## Generování dat pro experimenty s algoritmy

a) náhodným generováním (např. číselné posloupnosti, grafy)

b) použitím existujících dalších zdrojů (jmenné seznamy, texty apod.)

## Náhodné generování

Pro algoritmus přičítání jedničky k binárnímu číslu jsme potřebovali jako vstup posloupnost nul a jedniček takovou, že $P(a_i = 1) = \frac{1}{2}$, $P(a_i = 0) = \frac{1}{2}$. Jak je můžeme získat?

1) budeme házet mincí - když v $i$-tém hodu padne líc, bude $a_i = 1$, když padne rub, bude $a_i = 0$

2) použijeme počítač, na němž máme generátor náhodných čísel $x$ rovnoměrného rozdělení $R(0,1)$

když bude $i$-té vygenerované číslo menší než $1/2$, bude $a_i = 1$, jinak $a_i = 0$

dk. $P(x_i < \frac{1}{2}) = F(\frac{1}{2}) = \frac{1}{2}$

$P(x_i \geq \frac{1}{2}) = 1 - P(x_i < \frac{1}{2}) = \frac{1}{2}$

jak ale dostanu náhodné číslo $x$ z $R(0,1)$?

2.1) zvolím si přirozené číslo $M$ hodně velké

na lístečky napíšu postupně všechna čísla od 0 do $M-1$, vložím do kloboučku, zamíchám a se zavřenýma očima jeden lísteček vytáhnu - nechť je na něm číslo $X$

$P(X = k) = \frac{1}{M}$ pro $k = 0, 1, \ldots, M-1$

položím $Y = \frac{X}{M-1}$ a to je moje náhodné číslo z $R(0,1)$

2.2) použije pro výpočet $X$ nějaký rafinovaný vzorec

výsledek by měl mít dobré vlastnosti, jako kdyby se čísla tahala z klobouku

čísla pořízená tímto způsobem se nazývají pseudonáhodná

jejich výhody: dají se snadno a rychle vyrábět

dají se v případě potřeby vygenerovat znovu

nevýhody: nejsou výsledkem náhodného pokusu

nemusí být nerávislá

mohou se periodicky opakovat

jejich pravděpodobnosti nemusí přesně souhlasit s požadovaným rozdělením

nejpoužívanější typ - lineární kongruenční generátory tvaru

$$X_n = (a_0 + a_1 X_{n-1} + \dots + a_k X_{n-k}) \bmod M$$

kde konstanty $M, a_0, \dots, a_k$ je třeba vhodně volit - na nich závisí kvalita generátoru

požadavky na generátor: maximální možná perioda $M$

minimální seriální korelace

shoda s požadovaným rozdělením

kvalita se dá dobře dokázat teoreticky

ověřuje se statistickými testy

následující některé konkrétní používané generátory:

(přehled je z roku 1950)

Tabulka 1. Přehled vybraných typů generátorů z překladačů a statistických programových systémů.

| Číslo generátoru | Zdroj | Tvar | Poznámka |
|---|---|---|---|
| 1 | Překladač jazyka APL. | $V_{i+1} = (16807V_i) \bmod (2^{31}-1)$ | Tento generátor implicitně užívá i většina programových systémů napsaných v APL, např.Statgraphics. |
| 2 | Počítače typu ATARI ST (v OS ROM). | $V_{i+1} = (3141592621V_i+1) \bmod (2^{32})$ | - |
| 3 | Počítače typu IBM 360/370 , resp.operační systím VMS pro DEC VAX . | $V_{i+1} = ((2^{16}+3)V_i) \bmod (2^{31})$ | Tzv. generátor RANDU. Jedná se o jeden z nejméně kvalitních generátorů jež byl, a leckde bohužel doposud je, široce používán |
| 4 | Knihovna podprogramů IMSL, část Stat/Lib, v.1.ø . | a) $V_{i+1}=(950706376V_i) \bmod (2^{31}-1)$ | Nejlepší generátor typu $V_{i+1}=(aV_i)\bmod(2^{31}-1)$(dle hodnocení Fishmana a Moora (1982)). |
| | | b) $V_{i+1}=(397204094V_i)\bmod(2^{31}-1)$ | Viz LLRANDOMII . |
| | | c) $V_{i+1}=(16807V_i)\bmod(2^{31}-1)$ | Viz LLRANDOMI . |
| 5 | LLRANDOMI (viz Lewis a další (1988)). | $V_{i+1}=(16807V_i) \bmod (2^{31}-1)$ | Jeden z prvních úspěšných generátorů typu $V_{i+1}=(a\ V_i) \bmod (2^{31}-1)$, navržený již v roce 1969 Lewisem a kol. Postupně byl převzat řadou dalších firem. Po čase mu byly prokázány neoptimální vlastnosti ve vyšších dimenzích. |
| 6 | LLRANDOMII viz Lewis a další (1988)). | $V_{i+1}=(397204094V_i)\bmod(2^{31}-1)$ | Modifikace generátoru LLRANDOMI navržená v roce 1974 Lewisem a kol. pro odstanění "špatného" chování generátoru LLRANDOMI ve vyšších dimenzích. |

| Číslo generátoru | Zdroj | Tvar | Poznámka |
|---|---|---|---|
| 7 | Knihovna podprogramů vyvinutých v JET Propulsion Laboratories v Los Alamos. | $V_{i+1}=(5^{19}V_i) \bmod (2^{48})$ | Generátor navržený a široce používaný v laboratořích v Los Alamos na sálových počítačích. |
| 8 | Knihovna podprogramů NAG, v.11. | $V_{i+1}=(13^{13}V_i) \bmod (2^{59})$ | Značně obskurní volba konstant, jež nebyla nikdy řádně vysvětlena. NAG ani nikdy neposkytl podrobnější informaci o vlastnostech svého generátoru. |
| 9 | SASPC v.6.Ø3. | a) $V_{i+1}=(16807V_i) \bmod (2^{31}-1)$ | Viz LLRANDOMI. |
| | | b) $V_{i+1}=(397204094V_i) \bmod (2^{31}-1)$ | Viz LLRANDOMII. |
| 10 | Překladač jazyka SIMULA. | $V_{i+1}=(5^{2p+1}V_i) \bmod (2^{n})$ | Konstanty p a n jsou voleny dle typu počítače, na němž je jazyk implementován. |
| 11 | SPSS/PC v.2, resp. SPSSX v.8. | $V_{i+1}=(16807V_i) \bmod (2^{31}-1)$ | Viz LLRANDOMI. |
| 12 | Operační systém UNIX. | a) $V_{i+1}=(1103515245V_i + 12345) \bmod (2^{31})$ | Tzv. generátor "RAND". |
| | | b) $V_{i+1}=(25214903917V_i + 11) \bmod (2^{48})$ | Tzv. generátor "DRAND". |

## Metody pro generování jiných rozdělení

a) obecně platné

b) speciální

ad a) Metoda inverzní transformace

Věta: Nechť $F(x)$ je distribuční funkce, $F^{-1}(y) = \inf\{x : F(x) \geq y\}$ je odpovídající kvantilová fce a nechť $Y$ má rozdělení $R(0,1)$. Pak náhodná veličina $X = F^{-1}(Y)$ má rozdělení s distr. fcí $F(x)$.

Důkaz pro případ, kdy $F$ je spojitá a rostoucí:

$F^{-1}$ je inverzní fce k $F$

$$P(X < x) = P(F^{-1}(Y) < x) = P(Y < F(x)) = F(x)$$

o složitých případech se musí opatrněji - vychází se z toho, že $F$ je zleva spojitá a neklesající

metoda je obecně platná

pro některá rozdělení se hodí více, pro jiná méně

hodí se např. pro exponenciální rozdělení:

$$F(x) = 1 - e^{-\pi x} \text{ pro } x \geq 0$$
$$= 0 \text{ jinak}$$

inverzní fce: $F^{-1}(y) = -\frac{1}{\pi} \ln(1-y)$ , $0 < y < 1$

$$F^{-1}(F(x)) = -\frac{1}{\pi} \ln\left(1 - (1 - e^{-\pi x})\right) = -\frac{1}{\pi} \ln e^{-\pi x} = x$$

takže: vygeneruji $Y \sim R(0,1)$

položím $X = -\frac{1}{\pi} \ln(1-Y)$

stačí ale vzít $X = -\frac{1}{\pi} \ln Y$, protože $1-Y$ má také $R(0,1)$

$P(1-Y < x) = P(Y > 1-x) = 1 - P(Y < 1-x) = 1-(1-x) = x$

dále je vhodná po rozdělení, kdy $F$ je známá (analyticky) a dá se snadno invertovat

není např. vhodná pro normální rozdělení, kdy známá je hustota a distr. fce se k ní počítá jako $F(x) = \int_{-\infty}^{x} f(t)\,dt$

poličce kdy mohou být u diskrétních rozdělení

<u>Metoda zamítací</u> (von Neumann) - vhodná pro případ, kdy známe hustotu a ta je navíc ohraničená

Nechť $c = \max\{f(x),\ a \le x \le b\}$

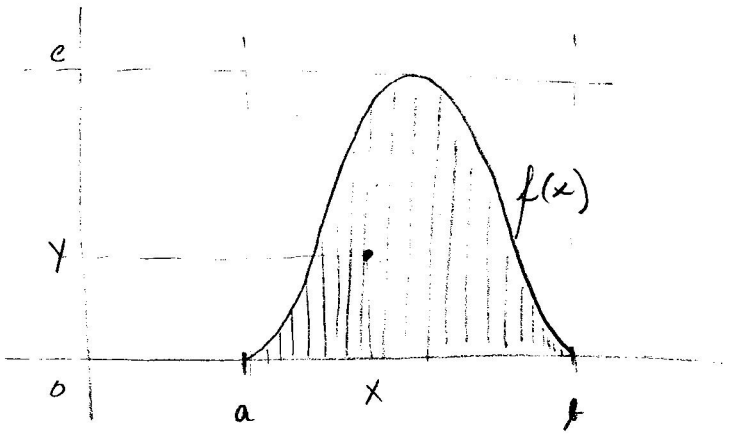
generujeme: $X$ rovnoměrně na $\langle a, b\rangle$

$Y$ rovnoměrně na $\langle 0, c\rangle$

je-li $Y \le f(X)$, vezmeme $X$ jako náhodné číslo s rozdělením $f(x)$

v opačném případě dvojici $X, Y$ zamítneme a generujeme novou

dk: $P(Y \le f(X)) = f(X)$




generujeme rovnoměrné rozd. na $\langle a, b\rangle \times \langle 0, c\rangle$

vybíráme body, které jsou pod $f(x)$

## Pro diskrétní náhodné veličiny:

Chci generovat nák. číslo $X$ rozdělení $p_0, p_1, \dots$, kde $p_i = P(X = i)$

metoda inverzní transformace:

položíme $X = 0$

$S = p_0$

vygeneruji $U \sim R(0,1)$

while $U > S$ do $X = X+1$, $S = S + p_X$ od

$X$ je nák. číslo s požadovaným rozdělením

dk: $P(X = k) = P(\text{cyklus proběhl } k\text{-krát}) =$

$= P(U > p_0 + p_1 + \dots + p_{k-1} \wedge U < p_0 + p_1 + \dots + p_k) =$

$= \sum_{i=0}^{k} p_i - \sum_{i=0}^{k-1} p_i = p_k$

nebo pomocí spojité náhodné veličiny s diskr. fcí $H$, pro

kterou $H(i+1) - H(i) = p_i$

vygenerujeme $U \sim R(0,1)$

položíme $X = [H^{-1}(U)]$ (celá část)

např. pro diskrétní rovnoměrné rozdělení na $\{1, \dots, n\}$

$H(x) = \frac{x-1}{n}$ pro $1 \le x \le n+1$

bude $H^{-1}(y) = 1 + ny$

$H^{-1}(H(x)) = 1 + n \cdot \frac{x-1}{n} = 1 + x - 1 = x$

dostaneme $X = [1 + nU]$

ad b) Normální rozdělení – metoda inverzní transformace se nepoužívá, protože výpočet kvantilové f-ce je poměrně zdlouhavý

Používá se centrální limitní věta na součty nezávislých stejně rozdělených náh. veličin s $R(0,1)$.

$X_1, \dots, X_n$ náhodná čísla z $R(0,1)$

$E \sum_{i=1}^{n} X_i = \frac{n}{2}$, $\text{var} \sum_{i=1}^{n} X_i = \frac{n}{12}$

$Y_n = \sqrt{\frac{12}{n}} \left( \sum_{i=1}^{n} X_i - \frac{n}{2} \right) \sim N(0,1)$ asymptoticky pro $n \to \infty$

v praxi se bere $n = 12$, takže $Y_{12} = \sum_{i=1}^{12} X_i - 6$.

Box – Müller: $X_1, X_2$ nezáv. náh. vel. z $R(0,1)$

$Z_1 = \sqrt{-2 \ln X_1} \sin(2\pi X_2)$

$Z_2 = \sqrt{-2 \ln X_1} \cos(2\pi X_2)$

$Z_1, Z_2$ jsou nezáv. náh. vel. z $N(0,1)$

Další důležitá spojitá rozdělení se dají generovat mezi sebou takto.

$\chi^2_n$-rozd: $X = \sum_{i=1}^{n} Z_i^2$, kde $Z_i$ jsou nezáv. s rozd. $N(0,1)$

$T_n$-rozd: $X = \frac{Y_1}{\sqrt{\frac{Y_2}{n}}}$, kde $Y_1, Y_2$ jsou nezáv.,

$Y_1 \sim N(0,1)$, $Y_2 \sim \chi^2_n$

$F_{n,m}$-rozd: $X = \frac{\frac{Y_1}{n}}{\frac{Y_2}{m}}$, kde $Y_1, Y_2$ jsou nezáv.,

$Y_1 \sim \chi^2_n$, $Y_2 \sim \chi^2_m$

## Generování diskrétních rozdělení

1) <u>binomické rozdělení</u> $P(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$

popisuje počet úspěchů v posloupnosti $n$ nezávislých náh. pokusů, v každém je psk úspěchu $p$

postup: položíme $X = 0$

opakujeme: generujeme $U$ rovnoměrně na $(0,1)$

je-li $U < p$, položíme $X \leftarrow X+1$

$X$ je náh. číslo s binom. rozd.

dle: $P(X=k) = \binom{n}{k} P^k(U<p) P^{n-k}(U>p)$

pro rovnoměrné rozd. $P(U<p) = p$

$P(U>p) = 1-p$

2) <u>geometrické rozdělení</u> $P(X=k) = (1-p)^k p$

v posloupnosti nezávislých pokusů popisuje čekání na první úspěch

$p$ ... psk úspěchu

postup: položíme $X = -1$

opakujeme, dokud $U > p$: položíme $X = X+1$

generujeme $U$ rovnoměrně na $(0,1)$

3) <u>Poissonovo rozdělení</u> – jako limitní k binomickému

Tyto metody vlastně simulují náhodný pokus.

Jsou jednoduché, ale časově náročné.

Používají se mnohé jiné a lepší, u kterých se dá ale někdy teoreticky těžko dokázat, že generují to, co mají.

Shoda vygenerovaných čísel s požadovaným rozdělením se obvykle statisticky testuje.

## $\chi^2$ - test dobré shody (shrnutí)

obor možných hodnot náhodné veličiny rozdělím na $k$ disjunktních intervalů $A_1, A_2, \dots, A_k$ - např. $(-\infty, x_1), (x_1, x_2), \dots, (x_{k-1}, \infty)$

označím $p_j = P(X \in A_j)$ - byla předpokládána nebo je umím spočítat (např. pro rovnoměrné rozdělení budou $p_j = \frac{1}{k}$ pro všech $j$, pro $N(0,1)$ je

$$P(X \in \langle x_j, x_{j+1}) ) = \Phi(x_{j+1}) - \Phi(x_j))$$

označím $n_j$ počet vygenerovaných hodnot, které padly do $A_j$ (empirické četnosti)

$\frac{n_j}{n}$, kde $n = \sum_{j=1}^{k} n_j$, je empirická relativní četnost, jejím teoretickým protějškem je $p_j$

test je založen na odchylkách $n_j - np_j$

konkrétně na statistice $\chi = \sum_{j=1}^{k} \frac{(n_j - np_j)^2}{np_j}$

platí, že $\chi$ má asymptoticky rozdělení $\chi^2_{k-1}$

přiřadí-li $\chi$ příslušnou kritickou hodnotu, hypotéza o shodě rozdělení se zamítne

Poznámka: Test se dá použít např. k ověření normality před použitím
t-testů, F-testů a podobných
pozor je, že když předem neznáme přesné parametry rozdělení,
neumím spočítat psi $p_j$, které potřebuju - metoda se pak
komplikuje tím, že se při ní zároveň musí počítat nějaké odhady

další testy:
porovnání teoretických momentů vyšších řádů příslušného rozdělení
s jejich výběrovými protějšky spočítanými z generovaných hodnot
$EX^k$ porovnám s $\frac{1}{n}\sum_{i=1}^{n} X_i^k$

Kolmogorov-Smirnovův test - je založen na rozdílu teoretické
a empirické distribuční fce

$F_n(x)$ = empirická d.f. = $\frac{\text{počet hodnot menších než } x}{\text{počet všech hodnot}} \doteq P(X < x) = F(x)$

konkrétně: spočítám $D_n = \sup_x |F_n(x) - F(x)|$
hypotézu zamítnu, když $D_n \geq D_n(\alpha)$, kde
$D_n(\alpha) \doteq \sqrt{\frac{1}{2n} \ln \frac{2}{\alpha}}$

(kritické hodnoty jsou tabelované)

## Generování náhodných posloupností

opakovaným použitím generátoru - musíme zajistit nezávislost generování

možné požadavky na posloupnost:

a) čísla se nesmí opakovat

b) posloupnost má být uspořádaná

c) má to být permutace

d) ?

ad a) generuje postupně čísla a kontroluje, jestli je už ve výběru nemám -
pomalá, lak teoreticky: náhodím celý výběr a začnu znovu
pakticky: místo čísla, které už mám, vygeneruje jiné

ad b) vygeneruje náhodný výběr, který pak setřídím
nebo použije některou metodu pro přímé pořízení uspořádaného výběru

ad c) jedna z možných metod:

vytvořím uspořádané pole $a(1)=1,\ a(2)=2,\ \ldots,\ a(n)=n$

pro $j = n, n-1, \ldots, 2$

vygeneruje číslo $i$ rovnoměrně na $\{1, \ldots, j\}$

prohodím $a(j)$ a $a(i)$

## Generování náhodných permutací – jiná metoda

Předpokládejme, že $b = (b_0, \dots, b_{N-1})$ je nějaká permutace čísel $0, \dots, N-1$.

Označme $a_k$ počet čísel $b_i$, která v permutaci následují za $k$ a jsou menší než $k$

(př.: $b = (0, 5, 7, 3, 1, 4, 2, 6)$, tj. $N = 8$

$a_1 = 0,\ a_2 = 0,\ a_3 = 2,\ a_4 = 1,\ a_5 = 4,\ a_6 = 0,\ a_7 = 5,$ ).

Nyní $a_k \le k$ pro všechna $k$.

Každé permutaci $b$ přiřadíme číslo

$$P(b) = a_1 1! + a_2 2! + \dots + a_{N-1}(N-1)!$$

Tím je definována vzájemně jednoznačná zobrazení množiny všech permutací na množinu čísel $0, 1, \dots, N!-1$.

Postup:

1) Generujeme náhodné číslo z diskrétního rovnoměrného rozdělení $R(0, 1, \dots, N!-1)$
2) určíme čísla $a_1, \dots, a_{N-1}$
3) z nich sestavíme odpovídající permutaci

Příklad: vygenerovali jsme náhodné číslo 19885 z rozdělení $R(0, 1, \dots, 8!-1)$

postupné dosazování:

$19885 = 3 \cdot 7! + 4765$ $\quad a_7 = 3$

$4765 = 6 \cdot 6! + 445$ $\quad a_6 = 6$

$445 = 3 \cdot 5! + 85$ $\quad a_5 = 3$

$85 = 3 \cdot 4! + 13$ $\quad a_4 = 3$

$13 = 2 \cdot 3! + 1$ $\quad a_3 = 2$

$1 = 0 \cdot 2! + 1$ $\quad a_2 = 0$

$1 = 1 \cdot 1! + 0$ $\quad a_1 = 1$

při konstrukci permutace začínáme od čísla 0

dostáváme: 0

10

102

1302

14302

145302

6145302

6145730 2

## Generování grafů

graf $G = (V, E)$ $V$... množina vrcholů, $|V| = n$

$E$... množina hran, tj. dvojic $(i,j)$, $i \in V$, $j \in V$

grafy s pevným počtem hran:

vygeneruji příslušný počet dvojic rovnoměrně na čtverci $V \times V$

je $(i,i)$ hrana? – pokud ne, tyto dvojice vyhážu a

vygeneruji místo nich jiné

dá se to použít pro orientované i neorientované grafy –

pro neorientované grafy vyhodím dvojici $(i,j)$, když

už mám ve výběru $(j,i)$

potíže nastanou při dalších požadavcích – jak např. vygenerovat

acyklický graf?

náhodný graf $G(n,p)$:

pro každou dvojici $(i,j)$ generuji náhodná čísla s alternativním

rozdělením s pravd. $p = P(\text{vede hrana z } i \text{ do } j)$

$1-p = P(\text{nevede} \quad \text{—}''\text{—} \quad)$

tj. generuji $U \sim R(0,1)$

když $U < p$, pak vede hrana

v opačném případě nevede

pro $p = \frac{1}{2}$ je to stejné, jako když náhodně (rovnoměrně)

vybírám z množiny všech možných grafů s daným počtem vrcholů

Náhodné grafy generované touto způsobem ale často vypadají tak, že celý graf je jedna silně souvislá komponenta, případně graf obsahuje 1 velkou silně souvislou komponentu a několik triviálních (jednoduchých) komponent.

Pokud potřebujeme graf, který by měl více netriviálních SSK, můžeme použít model $G(n,p,l)$. Parametr $l$ (tzv. lokalita) určuje okolí vrcholu, do kterého může vést hrana. Konkrétně $(i,j)$ může být hrana, když $(i-l+n) \bmod n \leq j \leq (i+l) \bmod n$.
$l$ by mělo být malé $(5,10)$, pro velká $l$ je $G(n,p,l)$ podobný grafu $G(n,p)$.

Acyklické orientované grafy můžeme generovat tomuto jako topologické uspořádání, tj. omezit se na dvojice $(i,j)$, kde $i<j$.