Universität Konstanz
Fachbereich Mathematik und Statsitik
Johannes Schropp
Jan Rohleff

Wintersemester 2023/2024

Numerik stochastischer Differentialgleichungen

`https://www.math.uni-konstanz.de/~rohleff/numsde.html`

3. Übungsblatt

Ausgabe: 27.11.2023, Abgabe: 04.12.2023, bis 10.00 Uhr

Aufgabe 3.1 (6 Punkte)

Sei $X : \Omega \to \mathbb{R}^d$ eine vektorwertige Zufallsvariable mit der Dichtefunktion f und sei $g : \mathbb{R}^d \to \mathbb{R}$ messbar. Für „gute“ Pseudozufallsvektoren X_i (identisch verteilt wie X) gilt mit der *Monte-Carlo-Integration* approximativ für große N

$$\frac{1}{N}\sum_{i=1}^{N} g(X_i) \approx E(g(X)) = \int_{\mathbb{R}^d} g(x)f(x)\,dx.$$

a) Berechnen Sie analytisch folgende Integrale:

$$\int_{[0,1]^d} \sum_{j=1}^{d} x_j^2\,dx \quad \text{und} \quad \int_{[-1,1]^d} e^{-\frac{1}{2}|x|^2}\,dx,$$

wobei $x = (x_1, \dots, x_d)^T \in \mathbb{R}^d$.
Hinweis: Das zweite Integral kann über das Gaußsche Fehlerintegral berechnet werden.

b) Schreiben Sie ein Matlab/Python-Programm zur approximativen Berechnung der Integrale aus Teil a) mittels Monte-Carlo-Integration für $d = 10$. Erzeugen Sie die dazu benötigten Pseudozufallsvektoren durch die Befehle `rand` bzw. `randn` (Python `np.random.rand` bzw. `np.random.randn`) Verwenden Sie $1 \le N \le 100000$ Zufallsvektoren, um so die Konvergenz des approximativen Integralwerts gegen den exakten Wert zu kontrollieren und zeichnen Sie den approximativen Integralwert in Abhängigkeit von N zusammen mit dem exakten Wert (für das zweite Integral ist der Befehl `erf` zu verwenden) in ein Schaubild.

Aufgabe 3.2 (6 Punkte)

a) Schreiben Sie eine Matlab/Python-Funktion **cdf** (cumulative distribution function), so dass durch die Befehle `[x,y] = cdf(w,[a,b])` und `plot(x,y)` die diskrete Verteilungsfunktion

$$F_N(z) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{(-\infty,z]}(w_i), \qquad z \in \mathbb{R}$$

der PZZ $w_1, \dots, w_N$ eingeschränkt auf das Intervall $[a, b]$ dargestellt wird.

Testen Sie das Programm für $N = 1000$ und $[a, b] = [-3, 3]$ mit auf $(0, 1)$ gleichverteilten und standardnormalverteilten PZZ (diese sollen in dem Vektor `w` an die `cdf`-Funktion übergeben werden).
Hinweis: Hilfreich ist hier der Matlab-Befehl `sort` (Python `np.sort`).

b) Ist $X_1, X_2, \ldots$ eine Folge von i.i.d.-Zufallsvariablen mit Erwartungswert $\bar{\mu}$ und Standardabweichung σ, so ist die Zufallsvariable

$$S^{(M)} = \frac{1}{\sigma\sqrt{M}} \sum_{i=1}^{M} (X_i - \bar{\mu}) \tag{1}$$

nach dem zentralen Grenzwertsatz für großes M annähernd standardnormalverteilt.

Schreiben Sie ein Matlab/Python-Programm, welches für ein gegebenes M ausgehend von einer Folge von Pseudozufallsvektoren im $\mathbb{R}^M$ (wobei jede Komponente wie X_i verteilt ist) N neue PZZ $S_1^{(M)}, S_2^{(M)}, \ldots, S_N^{(M)}$ der Form (1) erzeugt und anschließend mithilfe der `cdf`-Funktion aus Teil a) die diskrete Verteilungsfunktion von $S_1^{(M)}, S_2^{(M)}, \ldots, S_N^{(M)}$ im Intervall $[a, b]$ zeichnet.

Testen Sie Ihr Programm mit PZZ, die durch den `rand`-Befehl erzeugt werden und wählen Sie $[a, b] = [-3, 3]$, $N = 5000$. Zeichnen Sie zudem die Verteilungsfunktion der Standardnormalverteilung mit in das Schaubild. Sehen Sie für $M = 1,\ 10,\ 100$ die Konvergenz gegen die Standardnormalverteilung?

Aufgabe 3.3 (6 Punkte)

Gegeben sei die allgemeine lineare stochastische Differentialgleichung

$$\begin{aligned} dX_t &= (\alpha_t + \mu_t X_t)\, dt \ + \ (\beta_t + \sigma_t X_t)\, dW_t, \\ X_0 &= \bar{X}. \end{aligned}$$

Zeigen Sie für die ersten beiden Momente $m_1(t) := E(X_t)$ und $m_2(t) := E(X_t^2)$ die folgenden Identitäten:

$$\begin{aligned} m_1(t) &= e^{a(t)} E(\bar{X}) + e^{a(t)} \int_0^t \alpha_s e^{-a(s)}\, ds, \quad a(t) = \int_0^t \mu_s\, ds, \\ m_2(t) &= e^{b(t)} E(\bar{X}^2) + e^{b(t)} \int_0^t v(s) e^{-b(s)}\, ds, \quad b(t) = \int_0^t u(s)\, ds, \end{aligned}$$

wobei $u(t) = 2\mu_t + \sigma_t^2$, $v(t) = 2(\alpha_t + \sigma_t \beta_t) m_1(t) + \beta_t^2$.

Hinweis: Leiten Sie deterministische Anfangswertaufgaben für m_1 und m_2 her.