Cat\_2

Joy Kendi

2025-09-10

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.4 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.4   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

S0 <- 100  
K <- 100  
r <- 0.05  
sigma <- 0.20  
Time\_T <- 0.5  
tau <- Time\_T  
set.seed(1)  
  
BS\_Call <- function(S0, K, r, sigma, tau) {  
 if (sigma <= 0 || tau <= 0) return(pmax(exp(-r \* tau) \* (S0 - K), 0))  
   
 d1 <- (log(S0 / K) + (r + 0.5 \* sigma^2) \* tau) / (sigma \* sqrt(tau))  
 d2 <- d1 - sigma \* sqrt(tau)  
   
 S0 \* pnorm(d1) - K \* exp(-r \* tau) \* pnorm(d2)  
}  
  
BS\_Price <- BS\_Call(S0, K, r, sigma, tau)  
  
MC\_Call <- function(M, S0, K, r, sigma, tau, antithetic = TRUE) {  
 if (antithetic) {  
 m2 <- ceiling(M / 2)  
 z <- rnorm(m2)  
 z <- c(z, -z)[1:M]  
 } else {  
 z <- rnorm(M)  
 }  
 ST <- S0 \* exp((r - 0.5 \* sigma^2) \* tau + sigma \* sqrt(tau) \* z)  
 payoff <- exp(-r \* tau) \* pmax(ST - K, 0)  
 c(MC\_Price = mean(payoff), SE = sd(payoff) / sqrt(M))  
}  
  
M <- 100000  
Payout <- MC\_Call(M, S0, K, r, sigma, tau, antithetic = TRUE)  
CI\_Payout <- Payout["MC\_Price"] + c(-1, 1) \* 1.96 \* Payout["SE"]  
  
cat(sprintf(  
 "MC: price = %.6f, SE = %.6f, 95%% CI = [%.6f, %.6f]\n",  
 Payout["MC\_Price"], Payout["SE"], CI\_Payout[1], CI\_Payout[2]  
))

## MC: price = 6.926026, SE = 0.031130, 95% CI = [6.865011, 6.987040]

cat(sprintf("Black–Scholes: price = %.6f\n\n", BS\_Price))

## Black–Scholes: price = 6.888729

Ms <- round(exp(seq(log(500), log(2e5), length.out = 8)))  
result <- map\_dfr(Ms, function(m) {  
 out <- MC\_Call(m, S0, K, r, sigma, tau, antithetic = TRUE)  
 tibble(  
 M = m,  
 MC\_Price = out["MC\_Price"],  
 SE = out["SE"],  
 CI\_Low = MC\_Price - 1.96 \* SE,  
 CI\_High = MC\_Price + 1.96 \* SE  
 )  
})  
  
ggplot(result, aes(x = M, y = MC\_Price)) +  
 geom\_hline(yintercept = BS\_Price, linetype = 2, color = "steelblue") +  
 geom\_point() +  
 geom\_errorbar(aes(ymin = CI\_Low, ymax = CI\_High), width = 0) +  
 scale\_x\_log10() +  
 labs(  
 title = "Monte Carlo convergence vs M",  
 subtitle = sprintf("S0=%g, K=%g, r=%.2f%%, sigma=%.2f%%, T=%.2f (BS=%.4f)",  
 S0, K, 100\*r, 100\*sigma, tau, BS\_Price),  
 x = "Number of simulations M (log scale)",  
 y = "Price (95% CI)"  
 ) +  
 theme\_minimal()

