Computerphysik WS 2017/2018

Prof. Dr. Petra Imhof, FU Berlin

Übungsblatt 10: Zufallszahlen und Fouriertransformation (20 Bonus-Punkte*)

15. Januar 2018

* nur bei Erscheinen im Tutorium

Abgabe bis **21.01.2018** 23:55 Uhr

Aufgabe 10.1: Random Numbers (10 Punkte)

(a) (3 Punkte) Implementieren Sie den in der Vorlesung vorgestellten linearen kongruenten Zufallszahlengenerator mit der Iterationsvorschrift:

$$x_{i+1} = (a \cdot x_i + b)\%c. \tag{1}$$

Der Modulo-Operator wird in Python als % geschrieben.

Leiten Sie den Startwert x_0 von der Systemzeit wie folgt her

$$x_0 = t_6 + 70 \cdot (t_5 + 12 \cdot (t_4 + 3) \cdot (t_3 + 23 \cdot (t_2 + 59t_1))) \tag{2}$$

wobei t_6 das Jahr, t_5 der Monat, t_4 der Tag, t_3 die Stunde, t_2 die Minute und t_1 die Sekunde ist.

Verwenden Sie $a = 7^5$, $b = 0$ und $c = (2^{31} - 1)$.

Hinweis: Zur Bestimmung der Systemzeit können Sie zB `datetime.datetime.now()` textitverwenden.

(b) (1 Punkt) Generieren Sie eine Folge von $N = 10.000$ Zufallszahlen $r_1, \ldots, r_N$ im Bereich $[0, 1)$ mit der Vorschrift $r_i = n_i/c$ mit ihrem eigenen Zufallsgenerator und mit numpy.random.uniform. Berechnen Sie jeweils das arithmetische Mittel und die Standardabweichung der Zufallszahlen. Plotten Sie die Verteilung der Zufallszahlen jeweils in einem Histogramm.

(c) (2 Punkte) Definieren Sie eine Funktion, die die Zweipunktkorrelation bestimmt.

(d) (2 Punkt) Berechnen Sie die Zweipunktkorrelationen ihrer beiden random Number Datensätze für $l = [0, 10]$. Plotten Sie jeweils die Zweipunktkorrelationen über l. Kommentieren Sie ihre Ergebnisse.

(e) (2 Punkt) Erzeugen Sie wie zuvor zwei Datensätze an random numbers im Bereich $[0, 1)$ nun jedoch mit jeweils 10^7 Zahlen. Suchen Sie in beiden Datensätzen nach Zahlen $< 10^6$. Geben Sie diese Zahlen sowie die jeweils direkt nachfolgende Zahl aus, was beobachten Sie?

Aufgabe 10.2: Monte-Carlo Integration (5 Punkte)

Gegeben sei die aus Aufgabe 8.2 bekannte Funktion:

$$f(x) = -x^4 - x^3 + 13 \cdot x^2 + x - 12$$

(a) (1 Punkt) Implementieren sie eine Funktion, um mittels Monte-Carlo-Verfahren das Integral einer Funktion in einem Intervall $[a, b]$ zu bestimmen. Zur Erzeugung von Zufallszahlen verwenden Sie hier `numpy.random`

(b) (1 Punkte) Bestimmen Sie mittel Monte-Carlo Integration das Integral $von f(x)$ im Intervall $[-4, 3]$ mit $n = 100$ und vergleichen Sie den erhaltenen Wert mit dem analytischen Ergebnis.

(c) (3 Punkte) Variieren Sie nun die Anzahl Evaluationspunkte n so, dass sie gleich der Anzahl Stützpunkte einer Diskretisierung mit Schrittweite zwischen 1 und 10^{-6} sind (Anzhal der in Aufgabe 8.2 verwendeten Stützpunkte). Vergleichen Sie den Fehler gegenüber dem analytischen Ergebnis mit den Fehlern der Integration mittel Rechteck- Trapez- oder Simpsonregel mit gleich der Anzahl Stützstellen aus Aufgabe 8.2

10.1 Fouriertransformation (5 Punkte)

Gegebn ist ein Mess-Signal (signal.txt, Sampling Frequenz 1000 Hz, 1024 Samples), das aus einigen überlagerten Schwingungen resultiert.

(a) (1 Punkte): Implementieren Sie eine Funktion, welche eine diskrete Fouriertransformation einer Datenreihe durchführt.

(b) (2 Punkte): Implementieren Sie eine weitere Funktion, welche eine Fast Fouriertransformation einer Datenreihe durchführt.

(c) (2 Punkt): Bestimmen Sie mit ihren beiden Funktionen das Frequenzspektrum für das Signal und identifizieren Sie die Frequenzen der beiden dominanten Schwingunen.

Hinweis: Zur Kontrolle können Sie die Funktion `numpy.fft.fft` *nutzen.*