RMSC5102 Midterm Review

Leung Man Fung, Heman

Spring, 2020

Agenda

- Review
  - Stochastic Calculus
  - The Black-Scholes World
  - Monte Carlo Method
  - Random Variable Generation
  - Variance Reduction Technique
- Q&A

Stochastic Calculus

Wiener Process

- Stationary increment: $\mathrm{W_t} - W_s \sim N(0, t - s)$
- Independent increment: $W_{t_4} - W_{t_3} \perp W_{t_2} - W_{t_1}$
- **Starts at zero:** $\boldsymbol{P\left(W_{t_0} = 0\right) = 1}$

Finding SDE

- Strategy
  - Define $f(x, t)$ and dX_t
  - Apply Ito's lemma to $f(X_t, t)$
- Straight forward
- Example: HW1 1a, 4b; Exercise 2.2

Finding Stochastic Integral

- Strategy
  - Guess the function such that it will contain the integrand in its SDE
  - Use Ito's lemma to find the SDE of our guess
  - Rearrangement the terms and integrate both sides
- Indirect
- Example: HW1 3a, 4a, 4c; Exercise 3.2
- Note (reference: HW1 4a)
  - $W_0 = 0$ but it is possible that $f(W_0, 0) \neq 0$
  - Stochastic integral may not be further reducible

By Ito's lemma, $d(\frac{1}{2}e^{2W_t}) = e^{2W_t}dt + e^{2W_t}dW_t \Rightarrow e^{2W_t}dW_t = d(\frac{1}{2}e^{2W_t}) - e^{2W_t}dt$
Hence $\int_0^t e^{2W_s}dW_s = \frac{1}{2}e^{2W_t} - \frac{1}{2} - \int_0^t e^{2W_s}ds$

Geometric Brownian Motion

- SDE: $\mathrm{dS_t} = rS_t dt + \sigma S_t dW_t \Rightarrow \mathrm{S_t} = S_0 e^{\left(r-\frac{1}{2}\sigma^2\right)t+\sigma W_t}$
  - Use $S_T = S_0 e^{\left(r-\frac{1}{2}\sigma^2\right)T+\sigma\sqrt{T}Z}$ in simulation to avoid simulating intermediate prices
- Algorithm:
  - Generate $Z \sim N(0,1)$
  - Set $S_T = S_0 e^{\left(r-\frac{1}{2}\sigma^2\right)T+\sigma\sqrt{T}Z}$
- Example: HW1 1, 2; HW2 5
- Question may specify stock price dynamic other than GBM
  - Use the given dynamic to simulate S_T like generating random variables

The Black-Scholes World

Risk Neutral Valuation

- $V_t = e^{-r(T-t)} E[f(S_t, t)]$
- Take expectation w.r.t. real world probability?
  - E.g. with insider info you know price of a certain stock will likely go up
- Problem of the discount rate
  - If real world probability is used, discount rate has to accommodate the level of risk (think about the discount rate you use in DCF)
  - If risk neutral probability is used, discount rate = risk free rate (observable)
- Just give you another way of looking at risk neutral approach

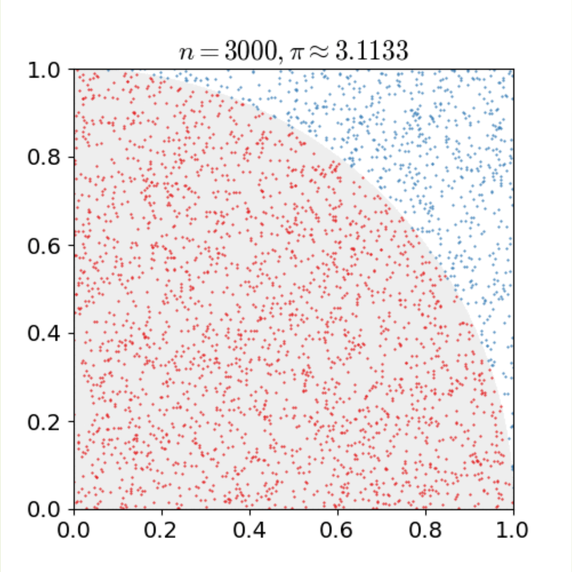
Black–Scholes–Merton Model

- Black-Scholes formula
  - $C(S_t, t) = \Phi(d_1)S_t - \Phi(d_2)Ke^{-r(T-t)}$,
  - $P(S_t, t) = Ke^{-r(T-t)} - \mathrm{S_t} + C(S_t, t) = \Phi(-d_2)Ke^{-r(T-t)} - \Phi(-d_1)S_t$
  - $d_1 = \frac{1}{\sigma\sqrt{T-t}}\left[\ln\left(\frac{S_t}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)\right]$
  - $d_2 = d_1 - \sigma\sqrt{T-t}$
- Note that $P(S_t, t)$ is derived from put-call parity
  - Put-call parity: $\mathrm{C_E} - P_E = S - Ke^{-r(T-t)}$
- Implied volatility
  - Value of volatility when back-solving an option pricing model (such as BS) with current market price

Monte Carlo Method

Key idea

- Use repeated random sampling to obtain numerical estimate
  - The estimate is usually average in our course
- Example: estimate π (picture credit: nicoguaro)



Standard Monte Carlo

- HW2 5a: price a European call option
  - Recall payoff function is $\max(S_T - K, 0)$
  - Estimate $E[\max(S_T - K, 0)]$ by sample average $\frac{1}{n}\sum_{i=1}^{n} \max(S_T^{(i)} - K, 0)$
- Algorithm
  - 1) Generate $Z \sim N(0,1)$
  - 2) Set $S_T = S_0 e^{\left(r - \frac{1}{2}\sigma^2\right)T + \sigma\sqrt{T}Z}$
  - **3) Compute $\boldsymbol{\pi_i = \max(S_T - K, 0)}$**
  - 4) Repeat 1 to 3 for $i = 1, \ldots, n$
  - 5) Option price = $\frac{\mathrm{e}^{-\mathrm{rT}}}{\mathrm{n}} \sum_{i=1}^{n} \pi_i$

Random Variable Generation

Assumption

- We can only generate $U(0,1)$ and $N(0,1)$ random variable
  - Any r.v. with other distribution cannot be generated directly (in algorithm)
  - If you write R code instead, an advantage will be given. You may use the native function like
    - sample() for discrete r.v.
    - rexp() for exponential r.v. etc.

Inverse Transform

- If we know $X \sim F_X$ (i.e. the cdf), we can generate X out of $U \sim U(0,1)$
  - The supporting theory is probability integral transform
- Algorithm (discrete)
  - Generate $U \sim U(0,1)$
  - $X = x_j$ if $\sum_{i=0}^{j-1} p_i \leq U < \sum_{i=0}^{j} p_i$
- Algorithm (continuous)
  - Generate $U \sim U(0,1)$
  - $X = F_X^{-1}(U)$ assuming the inverse exists
- Example: HW2 1, 4a

Rejection Sampling

- If we can simulate $Y \sim G_Y$ easily, we can use the proportional distribution as a basis to simulate X with pdf $f(x)$
- Algorithm
  - 1) Find $c = \max_y \frac{f(y)}{g(y)}$
  - 2) Generate Y_i from a density g: $U_1 \sim U(0,1) \Rightarrow Y_i = G^{-1}(U_1)$
  - 3) Generate $U_2 = U(0,1)$
  - 4) If $\mathrm{U}_2 \leq \frac{1}{c} \cdot \frac{f(Y_i)}{g(Y_i)}$, set $X_i = Y_i$, otherwise return to 2
- Example: HW2 3, 4b

Variance Reduction Technique

Antithetic Variables

- If we are able to generate negatively correlated underlying random variables, the estimator can have lower variance as compared with independent samples
  - This requires the target function $h(x)$ to be monotone
  - Show $h'(x) \geq 0$ or $h'(x) \leq 0$ within the target range for monotonicity
  - As $h(x)$ is monotone, $Cov\big(h(U), h(1-U)\big) \leq 0$ where $U \sim U(0,1)$
  - As half of your variables are antithetic, you only need to generate $\frac{n}{2}$ numbers for n samples
- Example: HW2 3, 4b

Antithetic Variables

- Algorithm:
  - 1) Generate $U \sim U(0,1)$
  - 2) Set $X_i = F^{-1}(U), Y_i = F^{-1}(1-U)$ (note: want X, Y same distribution but negative correlation)
  - 3) Repeat 1 and 2 for n times
  - 4) $\hat{\theta} = \frac{1}{2n}\sum_{i=1}^{n}[h(X_i) + h(Y_i)]$
- Note:
  - $F^{-1}(U)$ is monotone in general as cdf is monotone
  - Hence $h[F^{-1}(U)]$ is monotone if $h(\cdot)$ is monotone

Stratified Sampling

- If we have information about grouping in the population, then we may use conditional mean (mean of subgroup) as the sample from the population
- Algorithm:
  - Generate $V_{i,j} = \frac{1}{B}(U_{i,j} + i - 1)$ where $U_{i,j} \sim U(0,1)$ for $i = 1, \ldots, B; j = 1, \ldots, N_B$
  - Set $X_{i,j} = F^{-1}(V_{i,j})$
  - $\hat{\theta} = \frac{1}{B \times N_B} \sum_{j=1}^{N_B} \left[h(X_{1,j}) + \mathrm{h}(X_{2,j}) + \cdots + \mathrm{h}(X_{B,j}) \right]$ (remember to adjust for conditional probability)
- Example: currently none in HW so I will provide one on next page

Stratified Sampling

Exercise 1.1. Consider $\theta = \int_2^\infty (x-2)e^{-x}dx$. It is known that $\theta = E[f(X)]$ where $X \sim exp(1)$.
(a)What is $f(X)$?
(b)Provide an algorithm to sample X in $[2, \infty]$.
(c) Provide an algorithm and VBA programme to simulate θ by stratifying X on the interval $[2, \infty]$ with equal probability 1/4 for each stratified interval. The total sample should be 10000.

(a)

$$\theta = \int_2^\infty (x-2)e^{-x}dx = \int_0^\infty (x-2)\mathbb{I}(x \geq 2)e^{-x}dx$$

$$\therefore f(X) = (X-2)\mathbb{I}(X \geq 2)$$

(b)
Let $Y = X - 2|X \geq 2$. By memoryless property of exponential distribution, $Y \sim exp(1)$.
Therefore $X|X \geq 2$ can be sampled by $-\ln(U) + 2$, where $U \sim Unif(0,1)$.

(c)
Note that $E[(X-2)\mathbb{I}(X \geq 2)] = E[(X-2)|X \geq 2]\, P(X \geq 2)$.

1. Generate $U_j \sim Unif(0,1)$
2. Set $V_{ij} = -\ln\left[\frac{U_j+i}{4}\right] + 2$, for $i = 0,1,2,3$ and $j = 1,2,...,2500$.
3. Compute $Y_{ij} = V_{ij} - 2$.
4. Repeat step 1 to 3 for 2500 times for each $i = 0,1,2,3$.
5. $\theta_{stra} = \frac{1}{10000}\sum_{j=1}^{2500}(Y_{0j} + Y_{1j} + Y_{2j} + Y_{3j}) \times e^{-2}$.


IF YOURE GOING THROUGH HELL
KEEP GOING
-Winston Churchill
Karyera.c

Q&A