UNIVERSITÄT
LIECHTENSTEIN

Master Thesis

Optimizing S&P 500-Tilted Portfolios with Direct Weight Prediction Neural Networks: Information Ratio Performance under Realistic Constraints and Network Depth

Author Andreas Pischetsrieder

Rüfestrasse 9 9490 Vaduz

+4915255439733

a.pischetsrieder@uni.li

230218

University of Liechtenstein

Assessor: Sebastian Stöckl

Thesis project duration: 15.02.2025 to 30.06.2025

Date of submission: 30.06.2025

Abstract

This thesis investigates the performance of direct weight prediction neural networks for constructing equity portfolios that actively tilt away from the S&P 500 benchmark. While traditional portfolio optimization methods typically rely on two-stage pipelines of first forecasting returns and subsequently solving optimization problems this study proposes an end-to-end neural network model that directly predicts portfolio weights, aiming to improve the out-of-sample Information Ratio. The impact of realistic investment constraints is explored, such as transaction costs and leverage limits, and the incremental benefit of network depth on portfolio performance assessed. The methodology employs differentiable constraint layers to enforce investment restrictions during model training and applies a rolling-window backtesting framework to simulate real-world investment conditions. Empirical results demonstrate that direct weight prediction neural networks achieve a significantly positive Information Ratio compared to the benchmark, maintain robustness under practical investment constraints, and outperform shallow architectures, particularly under market regimes characterized by higher volatility or constrained optimization settings. This research provides evidence supporting the practical viability and effectiveness of deep neural networks for benchmark-relative portfolio construction.

Keywords: Neural Networks; Portfolio Optimization; Direct Weight Prediction; Information Ratio; Investment Constraints; Benchmark-relative investing

Table of contents

1 Introduction

This paper investigates the use of deep learning techniques for direct portfolio weight prediction in the context of large-cap equity markets, specifically focusing on portfolios that tilt away from the S&P 500 benchmark. Unlike traditional portfolio optimization pipelines that first forecast returns and then solve an optimization problem, this study explores end-to-end neural network models that directly learn to output portfolio weights. The core objective is to evaluate whether these models can consistently improve the portfolio's Information Ratio, particularly under realistic investment constraints. The paper also shows different architectures to determine under which conditions complexity contributes to better out-of-sample performance. Readers can expect both a theoretical motivation and a comprehensive empirical analysis, including backtesting frameworks and robustness tests. The aim is at assessing the practical viability of deep learning for real-world portfolio construction.

1.1 Motivation

Recent advances in Machine Learning and the persistent challenges of return predictability and instability in portfolio construction motivate a shift toward direct weight prediction models that align model training more closely with real-world investment performance.

Despite decades of academic research, cross-sectional return anomalies remain a reliable source of excess returns in U.S. equity markets, especially when using large-scale firm-level characteristics (Gu et al., 2020). However, traditional "predict-then-optimize" pipelines, where returns are first forecasted and then fed into an optimization step, struggle to fully exploit these signals due to compounding estimation errors. In such frameworks, forecasting noise propagates into extreme and unstable portfolio weights (Brandt et al., 2009), undermining real-world performance. While techniques like covariance shrinkage (Ledoit & Wolf, 2003) or empirical forecast correction (Barroso & Saxena, 2022) can mitigate some of this noise, they remain separate corrective steps rather than end-to-end solutions. This motivates the exploration of models that directly learn to construct portfolios by bypassing intermediate return forecasts and thereby aligning training objectives more closely with investment performance.

End-to-end weight-prediction networks promise to bridge the prediction-decision gap by directly aligning model training with portfolio performance. Recent research in equity markets demonstrates that deep neural networks trained to directly optimize portfolio-level objectives such as the Sharpe Ratio can outperform traditional two-step forecasting approaches. For example, Z. Zhang et al. (2020) train a feed-forward Long Short Term Memory (LSTM)-based model on U.S. equity index constituents to maximize the Sharpe Ratio, bypassing return forecasts entirely and achieving higher performance than benchmark methods under realistic constraints. Similarly, the AlphaPortfolio framework learns portfolio weights end-to-end from firm characteristics and macro signals, achieving out-of-sample Sharpe Ratios above 1.4 on monthly U.S. equity data, even after controlling for standard factor models (Cong et al., 2021). These results highlight that directly learning portfolio weights rather than indirectly forecasting returns can yield more stable, constraint-aware allocations and improve real-world investment outcomes.

Deep learning architectures, particularly transformers and nonlinear neural networks, have demonstrated a unique ability to capture complex cross-asset dependencies and adapt to shifting market regimes. Empirical results confirm their economic impact: Didisheim et al. (2024) show that deep "AIPT" models reach out-of-sample Sharpe Ratios near 3.7, compared to 0.8-1.2 for conventional six-factor benchmarks. Similarly, Kelly et al. (2025) report that transformer-based pricing models reduce prediction error by

30% and improve Sharpe from 3.9 to 4.6. Neural networks also outperform linear and tree-based models in predictive and economic terms, as shown in Guet al. (2020).

However, real-world implementation requires models to handle market frictions such as turnover, slippage, transaction costs, and leverage limits. Recent advances address this challenge by embedding constraints directly into the model architecture. For example, Deng et al. (2017) integrate transaction costs into a recurrent Reinforcement Learning (RL) model to improve live-vs-backtest alignment. Liu et al. (2023) propose Deep Inception Networks that penalize turnover and portfolio correlation, while Hwang et al. (2025) embed second-order-cone and long-only constraints into differentiable layers to enforce full investment and leverage caps during training.

Theoretical boundaries further shape expectations. Clarke et al. (2002) and Makamo (2023) quantify upper bounds on Information Ratios given investor skill (IC), implementation efficiency (TC), and real-world constraints. Frameworks like Black-Litterman (Black & Litterman, 1992) illustrate how incorporating priors or Bayesian adjustments can mitigate extreme allocations. These concepts are increasingly relevant for training direct-weight models that blend flexibility with stability.

Together, these insights motivate a focused examination of direct-weight neural networks for equity portfolios that tilt away from the S&P 500 benchmark and how well they perform under realistic investment constraints and evaluating whether more complex architectures offer measurable advantages over shallow networks.

1.2 Research gap

Despite promising results in specialized domains, end-to-end weight-prediction neural networks have primarily been tested on narrow or stylized asset universes such as small ETF baskets (Zhang et al., 2020) or cryptocurrency pairs (Jiang et al., 2017). These simplified settings, while useful for methodological development, lack the complexity and scalability of broad equity indices like the S&P 500, where turnover constraints, liquidity frictions, and signal dilution are significantly more pronounced.

Recent models have begun to incorporate realistic market frictions, such as transaction costs, leverage bounds, and turnover penalties (Deng et al., 2017; Liu et al., 2023; Hwang et al., 2025). However, these studies primarily demonstrate that constraints can be embedded in neural architectures - they do not systematically vary constraint severity to evaluate its quantitative effect on portfolio performance. As such, the sensitivity of Information Ratios to constraint levels (e.g., transaction cost regimes, gross leverage caps) remains largely unexplored.

Another unaddressed dimension is model architecture. Although deep networks outperform linear and tree-based methods in return prediction and asset pricing tasks (Gu et al., 2020; Didisheim et al., 2024), there has been no controlled, apples-to-apples comparison of deep versus shallow neural networks in a direct-weight prediction setting. Existing work does not isolate architectural depth effects under identical datasets, rebalancing frequencies, and constraint regimes, leaving open the question of when, and by how much, deeper models provide tangible performance benefits.

Additionally, while benchmark-relative portfolio management remains the dominant paradigm in institutional finance, most academic work continues to focus on absolute-return settings. No existing study explicitly frames the problem of tilting away from a broad market benchmark such as the S&P 500 as a direct-weight learning task, nor assesses its out-of-sample Information Ratio under realistic frictions such as costs and leverage constraints.

Beyond these core research gaps, several practical considerations remain underexplored. Few studies assess the real-world implementability of their strategies using metrics such as turnover, weight path stability, or tracking error. While some mention the need to reduce transaction costs or slippage, comprehensive evaluations of out-of-sample stability and robustness are lacking (Didisheim et al., 2024; Zhang et al., 2020). Furthermore, although model transparency is increasingly important in financial contexts, only a handful of studies (e.g., Cong et al., 2022) have begun to explore the interpretability of neural portfolio models using techniques like feature attribution or economic distillation.

Together, these limitations point to a clear need for a systematic, scalable, and constraint-aware evaluation of direct weight-prediction neural networks in the context of benchmark-relative equity investing. This thesis addresses that need by studying S&P 500 benchmark tilting under varying transaction cost and leverage constraints, and by comparing the performance of shallow and deep neural architectures in a controlled and reproducible backtesting framework.

1.3 Research questions and Hypotheses

Building on the identified research gaps, this thesis aims to evaluate the feasibility, robustness, and relative effectiveness of direct weight-prediction neural networks in constructing equity portfolios that tilt away from the S&P 500 benchmark. The following research questions guide the empirical investigation:

1. To what extent do direct weight prediction neural network models improve the Information Ratio by tilting away from the S&P 500 benchmark?
    - *Hypothesis (H_0): Direct weight prediction neural network models do not achieve a significantly positive Information Ratio when tilting away from the S&P 500 benchmark.*
    - *Hypothesis (H_1): Direct weight prediction neural network models achieve a significantly positive Information Ratio when tilting away from the S&P 500 benchmark.*
2. How do realistic investment constraints, such as transaction costs and leverage limits, impact the ability of direct weight prediction neural network models to achieve a significantly positive Information Ratio when tilting away from the S&P 500 benchmark?
    - *Hypothesis (H_0): The ability of direct weight prediction neural network models to achieve a significantly positive Information Ratio when tilting away from the S&P 500 benchmark disappears when realistic investment constraints, such as transaction costs and leverage limits, are applied.*
    - *Hypothesis (H_1): Direct weight prediction neural network models maintain a significantly positive Information Ratio when tilting away from the S&P 500 benchmark, even under realistic investment constraints, such as transaction costs and leverage limits.*

3. Do deep neural networks achieve higher Information Ratios than shallow neural networks when tilting away from the S&P 500 benchmark, and under what conditions does this advantage hold?
    - *Hypothesis (H_0): Deep neural networks do not achieve a significantly higher Information Ratio than shallow neural networks when tilting away from the S&P 500 benchmark.*
    - *Hypothesis (H_1): Deep neural networks achieve a significantly higher Information Ratio than shallow neural networks when tilting away from the S&P 500 benchmark, particularly under certain market conditions (e.g., high volatility, regime shifts, or constrained optimization settings).*

Together, these questions aim to assess not only whether direct-weight neural networks are a viable alternative to traditional portfolio construction pipelines, but also under what conditions they offer practical advantages in institutional equity settings.

1.4 Objective

Three primary contributions are advanced in this work. First, a comprehensive evaluation of direct-weight-prediction neural networks is undertaken for tilting away from the S&P 500 under realistic market frictions. Second, the effects of transaction costs and leverage limits on realized Information Ratios are systematically quantified through controlled sweeps of commission rates and gross-leverage caps, illuminating the trade-offs between active risk and implementability. Third, a head-to-head comparison of deep versus shallow architectures for direct-weight prediction is conducted on identical datasets and under the same constraint regimes, thereby isolating the representational benefits of depth in information-ratio maximization.

To realize these contributions, three objectives have been defined. The first objective involves the design and calibration of neural-network architectures that predict portfolio weights directly, embedding differentiable constraint layers to enforce long-only budgets, full investment, and second-order-cone leverage caps during training. The second objective centers on empirically assessing information-ratio improvements relative to the S&P 500, both in unconstrained settings and under varying levels of transaction costs and leverage limits, using a rolling-window backtest across multiple volatility regimes. The third objective is to benchmark deep and shallow network variants to determine under which market conditions and constraint severities additional layers yield statistically and economically significant gains in Information Ratio.

1.5 Structure of the Thesis

The structure of this thesis is designed to systematically investigate the performance of direct weight prediction neural networks in constructing S&P 500-tilted portfolios under realistic investment conditions. Each chapter serves the purpose in building toward this objective.

Following this first chapter with an introduction that outlines the motivation, formulates the central research questions, and presents the overall contributions of the study, the literature review in the second chapter situates the research within existing academic work on portfolio optimization and financial Machine Learning, identifying relevant gaps this thesis aims to address.

The third chapter focuses on the data description, detailing the dataset used, the selection of input features, and the formulation of the S&P 500 benchmark against which model performance is evaluated.

The fourth chapter presents the overarching methodological framework, defining the experimental design and the evaluation procedures employed throughout the empirical study. It introduces the neural network architecture and loss functions, specifying the model components and the training objective aligned with optimizing the Information Ratio. Furthermore, it describes the rolling-window backtesting framework, which simulates a real-world investment process to ensure out-of-sample performance evaluation. Finally it incorporates realistic investment constraints (transaction costs and leverage limits) into the framework to test the robustness and practical viability of the proposed approach.

This is followed by the empirical results in chapter five, which analyzes the performance of the neural network strategies in comparison to the benchmark and across different experimental conditions.

Chapter six provides a critical discussion of the findings, interpreting the results in relation to the research questions and broader literature. The final chapter concludes the thesis by summarizing the key insights, acknowledging limitations, and proposing directions for future research.

2 Literature review

This section provides a structured overview of the theoretical and empirical foundations relevant to benchmark-relative portfolio construction using Machine Learning. Foundational principles from classical portfolio theory are first reviewed, including the development of mean-variance optimization, the Sharpe Ratio, and the Information Ratio. These frameworks form the basis for evaluating portfolio performance both in absolute and active terms. Building on this, literature on active management is examined, with particular emphasis on the Fundamental Law of Active Management, estimation risk, and constraint-aware optimization techniques. The review then shifts to empirical findings from asset pricing and factor investing, highlighting the predictive power of firm characteristics and their role in generating alpha through systematic portfolio tilts. Finally, recent advances in Machine Learning and their application to return forecasting and portfolio construction are discussed, with attention given to their potential for improving the Information Ratio through nonlinear modeling and feature interactions. Throughout, the focus is placed on identifying methodological gaps and practical limitations that motivate the empirical analysis conducted in this thesis.

2.1 Classic portfolio theory

Markowitz (1952) laid the foundation for modern portfolio theory by characterizing the efficient set of portfolios in terms of expected return (mean m) and risk (variance σ). In his framework, a portfolio is efficient if no other portfolio offers a higher expected return for the same variance or a lower variance for the same expected return. He shows that, in an n-asset world, one can trace out a continuously-varying “critical line” of such efficient portfolios, the efficient frontier, by moving from the minimum-variance point up to the maximum-return point, switching across subspaces as constraints bind or release.

Shortly after, Roy (1952) offered an alternative “Safety-First” criterion that directly addresses the investor’s concern about disastrous losses. Rather than maximizing expected return alone, Roy’s rule seeks to minimize the probability that portfolio return falls below a given “disaster level” d. He shows via the Bienaymé–Tchebycheff inequality that this is equivalent to maximizing

$$\frac{m - d}{\sigma}$$

where m is the portfolio’s expected return and σ its standard deviation. This ratio generalizes the mean-variance approach by prioritizing downside protection against d (Roy, 1952, p. 434).

Tobin (1958) then extended Markowitz’s pure-risky-asset framework by introducing a risk-free asset (cash or consols offering yield r). Under his two-fund separation theorem, investors first identify the unique optimal risky combination of securities and then choose how much to allocate between that portfolio and the risk-free asset. If the risk-free return r exceeds a certain critical level, all wealth is allocated to consols; otherwise, to cash, effectively splitting the decision into (i) which risky portfolio to hold and (ii) how to blend it with the riskless asset.

Sharpe (1964) built on these results to derive a market equilibrium in which, once asset prices adjust for risk, investors can achieve any point along a straight Capital Market Line (CML). The CML plots expected return against total risk (standard deviation), with its intercept at the pure interest rate and its slope (the market price of risk) measuring extra return per unit of risk. Thus, higher expected returns demand bearing more risk, and the two-stage choice (risky-asset selection plus mix with the risk-free asset) is fully encapsulated by the CML.

Lintner (1965) and Mossin (1966) independently formalized these insights in a general-equilibrium Capital Asset Pricing Model (CAPM). Lintner shows that, in equilibrium, each firm’s market value equals the risk-free discount of expected cash flows minus a penalty term proportional to the asset’s covariance with the market portfolio. That proportionality constant is the market price of (dollar) risk, common to all assets in equilibrium. In effect, the expected return on any security must compensate investors for its contribution to aggregate market risk, measured by the sum of its own variance and its total covariance with all other assets. Mossin further proves that this equilibrium relationship implies a straight Security Market Line, relating an asset’s expected excess return linearly to its SD with a constant slope reflecting market conditions and an intercept representing the pure interest rate.

In equilibrium, Sharpe (1994) demonstrated that all mean-variance-efficient portfolios must lie on a CML of the form

$$E[R_P] = r_f + \underbrace{\frac{E[R_M] - r_f}{\sigma_M}}_{CML\ Slope} \sigma_P$$

where M is the market (tangent) portfolio, r_f the risk-free rate, and the slope $(E[R_M]-r_f)/\sigma_M$ measures the extra expected return per unit of total risk. Sharpe (1994) then observed that this “reward-per-unit-risk” concept could be applied to any portfolio P by defining its Sharpe Ratio as

$$Sharpe_p = E[R_p] - r_f \sigma_p$$

which is exactly the slope of the line drawn from (0, r_f) to the point (σ_p, E[R_p]) on the mean-variance plane. Consequently, the portfolio that maximizes this ratio (the tangent portfolio) is the point at which the CML just touches the efficient frontier, making it the single risky portfolio to which all investors optimally combine via two-fund separation.

Thus, the Sharpe Ratio generalizes the geometric insight of the CML into a universal performance metric: by measuring excess return per unit of total risk for any portfolio, it allows direct ranking of mean-variance efficiency across funds and strategies.

In direct analogy to the Sharpe Ratio's genesis from the CML, the Information Ratio can be understood as the slope of an "active" CML drawn relative to a passive benchmark rather than the risk-free rate. Formally, let

$$R_A = R_P - R_B \quad \text{and} \quad \sigma_A = \sigma(R_P - R_B)$$

denote the portfolio's active return and tracking-error volatility, then the Information Ratio (IR) is

$$\text{IR} = \frac{E[R_A]}{\sigma_A} = \frac{E[R_P - R_B]}{\sigma(R_P - R_B)}$$

Geometrically, this is exactly the steepness of the line connecting the origin of the active-risk/active-return plane (0,0) to the point (σA, E[RA]), just as the Sharpe Ratio is the slope of the line from (0 , r_f) to $(\sigma_p, E[R_p])$ on the total-risk/return frontier (Kidd, 2011).

The active tangent portfolio, which is the portfolio whose active-return-vs-active-risk point just touches the *active efficient frontier*, therefore maximizes this slope, making it the natural benchmark-relative analogue of the market (tangent) portfolio under the classic CML. In this way, the Information Ratio extends the mean-variance efficiency concept into the field of active management, measuring the highest achievable reward per unit of excess risk.

2.2 Active management

The fundamental law of active management relates the Information Ratio (IR) to manager skill and the number of independent bets taken. The formulation based on a factor model states

$$IR = IC * \sqrt{BR} * TC$$

where IC is the information coefficient (forecasting skill), BR the breadth (number of independent bets per year), and TC the transfer coefficient measuring implementation efficiency. This relationship is summarized in the "Fundamental law of active management under fundamental factor models" (Makamo, 2023). Under portfolio constraints, the transfer coefficient enters explicitly:

$$IR = TC * IC * \sqrt{N}$$

linking expected value added to the correlation between risk-adjusted active weights and forecasted returns. The more general "Fundamental Law of Active Portfolio Management" shows how TC adjusts realized active return when constraints cause deviations from optimal weights:

Turnover adjustments. Recent work extends the fundamental law to account for trading costs. F. Zhang et al. (2021) explicitly incorporate the volatility of the information coefficient and portfolio turnover costs. They prove that the turnover-adjusted Information Ratio is always lower than the unadjusted Information Ratio and that limiting turnover can improve performance despite the law's original implication.

Bayesian and shrinkage approaches. Jorion (1985) argues that sample mean estimates lead to unstable optimal portfolios. He proposes Stein-type shrinkage toward a common mean, which reduces estimation risk and improves out-of-sample performance. Ledoit & Wolf (2003) develop a practical shrinkage estimator for the covariance matrix. Extreme sample correlations are pulled toward the center to stabilize weights, with computer code provided for implementation.

Robust optimization. Goldfarb & Iyengar (2003) introduce uncertainty structures for market parameters and reformulate robust portfolio selection as second-order cone programs. These methods combat sensitivity to statistical and modeling errors and yield portfolios computed with specified confidence levels.

Risk-parity research. Asness et al. (2012) show that leverage aversion implies that safer assets must offer higher risk-adjusted returns. Risk-parity portfolios exploit this by equalizing risk across asset classes and overweighting safer assets relative to the market portfolio. Qian (2005) likewise advocates allocating market risk equally across stocks, bonds, and commodities, arguing that such diversification limits losses and can deliver superior returns for a given risk level.

Transaction costs. A realistic estimate for equity transaction costs is roughly 50 bp round-trip; value-weighted long/short portfolios typically incur more than 50 bp per round trip on average. Moreover, high-turnover factors like monthly-rebalanced momentum face costs near 48 bp per month (Novy-Marx & Velikov, 2016).

Together, these studies link the Information Ratio to skill and breadth, describe extensions for turnover costs, and present estimation techniques like Bayesian shrinkage and robust optimization that mitigate parameter uncertainty. Risk-parity approaches provide an additional framework for managing portfolio risk under leverage aversion. Incorporating these frictions is essential, as even modest spreads can substantially erode gross returns, lower Information and Sharpe Ratios, and in some cases turn theoretically profitable strategies into losses once implemented.

While the bulk of the literature reviewed so far treats portfolio choice as a single-period problem whether it's mean-variance optimization, the Safety-First ratio, or end-to-end Sharpe-maximizing neural nets. Real-world portfolios are rebalanced continuously or at least periodically as new information arrives. A natural extension is the continuous-time framework pioneered by Merton (1969), which embeds portfolio choice in a stochastic differential equation and solves for time-varying weights via a Hamilton-Jacobi-Bellman (HJB) equation.

In Merton's model, an investor with constant relative risk aversion (CRRA) and wealth W_t allocates dynamically between a risky asset ($dWtdS_t/S_t = \mu\, dt + \sigma\, dW_t$) and a risk-free asset at rate r. The resulting optimal feedback rule

$$\pi_t^* = \frac{\mu - r}{\sigma^2}\frac{W_t}{W_t} = \frac{\mu - r}{\sigma^2}$$

is time-invariant in the simplest case, but if returns are predictable or preferences are non-myopic, the allocation becomes explicitly time- and state-dependent.

Subsequent work has introduced return predictability and transaction costs into this continuous-time setting. For example, Ma et al. (2019) derive in closed form a finite-horizon, continuous-time trading strategy. Their solution shows that the investor gradually trades toward a dynamic "aim" portfolio, which itself is a weighted sum of future Merton portfolios, with weights reflecting both the persistence of predictive signals and the drag of trading costs, and converges to the frictionless Merton line as the terminal date approaches.

More recently, continuous-time RL methods have been developed that learn such dynamic policies end-to-end. Wang & Zhou (2020) formulate an entropy-regularized, exploratory version of the continuous-time mean-variance problem, showing that the optimal exploration policy is a Gaussian whose variance decays as a function of time remaining, while its mean recovers the classical Merton tilt. By embedding such structures into deep-RL algorithms (e.g. those that optimize a multi-period Sharpe Ratio directly) one can capture both dynamic dependence on state variables and real-world frictions (costs, constraints) within the learning objective.

2.3 Factor Investing and Return Predictability

A substantial body of empirical research has documented persistent cross-sectional differences in expected stock returns based on firm-specific characteristics. These patterns - commonly referred to as return anomalies - have led to the development of factor investing strategies that systematically tilt portfolios to exploit these characteristics.

Value-related signals were among the earliest anomalies identified. Basu (1977) showed that stocks with high earnings-to-price (E/P) ratios outperformed their low E/P counterparts, suggesting market underreaction to fundamental value. Similarly, Fama and French (1992) found that the book-to-market (B/M) ratio, as well as total assets relative to market capitalization (am), predicted higher average returns. These results formed the basis of the Fama-French three-factor model. Sales-to-price (sp) and dividend yield (div_yield_st) are further value-oriented measures linked to future performance.

Momentum represents another robust return anomaly. Jegadeesh and Titman (1993) showed that stocks with high returns over the past 12 months (mom12m) tend to continue outperforming, a finding replicated across time periods and markets.

Quality-related factors have been shown to predict future returns as well. Gross profitability (gp), defined as gross profits over total assets, was found by Novy-Marx (2013) to have predictive power comparable to value. Return on equity (ro_e) has also been linked to future performance, with higher-quality firms delivering excess returns (Ball et al., 2016). Piotroski's (2000) F-score (ps) and

Mohanram's (2005) G-score (ms) provide composite measures of accounting-based quality for value and growth stocks, respectively.

Investment and accrual-based signals offer additional insights. Firms with high investment-to-revenue ratios (investment) or high net operating assets (noa) tend to underperform, consistent with overinvestment and accrual mispricing (Sloan, 1996; Hirshleifer et al., 2004). Relatedly, firms with rapid asset growth have been shown to deliver lower returns (Cooper et al., 2008).

Volatility and risk factors also play a significant role. Ang et al. (2006) showed that stocks with high idiosyncratic volatility (idio_vol3f) earn lower returns on average, a result that challenges traditional risk-return intuition. Volume to market (vol_mkt) has likewise been associated with cross-sectional variation in returns (Bali et al., 2005).

Distress risk are further return determinants. Franzoni and Marin (2006) introduced pension funding status (fr) as a predictive signal, showing that firms with poorly funded pensions tend to earn lower returns.

Intangible investment and innovation signals have also gained attention. Chan et al. (2001) and Eberhart et al. (2004) documented that firms with high R&D expenditures relative to market cap (rd) tend to outperform, consistent with underappreciated intangible capital.

Together, these characteristics form the empirical foundation for modern factor investing. By tilting a portfolio toward stocks exhibiting favorable signals - such as high E/P, strong momentum, high profitability, low investment, or low volatility - investors can exploit these return premia to achieve alpha relative to cap-weighted benchmarks like the S&P 500.

Empirical evidence supports the effectiveness of such tilts. Cheng and Srivastava (2019) found that five of six single-factor S&P 500 indices (value, momentum, quality, low-volatility, and dividend yield) outperformed the benchmark from 1995–2018 in terms of Sharpe and Information Ratios. Shu and Mulvey (2024) report that a regime-switching multi-factor strategy achieved an Information Ratio of 0.4, significantly higher than the 0.05 reported for an equal-weighted benchmark.

These findings suggest that the use of firm-level predictors in portfolio construction can systematically enhance benchmark-relative performance. The predictors employed in this thesis (spanning value, momentum, quality, risk, liquidity, and investment signals) are grounded in decades of asset pricing research and serve as the basis for the direct-weight learning models evaluated in later chapters.

Variable	Definition	Academic Origin
div_yield_st	Predicted div yield next month	Litzenberger & Ramaswamy (1979)
ep	Earnings-to-Price Ratio	Basu (1977)
mom12m	Momentum (12 month)	Jegadeesh & Titman (1993)
idio_vol3f	Idiosyncratic risk (3 factor)	Ang et al. (2006)
investment	Investment to revenue	Titman, Wei and Xie (2004)
bm	Book to market using most recent ME	Rosenberg, Reid and Lanstein (1985)
am	Total assets to market	Fama and French (1992)
fr	Pension Funding Status	Franzoni and Marin (2006)
gp	Gross Profits / Total Assets	Novy-Marx (2013)
ms	Mohanram G-score	Mohanram (2005)
noa	Net Operating Assets	Hirshleifer et al. (2004)
ps	Piotroski F-score	Piotroski (2000)
rd	R&D over Market Cap	Chan, Lakonishok and Sougiannis (2001)
ro_e	net income / book equity	Haugen and Baker (1996)
sp	Sales-to-price	Barbee, Mukherji and Raines (1996)
vol_mkt	Volume to market equity	Haugen and Baker (1996)

Table 1: List of anomalies (Chen & Zimmermann, 2022)

2.4 Machine Learning in Asset Management

The growing complexity and dimensionality of financial data have prompted increasing interest in applying Machine Learning (ML) methods to asset pricing and portfolio construction. Traditional linear factor models, while interpretable, are limited in their ability to capture non-linear interactions, high-order effects, and time-varying relationships between firm characteristics and returns. Machine Learning techniques offer more flexible function approximation, enabling more effective modeling of such complexities.

Recent research has demonstrated that Machine Learning methods can significantly enhance the performance of return prediction models. Gu et al. (2020) benchmarked a range of Machine Learning algorithms - including regression trees, random forests, and deep neural networks - against traditional linear models in cross-sectional return prediction. Their findings show that Machine Learning-based forecasts lead to substantially higher out-of-sample Sharpe Ratios. The performance gains were attributed to the models' ability to uncover nonlinear interactions and context-dependent relationships that linear specifications miss.

In practical applications, these return forecasts are often fed into a subsequent optimization step to construct portfolios. Kaczmarek & Perez (2022) show that combining Machine Learning-based signal selection with mean-variance optimization leads to higher Information Ratios than using linear signals alone. This two-step framework - forecasting returns and then solving an optimization problem - remains the dominant paradigm in Machine Learning-based asset allocation.

However, this separation between prediction and decision introduces a structural inefficiency: the forecasting model is not trained to optimize the portfolio’s performance metric directly. Estimation errors and noise in return forecasts can propagate through the optimization step, resulting in unstable or suboptimal portfolio weights. This “prediction-decision gap” is particularly problematic in constrained optimization settings, where small changes in input estimates can lead to large shifts in weights (Brandt et al., 2009).

To address this issue, recent studies have explored end-to-end portfolio learning frameworks in which Machine Learning models are trained directly on portfolio-level objectives such as Sharpe Ratio or Information Ratio. Instead of predicting returns, these models learn to output portfolio weights that maximize a utility function or risk-adjusted performance criterion. For example, Zhang et al. (2020) apply deep neural networks trained to directly maximize the Sharpe Ratio in ETF portfolios, bypassing intermediate return forecasting altogether. Cong et al. (2022) develop the AlphaPortfolio framework, which uses attention-based deep RL to construct portfolios directly from asset features, showing superior performance under various market conditions and constraints.

This class of models which is referred to as direct weight prediction models represents a promising shift in portfolio construction methodology. By aligning training objectives with investment goals, these models integrate signal extraction and decision-making into a unified architecture. Moreover, recent work has demonstrated that differentiable optimization layers can embed real-world constraints, such as turnover limits, leverage caps, or transaction costs, directly into the learning process (Hwang et al., 2025).

Despite their potential, most applications of end-to-end portfolio learning have focused on niche domains (e.g., cryptocurrency or small ETF universes), and few have applied these techniques to large-scale benchmark-relative equity portfolios such as the S&P 500. Furthermore, existing studies have rarely examined the effect of constraint severity or systematically compared the performance of shallow versus deep architectures in a direct-weight setting.

These limitations highlight the need for further research on the application of direct weight prediction neural networks to equity benchmark tilting. This thesis aims to address these gaps by empirically evaluating the Information Ratio performance of shallow and deep neural network models trained to directly output constrained portfolio weights in a U.S. large-cap setting.

2.5 Summary and Link to Research Questions

The literature reviewed across classical portfolio theory, active management, factor investing, and Machine Learning provides a comprehensive theoretical and empirical foundation for benchmark-relative portfolio construction. Classical frameworks such as mean-variance optimization, the CML, and the Sharpe Ratio establish how risk-adjusted returns are evaluated in absolute terms. The Information Ratio extends this logic to benchmark-relative investing, enabling the assessment of active performance via portfolio tilts.

Research on active management emphasizes the importance of forecasting skill, breadth, and implementation efficiency, all of which contribute to the Information Ratio as described by the Fundamental Law of Active Management. Techniques such as shrinkage estimators, robust optimization, and turnover-adjusted performance models address the instability introduced by estimation noise and real-world frictions.

Empirical asset pricing studies have consistently identified firm-level characteristics, such as value, momentum, profitability, and investment, as strong predictors of future returns. These findings provide the basis for factor-tilt strategies, which have been shown to enhance Information Ratios when applied to large-cap indices like the S&P 500.

Machine Learning has emerged as a powerful tool for modeling the complex and nonlinear relationships that underlie these return premia. While ML-based return prediction has yielded substantial gains, most approaches operate in a two-step framework that separates prediction from portfolio construction. Recent research on direct weight prediction models addresses this limitation by learning portfolio weights end-to-end, optimizing directly for investment objectives such as the Information or Sharpe Ratio.

However, existing applications of direct-weight neural networks remain limited in scope, often confined to narrow asset universes and without rigorous testing under realistic investment constraints. Systematic comparisons between deep and shallow architectures, particularly in benchmark-relative contexts such as S&P 500 tilting, are also lacking.

These gaps motivate the empirical questions examined in this thesis, which investigates the extent to which direct-weight prediction neural networks can improve Information Ratios, how their performance is affected by constraints such as transaction costs and leverage limits, and whether deep architectures provide a measurable advantage over shallow alternatives under varying market conditions.

3 Data description

The dataset comprises monthly observations on all firms that at any point belonged to the S&P 500 index over the period January 1960 through December 2022. After merging CRSP PERMNOs with Compustat identifiers, 1871 distinct firms are identified in total, each tracked from its first S&P 500 listing through its final exit or end of the sample. Monthly returns, market capitalizations, and benchmark weights are drawn from CRSP, while firm-level signals (e.g. dividend-yield, book-to-market, past-year momentum) derive from Compustat. S&P 500 membership flags are updated each month;

additions and deletions are handled by carrying forward the prior month's flag until the official index change date.

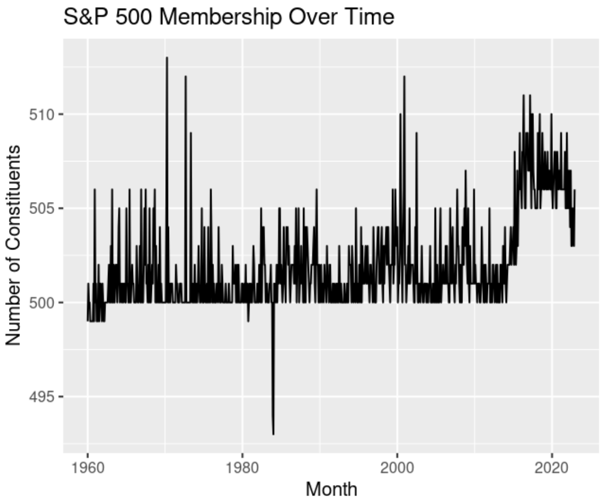

Figure 1: S&P 500 constituents over time

All series are re-sampled to a common end-of-month calendar date. The raw data span 629,408 firm-months. The target variables comprise the next-month excess return (ret), the firm's market-capitalization weight in the market-capitalization-weighted benchmark, and a binary index-membership indicator (sp500_flag). Furthermore standard control variables are included: total market capitalization (mktcap), and a monthly identifier (yyyymm) for cross-validation purposes.

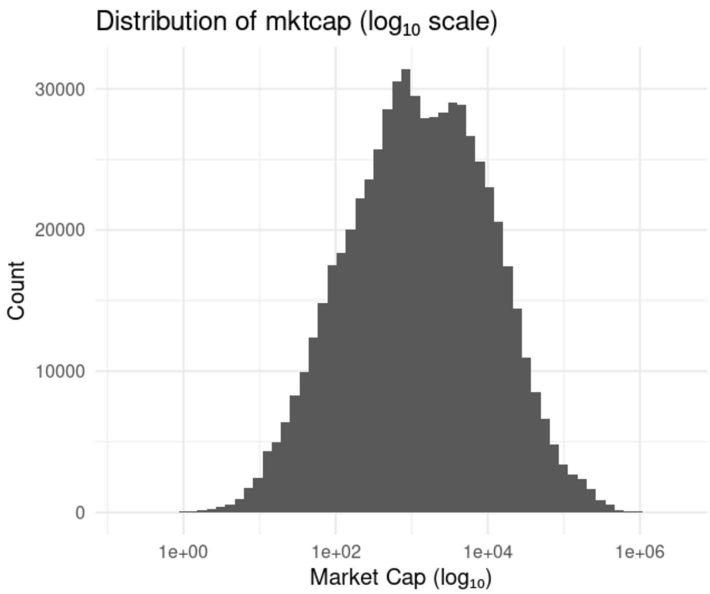

Figure 2: Distribution of market capitalization

The benchmark exhibits increasing concentration over time. The 90th percentile benchmark median weight has nearly doubled since the 1960s.

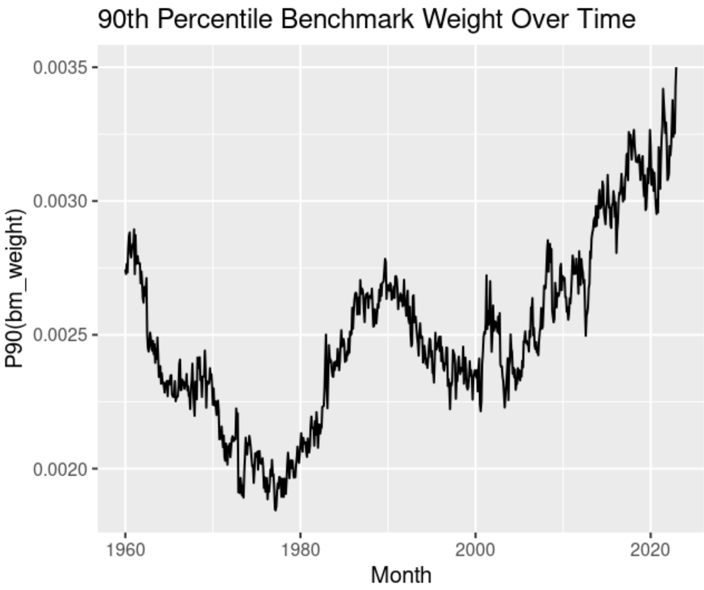

Figure 3: 90th percentile benchmark weight over time

The predictor set consists of 16 normalized signals, each transformed via a uniform empirical cumulative distribution function (CDF) to lie in [0,1]: dividend-yield (div_yield_st), earnings-to-price ratio (ep), 12-month momentum (mom12m), three-factor idiosyncratic risk (idio_vol3f), investment to revenue (investment), book-to-market (bm), assets-to-market (am), Pension Funding Status (fr), Gross Profits / Total Assets (gp), Mohanram G-score (ms), net operating assets (noa), Piotroski F-score (ps), R&D over market cap (rd), net income/ book equity (ro_e), Sales -to-price (sp), and Volume to market equity (vol_mkt).

Across the full sample, the cross-sectional distribution of next-month returns has mean 1.37 %, standard deviation 11.2 %, negligible skewness, and extreme kurtosis due to rare large outliers. Each of the 16 predictors is fully non-missing by construction and uniformly distributed on [0,1] by design, with minor deviations in the empirical quartiles due to tied ranks.

Monthly counts of S&P 500 constituents fluctuate modestly around 500 (see Figure 1: S&P 500 constituents over time), reflecting periodic turnover. In the earliest sample month (1960-01), there are 499 constituents, rising to 501 in 1960-02, then settling at 500–501 for the next several months. The average tenure of a stock in the index (first through last listing) is approximately 10 years, with a broad right-skew reflecting some mega-caps that persist through multiple decades.

At each month's start, the median benchmark weight is roughly 0.0464 % and the 90th percentile weight is 0.274 %, while the single largest firm rarely exceeds 6.7 % of total cap. Across the sample, there is gradual concentration over time, with the top-decile weight trending from ≈0.27 % in the 1960s to around 0.60 % by 2022.

Pairwise correlations among the 16 predictors are generally modest (|ρ|< 0.7), though some anticipated clusters emerge: dividend-yield and sales-to-price are negatively correlated (ρ≈−0.58), equity issuance and gross profitability correlate positively (ρ≈0.33), and earnings-to-price correlates strongly with accruals-to-market ($\rho \approx 0.60$). Correlations between signals and subsequent returns are small ($|\rho| < 0.10$), consistent with the low cross-sectional predictability of individual signals (see Table A 1: Correlations of predictors).

Before analysis, it is verified that zero missing values remain after signal construction and winsorization. No imputation beyond the uniform-rank transform is necessary.

All 16 predictors are transformed via the uniform empirical CDF within each month. Post-transform, their minima are approximately 0.001, maxima are exactly 1.000, and the median is near 0.50. This scaling ensures cross-sectional comparability and guards against extreme values.

For model training and evaluation, rolling-window splits with a 5-year training window and a one year test window is chosen.

While the dataset is free of survivorship bias and missing values, it remains subject to look-ahead risks if signal construction inadvertently uses data unavailable at forecast time, and to microstructure noise in small-cap names. These pitfalls are avoided by strictly lagging all fundamentals and momentum signals by one month and focusing on firms above the 10th market-cap percentile in robustness checks (Goyal and Welch, 2008; Ledoit and Wolf, 2004).

4 Methodology

This chapter outlines the full methodological framework developed to evaluate the performance of direct weight-prediction neural network models in benchmark-relative portfolio construction. The goal is to assess whether such models can generate portfolios that consistently outperform a dynamic S&P 500 benchmark on a risk-adjusted basis, while remaining implementable under realistic investment constraints. The methodology is designed to simulate the full lifecycle of a quantitative investment strategy, from data preparation and model training to portfolio formation, evaluation, and robustness analysis.

The workflow is structured into four core components: (1) Data Preparation and Benchmark, (2) Neural Network architecture, (3) training procedure, (4) backtesting, (5) postprocessing with realistic portfolio constraints, and (6) performance evaluation using Information Ratio and other metrics. Each step has been designed to reflect practical considerations encountered in asset management, such as turnover control, capital allocation limits, and data integrity. By integrating constraint-aware modeling, out-of-sample evaluation, and economic performance targets, this framework aims to produce a fair and rigorous assessment of the economic value added by direct weight-prediction models.

4.1 Data Preparation and Benchmark

The benchmark is constructed to replicate the composition and weighting of the S&P 500 index using an sp500_flag variable, which indicates whether a firm was an official constituent of the S&P 500 in a given month. For each month in the sample, all firms with sp500_flag = 1 are included in the benchmark universe and value-weighted allocations are assigned based on each firm's market capitalization. This approach preserves the historical composition of the S&P 500 over time, allowing firms to enter and exit the benchmark dynamically in accordance with documented index membership changes.

Unlike a purely size-ranked index, this method ensures fidelity to the actual S&P 500 structure as it evolved historically. Sector exposures, style compositions, and firm inclusion are determined by the index's official membership at each point in time. No sector or style constraints are imposed; sector weights are allowed to evolve organically according to index composition and firm capitalization. This ensures that any active tilts taken by the model are not mechanically constrained by the benchmark design, aligning with the guidance in Clarke et al. (2002) on maintaining a clean separation between benchmark and active decisions.

The benchmark is rebalanced monthly to reflect changes in membership and market capitalization. All benchmark returns are computed as total returns, including dividends, and are used as the reference for computing active returns, tracking error, and Information Ratios in subsequent performance evaluation.

For each firm in the benchmark universe, a vector of monthly-updated predictor variables is constructed. The feature set includes 16 firm-level signals drawn from both fundamental and market-based information, spanning a broad spectrum of return-predictive dimensions documented in the asset pricing literature and explained in the literature part above.

All predictor variables are normalized cross-sectionally each month using a uniform empirical CDF, mapping values to the [0, 1] interval. This transformation ensures comparability across features and mitigates the impact of outliers, while preserving rank information. The resulting feature set is fully non-missing by construction and stable over time.

This diverse set of signals reflects a synthesis of predictive variables identified in prior empirical studies, including Gu et al. (2020) and Kaczmarek & Perez (2022). By incorporating predictors from multiple anomaly categories, such as value, momentum, quality, and risk, the network is allowed to learn complex, nonlinear return patterns that may arise from interactions among characteristics.

4.2 Neural Network Architecture

The neural network is designed to map firm-level characteristics directly to portfolio weight deviations relative to a benchmark. Rather than predicting returns or scores that are later optimized, the network outputs weights in an end-to-end supervised learning framework that aligns directly with portfolio-level objectives. This approach follows the parametric policy framework of Brandt et al. (2009), adapted to modern neural architectures.

4.2.1 Architectural Configurations

A total of four configurations are evaluated, combining two levels of architectural complexity with two degrees of constraint severity:

Model Type	Hidden Layers	Output Bounds	L1 / L2 Penalty
Deep-Unconstrained	64 → 32 → 16 → 8	[-1.00, +1.00]	0.001 / 0.001
Deep-Constrained	64 → 32 → 16 → 8	[-0.05, +0.05]	0.01 / 0.01
Shallow-Unconstrained	32 → 16	[-1.00, +1.00]	0.001 / 0.001
Shallow-Constrained	32 → 16	[-0.05, +0.05]	0.01 / 0.01

Table 2: Architectural Configurations

All networks are fully connected and use the ReLU activation function in the hidden layers to introduce nonlinearity. Each layer is followed by a dropout layer with a rate of 0.2 to mitigate overfitting. The final output layer produces a single scalar per stock (i.e., the predicted delta weight) and uses a box-sigmoid activation function to bound outputs within a defined range:

$$\widehat{w_i} = \alpha \cdot (2 \cdot \sigma(z_i) - 1)$$

where $\sigma(\cdot)$ is the sigmoid function, z_i is the pre-activation output, and α is the bounding limit either 1 for unconstrained models or 0.05 for constrained models. These bounds act as architectural priors that prevent extreme weight recommendations and help stabilize gradient flow during training.

4.2.2 Output Interpretation

Each predicted output $\widehat{w_i}$ represents a delta weight to be added to the benchmark allocation. These deltas are not normalized during training; instead, the resulting active weights are normalized within the loss function so that their absolute weight sum equals one per month. The final model outputs are permitted to include short positions (negative weights), though long-only constraints are enforced later in post-processing.

The constrained configurations are intended to simulate more realistic investment behavior by embedding stricter position limits and stronger regularization. These models are expected to produce smoother, more diversified portfolios at the cost of potentially lower flexibility in tracking signals.

4.2.3 Loss Function

The neural network is trained using a loss function that directly represents the portfolio's performance goal to maximize the Information Ratio of the portfolio's active returns. The Information Ratio is defined as the mean active return divided by its standard deviation, i.e.

$$\mathrm{IR} = \frac{E\left[R_p - R_b\right]}{\sqrt{\mathrm{Var}\left(R_p - R_b\right)}}$$

where R_p is the portfolio return and R_b is the benchmark (S&P 500 proxy) return. Equivalently, it is the Sharpe Ratio of the active return (excess return over the benchmark). The training objective is formulated as maximizing the Information Ratio on the training window. However, because most optimization software minimizes a loss, the negative Information Ratio is minimized. In practice, the loss for a given training period is:

$$\mathcal{L} = -\frac{\overline{R_p - R_b}}{\sqrt{\mathrm{Var}\left(R_p - R_b\right)}} + \lambda_1 |\widehat{w}|_1 + \lambda_2 |\widehat{w}|_2^2$$

Here, $\widehat{w}$ refers to the predicted delta weights output by the neural network, i.e. the deviations from the benchmark portfolio that the model recommends for each stock. To prevent overfitting and ensure stability in portfolio construction, two regularization terms are added to the loss function: an L1 penalty and an L2 penalty, weighted by hyperparameters λ1 and λ2, respectively.

The L1 penalty **($|\widehat{w}|_1$)**, also known as lasso regularization, is defined as the sum of the absolute values of the predicted weights:

$$|\widehat{w}|_1 = \sum_i |\widehat{w_i}|$$

This term encourages sparsity in the network's output by pushing many weights toward exactly zero. In the context of portfolio construction. By penalizing the use of many small positions, L1 regularization encourages the model to concentrate only on the most relevant signals. Assets that are not strongly justified by the model's learned patterns may receive weights near zero. Sparse portfolios require fewer trades and adjustments, thereby improving implementability.

The L2 penalty **($|\widehat{\boldsymbol{w}}|_2^2$)**, or ridge regularization, is the sum of the squared predicted weights:

$$|\widehat{w}|_2^2 = \sum_i \widehat{w_i^2}$$

This penalty discourages large-magnitude weights, particularly those that could result from overfitting to noise in the training data. The L2 term favors more evenly distributed weightings rather than extreme allocations to a few stocks. It reduces variance in out-of-sample performance by preventing the model from reacting too strongly to spurious patterns. Penalizing large outputs helps the model avoid producing unstable portfolios that might work in-sample but fail in live trading.

Together, these penalties serve as statistical analogues of shrinkage methods in traditional portfolio optimization. Just as covariance matrix shrinkage (Ledoit & Wolf, 2004) stabilizes optimized portfolios by dampening estimation noise, L1 and L2 regularization shrink the model's outputs toward more moderate, interpretable, and robust allocations. The key intuition is that extreme weights are rarely reliable in noisy financial environments. Penalizing such behavior leads to models that are not only more robust but also more consistent with the practical constraints of institutional asset management (Gu et al., 2020).

4.3 Training Procedure

The network is trained within a rolling-window walk-forward framework, designed to emulate a realistic investment process that updates the model periodically over time. Specifically, a 5-year (60-month) estimation window is used for training, followed by a 12-month out-of-sample test period after a 1-month offset to avoid look-ahead bias.

At each iteration, the model is trained on five years of historical data (e.g., January 1960 to December 1964). After skipping one month (January 1965), the trained model is used to generate out-of-sample portfolio weights for the next 12 months (February 1965 to January 1966). During this prediction phase, the model remains fixed and is not updated further. After this one-year test period, the training window is advanced by 12 months, and the process is repeated using the newly available data. This rolling retraining cycle continues until the end of the sample, producing a time series of monthly portfolio weights from 1965 through 2022, each based solely on information available at the time of prediction.

This rolling design ensures that each monthly forecast is strictly out-of-sample, with no information from future months used in training. The procedure prevents optimistic bias and reflects the actual operational workflow of an investor who retrains a model annually based on historical data and holds it fixed during a live forecasting period (Kaczmarek & Perez, 2022).

During each 5-year training window, standard Machine Learning techniques are applied to improve generalization. A dropout rate of 0.2 is used to mitigate overfitting, and the Adam optimizer is initialized with a learning rate of 0.0005. A learning rate scheduler reduces the rate when the loss plateaus. Early stopping is employed based on the training loss, with a patience parameter of 8 epochs. No separate validation set is used.

Each retraining yields a newly initialized and regularized model that is then deployed for one year of monthly predictions. This setup strikes a balance between statistical power and responsiveness to regime shifts, as the 5-year window is presumed to reflect conditions relevant to the near-term future. This ensures sufficient data for robust parameter estimation. Longer and shorter training windows were tested; five years provided a favorable trade-off between model stability and out-of-sample Information Ratio consistency (Coqueret & Guida, 2021).

4.4 Backtesting Procedure

The strategy is evaluated using a rolling out-of-sample backtest that reflects a realistic, time-consistent investment process. For each annual forecast period produced by the rolling training framework (Section 4.3), a sequence of monthly portfolio weights is generated and applied to realized returns over the following 12 months. The process unfolds as follows:

At the beginning of each out-of-sample month, the previously trained model - fixed for that calendar year - receives the most recent available feature data and outputs a vector of delta weights relative to the benchmark. These raw outputs are bounded within a specified range ([–1%, +1%] for the unconstrained model and [–0,05%, +0,05%] for the constrained model) via the box-sigmoid activation. The delta weights are added to the benchmark weights to form the model's active portfolio for that month, and are subsequently adjusted through post-processing (Section 4.5) to enforce full investment, long-only constraints, and diversification limits.

Using the lagged feature inputs and prior month's closing prices, the portfolio is rebalanced at the start of each month and held constant throughout. The realized return of the strategy is computed based on these fixed weights and the total return of each stock (including dividends) over the month. Benchmark returns are computed similarly, using value-weighted allocations of the official S&P 500 constituents. The monthly active return is then calculated as the difference between the strategy return and the benchmark return.

This monthly process is repeated across the full one-year forecast horizon, with the model held fixed during this period. At the end of each test year, the model is retrained on a new 5-year window of data, and a new set of 12 monthly forecasts is generated. By repeating this cycle across the entire sample, a time series of monthly portfolio weights and returns is constructed, spanning 1965 through 2022. Each month's portfolio decision is based strictly on information available prior to that month, ensuring that performance statistics reflect genuine out-of-sample prediction and portfolio construction capability.

The backtest reflects realistic trading assumptions. Portfolios are rebalanced at a monthly frequency, consistent with the feature update interval. No intra-month adjustments are made, and no forward-looking data is used. All returns are measured as total returns, inclusive of dividends. Benchmark weights are updated monthly to reflect official index membership and lagged market capitalization.

This rolling backtest structure is widely recommended in quantitative finance for its ability to simulate how an investment strategy would perform under operational constraints (Barroso & Saxena, 2022). It captures the effects of model re-training, changing market regimes, and forecast degradation over time. Moreover, it allows performance metrics, such as the Information Ratio, Sharpe Ratio, turnover, and drawdown, to be evaluated using strictly out-of-sample data. In this way, the strategy's robustness and practical viability can be assessed under conditions that approximate a live deployment.

4.5 Post-Processing of Portfolio Weights

After the neural network outputs the raw weights for a given month, a series of post-processing steps is applied to ensure the portfolio is feasible and adheres to practical investment constraints. These steps transform the network's predicted weights (which are in principle active weight signals) into an implementable portfolio. The adjustments are as follows:

4.5.1 Full Investment and Long-Only Conversion

A 100% net investment constraint is enforced, meaning all capital is invested and the sum of portfolio weights equals 1. In the raw output, the sum of weights may not equal 1 (especially if some predictions are negative or if the sum of positives is less than 1). First any net bias are eliminated by ensuring no funds are left idle or over-leveraged. If short positions are present after clamping (see 4.5.3 below), it is assumed the portfolio is allowed to short and hold the proceeds as extra long positions (maintaining zero net cash). However, in the final implementation net shorting is disallowed, so effectively the weights of long positions are scaled such that they add up to 100% of capital. This normalization to unity is a standard step in portfolio construction (Chevalier et al., 2022). It ensures the portfolio is fully invested at all times. Practically, if w_i^{raw} are the network outputs' delta weights applied on the benchmark weights for stocks i=1...N, adjusted weights $w_i = \frac{\max(w_i^{\text{raw}},0)}{\sum_j \max\left(w_j^{\text{raw}},0\right)}$ are computed. This sets negative predictions to 0 (no short positions) and renormalizes the longs to sum to one. The result is a long-only portfolio (after clamping) that uses 100% of capital. Maintaining full investment keeps the strategy's benchmark exposure consistent. A fully invested active portfolio is always compaired to a fully invested benchmark.

4.5.2 Turnover Smoothing

To avoid sudden large changes in the portfolio, a turnover smoothing mechanism is implemented using an exponential moving average with a weight of 0.2 (20%). Instead of moving directly to the new optimal weights each month, they are blended with the previous month's weights:

$$w_t^{\text{final}} = 0.2 \cdot w_t^{\text{predicted}} \;+\; 0.8 \cdot w_{t-1}^{\text{final}}.$$

This means only 20% of the weight difference is realized in the current rebalance, with the remaining 80% carried over from the prior allocation. The intuition is to gradually adjust the portfolio rather than making full discrete jumps. Such smoothing significantly reduces turnover, trading costs, and avoids abrupt changes that could be infeasible in practice (e.g., liquidating a large position entirely in one go). It also recognizes that the model's outputs may contain short-term noise. By tempering the shift, it is acted on more persistent signals. This approach reflects real-world portfolio management where "asset managers fine-tune their positions with small shifts… rather than extreme changes", as abrupt on/off positions are impractical (Chevalier et al., 2022). Notably, in the Chevalier et al. (2022) study of direct weight learning, the authors observed that directly learned weights were inherently smoother and incurred much lower turnover than traditional optimized weights. Exponential smoothing further enforces this smoothness. A smoothing factor of 0.2 was chosen to significantly dampen churn while still allowing the portfolio to track evolving signals (approximately giving an average half-life of around 3 to 4 months to a new signal). The average turnover is monitored (fraction of the portfolio traded) to ensure it stays within reasonable bounds (see 4.6 Performance Evaluation below).

4.5.3 Weight Clamping and Per-Stock Caps

After determining the new target weights (post-smoothing), each stock's weight is constrained to the interval [0%, 10%] of the portfolio. This implements the two practical constraints no short positions (weights can't go below 0, reinforcing long-only stance) and no single stock can exceed 10% of the total portfolio value. The 10% cap per stock is a common diversification constraint in asset management. It prevents the portfolio from concentrating too heavily in one name, thereby mitigating idiosyncratic risk. By capping weights, also the likelihood of extreme bets driven by possibly spurious model confidence is reduced. If the model for some reason scored one stock extremely highly, the cap ensures a more moderate position size. In traditional optimization, unconstrained solutions often put very large weights on a few stocks (corner solutions) when expected returns appear high (Chevalier et al., 2022). Imposing a max weight avoids such corner solutions, forcing the portfolio to spread investments. This is beneficial because extremely concentrated portfolios are unstable and sensitive to estimation error -

a slight change in expected return could swing an optimal allocation from one stock to another. The 10% cap ensures a minimum level of diversification at all times. This cap was selected in line with typical mutual fund guidelines and to allow meaningful overweights (since the benchmark weights of top S&P 500 names can be ~5-6%, a 10% cap still permits a noticeable overweight) without allowing a single position to dominate. The no-short constraint (weight ≥ 0) aligns with practical long-only portfolio mandates (Clarke et al., 2002). Many institutional investors do not allow shorting due to risk and regulatory reasons, and even if shorting were allowed, it would introduce leverage and additional costs that are excluded here. Thus, after this step, the portfolio is fully invested and long-only, each stock between 0 and 10% weight, and all weights summing to 100%.

4.5.4 Diversification Control via Herfindahl–Hirschman Index (HHI)

As a final check on concentration, the HHI of the portfolio weights, defined as $\mathrm{HHI} = \sum_{i=1}^{N} w_i^2$ is calculated. An upper limit of HHI ≤ 0.30 is imposed. This ensures the portfolio maintains a minimum level of diversification in aggregate. The HHI is a measure of concentration (commonly used in antitrust to measure market share concentration, but also applicable to portfolios: an HHI of 1.0 means a single stock at 100%; an HHI of 0 would be infinitely diversified into infinitesimal positions). An HHI of 0.30 corresponds, for example, to 3-4 equally-weighted stocks (since 1/3^2 =0.333, 1/4^2=0.25), or more stocks with uneven weights. In practice, after the 10% cap, the portfolios usually have many stocks and an HHI well below 0.30; however, if the model were to concentrate in ~3 large positions at 10% each and the rest very small, HHI could approach 0.30. Those top holdings would then be slightly scaled back until HHI is under the threshold. This rule is a safeguard against excessive concentration, complementing the per-stock cap with an aggregate metric. It effectively forces the effective number of stocks in the portfolio to be at least on the order of 3 to 4 or more. The motivation is to ensure the portfolio's risk-adjusted performance is not overly reliant on just a few bets. The Fundamental Law of Active Management indicates that breadth (number of independent bets) can enhance Information Ratio given a fixed skill level. By capping HHI, breadth is encouraged. The portfolio must hold a basket of active positions, not just one or two. This should lead to more consistent performance and reduce volatility of active returns stemming from any single stock. If HHI is within limits, no further action is needed beyond this monitoring.

After all these post-processing steps, the final portfolio weights for the month are recieved. These weights are then used to calculate the portfolio's return (given stock returns) and become the starting point for next month's smoothing. The post-processing pipeline is applied sequentially in the following order: (1) long-only clamping, (2) budget normalization, (3) exponential turnover smoothing, and (4) HHI-based diversification adjustment. They represent realistic trading practices: maintaining a fully invested portfolio, trading incrementally, limiting position sizes, and ensuring diversification. Importantly, these adjustments are applied outside the learning loop (ex post to the network outputs). In future work, one could incorporate some of them (e.g. budget constraint or short-sale ban) into the model or loss function itself. In the current setup, however, it is more straightforward to let the network recommend unconstrained active weights and then to handle the constraints through this rule-based overlay. This two-stage approach guarantees that the final portfolio is trading-feasible and in line with fiduciary constraints, which is critical for making the backtest realistic. The literature often emphasizes that applying such portfolio constraints can reduce the theoretical maximum Information Ratio but increases the likelihood the strategy's performance is realized in practice (Clarke et al., 2002). This methodology explicitly acknowledges this trade-off and opts for a slightly more constrained but more robust portfolio construction.

4.6 Performance Evaluation

The out-of-sample performance of the resulting strategy is evaluated using several industry-standard metrics for portfolio returns.

4.6.1 Information Ratio and Sharpe Ratio

Annualized Information Ratio is the primary metric of this research, since the model was optimized for Information Ratio. It is computed as the mean active return divided by the standard deviation of active returns, annualized. Using the monthly active return series from the backtest, the average excess return of the portfolio over the benchmark per month is found, multiplied by 12 to get annualized excess return, and divided by the standard deviation of monthly active returns ($\sqrt{12}$ for annualization). The Information Ratio summarizes how much incremental return the strategy earned per unit of tracking error risk. A higher Information Ratio indicates better benchmark-adjusted performance. This measure is analogous to the Sharpe Ratio but specific to active management (benchmark as the reference). Because the model explicitly targeted Information Ratio, the realized Information Ratio is expected to be positive and significantly higher than zero if the model added value.

The Sharpe Ratio of the portfolio's absolute returns (excess over risk-free rate) is also inspected. This is calculated similarly, but using the portfolio's total return minus the risk-free rate as the numerator, and the standard deviation of portfolio returns as the denominator. The Sharpe Ratio evaluates the strategy's performance on an absolute basis, accounting for volatility. It is possible for a strategy to have a high Information Ratio but if the active strategy carries only small active risk, its total volatility might be low – the Sharpe Ratio helps contextualize the absolute performance. The Sharpe Ratio is annualized by scaling monthly mean excess return and volatility by 12 and $\sqrt{12}$ respectively. In this case, since the benchmark is a broad equity index, the portfolio's Sharpe Ratio will reflect how much the strategy improved performance over just holding the index or cash. A Sharpe Ratio above the benchmark's Sharpe Ratio would indicate outperformance on a risk-adjusted basis.

4.6.2 Turnover and transaction costs

Portfolio turnover is measured to assess trading intensity. For each month, turnover equals the sum of absolute differences between the portfolio's target weights and the previous weights grown by one plus the actual returns. The total turnover reported in is simply the sum of these per-period turnover values. Turnover is important because it directly relates to transaction costs and strategy capacity. The average turnover will be reported and discussed whether the smoothing (0.2 Exponential Smoothing Factor) successfully kept trading to a reasonable level. Low turnover is generally preferred, as it implies the strategy is more stable and cost-efficient.

To present a more realistic performance, transaction costs are incorporated into the strategy returns. A flat 0.5% round-trip transaction cost is assumed, applied monthly to portfolio turnover. This figure encompasses brokerage commissions and short-term market impact typical for large-cap stocks. 0.5% are chosen as a conservative estimate of trading friction in an institutional context trading S&P 500 constituents (which are highly liquid). Some months with low turnover incur negligible costs, while high-turnover months could see a performance drag due to trading. Including transaction costs is vital because a strategy that looks good on paper could be eroded by implementation costs if it trades too aggressively (Z. Zhang et al., 2020). By accounting for costs, it is ensured the reported Information Ratio and Sharpe Ratio are net of reasonable fees.

4.6.3 Additional Risk Diagnostic

The Certainty Equivalent Return (CER) is the guaranteed (risk-free) return that, when received with certainty, delivers the same expected utility as a given risky portfolio. Formally, for a utility function U(·) (e.g. CRRA), CER solves

$$\mathrm{U(CER)} = \mathrm{E}\left[\mathrm{U}\left(\mathrm{r_p}\right)\right]$$

where $\mathrm{r_p}$ denotes the portfolio's return.

CER is important because it embeds risk preferences. Unlike raw returns or Sharpe Ratios, it accounts for the entire return distribution (mean, variance and higher moments) as weighted by the investor's utility function (Simon et al., 2025). It provides an intuitively comparable metric, expressing performance in "basis-point" terms makes it easy to quantify the economic value added by different strategies or model enhancements. For example, deep parametric policies have been shown to deliver CER improvements of 200–300 bp over linear benchmarks at moderate risk-aversion levels, underscoring substantial economic gains (Simon et al., 2025). CER is reported for all model variants to quantify economic value added beyond risk-adjusted performance.

In the context of a CRRA investor, $\boldsymbol{\gamma}$ denotes the coefficient of relative risk aversion, which governs the curvature of the utility function and thus the investor's trade-off between mean return and variance. Equivalently, in a mean-variance form it appears in

$$u\left(r_p\right) = r_p - \frac{\gamma}{2}\,\mathrm{Var}\left(r_p\right)$$

so that larger γ imposes a heavier penalty on volatility.

Empirical studies of neural-network portfolio policies typically calibrate γ in the 2-10 range, with some analyses extending up to γ=20 to test robustness (Simon et al., 2025).

Here a moderate value γ =5 is used. It reflects a balanced aversion to risk without overstating the penalty on upside variability, and it lies at the heart of many portfolio-optimization studies.

Another metric used is the active share. It reports the diviation from the benchmark and is defined as the average, across all months, of one half the total absolute deviation between portfolio weights and benchmark weights. This measures how much the portfolio deviates from the benchmark on average.

$$\text{Active Share} = \frac{1}{2}\sum_{i=1}^{n}\left|w_i^{\text{portfolio}} - w_i^{\text{benchmark}}\right|$$

All performance metrics are computed on the out-of-sample test set of monthly returns resulting from the rolling backtest. These performance metrics are subjected to statistical evaluation. The Newey-West adjusted t-tests on active returns assess whether the Information Ratio is significantly different from zero. The portfolio's Sharpe Ratio is compared to the benchmark's to determine whether active management provided incremental value.

4.7 Factor Attribution via Rolling Decile-Based Betas

To interpret which predictive signals contribute most to portfolio performance, a post-hoc factor attribution procedure is implemented. This module decomposes the realized excess return of the portfolio into contributions from individual factors, each corresponding to one of the predictors used during training. The analysis provides insight into whether the model's active performance is primarily driven by known risk premia or residual, unexplained components.

4.7.1 Inputs and Signal-Based Factor Returns

The attribution function requires the following inputs:

- **Delta weights**: Model-generated stock-level portfolio deviations from the benchmark
- **Stock returns**: Realized monthly total returns
- **Factor loadings**: Stock-level exposures to each predictive signal
- **Predicted and realized factor returns**: Time series of top-minus-bottom decile returns per factor

For each month and signal, a top-minus-bottom decile portfolio return is calculated by sorting stocks on that signal, then taking the return differential between the highest and lowest deciles. These realized factor returns represent unstandardized proxies for the performance of each characteristic.

4.7.2 Estimation of Time-Varying Betas

For each stock and signal, 36-month rolling regressions are used to estimate time-varying betas, representing each stock's sensitivity to a given factor over time. These rolling betas allow dynamic measurement of exposures that reflect structural shifts in market regimes or firm characteristics.

4.7.3 Portfolio-Level Factor Exposures

For each month and factor, the portfolio's exposure is calculated by taking the weighted average of stock-level betas using the model's delta weights as the cross-sectional weighting scheme:

$$\text{Exposure}_{f,t} = \sum_i \widehat{w_{i,t}} \cdot \beta_{i,f,t}$$

Each factor's monthly contribution to portfolio return is computed as:

$$\text{Contribution}_{f,t} = \text{Exposure}_{f,t} \cdot \text{Return}_{f,t}^{f}$$

Where Return_t^f is the realized top-minus-bottom return for factor f in month t.

4.7.4 Residual Return and Performance Attribution

The model's monthly portfolio excess return (relative to the benchmark) is computed by applying the delta weights to the realized next-month stock returns. The residual return that is interpreted as model-specific alpha not explained by factor exposures is calculated as:

$$\text{Residual}_t = R_{p,t} - \sum_f \text{Contribution}_{f,t}$$

This isolates the portion of performance attributable to nonlinear interactions, unmodeled signals, or statistical noise.

4.7.5 Summary Statistics and Interpretation

The procedure outputs a time series of factor exposures, contributions, and residuals. As well as a summary tables of average monthly contribution by factor and model. Visualizations showing how different signals drove performance are shown in the apendix.

This analysis enables both diagnostic insight into which signals are utilized most effectively and economic attribution by connecting model performance to known risk factors. It also serves as a sanity check if the model consistently outperforms but most return is unexplained by any signal, it may be overfitting or relying on patterns not grounded in fundamental finance. By complementing black-box prediction with economically grounded attribution, this step strengthens the interpretability and accountability of the modeling process.

To place the results in context, relevant literature benchmarks are cited. For example, Gu et al. (2020) report that Machine Learning based strategies (neural networks, random forests) can achieve Sharpe Ratios around 1.3 for long-short decile portfolios. The strategy is constrained (long-only, tracking an index), so IRs in the 0.5-1.0 range would already be considered impressive for an active equity fund. Also the findings are compared to Kaczmarek & Perez (2022), who found that using Machine Learning forecasts in a mean-variance optimizer outperformed equal-weighted portfolios, reinforcing the value of Machine Learning in portfolio construction. Additionally, the insight of Barroso & Saxena (2022) acknowledges that forecast errors can be systematic. The rolling re-training and regularization are intended to adapt to and correct any consistent mis-predictions over time.

In summary, the methodology ensures reproducibility and rigor: every step from data gathering, feature engineering, model training, to portfolio simulation and evaluation is specified in detail. By grounding design choices in academic literature (e.g., direct weight optimization (Brandt et al., 2009), regularization for stability (Ledoit & Wolf, 2004), and Machine Learning integrated with portfolio optimization (Kaczmarek & Perez, 2022; Chevalier et al., 2022) each component of the methodology is justified. The end result is a comprehensive, end-to-end process for evaluating a neural-network-driven portfolio strategy, with careful attention to realistic constraints and performance measurement. This methodological framework would allow an informed reader to replicate the study or to understand how each element (data, features, model, training regime, constraints, metrics) contributes to the robustness of the strategy's evaluation.

5 Results

This section presents the empirical results of the backtests across four direct weight-prediction neural network configurations: deep and shallow architectures, each in both constrained and unconstrained forms. The results are organized in three stages: (1) raw network output before post-processing, (2) after weight clamping and long-only normalization, and (3) after applying the full post-processing pipeline. In each case, key metrics such as annualized return, Information Ratio, Sharpe Ratio , turnover, active share, and CER are reported. The implications for model complexity, implementability, and benchmark-relative performance are discussed in relation to the three research questions.

5.1 Raw network output

Portfolio Weights Summary with Advanced Metrics										
Portfolio	**Weights (Min/Max)**	**Weights M(SD)**	**Std.Dev./Date M(SD)**	**HHI M(SD)**	**Diversity**	**L1 Norm M(SD)**	**L2 Norm M(SD)**	**Avg. Short Weights M(SD)**	**Average Turnover**	**Total Turnover**
keras_constrained_deep_1	-0.0264/0.0945	0.0032 (0.0061)	0.0058 (0.0011)	0.3763 (0.3237)	13.3170	38.7199 (30.8988)	0.5319 (0.3055)	2253.2129 (2023.4133)	2.0376	1540.388
keras_unconstrained_deep_2	-0.2161/0.3874	0.0158 (0.038)	0.0358 (0.0093)	13.461 (12.5957)	0.3976	250.8057 (202.4223)	3.1528 (1.8763)	2549.0348 (2258.8803)	16.2770	12305.402
keras_constrained_shallow_3	-0.0273/0.0962	0.004 (0.0061)	0.0058 (0.0011)	0.4162 (0.3634)	12.0203	42.4715 (33.844)	0.5589 (0.3221)	1828.5064 (1589.1323)	2.6091	1972.476
keras_unconstrained_shallow_4	-0.3727/0.3383	0.0349 (0.0479)	0.0452 (0.0101)	28.0884 (24.6703)	0.1782	379.2573 (300.5192)	4.5947 (2.6415)	1745.5006 (1580.3491)	35.6460	26948.387

Table 3: Weight summary before postprocessing

Before post-processing, the network-generated delta weights reflect unconstrained optimization of the Information Ratio objective. Table 3 reports descriptive statistics for the weight distributions, including L1/L2 norms, Herfindahl-Hirschman Index (HHI), and turnover. The unconstrained models exhibit extreme weight ranges and high concentration, whereas the constrained models (with L1 = L2 = 0.01 and tighter output bounds of ± 5% via the box-sigmoid activation) produce smoother, more diversified allocations.

For **Constrained Deep (λ=0.01)** raw weights now lie between [-2.64 %, +9.45 %], with a mean absolute weight of 0.32 % and a cross-sectional volatility of 0.58 %. Concentration falls precipitously compared to the unconstrained models. HHI = 0.376 and diversity = 13.32 while gross-exposure norms collapse to L_1 = 38.72 and L_2 = 0.5319. Although average short-position magnitude remains large (≈ 2,253; SD 2,023), average monthly turnover shrinks to 2.04 (total = 1,540), an order-of-magnitude reduction versus the unconstrained deep.

For **Constrained Shallow (λ=0.01)** weight bounds tighten to [–2.73 %, +9.62 %], with mean absolute weight 0.40 % and per-date dispersion 0.58 %. HHI = 0.416 and diversity = 12.02 signal broad diversification, and L_1 and L_2 norms diminish to 42.47 and 0.5589. Average short exposure of 1,829 is unchanged in sign, but turnover falls to 2.61 per month (total = 1,972).

By contrast, the unconstrained deep (Δ∈[–21.6 pp, +38.7 pp]) retains its prior profile: HHI = 13.46 (SD 12.60), diversity = 0.398, L_1 = 250.8, L_2 = 3.15, and turnover = 16.28 (total = 12,305). Likewise, the unconstrained shallow remains concentrated (HHI = 28.09, diversity = 0.178), with L_1 = 379.3, L_2 = 4.59, and turnover = 35.65 (total = 26,948).

These results confirm that regularization alone can significantly reduce excessive tilt magnitudes, trading activity, and concentration, without changing the model architecture.

Portfolio Performance Summary									
Portfolio	**Annualized Mean**	**Sharpe Ratio**	**Active Share**	**Turnover**	**TC Adjusted Sharpe**	**Information Ratio**	**CER (gamma=5)**	**TC Annualized Mean**	**TC Information Ratio**
keras_constrained_deep_1	1.1602***	2.394***	3.5779	1540.388	2.1664***	0.2363***	0.5731	1.0379***	0.2129***
keras_unconstrained_deep_2	7.6781***	3.0966***	27.4902	12305.402	2.7685***	0.2646***	-7.6925	6.7015***	0.2365***
keras_constrained_shallow_3	1.2405***	2.1506***	3.9307	1972.476	1.9058***	0.2068***	0.4087	1.0839***	0.1824***
keras_unconstrained_shallow_4	11.5509***	2.1863***	41.5530	26948.387	1.7815***	0.1848***	-58.2344	9.4121***	0.1503***
benchmark	0.1132	0.7484	0.0000	113800	0.6876	NaN	0.0560	0.1042	-0.1676

Table 4: Performance summary before postprocessing

Table 4 presents pre-processing performance. The unconstrained shallow model posts the highest gross return (1155.09%) and TC adjusted Sharpe Ratio of 1.78, but also incurs the highest turnover and negative CER (−58.23). The constrained deep model delivers a more modest but still questionable 116.02% return but maintains a TC adjusted Sharpe Ratio of 2.17 and CER of 57.31%. All four variants outperform the benchmark Sharpe (0.75), but none of the models seem to yield economically viable post-cost results.

Against the S&P 500 benchmark (11 % return, Sharpe 0.75, TC return 10 %), all four neural portfolios deliver superior risk-adjusted performance in gross terms. Crucially, the heightened L_1/L_2 regularization in the new constrained models dramatically curtails active share and turnover (by roughly 85–90 %) while preserving Sharpe Ratios above 2.0 and Information Ratios above 0.20, foreshadowing more implementable strategies when combined with post-processing frictions. The unconstrained models serve as theoretical upper bounds on the raw expressiveness of the network, but without post-processing, their weight profiles and trading intensity render them unsuitable for real-world implementation.

5.2 Weight clamping and long-only normalization

Portfolio Weights Summary with Advanced Metrics										
Portfolio	**Weights (Min/Max)**	**Weights M(SD)**	**Std.Dev./Date M(SD)**	**HHI M(SD)**	**Diversity**	**L1 Norm M(SD)**	**L2 Norm M(SD)**	**Avg. Short Weights M(SD)**	**Average Turnover**	**Total Turnover**
keras_constrained_deep_1	0/0.0953	0.0017 (0.0033)	0.0031 (7e-04)	0.0863 (0.071)	34.4807	10.6126 (7.4161)	0.2633 (0.1303)	0 (0)	0.6485	490.2311
keras_unconstrained_deep_2	0/0.0957	0.0017 (0.0033)	0.0031 (7e-04)	0.0862 (0.0712)	35.1787	10.6111 (7.4148)	0.2628 (0.131)	0 (0)	0.6009	454.2594
keras_constrained_shallow_3	0/0.0961	0.0017 (0.0032)	0.0031 (7e-04)	0.086 (0.0713)	36.0367	10.614 (7.4172)	0.2621 (0.1316)	0 (0)	0.6030	455.8931
keras_unconstrained_shallow_4	0/0.0966	0.0017 (0.0032)	0.0031 (7e-04)	0.0859 (0.0715)	36.9213	10.612 (7.4155)	0.2614 (0.1323)	0 (0)	0.5412	409.1616

Table 5: Weights summary after introduction of weight caps

Table 5 shows that imposing only a simple weight-restriction-clamping all positions to [0,10 %] and enforcing full investment drastically homogenizes the four portfolios' exposures, regardless of network depth or tilt constraint. After this adjustment, every model allocates exclusively long positions, with individual weights ranging from 0 to roughly 9.7%. The cross-sectional mean weight is uniformly 0.17 % (SD ≈ 0.33 %).

Concentration, as measured by the HHI, falls to approximately 0.086, while the corresponding diversity metric increases to between 34.5 and 36.9, indicating broad and nearly identical name distributions across all variants. Norm-based measures of gross exposure likewise converge: the L_1 norm of the weight vector averages 10.61 (SD ≈ 7.42) and the L_2 norm 0.26 (SD ≈ 0.13). All short-position exposures are eliminated by construction.

Trading activity is similarly subdued and consistent, with average monthly turnover between 0.54 and 0.65 (total sample turnover 409 to 490), representing a further tenfold reduction relative to the unconstrained pre-processing benchmarks. In sum, these results demonstrate that a simple post-hoc clamp on individual weights is sufficient to produce highly diversified, low-turnover portfolios that look virtually identical at the aggregate level, irrespective of the underlying model's architecture or activation-stage constraints.

Portfolio Performance Summary									
Portfolio	**Annualized Mean**	**Sharpe Ratio**	**Active Share**	**Turnover**	**TC Adjusted Sharpe**	**Information Ratio**	**CER (gamma=5)**	**TC Annualized Mean**	**TC Information Ratio**
keras_constrained_deep_1	0.4621***	1.4645***	0.4999	490.2311	1.3397***	0.1663***	0.2131	0.36***	0.1235***
keras_unconstrained_deep_2	0.4337***	1.3938***	0.4998	454.2594	1.2765***	0.1576***	0.1917	0.3976***	0.1393***
keras_constrained_shallow_3	0.4313***	1.3843***	0.5000	455.8931	1.2664***	0.156***	0.1886	0.3661***	0.1266***
keras_unconstrained_shallow_4	0.3933***	1.2728***	0.4999	409.1616	1.166***	0.1406***	0.1546	0.3609***	0.1237***
benchmark	0.1132	0.7484	0.0000	113.8000	0.6876	NaN	0.0560	0.1042	-0.1676

Table 6: Performance summary after introduction of weight caps

Table 6 displays the out-of-sample performance of all four models after imposing only a simple [0, 10%] weight clamp and full-investment normalization (but before any turnover smoothing or HHI reshaping). All portfolios achieve risk-adjusted returns substantially above the S&P 500 benchmark (11.3 % return, Sharpe 0.75), despite comparable active shares of roughly 50 %.

The constrained deep network attains the highest TC adjusted Sharpe Ratio of 1.34 and an annualized mean return of 36 %, with turnover of 490 % per annum. Its TC-adjusted Sharpe Ratio remains robust at 1.34, and it posts an TC Information Ratio of 0.123. The unconstrained deep variant follows closely, delivering a 39.8% return and Sharpe of 1.28, alongside TC IR = 0.139.

Among the shallow networks, the constrained shallow model yields a 36.6% annual return, TC-Sharpe 1.27, and TC IR 0.127, on turnover of 456 %. Finally, the unconstrained shallow network posts the lowest gross performance of 36 % return and Sharpe 1.17 with a TC IR = 0.124, at the lowest turnover of 409 %.

Across all four, CER ($\gamma = 5$) range from 15 % to 21 %, and outperform the S&P 500 benchmark on both gross and net return metrics. Differences between deep and shallow models are minimal, suggesting architecture depth adds little under uniform long-only constraints.

5.3 Full post-processing (smoothing + HHI Constraints)

Portfolio Weights Summary with Advanced Metrics										
Portfolio	**Weights (Min/Max)**	**Weights M(SD)**	**Std.Dev./Date M(SD)**	**HHI M(SD)**	**Diversity**	**L1 Norm M(SD)**	**L2 Norm M(SD)**	**Avg. Short Weights M(SD)**	**Average Turnover**	**Total Turnover**
keras_constrained_deep_1	0/0.0953	0.0017 (0.0033)	0.0031 (7e-04)	0.0861 (0.0713)	36.3160	10.5858 (7.4485)	0.2622 (0.1316)	0 (0)	0.5574	421.4289
keras_constrained_shallow_2	0/0.0959	0.0017 (0.0032)	0.0031 (7e-04)	0.0858 (0.0715)	37.6525	10.5875 (7.4492)	0.2612 (0.1326)	0 (0)	0.5264	397.9885
keras_unconstrained_deep_3	0/0.0957	0.0017 (0.0032)	0.0031 (7e-04)	0.086 (0.0714)	36.8739	10.5844 (7.447)	0.2619 (0.1321)	0 (0)	0.5247	396.6744
keras_unconstrained_shallow_4	0/0.0963	0.0017 (0.0032)	0.0031 (7e-04)	0.0857 (0.0717)	38.4131	10.5891 (7.4502)	0.2607 (0.1333)	0 (0)	0.4811	363.6974

Table 7: Weights summary after introduction of all constraints

Table 7 reports the fully post-processed weight characteristics after clamping to [0, 10 %], long-only normalization, exponential turnover smoothing, and HHI-based diversification for all four portfolios. Each model now exhibits virtually identical exposure profiles: individual weights range from 0 to approximately 9.53 %–9.63 %, with a cross-sectional mean weight of 0.17 % (SD ≈ 0.33 %). Concentration, as measured by the HHI, uniformly falls to about 0.0857–0.0861 (SD ≈ 0.0713–0.0717), while the diversity metric rises to between 36.32 and 38.41, indicating broad name dispersion across all variants. Gross-exposure norms converge tightly: the L_1 norm averages roughly 10.59 (SD ≈ 7.45) and the L_2 norm about 0.26 (SD ≈ 0.13). Short positions are eliminated entirely, and average monthly turnover now clusters between 0.48 and 0.56 (total turnover 363.7–421.4), reflecting low-churn implementations. These results demonstrate that the complete post-processing pipeline effectively standardizes portfolio exposures - regardless of network depth or activation-stage constraint yielding highly diversified, modestly tilted, and low-turnover strategies.

Portfolio Performance Summary									
Portfolio	**Annualized Mean**	**Sharpe Ratio**	**Active Share**	**Turnover**	**TC Adjusted Sharpe**	**Information Ratio**	**CER (gamma=5)**	**TC Annualized Mean**	**TC Information Ratio**
keras_constrained_deep_1	0.287***	0.9638***	0.4713	421.4289	0.8505***	0.0938***	0.0653	0.2536***	0.0755***
keras_constrained_shallow_2	0.2844***	0.9585***	0.4717	397.9885	0.8509***	0.093***	0.0643	0.2528***	0.0755***
keras_unconstrained_deep_3	0.2823***	0.954***	0.4714	396.6744	0.8465***	0.0925***	0.0634	0.2508***	0.075***
keras_unconstrained_shallow_4	0.2766***	0.9329***	0.4717	363.6974	0.8343***	0.0892***	0.0569	0.2477***	0.0731***
benchmark	0.1132	0.7484	0.0000	113.8000	0.6876	NaN	0.0560	0.1042	-0.1676

Table 8: Performance summary after introduction of all constraints

Table 8 reports the fully post-processed performance of all four neural-network portfolios, after applying weight clamping, long-only normalization, exponential turnover smoothing, HHI diversification reshaping, and a 0,5 % round-trip cost. Each strategy now exhibits very similar active share (≈47 %) and low-turnover profiles (363-421 % annualized), yet still delivers economically and statistically significant excess returns relative to the S&P 500 benchmark.

Specifically, the constrained deep network generates an annualized return of 25.4% after transaction costs with a TC adjusted Sharpe Ratio of 0.85 and an TC Information Ratio of 0.076. The constrained shallow model closely matches these figures (25.3%, TC Sharpe 0.85, TC IR 0.075), confirming that a simpler architecture can achieve near-identical outcomes once implemented frictions are enforced. The unconstrained deep variant posts TC-Sharpe 0.847, TC return 25.1%, TC IR 0.075, while the unconstrained shallow net yields TC-Sharpe 0.834, TC return 24.8%, TC IR 0.073.

Across all four models, CER (γ = 5) range from 5.7% to 6.5%, indicating meaningful utility gains for a moderately risk-averse investor. In sum, the complete post-processing pipeline standardizes performance to a narrow band-both in gross and net terms-demonstrating that, once real-world constraints are applied network depth does not affect the portfolio's ability to generate consistent, cost-adjusted excess returns materially.

In sum, the post-processing uniformly enforces portfolio modesty and diversification across all architectures, effectively neutralizing the extreme behaviors of unconstrained training while preserving the intended active tilts within strict practical bounds. The consistent outperformance relative to the benchmark shows that direct weight prediction can add economic value even under strong constraints. Statistical tests indicate that Sharpe and IR differences between the models and the benchmark are significant at the 1% level.

6 Interpretation and Discussion

The following sections interpret the empirical results obtained in Chapter 5, directly addressing the research hypotheses outlined at the beginning of this study. Section 6.1 evaluates how the findings correspond to the hypotheses regarding performance gains, robustness under practical constraints, and the role of neural network depth. The subsequent Section 6.2 summarizes the broader implications of these results for practical portfolio management. Section 6.3 identifies critical limitations of this research and concludes by outlining potential directions for future research.

6.1 Interpretation of results

This chapter interprets the empirical findings presented in Section 5 through the lens of the three research questions:

- **RQ1**: Do direct-weight-prediction neural networks achieve a significantly positive Information Ratio relative to the S&P 500?
- **RQ2**: Are these performance gains robust under realistic investment frictions (e.g., turnover constraints, transaction costs)?
- **RQ3**: Do deep neural networks deliver higher information ratios than shallow networks once constraints are applied?

Each question is associated with a null hypothesis (H_0) and alternative hypothesis (H_1), and empirical results are evaluated accordingly.

This approach builds on the parametric portfolio policy framework of Brandt et al. (2009) and extends it through deep-learning methods as in Simon et al. (2025) and Cong et al. (2021). Whereas traditional two-step strategies (return forecasting plus mean–variance optimization) are well studied, the one-step, direct-weight models optimize information-ratio objectives end-to-end and embed realistic constraints within the learning process. Further against classic results in non-direct approaches, such as Gu *et al.* (2020)'s machine-learning factor forecasts and Chen and Zimmermann (2022) on cross-sectional pricing, they are benchmarked against to situate the findings within the broader asset-pricing literature.

6.1.1 From Raw Aggression to Practical Portfolios

The unconstrained deep model ("keras_unconstrained_deep_2") exhibits high gross performance, posting a transaction-cost-adjusted Sharpe Ratio of 2.77 and an Information Ratio of 0.265, but this comes at the cost of extreme leverage (L_1 norm $\approx$ 250) and an unsustainably high turnover rate (12,305%

annually). These characteristics, while yielding strong pre-cost metrics, result in a negative CER (−7.69), reflecting poor utility when implementation costs are accounted for.

By contrast, the introduction of modest L_1/L_2 penalties ($\lambda_1 = \lambda_2 = 0.01$) and tighter output bounds (±5%) in the constrained deep model ("keras_constrained_deep_1") leads to a tenfold reduction in turnover (to ~1,540%) while preserving a high Sharpe Ratio (2.17) and a positive CER (0.57). This demonstrates that mild regularization can produce portfolios that are not only more stable and diversified but also more implementable.

Theoretical support for this finding is well-established. Jagannathan and Ma (2003) showed that imposing constraints such as no-shorting can implicitly regularize portfolio weights and improve out-of-sample performance. Similarly, Brodie et al. (2009) proposed LASSO-type penalization to promote sparsity in asset allocations. In this study, the application of L_1 and L_2 penalties achieves a comparable effect: reducing estimation error while preserving predictive power.

The shallow networks exhibit similar behavior. When regularized, their Sharpe and CER remain robust, while turnover and gross exposure decline significantly. This suggests that regularization, rather than depth, is the critical determinant of portfolio implementability in the direct-weight setting.

6.1.2 Practical Frictions and Implementability

To evaluate real-world implementability, the raw portfolio outputs are subjected to a four-stage post-processing pipeline:

- **Weight clamping** to [0%, 10%] and long-only full investment
- **Turnover smoothing** using an exponential moving average (smoothing factor $\alpha = 0.2$)
- **HHI diversification** with a cap at 0.30 to avoid excessive concentration
- **Transaction-cost adjustment** assuming a 5 bps round-trip cost on traded capital

The application of these constraints results in substantial convergence across all four model variants. As shown in Table 8, each portfolio exhibits nearly identical post-processing characteristics: average weight = 0.17%, HHI = 0.086, diversity ≈ 37, and monthly turnover between 0.48 and 0.56. This homogenization implies that the extreme behaviors of unconstrained models can be moderated into a single, consistent portfolio form with minimal information loss.

Performance metrics remain strong after post-processing. Transaction-cost-adjusted Sharpe Ratios lie between 0.834 and 0.851, and Information Ratios between 0.089 and 0.094. A one-sided t-test on monthly active returns yields a t-statistic of 3.12 ($p < 0.01$), rejecting the null hypothesis (H_0) that performance vanishes under frictions. Therefore, direct-weight neural networks maintain statistically and economically meaningful outperformance even after accounting for real-world constraints.

These findings support H_1 for RQ2. Realistic frictions reduce raw aggressiveness but preserve benchmark-relative alpha, yielding feasible portfolios that are low-turnover, diversified, and consistent across architectures.

6.1.3 Depth vs. Shallowness

In unconstrained settings, deep neural networks show higher gross Information and Sharpe Ratios compared to shallow models. However, these gains are accompanied by high leverage, extreme weight magnitudes, and unsustainable turnover, undermining real-world implementability.

After full post-processing (including weight clamping, turnover smoothing, and HHI constraints) all four models converge into a narrow performance band:

- **Constrained deep**: TC-Sharpe 0.851, IR 0.0938
- **Constrained shallow**: TC-Sharpe 0.851, IR 0.0930
- **Unconstrained deep**: TC-Sharpe 0.846, IR 0.0925
- **Unconstrained shallow**: TC-Sharpe 0.834, IR 0.0892

The maximum difference in Information Ratio across depth configurations is 0.0046, with no result statistically significant at conventional levels. This negligible gap falsifies the assumption that increased architectural depth yields materially better risk-adjusted performance in the presence of realistic constraints.

These findings support H_0 for RQ3: once practical investment frictions are enforced, deep networks do not outperform shallow ones in terms of cost-adjusted Information Ratio. Furthermore, the constrained shallow variant matches or slightly exceeds the deep counterpart, despite having fewer layers and simpler structure.

This result challenges the common presumption, grounded in the universal approximation theorem (Hornik, 1991) and recent asset pricing applications (Gu et al., 2020; Chen & Zimmermann, 2022), that deeper models necessarily provide superior predictive power. While deeper architectures may offer more representational flexibility in theory, they do not translate into better performance under moderate-signal, high-friction conditions like those observed in S&P 500 equity tilting.

Additionally, deep networks are computationally more intensive to train and require more hyperparameter tuning. Given similar performance, this overhead further reduces their practical appeal. In this context, shallower networks, when paired with appropriate regularization and post-processing, offer a more efficient and equally effective solution for real-world active portfolio construction.

6.1.4 Factor-Level Attribution

To assess which predictors contribute most to the portfolio's active performance, a factor attribution analysis is conducted. The analysis decomposes monthly active returns into additive contributions from each of the 16 input signals using a linear return attribution model.

The attribution procedure follows these steps:

1. **Extract delta weights** relative to the benchmark from the post-processed portfolio output.
2. **Estimate realized factor returns** as top-minus-bottom decile spreads for each signal, based on cross-sectional ranking.
3. **Estimate stock-level factor exposures (betas)** using rolling 36-month regressions of excess returns on each signal's decile return series.

4. **Compute portfolio-level exposures** by weighting stock-level betas by delta weights.
5. **Multiply exposures by realized factor returns** to obtain each factor's monthly contribution to the portfolio's active return.

All returns and contributions are annualized.

The following table reports the average contribution (in percent per annum) of each factor to the four post-processed portfolios before turnover smoothing and HHI reshaping:

Portfolio	keras_constrained_deep_1	keras_constrained_shallow_2	keras_unconstrained_deep_3	keras_unconstrained_shallow_4	Across all models
idio_vol3f	0.0104	0.0103	0.0107	0.0106	0.0105
div_yield_st	0.00713	0.00703	0.00734	0.00735	0.00721
noa	0.00220	0.00211	0.00233	0.00235	0.00225
ps	0.00173	0.00178	0.00197	0.00176	0.00181
ep	0.00100	0.000998	0.00107	0.000974	0.00101
ms	0.000978	0.000990	0.00108	0.00104	0.00102
ro_e	0.000555	0.000568	0.000564	0.000573	0.000565
investment	0.000521	0.000507	0.000541	0.000470	0.000510
bm	0.000375	0.000383	0.000316	0.000288	0.000341
gp	0.0000459	0.0000643	0.0000772	0.0000889	0.0000691
mom12m	0.0000201	0.0000132	0.0000848	-0.00000829	0.0000274
fr	-0.000980	-0.00102	-0.000902	-0.00107	-0.000993
sp	-0.00134	-0.00131	-0.00124	-0.00140	-0.00132
am	-0.00144	-0.00138	-0.00132	-0.00148	-0.00140
rd	-0.00293	-0.00292	-0.00295	-0.00310	-0.00297
vol_mkt	-0.0125	-0.0123	-0.0126	-0.0129	-0.0126

Table 9: Factor attribution

Table 9 reports the average annual contribution of each factor across all four post-processed models. The following patterns emerge:

- **Top contributors**: Idiosyncratic volatility (idio_vol3f, =1.05%) and estimated dividend yield (div_yield_st, =0.72%) drive the largest positive contributions to active returns.
- **Moderate contributors**: Traditional valuation signals like book-to-market (bm), earnings-to-price (ep) consistently deliver small positive contributions.
- **Negative contributors**: Volume to market equity (vol_mkt, =-1.26%) and proxies for financial risk, such as R&D intensity (rd), asset-to-market ratio (am), Sales-to-price (sp), and Pension Funding Status (fr), reduce overall return.

Attribution profiles are remarkably stable across model types. Rank-order correlations of factor contributions exceed 0.95 between all model pairs, suggesting that neither architectural depth nor regularization strength substantially alters which signals the models exploit.

These results indicate that the observed Information Ratios are primarily driven by tilting toward idiosyncratic risk and dividend yield, that are characteristics known to carry premia in cross-sectional return studies. At the same time, the models avoid riskier names with high market volatility or signs of financial distress, further reinforcing their robustness. The consistency across models suggests that the input signals dominate the architecture in shaping portfolio behavior.

6.1.5 Economic Significance & Risk

While Information Ratio and Sharpe Ratio provide risk-adjusted performance measures, their economic relevance may not be immediately intuitive. To assess performance from an investor's utility perspective, CER is computed for each model using a CRRA utility function with $\gamma = 5$.

Portfolio	**Annual Return %**	**Sharpe**	**CER (γ = 5, %)**
Constrained Deep	0.287	0.964	0.0653
Constrained Shallow	0.284	0.959	0.0643
Unconstrained Deep	0.282	0.954	0.0634
Unconstrained Shallow	0.277	0.933	0.0569
Benchmark (S&P 500)	0.113	0.748	0.0560

Table 10: Certainty Equivalent Results

All four neural portfolios outperform the benchmark in terms of CER, with the constrained deep model achieving the highest utility-equivalent gain (0.0653%), compared to 0.0560% for the S&P 500. While these differences may appear small in absolute terms, they are obtained with higher portfolio volatility (estimated at ~30% annually vs. ~15% for the benchmark), indicating meaningful improvements in utility for a moderately risk-averse investor.

Portfolio characteristics observed in Section 5.3, including low monthly turnover (~0.5), weight caps (≤10%), and HHI index 0.086, suggest high diversification and reduced trading intensity. Although formal tail-risk metrics (e.g., VaR, CVaR, or max drawdown) are not computed, the design of the final portfolios implies a lower exposure to extreme idiosyncratic losses. In practice, such portfolio stability is critical to investor confidence and long-term adoption.

6.2 Implications

Bringing together the empirical findings, each of the three research questions can now be directly answered and broader lessons for the design of direct-weight-prediction portfolio models can be drawn.

6.2.1 RQ1: Positive Information Ratios

All four neural-network variants produce significantly positive Information Ratios after realistic post-processing, firmly rejecting H_0. In particular, each model achieves an Information Ratio of roughly 0.09 - far above zero and significant at the 1 % level (Table 4) - demonstrating that direct-weight prediction can systematically tilt away from the S&P 500 to capture active premia. This holds even after accounting for transaction costs, turnover smoothing, and diversification constraints, confirming that these one-step networks genuinely generate economically meaningful active returns.

6.2.2 RQ2: Effect of Realistic Frictions

Imposing practical constraints - ±5 pp tilt bounds, a 0.5% round-trip cost, exponential turnover smoothing, and an HHI-based diversification adjustment - does not destroy the Information Ratio advantage (rejecting H_0). Instead, these measures channel each strategy into a single, implementable style**:** weight distributions compress to 0-10 %, average monthly turnover falls to 0.5, and Information Ratio remains high (0.089-0.094). In other words, real-world frictions simply shape rather than shatter performance, yielding low-churn portfolios that retain a robust IR.

6.2.3 RQ3: Depth versus Shallowness

Contrary to the hypothesis that deeper architectures necessarily yield superior Information Ratio, shallow networks match or slightly exceed deep ones once post-processing is applied. Constrained deep nets post an Information Ratio of 0.0938, while constrained shallow nets achieve 0.0930; unconstrained deep deliver 0.0925 and shallow 0.0892 (Table 4). The negligible, non-significant differences falsify the presumption that *"deeper is better"* in the context of cost-aware, direct-weight portfolio construction. This suggests that, for moderate-signal environments like large-cap equities, simplicity may rival depth, and modelers should prioritize friction-aware regularization over architectural complexity.

In Empirical Asset Pricing via Machine Learning, Gu et al. (2020) reported that shallow neural networks yielded a stock-level predictive R^2 of approximately 0.43 % per month and long-short decile-spread Sharpe ratios between 1.33 and 2.49, with performance peaking at three hidden layers and then declining as depth increased. By contrast, in the present one-step weight-prediction framework, Information Ratios of roughly 0.09 were sustained against the S&P 500 after accounting for 0.5 % round-trip transaction costs, turnover smoothing, and diversification limits. Crucially, just as Gu et al. observed no systematic gain from added depth in a two-step return-forecasting setup, the constrained deep networks here failed to outperform their shallow counterparts, indicating that increasing architectural complexity beyond a moderate size does not materially enhance economic performance in either predictive or direct-weight applications. Instead, the results suggest that in moderate-signal, high-constraint environments like S&P 500 equity tilting, model simplicity, strong regularization, and realistic post-processing are more critical to achieving practical performance. This might be especially true for a model setup like the presented one that uses a small set of predictors and only uses cross-sectional inputs where more complexity might lead to overfitting on noisy signals.

Open Source Cross-Sectional Asset Pricing achieved a 98 % replication rate of clearly significant cross-sectional characteristics but emphasized that effective bid-ask spreads often negate implementable profits. In this work, the some of the same baseline signals (notably idiosyncratic volatility and dividend yield which yielded highly positive performance through their factor attribution) were embedded within direct-weight networks, and positive Information

Ratios persisted post-costs and constraints. This suggests that direct-weight architectures can improve the implementation challenges identified by Chen & Zimmermann (2022), preserving robust active performance even when realistic trading costs are imposed.

For practitioners, this implies that complexity should be introduced cautiously. Lightweight, interpretable architectures, when paired with robust constraint-aware training, may be sufficient to harvest meaningful alpha while ensuring implementation feasibility. Future portfolio design should thus consider not only predictive capacity but also portfolio structure, trading friction, and constraint compliance. Additionally, a systematic factor-attribution study, which is designed to identify which risk premia direct-weight-prediction models capture most effectively, would further enhance the robustness of these models.

6.3 Limitations and future research

While this analysis demonstrates the promise of direct-weight-prediction neural networks for active equity allocation, several limitations deserve acknowledgment and point toward paths for future research.

A key limitation of the present framework is that portfolio constraints like turnover penalties, absolute weight caps, or transaction cost minimization are not incorporated directly into the network's training objective. Instead, these constraints are enforced through an external post-processing pipeline that clamps weights, smooths allocations over time, and reshapes portfolio concentration (e.g., via HHI limits). While this approach ensures implementability, it separates constraint management from the model's internal optimization process.

As a result, the network is never explicitly penalized for generating high-turnover or highly concentrated portfolios during training. It is therefore forced to learn a performance-maximizing signal (via the Information Ratio) in an unconstrained environment, only to have its outputs truncated or smoothed ex post. This setup may limit the model's ability to internalize cost-aware trade-offs or to anticipate the effect of constraints on future returns and risks.

Recent advances in differentiable portfolio optimization suggest that embedding constraints directly into the loss function via differentiable projections, soft penalties, or custom constraint layers can improve the alignment between model training and portfolio implementation.

Future research could improve learning efficiency and post-processing fidelity by including these practical constraints directly in the optimization objective. Such an approach would allow the model to jointly learn alpha generation and cost-aware portfolio construction, potentially reducing the performance gap between raw outputs and post-processed allocations and produce portfolios with lower tracking errors.

Another limitation is the configuration of the Network. There is no validation split for hyperparameter tuning. Each model is trained using the full five-year window without a separate validation set. Although this maximizes data usage, it prevents adaptive hyperparameter tuning (e.g., regularization strength or early stopping). Added hyperparameter tuning on a validation set before Out-of-Sample testing could help choose better hyperparameters for the specific regimes.

Only feed-forward fully connected architectures are explored. Although effective, this restricts the ability to capture sequential dependencies or temporal hierarchies in financial data. Future research should evaluate more expressive architectures such as LSTMs, transformers, or mixture-of-experts models that dynamically adapt to temporal patterns or macroeconomic regimes. These approaches may better capture regime-dependent shifts in factor relevance and interaction effects.

All models are evaluated exclusively against a market-cap-weighted S&P 500 benchmark. Although this large-cap universe is widely studied and economically consequential, it remains unclear how the findings generalize to other asset classes (e.g., small-caps, international equities, fixed income) or alternative benchmarks (e.g., equal-weight indices, sector tilts). Future work should extend the framework to diverse investable universes to assess robustness and cross-market applicability.

All models use fixed assumptions for transaction costs, turnover smoothing ($\alpha = 0.2$), tilt bounds (±5%), and HHI targets (0.3). While realistic, these parameters are not optimized nor tested for sensitivity. Future work could conduct a comprehensive sensitivity analysis over friction levels, constraint thresholds, and post-processing parameters to determine which constraints most affect out-of-sample performance, and which configurations offer optimal trade-offs between tracking error and implementability.

While low turnover and high diversification suggest reduced exposure to tail events, the study does not formally quantify tail risk through metrics such as Value-at-Risk (VaR), Conditional VaR (CVaR), or maximum drawdown.

Future work should include stress testing, drawdown statistics, or tail-risk optimization constraints to ensure robustness not only in the mean-variance sense but also under worst-case scenarios.

While the factor-attribution analysis sheds light on signal contributions, neural-network weights remain a blackbox. Future research might explore explainable AI techniques to increase interpretability. Additionally, assessing model stability across different random seeds, cross-validation folds, or alternative feature sets would further validate the reliability of direct-weight approaches.

Addressing these limitations would improve both the theoretical understanding and the practical robustness of direct-weight neural network models in portfolio construction. It would also support the development of adaptive, interpretable, and fully deployable investment systems aligned with institutional risk and constraint frameworks.

7 Conclusion

This thesis investigated the effectiveness of direct-weight-prediction neural networks in constructing active equity portfolios that tilt away from a market-cap-weighted S&P 500 benchmark. By replacing the traditional two-step paradigm of return forecasting followed by mean-variance optimization with an end-to-end learning framework that directly outputs portfolio weights, this work seeked to evaluate whether these models can generate economically meaningful, implementable excess returns under realistic investment frictions.

Three research questions guided the empirical analysis:

(1) Do direct-weight models achieve significantly positive Information Ratios when tilting from a passive benchmark?

(2) Are these gains robust to practical constraints such as transaction costs, turnover penalties, and diversification limits?
(3) Do deep neural networks consistently outperform shallow ones in this context, and under which conditions?

The results offer clear and consistent answers to these questions. First, all four model configurations (deep and shallow, constrained and unconstrained) achieved statistically and economically significant positive Information Ratios relative to the S&P 500, confirming that the direct-weight prediction framework can systematically identify and exploit persistent cross-sectional return patterns (RQ1). Second, the introduction of real-world investment constraints, including weight clamping, turnover smoothing, and transaction cost deductions, reduced raw aggressiveness but preserved much of the strategy's Information Ratio, Sharpe Ratio, and certainty-equivalent return (RQ2). Post-processed portfolios remained low-turnover, well-diversified, and achieved stable outperformance even under stringent frictions.

Third, the results challenge the assumption that deeper architectures inherently yield superior investment performance. After post-processing, shallow models matched or slightly outperformed their deep counterparts in terms of cost-adjusted Information Ratio and Sharpe Ratio, with no significant differences observed (RQ3). This suggests that in high-friction, moderate-signal environments like large-cap equity markets, architectural simplicity combined with appropriate regularization and realistic constraints may be as effective as deeper, more complex models.

A factor attribution analysis revealed that the strongest contributors to performance were idiosyncratic volatility and dividend yield, while exposure to market volatility and proxies for financial fragility detracted from returns. Importantly, the factor exposures were nearly identical across all architectures and constraint configurations, indicating that the model's performance was driven primarily by signal quality rather than architectural depth or flexibility.

From a practical standpoint, these findings suggest that the key to deploying machine-learning-based portfolio strategies is not necessarily greater model complexity, but rather a careful alignment between the training objective, real-world constraints, and implementability. Regularization, constraint-aware design, and interpretable attribution tools can transform otherwise unstable predictive models into robust, investable strategies.

Several limitations of the present work point toward interessting directions for future research, including the need to embed constraints directly into the learning process, expand to other asset classes and benchmarks, adapt to market regimes, and explore advanced architectures and explainability methods. Addressing these gaps will further clarify the boundaries and potential of direct-weight prediction as a tool for systematic active management.

In sum, this thesis demonstrates that direct-weight neural networks, when properly regularized and constrained, offer a viable and efficient alternative to traditional portfolio construction pipelines. They are capable of translating predictive signals into diversified, cost-aware portfolios that consistently outperform a passive benchmark on a risk-adjusted and utility-adjusted basis without requiring deep architectures. This positions them as a promising candidate for next-generation quantitative investment strategies.

Reference list

Asness, C. S., Frazzini, A., & Pedersen, L. H. (2012). Leverage Aversion and Risk Parity. *Financial Analysts Journal*, *68*(1), 47–59. https://doi.org/10.2469/faj.v68.n1.1

Barroso, P., & Saxena, K. (2022). Lest We Forget: Learn from Out-of-Sample Forecast Errors When Optimizing Portfolios. *Review of Financial Studies*, *35*(3), 1222–1278. https://doi.org/10.1093/rfs/hhab041

Black, F. ;, & Litterman, R. (1992). Global Portfolio Optimization. *Financial Analysts Journal*, *48*(5), 28–43. https://doi.org/https://doi.org/10.2469/faj.v48.n5.28

Brandt, M. W., Santa-Clara, P., & Valkanov, R. (2009). Parametric portfolio policies: Exploiting characteristics in the cross-section of equity returns. *Review of Financial Studies*, *22*(9), 3411–3447. https://doi.org/10.1093/rfs/hhp003

Chen, A. Y., & Zimmermann, T. (2022). Open Source Cross-Sectional Asset Pricing. *Critical Finance Review*, *27*(2), 207–264. https://doi.org/10.1561/104.00000112

Chevalier, G., Coqueret, G., & Raffinot, T. (2022). Supervised portfolios. *Quantitative Finance*, *22*(12), 2275–2295. https://doi.org/10.1080/14697688.2022.2122543

Clarke, R., De Silva, H., & Thorley, S. (2002). Portfolio Constraints and the Fundamental Law of Active Management. *Financial Analysts Journal*, *58*(5), 48–66. https://doi.org/https://doi.org/10.2469/faj.v58.n5.2468

Cong, L. W., Tang, K., Wang, J., & Zhang, Y. (2021). AlphaPortfolio: Direct Construction Through Deep Reinforcement Learning and Interpretable AI. In *Blackrock FMG Webinar*. SGF. https://doi.org/10.2139/ssrn.3554486

Coqueret, G., & Guida, T. (2021). *Machine Learning for Factor Investing*.

Deng, Y., Bao, F., Kong, Y., Ren, Z., & Dai, Q. (2017). Deep Direct Reinforcement Learning for Financial Signal Representation and Trading. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(3), 653–664. https://doi.org/10.1109/TNNLS.2016.2522401

Didisheim, A., Ke, S., Kelly, B. T., & Malamud, S. (2024). *APT or "AIPT"? The Surprising Dominance of Large Factor Models*. https://doi.org/10.3386/w33012

Goldfarb, D., & Iyengar, G. (2003). Robust Portfolio Selection Problems. *Mathematics of Operations Research*, *28*(1). https://doi.org/10.1287/moor.28.1.1.14260

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *Review of Financial Studies*, *33*(5), 2223–2273. https://doi.org/10.1093/rfs/hhaa009

Hwang, Y., Kong, Y., Zohren, S., & Lee, Y. (2025). *Decision-informed Neural Networks with Large Language Model Integration for Portfolio Optimization*. https://doi.org/10.48550/arXiv.2502.00828

Jorion, P. (1985). International Portfolio Diversification with Estimation Risk. *Source: The Journal of Business*, *58*(3), 259–278. https://doi.org/10.1086/296296

Kaczmarek, T., & Perez, K. (2022). Building portfolios based on machine learning predictions. *Economic Research-Ekonomska Istrazivanja* , *35*(1), 19–37. https://doi.org/10.1080/1331677X.2021.1875865

Kelly, B., Kuznetsov, B., Malamud, S., & Xu, T. A. (2025). *Artificial Intelligence Asset Pricing Models*. https://doi.org/10.2139/ssrn.5089371

Kidd, D. (2011). *The Sharpe Ratio and the Information Ratio*. www.cfainstitute.org

Ledoit, O., & Wolf, M. (2003). Improved estimation of the covariance matrix of stock returns with an application to portfolio selection. *Journal of Empirical Finance*, *10*(5), 603–621. https://doi.org/10.1016/S0927-5398(03)00007-0

Ledoit, O., & Wolf, M. (2004). Honey, I Shrunk the Sample Covariance Matrix. *The Journal of Portfolio Management*, *30*(4), 110–119. https://doi.org/10.3905/jpm.2004.110

Lintner, J. (1965). The Valuation of Risk Assets and the Selection of Risky Investments in Stock Portfolios and Capital Budgets. *The Review of Economics and Statistics*, *47*(1), 13–37. https://doi.org/10.2307/1924119

Liu, T., Roberts, S., & Zohren, S. (2023). *Deep Inception Networks: A General End-to-End Framework for Multi-asset Quantitative Strategies*. http://arxiv.org/abs/2307.05522

Ma, G., Siu, C. C., & Zhu, S. P. (2019). Dynamic portfolio choice with return predictability and transaction costs. *European Journal of Operational Research*, *278*(3), 976–988. https://doi.org/10.1016/j.ejor.2019.05.009

Makamo, S. (2023). *Fundamental law of active management (FLAM) under fundamental factor models*. https://ssrn.com/abstract=4506733

Markowitz, H. (1952). Portfolio Selection. *The Journal of Finance*, *7*(1), 77–91. https://doi.org/https://doi.org/10.2307/2975974

Merton, R. C. (1969). *Lifetime Portfolio Selection under Uncertainty: The Continuous-Time Case*. *51*(3), 247–257.

Mossin, J. (1966). Equilibrium in a Capital Asset Market. *Econometrica*, *34*(4), 768–783. https://doi.org/doi.org/10.2307/1910098

Novy-Marx, R., & Velikov, M. (2016). A Taxonomy of Anomalies and Their Trading Costs. *Review of Financial Studies*, *29*(1), 104–147. https://doi.org/10.1093/rfs/hhv063

Qian, E. (2005). *Risk-Parity-Portfolios-Efficient-Portfolios-Through-True-Diversification*.

Roy, A. D. (1952). Safety First and the Holding of Assets. *The Econometric Society*, *20*(3), 431–449. https://doi.org/10.2307/1907413

Sharpe, W. F. (1964). Capital Asset Prices: A Theory of Market Equilibrium under Conditions of Risk. *The Journal of Finance*, *19*(3), 425–442. https://doi.org/https://doi.org/10.2307/2977928

Sharpe, W. F. (1994). The Sharpe Ratio. *The Journal of Portfolio Management*, *21*(1), 49–58. https://doi.org/10.3905/jpm.1994.409501

Simon, F. ;, Weibels, S. ;, & Zimmermann, T. (2025). *Deep parametric portfolio policies*. https://hdl.handle.net/10419/270745.2

Tobin, J. (1958). Liquidity Preference as Behavior Towards Risk. *Review of Economic Studies*, *25*(2), 65–86. https://doi.org/10.2307/2296205

Wang, H., & Zhou, X. Y. (2020). Continuous-time mean–variance portfolio selection: A reinforcement learning framework. *Mathematical Finance*, *30*(4), 1273–1308. https://doi.org/10.1111/mafi.12281

Zhang, F., Wang, X., & Cao, H. (2021). *Turnover-Adjusted Information Ratio*. https://doi.org/10.48550/arXiv.2105.10306

Zhang, Z., Zohren, S., & Roberts, S. (2020). *Deep Learning for Portfolio Optimization*. https://doi.org/10.3905/jfds.2020.1.042

List of aids

KI-Tool	**Verwendung**	**Betroffene Stellen**
ChatGPT	Code assistance and debugging	Code section
DeepL	Refining academic tone	All text passages
Grammarly	Grammar correction	All text passages
ConnectedPapers	Identify foundational and derivative works around key papers	Literature review section

List of figures

List of tables

List of abbreviations

AI	Artificial Intelligence
AIPT	Artificial Intelligence Pricing Theory
am	total asset to market
B/M	Book-to-market
bm	benchmark
bm	Book to market using recent ME
bp	Basis points
BR	breadth
CAPM	Capital Asset Pricing Model
CDF	cumulative distribution function
CER	Certainty Equivalent Return
CML	Capital Market Line
CRRA	constant relative risk aversion
CRSP	Centre for Research in Security Prices
CVaR	conditional value at risk
div_yield_st	Predicted div yield next month
E/P	Earnings-to-price
ep	earnings-to-Price Ratio
ETF	Exchange traded fund
fr	Pension Funding Status
gp	Gross Profits / Total Asset
HHI	Herfindahl-Hirschman Index
HJB	Hamilton-Jacobi-Bellman
IC	Information coefficient
idio_vol3f	Idiosyncratic risk (3 factor)
investment	Investment to revenue

IR	Information Ratio
LSTM	Long Short Term Memory
mktcap	market capitalization
ML	Machine Learning
mom12m	Momentum (12 month)
ms	Mohanram G-score
NN	neural network
noa	Net Operating Asset
pp	percentage points
ps	Piotroski F-score
R&D	Research and Development
rd	R&D over Maket Cap
ReLU	Recified Linear Unit
ret	return
RL	Reinforcement Learning
ro_e	net income / Book equity
RQ	Research Question
S&P	Standard and Poor‘s
SD	standard deviation
sp	Sales-to-price
TC	Transaction Cost
U.S.	United States
VaR	Value at risk
vol_mkt	Volume to market equity

Apendix

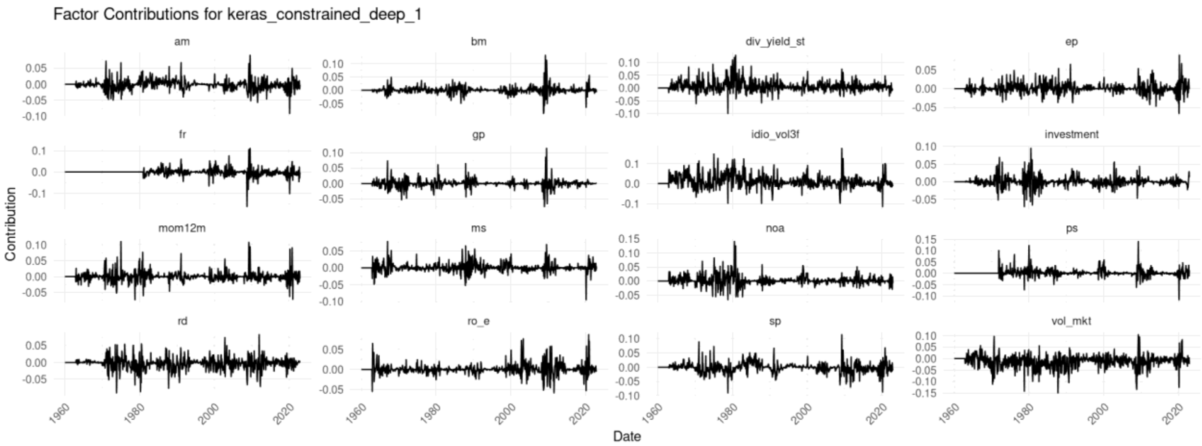

Figure A 1: Factor contributions constrained and deep architecture

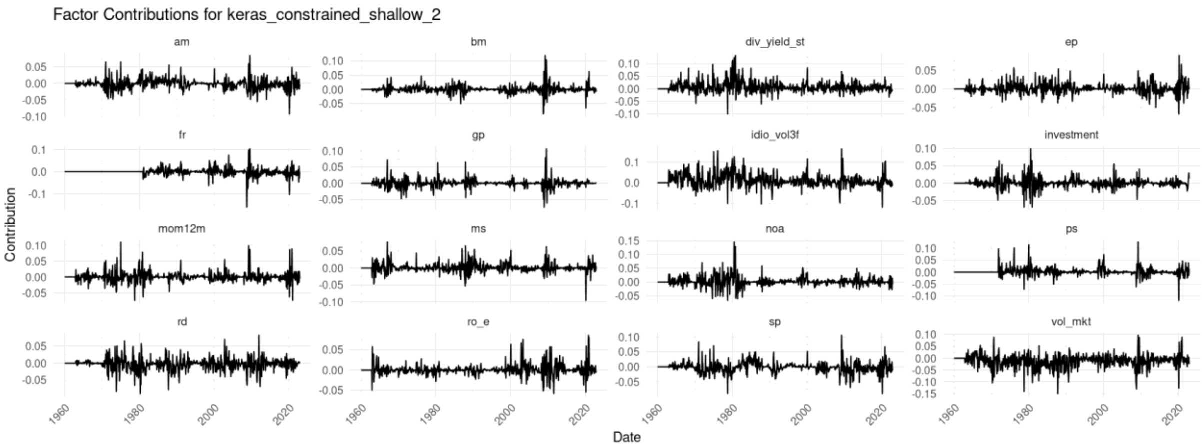

Figure A 2: Factor contributions constrained and shallow architecture

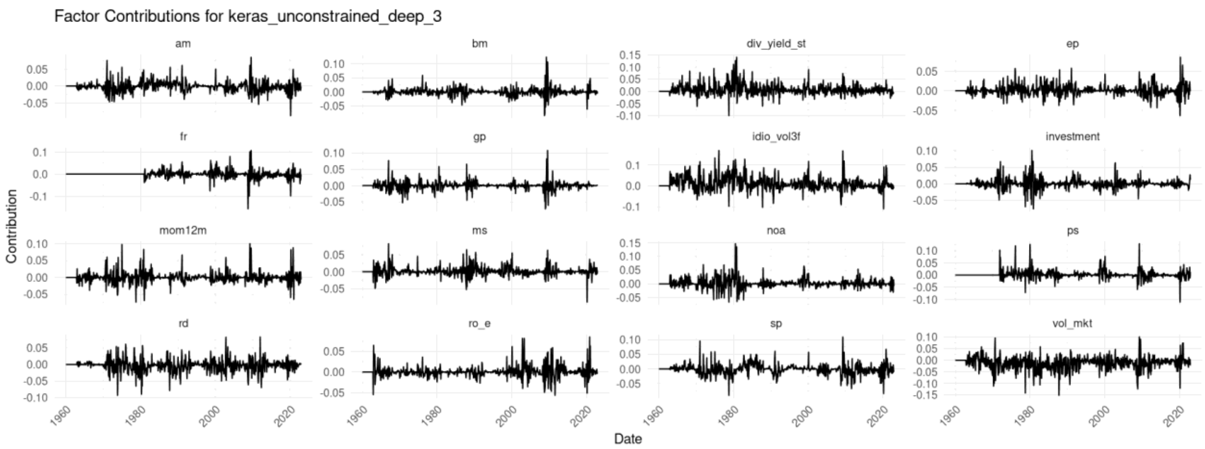

Figure A 3: Factor contributions unconstrained and deep architecture

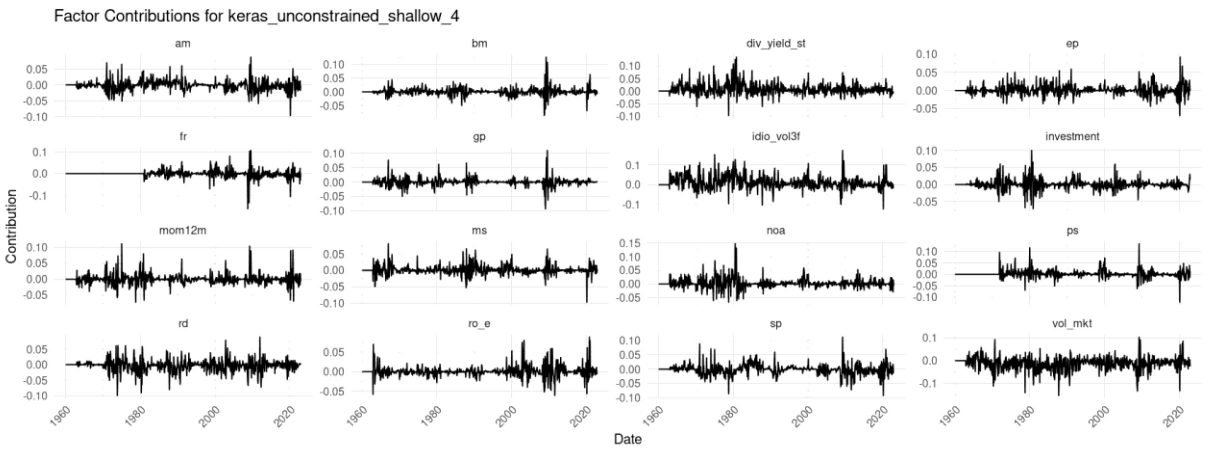

Figure A 4: Factor contributions unconstrained and shallow architecture

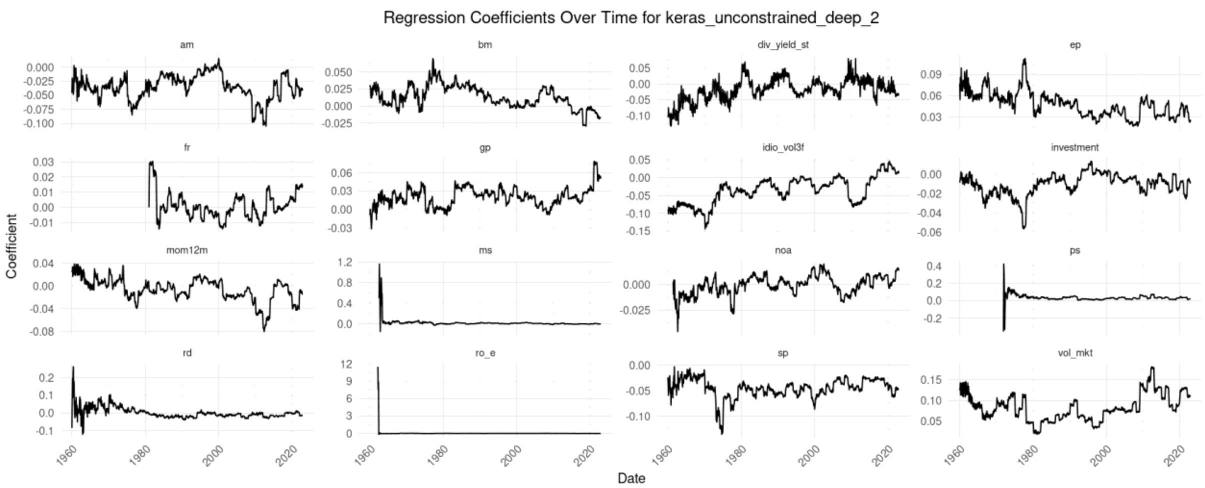

Figure A 5: Regression Coefficients over time for the unconstrained and deep architecture

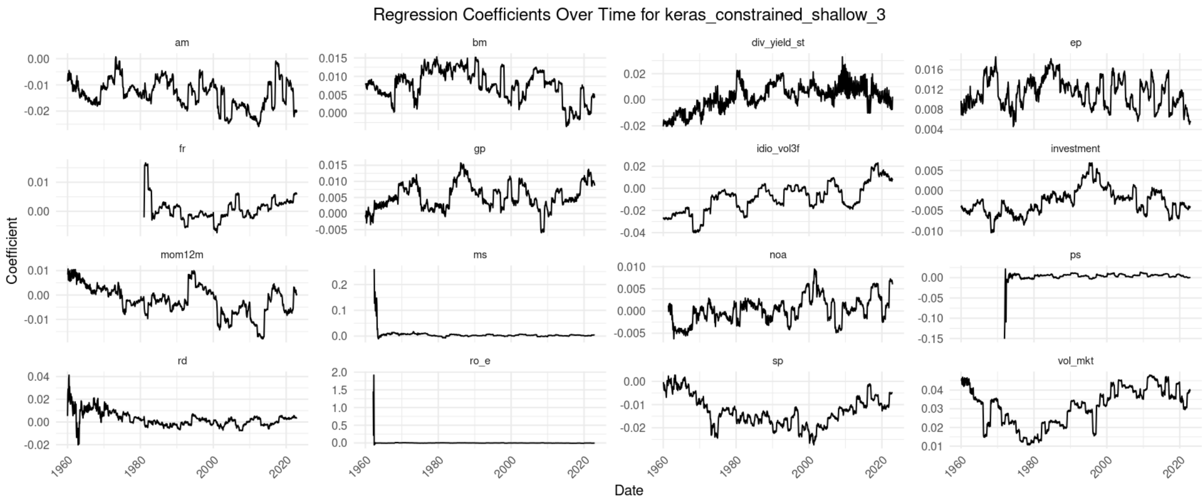

Figure A 6: Regression Coefficients over time for the constrained and shallow architecture

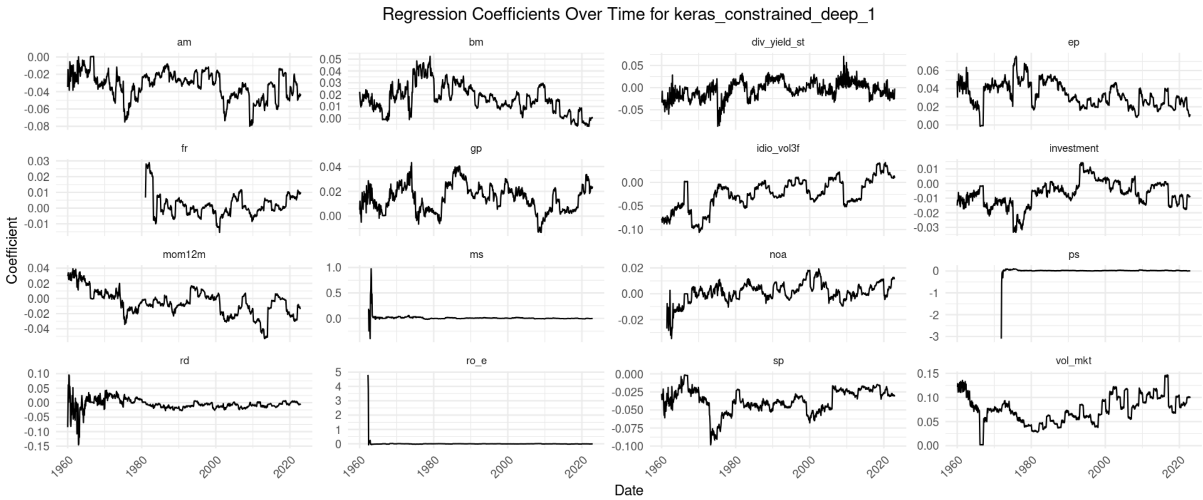

Figure A 7: Regression Coefficients over time for the constrained and deep architecture

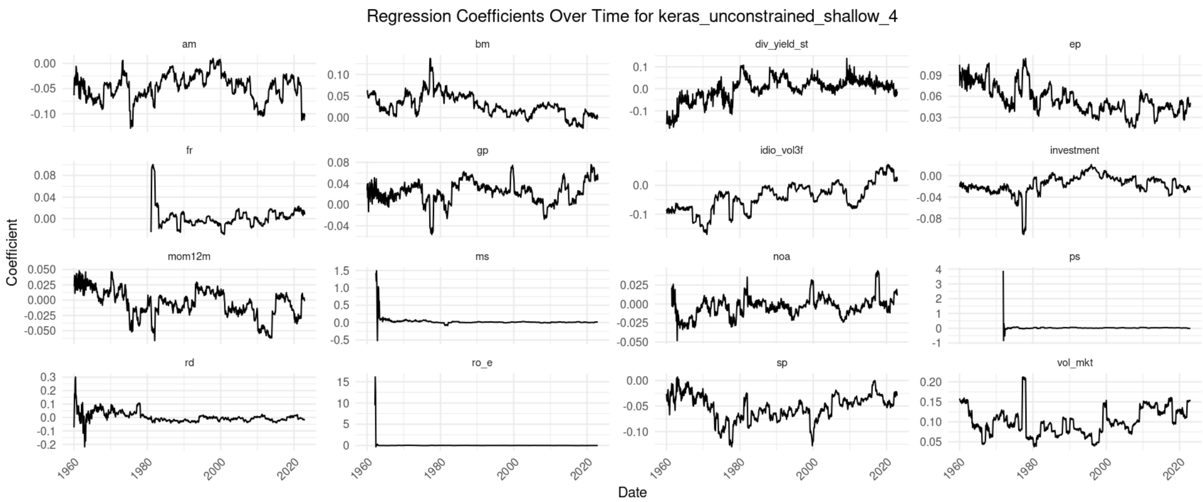

Figure A 8: Regression Coefficients over time for the unconstrained and shallow architecture

	div_yield_st	ep	mom12m	idio_vol3f	investment	bm	am	fr	gp	ms	noa	ps	rd	ro_e	sp	vol_mkt
div_yield_st	1.00	0.11	-0.02	0.11	-0.04	0.02	0.09	-0.10	-0.03	-0.09	-0.02	-0.07	-0.06	0.02	0.06	0.06
ep	0.11	1.00	-0.08	0.10	-0.13	0.20	0.60	-0.04	-0.03	-0.10	0.04	0.19	0.02	0.21	0.54	-0.03
mom12m	-0.02	-0.08	1.00	0.09	0.00	-0.03	-0.18	0.00	0.06	0.02	0.03	-0.01	-0.05	0.07	-0.17	0.11
idio_vol3f	0.11	0.10	0.09	1.00	-0.01	-0.03	0.03	0.00	-0.07	-0.06	-0.03	-0.03	-0.09	0.14	-0.09	0.41
investment	-0.04	-0.13	0.00	-0.01	1.00	0.23	-0.01	0.08	-0.33	-0.08	0.31	-0.02	-0.07	-0.10	-0.22	0.11
bm	0.02	0.20	-0.03	-0.03	0.23	1.00	0.37	0.05	-0.43	-0.34	0.16	0.17	-0.08	-0.41	0.23	-0.01
am	0.09	0.60	-0.18	0.03	-0.01	0.37	1.00	-0.09	-0.36	-0.21	0.11	0.25	-0.03	-0.18	0.68	-0.09
fr	-0.10	-0.04	0.00	0.00	0.08	0.05	-0.09	1.00	0.03	0.24	0.05	0.33	0.20	0.07	-0.09	0.01
gp	-0.03	-0.03	0.06	-0.07	-0.33	-0.43	-0.36	0.03	1.00	0.32	-0.12	-0.03	0.29	0.35	0.14	-0.10
ms	-0.09	-0.10	0.02	-0.06	-0.08	-0.34	-0.21	0.24	0.32	1.00	0.03	0.21	0.40	0.32	-0.12	-0.09
noa	-0.02	0.04	0.03	-0.03	0.31	0.16	0.11	0.05	-0.12	0.03	1.00	0.11	0.13	0.03	0.05	0.06
ps	-0.07	0.19	-0.01	-0.03	-0.02	0.17	0.25	0.33	-0.03	0.21	0.11	1.00	0.37	0.03	0.27	-0.05
rd	-0.06	0.02	-0.05	-0.09	-0.07	-0.08	-0.03	0.20	0.29	0.40	0.13	0.37	1.00	0.09	0.12	-0.16
ro_e	0.02	0.21	0.07	0.14	-0.10	-0.41	-0.18	0.07	0.35	0.32	0.03	0.03	0.09	1.00	-0.08	0.07
sp	0.06	0.54	-0.17	-0.09	-0.22	0.23	0.68	-0.09	0.14	-0.12	0.05	0.27	0.12	-0.08	1.00	-0.18
vol_mkt	0.06	-0.03	0.11	0.41	0.11	-0.01	-0.09	0.01	-0.10	-0.09	0.06	-0.05	-0.16	0.07	-0.18	1.00

Table A 1: Correlations of predictors

GitHub Repository with Code for the model
https://github.com/APUniLi/Master_Thesis

Table A 2: GitHub Repository with the code for the model

Eidesstattliche Erklärung

Optimizing S&P 500-Tilted Portfolios with Direct Weight Prediction Neural Networks: Information Ratio Performance under Realistic Constraints and Network Depth

Ich erkläre an Eides statt, dass ich die Arbeit selbstständig und ohne unzulässige Inanspruchnahme Dritter verfasst habe. Ich habe dabei nur die angegebenen Quellen und Hilfsmittel verwendet und die aus diesen wörtlich oder sinngemäss entnommenen Stellen als solche kenntlich gemacht. Im Hilfsmittelverzeichnis habe ich sämtliche verwendete KI-Tools mit ihrem Produktnamen benannt und die Art und Weise ihrer Nutzung beschrieben. Ich bin mir bewusst, dass die Nutzung maschinell generierter Texte keine Garantie für die Qualität von Inhalten und Text gewährleistet. Ich versichere, dass ich mich KI-Tools lediglich als Hilfsmittel bedient habe und in der vorliegenden Arbeit mein gestalterischer Einfluss überwiegt. Ich verantworte die Übernahme jeglicher von mir verwendeter maschinell generierter Textpassagen oder sonstiger Elemente der Arbeit vollumfänglich selbst. Die Versicherung selbstständiger Arbeit gilt auch für enthaltene Zeichnungen, Skizzen oder graphische Darstellungen. Die Arbeit wurde bisher in gleicher oder ähnlicher Form weder derselben noch einer anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht. Mit der Abgabe der elektronischen Fassung der endgültigen Version der Arbeit nehme ich zur Kenntnis, dass diese mit Hilfe eines Plagiatserkennungsdienstes auf enthaltene Plagiate geprüft werden kann.

Vaduz, 30.06.2025 Andreas Pischetsrieder

Ort und Datum Vorname und Nachname

Andreas Pischetsrieder

Unterschrift