Series 2a

1. Monte Carlo Approximation for Numerical Integration

Let $A, B \subseteq \mathbb{R}^2$ be the sets defined by

$$A = \left\{(x,y) \in \mathbb{R}^2 \colon (x-2)^2 + y^2 \leq 4\right\} \quad \text{and} \quad B = \left\{(x,y) \in \mathbb{R}^2 \colon x^2 + (y-2)^2 \leq 4\right\},$$

and consider the function $f \colon \mathbb{R}^2 \to \mathbb{R}$ defined by

$$f(x,y) = \mathbb{1}_{A\cap B}(x,y) \cdot |x|^{2/3}.$$

Let $(\Omega, \mathcal{F}, P)$ be a probability space, and let $Y_n, Z_n : \Omega \to \mathbb{R}, n \in \mathbb{N}$, be independent $\mathcal{U}_{(0,1)}$-distributed random variables. For each $N \in \mathbb{N}$, define the random variable $I_N \colon \Omega \to \mathbb{R}$ by

$$I_N = \frac{4}{N} \sum_{n=1}^{N} f(2Y_n, 2Z_n).$$

The random variables $I_N, N \in \mathbb{N}$, are thus Monte Carlo approximations of $\mathbb{E}_P[4f(2Y_1, 2Z_1)]$.

a) Prove or disprove the following statement: The random variables I_N, $N \in \mathbb{N}$, are P-unbiased estimators of the integral

$$\int_0^2 \int_0^2 f(x,y)\, \mathrm{d}x \mathrm{d}y.$$

b) Write a MATLAB function `MonteCarlo(`N`)` with input $N \in \mathbb{N}$ and output a realization of I_N.

c) Write a MATLAB function `MonteCarloPlot()` which, for each $k \in \{2,3,4,5,6\}$, plots five realizations of I_{10^k}, each represented by a blue star in a coordinate plane. Plot the values of k on the x-axis and the corresponding realizations of I_{10^k} on the y-axis. The plot should contain a total of 25 blue stars.

2. Approximative Realizations of a One-Dimensional Standard Brownian Motion

In this exercise, we do not distinguish between pseudo-random numbers and actual random numbers. Let A be the set defined by

$$A = \bigcup_{n=1}^{\infty} \{\mathbf{t} = (t_1, \ldots, t_n) \in [0,\infty)^n \colon \#_{\mathbb{R}}(\{t_1, \ldots, t_n\}) = n\},$$

and define the function length$\colon A \to \mathbb{N}$ such that for all $n \in \mathbb{N}$ and all $\mathbf{t} = (t_1, \ldots, t_n) \in [0,\infty)^n \cap A$,

$$\mathrm{length}(\mathbf{t}) = n.$$

Let further $Q : A \to \bigcup_{n=1}^{\infty} \mathbb{R}^{n\times n}$ be the function defined such that for all $n \in \mathbb{N}$ and for all $\mathbf{t} = (t_1, \ldots, t_n) \in [0,\infty)^n \cap A$, we have

$$Q(\mathbf{t}) = (\min\{t_i, t_j\})_{(i,j)\in\{1,\ldots,n\}^2}.$$

Write a MATLAB function `StandardBrownianMotion(t)` with input $\mathbf{t} \in A$ and output a realization of an $\mathcal{N}_{0,Q(\mathbf{t})}$-distributed random variable. The MATLAB function `StandardBrownianMotion(t)` may use at most length($\mathbf{t}$) realizations of an $\mathcal{N}_{0,I_{\mathbb{R}}}$-distributed random variable.

Test your implementation by calling the following MATLAB commands:

```
% Set random number generator to default
rng('default');

% Define parameters
N = 10^3;
preimage = linspace(0, 1, N+1);

% Generate and plot realizations of the standard Brownian motion
X = StandardBrownianMotion(preimage);
plot(preimage, X, 'b');
hold on;

X = StandardBrownianMotion(preimage);
plot(preimage, X, 'r');

X = StandardBrownianMotion(preimage);
plot(preimage, X, 'g');

% Add labels and a title
xlabel('t');
ylabel('X(t)');
title('Realizations of Standard Brownian Motion');
legend('BM (blue)', 'BM (red)', 'BM (green)');

hold off;
```

3. Brownian Motion and Monte Carlo Simulation

Let $(\Omega, \mathcal{F}, P)$ be a probability space and $W\colon [0,T]\times\Omega \to \mathbb{R}$ be a standard Brownian motion ($T > 0$).

a) For $N \in \mathbb{N}$, define a temporal discretization (or mesh) on the time interval $[0,T]$ by:

$$0 = t_0 < t_1 < \cdots < t_{N-1} < t_N = T, \quad t_n := \frac{n}{N}T. \tag{1}$$

Write a MATLAB function `BrownianMotion(`T`, `N`)` with inputs $T \in (0,\infty)$ and $N \in \mathbb{N}$, and output a realization of a $\big(W_0, W_{t_1}, \dots, W_{t_{N-1}}, W_T\big)\big(P_{\mathcal{B}(\mathbb{R}^{N+1})}\big)$-distributed random variable. (You may use your results from Problem 2.) Now, assume $T = 1$. Write a MATLAB function `BrownianMotionPlot()` which uses the function `BrownianMotion(`T`, `N`)` to generate and plot 5 sample paths of the process $\widetilde{W}^{1000}$ over the time interval $[0,1]$, where, for $N \in \mathbb{N}$, the process $\widetilde{W}^N\colon [0,T]\times\Omega \to \mathbb{R}$ is defined by the linear interpolation of W at the mesh points $\{t_0, t_1, \dots, t_N\}$, that is:

$$\widetilde{W}_t^N = \left(n + 1 - \frac{tN}{T}\right) W_{t_n} + \left(\frac{tN}{T} - n\right) W_{t_{n+1}},$$

for all $n \in \{0, 1, \dots, N-1\}$ and each $t \in [t_n, t_{n+1}]$.

b) Let $X = (X_t)_{t\in[0,1]}$ be the stochastic process defined by $X_t = 1 + t + 3W_t$. First, prove that

$$\mathbb{E}\big[e^{\sigma W_t}\big] = \exp\big(\tfrac{1}{2}\sigma^2 t\big), \quad \text{for all} \ \ \sigma \in \mathbb{R} \ \ \text{and} \ \ t \in [0,1].$$

Then, perform the following tasks:

(i) Plot 10 sample paths of X, using the discretization (1) for $N = 10^3$.

(ii) Compute $\mu := \mathbb{E}[\exp(X_1)]$ exactly and then approximate it using a Monte Carlo simulation with $M = 10^5$ samples. Provide a 95% confidence interval for your estimate based on the Central Limit Theorem (CLT), and compare the exact value of μ with your simulation result.

Submission Deadline: Wednesday, 23 October 2024, by 2:00 PM.