated with past technical information [35]. The results show that the hybrid system outperforms the moment strategy with higher returns.

2.2.2 Works on Evolutionary Algorithms

Genetic algorithms are a subtype of Evolutionary Algorithms used in optimization problems based on the theory of evolution and natural selection. With this in mind, GA seems a particularly exciting methodology to apply in the most different areas. Genetic algorithms are proposed to improve technical analysis by generating a combination of parameters that identify the optimal reversal points of financial trends [6]. By using GA as the enhancer of the technical indicators and combining it with ESN, this work brings a new perspective on how it is possible to use genetic algorithms on stock trading systems.
In a typical GA, it is common to create a “study population” to measure its performance on doing specific tasks and to comprehend how the evolutionary steps result in creating a better population. With that as a foundation, random types of individuals called “classifier equations”, composed of an array of weights that symbolize the value given to each technical rule, are created to manage a financial portfolio by optimizing its technical indicators [4]. The equation also assigns the score that an asset needs to adopt (long, short or closing position) in the portfolio. In order to estimate performance, the Return on Investment (ROI) function evaluates the efficiency of different investments during a specific range of time.

Usually, each candidate solution is assigned to a variety of chromosomes to stimulate the creation of unique individuals, providing more complete results. This work utilizes three different chromosomes to represent investment models with real constraints similar to the ones encountered by portfolio managers [5]. Each individual of the population is composed of a sequence of values (genes). The chromosome divides itself into two parts. The first group is the financial ratio weights, and the second is the trading parameters. Three types of chromosomes are used to have more broad results and to explore more on the GA approach. The algorithms used have two objectives: the Return and the Variance of the Returns. The system chooses the weights for each fundamental indicator and determines the values of the technical indicators. In this study, fitness is calculated using accumulated return and variance. Besides that, the algorithm selects the chromosomes for reproduction and applies the methods of crossover and mutation to create new elements.

In systems where the goal is optimizing real-world applications, the fitness function is the most critical measure of the implementation [36]. The fitness evaluation process focuses on simulating the performance of each trading individual in the evolving population and calculating the corresponding return related to the risk [7].

In another work, a new approach proposes a combination of an Evolutionary Artificial Bee Colony Algorithm (EABC) with Penalty Guided Support Vector Machines (PGSVM) in order to generate predictions more effectively [8]. In this system, the EABC algorithm uses a population of bees who are given velocity and flying direction in order to optimize the PGSVM. The results show that the EABC-PGSVM method outperforms other comparable methods saving CPU time and also enhances the hit ratio. Given the efficiency and potential for forecasting classification presented by the algorithm, the authors propose that the system may be applied in other fields.

### 2.2.3 Analysis of the Related Work

The works presented show that there is much potential in the ideas introduced, and those who showed to be fruitful should be applied in the future.

In Xiaowei Lin et al. (2011) work, two main ideas might be interesting to use in future work [6]. The
first one is using the GA as the technical indicators value enhancer. It is essential that after choosing the technical indicators, they get value tweaked to get the best out of them. Using GA to optimize the values assures that the conditions used are closest to the most fitting. The second idea that might be interesting to use is the RSI implementation. Usually, RSI implementation uses only the horizontal limits, which allow understanding if a stock is overbought or oversold, however, in this work, the authors draw oblique lines when the stock rises or falls under those limits. This approach may lead to a more precise prediction of when a stock is overbought or oversold.

Although there are multiple compelling aspects of Gorgulho et al. (2011) work, there are two which deserve special attention [4]. The first one is the application of the TI OBV, which seems a very reliable momentum indicator. The rising volume in stock might indicate the presence of “smart money” and the future rising of the price. The second idea to underline is the approach implemented to combat the restrictions over the different weights assigned to stipulate the technical rules. To combat it, a simulated artificial immune system takes into account information of individuals that would be discarded for not being the best. The algorithm maintains those individuals (antibodies) and makes them evolve in the same way as the main population in order to search more effectively. This idea of creating the “immune system” is a brilliant way to monitor discarded individuals that, in the future, can reveal exciting perks to help the search.

The cardinality constraint in a portfolio relates to the number of stocks that a portfolio should include [5]. In this study, 20 stocks were chosen as the best number for a portfolio to have, based on studies that report that a diverse portfolio of stocks is an excellent way to reduce risk [37]. A good idea to take from these is to develop a system that can optimize the best number of stocks to have in a portfolio while having it diverse and with low risk.

When the risk rate increases, the returns disperse in positive and negative values. However, after the introduction of the VIX indicator, the negative values disappear [7]. As a consequence, the VIX indicator enables the algorithm to exit the market when volatility increases, which results in a reduction of losses. VIX indicator is the example of an attractive strategy that may be implemented in future work, as it demonstrates value and potential in helping the trading simulator prevent losing money.

Although the results from Martinez et al. (2009) exhibit good returns (duplicating the investor’s capital) is critical to notice that the work is only using as its dataset two stocks of the BMFBOVESPA (Petróleo Brasileiro S.A. (PETR4) and Vale S.A. (VALE5)) [32]. In works similar to the one presented, it is common to use a larger dataset containing more stocks to perform a complete validation of the algorithms developed.

In another work, the authors outline some important conclusions that should be considered [31]. The first one is that integrating technical analysis with fundamental analysis can improve the return level. The second one is that including macroeconomic variables does not improve NN performance.
The third conclusion is concerning the rules extracted from the NN that shed some light on relevant information for return prediction. Common shares traded should be very low or very high, the capital expenditure must be medium, and RSI must be high or very high. This type of information should be taken into account in future works.

The Piotrowski F-Score \(2.8\) has proved to work separating winners from losers in value stock \([35]\). However, the same does not happen with low book-to-market ratio stocks \([38]\). For this reason, a new signal is proposed to analyze the fundamental of growth stocks, the G-Score. Similar to F-Score \(2.8\), G-Score comprises on the sum of eight fundamental signals. With that in mind, this indicator is a significant metric to use in future works to complement the F-Score \(2.8\) in the fundamental analysis of a company.

Below, the table 2.1 featuring the different algorithms and results obtained in the works introduced is presented.
<table>
<thead>
<tr>
<th>Model</th>
<th>Dataset</th>
<th>Feature</th>
<th>Where</th>
<th>Method</th>
<th>Accuracy</th>
<th>Best Results</th>
<th>Achieved</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>KPCA + LDA</td>
<td>Breast Cancer</td>
<td>80%</td>
<td>70/30</td>
<td>Leave One Out Cross Validation</td>
<td>95.1%</td>
<td>96.5%</td>
<td>2017</td>
<td></td>
</tr>
<tr>
<td>SVM with RBF kernel</td>
<td>Breast Cancer</td>
<td>80%</td>
<td>70/30</td>
<td>Leave One Out Cross Validation</td>
<td>94.5%</td>
<td>95.5%</td>
<td>2017</td>
<td></td>
</tr>
<tr>
<td>RF</td>
<td>Breast Cancer</td>
<td>80%</td>
<td>70/30</td>
<td>Leave One Out Cross Validation</td>
<td>94.3%</td>
<td>95.3%</td>
<td>2017</td>
<td></td>
</tr>
<tr>
<td>XGBoost</td>
<td>Breast Cancer</td>
<td>80%</td>
<td>70/30</td>
<td>Leave One Out Cross Validation</td>
<td>94.2%</td>
<td>95.2%</td>
<td>2017</td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>Breast Cancer</td>
<td>80%</td>
<td>70/30</td>
<td>Leave One Out Cross Validation</td>
<td>94.1%</td>
<td>95.1%</td>
<td>2017</td>
<td></td>
</tr>
</tbody>
</table>

**Table 2.1: Related Work Table.**
3 Architecture

Contents

3.1 System Overview ................................................. 35
3.2 Architecture ......................................................... 35
3.3 Flow Description .................................................... 36
3.4 User Layer ............................................................ 38
3.5 Data Preparation ..................................................... 39
3.6 Evolutionary Algorithm .......................................... 51
3.7 Portfolio Creator .................................................... 55
3.8 Trading Simulator .................................................. 57
3.9 Statistics .............................................................. 59
This chapter outlines the implemented architecture of this work. It starts with an overview of the system, followed by a workflow description and finalizes with an in-depth look at each solution module.

### 3.1 System Overview

The designed system attends two significant problems: The selection of good fundamental companies for portfolio composition and the picking of the best time to take positions in the market. The approach to solving the given problems is based on Artificial Intelligence, more specifically in the Evolutionary Algorithms branch. On the one hand, the evolutionary process selects the best companies for the portfolio by providing the most suitable weights to give the appropriate importance to each ratio of the F-Score (2.8). On the other hand, it provides weights for the technical indicators in order to change their importance, and to get better timing decisions according to how the market is behaving. This work proposes four new methods in order to maximize returns:

- Combine fundamental analysis with Evolutionary Algorithms in order to create good portfolios.
- Select the best timing to take positions in the market by combining technical indicators and Evolutionary Algorithms.
- Self-improve the evolutionary process by including the variation operators probabilities in the chromosomes.
- Use a dynamic portfolio to have the best companies to trade in each year.

### 3.2 Architecture

The designed architecture was developed to create a modular, optimized, versatile, scalable and reusable system. To accomplish that, the system was developed in Python\(^1\) for being a broadly-used programming language with numerous financial related libraries available, and the right amount of documentation to comprehend it [39].

Autonomous financial trading combined with Evolutionary Algorithms is a computationally laborious process. For that reason, the whole system was developed with particular attention to efficiency. Multiple functions, solutions and processes were tested against each other in order to reduce running times. Libraries such as NumPy\(^2\) and LineProfiler\(^3\) were used to make matrix operations and code profiling.

---

1. Python is an interpreted, high-level programming language. This coding language is utilized in multiple segments for its fast and powerful capabilities.
2. NumPy is a Python package that comprehends multiple scientific functions.
3. LineProfiler is a Python library that gives a detailed breakdown of execution times. With that information, it is possible to identify bottlenecks in the programs.
enabling them to reduce operating time [40, 41].

To accomplish the goals of the proposed system, it was necessary to design a framework that could handle all the steps required. The various steps are divided into different modules, and each module has a specific task that needs to achieve so that the system can be fully functional. In 3.1 it is possible to see the full system architecture, the various modules that constitute it and its data flow.

In order to better understand the architecture 3.1, a system flow diagram was created 3.2. The flow chart represents the interactions between modules and the way the developed system works.

There are two essential points to know before starting the system flow description: The colors of each module presented correspond to the colors of the architecture module to whom they belong. Moreover, the data inputted by the user, as well as the datasets created, are distributed throughout each module that requires them. All modules use the datasets created except the User Layer Module and Evolutionary Algorithm Module. Also, all the modules utilize the user preferences except the Data Preparation Module.

### 3.3 Flow Description

The data flow will now be described step by step:

1. The user starts by defining the configuration settings of the system variables. These variables range from: The number of generations to run the Evolutionary Algorithm, the percentage of the population to take part in elitism, the type of investment strategy to use and even the maximum portfolio size. The user can choose many more settings that are described in the User Module subsection below.

2. After the user has defined the configurations of the system, the data preparation takes place. The Data Preparation Module downloads the datasets from its sources.
3. After downloading takes place, the module inspects, cleans and validates the datasets in order to guarantee data quality.

4. Finally, the Data Preparation Module calculates the fundamental ratios and the technical indicators to use in the portfolio composition and in the trading environment.

5. Following, the cleaned data is fed to the Evolutionary Algorithm Module. In this step, the initial population is generated and sent to the Portfolio Creator Module.

6. In the Portfolio Creator Module, the portfolio is arranged, taking into account the fundamental part of the chromosomes generated to work as weights and the financial datasets used to calculate the F-score (2.8). After all the calculations have been made, the companies get ranked, and the best ones serve as a portfolio. After the portfolio creation, both the chromosomes and the portfolio get sent to the Trading Simulator Module.

7. In the Trading Simulator Module, the candidate solutions get evaluated. During the evaluation, the simulator uses the technical part of the chromosomes combined with the technical indicators.
to choose the best positions and timings for each company to trade. If the investment type is a buy-and-hold, then the algorithm buys all the companies on the first day and sells them all on the last day of trading.

8. At the end of each year, the simulator sells all the remaining stocks. If dynamic F-Score is enabled, then a new portfolio is generated for the upcoming year (step 6). If it is disabled, the simulation maintains the same portfolio for the upcoming year and heads to step 7.

9. When the trading period finishes, the positions data frame is sent to the Statistics Module. In this module, the fitness of each candidate is calculated.

10. After the calculation finishes, the module calculates the statistics of the simulation performance and stores them.

11. If the just ended simulation was not a test, nor the last training generation, the fitness values are sent to the Evolutionary Algorithm Module.

12. In the EA module, the candidate solutions undergo a process of selection to pick the parents of the offspring, crossover to reproduce the new offspring and mutation to perform small variations in the offspring. When all these processes are complete, the new population replaces the old one. If the self-adaptive parameter is enabled, then the crossover and mutation operators use the evolutionary part of the chromosome to get the probabilities of each variation operator.

13. When all the Evolutionary Algorithm steps are complete, the new population is sent to the Portfolio Creator Module (step 6), where all the process will loop until the training ends.

14. If the just ended simulation was not a test, but it was the last generation of evolution (training finished), the best fitness individual is selected and used as a chromosome to make an out of sample test, going back to step 6.

15. If the just ended simulation was a test, then the results are sent to the User Layer Module, printed to the user, and the system finishes.

Now that the architecture and the system flow have been introduced, each module belonging to the designed system will be described in full detail.

### 3.4 User Layer

The User Layer is the first and last step of the system. It enables the user to introduce its settings to the system and is also responsible for printing results and statistics at the end of the system terminus. To
kickoff, the user can select and change the settings of the tests to run. In the developed architecture, the user can choose:

- The number of generations a population is allowed to evolve.
- The percentage of the population that takes part in elitism.
- The probability of mutation and crossover.
- The number of candidate solutions to generate at the beginning of the evolutionary process.
- If the portfolio should be static or dynamic.
- If the evolutionary process should self-adapt or not.
- The investment strategy to use.
- The maximum portfolio size.
- The type of sliding window to use.
- The number of times to run the system.

### 3.5 Data Preparation

The Data Preparation Module handles three main duties: Data Downloading, Data Processing and Data Manipulation. Data Downloading retrieves and stores multiple required datasets from various sources. Data processing inspects, cleans and verifies the data downloaded. Data Manipulation calculates financial ratios and technical indicators from the pre-processed data for posterior use.

#### 3.5.1 Data Downloading

There are three main data sources in this work: Center for Research in Security Prices (CRSP)/Compustat to download all the fundamental information (yearly and quarterly) from the organizations of the SP500 [42]. Yahoo Finance to download SP500 index prices and stock prices of each company having a place in the SP500 [43]. And lastly, Chicago Board Options Exchange (CBOE) to download the volatility index [44].

**A) CRSP/Compustat**

In order to download the data CRSP/Compustat user application was used. For these datasets, all the available information (yearly and quarterly) from organizations of the SP500 was retrieved.

The file containing information on the companies in quarters holds data from 2005-01-31 to 2018-12-31, while the record with the data of the companies in years contains information from 2005-10-31 to 2018-12-31. The quarterly file is made out of 39335 rows x 634 columns, while the yearly document
is composed of 11483 rows x 925 columns. The full rundown of columns is accessible in the CRSP documentation [45].

After the download was finished, the two “.csv” files were imported to Python using the data analysis library Pandas ⁴, and each was stored in a different data frame [46].

Table 3.1 and 3.2 represent a sample of the files downloaded after being stored in datasets.

### Table 3.1: Quarter Dataset Sample.

<table>
<thead>
<tr>
<th>Index</th>
<th>gvkey</th>
<th>datadate</th>
<th>fyearq</th>
<th>tic</th>
<th>...</th>
<th>xopry</th>
<th>xrdy</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1013</td>
<td>2005-01-31</td>
<td>2005</td>
<td>ADCT</td>
<td>...</td>
<td>221.1</td>
<td>15.2</td>
</tr>
<tr>
<td>1</td>
<td>1013</td>
<td>2005-04-30</td>
<td>2005</td>
<td>ADCT</td>
<td>...</td>
<td>459.3</td>
<td>33.4</td>
</tr>
<tr>
<td>2</td>
<td>1013</td>
<td>2005-07-31</td>
<td>2005</td>
<td>ADCT</td>
<td>...</td>
<td>716.2</td>
<td>51.8</td>
</tr>
<tr>
<td>3</td>
<td>1013</td>
<td>2005-10-31</td>
<td>2005</td>
<td>ADCT</td>
<td>...</td>
<td>963.6</td>
<td>71.6</td>
</tr>
<tr>
<td>4</td>
<td>1013</td>
<td>2006-01-31</td>
<td>2006</td>
<td>ADCT</td>
<td>...</td>
<td>254.9</td>
<td>19.0</td>
</tr>
</tbody>
</table>

### Table 3.2: Annual Dataset Sample.

<table>
<thead>
<tr>
<th>Index</th>
<th>gvkey</th>
<th>datadate</th>
<th>fyear</th>
<th>tic</th>
<th>...</th>
<th>auop</th>
<th>au</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1045</td>
<td>2005-12-31</td>
<td>2005</td>
<td>AAL</td>
<td>...</td>
<td>1.0</td>
<td>4.0</td>
</tr>
<tr>
<td>1</td>
<td>1045</td>
<td>2006-12-31</td>
<td>2006</td>
<td>AAL</td>
<td>...</td>
<td>4.0</td>
<td>4.0</td>
</tr>
<tr>
<td>2</td>
<td>1045</td>
<td>2007-12-31</td>
<td>2007</td>
<td>AAL</td>
<td>...</td>
<td>4.0</td>
<td>4.0</td>
</tr>
<tr>
<td>3</td>
<td>1045</td>
<td>2008-12-31</td>
<td>2008</td>
<td>AAL</td>
<td>...</td>
<td>1.0</td>
<td>4.0</td>
</tr>
<tr>
<td>4</td>
<td>1045</td>
<td>2009-12-31</td>
<td>2009</td>
<td>AAL</td>
<td>...</td>
<td>1.0</td>
<td>4.0</td>
</tr>
</tbody>
</table>

(B) Yahoo Finance

So as to download data from Yahoo Finance, a Python library denominated “Yahoo! Finance market data downloader” was utilized [47]. From this framework, two documents were obtained containing the stock prices of companies in SP500 and the SP500 index prices.

On the one side, the file with the stock prices holds information from 2005-01-03 to 2018-12-31 and is composed of 3522 rows x 3840 columns. On the other side, the index file holds data from 2005-01-03 to 2018-12-28 and is composed of 3522 rows x 7 columns. Both files are time-series ⁵ with equally spaced points in time. In the case of the documents downloaded, the space between points represents a day.

The file with the index prices contains five different columns:

- **Open** - Price of the asset at market opening time.
- **High** - Highest price the asset achieved in that day.
- **Low** - Lowest price the asset attained in that day.
- **Close** - Price of the asset at market closing time.

---

⁴Pandas is an open-source library, that provides high-performance data analysis tools and data structures, for the Python programming language.

⁵A time series is a sequence of numerical data points in consecutive order.
- **Adj. Close** - Price of the asset at market closing time after adjustments for all splits and dividends.

- **Volume** - Number of shares traded during that day.

Similarly, the stock prices file has an identical structure, however, each first level column subdivides itself into a tremendous number of sub-columns representing each company listed in SP500. In both files, each entry represents a market day.

Table 3.3 and table 3.4 represent a sample of each file after being downloaded and stored in Pandas data frames.

**Table 3.3: SP500 Dataset Sample.**

<table>
<thead>
<tr>
<th>Index</th>
<th>Date</th>
<th>Open</th>
<th>High</th>
<th>Low</th>
<th>Close</th>
<th>Adj Close</th>
<th>Volume</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2005-01-03</td>
<td>1211.92</td>
<td>1217.80</td>
<td>1200.31</td>
<td>1202.07</td>
<td>1202.07</td>
<td>15108</td>
</tr>
<tr>
<td>1</td>
<td>2005-01-04</td>
<td>1202.07</td>
<td>1205.83</td>
<td>1185.39</td>
<td>1188.05</td>
<td>1188.05</td>
<td>17210</td>
</tr>
<tr>
<td>2</td>
<td>2005-01-05</td>
<td>1188.05</td>
<td>1192.72</td>
<td>1183.71</td>
<td>1183.73</td>
<td>1183.73</td>
<td>17389</td>
</tr>
<tr>
<td>3</td>
<td>2005-01-06</td>
<td>1183.73</td>
<td>1191.63</td>
<td>1183.27</td>
<td>1187.89</td>
<td>1187.89</td>
<td>15691</td>
</tr>
<tr>
<td>4</td>
<td>2005-01-07</td>
<td>1187.89</td>
<td>1192.19</td>
<td>1182.16</td>
<td>1186.18</td>
<td>1186.18</td>
<td>14779</td>
</tr>
</tbody>
</table>

**Table 3.4: Stock Prices Dataset Sample.**

<table>
<thead>
<tr>
<th>Date</th>
<th>Open</th>
<th>...</th>
<th>Volume</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>A</td>
<td>AABA</td>
<td>AAP</td>
</tr>
<tr>
<td>2005-01-03</td>
<td>17.23</td>
<td>38.36</td>
<td>29.33</td>
</tr>
<tr>
<td>2005-01-04</td>
<td>17.01</td>
<td>38.45</td>
<td>29.10</td>
</tr>
<tr>
<td>2005-01-05</td>
<td>16.57</td>
<td>36.68</td>
<td>28.83</td>
</tr>
<tr>
<td>2005-01-06</td>
<td>16.73</td>
<td>36.32</td>
<td>29.13</td>
</tr>
<tr>
<td>2005-01-07</td>
<td>16.22</td>
<td>35.99</td>
<td>29.06</td>
</tr>
</tbody>
</table>

(C) CBOE

In order to download the VIX the CBOE website was accessed, and the “.csv” file was downloaded directly.

The file is composed of 3522 rows x 6 columns and contains data from 2005-01-03 to 2018-12-28. The structure is very identical to the Yahoo Finance files. Both files are daily time series, plus the columns are equivalent except the “Volume” that does not exist in the VIX file.

The table presented below 3.5 is a sample of the file after its download and storing with Pandas.

### 3.5.2 Data Processing

After the data acquisition step is complete, a pre-processing of the data is required. This step is characterized by the execution of several cleanings, validations, and transformations to guarantee data quality. For that, small scripts were designed using Python, Pandas, and Numpy.
Table 3.5: VIX Dataset Sample.

<table>
<thead>
<tr>
<th>Index</th>
<th>Date</th>
<th>Open</th>
<th>High</th>
<th>Low</th>
<th>Close</th>
<th>Adj Close</th>
</tr>
</thead>
</table>

The workflow of this step follows:

1. **Inspection**  
   To detect inconsistency or errors in data.

2. **Cleaning**  
   Fix the problems detected.

3. **Validation**  
   Verify if the detected problems were fixed.

This sequential procedure was run iteratively until the quality of the data was assured.

In the vast majority of the files, the methods applied were fundamentally the same. All the fields containing dates were converted into the same configuration "yyyy-mm-dd" so that multiple operations can be applied over them. All the elements of a column were parsed into the same type (integers, strings, dates, booleans or others) so that there are no inconsistencies. Finally, all the fields that were not going to be used were discarded. The files containing prices (Yahoo Finance and CBOE) got all its columns deleted except the “Close” column. In the fundamental files (CRSP/Compustat), the fields not required to calculate F-Score (2.8) were discarded.

Furthermore, specific types of files needed specific types of procedures:

**Fundamental Files:**

- Companies without the identifier ticker (GVKEY) or name (tic) were removed.
- Companies with only one identifier were forced to complete the missing field.

**Prices Files:**

- Companies with missing values or outliers in the time series were deleted.
- The date field was implemented as an index.
- **VIX dataset was merged with the stock prices dataset.**
Datasets were resampled from daily to weekly. Finally, after all the processing was made, a script was created to perform an intersection between the companies present in price files and fundamental datasets. The purpose relied on avoiding the creation of portfolios composed of companies that do not exist in the price files.

3.5.3 Data Manipulation

After all the datasets are cleaned and its data is wholly assured, its time to use them to calculate ratios to use later in the program. There are two types of indicators to calculate: Financial Indicators and Technical Indicators. The first ones will serve to create portfolios, while the second ones will help the algorithm know when to enter and leave the market.

(A) Financial Indicators Calculation

A detailed analysis of the fundamental datasets led to conclude that not all the ratios required to calculate the F-Score \[(2.8)\] were present. However, the variables needed to achieve those ratios were present in the datasets. For that, the formulas presented in the Fundamental Indicators subchapter (ii) were implemented to assess the values needed.

In the table below 3.6, it is possible to see how the fundamental annual dataset becomes after all the values are obtained. It is relevant to notice that the table 3.6 is just a transposed sample of the original with all the variables “translated” for better-viewing and understanding experience.

While most of the fields were presented in 2.1.3, others require an introduction:

- **Gvkey** - Unique six-digit number key assigned to each company.
- **Tic** - Name identifier present in stock listings.
- **Fiscal Year** - Fiscal year the report belongs to.
- **Date** - Date of the report submission.
- **Cost of Goods Sold** - Production cost of the goods sold by a company.
- **Total Assets BY** - Total assets at the beginning of the year. This field is actually from the quarterly file. However, it is a required field to calculate the F-score \[(2.8)\], so it was transferred to the annual file.
- **Revenue and Total Sales** - According to *GuruFocus*: “Turnover...It is calculated as Revenue divided by Total Assets.” consequently was assumed that Total Sales \[(2.12)\] and Revenue \[(2.11)\] represent the same field \[49\].

---

\[49\] *GuruFocus* is a value investing platform dedicated to research, publish and comment financial information \[48\].
Quanl "GuruFocus" was used to validate the new fields created to assure that there were no errors in the calculations or even in the formulas [48].

After all the calculations and validations were completed, the new fundamental dataset was ready to create portfolios using fundamental information.

(B) Technical Indicators Calculation

In order to identify the best opportunities to trade, various technical indicators were calculated. Together these indicators tell if it is a good time to open, maintain or close a position.

So as to settle these decisions, the indicators were calculated for all the companies using TA-Lib, the “Close” field of the stock prices dataset and the appropriate periods for each TI [50]. Subsequently, specific rules were appointed to each TI to form a score. The score varies from Very Low Score (-1), Low Score (-0.5), Neutral (0), High Score (0.5), and Very High Score (1).

The indicators were selected, taking into consideration the goals to achieve and their performance in works presented in subchapter 2.2. The five indicators picked were: RSI, Exponential Moving Average (EMA), Moving Average Crossover (MAC), Rate of Change (ROC), the MACD and the VIX [4, 7, 33]. Each of the indicators will now be described in full detail.

---

7 TA-Lib is a Python library used by trading software developers to perform technical analysis of the financial market.
(i) **Relative Strength Index**

RSI is a momentum indicator that measures the significance of a price change to evaluate if it is overbought or oversold. In this TI the period chosen was 14 days as it’s the most common value to use in this indicator [18]. The formula follows:

\[ RSI_t(n) = 100 - \frac{100}{1 + RS_t} \]  

(3.1)

and

\[ RS_t = \frac{AverageGain_t}{AverageLoss_t} \]  

(3.2)

\[ AverageGain_t = \frac{AverageGain_{t-1} \cdot (n - 1) + CurrentGain}{n} \]  

(3.3)

\[ AverageLoss_t = \frac{AverageLoss_{t-1} \cdot (n - 1) + CurrentLoss}{n} \]  

(3.4)

where \( t \) represents today, \( n \) depicts the period chosen, and the initial \( AverageGain \) and \( AverageLoss \) are simple averages over \( n \). The scores of the RSI match the following rules:

**Table 3.7**: RSI Scoring Rules.

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Low</td>
<td>RSI value is equal or above 70</td>
</tr>
<tr>
<td>Low</td>
<td>RSI value is between 30 and 70 but its decreasing</td>
</tr>
<tr>
<td>Neutral</td>
<td>RSI value did not change since last period</td>
</tr>
<tr>
<td>High</td>
<td>RSI value is between 30 and 70 but its increasing</td>
</tr>
<tr>
<td>Very High</td>
<td>RSI value is equal or below 30</td>
</tr>
</tbody>
</table>

An example of RSI and its score system is shown in figure 3.3 below:

(ii) **Exponential Moving Average**

A moving average is a trend following indicator that depicts the mean price over a particular time frame. Since the prices are smoother and clearer to read, they are used to discover patterns. The exponential moving average is a specific type of MA that is calculated by applying a percentage of the current day price to the previous period moving average value. Thus, EMA places more weight on more recent values giving them more significance. In this TI, the period picked was 12 days as it represents a fast MA to perceive sudden changes in the price value [18]. The formula follows:
\[ EMA_t(n) = Price_t \cdot k + EMA_{t-1}(n) \cdot (1 - k) \]  
\[ k = \frac{2}{n + 1} \]

where \( t \) represents today, \( n \) depicts the period chosen and \( EMA_t(n) \) is a simple average over \( n \).

The scores of the EMA match the following rules:

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Low</td>
<td>Stock Price was above EMA value and now is below EMA value</td>
</tr>
<tr>
<td>Low</td>
<td>EMA value is decreasing</td>
</tr>
<tr>
<td>Neutral</td>
<td>EMA value did not change since last period</td>
</tr>
<tr>
<td>High</td>
<td>EMA value is increasing</td>
</tr>
<tr>
<td>Very High</td>
<td>Stock Price was below EMA value and now is above EMA value</td>
</tr>
</tbody>
</table>

An example of EMA and its score system is shown in figure 3.4 below:

(iii) **Moving Average Crossovers**

The MAC is another implementation of moving averages. In this strategy, two exponential moving averages were implemented to find market trends. The first is a fast-moving average characterized by having a shorter period (5) and consequently more sensibility to market shifts. The second EMA is described by having a slower period (20), and therefore a slower but more reliable response to market changes.

The MAC is known for using the crossovers between the exponential moving averages to find patterns in the market [18].
The scores of the MAC match the following rules:

**Table 3.9: MAC Scoring Rules.**

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Low</td>
<td>Faster EMA crosses below Slower EMA</td>
</tr>
<tr>
<td>Low</td>
<td>Both EMA values decreasing</td>
</tr>
<tr>
<td>Neutral</td>
<td>Both EMA values did not change since last period</td>
</tr>
<tr>
<td>High</td>
<td>Both EMA values increasing</td>
</tr>
<tr>
<td>Very High</td>
<td>Faster EMA crosses above Slower EMA</td>
</tr>
</tbody>
</table>

An example of MAC and its score system is shown in figure 3.5 below:

(iv) **Rate of Change**

ROC is a momentum indicator that displays the difference between the current price and the price
The signal generated creates a curve estimating the percentage that prices have changed over time. Doing so provides the trader with a reliable method to measure how quickly the price is moving. Rapid growth or decline may indicate overbought or oversold conditions. The most popular periods for this TI are between 12 and 25 for short/intermediate trading. For that reason, the selected period was 13 \[18\]. The formula follows:

\[
ROC = \frac{Price_t - Price_{t-n}}{Price_{t-n}} \times 100
\]

(3.7)

where \(t\) represents today and \(n\) depicts the period chosen. The scores of the ROC match the following rules:

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Low</td>
<td>ROC value was above 0 and now is below</td>
</tr>
<tr>
<td>Low</td>
<td>ROC value increasing and Stock Price decreasing</td>
</tr>
<tr>
<td>Neutral</td>
<td>ROC value did not change since last period</td>
</tr>
<tr>
<td>High</td>
<td>ROC value decreasing and Stock Price increasing</td>
</tr>
<tr>
<td>Very High</td>
<td>ROC value was below 0 and now is above</td>
</tr>
</tbody>
</table>

An example of ROC and its score system is shown in figure 3.6 below:

![Figure 3.6: Rate of Change Example.](image)

(v) **Moving Average Convergence Divergence**

The MACD is a trend following indicator that outlines the relation between a faster line (MACD) and a slower line (signal). The faster line represents a difference between 12 and 26 periods of exponential moving averages while the slower line denotes a 9 period EMA.

A commonly used technique converts the MACD indicator into a histogram that plots the difference between the two MACD lines. The value of this visualization is to spot the spread between the two
lines (widening or narrowing). On the one hand, when the histogram is over zero and starts to fall, it denotes a weakening in the uptrend. On the other hand, when the histogram is below zero and starts to move upward, it expresses a loss of momentum in the downtrend [17, 18]. The formula follows:

\[
MACD_t(n_s, n_l) = EMA_t(n_s) - EMA_t(n_l)
\]  
\[
Signal_t(n_t) = EMA_t(n_r)
\]  
\[
MACD_{Histogram}(n_s, n_l) = MACD_t(n_s, n_l) - Signal(n_r)
\]  

where \( t \) represents today, \( n_s \) depicts the short period, \( n_l \) illustrates the long period and \( n_r \) represents the signal period. It is also important to notice that the values used for \( Signal(n_t) \) are the \( MACD_t(n_s, n_l) \) result. The scores of the MACD match the following rules:

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Low</td>
<td>MACD Histogram value was above 0 and now is below</td>
</tr>
<tr>
<td>Low</td>
<td>MACD Histogram value decreasing to -( \infty )</td>
</tr>
<tr>
<td>Neutral</td>
<td>MACD Histogram value did not change since last period</td>
</tr>
<tr>
<td>High</td>
<td>MACD Histogram value increasing to +( \infty )</td>
</tr>
<tr>
<td>Very High</td>
<td>MACD Histogram value was below 0 and now is above</td>
</tr>
</tbody>
</table>

An example of MACD and its score system is shown in figure 3.7 below:

(vi) **Volatility Index**

The VIX is a sentiment indicator that tracks the SP500 volatility. Also known as the “Fear Index”, measures the amount of optimism or fear in the market through price volatility.

A good example of VIX usefulness is the 2009 market crash. Before the crash, VIX values would range between 10 to 30 however, when the market crash was imminent, VIX value skyrocketed determining that something abnormal was happening.

In this work, the VIX indicator was used combined with two Simple Moving Average (SMA) to smooth the values and get more reliable signals. A faster SMA was used with a period of 50 while a slower SMA was implemented with a period of 200.

The moving averages crossovers denote important shifts in the mentality of the investors. On the one hand, the slower moving average crossing above the faster-moving average represents an improved sentiment. On the other hand, the faster moving average crossing above the slower moving average represents an unimproved market sentiment. The SMA formula follows:
Figure 3.7: Moving Average Convergence Divergence Example.

\[ SMA_t(n) = \frac{1}{n} \sum_{i=0}^{n-1} Price_{t-i} \] (3.11)

where \( t \) represents today and \( n \) depicts the period chosen. The scores of the VIX match the following rules:

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Low</td>
<td>Fast SMA is above Slow SMA</td>
</tr>
<tr>
<td>Low</td>
<td>Fast SMA is below but approaching Slow SMA</td>
</tr>
<tr>
<td>Neutral</td>
<td>Both SMA values did not change since last period</td>
</tr>
<tr>
<td>High</td>
<td>Slow SMA is below but approaching Fast SMA</td>
</tr>
<tr>
<td>Very High</td>
<td>Slow SMA is above Fast SMA</td>
</tr>
</tbody>
</table>

An example of VIX and its score system is shown in figure 3.8 below:

After all the technical indicators had been calculated, they were validated using TradingView\(^8\) and after joined to the stock prices dataset as shown in table A.1 [51].

\(^8\)TradingView is a financial visualization platform used by traders to view stock prices, create charts, find patterns or even to calculate technical indicators.
3.6 Evolutionary Algorithm

In this work, Evolutionary Algorithms were selected as the model of AI to implement. These types of algorithms were picked since the reviewed papers revealed that these are able to achieve promising results when applied to financial forecasting. Furthermore, the same analysis showed that EA are capable of handling the goals proposed for this work [4–8].

The Evolutionary Algorithm Module is the center of this work. This module handles the creation and evolution of the system weights in order to achieve the best solution possible. In this work, the EA was developed on top of a Python library called DEAP\(^9\) [52]. The framework was used as a groundwork, where multiple adaptations were implemented to achieve the proposed goals. In this subsection, all the EA constraints implemented are described as the parameters used in this work.

3.6.1 Representation

In this work, the representation was developed in a modular way. There are three types of representation modules: Fundamental, Technical and Self-Adaptive.

(A) Fundamental Representation

The Fundamental representation module is composed of nine values depicting the weights of each fundamental F-Score (2.8) ratio. In figure 3.9, a representation of the fundamental chromosome is presented. It is important to notice that the colors of the representation modules presented below, match the colors of the architecture modules 3.1 where their values are utilized.

\(^9\)DEAP is an Evolutionary Computation library for quick prototyping and concepts testing.
(B) Technical Representation
The Technical representation module is composed of six weights assigned for each technical indicator score created in the Data Module. In figure 3.10, a representation of the technical chromosome is presented.

![Figure 3.10: Technical Chromosome composed of the selected technical indicators.](image)

(C) Self-Adaptive Representation
The Self-Adaptive representation module is composed of two genes that dictate the probabilities of mutation $P_m$ and crossover $P_c$ variation operators parameters. This representation module was created considering one of the biggest problems in EA: Parameter tuning.

Building an executable EA instance requires stipulating values for its parameters. These values determine whether it will find an optimal solution and whether it will do so efficiently. Parameter tuning is a commonly practiced approach to algorithm design, where, in most cases, values for the parameters are defined prior to the beginning of the algorithm, staying fixed throughout the whole process.

The traditional method to choose the parameters is based on conventions such as “mutation probability should be low”, “population size should be 200” and limited experimentation with distinct values. For example, considering three parameters and four values for each of them.

Nevertheless, this type of parameter tuning is not feasible. Trying all different combinations systematically is extremely time-consuming. Besides, for numerical parameters, the optimal values could lie between the points we are testing or not even be among the ones selected.

This idea becomes even more discouraging if we are looking for a generally good setup that is able to operate well on a range of problems/scenarios [28].

In this work, the self-adaptive representation allows the EA to incorporate the variation parameter values in the candidate solutions. With this methodology, the EA evolves and adapts the variation operator parameters, hoping to achieve optimal values for efficient optimization. In figure 3.11, a representation of the self-adaptive chromosome is presented.

![Figure 3.11: Self-Adaptive Chromosome composed of the selected technical indicators.](image)

In this work, the representation modules were combined in order to achieve various types of representations. The outcome of the combinations led to the creation of four types of representations.
Figure 3.11: Self-Adaptive Chromosome composed of the probability of mutation $P_m$ and crossover $P_c$ for the variation operators.

The first representation is used for a fundamental (Buy and Hold) type of investment and is composed only of the fundamental representation module. In this type of chromosome, the weights are used combined with the fundamental ratios to create a portfolio to trade.

The second type of representation is used for a technical investment strategy and combines the fundamental with the technical representation module. This type of representation uses its fundamental weights and ratios to create a portfolio but also, the technical indicators scores and respective weights to make positions in the market.

The third and fourth types of representations used in this work are variations of the first and second presented, with the addition of the self-adaptive representation module, to add self-adaptive capabilities. These last types of representations are used for fundamental and technical investments, respectively.

3.6.2 Fitness Function

For the evolution process to occur an evaluation metric to distinguish the best solutions from the worst must exist.

In this work, the fitness function utilized was the Rate of Return (ROR). This metric is used to evaluate the gain/loss of some investment over time, expressed as a percentage of the investment initial cost. The higher the value of this metric, the best the individual performance is as it can obtain high returns on its investments. The formula follows:

$$RateOfReturn = \frac{GainFromInvestment - CostOfInvestment}{CostOfInvestment} \times 100$$

3.6.3 Selection

In order to choose individuals to become parents, a selection process needs to occur. In this work, tournament selection was assigned for this job.

Although selection processes are very similar due to their stochastic nature, the tournament selection has a specific parameter that enables easy control of the selection pressure. The parameter is called tournament size ($k$) and represents the number of individuals to be randomly selected to a pool where only the best individual will get selected. The larger $k$, the more significant the probability of selecting a
high-fitness member is. In this work, the value of \( k \) was picked as 70% since it makes enough pressure to get the best fitness individuals and at the same time, leaves enough “space” for lower fitness solutions that may evolve to something interesting.

### 3.6.4 Variation Operators

After the parents have been selected, variation operators are employed to generate the new offspring. In this subsection, they are described in full detail.

**(A) Crossover/Recombination**

Although there are multiple types of crossover procedures, the 2 point crossover was selected. The nature of the operators presented in the variation operators section (iv) makes it hard to state which performs the best. Nevertheless, it is possible to examine their weaknesses and choose the one that fits better the algorithm.

Two main weaknesses make N-Point Crossover standout: the positional bias and the distributional bias. The first states that when the \( n \) value is odd (one-point crossover), there is a strong bias against keeping together genes placed at opposing sides of the chromosome. The second one, the distributional bias (which uniform crossover suffers from), represents the cases in which a specific number of genes might be more likely to be transmitted than others during recombination [53, 54].

In this work, the selected probability of crossover \( (P_c) \) was 100% to assure there was a chance to create better individuals in each generation. In a self-adaptive scenario, the \( P_c \) is determined by the value in the chromosome.

**(B) Mutation**

Similarly, as in the crossover operator, the mutation operator also has multiple types of variations. In this work, the mutation selected works as follows: For each gene of the chromosome, a probability/value is generated randomly. If the value generated is lower than the mutation probability \( (P_m) \), then a new value is generated for this specific gene.

The mutation operator is a tool for random exploration of the space containing all possible solutions (search space). Random exploration is welcome in the evolutionary process, not only to try solutions that would not be tested but also to avoid early convergence. However, an evolutionary process containing much random exploration could turn out to be similar to a brute-force algorithm where the optimal solution would be found not by evolution but by testing all the candidates. Furthermore, using a high mutation rate could lead to frequent destruction of candidate solutions with high fitness performing genes.

For those reasons in this work, the \( P_m \) selected was 20% as it is a low but significant value to occur and create small changes in the individual. In a self-adaptive scenario, the \( P_m \) is determined by the value in the chromosome.
3.6.5 Replacement

After the variation operators have completed their job, it's time for the old population to be replaced with the offspring. In this work, the previous generation population is almost entirely replaced with the offspring. A small percentage of the new population is reserved for the elite of the old population. Elitism was implemented in order to maintain the best solutions from previous iterations. When an individual from the offspring is already in the elite, only one is added to the new population to increase diversity. The percentage of the population that represents the elite is 30% since it is already a fair share of the population, giving space for the new offspring to show its abilities.

3.6.6 Initialization

At the beginning of the Evolutionary Algorithm, a specific number of candidate solutions are randomly generated to create a highly diverse population. In this work, 100 candidate solutions were randomly generated at the beginning of the algorithm. This value was selected since it already represents a good amount of solutions for the algorithm to explore.

3.6.7 Termination Condition

The evolutionary problem presented does not have a known solution. Thus, the purpose of the algorithm is to maximize its returns. However, even after achieving the best solution, the algorithm will still search eternally for a superior solution, for that reason, a termination condition was chosen. The termination condition serves two purposes: the first is to alert the algorithm that it is time to end, and the second is that by creating the same conditions for each test, the algorithms/simulations can be compared between each other. In this work, the termination condition selected was the number of generations. After a certain number of generations, the algorithm stops the evolutionary process to test the best solution found. In this thesis, the selected value was 20 generations since it gives enough iterations for the population to experiment and evolve to better individuals.

3.7 Portfolio Creator

The Portfolio Creator Module is where the portfolio is constructed according to the year to trade, the weights generated in the Evolutionary Algorithm and the size of the portfolio desired. In this work, the portfolio size established was 20 companies since it's essential to diversify the capital into various investments to minimize the risk exposure. The portfolio is always created, taking into account the year before entering the market. This strategy allows choosing the best companies of the
year before, hoping that they will behave as well in the next year.

In order to create the portfolio, the F-Score (2.8), along with the fundamental datasets created in the Data Preparation Module, is used to score the financial ratios. The formula awards the ratios with binary values according to its rules. After that, each score is multiplied by its corresponding weight from the chromosome to give the necessary importance to each ratio.

A simple example is demonstrated below using 2016 as the year to enter the market, multiple fictitious companies 3.13 and [0.2, 0.3, 0.6, 0.1, 0.1, 0.3, 0.4, 0.5, 0.1] as the weights of the fundamental chromosome generated from the Evolutionary Algorithm Module.

Table 3.13: Fictitious Example of the Fundamental Dataset.

<table>
<thead>
<tr>
<th>Fields</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gvkey</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Date</td>
<td>2015-12-31</td>
<td>2015-12-31</td>
<td>2015-12-31</td>
<td>2015-12-31</td>
</tr>
<tr>
<td>Fiscal Year</td>
<td>2015</td>
<td>2015</td>
<td>2015</td>
<td>2015</td>
</tr>
<tr>
<td>Tic</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Return on Assets Score</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Return on Assets Delta Score</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Cash Flow Operations Score</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Accrual Score</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Gross Margin Score</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Asset Turnover Score</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Leverage Score</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Current Score</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Eq Offer Score</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Weighted F-Score</td>
<td>0.8</td>
<td>1.2</td>
<td>2.5</td>
<td>0.7</td>
</tr>
</tbody>
</table>

To calculate the weighted F-Score for the company located in column 0, it is taken into account the financial ratio Scores $S$ of the respective company as well as the weights $W$ of the chromosome:

$$F - SCORE_0 = (ROA_{S0} * ROA_{W0}) + (\Delta ROA_{S0} * \Delta ROA_{W0}) + (CFO_{S0} * CFO_{W0}) + (ACCRUAL_{S0} * ACCRUAL_{W0}) + (\Delta MARGIN_{S0} * \Delta MARGIN_{W0}) + (\Delta TURN_{S0} * \Delta TURN_{W0}) + (\Delta LEVER_{S0} * \Delta LEVER_{W0}) + (\Delta LIQUID_{S0} * \Delta LIQUID_{W0}) + (EQ \_ OFFER_{S0} * EQ\_OFFER_{W0})$$

$$= (0 * 0.2) + (1 * 0.3) + (0 * 0.6) + (1 * 0.1) + (0 * 0.1) + (1 * 0.3) + (0 * 0.4) + (0 * 0.5) + (1 * 0.1)$$

$$= 0.8$$

After the F-Score (2.8) has been calculated for all the companies, they get ranked from highest to
lowest score. Subsequently, the companies with the best scores are selected to be part of the portfolio.

3.8 Trading Simulator

The Trading Simulator is a close to the real-life trading environment responsible for training/testing the chromosomes and portfolios created. This simulator runs through every data point in sequence and, for each allows the portfolio companies to take a position. Each company can only have one active position meaning that if the company is already bought, the only positions the system is allowed to execute are to maintain the investment or to sell it. Furthermore, the simulator invests the same amount of money into each company. At the end of each year, all the stocks unsold are sold, and depending on the settings selected, a new portfolio might be created for the upcoming year. When the simulation finishes, the results are sent to the Statistics Module to calculate its return that serves as fitness for the chromosome and portfolio used.

In this subchapter, each function of the trading simulator is described in detail.

3.8.1 Dynamic F-Score

The Dynamic F-Score is the name given to a feature developed to update the portfolio each year. The goal of this innovation relies on getting the best-performing companies from the year \( y - 1 \) ready to trade in the year \( y \), hoping that their fundamentals will enable good trading performance. When this feature is enabled, the Trading Simulator Module “asks” the Portfolio Creator Module to build a new portfolio for the upcoming year.

3.8.2 Investment Strategies

There are multiple approaches to the problem of when to take positions in the market. In this work, two types of strategies were implemented: Buy and Hold and Technical Investment.

1. The Buy and Hold strategy utilizes the portfolio created and buys stocks for each company on the first day of trading and sells them all on the last day of trading.

2. The Technical Investment utilizes the portfolio created and takes positions according to a weighted average between the technical indicators scores and its chromosomes weights.

In order to better understand the technical strategy implemented, an example using scores of a fictitious company during multiple days 3.14 and the technical chromosome \([0.6, 0.7, 0.4, 0.1, 0.9, 0.6]\) is demonstrated below.
Table 3.14: Fictitious Example of the Stock Prices Dataset.

<table>
<thead>
<tr>
<th>Date</th>
<th>RSI_Score</th>
<th>EMA_Score</th>
<th>MAC_Score</th>
<th>ROC_Score</th>
<th>MACD_Score</th>
<th>VIX_Score</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015-09-03</td>
<td>1</td>
<td>-0.5</td>
<td>-0.5</td>
<td>0</td>
<td>-0.5</td>
<td>1</td>
<td>Maintain</td>
</tr>
<tr>
<td>2015-09-08</td>
<td>0.5</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Buy</td>
</tr>
<tr>
<td>2015-09-13</td>
<td>0.5</td>
<td>0.5</td>
<td>0</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>Buy</td>
</tr>
<tr>
<td>2015-09-18</td>
<td>-1</td>
<td>0.5</td>
<td>0.5</td>
<td>0</td>
<td>0.5</td>
<td>1</td>
<td>Maintain</td>
</tr>
<tr>
<td>2015-09-23</td>
<td>-0.5</td>
<td>-0.5</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-0.5</td>
<td>Sell</td>
</tr>
</tbody>
</table>

To calculate the position to execute in 2015-09-08, it is taken into account the technical indicators Scores $S$ of the respective company as well as the weights $W$ of the chromosome:

$$
POSITION_{1} = \frac{(RSI_{S1} \ast RSI_{W1}) + (EMA_{S1} \ast EMA_{W1}) + (MAC_{S1} \ast MAC_{W1}) + (ROC_{S1} \ast ROC_{W1}) + (MACD_{S1} \ast MACD_{W1}) + (VIX_{S1} \ast VIX_{W1})}{RSI_{W1} + EMA_{W1} + MAC_{W1} + ROC_{W1} + MACD_{W1} + VIX_{W1}}
$$

$$
= \frac{(0.5 \ast 0.6) + (1 \ast 0.7) + (0.5 \ast 0.4) + (1 \ast 0.1) + (1 \ast 0.9) + (1 \ast 0.6)}{0.6 + 0.7 + 0.4 + 0.1 + 0.9 + 0.6}
$$

$$
= 0.84
$$

The result is then interpreted in the following way:

- If the result is $> 0.5$, then it represents a Buy position.
- If the result is $< -0.5$, then it represents a Sell position.
- Any other result represents a Maintain position.

During trading, buying and selling prices of stocks, as well as the number of its shares, stay stored in a data frame used to manage the portfolio. After the trading simulator terminates, the data frame is sent to the Statistics Module to calculate the performance of the candidate solution. Below, an example of this type of data frame 3.15 is presented.

Table 3.15: Trading Dataframe Example.

<table>
<thead>
<tr>
<th>operation_date</th>
<th>tic</th>
<th>buy_date</th>
<th>sell_date</th>
<th>buy_price</th>
<th>sell_price</th>
<th>n_shares</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>VLO</td>
<td>2015-01-26</td>
<td>2015-08-24</td>
<td>51.189</td>
<td>58.52</td>
<td>24</td>
</tr>
<tr>
<td>1</td>
<td>HON</td>
<td>2015-02-05</td>
<td>2015-04-21</td>
<td>97.567</td>
<td>96.6049</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>HAS</td>
<td>2015-02-10</td>
<td>2015-08-24</td>
<td>61.610</td>
<td>71.18</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>HPQ</td>
<td>2015-02-15</td>
<td>2015-10-28</td>
<td>17.504</td>
<td>12.7066</td>
<td>71</td>
</tr>
<tr>
<td>4</td>
<td>PH</td>
<td>2015-02-15</td>
<td>2015-05-01</td>
<td>123.629</td>
<td>120.7</td>
<td>10</td>
</tr>
</tbody>
</table>
3.9 Statistics

The Statistics Module has two main goals: The first is to calculate the fitness values for the candidate solutions, using the data frame received from the Trading Simulator Module and the fitness function (3.12). The second is to calculate statistics for the simulations in order to validate and evaluate the system performance. While fitness is calculated to rank the best candidate solutions in the EA module, the statistics are created for the user. With this type of information, he can evaluate the system, compare results, examine consistency, understand the relationship between risk and reward, comprehend the impact of the weights generated in the results of the algorithm and understand the behavior of the model.

After all the calculations are terminated, this module sends the fitness to the Evolutionary Algorithm Module and stores the statistics. In the case where the current iteration is the last, then the module sends the statistics to the User Module to print them to the user.

In this subchapter the metrics used to evaluate and validate the system are presented in detail.

3.9.1 Return Performance Metrics

In order to evaluate and compare the system performance, a list of metrics was selected. The formulas and methodologies pinpoint not only return related results but also examine its risk and its risk/reward. From this list, take part multiple metrics, including the fitness function ROR (3.12), which has already been presented. An in-depth description of each return performance metric is approached below.

(A) Maximum Drawdown (MDD)

This risk measurement represents the most significant peak loss to a minimum point over a specified period.

\[
\text{Maximum Drawdown} = \frac{\text{Trough Value} - \text{Peak Value}}{\text{Peak Value}} \quad (3.15)
\]

This ratio provides an understanding of the most significant percentage loss during the training and testing of the system.

(B) Risk-Return Ratio (RRR)

This risk-adjusted return metric assesses the risk of the investment by making a proportion of the Portfolio Return and its Maximum Drawdown.

\[
\text{RRR} = \frac{\text{Portfolio Return}}{\text{Maximum Drawdown}} \quad (3.16)
\]
This ratio provides an understanding of how much drawdown the investor is eager to take to achieve a specific amount of return. The higher the value, the better.

(C) Annualized Sharpe Ratio

The Sharpe Ratio is a popular metric used to weigh the risk associated with the return of an investment. The excess of return over the risk-free rate of return is standardized over the standard deviation of the portfolio. The higher the Sharpe Ratio is, the better.

\[
\text{Sharpe Ratio} = \frac{\text{Portfolio Return} - (\text{Risk Free Rate} \times \#\text{Trading Years})}{\sigma_{\text{Portfolio}} \times \sqrt{\#\text{Trading Days}}}
\] (3.17)

The risk-free rate expresses the return of an investment that has no type of risk involved. Although this sort of investment is only theoretical, its value is usually used in the Sharpe Ratio. The risk-free rate is taken from the United States Treasury Bills since it is considered the less risky investment known [55, 56]. In this work, the risk-free annual rate considered was 0.02%.

(D) Sortino Ratio

The Sortino Ratio is a risk-return measure and is a variation of the Sharpe Ratio. The main difference between the two remains on the fact that the second takes into account all the volatility, punishing the investment for good risk, while the first isolates the negative volatility from the total volatility, not punishing the investment. In order to achieve it, it uses the assets standard deviation of negative portfolio returns (downside) instead of using the total of the standard deviation of portfolio returns. The formulas between the two ratios are identical in the numerator however, in the denominator, the Sortino Ratio uses the portfolio downside. Just as the Sharpe Ratio, the higher the value, the better the performance of the portfolio.

\[
\text{Sortino Ratio} = \frac{\text{Portfolio Return} - (\text{Risk Free Rate} \times \#\text{Trading Years})}{\sigma_{\text{Downside}}}
\] (3.18)

3.9.2 Performance Metrics

In order to evaluate the evolution of the trading activity, a second pair of metrics was selected. The methodologies display an analysis of the designed system in a statistical manner. Below, those metrics are described in full detail.

(A) Number of Transactions

The number of transactions describes a pair of buy and sell positions on a specific company. Furthermore, a transaction represents an entry on the data frame 3.15.
(B) **Profitable Transactions (PT)**

The Profitable Transactions represents the percentage of investments that had profit.

\[
\text{ProfitableTransactions} = \frac{\#\text{Transactions}_{\text{SellPrice} > \text{Transactions}_{\text{BuyPrice}}}}{\#\text{Transactions}} \quad (3.19)
\]

(C) **Unprofitable Transactions (UT)**

The Unprofitable Transactions represents the percentage of investments that had a loss.

\[
\text{UnprofitableTransactions} = \frac{\#\text{Transactions}_{\text{BuyPrice} > \text{Transactions}_{\text{SellPrice}}}}{\#\text{Transactions}} \quad (3.20)
\]

(D) **Maximum Profit (MaxP)**

This metric represents the maximum profit obtained in a trade expressed as a percentage.

(E) **Minimum Profit (MinP)**

This metric represents the minimum profit obtained in a trade expressed as a percentage.

(F) **Average Profit (AvgP)**

This metric represents the average profit of the entire investment expressed as a percentage.

\[
\text{AverageProfit} = \frac{\sum \text{ProfitPerTransaction}}{\#\text{Transactions}} \quad (3.21)
\]

(G) **Average Return (AvgR)**

This metric represents the average return of the entire investment expressed as a percentage. The difference between this metric and the Average Profit is that this one takes into account the profits and losses while the Average Profit only takes into account the profits.

\[
\text{AverageReturn} = \frac{\sum \text{ReturnsPerTransaction}}{\#\text{Transactions}} \quad (3.22)
\]
4 Results

Contents

4.1 Results Overview ................................................................. 63
4.2 Sliding Window ................................................................. 63
4.3 Case Studies ................................................................. 64
The goal of this section is to present and examine the case studies developed in this work. In the first part, the case studies are presented and their structure explained. After that, an in-depth look is taken at each case study, and its subtests discussed. At the end of the section, the results are compared and a summary is made.

4.1 Results Overview

This thesis comprehends two central case studies: The Technical Investment and the Fundamental Investment. Each case study subdivides itself into four specific cases:

1. With dynamic F-score.
2. Without dynamic F-score.
3. With dynamic F-score and self-improvement.
4. Without dynamic F-score but with self-improvement.

In order to test the suggested cases, the architecture and parameters presented in chapter 3 were used. The subtests were submitted to a period of training and testing using a sliding window approach to expose the system to different environments. In order to obtain more robust results, the trains/tests were repeated ten times each and the metrics introduced in the Statistics subchapter 3.9 were utilized as system validation metrics. To understand how the system performance is related to a “real-life” trading method, the SP500 index prices were used as a “benchmark.”

4.2 Sliding Window

To adequately evaluate the performance of the designed solution, a sliding window scheme was developed. For a fair evaluation, the system must be tested on data that has not been used in its training, otherwise, the algorithm would automatically “know” the “path” to obtain better results since it had already trained multiple times in that environment.

The developed scheme comprises blocks of one year that together comprehend data from 2012-01-01 to 2018-12-31. The multiple arrangements of these blocks compose six types of sliding windows. Each sliding window name represents the number of training blocks first and the number of testing after. A simple example is the window 32 that represents three years of training and two years of testing. In order to compare the results from different sliding windows, each composition includes the same four years of testing.
According to the selected benchmark, the years utilized for testing own considerable turbulence. Between the start and conclusion of the testing years, the market possesses multiple falling, climbing, and even some sideways market moments that put any trading algorithm to the test.

The goal of this implementation was mostly to get various types of training sessions but also to understand the best approach to train the system since financial markets suffer from seasonality. The figure below 4.1 represents all the types of windows created.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td></td>
<td>Train</td>
<td>Train</td>
<td>Test</td>
<td>Test</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td>Train</td>
<td>Train</td>
<td>Test</td>
<td>Test</td>
<td>Train</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td>Train</td>
<td>Train</td>
<td>Test</td>
<td>Test</td>
<td></td>
<td>Train</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td>Train</td>
<td>Train</td>
<td>Test</td>
<td>Test</td>
<td></td>
<td>Train</td>
<td>Test</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>Train</td>
<td>Test</td>
<td>Test</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>Train</td>
<td>Test</td>
<td>Test</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.1: Sliding Window Scheme.

### 4.3 Case Studies

In this segment, the results from the case studies are presented and analyzed. For each case, a table containing its results and the benchmark utilized for comparison is displayed. The table is divided into three main sections: “Test settings”, “Return Performance” and “Performance.” The Test Settings contains the type of tests (subtests) and the windows of each test. In this part, the “epoch” is also presented, which varies from “Average” (average of the ten runs) to “Best” (best run). The “Return Performance” and “Performance” sections contain the metrics presented in the Statistics Module 3.9 in the subsections 3.9.1 and 3.9.2 respectively.

It is also necessary to notice that the colors of the sliding windows are the same as figure 4.1. Additionally, the results presented in the tables show a different color according to their value in the column. A lousy value is represented with a red color, a median value gets a yellow/orange gradient color, and a good value is represented with a green color.

To better understand the system behavior, at the end of each case study, a small table formed by
the weights who achieved the best results is introduced. The table does not just incorporate the weights that achieved the best performance in the subtest but also their respective average epoch. The “Epoch” field has the color of the sliding window (as in figure 4.1) which performed best in that specific subtest. Each weight cell is composed of the weight number but also a bar (representing the weight value) for a clear understanding and examination of the results.

4.3.1 Technical Investment

In this case study, the EA evolved multiple candidate solutions in order to achieve the best possible return. The candidates are composed of weights for the portfolio creation (fundamental representation module) and the technical investment strategy (technical representation module). In the last two subcases, the candidate solutions also included the variation probabilities in their representations (self-adaptive representation module) to create a self-adaptive EA.

The table below 4.1 shows the results obtained for the multiple case studies tested. This section also includes an in-depth look at each subtest of this case study as an examination of the simulations that achieved more relevant results.

(A) With Dynamic F-Score

This case is designed to understand the impact that technical indicators and a yearly change in the portfolio have in the trading returns. In the following figures 4.2, it is possible to see the average and best trading simulations for this test.

Upon inspection of 4.2(a), it is possible to perceive that the average simulations all have a very similar path. Starting with an initial drop that quickly stabilizes, followed by a hard climb that ends with a small dip (in comparison to the benchmark). Table 4.1 shows that the average runs achieve remarkable results in the MDD metric. While the benchmark gets a whopping MDD of 17%, the simulations MDD ranges from 8.42% to 13.34%. In terms of return, the average simulations are not capable of achieving high values. While the SP500 achieves 16% of return, the maximum value achieved by an average simulation is 8.60%. The risk-return metrics (RRR, Sharpe Ratio and Sortino Ratio) show that the average simulations do not have splendid results, however, test 11 achieves a RRR superior to the benchmark due to its almost 10% lowest drawdown and its ROR of 8.6%.

When inspecting 4.2(b), it is clear that the best simulations paths are similar to their averages with the advantage of being capable of surpassing the benchmark returns in half of the tests. The table 4.1 reveals that the best runs once again surpass the benchmark MDD. In terms of return, the simulations are able to achieve higher results than the benchmark in half the tests. The same happens with the risk-return metrics. Finally, it is worth noting that the tests 31, 21, and 11 entirely surpass the benchmark.
<table>
<thead>
<tr>
<th>Test Type</th>
<th>Without Score</th>
<th>With Score</th>
<th>Score Only</th>
<th>Dynamic - Score Only</th>
<th>Dynamic - Score Only Only</th>
<th>Benchmark</th>
<th>S&amp;P500</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
<tr>
<td>Static</td>
<td>0.08</td>
<td>0.12</td>
<td>0.20</td>
<td>0.25</td>
<td>0.29</td>
<td>0.05</td>
<td>0.21</td>
</tr>
</tbody>
</table>

Table 4.1: Technical Investment Results.
achieving better values in all return performance metrics. In particular, the 21 test yielded 7% lower MDD and three times higher Sharpe Ratio than the benchmark, making it a more profitable and safer investment in terms of risk.

(B) Without Dynamic F-Score

This case is designed to understand the impact that maintaining a fixed portfolio for a long time while using technical indicators has in the trading returns. Figures 4.3 show the average and best simulations of this subtest.
the 32 test which holds the most significant decline in the first dip, however, proceeds on a “bull run” that enables it to achieve the best return of this average simulations subtest (including the benchmark). The table 4.1 shows that most average simulations have a lower MDD than the benchmark. In terms of returns and risk-return metrics, the simulations do not have satisfying results. The average returns showed an almost 7% difference from the benchmark, and the average Sharpe Ratio presented values 13 times lower than the benchmark. Unfortunately, not all average simulations reported good results, however, the 32 simulation overcomes the benchmark in all the “Return Performance Metrics” except for the MDD. By achieving a better Sharpe and Sortino Ratio, the 32 test turns out to be a better investment than the benchmark as a result of its low risk and high profits.

When inspecting 4.3(b), it is observable that most of the best simulations have low falls in the first dip and are able to “climb” the market at a quick pace (simulation 12 is even able to “climb” it a faster pace than the benchmark). When the market starts to collapse again, the simulations are able to stop the losses early and end up with better returns than the benchmark. Table 4.1 shows that the benchmark has the biggest MDD of all simulations. However, 32 and 22 also have high drawdowns with values above 17%. In terms of return, all the simulations can surpass the benchmark with simulation 32 having returns over 27%. In terms of risk-return metrics, all the simulations surpass the benchmark in every single metric, suggesting that all these investments are safer and more profitable than the benchmark. A compelling case presented in 4.1 is the Sharpe and Sortino Ratio disparity between some tests. While the best Sharpe Ratio is attained by test 31, the best Sortino Ratio is achieved by test 32 and 12. This difference is mainly connected to the Sortino calculation separating the negative volatility to avoid punishing the investment for the “good volatility.”

(C) With Dynamic F-Score and Self-Improvement

In order to comprehend the influence of adding variation probabilities in the chromosomes structure, a derivation of the first subtest was created. The goal of this subtest is to understand the impact that including these capabilities, combined with technical investment and a dynamic portfolio produce on trading returns. The following figures 4.4, represent the average and best trading simulations for this subtest.

When analyzing 4.4(a), it is observable that the average simulations have very similar behavior. Starting with a small market drop, followed by a “bull run” that begins late (in comparison with the benchmark), and ending with a short decline. Upon analysis of 4.1, it is noticeable that all the simulations produce low values of MDD with an average 7% lower difference than the benchmark. In terms of return, the simulations also have inferior results than SP500 with a maximum return of 10.81%. The risk-return metrics reveal that the benchmark surpasses the simulations, however, tests 32 and 22 yield higher results of RRR.
The best simulations 4.4(b) start with a short fall in the market, accompanied by much volatility. After the rough start, they are able to “climb” the market at a good pace, and test 32 is even capable of getting close to one of the benchmark's highest peaks. A small drop characterizes the end of the test enabling the simulations to finish with better returns than the index. Table 4.1 shows that the best simulations exhibit lower values of MDD than the benchmark. The return and risk-return metrics reveal that the simulations surpassed the SP500 unveiling to be better investments than the index. Test 11 was the one who performed best yielding an almost 10% positive return difference and more than three times higher Sharpe Ratio than the benchmark.

(D) Without Dynamic F-Score but With Self-Improvement

To comprehend the importance of adding variation probabilities into the chromosome structure, a derivation of the second subtest was conceived. The purpose of this experiment is to understand the effect that these capacities, coupled with a technical investment strategy and a static portfolio, produce on trading returns. The following figures 4.5, picture the average and best trading simulations for this subtest.

The analysis of the figure 4.5(a) unveils two significant groups in the average simulations. The first is characterized by a turbulent beginning followed by a downfall. In this group, some simulations reveal the ability to start an early market climbing (test 12). However, most start raising their market price much later. After the “bull” momentum ceases, these simulations reveal a low price drop when compared to the index. The second group, which in reality is only simulation 22, has a very similar start when compared to the first. Nevertheless, after the fall, it begins raising its price much earlier, crossing the index and achieving better performance than it. At the end of the test, simulation 22 reveals a much
lower decline than the index finishing the test with a ROR two times higher than the SP500. During table 4.1 examination, it is noticeable that every simulation achieved an inferior MDD than the index. In terms of return and risk-return, the tests exhibited some under the benchmark results. However, the average for these simulations surpasses the benchmark in those metrics. As mentioned before, test 22 was able to produce remarkable statistics. Besides a stellar ROR of 35.67%, this simulation achieves (compared to the benchmark) an almost four times higher RRR, a 4.2 times higher Sortino Ratio and lastly a magnificent 4.9 times higher Sharpe Ratio. For these reasons, test 22 is considered a better investment than SP500.

When analyzing the best simulations in figure 4.5(b), it is also possible to differentiate two significant groups. The first one falls in the initial dips and starts “climbing” the market after the index. The second group (test 22 and 12) also drops in the first dips, however, starts raising its market price at around the same time as the benchmark. Both groups reveal small decays at the end of the test, performing a better ROR than the benchmark. Through analysis of table 4.1, it is perceivable that all best simulations perform a lower MDD, opposed to the index with an 8.81% average difference from it. Return-wise, the best simulations produce, on average, an approximately two times higher value than the benchmark and on the risk-return metrics surpass the SP500 in all areas. In this test, simulations 12 and 22 achieved remarkable results. The first is able to achieve 2.5 times higher (40.72%) return than the benchmark, while 22 achieves 3.5 times higher (57.34%) return than the index. Test 22 scores not only the highest ROR but also one of the lowest MDD, which is unusual since most times, a high ROR is correlated to a high MDD. In the risk-return metrics, these simulations also achieve outstanding results with the example of test 22 that achieves a 2.45 Sharpe Ratio. For the presented reasons, both tests display a much better investment than the SP500.
A small table 4.2 containing the weights of the best technical subtests simulations (and their respective averages) is now presented and analyzed to comprehend the importance given to each variable in the best epochs.

**Table 4.2: Weights Used in the Best Technical Results.**

<table>
<thead>
<tr>
<th>Subtest</th>
<th>Epoch</th>
<th>ROA</th>
<th>ΔROA</th>
<th>CFO</th>
<th>Accrual</th>
<th>Margin</th>
<th>Turn</th>
<th>Leverage</th>
<th>CR</th>
<th>EQ Offer</th>
<th>RSI</th>
<th>EMA</th>
<th>MAC</th>
<th>ROC</th>
<th>MACD</th>
<th>VIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Average</td>
<td>0.81</td>
<td>0.43</td>
<td>0.13</td>
<td>0.57</td>
<td>0.57</td>
<td>0.46</td>
<td>0.47</td>
<td>0.6</td>
<td>0.5</td>
<td>0.36</td>
<td>0.32</td>
<td>0.34</td>
<td>0.29</td>
<td>0.52</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.41</td>
<td>0.26</td>
<td>0.17</td>
<td>0.51</td>
<td>0.47</td>
<td>0.73</td>
<td>0.35</td>
<td>0.58</td>
<td>0.48</td>
<td>0.27</td>
<td>0.43</td>
<td>0.35</td>
<td>0.32</td>
<td>0.50</td>
<td>0.60</td>
</tr>
<tr>
<td>B</td>
<td>Average</td>
<td>0.59</td>
<td>0.37</td>
<td>0.54</td>
<td>0.5</td>
<td>0.48</td>
<td>0.61</td>
<td>0.75</td>
<td>0.37</td>
<td>0.6</td>
<td>0.3</td>
<td>0.15</td>
<td>0.1</td>
<td>0.28</td>
<td>0.17</td>
<td>0.81</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.51</td>
<td>0.58</td>
<td>0.15</td>
<td>0.46</td>
<td>0.7</td>
<td>0.29</td>
<td>0.91</td>
<td>0.41</td>
<td>0.58</td>
<td>0.45</td>
<td>0.14</td>
<td>0.13</td>
<td>0.23</td>
<td>0.06</td>
<td>0.89</td>
</tr>
<tr>
<td>C</td>
<td>Average</td>
<td>0.4</td>
<td>0.44</td>
<td>0.45</td>
<td>0.51</td>
<td>0.44</td>
<td>0.53</td>
<td>0.47</td>
<td>0.54</td>
<td>0.44</td>
<td>0.51</td>
<td>0.55</td>
<td>0.42</td>
<td>0.51</td>
<td>0.35</td>
<td>0.42</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.26</td>
<td>0.65</td>
<td>0.40</td>
<td>0.36</td>
<td>0.73</td>
<td>0.39</td>
<td>0.33</td>
<td>0.55</td>
<td>0.39</td>
<td>0.68</td>
<td>0.47</td>
<td>0.46</td>
<td>0.66</td>
<td>0.61</td>
<td>0.54</td>
</tr>
<tr>
<td>D</td>
<td>Average</td>
<td>0.57</td>
<td>0.28</td>
<td>0.35</td>
<td>0.53</td>
<td>0.57</td>
<td>0.41</td>
<td>0.69</td>
<td>0.55</td>
<td>0.18</td>
<td>0.51</td>
<td>0.56</td>
<td>0.37</td>
<td>0.45</td>
<td>0.38</td>
<td>0.5</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.59</td>
<td>0.38</td>
<td>0.22</td>
<td>0.43</td>
<td>0.7</td>
<td>0.51</td>
<td>0.14</td>
<td>0.49</td>
<td>0.36</td>
<td>0.31</td>
<td>0.46</td>
<td>0.33</td>
<td>0.29</td>
<td>0.46</td>
<td>0.51</td>
</tr>
</tbody>
</table>

An overall analysis of table 4.2 best epochs, reveals that the CFO and Accrual fundamental variables all yield lower values than their average. The same examination additionally unveils that the equivalent happens in the VIX technical variable, yet this time with higher than average values. Further investigation indicates that, on the one hand, A and C (subtests with dynamic F-Score) have lower than average best weights in the ROA, LEV and EQ OFFER. Moreover, the same subtests display higher than average best weight values in that MAC and MACD variables. On the other hand, B and D (subtests with dynamic F-Score) show that their best results are accomplished with lower than average weights in variables EMA and ROC. Other than that, Delta ROA and MARGIN variables perform their best simulations with weights above average, with the last one presenting best values around the same number. To complete the analysis, this table 4.2 likewise reveals that it should be given more significance to the MARGIN variable since it presents three (of the four) best values with weights around 0.70. Furthermore, it should not be given so much importance to the CFO variable since, in the table 4.2, three of the four best cases, reveal values under 0.30.

### 4.3.2 Fundamental Investment

In this case study, the EA also evolved multiple candidate solutions in order to achieve the best return possible. The candidates are formed of weights for the portfolio creation (fundamental representation module) that is later used in the Trading Simulator Module with a Buy-and-Hold investment strategy. In the last two subcases, the candidate solutions also included the variation probabilities in their representations (self-adaptive representation module) to create a self-adaptive EA.

The table below 4.3 shows the results obtained for the multiple case studies tested. It is relevant to notice that the “Transactions” field was suppressed since all simulations produced the same number of
transactions (80). This section also includes an in-depth look at each subtest of this case study as an examination of the simulations that achieved more relevant results.

(A) With Dynamic F-Score

This case is designed to understand the impact a yearly change in the portfolio has on the trading returns while using a buy and hold trading strategy. In the following figures 4.6, it is possible to see the average and best trading simulations for this test.

Upon inspection of 4.6(a), it is noticeable that the average simulations have significant similarities with the benchmark. They all begin by falling into two dips and later start “climbing” the market at a similar pace. At the end of the test, the simulations decline mostly for entering in a sideways market that ends up by declining two times. During this market action, test 21 stands out for its distance compared to the other simulations. After the two significant falls at the beginning, this test has the ability to rise in the market at a faster pace achieving its highest ROR of 54%. In 4.3, the results indicate that the average simulations achieved high values of MDD. In terms of returns and risk-returns metrics, the benchmark is also able to surpass the average simulations. Although the average results were not the most solid, test 21 succeeds in attaining better results in all metrics except the MDD, still making it a better investment than the benchmark.

In figure 4.6(b), it is shown that the best simulations also drop in the first dips, but this time most are able to overcome the benchmark and achieve high values. The 21 and 11 tests are a great example of that, achieving values of 89% and 59% ROR, respectively, at their highest points. After achieving their highest returns, the simulations end up following the trend and decline. By analyzing table 4.3, it is noticeable that even the best simulations were not able to beat the benchmark on the MDD. Their average MDD was 24% in contrast with the 17% MDD of the index. The return metrics show on average higher values (0.17% difference) to the index, however, their variance is high. On the one hand, there are meager returns such as the 31 and 12 tests, on the other hand, there are simulations with an extremely high rate of returns such as the 21 and 11 tests. In the risk-return metrics, the Sortino Ratio displayed on average values equal to the benchmark while the Sharpe Ratio and RRR were both under the benchmark by a difference of 0.05 and 0.28, respectively. Although the presented simulations did not yield amazing results, one unusual case was noticed. The 11 case achieved the poorest MDD with the highest value of 29.41%, however, it was able to achieve the best ROR of 40% and a Sharpe Ratio three times bigger than the benchmark.

(B) Without Dynamic F-Score

The main goal of this subtest is to understand the impact that maintaining a fixed portfolio for an extended period while using a buy and hold investment has in trading returns. Figures 4.7 show the
<table>
<thead>
<tr>
<th>Test Settings</th>
<th>Return Performance</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Test Type</strong></td>
<td>Sliding Window</td>
<td>Epoch</td>
</tr>
<tr>
<td><strong>With Dynamic F-Score</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>22.42%</td>
<td>0.57</td>
</tr>
<tr>
<td>Best</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Without Dynamic F-Score</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>24.99%</td>
<td>0.43</td>
</tr>
<tr>
<td>Best</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>With Dynamic F-Score and Self-Improvement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>21.24%</td>
<td>1.19</td>
</tr>
<tr>
<td>Best</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Without Dynamic F-Score but With Self-Improvement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>22.48%</td>
<td>1.42</td>
</tr>
<tr>
<td>Best</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Benchmark</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP500</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>With Dynamic F-Score and Self-Improvement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>23.24%</td>
<td>0.13</td>
</tr>
<tr>
<td>Best</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Benchmark</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Benchmark SP500</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>With Dynamic F-Score but With Self-Improvement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>23.39%</td>
<td>0.23</td>
</tr>
<tr>
<td>Best</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Benchmark</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP500</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>With Dynamic F-Score and Self-Improvement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>23.75%</td>
<td>0.35</td>
</tr>
<tr>
<td>Best</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Benchmark</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Benchmark SP500</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
average and best simulations of this subtest.

The analysis of the figure 4.7(a) allows detecting a clustered market action divided into two major groups. While the first composes itself of simulations able to surpass the benchmark, the second is constituted of solutions that were not. In this subtest, the simulations start by falling in the first dips, and afterward, some continue to raise their market price at a quick pace while others do it at a slower pace. In this subtest, simulations 22 and 12 stand out just by looking at 4.7(a). Equally important as their ability to “climb” the market at a quicker pace than the index, they are also able to achieve stellar returns at their highest point with 22 scoring 62% and 12 closing 51%. When analyzing table 4.3, it is recognizable that the average simulations obtain on average higher values of MDD than the benchmark, indicating a more significant number of declining moments in the simulations. In the return and risk-return metrics,
the average values also exhibit under the benchmark results. Although the average results yield inferior values, test 22 and 12 present a remarkable performance. On the one hand, both are able to achieve exceptional rate of returns (39% and 30%) and over the benchmark risk-return metrics. On the other hand, they achieve a spectacularly profitable transaction percentage with values around 70% contrasting with the “usual” values typically obtained of around 50% or lower.

When looking at the best simulations figure 4.7(b), it is possible once more to differentiate two types of market action: the first is able to surpass the benchmark with distinction, and the second is not even capable of crossing it. Besides this differentiation, the price behavior is very similar to its average tests with a turbulent start accompanied by two drops, followed by a market “climb” and finishing up with a declining market. In terms of statistics, the results 4.3 show that, on average, the simulations are able to achieve higher values of MDD and rate of returns. In the risk-return metrics, the tests surpass the benchmark in all fields except the Sortino Ratio. In these simulations, tests 22 and 12 presented the best values. One the one hand were able to achieve low maximum drawdowns, returns around 44% and 36% respectively, and also stellar results in the risk-return metrics. On the other hand, in both tests, over 70% of the positions executed turned into profitable transactions. For these reasons, both simulations can be considered a much better investment than the benchmark.

(C) With Dynamic F-Score and Self-Improvement

In order to comprehend the influence of combining variation probabilities in the chromosomes’ structure, a derivation of the first subtest was conceived. The goal of this test is to understand the impact that these capabilities, combined with a buy and hold strategy and a dynamic portfolio, have on trading returns. The following figures 4.8, represent the average and best trading simulations for this subtest.

![Figure 4.8: Fundamental Investment With Dynamic F-Score and Self-Improvement Simulations.](image-url)
During the analysis of figure 4.8(a), it is possible to perceive that the price action of the average simulations is extremely similar to the other fundamental tests presented. It starts by falling in two big dips, grows until it reaches its maximum and near the end of the test has two abrupt falls. In figure 4.8(a) test 21 distances positively from the benchmark, however, in the final part of the test declines in such an abrupt way that finishes with a lower ROR than the index. Table 4.3 results show that with the constraints utilized, the system is unable to overperform the benchmark. The average MDD metric yields 24.70% contrasting with the much lower 17.41% from the benchmark. In the ROR, the simulations achieve, on average, 3.41%, however, test 21 reaches a superior and closer to the benchmark 13.20% ROR. Finally, in the risk-return metrics, the Sharpe and Sortino Ratio only achieve a positive value in test 21, and the average RRR presents 6.5 times lower than the benchmark result.

Upon inspection of figure 4.8(b), it is perceivable that most experiments were able to surpass the benchmark during the test years. The best simulations start with two big dips, accompanied by high levels of volatility. After the turbulent times, these begin “escalating” the market at different paces, and after achieving their maximum ROR, get dragged by two abrupt declines. An intriguing aspect present is the ability of simulation 32 to escape from the first “dip”, although being unable to repeat the feat at the end of the test. Figure 4.8(b) also reveals that both 32 and 21 simulations “climb” the market at a quicker pace than the benchmark being capable of achieving the maximum ROR for all the simulations in this subtest. The results of table 4.3 reveal that none of the simulations was able to achieve lower MDD than the index however, test 11 gets the closest with a MDD of 18%. In the return metrics, the experiments on average surpass the benchmark and test 11 (which went unnoticed throughout the test) is able to achieve the best ROR at the end. In the risk-return metrics, the simulations on average are not capable of surpassing the benchmark. Nevertheless, tests such as 11 and 32 achieve better values than the benchmark. Although simulations 11, 32 and 21 were not able to surpass the index’s MDD, these can still be considered better investments than the benchmark for their good performance.

(D) Without Dynamic F-Score but With Self-Improvement

To comprehend the significance of appending variation probabilities in the structure of the chromosome, a derivation of the second subtest was conceived. The purpose of this experiment is to understand the influence that these capacities, coupled with a buy and hold investment strategy and a static portfolio, produce on trading returns. The following figures 4.9, picture the average and best trading simulations for this subtest.

Upon examination of the figure 4.9(a), it is reasonable to classify the simulations in two distinct groups. The first is formed by simulations that declined in the initial market dips and endured below the benchmark for the remaining test. The second group is composed of simulations that also declined on the same dips as the first, however, it demonstrated the ability to recover and remain above the
benchmark throughout the test achieving better returns than it. From the analysis of table 4.3, it is possible to infer that half of the simulations achieved a lower MDD than the benchmark. Although the remaining half achieves worse results, the average MDD exhibits a 3% positive difference from the SP500. In terms of return and risk-return metrics, the same occurs. Half of the simulations surpass the benchmark values, while the other half does not. On average, the ROR accomplishes a positive difference of 3.6% when compared to the benchmark. In the risk-return metrics, the average Sortino and Sharpe Ratio score a 0.09% and 0.15% positive difference from the index while the average RRR gets a 0.25% positive difference from the index. In this subtest, the simulations 32 and 22 stand out for their high returns of 34.37% and 39.49%, respectively. Besides their above-average returns, these tests achieved remarkable results in the other metrics. On the one hand obtained under the benchmark MDD and high results in the risk-return metrics (RRR of 2.05 and 2.36, Sharpe Ratio of 0.98 and 1.19 and finally Sortino Ratio of 0.56 and 0.68). On the other hand, in the “Performance” metrics, both tests achieved outstanding values in the profitable transactions metric, with results around 68%.

In figure 4.9(b), it is perceivable that all the best simulations decline in the first dips of the test. However, most present a better market price action than the benchmark. After their first decline, all start raising their prices, most finishing with higher returns than SP500. In table 4.3, it is noticeable that simulations in which their name ends in 1 achieve worse MDD results than the benchmark. However, the remaining half achieves remarkable results. In terms of return and risk-return, all simulations surpass the benchmark except for test 21. On average, the best simulations score a ROR and RRR two times higher than the index. Additionally, both Sharpe and Sortino Ratios achieve results three times higher or more than the benchmark. From this subtest, simulations 22 and 32 deserve special attention. Not only did both tests deliver exceptional returns (52.65% and 40.16% respectively) and risk-return metrics (1.78 Sharpe Ratio and 1.20 respectively), but they also produced high-grade PT values (77% and 75% respectively).
respectively). Test 22 is even capable of achieving the highest ROR of the subtest and the lowest MDD of all. For the aforementioned reasons, both tests can be considered better investments than SP500.

A small table 4.4 containing the weights of the best fundamental subtests simulations (and their respective averages) is now presented and analyzed to comprehend the importance given to each variable in the best epochs.

<table>
<thead>
<tr>
<th>Subtest</th>
<th>Epoch</th>
<th>ROA</th>
<th>ΔROA</th>
<th>CFO</th>
<th>Accrual</th>
<th>Margin</th>
<th>Turn</th>
<th>Lev</th>
<th>CR</th>
<th>EQ Offer</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Average</td>
<td>0.44</td>
<td>0.54</td>
<td>0.51</td>
<td>0.42</td>
<td>0.58</td>
<td>0.52</td>
<td>0.41</td>
<td>0.51</td>
<td>0.45</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.33</td>
<td>0.44</td>
<td>0.64</td>
<td>0.41</td>
<td>0.34</td>
<td>0.45</td>
<td>0.45</td>
<td>0.70</td>
<td>0.55</td>
</tr>
<tr>
<td>B</td>
<td>Average</td>
<td>0.58</td>
<td>0.35</td>
<td>0.49</td>
<td>0.51</td>
<td>0.51</td>
<td>0.45</td>
<td>0.63</td>
<td>0.37</td>
<td>0.30</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.40</td>
<td>0.27</td>
<td>0.58</td>
<td>0.32</td>
<td>0.39</td>
<td>0.22</td>
<td>0.21</td>
<td>0.74</td>
<td>0.85</td>
</tr>
<tr>
<td>C</td>
<td>Average</td>
<td>0.42</td>
<td>0.53</td>
<td>0.49</td>
<td>0.44</td>
<td>0.55</td>
<td>0.62</td>
<td>0.39</td>
<td>0.50</td>
<td>0.39</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.28</td>
<td>0.36</td>
<td>0.41</td>
<td>0.41</td>
<td>0.37</td>
<td>0.51</td>
<td>0.61</td>
<td>0.58</td>
<td>0.67</td>
</tr>
<tr>
<td>D</td>
<td>Average</td>
<td>0.45</td>
<td>0.38</td>
<td>0.50</td>
<td>0.41</td>
<td>0.71</td>
<td>0.61</td>
<td>0.64</td>
<td>0.48</td>
<td>0.31</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.41</td>
<td>0.66</td>
<td>0.42</td>
<td>0.38</td>
<td>0.60</td>
<td>0.44</td>
<td>0.35</td>
<td>0.89</td>
<td>0.47</td>
</tr>
</tbody>
</table>

Through the examination of the table 4.4 best epochs, it is possible to observe that variables ROA, Accrual, MARGIN and TURN all yield lower values than their average. From the same observation, it is possible to conclude that CR and EQ OFFER use higher weights in the best simulations than their averages. Further investigation indicates that, on the one hand, A and C (subtests with dynamic F-Score) have lower than average best weights in Delta ROA but achieve lower values in the LEV variable. On the other hand, B and D (subtests with dynamic F-Score) show that their best results are accomplished with lower than average weights in the variable LEV. To complete the analysis, this table 4.4 reveals that it should be given more significance to the CR variable since it presents 3 (of the four) best values with weights around 0.7/0.8.

4.3.3 Results Comparison

In this subsection, a comparison between the presented results is carried out as a performance analysis of the multiple proposed features. For that purpose, a new figure and two tables are introduced. The figure 4.10 divides itself into the technical and fundamental case studies approached, outlining their average training performance in the various developed subtests using the EA. Moreover, the technical subfigure 4.10(a) is composed of two y-axes, each assigned to the trace with the matching color. The first 4.5 and second table 4.6 represent the correlation matrices of each case study. The matrices incorporate the average epoch weights used in chromosomes testing and also the average results presented in table 4.1 and table 4.3 as data.
Both figures 4.5 and 4.6 utilize colors to help the viewing and interpretation experience. The orange color presents a moderate positive/negative correlation ($>0.6$ or $<-0.6$), yellow corresponds to a fairly strong positive/negative correlation ($>0.8$ or $<-0.8$), and green presents a perfect positive/negative correlation ($=1$ or $=-1$).

![Figure 4.10: Average Training of the Evolutionary Algorithm.](image)

(A) Technical Subtests

The first case study is characterized by achieving better results on average in the sliding window, composed of two years of training and two years of testing. Furthermore, this case study presents better results on average using a non-dynamic portfolio management strategy. Besides using a new approach for portfolio management, this study also tests the ability for the EA to self adapt during the evolutionary process. In that matter, the self-improved derivations subtests achieved better results on average than the “normal evolution” ones in all return performance metrics.

The examination of figure 4.10(a) leads to the conclusion that all subtests were able to evolve their populations during the training sessions. On the one hand, the simple dynamic EA achieved an average 50% ROR while its derivation (EA with self-improvement) achieved on average ROR of 45% not being able to surpass the original test (unlike the test sessions). On the other hand, it is perceivable that the simple non-dynamic EA achieved a staggering 156% average ROR surpassed by its derivation test using a self-improved EA that achieved an average 170% ROR as opposed to their testing sessions.

The analysis of the technical correlation matrix 4.5 allows the discovery of interesting information. There are four main conclusions from the matrix. The first is that return/risk-return/profit variables impact the number of transactions made during market trading. This conclusion comes from a negative correlation meaning that higher profits/returns lead to a lower number of transactions. The second conclusion comes from the TI VIX. According to the correlation value, the higher the weight given to VIX, the lower
Table 4.5: Technical Case Study Correlation Matrix.

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4.6: Fundamental Case Study Correlation Matrix.

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
the number of transactions made. The fact that this TI presents a positive correlation with MaxP, MinP, AvgP, and AvgR is evidence that when more weight is given to VIX, more profit and returns come from trading the markets using the ability of this TI of warning the system in a declining market, to avoid capital losses. The third conclusion comes from the relation between EMA with MinP and yields a negative correlation between them, meaning that a high weight given to EMA results in a lower MinP. Finally, the forth conclusion is related to the LEV variable, which represents a company’s debt issues. This correlation combines multiple variables. From the return/risk-return/profit to VIX and even a negative correlation with the number of transactions and the UT. By giving high weight to this indicator, the correlation matrix tells us that our profits and returns will get higher, and also our number of transactions and UT will get lower. This conclusion is important information to take into consideration in future portfolio compositions.

(B) Fundamental Investment

The second case study is characterized by achieving its best average results using two training years and one/two testing years. Additionally, this case study presents better results on average using a static portfolio management strategy. In terms of self-adaptive EA, the self-dynamic subtest presents, on average, nearly equivalent results as the original test. However, the same does not happen in the static portfolio subtest, where its self-adaptive derivation shows better results in every single performance return average metric.

Through the analysis of figure 4.10(b), it is noticeable that during training, the algorithm was able to evolve its population. On the one hand, the simple dynamic EA achieved a 55% average ROR while its self-adaptive derivation produced a nearby average value of 54% ROR as it happened in the test sessions. On the other hand, the non-dynamic subtest achieves, on average, a ROR of 53% approximately while its self-adaptive derivation produces a lower value around 50% ROR which is the opposite of what happens in the testing sessions.

The analysis of the fundamental correlation matrix 4.6 leads to several conclusions, however, most of them were already presented in the Technical Investment Section (A). The MDD presents a negative correlation between multiple return/risk-return/profit metrics yielding that the higher the value of MDD, the lower the trading returns. One interesting point to notice is that in the technical correlation matrix 4.5 this correlation did not show up mostly because there are technical indicators that avoid market declines when using a technical investment like the one used in the first case study.

(C) Results Summary

The technical case study achieved its best average simulation without using a dynamic portfolio strategy, however, it used a self-improved EA. This simulation achieved the best ROR of all average simulations with a 35.67% ROR, 11.90% MDD, Sharpe Ratio of 1.33, RRR of 3.62, Sortino Ratio of 0.63 and a sliding window of 22. From the same subtest and the same sliding window comes the best simulation.
with a 57.34% ROR, MDD of 7.67%, Sharpe Ratio of 2.45, RRR of 7.47, and a Sortino Ratio of 1.17.

Through the analysis of all the figures presented in 4.3.1 as the results table 4.1, it is possible to conclude that the technical system was able to avoid multiple losses and present fairly consistent profits. Thanks to the selected technical indicators, the system was able to abandon the market (creating the horizontal lines seen in the multiple figures) in multiple situations that later proved to be declining moments. However, during “bull markets”, this system showed some lack of “confidence” to pursue higher returns since it would take some time to start “climbing” the market after declining moments.

The fundamental case study achieved its best average simulation in the same subtest and sliding window as the technical case study. This simulation produced the best ROR of all average simulations with 39.49%, a 17.25% MDD, Sharpe Ratio of 1.19, RRR of 2.36, and Sortino Ratio of 0.68. From the same subtest and sliding window comes the best simulation with a 52.64% ROR, MDD of 13.75%, Sharpe Ratio of 1.78, RRR of 3.82, and Sortino Ratio of 1.

Through the analysis of all the figures presented in 4.3.2, as the results table 4.3, it is possible to conclude that the fundamental investment case study was able to achieve high profits. Thanks to the weighted F-Score scoring system and the multiple enhancements developed in this work, a simple buy and hold investment strategy proved to be highly profitable. On the one hand, not having technical indicators may have downsides since it allows the portfolio to drop multiple times during declining markets. On the other hand, during a “bull market” the portfolio is able to achieve stellar returns since it does not have a “bottleneck” to stop its profits.

In this thesis, both case studies achieved better returns using a self-improved EA and a static portfolio management strategy.

On the one hand, the self-improved EA results do not come as a surprise. Including the variation operators’ parameters in the chromosome allow the algorithm to perform a more specialized evolution since it adapts these values according to the fitness achieved. Consequently, this feature brings a more efficient evolutionary process and, therefore, higher returns during the testing period.

On the other hand, the static portfolio results come as a negative surprise. During the elaboration of the portfolio management strategies, the dynamic approach emerged as a “logical” solution. Using the best companies from the year before trading in the following year should yield good returns. This idea demonstrated not to be always true since the static portfolio strategy overcame the dynamic strategy.

The main reason for this to happen may relate to market seasonality and the world in general. While during a “bull market”, all companies tend to thrive, during a “bear market”, even the best companies decline. Not only that, but companies who thrive in specific momentums may not prosper in others. The reason behind it may not only be specified in financial statements but may also be connected to the state of the world. Events such as economic tensions between countries, legal agreements, wars, and scandals impact companies’ prices by creating fear or optimism in the investors. This fear/optimism then
produces a significant rise or decline in the companies prices, creating unexpected market momentums. Besides clarifying which weights were employed by the algorithm to perform the best simulations, information from 4.2 and 4.4 combined with 4.5 and 4.6 correlation matrices also allow to understand which variables a trader should give more/less importance to. On the one side, variables such as MARGIN, CR, VIX, and LEV should be given high weights and attention. On the other side, variables like CFO and EMA should not receive so much importance from the investors.

Lastly, a relevant point to notice is that although the training sometimes would yield high ROR that would not necessarily mean that their tests would be that great and vice-versa. Once again, the trading market is a very complex system that is connected to the world in general. Training a system that works flawlessly during several years may result in a good testing performance in some environments. However, it does not imply that it will be able to dominate and entirely surpass the market during testing years.
5

Conclusion

Contents

5.1 Conclusions ......................................................... 85
5.2 Future Work ....................................................... 86
In this work, a new method for maximizing the returns of financial market investments is presented, analyzed and compared to the SP500. The combination of two types of Evolutionary Algorithms with fundamental and technical investment strategies, created a new approach capable of exceeding the financial market returns.

The innovative procedure implemented in this thesis uses Evolutionary Algorithms to, on the one hand, pick the importance of ratios from the F-Score in portfolios composition. On the other hand, to determine the importance of selected technical indicators, that reveal the best timing for market positions placement.

Two types of Evolutionary Algorithms were employed. The first is the most common, characterized by having its parameters predefined and static throughout the whole evolution. The second EA developed is considered “self-adaptive” since it introduces the variation operator’s parameters’ values in the candidate representation, so they are able to evolve and adapt to different environments and provide better optimization.

Lastly, this thesis implements two types of portfolios. The dynamic portfolio updates its companies each trading year, while the static utilizes the same companies throughout the whole trading period.

In this work, the system was tested in an out of sample period belonging to a scheme of sliding windows. This scheme was created to study the best way of training and testing the system to maximize its returns.

The following chapter presents conclusive statements, achievements and some limitations of this thesis. Also, recommendations on what the future work can be are presented.

5.1 Conclusions

The analysis of the results obtained in this work allows concluding that Evolutionary Algorithms, combined with the stock market, represents a powerful tool when it comes to portfolio composition and technical trading. Not only that, but also the capability of using multiple data sources, extracting the critical information, transforming it into usable data to compose portfolios for trading in few seconds demonstrates significant potential in the developed system.

The presented results are promising and reveal that the technical and the fundamental case studies combined with Evolutionary Algorithms are able to surpass the SP500 returns. Both case studies achieved their best results using a static portfolio management strategy and a self-adaptive EA with returns more than two times higher than the benchmark in both average cases. Besides, both subtests also achieved these results in the same sliding window composed of two years of training and two years of testing.

On the one hand, the technical case study revealed its value during “bear markets” since it is able to diminish accentuated declines. Nonetheless, during “bull markets”, the technical case study lacks “con-
fidence” to pursue higher returns. On the other hand, the fundamental case study shows many more declines during the “bear market”. Nevertheless, it shows the ability to “climb” the market at a fast pace achieving significant returns during “bull market” market momentum’s.

From the presented results, it is also possible to conclude which variables should receive more or less importance during a portfolio composition and technical trading so that traders use that information to benefit in future investments.

Overall, the proposed goals of this work were achieved, showing that it is possible to surpass SP500 using not only a fundamental approach but also a technical approach if combined with EA.

5.2 Future Work

In this work, the most significant system limitation is related to the time each simulation took to complete. Although much was done to decrease time complexity, there is always room for improvement. In this subsection, multiple enhancements are proposed for future development:

- In order to decrease the time each simulation takes to complete, a new Python distribution should be considered. PyPy\(^1\) and Python3 are examples of some distributions that revealed (in some cases) to be faster than the Python version used to implement this system [57, 58].

- In this work, the variation operators’ parameters were included in the representation for optimization. In future work, other parameters such as the “percentage of the population that takes part in elitism” could also be included in the representation for optimization.

- Variables such as the “number of generations a population is allowed to evolve” and the “number of candidates solutions to generate at the beginning of the evolutionary process” could also be tested with higher values and fewer time constraints to understand if the system achieves even better results.

- Replacing the developed EA with a MOEA system using two financial ratios to maximize return and reduce risk.

- Implement feature diversification strategies in the EA, such as immigration schemes. Furthermore, with multiple species, the EA could optimize the evolutionary parameters using diversification metrics instead of using financial fitness functions.

- Allow the system to short sell companies during “bear markets” to profit from downturns.

\(^1\)PyPy is a compliant, fast alternative implementation of the Python language. This language has several advantages, such as speed, memory usage, and compatibility with multiple libraries.
• Change from a weekly dataset to the daily dataset when volatility spikes surge. This feature would enable more precise technical analysis during critical moments.

• Explore new technical indicators.

• Implement safeguard mechanisms in the trading simulator for cutting losses such as stop losses and take profits.
Bibliography


<table>
<thead>
<tr>
<th>Date</th>
<th>Close</th>
<th>VIX Score</th>
<th>…</th>
<th>A RSI Score</th>
<th>A EMA Score</th>
<th>A MACD Score</th>
<th>A ROC Score</th>
<th>A MAC Score</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-09-03</td>
<td>20.52</td>
<td>0.5</td>
<td>0</td>
<td>0.5</td>
<td>0.0</td>
<td>0.5</td>
<td>0.5</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>2012-09-08</td>
<td>22.45</td>
<td>0.5</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>2012-09-13</td>
<td>22.44</td>
<td>0.5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.5</td>
<td>0</td>
</tr>
<tr>
<td>2012-09-18</td>
<td>26.52</td>
<td>0.5</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>2012-09-23</td>
<td>27.74</td>
<td>0.5</td>
<td>0</td>
<td>0.5</td>
<td>0.0</td>
<td>0.5</td>
<td>0.5</td>
<td>0</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table A.1: Stock Prices Dataset With Calculated Technical Scores Sample.