Quantitative Finance

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/rquf20

Neural network approach to portfolio optimization with leverage constraints: a case study on high inflation investment

Chendi Ni, Yuying Li & Peter Forsyth

To cite this article: Chendi Ni, Yuying Li & Peter Forsyth (2024) Neural network approach to portfolio optimization with leverage constraints: a case study on high inflation investment, Quantitative Finance, 24:6, 753-777, DOI: 10.1080/14697688.2024.2357733

To link to this article: https://doi.org/10.1080/14697688.2024.2357733

Published online: 11 Jun 2024.

Quantitative Finance, 2024
Vol. 24, No. 6, 753–777, https://doi.org/10.1080/14697688.2024.2357733

Neural network approach to portfolio optimization with leverage constraints: a case study on high inflation investment

CHENDI NI†‡, YUYING LI*† and PETER FORSYTH†

†Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, N2L 3G1, Canada
‡Flap Technologies, 307 W 38th St, Floor 16, PMB 468, New York, NY, 10018, USA

(*Received 31 May 2023; accepted 11 May 2024; published online 12 June 2024*)

Motivated by the current global high inflation scenario, we aim to discover a dynamic multi-period allocation strategy to optimally outperform a passive benchmark while adhering to a bounded leverage limit. We formulate an optimal control problem to outperform a benchmark portfolio throughout the investment horizon. To obtain strategies under the bounded leverage constraint among other realistic constraints, we propose a novel leverage-feasible neural network (LFNN) to represent the control, which converts the original constrained optimization problem into an unconstrained optimization problem that is computationally feasible with gradient descent, without dynamic programming. We establish mathematically that the LFNN approximation can yield a solution that is arbitrarily close to the solution of the original optimal control problem with bounded leverage. We further validate the performance of the LFNN empirically by deriving a closed-form solution under jump-diffusion asset price models and show that a shallow LFNN model achieves comparable results on synthetic data. In the case study, we apply the LFNN approach to a four-asset investment scenario with bootstrap-resampled asset returns from the filtered high inflation regimes. The LFNN strategy is shown to consistently outperform the passive benchmark strategy by about 200 bps (median annualized return), with a greater than 90% probability of outperforming the benchmark at the end of the investment horizon.

Keywords: Cumulative tracking difference; Leveraged portfolio; Benchmark outperformance; Asset allocation; Machine learning

JEL Classifications: G11, G22
AMS Codes: 91G, 35Q93, 68T07

1. Introduction

Since the global outbreak of COVID-19 in March 2020, there has been a significant increase in worldwide inflation. Specifically, from May 2021 to February 2023, the 12-month change in the CPI index in the U.S. has not dropped below 5% (Bureau of Labor Statistics 2023). Prior to the pandemic, the U.S. economy experienced nearly four decades of low inflation. The abrupt shift from a long-term low-inflation environment to a high-inflation environment has created substantial uncertainty and volatility in the financial markets. In 2022, the technology-heavy NASDAQ stock index recorded a yearly return of -33.10% (NASDAQ 2023).

Equally concerning is the uncertainty around the duration of this round of high inflation. Some believe that the geopolitical tensions and the COVID-19 pandemic will overturn the trend of globalization and lead to global supply chain restructuring (Javorcik 2020) which may result in a higher cost of production in the foreseeable future. Moreover, Ball *et al.* (2022) suggests that the future inflation rate may remain high if the unemployment rate remains low.

Meanwhile, notable sovereign wealth funds like the Canada Pension Plan (CPP) and the Government Pension Fund Global of Norway, commonly known as the Oil Fund, have exhibited lackluster performance in the years leading up to the COVID crisis. Specifically, in comparison to passive benchmark portfolios consisting of fixed allocations to public stock and bond indexes, the actively managed portfolios overseen by CPP and the Oil Fund have struggled to achieve substantial outperformance.

This puzzling trend is noteworthy, given that sovereign wealth funds such as the CPP or the Oil Fund often

*Corresponding author. Email: yuying@uwaterloo.ca

incorporate alternative assets like private equity, which inherently introduces leverage to the portfolio. Despite having a broader array of assets to invest in, these funds face challenges in surpassing benchmark performance. Adding to the concerns is the fact that most sovereign wealth funds were established after the last wave of inflation regimes in the 1980s.

As noted, many pension plans are using alternative assets, such as private equity and private credit. Some authors have suggested that returns on private equity can be replicated using a leveraged small capitalization stock index (see Phalippou 2014, L'Her *et al.* 2016). Hence, it would appear that funds which hold alternative assets use an implicit form of leverage.

We also note that the CPP explicitly allows use of leverage (CPP Investments 2022), and there is a growing trend for other funds to use explicit leverage (Cumbo *et al.* 2024).

In this article, we adopt the perspective of such sovereign wealth funds and seek to answer a crucial question: Can we, with a portfolio allowing a bounded amount of leverage, actively manage the portfolio to outperform a passive benchmark, particularly in the context of a high-inflation regime?

To answer this question, we formulate a multi-period portfolio optimization problem with bounded leverage constraints, with the goal of optimizing specific objective functions that measure the outperformance over the benchmark. Literature in multi-period portfolio optimization can be traced back to Merton (1969), who considers a continuous-time asset allocation setting and utilizes the dynamic programming principle to derive the optimal portfolio that maximizes a constant relative risk aversion (CRRA) utility function of terminal wealth of the portfolio. Since then, extensive research has been conducted on multi-period portfolio optimization under the continuous rebalancing assumption (Merton 1971, Browne 1997, Zhou and Li 2000, Blanchet-Scalliet *et al.* 2008, Ang *et al.* 2014). Particularly, there is a large amount of existing literature on developing closed-form solutions for beating a stochastic benchmark under synthetic market assumptions (Browne 1999, 2000, Tepla 2001, Basak *et al.* 2006, Davis and Lleo 2008, Lim and Wong 2010a, Oderda 2015, Alekseev and Sokolov 2016, Al-Aradi and Jaimungal 2018).

Closed-form solutions are usually derived assuming continuous rebalancing and other unrealistic assumptions such as unlimited leverage and trading in insolvency. A discrete-time multi-period asset allocation problem, on the other hand, is often numerically solved using dynamic programming-based approaches. One particular method is the numerical PDE approach which uses dynamic programming to establish an associated Hamilton–Jacobi–Bellman (HJB) equation with the portfolio optimization problem (Wang and Forsyth 2010, Dang and Forsyth 2014). In Dang and Forsyth (2014), constraints such as no-shorting, no-leverage, and discrete rebalancing are considered, and the optimal portfolio is computed by solving a numerical HJB equation. However, a numerical HJB equation solution is only practical if there is a small number (three or fewer) of state variables (≤ 3). Under discrete rebalancing, with N_a assets and a benchmark strategy, the PDE problem has a dimension of $2N_a + 1$ (including time), which is intractable for an HJB equation for $N_a \geq 2$, see e.g. Dang and Forsyth (2014). Therefore, existing numerical PDE methods (Wang and Forsyth 2010, Dang and Forsyth 2014) that convert the problem into an HJB equation are not practical in these high-dimensional cases.

An alternative numerical solution is a simulation-based framework that utilizes approximate dynamic programming (ADP) (Cong and Oosterlee 2016, Li *et al.* 2020). In this approach, a large number of asset price paths are simulated. The ADP then backwardly estimates the conditional expectation via regression or other methodologies. Subsequently, this technique seeks the optimal investment policy by optimizing the estimated conditional expectation. However, the stability and convergence pose significant concerns in this context, since the optimization is conducted on the estimated value function, and this process may magnify the estimation errors, particularly in the case of high-dimensional portfolios (Li *et al.* 2020).

Recent advancements in artificial intelligence have inspired research that employs reinforcement learning (RL) techniques to address multi-period portfolio optimization challenges. One widely adopted approach is the use of the Q-learning method in portfolio optimization (Dixon *et al.* 2020, Gao *et al.* 2020, Lucarelli and Borrotti 2020, Park *et al.* 2020). This method uses a deep neural network to approximate the conditional expectation (Q-function). However, van Staden *et al.* (2023) point out that these methods require the evaluation of a high-dimensional performance criterion to obtain the optimal control which is comparatively low-dimensional, thus the Q-learning methods may be inefficient and are computationally prone to known issues such as error amplification over recursions (Wang *et al.* 2020).

Acknowledging these identified limitations, we propose to use a single neural network model to approximate the optimal control. This approach enables the solution of a standard finite-dimensional optimization problem using gradient descent and eliminates the need for dynamic programming. The direct approximation of the control leverages the lower dimensionality inherent in optimal control and thus avoids the challenge of solving high-dimensional conditional expectations encountered in Q-learning methods.

We note that the idea of using a neural network to directly approximate the control process is also used in Han (2016) and Tsang and Wong (2020), in which they propose a stacked neural network approach that includes individual sub-networks for each rebalancing step. In contrast, we propose to use a shallow neural network that includes time as an input feature, and thus avoids the need to have multiple sub-networks for each rebalancing step and greatly reduces the computational and modeling complexity. Furthermore, using time as a feature in the neural network approximation function is consistent with the observation that, under assumptions, the optimal control is naturally a continuous function of time, which we discuss in detail in section 4.1.

More similar to our work, Li and Forsyth (2019) and Ni *et al.* (2022) have also considered using a single neural network to approximate controls. These studies focus on portfolio optimization problems with long-only constraints. The neural network architecture proposed in these studies transforms the constrained portfolio optimization problems into unconstrained optimization problems, making them computationally easier to solve.

However, existing multi-period portfolio optimization literature, including the aforementioned neural network methods, does not accommodate a bounded leverage constraint. The scarcity of literature on portfolio optimization with bounded leverage is likely due to the heightened complexity introduced by the combination of long and short positions in the portfolio.

The main novelty of this article is the introduction of a leverage-feasible neural network (LFNN) model, which allows converting the leverage-constrained optimization problem into an unconstrained optimization problem. This model incorporates the bounded leverage constraint into the portfolio optimization framework, thus filling a gap in the existing literature. Additionally, in section 3.2, we provide a mathematical proof that, under reasonable assumptions, the solution of the unconstrained optimization problem obtained using the LFNN model can approximate the optimal control of the original problem arbitrarily well.

This mathematical justification validates the effectiveness of the LFNN approach.

The validity and effectiveness of the LFNN model are further substantiated through empirical evidence in section 4. Specifically, we derive a closed-form solution under a cumulative quadratic tracking difference (CD) objective and jump-diffusion asset price models. Using this closed-form solution as a benchmark, we demonstrate that the LFNN model achieves comparable performance with a shallow network structure.

In section 5, we present a case study on active portfolio optimization during a high-inflation regime. We use a simple filtering method to identify historical high-inflation periods and use bootstrap resampling to generate training and testing datasets with price paths for four assets: the equal-weighted and cap-weighted stock indexes, as well as the 30-day and 10-year U.S. treasury indexes. Utilizing the LFNN model and the cumulative quadratic tracking shortfall (CS) objective, we derive an optimal leverage-constrained strategy. Our results demonstrate that the LFNN model produces a strategy that consistently outperforms the fixed-mix 70/30 benchmark. Specifically, the strategy achieves a median (annualized) internal rate of return (IRR) that is more than 2% higher than the benchmark. Moreover, there is a probability of over 90% that the strategy will yield a higher terminal wealth compared to the benchmark.

These findings highlight the efficacy of the LFNN model in optimizing portfolios under high-inflation conditions. By incorporating the bounded leverage constraint and utilizing the CS objective, our approach enables investors to achieve superior performance relative to the selected benchmark and mitigate risks in a high-inflation environment.

The main contributions of this article are summarized below:

(i) We propose the novel leverage-feasible neural network (LFNN) model to convert the original complex leverage-constrained optimization problem into an unconstrained optimization problem that can be solved easily by standard optimization methods.

(ii) We prove that, with a suitable choice of the hyperparameter of the LFNN model, the solution of the parameterized unconstrained optimization problem can approximate the optimal control arbitrarily well. In addition, we show that a shallow LFNN model achieves comparable empirical performance as the clipped form solution on synthetic data, thus providing empirical evidence for the approximation capability of the LFNN model.

(iii) We derive the closed-form solution under a jump-diffusion asset price model and other typical assumptions (such as continuous rebalancing) for a two-asset case. The closed-form solution provides important insights into the properties of the optimal control as well as meaningful interpretations of the neural network models that approximate the controls.

(iv) In the case study on active portfolio optimization in high inflation, we apply the neural network method to bootstrap-resampled asset returns with four underlying assets, including the equal-weighted/cap-weighted stock indexes, and the 30-day/10-year treasury bond indexes. The dynamic strategy from the learned LFNN model outperforms a fixed-mix benchmark strategy consistently throughout the investment horizon, with a 2% higher median (annualized) internal rate of return (IRR). The numerical results demonstrate the capability of the LFNN model for achieving the desired investment targets.

2. Outperforming a dynamic benchmark under bounded leverage

2.1. Sovereign wealth funds and benchmark targets

Instead of taking a passive approach, some of the largest sovereign wealth funds often adopt an active management philosophy and use passive portfolios as the benchmark to evaluate the efficiency of active management. For example, the Canadian Pension Plan (CPP) uses a base reference portfolio of 85% global equity and 15% Canadian government bonds (CPP Investments 2022). Another example is the Government Pension Fund Global of Norway (also known as the Oil Fund) managed by Norges Bank Investment Management (NBIM), which uses a benchmark index consisting of 70% equity index and 30% bond index.† The benchmark equity index is constructed based on the market capitalization for equities in the countries included in the benchmark. The benchmark index for bonds specifies a defined allocation between government bonds and corporate bonds, with a weight of 70 percent to government bonds and 30 percent to corporate bonds (Norges Bank 2022).

However, the excess return that these well-known sovereign wealth funds have achieved over their respective passive benchmark portfolios cannot be described as impressive. In the 2022 fiscal year report, CPP claims to have beaten the base reference portfolio by an annualized 80 bps after fees over the past 5 years (CPP Investments 2022). On the other hand, NBIM reports a mere average of 27 bps of annual excess

† The Ministry of Finance of Norway sets the allocation fraction between the equity index and the bond index. It gradually raised the weight for equities from 60% to 70% from 2015 to 2018.

return over the benchmark over the last decade (see table 1). It is worth noting that these behemoth funds achieve seemingly meager results by hiring thousands of highly paid investment professionals and spending billions of dollars on day-to-day operations. For example, the CPP 2021 annual report (CPP Investments 2021) lists personnel costs as CAD 938 million, for 1936 employees, which translates to average costs of about CAD 500 000 per employee-year.

The stark contrast between the enormous spending of sovereign wealth funds and the meager outperformance of the funds relative to the passive benchmark portfolios is probably provocative to taxpayers and pensioners who invest their hard-earned money in the funds. In fact, the portfolios of these large pension funds often have exposures to alternative assets, such as private equity (CPP Investments 2022). Literature suggests that returns on private equity can be replicated using a leveraged small-cap stock index (Phalippou 2014, L'Her *et al.* 2016). Following this line of argument, one might contend that the portfolios managed by these sovereign wealth funds are inherently leveraged. This raises further questions about the limited outperformance, particularly when the benchmark portfolios do not incorporate leverage.

Equally concerning is the potential of a long, persistent inflation regime and the funds' ability to consistently beat the benchmark portfolio in such times. After all, both the CPP Investments and NBIM were established in the late 1990s, a decade after the last long inflation period ended in the mid-1980s.

These concerns prompt us to ask the following question: in a presumed persistent high-inflation environment, can an active leveraged portfolio consistently outperform the passive benchmark portfolio (preferably without spending billions of dollars in personnel costs)?

2.2. *Mathematical formulation*

In this section, we mathematically formulate the problem of outperforming a benchmark. Let $[t_0(=0), T]$ denote the investment horizon, and let $W(t)$ denote the wealth (value) of the portfolio actively managed by the manager at time $t \in [t_0, T]$. We refer to the actively managed portfolio as the 'active portfolio'. Furthermore, let $\hat{W}(t)$ denote the wealth of the benchmark portfolio at time $t \in [t_0, T]$. To ensure a fair assessment of the relative performance of the two portfolios, we assume both portfolios start with an equal initial wealth amount $w_0 > 0$, i.e. $W(t_0) = \hat{W}(t_0) = w_0 > 0$.

Technically, the admissible sets of underlying assets for the active and passive portfolio need not be identical. However, for simplicity, we assume that both the active portfolio and the benchmark portfolio can allocate wealth to the same set of N_a assets.

Let vector $\boldsymbol{S}(t) = (S_i(t) : i = 1, \ldots, N_a)^\top \in \mathbb{R}^{N_a}$ denote the asset prices of the N_a underlying assets at time $t \in [t_0, T]$. In addition, let vectors $\boldsymbol{p}^{(t)} = (p_i^{(t)} : i = 1, \ldots, N_a)^\top \in \mathbb{R}^{N_a}$ and $\hat{\boldsymbol{p}}^{(t)} = (\hat{p}_i^{(t)} : i = 1, \ldots, N_a)^\top \in \mathbb{R}^{N_a}$ denote the allocation fractions to the N_a underlying assets at time $t \in [t_0, T]$, respectively, for the active portfolio and the benchmark portfolio. In this article, we consider a passive benchmark portfolio, i.e. $\hat{\boldsymbol{p}}^{(t)}$ is a pre-defined constant vector independent of time such that $\hat{\boldsymbol{p}}^{(t)} \equiv \hat{\boldsymbol{p}}$. It is possible to let $\hat{\boldsymbol{p}}^{(t)} \equiv \hat{\boldsymbol{p}}(t)$, i.e. a glide path type benchmark strategy. However, this results in complex solutions which are not amenable to building intuition. In addition, constant weight benchmarks are very common in practice.

Under an optimal control perspective, the allocation vector $\boldsymbol{p}^{(t)}$ is a stochastic process denoting the control value at time t, which defines the evolution of the portfolio values. In general, the control is a function of the state variables that fully describe the state of the dynamic system. It is shown that under common assumptions on the asset price evolutions, e.g. jump-diffusion processes, the state variables for the optimal control are simply the portfolio values and time (Dang and Forsyth 2014). While we consider the simple case of the portfolio values and time as state variables in this article, incorporating additional features as state variables poses no technical challenges for the proposed methodology. Mathematically, $\boldsymbol{p}^{(t)} = p(\boldsymbol{X}(t)) = (p_i(\boldsymbol{X}(t)) : i \in \{1, \ldots, N_a\})^\top \in \mathbb{R}^{N_a}$, e.g. $\boldsymbol{X}(t) = (t, W(t), \hat{W}(t))^\top \in \mathcal{X} \subseteq \mathbb{R}^3$, and $p_i : \mathcal{X} \mapsto \mathbb{R}$. Our goal is to find the optimal control function $p : \mathcal{X} \mapsto \mathbb{R}^{N_a}$, so that the relative performance measure of the active portfolio (following control p) over the benchmark portfolio (following $\hat{\boldsymbol{p}}^{(t)}$) is maximized.

We assume that the active portfolio and the benchmark portfolio follow the same discrete rebalancing schedule denoted by $\mathcal{T} \subseteq [t_0, T]$. Specifically, we consider an equally spaced discrete schedule with N rebalancing events, i.e.

$$\mathcal{T} = \{t_i : \ i = 0, \ldots, N-1\}, \tag{1}$$

where $t_i = i\Delta t$, and $\Delta t = T/N$.†

Additionally, we assume both portfolios follow the same deterministic sequence of cash injections, defined by the set $\mathcal{C} = \{c(t), \ t \in \mathcal{T}_c\}$, where $\mathcal{T}_c \subseteq [t_0, T]$ is the schedule of the cash injections. When $\mathcal{T}_c$ is a discrete injection schedule, $c(t)$ is the amount of cash injection at t. For simplicity, we assume that $\mathcal{T}_c = \mathcal{T}$, so that the cash injections schedule is the same as the rebalancing schedule. At $t \in \mathcal{T}$, $W(t)$ and $\hat{W}(t)$ always denote the wealth after the cash injection (assuming there is a cash injection event happening at t).

Denote $\mathcal{A}$ as the set of admissible strategies, which satisfies the investment constraints on the controls. We assume that admissibility can vary with state, e.g. insolvency, and let $\{\mathcal{X}_i : i = 1, \ldots, k\}$ be a partition of $\mathcal{X}$ (the state variable space), i.e.

$$\begin{cases} \bigcup_{i=1}^{k} \mathcal{X}_i = \mathcal{X}, \\ \mathcal{X}_i \bigcap \mathcal{X}_j = \varnothing, \quad \forall 1 \le i < j \le k, \end{cases} \tag{2}$$

and $\{\mathcal{Z}_i \subseteq \mathbb{R}^{N_a} : i = 1, \ldots, k\}$ be the corresponding value sets of feasible controls.

We say that a control $p : \mathcal{X} \mapsto \mathbb{R}^{N_a}$ is an admissible strategy, i.e. $p \in \mathcal{A}$, if and only if

$$\text{(Feasibility of control):} \quad p(\boldsymbol{x}) \in \mathcal{Z}_i, \quad \forall \boldsymbol{x} \in \mathcal{X}_i, \quad \forall i \in \{1, \ldots, k\}. \tag{3}$$

† Technically, at $t = t_0$, the manager makes the initial asset allocation, rather than a 'rebalancing' of the portfolio. However, despite the different purposes, a rebalancing of the portfolio is simply a new allocation of the portfolio wealth. Therefore, for notational simplicity, we include t_0 in the rebalancing schedule.

Table 1. Norges Bank Investment Management, relative return to benchmark portfolio.

Year	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	Average
Excess return (%)	0.21	0.99	− 0.77	0.45	0.15	0.70	− 0.30	0.23	0.27	0.74	0.27

Then, the wealth evolution of the active portfolio and the benchmark portfolio can be described by the equations

$$\begin{cases} W(t_{j+1}) = \left(\sum_{i=1}^{N_a} p_i(\boldsymbol{X}(t_j)) \cdot \dfrac{S_i(t_{j+1}) - S_i(t_j)}{S_i(t_j)}\right) W(t_j) \\ \quad +c(t_{j+1}), \quad j = 0, \ldots, N-1, \\ \hat{W}(t_{j+1}) = \left(\sum_{i=1}^{N_a} \hat{p}_i(\hat{\boldsymbol{X}}(t_j)) \cdot \dfrac{S_i(t_{j+1}) - S_i(t_j)}{S_i(t_j)}\right) \hat{W}(t_j) \\ \quad +c(t_{j+1}), \quad j = 0, \ldots, N-1. \end{cases} \tag{4}$$

Let sets $\mathcal{W}_p = \{W(t), t \in \mathcal{T}\}$ and $\hat{\mathcal{W}}_{\hat{p}} = \{\hat{W}(t), t \in \mathcal{T}\}$ represent the trajectories of the portfolio values for the active portfolio and the benchmark portfolio, respectively, following the dynamics specified in equation (4). We introduce an investment metric denoted by $F(\mathcal{W}_p, \hat{\mathcal{W}}_{\hat{p}}) \in \mathbb{R}$, which quantifies the relative performance of the active portfolio in relation to the benchmark portfolio based on their respective value trajectories.

In this article, we assume the asset prices $\boldsymbol{S}(t) \in \mathbb{R}^{N_a}$ are stochastic. Consequently, the value trajectories $\mathcal{W}_p$, $\hat{\mathcal{W}}_{\hat{p}}$, and the performance metric $F(\mathcal{W}_p, \hat{\mathcal{W}}_{\hat{p}})$ are also stochastic. Therefore, when investment managers aim to optimize an investment objective, the evaluation commonly involves calculating the expectation of this metric.

Let $\mathbb{E}_p^{(t_0,w_0)}[F(\mathcal{W}_p, \hat{\mathcal{W}}_{\hat{p}})]$ denote the expectation of the performance metric F, given a specific initial wealth value $w_0 = W(0) = \hat{W}(0)$ at time $t_0 = 0$, and evaluated on the wealth trajectories following the admissible investment strategies $p \in \mathcal{A}$ and the benchmark investment strategy $\hat{p}$. As we assume the benchmark strategy to be predetermined, we keep the benchmark strategy $\hat{p}$ implicit in this notation for simplicity. Subsequently, we try to solve the following stochastic optimization (SO) problem:

(Stochastic optimization problem):

$$\inf_{p \in \mathcal{A}} \mathbb{E}_p^{(t_0,w_0)} \left[F(\mathcal{W}_p, \hat{\mathcal{W}}_{\hat{p}}) \right]. \tag{5}$$

2.3. Leverage constraints

As previously discussed, we assume that managers can take leverage to invest in public stock index funds to roughly mimic the pension fund portfolios with some exposure to private equity. Typically, due to risk mandates, the total leverage in the portfolio is subject to an upper bound.

Essentially, taking leverage to invest in stocks requires borrowing additional capital. For simplicity, we consider a self-financing portfolio and assume the borrowing activity is represented by shorting some bond assets within the portfolio.

In this section, we mathematically formulate the leverage constraints. Following the notations from section 2.2, we assume that the total N_a underlying assets are divided into two groups. The first group of N_L assets are long-only assets, which we index by the set $\{1, \ldots, N_L\}$. The second group of $N_a - N_L$ assets are shortable assets that can be shorted to create leverage and are indexed by the set $\{N_L + 1, \ldots, N_a\}$. Recall the notation of $p_i(\boldsymbol{X}(t))$ for the allocation fraction for asset i at time t, as a function of the state variable vector $\boldsymbol{X}(t)$.

For long-only assets, the wealth fraction needs to be non-negative, hence we have

(Long-only constraint):

$$p_i(\boldsymbol{X}(t)) \geq 0, \quad i \in \{1, \ldots, N_L\}, \ t \in \mathcal{T}. \tag{6}$$

Furthermore, the total allocation fraction for all assets should be one. Therefore, the following summation constraint needs to be satisfied

$$\text{(Summation constraint):} \quad \sum_{i=1}^{N_a} p_i(\boldsymbol{X}(t)) = 1, \quad t \in \mathcal{T}. \tag{7}$$

In practice, due to borrowing costs (from taking leverage) and risk management mandates, the level of leverage is often constrained. For this reason, we cap the maximum leverage by introducing a constant $p_{\max}$, which represents the total allocation fraction for long-only assets. Therefore,

(Maximum leverage constraint):

$$\sum_{i=1}^{N_L} p_i(\boldsymbol{X}(t)) \leq p_{\max}, \quad t \in \mathcal{T}. \tag{8}$$

Note that no leverage is permitted if $p_{\max} = 1$.

Finally, we make the following assumption on the scenario of shorting multiple shortable assets.

Assumption 2.1 (Simultaneous shorting) If one shortable asset has a negative weight, other shortable assets must have nonpositive weights. Mathematically, this assumption can be expressed as

(Simultaneous shorting):

$$\begin{cases} p_i(\boldsymbol{X}(t)) \leq 0, & \forall i \in \{N_L + 1, \ldots, N_a\}, \\ \quad \text{if } \sum_{i=1}^{N_L} & p_i(\boldsymbol{X}(t)) > 1, \ t \in \mathcal{T} \\ p_i(\boldsymbol{X}(t)) \geq 0, & \forall i \in \{N_L + 1, \ldots, N_a\}, \\ \quad \text{if } \sum_{i=1}^{N_L} & p_i(\boldsymbol{X}(t)) \leq 1, \ t \in \mathcal{T} \end{cases}. \tag{9}$$

Remark 2.1 (Remark on Assumption 2.1) A typical scenario would involve long-only equities and long-term bonds, while allowing shorting of short-term T-bills, i.e. effectively borrowing cash. Assumption 2.1 is an extension of this scenario.

The above constraints permit scenarios with non-negative portfolio wealth. Before we proceed to the handling of the negative portfolio wealth scenarios, we first define the following partition of the state space $\mathcal{X}$.

DEFINITION 2.1 (Partition of state space) *We define* $\{\mathcal{X}_1, \mathcal{X}_2\}$ *to be a partition of the state space* $\mathcal{X}$*, such that*

$$\begin{cases} \mathcal{X}_1 = \left\{ x = (t, W, \hat{W})^\top \in \mathcal{X} \,\middle|\, W \geq 0 \right\}, \\ \mathcal{X}_2 = \left\{ x = (t, W, \hat{W})^\top \in \mathcal{X} \,\middle|\, W < 0 \right\}. \end{cases} \tag{10}$$

Intuitively, we separate the state space $\mathcal{X}$ into two regions by the wealth of the active portfolio, one with non-negative wealth and the other with negative wealth. Since we allow shorting, and the portfolio is discretely rebalanced, it is possible for the investor to become insolvent. Then, we present the following assumption concerning the negative wealth (insolvency) scenarios.

ASSUMPTION 2.2 (No trading in insolvency) If the wealth of the active portfolio is negative, then all long-only asset positions should be liquidated, and all the debt (i.e. the negative wealth) should be allocated to the least-risky shortable asset (in terms of volatility). In particular, without loss of generality, we assume all debt is allocated to the shortable asset indexed with $N_L + 1$. Let $\boldsymbol{e}_i \in \mathbb{R}^{N_a} = (0, \ldots, 0, 1, 0, \ldots, 0)^\top$ denote the standard basis vector of which the ith entry is 1 and all other entries are 0. Then, we can formulate this assumption as follows.

(No trading in insolvency):

$$p(\boldsymbol{X}(t)) = \boldsymbol{e}_{N_L+1}, \quad \text{if } \boldsymbol{X}(t) \in \mathcal{X}_2. \tag{11}$$

REMARK 2.2 (Remark on Assumption 2.2) Essentially, when the portfolio wealth is negative, we assume the debt is allocated to a short-term bond asset and accumulates over time. This approach penalizes outcomes which generate insolvency. Note that any strategy which allows (i) leverage and (ii) is discretely rebalanced, has a finite probability of insolvency. However, the optimal control tends to minimize the probability of this outcome.

Summarizing the constraints, we can define two sets $\mathcal{Z}_1, \mathcal{Z}_2$:

$$\begin{cases} \mathcal{Z}_1 = \left\{ z \in \mathbb{R}^{N_a} \,\middle|\, \begin{cases} z_i \geq 0, \ \forall i \in \{1, \ldots, N_L\}, \\ \sum_{i=1}^{N_a} z_i = 1, \\ \sum_{i=1}^{N_L} z_i \leq p_{\max}, \\ z_i \leq 0, \ \forall i \in \\ \quad \{N_L + 1, \ldots, N_a\}, \text{ if } \sum_{i=1}^{N_L} z_i > 1, \\ z_i \geq 0, \ \forall i \in \\ \quad \{N_L + 1, \ldots, N_a\}, \text{ if } \sum_{i=1}^{N_L} z_i \leq 1 \end{cases} \right\}, \\ \mathcal{Z}_2 = \left\{ \boldsymbol{e}_{N_L+1} \right\}, \end{cases} \tag{12, 13}$$

Then, the corresponding space of feasible control vector values $\mathcal{Z}$ and the admissible strategy set $\mathcal{A}$ can be expressed as follows.

(Admissible set):

$$\begin{cases} \mathcal{Z} = \mathcal{Z}_1 \cup \mathcal{Z}_2, & (14) \\ \mathcal{A} = \left\{ p \,\middle|\, \begin{cases} p(\boldsymbol{X}(t)) \in \mathcal{Z}_1, \text{ if } \boldsymbol{X}(t) \in \mathcal{X}_1, \\ p(\boldsymbol{X}(t)) \in \mathcal{Z}_2, \text{ if } \boldsymbol{X}(t) \in \mathcal{X}_2, \end{cases} \quad \forall t \in \mathcal{T} \right\} & (15) \end{cases}$$

Consequently, the stochastic optimization problem (5), i.e.

$$\inf_{p \in \mathcal{A}} \mathbb{E}_p^{(t_0, w_0)} \left[F(\mathcal{W}_p, \hat{\mathcal{W}}_{\hat{p}}) \right], \tag{16}$$

is a stochastically constrained optimization problem, which is challenging to solve.

3. Neural network method

Solving a discrete-time multi-period optimal asset allocation problem often utilizes dynamic programming (DP). For example, in numerical PDE methods (Dang and Forsyth 2014), the portfolio optimization problem is converted to a numerical HJB equation, and the dimension of the problem increases with the number of assets, rendering the HJB equation intractable. For Q-learning methods (Dixon *et al.* 2020, Park *et al.* 2020, Lucarelli and Borrotti 2020, Gao *et al.* 2020), if there are N_a assets to invest in, then the use of Q-learning involves approximation of an action-value function ('Q' function) which is a $2N_a$-dimensional function (van Staden *et al.* 2023) which represents the conditional expectation of the cumulative rewards at an intermediate state.† Meanwhile, the optimal control itself is a mapping from the state space to the allocation fractions to the assets. If the state space of the optimal control is relatively low-dimensional,‡ then such methods are potentially unnecessarily high-dimensional.

On the contrary, simulation-based approaches employing approximate dynamic programming (ADP) have been utilized to circumvent the challenges posed by high dimensionality (Cong and Oosterlee 2016, Li *et al.* 2020). In these methods, the conditional expectation is estimated backward through regression (or other methods), before the optimal investment policy is determined based on the estimated conditional expectation. However, stability and convergence issues become noteworthy concerns in this context, due to the fact that the optimization is carried out on the estimated value functions (Li *et al.* 2020).

Instead of using dynamic programming methods, Han (2016), Tsang and Wong (2020) propose to approximate the optimal control function by neural network functions directly. In particular, they propose a stacked neural network approach that essentially uses a sub-network to approximate

† Intuitively, the dimensionality comes from tracking the allocation in the N_a assets for both the active portfolio and benchmark portfolio when evaluating the changes in wealth of both portfolios over one period in the action-value function.

‡ For example, the state space of problem (48) with assumptions of a fixed-mix strategy is a vector in $\mathbb{R}^3$.

the control at every rebalancing step. Therefore, the number of neural networks required grows linearly with the number of rebalancing periods.

It is also worth noting that the Deep Galerkin Method proposed in Sirignano and Spiliopoulos (2018) and the extensions in Al-Aradi *et al.* (2022) use a deep neural network to approximate the value function that satisfies an HJB PDE, which provides a mesh-free approach to avoid the curse-of-dimensionality. However, the PDE requires parametric model assumptions (i.e. posited stochastic differential equations for the underlying processes), and such methods are not applicable in a data-driven setting where no such assumptions are required.

In contrast to the aforementioned work, we follow the lines of Li and Forsyth (2019) and Ni *et al.* (2022) and propose to use a single neural network to approximate the optimal control function. The direct representation of the control function avoids the high-dimensional approximation or error amplification issues encountered in the DP-based methods. In addition, the proposed neural network is global in time, which considers time t as an input feature (along with the wealth of the active portfolio and benchmark portfolio), therefore avoiding the need for multiple sub-networks in the stacked neural network approach.

The numerical solution to the general problem (5) requires solving for the feedback control p. We approximate the control function $p(\cdot)$ with a neural network function $f_{\boldsymbol{\theta}} : \mathcal{X} \mapsto \mathbb{R}^{N_a}$, where $\boldsymbol{\theta} \in \mathbb{R}^{N_\theta}$ represents the parameters of the neural network (i.e. weights and biases). In other words,

$$p(\boldsymbol{X}(t)) \simeq f_{\boldsymbol{\theta}}(\boldsymbol{X}(t)). \tag{17}$$

Then, the optimization problem (5) can be converted to solving the following optimization problem.

(Parameterized optimization problem):

$$\inf_{\boldsymbol{\theta} \in \mathcal{Z}_{\boldsymbol{\theta}}} \mathbb{E}_{f_{\boldsymbol{\theta}}}^{(t_0,w_0)} \left[F(\mathcal{W}_{\boldsymbol{\theta}}, \hat{\mathcal{W}}_{\hat{p}}) \right]. \tag{18}$$

Here $\mathcal{W}_{\boldsymbol{\theta}}$ is the wealth trajectory of the active portfolio following the neural network approximation function parameterized by $\boldsymbol{\theta}$. $\mathcal{Z}_{\boldsymbol{\theta}} \subseteq \mathbb{R}^{N_\theta}$ is the feasibility domain of the parameter $\boldsymbol{\theta}$, which is translated from the constraints of the original problem, e.g. (14) and (15). Mathematically,

$$\mathcal{Z}_{\boldsymbol{\theta}} = \left\{ \boldsymbol{\theta} : \begin{cases} f_{\boldsymbol{\theta}}(\boldsymbol{X}) \in \mathcal{Z}_1, \text{ if } \boldsymbol{X} \in \mathcal{X}_1, \\ f_{\boldsymbol{\theta}}(\boldsymbol{X}) \in \mathcal{Z}_2, \text{ if } \boldsymbol{X} \in \mathcal{X}_2. \end{cases} \right\}. \tag{19}$$

Here $\mathcal{Z}_1, \mathcal{Z}_2$ are defined in (12), (13) and $\mathcal{X}_1, \mathcal{X}_2$ are partitions of the state space $\mathcal{X}$ defined in Definition 2.1.

Note here that $\mathcal{Z}_{\boldsymbol{\theta}}$ depends on the structure of the neural network function $f_{\boldsymbol{\theta}}$. Intuitively, $\mathcal{Z}_{\boldsymbol{\theta}}$ can be thought of as the preimage of $\mathcal{Z}$ under the neural network function. We are interested in the neural network model design that results in $\mathcal{Z}_{\boldsymbol{\theta}} = \mathbb{R}^{N_\theta}$, which means problem (18) becomes an unconstrained optimization problem. For long-only investment problems, the only constraints are the long-only constraint (6) and the summation constraint (7). Previous work has proposed a neural network architecture with a softmax activation function at the last layer so that the output (vector of allocation fractions) automatically satisfies the two constraints, and thus $\mathcal{Z}_{\boldsymbol{\theta}} = \mathbb{R}^{N_\theta}$ and problem (18) becomes an unconstrained optimization problem (see, e.g. Li and Forsyth 2019, Ni *et al.* 2022). However, as discussed in section 2.3, we consider the more complicated case where leverage and shorting are allowed. The problem thus involves more complicated constraints than the long-only case and therefore requires us to design a new model architecture to convert the constrained optimization problem to an unconstrained problem. We will discuss the design of the *leverage-feasible neural network* (LFNN) model in the next section, and how the LFNN model achieves this goal.

3.1. LFNN model

In this section, we propose the leverage-feasible neural network (LFNN) model, which yields $\mathcal{Z}_{\boldsymbol{\theta}} = \mathbb{R}^{N_\theta}$ for leverage constraints defined in equation (14), and therefore converts a constrained optimization problem (18) to an unconstrained problem.

We first define the commonly used fully connected feedforward neural network, e.g. (Lu and Lu 2020), as follows.

Definition 3.1 (Fully connected feedforward neural network $\tilde{f}_{\boldsymbol{\theta}}$) *A fully connected feedforward neural network (FNN) maps an input vector $\boldsymbol{x} \in \mathbb{R}^{d_0}$ to an output vector $\boldsymbol{h} \in \mathbb{R}^{d_{K+1}}$, and contains K hidden layers of sizes $d_1, \ldots, d_K$. The neural network is parameterized by the weight matrices $\boldsymbol{\theta}^{(k)} \in \mathbb{R}^{d_{k-1} \times d_k}$ and bias vectors $\boldsymbol{\theta}_b^{(k)} \in \mathbb{R}^{d_k}$, for $k = 1, \ldots, K+1$. Then, the output $\boldsymbol{h}$ is derived from the input $\boldsymbol{x}$ iteratively as follows.*

$$\begin{cases} \boldsymbol{x}^{(0)} = \boldsymbol{x}, \\ \boldsymbol{x}^{(k)} = \sigma\left(\left(\boldsymbol{\theta}^{(k)}\right)^\top \cdot \boldsymbol{x}^{(k-1)} + \boldsymbol{\theta}_b^{(k)} \right), \quad 1 \le k \le K, \\ \boldsymbol{h} = \left(\boldsymbol{\theta}^{(K+1)}\right)^\top \cdot \boldsymbol{x}^{(K)} + \boldsymbol{\theta}_b^{(K+1)}. \end{cases} \tag{20}$$

Here σ is the pointwise sigmoid activation function, i.e. for any vector $\boldsymbol{z}$, $[\sigma(\boldsymbol{z})]_i = \sigma(\boldsymbol{z}_i)$. For notational simplicity, we flatten and assembly all weight matrices and bias vectors into a single parameter vector $\boldsymbol{\theta} = (\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}_b^{(1)}, \ldots, \boldsymbol{\theta}^{(K+1)}, \boldsymbol{\theta}_b^{(K+1)})^\top \in \mathbb{R}^{N_\theta}$, where $N_\theta = \sum_{k=1}^{K+1} (d_{k-1} \cdot d_k + d_k)$. Furthermore, we use the 2-tuple $(K, (d_1, \ldots, d_K)^\top)$ to denote the hyperparameters, i.e. the number of hidden layers and the size of each hidden layer.

The function defined by the above fully connected feedforward neural network parameterized by $\boldsymbol{\theta}$ is denoted by $\tilde{f}_{\boldsymbol{\theta}}$.

Note that the size of $\boldsymbol{\theta}$ depends on hyperparameters $(K, (d_1, \ldots, d_K)^\top)$. However, for notational simplicity, we omit the 2-tuple in $\tilde{f}_{\boldsymbol{\theta}}$.

Building on $\tilde{f}_{\boldsymbol{\theta}}$, we propose the following *leverage-feasible neural network* (LFNN) model $f_{\boldsymbol{\theta}} : \mathcal{X} \mapsto \mathcal{Z}$:

$$\text{(LFNN)}: \quad f_{\boldsymbol{\theta}}(\boldsymbol{x}) := \psi\left(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{x}\right) \in \mathcal{Z}. \tag{21}$$

Here, $\psi(\cdot)$ is the *leverage-feasible activation function*. For $\boldsymbol{x} \in \mathcal{X}$ and $\boldsymbol{o} = (o_1, \ldots, o_{N_a+1})^\top \in \mathbb{R}^{N_a+1}$, let $\boldsymbol{y} =$

$(y_1,\ldots,y_{N_a}) = \psi(\boldsymbol{o},\boldsymbol{x})$, where $\psi(\cdot):\mathbb{R}^{N_a+1}\times\mathcal{X}\mapsto\mathcal{Z}$ is defined as follows.

$$\boldsymbol{y} = \psi(\boldsymbol{o},\boldsymbol{x}) = \begin{cases} \begin{cases} l = p_{\max}\cdot\sigma(o_{N_a+1}), \\ y_i = l \\ \quad \times \frac{e^{o_i}}{\sum_{k=1}^{N_L} e^{o_k}}, \quad i\in\{1,\ldots,N_L\}, \\ y_i = (1-l) \\ \quad \times \frac{e^{o_i}}{\sum_{k=N_L+1}^{N_a} e^{o_k}}, \quad i\in\{N_L{+}1,\ldots,N_a\}, \end{cases} & \text{if } \boldsymbol{x}\in\mathcal{X}_1, \\ \boldsymbol{e}_{N_L+1}, & \text{if } \boldsymbol{x}\in\mathcal{X}_2. \end{cases} \tag{22}$$

Here σ is the sigmoid function, N_L is the number of long-only assets, $\boldsymbol{e}_{N_L+1}$ is the standard basis vector with the (N_L+1)th entry being 1 and other entries being 0, and $p_{\max}$ is the maximum leverage allowed. We show that the leverage-feasible activation function ψ has the following property.

LEMMA 3.1 (Decomposition of ψ) *The leverage-feasible function ψ defined in* (22) *has the function decomposition that*

$$\psi(\boldsymbol{o},\boldsymbol{x}) = \varphi(\zeta(\boldsymbol{o}),\boldsymbol{x}), \tag{23}$$

where

$$\begin{cases} \zeta:\mathbb{R}^{N_a+1}\mapsto\tilde{\mathcal{Z}}, \zeta(\boldsymbol{o}) = \big(\mathit{Softmax}\big((o_1,\ldots,o_{N_L})\big), \\ \quad \mathit{Softmax}\big((o_{N_L+1},\ldots,o_{N_a})\big), p_{\max}\cdot\sigma(o_{N_a+1})\big)^\top, \\ \varphi:\tilde{\mathcal{Z}}\times\mathcal{X}\mapsto\mathcal{Z}, \varphi(\boldsymbol{z},\boldsymbol{x}) = \big(z_{N_a+1}\cdot(z_1,\ldots,z_{N_L}), \\ \quad (1-z_{N_a+1})\cdot(z_{N_L+1},\ldots,z_{N_a})\big)^\top\cdot\mathbf{1}_{\boldsymbol{x}\in\mathcal{X}_1} \\ \quad + \boldsymbol{e}_{N_L+1}\cdot\mathbf{1}_{\boldsymbol{x}\in\mathcal{X}_2}, \end{cases} \tag{24}$$

and

$$\tilde{\mathcal{Z}} = \Big\{\boldsymbol{z} = (z_1,\ldots,z_{N_a+1})^\top\in\mathbb{R}^{N_a+1}, \sum_{i=1}^{N_L} z_i = 1, \sum_{i=N_L+1}^{N_a} z_i = 1, z_{N_a+1}\le p_{\max}, z_i\ge 0, \forall i\Big\}. \tag{25}$$

Proof This is easily verifiable by the definition of ψ in (22). ■

REMARK 3.1 (Remark on Lemma 3.1) The leverage-feasible activation function ψ corresponds to a two-step decision process described by ζ and φ. Intuitively, ζ first determines the internal allocations within long-only assets and shortable assets, as well as the total leverage. Then, φ converts the internal allocations and total leverage into final allocation fractions, which depend on the wealth of the active portfolio.

With the LFNN model outlined above, we can show that the parameterized optimization problem (18) becomes an unconstrained optimization problem. Specifically, we present the following theorem regarding the feasibility domain $\mathcal{Z}_{\boldsymbol{\theta}}$ associated with the LFNN model (21).

THEOREM 3.1 (Unconstrained feasibility domain) *The feasibility domain $\mathcal{Z}_{\boldsymbol{\theta}}$ defined in* (19) *associated with the LFNN model* (21) *is* $\mathbb{R}^{N_{\boldsymbol{\theta}}}$.

Proof See Appendix A.1. ■

Following Theorem 3.1, the constrained optimization problem (5) can be transformed into the following unconstrained optimization problem

(Unconstrained parameterized problem):

$$\inf_{\boldsymbol{\theta}\in\mathbb{R}^{N_{\boldsymbol{\theta}}}} \mathbb{E}_{f_{\boldsymbol{\theta}}}^{(t_0,w_0)}\Big[F(\mathcal{W}_{\boldsymbol{\theta}},\hat{\mathcal{W}}_{\hat{p}})\Big]. \tag{26}$$

To solve this problem computationally, we can simulate realizations of the random trajectory, either from simulations of a parametric stochastic model or a block-resampling technique. Using return trajectories, we solve a multi-period scenario stochastic optimization problem directly using a NN to represent a control. When solving a multi-stage stochastic optimization problem based on stochastic programming, non-anticipatory constraints typically need to be imposed to ensure a decision is implementable. In the proposed NN framework, a control is immediately non-anticipatory since it is a function of time and the realized wealth (of both the controlled and benchmark portfolios) at the decision time. In addition, overfitting is avoided by choosing a parsimonious NN structure which gives comparable testing performance to training performance.

3.2. *Mathematical justification for LFNN approach*

By approximating the feasible control with a parameterized LFNN model, we have shown that the original constrained optimization problem is transformed into an unconstrained optimization problem, which is computationally more feasible.

However, an important question remains: is the solution to the parameterized unconstrained optimization problem (26) capable of yielding the optimal control of the original problem (5)? In other words, suppose $\boldsymbol{\theta}^*$ is the solution to (26), can $f_{\boldsymbol{\theta}^*}$ approximate solution to the original problem (5) with high accuracy?

In this section, we prove that under benign assumptions about the optimal control and appropriate choices of the hyperparameters of the LFNN model (21), solving the unconstrained problem (26) provides an arbitrarily close approximation to solving the original problem (5). We start by establishing the following Lemma.

LEMMA 3.2 (Structure of feasible control) *Any feasible control function $p:\mathcal{X}\mapsto\mathcal{Z}$, where $\mathcal{Z}$ is defined in* (15)*, has the following function decomposition*

$$p(\boldsymbol{x}) = \varphi(\omega(\boldsymbol{x}),\boldsymbol{x}), \quad \forall\boldsymbol{x}\in\mathcal{X}, \tag{27}$$

where $\varphi:\tilde{\mathcal{Z}}\times\mathcal{X}\mapsto\mathcal{Z}$ is defined in (24) *and $\omega:\mathcal{X}\mapsto\tilde{\mathcal{Z}}$.*

Proof See Appendix A.2. ■

Next, we propose the following benign assumptions on the state space and the optimal control.

ASSUMPTION 3.1 (Assumption on state space and optimal control)

(i) The space $\mathcal{X}$ of state variables is a compact set.
(ii) Following Lemma 3.2, for any $\boldsymbol{x} \in \mathcal{X}$, the optimal control $p^* : \mathcal{X} \mapsto \mathcal{Z}$ has the decomposition $p^*(\boldsymbol{x}) = \varphi(\omega^*(\boldsymbol{x}), \boldsymbol{x})$ for some $\omega^* : \mathcal{X} \mapsto \tilde{\mathcal{Z}}$. We assume $\omega^* \in C(\mathcal{X}, \tilde{\mathcal{Z}})$, where $C(\mathcal{X}, \tilde{\mathcal{Z}})$ denotes the set of continuous mappings from $\mathcal{X}$ to $\tilde{\mathcal{Z}}$.

REMARK 3.2 (Remark on Assumption 3.1) In our particular problem of outperforming a benchmark portfolio, the state variable vector is $(t, W(t), \hat{W}(t))^\top \in \mathcal{X}$ where $t \in [0, T]$. In this case, assumption (i) is equivalent to the assumption that the wealth of the active portfolio and benchmark portfolio is bounded, i.e. $\mathcal{X} = [0, T] \times [w_{\min}, w_{\max}] \times [\hat{w}_{\min}, \hat{w}_{\max}]$, where $w_{\min}, w_{\max}$ and $\hat{w}_{\min}, \hat{w}_{\max}$ are the respective wealth bounds for the portfolios. Intuitively, assumption (ii) states that the decision process for the optimal control to obtain the allocation fractions within the long-only assets and the shortable assets, and the total leverage, is a continuous function of state variables. This is a natural extension of the long-only case, in which it is commonly assumed that the allocation within long-only assets is a continuous function of state variables.

Before presenting the approximation theorem, we first briefly review the results of Kratsios and Bilokopytov (2020).

LEMMA 3.3 *Let $\mathcal{X} \subset \mathbb{R}^l$ be a compact set, and $\mathcal{Y} \subset \mathbb{R}^m$. Let $\rho : \mathbb{R}^n \mapsto \mathcal{Y}$ satisfy the following:*

(i) *ρ is continuous and has a right inverse on $Im(\rho)$, i.e. $\exists \overrightarrow{\rho} : Im(\rho) \mapsto \mathbb{R}^n$, s.t. $\rho(\overrightarrow{\rho}(z)) = z$, $\forall z \in Im(\rho)$.*
(ii) *$Im(\rho)$ is dense in $\mathcal{Y}$.*

Then, for any continuous $g : \mathcal{X} \mapsto \mathcal{Y}$, and any $\epsilon > 0$, there exists a choice of hyperparameters $(K, (d_1, \ldots, d_K)^\top)$ and parameter $\boldsymbol{\theta}$, such that the corresponding FNN $\tilde{f}_{\boldsymbol{\theta}} : \mathcal{X} \mapsto \mathbb{R}^n$ described in Definition 3.1 satisfies

$$\sup_{\boldsymbol{x} \in \mathcal{X}} \| \rho(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})) - g(\boldsymbol{x}) \| < \epsilon, \quad \forall \boldsymbol{x} \in \mathcal{X}. \tag{28}$$

Here $\| \cdot \|$ denotes the vector norm.

Proof This is a direct application of Theorem 3.3 of Kratsios and Bilokopytov (2020) (for general topological spaces) in the metric space. ■

Intuitively, the second assumption of Lemma 3.3 allows the use of an activation function (such as the softmax function) whose output forms an open set, as long as this open set is dense in $\mathcal{Y}$ (which can be a closed set). The two assumptions ensure the existence of a continuous mapping of which the image almost covers $\mathcal{Y}$.

We describe the following Lemma, which is a generalization of Lemma 3.3.

LEMMA 3.4 (Approximation of controls with a specific structure) *Assume a control function $p : \mathcal{X} \mapsto \mathcal{Z}$ has the structure*

$$p(\boldsymbol{x}) = \Phi(\Omega(\boldsymbol{x}), \boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{X}, \tag{29}$$

where $\mathcal{X}$ is compact, $\Omega \in C(\mathcal{X}, \mathcal{Y})$, i.e. Ω is a continuous mapping from $\mathcal{X}$ to $\mathcal{Y}$, and $\Phi : \mathcal{Y} \times \mathcal{X} \mapsto \mathcal{Z}$ is Lipschitz continuous on $\mathcal{Y} \times \mathcal{X}_i$, $\forall i = 1, \ldots, n$, where $\{\mathcal{X}_i, i = 1, \ldots, n\}$ is a partition of $\mathcal{X}$, i.e.

$$\begin{cases} \bigcup_{i=1}^n \mathcal{X}_i = \mathcal{X}, \\ \mathcal{X}_i \bigcap \mathcal{X}_j = \varnothing, \quad \forall 1 \le i, j \le n. \end{cases} \tag{30}$$

If there exists $m \in \mathbb{N}$ and $\Upsilon : \mathbb{R}^m \mapsto \mathcal{Y}$ such that

(i) *Υ is continuous and has a right inverse on $Im(\Upsilon)$.*
(ii) *$Im(\Upsilon)$ is dense in $\mathcal{Y}$.*

Then for any $\epsilon > 0$, there exists a choice of hyperparameters $(K, (d_1, \ldots, d_K)^\top)$ and model parameters $\boldsymbol{\theta}$ such that the fully connected feedforward neural network function $\tilde{f}_{\boldsymbol{\theta}}$ defined in Definition 3.1 satisfies

$$\sup_{\boldsymbol{x} \in \mathcal{X}} \| \Phi\left(\Upsilon\left(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})\right), \boldsymbol{x}\right) - p(\boldsymbol{x}) \| < \epsilon. \tag{31}$$

Proof Let

$$L_\Phi = \max_{1 \le i \le n} L_i, \tag{32}$$

where L_i is the Lipschitz constant for Φ on $\mathcal{Y} \times \mathcal{X}_i$.

Since $\Omega \in C(\mathcal{X})$ for compact $\mathcal{X}$, following Lemma 3.3, we know that $\forall \epsilon > 0$, there exist $(K, (d_1, \ldots, d_K)^\top)$ (the number of hidden layers and hidden nodes for each layer) and $\boldsymbol{\theta} \in \mathbb{R}^{N_{\boldsymbol{\theta}}}$ such that the corresponding FNN $\tilde{f}_{\boldsymbol{\theta}} : \mathcal{X} \mapsto \mathbb{R}^m$ described in Definition 3.1 satisfies

$$\sup_{\boldsymbol{x} \in \mathcal{X}} \| \Upsilon\left(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})\right) - \Omega(\boldsymbol{x}) \| < \epsilon / L_\Phi, \tag{33}$$

Then

$$\begin{aligned} &\sup_{\boldsymbol{x} \in \mathcal{X}} \| \Phi\left(\Upsilon\left(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})\right), \boldsymbol{x}\right) - p(\boldsymbol{x}) \| \\ &= \sup_{1 \le i \le n} \sup_{\boldsymbol{x} \in \mathcal{X}_i} \| \Phi \times \left(\Upsilon\left(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})\right), \boldsymbol{x}\right) - \Phi\left(\Omega(\boldsymbol{x}), \boldsymbol{x}\right) \| && (34) \\ &\le \sup_{1 \le i \le n} \sup_{\boldsymbol{x} \in \mathcal{X}_i} L_i \cdot \left(\| \Upsilon\left(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})\right) - \Omega(\boldsymbol{x}) \| \right) && (35) \\ &< \sup_{1 \le i \le n} \frac{L_i}{L_\Phi} \epsilon && (36) \\ &\le \epsilon. && (37) \end{aligned}$$

■

REMARK 3.3 (Application of Lemma 3.4 to portfolios with stochastic allocation constraints) Normally, the universal approximation theorem only applies to the approximation of continuous functions defined on a compact set (Hornik 1991). Lemma 3.4 extends the universal approximation theorem to a broader class of functions that have the structure of (29).

Furthermore, Lemma 3.4 provides guidance on constructing neural network functions that handle stochastic constraints on controls which are usually difficult to address in stochastic optimal control problems. Consider the following example: the control $p : \mathcal{X} \mapsto \mathbb{R}^{N_a}$ has stochastic constraints such that $p(\boldsymbol{x}) \in [a(\boldsymbol{x}), b(\boldsymbol{x})]$ where $a, b : \mathcal{X} \mapsto \mathbb{R}^{N_a}$ are deterministic functions. This is a common setting in portfolio optimization problems in which allocation fractions to specific assets are subject to thresholds tied to the performance of the portfolio (which is stochastic). With Lemma 3.4, by parameterizing the control $p(x) \approx f_{\boldsymbol{\theta}}(\boldsymbol{x}) = a(\boldsymbol{x}) + \sigma(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})) \cdot (b(\boldsymbol{x}) - a(\boldsymbol{x}))$, where $\sigma(\cdot)$ is the sigmoid function and $\tilde{f}$ is a standard FNN defined in Definition 3.1, one can easily construct the corresponding neural network that satisfies the constraints naturally and be guaranteed that such a neural network can approximate the control well.

Finally, we present the following approximation theorem for the LFNN.

THEOREM 3.2 (Approximation of optimal control) *Given the optimal control p^* of problem* (5) *and Assumption 3.1, $\forall \epsilon > 0$, there exist hyperparameters $(K, (d_1, \ldots, d_K)^\top)$, and $\boldsymbol{\theta} \in \mathbb{R}^{N_{\boldsymbol{\theta}}}$ such that the corresponding LFNN model $f_{\boldsymbol{\theta}}$ described in* (21) *satisfies the following:*

$$\sup_{\boldsymbol{x} \in \mathcal{X}} \|f_{\boldsymbol{\theta}}(\boldsymbol{x}) - p^*(\boldsymbol{x})\| < \epsilon. \tag{38}$$

Proof From (21) and Lemma 3.1, we know that

$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \psi\left(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{x}\right) = \varphi\left(\zeta\left(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})\right), \boldsymbol{x}\right), \tag{39}$$

where $\tilde{f}$ is the FNN defined in Definition 3.1 and $\varphi : \tilde{\mathcal{Z}} \times \mathcal{X} \mapsto \mathbb{R}^{N_a}, \zeta : \mathbb{R}^{N_a+1} \mapsto \tilde{\mathcal{Z}}$ are defined in (24).

It can be easily verified that ζ satisfies the following:

(i) ζ is continuous and has a right inverse, e.g.

$$\begin{aligned} &\zeta^{-1}(z) : Im(\zeta) \mapsto \mathbb{R}^{N_a+1}, \zeta^{-1}(z) \\ &\quad = \left(\log(z_1), \ldots, \log(z_{N_a}), \sigma^{-1}(z_{N_a+1}/p_{\max})\right)^\top, \end{aligned} \tag{40}$$

where σ^{-1} is the inverse function of the sigmoid function.

(ii) $Im(\zeta)$ is dense in $\tilde{\mathcal{Z}}$. This is because $\overline{Im(\zeta)}$, the closure of $Im(\zeta)$, is $\tilde{\mathcal{Z}}$.

Furthermore, consider the partition of $\mathcal{X}$, $\{\mathcal{X}_1, \mathcal{X}_2\}$, which is defined in Definition 2.1. It is easily verifiable that φ is Lipschitz continuous on $\tilde{\mathcal{Z}} \times \mathcal{X}_1$ and $\tilde{\mathcal{Z}} \times \mathcal{X}_2$ respectively.

Finally, according to Assumption 3.1, $p^*(\boldsymbol{x}) = \varphi(\omega^*(\boldsymbol{x}), \boldsymbol{x})$, where $\omega^* \in C(\mathcal{X}, \tilde{\mathcal{Z}})$.

Applying Lemma 3.4 with $\mathcal{Y} = \tilde{\mathcal{Z}}, \Omega(\cdot) = \omega^*(\cdot)$, $\Upsilon(\cdot) = \zeta(\cdot)$, and $\Phi(\cdot, \cdot) = \varphi(\cdot, \cdot)$, we know that there exist hyperparameters $(K, (d_1, \ldots, d_K)^\top)$, and model parameters $\boldsymbol{\theta} \in \mathbb{R}^{N_{\boldsymbol{\theta}}}$ such that the corresponding LFNN model $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \varphi(\zeta(\tilde{f}_{\boldsymbol{\theta}}(\boldsymbol{x})), \boldsymbol{x})$ satisfies the following:

$$\sup_{\boldsymbol{x} \in \mathcal{X}} \|f_{\boldsymbol{\theta}}(\boldsymbol{x}) - p^*(\boldsymbol{x})\| < \epsilon. \tag{41}$$

■

Theorem 3.2 shows that given any arbitrarily small tolerance $\epsilon > 0$, there exists a suitable choice of the hyperparameter of the LFNN model (e.g. the number of hidden layers and nodes) and a parameter vector $\boldsymbol{\theta}$, such that the corresponding parameterized LFNN function is within this tolerance of the optimal control function.† In other words, with a large enough LFNN model (in terms of the number of hidden nodes), solving the unconstrained parameterized problem (26) approximately solves the original optimization problem (5) with any required precision.

3.3. *Training LFNN*

Since the numerical experiments involve the solution and evaluation of the optimal parameters $\boldsymbol{\theta}^*$ of the LFNN model (21) in problem (26), we briefly review how the parameters are computed in experiments.

In numerical experiments, the expectation in (26) is approximated by using a finite set of samples of the set $\boldsymbol{Y} = \{Y^{(j)} : j = 1, \ldots, N_d\}$, where N_d is the number of samples, and $Y^{(j)}$ represents a time series sample of *joint* asset return observations $R_i(t)$, $i \in \{1, \ldots, N_a\}$, observed at $t \in \mathcal{T}$, where $\mathcal{T}$ is the rebalancing schedule.‡ Mathematically, problem (26) is approximated by

$$\inf_{\boldsymbol{\theta} \in \mathbb{R}^{N_{\boldsymbol{\theta}}}} \left\{ \frac{1}{N_d} \sum_{j=1}^{N_d} F\left(\mathcal{W}_{\boldsymbol{\theta}}^{(j)}, \hat{\mathcal{W}}_{\hat{p}}^{(j)}\right) \right\}. \tag{42}$$

Here $\mathcal{W}_{\boldsymbol{\theta}}^{(j)}$ is the wealth trajectory of the active portfolio following the LFNN parameterized by $\boldsymbol{\theta}$, and $\hat{\mathcal{W}}_{\hat{p}}^{(j)}$ is the wealth trajectory of the benchmark portfolio following the benchmark strategy $\hat{p}$, both evaluated on $Y^{(j)}$, the jth time series sample.

We use a shallow neural network model, specifically, an LFNN model with one single hidden layer with 10 hidden nodes, i.e. $K = 1$ and $d_1 = 10$. We use the 3-tuple vector $(t, W_{\boldsymbol{\theta}}(t), \hat{W}(t))^\top$ as the input (feature) to the LFNN network. At $t \in [t_0, T]$, $W_{\boldsymbol{\theta}}(t)$ is the wealth of the active portfolio of the strategy that follows the LFNN model parameterized by $\boldsymbol{\theta}$, and $\hat{W}(t)$ is the wealth of the benchmark portfolio.

An important computational advantage of the proposed neural network framework is its capability to directly compute model parameters using gradient descent-based methods. In essence, the model functions as a recurrent neural network (RNN), and the procedure for calculating the gradient for a specific path (indexed as j) is outlined as follows.

$$\nabla_{\boldsymbol{\theta}} F\left(\mathcal{W}_{\boldsymbol{\theta}}^{(j)}, \hat{\mathcal{W}}_{\hat{p}}^{(j)}\right) = \sum_{i=1}^{N} \frac{\partial F}{\partial W_{\boldsymbol{\theta}}^{(j)}(t_i)} \nabla_{\boldsymbol{\theta}} W_{\boldsymbol{\theta}}^{(j)}(t_i). \tag{43}$$

Let $\boldsymbol{R}(t_i) = (R_1(t_i), \ldots, R_1(t_i))^\top \in \mathbb{R}^{N_a}$ denote the return vector in $[t_{i-1}, t_i]$, then, the wealth dynamics for the value of the

† The distance is defined in (38), i.e. the supremum of the pointwise distance over the extended state space $\mathcal{X}$.

‡ Note that the corresponding set of asset prices can be easily inferred from the set of asset returns, or vice versa.

active portfolio described in (4) can be summarized as

$$W_{\boldsymbol{\theta}}^{(j)}(t_i) = f_{\boldsymbol{\theta}}\left(W_{\boldsymbol{\theta}}^{(j)}(t_{i-1}), \hat{W}^{(j)}(t_{i-1}), t_{i-1}\right)^{\top} \times (1 + \boldsymbol{R}(t_i))\, W_{\boldsymbol{\theta}}^{(j)}(t_{i-1}), \tag{44}$$

where $f_{\boldsymbol{\theta}}$ is the LFNN parameterized by $\boldsymbol{\theta}$.

Note that $\nabla_{\boldsymbol{\theta}} W_{\boldsymbol{\theta}}^{(j)}(t_0) = 0$, since the initial portfolio wealth is a constant value. Then, for any $i \in \{1, \ldots, N\}$, the gradients $\nabla_{\boldsymbol{\theta}} W_{\boldsymbol{\theta}}^{(j)}(t_i)$ in (43) can be obtained recursively using the chain rule, i.e.

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} W_{\boldsymbol{\theta}}^{(j)}(t_i) &= \nabla_{\boldsymbol{\theta}}\left(f_{\boldsymbol{\theta}}\left(W_{\boldsymbol{\theta}}^{(j)}(t_{i-1}), \hat{W}^{(j)}(t_{i-1}), t_{i-1}\right)^{\top} \times (1 + \boldsymbol{R}(t_i))\, W_{\boldsymbol{\theta}}^{(j)}(t_{i-1})\right) && (45)\\
&= \nabla_{\boldsymbol{\theta}}\left(f_{\boldsymbol{\theta}}\left(W_{\boldsymbol{\theta}}^{(j)}(t_{i-1}), \hat{W}^{(j)}(t_{i-1}), t_{i-1}\right)^{\top} \times (1 + \boldsymbol{R}(t_i)))\, W_{\boldsymbol{\theta}}^{(j)}(t_{i-1})\right.\\
&\quad \left. + \left(f_{\boldsymbol{\theta}}\left(W_{\boldsymbol{\theta}}^{(j)}(t_{i-1}), \hat{W}^{(j)}(t_{i-1}), t_{i-1}\right)^{\top} \times (1 + \boldsymbol{R}(t_i)))\, \nabla_{\boldsymbol{\theta}} W_{\boldsymbol{\theta}}^{(j)}(t_{i-1})\right. \right) && (46)\\
&= \left(\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}\left(W_{\boldsymbol{\theta}}^{(j)}(t_{i-1}), \hat{W}^{(j)}(t_{i-1}), t_{i-1}\right)\right) \times (1 + \boldsymbol{R}(t_i))\, W_{\boldsymbol{\theta}}^{(j)}(t_{i-1})\\
&\quad + \left(\frac{\partial f_{\boldsymbol{\theta}}(W_{\boldsymbol{\theta}}^{(j)}(t_{i-1}), \hat{W}^{(j)}(t_{i-1}), t_{i-1}}{\partial W_{\boldsymbol{\theta}}^{(j)}(t_{i-1})}\right)^{\top} \times (1 + \boldsymbol{R}(t_i))\, \nabla_{\boldsymbol{\theta}} W_{\boldsymbol{\theta}}^{(j)}(t_{i-1})\\
&\quad + \left(f_{\boldsymbol{\theta}}\left(W_{\boldsymbol{\theta}}^{(j)}(t_{i-1}), \hat{W}^{(j)}(t_{i-1}), t_{i-1}\right)^{\top} \times (1 + \boldsymbol{R}(t_i)))\, \nabla_{\boldsymbol{\theta}} W_{\boldsymbol{\theta}}^{(j)}(t_{i-1})\right. && (47)
\end{aligned}$$

Then, the optimal parameter $\boldsymbol{\theta}^*$ can be numerically obtained by numerically solving problem (42) using standard optimization algorithms such as ADAM (Kingma and Ba 2014). This process is commonly referred to as 'training' of the neural network model, and $\boldsymbol{Y}$ is often referred to as the training data set (Goodfellow *et al.* 2016). Computationally, we use the PyTorch framework (Paszke *et al.* 2017) to conduct the training of the neural network.

Note that the definition of the leverage-feasible activation function φ in Equation (24) depends on non-differentiable step functions $\mathbf{1}_{\boldsymbol{x}\in\mathcal{X}_1}$ and $\mathbf{1}_{\boldsymbol{x}\in\mathcal{X}_2}$. It can be easily verified that φ is piece-wise analytic under analytic partition (in short, PAP), which has the property of being differentiable almost everywhere (Lee *et al.* 2020), i.e. the set of inputs making the function non-differentiable is contained in a measure-zero set. Due to the recurrent nature of the neural network model, the objective function is essentially a chaining composition of φ and smooth functions (i.e. the wealth evolution function, sigmoid function, and affine function). According to Lee *et al.* (2020), the objective function is also PAP, thus is differentiable almost everywhere. In rare cases when the function is not differentiable, PyTorch computes either gradient or sub-gradient to allow optimization to proceed.†

In our specific case of portfolio optimization with leverage constraints, the non-differentiability only occurs when the wealth of the portfolio reaches zero at any time during the investment horizon. However, in numerical experiments, the wealth has not reached zero in any of the paths, most likely due to the mildly capped total leverage. Therefore, concerns about the non-differentiability of the objective function should be further alleviated.

Once $\boldsymbol{\theta}^*$ is numerically obtained, the resulting optimal strategy $f_{\boldsymbol{\theta}^*}$ is evaluated on a separate 'testing' data set $\boldsymbol{Y}^{test}$, which contains a different set of samples generated from either the same distribution of the training process or a different process (depending on experiment purposes) so that the 'out-of-sample' performance of $f_{\boldsymbol{\theta}^*}$ is assessed.

4. Accuracy assessment of LFNN

In Theorem 3.2, we prove that the LFNN can approximate the optimal control arbitrarily closely. However, does such an approximation require a very deep neural network structure with a large amount of hidden layers and nodes? What degree of accuracy can solutions from LFNN achieve?

To answer these questions, in this section, we assume the prices of the stock index and the bond index follow a double exponential jump-diffusion model and we mathematically derive an analytic solution to the stochastic optimal control problem (5) with no constraints on the strategy. We compare performance measures (e.g. in terms of objective function values) achieved by the analytic solution, which permits infinite leverage, with the solution computed from LFNN under realistic leverage constraints. We further derive an easily computed clipped solution (which is feasible but sub-optimal) based on the analytic solution which satisfies leverage constraints. We show empirically that a shallow LFNN model is able to achieve comparable performance against a closed-form solution and the clipped solution in this case. We note that the jump diffusion model is estimated from the historical high-inflation periods (which we will discuss in more detail in section 5.1).

4.1. *Closed-form solution*

We first present the closed-form solution under the following cumulative quadratic tracking difference (CD) objectives proposed in van Staden *et al.* (2024):

$$(CD(\beta)): \quad \inf_{p\in\mathcal{A}} \mathbb{E}_p^{(t_0,w_0)}\left[\int_{t_0}^{T}\left(W(t) - e^{\beta t}\hat{W}(t)\right)^2 \mathrm{d}t\right], \tag{48}$$

Here, β is the annualized performance premium rate, and $e^{\beta t}\hat{W}(t)$ is the elevated target proposed in Ni *et al.* (2022). The CD objective (48) measures the cumulative deviation of the wealth of the active portfolio relative to the elevated

† See Pytorch Documentation.

target, along the rebalancing schedule throughout the entire investment horizon.

The closed-form solution is derived under several assumptions. It not only provides us with insights for understanding the CD-optimal controls for problem (48), but also serves as the baseline for understanding the numerical results derived from the neural network strategy. Specifically, in this section, we consider the case that all asset prices follow jump-diffusion processes and portfolios with cash injections, which are aspects not frequently considered in benchmark outperformance literature (Browne 1999, 2000, Tepla 2001, Basak *et al.* 2006, Yao *et al.* 2006, Zhao 2007, Davis and Lleo 2008, Lim and Wong 2010b, Oderda 2015, Zhang and Gao 2017, Al-Aradi and Jaimungal 2018, Nicolosi *et al.* 2018, Bo *et al.* 2021).

We first summarize the assumptions for obtaining the closed-form solution to CD problem (48).

Assumption 4.1 (Two assets, no friction, unlimited leverage, trading in insolvency, constant rate of cash injection) The active portfolio and the benchmark portfolio have access to two underlying assets, a stock index, and a constant-maturity bond index. Both portfolios are rebalanced continuously, i.e. $\mathcal{T} = [t_0, T]$. There is no transaction cost and no leverage limit. Furthermore, we assume that trading continues in the event of insolvency, i.e. when $W(t) < 0$ for some $t \in [t_0, T]$. Finally, we assume both portfolios receive constant cash injections with an injection rate of c, which means during any time interval $[t, t + \Delta t] \subseteq [t_0, T], \forall \Delta t > 0$, both portfolios receive cash injection amount of $c\Delta t$.

Remark 4.1 (Remark on Assumption 4.1) For illustration purposes, we assume only two underlying assets. However, the technique for deriving the closed-form solution can be extended to multiple assets. We remark that unlimited leverage is unrealistic, and is only assumed for deriving the closed-form solution. In section 4.2, we will discuss the technique for handling the leverage constraint in more detail. We also acknowledge that it is not realistic to assume that the manager can continue to trade and borrow when insolvent. However, this assumption is typically required for obtaining closed-form solutions, see Zhou and Li (2000) and Li and Ng (2000) for the case of a multi-period mean-variance asset allocation problem.

Assumption 4.2 (Fixed-mix benchmark strategy) We assume that the benchmark strategy is a fixed-mix strategy (also known as the constant weight or constant proportion strategy). We assume the benchmark always allocates a constant fraction of $\hat{\varrho}$ ($\in \mathbb{R}$) of the portfolio wealth to the stock index, and a constant fraction of $1 - \hat{\varrho}$ to the bond index. Let $\hat{\boldsymbol{\varrho}} = (\hat{\varrho}, 1 - \hat{\varrho})^\top \in \mathbb{R}^2$ denote the vector of allocation fractions to the stock index and the bond index.

Finally, we assume the stock index price and bond index price follow the jump-diffusion processes described below.

Assumption 4.3 (Jump-diffusion processes) Let $S_1(t)$ and $S_2(t)$ denote the deflated (adjusted by inflation) price of the stock index and the constant maturity bond index at time $t \in [t_0, T]$. We assume $S_i(t)$, $i \in \{1, 2\}$ follow the jump-diffusion processes

$$\frac{\mathrm{d}S_i(t)}{S_i(t^-)} = (\mu_i - \lambda_i \kappa_i)\mathrm{d}t + \sigma_i \, \mathrm{d}Z_i(t) + d\left(\sum_{k=1}^{\pi_i(t)} (\xi_i^{(k)} - 1)\right), \quad i = 1, 2. \tag{49}$$

Here μ_i are the (uncompensated) drift rate, σ_i is the diffusive volatility, $Z_1(t), Z_2(t)$ are correlated Brownian motions, where $\mathrm{d}Z_1(t) \cdot \mathrm{d}Z_2(t) = \rho \, \mathrm{d}t$. $\pi_i(t)$ is a Poisson process with positive intensity parameter λ_i. $\{\xi_i^{(k)}, k = 1, \ldots, \pi_i(t)\}$ are i.i.d. positive random variables that describe jump multipliers associated with the assets. If a jump occurs for asset i at time $t \in (t_0, T]$, its underlying price jumps from $S_i(t^-)$ to $S_i(t) = \xi_i \cdot S_i(t^-)$.† $\kappa_i = \mathbb{E}[\xi_i - 1]$. ξ_i and $\pi_i(t)$ are independent of each other. Moreover, $\pi_1(t)$ and $\pi_2(t)$ are assumed to be independent.‡

Remark 4.2 (Motivation for jump-diffusion model) The assumption of stock index price following a jump-diffusion model is common in the financial mathematics literature (Merton 1976, Kou 2002). In addition, we follow the practitioner approach and directly model the returns of the constant maturity bond index as a stochastic process, see for example Lin *et al.* (2015) and MacMinn *et al.* (2014). As in MacMinn *et al.* (2014), we also assume that the constant maturity bond index follows a jump-diffusion process. During high-inflation regimes, central banks often make rate hikes to curb inflation, which causes sudden jumps in bond prices (Lahaye *et al.* 2011). We believe this is an appropriate assumption for bonds in high-inflation regimes.

Under the jump-diffusion model (49), the wealth processes for the active portfolio and benchmark portfolio are

$$\begin{cases} \mathrm{d}W(t) = \left(\sum_{i=1}^{N_a} p_i(\boldsymbol{X}(t^-)) \cdot \dfrac{\mathrm{d}S_i(t)}{S_i(t^-)}\right) W(t_j^-) + c \, \mathrm{d}t, \\ \mathrm{d}\hat{W}(t) = \left(\sum_{i=1}^{N_a} \hat{p}_i(\hat{\boldsymbol{X}}(t^-)) \cdot \dfrac{\mathrm{d}S_i(t)}{S_i(t^-)}\right) \hat{W}(t_j^-) + c \, \mathrm{d}t, \end{cases} \tag{50}$$

where $t \in (t_0, T]$, $W(t_0) = \hat{W}(t_0) = w_0$ and $X(t^-) = (t, W(t^-), \hat{W}(t^-))^\top \in \mathbb{R}^3$ is the state variable vector.

We now derive the closed-form solution of the CD problem (48) under Assumptions 4.1, 4.2 and 4.3. We first present the verification theorem for the HJB integro-differential equation (PIDE) satisfied by the value function and the optimal control of the CD problem (48).

Theorem 4.1 (Verification theorem for CD problem (48)) *For a fixed $\beta > 0$, assume that for all $(t, w, \hat{w}, \hat{\varrho}) \in [t_0, T] \times \mathbb{R}^3$, there exist functions $V(t, w, \hat{w}, \hat{\varrho}) : [t_0, T] \times \mathbb{R}^3 \mapsto \mathbb{R}$ and $p^*(t, w, \hat{w}, \hat{\varrho}) : [t_0, T] \times \mathbb{R}^3 \mapsto \mathbb{R}^2$ that satisfy the following*

† For any functional $\psi(t)$, we use the notation $\psi(t^-)$ as shorthand for the left-sided limit $\psi(t^-) = \lim_{\Delta t \downarrow 0} \psi(t - \Delta t)$.

‡ See Forsyth (2020) for the discussion on the empirical evidence for stock-bond jump independence. Also note that the assumption of independent jumps can be relaxed without technical difficulty if needed (Kou 2002), but will significantly increase the complexity of notations.

two properties. (i) V and $\boldsymbol{p}^*$ are sufficiently smooth and solve the HJB PIDE (51)*, and (ii) the function $\boldsymbol{p}^*(t,w,\hat{w},\hat{\varrho})$ attains the pointwise infimum in* (51) *below*

$$\begin{cases} \frac{\partial V}{\partial t} + (w - e^{\beta t}\hat{w})^2 + \inf_{\boldsymbol{p}\in\mathbb{R}^2} H(\boldsymbol{p};t,w,\hat{w},\hat{\boldsymbol{\varrho}}) = 0, \\ V(T,w,\hat{w},\hat{\varrho}) = 0, \end{cases} \tag{51}$$

where

$$\begin{aligned} H(\boldsymbol{p};t,w,\hat{w},\hat{\boldsymbol{\varrho}}) &= \left(w\cdot\boldsymbol{\alpha}^\top\boldsymbol{p} + c\right)\cdot\frac{\partial V}{\partial w} \\ &\quad + \left(\hat{w}\cdot\boldsymbol{\alpha}^\top\hat{\boldsymbol{\varrho}} + c\right)\cdot\frac{\partial V}{\partial \hat{w}} - \left(\sum_i \lambda_i\right) \\ &\quad \cdot V(t,w,\hat{w},\hat{\varrho}) + \frac{w^2}{2}\cdot\left(\boldsymbol{p}^\top\boldsymbol{\Sigma}\boldsymbol{p}\right)\cdot\frac{\partial^2 V}{\partial w^2} \\ &\quad + \frac{\hat{w}^2}{2}\cdot\left(\hat{\boldsymbol{\varrho}}^\top\boldsymbol{\Sigma}\hat{\boldsymbol{\varrho}}\right)\cdot\frac{\partial^2 V}{\partial \hat{w}^2} \\ &\quad + w\hat{w}\cdot\left(\boldsymbol{p}^\top\boldsymbol{\Sigma}\hat{\boldsymbol{\varrho}}\right)\cdot\frac{\partial^2 V}{\partial w\partial\hat{w}} \\ &\quad + \sum_i \lambda_i\int_0^\infty V(w + p_i w(\xi-1), \\ &\quad \hat{w} + \hat{p}_i\hat{w}(\xi-1), t, \hat{\varrho}) f_{\xi_i}(\xi)\,d\xi. \end{aligned} \tag{52}$$

Here $\boldsymbol{\alpha} = (\mu_1 - \lambda_1\kappa_1, \mu_2 - \lambda_2\kappa_2)^\top$ is the vector of (compensated) drift rates, $\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$ is the covariance matrix, and f_{ξ_i} is the density function for ξ_i.

Then, under Assumptions 4.1, 4.2 and 4.3, V is the value function and $\boldsymbol{p}^*$ is the optimal control for the CD problem (48)*.*

Proof See Appendix A.3 ■

Define several auxiliary variables

$$\begin{cases} \kappa_i^{(2)} = \mathbb{E}\left[(\xi_i - 1)^2\right], & (\sigma_i^{(2)})^2 = (\sigma_i)^2 + \lambda_i\kappa_i^{(2)},\ i\in\{1,2\}, \\ \vartheta = \sigma_1\sigma_2\rho - (\sigma_2^{(2)})^2, & \gamma = (\sigma_1^{(2)})^2 + (\sigma_2^{(2)})^2 - 2\sigma_1\sigma_2\rho, \\ \phi = \frac{(\mu_1-\mu_2)(\mu_1-\mu_2+\vartheta)}{\gamma}, & \eta = \frac{(\mu_1-\mu_2+\vartheta)^2}{\gamma} - (\sigma_2^{(2)})^2, \end{cases} \tag{53}$$

then we have the following proposition regarding the optimal control of problem (48).

Proposition 4.1 (CD-optimal control) *Suppose Assumptions* 4.1*,* 4.2 *and* 4.3 *are applicable, then the optimal control fraction of the wealth of the active portfolio to be invested in the stock index for the CD(β) problem* (48) *is given by $p^*(t,w,\hat{w},\hat{\varrho})\in\mathbb{R}$, where*

$$\begin{aligned} p^*(t,w,\hat{w},\hat{\varrho}) &= \frac{1}{W^*(t)}\left[\frac{(\mu_1-\mu_2)}{\gamma}h(t;\beta,c)\right. \\ &\quad + \frac{(\mu_1-\mu_2+\vartheta)}{\gamma}\left(g(t;\beta)\hat{W}(t) - W^*(t)\right) \\ &\quad \left. + g(t;\beta)\hat{W}(t)\cdot\hat{\varrho}\right]. \end{aligned} \tag{54}$$

Here $W^*(t)$ denotes the wealth process of the active portfolio following control $\boldsymbol{p}^*(t,W^*(t),\hat{W}(t),\hat{\varrho}) = (p^*(t,W^*(t),\hat{W}(t),\hat{\varrho}), 1 - p^*(t,W^*(t),\hat{W}(t),\hat{\varrho}))^\top$, where p^* is the optimal stock allocation described in (54)*, and $\hat{W}(t)$ is the wealth process of the benchmark portfolio following the fixed-mixed strategy described in Assumption 4.2. Here, h and g are deterministic functions of time,*

$$g(t;\beta) = -\frac{D(t;\beta)}{2A(t)}, \quad h(t;\beta,c) = -\frac{B(t;\beta,c)}{2A(t)}, \tag{55}$$

where A, D and B are deterministic functions defined as

$$\begin{aligned} A(t) &= \frac{e^{(2\mu_2-\eta)(T-t)} - 1}{(2\mu_2-\eta)}, \\ D(t;\beta) &= 2e^{\beta T}\left(\frac{e^{-\beta(T-t)} - e^{(2\mu_2-\eta)(T-t)}}{2\mu_2-\eta+\beta}\right), \end{aligned} \tag{56}$$

and

$$\begin{aligned} B(t;\beta,c) &= \frac{2c}{2\mu_2-\eta}\left(\frac{e^{(2\mu_2-\eta)(T-t)} - e^{(\mu_2-\phi)(T-t)}}{\mu_2+\phi-\eta}\right. \\ &\quad \left. - \frac{e^{(\mu_2-\phi)(T-t)} - 1}{\mu_2-\phi}\right) \\ &\quad + \frac{2ce^{\beta T}}{2\mu_2-\eta+\beta}\left(\frac{e^{(\mu_2-\phi)(T-t)} - e^{-\beta(T-t)}}{\mu_2-\phi+\beta}\right. \\ &\quad \left. - \frac{e^{(2\mu_2-\eta)(T-t)} - e^{(\mu_2-\phi)(T-t)}}{\mu_2+\phi-\eta}\right). \end{aligned} \tag{57}$$

Proof See Appendix A.4. ■

Remark 4.3 (Properties of optimal control) Equation (54) provides valuable insights into the behavior of the optimal control. Specifically, the optimal control follows a contrarian strategy, allocating more wealth to stocks when the active portfolio lags behind the benchmark. Interestingly, contrary to intuitive expectations, the optimal control aims for a level higher than the elevated target in order to meet the elevated target. For a more in-depth discussion, please refer to Appendix A.5.

4.2. Constraint-feasible clipped form solution

We first acknowledge that the closed-form solution p^* defined in (54) is obtained under several unrealistic assumptions, namely continuous rebalancing, unlimited leverage, and continuing trading in insolvency.† In practice, investors have constraints such as discrete rebalancing, limited leverage, and no trading when insolvent. For a meaningful comparison, instead of comparing the neural network strategy with the closed-form solution p^* directly, we compare the neural network strategy with an easily obtainable approximation to the closed-form solution that satisfies realistic constraints.

In particular, we consider the equally-spaced discrete rebalancing schedule $\mathcal{T}$ defined in (1). Then, the *clipped form*

† Note that we consider a two-asset scenario here, thus the scalar $p^*\in\mathbb{R}$ (allocation fraction for the stock index) fully describes the allocation strategy $\boldsymbol{p}^*$, since $\boldsymbol{p}^* = (p^*, 1-p^*)^\top$.

$\bar{p}_{\Delta t} : \mathcal{T} \times \mathbb{R}^3 \mapsto \mathbb{R}$ is defined as†

$$\begin{aligned}(\text{Clipped form}) : \quad & \bar{p}_{\Delta t}(t_i, \bar{W}_{\Delta t}(t_i), \hat{W}_{\Delta t}(t_i), \hat{\varrho}) \\ & = \min\left(\max\left(p^*(t_i, \bar{W}_{\Delta t}(t_i), \hat{W}_{\Delta t}(t_i), \hat{\varrho}), p_{\min}\right), p_{\max}\right).\end{aligned} \tag{58}$$

Here $[p_{\min}, p_{\max}]$, where $p_{\min} = 0$ and $p_{\max} \geq 1$, is the allowed range, $\bar{W}_{\Delta t}(t_i)$ is the wealth of the active portfolio at t_i following $\bar{p}_{\Delta t}$ from t_0 to t_i, $\hat{W}_{\Delta t}(t_i)$ is the wealth of the benchmark portfolio at t_i following the fixed-mix strategy described by constant allocation fraction $\hat{\varrho}$, but only rebalanced discretely according to $\mathcal{T}$. Clearly, the allocation strategy from $\bar{p}_{\Delta t}$ follows the discrete schedule of $\mathcal{T}$, and satisfies the leverage constraint that $\bar{p}_{\Delta t} \in [p_{\min}, p_{\max}]$. $p_{\Delta t}$ approaches the closed-form solution p^* as $\Delta t \downarrow 0, p_{\min} \downarrow -\infty$ and $p_{\max} \uparrow \infty$. We note that a similar clipping idea is explored in Vigna (2014) in the context of closed-form solutions for multi-period mean-variance asset allocation. However, it should be emphasized that the clipped form $\bar{p}_{\Delta t}$ with finite $(p_{\min}, p_{\max})$ is a feasible, but in general suboptimal, control of the leverage-constrained CD problem (26).

We then address the assumption that trading continues when insolvent, i.e. when the wealth of the portfolio reaches zero. While necessary for the mathematical derivation of the closed-form solution, we acknowledge that this is by no means reasonable for practitioners. Under the continuous rebalancing case (no jumps), if the control (allocation) is bounded, it is shown that the wealth of the portfolio can never be negative (Wang and Forsyth 2012). However, with discrete rebalancing, even with a bounded control, as long as the upper bound $p_{\max} > 1$, it is theoretically possible that the portfolio value becomes negative.

We address this assumption by applying an overlay on strategies so that in the case of insolvency, we assume the manager liquidates the long-only positions and allocates the debt (negative wealth) to a shortable (bond) asset (consistent with Assumption 2.2) to allow outstanding debt to accumulate until the end of the investment horizon. We remark that in practice, this overlay has little effect. In numerical experiments with 10 000 samples of observed wealth trajectories (based on calibrated jump-diffusion model or bootstrap-resampled data paths), we do not observe any single wealth trajectory that ever hits negative wealth for any strategy (e.g. neural network strategy, clipped form, etc).

In summary, the clipped form satisfies the realistic constraints and is a comparable benchmark for the neural network strategy. In the following section, we will numerically compare the performance of the clipped form, the neural network strategy, and the closed-form solution.

4.3. Comparison: LFNN strategy vs clipped-form solution

We assess and compare the performance of the neural network strategy and the clipped form, we assume the following investment scenario described in Table 2.

† Recall that we consider only two assets, a stock index and a bond index, with $\bar{p}_{\Delta t}$ being the fraction of wealth in stocks.

Table 2. Investment scenario.

Investment horizon T (years)	10
Assets	CRSP cap-weighted index (real)/30-day T-bill (U.S.) (real)
Index Samples	Concatenated 1940:8–1951:7, 1968:9–1985:10
Initial portfolio wealth/annual cash injection	100/10
Rebalancing frequency	Monthly, quarterly, semi-annually, annually
Maximum leverage	1.3
Benchmark equity percentage	0.7
Outperformance target rate β	1% (100 bps)

We assume the stock index and the bond index prices follow a double exponential jump model (49), see e.g. Kou (2002) and Kou and Wang (2004), i.e. for the jump variable ξ_i, $y_i = \log(\xi_i)$ follows the double exponential distribution with density functions $g_i(y_i)$ defined as follows

$$g_i(y_i) = \nu_i \iota_i e^{-\iota_i y_i} \mathbf{1}_{y_i \geq 0} + (1 - \nu_i) \varsigma_i e^{-\varsigma_i y_i} \mathbf{1}_{y_i < 0}, \quad i = 1, 2. \tag{59}$$

where ν_i is the probability for an upward jump, and ι_i and ς_i are parameters that describe the upward jump and downward jump respectively. The double exponential jump-diffusion model allows the flexibility of modeling asymmetric upward and downward jumps in asset prices, which seems an appropriate assumption for inflation regimes.

Using the threshold technique (Mancini 2009, Cont and Mancini 2011, Dang and Forsyth 2016), we calibrate the double exponential jump-diffusion models to the historical high-inflation periods (which we will discuss in more detail in section 5.1). The calibrated parameters can be found in Appendix 2. Then, we construct a training data set $\boldsymbol{Y}$ and a testing data set $\boldsymbol{Y}^{test}$ by sampling the calibrated model, each with 10 000 samples.

The neural network strategy follows the LFNN model learned from $\boldsymbol{Y}$. We then evaluate the performance of the neural network strategy and the approximate form (58) on the testing data set $\boldsymbol{Y}^{test}$. Specifically, we compare the value of the CD objective function for the neural network strategy and the clipped form on $\boldsymbol{Y}^{test}$. In particular, this training/testing process is repeated for various rebalancing frequencies from monthly to annually, as described in table 2.

In table 3, we can see that the neural network strategy consistently outperforms the clipped form in terms of the objective function value for all rebalancing frequencies. From Table 3 we can see that the objective function values of both the neural network strategy and the clipped form converge at roughly a first-order rate as $\Delta t \downarrow 0$. Assuming this to be true, we extrapolate the solution to $\Delta t = 0$ using Richardson extrapolation. These extrapolated values are estimates of the exact value of the continuous-time CD objective function (48) for the clipped form and the neural network strategy.

Table 3. CD objective function values. Results shown are evaluated on $\boldsymbol{Y}^{test}$, the lower the better.

Closed-form solution objective function value: 418					
Strategy	$\Delta t = 1$	$\Delta t = 1/2$	$\Delta t = 1/4$	$\Delta t = 1/12$	$\Delta t = 0$
Clipped form	545	504	479	467	461 (extrapolated)
Neural network	537	498	476	464	458 (extrapolated)

We can see that the neural network strategy still outperforms the clipped form in terms of the extrapolated objective function value. We can also see that the extrapolated neural network objective function value is lower than the (suboptimal) clipped form extrapolated value, but, of course, larger than the unconstrained closed-form solution.

Finally, we compare the neural network allocation strategy with the clipped form strategy. Specifically, in figure 1, we consider the case of monthly rebalancing and present the scatter plots of the allocation fraction in the stock index with respect to time t and the ratio between the wealth of the active portfolio $W(t)$ and the elevated target $e^{\beta t}\hat{W}(t)$. For simplicity, we call this ratio the 'tracking ratio'. We plot the 3-tuple $(\frac{W(t)}{e^{\beta t}\hat{W}(t)}, t, p_1(W(t), \hat{W}(t), t))$ (obtained from the evaluation of the strategies on samples from $\boldsymbol{Y}^{test}$) by using time t as the x-axis, the tracking ratio $\frac{W(t)}{e^{\beta t}\hat{W}(t)}$ as the y-axis, and the values of the corresponding allocation fraction to the cap-weighted index $p_1(W(t), \hat{W}(t), t)$ to color the scattered dots on the plot. A darker shade of the color indicates a higher allocation fraction.

As we can see from figure 1, the stock allocation fraction of the neural network strategy behaves similarly to the stock allocation fraction from the clipped form. Both strategies invest more wealth in the stock when the tracking ratio is lower, which is consistent with the insights we obtained in section A.5. In addition, the transition patterns of the allocation fraction of the two strategies are also highly similar. One can almost draw an imaginary horizontal dividing line around $\frac{W(t)}{e^{\beta t}\hat{W}(t)} = 0.9$ that separates high stock allocation and low stock allocation for both strategies.

We remark that a common criticism towards the use of neural networks is about the lack of interpretability compared to more interpretable counterparts such as the regression models (Rudin 2019). In this section, we see that, under the specified setting, the neural network strategy closely resembles the closed-form solution for the CD objective. The closed-form solution, in turn, complements the neural network model and offers an alternative way of interpreting results obtained from the neural network.

5. High-inflation case study

In this section, as a topic of recent interest, we present a case study that explores optimal asset allocation during high-inflation periods using the LFNN model through numerical experiments. To conduct our analysis, we need data specifically from high inflation periods. Such data can be acquired using parametric modeling or non-parametric sample generation methods. We note that our LFNN approach is agnostic to the choice of data modeling methods.

Table 4. Inflation regimes determined using a five-year moving window with a cutoff inflation rate of 0.05.

Time period	Average annualized inflation
1940:8–1951:7	.0564
1968:9–1985:10	.0661

5.1. *Filtering historical inflation regimes*

While there is no universally accepted method for identifying or modeling high inflation regimes, for the purpose of this demonstration, we employ a simple filtering technique to identify inflation regime data and generate the required samples for training the LFNN.

We use the U.S. CPI index and monthly data from the Center for Research in Security Prices (CRSP) over the 1926:1–2022:1 period.† ·‡ We select high-inflation periods as determined by the CPI index using the following filtering procedure. Using a moving window of k months, we determine the cumulative CPI index log return (annualized) in this window. If the cumulative annualized CPI index log return is greater than a cutoff, then all the months in the window are flagged as part of a high inflation regime. Note that some months may appear in more than one moving window. Any months which do not meet this criterion are considered to be in low-inflation regimes. Pseudo-code can be found in Appendix D in Ni *et al.* (2023).

Since the average annual inflation over the period 1926:1–2022:1 was 2.9%, and Federal Reserve policymakers have been targeting the inflation rate of 2% over the long run to achieve maximum employment and price stability (The Federal Reserve 2011), we use a cutoff of 5% as the threshold for high inflation. In addition, we use the moving window size of 5 years (see Appendix D.2 in Ni *et al.* 2023 for more discussion). This uncovers two inflation regimes: 1940:8–1951:7 and 1968:9–1985:10, which correspond to well-known market shocks (i.e. the second world war, and price controls; the oil price shocks and stagflation of the seventies). Table 4 shows the average annual inflation over the two regimes identified from our filter.

† The date convention is that, for example, 1926:1 refers to January 1, 1926.

‡ More specifically, results presented here were calculated based on data from Historical Indexes, © 2022 Center for Research in Security Prices (CRSP), The University of Chicago Booth School of Business. Wharton Research Data Services (WRDS) was used in preparing this article. This service and the data available thereon constitute valuable intellectual property and trade secrets of WRDS and/or its third-party suppliers. We also use the U.S. CPI index, also supplied by CRSP.

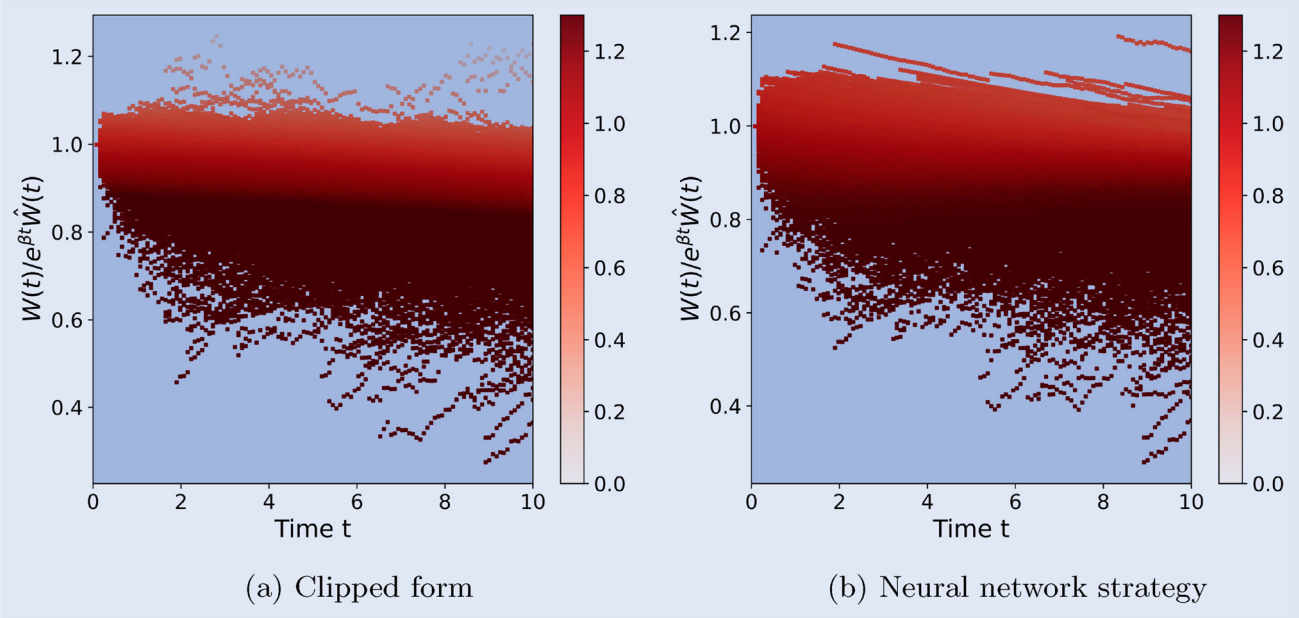


Figure 1. Stock allocation fraction w.r.t. tracking ratio $W(t)/(e^{\beta t}\hat{W}(t))$ and time t. Results are based on the evaluation on the testing data set $\boldsymbol{Y}^{test}$, and monthly rebalancing (i.e. $\Delta t = 1/12$). (a) Clipped form and (b) Neural network strategy.

For possible investment assets, we consider the 30-day U.S. T-bill index (CRSP designation 't30ind'), a constant maturity 10-year U.S. treasury index,§ and the cap-weighted stock index (CapWt) and the equal-weighted stock index (EqWt), also from CRSP.¶ All of these various indexes are adjusted for inflation by using the U.S. CPI index.

We find that the equal-weighted stock index has a higher average return and higher volatility than the cap-weighted stock index. In addition, we find that the 30-day T-bill index has a similar average return as the 10-year T-bond index, but much lower volatility† This indicates that the T-bill index is the better choice of a defensive asset during high inflation. Subsequently, we consider the equal-weighted stock index, the cap-weighted stock index, and the 30-day T-bill index.

5.2. *Bootstrap resampling*

Once we have obtained the filtered historical high-inflation data series from section 5.1, it becomes necessary to generate training and testing data sets from the original time series data. While one common approach is to assume and fit a parametric model to the underlying data, it is important to acknowledge the limitations associated with this choice.

Parametric models have several drawbacks, including the difficulty of accurately estimating their parameters (Black 1993). Even for a simple geometric Brownian motion (GBM) model, accurately estimating the drift rate can be challenging and prone to errors, requiring a long historical period of data coverage (Brigo *et al.* 2008). More complex models, such as the jump-diffusion model (49), introduce additional components to the stochastic model, which necessitates the estimation of extra parameters. Furthermore, parametric models inherently make assumptions about the true stochastic model for asset prices, which can be subject to debate.

Acknowledging the above limitations of parametric market data models, we turn to the alternative non-parametric method of bootstrap resampling as a data-generating process for numerical experiments. Unlike parametric models, non-parametric models such as bootstrap resampling do not make assumptions about the parametric form of the asset price dynamics. Intuitively speaking, the bootstrap resampling method randomly chooses data points from the historical time series data and reassembles them into new paths of time series data. The bootstrap was initially proposed as a statistical method for estimating the sampling distribution of statistics (Efron 1992). We use it as a data-generating procedure, as the philosophy behind bootstrap resampling is consistent with the idea that *'history does not repeat, but it rhymes.'* The bootstrap resampling provides an empirical distribution, which seems to be the least prejudiced estimate possible of the underlying distribution of the data-generating process. We also note that bootstrap resampling is widely adopted by practitioners (Alizadeh and Nomikos 2007, Cogneau and Zakamouline 2013, Dichtl *et al.* 2016, Scott and Cavaglia 2017, Shahzad *et al.* 2019, Cavaglia *et al.* 2022, Simonian and Martirosyan 2022) as well as academics (Anarkulova *et al.* 2022).

Specifically, we choose to use the stationary block bootstrap resampling method (Politis and Romano 1994). See

§ The 10-year treasury index was generated from monthly returns from CRSP back to 1941 (CRSP designation 'b10ind'). The data for 1926–1941 are interpolated from annual returns in Homer and Sylla (1996). The 10-year treasury index is constructed by (a) buying a 10-year treasury at the start of each month, (b) collecting interest during the month, and then (c) selling the treasury at the end of the month. We repeat the process at the start of the next month. The gains in the index then reflect both interest and capital gains and losses.
¶ The capitalization-weighted total returns have the CRSP designation 'vwretd', and the equal-weighted total returns have the CRSP designation 'ewretd'.
† see Appendix D.3 in Ni *et al.* (2023).

Appendix E.1 in Ni *et al.* (2023) for the detailed pseudo-code for bootstrap resampling. Compared to the traditional bootstrap method, the block bootstrap technique preserves the local dependency of data within blocks. Furthermore, the stationary block bootstrap uses random blocksizes which preserves the stationarity of the original time series data. An important parameter is the expected blocksize, which, informally, is a measure of serial correlation in the return data. A challenge in using block bootstrap resampling is the need to choose a single blocksize for multiple underlying time series data so that the bootstrapped data entries for different assets are synchronized in time. Subsequently, we use the expected blocksize of 6 months for all time series data. However, we have compared different numerical experiments using a range of blocksizes, including i.i.d. assumptions (i.e. expected blocksize equal to one month), and find that the results are relatively insensitive to blocksize, as discussed in more detail in Appendix E.2 in Ni *et al.* (2023).

Typically, the bootstrap technique resamples from data sourced from one contiguous segment of historical periods. However, the moving-window filtering algorithm has identified two non-contiguous historical inflation regimes. To apply the bootstrap method, there are two intuitive possibilities: (1) concatenate the returns of the two historical inflation regimes first, then bootstrap from the concatenated combined series, or (2) bootstrap within each regime (i.e. using circular block bootstrap resampling within each regime), then combine the returns of the bootstrapped resampled data points. We have experimented with both methods and find that the difference is minimal. In this article, we adopt the first method, i.e. we concatenate the returns of the historical regimes first, then bootstrap from the combined series. This method is also adopted by Anarkulova *et al.* (2022), where stock returns from different countries are concatenated and the bootstrap is applied to the combined data.

5.3. *A case study on high inflation investment: a 4-asset scenario*

5.3.1. Experiment setup. In this section, we conduct a case study on optimal asset allocation during a consistent high-inflation regime. The details of the investment specification are given in table 5. Briefly, the active portfolio and benchmark portfolio begin with the same initial wealth of 100 at $t_0 = 0$. Both portfolios are rebalanced monthly. The investment horizon is 10 years, and there is an annual cash injection of 10 for both portfolios, evenly divided over 12 months.

We consider an empirical case in which we allow the manager to allocate between four investment assets: the equal-weighted stock index, the cap-weighted stock index, the 30-day U.S. T-bill index, and the 10-year U.S. T-bond index. We assume that the stock indexes and the 10-year T-bond index are long-only assets. The manager can short the T-bill index to take leverage and invest in the long-only assets (with maximum total leverage of 1.3). In this experiment, we assume the borrowing premium rate is zero. Essentially, we assume that the manager can borrow short-term funding to take leverage at the same cost as the treasury bill. This may be a reasonable assumption for sovereign wealth funds, as they are state-owned and enjoy a high credit rating. We remark that the borrowing premium does not really affect the results significantly.† The annual outperformance target β is set to be 2% (i.e. 200 bps per year).

Table 5. Investment scenario.

Investment horizon T (years)	10
Equity market indexes	CRSP cap-weighted/ equal-weighted index (real)
Bond index	CRSP 30-day/10-year U.S. treasury index (real)
Index samples for bootstrap	Concatenated 1940:8–1951:7, 1968:9–1985:10
Initial portfolio wealth/annual cash injection	100/10
Rebalancing frequency	Monthly
Maximum leverage	1.3
Outperformance target rate β	2%
Regularization parameter ϵ	10^{-6}

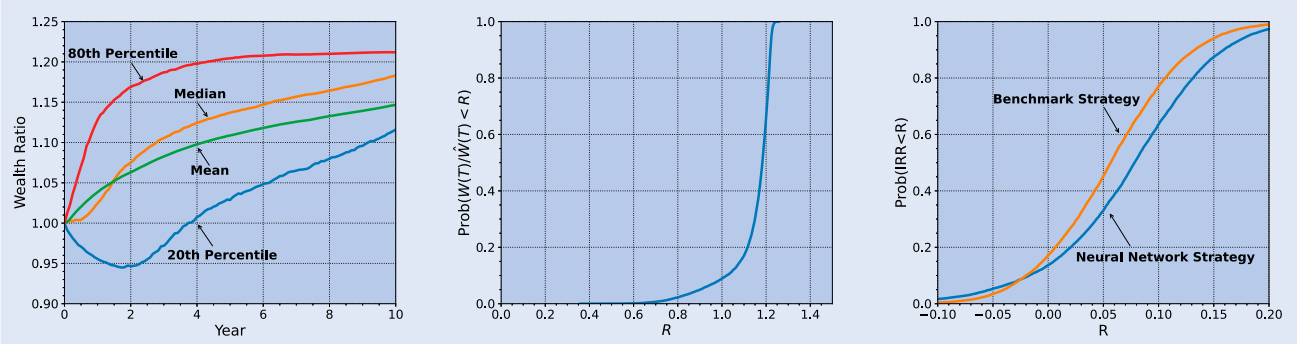
It is worth noting that we choose the benchmark portfolio to be a fixed-mix portfolio that maintains a 70% weight in the equal-weighted stock index and 30% in the 30-day U.S. T-bill index. We select this fixed-mix portfolio as the benchmark based on our observation that the equal-weighted stock index shows superior performance compared to the cap-weighted stock index during high-inflation environments. Surprisingly, when analyzing bootstrap-resampled data from the historical inflation regimes, we find that the fixed-mix portfolio consisting of 70% in the equal-weighted stock index and 30% in the 30-day U.S. T-bill index partially stochastically dominates the fixed-mix portfolio consisting of 70% in the cap-weighted stock index and 30% in the 30-day U.S. T-bill index. For more detailed information, interested readers can refer to Appendix F in Ni *et al.* (2023).

As discussed in the previous section, we use the stationary bootstrap resampling algorithm to generate a training data set $\boldsymbol{Y}$ and a testing data set $\boldsymbol{Y}^{test}$ (both with 10 000 resampled paths) from the concatenated index samples from two historical inflation regimes: 1940:8—1951:7 and 1968:9–1985:10, using an expected blocksize of 6 months. The testing data set $\boldsymbol{Y}^{test}$ is generated using a different random seed as the training data set $\boldsymbol{Y}$, and thus the probability of seeing the same sample in $\boldsymbol{Y}$ and $\boldsymbol{Y}^{test}$ is near zero (see Ni *et al.* 2022 for proof).

We remark that in this experiment, we train the LFNN model (21) on $\boldsymbol{Y}$ under the following discrete-time cumulative quadratic shortfall (CS) objective, instead of the CD objective (48) used in deriving the closed-form solution.

$$\text{(Parameterized } CS(\beta)\text{)}: \quad \inf_{\boldsymbol{\theta}\in\mathbb{R}^{N_{\boldsymbol{\theta}}}} \mathbb{E}^{(t_0,w_0)}_{f(\cdot;\boldsymbol{\theta})} \left[\sum_{t\in\mathcal{T}} \left(\min\left(W_{\boldsymbol{\theta}}(t) - e^{\beta t}\hat{W}(t), 0\right)\right)^2 + \epsilon W_{\boldsymbol{\theta}}(T)\right], \quad (60)$$

† See Appendix I in Ni *et al.* (2023) for a more detailed discussion.



(a) Percentiles of wealth ratio $\frac{W(t)}{\hat{W}(t)}$ (b) Terminal wealth ratio (c) CDF of IRR

Figure 2. Percentiles of wealth ratio $\frac{W(t)}{\hat{W}(t)}$ and CDF of terminal wealth ratio $\frac{W(T)}{\hat{W}(T)}$ and internal rate of return (IRR). Results are based on the evaluation of the learned neural network model on $\boldsymbol{Y}^{test}$. (a) Percentiles of wealth ratio $\frac{W(t)}{\hat{W}(t)}$. (b) Terminal wealth ratio and (c) CDF of IRR.

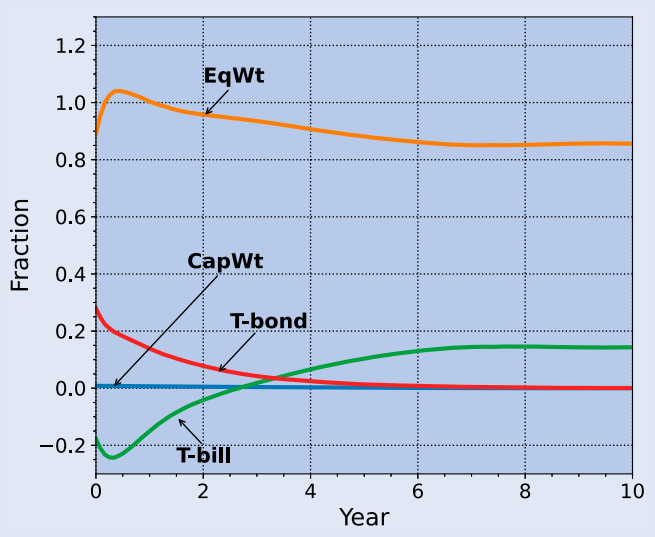


Figure 3. Mean allocation fraction over time, evaluated on $\boldsymbol{Y}^{test}$.

In equation (60), the term $\epsilon W_{\boldsymbol{\theta}}(T)$ regularizes the solution to avoid ill-posedness. This can occur along a path where $W_{\boldsymbol{\theta}}(t) \gg e^{\beta t}\hat{W}(t)$ as $t \to T$. See Chen *et al.* (2023) for more discussion.

The CS objective function only penalizes underperformance relative to the elevated target, whereas the CD objective penalizes the outperformance equally as the underperformance. Numerical comparisons of the two objective functions suggest that the CS objective function indeed yields more favorable investment results than the CD objective (see Appendix H in Ni *et al.* 2023).

In this section, unless stated otherwise, all the results presented are testing results.

5.4. *Experiment results*

The analysis of figure 2(a) reveals that the neural network strategy (the strategy following the training LFNN model) consistently outperforms the benchmark strategy in terms of the wealth ratio $W(t)/\hat{W}(t)$. Over time, both the mean and median wealth ratios demonstrate a smooth and consistent increase. Regarding tail performance (20th percentile), the neural network strategy initially falls behind the benchmark but gradually recovers and ultimately achieves 10% greater wealth at the terminal time. This observation indicates that the neural network strategy effectively manages tail risk.

An additional metric that holds significant interest for managers is the distribution of the terminal wealth ratio $\frac{W(T)}{\hat{W}(T)}$. This metric examines the relative performance of the strategies at the end of the investment period. Figure 2(b) illustrates that there is a greater than 90% chance that the neural network strategy outperforms the benchmark strategy in terms of terminal wealth. This outcome is particularly noteworthy as the objective function (60) does not directly target the terminal wealth ratio.

Given the constant cash injections in the portfolios, it is appropriate to employ the internal rate of return (IRR) as a measure of the portfolio's annualized performance. Figure 2(c) demonstrates that the neural network strategy has a more than 90% chance of producing a higher IRR. Furthermore, the median IRR of the neural network strategy exceeds that of the benchmark strategy by slightly over 2%, aligning with the chosen target outperformance rate of $\beta = 0.02$. This indicates that the neural network model consistently achieves the desired target performance across most outcomes.

The results from table 6 indicate that the 5th percentile of the terminal wealth for the neural network strategy is lower than that of the benchmark strategy. This suggests that in some scenarios, particularly during persistent bear markets when stocks perform poorly, the neural network strategy may experience lower terminal wealth compared to the benchmark strategy. The neural network strategy takes on more risk by allocating a higher fraction of wealth to the equal-weighted stock index, which is considered a riskier asset, in comparison to the benchmark portfolio.

It's important to note, however, that these scenarios occur with low probability. As depicted in figure 2(b), the neural network strategy exhibits a significantly high probability of outperforming the benchmark in terms of terminal wealth, exceeding 90%. This implies that while there might be instances where the neural network strategy suffers relative to the benchmark, the overall performance is consistently strong, resulting in a high likelihood of achieving superior

Table 6. Statistics of strategies. Results are based on the evaluation results on the testing data set.

Strategy	Median[W_T]	E[W_T]	std[W_T]	5th Percentile	Median IRR (annual)
Neural network	364.2	403.4	211.8	136.3	0.078
Benchmark	308.5	342.9	165.0	149.0	0.056

terminal value. Of course, it is not possible to outperform the benchmark with certainty, since this would imply an arbitrage opportunity.

To gain insight into the strong performance of the neural network strategy, we further examine its allocation profile.

We begin by examining the mean allocation fraction for the four assets over time, as depicted in figure 3. The first noteworthy observation from figure 3 is that, on average, the neural network strategy does not allocate wealth to the cap-weighted stock index. Initially, this might appear surprising; however, it aligns with historical data indicating significantly higher real returns for the equal-weighted stock index during periods of high inflation. Given that the objective is to outperform a benchmark heavily invested in the equal-weighted index (70%), it is logical to avoid allocating wealth to a comparatively weaker index in the active portfolio.

The second observation derived from figure 3 pertains to the evolution of mean bond allocation fractions. Initially, the neural network strategy shorts the 30-day T-bill index and assumes some leverage while heavily investing in the equal-weighted stock index during the first two years. This indicates a deliberate risk-taking approach early on to establish an advantage over the benchmark strategy. Subsequently, the allocation to the 10-year T-bond decreases, coinciding with the reduction in the allocation to the equal-weighted index. This suggests that the initial allocation to the T-bond was primarily for leveraging purposes, with the 10-year bond being the only defensive asset available. As leverage is no longer used in later years, the neural network strategy favors the T-bill over the 10-year bond.

Overall, despite the gradual decrease in stock allocation over time, the neural network strategy maintains an average allocation of more than 80% to the equal-weighted stock index. This is expected, as outperforming an aggressive benchmark with a 70% allocation to the equal-weighted stock index necessitates assuming higher levels of risk. Despite the higher allocation to riskier assets, the neural network strategy consistently delivers strong results compared to the benchmark strategy, as illustrated in figure 2.

Lastly, it is worth noting that the neural network strategy, trained under high-inflation regimes, exhibits remarkable performance on low-inflation testing datasets. This unexpected outcome highlights the robustness of the strategy. For further discussion on this topic, interested readers can refer to Appendix A.6.

6. Conclusion

In this paper, our primary objective is to propose a methodology that generates optimal dynamic allocation strategies under leverage constraints in order to outperform a benchmark during high inflation regimes. Imposing a leverage constraint in multi-period asset allocation is consistent with the practice in large sovereign wealth funds, which often have exposure to alternative assets.† Our proposed framework efficiently solves the constrained stochastic optimal control problems without dynamic programming, accommodating diverse objective functions, constraints, and data sources.

To overcome the limitations of unrealistic assumptions and derive a more practical solution, we introduce a novel leverage-feasible neural network (LFNN) model. The LFNN model approximates the optimal control directly, eliminating the need for high-dimensional approximations of conditional expectations commonly required in dynamic programming approaches. Additionally, the LFNN model converts the leverage-constrained optimization problem into an unconstrained optimization problem. Importantly, we justify the validity of the LFNN approach by mathematically proving that the solution to the parameterized unconstrained optimization problem can approximate the solution to the original constrained optimization problem with arbitrary precision.

To provide empirical evidence of the approximation accuracy of the proposed neural network model, we derive a closed-form control under the cumulative tracking difference (CD) objective function, and jump-diffusion asset price models. We find that the LFNN model achieves comparable performance to the clipped-form control on synthetic data.

Finally, we conduct a case study on multi-period portfolio optimization during high-inflation regimes. We apply the LFNN model to bootstrap-resampled data from filtered historical high-inflation data. In our numerical experiment, we consider an investment case with four assets in high inflation regimes. The results demonstrate that the neural network strategy consistently outperforms the benchmark strategy throughout the investment period. Specifically, the neural network strategy achieves a 2% higher median Internal Rate of Return (IRR) compared to the benchmark strategy and yields a higher terminal wealth with more than a 90% probability. The allocation strategy derived from the LFNN model suggests that managers should favor the equal-weighted stock index over the cap-weighted stock index and short-term bonds over long-term bonds during high-inflation periods.

Disclosure statement

No potential conflict of interest was reported by the author(s).

† Alternative assets often employ explicit leverage strategies.

Funding

Li's work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) [grant RGPIN-2020-04331]. Forsyth's work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) [grant RGPIN-2017-03760].

References

Al-Aradi, A., Correia, A., Jardim, G., de Freitas Naiff, D. and Saporito, Y., Extensions of the deep Galerkin method. *Appl. Math. Comput.*, 2022, **430**, 127287.

Al-Aradi, A. and Jaimungal, S., Outperformance and tracking: Dynamic asset allocation for active and passive portfolio management. *Appl. Math. Finance*, 2018, **25**(3), 268–294.

Alekseev, A.G. and Sokolov, M.V., Benchmark-based evaluation of portfolio performance: A characterization. *Ann. Finance*, 2016, **12**, 409–440.

Alizadeh, A.H. and Nomikos, N.K., Investment timing and trading strategies in the sale and purchase market for ships. *Transp. Res. Part B Methodol.*, 2007, **41**(1), 126–143.

Anarkulova, A., Cederburg, S. and O'Doherty, M.S., Stocks for the long run? evidence from a broad sample of developed markets. *J. Financ. Econ.*, 2022, **143**(1), 409–433.

Ang, A., Papanikolaou, D. and Westerfield, M.M., Portfolio choice with illiquid assets. *Manage. Sci.*, 2014, **60**(11), 2737–2761.

Ball, L.M., Leigh, D. and Mishra, P., Understanding US inflation during the COVID era. Technical report, 2022 (National Bureau of Economic Research).

Basak, S., Shapiro, A. and Tepla, L., Risk management with benchmarking. *Manage. Sci.*, 2006, **52**(4), 542–557.

Black, F., Estimating expected return. *Financ. Ana. J.*, 1993, **49**(5), 36–38.

Blanchet-Scalliet, C., El Karoui, N., Jeanblanc, M. and Martellini, L., Optimal investment decisions when time-horizon is uncertain. *J. Math. Econ.*, 2008, **44**(11), 1100–1113.

Bo, L., Liao, H. and Yu, X., Optimal tracking portfolio with a ratcheting capital benchmark. *SIAM J. Control Optim.*, 2021, **59**(3), 2346–2380.

Brigo, D., Dalessandro, A., Neugebauer, M. and Triki, F., A stochastic processes toolkit for risk management, 2008. arXiv preprint arXiv:0812.4210.

Browne, S., Survival and growth with a liability: Optimal portfolio strategies in continuous time. *Math. Oper. Res.*, 1997, **22**(2), 468–493.

Browne, S., Beating a moving target: Optimal portfolio strategies for outperforming a stochastic benchmark. *Finance Stoch.*, 1999, **3**, 275–294.

Browne, S., Risk-constrained dynamic active portfolio management. *Manage. Sci.*, 2000, **46**(9), 1188–1199.

Bureau of Labor Statistics, Consumer price index summary, 2023. Available online at: https://www.bls.gov/news.release/archives/cpi_04122023.pdf.

Cavaglia, S., Scott, L., Blay, K. and Hixon, S., Multi-asset class factor premia: A strategic asset allocation perspective. *J. Portf. Manage.*, 2022, **48**(4), 14–32.

Chen, M., Shirazi, M., Forsyth, P.A. and Li, Y., Machine learning and Hamilton–Jacobi–Bellman equation for optimal decumulation: A comparison study, 2023. arXiv:2306.10582.

Cogneau, P. and Zakamouline, V., Block bootstrap methods and the choice of stocks for the long run. *Quant. Finance*, 2013, **13**(9), 1443–1457.

Cong, F. and Oosterlee, C.W., Multi-period mean–variance portfolio optimization based on Monte-Carlo simulation. *J. Econ. Dyn. Control*, 2016, **64**, 23–38.

Cont, R. and Mancini, C., Nonparametric tests for pathwise properties of semimartingales. *Bernoulli*, 2011, **17**(2), 781–813.

CPP Investments, Annual Report 2021, 2021. Available online at: https://www.cppinvestments.com/the-fund/our-performance/financial-results/ (accessed 5 December 2022).

CPP Investments, Annual Report 2022, 2022. Available online at: https://www.cppinvestments.com/wp-content/uploads/2022/06/CPP-Investments_F2022-Annual-Report-EN.pdf (accessed 5 December 2022).

Cumbo, J., Yu, S. and Gara, A., US pension funds worth 1.5tn add risk through leverage, 2024. Available online at: https://www.ft.com/content/623b67f9-090c-457f-a327-dc9f767e327a (online, Financial Times).

Dang, D.-M. and Forsyth, P.A., Continuous time mean-variance optimal portfolio allocation under jump diffusion: A numerical impulse control approach. *Numer. Methods Partial Differ. Equ.*, 2014, **30**, 664–698.

Dang, D.-M. and Forsyth, P.A., Better than pre-commitment mean-variance portfolio allocation strategies: A semi-self-financing Hamilton–Jacobi–Bellman equation approach. *Eur. J. Oper. Res.*, 2016, **250**(3), 827–841.

Davis, M. and Lleo, S., Risk-sensitive benchmarked asset management. *Quant. Finance*, 2008, **8**(4), 415–426.

Dichtl, H., Drobetz, W. and Kryzanowski, L., Timing the stock market: Does it really make no sense? *J. Behav. Exp. Finance*, 2016, **10**, 88–104.

Dixon, M.F., Halperin, I. and Bilokon, P., *Machine Learning in Finance*, Volume 1406. 2020 (Springer).

Efron, B., Bootstrap methods: Another look at the jackknife. In *Breakthroughs in Statistics*, edited by S. Kotz, and N. L. Johnson, pp. 569–593, 1992 (Springer: New York).

Forsyth, P.A., Optimal dynamic asset allocation for DC plan accumulation/decumulation: Ambition-CVAR. *Insur. Math. Econ.*, 2020, **93**, 230–245.

Gao, Z., Gao, Y., Hu, Y., Jiang, Z. and Su, J., Application of deep Q-network in portfolio management. In *2020 5th IEEE International Conference on Big Data Analytics (ICBDA)*, pp. 268–275, 2020 (IEEE).

Goodfellow, I., Bengio, Y. and Courville, A., *Deep Learning*, 2016 (MIT Press).

Han, J., Deep learning approximation for stochastic control problems, 2016. arXiv preprint arXiv:1611.07422.

Homer, S. and Sylla, R.E., *A History of Interest Rates*, 1996 (Rutgers University Press).

Hornik, K., Approximation capabilities of multilayer feedforward networks. *Neural Netw.*, 1991, **4**(2), 251–257.

Javorcik, B., Global supply chains will not be the same in the post-COVID-19 world. In *COVID-19 and Trade Policy: Why Turning Inward Won't Work*, edited by R. E. Baldwin and S. J. Everett, Volume 111, 2020 (CEPR Press: London).

Kingma, D.P. and Ba, J., Adam: A method for stochastic optimization, 2014. arXiv preprint arXiv:1412.6980.

Kou, S.G., A jump-diffusion model for option pricing. *Manage. Sci.*, 2002, **48**(8), 1086–1101.

Kou, S.G. and Wang, H., Option pricing under a double exponential jump diffusion model. *Manage. Sci.*, 2004, **50**(9), 1178–1192.

Kratsios, A. and Bilokopytov, I., Non-Euclidean universal approximation. *Adv. Neural Inf. Process. Syst.*, 2020, **33**, 10635–10646.

Lahaye, J., Laurent, S. and Neely, C.J., Jumps, cojumps and macro announcements. *J. Appl. Econom.*, 2011, **26**(6), 893–921.

Lee, W., Yu, H., Rival, X. and Yang, H., On correctness of automatic differentiation for non-differentiable functions. *Adv. Neural Inf. Process. Syst.*, 2020, **33**, 6719–6730.

L'Her, J.F., Stoyanova, R., Shaw, K., Scott, W. and Lai, C., A bottom-up approach to the risk-adjusted performance of the buyout fund market. *Financ. Anal. J.*, 2016, **72**(4), 36–48.

Li, D. and Ng, W.-L., Optimal dynamic portfolio selection: Multiperiod mean-variance formulation. *Math. Finance*, 2000, **10**(3), 387–406.

Li, Y. and Forsyth, P.A., A data-driven neural network approach to optimal asset allocation for target based defined contribution pension plans. *Insur. Math. Econ.*, 2019, **86**, 189–204.

Li, Z., Tsang, K.H. and Wong, H.Y., Lasso-based simulation for high-dimensional multi-period portfolio optimization. *IMA J. Manage. Math.*, 2020, **31**(3), 257–280.
Lim, A. and Wong, B., A benchmark approach to optimal asset allocation for insurers and pension funds. *Insur. Math. Econ.*, 2010a, **46**(2), 317–327.
Lim, A.E. and Wong, B., A benchmarking approach to optimal asset allocation for insurers and pension funds. *Insur. Math. Econ.*, 2010b, **46**(2), 317–327.
Lin, Y., MacMinn, R.D. and Tian, R., De-risking defined benefit plans. *Insur. Math. Econ.*, 2015, **63**, 52–65.
Lu, Y. and Lu, J., A universal approximation theorem of deep neural networks for expressing probability distributions. *Adv. Neural Inf. Process. Syst.*, 2020, **33**, 3094–3105.
Lucarelli, G. and Borrotti, M., A deep Q-learning portfolio management framework for the cryptocurrency market. *Neural Comput. Appl.*, 2020, **32**(23), 17229–17244.
MacMinn, R., Brockett, P., Wang, J., Lin, Y. and Tian, R., The securitization of longevity risk and its implications for retirement security. In *Recreating Sustainable Retirement*, edited by O. Mitchell, R. Maurer, and P. Hammond, pp. 134–160, 2014 (Oxford University Press: Oxford).
Mancini, C., Non-parametric threshold estimation for models with stochastic diffusion coefficient and jumps. *Scand. J. Stat.*, 2009, **36**(2), 270–296.
Merton, R.C., Lifetime portfolio selection under uncertainty: The continuous-time case. *Rev. Econ. Stat.*, 1969, **51**(3), 247–257.
Merton, R.C., Optimum consumption and portfolio rules in a continuous-time model. *J. Econ. Theory*, 1971, **3**(4), 373–413.
Merton, R.C., Option pricing when underlying stock returns are discontinuous. *J. Financ. Econ.*, 1976, **3**(1–2), 125–144.
NASDAQ, NASDAQ Composite Index, 2023. Available online at: https://www.nasdaq.com/market-activity/index/comp/historical (accessed 15 February 2023).
Ni, C., Li, Y., Forsyth, P. and Carroll, R., Optimal asset allocation for outperforming a stochastic benchmark target. *Quant. Finance*, 2022, **22**(9), 1595–1626.
Ni, C., Li, Y. and Forsyth, P.A., Neural network approach to portfolio optimization with leverage constraints: A case study on high inflation investment, 2023. arXiv preprint arXiv:2304.05297v2.
Nicolosi, M., Angelini, F. and Herzel, S., Portfolio management with benchmark related incentives under mean reverting processes. *Ann. Oper. Res.*, 2018, **266**, 373–394.
Norges Bank, Investment Strategy, 2022. https://www.nbim.no/en/the-fund/how-we-invest/investment-strategy/ (accessed 5 December 2022).
Oderda, G., Stochastic portfolio theory optimization and the origin of rule based investing. *Quant. Finance*, 2015, **15**(8), 1259–1266.
Øksendal, B.K. and Sulem, A., *Applied Stochastic Control of Jump Diffusions*, Volume 498. 2007 (Springer).
Park, H., Sim, M.K. and Choi, D.G., An intelligent financial portfolio trading strategy using deep Q-learning. *Expert Syst. Appl.*, 2020, **158**, 113573.
Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. and Lerer, A., Automatic differentiation in Pytorch. In *NIPS 2017 Workshop on Autodiff*, Long Beach, CA, 2017. Available online at: https://openreview.net/forum?id=BJJsrmfCZ.
Phalippou, L., Performance of buyout funds revisited? *Rev. Finance*, 2014, **18**(1), 189–218.
Politis, D.N. and Romano, J.P., The stationary bootstrap. *J. Am. Stat. Assoc.*, 1994, **89**(428), 1303–1313.
Rudin, C., Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.*, 2019, **1**(5), 206–215.
Scott, L. and Cavaglia, S., A wealth management perspective on factor premia and the value of downside protection. *J. Portf. Manage.*, 2017, **43**(3), 33–41.
Shahzad, S.J.H., Bouri, E., Roubaud, D., Kristoufek, L. and Lucey, B., Is bitcoin a better safe-haven investment than gold and commodities? *Int. Rev. Financ. Anal.*, 2019, **63**, 322–330.
Simonian, J. and Martirosyan, A., Sharpe parity redux. *J. Portf. Manag.*, 2022, **48**(4), 183–193.
Sirignano, J. and Spiliopoulos, K., DGM: A deep learning algorithm for solving partial differential equations. *J. Comput. Phys.*, 2018, **375**, 1339–1364.
Tepla, L., Optimal investment with minimum performance constraints. *J. Econ. Dyn. Control*, 2001, **25**, 1629–1645.
The Federal Reserve, What is an acceptable level of inflation? 2011. https://www.federalreserve.gov/faqs/5D58E72F066A4DBDA80BBA659C55F774.htm (accessed 5 December 2022).
Tsang, K.H. and Wong, H.Y., Deep-learning solution to portfolio selection with serially dependent returns. *SIAM J. Financ. Math.*, 2020, **11**(2), 593–619.
van Staden, P.M., Forsyth, P.A. and Li, Y., Beating a benchmark: Dynamic programming may not be the right numerical approach. *SIAM J. Financ. Math.*, 2023, **14**(2), 407–451.
van Staden, P.M., Forsyth, P.A. and Li, Y., Across-time risk-aware strategies for outperforming a benchmark. *Eur. J. Oper. Res.*, 2024, **313**(2), 776–800.
Vigna, E., On efficiency of mean–variance based portfolio selection in defined contribution pension schemes. *Quant. Finance*, 2014, **14**(2), 237–258.
Wang, J. and Forsyth, P.A., Numerical solution of the Hamilton–Jacobi–Bellman formulation for continuous time mean variance asset allocation. *J. Econ. Dyn. Control*, 2010, **34**(2), 207–230.
Wang, J. and Forsyth, P.A., Comparison of mean variance like strategies for optimal asset allocation problems. *Int. J. Theor. Appl. Finance*, 2012, **15**(02), 1250014.
Wang, R., Foster, D.P. and Kakade, S.M., What are the statistical limits of offline RL with linear function approximation? 2020, **51**(3).
Yao, D.D., Zhang, S. and Zhou, X.Y., Tracking a financial benchmark using a few assets. *Oper. Res.*, 2006, **54**(2), 232–246.
Zhang, Q. and Gao, Y., Portfolio selection based on a benchmark process with dynamic value-at-risk constraints. *J. Comput. Appl. Math.*, 2017, **313**, 440–447.
Zhao, Y., A dynamic model of active portfolio management with benchmark orientation. *J. Bank. Finance*, 2007, **31**(11), 3336–3356.
Zhou, X.Y. and Li, D., Continuous-time mean-variance portfolio selection: A stochastic LQ framework. *Appl. Math. Optim.*, 2000, **42**, 19–33.

Appendices

Appendix 1. Technical details of LFNN model

A.1. Proof of Theorem 3.1

Theorem 3.1. (Unconstrained feasibility domain) The feasibility domain $\mathcal{Z}_{\boldsymbol{\theta}}$ defined in (19) associated with the LFNN model (17) is $\mathbb{R}^{N_{\boldsymbol{\theta}}}$.

Proof First, it is obvious that $\mathcal{Z}_{\boldsymbol{\theta}} \subseteq \mathbb{R}^{N_{\boldsymbol{\theta}}}$ by definition of (19). Next, we show that $\mathbb{R}^{N_{\boldsymbol{\theta}}} \subseteq \mathcal{Z}_{\boldsymbol{\theta}}$. To prove this, we need to show that for any $\boldsymbol{\theta} \in \mathbb{R}^{N_{\boldsymbol{\theta}}}$,

$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \boldsymbol{y} \in \begin{cases} \mathcal{Z}_1, & \text{if } \boldsymbol{x} \in \mathcal{X}_1, \\ \mathcal{Z}_2, & \text{if } \boldsymbol{x} \in \mathcal{X}_2, \end{cases} \quad \forall \boldsymbol{x} \in \mathcal{X}. \tag{A1}$$

Here, $f_{\boldsymbol{\theta}}$ is the LFNN function defined in (21), $\boldsymbol{y} = (y_1, \ldots, y_{N_a})^\top \in \mathbb{R}^{N_a}$ is the output of the LFNN model that represents the wealth allocation to the assets, $\mathcal{Z}_1$ and $\mathcal{Z}_2$ are the feasibility sets defined in (12) and (13), and $\boldsymbol{x} \in \mathcal{X}$ is a feature vector. To prove (A1), we verify the two scenarios for $\boldsymbol{x} \in \mathcal{X}_1$ and $\boldsymbol{x} \in \mathcal{X}_2$ respectively.

When $\boldsymbol{x} \in \mathcal{X}_2$, it is easily verifiable that $\boldsymbol{y} = \boldsymbol{e}_{N_L+1}$ via the definition of the leverage-feasible activation function (22).

Next, we verify that when $\boldsymbol{x} \in \mathcal{X}_1, \boldsymbol{y} \in \mathcal{Z}_1$. To prove this, we need to show that constraints of (6)–(9) are satisfied when $\boldsymbol{x} \in \mathcal{X}_1$.

We borrow the variables l and o_{N_a+1} in equation (22).

By definition of (22), it is obvious that the long-only constraint (6) holds for long-only assets.

It is also easy to verify that the summation constraint (7) is satisfied. This can be observed after the fact that

$$\sum_{i=1}^{N_L} y_i = l, \quad \text{and} \quad \sum_{i=N_L+1}^{N_a} y_i = 1 - l. \tag{A2}$$

The maximum leverage constraint (8) is also satisfied, as

$$\sum_{i=1}^{N_L} y_i = l = p_{\max} \cdot \sigma(-o_{N_a+1}) \leq p_{\max}. \tag{A3}$$

Finally, the simultaneous shorting constraint (2.1) is satisfied. To see this, we examine the scenario when leverage occurs, i.e. $\sum_{i=1}^{N_L} y_i = l > 1$. Then, by definition from (22), we know

$$y_i = (1-l) \cdot \frac{e^{o_i}}{\sum_{k=N+1}^{N_a} e^{o_k}} \leq 0, \quad \forall i \in \{N_L+1, \ldots, N_\} \tag{A4}$$

From (A4) it is clear that if $l \leq 1$, then $y_i \geq 0, \forall i$.

Therefore, for any $\boldsymbol{\theta} \in \mathbb{R}^{N_{\boldsymbol{\theta}}}$, (A1) is satisfied. This implies $\mathbb{R}^{N_{\boldsymbol{\theta}}} \subseteq \mathcal{Z}_{\boldsymbol{\theta}}$. ■

A.2. *Proof of Lemma 3.2*

Lemma 3.2. (Structure of feasible control). Any feasible control function $p : \mathcal{X} \mapsto \mathcal{Z}$, where $\mathcal{Z}$ is defined in (15), has the following function decomposition

$$p(\boldsymbol{x}) = \varphi(\omega(\boldsymbol{x}), \boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{X}, \tag{A5}$$

where $\varphi : \tilde{\mathcal{Z}} \times \mathcal{X} \mapsto \mathcal{Z}$ is defined in (24) and $\omega : \mathcal{X} \mapsto \tilde{\mathcal{Z}}$.

Proof We prove the Lemma by existence.

Define ω as

$$\omega(\boldsymbol{x}) = \begin{cases} \phi\left(p(\boldsymbol{x})\right), & \text{if } \boldsymbol{x} \in \mathcal{X}_1, \\ \left(\frac{1}{N_L}, \ldots, \frac{1}{N_L}, \frac{1}{N_a - N_L}, \ldots, \frac{1}{N_a - N_L}, 0\right)^\top, & \text{if } \boldsymbol{x} \in \mathcal{X}_2, \end{cases} \tag{A6}$$

where for any $\boldsymbol{z} = (z_1, \ldots, z_{N_a})^\top \in \mathcal{Z}_1$ (defined in 12), $\boldsymbol{y} = (y_1, \ldots, y_{N_a+1})^\top = \phi(\boldsymbol{z}) \in \mathbb{R}^{N_a+1}$ is defined as

$$\phi(\boldsymbol{z}) \equiv \boldsymbol{y} = \begin{cases} \begin{cases} y_i = \dfrac{z_i}{\sum_{j=1}^{N_L} z_j}, \quad i \in \{1, \ldots, N_L\}, \\ y_i = \dfrac{z_i}{1 - \sum_{j=1}^{N_L} z_j}, \quad i \in \{N_L, \ldots, N_a\}, \\ y_{N_a+1} = \sum_{j=1}^{N_L} z_j, \end{cases} & \text{if } \sum_{i=1}^{N_L} z_i \in (0,1) \cup (1, p_{\max}], \\ \begin{cases} y_i = z_i, \quad i \in \{1, \ldots, N_L\}, \\ y_i = 1/(N_a - N_L), \quad i \in \{N_L, \ldots, N_a\}, \\ y_{N_a+1} = 1, \end{cases} & \text{if } \sum_{i=1}^{N_L} z_i = 1, \\ \begin{cases} y_i = 0, \quad i \in \{1, \ldots, N_L\}, \\ y_i = z_i, \quad i \in \{N_L, \ldots, N_a\}, \\ y_{N_a+1} = 0, \end{cases} & \text{if } \sum_{i=1}^{N_L} z_i = 0, \end{cases} \tag{A7}$$

It can then be easily verified that $\omega : \mathcal{X} \mapsto \tilde{\mathcal{Z}}$, and that $p(\boldsymbol{x}) = \varphi(\omega(\boldsymbol{x}), \boldsymbol{x})$. ■

Appendix 2. Calibrated synthetic model parameters

Table A1. Estimated annualized parameters for double exponential jump-diffusion model (59) from CRSP cap-weighted stock index, 30-day U.S. T-bill index deflated by the CPI. Sample period: concatenated 1940:8–1951:7 and 1968:9–1985:10.

μ_1	σ_1	λ_1	ν_1	ι_1	ς_1	μ_2	σ_2	λ_2	ν_2	ι_2	ς_2	ρ
0.051	0.146	0.178	0.2	7.13	7.33	-0.014	0.017	0.321	0	N/A	44.48	0.14

Appendix 3. Technical details of closed-form solution

A.3. *Proof of Theorem (51)*

At any state $(t, w, \hat{w}) \in [t_0, T] \times \mathbb{R}^2$, define the value function $V(w, \hat{w}, t)$ to the CD problem (48) as

$$V(t, w, \hat{w}, \hat{\boldsymbol{\varrho}}))$$
$$= \inf_{\boldsymbol{p}} \left\{ \mathbb{E}_p \left[\int_t^T \left(W(s) - e^{\beta s} \hat{W}(s) \right)^2 \mathrm{d}s \,\middle|\, W(t) = w, \hat{W}(t) = \hat{w} \right] \right\}. \tag{A8}$$

By the dynamic programming principle, we have

$$V(t, w, \hat{w}, \hat{\boldsymbol{\varrho}})$$
$$= \inf_{\boldsymbol{p}} \left\{ \mathbb{E}_p \left[\left(V(t + \Delta t, W(t + \Delta t), \hat{W}(t + \Delta t), \hat{\boldsymbol{\varrho}}) \right.\right.\right.$$
$$\left.\left.\left. + \int_t^{t+\Delta t} \left(W(s) - e^{\beta s} \hat{W}(s) \right)^2 \mathrm{d}s \right) \,\middle|\, W(t) = w, \hat{W}(t) = \hat{w} \right] \right\} \tag{A9}$$

Rearrange equation (A9) to obtain

$$\inf_{\boldsymbol{p}} \left\{ \mathbb{E}_p \left[\left(dV(t, w, \hat{w}, \hat{\boldsymbol{\varrho}}) \right.\right.\right.$$
$$\left.\left.\left. + \int_t^{t+\Delta t} \left(W(s) - e^{\beta s} \hat{W}(s) \right)^2 \mathrm{d}s \right) \mid W(t) = w, \hat{W}(t) = \hat{w} \right] \right\} = 0 \tag{A10}$$

Then, apply It's Lemma with jumps (Cont and Mancini 2011), substitute $\mathrm{d}W$ and $\mathrm{d}\hat{W}$ terms with (50), and take limits as $\Delta t \downarrow 0$, we obtain (51).

The above results merely serve as an intuitive guide to obtain (51). The formal proof of (51) proceeds by using a suitably smooth test function, see for example (Øksendal and Sulem 2007).

A.4. *Proof of results for CD-optimal control*

In Section 4.1, we emphasized the dependence of B and D (defined in (57) and (56)) on parameters β and c for understanding the optimal control function. As β and c are fixed parameters, in this proof, we omit the dependence of B and D on them for notational simplicity.

The quadratic source term $(w - e^{\beta t}\hat{w})^2$ in Theorem (51) suggests the following *ansatz* for the value function V in Theorem 47 of the form

$$V(t, w, \hat{w}) = A(t)w^2 + B(t)w + C(t) + \hat{A}(t)\hat{w}^2 + \hat{B}(t)\hat{w} + D(t)w\hat{w}, \tag{A11}$$

where $A, B, C, \hat{A}, \hat{B}, D$ are unknown deterministic functions of time t. If (A11) is correct, then the pointwise infimum in (51) is attained by p^* satisfying the relationship

$$\left(w \cdot \frac{\partial^2 V}{\partial w^2} \right) \cdot p^* = -\frac{1}{\gamma} \left((\mu_1 - \mu_2) \cdot \frac{\partial V}{\partial w} + \left(\hat{\varrho}\gamma + \theta \right) \right.$$
$$\left. \cdot \hat{w} \cdot \frac{\partial^2 V}{\partial w \hat{w}} + \theta \cdot w \cdot \frac{\partial^2 V}{\partial w^2} \right), \tag{A12}$$

assuming $A(t) > 0$. Here γ and θ are defined in (53). (A11) implies that the relevant partial derivatives of V are of the form

$$\frac{\partial^2 V}{\partial w^2} = 2A(t), \quad \frac{\partial V}{\partial w} = 2A(t)w + B(t) + D(t)\hat{w}, \quad \frac{\partial^2 V}{\partial w \hat{w}} = D(t). \tag{A13}$$

Substituting (A13) into (A12), the optimal control p^* obtained is in the form of (54), where h and g are given by (55). Then, it only remains to determine the functions A, B, D. Substituting (A12) into PIDE (51), we can obtain the following ordinary differential equations (ODE) for A, B, D,

$$\begin{cases} \dfrac{\mathrm{d}A(t)}{\mathrm{d}t} = -(2\mu_2 - \eta)A(t) - 1, & A(T) = 0, \\ \dfrac{\mathrm{d}D(t)}{\mathrm{d}t} = -(2\mu_2 - \eta)D(t) + 2e^{\beta t}, & D(T) = 0, \\ \dfrac{\mathrm{d}B(t)}{\mathrm{d}t} = -(\mu_2 - \phi)B(t) - 2cA(t) - cD(t), & B(T) = 0, \end{cases} \tag{A14}$$

Solving the ODE system gives us the A, B, D defined in (56) and (57). We also note that $A(t) > 0$, thus completing the proof.

A.5. *Insights from CD-optimal control*

The CD-optimal control (54) provides insights into the behavior of the optimal allocation policy. For ease of exposition, we first establish the following properties of $g(t; \beta)$ and $h(t; \beta, c)$.

COROLLARY A.1 (Properties of $g(t; \beta)$) *The function $g(t; \beta)$ defined in* (55) *has the following properties for $t \in [t_0, T]$ and $\beta > 0$:*

(i) *For fixed $t \in [t_0, T]$, $g(t; \beta)$ is strictly increasing on $\beta \in (0, \infty)$.*
(ii) *For fixed $\beta > 0$, $g(t; \beta)$ is strictly increasing on $t \in [t_0, T]$.*
(iii) *$g(t; \beta)$ admits the following bounds:*

$$e^{\beta t} \le g(t; \beta) \le e^{\beta T}. \tag{A15}$$

COROLLARY A.2 (Properties of $h(t; \beta, c)$) *The function $h(t; \beta, c)$ defined in* (55) *has the following properties for $t \in [t_0, T]$, $\beta > 0$ and $c \ge 0$:*

(i) *For fixed $t \in [t_0, T]$ and $c > 0$, $h(t; \beta, c)$ is strictly increasing on $\beta \in (0, \infty)$.*
(ii) *$h(t; \beta, c) \ge 0$, $\forall (t, \beta, c) \in [t_0, T] \times (0, \infty) \times [0, \infty)$.*
(iii) *For fixed $t \in [t_0, T]$ and $\beta > 0$, $h(t; \beta, c)$ is strictly increasing on $c \in [0, \infty)$. $h(t; \beta, 0) \equiv 0$. Moreover, $h(t; \beta, c) \propto c$, i.e. $h(t; \beta, c)$ is proportional to c.*

The proof of Corollaries A.1 and A.2 follow similar steps as in van Staden *et al.* (2024), in which the authors derive the CD-optimal control under the assumption that the stock price follows the double-exponential jump-diffusion model and the bond is risk-free with the bond price $B(t)$ following

$$\frac{\mathrm{d}B(t)}{B(t)} = r. \tag{A16}$$

Under such assumptions, van Staden *et al.* (2024) shows that the CD-optimal control can be expressed in a similar form as in 50) with g and h functions. The g and h functions satisfy the same properties as in Corollaries A.1 and A.2.

In order to analyze the closed-form solution, we make the following assumptions.

ASSUMPTION A.1 (Drift rates of the two assets) We assume that the drift rates of the stock and the bond index μ_1 and μ_2 satisfy the following properties,

$$\mu_1 - \mu_2 > 0, \quad \mu_1 - \mu_2 + \vartheta > 0, \tag{A17}$$

where ϑ is defined in (54).

REMARK A.1 (Remark on drift rate assumptions) The first inequality $\mu_1 - \mu_2 > 0$ indicates that the stock index has a higher drift rate than the bond index, which is a standard assumption.† The second inequality $\mu_1 - \mu_2 + \vartheta > 0$ is also practically reasonable. ϑ is a variance term that is usually on a smaller scale compared to the drift

† In fact, in this two-asset case, this assumption does not cause loss of generality.

rates. In reality, it is unlikely that $\mu_1 - \mu_2 > 0$ but $\mu_1 - \mu_2 + \vartheta \leq 0$.†

Now we proceed to summarize the insights from the CD-optimal control (54). The first obvious observation is that the CD-optimal control is a contrarian strategy. This can be seen from the fact that fixing time and the wealth of the benchmark portfolio $\hat{W}(t)$, the allocation to the more risky stock index decreases when the wealth of the active portfolio $W^*(t)$ increases.

If we take a deeper look at (54), we can see that the CD-optimal control consists of two components: a cash injection component p^*_{cash} and a tracking component p^*_{track}. Mathematically,

$$p^*(t, w, \hat{w}, \hat{\varrho}) = p^*_{cash}(t, w, \hat{w}) + p^*_{track}(t, w, \hat{w}, \hat{\varrho}), \tag{A18}$$

where

$$\begin{cases} p^*_{cash}(t, w, \hat{w}) = \frac{1}{W^*(t)} \left[\frac{(\mu_1 - \mu_2)}{\gamma} h(t; \beta, c) \right], \\ p^*_{track}(t, w, \hat{w}, \hat{\varrho}) = \frac{1}{W^*(t)} \left[\frac{(\mu_1 - \mu_2 + \vartheta)}{\gamma} \left(g(t; \beta) \hat{W}(t) - W^*(t) \right) \right. \\ \quad \left. + g(t; \beta) \hat{W}(t) \cdot \hat{\varrho} \right]. \end{cases} \tag{A19}$$

Based on Assumption A.1 and Corollary A.2, the cash injection component p_{cash} is always non-negative. Furthermore, from Corollary A.2, we know that the stock allocation from the cash injection component is proportional to the cash injection rate c. In addition, as $t \uparrow T$, $h(t; \beta, c)$ increases, and thus the stock allocation from the cash injection component also increases with time.

On the other hand, the tracking component p_{track} does not depend on the cash injection rate c, but only concerns the tracking performance of the active portfolio. One key finding is that

$$\begin{cases} p^*_{track}(t, w, \hat{w}, \hat{\varrho}) \geq \hat{\varrho}, & \text{if } W^*(t) \leq g(t; \beta) \hat{W}(t), \\ p^*_{track}(t, w, \hat{w}, \hat{\varrho}) < \hat{\varrho}, & \text{if } W^*(t) > g(t; \beta) \hat{W}(t). \end{cases} \tag{A20}$$

This means that the CD-optimal control uses $g(t; \beta)\hat{W}(t)$ as the true target for the active portfolio to decide if the active portfolio should take more or less risk than the benchmark portfolio. This is a key observation, since the CD objective function (48) measures the difference between $W(t)$ and $e^{\beta t}\hat{W}(t)$. One would naively think that the optimal strategy would be based on the deviation from $e^{\beta t}\hat{W}(t)$. In contrast, from Corollary A.1, we know that the true target $g(t; \beta)\hat{W}(t)$ used for decision making is greater than $e^{\beta t}\hat{W}(t)$. The insight from this observation is that if the manager wants to track an elevated target $e^{\beta t}\hat{W}(t)$, she should aim higher than the target itself.

A.6. *Strategy performance in low inflation regimes*

Until now, the discussion has been centered around the imaginary scenario of a long, persistent inflation environment. We have shown that the neural network strategy outperforms the benchmark strategy with a high probability under a high inflation regime. However, what if we are wrong? More dramatically, if the high-inflation situation ends immediately and we enter a low-inflation environment, how will the neural network strategy learned under high-inflation regimes perform?

To answer these questions, we evaluate the neural network strategy learned under high-inflation regimes on a testing data set bootstrapped from low-inflation historical regimes. Specifically, we exclude the two inflation regimes (1940:8–1951:7 and 1968:9–1985:10) from the full historical data of 1926:1–2022:1 and obtain several low-inflation data segments. We concatenate the low-inflation data segments and use the stationary bootstrap resampling method to generate a testing data set. We adopt the investment scenario described in table 5 and evaluate the performance of the neural network strategy obtained in section 5.3 on this low-inflation data set.

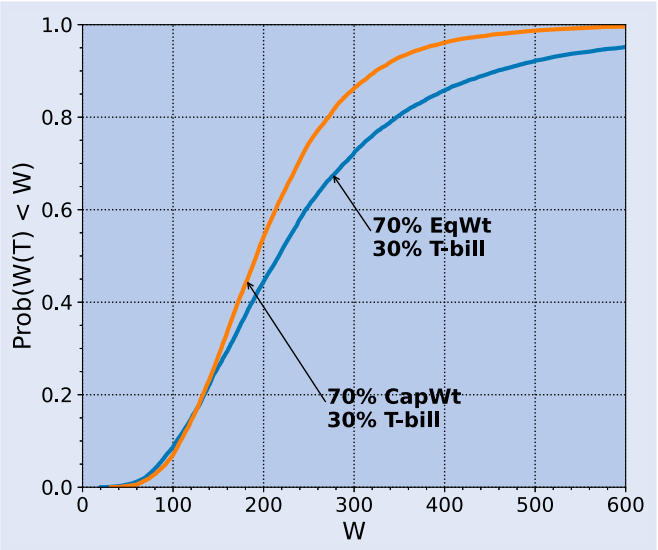


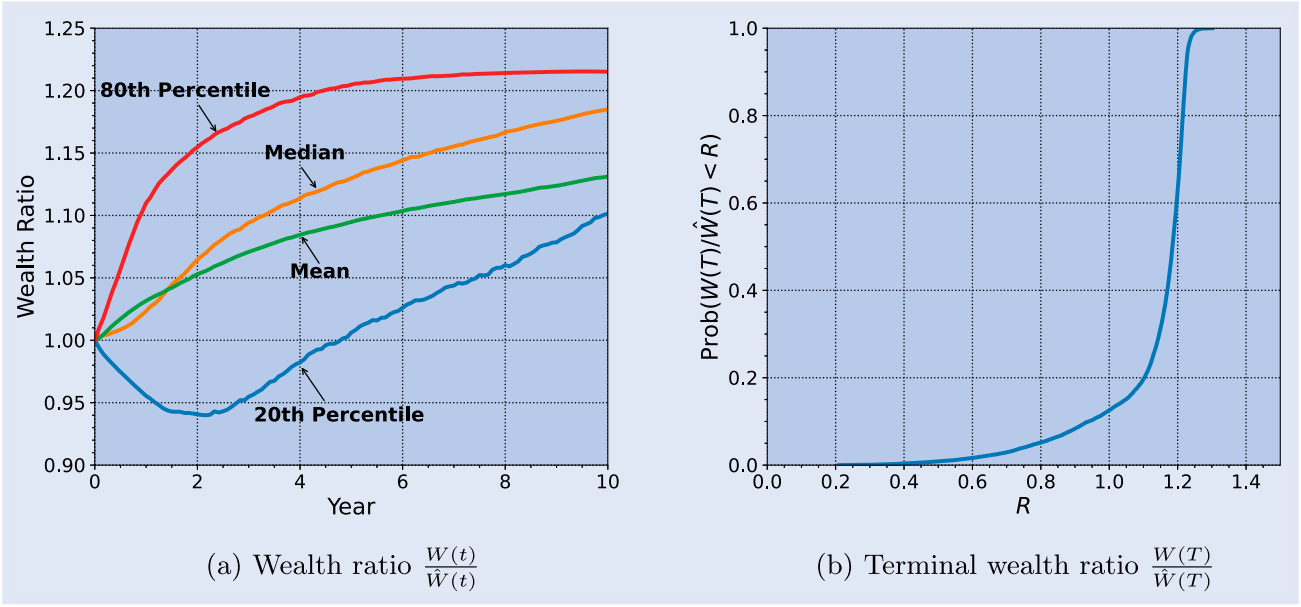
Figure A1. Cumulative distribution functions (CDFs) for cap-weighted and equal-weighted indexes, as a function of final real wealth W at $T = 10$ years. Initial stake $W_0 = 100$, no cash injections or withdrawals. Block bootstrap resampling, expected blocksize 6 months. 70% stocks, 30% bonds, rebalanced monthly. Bond index: 30-day U.S. T-bills. Stock index: CRSP capitalization-weighted or CRSP equal-weighted index. Underlying data excludes high inflation regimes. All indexes are deflated by the CPI. 10 000 resamples. Data set 1926:1–2022:1, excluding high inflation regimes (1940:8–1951:7 and 1968:9–1985:10).

Note that we continue to use the equal-weighted stock index/30-day T-bill fixed-mix portfolio as the benchmark. This is validated by figure A1, which plots the CDF of the terminal wealth of the fixed-mix portfolios using 70% equal-weighted stock index vs 70% cap-weighted stock index (both with 30% 30-day U.S. T-bill as the bond component). As we can see from figure A1, the fixed-mix portfolio with an equal-weighted stock index clearly has a more right-skewed distribution than the portfolio with a cap-weighted stock index. This seems to suggest that the equal-weighted index is the superior choice to use in the benchmark portfolio, even in low inflation regimes.

We then present the performance of the neural network strategy learned on high inflation data on the testing data set bootstrapped from low-inflation historical returns. Surprisingly, as we can see from figure A2(a), the performance of the neural network strategy learned under high-inflation regimes performs quite well in low-inflation environments. Compared to the testing results on the high inflation data set, there is a noticeable performance degradation; for example, the probability of outperforming the benchmark strategy in terminal wealth is now slightly less than 90%. However, the degradation is quite minimal. The neural network strategy still has more than an 85% chance of outperforming the benchmark strategy at the end of the investment horizon. As shown in table A2, the median IRR of the neural network strategy is still 2% higher than the median IRR of the benchmark strategy, meeting the investment target.

The above results indicate that the neural network strategy is surprisingly robust. Despite being specifically trained under a high-inflation scenario, the strategy performs admirably well in a low-inflation environment.

† For reference, based on the calibrated jump-diffusion model (49) on historical high-inflation regimes, $\mu_1 = 0.051, \mu_2 = -0.014, \vartheta = -0.00024$, and thus both inequalities are satisfied.



(a) Wealth ratio $\frac{W(t)}{\hat{W}(t)}$ (b) Terminal wealth ratio $\frac{W(T)}{\hat{W}(T)}$

Figure A2. Percentiles of wealth ratio over the investment horizon, and CDF of terminal wealth ratio. Results are based on the evaluation of the learned neural network model (from high-inflation data) on the low-inflation testing data set). (a) Wealth ratio $\frac{W(t)}{\hat{W}(t)}$ and (b) Terminal wealth ratio $\frac{W(T)}{\hat{W}(T)}$.

Table A2. Statistic of strategies. Results are based on the evaluation of the learned neural network model (from high-inflation data) on the low-inflation testing data set.

Strategy	Median[W_T]	E[W_T]	std[W_T]	5th Percentile	Median IRR (annual)
Neural network	429.7	489.6	301.9	151.6	0.100
Benchmark	368.3	420.8	238.2	175.7	0.079