IX. MONTE CARLO SIMULATION - PREDICTING STOCK PRICES USING MONTECARLO PACKAGE IN R

Exercise 1: Simulate a Single Stock Price

Problem:

Simulate a single stock&#39;s price for 1 year with:

 Initial price (S0=100S_0 = 100S0=100)

 Daily return mean (μ=0.001\mu = 0.001μ=0.001)

 Volatility (σ=0.02\sigma = 0.02σ=0.02)

 252 trading days.

Procedure:

1. Define a function to simulate daily prices.

2. Run the simulation.

3. Plot the price trajectory.

Code:

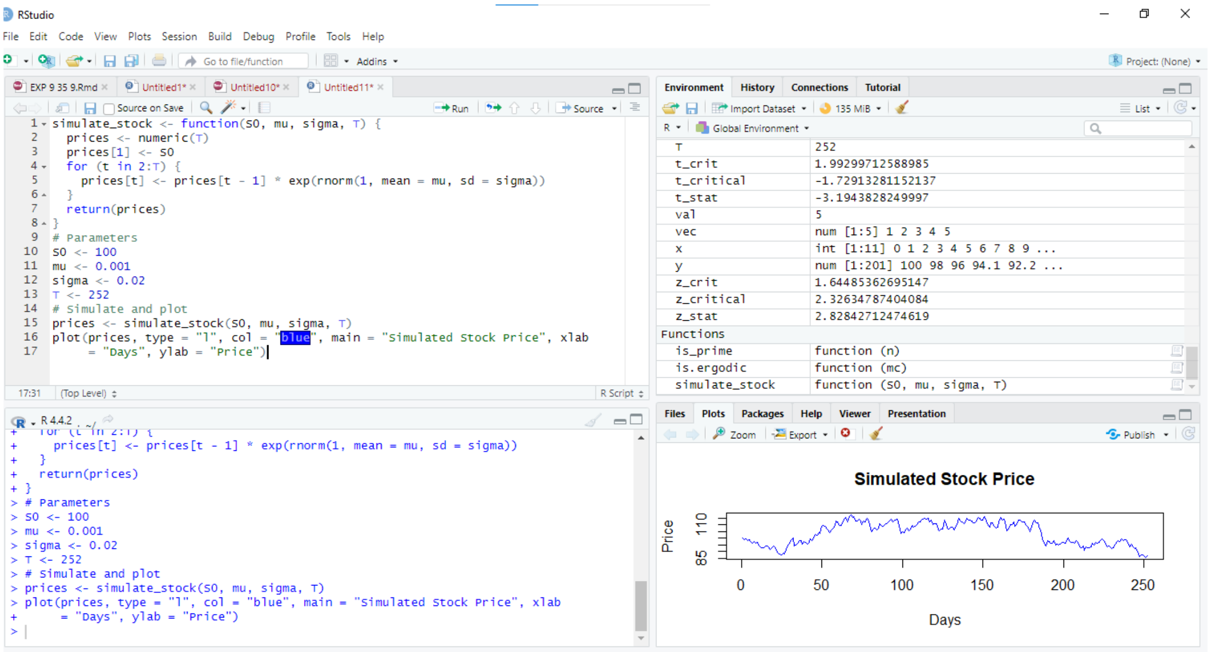
r

```
simulate_stock <- function(S0, mu, sigma, T) {
  prices <- numeric(T)
  prices[1] <- S0
  for (t in 2:T) {
    prices[t] <- prices[t - 1] * exp(rnorm(1, mean = mu, sd = sigma))
  }
  return(prices)
}
# Parameters
S0 <- 100
mu <- 0.001
sigma <- 0.02
T <- 252
# Simulate and plot
prices <- simulate_stock(S0, mu, sigma, T)
```

plot(prices, type = "l", col = "blue", main = "Simulated Stock Price", xlab

= "Days", ylab = "Price")

OUTPUT:



```
simulate_stock <- function(S0, mu, sigma, T) {
  prices <- numeric(T)
  prices[1] <- S0
  for (t in 2:T) {
    prices[t] <- prices[t - 1] * exp(rnorm(1, mean = mu, sd = sigma))
  }
  return(prices)
}
# Parameters
S0 <- 100
mu <- 0.001
sigma <- 0.02
T <- 252
# Simulate and plot
prices <- simulate_stock(S0, mu, sigma, T)
plot(prices, type = "l", col = "blue", main = "Simulated Stock Price", xlab
     = "Days", ylab = "Price")
```

T	252
t_crit	1.99299712588985
t_critical	-1.72913281152137
t_stat	-3.1943828249997
val	5
vec	num [1:5] 1 2 3 4 5
x	int [1:11] 0 1 2 3 4 5 6 7 8 9 ...
y	num [1:201] 100 98 96 94.1 92.2 ...
z_crit	1.64485362695147
z_critical	2.32634787404084
z_stat	2.82842712474619
Functions	
is_prime	function (n)
is.ergodic	function (mc)
simulate_stock	function (S0, mu, sigma, T)

```
+   for (t in 2:T) {
+     prices[t] <- prices[t - 1] * exp(rnorm(1, mean = mu, sd = sigma))
+   }
+   return(prices)
+ }
> # Parameters
> S0 <- 100
> mu <- 0.001
> sigma <- 0.02
> T <- 252
> # Simulate and plot
> prices <- simulate_stock(S0, mu, sigma, T)
> plot(prices, type = "l", col = "blue", main = "Simulated Stock Price", xlab
+      = "Days", ylab = "Price")
> 
```

Output:

A line graph of the stock price over 252 days, showing random variations.

Exercise 2: Simulate Multiple Price Paths

Problem:

Simulate 5 paths for the same stock.

Procedure:

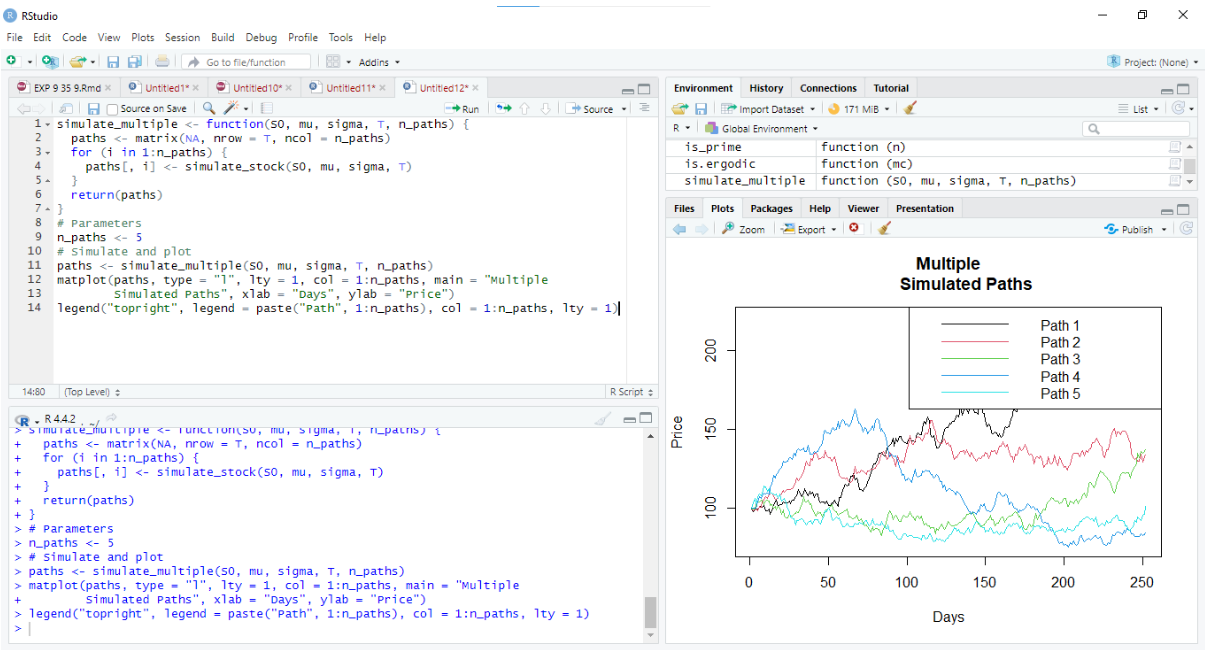
1. Modify the function to handle multiple simulations.
2. Overlay the paths in a single plot.

Code:

r

Copy code

```
simulate_multiple &lt;- function(S0, mu, sigma, T, n_paths) {
paths &lt;- matrix(NA, nrow = T, ncol = n_paths)
for (i in 1:n_paths) {
paths[, i] &lt;- simulate_stock(S0, mu, sigma, T)
}
return(paths)
}
# Parameters
n_paths &lt;- 5
# Simulate and plot
paths &lt;- simulate_multiple(S0, mu, sigma, T, n_paths)
matplot(paths, type = &quot;l&quot;, lty = 1, col = 1:n_paths, main = &quot;Multiple
Simulated Paths&quot;, xlab = &quot;Days&quot;, ylab = &quot;Price&quot;)
legend(&quot;topright&quot;, legend = paste(&quot;Path&quot;, 1:n_paths), col = 1:n_paths, lty
= 1)
```



```r
simulate_multiple <- function(S0, mu, sigma, T, n_paths) {
  paths <- matrix(NA, nrow = T, ncol = n_paths)
  for (i in 1:n_paths) {
    paths[, i] <- simulate_stock(S0, mu, sigma, T)
  }
  return(paths)
}
# Parameters
n_paths <- 5
# Simulate and plot
paths <- simulate_multiple(S0, mu, sigma, T, n_paths)
matplot(paths, type = "l", lty = 1, col = 1:n_paths, main = "Multiple
        Simulated Paths", xlab = "Days", ylab = "Price")
legend("topright", legend = paste("Path", 1:n_paths), col = 1:n_paths, lty = 1)
```

```
> simulate_multiple <- function(S0, mu, sigma, T, n_paths) {
+   paths <- matrix(NA, nrow = T, ncol = n_paths)
+   for (i in 1:n_paths) {
+     paths[, i] <- simulate_stock(S0, mu, sigma, T)
+   }
+   return(paths)
+ }
> # Parameters
> n_paths <- 5
> # Simulate and plot
> paths <- simulate_multiple(S0, mu, sigma, T, n_paths)
> matplot(paths, type = "l", lty = 1, col = 1:n_paths, main = "Multiple
+         Simulated Paths", xlab = "Days", ylab = "Price")
> legend("topright", legend = paste("Path", 1:n_paths), col = 1:n_paths, lty = 1)
> 
```

is_prime	function (n)
is.ergodic	function (mc)
simulate_multiple	function (S0, mu, sigma, T, n_paths)

Output:

A graph showing 5 distinct but overlapping paths for the stock price.

Exercise 3: Monte Carlo Simulation of Terminal Price

Problem:

Estimate the stock price at the end of 1 year using 1,000 simulations.

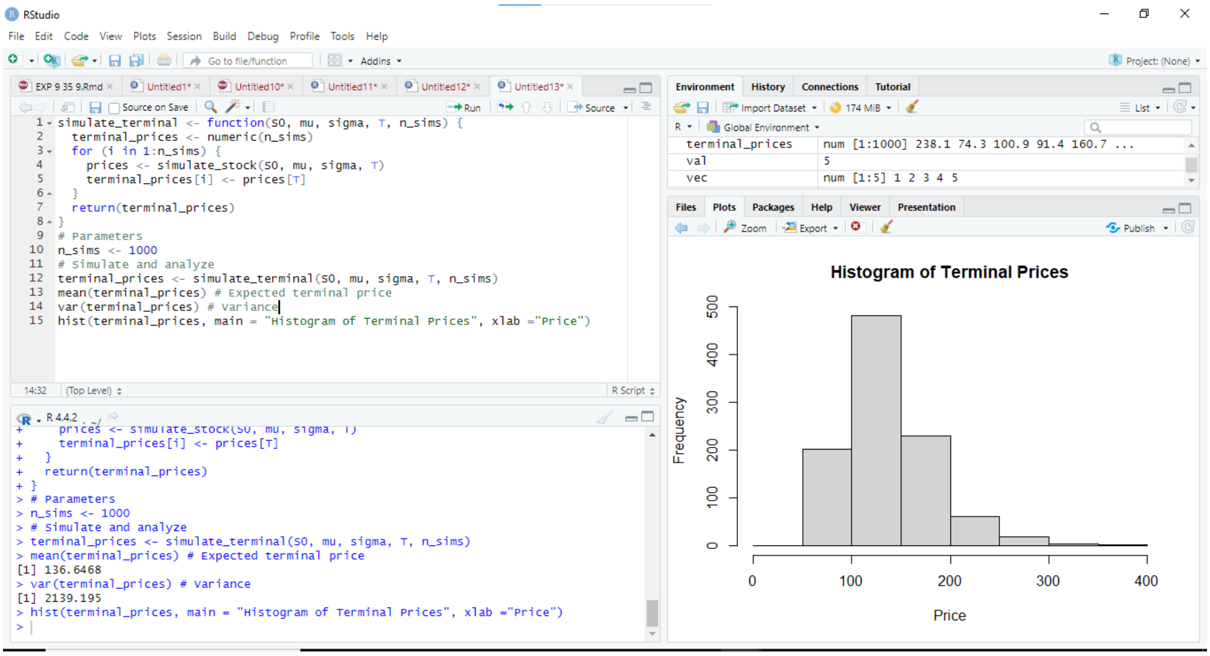
Procedure:

1. Simulate terminal prices.

2. Calculate the mean and variance.

Code:

```
simulate_terminal <- function(S0, mu, sigma, T, n_sims) {
  terminal_prices <- numeric(n_sims)
  for (i in 1:n_sims) {
    prices <- simulate_stock(S0, mu, sigma, T)
    terminal_prices[i] <- prices[T]
  }
  return(terminal_prices)
}
# Parameters
n_sims <- 1000
# Simulate and analyze
terminal_prices <- simulate_terminal(S0, mu, sigma, T, n_sims)
mean(terminal_prices) # Expected terminal price
var(terminal_prices) # Variance
hist(terminal_prices, main = "Histogram of Terminal Prices", xlab ="Price")
```



```
simulate_terminal <- function(S0, mu, sigma, T, n_sims) {
  terminal_prices <- numeric(n_sims)
  for (i in 1:n_sims) {
    prices <- simulate_stock(S0, mu, sigma, T)
    terminal_prices[i] <- prices[T]
  }
  return(terminal_prices)
}
# Parameters
n_sims <- 1000
# Simulate and analyze
terminal_prices <- simulate_terminal(S0, mu, sigma, T, n_sims)
mean(terminal_prices) # Expected terminal price
var(terminal_prices) # Variance
hist(terminal_prices, main = "Histogram of Terminal Prices", xlab ="Price")
```

```
+     prices <- simulate_stock(S0, mu, sigma, T)
+     terminal_prices[i] <- prices[T]
+   }
+   return(terminal_prices)
+ }
> # Parameters
> n_sims <- 1000
> # Simulate and analyze
> terminal_prices <- simulate_terminal(S0, mu, sigma, T, n_sims)
> mean(terminal_prices) # Expected terminal price
[1] 136.6468
> var(terminal_prices) # Variance
[1] 2139.195
> hist(terminal_prices, main = "Histogram of Terminal Prices", xlab ="Price")
> 
```

```
terminal_prices   num [1:1000] 238.1 74.3 100.9 91.4 160.7 ...
val               5
vec               num [1:5] 1 2 3 4 5
```

Output:

 Mean terminal price (e.g., ~101).

 Histogram of terminal prices.