Pstat 222 Spring 2025 – Asn 3

Hand in all Code used for computations.

Problem 1: Explicit FD scheme for 2D American Options

Consider the two-dimensional GBM model (namely a Black-Scholes model for two correlated stock prices):

$$dS_t^1 = (r - d_1)S_t^1 dt + \sigma_1 S_t^1 dW_t^1 \tag{1}$$
$$dS_t^2 = (r - d_2)S_t^2 dt + \sigma_2 S_t^2 dW_t^2 \tag{2}$$

where d_i are the dividend yields and the driving Wiener processes W^1, W^2 have constant correlation ρ. Take $S_0^1 = S_0^2 = 40$, $\sigma_1 = 0.3, \sigma_2 = 0.2, r = 0.06$, $d_1 = 0, d_2 = 0.03$ and $\rho = 0.6$.

(a) Implement an explicit FD solver (as a function that can be called) for the above problem and the **American Basket Call payoff** $\Phi(t, S^1, S^2) = e^{-rt}(\frac{S^1+S^2}{2} - K)_+$. Use $T = 1$ year for maturity.

Submit a 3d surface plot of the initial option value $V(0, S^1, S^2)$ and comment on what you see. Also, show a plot of the stopping region at $t = 0.5$ and across $S^1(t), S^2(t)$.

(b) Adjust your code to solve the **Bermudan** option case – namely that exercise is not at every time step, but only at a pre-specified frequency δ_{freq}. Take $\delta_{freq} = 1/6, 1/12, 1/24, 1/365$ (so every two months, every month, bi-monthly and daily) and compare the resulting Bermudan Put values at the initial condition $(0, S_0^1, S_0^2)$. Also compare to the European option value (no early exercise allowed). Discuss how the option value depends on δ_{freq}.

Problem 2: The Two-Factor Markovian Path Dependent Volatility Model

Consider the following model where the volatility is a function of the past returns.

$$dS_t = S_t\sigma(R_{1,t}, R_{2,t})\, dW_t, \qquad \sigma(R_1, R_2) = \beta_0 + \beta_1 R_1 + \beta_2\sqrt{R_2} \tag{3}$$
$$dR_{1,t} = \lambda_1\big(\sigma(R_{1,t}, R_{2,t})dW_t - R_{1,t}dt\big) \tag{4}$$
$$dR_{2,t} = \lambda_2\big(\sigma(R_{1,t}, R_{2,t})^2 - R_{2,t}\big)dt. \tag{5}$$

Note that there is a *single* Brownian motion everywhere, so the share price S_t and its volatility processes R_1, R_2 share the same stochastic shocks. Use $\beta_0 = 0.08, \beta_1 = -0.08, \beta_2 = 0.5, \lambda_1 = 62, \lambda_2 = 40$ and initial values $S_0 = 10, R_{1,0} = -0.044, R_{2,0} = 0.007$.

We are interested in computing the Delta in this model, i.e. the sensitivity of the option value with respect to the initial share price S_0. We will consider Calls with payoff $(S_T - K)_+$, $K = 10$ and horizon of $T = 0.33$ (approx 4 months).

(a) Consider the bump-and-revalue strategy: pricing the option starting at $S_0 \pm \epsilon$ and looking at the difference. The pricing is done via Monte Carlo; use $h = 0.001$ or about 3 time steps per day. For this to work, you need to use the same Brownian paths to average out the noise. Implement bump-and-revalue across a range of initial conditions $S_0 \in \{8, 8.1, 8.2, \dots, 12\}$, keeping $R_{1,0}, R_{2,0}$ fixed. Use $\epsilon = 0.005$ and $M = 1000$ *fresh* paths at each different S_0. Hand in a plot of the resulting $S_0 \mapsto \hat{\Delta}(S_0)$.

(b) Generate training data at $S_0 \in \{8, 8.1, 8.2, \dots, 12\}$ and $M = 1000$. Fit a GP surrogate with a Matern-5/2 kernel and optimized lengthscales (via MLE optimization). Use the analytical GP derivative to predict $\widehat{Delta} = \partial \hat{P}(S_0)/\partial S_0$. Compare to the results in part (a).

For reference see: Volatility is (Mostly) Path Dependent by Guyon and Lekeufack (2022).

Problem 3

Implement a Neural Network surrogate with exactly 1 hidden layer for pricing Calls in the model of Problem 2, again based on training samples coming from Monte Carlo in (a) with $M = 1000$ and 1000 training inputs. Approximate the Delta through directly evaluating the gradient of the fitted NN. Use 30 neurons and try the ReLU, tanh, and ELU activation functions. Comment on which activation function appears to work best and compare to the results in Problem 2. If you have time, try more hidden layers (same training set) and see if the architecture can improve the results.