Instituto BME

Master of Science Thesis

Optimizing Global Portfolio Weights using Artificial Intelligence and Macroeconomic Analysis

Master in Artificial Intelligence and Quantum Computing Applied to Financial Markets, 11th edition (mIA-X)

Lukasz Tadeusz Harazin

July 2024

Tutor: Fernando de la Calle Silos

TABLE OF CONTENTS

TABLE OF FIGURES

1 INTRODUCTION

The pursuit of optimal portfolio diversification and risk-adjusted returns has long been a cornerstone of investment management. Traditional methods often rely on Modern Portfolio Theory (MPT) and historical data analysis. However, these approaches can be limited by static assumptions and the inherent difficulty of accurately forecasting future market behaviour.

This thesis explores the potential of Artificial Intelligence (AI) and macroeconomic analysis to create a more dynamic and data-driven approach to global portfolio weight optimization. By leveraging the power of AI techniques to identify complex patterns and relationships within vast datasets, I aim to develop an investment algorithm that can adapt to evolving market conditions and generate superior portfolio management.

This research will delve into various AI techniques, such as machine learning methods of Random Forest, XG Boost and Learning to Rate as well as different neural network models. It will validate their applicability in the context of portfolio optimization. Moreover, it will examine the role of macroeconomic analysis in shaping investment strategies and how it can be integrated with AI for portfolio management.

The main motivation for this thesis is to evaluate an alternative to passive investing, via index funds and Exchange-Traded Funds (ETFs), which has been rapidly gaining market share for both individual and institutional investors. A desired result for this investment algorithm would be to achieve both superior absolute returns and risk-adjusted returns in comparison to MSCI All Country World Index (ACWI), taken as a benchmark.

This thesis will try to offer a fresh perspective on global portfolio management in the age of AI. It is hoped to provide new insights and practical applications for the world of finance, helping to navigate the complexities of the global investment landscape.

The research encompasses a comprehensive background study, meticulous data selection and preparation, detailed presentation of the developed models, and an exploration of cloud integration for data collection and model deployment. The culmination of this work is a set of conclusive findings and recommendations for future research.

The foundation of this thesis lies in the study of global diversification, emphasizing the importance of spreading investments across various markets to mitigate risk. The research covers constructing global portfolios using index funds and ETFs, tracking performance of the MSCI All Country World Index (MSCI ACWI) and other broad indices. It incorporates Modern Portfolio Theory and the Black-Litterman Model to establish theoretical and practical frameworks. Additionally, it explores advanced methods such as Hierarchical Risk Parity, Covariance Matrix Denoising, and Nested Clustered Optimization to enhance portfolio stability and performance.

Data selection and preparation are critical components of this research, involving diverse financial and macroeconomic datasets. This includes a thorough review of data sources and various economic indicators essential for analysis. The process encompasses data cleaning, filling missing values, calculating relevant indicators, formatting input data, and splitting the training dataset to ensure robust model development and evaluation.

The presentation of work chapter outlines the comprehensive approach taken in this thesis, starting with the base model and evaluating different inputs and outputs. It examines different neural network architectures and machine learning models. Next hyperparameter tuning is performed on the best performing models. The research also includes a comparison with classical methods and employs data augmentation techniques, particularly using economic calendar, to improve model accuracy.

Cloud integration is explored to facilitate data collection, storage, and model deployment, ensuring scalability and accessibility. This chapter also details the infrastructure used to support the implementation and deployment of the model developed in this research, highlighting the advantages of leveraging cloud technologies in financial modelling.

The thesis concludes with a summary of the work performed, presenting complete results and comparing the final model's performance. It identifies key findings and discusses the implications of the research, offering recommendations for future work to further enhance global portfolio optimization strategies.

2 BACKGROUND

This chapter provides a comprehensive background to the key concepts and theories that underpin the study. It begins by exploring the basic principles of global diversification, and then delves into the specifics of constructing a global portfolio using index funds and ETFs.

The chapter then introduces the MSCI ACWI, a key benchmark for global equity markets, before moving on to an in-depth discussion of Modern Portfolio Theory. This includes an exploration of the Efficient Frontier and Optimal Portfolios, as well as the application of Quadratic Programming to solve the optimal portfolio. The chapter also addresses the concept known as Markowitz's Curse, which refers to the potential pitfalls of Modern Portfolio Theory.

The latter part of the chapter introduces the Black-Litterman Model, a practical method for portfolio optimization that incorporates investor views. This is followed by a discussion on Hierarchical Risk Parity, a novel approach to portfolio construction that seeks to address some of the limitations of traditional methods.

The chapter concludes with a look at Covariance Matrix Denoising, a technique for improving the stability and performance of portfolio optimization, and Nested Clustered Optimization, a method for refining the portfolio construction process.

Each section provides a detailed overview of the topic, including its theoretical underpinnings, practical applications, and relevance to the broader field of portfolio management.

2.1 GLOBAL DIVERSIFICATION

Global diversification is an investment strategy designed to mitigate risk and enhance returns by spreading investments across various geographic regions and asset classes. By allocating assets in both developed and emerging markets, investors can balance the inherent risks and rewards associated with different economic environments. This approach leverages the distinct economic cycles, political landscapes, and currency fluctuations of each region, ensuring that the performance of a single market does not disproportionately impact the overall portfolio.

The benefits of global diversification are multifaceted, primarily revolving around risk reduction and potential for enhanced returns. Investing in multiple regions allows for a buffer against localized economic downturns and political instability, as adverse conditions in one market may be offset by favourable conditions in another. Additionally, the strategy taps into high-growth opportunities available in emerging markets, which can offer substantial returns compared to more mature economies.

However, implementing global diversification requires careful consideration of several challenges. Political and economic stability varies significantly across regions, with emerging markets often exhibiting higher volatility. Currency exchange risks can affect the returns on foreign investments, both positively and negatively. Additionally, investors must navigate different regulatory environments and potential information asymmetry, which can complicate investment decisions. Despite these complexities, global diversification remains a valuable strategy for long-term investors seeking to optimize their portfolio's risk-reward profile. (Vanguard UK Professional, 2024)

2.2 GLOBAL PORTFOLIO WITH INDEX FUNDS AND ETFS

In recent years there has been a growing trend of passive investing through index funds and global ETFs for both individual and institutional investors. Index funds and Exchange-Traded Funds (ETFs) based on the MSCI World and MSCI All Country World Index (ACWI) have become key instruments for achieving global diversification in an investment portfolio.

The MSCI World Index tracks large and mid-cap representation across 23 developed markets countries, covering approximately 85% of the free float-adjusted market capitalization in each country. The MSCI ACWI goes a step further by including both developed and emerging markets, covering large and mid-cap representation across 23 developed and 24 emerging markets countries. This allows investors to gain exposure to a wide range of global equities through a single investment.

Therefore, investing in index funds and ETFs based on the MSCI World and MSCI ACWI indices can be an effective strategy for achieving global diversification. It allows investors to spread their investments across different countries and sectors, thereby reducing the risk associated with any single investment or market.

2.3 MSCI ACWI

The MSCI All Country World Index (ACWI) is a stock index designed to provide a broad measure of equity-market performance throughout the world. It is maintained by Morgan Stanley Capital International (MSCI) and comprises the stocks of nearly 3,000 companies from 23 developed countries and 24 emerging markets (Mitchell, 2024). As of March 2024, it covers following markets:

Developed Markets			Emerging Markets		
Americas	EMEA	APAC	Americas	EMEA	APAC
Canada	Austria	Australia	Brazil	Czech Republic	China
USA	Belgium	Hong Kong	Chile	Egypt	India
	Denmark	Japan	Colombia	Greece	Indonesia
	Finland	New Zealand	Mexico	Hungary	Korea
	France	Singapore	Peru	Kuwait	Malaysia
	Germany			Poland	Philippines
	Ireland			Qatar	Taiwan
	Israel			Saudi Arabia	Thailand
	Italy			South Africa	
	Netherlands			Turkey	
	Norway			UAE	
	Portugal				
	Spain				
	Sweden				
	Switzerland				
	UK				

Figure 1Countries Included in MSCI AWCI Index as of March 2024

The MSCI ACWI captures large and mid-cap representation across these countries, covering approximately 99% of the global investable equity opportunity set. The index is used as a benchmark for global equity funds and as a guide to asset allocation. Approximately $4.6 trillion in assets are benchmarked to the index as of December 31, 2023. Along with MSCI World and FTSE All World Index it is the most selected index for global portfolios.

2.4 MODERN PORTFOLIO THEORY

Traditional way to optimize portfolio has been Modern Portfolio Theory (MPT), also known as mean-variance analysis. It is a fundamental framework used by investors and financial professionals to construct portfolios that aim to maximize expected returns while managing risk. Developed by American economist Harry Markowitz in 1952, MPT revolutionized portfolio management by introducing rigorous mathematical principles.

MPT recognizes that diversification is essential for managing risk. By combining assets with different risk profiles, investors can achieve better risk-adjusted returns. The central idea is that owning a mix of assets—such as stocks, bonds, and real estate—reduces the impact of adverse events affecting any single investment.

During a portfolio construction MPT evaluates investments within the context of an entire portfolio rather than in isolation. The expected return of a portfolio is a weighted average of the expected returns of its individual assets. The key is to find an optimal combination of assets that balances risk and return. Moreover, an important assumption is that investors are risk-averse, meaning they prefer less risk for a given level of return. Utility theory quantifies an investor's preferences and trade-offs between risk and reward.

The variance of return or its transformation, the standard deviation, is used as a measure of risk, because it is tractable when assets are combined into portfolios.

2.4.1 Efficient Frontier and Optimal Portfolios

The efficient frontier is a curve that represents the best possible trade-offs between risk and return. Portfolios lying on the efficient frontier offer the highest expected return for a given level of risk. The optimal portfolio is the point on the efficient frontier that aligns with an investor's risk tolerance. Visual representation is presented in the figure below (Lindquist, 2022)

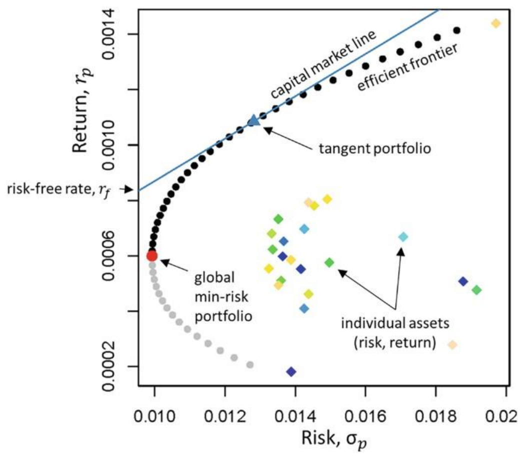

Figure 2 Efficient Frontier with Capital Market Line

Capital Market Line (CML) extends the efficient frontier by incorporating a risk-free asset (such as government bonds). The risk-free asset allows investors to create a risk-adjusted portfolio by combining it with risky assets. The tangent point between the CML and the efficient frontier identifies the optimal risky portfolio.

2.4.2 Solving Optimal Portfolio using Quadratic Programming

Quadratic optimization is a type of mathematical optimization that deals with quadratic functions. In the context of Modern Portfolio Theory (MPT), it plays a crucial role in determining the optimal allocation of assets in a portfolio.

Optimization can be done by minimizing variance to specific level of expected return or by maximizing expected return for a given variance level. Quadratic programming allows us to add various restrictions when it comes to number or assets, allowed weights or even minimal rebalancing when transitioning from one period to the next.

2.4.3 Markowitz's Curse

In his book 'Advances in Financial Machine Learning' Marcos Lopez de Prado points out various drawbacks of Modern Portfolio Theory with the main one being its instability. (Prado M. L., Advances in Financial Machine Learning, 2018) With more correlated instruments, we need more diversification and yet we might receive unstable results. In that case, the benefits of diversification are often offset by estimation errors. As a possible solution to that issue, the author proposes his own method known as Hierarchical Risk Parity, described later on.

2.5 BLACK-LITTERMAN MODEL

The Black-Litterman model is a mathematical model for portfolio allocation developed by Fischer Black and Robert Litterman at Goldman Sachs in 1990. It seeks to overcome problems that institutional investors have encountered in applying modern portfolio theory (MPT) in practice.

One of the limitations of MPT is that it assumes that past expected returns will continue in the future. The Black-Litterman model incorporates observed market data along with an investor's projections of future expected returns based on models like the capital asset pricing model (CAPM) or other. The model essentially modifies the default MPT allocation by considering expectations of future performance.

The Black-Litterman model was created to develop "a quantitative and disciplined approach to structuring international bond portfolios in a manner consistent with the portfolio manager's unique view of markets". It continues to be used and is recognized as a key mathematical tool in the investment world.

However, that model must be used with great care as it can introduce bias when investor's views are inaccurate or based on false assumption. In that sense it doesn't promise or guarantee optimal portfolio when working on poor assumptions. (Kenton, 2023)

2.6 HIERARCHICAL RISK PARITY

Hierarchical Risk Parity (HRP) is a portfolio optimization strategy introduced by Marcos Lopez de Prado which addresses three major concerns of quadratic optimizers and Markowitz's Critical Line Algorithm (CLA). It applies modern graph theory and machine learning techniques to build a diversified portfolio based on the information retrieved from the covariance matrix. As author proves in his paper (Prado M. L., Building Diversified Portfolios that Outperform Out-of-Sample, 2016), it delivers less risky portfolios out-of-sample compared to other traditional risk-parity methods.

The first step in the HRP method is Hierarchical Tree Clustering. This involves breaking down the assets into hierarchical clusters. Hierarchical clustering algorithms uncover a hierarchical structure of the considered investment universe, resulting in a tree-based representation.

Following the clustering of securities, the covariance matrix is reordered in a process known as Quasi-Diagonalization. This reorganization places similar stocks close to each other, ensuring that similar assets are grouped together.

The final step in the HRP method is Recursive Bisection. This involves assigning weights to each asset in the portfolio. The allocation is done in such a way that equities only compete with similar equities for spots in the portfolio.

Hierarchical Risk Parity is a robust method of portfolio construction, allowing user to introduce constraints and adjust the tree structure without compromising the algorithm's search.

2.7 COVARIANCE MATRIX DENOISING

In the context of financial applications, covariance matrices are often numerically ill-conditioned, because of few independent observations used to calculate large number of features. That leads to classical portfolio optimization solutions often underperforming the naïve allocation out-of-sample. Matrix denoising, introduced by Marcos Lopez de Prado in Machine Learning for Asset Managers (Prado M. M., 2020), is used to correct the covariance matrix and reduce its instability.

This technique is based on linear algebra, particularly eigenvalues and eigenvectors. In practice, it aims at removing market components present in the correlation matrix, used in portfolio optimization. That magnifies more subtle signals hiding under the general market trends. According to the author, matrix denoising is a highly recommended step before any optimization operations.

Matrix denoising has several practical implications in portfolio management. It can significantly improve investors' returns by providing more accurate risk assessments, enabling robust portfolio construction.

2.8 NESTED CLUSTERED OPTIMIZATION

Nested Clustered Optimization (NCO) is another portfolio optimization method developed by Marcos Lopez de Prado (Prado M. M., 2020). It applies hierarchy logic and k-means clustering to the Mean-Variance Optimization problem, which suffers from instability when the covariance matrix is large.

The algorithm consists of several steps:

- Denoising the Covariance Matrix: The covariance matrix of the outcomes is taken as an input.
- Choosing the Optimal Number of Clusters: The optimal number of clusters is determined using k-means algorithm or one of its variants.
- Clustering the Correlation Matrix: The covariance matrix is clustered into subsets of highly correlated variables.
- Finding Intra-Cluster Weights: Computes optimal intracluster allocations by using for example minimum variance allocation.
- Finding Inter-Cluster Weights: The optimal weights allocation for the reduced covariance matrix is computed by minimizing portfolio volatility.
- Computing Final Allocations: The final allocations are computed as a dot-product of the allocations between the clusters and inside the clusters.

The NCO algorithm can also maximize the Sharpe ratio by taking the expected returns as an input. The purpose of the NCO algorithm is to introduce hierarchy logic to the portfolio optimization problem, making it more robust and efficient. By containing the numerical instability at each level of the tree, it reduces instability propagation to parent cluster or the rest of the correlation matrix. Moreover, it can be utilized in combination with other methods like Black-Litterman or constrained optimization.

3 DATA SELECTION AND PREPARATION

This chapter delves into the critical process of data selection and preparation, which forms the backbone of any data-driven study or analysis. It begins by detailing the various data sources utilized in this study, including reputable organizations and platforms such as the Organisation for Economic Co-operation and Development (OECD), World Bank (WB), Bank of International Settlements (BIS), Yahoo Finance, Investing.com, among others.

The chapter then explores the selection and usage of macroeconomic data, providing a comprehensive list of economic indicators selected for the investment algorithm, and the concept of the Investment Clock. This section provides a robust framework for understanding all indicators used as inputs to the model.

The benchmark for the study, the MSCI ACWI, is then introduced, with a detailed discussion on its inclusions and exclusions, changes in country weights, and the selection of countries. This sets the stage for a comparative analysis of the investment strategies against a globally recognized benchmark.

The chapter then describes the specifics of the investment instruments used in the study, including the selection of ETF data, the creation of synthetic ETF prices, and the calculation of monthly returns. This provides a clear understanding of the financial instruments used in the research.

The final section, data preparation, discusses the various steps taken to prepare the data for analysis, including filling missing values, calculating available indicators data, formatting input data, and splitting training data.

3.1 DATA SOURCES

In the course of conducting research for this master's thesis, I have exclusively utilized open databases accessible via the internet. While these sources may not match the quality of paid data providers such as the Bloomberg Terminal, which is considered the gold standard, they nonetheless provide adequate data coverage to facilitate a detailed macroeconomic analysis.

There are various international organizations collecting broad range of macroeconomic statistics which are publicly available. Main organizations with open datasets are World Bank (WB), Organisation for Economic Co-operation and Development (OECD), International Monetary Fund (IMF), United Nations (UN) and Bank for International Settlements (BIS).

Additionally, there are multiple financial websites that offer a variety of resources for investors and traders. The most popular portals providing market data and economic indicators are Investing.com, Yahoo Finance, Trading Economics and FX Empire.

After extensive analysis of all available data sources, I have selected those with the highest data coverage for all countries selected for global portfolio. In the subsequent sections, I will provide a detailed description of all the data sources utilized in the investment algorithm.

3.1.1 Organisation for Economic Co-operation and Development (OECD)

The Organisation for Economic Co-operation and Development (OECD) compiles a comprehensive collection of economic, social, and environmental datasets encompassing member countries and selected non-member economies. These datasets include:

- National accounts, trade, labour, and productivity statistics
- Science and technology innovation indicators
- Entrepreneurship and formation data
- Environmental-economic accounting information
- Development resource flows and official development assistance (ODA) data

In performed macroeconomic analysis, a wide range of monthly and quarterly economic indicators have been used. It has the best data coverage for most countries included in MSCI ACWI index. Data can be downloaded using csv exports or developer's API. In my algorithm, I have managed to use multiple indicators from OECD.Stat website reaching back to 1999.

3.1.2 World Bank (WB)

World Bank Data is a comprehensive resource for global development data. It offers data by country or indicator, databases, pre-formatted tables, reports, and other resources. It includes tools for data analysis and visualization, access to microdata, international debt statistics, and development indicators. It also provides information on World Bank's finances and lending projects.

For my analysis, it includes even more countries than OECD, for example various Middle East countries. Unfortunately, it provides mainly yearly data with significant delay. That limits its use in terms of monthly portfolio rebalancing. Data can be exported using csv files or third-party python libraries providing an easy access to all indicators formatted automatically to pandas' data frames.

3.1.3 Bank of International Settlements (BIS)

The BIS Data Portal provides global financial statistics. It includes data on international banking activity, debt securities, credit to non-financial sectors, credit-to-GDP gaps, debt service ratios, global liquidity indicators, among others. All data can be downloaded in CSV or SDMX format. Moreover, databases can be accessed via REST API providing data in XML format.

In my work I have used its data for central bank rates. I haven't found any other place with such a complete daily data for interest rates for almost all central banks in both developed and emerging economies. Data is updated weekly what provides very precise and up to date information.

3.1.4 Yahoo Finance

Yahoo Finance is a financial website that provides investors with:

- Market Data: Real-time stock quotes, news, and analysis.
- Investment Tools: Portfolio management tools and research resources.
- Financial News: Up-to-date headlines and insights on global markets.

Data can be easily downloaded using open-source Python library yfinance which reads real time market data from publicly available APIs. This library has been used in this thesis to read historical prices for ETFs, stock indices and currency rates.

3.1.5 Investing.com

Investing.com is a leading financial website and platform that offers a variety of tools and resources for investors and traders. It offers:

- Real-time market data: Stocks, bonds, commodities, currencies, futures, options across 70+ exchanges.
- Analysis & news: Articles on market trends, company performance, and economic events.
- Investment tools: Economic calendar, earnings calendar, technical analysis tools, portfolio tracker.

In my work, its economic calendar has provided a great value. While it has slightly worse data coverage than OECD dataset, it does provide report date time for all indicators. That is immensely important for proper back testing because we know exactly what data was available at a certain point in time. While other databases only hold final revision for given indicators like GDP values, with exact investing calendar we can read all values for subsequent revisions and know on which days they were released. Economic calendar cannot be easily downloaded, so I have used techniques of web scrapping using Selenium library. I have managed to download over 320 thousand data points for 50 countries in MSCI ACWI index dating back to 1999.

Apart from economic calendar, I have also downloaded csv data for missing currency rates and selected stock indices that aren't available on Yahoo Finance. Moreover, it has got complete historical data for Manufacturing PMI indicators, hard to find anywhere else, which are very useful to calculate economic cycles.

3.1.6 Morgan Stanley Capital International (MSCI)

Morgan Stanley Capital International maintains a family of stock market indexes which are widely followed by investors around the world. MSCI indexes are widely used benchmarks for global stock markets. They track different segments (like developed, emerging or country-specific) and by market cap (where bigger companies influence more). This allows investors to see how their portfolios perform compared to a specific market segment.

In my algorithm, I have used MSCI indexes to both obtain MSCI ACWI benchmark and fill missing returns on selected exchange traded funds which were created after 1999.

3.1.7 Other sources

Apart from data sources listed above, I have explored other datasets and web portals:

- **The International Monetary Fund (IMF)** Data provides comprehensive economic, financial, and socio-demographic statistics. It includes data on direct investment, climate transition, greenhouse gas emissions, world economic outlook, international finance, global financial stability, fiscal monitor, and exchange rates. It covers many similar indicators to OECD. However, it has worse historical data coverage and more missing values for old data. In the end I opted to use OECD datasets instead.

- **Trading Economics** is a platform that provides similar features to Investing.com but is better protected against automatic web scrapping, making it harder to download data and offers paid subscriptions to access its datasets.
- **FX Empire** is another data provider resembling Investing.com and Trading Economics. Unfortunately, its interactive website makes it difficult to download data using web scrapping.
- **FRED** (Federal Reserve Economic Data) is a trusted source for economic data since 1991. It provides access to over 824,000 US and international time series. It does provide an excellent coverage for US economy, but lacks international indicators, mostly referencing data from OECD and other public database.
- **EBS Statistics** provide comprehensive data that supports all aspects of the ECB's work, including monetary policy, financial stability, and banking supervision. It covers European economies in great details but covers few international indicators, outside Euro Zone, required to optimise global portfolio.

3.2 MACROECONOMIC DATA

To perform macroeconomic analysis, we need to collect a wide range of economic indicators. In my research I have downloaded 70 indicators and divided them into 9 distinct categories of related measures. That has made it easier to get an overall view and select the best indicators for an investment algorithm.

Each of these categories provides a different perspective on the health of the economy and can be used together to get a comprehensive understanding of economic conditions. Below brief descriptions of each category are provided:

1. ***Stock Market***: This refers to the collection of markets and exchanges data. Moreover, provides an overview of stock market for each country with its relevant metrics.
2. ***GDP***: It refers to various measures of Gross Domestic Product when it comes to total value, growth rate or value per capita.
3. ***Labour***: Labour market indicators provide an overview of the economic health of the employment sector. They include metrics such as the unemployment rate and population size.
4. ***Prices***: This provides various measures of price changes over time.
5. ***Money***: This refers to various indicators like money supply, interest rates, and credit availability. These indicators can influence spending and investment activities in an economy.
6. ***Trade***: Trade indicators include metrics related to imports, exports, trade balance, and terms of trade. They provide insights into a country's competitiveness, the demand for its goods and services, and its economic ties with other countries.
7. ***Government***: Government indicators include government spending, budget deficits, and public debt. These indicators can show how government policy is affecting the economy.
8. ***Business***: Business indicators include measures of business confidence, industrial production, and manufacturing output. They provide insights into the health of the business sector and can be leading indicators for the overall economy.
9. ***Customer***: This refers to measures of consumer confidence, which provide an indication of consumers' attitudes about the health of the economy and their willingness to spend.

3.2.1 Full list of economic indicators

Full list of economic indicators downloaded for analysis is included in the table below. There are 5 columns to describe each indicator:

- Indicator – full name of the indicator.
- Source – data source from which the indicator has been downloaded.
- Freq – frequency of the data. Can be D-daily, M-monthly, Q-quarterly, or Y-yearly.
- Measure – additional description of applied measure.
- Data coverage – a percentage of available data for all 50 countries from MSCI ACWI for years 1999 to 2024.

Table 1 Full List of Macroeconomic Indicators

Indicator	Source	Freq	Measure	Data Coverage
1. Stock Market				
Stock Indices	Yahoo Finance, Investing.com	D	Index	
Currency Rates	Yahoo Finance, Investing.com	D	Exchange rates	
ETFs in USD	Yahoo Finance	D	US Dollars	
ETFs in EUR	Yahoo Finance	D	Euros	
MSCI Indices	MSCI	D	Index	
Stock Market Cap	World Bank	Y	Current US Dollars	76.2
Stock Market Cap Pct of GDP	World Bank	Y	Pct of GDP	76.2
Listed Domestic Companies Total	World Bank	Y	Count	80.5
Stocks Traded Total Value	World Bank	Y	Current US Dollars	78.2
Stocks Traded Total Value Pct of GDP	World Bank	Y	Pct of GDP	78.2
2. GDP				
GDP Annual Growth Rate	OECD	Q	Growth YoY	73.9
GDP Growth Rate	OECD	Q	Growth QoQ	73.1
GDP Per Capita	OECD	Q	Per Head, US dollars (2015)	60.1
GDP Current Prices US Dollars	OECD	Y	Current US Dollar	77
GDP Current Prices PPP	OECD	Y	Current US Dollar	77
GDP	World Bank	Y	Current US Dollar	98
GDP Per Capita	World Bank	Y	Current US Dollar	98
GDP (QoQ)	Investing.com	Q	Growth QoQ	36.2
GDP (YoY)	Investing.com	Q	Growth YoY	39.6
3. Labour				
Unemployment Rate	OECD	M	Level	53.6
Unemployment Rate	OECD	Q	Level	58.2
Population	World Bank	Y	Count	98
Unemployment Rate	Investing.com	M	Level	53.8
4. Prices				
Inflation Rate	OECD	M	Growth YoY	70.6
Inflation Rate	OECD	Q	Growth YoY	74.2
Inflation Rate MoM	OECD	M	Growth MoM	70
CPI	OECD	M	Index 2015 = 100	69.5
PPI Manufacture of food products	OECD	M	Index 2015 = 100	36.8
PPI Manufacturing	OECD	M	Index 2015 = 100	47.7

CPI (MoM) - Inflation Rate MoM	Investing.com	M	Growth MoM	61.8
CPI (YoY) - Inflation Rate	Investing.com	M	Growth YoY	80
PPI (MoM)	Investing.com	M	Growth MoM	26.2
PPI (YoY)	Investing.com	M	Growth YoY	35.6
5. Money				
Central Bank Rates	BIS	D	Level	
Overnight Interbank Rate	OECD	M	Level	73.7
Short Term Interest Rate	OECD	M	Level	68.4
Long Term Interest Rate	OECD	M	Level	66.4
Narrow Money M1	OECD	M	National Currency	49.5
Broad Money M3	OECD	M	National Currency	49.5
6. Trade				
Current Account to GDP	OECD	Q	Pct of GDP	70
Export of goods and services	OECD	Q	Growth YoY	70.3
Import of goods and services	OECD	Q	Growth YoY	70.3
Export - Value (goods)	OECD	M	US Dollars, monthly level	74.2
Import - Value (goods)	OECD	M	US Dollars, monthly level	74.4
Net Trade - Value (goods)	OECD	M	US Dollars, monthly level	74.2
Current Account Pct of GDP	World Bank	Y	Pct of GDP	91.4
Current Account	World Bank	Y	Current US Dollars	93.3
Trade Balance	Investing.com	M	Local Currency, monthly level	64.7
7. Government				
Government Debt to GDP	OECD	Y	Pct of GDP	54.8
Total Government Expenditure	OECD	Y	Current prices, local currency	61.9
Total Government Revenue	OECD	Y	Current prices, local currency	61.9
Government Budget	OECD	Y	Pct of GDP	77
Government Expense	World Bank	Y	Pct of GDP	77.8
8. Business				
OECD Business Confidence Indicator	OECD	M	Index, Amplitude adjusted	72.7
OECD Composite Leading Indicators	OECD	M	Index, Amplitude adjusted	34
Industrial Production	OECD	M	Growth YoY	60.6
Total Manufacturing	OECD	M	Index 2015=100, s.a.	66.7
Total Industry ex Construction	OECD	M	Index 2015=100, s.a.	60.3
Total Construction	OECD	M	Index 2015=100, s.a.	44.7
Changes in Inventories	OECD	Q	Current prices, local currency	62.3
Manufacturing PMI	Investing.com	M	Index	31.9
Services PMI	Investing.com	M	Index	13.8
Industrial Production	Investing.com	M	Growth YoY	42.3
9. Consumer				
OECD Consumer Confidence Indicator	OECD	M	Index, Amplitude adjusted	66.1
Private Consumption	OECD	Q	Growth YoY	67.7
Total Retail Sales Value	OECD	M	Index 2015=100, s.a.	52.7
Passenger Car Registration	OECD	M	Index 2015=100, s.a.	48.4
Permits Issued (Residential Buildings)	OECD	M	Index 2015=100, s.a.	42.6
Retail Sales	Investing.com	M	Growth YoY	46.8
Consumer Confidence	Investing.com	M	Index	33.9

3.2.2 Key Economic Indicators

Out of all indicators I have selected 6 most relevant measures to include in all models. Each of these indicators provides a different perspective on the health of the economy and can be used together to get a comprehensive understanding of economic conditions. Moreover, they are widely commented in both economic and general press after each release and are closely followed by traders and investors alike. Below are brief descriptions of each economic indicator:

1. ***GDP Annual Growth Rate***: This measures the percentage change in a country's Gross Domestic Product (GDP) over a year. It indicates whether an economy is expanding or contracting, serving as a critical indicator of its performance.
2. ***GDP Growth Rate:*** Like the annual growth rate, the GDP Growth Rate measures economic growth by comparing GDP from one period to the next, in my study over following quarters. It's an indicator of the health of an economy and helps policymakers adjust fiscal and monetary policy to achieve economic objectives.
3. ***Unemployment Rate***: This is the percentage of unemployed individuals in an economy among individuals currently in the labour force. It's calculated as the number of unemployed individuals divided by the total labour force, multiplied by 100. It's a key economic indicator as it signals the ability (or inability) of workers to obtain gainful work and contribute to the productive output of the economy.
4. ***Inflation Rate***: The Inflation Rate is the percentage increase in price for a basket of goods and services for a particular period. It's used to measure the general increase in the cost of goods and services.
5. ***Inflation Rate MoM (Month over Month)***: This measures the change in the Inflation Rate from one month to the next. It provides a more granular view of inflation trends and can be useful for identifying short-term changes in the rate of inflation.
6. ***Manufacturing PMI (Purchasing Managers' Index):*** The PMI is an index of the prevailing direction of economic trends in the manufacturing sector. It consists of a diffusion index that summarizes whether market conditions are expanding, staying the same, or contracting as viewed by purchasing managers. It's based on five major survey areas—each of which is weighted equally: New Orders, Inventory Levels, Production, Supplier Deliveries, and Employment.

3.2.3 Additional Economic Indicators

For a more comprehensive macroeconomic analysis, additional nine indicators have been selected that predominantly exhibit high data coverage and a monthly data frequency.

Out of all downloaded indicators, firstly, I have excluded yearly indicators from the selection. Indicators with an annual frequency tend to offer limited value for an investment algorithm that operates with monthly rebalancing. This is primarily due to the substantial delay, often up to two years, in their release. Consequently, when these are combined with monthly indicators, they frequently retain the same value over a six to nine-month rolling window, providing little information on the general trend.

Secondly, indicators with low data coverage were excluded as their utility is inherently restricted. The incorporation of such indicators would necessitate a significant amount of estimated or interpolated data, which could potentially introduce considerable noise into the model.

Additional indicators selected for extra analysis are:

1. ***Producer Price Index (PPI):*** This is a measure of the average change over time in the selling prices received by domestic producers for their output. It is a measure of inflation at the wholesale level.
2. ***Central Bank Rate:*** This is the interest rate set by the central bank of a country that commercial banks are charged to borrow money. It is a key tool used by central banks to implement monetary policy and control inflation.
3. ***Short Term Interest Rate:*** This refers to the interest rates on financial instruments that mature in less than one year. They are generally averages of daily rates and are based on three-month money market rates where available.
4. ***Long Term Interest Rate:*** This refers to the interest rates on financial instruments that mature in ten years or more. These rates are mainly determined by the price charged by the lender, the risk from the borrower, and the fall in the capital value.
5. ***Current Account to GDP***: This ratio provides an indication of a country's level of international competitiveness. A positive current account balance indicates that the nation is a net lender to the rest of the world, while a negative current account balance indicates that it is a net borrower.
6. ***Total Manufacturing:*** This refers to the total output of the manufacturing sector in an economy. It includes the production of goods and services across all manufacturing industries.
7. ***Industrial Production:*** This is a measure of the real output of the manufacturing, mining, electric, and gas industries. It is a key economic indicator that measures the level of production in the industrial sector of an economy.
8. ***OECD Consumer Confidence Indicator (CCI):*** This indicator 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.
9. ***Retail Sales***: Retail sales refer to the total amount of goods and services sold by retailers to consumers within a specific period. It is a vital economic indicator as it reflects consumer spending patterns, which account for a significant portion of overall economic activity. As a leading macroeconomic indicator, healthy retail sales figures typically elicit positive movements in equity markets.

3.2.4 Investment Clock

The investment clock is a conceptual framework used to illustrate the cyclical nature of economic activity and its impact on various asset classes. It was introduced by Merril Lynch in his research report in 2004 (Lynch, 2004). It segments the economic cycle into four distinct phases: recovery, overheat, stagflation, and recession. Each phase corresponds to specific investment opportunities and risks, guiding investors in their asset allocation decisions.

During recovery, equities tend to perform well as economic growth resumes; in overheat, commodities and real estate may flourish due to increased demand; stagflation is characterized by rising interest rates, benefiting cash and defensive stocks; and reflation sees bonds typically outperform as economic activity contracts. By aligning investment strategies with the appropriate phase of the economic cycle, the investment clock helps in optimizing portfolio performance.

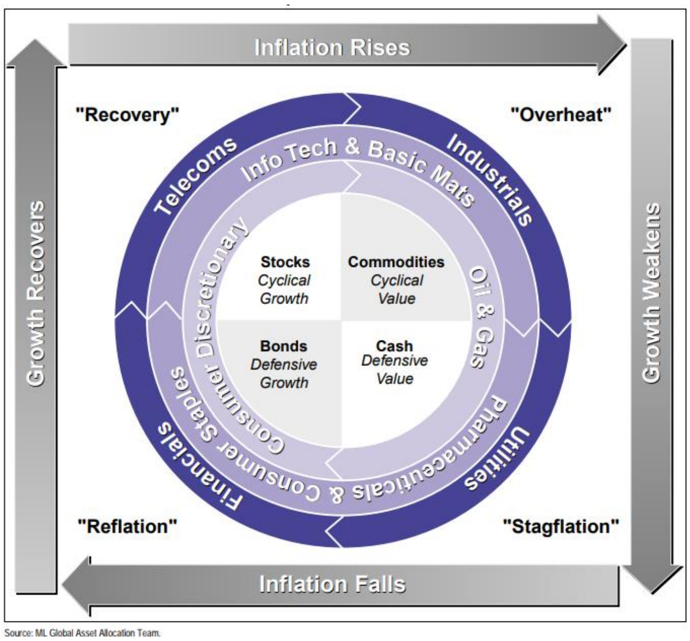

Figure 3 Investment Clock

Economic cycles can be estimated with a relatively simple model using Manufacturing Purchasing Manager' Index or OECD Business Confidence Indicators. These indices summarize market conditions by compiling results from a monthly survey of supply chain managers across different industries. While individual view can be completely wrong, combining hundreds of opinions gives relatively accurate outlook on market conditions.

A simple model takes relative value to the base level on the X axis. That base value is 50 for Manufacturing PMI and 100 for Business Confidence Indicator. On the Y axis, we take values of the first derivative, being a difference compared to the previous month. Additionally, to smooth out values and reduce noise, we can use exponential moving average on the index before calculating x and y values.

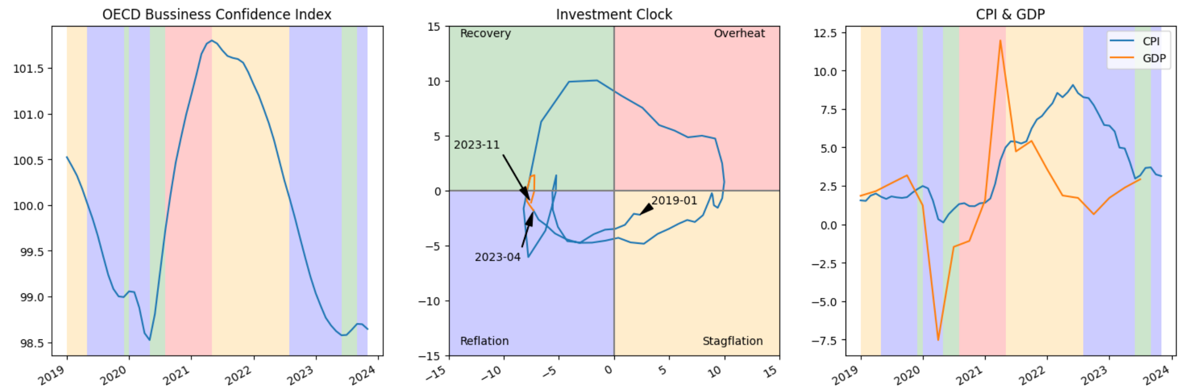

Figure 4 Investment Clock for United States, 2019-2023

A sample calculation, showing a complete economic cycle of United States in years 2019 to 2023 are presented on the figure above. On the left I have included input indicator, OECD Business Confidence Index in this case, and on the right CPI and GDP, showing inflation and economic growth with clear correlation to investment cycles. Additional colour coding helps to locate cycles on all three plots. When comparing results to the original diagram from Merrill Lynch report, we can confirm gross domestic product rising after COVID from recovery to the end of overheat phase. While consumer price index, representing inflation, peaks at the transition from stagflation to reflation phase.

In this thesis, my first idea was to capture economic cycles in selected countries and assign more weight to countries being in recovery and expansion, and less weight to countries being still in recession or entering contraction. Unfortunately, there are many difficulties in applying this strategy. Firstly, economic indicators are released with significant delay and investment cycles are often clearly seen only in retrospect. The best example are recessions, which are declared with many months of delay, and in some cases even after they have already ended. Secondly, the progression is far from lineal. There can be many false entries into the new phase with reversals to the previous phase in the next month or two.

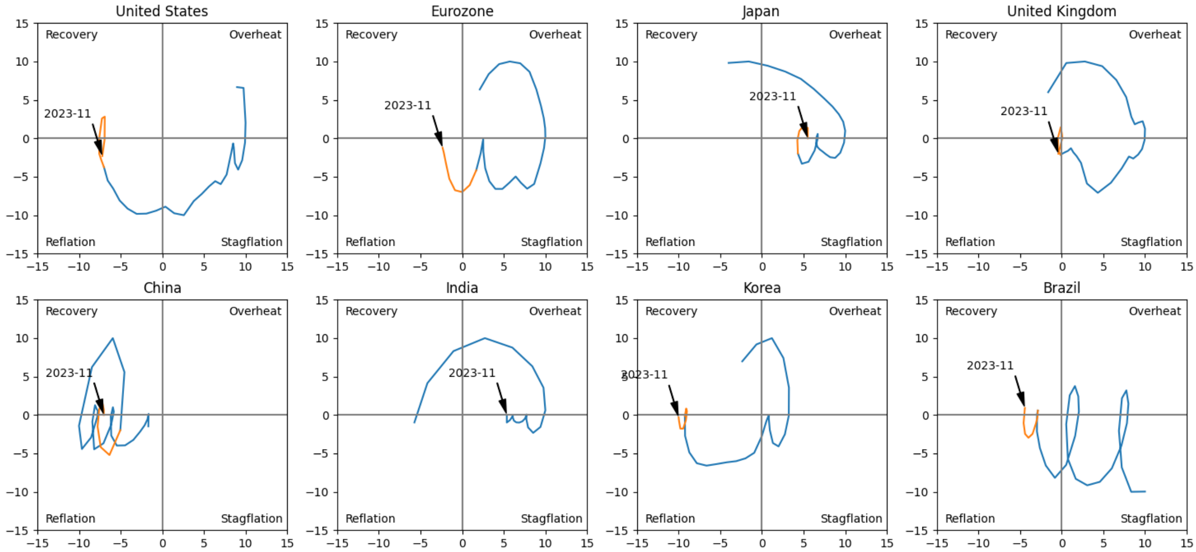

Figure 5 Investment Clock for Selected Countries, 2021-2023.

In the figure 5, presented above we can see various issues with that indicator when it is not following a circular trajectory. The most extreme examples are visible on charts for China and Brazil where there are numerous jumps back and forth between different phases of the cycle.

Investment clock indicator, passed as a pair of coordinates (x, y), was examined when evaluating different inputs to the algorithm. However, it seemed to introduce more noise than relevant information, negatively impacting results. In the end, it has not been included in final models.

Finally, the investment clock model could be potentially improved by including other indicators in like industrial production, credits, profits, inventories, and sales. That would require additional research and could be included in the future work.

3.3 Benchmark

MSCI All Country World Index (ACWI) has been selected as a benchmark for all designed algorithms. A more detailed description of that index has already been provided in chapter 2.3. In this section I would like to focus on the weights of different countries, its inclusions, and exclusions from the index and finally a selection of the countries for AI models.

Exact geographical breakdown of the index by country and region can be found on MSCI ACWI official website (MSCI, 2024). The figure presented below shows percentage weights of all countries and their changes over the past 11 years.

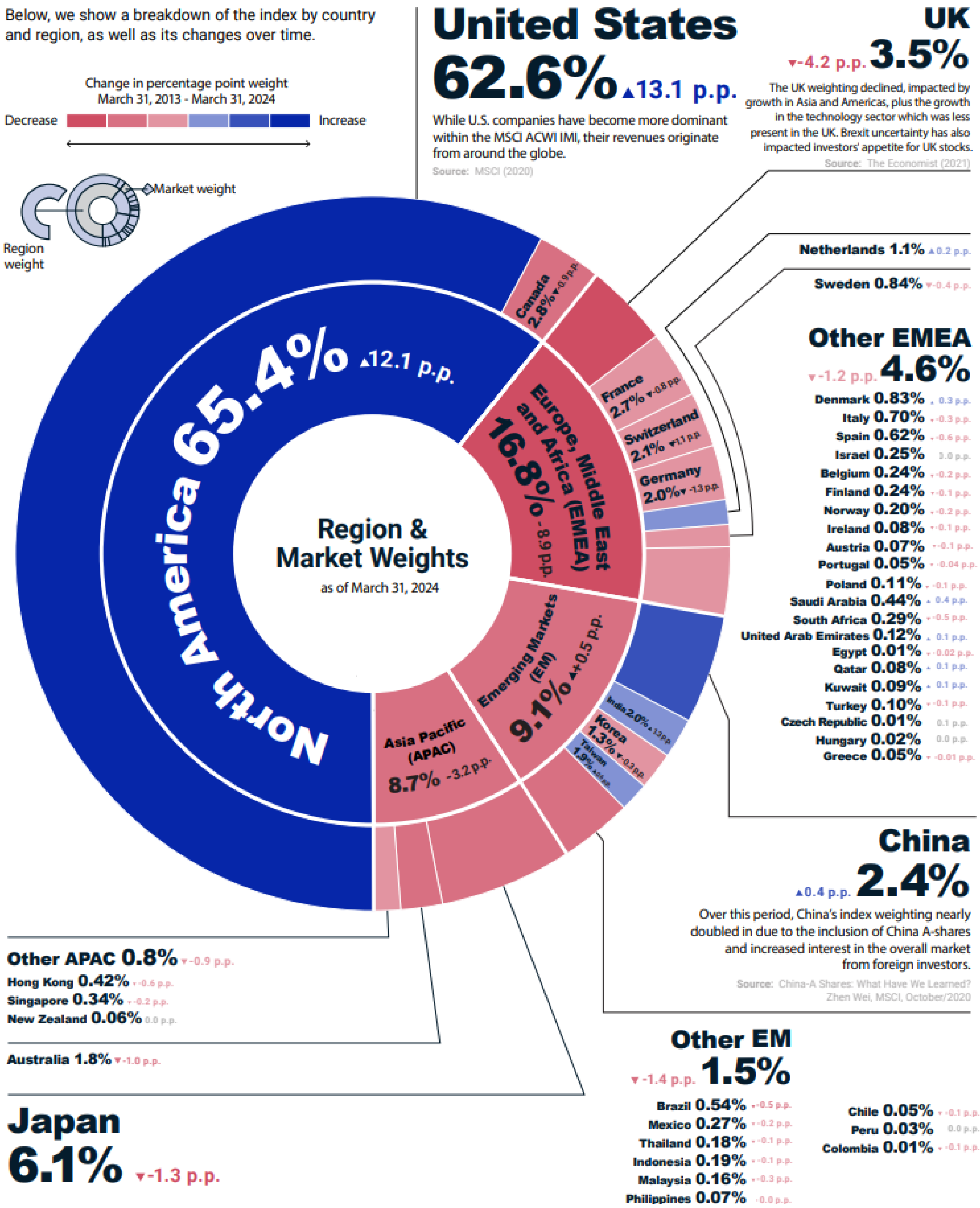

Figure 6 MSCI ACWI Complete Geographic Breakdown

3.3.1 MSCI ACWI Inclusions and Exclusions

To be able to perform backtesting correctly we cannot use current list of countries included in the index because that would give an algorithm unfair advantage using survivorship bias. Instead, we need to investigate all inclusions and exclusions from the index over the years.

All 23 developed economies have been present in the index since 1999. The only change has been an upgrade for Israel, Portugal, and Greece from emerging economies to developed ones.

However, there have been significant changes to emerging markets over the last 25 years. During my research I found an analysis prepared by MSCI for the Norwegian Ministry of Finance from October 2019 provides detailed additions to MSCI Emerging Markets (MSCI, 2019). That paper includes initial additions to MSCI Emerging Markets Index since 1988 when the index was created.

1988	1989	1992	1993	1995	1996	1997	2001
Argentina	Indonesia	South Korea	Colombia	Israel	China	Russia	Egypt
Brazil	Turkey		India	Poland	Czech Rep.	Portugal	Morocco
Chile			Pakistan	South Africa	Hungary		
Greece			Peru		Taiwan		
Jordan			Sri Lanka				
Malaysia			Venezuela[2]				
Mexico							
Philippines							
Portugal							
Thailand							

Figure 7 Initial Additions to MSCI Emerging Markets Index

Apart from these initial inclusions, there have been additional removals, additions, and reinstallations:

- Sri Lanka was removed in 2001.
- Venezuela was removed in 2006.
- Pakistan and Jordan were removed in 2009.
- Argentina was removed in 2010.
- United Arab Emirates and Qatar were added in 2014.
- Pakistan was re-instated in 2017.
- Argentina was re-instated in 2019.
- Saudi Arabia was added in 2019.
- Kuwait was added in 2020.
- Russia, Argentina, and Pakistan were removed in 2022.

3.3.2 MSCI ACWI Country Weights Changes

Country weights are changing constantly due to market capitalization changes, addition or removal of constituents, quarterly rebalancing of the index, and changes in free float. In the investment algorithm, the exact weights within the index are not required, but approximate values are used in portfolio optimization used as targets. Their usage will be described in detail in Target values for the base model in section 4.2.2.

Historical weights for all countries aren't publicly available on the MSCI website. However, during my research I have come across a study on emerging market portfolio strategies, where it lists composition of the MSCI ACWI in years 1987-2012. (Roberto Violi, 2018).

Combining weights from the research paper (Roberto Violi, 2018), MSCI report (MSCI, 2019) and MSCI geographical breakdown (MSCI, 2024), I have been able to approximate weights all countries in period 1998 to 2024.

Table 2 MSCI ACWI Country Weights Changes 1998 - 2024

MSCI World:	1998	2003	2008	2013	2021	2024
United States	46.56	53.99	41.80	47.45	58.60	62.60
Japan	11.26	8.44	8.60	7.50	6.20	6.10
United Kingdom	10.40	11.50	9.60	7.90	4.00	3.50
Canada	2.29	2.20	3.67	4.30	2.90	2.80
France	4.10	3.10	4.20	3.10	2.70	2.70
Switzerland	3.40	3.10	4.10	3.00	2.50	2.10
Germany	4.20	2.60	3.70	2.90	2.30	2.00
Australia	1.22	1.88	2.79	3.31	1.90	1.80
Netherlands	1.50	1.60	1.30	0.80	1.10	1.10
Sweden	1.20	1.20	1.30	1.20	1.20	0.84
Spain	1.20	1.50	1.90	0.96	0.61	0.62
Hong Kong	1.25	0.64	1.05	1.09	0.79	0.42
Italy	1.90	1.80	1.50	0.79	0.67	0.70
Singapore	0.41	0.33	0.48	0.71	0.32	0.34
Denmark	0.30	0.30	0.40	0.37	0.68	0.83
Finland	0.45	0.45	0.40	0.27	0.33	0.24
Belgium	0.30	0.50	0.50	0.49	0.27	0.24
Norway	0.40	0.40	0.60	0.34	0.24	0.20
Israel	0.17	0.13	0.24	0.21	0.26	0.25
Ireland	0.10	0.10	0.20	0.19	0.18	0.08
New Zealand	0.13	0.07	0.06	0.05	0.09	0.06
Austria	0.10	0.10	0.10	0.08	0.08	0.07
Portugal	0.10	0.10	0.10	0.06	0.05	0.05
MSCI EM:						
China	0.03	0.26	1.80	2.23	3.80	2.40
Taiwan	1.14	0.51	1.12	1.34	1.80	1.90
India	0.40	0.20	0.94	0.79	1.40	2.00
Korea	0.20	0.86	1.62	1.94	1.70	1.30
Brazil	1.02	0.27	1.51	1.64	0.65	0.54
Russia	0.37	0.19	1.15	0.76	0.38	0.00
South Africa	0.67	0.56	0.76	1.01	0.44	0.29
Mexico	0.81	0.31	0.51	0.63	0.23	0.27
Malaysia	0.36	0.22	0.28	0.46	0.18	0.16
Thailand	0.10	0.07	0.15	0.27	0.22	0.18
Indonesia	0.11	0.04	0.19	0.36	0.14	0.19
Türkiye	0.20	0.05	0.19	0.21	0.05	0.10
Poland	0.03	0.05	0.19	0.17	0.10	0.11
Chile	0.25	0.06	0.13	0.24	0.06	0.05
Argentina	0.29	0.02	0.05	0.00	0.02	0.00
Saudi Arabia	0.00	0.00	0.00	0.00	0.36	0.44
Greece	0.17	0.00	0.00	0.00	0.03	0.05
United Arab Emirates	0.00	0.00	0.00	0.00	0.09	0.12
Qatar	0.00	0.00	0.00	0.00	0.08	0.08

Kuwait	0.00	0.00	0.00	0.00	0.07	0.09
Philippines	0.09	0.02	0.06	0.12	0.07	0.07
Peru	0.08	0.02	0.07	0.08	0.02	0.03
Hungary	0.08	0.05	0.09	0.04	0.03	0.02
Czechia	0.06	0.02	0.09	0.04	0.01	0.01
Egypt	0.00	0.01	0.09	0.04	0.01	0.01
Colombia	0.06	0.00	0.03	0.16	0.02	0.01
Pakistan	0.05	0.01	0.02	0.00	0.01	0.00
Jordan	0.01	0.01	0.01	0.00	0.00	0.00
Sri Lanka	0.01	0.00	0.00	0.00	0.00	0.00
Venezuela	0.10	0.01	0.00	0.00	0.00	0.00
Marocco	0.00	0.01	0.03	0.01	0.00	0.00

3.3.3 Countries Selection

Over the past 25 years, the MSCI ACWI index has included 54 countries. However, incorporating all these countries into the portfolio could introduce excessive complexity to the model while providing minimal impact for countries with negligible weight.

Upon analysis of available macroeconomic data and historical ETF data, a decision has been made to use a 0.03% cutoff index weight to determine which countries would be included in the algorithm. This approach reduced the number of countries by half, to 27, while still covering over 98% of the total MSCI ACWI index. This effectively excluded all countries that only contribute a fraction of a percentage to the index and were also absent from the OECD database, thus lacking most of the macroeconomic indicators.

The model will include 18 developed economies and 9 emerging markets:

- MSCI World: United States, Japan, United Kingdom, Canada, France, Switzerland, Germany, Australia, Netherlands, Sweden, Spain, Hong Kong, Italy, Singapore, Denmark, Finland, Belgium, Norway
- MSCI Emerging Markets: China, Taiwan, India, Korea, Brazil, Russia, South Africa, Mexico, Malaysia

3.4 INVESTMENT INSTRUMENTS

There exist various investment instruments that can be utilized to construct a global stock portfolio, representing markets in specific countries. Given the selection of 27 countries for the investment algorithm, exposure to the main stock index is necessary, as purchasing individual stocks for numerous companies is not feasible with monthly rebalancing. There are three primary options to replicate the returns of a stock market index:

1. Derivatives, particularly futures: These are primarily used for risk hedging. However, in the context of investment algorithms, they do not provide the required flexibility due to their fixed settlement day and high unit price.
2. Index Funds: These have substantial coverage for major economies, but there is a lack of funds for smaller countries. Furthermore, it can take up to five days to transfer funds from one fund to another during rebalancing, potentially resulting in lost gains due to being out of the market. Nonetheless, in certain countries, such as Spain and Poland, they offer tax advantages for retail investors, eliminating the need to pay capital gains tax when transferring money from one fund to another.

3. Exchange-Traded Funds (ETFs): These funds are traded on stock exchanges. They offer high liquidity and instant market orders, similar to regular stocks. They have been selected as the preferred option for the investment algorithms in this thesis.

3.4.1 Selected ETF Data

In order to conduct the most reliable back-testing possible, the decision has been made to utilize historical data spanning from 1999 to the end of 2023. This encompasses a period of 25 years and nearly three full economic cycles. The inclusion of historical data from various stages of the cycle in both the training and testing phases is crucial to ensure the development of a reliable AI model that is not biased towards any specific part of the cycle.

Upon investigation of available Exchange-Traded Funds (ETFs), it was observed that the US markets offer the longest price history. This is because ETFs emerged in the United States in the early 1990s, while in Europe, the first ETFs were introduced in 2005 and only became widely available after 2008. To obtain market data from 1999, ETFs from the Nasdaq stock exchanges have been selected, with BlackRock offering the broadest range for the selected countries and the highest liquidity.

Below I present a list of ETFs downloaded from Yahoo Finance and used in this research.

Table 3 Selected Exchange Trades Funds (ETFs)

Country	Symbol	Full Name	Year Created
United States	SPY	SPDR S&P 500 ETF Trust (SPY)	1999
Japan	EWJ	iShares MSCI Japan ETF	1999
United Kingdom	EWU	iShares MSCI United Kingdom ETF	1999
Canada	EWC	iShares MSCI Canada ETF	1999
France	EWQ	iShares MSCI France ETF	1999
Switzerland	EWL	iShares MSCI Switzerland ETF	1999
Germany	EWG	iShares MSCI Germany ETF	1999
Australia	EWA	iShares MSCI Australia ETF	1999
Netherlands	EWN	iShares MSCI Netherlands ETF	1999
Sweden	EWD	iShares MSCI Sweden ETF	1999
Spain	EWP	iShares MSCI Spain ETF	1999
Hong Kong	EWH	iShares MSCI Hong Kong ETF	1999
Italy	EWI	iShares MSCI Italy ETF	1999
Singapore	EWS	iShares MSCI Singapore ETF	1999
Denmark	EDEN	iShares MSCI Denmark ETF	2012
Finland	EFNL	iShares MSCI Finland ETF	2012
Belgium	EWK	iShares MSCI Belgium ETF	1999
Norway	NORW	iShares MSCI Norway ETF	2012
China	MCHI	iShares MSCI China ETF	2011
Taiwan	EWT	iShares MSCI Taiwan ETF	2000
India	INDA	iShares MSCI India ETF	2012
Korea	EWY	iShares MSCI South Korea ETF	2000
Brazil	EWZ	iShares MSCI Brazil ETF	2000
Russia	ERUS	iShares MSCI Russia ETF	2010
South Africa	EZA	iShares MSCI South Africa ETF	2003
Mexico	EWW	iShares MSCI Mexico ETF	1999
Malaysia	EWM	iShares MSCI Malaysia ETF	1999

3.4.2 Synthetic ETF Prices

Unfortunately, not all ETFs have full historic prices since 1999. Countries like Denmark, Finland or India has prices starting from 2012 when these funds were launched. To be able to use these instruments in my algorithm I had to recreate ETFs for missing periods to have full history for all selected countries. These missing periods can by recreated using MSCI indexes or main stock indexes from given country with currency exchange rate applied.

My first approach has been to extrapolate index data using either min-max and Z-score normalization and rescaling. However, generated prices have been distorted, not representing correctly returns from underlying indexes.

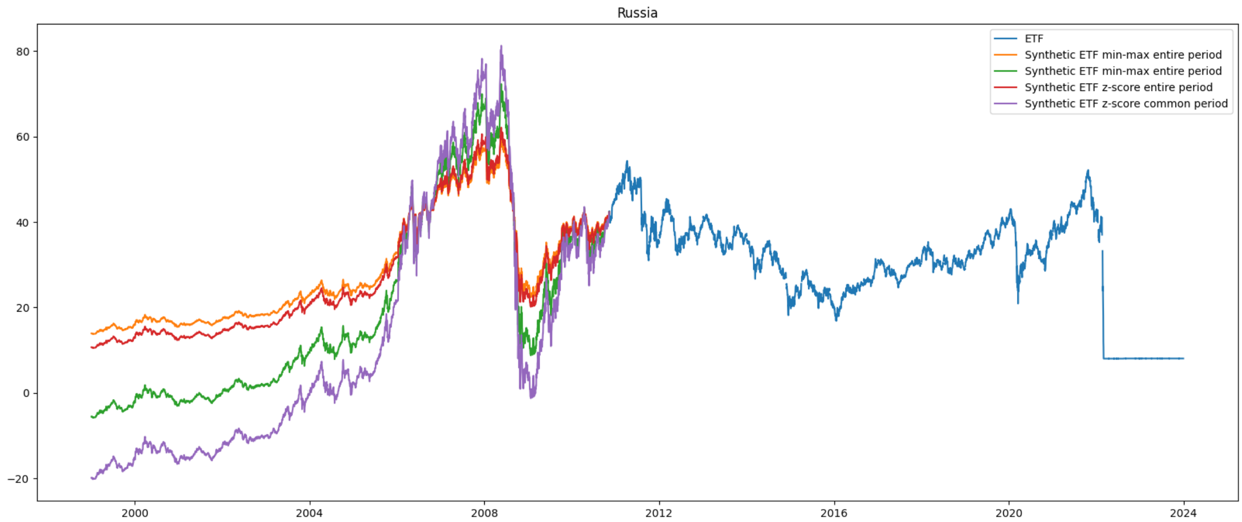

Figure 8 Synthetic ETFs using Different Normalization and Rescaling Methods.

In some cases, like in the figure above, prices were reaching impossible, negative values. To prevent that from happening and more importantly achieve the same relative returns, I have used reverse returns from indexes. That would start from the first date available ETFs and going backwards till start of 1999, applying cumulative product for reverse returns multiplying by the first price. This method has proven much more reliable, resulting in exact representation of underlying indexes but converted to real ETF prices.

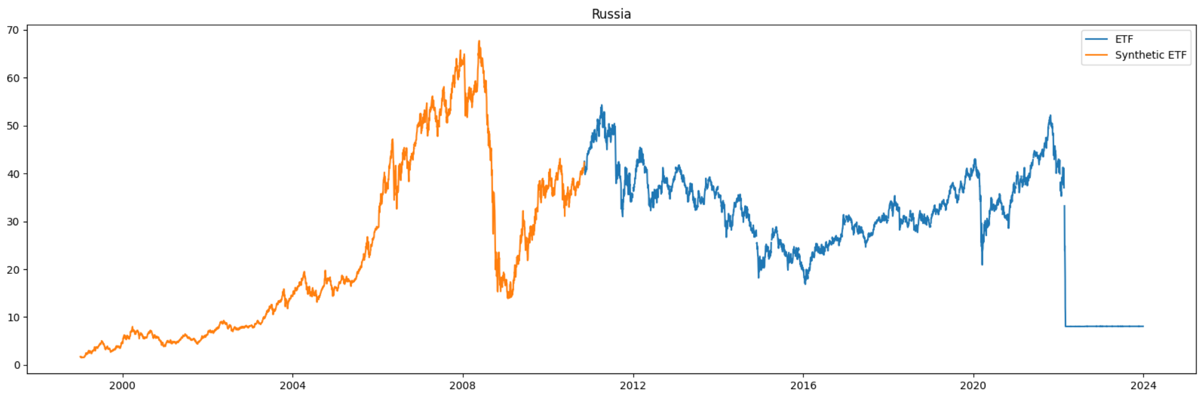

Figure 9 Synthetic ETF Prices Using Reverse Returns.

Using calculated synthetic ETFs together with the actual ones from Nasdaq stock exchange I have been able to calculate returns for all individual countries over the period of 25 years.

3.4.3 Monthly Returns

Apart from macroeconomic indicators, described earlier, the final input to be included in the model are monthly returns for the ETFs. Traditionally it was used for better diversification insights, helping to understand the correlation between different assets. In this research, adding returns alongside macroeconomic indicators should enhance optimization results by identifying historical trends and detect volatility patterns. Moreover, asset returns provide immediate response to market conditions, without a delay present in other macro indicators.

3.5 DATA PREPARATION

Data preparation plays a pivotal role in the success of machine learning models. It involves several critical tasks, including data cleaning, feature engineering, and handling imbalanced datasets. By ensuring that the data is well-structured and relevant, we enhance the accuracy and reliability of our models. Properly prepared data serves as the foundation for robust and effective machine learning solutions.

In this research there are several steps required to prepare downloaded indicators into correct input data ready for ML training. First, missing values must be filled while introducing minimal noise. Next, correct values must be selected for each rebalance date. That's because we cannot use indicators released in later date in relation to a given training step. Finally, indicators must be normalized, and additionally they can be factorized to reduce dimensionality.

3.5.1 Filling Missing Values

One of the principal problems with open data sources is much higher percentage of missing values in comparison to paid providers. In my research I have managed to combine values from Investing.com and OECD to reduce number of NAs for most key indicators available in both sources. The best example is an inflation rate where both sources have missing values for different countries and combining them has resulted in almost complete dataset.

Table 4 Missing Values in Data Sources for Inflation Rate YoY

	Inflation Rate (Investing)	**Inflation Rate (OECD)**	**Inflation Rate (Combined)**
United States	0	0	0
Japan	6	30	0
United Kingdom	1	0	0
Canada	2	0	0
France	230	0	0
Switzerland	0	0	0
Germany	0	0	0
Australia	257	200	157
Netherlands	0	0	0
Sweden	6	0	0
Hong Kong	1	300	1
Spain	0	0	0
Italy	2	0	0
Singapore	1	300	1
Denmark	0	0	0
Finland	144	0	0
Belgium	2	0	0

Norway	0	0	0
China	0	0	0
Taiwan	0	300	0
India	160	0	0
Korea	3	0	0
Brazil	0	0	0
Russia	1	21	0
South Africa	1	0	0
Mexico	0	0	0
Malaysia	1	300	1

Unfortunately for other key indicators there are still many missing values. Below is the summary of missing values count for key macroeconomic indicators.

Table 5 Missing Values for Key Macroeconomic Indicators

	GDP Annual Growth Rate (Combined)	**GDP Growth Rate (Combined)**	**Unemployment Rate (Combined)**	**Inflation Rate (Combined)**	**Inflation Rate MoM (Combined)**	**Manufacturing PMI (Investing)**
United States	0	0	0	0	0	160
Japan	0	0	0	0	5	109
United Kingdom	0	0	0	0	0	109
Canada	0	0	0	0	0	263
France	0	0	0	0	0	115
Switzerland	3	3	0	0	0	0
Germany	0	0	0	0	0	109
Australia	3	3	0	157	300	239
Netherlands	0	0	0	0	0	300
Sweden	0	0	0	0	0	0
Hong Kong	177	129	0	1	292	161
Spain	0	0	0	0	0	157
Italy	0	0	0	0	0	161
Singapore	141	180	239	1	292	300
Denmark	0	0	0	0	0	290
Finland	0	3	0	0	0	300
Belgium	0	0	0	0	0	300
Norway	3	0	0	0	0	63
China	0	144	238	0	0	75
Taiwan	129	300	2	0	0	161
India	3	3	300	0	0	159
Korea	0	0	0	0	0	167
Brazil	3	3	55	0	0	161
Russia	63	78	3	0	0	162
South Africa	3	3	253	0	0	174
Mexico	0	0	0	0	0	161
Malaysia	144	300	190	1	293	300

There are various strategies to fill missing values. The simplest one is to fill with 0 or -1 values. Other would be to use mean or median values. Finally, for more advanced methods, we can use predictive models or interpolation.

For selected indicators, there are certain countries not included in the OECD, like Singapore, Hong Kong, Taiwan and Malaysia, which tend to have more missing values. In this case, for the most accurate approximation, I have decided to use mean values of the 5 most correlated countries. That should give much more precise approximation than taking a mean value of all countries.

Correlation values are calculated on the ETFs returns for the last 12 months for a given date with missing values. For example, when filling missing values for May 2002, we can only use correlations calculated on returns till that date. We cannot use single calculation of correlations based on the entire 25-year period to fill all missing values at once. That's because in many instances we would be using information from future dates. Looking forward is one of the most common mistakes made when back testing investment algorithms that distorts results and achieve unrealistic performances.

3.5.2 Calculating Available Indicators Data

When working with macroeconomic indicators we must be very careful about release dates and consecutive revisions. We cannot calculate portfolio weights using indicators released in future dates or using revisions to original values that were updated later. We can only use indicator values available at the time of rebalancing.

In my research I have examined release dates for all selected indicators and used values only after the last revisions for all countries. That introduces certain delay for some indicators, having to wait for all countries to release their data, but it assures the model only uses available data.

Selected 15 indicators have different release schedule that must be considered:

- Quarterly indicators - GDP Annual Growth Rate, GDP Growth Rate and Current Account to GDP have the most delay of all indicators. They have up to 3 releases, with 2 provisional readings and one final value. For example, for the first quarter, we have releases at the end of April, at the end of May and the final one at the end of June. That way we can only use data from the first quarter starting from July, having the final values. Moreover, these quarterly values need to be converted into monthly format, using ffill method, to be compatible with other monthly indicators.
- Most monthly indicators like Unemployment Rate, Inflation Rate, OECD Business and Consumer Confidence Indicators, Short- and Long-Term Interest Rates provided by OECD, Total Manufacturing and Producer Price Index are released during the following month after the given month ends. As the rebalancing is done at the beginning of each month, we need to use 2-month delay. For example, in March we can use values from January, but not from February that haven't been released at that time.
- Manufacturing PMI has very little delay, being released at the beginning of the following month. On the 4th or 5th, we already have data for all countries from previous month. This indicator is one of the best predictors for economic cycle and its release dates have been selected for algorithm rebalancing.
- Central Bank Rates provided by Bank of International Settlement has daily values with minimal delay. In the algorithm, the most recent values can be used.
- Industrial Production and Retail Sales are taking more time to collect all relevant data. Their values are released up to 50 days after given month ends. That gives 3-month delay, one month more than most other monthly indicators. In this case, we can use data from January only starting from April.

3.5.3 Formatting Input Data

Having filled missing values and calculated available indicators for the given date, the next step is to format input data for machine learning models and neural networks. First, all indicators must be normalized to be on the same scale. Normalizing the data often leads to improved model performance by reducing the impact of outliers and extreme values. Moreover, it fosters stability in the optimization process, promoting faster convergence of the models during training.

Normalized indicators can be passed separately, concatenating all indicators for selected countries. However, with this approach input data will have significant size. When using all 15 selected indicators for 27 countries multiplied by 6 months of data, it results in 405 columns by 6 rows. In flatten version, required by machine learning models or simple deep neural network has 2430 inputs. With monthly rebalancing there are relatively few data points to train having in total 300 months for 25 years minus periods reserved for testing. Having that many inputs with so few training examples, the model might not be able to learn all complex relationships from macroeconomic analysis.

One solution to excessive input size is dimensionality reduction which aims to reduce the number of input features while preserving essential information. Reduced dimensions lead to faster training times without significant information loss. The easiest way to do that is to take either equal or fixed weights applied to all indicators to create a new single composite indicator with shape of 27x6 instead of 405x6.

Another, more complex method is to use Principal Component Analysis (PCA). PCA transforms the original features into a new set of uncorrelated features (principal components). It captures the most significant variance in the data, allowing us to represent it in a lower-dimensional space.

In my research I have examined different input formatting, from separate indicators, simple complex indicator to various number principal components. More details on the testing of various inputs and achieved results will be presented in the next chapter.

3.5.4 Splitting Training Data

Consistent with established practices, the input dataset has been partitioned into an 80% training set and a 20% test set. Additionally, for neural networks, an additional 10% has been reserved for validation purposes. Furthermore, the initial 12 months from 1999 have been excluded from the training set to account for the requisite initial knowledge needed for the first rebalancing, also called warm-up steps.

In summary, total 25-year period is divided into following date ranges:

- 01/1999 – 12/1999 – reserved for warm-up steps
- 01/2000 – 09/2016 – training data
- 10/2016 – 02/2019 – validation data
- 03/2019 – 12/2023 – test data

With broad economic periods being selected for both training and testing, more realistic models can be trained, not skewed toward any market conditions. Moreover, testing models through different stages of economic cycles will help to show the performance of the investment algorithm in different scenarios.

4 PRESENTATION OF WORK

This chapter presents the comprehensive body of work undertaken in this thesis. It begins with a detailed overview of the methodology employed, outlining the objectives and constraints, data collection and preparation, model selection, training, validation, backtesting, and the eventual deployment, combined with monitoring processes.

The chapter then delves into the specifics of the base model, discussing the preparation of features and targets, the definition of the model, and the processes of training, validation, and backtesting. This provides a clear understanding of the foundational model upon which the study is built.

The chapter then explores the evaluation of different inputs and outputs, providing insights into how variations in these factors can impact the performance of the model. This is followed by an exploration of different Neural Network Models, including Convolutional Neural Networks and Recurrent Networks, as well as Machine Learning Models, specifically the Random Forest Regressor and XG Boosting. This provides a comparative analysis of different model types and their performance.

The chapter then introduces the concept of 'Learning to Rate' and describes a different approach to portfolio optimization problem. The next section of the chapter focuses on the tuning of hyperparameters, a critical step in optimizing the performance of the models.

Next, classical models are backtested for comparison. The chapter ends with an examination of data augmentation and an analysis of a faulty results encountered during the research.

4.1 METHODOLOGY

Constructing an investment algorithm using Artificial Intelligence involves several critical steps, each incorporating a blend of financial theory, data science, and machine learning techniques.

4.1.1 Objectives and Constraints

First step is to define objectives and constraints. As stated in the introduction, in this thesis I would like to implement an investment algorithm that could achieve both superior absolute returns and risk-adjusted returns in comparison to MSCI All Country World Index (ACWI) benchmark.

There are three main constraints identified. The primary constraint is country selection that should correspond to the selected benchmark. In my case I have selected 27 countries that surpassed 0.3% threshold weight in MSCI ACWI index. In total they are covering between 98 and 99% of the benchmark. The second constraint would be to use long positions only and do not utilize any additional leverage. Finally, the last one is to allocate all available capital at each rebalancing without retaining any cash reserves.

4.1.2 Data Collection and Preparation

Data collection and preparation constitute the foundational steps in constructing an investment algorithm, involving a meticulous process of gathering comprehensive financial and macroeconomic data sources. Historical financial data, including ETFs prices, indices, and economic indicators, form the core dataset to capture broad market dynamics. The preparation phase entails rigorous data cleaning to address missing values, outliers, and noise, ensuring the integrity and reliability of the dataset. Subsequently, feature engineering is conducted to derive relevant features from the raw data and reduce excessive data dimensions when required, which are crucial for enhancing the predictive power of the algorithm.

This thorough and methodical approach to data collection and preparation ensures a robust foundation for the subsequent stages of model development and validation. As is common in most machine learning projects, this stage has constituted roughly half of the total time spent during my entire research period. Given its importance, an entire chapter, Chapter Three, is dedicated to this topic to describe it in sufficient detail.

4.1.3 Model Selection, Training and Validation

Having prepared the input data, the next stages involve model selection, training, and validation, which require a strategic approach to ensure accuracy and robustness. Initially, suitable machine learning models are chosen based on specific objectives. This thesis explores various machine learning ensemble methods, learning-to-rank model, and different neural network architectures.

During training, several substages are undertaken. Firstly, different inputs and outputs are investigated to determine which indicators carry the most relevant information and which dimensions yield optimal solutions. Secondly, various base machine learning models and neural networks are evaluated to select the most promising architectures. Finally, models are fine-tuned to determine the optimal number of months to employ, utilizing k-fold cross-validation, and hyperparameter tuning is performed through random search.

This comprehensive process of model selection, training, and validation ensures that the investment algorithm is well-calibrated to deliver reliable and stable performance in real-world trading scenarios.

4.1.4 Backtesting

Backtesting using historical simulation is a crucial method for evaluating the performance of an investment algorithm by applying it to historical market data. This process involves running the algorithm on past financial data to simulate how it would have performed in real trading conditions. The primary objective is to assess the algorithm's effectiveness, stability, and robustness over different market cycles, including periods of volatility and economic shifts.

To encompass all stages of the economic cycle, 20% of the data has been reserved for backtesting, covering the period from March 2019 to December 2023. In the backtesting process, various metrics such as cumulative returns, annual returns, Sharpe ratio, Sortino ratio, Calmar ratio, maximum drawdown, time under water, and information ratio have been calculated. These metrics facilitate a comprehensive comparison of the algorithm's performance against benchmarks and other models.

4.1.5 Deployment and Monitoring

Deployment of the investment algorithm involves a phased approach, beginning with paper trading and followed by continuous monitoring to ensure performance and stability. Paper trading, also known as simulated trading, allows the algorithm to be tested in a real-time market environment without financial risk. This stage involves executing trades based on current market conditions, thereby enabling the identification of any operational issues and the refinement of trading strategies. Additionally, continuous monitoring is performed together with iterative refinement process to maintain the algorithm's effectiveness and adapt to evolving market dynamics.

As a final step in this research, best performing model will be deployed using AWS Lambda function, implementing entire data pipeline, executing neural network model, and returning predicted portfolio weights for a current date.

4.2 BASE MODEL

As a base model in this research, a simple deep neural network model has been developed. It takes six key macroeconomic indicators as an input, uses a single intermediate dense layer, and has 27 outputs to represent weights in global portfolio. This section will present the entire process, including data preparation, calculation of target values, model definition, training, and backtesting. That will serve as the basic framework for all models developed in this research.

4.2.1 Features Preparation

Chapter three provides a detailed description of data collection and preparation. This section offers only a summary of the process.

Macroeconomic indicators are collected by six AWS Lambda functions and stored in a Microsoft SQL Server database, which is also hosted on AWS cloud. The first step involves downloading the indicators from the remote SQL database to local memory. Next, data from two principal sources, OECD and Investing.com, is combined. Missing values are then filled using mean values from the five most correlated countries. This process yields a complete dataset for all key indicators.

To avoid repeating this process multiple times, a custom caching mechanism has been implemented. This mechanism saves complete data frames to local CSV files, thus eliminating the need to access the database, calculate correlations, and fill missing values each time a model is trained. Instead, the clean dataset can be accessed from the local file.

Once all key indicators are ready, dates for monthly recalculations are read, and features for each month are prepared. For each indicator, the dataset is normalized and the latest six values available at the time of portfolio rebalancing are obtained. Finally, all six indicators are concatenated along the column axis.

The resulting array x has a shape of (288, 6, 162), representing 288 data points for all rebalancing periods over 24 years (12 months each), six rows for six months of historical data for each indicator, and 162 columns for six indicators multiplied by 27 countries.

4.2.2 Targets Preparation

The output from the model should consist of 27 decimal values, representing the weights for all selected countries in the portfolio. To train a model to predict the optimal portfolio, it is necessary to prepare training data with the desired values. In this research, maximum Sharpe ratio portfolios have been used, calculated for the subsequent period. For example, when calculating the optimal target on 04/01/2000, the returns for the period from 04/01/2000 to

04/02/2000 are taken to calculate the optimal, maximum Sharpe portfolio for that period, situated on the efficient frontier.

Calculations are performed using quadratic optimization to solve the problem of maximizing risk-adjusted returns. The optimization problem incorporates several constraints. It ensures that the sum of all weights equals 1 and then applies a permitted range of values for each individual country. As a reference, it uses the weight of the ACWI Index for a given year, allowing values between 50% and 200% of that weight. Additional rules are applied for outliers:

- For the United States, the maximum weight is restricted to 75%.
- For countries with an original weight under 1.5%, the maximum weight is increased to 3%.
- The minimum allowed weight for the smallest countries is 0.1%.

All weight restrictions, for selected countries in 2001, are shown in the following table.

Table 6 Sample weights restrictions for year 2001

	Ref ACWI Weight	Min Allowed Weight	Max Allowed Weight
United States	51.16%	25.58%	75.00%
Japan	9.59%	4.80%	19.18%
United Kingdom	11.09%	5.54%	22.18%
Canada	2.24%	1.12%	4.48%
France	3.51%	1.76%	7.02%
Switzerland	3.23%	1.62%	6.46%
Germany	3.25%	1.62%	6.50%
Australia	1.62%	0.81%	3.24%
Netherlands	1.56%	0.78%	3.12%
Sweden	1.20%	0.60%	3.00%
Hong Kong	0.89%	0.44%	3.00%
Spain	1.38%	0.69%	3.00%
Italy	1.85%	0.92%	3.70%
Singapore	0.36%	0.18%	3.00%
Denmark	0.30%	0.15%	3.00%
Finland	0.45%	0.22%	3.00%
Belgium	0.42%	0.21%	3.00%
Norway	0.40%	0.20%	3.00%
China	0.17%	0.10%	3.00%
Taiwan	0.76%	0.38%	3.00%
India	0.28%	0.14%	3.00%
Korea	0.60%	0.30%	3.00%
Brazil	0.57%	0.28%	3.00%
Russia	0.26%	0.13%	3.00%
South Africa	0.61%	0.31%	3.00%
Mexico	0.51%	0.26%	3.00%
Malaysia	0.28%	0.14%	3.00%

The final y array has a shape of (288, 27), representing targets for 288 data points and 27 columns for weights corresponding to all selected countries in the portfolio. The sum of each row equals 1.

4.2.3 Model Definition

For the base model, a simple deep neural network has been selected, consisting of three layers. The first layer formats the data into a one-dimensional array using a flatten layer. Other models that support sequences such as convolutional and recurrent networks, described later, will not require this formatting step.

The next part is a single hidden dense layer, also known as a fully connected layer, where each neuron is connected to every neuron in the preceding layer. This configuration enables the network to combine features learned in previous layers, capturing complex patterns in the data. Sixteen neurons in this layer are sufficient to achieve a low loss error, as more complex networks with additional trainable parameters tend to result in overfitting and less stable models. The ReLU activation method is used to identify non-linear relationships in the data.

The final layer is a dense layer with 27 neurons serving as the output. It uses SoftMax activation to ensure all outputs sum to 1. Although this activation method is more commonly used in classification problems, it works well in this scenario.

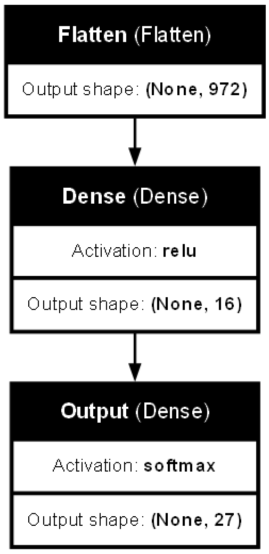

Figure 10 Base Deep Neural Network Model

Final model has a total of 16027 trainable parameters. Such a big number of parameters for this simple model is caused primarily by input shape of 972 elements in flatten format.

Model is compiled using Adam optimizer and use Mean Squared Error as a loss function.

4.2.4 Training and Validation

Model training is performed using 70% of input dataset with 10% used for validation during training and final 20% reserved for backtesting. The standard Adam optimizer converges quickly, typically within 20 epochs, and shows slight improvement when extended to a total of 100 epochs. The figure below also includes a second plot with a logarithmic scale that provides a clearer view of the finer details, especially at lower loss values. It reveals that after the initial rapid decrease, the loss values continue to decrease gradually with minor fluctuations, stabilizing as training progresses.

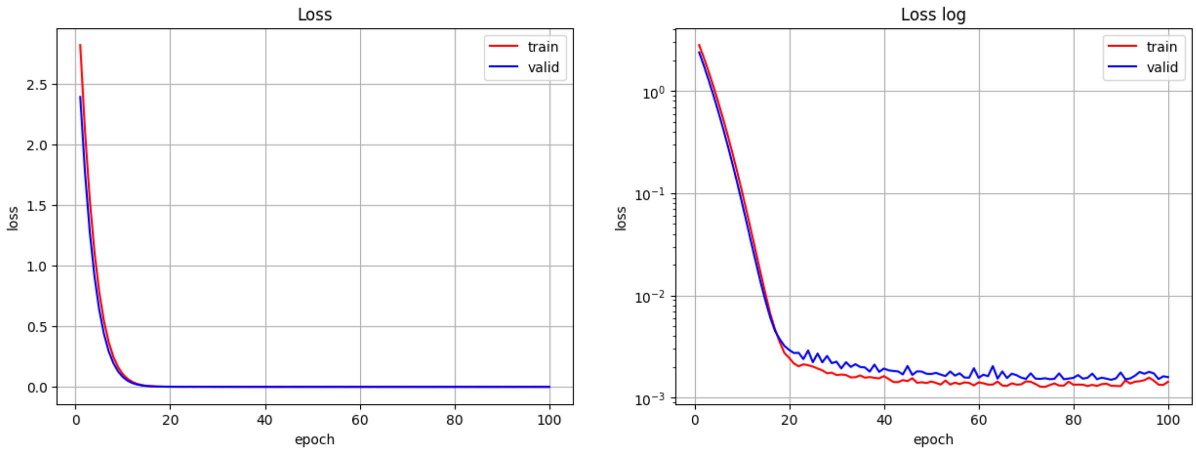

Figure 11 Training Loss Error for Base Deep Neural Network Model

When evaluating the mean squared error on the trained model, the results show consistent values with minimal overfitting. Results for three datasets are as follows:

- Train error: 0.001292
- Validation error: 0.001542
- Test error: 0.001789

4.2.5 Backtesting

Backtesting is conducted using logarithmic returns, a method that is more practical for portfolio rebalancing compared to simulating actual capital, which is more suitable for trading algorithms. The predictions from the trained models are validated to ensure they have only long values and are scaled to sum to 100% if necessary. These predictions are then forward filled for the entire subsequent period and multiplied by the returns of the ETFs.

After obtaining the total returns, transaction fees must be applied. While this step can be omitted in simple backtesting, including transaction fees enhances the accuracy of the backtesting process. Transaction fees can vary significantly across different brokers; therefore, a conservative rate of 0.3% of the transaction value has been selected. These fees are multiplied by the transaction delta during rebalancing and subtracted from the returns for that day.

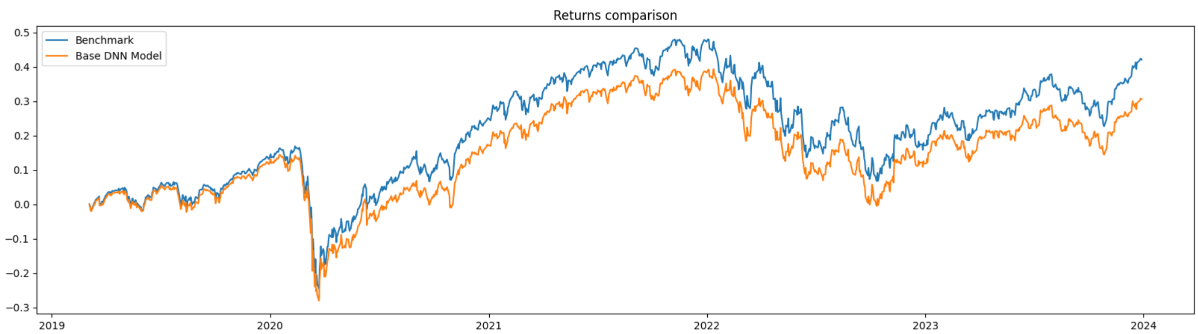

Figure 12 Returns Comparison for the Base DNN Model

With the total returns calculated, cumulative returns can then be determined using cumulative product function and used to display a comparison with a benchmark. The plot above shows cumulative returns from 2019 to 2024 for the MSCI ACWI benchmark (blue line) and a Base DNN

Model (orange line). Initially, both track similarly, but the benchmark recovers more quickly from the early 2020 drop associated with the COVID-19 pandemic and consistently outperforms the Base DNN Model throughout the period. The benchmark exhibits higher volatility but maintains superior returns, especially evident during the recovery phases and the general upward trend from late 2022 to 2024.

Using total returns multiple metrics can be calculated to compare results in more details. Following metrics have been selected for backtesting:

- Annual Returns: The percentage gain or loss of an investment over a year.
- Annual Volatility: The standard deviation of an investment's returns over a year, indicating its risk and price fluctuations.
- Sharpe Ratio: A measure of risk-adjusted return, calculated by dividing the excess return over the risk-free rate by the investment's standard deviation.
- Sortino Ratio: A variation of the Sharpe Ratio that only considers downside risk by dividing the excess return by the standard deviation of negative returns.
- Max Drawdown: The largest peak-to-trough decline in the value of an investment.
- Max Time Under Water: The longest duration an investment remains below its previous peak value.
- Calmar Ratio: A measure of risk-adjusted return, calculated by dividing the annualized return by the maximum drawdown.
- Information Ratio: A measure of portfolio returns above the returns of a benchmark, divided by the standard deviation of those excess returns.

Table 7 Backtesting Metrics for Base DNN Model

	Annual Returns	**Annual Volatility**	**Sharpe Ratio**	**Sortino Ratio**	**Max Drawdown**	**Max Time Under Water**	**Calmar Ratio**	**Information Ratio**
Benchmark	0.0768	0.2033	0.3775	0.4420	-0.3548	500	0.2163	0.0000
Base DNN Model	0.0576	0.2011	0.2864	0.3293	-0.3716	493	0.1550	-0.6104

Comparing metrics presented above, the MSCI ACWI benchmark outperformed the base DNN model with higher annual returns (7.68% vs. 5.76%), a better Sharpe Ratio (0.38 vs. 0.29), and a superior Sortino Ratio (0.44 vs. 0.33). Both had similar annual volatility (around 20%), but the Benchmark had a slightly smaller maximum drawdown (-35.48% vs. -37.16%), but a longer recovery time (500 vs. 493 days). Additionally, the Benchmark had a higher Calmar Ratio (0.22 vs. 0.16). Finally, a negative value for Information Ratio indicates inferior performance of the initial base model. Further models will try to improve these results to achieve proposed objective.

4.3 Evaluating Different Inputs

The initial phase in enhancing the base model's outcomes involves identifying the optimal inputs. A variety of input combinations can be examined from the gathered macroeconomic indicators to determine which ones yield the most effective and stable results.

The research has yielded 11 potential inputs that might improve base model. These inputs include:

- A single composite indicator utilizing a fixed weight. This is based on six key indicators, assigning a weight of 0.5 to the Manufacturing PMI, which is the most current at the point of rebalancing, and a weight of 0.1 to the remaining five indicators.
- A single input of Manufacturing PMI. This was deemed worth investigating independently as it is released with the least delay and often serves as a predictor of the economic cycle.
- Six separate key indicators, already utilized in the base model.
- 15 separate indicators, comprising six key indicators and nine additional indicators as outlined in sections 3.2.2 and 3.2.3.
- A single principal component computed using the PCA method, based on 6 key indicators.
- A single principal component computed using the PCA method, based on 15 indicators.
- Two principal components computed using the PCA method, based on 15 indicators.
- Three principal components computed using the PCA method, based on 15 indicators.
- A single principal component computed using the PCA method, based on 15 indicators plus monthly returns.
- A single principal component computed using the PCA method, based on 15 indicators plus 2 indicators for the investment clock, as described in section 3.2.4.
- A single principal component computed using the PCA method, based on 18 indicators. This includes 6 key indicators, 9 additional indicators, monthly returns, and investment clock coordinates.

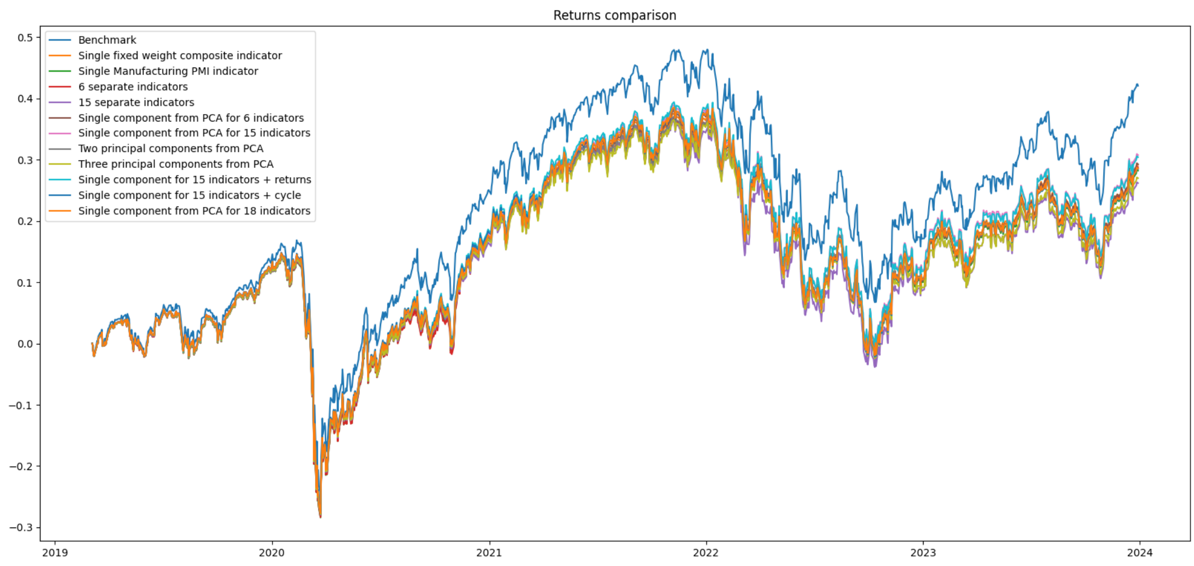

Figure 13 Returns Comparison for Various Inputs

The displayed graph for various inputs exhibits remarkably similar outcomes. A more detailed examination of the results can be achieved by comparing the annual returns and Sharpe ratios.

Table 8 Simple Backtesting Metrics for Various Inputs

	Annual Returns	**Annual Volatility**	**Annual Sharpe Ratio**
Benchmark	0.0768	0.2033	0.3775
Single fixed weight composite indicator	0.0552	0.2014	0.2741
Single Manufacturing PMI indicator	0.0536	0.2014	0.266
6 separate indicators	0.0546	0.2018	0.2704
15 separate indicators	0.0501	0.1999	0.2506
Single component from PCA for 6 indicators	0.0554	0.2014	0.2749
Single component from PCA for 15 indicators	0.0579	0.2005	0.2887
Two principal components from PCA	0.0541	0.2007	0.2695
Three principal components from PCA	0.0515	0.2000	0.2573
Single component for 15 indicators + returns	0.0573	0.2005	0.2859
Single component for 15 indicators + cycle	0.0541	0.2006	0.2697
Single component from PCA for 18 indicators	0.0542	0.2004	0.2704

Upon examining the backtesting metrics, it appears that the most effective results are obtained with inputs that incorporate a single component. This is primarily due to the limited number of training data points available for processing many inputs when they are passed separately. Furthermore, inputs utilizing the Principal Component Analysis (PCA) method yield superior results compared to fixed weights. However, the addition of a second and third component only seem to have detrimental effects.

In the final three tests, when examining monthly results and investment clock indicators, only returns seem to add value and in some tests yielded higher results than 15 indicators alone. However, the investment clock indicators result in worse returns when included in the model and will not be used. Further models in this thesis will use a single principal component based on 15 indicators plus monthly returns.

4.4 Evaluating Different Outputs

The next step in improving the model involves adjusting target values and exploring the application of additional restrictions on model outputs. In base model, normal constraint of country weights relative to reference MSCI ACWI index were used, but it can be extended to three different levels of constraints:

- Tight constraints allowing values between 70% and 150% of the reference weights.
- Normal constraints allowing values between 50% and 200% of the reference weights.
- Loose constraints allowing values between 30% and 300% of the reference weights.

For instance, considering the United Kingdom that had 11.09% weight in ACWI in year 2001, we obtain following ranges:

- Tight constraints would permit values between 7.76% and 16.64%.
- Normal constraints would permit values between 5.54% and 22.18%.
- Loose constraints would permit values between 3.33% and 33.27%.

Moreover, these constraints can also be applied to the model's output. While target values always fall within the permitted value ranges, the prediction model may yield weights that exceed these allowed values. By applying an additional optimization problem to the model's predicted weights, we can impose further restrictions to ensure that portfolio adhere to the same constraints. This approach may enhance the robustness of the model and is worth investigating.

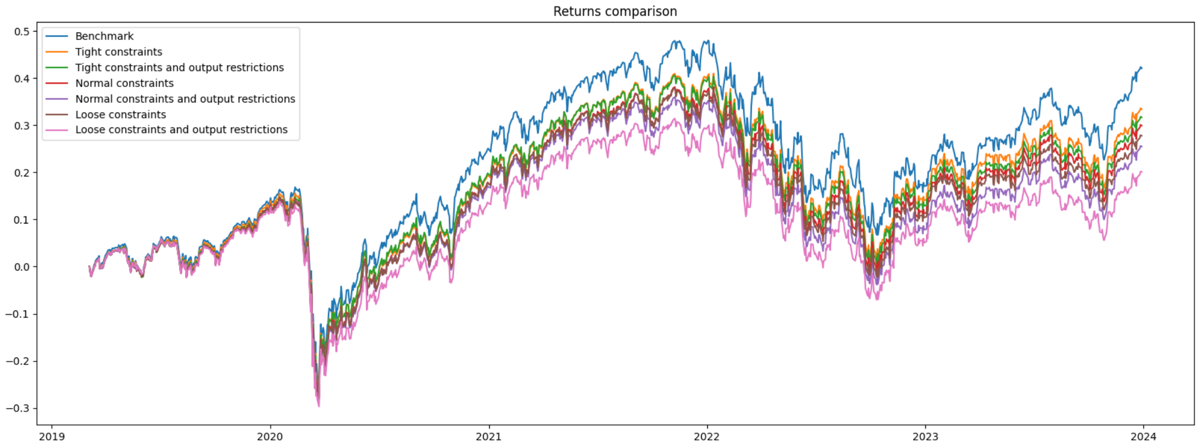

Figure 14 Returns Comparison for Various Outputs

Analysing the cumulative returns, different outputs seem to have much more profound impact on the portfolio results than when testing different input indicators.

Table 9 Simple Backtesting Metrics for Various Outputs

	Annual Returns	Annual Volatility	Annual Sharpe Ratio
Benchmark	0.0768	0.2033	0.3775
Tight constraint	0.0623	0.2007	0.3106
Tight constraint and output restrictions	0.0594	0.2008	0.296
Normal constraint	0.0565	0.2007	0.2815
Normal constraint and output restrictions	0.0489	0.2015	0.2428
Loose constraint	0.0528	0.1994	0.2651
Loose constraint and output restrictions	0.0395	0.2008	0.1965

Analysing results presented above, a direct correlation is observed, indicating that tighter constraints enhance both the returns and the Sharpe ratios. However, it's noteworthy that additional restrictions on the output adversely impact the model, resulting in suboptimal outcomes across all three levels of constraints. This suggests that while constraints are necessary to ensure robustness and adherence to certain parameters, overly restrictive conditions may hinder the model's ability to optimally allocate resources.

Moving forward, the investment algorithm will employ tight constraints without additional output restrictions, as they have demonstrated the most promising results both in terms of absolute and risk-adjusted returns.

4.5 Neural Network Models

After selecting optimal inputs and outputs, we can now examine various neural networks architectures. Apart from dense neural networks, it might be interesting to explore other models like convolutional networks and different types of recurrent networks which can handle sequences, potentially enabling them to better manage temporal data and identify more patterns than dense networks.

4.5.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a class of deep neural networks primarily used for processing structured grid data, such as images. They consist of layers that apply convolutional filters to the input, enabling the network to learn spatial hierarchies of features automatically and adaptively. Convolutional networks can also be effectively applied to temporal data, such as time series or sequential data, by leveraging their ability to capture local patterns and dependencies within the data. When used for temporal data, CNNs apply convolutional filters across time steps to identify temporal patterns and trends.

Architecture used in this thesis starts with a 1D convolutional layer with 32 filters and a linear activation function, producing an output shape of (None, 4, 32). This is followed by a Flatten layer that reshapes the output to (None, 128). A Dropout layer is then applied to reduce overfitting. Next are a Dense layer with 16 units and ReLU activation and the final Output layer with SoftMax activation, producing a 27outputs. That resembles a base model, described earlier, with extra convolutional and dropout layers.

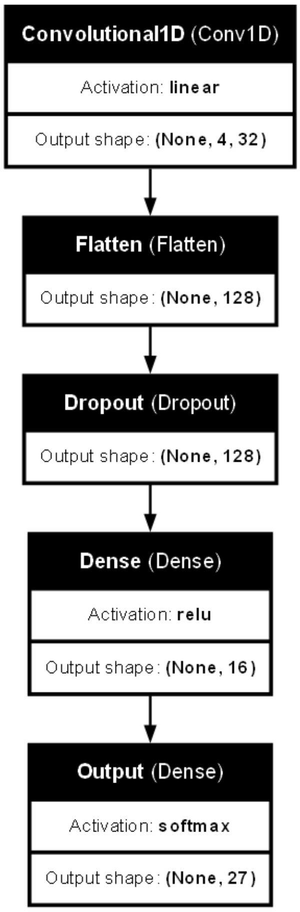

Figure 15 Base Convolutional Neural Network

Training loss error is very similar to base model. The only difference is that it is taking a bit longer to converge, thus requiring 150 epochs due to greater number of trainable parameters.

Backtesting results will be presented at the end together with recurrent networks.

4.5.2 Recurrent Networks

Recurrent Neural Networks (RNNs) are a class of neural networks designed for sequential data, where connections between nodes form directed cycles. This architecture allows RNNs to maintain an internal state and capture temporal dependencies, making them effective for tasks such as time series prediction, language modelling, and speech recognition.

In this research three different recurrent architectures have been examined. These are GRU, LSTM and Bidirectional RNN.

The simplest type of RNN is GRU (Gated Recurrent Unit) which uses gating units to manage the flow of information, helping to mitigate the vanishing gradient problem. It has two gates: a reset gate and an update gate, which control the update and forgetting process of the hidden state.

In the base recurrent model, a GRU layer has been used first before a standard Dense layer and the Output layer. Regularization is performed using dropout and recurrent dropout mechanisms within the GRU layer.

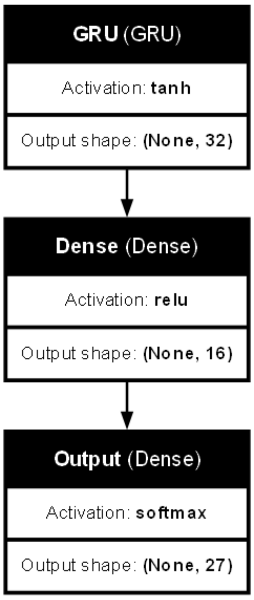

Figure 16 Base Recurrent Neural Network

Additionally, a GRU layer can be easily replaced with LSTM layer or a Bidirectional one. Long Short-Term Memory (LSTM) is more complex than GRU as it uses three gates: input, forget, and output gates, along with a cell state to regulate the flow of information, enabling it to remember important information over long sequences.

Finally, bidirectional RNNs process the input sequence in both forward and backward directions, using two separate hidden states. This allows the network to have information from both past and future contexts, improving performance on tasks where context from both directions is beneficial. Unfortunately, that architecture gives less stable results, varying significantly when training the same model multiple times. That can occur due to the bidirectional nature requires maintaining two sets of hidden states (forward and backward). Additionally, temporal information overlap might appear as the backward pass introduces future context that can sometimes interfere with the natural sequential order of data. This overlap can complicate the learning process, particularly for tasks where maintaining the original temporal sequence is crucial.

4.5.3 Backtesting for Base Neural Networks

When performing backtesting on the different models, obtained results are very close for all architectures with only Bidirectional model having slightly worse returns.

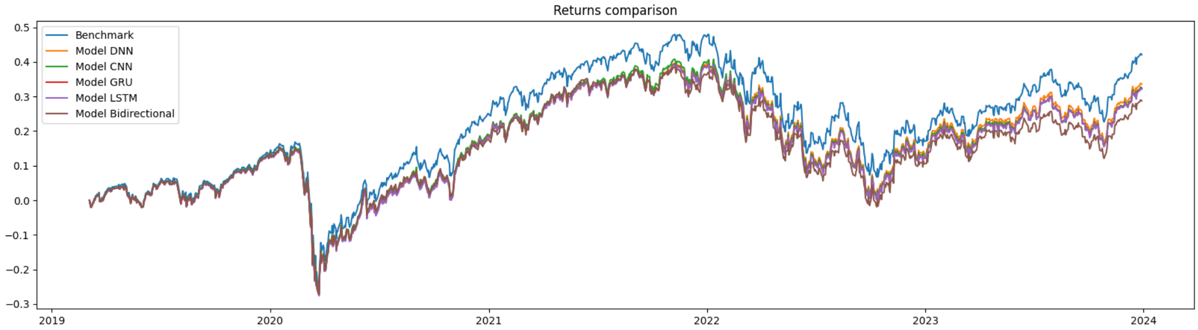

Figure 17 Results Comparison for Base Neural Network Models

Table 10 Backtesting Metrics for Base Neural Network Models

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.442	-0.3548	500	0.2163	0
Model DNN	0.0627	0.2013	0.3112	0.3601	-0.3677	499	0.1704	-0.5355
Model CNN	0.0608	0.2015	0.3019	0.3492	-0.3677	538	0.1654	-0.5912
Model GRU	0.0604	0.202	0.2989	0.3456	-0.3695	493	0.1633	-0.5967
Model LSTM	0.0601	0.202	0.2978	0.3442	-0.37	493	0.1625	-0.5972
Model Bidirectional	0.0545	0.2019	0.27	0.3133	-0.3684	538	0.1479	-0.6513

A comparative analysis of backtesting metrics across various architectural configurations reveals minimal performance differentiation. While dense and convolutional architectures demonstrated marginally superior returns and risk-adjusted returns across a range of ratios, the magnitude of these differences is statistically insignificant. This suggests that, for datasets of limited size, the specific architectural choice may exert a relatively weak influence on overall model performance.

Finally, based on the simplicity of the models and stability of the results, both dense neural networks and convolutional neural networks have been selected for further hyperparameters tuning.

4.6 Machine Learning Models

In addition to models based on neural networks, it could be beneficial to explore alternative machine learning models. Although a span of 25 years may appear to be a substantial duration for training an investment algorithm, it equates to merely 300 months when considering monthly rebalancing. After accounting for the initial year allocated for warm-up steps and the final four years reserved for backtesting, a total of 230 data points remain for model training. Given the complexity of deep neural networks, this quantity may prove insufficient, thereby necessitating the exploration of simpler methodologies. Among the various available methods, Random Forest and XG Boost have been chosen for further investigation.

4.6.1 Random Forest Regressor

The Random Forest Regressor is a supervised learning algorithm that utilizes an ensemble of decision tree regressors, each trained on different sub-samples of the dataset, to make predictions. This method enhances predictive accuracy and mitigates overfitting. The algorithm is particularly effective for regression problems due to its simplicity, high accuracy, and ability to control overfitting. By averaging the predictions of multiple decision trees, each with its own decision criteria, it generates a robust estimate of the expected output. This makes it a popular choice for real-world applications where continuous value prediction is required.

For the initial model, the parameters were set to their default values. It is noteworthy that, unlike neural networks, the input features must be flattened prior to their use, as the sklearn regressor only accommodates one-dimensional arrays. Lastly, it is crucial to rescale the outputs such that they sum to 1. This is because, unlike in neural networks where the SoftMax activation function is employed, the predicted weights in this model may exceed 100% if not appropriately rescaled.

4.6.2 XG Boosting

Like Random Forest, boosting is another methodology employing ensemble approach. It commences with the construction of a model on the training dataset, followed by the development of a subsequent model aimed at rectifying the inaccuracies of the initial model. This iterative process continues, with each newly introduced model concentrating on the instances that were deemed challenging and were misclassified by its predecessors. Upon the introduction of a new data point, it is processed through all the models, also known as weak learners. The class that secures the highest vote is then selected as the output for the test data.

XG Boost, which stands for eXtreme Gradient Boosting, is designed to be highly efficient, flexible, and portable, implementing machine learning algorithms under the Gradient Boosting framework. This algorithm has gained popularity for its performance in machine learning competitions and its efficiency in solving a wide range of machine learning problems.

The predictions generated by the XG Boost Regressor, akin to those from the Random Forest, do not inherently sum to one, necessitating a rescaling process. This rescaling appears to be more aggressive compared to that in the Random Forest model, which seems to have an adverse impact on the returns observed during backtesting.

4.6.3 Backtesting for Machine Learning Models

For the initial models with default parameters, the selected ensemble models do not appear to yield significant advancements for the investigated algorithm.

Figure 18 Results Comparison for Base Machine Learning Models

The returns generated by the Random Forest model are comparable to those from the neural network, while the outcomes from XG Boost are somewhat less favourable.

Table 11 Backtesting Metrics for Base Machine Learning Models

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.442	-0.3548	500	0.2163	0
Model Random Forest	0.0581	0.2021	0.2875	0.3338	-0.3669	538	0.1584	-0.698
Model XG Boosting	0.0498	0.2031	0.2454	0.2844	-0.3693	538	0.135	-0.7803

After analysing backtesting metrics, only Random Forest gives satisfactory results and will be examined further to see if hyperparameters tuning with cross validation can improve on these returns.

4.7 Learning to Rate

The final model under consideration is the Learning to Rate algorithm, also recognized as Learning to Rank. This algorithm is a specialized branch of machine learning dedicated to the development of models that address ranking problems. Unlike traditional problems that predict numerical values or class labels, ranking problems aim to predict a hierarchy among a set of items. Such problems are prevalent in numerous fields, including information retrieval, natural language processing, and recommendation systems.

In the context of optimizing a global portfolio, the Learning to Rate algorithm employs a distinct approach compared to the previously described models. While neural networks and ensemble methods directly output portfolio weights, the implemented Learning to Rate algorithm utilizes a multi-stage process.

Initially, the data processing method differs from neural networks. Targets are not passed as a two-dimensional array but are instead allowed only in one dimension with an additional query ID. In this scenario, each rebalancing yields 27 separate outputs, grouped under the same query ID. The same grouping applies to input features, which need to be concatenated vertically and passed with the same number of rows.

The initial ranking for training is determined by the Sharpe ratios calculated for each country for the subsequent period. A higher Sharpe value corresponds to a lower ranking value. The XGB Ranker from the XG Boost library is utilized in this process. The model is first trained on the training dataset and then used to predict scores for test data. The returned scores need to be reshaped and sorted by score value. Using these final scores, rankings can be assigned to all countries on the days of rebalancing.

In the final step, this ranking is applied as an additional constraint to the portfolio optimization problem. This calculates the Max Sharpe portfolio on the efficient frontier using quadratic programming with returns from the previous six months. These can be used as the final portfolio weights and verified during backtesting.

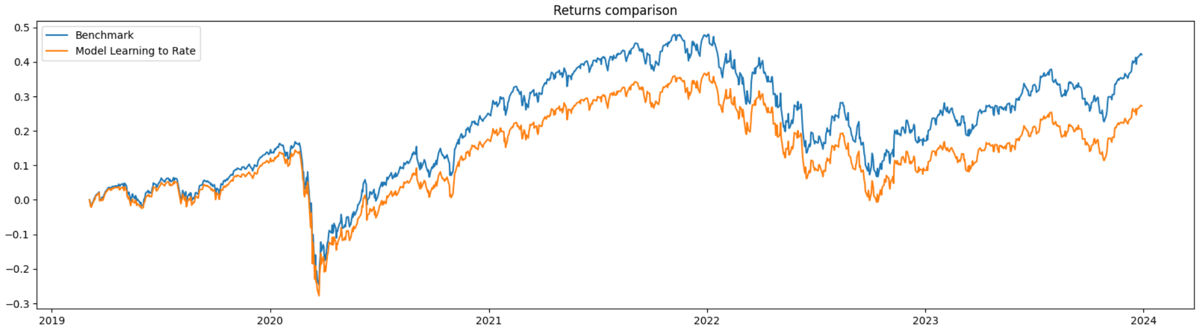

Figure 19 Returns Comparison for Learning to Rate Model

Table 12 Backtesting Metrics for Learning to Rate Model

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.442	-0.3548	500	0.2163	0
Model Learning to Rate	0.0519	0.2026	0.2562	0.2965	-0.369	499	0.1407	-1.0687

Analysing backtesting results, the returns achieved through the Learning to Rate algorithm, in the context of portfolio optimization, have been inferior compared to other models, such as neural networks or ensemble regressors. Despite this, the experiment has provided valuable insights.

The application of a ranking algorithm to a portfolio optimization problem is a novel approach that diverges from traditional methods. While the results may not have surpassed those of previous methods, the experiment has shed light on the potential of ranking algorithms in this domain.

4.8 Hyperparameters Tuning

Upon the selection of the most effective base models, these can be further enhanced through the tuning of hyperparameters. This process is divided into two segments: firstly, the selection of the optimal number of months, and secondly, the adjustment of model-specific parameters.

The selection of the optimal number of months is executed using Stratified K-Fold, a variant of K-Fold cross-validation techniques. This method optimizes data usage and minimizes variation in the results. The dataset is partitioned as previously, with 70% allocated for training and utilized in cross-validation, 10% for validation, which involves the calculation of the Sharpe ratio for each fold, and the remaining 20% is reserved for final backtesting with the tuned model.

Post the division of the training data into five folds, the model is trained on a single fold and the annual Sharpe ratio for each fold is calculated. The optimal number of months is selected based on the mean Sharpe ratio derived from all five folds.

Interestingly, different models have selected distinct optimal numbers of months. For Dense Neural Networks, the selection is 6 months, for Convolutional Neural Networks, it is 7 months, and for Random Forest, it is 9 months. These results exhibit a high level of consistency and do not fluctuate significantly even after multiple repetitions of cross-validation.

Having selected optimal number of months for each model, we can proceed to select model-specific hyperparameters. That is performed using Keras Tuner in case of DNN and CNN models and Randomized Search with CV from sklearn for Random Forest model.

For the neural networks, various numbers of units in hidden layers and levels of regularization have been tested. In comparison to the base model, the number of units in the Dense layer for Dense Neural Networks has been increased from 16 to 64, with L2 regularization of 0.1. For the Convolutional Model, most parameters from the base model were retained, with only small adjustment made to increase the units in the Dense layer from 16 to 32.

In the case of the Random Forest model, the randomized search has imposed additional restrictions such as a minimum sample leaf of 6, minimum samples split to 6, and a maximum depth of 10.

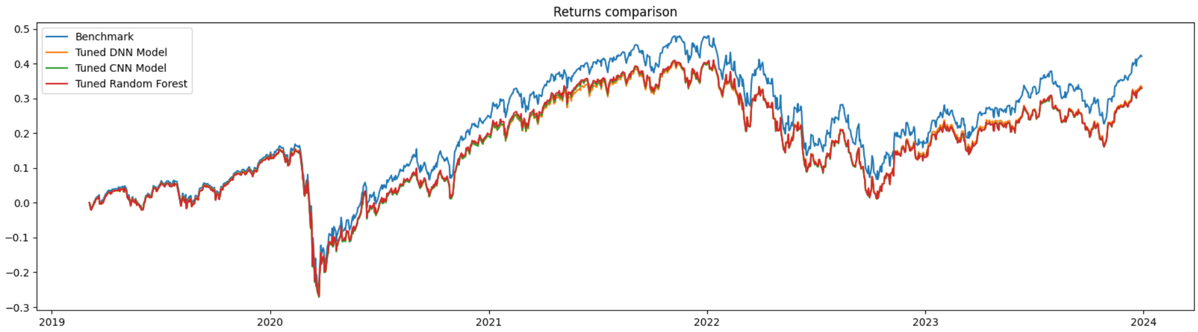

Figure 20 Returns Comparison for Tuned Models

Upon tuning, all three methods have achieved almost identical returns. That seems to indicate the maximum capacity of the selected methodology for provided features and targets.

Table 13 Backtesting Metrics for Tuned Model

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.442	-0.3548	500	0.2163	0
Tuned DNN Model	0.0623	0.2013	0.3094	0.358	-0.3667	499	0.1698	-0.5447
Tuned CNN Model	0.0614	0.2019	0.3041	0.3526	-0.3684	538	0.1667	-0.5633
Tuned Random Forest	0.0615	0.2021	0.3042	0.3534	-0.3662	493	0.1679	-0.585

Analysing the backtesting metrics, it is observed that the tuning of hyperparameters has resulted in only marginal improvements. The base models, which were manually adjusted to reduce the loss error, appear to exhibit performance metrics that are nearly identical.

In comparison to the benchmark, the annual returns are lower by one and a half percent (6.23% compared to 7.68%), with the volatility being almost identical. The Sharpe ratio is lower (0.30 compared to 0.37), and the Sortino ratio is also inferior, with an even greater difference than the Sharpe ratio (0.35 compared to 0.44). Tuned models also have a higher Max Drawdown and a lower Calmar Ratio, suggesting a higher potential loss and lower return relative to the risk taken. Lastly, the negative information ratio, ranging between -0.54 to -0.58, indicates that all models have underperformed in comparison to the benchmark.

4.9 CLASSICAL METHODS COMPARISON

To contextualize the results obtained in tuned models, several classical methods such as Minimum Variance, Risk Parity, and Hierarchical Risk Parity have been tested using the same data and backtested over identical period. These methods are elaborated upon in greater detail in Chapter 2 and here only brief summaries are presented below:

- Equally Weighted: This straightforward benchmark employs equal weights for all 27 countries.
- Minimum Variance: This investment strategy aims to minimize the overall risk (variance) of the portfolio.
- Risk Parity: This portfolio allocation strategy focuses on risk allocation rather than capital allocation. The objective is to distribute risk equally across various assets or asset classes within the portfolio.
- Hierarchical Risk Parity: Introduced by Marcos Lopez de Prado, this portfolio optimization method utilizes elements of graph theory and machine learning algorithms to group similar assets together.
- Hierarchical Risk Parity with Denoising v1: This variant of the HRP method includes an additional denoising step for the covariance matrix by fixing random eigenvalues.
- Hierarchical Risk Parity with Denoising v2: This variant of the HRP method incorporates an additional denoising step for the covariance matrix through targeted shrinkage.

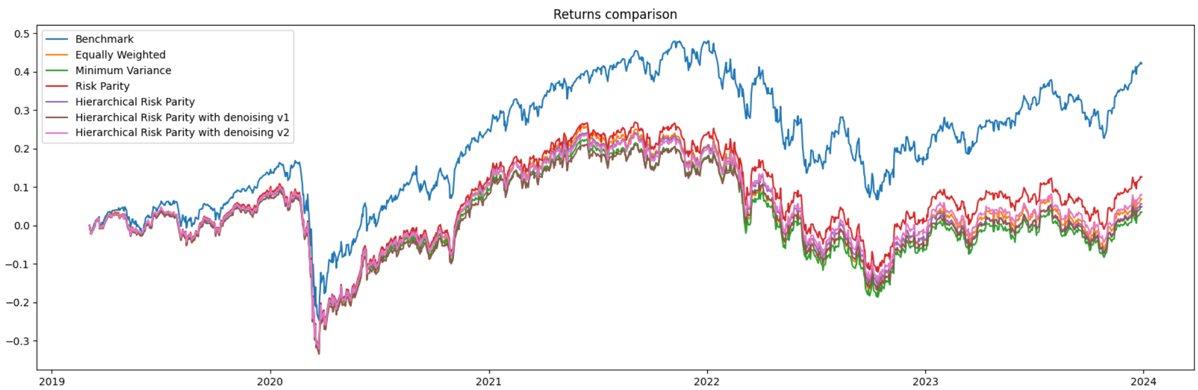

Figure 19 Returns Comparison for Classical Methods

When testing different portfolio optimization methods, obtained returns are much inferior compared to MSCI ACWI benchmarks. It appears that all methods prioritize minimum volatility and sacrifice total returns. Among all the methods, Risk Parity demonstrated marginally superior results, albeit still considerably inferior to the results of the previously tested AI models.

Table 14 Backtesting Metrics for Classical Methods

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.442	-0.3548	500	0.2163	0
Equally Weighted	0.0144	0.2103	0.0686	0.078	-0.3978	641	0.0363	-0.8602
Minimum Variance	0.0078	0.1995	0.039	0.0441	-0.3872	641	0.0201	-1.0246
Risk Parity	0.0256	0.1985	0.1289	0.1458	-0.3841	583	0.0666	-0.8046
Hierarchical Risk Parity	0.012	0.1965	0.0609	0.0687	-0.3811	641	0.0314	-0.9966

Hierarchical Risk Parity with denoising v1	0.0105	0.197	0.0533	0.0598	-0.3899	641	0.0269	-1.025
Hierarchical Risk Parity with denoising v2	0.0168	0.1962	0.0854	0.0959	-0.3846	641	0.0435	-0.9143

Upon examination of the annual returns, it is observed that the values are notably low, fluctuating between 0.78% and 2.56%. This is in stark contrast to the benchmark returns of 7.68%.

Further analysis of other risk-adjusted metrics reveals inferior performance, with values nearly three times lower than those of the MSCI ACWI. Achieved small reduction in the annual volatility doesn't compensate much lower returns. Finally, the information ratios, ranging from -0.8 to -1.02, signify a significant underperformance in comparison to the benchmark.

Finally, poor performances of Hierarchical Risk Parity methods are somehow surprising, given that much simpler Risk Parity method almost doubles both absolute and risk-adjusted returns in comparison to HRP.

4.10 Data Augmentation

In an additional attempt to enhance the results, a straightforward method of data augmentation was employed. The strategy involved transforming the training dataset from monthly periods to daily ones, thereby multiplying the data nearly 20 times from 288 data points to 6019. Regrettably, this simple transformation led to numerous identical inputs, as all features for the same month yielded identical indicators.

As for the targets, they were not identical to the inputs for a given month, but they exhibited only slight variations. In theory that could generalize result a bit better as it resulted in higher loss error, but still doesn't appear to improve backtesting metrics.

The results obtained with this data augmentation are almost identical to standard monthly data. The only significant difference observed was in the training process, which converged within 2-3 epochs instead of the usual 30-40 epochs. This can be attributed to the same data being passed over multiple batches, enabling the model to learn at a much faster pace.

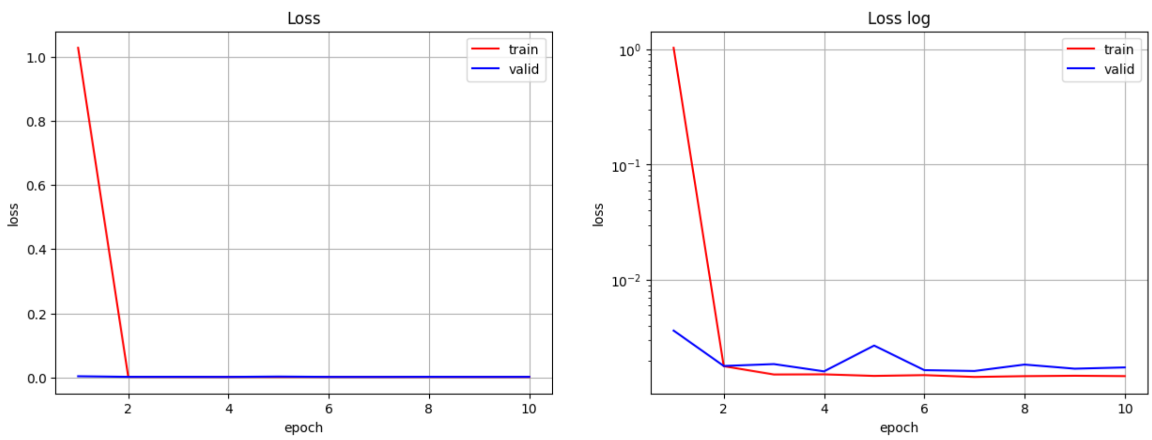

Figure 20 Training Loss Error with Simple Data Augmentation

4.10.1 Data Augmentation using Economic Calendar

To prevent data duplication as seen above, we can increase the number of data points by utilizing an economic calendar. Each indicator is assigned a specific report date and time, allowing us to determine the available data on any given day. This enables the use of the most recent data released without waiting for the start of the following month. Additionally, when preparing training data, distinct indicators can be utilized on different days, as multiple macroeconomic indicators are being published every day, preventing the duplication of monthly data.

However, this method has also its limitations. The economic calendar from Investing.com has limited data coverage prior to 2012, which effectively reduces the available data period by half. Furthermore, with a daily model, we are unable to incorporate indicators from other data sources due to the lack of exact release dates. Moreover, out of nine indicators from Investing.com used in monthly model, there is relatively low data coverage for PPI, Industrial Production, and Retail Sales – at around 50-60%. Having to fill data for almost half of the selected countries, these indicators do not offer significant benefits and have not been included in daily model. Consequently, we are left with only six key indicators, as detailed in Chapter 3.2.2.

With data selected, empty values in the input data are filled with the most correlated countries like in other models. Data is then normalized and converted to a single principal component using Principal Component Analysis. Target values are calculated daily, using portfolio optimization with tight constraints, which had provided optimal results for monthly models.

New training, validation and test data split is as follows:

- 2012 – warm-up steps for initial rebalancing.
- 02/01/2013 – 04/12/2017 – 1241 business days used for training.
- 05/12/2017 – 04/03/2019 – 311 business days used for validation.
- 05/03/2019 - 05/12/2023 – 58 monthly periods used for backtesting, which stays the same as for other model to be able to compare obtained results.

Dense Neural Network model uses practically the same architectures as for the Base DNN model. In this model, as opposed to tuned models, increasing number of units in hidden layers during hyperparameters tuning does not bring improvement. For this daily model 32 neurons are utilized, keeping model simple and thus reducing potential overfitting.

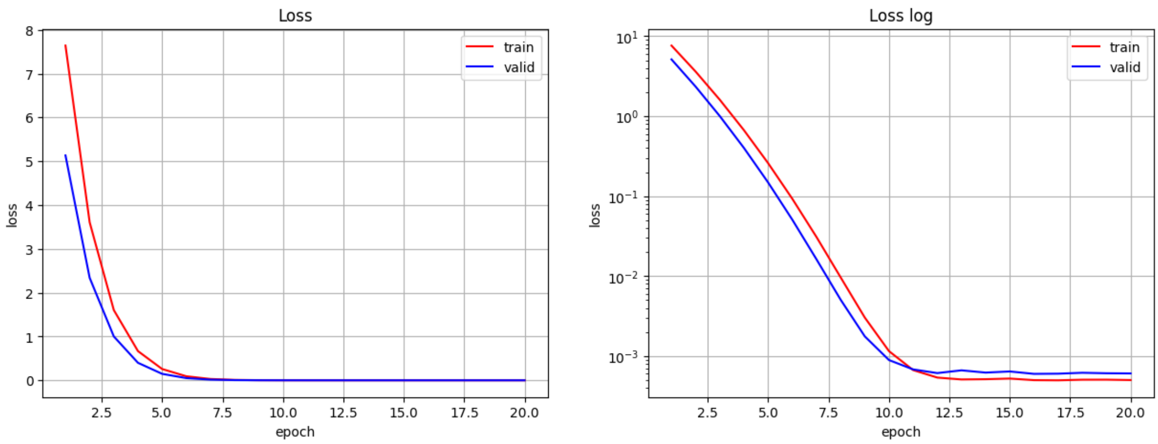

Figure 21 Training Loss Error with Data Augmentation using Economic Calendar

During training, the model converges very quickly, requiring only 12 epochs to reach minimal error value. That is more compared to simple data augmentation shown before, but still significantly less than in standard monthly model.

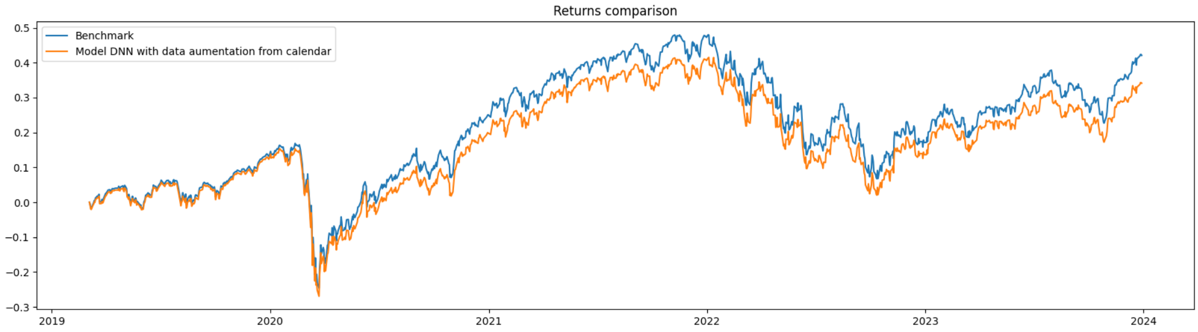

Figure 22 Returns Comparison for Daily DNN Model

Results obtained with Daily DNN Model, matched very closely Tuned DNN Model, but unfortunately still does not exceed the benchmark.

Table 15 Backtesting Metrics for DNN Model with Data Augmentation

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.442	-0.3548	500	0.2163	0
Model DNN with data augmentation from calendar	0.0635	0.2025	0.3134	0.3643	-0.3666	499	0.1732	-0.5888
Tuned DNN Model	0.0623	0.2013	0.3094	0.358	-0.3667	499	0.1698	-0.5447

Comparing backtesting metrics, both annual returns and various risk-adjusted returns gives slightly better results than Tuned DNN Model, described in Chapter 4.8. That proves that even with shorter data training period and less macroeconomic indicators, increased number of data point can produce very similar results.

Finally, data augmentation could be improved if a comprehensive economic calendar was accessible. While the examined open-source calendar from Investing.com have relatively good data coverage to key macroeconomic indicators, it lacks any extra indicators. In this scenario, a paid data provider, such as the Bloomberg terminal, could potentially enhance the quality of the data and improve the results.

4.11 Unrealistic Results Analysis

Before concluding this chapter, it is worth noting some unexpected results that emerged during the research. By applying loose constraints on the target values, the Random Forest model began to surpass the benchmark. Initially, this appeared to be a highly positive outcome. However, as further tunings were implemented, the results continued to improve, almost doubling the returns from the benchmark, shown on the figure below.

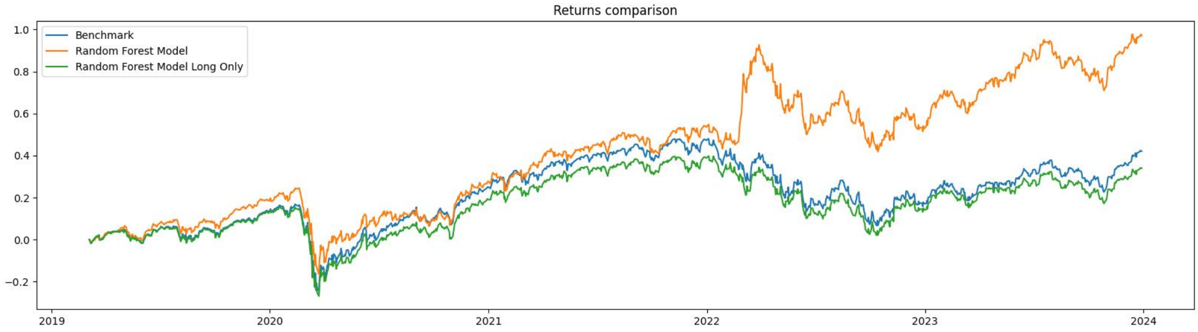

Figure 23 Results Comparison for Faulty Random Forest

Exceptionally good results should invariably prompt scrutiny, as they often indicate potential errors in the process. A thorough analysis revealed two errors associated with the allowance of negative values in both the loose constraints for target portfolios and in the predictions from the Random Forest model.

In the investment algorithm in this thesis, the portfolio weight should always total 1.0. However, with some negative outputs, a portfolio could take for example sample weights of 1.7 in a long position and 0.7 in a short position. The sum of these weights, 1.7 + (-0.7), equates to 1.0, returning the correct value of 100%, passing validation in backtesting. Regrettably, this would equate to utilising additional funds in certain periods, thereby gaining an unfair advantage.

One notable observation was the significant increase in February 2022. It appears that the algorithm took a substantial short position on Russia, which invaded Ukraine in the subsequent weeks. With the ETFs plummeting almost to zero after the invasion, this resulted in substantial gains that were sustained until the end of the backtesting period. While short positions were not permitted in this research, they could potentially be explored in future studies.

5 CLOUD INTEGRATION

In this research, due to large amounts of data for economic indicators, cloud computing solutions have been selected to help with the process of data collection and storage. They offer many advantages over traditional on-site solutions. Cloud computing enables rapid deployment, allowing developers to spin up or retire instances in seconds, thereby accelerating development processes. The inherent scalability and flexibility of cloud computing dynamically allocate resources based on workload, ensuring applications can handle sudden demand spikes effortlessly. In terms of data security, the risk of data loss due to hardware failures is mitigated by networked backups. Furthermore, the pay-as-you-go model of cloud computing ensures cost-effectiveness by charging only for the actual compute resources consumed during execution. Lastly, the accessibility of cloud-based applications and data from virtually any internet-connected device enhances the ease of data usage.

5.1 DATA COLLECTION AND STORAGE

In this project, I have selected solutions from Amazon Web Services (AWS), especially Lambda functions allowing periodical invocations of python code used to download the latest values for input features. All market data and macroeconomic indicators are stored in Microsoft SQL Server, managed within the Amazon Relational Database Service.

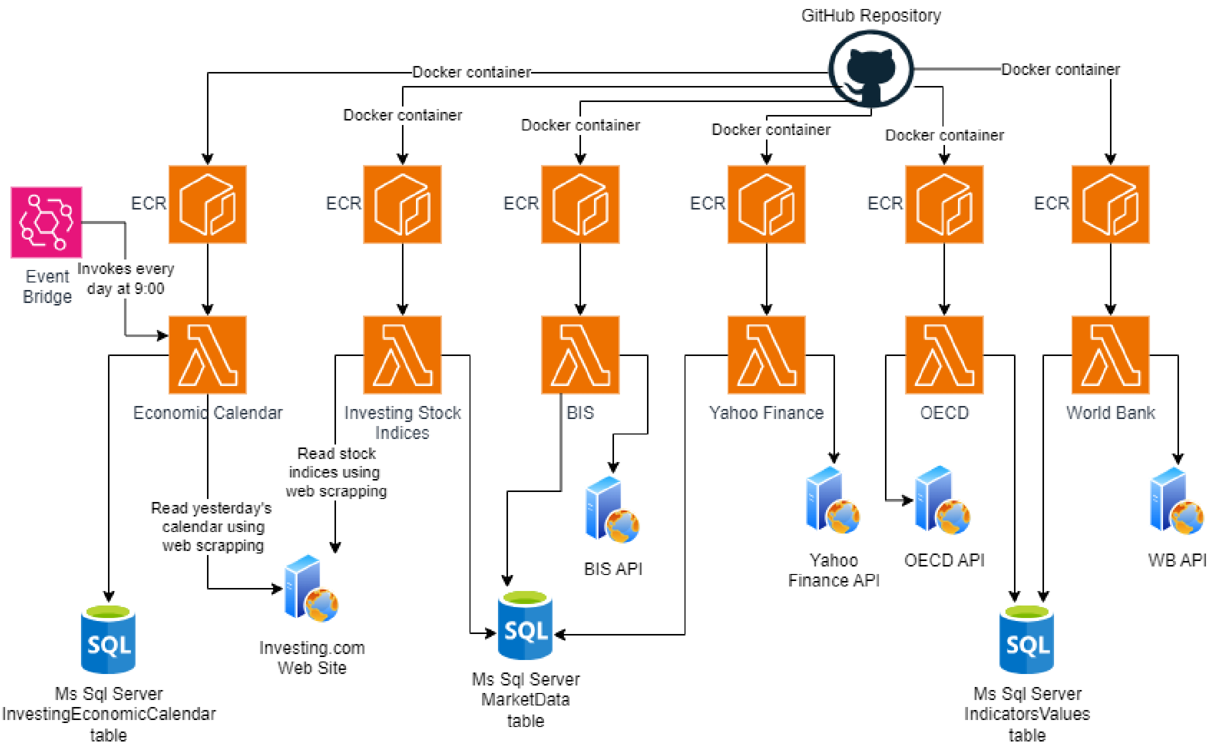

Figure 26 Cloud infrastructure diagram for data collection and data storage.

The figure above shows the general cloud architecture for data collection and data storage. It does not include all Event Briges, used to trigger remaining 5 Lambda functions as that would decrease the visibility of the diagram.

Keeping all the data up to date is handled by six Lambda functions that periodically retrieve the latest data from all data sources via API calls or techniques of web scrapping. These functions, which are both scalable and cost-effective, operating only couple seconds every morning, for data updated daily, and weekly for less frequently updated indicators. This might prove beneficial for extending back-testing over time and for production algorithm deployment.

The final significant component of the cloud infrastructure is the Elastic Container Registry, utilized for updating and storing Docker images used to run Lambda functions. The use of containers has been particularly advantageous for web scraping tasks of the economic calendar from the Investing.com website, as it facilitated the installation of additional chrome drivers and the selenium library, while being isolated from other functions and system updates.

Source code for all Lambda functions is versioned in GitHub Repository. Deployment process is implemented using Terraform, an infrastructure as code tool, what allows to provision and destroy an entire cloud infrastructure automatically by executing a single bash script. That can be used in times when a given data source is not needed or must be updated due to API changes.

Different Lambda functions read distinct data sources:

- Economic Calendar Lambda – uses web scrapping techniques with Selenium to read all the indicators from the previous day. While Investing.com page has some protections against automatic selection of data range from the calendar, it does allow to select yesterday values. For that reason, the initial web scrapping for data all the way to 1999, was performed in semi-automatic way, by selecting manually date range of several months, waiting for an entire period to load and then running a python function to read all presented indicators and saving them to database.
- Investing Stock Indices Lambda – uses web scrapping techniques with Selenium to read 8 stock indices from Investing.com web page that are not available in Yahoo Finance. They were used for synthetic ETFs data.
- BIS Lambda – reads central bank rates from Bank of International Settlements API. At the time of writing these rates were updated at the end of each week.
- Yahoo Finance Lambda – reads ETFs prices, stock indices values and currency rates from API using yfinance library. Can be extended to new instruments by simply adding new symbols to the database table which will be included automatically in the next update.
- OECD Lambda – reads all indicators from official API and saves values to the database if any new records are available.
- World Bank Lambda – reads worlds bank yearly indicators from API using third-party wbgapi library.

5.2 MODEL DEPLOYMENT

Daily DNN Model has been selected for cloud deployment using Amazon Web Services Lambda functions. While it does not utilize all downloaded indicators, it can calculate global portfolio weights using the latest indicators downloaded from economic calendar. These weights are calculated daily, triggered by CloudWatch Event. Additionally, an extra destination could be linked in the future, performing simulated trading with some external API from an online broker.

Like other Lambda functions used for data collection, this function uses a docker container and terraform scripts to easily create and destroy all required cloud resources on demand.

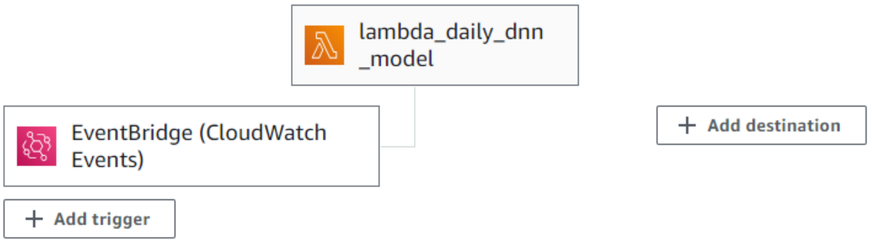

Figure 24 Diagram for Daily DNN Model Lambda

Lambda function performs multiple steps to calculate portfolio weights:

1. First, for all six key indicators it reads calendar entries from the last 12 months and calculates 6 most recent values for each indicator. However, a new month is used only after it surpasses a threshold of 10 countries that have already published values for that month.
2. Next, missing values are filled using correlations from ETFs returns, data is normalized, and it calculates a single principal component.
3. With input data prepared, it recreates neural network model and reloads weights from pre-trained model.
4. Finally, the restored model is used to predict global portfolio weights that are printed in the output and saved in logs.

In that model, an additional constrains are applied to restrict weight for Russia ETF that has been uninvertible since the invasion of Ukraine. That weight is reset to zero and the remaining countries are rescaled to sum 100%.

Sample portfolio weights calculated by the deployed model on 24/06/2024 are included in the table below.

Table 16 Portfolio Weights on 24/06/2024

Country	Portfolio Weight
United States	0.500290
Japan	0.072474
United Kingdom	0.063347
Canada	0.037189
France	0.027571
Switzerland	0.027813
Germany	0.025154
Australia	0.028330
Netherlands	0.011467
Sweden	0.011213
Hong Kong	0.012035
Spain	0.009721
Italy	0.010059
Singapore	0.009718
Denmark	0.010797
Finland	0.010170
Belgium	0.012804
Norway	0.008127
China	0.025176

Taiwan	0.013633
India	0.011064
Korea	0.019387
Brazil	0.013741
Russia	0.000000
South Africa	0.011311
Mexico	0.008134
Malaysia	0.009272

As mentioned before, Lambda functions are very cost effective and scalable. They require very few resources and with daily executions takes less than 25 minutes of billed durations during the entire month in case of model Lambda function. Detailed statistics for the last week of June 2024 are shown in the following table.

Table 17 Billed resources for Daily DNN Model Lambda

Timestamp	Billed Duration in MS	Memory Used in MB
2024-06-27T08:01:38.465Z	64866	788
2024-06-26T08:01:40.244Z	56380	289
2024-06-25T08:01:54.132Z	69760	459
2024-06-24T08:01:52.377Z	68663	461
2024-06-21T08:01:48.040Z	62976	456

6 CONCLUSIONS

The objective of this thesis was to implement an investment algorithm that would optimize global portfolio allocation using macroeconomic analysis. Processing multiple indicators, an artificial intelligence model would indicate in which countries it would be advantageous to allocate more capital based on promising trends in their economies. That could provide an alternative to passively managed funds tracking performances of broad global indexes like MSCI All Country World Index (ACWI).

6.1 PERFORMED WORK

This thesis has begun with a comprehensive overview of key concepts and methodologies relevant to global portfolio diversification. The discussion then delves into Modern Portfolio Theory, highlighting the efficient frontier and optimal portfolios, used as target values in developed models. Advanced portfolio optimization techniques have been also covered, including the Black-Litterman Model, which integrates investor views into the optimization process, Hierarchical Risk Parity for risk-based asset allocation, and techniques for covariance matrix denoising and Nested Clustered Optimization, which provide a robust framework for enhancing portfolio performance through advanced statistical methods.

Next, extensive research has been performed to examine all available open data providers and to identify relevant macroeconomic indicators critical to investment decisions. The concept of the Investment Clock has been explored to identify economic cycles and check its benefits in the investment algorithm. Afterwards, the MSCI ACWI benchmark has been investigated, focusing on the inclusions, exclusions, and changes in country weights for the last 25 years, as well as the criteria for country selection. The thesis has further explored investment instruments, detailing selected ETF data and synthetic ETF prices for missing data periods. Afterwards, data preparation processes have been meticulously performed, encompassing methods for filling missing values with mean values from the most correlated countries, calculating available indicator data for specific dates, formatting input data, and splitting the training data. Furthermore, target values have been computed as optimal Max Sharpe portfolios using quadratic optimization, thereby establishing a robust foundation for subsequent model training.

Subsequently, a variety of neural network architectures and machine learning models have been developed and validated using established methodology. Different neural network architectures have been explored to compare base dense neural networks with convolutional and recurrent models. Furthermore, various machine learning models such as Random Forest, XG Boosting and Learning to Rate have been implemented to evaluate simpler models that can sometimes prove to be superior for small datasets. Moreover, all models have also been backtested to eliminate the least performing models and select the most promising ones for further hyperparameters tuning. Additional data augmentation techniques have been also employed to explore way of improving model results with daily data.

Within cloud integration, data pipelines utilizing AWS Lambda Functions have been established to collect data from a variety of sources. All indicators and market data have been stored in a remote Ms SQL database. Finally, dense neural network model has been deployed using the most up-to-date economic data from economic calendar, performing daily predictions for portfolio weights.

6.2 COMPLETE RESULTS

During this thesis a variety of models have been developed. All results are listed below to provide a full comparison of the results in a single place. First, there is a separate comparison of different inputs and outputs for the base model.

Table 18 Backtesting Metrics for different inputs and outputs

	Annual Returns	Annual Volatility	Annual Sharpe Ratio
Benchmark	0.0768	0.2033	0.3775
Single fixed weight composite indicator	0.0552	0.2014	0.2741
Single Manufacturing PMI indicator	0.0536	0.2014	0.266
6 separate indicators	0.0546	0.2018	0.2704
15 separate indicators	0.0501	0.1999	0.2506
Single component from PCA for 6 indicators	0.0554	0.2014	0.2749
Single component from PCA for 15 indicators	0.0579	0.2005	0.2887
Two principal components from PCA	0.0541	0.2007	0.2695
Three principal components from PCA	0.0515	0.2000	0.2573
Single component for 15 indicators + returns	0.0573	0.2005	0.2859
Single component for 15 indicators + cycle	0.0541	0.2006	0.2697
Single component from PCA for 18 indicators	0.0542	0.2004	0.2704
Tight constraint	0.0623	0.2007	0.3106
Tight constraint and output restrictions	0.0594	0.2008	0.296
Normal constraint	0.0565	0.2007	0.2815
Normal constraint and output restrictions	0.0489	0.2015	0.2428
Loose constraint	0.0528	0.1994	0.2651
Loose constraint and output restrictions	0.0395	0.2008	0.1965

After analysing results for various inputs and output, a single principal component for 15 indicators together with monthly returns have been selected as a preferred input for further models. When it comes to outputs, tight constraints without additional output restrictions were selected as they presented significantly superior results over other outputs.

Next, multiple neural networks architectures and machine learning models have been developed. Backtesting metrics for all models are presented below.

Table 19 Backtesting Metrics for Various Models

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.4420	-0.3548	500	0.2163	0.0000
Base DNN Model	0.0576	0.2011	0.2864	0.3293	-0.3716	493	0.1550	-0.6104
Model DNN	0.0627	0.2013	0.3112	0.3601	-0.3677	499	0.1704	-0.5355
Model CNN	0.0608	0.2015	0.3019	0.3492	-0.3677	538	0.1654	-0.5912
Model GRU	0.0604	0.2020	0.2989	0.3456	-0.3695	493	0.1633	-0.5967
Model LSTM	0.0601	0.2020	0.2978	0.3442	-0.3700	493	0.1625	-0.5972
Model Bidirectional	0.0545	0.2019	0.2700	0.3133	-0.3684	538	0.1479	-0.6513
Model Random Forest	0.0581	0.2021	0.2875	0.3338	-0.3669	538	0.1584	-0.6980
Model XG Boosting	0.0498	0.2031	0.2454	0.2844	-0.3693	538	0.1350	-0.7803
Model Learning to Rate	0.0519	0.2026	0.2562	0.2965	-0.3690	499	0.1407	-1.0687
Tuned DNN Model	0.0623	0.2013	0.3094	0.3580	-0.3667	499	0.1698	-0.5447

Tuned CNN Model	0.0614	0.2019	0.3041	0.3526	-0.3684	538	0.1667	-0.5633
Tuned Random Forest	0.0615	0.2021	0.3042	0.3534	-0.3662	493	0.1679	-0.5850
Model DNN with data augmentation from calendar	0.0635	0.2025	0.3134	0.3643	-0.3666	499	0.1732	-0.5888
Final DNN Model	0.0621	0.2015	0.3081	0.3571	-0.3671	493	0.1691	-0.5525

Upon analysing the obtained results, it is evident that many models exhibit similar performance, with annual returns ranging from 6% to 6.3% and Sharpe ratios from 0.29 to 0.31. Only the baseline machine learning models and the bidirectional neural network model demonstrated slightly lower performance, achieving approximately 5% annual returns. In the end, dense neural network model has proven the most effective and provided the most stable results. It has been selected as a final model and deployed using Lambda function.

It is important to note that minor variations in results are expected even when using identical neural network models. When training multiple times, metrics can vary by approximately ±0.002 for different ratios. This variability is due to the initialization of model weights with random values, which influences the training process. Concern should arise only when results vary significantly upon repeated runs, as this may indicate overfitting. This phenomenon has been observed multiple times when testing overly complex models with multiple layers, where metrics such as the Sharpe ratio varied between 0.25 and 0.34. Such extreme results should be disregarded and not used to substantiate research findings, as they are unlikely to be reproducible in a production environment.

Finally, metrics for classical portfolio optimization method are included as well for easier comparison with other models.

Table 20 Backtesting Metrics for Classical Methods

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.442	-0.3548	500	0.2163	0
Equally Weighted	0.0144	0.2103	0.0686	0.078	-0.3978	641	0.0363	-0.8602
Minimum Variance	0.0078	0.1995	0.039	0.0441	-0.3872	641	0.0201	-1.0246
Risk Parity	0.0256	0.1985	0.1289	0.1458	-0.3841	583	0.0666	-0.8046
Hierarchical Risk Parity	0.012	0.1965	0.0609	0.0687	-0.3811	641	0.0314	-0.9966
Hierarchical Risk Parity with denoising v1	0.0105	0.197	0.0533	0.0598	-0.3899	641	0.0269	-1.025
Hierarchical Risk Parity with denoising v2	0.0168	0.1962	0.0854	0.0959	-0.3846	641	0.0435	-0.9143

These classical portfolio optimization methods have returned significantly inferior results both in comparison to the MSCI ACWI benchmark as well as to other AI models developed during this research. They seem to prioritize minimum volatility and sacrifice total returns as a result.

6.3 Results Comparison for Final Model

In the end, the top-performing model, using Dense Neural Network, has managed to achieve promising results, albeit without surpassing the benchmark. The following section compares the results from the Final DNN Model with the Benchmark and a traditional Risk Parity method.

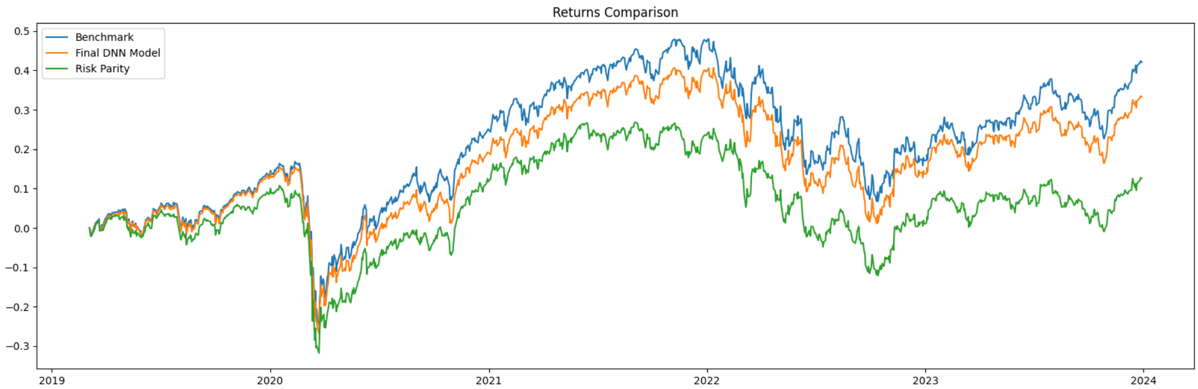

Figure 25 Returns Comparison for the Final DNN Model

Upon analysis of the results, it is observed that the Final Dense Neural Network (DNN) Model tracks the Benchmark with relative closeness but yielding slightly lower returns. However, it does exceed the returns from the traditional portfolio optimization method using Risk Parity by significant margin.

Table 21 Backtesting Metrics for the Final DNN Model

	Annual Returns	Annual Volatility	Sharpe Ratio	Sortino Ratio	Max Drawdown	Max Time Under Water	Calmar Ratio	Information Ratio
Benchmark	0.0768	0.2033	0.3775	0.442	-0.3548	500	0.2163	0
Final DNN Model	0.0621	0.2015	0.3081	0.3571	-0.3671	493	0.1691	-0.5525
Risk Parity	0.0256	0.1985	0.1289	0.1458	-0.3841	583	0.0666	-0.8046

Comparing backtesting metrics, we can observe that the Final DNN Model yields an annual return of 6.21%, which is slightly lower than the Benchmark's annual return of 7.68%. The model also exhibits an annual volatility of 0.2015, marginally lower than the Benchmark's 0.2033. When considering risk-adjusted indicators such as Sharpe Ratio, Sortino Ratio, Max Drawdown, Max Time Under Water, Calmar Ratio, and Information Ratio, the Final DNN Model generally underperforms the Benchmark, indicating a lower risk-adjusted return.

6.4 Findings and Future Work

There are several potential explanations for the underperformance of the developed algorithm. One limitation of the model is the lag in the indicators. Although various indicators are released throughout the month, they are only incorporated during the rebalancing of the following month, well after the market has already included that information in the prices of the assets. Additionally, there isn't always a direct correlation between economic indicators and market performance. For example, a robust economy could potentially lead to higher interest rates, which could adversely affect markets. Lastly, while macroeconomic data offers a snapshot of the current state of the economy, its predictive power for future market movements is limited.

Numerous other factors, such as investor sentiment, political events, and technological changes, also influence markets.

Further development of the algorithm could incorporate more complete economic calendar from paid data providers. That could improve data quality and allow to produce more robust training data with constantly changing indicators. Moreover, the algorithm could be redesigned to run on the daily basis, executing rebalancing when given turnover threshold is achieved rather than waiting for static monthly rebalancing at the beginning of each month.

In conclusion, although the proposed objectives have not been fully realized, the research conducted in this thesis has yielded invaluable insights. It has demonstrated the comprehensive process of developing an investment algorithm, with meticulous attention to details such as an in-depth analysis of the benchmark to circumvent survivorship bias, and the inclusion of transaction fees to achieve the most realistic results. This has been a truly intriguing and thought-provoking endeavour.

7 REFERENCES

Kenton, W. (2023). *Black-Litterman Model: Definition, Basics, and Example*. Retrieved from Investopedia: https://www.investopedia.com/terms/b/black-litterman_model.asp

Lindquist, W. R. (2022). Modern Portfolio Theory. In W. R. Lindquist, *Advanced REIT Portfolio Optimization*. vol 30. Springer, Cham.

Lynch, M. (2004). *The Investing Clock, Special Report #1: Making Money from Macro.* Global Asset Alllocation.

Mitchell, C. (2024). *MSCI All Country World Index (ACWI): Definition and Countries*. Retrieved from Investopedia: https://www.investopedia.com/terms/m/msci-acwi.asp

MSCI. (2019). *Selected geographical issues in the global listed equity market, Analysis for the Norwegian Ministry of Finance*.

MSCI. (2024). *ACWI IMI's Complete Geographic Breakdown*. Retrieved from https://www.msci.com/research-and-insights/visualizing-investment-data/acwi-imi-complete-geographic-breakdown

Prado, M. L. (2016). Building Diversified Portfolios that Outperform Out-of-Sample. *Journal of Portfolio Management*, 31.

Prado, M. L. (2018). *Advances in Financial Machine Learning*.

Prado, M. M. (2020). *Machine Learning for Asset Managers (Elements in Quantitative Finance)*. Cambridge University Press.

Roberto Violi, E. C. (2018). *Emerging market portfolio strategies, investment performance, transaction cost and liquidity risk*. ResearchGate.

Vanguard UK Professional. (2024). *The power of portfolio diversification*. Retrieved from Vanguard: https://www.vanguard.co.uk/professional/vanguard-365/investment-knowledge/principles-of-investing/the-power-of-diversification