Christian-Albrechts-Universität zu Kiel **SS 2023**
Mathematisches Seminar
Prof. Dr. Sören Christensen
Henrik Valett, Fan Yu, Oskar Hallmann Sheet 03

Computational Finance

Exercises for all participants

C-Exercise 08 (Numerical integration via MC) (4 points)
We want to evaluate definite integrals using Monte-Carlo integration. Write a function

```
MC_integration (N, f, a, b)
```

which computes the MC-estimator of a definite integral of a function f on $[a,b]$, i.e.

$$\frac{(b-a)}{N}\sum_{i=1}^{N} f(x_i)$$

where x_i is uniformly distributed on $[a,b]$. Test your function by computing the integral

$$\int_0^1 \sqrt{1-x^2}dx$$

with $N = 10, 100, 1000, 10000$ samples.
Useful Python commands: `np.random.uniform`

C-Exercise 09 (Sampling from a distribution by the acceptance/rejection method) (4 points)
We want to generate samples of the Beta distribution with parameters $\alpha = 2, \beta = 5$ which has the following density

$$f(x) = \begin{cases} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}, & \text{if } 0 \leq x \leq 1 \\ 0, & \text{otherwise.} \end{cases},$$

Write a Python function

```
Sample_dist_AR (N, alpha, beta)
```

that generates and returns $N \in \mathbb{N}$ independent samples from the distribution by means of the acceptance/rejection method. In your algorithm, you may sample only from the uniform distribution on $[0,1]$ using the function `numpy.random.uniform`. You can access the gamma function $\Gamma(x)$ using `scipy.special.gamma`.

Generate $N = 10000$ samples, and plot them in a histogram. Plot the density $f(x)$ in the same histogram using the right scaling.

Useful Python commands: `plt.hist`, `np.random.uniform`, `while`

C-Exercise 10 (Valuation of European options in the Black-Scholes model using Monte-Carlo) (4 points)
Write a Python function

```
Eu_Option_BS_MC (S0, r, sigma, T, K, M, f)
```

that computes the initial price $V(0) = e^{-rT}\mathbb{E}_Q[f(S(T))]$ of a European option with payoff $f(S(T))$ at maturity T for some strike price in the Black-Scholes model and the asymptotic 95%-confidence interval $[c_1, c_2]$ via the Monte-Carlo approach using $M \in \mathbb{N}$ simulations. In the B-S model we have the formula

$$S(T) = S(0)\exp\left((r - \sigma^2/2)T + \sigma\sqrt{T}X\right),$$

where X has law $N(0,1)$ under Q.
Test your function for a call option $f(x) = (x-100)^+$, $S_0 = 110$, $r = 0.04$, $\sigma = 0.2$, $T = 1$ and $M = 10000$ and compare the price with the BS-Formula.

Useful Python command: `numpy.random.normal`
Hint: The Black-Scholes formula for the European Call is given on exercise sheet 02.

T-Exercise 11 (Box-Muller method) (for math only)
Prove that the *Box-Muller method* indeed works. I.e. show that if you have two independent random variables U_1, U_2 which are uniformly distributed on the interval $[0,1]$ then the random variables X_1, X_2 defined via

$$X_1 = \sqrt{-2\log(U_1)}\cos(2\pi U_2)$$
$$X_2 = \sqrt{-2\log(U_1)}\sin(2\pi U_2)$$

are independent and standard normally distributed.

Hint: Use Theorem 2.1 to find the density of the vector (X_1, X_2).

Please include your name(s) as comment in the beginning of the file.
Do not forget to include comments in your Python-programs.
Submit until: Fri, 05.05.2022, 10:00