Solving Merton's Portfolio Problem

PINN Approach

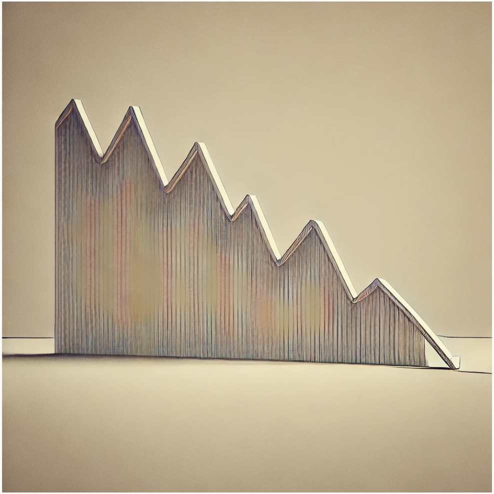
Jaehoon Yoo (Sungkyunkwan University)
Hyelin Choi (Sungkyunkwan University)
Jung Hun Phee (Yonsei University)

Objective

- Our goal is to solve Merton's Portfolio Problem using Physics-Informed Neural Networks (PINN).
- We will then compare this solution with the exact solution.

What is Portfolio

Motivation



Merton's Portfolio Problem

- $W_t > 0$: Wealth at time t
- Assume the current wealth is $W_0 > 0$, and you will live for T more years.
- You can invest in a risky asset and a riskless asset.

Informal Problem Statement

- Riskless asset: $dR_t = r \cdot R_t \cdot dt$
- Risky asset: $dS_t = \mu \cdot S_t \cdot dt + \sigma \cdot S_t \cdot dz_t$
- r: risk free rate
- μ: expected return on stock
- σ: volatility
- $\mu > r > 0$, $\sigma > 0$

The goal of Merton's portfolio problem is to maximize the lifetime-aggregated utility of consumption by selecting the optimal allocation and consumption at each time point.

- We focus on the Optimal Value Function $V^*(t, W_t)$

$$V^*(t, W_t) = \max_{\pi, c} \mathbb{E}_t \left[\int_t^T \frac{e^{-\rho(s-t)} \cdot c_s^{1-\gamma}}{1-\gamma} ds + \frac{e^{-\rho(T-t)} \cdot \epsilon^\gamma \cdot W_T^{1-\gamma}}{1-\gamma} \right]. \tag{1}$$

Optimal Value Function PDE

After some calculations, Equation (1) is transformed into the Optimal Portfolio Value Function PDE:

$$\frac{\partial V^*}{\partial t} - \frac{(\mu - r)^2}{2\sigma^2} \cdot \frac{\left(\frac{\partial V^*}{\partial W_t}\right)^2}{\frac{\partial^2 V^*}{\partial W_t^2}} + \frac{\partial V^*}{\partial W_t} \cdot r \cdot W_t + \frac{\gamma}{1-\gamma} \cdot \left(\frac{\partial V^*}{\partial W_t}\right)^{\frac{\gamma-1}{\gamma}} = \rho V^*. \quad (2)$$

The terminal condition is:

$$V^*(T, W_T) = \epsilon^\gamma \cdot \frac{W_T^{1-\gamma}}{1-\gamma}.$$

Exact Solution

Reducing Equation (2) to ODE, we get the solution:

$$V^*(t, W_t) = f(t)^\gamma \cdot \frac{x^{1-\gamma}}{1-\gamma}$$

where

$$f(t) = \begin{cases} \frac{1+(\nu\epsilon-1)\cdot e^{-\nu(T-t)}}{\nu} & \text{for } \nu \neq 0 \\ T - t + \epsilon & \text{for } \nu = 0. \end{cases}$$

and

$$\nu = \frac{\rho - (1-\gamma)\cdot\left(\frac{(\mu-r)^2}{2\sigma^2\gamma} + r\right)}{\gamma} \tag{3}$$

with terminal condition $V(T, W_T) = \epsilon^\gamma \cdot \frac{W_T^{1-\gamma}}{1-\gamma}$.

Results

Letting $\tau := T - t$ and $x := W_t$, equation (2) is written as

$$-\frac{\partial V}{\partial \tau} - \frac{(\mu - r)^2}{2\sigma^2} \cdot \frac{\left(\frac{\partial V}{\partial x}\right)^2}{\frac{\partial^2 V}{\partial x^2}} + \frac{\partial V}{\partial x} \cdot r \cdot x + \frac{\gamma}{1-\gamma} \cdot \left(\frac{\partial V}{\partial x}\right)^{\frac{\gamma-1}{\gamma}} = \rho V$$

The initial condition is:

$$V^*(0, W_0) = \epsilon^\gamma \cdot \frac{W_0^{1-\gamma}}{1-\gamma}.$$

Constants

$\mu = 0.07$ drift / expected return on stock
$r = 0.01$ risk free rate
$\sigma = 0.2$ volatility
$T = 1$ maturity
$\gamma = 0.3$ relative risk-aversion
$\rho = 0.02$ utility discount rate
$\epsilon = 0.1$ no bequest(small constant)
$L = 2.0;$ maximum wealth

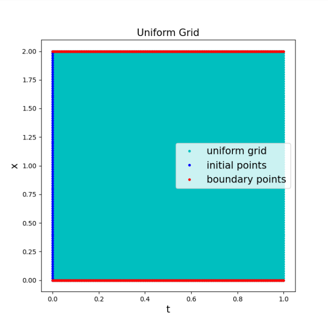
Initialization

Since we have to solve numerically, we must truncate the spatial domain. $(t_i, x_j) \in [0, T] \times [0, L]$ with $\Delta t = \frac{T-0}{n_t}$ and $\Delta x = \frac{L-0}{n_x}$

$i = 1, \ldots, n_t = 365$ number of time step

$j = 1, \ldots, n_x = 100$ Number of spatial grid

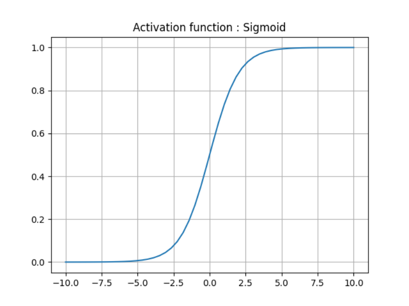
Initialized in Xavier initialization.



Structure

Our model has 4 hidden layers with 200 perceptrons for each layers.

Activation function : Sigmoid.



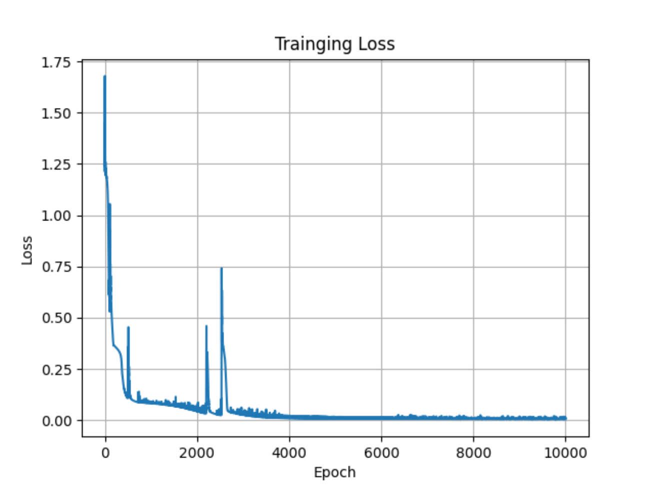
Training for 10000 epochs

Loss function and Optimizer

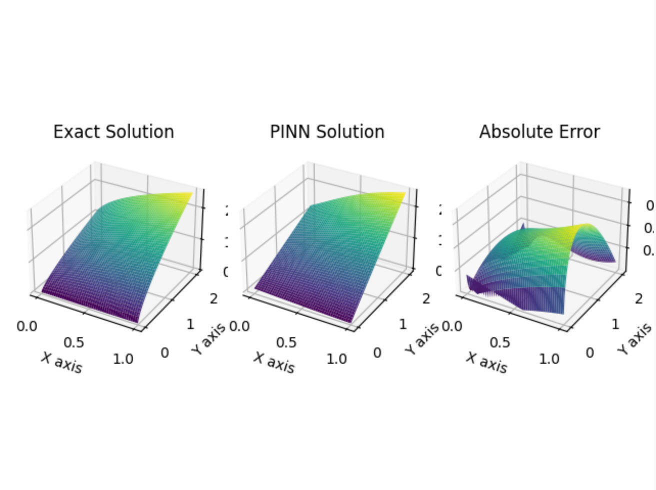
Optimizer : Adam with learning rate 1e-3

Loss function is sum of MSElosses of initial condition, boundary condition, and the given equation.

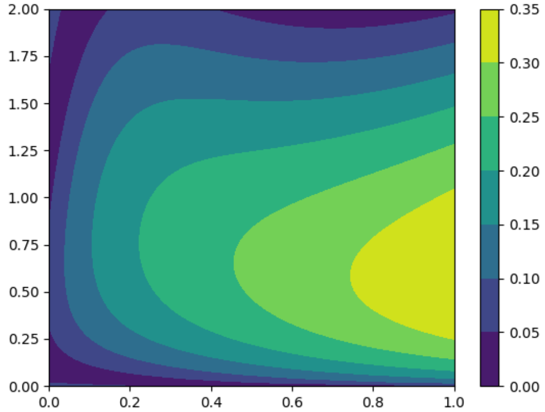
Results


Trainging Loss
Loss
Epoch
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
2000
4000
6000
8000
10000

Solutions



Exact Solution
PINN Solution
Absolute Error
X axis
Y axis
X axis
Y axis
X axis
Y axis


2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.0
0.2
0.4
0.6
0.8
1.0
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00

Discussion I

- It depends on the domain, especially for the case the domain has 0. Since the give PDE is non-linear, it divides with 0 that ∞.
- Also, 0 makes the dependencies on the choosing activation function. ReLU, Tanh, CeLU do not work for this case.
- According to [WLHW22], it may be more sufficient to train when choose proper loss function not with L^p loss function.
- How it makes difference between numerical differentiation and Autograd. There are many ways to differentiate or approximate to the derivative not just using Autograd. How much it makes difference?
- Too dangerous to use in finance.

Robert C. Merton.
Lifetime portfolio selection under uncertainty: The continuous-time case.
The Review of Economics and Statistics, 51(3):247–257, 1969.

Robert C Merton.
Optimum consumption and portfolio rules in a continuous-time model.
Journal of Economic Theory, 3(4):373–413, 1971.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis.
Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations, 2017.

Chuwei Wang, Shanda Li, Di He, and Liwei Wang.
Is l^2 physics informed loss always suitable for training physics informed neural network?
Advances in Neural Information Processing Systems, 35:8278–8290, 2022.