DEEP BELLMAN HEDGING

Abstract

This paper presents a novel reinforcement learning approach for hedging financial portfolios using a deep Bellman framework. Proposed a practical implementation that leverages historical market data to optimize hedging strategies for a diverse range of financial instruments, including options, futures, and swaps. The core of our methodology is encapsulated in the Bellman equation, which facilitates the computation of an optimal value function that maximizes risk-adjusted returns while accounting for trading frictions such as costs and liquidity constraints.

$$V^*\big(\text{current state}\big) \overset{!}{=} \sup_{\text{action } \mathbf{a}} \ : \ U\big[\ \text{discount factor} \cdot V^*\big(\text{future state}(\mathbf{a})\big) + \text{rewards}(\mathbf{a})\ \big]$$

Our approach is characterized by its ability to operate within realistic, continuous state and action spaces without imposing terminal maturity restrictions, allowing for dynamic adaptation to varying market conditions. We demonstrate the existence of finite solutions to our Bellman equation and provide empirical evidence of the model's efficacy through extensive numerical simulations. The trained model offers robust hedging solutions applicable to arbitrary initial portfolios and market states, significantly enhancing the practical utility of reinforcement learning in financial risk management.

1 Introduction

The evolution of hedging strategies in finance has undergone significant changes, particularly with the advent of machine learning techniques. Traditional methods for deep hedging, primarily based on the Black-Scholes framework and other classical quantitative finance models, have faced substantial limitations. These older approaches often rely on static models that do not adequately account for market dynamics and the complexities of real-world trading conditions.

Motivation

Historically, hedging strategies were developed using deterministic models that assumed constant volatility and liquidity, leading to oversimplified risk assessments. Such models typically neglected critical factors like trading frictions, transaction costs, and the non-linear nature of financial instruments. As a result, they often provided inaccurate pricing and risk management signals, compelling traders to rely on their heuristics to adjust these signals for real-world application. This reliance on manual adjustments not only introduced inconsistencies but also limited the scalability and automation of trading strategies.

Moreover, traditional methods often imposed boundary conditions related to terminal maturities, which restricted their applicability across various market states. This rigidity made it challenging to adapt to sudden market changes or shifts in portfolio compositions. Consequently, traders faced increased risks due to the inability of these models to dynamically adjust to evolving market conditions.

Transition to Deep Bellman Hedging

In response to these challenges, the Deep Bellman Hedging framework introduces a more robust and flexible approach to managing portfolios of financial instruments. By leveraging reinforcement learning techniques, this method allows for a dynamic programming solution that learns from historical data without imposing strict boundary conditions. The core of this methodology is encapsulated in the Bellman equation, which facilitates the optimization of hedging strategies across diverse market states and portfolios.
The Deep Bellman Hedging framework addresses the shortcomings of its predecessors by incorporating realistic trading frictions and enabling the use of a variety of derivatives such as forwards, swaps, futures, and options. This approach not only enhances the accuracy of risk assessments but also improves the overall performance of hedging strategies under real-life market dynamics. By eliminating the need for frequent retraining tied to specific market conditions, this method offers a more adaptive solution that can respond effectively to changing financial landscapes.

- Bellman Equation Framework: The core of Deep Bellman Hedging is based on a Bellman equation that defines the optimal value function V* as follows:

$$V^*\big(\text{current state}\big) \overset{!}{=} \sup_{\text{action } \mathbf{a}} : U\big[\text{ discount factor} \cdot V^*\big(\text{future state}(\mathbf{a})\big) + \text{rewards}(\mathbf{a})\ \big]$$

U represents a risk-adjusted return metric, emphasizing the integration of risk preferences into the decision-making process.

- Numerical Implementation: A significant advancement is the introduction of a numerical implementation method that allows arbitrary portfolios of derivatives to be represented as states using historical data. This eliminates the need for complex market simulators, making the approach more practical.
- Mimicking Trader Behavior: The model aims to replicate a trader's historical experience, providing an AI with similar "experience" as a human trader would have had. However, it acknowledges that human oversight remains essential to adjust for sudden market changes or extreme risk scenarios.

- Well-Posed Conditions: The research clarifies the conditions under which the corresponding Bellman equations are well-posed and admit unique finite solutions, contributing to the theoretical foundation necessary for reliable implementation in practice.

In **summary**, this paper aims to present a comprehensive overview of the limitations inherent in traditional deep hedging methods while showcasing how the Deep Bellman Hedging framework provides a superior alternative that aligns with modern financial practices and technological advancements.

2 Deep Bellman Hedging

We formulate our approach essentially as a continuous state Markov Decision Process (MDP) problem. We will make a decision from some point in time to another. That would typically be intraday or from day to day. To simplify our discussion we will assume we are making a decision "today" and then again "tomorrow". Variables which are valid tomorrow will be indicated by a $'$. We will strive to use bold letters for vectors. A product of two vectors is element wise, while "·" represents the dot product. We will use small letters for instances of data, and capital letters for random variables.

Notations and definitions

- Market State: Denoted by m, representing all available information today, including market prices, past prices, bid/ask data, and social media feeds. The set of all market is defined as:

$$\mathcal{M} \subset \mathbb{R}^N$$

- Tomorrow's Market State: Represented by the random variable M', which depends solely on the current market state m and not on trading activities.
- Cash Flows: The cash flows arising from holding an instrument x today are denoted as $r(x,m) \in$ R. For a vector instrument x, the cash flows are represented as r$(x;m)$.

Framework of Operation: Markov Decision Process (MDP)

The Markov Decision Process (MDP) is a mathematical framework used to model decision-making situations where outcomes are influenced by both random factors and the choices made by a decision-maker. In the context of finance, MDPs provide a structured way to analyze trading strategies and portfolio management under uncertainty. Here's a concise breakdown of its components:

- States: These represent the various conditions or situations that can exist in the process. For instance, in a financial context, a state could reflect the current market conditions, including asset prices and volatility.
- Actions: These are the available choices for the decision-maker at each state. A trader might choose to buy, sell, or hold an asset based on the current market situation.
- Transitions: This refers to how the process moves from one state to another based on the action taken. The outcome is often uncertain; for example, buying an asset might lead to different future market conditions.
- Rewards: Each action taken results in some benefit or cost, known as a reward. This metric helps the decision-maker evaluate the effectiveness of their actions.
- Policy: A policy is a strategy that dictates what action to take in each state. The objective is typically to identify the optimal policy that maximizes rewards over time.
- Discount Factor: This factor weighs immediate rewards more heavily than future rewards, reflecting the principle that receiving value now is generally preferable to receiving it later.

MDPs are particularly useful in fields such as finance, robotics, and artificial intelligence, where decision-making involves uncertainty about outcomes.

Setting Up Variables and Foundations for Further Operations

1. Market State *(m):* The current market state includes all relevant information such as prices, time, past prices, bid/ask spreads, and social media feeds. It is represented as $m \in M \subset R^N$.
2. Future Market State *(M'):* The market state for tomorrow is modeled as a random variable M', whose distribution depends solely on today's market state m. *Notation* $m \equiv m^{(t)}$ indicates today's market state, while $M' \equiv M^{(t+1)}$ represents tomorrow's state.
3. Conditional Expectation: The expectation of a function f of M', given the current market state m, is denoted by*:* $E[f(M')\,|\,m] = \int f(m')P[dm'\,|\,m]$ where $P[dm'\,|\,m]$is the conditional probability distribution of M'
4. Financial Instruments *(X):* This space includes various financial instruments like securities and derivatives. Cash flows from holding an instrument $x \in X$ today are denoted by $r(x,m) \in R$, encompassing expiry settlements, coupons, dividends, and payments from borrowing or lending assets.
5. Book Value (B(x,m)): The book value of an instrument x today is its market value represented as B(x,m). If x consists of multiple instruments, B(x,m) reflects their collective book values.

6. Future Value (B(x',M')): The value of an instrument tomorrow is denoted by $B(x',M')$, where x' indicates its value after time has passed.
7. Numeraire and Discounting: A numeraire is used for discounting cash flows over time. The discount factor from tomorrow to today is represented as $\beta(m)$, satisfying $\beta(m) \leq \beta* < 1$, ensuring proper valuation over time.
8. Discounted Profit-and-Loss (P&L): The change in book value or P&L for an instrument $x \in X$ is calculated as:

$$dB(x,m,M') = B(x',M') - B(x,m) + r(x,m)$$

This captures changes in book value due to market evolution and cash flows.

This framework effectively models how financial instruments evolve over time and how their values change while accounting for cash flows and the time value of money. By integrating these concepts into decision-making processes, traders can better navigate uncertainties inherent in financial markets.

Trading

In the realm of financial trading, effective portfolio management is crucial for mitigating risks and maximizing returns. A trader manages a portfolio, often referred to as a "book," which consists of various financial instruments such as currencies, securities, and over-the-counter (OTC) derivatives. The current state of the trader's portfolio can be represented as $s = (z,m)$ where z denotes the portfolio and m represents the prevailing market conditions. The combined state s resides in the space $S = X \times M$, allowing for a comprehensive analysis of both the portfolio and market dynamics.

To manage risks effectively, the trader has access to a set of n liquid hedging instruments, denoted as $h \equiv h(m) \equiv h(s) \in X^n$. These instruments can include forwards, options, swaps, and other derivatives. Importantly, the characteristics of these hedging instruments are not fixed; they vary across different market states in terms of time-to-maturity and strikes relative to at-the-money positions. This variability necessitates a flexible approach to hedging that adapts to changing market conditions.

When executing trades, the action of buying $a \in R^n$ units of hedging instruments incurs transaction costs $c\ (a;z,m)$, which are added to the book value. The transaction cost function is normalized such that $c\ (0;s) = 0$, ensuring that no cost is incurred when no actions are taken.

The function is also non-negative and convex, reflecting real-world trading dynamics where costs typically increase with larger trade sizes. This formulation allows for modeling trading restrictions based on the current portfolio position, such as short-sell constraints or risk exposure limits.

Upon taking an action a, the new portfolio value for the next time step becomes $z' + a \cdot h'$. The trader's decision-making process is governed by a trading policy π, which determines the next action based on the current state: $a = \pi\ (z,m)$. The set of all admissible policies is denoted as

$$P = \{\pi : X \times M \rightarrow A(z,m)\}$$

To evaluate the effectiveness of these actions, traders typically assess changes in book values alongside cash flows from hedging activities. The reward associated with taking an action a at time step t is defined as:

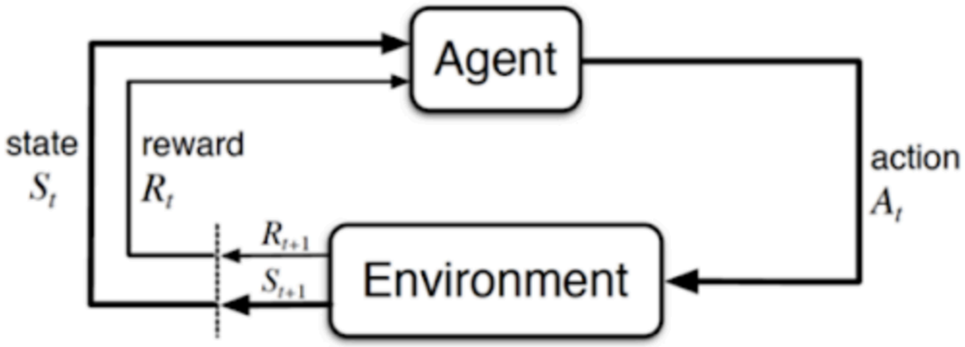
$$R(\mathbf{a}; z, \mathbf{m}, \mathbf{M}') := \underbrace{dB(z, \mathbf{m}, \mathbf{M}')}_{\substack{\text{Portfolio}\\ \text{P\&L}}} + \underbrace{\mathbf{a} \cdot d\mathbf{B}(\mathbf{h}, \mathbf{m}, \mathbf{M}')}_{\substack{\text{New Hedge}\\ \text{P\&L}}} - \underbrace{c(\mathbf{a}; z, \mathbf{m})}_{\substack{\text{Trading}\\ \text{Cost}}} \ .$$

This equation captures the profit and loss (P&L) resulting from changes in both the portfolio and hedging instruments while accounting for transaction costs. For any given policy π, we define its rewards as:

$$R(\pi; z, \mathbf{m}, \mathbf{M}') := R(\pi(z, \mathbf{m});\ z, \mathbf{m}, \mathbf{M}')$$

This framework emphasizes the importance of dynamic decision-making in trading environments where market conditions fluctuate continuously. By integrating transaction costs and varying market states into the reward structure, traders can better align their strategies with real-world trading scenarios.

In conclusion, this structured approach to risk management through dynamic hedging policies provides traders with a comprehensive toolset for navigating complex financial landscapes. By leveraging historical data and modern computational techniques, traders can optimize their portfolios while effectively managing risks associated with diverse financial instruments.



2.1 The Bellman Equation for Monetary Utilities

In the realm of reinforcement learning, particularly in financial applications, the objective is often to maximize the discounted expected future rewards derived from a given policy. The optimal value function $V^*$is typically expressed through a Bellman equation of the form:

$$V^*\big(\,z;\,\mathbf{m}\,\big) \overset{!}{=} \sup_{\mathbf{a}\in\mathcal{A}(z,\mathbf{m})} :\ \mathbb{E}\big[\ \beta(\mathbf{m})\ V^*\big(\,z' + \mathbf{a}\cdot\mathbf{h}';\,\mathbf{M}'\,\big) + R(\mathbf{a};z,\mathbf{m},\mathbf{M}')\,\big|\,\mathbf{m}\ \big]$$

This equation captures the essence of decision-making under uncertainty, where z represents the current portfolio, m denotes market conditions, and a is the action taken. However, in finance, it is often more appropriate to incorporate risk aversion into this framework. This leads us to consider an operator U that reflects a decision-maker's preference for outcomes with lower uncertainty when expected returns are identical.

Definition of the Deep Bellman Hedging Problem

The Deep Bellman Hedging problem seeks to identify a value function V^* that satisfies:

$$\left\{\begin{array}{lcl} V^*\big(\,z;\,\mathbf{m}\,\big) & \overset{!}{=} & (TV^*)(z,\mathbf{m}) \\ (Tf)(z,\mathbf{m}) & := & \sup_{\mathbf{a}\in\mathcal{A}(z,\mathbf{m})} :\ U\big[\ \beta(\mathbf{m})\ f\big(\,z' + \mathbf{a}\cdot\mathbf{h}';\,\mathbf{M}'\,\big) + R(\mathbf{a};z,\mathbf{m},\mathbf{M}')\,\big|\,\mathbf{m}\ \big] \end{array}\right.$$

In this context, the action a is contingent upon the current state $s = (z,m)$. Notably, the value function here represents the "excess value" of a portfolio over its book value. If $V^* = 0$, it implies that the optimal risk-adjusted value of the portfolio aligns with its book value.

Risk-Adjusted Return Metrics

In finance, various reasonable risk-adjusted return metrics U exist, including mean-volatility and mean-variance approaches. While these metrics are prevalent in practice, they exhibit non-monotonicity; thus, they do not guarantee that if $f(s) \geq g(s)$ for all states $s = (z,m)$, then $U[f(S')] \geq U[g(S')]$. This characteristic complicates standard convergence proofs for the Bellman equation.

To address these challenges, we focus on monetary utilities defined by specific properties:

- Normalization: A monetary utility U is normalized such that $U(0) = 0$.
- Monotonicity: It is monotone increasing; more wealth results in higher utility.
- Concavity: Reflecting diversification benefits.
- Cash-Invariance: The utility increases linearly with added cash amounts.

These properties ensure that optimal actions remain independent of current wealth levels and that our utilities exhibit risk aversion concerning probability measure P.

Optimized Certainty Equivalents (OCE)

We further define monetary utilities as optimized certainty equivalents (OCE) derived from a utility function $u : \mathrm{R} \rightarrow \mathrm{R}$, which is continuously differentiable (C1), monotone increasing, and concave. The OCE monetary utility can be expressed as:

$$U\left[\ f(\mathbf{S}')\ \middle|\ \mathbf{s}\ \right] := \sup_{y(\mathbf{s})\in\mathbb{R}} \mathbb{E}\left[\ u\left(\ f(\mathbf{S}') + y(\mathbf{s})\right)\ \middle|\ \mathbf{s}\ \right] - y(\mathbf{s})$$

The function $y(s)$ will be modeled using neural networks to capture complex relationships in financial data.

Examples of OCE Utility Functions

Several OCE utility functions can be employed:

- Expectation (Risk-Neutral): $u(x) := x$
- Worst Case: $u(x) := \inf\ x$
- Conditional Value at Risk (CVaR): $u(x) := (1+\lambda) \min\{0,X\}$
- Entropy: $u(x) := (1 - e^{-\lambda x})/\lambda$

These functions illustrate diverse approaches to quantifying risk and reward in financial contexts.

Coherence and Time-Consistency

A monetary utility is termed coherent if it maintains linearity with respect to position size. However, coherence may not align with intuitive risk perceptions where larger positions typically entail greater risks. Time-consistency ensures that iterative applications yield consistent results; only certain utilities like entropy and expectation maintain this property.

We establish that if rewards are finite—defined as:

$$\sup_{\mathbf{a}\in\mathcal{A}(z,\mathbf{m})} U\big[\, R(\mathbf{a};\mathbf{z},\mathbf{m},\mathbf{M}') \,\big|\, \mathbf{m} \,\big] < \infty \ .$$

then under the assumption that U is an OCE monetary utility, the Bellman equation has a unique finite solution. This result relies on the properties of monotonicity and cash-invariance inherent in our chosen monetary utility.

Using Cash Flows as Rewards

Instead of using full mark-to-market rewards (which include future variables like tomorrow's book value), an alternative approach is to define rewards based only on realized cash flows:

$$\tilde{R}(\mathbf{a},z,\mathbf{m}) := \underbrace{r(z,\mathbf{m})}_{\substack{\textit{Cashflows from}\\ \textit{our portfolio}}} \quad \underbrace{-\mathbf{a}\cdot\mathbf{B}(\mathbf{h},\mathbf{m}) - c(\mathbf{a},z,\mathbf{m})}_{\substack{\textit{Proceeds from}\\ \textit{trading } \mathbf{a}}} \ ,$$

While theoretically equivalent to the original formulation, this approach faces numerical challenges due to infrequent cash flows in many instruments (e.g., options only pay at maturity).

Extension to Multiple Time Steps

The Bellman equation can be extended over multiple time steps by defining cumulative discount factors and rewards:

$$(T_n f)(z,\mathbf{m}) := \sup_{\pi} :\ U\left[\ \beta_n\ f\big(z^{(n)} + \mathbf{A}^{(n)}\cdot\mathbf{H}^{(n)};\, \mathbf{M}^{(n+1)}\big) + \sum_{i=1}^{n} \beta_{i-1}\, R(\mathbf{A}^{(i)}, z^{(i)}, \mathbf{M}^{(i)}, \mathbf{M}^{(i+1)}) \ \big|\ \mathbf{M}^{(1)} = \mathbf{m}\ \right]$$

This allows for dynamic optimization across longer horizons.

Key Result

If rewards are finite and U is a monetary utility (e.g., CVaR or VaR), the Bellman equation has a unique finite solution. This result relies on monotonicity and cash-invariance of U, ensuring robust theoretical foundations for solving the hedging problem.

By incorporating monetary utilities and addressing numerical challenges in reward definitions, it provides a robust framework for optimizing portfolios under uncertainty. The flexibility to handle multiple time steps further enhances its practical relevance in dynamic trading environments.

In summary, this section delineates how integrating risk-adjusted return metrics into the Bellman equation framework enhances decision-making in financial contexts. By focusing on monetary utilities and their properties, we lay the groundwork for developing robust hedging strategies that account for both expected returns and associated risks.

3 Numerical Implementation

In this section, we outline an iterative algorithm designed to approach the optimal solution of our Bellman equation. The goal is to optimize the value function

$$V^{*}(z,m) = 0$$

Initialization

We begin with an initial guess for the value function:

$$V^{(0)}(z,m) := 0 \text{ for all portfolios and states } s = (z,m).$$

This serves as a baseline from which we iteratively refine our estimates.

Actor-Critic Scheme

The algorithm employs an actor-critic framework consisting of two main components:

1. Actor: Given the previous value function $V^{(n-1)}$, we seek to find an optimal neural network policy $\pi^{(n)} : X \times M \rightarrow \mathrm{R}^{n}$ that maximizes:

$$(TV^{(n-1)})(z, \mathbf{m}) := \sup_{\pi} : \; U\Big[\beta(\mathbf{m})\, V^{(n-1)}\big(z' + \pi(z, \mathbf{m}) \cdot \mathbf{h}'; \mathbf{M}'\big) + R(\pi; z, \mathbf{m}, \mathbf{M}') \,\Big|\, \mathbf{m}\Big] \;.$$

2. Here, $c\,(a;z,m) = \infty$ if $a \notin A\,(z,m)$, ensuring that any finite solution $\pi^{(n)}$ belongs to the set of admissible policies P.

3. For our optimized certainty equivalent (OCE) monetary utility, we need to jointly maximize both the policy network $\pi^{(n)}$ and another network $y^{(n)}$:

$$\sup_{\pi,y} : \mathbb{E}\Big[\beta(\mathbf{m})\,u\Big(V^{(n-1)}(z' + \pi(z,\mathbf{m})\cdot\mathbf{h}';\mathbf{M}') + y(z,\mathbf{m})\Big) - y(z,\mathbf{m}) + R(\pi; z,\mathbf{m},\mathbf{M}')\,\Big|\,\mathbf{m}\Big]$$

4. The choice of density Q, which represents the probability distribution over possible portfolio and market states, is critical in this context.
5. Critic (Interpolation): The next step involves estimating a new value function $V^{(n)}$ based on the obtained policy $\pi^{(n)}$ and network $y^{(n)}$. We fit a neural network such that:

$$V^{(n)}(z,\mathbf{m}) \equiv (TV^{(n-1)})(z,\mathbf{m})$$

6. This equation holds true if π is derived from the complete set of measurable functions and if all previous state information is included. The equality is guaranteed for coherent monetary utilities under certain conditions.

Numerical Solution

To solve the equation above, we utilize numerical libraries like TensorFlow or PyTorch. This allows us to sample value and we can then find network weights for $V^{(n)}$ by solving an interpolation problem:

$$\inf_{V} : \mathbb{E}\Big[\; d\Big(-V(Z,\mathbf{M}) + (TV^{(n-1)})(Z,\mathbf{M})\Big)\;\Big]$$

where $d(\cdot) = |\cdot|$. Classic interpolation techniques like kernel interpolators may also be employed instead of neural networks.

This iterative scheme intuitively improves both the estimation of monetary utility $V^{(n)}$ and the optimal action $a^{(n)}$. There is a consideration regarding the number of training epochs required at each step. Previous works suggest that a single training step may suffice.

Remarks on Implementation

- Remark 3: In some scenarios where samples of $TV^{(n-1)}$ are unavailable, trained networks for actions and rewards can be directly utilized. This leads us to solve:

$$\inf_{V} : \mathbb{E}\Big[\Big(-V(Z;\mathbf{M}) + \mathbb{E}\Big[\beta(\mathbf{M})\;u\Big(V^{(n-1)}(\cdots;\mathbf{M}') + y^{(n)}(Z,\mathbf{M})\Big) - y^{(n)}(Z,\mathbf{M}) + R(\pi^{(n)}; Z,\mathbf{M},\mathbf{M}')\,\Big|\,Z,\mathbf{M}\Big]\Big)\Big|$$

Although this nested expectation is numerically sub-optimal, it can be addressed by reformulating it as:

$$\inf_{V} : \mathbb{E}\left[\left(-V(\,Z;\,\mathbf{M}\,) + \beta(\mathbf{M})\;u\Big(V^{(n-1)}(\cdots,\mathbf{M}'\,) + y^{(n)}(Z,\mathbf{M})\Big) - y^{(n)}(Z,\mathbf{M}) + R(\pi^{(n)};Z,\mathbf{M},\mathbf{M}')\right)^{2}\right) \;.$$

which maintains the same gradient in V, ensuring that we achieve the same optimal solution.

3.1 Representing Portfolios

In applying the approach outlined in Section 3, a significant challenge arises in the need to represent portfolios in a numerically efficient manner. This section builds and proposes an enhanced representation of trader instruments, moving away from more cumbersome methods to a more streamlined approach.

Historical Market Data and Instrument Features

Assuming we have historical market data m^t at discrete time points $\tau^0,\ldots,\tau^N$, we consider that at each time τ^j, the portfolio contains instruments $x_t=(x^{t,1},\ldots,x^{t,m_t})$ where each $x^{t,i}\in X$. For each instrument $x^{t,i}$, we maintain a vector of historic risk metrics $f^{t,it}\in \mathrm{R}^F$ computed at time t. These metrics may include the book value, various Greeks (sensitivity measures), and other relevant calculations that assist traders in their risk management decisions.

- Predictive Power: It is assumed that these metrics possess significant predictive power regarding the behavior of instruments, as they are commonly used by human traders to guide their decisions.

Finite Markov Representation (FMR)

To efficiently represent our instruments, we utilize a Finite Markov Representation (FMR). This involves:

- Linear Features: We consider only linear features such that for any weight vector $w\in\mathrm{R}^{m_t}$, the feature vector of the weighted instrument $w\cdot x_t$ can be directly derived from $w\cdot f^t$. This eliminates the need for recomputation of features later.
- Aggregated Cash Flows: We denote by r_{it} the historic aggregated cash flows of instrument $x_{t,i}$ over the period $[\tau^t,\tau^{t+1}]$. The aggregated cash flows for all instruments at time t are represented as $r^t=(r^{1t},\ldots,r^{m_t t})$.

- Book Values: The book values of instruments at times $u = t,t+1$ are denoted by $B^{u} =(B_{u,1},\ldots,B_{u,m_t})$.

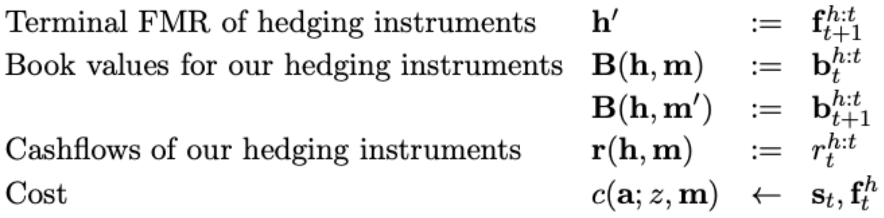
Hedging Instruments and Market Features

For hedging instruments, we assume access to their respective feature vectors $F_{h:t}$ at both time points t and $t+1$. It is crucial to note that the Greeks for hedging instruments at time $t+1$ differ from those computed at time $t+1$, reflecting changes in instrument mechanics.Additionally, we select a reasonable subset of market features at each time step τ_t, denoted by m. Although not all available state vectors will be used in practice, this selection is essential for effective modeling.

Random Scenario Generation

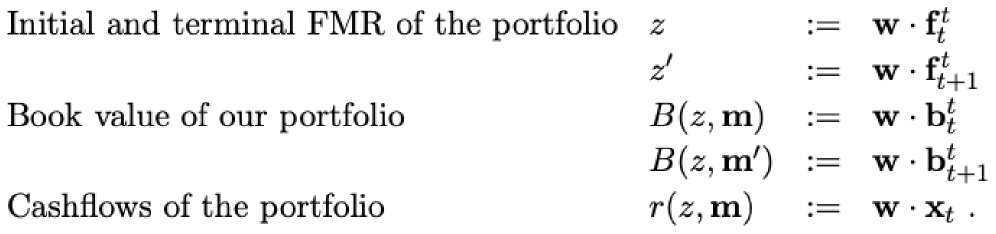
To facilitate training without relying on complex market simulations, we generate random scenarios through the following steps:

1. Random Time Selection: Choose a random time point $t \in \{0,...,N-1\}$ to determine market states $m=m_t$ and future states $m'=m_{t+1}$.
2. Identify Hedging Instruments: Associate hedging instruments with their finite Markov representation:



Terminal FMR of hedging instruments	$\mathbf{h}'$	$:=$	$\mathbf{f}^{h:t}_{t+1}$
Book values for our hedging instruments	$\mathbf{B}(\mathbf{h},\mathbf{m})$	$:=$	$\mathbf{b}^{h:t}_{t}$
	$\mathbf{B}(\mathbf{h},\mathbf{m}')$	$:=$	$\mathbf{b}^{h:t}_{t+1}$
Cashflows of our hedging instruments	$\mathbf{r}(\mathbf{h},\mathbf{m})$	$:=$	$r^{h:t}_{t}$
Cost	$c(\mathbf{a}; z,\mathbf{m})$	$\leftarrow$	$\mathbf{s}_t, \mathbf{f}^{h}_{t}$

3. Weight Vector Selection: Choose a random weight vector $w \in \mathrm{R}_{m_t}$ to define a sample portfolio as:



Initial and terminal FMR of the portfolio	z	$:=$	$\mathbf{w} \cdot \mathbf{f}^{t}_{t}$
	z'	$:=$	$\mathbf{w} \cdot \mathbf{f}^{t}_{t+1}$
Book value of our portfolio	$B(z,\mathbf{m})$	$:=$	$\mathbf{w} \cdot \mathbf{b}^{t}_{t}$
	$B(z,\mathbf{m}')$	$:=$	$\mathbf{w} \cdot \mathbf{b}^{t}_{t+1}$
Cashflows of the portfolio	$r(z,\mathbf{m})$	$:=$	$\mathbf{w} \cdot \mathbf{x}_t$.

The construction of a reasonable randomization for the weight vector is critical. If samples deviate too much from likely portfolios, model performance may suffer. Conversely, relying solely on historical portfolios limits the model's ability to adapt to variations. Generating diverse portfolios enhances sample size and improves learning.

This approach allows for training our actor-critic model using real data scenarios without necessitating a complex market simulator. Given the challenges associated with simulating book values and Greeks for numerous hypothetical derivative instruments, this method offers a practical solution. Future research may explore the feasibility of developing a simulator capable of generating synthetic market data alongside corresponding feature vectors for financial instruments.

Existence of a Unique Finite Solution for Deep Bellman Hedging

1. Measure Space: The space $S = Z \times M$ is considered, where Z represents future cash flows parameterized in $\mathrm{R}^{|Z|}$ and M denotes the market state. The future cash flows are modeled as suitably integrable adapted stochastic processes.
2. Function Space: The function space F consists of equivalence classes of functions f : $S \to \mathrm{R}$.
3. Bellman Operator: The operator T is defined as:

$$(Tf)(z, \mathbf{m}) := \sup_{\mathbf{a} \in \mathcal{A}(z, \mathbf{m})} : U\left[\beta(\mathbf{m}) f\left(z' + \mathbf{a} \cdot \mathbf{h}', \mathbf{M}'\right) + R(\mathbf{a}, z, \mathbf{m}, \mathbf{M}') \,\middle|\, \mathbf{m}\right]$$

This operator is central to the formulation of the Bellman equation $f = Tf$.

Proof Outline

1. Bounded Value Functions: The proof focuses on bounded functions f : $S \to \mathrm{R}$. We equip F with the supremum norm and aim to show that for any bounded function f, Tf remains bounded.
2. Monotonicity and Cash-Invariance: The properties of monotonicity and cash invariance imply that: $T f \leq T\ //f//\ = T0 +\ //f//$

   Consequently, it follows that $T0 < \infty$, establishing a bound on the operator.

3. Contraction Mapping: To demonstrate that T is a contraction mapping, we show:

$$// \mathrm{Tf - Tg} //\ \leq \beta^{*}\ // \mathrm{f - g} //$$

where $0 < \beta^{*} < 1$. This is achieved by leveraging the properties of monotonicity and cash invariance.

4. Application of Banach Fixed-Point Theorem: Since T is a contraction mapping in a complete metric space (the space of bounded functions), the Banach fixed-point theorem guarantees that there exists a unique fixed point $f*$ such that:

$$Tf^{*}=f^{*}$$

The proof establishes that under the given assumptions, the Deep Bellman Hedging operator admits a unique finite solution. This result is crucial for ensuring that the hedging strategies derived from this framework are well-defined and can be reliably implemented in practice. The methodology not only confirms convergence but also provides a robust foundation for further extensions into unbounded cases as noted in related literature

Conclusion

This whitepaper presented a rigorous framework for Deep Bellman Hedging, modeling financial trading as a continuous state Markov Decision Process (MDP). By defining market states, cash flows, and the time evolution of financial instruments, we captured the dynamics of trading decisions while incorporating the time value of money through discount factors.

The core contribution was proving the existence of a unique finite solution to the Deep Bellman Hedging equations. Using properties like monotonicity and cash invariance, and applying the Banach fixed-point theorem, we demonstrated convergence and uniqueness of solutions, ensuring reliable hedging strategies for portfolio management.

Our framework combines theoretical rigor with practical applicability, offering a robust foundation for managing risk and optimizing trading strategies. This work sets the stage for further advancements in financial modeling and decision-making in dynamic market environments.