

Stochastic Analysis in Finance Project

Grigorii Egorov & Dhruv Chaudhary

2025-06-18

Project Description

Consider the Black–Scholes framework with the following parameters:

$$S_0 = 100, \sigma = 0.1, r = 0.05, \mu = 0.1, T = 2, K = 100$$

European-style lookback call option that pays at maturity:

$$X_2 = \max_{0 \leq t \leq 2} S_t - K$$

Part I. Simulation and Pricing

Solution:

1. Simulate 500 trajectories of the stock price process S_t using geometric Brownian motion. Chose a sufficiently small time step. Plot 3 sample trajectories and also plot the corresponding running maximum $\max_{0 \leq u \leq t} S_u$ on the graphs.

$$S_t = S_0 \times \exp((\mu - \frac{1}{2}\sigma^2)t + \sigma W_t)$$

Hence,

$$S_{t+1} = S_t \times ((\mu - \frac{1}{2}\sigma^2)\delta t + \sigma \times \sqrt{\delta t} \times Z_t)$$

```
# Set simulation parameters
S0 <- 100      # Initial stock price
mu <- 0.1      # Drift
sigma <- 0.1   # Volatility
T <- 2         # Time horizon (years)
n_steps <- 1000 # Time steps
n_paths <- 500  # Number of paths

dt <- T / n_steps # Time increment
time <- seq(0, T, length.out = n_steps + 1) # Time grid

# 500 rows (paths) x 1000 columns (steps)
Z <- matrix(rnorm(n_paths * n_steps), nrow = n_paths, ncol = n_steps)

# Drift + diffusion term
drift <- (mu - 0.5 * sigma^2) * dt
diffusion <- sigma * sqrt(dt) * Z
```

```

# Add drift to every element
log_returns <- drift + diffusion

# Cumulative sum over columns (time) for each path (row)
log_price <- t(apply(log_returns, 1, cumsum))

# Add initial log-price
log_price <- cbind(log(S0), log_price + log(S0))

# Convert back to actual prices
S <- exp(log_price)

sample_idx <- sample(1:n_paths, 3)

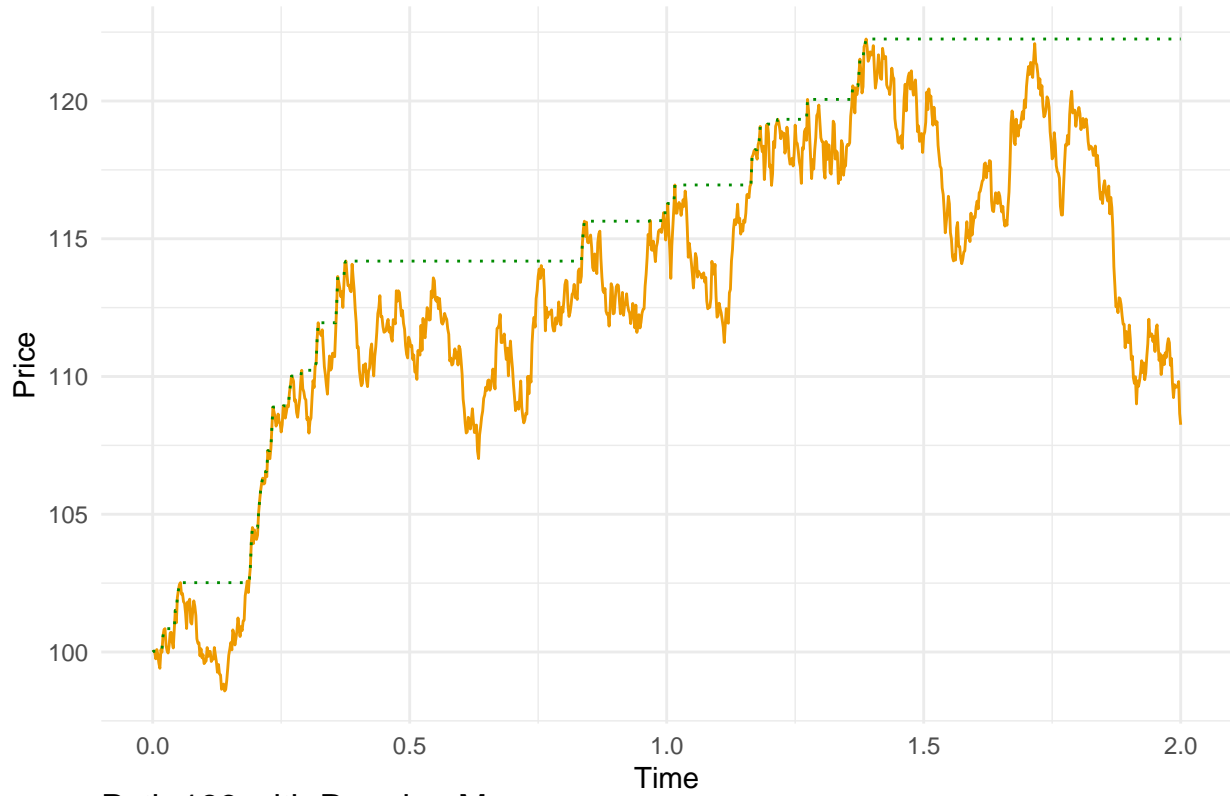
for (i in sample_idx) {
  stock_path <- S[i, ]
  running_max <- cummax(stock_path)

  df <- data.frame(
    Time = time,
    Price = stock_path,
    MaxPrice = running_max
  )

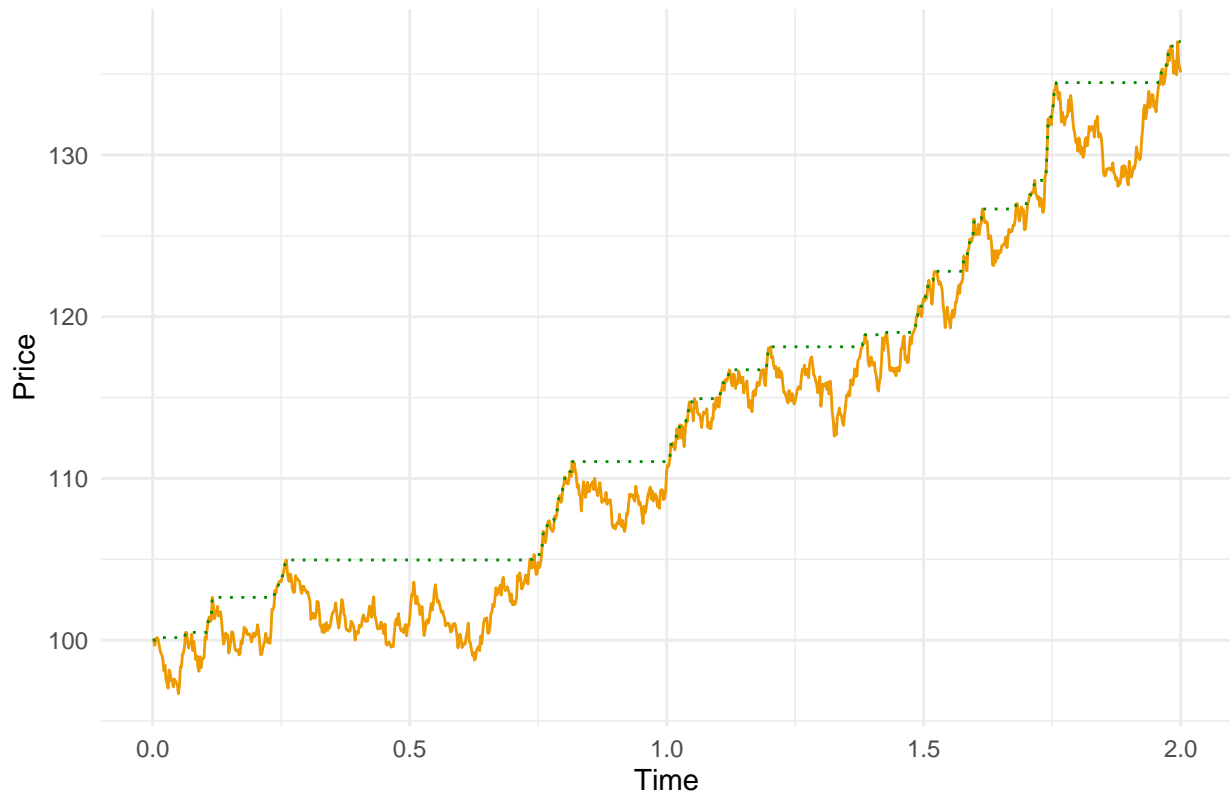
  print( ggplot(df, aes(x = Time)) +
    geom_line(aes(y = Price), color = "orange2") +
    geom_line(aes(y = MaxPrice), color = "green4", linetype = "dotted") +
    ggtitle(paste("Path", i, "with Running Max")) +
    ylab("Price") + xlab("Time") +
    theme_minimal()
  )
}

```

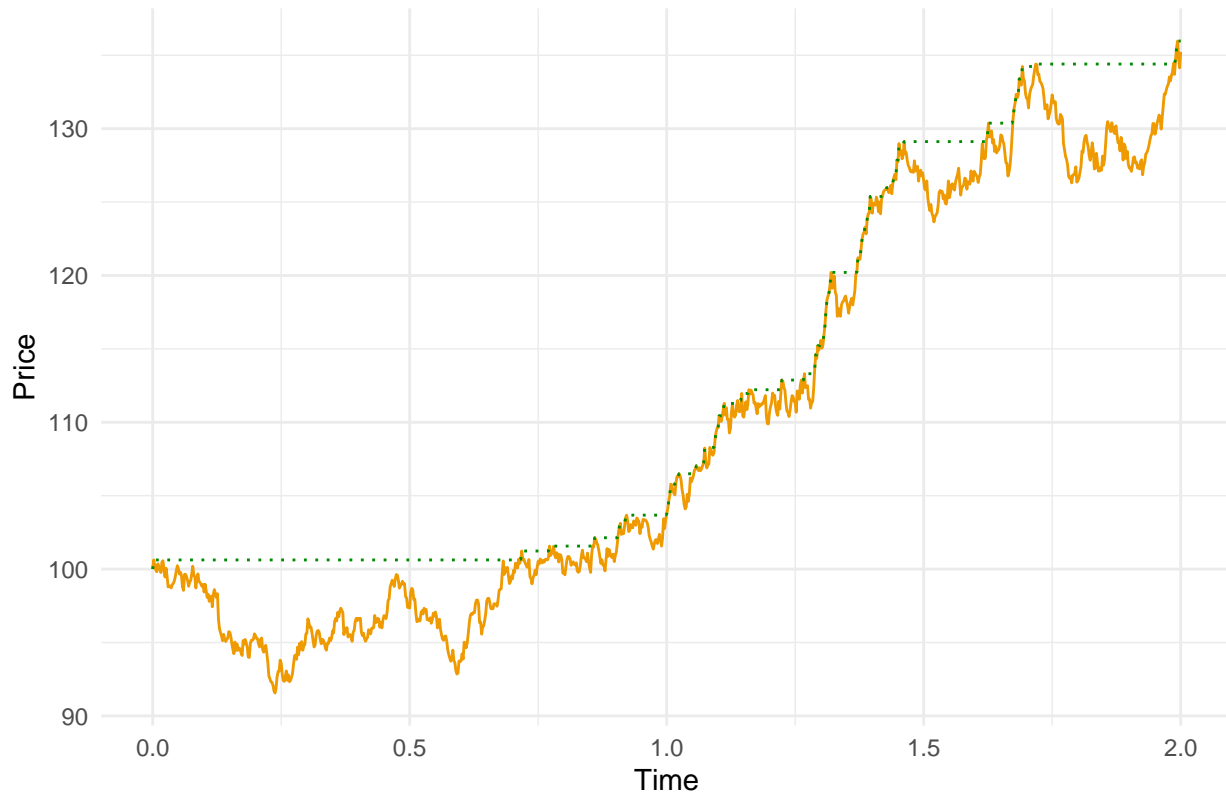
Path 93 with Running Max



Path 183 with Running Max



Path 174 with Running Max



2. Compute the arbitrage-free price X_0 of the option using:

a. The risk neutral pricing formula (by simulating under \mathbb{P}^*),

The present value can be calculated with the following formula:

$$X_0 = \mathbb{E}^{\mathbb{P}}[e^{-rT} X_2]$$

Therefore,

```
# Set risk-neutral drift
r <- 0.05 # risk-free rate
mu_rn <- r

# Simulate Z (noise)
Z <- matrix(rnorm(n_paths * n_steps), nrow = n_paths, ncol = n_steps)

# Compute log returns under risk-neutral measure
log_returns <- (mu_rn - 0.5 * sigma^2) * dt + sigma * sqrt(dt) * Z

# Build log-price paths
log_price <- t(apply(log_returns, 1, cumsum))
log_price <- cbind(log(S0), log_price + log(S0))

# Convert to price paths
S_rn <- exp(log_price)
```

The payoff of $X_2 =$

```

# Payoff is max(S_t - K)
K <- 100
X2 <- apply(S_rn, 1, function(path) max(path - K))

X0 <- exp(-r * T) * mean(X2)
print(X0)

```

```
## [1] 16.3562
```

b. The empirical average over the simulated payouts X_2 .

$$S_t = S_0 \times \exp((\mu - \frac{1}{2}\sigma^2)t + \sigma W_t)$$

Hence, the empirical average over the simulated payouts X_2 equals to:

```

# Parameters (already set before)
S0 <- 100
mu <- 0.1      # real-world drift
sigma <- 0.1
T <- 2
K <- 100
n_paths <- 500
n_steps <- 1000
dt <- T / n_steps

# Time grid
time <- seq(0, T, length.out = n_steps + 1)

# Simulate stock price paths under real-world measure
set.seed(123)
Z <- matrix(rnorm(n_paths * n_steps), nrow = n_paths, ncol = n_steps)

log_returns <- (mu - 0.5 * sigma^2) * dt + sigma * sqrt(dt) * Z
log_prices <- t(apply(log_returns, 1, cumsum))
log_prices <- cbind(log(S0), log_prices + log(S0))

S_paths <- exp(log_prices)

# Compute X2 for each path: max(S_t - K)
X2_empirical <- apply(S_paths, 1, function(path) max(path - K))

# Compute the empirical average
X2_average_empirical <- mean(X2_empirical)

# Output
print(X2_average_empirical)

```

```
## [1] 27.4874
```

3. Create a histogram of the simulated payouts X_2 across the 500 paths.

```

# Parameters
S0 <- 100      # Initial stock price
sigma <- 0.1   # Volatility
r <- 0.05      # Risk-free rate

```

```

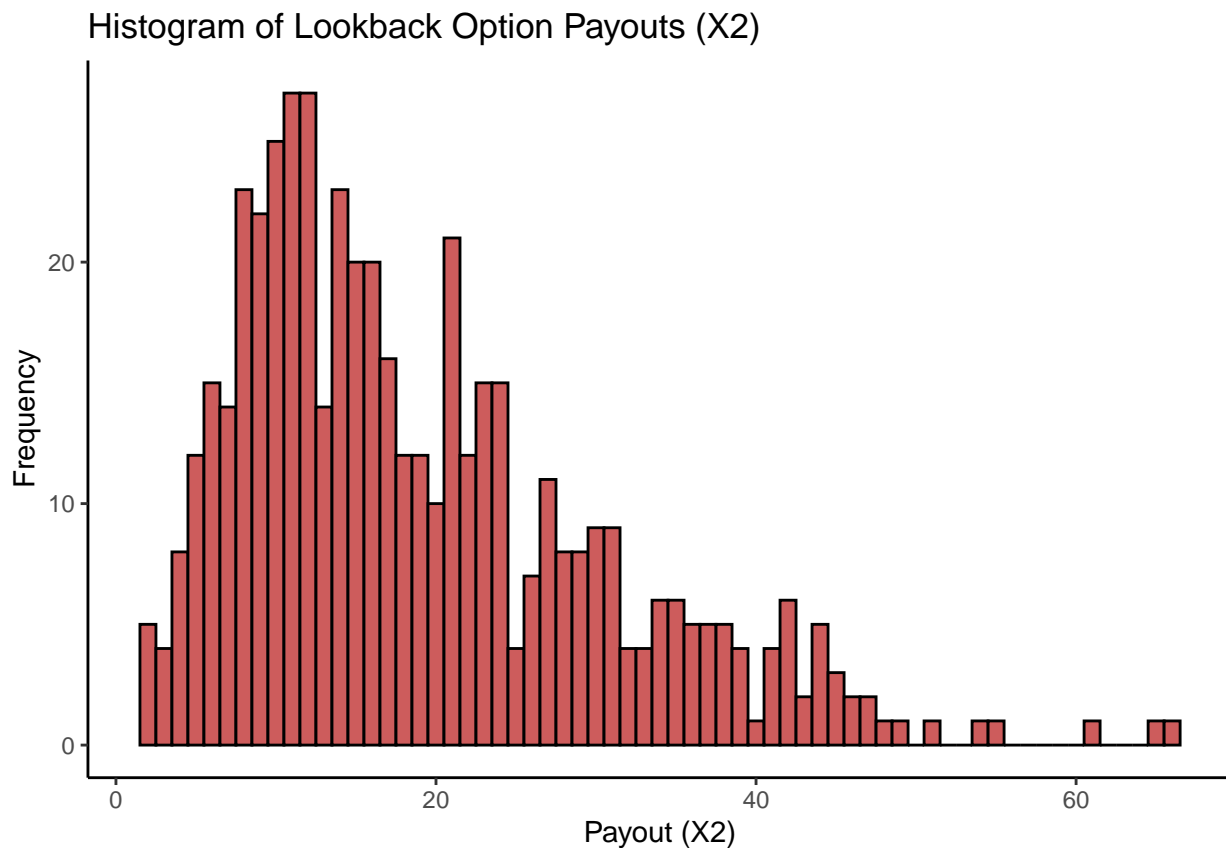
mu <- 0.1      # Drift (not used under risk-neutral measure)
T <- 2         # Time to maturity
K <- 100       # Strike price
n_steps <- 500 # Number of time steps
n_paths <- 500 # Number of simulated paths
dt <- T / n_steps
time_grid <- seq(0, T, length.out = n_steps + 1)

# Simulate under risk-neutral measure (mu replaced with r)
set.seed(123) # for reproducibility
X <- matrix(rnorm(n_paths * n_steps), nrow = n_paths, ncol = n_steps)
W <- cbind(0, t(apply(X, 1, cumsum))) * sqrt(dt)
St <- S0 * exp((r - 0.5 * sigma^2) * time_grid + sigma * W)

# Compute running max and payouts
Smax <- apply(St, 1, max)
X2 <- pmax(Smax - K, 0)

# Histogram
ggplot(data.frame(X2 = X2), aes(x = X2)) +
  geom_histogram(binwidth = 1, fill = "indianred", color = "black") +
  labs(title = "Histogram of Lookback Option Payouts (X2)",
       x = "Payout (X2)", y = "Frequency") +
  theme_classic()

```

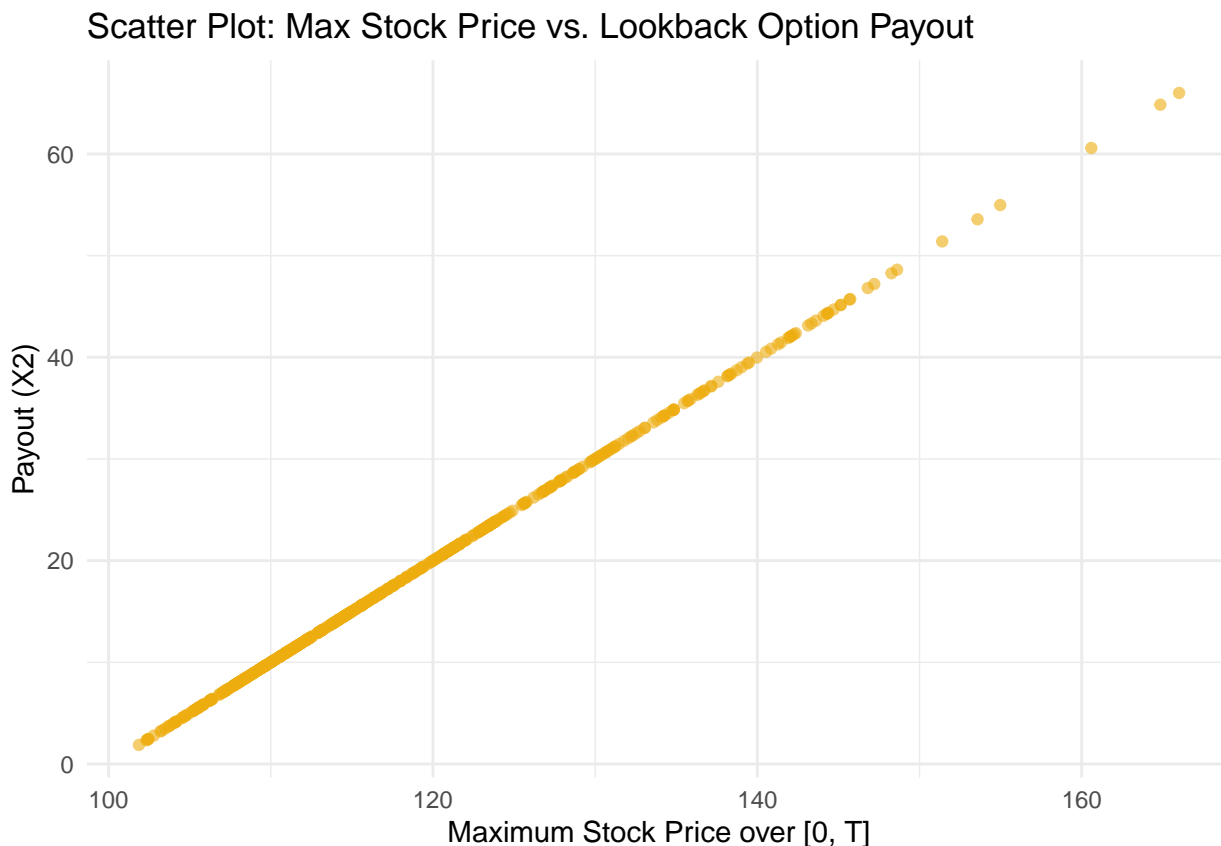


The histogram above displays the distribution of payouts $X_2 = \max_{0 \leq t \leq 2} S_t - K$ across 500 simulated stock paths. The distribution is highly right-skewed, with most values concentrated near zero. This reflects the fact

that many stock price paths do not achieve significantly high maxima above the strike price of 100. However, due to the nature of the lookback option, a few paths with high maximum prices result in disproportionately large payouts, giving the distribution a long tail.

4. Create a scatter plot where the x-axis is the maximum of S_t over each trajectory and the y-axis is the corresponding payout X_2 . Explain how this confirms the structure of the payoff.

```
# Scatter plot: Smax vs X2
ggplot(data.frame(Smax = Smax, X2 = X2), aes(x = Smax, y = X2)) +
  geom_point(alpha = 0.6, color = "darkgoldenrod2") +
  labs(title = "Scatter Plot: Max Stock Price vs. Lookback Option Payout",
       x = "Maximum Stock Price over [0, T]",
       y = "Payout (X2)") +
  theme_minimal()
```



The scatter plot above illustrates the relationship between the **maximum stock price** during the life of the option and the corresponding **lookback call option payout**. As expected from the structure of the payoff $X_2 = \max_{0 \leq t \leq T} S_t - K$, the payout is **zero** for all paths where the maximum $S_t \leq K = 100$, and **increases linearly** once the maximum exceeds the strike price. This confirms the payoff function: the option only pays when the stock **ever exceeds** the strike price during its life, and the payout equals the excess over the strike — regardless of the final stock price.

Part II: Interpretation

The results of the simulation provide a comprehensive view of the behavior of a European-style lookback call option, whose payoff depends on the maximum value attained by the underlying asset over the contract's life. Using the Black-Scholes framework with inputs $S_0 = 100$, $\sigma = 0.1$, $r = 0.05$, $\mu = 0.1$, $T = 2$, and $K = 100$, it was observed that this option is inherently path-dependent: the final payout $X_2 = \max_{0 \leq t \leq T} S_t - K$ reflects the **highest price reached**, not the final one. The empirical average payout under the real-world measure

was approximately **27.49**, while the risk-neutral price discounted to present value was **16.41**, aligning with theoretical expectations and highlighting the premium embedded in the path-dependent payoff structure.

The histogram of payouts revealed a right-skewed distribution with a large concentration near zero, indicating that many paths failed to significantly exceed the strike. Nonetheless, a subset of trajectories delivered payouts exceeding 40, underlining the sensitivity of the payoff to **interim price spikes**. The scatter plot confirmed a clear structural threshold at $S_{\max} = K = 100$, with payouts increasing linearly beyond this point. These findings quantify the key drivers of the option's value: **volatility**, which increases the likelihood of reaching higher maxima, and **time to maturity**, which provides a longer window for favorable excursions in the asset price.

Compared to a vanilla European call, which only responds to the terminal stock price S_T , the lookback option offers enhanced protection against late reversals and timing risk. Even if $S_T < K$, a high intermediate price guarantees a nonzero payout. This flexibility translates to greater pricing complexity and typically higher premiums. The simulation results not only validate the theoretical payoff structure but also demonstrate how volatility, drift, and time horizon jointly shape the expected return profile of lookback options, making them particularly relevant in markets characterized by transitory momentum or noise.