

**Verjetnostne metode v
računalništvu - zapiski s
predavanj prof. Marca**

Tomaž Poljanšek

študijsko leto 2023/24

Kazalo

1	Introduction	1
1.1	Probability	1
1.2	Random variables	2
2	Quicksort, min-cut	4
2.1	Quicksort	4
2.2	Min-cut	6
3	Complexity classes	9
4	Chernoff bounds	11

Poglavje 1

Introduction

1.1 Probability

(Ω, F, P_r) :

- $\emptyset \in F$,
- $A \in F \implies A^c \in F$,
- $A_1, A_2 \dots \in F \implies \cup_{i=1}^{\infty} A_i \in F$.

$P_r(A) \geq 0$,

$P_r(\cup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} P_r(A_i)$ if A_i disjoint,

$P_r(\cup_{i=1}^{\infty} A_i) \leq \sum_{i=1}^{\infty} P_r(A_i)$,

$\Omega = \{\omega_1, \omega_2 \dots\}$ - countable case.

$$\begin{pmatrix} \omega_1 & \omega_2 & \dots \\ p_1 & p_2 & \dots \end{pmatrix}$$

Primer.

`Alg():`

`while True:`

`B = sample as random from {0,1} # 1 with probability p`

`if B = 1:`

return

$$\Omega = \{1, 01, 001, 0001 \dots\}$$

$$\begin{pmatrix} 1 & 01 & 001 & 0001 & \dots \\ p & (1-p)p & (1-p)^2p & (1-p)^3p & \dots \end{pmatrix}.$$

1.2 Random variables

$X : \Omega \rightarrow \mathbb{Z}$.

$E[X] = \sum_{c \in \mathbb{Z}} c \cdot P_r(X = c)$ expected value of X .

Properties:

- $E[f(X)] = \sum_{c \in \mathbb{Z}} f(c) \cdot P_r(X = c)$,
- $E[aX + bY] = aE[X] + bE[Y]$,
- $E[X \cdot Y] = E[X] \cdot E[Y]$ if X, Y independent,
- $P_r(X \geq a) \leq \frac{E[X]}{a} \forall a > 0, X \geq 0$ Markov inequality.

Primer. (Continuing from before).

X = number of trials before return.

$X : \Omega \rightarrow \mathbb{Z}$.

$X : 1 \rightarrow 1, 01 \rightarrow 2, 003 \rightarrow 3 \dots$

$$\begin{pmatrix} 1 & 2 & 3 & 4 & \dots \\ p & (1-p)p & (1-p)^2p & (1-p)^3p & \dots \end{pmatrix} - \text{geometric distribution.}$$

Trditev 1.2.1. $E[X] = \frac{1}{p}$.

Dokaz 1.2.2. $X = \sum_{i=1}^{\infty} X_i$.

$$X_i = \begin{cases} 1 & \text{if trial } i \text{ is executed} \\ 0 & \text{else} \end{cases}$$

$$\begin{aligned} E[X] &= E\left[\sum_{i=1}^{\infty} X_i\right] = \sum_{i=1}^{\infty} E[X_i] = \\ &= \sum_{i=1}^{\infty} (1-p)^{i-1} = \frac{i=0}{\infty} (1-p)^i = \frac{1}{1-(1-p)} = \frac{1}{p}. \end{aligned}$$

$$E[X] = \frac{1}{p}.$$

$$P_r(X \geq 100 \cdot \frac{1}{p}) \leq \frac{E[X]}{\frac{1}{p}} = \frac{1}{100}.$$

Definicija 1.2.3. $H_n = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} = \sum_{i=1}^{\infty} \frac{1}{i}$.

Izrek 1.2.4. $H_n \leq 1 + \ln(n)$.

Dokaz 1.2.5.

$$H_n = 1 + \sum_{i=2}^n \frac{1}{i} \stackrel{\text{integral}}{\leq} 1 + \int_1^n \frac{dx}{x} = 1 + \ln(x)|_1^n = 1 + \ln(n).$$

Poglavje 2

Quicksort, min-cut

2.1 Quicksort

Input: set (no equal element) (unordered list) $S \in \mathbb{R}$
(or whatever you can compare linearly)

Output: ordered list

Code:

```
def Quicksort(S):  
    if |S| = 0 or 1:  
        return S  
    else:  
        a = uniformly at random from S  
         $S^- = \{b \in S \mid b < a\}$   
         $S^+ = \{b \in S \mid a < b\}$   
        return Quicksort( $S^-$ ), a, Quicksort( $S^+$ )
```

$C(n)$ - random variable, the number of comparisons in evaluation of Quicksort with $|S| = n$.

Izrek 2.1.1. $E[C(n)] = O(N \log(n))$.

Dokaz 2.1.2. $C(0) = C(1) = 0$.

$$\begin{aligned}
E[C(n)] &= n - 1 + \sum_{i=1}^n (E[C(i-1)] + E[C(n-i)]) \cdot P_r(a \text{ is } i\text{-it element}) \leq \\
&\leq n + \frac{2}{n} \sum_{i=1}^{n-1} E[C(i)].
\end{aligned}$$

Induction:

$n = 1 : \checkmark$

$n - 1 \rightarrow n$:

$$\begin{aligned}
E[C(n)] &\leq n + \frac{2}{n} \sum_{i=1}^n E[C(i)] \leq \\
&\leq n + \frac{2}{n} \sum_{i=1}^n 5i \log i \leq \\
&\leq n + \frac{2}{n} \sum_{i=1}^{\lfloor \frac{n}{2} \rfloor} 5i \log i + \frac{2}{n} \sum_{i=1+\lfloor \frac{n}{2} \rfloor}^{n-1} 5i \log i \leq \\
&\leq n + \frac{2}{n} \sum_{i=1}^{\lfloor \frac{n}{2} \rfloor} 5i \log \frac{n}{2} + \frac{2}{n} \sum_{i=1+\lfloor \frac{n}{2} \rfloor}^{n-1} 5i \log n \leq \\
&(\log \frac{n}{2} = \log n - 1) \\
&\leq n + \frac{2}{n} \left(\sum_{i=1}^n 5i \log n - \sum_{i=1}^{\frac{n}{2}} 5i \right) = \\
&= n + \frac{10}{n} \left(\frac{n(n-1)}{2} \log n - \frac{\frac{n}{2}(\frac{n}{2}+1)}{2} \right) \leq \\
&\leq n + 5(n-1) \log n - n < \\
&< 5n \log n.
\end{aligned}$$

$$P(C(n) \geq b \cdot 5n \log n) \stackrel{\text{Markov}}{\leq} \frac{1}{b}.$$

Dokaz 2.1.3.

2:

Let $S_1, S_2 \dots S_n$ sorted elements of S .

Define random variable $X_{ij} = \begin{cases} 1 & \text{if } S_i \text{ and } S_j \text{ are compared} \\ 0 & \text{else} \end{cases}$

$$C(n) = \sum_{1 \leq i < j \leq n} E[X_{ij}].$$

$$E[X_{ij}] = P(S_i \text{ and } X_j \text{ compared}).$$

S_{ij} - the last set including S_i and S_j .

$$E[X_{ij}] = \frac{2}{|S_{ij}|} \leq \frac{2}{j-i+1}.$$

$$|S_{ij}| \geq j - i + 1.$$

S_{ij} has everything in between.

$$\begin{aligned} \Rightarrow E[C(n)] &\leq \sum_{1 \leq i < j \leq n} \frac{2}{j-i+1} = \\ &= \sum_{k=j-i+1}^{n-1} \sum_{i=1}^{n-1} \frac{2}{k} \leq \\ &\leq 2 \cdot n \cdot H_n \leq \\ &\leq 2n(1 + \log n). \end{aligned}$$

2.2 Min-cut

G multigraph.

Cut: $U \subset V(G)$, $U \neq \emptyset, V(G)$.

$$(U, V(G) \setminus U) = \{uv \in E(G) \mid u \in U, v \in V(G) \setminus U\}.$$

Problem min-cut:

Input: G .

Output: $\min |(U, V(G) \setminus U)|$ - cut size.

Algorithm 1:

$x \in V(G)$

Call $\text{maxFlow}(G, x, y) \forall y \in V(G)$

Take \min

maxFlow is Edmonds-Karp algorithm $O(|V||E|^2)$.

Algorithm 2 (Stoer Wagner)

Is $O(|E||V| + |V|\log|V|)$.

Algorithm *randMinCut*:

```

G_0 = G
i = 0
while |V(G_i)| > 2:
    e_i = uniformly at random from G_i
    G_{i+1} = G_i / e_i
    i = i + 1
u, v = V(G_{n-2}) // n = |V(G)|
U = {w ∈ V(G) | w is merged into u}
return (U, V(G) \ U)

```

Izrek 2.2.1. Algorithm *randMinCut* gives you a minimal cut with probability greater or equal to $\frac{2}{n(n-1)}$.

Dokaz 2.2.2.

Fact 1: $\minCut(G_i) \leq \minCut(G)$;

\nexists : *minCut* remains.

Fact 2: $\minCut(G) \leq \delta(G)$.

$k := \minCut(G)$.

Let (A, B) be an optimal cut.

e_i not in (A, B) .

$$\begin{aligned}
 & P_r(\text{Algorithm not returning } (A, B)) \\
 &= P_r(e_0 \cap \dots \cap e_{n-3}) \\
 &= P_r(e_0 \cap \dots \cap e_{n-4}) \cdot P_r(e_{n-3} \mid e_0 \cap \dots \cap e_{n-4}) \\
 &= P_r(e_{n-3} \mid \cap_{i=0}^{n-4} e_i) \cdot P_r(e_{n-3} \mid \cap_{i=0}^{n-4} e_i) \\
 &\dots P_r(e_1 \mid e_0) \cdot P_r(e_0). (*)
 \end{aligned} \tag{2.1}$$

$$P_r(\bar{e}_i \mid e_{i-1} \cap \dots \cap e_0) = \frac{k}{|E(G_i)|} \stackrel{(**)}{\leq} \frac{k}{\frac{(n-i)k}{2}} = \frac{2}{n-i}$$

$$|E(G_i)| \geq \frac{(n-i)\delta(G)}{2} \geq \frac{(n-i)k}{2}. (**)$$

$$P_r(\epsilon_i \mid \epsilon_{i-1} \cap \dots \cap \epsilon_0) \geq 1 - \frac{2}{n-i} = \frac{n-2-i}{n-i}.$$

$$(*) \geq \frac{n-2}{n} \cdot \frac{n-3}{n-1} \cdots \frac{1}{3} = \frac{2}{n(n-1)}.$$

Izrek 2.2.3. Running *randMinCut* $n(n-1)$ times and taking best output gives correct solution with probability ≥ 0.86 .

Dokaz 2.2.4. A_i - event that i -th run gives sub-optimal solution.

$$\begin{aligned} P_r(\text{solution not correct}) &= P_r(A_1 \cap \dots \cap A_{n(n-1)}) \\ &= \prod_{i=1}^{n(n-1)} P_r(A_i) \leq \left(1 - \frac{2}{n(n-1)}\right)^{n(n-1)} \\ &\leq e^{-\frac{2}{n(n-1)} \cdot n(n-1)} = e^{-2} \leq 0.14. \end{aligned}$$

$$1 - x \leq e^x \quad \forall x \in \mathbb{R}.$$

If we run $n(n-1)\log(n)$ times $\rightarrow O\left(\frac{1}{n}\right)$.

$O(n^2 \log n \cdot n)$.

Improved: $O(n^2 \log^3 n)$.

Poglavje 3

Complexity classes

Decision problem - yes/no question on a set of inputs = asking $w \in \Pi$.

Randomized algorithms:

- Las Vegas algorithms: always gives correct solution, example: *Quicksort*.
- Monte Carlo algorithms: it can give wrong answers. Monte Carlo algorithms subtypes:

$$- \text{type}(1): \begin{cases} \text{if } \omega \in \Pi \implies \text{algorithm returns } „\omega \in \Pi“ \text{ with probability } \geq \frac{1}{2} \\ \text{if } \omega \notin \Pi \implies \text{algorithm returns } „\omega \in \Pi“ \text{ with probability } = 0 \end{cases}$$

$$- \text{type}(2): \begin{cases} \text{if } \omega \in \Pi \implies \text{algorithm returns } „\omega \in \Pi“ \text{ with probability } = 1 \\ \text{if } \omega \notin \Pi \implies \text{algorithm returns } „\omega \in \Pi“ \text{ with probability } \leq \frac{1}{2} \end{cases}$$

$$- \text{type}(3): \begin{cases} \text{if } \omega \in \Pi \implies \text{algorithm returns } „\omega \in \Pi“ \text{ with probability } \geq \frac{3}{4} \\ \text{if } \omega \notin \Pi \implies \text{algorithm returns } „\omega \in \Pi“ \text{ with probability } \leq \frac{1}{2} \end{cases}$$

type(1) and type(2): one-sided error, type(3): 2-sided error.

$\frac{1}{2}$, $\frac{3}{4}$ and $\frac{1}{4}$ arbitrary numbers, can be something different (for type(3) better than coin flip).

Primer. Decisional problem: does a graph G have $\text{minCut} \leq k$?

Run $randMinCut(G)$ $n(n-1)$ times.

```
Algorithm randMinCut:
  if one of runs gives  $|A, B| \leq k$ :
    return true
  else:
    return false
```

Complexity classes:

- RP (randomized polynomial time): decisional problems for which there exists Monte Carlo algorithm of type(1) with polynomial time complexity (worst case).
- co-RP: decisional problems for which there exists Monte Carlo algorithm of type(2) with polynomial time complexity (worst case).
- BRP (bounded-error probabilistic polynomial time): decisional problems for which there exists Monte Carlo algorithm of type(3) with polynomial time complexity (worst case).
- ZPP (zero-error probabilistic polynomial time): decisional problems for which there exists Las Vegas algorithm with expected polynomial time complexity (worst case).

$ZPP = RP \cap co-RP$.

Poglavje 4

Chernoff bounds

Izrek 4.0.1. Let $X_1, X_2 \dots X_n$ independent random variables with image $\{0, 1\}$.

Let $p_i = P_r(X_i = x_i)$, $X = \sum_{i=1}^n X_i$ and $\mu = E(X) = p_1 + \dots + p_n$.

For every $\delta \in (0, 1)$:

$$\begin{aligned} P_r(X - \mu \geq \delta\mu) &\leq e^{-\frac{\delta^2\mu}{3}} \\ P_r(\mu - X \leq \delta\mu) &\leq e^{-\frac{\delta^2\mu}{2}} \\ \implies P_r(|X - \mu| \geq \delta\mu) &\leq e^{-\frac{\delta^2\mu}{3}}. \end{aligned}$$

Probability falls extremely quickly after $E(X)$.