

# Übungsblatt 8: $\pi$ thon

2.12.2013

## Allgemeine Hinweise

- Abgabetermin für die Lösungen ist
  - **Dienstag, 10.12., 10:00** für die Übungsgruppen am Donnerstag und Freitag
  - **Mittwoch, 11.12., 10:00** für die Übungsgruppen am Montag und Dienstag
- Schickt die Lösungen bitte per Email an Euren Tutor.

In diesem Blatt geht es um die (näherungsweise) Berechnung der Kreiszahl  $\pi$ . Drei Methoden dafür sind in Abbildung 1 skizziert.

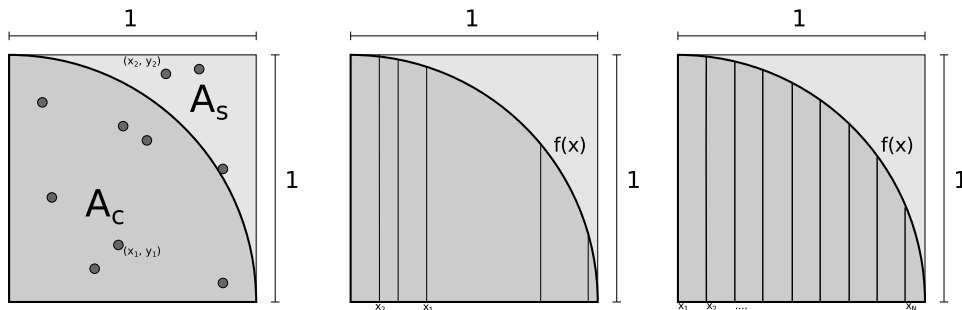


Abbildung 1: Drei Methoden zur Abschätzung von  $\pi$  (siehe Aufgaben 8.1, 8.2 und 8.3)

## Aufgabe 8.1: $\pi$ thon 1 (3 Punkte)

Eine einfache Methode, um eine Abschätzung für die Kreiszahl  $\pi$  zu erhalten, ist die Folgende. Die Idee dabei ist es, zufällig  $N_s$  Punkte  $(x_i, y_i)$  aus einem Einheitsquadrat (Quadrat mit Kantenlänge 1) zu ziehen ( $0 \leq x_i \leq 1, 0 \leq y_i \leq 1$ ).  $N_c$  sei die Anzahl der Punkte davon, die in einem Viertel des Einheitskreises (Kreis mit Radius 1) liegen (also bei denen  $x^2 + y^2 < 1$  ist). Da wir wissen, daß die Fläche des Einheitsquadrats  $A_s = 1$  ist, und die Fläche des Einheitskreises  $A_c = \pi$ , sollte also gelten

$$\frac{N_c}{N} \approx \frac{\frac{1}{4}A_c}{A_s} = \frac{1}{4}\pi \Rightarrow \pi \approx 4\frac{N_c}{N}$$

Wir nähern damit die Fläche des Kreissegments an. Dabei gilt: je größer  $N$ , desto genauer ist die Abschätzung. Wir können  $\pi$  also durch folgende Formel annähern:

$$\pi \approx 4\frac{1}{N} \sum_{i=1}^N \chi_{x^2+y^2 < 1}(x_i, y_i)$$

wobei

$$\chi_{x^2+y^2 < 1}(x, y) = \begin{cases} 1, & \text{falls } x^2 + y^2 < 1 \\ 0, & \text{sonst} \end{cases}$$

Das Python-Skript/group/cgl/2013/08/compute\_pi.py soll die Zahl  $\pi$  mit Hilfe der beschriebenen Methode annähern. Leider ist es fehlerhaft. Korrigiert das Skript!

### Aufgabe 8.2: $\pi$ thon 2 (3 Punkte)

Die in Aufgabe 8.1 beschriebene Methode ist nicht besonders effizient. Nun wollen wir die Genauigkeit (und damit die Effizienz) der Methode erhöhen. Dazu betrachten wir nun die Kreislinie als Funktion

$$f(x) = \sqrt{1 - x^2}$$

Wieder wollen wir die Fläche des Kreissegments (also die Fläche unter der Funktion) ausrechnen, d.h. wir *integrieren* die Funktion  $f(x)$ . Dazu ziehen wir diesmal also eine Reihe  $x_i$  von zufällige Werten ( $0 \leq x_i \leq 1$ ) und nähern  $\pi$  wie folgt an:

$$\pi \approx 4 \frac{1}{N} \sum_{i=1}^N f(x_i)$$

Wieder gilt: je größer  $N$ , desto genauer ist die Abschätzung.

**Hinweis:** Die oben beschriebene Methode, eine Funktion mit Hilfe von Zufallszahlen zu integrieren nennt sich *Monte-Carlo-Methode* (benannt nach den Spielcasinos des Stadtteils von Monaco).

Verändert das in Aufgabe 8.1 verwendete Python-Skript o, daß es die Zahl  $\pi$  mit Hilfe der hier beschriebenen Methode annähert.

### Aufgabe 8.3: $\pi$ thon 3 (4 Punkte)

Die oben beschriebene Monte-Carlo-Methode ist recht einfach, funktioniert bei beliebigen Funktionen und ist in vielen Fällen daher sehr nützlich. Im konkreten Falle der Berechnung von  $\pi$  ist es aber noch sehr viel effizienter, die Werte von  $x_i$  nicht zufällig zu ziehen, sondern stattdessen ein gleichmässiges Gitter von Werten zwischen 0 und 1 zu benutzen.

Verändert das in Aufgabe 8.2 verwendete Python-Skript so, daß es die Zahl  $\pi$  mit Hilfe der hier beschriebenen Methode annähert.