3. Übungsblatt zur Vorlesung
Statistische Methoden der Datenanalyse
Abgabe: 07.11.2017, 23:59 Uhr

WS 2017/2018
Prof. W. Rhode

Zeit	Raum	Abgabe im Moodle; Mails mit Betreff: [SMD1718]
Di. 10-12	CP-03-150	philipp2.hoffmann@udo.edu und jan.soedingrekso@udo.edu
Di. 16-18	P1-02-110	felix.neubuerger@udo.edu und tobias.hoinka@udo.edu
Di. 16-18	CP-03-150	simone.mender@udo.edu und maximilian.meier@udo.edu

Aufgabe 8: *Zufallszahlen verschiedener Verteilungen* **5 P.**

Die Zufallsvariable x möge der Wahrscheinlichkeitsdichte

$$f(x) = \begin{cases} 1 & 0 \leq x \leq 1 \\ 0 & \text{sonst} \end{cases}$$

(Gleichverteilung zwischen 0 und 1) genügen.

a) Mit welcher Wahrscheinlichkeit nimmt x einen Wert zwischen $\frac{1}{2}$ und $\frac{1}{3}$ an?

b) Mit welcher Wahrscheinlichkeit nimmt x den exakten Wert $\frac{1}{2}$ an?

c) Mit welcher Wahrscheinlichkeit liefert ein Zufallsgenerator auf einem Computer den exakten Wert $\frac{1}{2}$? Der Generator soll sein Ergebnis in Form einer binären Gleitkommazahl mit einer Mantisse von 23 Binärstellen darstellen.

d) Mit welcher Wahrscheinlichkeit liefert derselbe Zufallsgenerator den exakten Wert $\frac{2}{3}$?

Aufgabe 9: *Zufallszahlengeneratoren* **7 P.**

Linear-kongruente Zufallszahlengeneratoren erzeugen eine neue ganzzahlige Zufallszahl aus der vorhergehenden durch die Vorschrift

$$x_n = (a \cdot x_{n-1} + b) \mod m.$$

Division durch m ergibt dann eine zwischen 0 und 1 gleichverteilte reelle Zufallszahl.

a) Programmieren Sie einen solchen Zufallszahlengenerator mit $b = 3$ und $m = 1024$. Bestimmen Sie die Periodenlänge in Abhängigkeit des Parameters a, indem Sie für a Werte aus einem angemessenen Bereich verwenden. Stellen Sie den Zusammenhang von Periodenlänge und a in einem Plot dar. Wie groß ist die maximale Periodenlänge? Für welche Werte von a ist die Periodenlänge maximal? Lassen sich die erhaltenen Werte mit den Regeln für *gute* linear-kongruente Generatoren erklären? Hinweis: In dieser Aufgabe sollte der Startwert x_0 unverändert bleiben.

b) Verwenden Sie für die folgenden Aufgaben einen linear-kongruenten Zufallszahlengenerator mit den Parametern $a = 1601$, $b = 3456$ und $m = 10\,000$.

c) Erzeugen Sie so 10 000 Zufallszahlen und stellen Sie diese als Histogramm dar. Entspricht das Ergebnis den Anforderungen an einen guten Zufallszahlengenerator? Hängt es vom Startwert x_0 ab, und wenn ja, wie?

d) Stellen Sie Paare bzw. Tripletts aufeinanderfolgender Zufallszahlen als zweidimensionales bzw. dreidimensionales Streudiagramm (engl. scatter plot) dar. Entspricht das Ergebnis den Anforderungen an einen guten Zufallszahlengenerator?

e) Erstellen Sie Histogramme wie in c) und d) auch mit `numpy.random.uniform()`.

f) Wie oft liefert der Zufallsgenerator aus Aufgabenteil b) den exakten Wert $\frac{1}{2}$? Hängt diese Anzahl vom Startwert ab? Geben Sie einen möglichen Startwert an, sodass der Generator $\frac{1}{2}$ erzeugen kann.

Beispiel für ein dreidimensionales Streudiagramm in `matplotlib`:

```
import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits.mplot3d import Axes3D

x, y, z = np.random.normal(size=(3, 1000))

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

ax.init_view(45, 30)  # Elevation, Rotation
ax.scatter(
  x, y, z,
  lw=0,  # no lines around points
  s=5,   # smaller points
)

plt.show()
```

Aufgabe 10: *Gleichverteilung* **8 P.**

Gegeben sei ein Zufallszahlengenerator, der gleichverteilte Zahlen z von 0 bis 1 liefert. Geben Sie **effiziente Algorithmen** an, und implementieren Sie diese, mit denen Sie Zufallszahlen erzeugen können, die den folgenden Verteilungen gehorchen:

a) Eine Gleichverteilung in den Grenzen x_{min} bis x_{max}

b) Exponentialgesetz: $f(t) = Ne^{-t/\tau}$ in den Grenzen 0 bis ∞ (N = Normierungskonstante)

c) Potenzgesetz: $f(x) = Nx^{-n}$ in den Grenzen x_{min} bis x_{max} ($n \geq 2$, N = Normierungskonstante)

d) Cauchy-Verteilung:

$$f(x) = \frac{1}{\pi}\frac{1}{1+x^2}$$

in den Grenzen $-\infty$ bis ∞

e) Die durch das (im Moodle unter *empirisches_histogramm.npy* zu findene) Histogramm gegebene empirische Verteilung. Die Datei enthält Binzentren (*bin_mid*) und die Höhen (*hist*). Das Histogramm besteht aus 50 Bins zwischen 0,0 und 1,0.

Zum Einlesen und Darstellen dieses Histogramms können Sie z.B. so vorgehen:

```python
from matplotlib import pyplot as plt
import numpy as np
data = np.load("empirisches_histogramm.npy")
plt.hist(data['bin_mid'], bins=np.linspace(0., 1., 51),
         weights=data['hist'])
plt.show()
```