Gian Franco Lancioni - Análisis Cuantitativo en Finanzas 2022

Neural Options Pricing

(paper de Timothy DeLise 2021)

Contexto

- Actualidad, 2021
  - Desarrollo importante reciente de modelos generativos adversariales GANs (Arjovsky et. al [2017])
    - Entrenar una red generativa para que 'engañe' a una discriminativa entrenada en paralelo.
  - Redes neuronales como aproximaciones de procesos continuos, primero con ODEs hasta cubrir recientemente el terreno de las SDEs (Li et al [2020], Kidger et al [2020])
    - Herramientas prácticas incorporables al stack moderno de desarrollo en RNs (torchsde, Google Research).
  - A pesar de esto, ninguno de los papers de SDEs neuronales toca el tema pricing de opciones y fuera del nuevo enfoque prevalecen modelos supervisados (Liu et al., 2019, Broström and Kristiansson, 2018) frente a no supervisados.
    - **No vamos a entrenar predicciones de precios de opciones en función de datos de entrada.**

- Histórico
  - Mucho legado de modelos analíticos con asunciones fuertes.
  - Black and Scholes [1973].

Aprendiendo SDEs

Intentaremos aproximar el proceso target de un activo subyacente desconocido que asumimos sigue una SDE

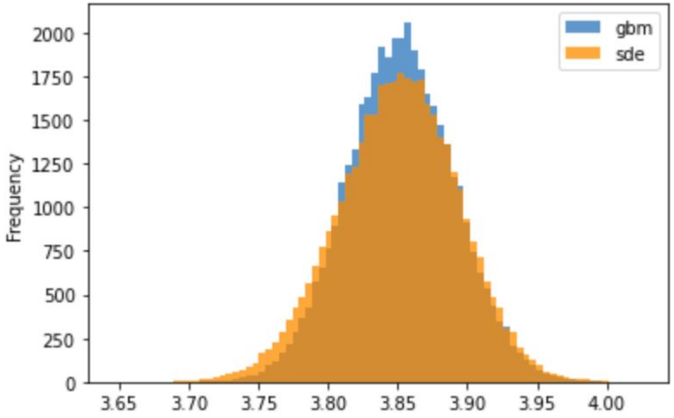
$$dR_t = \mu(R_t, t)\, dt + \sigma(R_t, t)\, dB_t$$

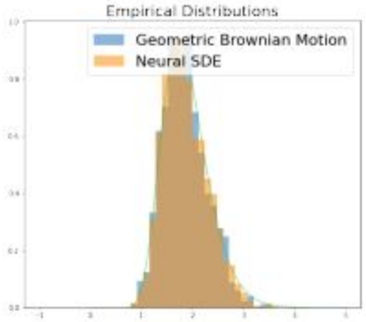
Por una SDE donde los procesos de drift y la difusión son RNs parametrizadas por pesos θ y Φ respectivamente

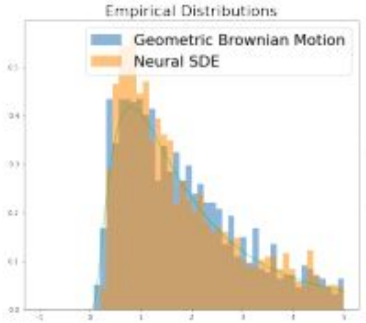
$$dS_t = \underbrace{\mu_\theta(t, S_t)}_{\text{drift}} dt + \underbrace{\sigma_\phi(t, S_t)}_{\text{difusión}} dB_t$$

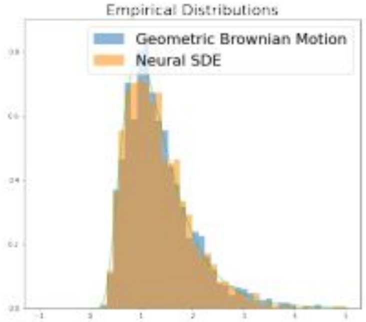
Aprendiendo SDEs

El criterio es que las distribuciones empíricas por batch se parezcan lo suficiente bajo alguna métrica como para poder afirmar que el precio (pasando a medidas libres de riesgo).





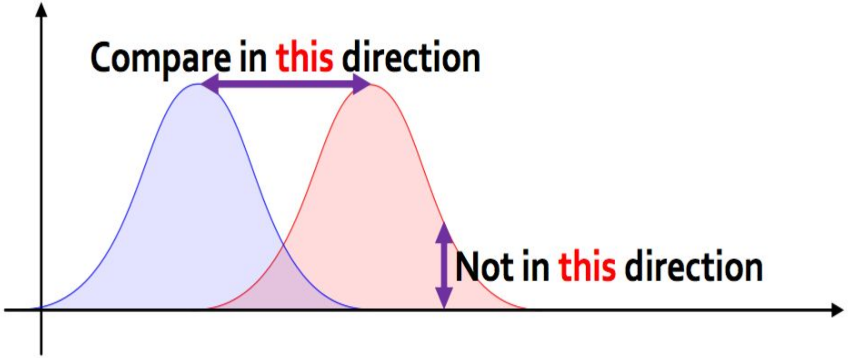




Wasserstein - Earth Mover's Distance

Ramdas et al. [2015]

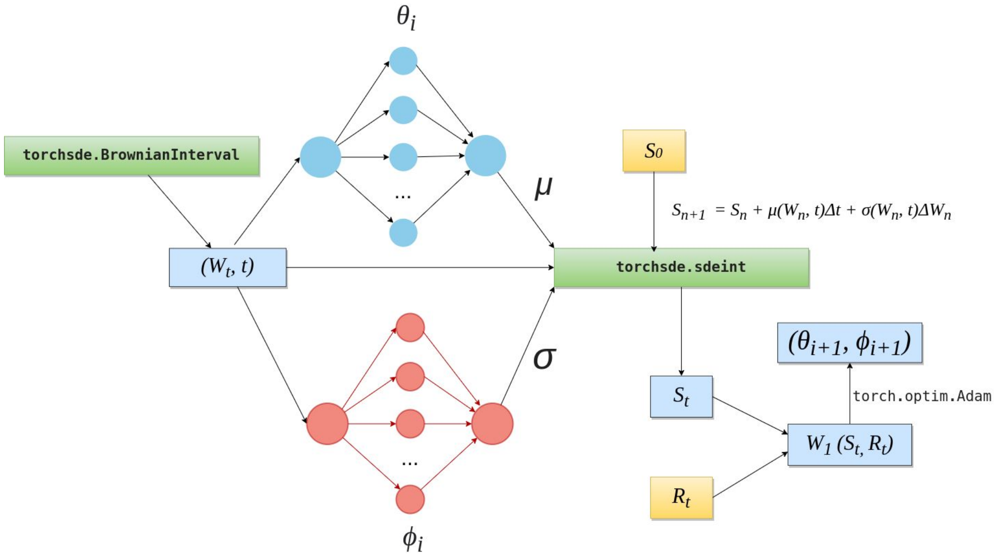
$$W_p(\mu, \nu) = \left(\int_0^1 \left| F^{-1}(z) - G^{-1}(z) \right|^p dz \right)^{1/p}$$



- Origen en teoría de transporte óptimo.
- Varias formulaciones y maneras de computar.
- En particular consideramos caso unidimensional y con p=1.
- Idealmente $W(R_t, S_t) \to 0$

Esquema por batch

- Generamos para un rango [0,1] una trayectoria discreta de movimiento browniano.
- Esta trayectoria se alimenta a las RNs de difusión y drift que, mediante un solver, van a generar trazas de la SDE para cada instante del anterior intervalo sobre todo el batch según sus pesos actuales.
- Una loss function va a evaluar la cercanía instante a instante del batch generado contra el proceso objetivo.
- La optimización de esta loss genera los pesos para las redes en la próxima iteración.



Pricing

Approach clásico pasando a medida de riesgo neutral. Usan **r=0**.

Dinámica de la SDE

$$S_T = S_t + \int_t^T \mu(s,\omega)ds + \int_t^T \sigma(s,\omega)dB_s$$

Función de precios de la call europea

$$f(S_T) = \max(S_T - K, 0)$$

Transformación del browniano por Girsanov

$$\tilde{B}_t = B_t + \int_0^t \frac{\mu(s,\omega)}{\sigma(s,\omega)}ds$$

$$\begin{aligned}
V_t &= \mathbb{E}_Q[f(S_T)] \\
&= \mathbb{E}_Q\left[f\left(S_t + \int_t^T \mu(s,\omega)ds + \int_t^T \sigma(s,\omega)dB_s\right)\right] \\
&= \mathbb{E}_Q\left[f\left(S_t + \int_t^T \sigma(s,\omega)\left(dB_s + \frac{\mu(s,\omega)}{\sigma(s,\omega)}ds\right)\right)\right] \\
&= \mathbb{E}\left[f\left(S_t + \int_t^T \sigma(s,\omega)dB_s\right)\right]
\end{aligned}$$

Pricing

Conjetura:

Para dos procesos de Ito R y S con medidas martingala W y V respectivamente, sea f una función Lipschitz-1 vale:

$$\left|\mathbb{E}_W[f(S_T)] - \mathbb{E}_V[f(R_T)]\right| \leq \left|\mathbb{E}[f(S_T)] - \mathbb{E}[f(R_T)]\right|$$

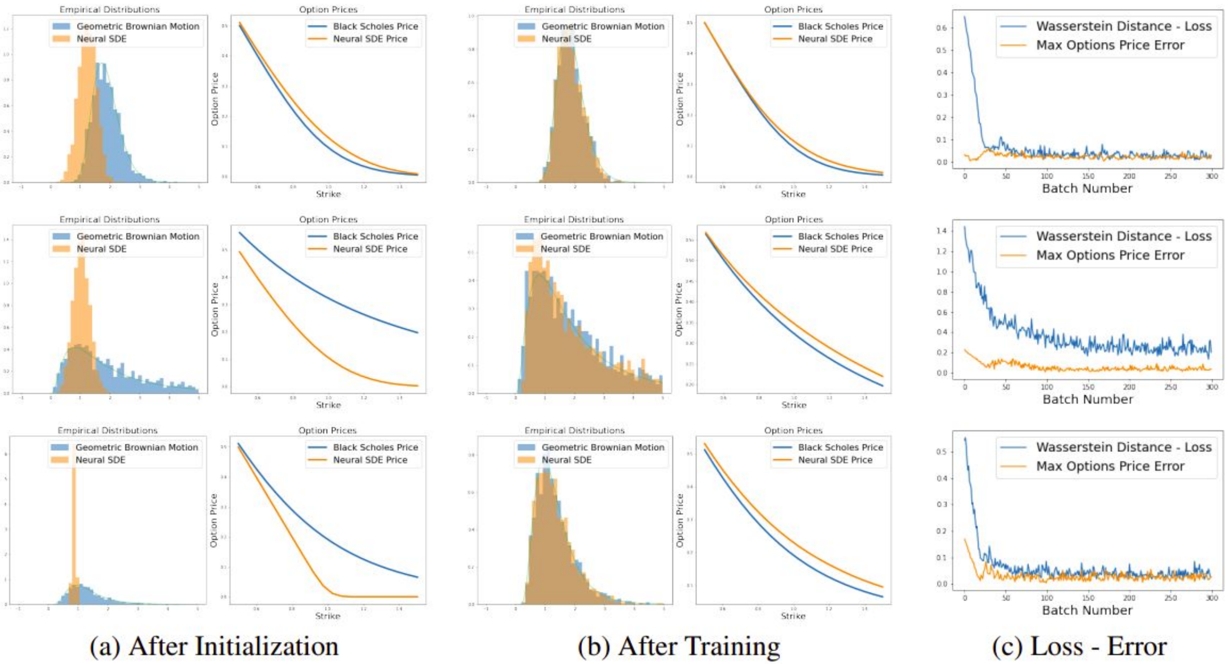
Además, por formulación dual de la distancia de Wasserstein con d=1, p=1, también vale

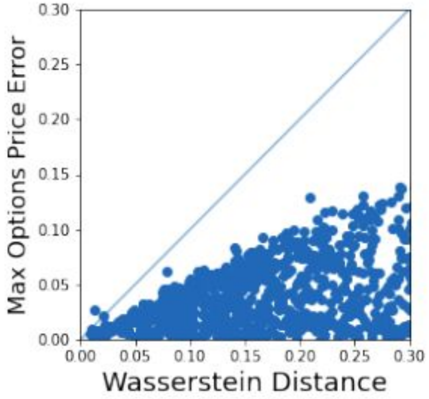
$$\mathbb{E}_{x\sim\mu}[f(x)] - \mathbb{E}_{x\sim\nu}[f(x)] \leq W_1(\mu,\nu)$$

Siendo la función de precio de opciones call europeas una función Lipschitz-1 finalmente esto nos da una cota superior en la distancia de Wasserstein al error en pricing de nuestro modelo.

Esto significa que si el modelo generativo converge al proceso target entonces también lo hará nuestra estimación de precios.

Resultados


Empirical Distributions
Geometric Brownian Motion
Neural SDE
Option Prices
Black Scholes Price
Neural SDE Price
Option Price
Strike
Wasserstein Distance - Loss
Max Options Price Error
Batch Number
(a) After Initialization
(b) After Training
(c) Loss - Error


Max Options Price Error
Wasserstein Distance
0.00
0.05
0.10
0.15
0.20
0.25
0.30

Discusión

- Como dijimos antes, primer approach a pricing usando SDEs neuronales como herramientas.
- Enfoque no supervisado.
- ¿Y la GAN?
  - https://blog.dataiku.com/using-deep-learning-for-better-option-pricing
  - No hace falta modelar la SDE para aprender a extender el *sample of 1* de los activos con modelos generativos.
  - “Simply fitting SDE parameters to maximize likelihood will in general cause overfitting, and will result in the diffusion function going to zero.” Li et al [2020]
  - Kridger et al. la implementaron posteriormente, no es trivial: formulan a la red discriminativa como una CDE (Control DIfferential Equation).