Big Data Finance: Trading strategy creation using Deep Reinforcement Learning.

A thesis submitted to the University of Manchester for the degree of Doctor of Philosophy in the Faculty of Science and Engineering

2021

By Hugo Gamboa Valero
Department of Computer Science
School of Engineering

Supervisors:
Prof. John Keane
Prof. Xiao-Jun Zeng

This page is intentionally not left blank.

Contents

Word count: 38000.

List of Figures

List of Tables

Abstract

Deep Reinforcement Learning has been successfully used to control an agent in a high-dimensional space. For this reason, an important question is whether these powerful capabilities can be applied to highly dynamic spaces such as the stock market. This thesis explores how multimodal financial data can be processed to obtain low-dimensional features representing the economic state of a given country and exploited by a trading system to maximize wealth and minimize risks.

Allocating resources is crucial when investing in the stock market, given that errors can be very costly. In particular, it is essential that investors navigate the dangerous waters of the collapsing markets unscathed during an economic crisis. With recent advances in ML, it might be possible to do so by helping investors make better decisions when allocating their wealth in a way that maximizes wealth while minimizing market risks.

In order to achieve these goals, custom loss functions that control the maximum amount of cash allocated to a specific asset while selecting those shares that maximize wealth are combined with a Deep Reinforcement Learning (DRL) system trained to learn stock trading and evaluated using a set of financial metrics, including ROI and the Sharpe ratio. Two loss functions are used: a barrier method that limits the cash allocation to a maximum value of 35% and the same barrier method combined with a penalty method that punishes the trading agent when the action did not add up to one.

When comparing these loss functions to a baseline, they earned higher ROI and obtained better financial metrics. They outperformed the benchmark (S&P 500) and the baseline. In particular, during periods of economic turmoil, they lost significantly less wealth than the benchmark and the baseline models. Conversely, during periods of economic growth, they earned more wealth than the benchmark and the baseline models. Our models developed a diversification strategy that allocated almost an equal amount of resources to each asset in the S&P 500. Based on the analysis of portfolios, this diversification strategy demonstrated that it could earn higher ROI than other strategies and the benchmark and increase the initial capital 12-fold.

This page is intentionally not left blank.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

This page is intentionally not left blank.

Copyright

This page is intentionally not left blank.

Acknowledgements

First and foremost, I would like to thank my supervisors, Prof. John Keane and Dr. Xiao-Jun Zeng. I am grateful for their guidance and expert advice during these years.

I would also like to thank my family and friends for their encouragement and constant support.

This study was done with financial support of the Mexican Council of Science and Technology (CONACyT) under the scholarship number CVU 669499.

This page is intentionally not left blank.

1 Introduction

It is hard to see how any rational man can ever invest

—John Maynard Keynes.

This chapter provides the motivation for this thesis, its aim and objectives, and its contributions. In addition, it also presents the system overview and the thesis structure.

1.1 Motivation

Financial crises have significantly impacted society's well-being as they erode wealth or redistribute it to society's highest echelons [1]. The 2008 financial crisis showed the perils of letting the "experts" manage the world's wealth.

As we move to a world where Artificial Intelligence becomes more integrated into each part of our lives, it might be possible to decentralize investment and empower regular people via ML systems to invest intelligently and prevent the risky behavior and bad investments caused by misleading rumors or disinformation [2, 3, 4].

Unfortunately, ML in the financial markets has not obtained the same high-level performance as in other areas such as computer vision, natural language processing, or control system [5]. This underperformance is probably a consequence of the dynamical nature of interconnected economies and erroneous financial models used to simplify the economic system. An ML system capable of modeling these dynamic economic interconnections without simplifying these relationships more than necessary should design profitable strategies while maintaining reasonable levels of risk and earn higher returns on investment.

At its core, there are two main problems with using ML to invest in the market: 1) how to reduce the high-dimensional space created by the complexities of the economic system, and 2) when is the best moment to enter, remain, or exit the market.

Deep Learning models capable of learning the dynamics of high-dimensional data through their non-linearities might solve the first point. RL models can learn optimal control by discovering which action or actions obtain the highest return at a given state might solve the second point [5, 6, 7].

As a result, the combination of these approaches might help investors allocate their wealth effectively. While Deep Learning (DL) would simplify high-dimensional states, RL would guide the agent through an intelligent exploration of the environment while maximizing earnings.

However, even the best ML model will perform poorly with insufficient or incorrect data. Thus, it is also crucial to obtain accurate data from various sources to offer a complete picture of the global economy. Using those diverse data sources, the ML model might overcome two critical problems of current models: a limited grasp of context and causality. The former is the circumstances that allow us to understand an event or statement [8], and the latter is the connection between cause and effect [9]. These are essential elements when deciding what stocks to select, given the current state of the economy.

For instance, after the system receives information about a plague affecting coffee production in the Asia-Pacific region, it should conclude that coffee yield will decrease. This shortage, as supply may not be enough to cover demand, should impact the market value of affected producers and their competitors, as well as the price of coffee. Ultimately, as prices increase, the earnings of coffee stores whose arabica coffee comes mainly from this region—such as Starbucks—should decrease.

1.2 Aims and objectives

Identifying investment strategies that increase wealth consistently is an elusive task for automated systems. This thesis aims to develop ML models that can learn stock trading by designing a set of loss functions that promote strategies that maximize wealth and encourage low-risk levels. The influence of a range of factors on investment is investigated to assess if an automated system can exploit them to find profitable trading strategies.

The following objectives were identified to achieve this aim:

- Collect financial and economic information from private providers and government agencies.
- Apply feature engineering techniques to the datasets without introducing data biases and determine the procedures that achieve the highest prediction accuracy and ROI.
- Develop NLP models to reduce the textual dataset to a financial dataset and to expand this financial dataset with features created from financial articles such as the sector and listed companies features.
- Add new features—factors, technical indicators, and anomaly signals—to the original datasets and identify which ones have the most substantial influence on the prediction accuracy and ROI.
- Determine which loss functions can better control wealth allocation to maximize

ROI and minimize risk via financial performance metrics and use these losses to develop a DRL system that selects appropriate trading strategies.

1.3 Contributions

The main contributions of this project are:

- Identification of correct feature engineering procedures that handle multimodal data—from collected financial and economic datasets and new features—and avoid introducing biases.
- Development of DRL algorithms that process low- and high-dimensional data and create profitable trading strategies that can outperform some financial experts and systems as evaluated by a set of financial performance metrics.
- Design of loss functions that encourage agents to find investment strategies that vary wealth allocation and promote portfolio diversification.

1.4 System overview

The system architecture is formed by two main blocks: the feature engineering block and the DRL block. They are shown in figure 1.1. While the first block is responsible for adding, removing, and fixing features using different types of datasets, the second block is in charge of investing in the stock market and finding suitable trading strategies.

Before creating the investment system, it is necessary to find adequate ways to acquire data, engineer features, select the ML models, and test them.

First, different numerical and textual data types will be acquired based on the literature review and availability. These datasets will be studied to find the most suitable preprocessing methods for each of them and identify which of those sets contains relevant financial information that an agent can leverage to increase its wealth. These preprocessing methods include correct approaches to normalization so that models do not receive data suffering from a variety of biases such as look-ahead bias, data-mining bias, time-period bias, among others.

Second, multiple ML models will be evaluated by performing tasks such as forecasting, dimensionality reduction, and anomaly detection. With forecasting models, the goal is

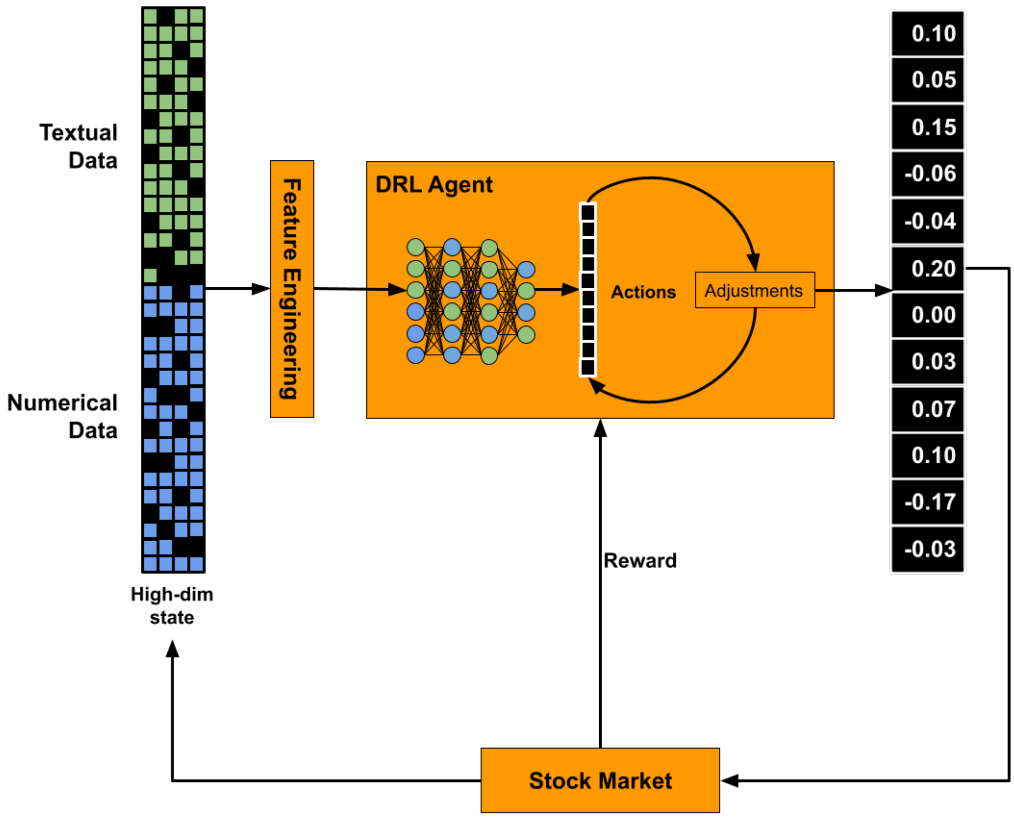


Figure 1.1: System architecture.

to contrast their accuracy, processing time, and earnings. Similarly, for dimensionality reduction and anomaly detection models, the objective is to evaluate only their accuracy and processing time.

Then, this knowledge will be combined to train an RL agent. This agent will explore a trading environment to develop profitable trading strategies with conservative risk levels evaluated using financial metrics.

Lastly, the agent will output a vector representing its wealth distribution over the available assets (including cash). Each value represents a percentage value, and the sign of the value represents whether agent is taking a long or short position.

1.5 Thesis structure

The structure of this thesis is as follows:

- Chapter 2 presents the foundational information used as the basis in this thesis. In particular, knowledge related to Finance—such as the stock market, risk, and portfolio performance—and information regarding ML are included.
- Chapter 3 contains the Finance and ML literature review.
- Chapter 4 includes a description of the types of datasets used for the experiments.
- Chapter 5 contains information regarding the methodology applied in this work.
- Chapter 6 details the experiments and describes and critically assesses the experiments' results and the system's financial performance results once integrated.
- Chapter 7 talks about future work and possible routes to further this research and the conclusions.

1.6 Summary

This chapter detailed the motivation, aim, objectives, and contributions of this thesis, the system overview, and thesis structure.

In chapter 2, key terms and concepts that form the backbone of this research are introduced.

2 Background

Don't confuse brains and a bull market.

In this chapter, key concepts on which this thesis is based are introduced.

Section 2.1 presents a definition of market, risk, portfolio's measurements, and approaches to investing. In terms of markets, an overview of electronic markets and microstructure concepts are included. Regarding risk, a definition and types of risk are briefly mentioned. Concerning portfolios, common types of measurements to compare their performance are included. As for approaches to investing, this subsection introduces fundamental analysis, technical analysis, and quantitative analysis.

Section 2.2 contains a brief description of univariate time series and their components.

Finally, in section 2.3, the ML models used in this thesis, RL definitions, and numerical optimization methods to develop the loss functions are presented.

2.1 Finance

2.1.1 Electronic markets and market microstructure concepts

A market, according to economists, is any situation in which exchange takes place [10]. An electronic market is one where buyers and sellers meet virtually to exchange assets [10].

This thesis studies the U.S. stock market where financial securities are exchanged. Among those securities, the focus is on shares listed on major stock exchanges—i.e., the New York Stock Exchange, NASDAQ, and American Stock Exchange.

A share (of corporate stock) is an equity security that confers ownership in a fraction (or a share) of a company. Corporations issue shares—through Initial Public Offerings or IPOs—to raise capital for diverse economic activities such as renewing equipment or acquiring another company [10].

Once the shares are sold through the IPO at an initial price, traders can exchange them in the secondary market. Any change in price in the secondary market does not affect the company in any way unless the company had acquired some of the shares back—an operation called share buyback or repurchase [10].

Secondary market participants involve corporate managers and traders. Corporate managers participate in the market by increasing or decreasing share supply, while traders are those who exchange these shares [11].

Market makers, who set bids (buy orders) and offers (sell orders) to facilitate exchange in a particular asset, provide liquidity to the market. These agents profit from the [quoted] **spread**, that is, the difference between the price some investors are willing to pay to buy assets (bid price) and the price other investors are willing to accept to sell those assets (ask price) [11].

The minimum distance between these prices (two adjacent price levels) that do not trigger a trade is one *tick*. When these bid and ask prices are equal, they trigger a trade (cross), and the market gets locked [11].

2.1.1.1 Types of Orders

There are two basic types: MO and Limit Order (LO) to send buy or sell orders to an exchange. The former is an aggressive buy or sell order that seeks to execute a trade immediately at the best possible price, and the latter is a passive buy or sell order that attempts to execute a trade at a given price [12, 11].

Exchanges use a record of outstanding limit orders called Limit Order Book (LOB) to keep track of orders. The MOs consume these LOs at the best available price which depends on the type of MO. When the MO is a sell order, the best available price implies the highest price among the buy LOs. Conversely, when the MO is a buy order, the best available price means the lowest price among the sell LOs [11].

Given that exchanges receive millions of orders per day, it is expected to have several LOs at a given level and to receive more MOs than there are LOs at a certain level. Walking the book—or going deeper into the book—refers to the consumption of selling (buying) LOs by buying (selling) MOs at higher (lower) price levels [11].

An example of an LOB is showed in Figure 2.1. The image on the left is the initial state. At time $t+1$, an MO of 250 shares gets to exchange one and consumes the bids at the best available price, i.e., \$23.09. However, there are only 200 shares at that price level which means that, depending on the order type, the remaining 50 shares of that order might consume the 50 shares at \$23.07 (top right image) or be re-routed to another exchange with a better price (bottom right), that is, \$23.09 in exchange two.

In addition to the orders mentioned above, other types include day orders, hidden orders, fill-or-kill orders, good-till-time orders, among others. [11].

2.1.1.2 Types of market positions

At the heart of any investment strategy, there are two primary actions that economic agents can take: long positions and short positions.

Long position: Taking a long position on a stock (or other security) refers to an operation in which an investor buys shares of a stock and sells them later. In this operation, the investor hopes that the stock price increases to earn the difference between the buying price and selling price.

Short position: Taking a short position on a stock (or other security) refers to an

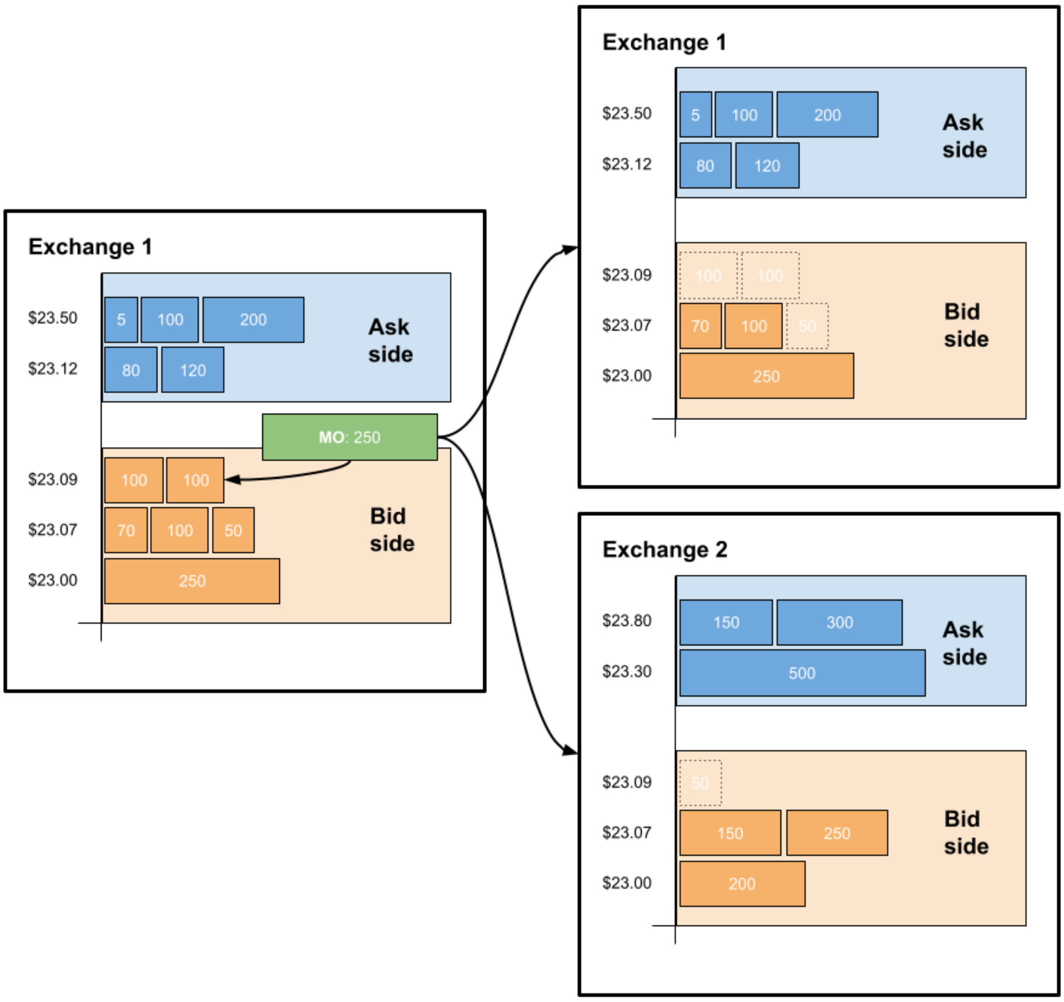


Figure 2.1: Example of a limit order book. An MO gets to exchange 1 and consumes the bids at \$23.09. Then depending on the order, the remaining shares might consume the bids at the next price level (top right) or be re-routed (bottom right).

operation in which an investor borrows shares of a company from a lender to sell them immediately in the market and repurchase them at a later date to return them to the lender plus a loan fee. In this operation, the investor hopes that the stock price decreases to earn the difference between the selling price and the repurchase price.

Contrary to a long position where there is a maximum loss of 100% of the original capital—when the price reaches \$0—a short position does not have any maximum loss limit as the price can grow indefinitely. For this reason and the hazards mentioned in section 2.1.2.1, taking short positions (i.e., short-selling) can be risky if it is not well-thought.

2.1.2 Risk

Investors take decisions where the outcome of some events is uncertain. For this reason, understanding uncertainty can help them make better decisions and earn higher returns. There are two types of uncertain events: risks and opportunities.

A **risk** (an **opportunity**) is a random event that may occur and, if it did occur, it would have a negative (positive) impact on the organization's goals [13].

2.1.2.1 Risk types

There are several types of risks, but they are grouped into three classes [14]:

1. **Systemic risk** (also known as non-diversifiable risk or systematic risk) is the risk that affects the whole market and, as such, cannot be eliminated through diversification. For this reason, this is the only risk that is compensated [15].

2. **Idiosyncratic risk** is the risk that can be reduced or removed by spreading the risk over different assets. In contrast to the systemic risk, the idiosyncratic risk is not remunerated [15].

   Idiosyncratic risk includes credit, political, financial, economic risks, among others—definitions and examples of these risks can be found in section A.1.2 in Appendix A.

3. **Residual risk** is the risk that remains after risk management measures have reduced other risks [16].

2.1.3 Portfolio's measurements

The wealth change and risk level are two dimensions to consider to evaluate a portfolio's performance. The assumptions regarding these dimensions are that more return is favored over less and that less risk is preferred to more risk [17].

Although wealth change or return is a straightforward concept, the second element, risk, does not have a widely accepted way to be measured, and as a result, different proxy measures have been developed to gauge this variable [17].

2.1.3.1 Volatility measures

Volatility measures indicate the amount of data points variation from an average value. In terms of financial markets, they refer to the risk or uncertainty of asset price changes. High volatility means that prices move drastically in either direction. Conversely, low volatility means that prices remain stable without extreme fluctuations. The expectation is that volatility measures allow investors to separate companies whose prices are resistant to positive or negative events from those whose prices move drastically due to these events [14, 13, 18].

Variance and standard deviation

The variance and, by extension, the standard deviation are the foundational operations of other financial metrics such as the Sharpe ratio. While the variance indicates how much data is spread around an average value, the standard deviation measures the average distance between data points from the mean [18, 15, 19].

In particular, in the stock market, the square root of the variance, i.e., the standard deviation, is a risk measure [20, 15]. Moreover, the input data used for the calculation includes stock's returns, prices, and volume.

One limitation is that both operations are poorly suited for forecasting [20, 21].

Portfolio's variance and standard deviation

When there are two or more assets, a portfolio's variance and standard deviation of returns are given by the following formulas [15]:

$$var(R_p) = \sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j cov(R_i, R_j) \tag{2.1}$$

$$\sigma(R_p) = \sqrt{var(R_p)} \tag{2.2}$$

where w_i and w_j are the weight (percentage) of asset i and j in the portfolio p, R_i and R_j represent the returns of assets i and j, and $cov(R_i, R_j)$ is the covariance of these returns.

Beta coefficient

The beta coefficient is another volatility measure that compares an asset or portfolio's volatility to a benchmark—usually the market's volatility. In other words, it is a measure of systematic risk [21, 15, 14].

$$\beta = \frac{Cov\,(R_i, R_m)}{var\,(R_m)} \tag{2.3}$$

where R_i represents the return of asset i and R_m is the return of the benchmark, $cov(R_i, R_m)$ is the covariance of returns of asset i and benchmark m, and $var\,(R_m)$ corresponds to the variance of the benchmark m.

This coefficient indicates how much and in which direction a stock moves when the market moves. These β values are described next. For $\beta = 1.0$, the stock has the same amount of systematic risk as the market and moves in the same direction. For $0.0 < \beta < 1.0$, the stock shows less systematic risk than the market, but the stock prices still move in the same direction as the market. For $\beta > 1.0$, the stock has a higher systematic risk than the market. When $\beta = 0$, it indicates that the stock's prices move independently from the market. Finally, in the case of negative β values, the same logic applies, but the stock prices move in the opposite direction as those of the market [21, 15].

2.1.3.2 Performance measurements

Performance measurements (PM) are values used to compare different investment options. Some PM use only the wealth change over a period of time, while others combine wealth change and risk level. In the second case, a value—named *risk-adjusted return*—is computed based on the change in wealth over time of the investment options in proportion to the level of risk taken by the investor [17].

Return-to-risk methods and benchmarking

Return-to-risk (or risk-adjusted) methods allow investors to compare strategies directly and between their strategies and a benchmark [22, 17].

Benchmarking

A benchmark is a quantifiable standard used, as mentioned above, as an alternative to compare portfolios' returns in a process called benchmarking [22, 17].

This benchmark should be simpler or less costly than the investors' portfolio as the key idea is that using this benchmark alternative requires less effort. Nowadays, there is a myriad of benchmark, i.e., from those attempting to mimic a country's economy such as the Dow Jones Industrial Average or S&P 500 to those based on factors such as some of the Russell U.S. Indexes [17, 23].

The result of benchmarking is a value—termed *excess return*—representing the additional investor's portfolio return over (or below) the benchmark's portfolio return. The *excess return* is defined as [17]:

$$ER = r_i - r_{RF} \tag{2.4}$$

where r_i is the return on asset i and r_{RF} is the risk-free rate.

Return-to-risk methods

The return-to-risk measures used in this thesis are the Sharpe ratio, Omega ratio, and Sortino ratio. As these ratios measure risk and return differently, investors benefit from combining these measures [22, 17].

For further details on these ratios, check section A.1.3 in Appendix A.

Remarks on return-to-risk methods

Given that the Sharpe ratio computes the risk-adjusted return using total risk, it is better suited for evaluating entire portfolios as each of the asset's residual risk might be scattered in a well-diversified portfolio. Regarding the Sortino ratio, it is suitable when returns are suspected of having an asymmetric or skewed distribution or when an investor defines the risk using a specific return target [17, 22, 24].

According to [17], there are two limitations with return-to-risk measures: First, their susceptibility to outliers given their use of averages and standard deviations. Second, the distribution of future returns and past returns might be different. In other words, "past performance is no guarantee of future results."

2.1.4 Approaches to investing in the stock market

The study of stock markets is a complex task due to many interactions occurring within and between these markets.

In the case of stock markets, stock prices are affected by the following factors:

- Traders' actions and expectations;
- Supply and demand of stocks;
- Companies results;
- Sectors performances;
- Intervention from governments through monetary and fiscal policies;
- Regional development; and
- World economic outlook.

The first point presents a crucial problem in modeling the markets because, just as Yanis Varoufakis suggests, "economics is part of the phenomenon it tries to explain" [25]. This means that even if experts had the perfect model of the Economy, it would still fail because traders would adapt to the model's expected outcome.

The most common approaches to investing are technical analysis, fundamental analysis, quantitative analysis, or a combination of them.

2.1.4.1 Technical Analysis

Technical Analysis (TA) involves finding patterns in the market history—including stock prices, volumes, market sentiment indicators, etcetera—to identify trading opportunities [26, 27].

TA is commonly performed using technical indicators and candlestick charts.

- **Technical indicators**: They are calculated using stock prices in an attempt to reflect market psychology.

Figure 2.2: Candlestick chart created using Apple stock prices from 2013-09-17 to 2013-11-25.

The technical indicators used in this thesis are listed in Table 4.1, in chapter 4. For more information regarding their definitions and formulas, the interested reader can visit section B.2 in Appendix B.

- **Candlestick** : A candlestick is an alternative to synthesize assets' OHLC (open, high, low, close) prices and display that information in a way that that investors quickly understand. It consists of a body, shadow, and color defined as follows [28].
  - Body: it is a rectangle formed by a range from the open price to the close price.
  - Shadow: it is a line connecting the high price and low price.
  - Color: it is an indicator of the price movement. When $close > open$, there is a price increase, and the candlestick is green (or white). When $open > close$, there is a price decrease, and the candlestick is red (or black).

A series of candlesticks can be observed in Figure 2.2.

Technical analysis investors believe that candlesticks can reflect market psychology through patterns [27].

2.1.4.2 Fundamental Analysis

Fundamental Analysis (FA) refers to assessing a company's worth by using data that provides information about its financial performance, valuation, management team information, and macroeconomic trends [29, 30].

In order to assess a company's worth, investors use mainly three reports: the income statement, the balance sheet, and the statement of cash flow [29].

- **Income statement**: It details the company's performance during each quarter and year. All income statements indicate, at the very least, revenue, production costs, operating expenses, Earnings Before Interest and Taxes (EBIT), interest expense, taxes, net income, and Earnings Per Share (EPS) [29].

  EPS is a key element that indicates the portion of a company's profit attached to each share. It is calculated by dividing net income by the number of shares that are owned by the public and the company's employees [29].

- **Balance sheet**: It summarizes the company's assets, liabilities, and shareholder's equity at a particular point in time. Assets are things owned by the company—tangible such as buildings and equipment, and intangible such as patents, liabilities refer to debt and things the company owes, and equity represents the capital invested in the company and the profits the company has kept [29].

  The *capital structure*—how debt and equity are used to finance assets—determines a company's financial health and indicates whether or not it can withstand economic turmoil [29].

- **Statement of cash flow**: It analyzes how the cash is distributed into operating, investing, and financing activities. This statement unifies the balance sheet and income statement without the bias introduced by accounting methods and provides an alternative view by tracing the activities in which cash is spent. It is calculated by adding back items that did not use cash to the net income and subtracting those that used cash from the net income [29].

2.1.4.3 Quantitative Analysis

Quantitative Analysis (QA) is the application of mathematics—especially probability theory and statistics—to financial markets to estimate numerical values that represent market behavior, the price of financial assets, or the amount of risk of trading [31, 32]. Quantitative finance has a greater focus on the stock and bond markets. When applied to the former, investment strategies are developed to increase the returns of portfolios and reduce risks. The most common types of strategies are momentum, reversals, based on exogenous factors, econometric, and factor investing [15, 33, 32].

Of these types of strategies, this work focuses on **factor investing strategies** that use one or more factors—i.e., any attribute that can explain the risk and return performance of assets—to select profitable assets [33, 34, 35].

At its core, factor investing states that investors are compensated for assuming risks—instead of for holding assets—and views diversification as a collection of risk factors—instead of a collection of asset classes [33, 34, 35].

The factors considered in this work are momentum, size, profitability, value, and quality factor. Formulas to calculate these factors can be found in section A.1.4.1 of Appendix A.

- **The momentum factor** is a factor based on momentum—the short-term tendency of financial assets to maintain their previous performance, i.e., if they have performed well (poorly), they will continue to perform well (poorly) in the short-term future [35].

  There are two types of momentum factor [35]:

  - **The cross-sectional momentum factor** measures relative performance by contrasting the return of an asset with the returns of assets within the same class.

  - **The trend-following factor**—or **time-series momentum factor**—measures absolute performance by comparing the trend of an asset with itself.

- **The size factor** is an attribute based on market capitalization that explains small-cap stock returns from large-cap stock returns. Size factor depends on the state of the economy. When the economy is growing, small-cap companies earn a higher premium than large-cap companies, but during restrictive periods, the size effect is not statistically significant. This is because small-cap businesses are riskier than large-cap ones as they are more volatile, are more vulnerable to changes in credit conditions, and have lower levels of profitability [35, 36]. As a result, investors demand a higher payout for taking greater risk.

- **The profitability factor** explains why firms with high earnings have subsequent high returns after controlling for Book-to-Market (BtM) ratio and investment. According to Novy-Marx [37], profitability has roughly the same power as BtM predicting the cross-section of average returns, which is why profitability is used instead.
- **The quality factor** explains the quality premium of companies with high margins and asset turnover, and low earnings volatility, financial leverage, operating leverage, and stock-specific risk—known as high-quality companies—over those companies with the opposite attributes—known as low-quality companies [35].
- **The value factor** is an attribute that helps clarify why cheap assets outperform relatively expensive ones.

2.2 Time series

Time series are sequences of observations—or random variables in some definitions—collected over time [38, 19].

They are usually grouped according to the nature of their observations or the number of variables. Based on the data points, they can be classified as continuous or discrete. Similarly, based on the number of variables, time series can be classified as univariate or multivariate [38, 19].

The following sections introduce characteristics of time series which are shared by both univariate and multivariate time series.

2.2.1 Univariate time series

Time series are named univariate time series when there is only a single time series [38, 19].

2.2.1.1 Descriptive measures

These measures are used for describing how time series behave. For univariate time series, the mean function, the autocovariance function, and the autocorrelation function (ACF) are commonly used [38, 19].

2.2.1.2 Time series components

To perform time series analysis and to improve forecasting accuracy, time series can be decomposed into a set of unobservable components that display different types of temporal variations. These components are [39, 40]:

- A secular trend (T_t): it is a long-term increase or decrease in data.
- Cyclical movements (C_t): they are the rises and falls of values with variable periods or frequencies. It is also named business cycle, and it usually involves a quasi-periodic oscillation averaging from three to five years [40].
- Seasonal variations (S_t): they are fluctuations with a fixed and known frequency.
- Calendar variations: they are fluctuations due to calendar events. The main variations are moving holidays, trading days, and other calendar effects.
- A remainder or irregular component (R_t): it is anything not included in the other components. They represent variations caused by unpredictable events such as strikes, floods, data processing errors, etcetera.

The cyclical and the trend component are usually estimated together—and named the trend-cycle component—because the trend's definition as a long-term smooth movement is statistically vague and relative as an estimate may become a cycle as more data is received [40].

2.3 Machine Learning

ML is a branch of Artificial Intelligence that studies algorithms that allow computers to acquire knowledge by extracting patterns from raw data [41]. It comprises a set of methods that automatically detect patterns in data and use them to achieve a particular task, such as classification, regression, forecasting, clustering, etcetera. [42].

2.3.1 ML models

ML architectures are the essence of the investing agents. When these models are trained correctly, agents might identify trading opportunities and develop trading strategies. This thesis uses the following ML algorithms: SVM, RF, and NN, which are described in the following section.

2.3.1.1 SVM

SVM is a popular type of statistical model widely used due to its flexibility and effectiveness as an ML tool that can be used in classification and regression problems [43].

In the case of classification, the model attempts to find a linearly separable hyperplane in the same dimensional space as the training data, but if this hyperplane does not exists, the SVM algorithm moves the input data to a higher dimensional space via a non-linear mapping where the model searches for a hyperplane that separates the data in this new space [43, 44].

In the case of regression, the model maintains the same features as before, but it tries to find a hyperplane that maps the input space to real numbers while minimizing the error [45].

Kernel functions and the kernel trick

Most of the time, data in real-life applications is not linearly separable. A straightforward solution is to apply a basis function on the input data to transform it into high-dimensional data so that these added dimensions capture non-linear interactions within the original data. The advantage of this approach is that the problem stays convex and well-behaved; however, a significant disadvantage is that working on high-dimensional spaces is expensive because algorithms working there become prohibitively slow [46, 42, 47].

The **kernel trick** is a way to solve this problem by creating a function that avoids the high-dimensional transformation and directly computes the results in the original feature space using the inner product of the input data [48, 49, 46, 42, 47].

Kernels

Three popular kernels are tested in this thesis: Radial Basis Function (RBF) kernel, polynomial kernel, and sigmoid kernel.

An **RBF** kernel is a real-valued function which depends on the magnitude of the distance between the arguments so that $\kappa(\boldsymbol{x}, \boldsymbol{x'}) = \kappa(\|\boldsymbol{x} - \boldsymbol{x'}\|)$ [42, 47]. It is defined as:

$$\kappa(\boldsymbol{x}, \boldsymbol{x'}) = exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x'}\|^2}{2\sigma^2}\right) \tag{2.5}$$

Where σ^2 represents the bandwidth [42].

In other words, the kernel takes into account all points, but the nearby observations receive higher weight in the calculation due to the use of the Euclidean distance. When points are closer, κ approaches one, and when points are far away from each other, κ approaches zero.

A **Polynomial kernel** is a type of Mercer kernel that uses a polynomial function to map low-dimensional points into a high-dimensional feature space [42]. It is defined as:

$$\kappa(\boldsymbol{x}, \boldsymbol{x'}) = \left(\gamma \boldsymbol{x}^T \boldsymbol{x'} + r\right)^M, \; where \; r > 0 \tag{2.6}$$

where M indicates the degree of the polynomial, r is a parameter that controls the influence of higher-order and lower-order terms in the polynomial, γ is a parameter used to configure the sensitivity to differences in the input vectors and $\boldsymbol{x}$ and $\boldsymbol{x'}$ are vectors in the input space.

A **Sigmoid kernel** is a non-Mercer kernel that is defined by:

$$\kappa(\boldsymbol{x}, \boldsymbol{x'}) = tanh(\gamma \boldsymbol{x}^T \boldsymbol{x'} + r) \tag{2.7}$$

where γ, r, and $\boldsymbol{x}$ and $\boldsymbol{x'}$ have the same meaning as before.

Algorithm

The SVM is a generalization of the *Support Vector Classifier (SVC)* in which a non-linear kernel is added so that the feature space can be transformed into a high-dimensional space, and a non-linear boundary can be fit to separate two or more classes [50, 42, 44].

The optimization problem has the following equations:

$$max_{\alpha} - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) + \sum_{i=1}^{n}\alpha_i \tag{2.8}$$

$$\text{subject to } 0 \leq \alpha_i \leq C,\ i = 1, \ldots, n, \sum_{i=1}^{n}\alpha_i y_i = 0 \tag{2.9}$$

where y_i and y_j correspond to the class labels, n is the number of observations, α_i and α_j are the support vectors, $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is a kernel, and C is a non-negative regularization parameter.

The SVM general decision boundary for any kernel is given by [51, 44]:

$$f(\boldsymbol{x}) = \sum_{i}^{n}\alpha_i y_i \kappa(x_i, \boldsymbol{x}) + \beta_0 \geq 0 \tag{2.10}$$

where β_0 is a hyperplane parameter, and $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$, n, y_i, and α_i have the same meaning as before.

The use of a kernel helps the model create non-linear boundaries.

2.3.1.2 Random Forest

RF exploits the notion that tree structures vary significantly with minor changes in data by training a set of decision trees—each presented with different versions of the training data—to predict the model's output and averaging the trees (regression) or using their most voted class (classification). This variability results in a model that increases the predictive accuracy over individual decision trees [18, 52, 42].

The critical point of this algorithm is that the decision trees in the model get decorrelated by creating randomized versions of the training data—using bootstrap sampling alone or in combination with subsampling of the observations—and the variables. In the second case, trees are decorrelated because using a subset of m features each time a split is considered prevents the trees from selecting the strongest predictor every time [18, 52, 42, 44].

As a result, the variance of an average of B identically distributed (i.d.) data points, each with variance σ^2, $\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2$—where ρ is the correlation coefficient)—reduces to only $\frac{1}{B}\sigma^2$ when the data points are identically and independently distributed (i.i.d.) [18, 42].

Random forests have three advantages in terms of datasets, performance, and statistical metrics, and one disadvantage involving model complexity.

In the case of data sets, random forests inherit the majority of the benefits of trees, including estimating the importance of variables and handling large data sets with thousands of features, unbalanced datasets, and missing data points [53, 18, 42, 54].

In terms of performance, the algorithm achieves relatively high accuracy for classification tasks while running quickly [53, 18].

Concerning statistical metrics such as variance and bias, creating a model ensemble is especially suitable for high-variance, low-bias methods (such as trees) as averaging these methods significantly reduces the prediction variance [42, 54].

As for model complexity, the main disadvantage of using random forests is that this element increases due to the myriad of splits created by the set of trees. Thus, one of those benefits not inherited from trees is model interpretability [53, 18, 42, 54].

The Random Forests algorithm is described in **Algorithm 1**.

Algorithm 1: Random Forest algorithm for regression and classification taken from [54].

- During the training phase:
  1. For b = 1 to B (number of i.d. data points):
     (a) Draw a bootstrap sample of N elements from the training data.
     (b) Grow a tree T_b with this bootstrapped data until the minimum node size n_{min} is reached by slightly modifying the Classification And Regression Tree (CART) steps:
        i. Randomly select m elements from the p input points (with common values for m ranging from 0 to $\sqrt{p}$).
        ii. Find the best split point among the m data points.
        iii. Split the parent node into two child nodes.
  2. Output the ensemble of trees T_{b1}^{B}.
- During the prediction phase with a new point x:
  - Regression: $\tilde{f}_{RF}^{B}(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$
  - Classification: $\tilde{C}_{RF}^{B}(x) = majority\ vote\tilde{C}_b(x)_1^B$ where $\tilde{C}_b(x)$ is the class prediction of the bth tree.

2.3.1.3 Deep Learning

DL is a subbranch of ML that focuses on algorithms that build hierarchical structures where complex concepts are formed out of simpler concepts.

In DL, models are trained by adjusting a set of parameters using an optimizer. This optimizer is guided by a loss function that generates a value indicating how well the model is learning to perform a particular task.

Most DL models are based on Deep Neural Networks (DNN). They come in various shapes and forms, but at their core, they are formed by different types of layers that have specific functions that exploit the structure within data. The only difference between a regular NN and a DNN is the number of hidden layers; any network with two or more hidden layers can be classified as a DNN.

Fully-Connected (FC) Network

It is a network with fully-connected layers in which each neuron within a layer is connected to all the neurons in the previous layer [42].

Each of the neurons in the next layer receives an input $x_1, \ldots, x_{H_{layer}}$ where H_{layer} is the number of output elements from the previous layer—or the number of input elements in the first layer—and applies a linear combination—or **activation**—of these elements. The activations are controlled by a set of adjustable parameters called weights and are further processed using a nonlinear activation function $f(\cdot)$ to generate the neuron's output.

For the first layer, this output is:

$$f_s(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{f}\left(\sum_{i=0}^{H_1} w_{s,i}^{input} x_i\right) \tag{2.11}$$

where $s = 1, \ldots, H_2$ is the number of outputs, H_1 is the number of inputs, and $w_{s,i}^{input}$ refers to the weights in the input layer (including the biases $w_{j,0}$).

Similarly, for the second layer, the output is:

$$f_{\cdot}(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{f}\left(\sum_{s=0}^{H_2} w_{\cdot,s}^{2} f_s\left(\boldsymbol{x}\right)\right) \tag{2.12}$$

where $\cdot$ refers to the number of outputs in the current layer, $s = 0, \ldots, H_2$ is now the number of inputs, $w_{\cdot,s}^{input}$ refers to the weights in the second layer, and $f_s(\boldsymbol{x}, \boldsymbol{w})$ is the output from the previous layer.

Based on the previous outputs, the overall network function is calculated as:

$$y_k(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{f}\left(\sum_{j=0}^{H_{layer}} w_{k,j}^{layer} \boldsymbol{f}\left(\sum_{r=0}^{H_{layer-1}} w_{j,r}^{layer-1} \boldsymbol{f}\left(\ldots\right)\right)\right) \tag{2.13}$$

where $k = 1, \ldots, K$ with K being the number of outputs, and w, $f(\ldots)$, and H have the same meaning as before.

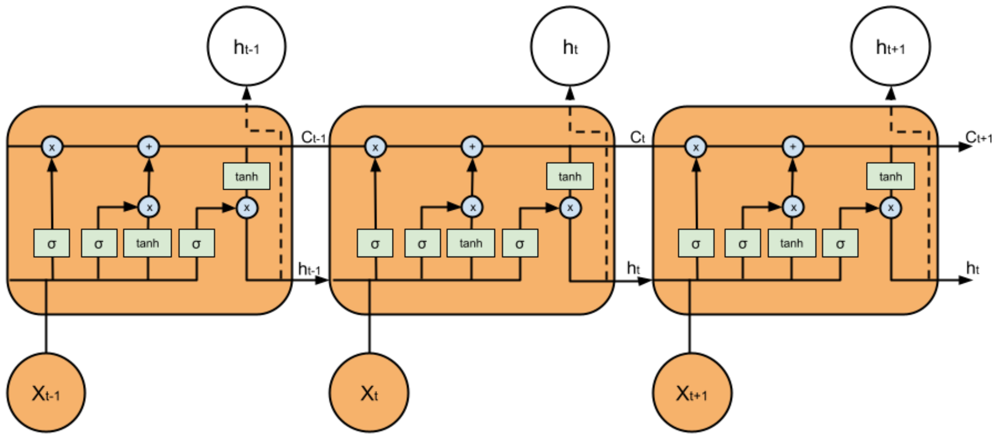


Figure 2.3: LSTM architecture.

Several nonlinear activation functions are available, but those used in this thesis are: the Rectified Linear Unit (ReLU), sigmoid, tanh, swish, and softmax function. Their formulas are listed in Table 2.1.

Table 2.1: Activation functions.

Function	**Equation**
ReLU	$\max(0, x)$
sigmoid	$\frac{1}{1+e^{-x}}$
tanh	$\frac{e^x - e^{-x}}{e^x + e^{-x}}$
swish	$\frac{x}{1+e^{-x}}$
softmax	$\frac{e^{x_i}}{\sum_{j=1}^{Classes} e^{x_j}}$

Long-Short Term Memory Network

LSTM is a type of Recurrent Neural Network (RNN) capable of modeling sequential data such as time series using a set of parameters called states shared across all timesteps to track information from previous steps. LSTM improves over the original RNN by learning long-sequences and reducing the vanishing gradient problem.

The LSTM is shown in Figure 2.3. In this diagram, each line represents a vector from one node to another. Each rectangle represents a NN layer whose parameters are adjusted during training to regulate the gates and add or remove information to the cell state (C). This adjustment explains why all the branches are multiplied by, or directly output, a sigmoid function with range [0, 1]—0 blocks the signal, and 1 allows it [55, 56].

This model differs from RNN in that it adds four gates to the RNN structure: update, relevance, forget, and output gates. The update gate determines how much previous information should affect the current value. The relevance gate specifies whether the previous information should be dropped or kept. The forget gate decides if a cell should be erased or not. Finally, the output gate controls how much information to reveal of a cell [55, 56].

2.3.1.4 Comparison between models

The methods described in the previous sections have different advantages and disadvantages. While tree-based models are more robust in terms of data and have better computational scalability than NN and SVM models, they are only slightly better in terms of interpretability. Conversely, NN and SVM models have higher predictive power and can extract linear combinations of features better than trees [54].

2.3.2 Dimensionality reduction

Most ML algorithms suffer from the curse of dimensionality in which the density of data points around a neighborhood decreases exponentially as the number of dimensions increases [47, 42].

For this reason, dimensionality reduction algorithms can be used to obtain the most important features and reduce the complexity of the data so that the ML models converge more quickly. They are grouped in two major branches: linear projections and manifold learning [57].

Linear projections use linear transformations to map high-dimensional data into low-dimensional data. This group includes algorithms such as principal component analysis (PCA), single-layer autoencoders, and random projection [57].

Manifold learning, or nonlinear dimensionality reduction, methods also map high-dimensional data into low-dimensional data, but they use nonlinear transformations. This group contains algorithms such as multi-layer autoencoders, t-distributed stochastic embedding (t-SNE), among others [57].

Given the nonlinear nature of financial data, a multi-layer autoencoder is used for dimensionality reduction and anomaly detection in this thesis.

2.3.2.1 Autoencoders

Autoencoders are unsupervised learning models based on neural networks used for different tasks such as dimensionality reduction, anomaly detection, and data generation [47, 42, 57].

The model is formed by two parts: an encoder that learns the most relevant aspects of the data and a decoder that reconstructs the original input. In other words, the input and output are the same, and the goal is to minimize the reconstruction error. As a result, the encoder's output contains the mapping to the low-dimensional space.

When designing the model, some strategies constrain the system from learning the trivial identity mapping: 1) An architecture with a narrow bottleneck that makes the model look like an hourglass (as shown in Figure 2.4). 2) An architecture with sparsity constraints on the activation of the hidden units. 3) An architecture in which noise is added to the inputs (called denoising autoencoder) [42].

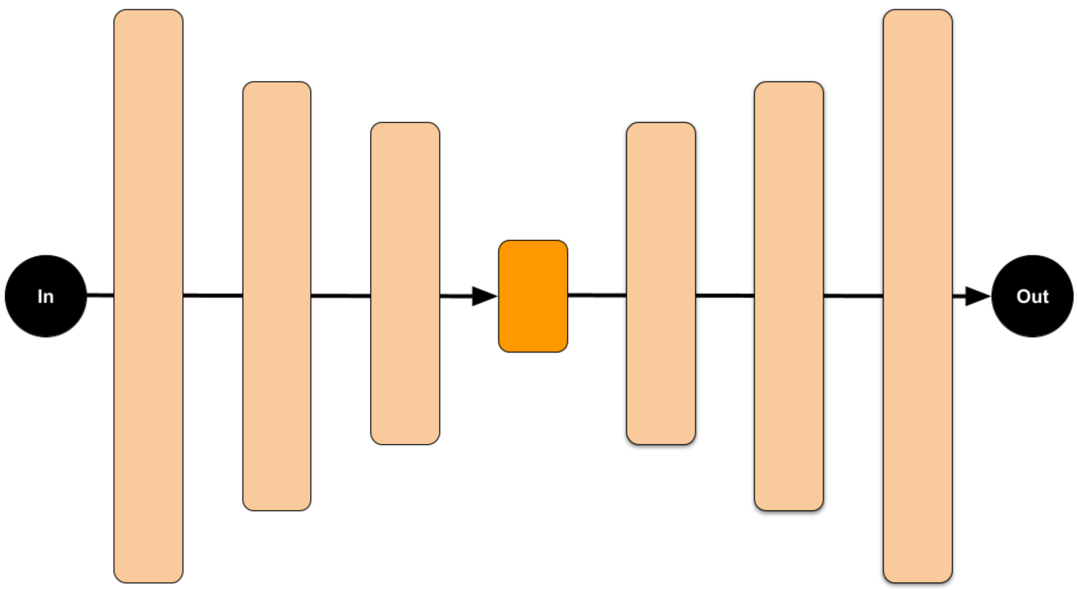


Figure 2.4: Autoencoder diagram. An input passes through the encoder's hidden layers (left side), reaches the center where an encoding is generated, and moves through the decoder's hidden layers (right side), where the encoding is reconstructed.

Usually, the encoder and decoder are symmetrical structures, although this is not a requirement. Moreover, the structures can be based on any neural network, i.e., Convolutional Neural Network (CNN) models, LSTM models, or FC models.

When an autoencoder contains only one hidden layer with linear functions, the weights to the K hidden units span the same subspace as the first K principal components of the data [42].

2.3.3 Anomaly detection

It is a process that involves identifying outliers or unexpected events in a dataset [58]. This process can be supervised, semi-supervised or unsupervised depending on the type of data available, and is usually applied to detect fraud, spam, or pertaining to this thesis, investing opportunities as a result of divergences in asset valuation, salient economic news, volatility, etcetera [47, 42, 59].

Common algorithms used for anomaly detection include isolation forest, cluster-based local outlier factor, histogram-based outlier detection, KNN, and autoencoders.

As defined in section 2.3.2.1, autoencoders are versatile models that use the training data to determine a threshold and to separate normal data from outliers on the test data.

2.3.4 Reinforcement Learning

RL is a branch of ML in which agents learn an optimal set of actions by interacting with an environment to maximize a reward signal. RL systems contain three main elements: a policy (π), a reward signal (R), and a value function ($V(s)$) [7].

A policy refers to a mapping from states of the environment to actions to be taken once the agent gets to those states; a reward signal is a value given by the environment in response to the agent's actions; a value function indicates how good it is for an agent to be in a particular state.

Policies can be categorized as deterministic and stochastic. The former are policies that map a state to one action and are denoted by $\mu : a_t = \mu(S_t)$. The latter are policies that map a state to a probability distribution over actions and are represented as $\pi : a_t \sim \pi(\cdot|S_t)$.

In addition to those main elements, a model of the environment is an optional component that provides information regarding transition probabilities and rewards. Methods that use it are called ***model-based methods***, and those that do not are called ***model-free methods***.

For model-based methods, the environment is given, and the agent focuses on planning. For model-free methods, the environment is initially unknown, but using exploration and exploitation, the agent learns a representation of the environment and a set of actions. Exploitation refers to selecting the action that up to that moment leads to the highest expected reward. In contrast, exploration refers to testing other actions to find an alternative that earns a higher expected reward.

At each time step t, the agent observes a representation of the environment state $S_t \in S$ (where S is the set of possible states) and takes an exploratory or exploitative action $A_t \in A(S_t)$ (where $A(S_t)$ is the set of actions available in state S_t). At time step $t+1$, the agent reach a new state S_{t+1} and receives a reward R_{t+1}.

The sum of these rewards gives the **expected reward**. When a terminal state T exists, it is defined as:

$$G_t = R_{t+1} + R_{t+2} + ... + R_T \tag{2.14}$$

where R_{t+i} is the reward at time $t+i$.

For continuous tasks, it is defined as:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ... = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2.15}$$

where R_{t+i} is the reward at time $t+i$, and γ is a discount factor.

2.3.4.1 Value function

Value functions define a partial ordering over policies which the agents use to select optimal policies. There are two types of value functions: state-value functions, and action-value functions [7].

A state-value function is defined as:

Definition 2.3.1. The value of a state s under a policy π, denoted $V_\pi(s)$, is the expected return when starting in state s and following policy π after that.

$$V_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}|S_t = s\right] \tag{2.16}$$

An action-value function is defined as:

Definition 2.3.2. The value of taking an action a in state s under a policy π, denoted $q_\pi(s, a)$, is the expected return starting from s, taking the action a, and following policy π after that.

$$q_\pi(s, a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}|S_t = s, A_t = a\right] \tag{2.17}$$

In this thesis, agents are referred to as traders. Actions are a set of continuous elements representing percentages of cash allocation. States are representations that summarize financial information regarding stocks, and rewards refer to ROI.

2.3.4.2 Deep Reinforcement Learning

RL agents face the challenge of exploring the state space sufficiently so that an accurate estimate of the reward can be calculated. When the state space is small, exploration is simple, but when it is large, exploration becomes infeasible [7]. DL is an efficient way to ameliorate this problem. It works as a function approximator which means that instead of storing each value, DL learns a function that maps states (or state-action pairs) to their corresponding value [7, 5].

Deep Q-Learning

Deepmind proposed DRL in 2013 as a solution to learning control policies from high-dimensional states [60, 61]. The model uses a CNN to extract low-dimensional representations from the video game frames and approximate a Q-value function for each of the actions.

The algorithm initializes a replay memory—a structure that stores tuples formed by the state, action, and reward at time t and the state at time $t+1$—and a NN. Then, in each episode, an agent explores or exploits the environment according to an ϵ-greedy policy. This policy selects a random action with probability p (exploration) or the best action at a given state (exploitation). The selected action is executed in the emulator to get a reward and a state $t+1$, and form the tuple to store in the replay memory. Next, mini-batches are sampled from the replay memory, and the NN is updated by performing a gradient descent step. This cycle is repeated by the authors for 100 epochs, with each epoch performing 50000 minibatch weight updates.

2.3.5 Optimization

Optimization is the process of minimizing or maximizing a function subject to constraints on its variables [62].

The optimization problem can be defined as [62]:

$$\min_{x\in\mathbb{R}^n} f(x) \tag{2.18}$$

$$s.t.\ c_i(x) = 0, i \in \mathcal{E} \tag{2.19}$$

$$c_i(x) \leq 0, i \in \mathcal{I} \tag{2.20}$$

where f and the functions c_i are smooth, real-valued functions on a subset of $\mathbb{R}^n$, and $\mathcal{E}$ and $\mathcal{I}$ refer to two finite sets of equality and inequality constraints, respectively.

A global minimizer of function is a point where the function attains its lowest value. In contrast, a local minimizer of a function is a point that has the smallest value of a function within a neighborhood.

2.3.5.1 Lagrangian function

The Lagrangian for the constrained optimization problem is defined as [62]:

$$L(x, \lambda, v) = f(x) - \sum_{i\in\mathcal{E}\cup\mathbb{I}} \lambda_i c_i(x) \tag{2.21}$$

where λ_i is a Lagrangian multiplier, and f and c_i have the same meaning as before.

The minimum Lagrangian function is found by applying the ∇ operator:

$$\nabla L(x, \lambda, v) = \nabla f(x) - \sum_{i\in\mathcal{E}\cup\mathbb{I}} \lambda_i \nabla c_i(x) = 0 \tag{2.22}$$

2.3.5.2 Penalty function

Penalty functions are another alternative to convert constrained problems into unconstrained problems. An artificial penalty is introduced with these functions and gets activated when the constraint is violated [62].

The combined objective /penalty function is defined as follows:

$$Q(x;\mu_P) = f(x) + \frac{1}{2\mu_P}\sum_{i\in\mathcal{E}} c_i^2(x) + \frac{1}{2\mu_P}\sum_{i\in\mathcal{I}} \max\left(-c_i, 0\right)^2 \tag{2.23}$$

where f and c_i, $\mathcal{E}$ and $\mathcal{I}$ have the same meaning as before, and $\mu_P > 0$ is the penalty parameter that penalizes constraint violations with increased severity as it approaches 0; the first sum refers to the equality constraints; the second sum involves the inequality constraints [62].

One important problem is that adding penalty functions can create severe slope changes at the boundary and interfere with typical minimization programs [62].

2.3.5.3 Barrier method

The Barrier method is another alternative to represent constraints, but in this case, it is only applied to inequality constraints. This method introduces a **barrier function** $B(x) \geq 0$ that approaches infinity as the function gets closer to any constraints [62].

There are two types of barrier functions: inverse barrier and logarithmic barrier. However, the second is the most important barrier function as it is the one that will be used in this thesis [62].

The combined objective /barrier method is defined as follows:

$$B(x;\mu) = f(x) - \mu_B \sum_{i\in\mathcal{I}} \log c_i(x) \tag{2.24}$$

where f and c_i, and $\mathcal{I}$ have the same meaning as before, and μ_B is referred as the *barrier parameter*.

It is possible to combine the barrier method with the quadratic penalty function to use equality constraints as follows:

$$B(x; \mu_B, \mu_P) = f(x) - \mu_B \sum_{i \in \mathcal{I}} \log c_i(x) + \frac{1}{2\mu_P} \sum_{i \in \mathcal{E}} c_i^2(x) \tag{2.25}$$

where μ_P, μ_B, f and c_i, $\mathcal{E}$, and $\mathcal{I}$ have the same meaning as in the previous equations.

2.4 Summary

This chapter introduced key finance, time series, and ML concepts. Section 2.1 included information related to the stock market, market microstructure, risk, portfolio measurements, and approaches to market investment. Section 2.2 briefly introduced the concept of time series and tools used to obtain some of their properties. Finally, section 2.3 presented the ML models, RL definitions, and numerical optimization functions used in this thesis to train the trading agent.

In chapter 3, the financial and ML literature is reviewed to find the most suitable data, processing, features, and models to develop a successful trading system.

This page is intentionally not left blank.

3 Literature Review

Learn every day, but especially from the experiences of others. It's cheaper!

—John Bogle.

This chapter involves assessing related work in the areas of finance and ML.

The finance section, sections 3.1 to 3.4, presents research challenging the existence of perfect markets and individuals to support the thesis that DRL agents can learn to exploit flaws in the stock market.

The ML section, sections 3.5 to 3.7, explores papers that describe models using high-dimensional spaces to create dense representations of economic information (including financial information), agents trained using low-dimensional states to develop strategies that maximize their rewards, and models that combine DL and RL where the DL block works as a function approximator of low-dimensional states, while the RL block works as a strategy creator.

3.1 Finance

Financial markets have complex interconnections among them. These relations happen among simple assets such as stocks, currencies, or bonds and between cities, countries, regions, and the world, exemplified by financial crises [63, 10, 64]. Researchers have found diverse relations between assets, assets and markets, and markets.

Ftiti et al. [65] studied the relationship between oil prices and stock markets indexes of the G7 countries using evolutionary spectral analysis and wavelet analysis applied to short-term and medium-term data adjusted for inflation. They found that the co-movement of stock return and oil price growth was confined to the short- to medium-term time-scale fluctuations, and that this co-movement was relatively weaker for the long time-scale fluctuations.

Shen and Jiang [66] investigated a correlation between the U.S. markets and the closing prices of the markets that stopped trading immediately before the opening times of the U.S. markets as those markets contain information regarding the U.S. markets' sentiments and their daily trends. The authors found a high accuracy correlation between these markets and the NASDAQ. This result implies that markets are correlated, and the information in one could affect other markets.

In [67], authors researched the effect of macroeconomic news on currency jumps—discontinuous movements—and cojumps using intra-day data sampled at 5-minutes intervals. The currencies included the dollar exchange rate for the British Pound (GBP/USD), Euro (EUR/USD), Japanese Yen (JPY/USD), and Swiss Franc (CHF/USD) and covered the period from January 2005 to December 2010, but unlike previous studies that used low-frequency data, the authors utilized 5-minute interval data due to its ability to better capture the immediate news price response.

The authors found slightly more negative jump events in which the U.S. dollar appreciated than positive jump events in which it depreciated. Moreover, they found that except for four-time points—4:35 am (only U.K.), 8:35 am, 10:05 am, and 6:05 pm—most jumps are evenly distributed across the day, with these exceptions being mostly triggered by U.S. and U.K. announcements. In particular, the 5-minute interval starting at 8:35 am explains 81%, 80%, 89%, and 78% of the jumps in GBP, EUR, JPY, and CHF currencies, respectively, started after at least one U.S. news release.

Regarding cojumps, the authors found that there is a higher probability of a cojump between EUR and CHF (0.078%) than between GBP and EUR (0.034%) and GBP and CHF (0.037%). Additionally, 23% to 44% of bivariate cojumps and 25% to 52% of trivariate cojumps are started by macroeconomic news.

Finally, in terms of news, out of 34 macroeconomic events, the authors found, using a regression model, that only 15 of them have significant explanatory power. Of these 15, the U.S. nonfarm payroll, GDP, trade balance, and unemployment have a negative and significant impact on jump returns of all currencies. The effects of macroeconomic events are not exclusive to currency exchange rate data. They also apply to other high-frequency assets such as stock index futures and bond futures [68].

Financial crises have a significant impact on the economies as they limit credit for agents, affecting the level of activity leading to periods of low growth and recession. These crises initially affect a few institutions or a particular sector of the economy but spread to the rest of the economy by contagion [69, 10].

[70] studied financial network structures and how cascades of failures in one sector are transmitted and amplified in the network. Researchers posit that these contagions occur in stages, starting with companies in one group and extending to others in interdependent groups. Their methodology to calculate the probability of cascades considers the level of company exposure to other companies' risks (integration) and the degree of concentration (diversification)—how much risk is cross-held by other organizations. Researchers examined integration and diversification trade-offs using analytical results of networks with tractable cascade failures and random cross-holding networks simulations.

[69] built a model to analyze one channel of financial contagion in which liquidity preference shock in one region spread throughout the economy. The economy consists of several regions that contain a large number of identical consumers with different consumption preferences that decide whether to deposit on a bank or use their one unit of homogeneous consumption good, and banks that can invest in short-term and long-term assets. The authors excluded from the model other propagation mechanisms such as the effect of international currency markets and incomplete information in agents. Based on their model, the authors state that what leads to contagion is caused by banks hoarding liquidity when there is an excess of demand for it and spillovers resulting from interregional cross-holdings of deposits.

3.1.1 Inefficient markets

Classical economic theories are not based on reality; instead, these theories use a highly restrictive assumption: rational agents can instantly solve complex dynamic optimization problems [3]. This assumption presents an important problem because they fail to predict market behavior accurately [71, 3, 72, 73].

In financial markets, the Efficient Market Hypothesis (EMH) has been suggested to be a valid assumption. EMH states that, at any given time, share prices reflect the relevant information about their corresponding companies and adjust instantly to new information [74]. Researchers and investors have demonstrated that markets are not efficient and cast doubts about the existence of the EMH [71, 3].

Behavioral economists and psychologists have proved the existence of agents who are not completely rational in different experiments [71, 3, 72, 73].

In [71], Camerer describes the results of a two-player game called the ultimatum game that is played as follows: One of the players is given money and asked to make an offer to the other player to split this amount. The other player then decides whether to accept it—which means that both players keep the agreed sum—or not—which means that neither receives anything. Regardless of the country in which this game has been played, the conclusion is the same: the second player refuses the offer if the distribution is not perceived as fair, which goes against the concept of totally rational agents who would accept any offer greater than zero. These results are not affected by increasing the amount of money in play or removing the possibility of the second player rejecting the offer. In addition, results also suggest that people's utility functions take non-financial elements into account, such as fairness, pride, family, etcetera. [75, 71, 3, 72].

In [76], Bondt and Thaler tested the overreaction hypothesis using NYSE common stocks data between January 1926 and December 1982. The test consisted of constructing two types of portfolios; one of them contained the 35 stocks with the lowest returns over the past 36 months—the loser portfolio—while the other had the 35 stocks with the highest returns over the same period—the winner portfolio—which resulted in a total of 16-loser portfolios and 16-winner portfolios. After portfolio creation, they calculated the cumulative average residual return for each portfolio using the next 36 monthly returns and averaged over each portfolio type. The authors showed that the average difference in returns between the loser portfolios and the winner portfolios was almost 8% annually over the test period.

In quantitative finance, there are also voices doubting the existence of the EMH in any of their versions—i.e., weak efficiency, semi-strong efficiency, and strong efficiency. [32] reasoned that information is imperfect as it contains human errors and is asymmetric—i.e., it is not immediately (or at all) available for all market participants due to cost or transfer and processing time. For this reason, he concludes that markets are not information efficient.

3.1.1.1 Market risks and arbitrage

Arbitrage refers to profiting from price disparities in two markets by buying the asset in the cheap market and selling it in the expensive market.

EMH supporters believe that arbitrage—profiting from price disparities in two markets by buying an asset in a cheap market and selling an equivalent asset or the same asset in an expensive market— is not possible due to the Law Of One Price (LOOP). This law states that "identical goods must have identical prices" [3], but it only holds in perfectly competitive markets without transaction costs or barriers to trade [3].

In the real markets, there are different situations of price disparity. Froot and Dabora [77] studied twin securities—stocks listed on more than one exchange—and found that many exhibit deviations from the LOOP theoretical price. The reluctance of investors to close these valuation anomalies results from two types of risks: fundamental risk and Noise Trader Risk (NTR) risk [3] —described in section A.1.2.

Thaler and Lamont [78] examined equity carve-outs and found overwhelming evidence of widespread market mispricing. A carve-out is a corporate reorganization in which a subsidiary is created, but the parent company retains management control.

The highest-profile case the authors examined was the split of 3Com and Palm. In this case, Palm share prices, the same day the split was done, were higher than 3Com prices ($95.06 vs. $81.81), even though 3Com was the most valuable company of the two. This statement is based on two facts: 1) 3Com investors would receive 1.5 Palm shares for every 3Com share they owned as soon as the split was authorized, and 2) the same day the split was done, 3Com still had 95% of Palm shares. This meant that the market valued 3Com shares negatively, but the most interesting result was that the misprice persisted for over two months [3].

3.2 Fundamental analysis

Research on fundamental analysis is limited. Searching for the terms 'technical analysis and financial markets' and 'factor analysis and financial markets' on Google Scholar returns more than 22,000 results for each query, while searching for 'fundamental analysis and financial markets' only returns 13,600 results.

This could be explained due to the small number of data points in fundamental datasets or the difficulty in collecting this data. On the one hand, the former is a consequence of companies' results being published quarterly which contrasts with other datasets that are updated daily. On the other hand, the latter results from either researchers not having access to expensive databases from financial providers—e.g., Bloomberg ($24,000 per year), Thomson Reuters ($22,000 per year), or Capital IQ ($13,000 per user per year) [79]—and having to parse the web in search of this data or some of these services having steep learning curves to obtain the correct information.

Beyaz et al. [80] trained ML architectures to predict the percentage change in the stock price of a given company after 126 and 252 days using technical indicators, fundamental indicators, or a combination of both. The authors used an 80/20% data split with 10 random starting points and constant train/test size of 1892 and 473 data points, respectively. Then, these datasets were used to train and test NN and Support Vector Regression (SVR) models, which were evaluated using the RMSE metric.

On the one hand, the authors found that fundamental indicators outperformed technical indicators regardless of the company's sector. However, a combination of both types of indicators yielded the best-performing model. On the other hand, they identified the SVR model as the superior architecture regardless of forecasting horizon, company's sector, and type of indicator.

3.3 Technical analysis

Several researchers have investigated the profitability of technical analysis using diverse indicators, including Simple Moving Average (SMA), Exponential Moving Average (EMA), Relative Strength Index (RSI), Elliot Waves patterns, among others. These indicators are used as inputs to various models ranging from ML architectures to econometric models [81, 82, 83, 84].

Regardless of the model and the indicators selected, the key assumption is that market behavior and price prediction are based on historical data.

Silva de Souza et al. [84] developed an automated trading system based on moving averages to assess the profitability of these indicators. They created buy and sell signals using the crossover of two series: a long-term and a short-term moving average for a portfolio composed of diverse holdings. The dataset was obtained from the daily closing quotations of 1454 assets traded on the Brazil, Russia, India, China, and South Africa (BRICS) stock exchanges. It covered the period from 2000 to 2016, except for Brazil and Russia, in which it covered the interval from 2007 to 2016.

The authors created 4,428 strategies based on three types of long-term/short-term moving average pairs—SMA-SMA, SMA-EMA, and EMA-EMA. While the short-term element varied from 5 to 40 periods, the long-term element varied from 80 to 120. Each of these 4,428 strategies was tested without transactions and with 2% and 5% transaction costs.

Jabbur et al. [85] used automated market maker agents (see section 2.1.1 for further details) to identify intraday trading opportunities in the stock market. The authors built a discrete-event order book simulator that took into account price priority and time precedence rules, latency between the participants and the stock market, and self trading to assess the impact of *granularity*—time interval used to calculate the candlesticks—and *close variation*—quote spread—on trading results.

Using five liquid stocks from the Brazilian stock market (Bovespa) index, the authors created four types of candlesticks: 1-, 5-, 10-, and 15-minutes candlesticks (see section 2.1.4.1 for further details on candlesticks), and designed strategies based on Elliot Waves patterns that later were implemented in the automatic trading agents.

The authors concluded that small granularity is detrimental to earnings for both types (breakout and correction) of strategies as more *opportunities* increase the number of trades and the risk of adverse stock price movements. Conversely, 10- and 15-minutes intervals earned the highest ROI. As for close variation, the authors concluded that stocks with higher prices get better results with large close variation, while the opposite occurs for stocks with lower prices.

From a psychological perspective, supporters suggest that technical indicators reflect the crowd's mass psychology, which fluctuates between periods of fear or pessimism and intervals of confidence or optimism [84].

However, some of these researchers—Lopez de Prado, Shiller, and others [86, 87]—have questioned the methodology and veracity of the experiments and the rationality of investing using technical indicators.

3.4 Quantitative analysis

Although quantitative analysis includes other types of data, in this section, factor investing papers are reviewed.

3.4.1 Factor investing

Risk is an intrinsic characteristic of investing. As shown in section 2.1.2.1, different types of risks affect assets and their returns. Factor investing is a new type of model from the asset management industry for long-term investment that helps investors select robust and diversified portfolios, monitor risks, enhance returns, and reduce risks. Global institutions have widely adopted this type of investing as nine out of ten investors are using factors in their investment management and a large proportion of institutions plan to increase their use of them over the next few years significantly [88].

In their paper "The death of diversification has been greatly exaggerated," [89] argued that factor diversification strategies reduce portfolio volatility and market directionality more effectively than traditional asset class diversification. They showed that factor diversification could improve the portfolio's Sharpe ratio to 0.7—an increase of 46% over the asset class-diversified portfolio—and reduce the average correlation to almost zero—versus 0.4 of the other portfolio. The factors used in their diversification strategy were the value, momentum, carry, and trend-following factors.

Value factor was made famous by Eugene Fama and Kenneth French, but since 2006 the value premium has been shrinking due to value stocks underperforming growth stocks [90]. However, academic researchers keep finding value premiums and explaining the context in which it appears.

In "the value premium" [91], Zhang noted that the economic outlook influences the value premium. During an economic downturn, value stocks become riskier than growth stocks, but during economic prosperity, the risk decreases so that value stocks are slightly less risky than growth stocks. Yogo [92] made a similar observation suggesting that value stocks depend on the economic outlook and are more pro-cyclical than growth stocks. The author found that value stocks earn low returns during recessions.

Black, Mao, and McMillan [93] examined the value premium price index (VPPI) and economic activity—assessed via macroeconomic variables such as the seasonally adjusted industrial production, inflation, long-term interest rates, and seasonally adjusted money supply. The VPPI is a price index "from buying value stocks and short selling growth stocks" [93] that is calculated as $VPPI = ln(VTRI_t) - ln(GTRI_t)$ where $VTRI$ is the total return index of value stocks, and $GTRI$ is the total return index of growth stocks. The data covers the period from January 1959 to December 2005, resulting in 564 monthly observations.

The authors found cointegration between VPPI and the macroeconomic variables. When the relationship is negative, the value premium decreases in a growing economy given that the value stock prices rise more than growth stock prices and increases in a contracting economy because the value stock prices fall faster than growth stock prices. According to Black et al., the cointegrations are as follows:

1. VPPI and industrial production have a negative relationship.

2. VPPI and money supply have negative cointegration because money supply is suggested to increase stock prices [94].

3. VPPI and long-term interest rates have a positive relation, given that stock prices fall with interest rates rise.

4. VPPI and inflation have a negative cointegration, although it was not statistically significant.

The authors also found that value stocks are affected more by bad economic news, while growth stocks, by good economic news.

In [95], the authors examined the effect of monetary conditions on a three-factor model—pre-ranking beta (β), size or market equity (ME), and BtM equity— and average stock returns. Jensen and Mercer built equally weighted portfolios sorted based on *fiscal year-ends* accounting data from the previous year, starting in July 1965 and ending in June 1997. Portfolios were re-formed at the end of every June using newly-ranked stocks, which were created by applying each of the factors in series, i.e., first creating β-ranked quintiles, then β:ME-ranked quintiles, and finally, β:ME:BM-ranked quintiles. To categorize the Federal Reserve's monetary policy in either expansive or restrictive, authors used three criteria: the discount rate (check section B.4.4 for an explanation), the federal funds rate, and the Boschen and Mills measure—a narrative-based measure that examines the Federal Reserve Open Market Committee records and similar documents to assess the monetary policy stance. Jensen and Mercer found a significant small-cap company premium during expansionary monetary policy periods and no statistically significant effect during restrictive periods. The authors also found that the three-risk factors contribute significantly to explaining cross-sectional returns.

In a paper called "The firm size effect and the economic cycle" [36], Kim and Burnie tested the hypothesis that small firms perform well during economic expansions and poorly in economic contractions. They created ten size-based portfolios using companies' market values in 1976 and repeated this process until 1995. Then, statistics—i.e. mean and standard deviation—of monthly returns for each portfolio were calculated under different scenarios, namely for January-December, only January, February-December, economic expansion and contraction based on the leading indicator from the U.S. Bureau of Economic Analysis (BEA), National Bureau of Economic Research's boom and recession, and monthly bull and bear markets. The authors confirmed previous findings of small firms having greater mean returns than large firms—specially in January—but also greater risks, and found that portfolio mean returns are higher in expansion months, boom periods, and bull markets than in contraction months, recession periods, and bear markets, respectively. Kim and Burnie also noticed a significant size effect in expansion months, boom periods, and bull markets, and almost no effect in contraction months, recession periods, and bear markets.

Yogo [92] discovered that small-cap companies are more pro-cyclical than large-cap companies and earn low returns during recessions when the marginal utility of consumption is highest.

In their paper [96], Asness et al. challenged the notion that the size premium has had a weak historical record. By controlling for the quality of a firm, the authors found that a significant and robust size effect appears in which small quality and small junk stocks outperform large quality and large junk stocks, respectively. Authors used equity data between July 1926 and December 2012 to build decile portfolios based on the Small Minus Big (SMB) (size factor), market capitalization sorts, High Minus Low (HML) (value factor), RMRF (market factor), momentum factor, STREV (short-term reversal factor), and non-price-based measures of size. To build quality and junk portfolios, they used profitability, growth, safety, payout, credit, and investment measures individually, as well as their combinations. Authors found that junk stocks (very small, low average returns, and distressed and illiquid) show a strong negative relation between size and quality, explaining their infrequent size premium. Asness et al. also found that higher-quality stocks were more liquid, while high-beta stocks tended to be more speculative and have poor historical returns as they were those with low-quality attributes.

A more robust factor than the size factor, the momentum factor has been present without diminishing since at least 1801 in the U.S., and the Victorian age in the U.K., in 40 other countries, and across other asset classes [97, 98]. Similar to other factors, there is no consensus about why the momentum premium persists.

In their article titled "Fact, Fiction and Momentum Investing," Asness et al. [98] debunked ten common myths of momentum factor using academic papers from top-level seminars and conferences, as well as data taken from Kenneth French's website. The evidence presented shows that the momentum factor outperforms both value and size over a period of 87-years with a return from 1927-2013 of 8.3% versus 4.7% and 2.9%, and Sharpe ratio of 0.5 vs. 0.39, and 0.26 for momentum, value, and size, respectively. This result persists even when trading costs and taxes are taken into account. In addition, the authors show that some of those myths apply instead to the value factor—e.g., being stronger among small-cap stocks than large-cap stocks with returns per year of 5.9% (small caps) and 3.5% (large caps) over the entire sample period or experiencing return degradation.

Eugene Fama and Kenneth French [99] split international stock returns from 23 countries covering the period from November 1989 to March 2011 into four regions—i.e., North America, Japan, Asia Pacific and Europe— and found strong momentum returns in all regions but Japan, as well as a momentum premium in all size groups, especially in micro-cap stocks.

In [100], Geczy and Samonov studied the momentum factor across six assets—equities, bonds, currencies, commodities, sectors, and stocks—over 215 years, starting in January 1800 and ending in May 2014. Their dataset contains 47 country-equity indices, 48 currencies, 43 government-bond indices, 76 commodities, 301 country-sectors, and 34006 U.S. stocks. For each asset class, the authors built the price-only and total return versions of momentum return time series using a momentum measure consisting of a 10-month change in price with a two-month skip—i.e., $P_{i,t-2}/P_{i,t-12}$.

They confirmed the importance of the momentum premium within and across asset classes.

3.4.1.1 But... what about smart betas?

Smart beta is a term used by some investors and financial institutions to refer to a subset of factor investing techniques. Strategies based on them are a combination of active strategies and passive ones [101, 102, 35, 103].

According to smart beta supporters, given the relatively passive strategies used in smart betas products, it might be possible to obtain greater returns by investing in these *disruptive financial products* than by investing in other low-cost and low-risk stock market index funds. They also claim that any investor can manage a low-turnover portfolio without taking any extra risk, beat the market, and pay lower fees than those charged by active managers as these products are simple and based on transparent rules [101, 102, 35, 103]. Kahn and Lemmon performed a questionable data analysis of a sample of 79 global equity managers from the eVestment database who reported fee data for a $50 million investment and monthly returns over a period of five years—January 2010 to December 2014. They calculated the regression on six smart beta factors using the active returns and found that static exposure to these factors explains the amount of smart beta delivered by active managers [102].

For smart beta detractors [104, 35, 103, 105], these products are marketing gimmicks. They are criticized for including questionable factors—i.e., factors unsupported by empirical evidence—and earning lower excess returns than regular factor investing strategies [104], for being factor investing strategies embellished by the marketing team [35, 103], or for introducing selection bias when designing investment strategies that track a particular risk profile based solely on daily, weekly, monthly or yearly historical market data [105].

In "Is Smart Beta Really Smart?" [103], Burton criticizes smart beta strategies and suggests that most of their claims are false as they can be debunked by examining their results.

The author questions smart beta's three core claims: greater returns, lower risk, and lower fees after analyzing the records smart beta portfolios with real money. The author concludes that once the returns are risk-adjusted, smart beta portfolios do not produce alphas and might even have higher fees than traditional capitalization-weighted index funds once the transaction costs and taxes from the required periodic rebalancing are taken into account. Burton reasoned that these *superior returns* earned by a subset of assets in the smart beta portfolio do not come from investors using traditional broad-base index funds—as they earn the average market return. Instead, these above-average returns come from active managers or from assuming extra risk.

3.5 Machine Learning

There have been different attempts to solve the investment problem using ML. Before the DL boom, most researchers used classical ML methods such as SVM and regular NN to attack this problem with contradicting results regarding which model outperforms the others [106, 107].

An area of improvement in this field is the need for replicability. Some publications lack a proper description of their input data, preprocessing step, feature selection process, hyperparameters selection procedure, or other element of their methodology [106, 107, 86, 105, 108].

The following reasons might cause the partial or missing information:

- Lack of knowledge: Sometimes, knowledge is hidden in obscure journals or books, and other times, it is spread across different fields, but researchers are not fortunate enough to stumble upon it. Thus, there are limits to what researchers know or understand. As Epstein notes in [109]:

  > Only years later . . . did I realize that I had committed statistical malpractice in one section of the thesis . . . Like many a grad student, I had a big database and hit a computer button to run a common statistical analysis, never having been taught to think deeply (or at all) about how that statistical analysis even worked. The stat program spit out a number summarily deemed "statistically significant." Unfortunately, it was almost certainly a false positive, because I did not understand the limitations of the statistical test in the context in which I applied it. Nor did the

scientists who reviewed the work.

- Assumed information: Although these steps are sometimes mentioned in statistical and data science books, research papers omit this information, perhaps because authors assumed it is common knowledge or some journals have a strict limit on the number of words, forcing authors to discard this information.
- Fail to remember: Another possibility is that authors forget to add this information in writing and rewriting their research papers.
- Research misconduct: Although it happens in a small fraction of cases, some researchers are willing to commit research misconduct to pursue career goals, including concealing or fabricating information—a list of scientific misconduct incidents can be reviewed in [110]).

In other cases, research papers use short timeframes of only a few years when using daily prices for their analysis. This bias the results positively (negatively) if the timeframe only covers a period where the economy performs well (poorly) [107].

3.5.1 Deep Learning

In [111], Takeuchi and Lee proposed a model formed by an autoencoder—based on stacked restricted Boltzmann machines—and a feed-forward NN to discover features in time series of stock prices that could predict future returns. The authors used 33 input variables—12-monthly returns, 20-daily returns, and a January indicator variable—for each stock and labeled their data as one or zero based on whether the return over the next month was above or below the median. The median was computed using the cross-sectional standard normalization of 12-monthly- and 20-daily-cumulative returns.

Despite obtaining a low accuracy rate of 53.36%, the model was used to enhance a momentum strategy which resulted in higher monthly returns when compared to a normal momentum strategy—3.35% vs. 1.10%, fewer risky trades, and the creation of a trading strategy that takes the short-term reversal effect into account.

3.5.2 NLP

The use of ML and other statistical models to extract valuable information from financial texts has been an elusive objective for years. As more techniques and new approaches are discovered, more accurate and useful information gets extracted.

For NLP of financial information, these techniques involve simple and advanced feature transformations such as word frequency [112], information extraction techniques [113, 114, 115, 116, 117, 118, 119], event embeddings [120, 121], or word embeddings [122, 123, 124, 125].

For simple transformation, the idea is to pass basic features so that the model performs most of the processing. For advanced features transformation, the point is to help the model by extracting grammatical relationships between objects, compressing grammatical and syntactical information, or synthesizing events so that the ML architecture can focus only on the main task, i.e., news classification, sentiment analysis, price prediction, etc. [112, 124, 121, 125, 123].

Prior to the introduction of word embeddings and the wide adoption of NN, NLP used mostly simple feature transformations. A summary of those methods is included in [126]. In this paper, the authors reviewed the literature on market prediction based on text-mining of unstructured fundamental data and compared the diverse frameworks used to find relationships between the textual information and the economy. Researchers proposed three fields of study to classify the literature: linguistics, machine learning, and behavioral economics. The authors summarized the literature as follows:

- Regarding input data, text mining systems combined textual and market data.

  For textual data, general and financial news sources included the Wall Street Journal, Dow Jones, Bloomberg, Forbes, and Yahoo! Finance. However, most systems used financial news as they contain less noise than general news. In particular, news headlines offered less noise than news text. The rest of the systems performed text mining on social media and blog posts, and annual reports, press releases and corporate disclosures.

  In terms of market data, stock market indexes—e.g., Dow Jones Industrial Average and S&P 500—and stock prices of one or more companies were commonly added as features. In only ten percent of the reviewed articles, FOREX data was used instead. In both cases, time frames vary greatly, with the majority covering only a few years of daily data points and only three covering more than ten years. Similarly, only a minority worked with intraday data ranging from 5-minutes to 3-hours data points.

- Concerning data preprocessing, systems applied feature selection, dimensionality reduction, or feature representation.

  – Feature selection: Three-quarters of the researchers used Bag-of-words (BOW), almost one quarter applied N-grams, and the rest utilized other methods. In the case of BOW, additional features such as noun phrases and name entities were added, with the latter having success identifying named entities on tweets. In the case of N-grams, language-dependency problems were caused by advanced word sequence and syntactic structures provided by this technique.

  – Dimensionality reduction: To fight the curse of dimensionality (see section 2.3.2), researchers employed the diverse techniques: selecting top-N terms, setting a minimum-occurrence-threshold, using a specialized dictionary to map specific terms, applying word stemming, utilizing case-folding to change upper to lower case, and tokenization techniques (See section A.2.1.1.3.1).

  – Feature representation: To represent selected features from the previous step as a numeric value, researchers transformed these features using one of the six most popular methods, i.e., binary encoding, information gain, Chi-square statistics, document frequency, accuracy balanced, and term frequency-inverse document frequency.

- In terms of ML algorithms, most researchers used them for classification, and only a small number of researchers used them for regression. The authors grouped these algorithms into six classes: SVM, regression algorithms, naive Bayes, decision rules or trees, combinatory algorithms, and multi-algorithm experiments.

The next step for NLP was the used of word and event embeddings in an attempt to condense information into vectors.

3.5.2.1 Event embeddings

Nascimento and Cristo [120] proposed combining structured events and numeric features to predict the S&P 500 daily index.

These structured events were extracted from Bloomberg and Reuter news datasets. First, events were created using the Open Information Extraction approach from [127] on those datasets to split them into relation triples of the form E = *(Timestamp, Actor, Action, Object)*. Second, these event triples were transformed in 100-D event-embeddings using the Skip-gram algorithm. Finally, these event-embeddings were added to form a structured event combined with numerical features—formed by a linear combination of S&P 500 daily prices and quadratic and cubic time transformations—to form the input dataset.

The authors tested these features and an alternative version without the event-embeddings using a linear autoregressive model and a random forest—consisting of one thousand 50-nodes trees and compared each version using Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) metrics. They found that adding structured events embeddings decreased the RMSE in 10% (random forest) and 7% (AR), and the MAPE in 6% (RF) and 7% (AR).

Xiao Ding et al. [121] used three ML architectures to model the short-, medium-, and long-term influences of events on market movements. The first model, a neural tensor network (NTN), was responsible for creating event embeddings using Reuters and Bloomberg news so that similar events were mapped to similar dense vectors even when this news did not share common words. The second model, a Deep Convolutional Neural Network (DCNN) was used to perform semantic composition and extract salient global features. As for the third model, a feed-forward NN was used to combine these global features with the stock trends. The authors concluded that event-driven stock prediction earned greater ROI than word-embeddings stock prediction and that modeling semantic compositionality allows the architecture to learn deeper semantic relations between event embeddings.

3.5.2.2 Word embeddings

Authors often use word embeddings in combination with other data types to help a model learn additional capabilities. This is the case with Tsai et al., and Peng and Jiang.

Tsai et al. [112] combined word embeddings created with the (CBOW) Word2Vec algorithm and syntactic information to process 10-K financial reports and find new financial keywords that improve predictability. The authors transformed a lexicon from Loughran and McDonald containing financial-specific sentiment words into word embeddings. For each word in the lexicon, they located its closest 20 words—based on the cosine distance—to expand this lexicon. In addition, to incorporate syntactic information, these words were tagged with their corresponding Part-Of-Speech (POS) tag attachments.

To evaluate whether this expanded lexicon (EXP-SYN) could improve predictability, the authors compared EXP-SYN's predictability of four financial measures to the predictability of other lexicons—the original financial lexicon, a lexicon expanded using LDA, and a lexicon without the syntactic information expansion. The 10-K financial reports were transformed with these lexicons and used to train an SVR model to predict either the post-event volatility, stock volatility, abnormal trading volume, or excess return for each company report.

The authors concluded that their method could effectively discover predictability keywords and capture syntactic and contextual regularities between words.

In the case of Peng and Jiang [128], they trained a DNN model to predict future stock movements by combining textual features extracted from the Bloomberg and Reuters datasets and price information.

On the one hand, the financial datasets were preprocessed by splitting each article into sentences. Then each sentence was labeled—once for each company mentioned in it—with the date of the article, the company and a label ("positive" or "negative") according to the next day closing price, or discarded when the sentence did not contain any company. Lastly, labeled sentences were grouped by date and company name to create the samples.

Using these samples, the authors tested models using the following extracted features: bag of keywords (BoK), polarity scores, and category tag.

Word embeddings were created for the BoK by manually selecting *seed words* used to find 1000 similar words—these seed words were believed to have a strong indication of price movement, e.g., surge, rise, drop, plunge, etc. Polarity scores were computed to measure how each word was related to price movements. As for the category tag, a list of categories was manually defined and expanded using the top 100 closest words.

In the case of numerical features, samples containing the closing prices of the previous five days per target date is standardized (see section 4.2.1.2) based on the mean and variance of these closing prices in the training dataset. Then, the first- and second-order differences were attached to the dataset resulting in a feature vector of the form $(P, \Delta P, \Delta\Delta P)$.

The authors found that the DNN structure, combined with all the extracted features, yielded the best performance.

3.5.2.3 From word embeddings to Bidirectional Encoder Representations from Transformers (BERT)

Word embeddings were popularized by Mikolov et al.'s Word2Vec [129] and Pennington et al.'s Glove [130]. Although they were powerful models, an important disadvantage was that they could not represent polysemy—coexistence of multiple meanings for a word or phrase [131].

Researchers solved this problem with the Embeddings from Language Models (ELMo) architecture [132]. This architecture creates contextualized word embeddings containing complex word characteristics such as semantics and syntax, and linguistic context via a two-layer bidirectional Language Model (LM)s (biLMs) [122, 132]. Each biLMs has character convolutions in which each layer contains forward and backward LSTMs—4096 units and 512 dimension projections.

The output of each layer are intermediate word vectors created by combining the forward LSTM—containing information about the words and the context in the past—and the backward LSTM—containing the same information from elements in the future. The weighted sum of these two intermediate word vectors and the raw word vectors become the final ELMo representation [122, 132].

The model is trained to predict the next word in a sequence of words as an unsupervised task. With this, the pre-trained ELMo model can be added to a wide range of NLP architectures and improve their metrics[122, 132].

Given that ELMo contains bidirectional LSTM, it suffered the vanishing gradient problem [133].

Vaswani et al. [134] introduced a network architecture, named Transformer, that is based on an attention mechanism. This mechanism solves the vanishing gradient problem and, as a result, allows learning dependencies between distant words while reducing the number of operations required to relate signals from two arbitrary positions in the sequence.

The architecture is based on an encoder/decoder structure where the encoder maps an input sequence of symbol representations to a sequence of continuous representations, while the decoder maps these continuous representations to an output sequence. To compensate for the loss of positional information, they used sine and cosine functions of different frequencies to attach that type of information to the embeddings.

Their model outperformed previous SOTA models in different NLP tasks.

In [123], the authors developed an NLP model based on the papers mentioned in this section. The authors designed a deeply bi-directional model based on the transformer model. Similar to ELMo, the point of using a bi-directional model was to use right-to-left and left-to-right contexts.

The model was pre-trained on unlabeled data over different pre-training tasks: *masked language modeling* where the model predicts randomly masked words using only their context; and *next sentence prediction*, where the model predicts the likelihood that a sentence B followed sentence A.

Two models were created, *BERT base* with 12 layers (or transformer blocks), 12 attention heads, and 110 million parameters, and *BERT large* with 24 layers, 16 attention heads and, 340 million parameters.

The BERT model can be used to create contextualized word embeddings or as a block in NLP tasks such as sentence pair classification, single sentence classification, question-answering, and single sentence tagging.

The authors concluded that BERT is effective for fine-tuning and feature-based approaches as it competes with SOTA methods.

3.5.2.4 BERT applied to Finance

Yang et al. [125] pre-trained a BERT model on financial-domain specific corpora and tested it on three financial sentiment classification tasks. The corpora used by the authors included the corporate reports 10-K and 10-Q, earning call transcripts, and analyst reports with a total of 4.9 billion tokens.

The three financial sentiment classification tasks included two sentiment label prediction tasks (a neutral, positive, or negative label) and one sentiment score prediction (a value from -1 to 1). For the first two tasks, the financial phrase bank and the analystTone dataset were used, and for the third task, the FiQA dataset was used.

The authors compared FinBERT against BERT and concluded that their uncased FinBERT model (trained from scratch) achieved higher accuracy over both uncase and case BERT models.

In another paper similarly titled "FinBERT: Financial Sentiment Analysis with Pre-trained Language Models" [124], Tan Araci trained a BERT model to perform financial sentiment analysis.

The author also used the FiQA and Financial Phrase Bank datasets to perform financial sentiment analysis, and pre-trained the FinBERT model using the TRC2-financial dataset from Reuters. Then, the FinBERT model was compared to two other pre-trained language models, ULM-Fit and ELMo, using three metrics: accuracy, cross-entropy loss, and macro F1 average.

The author showed that FinBERT was capable of outperforming the other models across all metrics.

3.6 Reinforcement Learning

In [135], Ritter adapted an RL agent's reward to be a function of wealth increments to maximize the utility. The author developed a market simulation with the following characteristics: it obeyed the Markov property; it took into account arbitrage, trading costs, impact costs, and the mean-reversion effect; trade size was limited to a maximum of K lots which made the action space the set of values from $-K, -K+1, ..., K$; the state space was defined by the price—ranging from 0.1 to 100 with a tick size of 0.1—and a maximum position size of M stock lots—ranging from $-M \ to \ M$.

The Q-learning agent was trained using the market simulation. At each step, the agent acted on the market (by buying, selling, or keeping the same stock lots). When a trading operation was performed, trading cost and impact cost (the effect a trade has on the stock price) were applied, and a reward was given.

The author noticed that the agent designed an *arbitrage-like trading strategy* and learned other characteristics of the environment—such as arbitrage, trading costs, or mean-reversion—directly from the rewards without explicitly receiving information about them. They concluded that an RL agent is capable of maximizing the expected utility in a model-free context.

Tan et al. [136] combined an Adaptive Network Fuzzy Inference System (ANFIS) with an RL to trade. The system used cycles to identify trends, an ANFIS network to predict inflection points in these cycles, and an RL agent to select the best trading action. The system achieved an ROI of 240.32% in 13 years, which outperformed the market by around 50%.

A reusable element of this work is the optimization of the RL agent to maximize ROI.

Deng et al. [137] combined a fuzzy logic system, a DNN, and RNN to teach an agent to generalize financial signals.

The fuzzy logic block reduced uncertainty in data introduced by speculation, the global economy, financial rumors, etc. This block received a feature vector formed by the most recent m return values and automatically learned data membership to one of three fuzzy groups—increasing, decreasing, and no trend. In this case, $m = 50$ with 45 features formed by raw price changes of the last 45 minutes and five features formed by the 3-hours-, 5-hours-, 1-day-, 3-days-, and 10-days momentum changes.

The output passed to the DNN block, which provided feature learning capabilities to the framework and outputs a 150-dimension vector. This vector is the input to the RNN block in which the trading actions—buy, sell, or neutral—are learned.

The system earned greater ROI when compared to a DCNN, an RNN, and an LSTM. This paper showed that a DRL system could learn to select trading actions that maximize ROI despite the uncertainty in the input data.

3.6.1 Partially Observable Markov Decision Process

Brown et al. [138] developed a system, named Libratus, capable of playing heads-up no-limit Texas hold'em. In this type of poker game, each of two players seeks to obtain the best hand after being dealt two cards face down, and then five community cards face up—in three stages. Rounds of betting—in which players can check, call, raise or fold—take place before revealing the community cards and after each stage [139].

Libratus consists of three modules: the blueprint module, the nested subgame solving module, and the self-improver.

In the blueprint module, the game is abstracted by reducing the betting space (action space) and hand space (state space). The action space is reduced by considering only $100-dollars increments, while the state space is simplified by grouping similar poker hands into abstract buckets.

Using this abstraction, the AI plays simulated games of poker against itself using an improved version of an algorithm called Monte Carlo Counterfactual Regret Minimization (MCCF). These simulations create a tree in which each node contains a regret value representing the AI's regret of not selecting that action in the past. As more games are played, these regret values approach zero. The regret values control the exploration by either skipping unpromising actions with very negative values or selecting high-value branches with higher probability.

The second module constructs a more detailed abstraction and solves this subgame in real-time, making sure that this solution fits within the larger blueprint strategy without assuming the opponent's strategy.

An important point is that subgames cannot be solved in isolation as the optimal strategy might depend on other branches in the tree, and for this reason, it is not possible to discard other branches as the game moves down the tree. This contrasts with AlphaZero as complete information games can discard branches in which the game is not being currently played.

The self-improver module enhances the blueprint strategy by interacting with human poker players and filling missing branches of the tree based on the strategies they often use.

Researchers found that Libratus beat top professional poker players by adapting to human strategies and using unpredictable strategies such as huge overbets, many different bet sizes, or even "donk betting"—a type of bet that usually inexperienced players make.

3.7 Deep Reinforcement Learning

In [140], Foerster et al. proposed a Learning with Opponent-Learning Awareness (LOLA) architecture in which the learning rule explicitly accounted for the impact of other agents' parameter updates. The authors modeled different scenarios from those in which the LOLA agent observed another agent's update rule to those in which the LOLA agent used another agent's trajectories to estimate its behavior.

In addition, a higher-order LOLA was proposed to model the behavior of other agents by assuming that those agents used a first-order LOLA learning rule.

It was found that LOLA agents learned to cooperate with high social welfare even though they only knew that other agents were trying to maximize their return. In other words, agents found the optimal cooperative solution (according to game theory) in each game without being programmed to cooperate. In addition, the authors concluded that adding higher-order LOLA led to suboptimal solutions.

In [141], Tampuu et al. used an NN as a function approximator to train an RL agent to play pong alongside another agent without explicitly modeling the agents. The authors used similar methods to other game-playing papers, such as replay memory and frame skipping. While the former is a method that consists in storing experiences—formed by tuples (state, action, reward, new state)—and uniformly sampling these experiences during training to reduce learning instability, avoid local minima, and to ensure that sequences are uncorrelated; the latter is a method that samples one frame every n images to provide agents with enough information to discern motion.

A wide array of game statistics were collected to study the agents' game style as they varied the reward. The statistics included the average paddle-bounces per point, the average wall-bounces per paddle-bounce, and the average serving time per point. It was found that agents moved from fully cooperative—when the reward was -1—to fully competitive—when the reward was +1—by analyzing the statistics and noticing that competitive agents used the upper and lower edges to bounce the ball while cooperative agents did not bounce the ball. These actions suggest that agents are aware that bouncing the ball increase game difficulty.

In a series of papers, Silver et al. [60, 142, 143] showed that by combining DL and RL, it is possible to teach an agent to play a variety of games by improving iteratively after each game.

In [142], an RL agent uses DNN and a Monte Carlo Tree Search (MCTS) to learn, tabula rasa, the game of Go. The DNN is formed by residual blocks containing convolutional layers and is fed a raw board representation of the position and history to produce the probability of selecting each move and a scalar value indicating the player's probability, in turn, of winning from the current position. The MCTS uses DNN output to guide its simulations to compute a vector of search probabilities recommending moves to play. This vector is proportional to the exponential visit count for each move.

It was found that agents achieve higher performance than humans, and can discover new knowledge by discarding inefficient moves in favor of better ones.

In [143], a general RL algorithm named AlphaZero is used to train an agent tabula rasa in the games of chess and shogi. The agent learns by self-playing games guided by a DNN and an MCTS just as in the previous model; however, AlphaZero is different in three ways: there is no data augmentation given that chess and shogi board positions are not symmetric; it optimizes the expected outcome taking into account three values (win, lose, draw); and a single NN is updated continually, but noise is added to prior policies to ensure exploration.

AlphaZero defeated previous Go algorithms, as well as state-of-the-art chess and shogi programs. In addition, it was reported that AlphaZero independently discovered and played the most common human openings [143].

In a paper titled "Adversarial Deep Reinforcement Learning in Portfolio Management" [144], the authors implemented three continuous RL algorithms, i.e., Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO), and adversarial Policy Gradient (PG) to manage a portfolio consisting of $m+1$ assets (including a risk-free asset). Their assumptions regarding the market were as follows: a continuous market—i.e., $open\ price_t = close\ price_{t-1}$—having the Markov property, daily reallocation at the end of the day, and the existence of transaction costs.

The authors used a dual Deep Q-Network (DQN) with experience replay to stabilize the DDPG training. Nevertheless, the DDPG and PPO models had an unsatisfactory performance during training as they could not find optimal policies. Liang et al. proposed adversarial training to solve this problem in which random noise—$\mathcal{N}(0, 0.002)$—was added to market prices, and the objective function was modified to include the volatility of asset returns.

The authors performed their experiments using China stock data from which five assets with at least 1200 trading days were randomly selected. This data was normalized using the last day of the period's closing price, and the missing data points for holidays and weekends were filled forward using the close price on the previous day for prices and zero for volume.

Their backtest results show that the adversarial PG outperforms the other methods and metrics. However, they noted that performance was unstable due to DRL being susceptible to the data quality and noise.

Jiang et al. [145] developed a financial-model-free RL framework for the portfolio management problem. The framework comprises an Ensemble of Identical Independent Evaluators (EIIE) topology, a Portfolio-Vector Memory (PVM), an Online Stochastic Batch Learning (OSBL) scheme, and a reward function.

The first element represents a group of NNs that evaluates an asset's short-term growth potential based on its history and output weights for the next trading period. Three models were used as EIIE, i.e., CNN, RNN, and LSTM. They are trained using an OSBL scheme because it is compatible with pre-trade training, online training during backtests, and online trading. The third element, PVM, records the portfolio weights of each period. As for the reward function, it represents the average of the periodic logarithmic returns.

Although their framework can be used in any market, the authors applied it to the cryptocurrency market. The trading period is divided into intervals of equal length—30 minutes—and within each period, OHLC prices are recorded. At the beginning of each period, funds can be reallocated across m assets, including a cash asset—i.e., Bitcoin in their research—by buying or selling assets at the opening price of that interval plus transaction fees. When $portfolio_t - portfolio_{t-1} > 0$, the reallocation is a buy operation, and when $portfolio_t - portfolio_{t-1} < 0$, it is a sell operation.

In addition, the authors assume zero slippage and zero market impact. The former means that the cryptocurrency market is liquid enough so that each trade can be executed immediately at the price when an order was placed. The latter refers to the negligible effect of the capital invested on the market.

Similarly, in a paper titled "Financial Trading as a Game" [146], Huang developed a trading system for 12 currencies using a deep recurrent Q-network. The exchange market was modeled as a Markov Decision Process (MDP) with states containing 198 dimensions: a) three time-features for minutes, hours, and days of week encoded using a sinusoidal function; b) 192 market features formed by the eight most recent log returns on closing price and tick volume normalized using a 96 period for each of the 12 currencies used; and c) three position-features indicating the agent's current position.

The author used tick-by-tick forex data from January 2012 to December 2017 resampled to 15-minutes intervals and an LSTM architecture whose weights were initialized using a Gaussian distribution $\mathcal{N}(0, 0.001)$.

The author found that the agent stayed profitable under most simulation settings and that the strategies discovered had low or no correlation with the baselines.

3.8 Common problems in this research area

Research in this area is prone to overfitting, which is a serious problem that explains why investment models and strategies that look good on paper often underperform in practice [105].

In this thesis, overfitting denotes two definitions: the case where an ML model memorizes the training data and fails to generalize, and the case, referred to as backtest overfitting, where too many variations of investment strategies are tried, relative to the amount of data available leading to false positives.

While it is clear how backtest overfitting occurs, in finance, the first situation appears in the following situations:

- Training complex models that end up memorizing the data. For instance, using a large number of neurons overfits the data quicker.
- The usage of models that converge to a solution without guaranteeing that it is a global optimum or does not overfit, like the SVM model.
- The application of methods that do not ensure i.i.d. data, such as standard cross-validation methods.

Some alternatives to prevent it are listed next [147]:

- Applying **early stopping** ensures that ML models stop training if the loss function on the validation set has not improved after a specific number of iterations (i.e., tolerance).

- Using **bagging** (bootstrap aggregation) can reduce the variance in forecasts. It involves creating N training sets using random sampling with replacement, fitting N estimators (one per training set), and averaging the individual forecasts (or using majority voting in the case of categorical variables) from the N models. RFs include a similar mechanism to bagging but add a second layer of randomness, i.e., node split, with the purpose of further decorrelating the estimators.

- Adding dropout layers in NNs decorrelate neurons that fire together by randomly turning off neuron connections during training.

- Removal of overlapping periods to reduce or eliminate leakage—when the training set contains information also appearing in the test set.

- Running the experiments a limited number of times.

- Using different seeds for random initialization to reduce the likelihood of getting nice-looking results by chance.

- Applying cross-validation for time series to avoid introducing look-ahead bias and giving the model future information beforehand.

Lopez de Prado and other researchers have questioned these, and other problems in a series of papers [86, 105, 108]. In essence, any scheme based on searching over a large set of strategies is susceptible to overfitting, given that the probability of selecting a false positive and making backtest results worthless increases as the number of trials grows. For this reason, failing to report the number of these trials is misleading. This comments extends to any tasks that use *low-frequency* historical data (daily, weekly, monthly, or yearly) to discern patterns, such as technical analysis, or to design investment portfolios with a particular risk profile, such as with smart beta [105].

In this thesis, overfitting is reduced by using: early stopping, running experiments a limited number of times, applying cross-validation for time series, using different seeds for random initialization, adding dropout layers, and using models—such as RF—that create multiple estimators.

3.9 Summary

Based on the literature review, it is possible to conclude that:

There remain researchers who justify the existence of EMH by testing it under unrealistic and restricted market scenarios, and rationalize their support with statements such as "strictly speaking the EMH is false, but in spirit is profoundly true" [148]. However, EMH does not appear to exist under regular conditions as elements such as asymmetric information, human differences in intelligence (due to genetics, economic disparities, and other factors), corruption, etc., seem to make the market skew away from the efficient market. As such, investors can take advantage of irrational investors and inefficiencies in the markets.

Economies and markets are interconnected locally, regionally, and globally to other economies and markets. Although the level of correlation varies among economies and markets and depends on different economic variables, researchers suggest that elements such as shares, stock market indexes, currency exchange rates, macroeconomic indicators, and news can help investors identify these relations.

Data points from these elements are collected at different intervals ranging from seconds to trimesters. Although it is suggested that high-frequency currency data captures the immediate price response of macroeconomic news more accurately than low-frequency data [67], most datasets in this thesis are low-frequency quarterly, monthly, weekly, and daily data. For this reason, combining high-frequency data with low-frequency data might bias the models towards the data with more data points.

In addition, data has two main types: numerical and textual datasets. In this thesis, numerical data includes macroeconomic information, fundamental indicators, technical indicators, quantitative information, while textual datasets include only financial news.

In terms of macroeconomic events, the literature suggests that some of them influence asset movements. These macroeconomic indicators include the GDP, unemployment rate, the Federal Reserve Bank (FED)'s rate, etc. For this reason, in this thesis, macroeconomic variables—described further in section B.4—are also considered in the model.

Concerning fundamental indicators, research is limited but it suggests that fundamental indicators have better performance than technical indicators.

In terms of technical indicators, research has had mixed results. Supporters of technical analysis suggest that it is possible to identify trading patterns in data and apply these patterns to the market to obtain better profits than other methods such as buy and hold. This part is confirmed by most research only when trading costs are ignored. When these costs are taken into account, profitability vanishes except for the Taiwanese, Mexican, and Thai markets [83, 84].

In some cases where profitability is maintained after considering costs, research suffers from data mining bias as several combinations of the technical indicators are tried [82, 83]. This is the case, for instance, in [84], where the authors tried 4,428 strategies to find profitable long-term/short-term moving average pairs that outperformed a buy and hold strategy.

Given that these strategies are fixed, perhaps giving technical indicators to an RL agent might be useful. For this reason, these indicators are also included in the dataset.

As for quantitative information, factor investing is said to earn higher returns than technical and fundamental analysis because it selects assets based on the type of risk instead of the type of asset. This approach results in a drastic reduction in the average correlation among assets—some researchers affirm that this correlation is almost zero [89].

In the case of textual data, several models have been used to extract information. However, a problem in most financial models is the level of noise introduced by considering the whole dataset. Usually, researchers reduce this noise by only keeping the financial news [126]. For this reason, that alternative is also preferred in this thesis.

Results from Xiao Ding et al. [121] suggest that extracting complex semantic relationships—in the form of event embeddings—from financial news perform better than using regular word embeddings. However, transformer-based embeddings are becoming more prevalent given their SOTA metrics [149] and seem like a viable alternative to those embeddings.

Moving to the ML research, there seems to be a reproducibility problem as researchers in the area have failed to report their whole methodology, including data retrieval, preprocessing, hyperparameters selection, to name a few. In addition, some of these models are tested using a small timeframe of a few years—in some cases, during economic expansion—which calls into question some of their conclusions [86, 108].

Despite these problems, ML models have been used to forecast different economic elements such as stock prices, currency prices, etc., with mixed results. In the case of numerical financial data, the models most commonly applied are SVM and NN, with inconclusive results on which model outperforms the other. In this thesis, SVM, RF, and NN architectures are evaluated to find a suitable *brain* for the trading agent.

In terms of text mining models, most publications utilize a small set of data types, i.e., stock prices and FOREX data [126]. This might impede an ML model from discovering market dynamics. Moreover, most of them cover only a limited number of years—fewer than five years—which call into question the models' generalization capabilities.

Some researchers have opted to use market simulation. Although they are ideal for testing RL investing agents, caution should be taken with some assumptions made during the implementation of these simulations. In particular, with random price movements and trading costs.

The former is an unrealistic assumption, given that not all price movements occur randomly. Events such as the publication of tweets from relevant people, the modification of credit ratings, the release of economic indicators, among other causes, can (and often do) modify market prices. In some cases, even the time or date in which these events get published does not happen randomly, especially with government data given that a publication calendar is released months before the publication date.

The latter, in some cases, is faulty. For instance, in [135], the author added the trading impact cost and assumed that this cost always increases stock prices after each trade. However, in those cases of decreasing price movements, such as during panic selling, prices are not affected by buyers but by sellers, and the impact cost is negative as bid orders are being consumed.

With the wide adoption of deep learning and RL and new techniques, such as MCCF and MCTS, it has been possible to solve complex games—i.e., poker, go, chess, Atari games, etc.—by drastically abstracting high-dimensional spaces to maximize a reward signal. Thus, it seems reasonable to see stock trading as a game with partially observable states and use deep learning and RL to solve it.

In the next chapter, the methodology used to assess all the information suggested in the literature review is described.

This page is intentionally not left blank.

4 Methodology

Frankly, I tremble when I think that two of the key prices in any economy, the interest rate and the exchange rate, are decisively influenced by the financial decisions of some of the most credulous and worst-informed people in this world
—Jaime Francisco Javier Ros Bosch.

This chapter describes the methodology followed in this thesis.

Section 4.1 lists the different data types, where they were extracted from, and the results from exploring these datasets.

In section 4.2, the preprocessing, feature selection, feature extraction, and feature construction steps applied to each of the datasets are described.

Section 4.3 contains information regarding all the trained and tested models and their hyperparameters and metrics to tune and evaluate them.

Finally, in section 4.4, the method to test the model is presented, along with the justification for using that method.

This thesis follows a standard ML pipeline to develop the DRL investing system. The pipeline involves the following blocks: data acquisition, feature engineering, model selection, model testing, and deployment. An overview of these blocks can be observed in Figure 4.1.

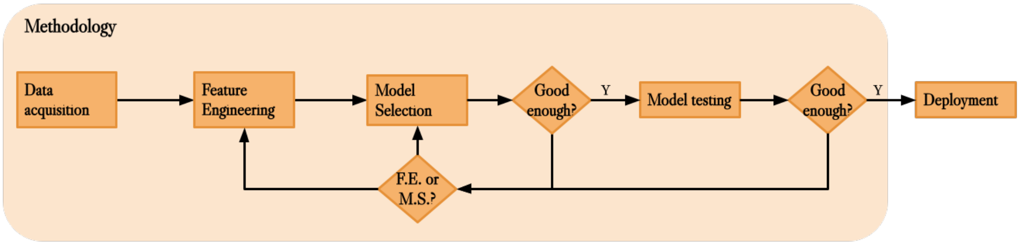


Figure 4.1: Machine Learning Pipeline. Overview of the steps from data acquisition to model deployment.

4.1 Data acquisition

Involving collecting and exploring data, data acquisition is probably the most important step in an ML pipeline. Data that is biased or contains other types of errors is useless— or even dangerous [150]. Just as the saying goes: "garbage in, garbage out." For this reason, collecting data, labeling data, or recording a reward signal correctly needs to be performed cautiously.

In general, stock trading is challenging due to the vast amounts of data, dynamic interactions between economic agents, and the constantly changing methodologies to report the data. For instance, macroeconomic models used by governments are different and are changed from time to time.

4.1.1 Data Collection

There are different types of data used by investors to predict price movements or rationalize their speculative decisions. In this thesis, eight types of data are included to help the model guide its investment decisions. These eight data types include stock market, exchange indexes, technical and fundamental data, currencies, commodities, macroeconomic data, and textual data.

While stock market, technical and fundamental data are included to provide an overview of the U.S. economy at a local level, exchange indexes, currencies, commodities, and macroeconomic data are added to situate the U.S. economy within the international context. As for textual data, it is incorporated to support the other types of data given that financial news includes information about possible future events—e.g., companies acquisitions or mergers pending the approval of some government institution—attempts to explain market movements—regardless of how accurate or truthful these are—or provides opinions about specific topics.

Data covers the time frame from 1987/01/02 to 2013/11/01, except for some cases where it is not available yet—for companies such as Amazon or Google that had not been founded—or currencies that had not been introduced (i.e., Euro).

The fundamental and textual datasets limit the time frame. Fundamental data was only available from 1990 onward (see subsection 4.1.1.5), while textual datasets covered only 1987 to 2013 (see subsection 4.1.1.9). Although the datasets could have been expanded to include more recent data points, the 26-year time frame contained a significant sample of diverse economic events. As more data points would have required additional training time, it was not considered crucial to parse the web to acquire these extra data points.

Figure 4.2 shows the time frame covered by the datasets, and Table 4.1 lists the types of datasets included and the features selected in each of them.

In this thesis, data can suffer from the following type of biases: data-mining bias, sample bias, look-ahead bias, and time-period bias. Thus, special care was taken when handling data to avoid introducing these biases—the interested reader can visit section A.1.1 in Appendix A for a description of them.

Data-mining bias was prevented by using a limited number of runs per experiment, cross-validation for time series, and random initializations. These steps decreased the likelihood that good performance had been the result of chance.

Sampling bias was reduced due to the use of a financial index (i.e., the S&P 500 index) which is created with the specific goal of selecting a sample of companies that are representative of the economy, sector, or industry.

Look-ahead bias was eliminated using cross-validation for time series and careful application of pre-processing steps. These steps prevented the model from knowing future information beforehand.

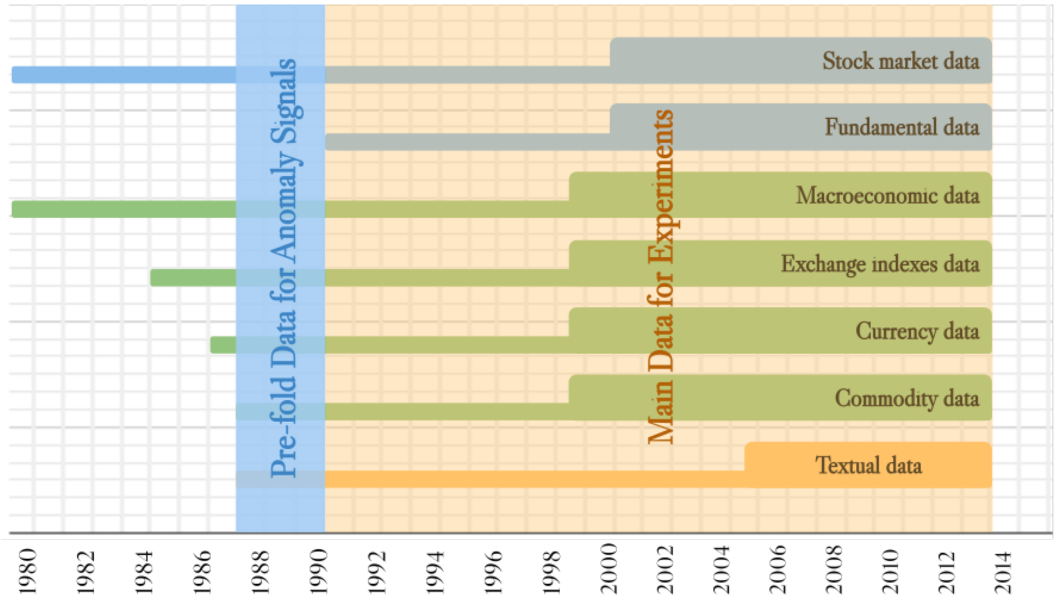


Figure 4.2: Time frame covered by datasets. Figure shows the available data points for the different datasets separated in two blocks: the blue block used as a pre-fold set in the anomaly detector model (see section 4.2.3), and the orange block used in the experiments.

Lastly, time-period bias was avoided via a broad timeframe that included a diverse set of positive and negative economic events.

4.1.1.1 Stock market data

A key data set to study the behavior of markets and teach an agent how to invest is stock market data. However, there are thousands of companies only in the U.S. [151]. For this reason, it would be impossible to include the information of all of those companies. One alternative to solve this problem is to sample companies, but this has the risk of adding companies that are not representative of their industry. A better alternative is to use companies that are part of a financial index such as the S&P 500, which is the alternative used in this thesis.

With 425 stocks selected based on data availability from Yahoo Finance, a data set containing OHLC prices, adjusted-close price, and volume is formed. The list of companies is shown in Table B.1 in Appendix B.

Table 4.1: Data types used in the experiments.

Type of data	Features
Stock market data	Open, close, low, high, and adjusted-close prices, and volume.
Benchmark (S&P 500)	Open, close, low, high, and adjusted-close prices, and volume.
Technical data	MACD, BB, TUO, TDI, Williams %R, ADX, EVWMA, VWMA, CCI, and RSI.
Fundamental data	See Table B.3.
Stock exchange indexes	AS30 (Australia), CAC 40 (France), DAX (Germany), DJIA (US), FTSE MIB (Italy), IBEX (Spain), IBOVESPA (Brazil), IPC (Mexico), KOSPI (Korea), MERV (Argentina), NASDAQ (US), Nikkei (Japan), SENSEX(India), SHCOMP (China), and FTSE 100 (UK)
Commodities	Agricultural raw material index, aluminum, banana, beef, coal, coffee (Arabica and robusta), copper, corn, cotton, gasoline, gold (London), industrial material index, iron, metal index, oil (Brent, Dubai and WTI), rubber, shrimp, soybeans, sugar, sunflower oil, uranium, and wheat.
Currency exchange rates	BRL / USD, CAD / USD, CNY / USD, DKK / USD, HKD / USD, INR / USD, JPY / USD, KRW / USD, MXN / USD, NOK / USD, SEK / USD, TWD / USD, USD / EUR.
Macroeconomic data	See Table 4.3
Textual data	New York Times Annotated Corpus; Bloomberg News dataset; Mergers and acquisitions dataset from Thomson ONE.

4.1.1.2 S&P 500

As mentioned in section 2.1.3.2, a benchmark is used to compare portfolios. Thus, having this information available will be useful. On the one hand, it might help the model discard unprofitable investment strategies sooner as the agent is not limited to compare only among its investment strategies, but can compare with the benchmark. On the other hand, it will help evaluate the agent's performance after the experiments within the market context.

Since S&P 500 stock market data is being used, it seems reasonable to use that capitalization-weighted index as a benchmark as it attempts to mirror the U.S. economy by carefully selecting assets based on their liquidity, sector, industry, and capitalization [17].

Like the stock data, the S&P 500 index data includes OHLC and adjusted-close prices, and volume information.

4.1.1.3 Stock exchange indexes

Given that economies do not exist in a vacuum, it is imperative to consider the effects of economic markets given that assets, markets, economies and regions in other time zones can provide relevant investing information in terms of market sentiments and expectations ahead of market openings in other regions [66, 67, 65].

To capture this information, stock exchange indexes from the top economies—as measured by the International Monetary Fund (IMF)'s 2017 GDP ranking—are used. Depending on their availability, datasets are retrieved from Yahoo Finance which results in 16 datasets from 14 countries, listed in Table 4.1.

4.1.1.4 Technical data

Technical data consists of technical indicators calculated using the Technical Trading Rules (TTR) package in R from stock market data. They are used due to their popularity and apparent efficiency [152, 26, 27].

The indicators used in this thesis are listed in Table 4.1, and further details can be consulted in section B.2.1 in Appendix B.

4.1.1.5 Fundamental data

Fundamental data is extracted from Thomson ONE. It included the income statement, balance sheet, and statement of cash flow data, but due to its high dimensionality, financial metrics and ratios were calculated from them instead [153].

To evaluate most companies, [153] suggests the first five groups, and to evaluate banks, [154] recommends key bank values. These groups are defined as follows:

- **Return and profitability ratios**: These are ratios that serve as fundamental tools in the valuation process. They indicate the quality of a company in terms of profitability within the context of other indicators.
- **Financial stability ratios**: These ratios validate and quantify a company's financial stability, which helps investors select companies with stable capital structures and sufficient cash flow.
- **Working capital management ratios**: They tell investors which companies have sufficient cash flow to meet their short-term operating cost and short-term debt obligations and which companies risk being unable to pay their liabilities.

  The term *working capital management* refers to a business strategy that optimizes inventory, receivables, and cash on hand to increase profitability.
- **Valuation ratios**: These ratios help investors compare share prices of different companies and determine their value.
- **Key bank values**: They are used to assess bank profitability, activities, and growth.

Table 4.2 shows the ratios included in this thesis, and additional information regarding them is presented in section B.3 in Appendix B.

4.1.1.6 Currency data

Additionally to the stock exchange index information, currency exchange rates also convey information regarding market correlations and mood [67].

Currency exchange rates are retrieved from the Federal Reserve Bank of St. Louis (FRED) based on two criteria: a country's rank from a list of nations ordered according

Table 4.2: Fundamental data.

Group	**Ratios**
Return and profitability ratios	Asset turnover, operating cash flow margin, EBIT margin, EBITDA margin, net profit margin, ROA, ROE, ROI, and ROCE.
Financial stability ratios	CAPEX ratio, current assets to total assets ratio, debt to EBITDA, dynamic gearing ratio, equity and long term debt to fixed assets ratio, equity ratio, equity to fixed assets ratio, gearing, goodwill ratio, productive asset investment ratio, and noncurrent assests to total assets ratio.
Working capital management ratios	Days payable outstanding, days sales outstanding, cash ratio, quick ratio, current ratio, inventory intensity, inventory turnover, inventory days, and cash conversion cycle.
Valuation ratios	Market capitalization, price to earnings ratio, price to book ratio, price to cash flow ratio, price to free cash flow ratio, free cash flow yield, price to sales ratio, enterprise value (EV), EV to EBITDA, EV to EBIT, EV to free cash flow, and EV to sales.
Key bank values	Earning assets ratio, loan loss provision, loans to deposits, non-interest income, operating expense ratio, spread, and tangible common equity ratio.

to the IMF's 2017 GDP ranking and availability. This process results in 13 currency pairs.

4.1.1.7 Commodity data

Commodity data is also queried from FRED. Commodity datasets are selected based on data availability and subjective assessment of which is more likely to affect the market based on the literature review.

Commodities related to food were included because their prices might affect inflation and the companies that use it as input for their products, such as Starbucks.

Commodities related to production, i.e., gold, iron, the industrial material index, copper, and rubber were selected because their prices affect, directly, companies using them as input for their products (e.g., Alcoa producing aluminum trim coil) and, indirectly, companies using intermediate goods made with these commodities as input (e.g., Volkswagen using aluminum in its cars).

4.1.1.8 Macroeconomic data

To evaluate the economy's overall health, different institutions, government agencies, and organizations collect information about the economy using surveys of people and companies.

This thesis uses the indexes and reports listed in Table 4.3 and described in section B.4 in Appendix B. They were selected based on data availability and the literature review regarding variables affecting the global economy.

Macroeconomic, fundamental, and commodity data have a variety of sampling frequencies. These values were resampled to a 1-day resolution and, to minimize the sampling frequencies' effects, two steps were taken: 1) collect originally published data and 2) carry forward the last observation.

In the first case, attempts to collect the original data points were made for all the datasets assuming that these values were the ones observed by the market on a given day. The only exception was the GDP report—revised periodically and up to three years after its release—as restated values were used with a negligible effect on the results.

In the second case, it was assumed that, until new official reports or values were published, the most recent value was the value used by the market.

Table 4.3: List of Macroeconomic data, frequency, and publisher.

<table>
<tr><th>Report</th><th>Frequency</th><th>Published by</th></tr>
<tr><td>The Consumer Sentiment Index</td><td>Monthly</td><td>University of Michigan (UM)</td></tr>
<tr><td>Consumer Confidence Index</td><td>Monthly</td><td>OECD databank</td></tr>
<tr><td>Employment Situation Report</td><td rowspan="3">Monthly</td><td rowspan="3">U.S. Bureau of Labor Statistics</td></tr>
<tr><td>Consumer Price Index</td></tr>
<tr><td>Producer Price Index</td></tr>
<tr><td>Median Consumer Price Index</td><td>Monthly</td><td>Cleveland FED</td></tr>
<tr><td>Industrial Production and Capacity Utilization Report</td><td>Monthly</td><td>The Federal Reserve</td></tr>
<tr><td>Advance Report On Durable Goods</td><td rowspan="4">Monthly</td><td rowspan="4">U.S. Census Bureau</td></tr>
<tr><td>New Residential Construction</td></tr>
<tr><td>New Residential Sale</td></tr>
<tr><td>Retail Trade Report</td></tr>
<tr><td>Gross Domestic Product</td><td>Quarterly</td><td rowspan="2">U.S. Bureau of Economic Analysis</td></tr>
<tr><td>Personal Income and Outlays Report</td><td>Monthly</td></tr>
<tr><td>Delinquencies on All Loans and Leases, Commercial and Industrial, All Commercial Banks</td><td>Quarterly</td><td rowspan="3">FRED</td></tr>
<tr><td>Money supply (M3)</td><td>Monthly</td></tr>
<tr><td>Unemployment Insurance Weekly Claims Report</td><td>Weekly</td></tr>
</table>

Continued on next page

Table 4.3: List of Macroeconomic data, frequency, and publisher.

<table>
<tr><th>Report</th><th>Frequency</th><th>Published by</th></tr>
<tr><td>30-Year Fixed Rate Mortgage Average in the United States</td><td></td><td rowspan="4"></td></tr>
<tr><td>Interest rates</td><td rowspan="3">Daily</td></tr>
<tr><td>TED spread</td></tr>
<tr><td>Bank Prime Loan Rate</td></tr>
</table>

4.1.1.9 Textual data

Although numeric information is useful, other data types cannot be directly quantified but are equally valuable. For instance, satellite and CCTV images can provide information about parked vehicles at supermarkets that work as a proxy to estimate quarter revenue. Tweets and Google trends can provide real-time information about events worldwide, and news can present further context and information about an event [147].

In this thesis, news data is the best alternative because obtaining satellite and CCTV images is expensive, and tweets and Google trends do not contain enough details. Conversely, news data is cheaper to get using web scraping techniques, and contains more detail about an event that can help a model make an informed decision [147].

In order to cover the period of study from 1987 to 2013, three datasets are used: New York Times Annotated Corpus (NYTAC), the Bloomberg news, and the mergers and acquisitions datasets.

The NYTAC contains over 1.8 million articles (excluding wire services articles) written by the New York Times between January 1, 1987, and June 19, 2007 [155].

Most articles contain metadata added by the New York Times Newsroom, Indexing Service, and staff in the form of tags or summaries. Tags are normalized and grouped according to five categories—persons, places, organizations, titles and topics—and summaries are written by library scientists. While more than 1.5 million documents are manually tagged with at least one of these tags and over 275,000 algorithmically tagged, only 650,000 articles include a summary [155].

The text is formatted using the News Industry Text Format (NITF), an XML specification that "provides a standardized representation for the content and structure of discrete news articles" [155] to answer who, what, when, where, and why questions.

The Bloomberg dataset contains 450,341 news from Bloomberg spanning from October 20, 2006, to November 20, 2013. Unfortunately, because this dataset was scraped, it does not contain the additional metadata that the NYTAC has [155].

The mergers and acquisitions dataset contains information about company mergers and acquisitions events between January 2, 1987, and November 1, 2013. The dataset was retrieved from Thomson ONE.

It includes 263,216 rows, each of which has the ten following fields:

- Target: company to acquire or to be merged with, or the assets to acquire.
- TargetNation: country in which the company or assets are located.
- DateAnnounced: date of the announcement.
- Acquiror: company proposing the operation.
- AcquirorNation: country in which the acquiror is located.
- TransactionValue: value of the transaction formatted as a percentage, decimal, or Dollar value.
- TargetBusinessDesc: business description of the target company.
- PercSharesAcq: percentage of company shares acquired (when applicable).
- Synopsis: context description of the acquisition or merger.
- DealType: type of deal.

4.1.2 Data exploration

Data exploration is used to identify features that need to be fixed or discarded and features that can be combined or transformed.

Regarding the first group, missing or erroneous values appear when values are not measured, measured but lost, or measured with a damaged instrument—causing an erroneous and unusable value—and might cause problems further in the pipeline as the next blocks often expect complete data [156, 157].

Concerning the second point, adjusting and transforming data are utilized with two goals: to homogenize data variability and simplify the data structure [38, 42].

After data exploration, the following information was gathered:

- All numerical data sources contain missing values.
- The commodities dataset contains invalid values.
- The currency dataset contains missing values for the Euro as it was introduced in 1999.
- The fundamental dataset is sparse and contains a significant number of features.
- News sources are formed by different types of news and might contain high levels of noise in terms of irrelevant news from a financial perspective.
  - The maximum number of characters in an article is 232721.
  - The minimum number of characters in an article is 1.
  - The average number of characters per article is 3260.
- The Bloomberg and Reuters datasets do not contain metadata as the NYTAC does.
- The merger and acquisitions dataset contains a significant number of typos and the target column has noise in the form of amounts, e.g., the area of acquired property.
- Data distribution varies across datasets which might suggest that different normalization techniques are needed.

This information is summarized in Table 4.4. Descriptive statistics are included in Tables 4.5, 4.6, and 4.7 in this section, and in Tables C.1, C.2, C.3, C.4, C.5, and C.6 in appendix C. In addition, a sample of some datasets' histograms is included in Figures 4.3, 4.4, and 4.5 in this section.

Table 4.4: Results of data exploration.

Dataset	**Subset**	**Percentage of NANs**	**Number of features**
Stocks data		22.01%	2550
	Balance sheet	95.88%	
Fundamental data	Cash flow	95.9%	18770

Continued on next page

Table 4.4: Results of data exploration.

Dataset	**Subset**	**Percentage of NANs**	**Number of features**
	Income statement	95.53%	
Technical data		22.22%	9775
Benchmark		0.00%	6
Exchange indexes		30.67%	90
Macroeconomic data		95.03%	432
Currency data		11.9%	13
Commodities		88.64%	25

Table 4.5: Descriptive statistics of all the stock market prices.

Fields	**Close**	**High**	**Low**	**Open**	**Volume**
count	2.42e+06	2.42e+06	2.42e+06	2.42e+06	2.42e+06
mean	2.96e+01	3.00e+01	2.92e+01	2.96e+01	4.52e+06
std	5.37e+01	5.43e+01	5.31e+01	5.37e+01	1.40e+07
min	6.50e-04	1.30e-03	6.50e-04	6.50e-04	0.00e+00
25%	9.22e+00	9.36e+00	9.08e+00	9.22e+00	4.91e+05
50%	2.09e+01	2.11e+01	2.06e+01	2.09e+01	1.41e+06
75%	3.75e+01	3.80e+01	3.70e+01	3.75e+01	3.77e+06
max	2.074e+03	2.08e+03	2.04e+03	2.07e+03	1.86e+09

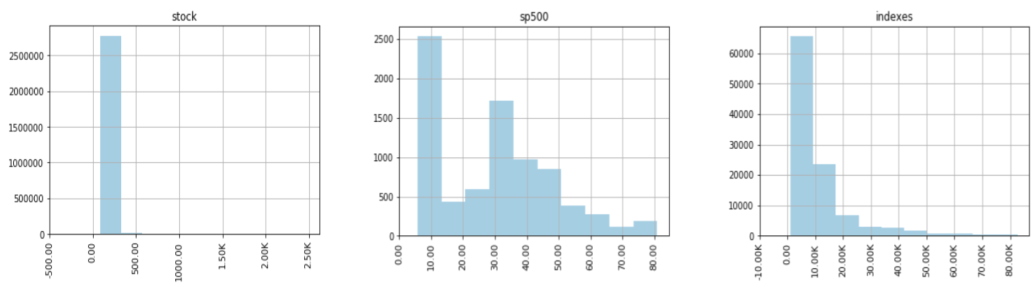


Figure 4.3: Stock, benchmark and indexes histograms.

Table 4.6: Descriptive statistics of a random features sample of technical data.

Fields	ADX	atr	di	trueHigh	up
count	7261.00	7274.00	7279.00	1014.00	7269.00
mean	26.58	0.75	0.79	40.056	14.56
std	12.06	0.38	10.88	10.89	11.55
min	6.09	0.21	-60.40	22.18	1.95
25%	18.09	0.44	-4.51	29.84	7.46
50%	23.64	0.68	1.38	40.27	11.27
75%	31.85	0.96	6.91	50.42	16.04
max	83.85	2.29	42.69	59.87	60.44

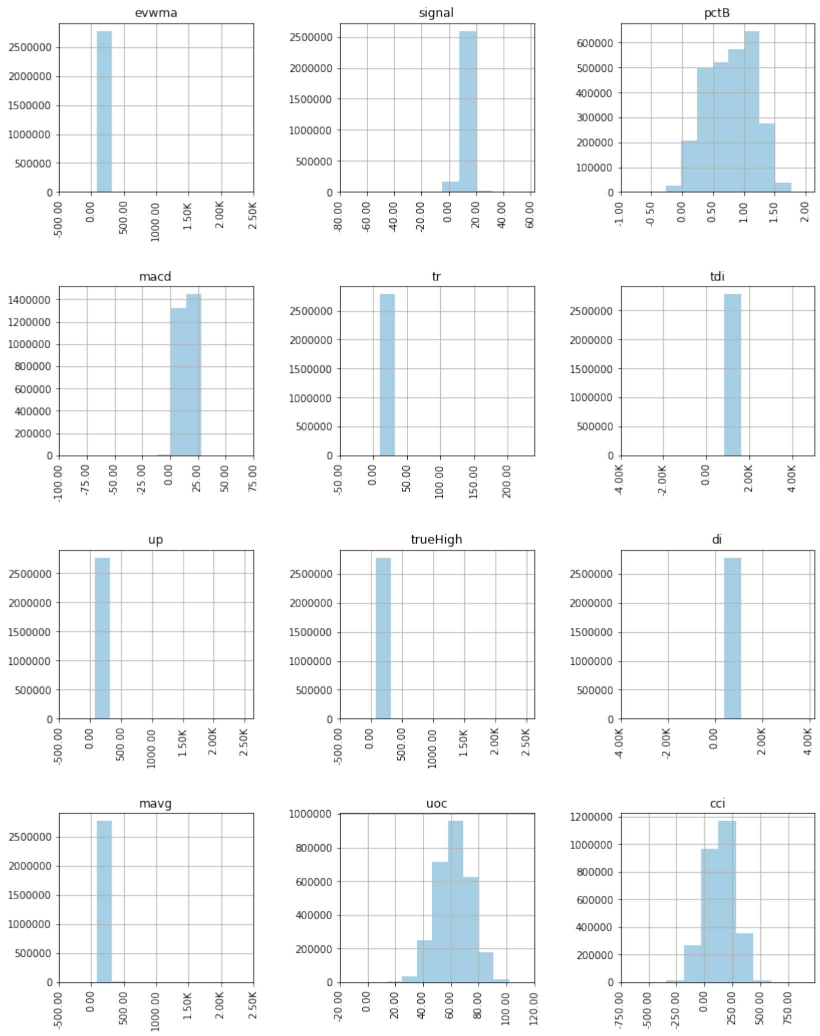


Figure 4.4: Features sample histogram of technical data.

Table 4.7: Descriptive statistics of a random features sample of currency data.

Fields	DEXCHUS	DEXHKUS	DEXJPUS	DEXKOUS	DEXTAUS
count	6980.00	7040.00	7041.00	6997.00	6814.00
mean	6.83	7.77	115.44	996.47	30.60
std	1.68	0.027	20.77	211.87	3.22
min	3.19	7.71	75.72	667.20	24.51
25%	5.74	7.75	102.94	801.30	27.51
50%	7.42	7.77	115.14	985.00	30.83
75%	8.28	7.80	127.05	1163.00	32.99
max	8.74	7.83	202.70	1960.00	39.87

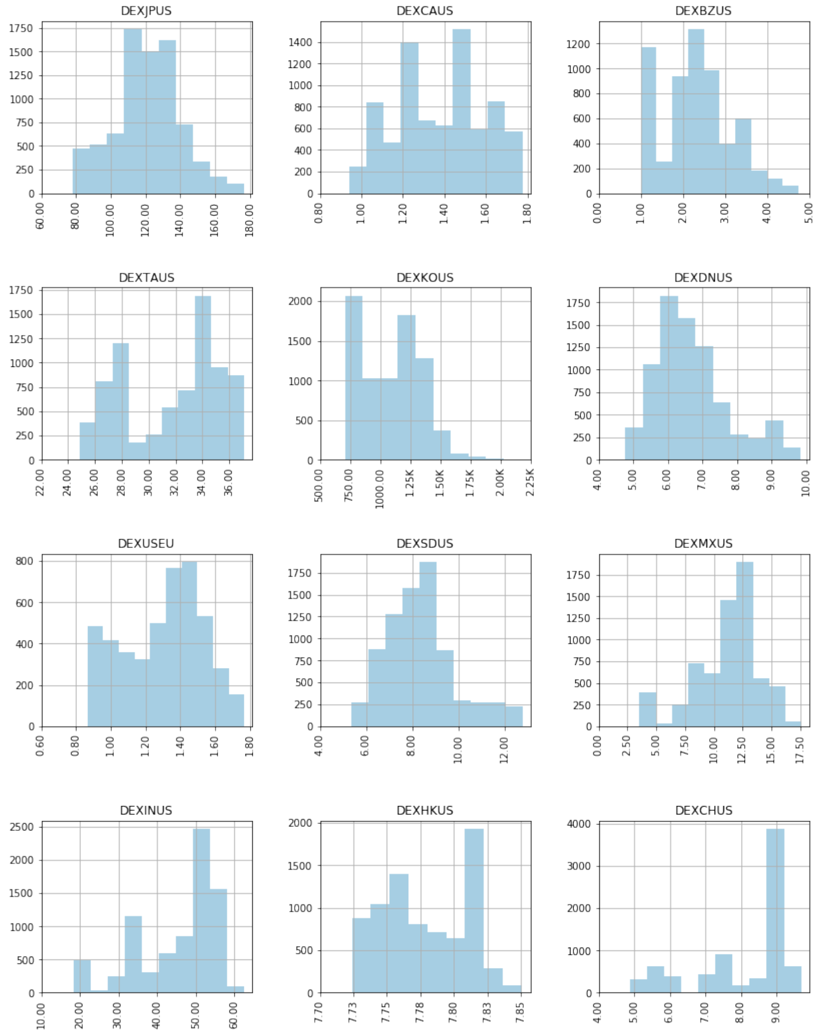


Figure 4.5: Currency data histogram.

4.2 Feature engineering

Feature engineering refers to the process of extracting, transforming, or creating features that better represent the underlying problem and are suitable for the ML models to train the algorithms faster, reduce model complexity and overfitting, and increase interpretability and model accuracy [42, 158, 159].

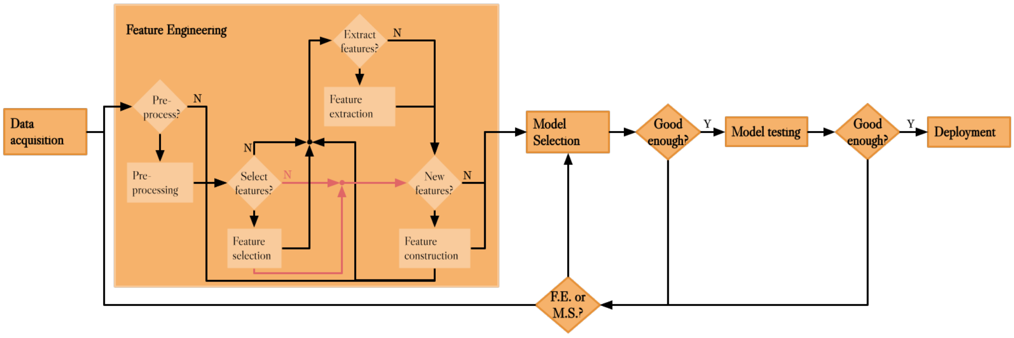


Figure 4.6: Feature engineering steps. The diagram contains the optional steps during feature engineering.

Feature engineering consists of four steps: preprocessing, feature selection, feature extraction, and feature construction [159, 160]. These blocks are displayed in Figure 4.6. While preprocessing is always applied at the beginning of the feature engineering process, feature selection, feature extraction, and feature creation can be applied in any order or even recursively as implied by Motoda and Liu in [161].

4.2.1 Preprocessing

Consisting of data imputation and data transformation methods, preprocessing is a step that is used to fix data so that ML models can learn from data correctly. Although this step might help or hinder convergence, it might not even be necessary depending on the results of the data exploration process and the type of ML model to use [38, 42].

4.2.1.1 Data imputation

The process of replacing missing values with reasonable values is called imputation. The methods used for this process depend on the assumption of the missing-data mechanism. These mechanisms are categorized according to three groups: Missing Completely At

Random (MCAR), Missing At Random (MAR), and Missing Not At Random (MNAR) [162].

The first one, MCAR, states that data missing from a dataset does not depend on the values of the missing or observed data. MAR, a more flexible version of MCAR, states that missing data may depend on the observed components but is conditionally independent of the missing components given the observed elements. Lastly, when the MAR conditions are violated, the mechanism depends on the missing data, and the imputation mechanism becomes MNAR [162, 163].

In this thesis, most financial datasets fall under MAR, while textual datasets fall under MCAR. Financial MAR points are caused by holidays that vary from country to country. According to [163], missingness due to holidays can be considered as missing at random because the "probability of a specific day being designated as a holiday in any country is unrelated to the stock price on that particular day if the price had been observed." At the same time, financial MNAR points are caused by fixed events such as weekends and some calendar events (refer to section 2.2.1.2 for further details).

[162] also states that the most realistic and frequently used assumption on the missing-data mechanism is MAR, given that, empirically, models that use this assumption have been found to have higher accuracy than standard nonignorable models (i.e., models in which missing-data mechanisms are not ignored; instead, the missing data process is explicitly modeled).

For time-independent data, the list-wise deletion method is used to impute data. This method is the simplest and most direct method to handle missing data. It consists of removing the data that has missing values with the assumption of MCAR. However, if this assumption is violated, this method can result in serious bias in analytic results [162].

For time-dependent data, popular methods to handle missing data include Last Observation Carried Forward (LOCF), Next Observation Carried Backward (NOCB), linear interpolation, and spline interpolation—the interested reader can find more information in section A.2.1.1.1 in A. In this thesis, the LOCF imputation method is used to avoid introducing look-ahead bias [162, 164].

The datasets described in section 4.1 contain missing values that were imputed based on the data type. The process used in each of these data types is described next.

Numerical data

Most numerical datasets were imputed using the LOCF method and a constant value (zero), but the imputation for some datasets, based on their type, exploited their structure. Both methods were used to avoid introducing look-ahead bias into the data and giving future information to the model beforehand.

The imputation process is described next:

- For datasets with OHLC data—benchmark, stock, and stock exchange index data—missing values were imputed by carrying the closing price forward.
- For fundamental data, missing features were attached as zero vectors to standardize the dimensions of the financial and regular companies' datasets. As mentioned in subsection 4.1.1.5, financial institutions are evaluated using a different subset of values and ratios. In addition, missing values were imputed with the LOCF method.
- For the rest of the datasets—commodity, technical, macroeconomic, and currency data—missing values were only imputed using the LOCF method.
- In all cases, data points happening before the first value in the series were filled with zero—which was the scenario for companies created between 1987 and 2013, or currencies introduced after 1987, such as the Euro.

Mergers and acquisitions dataset

Preprocessing of the mergers and acquisitions dataset consisted of the following steps:

1. Keep five columns: DateAnnounced, Target, TargetNation, Acquiror, and Synopsis.
2. Delete rows from companies not included in the S&P500 index.
3. Normalize TargetNation column using sine transformation so that the range of values falls within the range $[-1, 1]$.

This process results in a dataset with 10,170 rows.

4.2.1.2 Data transformation

The consensus about the appropriate use of transformations is that, when fitting any model, the *right* transformation often achieves a better fit and forecast [38]. Given that finding the *right* transformation requires the *right* statistical model and viceversa, one approach suggested in [38] is to proceed iteratively using a transformation, fitting a model, then checking the transformation, and so on.

Numerical features

For numerical features, the techniques tested in this thesis to remove undesired effects—e.g., outliers' influence—include differencing, max-min normalization, standardization, feature clipping, robust scaling, quantile transformation, and power transformations.

Differencing

This technique involves taking the difference between consecutive observations. It can help stabilize the mean of a time series by "removing changes in the level of a time series, and therefore eliminating (or reducing) trend and seasonality" [39].

When differenced data is not stationary, it can be differenced more than one time until it becomes stationary. However, n-order differencing is rarely necessary [39].

To perform differencing, the following calculation is used:

$$x_t^{diff}(d) = x_t - x_{t-d}$$

where x_t is the input value at time t, x_{t-d} is the input value at time $t-d$, and d is an integer representing the number of periods.

Differencing is used in the log-transformation method described below.

Max-min Normalization

This normalization adjusts the input values to the range $[0, 1]$ [165]. According to [166], this method is a good alternative when the upper and lower bound are known, data contains few or no outliers, and it is approximately uniformly distributed across that range.

The formula to perform this normalization is:

$$x_{max\text{-}min\ normalization} = \frac{x - min}{max - min}$$

where min, max and x are the minimum, maximum, and input value, respectively.

A variation of this technique that used a expanding window was tested to normalize different data types, but ultimately discarded because it was not effective.

Standardization

Standardization centers the data—$\mu = 0$—and standardizes the variance for each feature—$\sigma^2 = 1$—by subtracting the mean from the input feature and dividing by its standard deviation. It helps NN models move the data to a region near the origin which is where activation functions work better and training becomes more stable [165, 166].

Standardization is calculated as follows:

$$x_{standardization} = \frac{x - \mu}{\sigma}$$

where μ refers to the mean, σ is the standard deviation, and x is the input value.

Similarly, an expanding-window standardization was used to normalize data types with large values, but log-normalization was preferred with stock prices and similar data types.

Feature clipping

This technique is used when data contains extreme outliers. It consists of setting to a fixed value those feature points above (or below) a certain number. It can also be combined with the previous method, standardization, to remove values above or below three standard deviations [166].

Feature clipping is performed as follows:

$$x_{clipped} = \begin{cases} x, & \text{if } x \leq z \\ z, & \text{otherwise} \end{cases} \tag{4.1}$$

where z is the clipping value and x refers input value.

This method is applied by the Adam optimizer used to train diverse ML models.

Robust scaling

Robust scaling adjusts the input features using the median and interquartile range (IQR)—the values between 25% and 75% of the ordered data. It centers the values using the median and then scales them using the IQR [165, 167]. Unlike standardization, where outliers influence the mean and variance, robust scaling eliminates the effect of outliers [167].

The formula to calculate robust scaling is:

$$x_{median} = x - median$$
$$x_{robust} = \frac{x_{median} - Q_1(x_{median})}{Q_3(x_{median}) - Q_1(x_{median})}$$

where $median$ corresponds to the median value, $Q_1(x_{median})$ and $Q_3(x_{median})$ are first and third quartile value of the transformed data, respectively.

Robust scaling was tested in data normalization, but ultimately discarded in favor of log-normalization.

Quantile transformation

Quantile transformation is another robust non-linear transformation that maps input values to a distribution. This transformation spreads out the most frequent values and reduces the effect of outliers [168].

It estimates the features' cumulative distribution function and uses it to map the values to a uniform or normal distribution by applying a quantile function. Outlier values are clipped to the lower or upper bound of the output distribution [168]. Although clipping the values alters linear correlations, it allows comparing input variables with different scales [168].

Quantile transformation can be calculated as follows:

$$x_{cdf} = cdf(x)$$
$$x_{quantile} = QF(x_{cdf})$$

where cdf corresponds to the cumulative distribution function and QF is the quantile function.

Power transformations

They refer to a family of parametric, monotonic transformations that adjusts data so that it is as close as possible to a normal distribution to ameliorate heteroscedasticity—non-constant variance—and other problems [165].

Two members of the power transformation family are the *Box-Cox transformation* and the *Yeo-Johnson transformation*. Although the log transformation is often the preferred transformation for economic and financial time series, a more general transformation can sometimes be more suitable. However, one important restriction is that it can only be applied to data spanning a wide range, that is $\frac{\max(x_t)}{\min(x_t)} > 3$ [38].

While the Box-Cox transformation is very sensitive to outliers and can only be used with positive values, the Yeo-Johnson transformation can be used on variables with any value [160]. The interested reader can visit Appendix A, section A.2.1.1.2.1 for the Box-Cox and the Yeo-Johnson transformation equations.

Log transformation

As mentioned above, the log transformation is a particular case of the Box-Cox family of transformations. It is used when the main concern is relative changes. It makes the variability of time series showing more or less stationary percentage growth over time more stable [38]. A log-transform value is calculated as follows:

$$x_{log\text{-}transform} = log(x)$$

It is often an appropriate transformation for economic and financial time series as it can be used to make highly skewed distribution less skewed [169, 38, 170, 166].

Based on this type of transformation, an alternative when using stock prices is **log returns**. This transformation has the following form:

$$x_{log\text{-}returns} = log(1 + r_i)$$
$$r_i = \frac{p_i - p_j}{p_j}$$

where p_i and p_j are the prices at time i and j, respectively.

It has three benefits [171, 38]:

- Log-normality: if it is assumed that prices are log-normal, the transformation makes data normally distributed.
- Time-additivity: the sum of normally-distributed variables is normal when all variables are uncorrelated and provides numerical stability as the addition of small numbers is numerically safe, while multiplication can lead to arithmetic underflow.
- Approximately linear: for small values of r ($r \ll 1$), the logarithmic function is approximately a linear, $log(1 + r) \approx r$.

Categorical features

Categorical features are those features that take their values from a discrete set of categories or labels. When the elements of this set have an intrinsic order, they are named **ordinal categorical features**—Moody's and S&P 500's credit rating scales are examples of this type of categorical features. When that information is not present, they are called **nominal categorical variables**—the set of labels used to indicate companies' sectors is an example of this type of feature [160].

While ordinal categorical features can be transformed into numeric values that preserve the order by encoding that information directly, nominal categorical features can be converted to numeric values using one-hot encoding, label encoding, binary encoding, etc. [160, 165].

In this thesis, features with a limited number of values such as financial sectors, company names, and industries were encoded using a cosine function to obtain categorical features, prevent sparsity, and place values within NN's activation region.

Textual features

In the case of text, Jurafsky [172] suggests applying three preprocessing steps: tokenizing words, normalizing word formats, and segmenting sentences.

Tokenization is the segmentation of text into tokens which might refer to words, morphemes, or subwords. Normalization is a process to reduce the dimensionality of the vocabulary by applying standardization, lemmatization, or stemming. Segmenting sentences refers to the task of separating strings into sentences via rules or ML techniques [172, 173]. Additional information is provided in Appendix A, sections A.2.1.1.3.1, A.2.1.1.3.2, and A.2.1.1.3.3.

In this thesis, different preprocessing steps were applied to textual features based on the tasks at hand.

- Word normalization and sentence segmentation were used to fix the merger and acquisition dataset.
  - Several RegEx filters were applied to the sentences to reduce word variations (e.g., Corp, Corp., Corporation, Co, Co., etc.).
  - A list of incorrect words was created to correct typos.
  - A dictionary with each of the company's name variations was created to normalize their names using a RegEx filter.
- Tokens were extracted from news articles using a BERT tokenizer to extend the Bloomberg dataset with taxonomy tags.

4.2.2 Feature selection

Feature selection refers to choosing one or more subsets from the preprocessed dataset and remove *irrelevant* features that do not help the ML model solve a particular task.

The methods to perform feature selection are divided into filter methods, wrapper methods, and embedded methods. However, in this thesis, feature selection methods are not applied to the datasets; instead, feature selection is guided by the literature review. In other words, it is assumed that the indicators listed in the literature review that led to promising results were correctly identified.

The following list contains the datasets used in this thesis and the research that used these indicators:

- **Stock market** data is used in most papers in this area.
- The **technical indicators** listed in table 4.1 are proposed in the following papers: [82, 80, 174, 175, 176, 177, 27, 26, 83, 84, 178]
- The use of **fundamental indicators** to valuate regular companies is proposed in [82, 80, 179, 174, 29, 175], but another set of fundamental indicators is listed in [154] for financial institutions.
- Although only a few papers, [68, 66], suggest a direct relation between different countries—via **stock exchange indexes**—the literature implies a correlation between different countries, sectors, and industries. For this reason, stock exchange indexes are also included.
- Research and books, such as [180, 100, 181, 10], propose that **commodities** impact sectors, industries, and economies. For this reason, commodities listed in these documents and other commodities that were thought to correlate with the price of companies, such as coffee price and Starbucks' share price, are included in table 4.1.
- In the case of **currency exchange rates**, researchers such as [182, 67, 180] indicate that currencies offer a premium. It is hypothesized that data from more influential economies with more stable currencies might contain helpful information.
- Books and papers such as [68, 67, 183, 181, 184] list different **macroeconomic indicators** that impact the economy.
- **Factors** such as momentum, profitability, quality, size, and value, were found to generate a premium— [100, 96, 93, 37, 91, 185, 35].
- **Textual data** in the form of news is suggested (and used) in the following papers:

[179, 177, 178, 181, 128, 121, 186].

In the case of the NYTAC, only the date, taxonomy, lead, and body of the news were preserved because other metadata had missing values and was not as relevant. For the Bloomberg dataset, the date, lead, and body of the news articles were kept. As for the merger and acquisition dataset, the TransactionValue, TargetBusinessDesc, Target, PercShareAcq, and DealType features were removed as these features did not have relevant information or were noisy with a significant number of unique values containing measures, such as the size of a property or the number of assets.

For the rest of the numerical data sources, section 4.1 contains the datasets and features selected, and chapter 3 lists the research papers suggesting those datasets and features.

4.2.3 Feature extraction

Feature extraction involves identifying salient and stable features—in the original dataset or the output from the previous step—that contain the central properties of a dataset and representing them in a low-dimensional space to facilitate learning [42, 187].

Word embeddings

Feature extraction was applied to textual datasets in the form of word embeddings and news embeddings.

Word embeddings were created using a BERT model and used for the taxonomy classification model and the experiment with the merger and acquisition dataset.

Similarly, news embeddings were created using SBERT [188] and applied to the financial dataset to summarize news articles for the dataset types experiment (see chapter 5 for more details on experiments).

Dimensionality reduction

The goal of an autoencoder in this part is to perform dimensionality reduction by decreasing the reconstruction error between the input and output while learning the essential elements of the dataset.

In this thesis, the expectation is that high-dimensional data gets transformed into low-dimensional data containing a *summary* of the local and global economic context by applying dimensionality reduction. The critical point of the dimensionality reduction model is that features are not simply discarded; instead, the model combines them while extracting the essential parts—in other words, key information is not lost; noise and irrelevant information are removed.

In this work, an autoencoder was trained to reduce the dimensionality of most feature types. The exception was the fundamental dataset, given that its low number of data points and high dimensionality caused a significant reconstruction error. Once the dimensionality reduction model finished training, its encoder was used to reduce the dimensionality of each dataset in half (based on the hyperparameters tuning experiment's results described in section 4.3.2).

Anomaly signals

Anomaly signals were created using an anomaly detector model. These features were precalculated to accelerate training and were only available for stock, technical, benchmark, commodity, currency data, and stock exchange indexes because they had enough data points available, and their values were suitable for anomaly detection. This was not true for ranked data such as factors and ranking features.

Datasets were split into six folds to precalculate the anomaly signals: five folds covering the interval from January 1, 1990, to October 1, 2013, and another *pre-fold* from January 1, 1987, to December 31, 1989. The reason for a pre-fold is that an anomaly detector system compares what is normal (i.e., the training fold) to what is abnormal (i.e., the test fold). However, it would have not been possible to calculate the training anomaly signals for the first fold without the pre-fold which means the RL agent wouldn't have had this first-fold data available during training in other experiments.

4.2.4 Feature construction

Feature construction is a process in which new features are added to the dataset. These new features are expected to contain new information and create new patterns to be exploited by ML models to increase performance [159]. Two possible approaches were followed to create new features:

4.2.4.1 New features from existing features

The first approach creates numerical or textual features based on one or more attributes in the original dataset. The main idea is that these additional features will introduce new information to the dataset, which will help ML algorithms to perform better on a given task [159].

In this thesis, time-dependent features and textual features were created from existing ones.

Time-dependent features

Time-dependent features included calendar features, cumulative and ranking features, factor features, and anomaly signals. These features are detailed in the following subsections.

Calendar features

Calendar features were created from date information and added to the dataset. These features are listed next.

- **Timestamp features**: Companies in different sectors are affected by events occurring during the year on specific dates. For instance, retail companies experience an increase in their customer base during Christmas, or movie studios release their blockbuster movies during the summer. As mentioned in [189, 190, 191], evidence suggests the existence of significant calendar effects with positive returns.

  For this reason, features derived from date variables might be beneficial to the RL agent. Although these features can be represented as numerical or categorical types, these representations remove the features' cyclical nature [192]. For instance, the day difference between December 31 and January 1 using a numerical type is equal to 364 (or 365 in a leap year) days instead of the real distance of one day.

  To represent dates correctly and to prevent losing cyclical information, one approach is to encode date features using a cyclical function such as the cosine function. In this thesis, dates are represented using this function to create four features: month, day of the year, day of the month, and day of the week.

The key advantages of using the cosine (or sine) function to encode date information are: 1) the cyclical nature of dates is preserved so that the model can take advantage of, for instance, specific events occurring on a given day of the week (the Weekend Effect [193]), day of the year (the day after Christmas [194]), day of the month (double witching and quadruple witching), or month (the September Effect [195]); 2) features are compact, which prevents sparsity (that other methods such as one-hot encoding does not); and 3) features' values are situated within the NN's activation region (which helps the model to converge faster).

The four date features are defined as follows:

$$feature_{month} = \cos\left(\frac{2 \cdot \pi \cdot month}{12}\right) \tag{4.2}$$

$$feature_{day\ of\ year} = \begin{cases} \cos\left(\frac{2 \cdot \pi \cdot day\ of\ year}{366}\right), & \text{if leap year} \\ \cos\left(\frac{2 \cdot \pi \cdot day\ of\ year}{365}\right), & \text{otherwise} \end{cases} \tag{4.3}$$

$$feature_{day\ of\ month} = \cos\left(\frac{2 \cdot \pi \cdot day\ of\ month}{days\ in\ month}\right) \tag{4.4}$$

$$feature_{day\ of\ week} = \cos\left(\frac{2 \cdot \pi \cdot day\ of\ week}{7}\right) \tag{4.5}$$

The result of applying these functions to create temporal features can be observed in figures 4.7 and 4.8. These figures show that the cyclical nature of the temporal features from a weekly, monthly, and yearly perspective is preserved.

- **Witching features**: There are specific dates of interest for the markets where different derivative contracts expire—stock options, index options, stock index futures, or single stock futures. These contracts need to be *renewed* when economic agents only want to remain exposed to the risk of these derivative instruments. Otherwise, they may need to exchange the *underlying security*—a security on which the derivative contract's value is based—depending on whether the instrument is optional or want to let the contract expire [196, 197, 198].

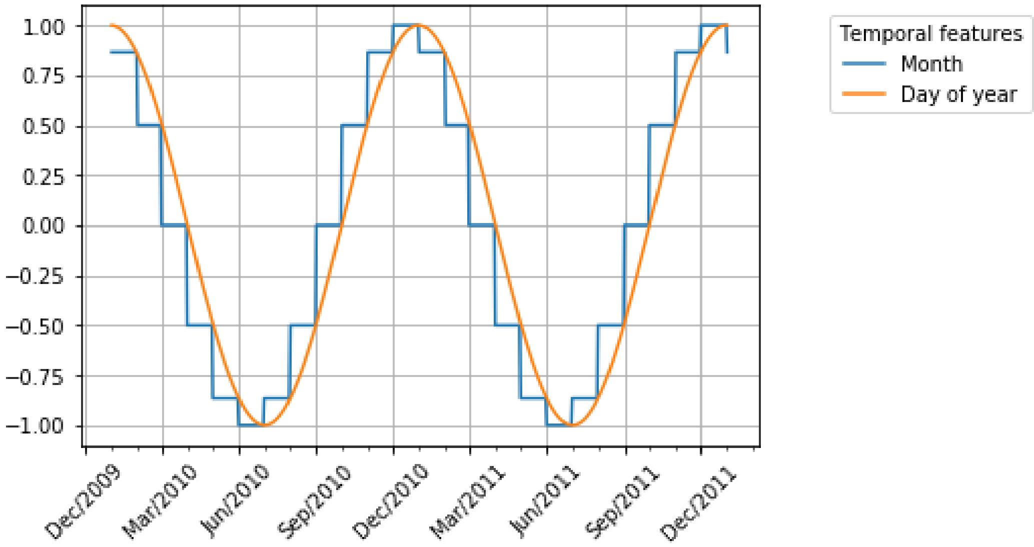


Figure 4.7: Temporal features using a cosine function: Month and day of the year.

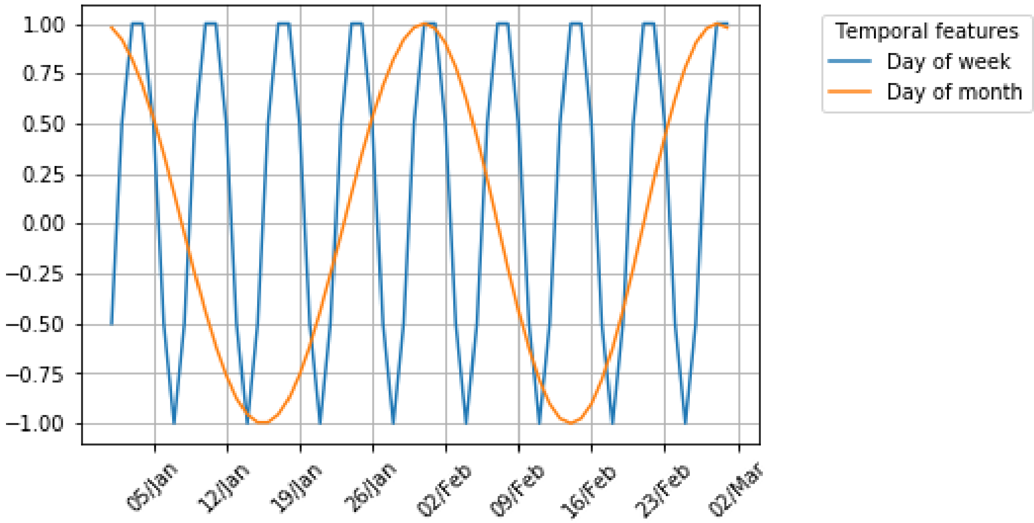


Figure 4.8: Temporal features using a cosine function: Day of the week and the month.

Derivative instruments are used to speculate and manage risk [196]. For instance, an airline company concerned about fuel prices increasing significantly in the future can acquire a fuel call option which is a derivative with fuel as the *underlying security* that gives the airline the right to buy fuel at a given price in the future, while it gives the vendor the obligation to sell it. The company can use this call option to mitigate this risk by paying an upfront premium cost. If the price has increased on the expiration date, the airline only pays the price agreed on the option contract. Conversely, if the price has decreased, the airline buys the fuel at a lower price directly from the market without exercising the purchase option (and letting the contract expire).

The third Friday of each month except for March, June, September, and December, when two different derivative contracts expire, is a day named **double witching**. Similarly, the third Friday of March, June, September, and December, when four different derivative contracts expire, is a day called **quadruple witching**. What is special about these days is the significant increase in trading volume and volatility—in particular, during the final hour of trading—as investors try to close or change their contracts [197, 198].

For this reason, adding double and quadruple witching as categorical features might help an ML model to take advantage of these agents' urgency to drop these contracts (and their obligations).

Cumulative and ranking features

Cumulative features are based on stock returns aggregated over three windows, i.e., 30, 60, and 90 days. Similarly, ranking features are created from cumulative features, but instead of representing these features as ordinal numbers, they are encoded as decimal numbers in the interval $[-1, 1]$, with -1 given to the company with the lowest cumulative return and $+1$ assigned to the company with the highest cumulative return.

This encoding is used as a pseudo-normalization to avoid big numbers (for those scenarios with thousands of companies to pick from).

Factor features

Factor features are calculated using the momentum, profitability, quality, size, and value factors defined in chapter 2, section 2.1.4.3, and in Appendix A, section A.1.4.1.

These factors are constructed using fundamental indicators and stock prices. Then, similar to the ranking features, these factors are encoded as decimal numbers in the same interval $[-1, 1]$, and aggregated to avoid having sparse features and increasing the number of features significantly.

Textual features

Although [199, 159] consider word embeddings as feature creation, it seems more suitable to label it as feature extraction given that in the case of feature creation, new features are combined with the original dataset, while in feature extraction, new features are used instead of the original dataset. For this reason, no textual features were created from existing ones.

4.2.4.2 New features created by ML model

The second approach creates new attributes by teaching an ML architecture to perform a given task such as classification, regression, clustering, among others. The output from these tasks can be added to the input dataset as new attributes that might help increase performance regardless of data type [159].

This approach was taken with the Bloomberg dataset, for which a semi-supervised model based on BERT was trained using the NYTAC taxonomy labels and then used to predict the topic tags for the Bloomberg dataset. However, only those tags that the model was confident about ($p > 0.75$) and the corresponding news were added to the dataset. Lastly, news articles from the Bloomberg and NYTAC datasets with a business, economic, or financial tag were kept while the rest were discarded. The resulting dataset contained the news set from January 1, 1987, to November 20, 2013, tagged by topic.

In addition, a question-answering model using a variant of the BERT model was used to add to the financial dataset the name of the company, the sector, and the main actor mentioned in a news article. Although it worked for some news, it introduced more noise than good results, which is why these features were discarded.

4.3 Model selection

This step refers to fitting and estimating the performance of multiple models to select the best one. Model selection involves the following steps—shown in Figure 4.9: model training, hyperparameter tuning, and model evaluation.

These blocks are iterative, and as soon as the model's performance is acceptable, the iteration ends, and the model moves to the next block. However, if it is never acceptable, it is possible to move back to previous steps in the pipeline [44, 54, 47].

In this thesis, model selection is performed via resampling methods which involves model training, hyperparameter tuning, and model evaluation.

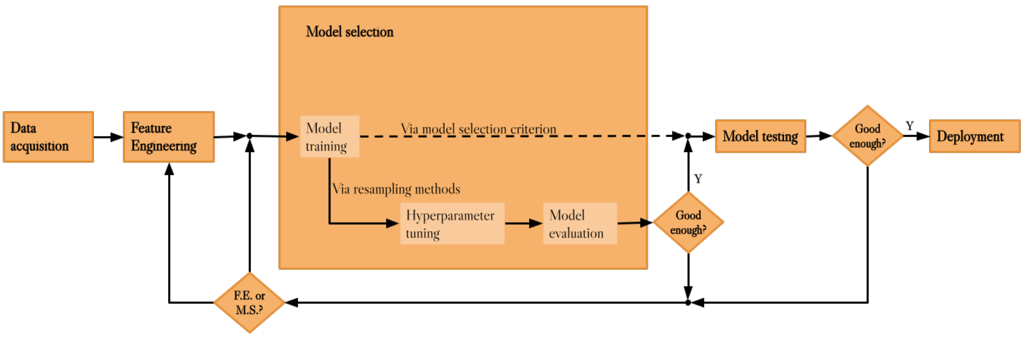


Figure 4.9: Model selection block. The two paths towards model testing: via resampling methods or model selection criterion.

- **Model training**: In this first block, models' parameters are adjusted to fit the training data and optimize a cost function.
- **Hyperparameter tuning**: It is a meta-optimization task that involves searching for the best hyperparameters by evaluating them on a validation set. Unlike model parameters learned during training, hyperparameters are a set of dataset-dependent values specified outside of the training procedure that help prevent overfitting. There are different hyperparameters; some control the model's structure—model hyperparameters—while others govern the cost function—cost hyperparameters [200, 201].

Three classes of methods can be used to find the best hyperparameters: grid search, random search, and smart hyperparameter tuning [200, 47].

- **Model evaluation**: In this last block, metrics are used to assess models' performances.

4.3.1 ML Model Selection

First, it is necessary to find the best ML classifier to identify trading opportunities. In order to do so, ML model selection blocks are:

- **Model training**: Four ML architectures are used: SVM, RF, NN, and LSTM. These models are trained using cross-validation for time series.
- **Hyperparameters tuning**: Despite being computationally expensive, this work uses grid search to study the four architectures' hyperparameters.
  - For the SVM architecture, the hyperparameters include C and *kernels*.
  - For the RF model, the hyperparameters are the number of estimators, the maximum tree depth, and the minimum number of samples to split.
  - For the NN and LSTM models, the hyperparameters comprise the number of neurons per layer, the activation functions per layer, and the number of epochs.
- **Model evaluation**: Accuracy and ROI metrics are used to evaluate the models.

4.3.2 Autoencoder Model Selection

This thesis uses autoencoders for Anomaly Detection and Dimensionality Reduction. The model selection blocks for this model are defined as follows:

- **Model training**: Autoencoders for both tasks are trained similarly. An input data source is received, and the main goal is to reduce the reconstruction error between this input and the autoencoder's output.

  The difference is that for anomaly detection, the Mean Squared Error (MSE) values are used to obtain an indicator of what is *normal* in the dataset so that *abnormal* values in the test set raise the alarm.

For dimensionality reduction, the encoder's output, which contains the low-dimensional representation, is utilized.

- **Hyperparameters tuning for dimensionality reduction**: The following hyperparameters were varied in this scenario:
  - Number of encoding features: The number of neurons in the encoder's output that controlled the reduction level was varied from 100% to 16%.
  - Activation functions: Autoencoders were built using one type of activation function for all the layers. These activation functions included the sigmoid, tanh, ReLU, and swish functions.
  - Input data: The models receive one of two options. Either data with similar features, e.g., OHLC prices, or data with drastically varying features, e.g., OHLC prices and Volume.
- **Hyperparameters tuning for anomaly detection**: The only hyperparameter varied for anomaly detection was the function to calculate normality. The other hyperparameters were taken from the best-performing dimensionality reduction model.

  These functions included the mean, minimum, maximum, median, and bands placed one or two standard deviations above the mean.
- **Model evaluation**: To evaluate dimensionality reduction, the reconstruction error as measured by the MSE was used. As for the anomaly detector, features were plotted against stock prices and subjectively evaluated.

4.3.3 RL Model Selection

A trading system is built based on the results from the previous sections. The ML model selection section indicated the most suitable model to forecast the market direction. In addition, the ablation experiment (described in chapter 5) provided information regarding which data sources were more influential when trading.

Based on that knowledge, the trading system selection follows these steps:

- **Model training**: In the case of the RL model, training begins when the stock market environment randomly selects a starting date and returns an initial state to the DRL agent. This high-dimensional state contains variable information that can include any of the datasets explored throughout this thesis. This information is supposed to indicate the state of the global economy.

With this information, the agent uses a NN model to handle high-dimensional data and determines the percentage of cash allocated to each company. Then, the agent takes an action and adjusts the cash allocation. This cash allocation is rounded down to buy an integer number of shares. At the same time, the fees and total reward are calculated and subtracted from the agent's cash.

As the simulation moves one step ahead, the agent takes the action (i.e., the adjusted cash allocation) and receives the reward signal that guides future actions until the end of the episode. At that point, the agent reviews the actions and rewards during the episode and tries to learn which ones lead to the best performance based on the loss function. In essence, the model is based on a Deep Q-Learning algorithm.

The proposed model considers transaction fees (i.e., Saxo capital markets' fee of 0.1% per transaction) but ignores latency between the agent and the stock market—although it could be added to the simulator and provided as a feature to the agent. It also ignores other trading costs that depend on market dynamics, such as trading impact costs, because a mathematical formulation such as those used in [135] is a subjective simplification that might be unrealistic, as suggested in chapter 3's summary.

The NN model is optimized as follows. Every time the agent has to take an action, the algorithm enters a cycle of optimization for the current state and suggests an action. Then, the algorithm evaluates if the action is valid, that is, if the sum is close to one—values above one would mean that the agent is using more money than available. In contrast, values below one would indicate that the money is disappearing. If the action is invalid, the algorithm repeats this process in an attempt to optimize the action. When several cycles have passed without converging, the algorithm exits the cycle and uses a softmax operation to output a cash distribution equal to one.

- **Hyperparameters tuning**: Two elements are varied: input data and loss functions.

Input datasets are varied according to the ablation test. Only those that obtained good performance metrics advanced to this phase.

In terms of loss functions, there are three used in this work:

- **Loss M**: a loss function that maximizes actions given stock prices. It is defined as follows:

$$\text{loss}_M = -\sum_{i=1}^{A} a_i \cdot pc_i \tag{4.6}$$

  where A is the number of assets (and actions) available for trading, a_i corresponds to the percentage allocation for asset i, and pc_i refers to the price change for asset i.

- **Loss B**: a logarithmic barrier loss function that maximizes actions given stock prices while allocating the agent's wealth in a given asset up to a threshold value. It is defined as follows:

$$f(x) = -\sum_{i=1}^{A} a_i \cdot pc_i \tag{4.7}$$

$$B(x;\mu_B) = -\mu_B \sum_{i=1}^{A} \log(\text{threshold} - a_i) \tag{4.8}$$

$$\text{loss}_B = f(x) + B(x;\mu_B) \tag{4.9}$$

  where A, a_i, and pc_i have the same meaning as before, μ_B refers to the barrier parameter, and threshold is a value indicating the maximum cash allocation for any asset.

- **Loss BQ**: a combination of a logarithmic barrier loss function and a penalty loss function. The function $f(x)$ maximizes earnings, while the constraints limit the cash allocation. The equality constraint penalizes cash allocations that do not add up to 100%, while the inequality constraint protects against allocating wealth to a given asset beyond a threshold value. It is defined as follows:

$$f(x) = -\sum_{i=1}^{A} a_i \cdot pc_i \tag{4.10}$$

$$B(x;\mu_B) = -\mu_B \sum_{i=1}^{A} \log(\text{threshold} - a_i) \tag{4.11}$$

$$Q(x;\mu_P) = \frac{1}{2\mu_P}(1.0 - \sum_{i=1}^{A} a_i)^2 \tag{4.12}$$

$$\text{loss}_{BQ} = f(x) + B(x;\mu_B) + Q(x;\mu_P) \tag{4.13}$$

where A, a_i, pc_i, μ_B, and threshold have the same meaning as before, and μ_P corresponds to the penalty parameter.

The M loss is used as a baseline that computes the portfolio's expected return—calculated as a weighted sum of the individual assets' returns.

The other two loss functions (losses B and BQ) are created subjectively based on the literature review on optimization to obtain high model evaluation metrics.

The rationale for selecting a logarithmic barrier loss function is to discourage allocating wealth in a small number of stocks. By setting a threshold of 35%, the agent incurs a greater loss as it allocates a larger proportion of wealth to a given stock.

The reason to add a penalty function is to force the agent to use 100% of its wealth on all assets (cash plus stocks). By adding the penalty function, the agent gains the ability to short-sell.

- **Model evaluation**: The return-to-risk methods (listed in section A.1.3 of Appendix A), the annual return, and maximum drawdown are utilized to evaluate the trading system.

4.4 Model testing

Resampling methods are used to compare these metrics and select the best performing model. These methods estimate the test error through the creation of new versions of the original dataset [44, 201, 54, 202]. They include bootstrap, train/validation/test split, and cross validation.

As mentioned above, in this work, cross-validation is used. In particular, due to financial data being time-dependent, **time-series cross-validation** is selected to avoid introducing look-ahead bias to the model. Figure 4.10 displays the different folds in which the data is divided during training and testing. In each iteration, a fold is added as a test fold, while the previous test fold becomes the training fold. The key idea is to prevent future information to be passed to the model beforehand.

For more information regarding these resampling methods, section A.2.2 in Appendix A is the place to go.

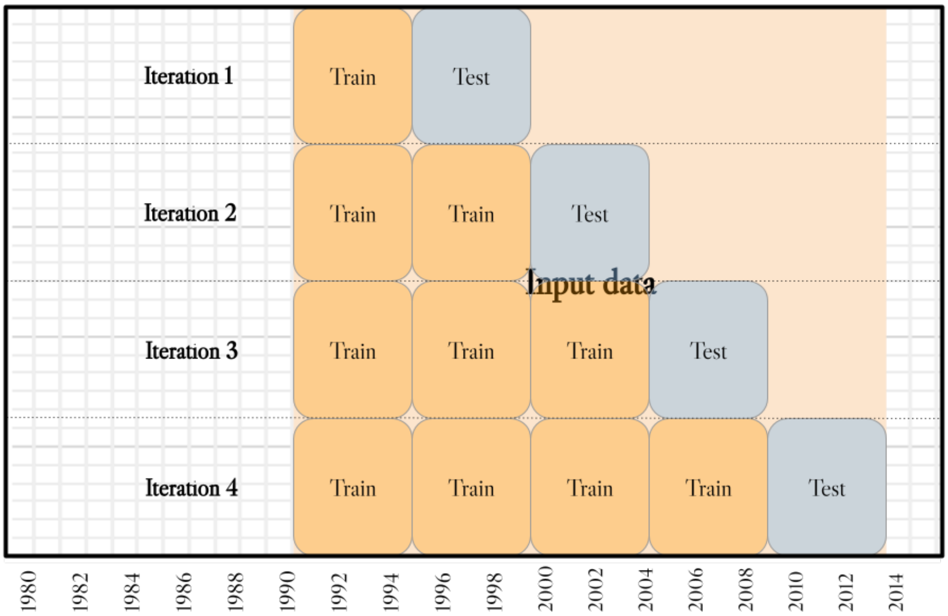


Figure 4.10: Time-series cross-validation. Figure shows the way in which the data is split into folds to prevent look-ahead bias.

For all the experiments, a $k = 5$ is selected as it divides the interval of analysis into blocks of 4.6 years, each containing at least one financial crisis. That is, between 1990-1994, a recession in the U.S. and the beginning of the Tequila effect appeared, between 1994-1999, the 1997 Asian financial crisis happened, between 1999-2004, the dot-com bubble ended, between 2004-2008, the 2008 global financial crisis occurred, and between 2008 and 2013, a period of recession and the 2013 U.S. debt-ceiling crisis affected the economy.

4.5 Summary

This chapter went through the blocks of an ML pipeline to explain the methodology followed to obtain the most suitable ML models and their hyperparameters for diverse

ML tasks, including classification, anomaly detection, dimensionality reduction, and stock trading.

In addition, details regarding the data used during the training process and the experiments and the feature engineering steps to fix and impute data, select, extract, or create features were provided.

Finally, the model testing contained information concerning which sampling method is used during experimentation and model training. By utilizing time-series cross-validation, no look-ahead bias is added.

In the next chapter, the experiments and their results are described.

This page is intentionally not left blank.

5 Experiments

It's not an experiment if you know it's going to work

—Jeff Bezos.

This chapter describes the experiments and their results for the blocks presented in the methodology.

Section 5.1 shows the output of exploring and analyzing the data in search of patterns to use or errors to fix in the following stages.

Section 5.2 describes the experiments regarding news embeddings, dimensionality reduction, and anomaly detection.

Section 5.3 contains the important part of this thesis. This section attempts to find a suitable ML model that can be adapted to the RL trading agent.

This section also contains the ablation test used to determine which data features affect the ROI the most.

5.1 Data acquisition

This section only describes the results of data exploration as no experiment was performed for data collection.

5.1.1 Data exploration

Data exploration describes the key points noticed during data analysis that might help during feature engineering.

5.1.2 Stock market data exploration

In this subsection, the descriptive statistics of stock market returns are computed to summarize data and identify any salient characteristics. In addition, general assumptions about the distribution of financial returns are also tested.

Daily returns are calculated using close prices. Missing data and infinite values are removed from the analysis, accounting for less than 5% of the data.

In addition, betas are also calculated using the same daily returns with respect to the S&P 500 values.

5.1.2.1 Stock market returns distribution

The results are shown in Table 5.1:

Table 5.1: Percentage of positive, negative, and unchanged daily returns.

Number of elements	2407169
Percentage of daily returns with positive change	48.38%
Percentage of daily returns with negative change	46.97%
Percentage of daily returns without change	4.65%

The descriptive statistics of the daily returns are listed in Table 5.2. It can be observed that skewness and kurtosis are both positive, which means that the data is skewed right and is heavy-tailed. In other words, positive returns occur more often than negative returns, and extreme returns occur more frequently than in a normal distribution.

Table 5.2: Descriptive statistics of S&P 500 daily returns.

Min	-0.9667
Max	2.2500
Mean	0.00075909
Variance	0.00063864
Skewness	1.2544
Kurtosis	86.7405

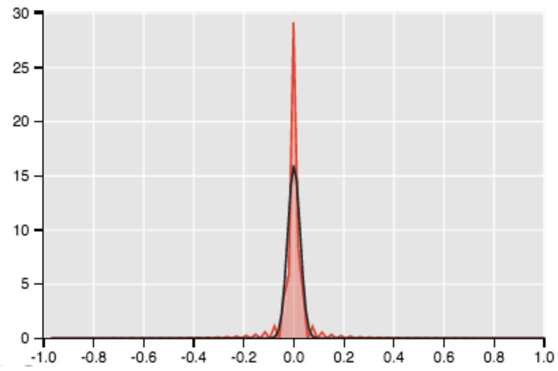


Figure 5.1: Returns distribution using 1000 bins compared to a Gaussian distribution.

In addition, the daily return distribution is compared to a Gaussian distribution given that some parts of the financial industry assume normally distributed returns [203, 204, 205].

In Figure 5.1, it can be noticed that the returns distribution does not approximate a Gaussian distribution. Fama and French wrote that "distributions of daily and monthly stock returns are rather symmetric about their means, but the tails are fatter (i.e., there are more outliers) than would be expected with normal distribution" [206]. Fat tails are a problem because some traders use this or other assumptions to invest in the stock market and introduce noise, making it difficult for models to recognize patterns. So, even though it is a mistake to trade using these assumptions, it is impossible to isolate this effect.

These findings have been noticed in the literature [207]; however, there has not been any significant attempt to modify the neoclassical assumptions, which are at the heart of the financial markets [208].

5.1.2.2 Beta histogram

Similarly, returns can be used to calculate the systemic risk of all the companies forming the S&P 500 index. Figure 5.2 contains the distribution of betas for all the companies combined and all the companies divided by industry.

On the left, it can be noticed that the distribution is unimodal, and the majority of its betas are close to 1.0, indicating that those returns have the same volatility as the S&P 500. On the right, the image shows four significant peaks: health care with betas lower than 1.0, industrials and consumer discretionary with betas near 1.0, and financials with betas greater than 1.0.

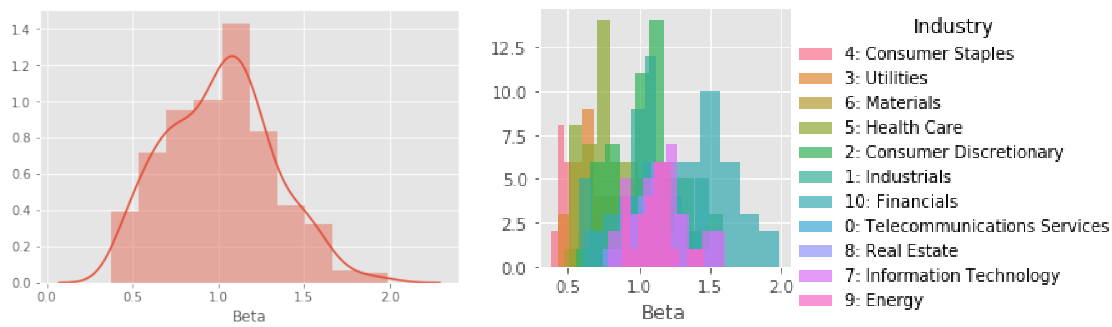


Figure 5.2: Distribution of betas of the S&P 500 index. Left: All companies. Right: Companies divided by industry.

Once the image on the right is further separated, a better picture of the systemic risk of each industry can be presented. This is showed in Figure 5.3. While most industries' betas fall between 1.0 and 1.5, the consumer staples, utilities, and material industries' betas have values lower than 1.0, which implies less volatility. Companies in the financial industry exhibit a wider set of beta values ranging from 0.6 to 2.0. The majority of them have a beta of 1.5—this means that companies are 50% more volatile than the S&P 500 index.

There are two points in showing the beta histogram in Figure 5.3. On the one hand, it provides general insight on the behavior of stock prices with respect to the market and helps validate that the S&P 500 index is a representative market sample—which seems to be, given that the systemic risk or beta coefficient is close to 1.0.

On the other hand, it indicates that industry information could help the model classify companies according to their risk levels. For example, a model could exploit strategies that invest in companies with large betas during economic growth—expecting the price of these companies to increase more than the price of the benchmark—and in companies with small betas during economic contraction—anticipating the price of these companies to decrease less than the price of the benchmark.

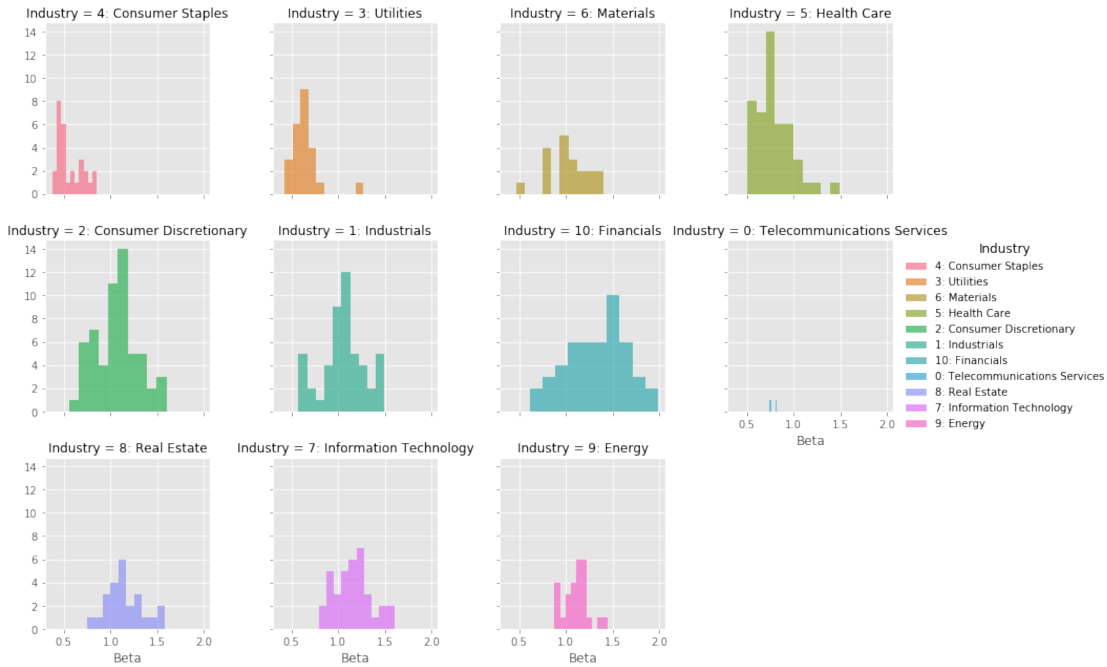


Figure 5.3: Distribution of betas of the S&P 500 index.

5.2 Feature engineering

5.2.1 Clustering news using news embeddings

This experiment assesses whether textual data can be summarized using embeddings so that a variable number of daily news articles can be passed to an ML model while preserving salient information.

The news dataset is filtered to keep 10,508 articles mentioning S&P 500 companies. Then, each news is transformed to one news embedding using a variation of the BERT model, i.e., RoBERTa. The resulting embeddings are then passed to a KMeans model to group them into one of 50 clusters.

A subsample of these articles was combined with their corresponding cluster, and name, sector, and industry of the company referred in the news and projected to a two-dimensional space using the t-SNE algorithm.

Figure 5.4 shows the 50 clusters and a corresponding label based on the name, sector, or industry.

It can be noticed that the news embeddings capture key information that the KMeans and t-SNE algorithms use to project vectors with similar information close to each other.

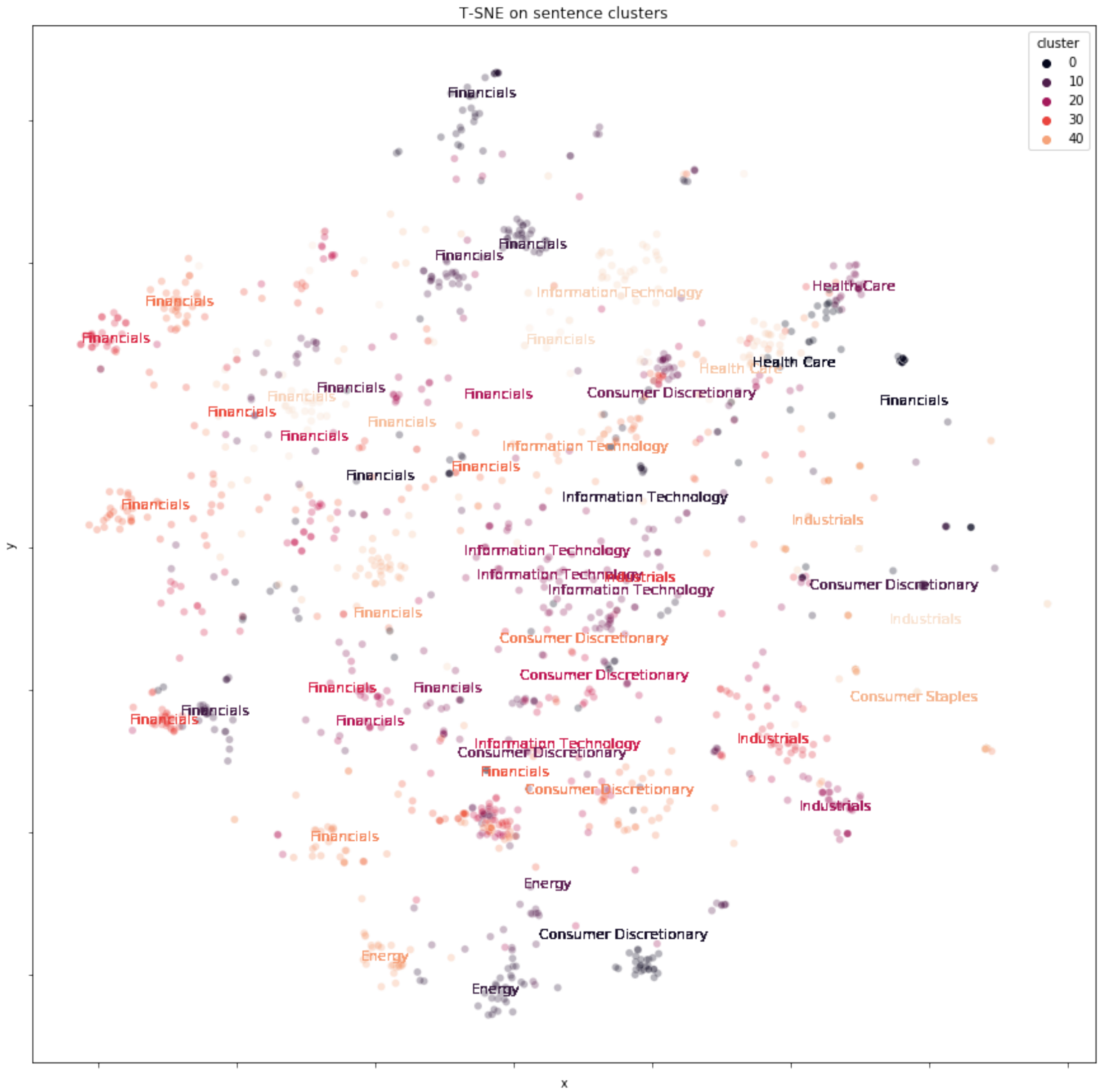


Figure 5.4: 2D-cluster projection of news embeddings by sector.

It can be observed in figures—5.4, 5.5, and 5.6—that news embeddings can extract critical information from news articles as similar articles are close to each other.

The grouping capabilities of news embeddings can also be confirmed in Figure 5.7, where the news in one cluster is sampled to review the key information contained within the embeddings. More news from that cluster can be found in Table C.12 in Appendix C.

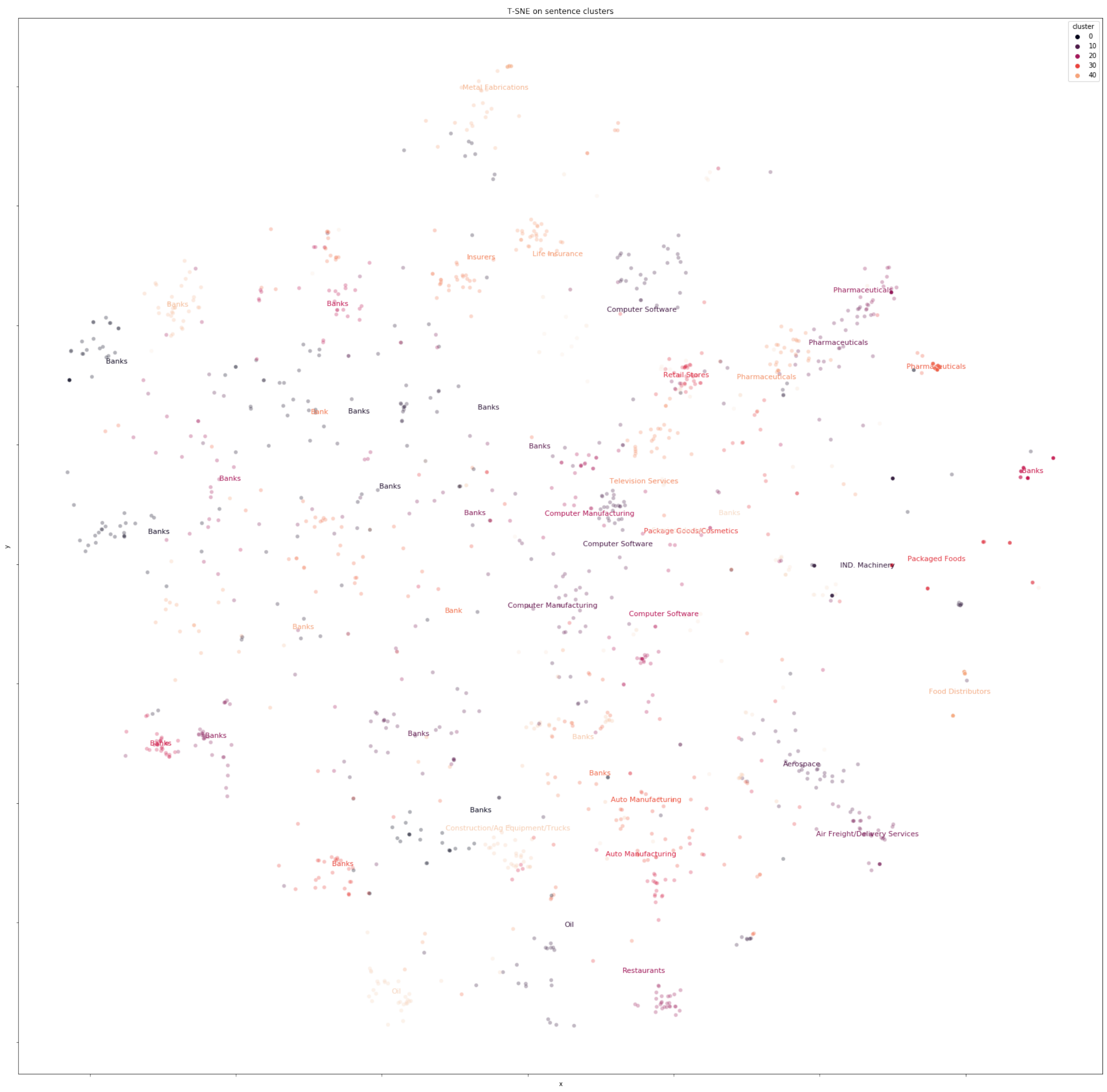


Figure 5.5: 2D-cluster projection of news embeddings by industry.

By examining the figure and table, the news articles contain information about pharmaceutical companies, such as Johnson and Johnson and Pfizer. In particular, this cluster seems to include negative news from Johnson and Johnson, suggesting that news embeddings summarized the information based on a deeper understanding of the news and not only based on a superficial attribute, such as its industry. The only exception appears to be the one colored in orange as it contains the name Johnson and Johnson without being a pharmaceutical story.

These results show that news embeddings can extract the article's essence into vectors that get clustered near each other in terms of sector, industry, company, and sentiment polarity. With these embeddings, the RL model might encapsulate high-dimensional articles in low-dimensional states that help it redistribute cash from companies with

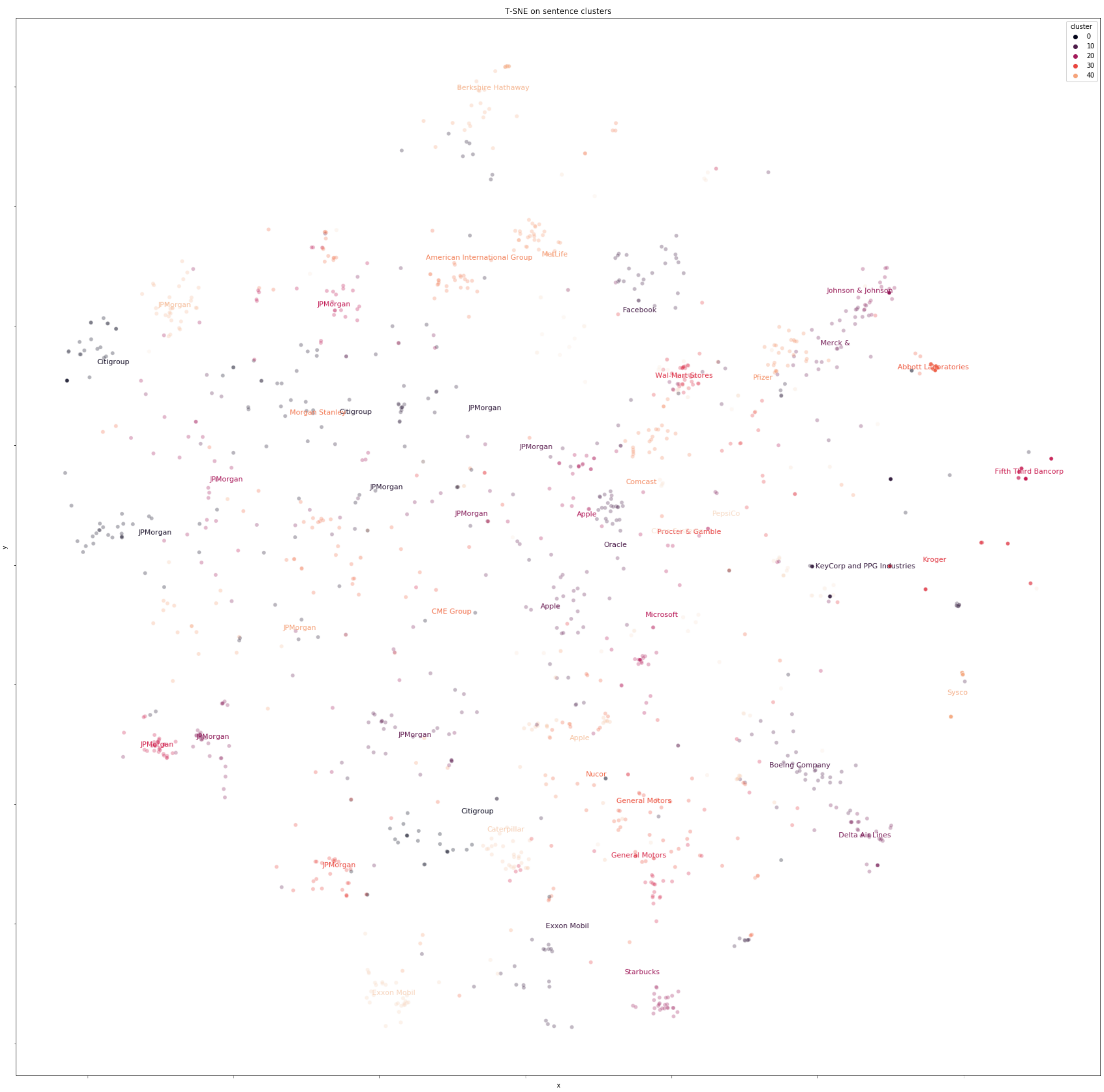


Figure 5.6: 2D-cluster projection of news embeddings by corporation.

specific types of negative news to companies with positive ones—*specific types* in the sense that not all positive or negative news articles affect stock prices.

These news embeddings have a fixed size that can be directly passed to the RL model. However, there are several news articles written per day. For that reason, three approaches were considered: 1) to average the embeddings to represent the general expectation over the economic environment; 2) to add the embeddings to represent a particular economic situation; 3) to combine each news with the economic state of the day.

After careful consideration, options 2) and 3) were discarded, and option 1) was used in this thesis. On the one hand, option 2) was discarded because adding the embedding vectors resulted in a drastically isolated point. On the other hand, option 3) was discarded because it required more complex logic to split the training and test folds so that all the news embeddings from one day were put on only one of these folds to prevent look-ahead bias.

In the latest in a series of account consolidations, Johnson & Johnson has consolidated its accounts for oral care products, naming McCann-Erickson New York to handle Reach toothbrushes, Act fluoride rinse and Johnson & Johnson dental floss.

By the end of the decade, Prudential insurance policies, Johnson & Johnson Band-Aid boxes and Time Warner's billion copies of magazines a year may all be made of recycled paper, under a project announced yesterday by these and other companies working with the Environmental Defense Fund.

In the months before a Johnson & Johnson unit removed a controversial home H.I.V. test from the market, Federal regulators issued stern directives to the company to clean up its quality control procedures, Government records show.

In a sign that American drug companies are cracking open the Japanese market, Johnson & Johnson will announce today that Tylenol will be available in Japan early this fall.

The government is conducting a criminal investigation into a Johnson & Johnson factory that makes an anemia drug that has been linked to a spate of serious illnesses in Europe and Canada, according to court documents and people close to the situation.

Based on antibodies found in llamas, said an experimental rheumatoid-arthritis medicine licensed to Pfizer Inc. (PFE) was found effective in a mid-stage clinical study, validating the Belgian company's technology for the first time.

Figure 5.7: News extracts from one of the clusters using news embeddings.

5.2.2 Dimensionality reduction

5.2.2.1 Best activation function

In this experiment, an autoencoder is evaluated in a dimensionality reduction task. The dataset contains only stock features (OHLC, adjacent close, and volume) of 203 companies to guarantee a confidence level of 95% and a margin of error of 5%. Each company's dataset was reconstructed individually, but the reconstruction errors were averaged over the number of companies.

Four experiments were run: a baseline with the same number of input and output features (6 dimensions), a model with 50% dimensionality reduction (3 dimensions), a model with 84% dimensionality reduction (1 dimension), and a model with the same number of input and output features, but removing dissimilar features, i.e., OHLC prices were preserved, and volume was discarded (adjacent close was also discarded, but it could have been used given that it is similar to the OHLC prices).

The model was created using three layers for the encoder with 64, 32, and n (the target dimension) neurons and three layers for the decoder with 32, 64, and s (the original input dimension) neurons. In addition, four activation functions are tested, but each model uses the same activation function in all six layers. These activation functions were the sigmoid, tanh, ReLU, and swish activation functions. The architectures were trained using an Adam optimizer with a learning rate adjusted using an exponential decay rate of 0.96 for every 10 thousand steps.

Table C.9 contains the results of this experiment. *Tanh* was the best activation function for the dimensionality reduction task as it outperformed the second place, the swish activation function, in most experiments. In the last experiment, it can also be noticed that the reconstruction error for models using features with similar characteristics was the smallest. This result confirms pre-experimental observations where reconstruction errors for models using drastically different datasets (e.g., fundamental data and currency rates) were large and never converged. In contrast, datasets derived from others could be reduced together (e.g., stock, cumulative, and ranking features, and factor and fundamental features).

5.2.2.2 Accuracy and ROI

An autoencoder was used for this experiment. The encoder reduced the number of features in the input by 50%, with 128 and 64 neurons for the first two layers. The decoder also had three layers with 64 and 128 neurons in the first two hidden layers and the original shape in the last layer. The activation function used was tanh, as the previous experiment showed it had the lowest reconstruction error. The model also uses an Adam optimizer with the same configuration as in the previous experiment.

Different datasets covering the timeframe from January 1, 1990, to January 1, 2013, were tested. They included stock, technical, fundamental, factor, textual, macroeconomic, temporal, commodity, currency, benchmark, cumulative and ranking features, and stock exchange indexes. These datasets were reduced one by one because previous experiments had shown that combining datasets with dissimilar characteristics, e.g., OHLC data with volume or with GDP, resulted in a higher reconstruction error.

The experiment was repeated five times for the NN and RF models and the results were averaged. The task consisted of predicting the price change direction of one company.

The results are shown in tables C.10 and C.11 in Appendix C.

In Table C.10, the NN model consistently outperformed both RF models in terms of accuracy. On the other hand, while both NNs got similar accuracies, the dimensionality reduction NN model obtained the highest accuracy with 68.60% and stock, technical, fundamental, textual, and macroeconomic features. This result was a surprise because the model with fewer dimensions was expected to perform a little bit worse. However, this result might be explained due to the weights' random initialization or useless information (and noise) being eliminated during this process.

In Table C.11, average earnings by models with and without dimensionality reduction are compared. In terms of frequency, neither of those models outperformed the others. However, in terms of earnings, the RF with dimensionality reduction got the highest ROI, i.e., 38.60%, using stock, technical, fundamental, factor, textual and temporal features.

5.2.3 Anomaly signals

For the anomaly detector model, an autoencoder was trained to generate the MSE for each data type. Then, a function was used to separate normal data points from outliers in different datasets. These outliers could signal investment opportunities to be exploited by the agent.

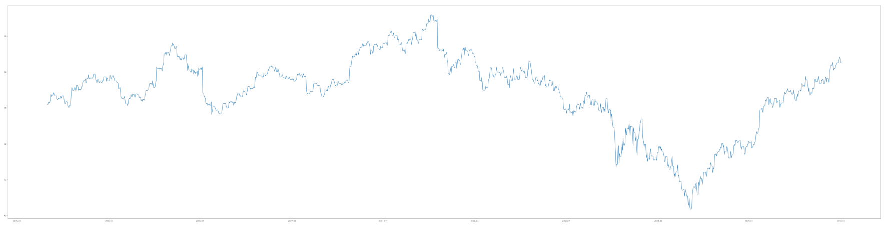
The data was preprocessed using a log transformation to train the anomaly detector. The log transformation is used as a first approximation to this problem, but if time permits, additional transformations will be tested. This is because this transformation is often appropriate for economic and financial time series [169, 38, 170].

The four operations used were minimum, mean, median, and maximum. In each experiment, one of these operations was applied to the loss values to obtain a threshold value representing normal data.

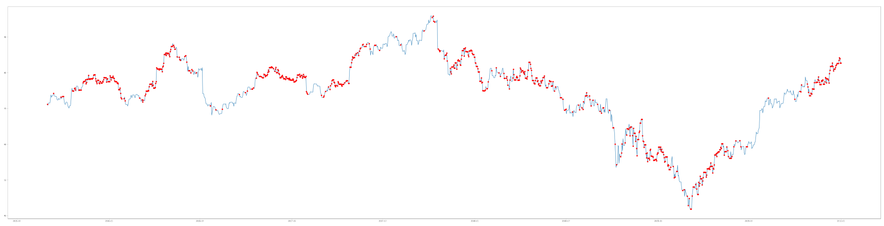
In addition, two anomaly detector models were tested: an FC autoencoder and an LSTM autoencoder.

Figure 5.8 shows the anomaly signals raised by the NN model. In the case of the max function, no signals were identified because the threshold created by this function was too drastic that no values in the test set passed the threshold. Conversely, the min function created a threshold so low that every point was a signal. For this reason, it did not seem feasible to use any of these four operations in the RL model.

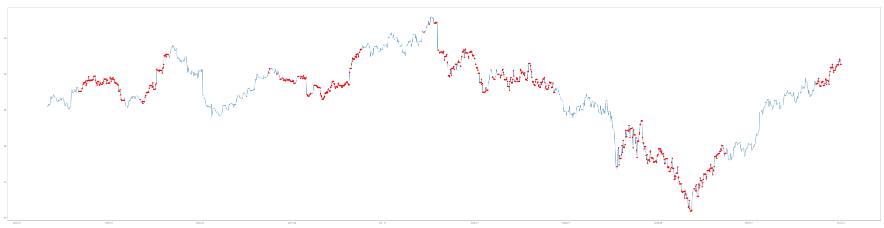
For the other two functions, mean and median, it can be noticed in the same figure that although these functions generated fewer anomaly signals than the min function, these



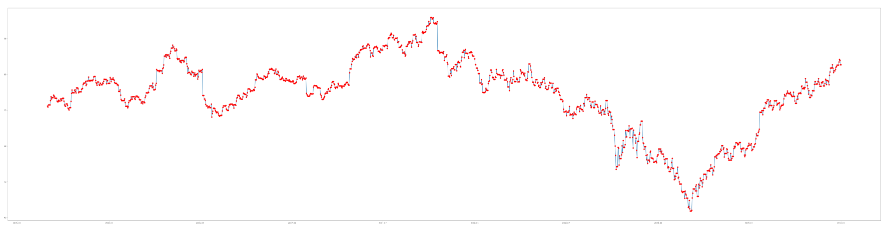
(a) **Max** function



(b) **Median** function



(c) **Mean** function



(d) **Min** function

Figure 5.8: Anomaly detection signals raised by the NN model using the max, median, mean, and min functions.

signals were still numerous causing a lot of false positives. For this reason, they were not suitable for anomaly detection.

Similar results were observed for the LSTM model and can be found in section C.6 in Appendix C.

5.2.3.1 Bollinger anomaly signals

An alternative to the functions used in the previous experiment was adding Bollinger bands one or two standard deviations away from the mean.

The bands were used as follows:

- When the value touched the lower threshold, a buy signal was raised.
- When the value touched the upper threshold, a sell signal was raised.

The result of this is shown in Figure 5.9. It can be noticed that the number of anomalies was drastically reduced to just a few. In addition, some of those trading signals appeared to be profitable.

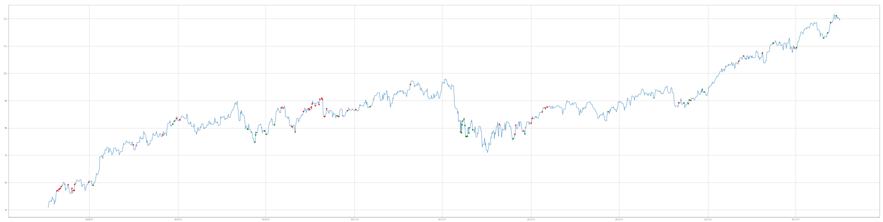


Figure 5.9: Anomaly detection with NN model using Bollinger bands.

5.3 Model selection

5.3.1 ML Model Selection

This experiment involves testing the NN, LSTM, RF, and SVM models. A grid search of their hyperparameters was used to test these models. These hyperparameters are listed next:

- **SVM**: C (0.01 - 1) and the kernels (RBF, polynomial, and sigmoid) were varied.
- **NN** and **LSTM**: neurons ([(25, 50, 2),(100, 200, 2),(256, 512, 2)]) and activations (ReLU, tanh, swish, and sigmoid, plus a softmax layer as the last layer) were varied.

- **RF**: the number of estimators (25, 50, 100, 200), the maximum depth (25, 50, 100, 200, 500), and the minimum samples to split (2, 25, 50, 200) were varied.

For RF, LSTM, and NN, the number of trials was 5. For SVM, the number of trials was only one as it always converges to the optimal solution. Additionally, these models were evaluated using F1 score, accuracy, and log loss.

The results of this experiment are presented in section C.4, Appendix C.

Unfortunately, the results seem to vary significantly. In the case of RF, maximum depth seemed to be the parameter that affected the metrics. When the maximum depth increased, the accuracy increased, but the F1 score and the log loss decreased, and the other way around.

In the case of NN and LSTM, epochs did not significantly affect these metrics, but the number of neurons and the activation function influenced them. In particular, tanh and swish seemed to obtain the highest accuracy and the lowest F1 score. However, something curious was that the models achieved high accuracy when the number of neurons was small, i.e., the combination (25, 50, 2) outperformed the combinations (100,200,2) and (256, 512, 2). This difference might be explained by the models with more neurons overfitting the data quicker.

Finally, in the case of SVM, the kernels seemed to influence the metrics. In particular, the polynomial kernel achieved high accuracy and high log loss with almost any value of C. Regarding the RBF kernel, it seemed to work only to increase accuracy as both the F1 score and the log loss collapsed with this kernel.

One problem with SVM is that the model did not converge beyond a certain number of data points and features. For that reason, this model was discarded for the subsequent experiments.

5.3.1.1 Ablation of data sources

It is also important to identify which datasets influence trading results the most because using irrelevant datasets could introduce noise and affect convergence, and because by using less data, models converge faster. To this end, feature ablation—a technique to calculate features importances—is used.

This experiment was designed to do so by training the top ML models from the previous subsection, 5.3.1, to predict the price direction of a company (3M).

Different data sources were combined and input into the model. After training, the model was evaluated using earnings and accuracy.

Given that there are 13 datasets, it would be unrealistic to try all data combinations as it would require 8192 experiments. Plus, as Lopez de Prado mentions, it would probably suffer from data-mining bias. Instead, the following procedure was followed to reduce these combinations drastically:

- Stock features were always added.
- Technical, fundamental, factors, textual, macroeconomic, and temporal features were combined systematically, resulting in $2^6 = 64$ combinations.
- Rankings, cumulative, benchmark, stock exchange indexes, currency, and commodity features were added in tandem, one after the other, only after all the features in the previous group had been added. This process resulted in 6 additional combinations for a total of 70.

Tables C.7 and C.8 in Appendix C contain the results of this experiment.

Table C.7 indicates that the NN achieved the highest accuracy, 68.59%, among the models when it used stock, technical, fundamental, textual, and macroeconomic features.

The second table, Table C.8, shows a different picture. The three models were more competitive than before, but the LSTM model outperformed the other two by earning the highest ROI with all but commodity, currency and index features.

Although performing the feature ablation experiment several times does not help measure the relative importance of the dataset types—i.e., identify which features have greater impact in forecasting daily price changes—it does allow to identify the overall conclusion, that is, NNs outperform LSTM and RF in terms of accuracy, but underperform LSTM in terms of ROI.

5.3.2 Historical RL Model Selection

5.3.2.1 RL model experiment

In this experiment, an agent was trained using RL models to assess which one obtains higher earnings and to evaluate subjectively whether their strategies are sensible.

Every episode, technical data from a new company was selected, and the agent interacted with the environment until he ran out of money or the data for that company was exhausted.

Technical data contained ten non-normalized features and takes three values -1, 0, and 1.

Stock market prices were used to compute the rewards.

The state was defined as a tuple $(3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 11)$, where the last digit represented the number of industries.

Q-learning state representation experiment.

This experiment compared two ways of representing states for the Q-learning model resulting in: tuples with three dimensions for each technical signal—i.e., $(3_1, ..., 3_k, 11)$—and tuples with one dimension of m-levels where $m = (2 \cdot technical\ signals) + 1 = 21$, and 11 industry sectors—i.e., $(m, 11)$.

The calculation of the number of complex and simple states and Q-table indexes can be observed in Figure 5.10. The numbers of Q-table indexes in Table 5.3 are derived from this figure.

State tuples in the first scenario were directly used as matrix indexes to update the Q table, and in the second scenario, they were first summed, and the result was then used as a matrix index.

Table 5.3 displays the experiment results where the complex state achieves lower average reward and average earnings while maintaining a lower standard deviation than the simple state. Conversely, the simple state achieves better results at the expense of a higher standard deviation. It generates a Q-table with significantly fewer elements ($2,079$ vs. $5,845,851$) and fewer zeros (26.26% vs. 98.91%) which means that the simple state generates less sparse Q-tables. However, in both cases, the Q-tables are large, which means that the agent would need to explore the space for many iterations. A

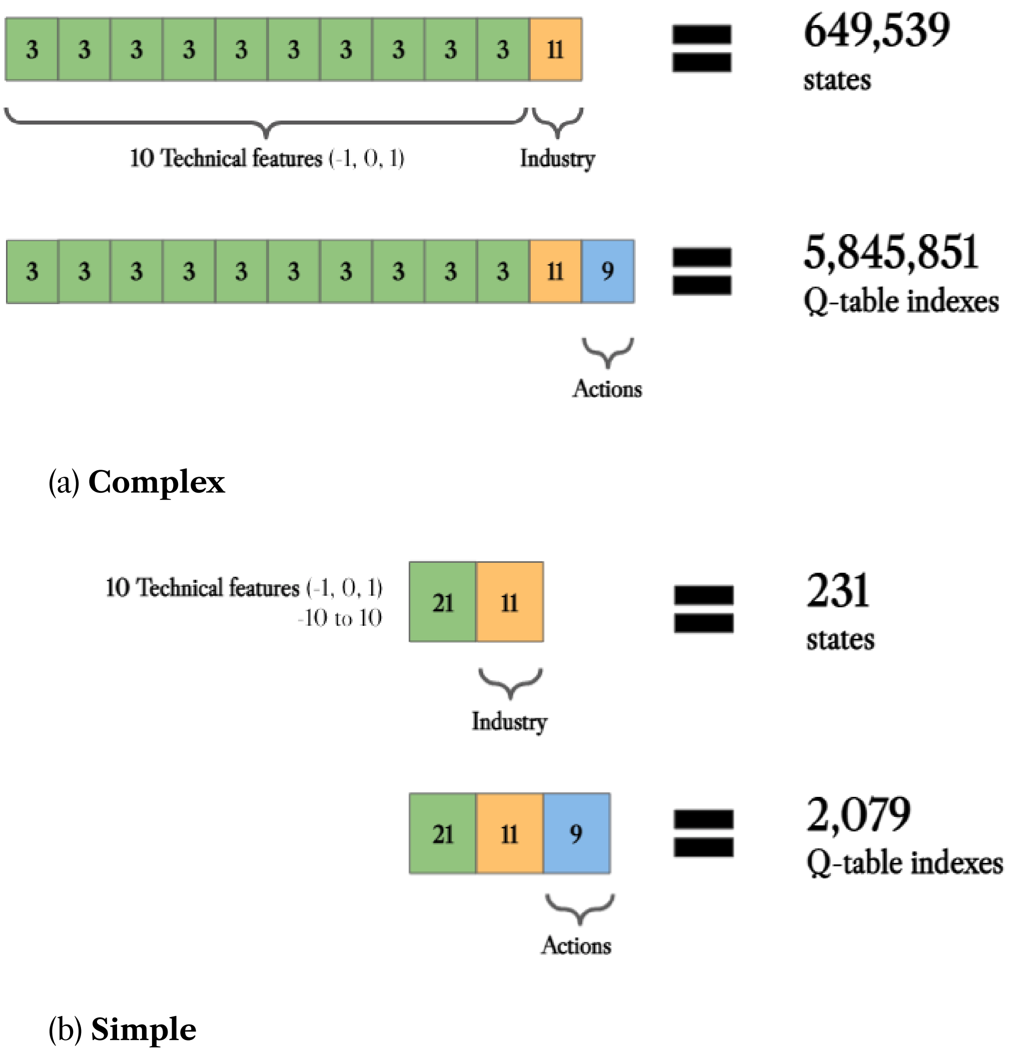


Figure 5.10: Calculation of the number of complex- and simple-states and Q-table indexes.

significant limitation would be that stock market data is insufficient for this approach compared to the billions of states that can generate other environments, such as the Atari games or the Go game.

In addition, neither using a simple nor complex model resulted in positive earnings suggesting that the agent does not find any profitable strategy.

Table 5.3: Accuracy of complex and simple states.

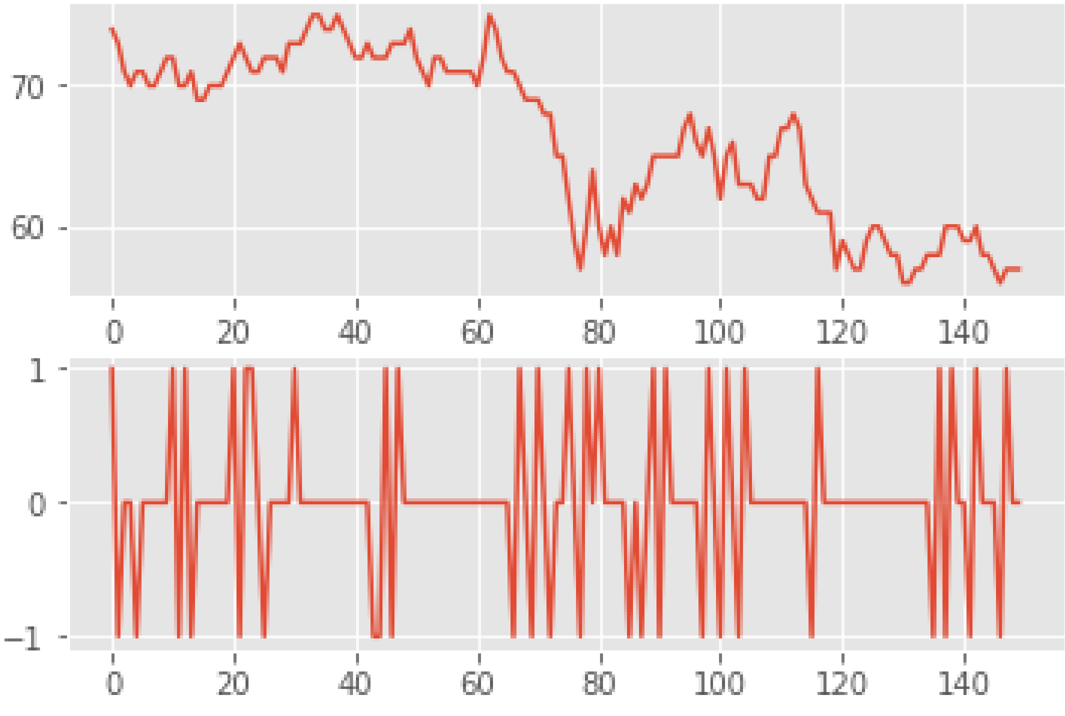
	Q-learning with complex state	Q-learning with simple state
Average reward	-0.0028	**-0.0021**
Average earnings	99734.62	**99798.76**
Standard deviation	**0.0135**	0.0221
Elements in Q-table	5,845,851	**2,079**
Percentage of zeros	98.91	**26.26**

In Figure 5.11, it can be observed that the simple agent selected actions less randomly and, it appears like the agent refrained from buying when there was a downward trend.

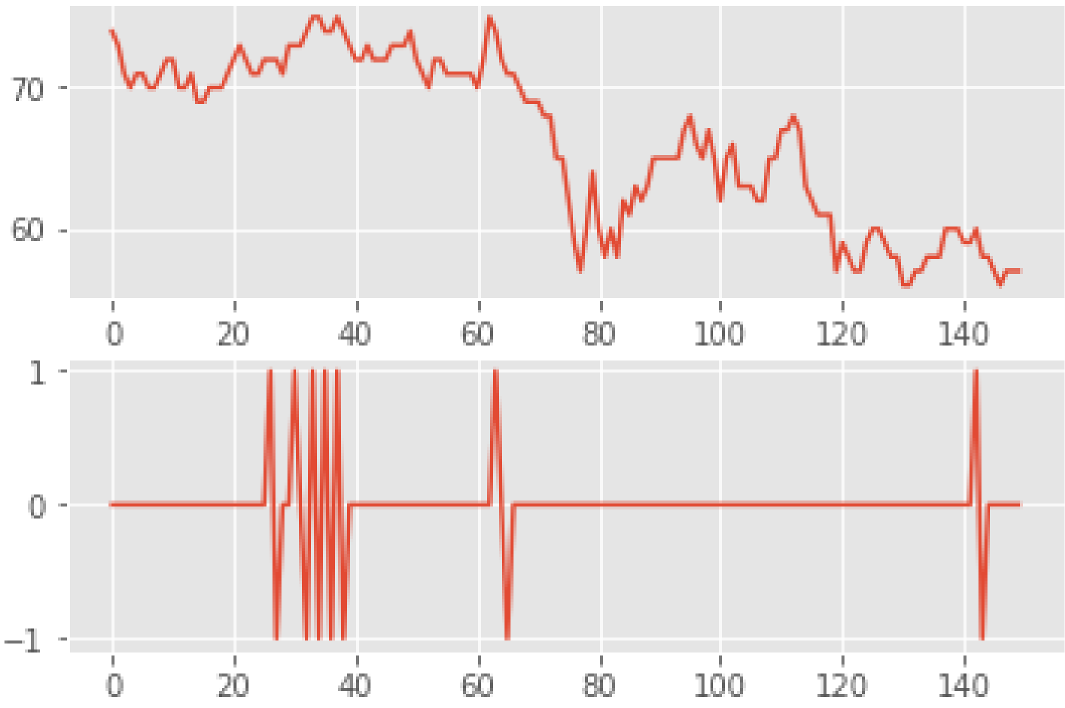
Given the high-dimensionality of simple and complex states, an alternative is needed to drastically summarize them and select better actions that maximize the agent's earnings.

5.3.3 RL Model Selection

For the experiments in this section, the stock market environment was run for ten epochs with random date initialization during training but starting at $t = 0$ during testing.



(a) **Complex** state



(b) **Simple** state

Figure 5.11: Prices and actions taken by the complex-state and simple-state agent.

5.3.3.1 Loss functions

The agent was trained using a NN model with different types of losses (refer back to chapter 4 for additional details on these losses) to optimize. The output of this network had a softmax activation function to allow only long positions for losses M and B but a linear output to allow short-selling for loss BQ. However, to avoid getting stuck on local optima, the agent had a limit in the number of iterations it performed.

The results of this experiment are shown in Figure 5.12. The image shows that losses that did not limit cash allocation, such as **loss M**, achieved significantly higher ROI during periods of strong economic growth (i.e., 1993-2000). However, when the economy grew slowly, suffered a financial crisis, or was in the middle of a recession (2001-2013), the agent suffered large losses. Conversely, when losses had cash allocation limits, such as losses B and BQ, ROIs were less impressive in times of economic growth but positive in times of economic turmoil.

A disadvantage of using the BQ loss was that it took a significant amount of time for each action to converge. When it did not converge, the stop mechanism used a softmax action that biased the agent towards long positions from that moment onward, preventing the agent from taking advantage of short-selling. Conversely, the B loss was significantly faster at the expense of not being able to short-sell.

When the stop mechanism was removed, actions were much larger than 1.0, which meant that, even when there was a penalty for trading using more cash than the cash available, the agent willingly accepted that loss penalty and the market risk to earn more money. Was the agent mimicking the risky behavior of Wall Street's fund managers?

5.3.3.2 Fixed loss experiment

For this experiment, the loss type (loss B) was fixed while the models varied the number of companies, the input data, and the type of data transformation used.

The nine models used are listed in Table 5.4.

The results are shown in Figure 5.13. It can be noticed that there is no significant difference between the models using 50 companies regardless of the configuration. It appears that the agent could not find any helpful feature to exploit and increase its wealth. Instead, it decided to diversify its portfolio to eliminate the portfolio risk (as

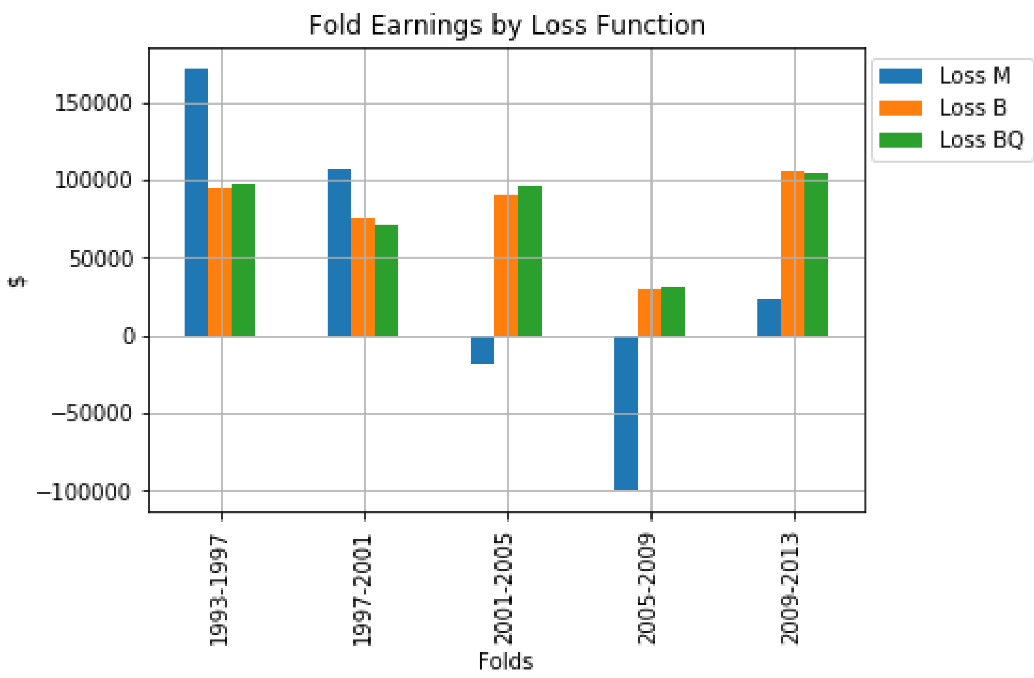


Figure 5.12: ROI per fold and loss functions.

mentioned in Appendix A section A.1.5).

The diversification scenario is supported by the logs, which track the maximum and minimum asset allocation at the end of each test phase. On average, for all the models using loss B, the difference between the maximum and minimum allocations is less than 0.05244% which means that the agent's wealth is allocated almost evenly across assets.

However, there is a difference between the model using 425 companies and all the models using 50 companies. In the same figure, it can be noticed that the models earned similar ROIs when the economy was growing. This changed during slow-growth and the recession periods as the models investing in 50 companies performed better than the model investing in 425 companies. Conversely, during the 2008 financial crisis, the diversified model outperformed the concentrated model, but only by half of what the concentrated model earned during the slow-growth and recession periods.

Another difference between the diversified and concentrated model was the number of trades made. As the number of companies increased, the number of trades also grew. This result seems reasonable as the agent had to trade at least once the shares of those 425 companies.

Table 5.4: Model configurations.

Model	No. of Corp.	Transformation	Datasets
1	425	% change	Stock, cumulative, ranking, dates.
2	50	% change	Stock anomalies, cumulative, ranking, dates.
3	50	% change	Dates.
4	50	% change	Stocks.
5	50	% change	Stock Dim Redux.
6	50	% change	Stock anomalies.
7	50	% change	Best dataset for accuracy.
8	50	Rolling norm	Stocks, cumulative, ranking, dates.
9	50	% change	Stocks, cumulative, ranking, dates.

5.4 Model testing

This section uses the results from the previous experiments to obtain the financial metrics via Pyfolio and compare the trading agents. These numbers are shown in Tables 5.5 and 5.6.

Table 5.5: Results of different loss functions.

	Loss M	**Loss B**	**Loss BQ**
Annual return	1.80%	9.60 %	9.53%
Sharpe ratio	0.19	0.77	0.76
Omega ratio	1.05	1.18	1.18
Sortino ratio	0.28	1.11%	1.10
Max drawdown	-82.82 %	-39.04%	-39.364%

The table indicates that the loss M model performs worse than the loss B and BQ models, with lower metrics. In particular, the loss M model suffered a maximum drawdown of 82.82%, which represents a significant wealth loss. The diversified models also suffered drawdowns, but they were 50% lower than the loss M model's value. In addition, the diversified models obtained very similar results, but the loss BQ model took more time to train. Thus, the loss B model seems to be better than the other two models.

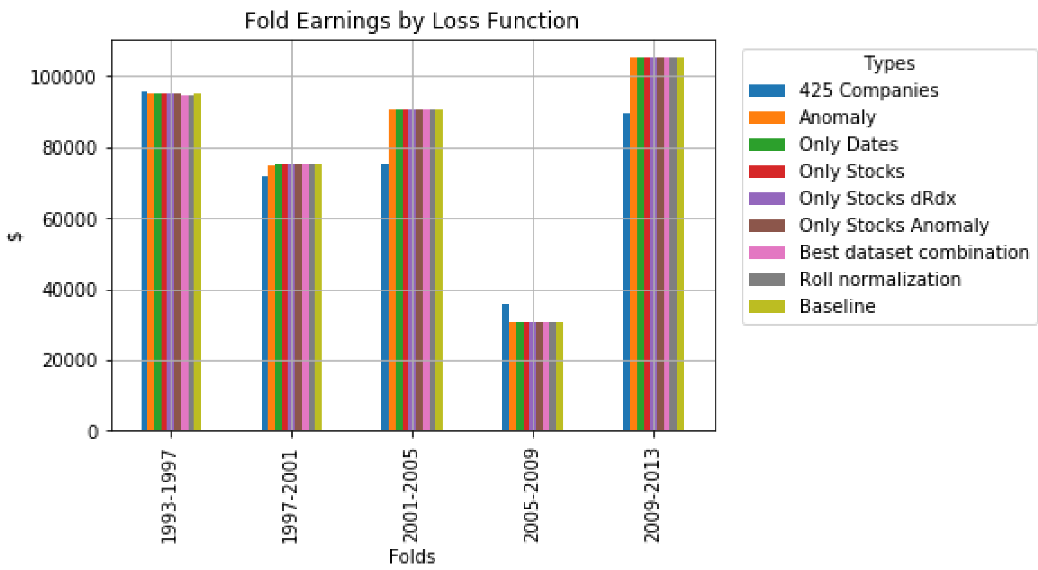


Figure 5.13: ROI per fold using a fixed loss while varying input data, number of companies, and data transformation methods.

Although the agents could trade by diversifying their portfolios, the results suggest that adding that exit loop into the algorithm might have forced the agents to converge prematurely. That is, the agents were not even trying to short-sell the stocks.

In particular, it did not matter the type of input data, preprocessing, additional features added as the agents diversified their wealth without taking any risk.

Table 5.6 was intended to be bigger, but most of the models obtained similar results. This table shows the financial metrics for different loss B models. As it can be noticed, these values are very similar, suggesting that the agents ignored the data and diversified their initial capital across the assets. Their Sharpe ratios indicate that the diversification strategy earns positive returns, but the risk of an adverse event might affect earnings. The Omega ratio confirms that the strategy earns above-average returns, with a small probability of suffering extreme losses. As for the Sortino ratio, it indicates a positive return per unit of downside risk. Finally, the *L 425 model* achieves the lowest maximum drawdown due to being completely diversified.

Figure 5.14 compares four models: loss M, B, and BQ models, and the 425-companies model. Each column contains, from top to bottom, the comparison between the portfolio's and benchmark's cumulative returns, the 6-month rolling Sharpe ratio, the top 5 drawdown periods, the monthly and annual returns, and the distribution of monthly returns.

Table 5.6: Financial metrics of different model variations using loss B.

	L 425	L Anomaly	Best DC
Annual return	9.004%	9.602%	9.60%
Sharpe ratio	0.780	0.770	0.77
Omega ratio	1.18	1.18	1.18
Sortino ratio	1.12	1.11%	1.11
Max drawdown	-36.069%	-39.084%	-39.112%

The first plot, portfolio's vs. benchmark's cumulative returns, is similar for the loss B, loss BQ, and 425-companies models but different for the loss M model. For the loss M model, the returns collapsed during the financial crisis and barely recovered by the end of the investment period. Although the agent almost duplicated the initial capital, it underperformed the benchmark and the other portfolios.

For the other three models, the plots have a similar behavior due to the agents diversifying their capital across the assets, causing the cumulative returns to follow the benchmark's returns closely. However, during the 2008 financial crisis, they diverged, and the agents' cumulative returns outperformed the benchmark's and loss M model's cumulative returns. As a result, these three models earn 12 (425-companies model) and 14 (the other two models) times their initial capital.

In the second plot, the rolling Sharpe ratio is similar for the diversified models, ranging from -2 to 4 and an average value close to 1.0. For the loss M model, the rolling Sharpe ratio ranges from -4 to 4 and an average value near 0.2, suggesting the model's strategy is significantly more risky than the diversified models' strategy. This is confirmed by the previous results, given that the loss B model's earnings were underwhelming.

The top 5 drawdown periods plot shows those intervals during the investment period where the strategies lost money. For the loss M model, a vast purple block on the right side of the plot indicates the collapse of the agent's strategy (i.e., investing all its capital in only one stock) as it loses most of its earnings. There are small purple blocks for the diversified portfolios, which means that their diversification strategies were effective. Their earnings at the end of the investment period can be confirmed.

Finally, the returns plots show another perspective of the previous results. For the loss M model, the monthly returns plot shows strong returns during the first years, but meager returns from 2001 onward, confirmed by the distribution plot, which has a prominent peak near 0%. In terms of annual returns, the blue plot shows that the average value was near 5% and was negative during the 2008 financial crisis, with a shocking loss of 40%.

In the case of the diversified models, the monthly returns showed strong results throughout the period, especially in 2003, when the model achieved a 38% annual return. Although these models could not avoid losses during the 2008 financial crisis, they lost less capital than the benchmark and the loss M agent.

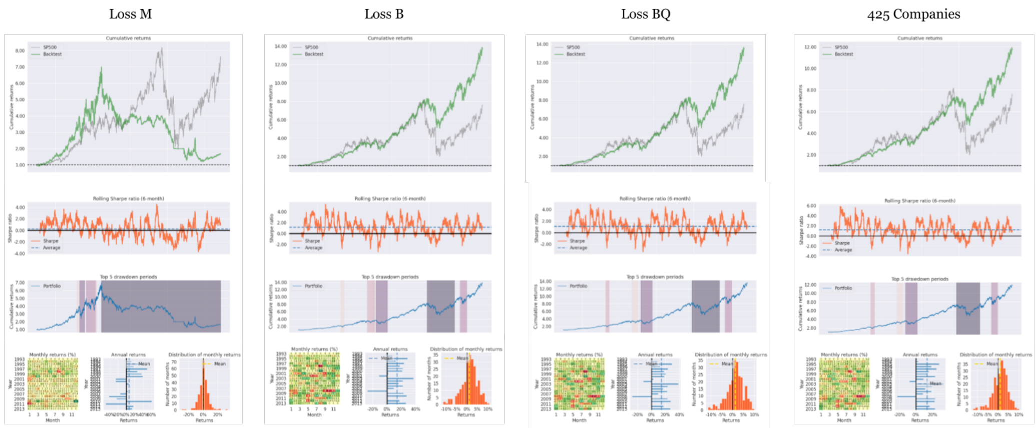


Figure 5.14: Analysis of portfolios. Comparison between portfolios and benchmark, rolling Sharpe ratio, top 5 drawdown periods, and returns

5.5 Summary

In this chapter, we performed the experiments and analyzed the results. Although there were some disappointing results, the models were able to generate positive ROIs that are higher than the benchmark.

In the following chapter, we conclude this thesis.

This page is intentionally not left blank.

6 Conclusions

The most dangerous people on the planet are those who sincerely believe something that is false
—Steve Keen.

In the last chapter, key findings and contributions of this thesis are summarized, and directions in which this work can be expanded and enhanced are suggested.

6.1 Conclusions

This thesis studied the investment problem and attempted to answer what information is needed to summarize the stock market, what ML model works best for financial data, and whether an ML model can learn to distribute wealth so that profit is maximized while risks are minimized.

Regarding the first objective, diverse financial and economic datasets were collected from private provides and government agencies with the assumption that these datasets contained enough information to represent the economic state of the stock market. The collected data included stock market information, fundamental data, stock exchange indexes, commodities, currency exchange rates, macroeconomic data, and textual data covering the time frame from 1987/01/02 to 2013/11/01.

To fulfill the second objective, the application of feature engineering techniques to the different types of datasets allowed the creation of a multimodal input that helped the model represent the financial and economic state of the U.S. economy without introducing biases.

Datasets were preprocessed using different techniques to identify the most suitable technique for each data type. It was found that the use of the LOCF method and a zero value (for data that did not exist at a particular point in time), a limited number of runs per experiment, cross-validation for time series, random initialization, a broad time frame, and a suitable financial index (i.e., S&P 500) helped avoid introducing the look-ahead, data-mining, sampling, and time-period biases.

Regarding data transformation, log transformation was the best option for stock prices and similar data, and rolling normalization helped standardize datasets with large values.

Unfortunately, it was impossible to determine the most suitable transformation or the ideal input features in terms of ROI maximization because the ML models could not effectively reduce the datasets' high-dimensional space.

In terms of feature extraction, word and event embeddings assisted NLP models to achieve their goals (i.e., creating new features and summarizing news articles). Low-dimensional data containing the essential datasets' elements—obtained using dimensionality reduction models—helped models train faster and obtain similar results than the original data, and anomaly signals—triggered by anomaly detection models—indicated abnormal events in the economy.

These last two models—i.e., anomaly detection and dimensionality reduction models—were based on the autoencoder architecture.

The anomaly detection models (NN and LSTM) analyzed training points and applied five functions to extract a threshold and identify abnormal data values. While minimum, median, and mean functions raised a significant number of anomalies, the maximum function did not raise any anomalies. The function that seemed to work better was a variation of the Bollinger Bands, as it raised only a few anomalies. Unfortunately, the models did not take advantage of the anomaly data.

The dimensionality reduction model was used to reduce the number of features in half. While the reconstruction error was high when different data sources were reduced, separating the datasets by type and reducing them one by one resulted in drastically low reconstruction errors. This trick also worked for datasets with similar data types, such as stock market data and the cumulative and ranking datasets.

As for feature creation, a series of features were added to help models identify more accurately economic states. These new data included time-dependent features using numerical datasets—such as calendar features, cumulative and ranking features, and factor features—technical indicators using stock prices, and taxonomy features using textual datasets.

To accomplish the third objective, NLP models were trained to create taxonomic labels for the incomplete Bloomberg dataset—which helped include financial and economic articles—add new features—that were ultimately discarded as they added noise—and create word and event embeddings.

In the case of event embeddings, it was found that they are a good alternative to summarize news articles, and that they can be combined (averaging them) to obtain a summary of the economic state.

The fourth objective was not fulfilled. Data ablation was performed by training ML models—NN, LSTM, and RF—to predict the direction of stock prices, but repeating the experiment produced slightly different outcomes. These outcomes preserved the overall tendency—i.e., NN models obtained higher accuracy than LSTM and RF, and RF got higher ROI than the other models except for the last re-run where LSTM outperformed both of them—in all cases, it did not help identify a set of features that had a clear advantage over the others.

As for the last objective, a NN was used on the RL trading system to summarize high-dimensional data into low-dimensional states. Then, the agent learned to take suitable actions in the stock market based on these states and guided by loss functions that encourage diversification (low-risk) and wealth maximization. Two loss functions were used: a barrier method that limited the cash allocation to a maximum value of 35% and the same barrier method combined with a penalty method that punished the trading agent when the action did not add up to one.

With these loss functions, the agent could earn a higher ROI than using any other loss function. When these losses were tested in periods of economic turmoil, the agent could outperform the benchmark and other agents with a regular loss M function.

In particular, the loss M model performed worse than the other models by any financial metric. It got lower Sharpe, Sortino, and Omega ratios. In addition, it also suffered the greatest loss with a maximum drawdown of 82.82%, resulting in an annual return of 1.80%. Conversely, the diversified models got higher Sharpe, Sortino, and Omega ratios. Although they were affected by the 2008 financial crisis, their maximum drawdown was approximately 39%, resulting in an annual return of 9.60% (loss B) and 9.53% (loss BQ).

The agent developed a diversification strategy that allocated almost an equal amount of money to each asset in the S&P 500. Based on the analysis of portfolios, a diversification strategy demonstrated that it could earn higher ROI than other strategies and the benchmark. The key lesson is that a strategy that does not lose much capital during a financial crisis is better than any strategy that loses a significant amount of capital in adverse periods regardless of how well it performs during periods of economic growth. Using a diversification strategy, the agent could outperform the benchmark and other strategies and increased the initial capital 12-fold (compared to the benchmark's seven-fold increase). These results were similar for models that used 50 companies and those that used 425 companies. The main difference was that, during periods of economic growth, the diversified model earned slightly lower ROIs, and during periods of economic turmoil, the model earned slightly higher ROIs.

The aim of the thesis was achieved as it was possible to develop ML models that learn stock trading guided by a set of loss functions that promoted diversification and wealth maximization.

Although some results were encouraging, it is clear that work is still needed for ML models to compete or even beat human investors. In particular, it is difficult for ML models to adapt to specific changes automatically, model human behavior, and identify misleading information from economic agents—i.e., governments, companies and investors.

In terms of adaptability, continual learning research tries to create agents that can adapt to evolving environments. However, current ML models cannot adapt automatically to changes, such as new or updated regulations, disruptive technologies, new financial instruments, etc. This means that the models waste resources as they need to be retrained to consider these changes and miss investment opportunities because they fail to account for the effects of these changes in the market.

Concerning modeling human behavior, it becomes complicated as people are emotional beings with a thin layer of rationality and have a high variability level. On the one hand, emotions cause individuals to develop biases that cloud their judgment—e.g., loss aversion. On the other hand, variability among individuals produces people with different skill levels and diverse characteristics. Thus, it is challenging for ML algorithms to predict the precise behavior of one investor, let alone of all the investors.

As for misleading information, governments, companies, and investors embellish, manipulate, omit, or hide economic data to have an advantage over other economic agents [209, 210, 211, 212, 213, 214, 215].

If a model could solve these problems, a new era of stock trading would commence.

6.2 Suggestions of Future Work

This work can be extended in different ways.

Although algorithmic trading accounts for 60-73% of the U.S. equity trading, the rest of the trades are still made by at-times-irrational investors. For that reason, one option would be to focus on ML architectures that learn human behavior in competitive games using behavioral game theory foundations to predict stock price movements [216].

The ML model could exploit irrational behavior by identifying discrepancies in the market. For instance, finding cult stocks—those stocks that have "a sizable investor following despite the underlying company lacking when it comes to underlying fundamentals" [217]—and invest in them. For instance, as of December 2020, Tesla fits the cult stock definition with a P/E ratio of over 900, a forward P/E ratio of over 120, and a market capitalization of 460 billion dollars—more than the world's three largest carmakers combined [218].

Similarly, a second option would be to train ML architectures to predict algorithmic trading behavior. Despite trading algorithms being faster than ML ones, it might be possible to teach an ML model to think a few steps ahead to account for trading algorithms' actions.

A third option would be to use other features. Two alternatives are:

- Information about which companies are included or excluded when financial indexes (e.g., S&P 500) are rebalanced is a good predictor of stock price changes as large financial institutions tracking the index are forced to buy or sell these stocks.

- Influential people's comments, press releases, and tweets such as those about companies, other investors, countries' economic outlooks, cryptocurrencies, and regular currencies could be good indicators of stock price movements.

  For instance, Trump's tweets on Boeing, Carl Icahn's statement on Herbalife's 13% stake acquisition, or Elon Musk's tweet on Tesla's price being 'too high' caused a significant price movement in the stock prices of those companies [219, 220, 221].

A fourth option would be to include additional assets, namely derivative securities. By letting an ML agent use these instruments for risk management, it could, for instance, buy bonds in a country with high- interest rates and use a currency future—a contract to exchange a currency for another at a given exchange rate on a specific date—to eliminate the risk that the foreign currency depreciates and the bond gains get lost when they are exchange back to the local currency.

Another alternative would be to teach a system to identify and build directed acyclic graphs containing economic agents and elements' causal relationships. With this knowledge, a system could take more accurate and logical decisions. For example, suppose a virus of unknown origin caused a pandemic. In that case, the system could identify the companies that would benefit from it (i.e., technology businesses, multinational retail corporations, e-commerce businesses, antibacterial-items producers, etc.) or quickly adapt and enter another market.

Related to this work, given that the trading agent refused to trade actively, using any method that forces exploration, such as adding random noise, would help the agent select better actions. In addition, in this work, the RL model was not as advanced as current RL models. For that reason, exploring more powerful RL models could improve the trading agent and the strategies.

Finally, some of the results suggest that the RF model could be an interesting alternative if it were integrated into the RL framework because it obtained higher ROIs than the other models despite having lower accuracy.

This page is intentionally not left blank.

A Additional Concepts

The following sections contain additional Economics and ML concepts for the interested reader.

A.1 Additional Economics information

This section includes information regarding data biases, types of idiosyncratic risks, and why diversification is important.

A.1.1 Data Bias

Data bias represents a group of errors in which the sample distribution of specific data points drastically differs from that of the population distribution [222]. It provides incorrect or inaccurate information to ML models and makes them converge to incorrect solutions that can result in ML systems that discriminate unfairly based on race or gender [223, 224].

In [225, 222, 226], they are defined as follows:

- **Data-mining bias** appears when a statistically significant pattern is found due to repeating a search in the same data multiple times, making the results seem more significant than they are.

  There are two ways of eliminating this bias: 1) limiting the number of experiments performed with a data sample and 2) using an out-of-sample test.

  The first point reduces the possibility of obtaining a statistically significant result by chance.

  The second point ensures that, even if a statistically significant result is obtained, it is unlikely to remain that way by chance once the out-of-sample data is used.

When the result is statistically significant in this data, it is more likely that there is some economic significance.

In financial markets, this bias occurs with brokers or investors who excessively analyze historical data looking for statistically significant patterns or signals that are non-existent or turn out to be irrelevant.

- **Sample bias** manifests when sample data excludes elements representing the population distribution due to a flaw in the sample selection process.

- **Survivorship bias** is a type of sample bias in which, by sampling the current versions of market funds or indexes, those companies that went bankrupt are ignored. This type of bias can result in the "overestimation of historical performance and general attributes of a fund or market index" [227].

  For instance, using tech companies to analyze market performance during the 2020 health crisis as these companies are not representative of the economy and can perform their activities remotely.

- **Look-ahead bias** happens when data is processed in a way that future information is available beforehand.

  For instance, standardizing a price series of n elements using its average price gives future information to the time series' previous values. This error can be observed in the following time series: $(2, 1, 1, 1, 1, 1, 1, 1, 1, 20)$ with a mean value equal to 3. This value passes future information to the previous elements and indicates to a model—as it processes a new value each time step—that the last one will be a large number. Thus the model is biased, and it could take advantage of this future information by waiting and investing everything at the second-to-last moment.

  Similarly, using any macroeconomic report—see section B.4—without shifting the date one period forward provides future information given that macroeconomic data is reported with a one-period delay.

- **Time-period bias** appears when the time-period length or starting point influences the results. In terms of length, shorter periods might give more noteworthy results at the expense of lacking statistical significance. In comparison, more extended periods might be statistically significant at the expense of containing relevant results.

Regarding starting points, there might be a noticeable difference between two points. e.g., buying stocks at the bottom of a crisis than just before the crisis.

A.1.2 Idiosyncratic risks

As described in [228, 17, 15], idiosyncratic risk includes:

1. **Credit risk** refers to the probability of losing money due to companies or governments defaulting on their loans. The greater the uncertainty about the payment capacity, the higher the lenders' interest rate.

2. **Political risk** arises from the odds that a country's government instability, corruption, religious and ethnic tensions become a problem for investors.

3. **Financial risk** emerges from the possibility that a country cannot pay its debt obligations (called sovereign default) due to a high foreign debt-to-GDP ratio—such as Greece's debt crisis in 2009—or unstable exchange rate—such as Mexico during the Tequila crisis in 1994.

4. **Economic risk** appears from the prospect of a country's economic slowdown as measured by GDP growth rate, GDP per capita, inflation rate, among others.

5. **Foreign exchange risk** is caused by the likelihood of unfavorable exchange rate fluctuations.

   For example, a U.S. company that buys raw material on credit from an Italian company is exposed to the risk of dollar depreciation. If this were the case, the U.S. company would pay a higher price than if it had liquidated the total amount immediately; however, paying the full amount would limit its growth as it could only buy a limited amount of raw material.

6. **Arbitrage and short-selling risks** are the risks faced by investors when they perform any of these operations. There are two types of risk for these operations:

   (a) **Fundamental risk** refers to the risk that an event in the market moves the price differently from what the arbitrageur or short-seller expects.

   (b) **NTR** is the potential of an adverse price movement caused by other traders with different beliefs than the arbitrageur or short-seller. It is further subdivided into:

i. **Horizon risk** stems from the amount of time needed for a price to reach the expected target.

ii. **Margin risk** describes the risk of facing margin calls—demand for a partial payment to cover possible losses.

iii. **Short-covering risk** is the risk of involuntary liquidation due to the owners requesting their stocks back.

For instance, the Herbalife and Bill Ackman's controversy in which Ackman short-sold Herbalife stock thinking its price would decrease is an example of horizon risk and margin risk. Despite what appeared a correct fundamental analysis, less rational participants drove the price up for five years which caused margin calls. In 2018, he closed his position losing around 1 billion dollars [229].

A.1.3 Return-to-risk methods

In this subsection, three return-to-risk methods are described: the Sharpe ratio, the Omega ratio, and the Sortino ratio.

- **Sharpe ratio**: it is a measure that assumes normally distributed data and indicates how much investors are compensated for the risk taken but does not incorporate information about the correlation between the portfolio and other assets [22, 17, 24]. It has the following definition:

$$Sharpe\ Ratio = \frac{\mathbb{E}\left[R_p - R_{rf}\right]}{\sigma_p} \tag{A.1}$$

  where σ_p is the portfolio's standard deviation, R_p is the portfolio return, and R_{rf} refers to the risk-free rate [17, 24, 22].

  When comparing portfolios, the portfolio with the highest Sharpe ratio either earns the greatest return with the same level of risk or earns the same return with the lowest level of risk [22, 17].

  The main disadvantages of the Sharpe ratio are that, as a risk measure, the standard deviation considers price changes in both directions as equally risky and that this ratio assumes normally distributed stock market data [17, 24].

- **Omega ratio** is an alternative to the Sharpe ratio that considers the higher moments of the distribution. It is defined as the probability-weighted ratio of gains divided by the losses for a minimum acceptable return [230].

  It is calculated as follows:

$$\omega(\theta) = \frac{\int_{r}^{\infty} (1 - F(x))\, dx}{\int_{-\infty}^{r} F(x) dx} \tag{A.2}$$

  where F is the cumulative distribution function of the returns, and r refers to the minimum acceptable return that the investor considers a gain or a loss.

- **Sortino ratio** : it divides the portfolio's return by the portfolio's downside risk. Unlike the Sharpe ratio that assumes normally distributed data, the downside risk helps the Sortino ratio remove this requirement [22, 17]. The Sortino ratio is calculated as follows:

$$Sortino\ ratio = \frac{\mathbb{E}\left[R_p - \tau\right]}{\sigma_d} \tag{A.3}$$

  where R_p remains the same as before, τ is a target value, and σ_d is the standard deviation of negative asset returns relative to the target τ. This τ sometimes takes the value of the risk-free rate, R_{rf}, and even though the two versions differ only by a constant, portfolio rankings are likely to be the same [17].

A.1.4 Quantitative data

A.1.4.1 Factors

- **The momentum factor**: Also called Up Minus Down (UMD), the following formula is applied to calculate the momentum factor [35]:

$$UMD_t = \mathbb{E}[r_{TS}] - \mathbb{E}[r_{BS}]$$

  Where $\mathbb{E}[r_{TS}]$ is the average return of the top 30% of stocks, and $\mathbb{E}[r_{BS}]$ is the average return of the bottom 30% of stocks ordered by the momentum value.

Although ETFs use a different momentum calculation, in academia, it is usually computed as follows [231, 232, 35]:

$$Momentum_t = \frac{P_{t-1} - P_{t-11}}{P_{t-11}}$$

where P_{t-1} and P_{t-11} are the prices at month 1 and 11—with the first month being 0.

The formula excludes the most recent month to eliminate or reduce the inversion effect—in which winner stocks in one month tend to be loser stocks in the following month [232].

- **The size factor**: To calculate the size factor—also known as SMB—the following formula is used:

$$SMB = \mathbb{E}_{12}[r_{SCS}] - \mathbb{E}_{12}[r_{LCS}]$$

Where $\mathbb{E}_{12}[r_{SCS}]$ is the annual average return of small-cap stocks, and $\mathbb{E}_{12}[r_{LCS}]$ is the annual average return of large-cap stocks. Stocks in deciles 6-10 of the Center of Research in Security Prices (CRSP) index are classified as **small-cap stocks**, while those in deciles 1-5 are classified as **large-cap stocks** [35].

- **The profitability factor**: The following formula is used to compute the profitability factor—also known as Robust Minus Weak (RMW) [37]:

$$RMW = \mathbb{E}_{12}[r_{HPS}] - \mathbb{E}_{12}[r_{LPS}]$$

Where $\mathbb{E}_{12}[r_{HPS}]$ is the annual average return of the top 30% of high-profitability companies, and $\mathbb{E}_{12}[r_{LPS}]$ is the annual average return of the bottom 30% of low-profitability companies.

From which, profitability is computed as follows:

$$profitability = \frac{sales - COGS}{assets}$$

Where $COGS$ refers to the cost of goods sold.

- **The quality factor**: This factor is computed as follows:

$$RMW = r_{HQC} - r_{LQC}$$

Where r_{HQC} are the returns of high-quality companies, and r_{LQC} refers to the returns of low-quality companies.

- **The value factor**: To calculate the value factor—also referred to as HML (**H**igh BtM stocks **M**inus the return on **L**ow BtM stocks)—the following formula can be used [35]:

$$HML = \mathbb{E}_{12}[r_{VS}] - \mathbb{E}_{12}[r_{GS}]$$

Where $\mathbb{E}_{12}[r_{VS}]$ is the annual average return of value stocks, and $\mathbb{E}_{12}[r_{GS}]$ is the annual average return of growth stocks. Stocks within 30% of the highest BtM ratio are **value stocks**, and those within 30% of the lowest BtM ratio are **growth stocks**.

A.1.5 The importance of diversification

Diversification, also known as "don't put all your eggs in one basket," is a practical method to distribute risk among a group of assets. Mathematically, investing an equal amount on N i.i.d. assets with returns $R_1, \ldots, R_N$ generates the following portfolio return:

$$R_p = \frac{1}{N} \sum_{i=1}^{N} R_i \tag{A.4}$$

When N is sufficiently large, this return is approximately normally distributed. The variance of this portfolio, given the Central Limit Theorem, is:

$$var(R_p) = \frac{1}{N^2} \sum_{i-1}^{N} var(p_i) = \frac{1}{N^2} N \cdot \sigma^2 = \frac{\sigma^2}{N} \qquad \lim_{N \to \infty} \frac{\sigma^2}{N} = 0 \tag{A.5}$$

Where σ^2 is the assets' variance [15].

In other words, as the number of assets increases, the risk level of the portfolio decreases until it is eliminated. In practice, there is a non-diversifiable risk (see section 1) that is never removed given that assets are not i.i.d. [15, 147, 228].

A.2 ML information

This section includes more information regarding feature engineering and resampling methods.

A.2.1 Feature engineering

The following subsection contains additional information regarding preprocessing, feature selection, feature extraction, and feature construction.

A.2.1.1 Preprocessing

A.2.1.1.1 Time series data imputation

For time-series data, popular methods to handle missing data include LOCF, NOCB, linear interpolation, and spline interpolation [164, 162].

- **LOCF** is a procedure in which missing values are filled with the last available value before them.

$$x_{mis,t} = x_{obs,t-k} \tag{A.6}$$

  where the missing value at time t, $x_{mis,t}$, is replaced by the last available value before the missing value at time $t-k$, $x_{obs,t-k}$. When k is fixed, the interval is restricted, and an available value is not guaranteed—in which case the missing value remains, and other methods might need to be used to replace the missing value. Conversely, when k is variable, the entire interval until t is considered, and a value is guaranteed.

An example of this method can be observed in the first row of table A.1. The method takes the last available value before the gap of missing values (i.e., 2) to fill in the gap.

- **NOCB** is a method in which missing values are filled with the next available value after them.

$$x_{mis,t} = x_{obs,t+k} \tag{A.7}$$

where the missing value at time t, $x_{mis,t}$, is replaced by the next available value after the missing value at time $t+k$, $x_{obs,t+k}$. The same considerations for k mentioned in the previous item apply here, except that $t+k$ involves the values after t.

The second row of table A.1 shows an example of the NOCB method. The method takes the next available value after the gap of interrupted values (i.e., 5) to replace the missing values.

- **Linear interpolation** is an approach that replaces missing values using points from a line which is computed from the first available data point before and after a gap of missing values.

The equation to calculate the line is given by:

$$P_y = P_{y_0} + (P_x - P_{x_0})\left(\frac{P_{y_1} - P_{y_0}}{P_{x_1} - P_{x_0}}\right) \tag{A.8}$$

where $P_{x_0}, P_{x_1}, P_{y_0}, P_{y_1}$ are the x and y points used to create the line, and P_y is the value to interpolate.

In table A.1, the values 2 and 5 in the original dataset ($X = [1, 4, 5, 2, na, na, 5, 9]$) are taken to create the line $P_y = 2 + (P_x - 0)\left(\frac{5-2}{3-0}\right) = P_x + 2$. This line is then used to interpolate values 3 and 4 for $P_x = 1$ and $P_x = 2$, respectively.

- **Spline interpolation** is a technique that creates a piecewise polynomial—a polynomial defined by multiple subfunctions that are applied to different intervals in the domain—called spline to substitute the missing values in the dataset [44].

Intervals are delimited by knots and are used to fit different polynomial functions. The piecewise polynomial increases its flexibility the more knots it has, but this flexibility causes significant discontinuities. Splines use constraints to create a continuous curve and remedy these discontinuities [44].

A cubic spline with K knots can be modeled using:

$$P_{y_i} = \beta_0 + \beta_1 b_1(P_{x_i}) + \beta_2 b_2(P_{x_i}) + \cdots + \beta_{K+3} b_{K+3}(P_{x_i}) + \epsilon_i \tag{A.9}$$

where $b_1, b_2, \ldots, b_{K+3}$ are basis functions that are selected appropriately, $\beta_0, \ldots, \beta_{K+3}$ are the spline coefficients, and P_{y_i} is the value to interpolate.

For instance, in the last row of table A.1, the Python Pandas library is used to impute the missing values using the cubic spline interpolation.

Table A.1: Example of time series imputation methods.

Imputation Method	Original data	Imputed data
LOCF		$[1, 4, 5, 2, \mathbf{2}, \mathbf{2}, 5, 9]$
NOCB		$[1, 4, 5, 2, \mathbf{5}, \mathbf{5}, 5, 9]$
Linear interpolation	$[1, 4, 5, 2, na, na, 5, 9]$	$[1, 4, 5, 2, \mathbf{3}, \mathbf{4}, 5, 9]$
Cubic spline interpolation		$[1, 4, 5, 2, \mathbf{2.31}, \mathbf{2.32}, 5, 9]$

A.2.1.1.2 Numerical Transformation Methods

A.2.1.1.2.1 Power transformations: Box-Cox transformation

The normalized **Box-Cox transformation** is defined as [38, 160]:

$$x_{\text{Box-Cox}} = \begin{cases} \frac{x^{\lambda}-1}{\lambda \dot{x}^{\lambda-1}} & \text{if } x \neq 0 \\ \dot{x} \ln(x) & \text{if } x = 0 \end{cases} \tag{A.10}$$

where $\dot{x} = \exp \frac{1}{T} \sum_{t=1}^{T} \ln x_t$ is the geometric mean of the data points and works as a scale factor that allows comparing different models, and λ is a transformation parameter. To find the optimal λ, several values of this variable are used to transform the data and fit models. Then, the model that results in the smallest residual sum of squares indicates which λ is selected [38].

When $\lambda = 0$, the theoretical formulation (the Box-Cox transformation without the $\dot{x}$ scale factor) becomes the log transformation.

A.2.1.1.2.2 Power transformations: Yeo-Johnson transformation

The Yeo-Johnson transformation is defined as:

$$x_{\text{Yeo-Johnson}} = \begin{cases} \frac{(x+1)^{\lambda}-1}{\lambda} & \text{if } \lambda \neq 0 \text{ and } x \geq 0 \\ \ln{(x+1)} & \text{if } \lambda = 0 \text{ and } x \geq 0 \\ -\frac{(-x+1)^{2-\lambda}-1}{2-\lambda} & \text{if } \lambda \neq 2 \text{ and } x < 0 \\ -ln(-x+1) & \text{if } \lambda = 2 \text{ and } x < 0 \end{cases} \tag{A.11}$$

where λ is the same transformation parameter.

A.2.1.1.3 Textual Transformation Methods

Jurafsky [172] mentions three preprocessing steps: tokenizing words, normalizing word formats, and segmenting sentences.

A.2.1.1.3.1 Tokenizing words

Tokenization is the segmentation of text into words, morphemes, or subwords. The first element represents a single distinct meaningful element. The second, morpheme, refers to a unit of meaning which can be classified as a stem—a central word that provides the main meaning—or an affix—a word that provides additional meaning.

There are three approaches to tokenizing text:

1. Tokenize text using one or more special characters—usually a space and punctuation—to mark word boundaries.

2. Tokenize text from languages without a word-boundary character using a supervised neural sequence model trained with a hand-segmented training set. Tokenization of these types of languages is complex given that characters might combine to form more complex concepts. Thus, splitting a word incorrectly can change the meaning of the sentence [172].

3. Tokenize text by letting an algorithm define the token size. This is helpful because the algorithm can add an entry in the dictionary for multiword expressions, single words, affixes, stems, and subwords. The advantage of learning affixes, stems, and subwords is that they can help ML systems handle unknown words. For example, if the training set contains the words *smart* and *smartest*, but *smarter* only appears in the test set, using this approach, the system can still represent this unknown word combining affixes (-*er*) and stems (*smart*) in the dictionary [172].

A.2.1.1.3.2 Normalizing word formats

Given the constant growth of words in the English language, normalization can reduce vocabulary size and standardize words [172]. There are four techniques to normalize text:

- **Normalization**: it standardizes words or tokens to a single form. For instance, *the USA*, *the US*, *America*, and *the States*, all refer to the same idea. Thus, it can be normalized to *the USA*.
- **Case folding**: another type of standardization which transform upper cases to lower cases. However, this method is only applied when the advantages of generalization outweighs the loss of meaning.
- **Lemmatization**: it determines the root of words so that those that share the same root have similar behavior. For instance, searching on an online store for the word "dessert" or "desserts" should retrieve similar results. Ideally, lemmatization is performed using a morphological parser to extract stems and affixes; however, this type of parser can be complex.
- **Stemming**: it is a simpler method used to remove word-final affixes to simulate lemmatization. It is faster but not as accurate.

A.2.1.1.3.3 Segmenting sentences

This preprocessing step involves deciding based on rules or ML techniques, where to separate one string into sentences.

The symbols most commonly used for segmentation are punctuation characters such as periods, question marks, and exclamation points. However, in the case of periods, it can be unclear if they are acting as an abbreviation—e.g., *M.Sc.*—a sentence boundary marker or both—e.g., *Inc.* Abbreviation dictionaries built manually or using ML methods can be added to the preprocessing pipeline to help reduce ambiguous cases.

A.2.2 Resampling methods

Resampling methods involve the following steps [44, 201]:

- Repeatedly draw samples from a training set.
- Refit a model of interest on each sample.
- Study refitted model to obtain additional information about it.

Although these techniques are computationally expensive and ignore model complexity—a problem because high complexity might cause overfitting—they can be used with any loss function and nonlinear, adaptive fitting techniques to assess model performance [54, 202].

There are two resampling methods commonly used in ML: train/validation/test split and cross-validation.

- **Train/validation/test split** is a method in which data is separated into train, validation, and test groups, usually 70:15:15 or 70:20:10. The problem with this method is that the model might suffer from sampling bias if one group is assigned mostly one type of data [47].

- **Cross-Validation** is a set of methods that split data into two blocks. One block is used as a training set, while the other is used as a test set. These methods have lower sampling bias—which overestimates the test error rate—than the train/validation/test split and more stable test error rates [44, 54].

  Variations of the original concept are listed below:

  - **k-fold cross-validation** is a similar technique in which data is divided into blocks of roughly equal size called folds. The model is then trained using $k-1$ folds and tested in the held-out fold. This process repeats k times—until each fold has been used as the test set—resulting in k test error estimates [47, 44].

    The k-fold CV estimate is calculated by averaging these error values:

$$CV_k = \frac{1}{k}\sum_{i=1}^{k} Error_i \tag{A.12}$$

- **Leave-one-out cross-validation** (LOOCV) is a variant of k-fold CV in that $k = N$ where N is the number of data points.

  Although it also reduces the effect of sampling bias, LOOCV has two key disadvantages over k-fold CV: 1) it is computationally expensive because it requires N models to be trained. 2) Despite lower bias than the k-fold CV, it has a higher variance, which often gives less accurate estimates of the prediction error rate.

  The lower bias and higher variance are caused by the N training sets created by LOOCV. The use of $N - 1$ data points to train each model ensures high average prediction accuracy (low bias). Similarly, the N training sets contain highly correlated data points which cause the sum of correlated variables to increase with the amount of covariance (high variance) [54, 233].

  For this reason, k-fold CV is preferred as a good compromise [47, 44, 54].

  The LOOCV estimate for the test error is the average test error estimates:

$$CV_n = \frac{1}{n}\sum_{i=1}^{n} Error_i \tag{A.13}$$

- **Time series cross-validation** (TSCV) is another variant of k-fold CV used for time series. The key difference is that, given the correlation between data points in a time series, the train folds contain those values occurring before the points in the test fold. In other words, the number of folds increases over time from 2—one train fold and one test fold—to k—$k - 1$ train folds and one test fold.

  Similarly, the TSCV estimate is calculated as the k-fold CV. That is:

$$TSCV_k = \frac{1}{k}\sum_{i=1}^{k} Error_i \tag{A.14}$$

B Project Data

B.1 Stock market data

The following table lists the 425 companies used in this project.

Table B.1: The 425 companies used in the stock market dataset.

3M Company	Abbott Laboratories	AbbVie	Accenture plc	Activision Blizzard
Acuity Brands Inc	Adobe Systems Inc	Advance Auto Parts	AES Corp	Affiliated Managers Group Inc
AFLAC Inc	Agilent Technologies Inc	Air Products & Chemicals Inc	Akamai Technologies Inc	Alaska Air Group Inc
Albemarle Corp	Alexion Pharmaceuticals	Alliance Data Systems	Alliant Energy Corp	Allstate Corp
Alphabet Inc Class A	Altria Group Inc	Amazon.com Inc	Ameren Corp	American Airlines Group
American Electric Power	American Express Co	American International Group Inc.	American Tower Corp A	American Water Works Company Inc
Ameriprise Financial	Amerisource Bergen Corp	AMETEK Inc	Amgen Inc	Amphenol Corp
Analog Devices Inc.	Anthem Inc.	Aon plc	Apache Corporation	Apartment Investment & Mgmt

Apple Inc.	Applied Materials Inc	Archer-Daniels-Midland Co	Arthur J. Gallagher & Co.	Assurant Inc
AT&T Inc	Autodesk Inc	Automatic Data Processing	AutoNation Inc	AutoZone Inc
AvalonBay Communities Inc.	Avery Dennison Corp	Ball Corp	Bank of America Corp	Baxter International Inc.
Becton Dickinson	Bed Bath & Beyond	Best Buy Co. Inc.	BIOGEN IDEC Inc.	BlackRock
Block H&R	Boeing Company	BorgWarner	Boston Properties	Boston Scientific
Bristol-Myers Squibb	Broadcom	Cabot Oil & Gas	Campbell Soup	Capital One Financial
Cardinal Health Inc.	Carmax Inc	Carnival Corp.	Caterpillar Inc.	CBOE Holdings
Centene Corporation	CenterPoint Energy	CenturyLink Inc	Cerner	CF Industries Holdings Inc
Charles Schwab Corporation	Charter Communications	Chesapeake Energy	Chevron Corp.	Chipotle Mexican Grill
C. H. Robinson Worldwide	Chubb Limited	Church & Dwight	CIGNA Corp.	Cimarex Energy
Cincinnati Financial	Cintas Corporation	Cisco Systems	Citigroup Inc.	CME Group Inc.
CMS Energy	Coca Cola Company	Cognizant Technology Solutions	Colgate-Palmolive	Comcast Corp.
Comerica Inc.	ConAgra Foods Inc.	Concho Resources	Conoco Phillips	Consolidated Edison

Constellation Brands	Corning Inc.	Costco Co.	Crown Castle International Corp.	CSX Corp.
Cummins Inc.	CVS Health	Danaher Corp.	Darden Restaurants	DaVita Inc.
Deere & Co.	Delphi Automotive	Delta Air Lines	Dentsply Sirona	Devon Energy Corp.
Digital Realty Trust	Discover Financial Services	Discovery Comms-A	Discovery Comms-C	Dollar General
Dollar Tree	Dominion Resources	Dover Corp.	D. R. Horton	DTE Energy Co.
Duke Energy	Eastman Chemical	Eaton Corporation	eBay Inc.	Ecolab Inc.
Edison Int’l	Edwards Lifesciences	Electronic Arts	Emerson Electric Company	Entergy Corp.
EOG Resources	EQT Corporation	Equifax Inc.	Equinix	Equity Residential
Essex Property Trust Inc.	Estee Lauder Cos.	E*Trade	Eversource Energy	Exelon Corp.
Expedia Inc.	Expeditors Int’l	Extra Space Storage	Exxon Mobil Corp.	F5 Networks
Facebook Inc.	Fastenal Co	Federal Realty Investment Trust	FedEx Corporation	Fidelity National Information Services
Fifth Third Bancorp	FirstEnergy Corp	First Solar Inc	Fiserv Inc	FLIR Systems
Flowserve Corporation	Fluor Corp.	FMC Corporation	Foot Locker Inc	Ford Motor
Fortune Brands Home & Security	Franklin Resources	Freeport-McMoRan Inc.	Gap (The)	Garmin Ltd.

General Dynamics	General Electric	General Mills	General Motors	Genuine Parts
Gilead Sciences	Global Payments Inc	Goldman Sachs Group	Goodyear Tire & Rubber	Grainger (W.W.) Inc.
Halliburton Co.	Hanesbrands Inc	Harley-Davidson	Hartford Financial Svc.Gp.	Hasbro Inc.
HCA Holdings	Helmerich & Payne	Henry Schein	Hess Corporation	Hologic
Home Depot	Honeywell Int'l Inc.	Hormel Foods Corp.	Host Hotels & Resorts	HP Inc.
Humana Inc.	Huntington Bancshares	IDEXX Laboratories	Illinois Tool Works	Illumina Inc
Incyte	Ingersoll-Rand PLC	Intel Corp.	Intercontinental Exchange	International Business Machines
International Paper	Interpublic Group	Intl Flavors & Fragrances	Intuit Inc.	Intuitive Surgical Inc.
Invesco Ltd.	Iron Mountain Incorporated	J. B. Hunt Transport Services	JM Smucker	Johnson Controls International
Johnson & Johnson	JPMorgan Chase & Co.	Juniper Networks	Kansas City Southern	Kellogg Co.
KeyCorp	Kimberly-Clark	Kimco Realty	Kinder Morgan	KLA-Tencor Corp.
Kohl's Corp.	Kroger Co.	Laboratory Corp. of America Holding	Lam Research	L Brands Inc.
Leggett & Platt	Lennar Corp.	Lilly (Eli) & Co.	Lincoln National	LKQ Corporation
Lockheed Martin Corp.	Loews Corp.	Lowe's Cos.	LyondellBasell	Macerich

Macy's Inc.	Marathon Oil Corp.	Marathon Petroleum	Marriott Int'l.	Marsh & McLennan
Martin Marietta Materials	Masco Corp.	Mastercard Inc.	Mattel Inc.	McCormick & Co.
McDonald's Corp.	McKesson Corp.	Medtronic plc	Merck & Co.	MetLife Inc.
Mettler Toledo	Microchip Technology	Micron Technology	Microsoft Corp.	Mid-America Apartments
Mohawk Industries	Molson Coors Brewing Company	Mondelez International	Monster Beverage	Moody's Corp
Morgan Stanley	Motorola Solutions Inc.	M&T Bank Corp.	Murphy Oil	Mylan N.V.
NASDAQ OMX Group	NetApp	Netflix Inc.	Newell Brands	Newmont Mining Corp. (Hldg. Co.)
NextEra Energy	Nielsen Holdings	Nike	NiSource Inc.	Noble Energy Inc
Nordstrom	Norfolk Southern Corp.	Northern Trust Corp.	Northrop Grumman Corp.	NRG Energy
Nucor Corp.	Nvidia Corporation	Occidental Petroleum	Omnicom Group	ONEOK
Oracle Corp.	O'Reilly Automotive	PACCAR Inc.	Parker-Hannifin	Patterson Companies
Paychex Inc.	Pentair Ltd.	People's United Financial	PepsiCo Inc.	PerkinElmer
Perrigo	Pfizer Inc.	PG&E Corp.	Philip Morris International	Phillips 66
Pinnacle West Capital	Pioneer Natural Resources	PNC Financial Services	Polo Ralph Lauren Corp.	PPG Industries

PPL Corp.	Principal Financial Group	Procter & Gamble	Progressive Corp.	Prologis
Prudential Financial	Public Serv. Enterprise Inc.	Public Storage	Pulte Homes Inc.	PVH Corp.
QUALCOMM Inc.	Quanta Services Inc.	Quest Diagnostics	Range Resources Corp.	Realty Income Corporation
Regency Centers Corporation	Regeneron	Regions Financial Corp.	Republic Services Inc	Robert Half International
Rockwell Automation Inc.	Roper Industries	Ross Stores	Royal Caribbean Cruises Ltd	Ryder System
Salesforce-.com	Schlumberger Ltd.	Seagate Technology	Sealed Air	Sempra Energy
Sherwin-Williams	Signet Jewelers	Simon Property Group Inc	Skyworks Solutions	SL Green Realty
Snap-On Inc.	Southern Co.	Southwest Airlines	Southwestern Energy	Stanley Black & Decker
Starbucks Corp.	State Street Corp.	Stericycle Inc	Stryker Corp.	Sysco Corp.
Target Corp.	TE Connectivity Ltd.	Tegna Inc.	Teradata Corp.	Texas Instruments
Textron Inc.	The Bank of New York Mellon Corp.	The Clorox Company	The Cooper Companies	The Hershey Company
The Mosaic Company	Thermo Fisher Scientific	The Travelers Companies Inc.	The Walt Disney Company	Tiffany & Co.
TJX Companies Inc.	Tractor Supply Company	TransDigm Group	Transocean	TripAdvisor

T. Rowe Price Group	Tyson Foods	UDR Inc	Ulta Salon Cosmetics & Fragrance Inc	Under Armour
Union Pacific	United Continental Holdings	United Health Group Inc.	United Parcel Service	United Rentals Inc.
Universal Health Services Inc.	Unum Group	Urban Outfitters	U.S. Bancorp	Valero Energy
Varian Medical Systems	Ventas Inc	Verisign Inc.	Verisk Analytics	Verizon Communications
Vertex Pharmaceuticals Inc	V.F. Corp.	Visa Inc.	Vornado Realty Trust	Vulcan Materials
Walgreens Boots Alliance	Wal-Mart Stores	Waste Management Inc.	Waters Corporation	Wec Energy Group Inc
Wells Fargo	Western Digital	Western Union Co	Weyerhaeuser Corp.	Whirlpool Corp.
Williams Cos.	Willis Towers Watson	Wynn Resorts Ltd	Xcel Energy Inc	Xerox Corp.
Xilinx Inc	Xylem Inc.	Yum! Brands Inc	Zimmer Biomet Holdings	Zions Bancorp

B.2 Technical data

This section presents the technical indicators used in this work.

B.2.1 Indicators

- **MACD**: it uses two indicators, a longer EMA and a shorter EMA of closing prices to create a trend-following momentum oscillator that can be used to trigger buying and selling signals [234].

A MACD line is computed by subtracting the longer EMA from the shorter EMA, which is then used to calculate a signal line using a nine-day EMA of this MACD line. The signal line is used to indicate bullish or bearish crossovers. The former—a buy signal—occurs when the MACD line turns up and crosses the signal line, while the latter—a sell signal—appears when the MACD line turns down and crosses the signal line.

Usually, the longer EMA is computed using 26 days, the shorter EMA is calculated using 12 days, and the signal line is determined using nine days.

It is calculated as follows [234]:

$$\text{k-p SMA}_t = \frac{1}{t-k}\sum_{i=t-k}^{t} p_i$$
$$\text{k-p EMA}_t = \frac{2}{k+1}\cdot(p_t - \text{EMA}_{t-1}) + \text{EMA}_{t-1}$$
$$\text{MACD line} = (\text{12-day EMA}_t - \text{26-day EMA}_t)$$
$$\text{Signal line} = \text{9-day EMA}_t \text{ of MACD line}$$

where k-p refers to a k-period such as 9-, 12-, 26-days, and p_t indicates the price at time t.

The SMA_{t-1} is used to calculate the first EMA value. It replaces EMA_{t-1} in the formula. For instance, for a 15-days EMA, the previous 15 price values to time t are taken to compute SMA_0. Then, the EMA calculation becomes $\text{15-days-EMA}_1 = \frac{2}{k+1}\cdot(p_1 - \text{SMA}_0) + \text{SMA}_0$.

- **BB**: They are volatility bands placed above and below an SMA [235]. The volatility is measured by the standard deviation of the $n-$most recent prices.

  Common values used in the BB are 20 days for the SMA with bands situated two standard deviations above and below the SMA. A buy signal appears when the price touches the lower band, while a sell signal occurs when the price touches the upper band [235].

  It is computed using this formula [235]:

$$\text{Upper band} = \text{k-p SMA}_t + 2\cdot\sigma_{\text{k-p }p}$$

$$\text{Lower band} = \text{k-p SMA}_t - 2 \cdot \sigma_{\text{k-p } p}$$

where $k - p$ and p have the same meaning as before, and $\sigma_{\text{k-p}-p}$ is the volatility.

- **TUO**: It is a momentum oscillator that uses three different time frames to indicate buying and selling signals. Its most common values are 7, 14, and 28 days [236].

  It is defined as [236]:

$$\begin{aligned}
&\text{BP} = C_t - \min\left(L, C_{t-1}\right) \\
&\text{TR} = \max\left(H_t, C_{t-1}\right) - \min\left(L_t, C_{t-1}\right) \\
&\text{k-p BP} = \sum_{i=t-k}^{t} \text{BP}_i \\
&\text{k-p TR} = \sum_{i=t-k}^{t} \text{TR}_i \\
&UO = \frac{100}{4+2+1}\left[4 \cdot \frac{\text{7-p BP}}{\text{7-p TR}} + 2 \cdot \frac{\text{14-p BP}}{\text{14-p TR}} + \frac{\text{28-p BP}}{\text{28-p TR}}\right]
\end{aligned}$$

  where k-p has the same meaning as before, BP refers to buying pressure, H_t, L_t, C_t are the high, low, and close prices at time t, and TR means true range.

  A buy signal is raised when the TUO value is greater than 70, while a sell signal appears when the TUO value is lower than 30 [236].

- **TDI**: It is an indicator that tries to identify when a trend starts and when it ends. It has the following formula [237]:

$$\begin{aligned}
&\text{momentum}_k = p_t - p_{t-k} \\
&\text{AM}_k = \left|\sum_{i=t-k}^{t} \text{momentum}_k\right| \\
&\text{SAM}_{k,d} = \sum_{i=t-d}^{t} \text{AM}_k \\
&\text{TDI} = \text{AM}_{20} - \left(\text{SAM}_{20,40} - \text{SAM}_{20,20}\right) \\
&\text{DI}_{20,20} = \sum_{i=t-20}^{t} \text{momentum}_{20}
\end{aligned}$$

where p_t is the price at time t, AM_k refers to the absolute momentum over a k interval, $\mathrm{SAM}_{k,d}$ corresponds to the sum of absolute momentums, DI is the direction indicator, and t, k, and d refer to days, such that $d \leq k < t$.

TDI signals a trend if the value is positive and a consolidation if it is negative. The direction indicator is used to enter a long position if both TDI and DI are positive and a short position if TDI is positive and DI is negative [237].

- **Williams %R**: It is a momentum indicator that "reflects the level of the close relative to the highest high for the look-back period" [238]. Values ranging from 0 to -20 indicate overbuying, while values ranging from -80 to -100 denote overselling. This indicator can be calculated using [238]:

$$\%R = 100 \cdot \frac{H_{k-p} - C_t}{H_{k-p} - L_{k-p}}$$

where H_{k-p} is the maximum high price within a k-period, C_t corresponds to the close price at time t, and L_{k-p} refers to the minimum low price within a k-period.

- **ADX**: It is an indicator derived from the smoothed averages of the difference between the Plus Directional Indicator (+DI) and Minus Directional Indicator (-DI) that measures the trend's strength over time [239, 240]. The following values are required to calculate the ADX indicator [241, 239, 242, 243, 240, 244]:

$$\mathrm{DM}_t^+ = \begin{cases} H_t - H_{t-1}, & \text{if } (H_t - H_{t-1} > L_{t-1} - L_t)\, and\, (H_t - H_{t-1} \geq 0) \\ 0, & \text{otherwise} \end{cases}$$

$$\mathrm{DM}_t^- = \begin{cases} L_{t-1} - L_t, & \text{if } (H_t - H_{t-1} < L_{t-1} - L_t)\, and\, (L_{t-1} - L_t \geq 0) \\ 0, & \text{otherwise} \end{cases}$$

where H_t and L_t have the same definition as before. These values and the true range, TR, are then smoothed out as follows:

$$\mathrm{D_{f\text{-}k}} = \sum_{i=1}^{k} D_i$$

$$\mathrm{SM}_{\mathrm{D}_t,k} = \mathrm{D_{f\text{-}k}} - \frac{\sum_{i=t-k}^{t} D_i}{k-1} + D_t$$

where $\mathrm{D_{f\text{-}k}}$ represents the first k periods of D, and D corresponds to TR_t, DM_t^+, and DM_t^- values used to calculate their smoothed versions, i.e., ATR, $\mathrm{SM}_{\mathrm{DM}_t^+,k}$, and $\mathrm{SM}_{\mathrm{DM}_t^-,k}$, respectively.

With these values, the +DI and -DI are obtained using:

$$\mathrm{DI}_k^+ = \frac{\mathrm{SM}_{\mathrm{DM}_t^+,k}}{\mathrm{ATR}_t}$$

$$\mathrm{DI}_k^- = \frac{\mathrm{SM}_{\mathrm{DM}_t^+,k}}{\mathrm{ATR}_t}$$

For ADX, the values are smoothed over 14 periods, resulting in the ADX calculation:

$$\mathrm{DX}_t = 100 \cdot \frac{|\mathrm{DI}_{14}^+ - \mathrm{DI}_{14}^-|}{\mathrm{DI}_{14}^+ + \mathrm{DI}_{14}^-}$$

$$\mathrm{ADX}_t = \frac{13 \cdot \mathrm{ADX}_{t-1} + \mathrm{DX}_t}{14}$$

ADX values are divided in four groups: 0-25, 26-50, 51-75, and 76-100. These groups indicate an absent of trend, weak trend, strong trend, very strong trend, and extremely strong trend, respectively. A buy signal is triggered when +DI crosses above -DI and ADX is above 25 but fails when the price moves below the low value on the signal day. Conversely, a sell signal is raised when -DI crosses above +DI and ADX is above 25 but fails when the price moves above the high value on the signal day [244].

- **EVWMA**: It is a moving average that uses *volume* to define its period. It is calculated using the following formula [245]:

$$k - p\ EVWMA_t = \frac{(N - V_t) \cdot \text{k-p EVWMA}_{t-1} + V_t \cdot p_t}{N}$$

Where V indicates the volume at time t, p is the price at time t, and N refers to the volume period, which can be a constant value or a multiple of recent average volume—the N value influences the sensitivity of the averaging process [246].

A buy signal occurs when the price crosses above the EVWMA, while a sell signal appears when the price crosses below the EVWMA. A significant divergence between the price and the EVWMA value indicates an overbought or oversold market [245].

- **VWMA**: It is a moving average used to identify emerging, existing, and ending trends by calculating the volume-weighted moving average price [247, 248].

  It is defined as [247, 248]:

$$\text{k-p VWMA}_t = \frac{\sum_{i=t-k}^{t} w_i \cdot V_i}{\sum_{i=t-k}^{t} V_i}$$

  where k-p and V_i have the same meaning as before, and w_i represents the weights associated with a given volume.

  A bearish trend is confirmed when a VWMA crosses below an SMA, and the price moves below both moving averages. Conversely, a bullish trend is confirmed when a VWMA moves above an SMA and the price breaks above both moving averages [247, 248].

- **CCI**: It is an indicator used to identify starting and ending trends and to warn of extreme conditions [249].

  It is computed using the following formulas [249]:

$$\begin{aligned}
\text{TP}_t &= \frac{H_{k-p} + L_{k-p} + C_t}{3} \\
\text{MDev}_t &= \sum_{i=t-k}^{t} abs(\text{TP}_i - \text{k-p SMA}_{\text{TP}}) \\
\text{CCI}_t &= \frac{\text{TP} - \text{20-p SMA}_{TP}}{0.015 \cdot MDev_t}
\end{aligned}$$

  where k-p, H_{k-p}, L_{k-p}, and C_t have the same meaning as before, TP refers to the typical price, MDev_t is the mean deviation, and $\text{k-p SMA}_{\text{TP}}$ is the simple moving average of the typical price over a k-period.

  CCI values below -100 signal the start of a downtrend (overbought market), while values above $+100$ indicate the start of an uptrend (oversold market) [249].

- **RSI**: It is a momentum oscillator with values ranging between zero and 100 that determines the speed and change of price movement. It has this definition [250]:

$$\mu_{G,t} = \frac{(\mu_{G,t-1} \cdot (k-1)) + \text{Gain}_t}{k}$$
$$\mu_{L,t} = \frac{(\mu L, t-1 \cdot (k-1)) + \text{Loss}_t}{k}$$

$$RSI_t = 100 - \frac{100}{1 + \frac{\mu_{G,t}}{\mu_{L,t}}}$$

where $\mu_{G,t}$ refers to the average gain with the first value—$\mu_{G,0}$—initialized as the sum of gains over the past k-periods, $\mu_{L,t}$ is the average loss with an initial value $\mu_{L,0}$ equal to the sum of losses over the past k-periods, and k is the number of periods—with a typical value of 14.

RSI signals an overbought market when its value is above 70 and an oversold market when its value is below 30 [250].

B.3 Fundamental data

B.3.1 Return and profitability ratios

Ratios in this group include [153]:

- **Asset turnover**: It measures how effectively a company uses its total capital and allows comparing companies within the same industry.

$$Asset\ turnover = \frac{sales}{average\ total\ assets}$$

- **Operating cash flow margin**: It represents the fraction of cash flow from operating activities—those activities directly associated with core business's activities that generate revenue [251]—with respect to sales.

$$Operating\ cash\ flow\ margin = \frac{operating\ cash\ flow}{net\ sales}$$

- **EBIT margin**: It is the proportion of EBIT with respect to sales.

$$EBIT\ margin = \frac{EBIT}{sales}$$

- **EBITDA margin**: It is the proportion of EBITDA with respect to sales.

$$EBITDA\ margin = \frac{EBITDA}{sales}$$

- **Net profit margin**: It shows the proportion of earnings left from sales after removing production costs and taxes.

$$Net\ profit\ margin = \frac{net\ profit}{sales}$$

- **ROA**: It measures how effectively a company is using its assets to generate earnings [252].

$$ROA = \frac{net\ income}{total\ assets} = \frac{net\ profit + interest\ expenses}{average\ balance\ sheet\ total}$$

- **ROE**: It indicates the share of profit the company earned on the capital provided by shareholders.

$$ROE = \frac{net\ profit}{average\ shareholder's\ equity}$$

- **ROI**: It is the fraction of earnings relative to the investment.

  This ratio has different formulations, but the most general form is:

$$ROI = \frac{income}{investment}$$

  Income could be replaced with operating income, EBIT, net income, or net cash inflows; while investment could be substituted with total assets, working capital, stockholders' equity, or initial cash outlay [253].

This work uses the formulation from [153], i.e., $income = EBIT$ and $investment = total\ assets$.

- **ROCE**: It indicates how successfully a company invests its capital.

$$ROCE = \frac{ebit}{average\ capital\ employed}$$

Where *capital employed* is the total amount of capital used for the acquisition of profits.

$$capital\ employed = noncurrent\ assets + net\ working\ capital$$

B.3.2 Financial stability ratios

This group includes [153]:

- **CAPEX ratio**: It indicates the fraction of operating cash flow used in capital expenditure—long-term assets.

$$CAPEX\ ratio = \frac{capital\ expenditure}{operating\ cash\ flow}$$

- **Current assets to total assets ratio**: This ratio measures current assets as a proportion of total assets. It needs to be combined with other indicators because its meaning is variable. For instance, a high value could mean flexibility or large inventories, while a low value could mean that most assets are buildings and machinery.

$$Current\ assets\ to\ total\ assets\ ratio = \frac{current\ assets}{total\ assets}$$

- **Debt to EBITDA**: It is a proportion of the company's net debt to the EBITDA. It gauges if a company is earning enough to cover its net debt.

$$Debt\ to\ EBITDA = \frac{net\ debt}{EBITDA}$$

- **Dynamic gearing ratio**: This ratio indicates the debt repayment period in years.

$$Dynamic\ gearing = \frac{financial\ liabilities - cash\ and\ equivalents}{free\ cash\ flow}$$

- **Equity and long-term debt to fixed assets ratio**: It indicates how much non-current assets are protected by long-term debt and equity.

$$\begin{array}{c} Equity\ and\ long\text{-}term \\ debt\ to\ fixed\text{-}assets\ ratio \end{array} = \frac{shareholders'\ equity - long\text{-}term\ borrowed\ capital}{noncurrent\ assets}$$

- **Equity ratio**: It is the proportion of total assets funded by shareholders' equity.

$$Equity\ ratio = \frac{shareholders'\ equity}{balance\ total}$$

- **Equity to fixed assets ratio**: It measures how much noncurrent assets are backed up by equity.

$$Equity\ to\ fixed\ assets\ ratio = \frac{shareholders'\ equity}{noncurrent\ assets}$$

- **Gearing**: It shows what proportion of financial debt is funded by equity.

$$Gearing = \frac{financial\ liabilities - cash\ and\ equivalents}{shareholders'\ equity}$$

- **Goodwill ratio**: This ratio is the premium value paid by a company purchasing another as a proportion of shareholders' equity.

$$Goodwill\ ratio = \frac{goodwill}{shareholders'\ equity}$$

- **Productive asset investment ratio**: It measures capital expenditure as a proportion of depreciation expenses. If CAPEX is greater than depreciation, it means that the company is usually expanding.

$$Productive\ asset\ investment\ ratio = \frac{CAPEX}{depreciation\ expenses}$$

- **Noncurrent assets to total assets ratio**: It is noncurrent assets as a proportion of total assets.

$$Noncurrent\ assets\ to\ total\ assets\ ratio = \frac{noncurrent\ assets}{total\ assets}$$

B.3.3 Working capital management ratios

The ratios used in this group are [153]:

- **Days payable outstanding**: It indicates how long it takes a company to pay its suppliers.

$$Days\ payables\ outstanding = \frac{average\ accounts\ payables \cdot 360}{cost\ of\ sales}$$

- **Days sales outstanding**: It calculates how long it takes a company to collect bills from customers.

$$Days\ sales\ outstanding = \frac{average\ accounts\ receivables \cdot 360}{sales}$$

- **Cash ratio**: This ratio measures a company's liquidity using the proportion of liquid assets to current liability.

$$Cash\ ratio = \frac{cash\ on\ hand + short\text{-}term\ investments}{current\ liabilities}$$

- **Quick ratio**: This is similar to the cash ratio, but it adds receivables as liquid assets.

$$Quick\ ratio = \frac{cash\ on\ hand + \substack{short\text{-}term \\ investments} + receivables}{current\ liabilities}$$

- **Current ratio**: It indicates whether or not current assets are enough to cover current liabilities.

$$Current\ ratio = \frac{current\ assets}{current\ liabilities}$$

- **Inventory intensity**: It measures the ratio of inventory as a proportion of total assets.

$$Inventory\ intensity = \frac{raw\ materials\ and\ supplies}{total\ assets}$$

- **Inventory turnover**: It shows the rate at which a company replaces its inventory within a period.

$$Inventory\ turnover = \frac{cost\ of\ sales}{average\ inventory}$$

- **Inventory days**: It calculates the average number of days products remain in a company's stock.

$$Inventory\ days = \frac{360}{inventory\ turnover}$$

- **Cash conversion cycle**: It refers to the number of days cash flow from sales is tied up in inventory and other resources before it is converted into cash received [254].

$$Cash\ conversion\ cycle = days\ sales\ outstanding + inventory\ days - days\ payable\ outstanding$$

B.3.4 Valuation ratios

These ratios include:

- **Market capitalization**: It refers to the current value of a company.
- **Price to earnings ratio**: It calculates a company's market valuation as a proportion of its earnings.
- **Price to book ratio**: It computes the premium, i.e., a company's current market price to its book value.
- **Price to cash flow ratio**: It is a company's market value relative to its operating cash flow.

- **Price to free cash flow ratio**: It is considered a more exact measure than the price to cash flow ratio as it subtracts CAPEX from total operating cash flow.
- **Free cash flow yield**: It is the inverse of price to free cash flow.
- **Price to sales ratio**: It gauges a business valuation in terms of its sales.
- **Enterprise value (EV)**: It represents a company's market value.
- **EV to EBITDA**: It computes the enterprise value as a proportion of EBITDA.
- **EV to EBIT**: It calculates the enterprise value as a fraction of EBIT.
- **EV to free cash flow**: It is a company's market value divided by its free cash flow. It is less susceptible to fluctuations than other ratios.
- **EV to sales**: It is the enterprise value compared to sales.

B.3.5 Key bank values

Besides ROE and ROA defined in section B.3.1, they include the following ratios:

- **Earning assets ratio**: It computes earning assets—income-producing investments—as a proportion of total assets.
- **Loan loss provision**: It is the annual loan loss provision as a proportion of total loans.
- **Loans to deposits**: It calculates the fraction of loans covered by deposits.
- **Non-interest income**: It is the proportion of non-interest income to total revenue.
- **Operating expense ratio**: It computes the non-interest income as a fraction of total revenue.
- **Spread**: It is the quotient of net interest income divided by total loans. The net interest income is the difference between interest income and interest expense.
- **Tangible common equity ratio**: It represents the value of tangible common equity to total assets.

B.4 Macroeconomic data

Macroeconomic data is used as an attempt to measure the world's economic health. These datasets are published periodically by institutions, government agencies, and organizations. The following subsections describe the macroeconomic variables used in this thesis.

B.4.1 Consumer Sentiment Index

The University of Michigan publishes a report which contains three indexes: Consumer Sentiment Index, Current Conditions Index, and Consumer Expectations Index. Of them, the Consumer Sentiment Index is the most popular. This index measures consumers' outlook on national economic conditions and purchasing plans [181, 255].

The Consumer Sentiment Index is an index that measures consumers' outlook about national economic conditions and purchasing plans. Normal values fall between 70 and 100, while values below 80 and over 90 suggest poor economic performance and economic growth, respectively [181, 255].

B.4.2 Consumer Confidence Index

This index provides "an indication of future developments of households' consumption and saving, based upon answers regarding their expected financial situation, their sentiment about the general economic situation, unemployment and capability of savings" [256].

A value above 100 indicates a positive consumers' future economic outlook as indicated by their propensity to save less and spend more in the next year. Conversely, a value below 100 indicates a negative outlook as measured by their inclination to save more and consume less [256].

B.4.3 Delinquencies on All Loans and Leases, Commercial and Industrial, All Commercial Banks

Delinquent loans and leases refer to "those past due thirty days or more and still accruing interest as well as those in nonaccrual status" [257].

This time series is important as it indicates the value of those loans and leases that are not paid on time. During an economic downturn, as unemployment increases, people struggle to pay these loans, and this indicator increases. Conversely, during economic growth, this indicator decreases as more people pay their loans and leases [181].

B.4.4 Money supply

It is the currency and liquid instruments—those that can be exchanged quickly for legal tender—in a country's economy [258]. In most countries, central banks have four policy tools to achieve their monetary policy goals: the discount rate, reserve requirements, open market operations, and interest on reserves [259, 260, 258].

The **discount rate** is the interest rate central banks charge commercial banks for short-term loans. Lowering discount rates is considered expansionary as borrowing money from banks becomes cheaper, encouraging investment from businesses and spending from consumers. Conversely, increasing discount rates is considered contractionary as borrowing money from banks becomes more expensive, discouraging investment and consumption [259, 260, 258].

Reserve requirements refer to deposits that banks must hold in cash at the central bank or their vaults. When this requirement is high—a contractionary policy—banks have less cash available to loan, and when this requirement is low—an expansionary policy—banks have more cash to loan customers [259, 260, 258].

Open market operations involve buying and selling government securities from commercial banks and institutions. When the objective is to increase the money supply, a central bank purchases government securities, and when the intention is to reduce the money supply, it sells them [259, 260, 258].

Interest on reserves is a new tool—introduced by some central banks after the Financial Crisis of 2008—in which central banks pay interest on excess reserves held at a central bank. Excess reserves refer to the funds above the reserve requirements. Lowering this interest encourages banks to move the cash out of the central bank and borrow it at higher interest rates. Conversely, increasing this interest can discourage banks from moving the money out or even encourage banks to move money back in [259, 260, 258].

The money supply is an important macroeconomic variable because it affects interest rates, and prices of goods and services, i.e., saving, investment and consumption [259, 260, 258].

B.4.5 Unemployment Insurance Weekly Claims Report

The Department of Labor releases the Unemployment Insurance Weekly Claims Report. Investors use this report to look for early signs of economic growth or weakness. When the number of insurance claims is large (i.e., more than 400,000 weekly claims or greater than 3 million continued claims), the markets get agitated and perform poorly [261, 181].

B.4.6 30-Year Fixed-Rate Mortgage Average in the United States

As the most common mortgage rate chosen by first-time homeowners, this rate, combined with other indicators, is helpful to gauge three elements of the economy: housing affordability for families, residential construction for developers, and economic outlook for the government [262].

B.4.7 Interest Rates

Interest rates represent a percentage charged (or paid) on the total amount that a person borrows (or saves). A central bank's interest rate is one of the most critical rates as it influences other interest rates in the economy. Its effect is similar to those of the discount rate [263].

Central banks cannot directly influence interest rates; instead, they can use any monetary policy tools mentioned in the money supply subsection (B.4.4) [263].

B.4.8 TED spread

The TED spread is the difference between the three-month Treasury bill and the three-month LIBOR based on U.S. dollars [264]. The LIBOR (London Inter-Bank Offered Rate) is the benchmark interest rate used by major global banks to lend among them [265].

While the Treasury bill (T-bill) is the rate at which the U.S. government borrows money from investors, the LIBOR represents the rate that large banks pay investors for saving their money with them. Because the T-bills are considered risk-free, the TED spread is ultimately an indicator of credit risk [264].

During periods of economic downturn, the TED spread widens as defaults are more likely to occur. On the contrary, when the economy is growing, it narrows as default risk decreases.

B.4.9 Bank Prime Loan Rate

It refers to the interest rate charged by commercial banks to their customers with good credit. This rate is important because it is the minimum rate charged by commercial banks and used as the starting point for other interest rates [266].

B.4.10 Employment Situation Report

This report is one of the most important indicators for the markets released by the US Bureau of Labor Statistics (BLS). It contains information on average hourly compensation, the number of new jobs created, the total employment, and the unemployment rate [181, 267].

Only people who are 16 years old or older and civilians are included in the report and classified as employed, unemployed, or not in the labor force [181, 267].

Governments and markets use unemployment numbers as another piece to evaluate a nation's economic health [181, 267].

B.4.11 Consumer Price Index

The Consumer Price Index measures inflation by averaging the price changes of household goods and services over time using 36 months from 1982 to 1984 as its reference value of 100 [181].

In addition to price changes, inflation affects the economy due to the central bank's actions. For instance, the FED might increase interest rates to reduce inflation. This increment discourages investment because borrowing becomes more expensive and slows down economic growth and inflation.

For this reason, the markets pay attention to the CPI as an indicator of future government intervention—via fiscal policies—or FED intervention—via monetary policies.

B.4.12 Producer Price Index

The Producer Price Index measures the average change over time in the selling prices—from the first commercial transaction—received by domestic producers for their output [268].

It is used by the government and businesses to "make better-informed decisions for things like contract adjustment, tracking specific products and industries, forecasting, LIFO inventory valuation, and as an economic indicator" [269] as it is accurate, timely, objective, and relevant.

B.4.13 Median Consumer Price Index

The Cleveland Fed calculates the median CPI by looking at the prices of the goods and services published by the BLS, ranking the inflation rates of the components of the CPI, and selecting the one that falls in the middle of the ordered distribution [270, 181].

It is used as an alternative to the CPI because it excludes items whose prices vary frequently, such as food and energy, which prevents the CPI from measuring the underlying rate of inflation—the inflation likely to persist over medium-run horizons of several years—appropriately [270, 181].

B.4.14 Industrial Production and Capacity Utilization Report

This report contains information about manufacturing, mining, and electric and gas utilities. Investors use the report to have a comprehensive perspective of a cross-section of manufacturing industries, while economists use it to gauge early signs of inflationary pressure [181].

It comprises two indexes: the Industrial Production Index and the Capacity Utilization Index. The former measures how much is currently produced—industrial production volume as a percentage of the output in the base year—and strongly correlates with the GDP, while the latter indicates how much could be produced and the amount of the nation's current industrial capacity in use [181].

In terms of capacity utilization, the highest value is named fullest capacity and it is defined by the FED as sustainable maximum output—i.e., the output level a facility can sustain after factoring in downtime for maintenance and the continuous availability of labor and materials required for production [181].

B.4.15 Advance Report On Durable Goods

The Manufacturers' Shipments, Inventories, and Orders of the Advance Report On Durable Goods is a survey that provides monthly statistical data on economic conditions in the domestic manufacturing sector [271].

The advance report on durable goods contains "statistics on manufacturers' value of shipments, new orders (net of cancellations), end-of-month order backlog (unfilled orders), end-of-month total inventory, materials and supplies, work-in-process, and finished goods inventories (at current cost or market value)" [272] which indicate future business trends.

The government uses the report to make GDP estimates and develop fiscal and monetary policy, while investors, researchers and the media use it for analysis and economic forecasting [272].

B.4.16 New Residential Construction and New Residential Sale report

These reports are based on two surveys: the Building Permits Survey and the Survey of Construction.

The Building Permits Survey contains national, state, and local statistics on new privately-owned residential construction. This includes statistics on building permits, construction authorized but not started, housing starts, housing under construction, and housing completions [273].

The Survey of Construction provides national and regional statistics on "starts, completions, and characteristics of new, privately-owned single-family and multifamily housing units and on sales of new single-family houses" [274].

B.4.17 Retail Trade Report

The report produces "the most comprehensive data on retail economic activity in the United States" [275] and can be used to identify changes in consumer spending patterns and anticipate economic changes [276].

The Retail Trade Report allows investors to measure consumer demand for durable and nondurable goods. Durable goods are those that last at least three years, while nondurable goods are those that last less than three years [276].

B.4.18 GDP

The GDP is a measure of the country's economic output that considers the market value of final goods and services produced by labor and property located within a country. It includes details about personal income and spending, national income and spending, corporate spending and production, and inflation [277, 181].

The economic report is essential because it contains a country's annualized economic real growth during the last quarter. It affects the stock, currency, and commodity markets to different degrees. Moreover, it helps to identify periods of recession—defined as a period with two consecutive quarters of GDP decline— and depression—defined as any time the GDP falls by 10% or more [277, 181].

Despite its importance, there are two key factors to take into account when using this indicator [277, 181]:

1. BEA makes periodic revisions in which it can change definitions, concepts, and statistical methods used to collect and tabulate the data.
2. The GDP report can be revised up to three years after its release.

These factors are worthy of taking into account as they are sources of look-ahead bias.

B.4.19 Personal Income and Outlays Report

Personal income refers to the income people get from all sources such as wages and salaries, social security, and other government benefits; personal outlays refer to the sum of personal consumption expenditure of durable and nondurable goods and services, personal interest payments, and personal current transfer payments; and personal savings is what is left of personal income after subtracting personal spending [278, 279].

It is useful for the government and investors because it provides another perspective of economic performance [278, 279].

This page is intentionally not left blank.

C Extended Results

C.1 Data exploration results

C.1.1 Fundamental data

Table C.1: Descriptive statistics of a random features sample of balance sheet data.

Fields	Common Stock	Treasury Stock	Common Equity	Current Liabilities	Deferred Taxes
count	7.80e+01	8.10e+01	7.40e+01	8.10e+01	6.10e+01
mean	3.71e+06	8.42e+08	1.96e+10	5.66e+08	9.41e+08
std	2.48e+06	6.90e+08	1.30e+10	3.07e+08	7.51e+08
min	1.20e+06	0.00e+00	3.59e+09	2.34e+08	-1.29e+08
25%	2.40e+06	1.93e+08	8.93e+09	2.98e+08	3.64e+08
50%	2.60e+06	7.49e+08	1.66e+10	4.62e+08	7.47e+08
75%	5.60e+06	1.37e+09	2.67e+10	8.30e+08	1.62e+09
max	2.05e+07	2.59e+09	5.12e+10	1.29e+09	2.50e+09

Table C.2: Descriptive statistics of a random features sample of cash flow data.

Fields	Dec(Inc) Inventories	OS - Financing	Cash Dividends Paid	Extraordinary Items	OS - Investing
count	7.30e+01	3.90e+01	37.0	75.0	7.90e+01
mean	-1.81e+07	2.93e+07	0.0	0.0	2.19e+06
std	2.51e+07	5.61e+07	0.0	0.0	5.28e+06
min	-1.24e+08	0.00e+00	0.0	0.0	0.00e+00
25%	-2.39e+07	2.50e+06	0.0	0.0	0.00e+00
50%	-1.22e+07	1.40e+07	0.0	0.0	0.00e+00
75%	-2.50e+06	2.60e+07	0.0	0.0	1.10e+06
max	1.50e+07	2.69e+08	0.0	0.0	2.22e+07

Table C.3: Descriptive statistics of a random features sample of income statement data.

Fields	Investment Income	Cost of Goods Sold	Discontinued Operations	EPS Incl Extraordinary Items	Interest Expense Total
count	8.00e+01	8.10e+01	59.0	76.000000	8.00e+01
mean	4.40e+09	3.40e+08	0.0	0.53	1.76e+08
std	2.25e+09	6.28e+07	0.0	0.55	1.03e+08
min	1.28e+08	2.04e+08	0.0	-0.24	6.52e+07
25%	3.25e+09	2.94e+08	0.0	0.09	9.97e+07
50%	3.95e+09	3.38e+08	0.0	0.26	1.40e+08
75%	5.25e+09	3.91e+08	0.0	0.88	2.11e+08

Continued on next page

Table C.3: Descriptive statistics of a random features sample of income statement data.

Fields	Investment Income	Cost of Goods Sold	Discontinued Operations	EPS Incl Extraordinary Items	Interest Expense Total
max	1.68e+10	4.86e+08	0.0	2.13	4.42e+08

C.1.2 Exchange indexes data

Table C.4: Descriptive statistics of exchange indexes data.

Fields	Close	High	Low	Open	Volume
count	99394.00	99396.00	99396.00	99396.00	9.23e+04
mean	9699.77	9774.23	9622.23	9700.82	2.91e+08
std	12291.39	12386.35	12192.59	12291.81	5.13e+08
min	0.19	0.20	0.19	0.196800	0.00e+00
25%	2044.56	2057.64	2030.00	2043.85	4.26e+04
50%	5051.53	5086.81	5016.00	5050.87	6.98e+07
75%	11382.63	11466.06	11292.72	11383.36	3.02e+08
max	73517.00	73920.00	72534.00	73508.00	5.64e+09

C.1.3 Macroeconomic data

Table C.5: Descriptive statistics of a random features sample of macroeconomic data.

Fields	Finished goods inventories	Electronics and appliance stores	Gross private domestic investment...	Exports	Air transportation
count	341.00	3.23e+02	142.00	142.00	347.00
mean	8670.59	7.64e+09	94.76	85.85	509914.12
std	2349.37	1.38e+09	6.01	8.84	46337.36
min	4943.00	3.87e+09	80.53	72.08	439300.00
25%	6608.00	6.96e+09	90.28	79.10	472250.00
50%	8038.00	8.12e+09	95.97	81.81	509100.00
75%	11080.00	8.65e+09	99.69	94.64	530450.00
max	12685.00	9.56e+09	104.13	100.85	633600.00

C.1.4 Commodity data

Table C.6: Descriptive statistics of a random features sample of commodity data.

Fields	Coffee Arabica	Coffee robustas	Corn	Cotton	Shrimp
count	323.00	323.00	323.00	323.00	323.00
mean	121.40	74.73	137.70	73.04	12.31
std	51.63	30.80	63.66	24.68	2.21
min	52.02	21.26	66.93	37.22	7.94
25%	79.84	48.66	97.42	57.72	10.47
50%	116.38	76.24	110.12	68.21	11.90
75%	144.70	95.89	159.53	82.31	14.06

Continued on next page

Table C.6: Descriptive statistics of a random features sample of commodity data.

Fields	Coffee Arabica	Coffee robustas	Corn	Cotton	Shrimp
max	302.71	182.96	332.95	229.67	16.89

C.2 Datasets results

Table C.7: FCNN, LSTM and RF's average accuracies according to dataset type.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	FCNN	LSTM	RF
✓													**0.6838**	0.6764	0.6575
✓						✓							**0.6813**	0.6692	0.6497
✓					✓								**0.6820**	0.6670	0.6569
✓					✓	✓							**0.6815**	0.6706	0.6557
✓				✓									0.6814	**0.6817**	0.6678
✓				✓		✓							**0.6827**	0.6797	0.6604
✓				✓	✓								**0.6783**	**0.6783**	0.6617
✓				✓	✓	✓							**0.6842**	0.6831	0.6629
✓			✓										**0.6839**	0.6777	0.6480
✓			✓			✓							**0.6813**	0.6805	0.6540
✓			✓		✓								0.6771	**0.6829**	0.6532
✓			✓		✓	✓							**0.6832**	0.6735	0.6566
✓			✓	✓									0.6763	**0.6838**	0.6613
✓			✓	✓		✓							**0.6832**	0.6666	0.6614
✓			✓	✓	✓								**0.6825**	0.6761	0.6610
✓			✓	✓	✓	✓							**0.6810**	0.6736	0.6577
✓		✓											**0.6850**	0.6781	0.6676
✓		✓				✓							**0.6851**	0.6780	0.6581
✓		✓			✓								**0.6845**	0.6732	0.6666
✓		✓			✓	✓							**0.6848**	0.6725	0.6634

Continued on next page

Table C.7: FCNN, LSTM and RF's average accuracies according to dataset type.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	FCNN	LSTM	RF
✓		✓		✓									**0.6857**	0.6837	0.6638
✓		✓		✓		✓							**0.6854**	0.6731	0.6660
✓		✓		✓	✓								**0.6851**	0.6831	0.6629
✓		✓		✓	✓	✓							**0.6854**	0.6800	0.6661
✓		✓	✓										**0.6831**	0.6757	0.6611
✓		✓	✓			✓							**0.6847**	0.6791	0.6634
✓		✓	✓		✓								**0.6852**	0.6805	0.6577
✓		✓	✓		✓	✓							**0.6838**	0.6811	0.6577
✓		✓	✓	✓									**0.6856**	0.6761	0.6627
✓		✓	✓	✓		✓							**0.6842**	0.6764	0.6639
✓		✓	✓	✓	✓								**0.6848**	0.6827	0.6629
✓		✓	✓	✓	✓	✓							**0.6846**	0.6568	0.6676
✓	✓												**0.6825**	0.6690	0.6636
✓	✓					✓							**0.6827**	0.6697	0.6571
✓	✓				✓								**0.6839**	0.6586	0.6572
✓	✓				✓	✓							**0.6838**	0.6752	0.6632
✓	✓			✓									**0.6819**	0.6761	0.6605
✓	✓			✓		✓							**0.6838**	0.6696	0.6668
✓	✓			✓	✓								**0.6800**	**0.6800**	0.6675
✓	✓			✓	✓	✓							**0.6815**	0.6725	0.6582
✓	✓		✓										**0.6817**	0.6797	0.6516
✓	✓		✓			✓							**0.6836**	0.6707	0.6531
✓	✓		✓		✓								0.6824	**0.6832**	0.6511
✓	✓		✓		✓	✓							0.6782	**0.6791**	0.6566
✓	✓		✓	✓									0.6775	**0.6820**	0.6668
✓	✓		✓	✓		✓							**0.6841**	0.6814	0.6617
✓	✓		✓	✓	✓								0.6829	**0.6839**	0.6603

Continued on next page

Table C.7: FCNN, LSTM and RF's average accuracies according to dataset type.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	FCNN	LSTM	RF
✓	✓		✓	✓	✓	✓							**0.6838**	0.6712	0.6625
✓	✓	✓											**0.6847**	0.6755	0.6690
✓	✓	✓				✓							**0.6855**	0.6739	0.6631
✓	✓	✓			✓								**0.6843**	0.6676	0.6614
✓	✓	✓			✓	✓							**0.6843**	0.6711	0.6595
✓	✓	✓		✓									**0.6851**	0.6836	0.6670
✓	✓	✓		✓		✓							**0.6852**	0.6767	0.6662
✓	✓	✓		✓	✓								**0.6859**	0.6836	0.6631
✓	✓	✓		✓	✓	✓							**0.6850**	0.6801	0.6641
✓	✓	✓	✓										**0.6851**	0.6781	0.6666
✓	✓	✓	✓			✓							**0.6847**	0.6724	0.6623
✓	✓	✓	✓		✓								**0.6850**	0.6805	0.6606
✓	✓	✓	✓		✓	✓							0.6809	**0.6811**	0.6549
✓	✓	✓	✓	✓									**0.6854**	0.6820	0.6639
✓	✓	✓	✓	✓		✓							**0.6846**	0.6699	0.6674
✓	✓	✓	✓	✓	✓								**0.6850**	0.6833	0.6643
✓	✓	✓	✓	✓	✓	✓							0.6831	**0.6843**	0.6657
✓	✓	✓	✓	✓	✓	✓						✓	**0.6806**	0.6692	0.6650
✓	✓	✓	✓	✓	✓	✓					✓	✓	**0.6789**	0.6694	0.6610
✓	✓	✓	✓	✓	✓	✓				✓	✓	✓	**0.6827**	0.6690	0.6625
✓	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	**0.6837**	0.6569	0.6569
✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	**0.6769**	0.6692	0.6641
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**0.6790**	0.6748	0.6562

Table C.8: FCNN, LSTM and RF's average earnings according to dataset type.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	FCNN	LSTM	RF
✓													**1.2243**	1.2100	1.0733
✓						✓							**1.2000**	1.1596	0.9784
✓					✓								1.2064	**1.2238**	1.1836
✓					✓	✓							**1.2161**	1.1881	1.0235
✓				✓									1.2274	**1.2342**	1.1973
✓				✓		✓							**1.2244**	1.1653	1.1804
✓				✓	✓								**1.1916**	1.1798	1.1537
✓				✓	✓	✓							**1.2290**	1.2258	1.1856
✓			✓										**1.2250**	1.1799	1.0375
✓			✓			✓							1.2182	**1.2411**	0.9591
✓			✓		✓								1.1642	1.2177	**1.2329**
✓			✓		✓	✓							1.2145	1.1929	**1.2908**
✓			✓	✓									1.2091	**1.2249**	1.1932
✓			✓	✓		✓							**1.2240**	1.1761	1.1639
✓			✓	✓	✓								**1.2255**	1.1883	1.1953
✓			✓	✓	✓	✓							1.2117	**1.2828**	1.2546
✓		✓											**1.2359**	1.2039	1.1493
✓		✓				✓							**1.2499**	1.1776	0.9849
✓		✓			✓								**1.2362**	1.1825	1.0561
✓		✓			✓	✓							**1.2366**	1.1623	1.2138
✓		✓		✓									**1.2404**	1.2275	1.1961
✓		✓		✓		✓							**1.2409**	1.1850	1.1589
✓		✓		✓	✓								**1.2298**	1.2180	1.1250
✓		✓		✓	✓	✓							**1.2389**	1.1942	1.1288
✓		✓	✓										1.2126	1.1702	**1.2292**
✓		✓	✓			✓							**1.2326**	1.2070	1.1837
✓		✓	✓		✓								**1.2353**	1.2165	1.0922

Continued on next page

Table C.8: FCNN, LSTM and RF's average earnings according to dataset type.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	FCNN	LSTM	RF
✓		✓	✓		✓	✓							**1.2128**	1.2067	1.0152
✓		✓	✓	✓									**1.2358**	1.2348	1.2083
✓		✓	✓	✓		✓							**1.2363**	1.2043	1.1413
✓		✓	✓	✓	✓								1.2201	**1.2211**	1.2120
✓		✓	✓	✓	✓	✓							1.2387	**1.2713**	1.1913
✓	✓												**1.2233**	1.1666	1.1321
✓	✓					✓							**1.2152**	1.1667	1.0470
✓	✓				✓								**1.2242**	1.1465	1.0244
✓	✓				✓	✓							1.2222	**1.2378**	1.0745
✓	✓			✓									**1.2055**	1.2006	1.1034
✓	✓			✓		✓							**1.2254**	1.1989	1.0996
✓	✓			✓	✓								**1.1999**	1.1740	1.1890
✓	✓			✓	✓	✓							**1.2027**	1.1797	1.1198
✓	✓		✓										**1.2078**	1.1924	1.0952
✓	✓		✓			✓							**1.2252**	1.1824	1.0321
✓	✓		✓		✓								1.2144	**1.2246**	1.0240
✓	✓		✓		✓	✓							**1.2049**	1.1803	1.0158
✓	✓		✓	✓									**1.2155**	1.2147	1.2095
✓	✓		✓	✓		✓							**1.2246**	1.2026	1.1520
✓	✓		✓	✓	✓								1.2172	**1.2241**	1.1843
✓	✓		✓	✓	✓	✓							**1.2206**	1.2109	1.1661
✓	✓	✓											**1.2272**	1.1901	1.0273
✓	✓	✓				✓							**1.2443**	1.1678	0.9985
✓	✓	✓			✓								**1.2392**	1.1570	1.1526
✓	✓	✓			✓	✓							**1.2371**	1.1394	1.0348
✓	✓	✓		✓									**1.2332**	1.2283	1.1929
✓	✓	✓		✓		✓							1.2237	**1.2347**	1.1455

Continued on next page

Table C.8: FCNN, LSTM and RF's average earnings according to dataset type.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	FCNN	LSTM	RF
✓	✓	✓		✓	✓								**1.2404**	1.2201	1.1466
✓	✓	✓		✓	✓	✓							1.2309	1.2123	**1.2619**
✓	✓	✓	✓										**1.2367**	1.1878	1.2082
✓	✓	✓	✓			✓							**1.2383**	1.1868	1.0148
✓	✓	✓	✓		✓								**1.2284**	1.1932	1.1721
✓	✓	✓	✓		✓	✓							**1.2286**	1.2243	1.0782
✓	✓	✓	✓	✓									**1.2298**	1.2170	1.1770
✓	✓	✓	✓	✓		✓							**1.2314**	1.2149	1.1448
✓	✓	✓	✓	✓	✓								**1.2345**	1.2242	1.1768
✓	✓	✓	✓	✓	✓	✓							**1.2412**	1.2269	1.1690
✓	✓	✓	✓	✓	✓	✓						✓	1.2700	**1.2998**	1.1883
✓	✓	✓	✓	✓	✓	✓					✓	✓	1.2475	1.2746	**1.2978**
✓	✓	✓	✓	✓	✓	✓				✓	✓	✓	1.2539	**1.3030**	1.1678
✓	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	1.2298	**1.2346**	1.1002
✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	1.2580	**1.2644**	1.1456
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**1.2346**	1.2102	1.1219

C.3 Dimensionality reduction results

Table C.9: Reconstruction error in dimensionality reduction for financial data using four NN models.

Experiment	Transformation	Sigmoid	Tanh	ReLU	Swish
Reconstruction error original size (6 dims)	No	0.02495	**0.00145**	0.01569	0.00538
	MinMax	0.01452	**0.00126**	0.05841	0.00285
	Gaussian	0.59082	**0.17532**	0.39921	0.24531
	Quantile	0.06206	**0.00249**	0.04324	0.00500
Reconstruction error with 3 dims	No	0.02496	**0.00152**	0.01746	0.00552
	MinMax	0.01452	**0.00259**	0.07234	0.00538
	Gaussian	0.60281	**0.25280**	0.43688	0.27962
	Quantile	0.06211	**0.01325**	0.06629	0.01746
Reconstruction error with 1 dim	No	0.02496	**0.00466**	0.01971	0.00799
	MinMax	0.01452	0.01023	0.13202	**0.01007**
	Gaussian	0.63746	0.50097	0.57318	**0.48848**
	Quantile	0.06211	**0.04038**	0.13326	0.04060
Reconstruction error original size without vol. (4 dims)	No	0.00046	**0.00001**	0.00024	0.00007
	MinMax	0.00598	**0.00043**	0.07296	0.00121
	Gaussian	0.33209	**0.08601**	0.26353	0.14301
	Quantile	0.05555	**0.00200**	0.04586	0.00475

Table C.10: FCNN and RF's average accuracies with and without dimensionality reduction and log normalization datasets.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	D-Redux FCNN	No D-Redux FCNN	D-Redux RF	No D-Redux RF
✓													**0.6841**	0.6838	0.6628	0.6575
✓						✓							**0.6841**	0.6813	0.6564	0.6497
✓					✓								**0.6839**	0.6820	0.6527	0.6569
✓					✓	✓							**0.6838**	0.6815	0.6586	0.6557
✓				✓									**0.6841**	0.6814	0.6641	0.6678
✓				✓		✓							**0.6839**	0.6827	0.6676	0.6604
✓				✓	✓								**0.6803**	0.6783	0.6651	0.6617
✓				✓	✓	✓							0.6837	**0.6842**	0.6629	0.6629
✓			✓										**0.6839**	**0.6839**	0.6474	0.6480
✓			✓			✓							**0.6841**	0.6813	0.6493	0.6540
✓			✓		✓								**0.6841**	0.6771	0.6474	0.6532
✓			✓		✓	✓							**0.6841**	0.6832	0.6524	0.6566
✓			✓	✓									**0.6839**	0.6763	0.6657	0.6613
✓			✓	✓		✓							**0.6833**	0.6832	0.6624	0.6614
✓			✓	✓	✓								0.6768	**0.6825**	0.6633	0.6610
✓			✓	✓	✓	✓							0.6787	**0.6810**	0.6613	0.6577
✓		✓											0.6847	**0.6850**	0.6744	0.6676
✓		✓				✓							0.6845	**0.6851**	0.6636	0.6581
✓		✓			✓								**0.6848**	0.6845	0.6590	0.6666
✓		✓			✓	✓							**0.6850**	0.6848	0.6669	0.6634
✓		✓		✓									0.6851	**0.6857**	0.6706	0.6638
✓		✓		✓		✓							0.6851	**0.6854**	0.6682	0.6660
✓		✓		✓	✓								0.6843	**0.6851**	0.6669	0.6629
✓		✓		✓	✓	✓							0.6850	**0.6854**	0.6712	0.6661

Continued on next page

Table C.10: FCNN and RF's average accuracies with and without dimensionality reduction and log normalization datasets.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	D-Redux FCNN	No D-Redux FCNN	D-Redux RF	No D-Redux RF
✓		✓	✓										**0.6850**	0.6831	0.6715	0.6611
✓		✓	✓			✓							**0.6848**	0.6847	0.6639	0.6634
✓		✓	✓		✓								0.6837	**0.6852**	0.6585	0.6577
✓		✓	✓		✓	✓							**0.6848**	0.6838	0.6629	0.6577
✓		✓	✓	✓									0.6852	**0.6856**	0.6687	0.6627
✓		✓	✓	✓		✓							**0.6854**	0.6842	0.6670	0.6639
✓		✓	✓	✓	✓								**0.6857**	0.6848	0.6659	0.6629
✓		✓	✓	✓	✓	✓							**0.6852**	0.6846	0.6660	0.6676
✓	✓												**0.6841**	0.6825	0.6666	0.6636
✓	✓					✓							**0.6841**	0.6827	0.6610	0.6571
✓	✓				✓								0.6834	**0.6839**	0.6569	0.6572
✓	✓				✓	✓							**0.6838**	**0.6838**	0.6540	0.6632
✓	✓			✓									**0.6841**	0.6819	0.6618	0.6605
✓	✓			✓		✓							**0.6843**	0.6838	0.6624	0.6668
✓	✓			✓	✓								**0.6848**	0.6800	0.6655	0.6675
✓	✓			✓	✓	✓							**0.6842**	0.6815	0.6660	0.6582
✓	✓		✓										**0.6839**	0.6817	0.6582	0.6516
✓	✓		✓			✓							**0.6839**	0.6836	0.6608	0.6531
✓	✓		✓		✓								**0.6828**	0.6824	0.6492	0.6511
✓	✓		✓		✓	✓							**0.6836**	0.6782	0.6473	0.6566
✓	✓		✓	✓									**0.6791**	0.6775	0.6633	0.6668
✓	✓		✓	✓		✓							**0.6841**	**0.6841**	0.6637	0.6617
✓	✓		✓	✓	✓								0.6675	**0.6829**	0.6642	0.6603
✓	✓		✓	✓	✓	✓							0.6837	**0.6838**	0.6617	0.6625
✓	✓	✓											**0.6847**	**0.6847**	0.6758	0.6690

Continued on next page

Table C.10: FCNN and RF's average accuracies with and without dimensionality reduction and log normalization datasets.

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	D-Redux FCNN	No D-Redux FCNN	D-Redux RF	No D-Redux RF
✓	✓	✓				✓							0.6851	**0.6855**	0.6662	0.6631
✓	✓	✓			✓								**0.6856**	0.6843	0.6631	0.6614
✓	✓	✓			✓	✓							**0.6848**	0.6843	0.6600	0.6595
✓	✓	✓		✓									0.6848	**0.6851**	0.6698	0.6670
✓	✓	✓		✓		✓							0.6850	**0.6852**	0.6710	0.6662
✓	✓	✓		✓	✓								**0.6860**	0.6859	0.6662	0.6631
✓	✓	✓		✓	✓	✓							**0.6854**	0.6850	0.6671	0.6641
✓	✓	✓	✓										0.6848	**0.6851**	0.6704	0.6666
✓	✓	✓	✓			✓							**0.6847**	**0.6847**	0.6642	0.6623
✓	✓	✓	✓		✓								**0.6850**	**0.6850**	0.6610	0.6606
✓	✓	✓	✓		✓	✓							**0.6851**	0.6809	0.6631	0.6549
✓	✓	✓	✓	✓									0.6822	**0.6854**	0.6722	0.6639
✓	✓	✓	✓	✓		✓							**0.6847**	0.6846	0.6639	0.6674
✓	✓	✓	✓	✓	✓								**0.6854**	0.6850	0.6693	0.6643
✓	✓	✓	✓	✓	✓	✓							**0.6851**	0.6831	0.6696	0.6657
✓	✓	✓	✓	✓	✓	✓						✓	**0.6854**	0.6806	0.6706	0.6650
✓	✓	✓	✓	✓	✓	✓					✓	✓	**0.6857**	0.6789	0.6605	0.6610
✓	✓	✓	✓	✓	✓	✓				✓	✓	✓	**0.6839**	0.6827	0.6650	0.6625
✓	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	0.6735	**0.6837**	0.6627	0.6569
✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	**0.6848**	0.6769	0.6673	0.6641
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	**0.6824**	0.6790	0.6670	0.6562

Table C.11: FCNN and RF's average earnings with and without dimensionality reduction and log normalization datasets

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	D-Redux FCNN	No D-Redux FCNN	D-Redux RF	No D-Redux RF
✓													**1.2245**	1.2243	1.1481	1.0733
✓						✓							**1.2245**	1.2000	1.1123	0.9784
✓					✓								1.2250	1.2064	**1.2252**	1.1836
✓					✓	✓							**1.2257**	1.2161	1.1534	1.0235
✓				✓									1.2245	1.2274	**1.2365**	1.1973
✓				✓		✓							1.2205	**1.2244**	1.1021	1.1804
✓				✓	✓								**1.2100**	1.1916	1.1681	1.1537
✓				✓	✓	✓							1.2189	**1.2290**	1.1244	1.1856
✓			✓										**1.2250**	**1.2250**	1.2106	1.0375
✓			✓			✓							**1.2245**	1.2182	1.1573	0.9591
✓			✓		✓								1.2245	1.1642	1.1915	**1.2329**
✓			✓		✓	✓							1.2245	1.2145	1.1746	**1.2908**
✓			✓	✓									1.2122	1.2091	**1.2689**	1.1932
✓			✓	✓		✓							**1.2275**	1.2240	1.2101	1.1639
✓			✓	✓	✓								1.2226	**1.2255**	1.1622	1.1953
✓			✓	✓	✓	✓							1.2062	1.2117	1.1134	**1.2546**
✓		✓											1.2272	**1.2359**	1.2236	1.1493
✓		✓				✓							1.2316	**1.2499**	1.1497	0.9849
✓		✓			✓								1.2356	**1.2362**	1.1269	1.0561
✓		✓			✓	✓							1.2302	**1.2366**	1.1652	1.2138
✓		✓		✓									1.2290	1.2404	**1.2813**	1.1961
✓		✓		✓		✓							1.2262	**1.2409**	1.1665	1.1589
✓		✓		✓	✓								1.2340	1.2298	**1.2992**	1.1250
✓		✓		✓	✓	✓							1.2277	**1.2389**	1.1881	1.1288

Continued on next page

Table C.11: FCNN and RF's average earnings with and without dimensionality reduction and log normalization datasets

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	D-Redux FCNN	No D-Redux FCNN	D-Redux RF	No D-Redux RF
✓		✓	✓										1.2284	1.2126	**1.2384**	1.2292
✓		✓	✓			✓							1.2299	**1.2326**	1.1575	1.1837
✓		✓	✓		✓								1.2349	1.2353	**1.2471**	1.0922
✓		✓	✓		✓	✓							**1.2272**	1.2128	1.1461	1.0152
✓		✓	✓	✓									1.2305	1.2358	**1.2525**	1.2083
✓		✓	✓	✓		✓							1.2305	**1.2363**	1.1874	1.1413
✓		✓	✓	✓	✓								**1.2358**	1.2201	1.2080	1.2120
✓		✓	✓	✓	✓	✓							1.2299	**1.2387**	1.1598	1.1913
✓	✓												**1.2245**	1.2233	1.1573	1.1321
✓	✓					✓							**1.2245**	1.2152	1.1808	1.0470
✓	✓				✓								1.2195	1.2242	**1.2312**	1.0244
✓	✓				✓	✓							**1.2243**	1.2222	1.0735	1.0745
✓	✓			✓									1.2247	1.2055	**1.2636**	1.1034
✓	✓			✓		✓							1.2208	**1.2254**	1.1809	1.0996
✓	✓			✓	✓								**1.2276**	1.1999	1.1545	1.1890
✓	✓			✓	✓	✓							**1.2049**	1.2027	1.1606	1.1198
✓	✓		✓										1.2250	1.2078	**1.2631**	1.0952
✓	✓		✓			✓							1.2250	**1.2252**	1.1800	1.0321
✓	✓		✓		✓								**1.2233**	1.2144	1.1259	1.0240
✓	✓		✓		✓	✓							**1.2264**	1.2049	1.1007	1.0158
✓	✓		✓	✓									**1.2619**	1.2155	1.1859	1.2095
✓	✓		✓	✓		✓							1.2245	**1.2246**	1.1402	1.1520
✓	✓		✓	✓	✓								**1.3101**	1.2172	1.2229	1.1843
✓	✓		✓	✓	✓	✓							1.2167	1.2206	**1.2318**	1.1661

Continued on next page

Table C.11: FCNN and RF's average earnings with and without dimensionality reduction and log normalization datasets

Stock	Technical	Fundamental	Factors	Textual	Macro	Dates	Commodity	Currency	Indexes	Benchmark	Cumulative	Rankings	D-Redux FCNN	No D-Redux FCNN	D-Redux RF	No D-Redux RF
✓	✓	✓											**1.2287**	1.2272	1.2116	1.0273
✓	✓	✓				✓							1.2293	**1.2443**	1.2012	0.9985
✓	✓	✓			✓								1.2381	**1.2392**	1.1818	1.1526
✓	✓	✓			✓	✓							1.2360	**1.2371**	1.1313	1.0348
✓	✓	✓		✓									1.2257	1.2332	**1.2808**	1.1929
✓	✓	✓		✓		✓							1.2299	1.2237	**1.2562**	1.1455
✓	✓	✓		✓	✓								1.2347	**1.2404**	1.1847	1.1466
✓	✓	✓		✓	✓	✓							1.2324	1.2309	1.2034	**1.2619**
✓	✓	✓	✓										1.2272	1.2367	**1.2460**	1.2082
✓	✓	✓	✓			✓							1.2333	**1.2383**	1.1567	1.0148
✓	✓	✓	✓		✓								**1.2408**	1.2284	1.1883	1.1721
✓	✓	✓	✓		✓	✓							1.2262	**1.2286**	1.1908	1.0782
✓	✓	✓	✓	✓									**1.2566**	1.2298	1.2223	1.1770
✓	✓	✓	✓	✓		✓							1.2272	1.2314	**1.3860**	1.1448
✓	✓	✓	✓	✓	✓								**1.2468**	1.2345	1.1456	1.1768
✓	✓	✓	✓	✓	✓	✓							1.2312	**1.2412**	1.1968	1.1690
✓	✓	✓	✓	✓	✓	✓						✓	1.2354	**1.2700**	1.2000	1.1883
✓	✓	✓	✓	✓	✓	✓					✓	✓	1.2439	1.2475	1.1683	**1.2978**
✓	✓	✓	✓	✓	✓	✓				✓	✓	✓	1.2062	1.2539	**1.2732**	1.1678
✓	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	1.2228	**1.2298**	1.1621	1.1002
✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	1.2406	**1.2580**	1.2166	1.1456
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	1.2283	**1.2346**	1.2101	1.1219

C.4 ML hyperparameters results

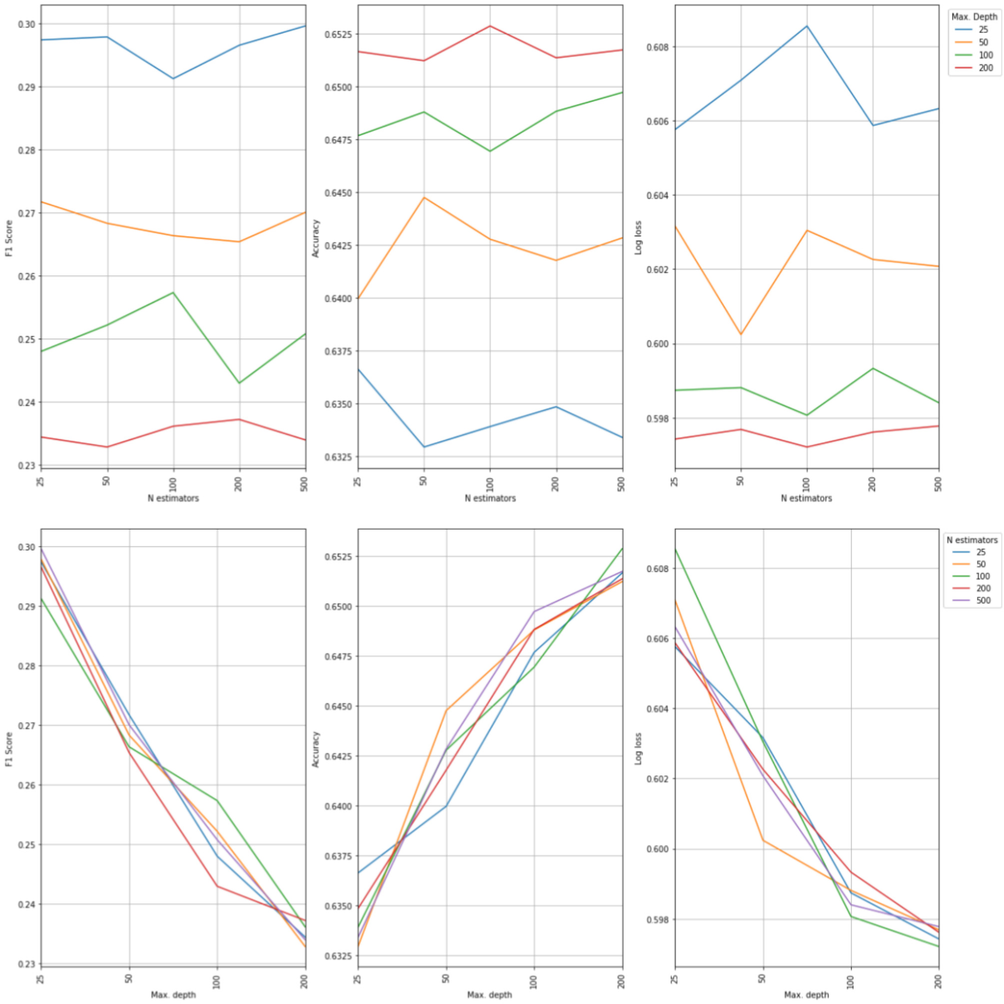


Figure C.1: RF results.

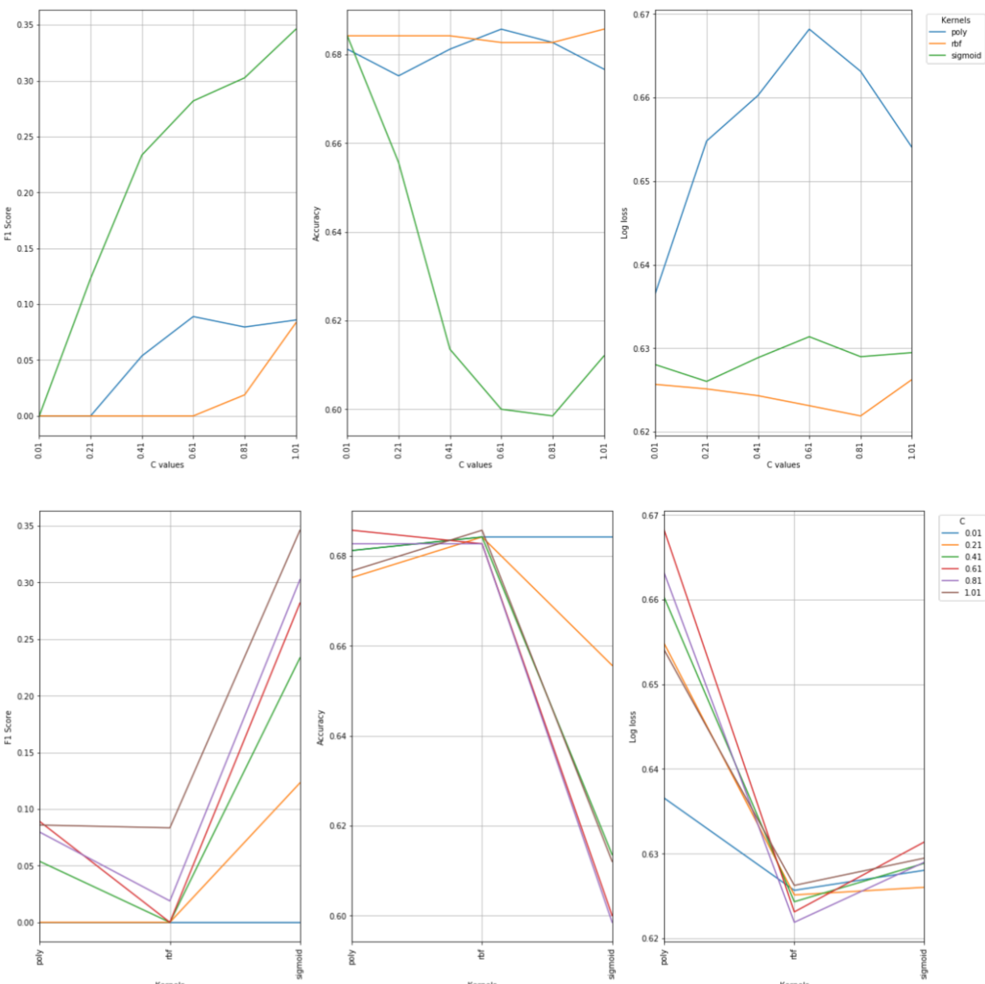


Figure C.2: SVM results.

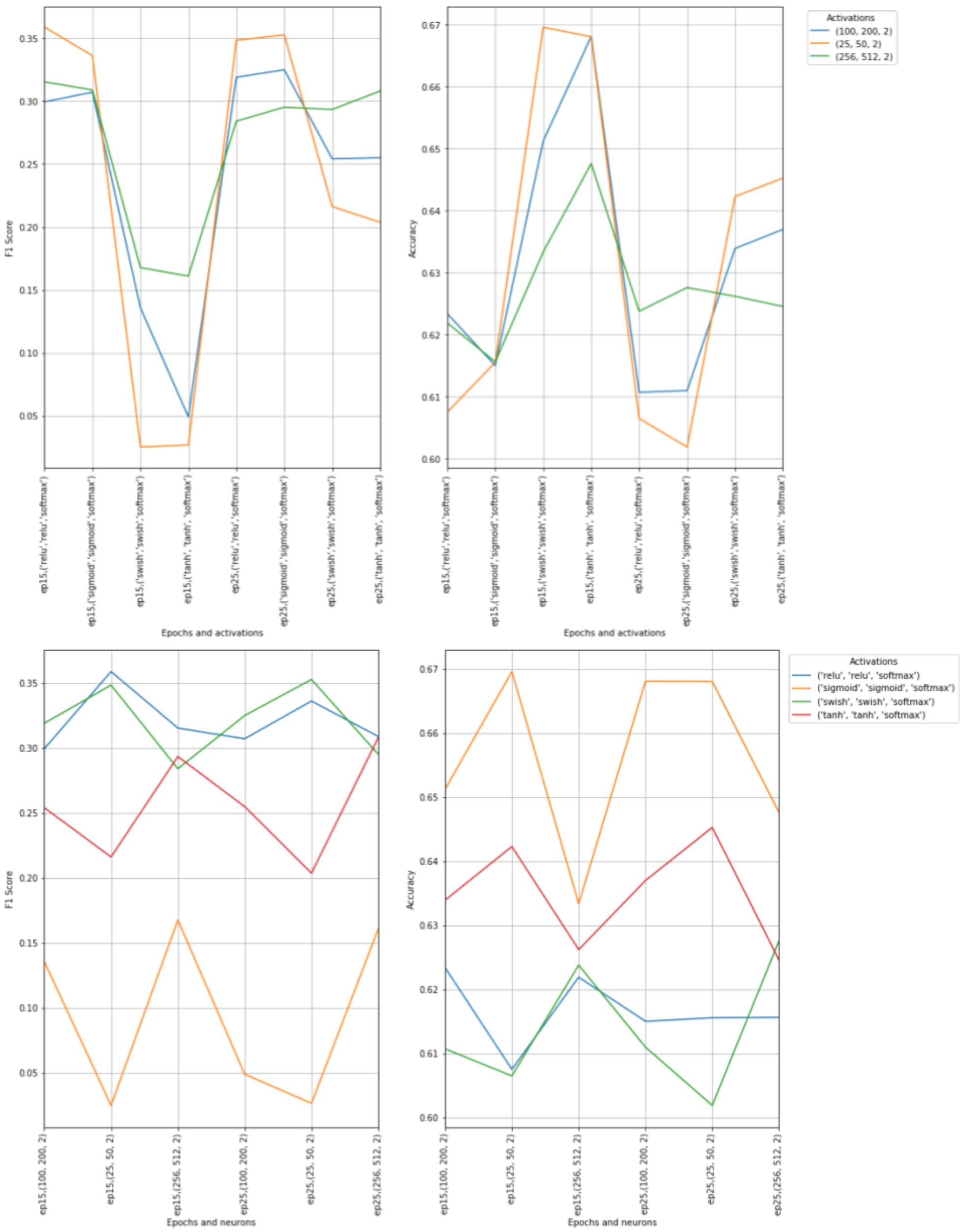


Figure C.3: NN results.

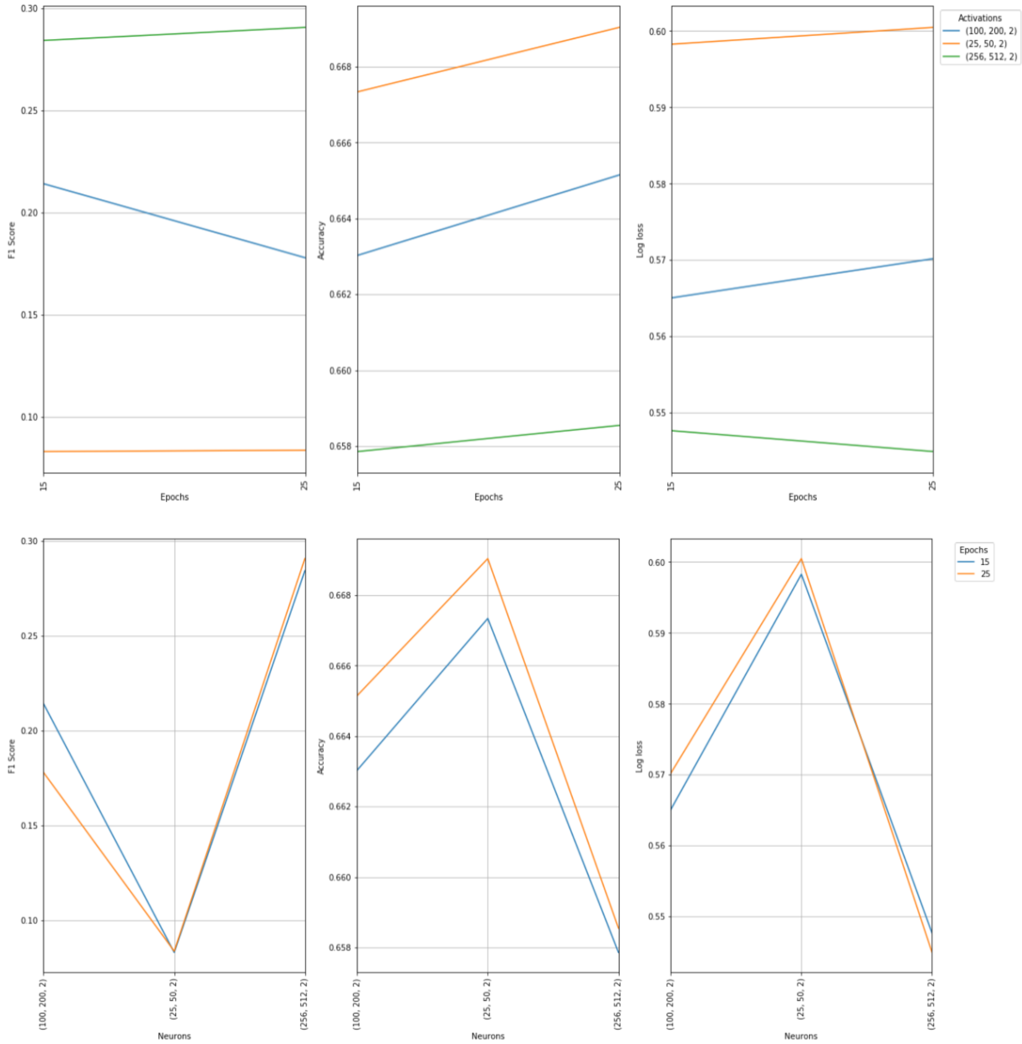


Figure C.4: LSTM results.

C.5 Analysis of news embedding clusters

Table C.12: Additional news extracts from one of the clusters using news embeddings.

Johnson & Johnson is in danger of once again becoming an also-ran in a market it pioneered. Having introduced in the United States a promising new drug-coated device to treat heart disease nearly a year before anyone else, the company has already lost its lead, analysts say.
Johnson & Johnson said yesterday that it had voluntarily recalled 300 Cypher heart stents after an internal audit of manufacturing records showed that six of them were not as thoroughly coated as specifications required.
Heart doctors at the Cleveland Clinic, one of the nation's largest centers for cardiac care, have voted unanimously to severely curtail or even ban the clinic's use of the Johnson & Johnson drug Natrecor.
Johnson & Johnson (JNJ), the health-care company beset by product recalls the last two years, said it was asking retailers to return about 12 million bottles of Motrin over concerns the painkiller may dissolve too slowly.
Johnson & Johnson (JNJ) rejected requests to offer patent rights on its HIV medicines to generic drug companies through a pool system over concern it could increase resistance to the drugs, the Financial Times said, citing an interview with head of pharmaceuticals, Paul Stoffel.
(Corrects Jones's present employer in 27th paragraph of story published Sept. 25.) Johnson & Johnson (JNJ) promoted illegal marketing of its antipsychotic drug Risperdal by paying physicians to give favorable speeches.
(Corrects school name in third paragraph of article published Nov. 13.) Johnson & Johnson (JNJ) hasn't given up on the ability of its experimental drug bapineuzumab to alter the course of Alzheimer's disease, even after two key studies failed to find a benefit, the company's neuroscience unit head said.

Johnson & Johnson (JNJ) anticipates submitting more than 10 new medicines for approval by regulators worldwide by 2017, including a modified version of a decades-old anesthetic that has been misused as a date-rape drug.

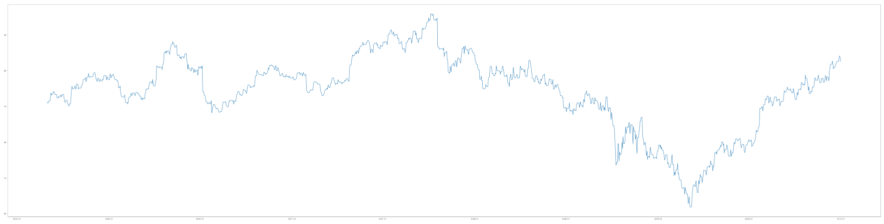
Pfizer Inc. said yesterday that it had awarded the assignment to create campaigns aimed at consumers for Relpax, an anti-migraine drug awaiting approval from the Food and Drug Administration, to D'Arcy Masius Benton & Bowles Communications in New York, part of the MacManus Group.

Alzheimer's disease patients who switched to Pfizer Inc. and Eisai Co. NPG45T8UG6CD were more likely to maintain or improve brain function.

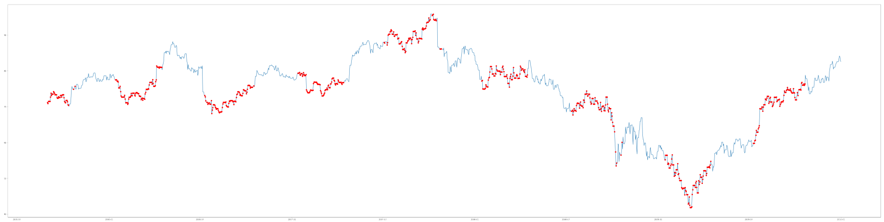
The world's biggest generic-drug company, may have sold 14 million pounds ($22.5 million) of its generic version of Pfizer Inc. (PFE)'s anti-cholesterol drug Lipitor last month, although the patent doesn't expire until next year, the Financial Times reported, citing unidentified industry analysts.

C.6 Anomaly detection

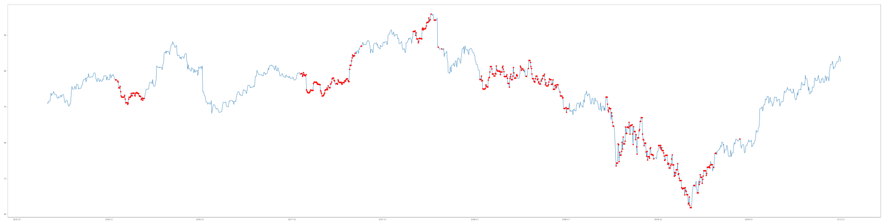
The following section contains additional images from the LSTM anomaly detector.



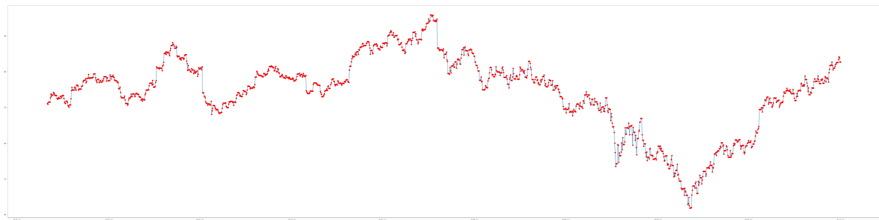
(a) **Max** function



(b) **Median** function



(c) **Mean** function



(a) **Min** function

Figure C.5: Anomaly detection signals raised by the LSTM model using the max, median, mean, and min functions.

Bibliography

[1] M. Kuhn, M. Schularick, and U. Steins, "Research: How the financial crisis drastically increased wealth inequality in the u.s.."

[2] D. Ariely, *Predictably irrational: The hidden forces that shape our decisions*. Harper Collins., 2009.

[3] J. Montier, *Behavioural finance : insights into irrational minds and markets*. West Sussex, England Hoboken, NJ: J. Wiley, 2002.

[4] M. Hancock, *Masters of nothing : how the crash will happen again unless we understand human nature*. London: Biteback, 2011.

[5] N. de Freitas, "Deep learning at oxford 2015. [online] accessed: July 3, 2017.."

[6] D. Silver, "Reinforcement learning course. [online] accessed: July 15, 2017.."

[7] R. Sutton, *Reinforcement learning : an introduction*. Cambridge, Massachusetts: The MIT Press, 2018.

[8] "Word definition: Context. [online] accessed: October 17, 2020.," October 2020.

[9] "Word definition: Causality. [online] accessed: October 17, 2020.," October 2020.

[10] J. Stiglitz, *Economics*. New York: W.W. Norton, 2006.

[11] Á. Cartea, *Algorithmic and high-frequency trading*. Cambridge, United Kingdom: Cambridge University Press, 2015.

[12] W. Kenton, "Limit order book. [online] accessed: October 17, 2020.."

[13] D. Vose, *Risk analysis : a quantitative guide*. Chichester, England Hoboken, NJ: Wiley, 2008.

[14] S. S. Investing, "Investment performance measurement: Return, volatility, alpha & beta. [online] accessed: January 23, 2021.."

[15] F. Fabozzi, *Robust portfolio optimization and management*. Hoboken, N.J: John Wiley, 2007.

[16] R. Slabotsky, "Inherent risk vs. residual risk explained in 90 seconds. [online] accessed: January 21, 2021.."

[17] J. Christopherson, *Portfolio performance measurement and benchmarking*. New York: McGraw-Hill, 2009.

[18] B. Efron, *Computer age statistical inference : algorithms, evidence, and data science*. New York, NY: Cambridge University Press, 2016.

[19] R. Shumway, *Time series analysis and its applications : with R examples*. Cham, Switzerland: Springer, 2017.

[20] J. Kuepper, "Volatility definition. [online] accessed: January 24, 2021.."

[21] W. Kenton, "Beta. [online] accessed: January 25, 2021.."

[22] WallStreetMojo, "Risk adjusted return. [online] accessed: January 18, 2021.."

[23] F. Russell, "Russell us indexes. [online] accessed: January 19, 2021.."

[24] W. F. Sharpe, "The sharpe ratio. [online] accessed: January 21, 2021.."

[25] Y. Varoufakis, "How capitalism works—and how it fails. [online] accessed: December 21, 2018.." `https://www.youtube.com/watch?v=QkmGfY9iY9Y`.

[26] B. Rockefeller, *Technical analysis for dummies*. Hoboken, NJ Chichester: Wiley John Wiley, 2004.

[27] R. Edwards, *Technical analysis of stock trends*. New Delhi: Vision, 2009.

[28] I. blog, "16 candlestick patterns every trader should know. [online] accessed: February 10, 2021.."

[29] M. Krantz, *Fundamental analysis for dummies*. Hoboken, N.J: John Wiley, 2010.

[30] B. Graham, *The intelligent investor : a book of practical counsel*. New York: HarperBusiness Essentials, 2003.

[31] W. Kenton, "Quantitative analysis (qa). [online] accessed: February 15, 2021.."

[32] S. Bell, *Quantitative finance for dummies*. Chichester, West Sussex, United Kingdom: John Wiley & Sons Inc, 2016.

[33] J. Chen and M. Boyle, "Factor investing. [online] accessed: February 15, 2021.."

[34] M. team., "Factor investing. [online] accessed: March 21, 2019.."

[35] A. Berkin, *Your complete guide to factor-based investing : the way smart money invests today*. St. Louis, MO: BAM ALLIANCE Press, 2016.

[36] M. K. Kim and D. A. Burnie, "The firm size effect and the economic cycle," *The Journal of financial research*, vol. 25, no. 1, pp. 111–124, 2002.

[37] R. Novy-Marx, "The other side of value: The gross profitability premium," *Journal of financial economics*, vol. 108, no. 1, pp. 1–28, 2013.

[38] S. Bisgaard, *Time series analysis and forecasting by example*. Hoboken, N.J: Wiley, 2011.

[39] R. Hyndman, *Forecasting : principles and practice*. Melbourne: OTexts, 2018.

[40] E. Dagum, *Seasonal adjustment methods and real time trend-cycle estimation*. Switzerland: Springer, 2016.

[41] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[42] K. Murphy, *Machine learning : a probabilistic perspective*. Cambridge, Mass: MIT Press, 2012.

[43] S. Department, "Lesson 10: Support vector machines."

[44] G. James, *An introduction to statistical learning : with applications in R*. New York, NY: Springer, 2013.

[45] S. Sayad, "Support vector machine - regression (svr). [online] accessed: December 30, 2020.."

[46] K. Weinberger, "Kernels," 2018.

[47] C. Bishop, *Pattern recognition and machine learning*. New York: Springer, 2006.

[48] G. Zhang, "What is the kernel trick? why is it important? [online] accessed: December 30, 2020.."

[49] S. Sayad, "Support vector machine - classification (svm). [online] accessed: December 30, 2020.."

[50] A. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, pp. 199–222, 2004.

[51] R. Davis, "Svm and boosting."

[52] S. documentation., "Decision trees. [online] accessed: January 28, 2021.."

[53] R. Nisbet, *Handbook of statistical analysis and data mining applications*. London, United Kingdom: Academic Press, an imprint of Elsevier, 2018.

[54] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference and prediction*. Springer, 2 ed., 2009.

[55] D. Britz, "Recurrent neural networks tutorial, part 1 – introduction to rnns. [online] accessed: October 17, 2020.."

[56] A. Amidi and S. Amidi, "Recurrent neural networks cheatsheet. [online] accessed: October 17, 2020.."

[57] A. Patel, *Hands-on unsupervised learning using Python : how to build applied machine learning solutions from unlabeled data*. Sebastopol, CA: O'Reilly Media, 2019.

[58] S. Li, "Anomaly detection for dummies. [online] accessed: October 20, 2020.."

[59] Sciforce, "Anomaly detection, a key task for ai and machine learning, explained. [online] accessed: October 20, 2020.."

[60] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller, "Playing atari with deep reinforcement learning," *CoRR*, vol. abs/1312.5602, 2013.

[61] "Playing atari with deep reinforcement learning. [online] accessed: December 29, 2018.." `https://deepmind.com/research/publications/playing-atari-deep-reinforcement-learning/`.

[62] J. Nocedal, *Numerical optimization*. New York: Springer, 1999.

[63] C. Lagarde, "The interconnected global economy: Challenges and opportunities for the united states—and the world. [online] accessed: November 14, 2020.."

[64] Y. Xu and J. Corbett, "Using network method to measure financial interconnection," Working Paper 26499, National Bureau of Economic Research, November 2019.

[65] Z. Ftiti, K. Guesmi, and I. Abid, "Oil price and stock market co-movement: What can we learn from time-scale approaches?," *International Review of Financial Analysis*, vol. 46, pp. 266–280, 2016.

[66] S. Shen, H. Jiang, and T. Zhang, "Stock market forecasting using machine learning algorithms," 2012.

[67] A. Chatrath, H. Miao, S. Ramchander, and S. Villupuram, "Currency jumps, co-jumps and the role of macro news," *Journal of International Money and Finance*, vol. 40, pp. 42 – 62, 2014.

[68] J. Lahaye, S. Laurent, and C. J. Neely, "Jumps, cojumps and macro announcements," *Journal of Applied Econometrics*, vol. 26, no. 6, pp. 893–921, 2011.

[69] F. Allen and D. Gale, "Financial contagion," *The Journal of political economy*, vol. 108, no. 1, pp. 1–33, 2000.

[70] M. Elliott, B. Golub, and M. O. Jackson, "Financial networks and contagion," *The American economic review*, vol. 104, no. 10, pp. 3115–3153, 2014.

[71] C. Camerer, *Behavioral Game Theory : Experiments in Strategic Interaction*. Princeton: Princeton University Press, 2011.

[72] W. Forbes, *Behavioural finance*. New York: Wiley, 2009.

[73] R. Shiller, *Irrational exuberance*. Princeton, N.J: Princeton University Press, 2000.

[74] J. Chen, "Efficient market hypothesis. [online] accessed: December 21, 2018.."

[75] A. Dixit, *Games of strategy*. New York: Norton, 1999.

[76] W. F. M. D. Bondt and R. Thaler, "Does the stock market overreact ?," in *The Journal of Finance*, vol. 40 of *Papers and Proceedings of the Forty-Third Annual Meeting American Finance Association*, pp. 793–805, Wiley, Wiley, 2007.

[77] K. A. Froot and E. M. Dabora, "How are stock prices affected by the location of trade?," *Capital Markets: Market Efficiency*, 1998.

[78] O. A. Lamont and R. Thaler, "Can the market add and subtract? mispricing in tech stock carveâĂŘouts," *Journal of Political Economy*, vol. 111, pp. 227 – 268, 2003.

[79] W. S. Prep., “Bloomberg vs. capital iq vs. factset vs. thomson reuters eikon. [online] accessed: August 25, 2021..”

[80] E. Beyaz, F. Tekiner, X.-j. Zeng, and J. Keane, “Comparing technical and fundamental indicators in stock price forecasting,” in *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 1607–1613, 2018.

[81] C. Evans, K. Pappas, and F. Xhafa, “Utilizing artificial neural networks and genetic algorithms to build an algo-trading model for intra-day foreign exchange speculation,” *Mathematical and computer modelling*, vol. 58, no. 5-6, pp. 1249–1266, 2013.

[82] C. Hargreaves and Y. Hao, “Does the use of technical and fundamental analysis improve stock choice? : A data mining approach applied to the australian stock market,” in *2012 International Conference on Statistics in Science, Business and Engineering (ICSSBE)*, pp. 1–6, Sept 2012.

[83] H. Yu, G. V. Nartea, C. Gan, and L. J. Yao, “Predictive ability and profitability of simple technical trading rules: Recent evidence from southeast asian stock markets,” *International Review of Economics & Finance*, vol. 25, pp. 356 – 371, 2013.

[84] M. J. de Souza, D. G. F. Ramos, M. G. Pena, V. A. Sobreiro, and H. Kimura, “Examination of the profitability of technical analysis based on moving average strategies in brics,” *Financial Innovation*, vol. 4, no. 1, p. 3, 2018.

[85] E. Jabbur, E. Silva, D. Castilho, A. M. Pereira, and H. Brandão, “Design and evaluation of automatic agents for stock market intraday trading,” *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, vol. 3, pp. 396–403, 2014.

[86] M. L. de Prado, “A data science solution to the multiple-testing crisis in financial research,” 2019.

[87] R. Shiller, *Market volatility*. Cambridge, Mass: MIT Press, 1989.

[88] E. Jurczenko, *Factor investing : from traditional to alternative risk premia*. London, UK Kidlington, Oxford, UK: ISTE Press Elsevier, 2017.

[89] A. Ilmanen and J. Kizer, "The death of diversification has been greatlyexaggerated," *The Journal of Portfolio Management*, vol. 38, pp. 15 – 27, 2012.

[90] J. Lee, "Is value dead? debate rages among quant greats from fama to aqr. [online] accessed: December 11, 2020.."

[91] L. ZHANG, "The value premium," *The Journal of finance (New York)*, vol. 60, no. 1, pp. 67–103, 2005.

[92] M. Yogo, "A consumption-based explanation of expected stock returns," *The Journal of finance (New York)*, vol. 61, no. 2, pp. 539–580, 2006.

[93] A. J. Black, B. Mao, and D. G. McMillan, "The value premium and economic activity: Long-run evidence from the united states," *Journal of asset management*, vol. 10, no. 5, pp. 305–317, 2009.

[94] S. B. Bulmash and G. W. Trivoli, "Time–lagged interactions between stocks prices and selected economic variables," 1991.

[95] G. R. Jensen and J. M. Mercer, "Monetary policy and the crossâĂŘsection of expected stock returns," *The Journal of financial research*, vol. 25, no. 1, pp. 125–139, 2002.

[96] C. Asness, A. Frazzini, R. Israel, T. J. Moskowitz, and L. H. Pedersen, "Size matters, if you control your junk," *Journal of Financial Economics*, vol. 129, no. 3, pp. 479 – 509, 2018.

[97] D. Bortolotti, "Understanding the momentum factor. [online] accessed: December 11, 2020.."

[98] C. Asness, A. Frazzini, R. Israel, and T. J. Moskowitz, "Fact, fiction and momentum investing," *The Journal of Portfolio Management*, vol. Special 40th Anniversary Issue, 2014.

[99] E. Fama and K. French, "Size, value, and momentum in international stock returns," *Journal of Financial Economics*, vol. 105, pp. 457–472, 2012.

[100] C. Geczy and M. Samonov, "Two centuries of commodity futures premia: Momentum, value and basis," *SSRN Electronic Journal*, 01 2017.

[101] BlackRock, "What is factor investing? [online] accessed: December 20, 2020.."

[102] R. N. Kahn and M. Lemmon, "The asset manager's dilemma: How smart beta is disrupting the investment management industry," *Financial Analysts Journal*, vol. 72, pp. 15 – 20, 2016.

[103] B. G. Malkiel, "Is smart beta really smart?," *The Journal of Portfolio Management*, vol. 40, no. 5, pp. 127–134, 2014.

[104] N. Rabener, "Smart beta or smart marketing? [online] accessed: December 20, 2020.."

[105] D. H. Bailey, J. Borwein, and M. L. de Prado, "Stock portfolio design and backtest overfitting," *Big Data and Innovative Financial Technologies Research Paper Series*, 2016.

[106] T. B. Trafalis and H. Ince, "Support vector machine for regression and applications to financial forecasting," in *IJCNN*, 2000.

[107] Z. min Wang and C. Wu, "Application of support vector regression method in stock market forecasting," *2010 International Conference on Management and Service Science*, pp. 1–4, 2010.

[108] D. H. Bailey, J. Borwein, M. L. de Prado, and Q. J. Zhu, "Pseudo-mathematics and financial charlatanism: The effects of backtest overfitting on out-of-sample performance," *Optimization eJournal*, 2014.

[109] D. Epstein, *Range : how generalists triumph in a specialized world*. London: Macmillan, 2019.

[110] Wikipedia, "List of scientific misconduct incidents. [online] accessed: September 5, 2021.."

[111] L. Takeuchi, "Applying deep learning to enhance momentum trading strategies in stocks," 2013.

[112] M.-F. Tsai, C.-J. Wang, and P.-C. Chien, "Discovering finance keywords via continuous-space language models," *ACM Trans. Management Inf. Syst.*, vol. 7, pp. 7:1–7:17, 2016.

[113] G. Angeli, M. J. J. Premkumar, and C. D. Manning, "Leveraging linguistic structure for open domain information extraction," in *ACL*, 2015.

[114] J. Christensen, Mausam, S. Soderland, and O. Etzioni, “An analysis of open information extraction based on semantic role labeling,” in *K-CAP*, 2011.

[115] S. Soderland, B. Roof, B. Qin, S. Xu, Mausam, and O. Etzioni, “Adapting open information extraction to domain-specific relations,” *AI Magazine*, vol. 31, pp. 93–102, 2010.

[116] K. Narasimhan, A. Yala, and R. Barzilay, “Improving information extraction by acquiring external evidence with reinforcement learning,” in *EMNLP*, 2016.

[117] Mausam, “Open information extraction systems and downstream applications,” 2016.

[118] O. Etzioni, A. Fader, J. Christensen, S. Soderland, and Mausam, “Open information extraction: The second generation,” in *IJCAI*, 2011.

[119] Mausam, M. Schmitz, S. Soderland, R. Bart, and O. Etzioni, “Open language learning for information extraction,” in *EMNLP-CoNLL*, 2012.

[120] J. B. Nascimento and M. Cristo, “The impact of structured event embeddings on scalable stock forecasting models,” in *Proceedings of the 21st Brazilian Symposium on Multimedia and the Web*, WebMedia ’15, (New York, NY, USA), pp. 121–124, ACM, 2015.

[121] X. Ding, Y. Zhang, T. Liu, and J. Duan, “Deep learning for event-driven stock prediction,” in *IJCAI*, 2015.

[122] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” in *NAACL-HLT*, 2018.

[123] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *NAACL-HLT*, 2019.

[124] D. Araci, “Finbert: Financial sentiment analysis with pre-trained language models,” *ArXiv*, vol. abs/1908.10063, 2019.

[125] K. Z. Z. P. K. Y. Yang, M. C. S. Uy, and A. Huang, “Finbert: A pretrained language model for financial communications,” *ArXiv*, vol. abs/2006.08097, 2020.

[126] A. K. Nassirtoussi, S. R. Aghabozorgi, Y. W. Teh, and D. C. L. Ngo, “Text mining for market prediction: A systematic review,” *Expert Syst. Appl.*, vol. 41, pp. 7653–7670, 2014.

[127] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni, "Open information extraction from the web," in *CACM*, 2008.

[128] Y. Peng and H. Jiang, "Leverage financial news to predict stock price movements using word embeddings and deep neural networks," in *HLT-NAACL*, 2016.

[129] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013.

[130] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *EMNLP*, 2014.

[131] "Word definition: Polysemy. [online] accessed: December 17, 2020.," 2020.

[132] J. Alammar, "The illustrated bert, elmo, and co. (how nlp cracked transfer learning). [online] accessed: March 15, 2021.."

[133] C. Manning, "Cs224n: Natural language processing with deep learning," 2021.

[134] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *ArXiv*, vol. abs/1706.03762, 2017.

[135] G. Ritter, "Machine learning for trading," 2017.

[136] Z. Tan, H. C. Quek, and P. Y. K. Cheng, "Stock trading with cycles: A financial application of anfis and reinforcement learning," *Expert Syst. Appl.*, vol. 38, pp. 4741–4755, 2011.

[137] Y. Deng, F. Bao, Y. Kong, Z. Ren, and Q. Dai, "Deep direct reinforcement learning for financial signal representation and trading," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, pp. 653–664, 2017.

[138] N. Brown and T. Sandholm, "Superhuman ai for heads-up no-limit poker: Libratus beats top professionals," *Science*, 2017.

[139] Wikipedia, "Texas hold'em. [online] accessed: December 21, 2018.."

[140] J. N. Foerster, R. Y. Chen, M. Al-Shedivat, S. Whiteson, P. Abbeel, and I. Mordatch, "Learning with opponent-learning awareness," *CoRR*, vol. abs/1709.04326, 2017.

[141] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente, “Multiagent cooperation and competition with deep reinforcement learning,” in *PloS one*, 2017.

[142] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. R. Baker, M. Lai, A. Bolton, Y. Chen, T. P. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis, “Mastering the game of go without human knowledge.,” *Nature*, vol. 550 7676, pp. 354–359, 2017.

[143] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, “Mastering chess and shogi by self-play with a general reinforcement learning algorithm,” 2017.

[144] Z. Liang, K. Jiang, H. Chen, J. Zhu, and Y. Li, “Adversarial deep reinforcement learning in portfolio management,” *arXiv: Portfolio Management*, 2018.

[145] Z. Jiang, D. Xu, and J. Liang, “A deep reinforcement learning framework for the financial portfolio management problem,” *CoRR*, vol. abs/1706.10059, 2017.

[146] C.-Y. Huang, “Financial trading as a game: A deep reinforcement learning approach,” *CoRR*, vol. abs/1807.02787, 2018.

[147] M. Prado, *Advances in financial machine learning*. Hoboken, New Jersey: John Wiley & Sons, Inc, 2018.

[148] M. Sewell, *Application of Machine Learning to Financial Time Series Analysis*. Thesis (doctoral), University College London, 2017.

[149] B. Jha, “Google bert: From 10% to 100%. [online] accessed: December 20, 2020.,” October 2020.

[150] W. D. Heavenarchive, “Predictive policing algorithms are racist. they need to be dismantled. [online] accessed: November 24, 2020..”

[151] S. Council, “Facts & data on small business and entrepreneurship. [online] accessed: March 11, 2021..”

[152] H.-C. L. Jying-Nan Wang, Yuan-Teng Hsu, “How useful are the various volatility estimators for improving garch-based volatility forecasts? evidence from the nasdaq-100 stock index,” *International Journal of Economics and Financial Issues*, vol. 4, no. 3, pp. 651–656, 2014.

[153] N. Schmidlin, *Art of company valuation and financial statement analysis : a value investor's guide with real-life case studies*. Chichester, West Sussex: Wiley, 2014.

[154] P. Mohanram, S. Saiy, and D. Vyas, "Fundamental analysis of banks: the use of financial statement information to screen winners from losers," *Review of Accounting Studies*, vol. 23, pp. 200–233, March 2017.

[155] L. D. Consortium., "The new york times annotated corpus. [online] accessed: October 17, 2020.."

[156] S. Moritz, A. Sardá, T. Bartz-Beielstein, M. Zaefferer, and J. Stork, "Comparison of different methods for univariate time series imputation in r," *ArXiv*, vol. abs/1510.03924, 2015.

[157] W. Badr, "6 different ways to compensate for missing values in a dataset (data imputation with examples). [online] accessed: January 12, 2021.."

[158] S. Kaushik, "Introduction to feature selection methods with an example (or how to select the right variables?). [online] accessed: November 21, 2020.."

[159] S. Ozdemir, *Feature engineering made easy : identify unique features from your dataset in order to build powerful machine learning systems*. Birmingham, UK: Packt Publishing, 2018.

[160] S. Galli, *Python feature engineering cookbook: over 70 recipes for creating, engineering, and transforming features to build machine learning models*. Birmingham, UK: Packt Publishing, 2020.

[161] H. Motoda and H. Liu, "Feature selection extraction and construction," 2002.

[162] X. Liu, *Methods and applications of longitudinal data analysis*. Amsterdam: Academic Press, 2015.

[163] S. Oh, "Multiple imputation on missing values in time series data," Master's thesis, Duke University, 2015.

[164] J. J. Wang, "Handle missing values in time series for beginners. [online] accessed: January 27, 2021.."

[165] S. documentation., "Preprocessing data. [online] accessed: December 26, 2020.."

[166] Google, "Normalization. [online] accessed: December 30, 2020.."

[167] S. documentation., "Robust scaler. [online] accessed: October 24, 2020.."

[168] S. documentation., "Quantile transformer. [online] accessed: October 24, 2020.."

[169] S. Huang, *Generalized Classes of Distributions with Applications to Income and Lifetime Data.* PhD thesis, Georgia Southern University, 2014.

[170] D. M. Lane, "Log transformation. [online] accessed: October 20, 2020.."

[171] "Why log returns [online] accessed: January 22, 2019.." `https://quantivity.wordpress.com/2011/02/21/why-log-returns/`.

[172] D. Jurafsky and M. H. James, "Speech and language processing." Unpublished., August 2020.

[173] A. Géron, *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems*. Sebastopol, CA: O'Reilly Media, 2019.

[174] K. Li and P. Mohanram, "Fundamental analysis: Combining the search for quality with the search for value," 2016.

[175] N. Eiamkanitchat, T. Moontuy, and S. Ramingwong, "Fundamental analysis and technical analysis integrated system for stock filtration," *Cluster Computing*, vol. 20, pp. 883–894, 2016.

[176] O. B. Sezer, A. M. Ozbayoglu, and E. Dogdu, "An artificial neural network-based stock trading system using technical analysis and big data framework," in *ACM Southeast Regional Conference*, 2017.

[177] Y. Z. Zhai, A. L. Hsu, and S. K. Halgamuge, "Combining news and technical indicators in daily stock price trends prediction," in *ISNN*, 2007.

[178] Z. Zhou, M. Gao, Q. Liu, and H. Xiao, "Forecasting stock price movements with multiple data sources: Evidence from stock market in china," *Physica A: Statistical Mechanics and its Applications*, vol. 542, p. 123389, 2020.

[179] P. C. Tetlock, M. Saar-Tsechansky, and S. A. Macskassy, "More than words : Quantifying language to measure firms ' fundamentals," 1998.

[180] J. Ayres, C. Hevia, and J. P. Nicolini, "Real exchange rates and primary commodity prices," IDB Working Paper Series, Inter-American Development Bank, 2019.

[181] M. Griffis, *Economic indicators for dummies*. Hoboken, N.J: John Wiley & Sons, 2011.

[182] P. D. Corte, S. J. Riddiough, and L. Sarno, "Currency premia and global imbalances," *The Review of financial studies*, vol. 29, no. 8, pp. 2161–2193, 2016.

[183] L. A. Smales, "Order imbalance, market returns and macroeconomic news: Evidence from the australian interest rate futures market," *Research in International Business and Finance*, vol. 26, no. 3, pp. 410–427, 2012.

[184] L. Ferrara and O. Vigna, "Cyclical relationships between gdp and housing market in france: Facts and factors at play," Working papers 268, Banque de France, 2009.

[185] L. Swedroe, "Explaining the currency carry premium. [online] accessed: December 13, 2020.."

[186] D. Duong, T. Nguyen, and M. Dang, "Stock market prediction using financial news articles on ho chi minh stock exchange," in *Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication*, IMCOM '16, (New York, NY, USA), pp. 71:1–71:6, ACM, 2016.

[187] M. A. Trovero and M. Leonard, "Sas 2020-2018 time series feature extraction," 2018.

[188] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *EMNLP/IJCNLP*, 2019.

[189] V. Sakalauskas and D. Kriksciuniene, "Research of the calendar effects in stock returns," in *Business Information Systems Workshops*, Lecture Notes in Business Information Processing, pp. 69–78, Berlin, Heidelberg: Springer Berlin Heidelberg.

[190] T. Carrazedo, J. D. Curto, and L. Oliveira, "The halloween effect in european sectors," *Research in International Business and Finance*, vol. 37, pp. 489–500, 2016.

[191] V. Kayacetin and S. Lekpek, "Turn-of-the-month effect: New evidence from an emerging stock market," *Finance Research Letters*, vol. 18, pp. 142–157, 2016.

[192] I. London, "Encoding cyclical continuous features—24-hour time. [online] accessed: March 2, 2021.."

[193] W. Kenton, "Weekend effect. [online] accessed: August 25, 2021.."

[194] S. Langlois, "The best day of the year for the stock market is almost here. [online] accessed: August 25, 2021.."

[195] J. Chen, "September effect. [online] accessed: February 15, 2021.."

[196] Z. Zarska, *Finanzas internacionales*. México: Mc Graw Hill, 2013.

[197] J. Chen, "Double witching. [online] accessed: February 15, 2021.."

[198] J. Chen, "Quadruple witching. [online] accessed: February 15, 2021.."

[199] S. Li, "Hands-on feature engineering for natural language processing.," in *QCon*, 2019.

[200] A. Zheng, *Evaluating Machine Learning Models*. City: O'Reilly Media, Inc, 2015.

[201] F. Brunet, *Contributions to Parametric Image Registration and 3D Surface Reconstruction*. PhD thesis, Université d'Auvergne, November 2010.

[202] J. Brownlee, "Probabilistic model selection with aic, bic, and mdl. [online] accessed: November 21, 2020.."

[203] P. Westfall, "Trading with gaussian models of statistics. [online] accessed: December 29, 2018.."

[204] K. Zucchi, "Lognormal and normal distribution. [online] accessed: February 15, 2021.."

[205] "Why are stock prices considered log-normal? [online] accessed: January 15, 2018.." `https://finance.zacks.com/stock-prices-considered-lognormal-10606.html`.

[206] "Q&a with ifa: Are monthly stock market returns normally distributed? [online] accessed: January 15, 2018.." `https://www.ifa.com/articles/with_stock_returns_normally_distributed/`.

[207] P. Harrison, "Similarities in the distribution of stock market price changes between the eighteenth and twentieth centuries," *The Journal of Business*, vol. 71, no. 1, pp. 55–79, 1998.

[208] S. Keen, *Debunking economics : the naked emperor dethroned*. London New York, NY: Zed, 2011.

[209] S. Leng, “China’s economic census uncovers more fake data as officials promise ’zero tolerance’ to data manipulation. [online] accessed: November 29, 2020..”

[210] M. Kumar, “India’s incredulous data: Economists create own benchmarks. [online] accessed: November 29, 2020..”

[211] L. Dunsmuir, “Explainer: May’s jobs report stunner: real deal or head fake? [online] accessed: November 29, 2020..”

[212] S. McDonell, “China’s luckin coffee slumps on ’fake’ data news. [online] accessed: November 29, 2020..”

[213] R. Hotten, “Volkswagen: The scandal explained. [online] accessed: November 29, 2020..”

[214] E. Flitter, “The price of wells fargo’s fake account scandal grows by $3 billion. [online] accessed: November 29, 2020..”

[215] A. Hayes and K. Khartit, “Who is bernie madoff. [online] accessed: November 29, 2020..”

[216] M. Intelligence, “Algorithmic trading market - growth, trends, and forecasts (2020 - 2025). [online] accessed: November 25, 2020..”

[217] M. Hargrave, “Cult stock. [online] accessed: November 25, 2020..”

[218] J. Chen, “Tesla (tsla) will be part of the s&p 500 come december. [online] accessed: November 25, 2020..”

[219] T. Kilgore, “Boeing’s stock drops after trump tweet about canceling air force one order. [online] accessed: November 29, 2020..”

[220] J. O’Toole, “Icahn challenges ackman with 13% herbalife stake. [online] accessed: November 29, 2020..”

[221] K. Korosec, “Tesla shares fall on elon musk ’stock price too high’ tweet. [online] accessed: November 29, 2020..”

[222] H. Lim, “7 types of data bias in machine learning. [online] accessed: October 17, 2020..”

[223] N. Kayser-Bril, “Google apologizes after its vision ai produced racist results. [online] accessed: October 17, 2020..”

[224] J. Buolamwini, "Artificial intelligence has a problem with gender and racial bias. here's how to solve it. [online] accessed: October 17, 2020.."

[225] "Data-mining bias, sample selection bias, survivorship bias, look-ahead bias, and time-period bias. [online] accessed: October 17, 2020.."

[226] C. F. Institute, "What is time period bias? [online] accessed: October 17, 2020.."

[227] J. Chen, "Survivorship bias. [online] accessed: March 31, 2021.."

[228] R. Carbaugh, *International economics*. Mason, Ohio: South-Western Cengage Learning, 2009.

[229] Z. Thiha Tun, "Bill ackman's greatest hits and misses. [online] accessed: February 15, 2021.."

[230] B. F. team., "Omega ratio (cci). [online] accessed: April 10, 2021.."

[231] S. Jansen, *Hands-on machine learning for algorithmic trading : design and implement investment strategies based on smart algorithms that learn from data using Python*. Birmingham, UK: Packt Publishing, 2018.

[232] T. Davidson, "Quantitative momentum strategy. [online] accessed: December 11, 2020.."

[233] X. Bourret Sicotte, "Bias and variance in leave-one-out vs k-fold cross validation. [online] accessed: February 9, 2021.."

[234] S. team., "Macd (moving average convergence/divergence oscillator). [online] accessed: December 29, 2018.."

[235] S. team., "Bollinger bands. [online] accessed: December 29, 2018.."

[236] S. team., "Ultimate oscillator. [online] accessed: December 29, 2018.."

[237] "Trend detection index. [online] accessed: December 29, 2018.." http://traders.com/documentation/feedbk_docs/2001/10/Abstracts_new/Pee/pee.html.

[238] S. team., "Williams r. [online] accessed: December 29, 2018.."

[239] J. Maverick, "How is the average directional index (adx) calculated and what is the formula? [online] accessed: February 25, 2021.."

[240] C. F. Institute, "Adx indicator — technical analysis. [online] accessed: February 25, 2021.."

[241] C. F. Institute, “Directional movement index (dmi). [online] accessed: February 25, 2021..”

[242] C. Mitchell, “Negative directional indicator (-di). definition and uses. [online] accessed: February 25, 2021..”

[243] C. Mitchell, “Positive directional indicator (+di). definition and uses. [online] accessed: February 25, 2021..”

[244] S. team., “Average directional index (adx). [online] accessed: February 25, 2021..”

[245] L. team., “evwma - elastic volume weighted moving average. [online] accessed: December 29, 2018..”

[246] C. P. Fries, “Introduction to elastic volume weighted moving average (evwma) approximating the average price paid per share. [online] accessed: February 25, 2021..”

[247] “Volume weighted moving average (vwma) [online] accessed: December 29, 2018..” `https://tradingsim.com/blog/volume-weighted-moving-average/`.

[248] R. Documentation, “Sma. [online] accessed: December 29, 2018..”

[249] S. team., “Commodity channel index (cci). [online] accessed: December 29, 2018..”

[250] S. team., “Relative strength index (rsi). [online] accessed: December 29, 2018..”

[251] A. Hayes, “Operating activities. [online] accessed: October 17, 2020..”

[252] M. Hargrave, “Return on assets. [online] accessed: October 20, 2020..”

[253] Accountingverse, “Return on investment (roi). [online] accessed: October 20, 2020..”

[254] A. Hayes, “Cash conversion cycle. [online] accessed: October 20, 2020..”

[255] U. of Michigan., “Surveys of consumers. [online] accessed: October 17, 2020..”

[256] OECD, “Consumer confidence index (cci). [online] accessed: October 17, 2020..”

[257] FRED, “Board of governors of the federal reserve system (us), delinquencies on all loans and leases, commercial and industrial, all commercial banks [dallciacbep]. [online] accessed: October 20, 2020..”

[258] J. Chappelow, “Money supply. [online] accessed: October 17, 2020..”

[259] F. R. B. of St. Louis, "How monetary policy works. [online] accessed: October 17, 2020.."

[260] P. Bajpai, "How central banks control the supply of money. [online] accessed: October 17, 2020.."

[261] F. R. B. of St. Louis, "Fred economic data. [online] accessed: January 22, 2018.."

[262] FreddieMac, "Choose your term. [online] accessed: October 20, 2020.."

[263] B. of England, "What are interest rates? [online] accessed: October 17, 2020.."

[264] J. Chen, "Ted spread. [online] accessed: October 20, 2020.."

[265] J. Kagan, "London interbank offered rate (libor). [online] accessed: October 20, 2020.."

[266] J. Chen, "Prime rate. [online] accessed: March 11, 2021.."

[267] BLS., "Employment situation report. [online] accessed: October 17, 2020.."

[268] BLS., "Producer price indexes. [online] accessed: October 17, 2020.."

[269] BLS., "Producer price index videos. [online] accessed: October 17, 2020.."

[270] F. R. B. of Cleveland, "Median cpi. [online] accessed: October 17, 2020.."

[271] U. C. Bureau, "Manufacturers' shipments, inventories, & orders. [online] accessed: October 17, 2020.."

[272] U. C. Bureau, "How the data are collected. [online] accessed: October 17, 2020.."

[273] U. C. Bureau, "Building permits survey. [online] accessed: October 17, 2020.."

[274] U. C. Bureau, "New residential construction. [online] accessed: October 17, 2020.."

[275] U. C. Bureau, "Monthly retail trade. [online] accessed: October 17, 2020.."

[276] U. C. Bureau, "Advance monthly retail trade survey methodology. [online] accessed: October 17, 2020.."

[277] B. of Economic Analysis, "Gross domestic product. [online] accessed: October 17, 2020.."

[278] B. of Economic Analysis, "Personal income and outlays, august 2020. [online] accessed: October 17, 2020.."

[279] J. Chappelow, “Personal income and outlays. [online] accessed: October 17, 2020..”