Kapitel 3: Momente und Abweichungen

Bisherige Überlegungen: Bestimme Erwartungswert einer Zufallsvariablen (z.B. Laufzeit eines Las Vegas Algorithmus).

Jedoch ebenso wichtig: Wie hoch ist die Wahrscheinlichkeit, dass eine Zufallsvariable weit vom Erwartungswert abweicht?

1. Hilfsmittel Markovs Ungleichung

Sei X eine Zufallsvariable und $f: \mathbb{R} \to \mathbb{R}$ eine Funktion.
Dann definieren wir

$$E[f(X)] = \sum_x f(x) \cdot \text{Prob}[X = x]$$

Markovs Ungleichung: Sei Y eine Zufallsvariable, die nur nicht-negative Werte annimmt. Dann gilt für alle $t \geq 0$:

$$\text{Prob}[Y \geq t] \leq \frac{E[Y]}{t}$$

Beweis: Definiere Funktion f durch

$$f(y) = \begin{cases} 1 & \text{falls } y \geq t,\ y \geq 0 \\ 0 & \text{sonst } y \geq 0 \end{cases}$$

$$\Rightarrow \text{Prob}[Y \geq t] = E[f(Y)]$$

Wegen $f(y) \leq \frac{y}{t}$ folgt:

$$\text{Prob}[Y \geq t] = E[f(Y)] \leq E\left[\frac{Y}{t}\right] = \frac{E[Y]}{t}$$

$\square$

2. Hilfsmittel: Chebyshevs Schranke

- Sei X eine Zufallsvariable.
  Im folgenden schreiben wir auch μ_X für den Erwartungswert $E[X]$
- Die Varianz $var[X]$ (oder auch σ_X^2) ist $E[(X-\mu_X)^2]$
- Die Standardabweichung σ_X von X ist die positive Quadratwurzel von σ_X^2.

Chebyshevs Schranke: Sei X eine Zufallsvariable. Dann gilt für alle $t \geq 0$:

$$Prob[|X-\mu_X| \geq t \cdot \sigma_X] \leq \frac{1}{t^2}$$

Beweis: Sei $Y = (X-\mu_X)^2$.

$\Rightarrow Prob[|X-\mu_X| \geq t \cdot \sigma_X] =$

$Prob[(X-\mu_X)^2 \geq t^2 \cdot \sigma_X^2] =$

$Prob[Y \geq t^2 \cdot E[Y]] \leq$

Markovs Ungleichung

$$\frac{E[Y]}{t^2 \cdot E[Y]} = \frac{1}{t^2}$$

Eine Anwendung von Chebyshevs Schranke: Randomisierte Auswahl

Gegeben: Menge S von n verschiedenen Zahlen, $k \in \{1, \dots, n\}$

Gesucht: Das k-kleinste Element aus $S = S_{(k)}$

Für $t \in S$ sei $r_S(t)$ der Rang von t in S:

$$S_{(r_S(t))} = t, \quad r_S(S_{(k)}) = k$$

Algorithmus LazySelect

(1) Wähle $n^{3/4}$ viele Elemente aus S zufällig, gleichverteilt und mit Zurücklegen aus.
$\rightarrow$ Multimenge R der Größe $n^{3/4}$

(2) Sortiere R mit $\mathcal{O}(n^{3/4} \cdot \log(n)) \subseteq o(n)$ vielen Vergleichen, z.B. mit heapsort.

(3) $x := k \cdot n^{-1/4}$
$l := \max\{1, \lfloor x - \sqrt{n} \rfloor\}$; $h := \min\{n^{3/4}, \lceil x + \sqrt{n} \rceil\}$
$a := R_{(l)}$; $b := R_{(h)}$;
Vergleiche a und b mit jedem Element aus S ($2n$ Vergleiche), und bestimme so $r_S(a)$ und $r_S(b)$.

(4) if $k < n^{1/4}$ then $P := \{y \in S \mid y \leq b\}$
elseif $k > n - n^{1/4}$ then $P := \{y \in S \mid y \geq a\}$
elseif $k \in [n^{1/4}, n - n^{1/4}]$ then $P := \{y \in S \mid a \leq y \leq b\}$

(5) if $S_{(k)} \in P$ und $|P| \leq 4 \cdot n^{3/4} + 2n^{1/4} - 1$ then goto (6)
else goto (1)

(6) Sortiere P mit $\mathcal{O}(|P| \cdot \log(|P|)) \subseteq \mathcal{O}(n^{3/4} \cdot \log(n)) \subseteq o(n)$ Vergleichen

Bestimme dann $P_{(k - r_S(a) + 1)} = S_{(k)}$
bzw. $P_{(k)} = S_{(k)}$ falls $P = \{y \in S \mid y \leq b\}$

- Beachte: In (5) können wir in Zeit $\mathcal{O}(1)$ überprüfen, ob $S_{(k)} \in P$ gilt, da $r_S(a)$ und $r_S(b)$ bekannt sind.

- Wir wollen im folgenden zeigen, dass LazySelect mit hoher Wahrscheinlichkeit nur einmal (1)–(5) durchläuft.
Im diesem Fall ist die Gesamtlaufzeit $2n + o(n)$.

- **Bemerkung**: Der beste deterministische Auswahl-Algorithmus hat eine Worst Case Laufzeit von $3n$ (und ist recht kompliziert).

- Wir untersuchen hier nur den (schwierigsten Fall) $P = \{y \in S \mid a \leq y \leq b\}$ d.h. $k \in [n^{1/4}, n - n^{1/4}]$

- Wann haben wir in Schritt (5) $P \not\ni S$ und gehen wieder zu Schritt (1):

Möglichkeit A: ~~[illegible]~~ $S_{(k)} \notin P$, d.h.

$R_{(l)} = a > S_{(k)}$ oder $R_{(h)} = b < S_{(k)}$

Möglichkeit B:

$|P| > 4 \cdot n^{3/4} + 2n^{1/4} - 1$

Wir werden zeigen, dass beide Möglichkeiten nur mit kleiner Wahrscheinlichkeit eintreten.

Exkurs: unabhängige Zufallsvariablen

Definition: Zwei Zufallsvariablen X und Y sind unabhängig, falls für alle $x, y \in \mathbb{R}$ gilt:

$$\text{Prob}[X = x \wedge Y = y] = \text{Prob}[X = x] \cdot \text{Prob}[Y = y]$$

Satz (ohne Beweis): Seien X und Y unabhängige Zufallsvariablen. Dann gilt: $E[X \cdot Y] = E[X] \cdot E[Y]$

Satz: Seien $X_1, X_2, \ldots, X_m$ unabhängige Zufallsvariablen. Sei $X = \sum_{i=1}^{m} X_i$.

Dann gilt: $\sigma_X^2 = \sum_{i=1}^{m} \sigma_{X_i}^2$ (*)

Beweis: Sei $\mu_i = E[X_i]$,

$\mu = \sum_{i=1}^{m} \mu_i = E[X]$

Es gilt: $\sigma_X^2 = E[(X-\mu)^2] =$

$= E\left[\left(\sum_{i=1}^{m}(X_i-\mu_i)\right)^2\right] \overset{\text{Linearity of expectations}}{=}$

$= \sum_{i=1}^{m} E[(X_i-\mu_i)^2] + 2\cdot\sum_{i<j} E[(X_i-\mu_i)(X_j-\mu_j)]$

unabhängig, da X_i und X_j unabhängig.

$= \sum_{i=1}^{m} \sigma_{X_i}^2 +$

$2\cdot\sum_{i<j} E[X_i-\mu_i]\cdot\underbrace{E[X_j-\mu_j]}_{= \underbrace{E[X_j]}_{\mu_j} - \mu_j = 0}$

$= \sum_{i=1}^{m} \sigma_{X_i}^2$ $\square$

<u>Zur Möglichkeit A</u>:

<u>Behauptung (1)</u>

$$\text{Prob}[a > S_{(k)}] \in O(m^{-\frac{1}{4}})$$

<u>Beweis</u>: $a > S_{(k)}$

$\Longleftrightarrow$

$R_{(l)} > S_{(k)}$

$\Longleftrightarrow$

Weniger als l der samples in R sind $\leq S_{(k)}$

- Sei $X_i := \begin{cases} 1 & \text{falls (i-te sample in } R) \leq S_{(k)} \\ 0 & \text{sonst} \end{cases}$

$\rightarrow \text{Prob}[X_i = 1] = \frac{k}{m}$

$\text{Prob}[X_i = 0] = 1 - \frac{k}{m}$

Sei $X := \sum_{i=1}^{m^{3/4}} X_i$ = Anzahl der samples in R, die $\leq S_{(k)}$ sind.

$$\rightarrow \mu_X = m^{3/4} \cdot E[X_i] = \frac{1}{m} \cdot k \cdot m^{3/4}$$

$$= k \cdot m^{-\frac{1}{4}} \overset{\nabla}{\overset{\circ}{=}} x$$

$$\sigma_X^2 \overset{(*)}{=} \sum_{i=1}^{m^{3/4}} \sigma_{X_i}^2$$

• <u>Lemma</u>: Sei Y eine Zufallsvariable mit $\text{Prob}[Y=1] = p$ und $\text{Prob}[Y=0] = 1-p$ (<u>Bernoulli Verteilung</u>).

Dann gilt $\sigma_Y^2 = p \cdot (1-p)$

<u>Beweis</u>: Übung

• Also gilt: $\sigma_X^2 = m^{3/4} \cdot \left(1 - \frac{k}{m}\right) \cdot \frac{k}{m}$

$$\leq \frac{1}{4} \cdot m^{3/4}$$

(beachte: die Funktion $(1-x) \cdot x$ wird im Intervall $[0,1]$ maximal für $x = \frac{1}{2}$)

$$\Rightarrow \sigma_X \leq \frac{1}{2} \cdot m^{3/8}$$

$$\xrightarrow{\text{Chebyshev}} \text{Prob}[|X - \overset{x}{\overset{\shortparallel}{\mu_X}}| \geq \sqrt{m}] \leq$$

$$\leq \text{Prob}[|X - \underset{x}{\underset{\shortparallel}{\mu_X}}| \geq \underbrace{2 \cdot m^{1/8} \cdot \sigma_X}_{\leq \sqrt{m}}] \leq$$

• $$\leq \frac{1}{(2 \cdot m^{1/8})^2} \in O(m^{-\frac{1}{4}}) \quad (**)$$

Nun schätzen wir $\text{Prob}[a > S_{(k)}]$ ab:

$$\text{Prob}[a > S_{(k)}] = \text{Prob}[X < \ell] =$$

• $$= \text{Prob}[X < \max\{1, \lfloor x - \sqrt{m} \rfloor\}]$$

<u>1. Fall</u>: $\lfloor x - \sqrt{m} \rfloor \geq 1$

Dann gelten folgende Implikationen:

$$X < \ell \Leftrightarrow X < \lfloor x - \sqrt{m} \rfloor \Rightarrow$$

$$X < x - \sqrt{m} \Rightarrow X - x < -\sqrt{m}$$

$\Rightarrow |X - x| \geq \sqrt{m}$

$\overset{(**)}{\Longrightarrow}$ $\text{Prob}[X < l] \leq \text{Prob}[|X - x| \geq \sqrt{m}]$

$\in \mathcal{O}(m^{-1/4})$

<u>2. Fall</u>: $\lfloor x - \sqrt{m} \rfloor < 1$

$\Rightarrow l = 1$

$\Rightarrow \text{Prob}[X < l] = \text{Prob}[X = 0]$

$= \left(1 - \frac{k}{m}\right)^{m^{3/4}}$

$\leq e^{-k \cdot m^{-1/4}} \underset{\substack{\uparrow \\ k \geq m^{1/4}}}{\leq} e^{-1} \in \mathcal{O}(m^{-\frac{1}{4}})$

Dies beendet den Beweis von Behauptung (1)

Analog zeigt man:

$\text{Prob}[b < S_{(k)}] \in \mathcal{O}(m^{-\frac{1}{4}})$

Damit ergibt sich insgesamt:

$\underline{\text{Prob}[S_{(k)} \notin P]} =$

$= \text{Prob}[a > S_{(k)} \vee b < S_{(k)}] \in \underline{\mathcal{O}(m^{-\frac{1}{4}})}$

<u>Zur Möglichkeit B</u>:

<u>Behauptung (2)</u>

$\text{Prob}[|P| > 4 \cdot m^{3/4} + 2m^{1/4} - 1] \in \mathcal{O}(m^{-\frac{1}{4}})$

<u>Beweis von Behauptung (2)</u>:

Definiere $k_l := \max\{1, k - 2 \cdot m^{3/4} - m^{1/4} + 1\}$

$k_h := \min\{k + 2 \cdot m^{3/4} + m^{1/4} - 1, m\}$

Wegen $P = \{y \in S \mid a \le y \le b\}$ gelten folgende Implikationen:

$$a \ge S_{(k_\ell)} \wedge b \le S_{(k_h)}$$

$$\Rightarrow |P| \le k_h - k_\ell + 1$$

$$\le k + 2 \cdot m^{3/4} + m^{1/4} - 1 - k + 2m^{3/4} + m^{1/4} - 1 + 1$$

$$= 4 \cdot m^{3/4} + 2m^{1/4} - 1$$

D.h.: $\left(|P| > 4 \cdot m^{3/4} + 2m^{1/4} - 1\right) \to$

$$\left(a < S_{(k_\ell)} \text{ oder } b > S_{(k_h)}\right)$$

$$\Rightarrow \text{Prob}\left[|P| > 4 \cdot m^{3/4} + 2m^{1/4} - 1\right]$$

$$\le \text{Prob}\left[a < S_{(k_\ell)}\right] + \text{Prob}\left[b > S_{(k_h)}\right]$$

<u>Wir zeigen nun</u>: $\text{Prob}\left[a < S_{(k_\ell)}\right] \in O(m^{-\frac{1}{4}})$

analog kann auch $\text{Prob}\left[b > S_{(k_h)}\right] \in O(m^{-\frac{1}{4}})$ gezeigt werden.

Es gilt: $a < S_{(k_\ell)}$

$$\Longleftrightarrow$$

$$R_{(\ell)} < S_{(k_\ell)}$$

$$\Longleftrightarrow$$

mindestens ℓ samples in R sind $< S_{(k_\ell)}$.

$$X_i := \begin{cases} 1 & \text{falls (i-te sample)} < S_{(k_\ell)} \\ 0 & \text{sonst} \end{cases}$$

$$X := \sum_{i=1}^{m^{3/4}} X_i$$

$E[X_i] = \text{Prob}[X_i = 1] = \frac{k_\ell - 1}{m}$

Also: $a < S_{(k_\ell)} \Leftrightarrow X \ge \ell$

$$\mu_X = m^{3/4} \frac{k_\ell - 1}{m} = m^{-\frac{1}{4}} (k_\ell - 1)$$

$\sigma_X \le \frac{1}{2} m^{3/8}$ (folgt wie bei Behauptung (1))

$\xrightarrow{\text{Chebyshev}} \operatorname{Prob}\left[|X-\mu_X| \geq \sqrt{n}\right] \in \mathcal{O}(n^{-\frac{1}{4}})$

<u>1. Fall</u>: $1 > k - 2\cdot n^{3/4} - n^{1/4} + 1$

$\Rightarrow k_\ell = \max\{1, k - 2\cdot n^{3/4} - n^{1/4} + 1\}$

$\Rightarrow \operatorname{Prob}[a < S_{(k_\ell)}]$

$= \operatorname{Prob}[a < S_{(1)}]$

$= \operatorname{Prob}[\text{mind. } \ell \text{ samples in } R \text{ sind} < S_{(1)}]$

$= 0 \in \mathcal{O}(n^{-\frac{1}{4}})$

<u>2. Fall</u>: $1 \leq k - 2\cdot n^{3/4} - n^{1/4} + 1$

$\Rightarrow k_\ell = k - 2\cdot n^{3/4} - n^{1/4} + 1$

$\Rightarrow \mu_X = n^{-\frac{1}{4}}(k - 2\cdot n^{\frac{3}{4}} - n^{1/4})$

Es gelten nun folgende Implikationen:

$a < S_{(k_\ell)} \Rightarrow X \geq \ell$

$\Rightarrow X \geq \max\{1, \lfloor x - \sqrt{n}\rfloor\}$

$\Rightarrow X \geq \lfloor x - \sqrt{n}\rfloor$

$\Rightarrow X \geq x - \sqrt{n} - 1 =$

$= k\cdot n^{-\frac{1}{4}} - \sqrt{n} - 1$

$\Rightarrow X + 2\cdot\sqrt{n} - k\cdot n^{-\frac{1}{4}} + 1 \geq \sqrt{n}$

$\Rightarrow \underbrace{X - n^{-1/4}(k - 2\cdot n^{3/4} - n^{1/4})}_{X - \mu_X} \geq \sqrt{n}$

$\Rightarrow |X - \mu_X| \geq \sqrt{n}$

also: $\operatorname{Prob}[a < S_{(k_\ell)}] \leq$

$\operatorname{Prob}[|X - \mu_X| \geq \sqrt{n}] \in \mathcal{O}(n^{-\frac{1}{4}})$

Satz: Mit Wahrscheinlichkeit $1-O(n^{-\frac{1}{4}})$ findet LazySelect $S_{(k)}$ im ersten Durchlauf von (1) – (5). In diesem Fall ist die Laufzeit $2n + o(n)$.

Wahrscheinlichkeitsverstärkung mittels Zwei-Punkt-Sampling

Zufallsvariablen $X_1, X_2, \dots, X_m$ sind paarweise unabhängig falls für alle $i,j \in \{1,\dots,m\}$ mit $i \neq j$ gilt

$$\text{Prob}[X_i = x \mid X_j = y] = \text{Prob}[X_i = x]$$

Im diesem Fall gilt für $X = X_1 + X_2 + \dots + X_m$

$$\sigma_X^2 = \sum_{i=1}^{m} \sigma_{X_i}^2$$

Bei der Derandomisierung von Lubys Algorithmus haben wir eigentlich folgendes gezeigt:

Satz: Sei p eine Primzahl. Wähle a und b unabhängig und gleichverteilt aus $\mathbb{F}_p = \{0, 1, \dots, p-1\}$ aus.
Für $i \in \{0, \dots, p-1\}$ sei $Y_i = a \cdot i + b \mod p$.
Dann ist auch Y_i gleichverteilt über $\mathbb{F}_p$ ($\text{Prob}[Y_i = y] = \frac{1}{p}$ für alle $y \in \mathbb{F}_p$) und $Y_0, Y_1, \dots, Y_{p-1}$ sind paarweise unabhängig.

Zur Erinnerung: RP ist die Klasse aller Sprachen L für die es einen polynomiellen Monte Carlo Algorithmus A gibt mit:

- $x \in L \Rightarrow \text{Prob}[A \text{ akzeptiert } x] \geq \frac{1}{2}$
- $x \notin L \Rightarrow \text{Prob}[A \text{ akzeptiert } x] = 0$

Wir stellen uns den Algorithmus A im folgenden wie folgt vor:

(1) A wählt zufällig und gleichverteilt eine Zahl $r \in \{0, \dots, n-1\}$.
(werfe alle Münzen zu Beginn und sei $\{0, \dots, n-1\}$ der entstehende Wahrscheinlichkeitsraum).

Wegen Bertrands Postulat können wir außerdem davon ausgehen, dass n eine Primzahl ist.

(2) A berechnet deterministisch ein Bit $A(x, r)$ (in Polynomialzeit).

Dabei gilt:

- $x \in L \Rightarrow$ Für mind. Hälfte aller $r \in \mathbb{F}_n$ gilt $A(x,r) = 1$
- $x \notin L \Rightarrow A(x,r) = 0$ für alle $r \in \mathbb{F}_n$

<u>Beachte:</u> n kann exponentiell mit der Inputlänge $|x|$ wachsen.
Aber: nur $\log(n)$ viele Zufallsbits sind notwendig, um r zufällig und gleichverteilt aus $\mathbb{F}_n$ auszuwählen.

- Für $x \in L$ ist die Fehlerwahrscheinlichkeit $\left(= \text{Prob}[A \text{ akzeptiert } x \text{ nicht}]\right) \leq \frac{1}{2}$

Wir wollen nun die Fehlerwahrscheinlichkeit auf $\frac{1}{t}$ für ein $t \leq n$ reduzieren.

<u>Methode 1</u> Naive Wahrscheinlichkeitsverstärkung.

Wir wiederholen Algorithmus A m mal.
Gibt A bei einer Wiederholung 1 aus, so geben wir insgesamt 1 aus, sonst geben wir 0 aus.

Für $x \in L$ ist die Fehlerwahrscheinlichkeit jetzt $\leq \frac{1}{2^m}$

Setze $m := \lceil \log(t) \rceil$

→ Fehlerwahrscheinlichkeit $\leq \frac{1}{t}$.

Unser neuer Monte Carlo Algorithmus benötigt insgesamt

$$m \cdot \log(n) = \lceil \log(t) \rceil \cdot \log(n)$$

viele Zufallsbits.

Methode 2 Zwei-Punkt-Sampling

(1) Wähle a und b unabhängig und gleichverteilt aus $\mathbb{F}_n$ aus.

↳ $2 \cdot \log(n)$ Zufallsbits notwendig

(2) Berechne $r_i := a \cdot i + b \mod n$ sowie $A(x, r_i) \in \{0,1\}$ für $0 \leq i \leq t-1$

(3) Ist ein $A(x, r_i)$ gleich 1, so geben wir insgesamt 1 aus, sonst geben wir 0 aus.

Sei B dieser neue Monte Carlo Algorithmus.

- $x \notin L \Rightarrow \text{Prob}[B \text{ akzeptiert } x] = 0$

- Behauptung: Für $x \in L$ gilt $\text{Prob}[B \text{ akzeptiert } x \text{ nicht}] \leq \frac{1}{t}$

Beweis: Sei $Y := \sum_{i=0}^{t-1} A(x, r_i)$

↳ $\text{Prob}[B \text{ akzeptiert } x \text{ nicht}] = \text{Prob}[Y = 0]$

- Die Zufallsvariablen $r_i = a \cdot i + b \mod n$ sind gleichverteilt und paarweise unabhängig!

→ Auch die Zufallsvariablen $A(x, r_i)$ $(0 \leq i \leq t-1)$ sind paarweise unabhängig (beweisen Sie dies als Übung).

- Sei nun $x \in L$.

$$\Rightarrow E[Y] = \sum_{i=0}^{t-1} E[A(x, r_i)] = \sum_{i=0}^{t-1} \text{Prob}[A(x, r_i) = 1]$$

$$\geq \sum_{i=0}^{t-1} \frac{1}{2} = \frac{t}{2}$$

v_i gleichverteilt aus $\mathbb{F}_n$

$$\sigma_Y^2 = \sum_{i=0}^{t-1} \sigma_{A(x,v_i)}^2$$

$v_0, \dots, v_{t-1}$ paarweise unabhängig.

$$\sigma_{A(x,v_i)}^2 = \text{Prob}[A(x,v_i)=0] \cdot \text{Prob}[A(x,v_i)=1]$$

$$= \left(1 - \text{Prob}[A(x,v_i)=1]\right) \cdot \text{Prob}[A(x,v_i)=1]$$

$$\leq \frac{1}{4}$$

$$\Rightarrow \sigma_Y^2 \leq \frac{t}{4} \Rightarrow \sigma_Y \leq \frac{\sqrt{t}}{2}$$

$$\Rightarrow \text{Prob}[Y=0] \leq \text{Prob}\left[|Y - E[Y]| \geq \frac{t}{2}\right]$$

$$\leq \text{Prob}\left[|Y - E[Y]| \geq \sqrt{t} \cdot \sigma_Y\right]$$

$$\leq \frac{1}{t}$$

Chebyshev

$\square$

Stabile Matchings

Gegeben: n Frauen $(a, b, c, \ldots)$ und
n Männer $(x, y, z, \ldots)$
Jede Frau a (Mann x) hat eine Präferenzliste, welche alle Männer (Frauen) enthält.

- Für eine Person p (Mann oder Frau) und zwei potentielle Partner p_1 und p_2 schreiben wir $\text{präf}_p(p_1) > \text{präf}_p(p_2)$ falls p_1 in der Präferenzliste von p vor p_2 steht (p bevorzugt p_1 gegenüber p_2)

- Ein Matching ist eine 1-1-Zuordnung von Männern zu Frauen (jede Person ist mit genau einem Partner verheiratet).

Wir schreiben $\text{partner}(p) = q$ falls p mit q verheiratet ist.

- Ein unzufriedenes Paar ist ein Paar (x, a) ($x \in$ Männer, $a \in$ Frauen) mit:
  - $\text{präf}_x(a) > \text{präf}_x(\text{partner}(x))$
  - und $\text{präf}_a(x) > \text{präf}_a(\text{partner}(a))$

- Ein Matching ist stabil, falls kein unzufriedenes Paar existiert.

Frage: Existiert für beliebe Präferenzlisten ein stabiles Matching?

Beispiel:

$x: a\,b\,c\,d$ $a: x\,y\,z\,z'$

$y: b\,a\,c\,d$ $b: z'\,z\,y\,x$

$z: a\,d\,c\,b$ $c: x\,y\,z\,z'$

$z': d\,c\,a\,b$ $d: z\,z'\,x\,y$

Das folgende Matching ist nicht stabil

x —— a
y —— b
z —— c
z' —— d

(z, d) ist ein unzufriedenes Paar.

Das folgende Matching ist stabil:

x —— a
y —— b
z —— d
z' —— c

Wie findet man ein stabiles Matching?

Proposal-Algorithmus

Männer machen Angebote, Frauen lehnen ab.

forall $x \in$ Männer do
    Kandidaten(x) = alle Frauen
endfor

while "es existiert unverheirateter Mann" do

- sei x ein unverheirateter Mann;
- sei Frau $a \in$ Kandidaten(x) so, dass: $\text{präf}_x(a) \geq \text{präf}_x(b)$ für alle $b \in$ Kanditaten(x).
- if "a ist ledig oder $\text{präf}_a(x) > \text{präf}_a(\text{partner}(a))$"
  then

partner(a) := x , partner(x) := a
(der alte Partner von a (~~falls~~ → ~~nicht ledig~~ war) ist jetzt ledig)

endif
Kandidaten(x) := Kandidaten(x) \ {a}
endwhile

<u>Satz</u>: Der Proposal-Algorithmus terminiert nach höchstens n^2 (n = Anzahl Männer = Anzahl Frauen) Durchläufen durch die while-Schleife mit einem stabilen Matching.

<u>Beweis</u>:

(1) Ist eine Frau a irgendwann verheiratet, so bleibt sie es auch (gilt nicht für Männer)
Außerdem nimmt die Präferenz ihrer Partner nur zu:

x_1 = partner(a) zum Zeitpunkt t_1
x_2 = partner(a) zum Zeitpunkt $t_2 \geq t_1$
$\Rightarrow \text{präf}_a(x_2) \geq \text{präf}_a(x_1)$

D.h. in jedem Schleifendurchlauf heiratet eine Frau zum ersten mal, oder sie heiratet einen von ihr bevorzugten Partner.

(2) Gilt für einen Mann x irgendwann Kandidaten(x) $= \emptyset$, so muß der Algorithmus terminieren:

Kandidaten(x) $= \emptyset \Rightarrow$
Alle Frauen sind verheiratet (wegen (1)). $\Rightarrow$

Alle Männer sind verheiratet.

Da in jedem Schleifendurchlauf für einen Mann die Kandidaten-Menge kleiner wird, und es am Anfang n Kandidatenmengen der Größe n gibt, folgt:

Proposal-Algorithmus terminiert nach höchstens n^2 Schleifendurchläufen

(3) Behauptung: Das berechnete Matching ist stabil:

Annahme: (x, a) ist am Ende ein unzufriedenes Paar. $\Rightarrow$

- $\text{präf}_x(a) > \text{präf}_x(\underbrace{\text{partner}(x)}_{b})$

- $\text{präf}_a(x) > \text{präf}_a(\underbrace{\text{partner}(a)}_{y})$

wegen $\text{präf}_x(a) > \text{präf}_x(b)$ muß Mann x der Frau a ein Angebot gemacht haben (etwa zum Zeitpunkt t) bevor er Frau $b = \text{partner}(x)$ geheiratet hat.

Dann muß

a) Frau a Mann x zum Zeitpunkt t geheiratet haben und sich später von x geschieden haben (a ist am Ende mit $y \neq x$ verheiratet)

<u>oder</u>

(b) Frau a zum Zeitpunkt t mit einem Mann z verheiratet gewesen sein, für den gilt $\text{präf}_a(z) > \text{präf}_a(x)$

Im Fall (a) gilt wegen (1):

$$\text{präf}_a(x) \leq \text{präf}_a(y) \ \lightning$$

Partner von a am Ende.

Im Fall (b) folgt analog mit (1):

$$\text{präf}_a(x) < \text{präf}_a(z) \leq \text{präf}_a(y) \ \lightning$$

$\square$

Im weiteren soll das Average-Case Verhalten des Proposal-Algoritmus betrachtet werden.

<u>Annahme</u>:

1. Die Präferenzlisten der Frauen sind fest aber beliebig gewählt.
2. Die Präferenzlisten der Männer werden zufällig und gleichverteilt gewählt.

Sei Zufallsvariable $T_P :=$ Anzahl der Angebote vom Männer an Frauen im Proposal-Algorithmus unter der oben genannten Inputverteilung.

<u>1. Vereinfachung</u>:

Anstatt die Präferenzlisten der Männer zu Beginn zufällig zu wählen, gehen wir davon aus, dass:
Im jedem Schritt ein Mann x eine Frau a zufällig aus Kandidaten(x) auswählt und a ein Angebot macht.

<u>Prinzip der verzögerten Entscheidung</u>

Anstatt zu Beginn den gesamten Input zufällig zu wählen, legen wir in jedem Schritt nur soviel zufällig von der Eingabe frei, wie für die Durchführung dieser Schritts notwendig ist.

2. Vereinfachung: Wir lassen den Schritt "Kandidaten(x) := Kandidaten(x) \ {a}" im Proposal-Algorithmus weg.

D.h. in jedem Schritt wählt ein lediger Mann x zufällig und gleichverteilt eine Frau a aus und macht a ein Angebot.

Es ist dabei möglich, dass a den Mann x bereits zu einem früheren Zeitpunkt abgewiesen hat.

↳ vergessliche Algorithmus

Zufallsvariable T_V := Anzahl der Angebote von Männer an Frauen im vergesslichen Algorithmus.

Beobachtungen

- Weist Frau a einen Mann x zum Zeitpunkt t ab, so wird sie dies auch zum Zeitpunkt $t' > t$ tun.

  ⇒ Proposal-Algorithmus und vergesslicher Algorithmus produzieren gleiches Matching.

- $E[T_P] \leq E[T_V]$

  $Prob[T_P > m] \leq Prob[T_V > m]$

- Angenommen in einem Ablauf des vergesslichen Algorithmus hat zu einem gewissen Zeitpunkt jede Frau mindestens ein Angebot erhalten.

  ⇒ Jede Frau ist verheiratet

  ⇒ Jeder Mann ist verheiratet

  ⇒ Algorithmus terminiert.

Coupon-Sammler Problem

Gegeben: n Typen von Coupons. Seien die Typen $1, 2, \dots, n$

Im jedem Schritt wird ein Coupon eines zufällig und gleichverteilt ausgewählten Typs gezogen.

Zufallsvariable X = Anzahl der Schritte, bis von jedem Typ mindestens ein Coupon gezogen wurde.

$\hookrightarrow E[X] = E[T_v]$, $\text{Prob}[X > m] = \text{Prob}[T_v > m]$

Sei $C_1, C_2, \dots, C_X$ die Folge der in jedem Schritt ausgewählten Coupons-Typen.

$C_i \in \{1, \dots, n\}$

- C_i ist ein Erfolg, falls $C_i \notin \{C_1, \dots, C_{i-1}\}$

$\hookrightarrow C_1$ und C_X sind Erfolge

C_i ist Erfolg $\Leftrightarrow$ Im i-ten Schritt wird neuer Coupon-Typ ausgewählt.

- Seien $\underset{\substack{\| \\ C_1}}{C_{i_1}}, C_{i_2}, \dots, \underset{\substack{\| \\ C_X}}{C_{i_n}}$ die Erfolge

$(1 = i_1 < i_2 < \dots < i_n = X)$

$X_k := i_{k+1} - i_k$ für $0 \leq k < n$, wobei $i_0 := 0$

X_k ist die Anzahl der Schritte nach dem k-ten Erfolg bis inklusive dem $(k+1)$-ten Erfolg.

$$\Rightarrow \sum_{k=0}^{n-1} X_k = i_n - i_0 = X$$

$$\text{Prob}[X_k = \ell] = \left(1 - \frac{n-k}{n}\right)^{\ell-1}\left(\frac{n-k}{n}\right)$$

$$= q^{\ell-1}\cdot(1-q) \text{ für } q = 1 - \frac{n-k}{k} = \frac{k}{n}$$

(geometrische Verteilung)

$$\Rightarrow E[X_k] = \sum_{\ell=1}^{\infty} q^{\ell-1}(1-q)\cdot \ell$$

$$= \sum_{\ell=0}^{\infty} q^{\ell}(\ell+1) - \sum_{\ell=1}^{\infty} q^{\ell}\cdot \ell$$

$$= \sum_{\ell=0}^{\infty} q^{\ell} = \frac{1}{1-q} = \frac{n}{n-k}$$

$$\Rightarrow E[X] = \sum_{k=0}^{n-1} E[X_k] = \sum_{k=0}^{n-1} \frac{n}{n-k} =$$

$$= n\cdot \sum_{k=0}^{n-1} \frac{1}{n-k} = n\cdot \sum_{k=1}^{n} \frac{1}{k} = n\cdot H_n$$

$$\Rightarrow E[X] \in n\cdot \overset{\ln}{\cancel{\log}}(n) + O(n)$$

<u>Satz 1</u>: Sei n = Anzahl verschiedener Coupon-Typen. Dann ist die mittlere Anzahl von Schritten, bis jeder Coupon-Typ mindestens einmal gezogen wurde $\in n\cdot \overset{\ln}{\cancel{\log}}(n) + O(n)$

<u>Korollar</u>: Die mittlere Anzahl von Angeboten im Proposal-Algorithmus (bei zufällig ausgewählten Präferenzlisten für die Männer) ist $\leq n\cdot \overset{\ln}{\cancel{\log}}(n) + O(n)$

Wie weit weicht X von $E[X]$ ab?

Satz 2: Sei n = Anzahl der Coupon-Typen, X = Anzahl der Schritte bis von jedem Typ mindestens ein Coupon gezogen wurde. Sei $c \in \mathbb{R}$ Konstante.

Sei $m = n \cdot \ln(n) + c \cdot n$

$$\Rightarrow \lim_{n \to \infty} \mathrm{Prob}[X > m] = 1 - e^{-(e^{-c})}$$

Zum Beweis von Satz 2 benötigen wir einige Hilfsmittel.

Lemma 1 Sei $c \in \mathbb{R}$ eine Konstante und sei $k \in \mathbb{N}$ fest gewählt.

Sei $m = n \cdot \ln(n) + c \cdot n$

$$\Rightarrow \lim_{n \to \infty} \binom{n}{k}\left(1 - \frac{k}{n}\right)^m = \frac{e^{-c \cdot k}}{k!}$$

Beweis von Lemma ~~2~~ 1

O.B.d.A sei $k^2 \leq n$ im folgenden.

Konstante

Für $n \geq 1$, $t \in \mathbb{R}$ mit $|t| \leq n$ gilt

$$e^{-t} \cdot \left(1 - \frac{t^2}{n}\right) \leq \left(1 - \frac{t}{n}\right)^n \leq e^{-t} \quad (*)$$

Hieraus folgt:

$$\left(1 - \frac{t}{n}\right)^{\ln(n)} = \left(1 - \frac{t}{n}\right)^{n \cdot \frac{\ln(n)}{n}}$$

$$\geq \underbrace{e^{-t \cdot \frac{\ln(n)}{n}}}_{\substack{\to 1 \\ \text{für } n \to \infty}} \cdot \underbrace{\left(1 - \frac{t^2}{n}\right)^{\frac{\ln(n)}{n}}}_{\substack{\to 1 \\ \text{für } n \to \infty}} \quad (**)$$

- Aus $(*)$ folgt für $\left(1 - \frac{k}{n}\right)^m = \left(1 - \frac{k}{n}\right)^{n \cdot \frac{m}{n}}$:

$$e^{-k \cdot \frac{m}{n}}\left(1 - \frac{k^2}{n}\right)^{\frac{m}{n}} \leq \left(1 - \frac{k}{n}\right)^m \leq e^{-k \cdot \frac{m}{n}}$$

• Wegen $e^{-k\cdot\frac{m}{n}} = e^{-k\cdot(\ln(n)+c)} =$

$= \frac{1}{n^k}\cdot e^{-k\cdot c}$ folgt:

$$\frac{e^{-k\cdot c}}{k!}\cdot\frac{n(n-1)\cdots(n-k+1)}{n^k}\left(1-\frac{k^2}{n}\right)^{\frac{m}{n}} \leq$$

$$\binom{n}{k}\cdot\left(1-\frac{k}{n}\right)^{m} \leq$$

$$\cdot\frac{e^{-k\cdot c}}{k!}\;\frac{n(n-1)\cdots(n-k+1)}{n^k} \qquad (***)$$

• $\left(1-\frac{k^2}{n}\right)^{\frac{m}{n}} = \left(1-\frac{k^2}{n}\right)^{c+\ln(n)} =$

$$\left(1-\frac{k^2}{n}\right)^{c}\left(1-\frac{k^2}{n}\right)^{\ln(n)} \overset{(**)\text{ mit } k^2\leq n}{\geq}$$

$$\left(1-\frac{k^2}{n}\right)^{c}\cdot e^{-k^2\cdot\frac{\ln(n)}{n}}\cdot\left(1-\frac{k^4}{n}\right)^{\frac{\ln(n)}{n}}$$

Hieraus folgt mit (***)

$$\frac{e^{-k\cdot c}}{k!}\cdot\underbrace{\frac{n\cdot\ldots\cdot(n-k+1)}{n^k}\cdot\left(1-\frac{k^2}{n}\right)^{c}\cdot e^{-k^2\cdot\frac{\ln(n)}{n}}\cdot\left(1-\frac{k^4}{n}\right)^{\frac{\ln(n)}{n}}}_{\to 1 \text{ für } n\to\infty}$$

$$\leq \binom{n}{k}\cdot\left(1-\frac{k}{n}\right)^{m} \leq$$

$$\frac{e^{-k\cdot c}}{k!}\cdot\underbrace{\frac{n\cdot\ldots\cdot(n-k+1)}{n^k}}_{\to 1 \text{ für } n\to\infty}$$

$$\Rightarrow \lim_{n\to\infty}\binom{n}{k}\left(1-\frac{k}{n}\right)^{m} = \frac{e^{-k\cdot c}}{k!} \qquad \square$$

Inklusions-Exklusions-Prinzip

Seien $\mathcal{E}_1, \dots, \mathcal{E}_m$ beliebige Ereignisse. Dann gilt:

$$\mathrm{Prob}\Big[\bigcup_{i=1}^{m} \mathcal{E}_i\Big] = \sum_{k=1}^{m} (-1)^{k+1} \cdot P_k \quad \text{mit}$$

$$P_k = \sum_{1 \le i_1 < i_2 < \dots < i_k \le m} \mathrm{Prob}\Big[\bigcap_{j=1}^{k} \mathcal{E}_{i_j}\Big]$$

Boole-Bonferroni-Ungleichung

$$\sum_{k=1}^{2\cdot l} (-1)^{k+1} \cdot P_k \le \mathrm{Prob}\Big[\bigcup_{i=1}^{m} \mathcal{E}_i\Big] \le$$

$$\sum_{k=1}^{2\cdot l+1} (-1)^{k+1} \cdot P_k \quad \text{falls } 2\cdot l+1 \le m$$

Beweis vom Satz 2:

- Sei $\mathcal{E}_i^r$ das Ereignis, dass Coupon-Typ i nicht in den ersten r vielen Schritten gezogen wird.

$$\mathrm{Prob}[\mathcal{E}_i^r] = \Big(1 - \frac{1}{m}\Big)^r$$

- $X > n \Leftrightarrow \bigcup_{i=1}^{m} \mathcal{E}_i^n$

$$\text{Sei } \underset{(k \le m)}{P_k^n} := \sum_{1 \le i_1 < i_2 < \dots < i_k \le m} \mathrm{Prob}\Big[\bigcap_{j=1}^{k} \mathcal{E}_{i_j}^n\Big]$$

Symmetrie $\longrightarrow$

$$P_k^n = \binom{m}{k} \cdot \mathrm{Prob}\Big[\bigcap_{j=1}^{k} \mathcal{E}_j^n\Big]$$

$$= \binom{m}{k}\Big(1 - \frac{k}{m}\Big)^n$$

- Für jedes feste $k \in \mathbb{N}$ gilt nach Lemma 2:

$$\lim_{n\to\infty} P_k^n = \frac{e^{-c\cdot k}}{k!} =: P_k$$

- Für $l \leq n$ sei $S_l^n := \sum_{j=1}^{l} (-1)^{j+1} \cdot P_j^n$

Boole-Bonferroni-Ungleichung $\longrightarrow$

(*) $S_{2\cdot l}^n \leq \mathrm{Prob}\left[\bigcup_{i=1}^{n} \mathcal{E}_i^n\right] \leq S_{2\cdot l+1}^n$

- Für $l \leq n$ sei $S_l := \sum_{j=1}^{l} (-1)^{j+1} \cdot P_j$

$$= \sum_{j=1}^{l} (-1)^{j+1} \cdot \frac{e^{-c\cdot j}}{j!}$$

$\longrightarrow \lim_{l\to\infty} S_l = f(c) := 1 - e^{-(e^{-c})}$

$\longrightarrow$ Für jedes $\varepsilon > 0$ existiert $l^* > 0$ mit:

$$\forall\, l > l^* : \left| S_l - f(c) \right| < \varepsilon$$

- Wegen $\lim_{n\to\infty} P_k^n = P_k$ (für jedes feste k)

gilt: $\lim_{n\to\infty} S_l^n = S_l$ (für jedes feste l)

$\longrightarrow$ Für jedes $\varepsilon > 0$ und jedes $l \in \mathbb{N}$ gilt $|S_l^n - S_l| < \varepsilon$ für genügend große n.

$\longrightarrow$ Für jedes $\varepsilon > 0$ existiert $l^* > 0$, so dass für jedes feste $l > l^*$ gilt: $\left| S_l^n - f(c) \right| < 2\cdot\varepsilon$ falls n genügend groß.

$\longrightarrow |S^m_{2\cdot l} - S^m_{2\cdot l+1}| < 4\cdot\varepsilon$ falls n genügend groß.

Mit (*) ergibt sich:

$$\Big|\underbrace{\mathrm{Prob}\Big[\bigcup_{i=1}^{m} \mathcal{E}_i^m\Big]}_{\mathrm{Prob}[X>m]} - f(c)\Big| < 4\varepsilon$$

falls n genügend groß.

$$\longrightarrow \lim_{n\to\infty} \mathrm{Prob}[X>m] = f(c) = 1-e^{-(e^{-c})} \quad \square$$

Aus Satz 2 folgt für jedes $c\in\mathbb{R}, c\geq 0$

$$\lim_{n\to\infty} \mathrm{Prob}[X > n\cdot(\ln(n)+c)] = 1-e^{-e^{-c}}$$

$$\lim_{n\to\infty} \mathrm{Prob}[X \leq n\cdot(\ln(n)-c)] = e^{-e^{c}}$$

$$\rightarrow \lim_{n\to\infty} \mathrm{Prob}[n(\ln(n)-c) < X \leq n(\ln(n)+c)]$$

$$= 1-\left(1-e^{-e^{-c}} + e^{-e^{c}}\right)$$

$$= \underbrace{e^{-e^{-c}} - e^{-e^{c}}}_{\text{geht sehr schnell gegen 1 für große } c.}$$