A global-in-time neural network approach to dynamic portfolio optimization

Pieter M. van Staden[*] Peter A. Forsyth[†] Yuying Li[‡]

September 25, 2024

Abstract

We discuss a neural network approach, which does not rely on dynamic programming techniques, to solve dynamic portfolio optimization problems subject to multiple investment constraints. The approach allows for objectives of a very general form encompassing both time-consistent and time-inconsistent objectives, as well as objectives requiring multi-level optimization. The number of parameters of the neural network remains independent of the number of portfolio rebalancing events. Compared to reinforcement learning, this technique avoids the computation of high-dimensional conditional expectations. The approach remains practical when considering large numbers of underlying assets, long investment time horizons or very frequent rebalancing events. We prove convergence of the numerical solution to the theoretical optimal solution of a large class of problems under fairly general conditions, and present ground truth analyses for a number of popular formulations, including mean-variance, mean-semi-variance, and mean-conditional value-at-risk problems. Numerical experiments show that if the investment objective functional is separable in the sense of dynamic programming, the correct time-consistent optimal investment strategy is recovered, otherwise we obtain the correct pre-commitment (time-inconsistent) investment strategy. This method is agnostic as to the underlying data generating assumptions, and results are illustrated using (i) parametric models for underlying asset returns, (ii) stationary block bootstrap resampling of empirical returns, and (iii) generative adversarial network (GAN)-generated synthetic asset returns.

Keywords: Asset allocation, portfolio optimization, neural network, dynamic programming

JEL classification: G11, C61

1 Introduction

We develop a flexible neural network approach to obtain the numerical solution of a large class of dynamic (i.e. multi-period) portfolio optimization problems, while allowing for multiple investment constraints.

This method presents a significant generalization of our previous work (Li and Forsyth (2019)), and is also related to a large and growing existing literature on the use of neural networks to approximate the optimal control function directly in stochastic optimal control problems, avoiding use of dynamic programming methods (Buehler et al., 2019; Han et al., 2018; Han and Weinan, 2016; Reppen and Soner, 2023; Reppen et al., 2023; Tsang and Wong, 2020). In the taxonomy of Powell (2023), all of these methods are simply variations of the “policy function approximation” approach to stochastic optimal control, and upon cursory inspection are therefore expected to share many common properties.

However, in basic formulation, Buehler et al. (2019); Han and Weinan (2016); Tsang and Wong (2020) rely on a “sub”-neural network to approximate the control at each rebalancing step. Consequently, the number of neural network parameters required increases linearly with the number of portfolio rebalancing events.

Alternatively, a *single* neural network with time as an input feature can be used to approximate the optimal control (Buehler et al., 2019; Li and Forsyth, 2019; Reppen and Soner, 2023; Reppen et al., 2023)). In the taxonomy of Hu and Laurière (2023), these methods can be classified as “global-in-time” machine learning approaches to stochastic control problems. Such an approach implies that the optimal investment strategy at each rebalancing event would simply involve evaluating the trained neural network by specifying the time

[*] National Australia Bank, Melbourne, Victoria, Australia 3000. The research results and opinions expressed in this paper are solely those of the authors, are not investment recommendations, and do not reflect the views or policies of the NAB Group. pieter.vanstaden@gmail.com

[†] Cheriton School of Computer Science, University of Waterloo, Waterloo ON, Canada, N2L 3G1, paforsyt@uwaterloo.ca

[‡] Cheriton School of Computer Science, University of Waterloo, Waterloo ON, Canada, N2L 3G1, yuying@uwaterloo.ca

and other relevant input features. The NN approach presented in this paper is an example of a global-in-time approach, and compared to many other NN approaches, our approach can be viewed as parsimonious in the sense that the number of parameters does not scale with the number of rebalancing events. This ensures that our approach remains feasible even for problems with very long time horizons (see e.g. Forsyth et al. (2019)) or with a shorter time horizon but with frequent trading/rebalancing (Forsyth et al., 2011).

Moreover, global-in-time techniques, including the approach presented in this paper, only require the solution of a single optimization problem to determine the parameters of the neural network. This avoids the error amplification problems associated with the backward time-recursion in techniques based on dynamic programming (DP), e.g. Q-learning (Dixon et al., 2020; Gao et al., 2020; Park et al., 2020), or other DP-based techniques (Bachouch et al., 2022; Van Heeswijk and Poutré, 2019).

Given the context of the existing literature, the contributions of this paper are as follows:

- The existing literature focuses on objective functionals that are separable in the sense of dynamic programming, although this may not be required. In this paper, we consider a much larger class of objectives, with generalization along two dimensions:

  (i) We consider objectives of a very general form encompassing both time-consistent and time-inconsistent objectives (see Bjork et al. (2021)). We demonstrate how our method can be used to solve pre-commitment (time-inconsistent) problems to obtain the resulting induced time-consistent strategy (Bjork et al. (2021); Forsyth (2020); Strub et al. (2019a,b)) directly, without first requiring a theoretical derivation of the induced time-consistent problem. Additionally, we demonstrate the application of this method to objectives involving mean-semi-variance (e.g. Sortino ratio), for which no equivalent DP principle is known.

  (ii) We extend the class of objectives considered to include problems involving multi-level optimization such as Mean-CVaR in a dynamic setting (see for example Forsyth (2020); Miller and Yang (2017)). We also allow for a broader class of inner objectives which may not be separable in the sense of dynamic programming.

- We present theoretical results establishing the convergence of the proposed approach for the general class of objectives as discussed above. We prove convergence to the optimal strategy (assuming it exists) in the limit as the number neural network parameters (nodes in each layer) increases, provided that the number of samples in the training data also increases at the appropriate rate.

  The broad outlines of our proofs follow along the lines of convergence analyses in the literature (Reppen and Soner, 2023; Tsang and Wong, 2020). However, the details of the convergence proofs differ, in particular due to the nested structure of the objectives and the precise form of the techniques used. Note that while the main convergence results are necessarily obtained under some fairly strong assumptions, the convergence analysis remains valuable in that it proves useful theoretical convergence properties of the NN approach. As discussed in the next point, we also demonstrate that convergence is attainable in practical settings to sufficient levels of accuracy.

- Numerical examples show that the computed solutions using our technique confirm the theoretical equivalence results regarding the original and embedded formulations of the dynamic Mean-Variance problem. On the other hand, if no equivalent time-consistent formulation exists, then we obtain the correct pre-commitment (time-inconsistent) investment strategy.

  The neural network approach is also validated by comparing with analytical solutions which assume continuous rebalancing. This demonstrates that the global-in-time approach permits accurate solutions, even for the case of an infinite number of rebalancing times. As a result, our approach can be used without change for both frequent or infrequent rebalancing. We also verify that the approach generates comparable solutions to existing ground truth results for the Mean-CVaR problem.

  The approach merely assumes the existence of available training data, regardless of the assumptions of the underlying data generation method itself. In particular, there is no need to specify for example a set of parametric models for underlying asset dynamics, since the approach also works for model-independent data-driven generation methods, such as generative adversarial network (GAN)-generated synthetic asset returns or bootstrap-resampled paths of empirical asset returns. To demonstrate that the approach remains feasible and accurate regardless of the underlying data generation method, numerical examples are presented using data based on (i) parametric stochastic models for the underlying asset dynamics, (ii) bootstrap resampling of empirical asset returns, and (iii) GAN-generated synthetic asset returns.

The remainder of the paper is organized as follows: Section 2 discusses the large class of portfolio optimization problems that can be solved using this methodology, along with issues related to time-consistency and time-inconsistency of the optimal strategies. Section 3 formalizes the problem formulation, while Section 4 provides a summary of the proposed approach, with additional technical and practical details provided in Appendix A and Appendix B. Section 5 presents the convergence analysis of the proposed approach. Finally, Section 6 provides ground truth analyses, with Section 7 concluding the paper and discussing possible avenues for future research.

2 Problem overview and selected applications

The neural network approach and convergence analysis presented in this paper applies to the solutions of a large class of dynamic (i.e. multi-period) portfolio optimization problems that can be expressed in the following form,

$$\inf_{\xi\in\mathbb{R}}\inf_{\mathcal{P}\in\mathcal{A}}\left\{ E_{\mathcal{P}}^{t_0,w_0}\left[F\left(W\left(T\right),\xi\right)+G\left(W\left(T\right),E_{\mathcal{P}}^{t_0,w_0}\left[W\left(T\right)\right],w_0,\xi\right)\right]\right\}. \tag{2.1}$$

While rigorous definitions and assumptions are discussed in subsequent sections, for introductory purposes we simply note that in general, $F:\mathbb{R}^2\rightarrow\mathbb{R}$ and $G:\mathbb{R}^4\rightarrow\mathbb{R}$ denote some continuous functions and $\xi\in\mathbb{R}$ some auxiliary variable, with $T>0$ denoting the investment time horizon, $W\left(t\right),t\in\left[t_0,T\right]$, the controlled wealth process, and $\mathcal{P}$ representing the investment strategy (or control) implemented over $\left[t_0,T\right]$. Typically, $\mathcal{P}$ specifies the amount or fraction of wealth to invest in each of the underlying assets at each portfolio rebalancing event, which in practice occurs at some discrete subset of rebalancing times in $\left[t_0,T\right]$. $\mathcal{A}$ denotes the set of admissible investment strategies encoding the investment constraints faced by the investor. Finally, $E_{\mathcal{P}}^{t_0,w_0}\left[\cdot\right]$ denotes the expectation given control $\mathcal{P}$ and initial wealth $W\left(t_0\right)=w_0$.

We make the following general observations regarding (2.1):

- The function G forming part of the objective is allowed to be a nonlinear function of $E_{\mathcal{P}}^{t_0,w_0}\left[W\left(T\right)\right]$, which could result in an optimal control that is not time-consistent (Bjork et al. (2021)). The theoretical and practical benefits of solving problems which might be time-inconsistent are discussed in more detail below.

- For every fixed value of the auxiliary variable $\xi\in\mathbb{R}$ in the outer optimization problem of (2.1), the inner problem $\inf_{\mathcal{P}\in\mathcal{A}}\left\{\cdot\right\}$ takes on the structure of a standard (if possibly time-inconsistent) stochastic optimal control problem. As discussed below, this problem structure arises in the case of Mean-Conditional Value-at-Risk (CVaR) optimization.

- Although (2.1) is written for objective functions involving the terminal portfolio wealth $W\left(T\right)$, the approach and convergence analysis could be generalized without difficulty to objective functions that are wealth path-dependent, i.e. functions of $\left\{W\left(t\right):t\in\mathcal{T}\right\}$ for some subset $\mathcal{T}\subseteq\left[t_0,T\right]$ - see Forsyth et al. (2023); Van Staden et al. (2024) for examples. However, since a sufficiently rich class of problems are of the form (2.1), this will remain the main focus of this paper.

For purposes of concreteness, we highlight some specific examples of problems of the form (2.1):

(i) Utility maximization (see for example Vigna (2014)), in which case there is no outer optimization problem and $G\equiv0$, while $w\rightarrow U\left(w\right)$ denotes the investor's utility function, so that (2.1) therefore reduces to

$$\sup_{\mathcal{P}\in\mathcal{A}}\left\{E_{\mathcal{P}}^{t_0,w_0}\left[U\left(W\left(T\right)\right)\right]\right\}. \tag{2.2}$$

(ii) Mean-variance (MV) optimization (see e.g. Li and Ng (2000); Zhou and Li (2000)), with $\rho>0$ denoting the scalarization (or risk aversion) parameter, where the problem

$$\sup_{\mathcal{P}\in\mathcal{A}}\left\{E_{\mathcal{P}}^{t_0,w_0}\left[W\left(T\right)\right]-\rho\cdot Var_{\mathcal{P}}^{t_0,w_0}\left[W\left(T\right)\right]\right\}, \tag{2.3}$$

can also be written in the general form (2.1).

(iii) Mean-CVaR optimization, in which case we do have both an inner and an outer optimization problems (see e.g. Forsyth (2020); Miller and Yang (2017), resulting in a problem of the form

$$\inf_{\xi \in \mathbb{R}} \inf_{\mathcal{P} \in \mathcal{A}} \left\{ E_{\mathcal{P}}^{t_0, w_0} \left[F \left(W \left(T \right), \xi \right) \right] \right\}, \tag{2.4}$$

for a particular choice of the function F (see (3.15) below).

(iv) To illustrate the flexibility and generality of the proposed approach, we also consider a "mean semi-variance" portfolio optimization problem that is inspired by the popular Sortino ratio (Bodie et al. (2014)) in the case of one-period portfolio analysis, where only the variance of downside outcomes relative to the mean is penalized. In the case of dynamic trading strategies, this suggests an objective function of the form

$$\sup_{\mathcal{P} \in \mathcal{A}} \left\{ E_{\mathcal{P}}^{t_0, w_0} \left[W \left(T \right) - \rho \cdot \left(\min \left\{ W \left(T \right) - E_{\mathcal{P}}^{t_0, w_0} \left[W \left(T \right) \right], 0 \right\} \right)^2 \right] \right\}, \tag{2.5}$$

where, as in the case of (2.3), the parameter $\rho > 0$ encodes the trade-off between risk and return. Note that (2.5) is not separable in the sense of dynamic programming, and in the absence of embedding results (analogous to those of Li and Ng (2000); Zhou and Li (2000) in the case of MV optimization (2.3)), problem (2.5) cannot be solved using traditional dynamic programming-based methods.

However, we emphasize that (2.2)-(2.5) are only a selection of examples, and the proposed approach and theoretical analysis remains applicable to problems that can be expressed in the general form (2.1).

Portfolio optimization problems of the form (2.1) can give rise to investment strategies that are not time-consistent due to the presence of the (possibly non-linear) function G (Bjork et al. (2021)). This gives rise to two related problems:

(i) Since (2.1) cannot be solved directly using a dynamic programming-based approach, some other solution methodology has to be implemented, or some re-interpretation of the problem or the concept of "optimality" might be required. For example, the approach of Bjork et al. (2017, 2021); Bjork and Murgoci (2014), including recent developments such as Dai et al. (2023), can be interpreted as transforming a time-inconsistent problem such as (2.1) into a time-consistent problem by imposing a constraint of time-consistency on the strategies, which effectively changes the nature of the objective function - see for example Bjork and Murgoci (2014); Cong and Oosterlee (2016); Forsyth (2020); Van Staden et al. (2019) where this is explored in more detail. However, if the goal is to solve a truly time-inconsistent problem such as (2.1) without enforced time-consistency, a solution method that does not rely on dynamic programming is required.

(ii) If the investment strategies are time-inconsistent, this can raise questions as to whether these strategies are feasible to implement as practical investment strategies.

We make the following general observations:

- Provided a problem can be solved using dynamic programming (DP), a DP-based approach is arguably preferable in many low-dimensional settings, since it introduces significant simplification by reducing a dynamic problem to a recursive yet static inter-temporal problem. However, in higher dimensional settings or problems with many rebalancing events (whether due to a long time horizon or simply more frequent rebalancing), it may be desirable to avoid using DP even if (2.1) *can* be solved using DP techniques. For example, it is well known that DP has an associated "curse of dimensionality", in that as the number state variables increases linearly, the computational burden increases exponentially (Bellman (1957); Fernández-Villaverde et al. (2020); Han and Weinan (2016)). In addition, since DP techniques necessarily incur estimation errors at each time step, significant error amplification can occur which is further exacerbated in high-dimensional settings (Li et al., 2020; Tsang and Wong, 2020; Wang and Foster, 2020).

  Instead of relying on DP-based techniques and attempting to address the challenges of dimensionality using machine learning techniques (see for example Bachouch et al. (2022); Dixon et al. (2020); Fernández-Villaverde et al. (2020); Gao et al. (2020); Henry-Labordère (2017); Huré et al. (2021); Lucarelli and Borrotti (2020); Park et al. (2020)), the proposed method fundamentally avoids DP techniques altogether.

  This is especially relevant in our setting, since we have shown that in some cases, DP can be *unnecessarily* high-dimensional (see Van Staden et al. (2023)). This occurs since the objective functional (or performance criteria (Oksendal and Sulem (2019))) is typically high-dimensional while the optimal investment strategy remains relatively low-dimensional.

  The proposed method therefore forms part of the significant recent interest in developing machine learning techniques to solve multi-period portfolio optimization problems that avoids using DP techniques altogether (see for example Buehler et al. (2019); Ni et al. (2022); Reppen and Soner (2023); Reppen et al. (2023); Tsang and Wong (2020); Van Staden et al. (2023)).

- Time-inconsistent problems naturally arise in financial applications (see Bjork et al. (2021)), and as a result their solution is often an area of active research. Examples include the mean-variance problem, which remained an open problem for decades until the solution using the embedding technique of Li and Ng (2000); Zhou and Li (2000). As a result, being able to obtain a numerical solution to problems of the form (2.1) directly is potentially useful.

  From a practical point of view, in many cases, time-inconsistent problems generate an induced time consistent objective function (Forsyth (2020); Strub et al. (2019a,b)). The optimal policy for this induced time consistent objective function is identical to the pre-commitment policy at time zero. The induced time consistent strategy is, of course implementable (Forsyth (2020)), in the sense that the investor has no incentive to deviate from the strategy determined at time zero, at later times. As mentioned above, an alternative approach to handling time-inconsistent problems is to impose a constraint of time-consistency on the strategies which results in an the equilibrium control (see for example Bjork et al. (2021); Dai et al. (2023)). As shown in Bjork and Murgoci (2010), for every equilibrium control, there exists a standard, time consistent problem which has the same optimal control but under a different objective function.

  These considerations imply that the question of time-consistency is a often matter of perspective, since there may be alternative objective functions which give rise to the same pre-commitment control, yet are time-consistent. In fact, other subtle issues arise in comparing pre-commitment and time consistent controls, see Vigna (2020, 2022) for further discussion.

  Furthermore, over very short time horizons such as those encountered in optimal trade execution, time consistency or its absence may not be of much concern to the investor or market participant (see for example Forsyth et al. (2011); Tse et al. (2013)).

  In addition, as noted by Bernard and Vanduffel (2014), if the strategy is realized in an investment product sold to a retail investor, then the optimal policy from the investor's point of view is in fact of pre-commitment type, since the retail client does not herself trade in the underlying assets during the lifetime of the contract.

As a result of these observations, we will consider problem (2.1) in its general form. As discussed in the Introduction, we present ground truth analyses confirming that the proposed approach is very effective in solving portfolio optimization problems of the form (2.1). The results illustrate numerically that if (2.1) is not separable in the sense of DP, our approach recovers the correct pre-commitment (time-inconsistent) optimal control, otherwise it recovers the correct time-consistent optimal control.

3 Problem formulation

We start by formulating portfolio optimization problems of the form (2.1) more rigorously in a setting of discrete portfolio rebalancing and multiple investment constraints. Throughout, we work on filtered probability space $\left(\Omega, \mathcal{F}, \{\mathcal{F}(t)\}_{t\in[t_0,T]}, \mathbb{P}\right)$ satisfying the usual conditions, with $\mathbb{P}$ denoting the actual (and not the risk-neutral) probability measure.

Let $\mathcal{T}$ denote the set of N_{rb} discrete portfolio rebalancing times in $[t_0 = 0, T]$, which we assume to be equally-spaced to lighten notation,

$$\mathcal{T} \quad = \quad \{t_m = m\Delta t |\, m = 0, ..., N_{rb} - 1\}, \qquad \Delta t = T/N_{rb}, \tag{3.1}$$

where we observe that the last rebalancing event occurs at time $t_{N_{rb}-1} = T - \Delta t$.

At each rebalancing time $t_m \in \mathcal{T}$, the investor observes the $\mathcal{F}(t_m)$-measurable vector $\boldsymbol{X}(t_m) = (X_i(t_m) : i = 1, ..., \eta_X) \in \mathbb{R}^{\eta_X}$, which can be interpreted informally as the information taken into account by the investor in reaching their asset allocation decision. As a concrete example, we assume below that $\boldsymbol{X}(t_m)$ includes at least the wealth available for investment, an assumption which can be rigorously justified using analytical results (see for example Van Staden et al. (2023)).

Given $\boldsymbol{X}(t_m)$, the investor then rebalances a portfolio of N_a assets to new positions given by the vector

$$\boldsymbol{p}_m(t_m, \boldsymbol{X}(t_m)) = (p_{m,i}(t_m, \boldsymbol{X}(t_m)) : i = 1, .., N_a) \in \mathbb{R}^{N_a}, \tag{3.2}$$

where $p_{m,i}(t_m, \boldsymbol{X}(t_m))$ denotes the fraction of wealth $W(t_m)$ invested in the ith asset at rebalancing time t_m. The subscript "m" in the notation $\boldsymbol{p}_m$ emphasizes that in general, each rebalancing time $t_m \in \mathcal{T}$ could be associated with potentially a different function $\boldsymbol{p}_m : \mathbb{R}^{\eta_X+1} \to \mathbb{R}^{N_a}$, while the subscript is removed below when we consider a single function that is simply evaluated at different times, in which case we will write $\boldsymbol{p} : \mathbb{R}^{\eta_X+1} \to \mathbb{R}^{N_a}$.

For purposes of concreteness, we assume that the investor is subject to the constraints of (i) no short-selling and (ii) no leverage being allowed, although the proposed methodology can be adjusted without difficulty to treat different constraint formulations[1]. For illustrative purposes, we therefore assume that each allocation (3.2) is only allowed to take values in $(N_a - 1)$-dimensional probability simplex $\mathcal{Z}$,

$$\mathcal{Z} = \left\{(y_1, ..., y_{N_a}) \in \mathbb{R}^{N_a} : \sum_{i=1}^{N_a} y_i = 1 \text{ and } y_i \geq 0 \text{ for all } i = 1, ..., N_a\right\}. \tag{3.3}$$

In this setting, an investment strategy or control $\mathcal{P}$ applicable to $[t_0, T]$ is therefore of the form,

$$\mathcal{P} = \{\boldsymbol{p}_m(t_m, \boldsymbol{X}(t_m)) = (p_{m,i}(t_m, \boldsymbol{X}(t_m)) : i = 1, .., N_a) : t_m \in \mathcal{T}\}, \tag{3.4}$$

while the set of admissible controls $\mathcal{A}$ is defined by

$$\mathcal{A} = \{\mathcal{P} = \{\boldsymbol{p}_m(t_m, \boldsymbol{X}(t_m)) : t_m \in \mathcal{T}\} | \, \boldsymbol{p}_m(t_m, \boldsymbol{X}(t_m)) \in \mathcal{Z}, \forall t_m \in \mathcal{T}\}. \tag{3.5}$$

The randomness in the system is introduced through the returns of the underlying assets, $R_i, i \in \{1, ..., N_a\}$, which are adapted to the natural filtration. In the context of discrete rebalancing, let $R_i(t_m)$ denote the return of asset i over the interval $(t_{m-1}, t_m]$, observable at time t_m. We make no assumptions regarding the underlying asset dynamics, but at a minimum, we do require $(\mathbb{P})$ integrability, i.e. $\mathbb{E}|R_i(t_m)| < \infty$ for all $i \in \{1, ..., N_a\}$ and $m \in \{1, ..., N_{rb}\}$. Informally, we will refer to the set

$$\boldsymbol{Y} = \left\{(Y_i(t_m) \coloneqq 1 + R_i(t_m) : i = 1, ..., N_a)^\top : m \in \{1, ..., N_{rb}\}\right\} \tag{3.6}$$

as the *path* of (joint) asset returns over the investment time horizon $[t_0, T]$.

To clarify the subsequent notation, for any functional $\psi(t), t \in [t_0, T]$ we will use the notation $\psi(t^-)$ and $\psi(t^+)$ as shorthand for the one-sided limits $\psi(t^-) = \lim_{\epsilon \downarrow 0} \psi(t - \epsilon)$ and $\psi(t^+) = \lim_{\epsilon \downarrow 0} \psi(t + \epsilon)$, respectively.

Given control $\mathcal{P} \in \mathcal{A}$, asset returns $\boldsymbol{Y}$, initial wealth $W(t_0^-) \coloneqq w_0 > 0$ and a (non-random) cash contribution schedule $\{q(t_m) : t_m \in \mathcal{T}\}$, the portfolio wealth dynamics for $m = 0, ..., N_{rb} - 1$ are given by the general recursion

$$W(t_{m+1}^-; \mathcal{P}, \boldsymbol{Y}) = \left[W(t_m^-; \mathcal{P}, \boldsymbol{Y}) + q(t_m)\right] \cdot \sum_{i=1}^{N_a} p_{m,i}(t_m, \boldsymbol{X}(t_m)) \cdot Y_i(t_{m+1}). \tag{3.7}$$

Note that we write $W(u) = W(u; \mathcal{P}, \boldsymbol{Y})$ to emphasize the dependence of wealth on the control $\mathcal{P}$ and the (random) path of asset returns in $\boldsymbol{Y}$ that relates to the time period $t \in [t_0, u]$. In other words, despite using $\boldsymbol{Y}$ in the notation for simplicity, $W(u; \mathcal{P}, \boldsymbol{Y})$ is $\mathcal{F}(u)$-measurable. Finally, since there are no contributions or rebalancing at maturity, we simply have $W(t_{N_{rb}}^-) = W(T^-) = W(T) = W(T; \mathcal{P}, \boldsymbol{Y})$.

3.1 Investment objectives

Given this general investment setting and wealth dynamics (3.7), our goal is to solve dynamic portfolio optimization problems of the general form

$$\inf_{\xi \in \mathbb{R}} \inf_{\mathcal{P} \in \mathcal{A}} J(\mathcal{P}, \xi; t_0, w_0), \tag{3.8}$$

[1] As discussed in Section 4 and Appendix A, adjustments to the output layer of the neural network may be required.

where, for some given continuous functions $F:\mathbb{R}^2\to\mathbb{R}$ and $G:\mathbb{R}^3\to\mathbb{R}$, the objective functional J is given by

$$J\left(\mathcal{P},\xi;t_0,w_0\right) \quad = \quad E_{\mathcal{P}}^{t_0,w_0}\left[F\left(W\left(T;\mathcal{P},\boldsymbol{Y}\right),\xi\right)+G\left(W\left(T;\mathcal{P},\boldsymbol{Y}\right),E_{\mathcal{P}}^{t_0,w_0}\left[W\left(T;\mathcal{P},\boldsymbol{Y}\right)\right],w_0,\xi\right)\right]. \qquad (3.9)$$

Note that the expectations $E^{t_0,w_0}\left[\cdot\right]$ in (3.9) are taken over $\boldsymbol{Y}$, given initial wealth $W\left(t_0^-\right)=w_0$, control $\mathcal{P}\in\mathcal{A}$ and auxiliary variable $\xi\in\mathbb{R}$. In addition to the assumption of continuity of F and G, we will make only the minimal assumptions regarding the exact properties of J, including that $\xi\to F\left(\cdot,\xi\right)$ and $\xi\to G\left(\cdot,\cdot,w_0,\xi\right)$ are convex for all admissible controls $\mathcal{P}\in\mathcal{A}$, and the standard assumption (see for example Bjork et al. (2021)) that an optimal control $\mathcal{P}^*\in\mathcal{A}$ exists.

For illustrative and ground truth analysis purposes, we consider a number of examples of problems of the form (3.8)-(3.9).

As noted in the Introduction, the simplest examples of problems of the form (3.8) arise in the special case where $G\equiv 0$ and there is no outer optimization problem over ξ, such as in the case of standard utility maximization problems. As concrete examples of this class of objective functions, we will consider the quadratic target minimization (or quadratic utility) described in for example Vigna (2014); Zhou and Li (2000),

$$\left(DSQ\left(\gamma\right)\right): \qquad \inf_{\mathcal{P}\in\mathcal{A}}\left\{E^{t_0,w_0}\left[\left(W\left(T;\mathcal{P},\boldsymbol{Y}\right)-\gamma\right)^2\right]\right\}, \qquad \gamma>0, \qquad (3.10)$$

as well as the (closely-related) one-sided quadratic loss minimization used in for example Dang and Forsyth (2016); Li and Forsyth (2019),

$$\left(OSQ\left(\gamma\right)\right): \qquad \inf_{\mathcal{P}\in\mathcal{A}}\left\{E^{t_0,w_0}\left[\left(\min\left\{W\left(T;\mathcal{P},\boldsymbol{Y}\right)-\gamma,0\right\}\right)^2-\epsilon\cdot W\left(T;\mathcal{P},\boldsymbol{Y}\right)\right]\right\}, \qquad \gamma>0. \quad (3.11)$$

The term $\epsilon W(\cdot)$ in equation (3.11) ensures that the problem remains well-posed[2] in the event that $W(t)\gg\gamma$. Observe that problems of the form (3.10) or (3.11) are separable in the sense of dynamic programming, so that the resulting optimal control is therefore time-consistent.

As a classical example of the case where G is nonlinear and the objective functional (3.9) is not separable in the sense of dynamic programming, we consider the mean-variance (MV) objective with scalarization or risk-aversion parameter $\rho>0$ (see for example Bjork et al. (2017)),

$$\begin{aligned}\left(MV\left(\rho\right)\right): \qquad & \sup_{\mathcal{P}\in\mathcal{A}}\left\{E^{t_0,w_0}\left[W\left(T;\mathcal{P},\boldsymbol{Y}\right)\right]-\rho\cdot Var^{t_0,w_0}\left[W\left(T;\mathcal{P},\boldsymbol{Y}\right)\right]\right\}, \qquad \rho>0.\\ = \qquad & \sup_{\mathcal{P}\in\mathcal{A}} E_{\mathcal{P}}^{t_0,w_0}\left[W\left(T;\mathcal{P},\boldsymbol{Y}\right)-\rho\cdot\left(W\left(T;\mathcal{P},\boldsymbol{Y}\right)-E_{\mathcal{P}}^{t_0,w_0}\left[W\left(T;\mathcal{P},\boldsymbol{Y}\right)\right]\right)^2\right]. \qquad (3.12)\end{aligned}$$

Note that issues relating to the time-inconsistency of the optimal control of (3.12) are discussed in Remark 3.1 below, along with the relationship between (3.10) and (3.12).

As an example of a problem involving both the inner and outer optimization in (3.8), we consider the Mean - Conditional Value-at-Risk (or Mean-CVaR) problem, subsequently simply abbreviated the MCV problem. First, as a measure of tail risk, the CVaR at level α, or α-CVaR, is the expected value of the worst α percent of wealth outcomes, with typical values being $\alpha\in\{1\%,5\%\}$. As in Forsyth (2020), a *larger* value of the CVaR is preferable to smaller value, since our definition of α-CVaR is formulated in terms of the terminal *wealth*, not in terms of the *loss*. Informally, if the distribution of terminal wealth $W\left(T\right)$ is continuous with PDF $\hat{\psi}$, then the α-CVaR in this case is given by

$$\text{CVAR}_\alpha \quad = \quad \frac{1}{\alpha}\int_{-\infty}^{w_\alpha^*} w\cdot\hat{\psi}\left(w\right)\cdot dw, \qquad (3.13)$$

where w represents possible values of the random variable $W\left(T\right)$, and w_α^* is the corresponding Value-at-Risk (VaR) at level α defined such that $\int_{-\infty}^{w_\alpha^*}\hat{\psi}\left(w\right)dw=\alpha$. More formally, we follow for example Forsyth (2020) in defining the MCV problem with scalarization parameter $\rho>0$ as

$$\sup_{\mathcal{P}\in\mathcal{A}}\left\{\rho\cdot E^{t_0,w_0}\left[W\left(T\right)\right]+\text{CVAR}_\alpha\right\}, \qquad \rho>0. \qquad (3.14)$$

[2] Although this is a mathematical necessity (see e.g. (Li and Forsyth, 2019)), in practice, if we use a very small value of ϵ, then this has no perceptible effect on the summary statistics. In the numerical results of Section 6, we use $\epsilon=10^{-6}$; see Appendix B for a discussion.

However, instead of (3.13), we use the definition of CVaR from Rockafellar and Uryasev (2002) that is applicable to more general terminal wealth distributions, so that the MCV problem definition used subsequently aligns with the definition given in Forsyth (2020); Miller and Yang (2017)),

$$\left(MCV\left(\rho\right)\right): \qquad \inf_{\xi\in\mathbb{R}}\inf_{\mathcal{P}\in\mathcal{A}} E^{t_0,w_0}\left[-\rho\cdot W\left(T;\mathcal{P},\boldsymbol{Y}\right)-\xi+\frac{1}{\alpha}\max\left(\xi-W\left(T;\mathcal{P},\boldsymbol{Y}\right),0\right)\right], \quad \rho>0. \quad (3.15)$$

Finally, as noted in the Introduction, we apply the ideas underlying the Sortino ratio where the variance of returns below the mean are penalized, to formulate the following objective function for dynamic trading,

$$\left(MSemiV\left(\rho\right)\right): \sup_{\mathcal{P}\in\mathcal{A}}\left\{E_{\mathcal{P}}^{t_0,w_0}\left[W\left(T;\mathcal{P},\boldsymbol{Y}\right)-\rho\cdot\left(\min\left\{W\left(T;\mathcal{P},\boldsymbol{Y}\right)-E_{\mathcal{P}}^{t_0,w_0}\left[W\left(T;\mathcal{P},\boldsymbol{Y}\right)\right],0\right\}\right)^2\right]\right\}, \quad (3.16)$$

which we refer to as the "Mean- Semi-variance" problem, with scalarization (or risk-aversion) parameter $\rho>0$.[3]

The following remark discusses issues relating to the possible time-inconsistency of the optimal controls of (3.12) , (3.15) and (3.16).

Remark 3.1. (Time-*in*consistency and induced time-consistency) Formally, the optimal controls for problems $MV\left(\rho\right)$, $MCV\left(\rho\right)$ and $MSemiV\left(\rho\right)$ are not time-consistent, but instead are of the pre-commitment type (see Basak and Chabakauri (2010); Bjork and Murgoci (2014); Forsyth (2020)). However, in many cases, there exists an induced time consistent problem formulation which has the same controls at time zero as the pre-commitment problem (see Forsyth (2020); Strub et al. (2019a,b)).

As a concrete example of induced time-consistency, the embedding result of Li and Ng (2000); Zhou and Li (2000) establishes that the $DSQ\left(\gamma\right)$ objective is the induced time-consistent objective function associated with the $MV\left(\rho\right)$ problem, which is a result that we exploit for ground truth analysis purposes in Section 6.

Similarly, there is an induced time consistent objective function for the Mean-CVAR problem $MCV\left(\rho\right)$ in (3.15) - see Forsyth (2020).

Consequently, when we refer to a strategy as optimal, for either the Mean-CVAR ($MCV\left(\rho\right)$) or Mean-Variance ($MV\left(\rho\right)$) problems, this will be understood to mean that at any $t>t_0$, the investor follows the associated induced time-consistent strategy rather than a pre-commitment strategy.

In the Mean-Semi-variance ($MSemiV\left(\rho\right)$) case as per (3.16), there is no obvious induced time consistent objective function. In this case, we seek the pre-commitment policy.

For a detailed discussion of the many subtle issues involved in the case of time-inconsistency, induced time-consistency, and equilibrium controls, see for example Bjork et al. (2021); Bjork and Murgoci (2014); Forsyth (2020); Strub et al. (2019a,b); Vigna (2020, 2022). □

4 Neural network approach

In this section, we provide an overview of the neural network (NN) approach. Additional technical details and practical considerations are discussed in Appendices A and B, while the theoretical justification via convergence analysis will be discussed in Section 5 (and Appendix B).

Recall from (3.2) that $\boldsymbol{X}\left(t_m\right)\in\mathbb{R}^{\eta_X}$ denotes the information taken into account in determining the investment strategy (3.2) at rebalancing time t_m. Using the initial experimental results of Li and Forsyth (2019) and the analytical results of Van Staden et al. (2023) applied to this setting, we assume that $\boldsymbol{X}\left(t_m\right)$ includes at least the wealth available for investment at time t_m, so that

$$W\left(t_m^+;\mathcal{P},\boldsymbol{Y}\right)\coloneqq W\left(t_m^-;\mathcal{P},\boldsymbol{Y}\right)+q\left(t_m\right) \quad\in\quad \boldsymbol{X}\left(t_m\right), \qquad \forall t_m\in\mathcal{T}. \quad (4.1)$$

However, we emphasize that $\boldsymbol{X}\left(t_m\right)$ may include additional variables in different settings. For example, in non-Markovian settings or in the case of certain solution approaches involving auxiliary variables, it is natural to "lift the state space" by including additional quantities in $\boldsymbol{X}$ such as relevant historical quantities related to market variables, or other auxiliary variables - see for example Forsyth (2020); Miller and Yang (2017); Tsang and Wong (2020).

Let $\mathcal{D}_{\boldsymbol{\phi}}\subseteq\mathbb{R}^{\eta_X+1}$ be any given set such that $\left(t_m,\boldsymbol{X}\left(t_m\right)\right)\in\mathcal{D}_{\boldsymbol{\phi}}$ for all $t_m\in\mathcal{T}$. Let $C\left(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z}\right)$ denote the set of all continuous functions from $\mathcal{D}_{\boldsymbol{\phi}}$ to $\mathcal{Z}\subset\mathbb{R}^{N_a}$ (see (3.3)). We will use the notation $\boldsymbol{X}^*$ to denote the

[3] In continuous time, the unconstrained Mean-Semi-variance problem is ill-posed (Jin et al. (2005)). However, we will impose bounded leverage constraints, which is, of course, a realistic condition. This makes problem ($MSemiV\left(\rho\right)$) well posed.

information taken into account by the optimal control, since in the simplest case implied by (4.1), we simply have $\boldsymbol{X}^* = W^*$, where W^* denotes the wealth under the optimal strategy. We make the following assumption.

Assumption 4.1. *(Properties of the optimal control) Considering the general form of the problem (3.8), we assume that there exists an optimal feedback control $\mathcal{P}^* \in \mathcal{A}$. Specifically, we assume that at each rebalancing time $t_m \in \mathcal{T}$, the time t_m itself together with the information vector under optimal behavior $\boldsymbol{X}^*(t_m)$, which includes at least the wealth $W^*(t_m^+)$ available for investment (see (4.1)), are sufficient to fully determine the optimal asset allocation $\boldsymbol{p}_m^*(t_m, \boldsymbol{X}^*(t_m))$.*

Furthermore, we assume that there exists a continuous function $\boldsymbol{p}^* \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})$ such that $\boldsymbol{p}_m^*(t_m, \boldsymbol{X}^*(t_m)) = \boldsymbol{p}^*(t_m, \boldsymbol{X}^*(t_m))$ for all $t_m \in \mathcal{T}$, so that any given optimal control $\mathcal{P}^*$ can be expressed as

$$\mathcal{P}^* \quad = \quad \{\boldsymbol{p}^*(t_m, \boldsymbol{X}^*(t_m)) : \forall t_m \in \mathcal{T}\}, \quad \textit{where} \quad \boldsymbol{p}^* \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z}). \tag{4.2}$$

We make the following observations regarding Assumption 4.1:

(i) Continuity of $\boldsymbol{p}^*$ in space *and time*: While assuming the optimal control is a continuous map in the state space $\boldsymbol{X}$ is fairly standard in the literature, especially in the context of using neural network approximations (see for example Han and Weinan (2016); Huré et al. (2021); Tsang and Wong (2020)), the assumption of continuity in time in (4.2) is therefore worth emphasizing, since it identifies this approach as a "global in time" approach in the taxonomy of Hu and Laurière (2023), and relates this approach to some specific applications of Buehler et al. (2019); Reppen and Soner (2023); Reppen et al. (2023). This assumption enforces the requirement that in the limit of continuous rebalancing (i.e. when $\Delta t \to 0$), the control remains a continuous function of time, which is a practical requirement for any reasonable investment policy. In particular, this ensures that the asset allocation retains its smooth behavior as the number of rebalancing events in $[0, T]$ is increased, which we consider a fundamental requirement ensuring that the resulting investment strategy is reasonable. In addition, in Section 6 we demonstrate how the known theoretical solution to a problem assuming continuous rebalancing ($\Delta t \to 0$) can be approximated very well using $\Delta t \gg 0$ in the NN approach, even though the resulting NN approximation is only truly optimal in the case of $\Delta t \gg 0$.

(ii) The control is a *single* function for *all* rebalancing times; note that the function $\boldsymbol{p}^*$ is not subscripted by time. If the portfolio is rebalanced only at discrete time intervals, the investment strategy can be found (as suggested in (4.2)) by evaluating this continuous function at discrete time intervals, i.e. $(t_m, \boldsymbol{X}(t_m)) \to \boldsymbol{p}^*(t_m, \boldsymbol{X}(t_m)) = (p_i^*(t_m, \boldsymbol{X}(t_m)) : i = 1, ..., N_a)$, for all $t_m \in \mathcal{T}$. We discuss below how we solve for this (single) function directly, without resorting to dynamic programming, which avoids not only the challenge with error propagation due to value iteration over multiple timesteps, but also avoids solving for the high-dimensional conditional expectation (also termed the performance criteria by Oksendal and Sulem (2019)) if we are only interested in the relatively low-dimensional optimal control (see for example Van Staden et al. (2023)).

These observations ultimately suggest the NN approach discussed below, while the soundness of Assumption 4.1 is experimentally confirmed in the ground truth results presented in Section 6.

Given Assumption 4.1 and in particular (4.2), we therefore limit our consideration to controls of the form

$$\mathcal{P} \quad = \quad \{\boldsymbol{p}(t_m, \boldsymbol{X}(t_m)) : \forall t_m \in \mathcal{T}\}, \quad \text{for some} \quad \boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z}). \tag{4.3}$$

To simplify notation, we identify an arbitrary control $\mathcal{P}$ of the form (4.3) with its associated function $\boldsymbol{p} = (p_i : i = 1, ..., N_a) \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})$, so that the objective functional (3.9) is written as

$$J(\boldsymbol{p}, \xi; t_0, w_0) \quad = \quad E^{t_0, w_0}\left[F(W(T; \boldsymbol{p}, \boldsymbol{Y}), \xi) + G\big(W(T), E^{t_0, w_0}[W(T; \boldsymbol{p}, \boldsymbol{Y})], w_0, \xi\big)\right]. \tag{4.4}$$

In (4.4), $W(\cdot; \boldsymbol{p}, \boldsymbol{Y})$ denotes the controlled wealth process using a control of the form (4.3), so that the wealth dynamics (3.7) for $t_m \in \mathcal{T}$ (recall $t_{N_{rb}}^- = T$) now becomes

$$W\left(t_{m+1}^-; \boldsymbol{p}, \boldsymbol{Y}\right) \quad = \quad \left[W\left(t_m^-; \boldsymbol{p}, \boldsymbol{Y}\right) + q(t_m)\right] \cdot \sum_{i=1}^{N_a} p_i(t_m, \boldsymbol{X}(t_m)) \cdot Y_i(t_{m+1}). \tag{4.5}$$

Therefore, using Assumption 4.1 and (4.4)-(4.5), problem (3.8) is therefore expressed as

$$V\left(t_0, w_0\right) \quad = \quad \inf_{\xi\in\mathbb{R}} \inf_{\boldsymbol{p}\in C\left(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z}\right)} J\left(\boldsymbol{p}, \xi; t_0, w_0\right). \tag{4.6}$$

We now provide a brief overview of the proposed methodology to solve problems of the form (4.6). This consists of two steps discussed in the following subsections, namely (i) the NN approximation (with a finite number of unknown parameters) to the control, and (ii) computational estimate of the optimal control from training on a finite *number* n of joint paths of asset returns, $n \in \mathbb{N}$.

Intuitively, convergence analysis requires having both sufficient training data and a sufficiently complex NN to obtain a sufficiently accurate solution. In convergence analysis in §5, we assume, without loss of generality, a fixed number $\mathcal{L}^h \geq 1$ of hidden layers. Convergence analysis then requires that, as the number n of observations in the training data, $n \to \infty$ goes to infinity, the number of nodes $\hbar$ in each hidden layer of the NN also increases to infinity, but at a slower rate of for example $o\left(n^{1/4}\right)$ in the case of a NN with a single hidden layer.

Subsequently, we use n as an *index* for identifying a given NN structure for simplicity and denote the number of nodes in each hidden layer by $\hbar(n)$, explicitly indicating that there is a functional relationship between the number of training paths and the number of nodes in each layer, with increasing values of the index n signifying an increase in the number of nodes in each hidden layer of the NN. We simultaneously establish the validity of the NN approximation to the optimal control (Theorem 5.1) as well as the validity of the computational estimate (Theorem 5.2) by having a single parameter $n \to \infty$, which not only greatly reduces the notational burden, but also emphasizes the interconnectedness between different aspects of the NN approach. Note that the actual number of hidden nodes are significantly fewer than n.

Using n as an index, we subsequently provide details for NN approximation to control and computational estimate of the optimal control respectively.

4.1 Step 1: NN approximation to control

More formally, consider a fully-connected, feedforward NN $\boldsymbol{f}_n$ with parameter vector $\boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n}$ and a fixed number $\mathcal{L}^h \geq 1$ of hidden layers, where each hidden layer contains $\hbar\left(n\right) \in \mathbb{N}$ nodes. The NN has $\left(\eta_X + 1\right)$ input nodes, mapping feature (input) vectors of the form $\boldsymbol{\phi}\left(t\right) = \left(t, \boldsymbol{X}\left(t\right)\right) \in \mathcal{D}_{\boldsymbol{\phi}}$ to N_a output nodes. For a more detailed introduction to neural networks, see for example Goodfellow et al. (2016).

Additional technical and practical details can be found in Appendices A and B. For this discussion, we simply note that the index $n \in \mathbb{N}$ is used for the purposes of the analytical results and convergence analysis, where we fix a choice of $\mathcal{L}^h \geq 1$ while $\hbar\left(n\right), n \in \mathbb{N}$ is assumed to be a monotonically increasing sequence such that $\lim_{n\to\infty} \hbar\left(n\right) = \infty$ (see Section 5 and Appendix A). However, for practical implementation, a fixed value of $\hbar\left(n\right) \in \mathbb{N}$ is chosen (along with $\mathcal{L}^h \geq 1$) to ensure the NN has sufficient depth and complexity to solve the problem under consideration (see Appendix B).

Any NN considered is constructed such that $\boldsymbol{f}_n : \mathcal{D}_{\boldsymbol{\phi}} \to \mathcal{Z} \subset \mathbb{R}^{N_a}$. In other words, the values of the N_a outputs are automatically in the set $\mathcal{Z}$ defined in (3.3) for any $\boldsymbol{\phi} \in \mathcal{D}_{\boldsymbol{\phi}}$,

$$\boldsymbol{f}_n\left(\boldsymbol{\phi}\left(t\right); \boldsymbol{\theta}_n\right) \quad = \quad \left(f_{n,i}\left(\boldsymbol{\phi}\left(t\right); \boldsymbol{\theta}_n\right) : i = 1, ..., N_a\right) \in \mathcal{Z}. \tag{4.7}$$

As a result, the outputs of the NN $\boldsymbol{f}_n$ in (4.7) can be interpreted as portfolio weights satisfying the required investment constraints. A more detailed discussion of the NN structure can be found in Assumption A.1 in Appendix A, along with an illustration in Figure A.1.

For some fixed value of the index $n \in \mathbb{N}$, let $\mathcal{N}_n$ denote the set of NNs constructed in the same way as $\boldsymbol{f}_n$ for the fixed and given values of $\mathcal{L}^h$ and $\hbar\left(n\right)$. While a formal definition of the set $\mathcal{N}_n$ is provided in Appendix A, here we simply note that each NN $\boldsymbol{f}_n\left(\cdot; \boldsymbol{\theta}_n\right) \in \mathcal{N}_n$ only differs in terms of the parameter values constituting its parameter vector $\boldsymbol{\theta}_n$ (i.e. for a fixed n, each $\boldsymbol{f}_n \in \mathcal{N}_n$ has the same number of hidden layers $\mathcal{L}^h$, hidden nodes $\hbar\left(n\right)$, activation functions etc.).

Observing that $\mathcal{N}_n \subset C\left(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z}\right)$, our first step is to approximate (4.6) by performing the optimization over $\boldsymbol{f}_n\left(\cdot; \boldsymbol{\theta}_n\right) \in \mathcal{N}_n$ instead. In other words, we approximate the control $\boldsymbol{p}$ by a neural network $\boldsymbol{f}_n \in \mathcal{N}_n$,

$$\boldsymbol{p}\left(\boldsymbol{\phi}\left(t\right)\right) \quad \simeq \quad \boldsymbol{f}_n\left(\boldsymbol{\phi}\left(t\right); \boldsymbol{\theta}_n\right), \qquad \text{where} \qquad \boldsymbol{\phi}\left(t\right) = \left(t, \boldsymbol{X}\left(t\right)\right), \boldsymbol{p} \in C\left(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z}\right), \boldsymbol{f}_n \in \mathcal{N}_n. \tag{4.8}$$

We identify the NN $\boldsymbol{f}_n\left(\cdot; \boldsymbol{\theta}_n\right)$ with its parameter vector $\boldsymbol{\theta}_n$, so that the (approximate) objective functional using approximation (4.8) is written as

$$J_n\left(\boldsymbol{\theta}_n,\xi;t_0,w_0\right) \;=\; E^{t_0,w_0}\left[F\left(W\left(T;\boldsymbol{\theta}_n,\boldsymbol{Y}\right),\xi\right)+G\left(W\left(T;\boldsymbol{\theta}_n,\boldsymbol{Y}\right),E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n,\boldsymbol{Y}\right)\right],w_0,\xi\right)\right].\tag{4.9}$$

Combining (4.7) and (4.8), the wealth dynamics (4.5) is expressed as

$$W\left(t_{m+1}^{-};\boldsymbol{\theta}_n,\boldsymbol{Y}\right) \;=\; \left[W\left(t_m^{-};\boldsymbol{\theta}_n,\boldsymbol{Y}\right)+q\left(t_m\right)\right]\cdot\sum_{i=1}^{N_a} f_{n,i}\left(\boldsymbol{\phi}\left(t_m\right);\boldsymbol{\theta}_n\right)\cdot Y_i\left(t_{m+1}\right).\tag{4.10}$$

Using (4.8) and (4.9), for fixed and given values of $\mathcal{L}^h$ and $\hbar(n)$, we therefore approximate problem (4.6) by

$$\begin{aligned} V_n\left(t_0,w_0\right) &= \inf_{\xi\in\mathbb{R}}\inf_{\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n)\in\mathcal{N}_n} J_n\left(\boldsymbol{\theta}_n,\xi;t_0,w_0\right) &(4.11)\\ &= \inf_{\xi\in\mathbb{R}}\inf_{\boldsymbol{\theta}_n\in\mathbb{R}^{\nu_n}} J_n\left(\boldsymbol{\theta}_n,\xi;t_0,w_0\right) &\\ &= \inf_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} J_n\left(\boldsymbol{\theta}_n,\xi;t_0,w_0\right). &(4.12)\end{aligned}$$

We highlight that the optimization in (4.12) is unconstrained since, by construction, each NN $\boldsymbol{f}_n\left(\cdot;\boldsymbol{\theta}_n\right)\in\mathcal{N}_n$ always generates outputs in $\mathcal{Z}$.

The notation $\left(\boldsymbol{\theta}_n^*,\xi^*\right)$ and the associated NN $\boldsymbol{f}_n^*\left(\cdot;\boldsymbol{\theta}_n^*\right)\in\mathcal{N}_n$ are subsequently used to denote the values achieving the optimum in (4.12) for given values of $\mathcal{L}^h$ and $\hbar(n)$. Note however that we do *not* assume that the optimal control $\boldsymbol{p}^*\in C\left(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z}\right)$ satisfying Assumption 4.1 is also a NN in $\mathcal{N}_n$, since by the universal approximation results (see for example Hornik et al. (1989)), we would expect that the error in approximating (4.6) by (4.12) can be made arbitrarily small for sufficiently large $\hbar(n)$. These claims are rigorously confirmed in Section 5 below, where we consider a sequence of NNs $\boldsymbol{f}_n\left(\cdot;\boldsymbol{\theta}_n\right)\in\mathcal{N}_n$ obtained by letting $\hbar(n)\to\infty$ as $n\to\infty$ (for any fixed value of $\mathcal{L}^h\geq 1$).

4.2 Step 2 : Computational estimate of the optimal control

In order to solve the approximation (4.12) to problem (4.6), we require estimates of the expectations in (4.9). For computational purposes, suppose we take as given a set $\mathcal{Y}_n\in\mathbb{R}^{n\times N_a\times N_{rb}}$, consisting of $n\in\mathbb{N}$ independent realizations of the paths of joint asset returns $\boldsymbol{Y}$,

$$\mathcal{Y}_n \;=\; \left\{\boldsymbol{Y}^{(j)}: j=1,...,n\right\}.\tag{4.13}$$

We highlight that each entry $\boldsymbol{Y}^{(j)}\in\mathcal{Y}_n$ consists of a *path* of joint asset returns (see (3.6)), and we assume that the paths are independent, we do *not* assume that the asset returns constituting each path are independent. In particular, both cross-correlations and autocorrelation structures within each path of returns are permitted.

Constructing the set $\mathcal{Y}_n$ in practical applications is further discussed in Appendix B. In the numerical examples in Section 6, we use examples where $\mathcal{Y}_n$ is generated using (i) Monte Carlo simulation of parametric asset dynamics, (ii) stationary block bootstrap resampling of empirical asset returns, (Anarkulova et al. (2022)) and (iii) generative adversarial network (GAN)-generated synthetic asset returns (Yoon et al. (2019)). While we let $n\to\infty$ in (4.13) for convergence analysis purposes, in practical applications (e.g. the results of Section 6) we simply choose n sufficiently large such that we are reasonably confident that reliable numerical estimates of the expectations in (4.9) are obtained.

Given a NN $\boldsymbol{f}_n\left(\cdot;\boldsymbol{\theta}_n\right)\in\mathcal{N}_n$ and set $\mathcal{Y}_n$, the wealth dynamics (4.10) along path $\boldsymbol{Y}^{(j)}\in\mathcal{Y}_n$ is given by

$$W^{(j)}\left(t_{m+1}^{-};\boldsymbol{\theta}_n,\mathcal{Y}_n\right) \;=\; \left[W^{(j)}\left(t_m^{-};\boldsymbol{\theta}_n,\mathcal{Y}_n\right)+q\left(t_m\right)\right]\cdot\sum_{i=1}^{N_a} f_{n,i}\left(\boldsymbol{\phi}^{(j)}\left(t_m\right);\boldsymbol{\theta}_n\right)\cdot Y_i^{(j)}\left(t_{m+1}\right),\tag{4.14}$$

for $m=0,...,N_{rb}-1$. We introduce the superscript (j) to emphasize that the quantities are obtained along the jth entry of (4.13).

The computational estimate of $J_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0\right)$ in (4.9) is then given by

$$\begin{aligned}\hat{J}_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0, \mathcal{Y}_n\right) &= \frac{1}{n}\sum_{j=1}^{n} F\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), \xi\right) \\ &\quad + \frac{1}{n}\sum_{j=1}^{n} G\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), \frac{1}{n}\sum_{k=1}^{n} W^{(k)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi\right), \qquad (4.15)\end{aligned}$$

so that we approximate problem (4.12) by

$$\hat{V}_n\left(t_0, w_0; \mathcal{Y}_n\right) = \inf_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} \hat{J}_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0, \mathcal{Y}_n\right). \qquad (4.16)$$

The numerical solution of (4.16) can then proceed using standard (stochastic) gradient descent techniques. For subsequent reference, let $\left(\hat{\boldsymbol{\theta}}_n^*, \hat{\xi}_n^*\right)$ denote the optimal point in (4.16) relative to the training data set $\mathcal{Y}_n$ in (4.16). For more details and practical considerations regarding the construction of training and testing data sets, as well as the stochastic gradient descent technique used, please refer to Appendix B.

In the case of sufficiently large datasets (4.13), in other words as $n \to \infty$, we would expect that the error in approximating (4.12) by (4.16) can be made arbitrarily small. However, as noted above, as $n \to \infty$ and the number of hidden nodes $\hbar(n) \to \infty$ (for any fixed $\mathcal{L}^h \geq 1$), (4.12) is also expected to approximate (4.6) more accurately. As a result, we obtain the necessary intuition for establishing the convergence of (4.16) to (4.6) under suitable conditions, which is indeed confirmed in the results of Section 5.

Remark 4.1. (Extension to wealth path-dependent objectives) As noted in the Introduction, the NN approach as well as the convergence analysis of Section 5 can be extended to objective functions that depend on the entire wealth path $\{W(t) : t \in \mathcal{T}\}$ instead of just the terminal wealth $W(T)$. This is achieved by simply modifying (4.15) appropriately and ensuring the wealth is assessed at the desired intervals using (4.14). The practical consequences of applying this approach to path-dependent objectives include the additional input(s) that are typically required for the NN due to the augmented state space $\boldsymbol{X}$, along with a slightly more costly sampling process to obtain the NN training data set $\mathcal{Y}_n$. An example of the proposed approach being applied to a path-dependent objective function can be found in Van Staden et al. (2024). □

4.3 Advantages of the NN approach

The following observations highlight some advantages of the proposed NN approach:

(i) The approach does not rely on dynamic programming (DP) methods for the solution of problem (4.16), and therefore does not require value iteration or backward time stepping. In particular, we observe that due to the explicit time-dependence of the NN feature vector, the optimization problem (4.16) itself only indirectly depends on the number of rebalancing events, while time recursion is limited to the (computationally inexpensive) wealth dynamics (4.14). As result, problems relating to the error amplification associated with DP methods (Li et al. (2020); Tsang and Wong (2020); Wang and Foster (2020)) are avoided, and only a single optimization problem that is independent of the number of portfolio rebalancing events is solved, in contrast to DP-based methods (see for example Bachouch et al. (2022); Van Heeswijk and Poutré (2019)). In addition, NNs appear to be very effective in handling the Bellman's "curse of dimensionality" (Bellman (1957)), which can be summarized as the problem of exponential growth in computational complexity as the state variables (or dimensions) of the problem increase linearly in number. In fact, it has been argued that one of the principal reasons for considering NN-based methods is precisely their ability to deal with this challenge (Hu and Laurière (2023)), and there exists a rich literature of using a variety of NN-based methods for different applications to support this conclusion (see for example Han et al. (2018), Jentzen et al. (2021b), Becker et al. (2021) Hutzenthaler et al. (2020), Hu and Laurière (2023) and Reppen et al. (2023)).

    Not relying on DP techniques also makes the approach significantly more flexible, in that it can directly handle objective functions that are not separable in the sense of DP, without requiring theoretical results such as embedding in the case of MV optimization (see for example Li and Ng (2000); Zhou and Li (2000)). As an example of this, we present the solution of the mean - semi-variance problem (3.16) in Section 6.

(ii) The NN parameter vector $\boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n}$ does *not* depend on the rebalancing time $t_m \in \mathcal{T}$ or on the sample path j. In particular, the number of NN parameters does not increase with the number of rebalancing events. This contrasts our approach with the approaches of for example Han and Weinan (2016); Tsang and Wong (2020),[4] where the number of parameters scale with the number of rebalancing events. As a result, the NN approach presented here can lead to potentially significant computational advantages in the cases of (i) long investment time horizons or (ii) short trading time horizons with a frequent number of portfolio rebalancing events.

A natural question might be whether the NNs in the proposed approach are required to be very deep, thus potentially exposing the training of the NN in (4.16) to problem of vanishing or exploding gradients (see for example Goodfellow et al. (2016)). However, the ground truth results presented in Section 6 demonstrate that we obtain very accurate results with relatively shallow NNs (at most two hidden layers). We suspect this might be due to the optimal control being relatively low-dimensional compared to the high-dimensional objective functionals in portfolio optimization problems with discrete rebalancing (see Van Staden et al. (2023) for a rigorous analysis), while in this NN approach approach the optimal control is obtained directly without requiring the solution of the (high-dimensional) objective functional at rebalancing times.

Note that these advantages also contrast the NN approach with Reinforcement Learning-based algorithms to solve portfolio optimization problems, as the following remark discusses.

Remark 4.2. (Contrast of NN approach to Reinforcement Learning). Reinforcement learning (RL) algorithms (for example, Q-learning) relies fundamentally on the DP principle for the numerical solution of the portfolio optimization problem (see for example Gao et al. (2020); Lucarelli and Borrotti (2020); Park et al. (2020)). This requires, at each value iteration step, the approximation of a (high-dimensional) conditional expectation. As a result, RL is associated with standard DP-related concerns related to error amplification and the curse of dimensionality discussed above, and also cannot solve general problems of the form (2.1) without relying on for example an embedding approach to obtain an associated problem that can be solved using DP methods. □

5 Convergence analysis

In this section, we present the theoretical justification of the proposed NN approach as outlined in Section 4. We confirm that the numerical solution of (4.16) can be used to approximate the theoretical solution of (4.6) arbitrarily well (in probability) under suitable conditions. This section only summarizes the key convergence results which are among the main contributions of this paper, while additional technical details and proofs are provided in Appendix A. Note that while the theoretical convergence results in this section are necessarily obtained under some fairly strong assumptions, perhaps the strongest being the standard assumption (see for example Tsang and Wong (2020)) that the chosen numerical optimization technique attains a global minimum, it still demonstrates useful theoretical convergence properties of the proposed NN approach. To confirm that convergence is also attainable in practical settings, Section 6 presents a variety of numerical experiments where the NN approach easily attains an independently obtained or known result to a high degree of accuracy.

We start with Theorem 5.1, which confirms the validity of Step 1 (Subsection 4.1), namely using a NN $\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n) \in \mathcal{N}_n$ to approximate the control. Note that Theorem 5.1 relies on two assumptions, presented in Appendix A.2: We emphasize that Assumption A.3 is purely made for purposes of convenience, since its requirements can easily be relaxed with only minor modifications to the proofs (as discussed in Remark A.1), but at the cost of significant notational complexity and no additional insights. In contrast, Assumption A.2 is critical to establish the result of Theorem 5.1, and requires that the optimal investment strategy (or control) satisfies Assumption 4.1, places some basic requirements on F and G, and assumes that the sequence of NNs $\{\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n), n \in \mathbb{N}\}$ is constructed such that the number of nodes in each hidden layer $\hbar(n) \to \infty$ as $n \to \infty$ (no assumptions are yet required regarding the exact form of $n \to \hbar(n)$).

Theorem 5.1. *(Validity of NN approximation) We assume that Assumption A.2 holds, and for ease of exposition, we also assume that Assumption A.3 holds. Then the NN approximation to the control in (4.8) is valid, in the sense that $V(t_0, w_0)$ in (4.6) can be approximated arbitrarily well by $V_n(t_0, w_0)$ in (4.12) for sufficiently large n, since*

$$\begin{aligned}\lim_{n\to\infty} \left|V_n(t_0,w_0) - V(t_0,w_0)\right| &= \lim_{n\to\infty}\left|\inf_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} J_n(\boldsymbol{\theta}_n,\xi;t_0,w_0) - \inf_{\xi\in\mathbb{R}}\inf_{\boldsymbol{p}\in C(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z})} J(\boldsymbol{p},\xi;t_0,w_0)\right| \\ &= 0. \end{aligned} \tag{5.1}$$

[4] Tsang and Wong (2020) use a stacked NN approach, with a different NN at each rebalancing time.

Proof. See Appendix A.3. □

Having justified Step 1 of the approach, Theorem 5.2 now confirms the validity of Step 2 of the NN approach (see Subsection 4.2), namely using the computational estimate $\boldsymbol{f}_n^*\left(\cdot;\hat{\boldsymbol{\theta}}_n^*\right) \in \mathcal{N}_n$ from (4.16) as an approximation of the true optimal control $\boldsymbol{p}^* \in C\left(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z}\right)$. Note that in addition to the assumptions of Theorem 5.1, Theorem 5.2 also requires Assumption A.4, which by necessity includes computational considerations such as the structure of the training dataset $\mathcal{Y}_n$, the rate of divergence of the number of hidden nodes $\hbar(n) \to \infty$ as $n \to \infty$, and assumptions regarding the optimization algorithm used in solving problem (4.16).

Theorem 5.2. *(Validity of computational estimate) We assume that Assumption A.2, Assumption A.3 and Assumption A.4 hold. Then the computational estimate to the optimal control (4.2) obtained using (4.8) and (4.16) is valid, in the sense that the value function $V(t_0, w_0)$ in (4.6) can be approximated arbitrarily well in probability by $\hat{V}_n(t_0, w_0; \mathcal{Y}_n)$ in (4.16) for sufficiently large n, since*

$$
\begin{aligned}
\left|\hat{V}_n\left(t_0, w_0; \mathcal{Y}_n\right) - V\left(t_0, w_0\right)\right| &= \left| \inf_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\eta_n + 1}} \hat{J}_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0, \mathcal{Y}_n\right) - \inf_{\xi \in \mathbb{R}} \inf_{\boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})} J\left(\boldsymbol{p}, \xi; t_0, w_0\right) \right| \\
&\xrightarrow{P} 0, \qquad \text{as } n \to \infty. \qquad (5.2)
\end{aligned}
$$

Proof. See Appendix A.3. □

Taken together, Theorem 5.1 and Theorem 5.2 establish the theoretical validity of the NN approach to solve problems of the form (2.1).

6 Numerical results

In this section, we present numerical results obtained by implementing the NN approach described in Section 4. For illustrative purposes, the examples focus on investment objectives as outlined in Subsection 3.1. Since the approach does not depend on particular methods or assumptions for the generation of training data, the numerical results illustrate applications of the approach under three different data generation techniques for obtaining the training data set $\mathcal{Y}_n$ of the NN: (i) parametric stochastic models for the underlying asset dynamics, (ii) bootstrap resampling of empirical asset returns, and (iii) GAN-generated synthetic asset returns. Note that the question of whether one data generation technique is to be preferred over another in a particular setting is outside the scope of this paper. Instead, for the purposes of this paper, it is sufficient to emphasize that the approach can be used regardless of how the underlying data has been obtained.

The main focus of the numerical results is the demonstration of key aspects of the proposed NN method, such as (i) the ability to deal with cases where there are both inner and an outer optimization problems in the general objective (2.1) as in the case of Mean-CVaR, (ii) the ability to solve problems such as "mean semi-variance" (2.5) that cannot be solved using traditional dynamic programming-based methods, (iii) the ability to accurately approximate the analytical solution of a portfolio optimization problem under *continuous* rebalancing (i.e. theoretically infinite number of rebalancing events), and (iv) the ability to numerically recover the theoretical embedding result of Li and Ng (2000); Zhou and Li (2000). Given the theoretical results of Section 5, the NN approach is also expected to handle portfolio problems in higher dimensions than those illustrated in this section without difficulty, with practical applications being provided in our other work (Van Staden et al. (2023, 2024)).

6.1 Closed-form solution: $DSQ(\gamma)$ with continuous rebalancing

Under certain conditions, some of the optimization problems in Subsection 3.1 can be solved analytically. In this subsection, we demonstrate how a closed-form solution of problem $DSQ(\gamma)$ in (3.10), assuming *continuous* rebalancing (i.e. if we let $\Delta t \to 0$ in (3.1)), can be approximated very accurately using a very simple NN (1 hidden layer, only 3 hidden nodes) using discrete rebalancing with $\Delta t \gg 0$ in (3.1). Note that analytical solutions such as this particular example typically require unrealistic assumptions, including infinite leverage being allowed, trading continuing in the event of insolvency and the aforementioned continuous rebalancing. The goal of this example is not to discuss the merits of a particular analytical solution, but to simultaneously illustrate how parsimonious the NN approach is compared to alternative NN approaches where the parameter vector increases in size as the number of rebalancing events increases, as well as how useful the imposition of time-continuity is in ensuring the smooth behavior of the (approximate) optimal control by using the extreme and unrealistic case of continuous rebalancing. Note that all subsequent examples (see Sections 6.2, 6.3 and 6.4) impose multiple realistic investment constraints on the investment problems, including the assumption of discrete rebalancing.

In this subsection, we assume parametric dynamics for the underlying assets due to the use of an associated analytical solution (based on the same parametric dynamics) as the ground truth, but we emphasize that parametric underlying models are not required in Sections 6.3 and 6.4. For concreteness, we consider the scenario of two assets, $N_a = 2$, with unit values $S_i, i = 1, 2$, evolving according to the following dynamics,

$$\frac{dS_i\left(t\right)}{S_i\left(t^-\right)} \quad = \quad \left(\mu_i - \lambda_i \kappa_i^{(1)}\right) \cdot dt + \sigma_i \cdot dZ_i\left(t\right) + d\left(\sum_{k=1}^{\pi_i(t)} \left(\vartheta_i^{(k)} - 1\right)\right), \qquad i = 1, 2. \tag{6.1}$$

Note that (6.1) takes the form of the standard jump diffusion models in finance - see e.g. Kou (2002); Merton (1976) for more information. For each asset i in (6.1), μ_i and σ_i denote the (actual, not risk-neutral) drift and volatility, respectively, Z_i denotes a standard Brownian motion, $\pi_i\left(t\right)$ denotes a Poisson process with intensity $\lambda_i \geq 0$, and $\vartheta_i^{(k)}$ are i.i.d. random variables with the same distribution as ϑ_i, which represents the jump multiplier of the ith risky asset with $\kappa_i^{(1)} = \mathbb{E}\left[\vartheta_i - 1\right]$ and $\kappa_i^{(2)} = \mathbb{E}\left[\left(\vartheta_i - 1\right)^2\right]$. While the Brownian motions can be correlated with $dZ_1\left(t\right) dZ_2\left(t\right) = \rho_{1,2} \cdot dt$, we make the standard assumption that the jump components are independent (see for example Forsyth and Vetzal (2022)).

For this subsection only, we treat the first asset ($i = 1$ in (6.1)) as a "risk-free" asset, and set $\mu_1 = r > 0$ where r is the risk-free rate, so that we have $\lambda_1 = 0$, $\sigma_{1j} = 0\ \forall j$, and $Z_1 \equiv 0$, while the second asset ($i = 2$ in (6.1)) is assumed to be a broad equity market index (the "risky asset"). In this scenario, if problem $DSQ\left(\gamma\right)$ in (3.10) is solved subject to dynamics (6.1) together with the assumptions of costless continuous trading, infinite leverage, and uninterrupted trading in the event of insolvency, then the $DSQ\left(\gamma\right)$-optimal control can be obtained analytically as

$$\boldsymbol{p}^*\left(t, W^*\left(t\right)\right) = \left[1 - p_2^*\left(t, W^*\left(t\right)\right), p_2^*\left(t, W^*\left(t\right)\right)\right] \in \mathbb{R}^2, \tag{6.2}$$

where the fraction of wealth in the broad stock market index (asset $i = 2$) is given by (Zweng and Li (2011))

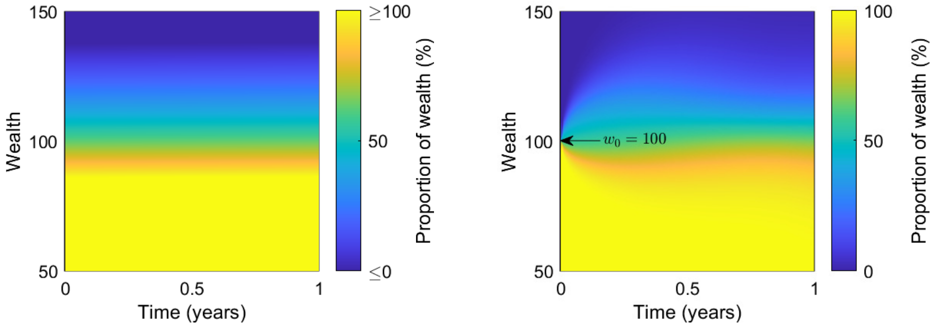
$$p_2^*\left(t, W^*\left(t\right)\right) = \frac{\mu_2 - r}{\sigma_2^2 + \lambda_2 \kappa_2^{(2)}} \cdot \left[\frac{\gamma e^{-r(T-t)} - W^*\left(t\right)}{W^*\left(t\right)}\right], \qquad w_0 < \gamma e^{-r(T-t)}. \tag{6.3}$$

By design, the NN approach is not constructed to solve problems with unrealistic assumptions such as continuous trading, infinite leverage and short-selling, or trading in the event of bankruptcy, all of which are required to derive (6.3). However, if the implicit quadratic wealth target for the DSQ problem (i.e. the value of γ, see Vigna (2014)) is not too aggressive, the analytical solution (6.3) does not require significant leverage or lead to a large probability of insolvency. In such a scenario, we can use the NN approach to approximate (6.3).

We select $w_0 = 100$, $T = 1$ year and $\gamma = 138.33$, and simulate $n = 2.56 \times 10^6$ paths of the underlying assets using (6.1) and parameters as in Table C.1 (Appendix C). On this set of paths, the true analytical solution (6.3) is implemented using 7,200 time steps. In contrast, for the NN approach, we use only 4 rebalancing events in $[0, T = 1]$, and therefore aggregate the simulated returns in quarterly time intervals to construct the training data set $\mathcal{Y}_n$. We consider only a very shallow NN, consisting of a single hidden layer and only 3 hidden nodes.

Figure 6.1 compares the resulting optimal investment strategies by illustrating the optimal proportion of wealth invested in the the broad equity market index (asset $i = 2$) as a function of time and wealth. We emphasize that the NN strategy in Figure 6.1(b) is not expected to be exactly identical to the analytical solution in Figure 6.1(a), since it is based on fundamentally different assumptions such as discrete rebalancing and investment constraints (3.5).

However, requiring that the NN feature vector includes time in the proposed NN approach, together with a NN parameter vector that does not depend on time, we guarantee the smooth behavior in time of the NN approximation observed in Figure 6.1(b). As a result, Table 6.1 shows that the shallow NN strategy trained with $\Delta t \gg 0$ results in a remarkably accurate approximation to the true analytical solution where $\Delta t \rightarrow 0$, since we obtain nearly identical optimal terminal wealth distributions. Note that the NN strategy in Figure 6.1(b) has been trained with a fixed initial wealth $w_0 = 100$, with no training data provided for wealth not equal to



(a) Closed-form solution, $\Delta t \to 0$ ($w_0 \in \mathbb{R}$) **(b)** Shallow NN, $\Delta t = 0.25$ ($w_0 = 100$)

Figure 6.1: Closed-form solution - $DSQ\,(\gamma)$ with continuous rebalancing: Optimal proportion of wealth invested in the broad equity market index as a function of time and wealth. The NN approximation is obtained for a specific initial wealth of w_0 =100, and only four rebalancing events in $[0, T]$.

Table 6.1: Closed-form solution - $DSQ\,(\gamma)$ with continuous rebalancing: Percentiles of the simulated ($n = 2.56 \times 10^6$) terminal wealth distributions obtained by implementing the optimal strategies in Figure 6.1. In both cases, a mean terminal wealth of 105 is obtained. Note that the NN approximation was obtained under the assumption of quarterly rebalancing only, no leverage or short-selling, and therefore no trading in insolvency.

		$W\,(T)$ percentiles				
Solution approach	Rebalancing	5th	20th	50th	80th	95th
Closed-form solution	Continuous, $\Delta t \to 0$	86.81	98.02	106.35	112.82	118.15
Shallow NN approximation	Discrete, $\Delta t = 0.25$, total of $N_{rb} = 4$ only	86.62	97.30	105.67	112.54	118.85

100 as $t \downarrow 0$. If higher accuracy is required in the NN solution, the initial wealth can be randomized during training to decrease the error in Figure 6.1(b) relative to Figure 6.1(a) further, but for most practical purposes the results obtained using Figure 6.1(b) can be considered as being sufficiently accurate (see Table 6.1).

6.2 Ground truth: Problem $MCV\,(\rho)$

In the case of the Mean-CVaR problem $MCV\,(\rho)$ in (3.15), Forsyth and Vetzal (2022) obtain an MCV-optimal investment strategy subject to the same investment constraints as in Section 3 (namely discrete rebalancing, no short-selling or leverage allowed, and no trading in insolvency) using the partial (integro-)differential equation (PDE) approach of Forsyth (2020). Since the PDE approach is based on the assumption of parametric underlying asset dynamics, we use the PDE solution as ground truth and assume the same underlying dynamics to generate the training data for the NN-based solution using the proposed approach. Note that parametric models for the underlying asset dynamics are not required in Sections 6.3 and 6.4.

For ground truth analysis purposes, we therefore consider the same investment scenario as in Forsyth and Vetzal (2022), where two underlying assets are considered, namely 30-day US T-bills and a broad equity market index (the CRSP VWD index) - see Appendix C for definitions. However, in contrast to the preceding section where one asset was taken as the risk-free asset, both assets are now assumed to evolve according to dynamics of the form (6.1), using the double-exponential Kou (2002) formulation for the jump distributions. The NN training data set is therefore constructed by simulating the same underlying dynamics. While further details regarding the context and motivation for the investment scenario can be found in Forsyth and Vetzal (2022), here we simply note that the scenario involves $T = 5$ years, quarterly rebalancing, a set of admissible strategies satisfying (3.5), and parameters for (6.1) as in Table C.2.

As discussed in Appendix B, the inherently higher complexity of the Mean-CVaR optimal control requires the NN to be deeper than in the case of the problem considered in Subsection 6.1. As a result, we consider approximating NNs with two hidden layers, each with 8 hidden nodes, while relatively large mini-batches of 2,000 paths were used in the stochastic gradient descent algorithm (see Appendix B) to ensure sufficiently accurate sampling of the tail of the returns distribution in selecting the descent direction at each step. Note that despite using a deeper NN, this NN structure is still very parsimonious and relatively shallow compared to the rebalancing time-dependent structures considered in for example Han and Weinan (2016), where a new set of parameters is introduced at each rebalancing event.

Table 6.2 compares the PDE results reported in Forsyth and Vetzal (2022) with the corresponding NN results. Note that the PDE optimal control was determined by solving a Hamilton-Jacobi-Bellman PDE numerically. The statistics for the PDE generated control were computed using $n = 2.56 \times 10^6$ Monte Carlo simulations of the joint underlying asset dynamics in order to calculate the results of Table 6.2, while the NN was trained on $n = 2.56 \times 10^6$ paths of the same underlying asset dynamics but which were independently simulated. While some variability of the results are therefore to be expected due to the underlying samples, the results in Table 6.2 demonstrate the robustness of the proposed NN approach.

Table 6.2: Ground truth - problem $MCV\left(\rho\right)$: The PDE results are obtained from Forsyth and Vetzal (2022) for selected points on the Mean-CVaR "efficient frontier". The "Value function" column reports the value of the objective function (3.14) under the corresponding optimal control, while "% difference" reports the percentage difference in the reported value functions for the NN solution compared to the PDE solution.

ρ	5% CVaR		$E^{t_0,w_0}\left[W\left(T\right)\right]$		Value function		% difference
	PDE	NN	PDE	NN	PDE	NN	
0.10	940.60	940.55	1069.19	1062.97	1047.52	1046.85	-0.06%
0.25	936.23	937.39	1090.89	1081.99	1208.95	1207.88	-0.09%
1.00	697.56	690.11	1437.73	1444.16	2135.29	2134.27	-0.05%
1.50	614.92	611.65	1508.10	1510.07	2877.07	2876.76	-0.01%

6.3 Ground truth: Problems $MV\left(\rho\right)$ and $DSQ\left(\gamma\right)$

In this subsection, we demonstrate that if the investment objective (2.1) is separable in the sense of dynamic programming, the correct time-consistent optimal investment strategy is recovered, otherwise we obtain the correct pre-commitment (time-inconsistent) investment strategy.

To demonstrate this, the theoretical embedding result of Li and Ng (2000); Zhou and Li (2000), which establishes the equivalence of problems $MV\left(\rho\right)$ and $DSQ\left(\gamma\right)$ under fairly general conditions, can be exploited for ground truth analysis purposes as follows. Suppose we solved problems $MV\left(\rho\right)$ and $DSQ\left(\gamma\right)$ on the same underlying training data set. We remind the reader that in the proposed NN approach, problem $MV\left(\rho\right)$ can indeed be solved directly without difficulty, which is not possible in dynamic programming-based approaches. Then, considering the numerical results, there should be values of parameters $\rho \equiv \tilde{\rho}$ and $\gamma \equiv \tilde{\gamma}$ such that the optimal strategy of $MV\left(\rho \equiv \tilde{\rho}\right)$ corresponds exactly to the optimal strategy of $DSQ\left(\gamma \equiv \tilde{\gamma}\right)$, with a specific relationship holding between $\tilde{\rho}$ and $\tilde{\gamma}$. The NN approach can therefore enable us to numerically demonstrate the embedding result of Li and Ng (2000); Zhou and Li (2000) in a setting where the underlying asset dynamics are not explicitly specified and where multiple investment constraints are present. We start by recalling the embedding result.

Proposition 6.1. *(Embedding result of Li and Ng (2000); Zhou and Li (2000)) Fix a value $\tilde{\rho} > 0$. If $\mathcal{P}^* \in \mathcal{A}$ is the optimal control of problem $MV\left(\rho \equiv \tilde{\rho}\right)$ in (3.12), then $\mathcal{P}^*$ is also the optimal control for problem $DSQ\left(\gamma = \tilde{\gamma}\right)$ in (3.10), provided that*

$$\tilde{\gamma} \quad = \quad \frac{1}{2\tilde{\rho}} + E^{t_0,w_0}\left[W^*\left(T;\mathcal{P}^*,\boldsymbol{Y}\right)\right]. \tag{6.4}$$

Proof. See Li and Ng (2000); Zhou and Li (2000). We also highlight the alternative proof provided in Dang and Forsyth (2016), which shows that this result is valid for any admissible control set $\mathcal{A}$. □

Since (6.4) is valid for any admissible control set $\mathcal{A}$, we consider a factor investing scenario where portfolios are constructed using popular long-only investable equity factor indices (Momentum, Value, Low Volatility, Size), a broad equity market index (the CRSP VWD index), 30-day T-bills and 10-year Treasury bonds (see Appendix C for definitions). For illustrative purposes in the case of an investor primarily concerned with long-run factor portfolio performance, we use a horizon of $T = 10$ years, $w_0 = 120$, annual contributions of $q\left(t_m\right) = 12$, and annual rebalancing.

Given historical returns data for the underlying assets, we construct training and testing (out-of-sample) data sets for the NN, $\mathcal{Y}_n$ and $\mathcal{Y}_{\hat{n}}^{test}$, respectively, using stationary block bootstrap resampling of empirical historical asset returns (see Appendix C), which is popular with practitioners (Anarkulova et al. (2022); Cavaglia et al. (2022); Cogneau and Zakalmouline (2013); Dichtl et al. (2016); Scott and Cavaglia (2017); Simonian and Martirosyan (2022)) and is designed to handle weakly stationary time series with serial dependence. See Ni et al. (2022) for a discussion concerning the probability of obtaining a repeated path in block bootstrap resampling (which is negligible for any realistic number of samples). Due to availability of historical data we use inflation-adjusted monthly empirical returns from 1963:07 to 2020:12. The training data set ($n = 10^6$) is obtained using an expected block size of 6 months of joint returns from 1963:07 to 2009:12, while the testing data set ($n = 10^6$) uses an expected block size of 3 months and returns from 2010:01 to 2020:12. We consider NNs with two hidden layers, each with only eight hidden nodes.

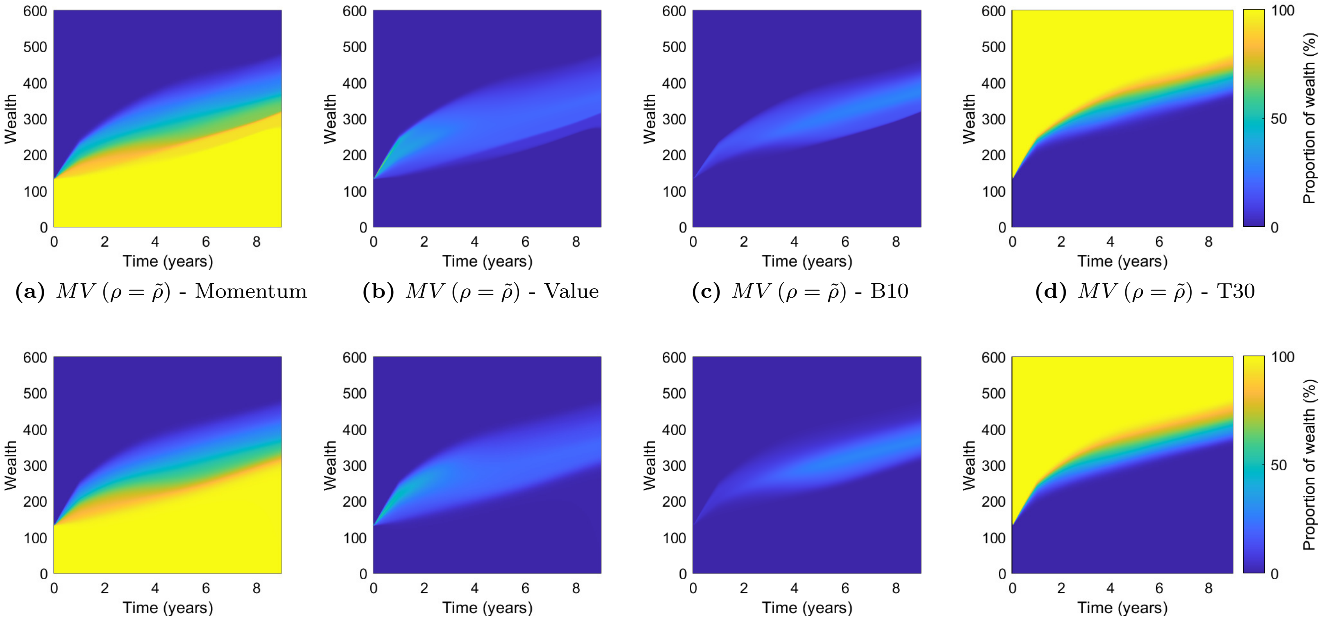
Choosing two values of $\tilde{\rho} > 0$ to illustrate different levels of risk aversion (see Table 6.3), we solve problem $MV\left(\rho = \tilde{\rho}\right)$ in (3.12) directly using the proposed approach to obtain the optimal investment strategy $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{mv}\right)$. Note that since we consider a fixed NN structure in this setting rather than a sequence of NNs, we drop the subscript "n" in the notation $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{mv}\right)$. Using this result together with (6.4), we can approximate the associated value of $\tilde{\gamma}$ by

$$\tilde{\gamma} \quad \simeq \quad \frac{1}{2\tilde{\rho}} + \frac{1}{n}\sum_{j=1}^{n} W^{*(j)}\left(T; \hat{\boldsymbol{\theta}}^*_{mv}, , \mathcal{Y}_n\right), \tag{6.5}$$

and solve problem $DSQ\left(\gamma = \tilde{\gamma}\right)$ independently using the proposed approach on the same training data set $\mathcal{Y}_n$.

According to Proposition 6.1, the resulting investment strategy $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{dsq}\right)$ should be (approximately) identical to the strategy $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{mv}\right)$ if the proposed approach works as required. Note that the parameter vectors are expected to be different (i.e. $\hat{\boldsymbol{\theta}}^*_{dsq} \neq \hat{\boldsymbol{\theta}}^*_{mv}$) due to a variety of reasons (multiple local minima, optimization using SGD, etc.), but the resulting wealth distributions and asset allocation should agree, i.e. $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{dsq}\right) \simeq \boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{mv}\right)$.

Figure 6.2 demonstrates the investment strategies $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{mv}\right)$ and $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{dsq}\right)$ obtained by training the NNs on the same training data set using values of $\tilde{\rho} = 0.017$ and $\tilde{\gamma} = 429.647$, respectively. Note that the values $\tilde{\rho}$ and $\tilde{\gamma}$ are rounded to three decimal places, and Figure 6.2 corresponds to Results set 1 in Table 6.3. In this example, only four of the underlying candidate assets have non-zero investments, which is to be expected due to the high correlation between long-only equity factor indices.



(a) $MV\left(\rho = \tilde{\rho}\right)$ - Momentum **(b)** $MV\left(\rho = \tilde{\rho}\right)$ - Value **(c)** $MV\left(\rho = \tilde{\rho}\right)$ - B10 **(d)** $MV\left(\rho = \tilde{\rho}\right)$ - T30

(e) $DSQ\left(\gamma = \tilde{\gamma}\right)$ - Momentum **(f)** $DSQ\left(\gamma = \tilde{\gamma}\right)$ - Value **(g)** $DSQ\left(\gamma = \tilde{\gamma}\right)$ - B10 **(h)** $DSQ\left(\gamma = \tilde{\gamma}\right)$ - T30

Figure 6.2: Ground truth - problems $MV\left(\rho = \tilde{\rho}\right)$ and $DSQ\left(\gamma = \tilde{\gamma}\right)$: investment strategies $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{mv}\right)$ and $\boldsymbol{f}\left(\cdot; \hat{\boldsymbol{\theta}}^*_{dsq}\right)$ obtained by training the NNs using values of $\tilde{\rho} = 0.017$ and $\tilde{\gamma} = 429.647$ (rounded to three decimal places), respectively. Each figure shows the proportion of wealth invested in the asset as a function of the minimal NN features, namely time and available wealth. Zero investment under the optimal strategies in the broad market index and the Size factor.

Table 6.3 confirms that the associated optimal terminal wealth distributions of $MV\left(\rho=\tilde{\rho}\right)$ and $DSQ\left(\gamma=\tilde{\gamma}\right)$ indeed correspond, both in-sample (training data set) and out-of-sample (testing data set).

Table 6.3: Ground truth - problems $MV\left(\rho=\tilde{\rho}\right)$ and $DSQ\left(\gamma=\tilde{\gamma}\right)$: Terminal wealth results obtained using $n=10^6$ joint paths for the underlying assets. Note that the values of $\tilde{\rho}$ and $\tilde{\gamma}$ are rounded to three decimal places, .

	Results set 1: $\tilde{\rho}=0.017$, $\tilde{\gamma}=429.647$				Results set 2: $\tilde{\rho}=0.0097$, $\tilde{\gamma}=493.196$			
$W\left(T\right)$	Training data		Testing data		Training data		Testing data	
distribution	MV	DSQ	MV	DSQ	MV	DSQ	MV	DSQ
Mean	400.2	400.3	391.2	391.6	441.5	441.8	441.8	441.5
Stdev	55.4	55.4	26.2	25.7	79.6	79.7	39.4	39.5
5th percentile	276.5	276.4	346.6	347.5	255.2	254.6	367.8	367.1
25th percentile	391.8	392.3	382.4	382.8	422.4	423.6	430.9	430.7
50th percentile	416.1	416.3	396.5	396.8	469.8	470.1	451.3	451.2
75th percentile	429.9	429.8	406.4	406.7	487.7	489.6	465.0	464.8
95th percentile	452.1	452.1	418.9	419.0	516.1	516.5	480.9	480.2

The proposed NN approach therefore clearly works as expected, in that we demonstrated that the result of Proposition 6.1 in a completely model-independent way in a portfolio optimization setting where no known analytical solutions exist. In particular, we emphasize that no assumptions were made regarding parametric underlying asset dynamics, the results are entirely data-driven. As a result, we can interpret the preceding results as showing that the approach correctly recovers the time-inconsistent (or pre-commitment) strategy without difficulty if the objective is not separable in the sense of dynamic programming, such as in the case of the $MV\left(\rho\right)$ problem, whereas if the objective is separable in the sense of dynamic programming, such as in the case of the $DSQ\left(\gamma\right)$ problem, the approach correctly recovers the associated time-consistent strategy.

6.4 Mean - Semi-variance strategies

Having demonstrated the reliability of the results obtained using the proposed NN approach with the preceding ground truth analyses, we now consider the solution of the Mean - Semi-variance problem (3.16). To provide the necessary context to interpret the $MSemiV\left(\rho\right)$-optimal results, we compare the results of the optimal solutions of the $MCV\left(\rho=\rho_{mcv}\right)$, $MSemiV\left(\rho=\rho_{msv}\right)$, and $OSQ\left(\gamma=\gamma_{osq}\right)$ problems, where the values of ρ_{mcv}, ρ_{msv} and γ_{osq} are selected to obtain the same expected value of terminal wealth on the NN training data set. This is done since the MCV- and OSQ-optimal strategies have been analyzed in great detail (Dang and Forsyth (2016); Forsyth (2020)), and are therefore well understood. Note that since all three strategies are related to the maximization of the mean terminal wealth and while simultaneously minimizing some risk measure (which is implicitly done in the case of the OSQ problem, see Dang and Forsyth (2016)), it is natural to compare the strategies on the basis of equal expectation of terminal wealth.

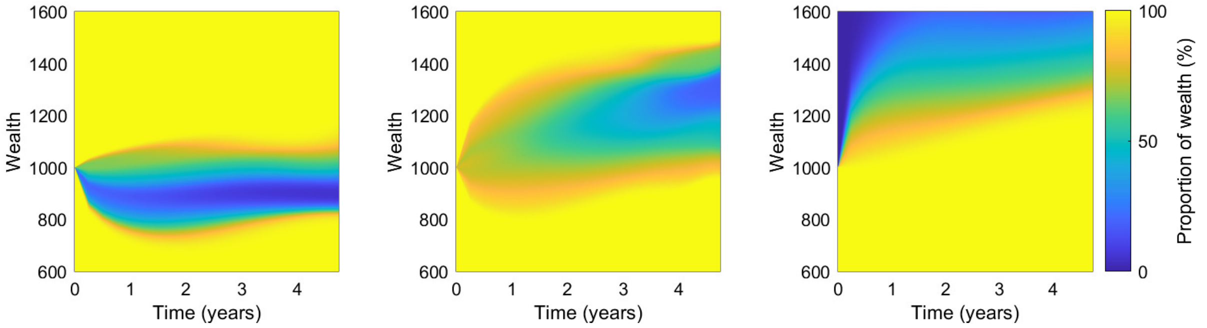
To highlight the main qualitative features of the $MSemiV\left(\rho\right)$-optimal results, we consider a simple investment scenario of two assets, namely 30-day T-bills and a broad equity market index (the VWD index) - see Appendix C for definitions. We choose T =5 years, $w_0=1000$, and zero contributions to demonstrate a lump sum investment scenario with quarterly rebalancing.

To illustrate the flexibility of the NN approach to underlying data generating assumptions, the NN training data sets are constructed using generative adversarial network (GAN)-generated synthetic asset returns obtained by implementing the TimeGAN algorithm proposed by Yoon et al. (2019). In more detail, using empirical monthly asset returns from 1926:01 to 2019:12 for the underlying assets (data sources are specified in Appendix C), the TimeGAN is trained with default parameters as in Yoon et al. (2019) using block sizes of 6 months to capture both correlation and serial correlation aspects of the (joint) time series.[5] Once trained, the TimeGAN is then used to generate a set of $n=10^6$ paths of synthetic asset returns, which is used as the training data set to train the NNs corresponding to the MCV, MSemiV and OSQ-optimal investment strategies.

[5] It appears that the actual code in Yoon et al. (2019) implements the following steps: (i) takes as input actual price data, (ii) forms rolling blocks of price data and (iii) forms a single synthetic price path (which is the same length as the original path) by randomly sampling (without replacement) from the set of rolling blocks. Step (iii) corresponds to the non-overlapping block bootstrap using a fixed block size. This should be contrasted with stationary block bootstrap resampling of Politis and Romano (1994). Step (i) does not make sense as input to a bootstrap technique, since the data set is about 10 years long, with an initial price of \$50 and a final price of \$1200. We therefore changed Step (i), so that all data was converted to returns prior to being used as input.

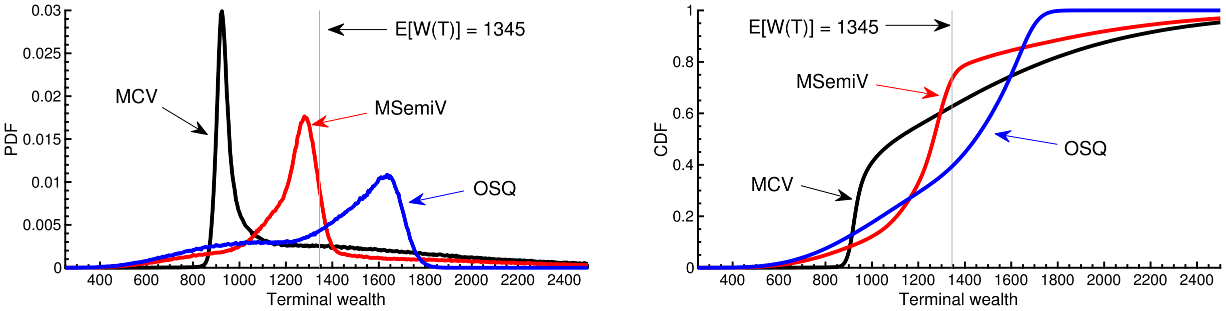
Figure 6.3 illustrates the resulting optimal investment strategies, and we observe that the MSemiV-optimal strategy is fundamentally different from the MCV and OSQ-optimal strategies, while featuring elements of both. Specifically, Figure 6.4, which illustrates the resulting optimal terminal wealth distributions (with the same expectation), demonstrates that the MSemiV strategy, like the MCV strategy, can offer better downside protection than the OSQ strategy, while the MSemiV strategy retains some of the qualitative elements of the OSQ distribution such as the left skew.

Having illustrated that the MSemiV problem can be solved in a dynamic trading setting using the proposed NN approach to obtain investment strategies that offer potentially valuable characteristics, we leave a more in-depth investigation of the properties and applications of MSemiV-optimal strategies for future work.



(a) *MCV* ($\rho = \rho_{mcv}$) **(b)** *MSemiV* ($\rho = \rho_{msv}$) **(c)** *OSQ* ($\gamma = \gamma_{osq}$)

Figure 6.3: Optimal investment strategies for the *MCV* ($\rho = \rho_{mcv}$), *MSemiV* ($\rho = \rho_{msv}$), and *OSQ* ($\gamma = \gamma_{osq}$) strategies, obtaining identical expectation of terminal wealth on the training data set. Each figure shows the proportion of wealth invested in the broad equity market index as a function of the minimal NN features, namely time and available wealth.



(a) PDFs of $W^*(T)$ **(b)** CDFs of $W^*(T)$

Figure 6.4: PDFs and CDFs of optimal terminal wealth obtained under the *MCV* ($\rho = \rho_{mcv}$), *MSemiV* ($\rho = \rho_{msv}$), and *OSQ* ($\gamma = \gamma_{osq}$) strategies, where the values of ρ_{mcv}, ρ_{msv} and γ_{osq} are selected to obtain the same expected value of optimal terminal wealth on the NN training data set.

7 Conclusion

In this paper, we presented a flexible NN approach, which does not rely on dynamic programming techniques, to solve a large class of dynamic portfolio optimization problems. We considered objectives of a very general form, encompassing both time-consistent and time-inconsistent objectives, as well as objectives requiring multi-level optimization. In the proposed approach, a single optimization problem is solved, issues of instability and error propagation involved in estimating high-dimensional conditional expectations are avoided, and the resulting NN is parsimonious in the sense that the number of parameters does not scale with the number of rebalancing events.

We also presented theoretical convergence analysis results which show that the numerical solution obtained using the proposed approach can recover the optimal investment strategy, provided it exists, regardless of whether the resulting optimal investment strategy is time-consistent or (formally) time-inconsistent.

Numerical results confirmed the advantages of the NN approach, and showed that accurate results can be obtained in ground truth analyses in a variety of settings. The numerical results also highlighted that the approach remains agnostic as to the underlying data generating assumptions, so that for example empirical asset returns or synthetic asset returns can be used without difficulty.

We conclude by noting that the NN approach is not necessarily limited to portfolio optimization problems such a those encountered during the accumulation phase of pension funds, and could be extended to address the significantly more challenging problems encountered during the decumulation phase of defined contribution pension funds (see for example Forsyth (2022)). We leave this extension for future work.

8 Declarations

The authors have no competing interests to declare that are relevant to the content of this article.

9 Acknowledgements

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

References

Alexander, S., T. Coleman, and Y. Li (2006). Minimizing CVaR and VaR for a portfolio of derivatives. *Journal of Banking and Finance* 30, 583–605.

Anarkulova, A., S. Cederburg, and M. S. O'Doherty (2022). Stocks for the long run? Evidence from a broad sample of developed markets. *Journal of Financial Economics* 143:1, 409–433.

Bachouch, A., C. Huré, N. Langrené, and H. Pham (2022). Deep neural networks algorithms for stochastic control problems on finite horizon: Numerical applications. *Methodology and Computing in Applied Probability* 24, 143–178.

Basak, S. and G. Chabakauri (2010). Dynamic mean-variance asset allocation. *Review of Financial Studies* 23, 2970–3016.

Beck, C., A. Jentzen, and B. Kuckuck (2022). Full error analysis for the training of deep neural networks. *Infinite dimensional analysis, quantum probability and related topics* 25(2).

Becker, S., P. Cheridito, A. Jentzen, and T. Welti (2021). Solving high-dimensional optimal stopping problems using deep learning. *European Journal of Applied Mathematics* 32(3), 470–514.

Bellman, R. (1957). A Markovian decision process. *Journal of Mathematics and Mechanics* (5), 679–684.

Bernard, C. and S. Vanduffel (2014). Mean-variance optimal portfolios in the presence of a benchmark with applications to fraud detection. *European Journal of Operational Research* 234, 469–480.

Bjork, T., M. Khapko, and A. Murgoci (2017). On time-inconsistent stochastic control in continuous time. *Finance and Stochastics* 21, 331–360.

Bjork, T., M. Khapko, and A. Murgoci (2021). *Time-inconsistent control theory with finance applications.* Springer Finance.

Bjork, T. and A. Murgoci (2010). A general theory of Markovian time inconsistent stochastic control problems. *Working paper* .

Bjork, T. and A. Murgoci (2014). A theory of Markovian time-inconsistent stochastic control in discrete time. *Finance and Stochastics* 18, 545–592.

Bodie, Z., A. Kane, and A. J. Marcus (2014). *Investments.* McGraw Hill New York, 10th edition edition.

Buehler, H., L. Gonon, J. Teichmann, and B. Wood (2019). Deep hedging. *Quantitative Finance* 19(8), 1271–1291.

Cavaglia, S., L. Scott, K. Blay, and S. Hixon (2022). Multi-asset class factor premia: A strategic asset allocation perspective. *The Journal of Portfolio Management* 48:9, 14–32.

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

Cong, F. and C. Oosterlee (2016). On pre-commitment aspects of a time-consistent strategy for a mean-variance investor. *Journal of Economic Dynamics and Control* 70, 178–193.

Dai, M., Y. Dong, and Y. Jia (2023). Learning equilibrium mean-variance strategy. *Mathematical Finance* 33(4), 1166–1212.

Dang, D. and P. Forsyth (2016). Better than pre-commitment mean-variance portfolio allocation strategies: A semi-self-financing Hamilton–Jacobi–Bellman equation approach. *European Journal of Operational Research* 250:1, 827–841.

Dichtl, H., W. Drobetz, and M. Wambach (2016). Testing rebalancing strategies for stock-bond portfolos across different asset allocations. *Applied Economics* 48, 772–788.

Dixon, M. F., I. Halperin, and P. Bilokon (2020). *Machine learning in finance.* Springer International Publishing.

Fama, E. and K. French (2015). A five-factor asset pricing model. *Journal of Financial Economics* 116(1), 1–22.

Fama, E. F. and K. R. French (1992). The cross-section of expected stock returns. *Journal of Finance* 47, 427–465.

Fama, E. F. and K. R. French (2010). Luck versus skill in the cross-section of mutual fund returns. *The Journal of Finance* 65(5), 1915–1947.

Fama, E. F. and K. R. French (2012). Size, value, and momentum in international stock returns. *Journal of Financial Economics* 105, 457–472.

Fernández-Villaverde, J., G. Nuño, G. Sorg-Langhans, and M. Vogler (2020). Solving high-dimensional dynamic programming problems using deep learning. *Working paper* .

Forsyth, P. (2020). Multiperiod mean conditional value at risk asset allocation: Is it advantageous to be time consistent? *SIAM Journal on Financial Mathematics* 11(2), 358–384.

Forsyth, P., J. Kennedy, S. Tse, and H. Windcliff (2011). Optimal trade execution: A mean quadratic variation approach. *Journal of Economic Dynamics and Control* 36:12, 1971–1991.

Forsyth, P. and K. Vetzal (2017). Dynamic mean variance asset allocation: Tests for robustness. *International Journal of Financial Engineering* 4:2. 1750021 (electronic).

Forsyth, P., K. Vetzal, and G. Westmacott (2019). Management of portfolio depletion risk through optimal life cycle asset allocation. *North American Actuarial Journal* 23(3), 447–468.

Forsyth, P. A. (2022). A stochastic control approach to defined contribution plan decumulation: The nastiest, hardest problem in finance. *North American Actuarial Journal* 26:2, 227–252.

Forsyth, P. A., P. M. Van Staden, and Y. Li (2023). Beating a constant weight benchmark: easier done than said. *International Journal of Theoretical and Applied Finance* Paper 350001 (electronic).

Forsyth, P. A. and K. R. Vetzal (2022). Multi-period mean expected-shortfall strategies: Cut your losses and ride your gains. *Applied Mathematical Finance* 29:5, 402–438.

Funahashi, K.-I. (1989). On the approximate realization of continuous mappings by neural networks. *Neural Networks* 2, 183–189.

Gao, B. and L. Pavel (2018). On the properties of the softmax function with application in game theory and reinforcement learning. *Working paper* ArXiv 1704.00805.

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

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep learning.* MIT press.

Granziol, D., X. Wan, S. Albanie, and S. Roberts (2021). Iterative averaging in the quest for best test error. *Working paper* .

Han, J., A. Jentzen, and E. Weinan (2018). Solving high-dimensional partial differential equations using deep learning. *PNAS* 115(34), 8505–8510.

Han, J. and E. Weinan (2016). Deep learning approximation for stochastic control problems. *NIPS Deep Reinforcement Learning Workshop* .

Henry-Labordère, P. (2017). Deep primal-dual algorithm for BSDEs: Application of machine learning to CVA and IM. *Working paper* .

Homer, S. and R. Sylla (2015). *A History of Interest Rates.* New York: Wiley.

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

Hornik, K., M. Stinchcombe, and H. White (1989). Multilayer feedforward networks are universal approximators. *Neural Networks* 2, 359–366.

Hu, R. and M. Laurière (2023). Recent developments in machine learning methods for stochastic control and games. *Working paper* .

Huré, C., H. Pham, A. Bachouch, and N. Langrené (2021). Deep neural networks algorithms for stochastic control problems on finite horizon: Convergence analysis. *SIAM Journal on Numerical Analysis* 59(1).

Hutzenthaler, M., A. Jentzen, T. Kruse, and T. A. Nguyen (2020). A proof that rectified deep neural networks overcome the curse of dimensionality in the numerical approximation of semilinear heat equations. *SN partial differential equations and applications* (1), 1–34.

Jentzen, A., B. Kuckuck, A. Neufeld, and P. von Wurstemberger (2021a). Strong error analysis for stochastic gradient descent optimization algorithms. *IMA Journal of Numerical Analysis* 41(1), 455–492.

Jentzen, A., D. Salimova, and T. Welti (2021b). A proof that deep artificial neural networks overcome the curse of dimensionality in the numerical approximation of kolmogorov partial differential equations with constant diffusion and nonlinear drift coefficients. *Communications in Mathematical Sciences* 19(5), 1167–1205.

Jin, H. Q., J. A. Yan, and X. Y. Zhou (2005). Continuous-time mean-risk portfolio selection. *Annales Henri Poincaré* 18, 171–183.

Kingma, D. P. and J. L. Ba (2015). Adam: A method for stochastic optimization. *Published as a conference paper at ICLR 2015* .

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

Kratsios, A. and E. Bilokopytov (2020). Non-euclidean universal approximation. *Proceedings of the 34th Conference on Neural Information Processing Systems* (NeurIPS 2020).

Leshno, M., V. Y. Lin, A. Pinkus, and S. Schocken (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks* 6, 861–867.

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

Li, Y. and P. Forsyth (2019). A data-driven neural network approach to optimal asset allocation for target based defined contribution pension plans. *Insurance: Mathematics and Economics* 86, 189–204.

Li, Z., K. H. Tsang, and H. Y. Wong (2020). Lasso-based simulation for high-dimensional multi-period portfolio optimization. *IMA Journal of Management Mathematics* 31(3), 257–280.

Lucarelli, G. and M. Borrotti (2020). A deep q-learning portfolio management framework for the cryptocurrency market. *Neural Computing and Applications* 32(23), 17229–17244.

Merton, R. (1976). Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics* 3, 125–144.

Miculescu, R. (2000). Approximation of continuous functions by Lipschitz functions. *Real Analysis Exchange* 26(1), 449–452.

Miller, C. and I. Yang (2017). Optimal control of conditional value-at-risk in continuous time. *SIAM Journal on Control and Optimization* 55(2), 856–884.

Mucke, N., G. Neu, and L. Rosasco (2019). Beating SGD saturation with tail-averaging and minibatching. *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)* .

Neu, G. and L. Rosasco (2018). Iterate averaging as regularization for stochastic gradient descent. *Proceedings of Machine Learning Research, 31st Annual Conference on Learning Theory* 75, 1–21.

Ni, C., Y. Li, P. Forsyth, and R. Carroll (2022). Optimal asset allocation for outperforming a stochastic benchmark target. *Quantitative Finance* 22:9, 1595–1626.

Oksendal, B. and A. Sulem (2019). *Applied Stochastic Control of Jump Diffusions.* Springer, 3rd edition.

Park, H., M. K. Sim, and D. G. Choi (2020). An intelligent financial portfolio trading strategy using deep q-learning. *Expert Systems with Applications* 158.

Politis, D. and J. Romano (1994). The stationary bootstrap. *Journal of the American Statistical Association* 89, 1303–1313.

Polyak, B. T. and A. B. Juditsky (1992). Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization* 30(4), 838–855.

Powell, W. (2023). A universal framework for sequential decision problems. *OR/MS Today* February. `https://tinyurl.com/PowellORMSfeature/`.

Reppen, A. M. and H. M. Soner (2023). Deep empirical risk minimization in finance: looking into the future. *Mathematical Finance* 33(1), 116–145.

Reppen, A. M., H. M. Soner, and V. Tissot-Daguette (2023). Deep stochastic optimization in finance. *Digital Finance* 5, 91–111.

Rockafellar, R. and S. Uryasev (2002). Conditional value-at-risk for general loss distributions. *Journal of Banking and Finance* 26:7, 1443–1471.

Scott, L. and S. Cavaglia (2017). A wealth management perspective on factor premia and the value of downside protection. *The Journal of Portfolio Management* 43:3, 1–9.

Shapiro, A. and Y. Wardi (1996). Convergence analysis of gradient descent stochastic algorithms. *Journal of Optimization Theory and Applications* 91(2).

Simonian, J. and A. Martirosyan (2022). Sharpe parity redux. *The Journal of Portfolio Management* 48:9, 183–193.

Sonoda, S. and N. Murata (2017). Neural network with unbounded activation functions is universal approximator. *Applied and Computational Harmonic Analysis* 43, 233–268.

Strub, M., D. Li, and X. Cui (2019a). An enhanced mean-variance framework for robo-advising applications. SSRN 3302111.

Strub, M. S., D. Li, X. Cui, and J. Gao (2019b). Discrete-time mean-CVaR portfolio selection and time-consistency induced term structure of the CVaR. *Journal of Economic Dynamics and Control* 108(103751).

Tsang, K. H. and H. Y. Wong (2020). Deep-learning solution to portfolio selection with serially dependent returns. *SIAM Journal on Financial Mathematics* 11(2), 593–619.

Tse, S., P. Forsyth, J. Kennedy, and H. Windcliff (2013). Comparison between the mean-variance optimal and the mean-quadratic-variation optimal trading strategies. *Applied Mathematical Finance* 20(5), 415–449.

Van Heeswijk, W. and H. L. Poutré (2019). Approximate dynamic programming with neural networks in linear discrete action spaces approximate dynamic programming with neural networks in linear discrete action spaces. *Working paper* .

Van Staden, P. M., D. Dang, and P. Forsyth (2019). Mean-quadratic variation portfolio optimization: A desirable alternative to time-consistent mean-variance optimization? *SIAM Journal on Financial Mathematics* 10(3), 815–856.

Van Staden, P. M., P. A. Forsyth, and Y. Li (2023). Beating a benchmark: dynamic programming may not be the right numerical approach. *SIAM Journal on Financial Mathematics* 14(2).

Van Staden, P. M., P. A. Forsyth, and Y. Li (2024). Across-time risk-aware strategies for outperforming a benchmark. *European Journal of Operational Research* 313(2), 776–800.

Vigna, E. (2014). On efficiency of mean-variance based portfolio selection in defined contribution pension schemes. *Quantitative Finance* 14(2), 237–258.

Vigna, E. (2020). On time consistency for mean-variance portfolio selection. *International Journal of Theoretical and Applied Finance* 23(6).

Vigna, E. (2022). Tail optimality and preferences consistency for intertemporal optimization problems. *SIAM Journal on Financial Mathematics* 13(1).

Wang, R. and D. P. Foster (2020). What are the statistical limits of offline RL with linear function approximation? *Working paper* .

Yoon, J., D. Jarrett, and M. Van der Schaar (2019). Time-series generative adversarial networks. *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)* `https://proceedings.neurips.cc/paper/2019/file/c9efe5f26cd17ba6216bbe2a7d26d490-Paper.pdf`.

Zhou, X. and D. Li (2000). Continuous time mean variance portfolio selection: a stochastic LQ framework. *Applied Mathematics and Optimization* 42, 19–33.

Zweng, Y. and Z. Li (2011). Asset liability management under benchmark and mean-variance criteria in a jump diffusion market. *Journal of Systems Science and Complexity* 24, 317–327.

Appendix A: NN approach: technical details and analytical results

In this appendix, additional analytical results, relating to the convergence analysis presented in Section 5, are presented.

A.1: NN structural assumptions

In this section, we discuss the NN structural assumptions. First, we introduce the necessary notation - for a more detailed treatment of NNs, see for example Goodfellow et al. (2016). Consider a fully-connected, feed-forward NN $\boldsymbol{f}_n$ with $\mathcal{L}^h \geq 1$ hidden layers. The NN layers are indexed by $\ell \in \{0, ..., \mathcal{L}\}$, where $\ell = 0$ and $\ell = \mathcal{L}^h + 1 \equiv \mathcal{L}$ denote the input and output layers, respectively. Let $\eta_{n,\ell} \in \mathbb{N}$ denote the number of nodes in layer ℓ of $\boldsymbol{f}_n$. With the exception of the input layer, each layer $\ell \in \{1, ..., \mathcal{L}\}$ is associated with a weights matrix $\boldsymbol{x}_n^{[\ell]} \in \mathbb{R}^{\eta_{n,\ell} \times \eta_{n,\ell-1}}$ into the layer, an optional bias vector $\boldsymbol{b}_n^{[\ell]} \in \mathbb{R}^{\eta_{n,\ell}}$, as well as an activation function $\mathfrak{a}_n^{[\ell]} : \mathbb{R}^{\eta_\ell} \rightarrow \mathbb{R}^{\eta_\ell}$ which is applied to the weighted inputs into the layer.

The parameter vector of the NN $\boldsymbol{f}_n$, which consists of all weights and biases, is denoted by $\boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n}$, where $\nu_n \in \mathbb{N}$ denotes the total number of weights and biases. In other words, the weights matrices $\left\{\boldsymbol{x}_n^{[\ell]} : \ell = 1, ..., \mathcal{L}\right\}$ and optional bias vectors $\left\{\boldsymbol{b}_n^{[\ell]} : \ell = 1, ..., \mathcal{L}\right\}$ are transformed into a single vector $\boldsymbol{\theta}_n = (\theta_1, ..., \theta_{\nu_n})$, where each $\theta_{n,i} \in \boldsymbol{\theta}_n$ can be uniquely mapped to a single weight or bias in some layer.

Note that no activation function is applied at the input layer ($\ell = 0$), so that the $\eta_0 \equiv \eta_{n,0}$ output values of the input layer corresponds to feature (input) vector of the NN, which will be denoted by $\boldsymbol{\phi} \in \mathbb{R}^{\eta_0}$. Recalling that $\eta_{\mathcal{L}} \equiv \eta_{n,\mathcal{L}}$ is the number of nodes in the output layer ($\ell = \mathcal{L}$) and setting the bias vectors $\boldsymbol{b}_n^{[\ell]} \equiv \boldsymbol{0}$ for convenience, the NN can therefore be written as a single function $\boldsymbol{f}_n(\boldsymbol{\phi}; \boldsymbol{\theta}_n) : \mathbb{R}^{\eta_0} \rightarrow \mathbb{R}^{\eta_{\mathcal{L}}}$, where

$$\boldsymbol{f}_n(\boldsymbol{\phi}; \boldsymbol{\theta}_n) \quad := \quad \left(f_{n,1}(\boldsymbol{\phi}; \boldsymbol{\theta}_n), ..., f_{n,\eta_{\mathcal{L}}}(\boldsymbol{\phi}; \boldsymbol{\theta}_n)\right), \qquad \boldsymbol{\phi} \in \mathbb{R}^{\eta_0}, \boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n} \tag{A.1}$$

We highlight that the output of the ith node in the output layer is given by $f_{n,i}(\boldsymbol{\phi}; \boldsymbol{\theta}_n) = \mathfrak{a}_{n,i}^{[\mathcal{L}]}$.

Given this standard fully-connected, feedforward NN formulation, we introduce the following NN structural assumption.

Assumption A.1. *(NN structure) Let $\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n), n \in \mathbb{N}$, be a sequence of fully-connected feedforward neural networks, and let $\hbar(n), n \in \mathbb{N}$ be a monotonically increasing sequence (i.e. $\hbar(n) < \hbar(n+1)$, $\forall n \in \mathbb{N}$) such that $\lim_{n\to\infty} \hbar(n) = \infty$. For each $n \in \mathbb{N}$, the NN $\boldsymbol{f}_n$ is constructed to satisfy the following structural assumptions.*

(i) The number of hidden layers $\mathcal{L}^h \geq 1$ ($\mathcal{L}^h \in \mathbb{N}$) remains fixed for all $n \in \mathbb{N}$. For notational simplicity, we assume that each of the $\mathcal{L}^h$ hidden layers of the NN $\boldsymbol{f}_n$ has the same number $\hbar(n)$ of hidden nodes,

$$\eta_{n,\ell} \quad \equiv \quad \hbar(n), \qquad \forall \ell = 1, ..., \mathcal{L}-1, \qquad \textit{for some } \hbar(n) \in \mathbb{N}. \tag{A.2}$$

(ii) For convenience, we assume that the sigmoid activation function σ^h is applied at each hidden node,

$$\sigma^h(y) = \frac{1}{1+e^{-y}} \quad \equiv \quad \mathfrak{a}_{n,i}^{[\ell]}(y), \quad \textit{where } y = \left(\sum_{k=1}^{\eta_{n,\ell-1}} x_{n,ik}^{[\ell]} \mathfrak{a}_{n,k}^{[\ell-1]}\right) + b_{n,i}^{[\ell]}, \tag{A.3}$$

for all $\ell = 1, ..., \mathcal{L}^h$ and $i = 1, ..., \hbar(n)$. Note that in principle, any of the popular activation functions can be used instead of (A.3), with minor modifications to the theoretical analysis presented in this paper.

(iii) The NN $\boldsymbol{f}_n$ has $\eta_0 = \eta_X + 1 \equiv \eta_{n,0}$ input nodes (i.e. the number of input nodes are independent of $n \in \mathbb{N}$), with feature (input) vectors $\boldsymbol{\phi} \in \mathbb{R}^{\eta_0}$ of the form

$$\boldsymbol{\phi} := \boldsymbol{\phi}(t) \quad := \quad (t, \boldsymbol{X}(t)) \in \mathcal{D}_{\boldsymbol{\phi}} \subseteq \mathbb{R}^{\eta_X+1}, \qquad \textit{with } \boldsymbol{X}(t) = \left(W(t^+), \hat{\boldsymbol{X}}(t)\right), \tag{A.4}$$

where $W(t^+)$ denotes the wealth available for investment at time t after any contributions to the portfolio at time t, while $\hat{\boldsymbol{X}}(t)$ denotes a vector of additional information taken into account by the investment strategy. We emphasize that (A.4) clarifies that at time $t \in [t_0, T]$, at least time t itself and $W(t^+)$ are always assumed to be inputs into the NN.

(iv) The NN $\boldsymbol{f}_n$ has $N_a = \eta_{n,\mathcal{L}}$ output nodes (i.e. the number of output nodes are independent of $n \in \mathbb{N}$), with the output of node i, denoted by $f_{n,i}(\boldsymbol{\phi}(t);\boldsymbol{\theta}_n)$, being associated with the proportion of available wealth $W(t^+)$ invested in asset $i \in \{1, ..., N_a\}$ after rebalancing the portfolio at time t.

(v) The output layer ($\ell = \mathcal{L} = \mathcal{L}^h + 1$) of each NN $\boldsymbol{f}_n$ uses the softmax activation function (see for example Gao and Pavel (2018)). Therefore we have $\mathfrak{a}_n^{[\mathcal{L}]} = \boldsymbol{\psi} : \mathbb{R}^{N_a} \to \mathbb{R}^{N_a}$, where the ith component of $\boldsymbol{\psi} = (\psi_i : i = 1, .., N_a)$ is given by

$$\psi_i = \mathfrak{a}_{n,i}^{[\mathcal{L}]} = \frac{\exp\left\{z_{n,i}^{[\mathcal{L}]}\right\}}{\sum_{m=1}^{N_a} \exp\left\{z_{n,m}^{[\mathcal{L}]}\right\}}, \quad \textit{where } z_{n,i}^{[\mathcal{L}]} = \sum_{k=1}^{N_a} x_{n,ik}^{[\mathcal{L}]} \mathfrak{a}_{n,k}^{[\mathcal{L}-1]} + b_{n,i}^{[\mathcal{L}]}, \quad i = 1, ..., N_a. \tag{A.5}$$

For a given $n \in \mathbb{N}$, we define the set $\mathcal{N}_n$ as the set of all neural networks satisfying Assumption (A.1),

$$\mathcal{N}_n = \left\{ \boldsymbol{f}_n : \mathcal{D}_{\boldsymbol{\phi}} \to \mathcal{Z} \middle| \, \boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n) \text{ satisfies Assumption } A.1 \text{ with } \hbar(n) \text{ nodes in each hidden layer} \right\}. \tag{A.6}$$

In other words, each $\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n) \in \mathcal{N}_n$ has the same number of hidden nodes $\hbar(n)$ in each hidden layer, but a potentially different parameter vector $\boldsymbol{\theta}_n$ (i.e. different values associated with the weights and biases).

We make the following observations regarding Assumption A.1, noting that an illustration of the NN structure is provided in Figure A.1:

- Any NN constructed to satisfy Assumption A.1 will, for any input vector $\boldsymbol{\phi}(t)$, automatically generate an output in the set $\mathcal{Z}$ defined in (3.3), hence the definition (A.6) noting that $\boldsymbol{f}_n : \mathcal{D}_{\boldsymbol{\phi}} \to \mathcal{Z}$. In other words, the given constraints are automatically satisfied. However, different sets of constraints would require modifications to the output activation, or post-processing of NN outputs, without affecting the technical results[6]. Since we are illustrating the approach using the particular form of $\mathcal{Z}$ in (3.3) because of its wide applicability (no short-selling and no leverage), a softmax output layer is used to ensure the NN output

[6] For example, position limits and limited leverage can be introduced using minor modifications to the output layer. Perhaps the only substantial challenge is offered by unrealistic investment scenarios, such as insisting that trading should continue in the event of bankruptcy, in which case consideration should be given to the possibility of wealth being identically zero or negative.

remains in $\mathcal{Z} \subset \mathbb{R}^{N_a}$ for any $\boldsymbol{\phi}(t)$ (see (4.7)). While we assume sigmoid activations for the hidden nodes for concreteness and convenience (see Assumption A.1), any of the commonly-used activation functions can be implemented with only minor modifications to the technical results presented in Section 5.

- Note that further assumptions regarding the rate of at which the sequence $\hbar(n)$ increases relative to that of the sequence $\{n\}_{n\in\mathbb{N}}$ will be introduced in the convergence analysis of Section 5 (see Assumption A.4.
- In practical applications, it is not necessary to consider a sequence of NNs; instead, we will use a single NN $\boldsymbol{f}_{\tilde{n}}$ with $\hbar(\tilde{n})$ hidden nodes in each of the hidden layers to get a reasonable trade-off between accuracy and computational efficiency. However, we emphasize that any such $\boldsymbol{f}_{\tilde{n}}$ is still constructed to satisfy Assumption A.1.

A.2: Assumptions for convergence analysis

Assumption A.2 introduces the main assumptions used in rigorously justifying the approximation (4.8) and therefore to prove Theorem 5.1.

Assumption A.2. *(Convergence analysis: NN approximation to control) To establish the validity of the NN approximation to the control, we make the following assumptions:*

(i) The optimal investment strategy (or control) satisfies Assumption 4.1.

(ii) The functions F and G in the objective functional $J(\boldsymbol{p},\xi;t_0,w_0)$ (see (4.4)) are continuous, and $\xi \to F(\cdot,\xi)$ and $\xi \to G(\cdot,\cdot,\cdot,\xi)$ are convex for any admissible strategy $\boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z})$. Note that in for example the Mean - Conditional Value-at-Risk problem (3.15) where there is an inner and outer optimization problem, this assumption is standard in computational settings (Forsyth (2020)).

(iii) The NN approximation (4.8) *of the investment strategy $\boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z})$ is implemented by a NN $\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n) \in \mathcal{N}_n$, where $\mathcal{N}_n$ is given by (A.6). In other words, each approximating NN in the sequence of NNs $\boldsymbol{f}_n, n \in \mathbb{N}$ is constructed according to Assumption A.1.*

Note that Assumption A.2(iii) specifically requires that Assumption A.1 is satisfied, so each $\boldsymbol{f}_n, n \in \mathbb{N}$, has $\hbar(n)$ nodes in each hidden layer, where we recall that the sequence $\hbar(n), n \in \mathbb{N}$, is monotonically increasing and satisfies $\hbar(n) \to \infty$ as $n \to \infty$. However, we make no further assumptions yet regarding the form of $n \to \hbar(n)$.

For ease of exposition, we introduce Assumption A.3 below. We emphasize that Assumption A.3 is purely for the sake of convenience, with Remark A.1 below discussing briefly how each component of Assumption A.3 can be relaxed with only minor (but tedious and notationally demanding) modifications to the subsequent proofs.

Assumption A.3. *(Convergence analysis: Assumptions for ease of exposition) For convenience, we introduce the following assumptions which can be relaxed without difficulty, as discussed in Remark A.1 below.*

(i) We assume that the optimal control $\mathbf{p}^*$ as per Assumption 4.1 is a function of time and wealth only, i.e. $\boldsymbol{X}^*(t_m) = W^*(t_m^+)$ for each $t_m \in \mathcal{T}$ in (4.2). As a result, we work with the minimal form of the NN feature vector satisfying Assumption A.1. Specifically, in the subsequent results we will always assume that $\boldsymbol{X}(t) = W(t^+)$, so that we will consider feature vectors (A.4) of the form

$$\boldsymbol{\phi}(t) \quad = \quad \left(t, W\left(t^+;\boldsymbol{\theta}_n,\boldsymbol{Y}\right)\right) \in \mathcal{D}_{\boldsymbol{\phi}} \subseteq \mathbb{R}^2. \tag{A.7}$$

(ii) The wealth process with dynamics given by (4.10) remains bounded. In other words, we assume that there exists a value $w_{max} > 0$ such that

$$0 \leq W(t;\boldsymbol{\theta}_n,\boldsymbol{Y}) \leq w_{max} \quad a.s. \qquad for\ all\ t \in [t_0,T], \boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n}, \tag{A.8}$$

so that $\mathcal{D}_{\boldsymbol{\phi}}$ in (A.7) satisfies

$$\mathcal{D}_{\boldsymbol{\phi}} \quad = \quad [t_0,T] \times [0,w_{max}]. \tag{A.9}$$

The following remark discusses how Assumption A.3 can be relaxed.

Remark A.1. (Relaxing Assumption A.3) As noted above, Assumption A.3 has been introduced for ease of exposition. We therefore briefly describe how each element of element of Assumption A.3 can be relaxed without difficulty.

(i) In the case where the state $\boldsymbol{X}^*(t_m)$ depends on variables in addition to the portfolio wealth, for example historical returns or additional variables (see for example Forsyth (2020); Tsang and Wong (2020)), it is straightforward to incorporate these extra values without materially impacting the key aspects of the convergence analysis. However, it is essential that portfolio wealth is included in $\boldsymbol{X}^*(t_m)$ as per Assumption 4.1.

(ii) The assumption of bounded wealth (A.8) is clearly practical, in that while it is undoubtedly true that the entire wealth of the world is very large, it remains finite. However, from a theoretical perspective, the only reason we introduce (A.8) is to ensure that, given the minimal form of the feature vector (A.7), the controls take inputs in a compact domain (A.9). While boundedness assumptions can be relaxed without theoretical difficulty using straightforward localization arguments (see for example Huré et al. (2021); Tsang and Wong (2020)), this simply introduces yet further notational complexity without providing additional insights into the fundamental arguments underlying the subsequent proofs.

□

In the convergence analysis of Step 2 of the proposed approach, namely the computational estimate of the optimal control obtained using (4.16), we need to introduce some additional assumptions (Assumption A.4 below) since this step involves the training dataset $\mathcal{Y}_n$ of the NN and numerical solution of problem (4.16).

Assumption A.4. *(Convergence analysis: Computational estimate of optimal control) We introduce the following assumptions:*

(i) The training data set $\mathcal{Y}_n = \left\{\boldsymbol{Y}^{(j)} : j \in \{1, ..., n\}\right\}$ *used for training the NN (see (4.13) and associated discussion) is constructed with independent joint asset return paths* $\boldsymbol{Y}^{(j)} \in \mathcal{Y}_n$*. As noted before, this does not assume that the joint asset returns along a given path are independent or serially independent.*

(ii) Number of nodes in each hidden layer $\hbar(n), n \in \mathbb{N}$*: As* $n \to \infty$ $(n \in \mathbb{N})$*, in the case of one hidden layer* $\left(\mathcal{L}^h = 1\right)$*, we assume that* $\hbar(n) = o\left(n^{1/4}\right)$*. For deeper NNs* $\left(\mathcal{L}^h > 1\right)$*, we assume that* $\hbar(n) = o\left(n^{1/6}\right)$*.*

(iii) For each $n \in \mathbb{N}$*, the optimization algorithm used in solving problem (4.16) attains the minimum* $\left(\hat{\boldsymbol{\theta}}_n^*, \hat{\xi}_n^*\right) \in \mathbb{R}^{\nu_n+1}$ *corresponding to a given training data set* $\mathcal{Y}_n$*.*

Since stochastic gradient descent (SGD) is used in training the NN, Assumption A.4(iii) is very strong; however, it is a standard assumption in convergence analyses in the literature (see for example Huré et al. (2021); Tsang and Wong (2020)) in order to focus on the key aspects of a proposed approach. For detailed treatments of theoretical aspects regarding optimization errors (i.e. the differences between the attained values and the true minima) arising when training NNs, the reader is referred to for example Beck et al. (2022); Jentzen et al. (2021a). Note that Assumption A.4(ii), which can also be found in Tsang and Wong (2020), is used to establish a version of the law of large numbers that is applicable to our setting.

Remark A.2 (Increase in number of training samples as $\hbar$ increases)**.** Informally, Assumption A.4(ii) requires that the number of training samples n grows faster than $O(\hbar^4)$ for $\mathcal{L}^h = 1$ and $O(\hbar^6)$ for $\mathcal{L}^h > 1$, where $\hbar$ is the number of nodes in each hidden layer. Since we require a large $\hbar$ (number of nodes in each layer) for good function approximation, this would suggest that convergence in terms of both function approximation and sampling error requires a very large number of sample paths. This would appear to result in a barrier to obtaining accurate results, for practical numbers of samples. However, our numerical examples seem to produce solutions with reasonable errors, hence the requirements of Assumption A.4(ii) are probably not sharp. Regardless, we can certainly expect that the number of samples should be significantly increased as we increase $\hbar$.

A.3: Proof of Theorem 5.1

Before presenting the proof of Theorem 5.1, we first prove some auxiliary results that are preliminary requirements for the proof.

We start with Lemma A.5, which combines and applies selected universal approximation results to our setting. The use of the notation $\boldsymbol{f}^*_n(\cdot,\boldsymbol{\theta}^*_n) \in \mathcal{N}_n$ in Lemma A.5, which has been defined in Subsection 4.1 as the NN using the optimal parameter vector consistent with problem (4.11)-(4.12), will be become clear in the subsequent results.

Lemma A.5. *(Convergence to optimal control) Suppose that Assumption A.2 and Assumption A.3 hold. As per (4.2), let* $\boldsymbol{p}^* = (p^*_i : i = 1, ..., N_a) \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})$ *denote the optimal control associated with problem (4.6). Then there exists a sequence of neural networks,* $\boldsymbol{f}^*_n \in \mathcal{N}_n$, $n \in \mathbb{N}$, *where each* $\boldsymbol{f}^*_n = \left(f^*_{n,i} : i = 1, ..., N_a\right)$ *has parameter vector* $\boldsymbol{\theta}^*_n \in \mathbb{R}^{\nu_n}$*,such that*

$$\lim_{n\to\infty} \sup_{\boldsymbol{\phi}\in\mathcal{D}_{\boldsymbol{\phi}}} \left|f^*_{n,i}(\boldsymbol{\phi};\boldsymbol{\theta}^*_n) - p^*_i(\boldsymbol{\phi})\right| \quad = \quad 0, \qquad \forall i = 1, ..., N_a, \tag{A.10}$$

Proof. For ease of reference, recall that we have defined $\mathcal{N}_n$ as the set of NNs with $\hbar(n)$ hidden nodes in each of the (fixed number of) $\mathcal{L}^h \geq 1$ hidden layers, constructed according to Assumption A.1,

$$\mathcal{N}_n = \{\boldsymbol{f}_n : \mathcal{D}_{\boldsymbol{\phi}} \to \mathcal{Z} |\, \boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n) \text{ satisfies Assumption } A.1 \text{ with } \hbar(n) \text{ nodes in each hidden layer}\}. \tag{A.11}$$

Consider another sequence of NNs, $\overset{\circ}{\boldsymbol{f}}_n, n \in \mathbb{N}$, where each $\overset{\circ}{\boldsymbol{f}}_n : \mathcal{D}_{\boldsymbol{\phi}} \to \mathbb{R}^{N_a}$ is structurally identical to the corresponding $\boldsymbol{f}_n \in \mathcal{N}_n$ in terms of Assumption A.1, *except* that $\overset{\circ}{\boldsymbol{f}}_n$ uses the identity as the (linear) output activation function. Specifically, we assume that $\overset{\circ}{\boldsymbol{f}}_n$ does not apply the activation (A.1) at its output layer, but instead replaces (A.1) with $\overset{\circ}{\mathfrak{a}}{}^{[\mathcal{L}]}_n = \left(\overset{\circ}{\mathfrak{a}}{}^{[\mathcal{L}]}_{n,i} : i = 1, ..., N_a\right) : \mathbb{R}^{N_a} \to \mathbb{R}^{N_a}$ where

$$\overset{\circ}{\mathfrak{a}}{}^{[\mathcal{L}]}_{n,i}\left(\boldsymbol{z}^{[\mathcal{L}]}_n\right) \quad = \quad z^{[\mathcal{L}]}_{n,i} = \sum_{k=1}^{N_a} x^{[\mathcal{L}]}_{n,ik}\mathfrak{a}^{[\mathcal{L}-1]}_{n,k} + b^{[\mathcal{L}]}_{n,i}, \qquad \forall i = 1, ..., N_a. \tag{A.12}$$

For any given $n \in \mathbb{N}$, the relationship between $\boldsymbol{f}_n$ and $\overset{\circ}{\boldsymbol{f}}_n$ are illustrated in Figure A.1. Note that the entire parameter vector $\boldsymbol{\theta}_n$ of $\boldsymbol{f}_n$ is inherited by $\overset{\circ}{\boldsymbol{f}}_n$, since all the weights, biases, and hidden layers and nodes of $\overset{\circ}{\boldsymbol{f}}_n$ and $\boldsymbol{f}_n$ are identical. As a result, we define the set $\overset{\circ}{\mathcal{N}}_n$

$$\overset{\circ}{\mathcal{N}}_n \quad = \quad \left\{ \overset{\circ}{\boldsymbol{f}}_n : \mathcal{D}_{\boldsymbol{\phi}} \to \mathbb{R}^{N_a} \middle| \overset{\circ}{\boldsymbol{f}}_n(\cdot;\boldsymbol{\theta}_n) \text{ satisfies Assumption } A.1, \text{ except} \right.$$
$$\text{output activation (A.5) is replaced by (A.12).}\}, \tag{A.13}$$

where we note that the outputs of $\overset{\circ}{\boldsymbol{f}}_n$ take values which are no longer in $\mathcal{Z} \subset \mathbb{R}^{N_a}$, but instead merely in $\mathbb{R}^{N_a}$. The main benefit of working with $\overset{\circ}{\boldsymbol{f}}_n \in \overset{\circ}{\mathcal{N}}_n$ instead of $\boldsymbol{f}_n \in \mathcal{N}_n$, is that the linear output layer (A.12) means that each $\overset{\circ}{\boldsymbol{f}}_n \in \overset{\circ}{\mathcal{N}}_n$ is in the standard form used by most universal approximation theorems for NNs (see for example Funahashi (1989); Hornik (1991); Hornik et al. (1989); Leshno et al. (1993)).

Recalling for convenience the definition of the softmax function $\boldsymbol{\psi} = (\psi_i : i = 1, .., N_a) : \mathbb{R}^{N_a} \to \mathbb{R}^{N_a}$ in (A.5),

$$\psi_i(\boldsymbol{y}) \quad = \quad \frac{\exp\{y_i\}}{\sum_{i=1}^{N_a} \exp\{y_j\}}, \qquad \forall \boldsymbol{y} = (y_i : i = 1, .., N_a) \in \mathbb{R}^{N_a}, \tag{A.14}$$

we therefore observe that for any $n \in \mathbb{N}$, the NN $\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n) \in \mathcal{N}_n$ can be expressed as a transformation of the corresponding NN $\overset{\circ}{\boldsymbol{f}}_n(\cdot;\boldsymbol{\theta}_n) \in \overset{\circ}{\mathcal{N}}_n$, provided both NNs use the same parameter vector $\boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n}$:

$$\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n) = \boldsymbol{\psi} \circ \overset{\circ}{\boldsymbol{f}}_n(\cdot;\boldsymbol{\theta}_n), \qquad \text{where } \overset{\circ}{\boldsymbol{f}}_n(\cdot;\boldsymbol{\theta}_n) \in \overset{\circ}{\mathcal{N}}_n. \tag{A.15}$$

As per Assumption A.1, recall that $\hbar(n), n \in \mathbb{N}$ satisfies $\hbar(n) < \hbar(n+1), \forall n \in \mathbb{N}$ such that $\lim_{n\to\infty} \hbar(n) = \infty$. Inspired by the notation of Hornik (1991), we define the sets $\mathcal{N}_\infty$ and $\overset{\circ}{\mathcal{N}}_\infty$ as the sets of NNs constructed according to (A.11) and (A.13), respectively, but with an arbitrarily large number of hidden nodes,

$$\mathcal{N}_{\infty} = \bigcup_{n\in\mathbb{N}} \mathcal{N}_n, \qquad \text{and} \qquad \overset{\circ}{\mathcal{N}}_{\infty} = \bigcup_{n\in\mathbb{N}} \overset{\circ}{\mathcal{N}}_n. \tag{A.16}$$

Since $\mathcal{D}_{\boldsymbol{\phi}} \subset \mathbb{R}^{\eta_X+1}$ is compact by (A.9) as per Assumption A.3 (note that this requirement can be relaxed without difficulty as discussed in Remark A.1), we know by the results of Hornik (1991); Hornik et al. (1989) that $\overset{\circ}{\mathcal{N}}_{\infty}$ is uniformly dense in $C\left(\mathcal{D}_{\boldsymbol{\phi}}, \mathbb{R}^{N_a}\right)$. In other words, for any function $\overset{\circ}{\boldsymbol{g}} = \left(\overset{\circ}{g}_i : i = 1, ..., N_a\right) \in C\left(\mathcal{D}_{\boldsymbol{\phi}}, \mathbb{R}^{N_a}\right)$ and any $\epsilon > 0$, there exists a value of $n = n_\epsilon$ sufficiently large such that the corresponding NN $\overset{\circ}{\boldsymbol{f}}_{n_\epsilon} = \left(\overset{\circ}{f}_{n_\epsilon,i} : i = 1, ..., N_a\right) \in \overset{\circ}{\mathcal{N}}_{n_\epsilon}$ such that

$$\sup_{\boldsymbol{\phi}\in\mathcal{D}_{\boldsymbol{\phi}}} \left| \overset{\circ}{f}_{n_\epsilon,i}\left(\boldsymbol{\phi}; \boldsymbol{\theta}_{n(\epsilon)}\right) - \overset{\circ}{g}_i\left(\boldsymbol{\phi}\right) \right| \quad < \quad \epsilon, \qquad \forall i = 1, ..., N_a. \tag{A.17}$$

Note that (A.17) holds for any given number $\mathcal{L}^h \geq 1$ of hidden layers (see for example Corollary 2.7 in Hornik et al. (1989)).

Using the results of Gao and Pavel (2018) , the softmax (A.14) is (Lipschitz) continuous and surjective, since $\psi_i\left(\boldsymbol{y}\right) = \psi_i\left(\boldsymbol{y} + c\right)$ for any $\boldsymbol{y} \in \mathbb{R}^{N_a}$ and $c \in \mathbb{R}$, where $\boldsymbol{y} + c := \left(y_i + c : i = 1, .., N_a\right)$. In addition, it has a continuous right-inverse; as an example, we can simply consider the function $\overleftarrow{\boldsymbol{\psi}}\left(\boldsymbol{z}\right) = \left(\log\left(z_i\right) : i = 1, ..., N_a\right)$ where each $z_i \in (0, 1)$ and that $\sum_i z_i = 1$. Furthermore, by Assumption A.1, no activation function is applied at the input layer (i.e. the "input activation" is trivially injective and continuous). Using these properties of the input and output layers of any $\boldsymbol{f}_n \in \mathcal{N}_n$ together with the results (A.15) and (A.17), we can conclude by the results of Kratsios and Bilokopytov (2020) that the set $\mathcal{N}_\infty$ is uniformly dense in $C\left(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z}\right)$.

Applying this result specifically to the optimal control $\boldsymbol{p}^* \in C\left(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z}\right)$ as per Assumption 4.1, we can conclude that, for any $\epsilon > 0$, there exists a value $n = n_\epsilon$ sufficiently large such that the corresponding NN $\boldsymbol{f}^*_{n_\epsilon}\left(\cdot; \boldsymbol{\theta}^*_{n_\epsilon}\right) \in \mathcal{N}_{n_\epsilon}$ satisfies

$$\sup_{\boldsymbol{\phi}\in\mathcal{D}_{\boldsymbol{\phi}}} \left| f^*_{n_\epsilon,i}\left(\boldsymbol{\phi}; \boldsymbol{\theta}^*_{n_\epsilon}\right) - p^*_i\left(\boldsymbol{\phi}\right) \right| \quad < \quad \epsilon, \qquad \forall i = 1, ..., N_a. \tag{A.18}$$

Note that the exact output of the NN $\boldsymbol{f}^*_{n_\epsilon} \in \mathcal{N}_{n_\epsilon}$, which we recall has $\hbar\left(n_\epsilon\right)$ hidden nodes in each hidden layer, can be attained by a NN with $\hbar\left(n_\epsilon + k\right)$, $k \in \mathbb{N}$ hidden nodes, since we can always set the weights and biases corresponding to the additional $\hbar\left(n_\epsilon + k\right) - \hbar\left(n_\epsilon\right)$ nodes identically to zero. In other words, (A.18) implies the existence of a sequence of NNs $\boldsymbol{f}^*_n\left(\cdot; \boldsymbol{\theta}^*_n\right), n \in \mathbb{N}$, where each $\boldsymbol{f}^*_n\left(\cdot; \boldsymbol{\theta}^*_n\right) \in \mathcal{N}_n$, such that for any $\epsilon > 0$ and sufficiently large $n_\epsilon \in \mathbb{N}$, we have

$$\sup_{\boldsymbol{\phi}\in\mathcal{D}_{\boldsymbol{\phi}}} \left| f^*_{n,i}\left(\boldsymbol{\phi}; \boldsymbol{\theta}^*_n\right) - p^*_i\left(\boldsymbol{\phi}\right) \right| \quad < \quad \epsilon, \qquad \forall n \geq n_\epsilon, \; i = 1, ..., N_a, \tag{A.19}$$

completing the proof of (A.10). □

If Assumption 4.2 and Assumption A.3 are applicable, the wealth dynamics (4.5) using the optimal control is given by

$$W^*\left(t^-_{m+1}; \boldsymbol{p}^*, \boldsymbol{Y}\right) \quad = \quad W^*\left(t^+_m; \boldsymbol{p}^*, \boldsymbol{Y}\right) \cdot \sum_{i=1}^{N_a} p^*_i\left(t_m, W^*\left(t^+_m; \boldsymbol{p}^*, \boldsymbol{Y}\right)\right) \cdot Y_i\left(t_{m+1}\right), \quad t_m \in \mathcal{T}, \tag{A.20}$$

where we recall that $W^*\left(t^+_m; \boldsymbol{p}^*, \boldsymbol{Y}\right) = W^*\left(t^-_m; \boldsymbol{p}^*, \boldsymbol{Y}\right) + q\left(t_m\right)$, $\boldsymbol{X}^*\left(t_m\right) = W^*\left(t^+_m; \boldsymbol{p}^*, \boldsymbol{Y}\right)$ and $W^*\left(t^-_{N_{rb}}\right) := W^*\left(T\right)$. Furthermore, associated with every NN in the sequence $\boldsymbol{f}^*_n\left(\cdot, \boldsymbol{\theta}^*_n\right) \in \mathcal{N}_n$ identified in Lemma A.5, we have the corresponding wealth dynamics as per (4.10) that satisfies

$$W^*\left(t^-_{m+1}; \boldsymbol{\theta}^*_n, \boldsymbol{Y}\right) = W^*\left(t^+_m; \boldsymbol{\theta}^*_n, \boldsymbol{Y}\right) \cdot \sum_{i=1}^{N_a} f^*_{n,i}\left(t_m, W^*\left(t^+_m; \boldsymbol{\theta}^*_n, \boldsymbol{Y}\right); \boldsymbol{\theta}^*_n\right) \cdot Y_i\left(t_{m+1}\right), \quad t_m \in \mathcal{T}, n \in \mathbb{N}. \tag{A.21}$$

The following lemma justifies the use of the notation W^* in the wealth dynamics (A.21).

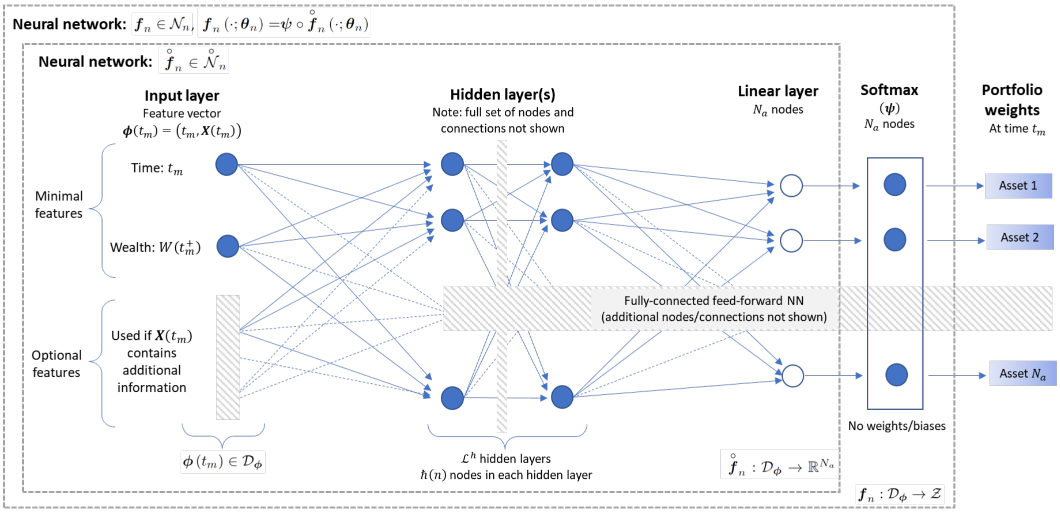

Neural network: $\boldsymbol{f}_n \in \mathcal{N}_n$, $\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n) = \boldsymbol{\psi} \circ \overset{\circ}{\boldsymbol{f}}_n(\cdot;\boldsymbol{\theta}_n)$
Neural network: $\overset{\circ}{\boldsymbol{f}}_n \in \overset{\circ}{\mathcal{N}}_n$
Input layer
Feature vector
$\boldsymbol{\phi}(t_m) = (t_m, \boldsymbol{X}(t_m))$
Minimal features
Time: t_m
Wealth: $W(t_m^+)$
Optional features
Used if $\boldsymbol{X}(t_m)$ contains additional information
$\boldsymbol{\phi}(t_m) \in \mathcal{D}_{\phi}$
Hidden layer(s)
Note: full set of nodes and connections not shown
$\mathcal{L}^h$ hidden layers
$\hbar(n)$ nodes in each hidden layer
Linear layer
N_a nodes
Softmax
$(\boldsymbol{\psi})$
N_a nodes
No weights/biases
Portfolio weights
At time t_m
Asset 1
Asset 2
Asset N_a
Fully-connected feed-forward NN
(additional nodes/connections not shown)
$\overset{\circ}{\boldsymbol{f}}_n : \mathcal{D}_{\phi} \to \mathbb{R}^{N_a}$
$\boldsymbol{f}_n : \mathcal{D}_{\phi} \to \mathcal{Z}$

Figure A.1: Illustration of the interpretation of the NN $\boldsymbol{f}_n(\cdot;\boldsymbol{\theta}_n)$ as a composition of the softmax $\boldsymbol{\psi}$ and the NN $\overset{\circ}{\boldsymbol{f}}_n(\cdot;\boldsymbol{\theta}_n)$ as per equation (A.15). Note that a softmax output layer is used to reflect the given constraints of no short-selling and no leverage, but different admissible control set formulations can be handled without difficulty by modifying the output activation.

Lemma A.6. *(Convergence to optimal wealth) Suppose that Assumption A.2 and Assumption A.3 hold. Let $\boldsymbol{f}_n^*(\cdot,\boldsymbol{\theta}_n^*) \in \mathcal{N}_n$ be the sequence identified in Lemma A.5 such that (A.10) holds. Then the wealth dynamics $W^*(t;\boldsymbol{\theta}_n^*,\boldsymbol{Y})$ associated with each $\boldsymbol{f}_n^*$, obtained as per (A.21), converges to the true optimal wealth dynamics $W^*(t;\boldsymbol{p}^*,\boldsymbol{Y})$ as $n\to\infty$ almost surely. In more detail, we have*

$$\lim_{n\to\infty} W^*\left(t_m^-;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right) = W^*\left(t_m^-;\boldsymbol{p}^*,\boldsymbol{Y}\right) \quad a.s., \qquad \forall t_m \in \mathcal{T}, \tag{A.22}$$

and

$$\lim_{n\to\infty} W^*\left(T;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right) = W^*\left(T;\boldsymbol{p}^*,\boldsymbol{Y}\right) \quad a.s. \tag{A.23}$$

Proof. Note that (A.23) is stated separately since the terminal time T is not a rebalancing time (see (3.1)) and the terminal wealth is critical in the evaluation of the objective functional.

At the start of the time horizon $[t_0,T]$, we are given the initial wealth $W(t_0^-) = w_0 > 0$. Therefore, at the first rebalancing time $t_0 \in \mathcal{T}$, the wealth available for investment does not depend on the control, so that

$$w_0^+ \coloneqq w_0 + q(t_0) = W^*\left(t_0^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right) = W^*\left(t_0^+;\boldsymbol{p}^*,\boldsymbol{Y}\right), \qquad \forall n \in \mathbb{N}. \tag{A.24}$$

Using dynamics (A.20) and (A.21) to compare the wealth at time $t_0 + \Delta t = t_1 \in \mathcal{T}$, we have

$$\begin{aligned} \lim_{n\to\infty} W^*\left(t_1^-;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right) - W^*\left(t_1^-;\boldsymbol{p}^*,\boldsymbol{Y}\right) &= w_0^+ \cdot \sum_{i=1}^{N_a} \left[\lim_{n\to\infty} f_{n,i}^*\left(t_0,w_0^+;\boldsymbol{\theta}_n^*\right) - p_i^*\left(t_0,w_0^+\right)\right] \cdot Y_i(t_1) \\ &= 0 \quad \text{a.s.}, \end{aligned} \tag{A.25}$$

which follows from Lemma A.5 and the fact that $Y_i(t_1) < \infty$ a.s. by assumption (see definition (3.6)).

For purposes of induction, assume that at some $t_m \in \mathcal{T}$, we have

$$\lim_{n\to\infty} W^*\left(t_m^-;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right) = W^*\left(t_m^-;\boldsymbol{p}^*,\boldsymbol{Y}\right) \qquad \text{a.s.} \tag{A.26}$$

If (A.26) holds, then we have

$$\lim_{n\to\infty} W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right) = \lim_{n\to\infty}\left[W^*\left(t_m^-;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right) + q(t_m)\right] = W^*\left(t_m^+;\boldsymbol{p}^*,\boldsymbol{Y}\right) \qquad \text{a.s.}, \tag{A.27}$$

as well as

$$
\begin{aligned}
& \lim_{n\to\infty} \left| f^*_{n,i}\left(t_m, W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right);\boldsymbol{\theta}_n^*\right) - p_i^*\left(t_m, W^*\left(t_m^+;\boldsymbol{p}^*,\boldsymbol{Y}\right)\right)\right| \\
\leq\ & \lim_{n\to\infty} \left| f^*_{n,i}\left(t_m, W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right);\boldsymbol{\theta}_n^*\right) - p_i^*\left(t_m, W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right)\right)\right| \\
& + \lim_{n\to\infty} \left| p_i^*\left(t_m, W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right)\right) - p_i^*\left(t_m, W^*\left(t_m^+;\boldsymbol{p}^*,\boldsymbol{Y}\right)\right)\right| \\
\leq\ & \lim_{n\to\infty} \sup_{\boldsymbol{\phi}\in\mathcal{D}_{\boldsymbol{\phi}}} \left| f^*_{n,i}\left(\boldsymbol{\phi};\boldsymbol{\theta}_n^*\right) - p_i^*\left(\boldsymbol{\phi}\right)\right| \\
=\ & 0 \qquad \text{a.s.,} \qquad \forall i = 1,...,N_a, \qquad \text{(A.28)}
\end{aligned}
$$

which is a consequence of Lemma A.5 and the continuity of $\boldsymbol{p}^* \in C\left(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z}\right)$. From (A.27) and (A.28), we therefore conclude that

$$
\begin{aligned}
& \lim_{n\to\infty} \left| W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right)\cdot f^*_{n,i}\left(t_m, W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right);\boldsymbol{\theta}_n^*\right) - W^*\left(t_m^+;\boldsymbol{p}^*,\boldsymbol{Y}\right)\cdot p_i^*\left(t_m, W^*\left(t_m^+;\boldsymbol{p}^*,\boldsymbol{Y}\right)\right)\right| \\
=& 0 \qquad \text{a.s.,} \qquad \forall i = 1,...,N_a. \qquad \text{(A.29)}
\end{aligned}
$$

Using dynamics (A.20) and (A.21) to compare the wealth at time $t_m + \Delta t = t_{m+1}$, we have

$$
\begin{aligned}
\lim_{n\to\infty} W^*\left(t_{m+1}^-;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right) - W^*\left(t_{m+1}^-;\boldsymbol{p}^*,\boldsymbol{Y}\right) &= \lim_{n\to\infty} W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right)\cdot\sum_{i=1}^{N_a} f^*_{n,i}\left(t_m, W^*\left(t_m^+;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right);\boldsymbol{\theta}_n^*\right)\cdot Y_i\left(t_{m+1}\right) \\
& \quad -W^*\left(t_m^+;\boldsymbol{p}^*,\boldsymbol{Y}\right)\cdot\sum_{i=1}^{N_a} p_i^*\left(t_m, W^*\left(t_m^+;\boldsymbol{p}^*,\boldsymbol{Y}\right)\right)\cdot Y_i\left(t_{m+1}\right) \\
&= 0 \qquad \text{a.s.,} \qquad \text{(A.30)}
\end{aligned}
$$

which follows from (A.29) and $Y_i\left(t_m\right) < \infty$ a.s. By induction, we therefore conclude that (A.22) holds if $t_{m+1} \in \mathcal{T}$ (i.e. if $m < N_{rb} - 1$), and (A.23) holds in the case where $t_{m+1} = t_{N_{rb}} = T$ (i.e. $m = N_{rb} - 1$). □

The following lemma establishes the convergence of the sequence of objective functionals using the NN approximations identified in Lemma A.5.

Lemma A.7. *(Convergence of objective functionals) Suppose that Assumption A.2 and Assumption A.3 hold. Let $\boldsymbol{f}_n^*\left(\cdot,\boldsymbol{\theta}_n^*\right) \in \mathcal{N}_n$ be the sequence identified in Lemma A.5 such that (A.10) holds. Then*

$$
\lim_{n\to\infty} J_n\left(\boldsymbol{\theta}_n^*,\xi;t_0,w_0\right) = J\left(\boldsymbol{p}^*,\xi;t_0,w_0\right), \qquad \forall\xi\in\mathbb{R}, \qquad \text{(A.31)}
$$

where J_n is defined in (4.9), and J is defined in (4.4).

Proof. Let $\xi \in \mathbb{R}$ be arbitrary. By Lemma A.6 and the continuity of F, we have

$$
\lim_{n\to\infty} F\left(W^*\left(T;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right),\xi\right) = F\left(W^*\left(T;\boldsymbol{p}^*,\boldsymbol{Y}\right),\xi\right) \qquad \text{a.s.} \qquad \text{(A.32)}
$$

Therefore, by using the boundedness of wealth as per Assumption A.3, the dominated convergence theorem gives

$$
\lim_{n\to\infty} E^{t_0,w_0}\left[F\left(W^*\left(T;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right),\xi\right)\right] = E^{t_0,w_0}\left[F\left(W^*\left(T;\boldsymbol{p}^*,\boldsymbol{Y}\right),\xi\right)\right]. \qquad \text{(A.33)}
$$

Similarly, by the continuity of G, Lemma A.6, the boundedness of wealth and the dominated convergence theorem, we have

$$
\begin{aligned}
& \lim_{n\to\infty} E^{t_0,w_0}\left[G\left(W^*\left(T;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right),E^{t_0,w_0}\left[W^*\left(T;\boldsymbol{\theta}_n^*,\boldsymbol{Y}\right)\right],w_0,\xi\right)\right] \\
=\ & E^{t_0,w_0}\left[G\left(W^*\left(T;\boldsymbol{p}^*,\boldsymbol{Y}\right),E^{t_0,w_0}\left[W^*\left(T;\boldsymbol{p}^*,\boldsymbol{Y}\right)\right],w_0,\xi\right)\right]. \qquad \text{(A.34)}
\end{aligned}
$$

Finally, using the definitions of J in (4.4) and J_n in (4.9), we combine (A.33) and (A.34) to conclude (A.31). □

Proof of Theorem 5.1

Using the preceding results, we are finally in the position to prove Theorem 5.1. Note that this proof also motivates the use of the notation $\boldsymbol{f}_n^* (\cdot, \boldsymbol{\theta}_n^*)$ and its associated wealth $W^* (T; \boldsymbol{\theta}_n^*, \boldsymbol{Y})$ for the sequence of NNs identified in Lemma A.5 and subsequently used in Lemmas A.6 and A.7 above.

Since $\xi \to F(w, \xi)$ and $\xi \to G(w, x, w_0, \xi)$ are convex by Assumption A.2, and the convexity is preserved by taking the expectation of F, we have the result that $\xi \to J_n(\boldsymbol{\theta}_n, \xi; t_0, w_0)$ and $\xi \to J(\boldsymbol{p}, \xi; t_0, w_0)$ are also convex, so that the infimum over $\xi \in \mathbb{R}$ in each case can be attained and is unique. With $\boldsymbol{p}^*$ still denoting the optimal control, define ξ^* as the value

$$\xi^* \quad := \quad \inf_{\xi \in \mathbb{R}} J(\boldsymbol{p}^*, \xi; t_0, w_0). \tag{A.35}$$

Since $\mathcal{N}_n \subset C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})$, we have, for all $\xi \in \mathbb{R}$ and all $n \in \mathbb{N}$,

$$\inf_{\boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n}} J_n(\boldsymbol{\theta}_n, \xi; t_0, w_0) = \inf_{\boldsymbol{f}_n(\cdot; \boldsymbol{\theta}_n) \in \mathcal{N}_n} J(\boldsymbol{f}_n, \xi; t_0, w_0) \quad \geq \quad \inf_{\boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})} J(\boldsymbol{p}, \xi; t_0, w_0). \tag{A.36}$$

Taking the infimum in (A.36) over $\xi \in \mathbb{R}$, and exchanging the order of minimization, we therefore have

$$\begin{aligned} \inf_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} J_n(\boldsymbol{\theta}_n, \xi; t_0, w_0) \quad &= \quad \inf_{\boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n}} \inf_{\xi \in \mathbb{R}} J_n(\boldsymbol{\theta}_n, \xi; t_0, w_0) \\ &\geq \quad \inf_{\boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})} \inf_{\xi \in \mathbb{R}} J(\boldsymbol{p}, \xi; t_0, w_0), \qquad \forall n \in \mathbb{N}. \end{aligned} \tag{A.37}$$

Taking limits in (A.37), we obtain

$$\lim_{n \to \infty} \inf_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} J_n(\boldsymbol{\theta}_n, \xi; t_0, w_0) \quad \geq \quad \inf_{\xi \in \mathbb{R}} \inf_{\boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})} J(\boldsymbol{p}, \xi; t_0, w_0). \tag{A.38}$$

Now consider specifically the sequence $\boldsymbol{f}_n^* (\cdot, \boldsymbol{\theta}_n^*)$ identified in Lemma A.5 and the value ξ^* in (A.35). Since $\boldsymbol{f}_n^* (\cdot, \boldsymbol{\theta}_n^*) \in \mathcal{N}_n$ (so that $\boldsymbol{\theta}_n^* \in \mathbb{R}^{\nu_n}$) and $\xi^* \in \mathbb{R}$, we have

$$\inf_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} J_n(\boldsymbol{\theta}_n, \xi; t_0, w_0) \quad \leq \quad J_n(\boldsymbol{\theta}_n^*, \xi^*; t_0, w_0), \qquad \forall n \in \mathbb{N}. \tag{A.39}$$

By Lemma A.7, we have

$$\lim_{n \to \infty} J_n(\boldsymbol{\theta}_n^*, \xi^*; t_0, w_0) \quad = \quad J(\boldsymbol{p}^*, \xi^*; t_0, w_0), \qquad \text{where } \xi^* \text{ is given by (A.35).} \tag{A.40}$$

Therefore, taking limits in (A.39) and using (A.40), we obtain the inequality

$$\begin{aligned} \lim_{n \to \infty} \inf_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} J_n(\boldsymbol{\theta}_n, \xi; t_0, w_0) \quad &\leq \quad \lim_{n \to \infty} J_n(\boldsymbol{\theta}_n^*, \xi^*; t_0, w_0) \\ &= \quad J(\boldsymbol{p}^*, \xi^*; t_0, w_0) \\ &= \quad \inf_{\xi \in \mathbb{R}} \inf_{\boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})} J(\boldsymbol{p}, \xi; t_0, w_0). \end{aligned} \tag{A.41}$$

Combining (A.38) and (A.41), we therefore have equality in both (A.38) and (A.41), and obtain

$$\lim_{n \to \infty} \inf_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} J_n(\boldsymbol{\theta}_n, \xi; t_0, w_0) \quad = \quad \inf_{\xi \in \mathbb{R}} \inf_{\boldsymbol{p} \in C(\mathcal{D}_{\boldsymbol{\phi}}, \mathcal{Z})} J(\boldsymbol{p}, \xi; t_0, w_0), \tag{A.42}$$

which concludes the proof of Theorem 5.1. Finally, the notation $\boldsymbol{f}_n^* (\cdot, \boldsymbol{\theta}_n^*)$ in Lemma A.5 is motivated by the fact that equality holds in (A.41). □

A.4: Proof of Theorem 5.2

We start with the following auxiliary result, which is essentially a version of the law of large numbers applicable to the current setting.

Lemma A.8. *(Applicable version of the law of large numbers) Suppose that Assumption A.2, Assumption A.3* *and Assumption A.4 hold. Then*

$$\sup_{\boldsymbol{\theta}_n \in \mathbb{R}^{\nu_n}} \left| \frac{1}{n} \sum_{j=1}^{n} W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right) - E^{t_0,w_0}\left[W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right] \right| \quad \xrightarrow{P} \quad 0, \ \textit{as } n \to \infty, \tag{A.43}$$

and

$$\sup_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} \left| \frac{1}{n} \sum_{j=1}^{n} F\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), \xi\right) - E^{t_0,w_0}\left[F\left(W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right), \xi\right)\right] \right| \quad \xrightarrow{P} \quad 0, \ \textit{as } n \to \infty. \tag{A.44}$$

Proof. Since for any fixed number of hidden layers, our NN formulation also requires $\mathcal{O}\left(\hbar_n\right)$ evaluations of the exponential function, exactly the same steps as in Tsang and Wong (2020) (specifically, see Corollary 7.4 and Theorem 4.3 in Tsang and Wong (2020)) can be used to establish (A.43) and (A.44). □

The following lemma establishes a required auxiliary result involving the function G.

Lemma A.9. *(Convergence of G in probability) Suppose that Assumption A.2, Assumption A.3 and Assumption A.4 hold. Then*

$$\begin{aligned} \sup_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} & \left| \frac{1}{n} \sum_{j=1}^{n} G\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), \frac{1}{n} \sum_{k=1}^{n} W^{(k)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi\right) \right. \\ & \left. - E^{t_0,w_0}\left[G\left(W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right), E^{t_0,w_0}\left[W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi\right)\right] \right| \quad \xrightarrow{P} \quad 0, \end{aligned} \tag{A.45}$$

as $n \to \infty$.

Proof. For given values of $\xi \in \mathbb{R}$, $w_0 > 0$ and $w \in \mathbb{R}$, consider the function $x \to G\left(x, w, w_0, \xi\right)$. By the results of Lemma A.8, we have

$$\sup_{(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n + 1}} \left| \frac{1}{n} \sum_{j=1}^{n} G\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w, w_0, \xi\right) - E^{t_0,w_0}\left[G\left(W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right), w, w_0, \xi\right)\right] \right| \quad \xrightarrow{P} \quad 0, \tag{A.46}$$

as $n \to \infty$. Keeping x fixed, consider the function $w \to G\left(x, w, w_0, \xi\right) : [0, w_{max}] \to \mathbb{R}$. Since G is continuous, there exists a sequence of functions $\left(G_m\right)_{m \in \mathbb{N}}$, where for each $m \in \mathbb{N}$, the function $w \to G_m\left(x, w, w_0, \xi\right) : [0, w_{max}] \to \mathbb{R}$ is L_m-Lipschitz, such that $\left(G_m\right)$ converges uniformly to G on $[0, w_{max}]$ - see for example Miculescu (2000). Therefore, for an arbitrary value of $\epsilon > 0$, there exists a sufficiently large value $\tilde{m} \in \mathbb{N}$ such that

$$\left|G_{\tilde{m}}\left(x, w, w_0, \xi\right) - G\left(x, w, w_0, \xi\right)\right| < \frac{\epsilon}{2}, \qquad \forall w \in [0, w_{max}]. \tag{A.47}$$

Observing that $\frac{1}{n} \sum_{j=1}^{n} W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right) \in [0, w_{max}]$ and by the monotonicity of expectation we also have $E^{t_0,w_0}\left[W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right] \in [0, w_{max}]$, we use (A.47) to obtain

$$\begin{aligned} & \left| G_{\tilde{m}}\left(x, \frac{1}{n} \sum_{j=1}^{n} W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi\right) - G\left(x, \frac{1}{n} \sum_{j=1}^{n} W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi\right) \right| \\ & + \left| G_{\tilde{m}}\left(x, E^{t_0,w_0}\left[W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi\right) - G\left(x, E^{t_0,w_0}\left[W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi\right) \right| \\ < \ & \epsilon, \end{aligned} \tag{A.48}$$

for any given values of $\xi \in \mathbb{R}$ and $w_0 > 0$. In addition, since $G_{\tilde{m}}$ is $L_{\tilde{m}}$-Lipschitz, we have

$$\begin{aligned} & \left| G_{\tilde{m}}\left(x, \frac{1}{n} \sum_{j=1}^{n} W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi\right) - G_{\tilde{m}}\left(x, E^{t_0,w_0}\left[W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi\right) \right| \\ \leq \ & L_{\tilde{m}} \cdot \left| \frac{1}{n} \sum_{j=1}^{n} W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right) - E^{t_0,w_0}\left[W\left(T; \boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right] \right|. \end{aligned} \tag{A.49}$$

Using (A.48) and (A.49) as well as the triangle inequality, we therefore have

$$\begin{aligned}
&\left| G\left(x, \frac{1}{n}\sum_{j=1}^{n} W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi \right) - G\left(x, E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right]\right), w_0, \xi \right| \\
< \quad & \epsilon + L_{\widetilde{m}} \cdot \left| \frac{1}{n}\sum_{j=1}^{n} W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right) - E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right] \right|, \qquad \text{(A.50)}
\end{aligned}$$

for any given values of $\xi \in \mathbb{R}$ and $w_0 > 0$. Taking the supremum over $(\boldsymbol{\theta}_n, \xi) \in \mathbb{R}^{\nu_n+1}$ in (A.50), using the result (A.43) from Lemma A.8 as well as the fact that $\epsilon > 0$ was arbitrary, we therefore have

$$\sup_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \left| G\left(x, \frac{1}{n}\sum_{k=1}^{n} W^{(k)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi \right) - G\left(x, E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi\right) \right| \xrightarrow{P} 0. \quad \text{(A.51)}$$

The results (A.46) and (A.51), together with the triangle inequality, therefore gives

$$\begin{aligned}
&\sup_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \left| \frac{1}{n}\sum_{j=1}^{n} G\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), \frac{1}{n}\sum_{k=1}^{n} W^{(k)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi \right) \right. \\
&\qquad\qquad \left. - E^{t_0,w_0}\left[G\left(W\left(T;\boldsymbol{\theta}_n,\boldsymbol{Y}\right), E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi \right)\right] \right| \\
\leq \quad & \frac{1}{n}\sum_{j=1}^{n} \sup_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \left| \frac{1}{n}\sum_{j=1}^{n} G\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), \frac{1}{n}\sum_{k=1}^{n} W^{(k)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), w_0, \xi \right) \right. \\
&\qquad\qquad \left. - G\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi \right) \right| \\
& + \sup_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \left| \frac{1}{n}\sum_{j=1}^{n} G\left(W^{(j)}\left(T; \boldsymbol{\theta}_n, \mathcal{Y}_n\right), E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi \right) \right. \\
&\qquad\qquad \left. - E^{t_0,w_0}\left[G\left(W\left(T;\boldsymbol{\theta}_n,\boldsymbol{Y}\right), E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n, \boldsymbol{Y}\right)\right], w_0, \xi \right)\right] \right| \\
\xrightarrow{P} \quad & 0 \qquad \text{as } n \to \infty. \qquad \text{(A.52)}
\end{aligned}$$

□

Proof of Theorem 5.2

The expression in (5.2), together with the triangle inequality, imply that

$$\begin{aligned}
&\left| \inf_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \hat{J}_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0, \mathcal{Y}_n\right) - \inf_{\xi\in\mathbb{R}} \inf_{\boldsymbol{p}\in C(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z})} J\left(\boldsymbol{p}, \xi; t_0, w_0\right) \right| \\
\leq \quad & \left| \inf_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \hat{J}_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0, \mathcal{Y}_n\right) - \inf_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\eta_n+1}} J_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0\right) \right| \qquad \text{(A.53)} \\
& + \left| \inf_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} J_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0\right) - \inf_{\xi\in\mathbb{R}} \inf_{\boldsymbol{p}\in C(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z})} J\left(\boldsymbol{p}, \xi; t_0, w_0\right) \right|. \qquad \text{(A.54)}
\end{aligned}$$

Using the definitions of $\hat{J}_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0, \mathcal{Y}_n\right)$ in (4.15) and $J_n\left(\boldsymbol{\theta}_n, \xi; t_0, w_0\right)$ in (4.9), the expression (A.53) gives

$$\begin{aligned}
&\left|\inf_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \hat{J}_n\left(\boldsymbol{\theta}_n,\xi;t_0,w_0,\mathcal{Y}_n\right) - \inf_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\eta_n+1}} J_n\left(\boldsymbol{\theta}_n,\xi;t_0,w_0\right)\right| \\
\leq\ & \sup_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \left|\hat{J}_n\left(\boldsymbol{\theta}_n,\xi;t_0,w_0,\mathcal{Y}_n\right) - J_n\left(\boldsymbol{\theta}_n,\xi;t_0,w_0\right)\right| \\
\leq\ & \sup_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \left|\frac{1}{n}\sum_{j=1}^{n} F\left(W^{(j)}\left(T;\boldsymbol{\theta}_n,\mathcal{Y}_n\right),\xi\right) - E^{t_0,w_0}\left[F\left(W\left(T;\boldsymbol{\theta}_n,\boldsymbol{Y}\right),\xi\right)\right]\right| \qquad \text{(A.55)} \\
&+ \sup_{(\boldsymbol{\theta}_n,\xi)\in\mathbb{R}^{\nu_n+1}} \left|\frac{1}{n}\sum_{j=1}^{n} G\left(W^{(j)}\left(T;\boldsymbol{\theta}_n,\mathcal{Y}_n\right),\frac{1}{n}\sum_{k=1}^{n} W^{(k)}\left(T;\boldsymbol{\theta}_n,\mathcal{Y}_n\right),w_0,\xi\right)\right. \\
&\qquad\left. - E^{t_0,w_0}\left[G\left(W\left(T;\boldsymbol{\theta}_n,\boldsymbol{Y}\right),E^{t_0,w_0}\left[W\left(T;\boldsymbol{\theta}_n,\boldsymbol{Y}\right)\right],w_0,\xi\right)\right]\right|. \qquad \text{(A.56)}
\end{aligned}$$

As per Lemma A.8 and Lemma A.9, (A.55) and (A.56) converge to zero in probability as $n \to \infty$. As a result, since (A.53) therefore converges to zero in probability as $n \to \infty$ and, by Theorem 5.1, (A.54) converges to zero as $n \to \infty$, we conclude that the result (5.2) of Theorem 5.2 holds. □

Appendix B: NN approach: Selected practical considerations

We summarize some practical considerations with respect to the NN approach:

(i) Constructing training and testing datasets $\mathcal{Y}_n$ and $\mathcal{Y}_{\hat{n}}^{test}$: Since $\mathcal{Y}_n$ is used in (4.16) to obtain the optimal NN parameter vector $\hat{\boldsymbol{\theta}}_n^*$, it is usually referred to as the NN "training" dataset (see for example Goodfellow et al. (2016)). Similarly, we can also construct a "testing" dataset $\mathcal{Y}_{\hat{n}}^{test}$, that is of the same structure as (4.13), but typically based on a different implied distribution of $\boldsymbol{Y}$ as a result of different data generation assumptions. For example, $\mathcal{Y}_{\hat{n}}^{test}$ can be obtained using a different time period of historical data for its construction, or different process parameters if there are parametric asset dynamics specified. The resulting approximation $\boldsymbol{f}_n^*\left(\cdot;\hat{\boldsymbol{\theta}}_n^*\right) \in \mathcal{N}_n$ to the optimal control $\boldsymbol{p}^* \in C\left(\mathcal{D}_{\boldsymbol{\phi}},\mathcal{Z}\right)$ obtained using the training dataset in (4.16) can then be implemented on the testing dataset for out-of-sample testing or scenario analysis.

Since $\mathcal{Y}_n$ and $\mathcal{Y}_{\hat{n}}^{test}$ correspond to finite samples of $\boldsymbol{Y}$ and $\boldsymbol{Y}^{test}$, any data generation technique generating paths of underlying asset returns can be used for the construction of training and testing data sets. As illustrated in Section 6, data generation techniques like (i) Monte Carlo simulation of parametric asset dynamics, (ii) block bootstrap resampling of empirical returns, or for example (iii) GAN-generated synthetic returns can all be employed without difficulty, but we emphasize that the approach remains agnostic regarding the underlying data generation methodology. Note that the underlying data generation assumptions typically differ for $\mathcal{Y}_n$ and $\mathcal{Y}_{\hat{n}}^{test}$, respectively, depending on for example the time periods of empirical data considered for in-sample and out-of-sample testing.

As for the number of paths n in each of $\mathcal{Y}_n$ and $\mathcal{Y}_{\hat{n}}^{test}$, experiments show that in the case of measures of tail risk in the objectives such as CVaR (see (3.15)), a significantly larger number of paths are required in order to obtain a sufficiently large sample of tail outcomes in the training and testing data, than for example in cases where variance is the risk measure. To give a concrete examples, at least 2 million paths in the training set of the NN in Subsection 6.2 is required to produce reliable results for the CVaR, whereas 1 million paths in the training set of the NN in Subsection 6.1 are more than sufficient to obtain reliable results.

(ii) Depth (number of hidden layers $\mathcal{L}^h$) and width (number of nodes in each hidden layer $\hbar(n)$) of the NN: As the examples in Section 6 show, remarkably accurate can be obtained with NNs no deeper than 2 hidden layers and a relatively small number of nodes in each hidden layer. For objectives involving more complex investment strategies such as MCV and Mean - Semi-variance (where, even in the case of two assets, the behavior of the optimal strategy is clearly more complex than in the case of for example the MV-optimal strategy), experiments show that two hidden layers lead to stable and reliable results, with the number of hidden nodes in each hidden layer chosen to be slightly more than the number of assets, for example $\hbar(n) = N_a + 2$. For objectives such as DSQ and MV, a single hidden layer is often sufficient.

(iii) Activation functions: As highlighted in Assumption A.1, we use logistic sigmoid activations as a concrete example for convergence analysis purposes, but that these theoretical results can be modified for any of the commonly-used activations (see for example Sonoda and Murata (2017)). Note that since NNs of one or two hidden layers were found to be very effective in solving the problems under consideration, we did not encounter any problems related to vanishing or exploding gradients in the case of logistic sigmoid activations. However, if deeper NNs are required, activation functions could be changed to e.g. ReLU or ELU without affecting the theoretical foundations for the proposed approach.

(iv) For the solution of (4.16) by gradient descent, we used the Gadam algorithm of Granziol et al. (2021). This is simply a combination of the Adam algorithm (Kingma and Ba (2015)) with tail iterate averaging for improved convergence properties and variance reduction (Mucke et al. (2019); Neu and Rosasco (2018); Polyak and Juditsky (1992)). For the Adam algorithm component, the default algorithm parameters of Kingma and Ba (2015) performed well in our setting, typically with no more than 50,000 SGD steps. Note that the mini-batch size selected depends on the problem to be solved: we found that mini-batch sizes of at least 1,000 paths of the training data set $\mathcal{Y}_n$ are required for measures of tail risk in the objective (such as CVaR), since smaller batch sizes typically means that the tail of the returns distribution is not sufficiently well represented in choosing the descent direction, leading to unreliable results in ground truth analyses.

While the technical results of Section 6 formally do not require continuous differentiability (in addition to continuity) of the functions F and G, improved convergence properties of the SGD algorithm can be obtained if the objective is at least continuously differentiable (see for example Shapiro and Wardi (1996)). For implementation purposes, we can therefore smooth objectives like (3.15) in a straightforward way, by for example replacing $\max(x,0)$ in (3.15) with a continuously differentiable approximation used in Alexander et al. (2006),

$$\max(x,0) \quad \simeq \quad \begin{cases} x, & \text{if } x > \lambda_{mcv}, \\ \frac{1}{4\lambda_{mcv}}x^2 + \frac{1}{2}x + \frac{1}{4}\lambda_{mcv}, & \text{if } -\lambda_{mcv} \leq x \leq \lambda_{mcv}\ , \\ 0, & \text{otherwise,} \end{cases} \tag{B.1}$$

where λ_{mcv} is some small smoothing parameter (e.g. $\lambda_{mcv} = 10^{-3}$).

In addition to considering the smoothing of certain objectives, minor modifications to objective functions to avoid (mathematical) ill-posedness may be desirable in certain situations. For example, in the case of the OSQ objective (3.11), the term $\epsilon W(\cdot)$ is added to ensure the problem remains well-posed even if $W(t) \gg \gamma$. In this case, when implementing the numerical solution, small values of ϵ (for example $\epsilon = 10^{-6}$ was chosen in the numerical results of Section 6) do not have a noticeable effect on either the summary statistics or the optimal controls.

Appendix C: Additional parameters for numerical results

In this appendix, additional parameters related to the numerical results of Section 6 are discussed.

The historical returns data for the basic assets such as the T-bills/bonds and the broad market index were obtained from the CRSP [7], whereas factor data for Size and Value (see Fama and French (2015, 1992)) were obtained from Kenneth French's data library[8] (KFDL). The detailed time series sourced for each asset is as follows:

(i) T30 (30-day Treasury bill): CRSP, monthly returns for 30-day Treasury bill.

(ii) B10 (10-year Treasury bond): CRSP, monthly returns for 10-year Treasury bond.[9]

(iii) Market (broad equity market index): CRSP, monthly returns, including dividends and distributions, for a capitalization-weighted index consisting of all domestic stocks trading on major US exchanges (the VWD index).

[7] Calculations were based on data from the Historical Indexes 2020©, Center for Research in Security Prices (CRSP), The University of Chicago Booth School of Business. Wharton Research Data Services 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.

[8] See https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

[9] The 10-year Treasury index was constructed from monthly returns from CRSP back to 1941. The data for 1926-1941 were interpolated from annual returns in Homer and Sylla (2015)

(iv) Size (Portfolio of small stocks): KFDL, “Portfolios Formed on Size”, which consists of monthly returns on a capitalization-weighted index consisting of the firms (listed on major US exchanges) with market value of equity, or market capitalization, at or below the 30th percentile (i.e. smallest 30%) of market capitalization values of NYSE-listed firms.

(v) Value (Portfolio of value stocks): KFDL, “Portfolios Formed on Book-to-Market”, which consists of monthly returns on a capitalization-weighted index of the firms (listed on major US exchanges) consisting of the firms (listed on major US exchanges) with book-to-market value of equity ratios at or above the 70th percentile (i.e. highest 30%) of book-to-market ratios of NYSE-listed firms.

(vi) LowVol (Portfolio of low volatility stocks): KFDL, “Portfolios Formed on Variance”: Monthly returns on a capitalization-weighted portfolio formed at the end of each month, consisting of the firms (listed on major US exchanges) with daily return variance calculated over the preceding 60 days at or below the 20th percentile (i.e. lowest 20%) of the same quantity calculated for NYSE-listed firms.

(vii) Momentum (Portfolio of stocks with high past returns, “winners”): KFDL, “Sorts involving Prior Returns”: Monthly returns on the long portfolio (“winners”) component of the Fama-French version of the long-short momentum factor (see Fama and French (2010, 2012)). This consists of the equal-weighted average of the returns of two capitalization-weighted sub-indices, named “Small High” and “Big High” as per the KFDL data description. For these sub-indices, firms listed on major US exchanges with market capitalization below (resp. above) the monthly median NYSE market capitalization is classified as “small” (resp. “big”). Firms listed on major US exchanges with 2-12 month prior returns above the 70th percentile (i.e. highest 30%) of the NYSE-listed firms’ prior returns are classified as having “high” prior returns. The “Small High” and “Big High” sub-indices are formed at the end of each month as the intersection of the “small” and “big” portfolios, respectively, with the portfolio of firms with high prior returns.

The historical asset returns time series are inflation-adjusted using inflation data from the US Bureau of Labor Statistics[10].

For the purposes of obtaining the parameters for (6.1) in Subsections (6.1) and (6.2), we use the same calibration methodology as outlined in Dang and Forsyth (2016); Forsyth and Vetzal (2017), and assume the jump dynamics of the Kou (2002) model.

In particular, we assume that in the dynamics (6.1), $\log \vartheta_i$ has a asymmetric double-exponential distribution,

$$f_{\vartheta_i}\left(\vartheta_i\right) \quad = \quad \nu_i \zeta_{i,1} \vartheta_i^{-\zeta_{i,1}-1} \mathbb{I}_{\left[\vartheta_i \geq 1\right]}\left(\vartheta_i\right) + \left(1-\nu_i\right) \zeta_{i,2} \vartheta_i^{\zeta_{i,2}-1} \mathbb{I}_{\left[0 \leq \vartheta_i < 1\right]}\left(\vartheta_i\right), \tag{C.1}$$

where $\upsilon_i \in [0,1]$ and $\zeta_{i,1} > 1, \zeta_{i,2} > 0$. In (C.1), ν_i denotes the probability of an upward jump given that a jump occurs. The resulting parameters are obtained using the filtering technique for the calibration of jump diffusion processes - see Dang and Forsyth (2016); Forsyth and Vetzal (2017) for the relevant methodological details. For calibration purposes, a jump threshold equal to 3 has been used in the methodology of Dang and Forsyth (2016).

Table C.1 and Table C.2 summarize the parameters for the asset dynamics for Subsections (6.1) and (6.2), respectively.

Table C.1: Calibrated, inflation-adjusted parameters for asset dynamics in Subsection 6.1: Ground truth - $DSQ\left(\gamma\right)$ with continuous rebalancing. In this example, the first asset is assumed to be a risk-free asset, so we set $\mu_1 = r$, while the second asset follows jump dynamics. The parametric asset returns are (trivially) uncorrelated, and parameters are based on the inflation-adjusted returns of the T30 and VWD time series, respectively, over the period 1926:01 to 2019:12

Parameter	μ_i	σ_i	λ_i	υ_i	$\zeta_{i,1}$	$\zeta_{i,2}$
Asset 1 (T30)	0.0043	-	-	-	-	-
Asset 2(VWD)	0.0877	0.1459	0.3191	0.2333	4.3608	5.504

[10] The annual average CPI-U index, which is based on inflation data for urban consumers, were used - see `http://www.bls.gov.cpi`

Table C.2: Calibrated, inflation-adjusted parameters for asset dynamics in Subsection 6.2: Ground truth - problem $MCV(\rho)$. In this example, there are two assets with jump dynamics (see Forsyth and Vetzal (2022)), with parameters based on the inflation-adjusted returns of the T30 and VWD time series over the period 1926:01 to 2019:12. The Brownian motions in (6.1) have correlation $dZ_1 dZ_2 = \rho_{1,2} dt$.

Parameter	μ_i	σ_i	λ_i	υ_i	$\zeta_{i,1}$	$\zeta_{i,2}$	$\rho_{1,2}$
Asset 1 (T30)	0.0045	0.0130	0.5106	0.3958	65.85	57.75	0.08228
Asset 2(VWD)	0.0877	0.1459	0.3191	0.2333	4.3608	5.504	0.08228