



Miniprojekt 4

Abgabe bis Freitag, 2.2.2018, 9.45 Uhr

Problem der 100 Gefangenen

100 zum Tode verurteilten Gefangenen mit den Nummern 1 bis 100 wird vom Leiter eines Gefängnisses eine letzte Chance auf Begnadigung gegeben. In einen Raum mit 100 Schubladen wird in jede Schublade zufällig die Nummer genau eines Gefangenen gelegt. Nun betreten die Gefangenen nacheinander den Raum und jeder darf 50 Schubladen in einer beliebigen Reihenfolge öffnen und diese danach mit ihrem Inhalt wieder schließen. Findet so jeder der Gefangenen seine eigene Nummer werden alle begnadigt. Findet jedoch einer der Gefangenen seine Nummer nicht, bleiben alle zum Tode verurteilt. Bevor der erste Gefangene den Raum betritt dürfen die Gefangenen eine Strategie besprechen, danach ist keinerlei Kommunikation möglich.

Würden die Gefangenen die Schubladen zufällig ausählen, hätte jeder eine Chance von $1/2$ seine eigene Nummer zu finden. Daher ist die Wahrscheinlichkeit, dass alle Gefangenen ihre Nummern bei dieser Strategie finden gleich $(1/2)^{100} \approx 8 \cdot 10^{-31}$.

Mit der folgenden Strategie können die Gefangenen eine Überlebenschance von über 30% erreichen:

Zuerst ordnen die Gefangenen den Schubladen jeweils die Zahlen 1 bis 100 zu und gehen wie folgt vor:

- 1) Jeder Gefangene öffnet als erstes die Schublade mit seiner eigenen Nummer.
- 2) Der Gefangene öffnet die Schublade dessen Nummer er in der vorherigen Schublade gefunden hat.
- 3) Der Gefangene wiederholt Schritt 2.

Die Strategie wird abgebrochen, falls der Gefangene seine eigene Nummer gefunden hat oder er bereits 50 Schubladen geöffnet hat.

Aufgaben:

- a) Schreiben sie eine Funktion `my_gefangene(p)` mit einer Permutation als Input, welche die obige Strategie ausführt und das Ergebnis zurückgibt. Dabei soll die Funktion 1 zurückgeben falls alle Gefangenen ihre Nummer finden und 0 falls nicht.
- b) Schreiben sie eine Funktion `my_simulation(n)` mit einer natürlichen Zahl $n \in \mathbb{N}$ als Input, welche die Funktion aus a) mit n verschiedenen Permutationen ausführt und die resultierende geschätzte Gewinnwahrscheinlichkeit ausgibt. Dazu sollen n zufällige Permutationen der Länge 100 erzeugt werden und die Gewinnwahrscheinlichkeit durch $\frac{\#\text{Erfolge}}{n}$ geschätzt werden. Berechnen sie damit die geschätzte Erfolgswahrscheinlichkeit für $n = 10, 100, 1000$.

Hinweis: Mit dem folgenden Befehl werden zufällige Permutationen der Länge 100 in Sage erzeugt:

```
sage: Permutations(100).random_element()
```

- c) Schreiben sie eine Funktion `my_hist(n)` mit einer natürlichen Zahl $n \in \mathbb{N}$ als Input. Die Funktion soll zu n verschiedenen Permutationen der Länge 100 die maximalen Zykellängen der Permutationen in einem Histogramm plotten. Plotten sie das Histogramm für $n = 1.000.000$.

Hinweis: Mit den folgenden Befehlen wird ein Histogramm in Sage erzeugt:

```
sage: T = stats.TimeSeries(vector)
sage: T.plot_histogram(normalize=False, bins=10)
```

- d) Schreiben sie ein LaTeX-Dokument mit Titelseite und einer Dokumentation ihrer Ergebnisse, sowie dem vollständigen Lösungsweg der einzelnen Teilaufgaben.

Gesamtpunktzahl: 8/8/8/8=32 Punkte