FJFI, ČVUT, Praha
MMC, ZS 2020/2021

Zikkurat metoda pro generování náhodných čísel

Daniel Wohlrath

1. prosince 2020

1 Úvod

Zikkurat metoda je algoritmus pro generování pseudonáhodných čísel. Náležíc do třídy zamítacích metod, závisí tento algoritmus na generátoru rovnoměrně rozdělených čísel, typicky pseudonáhodného generátoru nebo předpočítaných tabulek.

Algoritmus je primárně určen ke generování hodnot z monotónně klesající hustoty pravděpodobnosti. Může být také aplikován na unimodální symetrické hustoty, jako je například hustota normálního rozdělení, viz podkapitola 3.1. S mírnou modifikací ho lze však využít i pro komplikovanější distribuce, viz podkapitola 3.2. Metoda byla vyvyinuta v osmdesátých letech minulého století dvojicí George Marsaglia a Wai Wan Tsang. Označení *Zikkurat* je spojeno s postupem algoritmu, při kterém se příslušná hustota pravděpodobnosti postupně pokrývá obdélníky, které připomínají babylonskou stavbu - zikkurat.

2 Zikkurat metoda

2.1 Vytvoření obdélníků

Metoda je založena na pokrytí oblasti pod klesající hustotou pravděpodobnosti množinou obdélníků se stejným obsahem a "spodní lištou", skládající se z obdélníku menšího obsahu a chvostu distribuce, viz obrázek 1.

V praxi se používá 63, 127 nebo 255 obdélníků o obsahu A. Pokud tedy zvolíme například $n = 127$ a konečný bod x_{128}, pak vypočteme požadovaný obsah obdélníků A jako

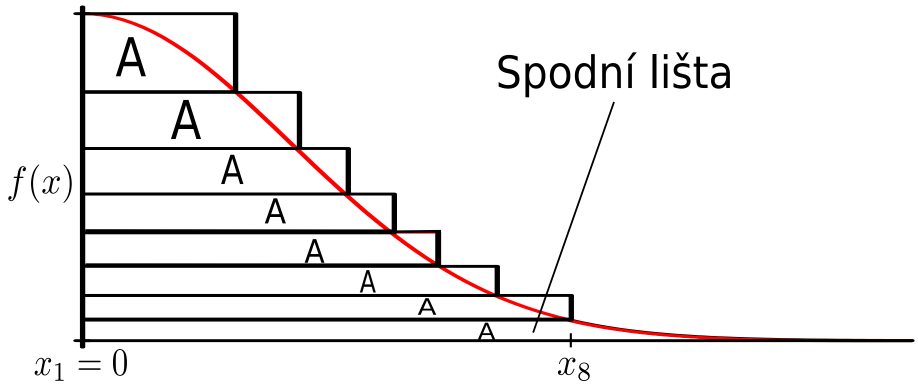
$$A = f(x_{128}) \cdot x_{128} + T\,, \tag{1}$$

kde T značí obsah pod křivkou pro $x > x_{128}$. Při značení $f(x_i) = f_i$ poté můžeme ze vztahů

$$f_i = \frac{A}{x_{i+1}} \cdot f_{i+1}\,,$$
$$x_i = f^{-1}(f_i)$$

pro $i \in \{1, \ldots, 127\}$ postupně určit příslušné hraniční body obdélníků. Úlohou je poté najít[1] takové x_{128}, aby $x_1 = 0$.

[1] např. metodou bisekce



Obrázek 1: Pokrytí klesající funkce f (červeně) 7 obdélníky stejného obsahu o velikosti A a spodní lištou s obsahem A

2.2 Generování náhodných čísel

Pokud již máme k dispozici vyhovující množinu bodů $\{x_1, \ldots, x_{128}\}$, lze přistoupit k druhé částí algoritmu, a sice ke generování náhodných čísel z této hustoty.

Nejdříve očíslujme od shora dolů jednotlivé obdélníky (vrstvy) 1 až 128, tedy první vrstva je nejuzší a zároveň nejvyšší, 128. vrstva se dotýká osy x a může být i nekonečné délky. Poté generujeme rovnoměrně rozdělená čísla i z množiny $\{1, \ldots, 128\}$ a U z intervalu $(0, 1)$. Pro $i \neq 128$ platí, že pokud je

$$U \cdot x_{i+1} < x_i \,, \tag{2}$$

pak je $U \cdot x_{i+1}$ realizací náhodné veličiny s danou hustotou. Pokud nerovnost (2) nenastává[2], generujeme z rovnoměrného rozdělení na intervalu $(0, 1)$ další číslo Z, a položíme pomocnou proměnou $\tilde{y}$ rovnu:

$$\tilde{y} = Z \cdot (f_i - f_{i+1}) \,.$$

Pokud

$$\tilde{y} < f(U \cdot x_{i+1}) \,,$$

poté $U \cdot x_{i+1}$ je žádanou realizací, jinak generujeme nové i reprezentující zkoumanou vrstvu.

2.2.1 Dolní vrstva a fall-back algoritmus

Nyní rozebereme případ, kdy $i = 128$, tedy jsme vybrali dolní vrstvu přilehlou k ose x. Dolní vrstvu můžeme rozdělit na obdélníkovou *hlavní část* a chvost, mající při zachování značení ve výrazu (1) popořadě obsahy $f_{128} \cdot x_{128}$ a T. Pokud zadefinujeme pomocnou proměnou $x_{257} = A/f_{256}$, můžeme poté částečně využít předchozího postupu:

Generujeme číslo $\hat{U} \sim R(0, 1)$[3] a pokud

$$\hat{U} \cdot x_{257} < x_{256} \,, \tag{3}$$

pak je $\hat{U} \cdot x_{257}$ hledanou realizací. Pokud nerovnost (3) nenastává, jsme nuceni generovat náhodné číslo z chvostu, pomocí tzv. *fall-back algoritmu*, který závisí na konkrétním tvaru distribuce.

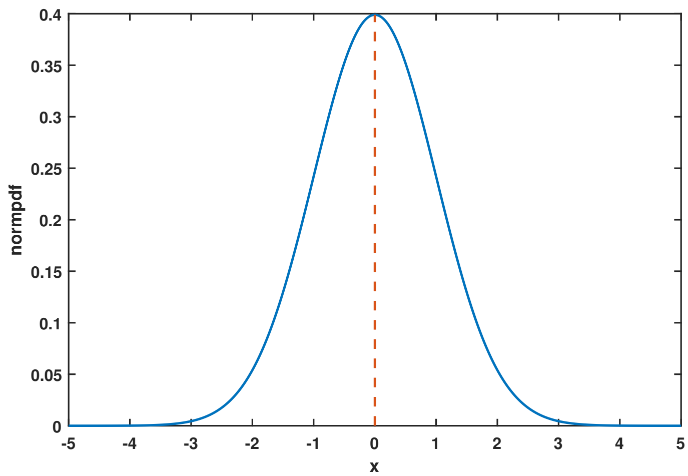
3 Realizace metody

3.1 Normální rozdělení

Při generování náhodných čísel ze standardizovaného normálního rozdělení rozdělíme hustotu na 2 části, viz obrázek 2, a aplikujeme metodu Zikkurat na její pravé křídlo. Využijeme faktu, že metoda nikde nevyžaduje

[2] Pro vrstvu č. 1 nenastane nikdy

[3] R značí rovnoměrné rozdělení



Obrázek 2: Rozdělení hustoty normálního rozdělení na 2 části

normovanost funkce[4], proto budeme pracovat pouze s funkcí $f(x) = \exp(\frac{-x^2}{2})$. Správné rozdělení 128 obdélníků získáme počáteční volbou $x_{128} = 3.442619855899$, která odpovídá obsahu obdélníku A přibližně 0.0099. Pokud generujeme ze spodní vrstvy 128, má fall-back algoritmus dle [1] tvar:

Generuj $U_1, U_2 \sim R(0,1)$, poté

1. polož $x = -\ln(U_1)/x_{128}$
2. polož $y = -\ln(U_2)$
3. pokud $2y > x$, pak $x + x_1$ je hledaná realizace
4. v opačném případě se vrať na krok 1.

Na závěr rozhodneme jednotlivě o příslušných realizacích, zda leží v levém nebo pravém křídle hustoty. Generujíc náhodné čísla z alternativního rozdělení vynásobíme námi dříve získanou realizaci faktorem -1 s pravděpodobností $1/2$ a nepřenásobíme ničím s doplňkovou pravděpodobností, tedy také $1/2$.

Při generování 10^6 náhodných čísel výše popsanou metodou byl proveden Lillieforsův test, který normalitu generovaných čísel nezamítnul s p-hodnotou menší než 0.12. Kolmogorův-Smirnovův test také hypotézu nezamítnul, a to s p-hodnotou 0.36. Na závěr pak Pearsonův χ^2 test také nulovou hypotézu nezamítnul, a sice s p-hodnotou 0.35. Na obrázku 3 je znázorněna dobrá shoda rozdělení generovaných čísel se standardizovaným normálním rozdělením.

3.2 GIG rozdělení

Zobecněné Gaussovo inverzní rozdělení, neboli GIG[5] rozdělení má hustotu pravděpodobnosti ve tvaru

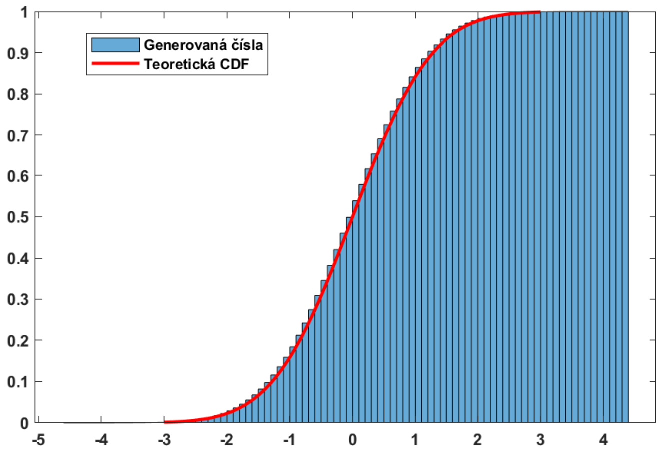
$$f(x) = \frac{(\frac{a}{b})^{p/2}}{2\mathcal{K}_p(\sqrt{ab})}\Theta(x)x^{p-1}\exp\left(-\frac{ax+\frac{b}{x}}{2}\right), \tag{4}$$

kde $\Theta(x)$ je Heavisidova skoková funkce, $\mathcal{K}_p$ je Macdonaldova funkce a pro parametry platí $a, b > 0\,, p \in \mathbb{R}$.

Pro speciální volbu parametrů $p = 6, b = 2$ a z podmínky na střední hodnotu $\int x f(x)\,\mathrm{d}x = 1$ dostáváme následně $a = 14.2655$. Pokud budeme chtít aplikovat metodu Zikkurat na tuto komplikovanější distribuci, je třeba ji rozdělit na dvě klesající části.

[4] Tedy $\int f = 1$

[5] Generalized inversed Gaussian distribution



Obrázek 3: Shoda teoretické a empirické distribuční funkce při generování náhodných čísel z normálního rozdělení

Pro náš speciální případ nabývá hustota maxima v bodě $x_{\max} = 0.8634$. Dosazením do kumulativní distribuční funkce GIGu zjistíme, že pravděpodobnost, že se realizuje veličina v intervalu $(0, x_{\max})$ je rovna $p = 0.3912$. Proto se vždy před vykonáním algoritmu rozhodneme[6], zda následující náhodné číslo budeme generovat z pravé nebo levé strany distribuce, viz obrázek 4a.

Levé křídlo hustoty má kompaktní nosič, proto se v této části nebude realizovat žádná podoba fall-back algoritmu, jako tomu bylo například při generování z normálního rozdělení. Levé křídlo jsme rozdělili na 128 obdélníků o stejném obsahu, a provedli první část Zikkurat algoritmu (bez fall-back).

Pravé křídlo hustoty již nekonečný chvost má, proto je třeba navrhnout určitou verzi fall-back algoritmu. V textu [2] je navrhnut velmi obecný postup, využívající znalost kumulativní distributivní funkce a kvantilové funkce, resp. komplementární kumulativní distribuční funkce a inverzní komplementární kumulativní distribuční funkce[7]. Autoři ve svém textu zdůvodňují, že je vhodné generovat číslo z pravého chvostu distribuce následujícím postupem:

1. generuj $U \sim R(0,1)$,
2. polož $x^* = ICCDF(U * CDF(x_n))$, kde n je počet vrstev zikkuratu, tedy v našem případě 128,
3. x^* je poté hledaná realizace.

Jelikož ICCDF k dispozici nemáme, přistoupili jsme k problému aproximativně. Hledali jsme takové x^*, že $CCDF(x^*) = U * CCDF(x_{128})$.

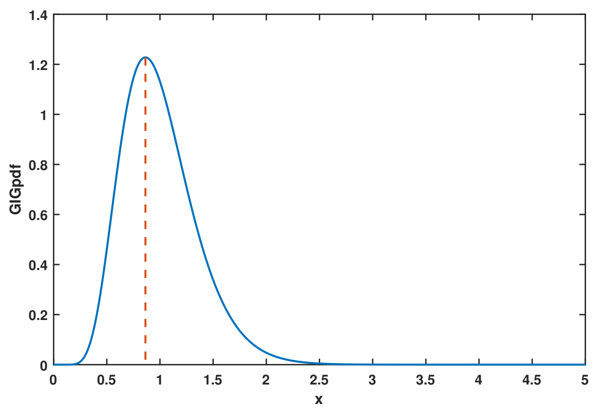
Při generování 10^6 náhodných čísel jsme pro ověření generátoru využili grafické porovnání histogramu a teoretické hustoty viz obrázek 4b a Pearsonovo χ^2 testu dobré shody. Hypotéza, že generovaná čísla pocházejí z GIG rozdělení s danými parametry nezamítnul, a to s p-hodnotou 0.27.

4 Závěr

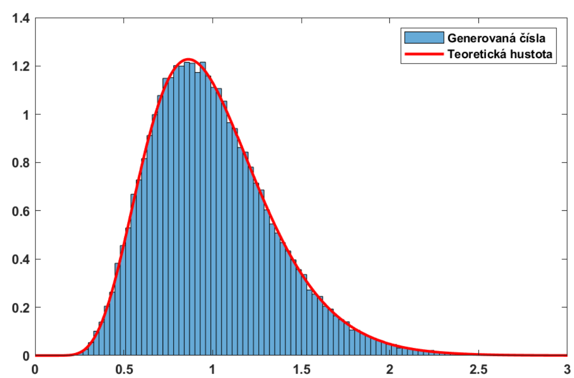
Představili jsme postup metody Zikkurat pro generování náhodných čísel. Nejdříve jsme demonstrovali její klasické použití na standardizovaném normálním rozdělení, přičemž Kolmogorovův-Smirnovovův, Lilieforsův a Pearsonův χ^2 testy dobrých shod nezamítli hypotézu normality generovaných hodnot na hladině významnosti 5%.

[6] Např. pomocí generování náhodných čísel z rovnoměrného rozdělení $(0,1)$

[7] Popořadě značené cdf a icdf resp. ccdf a iccdf



(a)



(b)

Obrázek 4: (a) Rozdělení GIG hustoty na 2 části (b) Porovnání histogramu generovaných hodnot a teoretické GIG hustoty

Metodu jsme se následně pokusili upravit, abychom mohli generovat náhodná čísla i z nemonotónních, nesymetrických hustot. Provedený způsob modifikace jsme se demonstrovali na GIG rozdělení pro speciální volbu parametrů. Úspěšnost předvedené demonstrace byla při generování 10^6 čísel znázorněna pomocí porovnání histogramu a teoretické hustoty a vyhodnocení Pearsonova chi^2 testu na hladině významnosti 5%.

Reference

[1] Wai Wan Tsang George Marsaglia. The ziggurat method for generating random variables. *Journal of Statistical Software*, 2000.

[2] Mohammad A. Charsooghi Morteza Jalalvand. Generalized ziggurat algorithm for unimodal and unbounded probability density functions with zest. 2018.