Neural Network Solvers for Parametric PDEs in Stochastic Volatility Models

Supervisor: Linus Wunderlich

Ivelina Mladenova

Queen Mary University of London, September 5, 2025

Outline

This part

Motivation: What is an option worth?

Option Contract

Option contracts give the holder the **right but not obligation** to buy (or sell) an underlying asset at some point in the future at a predetermined price.

Motivation: What is an option worth?

Option Contract

Option contracts give the holder the **right but not obligation** to buy (or sell) an underlying asset at some point in the future at a predetermined price.

Notation

- S_t: stock price at time t; S_T: stock price at maturity T.
- K: strike price (the predetermined price).
- r: risk-free interest rate (continuously compounded).
- T: option maturity (in years); $\tau = T - t$ time to maturity.
- $(\cdot)^+ := \max(\cdot, 0)$; e.g., call payoff $(S_T - K)^+$.

Motivation: A (European) call option

Option Contract

Option contracts give the holder the **right but not obligation** to buy (or sell) an underlying asset at some point in the future at a predetermined price.

Notation

- S_t: stock price at time t; S_T: stock price at maturity T.
- K: strike price (the predetermined price).
- r: risk-free interest rate (continuously compounded).
- T: option maturity (in years); $\tau = T - t$ time to maturity.
- $(\cdot)^+ := \max(\cdot, 0)$; e.g., call payoff $(S_T - K)^+$.
- A (European) call option pays $(S_T - K)^+$ at maturity T.

The value of these contracts at expiration is easily determined, but what should we pay **today**?

Motivation: The Pricing Idea

A (European) call option pays $(S_T - K)^+ := \max(S_T - K, 0)$ at maturity T.

The Pricing idea: present value of a risk-neutral expectation

$$V(t, S_t) = e^{-r(T-t)}\,\mathbb{E}^{\mathbb{Q}}\big[(S_T - K)^+ \mid \mathcal{F}_t\big].$$

where $\mathbb{Q}$ is the risk-neutral measure. [1]

[1] We use the risk-neutral measure because under no-arbitrage, the discounted price process must be a martingale under $\mathbb{Q}$, ensuring the price today equals the expected discounted future payoff.

Motivation: The Pricing Idea

A (European) call option pays $(S_T - K)^+ := \max(S_T - K, 0)$ at maturity T.

The Pricing idea: present value of a risk-neutral expectation

$$V(t, S_t) = e^{-r(T-t)}\,\mathbb{E}^{\mathbb{Q}}\big[(S_T - K)^+ \mid \mathcal{F}_t\big].$$

where $\mathbb{Q}$ is the risk-neutral measure. [2]

To compute it, we need two ingredients:

- A **model** for the stock price trajectory S_t, at some time t;
- A **method** to evaluate the expectation.

[2]We use the risk-neutral measure because under no-arbitrage, the discounted price process must be a martingale under $\mathbb{Q}$, ensuring the price today equals the expected discounted future payoff.

This part

Geometric Brownian Motion (GBM)

Model

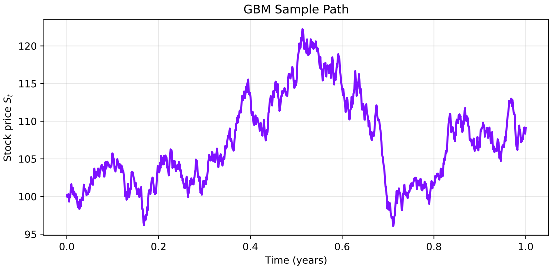
$$dS_t = \mu S_t \, dt + \sigma S_t \, dW_t, \quad S_0 > 0.$$

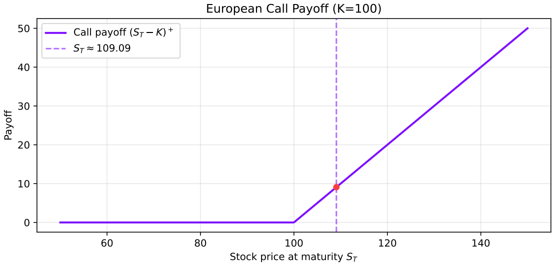
Under the risk-neutral measure $\mathbb{Q}$ (replace μ by r):

$$dS_t = r S_t \, dt + \sigma S_t \, dW_t^{\mathbb{Q}}.$$

- Closed-form solution: $S_t = S_0 \exp\left((r - \frac{1}{2}\sigma^2)t + \sigma W_t^{\mathbb{Q}}\right)$.
- Log-returns are normally distributed, volatility is *constant*.

Geometric Brownian Motion (GBM)





The Method: From GBM to the Black-Scholes PDE (idea only)

Two equivalent routes (sketch), (Hull):

1. **No-arbitrage / replication**: Hedge the option with $\Delta = \partial_S V$; eliminate risk; earn $r \Rightarrow$ PDE.
2. **Feynman-Kac**: Write price as discounted expectation under $\mathbb{Q}$; identify the governing PDE.

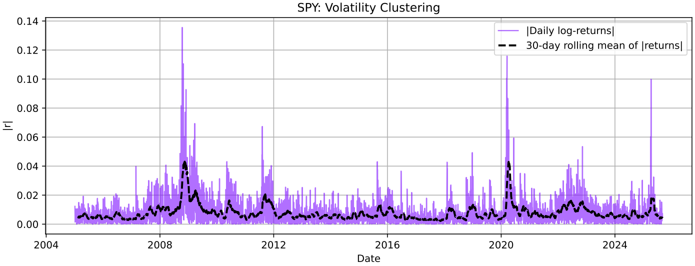
Black-Scholes PDE

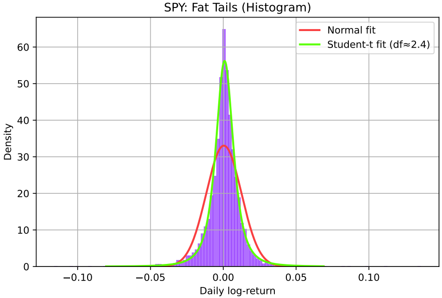
$$\partial_t V + \frac{1}{2}\sigma^2 S^2 \, \partial_{SS} V + rS \, \partial_S V - rV = 0, \quad V(T, S) = (S - K)^+.$$

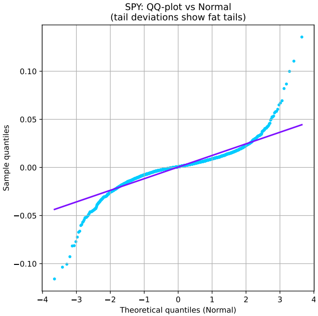
Leads to the **Black-Scholes formula** for European calls/puts, (Black and Scholes).

This part

Stylized Facts: clustering & fat tails (SPY, data from 2005-01-01)







Stylized Facts: What GBM Misses

In 2001, Rama Cont released a paper on the empirical observations of stock prices, some of are:

- **Volatility is not constant**: it *clusters* over time.
- **Returns have fat tails** and negative skew.
- **Implied volatility smiles/smirks**: BS needs a different σ for each strike/maturity. See (Gatheral).
- And more, see (Cont).

This part

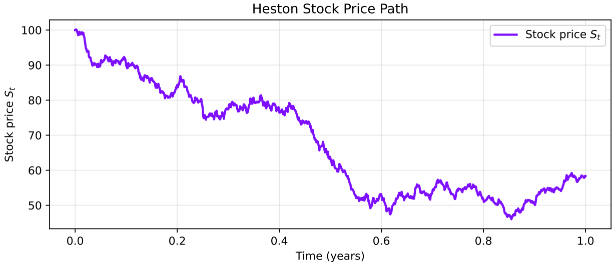
The Heston Stochastic-Volatility Model

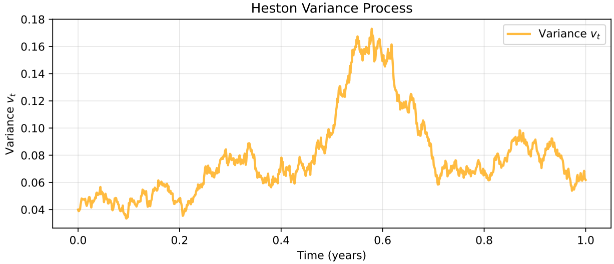
Dynamics under $\mathbb{Q}$

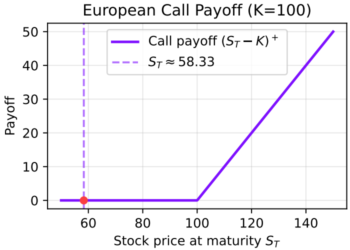
$$
\begin{aligned}
dS_t &= rS_t\,dt + \sqrt{v_t}\,S_t\,dW_t^{(1)}, \\
dv_t &= \kappa(\theta - v_t)\,dt + \sigma\sqrt{v_t}\,dW_t^{(2)}, \\
\mathrm{corr}\big(dW_t^{(1)}, dW_t^{(2)}\big) &= \rho.
\end{aligned}
$$

- Mean-reverting variance $v_t \geq 0$ (CIR-type process).
- Captures **volatility clustering** and **implied-volatility smiles**.
- Still *affine* $\Rightarrow$ admits a closed-form *characteristic function* for $\log S_T$.
- See (Heston).

The Heston Stochastic-Volatility Model and Payoff







European call payoff, $K = 100$, marker at S_T.

Outline

This part

Numerical Methods for Pricing Options

Monte Carlo

- Simulate (S_t, v_t); discount payoffs; variance reduction for efficiency.

Fourier / Characteristic Function

- Price via risk-neutral characteristic function $\phi(u) = \mathbb{E}^{\mathbb{Q}}[e^{iu \log S_T}]$; invert to get option price (Carr and Madan).
- **COS method**: Fourier-cosine expansion from ϕ; high accuracy with few terms (Fang and Oosterlee).

Numerical Methods for Pricing Options

Monte Carlo

- Simulate (S_t, v_t); discount payoffs; variance reduction for efficiency.

Fourier / Characteristic Function

- Price via risk-neutral characteristic function $\phi(u) = \mathbb{E}^{\mathbb{Q}}[e^{iu \log S_T}]$; invert to get option price (Carr and Madan).
- **COS method**: Fourier-cosine expansion from ϕ; high accuracy with few terms (Fang and Oosterlee).

PDE solvers

- Finite differences in (t, S, v); careful boundaries and stability.

Deep Learning (DGM)

- Learn $V_\theta(t, S, v)$ by minimizing PDE residual + BC/IC losses; mesh-free, scalable (Sirignano and Spiliopoulos).

This part

Heston PDE: essentials for pricing

Domain/terminal: $(t, S, v) \in [0, T) \times (0, \infty) \times [0, \infty), \quad V(T, S, v) = (S - K)^+$.

Boundary conditions (European call)

- $S \to 0$: $V \approx 0$.
- $S \to \infty$: $V \approx S - Ke^{-r\tau}$ (or enforce $V_S \approx 1$).
- $v \to 0$: *degenerate* limit (diffusions drop):

$$\partial_t V + rSV_S + \kappa\theta\, V_v - rV = 0.$$

Parameters & positivity

$$\kappa > 0, \quad \theta > 0, \quad \sigma \geq 0, \quad \rho \in [-1, 1], \quad r \geq 0, \quad v_0 \geq 0.$$

Feller condition: $2\kappa\theta \geq \sigma^2$

No-arbitrage / martingale pricing: for $V(t, S, v) = \mathbb{E}^{\mathbb{Q}}[e^{-r(T-t)}\,\Phi(S_T) \mid S_t = S, v_t = v]$ (with payoff Φ), Itô + martingale condition $\Rightarrow$ *Feynman–Kac*

$$\partial_t V + \mathcal{L}V - rV = 0, \qquad V(T, S, v) = \Phi(S)$$

Heston PDE: essentials for pricing

Domain/terminal: $(t, S, v) \in [0, T) \times (0, \infty) \times [0, \infty)$, $\quad V(T, S, v) = (S - K)^+$.

Boundary conditions (European call)

- $S \to 0$: $V \approx 0$.
- $S \to \infty$: $V \approx S - Ke^{-r\tau}$ (or enforce $V_S \approx 1$).
- $v \to 0$: *degenerate* limit (diffusions drop):

$$\partial_t V + rSV_S + \kappa\theta\, V_v - rV = 0.$$

Parameters & positivity

$$\kappa > 0, \quad \theta > 0, \quad \sigma \geq 0, \quad \rho \in [-1, 1],$$
$$r \geq 0, \quad v_0 \geq 0.$$

Feller condition: $\quad 2\kappa\theta \geq \sigma^2$

No-arbitrage / martingale pricing: for $V(t, S, v) = \mathbb{E}^{\mathbb{Q}}[e^{-r(T-t)}\,\Phi(S_T) \mid S_t = S, v_t = v]$ (with payoff Φ), Itô + martingale condition $\Rightarrow$ *Feynman–Kac*

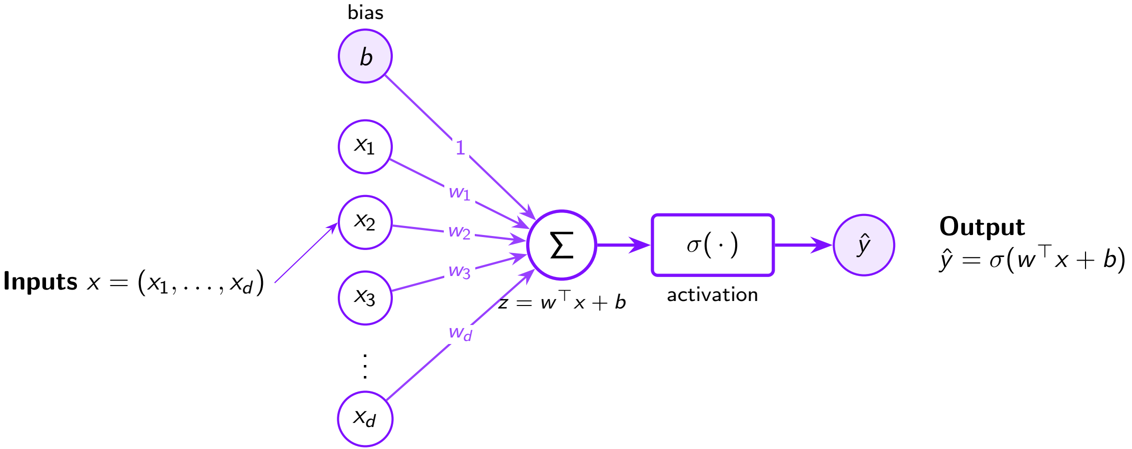
$$\partial_t V + \mathcal{L}V - rV = 0, \qquad V(T, S, v) = \Phi(S)$$

which gives (expanding $\mathcal{L}$) the familiar Heston PDE:

$$\boxed{\partial_t V + \tfrac{1}{2}vS^2 V_{SS} + \rho\sigma S\sqrt{v}\, V_{Sv} + \tfrac{1}{2}\sigma^2 v\, V_{vv} + rSV_S + \kappa(\theta - v)V_v - rV = 0}$$

This part

Perceptron: the single neuron



Notation: Neural Network parameters: $w = (w_1, \ldots, w_d)$ weights, b bias,
Pre-activation $z = w^\top x + b$
σ activation function (e.g. ReLU/Tanh/Sigmoid), single neuron/node output $\hat{y} = \sigma(z)$.

MLP: parameters & supervised loss (quick refresher)

Model (L-layer MLP): map $x \in \mathbb{R}^d \mapsto \hat{y}_\theta$

$$
\begin{aligned}
h^{(0)} &= x, & h^{(\ell)} &= \sigma^{(\ell)}\big(z^{(\ell)}\big), \\
z^{(\ell)} &= W^{(\ell)} h^{(\ell-1)} + b^{(\ell)}, & \ell &= 1, \dots, L, \\
\hat{y}_\theta &= h^{(L)}.
\end{aligned}
$$

Parameters: $\theta = \{W^{(\ell)}, b^{(\ell)}\}_{\ell=1}^{L}$. Weights: $W^{(\ell)} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$, biases: $b^{(\ell)} \in \mathbb{R}^{n_\ell}$.

MLP: parameters & supervised loss (quick refresher)

Model (L-layer MLP): map $x \in \mathbb{R}^d \mapsto \hat{y}_\theta$

$$\begin{aligned} h^{(0)} &= x, & h^{(\ell)} &= \sigma^{(\ell)}\big(z^{(\ell)}\big), \\ z^{(\ell)} &= W^{(\ell)} h^{(\ell-1)} + b^{(\ell)}, & \ell &= 1, \ldots, L, \\ \hat{y}_\theta &= h^{(L)}. \end{aligned}$$

Parameters: $\theta = \{W^{(\ell)}, b^{(\ell)}\}_{\ell=1}^{L}$. Weights: $W^{(\ell)} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$, biases: $b^{(\ell)} \in \mathbb{R}^{n_\ell}$.

Classic Supervised loss (labels available):

$$\mathcal{L}_{\text{sup}}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \ell\big(y_i, \hat{y}_\theta(x_i)\big) \; + \; \lambda\, \Omega(\theta),$$

- Regression (e.g. fit prices): $\ell(y, \hat{y}) = \|y - \hat{y}\|_2^2$ (MSE).
- Regularization: $\Omega(\theta) = \sum_\ell \|W^{(\ell)}\|_F^2$ (weight decay).

Interpretation: We have target labels (x_i, y_i) and train the network to match them.

This part

From supervised to DGM loss

Goal: learn the call/put fair price $V_\theta(t, S, v)$ *without labels* by enforcing the Heston PDE and conditions.

Model: the Deep Galerkin Method (DGM) **has no labels** (Sirignano and Spiliopoulos).
Instead it uses:

- **inputs** are time, states and the parameters $(t, S, v, \kappa, \theta, \sigma, \rho, r)$;
- **outputs** the approximated option price $V_\theta(t, S, v)$.

From supervised to DGM loss

Goal: learn the call/put price $V_\theta(t, S, v)$ *without labels* by enforcing the Heston PDE and conditions.

Model: the Deep Galerkin Method (DGM) **has no labels**, see (Sirignano and Spiliopoulos). Instead it uses:

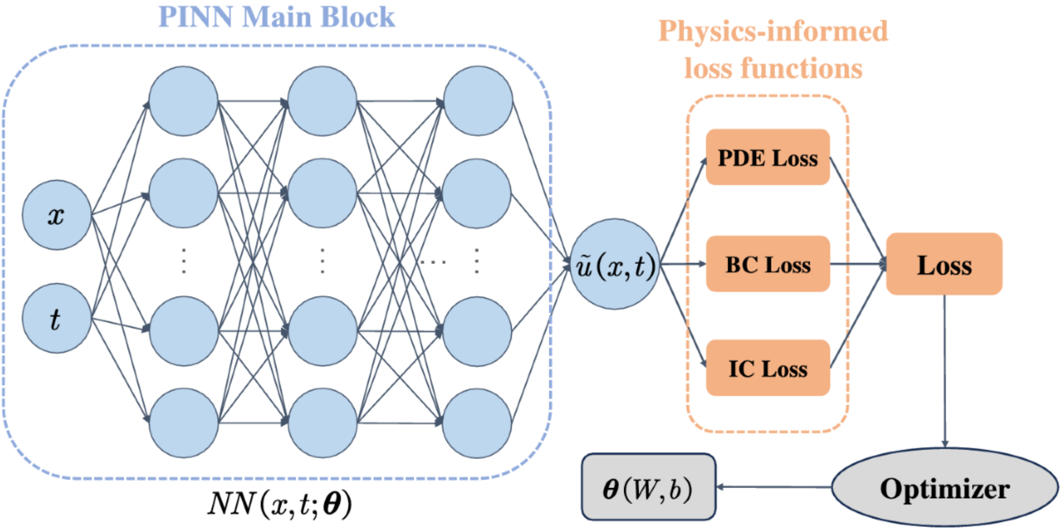
- **inputs** are time, states and the parameters $(t, S, v, \kappa, \theta, \sigma, \rho, r)$;
- **outputs** the approximated option price $V_\theta(t, S, v)$.

Composite ("physics-informed") loss:

$$\mathcal{L}_{\mathrm{DGM}}(\theta) = \underbrace{\mathbb{E}_\Omega\big[\|\mathcal{N}[V_\theta]\|^2\big]}_{\text{PDE residual}} + \lambda_{\mathrm{term}} \underbrace{\mathbb{E}_{\partial\Omega_T}\big[\|V_\theta(T, S, v) - (S-K)^+\|^2\big]}_{\text{terminal}} + \lambda_{\mathrm{bdry}} \underbrace{\mathbb{E}_{\partial\Omega}\big[\|\mathcal{B}[V_\theta]\|^2\big]}_{\text{boundaries}},$$

where $\mathcal{N}$ is the Heston operator and $\mathcal{B}$ enforces BCs.

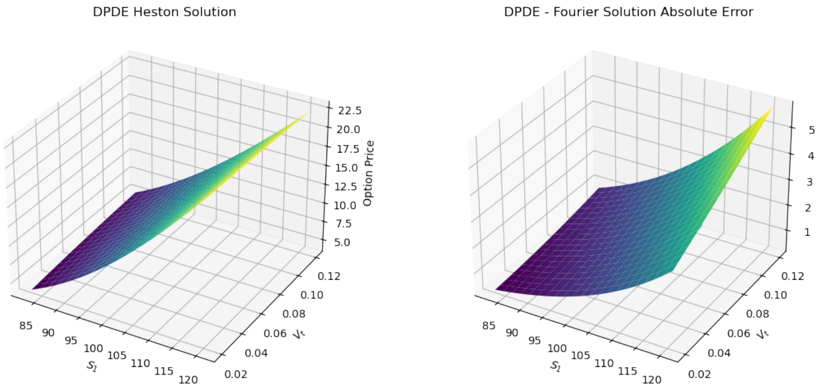
From supervised to DGM loss



Source: (Luo et al.)

This part

Direct application of the Parametric DGM fails to yeild satisfactory results



Parametric DGM solution for the Heston model.

Black Scholes-residual (with $\sigma(v) = \sqrt{v}$): new DGM loss

Decompose: $V = u_{\mathrm{BS}}(x,t,v) + U$, where for each fixed v, u_{BS} solves the Black Scholes PDE in $x = \log S$ with volatility $\sigma(v) = \sqrt{v}$ and terminal $(S-K)^+$.

Source term (apply Heston operator to u_{BS}):

$$\boxed{g(x,t,v) \;=\; \kappa(\theta - v)\,u_v \;+\; \tfrac{1}{2}\sigma^2\,v\,u_{vv} \;+\; \rho\,\sigma\,v\,u_{xv}}$$

(the x-drift/diffusion parts cancel since both have $\frac{1}{2}v\,\partial_{xx}$).

PDE for the correction U:

$$\boxed{\partial_t U + \mathcal{L}_H U - rU = g(x,t,v)}$$

$$\mathcal{L}_H = rS\,\partial_S + \kappa(\theta - v)\,\partial_v + \tfrac{1}{2}vS^2\,\partial_{SS} + \rho\sigma S\sqrt{v}\,\partial_{Sv} + \tfrac{1}{2}\sigma^2 v\,\partial_{vv}.$$

NEW DGM loss:

$$\boxed{\mathcal{L}(\theta) = \mathbb{E}_{\Omega}\big[\|\mathcal{N}_H[U_\theta] - g\|^2\big] + \lambda_T\,\mathbb{E}_{\partial\Omega_T}\big[\|U_\theta\|^2\big] + \lambda_B\,\mathbb{E}_{\partial\Omega}\big[\|\widetilde{\mathcal{B}}[U_\theta]\|^2\big]}$$

Black Scholes-residual (with $\sigma(v) = \sqrt{v}$): Boundaries

Boundaries/terminal for U (European call):

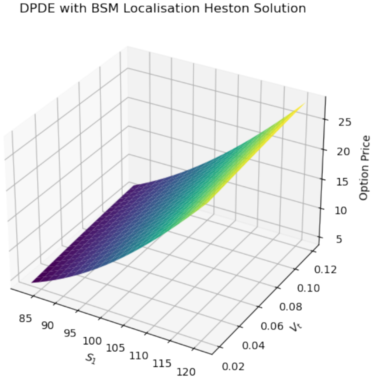
- $t = T$: $U(T, S, v) = 0$ (since $u_{\mathrm{BS}}(T, S, v) = (S - K)^+$).
- $S \to 0$, $S \to \infty$: $U \to 0$ (BS matches call asymptotics).
- $v \to 0$: degenerate first-order boundary for U with source $g|_{v=0}$.

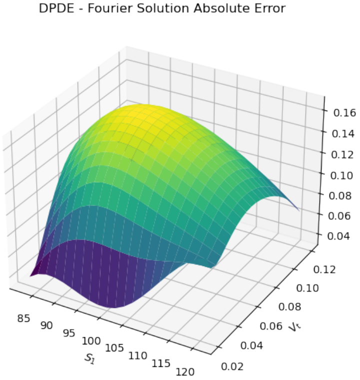
In practice, we don't sample exactly at $v = 0$ since $\sqrt{v}$ and $\partial_v \sqrt{v} = 1/(2\sqrt{v})$ are ill-conditioned at 0. So we clip $0 < v_{\mathrm{min}} \leq v$ to avoid the $\sigma(v) = \sqrt{v}$ singularity.

In fact, we truncate the domain to $v \in [v_{min}, v_{max}]$ with a tiny $v_{min} \sim 10^{-6}$.

Note. If instead you anchor at a constant σ_0 (no v-dependence), then $g = \frac{1}{2}(\sigma_0^2 - v)S^2\, \partial_{SS} u_{\mathrm{BS}}$ and the v–derivative terms vanish.

Using the known Black-Scholes solution helps improve the accuracy





Parametric DGM solution for the Heston model.

Summary

- **GBM $\Rightarrow$ Black–Scholes:** elegant baseline; misses smiles and volatility clustering.
- **Heston:** stochastic variance captures stylized facts; affine structure enables fast Fourier/COS pricing.
- **Classical pricers:** Fourier/COS (vanillas: fastest/most accurate), PDE/FD (Greeks/control), MC (path dependence).
- **DGM / PINNs:** mesh-free PDE solvers trained on residual + terminal/boundary losses; accuracy depends on sampling/weights and tends to be slower to high precision.
- **BS-localized DGM:** learn a correction U to a BS anchor (control-variate). More stable; in our runs achieves $\sim 10^{-2}$–10^{-1} relative error despite the large epoch number/training time; and still behind Fourier/COS for 2D Heston Europeans.

Takeaway: For European options under Heston, *Fourier/COS* remain best in accuracy/speed. *DGM/PINNs* are promising when labels are scarce or in harder settings (higher dimensions, early exercise, constraints).

The End

References I

- Black, Fischer, and Myron Scholes, "The Pricing of Options and Corporate Liabilities". *Journal of Political Economy*, vol. 81, no. 3, 1973, pp. 637–54.
- Carr, Peter, and Dilip Madan, "Option Valuation Using the Fast Fourier Transform". *Journal of Computational Finance*, vol. 2, no. 4, 1999, pp. 61–73.
- Cont, Rama, "Empirical Properties of Asset Returns: Stylized Facts and Statistical Issues". *Quantitative Finance*, vol. 1, no. 2, 2001, pp. 223–36.
- Fang, Fang, and Cornelis W. Oosterlee, "A Novel Pricing Method for European Options Based on Fourier–Cosine Series Expansions". *SIAM Journal on Scientific Computing*, vol. 31, no. 2, 2008, pp. 826–48.
- Gatheral, Jim, *The Volatility Surface: A Practitioner's Guide*. Wiley, 2006.
- Heston, Steven L., "A Closed-Form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options". *Review of Financial Studies*, vol. 6, no. 2, 1993, pp. 327–43.

References II

- Hull, John C., *Options, Futures, and Other Derivatives*. 10th ed., Pearson, 2018.
- Luo, Kuang, et al., "Physics-informed neural networks for PDE problems: a comprehensive review". *Artificial Intelligence Review*, vol. 58, no. 10, 2025, pp. 1–43.
- Merton, Robert C., "Theory of Rational Option Pricing". *Bell Journal of Economics and Management Science*, vol. 4, no. 1, 1973, pp. 141–83.
- Sirignano, Justin, and Konstantinos Spiliopoulos, "DGM: A Deep Learning Algorithm for Solving Partial Differential Equations". *Journal of Computational Physics*, vol. 375, 2018, pp. 1339–64.

Heston PDE: extra notes

- Degeneracy at $v=0$: second-order terms vanish; use the reduced first-order boundary PDE.
- Large v: truncate $v \in [0, v_{\max}]$; mild growth or $V_v=0$ as a practical BC.
- Well-posedness (Feynman–Kac): coefficients locally Lipschitz with linear growth; payoff $\lesssim$ linear in S.
- Solution regularity: classical if $V \in C^{1,2}$; else weak/viscosity solutions for degenerate parabolic case.

Extra notes on parameters:

- Feller condition (variance stays strictly positive): $2\kappa\theta > \sigma^2$ (sufficient).
- Correlation $\rho < 0$ reproduces equity skews.
- Affine structure enables fast Fourier pricing via characteristic functions.

Extra: Boundary Conditions (common choices)

- As $S \to 0$: for calls, $V \to 0$.
- As $S \to \infty$: for calls, $V \sim S - Ke^{-r\tau}$ (linear growth).
- At $v \to 0$: degenerates to BS with $\sigma = 0$; enforce smoothness.
- At large v: growth control (e.g., penalty or truncation domain for v).

From supervised to DGM loss

Key differences DGM/PINNs vs supervised methods:

- *What is data?* Supervised uses labeled pairs (x_i, y_i); DGM uses *collocation points* and physics (no labels).

From supervised to physics-informed: DGM loss

Key differences DGM/PINNs vs supervised methods:

- *What is data?* Supervised uses labeled pairs (x_i, y_i); DGM uses *collocation points* and physics (no labels).
- *Loss target:* Supervised matches observed prices; DGM drives PDE residual $\to 0$ and satisfies terminal/boundary conditions.

From supervised to physics-informed: DGM loss

Key differences DGM/PINNs vs supervised methods:

- *What is data?* Supervised uses labeled pairs (x_i, y_i); DGM uses *collocation points* and physics (no labels).
- *Loss target:* Supervised matches observed prices; DGM drives PDE residual $\to 0$ and satisfies terminal/boundary conditions.
- *Sampling:* Supervised: sample given dataset; DGM: sample interior Ω, terminal slice $t = T$, and spatial boundaries $\partial\Omega$.

From supervised to physics-informed: DGM loss

Key differences DGM/PINNs vs supervised methods:

- *What is data?* Supervised uses labeled pairs (x_i, y_i); DGM uses *collocation points* and physics (no labels).
- *Loss target:* Supervised matches observed prices; DGM drives PDE residual $\to 0$ and satisfies terminal/boundary conditions.
- *Sampling:* Supervised: sample given dataset; DGM: sample interior Ω, terminal slice $t{=}T$, and spatial boundaries $\partial\Omega$.
- *When to use:* Supervised if you trust labels (Fourier/COS pricer or market quotes). DGM if labels are scarce/expensive or for high-dim PDEs.

From supervised to physics-informed: DGM loss

Key differences DGM/PINNs vs supervised methods:

- *What is data?* Supervised uses labeled pairs (x_i, y_i); DGM uses *collocation points* and physics (no labels).
- *Loss target:* Supervised matches observed prices; DGM drives PDE residual $\to 0$ and satisfies terminal/boundary conditions.
- *Sampling:* Supervised: sample given dataset; DGM: sample interior Ω, terminal slice $t{=}T$, and spatial boundaries $\partial\Omega$.
- *When to use:* Supervised if you trust labels (Fourier/COS pricer or market quotes). DGM if labels are scarce/expensive or for high-dim PDEs.

Training: Gradients computed by automatic differentiation for $\mathcal{N}[V_\theta]$ within the training loop; optimize θ with SGD/Adam. Choose $\lambda_{\mathrm{term}}, \lambda_{\mathrm{bdry}}$ to balance residual vs constraints (often start larger for terminal).