Communications in Optimization Theory

Available online at http://cot.mathres.org

DEEP LEARNING METHODS FOR S SHAPED UTILITY MAXIMISATION WITH A RANDOM REFERENCE POINT

ASHLEY DAVEY, HARRY ZHENG*

Department of Mathematics, Imperial College, London SW7 2BZ, UK

Dedicated to Professor Richard B. Vinter on the occasion of his 75th birthday

Abstract. We consider the portfolio optimisation problem where the terminal function is an S-shaped utility applied at the difference between the wealth and a random benchmark process. We develop several numerical methods for solving the problem using deep learning and duality methods. We use deep learning methods to solve the associated Hamilton-Jacobi-Bellman equation for both the primal and dual problems, and the adjoint equation arising from the stochastic maximum principle. We compare the solution of this non-concave problem to that of concavified utility, a random function depending on the benchmark, in both complete and incomplete markets. We give some numerical results for power and log utilities to show the accuracy of the suggested algorithms.

Keywords. Concavification; Dual control; Deep neural network; Incomplete market; S-shaped utility maximisation; Random reference point.

2020 Mathematics Subject Classification. 93E20, 91G80, 90C46, 49M29.

1. Introduction

In this paper we apply the deep learning method to solve utility maximisation problem with a benchmark that is a non-negative random variable. The agent wishes to maximise an S-shaped utility, a function which is concave on $(0,\infty)$ and convex on $(\infty,0)$, applied at the difference between their terminal wealth and the benchmark. This problem is non-standard and difficult to solve due to the non-concavity of the problem. We suggest several numerical methods for solving the problem utilising the concave envelope of the utility and the dual control method.

The random benchmark may represent an asset or portfolio of assets (that could be partly non-tradable) whose payoff we try to replicate. Unlike the deterministic case, the comparison to a random variable allows us to set a reference for utility maximisation that accounts for natural fluctuations in the market. In particular, the benchmark may be proportional to the stock market index. This makes sense intuitively, as the targets for an investor may increase or decrease over

---
*Corresponding author.

E-mail address: ashley.davey18@imperial.ac.uk (A. Davey), h.zheng@imperial.ac.uk (H. Zheng).

time when the market is favourable or challenging. The benchmark may also be affected by factors which cannot be traded, leading to incompleteness of the problem.

The treatment of utility maximisation problems is standard in the literature, see [17] for exposition of the dynamic programming principle (DPP) and the stochastic maximum principle (SMP), and the corresponding Hamilton-Jacobi-Bellman (HJB) equation and the adjoint backward stochastic differential equation (BSDE). The duality approach has been regularly applied to this problem in both complete and incomplete markets, see [4, 13, 14]. The control problem we wish to solve can be seen as an extension of regular utility maximisation problem, where the utility function is random, and also non-concave in the state variable.

Having randomness in the dynamics and gains functions leads to a SMP involving Malliavin derivatives [16], but as long as the random utility has sufficient integrability, the adjoint BSDE admits a solution, we may apply the SMP in the same way as the standard approach of, say, [21]. The duality approach has been considered for a random utility that arises from a random endowment at terminal time, see [5, 10].

The introduction of an S-shaped structure allows us to compare two random variables. However, it is a tricky function to deal with as it is non-concave, and often this utility will have infinite derivative when the variables are equal. Non-concave and potentially discontinuous utilities have been treated in the literature, see [1, 2, 19]. Maximisation of an S-shaped utility compared to a benchmark has been considered in [8, 20], but unlike our setting the benchmark is deterministic.

A key notion for studying non-concave problems is the so called concavification principle. This principle states that we can instead maximise the concave envelope of the utility function and get the same solution as if we solved the non-concave problem directly. The concavified problem is solved with standard approaches. In the deterministic benchmark, complete market and no control constraint case, the concavification principle is known to hold, see [19]. The value function is a smooth solution to the HJB equation in the Markovian setting, see [1], and is the conjugate function of the dual value function that has a simple integral representation and may have a closed-form expression for some utility functions, see [8]. The optimal control drives the wealth to points where the utility matches its concave envelope. In our case, that is the set $\{0\} \cup [Z, \infty)$ for some $Z > 0$ greater than the benchmark. If the wealth is in the range $(0, Z)$ before the terminal time, the optimal control involves considerable risk and creates two possibilities: the terminal wealth either beats the benchmark by a certain amount or becomes 0. However, if in some cases this optimal policy cannot be attained due to control constraints or incompleteness, the value function is not concave and the concavification principle does not apply, see [6].

We use the deep learning method to solve the primal, concavified and dual control problems arising in our setting. A key benefit of deep learning methods is the scalability with dimension [9]. We introduce various PDEs, which can be solved numerically using a Physics Informed Neural Network (PINN), introduced in [18]. The solution function is taken to be a neural network in state and time, and the PDE and terminal condition are evaluated and used as loss functions to find the optimal neural network. To solve the adjoint BSDE associated to the SMP, we use a combination of the deep solver of [11], the deep BSDE method in [12] and the deep SMP method in [7]. In this method the driving process is taken as a neural network and the wealth process as input. The adjoint equation is simulated using Monte Carlo (in the forward

direction), with the driver process determined using the SMP. The terminal condition is used as a loss function for the initial value of the adjoint process. We also use a neural network for the control process, as in [7], and optimise the control process using the gains function as a loss function, accelerated with the terminal condition for the adjoint equation.

The remainder of the paper is outlined as follows. Section 2 formulates the problem and defines the concavified problem. Section 3 considers the case with a scalable power utility and reduces the dimension of the problem, and the PDE method is used to solve the reduced problem and its concavified and dual problems. Section 4 describes the PDE and SMP methods for problems with general utilities. Section 5 presents numerical results comparing all the methods for a range of problems, and Section 6 concludes.

2. PROBLEM FORMULATION

Assume $(\Omega, \mathscr{F}, \{\mathscr{F}_t\}, \mathbb{P})$ is a filtered probability space with $\{\mathscr{F}_t\}$ the natural filtration generated by two standard Brownian motions W, W^R with correlation $\rho \in [-1,1]$, augmented with all $\mathbb{P}$-null sets. Assume the investment period is $[0,T]$ and the risk-free asset price S_0 and the risky asset price S satisfy the following stochastic differential equations (SDEs), for $t \in [0,T]$,

$$dS_t^0 = \alpha S_t^0 dt, \quad dS_t = \mu S_t dt + \sigma S_t dW_t,$$

where α, μ, σ are positive constant, representing the riskless interest rate, stock growth and volatility rates. An investor invests a proportion $\pi \in \mathbb{R}$ of the wealth X in S and the remainder in S^0. The wealth process X satisfies, for $t \in [0,T]$,

$$dX_t = (\alpha X_t + \pi_t X_t \theta \sigma) dt + X_t \pi_t \sigma dW_t$$

with initial wealth $X_0 = x_0 > 0$, where $\theta = (\mu - \alpha)/\sigma$ is the market price of risk and $\pi \in \mathscr{H}_0^2(\mathbb{R})$, the set of all square integrable and progressively measurable processes on $[0,T]$ taking values in $\mathbb{R}$. The investor wants to maximise an S-shaped utility of the excess of the wealth X compared to some positive benchmark process R satisfying, for $t \in [0,T]$,

$$dR_t = aR_t dt + bR_t dW_t^R$$

with initial value $R_0 = r_0 > 0$, where a, b are nonnegative constants. The utility function $U \colon \mathbb{R} \to \mathbb{R}$ is given by, for $z \in \mathbb{R}$,

$$U(z) := \begin{cases} U_1(z) & z \geq 0, \\ -U_2(-z) & z < 0, \end{cases}$$

where $U_1, U_2 \colon [0,\infty) \to \mathbb{R}$ are strictly increasing, strictly concave, continuously differentiable with $U_1(0) = U_2(0) = 0$. The value function is defined by, for $(t,x,r) \in [0,T] \times (0,\infty)^2$,

$$v(t,x,r) := \sup_{\pi \in \mathscr{H}_t^2(\mathbb{R})} \mathbb{E}\left[U(X_T - R_T)|X_t = x, R_t = r\right]. \tag{2.1}$$

The combination of two utilities leads to an S-shaped function U. To solve this problem we use the concave envelope and Fenchel Legendre transforms, applied to x variable, that is, the concave envelope is the smallest concave function (in x) $\bar{U}(x,r)$ that is greater than or equal to $U(x-r)$. By [3] there exists a function $\eta \colon [0,\infty) \to [0,\infty)$ such that $\eta(r) \geq r$ for all $r \in [0,\infty)$ and

$$U_1(\eta(r) - r) + U_2(r) - \eta(r)U_1'(\eta(r) - r) = 0, \tag{2.2}$$

and the concave envelope is given by

$$\bar{U}(x,r) = \begin{cases} -U_2(r) + xU_1'(\eta(r) - r) & x < \eta(r), \\ U_1(x - r) & x \geq \eta(r). \end{cases} \tag{2.3}$$

Note $\bar{U}(\cdot, r)$ is concave, increasing, and C^1 for fixed $r \geq 0$. Define the value function of the concavified problem by, for $(t, x, r) \in [0, T] \times (0, \infty)^2$,

$$\bar{v}(t, x, r) = \sup_{\pi \in \mathscr{H}_t^2(\mathbb{R})} \mathbb{E}\left[\bar{U}(X_T, R_T) | X_t = x, R_t = r\right]. \tag{2.4}$$

The Fenchel Legendre transform of U (in x) is given by, for $(y, r) \in (0, \infty)^2$,

$$\tilde{U}(y, r) := \sup_{x > 0} \{U(x - r) - xy\} = \sup_{x > 0} \{\bar{U}(x, r) - xy\}.$$

Substituting (2.3) into the above definition, we have

$$\tilde{U}(y, r) = \begin{cases} \tilde{U}_1(y) - ry & 0 < y \leq U_1'(\eta(r) - r), \\ -U_2(r) & y > U_1'(\eta(r) - r), \end{cases} \tag{2.5}$$

where $\tilde{U}_1$ is the Fenchel Legendre transform of U_1, and η is determined by (2.2).

It is clear that function $\bar{v}$ is concave in x, $\bar{v} \geq v$ for all for $(x, r) \in (0, \infty)^2$, and $v(T, x, r) \neq \bar{v}(T, x, r)$ on the non-concavity region $C := \{(x, r) \in [0, \infty)^2 : U(x - r) \neq \bar{U}(x, r)\}$. However, under some circumstances equality can hold at terminal time. This is due to the optimal state process X^* satisfying $\mathbb{P}((X_T^*, R_T) \in C) = 0$. The optimal control 'steers' the state process either to a region where U is concave or to 0, points at which $\bar{U}$ equals U.

Theorem 2.1. [3, Theorem 1]. *Suppose the market is complete ($|\rho| = 1$) and the function*

$$\psi(\lambda) = -\mathbb{E}\left[\zeta_T \partial_y \tilde{U}(\lambda \zeta_T, R_T)\right]$$

satisfies $\psi(\lambda) < \infty$ for all $\lambda > 0$, where $\zeta_t = e^{-(r + \frac{1}{2}\theta^2)T - \theta W_T}$. Then there exists $\lambda^* > 0$ such that $\psi(\lambda^*) = x_0$ and

$$X_T^* = -\partial_y \tilde{U}(\lambda^* \zeta_T, R_T).$$

In particular, we have $\mathbb{P}((X_T^*, R_T) \in C) = 0$.

Theorem 2.1 explicitly describes the behaviour of the optimal state process X^* in terms of the state price density ζ. In particular, given the structure of the dual utility $\tilde{U}$ in (2.5), we see that there exists a threshold random variable $\zeta^* := \frac{1}{\lambda^*} U_1'(\eta(R_T) - R_T)$ such that $X_T^* = 0$ if and only if $\zeta_T \geq \zeta^*$.

Remark 2.2. This result comes from [3] in which a utility of the form $U(A(X_T - R_T)^+ + B)$ is considered for some $A, B > 0$, so the utility is constant when $X_T < R_T$, with $U_2 \equiv -U(B)$. However, the proof still holds for a general S-shaped utility with $U(-R_T) = -U_2(R_T) > -\infty$ almost surely.

Given this concavification principle we have $v(t, x, r) = \bar{v}(t, x, r)$ for all $t < T$ in the complete market setting, and we may proceed to solve the concavified problem.

3. PINN FOR POWER UTILITY

In this section we consider the utility, for $z \in \mathbb{R}$,

$$U(z) = \begin{cases} z^p & z \geq 0, \\ -K|z|^p & z < 0, \end{cases}$$

for some $K \geq 0$ and $p \in (0,1)$. In this setting, the dual utility (2.5) becomes, for $y, r > 0$,

$$\tilde{U}(y,r) = \begin{cases} (1-p)\left(\frac{y}{p}\right)^{\frac{p}{p-1}} - ry & 0 < y \leq p(\eta(r)-r)^{p-1}, \\ -Kr^p & y > p(\eta(r)-r)^{p-1}. \end{cases}$$

3.1. Problem Reduction.

Define a non-negative process $Z_t = \frac{X_t}{R_t}$ for $t \in [0,T]$. Then Z_t satisfies the following SDE:

$$dZ_t = Z_t \left(\alpha - a + b^2 + \pi_t \sigma(\theta - \rho b)\right) dt + Z_t \pi \sigma dW_t - Z_t b dW_t^R.$$

Define an exponential martingale $F_t = e^{pbW_t^R - \frac{1}{2}p^2 b^2 t}$ for $t \in [0,T]$ and a new measure $\mathbb{Q}$ by $\frac{d\mathbb{Q}}{d\mathbb{P}} = F_T$. Then by Girsanov's theorem, the processes

$$\tilde{W}_t = W_t - \rho p b t, \quad \tilde{W}_t^R = W_t^R - pbt,$$

are $\mathbb{Q}$ Brownian motions with correlation ρ. SDE for Z can be written equivalently as

$$dZ_t = Z_t \left(\alpha_0 + \pi_t \sigma \bar{\theta}\right) dt + Z_t \pi \sigma d\tilde{W}_t - Z_t b d\tilde{W}_t^R,$$

where $\alpha_0 := \alpha - a - b^2(p-1)$ and $\bar{\theta} := \theta + \rho b(p-1)$. Furthermore, by Bayes' formula (eg [17, Prop. 1.2.12]), we have, for $(t,x,r) \in [0,T] \times [0,\infty) \times (0,\infty)$,

$$v(t,x,r) = \sup_{\pi \in \mathscr{H}_t^2(\mathbb{R})} \mathbb{E}\left[R_T^p U(Z_T - 1)|Z_t = z\right]$$

$$= \sup_{\pi \in \mathscr{H}_t^2(\mathbb{R})} \mathbb{E}^{\mathbb{Q}}\left[\frac{F_t R_T^p}{F_T} U(Z_T - 1)|Z_t = z\right] \tag{3.1}$$

$$= r^p H_{T-t} g(t,z),$$

where $z = x/r$, $H_s := e^{p\left(a + \frac{1}{2}b^2(p-1)\right)s}$ for $0 \leq s \leq T$, and

$$g(t,z) := \sup_{\pi \in \mathscr{H}_t^2(\mathbb{R})} \mathbb{E}^{\mathbb{Q}}[U(Z_T - 1)|Z_t = z]. \tag{3.2}$$

The HJB equation associated to the problem (3.2) is

$$\partial_t g + \sup_{\pi \in \mathbb{R}} \left\{ \partial_z g z \left(\alpha_0 + \pi \sigma \bar{\theta}\right) + \frac{1}{2} \partial_{zz} g z^2 (\pi^2 \sigma^2 - 2\rho b \sigma \pi + b^2) \right\} = 0, \tag{3.3}$$

with the terminal condition $g(T,z) = U(z-1)$ for $z \in [0,\infty)$. The candidate optimal control is given by, for $(t,z) \in [0,T] \times [0,\infty)$,

$$\pi^*(t,z) = -\frac{\partial_z g(t,z)\bar{\theta}}{\partial_{zz} g(t,z) z \sigma} + \frac{\rho b}{\sigma}, \tag{3.4}$$

which leads to a nonlinear PDE

$$\partial_t g + \partial_z g z \alpha_0 + \frac{1}{2} \partial_{zz} g z^2 b^2 - \frac{1}{2} \frac{\left(\partial_z g \bar{\theta} - \partial_{zz} g z \rho b\right)^2}{\partial_{zz} g z \sigma} = 0.$$

Problem (3.2) is simpler as the reference is now a constant. However the function U is still a non-concave function for $z \geq 0$ and non-differentiable at $z = 1$. Define, for $(t,z) \in [0,T] \times [0,\infty)$,

$$\bar{g}(t,z) := \sup_{\pi \in \mathscr{H}_t^2(\mathbb{R})} \mathbb{E}^{\mathbb{Q}}[\bar{U}(Z_T,1)|Z_t = z],$$

where $\bar{U}(z,1)$ is the concave envelope of $U(z-1)$ for $z \geq 0$. The concavified value function $\bar{g}$ satisfies the same equation (3.3) with the concave terminal condition $\bar{g}(T,z) = \bar{U}(z,1)$. In both cases we can use the fact that Z is a geometric Brownian motion for any π, with $Z_T = 0$ if and only if $Z_t = 0$ for any $t < T$, so we can use the boundary condition

$$g(t,0) = \bar{g}(t,0) = U(-1) = -K, \quad 0 \leq t \leq T.$$

If $|\rho| = 1$, it is known that g coincides with $\bar{g}$ on $[0,T) \times (0,\infty)$, see Section 3.3. If $|\rho| < 1$, it is not yet clear if there exists a duality gap between g and $\bar{g}$.

3.2. **Dual Problem.**

Due to incompleteness arising from Brownian motion W^R, the dual problem contains a dynamic dual control and is a true control problem. We determine the dual problem explicitly. Firstly, note that the process Z is non-negative as X and R are non-negative, with R strictly positive. The dual process Y_t has the form, for $0 \leq t \leq T$,

$$dY_t = -Y_t \gamma_t dt - Y_t \beta_t d\tilde{W}_t - Y_t \xi_t d\tilde{W}_t^R,$$

where γ, β, ξ and $Y_0 = y$ are to be determined. We have

$$d(YZ)_t = Y_t Z_t \left(\alpha_0 - \gamma_t + b(\xi_t + \rho\beta_t) + \pi_t \sigma \left(\bar{\theta} - (\rho\xi_t + \beta_t) \right) \right) dt$$
$$+ Y_t Z_t (\pi_t \sigma - \beta_t) d\tilde{W}_t - Y_t Z_t (\xi_t + b) d\tilde{W}_t^R.$$

To ensure that YZ is a $\mathbb{Q}$-supermartingale for all $\pi \in \mathscr{H}_0^2(\mathbb{R})$, we must have

$$\alpha_0 - \gamma_t + b(\xi_t + \rho\beta_t) + \pi_t \sigma \left(\bar{\theta} - (\rho\xi_t + \beta_t) \right) \leq 0,$$

which implies $\beta_t = \bar{\theta} - \rho\xi_t$. We may choose a minimal γ_t as $\gamma_t = \alpha_0 + b(1-\rho^2)\xi_t + b\rho\bar{\theta} = \bar{\alpha} + b(1-\rho^2)\xi_t$, where $\bar{\alpha} := \alpha - a + \rho b\theta$. This gives us the dual process, for $0 \leq t \leq T$,

$$dY_t = -Y_t \left(\bar{\alpha} + b(1-\rho^2)\xi_t \right) dt - Y_t \left(\bar{\theta} - \rho\xi_t \right) d\tilde{W}_t - Y_t \xi_t d\tilde{W}_t^R$$

with dual control process ξ. We then have, for any $\pi, \xi, Y_t = y, Z_t = z, 0 \leq t \leq T$,

$$\mathbb{E}^{\mathbb{Q}}[\bar{U}(Z_T,1)|Z_t = z] \leq \mathbb{E}^{\mathbb{Q}}[\tilde{U}(Y_T,1)|Y_t = y] + yz. \tag{3.5}$$

Define the dual value function $\tilde{g} \colon [0,T] \times (0,\infty) \to \mathbb{R}$ by

$$\tilde{g}(t,y) := \inf_{\xi \in \mathscr{H}_t^2(\mathbb{R})} \mathbb{E}^{\mathbb{Q}}\left[\tilde{U}(Y_T,1) \big| Y_t = y \right]. \tag{3.6}$$

The HJB equation associated to the dual problem (3.6) is

$$\partial_t \tilde{g} + \inf_{\xi \in \mathbb{R}} \left\{ -y\partial_y \tilde{g} \left(\bar{\alpha} + b(1-\rho^2)\xi \right) + \frac{1}{2}y^2 \partial_{yy} \tilde{g} \left(\bar{\theta}^2 + \xi^2(1-\rho^2) \right) \right\} = 0.$$

The candidate dual optimal control is given by, for $(t,y) \in [0,T] \times [0,\infty)$,

$$\xi^*(t,y) = \frac{\partial_y \tilde{g}(t,y)b}{y\partial_{yy}\tilde{g}(t,y)},$$

which leads to a nonlinear PDE

$$\partial_t \tilde{g} - \bar{\alpha} y \partial_y \tilde{g} - \frac{1}{2}(1-\rho^2)\frac{(b\partial_y \tilde{g})^2}{\partial_{yy}\tilde{g}} + \frac{1}{2}y^2 \partial_{yy}\tilde{g}\bar{\theta}^2 = 0$$

with the terminal condition $\tilde{g}(T,y) = \tilde{U}(y,1)$.

In the case $|\rho| = 1$, the HJB equation is independent of ξ, the dual value function is simply given by $\tilde{g}(t,y) = \mathbb{E}^{\mathbb{Q}}\left[\tilde{U}\left(Y_T,1\right)|Y_t = y\right]$ and can be computed explicitly, where Y_T is a lognormal random variable.

Maximising the left side of (3.5) over π and minimising the right side of (3.5) over ξ and y yields the weak duality relation

$$g(t,z) \leq \bar{g}(t,z) \leq \inf_{y>0}\left\{\tilde{g}(t,y) + zy\right\},$$

for all $t \in [0,T]$ and $z \in (0,\infty)$. Since we mainly focus on numerical computation of value functions with deep neural network (DNN), we assume in the rest of the paper that $g,\bar{g},\tilde{g}$ are $C^{1,2}$, the candidate optimal controls are well defined, and all satisfy required integrability conditions when substituted by underlying controlled state processes. We can then characterize and compute them with the corresponding HJB equations.

3.3. Dual Value Function and Concavification Principle in Complete Market Case.

In this section we take $|\rho| = 1$, and evaluate the function $\tilde{g}$ given by (3.6) explicitly. We use this explicit representation to solve the primal concavified problem and show that its solution coincides with the non-concave primal problem. We consider a general constant reference $\bar{R} \in [0,\infty)$, taking $\bar{R} = 1$ recovers (3.6) and $\bar{R} = 0$ recovers the standard utility maximisation problem.

The dynamics of Y in the complete market setting are given by

$$dY_t = -\bar{\alpha}Y_t dt - \bar{\theta}Y_t d\tilde{W}_t, \quad 0 \leq t \leq T,$$

where $\bar{\alpha} = \alpha - a + \rho b\theta$ and $\bar{\theta} = \theta + \rho b(p-1)$. Since Y follows a geometric Brownian motion, after some lengthy calculation, we get that, for $(t,y) \in [0,T] \times (0,\infty)$,

$$
\begin{aligned}
\tilde{g}(t,y) &= \mathbb{E}^{\mathbb{Q}}\left[\tilde{U}(Y_T,\bar{R})\big|Y_t = y\right] \\
&= y^{\frac{p}{p-1}}p^{\frac{p}{1-p}}(1-p)\exp\left(-\frac{p}{p-1}\left(\bar{\alpha} - \frac{|\bar{\theta}|^2}{2(p-1)}\right)\tau\right)\Phi\left(-k(\tau,y) - \frac{p}{p-1}\bar{\theta}\sqrt{\tau}\right) \\
&\quad -\bar{R}y\exp(-\bar{\alpha}\tau)\Phi(-k(\tau,y) - \bar{\theta}\sqrt{\tau}) + u_0\Phi(k(\tau,y)) \\
\partial_y \tilde{g}(t,y) &= -y^{\frac{1}{p-1}}p^{\frac{1}{1-p}}\exp\left(-\frac{p}{p-1}\left(\bar{\alpha} - \frac{|\bar{\theta}|^2}{2(p-1)}\right)\tau\right)\Phi\left(-k(\tau,y) - \frac{p}{p-1}\bar{\theta}\sqrt{\tau}\right) \\
&\quad -\bar{R}\exp(-\bar{\alpha}\tau)\Phi(-k(\tau,y) - \bar{\theta}\sqrt{\tau}) \\
\partial_{yy}\tilde{g}(t,y) &= y^{\frac{2-p}{p-1}}p^{\frac{1}{1-p}}\frac{1}{1-p}\exp\left(-\frac{p}{p-1}\left(\bar{\alpha} - \frac{|\bar{\theta}|^2}{2(p-1)}\right)\tau\right)\Phi\left(-k(\tau,y) - \frac{p}{p-1}\bar{\theta}\sqrt{\tau}\right) \\
&\quad +\phi\left(k(\tau,y)\right)\frac{\hat{z}\hat{u}}{\bar{\theta}\sqrt{\tau}y^2},
\end{aligned}
$$

where Φ is the cumulative distribution function of a standard normal variable, $\phi(x) = \Phi'(x)$, $\tau = T - t$, $\hat{z} = \eta(\bar{R})$, $\hat{u} = U'(\hat{z} - \bar{R}) = p(\hat{z} - \bar{R})^{p-1}$, $u_0 = -U_2(\bar{R}) = -K\bar{R}^p$, and

$$k(\tau, y) := \frac{\log(y) - \log(\hat{u}) - (\bar{\alpha} + \frac{1}{2}|\bar{\theta}|^2)\tau}{\bar{\theta}\sqrt{\tau}}.$$

Note that $\tilde{g} \in C^{1,2}$, $\partial_{yy}\tilde{g}(t,y) > 0$, $\lim_{y\to 0}\partial_y\tilde{g}(t,y) = -\infty$ and $\lim_{y\to\infty}\partial_y\tilde{g}(t,y) = 0$ for all $(t,y) \in [0,T) \times (0,\infty)$, so $\tilde{g}(t,y)$ is a strictly convex decreasing function in y for all $t \in [0,T)$. The optimal state process for the concavified problem is given by

$$Z_t^* = -\partial_y\tilde{g}(t, Y_t^*), \quad 0 \le t \le T, \tag{3.7}$$

where Y^* satisfies the dual state dynamics with the initial value determined from the equation $z_0 = -\partial_y\tilde{g}(0, Y_0^*)$, where $z_0 = \frac{x_0}{r_0}$ is the initial scaled wealth.

The next result shows that $U(Z_T^* - \bar{R}) = \bar{U}(Z_T^*, \bar{R})$ and therefore Z^* is optimal for the original problem.

Lemma 3.1. *The optimal state process Z^* given by (3.7) satisfies $Z_T^* \in \{0\} \cup [\hat{z}, \infty)$ almost surely.*

Proof. In the limit $\tau \to 0$ (or $t \to T$), we have $k(\tau, y) \to \infty$ if $y > \hat{u}$ and $-\infty$ if $y < \hat{u}$. If $y > \hat{u}$ then $\Phi(-k(\tau, y)) \to 0$, hence $Z_t^* \to 0$. If $y < \hat{u}$ then $\Phi(-k(\tau, y)) \to 1$, hence, noting that $y \mapsto y^{\frac{1}{p-1}}$ is a decreasing function,

$$Z_t^* \to y^{\frac{1}{p-1}}p^{\frac{1}{1-p}} + \bar{R} > p^{\frac{1}{1-p}}\left(p(\hat{z} - \bar{R})^{p-1}\right)^{\frac{1}{p-1}} + \bar{R} = \hat{z}.$$

If $y = \hat{u}$ then $\Phi(-k(\tau, y)) \to \Phi(0) = \frac{1}{2}$ and $Z_t^* \to \frac{1}{2}\hat{z} \notin \{0\} \cup [\hat{z}, \infty)$, but $\mathbb{P}(Y_T^* = \hat{u}) = 0$. $\qquad\square$

3.4. PINN Method.
We begin by describing the PINN method for general second order PDEs, with dimensional state in the region $(0,\infty)^d$. Let $F: [0,T) \times (0,\infty)^d \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d\times d} \to \mathbb{R}$, $G: (0,\infty)^d \to \mathbb{R}$ and $H: [0,T) \times \mathbb{R}^d \to \mathbb{R}$ be functions. Consider the PDE

$$F(t, x, u, \partial_t u, \partial_x u, \partial_{xx} u) = 0, \quad t \in [0,T), X \in (0,\infty)^d,$$
$$u(T, X) = G(X), \quad X \in (0,\infty)^d,$$
$$u(t, X) = H(t, X), \quad t \in [0,T), X \in \partial(0,\infty)^d.$$

where $\partial(0,\infty)^d := \{(x_i)_{i=1}^d \in (0,\infty)^d : x_i = 0 \text{ for some } i = 1, \ldots, d\}$ denotes the finite boundary of $(0,\infty)^d$. Let $(t, X) \in [0,T] \times \mathbb{R}^d$. We solve the PDE using the PINN method. The PDE solution is approximated by a neural network

$$u^\Theta(t, X) := \mathcal{N}(t, X; \Theta)$$

based on some parameter set Θ. The function $\mathcal{N}$ is defined as a series of alternating compositions of linear with a nonlinear activation function $h: \mathbb{R} \to \mathbb{R}$ (acting element-wise)

$$\mathcal{N}(t, X; \Theta) = f_M \circ h \circ f_{M-1} \cdots \circ h \circ f_0((t, X)),$$

for some fixed $M \in \mathbb{N}$ (representing the number of 'layers' of the network), where $f_i(z) = A_i z + b_i$ for some matrix A_i and vector b_i of appropriate (fixed) dimension, and we view (t, X) as a single vector in $\mathbb{R}^{1+d}$. The dimension of the final layer is 1, but the middle layers have some pre-determined higher dimension. The larger the dimensions of b_i for $i = 0, \ldots, M-1$, the better approximation capabilities of the neural network [15], but also the more variables in

the network, leading to longer training times. For the activation function, in practise we take $h(z) = \tanh(z)$. The parameter set Θ is made up of the individual elements of $(A_i, b_i)_{i=0}^{M}$, which are optimised by the PINN algorithm, using a loss function defined in the sequel. At step 0 we begin with a parameter set Θ_0 with elements initialised randomly.

We choose a grid size $\mathscr{X} := (X_{\min}, X_{\max})^d \in (0, \infty)^d$, then uniformly sample three batches: the collocation data within the time horizon, given by $(t_j, X_j) \in [0, T] \times \mathscr{X}$ for $j = 1, \ldots, \mathscr{M}_c$, the boundary data $(T, X_j^b) \in \{T\} \times \mathscr{X}$ for $j = 1, \ldots, \mathscr{M}_b$ and the zero state data $(t_j^s, X_j^s) \in [0, T] \times \partial(X_{\min}, X_{\max})^d$ for $j = 1, \ldots, \mathscr{M}_s$. Using these sample data we define the loss function, accounting for the PDE and the terminal and boundary conditions, as a combination of sample averages

$$
\begin{aligned}
L(\Theta) \;=\; & \frac{1}{\mathscr{M}_c} \sum_{j=1}^{\mathscr{M}_c} \left| F\left(t_j, X_j, u^\Theta\left(t_j, X_j\right), \partial_t u^\Theta\left(t_j, X_j\right), \partial_x u^\Theta\left(t_j, X_j\right), \partial_{xx} u^\Theta\left(t_j, X_j\right) \right) \right|^2 \\
& + \frac{1}{\mathscr{M}_b} \sum_{j=1}^{\mathscr{M}_b} \left| u^\Theta\left(T, X_j^b\right) - G\left(X_j^b\right) \right|^2 + \frac{1}{\mathscr{M}_s} \sum_{j=1}^{\mathscr{M}_s} \left| u^\Theta\left(t_j^s, X_j^s\right) - H(t_j^s) \right|^2.
\end{aligned}
$$

In this loss function we assume that $X_{\min} \approx 0$. We may then perform the following update (acting element-wise on the parameters of the neural networks)

$$
\Theta_{k+1} = \Theta_k - \varepsilon \partial_\Theta L(\Theta_k), \quad k \in \mathbb{N},
$$

given some learning rate $\varepsilon > 0$. We repeat this step until convergence - either a fixed number of iteration steps, or a suitably low loss is attained.

We can use the PINN to solve the primal and dual HJB equations. For example, for the primal problem, upon convergence of the algorithm, with some outputted parameter set Θ and neural network function denoted by g^Θ, we rescale and output the following approximation of v at time 0. Given some range $(Z_{\min}, Z_{\max}) \subset \mathbb{R}$ of z values used in the PINN, and some output size $\mathscr{B} \in \mathbb{N}$, we define

$$
\begin{aligned}
Z_i &= \left(Z_{\min} + \frac{i}{\mathscr{B}}(Z_{\max} - Z_{\min}) \right), \\
X_i &= r_0 Z_i, \\
v_i &= r_0^p H_T g^\Theta(0, Z_i), \\
\pi_i &= -\frac{\partial_z g^\Theta(0, Z_i) \bar{\theta}}{\partial_{zz} g^\Theta(0, Z_i) Z_i \sigma} + \frac{\rho b}{\sigma},
\end{aligned}
\tag{3.8}
$$

for $i = 1, \ldots, \mathscr{B}$, where H is defined in (3.1).

The dual problem can be solved similarly. We do not optimise the dual start value y, which is needed to solve the primal problem. However we end up with a range of values of $\partial_y \tilde{g}(t, y)$ which correlate with the z. When there is no duality gap, we have the following relation between

the dual and primal problems for $t \in [0, T]$

$$
\begin{aligned}
\bar{g}(t, Z_t^*) &= \tilde{g}(t, Y_t^*) + Y_t^* Z_t^*, \\
Z_t^* &= -\partial_y \tilde{g}(t, Y_t^*), \\
\partial_z \bar{g}(t, Z_t^*) &= Y_t^*, \\
\partial_{zz} \bar{g}(t, Z_t^*) &= \frac{-1}{\partial_{yy} \tilde{g}(t, Y_t^*)}.
\end{aligned}
\tag{3.9}
$$

Remark 3.2. In practice we do not know if these are true, especially in the incomplete market case, where the value function may not be differentiable and there may be a duality gap. We can still evaluate these terms though, as the neural network function is continuously differentiable, and compare them to their primal counterparts to see if there is a gap.

Upon convergence of the algorithm, with some outputted parameter set $\tilde{\Theta}$ and neural network function $\tilde{g}^{\tilde{\Theta}}$, we rescale and output the following approximation of v at time 0. Given some range $(Y_{\min}, Y_{\max}) \subset \mathbb{R}$ of y values used in the PINN, and some output size $\mathscr{B} \in \mathbb{N}$, we define

$$
\begin{aligned}
Y_i &= \left(Y_{\min} + \frac{i}{\mathscr{B}}(Y_{\max} - Y_{\min}) \right), \\
X_i^d &= -r_0 \partial_y \tilde{g}^{\tilde{\Theta}}(0, Y_i), \\
v_i^d &= r_0^p H_T \left(\tilde{g}^{\tilde{\Theta}}(0, Y_i) - Y_i \partial_y \tilde{g}^{\tilde{\Theta}}(0, Y_i) \right), \\
\pi_i^d &= -\frac{Y_i \bar{\theta} \partial_{yy} \tilde{g}^{\tilde{\Theta}}(0, Y_i)}{\partial_y \tilde{g}^{\tilde{\Theta}}(0, Y_i) \sigma} + \frac{\rho b}{\sigma},
\end{aligned}
\tag{3.10}
$$

for $i = 1, \ldots, \mathscr{B}$, where H is defined in (3.1).

4. GENERAL UTILITY

We return to the original problem for general utilities U_1, U_2, given by (2.1). Since we mainly focus on numerical computation of value functions with DNN, we again assume all regularity and integrability conditions are satisfied for all functions whenever needed and all equations are well defined with existence and uniqueness of solutions. We can then characterize value functions, optimal controls, etc. with the DPP or the SMP and solve the corresponding HJB equation or the adjoint BSDE with DNN.

4.1. **PINN for General Utility.** The value function v is a classical solution to the HJB equation

$$
\begin{aligned}
0 &= \partial_t v + \sup_{\pi \in \mathbb{R}} \left\{ x(\alpha + \pi \sigma \theta) \partial_x v + \frac{1}{2} x^2 \pi^2 \sigma^2 \partial_{xx} v + ar \partial_r v + \frac{1}{2} b^2 r^2 \partial_{rr} v + \rho \sigma \pi b r x \partial_{xr} v \right\}, \\
&= \partial_t v + x\alpha \partial_x v + ar \partial_r v + \frac{1}{2} b^2 r^2 \partial_{rr} v - \frac{1}{2} \frac{(\theta \partial_x v + \rho br \partial_{xr} v)^2}{\partial_{xx} v}
\end{aligned}
\tag{4.1}
$$

with the terminal condition $v(T, x, r) = U(x - r)$. The optimiser π^* is given by, for $(t, x, r) \in [0, T] \times (0, \infty)^2$,

$$
\pi^*(t, x, r) = -\frac{\theta \partial_x v(t, x, r) + \rho br \partial_{xr} v(t, x, r)}{\sigma x \partial_{xx} v(t, x, r)}.
\tag{4.2}
$$

Similarly for the concave problem (2.4) the function $\bar{v}$ satisfies (4.1) with the terminal condition $\bar{v}(T,x,r) = \bar{U}(x,r)$. We may define the dual problem here, similar to (3.6), for $(t,x,r) \in [0,T] \times [0,\infty)^2$,

$$\tilde{v}(t,y,r) = \inf_{\xi \in \mathscr{H}_t^2(\mathbb{R})} \mathbb{E}\left[\tilde{U}(Y_T, R_T)\big| Y_t = y, R_t = r\right]$$

subject to

$$dY_t = -Y_t \alpha dt - Y_t(\theta - \rho \xi_t)dW_t - Y_t \xi_t dW_t^R, \quad 0 \le t \le T.$$

The HJB equation is

$$0 = \partial_t \tilde{v} + \inf_{\xi}\left\{-y\alpha \partial_y \tilde{v} + \frac{1}{2}y^2\left(\theta^2 + \xi^2(1-\rho^2)\right)\partial_{yy}\tilde{v}\right.$$
$$\left. + ar\partial_r v + \frac{1}{2}b^2r^2\partial_{rr}\tilde{v} - rby\left(\rho\theta + \xi(1-\rho^2)\right)\partial_{yr}\tilde{v}\right\},$$
$$= \partial_t \tilde{v} - y\alpha\partial_y \tilde{v} + \frac{1}{2}y^2\theta^2\partial_{yy}\tilde{v} + ar\partial_r\tilde{v} + \frac{1}{2}b^2r^2\partial_{rr}\tilde{v} - rby\rho\theta\partial_{yr}\tilde{v} - \frac{1}{2}\frac{(rb\partial_{yr}\tilde{v})^2(1-\rho^2)}{\partial_{yy}\tilde{v}}$$

with the terminal condition $\tilde{v}(T,y,r) = \tilde{U}(y,r)$.

Again, we have three PDEs that the PINN method may be applied to. The difference here is that the state process becomes multidimensional, with $d = 2$. The output of these algorithms is the same as (3.8) and (3.10), applied at a range of r values. We can use the boundary conditions, for $(t,r) \in [0,T] \times [0,\infty)$,

$$v(t,0,r) = \bar{v}(t,0,r) = \mathbb{E}\left[-U_2(R_T)\right].$$

Additionally, when $r = 0$ we can use the standard utility maximisation result, which is a special case of the solution in Section 3.3, given by, for $(t,x) \in [0,T] \times [0,\infty)$,

$$v(t,x,0) = e^{p\left(\alpha + \frac{\theta^2}{2(1-p)}\right)(T-t)}x^p.$$

4.2. SMP for General Utility.

As an alternative to the HJB method, which requires a Markovian structure, we may apply the SMP instead. In this method we treat the reference R as a random variable (that we can simulate) and treat the utility as a random function, taking in $(\omega, x) \in \Omega \times (0,\infty)$ as input. To apply the SMP as it is given in the literature, for example [21], we assume a complete market here, so R_T is replicable. We also require the terminal utility to be concave, so we apply the SMP to the concave problem.

For this section we change the problem slightly, defining $\Pi_t = \pi_t X_t$ and letting this be our control variable. This means that the control represents the amount invested in the stock, rather than the proportion of wealth invested. The state process dynamics becomes

$$dX_t = (rX_t + \sigma\theta\Pi_t)dt + \sigma\Pi_t dW_t, \quad 0 \le t \le T. \tag{4.3}$$

In this case, X is no longer a geometric Brownian motion and may be negative. We therefore require Π to be in the admissible control set, defined by, for $(t,x) \in [0,T] \times [0,\infty)$,

$$\mathscr{A}(t,x) = \left\{\Pi \in \mathscr{H}_t^2(\mathbb{R})\big| X_s^{t,x,\Pi} \ge 0 \text{ a.s. for all } s \in [t,T]\right\},$$

where $X^{t,x,\Pi}$ satisfies (4.3) with $X_t^{t,x,\Pi} = x$. Now we define the random utility $U^R \colon \Omega \times [0,\infty) \to \mathbb{R}$ by $U^R(\omega, x) := \bar{U}(x, R_T(\omega))$, and the problem becomes, for $(t,x,r) \in [0,T] \times [0,\infty)^2$,

$$v(t,x,r) = \sup_{\Pi \in \mathscr{A}(t,x)} \mathbb{E}\left[U^R(X_T) \middle| X_t = x, R_t = r\right].$$

If this expectation cannot be determined explicitly, we evaluate it numerically. Let $(\hat{X}, \hat{\Pi})$ be an optimal state-control pair. Define the adjoint BSDE by, $t \in [0,T]$,

$$dp_t = -\alpha p_t dt + q_t dW_t, \quad p_T = -\partial_x \bar{U}(\hat{X}_T, R_T). \tag{4.4}$$

The SMP (see [21, Theorem 3.2]) states that the pair $(\hat{X}, \hat{\Pi})$ is optimal if and only if for all $\Pi \in \mathbb{R}$, $t \in [0,T]$,

$$\sigma(\theta p_t + q_t)(\hat{\Pi}_t - \Pi) \geq 0,$$

which implies $q_t = -\theta p_t$, and p has a closed-from expression $p_t = p_0 e^{-(\alpha + \frac{1}{2}\theta^2)t - \theta W_t}$ for $0 \leq t \leq T$. We need to choose the initial point p_0 and the control $\hat{\Pi}$ such that the terminal condition in (4.4) is satisfied. To this end, we define two neural networks as follows. For $(t,x) \in [0,T] \times [0,\infty)$,

$$\Pi^{\Theta_\Pi}(t,x) = \mathscr{N}(t,x;\Theta_\Pi), \quad p^{\Theta_p}(x) = \mathscr{N}(x;\Theta_p),$$

where Θ_Π and Θ_p denote the parameters of two neural networks, with some fixed number of hidden layers and nodes per layer. Let $t_i = i\Delta t$, $i = 0, \ldots, N$, $\Delta t = T/N$. We choose a grid size $\mathscr{X} := (X_{\min}, X_{\max}) \in (0,\infty)$. We then define the loss function, for a given batch of $X_0^j \in \mathscr{X}, \Delta W_i^j \sim \text{Norm}(0,1)$, $j = 1, \ldots, \mathscr{M}$, $i = 1, \ldots, N-1$ as

$$\mathscr{L}(\Theta_\Pi, \Theta_p) = \frac{1}{\mathscr{M}} \sum_{j=1}^{\mathscr{M}} \left| e^{-(\alpha + \frac{1}{2}\theta^2)T - \theta W_T^j} p^{\Theta_p}(X_0^j) + \partial_x \bar{U}(X_N^j, R_T) \right|^2 - \frac{1}{\mathscr{M}} \sum_{j=1}^{\mathscr{M}} \bar{U}(X_N^j, R_T), \tag{4.5}$$

where

$$X_{i+1}^j = X_i^j + \left(\alpha X_i^j + \Pi^{\Theta_\Pi}(t_i, X_i^j)\theta\sigma\right)\Delta t + \Pi^{\Theta_\Pi}(t_i, X_i^j)\sigma\sqrt{\Delta t}\Delta W_i^j,$$

$$W_N^j = \sqrt{\Delta t} \sum_{i=0}^{N-1} \Delta W_i^j.$$

We then iteratively perform the gradient descent update, for $k \in \mathbb{N}$,

$$\Theta_\Pi^{k+1} = \Theta_\Pi^k - \lambda \partial_\Pi \mathscr{L}(\Theta_\Pi^k, \Theta_p^k)$$
$$\Theta_p^{k+1} = \Theta_p^k - \lambda \partial_p \mathscr{L}(\Theta_\Pi^k, \Theta_p^k),$$

where $\partial_\Pi \mathscr{L}$ and $\partial_p \mathscr{L}$ denote the partial derivatives of $\mathscr{L}$ with respect to the first and second element respectively, and $\lambda > 0$ is some learning rate. We interpret this update as the operation occurring to each individual element of the sets Θ_Π and Θ_p. Note that while the first element of (4.5) should go to 0, the second term evaluates the gains function so will not be 0, but should converge to the value function as $k, \mathscr{M} \to \infty$. If we only had the second term in the loss function this method would not involve the adjoint equation and reduce to the method of [11].

Our $\mathscr{M}$ is typically small (in practise we use $\mathscr{M} = 500$), so the loss function does not provide a precise approximation of the value function. To output the value function for this method, we

compute $\mathbb{E}[\bar{U}(X_T, R_T)]$ where X is driven by Π^{Θ_Π} at time 0 using Monte Carlo with a much higher sample size, with the converged control parameter set Θ_Π.

Upon convergence of the algorithm, with some outputted parameter set Θ_Π and neural network function Π^{Θ_Π}, we output the following approximation of v at time 0. Given some output size $\mathscr{B} \in \mathbb{N}$ we define

$$
\begin{aligned}
X_i^s &= \left(X_{\min} + \frac{i}{\mathscr{B}}(X_{\max} - X_{\min}) \right), \\
v_i^s &= \mathbb{E}^s \left[\bar{U}(X_T^{i,\Pi^{\Theta_\Pi}}, R_T) \right], \\
\Pi_i^s &= \Pi^{\Theta_\Pi}(0, X_i),
\end{aligned}
\tag{4.6}
$$

for $i = 1, \ldots, \mathscr{B}$, where $\mathbb{E}^s$ here indicates a sample mean of sufficiently large sample size, and $X^{i,\Pi^{\Theta_\Pi}}$ is the wealth process satisfying (4.3) starting at X_i and driven by Π^{Θ_Π}.

5. NUMERICAL EXAMPLES

We take the problem coefficients $\alpha = 0.05$, $\sigma = 0.2$, $\theta = 0.5$, $K = 0.5$, $T = 0.5$, $a = 0.03$, $b = 0.1$. When we consider the power utility, we take $p = 0.5$. We take the neural network to have 4 layers, with 2 hidden layers consisting of 50 nodes, and a tanh activation function. We sample points using $\mathscr{Z} = \mathscr{X} = (0.05, 5)$ and $\mathscr{Y} = (0.25, 1.0)$ for the SMP, primal and dual PINN methods respectively. We sample R_0 using the same range for the general PINN methods, and fixed to a constant for the SMP method (either 0.5, 1.0, or 5.0 for our results). We take sample sizes of $\mathscr{M}_c = 1000$, $\mathscr{M}_b = \mathscr{M}_s = 100$, $\mathscr{M} = 500$, and run Monte Carlo simulations with $N = 100$. We use 20000 iteration steps for the PINN methods and 1000 steps for the SMP method, with a learning rate of $\varepsilon = 0.001$ and $\varepsilon = 0.01$ for the PINN and SMP methods respectively. If the loss function is below 5×10^{-5}, we terminate the algorithm early. We generate equally spaced points along our grids, evaluate the neural network functions with converged parameter sets, and compare this to a plot of the solutions (red), evaluated (also numerically) using the dual solutions in Section 3.3.

We run the algorithms using a PC with a Intel Core i9 9900K and 16.0GB DDR4 RAM, without GPU acceleration. The runtime for the algorithms for this setup is roughly 1 minute for the scaled PINN methods and 2 minutes for the general PINN methods, though the primal concavified PINN method often terminates early due to low loss. The SMP method takes about 1 minute. We run the algorithms 10 times and present the average results.

See `https://github.com/Ashley-Davey/ML-for-S-Utility` for the code to reproduce the results in this section.

5.1. **Converting from Scaled Problem to Regular Problem.**
For the problem with scalable utility, we use the PINN methods to find the function g, and this function is found in the solution as well. To convert to the value function v we use (3.1). To recover the optimal control in (4.2) we use (3.4).

5.2. **Converting from Dual Problem to Primal Problem.**
We solve for the function $\tilde{v}$ using the dual problem. To convert to the primal problem we output the candidate primal value function as, for $(t, x, r) \in [0, T] \times [0, \infty)^2$,

$$
v^d(t, x, r) = \tilde{v}(t, y(t, x, r), r) + xy(t, x, r),
\tag{5.1}
$$

where $y(t,x,r)$ is the minimiser of $y \mapsto \tilde{v}(t,y,r) + xy$. If there is no duality gap, we have $v^d = v$ as the true value function. To recover the candidate optimal control representing (4.2) we note

$$\partial_x v^d(t,x,r) = y(t,x,r),$$

$$\partial_{xx} v^d(t,x,r) = \frac{-1}{\partial_{yy}\tilde{v}(t,y(t,x,r),r)},$$

$$\partial_{xr} v^d(t,x,r) = \frac{-\partial_{yr}\tilde{v}(t,y(t,x,r),r)}{\partial_{yy}\tilde{v}(t,y(t,x,r),r)}.$$

Therefore the candidate optimal control is given by

$$\pi^d(t,x,r) = \frac{\theta y(t,x,r)\partial_{yy}\tilde{v}(t,y(t,x,r),r) - \rho br \partial_{yr}\tilde{v}(t,y(t,x,r),r)}{\sigma x}.$$

5.3. **Complete Market Problem, Scalable Utility.** For this example we take $\rho = 1$, $U_1(x) = \sqrt{x}$, $U_2(x) = \frac{1}{2}\sqrt{x}$. We compare our methods to the solution found using the dual method given in Section 3.3. We apply 5 methods to this problem, the PINN method may be applied to the scaled problem, to the primal problem with both original (labelled as g) and concavified utility (labelled as $\bar{g}$), and to the dual problem. We plot X against $\Pi = \pi X$ and v as they are given in (3.8) and (3.10). We also apply the SMP algorithm, plotting X against Π and v as they are given in (4.6).

Figure 1 shows accuracy of the control and value function for all methods. There is variance in the control function due to the dependence of π on the second derivative $\partial_{xx} v$, which is often approximated with less accuracy than the value function its self. In terms of control structure, we see a bump between 0 and the reference R, which matches our intuition that the agent is more risky here to ensure the wealth moves to either 0 or $\eta(R_T)$ by the terminal time, so the concave utility of wealth agrees with the non-concave utility. As time approaches the terminal time, the control in this region goes to infinity. Tables 1 and 2 provide the value function approximations for all our methods with different initial values.

Remark 5.1. The non-concave method performs poorly when the state x is much lower than the reference r. This is likely due to the jump between the concave and non-concave utilities leading to a discontinuity in time of the value function at terminal time. Indeed, in the deterministic reference case with complete market, by [1, Theorem 3.9] the value function $v(t,x,r)$ will converge to $\bar{U}(x,r)$ as $t \to T$. This leads to a lack of continuity in the value function (it is only a viscosity solution to the HJB with non-concave terminal condition). We therefore cannot expect the value function to be replicated uniformly by a neural network function that is continuous, and equal to U at terminal time.
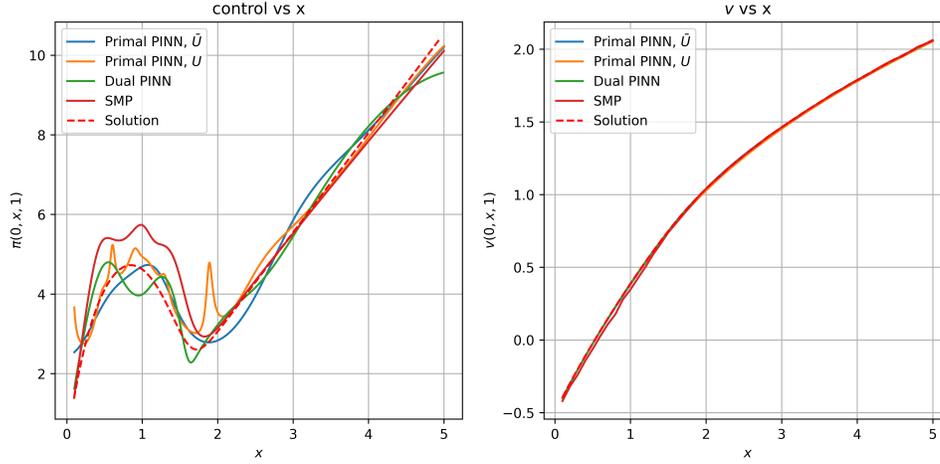
FIGURE 1. The (amount invested) control and value functions Π, v against state x, evaluated at $t = 0$ with complete market ($\rho = 1.0$), $r = 1.0$, and scalable utility. The red dotted line is the solution derived in Section 3.3.

(x, r)	Primal	Primal Non-Conc	Dual	Solution
(0.5, 1.0)	-0.0256	-0.0337	-0.0030	-0.0149
(1.0, 1.0)	0.3844	0.3750	0.3940	0.3872
(5.0, 1.0)	2.0618	2.0606	2.0610	2.0653
(1.0, 0.5)	0.7316	0.7274	0.7246	0.7353
(1.0, 5.0)	-0.6822	-0.6988	-0.6670	-0.6635

TABLE 1. The value functions v against state x and reference point r, evaluated at $t = 0$ with complete market ($\rho = 1.0$), and scalable utility. Results show the PINN-based algorithms that reduce dimensionality of the problem, solving the scaled problem with $z = \frac{x}{r}$, then rescaling.

(x, r)	Primal General	Primal Non-Conc General	Dual General	SMP	Solution
(0.5, 1.0)	-0.0188	-0.0577	-0.0050	-0.0450	-0.0149
(1.0, 1.0)	0.3793	0.3340	0.3866	0.3735	0.3872
(5.0, 1.0)	2.0546	2.0631	2.0787	2.0678	2.0653
(1.0, 0.5)	0.7335	0.7429	0.7320	0.7361	0.7353
(1.0, 5.0)	-0.6758	-0.8197	-0.6551	-0.6885	-0.6635

TABLE 2. The value functions v against state x and reference point r, evaluated at $t = 0$ with complete market ($\rho = 1.0$), and scalable utility. Results show the PINN and SMP algorithms that solve the original problem without scaling.

5.4. **Incomplete Market Problem.** For this example we take $\rho = 0$, $U_1(x) = \sqrt{x}$, $U_2(x) = \frac{1}{2}\sqrt{x}$. Figure 2 shows accuracy of the control and value function for all methods. We do not

have an analytical solution in this case to compare to. Compared to Figure 1, we see little difference in the structure of the control, and concavity in the value function. The value function approximations for all our methods are provided in Tables 3 and 4. We apply all the methods from the previous example, even though the SMP and concavificiation principles might not apply in the incomplete market case. However, the concave primal, and dual PINN methods appear to agree, suggesting the concavification principle still occurs. As the primal and dual values coincide, up to numerical errors, we conclude that the payoff $X_T = -\partial_y \tilde{U}(Y_T, R_T)$ has been replicated by the primal algorithm, where Y_T is the optimal payoff for the dual problem. Given this, we conjecture that the concavification principle holds for this problem. These numerical results show robustness of our methods outside of problem classes that are well understood.
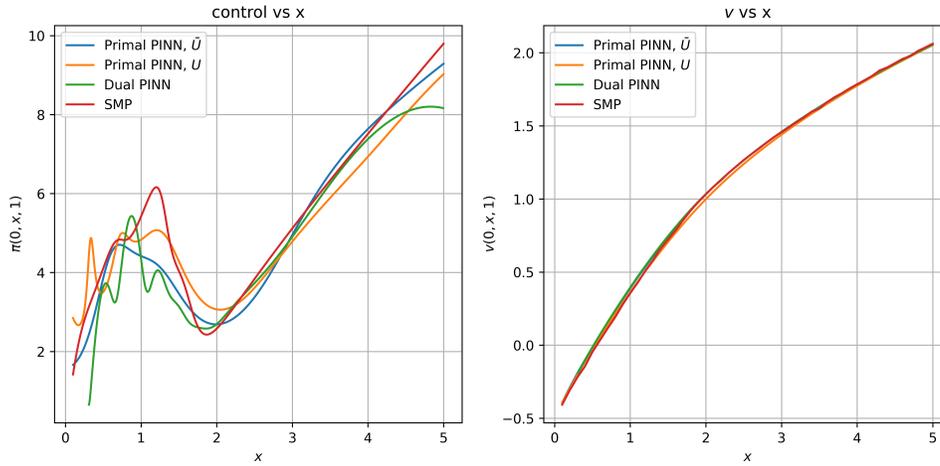

FIGURE 2. The (amount invested) control and value functions Π, v against state x, evaluated at $t = 0$ with incomplete market ($\rho = 0.0$), $r = 1.0$, and scalable utility.

(x, r)	Primal	Primal Non-Conc	Dual
(0.5, 1.0)	-0.0144	-0.0391	0.0084
(1.0, 1.0)	0.3931	0.3535	0.4045
(5.0, 1.0)	2.0583	2.0521	2.0706
(1.0, 0.5)	0.7288	0.7022	0.7211
(1.0, 5.0)	-0.6650	-0.6888	-0.6162

TABLE 3. The value functions v against state x and reference point r, evaluated at $t = 0$ with incomplete market ($\rho = 0.0$), and scalable utility. Results show the PINN-based algorithms that reduce dimensionality of the problem, solving the scaled problem with $z = \frac{x}{r}$, then rescaling.

(x, r)	Primal General	Primal Non-Conc General	Dual General	SMP
(0.5, 1.0)	-0.0066	-0.0635	0.0080	-0.0274
(1.0, 1.0)	0.3895	0.3042	0.4087	0.3999
(5.0, 1.0)	2.0517	2.0516	2.0610	2.0578
(1.0, 0.5)	0.6930	0.6844	0.6987	0.7297
(1.0, 5.0)	-0.6551	-0.7665	-0.6667	-0.6725

TABLE 4. The value functions v against state x and reference point r, evaluated at $t = 0$ with incomplete market ($\rho = 0.0$), and scalable utility. Results show the PINN and SMP algorithms that solve the original problem without scaling.

5.5. **General Utility.** For this example we take $\rho = 1$, $U_1(x) = \sqrt{x}$, $U_2(x) = \frac{1}{2}\log(x+1)$. The value function approximations for our methods are provided in Table 5. We cannot reduce this problem, so we apply the methods for solving general S-shaped utilities. In this example we compare the general PINN algorithms for the primal, primal concave, and dual problems, and the SMP. The results we see are familiar. In this setting the concavification principle holds, and we expect no duality gap. However, the non-concave PINN method does not perform well in the regions with $x < r$, see Remark 5.1.

(x, r)	Primal General	Primal Non-Conc General	Dual General	SMP
(0.5, 1.0)	0.0770	0.0327	0.0770	0.0496
(1.0, 1.0)	0.4342	0.3638	0.4415	0.4230
(5.0, 1.0)	2.0537	2.0654	2.0642	2.0613
(1.0, 0.5)	0.7045	0.6184	0.7111	0.7274
(1.0, 5.0)	-0.4916	-0.6596	-0.4742	-0.4829

TABLE 5. The value functions v against state x and reference point r, evaluated at $t = 0$ with complete market ($\rho = 1.0$), and general utility with no scaling property.

6. CONCLUSIONS

In this paper we study the S-shaped utility maximisation problem, with a random benchmark that may not be replicable due to incompleteness. We introduce the deep neural network method for solving the HJB equation and the adjoint BSDE with the optimality condition from the SMP. We present numerical results for different solution methods (HJB equations for primal, concavified primal, dual value functions, SMP for adjoint BSDE, reduced and full problems) and show that the DNN method is viable and robust in solving nonlinear PDEs and BSDEs. This paper has not addressed theoretical convergence of DNN nor general concavification principle. Convergence analysis for the PINN method in particular is not available beyond linear PDEs, and our HJB equations are fully nonlinear PDEs. We leave these to future research.

Acknowledgements

References

[1] B. Bian, X. Chen, and Z. Q. Xu, Utility maximization under trading constraints with discontinuous utility, SIAM Journal on Financial Mathematics 10 (2019) 243-260.

[2] L. Carassus and H. Pham, Portfolio optimization for piecewise concave criteria functions, The 8th workshop on stochastic numerics (2009) 81-108.

[3] J. N. Carpenter, Does option compensation increase managerial risk appetite? The Journal of Finance 55 (2000) 2311-2331.

[4] J. Cvitanić and I. Karatzas, Convex duality in constrained portfolio optimization, The Annals of Applied Probability 2 (1992) 767-818.

[5] J. Cvitanić, W. Schachermayer, and H. Wang, Utility maximization in incomplete markets with random endowment, Finance and Stochastics 5 (2001) 259-272.

[6] M. Dai, S. Kou, S. Qian, and X. Wan, Non-concave utility maximization without the concavification principle, Management Science 68 (2022) 8368-8385.

[7] A. Davey and H. Zheng, Deep learning for constrained utility maximisation, Methodology and Computing in Applied Probability 24 (2022) 661-692.

[8] Y. Dong and H. Zheng, Optimal investment with s-shaped utility and trading and value at risk constraints: An application to defined contribution pension plan, European Journal of Operational Research 281 (2020) 341-356.

[9] P. Grohs, F. Hornung, A. Jentzen, and P. VonWurstemberger, A proof that artificial neural networks overcome the curse of dimensionality in the numerical approximation of black-scholes partial differential equations, Memoirs of the American Mathematical Society 284 (2018) 1410.

[10] L. Gu, Y. Lin, and J. Yang, On the dual problem of utility maximization in incomplete markets, Stochastic Processes and their Applications 126 (2016) 1019-1035.

[11] J. Han, Deep learning approximation for stochastic control problems. arXiv:1611.07422, 2016.

[12] J. Han, A. Jentzen, and W. E, Solving high-dimensional partial differential equations using deep learning, Proceedings of the National Academy of Sciences 115 (2018) 8505-8510.

[13] I. Karatzas, J. P. Lehoczky, S. E. Shreve, and G.-L. Xu, Martingale and duality methods for utility maximization in an incomplete market, SIAM Journal on Control and optimization 29 (1991) 702-730.

[14] D. Kramkov and W. Schachermayer, The asymptotic elasticity of utility functions and optimal investment in incomplete markets, Annals of Applied Probability 9 (1999) 904-950.

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

[16] T. Meyer-Brandis, B. Øksendal, and X. Y. Zhou, A mean-field stochastic maximum principle via malliavin calculus, Stochastics. An International Journal of Probability and Stochastic Processes 84 (2012) 643-666.

[17] H. Pham, Continuous-Time Stochastic Control and Optimization with Financial Applications, Springer, 2009.

[18] M. Raissi, P. Perdikaris, and G. E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, Journal of Computational physics 378 (2019) 686-707.

[19] C. Reichlin, Utility maximization with a given pricing measure when the utility is not necessarily concave. Mathematics and Financial Economics 7 (2013) 531-556.

[20] A. S. Tse and H. Zheng, Speculative trading, prospect theory and transaction costs, Finance and Stochastics 27 (2023) 49-96.

[21] J. Yong and X. Y. Zhou, Stochastic Controls: Hamiltonian Systems and HJB Equations, Springer, 2012.