Data-driven stock selection using Machine Learning for StockWatch

Project description: Bachelor Thesis IEM

Author: R.J. de Mink (Ruben)

Date: 07-07-2025

Company: StockWatch

UNIVERSITY OF TWENTE.

Cover photo: by (Dall-E, 2025)

Publication date: 07-07-2025

Student

R.J. de Mink (Ruben)

Industrial Engineering and Management

University of Twente

1st Supervisor:

dr. M.R. Machado (Marcos)

University of Twente

2nd Supervisor

dr. H. Chen (Hao)

University of Twente

Company supervisor

N. Koerts (Niels)

CEO

StockWatch

Preface

Dear reader,

This report is the result of my bachelor thesis assignment for the program Industrial Engineering and Management at the University of Twente. For this final project, I developed a data-driven model for StockWatch with the goal of systematically identifying outperforming stocks and improving the company's stock selection process.

For this thesis, I would like to thank my supervisor at the University of Twente, Marcos Machado, who guided me through this process by giving valuable feedback and guidance. I also thank my second supervisor, Hao Chen, for his fresh perspective on my project.

I also want to thank StockWatch, and in particular Arend Jan Kamp and Niels Koerts for giving me the freedom and opportunity to carry out my thesis at the company. As my company supervisor, I would also like to thank Niels Koerts for his guidance and feedback on the project, as well as the trust I got during the execution of this project.

Finally, I would like to thank my family and friends for their support and trust throughout this graduation project.

Ruben de Mink

July 7, 2025

Management Summary

This thesis presents the development of a data-driven model, which is designed for StockWatch to improve their stock selection process and identify outperforming stocks. StockWatch is a startup from Amsterdam that creates content about stocks. They write articles, make podcasts, and conduct thorough analyses about stocks. The company operates with a subscription model, with paid subscribers having access to all exclusive content, including stock portfolios on StockWatch's platform.

Objective

The motivation for this project comes from the current stock selection process, which lacks a clear systematic approach. StockWatch has access to an extensive LSEG database, which they do not fully utilize. This leads to their stock selection process being time-consuming, as there is no clear pre-selection of interesting stocks. Additionally, this leads to StockWatch's portfolios being unable to outperform the benchmark. The goal of this study is to develop a model, using the quantitative data from the LSEG database, to improve the stock selection process and identify outperforming stocks for StockWatch to support them in outperforming the benchmark on a risk-adjusted basis.

Methodology

We conduct this research using an adapted version of the Managerial Problem Solving Method (MPSM). This process begins with an introduction of the core problem faced by the company, which results in the following research question:

How can a data-driven model be developed to support StockWatch in systematically identifying outperforming stocks?

This is followed by a literature review, which is conducted using a Systematic Literature Review (SLR) methodology. This literature review consists of studying traditional asset pricing models, such as CAPM and the Fama-French models, as well as more recent Machine Learning models applied in the context of stock selection. Based on these insights, we design the solution planning and execute the modelling phase based on the CRISP-ML framework.

Results

During the model development phase, we develop three different models: one based on an Ordinary Least Squares (OLS) regression using the Fama-French 6-factor characteristics, and two Machine Learning models (a Random Forest and a Neural Network model) using a larger set of 19 stock-specific and 5 macroeconomic characteristics. All models aim to predict excess returns and support StockWatch in selecting outperforming stocks for its portfolios. Additionally, we develop an ensemble model, which combines the Random Forest and Neural Networks models with a 50/50 weight, which proves to be the optimal configuration among all experiments. To evaluate this model, we sort all stocks from the dataset into deciles based on their expected excess returns, we depict the realized excess returns of each decile in Figure 1, which shows the strong ranking ability of the model. This final ensemble model achieves a 9.6% R-squared value on an out-of-sample test and a Sharpe ratio of 0.94 for the top decile, producing 22.6% excess returns per year. In a cross-validation test, the model achieves a 15.3% R-squared value, and a Sharpe ratio of 1.13 for the top decile, delivering 68% higher returns per unit of volatility than the dataset average. The consistency across the decile rankings across both validation tests shows the predictive ability and robustness of the model.

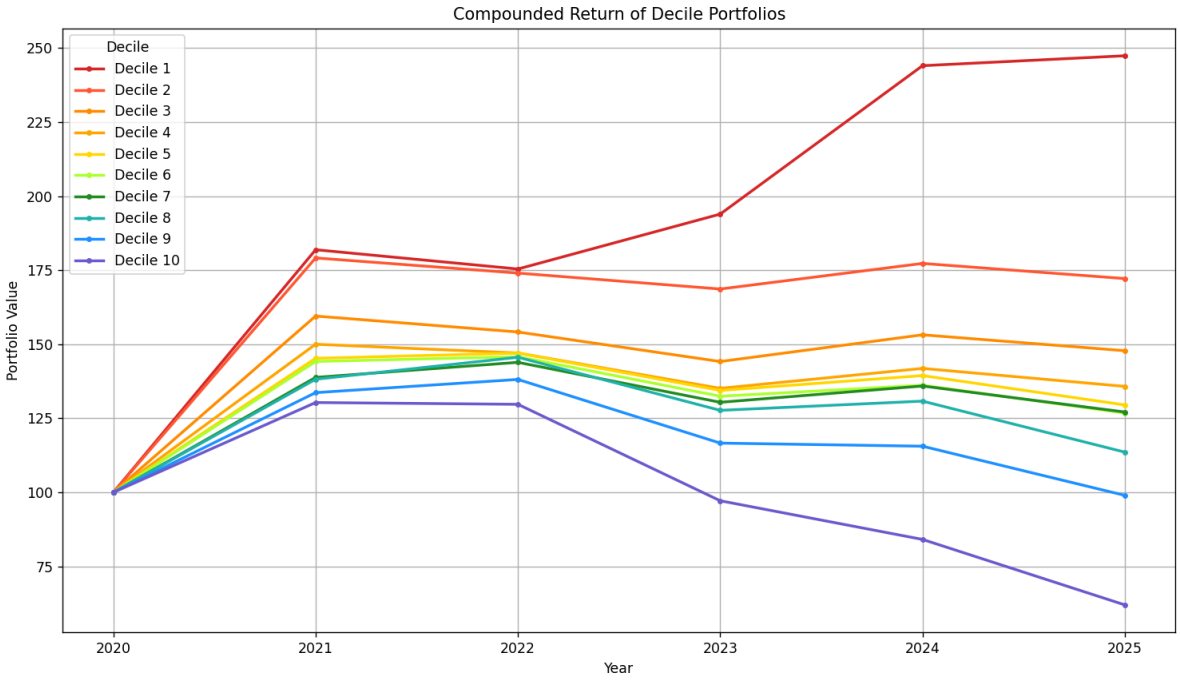

Figure 1: Decile performance graph out-of-sample

Conclusions and Recommendations

To make the model more practical for StockWatch, we develop a screener to further streamline their stock selection process. This tool generates a ranked list of all stocks from the S&P 500, AEX and EURO STOXX 50, ordered by predicted excess returns. The screener functionality includes filtering options to help users narrow down the set of stocks in an efficient way. We set up a Google Drive to practically implement this model for StockWatch, to use in their stock selection process, without the requirement of having technical software or coding knowledge.

This thesis concludes with an interesting recommendation for StockWatch: to launch a Big Data portfolio using this model as a foundation. This would provide additional value to subscribers and could be a valuable addition to the portfolios on the company's platform. This Big Data portfolio would require some additional tweaking of the model but is something interesting for StockWatch to pursue in the near future.

Contents

List of Tables

List of Figures

List of abbreviations

MPSM: Managerial Problem-Solving Method

SLR: Systematic Literature Review

KPI: Key Performance Indicator

OLS: Ordinary Least Squares

CAPM: Capital Asset Pricing Model

FF: Fama-French

SMB: Small minus Big

HML: High minus Low

ML: Machine Learning

NN: Neural Networks

RF: Random Forest

CAGR: Compound Annual Growth Rate

ROA: Return on Assets

ROIC: Return on Invested Capital

1 Introduction

This first chapter introduces the company StockWatch, together with the problem they are currently facing, which is the core of this thesis. Section 1.1 identifies the problem by means of a problem cluster, which ultimately identifies the company's core problem. In Section 1.2, we describe the problem-solving approach adopted in this thesis. Section 1.3 introduces the central knowledge question and the sub-questions that guide this study. We discuss the scope and limitations, and validity and reliability in Sections 1.4 and 1.5. This chapter concludes with the main deliverables this thesis aims to produce, in Section 1.6.

1.1 Problem Identification

StockWatch[1] is a startup from Amsterdam that creates content about stocks. They write articles, make quantitative analyses and podcasts about stocks. StockWatch generates revenue through a subscription model, with paid subscribers being able to access all exclusive content. Additionally, they allow paid advertisements on their website and during their podcasts. StockWatch has some stock portfolios on their website, which they also invest in. With the goal of maximizing stock returns, the company aims to develop a quantitative model to identify outperforming stocks. For that, the company aims to use a wide range of quantitative data, ranging from macroeconomic data to financial data about companies, such as their dividend yield and profit margins. Although StockWatch has access to an extensive LSEG[2] database containing all this data, it is not fully utilized yet.

1.1.1 Problem Description and Action Problem

StockWatch has some stock portfolios on their website, visible for paid subscribers. To select potential stocks for this portfolio, manual research is conducted about a lot of different stocks. This research is time-consuming, and additionally, it is not a trivial task to find stocks that are interesting for StockWatch's subscribers which also outperform the market.

Based on informal interviews with an expert from the company (Koerts, 2025), we identified the current situation at StockWatch. Currently, StockWatch selects stocks for further investigation without any systematic methodology. Stocks that look interesting based on their financial data or stocks that are popular amongst StockWatch subscribers are investigated (Koerts, 2025). This approach results in an extensive list of potential stocks. On all of these stocks, StockWatch performs thorough research, including an evaluation of the company's business model, financials, and the company's valuation. Based on this assessment, StockWatch decides whether to add the company to the portfolio. For all these stocks, even the ones not making it into the portfolio, StockWatch monitors earnings quarterly and closely follows the news and the stock price (Koerts, 2025). Despite this process being very time-consuming, many of these stocks turn out to lack investment potential and fail to outperform the benchmark.

The most important Key Performance Indicator (KPI) for StockWatch is that their portfolios perform well, this means performing better than the benchmark[3] on a risk-adjusted[4] basis. The primary reason relates to the need to attract subscribers, as potential customers do not want to subscribe to a stock information platform whose portfolios perform poorly. The two portfolios currently on StockWatch's platform are: an ETF portfolio, which is not the focus of this thesis, and a stock-only portfolio with around 20 different stocks from companies mainly from Europe and the United States (U.S.). Since StockWatch is a relatively new company, the portfolios on their website do not yet have a long enough track record (two years) to enable conclusions related to the performance of the portfolio in a long term.

[1] Link to StockWatch website: https://www.stockwatch.nl/
[2] Link to LSEG website: https://www.lseg.com/en
[3] Benchmark: MSCI World Index. Section 1.1.3 gives a more detailed explanation.
[4] We adjust by using β. Section 2.1 gives a more detailed explanation.

But since their inception, the portfolios have performed in line with the benchmark. The goal of StockWatch is to outperform the benchmark in the future, on a risk-adjusted basis. Thus, this goal can be achieved by tackling an action problem. We define an action problem as: "The discrepancy between norm and reality as perceived by the problem owner" (Heerkens, 2017). For StockWatch we identified the following action problem: *The StockWatch portfolios are not able to outperform the benchmark.*

1.1.2 Problem Cluster and Core Problem

Figure 2 identifies the root cause of the action problem presented in Section 1.1.1 through a problem cluster.

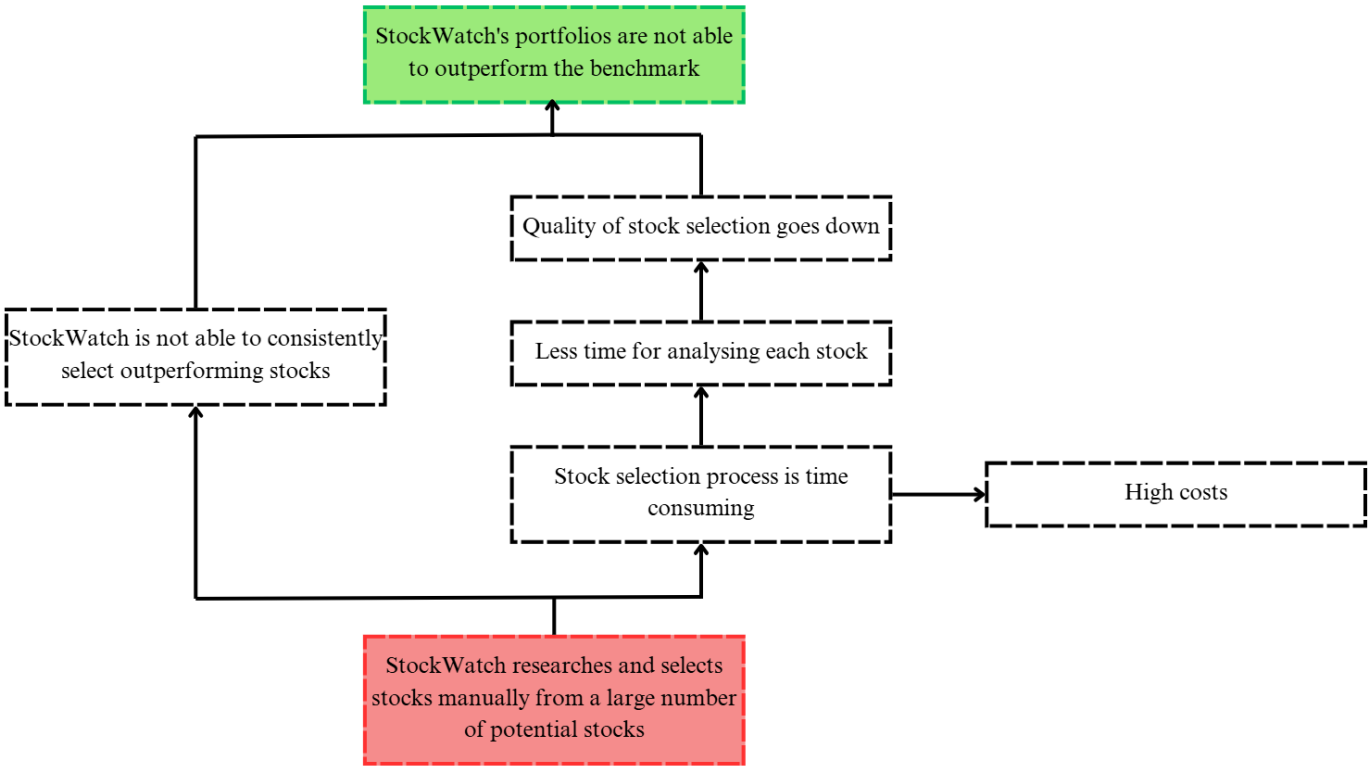

Figure 2: Problem cluster

The core problem is the root problem of the problem cluster (Heerkens, 2017). The problem cluster depicts that the core problem resulting in the portfolios being unable to outperform the benchmark is as follows: *StockWatch researches and selects stocks manually from a large number of potential stocks.*

The core problem has two negative causes, namely the inability to consistently select outperforming stocks and the stock selection process being time-consuming. The latter results in increased costs for StockWatch. Additionally, because such a large number of stocks must be analysed, the time available to analyse one individual stock decreases, which lowers the quality of the analysis. This, in turn, makes it more difficult to outperform the market. Therefore, addressing the core problem of having to select stocks manually from a large set of stocks would benefit StockWatch in two ways.

1.1.3 Norm and Reality

Since StockWatch is a relatively new company, the portfolios that are on their website do not yet have a long enough track record to determine whether they outperform the benchmark in the long run. However, since their inception, the portfolios have performed in line with the benchmark, and have not been able to outperform it. Based on the current context at StockWatch, norm and reality are defined as:

Reality: *StockWatch's portfolios perform in line with the benchmark*

Norm: *StockWatch's portfolios should outperform the benchmark on a risk-adjusted basis.*

The benchmark we compare to in this study is the MSCI World index[5], since StockWatch invests in the U.S. as well as in Europe. For risk, this study utilizes the beta (β) of the stock, similar to CAPM (Sharpe, 1964). We provide more information about this in Section 2.1.

1.2 Problem-Solving Approach

The problem-solving approach adopted in this research is a deviation from the Managerial Problem-Solving Method (MSPM) model. The MPSM is a framework developed by Heerkens and used to solve engineering problems in a series of phases (Heerkens, 2017).

- Phase 1: Problem Identification
- Phase 2: Problem Approach
- Phase 3: Literature Review
- Phase 4: Solution Planning
- Phase 5: Model Development
- Phase 6: Model Testing and Evaluation
- Phase 7: Model Implementation

We use these phases to address the problem described in Section 1.1.2. First, we identify the problem, which is done in Section 1.1. Then we formulate the problem approach. After these two phases, this research slightly alters the MSPM to better suit the problem-solving approach. Since the problem itself is already clear, a more suitable approach is to switch to the research cycle in phase three and conduct literature review, to eventually plan out the solution in phase four. Chapter 2 contains the planning of the solution, and Chapter 3 includes the literature review. In Chapter 4, we develop the model as part of phase five of the problem-solving approach. We test and evaluate the performance of this model in Chapter 5. Finally, Chapter 6 describes the implementation of the model for StockWatch. This structured approach results in a data-driven model with screener that enables StockWatch to select outperforming stocks.

1.3 Knowledge Problem and Research Questions

Following the problem-solving approach from the previous section, we formulate the following knowledge problem to address the core problem identified in the problem cluster in Section 1.1.2:

How can a data-driven model be developed to support StockWatch in systematically identifying outperforming stocks?

This research aims to develop a model, using quantitative data, which provides StockWatch with a smaller set of stocks to investigate, rather than manually analysing a large number of stocks. While developing this model, we aim to find a methodology that identifies stocks that potentially outperform the market. This enables StockWatch to find outperforming stocks and additionally save hours of human work that need to investigate a lower number of stocks, since the data-driven model functions as a supporting tool that reduces the number of stocks to be researched. For this, we define five sub-questions and discuss how we tackle them below:

I. How is the stock selection process currently done by StockWatch?

While solving this question, we aim to gain insight into the current way of selecting stocks, which is done through observation and an informal interview with the company supervisor. We provide an answer to this question in Section 1.1.1. The description of the current process helps with identifying which

[5] MSCI World Index factsheet: https://www.msci.com/documents/10199/178e6643-6ae6-47b9-82be-e1fc565ededb

improvements could be made in the stock selection process and highlights where the quantitative model will add value.

II. Which asset pricing models exist that predict an outperformance for certain stocks?

This second question aims to explore which asset pricing models have already been developed to explain stock outperformance. We tackle this question by performing a literature review conducted in a systematic way (SLR), we explain this procedure in Appendix G. Classical models like CAPM and Fama-French, which are commonly used in the literature are analysed. Additionally, more recently developed Machine Learning (ML) algorithms are discussed as well. Chapter 3 presents this theoretical foundation and belongs to the third and fourth phase of the problem-solving approach.

III. Which stock-specific and macroeconomic characteristics are useful for predicting stock outperformance?

The third question focuses on identifying which stock-specific and macroeconomic characteristics are useful for predicting stock outperformance. We explore this by conducting literature review, supplemented with data analysis through the usage of the LSEG database. We answer this question, which belongs to the third and fourth phase of the problem-solving approach, in Section 4.1.1.

IV. How can existing asset pricing models be enhanced by ML to develop a practical stock selection model for StockWatch?

The fourth question investigates how existing asset pricing models can be enhanced by ML techniques, which involves some literature research building on sub-question two. The aim is to build the model based on the characteristics selected from sub-question three and then enhance the model using ML algorithms. We answer this question, which belongs to the fourth and fifth phase of the problem-solving approach, in Section 4.3.

V. How can the performance of the developed model be tested and evaluated against the benchmark?

The last question focuses on how the performance of the developed model can be evaluated. This involves proving that the stocks that are selected as outperformers, actually perform better on a risk-adjusted basis than the benchmark index. We conduct a literature review to find the definitions and methods to compare the developed model against other existing models and support the evaluation and conclusions of this thesis. We answer this question, which belongs to the sixth phase of the problem-solving approach, in Chapter 5.

1.4 Scope and Limitations

An important limitation of this study is that the developed model is trained exclusively on the S&P 500 index from the U.S., since this makes it much easier to have a complete dataset. However, this choice might limit the model's applicability to other equity markets. We give an explanation for this choice in Section 4.1. This study focuses on stock-specific characteristics as well as macroeconomic characteristics. The macroeconomic characteristics used are likewise limited to data from the U.S. These variables are all quantitative, so as a result qualitative factors like sentiment, breaking news or statements by political actors are beyond the scope of this model, even though they might influence market movements (Tetlock, 2007). This represents another limitation of the model, and the inclusion of qualitative factors in this model would be an interesting addition for future work.

Another limitation is that the final model is not a perfect tool to solely decide which stocks to select. The primary goal of the model is to identify potentially interesting stocks for StockWatch, providing a basis for further investigation and portfolio selection. It is intended as a supporting tool, rather than one that directly indicates which stocks to buy. Reasons for this are that the stocks with the highest expected

excess returns are shown, but this does not mean that it is optimal to only buy these stocks, as this imposes a significant risk of portfolio overlap caused by holding many similar stocks. This would mean that company-specific risk is not sufficiently reduced and therefore the portfolio cannot be optimal (Sharpe, 1964). Moreover, this model is just a predictor of excess returns and relying solely on the model for portfolio selection is not advised.

1.5 Validity and Reliability

This study defines validity as: "Extent to which data collection methods accurately measure what they are intended to measure" (Saunders, 2019). To ensure validity, we choose stock-specific and macroeconomic characteristics carefully using academic literature. Moreover, we critically evaluate our model using industry standard evaluation metrics such as the R-squared value and the Sharpe ratio, additionally we construct deciles to test whether the selected stocks actually outperform. We perform these tests on an out-of-sample as well as a cross-validation basis.

We define reliability as: "Extent to which data collection technique or techniques will yield consistent findings, similar observations would be made or conclusions reached by other researchers or there is transparency in how sense was made from the raw data" (Saunders, 2019). To ensure reliability, the data collection process and the method of developing the model is thoroughly documented. Such that other researchers can replicate the model development and obtain the same results as this research does. This includes clearly describing the data and characteristics used, the stocks included and the analysed time period. Additionally, we include an explanation of how the ML model was trained and tested. We evaluate the model's performance using academically accepted metrics to make comparisons between models possible, and ensure findings are consistent.

1.6 Deliverables

Building on the problem analysis from the previous sections, this chapter explains the main deliverables of this study. The following four deliverables contribute directly to solving the core problem faced by StockWatch, and together form a complete answer to the main research question:

- **Review of existing models:** Analysis of the most important existing asset pricing models, including both traditional asset pricing models and ML algorithms applied in the context of asset pricing.

- **ML model:** An ML model that ranks all stocks based on their expected excess returns, making use of the stock-specific and macroeconomic characteristics selected from the LSEG database.

- **Model validation:** Validation of this model by comparing the risk-adjusted performance of the selected stocks to other stocks by means of out-of-sample and cross-validation. This includes critical evaluation done by experts from StockWatch, and a discussion of limitations of the model.

- **Stock selection tool:** A practical tool for StockWatch developed in Python, which implements this ML model to rank a list of inputted stocks based on their predicted performance. This tool, which also includes screening functionalities, can be used by StockWatch to streamline their stock selection process.

2 Theoretical Framework

In this chapter, we outline the theoretical framework on which this thesis is built. Section 2.1 defines the key concepts and variables according to academic literature. Section 2.2 outlines the theoretical perspective of this thesis, including the description of the CRISP-ML methodology used.

2.1 Key Concepts and Variables

An *asset pricing model* is a framework in finance that is used to determine the fair value of financial assets such as stocks. It explains the relationship between risk and expected return (Cochrane, 2001). In Chapter 3, we further explain the wide range of existing asset pricing models. This thesis limits the term asset pricing models to those concerning stocks.

The *stock selection process* refers to the entire process of identifying which stocks to investigate, to ultimately deciding whether to buy a stock. This process includes screening for potential attractive stocks, conducting an analysis of the company's fundamentals, management, financials, and valuation, and lastly assessing whether this company could be one to include in the portfolio. To solve research question one, we analyse the full process mainly by means of an informal interview with the company supervisor.

For *stock outperformance*, we use the following definition: "Return on an investment that exceeds the return of a benchmark with similar risk" (Ying, T., Q. u., & Y., 2019). This thesis uses the terms *excess returns* and *outperformance* interchangeably, both referring to returns that are always *risk-adjusted*. Adjusting for risk can be done in various ways. This study uses β as risk measure for a stock, with β being the sensitivity of a stock to market movements (Brealey, 2022). Another risk measure is the *Sharpe ratio*, which uses the standard deviation of returns and measures the return per unit of standard deviation, subtracting the risk-free rate (Sharpe, 1964). We use the Sharpe ratio as one of the evaluation metrics to evaluate the portfolio deciles from our model in Chapter 5.

Stock-specific characteristics are attributes of a stock itself, like its dividend yield, its momentum, its earnings per share, and other variables. *Macroeconomic characteristics* are general variables like the current interest rate or unemployment (Wang, 2024).

A well-known theory in the asset pricing field is the *Efficient Market Hypothesis (EMH)* by (Fama E. , 1970), which states that stock prices already reflect all available information, making consistent outperformance nearly impossible (Fama E. , 1970). In this thesis we use EMH as a comparison benchmark, and the goal of our model is to disprove EMH, and to show that there are variables that actually influence stock prices.

For *ML* we use the following definition: "The use of computer systems which are able to learn and adapt without following instructions, by using statistical models and algorithms to analyse and draw conclusions from patterns in data" (Oxford English Dictionary). This thesis uses ML to further enhance the model's performance, we present more information about the specific ML algorithms used in this study in Section 3.6.

2.2 Theoretical Perspective

The theoretical perspective in this thesis links the theory of asset pricing using factor models[6], with ML algorithms, to further strengthen the prediction of excess returns of stocks. The field of asset pricing links to accounting and finance, and the traditional way of valuing assets as described in *Principles of Corporate Finance* (Brealey, 2022). However, those asset pricing models can be further enhanced by ML algorithms as demonstrated by (Wang, 2024), which uses neural networks (NN) to predict stock

[6] Factor models are models that explain the return of a stock based on its exposure to certain factors/characteristics. We explain such models in Chapter 3.

outperformance. In addition to NN, this thesis uses random forests (RF) since their structure of combining multiple decision trees is well-suited for analysing the different characteristics which have been identified during the literature research. Finally, we explore combinations of the RF and NN algorithms to develop the optimal model for StockWatch.

For the ML aspect of this thesis, which belongs to the fourth research question, the CRISP-ML[7] methodology depicted in Figure 3 is an appropriate framework. This structured process starts with a thorough business and data understanding and identifying the scope of the project as done in Section 1.4 (Visengeriyeva, 2025). The next phase involves data engineering, including data selection, cleaning, and preparation (Visengeriyeva, 2025). This is followed by the model development phase, where the model is selected, specialised, and trained (Visengeriyeva, 2025). Subsequently, we evaluate the model before we deploy it for StockWatch. Finally, the model operations phase consists of implementing and maintaining the developed model in a practical way. For this, the screener model, which we explain in Chapter 6, is developed for StockWatch.

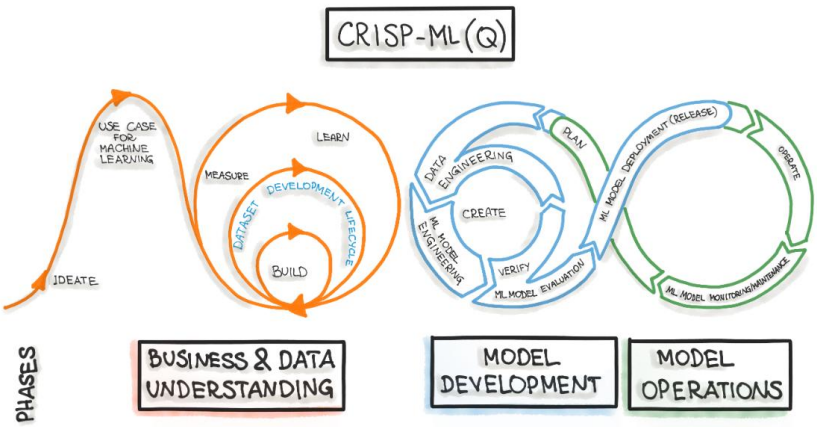

Figure 3: CRISP-ML framework (Visengeriyeva, 2025)

This thesis uses a deductive as well as an inductive approach. First, we apply a deductive approach to select the characteristics to include in the model. This starts by means of a literature review to form a hypothesis about the characteristics to use. Then, we evaluate this hypothesis to find out if the model has consistent predictive ability. Next to deduction, we also use induction during the development of the model in this study, since the vast amount of data from the LSEG dataset can reveal patterns itself, especially when using ML algorithms.

[7] CRISP-ML website: https://ml-ops.org/content/crisp-ml

3 Literature Review

This chapter reviews the literature on existing asset pricing models and provides an overview of these models and how they work. In the first section we introduce and explain some well-known existing models such as EMH, APT, CAPM, and Fama-French. In addition, we explore some more recent ML methods like NN and RF. We perform this review by conducting an SLR, we briefly explain this SLR protocol in Appendix G. The purpose of this literature review is to get insights into possible ways of developing asset pricing models, to subsequently use this in the development of the model for StockWatch.

3.1 Existing Theories and Asset Pricing Models

Stock outperformance has been a central theme in asset pricing research. The ability to explain and predict which stocks provide excess returns and why, has led the evolution of the development of many different asset pricing models. In a study from (Ying, T., Q. u., & Y., 2019), excess returns are defined as return on an investment that exceeds the return of a benchmark with similar risk.

The foundation of asset pricing began with theories like the dividend discount model (DDM), which states that a stock price should equal the expected value of future dividends, discounted against a discount rate that reflects the risk of a certain stock (Brealey, 2022), as we show in Equation 1.

$$P_0 = \sum_{t=1}^{\infty} \frac{Div_t}{(1+r)^t}$$

Equation 1: DDM

Where P_0 equals the price of a stock at time t = 0, and r is the discount rate.

The first theory we draw upon is the Efficient Market Hypothesis (EMH), which states that stock prices already reflect all available information, making consistent outperformance nearly impossible (Fama E. , 1970). The EMH builds on the DDM, but it adds “given all information available at time t=0”, which is referred to as I_0, in Equation 2.

$$P_0 = \sum_{t=1}^{\infty} \frac{Div_t}{(1+r)^t} | I_0$$

Equation 2: EMH

This implies that markets use all available information when predicting the future dividends of a company. If the EMH is true, it would be impossible to make excess returns by trading on already available information (Brealey, 2022). Since a lot of anomalies and market inefficiencies seemed to exist, alternative models to EMH were constructed to explain why and how these anomalies occurred, such as (Sharpe, 1964) with the Capital Asset Pricing Model (CAPM) and (Ross, 1976) with the Arbitrage Pricing Theory (APT), ultimately to predict excess returns of stocks.

3.2 CAPM

One of the very first asset pricing models contradicting the EMH was the CAPM, from (Sharpe, 1964). This model starts with the assumption that riskier stocks should have higher returns than stocks that have a lower amount of risk (Sharpe, 1964). An important parameter here is the β, which is the sensitivity of a stock to market movements (Brealey, 2022). In other words, how much a company’s stock goes up, on average, when the market goes up by 1%. Stocks with βs higher than 1.0 move more than one-for-one with the market, and stocks with βs lower than 1.0 move less than one-for-one with

the market. Since riskier stocks require a higher return and market risk is measured by β, the following formula was formulated:

$$r_i - rf = \beta_i(r_m - rf)$$

Equation 3: Risk premium

Equation 3 shows that the expected return on a stock minus the risk-free rate, so the expected risk premium of that stock, should equal β times the expected market risk premium (Brealey, 2022). With the market risk premium or the equity risk premium being defined as the expected return on the market minus the risk-free rate (Sharpe, 1964). Equation 4 shows the implication of this for the return of a stock:

$$r_i = rf + \beta_i(r_m - rf)$$

Equation 4: Return of a stock

In this thesis, we use the 10-year yield on U.S. treasury bonds as the risk-free rate, since we assume these to be risk-free (Brealey, 2022).

However, CAPM is criticised on its ability to predict the expected returns of stocks. Some studies such as (Barillas & Shanken, 2018) and (Wang, 2024) developed models that outperform CAPM, but also earlier studies like (Fama & French, 1992) who developed the Fama-French model, and (Alexis Akira Toda, 2017) which assesses CCAPM, have shown that CAPM does not fully predict expected returns of stocks.

Some more sophisticated versions of CAPM have been developed, such as CCAPM, the Consumption Capital Asset Pricing Model by (Breeden, Gibbons, & Litzenberger, 1989). This is an extension of CAPM that uses a consumption β instead of the market β. The consumption β is the coefficient of a regression of an asset's returns and consumption growth (Breeden, Gibbons, & Litzenberger, 1989). Another extension of CAPM is the ZCAPM which uses a zero-β portfolio as a benchmark for asset pricing, and has been proven to outperform CAPM in predicting excess returns (Kolari, Liu, & Huang, 2021).

3.3 Fama-French

CAPM states that returns are uncorrelated with firm-specific characteristics, but this implication of CAPM has also been criticized (Sharpe, 1964). The first real evidence of firm-specific characteristics that were able to predict outperformance of stocks, were so-called SMB[8] (small-firm minus big-firm) and HML[9] (high minus low book-to-market value) factors (Fama & French, 1992). The first Fama-French model is the three-factor model (FF-3), see Equation 5. The first factor in this model is the same as in CAPM, namely the market risk: $r_m - rf$, which is the excess return of a market portfolio over the risk-free rate (Brealey, 2022). The second and third factors are the SMB and HML factors. The way these factors were computed was by constructing different portfolios based on the market capitalization and the book-to-market value (Fama & French, 1992). In FF-3 this was done by creating ten deciles on size and ten on book-to-market value, these combinations resulted in one-hundred different portfolios based on the two factors. Furthermore, the authors measured the average monthly returns of these portfolios from 1963 to 1990, and a large difference was found between the portfolios with small-firms with low book-to-market value and the large firms with high book-to-market value (Fama & French, 1992).

[8] SMB implies that small firms outperform large firms.

[9] HML implies that companies with a high book value compared to its market value (value stocks) outperform companies with a low book-to-market value (growth stocks).

$$r_i - rf = \beta_{1,i}(r_m - rf) + \beta_{2,i} * SMB + \beta_{3,i} * HML$$

Equation 5: Three-factor model (FF-3)

In Equation 5 the βs measure the sensitivity of a specific stock to a certain factor, and the SMB and HML are the size and value factors (Fama & French, 1992). SMB is the size factor, which represents the historical excess returns of small cap stocks over large cap stocks. And HML being the value factor, representing the historical excess returns of value stocks over growth stocks.

The βs or coefficients in Equation 5 are estimated using an Ordinary Least Squares (OLS) regression. OLS is a statistical method that estimates the relationship between a dependent variable and one or more independent variables by minimizing the sum of the squared differences between the observed and predicted values (Wooditch, Johnson, Solymosi, Ariza, & Langton, 2021). In Equation 5, the regression estimates how well the SMB and HML factors explain the variation in excess stock returns (Fama & French, 1992). By applying this method, the model calculates the coefficients, which allow for interpretation of the explanatory power of the HML and SMB coefficients.

Upon the proposal of Equation 5, extended versions of FF-3 have been published. Momentum has been added to this model as a fourth factor in the Carhart model, which added a WML factor (winners minus losers) which explained the outperformance for previous 'winners' (Carhart, 1997). Some years later Eugene Fama and Kenneth French, the authors of FF-3, proposed their extended version namely the FF-5 model, which includes two extra factors, namely profitability and investment (Fama & French, 2015).

3.4 Arbitrage Pricing Theory

Another theory that contradicts CAPM is the Arbitrage Pricing Theory (APT). This general model states that returns of a stock depend on several risk factors (Ross, 1976), which we show in Equation 6.

$$r_i = a_i + \beta_{1,i}(risk\ factor\ 1) + \beta_{2,i}(risk\ factor\ 2) + \beta_{n,i}(risk\ factor\ n) + noise$$

Equation 6: APT

Where n can take many values depending on how many risk factors have been modelled. The APT states that the expected risk premium on a stock depends on the expected risk premium associated with each risk factor and the stock's sensitivity to each of these risk factors (Brealey, 2022). This yields the formula from Equation 7 for the expected risk premium:

$$r_i - rf = a_i + \beta_{1,i}(risk\ factor\ 1 - rf) + \beta_{2,i}(risk\ factor\ 2 - rf)$$
$$+ \beta_{n,i}(risk\ factor\ n - rf) + noise$$

Equation 7: Risk premium according to APT

It is important to highlight that APT itself does not state what these risk factors are, but it serves as a general model for later research (Brealey, 2022).

3.5 Other Models

While the previously described models serve as foundational theories for asset pricing, numerous extensions and alternative models have been developed, all aiming to improve the prediction of excess returns. One of these models is the Stochastic Discount Factor (SDF), which states that the price of an asset today is equal to the expected value of its future payoff, discounted by a stochastic discount factor (Cochrane, 2001). The SDF explains why stocks that have a high covariance with the economy have lower expected returns, and it is also the basis for the economic theory of the marginal rate of substitution. Additionally, the previously mentioned CCAPM is built upon the SDF framework. The drawback of this approach is that estimating the SDF function is significantly more challenging than using the market β from CAPM (Cochrane, 2001). Although a large number of models have been

developed, most of these models have already been outperformed by newer extensions or completely new models that provide better estimates of stock outperformance. Furthermore, recent studies that compare and evaluate different factor models and propose improvements, include (Avramov, Cheng, Metzker, & Voigt, 2023), (Fletcher, 2018) and (Barillas & Shanken, 2018).

3.6 Machine Learning

Since model development occurs at a fast pace, and new advanced extensions of asset pricing models come out almost every year, the development of having self-learning models is a great invention for asset pricing theory. ML and its ability to process vast amounts of data and identify complex patterns is a suitable tool for asset pricing like done by (Wang, 2024). Given that asset pricing relies on extensive historical data and is aimed at forecasting future returns, ML techniques provide new insights and ways to discover relationships that previous asset pricing models could not (Wang, 2024).

ML can be used in all aspects of asset pricing, from predicting returns based on historical data to explaining and recalculating the factors or firm-specific characteristics of already established models. ML can also be used to identify new factors which affect stock prices, or find complex relationships between certain factors (Meiyun Wang, 2024).

One of these ML applications is NN, this is an ML technique that is inspired by the human brain. Neural networks consist of interconnected nodes or neurons that process and transmit information (Wang, 2024). These networks are organized into layers: the input layer receives the initial data, the hidden layer or layers process the data, and the output layer produces the final results (Wang, 2024), we depict these layers in Figure 4. The connections between the neurons have weights that determine the strength of the signals, and the neural network learns by constantly adjusting these weights to improve accuracy. (Wang, 2024) used NN to develop a new asset pricing model based on macroeconomic and firm-specific variables. A common drawback of NN is overfitting, where models perform well on training data, but underperform on the test data. This means the model is not able to generalize training data onto a new, unseen dataset (Wang, 2024).

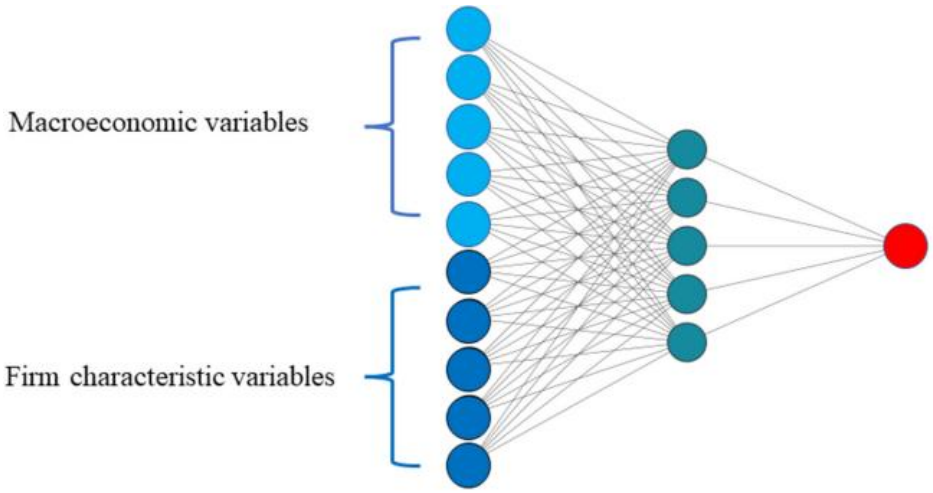

Figure 4: Layers of a neural network (Wang, 2024)

Another ML technique that is frequently used in asset pricing is RF. This is a technique that combines several decision trees into one model (Kaczmaczyk & Hernes, 2020). The model learns and identifies patterns and relations in the data and these decision trees are used to classify output (for example in groups of stocks with low expected returns and groups with high expected returns). An advantage of RF compared to NN is that it has a lower chance of overfitting, since the RF averages all decision trees into one model (Qian & Zhang, 2025). Since the RF model relies on decision trees that split data based on

whether values are higher or lower than a certain threshold, it is also very robust against outliers, which are common in the LSEG dataset we use in this study.

3.7. Implications for Model Development

The pursuit of understanding and predicting stock performance and excess returns has led to the evolution of several asset pricing models, which started with simpler models and gradually involved into more sophisticated models using ML algorithms. For this study, we develop an OLS model using Fama-French characteristics as a benchmark model for FF-6, which extends the FF-5 model by adding a momentum factor, following the approach of (Carhart, 1997). This benchmark model predicts excess returns, defined as the difference between a stock's actual return and its expected return under the CAPM framework from Equation 4 (Sharpe, 1964).

To enhance the predictive ability of the model, we incorporate additional characteristics and develop two ML models: an RF and an NN model. We include an NN model in line with (Wang, 2024) and other recent financial studies. The RF model complements this by offering more robustness to outliers and having a lower chance of overfitting. Finally, we construct an ensemble model, combining the predictions of the RF and NN models.

4 Model Development

This chapter corresponds to the fifth phase of the problem-solving approach, the model development phase. Figure 5 illustrates the overall structure of this phase through a flowchart that presents each step taken from the literature review to the final model evaluation and implementation.

This chapter begins with data collection in Section 4.1, consisting of characteristic selection, outlier removal, handling of missing values, and data scaling. In Section 4.2, we introduce a baseline model using an OLS regression. This model is based on the Fama-French six factor model (FF-6) (Fama & French, 2018). This FF-6 based model serves as a comparison for evaluating the more advanced models developed later in this study.

Section 4.3 explains the ML algorithms used to develop the more extended models. First, we develop an RF model, followed by an NN model. For each algorithm, we construct two versions: one using only stock-specific characteristics, and another that incorporates macroeconomic characteristics as well. Finally, we construct an ensemble model which combines both the RF and the NN models into a single predictive model.

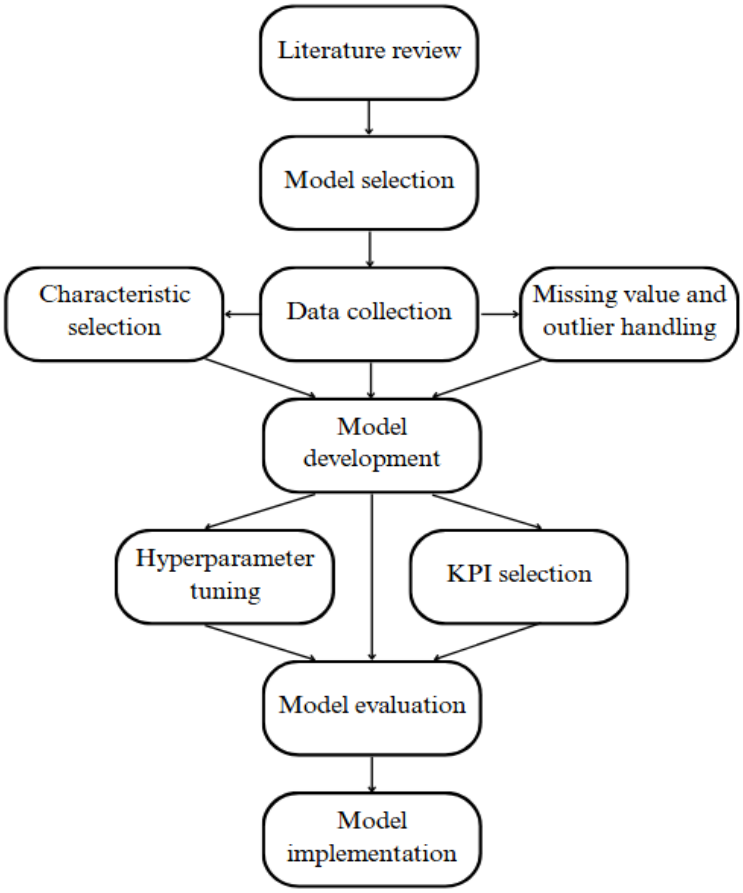

Figure 5: Flowchart of model development

4.1 Data Collection

The data used for the development of the models consist of all S&P 500 stocks from 2000 until 2025, following the approach of several other studies such as (Qian & Zhang, 2025). We selected the S&P 500 since this is the largest and most influential stock index in the world, and it is used as a benchmark by investors all over the world (S&P Dow Jones Indices S&P500, 2025). The S&P 500 represents the top 500 companies from the U.S. stock market and offers by far the most complete data for many characteristics, which enhances the quality of the model development. Many earlier studies such as (Ross, 1976), (Fama & French, 1992) and (Carhart, 1997) also limited themselves to stocks from the U.S. For StockWatch, the model is implemented on stocks from the U.S., as well as European (mainly Dutch) stocks.

For selecting a suitable timeframe, the trade-off here was having more years of data with more missing values or opting for a shorter timeframe with more complete data. From 2000 onwards the LSEG data was relatively complete, so in total 25 complete years of data are used as done by (Fabian Hollstein, 2020) and (Fama & French, 1992). This is more than enough to have a reliable amount of data, although using additional years would be preferred for more reliability. The dataset is structured such that, for each stock and for each year there is a corresponding value for each characteristic. Some of these characteristics could directly be obtained from the LSEG database, while we had to compute others using Python. To ensure the most up-to-date input, we use April 18, 2025 as the endpoint for all input data. For consistency and to create a fair starting point, we set April 18, 2000 as the starting point, ensuring exactly 25 years of data.[10]

4.1.1 Characteristics

To select the stock-specific characteristics to use in this model, we started with the FF-6 model (Fama & French, 2018), which uses the SMB and HML factors as explained in Section 3.3, plus a profitability and investment factor. In our study we include the SMB factor by using the market capitalization, and the HML factor by means of the Price-to-Book value. We replaced the profitability factor with Gross Profit as done by (Novy-Marx, 2013) and this study uses the Return on Invested Capital (ROIC) as investment factor. Lastly, we use the stock price Momentum, which is one of the most well-known market anomalies (Carhart, 1997). This study includes the 6-month as well as the 1-year Momentum. Since these were not available in LSEG, we calculated these Momentum variables in Python, using Equation 8 and 9, where *t* is in years.

$$Momentum\ 1y_t = \frac{Total\ return\ index_t - Total\ return\ index_{t-1}}{Total\ return\ index_{t-1}}$$

Equation 8: 1-YearMomentum formula

$$Momentum\ 6m_t = \frac{Total\ return\ index_t - Total\ return\ index_{t-\frac{1}{2}}}{Total\ return\ index_{t-\frac{1}{2}}}$$

Equation 9: 6-Month Momentum formula

This study uses the Total Return Index for computing the Momentum values, this means that the total shareholder return is used, consisting of stock price appreciation and dividends. Figure 6 presents a histogram with all raw data for the 1-year stock returns.

[10] It is easily possible for StockWatch to refresh all data sheets (for example every quarter) to ensure the most up to date info is used for their model. To keep all evaluation and analysis fair, this study does not refresh data after April 18, 2025.

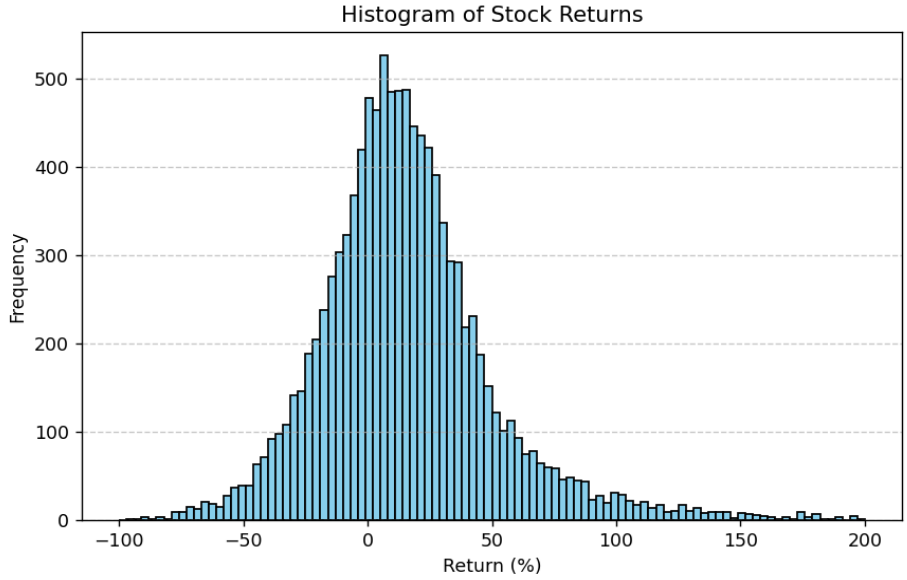

Figure 6: Histogram of 1-year Stock Returns

These two Momentum factors calculated from the Total Return Index, combined with four factors from the FF-5 model (excluding the β-factor, since we study excess returns), form the six factors used in the basic OLS model, which serves as a benchmark for the more sophisticated models later.

The ML models developed in this study are capable of working with many more variables than just these six, therefore we selected an extended list of company-specific as well as macroeconomic characteristics. These additional characteristics are based on studies from (Avramov, Cheng, Metzker, & Voigt, 2023), (Wang, 2024) and (Gu, Kelly, & Xiu, 2020). Not all these characteristics were used, after testing and assessing whether these variables or a close enough replacement was available in LSEG, we selected a final set of 19 stock-specific and 5 macroeconomic characteristics. Table 1 shows this final set of characteristics along with their abbreviations used in this study.

Company-specific characteristics	**Abbreviation**
6-Month Momentum* [11]	Momentum6M
1-Year Momentum*	Momentum1Y
Return on Invested Capital*	ROIC
Gross Profit Margin*	GrossProfitMargin
Dividend Yield	DividendYield
Market Capitalization*	MarketCap
12-Month Forward Earnings Yield	ForwardEarningsYield
Price-to-Book Value*	PriceToBook
Dividend per Share 5-year CAGR	DPSCAGR5y
Earnings per Share 5-year CAGR	EPSCAGR5y
Environmental, Social, and Governance Score	ESG
Enterprise Value to 12-month forward EBITDA Ratio	EVEBITDA
12-Month forward Net Debt to EBITDA Ratio	DebtEBITDA
Return on Assets	ROA
Net Profit Margin	NetProfitMargin
Dividend Payout Ratio as percentage of Earnings	PayoutRatio

[11] Characteristics indicated with a * are used in the FF-6 OLS model.

Capital Expenditures as percentage of Total Sales	Capex
Trading Volume (7-day average)	TradingVolume
Current Ratio	CurrentRatio
Macroeconomic characteristics	
US Consumer Price Index all Items Annual Inflation Rate	CPI Annual Inflation Rate
US Consumer Confidence Index	Consumer Confidence Index
US Unemployment Rate	Unemployment Rate
US Federal Funds Target Rate	Federal Funds Target Rate
US Personal Consumption Expenditures Price Index excluding Food and Energy	PCE Index

Table 1: Characteristics used in this study

We selected these characteristics to fully capture the key financial aspects of a firm. They include profitability measures such as Return on Assets (ROA) and Gross Profit Margin; investment-related factors like ROIC and Capital Expenditures; valuation metrics including the Price-to-Book value and Forward Earnings Yield; leverage indicators such as the Net Debt to EBITDA ratio, as well as stock-related features like Momentum and Trading Volume.

This final set of characteristics incorporates forward-looking metrics such as the 12-month Forward Earnings Yield and 12-month Forward Net Debt to EBITDA ratio, which are estimated using analyst expectations. We also included backward-looking metrics such as the Dividend and Earnings per Share growth over the last 5 years, which is expressed as a compound annual growth rate (CAGR). To avoid look-ahead bias, only data that is available at or before time t is used to predict year $t+1$ returns.

This thesis derives the macroeconomic characteristics from the same studies as the stock-specific characteristics. We designed these macroeconomic characteristics to capture the broader economic environment of the U.S. These include the Consumer Price Index (CPI) as a measure of inflation, the Unemployment Ratio to represent labour market conditions and the Federal Funds Target Rate representing the stance of monetary policy in the U.S. Lastly, we used two consumer-related variables: the Personal Consumption Expenditure (PCE) Index, reflecting consumer spending trends, and the Consumer Confidence Index, indicating household confidence. All these macroeconomic indicators are updated annually and are consistent across all stocks in a given year.

Table 2 shows some descriptive statistics for the stock-specific characteristics used in this study. This table shows the raw values, before any outliers were removed from the model.

Characteristic	**Mean**	**Min**	**Q1**	**Median**	**Q3**	**Max**
Momentum6M (%)	10.50	-85.44	-1.83	8.79	20.68	760.84
Momentum1Y (%)	17.88	-96.59	-4.89	12.58	32.16	6180.89
ROIC (%)	12.51	-1092.34	5.67	10.21	16.74	3877.00
GrossProfitMargin (%)	40.49	-3878.08	27.65	41.20	57.11	100.00
DividendYield (%)	1.74	0.00	0.00	1.42	2.70	15.88
MarketCap	4.34e+10	1.14e+07	6.68e+09	1.57e+10	3.69e+10	3.79e+12
ForwardEarningsYield (%)	5.37	-2568.78	4.18	5.75	7.58	353.57
PriceToBook	3.31	-2147.35	1.73	2.83	4.93	1164.93

DPSCAGR5y (%)	5.83	-100.00	0.00	5.11	12.17	168.67
EPSCAGR5y (%)	11.73	-100.00	1.86	9.96	19.08	261.33
ESG	53.01	0.60	37.20	55.08	69.64	95.16
EVEBITDA	11.94	-1276.92	7.97	10.63	14.66	1022.03
DebtEBITDA	1.20	-321.76	-0.13	1.04	2.35	72.82
ROA (%)	7.37	-150.45	3.44	6.66	11.21	157.56
NetProfitMargin (%)	-36.08	-246377	4.59	9.51	16.21	422.31
PayoutRatio (%)	25.44	0.00	0.00	21.50	41.83	100.00
Capex (%)	72.86	0.00	2.32	4.25	10.00	516919.50
TradingVolume	1.34e+06	38.00	2.42e+05	5.93e+05	1.24e+06	1.83e+08
CurrentRatio	1.95	0.14	1.04	1.45	2.19	529.42

Table 2: Descriptive statistics for characteristics used in this study

4.1.2 Multicollinearity

Figure 7 shows the correlation between all stock-specific characteristics. ROIC and ROA have a high correlation coefficient (0.89). Additionally, the Payout Ratio and the Dividend Yield (0.75) and Market Capitalization and Trading Volume (0.75), also show high correlations.

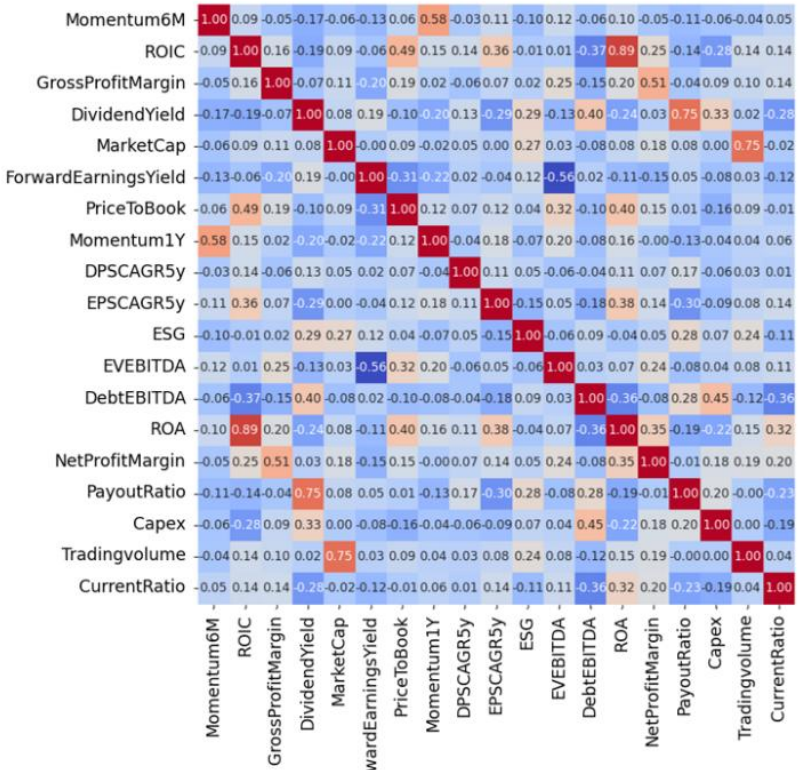

Figure 7: Correlation matrix

While multicollinearity might pose some challenges in interpreting OLS values, our main focus lies on the ML models, which do not have problems with multicollinearity. Given that all of these characteristics do measure different features of a company and they all provide unique information for the model, this

study keeps all variables in the model (Ghanoum, 2021). Additionally, we calculated Variance Inflation Factors, and these values were low (< 5) as can be seen in Appendix C.

4.1.3 Outliers and Missing Value Handling

For the NN and the OLS model, we removed outliers from the dataset since these models are very sensitive to extreme outliers, resulting in a decrease in forecasting performance. (Khamis, Ismail, Haron, & Mohammed, 2005). We chose this approach because certain financial ratios like the EPS growth were skewed by a very low starting point, yielding an unreasonably high growth percentage. The RF model is more robust to outliers and performed well on the full dataset, therefore we retained the outliers in this model.

We removed outliers according to financial knowledge and talks with experts from StockWatch (Koerts, Outlier removal, 2025). Examples of such outliers include a 1-year stock return worse than -95% or better than +200%, or negative Price-to-Book values. A complete overview of all outlier conditions for the NN and OLS model is given in Table 12 in Appendix D, along with the number of outliers that were removed by the given conditions.

We handled missing values[12] according to the SimpleImputer class from scikit-learn (Pedregosa, 2011). Each feature's missing value is replaced by the median value of that feature calculated across the training data like done in (Gu, Kelly, & Xiu, 2020) and (Leippold, Wang, & Zhou, 2022). We preferred the median values over the mean values because they are more robust to skewed distributions and outliers. This approach of imputation allows observations with limited missing values to still be included in the model, thereby preserving a larger number of observations, improving the robustness of the model.

In this study, we adopt threshold-based missing value removal (Hvitfeldt, 2024). If one observation has more than three missing values, which means more than 20% of the data is missing for that specific observation, we exclude this entire observation from the model, as this would impose that a significant portion of the observation's data is imputed, which increases the risk of introducing unwanted noise into the model (Hvitfeldt, 2024). In context of this study, one observation corresponds to one year of data for one specific stock, consisting of all 19 stock-specific characteristics.

4.1.4 Scaling

Before training the NN model, we standardized all input features using z-score normalization via the StandardScaler from scikit-learn (Pedregosa, 2011), as done by (Wang, 2024). This transforms each feature by subtracting its mean and dividing by the standard deviation from the training set (Pedregosa, 2011). This ensures that all features have a zero mean and a variance of one. We scaled the target variable, the 1-year excess return, using this same method as well. We reversed this transformation to interpret the final results in original units. For NN models, scaling is necessary to train and forecast more effectively, especially in time-series and dynamic forecasting models (Bhanja & Das, 2019). Since RF is a tree-based model, scaling is not necessary as decisions of these trees are made by comparing values and their ordering to certain thresholds, not by computing absolute distances between feature values (Biau & Scornet, 2015). This means that in this study we scaled all input variables for the NN models, while keeping the values in original units for the RF models.

4.2 OLS Model

We designed the OLS model to predict the excess returns of individual stocks. We define the excess return of stock i in year t according to Equation 10, consistent with the CAPM framework (Sharpe, 1964).

[12] Missing values occur in the LSEG database because certain stock data was not available in specific years. This could be due to companies not having reported this data, or because the company was not yet publicly traded in the given year.

$$Excess\ return_{i,t} = return_{i,t} - \beta_{i,t} * market\ risk\ premium - rf_t$$

Equation 10: Excess return formula

Here the market risk premium refers to the excess return of the market above the risk-free rate, as explained in Section 3.2. In this formula, we adjust the stock's return for the stock's exposure to market risk, reflected by its β value and the risk-free rate in a given year. In theory, if CAPM holds and the market risk premium used is correct, the average excess return of any stock should be zero (Brealey, 2022).

In this study we choose β as the risk measure, following CAPM (Sharpe, 1964) and aligned with the methodology StockWatch uses in their analyses. The β we use is the 5-year β of the stock, from LSEG. Since we use β to adjust for market risk when calculating the excess returns, the resulting excess returns are already risk-adjusted. We assume the U.S. market risk premium to be 6% based on the long-run average estimated by (Ritholtz Wealth Management, 2024) over the past 30 years. We use the U.S. 10-year treasury yield as risk-free rate, also from the LSEG database. With this information the excess return for each stock is calculated annually, using Equation 10.

The objective of the models in this study is to predict these excess returns using the characteristics listed in Table 1, ultimately to be able to select stocks with the highest excess returns, to result in risk-adjusted outperformance for StockWatch. The model's goal is to predict excess returns for the upcoming year based on the observed characteristic values in the current year. To achieve this, we trained the model on historical data, using all observations in the training set to learn relationships between the predictor variables and excess returns. These learned patterns are then applied to the test set to generate out-of-sample predictions. We choose to predict yearly excess returns rather than monthly returns, as this aligns with StockWatch's methodology of longer-term stock selection. Additionally, companies typically report financial data quarterly or sometimes semi-annually, making monthly predictions less suitable. Our model can be viewed as a time-series model, with Momentum1Y as lagged indicator since it reflects the information of the predictor variable from the previous year (although excess return, being risk-adjusted, is not exactly identical to stock performance from the prior year).

As a benchmark for the developed models, we first developed a simple OLS regression using six characteristics similar to those of the FF-6 model, to enable fair comparison. Equation 11 shows the formula for this OLS model, where β_0 represents the intercept and $\varepsilon_{i,t}$ the error term capturing the unexplained variation.

$$\begin{aligned} &Excess\ return_{i,t} \\ &= \beta_0 + \beta_1 * ROIC_{i,t} + \beta_2 * GrossProfitMargin_{i,t} + \beta_3 * MarketCap_{i,t} + \beta_4 \\ &* PriceToBook_{i,t} + \beta_5 * Momentum1Y_{i,t} + \beta_6 * Momentum6M_{i,t} + \varepsilon_{i,t} \end{aligned}$$

Equation 11: OLS FF-6 regression formula

Unlike the Fama-French framework explained in Section 3.3, which relies on factor portfolios to estimate the βs and primarily aims to explain past return anomalies and factors, this study applies a direct OLS regression at individual stock levels. The focus for StockWatch lies on predicting future excess returns and not on explaining historical factor premia. And while the Fama-French method offers insights into the historical performance of a few factors, it is not suited for modelling future returns using a broad set of characteristics like done in our study. Moreover, the Fama-French framework is linear and limited in flexibility, making it less compatible with the ML models later in this study, which aim to explore more complex and non-linear relationships. Therefore, we do not make use of the factor portfolio method to calculate the βs, but we use a direct OLS regression, which we later on extend into ML models.

Therefore, we solved this regression model using a regular OLS regression in Python for the βs, which results in Equation 12 for a stock's predicted excess return.

$$\begin{aligned} &Excess\ return_{i,t} \\ &= 0.0423 + 0.0029 * ROIC_{i,t} + 0.0004 * GrossProfitMargin_{i,t} + 2.912 * 10^{-11} \\ &* MarketCap_{i,t} - 0.0034 * PriceToBook_{i,t} - 0.1159 * Momentum1Y_{i,t} + 0.0167 \\ &* Momentum6M_{i,t} + \varepsilon_{i,t}{}^{13} \end{aligned}$$

Equation 12: OLS FF-6 formula

This OLS model explains and quantifies the relationship between the six stock-specific characteristics and their influence on expected excess returns. When trained on data from 2000-2019 and evaluated out-of-sample on 2020-2024, the FF-6 OLS model performs quite poorly with an R-squared value of 0.73%, indicating limited predictive power. We compare this model against the ML models in Chapter 6, to assess if these more flexible ML models can deliver improved performance.

To explore whether adding additional characteristics enhances predictive ability, we constructed a second OLS model using all 19 stock-specific characteristics. This extended model achieved a significantly higher R-squared value of 3.15% in the same out-of-sample test, suggesting that incorporating a broader set of characteristics enhances explanatory power beyond the traditional FF-factors. The addition of the 13 additional characteristics yields an F-value of 3.60, with a p-value < 0.01, indicating that the addition of the extra characteristics is statistically significant at a 5% significance level. A table with summary statistics of both OLS models including p-values and the model F-test is provided in Appendix A.

4.3 Machine Learning Model

To further enhance the predictive ability of the model, we use ML. In the literature review done in Section 3.6, we identified RF and NN as the most suitable algorithms to use for these asset pricing models. These two algorithms can find complex non-linear relationships between variables as opposed to only linear relationships found in the OLS model. Moreover, RF and NN are able to capture interactions between the independent variables, something that a regression using OLS is not able to.

For the RF and NN models, we use the same dataset, with 19 stock-specific characteristics. Additionally, we constructed two models using the stock-specific characteristics as well as the 5 macroeconomic characteristics. This results in the following four models: RF, RF+Macro, NN and NN+Macro.

4.3.1 Test and Training Split

For the training and test split of the models, it is common to use 80% of the data for training and hold out 20% for testing (Géron, 2017). This means that for the development of the model, the training set consists of the years 2000-2019, which we subsequently test on the 2020-2024 period. This allows us to obtain initial out-of-sample results for the R-squared value of the models. We perform more rigorous out-of-sample and cross-validation testing in Chapter 5.

4.3.2 Model Architecture and Hyperparameters

RF model: We implement the RF model using the RandomForestRegressor from scikit-learn (Pedregosa, 2011). This regressor fits a number of decision tree regressors on various sub-samples of the dataset and then averages them to improve the predictive accuracy and to control overfitting (Pedregosa, 2011). Three important hyperparameters that determine the structure of this model are: n_estimators, max_depth, and max_features. The n_estimators parameter determines the number of trees in the forest, where more trees reduce variance of the model, but increase computational difficulty (Pedregosa, 2011). Max_depth is the maximum depth of one tree, deeper trees can find more complex

[13] MarketCap and Momentum6M are not significant at a 5% significance level.

relationships but might run into the problem of overfitting (Pedregosa, 2011). Max_features is the number of features which can be considered at a split of the decision tree, if max_features is set smaller than 1, not all features are taken into account at each split (Pedregosa, 2011).

NN model: We implement the NN model using the MLPRegressor also from scikit-learn. This regressor is a feedforward neural network that maps inputs to outputs using one or more hidden layers (Pedregosa, 2011). Three hyperparameters that define the structure and behaviour of this model are: hidden_layer_sizes, alpha, and learning_rate_init. The hidden_layer_sizes determines the number of layers and the number of neurons in each layer, where more neurons or adding additional layers allows the model to learn more complex relationships, but increases the training time and the chance of overfitting (Pedregosa, 2011). The alpha parameter is a regularization term that penalizes large weights in the model to reduce overfitting (Pedregosa, 2011). The learning_rate_init sets the initial step size in updating the weights, when setting this value lower, the convergence of the model will be slower, but training will be more stable (Pedregosa, 2011).

4.3.3 Hyperparameter Tuning

To optimize the hyperparameters explained in the previous section, we use RandomizedSearchCV, also imported from scikit-learn (Pedregosa, 2011). This randomized search method involves randomly sampling a specified number of combinations from a defined parameter space (Géron, 2017). This method allows us to find out the best combination of hyperparameters in an efficient way. This study evaluated combinations of hyperparameters on their R-squared value. We prefer the randomized search over the grid search, because of its ability to explore a high number of combinations (100 in this study), without having to evaluate every single combination (Pedregosa, 2011).

We selected the ranges of the hyperparameters from the scikit-learn library. Additionally, we tested commonly used parameter ranges in Python through observation and careful inspection into signs of underfitting or overfitting (Pedregosa, 2011). Table 3 shows the full search ranges and optimal values identified for each model. The last column shows the out-of-sample R-squared obtained in the same way as done for the OLS model, we interpret these results in Chapter 5.

Model	**Hyperparameter**	**Range**	**Optimal value**	**R-squared**
RF	N_estimators	{100,150,200,250,300,350,400,450,500,550,600,650,700,750}	500	8.0%
	Max_depth	{5,10,15,20,25,None}	25	
	Max_Features	{sqrt, log2, None}	sqrt	
RF + MACRO	N_estimators	{100,150,200,250,300,350,400,450,500,550,600,650,700,750}	300	-10.5%
	Max_depth	{5,10,15,20,25,None}	5	
	Max_Features	{sqrt, log2, None}	sqrt	
NN	Hidden_layer_sizes	{(16,), (32,), (64,), (32,16), (64,32)}	(32,)	7.5%
	Alpha	{0.0001,0.0005,0.001,0.005,0.01,0.05,0.1,0.5,1,1.5,2,3,5}	1	
	Learning_rate_init	{0.0001,0.0005,0.001,0.005,0.01,0.05}	0.0001	
NN + MACRO	Hidden_layer_sizes	{(16,), (32,), (64,), (32,16), (64,32)}	(16,)	-9.7%
	Alpha	{0.0001,0.0005,0.001,0.005,0.01,0.05,0.1,0.5,1,1.5,2,3,5}	2	
	Learning_rate_init	{0.0001,0.0005,0.001,0.005,0.01,0.05}	0.0001	

Table 3: Hyperparameter ranges used

4.3.4 Ensemble Model Simple Average

To further enhance the predictive ability of the model, we construct an ensemble model. This model uses both the RF and the NN model, and predicts excess returns based on the weighted average of the two models. Since both models perform almost equally well individually (R-squared of 8.0% and 7.5%), we start with a weight of 50% for both models as shown in Equation 13. We did not use the models with macroeconomic characteristics since the predictive ability of these models was found to be significantly lower than those without these characteristics.

$$Excess\ return = 0.5 * Excess\ return(RF) + 0.5 * Excess\ return(NN)$$

Equation 13: Ensemble model

For this model, we performed hyperparameter tuning with the same ranges from Table 3, Table 4 shows the optimal values identified here.

Hyperparameter	**Optimal value**	**R-squared**
RF		9.6%
N_estimators	200	
Max_depth	None	
Max_features	None	
NN		
Hidden_layer_sizes	(32,)	
Alpha	0.0005	
Learning_rate_init	0.0001	

Table 4: Hyperparameters and performance of the ensemble model

4.3.5 Ridge Stacking

An effective approach to determine the optimal weights for combining predictions from two ML models is through a Ridge regression, a linear stacking technique. Ridge regression is a linear regression model which includes an L2 regularization term. It tries to shrink the error term plus a penalty term to shrink the size of the coefficients (Pedregosa, 2011). Equation 14 presents the objective function of the Ridge approach, where α controls the strength of the regularization, the βs are the model coefficients and the $\hat{y}$s are the predicted values (Pedregosa, 2011).

$$Objective\text{: } \min \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \alpha \sum_{j=1}^{p} \beta_j^2$$

Equation 14: Ridge formula

In this model, the Ridge regression assigns a 93.7% weight to the RF model and a 18.5% weight to the NN model, using an alpha value of 1. As expected in a Ridge regression, these weights do not sum up to one, since the optimization is unconstrained. The resulting R-squared value was 7.2%, which did not improve upon the simple average model.

4.3.6 Spearman IC

Another approach, commonly used in finance, to determine optimal weights for combining ML models is by using the Spearman Information Coefficient, which uses the Spearman rank correlation (Spearman, 1904). We develop a Python model that evaluates all combinations of weights of the RF and NN models and selects the best combination based on the correlation between the predicted and actual values (Spearman, 1904). The final result assigns a weight of 87% to the RF model and 13% to the NN model and achieves an R-squared value of 7.1% on the test set.

5 Model Evaluation

As we outline in Section 5.1 on the evaluation procedure, we validate all models thoroughly. In Section 5.2, we evaluate the models using an out-of-sample test and select the best performing model. To assess ranking performance, we divide stocks into deciles based on predicted returns. From these decile portfolios we analyse the realized returns and Sharpe ratios. In Section 5.3, we evaluate the selected model using a Monte Carlo cross-validation across the entire 25-year dataset. This ensures assessment of the model's robustness over both time and cross-sectional variation. Finally, Section 5.4 examines Feature Importances to better understand the individual characteristics used and their role in the predictive model. This thesis adopts two different methods for that: the Feature Importance scores from scikit-learn (Pedregosa, 2011) and SHAP (SHapley Additive exPlanations) values from (Lundberg & Lee, 2017).

5.1 Evaluation Procedure and Metrics

The main evaluation metric used to evaluate and quantify the predictive ability of the developed models is the R-squared value, as employed in almost all comparable studies such as (Wang, 2024), (Gu, Kelly, & Xiu, 2020) and (Fama & French, 2015). R-squared measures the goodness-of-fit of the model and represents the proportion of variance in the dependent variable that is explained by the independent variables (Géron, 2017). In stock return prediction, the focus is not to precisely forecast exact returns, which is an unrealistic objective given the inherent noise that is present in financial markets, but rather to find systematic relationships between stock characteristics and excess returns. R-squared is therefore a suitable metric for evaluating the explanatory power of our model. Equation 15 shows the formula of the R-squared value, where y_i is the actual value of the target variable (excess return), $\hat{y}_i$ is the predicted value of excess returns from the model, and $\bar{y}$ is the mean of all observed excess return values.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Equation 15: R-squared formula

In addition to R-squared, we use two other evaluation metrics for evaluating the model: the Mean Absolute Error (MAE) and the Mean Squared Error (MSE). The MAE calculates the average absolute difference between the observed and predicted values, while the Mean Squared Error (MSE) computes the average of the squared differences (Géron, 2017). Their respective formulas are shown in Equation 16 and 17.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Equation 16: MAE formula

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Equation 17: MSE formula

We evaluate these metrics on an out-of-sample basis, using the 80/20 train/test split, as we explained in Section 5.3.1 (Géron, 2017). This means using the first 80% of the available years for training the model, while reserving the remaining 20% for testing the model. This setup simulates actual model implementation, by forecasting on a previously unseen period.

In addition to this time-based split, we perform a Monte Carlo cross-validation. Where in each iteration of this procedure, we randomly select 80% of the stocks for training, and we test the model on the remaining 20% of the stocks. This method allows for testing the model across the full 25-year time period, providing a robust assessment of the model's predictive ability over different market conditions.

Furthermore, we construct decile portfolios for both these testing methods, by ranking stocks based on their predicted excess returns. Subsequently, we analyse the actual realized returns of each decile to assess the model's raking ability. If the top deciles consistently outperform the lower deciles, this indicates that the model is successful in identifying outperforming stocks.

From these decile portfolios, another important evaluation metric is calculated: the Sharpe ratio. The Sharpe ratio is a measure of an investment's risk-adjusted performance, calculated by comparing this risk-adjusted performance with the risk-free rate (Sharpe, 1966). We calculate the Sharpe ratio using Equation 18, where R_p is the return of the portfolio, rf is the risk-free rate and σ_p is the standard deviation of portfolio returns (Sharpe, 1966).

$$Sharpe\ ratio = \frac{R_p - rf}{\sigma_p}$$

Equation 18: Sharpe ratio formula

The final performance metric, which we also calculate based on the decile portfolios is the Sortino ratio. The Sortino ratio is similar to the Sharpe ratio, however it only considers downside risk by replacing the overall standard deviation with the downside standard deviation (Red Rock Capital, 2010). This downside standard deviation is denoted as σ_{dp}, which measures only the volatility of returns falling below a certain threshold (Minimum Acceptable Return or MAR), for which we use the risk-free rate in this study. Equation 19 shows the formula we use for calculating the Sortino ratio. The key advantage of the Sortino ratio compared to the Sharpe ratio is that it only includes harmful downside volatility, while ignoring upside fluctuations (Red Rock Capital, 2010).

$$Sortino\ ratio = \frac{R_p - rf}{\sigma_{dp}}$$

Equation 19: Sortino ratio

5.2 Out-of-Sample Validation

Figures 8 and 9 show the results of the out-of-sample test using an 80/20 training/test split. The ensemble model where RF and NN both got a weight of 50% has the highest out-of-sample R-squared value of 9.6%. This model also has the lowest MAE value among all developed models of 0.260 and the second lowest MSE value of 0.135.

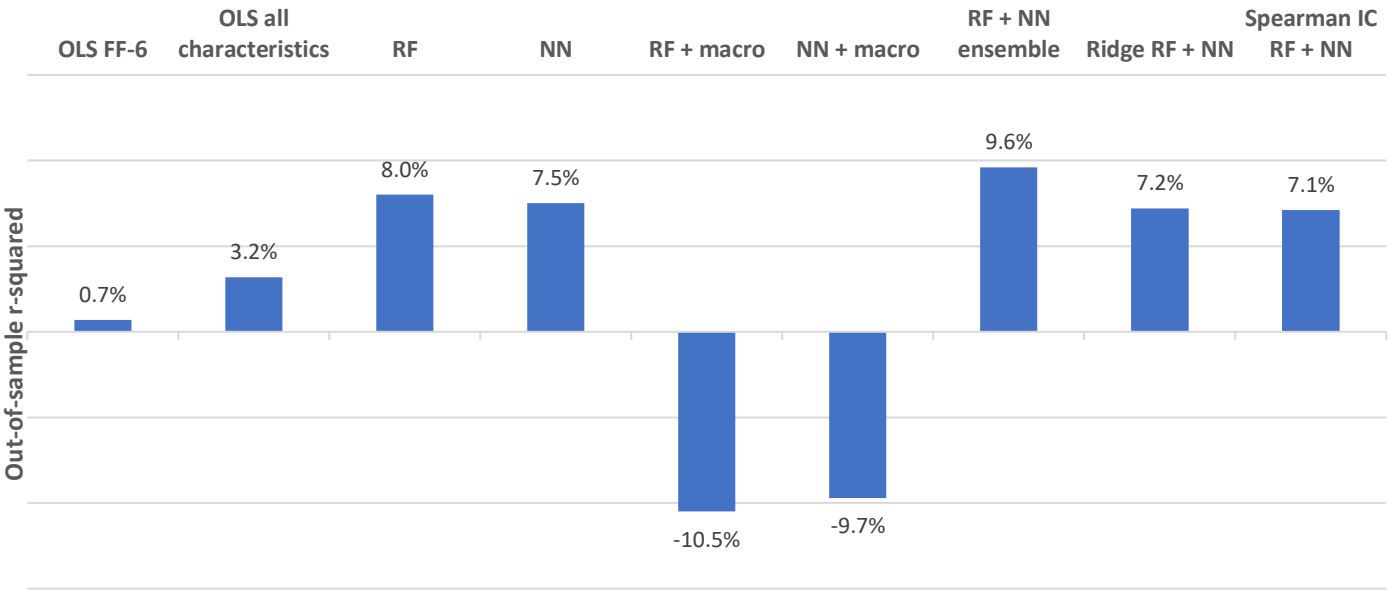

Figure 8: R-squared comparison out-of-sample

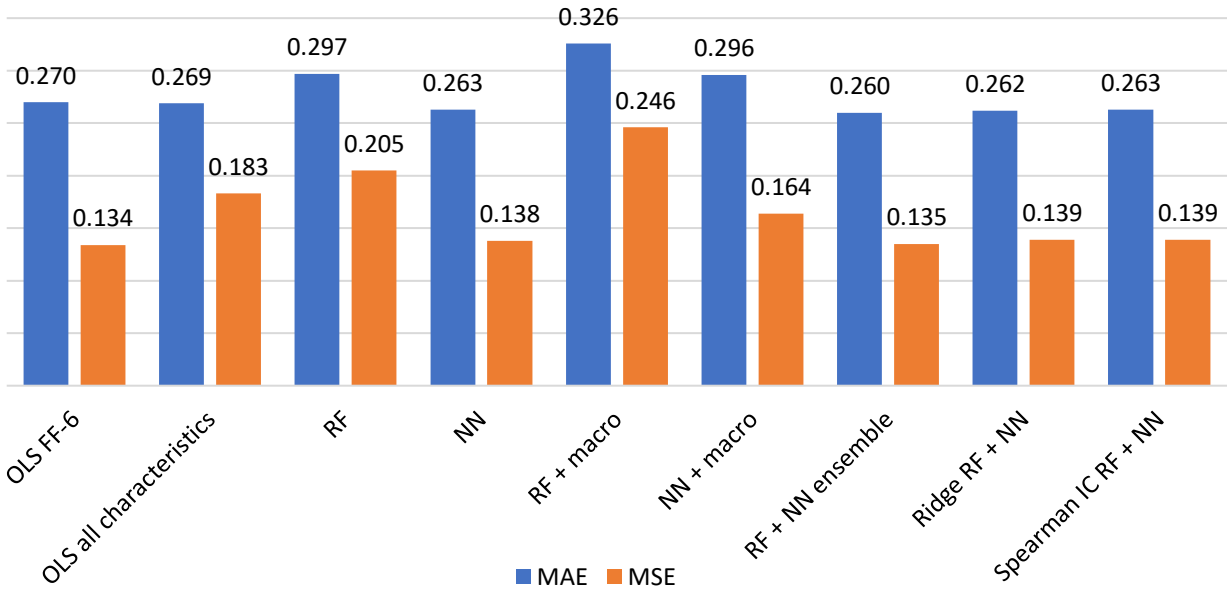

Figure 9: MAE and MSE comparison out-of-sample

The two models that included macroeconomic characteristics both produced a negative R-squared value on the test set, indicating they have performed worse than simply predicting the mean of the target

variable. This suggests that the model failed to identify patterns beyond the training data and may have been affected by overfitting. In the case of the RF + Macro model, the best performing configuration of hyperparameters selected a Max_Depth of only 5, which means that trees only grow up to 5 branches thus avoiding more complex relationships. Thus, regularization was already applied to prevent overfitting, but this did not result in a reliable performance. For the NN + Macro model, the Alpha value was 2, which would typically help to reduce overfitting by penalizing large weights. Despite attempts to improve the models by removing some macroeconomic characteristics and expanding the hyperparameter search space, no configuration achieved a positive R-squared. Furthermore, when sorting stocks into decile portfolios, no clear ranking became visible between the top and bottom deciles. As a result, we deemed the macroeconomic characteristics ineffective in this study and excluded those from further models. The ineffectiveness of the macroeconomic characteristics is in line with the EMH, which suggests that this public information is already reflected in the stock prices. However, this outcome contrasts with findings from other studies such as (Wang, 2024), where macroeconomic characteristics did show predictive power. One explanation for this contrast is the shorter time span of our study of 25 years, which included several macroeconomic shocks such as the dot-com bubble, the 2008 financial crisis, periods of zero interest rate levels, and the COVID-19 pandemic. These events might have introduced so much noise into the macroeconomic variables and therefore broken the relationships discovered in other studies.

5.2.1 Ensemble Model

To select the best-performing method for ensembling the two models (Zhou, 2012) recommends simply using the average of the models, when the two constituent models have similar performance. In this study, the RF and NN models perform quite similarly with R-squared values of 8.0% and 7.5%. Additionally, the attempts of using Ridge stacking in Section 4.3.5 and Spearman IC in Section 4.3.6, did not improve the R-squared value of 9.6% from the simple average model. This could be due to the fact that Spearman IC focuses more on correlation with the actual values, which does not mean the R-squared value will be optimized.

In an additional effort to find the best weights for the RF and NN models in this study, we built a Python loop to find the best configuration, resulting in 50/50 being one of the optimal configurations as we show in Appendix E. Therefore, this is the final model on which we build the StockWatch screener.

Table 5 presents the results of the selected ensemble model, showing the average predicted and actual excess returns, and the Sharpe and Sortino ratios per decile, from the out-of-sample test. The deciles are ranked perfectly according to their predicted excess returns, with decile 1 outperforming decile 2, decile 2 outperforming decile 3, and so on. This indicates that the model is effective at ranking stocks based on excess returns, which are already risk-adjusted through the model's construction.

Decile	Predicted excess return	Actual excess return	Sharpe ratio	Sortino ratio
1	28.02%	22.61%	0.94	2.33
2	14.71%	14.49%	0.63	1.53
3	10.15%	9.69%	0.63	1.17
4	7.33%	7.63%	0.63	1.43
5	5.12%	6.21%	0.63	1.20
6	3.22%	5.96%	0.62	1.20
7	1.30%	5.48%	0.68	0.84
8	-0.77%	3.59%	0.55	0.89
9	-3.56%	0.53%	0.43	0.80
10	-11.08%	-7.84%	0.03	0.05

Table 5: Decile performance out-of-sample test

Decile 1 achieves a Sharpe ratio of 0.94, which indicates strong risk-adjusted performance. While a Sharpe ratio above 1 is generally considered very good in literature, a value of 0.94 is acceptable, especially considering the high volatility in the given test period, including the COVID-19 market crash and the Russia-Ukraine war with a period of high inflation (Gu, Kelly, & Xiu, 2020). This is also reflected by the average Sharpe ratio of the entire set of stocks, which is only 0.62, meaning that decile 1 achieves more than 50% higher excess return per unit of volatility, compared to the average stock in the dataset (Sharpe, 1966). Furthermore, the model's performance compares favourably against benchmarks from the literature. For instance, (Gu, Kelly, & Xiu, 2020) report an out-of-sample Sharpe ratio of 0.77 for their ML model, while a simple buy-and-hold investor achieves only 0.51 on their dataset (Gu, Kelly, & Xiu, 2020).

The Sortino ratios show the same patterns as the Sharpe ratio, generally decreasing across deciles. Decile 1 achieves a Sortino ratio of 2.33, which is considered very strong. Sortino ratios above 2 indicate very good risk-adjusted returns (Charles Schwab, 2024), while a Sortino ratio below 1 is deemed unacceptable. These results show the model's strong performance in ranking stocks based on downside-adjusted performance.

Figure 10 shows the compounded excess returns of investing in each decile for the entire test period. This assumes buying the stocks in a given decile each year, with no transaction costs and no short selling. This graph also shows that decile 1 easily outperforms the other deciles, with decile 10 performing the worst with negative excess returns.

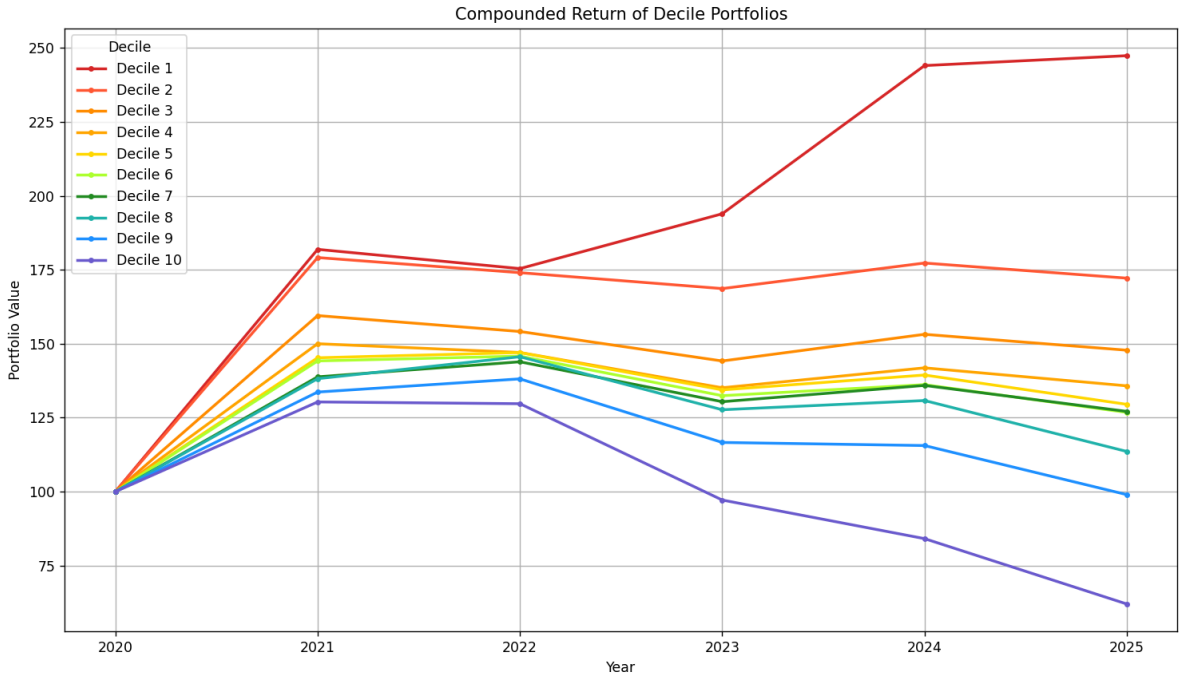

Figure 10: Decile performance graph out-of-sample

We performed these tests on the average of 10 random seeds for the RF and NN models, to ensure robustness. In Appendix B, we included out-of-sample tests in a rolling time window, showing also other time periods. This rolling window test yields similar results as the 80/20 training/test split from Figure 10 and proves the model's test results remain stable under varying conditions.

5.3 Monte Carlo Cross-Validation

In addition to the out-of-sample tests, we also use cross-validation to verify the model's performance. Here we can use the entire 25-year time period, using an 80/20 training/test split on the stocks. This Monte Carlo simulation randomly selects 400 stocks to be part of the training set and reserves the

remaining 100 stocks for the test set. We run this simulation for 50 repetitions, with random seeds for the RF and NN models as well. This test yields an R-squared value of 15.29%, an MAE of 0.219, and an MSE of 0.093.

The decile results of this evaluation in Table 6 are quite similar to the out-of-sample test, demonstrating again that the top deciles outperform the bottom ones on a consistent basis. In this case, the realized Sharpe ratio of the top decile was even higher, at 1.13, indicating very strong risk-adjusted performance. Compared to the Sharpe ratio of the overall dataset of 0.67, this represents a gain of over 68% in excess return per unit of volatility.

The Sortino ratio for decile 1 remains very strong (2.78). In contrast, Sortino ratios for the lower deciles are remarkably lower. The bottom six deciles all report Sortino ratios below 1, which we interpret as having less than 1% excess return per unit of downside deviation.

Decile	Predicted excess return	Actual excess return	Sharpe ratio	Sortino ratio
1	26.32%	21.21%	1.13	2.78
2	15.52%	12.98%	0.87	1.16
3	11.60%	10.62%	0.82	1.18
4	8.94%	8.89%	0.82	0.95
5	6.79%	7.01%	0.76	0.82
6	3.89%	5.23%	0.65	0.65
7	3.07%	3.16%	0.57	0.62
8	0.98%	1.20%	0.46	0.50
9	-1.71%	-2.00%	0.26	0.42
10	-7.66%	-7.54%	-0.02	-0.02

Table 6: Decile performance cross-validation test

Figure 11 shows a graph of decile performances for this Monte Carlo cross validation, it uses a logarithmic scale to clearly see the bottom performing deciles as well.

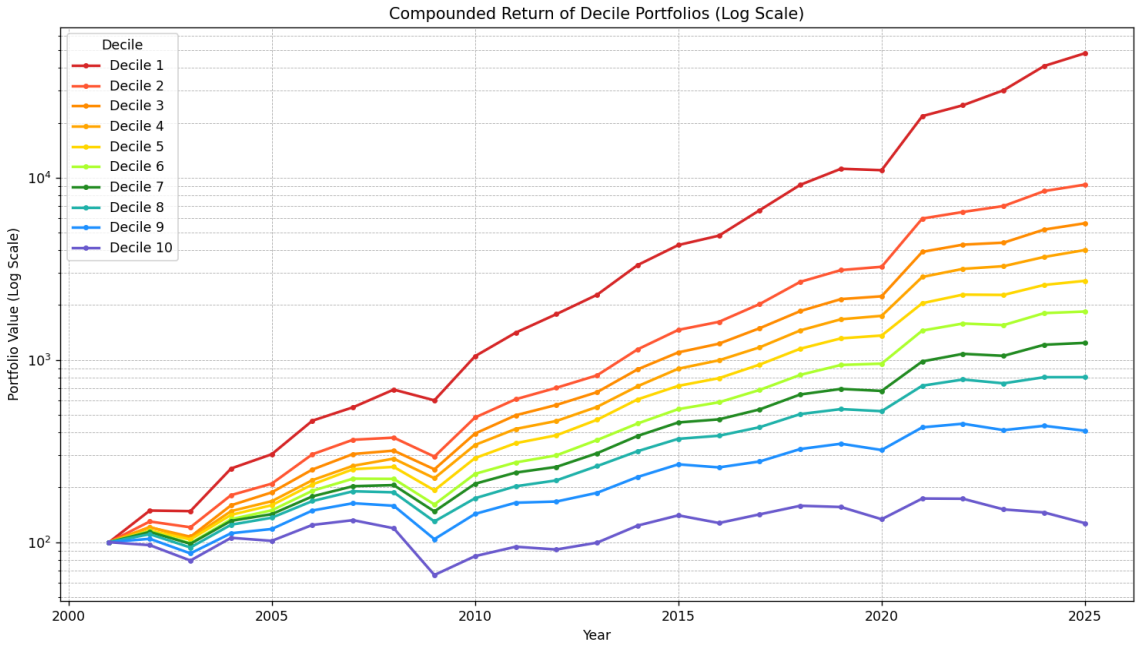

Figure 11: Decile performance graph cross-validation

5.4 Feature Importances

We derive Feature Importances for RF directly from the RandomForestRegressor from scikit-learn (Pedregosa, 2011). These Feature Importances reflect how much each feature contributes to decreasing impurity in the model, or in other words, to reducing the variance in the regression (Pedregosa, 2011). For NN this approach is not possible, therefore we used Permutation Importances instead, which are more computationally costly, but allow to find a proxy for assessing Feature Importances (Pedregosa, 2011). Table 7 shows the average Feature Importances from the RF and the NN model, which were computed to approximate the Feature Importances of the ensemble model.

Characteristic	Average Feature Importance
MarketCap	0.209
TradingVolume	0.202
Momentum1Y	0.098
PriceToBook	0.070
NetProfitMargin	0.043
EVEBITDA	0.040
ROIC	0.032
DividendYield	0.031
EPSCAGR5Y	0.030
ForwardEarningsYield	0.030
CurrentRatio	0.030
Momentum6M	0.029
GrossProfitMargin	0.028
ROA	0.027
Capex	0.023
DebtEBITDA	0.023
ESG	0.022
PayoutRatio	0.021
DPSCAGR5Y	0.013

Table 7: Feature Importances

To gain deeper insight into the direction of each feature, we used SHAP values to interpret the contribution of each individual characteristic to the model.

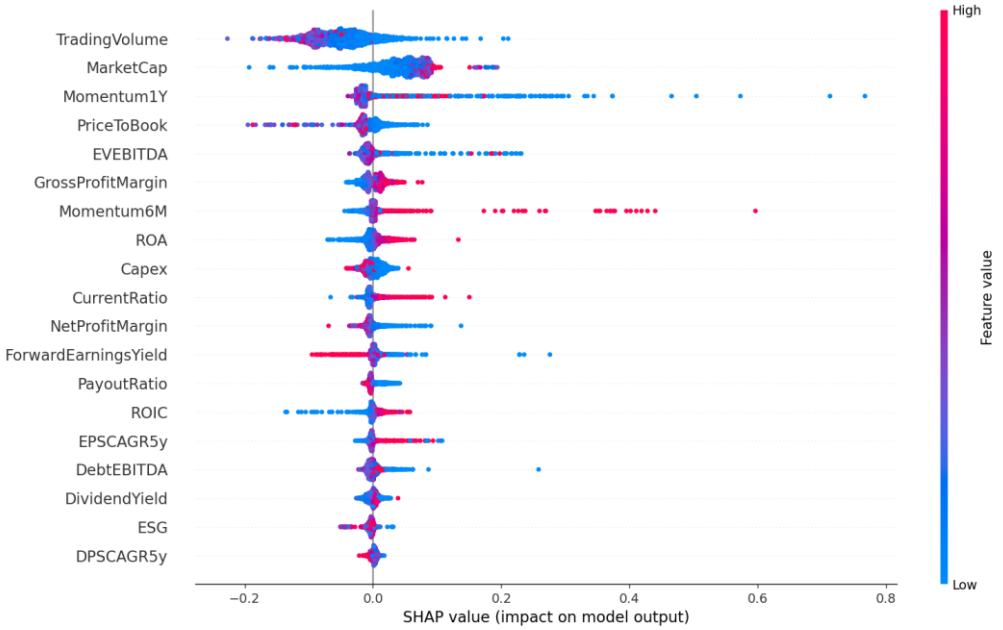

Figure 12: SHAP values RF

In the RF component of the ensemble model, shown in Figure 12, we find several interesting results. The TradingVolume and MarketCap SHAP plots appear to be near mirror images of each other, an expected outcome given their high correlation of 0.75. High values of TradingVolume reduce the model's output predictions while high values of MarketCap increase output predictions. This combined suggests that stocks with a relatively low TradingVolume compared to its MarketCap, tend to outperform, whereas those with unusually high TradingVolume for their MarketCap may underperform. The latter conclusion might be explained by the fact that this group of stocks includes "hype stocks", companies that receive excessive attention on social media or trading forums. This explains their high trading volume, driving prices above fundamental value leading to a bubble in the stock price, and therefore these stocks eventually underperform when the bubble bursts (Brealey, 2022). We draw the same conclusion from the NN model's SHAP plot, depicted in Figure 13, where the inverse relationship between these variables is even more clear. Other interesting, but more expected results, are that low PriceToBook values tend to increase estimates of excess returns, in line with the FF-5 model's HML factor (Fama & French, 2015). And that high Momentum6M values, increase predictions of excess returns, as explained in the FF-6 and (Carhart, 1997) models. Overall, the SHAP values of the RF and NN model show similar patterns, with high (red) and low (blue) feature values for a given characteristic generally appearing on the same side of the SHAP axis in both models, indicating that the feature contributes in the same direction across the ensemble model.

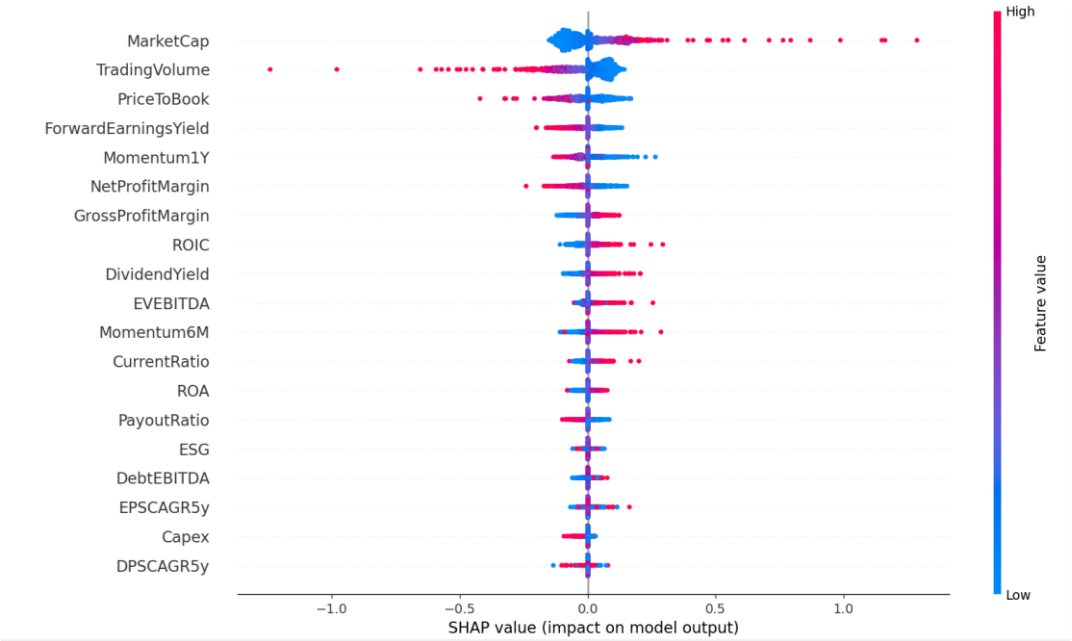

Figure 13: SHAP values NN

5.5 Summary

This chapter evaluates the performance of the models developed in Chapter 4, as outlined in the evaluation procedure in Section 5.1. The findings show that the ensemble model, which equally weights the RF and the NN model, achieves the strongest predictive performance on the selected evaluation metrics.

On the out-of-sample validation presented in Section 5.2, the ensemble model achieves an R-squared value of 9.6%, with an MAE of 0.260 and an MSE of 0.135, outperforming the other configurations. When sorting the stocks into decile portfolios, a clear ranking pattern emerges, confirming the model's ability to rank stocks effectively. Decile 1 achieves a Sharpe ratio of 0.94, significantly outperforming the overall Sharpe ratio of 0.62 of the entire dataset and delivers 22.61% excess returns per year. We confirm the robustness of the model through a Monte Carlo cross-validation across the entire time

period. This simulation yields an even higher R-squared value of 15.29% with an MAE of 0.219 and an MSE of 0.093. Once again, the decile analysis confirms the ranking ability of the model with Decile 1 achieving a Sharpe ratio of 1.13, compared to 0.67 for the entire dataset.

The OLS model developed as a benchmark for the FF-6 model, scored much lower, with an R-squared value of 0.73%, showing the developed models in this thesis using ML, easily outperform traditional models. Furthermore, decile tests confirm that the developed model can pick outperforming stocks on a cross-validation as well as out-of-sample basis. These results justify the implementation of this model for StockWatch, supplemented with a screening tool to help them reduce time and effort on manual stock analysis, while improving the quality of the selected stocks. We discuss this implementation in Chapter 6.

6 Model Implementation

6.1 Screener Model

To make the model more practical to use for StockWatch, we develop a stock ranking list based on the validated ensemble model from Section 4.3.4. Since this model showed strong performance across all validation tests, it is now selected to give recommendations and streamline StockWatch's stock selection process by identifying interesting stocks. We achieve this by including a ranking list which orders all stocks based on their predicted excess return for the upcoming year.

To handle missing data in this ranking list, we implement a fallback mechanism. This works as follows: if a data point is missing for the current year, the model uses (if available) the data point from the previous year from the same stock. If the data point from this previous year is also missing, it counts as a missing value, and the value is imputed similarly to the method proposed in Section 4.1.3. This approach ensures the use of recent data, while preserving enough data points for an accurate prediction. We made an exception for the momentum variables for which this mechanism would not work due to the time-sensitive nature of these values. Therefore, we manually verified momentum values to be complete for each stock across all years.

For this ranking tool, we extend the model beyond the S&P 500 index, to include the AEX index and the EURO STOXX 50 index (EU50). The AEX index represents the largest 25 companies listed on the Dutch stock exchange, while the EU50 tracks the largest 50 stocks from the Eurozone (Stoxx, 2025). We include these additional indices since they are interesting for StockWatch's subscriber base, who closely follow the Dutch and European markets as well. For the EU50 and AEX index, the missing value threshold is set to 5, instead of the 3 used in the model development for the S&P 500. This adjustment ensures that a sufficient number of stocks from these indices are kept in the ranking, since data for these indices was limited compared to the S&P 500. Since we do not use the EU50 and AEX index during the training of the model, this relaxation does not introduce any biases into the model.

To improve usability for StockWatch, we integrate some additional screening functionalities into the model and ranking list. These filters allow users of the model to screen the ranking list on specific criteria, such as selecting only stocks from the AEX index, or only including companies with a PE ratio lower than 30. These filters make it easier to navigate through the full ranking list and help streamline the stock selection process. By being able to narrow the search to focus on relevant or interesting stocks, the model becomes even more practical to use and more time-efficient for StockWatch. We provide an overview of the possible filters in Appendix F.

6.2 Implementation

To ensure easy implementation for StockWatch, we upload the final screening model to a newly created Google Drive, to run the model in the cloud without the need to install complex coding programmes. No further steps must be taken to implement the model, and the model can be used anytime in the stock selection process. For example, to support the identification of potential new stocks, or to assess the expected excess return of an interesting stock. The Python model is easy to run and does not require any coding skills to use in practice. The screening filters are given in the Command Line Input (CLI), and questions are asked which require a one-word answer to enable easy screening. It is also possible to skip the screener, by simply clicking enter.

To maintain the most up-to-date data, it is important to update the Excel sheets containing the LSEG data periodically. For most variables, updating them quarterly is sufficient since companies also share their financial statements on a quarterly basis. However, for the momentum variables, it is advised to adopt monthly updates, especially in volatile times, since momentum can change rapidly if the stock price of a company has risen or fallen fast over the last month.

Although this model has been validated and achieved robust results on all tests, there are risks associated with using this model for individual stock selection. The model is not a perfect tool, it may still select stocks that underperform or give inaccurate predictions for certain stocks. While the average performance of the model is strong, we do not advise to make individual investment decisions solely on the model's output. We discuss some more limitations in Section 7.3. Furthermore, it is important to keep monitoring the model's performance (for example annually), by using the same evaluation metrics used in Chapter 5. Even though the model has performed well in all time periods of the rolling window tests, there is always a risk of the model's predictive ability decreasing, for example during unusual economic events or market disruptions.

Once the screener is implemented in StockWatch, an interesting opportunity for StockWatch to pursue with this screener, is to develop a Big Data or ML portfolio based on this model. We provide further details on this recommendation in Section 7.2.

6.3 Validation

While we validated the ensemble model and its performance already in Chapter 5, we provide additional validation of the screener model by an expert at StockWatch (Koerts, 2025) in this section. This provides practical insights and feedback into the usability of the screener model, and the potential of this model for the launch of a StockWatch Big Data portfolio.

According to the expert, the screening functionalities are a useful addition to the ranking list, increasing time-efficiency by not having to go through the entire list of stocks (Koerts, 2025). The model's results seem logical and in line with financial knowledge, for example the Momentum6M variable shows a positive relationship with expected excess returns as discovered in (Carhart, 1997), and stocks with a lower PriceToBook value tend to have higher expected excess returns as found in (Fama & French, 1992). (Koerts, 2025) highlights the relationship between TradingVolume and MarketCap discovered in Section 5.4 as very interesting and worth of further investigation. A suggestion for future improvement is to add a ratio of TradingVolume as percentage of MarketCap to the model (Koerts, 2025).

A drawback of the model for its use in a future Big Data portfolio is that the results are difficult to interpret, since the ML algorithms have a complex and non-linear structure. This makes it a challenging task to trace back individual characteristics and their influence on the model's output. This makes it hard to explain to StockWatch's subscribers why certain stocks were selected over others (Koerts, 2025). To address this issue, we propose some solutions in the recommendations in Section 7.2. Overall, the expert confirms that this model is a valuable tool for StockWatch and will definitely be used going forward. The Big Data portfolio is something StockWatch already wanted to implement, and this model provides a strong foundation for this portfolio to be launched in the near future (Koerts, 2025).

7 Conclusion and Recommendations

7.1 Conclusion

This research began with the central research question: *How can a data-driven model be developed to support StockWatch in systematically identifying outperforming stocks?* With the goal of creating a model for StockWatch that could improve the efficiency and quality of their stock selection process.

In the introduction of this study, we outlined the current stock selection process at StockWatch. We found that no systematic procedure was used, and that the large set of potential stocks to investigate made it difficult for StockWatch to find outperforming stocks. A data-driven model which utilizes the extensive LSEG database that StockWatch has access to, was needed to improve this process and increase the performance of StockWatch's portfolios. The literature review concluded that CAPM was suitable as a basis for the excess return formula, using β as risk indicator, aligning with how StockWatch conducts their analyses. Furthermore, recent literature supported the use of ML algorithms to enhance these simpler asset pricing models.

This study began with the FF-6 model as the basis for a first set of characteristics. Since ML can handle more characteristics, this initial set was extended to include a total of 19 stock-specific and 5 macroeconomic characteristics, based on prior studies. These characteristics together cover a complete financial overview of a company, using forward as well as backward-looking data, including all financial dimensions such as profitability, growth, valuation, leverage, and momentum.

To develop a model with these characteristics, we identified RF and NN as suitable ML algorithms to complement and enhance the OLS model. After developing all individual models and creating ensemble variants, we found that the ensemble model, using a 50/50 weight for RF/NN, excluding the macroeconomic characteristics, was the best performing model for StockWatch.

We concluded in Section 5.1 that the main metrics to use for evaluating the models were the R-squared, MSE and MAE values. On top of that, to assess ranking ability, we constructed decile portfolios, using an out-of-sample as well as a cross-validation approach, with an 80/20 training/test split. From these decile portfolios, we calculated the excess returns of each decile portfolio, and the Sharpe and Sortino ratios of each decile, and assessed these to evaluate the risk-adjusted performance of our model.

All in all, the final ensemble model achieved strong performance, achieving a 9.6% R-squared on the out-of-sample test and a Decile 1 Sharpe ratio of 0.94, delivering 22.61% excess returns per year. On the cross-validation test the model achieved a 15.3% R-squared value, and a Decile 1 Sharpe ratio of 1.13, producing 68% higher returns per unit of volatility than the average of the entire dataset. A clear decile ranking emerged in both validation tests, showing the predictive ability and robustness of the model, which justifies the implementation of this model for StockWatch.

7.2 Recommendations

We recommend integrating the model developed in this study into StockWatch's stock selection process. It can help find potential investments for the portfolios on their platform. The model's filtering functionality allows users to narrow down the search space of potential stocks and make the stock selection process more efficient. Furthermore, the model has proven to be able to select outperforming stocks, so the performance of the portfolios on StockWatch's platform can be increased as well using this model.

When using this model, StockWatch must consider several crucial factors. As described in the implementation section, it is important to regularly update the input data to maintain accurate predictions. Additionally, the company should treat the model as a supporting tool, not as a replacement

for doing thorough research. Investment decisions should ultimately be decided on stock analysis done by experts.

Another recommendation for StockWatch is to launch a Big Data portfolio, for which this model would be very suitable as a starting point. This is something that provides additional value for paid subscribers and is a valuable addition to the portfolios already on StockWatch's platform. Since subscribers find it important to see why and how certain decisions were made, using our model for the Big Data portfolio would require some tweaking. A possibility is to remove some characteristics from the model, making explanations easier. Additionally, StockWatch could use SHAP and Feature Importances to explain to subscribers how the model works, instead of focusing on explaining every individual stock the model has selected. Lastly it is possible to simplify the model, for example only using one of the two ML algorithms instead of the ensemble model, to make it easier to give transparent explanations to subscribers. Before launching this ML-supported Big Data portfolio to subscribers, we recommend reviewing the model under the EU AI Act and related financial regulations, to ensure compliance with regulatory requirements regarding transparency, explainability, and risk management.

7.2.1 Top 20 Stock Recommendations

After reading this thesis, many readers are likely curious about the top stock recommendations from the model for 2025. Table 8 shows the top 20 stock recommendations generated by the final model for 2025. The last column shows the predicted excess return, which is based on the ensemble model, using all data available on or before April 18, 2025. Naturally, it is important to note that this tool is not a perfect predictor of stock prices, and that these results should not be interpreted as financial advice.

Stock	**Index**	**Predicted Excess Return 2025 (%)**
COSTAR GP.	SP500	81.0
MERCEDES-BENZ GROUP	EU50	52.1
LYONDELLBASELL INDS.CL.A	SP500	48.3
DOW ORD SHS	SP500	46.9
BMW	EU50	45.3
APA	SP500	44.4
MICROCHIP TECH.	SP500	44.1
TOTALENERGIES	EU50	43.3
KERING	EU50	43.1
ENI	EU50	34.2
PERNOD-RICARD	EU50	31.4
HOLOGIC	SP500	30.4
TESLA	SP500	30.4
VALERO ENERGY	SP500	30.1
HALLIBURTON	SP500	30.0
LVMH	EU50	29.6

STANLEY BLACK & DECKER	SP500	28.7
BALL	SP500	27.0
INTUITIVE SURGICAL	SP500	26.0
NUCOR	SP500	25.6

Table 8: Top 20 stocks for 2025

7.3 Limitations and Future Work

There are some limitations in this study which may affect the robustness of the model, or present opportunities for further research. First of all, the sample size could be enlarged, due to limitations in our dataset, this study includes only U.S. stocks from the S&P 500 from 2000-2025, as data before the year 2000 was incomplete. A larger dataset, including international stocks and data from before 2000, would have improved the model's robustness, and allow for model training during different market conditions. Another addition would be to add non quantitative data as well, such as whether a company is founder-led or not, or other characteristics, for example about their management.

Another limitation is the potential for survivorship bias, since the dataset includes stocks that are currently in the SP500, meaning stocks which have gone bankrupt during our sample period have been removed from the dataset. This does not compromise the decile ranking or the R-squared values but does help explain why even the lowest ranked deciles showed neutral or even positive excess returns, also partly due to an extremely strong stock market period in the included years. As a result, the MRP value of 6% might have been too low for this time period. However, since our model is developed to predict future performance, this value was kept stable at 6%. To mitigate survivorship bias in further research, using a fixed S&P 500 composition from 2000 is advised, which was unfortunately not available in the given LSEG dataset.

Lastly, a suggestion for future work would be to improve the ensembling method. In this study a 50/50 weight between RF and NN was found to be optimal, but more advanced techniques of combining models like boosting or developing a nested model (in which one model's output serves as input to another model), would have possibly improved the model's predictive ability.

Bibliography

Alan Gregory, R. T. (2013). Constructing and Testing Alternative Versions of the Fama-French and Carhart Models in the UK. *Journal of business finance and accounting*. doi:10.1111/jbfa.12006

Alexis Akira Toda, K. J. (2017). Fat tails and spurious estimation of consumption-based asset pricing models. *Journal of applied econometrics*. doi:10.1002/jae.2564

Anisha Ghosh, O. L. (2023). Estimation with mixed data frequencies: A bias-correction approach. *Journal of empirical finance*. doi:10.1016/j.jempfin.2023.07.005

Aradi, A. A., & Jaimungal, S. (2021). Active and passive portfolio management with latent factors. *Quantitative finance*, 1437-1459. doi:10.1080/14697688.2021.1881598

Ashby, M. W., & Linton, O. B. (2024). Do Consumption-Based Asset Pricing Models Explain the Dynamics of Stock Market Returns? *Journal of Risk and Financial Management*. doi:10.3390/jrfm17020071

Avramov, D., Cheng, S., Metzker, L., & Voigt, S. (2023). Integrating Factor Models. *The journal of finance*, 1593-1646. doi:10.1111/jofi.13226

Ball, R., Sadka, G., & Sadka, R. (2009). Aggregate Earnings and Asset Prices. *Journal of accounting research*, 1097-1133. doi:10.1111/j.1475-679X.2009.00351.x

Barillas, F., & Shanken, J. (2018). Comparing Asset Pricing Models. *The journal of finance*, 715-754. doi:10.1111/jofi.12607

Bergeron, C. (2022). Benchmark, relative return, and asset pricing. *Taylor & Francis Journals*, 1498-1503. doi:10.1080/13504851.2021.1940080

Bhanja, S., & Das, A. (2019). Impact of Data Normalization on Deep Neural Network for Time Series Forecasting. doi:10.48550/arXiv.1812.05519

Biau, G., & Scornet, E. (2015). A Random Forest Guided Tour. doi:10.1007/s11749-016-0481-7

Brealey, R. A. (2022). *Principles of corporate finance.* Boston: McGraw-Hill.

Breeden, D. T., Gibbons, M. R., & Litzenberger, R. H. (1989). Empirical Tests of the Consumption-Oriented CAPM. *Empirical Tests of the Consumption-Oriented CAPM*, 231-262. doi:10.1111/j.1540-6261.1989.tb05056.x

Burnside, C. (2015). Identification and Inference in Linear Stochastic Discount Factor Models with Excess Returns. *Journal of financial econometrics*, 295-330. doi:10.1093/jjfinec/nbv018

Carceles-Poveda, E., & Giannitsarou, C. (2008). Asset pricing with adaptive learning. *Review of Economic Dynamics*, 629-651. doi:10.1016/j.red.2007.10.003

Carhart, M. M. (1997). On Persistence in Mutual Fund Performance. *The Journal of Finance*, 57-82. doi:10.1111/j.1540-6261.1997.tb03808.x

Cecchetti, S. G. (2000). Asset Pricing with Distorted Beliefs: Are Equity Returns Too Good to Be True. *The American Economic Review*, 787–805. doi:10.1257/aer.90.2.787

Chandra, M. V. (2021). Weather Effects on Stock Market Returns in the United States. *Honors Theses and Capstones*, 585.

Charles Schwab. (2024, June 27). *Using the Sortino Ratio to Gauge Downside Risk*. Retrieved June 27, 2025, from Charles Schwab: https://www.schwab.com/learn/story/using-sortino-ratio-to-gauge-downside-risk

Cochrane, J. (2001). Asset pricing. *Princeton University Press*.

Cooper, D. R., & Schindler, P. S. (2013). What is good research? In D. R. Cooper, & P. S. Schindler, *Business Research Methods* (pp. 15-18). McGraw-Hill.

Cruz-Martínez, & Rafael, R. (2022, September 8). *Practice guide: Setting up your systematic search strategy.* doi:10.5281/zenodo.7062726

Dall-E. (2025, May 30). *Dall-E 3*. Retrieved from OpenAI: https://openai.com/index/dall-e-3/

Erdős, P., Ormos, M., & Zibriczky, D. (2011). Non-parametric and semi-parametric asset pricing. *Economic modelling*, 1150-1162. doi:10.1016/j.econmod.2010.12.008

Fabian Hollstein, M. P. (2020). The Conditional Capital Asset Pricing Model Revisited: Evidencefrom High-Frequency Betas. *Management science*, 2474-2494. doi:10.1287/mnsc.2019.3317

Fabozzi, F. J., Huang, D., Jiang, F., & Wang, J. (2024). What difference do new factor models make in portfolio allocation? *Journal of International Money and Finance*. doi:10.1016/j.jimonfin.2023.102997

Fama, E. (1970). Efficient Capital Markets: A Review of Theory and Empirical Work. *The Journal of Finance*, 383-417. doi:10.1111/j.1540-6261.1970.tb00518.x

Fama, E. F., & French, K. R. (1992). The Cross-Section of Expected Stock Returns. *The journal of finance*, 427-465. doi:10.1111/j.1540-6261.1992.tb04398.x

Fama, E. F., & French, K. R. (2015). A five-factor asset pricing model. *Journal of Financial Economics*, 1-22. doi:10.1016/j.jfineco.2014.10.010

Fama, E. F., & French, K. R. (2018). Choosing factors. *Journal of Financial Economics*, 234-252. doi:10.1016/j.jfineco.2018.02.012

Ferreira, E., Gil-Bazo, J., & Orbe, S. (2011). Conditional beta pricing models: A nonparametric approach. *Journal of banking & finance*, 3362-3382. doi:10.1016/j.jbankfin.2011.05.016

Fletcher, J. (2018). Bayesian tests of global factor models. *Journal of empirical finance*, 279-289. doi:10.1016/j.jempfin.2018.01.006

Géron, A. (2017). *Hands-On Machine Learning with Scikit-Learn & TensorFlow* . Sebastopol: O'Reilly Media.

Ghanoum, T. (2021). Why multicollinearity isn't an issue in Machine Learning. *TDS*.

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *The review of financial studies*, 2223-2273. doi:10.1093/rfs/hhaa009

Guvenen, F. (2009). A parsimonious macroeconomic model for asset pricing. *Econometrica*. doi:10.3982/ECTA6658

Healy, J. V., Gregoriou, A., & Hudson, R. (2024). Automated Machine Learning and Asset Pricing. *Risks*. doi:10.3390/risks12090148

Heerkens, H. (2017). *Solving Management Problems Systematically.* Noordhof. doi:10.4324/9781003186038

Hollstein, F., & Prokopczuk, M. (2022). Testing Factor Models in the Cross-Section. *Journal of banking & finance*. doi:10.1016/j.jbankfin.2022.106626

Hvitfeldt, E. (2024). Remove Missing Values. In E. Hvitfeldt, *Feature Engineering A-Z.* Retrieved May 29, 2025

James C. Van Horne, J. M. (2008). *Fundamentals of Financial Management.* Pearson.

John Y. Campbell, J. M. (1993). WHERE DO BETAS COME FROM - ASSET PRICE DYNAMICS AND THE SOURCES OF SYSTEMATIC-RISK. *The Review of Financial Studies*, 567-592. doi:10.1093/rfs/6.3.567

Juan-Pedro Gómez, F. Z. (2003). Asset pricing implications of benchmarking: A two-factor CAPM. *European Journal of Finance*, 343 - 357.

Kaczmaczyk, K., & Hernes, M. (2020). Financial decisions support using the supervised learning method based on random forests. *Procedia Computer Science*, 2802-2811.

Keith Cuthbertson, D. N. (2022). Mutual fund performance persistence: Factor models and portfolio size. *International Review of Financial Analysis*. doi:10.1016/j.irfa.2022.102133

Khamis, A., Ismail, Z., Haron, K., & Mohammed, A. T. (2005). The Effects of Outliers Data on Neural Network Performance. *Journal of Applied Sciences*, 1394-1398. doi:10.3923/jas.2005.1394.1398

Koerts, N. (2025, March). Current situation stock selection process. (R. d. Mink, Interviewer)

Koerts, N. (2025, May 14). Outlier removal. (R. d. Mink, Interviewer)

Koerts, N. (2025, May 28). Validation Screener model. (R. d. Mink, Interviewer)

Kolari, J. W., Liu, W., & Huang, J. Z. (2021). *A new model of capital asset prices.* Milan: Palgrave Macmillan. doi:10.1007/978-3-030-65197-8

Kozak, S., Nagel, S., & Santosh, S. (2018). Interpreting Factor Models. *The Journal of Finance*, 1183-1223. doi:10.1111/jofi.12612

Leippold, M., Wang, Q., & Zhou, W. (2022). Machine learning in the Chinese stock market. *Journal of Finance and Economics*, 64-82. doi:10.1016/j.jfineco.2021.08.017

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 4768-4777. doi:10.5555/3295222.3295230

Martin Lettau, M. P. (2020). Estimating latent asset-pricing factors. *Journal of Econometrics*, 1-31. doi:10.2139/ssrn.3175556

Meiyun Wang, K. I. (2024). LLMFactor: Extracting Profitable Factors through Prompts for Explainable Stock Movement Prediction. doi:10.18653/V1/2024.FINDINGS-ACL.185

Nagel, S., & Xu, Z. (2022). Asset pricing with Fading Memory. *The review of financial studies*. doi:10.1093/rfs/hhab086

Nard, G. D., Hediger, S., & Leippold, M. (2022). Subsampled factor models for asset pricing: The rise of Vasa. *Journal of forecasting*.

Novy-Marx, R. (2013). The other side of value: The gross profitability premium. *Journal of Financial Economics*, 1-28. doi:10.1016/j.jfineco.2013.01.003

OpenAI. (2025). *ChatGPT version 4o*. Retrieved from ChatGPT: https://chatgpt.com/

Oxford English Dictionary. (n.d.). Oxford: Oxford University Press. Retrieved April 3, 2025

Pedregosa, F. V. (2011). Machine learning in Python. *Journal of Machine Learning Research*, 2825-2830. doi:10.5555/1953048.2078195

Qian, H. (2013). A flexible state space model and its applications. *Journal of time series analysis*, 79-88. doi:10.1111/jtsa.12051

Qian, Y., & Zhang, Y. (2025). Long-term forecasting in asset pricing: Machine learning models' sensitivity to macroeconomic shifts and firm-specific factors. *The North American Journal of Economics and Finance*. doi:10.1016/j.najef.2025.102423

Raymond Kan, C. R. (2009). Model Comparison Using the Hansen-Jagannathan Distance. *The review of financial studies*, 3449-3490. doi:10.1093/rfs/hhn094

Red Rock Capital. (2010). *Sortino: A 'Sharper' Ratio.* Retrieved June 27, 2025, from https://www.cmegroup.com/education/files/rr-sortino-a-sharper-ratio.pdf

Ritholtz Wealth Management. (2024). *Ritholtz Wealth Management*. Retrieved May 23, 2025

Robert Novy-Marx, M. V. (2022). Betting against betting against beta. *Journal of financial economics*, 80-106. doi:10.1016/j.jfineco.2021.05.023

Rocciolo, F., Gheno, A., & Brooks, C. (2022). Explaining abnormal returns in stock markets: An alpha-neutral version of the CAPM. *International review of financial analysis*. doi:10.1016/j.irfa.2022.102143

Ross, S. A. (1976). The arbitrage theory of capital asset pricing. *Journal of Economic Theory*, 341-360. doi:10.1016/0022-0531(76)90046-6

S&P Dow Jones Indices S&P500. (2025, May 16). Retrieved May 16, 2025, from S&P Dow Jones Indices: https://www.spglobal.com/spdji/en/indices/equity/sp-500/#overview

Saunders, M. N. (2019). *Research methods for business students* (8th ed.). Pearson.

Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *Journal of Finance*, 425-442. doi:10.1111/j.1540-6261.1964.tb02865.x

Sharpe, W. F. (1966). Mutual Fund Performance. *Journal of Business*. doi:10.1086/294846

Spearman, C. (1904). The proof and measurement of association between two things. *The American Journal of Psychology*, 72-101. doi:10.2307/1412159

Stoxx. (2025). *EURO STOXX 50*. Retrieved June 5, 2025, from Stoxx: https://stoxx.com/index/sx5e/

Tetlock, P. C. (2007). Giving Content to Investor Sentiment: The Role of Media in the Stock Market. *The Journal of Finance*, 1139-1168. doi:10.1111/j.1540-6261.2007.01232.x

Umlandt, D. (2023). Score-driven asset pricing: Predicting time-varying risk premia based on cross-sectional model performance. *Journal of econometrics*. doi:10.1016/j.jeconom.2023.05.007

Visengeriyeva, L. (2025). *CRISP-ML(Q). The ML Lifecycle Process.* Retrieved May 8, 2025, from MLOps: https://ml-ops.org/content/crisp-ml

Wang, C. (2024). Stock return prediction with multiple measures using neural network models. *Financial innovation*. doi:10.1186/s40854-023-00608-w

Wayne E Ferson, C. R. (1997). Fundamental determinants of national equity market returns: A perspective on conditional asset pricing. *Journal of banking & finance*, 1625-1665. doi:10.1016/S0378-4266(97)00007-4

Wooditch, A., Johnson, N. J., Solymosi, R., Ariza, J. M., & Langton, S. (2021). Ordinary Least Squares Regression. In A. Wooditch, N. J. Johnson, R. Solymosi, J. M. Ariza, & S. Langton, *A Beginner's Guide to Statistics for Criminology and Criminal Justice Using R* (pp. 245-268). Springer.

Ying, Q. Y., T., A., Q. u., A., & Y., &. R. (2019). Stock Investment and Excess Returns: A Critical Review in the Light of the Efficient Market Hypothesis. *Journal of Risk and Financial Management*. doi:10.3390/jrfm12020097

Zhou, Z.-H. (2012). *Ensemble Methods (Machine Learning and Pattern Recognition Series)* (1 ed.). London: Chapman & Hall.

Declaration of AI Usage

During the preparation of this work, the author used Artificial Intelligence (AI) in order to develop the image for the cover page (Dall-E, 2025). Additionally, AI was used for debugging and generating fixes for parts of the Python code, helping with the development of the ML algorithms. For this, ChatGPT model 4o was used (OpenAI, 2025). This model was also used for proofreading this research.

I hereby declare that all final content is critically reviewed, verified, and edited by the author to ensure an accurate and valid thesis.

Appendix A

OLS Model FF-6

Summary statistics out-of-sample test:

Model F-statistic: 17.09, P-value: $1.09 * 10^{-19}$, R-squared: 0.73%, MSE: 0.134, MAE: 0.270

Variable	Coefficient	Std. Error	t-Statistic	P-Value	95% CI Lower	95% CI Upper
Constant	0.0235	0.011				
ROIC	0.0029	0.001	5.856	0.000	0.002	0.004
GrossProfitMargin	0.0007	0.000	3.714	0.000	0.000	0.001
MarketCap*[14]	-5.425e-11	5.79e-11	-0.937	0.349	-1.68e-10	5.92e-11
PriceToBook	-0.0045	0.001	-4.240	0.000	-0.007	-0.002
Momentum1Y	-0.1101	0.015	-7.309	0.000	-0.140	-0.081
Momentum6M	0.0874	0.025	3.486	0.000	0.038	0.137

Table 9: OLS FF-6 output

OLS Model all Characteristics

Summary statistics out-of-sample test:

Model F-statistic: 13.77, P-value: $5.62 * 10^{-43}$, R-squared: 3.15%, MSE: 0.183, MAE: 0.269

Variable	Coefficient	Std. Error	t-Statistic	P-Value	95% CI Lower	95% CI Upper
Constant	0.2100	0.029				
Momentum6M*	-0.0539	0.038	-1.435	0.151	-0.128	0.020
ROIC*	-5.862e-5	0.001	-0.075	0.940	-0.002	0.001
GrossProfitMargin*	0.0003	0.000	1.108	0.268	-0.000	0.001
DividendYield	0.0375	0.007	5.343	0.000	0.024	0.051
MarketCap	3.734e-10	9.49e-11	3.935	0.000	1.87e-10	5.59e-10
ForwardEarningsYield	-0.0129	0.002	-5.893	0.000	-0.017	-0.009
PriceToBook*	4.501e-5	0.000	0.435	0.664	-0.000	0.000
Momentum1Y	-0.1616	0.022	-7.246	0.000	-0.205	-0.118
DPSCAGR5y	-0.0016	0.000	-4.799	0.000	-0.002	-0.001
EPSCAGR5y	0.0012	0.000	3.961	0.000	0.001	0.002
ESG	-0.0007	0.000	-2.579	0.010	-0.001	-0.000

[14] Characteristics indicated with a * are not significant at a 5% significance level.

EVEBITDA	-0.0021	0.001	-2.259	0.024	-0.004	-0.000
DebtEBITDA	0.0059	0.003	2.019	0.044	0.000	0.012
ROA*	0.0022	0.002	1.270	0.204	-0.001	0.005
NetProfitMargin	-0.0015	0.001	-2.045	0.041	-0.003	-0.000
PayoutRatio	-0.0021	0.000	-5.260	0.000	-0.003	-0.001
Capex*	-0.0008	0.001	-1.663	0.096	-0.002	0.000
TradingVolume*	-6.4e-09	3.49e-09	-1.834	0.067	-1.32e-08	4.43e-10
CurrentRatio	0.0150	0.005	3.198	0.001	0.006	0.024

Table 10: OLS all characteristics output

F-test on OLS Models

H_0: The 13 additional variables do not significantly improve the OLS-FF6 model

H_1: The 13 additional variables do significantly improve the OLS-FF6 model

The F-statistic is calculated according to Equation 20:

$$F = \frac{(R^2_{unrestricted} - R^2_{restricted})/(k_{unrestricted} - k_{restricted})}{(1 - R^2_{unrestricted})/(N - k_{unrestricted})}$$

Equation 20: F-statistic

Where:

$R^2_{unrestricted}$ = model with 19 characteristics = 3.15%

$R^2_{restricted}$ = model with 6 characteristics = 0.73%

$k_{unrestricted}$ = 19 characteristics + 1 = 20

$k_{restricted}$ = 6 characteristics + 1 = 7

N = number of observations in test set = 1893

$$F = \frac{(0.0315 - 0.0073)/(20 - 7)}{(1 - 0.0315)/(1893 - 20)}$$

Equation 21: F-statistic

$$F = 3.60$$

Critical value for df_1 = 13 and df_2 = 1873 is 1.73. F-statistic from Equation 21 is 3.60.

Since F = 3.60 > 1.73, we reject the null hypothesis at a 5% significance level, and we conclude that the 13 additional characteristics contribute significantly to the model's predictive power.

Appendix B

Rolling Window Test Out-of-Sample

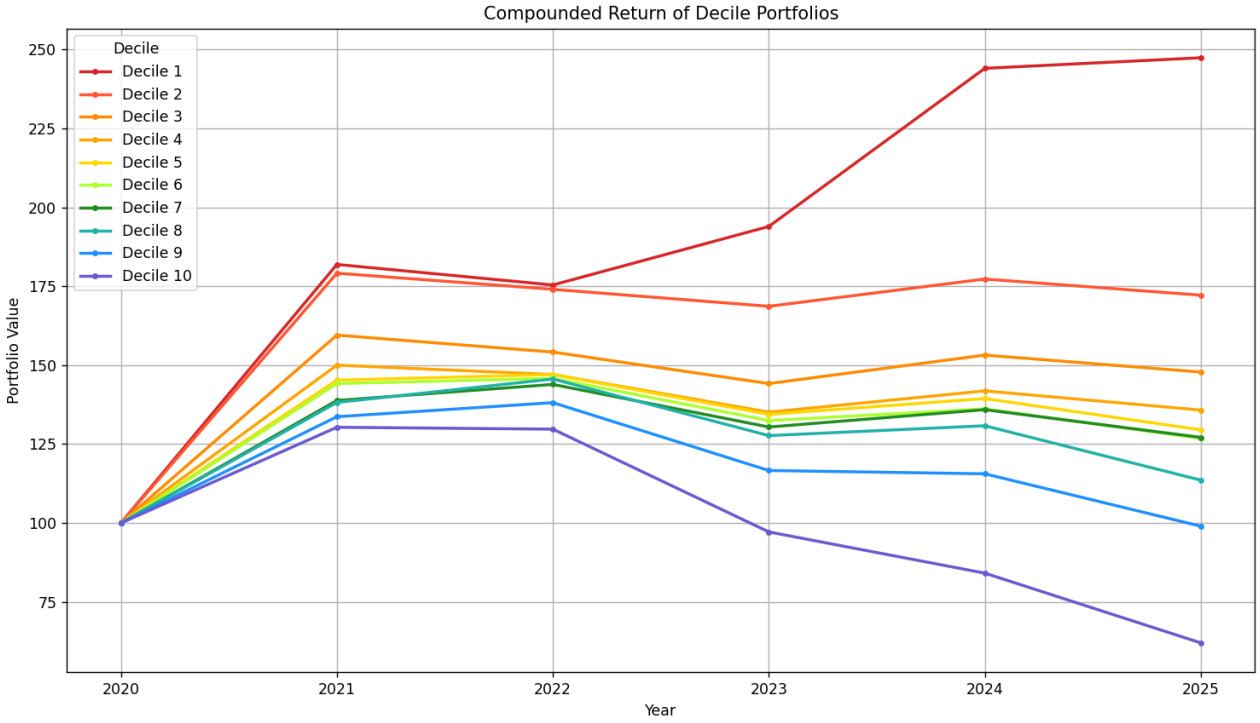

Figure 14: Compounded return of decile portfolios 2020-2025

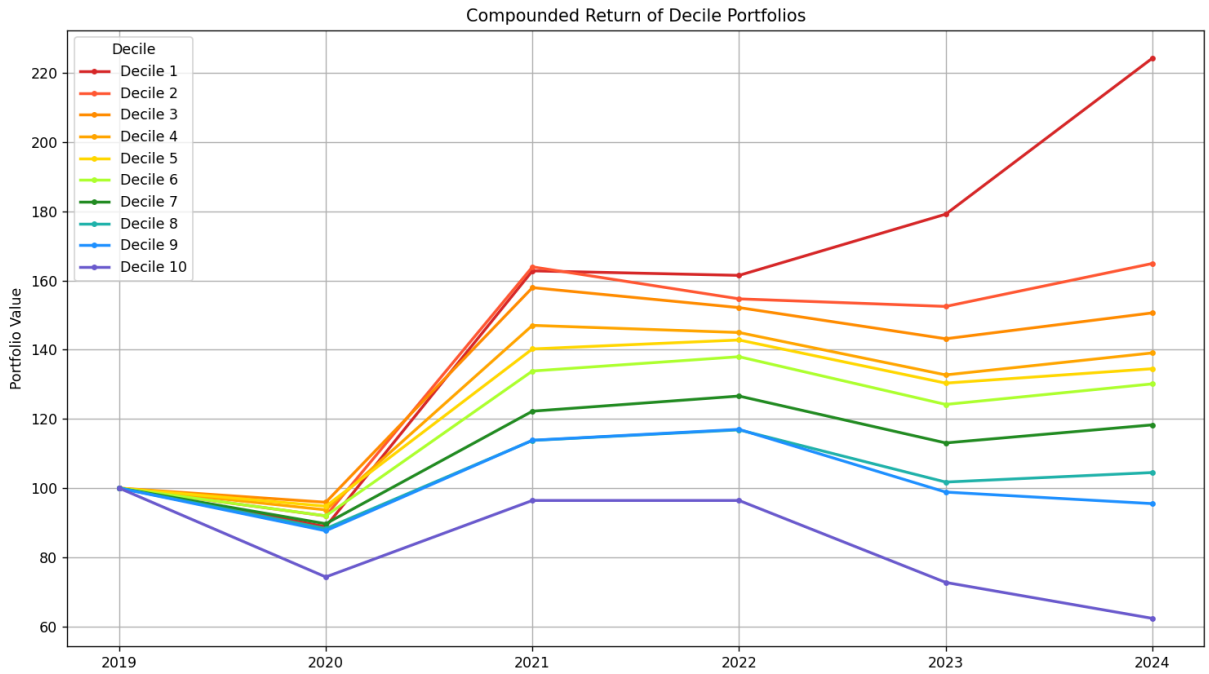

Figure 15: Compounded return of decile portfolios 2019-2024

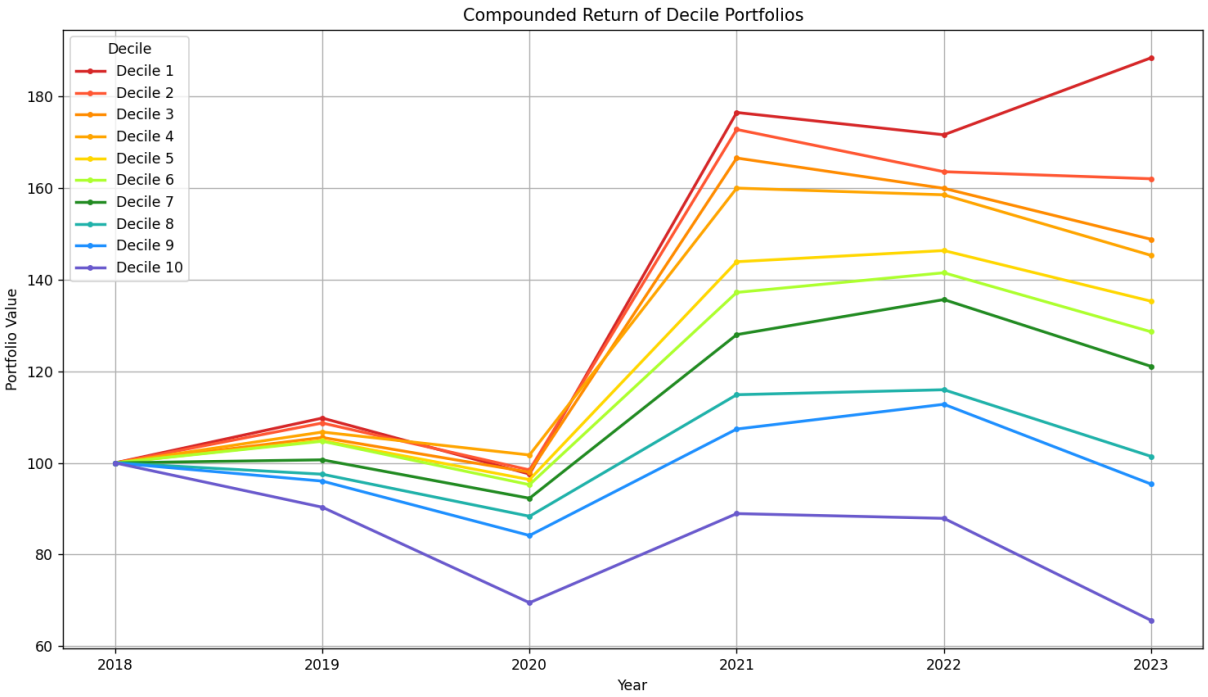

Figure 16: Compounded return of decile portfolios 2018-2023

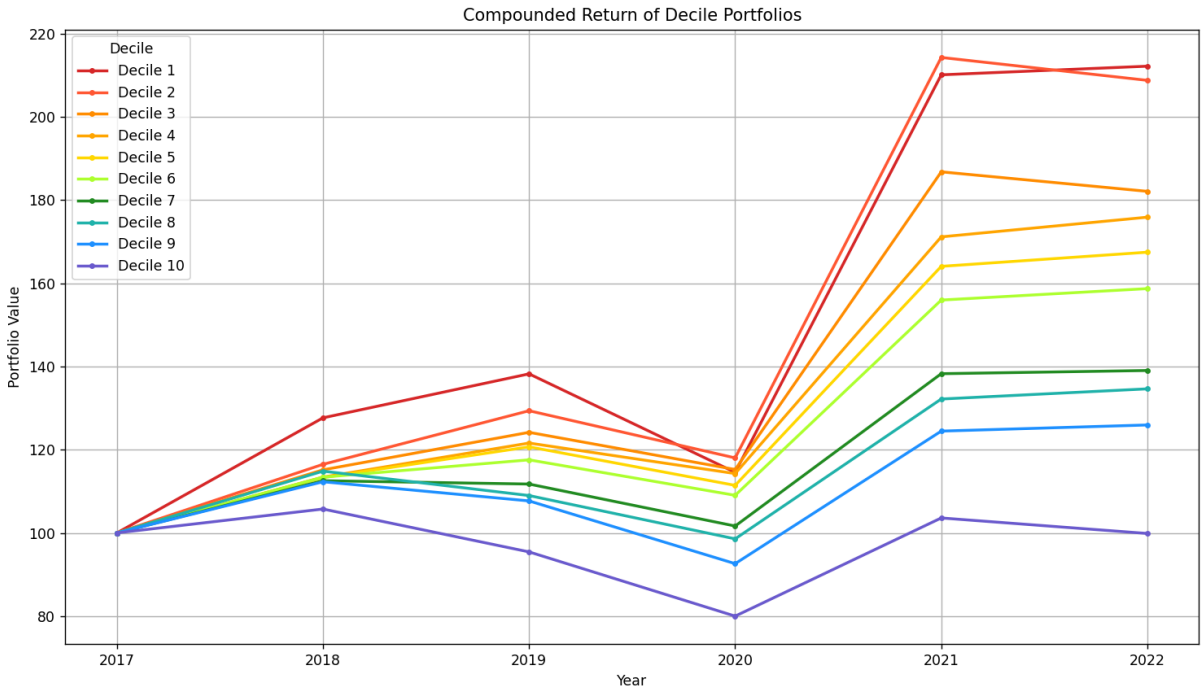

Figure 17: Compounded return of decile portfolios 2017-2022

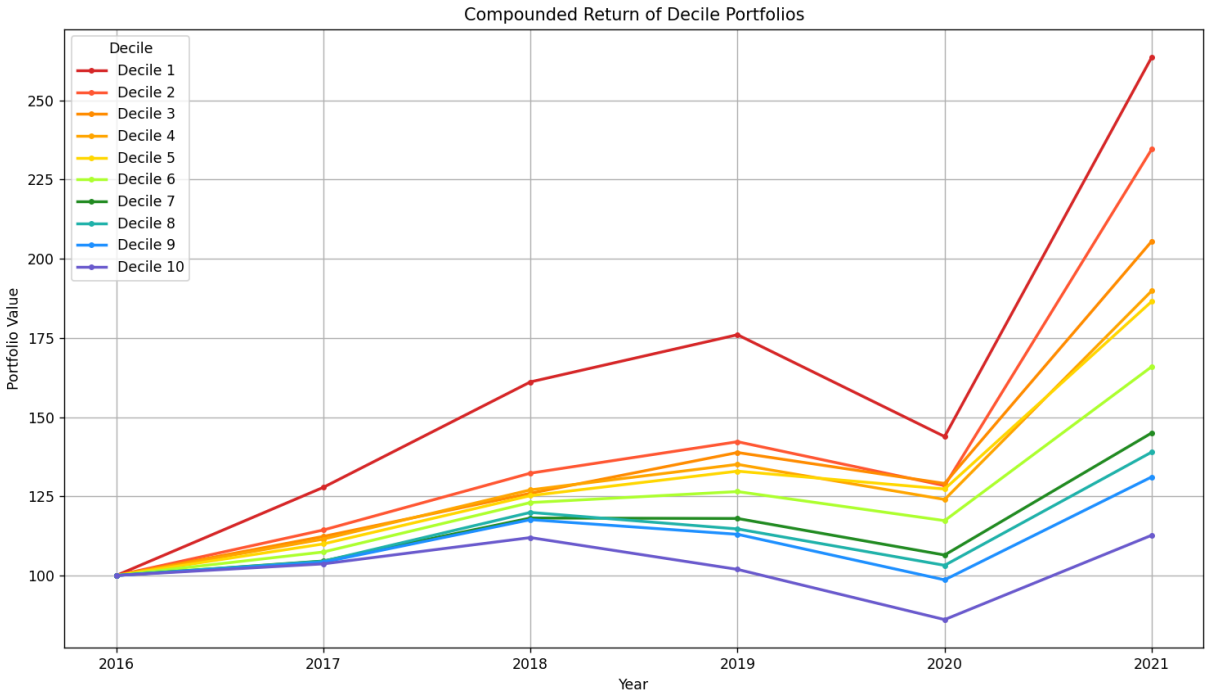

Figure 18: Compounded return of decile portfolios 2016-2021

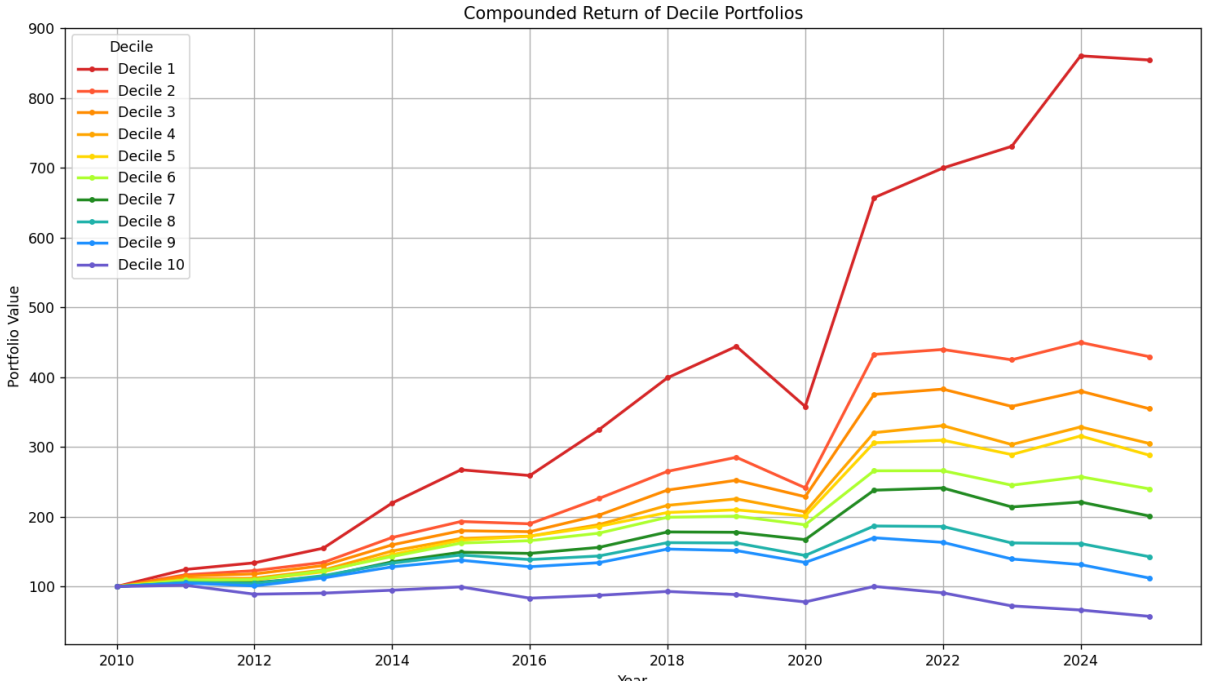

Figure 19: Compounded return of decile portfolios 2010-2025

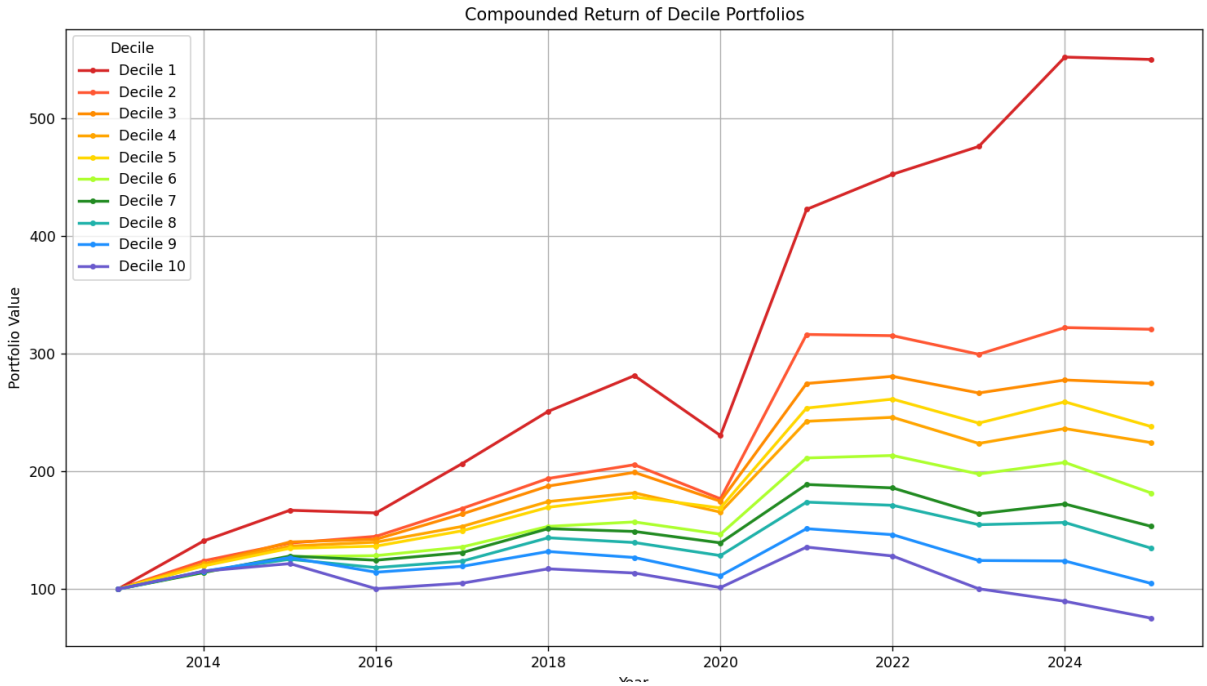

Figure 20: Compounded return of decile portfolios 2013-2025

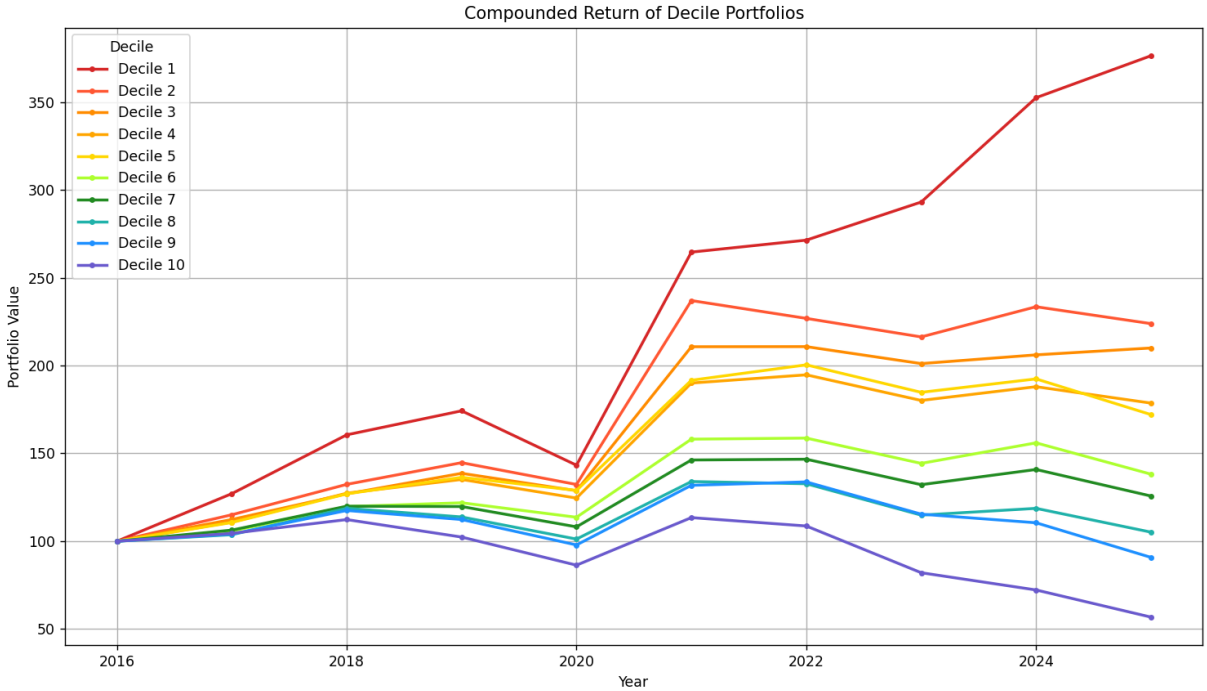

Figure 21: Compounded return of decile portfolios 2016-2025

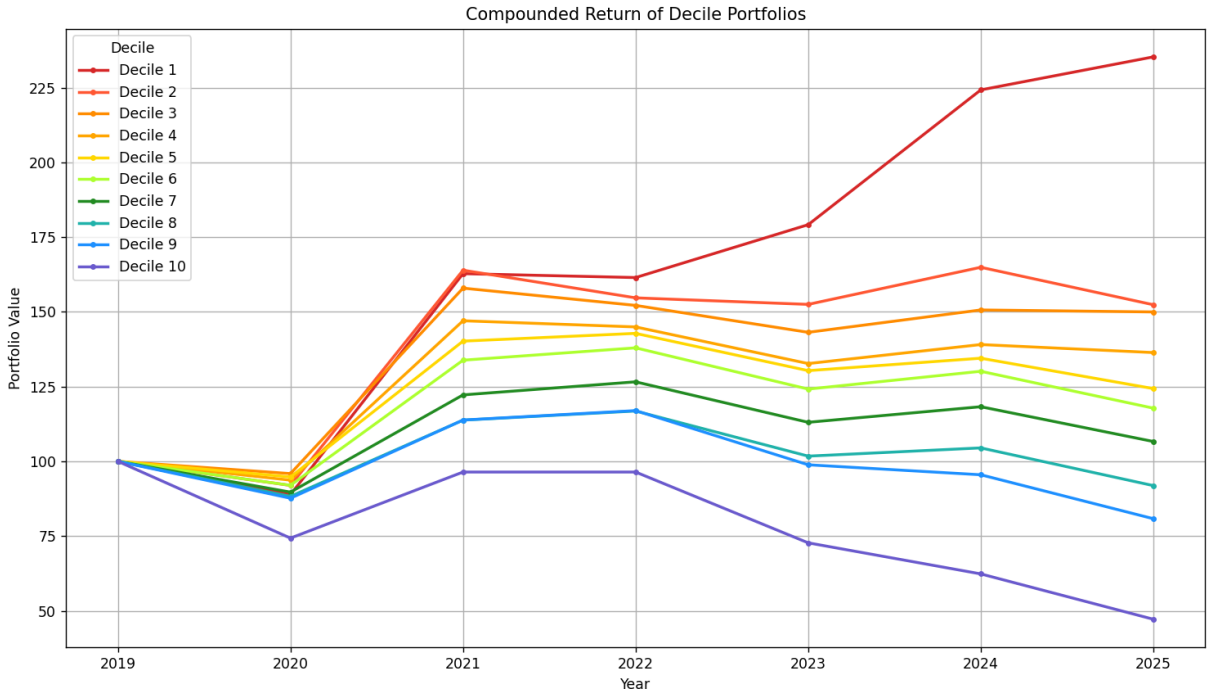

Figure 22: Compounded return of decile portfolios 2019-2025

Appendix C

Variance Inflation Factor (VIF) Values

Characteristic	VIF-value
Momentum6M	1.61
ROIC	3.91
GrossProfitMargin	1.48
DividendYield	4.33
MarketCap	2.56
ForwardEarningsYield	1.28
PriceToBook	1.01
Momentum1Y	1.68
DPSCAGR5y	1.11
EPSCAGR5y	1.35
ESG	1.29
EVEBITDA	1.50
DebtEBITDA	1.74
ROA	4.60
NetProfitMargin	1.69
PayoutRatio	4.14
Capex	1.30
TradingVolume	2.61
CurrentRatio	1.39

Table 11: Variance Inflation Factor values

Appendix D

Outlier Removal Conditions

Characteristic	Outlier Condition	Number of outliers removed[15]
ROIC	< 0 or > 80	23
GrossProfitMargin	< 0 or > 100	2
DividendYield	< 0 or > 10	1
ForwardEarningsYield	< 0 or > 25	9
PriceToBook	< 0 or > 50	18
Momentum1Y	< -0.95 or > 2	2
Momentum6M	< -0.95 or > 2	0
ROA	< 0 or > 80	22
EPSCAGR5y	> 100	2
DPSCAGR5y	> 100	1
EVEBITDA	< 0 or > 80	4
DebtEBITDA	> 30	0
NetProfitMargin	< 0 or > 80	33
PayoutRatio	> 110	0
Capex	> 50	13
CurrentRatio	< 0 or > 10	2
TradingVolume	-	-
MarketCap	-	-
ESG	-	-

Table 12: Outlier removal conditions

Appendix E

Results of Out-of-Sample Test Optimal Weights RF/ NN Ensemble Model

RF Weight (%)	NN Weight (%)	R^2	MAE	MSE
0	100	0.068	0.266	0.139
5	95	0.074	0.265	0.138
10	90	0.079	0.264	0.138
15	85	0.083	0.263	0.137
20	80	0.087	0.262	0.136
25	75	0.090	0.262	0.136
30	70	0.092	0.261	0.136

[15] Number of outliers removed, shown as an average per year. If a stock has outlier values for multiple characteristics, it counts separately for both characteristics.

35	65	0.094	0.261	0.135
40	60	0.095	0.260	0.135
45	55	0.095	0.260	0.135
50	**50**	**0.096**	**0.260**	**0.135**
55	45	0.095	0.260	0.135
60	40	0.091	0.260	0.136
65	35	0.089	0.261	0.136
70	30	0.085	0.261	0.137
75	25	0.081	0.261	0.137
80	20	0.076	0.262	0.138
85	15	0.071	0.262	0.139
90	10	0.065	0.263	0.140
95	5	0.058	0.264	0.141
100	0	0.050	0.265	0.142

Table 13: Optimal configurations weights RF/NN

Appendix F

Overviews of Filters in Screener Model

Stocks are ranked based on the ensemble model with a weight of 50/50 for both RF and NN. Additional filtering is possible based on the following filters in Table 14.

Index: Choice between (S&P 500, AEX and EURO STOXX 50)
Minimum MarketCap (in billions of dollars)
Maximum MarketCap (in billions of dollars)
Minimum PE-ratio
Maximum PE-ratio
Minimum Momentum6M (in %)
Maximum Momentum6M (in %)

Table 14: Filters in screener model

Appendix G

SLR Protocol:

This thesis uses a systematic approach to conduct the literature review included in Chapter 3. This SLR[16] consists of 7 steps to get a structured answer to the following knowledge question: *Which asset pricing models exist that predict an outperformance for certain stocks?*

1. Definition of knowledge problem and research question.
2. Defining inclusion and exclusion criteria.
3. Identification and selection of academic databases and sources to use.
4. Describing search terms and queries.
5. Making a flowchart and table with number of search results, including screening.
6. Making a conceptual matrix of all selected sources.
7. Integrating the theory organized around concepts and answering the knowledge problem.

We followed these 7 steps closely during the literature research involved in answering the given knowledge question.

[16] More information on how to perform an SLR is given in this practice guide: https://zenodo.org/records/7062727 (Cruz-Martínez & Rafael, 2022)