
```
% Alexander Lerma
% Math 485

function [ call ] = monte_carlo( st, dt, k, r, sigma, n, payoff )
%SIMULATE_BM Compute n paths of brownian motion to price a call option
% param payoff:a lambda function, default to payoff of a call option if not
% set
    if nargin < 7
        payoff = @(st) max(0, st - k);
    end

    stock_prices = NaN([1, n + 1]); % need room for st, n + 1
    stock_prices(1) = st;

    payoffs = NaN([1, n + 1]);

    i = 2;
    expon = exp((r - (sigma ^ 2) / 2) * dt);
    rvs = randn([1, n]);
    for rv = rvs
        brownian = exp(sigma * sqrt(dt) * rv);
        stock_prices(i) = stock_prices(i-1) * expon * brownian;
        payoffs(i) = payoff(stock_prices(i));
        i = i + 1;
    end

    payoffs(1) = mean(payoffs(1, 2:length(payoffs)));
    call = payoffs(1);
end

Error using monte_carlo (line 10)
Not enough input arguments.
```

Published with MATLAB® R2014b