

Symulacje stochastyczne i teoria Monte Carlo

Tomasz Rolski

Instytut Matematyczny, Uniwersytet Wrocławski

Wrocław, 1 luty 2018

Prezentowany materiał jest w trakcie opracowywania. Tekst może zawierać błędy. Układ skryptu jak i oznaczenia mogą ulegać zmianom. Autor będzie wdzięczny za wszelkie uwagi oraz listy błędów, które można przesłać e-mailem na adres: rolski@math.uni.wroc.pl. Wersja v2.5.

Spis treści

I	Wstęp	3
1	O skrypcie	3
2	Symulacje prostych zadań probabilistycznych	5
2.1	Własności symetrycznego błędzenia przypadkowego	5
2.2	Zagadnienie sekretarki	8
2.3	Zagadnienie komiwojażera	9
2.4	Obliczanie całek metodą MC.	9
3	Informacja o metodach quasi Monte Carlo	10
4	Uwagi bibliograficzne	11
II	Teoria generatorów	13
1	Wstęp	13
1.1	Formalna definicja ciągu liczb losowych; liczby pseudo- losowe	15
1.2	Rozkład jednostajny na $\{0, \dots, M - 1\}$	16
2	Losowe permutacje	17
2.1	Algorytm na losową permutację	17
2.2	Tasowanie kart	18
3	Generatory liczb losowych	19
3.1	Generatory kongruencyjne	19
3.2	Generatory w MATLABie	22
3.3	Inne generatory	25
3.4	Generatory pseudolosowych ciągów bitów.	27
4	Testowanie dobrych własności generatora	29
4.1	Funkcja p -wartość	30
4.2	Testy oparte na schematach urnowych	33
4.3	Testy dla ciągów bitów	38
5	Zadania	41

III	Symulacja zmiennych losowych	49
1	Metoda ITM; rozkład wykładniczy i Pareto	49
2	Metoda ITM oraz ITR dla rozkładów kratowych; rozkład geometryczny	52
3	Metody ad hoc.	55
3.1	$B(n, p)$	55
3.2	$Erl(n, \lambda)$	56
3.3	$Poi(\lambda)$	56
3.4	Rozkład chi-kwadrat	58
4	Rozkład jednostajny	58
5	Metoda superpozycji	59
6	Metoda eliminacji	61
7	Generowanie zmiennych gaussowskich	66
7.1	Metody ad hoc dla $N(0,1)$	66
7.2	Metoda ITM dla zmiennych normalnych	70
7.3	Generowanie wektorów losowych $N(\mathbf{m}, \Sigma)$	71
8	Przykładowe symulacje	73
8.1	Uwagi bibliograficzne	74
9	Zadania	79
IV	Analiza wyników; niezależne replikacje	87
1	Przypadek estymacji nieobciążonej	87
1.1	Obliczanie całek metodą Monte Carlo	92
1.2	Symulacja funkcji od wartości oczekiwanej	94
V	Techniki redukcji wariancji	103
1	Metoda warstw	104
2	Zależności zmniejszają wariancję	112
2.1	Zmienne antytetyczne	112
2.2	Wspólne liczby losowe	115
3	Metoda zmiennych kontrolnych	117
4	Warunkowa metoda MC	118
5	Losowanie istotnościowe	121
VI	Symulacje stochastyczne w badaniach operacyjnych	133
1	Procesy zgłoszeń	134
1.1	Jednorodny proces Poissona	134
1.2	Niejednorodny proces Poissona	136

1.3	Proces odnowy	138
1.4	Co to jest proces Poissona na E i jak się go symuluje	138
2	Metoda dyskretnej symulacji po zdarzeniach	139
2.1	Modyfikacja programu na odcinek $[0, tmax]$	142
2.2	Symulacja kolejki z jednym serwerem	146
2.3	Czasy odpowiedzi, system z c serwerami	149
3	ALOHA	151
4	Prawdopodobieństwo awarii sieci	153
5	Uwagi bibliograficzne	156
VII Analiza symulacji stabilnych procesów stochastycznych.		165
1	Procesy stacjonarne i analiza ich symulacji	166
2	Przypadek symulacji obciążonej	170
3	Symulacja gdy X jest procesem regenerującym się.	175
4	Przepisy na szacowanie rozpędówki	177
4.1	Metoda niezależnych replikacji	179
4.2	Efekt stanu początkowego i załadowania systemu	179
5	Uwagi bibliograficzne	182
VIII Łańcuchy Markowa i Metody Monte Carlo		187
1	Łańcuchy Markowa	187
2	MCMC	194
2.1	Algorytmy Metropolis	195
2.2	Przykład: Algorytm Metropolis na grafie G	197
2.3	Zastosowanie algorytmu Metropolis do losowania do- zwolonej konfiguracji w modelu z twardą skorupą	199
2.4	Rozkład Boltzmana na grafie $G = (V, E)$	200
2.5	Zastosowanie algorytmu SA do problemu komiwojażera	203
2.6	Trochę teorii niejednorodnych łańcuchów Markowa	204
2.7	Rozkłady Gibbsa	205
3	Uwagi bibliograficzne	206
4	Zadania	207
IX Symulacja procesów matematyki finansowej		209
1	Ruch Browna	211
1.1	Konstrukcja ruchu Browna via most Browna	212
2	Geometryczny ruch Browna	213
2.1	Model Vasicka	214

2.2	Schemat Eulera	216
3	Uwagi bibliograficzne	217
X	Sumulacja zdarzeń rzadkich	221
1	Efektywne metody symulacji rzadkich zdarzeń	221
2	Dwa przykłady efektywnych metod symulacji rzadkich zdarzeń	223
2.1	Technika wykładniczej zamiany miary	224
2.2	$\mathbb{P}(S_n > (\mu + \epsilon)n)$ - przypadek lekko-ogonowy	225
2.3	$\mathbb{P}(S_n > x)$ - przypadek ciężko-ogonowy	226
2.4	Uwagi bibliograficzne	226
XI	Dodatek	229
1	Elementy rachunku prawdopodobieństwa	229
1.1	Rozkłady zmiennych losowych	229
1.2	Podstawowe charakterystyki	231
1.3	Niezależność i warunkowanie	232
1.4	Transformaty	235
1.5	Zbieżności	236
2	Elementy procesów stochastycznych	237
2.1	Ruch Browna	237
3	Rodziny rozkładów	239
3.1	Rozkłady dyskretne	239
3.2	Rozkłady dyskretne wielowymiarowe	240
3.3	Rozkłady absolutnie ciągłe	241
3.4	Rozkłady z ciężkimi ogonami	242
3.5	Rozkłady wielowymiarowe	242
4	Tablice rozkładów	244
5	Elementy wnioskowania statystycznego	251
5.1	Teoria estymacji	251
5.2	Przedziały ufności	252
5.3	Statystyka Pearsona	253
5.4	Dystrybuanty empiryczne	253
5.5	Wykresy kwantylowe	254
5.6	Własność braku pamięci rozkładu wykładniczego	259
6	Procesy kolejkowe	260
6.1	Standardowe pojęcia i systemy	260
7	Przegląd systemów kolejkowych	261
7.1	Systemy z nieskończoną liczbą serwerów	261

<i>SPIS TREŚCI</i>	1
7.2 Systemy ze skończoną liczbą serwerów	263
7.3 Systemy ze stratami	266
8 Błądzenie przypadkowe i technika zamiany miary	267
8.1 Błądzenie przypadkowe	267
8.2 Technika wykładniczej zamiany miary	268
XI Projekty studenckie	271

Rozdział I

Wstęp

1

1 O skrypcie

Stochastyczne symulacje i teoria Monte Carlo są bardzo często są nazwami tej samej teorii. Jednakże w opinii autora tego skryptu warto je rozróżnić. Pierwszy termin będzie więc dotyczył teorii generatorów oraz metod generowania liczb losowych o zadanych rozkładach. Właściwie powinno się mówić o liczbach pseudo-losowych, bo tylko takie mogą być generowane na komputerze, ale w tekście często będziemy mówili o zmiennych lub liczbach losowych. Natomiast przez teorię Monte Carlo będziemy tutaj rozumieć teoretycznymi podstawami opracowania wyników, planowania symulacji, konstruowaniu metod pozwalających na rozwiązywanie konkretnych zadań, itp.

W przeciwieństwie do metod numerycznych, gdzie wypracowane algorytmy pozwalają kontrolować deterministycznie błędy, w przypadku obliczeń za pomocą metod stochastycznych dostajemy wynik losowy, i jest ważne aby zrozumieć jak taki wynik należy interpretować, co rozumiemy pod pojęciem błędu, itd. Do tego przydają się pojęcia i metody statystyki matematycznej.

Skrypt jest pisany z myślą o studentach matematyki Uniwersytetu Wrocławskiego, w szczególności studentach interesujących się informatyką oraz zastosowaniami rachunku prawdopodobieństwa. Dlatego, ideą przewodnią tego skryptu, jest przeprowadzanie symulacji rozmaitych zadań rachunku prawdopodobieństwa, przez co czytelnik poznaje przy okazji obszary, które

¹Poprawiane 17.1.2018

trudno poznać bez specjalistycznej wiedzy z rachunku prawdopodobieństwa i procesów stochastycznych. Dlatego też nie zakłada się od czytelnika szczegółowej wiedzy z tych przedmiotów, a tylko zaliczenie podstawowego kursu uniwersyteckiego z rachunku prawdopodobieństwa i statystyki matematycznej. Natomiast po ukończeniu kursu, oprócz znajomości symulacji stochastycznej i teorii Monte Carlo, autor skryptu ma nadzieję, że czytelnik będzie zaznajomiony z wieloma klasycznymi zadaniami teorii prawdopodobieństwa, modelami stochastycznymi w badaniach operacyjnych i telekomunikacji, itd.

W dzisiejszej literaturze rozróżnia się generatory liczb losowych od generatorów liczb pseudolosowych. W tym wykładzie zajmujemy się liczbami pseudolosowymi, to jest otrzymanymi za pomocą pewnej rekurencji matematycznej. Natomiast ciągi liczb losowych możemy otrzymać na przykład przez rzuty monetą, rzuty kostką, obserwacji pewnych zjawisk atmosferycznych, zapisu rejestrów itp. Rozróżnia się hardwerowe generatory liczb losowych od softwerowych generatorów liczb losowych. Jednakże tutaj nie będziemy się tym zajmowali. Oprócz liczb pseudolosowych mamy też liczby quasi-losowe. Są to ciągi liczbowe które mają które spełniają własność 'low discrepancy'.

Skrypt składa się z rozdziałów dotyczących:

- zarysu teorii generatorów liczb pseudolosowych,
- sposobów generowania zmiennych losowych o zadanych rozkładach,
- podstawowych pojęć dotyczących błędów, poziomu istotności, liczby replikacji i ich związków,
- przykładowych zadań rozwiązywanych metodami symulacji stochastycznej,
- metodami zmniejszenia liczby replikacji przy zadanym poziomie błędów i poziomie istotności.

Istotnym składnikiem tego skryptu jest dodatek, w którym znajdują się pojęcia z rachunku prawdopodobieństwa, przegląd rozkładów i ich własności, potrzebne wiadomości ze statystyki matematycznej, zarys teorii kolejek, itd. Niektóre fragmenty znajdujące się w dodatku muszą zostać przerobione na wykładzie, bowiem trudno zakładać ich znajomość, a bez których wykład byłby niepełny.

2 Symulacje prostych zadań probabilistycznych

Liczby losowe generujemy na komputerze. Na przykład w MATLAB'ie przez komendę

```
rand
```

Na podstawie liczb losowych możemy generować liczby losowe o zadanym rozkładzie. W MATLABIE jest również komenda

```
randi(M)
```

2.1 Własności symetrycznego błędzenia przypadkowego

W rozdziale 3-cim książki Fellera [13] rozważa się zadania związane z ciągiem rzutów symetryczną monetą. Jest dwóch graczy A i B. Jeden z nich obstawia orła a drugi reszkę. Jeśli gracz A odgadnie wynik, to wygrywa +1 od gracza B, w przeciwnym razie gracz B wygrywa 1 od gracza A. Zakładamy, że wyniki kolejnych rzutów są niezależne. Przypuśćmy, że gracze zaczynają grać z pustym kontem; tj $S_0 = 0$. Niech S_n będzie wygraną gracza A po n rzutach (tj. przegrana gracza B po n rzutach). Zauważmy, że wygodnie będzie rozpatrywanie $2N$ rzutów, i remisy mogą być jedynie w momentach $2, 4, \dots, 2N$. Możemy postawić różne pytania dotyczące przebiegu gry, tj. przebiegu $(S_n)_{n=1, \dots, 2N}$. Możemy też przedsatwić błędzenie w postaci łamanej, łącząc prostymi punkty $(i-1, S_{i-1}), (i, S_i)$. Wtedy będziemy mieli segment powyżej osi x -w, tj. A prowadzi na segmencie $(i-1, S_{i-1}) \rightarrow (i, S_i)$ jeśli $S_{i-1} \geq 0$ lub $S_i \geq 0$.²

1. Niech $R_{2N} = \max\{i : S_i = 0\}$ będzie ostatnim momentem remisu. Co można powiedzieć o rozkładzie R_{2N} ?
2. Jakie jest prawdopodobieństwo $P(\alpha, \beta)$, że przy $2N$ rzutach gracz A będzie prowadził pomiędzy 100α a 100β procent czasu?
3. Jak wyglądają oscylacje $(S_n)_{0 \leq n \leq 2N}$.
4. Jakie jest prawdopodobieństwo, że jeden z graczy będzie cały czas prowadził.

²Feller t.1 str 79

Tego typu zadania formalizujemy następująco, co następnie pozwoli na symulacje. Niech $\xi_1, \xi_2, \dots$, będzie ciągiem niezależnych zmiennych losowych o jednakowym rozkładzie $\mathbb{P}(\xi_j = 1) = \mathbb{P}(\xi_j = -1) = 1/2$ i definiujemy

$$S_0 = 0, \quad S_n = \sum_{j=1}^n \xi_j, \quad n = 1, \dots$$

Ciąg $S_0 = 0, S_1, \dots$ nazywamy *prostym symetrycznym błędzeniem przypadkowym*.

Zauważmy, że formalnie definiujemy

$$L_{2n} = \sup\{m \leq 2n : S_m = 0\}.$$

Przez

$$L_{2n}^+ = \#\{j = 1, \dots, 2n : S_{j-1} \geq 0 \quad \text{lub} \quad S_j \geq 0\}.$$

Inna charakterystyką może być, że gracz A ostro prowadzi w momencie m jeśli $S_m > 0$. Łączny czas ostrego prowadzenia A jest

$$\#\{1 \leq k \leq 2N : S_k > 0\}.$$

Przykładową symulację prostego symetrycznego błędzenia przypadkowego $S_0 = 0, S_1, \dots, S_N$ można otrzymać za pomocą następującego programu. ³

Proste symetryczne błędzenia przypadkowego

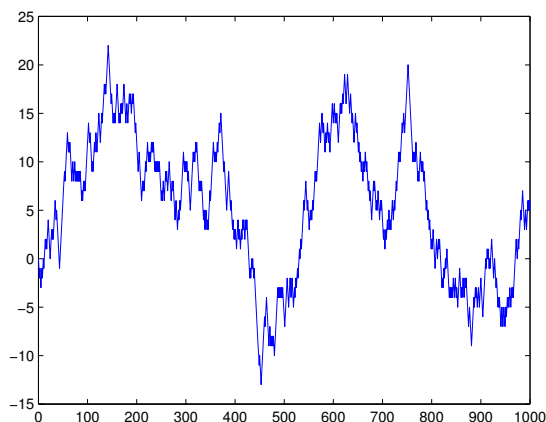
```
% N number of steps;
xi=2*floor(2*rand(1,N))-1;
A=triu(ones(N));
y=xi*A;
s=[0,y];
x=0:N;
plot(x,s)
```

Przykładowy wykres błędzenia przypadkowego podany jest na rys. 2.1

Na wszystkie pytania znane są odpowiedzi analityczne w postaci twierdzeń granicznych gdy $N \rightarrow \infty$; patrz Feller [13]. Na przykład wiadomo, że (patrz książka Fellera [13] lub Durreta [12])

$$\mathbb{P}(\alpha \leq L_{2N}^+/2N \leq \beta) \rightarrow \int_{\alpha}^{\beta} \pi^{-1}(x(1-x))^{-1/2} dx,$$

³dem2-1.m



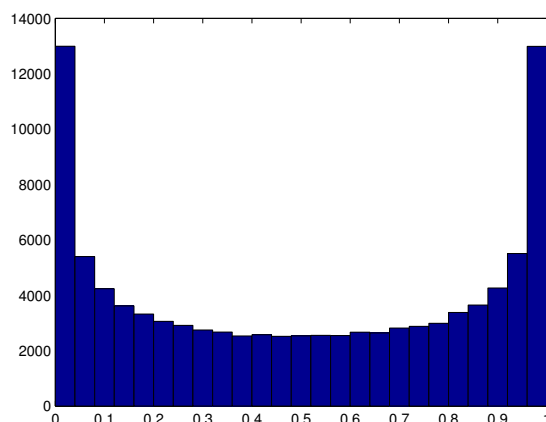
Rysunek 2.1: Przykładowa realizacja prostego błędzenia przypadkowego; $N = 1000$.

dla $0 \leq \alpha < \beta \leq 1$. Jest to tak zwane *prawo arcusa sinusa*. Nazwa bierze się stąd, że

$$\int_0^t \pi^{-1}(x(1-x))^{-1/2} dx = \frac{2}{\pi}(\arcsin(\sqrt{t})) .$$

Podobny rezultat jest dla L_{2N} ; patrz Durrett str. 197. My spróbujemy znaleźć odpowiedzi przez symulacje. Mając algorytm na symulacje błędzenia przypadkowego proponujemy zrobić histogramy dla powtórzeń L_{2N}^+ – łącznego czasu prowadzenia przez A, L_{2N} . Jest jasne też, że gdy $N \rightarrow \infty$, to wielkości L_{2N}^+ , L_{2N} rosną do nieskończoności. Ale jak się okazuje należy rozpatrywać frakcje czasu, czyli te wielkości podzielić przez $2N$. Metoda Monte Carlo polega na wielokrotnym powtórzeniu eksperymentu, tj. w naszym wypadku obliczenia powyższych wielkości. Niech R będzie liczbą przeprowadzonych eksperymentów, zwanych w teorii Monte Carlo *replikacjami*. Na rys 2.2 mamy histogram dla L_{1000}^+ przy $R = 1000000$ replikacjach.

⁴Plik arcsinNew3.m Oblicza prawdopodobieństwo, że prowadzenie będzie krótsze niż 100*x % czasu, plik arcsinNew.m podaje histogram dla Lplus/2N z $R = 100000$, $D = 2N = 1000$, i $n = 25$ jednakowej długości klas; obrazek arcsin.eps



Rysunek 2.2: Histogram dla L_{1000}^+ ; $R=100000$ replikacji, 25 klas

2.2 Zagadnienie sekretarki

Jest to problem z dziedziny optymalnego zatrzymania. Rozwiązanie jego jest czasami zwane regułą 37%. Można go wysłowić następująco:

- Jest jeden etat do wypełnienia.
- Na ten 1 etat jest n kandydatów, gdzie n jest znane.
- Komisja oceniająca jest w stanie zrobić ranking kandydatów w ścisłym sensie (bez remisów).
- Kandydaci zgłaszają się w losowym porządku. Zakłada się, że każde uporządkowanie jest jednakowo prawdopodobne.
- Po rozmowie kwalifikacyjnej kandydat jest albo zatrudniony albo odrzucony. Nie można potem tego cofnąć.
- Decyzja jest podjęta na podstawie względnego rankingu kandydatów spotkanych do tej pory.
- Celem jest wybranie najlepszego zgłoszenia.

Okazuje się, że optymalna strategia, to jest maksymizująca prawdopodobieństwo wybrania najlepszego kandydata jest, należy szukać wśród następujących strategii: opuścić k kandydatów i następnie przyjąć pierwszego lepszego

od dotychczas przeglądanych kandydatów. Jeśli takiego nie ma to bierzemy ostatniego. Dla dużych n mamy $k \sim n/e$ kandydatów. Ponieważ $1/e \approx 0.37$ więc stąd jest nazwa 37 %. Dla małych n mamy

n	1	2	3	4	5	6	7	8	9
k	0	0	1	1	2	2	2	3	3
P	1.000	0.500	0.500	0.458	0.433	0.428	0.414	0.410	0.406

To zadanie można prosto symulować jeśli tylko umiemy generować losową permutację. Możemy też zadać inne pytanie. Przypuśćmy, że kandydaci mają rangi od 1 do n , jednakże nie są one znane komisji zatrudniającej. Tak jak poprzednio komisja potrafi jedynie uporządkować kandydatów do tej pory starających się o pracę. Możemy zadać pytanie czy optymalny wybór maksymizujący prawdopodobieństwo wybrania najlepszego kandydata jest również maksymizujący wartość oczekiwaną rangi zatrudnianej osoby. Na to pytanie można sobie odpowiedzieć przez symulację.

2.3 Zagadnienie komiwojażera

Komiwojazer opuszcza miasto 0 i musi odwiedzić wszystkie miasta $1, \dots, n$ zanim powróci do miasta 0. Odległości pomiędzy miastem i a miastem j są zadane macierzą $(c_{ij})_{i,j=0,\dots,n}$. Jeśli nie ma drogi z i do j to $c_{ij} = \infty$. Należy znaleźć drogę, która minimizuje odległość $H(\pi)$, gdzie dla zadanej permutacji π liczb $1, \dots, n$

$$H(\pi) = c_{0\pi_1} + c_{\pi_1\pi_2} + \dots + c_{\pi_{n-1}\pi_n} + c_{\pi_n,0}$$

Zadanie to dla dużych n jest trudne do wykonania metodą deterministyczną ze względu na dużą liczbę obliczeń. Natomiast okazuje się, że zrandomizowane algorytmy dają niespodziewanie dobre rezultaty. Oczywiście na początku można poszukać 'dobrej' drogi przez symulację.

2.4 Obliczanie całek metodą MC.

Metoda MC jest często wykorzystana do liczenia całek. Oczywiście w nietrywialnych zadaniach są to całki wielowymiarowe, często po obszarze nie zadanym prostym wzorem, które wykluczają użycie standardowych metod numerycznych. Używając metody MC do obliczania tej całki, korzystamy

z tego, że jeśli $\mathbf{U}$ jest liczbą losową z $[0, 1)^d$ (tzn. składowe wektora $\mathbf{U}$ są niezależne o jednakowym rozkładzie jednostajnym na $[0, 1)$), to z mocnego prawa wielkich liczb, z prawdopodobieństwem 1

$$\frac{1}{n} \sum_{i=1}^n f(\mathbf{U}_i) \rightarrow \mathbb{E} f(\mathbf{U}) = \int_{[0,1]^d} f(\mathbf{u}) \, d\mathbf{u}.$$

Szybkość tej metody jest rzędu $O(\frac{1}{\sqrt{n}})$ niezależnie od wymiaru d . Ten fakt pokażemy później.

3 Informacja o metodach quasi Monte Carlo

Alternatywną metodą do Monte Carlo (MC) używającą liczb pseudo-losowych są metody quasi-MC używającą liczb quasi-losowych. Jest to metoda szczególnie użyteczna przy obliczeniu całek $\int_{[0,1]^d} f(\mathbf{u}) \, d\mathbf{u}$. Zamiast generowania ciągu liczb losowych, lub pseudolosowych teraz przy użyciu metody quasi-MC dany jest ciąg deterministyczny $\mathbf{x}_1, \mathbf{x}_2, \dots$, który posiada pewne własności o których za chwilę, i obliczymy szukaną całkę korzystając z tego, że

$$\frac{1}{n} \sum_{i=1}^n f(\mathbf{x}_i) \rightarrow \mathbb{E} f(\mathbf{U}) = \int_{[0,1]^d} f(\mathbf{u}) \, d\mathbf{u}.$$

Jeśli ciąg $(\mathbf{x}_i)$ jest ciągiem quasi-losowym, to wtedy szybkość zbieżności jest rzędu $O(n^{-(1-\epsilon)})$, gdzie $0 < \epsilon < 1$ jest dowolne. Dokładniejsza postać rzędu zbieżności będzie podana za chwilę. Oczywiście potrzebne są też pewne założenia z funkcji f , które w praktyce są spełnione. Metody quasi MC są szeroko stosowane do wycen produktów finansowych.

Teoria liczb quasi-losowych opiera się na pojęciu *dyskrepacji* ciągu $\mathbf{x}_1, \mathbf{x}_2, \dots \in [0, 1)^d$. Zaczniemy od dowolnego ciągu $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n \in [0, 1)^d$ i niech $\mathcal{A}$ będzie pewną rodziną podzbiorów z $[0, 1)^d$. Wtedy przez dyskrepację rozumiemy wielkość

$$D(\mathbf{x}; \mathcal{A}) = \sup_{A \in \mathcal{A}} \left| \frac{\#\{i : \mathbf{x}_i \in A\}}{n} - \text{vol}(A) \right|$$

gdzie $\text{vol}(A)$ jest objętością zbioru A . W tym szkicu będziemy rozpatrywać rodzinę $\mathcal{A}^*$ podzbiorów postaci $\prod_{j=1}^d [0, u_j)$, gdzie $0 < u_j < 1$. Wtedy mówimy o gwiazdkowej dyskrepacji i oznaczamy $D^*(\mathbf{x}) = D(\mathbf{x}; \mathcal{A}^*)$. Dla każdego ustalonej długości ciągu n , możemy to zadanie wariacyjne rozwiązać i mamy $D^*(\mathbf{x}) \geq \frac{1}{2n}$, i równość jest osiągnięta dla $x_i = (2i - 1)/(2n)$,

$i8 = 1, \dots, n$. Rozważmy teraz dowolny ciąg $\mathbf{x}_1, \mathbf{x}_2, \dots$. Okazuje się, że teraz problem jest dużo trudniejszy niż w przypadku skończonego ciągu.

Główny konceptem tej teorii jest następujące pojęcie. Mówimy, że ciąg $\mathbf{x}_1, \mathbf{x}_2, \dots \in [0, 1]^d$ jest *niskiej dyskrepancji* jeśli

$$D^*(\mathbf{x}_1, \dots, \mathbf{x}_n) \sim O\left(\frac{(\log n)^d}{n}\right).$$

Okazuje się, że istnieją takie ciągi. Na przykład ciąg Haltona, ciąg Sobola, czy ciąg Faure. Dla tych ciągów można pokazać, że

$$D^*(\mathbf{x}_1, \dots, \mathbf{x}_n) \leq C_d \frac{(\log n)^d}{n} + O\left(\frac{(\log n)^{d-1}}{n}\right).$$

W MATLABie w toolboxie *Statistics and machine learning* są zaimplementowane procedury generujące ciągi haltona i sobola (haltonset i sobolset).

4 Uwagi bibliograficzne

Na rynku jest sporo dobrych podręczników o metodach stochastycznej symulacji i teorii Monte Carlo. Zaczniemy od najbardziej elementarnych, jak na przykład książka Rossa [40]. Z pośród bardzo dobrych książek wykorzystujących w pełni nowoczesny aparat probabilistyczny i teorii procesów stochastycznych można wymienić książkę Asmussena i Glynn [4], Madrasa [33], Fishmana, Ripleya [38]. Wśród książek o symulacjach w szczególnych dziedzinach można wymienić książkę Glassermana [16] (z inżynierii finansowej), Fishmana [15] (symulację typu *discrete event simulation*). W języku polskim można znaleźć książkę Zielińskiego [50] oraz bardziej współczesną Wieczorkowskiego i Zielińskiego [48]. Wykłady prezentowane w obecnym skrypcie były stymulowane przez notatki do wykładu prowadzonego przez J. Resinga [37] na Technische Universiteit Eindhoven w Holandii. W szczególności autor skryptu dziękuje Johanowi van Leeuwenowi za udostępnienie materiałów. W szczególności w niektórych miejscach autor tego skryptu wzoruje się na wykładach Johana van Leeuwena i Marko Bono z Uniwersytetu Technicznego w Eindhoven.

Rozdział II

Teoria generatorów

1 Wstęp

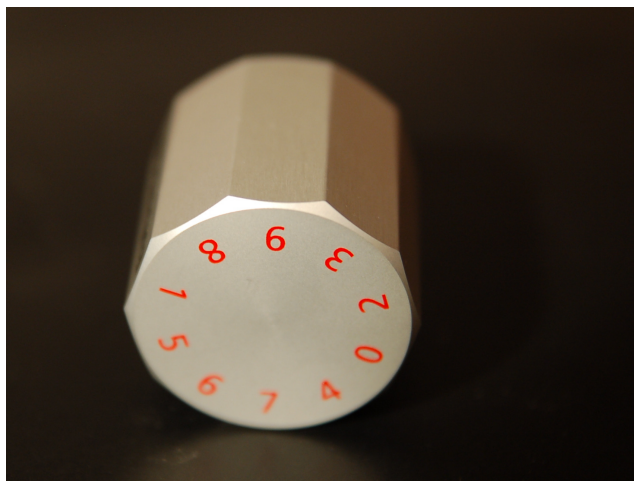
Podstawą symulacji stochastycznej jest pojęcie *liczby losowej*. Nie będziemy tutaj wdawać się szczegółowo w filozoficzne dysputy znaczenia słowa “losowy”. Zacytujmy tylko co na ten temat pisze Knuth w swoim fundamentalnym dziele [23], str 2: “W pewnym sensie nie ma czegoś takiego jak liczba losowa; czy na przykład 2 jest liczbą losową. Można natomiast mówić o *ciągu niezależnych zmiennych losowych* o określonym *rozkładzie*, a to oznacza, z grubsza rzecz biorąc, że każda z liczb została wybrana zupełnie przypadkowo, bez żadnego związku z wyborem pozostałych liczb w ciągu, i że każda liczba mieści się w dowolnym ustalonym zakresie wartości z określonym prawdopodobieństwem.” W skrypcie terminy liczba losowa i zmienna losowa są używane wymiennie.

Podstawowym rozkładem w tej teorii jest rozkład jednostajny $U(0,1)$. Jednakże na komputerze nie możemy otrzymać wszystkich liczb z odcinka $[0, 1]$. Dlatego tak naprawdę generujemy liczby przyjmujące wartości

$$0, 1/M, 2/M, \dots, (M-1)/M$$

dla pewnego $M = 2^k$ i będziemy oczekiwać, że ona ma rozkład jednostajny na tym zbiorze. W tym wypadku będziemy mówili często o liczbie losowej bez wymieniania nazwy rozkładu, tj. zmiennej losowej U o rozkładzie jednostajnym $U(0,1)$ na $(0,1)$. Oznaczenia rozkładów, wraz definicjami, ich podstawowymi charakterystykami są podane w dodatku XI.1.1 oraz tablicy XI.4.

Przed nastaniem ery komputerowej, tworzone tablice liczb losowych. Najprostsza tablicę można zrobić samemu, wyciągając na przykład kule z urny, rzucając kostką, itd, następnie te wyniki zapisując. Można też używać “kostki” dającej wyniki od 0,1 do 9; patrz Rys. 1.1 i przez kolejne rzuty produkować



Rysunek 1.1: Kostka z Eurandom

wać ciąg *cyfr losowych*. Następnie grupując względem ustalonej liczby cyfr, poprzedzając je zerem i kropką¹ otrzymać żądane tablice liczb losowych. Takie tablice mają często “dobre własności” (co pod tym rozumiemy opowiemy później) ale gdy potrzebujemy bardzo dużo liczb losowych (na przykład setki tysięcy) opisana metoda jest bezużyteczna.

Dzisiaj do tworzenia ciągu liczb losowych (lub jak się mówi do ich generowania) stosuje się komputery. Chyba pierwszym, który to zrobił był John von Neumann około 1946 r. Zaproponował on aby tworzyć następną liczbę podnosząc do kwadratu poprzednią i wycięciu środkowych. “Metoda środka kwadratu” okazała się niezbyt dobra; bowiem ma tendencję do wpadania w koleinę – krótki cykl powtarzających się elementów. Na przykład ciąg dwucyfrowych liczb zaczynających się od 43 według tego algorytmu jest 84, 05, 00, 00, Mianowicie mamy $43^2 = 1849$, $84^2 = 7056$ i $05^2 = 25$. Powstała cała obszerna teoria dotycząca generatorów, tj. algorytmów produkujących ciągi “liczb losowych”, analizy dobroci generatorów. Teoria ta wykorzystuje osiągnięcia współczesnej algebry i statystyki.

¹W skrypcie używamy konwencję 0.1234 na zapis liczby z przedziału (0,1) a nie 0,1234.

Niestety komputer nie jest w stanie wygenerować idealnego ciągu niezależnych zmiennych losowych o rozkładzie jednostajnym. Dlatego czasem mówi się o liczbie pseudo-losowej generowanej przez komputer. Rozumiejąc te ograniczenia będziemy przeważnie dalej pisać o ciągu zmiennych losowych o rozkładzie $U(0,1)$.

Ogólny schemat generowania liczb losowych na komputerze można scharakteryzować następująco przez piątkę: (S, s_0, f, U, g) , gdzie

- S jest skończoną przestrzenią stanów,
- $s_0 \in S$ jest wartością początkową rekurencji (ziarno),
- $s_{i+1} = f(s_i)$, gdzie $f : S \rightarrow S$,
- natomiast U skończoną przestrzenią wartości oraz
- $g : S \rightarrow U$, i $u_i = g(s_i)$ ($i = 0, 1, \dots$ jest wygenerowanym ciągiem liczb pseudolosowych).

Wtedy mówimy, że (S, s_0, f, U, g) jest generatorem liczb losowych (GLL).

Łatwo zauważyć, że ciąg generowany przez $X_{n+1} = f(X_n)$ zawsze wpada w pętlę: w końcu pojawia się cykl, który nieustannie się powtarza. Długość takiego cyklu nazywa się *okresem*. Dobroć generatora zależy od okresu. Zależy nam na bardzo długich okresach. Zauważmy, że teoretycznie cykl nie może być dłuższy niż moc S .

1.1 Formalna definicja ciągu liczb losowych; liczby pseudo-losowe

Przez ciąg liczb losowych rozumiemy ciąg U_j ($j = 1, \dots$) zmiennych losowych ($U_i : (\Omega, \mathcal{F}, \mathbb{P}) \rightarrow [0, 1]$) o jednakowym rozkładzie jednostajnym $\mathcal{U}[0, 1]$, tzn.

(i) U_j ma rozkład jednostajny, tzn.

$$\mathbb{P}(U_j \leq t) = \begin{cases} 0 & t < 0, \\ t & 0 \leq t < 1 \\ 1 & t \geq 1, \end{cases}$$

(ii) $U_1, U_2, \dots$ są niezależne o jednakowym rozkładzie, tj. dla $n = 1, 2, \dots$

$$\mathbb{P}(U_1 \leq t_1, \dots, U_n \leq t_n) = t_1 t_2 \cdots t_n,$$

dla $0 \leq t_1, \dots, t_n \leq 1$.

Przez *losowy ciąg bitów* rozumiemy ciąg zmiennych losowych $\xi_1, \xi_2, \dots$ ($\xi_j : (\Omega, \mathcal{F}, \mathbb{P}) \rightarrow \{0, 1\}$), taki, że

$$(i) \quad \mathbb{P}(\xi_j = 0) = \mathbb{P}(\xi_j = 1) = 0.5,$$

(ii) zmienne losowe $\xi_1, \xi_2, \dots$ są niezależne o jednakowym rozkładzie.

Zauważmy ważną i nietrywialną własność takich ciągów liczb (bitów) losowych. Jeśli $n_1 < n_2 < \dots$ jest podciągiem liczb naturalnych, to ciąg $U_{n_1}, U_{n_2}, \dots$ jest ciągiem liczb losowych. Podobnie jest dla losowego ciągu bitów.

Przedmiotem dalszych rozważań w tym wykładzie będą ciągi liczb (bitów) pseudo-losowych, tj. takich które imitują ciągi liczb (bitów) losowych. Takie ciągi można otrzymywać na różne sposoby, ale my zajmiemy się generowanymi przez odpowiednie algorytmy na komputerach. Mówi się o generatorach liczb pseudolosowych (w języku angielskim *pseudo-random number generator* (PRNG)). Ponieważ te dwa pojęcia się zlewają, w dalszej części wykładu będziemy opuszczali słowo *pseudo*. W literaturze często pisze się po prostu *random number generator* (RNG), który oczywiście musi być generatorem liczb pseudolosowych. Człowiek nie jest bowiem w stanie generować idealny ciąg liczb które są realizacjami ciągu niezależnych zmiennych losowych o jednakowym rozkładzie. Kontynuując rozważania terminologiczne, zauważmy, że jeśli jest mowa o *R replikacjach* liczby losowej U , to jeśli nie będzie powiedziane co innego, mamy ciąg $U_1, \dots, U_R$ zmiennych losowych o jednakowym rozkładzie jednostajnym $U(0, 1)$.

1.2 Rozkład jednostajny na $\{0, \dots, M - 1\}$

W niektórych zadaniach potrzebny jest ciąg liczb losowych $Y_1, Y_2, \dots$ przyjmujących wartości od $0, \dots, M - 1$. Ciąg ten nazywamy losowy, jeśli jest realizacją z próby niezależnych zmiennych losowych o jednakowym rozkładzie jednostajnym na $M = \{0, \dots, M - 1\}$, tj.

$$\mathbb{P}(Y_j = i) = \frac{1}{M} \quad i = 0, \dots, M - 1.$$

Mając do dyspozycji generator, na przykład MATLABowski

`rand`

możemy takie liczby w prosty sposób otrzymać następująco: $Y_i = \lfloor MU_i \rfloor$. Jeśli więc chcemy otrzymać ciąg 100 elementowy z $M = 10$ to piszemy skrypt

```
%Generate uniform integers on the set 0:M-1:
n = 100; M=10;
Y = floor(M.*rand(n,1));
```

Polecamy zastanowić się, jak zmienić powyższy skrypt aby otrzymać cyfry losowe równomiernie rozłożone na $1, \dots, M$.

2 Losowe permutacje

Niech $\mathcal{S}_n$ będzie zbiorem wszystkich permutacji zbioru $[n] = \{1, \dots, n\}$. Przez losową permutację rozumiemy taką operację która generuje każdą permutację z $\mathcal{S}_n$ z prawdopodobieństwem $1/n!$.

2.1 Algorytm na losową permutację

Podamy teraz algorytm na generowanie losowej permutacji liczb $1, \dots, n$. A więc celem jest wygenerowanie 'losowej' permutacji tych liczb, tj. takiej, że każdy wynik można otrzymać z prawdopodobieństwem $1/n!$. Zakładamy, że mamy do dyspozycji generator liczb losowych $U_1, U_2, \dots$, tj. ciąg niezależnych zmiennych losowych o jednakowym rozkładzie jednostajnym $U(0, 1)$. Podajemy teraz algorytm na generowanie losowej permutacji ciągu $1, \dots, n$

Losowa permutacja

1. podstaw $t = n$ oraz $A[i] = i$ dla $i = 1, \dots, n$;
2. generuj liczbę losową u pomiędzy 0 i 1 ;
3. podstaw $k = 1 + \lfloor tu \rfloor$; zamień $A[k]$ z $A[t]$;
4. podstaw $t = t - 1$;
jeśli $t > 1$, to powrót do kroku 2;
w przeciwnym razie stop i $A[1], \dots, A[n]$
podaje losową permutację.

Złożoność tego algorytmu jest $O(n)$ (dlaczego?). Algorytm jest konsekwencją zadania 5.2.

W MATLAB-ie generujemy losową permutację za pomocą instrukcji
`randperm`

2.2 Tasowanie kart

Tasowanie kart jest procedurą która prowadzi do permutacji losowej. Mianowicie, powiedzmy mamy talię 52 kart. Chcemy aby były uporządkowane “losowo”, tzn. każda konfiguracja miała prawdopodobieństwo $1/52!$. W kartach przeprowadza się procedurę tasowania kart aby dostać uporządkowanie losowe. Oczywiście musimy formalnie zdefiniować matematycznie jak zdefiniować tasowanie.

W ostatnich latach napisano sporo prac o teoretycznych własnościach tasowania kart. Talię kart można utożsamić z ciągiem liczb $[m] = (1, \dots, m)$, które można uporządkować na $m!$ sposobów, noszących nazwy permutacji. Rozpatrzmy zbiór $\mathcal{S}_m$ wszystkich permutacji ciągu $[m]$. Permutacją losową będziemy nazywali odwzorowanie $X : \Omega \rightarrow \mathcal{S}_m$ z przestrzeni probabilistycznej $(\Omega, \mathcal{F}, \mathbb{P})$ takie, że $\mathbb{P}(X = \pi) = 1/m!$ dla dowolnej permutacji $\pi \in \mathcal{S}_m$. Takie odwzorowanie możemy otrzymać przez algorytm na permutację losową z 2.1 na podstawie zadania 5.2. Możemy też wprowadzać różne sposoby tasowania kart, które naśladują tasowania prawdziwych kart. Każde tasowanie jest losowe, a więc zaczynając od $X_0 = \pi$, po n tasowaniach mamy permutacje X_n . Pytaniem jest jak daleko X_n jest od permutacji losowej. Możemy na przykład zmierzyć odległość wahania

$$d_n = \frac{1}{2} \sum_{\pi \in \mathcal{S}_m} |\mathbb{P}(X_n = \pi) - \frac{1}{m!}|.$$

Można wymyślać różne schematy tasowania kart. Niech $\pi_1, \pi_2, \dots, \pi_m$ będzie talią m kart.

Proste tasowanie kart ² W każdym kroku bierzemy kartę z góry i wkładamy ją losowo do talii. Możemy to sformalizować następująco. Oznaczmy miejsca między kartami $k_1 \sqcup k_2 \sqcup \dots \sqcup k_m \sqcup$. W każdym kroku wybieramy losowo jedno z tych miejsc i następnie wkładamy w to miejsce górną kartę.

Losowe inwersje ³ Przez losową inwersję permutacji $\pi = (\pi_1, \dots, \pi_m)$ nazywamy wylosowanie pary π_i, π_j . Możemy to robić albo wyciągając dwie jednocześnie (parę losową), lub pojedynczo, i wtedy gdy dwie są takie same to nic nie robimy. Powtarzamy losowania n razy. Ponieważ na każdym etapie wynik jest losowy, a więc po n krokach mamy permutację X_n .

²top-to-random

³random exchange

Time reverse riffle shuffle Generujemy odwrotne tasowanie. Oznaczamy tył każdej karty losowo i niezależnie liczbą 0 lub 1. Następnie wyciągamy z talii wszystkie oznaczone zerem, zachowując ich uporządkowanie i kładziemy je na pozostałych kartach.

Pytaniem jest ile należy zrobić tasowań aby być "blisko" permutacji losowej.

Dla *time reverse riffle shuffle* z $m = 52$ tak jak jest to w prawdziwej talii dokładne wielkości d_n są następujące:

n	1	2	3	4	5	6	7	8	9	10
d_n	1.000	1.000	1.000	1.000	.924	.614	.334	.167	.085	.043

(wg. pracy Diaconisa [11]). Te wyniki sugerują, że $n = 7$ powinno wystarczyć do otrzymania w miarę bliskiego permutacji losowej (czyli dobrze potasowanej talii). To było treścią artykułu w New York Times ([24] pod takim tytułem. Tabela powyższa pokazuje pewien problem z którym często spotykamy się gdy symulujemy charakterystykę procesu, który się stabilizuje, i chcemy obliczyć wielkość w warunkach stabilności. Jest więc pytanie kiedy osiągneliśmy warunki stabilności. Do tego powrócimy później. Symulacja liczb w tabelce byłaby trudna dla $m = 52$, ale możemy spróbować policzyć d_n przez symulacje dla mniejszych m .

Wspólną cechą tych schenmtów losowań, jest to, że po formalizacji matematycznej, prowadzą do teorii łańcuchów Markowa. To, że tasowania zbiegają do permutacji losowej jest prostą konsekwencją podstawowych twierdzeń tej teorii. Natomiast odpowiedź na pytanie jak szybka jest ta zbieżność jest nietrywialna i wokół niej koncentrują się badania.

3 Generatory liczb losowych

3.1 Generatory kongruencyjne

Najprostszym przykładem generatora liczb pseudo-losowych jest ciąg u_n zadany przez x_n/M , gdzie

$$x_{n+1} = (ax_n + c) \mod M. \quad (3.1)$$

Należy wybrać

M , moduł;	$0 < M$
a , mnożnik;	$0 \leq a < M$
c , krok	$0 \leq c < M$
x_0 , wartość początkowa;	$0 \leq x_0 < M$,

(ang. odpowiednio *modulus*, *multiplier* and *increment*). Aby otrzymać ciąg z przedziału $(0, 1)$ musimy teraz wybrać $u_n = x_n/M$. Ciąg (x_n) ma okres nie dłuższy niż M .

Następujące twierdzenie (patrz twierdzenie A na stronie 17 swojej książki Knuth [23]) podaje warunki na to aby kongruencja liniowa miała okres M .

Twierdzenie 3.1 *Niech $b = a - 1$. Na to aby ciąg (x_n) zadany przez (3.1) miał okres równy M potrzeba i wystarcza aby*

- c jest względnie pierwsze z M ,
- b jest wielokrotnością p dla każdej liczby pierwszej p dzielącej M ,
- jeśli 4 jest dzielnikiem M , to $a \equiv 1 \pmod{4}$.

Podamy teraz przykłady generatorów liczb pseudolosowych otrzymanych za pomocą metody kongruencji liniowej.

Przykład 3.2 Można przyjąć $M = 2^{31} - 1 = 2147483647$, $a = 7^5 = 16867$, $c = 0$. Ten wybór był popularny na 32-bitowych komputerach. Innym przykładem proponowanego ciągu o niezłych własnościach jest wybór $M = 2147483563$ oraz $a = 40014$. Dla 36-bitowych komputerów wybór $M = 2^{35} - 31$, $a = 5^5$ okazuje się dobrym.⁴

Przykład 3.3 [Resing [37]]. Jeśli $(a, c, M) = (1, 5, 13)$ i $X_0 = 1$, to mamy ciąg (X_n) : 1, 6, 11, 3, 8, 0, 5, 10, 2, 7, 12, 4, 9, 1, ..., który ma okres 13.

Jeśli $(a, c, M) = (2, 5, 13)$, oraz $X_0 = 1$, to otrzymamy ciąg 1, 7, 6, 4, 0, 5, 2, 9, 10, 12, 3, 11, 1, ... który ma okres 12. Jeśli natomiast $X_0 = 8$, to wszystkie elementy tego ciągu są 8 i ciąg ma okres 1.

Uogólniona metoda kongruencji liniowej generuje ciągi według rekurencji

$$x_{n+1} = (a_1 x_n + a_2 x_{n-1} + \dots + a_k x_{n-k}) \pmod{M}$$

Jeśli M jest liczbą pierwszą, to odpowiedni wybór stałych a_j pozwala na otrzymanie okresu M^k . W praktyce wybiera się $M = 2^{31} - 1$ na komputerach 32 bitowych.

⁴Podajemy za Asmussenem i Glynem oraz Rossem (str.37)

Terminologia i skróty Generatory oparte o metodę kongruencji liniowej noszą nazwę liniowych generatorów kongruencji liniowej (ang. linear congruential generator). Będziemy je oznaczać przez $LCG(M, a, c)$. W przypadku generatorów opartych o uogólnioną kongruencję będziemy je oznaczali przez $GLCG(M, (a_1, \dots, a_k))$.

Za L'Ecuyer [29] podamy teraz kilka popularnych generatorów używanych w komercyjnych pakietach.

Przykład 3.4

Java

$$\begin{aligned} x_{i+1} &= (25214903917x_i + 11) \mod 2^{48} \\ u_i &= (2^{27} \lfloor x_{2i}/2^{22} \rfloor + \lfloor x_{2i+1}/2^{21} \rfloor) / 2^{53}. \end{aligned}$$

VB

$$\begin{aligned} x_i &= (1140671485x_{i-1} + 12820163) \mod 2^{24} \\ u_i &= x_i / 2^{24}. \end{aligned}$$

Excel

$$u_i = (0.9821U_{i-1} + 0.211327) \mod 1.$$

Jeśli $c = 0$, to generator nazywa się liniowy multiplikatywny kongruencyjny. W Lewis et al.⁵ zaproponowano następujące parametry takiego generatora: $M = 2^{31} - 1$, $a = 1680$.

Spotyka się też generatory bazujące na kongruencjach wyższych rzędów, np. kwadratowych czy sześciennych. Przykłady takich generatorów można znaleźć w Rukhin et al 2010.⁶

Generator kwadratowej kongruencji I (QCG I). Ciągi zadan są rekurencją

$$x_{i+1} = x_i^2 \mod p$$

⁵Lewis, P.A.P., Goodman, A.S. and Miller, J.M. (1969)

⁶Rukhin, A., Soto, J. Nechvatal, J, Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J. and Vo, S. 2010, A statistical test suite for random number generators for cryptographic applications. National Institute of Standard and Technology, Special Publication 800-22, Rev 1a.

gdzie p jest 512 bitową liczbą pierwszą zadaną w układzie szesnastkowym

$p = 987b6a6bf2c56a97291c445409920032499f9ee7ad128301b5d0254aa1a9633$
 $fdbd378d40149f1e23a13849f3d45992f5c4c6b7104099bc301f6005f9d8115e1$

Generator kwadratowej kongruencji II (QCG II). Ciąg jest zadany rekurencją

$$x_{i+1} = 2x_i^2 + 3x_i + 1 \pmod{2^{512}}.$$

Generator sześcienniej kongruencji (CCG). Ciąg jest zadany rekurencją

$$x_{i+1} = 2x_i^3 \pmod{2^{512}}.$$

Generator Fibonacciego. Generator Fibonacciego zadany jest rekurencją

$$x_n = x_{n-1} + x_{n-2} \pmod{M} \quad (n \geq 2).$$

Modyfikacją dającą lepsze własności niezależności jest opóźniony generator Fibonacciego (lagged Fibonnaci generator) zadany rekurencją

$$x_n = x_{n-k} + x_{n-l} \pmod{M} \quad (n \geq \max(k, l)).$$

Inną modyfikacją jest *subtractive lagged Fibonnaci generator* zadany rekurencją

$$x_n = x_{n-k} - x_{n-l} \pmod{M} \quad (n \geq \max(k, l)).$$

Knuth zaproponował ten generator z $k = 100$, $l = 37$ oraz $M = 2^{30}$.

3.2 Generatory w MATLABie

W MATLABie⁷ są zaimplementowane 3 generatory: 'state', 'seed' i 'twister'. Jeśli nic nie zostanie zadeklarowane to jest w użyciu 'state'. W przeciwnym razie musimy zadeklarować wpisując komendę `rand(method,s)`, gdzie

- w 'method' należy wpisać jedną z metod 'state', 'seed' lub 'twister',
- wartość 's' zależy od zadeklarowanej metody:
 - w przypadku 'state' or 'twister', s musi być albo liczbą całkowitą od 0 do $2^{32} - 1$ lub wartością wyjściową `rand(method)`,
 - w przypadku 'seed' wielkość s musi być albo liczbą całkowitą od 0 do $2^{31} - 1$ lub wartością wyjściową `rand(method)`.

⁷MATLAB Version 7.2.0.294 (R2006a).

Przykład 3.5 Proponujemy na początek następujący eksperyment w MATLABie. Po nowym wywołaniu MATLAB napisać rozkaz

```
rand
```

a następnie

```
rand('state',0).
```

W obu przypadkach dostaniemy taką samą liczbę 0.9501. Natomiast

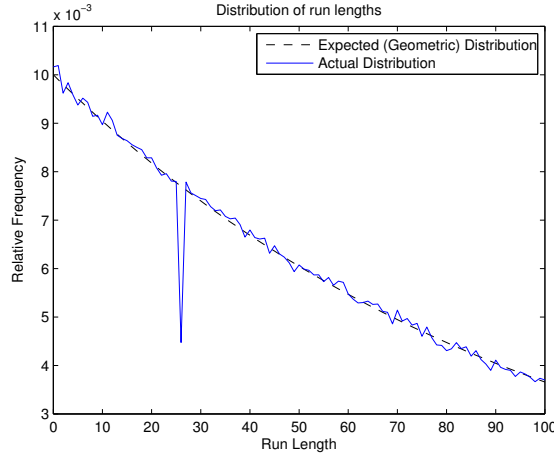
```
rand('twister',0)
```

Przykład 3.6 Dla generatora 'state' znaleziono przykłady na odchylenie od losowości, o których chcemy teraz opowiedzieć. Należy jednak zdawać sobie sprawę, że podobne przykłady można znaleźć dla wszystkich generatorów. Rekomendujemy czytelnikowi zrobienie następującego eksperymentu, w którym dla kolejnych liczb losowych $U_1, \dots, U_n$ będziemy rejestrować odstępów $X_1, X_2, \dots$ pomiędzy liczbami mniejszymi od δ . Oczywiście z definicji takie odstępów mają rozkład geometryczny z funkcją prawdopodobieństwa $\delta(1-\delta)^k$, $k = 0, 1, \dots$. Ponadto kolejne odstępów $X_1, X_2, \dots$ są niezależne o jednakowym rozkładzie geometrycznym. Jeśliby więc zrobić histogram pojawiania się poszczególnych długości to powinien dla dużej liczby wygenerowanych liczb losowych przypominać funkcję $\delta(1-\delta)^k$. Polecamy więc czytelnikowi przeprowadzenie następujących eksperymentów w MATLABIE; najpierw dla metody 'state' a potem 'twister', z różnymi n, δ według następującego algorytmu:

```
rand('state',0);
n = 5*10^7;
delta = .01;
runs = diff(find(rand(n,1)<delta))-1;
y = histc(runs, 0:100) ./ length(runs);
plot(0:100,(1-delta).^(0:100).*delta,'k--', 0:100,y,'b-');
title('Distribution of run lengths')
xlabel('Run Length'); ylabel('Relative Frequency');
legend({'Expected (Geometric) Distribution' 'Actual Distribution'})
```

Dla pierwszej z tych metod możemy zaobserwować tajemniczą odchyłkę przy $k = 27$; patrz rys. 3.2.⁸

⁸plik przykładZlyGenState.m



Rysunek 3.2: Histogram pojawiania się małych wielkości; generator 'state', $n = 5 * 10^7$, $\delta = 0.01$

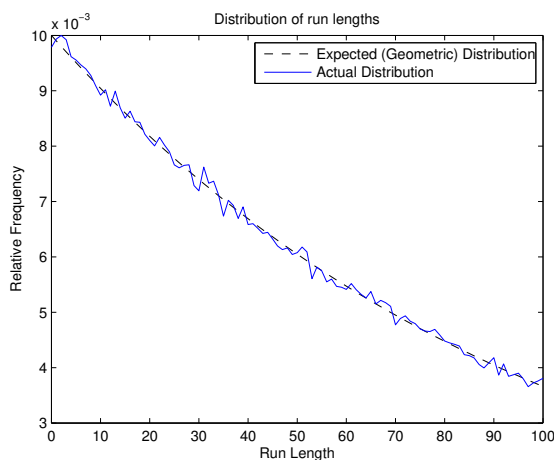
Jest to spowodowane tym, że generator niepoprawnie generuje małe wartości. Można się o tym przekonać eksperymentując z δ . Tak naprawdę, dobrze dopasowaną do zadanego $n = 5 * 10^7$ jest $\delta = 0.05$ jednakże wtedy rysunki się zlewają. Natomiast dla bardzo małych δ należałoby zwiększyć n . Wyjaśnieniem jak dobrać n zajmiemy się w następnych rozdziałach. Dla porównania na rys. 3.3 pokazujemy wynik dla podobnej symulacji ale z generatorem 'twister'.⁹

Dla kontrastu na rys. 3.4 podajemy wynik symulacji jeśli zamiast odstępów pomiędzy liczbami mniejszymi od $\delta = 0.01$ rozważamy odstępów pomiędzy liczbami większymi od $\delta = 0.99$.¹⁰ W tym przypadku odstępów $X_1, X_2, \dots$ mają ten sam rozkład geometryczny.

Generator Mersenne Twister Jednym z generatorów używanych przez MATLAB (ale nie tylko bo jest zaimplementowany w wielu innych pakietach) jest generator Mersenne Twister MT19937, który produkuje ciągi 32 bitowe liczb całkowitych. Charakteryzuje się on bardzo długim okresem $2^{19937} - 1$. Jest to liczba pierwsza tzw. Mersenne'a. Dla wielu zastosowań wystarczy już krótszy okres. Generator wymaga bardzo dużej przestrzeni CPU i jest przez to powolny. Problemem jest też wybór ziarna, które powinno być wysoce

⁹plik przykladZlyGenTwister.m

¹⁰plik przykladZlyGenStateAnty.m



Rysunek 3.3: Histogram pojawiania się małych wielkości; generator 'twister', $n = 5 * 10^7$, $\delta = 0.01$

nieregularne, ponieważ w przypadku zbyt regularnego, aby otrzymać wyniki zbliżone do postulowanej imitacji losowego ciągu bitów, musi się wykonać wiele iteracji.

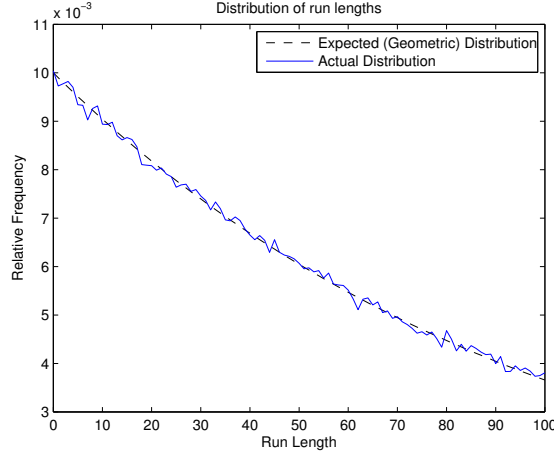
3.3 Inne generatory

Wiele problemów przy rozwiązaniu wymaga bardzo długich okresów. Takie zapewniają na przykład tzw. generatory mieszane.

Generator mieszany ¹¹ Na przykład niech

1. $x_n = (A_1 x_{n-2} - A_2 x_{n-3}) \bmod M_1$
2. $y_n = (B_1 y_{n-1} - B_2 y_{n-3}) \bmod M_2$
3. $z_n = (x_n - y_n) \bmod M_1$
4. jeśli $z_n > 0$, to $u_n = z_n / (M_1 + 1)$; w przeciwnym razie $u_n = M_1 / (M_1 + 1)$.

¹¹Kelton et al (2004) W.D. Kelton, R.P.Sadowski and D.T. Sturrock (2004)



Rysunek 3.4: Histogram pojawiania się dużych wielkości; generator 'state', $n = 5 * 10^7$, $\delta = 0.01$

z wyborem $M_1 = 4294967087$, $M_2 = 4294944443$, $A_1 = 1403580$, $A_2 = 810728$, $B_1 = 527612$, $B_2 = 1370589$. Jako warunki początkowe należy przyjąć pierwsze trzy x-sy oraz trzy y-ki. Algorytm został stworzony przez L'Ecuyera [28] przez komputerowe dopasowanie parametrów zapewniające optymalne własności generatora.

RC4 RC4 generuje pseudolosowy strumień bitów (tzw. strumień szyfrujący) i jest używany do szyfrowania. Nas interesuje tutaj jedynie część dotycząca generowania. Będziemy rozważali algorytm RC4 który generuje liczby $x_i \in \bar{M}$, gdzie $M = 2^k$. W standardowej wersji $k = 8$. W pamięci wewnętrznej przechowuje się (S, i, j) , gdzie S jest permutacją $\bar{M}$ i $i, j \in \bar{M}$.

W celu wygenerowania strumienia szyfrującego wykorzystuje się tajny stan początkowy, tzw. zewnętrzny klucz $K = (K(1), \dots, K(M))$. Czasami K jest M bitową liczbą; $K(i) \in \{0, 1\}$. W przypadku $k = 8$, K jest 32-bitową liczbą. Klucz może też mieć inną długość - keylen. W przypadku $k = 8$ poleca się keylen pomiędzy 40 a 128 bitów. Można też otrzymać K z aktualnego czasu komputera. Kluczem też może być ciąg $K(i) \in \bar{M}$. Na podstawie klucza generuje się początkowe S, i, j (tzw. KSA *key scheduling algorithm*). Następnie tworzony jest strumień bitów wyjściowych za pomocą algorytmu pseudolosowej generacji (PRGA - *pseudo-random generation algorithm*).

- (i) **Procedura inicjowania klucza - KSA.** Początkowo w tablicy S tworzona jest permutacja identycznościowa. Następnie S jest przetwarzane w 256 iteracjach i mieszane jednocześnie z bajtami klucza. Przez K oznaczamy klucz; keylen jest jego długością.

```

for i=0:M-1
    S(i)= i
j= 0
for i=0:M-1
    j= (j + S(i) + K(i mod keylen)) mod M
    zamień(S(i),S(j))

```

- (ii) **Algorytm pseudolosowej generacji - PRGA** Algorytm pseudolosowej generacji modyfikuje stan tablicy S i generuje bajt strumienia szyfrującego tyle razy, ile wynosi długość tekstu oryginalnego. Za każdym razem algorytm zwiększa i o 1, dodaje do j wartość w S wskazywaną przez i , zamienia ze sobą wartości $S(i)$ i $S(j)$ oraz jako wynik daje wartość z S wskazywaną przez $S(i) + S(j) \pmod{M}$. Każda wartość S jest zamieniana co najmniej raz na każde 256 iteracji.

```

i = 0
j = 0
r = 1
while true do :
    i = (i + 1) mod M
    j = (j + S(i)) mod M
    r = r + 1
    zamień(S(i),S(j))
    wynik x_r = S((S(i) + S(j)) mod M)
end while

```

3.4 Generatory pseudolosowych ciągów bitów.

Przez ciągi pseudolosowe bitów rozumiemy ciągi zero-jedynkowe. Wiele generatorów liczb pseudo-losowych jest generowany przez losowe ciągi bitów, które są dzielone na bloki zadanej długości, i następnie zamieniane na odpowiedni format. Na przykład na liczby z przedziału $[0, 1]$. Pytanie odwrotne

jak z ciągu niezależnych zmiennych losowych o jednakowym rozkładzie $U(\bar{M})$ otrzymać o wartościach w $\bar{M}$ otrzymać losowy ciąg bitów. Można to zrobić jeśli $M = 2^k$. Szczegóły można znaleźć w zadaniu 5.1.

Innym powodem dla którego potrzeba rozpatrywać (pseudo)losowe ciągi bitów jest kryptografia.

Zanim podamy konkretne generatory musimy przypomnieć/wprowadzić kilka pojęć. Będziemy mieli do czynienia z ciągami bitów, tj. z ciągami zero-jedynkowymi, zazwyczaj ustalonej długości. Jest to związane z konstrukcją komputera, gdzie takie ciągi są trzymane w tzw. rejestrach. Najczęściej spotykane są rejestry 32 lub 64 bitowe.

Zrobimy teraz przegląd operacji na rejestrach. Niech $p, q \in \{0, 1\}$ a $\mathbf{p} = (p_1, \dots, p_n)$ i $\mathbf{q} = (q_1, \dots, q_n)$.

- XOR (lub EOR, EXOR, $\oplus$). XOR jest prawdziwe wtedy i tylko wtedy gdy nieparzysta liczba zdań na wejściu jest prawdziwa. Operację możemy zapisać w postaci tabelki

p	q	$p \oplus q$
0	0	0
0	1	1
1	0	1
1	1	0

W przypadku dwóch ciągów bitów operacje wykonujemy na kolejnych parach. Na przykład ...

- Przesunięcia. Bit shifts. Lewy szift jest często oznaczony przez $<<$, natomiast prawy przez $>>$. Krotność jest zaznaczona przez drugi argument. Na przykład

$$x = y << 2$$

oznacza, że x jest otrzymana przez przesunięcie w lewo o dwa bity. W lewym przesunięciu na opuszczone miejsca podstawia się 0-ra.

Teraz podamy kilka przykładów generatorów wykorzystujących operacje $\oplus$ i $<<$.

Exclusive OR Generator (XORG) Zaczynamy od $2^{16}-1 = 127$ -bitowego ziarna

$$\begin{aligned} x_1, \dots, x_{127} \\ = 0001011011011001000101111001001010011011011010001000000101011111101010010 \\ 0001010110110000000000100110000101110011111111100111 \end{aligned}$$

Następne bity otrzymujemy stosując algorytm:

$$x_i = x_{i-1} \oplus x_{i-127} \quad j \geq 128.$$

4 Testowanie dobrych własności generatora

Ciąg liczb pseudolosowych otrzymany za pomocą generatora “ma wyglądać” jak ciąg niezależnych zmiennych losowych o rozkładzie jednostajnym. Po wygenerowaniu próbki liczącej tysiące, trudno coś sprawdzić jak wygląda i dlatego trzeba się odwołać do metod statystycznych. Przygotowywaniu testów do sprawdzenia losowości ciągu liczb zajmowano się od początku istnienia teorii generatorów. Jednakże obecnie teoria nabrała rumieńców gdy zauważono, jej użyteczność w kryptografii. Zauważono bowiem, że idealną sytuacją jest gdy zaszyfrowany ciąg wygląda zupełnie losowo, ponieważ odstępstwa od losowości mogą zwiększyć szanse złamania protokołu kryptograficznego, odkrycia klucza prywatnego, itp. Ze względu na wspomniane zapotrzebowanie przygotowywane są zestawy testów. Najbardziej znane to

- Diehard tests. Jest to zestaw opracowany przez George’a Marsaglia z 1995 r.
- TestU01. Zestaw opracowany przez Pierre L’Ecuyer i Richard Simard z 2007 r.
- NIST Test Suite Zestaw opracowany przez organizację *National Institute of Standards and Technology* Zestaw ten jest napisany pod kątem zastosowań w kryptografii.

Dla ilustracji zrobimy teraz przegląd kilku podstawowych metod. Musimy wpieryw sprawdzić czy generowane liczby przypominają zmienne losowe o rozkładzie jednostajnym. Do tego służą w statystyce testy zgodności: χ^2 oraz test Kołmogorowa-Smirnowa.

Będziemy przyjmować hipotezę $\mathcal{H}_0$ mówiącą, że zmienne losowe $U_1, \dots$ są niezależne o jednakowym rozkładzie $U(0,1)$. Przypomnijmy, że dla zmiennych losowych $U_1, \dots$ definiujemy *dystribuatę empiryczną*

$$\hat{F}_n(t) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(U_i \leq t), \quad 0 \leq t \leq 1.$$

W użyciu są, między innymi, następujące typy testów:

- testy równomierności – sprawdzający czy wygenerowane liczby są równomiernie rozmieszczone między zerem a jedynką,
- test serii – podobnie jak równomierności tylko dla kolejnych t -rek,
- test odstępów – notujemy odstęp między liczbami losowymi, które nie należą do $[a, b]$, gdzie $0 \leq a < b \leq 1$, i sprawdzamy zgodność z rozkładem geometrycznym.
- testy oparte na klasycznych zadaniach kombinatorycznych,
- testy oparte na własnościach relizacji błędzenia przypadkowego.

Od generatora pseudolosowych ciągu bitów należy wymagać:

- Jednostajności: jakkolwiek badany podciąg powinien średnio zawierać połowę zer i połowę jedynek,
- Skalowalności: Test zastosowany do całego ciągu powinien dać te same rezultaty dla dowolnego wybranego podciągu.
- Zgodności: Dobroć generatora powinna być testowana dla dowolnych ziaren. Jest to szczególnie ważne w kontekście zastosowań kryptograficznych. Są znane generatory (np. Mersenne twister), którego "dobre" własności zależą od losowego wybrania ziarna.

4.1 Funkcja p -wartość

Używając metod statystyki możemy weryfikować hipotezę. W naszym przypadku weryfikujemy hipotezę dotyczącą niezależności i jednakowości rozkładów (jednostajnych) sprawdzanego ciągu. Przyjmuje się poziom istotności

(błąd pierwszego rodzaju α - prawdopodobieństwo odrzucenia hipotezy zerowej, jeśli jest ona prawdziwa) i następnie dla ustalonej statystyki testowej T znajduje się obszar akceptacji. Najczęściej mamy do czynienia z testami jednostronnymi z obszarami akceptacji $T > x_0$ lub $T \leq x_0$ lub $|T| \leq t_0$. Wtedy x_0 jest dobierane tak aby odpowiednio $\mathbb{P}(T \notin \text{obszaru akceptacji}) = \alpha$.

W przeciwieństwie do tego co się uczy na statystyce, że najpierw ustala się poziom istotności testu i obszar akceptacji, do testowania generatorów zalecana jest następująca procedura. Niech T ma dystrybucję $G(x)$ i w zagadnieniu obszar akceptacji testu jest postaci $(-\infty, a]$. Obliczamy statystykę testową T dla bardzo dużej liczby prób (na przykład $n = 2^{10}$) i następnie jeśli $T = t_0$ to obliczamy wartości tzw. p -wartości¹²:

$$p = 1 - G(t_0) . \quad (4.2)$$

Czasami spotykamy testy z obszarem akceptacji postaci (a, ∞) i wtedy obliczamy lewą p wartość

$$p_1 = G(t_0) . \quad (4.3)$$

Jeśli jedna z tych wartości jest bardzo bliska 0 to może wskazywać jakiś problem z generatorem. Przez małą liczbę L'Ecuyer poleca $p = 10^{-15} - 10^{-10}$. Natomiast mamy przypadek "podejrzany" gdy p są bliskie 0.005. Jednakże wydaje się, że te zalecenia są zbyt ostre.

Ogólnie zalecane jest wygenerowanie $m = 1000 - 10000$ ciągów długości $n = 2^{20} - 2^{32}$. Można powtarzać eksperyment dla różnych $n = 2^k$ zwiększając k i obserwując czy nie ma załamania się wartości p . Po przeprowadzeniu eksperymentu otrzymujemy ciąg $p_1, \dots, p_m$ p -wartości. Przyjmujemy wartość krytyczną, na przykład zalecane w raporcie NIST [41] $\alpha = 1/100 - 1/1000$. W przypadku $\alpha = 1/100$ to spodziewamy się że co najwyżej około 1% eksperymentów obleje test (to znaczy poda małą p -wartość). A więc w przeciwnym przypadku mamy podejrzenie, że generator jest kiepski.

Powtarzamy ten eksperyment dla $n = 2^k$ obserwując czy nie ma załamania się wartości p .

Zgodność z rozkładem jednostajnym; test λ Kołmogorowa

Teraz zajmiemy się testowaniem hipotezy, że rozkład brzegowy U_i jest $U(0,1)$. Do tego służą testy zgodności, na przykład test Kołmogorowa-Smirnowa.

Generujemy liczby $U_1, \dots, U_n$, i definiujemy odpowiadającą im

¹² p -value

Pamiętając, że naszą hipotezą jest rozkład jednostajny $U(0, 1)$, tj. $F(t) = t$ dla $t \in (0, 1)$, naszą statystyką testową jest

$$D_n = \sup_{0 \leq t \leq 1} |\hat{F}_n(t) - t|.$$

Z twierdzenia Gliwienko–Cantelli mamy, że jeśli U_i są rzeczywiście o rozkładzie jednostajnym, to z prawdopodobieństwem 1 ciąg D_n zmierza do zera. Natomiast unormowane zmienne $\sqrt{n}D_n$ zbiegają według rozkładu do rozkładu λ -Kolmogorowa, tj.

$$\lim_{n \rightarrow \infty} \mathbb{P}(\sqrt{n}D_n \leq t) = K(t) = \sum_{j=-\infty}^{\infty} (-1)^j \exp(-2j^2 t^2), \quad t > 0.$$

Rozkład ten jest stablicowany, patrz np tablice Zielińskich [52]. W szczególności mamy $\lambda_{0.1} = 1.224$, $\lambda_{0.05} = 1.358$ oraz $\lambda_{0.01} = 1.628$, gdzie

$$1 - K(\lambda_\alpha) = \alpha.$$

Do wyznaczenia D_n mamy następujący algorytm. Niech $U_{(1)}, \dots, U_{(n)}$ będzie statystyką porządkową z próby $U_1, \dots, U_n$, tj. ich niemalejące uporządkowanie oraz

$$\begin{aligned} D_n^+ &= \max_{1 \leq i \leq n} \left(\frac{i}{n} - U_{(i)} \right), \\ D_n^- &= \max_{1 \leq i \leq n} \left(U_{(i)} - \frac{i-1}{n} \right). \end{aligned}$$

Wtedy

$$D_n = \max(D_n^+, D_n^-).$$

Innym testem zgodności jest test χ^2 . Pokażemy jego zastosowanie do sprawdzenia nie tylko zgodności z rozkładem jednostajnym ale również niezależności.

Test częstości par ¹³ Chcemy przetestować czy ciąg $u_1, u_2, \dots \in [0, 1]$ jest realizacją z próby prostej rozkładzie jednostajnym na $[0, 1]$ $U_1, U_2, \dots$. Zamiast ciągu liczb rzeczywistych $u_1, \dots \in [0, 1]$ przechodzimy do ciągu

¹³Knuth v. ang. str. 60.

$y_1, y_2, \dots \in \{0, 1, \dots, M-1\}$ stosując przekształcenie $y_j = \lfloor Mu_j \rfloor$. W zależności od komputera musimy wybrać $M = 2^k$. Na przykład stosuje się $M = 64 = 2^6$ gdy komputer pracuje w systemie dwójkowym, i wtedy y_k reprezentuje 6 najbardziej znaczących bitów w dwójkowej reprezentacji u_j . Dla naszego testu par rozpatrujemy ciąg n elementowy

$$(y_1, y_2), (y_3, y_4), \dots, (y_{2n-1}, y_{2n}).$$

Wtedy liczymy X_{qr} liczbę powtórzeń $y_{2j-1} = q, y_{2j} = r$ dla każdej pary $0 \leq q, r \leq M-1$ i stosujemy test χ^2 z $M^2 - 1$ stopniami swobody z prawdopodobieństwem teoretycznym $1/M^2$ dla każdej pary (q, r) .

Test odstępów Przykład pokazujący nieprawidłowości generatora 'state' w MATLABie (patrz podrozdział 3.2 wykorzystywał odstępy pomiędzy małymi wielkościami. Ogólnie *test odstępów* bada odstępy pomiędzy pojawieniem się wielkości $u_j \in (\alpha, \beta]$, gdzie $0 \leq \alpha < \beta \leq 1$. Rozkład odstępu jest geometryczny z prawdopodobieństwem odstępu równym j wynoszącym $p(1-p)^{j-1}$ ($j = 1, 2, \dots$). Tutaj $p = \beta - \alpha$ i testujemy zgodność rozkładem chi kwadrat.

4.2 Testy oparte na schematach urnowych

Cała grupa testów jest oparta na schemacie rozmieszczenia n kul w k urnach. Zdefiniujemy najpierw kostkę w przestrzeni $[0, 1)^t$. Dzielimy każdy z t boków na d (najlepiej $d = 2^\ell$) równych odcinków $[i/d, (i+1)/d)$, $i = 0, \dots, d-1$, i następnie produkując je t razy otrzymujemy d^t małych kostek. Oznaczmy $k = d^t$, i ponumerujmy kostki od 1 do k . To będą nasze urny. Teraz generujemy nt liczb losowych $U_1, \dots, U_{nt}$ i grupujemy je w n wektorów z $[0, 1)^t$:

$$\begin{aligned} \mathbf{U}_1 &= (U_1, \dots, U_t), \mathbf{U}_2 = (U_{t+1}, \dots, U_{2t}), \\ &\dots, \mathbf{U}_n = (U_{(n-1)t+1}, \dots, U_{nt}). \end{aligned}$$

Każdy taki d wymiarowy wektor traktujemy jako kulę.

Test serii Niech X_j będzie liczba wektorów $\mathbf{U}_i$, które wpadną do kostki $j = 1, \dots, k$ i zdefiniujmy statystykę

$$\tilde{\chi}^2 = \sum_{j=1}^k \frac{(X_j - n/k)^2}{n/k}.$$

Jeśli jest prawdziwa hipoteza $\mathcal{H}_0$ mówiąca, że zmienne losowe są niezależne o jednakowym rozkładzie $U(0,1)$, to $\tilde{\chi}^2$ ma średnią $\mathbb{E} \chi^2 = \mu = k - 1$ oraz $\text{Var} \chi^2 = \sigma^2 = 2(k - 1)(n - 1)$. W przypadku, gdy $t = 1$ mamy test zgodności z rozkładem brzegowym jednostajnym, bez testowania niezależności. Ze względu na dużą liczbę kostek gdy t jest duże, używa się testu dla $t = 2$, tj. gdy testujemy jednostajność poszczególnych liczb i parami niezależność. Jeśli chcemy sprawdzić niezależność dla większych t to stosujemy inne testy, jak np. test kolizji czy test urodzin. Na czym one polegają wyjaśnimy w następnym podrozdziale. Wiadomo, że

- jeśli $n \rightarrow \infty$ i k jest ustalone to $\tilde{\chi}^2$ dąży do rozkładu chi-kwadrat z $k - 1$ stopniami swobody,
- jeśli $n \rightarrow \infty$ i $k \rightarrow \infty$ tak aby $n/k \rightarrow \gamma$, to $T = (\tilde{\chi}^2 - \mu)/\sigma$ dąży do rozkładu standardowego normalnego. Zalecane jest aby $0 \leq \gamma < 5$, bo w przeciwnym razie $\tilde{\chi}^2$ jest lepiej aproksymowany przez rozkład chi-kwadrat.¹⁴

Do przyjęcia lub odrzucenia hipotezy stosujemy test przy $k \gg n$, odrzucając hipotezę jeśli χ^2 jest bardzo małe lub duże.¹⁵

Teraz przejdziemy do testów powstałych na bazie klasycznych zadań kombinatorycznych rachunku prawdopodobieństwa.

Test pokerowy Tak jak w teście częstotliwości par rozpatrujemy ciąg liczb $y_1, y_2, \dots, y_{5n} \in \bar{M}$, który przy założeniu hipotezy zerowej, jest realizacją ciągu zmiennych losowych $Y_1, Y_2, \dots$ o jednakowym rozkładzie jednostajnym $U(\bar{M})$. Rozpatrzmy pierwszą piątkę $Y_1, \dots, Y_5$. Wprowadzamy następujące wzorce, które są rozsądną modyfikacją klasyfikacji pokerowych. Chociaż ideą były wzorce z pokera (patrz zadanie 5.9), ale te wzorce są trudne w implementacji. Definiujemy więc za Knuth'em [23]¹⁶

5 różnych = wszystkie różne,

4 różne = jedna para,

3 różne = dwie pary lub trójkę,

¹⁴Sprawdzić w książkach statystycznych

¹⁵Referencje

¹⁶w ang wydaniu 1981 sekcja 3.3.2, p. 62

2 różne = full lub czwórkę

1 różna = wszystkie takie same.

Musimy policzyć prawdopodobieństwo teoretyczne p_r , że w piątce $Y_1, \dots, Y_5$ będziemy mieli r różnych ($r = 1, \dots, 5$). Niech $S_2(5, r)$ oznacza liczbę możliwych sposobów podziału zbioru pięcio elementowego na r niepustych podzbiorów (są to tzw. liczby Stirlinga 2-go rodzaju). Zostawiamy czytelnikowi do pokazania, że $S_2(5, 1) = 1, S_2(5, 2) = 15, S_2(5, 3) = 25, S_2(5, 4) = 10, S_2(5, 5) = 1$. Ponieważ liczba różnych ciągów 5 elementowych wynosi $d(d-1) \cdots (d-r+1)$, więc szukane prawdopodobieństwo równa się

$$p_r = \frac{M(M-1) \cdots (M-r+1)}{M^5} S_2(5, r).$$

Dzielimy obserwacje na piątki $(y_{5(i-1)+1}, y_{5(i-2)+1}, \dots, y_{5i})$ ($i = 1, \dots, n$) i następnie obliczamy próbkowe częstotliwości $\pi_1, \dots, \pi_r$. Testujemy zgodność z prawdopodobieństwami teoretycznymi p_r testem chi kwadrat z 4 ma stopniami swobody.

W następnych testach będziemy rozważać losowe rozmieszczenie kul w komórkach. Za wzorzec przyjmujemy zadanie o urodzinach znane z wykładów rachunku prawdopodobieństwa.

Test kolizji Następujące zadanie jest podstawą tzw. *testu kolizji*, używanego do sprawdzania generatorów. Rozmieszczamy 'losowo' po kolei n kul w k (tak jak w zadaniu o urodzinach) komórkach (interesuje nas przypadek gdy $k \gg n$). Rozważamy liczbę wszystkich kolizji C , gdzie *kolizję* definiujemy następująco. Po rozmieszczeniu kolejnej kuli, jeśli komórka do tej pory była pusta, to zapisujemy, że od tego momentu jest zajęta. Jeśli dla kolejnej wylosowanej kuli któraś komórka jest zajęta, to odnotowujemy *kolizję*. Podobnie jak w teście pokerowym można plokazać, że

$$\mathbb{P}(C = c) = \frac{k(k-1) \cdots (k-n+c+1)}{k^n} S_2(n, n-c),$$

gdzie $S_2(n, d)$ jest liczbą Stirlinga drugiego rodzaju.

Polecamy czytelnikowi sprawdzenie czy jest prawdziwa aproksymacja $n^2/(2k)$ dla $\mathbb{E}C$ przy czym k jest dużo większe niż n (patrz Knuth [1]). Na przykład dla $k = 2^{20}$ i $k = 2^{14}$ można pokazać następujące prawdopodobieństwa

$p = \mathbb{P}(C \leq c)$:

c	101	108	119	126	134	145	153
p	.009	.043	.244	.476	.742	.946	.989

W zadaniach o kulach i urnach rozpatruje się problem zajętości, to znaczy rozpatruje się zmienną Z niezajętych kul. Zostawiamy czytelnikowi do pokazania, że

$$Z = k - n + C$$

co pozwala wykorzystać znane fakty z klasycznego zadania o rozmieszczeniu.

¹⁷ Mianowicie wiadomo (patrz zadanie 5.10), że gdy k, n rosną tak, że $\lambda = ke^{-\frac{n}{k}}$ pozostaje ograniczone, to dla każdego c

$$\mathbb{P}(Z = c) \rightarrow \frac{\lambda^c}{c!} e^{-\lambda}.$$

A więc

$$\mathbb{E} Z \sim ke^{-\frac{n}{k}}$$

skąd mamy

$$\begin{aligned} \mathbb{E} C &= \mathbb{E} Z - k + n \\ &\sim ke^{-\frac{n}{k}} - k + n = k\left(1 - \frac{n}{k} + \frac{n^2}{2k^2} + o\left(\frac{n^2}{k^2}\right)\right) - k + n \\ &= \frac{n^2}{2k} + o\left(\frac{n^2}{k}\right). \end{aligned}$$

Na tej podstawie zbudowano następujący test do zbadania losowości ciągu $y_1, \dots, y_{20n}$. Na przykład niech $n = 2^{14}$ i $k = 2^{20}$. Wtedy możemy użyć testu do sprawdzenia generatora dla 20-sto wymiarowych wektorów: $v_j = (y_{20(j-1)+1}, \dots, y_{20j})$ dla $1 \leq j \leq n$.¹⁸

Dni urodzin W rozdziale 2-gim książki Fellera [13] mamy zadanie o dniach urodzin. Rozważamy grupę (na przykład studencką) n osób i notujemy ich dzień urodzin. Przyjmujemy, że jest $k = 365$ dni w roku. Zakładając stałą częstość urodzin w ciągu roku, zadanie to jest zadaniem o rozmieszczeniu losowym n kul w k komórkach. Klasycznym pytaniem jest o prawdopodobieństwo przynajmniej jednej sytuacji wielokrotnych urodzin, tj. o prawdopodobieństwo, że przynajmniej w jednej komórce mamy więcej niż jedną

¹⁷Uzupełnic z Fellera t.1

¹⁸Knuth str. 73–75. Praca Tsang et al

kulę. To prawdopodobieństwo łatwo da się obliczyć

$$p = 1 - \frac{365!}{(365 - n)!(365)^n}.$$

Inne możliwe pytanie, na które można łatwo odpowiedzieć przez symulację są następujące.

1. Jakie jest prawdopodobieństwo, że wśród n losowo wybranych osób przynajmniej dwie mają urodziny w przeciągu r dni jeden od drugiego?
2. Przypuśćmy, że osoby pojawiają się kolejno. Jak długo trzeba czekać aby pojawiły się dwie mające wspólne urodziny?

Test odstępów dni urodzin Innym znanym testem sprawdzającym dobroć generatorów jest tzw. *test odstępów dni urodzin*.¹⁹ Test ten jest oparty na statystyce K zdefiniowanej następująco. Zapisujemy dni urodzin kolejnych osób $Y_1, \dots, Y_n \in \bar{k} = \{1, \dots, k\}$ i ustawiamy w porządku niemalejącym $Y_{(1)} \leq \dots \leq Y_{(n)}$. Następnie definiujemy, tzw. *odstępy*

$$S_1 = Y_{(2)} - Y_{(1)}, \dots, S_{n-1} = Y_{(n)} - Y_{(n-1)}$$

i K będzie liczbą równych odstępów Tj. ile razy mamy równość pomiędzy S -ami. Okazuje się, że jeśli n jest duże i $\lambda = n^3/(4k)$ jest małe, to przy założeniu naszej hipotezy $\mathcal{H}_0$, zmienna K ma w przybliżeniu rozkład Poissona z parametrem λ . Jednym z wyborów jest $n = 2^{10}$ i $k = 2^{24}$. Wtedy $\lambda = 16$.

²⁰ A więc jeśli $K = y$, to p -wartość wynosi

$$\mathbb{P}(K \geq y | \mathcal{H}_0) \approx 1 - \sum_{j=0}^{y-1} \frac{\lambda^j}{j!} e^{-\lambda}.$$

Okazuje się, że wtedy błąd przybliżenia jest rzędu $O(\lambda^2/n)$. W pracy L'Ecuyera i Simarda [30] zalecane jest aby $n^3 \leq k^{5/4}$.

¹⁹birthday spacing test

²⁰Trzeba poszukać gdzie jest dowód; Knuth z roku 1997; Exercises 3.3.2-28-30

4.3 Testy dla ciągów bitów

Niech $e_1, e_2, \dots$ będzie ciągiem zero-jedynkowym, który przy założeniu hipotezy zerowej jest realizacją losowego ciągu bitów $\xi_1, \xi_2, \dots$. Oznaczmy zmienne losowe $\eta_j = 2\xi_j - 1$, które przyjmują wartości $-1, 1$ z prawdopodobieństwem $1/2$ oraz przez *symetryczne błędzenie przypadkowe* rozumiemy ciąg

$$S_0, S_1 = \eta_1, \dots, S_n = \eta_1 + \dots + \eta_n.$$

W dalszej części oznaczamy $f_i = 2e_i - 1$. $s_0 = 0, s_1 = f_1, s_2 = f_1 + f_2, \dots$

Test częstotliwości Przy założeniu hipotezy zerowej, że ciągi pochodzą z obserwacji losowego ciągu bitów, z centralnego twierdzenia granicznego mamy, że zmienna losowa $S_n/\sqrt{n}$ ma w przybliżeniu rozkład normalny $\mathcal{N}(0, 1)$, natomiast $T = |S_n/\sqrt{n}|$ ma rozkład normalny połówkowy. Mianowicie zauważamy, że $\mathbb{E} \eta_j = 0$ oraz $\text{Var} \eta_j = 1$. Niech $F(t) = \mathbb{P}(T \leq t)$. Dla danego ciągu $e_1, \dots$ weryfikujemy hipotezę o jego losowości obliczając statystykę $T = t_0$ i następnie obliczamy p -wartość z wzoru $1 - F(t_0)$.

Test częstotliwości w blokach Niech $n = bm$. Testowany ciąg $\xi_1, \dots, \xi_n$, który przy założeniu hipotezy $\mathcal{H}_0$ jest losowym ciągiem bitów, długości n dzielimy na m bloków długości b . W każdym kolejnym bloku obliczamy frakcję jedynek:

$$\Pi_i = \frac{1}{b} \sum_{j=1}^b \xi_{(i-1)b+j}, \quad i = 1, \dots, m.$$

Mamy

$$\mathbb{E} \Pi_i = \frac{1}{2}, \quad \text{Var} \Pi_i = \frac{1}{4b}.$$

Dla odpowiednio dużych m , zmienna

$$\frac{1}{\sqrt{b}/2} \sum_{j=1}^b (\xi_{(i-1)b+j} - (b/2)),$$

ma w przybliżeniu standardowy rozkład normalny $\mathcal{N}(0, 1)$. Niech $Z_1, \dots, Z_m$ będzie ciągiem niezależnych zmiennych losowych o rozkładzie standardowym normalnym, i jeśli oznaczmy przez $\sim_d$ - rozkład w przybliżeniu, to

$$\Pi_i - (b/2) \sim_d \frac{1}{2\sqrt{b}} Z_i, \quad i = 1, \dots, m.$$

A więc

$$\sum_{i=1}^m Z_i^2 \sim_d 4b \sum_{i=1}^m (\Pi_i - 1/2)^2,$$

i stąd mamy, że przy założeniu hipotezy $\mathcal{H}_0$ statystyka

$$T = 4b \sum_{i=1}^m (\Pi_i - 1/2)^2$$

ma rozkład chi-kwadrat z m stopniami swobody. Jak zwykle teraz obliczamy p -wartość.

Test absolutnego maksimum błędzenia przypadkowego Dla prostego symetrycznego błędzenia przypadkowego S_n następujące twierdzenia Erdösa – Kaca jest prawdziwe:

$$\lim_{n \rightarrow \infty} \mathbb{P}(\max_{1 \leq j \leq n} S_n / \sqrt{n} \leq x) = \frac{4}{\pi} \sum_{j=0}^{\infty} \frac{(-1)^j}{2j+1} \exp(-(2j+1)^2 \pi^2 / 8x^2), \quad x \geq 0$$

21

Test oparty na wycieczkach błędzenia przypadkowego. Dla ciągu $s_0 = 0, s_1, s_2, \dots, s_n$ niech $j_1 < j_2 < j_3 < \dots$ będą indeksy kolejnych powrotów (s_n) do zera; tj. $s_i \neq 0$ dla $j_l < i < j_{l+1}$ dla $l = 1, 2, \dots$. Przez wycieczkę rozumie się $(s_0, s_1, \dots, s_{j_1}), (s_{j_1}, \dots, s_{j_2}), \dots$. Niech dla prostego symetrycznego błędzenia przypadkowego $(S_j)_{1 \leq j \leq n}$ niech J będzie liczbą wycieczek. Test budujemy korzystając z twierdzenia granicznego

$$\lim_{n \rightarrow \infty} \mathbb{P}\left(\frac{J}{\sqrt{n}} \leq y\right) = \sqrt{\frac{2}{\pi}} \int_0^y e^{-x^2/2} dx. \quad (4.4)$$

Rozważmy teraz liczbę wizyt $L(x)$ do stanu x podczas jednej wycieczki. Wtedy

$$\mathbb{P}(L(x) = k) = \begin{cases} 1 - \frac{1}{2|x|} & \text{dla } k = 0 \\ \frac{1}{x^2} \left(1 - \frac{1}{2|x|}\right)^{k-1} & \text{dla } k \geq 1 \end{cases} \quad (4.5)$$

²¹Bull. Amer. Math. Soc. Volume 52, Number 4 (1946), 292-302. On certain limit theorems of the theory of probability P. Erdős and M. Kac

Testujemy generator na bazie twierdzenia granicznego (4.4) oraz korzystając z (4.4) stosujemy test chi-kwadrat.

Do podrozdziału II podstawowym tekstem jest monografia Knutha [22]; wydanie polskie [23]. W literaturze polskiej są dostępne książki Zielińskiego [51, 48], czy artykuł przeglądowy Kotulskiego [26]. Więcej na temat konstrukcji generatorów można znaleźć w artykułach przeglądowym L'Ecuyera [27, 30]. Omówienie przypadków efektu 'nielosowości' dla generatora *state* w MATLABie można znaleźć w pracy Savicky'ego [42] Natomiast przykład 3.6 jest zaczerpnięty z

<http://www.mathworks.com/support/solutions/en/data/1-10HYAS/index.html?solution=1-10HYAS>. Testowanie dobrych własności generatora jest bardzo obszerną dziś dziedziną. Podstawowym źródłem informacji jest monografia Knutha [23], rozdział III, również prace L'Ecuyera [27, 29, 30]. Podrozdział 4.3 jest oparty na raporcie [41]. Podrozdział 2.2 jest oparty na pracy Diaconisa [11]. Artykuł w New York Times był autorstwa Giny Kolaty [24]. Temu tematowi, i w ogólności szybkości zbieżności do rozkładu stacjonarnego łańcucha Markowa poświęcona jest książka [31].

Metoda quasi MC jest opisana w książce Niederreitera [35] i Glasserman [?].

5 Zadania

22

Zadania teoretyczne

- 5.1 Niech $X_1, X_2, \dots$ będą niezależnymi zmiennymi losowymi o jednakowym rozkładzie $\mathcal{U}(\bar{M})$, gdzie M jest liczbą naturalną oraz $\bar{M} = \{0, 1, \dots, M-1\}$. Niech $M = 2^k$ oraz $b(x)$ dla $x \in \bar{M}$ będzie rozwinięciem dwójkowym przedstawionym w reprezentacji k cyfr zero lub jeden. Pokazać, że ciąg zmiennych losowych zero-jedynkowych zdefiniowany przez

$$\xi_1, \xi_2, \dots = b(X_1), b(X_2) \dots$$

jest ciągiem prób Bernoulliego. Pokazać na przykładzie, że tak nie musi być jeśli M nie jest potęgą dwójki.

- 5.2 Uzasadnić, że procedura generowania losowej permutacji z wykładu daje losową permutację po rozwiązaniu następującego zadania z rachunku prawdopodobieństwa, w którym indukcyjnie definiujemy ciąg n permutacji zbioru $[n]$. Zerową permutacją π_0 jest permutacja identycznościowa. Dla $k = 1, 2, \dots, n-1$ definiujemy indukcyjnie π_k -tą z π_{k-1} -szej następująco: zamieniamy element k -ty z J_k -tym gdzie J_k jest losowe (tj. o rozkładzie jednostajnym) na $\{k, \dots, n\}$, niezależnie od $J_1, \dots, J_{k-1}$. Pokazać, że π_{n-1} jest permutacją losową.

- 5.3 Dla przykładu 4.2, przy $n = 2^{14}$ i $k = 2^{20}$ uzupełnić tabelkę:

liczba kolizji $\leq$	100	110	120	130	140	150	160	170
z prawdopodobieństwem	?	?	?	?	?	?	?	?

- 5.4 Dany jest następujący ciąg zero-jedynkowy : 101011011111100001010100
 0101100010100010101110110
 100101010011010101011111
 0111000101011000100000001
 0011100111101001111001111
 0001110110001011100111000
 1011000001111001110001011

²²Poprawiane 13.10.2017

0100110110100101011010100
 1111000010011011001000001
 0001010001100100001100111
 1111011110011001001001110

Przeprowadzić testy sprawdzające zgodność z hipotezą, że pochodzi z ciągu symetrycznych prób Bernoulliego.

5.5 Przez *funkcję błędu* $\operatorname{erf}(x)$ rozumie się

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt,$$

oraz przez *uzupełniającą funkcję błędu* $\operatorname{erfc}(x)$

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt.$$

Pokazać, że jeśli $\Phi(x)$ jest dystrybuantą standardowego rozkładu normalnego, to

$$\Phi(x) = \frac{1}{2} + \frac{1}{2}\operatorname{erf}(x/\sqrt{2}) = \frac{1}{2}\operatorname{erfc}(-x/\sqrt{2}).$$

5.6 Udowodnić nierówność Czernowa. Niech $S_n = \xi_1 + \dots + \xi_n$, gdzie (ξ_i) są próbami Bernoulliego $\operatorname{Ber}(p)$. Pokazać, że dla $\delta < 1$

$$\begin{aligned} \mathbb{P}(S_n > (1 + \delta)np) &\leq \left(\frac{e^\delta}{(1 + \delta)^{1+\delta}} \right)^{np}, \\ \mathbb{P}(S_n < (1 - \delta)np) &\leq \left(\frac{e^{-\delta}}{(1 - \delta)^{1-\delta}} \right)^{np}. \end{aligned}$$

5.7 Obliczamy całkę $\bar{f} = \int_{[0,1]^d} f(\mathbf{u}) d\mathbf{u}$ przy założeniu, że $\sigma_f^2 = \operatorname{Var} f(\mathbf{U}) < \infty$. Korzystając z nierówności Czebyszewa pokazać, że z prawdopodobieństwem co najmniej δ mamy

$$\left| \frac{1}{n} \sum_{i=1}^n f(\mathbf{U}_i) - \bar{f} \right| < \frac{\sigma_f}{\sqrt{\delta n}}.$$

- 5.8 Rzucono parę kostek $n = 144$ razy i otrzymano następujące wyniki. W tabelce przez E_s oznaczamy liczbę oczekiwanych obserwacji a przez O_s liczbę zaobserwowanych.

s	2	3	4	5	6	7	8	9	10	11	12
E_s	4	8	12	16	20	24	20	16	12	8	4
O_s	2	4	10	12	22	29	21	15	14	9	6

Zweryfikować testem chi-kwadrat hipotezę, że rzuty są niezależne i na każdej kostce wyniki są jednakowo prawdopodobne. Obliczyć p -funkcję.

- 5.9 Pokazać, że dla układów pokerowych mamy następujące prawdopodobieństwa: Losujemy 5 kart z 52 i liczymy liczbę wystąpień określonych sekwencji. Liczba wszystkich możliwych sekwencji wynosi $W = \binom{52}{5} = 2\,598\,960$. Prawdopodobieństwa wystąpień sekwencji kart w pokerze: -
 Jedna para: $1098240/W = 0.422569 = 1/2,4$
 - Dwie Pary: $123552/W = 0.047539 = 1/21$
 - Trójka: $54912/W = 0.0211 = 1/44$
 - Full: $3744/W = 0.00144 = 1/694$
 - Kareta: $624/W = 0.00024 = 1/4165$
 - Kolor: $5148/W = 0.00198 = 1/50$
 - Strit: $9 * 4^5/W = 9216/W = 0.003546 = 1/282$
 - Poker: $9 * 4/W = 36/W = 0.0000138 = 1/72193$.

- 5.10 [Feller, t. 1, p.94] Klasyczne zadanie o rozmieszczeniu n kul w m komórkach zakłada ich "losowość", tzn. każde rozmieszczenie ma prawdopodobieństwo m^{-n} . Pokazać, że prawdopodobieństwo $p_c(n, m)$ znalezienia dokładnie c pustych wynosi

$$\binom{m}{c} \sum_{j=0}^{m-c} (-1)^j \binom{m-c}{j} \left(1 - \frac{n+j}{m}\right).$$

Pokazać, że gdy m, n rosną tak, że $\lambda = me^{-\frac{n}{m}}$ pozostaje ograniczone, to dla każdego c

$$p_c(n, m) \rightarrow \frac{\lambda^c}{c!} e^{-\lambda}.$$

Zadania laboratoryjne

Zadania na wykorzystanie instrukcji MATLAB-owskich: rand, rand(m,n), rand('state', 0), randperm, randn.

- 5.11 Przeprowadzić test szybkości generatora: `rand`, `rand('state')`, `rand('seed')`, `rand('twister')` przy użyciu następującego skryptu:

```
clc; clear;
tic
for j=1:1000000
    rand;
end
toc
```

- 5.12 Napisać program na sprawdzenie generatorów MATLABowskich przy użyciu
- testu Kołmogorowa–Smirnowa,
 - testu serii z $t = 3$. Obliczyć p -wartość dla $d = 4$, $n = 2^m$; ($m = 10, 11, \dots, 20$).

- 5.13 [Savicky] Uruchomić procedurę

- 5.14 Będziemy rozważali tablicę $U_{i=1,\dots,28;j=1,\dots,100000}$ liczb losowych z której wytniemy wszystkie kolumny poza takimi dla których w pierwszym wierszu wynik jest mniejszy od $1/16$. W przypadku, gdy ta tablica składa się z niezależnych zmiennych losowych o jednakowym rozkładzie otrzymamy tablicę o losowej liczbie kolumn, składającej się z niezależnych zmiennych losowych o rozkładzie $U(0, 1]$. Teraz przeprowadzimy eksperyment z generatorami MATLABowskim 'state' i 'twister':

```
Z=rand(28,100000);
condition = Z(1,:) < 1/16;
scatter(Z(2,condition), Z(28,condition), 'r');
```

i porównać z

```
Z=rand(28,100000);
condition = Z(1,:) < 1/16;
scatter(Z(16,condition), Z(28,condition), 'r');
```

W przypadku generatora 'state' otrzymujemy ewidentnie różne obrazki. A co się dzieje gdyby użyć metody 'twister'. Zastanowić się jak powinna wyglądać chmura w wypadku 'idealnym'.

- 5.15 Napisać procedurę na losowanie
1. z n obiektów wybieramy losowo k ze zwracaniem,
 2. z n obiektów wybieramy k bez zwracania.
 3. wybierającą losowy podzbiór k elementowy $\{1, \dots, n\}$.
- 5.16 Napisać procedurę generowania losowej permutacji `randpermmy` na podstawie algorytmu z podrozdziału 2.1. Porównać szybkość swojej procedury z matlabowską `randperm`.
- 5.17 Obliczyć przez symulacje prawdopodobieństwo p_n tego, że w permutacji losowej liczb $1, \dots, n$, żadna liczba nie jest na swoim miejscu. Zrobić obliczenia dla $n = 1, \dots, 10$. Do czego może zdążyć p_n gdy $n \rightarrow \infty$.
- 5.18 Napisać procedurę n rzutów monetą $(\xi_1, \dots, \xi_n)$.
Jeśli orzeł w i -tym rzucie to niech $\xi_i = 1$ jeśli reszka to $\xi_i = -1$. Niech $S_0 = 0$ oraz $S_k = \xi_1 + \dots + \xi_k$. Obliczyć pierwszy moment i wariancję ξ_1 . Zrobić wykres $S_0, S_1, \dots, S_{1000}$. Na wykresie zaznaczyć linie: $\pm \text{Var}(\xi)\sqrt{n}$, $\pm 2\text{Var}(\xi)\sqrt{n}$, $\pm 3\text{Var}(\xi)\sqrt{n}$.
Zrobić zbiorczy wykres 10 replikacji tego eksperymentu.
- 5.19 Rzucamy N razy symetryczną monetą i generujemy błędzenie przypadkowe $(S_n)_{0 \leq n \leq N}$. Napisać procedury do obliczenia:
1. Jak wygląda absolutna różnica pomiędzy maximum i minimum błędzenia przypadkowego gdy N rośnie?
- 5.20 Rozważamy ciąg $(S_n)_{0 \leq n \leq 2N}$. Przez prowadzenie w odcinku $(i, i+1)$ rozumiemy, że $S_i \geq 0$ oraz $S_{i+1} \geq 0$. Niech X będzie łączną długością prowadzeń dla N rzutów.
1. Jakie jest prawdopodobieństwo, że jeden gracz będzie przeważał pomiędzy 50% a 55% czasu? Lub więcej niż 95% czasu? Zrobić obliczenia symulacją. Eksperymentalnie sprawdzić ile powtórzeń należy zrobić aby osiągnąć stabilny wynik.
 3. Niech $N = 200$. Zrobić histogram wyników dla 10000 powtórzeń tego eksperymentu.
- 5.21 Napisać procedurę do zadania dni urodzin.
1. Jakie jest prawdopodobieństwo, że wśród n losowo wybranych osób przynajmniej dwie mają ten sam dzień urodzin?
 2. Jakie jest prawdopodobieństwo, że wśród n losowo wybranych osób

przynajmniej dwie mają urodziny w przeciągu r dni jeden od drugiego?

3. Przypuśćmy, że osoby pojawiają się kolejno. Jak długo trzeba czekać aby pojawiły się dwie mające wspólne urodziny? Zrobić histogram dla 1000 powtórzeń.

5.22 [Test odstępów dni urodzin] Używając symulacji znaleźć funkcję prawdopodobieństwa statystyki R . Sprawdzić czy wyniki się pokrywają jeśli do obliczeń używa się różnych generatorów.

5.23 [Zagadnienie komiwojażera] Komiwojażer opuszcza miasto 0 i musi odwiedzić wszystkie miasta $1, \dots, n$ przed powrotem do punktu wyjściowego. Odległości pomiędzy miastem i a miastem j jest c_{ij} . Jeśli nie ma drogi do kładziemy $c_{ij} = \infty$. Jest $n = 100$ miast do odwiedzenia, których odległości są zadane w następujący sposób. Zacząć od $\text{rand}(\text{'state'}, 0)$. Przydzielić odległości kolejno wierszami c_{ij} gdzie $i = 0, \dots, 100$ oraz $j = 0, 1, \dots, 100$ - odległość między adresem i oraz j . (W ten sposób wszyscy będą mieli taką samą macierz odległości). Znaleźć jak najlepszą drogę $0, \pi_1, \dots, \pi_{100}, 0$ minimizującą przejechaną trasę. Czy uda się odnaleźć najkrótszą trasę. Znaleźć najkrótszą po 100, 200, itd. losowaniach.

Projekt

Projekt 1 Bob szacuje, że w życiu spotka 3 partnerki, i z jedną się ożeni. Zakłada, że jest w stanie je porównywać. Jednakże, będąc osobą honorową, (i) jeśli zdecyduje się umawiać z nową partnerką, nie może wrócić do już odrzuconej, (ii) w momencie decyzji o ślubie randki z pozostałymi są niemożliwe, (iii) musi się zdecydować na którąś z spotkanych partnerek. Bob przyjmuje, że można zrobić między partnerkami ranking: 1=dobra, 2=lepsz, 3=najlepsza, ale ten ranking nie jest mu wiadomy. Bob spotyka kolejne partnerki w losowym porządku.

Napisz raport jaką strategię powinien przyjąć Bob. W szczególności w raporcie powinny się znaleźć odpowiedzi na następujące pytania. Przy sporządzaniu raport przeprowadzić analizę błędu, korzystając z fundamentalnego wzoru

$$b = \frac{1.96\sigma}{\sqrt{n}}$$

(patrz podsekcja IV.1.3, gdzie b - błąd, n - liczba replikacji, oraz σ wariancja estymatora). A więc jeśli chcemy wyestymować r z błędem mniejszym niż $b = 0.02$ możemy skorzystać z fundamentalnego wzoru z wariancją uzyskaną wcześniej z pilotażowej symulacji 1000 replacji.

- (a) Bob decyduje się na ślub z pierwszą spotkaną partnerką. Oblicz prawdopodobieństwo P (teoretyczne), że ożeni się z najlepszą. Oblicz też oczekiwany rangę r
- (b) Użyj symulacji aby określić P i r dla następującej strategii. Bob nigdy nie żeni się z pierwszą, żeni się z drugą jeśli jest lepsza od pierwszej, w przeciwnym razie żeni się z trzecią.
- (c) Rozważyć zadanie z 10-cioma partnerkami, które mają rangi $1, 2, \dots, 10$. Obliczyć teoretycznie prawdopodobieństwo P i oczekiwaną rangę r dla strategii jak w zadaniu (a).
- (d) Zdefiniować zbiór strategii, które są adaptacją strategii z zadania (b) na 10 partnerek. Podać jeszcze inną strategię dla Boba. Dla każdej strategii obliczyć P i r . Wsk. Należy pamiętać, że jak się nie zdecydujemy na jakąś kandydatkę, to nie ma potem do niej powrotu.

Projekt 2 Rozpatrzeć następujące generatory liczb losowych: “state”, “twister” oraz excellowski $u_i = (0.9821u_{i-1} + 0.211327) \bmod 1$. Ponadto

rozpatrzeć pseudo-losowe ciągi bitów `date.e`, `date.sqrt2` (z strony www.math.uni.wroc.pl/~olski).

(i) Dla tych ciągów przeprowadzić analizę ich “losowości”, korzystając z przynajmniej 3 testów (odpowiednio dla każdego ciągu). Na przykład Kołmogorowa-Smirnow, test serii i test odstępów dni urodzin. Zamiast tego ostentego można skorzystać z testu kolizji.

(ii) Dla generatora “twister” zbadać dwa przypadki. Pierwszy gdy generator startuje z ziarna $u_0 = 0$, a drugi gdy z ziarna $u_0 = 1812433253$.

Rozdział III

Symulacja zmiennych losowych

1

1 Metoda ITM; rozkład wykładniczy i Pareto

Naszym celem jest przeprowadzenie losowania w wyniku którego otrzymamy liczbę losową Z z dystrybucją F mając do dyspozycji zmienną losową U o rozkładzie jednostajnym $U(0,1)$. W dalszym ciągu będziemy często bez wyjaśnienia pisali $U_1, U_2, \dots$ dla ciągu niezależnych zmiennych losowych o jednakowym rozkładzie $\mathcal{U}(0,1)$.

Niech

$$F^{\leftarrow}(t) = \inf\{x : t \leq F(x)\}.$$

W przypadku gdy $F(x)$ jest funkcją ściśle rosnącą, $F^{\leftarrow}(t)$ jest funkcją odwrotną $F^{-1}(t)$ do $F(x)$. Dlatego $F^{\leftarrow}(t)$ będziemy nazywać się *uogólnioną funkcją odwrotną*. Zbadamy teraz niektóre własności uogólnionej funkcji odwrotnej $F^{\leftarrow}(x)$. Jeśli F jest ściśle rosnąca to jest oczywiście prawdziwa następująca równoważność:

- $t = F(x)$ wtedy i tylko wtedy gdy $F^{\leftarrow}(t) = x$.

Jednakże można podać przykład taki, że $F^{\leftarrow}(t) = x$, ale $t < F(x)$ (patrz rysunek [??2rys1??]). Natomiast jest prawdą, że zawsze $F^{\leftarrow}(F(x)) = x$, chociaż łatwo podać przykład, że nie musi być $F(F^{\leftarrow}(t)) = t$ (zostawiamy czytelnikowi znalezienie takiego przykładu w formie rysunku takiego jak w [??2rys1??]).

¹Poprawiane 17.1.2018

W skrypcie będziemy oznaczać *ogon dystrybuanty* $F(x)$ przez

$$g(x) = \bar{F}(x) = 1 - F(x).$$

Przydatne będą następujące własności $F^{\leftarrow}(u)$.

Lemat 1.1 (a) $u \leq F(x)$ wtedy i tylko wtedy gdy $F^{\leftarrow}(u) \leq x$.

(b) Jeśli $F(x)$ jest funkcją ciągłą, to $F(F^{\leftarrow}(x)) = x$.

(c) $\bar{F}^{\leftarrow}(x) = F^{\leftarrow}(1 - x)$.

Fakt 1.2 (a) Jeśli U jest zmienną losową, to $F^{\leftarrow}(U)$ jest zmienną losową z dystrybucją F .

(b) Jeśli F jest ciągłą i zmienna $X \sim F$, to $F(X)$ ma rozkład jednostajny $U(0,1)$

Dowód (a) Korzystając z lematu 1.1 piszemy

$$\mathbb{P}(F^{\leftarrow}(U) \leq x) = \mathbb{P}(U \leq F(x)) = F(x) .$$

(b) Mamy

$$\mathbb{P}(F(X) \geq u) = \mathbb{P}(X \geq F^{\leftarrow}(u)) .$$

Teraz korzystając z tego, że zmienna losowa X ma dystrybuantę ciągłą, a więc jest bezatomowa, piszemy

$$\mathbb{P}(X \geq F^{\leftarrow}(u)) = \mathbb{P}(X > F^{\leftarrow}(u)) = 1 - F(F^{\leftarrow}(u)) ,$$

które na mocy lematu 1.1 jest równe $1 - u$. □

Oznaczmy przez

$$\begin{aligned} g(t) &= F^{\leftarrow}(t) \\ \bar{g}(t) &= g(1 - t) \end{aligned}$$

Zauważmy, że

$$\bar{g}(t) = \inf\{u : \bar{F}(u) \leq 1 - t\} .$$

Korzystając z powyższego faktu mamy następujący algorytm na generowanie zmiennej losowej o zadanym rozkładzie F jeśli mamy daną funkcję $g(t)$.

Algorytm ITM

$Z = g(\text{rand}) ;$

Zauważmy jeszcze coś co będzie przydatne w dalszej części, że jeśli $U_1, U_2, \dots$ jest ciągiem zmiennych losowych o jednakowym rozkładzie jednostajnym, to $U'_1, U'_2, \dots$, gdzie $U'_j = 1 - U_j$, jest też ciągiem niezależnych zmiennych losowych o rozkładzie jednostajnym.

Niestety jawne wzory na funkcję $g(x)$ czy $\bar{g}(x)$ znamy w niewielu przypadkach, które omówimy poniżej.

Rozkład wykładniczy $\text{Exp}(\lambda)$. W tym przypadku

$$F(x) = \begin{cases} 0, & x < 0 \\ 1 - \exp(-\lambda x), & x \geq 0. \end{cases} \quad (1.1)$$

Jeśli $X \sim \text{Exp}(\lambda)$, to $\mathbb{E} X = 1/\lambda$ oraz $\text{Var } X = 1/(\lambda^2)$. Jest to zmienna z lekkim ogonem ponieważ

$$\mathbb{E} e^{sX} = \frac{\lambda}{\lambda - s}$$

jest określona dla $0 \leq s < \lambda$. W tym przypadku funkcja odwrotna

$$\bar{g}(x) = -\frac{1}{\lambda} \log(x) .$$

Zmienną losową $\text{Exp}(\lambda)$ o rozkładzie wykładniczym generujemy następującym algorytmem.

Algorytm ITM-Exp.

% Dane lambda. Otrzymujemy Z o rozkładzie wykładniczym z par. lambda
 $Z = -\log(\text{rand})/\lambda ;$

Zajmiemy się teraz przykładem zmiennej ciężko-ogonowej.

Rozkład Pareto $\text{Par}(\alpha)$. Rozkład Pareto definiujemy przez

$$F(x) = \begin{cases} 0, & x < 0 \\ 1 - \frac{1}{(1+x)^\alpha}, & x \geq 0. \end{cases} \quad (1.2)$$

Jeśli $X \sim \text{Par}(\alpha)$, to $\mathbb{E}X = 1/(\alpha - 1)$ pod warunkiem, że $\alpha > 1$ oraz $\mathbb{E}X^2 = 2/((\alpha - 1)(\alpha - 2))$ pod warunkiem, że $\alpha > 2$. Jest to zmienna z ciężkim ogonem ponieważ

$$\mathbb{E}e^{sX} = \infty$$

dla dowolnego $0 < s$. Funkcja odwrotna

$$\bar{g}(x) = x^{-1/\alpha} - 1.$$

Zauważmy, że rozkład Pareto jest tylko z jednym parametrem kształtu α . Dlatego jeśli chcemy mieć szerszą klasę dopuszczającą dla tego samego parametru kształtu zadaną średnią m skorzystamy z tego, że zmienna $m(\alpha - 1)X$ ma średnią m (oczywiście zakładając, że $\alpha > 1$). Można więc wprowadzić dwuparametrową rodzinę rozkładów Pareto $\text{Par}(\alpha, c)$. Zmienna Y ma rozkład $\text{Par}(\alpha, c)$ jeśli jej dystrybucja jest

$$F(x) = \begin{cases} 0, & x < 0 \\ 1 - \frac{c^\alpha}{(c+x)^\alpha}, & x \geq 0. \end{cases} \quad (1.3)$$

W literaturze nie ma jednoznacznej definicji rozkładu Pareto, i dlatego w poręcznikach można znaleźć inne warianty podanej definicji.

Algorytm ITM-Par.

```
% Dane alpha. Otrzymujemy Z o rozkładzie Par(alpha).
Z=rand^((-1/\alpha))-1.
```

2 Metoda ITM oraz ITR dla rozkładów krato- wych; rozkład geometryczny

Przypuśćmy, że rozkład jest kratowy skoncentrowany na $\{0, 1, \dots\}$. Wtedy do zadania rozkładu wystarczy znać funkcję prawdopodobieństwa $\{p_k, k = 0, 1, \dots\}$.

2. METODA ITM ORAZ ITR DLA ROZKŁADÓW KRATOWYCH; ROZKŁAD GEOMET

Fakt 2.1 Niech U będzie liczbą losową. Zmienna

$$Y = \min\{k : U \leq \sum_{i=0}^k p_i\}$$

ma funkcję prawdopodobieństwa $\{p_k\}$.

Dowód Zauważmy, że

$$\mathbb{P}(Y = l) = \mathbb{P}\left(\sum_{i=0}^{l-1} p_i < U \leq \sum_{i=0}^l p_i\right),$$

i odcinek $(\sum_{i=0}^{l-1} p_i, \sum_{i=0}^l p_i]$ ma długość p_l . □

Stąd mamy następujący algorytm na generowanie kratowych zmiennych losowych Y .

Algorytm ITM-d. z

```
% Dane p_k=p(k). Otrzymujemy Y z funkcją prawd. {p_k}.
U=rand;
Y=0;
S=p(0);
while S<U
    Y=Y+1;
    S=S+p(Y);
end
```

Fakt 2.2 Niech N będzie liczbą wywołań w pętli while w powyższym algorytmie. Wtedy N ma funkcję prawdopodobieństwa

$$\mathbb{P}(N = k) = p_{k-1},$$

dla $k = 1, 2, \dots$, skąd średnia liczba wywołań wynosi $\mathbb{E} N = \mathbb{E} Z + 1$.

W powyższego faktu widzimy więc, że jeśli zmienna Z ma średnią nieskończoną, to liczba wywołań ma również średnią nieskończoną i algorytm nie jest dobry.

Rozkład geometryczny $\text{Geo}(p)$ ma funkcję prawdopodobieństwa

$$p_k = (1 - p)p^k, \quad k = 0, 1, \dots$$

Zmienna $X \sim \text{Geo}(p)$ ma średnią $\mathbb{E} X = p/q$. Zmienna

$$X = \lfloor \log U / \log p \rfloor$$

ma rozkład $\text{Geo}(p)$, ponieważ

$$\begin{aligned} \mathbb{P}(X \geq k) &= \mathbb{P}(\lfloor \log U / \log p \rfloor \geq k) \\ &= \mathbb{P}(\log U / \log p \geq k) \\ &= \mathbb{P}(\log U) \leq k \log p \\ &= \mathbb{P}(U \leq p^k) = p^k. \end{aligned}$$

W niektórych przypadkach następująca funkcja

$$c_{k+1} = \frac{p_{k+1}}{p_k}$$

jest prostym wyrażeniem. Podamy teraz kilka takich przykładów.

- $X \sim \text{B}(n, p)$. Wtedy

$$p_0 = (1 - p)^n, \quad c_{k+1} = \frac{p(n - k)}{(1 - p)(k + 1)}, \quad k = 0, \dots, n - 1.$$

- $X \sim \text{Poi}(\lambda)$. Wtedy

$$p_0 = e^{-\lambda}, \quad c_{k+1} = \frac{\lambda}{k + 1}, \quad k = 0, 1, \dots$$

- $X \sim \text{NB}(r, p)$. Wtedy

$$p_0 = (1 - p)^r, \quad c_{k+1} = \frac{(r + k)p}{k + 1}, \quad k = 0, 1, \dots$$

W przypadku gdy znana jest jawna i zadana w prostej postaci funkcja c_{k+1} zamiast algorytmu ITM-d możemy używać następującego ITR. Oznaczamy $c_{k+1} = c(k+1)$ i $p_0 = p(0)$.

Algorytm ITR.

```
% Dane p(0) oraz c(k+1), k=0,1,... Otrzymujemy X z f. prawd. {p_k}.
S=p(0);
P=p(0);
X=0;
U=rand;
while (U>S)
    X=X+1;
    P=P*c(X);
    S=S+P;
end
```

3 Metody ad hoc.

Są to metody specjalne, które wykorzystują własności konkretnych rozkładów.

3.1 $B(n, p)$

Niech $X \sim B(n, p)$. Wtedy

$$X =_d \sum_{j=1}^n \xi_j,$$

gdzie $\xi_1, \dots, \xi_n$ jest ciągiem prób Bernoulliego, tj. ciąg niezależnych zmiennych losowych $\mathbb{P}(\xi_j = 1) = p$, $\mathbb{P}(\xi_j = 0) = 1 - p$.

Algorytm DB

```
% Dane n,p. Otrzymujemy X z f. prawd. B(n,p)
X=0;
for i=1:n
    if (rand<= p)
        X=X+1;
    end
end
```

Polecamy czytelnikowi porównać tą metodę z metodą wykorzystującą ITR.

3.2 Erl(n, λ)

Jeśli $\xi_1, \dots, \xi_n$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $\text{Exp}(\lambda)$, to

$$X = \xi_1 + \dots + \xi_n$$

ma rozkład Erlanga $\text{Erl}(n, \lambda)$. Wiadomo, że rozkład Erlanga jest szczególnym przypadkiem rozkładu $\text{Gamma}(n, \lambda)$ z gęstością

$$f(x) = \frac{1}{\Gamma(n)} \lambda^n x^{n-1} e^{-\lambda x}, \quad x \geq 0.$$

Przez proste różniczkowanie można się przekonać, że ogon dystrybucyjności rozkładu Erlanga $\text{Erl}(k+1, 1)$ wynosi

$$e^{-x} \left(1 + x + \frac{x^2}{2!} + \dots + \frac{x^k}{k!} \right).$$

Będziemy tego faktu potrzebować poniżej.

Ponieważ liczbę losową o rozkładzie wykładniczym $\text{Exp}(\lambda)$ generujemy przez $-\log(\text{rand})/\lambda$, więc $X \sim \text{Erl}(n, \lambda)$ możemy generować przez

$$-\log(U_1 \cdot \dots \cdot U_n)/\lambda.$$

Należy zauważyć, że w przypadku dużych n istnieje ryzyko wystąpienia błędu zmienneopozycyjnego niedomiaru przy obliczaniu iloczynu $U_1 \cdot \dots \cdot U_n$.

3.3 Poi(λ).

Teraz podamy algorytm na generowanie zmiennej $X \sim \text{Poi}(\lambda)$ o rozkładzie Poissona. Niech $\tau_1, \tau_2, \dots$ będą niezależnymi zmiennymi losowymi o jednakowym rozkładzie $\text{Exp}(1)$. Niech

$$N = \#\{i = 1, 2, \dots : \tau_1 + \dots + \tau_i \leq \lambda\}. \quad (3.4)$$

Fakt 3.1 *Mamy*

$$N =_d \text{Poi}(\lambda).$$

Dowód Zauważmy, że

$$\{N \leq k\} = \left\{ \sum_{j=1}^{k+1} \tau_j > \lambda \right\}.$$

Ponieważ

$$\mathbb{P}\left(\sum_{j=1}^k \tau_j > x\right) = e^{-x}\left(1 + x + \frac{x^2}{2!} + \dots + \frac{x^k}{k!}\right),$$

i pamiętając o konwencji $\sum_{j=1}^0 = 0$, mamy więc

$$\mathbb{P}(N = 0) = \mathbb{P}(N \leq 0) = e^{-x},$$

oraz

$$\begin{aligned} \mathbb{P}(N = k) &= \mathbb{P}(\{N \leq k\} \cap \{N \leq k-1\}^c) \\ &= \mathbb{P}(\{N \leq k\}) - \mathbb{P}(\{N \leq k-1\}) \\ &= \frac{\lambda^k}{k!} e^{-\lambda}. \end{aligned}$$

□

Wzór (3.4) można przepisać następująco:

$$N = \begin{cases} 0, & \tau_1 > \lambda \\ \max\{i \geq 1 : \tau_1 + \dots + \tau_i \leq \lambda\}, & \text{w przeciwnym razie.} \end{cases}$$

Na tej podstawie możemy napisać następujący algorytm.

Algorytm DP

```
% Dane lambda. Otrzymujemy X o rozkładzie Poi(lambda).
X=0;
S=-log(rand);
while (S<=lambda)
    Y=-log(rand);
    S=S+Y;
    X=X+1;
end
```

Algorytm DP można jeszcze uprościć korzystając z następujących rachunków. Jeśli $\tau_1 \leq \lambda$ co jest równoważne $-\log U_1 \leq \lambda$ co jest równoważne

$U_1 \geq \exp(-\lambda)$, mamy

$$\begin{aligned} X &= \max\{n \geq 0 : \tau_1 + \dots + \tau_n \leq \lambda\} \\ &= \max\{n \geq 0 : (-\log U_1) + \dots + (-\log U_n) \leq \lambda\} \\ &= \max\{n \geq 0 : \log \prod_{i=1}^n U_i \geq -\lambda\} \\ &= \max\{n \geq 0 : \prod_{i=1}^n U_i \geq e^{-\lambda}\}. \end{aligned}$$

Powyżej przyjmujemy konwencję, że jeśli $n = 0$, to $\tau_1 + \dots + \tau_n = 0$.

3.4 Rozkład chi-kwadrat

Jeśli ξ ma rozkład normalny $N(0,1)$ to prosty argument pokazuje, że ξ^2 ma rozkład $\text{Gamma}(1/2, 1/2)$. Jeśli więc zmienne $\xi_1, \dots, \xi_n$ są niezależne o jednakowym rozkładzie $N(0,1)$ to

$$\xi_1^2 + \dots + \xi_n^2 \tag{3.5}$$

ma rozkład $\text{Gamma}(n/2, 1/2)$. Taki rozkład nosi nazwę *chi kwadrat* z n stopniami swobody, który jest jednym z fundamentalnych rozkładów statystyki matematycznej. Jak generować zmienne losowe o rozkładzie $N(0,1)$ będziemy się uczyć dalej. W MATLABie generuje się takie liczby losowe za pomocą instrukcji

`randn`

4 Rozkład jednostajny

Będziemy rozważali wektor losowy X przyjmujący wartości w obszarze $A \subset E$, gdzie A jest podzbiorem $\mathbb{R}^d$. Przez $\text{vol}(D)$ oznaczamy miarę podzbioru D . Na przykład w przypadku $d = 1$ jest to długość, itd. Zakładamy, że $\text{vol}(A) < \infty$. Mówimy, że wektor losowy X ma rozkład jednostajny $U(A)$ w obszarze $A \subset \mathbb{R}^n$, jeśli

$$\mathbb{P}(X \in D) = \frac{\text{vol}(D)}{\text{vol}(A)}.$$

Naszą podstawową zmienną jest U o rozkładzie jednostajnym $U(0,1)$ na odcinku $(0,1)$. Jeśli więc potrzebujemy zmiennej $V \sim U(a,b)$ jednostajnej na odcinku (a,b) to musimy przekształcić zmienną U liniowo:

$$V = (b - a)U + a .$$

Zostawiamy uzasadnienie tego faktu czytelnikowi. Aby generować rozkład jednostajny w dowolnym obszarze A , np. kuli lub elipsoidy, zrobimy następującą obserwację. Przypuśćmy, że A i B będą podzbiorami $\mathbb{R}^n$ dla których istnieje objętość $\text{vol}(B) = \int_B d\mathbf{x} = |B|$ i $\text{vol}(A) = \int_A d\mathbf{x}$ oraz niech $A \subset B$. Kluczem do napisania algorytmu na generowanie $\mathbf{Y} \sim U(A)$ jest następująca obserwacja. Jeśli $\mathbf{X}$ ma rozkład jednostajny $U(B)$, to $Y_d = (X|X \in A)$ ma rozkład jednostajny $U(A)$. Mianowicie trzeba skorzystać, że dla $D \subset A$

$$\mathbb{P}(Y \in D) = \mathbb{P}(X \in D|X \in A) = \frac{\text{vol}(D)/\text{vol}(B)}{\text{vol}(A)/\text{vol}(B)} = \frac{\text{vol}(D)}{\text{vol}(A)} .$$

Stąd mamy następujący algorytm na generowanie wektora losowego $\mathbf{X} = (X_1, \dots, X_n)$ o rozkładzie jednostajnym $U(A)$, gdzie A jest ograniczonym podzbiorem $\mathbb{R}^n$. Mianowicie taki zbiór A jest zawarty w pewnym prostokącie $B = [a_1, b_1] \times \dots \times [a_n, b_n]$. Łatwo jest teraz wygenerować wektor losowy $(V_1, \dots, V_n)$ o rozkładzie jednostajnym $U(B)$ ponieważ wtedy zmienne V_i są niezależne o rozkładzie odpowiednio $U[a_i, b_i]$ ($i = 1, \dots, n$).

Algorytm $U(A)$

%na wyjściu X o rozkładzie jednostajnym w A .

1. $U = \text{rand}(1,n)$;
2. for $i=1:n$
 - $V(i) = (b(i) - a(i)) * U(i) + a(i)$
- end
3. jeśli $V = (V(1), \dots, V(n))$ należy do A to $X := V$,
4. w przeciwnym razie idź do 1.

5 Metoda superpozycji

Jest to metoda bazująca na wyrażeniu rozkładu poszukiwanej liczby losowej X jako mieszanki rozkładów. W przypadku gdy X ma gęstość $f(x)$ zakła-

damy, że

$$f(x) = \sum_{j=1}^n p_j g_j(x),$$

gdzie $g_j(x)$ są pewnymi gęstościami rozkładów (n może być ∞). Przypuśćmy dalej, że potrafimy symulować zmienne losowe o tych gęstościach. Aby wygenerować liczbę losową z gęstością $f(x)$, losujemy najpierw J z funkcją prawdopodobieństwa (p_j) i jeśli $J = j$ to losujemy liczbę losową z gęstością $g_j(x)$ i tę wielkość przyjmujemy jako szukaną wartość X . Metoda ta ma w symulacji zastosowanie szersze niż by się wydawało. Bardzo ważnym przypadkiem jest gdy

$$f(x) = \sum_{j=0}^{\infty} p_j(\lambda) g_j(x) \quad (5.6)$$

gdzie $p_n(\gamma) = \frac{\lambda^n}{n!} e^{-\lambda}$ jest rozkładem Poissona a $g_n(x)$ jest gęstością gamma $\Gamma(\alpha_n, \beta)$.

Przykład 5.1 [Niecentralny rozkład chi-kwadrat] W statystyce, matematyce finansowej i innych teoriach pojawia się rozkład chi-kwadrat. Najbardziej ogólna wersja *niecentralnego rozkładu chi-kwadrat z d stopniami swobody i parametrem niecentralności λ* (tutaj zakładamy jedynie $d > 0$ i $\lambda > 0$) ma gęstość postaci (5.6) z

$$p_n = \frac{(\frac{\lambda}{2})^n}{n!} e^{-\lambda/2}, \quad g_n \sim \Gamma(j + \frac{1}{2}, \frac{1}{2}).$$

W przypadku gdy d jest naturalne, jest to gęstość rozkładu $X = \sum_{j=1}^d (Z_j + a_j)^2$, gdzie $\lambda = \sum_{j=1}^d a_j^2$ oraz $Z_1, \dots, Z_d$ są niezależne o jednakowym rozkładzie $\mathcal{N}(0, 1)$. Można pokazać, że

$$X =_d Y + W, \quad (5.7)$$

gdzie Y, W są niezależne, Y ma rozkład chi-kwadrat z $d - 1$ stopniami swobody oraz $W =_d (Z + \sqrt{\lambda})^2$.

Przykład 5.2 [Dagpunar [9]] W fizyce atomowej rozpatruje się rozkład z gęstością

$$f(x) = C \sinh((xc)^{1/2}) e^{-bx}, \quad x \geq 0$$

gdzie

$$C = \frac{2b^{3/2}e^{-c/(4b)}}{\sqrt{\pi c}}.$$

Po rozwinięciu w szereg $\sinh((xc)^{1/2})$ mamy

$$\begin{aligned} f(x) &= C \sum_{j=0}^{\infty} \frac{(xc)^{(2j+1)/2} e^{-bx}}{(2j+1)!} \\ &= C \sum_{j=0}^{\infty} \frac{\Gamma(j + \frac{2}{3}) c^{j+\frac{1}{2}} b^{j+3/2} x^{j+\frac{1}{2}} e^{-bx}}{b^{j+\frac{3}{2}} (2j+1)! \Gamma(j + \frac{2}{3})} \\ &= \sum_{j=0}^{\infty} C \frac{\Gamma(j + \frac{2}{3}) c^{j+\frac{1}{2}}}{b^{j+\frac{3}{2}} (2j+1)!} g_j(x), \end{aligned}$$

gdzie $g_j(x)$ jest gęstością rozkładu $\Gamma(j + \frac{3}{2}, b)$. Teraz korzystając z tego, że

$$\Gamma(j + \frac{3}{2}) = (j - 1 + \frac{3}{2})(j - 2 + \frac{3}{2}) \dots (1 + \frac{3}{2})(\frac{3}{2})\Gamma(\frac{3}{2})$$

(patrz Abramowitz & Stegun [2] wzór 6.1.16) oraz tego, że $\Gamma(\frac{3}{2}) = \pi^{\frac{1}{2}}/2$ (tamże, wzór 6.1.9) mamy

$$\begin{aligned} f(x) &= C \sum_{j=0}^{\infty} \frac{\pi^{1/2} c^{j+\frac{1}{2}}}{2^{2j+1} j! b^{j+\frac{3}{2}}} g_j(x) \\ &= \sum_{j=0}^{\infty} \left(\frac{c}{4b}\right)^j / j! e^{-c/(4b)} g_j(x), \end{aligned}$$

czyli jest to mieszanka gęstości $\Gamma(j + \frac{3}{2}, b)$, (p_j) będące rozkładem Poissona z parametrem $c/(4b)$. Teraz jeszcze warto zauważyć, że jeśli Y jest liczbą losową o rozkładzie $\text{Gamma}(j + \frac{3}{2}, \frac{1}{2})$ to $Y/(2b)$ liczbą losową o szukany rozkładzie $\text{Gamma}(j + \frac{3}{2}, b)$. Zauważmy na koniec, że $\text{Gamma}(j + \frac{3}{2}, \frac{1}{2})$ jest rozkładem chi kwadrat z $2j + 1$ stopniami swobody. Jak pokazaliśmy wcześniej taki rozkład prosto się symuluje.

6 Metoda eliminacji

Podamy teraz pewną ogólną metodę na generowanie zmiennych losowych o pewnych rozkładach, zwaną metodą eliminacji. Czasami mówi się o metodzie akceptacji-eliminacji (ang. acceptance-elimination method). Metoda

eliminacji pochodzi od von Neumanna. Zaczniemy od zaprezentowania jej dyskretnej wersji. Przypuśćmy, że chcemy generować zmienną losową X z funkcją prawdopodobieństwa $(p_j, j \in E)$, gdzie E jest przestrzenią wartości X (na przykład $\mathbb{Z}$, $\mathbb{Z}_+$, $\mathbb{Z}^2$, itd), podczas gdy znamy algorytm generowania liczby losowej Y z funkcją prawdopodobieństwa $(p'_j, j \in E)$ oraz gdy istnieje $c > 0$ dla którego

$$p_j \leq cp'_j, \quad j \in E.$$

Oczywiście takie c zawsze znajdziemy gdy E jest skończona, ale tak nie musi być gdy E jest przeliczalna; polecamy czytelnikowi podać przykład. Oczywiście $c > 1$, jeśli funkcje prawdopodobieństwa są różne (dlaczego?). Generujemy niezależne $U \sim U(0, 1)$ i Y z funkcją prawdopodobieństwa $(p'_j, j \in E)$ i definiujemy *obszar akceptacji* przez

$$A = \{cU p'_Y \leq p_Y\}.$$

Podstawą naszego algorytmu będzie następujący fakt.

Fakt 6.1

$$\mathbb{P}(Y = j|A) = p_j, \quad j \in E$$

oraz $\mathbb{P}(A) = 1/c$

Dowód Bez straty ogólności możemy założyć, że $p'_j > 0$ dla dowolnego $j \in E$ (dlaczego?). Jeśli by dla jakiegoś j mielibyśmy $p'_j = 0$, to wtedy również $p_j = 0$ i ten stan można wyeliminować. Dalej mamy

$$\begin{aligned} \mathbb{P}(Y = j|A) &= \frac{\mathbb{P}(\{Y = j\} \cap \{cUp'_Y \leq p_Y\})}{\mathbb{P}(A)} \\ &= \frac{\mathbb{P}(Y = j, U \leq p_j/(cp'_j))}{\mathbb{P}(A)} \\ &= \frac{\mathbb{P}(Y = j)\mathbb{P}(U \leq p_j/(cp'_j))}{\mathbb{P}(A)} \\ &= \frac{p'_j p_j/(cp'_j)}{p'_j \mathbb{P}(A)} = p_j \frac{c}{\mathbb{P}(A)}. \end{aligned}$$

Teraz musimy skorzystać z tego, że

$$1 = \sum_{j \in E} \mathbb{P}(Y = j|A) = \frac{1}{c\mathbb{P}(A)}.$$

□

Następujący algorytm podaje przepis na generowanie liczby losowej X z funkcją prawdopodobieństwa $(p_j, j \in E)$.

Algorytm RM-d

```
% dane f. prawd. (p(j)) i (p'(j))
1. generuj Y z funkcją prawdopodobieństwa (p'(j));
2. jeśli c rand p'(Y) <= p(Y), to podstaw X:=Y;
3. w przeciwnym razie idź do 1.
```

Widzimy, że liczba powtórzeń L w pętli ma rozkład obcięty geometryczny $\text{TGeo}(\mathbb{P}(A)) = \text{TGeo}(1 - \frac{1}{c})$; $\mathbb{P}(L = k) = \frac{1}{c}(1 - \frac{1}{c})^{k-1}$, $k = 1, 2, \dots$. Oczekiwana liczba powtórzeń wynosi c , a więc należy dążyć aby c było jak najmniejsze.

Podamy teraz wersję ciągłą metody eliminacji. Przypuśćmy, że chcemy generować liczbę losową X z gęstością $f(x)$ na E (możemy rozważać $E = \mathbb{R}_+, \mathbb{R}, \mathbb{R}^d$, itd.), podczas gdy znamy algorytm generowania liczby losowej Y o wartościach w E z gęstością $g(x)$. Niech $c > 0$ będzie takie, że

$$f(x) \leq cg(x), \quad x \in E.$$

Oczywiście $c > 1$, jeśli gęstości są różne. Również bez straty ogólności możemy przyjąć, że jeśli dla pewnego $x \in E$, $g(x) = 0$, to też $f(x) = 0$. Niech $U \sim U(0, 1)$ i definiujemy *obszar akceptacji* przez

$$A = \{cUg(Y) \leq f(Y)\}.$$

Zakładamy, że U i Y są niezależne. Podstawą naszego algorytmu będzie następujący fakt, który wysłowimy dla $E \subset \mathbb{R}^n$.

Fakt 6.2

$$\mathbb{P}(Y \in B|A) = \int_B f(y) dy, \quad \text{dla } B \subset E$$

oraz $\mathbb{P}(A) = 1/c$.

Dowód Łączny rozkład wektora (U, Y) ma gęstość $du \times g(y) dy$ na $[0, 1) \times E$.
Mamy

$$\begin{aligned} \mathbb{P}(Y \in B|A) &= \frac{\mathbb{P}(Y \in B, cUg(Y) \leq f(Y))}{\mathbb{P}(A)} \\ &= \frac{\int_E \int_{[0,1)} \mathbf{1}(y \in B, cug(y) \leq f(y)) du g(y) dy}{\mathbb{P}(A)} \\ &= \frac{\int_B (f(y)/(cg(y))) g(y) dy}{\mathbb{P}(A)} \\ &= \int_B f(y) dy \frac{1}{c\mathbb{P}(A)}. \end{aligned}$$

Podstawiając $B = E$ widzimy, że $\mathbb{P}(A) = 1/c$. Ponieważ równość jest dla dowolnych $B \subset E$, więc f jest szukaną gęstością, co kończy dowód. $\square$

Następujący algorytm podaje przepis na generowanie liczby losowej X z gęstością $f(x)$.

Algorytm RM

```
%zadana gęstość f(x)
1. generuj Y z gęstością g(y);
2. jeśli c* rand * g(Y) <= f(Y) to podstaw X:=Y
3. w przeciwnym razie idź do 1.
```

Przykład 6.3 [Normalna zmienna losowa] Pokażemy, jak skorzystać z metody eliminacji aby wygenerować zmienną losową Z o rozkładzie normalnym $N(0, 1)$. Najpierw wygenerujemy zmienną losową X z gęstością

$$f(x) = \frac{2}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}, \quad 0 < x < \infty. \quad (6.8)$$

Taki rozkład nosi nazwę połówkowego normalnego (ang. half-normal distribution). Jest to gęstość zmiennej losowej Z , gdzie Z ma rozkład standardowy normalny. Zmienna losowa w tle Y ma rozkład wykładniczy $\text{Exp}(1)$ z gęstością $g(x) = e^{-x}$, $x > 0$. Należy znaleźć maksimum funkcji

$$\frac{f(x)}{g(x)} = \frac{2}{\sqrt{2\pi}} e^{x - \frac{x^2}{2}}, \quad 0 < x < \infty,$$

co się sprowadza do szukania wartości dla której $x - x^2/2$ jest największa w obszarze $x > 0$. Maksimum jest osiągnięte w punkcie 1 co pociąga

$$c = \max \frac{f(x)}{g(x)} = \sqrt{\frac{2e}{\pi}} \approx 1.32 .$$

Akceptacja jest przy $U \leq f(Y)/(cg(Y))$. Zauważmy, że przy symulacjach możemy wykorzystać

$$\frac{f(x)}{cg(x)} = e^{x-x^2/2-1/2} = e^{-(x-1)^2/2} .$$

Teraz aby wygenerować zmienną losową Z o rozkładzie $\mathcal{N}(0, 1)$ musimy zauważyć, że należy wpierw wygenerować X z gęstością $f(x)$ i następnie wylosować znak plus z prawdopodobieństwem $1/2$ i minus z prawdopodobieństwem $1/2$ (patrz Zadanie 9.17). To prowadzi do następującego algorytmu na generowanie liczby losowej o rozkładzie $\mathcal{N}(0, 1)$.

Algorytm RM-N

```
% na wyjściu Z o rozkładzie N(0,1)
1. generuj Y=-\log rand;
2. jeśli U<= exp(-(Y-1)^2/2) to podstaw X:=Y;
3. w przeciwnym razie idź do 1.
4. generuj I=2*floor(2*rand)-1;
5. Podstaw Z:=I*X;
```

Przykład 6.4 [Rozkład Gamma(α, λ); Madras [33]]

Przypadek $\alpha < 1$ Bez straty ogólności można założyć $\lambda = 1$ (Dlaczego? Jak otrzymać z X z gęstością $f(x)$ o rozkładzie Gamma($\alpha, 1$) zmienną Y o rozkładzie Gamma(α, β)? Patrz Zadanie 9.17). Teraz za $g(x)$ weźmiemy

$$g(x) = \begin{cases} Cx^{\alpha-1}, & 0 < x < 1 \\ Ce^{-x}, & x \geq 1 \end{cases}$$

Z warunku, że $\int_0^\infty g(x) dx = 1$ wyznaczamy

$$C = \frac{\alpha e}{\alpha + e} .$$

Jeśli więc położymy $g_1(x) = \alpha x^{\alpha-1} \mathbf{1}(0 < x < 1)$ i $g_2(x) = e e^{-x} \mathbf{1}(x \geq 1)$ to

$$g(x) = p_1 g_1(x) + p_2 g_2(x) ,$$

gdzie $p_1 = C/\alpha$ oraz $p_2 = C/e$. A więc możemy symulować liczbę losową Y z gęstością g używając metody superpozycji. Zostawiamy czytelnikowi napisanie procedury generowania $g_1(x)$ oraz $g_2(x)$. Teraz należy zauważyć, że istnieje c takie, że

$$f(x) \leq c g(x), \quad x \in \mathbb{R}_+,$$

co pozwala wykorzystać metodę eliminacji do symulacji zmiennej X . Zostawiamy czytelnikowi napisanie całej procedury.

Przypadek $\alpha > 1$. Oczywiście przypadek $\alpha = 1$ jest nam dobrze znany, ponieważ wtedy mamy do czynienia z liczbą losową o rozkładzie wykładniczym. Jeśli $\alpha > 1$, to przyjmujemy $g(x) = b e^{-bx} \mathbf{1}(0 \leq x < \infty)$ z $\lambda < 1$. Oczywiście w tym przypadku

$$c = c(\lambda) = \sup_{x \geq 0} \frac{f(x)}{g(x)} = \sup_{x \geq 0} \frac{1}{b\Gamma(\alpha)} x^{\alpha-1} e^{-(1-\lambda)x} < \infty .$$

Stałą b powinno się dobrać tak aby $c(\lambda)$ było minimalne. Można pokazać, że

$$c = c(b) = \frac{1}{b\Gamma(\alpha)} \frac{(\alpha-1)^{\alpha-1}}{(1-b)^{\alpha-1}} e^{-(\alpha-1)}.$$

7 Generowanie zmiennych gaussowskich

7.1 Metody ad hoc dla $N(0,1)$.

Przedstawimy dwa warianty metody, zwanej Boxa-Mullera, na generowanie pary niezależnych zmiennych losowych Z_1, Z_2 o jednakowym rozkładzie normalnym $N(0,1)$. Pomysł polega na wykorzystaniu współrzędnych biegunowych. Zaczniemy od udowodnienia lematu.

Lemat 7.1 *Niech Z_1, Z_2 będą niezależnymi zmiennymi losowymi o jednakowym rozkładzie normalnym $N(0,1)$. Niech (R, Θ) będzie przedstawieniem punktu (Z_1, Z_2) we współrzędnych biegunowych. Wtedy R i Θ są niezależne, R ma gęstość*

$$f(x) = x e^{-x^2/2} \mathbf{1}(x > 0) \tag{7.9}$$

oraz Θ ma rozkład jednostajny $U(0, 2\pi)$.

Dowód Oczywiście, (Z_1, Z_2) można wyrazić z (R, Θ) przez

$$\begin{aligned} Z_1 &= R \cos \Theta, \\ Z_2 &= R \sin \Theta \end{aligned}$$

a wektor zmiennych losowych Z_1, Z_2 ma łączną gęstość $\frac{1}{2\pi} \exp(-(x_1^2 + x_2^2)/2)$. Skorzystamy z wzoru (XI.1.1). Przejście do współrzędnych biegunowych ma jacobian

$$\begin{vmatrix} \cos \theta & -r \sin \theta \\ \sin \theta & r \cos \theta \end{vmatrix} = r$$

a więc (R, Θ) ma gęstość

$$f_{(R, \Theta)}(r, \theta) = \frac{1}{2\pi} e^{-\frac{r^2}{2}} r \, dr d\theta, \quad 0 < \theta \leq 2\pi, \quad r > 0.$$

□

Zauważmy, że rozkład zadany wzorem (7.9) nazywa się *rozkładem Raleigha*.

W następującym lemacie pokażemy jak generować zmienne o rozkładzie Raleigha.

Lemat 7.2 *Jeśli R ma rozkład Raleigha, to $Y = R^2$ ma rozkład wykładniczy $\text{Exp}(1/2)$. W konsekwencji, jeśli U ma rozkład jednostajny $U(0,1)$, to zmienna losowa $(-2 \log U)^{1/2}$ ma rozkład Raleigha.*

Dowód W tym celu zauważmy, że $Y = R^2$ ma rozkład wykładniczy $\text{Exp}(1/2)$ bo

$$\begin{aligned} \mathbb{P}(R^2 > x) &= \mathbb{P}(R > \sqrt{x}) \\ &= \int_{\sqrt{x}}^{\infty} r e^{-r^2/2} \, dr = e^{-x/2}, \quad x > 0. \end{aligned}$$

Ponieważ $(-2 \log U)$ ma rozkład wykładniczy $\text{Exp}(1/2)$, więc jak wcześniej zauważyliśmy, pierwiastek z tej zmiennej losowej ma rozkład Raleigha.

□

W rezultacie mamy następujący fakt.

Fakt 7.3 Niech U_1 i U_2 będą niezależnymi zmiennymi losowymi o jednakowym rozkładzie jednostanym $U(0, 1)$. Niech

$$\begin{aligned} Z_1 &= (-2 \log U_1)^{1/2} \cos(2\pi U_2), \\ Z_2 &= (-2 \log U_1)^{1/2} \sin(2\pi U_2). \end{aligned}$$

Wtedy Z_1, Z_2 są niezależnymi zmiennymi losowymi o jednakowym rozkładzie normalnym $N(0, 1)$.

Algorytm BM

```
% na wyjściu niezależne Z1, Z2 o rozkładzie N(0,1)
Generuj U1,U2
R<- -2*log U1
V <- 2*Pi*U2
Z1 <- sqrt(R)* cos V
Z2 <- sqrt(R)* sin V
zwrot Z1,Z2
```

Dowód Mamy $Z_1 = R \cos \Theta$, $Z_2 = R \sin \Theta$, gdzie R, Θ są niezależne, R ma rozkład Raleigha a Θ jednostajny na $(0, 2\pi]$. Trzeba wygenerować zmienną Θ o rozkładzie jednostajnym na $(0, 2\pi]$ (proste) oraz R z gęstością $re^{-\frac{r^2}{2}}$, $r > 0$ co wiemy jak robić z lematu 7.2

Druga metoda jest modyfikacją Box-Mullera i pochodzi od Marsaglia-Breya. Opiera się ona na następującym fakcie.

Fakt 7.4 Niech

$$\begin{aligned} Y_1 &= \{-2 \log(V_1^2 + V_2^2)\}^{1/2} \frac{V_1}{(V_1^2 + V_2^2)^{1/2}}, \\ Y_2 &= \{-2 \log(V_1^2 + V_2^2)\}^{1/2} \frac{V_2}{(V_1^2 + V_2^2)^{1/2}}. \end{aligned}$$

i V_1, V_2 są niezależnymi zmiennymi losowymi o jednakowym rozkładzie jednostajnym $U(-1, 1)$. Wtedy

$$(Z_1, Z_2) =_d ((Y_1, Y_2) | V_1^2 + V_2^2 \leq 1)$$

jest parą niezależnych zmiennych losowych o jednakowym rozkładzie normalnym $N(0, 1)$.

Dowód powyższego faktu wynika z następującego lematu. Niech K będzie kołem o promieniu 1 i środku w początku układu $\mathbb{R}^2$.

Lemat 7.5 *Jeśli $(W_1, W_2) \sim U(K)$, i (R, Θ) jest przedstawieniem punktu (W_1, W_2) w współrzędnych biegunowych, to R i Θ są niezależne, R^2 ma rozkład jednostajny $U(0,1)$ oraz Θ ma rozkład jednostajny $U(0, 2\pi)$.*

Dowód Łączny rozkład (W_1, W_2) ma gęstość

$$\frac{1}{\pi} \mathbf{1}((w_1, w_2) \in K) dw_1 dw_2.$$

Mamy

$$W_1 = R \cos \Theta, \quad W_2 = R \sin \Theta.$$

Niech $B = \{t_1 < \theta \leq t_2, a < r \leq b\}$. Wtedy

$$\begin{aligned} \mathbb{P}(t_1 < \Theta \leq t_2, a < R \leq b) &= \mathbb{P}((W_1, W_2) \in B) \\ &= \frac{1}{\pi} \int_B \mathbf{1}((x, y) \in K) dx dy \\ &= \frac{1}{\pi} \int_{t_1}^{t_2} \int_a^b r dr d\theta \\ &= \frac{1}{2\pi} (t_2 - t_1) \times (b^2 - a^2) \\ &= \mathbb{P}(t_1 < \Theta \leq t_2) \mathbb{P}(a < R \leq b). \end{aligned}$$

Stąd mamy, że R i Θ są niezależne, Θ o rozkładzie jednostajnym $U(0, 2\pi)$. Ponadto $\mathbb{P}(R \leq x) = x^2 \mathbf{1}(0 \leq x \leq 1)$, czyli R^2 ma rozkład jednostajnym $U(0,1)$.

Algorytm MB

```
% na wyjściu niezależne Z1, Z2 o rozkładzie N(0,1)
while (X>1)
  generate U1,U2
  U1 <- 2*U1-1, U2 <- 2*U2-1
  X = U1^2+U2^2
end
Y <- sqrt((-2* log X)/X)
Z1 = U1*Y, Z2=U2*Y
return Z1,Z2
```

7.2 Metoda ITM dla zmiennych normalnych

² Aby generować standardowe zmienne normalne $\mathcal{N}(0, 1)$ metodą ITM musimy umieć odwrócić dystrybuantę $\Phi(x)$. Niestety w typowych pakietach nie ma zaszytej funkcji $\Phi^{-1}(u)$ (tutaj mamy klasyczną funkcję odwrotną). Istnieją rozmaite algorytmy obliczające $\Phi^{-1}(u)$. Jak wszystkie algorytmy zaszyte w pakietach lub kalkulatorach podają przybliżone wartości z pewnym błędem, który powinien być podany. Zagadnieniem jest rozwiązanie równania $\Phi(x) = u$ dla zadanego $u \in (0, 1)$. Punktem wyjścia może być standardowa metoda Newtona szukania zera funkcji $x \rightarrow \Phi(x) - u$. Zauważmy najpierw, że ponieważ

$$\Phi^{-1}(1 - u) = -\Phi(u)$$

więc wystarczy szukać rozwiązania albo dla $u \in [1/2, 1)$ lub $(0, 1/2)$. Metoda Newtona jest metodą iteracyjną zadaną rekurencją

$$\begin{aligned} x_{n+1} &= x_n - \frac{\Phi(x_n) - u}{\phi(x_n)} \\ &= x_n + (u - \Phi(x_n))e^{0.5x_n^2 + c}, \end{aligned} \quad (7.10)$$

gdzie $c = \log \sqrt{2\pi}$. Sugeruje się aby zacząć z punktu

$$x_0 = \pm \sqrt{|-1.6 \log(1.0004 - (1 - 2u)^2)|}.$$

Powyżej znak zależy od tego czy $u \geq 1/2$ (plus) czy $u < 1/2$ (minus). Powyższa procedura wymaga znajomości funkcji Φ i oczywiście wykładniczej. Poniżej przedstawiamy inny algorytm, tzw. algorytm Beasley'a-Zamana-Moro (za książką Glassermana [16]).

```
function x=InvNorm(u)
```

```
a0=2.50662823884;  
a1=-18.61500062529;  
a2=41.39119773534;  
a3=-25.44106049637;
```

```
b0=-8.47351093090;  
b1=23.08336743743;  
b2=-21.06224101826;
```

²Glasserman [16] str. 67.

```

b3=3.13082909833;

c0=0.3374754822726147;
c1=0.9761690190917186;
c2=0.1607979714918209;
c3=0.0276438810333863;
c4=0.0038405729373609;
c5=0.0003951896511919;
c6=0.0000321767881768;
c7=0.0000002888167364;
c8=0.0000003960315187;

y<-u-0.5;
if |y|<0.42;
r <- y*y
x <- (((a3*r+a2)*r+a1)*r+a0)/((((b3*r+b2)*r+b1)*r+b0)+1)
else
r <- u
if (y>0) r <- 1-u
r <- log (-log)(r)
x <- c0+r*(c1+r*(c2+r*(c3+r*(c4+r*(c5+r*(c6+r*(c7+r*c8))))))
if (y<0) x<- -x
return x

```

Aby to jeszcze polepszyć można zastosować do wyniku jeszcze jeden krok według rekurencji (7.10) wychodząc z otrzymanego $x = \Phi^{-1}(u)$. Glasserman [16] podaje, że wtedy błąd nie przekracza 10^{-15} .

7.3 Generowanie wektorów losowych $N(\mathbf{m}, \Sigma)$.

Wektor losowy $\mathbf{X}$ o rozkładzie wielowymiarowym normalnym $N(\mathbf{m}, \Sigma)$ ma średnią $\mathbf{m}$ oraz macierz kowariancji Σ oraz dowolna kombinacja $\sum_{j=1}^n a_j X_j$ ma rozkład normalny. Oczywiście gdy mówimy o n -wymiarowym wektorze normalnym $\mathbf{X}$, to $\mathbf{m}$ jest n wymiarowym wektorem, Σ jest macierzą $n \times n$. Przypomnijmy też, że macierz kowariancji musi być symetryczna i nieujemnie określona. W praktyce tego wykładu będziemy dodatkowo zakładać, że macierz Σ jest niesingularna. Wtedy rozkład $\mathbf{X} \sim N(\mathbf{0}, \Sigma)$ ma gęstość zadaną

wzorem

$$f(x_1, \dots, x_n) = \frac{1}{(2\pi \det(\Sigma))^{n/2}} e^{-\frac{\sum_{j,k=1}^n c_{jk} x_j x_k}{2}}.$$

gdzie

$$\mathbf{C} = (c_{jk})_{j,k=1}^n = \Sigma^{-1}$$

Przykład 7.6 Dla $n = 2$ macierz kowariancji można przedstawić jako

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix},$$

gdzie $\rho = \text{Corr}(X_1, X_2)$, $\sigma_i^2 = \text{Var } X_i$ ($i = 1, 2$). Wtedy

$$\mathbf{C} = \frac{1}{1 - \rho^2} \begin{pmatrix} \frac{1}{\sigma_1^2} & -\frac{\rho}{\sigma_1\sigma_2} \\ -\frac{\rho}{\sigma_1\sigma_2} & \frac{1}{\sigma_2^2} \end{pmatrix}.$$

Aby wygenerować taki dwuwymiarowy wektor normalny $\mathbf{X} \sim N(0, 1)$, niech

$$Y_1 = \sigma_1(\sqrt{1 - |\rho|}Z_1 + \sqrt{|\rho|}Z_3), \quad Y_2 = \sigma_2(\sqrt{1 - |\rho|}Z_2 + \text{sign}(\rho)\sqrt{|\rho|}Z_3),$$

gdzie Z_1, Z_2, Z_3 są niezależnymi zmiennymi losowymi o rozkładzie $N(0, 1)$. Teraz dostaniemy szukany wektor losowy $(X_1, X_2) = (Y_1 + m_1, Y_2 + m_2)$. To że (Y_1, Y_2) ma dwuwymiarowy rozkład normalny jest oczywiste. Należy pokazać, że macierz kowariancji tego wektora jest taka jak trzeba.

Dla dowolnego n , jeśli nie ma odpowiedniej struktury macierzy kowariancji Σ pozwalającej na znalezienie metody ad hoc polecana jest następująca metoda generowania wektora losowego $N(\mathbf{m}, \Sigma)$. Wektory poniżej są wektorami kolumnowymi. Niech $\mathbf{X} = (X_1, \dots, X_n)^T$ oraz macierz $\mathbf{A}$ będzie taka, że

$$\Sigma = \mathbf{A}\mathbf{A}^T. \quad (7.11)$$

Macierz $\mathbf{A}$ możemy nazywać pierwiastkiem z macierzy Σ . Każda macierz kowariancji jest symetryczną macierzą nieujemnie określoną, a wiadomo, że dla takich macierzy istnieje pierwiastek.

Fakt 7.7 Jeśli $\mathbf{Z} = (Z_1, \dots, Z_n)$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie normalnym $N(0, 1)$ to

$$\mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{m}$$

ma rozkład $N(\mathbf{m}, \Sigma)$.

Aby obliczyć ‘pierwiastek’ $\mathbf{A}$ z macierzy kowariancji Σ można skorzystać z gotowych metod numerycznych. Na przykład w MATLAB-ie jest instrukcja `chol( $\Sigma$ )`, która używa metody faktoryzacji Choleskiego. W wyniku otrzymujemy macierz górnotrójkątną. Zauważmy, że jeśli Σ jest nieosobliwa, to istnieje jedyna A dla której faktoryzacja (7.11) zachodzi i na przekątnej macierz A ma dodatnie elementy. W przypadku gdy Σ jest singularna to taki rozkład nie jest jednoznaczny.

Proponujemy następujący eksperyment w MATLABIE.

```
>> Sigma=[1,1,1;1,4,1;1,1,8]
```

```
Sigma =
```

```

    1    1    1
    1    4    1
    1    1    8
```

```
>> A=chol(Sigma)
```

```
A =
```

```

1.0000    1.0000    1.0000
         0    1.7321         0
         0         0    2.6458
```

```
>> A'*A
```

```
ans =
```

```

1.0000    1.0000    1.0000
1.0000    4.0000    1.0000
1.0000    1.0000    8.0000
```

8 Przykładowe symulacje

W dwóch wykresach na rys. 8.2 i 8.3 podajemy wyniki S_n/n , $n = 1, \dots, 5000$ dla odpowiednio rozkład wykładniczego $\text{Exp}(1)$ i Pareto ze średnią 1. Zmienne

Pareto $\text{Par}(1,2)$ są w tym celu pomnożone przez 0.2. Warto zwrócić uwagę na te symulacje. Widać, że dla zmiennych Pareto zbieżność jest dramatycznie wolniejsza. Jest to efekt ciężko-ogonowości.

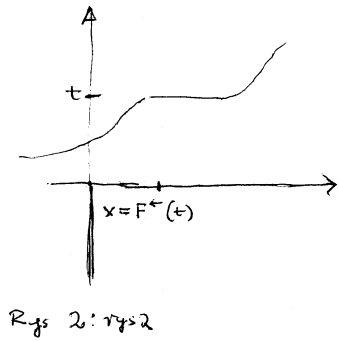
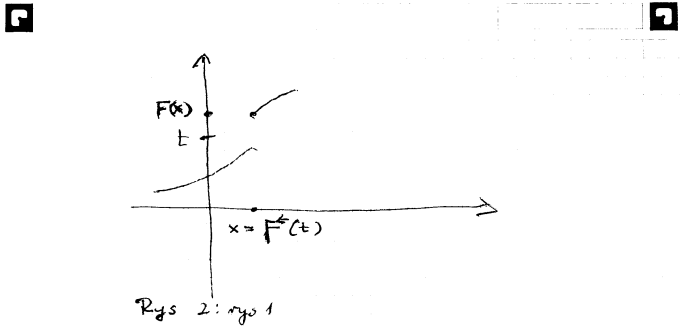
Na rys. 8.4 podane są wyniki symulacji, kolejno 5000 replikacji rozkładu normalnego $N(1,1)$, 5000 replikacji rozkładu wykładniczego $\text{Exp}(1)$, oraz 5000 replikacji rozkładu Pareto(1,2) przemnożonego przez 0.2. Polecamy zwrócić uwagę na charakter poszczególnych fragmentów na rys. 8.4. Chociaż wszystkie zmienne mają średnią 1, to w pierwszym fragmencie (dla zmiennych o rozkładzie normalnym) zmienne są mało rozrzucone, w drugiej części trochę bardziej, natomiast duży rozrzut można zaobserwować w części ostatniej (tj. dla zmiennych Pareto). Jest to spowodowane coraz cięższym ogonem rozkładu. Do symulacji zmiennych losowych o rozkładzie normalnym $N(0,1)$ użyto MATLABowskiej komendy `randn`. Aby uzyskać zmienne $N(1,1)$ trzeba wziąć `randn+1`.

Dla ilustracji na rysunkach 8.5, 8.6 i 8.7 podane są przykładowe symulacje dwuwymiarowego wektora normalnego z zerowym wektorem średniej, $\text{Var}(X_1) = 3$, $\text{Var}(X_2) = 1$ i współczynnika korelacji $\rho = 0.1, 0.6, 0.9$.³

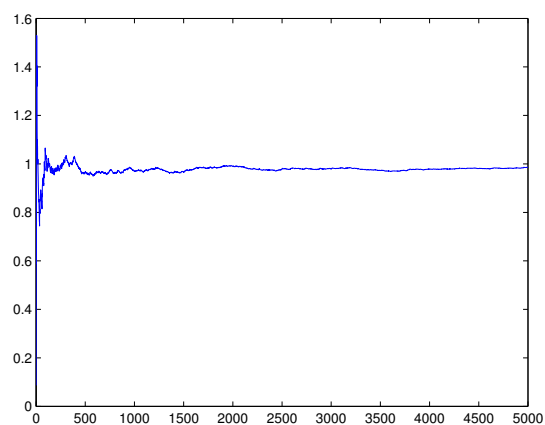
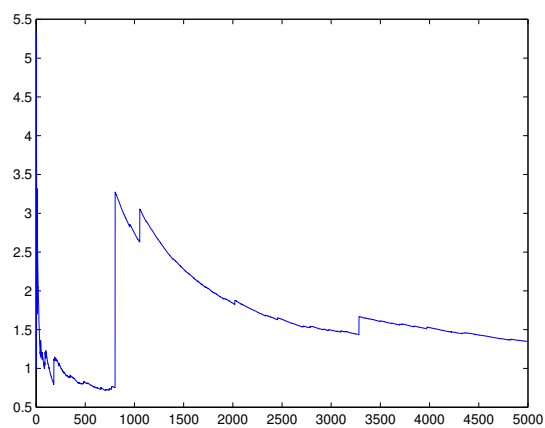
8.1 Uwagi bibliograficzne

Na temat generowania zmiennych losowych o zadanych rozkładach istnieje obszerna literatura książkowa: patrz e.g. elementarny podręcznik Rossa [40], fundamentalna monografia Asmussena i Glynn [4], obszerna monografia Fishmana [14], [9], bardzo ciekawy nieduży podręcznik Madrasa [33].

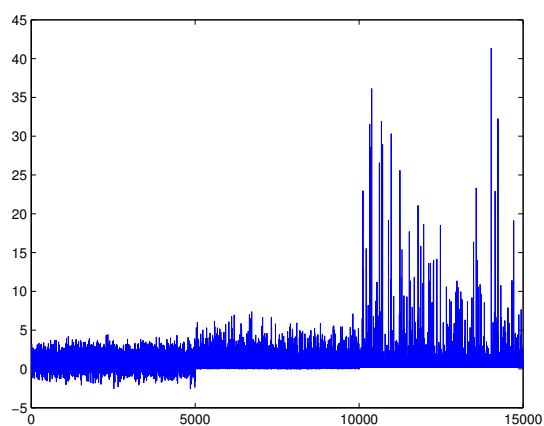
³skrypt: `normalnyRho.m`



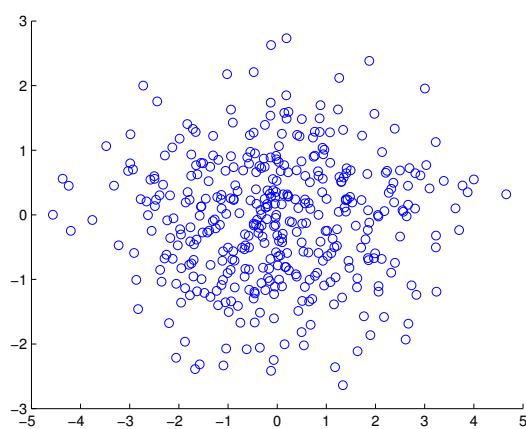
Rysunek 1.1: Rys 2-1

Rysunek 8.2: Rozkład wykładniczy $\text{Exp}(1)$, 5000 replikacji

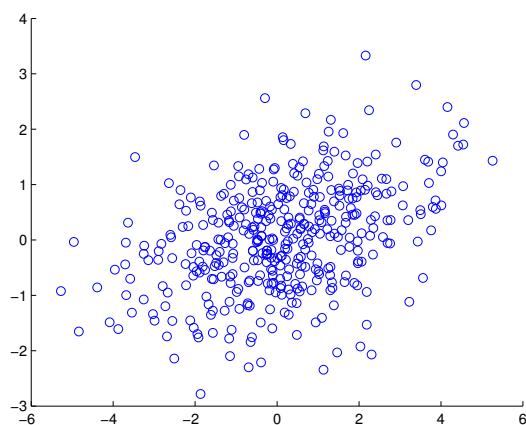
Rysunek 8.3: Rozkład Pareto(1.2), 5000 replikacji



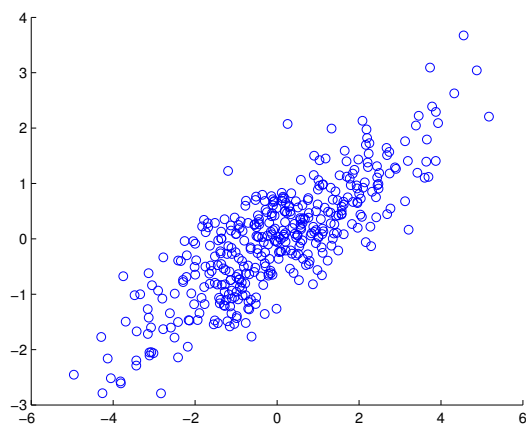
Rysunek 8.4: Rozkład $N(1,1)$ (5000 repl.), $\text{Exp}(1)$ (5000 repl.), $\text{Pareto}(1.2)$, (5000 repl.)



Rysunek 8.5: Wektor normalny, $\rho = 0.1$, $\sigma_1^2 = 3$, $\sigma_2^2 = 1$, 400 replikacji



Rysunek 8.6: Wektor normalny, $\rho = 0.6$, $\sigma_2^2 = 3$, $\sigma_1^2 = 1$, 400 replikacji



Rysunek 8.7: Wektor normalny, $\rho = 0.9$, $\sigma_1^2 = 3$, $\sigma_2^2 = 1$, 400 replikacji

9 Zadania

4

Zadania teoretyczne

- 9.1 Niech X ma dystrybuantę F i chcemy wygenerować zmienną losową o warunkowym rozkładzie $X|X \in (a, b)$ gdzie $\mathbb{P}(X \in (a, b)) > 0$. Niech

$$V = F(a) + (F(b) - F(a))U.$$

Jaki rozkład ma V . Pokazać, że $Y = F^{-\leftarrow}(V)$ ma żadaną warunkową dystrybuantę G , gdzie

$$G(x) = \begin{cases} 0 & x < a, \\ \frac{F(x) - F(a)}{F(b) - F(a)}, & a \leq x < b, \\ 1 & x \geq b \end{cases}$$

- 9.2 Podać procedurę generowania liczb losowych o rozkładzie *trójkątnym* $\text{Tri}(a, b)$, z gęstością

$$f(x) = \begin{cases} c(x - a), & a < x < (a + b)/2, \\ c(b - x), & (a + b)/2 < x < b, \end{cases}$$

gdzie stała normująca $c = 4/(b - a)^2$. Wsk. Rozważyć najpierw gęstość $U_1 + U_2$.

- 9.3 Pokazać, że

$$X = \log(U/(1 - U))$$

ma rozkład *logistyczny* o dystrybuancie $F(x) = 1/(1 + e^{-x})$.

- 9.4 Podać dwie procedury na generowanie liczby losowej z gęstością postaci

$$f(x) = \sum_{j=1}^N a_j x^j, \quad 0 \leq x \leq 1,$$

gdzie $a_j > 0$. Wsk. Obliczyć gęstość $\max(U_1, \dots, U_n)$ i następnie użyć metody superpozycji. Jaki warunek muszą spełniać (a_i) ?

⁴Poprawiane 1.12.2017

9.5 Podać procedurę generowania zmiennej losowej X z gęstością $f(x) = n(1-x)^{n-1}$.

9.6 Podać dwie procedury generowania liczby losowej o rozkładzie Cauchego.

9.7 Niech F jest dystrybuantą rozkładu Weibulla $W(\alpha, c)$,

$$F(x) = \begin{cases} 0 & x < 0 \\ 1 - \exp(-cx^\alpha), & x \geq 0. \end{cases}$$

Zakładamy $c > 0, \alpha > 0$. Pokazać, że

$$X = \left(\frac{-\log U}{c} \right)^{1/\alpha}$$

ma rozkład $W(\alpha, c)$.

9.8 Pokazać, że $X = \tan(\pi(U - 1/2))$ ma rozkład Cauchego, tj o gęstości

$$f(x) = \frac{1}{\pi} \frac{1}{1+x^2}.$$

9.9 Rozkład Gumbela z parametrami $\mu \in \mathbb{R}$ i $\sigma > 0$ ma dystrybuantę

$$F(x; \mu, \sigma) = \exp(e^{-\frac{x-\mu}{\sigma}}), \quad x \in \mathbb{R}.$$

Pokazać, że zmienna $X = -\log(-\log U)$ ma rozkład Gumbela z parametrami $\mu = 0$ i $\sigma = 1$. Jak z X uzyskać rozkład Gumbela z dowolnymi parametrami μ, σ .

9.10 Podać przykład pokazujący, że c w metodzie eliminacji (zarówno dla przypadku dyskretnego jak i ciągłego) nie musi istnieć skończone.

9.11 a) Udowodnić, że $\mathbb{P}(\lfloor U^{-1} \rfloor = i) = \frac{1}{i(i+1)}$, dla $i = 1, 2, \dots$

b) Podać algorytm i obliczyć prawdopodobieństwo akceptacji na wygenerowanie metodą eliminacji dyskretnej liczby losowej X z funkcją prawdopodobieństwa $\{p_k, k = 1, 2, \dots\}$, gdzie

$$p_k = \frac{6}{\pi^2} \frac{1}{k^2}, \quad k = 1, 2, \dots$$

c) Podać algorytm i obliczyć prawdopodobieństwo akceptacji na wygenerowanie metodą eliminacji dyskretnej liczby losowej X z funkcją prawdopodobieństwa $\{p_k, k = 1, 2, \dots\}$, gdzie $p_k = \frac{90}{\pi^4} \frac{1}{n^4}$.

9.12 Podaj metodę generowania liczby losowej z gęstością:

$$f(x) = \begin{cases} e^{2x}, & -\infty < x < 0 \\ e^{-2x}, & 0 \leq x < \infty \end{cases}$$

Taki rozkład nazywa się *podwójny wykładniczy*.

9.13 W przykładzie 6.4, przypadek $\alpha > 1$, dobrać optymalne b tak aby $c = c(b)$ było minimalne.

9.14 *Rozkład beta* $\text{Beta}(\alpha, \beta)$ ma gęstość

$$f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)}, \quad 0 < x < 1.$$

gdzie beta funkcja jest zdefiniowana przez

$$B(\alpha, \beta) = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1} dx.$$

Przy założeniu $\alpha, \beta > 1$ podać algorytm wraz z usadnieniem generowania liczb losowych $\text{Beta}(\alpha, \beta)$ metodą eliminacji. Wsk. Przyjąć $g(x) = \mathbf{1}(0 < x < 1)$.

9.15 Gęstość rozkładu zadana jest wzorem

$$f(x) = Cxe^{-x^4}, \quad x \geq 0,$$

gdzie $C = 4/\sqrt{\pi}$. Podać dwie procedury generowania zmiennych losowych z gęstością f .

9.16 Gęstość dwuwymiarowego wektora losowego $\mathbf{Y} = (Y_1, Y_2)$ jest dana

$$f(x_1, x_2) = ce^{-|\mathbf{x}|^4} = ce^{-(x_1^2+x_2^2)^2}, \quad \mathbf{x} \in \mathbb{R}^2,$$

gdzie c jest znaną stałą normującą. Zaproponować algorytm na symulację wektora losowego $\mathbf{Y}$. (Wsk. Przedyskutować dwie metody: eliminacji oraz przejście do współrzędnych biegunowych oraz zastanowić się którą lepiej używać).

9.17 a) Niech η będzie zmienną losową niezależną od X , $\mathbb{P}(\eta = -1) = \mathbb{P}(\eta = 1) = 1/2$, oraz X ma rozkład z gęstością daną w (6.8). Pokazać, że ηX ma rozkład standardowy normalny $N(0,1)$.

b) Pokazać, że jeśli X ma rozkład $\text{Gamma}(\alpha, 1)$ to X/β ma rozkład $\text{Gamma}(\alpha, \beta)$.

- 9.18 Podać dwa algorytmy na generowanie punktu losowego o rozkładzie $U(B)$, gdzie $B = \{x \in \mathbb{R}^2 : |x| \leq 1\}$. Wsk. Do jednego z algorytmów wykorzystać lemat 7.5.
- 9.19 Niech (X, Y) będzie wektorem losowym z gęstością $f(x, y)$ i niech $B \subset \mathbb{R}^2$ będzie obszarem takim, że

$$\int_B f(x, y) dx dy < \infty.$$

Pokazać, że $X|(X, Y) \in B$ ma gęstość

$$\frac{\int_{y:(x,y) \in B} f(x, y) dy}{\int_B f(x, y) dx dy}.$$

- 9.20 [Marsaglia [34]] Niech J ma funkcję prawdopodobieństwa $(p_m)_{m=0}^\infty$ z $p_m = 1/(ce^{m+1})$ i $c = 1/(e - 1)$. Niech I ma funkcję prawdopodobieństwa $(q_n)_{n=1}^\infty$, gdzie $q_n = c/n!$. Zakładamy, że I, J oraz $U_1, \dots$ są niezależne. Pokazać, że $X = J + \min(U_1, \dots, U_I)$ ma rozkład wykładniczy $\text{Exp}(1)$. Powyższa własność rozkładu wykładniczego daje możliwość napisania algorytmu na generowanie wykładniczych zmiennych losowych bez użycia kosztownej funkcji $\log$. Zastanowić się jak można szybko generować liczby losowe I, J . Wsk. Zauważyć, że I ma rozkład Poissona $\text{Poi}(e)$ przesunięty o 1, natomiast J ma rozkład geometryczny $\text{Geo}(e^{-1})$. Wsk. Zauważyć, że $J \sim \text{Geo}(e^{-1})$ oraz $c_{k+1} = 1/(k + 1)$ ⁵
- 9.21 Uzasadnić dwie metody generowania rozkładu wielomianowego $M(n, \mathbf{a})$. Pierwsza metoda jest uogólnieniem metody ad hoc dla rozkłady dwumianowego (patrz podrozdział 3). Polega na generowaniu niezależnych wektorów losowych przyjmujących wartości $(0, \dots, 1, \dots, 0)$ (jedyne na i -tym miejscu) z prawdopodobieństwem a_i i następnie zsumowanie liczby jedynek na każdej koordynacie. Natomiast druga metoda wykorzystuje unikalną własność rozkładu wielomianowego wektora $\mathbf{X} = (X_1, \dots, X_d)$: rozkład X_{j+1} pod warunkiem $X_1 = k_1, \dots, X_j = k_j$ jest dwumianowy $B(n - k_1 - \dots - k_j, a_{j+1}/(a_1 + \dots + a_j))$ i X_1 ma rozkład $B(n, a_1)$.

⁵A jak szybko generować liczby losowe I, J ?

9.22 a) Pokazać, że jeśli $Z_1, \dots, Z_n$ są niezależnymi zmiennymi o rozkładzie normalnym $N(0,1)$ i $\|Z\| = \sum_{j=1}^n Z_j^2$, to punkt $X = (Z_1/\|Z\|, \dots, Z_n/\|Z\|)$ ma rozkład jednostajny na sferze $S_{n-1} = \{x = (x_1, \dots, x_n) : \sum_{j=1}^n x_j^2 = 1\}$.

b) Jeśli ponadto punkt ze sfery X z punktu a) przemnożymy przez $U^{1/n}$, gdzie U i X są niezależne, to otrzymamy punkt losowy z kuli $B_n = \{x : \sum_{j=1}^n x_j^2 \leq 1\}$.

9.23 ⁶ Niech $\mathbf{X} \in \mathbb{R}^n$ będzie wektorem losowym i $Y \in \mathbb{R}$ będzie zmienną losową.

(i) Jeśli $\mathbf{X}$ ma gęstość $g(\mathbf{x})$ oraz dla $c > 1$, zmienna $(Y|\mathbf{X} = \mathbf{x})$ ma rozkład $U(0, cg(\mathbf{x}))$, to wektor losowy $(\mathbf{X}, Y) \in \mathbb{R}^{n+1}$ ma rozkład $U(\mathcal{A})$, gdzie

$$\mathcal{A} = \{(\mathbf{x}, y) : \mathbf{x} \in \mathbb{R}^n, y \in \mathbb{R}, 0 < y < cg(\mathbf{x})\}.$$

(ii) Jeśli $(n+1)$ wymiarowy wektor losowy

$$(\mathbf{X}, Y)$$

ma rozkład $U(\mathcal{B})$, gdzie

$$\mathcal{B} = \{(\mathbf{x}, y) : \mathbf{x} \in \mathbb{R}^n, y \in \mathbb{R}, 0 < y < f(\mathbf{x})\},$$

to $\mathbf{X}$ ma gęstość $f(\mathbf{x})$.

(iii) Zastanowić się, jak z (i) (ii) oraz rezultatu z sekcji 4 można udowodnić fakt 6.2.

Zadania laboratoryjne

9.24 Podać wraz z uzasadnieniem jak generować liczbę losową X o rozkładzie $U[a,b]$. Napisać MATLABowska funkcję $\text{Unif}(a,b)$ na generowanie liczby losowej X .

⁶wg. Bernard D. Flury, Acceptance-rejection sampling made easy. SIAM Review, vol 32, 474–476, 1990.

- 9.25 W literaturze aktuarialnej rozważa się *rozkład Makehama* przyszłego czasu życia T_x dla x -latka z ogonem rozkładu

$$\mathbb{P}(T_x > t) = e^{-At - m(c^{t+x} - c^x)}.$$

Napisać procedurę `Makeham(A,m,c,x)` generowania liczb losowych o takim rozkładzie. Przyjąć $A = 5 \cdot 10^{-4}$, $m = 7.5858 \cdot 10^{-5}$ oraz $c = \log 1.09144$ (tzw. G82). Wsk. Zinterpretować probabilistycznie jaką operacja prowadzi do faktoryzacji $e^{-At - m(c^{t+x} - c^x)} = e^{-At} e^{-m(c^{t+x} - c^x)}$. Zastanowić się czy rozkład z ogonem dystrybucyj $e^{-m(c^{t+x} - c^x)}$ można przedstawić jako warunkowy.

- 9.26 Napisać procedurę `Poi(lambda)` generowania liczb losowych o rozkładzie $\text{Poi}(\lambda)$. Uzasadnić jej poprawność.
- 9.27 Napisać funkcję `Gamma(a,b)` na generowanie liczby losowej o rozkładzie $\text{Gamma}(a,b)$. Wsk. Algorytm musi się składać z dwóch części: $a < 1$ oraz $a \geq 1$.
- 9.28 Przeprowadzić następujący eksperyment z MATLABowskimi generatorami. Obliczyć objętość kuli $k = 30$ wymiarowej o promieniu 1 i porównać z wynikiem teoretycznym na objętość takiej kuli:

$$V_k = \frac{2}{k} \frac{\pi^{k/2}}{\Gamma(k/2)}.$$

W zadaniu należy przeprowadzić symulacje punktu z $(0,1)^k$ i następnie sprawdzić czy leży on w kuli. Zastanowić się ile należy wziąć replikacji.

- 9.29 Porównać dwie procedury generowania rozkładu normalnego $N(0,1)$ z podsekcji 7.1. Porównać szybkość tych procedur mierząc czas wykonania 100 000 replikacji dla każdej z nich.
- 9.30 Korzystając z zadania 9.21 porównać dwie procedury generowania rozkładu wielomianowego. Do generowania rozkładu dwumianowego wykorzystać procedurę `ITR`. Przeprowadzić eksperyment z $n = 10$ i $a_1 = 0.55$, $a_2 = \dots = a_{10} = 0.05$. Porównać szybkość tych procedur mierząc wykonanie 100 000 replikacji dla każdej z nich.

9.31 Napisać procedurę generowania liczb losowych z gęstością

$$f(x) = pe^{-x} + \frac{(1-p)}{0.2} \left(\frac{1}{1+(x/0.2)} \right)^{1.2}.$$

Zrobić wykładniczy i paretoowski wykres kwantylowy dla 300 zmiennych losowych wygenerowanych z $p = 0.05$ i $p = 0.95$.

Projekt

- 9.32 Napisać procedurę MATLAB-owską generowania liczby losowej $N(0,1)$ (tablicy liczb losowych $N(0,1)$)
- (i) metodą eliminacji (`randnrej`),
 - (ii) 1-szą metodą Boxa-Mullera (`randnbmfirst`),
 - (iii) 2-gą metodą Boxa-Mullera (`randnbmsec`). Przeprowadzić test z pomiarem czasu 100 000 replikacji dla każdej z tych metod oraz MATLAB-owskiej `randn`. Użyć instrukcji `TIC` i `TOC`. Każdy eksperyment zacząć od `rand('state',0)`. Sporządzić raport z eksperymentów. Zrobić wykresy kwantylowe dla każdej z procedur z 1000 replikacji.
- 9.33 Używając rozważania z ćwiczenia 9.20 napisać algorytm na generowanie liczb losowych o rozkładzie wykładniczym $\text{Exp}(1)$. Zastanowić się jak optymalnie generować liczby losowe I, J z zadania 9.20.

Rozdział IV

Analiza wyników; niezależne replikacje

¹ Przypuśćmy, że chcemy obliczyć wartość I o której wiemy, że można przedstawić w postaci

$$I = \mathbb{E} Y$$

dla pewnej zmiennej losowej Y , takiej, że $\mathbb{E} Y^2 < \infty$. Ponadto umiemy również generować zmienną losową Y . Zauważmy, że dla zadanego I istnieje wiele zmiennych losowych, dla których $I = \mathbb{E} Y$. Celem tego rozdziału będzie zrozumienie, jak najlepiej dobrać Y do konkretnego problemu. Przy okazji zobaczymy jak dobrać liczbę replikacji potrzebną do zagwarantowania żądanej dokładności.

1 Przypadek estymacji nieobciążonej

Przypuśćmy, że chcemy obliczyć wartość I o której wiemy, że można przedstawić w postaci

$$I = \mathbb{E} Y$$

dla pewnej liczby losowej Y , takiej, że $\mathbb{E} Y^2 < \infty$. W tym podrozdziale będziemy przyjmować, że $Y_1, \dots, Y_n$ będą niezależnymi replikacjami liczby losowej Y i będziemy rozważać *estymatory* szukanej wielkości I postaci

$$\hat{Y}_n = \frac{1}{n} \sum_{j=1}^n Y_j.$$

¹Poprawiane 17.1.2018

Korzystając z elementarnych faktów teorii prawdopodobieństwa, możemy zauważyć, że estymator $\hat{I}_n$ jest *nieobciążony* t.j.

$$\mathbb{E} \hat{Y}_n = I ,$$

oraz *mocno zgodny* to znaczy z prawdopodobieństwem 1 zachodzi zbieżność dla $n \rightarrow \infty$

$$\hat{Y}_n \rightarrow I ,$$

Mocna zgodność pociąga za sobą ważną dla nas *słabą zgodność*, to znaczy, że dla każdego $b > 0$ mamy

$$\mathbb{P}(|\hat{Y}_n - I| \geq b) \rightarrow 0.$$

Zauważmy ponadto, że warunek $|\hat{Y}_n - I| \leq b$ oznacza, że błąd bezwzględny nie przekracza b . Stąd widzimy, że dla każdego b i liczby $0 < \alpha < 1$ istnieje n_0 (nas interesuje najmniejsze naturalne) takie, że

$$\mathbb{P}(\hat{Y}_n - b < I \leq \hat{Y}_n + b) < 1 - \alpha, \quad n \geq n_0. \quad (1.1)$$

A więc z prawdopodobieństwem większym niż $1 - \alpha$ szukana wielkość I należy do przedziału losowego $[\hat{Y}_n - b, \hat{Y}_n + b]$ i dlatego α nazywa się *poziomem istotności* lub *poziomą ufnością*. Równoważnie możemy pytać się o błąd b dla zadanej liczby replikacji n oraz poziomu ufności α , lub przy zadanym n i b wyznaczyć poziom ufności α . Zgrubną odpowiedź dostaniemy z nierówności Czebyszewa.

Uwaga Niech b_n będzie zadanym błędem przy n replikacjach. Wtedy z nierówności Czebyszewa można oszacować z góry prawdopodobieństwa przekroczenia zadanego błędu b_n :

$$\mathbb{P}(|\hat{Y}_n - I| \geq b_n) \leq \frac{\text{Var } \hat{Y}_n}{b_n^2}$$

Jeśli więc chcemy aby błąd był na poziomie α , to

$$\alpha = \frac{\text{Var } \hat{Y}_n}{b_n^2} = \frac{\sigma_Y^2}{nb_n^2}$$

i stąd rozwiązując równanie mamy

$$b_n = \frac{\sigma_Y / \alpha^{1/2}}{n^{1/2}}$$

czyli $b_n = O(n^{-1/2})$.

Wyprowadzimy teraz bardziej dokładny niż (1.1) wzór na związek pomiędzy n, b, α . Ponieważ w metodach Monte Carlo n jest zwykle duże, możemy skorzystać z *centralnego twierdzenia granicznego* (CTG), które mówi, że jeśli tylko $Y_1, \dots, Y_n$ są niezależne i o jednakowym rozkładzie oraz $\sigma_Y^2 = \text{Var } Y_j < \infty$, to dla $n \rightarrow \infty$

$$\mathbb{P} \left(\sum_{j=1}^n \frac{Y_j - I}{\sigma_Y \sqrt{n}} \leq x \right) \rightarrow \Phi(x), \quad (1.2)$$

gdzie $\sigma_Y = \sqrt{\text{Var } Y_1}$ oraz $\Phi(x)$ jest dystrybuantą standardowego rozkładu normalnego. Niech $z_{1-\alpha/2}$ będzie $\alpha/2$ -kwantylem rozkładu normalnego tj.

$$z_{1-\alpha/2} = \Phi^{-1}(1 - \frac{\alpha}{2}).$$

Stąd mamy, że

$$\lim_{n \rightarrow \infty} \mathbb{P}(-z_{1-\alpha/2} < \sum_{j=1}^n \frac{Y_j - I}{\sigma_Y \sqrt{n}} \leq z_{1-\alpha/2}) = 1 - \alpha.$$

A więc, pamiętając, że $I = \mathbb{E} Y_1$,

$$\mathbb{P}(-z_{1-\alpha/2} < \sum_{j=1}^n \frac{Y_j - I}{\sigma_Y \sqrt{n}} \leq z_{1-\alpha/2}) \approx 1 - \alpha.$$

Przekształcając otrzymujemy równoważną postać

$$\mathbb{P}(\hat{Y}_n - z_{1-\alpha/2} \frac{\sigma_Y}{\sqrt{n}} \leq I \leq \hat{Y}_n + z_{1-\alpha/2} \frac{\sigma_Y}{\sqrt{n}}) \approx 1 - \alpha,$$

skąd mamy *fundamentalny związek* wiążący n, α, σ_Y^2 :

$$b = z_{1-\alpha/2} \frac{\sigma_Y}{\sqrt{n}}. \quad (1.3)$$

Uwaga Zauważmy, że CTG (1.2) można wyrazić następująco:

$$\sqrt{n} \hat{Y}_n \rightarrow_d N(0, \sigma_Y^2).$$

Jeśli nie znamy σ_Y^2 , a to jest raczej typowa sytuacja, to możemy zastąpić tę wartość przez wariancję próbkową

$$\hat{S}_n^2 = \frac{1}{n-1} \sum_{j=1}^n (Y_j - \hat{Y}_n)^2,$$

ponieważ z teorii prawdopodobieństwa wiemy, że jest również prawdą

$$\mathbb{P} \left(\sum_{j=1}^n \frac{Y_j - I}{\hat{S}_n \sqrt{n}} \leq x \right) \rightarrow \Phi(x) . \quad (1.4)$$

Estymator $\hat{S}_n^2$ jest nieobciążony, tj. $\mathbb{E} \hat{S}_n^2 = \sigma_Y^2$ oraz zgodny. Przekształcając (1.4)

$$\mathbb{P}(-z_{1-\alpha/2} < \sum_{j=1}^n \frac{Y_j - I}{\hat{S}_n \sqrt{n}} \leq z_{1-\alpha/2}) \approx 1 - \alpha ,$$

i po dalszych manipulacjach otrzymujemy równoważną postać

$$\mathbb{P}(\hat{Y}_n - z_{1-\alpha/2} \frac{\hat{S}_n}{\sqrt{n}} \leq I \leq \hat{Y}_n + z_{1-\alpha/2} \frac{\hat{S}_n}{\sqrt{n}}) \approx 1 - \alpha .$$

Stąd możemy napisać inną wersję fundamentalnego związku

$$b = z_{1-\alpha/2} \frac{\hat{S}_n}{\sqrt{n}}$$

W teorii Monte Carlo przyjęło rozważać się $\alpha = 0.05$, i z tablic możemy odczytać, że $z_{1-0.025} = 1.96$ i tak będziemy robić w dalszej części wykładu. Jednakże zauważmy, że istnieje możliwość wnioskowania na poziomie ufności, na przykład, 99%. Wtedy odpowiadająca $z_{1-0.005} = 2.58$. Polecamy czytelnikowi obliczenie odpowiednich kwantyli dla innych poziomów istotności wykorzystując tablicę rozkładu normalnego. A więc na poziomie istotności 0.05 fundamentalny związek jest

$$b = \frac{1.96\sigma_Y}{\sqrt{n}} . \quad (1.5)$$

A więc aby osiągnąć dokładność b należy przeprowadzić liczbę $n = 1.96^2 \sigma_Y^2 / b^2$ prób.

Jak już wspomnieliśmy, niestety w praktyce niedysponujemy znajomością σ_Y^2 . Estymujemy więc σ_Y^2 estymatorem $\hat{S}_n^2$. Aby oszacować niezbędną liczbę prób do osiągnięcia zadanej dokładności, można to zrobić najpierw pilotażową symulacją przy użyciu m replikacji w celu wyestymowania $\hat{S}_m^2$, a dopiero po oszacowaniu potrzebnego n z wzoru (1.5) przeprowadzić właściwe

obliczenia z n replikacjami w celu obliczenie I . Wtedy zapisujemy wynik symulacji w postaci przedziału ufności

$$(\hat{Y}_n - \frac{z_{1-\alpha/2}\hat{S}_n}{\sqrt{n}}, \hat{Y}_n + \frac{z_{1-\alpha/2}\hat{S}_n}{\sqrt{n}})$$

lub w formie $(\hat{Y}_n \pm \frac{z_{1-\alpha/2}\hat{S}_n}{\sqrt{n}})$.

W MATLABie średnią i odchylenie standardowe liczymy za pomocą instrukcji `mean(X)` oraz `std(X)`, gdzie X jest wektorem. Jeśli potrzebujemy liczyć odchylenie standardowe samemu to przydatne są następujące wzory rekurencyjne. Podamy teraz wzór rekurencyjny na obliczanie $\hat{Y}_n$ oraz $\hat{S}_n^2$.

Algorytm

$$\hat{Y}_n = \frac{n-1}{n}\hat{Y}_{n-1} + \frac{1}{n}Y_n$$

i

$$\hat{S}_n^2 = \frac{n-2}{n-1}\hat{S}_{n-1}^2 + \frac{1}{n}(Y_n - \hat{Y}_{n-1})^2$$

dla $n = 2, 3, \dots$ i

$$\hat{Y}_1 = Y_1 \quad \hat{S}_1^2 = 0.$$

Przykład 1.1 [Asmussen & Glynn [4]] Przypuśćmy, że chcemy obliczyć $I = \pi = 3.1415\dots$ metodą Monte Carlo. Niech $Y = 4\mathbf{1}(U_1^2 + U_2^2 \leq 1)$, gdzie U_1, U_2 są niezależnymi jednostajnymi $U(0,1)$. Wtedy $\mathbf{1}(U_1^2 + U_2^2 \leq 1)$ jest zmienną Bernoulliego z średnią $\pi/4$ oraz wariancją $\pi/4(1 - \pi/4)$. Tak więc

$$\mathbb{E}Y = \pi$$

oraz wariancja Y równa się

$$\text{Var } Y = 16\pi/4(1 - \pi/4) = 2.6968.$$

Aby więc, przy zadanym standardowym poziomie istotności, dostać jedną cyfrę wiodącą, tj. 3 należy zrobić $n \approx 1.96^2 \cdot 2.7/0.5^2 \approx 42$ replikacji, podczas gdy aby dostać drugą cyfrę wiodącą to musimy zrobić $n \approx 1.96^2 \cdot 2.7/0.05^2 \approx 4130$ replikacji, itd.

Wyobraźmy sobie, że chcemy porównać kilka eksperymentów. Poprzednio nie braliśmy pod uwagę czasu potrzebnego na wygenerowanie jednej replikacji

Y_j . Przyjmijmy, że czas można zdyskretyzować ($n = 1, 2, \dots$), i że dany jest łączny czas m (zwany *budżetem*) którym dysponujemy dla wykonania całego eksperymentu. Niech średnio jedna replikacja trwa czas t . A więc średnio można wykonać w ramach danego budżetu jedynie m/t replikacji. A więc mamy

$$\hat{Y}_n = \frac{1}{n} \sum_{j=1}^n Y_j \approx \frac{1}{m/t} \sum_{j=1}^{m/t} Y_j.$$

Aby porównać ten eksperyment z innym, w którym posługujemy się liczbami losowymi $Y'_1, \dots$, takimi, że $Y = \mathbb{E} Y'$, gdzie na wykonanie jednej replikacji Y'_j potrzebujemy średnio t' jednostek czasu, musimy porównać wariancje

$$\text{Var } \hat{Y}_n, \quad \text{Var } \hat{Y}'_n.$$

Rozpisując mamy odpowiednio $(t \text{Var } Y)/m$ i $(t' \text{Var } Y')/m$. W celu wybrania lepszego estymatora musimy więc porównać powyższe wielkości, i mniejszą z wielkość nazwiemy *efektywnością* estymatora.

1.1 Obliczanie całek metodą Monte Carlo

Przypuśćmy, że szukaną wielkość I możemy zapisać jako

$$I = \int_B k(x) dx,$$

gdzie B jest podbiorem $\mathbb{R}^d$ i $\text{vol}(B) < \infty$. Będziemy dalej zakładać, że $\text{vol}(B) = 1$ (jeśli ten warunek nie jest spełniony, to zawsze możemy zadanie przeskalować). Przykładem może być $B = [0, 1]$ lub $[0, 1]^d$.

Zgrubna metoda Monte Carlo (CMC) Połóżmy $Y = k(V)$, gdzie V ma rozkład jednostajny na B . Wtedy mamy $\mathbb{E} Y = I$. Niech $V_1, \dots, V_n$ będą niezależnymi kopiami V o rozkładzie jednostajnym $U(B)$. Wtedy

$$\hat{Y}^{\text{CMC}} = \hat{Y}_n^{\text{CMC}} = \frac{1}{n} \sum_{j=1}^n k(V_j)$$

jest *zgrubnym estymatorem Monte Carlo* (crude Monte Carlo) i stąd skrót CMC. Będziemy dalej mówili o estymatorach CMC.

Przykład 1.2 [Madras [33]] Niech $k : [0, 1] \rightarrow \mathbb{R}$ będzie funkcją całkowalną z kwadratem $\int_0^1 k^2(x) dx < \infty$. Niech $Y = k(U)$. Wtedy $\mathbb{E}Y = \int_0^1 k(x) dx$ oraz $\mathbb{E}Y^2 = \int_0^1 k^2(x) dx$. Stąd

$$\sigma_Y = \sqrt{\int_0^1 k^2(x) dx - \left(\int_0^1 k(x) dx\right)^2}.$$

W szczególności rozważmy

$$k(x) = \frac{e^x - 1}{e - 1}.$$

Wtedy oczywiście mamy

$$\begin{aligned} \int_0^1 k(x) dx &= \frac{1}{e-1} \left(\int_0^1 e^x dx - 1 \right) = \frac{e-2}{e-1} = 0.418 \\ \int_0^1 k^2(x) dx &= 0.2567. \end{aligned}$$

Stąd $\sigma_Y^2 = 0.0820$ a więc $n = 0.3280/b^2$.

Metoda chybił-trafił. Do estymacji $I = \int_B k(x) dx$ możemy używać innych estymatorów. Załóżmy dodatkowo, że $B = [0, 1]$ i $0 \leq k(x) \leq 1$. Umieścmy teraz krzywą $k(x)$ w kwadracie $[0, 1] \times [0, 1]$ rozbijając kwadrat na dwie części: nad i pod krzywą. Część do której celujemy, rzucając “losowo” punkt (U_1, U_2) na kwadrat, jest pod krzywą. Ten pomysł prowadzi do tzw. estymatora *chybił-trafił* (Hit or Miss) (HoM). Formalnie definiujemy

$$Y = \mathbf{1}(k(U_1) > U_2).$$

Mamy bowiem

$$I = \mathbb{E}Y = \int_0^1 \int_0^1 \mathbf{1}_{k(u_1) > u_2} du_2 du_1 = \int_0^1 k(u_1) du_1.$$

Wtedy dla $U_1, \dots, U_{2n}$ obliczamy replikacje $Y_1, \dots, Y_n$, skąd obliczamy estymator

$$\hat{Y}_n^{\text{HoM}} = \frac{1}{n} \sum_{j=1}^n Y_j.$$

zwanym estymatorem *chybił trafił*. Wracając do przykładu 1.2 możemy łatwo policzyć wariancję $\text{Var } Y = I(1 - I) = 0.2433$. Stąd mamy, że $n = 0.9346/b^2$, czyli dla osiągnięcia tej samej dokładności trzeba około 3 razy więcej replikacji Y -ków aniżeli przy użyciu metody CMC. W praktyce też na wylosowanie Z w metodzie chybił trafił potrzeba dwa razy więcej liczb losowych.

Jak widać celem jest szukanie estymatorów z bardzo małą wariancją.

1.2 Symulacja funkcji od wartości oczekiwanej

Chcemy obliczyć $z = f(I)$, gdy wiadomo jak symulować liczbę losową Y i $I = \mathbb{E} Y$. Zakładamy, że f jest funkcją ciągłą w otoczeniu punktu I . Wtedy naturalnym estymatorem jest

$$\hat{Z}_n = f(\hat{Y}_n) ,$$

gdzie jak zwykle $\hat{Y}_n = (1/n) \sum_{j=1}^n Y_j$. Niestety estymator $\hat{Z}_n$ jest przeważnie obciążony dla $f(I)$. Przy założeniu, że $\sigma^2 = \text{Var } Y < \infty$ mamy

$$\sqrt{n}(\hat{Y}_n - I) \xrightarrow{\mathcal{D}} N(0, \sigma^2) .$$

Zakładając ponadto, że $f(x)$ jest ciągle różniczkowalna w otoczeniu I , ze wzoru Taylora $f(x) - f(I) = f'(I)(x - I) + o(x - I)$, mamy

$$\sqrt{n}(f(\hat{Y}_n) - f(I)) = f'(I)\sqrt{n}(\hat{Y}_n - I) + \sqrt{n} o((\hat{Y}_n - I)).$$

Ponieważ

$$\sqrt{n} o((\hat{Y}_n - I)) = \sqrt{n}(\hat{Y}_n - I) o(1) \xrightarrow{\text{Pr}} 0$$

(detale można znaleźć w książce van der Vaarta [44]), więc

$$\sqrt{n}(f(\hat{Y}_n) - f(I)) \xrightarrow{\mathcal{D}} N(0, f'(I)^2 \sigma^2).$$

Technika dowodowa polegająca na rozwinięciu Taylorowskim funkcji nazywana jest *metodą delty*.

Rozpatrzmy teraz ogólną sytuację gdy estymator $\hat{Y}_n$ wielkości I jest obciążony z *obciążeniem* $\delta_n = \mathbb{E} \hat{Y}_n - I$. Zakładamy, że

B.1 $\sqrt{n}(\hat{Y}_n - \mathbb{E} \hat{Y}_n) \rightarrow_d N(0, \sigma^2)$, gdzie $0 < \sigma^2 < \infty$ gdy $n \rightarrow \infty$,

B.2 $\sqrt{n}\delta_n \rightarrow 0$, gdy $n \rightarrow \infty$.

Wtedy korzystając z własności zbieżności według rozkładu (patrz (XI.1.15)) własności B.1 i B.2 pociągają

$$\sqrt{n}(\hat{Y}_n - I) \xrightarrow{\mathcal{D}} N(0, \sigma^2) .$$

A więc jak poprzednio metodą delty można udowodnić następujący fakt.

Fakt 1.3 *Jesli funkcja f jest ciągle różniczkowalna w otoczeniu I oraz spełnione są założenia B.1-B.2, to*

$$\sqrt{n}(f(\hat{Y}_n) - f(I)) \xrightarrow{\mathcal{D}} N(0, w^2) ,$$

gdzie $w^2 = (f'(I))^2 \sigma^2$.

Teraz heurystycznie pokażemy, że jeśli funkcja f jest dwukrotnie ciągle różniczkowalna w I , oraz

B.2' dla pewnego $0 < \delta < \infty$

$$n\delta_n \rightarrow \delta$$

(zauważmy, że B.2' jest mocniejszy niż B.2), to

$$\mathbb{E}(f(\hat{Y}_n) - f(I)) \sim \frac{1}{n} \left(\delta f'(I) + \frac{\sigma^2 f''(I)}{2} \right) .$$

Mianowicie z twierdzenia Taylora mamy

$$f(\hat{Y}_n) - f(I) = f'(I)(\hat{Y}_n - I) + \frac{f''(I)}{2}(\hat{Y}_n - I)^2 + o((\hat{Y}_n - I)^2) . \quad (1.6)$$

i dalej mnożymy (1.6) przez n oraz korzystamy z tego, że

- $nf'(I)(\mathbb{E} \hat{Y}_n - I) \rightarrow \delta f'(I)$,
- $n \frac{f''(I)}{2} \mathbb{E}(\hat{Y}_n - I)^2 \rightarrow \frac{f''(I)}{2} \sigma^2$,
- $n \mathbb{E}[o((\hat{Y}_n - I)^2)] \rightarrow 0$.

Niestety tym razem aby przeprowadzić formalny dowód musielibyśmy skorzystać z własności jednostajnej całkowalności co jest raczej niemożliwe bez dodatkowych założeń.

Omówimy teraz rozszerzenie faktu 1.3 na przypadek wielowymiarowy. Mianowicie chcemy wyestymować $z = f(I^1, \dots, I^d)$ jeśli wiemy, że $(I^1, \dots, I^d) = \mathbb{E}(\hat{Y}^1, \dots, \mathbb{E}\hat{Y}^d)$ oraz potrafimy symulować $(\hat{Y}^1, \dots, \hat{Y}^d)$. Wygodnie jest wprowadzić oznaczenia wektorowe

$$\hat{\mathbf{Y}} = (\hat{Y}^1, \dots, \hat{Y}^d), \quad \mathbf{I} = (I^1, \dots, I^d).$$

Przez $(\nabla f)(\mathbf{I})$ oznaczamy wektor $(f^{(1)}(\mathbf{I}), \dots, f^{(d)}(\mathbf{I}))$, gdzie $f^{(i)}(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial x_i}$. Dowód następującego faktu przy użyciu metody delty można znaleźć w książce van der Vaart [44].

Fakt 1.4 *Jeśli funkcja $f : \mathbb{R}^d \rightarrow \mathbb{R}$ jest ciągle różniczkowalna w otoczeniu punktu $\mathbf{I}$ oraz*

$$\sqrt{n}(\hat{\mathbf{Y}}_n - \mathbf{I}) \xrightarrow{\mathcal{D}} N(\mathbf{0}, \Sigma)$$

dla pewnej skończonej macierzy kowariancji Σ to

$$\sqrt{n} \left(f(\hat{\mathbf{Y}}_n) - f(\mathbf{I}) \right) \xrightarrow{\mathcal{D}} N(\mathbf{0}, (\nabla f)(\mathbf{I})\Sigma(\nabla f)(\mathbf{I})^T).$$

W szczególności więc jeśli $\mathbf{Y}$ ma skończoną macierz kowariancji $\Sigma = (\sigma_{ij})_{i,j=1,\dots,d}$ oraz $\hat{\mathbf{Y}}_n = (1/n) \sum_{j=1}^n \mathbf{Y}_j$, gdzie $\mathbf{Y}_1, \dots, \mathbf{Y}_n$ są niezależnymi replikacjami $\mathbf{Y}$ to dla estymatora $\hat{Z}_n = f(\hat{\mathbf{Y}}_n)$ przedział istotności na poziomie $\alpha = 0.05$ jest $f(\hat{\mathbf{Y}}_n) \pm 1.96\sigma/\sqrt{n}$, gdzie

$$\sigma^2 = (\nabla f)(\mathbf{I})\Sigma((\nabla f)(\mathbf{I}))^T = \sum_{i,j=1}^d f^{(i)}(\mathbf{I})f^{(j)}(\mathbf{I})\sigma_{ij}. \quad (1.7)$$

Niestety wzór na wariancję zawiera $\mathbf{I}$, którą to wielkość chcemy symulować. Jeśli więc Σ jest znana, to proponuje się przedziały ufności w postaci $f(\hat{\mathbf{Y}}_n) \pm 1.96\hat{\sigma}/\sqrt{n}$, gdzie

$$\hat{\sigma}^2 = \nabla f(\mathbf{I})\Sigma((\nabla f)(\hat{\mathbf{Y}}))^T = \sum_{i,j}^d f^{(i)}(\hat{\mathbf{Y}}_n)f^{(j)}(\hat{\mathbf{Y}}_n)\sigma_{ij}. \quad (1.8)$$

Estymator macierzy kowariancji Σ Naturalnym estymatorem wektora średniej $\mathbf{I}$ i macierzy kowariancji Σ wektora losowego $\mathbf{Y}$, gdy $\mathbf{Y}_1, \dots, \mathbf{Y}_n$ są niezależnymi replikacjami $\mathbf{Y}$ jest

$$\begin{aligned}\hat{\mathbf{Y}}_n &= \frac{1}{n} \sum_{j=1}^n \mathbf{Y}_j, \\ \hat{\sigma}_{i,j} &= \frac{1}{n-1} \sum_{m=1}^n (Y^i - \hat{Y}_n^i)(Y^j - \hat{Y}_n^j).\end{aligned}$$

Natomiast estymatorem σ^2 z (1.7) jest

$$\hat{\sigma}^2 = \frac{1}{n-1} \sum_{m=1}^n \nabla f(\hat{\mathbf{Y}}_n)(\mathbf{Y}_m - \hat{\mathbf{Y}}_n)(\mathbf{Y}_m - \hat{\mathbf{Y}}_n)(\nabla f(\hat{\mathbf{Y}}_n))^T.$$

Zadania teoretyczne

2

1.1 Zaproponować algorytm obliczenia całki

$$\int_{-\infty}^{\infty} \psi(x) f(x) dx$$

(zakładamy, że jest ona absolutnie zbieżna), gdzie

$$f(x) = \begin{cases} e^{2x}, & -\infty < x < 0 \\ e^{-2x}, & 0 < x < \infty. \end{cases}$$

1.2 Przypuśćmy, że chcemy obliczyć I metodą Monte Carlo, gdzie $I = I' I''$ oraz wiemy, że $I' = \mathbb{E} Y'$, $I'' = \mathbb{E} Y''$. Niech $Y'_1, \dots, Y'_R$ będzie R niezależnych replikacji Y' oraz niezależnie od nich niech $Y''_1, \dots, Y''_R$ będzie R niezależnych replikacji Y'' . Pokazać, że następujące estymatory są nieobciążone dla I :

$$\hat{Z}_1 = \left(\frac{1}{R} \sum_{i=1}^R Y'_i \right) \left(\frac{1}{R} \sum_{i=1}^R Y''_i \right)$$

$$\hat{Z}_2 = \frac{1}{R} \sum_{i=1}^R Y'_i Y''_i.$$

Pokazać, że $\hat{Z}_1$ ma mniejszą wariancję. Wsk. Pokazać, że dla niezależnych zmiennych losowych X, Y mających drugie momenty mamy

$$\text{Var}(XY) = \text{Var} X \text{Var} Y + (\mathbb{E} X)^2 \text{Var} Y + (\mathbb{E} Y)^2 \text{Var} X.$$

1.3 Zanalizować przykład II.3.6 pod kątem odpowiedniości liczby replikacji. Jak dobrać δ aby poza $k = 27$ na wykresie II.3.2 trudno byłoby rozróżnić krzywą wysumulowaną od teoretycznej. Czy lepiej byłoby przyjąć $\delta = 0.05$.

- 1.4 [Igła Buffona a liczba π] W zadaniu Buffona o liczbie π oblicza się prawdopodobieństwo

$$p = \frac{2L}{\pi}$$

przecięcia jednej z równoległych prostych oddalonych od siebie o 1 przez igłę o długości L . Wiadomo, że przeprowadzając niezależne eksperymenty mamy nieobciążony estymator $\hat{p}$ prawdopodobieństwa p . Wtedy do obliczenia wykorzystujemy wzór $\pi = (2L)/p$, skąd mamy estymator $\hat{\pi} = (2L)/\hat{p}$. Zaplanować obliczenie $f(I)$ z dokładnością $b = 0.01$ na poziomie $\alpha = 0.05$, gdy $f(x) = (2L)/x$ oraz $I = p = \mathbb{E}Y$, gdzie $Y = 1$, gdy igła przecina linię, natomiast $Y = 0$ w przeciwnym razie.

- 1.5 Ile należy zrobić replikacji na poziomie ufności $\alpha = 0.05$ aby błąd $b = \sigma/50$.
- 1.6 Wiadomo, że jeśli $Y_1, \dots, Y_n$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $N(0,1)$ to

$$\frac{(n-1)\hat{S}^2/\sigma^2 - n + 1}{\sqrt{2n-2}} \xrightarrow{\mathcal{D}} N(0,1), \quad (1.9)$$

gdy $n \rightarrow \infty$; patrz [44], str. 27. Natomiast jeśli zmienne są o rozkładzie dowolnym, to

$$\sqrt{n} \left(\hat{S}_n^2 - \sigma^2 \right) \xrightarrow{\mathcal{D}} N(0, \mu_4 - \sigma^4), \quad (1.10)$$

gdzie $\mu_4 = \mathbb{E}(Y - \mathbb{E}Y)^4$ jest momentem centralnym rzędu 4. Ile należy zrobić prób aby wyestymować σ^2 z dokładnością do 5-ciu % na poziomie $\alpha = 0.05$. Rozważyć oba warianty przy użyciu (1.9), (1.10).

Zadania laboratoryjne

- 1.7 a. Napisać algorytm na wygenerowanie punktu losowego o rozkładzie jednostajnym $U(\Delta)$ w trójkącie Δ o wierzchołkach $(1, 1), (-1, -1), (1, -1)$. Policzć średnią liczbę liczb losowych $U_1, U_2, \dots$ potrzebnych do wygenerowania jednej liczby V o rozkładzie $U(\Delta)$.
- b. Napisać algorytm na obliczenia całki

$$\int_{\Delta} e^{-(x^2+y^2)} dx dy$$

zgrubną metodą MC.

- 1.8 Wyprodukować 10 000 przedziałów ufności postaci $(\hat{Y}_n \pm \frac{z_{1-\alpha/2}\hat{S}_n}{\sqrt{n}})$ dla $Y = \mathbf{1}(U \leq u)$, $n = 1000$ oraz $I = \mathbb{E}Y$ następnie sprawdzić jaki procent tych przedziałów pokryje $I = 0.5, 0.9, 0.95$ gdy odpowiednio $u = 0.5, 0.9, 0.95$. Wyjaśnić otrzymane wyniki.

Projekt

- 1.9 Obliczyć składkę netto z dokładnością do czterech miejsc po przecinku, dla 40-latka w ubezpieczeniu na całe życie, gdy techniczna stopa procentowa wynosi $r = 0.05$, przyszły czas życia $T = T_{40}$ ma rozkład Makehama (można użyć funkcji Makeham z zadania III.9.25). Wzór na składkę netto gdy suma ubezpieczenia wynosi 1 zł jest

$$\pi = \frac{\mathbb{E} e^{-rT}}{\mathbb{E} \int_0^T e^{-rt} dt} .$$

Przeprowadzić analizę błędu.

Rozdział V

Techniki redukcji wariancji

¹ W tym rozdziale zajmiemy się konstrukcją i analizą metod prowadzących do redukcji estymatora wariancji. Jak zauważyliśmy w poprzednim rozdziale, metoda MC polega na tym, że aby obliczyć powiedzmy I , znajdujemy zmienną losową Y taką, że $I = \mathbb{E}Y$. Wtedy $\hat{Y} = \sum_{j=1}^n Y_j/n$ jest estymatorem MC dla I , gdzie $Y_1, \dots$ są niezależnymi replikacjami Y . Wariancja $\text{Var } \hat{Y} = \text{Var } Y/n$, a więc widzimy, że musimy szukać takiej zmiennej losowej Y dla której $\mathbb{E}Y = I$, i mającej możliwie małą wariancję. Ponadto musimy wziąć pod uwagę możliwość łatwego generowania replikacji Y . Zauważmy też, że czasami pozbycie się założenia, że $Y_1, \dots$ są niezależnymi replikacjami Y może też być korzystne.

W tym rozdziale zajmiemy się następującymi technikami:

1. metoda warstw,
2. metoda zmiennych antytetycznych,
3. metoda wspólnych liczb losowych,
4. metoda zmiennych kontrolnych,
5. warunkowa metoda MC,
6. metoda losowania istotnościowego.

¹Poprawiane: 17.1.2018

1 Metoda warstw

Jak zwykle chcemy obliczyć $I = \mathbb{E}Y$. Przyjmijmy, że Y jest o wartościach w E (może być $\mathbb{R}$, $\mathbb{R}^d$ lub podzbiór) i niech $A^1, \dots, A^m$ będą rozłącznymi zbiorami takimi, że $\mathbb{P}(Y \in \bigcup_{j=1}^m A^j) = 1$. Rozbicie definiuje nam *warstwy* $W^j = \{Y \in A^j\}$. Możemy się spodziewać, że przy odpowiednim doborze warstw i zaprojektowaniu liczby replikacji w warstwach będzie możliwa redukcja wariancji.

Oznaczmy

- prawdopodobieństwo wylosowania warstwy $p_j = \mathbb{P}(Y \in A^j)$ (to przyjmujemy za znane i dodatnie)
- oraz $I^j = \mathbb{E}[Y|Y \in A^j]$ ($j = 1, \dots, m$) (oczywiście to jest nieznane bo inaczej nie trzeba by nic obliczać).

Ze wzoru na prawdopodobieństwo całkowite

$$\mathbb{E}Y = p_1 I^1 + \dots + p_m I^m.$$

Metoda polega na losowaniu w poszczególnych warstwach, korzystając z tego, że potrafimy losować (o tym później)

$$Y^j =_d (Y|Y \in A_j).$$

Niech teraz n będzie ogólną liczbą replikacji, którą podzielimy na odpowiednio n_j replikacji Y^j w warstwie j . Niech $Y_1^j, \dots, Y_{n_j}^j$ będą więc niezależnymi replikacjami Y^j w warstwie j . Zakładamy również niezależność replikacji pomiędzy poszczególnymi warstwami. Oczywiście $n = n_1 + \dots + n_m$. Definiujemy

$$\hat{Y}_{n_j}^j = \frac{1}{n_j} \sum_{i=1}^{n_j} Y_i^j$$

jako estymator I^j . Niech $\sigma_j^2 = \text{Var} Y^j$ będzie wariancją w warstwie j . Zauważmy, że $\hat{Y}^j$ jest estymatorem nieobciążonym I^j oraz

$$\text{Var} \hat{Y}_{n_j}^j = \frac{\sigma_j^2}{n_j}.$$

Teraz definiujemy *estymator warstwowy*

$$\hat{Y}_n^{\text{str}} = p_1 \hat{Y}_{n_1}^1 + \dots + p_m \hat{Y}_{n_m}^m.$$

Zakładamy, że wiadomo jak generować poszczególne Y^j . Zauważmy, że estymator warstwowy jest nieobciążony

$$\begin{aligned}\mathbb{E} \hat{Y}^{\text{str}} &= p_1 \mathbb{E} \hat{Y}^1 + \dots + p_m \mathbb{E} \hat{Y}^m \\ &= \sum_{i=1}^m p_i \frac{\mathbb{E} Y \mathbf{1}_{\{Y \in A_i\}}}{p_i} = \mathbb{E} Y = I.\end{aligned}$$

Ponadto jego wariancja jest

$$\text{Var} \hat{Y}_n^{\text{str}} = \sum_{j=1}^m \frac{p_j^2}{n_j} \sigma_j^2.$$

Niech dla $\mathbf{x} = (x_1, \dots, x_m)$ (u nas $x_j \geq 0$ i $\sum_{j=1}^m x_j = 1$)

$$\sigma_{\text{str}}^2(\mathbf{x}) = \sum_{j=1}^m \frac{p_j}{x_j} \sigma_j^2.$$

Pamiętając o niezależności poszczególnych replikacji mamy

$$\begin{aligned}n \text{Var} \hat{Y}_n^{\text{str}} &= n \sum_{j=1}^m p_j^2 \text{Var} \hat{Y}^j = p_1^2 \frac{n}{n_1} \sigma_1^2 + \dots + p_m^2 \frac{n}{n_m} \sigma_m^2 \\ &= \frac{p_1^2}{n_1/n} \sigma_1^2 + \dots + \frac{p_m^2}{n_m/n} \sigma_m^2\end{aligned}\tag{1.1}$$

$$\sigma_{\text{str}}^2((n_1/n, \dots, n_m/n)) .\tag{1.2}$$

Zastanowimy się co się stanie gdy przechodzimy z $n \rightarrow \infty$ tak aby

(*) $q_j(n) = n_j/n$ były zbieżne, powiedzmy do q_j ($j = 1, \dots, m$).

Oczywiście $\mathbf{q} = (q_1, \dots, q_m)$ jest funkcją prawdopodobieństwa. Wtedy

$$\sigma_{\text{str}}^2((n_1/n, \dots, n_m/n)) \rightarrow \sigma^2(\mathbf{q}) .$$

Kluczowym dla analizy wyników jest następujący fakt.

Fakt 1.1 *Jeśli z n i $n_1, \dots, n_m$ przechodzimy do nieskończoności w sposób (*), to*

$$\frac{\sqrt{n}}{\sigma_{\text{str}}(\mathbf{q})} (\hat{Y}_n^{\text{str}} - I) \xrightarrow{\mathcal{D}} N(0, 1) .$$

Dowód Musimy pamiętać, że wwarstwach replikacje są niezależne o jednakowym rozkładzie i o niezależności między warstwami. Mamy

$$\begin{aligned}
 \sqrt{n}(\sum_{j=1}^m p_j \hat{Y}^j - I) &= \sqrt{n}(\sum_{j=1}^m p_j \hat{Y}^j - \sum_{j=1}^m p_j I^j) \\
 &= \sum_{j=1}^m p_j \sigma_j \sqrt{\frac{n}{n_j}} \left[\frac{\sqrt{n_j}}{\sigma_j} (\hat{Y}^j - I^j) \right] \\
 &= \sum_{j=1}^m \frac{p_j \sigma_j}{\sqrt{q_j(n)}} \left[\frac{\sqrt{n_j}}{\sigma_j} (\hat{Y}^j - I^j) \right] \\
 &\xrightarrow{\mathcal{D}} N(0, \sum_{j=1}^m p_j^2 \frac{\sigma_j^2}{q_j}),
 \end{aligned}$$

ponieważ

$$\frac{\sqrt{n_j}}{\sigma_j} (\hat{Y}^j - I^j) \xrightarrow{\mathcal{D}} N_j$$

gdzie $N_1, \dots, N_m$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $N(0,1)$. $\square$

Dobór liczby replikacji w warstwach Możemy teraz postawić dwa pytania.

- Mając zadane warstwy (a więc znając (p_j) oraz (σ_j^2)) jak dobrać dla zadanej całkowitej liczby replikacji n , liczby replikacji (n_j) na poszczególnych warstwach aby wariancja $n \text{Var } \hat{Y}_n^{\text{str}}$ była minimalna.
- Jak dobrać warstwy aby zmniejszyć (zminimizować) wariancję.

Zajmiemy się odpowiedzią na pierwsze pytanie. Okazuje się, że optymalny dobór liczby replikacji n_j można wyliczyć z następującego twierdzenia.

Twierdzenie 1.2 *Niech n będzie ustalone i $n_1 + \dots + n_m = n$. Wariancja $p_1^2 \frac{n}{n_1} \sigma_1^2 + \dots + p_m^2 \frac{n}{n_m} \sigma_m^2$ estymator warstwowego $\hat{Y}^{\text{str}}$ jest minimalna gdy*

$$n_j = \frac{p_j \sigma_j}{\sum_{j=1}^m p_j \sigma_j} n.$$

Dowód Musimy zminimizować $\sigma_{\text{str}}^2(\mathbf{x})$ przy warunku $x_1 + \dots + x_m = 1$, $x_j \geq 0$. Jeśli podstawimy do wzoru (1.2)

$$q_j^* = \frac{p_j \sigma_j}{D}$$

gdzie $D = \sum_{j=1}^m p_j \sigma_j$ to $\sigma_{\text{spr}}^2(\mathbf{q}^*) = D^2$. Ale z nierówności Cauchy'ego-Schwarza

$$\left(\sum_{j=1}^n x_j z_j \right)^2 \leq \sum_{j=1}^n x_j^2 \sum_{j=1}^n z_j^2,$$

mamy

$$\begin{aligned} D^2 &= \left(\sum_{j=1}^m p_j \sigma_j \right)^2 = \left(\sum_{j=1}^m \sqrt{x_j} \frac{p_j \sigma_j}{\sqrt{x_j}} \right)^2 \\ &\leq \sum_{j=1}^m x_j \sum_{j=1}^m \frac{p_j^2 \sigma_j^2}{x_j} \\ &= \sigma_{\text{str}}^2(\mathbf{x}). \end{aligned}$$

□

Niestety ten wzór jest mało użyteczny ponieważ nie znamy zazwyczaj wariancji σ_j^2 , $j = 1, \dots, m$. Chyba, że jak poprzednio przyjmiemy empiryczną wariancję $\hat{S}_j^2$ parametru σ_j^2 w warstwie j . Mamy za to następujące fakt, że tzw. *alokacja proporcjonalna* zmniejsza wariancję. Mówimy o takiej alokacji, jeśli $n_i = np_i$ $i = 1, \dots, m$. Wtedy z wzoru ([?e.war:str?]) mamy

$$\sigma_{\text{str}}^2(n_1/n, \dots, n_m/n) = \sum_{j=1}^m p_j \sigma_j^2. \quad (1.3)$$

W przypadku alokacji proporcjonalnej mamy $\mathbf{p} = (n_1/n, \dots, n_m/n)$.

Fakt 1.3 *Jesli alokacja jest proporcjonalna, to*

$$\sigma_{\text{str}}^2(\mathbf{p}) \leq \sigma_Y^2.$$

Dowód Z wzoru na prawdopodobieństwo całkowite

$$\mathbb{E} Y^2 = \sum_{j=1}^m p_j \mathbb{E} [Y^2 | Y \in A^j] = \sum_{j=1}^m p_j (\sigma_j^2 + (I^j)^2).$$

Ponieważ $I = \sum_{j=1}^m p_j I^j$ więc

$$\begin{aligned} \text{Var } Y &= \sum_{j=1}^m p_j (\sigma_j^2 + (I^j)^2) - \left(\sum_{j=1}^m p_j I^j \right)^2 \\ &= \sum_{j=1}^m p_j \sigma_j^2 + \sum_{j=1}^m p_j (I^j)^2 - \left(\sum_{j=1}^m p_j I^j \right)^2. \end{aligned}$$

Z nierówności Jensena

$$\left(\sum_{j=1}^m p_j I^j \right)^2 \leq \sum_{j=1}^m p_j (I^j)^2,$$

skąd

$$\text{Var } Y \geq \sum_{j=1}^m p_j \sigma_j^2 = \sigma^2(\mathbf{p}).$$

□

A więc przy alokacji proporcjonalnej bierzemy $n_j = \lfloor np_j \rfloor$, i wtedy wariancja się zmniejsza.

Modyfikacje Zamiast jednej zmiennej Y możemy rozważyć parę Y, X i warstwy zdefiniować następująco. Niech $A^1, \dots, A^m$ będą rozłącznymi zbiorami takimi, że $\mathbb{P}(X \in \bigcup_{j=1}^m A^j) = 1$. Wtedy j -ta warstwa jest zdefiniowana przez $W^j = \{X \in A^j\}$ i

$$I = \mathbb{E} Y = \sum_{j=1}^m p_j \mathbb{E}[Y | X \in A^j],$$

gdzie $p_j = \mathbb{P}(X \in A_j)$. Aby napisać wzór na estymator warstwowy musimy przededefiniować pojęcie

$$Y^j =_{\text{d}} (Y | X \in A^j).$$

Zostawiamy czytelnikowi napisanie wzoru na estymator warstwowy w tym przypadku. Zauważmy, że w przypadku ogólnym warstwy W^j ($j = 1, \dots, m$) tworzą partycję przestrzeni zdarzeń elementarnych Ω . Najogólniej mówiąc przez warstwy rozumiemy rozbitcie Ω na zdarzenia $W^1, \dots, W^m$.

Przykłady

Przykład 1.4 [Asmussen & Glynn [4]]. Rozważmy problem obliczenia całki

$$I = \int_0^1 g(x) dx. \quad (1.4)$$

Niech $Y = g(U)$ oraz $X = U$. Wtedy możemy wprowadzić warstwy następująco: $W^j = \{U \in (\frac{j-1}{m}, \frac{j}{m}]\}$, $j = 1, \dots, n$. W tym przypadku bardzo prosto można przeprowadzić losowanie $Y^j =_d (Y|X \in (j-1/m, j/m])$. Mianowicie zmienna

$$V^j = \frac{j-1}{m} + \frac{U}{m}$$

ma rozkład jednostajny na $(\frac{j-1}{m}, \frac{j}{m}]$. Ponadto mamy $p_j = 1/m$. Gdybyśmy zastosowali alokację proporcjonalną z $m = n$ (wtedy mamy $n_j = 1$), to estymatorem warstwowym I jest

$$\hat{Y}_n^{\text{str}} = \frac{1}{n} \sum_{j=1}^n g(V^j).$$

Porównajmy z całkowaniem numerycznym według wzoru:

$$\hat{g}_n = \frac{1}{n} \sum_{j=1}^n g\left(\frac{2j-1}{2n}\right).$$

Zauważmy, że $\mathbb{E} V^j = \frac{2j-1}{2n}$ jest środkiem przedziału $(\frac{j-1}{n}, \frac{j}{n})$.

Zobaczmy teraz jak estymator warstwowy zmniejsza wariancję przy obliczeniu całki (1.4), gdzie g jest funkcją ciągle różniczkowalną z

$$M = \sup_{x \in [0,1]} |g'(x)| < \infty.$$

Zauważmy, że

$$\begin{aligned} Y^j &= _d \left(g(U) \middle| \frac{j-1}{n} \leq U < \frac{j}{n} \right) \\ &= _d g\left(\frac{j-1}{n} + \frac{U}{n}\right). \end{aligned}$$

Wariancja naszego estymatora warstwowego wynosi

$$\sigma_{\text{str}}^2(1/n, \dots, 1/n) = \frac{1}{n} \sum_{j=1}^n \sigma_j^2,$$

gdzie

$$\sigma_j^2 = \text{Var} \left(g\left(\frac{j-1}{n} + \frac{1}{n}U\right) \right).$$

Korzystając z twierdzenia o wartości średniej

$$g\left(\frac{j-1}{n} + \frac{U}{n}\right) = g\left(\frac{j-1}{n}\right) + g'(\theta_j)\frac{U}{n}$$

gdzie θ_j jest punktem z odcinka $((j-1)/n, (j-1+U)/n)$. mamy

$$\sigma_j^2 = \text{Var} \left(g\left(\frac{j-1}{n} + \frac{U}{n}\right) \right) = \frac{\text{Var}(g'(\theta_j)U)}{n^2}.$$

Teraz

$$\sigma_j^2 \leq \frac{\mathbb{E}(g'(\theta_j)^2 U^2)}{n^2} \leq \frac{M^2}{n^2}.$$

A więc wariancja estymatora warstwowego z alokacją proporcjonalną jest rzędu

$$\sigma_{\text{str}}^2 = \sum_{j=1}^n p_j \sigma_j^2 \leq \sum_{j=1}^n \frac{1}{n} \frac{M^2}{n^2} = \frac{M^2}{n^2} = O(n^{-2}).$$

Korzystając z rozważań z Uwagi IV.1 możemy je przenieść na nasz przypadek. Mamy bowiem

$$\mathbb{P}(|\hat{Y}_n^{\text{str}} - I| \geq b_n) \leq \frac{\text{Var} \hat{Y}_n^{\text{str}}}{b_n^2} \leq \frac{O(n^{-2})}{b_n^2}.$$

Z równania $\frac{O(n^{-2})}{b_n^2} = \alpha$ mamy, że $b_n = O(n^{-1})$, co jest lepszą szybkością zbieżności niż kanoniczna $O(n^{-1/2})$.

Przykład 1.5 [Asmussen & Glynn [4]]. Znajdziemy teraz estymator dla $I = \int_0^1 g(x) dx$, który ma zbieżność szybszą niż kanoniczna $n^{-1/2}$, gdzie n jest liczbą replikacji. To nie będzie estymator wartwowy ale wykorzystamy pomysły z poprzedniego przykładu 1.4. Zakładamy, że g jest funkcją ciągle różniczkowalną z

$$M = \sup_{x \in [0,1]} g'(x) < \infty.$$

Rozważmy więc estymator postaci

$$\hat{Y}_n = \frac{1}{n} \sum_{j=1}^n g\left(\frac{j-1+U_j}{n}\right).$$

Jest on nieobciążony bo

$$\begin{aligned}\mathbb{E} \hat{Y}_n &= \frac{1}{n} \sum_{j=1}^n \int_0^1 g\left(\frac{j-1+u}{n}\right) du \\ &= \sum_{j=1}^n \int_{(j-1)/n}^{j/n} g(u) du = \int_0^1 g(u) du.\end{aligned}$$

Teraz zanalizujemy wariancję. Z twierdzenia o wartości średniej

$$g\left(\frac{j-1}{n} + \frac{U_j}{n}\right) = g\left(\frac{j-1}{n}\right) + g'(\theta_j) \frac{U_j}{n},$$

gdzie $\theta_j \in (\frac{j-1}{n}, \frac{j}{n})$. A więc

$$\begin{aligned}\text{Var } Y &= \frac{1}{n} \sum_{j=1}^n \text{Var } g\left(\frac{j-1+U_j}{n}\right) \\ &= \frac{1}{n^2} \sum_{j=1}^n \left(\frac{g'(\theta_j)U_j}{n}\right)^2 \leq \frac{M^2}{n^3}.\end{aligned}$$

W tym przypadku błąd $b_n = O(1/n^{3/2})$.

Przykład 1.6 Losowanie warstwowe dla dowolnych rozkładów Teraz omówimy metody symulacji zmiennej o zadanym rozkładzie, używające koncepcji wartw. Przypomnijmy, że

$$V^i = \frac{i-1}{m} + \frac{U_i}{m}, \quad i = 1, \dots, m$$

ma rozkład jednostajny w $(i-1)/m, i/m]$. Jeśli więc położymy $n = m$ to wtedy $n_j = 1$ i $V^1, \dots, V^n$ tworzy warstwową próbę z rozkładu jednostajnego.

Rozważymy teraz symulację zmiennej losowej Y z dystrybuantą $F(x)$ metodą warstw. Dla ułatwienia założymy, że F jest ściśle rosnąca. Rozważania przeprowadzamy na prostej $\mathbb{R}$, ale prosta modyfikacja pozwala rozważać rozkłady które są skoncentrowane na podzbiorze $\mathbb{R}$. Niech $p_1, \dots, p_m$ będą zadanymi rozmiarami warstw, gdzie $p_j > 0$ oraz $\sum_{j=1}^m p_j = 1$. Prostą $\mathbb{R}$ rozbijamy na m odcinków:

$$A^1 = (a_0, a_1], A^2 = (a_1, a_2], \dots, A^m = (a_{m-1}, a_m)$$

, gdzie $a_0 = -\infty$ oraz $a_m = \infty$, tak aby $\mathbb{P}(Y \in (a_{j-1}, a_j]) = p_j$ (przyjmujemy, że rozkład jest skoncentrowany na całej prostej). Mianowicie a_j -ty wyznaczamy następująco:

$$\begin{aligned} a_0 &= -\infty \\ a_1 &= F^{-1}(p_1) \\ &\vdots \\ a_m &= F^{-1}(p_1 + \dots + p_m) = F^{-1}(1). \end{aligned}$$

W przypadku gdy nośnik E rozkładu F jest właściwym podzbiorem $\mathbb{R}$ to jak już zauważyliśmy trzeba rozbić zmodyfikować.

Teraz $Y^j =_d (Y|Y \in A_j)$ możemy symulować następująco, jeśli mamy procedurę obliczania odwrotnej dystrybuanty $F^{-1}(u)$ (może to być też uogólniona odwrotna). Mamy $p_j = \mathbb{P}(A^j)$

Fakt 1.7 *Niech $V^j = a_{j-1} + (a_j - a_{j-1})U$. Wtedy*

$$F^{-1}(V^j) =_d (Y|Y \in A^j).$$

W książce Glassermana jest opisany następujące eksperymenty. W pierwszym eksperymencie przeprowadzono 500 replikacji z rozkładu normalnego i zrobiono histogram z 25 klasami. Natomiast drugi eksperyment przeprowadzono przy losowaniu w 100 jednakowo prawdopodobnych warstwach losując w każdej 5 replikacji. Wyniki są przedstawione na rys. [????].

2 Zależności zmniejszają wariancję

2.1 Zmienne antytetyczne

Rozpatrzmy teraz metody symulacji $I = \mathbb{E}Y$, w których replikacje nie muszą być niezależne. Będzie nam potrzebna następująca nierówność, która jest natychmiastowym wnioskiem z twierdzenia XI.1.2.

Wniosek 2.1 *Jesli funkcja $\psi(x_1, \dots, x_k)$, $0 \leq x_1, \dots, x_k \leq 1$ jest monotoniczna (niemalejąca lub nierosnąca), to dla niezależnych zmiennych $U_1, \dots, U_k$ o rozkładzie jednostajnym*

$$\text{Cov}(\psi(U_1, \dots, U_k), \psi(1 - U_1, \dots, 1 - U_k)) \leq 0.$$

Mówimy wtedy, że zmienne $\psi(U_1, \dots, U_k), \psi(1 - U_1, \dots, 1 - U_k)$ są ujemnie skorelowane. Zauważmy ponadto, że U i $1 - U$ mają ten sam rozkład jednostajny $U(0, 1)$. Rozpatrzmy teraz n zmiennych losowych $Y_1, Y_2, \dots, Y_n$, przy czym n jest parzyste. Ponadto zakładamy, że

- pary $(Y_{2i-1}, Y_{2i})_i, i = 1, \dots, n/2$ są niezależne o jednakowym rozkładzie,
- zmienne losowe $(Y_j)_{j=1}^n$ mają ten sam rozkład brzegowy co Y .

Jeśli rozważamy estymator CMC

$$\hat{Y}^{\text{CMC}} = \frac{1}{n} \sum_{j=1}^n Y'_j$$

gdzie $Y'_1, Y'_2, \dots$ są niezależnymi replikacjami Y , to wariancja wynosi $\text{Var } Y/n$. Zauważmy, że jeśli będziemy rozpatrywać estymator

$$\hat{Y}_n^* = \frac{1}{n} \sum_{j=1}^n Y_j$$

to jego wariancja

$$\begin{aligned} \text{Var}(\hat{Y}_n^*) &= \frac{\frac{n}{2} \text{Var}(Y_1 + Y_2)}{n^2} \\ &= \frac{1}{2n} (2\text{Var}(Y) + 2\text{cov}(Y_1, Y_2)) \\ &= \frac{1}{2n} (2\text{Var}(Y) + 2\text{Var}(Y_1)\text{Corr}(Y_1, Y_2)) \\ &= \frac{1}{n} \text{Var}(Y)(1 + \text{Corr}(Y_1, Y_2)) . \end{aligned}$$

A więc jeśli korelacja $\text{Corr}(Y_1, Y_2)$ jest ujemna to redukujemy wariancję.

Przypuśćmy teraz, że dla pewnej funkcji $\psi : [0, 1]^k \rightarrow \mathbb{R}$ mamy $Y_1 = \psi(U_1, \dots, U_k)$, i że ψ spełnia założenie z wniosku 2.1. Wtedy $Y_2 = \psi(1 - U_1, \dots, 1 - U_k)$ i zmienne Y_1, Y_2 są ujemnie skorelowane. Dalej definiujemy $Y_3 = \psi(U_{k+1}, \dots, U_{2k}), Y_4 = \psi(1 - U_{k+1}, \dots, 1 - U_{2k})$ itd. Zauważmy, że tak zdefiniowany ciąg $Y_1, \dots, Y_n$ spełnia nasze postulaty. Wtedy

$$\hat{Y}^{\text{anth}} = \frac{\sum_{j=1}^n Y_j}{n}$$

nazywamy *estymatorem antytetycznym* i jeśli $\text{corr}(Y_1, Y_2) < 0$, to z przedstawionych rozważań ma mniejszą wariancję od estymatora CMC, który ma wariancję $\text{Var } Y/(n)$.

Przykład 2.2 Rozpatrzmy teraz następujący problem. Przypuśćmy, że mamy N zadań do wykonania. Czasy zadań są niezależnymi zmiennymi losowymi o jednakowym rozkładzie z dystrybuantą F . Mamy do dyspozycji c serwerów. Możemy te zadania wykonać na wiele sposobów.

- według najdłuższego zadania (Longest Processing Time First - LPTF),
- według najkrótszego zadania (Shortest Processing Time First - SPTF),
- w kolejności numerów (First Come First Served – FCFS),
- lub alternując – (ALT), tzn. zadanie pierwsze do serwera pierwszego, zadanie 2-gie do serwera drugiego, itd. zadanie c -te do serwera c -tego, a potem od początku.

Następne zadanie zaczyna być opracowywane jedynie w momentach zakończenia poprzedniego zadania zadań. Oznacza to, że niedopuszczamy przerywań w trakcie pracy. Celem jest policzenie $\mathbb{E} C^{\text{SPTF}}$, $\mathbb{E} C^{\text{LPTF}}$ i $\mathbb{E} C^{\text{FCFS}}$, gdzie C jest czasem do zakończenia ostatniego zadania według procedury w super-indeksie. Zilustrujemy działanie dyscyplin na następującym prostym przykładzie. Powiedzmy, że $N = 5$ i $c = 2$ oraz zadania są wielkości 3, 1, 2, 4, 5. Wtedy postępując wg. SPTF zaczynamy z zadaniami wielkości 1 i 2 i po jednej jednostce czasu resztowe wielkości tych zadań są 3, 0, 1, 4, 5 a więc mamy teraz zadanie z $N = 4$ i $c = 2$. Postępując tak dalej widzimy, że $C^{\text{SPTF}} = 9$. Postępując wg. LPTF zaczynamy z zadaniami wielkości 4 i 5 i po 4-ch jednostkach czasu resztowe wielkości tych zadań są 3, 1, 2, 0, 1 a więc mamy teraz zadanie z $N = 4$ i $c = 2$. Postępując tak dalej widzimy, że $C^{\text{LPTF}} = 8$.

Możemy zastanawiać się jaka dyscyplina jest optymalna. Będzie to zależało od rozkładu F . Przypuśćmy więc, że rozmiary zadań są niezależne o jednakowym rozkładzie wykładniczym $\text{Exp}(1)$. Procedura ALT jest łatwa do zanalizowania. Załóżmy, że N jest parzyste. Wtedy $\mathbb{E} C^{\text{ALT}} = \mathbb{E} \max(X_1, X_2)$, gdzie

$$X_1 = \sum_{i=1}^{N/2} \xi_{1i}, \quad X_2 = \sum_{i=1}^{N/2} \xi_{2i},$$

gdzie ξ_{ij} są niezależnymi zmiennymi losowym o jednakowym rozkładzie wykładniczym $\text{Exp}(1)$. Niestety dla procedur LPTF i SPTF i nawet FCFS nie mamy wzorów teoretycznych. Przeprowadzimy więc symulację dla $N = 10$ zadań i $c = 2$. W każdym przypadku czas do zakończenia wszystkich zadań C jest pewną funkcją $C = \psi(U_1, \dots, U_{10})$, gdzie ψ zależy od dyscypliny. Nie mamy wzoru na konkretne ψ dla dyscyplin LPTF, SPTF, FCFS, ale heurystycznie możemy powiedzieć, że powinna być “niemalejąca”. W tablicy 2.1 podajemy wyniki symulacji $\mathbb{E}C$ dla różnych dyscyplin, przy użyciu metody CMC oraz antytecznej (anthy), dla $N = 10$ zadań oraz $m = 1000$ replikacjami; I jest wysymulowaną wartością $\mathbb{E}C = I$, s jest odchyleniem standardowym $\sqrt{\text{Var}(C)}$, oraz b jest połową długości przedziału ufności na poziomie $\alpha = 0.05$ (czyli można rzec błąd). Dla porównania w tablicy 2.2

dyscyplina	I	s	b
FCFS	5.4820	1.8722	0.1160
SPTF	5.8993	2.0259	0.1256
LPTF	5.0480	1.6908	0.1048
FCFSanthy	5.5536	0.8999	0.0558
SPTFanthy	5.9673	1.0127	0.0628
LPTFanthy	5.0913	0.6998	0.0434.

Tablica 2.1: Szacowanie C przy użyciu CMC i anthy; $N = 10$, liczba replikacji $m = 1000$.

podane są wyniki symulacji z $m = 100000$ replikacji. Wszystkie symulacje są robione na tych samych liczbach pseudolosowych, ponieważ algorytmy zaczynają się od $\text{rand}(\text{'state'}, 0)$.²

2.2 Wspólne liczby losowe

Przypuśćmy, że chcemy sprawdzić różnicę $d = \mathbb{E}k_1(X) - \mathbb{E}k_2(X)$, gdzie X jest wektorem losowym. Można to zrobić na dwa sposoby. Najpierw estymujemy $\mathbb{E}k_1(X)$ a następnie, niezależnie $\mathbb{E}k_2(X)$ i wtedy liczymy d . To nie zawsze jest optymalne, szczególnie gdy różnica d jest bardzo mała. Inny sposobem jest zapisanie $d = \mathbb{E}(k_1(X) - k_2(X))$ i następnie estymowania d . W pierwszym przypadku, $Y_1 = k_1(X_1)$ jest niezależne od $Y_2 = k_2(X_2)$, gdzie X_1, X_2 są niezależnymi replikacjami X . A więc wariancja

²algorytmy: FCFS.m, FCFSanty.m, SPTF.m, SPTFanty.m, LPTF.m LPTFanty.m.

dyscyplina	I	s	b
FCFS	5.4909	1.7909	0.0111
SPTF	5.8839	1.9116	0.0118
LPTF	5.0472	1.5945	0.0099
FCFSanthy	5.5015	0.8773	0.0054
SPTFanthy	5.8948	0.9748	0.0060
LPTFanthy	5.0556	0.6858	0.0043.

Tablica 2.2: Szacowanie C przy użyciu CMC i anthy; $N = 10$, liczba replikacji $m = 100000$

$\text{Var}(k_1(X_1) - k_2(X_2)) = \text{Var}(k_1(X)) + \text{Var}(k_2(X))$. W drugim przypadku $\text{Var}(k_1(X) - k_2(X)) = \text{Var}(k_1(X)) + \text{Var}(k_2(X)) - 2\text{Cov}(k_1(X), k_2(X))$. Jeśli więc $\text{Cov}(k_1(X), k_2(X)) > 0$ to istotnie zmniejszymy wariancję. Tak będzie w przypadku gdy obie funkcje są jednakowo monotoniczne.

Przykład 2.3 Kontynuujemy przykład 2.2. Tym razem pytamy się o oczekiwaną różnicę $C^{\text{SPTF}} - C^{\text{LPTF}}$, tj. $I = \mathbb{E}(C^{\text{SPTF}} - C^{\text{LPTF}})$. Można po prostu odjąć obliczone odpowiednie wielkości w poprzednim przykładzie, ale wtedy za wariancję trzeba by wziąć sumę odpowiednich wariancji, i przy tej samej długości symulacji błąd byłby większy. Można też wykorzystać metodę wspólnych liczb losowych. W tablicy 2.3 podajemy wyniki symulacji $I = \mathbb{E}(C^{\text{SPTF}}) - C^{\text{LPTF}}$ przy użyciu metody wspólnej liczby losowej (JCN – *Joint Common Number*). Podobnie jak poprzednio, robimy też symulacje przy użyciu zgrubnej metody Monte Carlo (CMC), dla $N = 10$ zadań, z $m = 1000$ replikacjami, s jest odchyleniem standardowym $\sqrt{\text{Var}((C^{\text{SPTF}} - C^{\text{LPTF}}))}$, oraz b jest połową długości przedziału ufności na poziomie $\alpha = 0.05$.³ Wszystkie symulacje są robione na tych samych liczbach

dyscyplina	I	s	b
SmLPTFjoint	0.8513	0.5431	0.0352
SmLPTF	1.0125	2.5793	0.1599

Tablica 2.3: Szacowanie I przy użyciu JCN i CMC; $N = 10$, liczba replikacji $m = 1000$

bach pseudolosowych, ponieważ algorytmy zaczynają się od `rand('state',0)`. Dla porównania podajemy w tablicy 2.4 wyniki dla 100000 replikacji.

³skrypty SmLPTF.m, SmLPTFjoint.m.

dyscyplina	I	s	b
SmLPTFjoint	0.8366	0.5166	0.0032
SmLPTF	0.8455	2.4953	0.0155

Tablica 2.4: Szacowanie I przy użyciu JCN i CMC ; $N = 10$, liczba replikacji $m = 100000$

3 Metoda zmiennych kontrolnych

Zacznijmy, że jak zwykle aby obliczyć I znajdujemy taką zmienną losową Y , że $I = \mathbb{E} Y$. Wtedy $\hat{Y}^{\text{CMC}} = \sum_{j=1}^n Y_j/n$ jest estymatorem MC dla I , gdzie $Y_1, \dots, Y_n$ są niezależnymi replikacjami Y . Wariancja tego estymatora wynosi $\text{Var} \hat{Y}_n^{\text{CMC}} = \text{Var} Y/n$. Teraz przypuśćmy, że symulujemy jednocześnie drugą zmienną losową X (a więc mamy niezależne replikacje par $(Y_1, X_2), \dots, (Y_n, X_n)$) dla której znamy $\mathbb{E} X$. Niech $\hat{X}_n = \sum_{j=1}^n X_j/n$ i definiujemy

$$\hat{Y}_n^{CV} = \hat{Y}_n^{\text{CMC}} + c(\hat{X}_n - \mathbb{E} X) = \text{Var} Y + c^2 \text{Var} X + 2c \text{Cov}(Y, X).$$

Obliczmy wariancję nowego estymatora:

$$\text{Var}(\hat{Y}_n^{CV}) = \text{Var}(Y + cX)/n.$$

Wtedy $\text{Var}(\hat{Y}_n^{CV})$ jest minimalna gdy c ,

$$c = -\frac{\text{Cov}(Y, X)}{\text{Var}(X)}$$

i minimalna wariancja wynosi $\sigma_Y^2(1 - \rho^2)/n$, gdzie $\rho = \text{corr}(Y, X)$. A więc wariancja jest zredukowana przez czynnik $1 - \rho^2$. Widzimy więc, że należy starać się mieć bardzo skorelowane Y z X . Ale musimy też umieć symulować zależną parę (Y, X) .

W praktyce nie znamy $\text{Cov}(Y, X)$ ani $\text{Cov}(X)$ i musimy te wielkości symulować stosując wzory

$$\hat{c} = -\frac{\hat{S}_{YX}^2}{\hat{S}_X^2}$$

gdzie

$$\begin{aligned}\hat{S}_X^2 &= \frac{1}{n-1} \sum_{j=1}^n (X_j - \hat{X})^2 \\ \hat{S}_{YX}^2 &= \frac{1}{n-1} \sum_{j=1}^n (Y_j - \hat{Y})(X_j - \hat{X}).\end{aligned}$$

Przykład 3.1 Kontynuujemy przykład IV.1.1. Liczyliśmy tam π za pomocą estymatora $Y = 4 \mathbf{1}(U_1^2 + U_2^2 \leq 1)$. Jako zmienną kontrolną weźmiemy $X = \mathbf{1}(U_1 + U_2 \leq 1)$, które jest mocno skorelowane z Y oraz $\mathbb{E} X = 1/2$. Możemy policzyć $1 - \rho^2 = 0.727$. Teraz weźmiemy inną zmienną kontrolną $X = \mathbf{1}(U_1 + U_2 > \sqrt{2})$. Wtedy $\mathbb{E} X = (2 - \sqrt{2})^2/2$ i $1 - \rho^2 = 0.242$. Zauważmy (to jest zadanie) że jeśli weźmiemy za zmienną kontrolną X lub $a + bX$ to otrzymamy taką samą redukcję wariancji.

Przykład 3.2 Chcemy obliczyć całkę $\int_0^1 g(x) dx$ metodą MC. Możemy położyć $Y = g(U)$ i jako kontrolną zmienną przyjąć $X = f(U)$ jeśli znamy $\int_0^1 f(x) dx$.

4 Warunkowa metoda MC

Punktem wyjścia jest jak zwykle obliczanie $I = \mathbb{E} Y$. Estymatorem CMC jest wtedy

$$\hat{Y}_n = \frac{1}{n} \sum_{j=1}^n Y_j,$$

gdzie $Y_1, \dots, Y_n$ są niezależnymi kopiami Y . Oprócz zmiennej Y mamy też X . Dla ułatwienia rozważań założmy, że wektor (Y, X) ma łączną gęstość $f_{Y,X}(y, x)$. Wtedy brzegowa gęstość X jest

$$f_X(x) = \int f_{Y,X}(y, x) dy.$$

Niech

$$\begin{aligned}\phi(x) &= \mathbb{E}[Y|X=x] \\ &= \frac{\int y f_{Y,X}(y, x) dy}{f_X(x)}.\end{aligned}$$

Z twierdzenia o prawdopodobieństwie całkowitym

$$I = \mathbb{E} Y = \mathbb{E} \phi(X).$$

Wtedy definiujemy estymator warunkowy

$$\hat{Y}_n^{\text{cond}} = \frac{1}{n} \sum_{j=1}^n Y_j^{\text{cond}}$$

gdzie $Y_1^{\text{cond}}, \dots, Y_n^{\text{cond}}$ są replikacjami $Y^{\text{cond}} = \phi(X)$. Estymator $\hat{Y}_n^{\text{cond}}$ jest nieobciążony. Pokażemy, że ma mniejszą wariancję niż estymator CMC. W tym celu skorzystamy z nierówności Jensena

$$\mathbb{E}[Y^2|X=x] \geq (\mathbb{E}[Y|X=x])^2.$$

Jeśli X ma gęstość f_X , to

$$\begin{aligned} \text{Var}(Y) &= \int \mathbb{E}[Y^2|X=x] f_X(x) dx - I^2 \\ &\geq \int (\mathbb{E}[Y|X=x])^2 f_X(x) dx - I^2 \\ &= \int \phi^2(x) f_X(x) dx - I^2 \\ &= \text{Var}(\phi(V)) \end{aligned}$$

Dla czytelników znających ogólną teorię warunkowych wartości oczekiwanych wystarczy przytoczyć ogólny wzór

$$\text{Var} Y = \mathbb{E} \text{Var}(Y|X) + \text{Var} \mathbb{E}(Y|X),$$

który można uzasadnić następująco. Wychodząc prawej strony

$$\begin{aligned} &\mathbb{E}[\text{Var}(Y|X)] + \text{Var}(\mathbb{E}[Y|X]) \\ &= \mathbb{E}(\mathbb{E}[Y^2|X] - (\mathbb{E}[Y|X])^2) + \mathbb{E}(\mathbb{E}[Y|X])^2 - (\mathbb{E}(\mathbb{E}[Y|X]))^2 \\ &= \mathbb{E} Y^2 - (\mathbb{E} Y)^2. \end{aligned}$$

Stąd mamy od razu

$$\text{Var} Y \geq \text{Var} \mathbb{E}(Y|X).$$

Przykład 4.1 Kontynuujemy przykład IV.1.1. Liczyliśmy tam π za pomocą estymatora $Y = 4\mathbf{1}(U_1^2 + U_2^2 \leq 1)$. Przyjmując za $X = U_1$ mamy

$$\phi(x) = \mathbb{E}[Y|X = x] = 4\sqrt{1 - x^2}.$$

Estymator zdefiniowany przez $Y^{\text{cond}} = 4\sqrt{1 - U_1^2}$ ma wariancję

$$\text{Var } \hat{Y}_n^{\text{cond}} = \frac{1}{n} \left(\int_0^1 16(1 - x^2) dx - \pi^2 \right) = \frac{0.797}{n}.$$

A więc redukuje wariancję 3.38 razy w stosunku do estymatora CMC z przykładu. Ten estymator będziemy dalej analizować w przykładzie 5.1. Zauważmy, że w przykładzie IV.1.1 w przypadku metody CMC policzyliśmy, że $\text{Var}(Y) = 2.6968$.

Przykład 4.2 Niech $I = \mathbb{P}(\xi_1 + \xi_2 > y)$, gdzie ξ_1, ξ_2 są niezależnymi zmiennymi losowymi z jednakową dystrybuantą F , która jest znana, a dla której nie mamy analitycznego wzoru na F^{*2} . Przez $\bar{F}$ jak zwykle oznaczamy ogon dystrybuanty $1 - F(x)$. Oczywiście za Y możemy wziąć

$$Y = \mathbf{1}(\xi_1 + \xi_2 > y)$$

a za $X = \xi_1$. Wtedy

$$Y^{\text{cond}} = \bar{F}(y - \xi_1).$$

Niech $X_1, \dots, X_n$ będą niezależnymi zmiennymi losowymi o takim samym rozkładzie jak ξ_1 . Wtedy $\hat{Y}_n^{\text{cond}} = (1/n) \sum_{j=1}^n \bar{F}(y - X_j)$ będzie nieobciążonym estymatorem I z wariancją mniejszą niż w zgrubnym estymatorze.

Przykład 4.3 Będziemy w pewnym sensie kontynuować rozważania z przykładu 4.2. Niech $S_n = \xi_1 + \dots + \xi_n$, gdzie $\xi_1, \xi_2, \dots, \xi_n$ są niezależnymi zmiennymi losowymi z jednakową gęstością f . Estymator warunkowy można też używać do estymowania gęstości $f^{*n}(x)$ dystrybuanty $F^{*n}(x)$, gdy jest dana jawna postać f . Wtedy estymator warunkowy dla $f^{*n}(y)$ zbudujemy na podstawie

$$Y^{\text{cond}} = f(y - S_{n-1}).$$

5 Losowanie istotnościowe

Przypuśćmy, że chcemy obliczyć I , które możemy przedstawić w postaci

$$I = \int_E k(x)f(x) dx = \mathbb{E} k(Y),$$

gdzie E jest podzbiorem $\mathbb{R}$ lub $\mathbb{R}^d$, f jest funkcją gęstości na E , oraz funkcja $k(x)$ spełnia $\int_E (k(x))^2 f(x) dx < \infty$. Na przykład E jest odcinkiem $[0, 1]$ lub półprostą $[0, \infty)$. Zakładamy, że Y określona na przestrzeni probabilistycznej $(\Omega, \mathcal{F}, \mathbb{P})$ jest o wartościach w E . Estymatorem CMC dla I jest

$$\hat{Y}_n^{\text{CMC}} = \frac{1}{n} \sum_{j=1}^n Y_j,$$

gdzie $Y_1, \dots, Y_n$ są niezależnymi kopiami Y . Niech $\tilde{f}(x)$ będzie gęstością na E , o której będziemy zakładać, że jest ściśle dodatnia na E i niech X będzie zmienną losową z gęstością $\tilde{f}(x)$. Wtedy

$$I = \int_E k(x) \frac{f(x)}{\tilde{f}(x)} \tilde{f}(x) dx = \mathbb{E} k(X) \frac{f(X)}{\tilde{f}(X)} = \mathbb{E} k_1(X),$$

gdzie $k_1(x) = k(x)f(x)/\tilde{f}(x)$.

Teraz definiujemy *estymator istotnościowy (importance sampling)*

$$\hat{Y}_n^{\text{IS}} = \frac{1}{n} \sum_{j=1}^n k(X_j) \frac{f(X_j)}{\tilde{f}(X_j)},$$

gdzie $X_1, \dots, X_n$ są niezależnymi kopiami X . Oczywiście $\hat{Y}_n^{\text{IS}}$ jest nieobciążony bo

$$\mathbb{E} k_1(X) = \int_E k(x) \frac{f(x)}{\tilde{f}(x)} \tilde{f}(x) dx = I,$$

z wariancją

$$\text{Var}(\hat{Y}_n^{\text{IS}}) = \frac{1}{n} \text{Var} \left[k(X) \frac{f(X)}{\tilde{f}(X)} \right] = \frac{1}{n} \left(\int_E \left(k(x) \frac{f(x)}{\tilde{f}(x)} \right)^2 \tilde{f}(x) dx - I^2 \right).$$

Okazuje się, że odpowiednio wybierając $\tilde{f}(x)$ można istotnie zmniejszyć wariancję. Niestety, jak pokazuje następujący przykład, można też istotnie sytuację pogorszyć.

Przykład 5.1 Funkcja ćwiartki koła jest zadana wzorem

$$k(x) = 4\sqrt{1-x^2}, \quad x \in [0, 1],$$

i oczywiście

$$\int_0^1 4\sqrt{1-x^2} dx = \pi.$$

W tym przykładzie $Y = U$ jest o rozkładzie jednostajnym na $E = [0, 1]$. Zauważmy, że $f(x) = \mathbf{1}_{[0,1]}(x)$. Wtedy estymator CMC jest dany przez

$$\hat{Y}_n^{\text{CMC}} = \frac{1}{n} \sum_{j=1}^n 4\sqrt{1-U_j^2}. \quad (5.5)$$

Jest to estymator który spotkaliśmy w przykładzie [??example:pi3??] i ma wariancję

$$\frac{1}{n} \left(\int_0^1 16(1-x^2) dx - \pi^2 \right) = \frac{0.797}{n}.$$

Niech teraz $\tilde{f}_1(x) = 2x$. Wtedy estymatorem istotnościowym jest

$$\hat{Y}_n^{\text{IS}} = \frac{1}{n} \sum_{i=1}^n \frac{4\sqrt{1-X_i^2}}{2X_i},$$

gdzie $X_1, \dots, X_n$ są niezależnymi replikacjami liczby losowej X z gęstością $2x$, ma wariancję

$$\text{Var}(\hat{Y}_n^{\text{IS}}) = \frac{1}{n} \text{Var} \frac{k(X)}{\tilde{f}_1(X)}.$$

Na koniec policzymy, że wariancja

$$\text{Var} \frac{k(X)}{\tilde{f}_1(X)} = \mathbb{E} \left(\frac{4\sqrt{1-X^2}}{2X} \right)^2 - I^2 = 8 \int_0^1 \frac{1-x^2}{x} dx - I^2 = \infty.$$

Teraz pokażemy inną gęstość $\tilde{f}_2$ dla której estymator istotnościowy, jest z mniejszą wariancją niż dla estymatora (5.5). Niech teraz

$$\tilde{f}_2(x) = (4-2x)/3.$$

Wtedy

$$\hat{Y}_n^{\text{IS}} = \frac{1}{n} \sum_{i=1}^n \frac{4\sqrt{1-X_i^2}}{(4-2X_i)/3},$$

i

$$\text{Var} \frac{k(X)}{\tilde{f}(X)} = \int_0^1 \frac{16(1-x^2)}{(4-2x)/3} - \pi^2 = 0.224$$

a więc estymator IS jest ponad 3 razy lepszy od estymatora CMC.

Założmy na chwilę, że funkcja k jest nieujemna i rozważmy nową gęstość

$$\tilde{f}(x) = k(x)f(x) / \int_E k(x)f(x) dx = k(x)f(x)/I.$$

Wtedy wariancja estymatora IS jest zero, ponieważ

$$\text{Var } Y^{\text{IS}} = \int_E \left(\frac{k(x)f(x)}{k(x)f(x)/I} \right)^2 \frac{k(x)f(x)}{I} / dx - I^2 = 0$$

Oczywiście taki estymator nie ma praktycznego sensu, bo do jego definicji potrzebujemy I , które właśnie chcemy obliczyć. Ale ta uwaga daje nam wskazówkę jak dobrać $\tilde{f}$. Mianowicie ma to być gęstość przypominająca kształtem $k(x)f(x)$. Proponujemy porównać $k(x)f(x)$ z odpowiednimi gęstościami $\tilde{f}_1$ i $\tilde{f}_2$ z przykładu 5.1.

Podany teraz materiał jest opcjonalny. Zauważmy, że te same rozważania można przeprowadzić formalnie inaczej. Niech $(\Omega, \mathcal{F})$ będzie przestrzenią mierzalną i $X_1, X_2, \dots$ będą zmiennymi losowymi na tej przestrzeni. Przyjmijmy, że $\mathcal{F}_n$ jest generowana przez $\{X_1 \in B_1, \dots, X_n \in B_n\}$, gdzie $B_j \in \mathcal{B}(\mathbb{R})$ ($j = 1, \dots, n$) oraz, że $\mathcal{F} = \sigma(\bigcup_{n \geq 1} \mathcal{F}_n)$. Teraz $\mathbb{P}$ i $\tilde{\mathbb{P}}$ są miarami probabilistycznymi na Ω takimi, że odpowiednio $X_1, X_2, \dots$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie odpowiednio z gęstością f lub $\tilde{f}$.

Fakt 5.2 *Niech $\tilde{f} > 0$ p.w.. Wtedy na $\mathcal{F}_n$*

$$\mathbb{P}(d\omega) = L_n(\omega) \tilde{\mathbb{P}}(d\omega),$$

gdzie

$$L_n = \prod_{j=1}^n \frac{f(X_j)}{\tilde{f}(X_j)}. \quad (5.6)$$

Jeśli Y jest zmienną losową mierzalną $\mathcal{F}_n$, to

$$I = \mathbb{E} Y = \tilde{\mathbb{E}} [Y L_n]. \quad (5.7)$$

Dowód Niech $A = \{X_1 \in B_1, \dots, X_n \in B_n\} \in \mathcal{F}_n$. Wtedy

$$\begin{aligned} \mathbb{P}(A) &= \mathbb{P}(X_1 \in B_1, \dots, X_n \in B_n) = \int_{B_1} \dots \int_{B_n} f(x_1) \dots f(x_n) dx_1 \dots dx_n \\ &= \int_{B_1} \dots \int_{B_n} \frac{f(x_1) \dots f(x_n)}{\tilde{f}(x_1) \dots \tilde{f}(x_n)} \tilde{f}(x_1) \dots \tilde{f}(x_n) dx_1 \dots dx_n. \end{aligned}$$

To znaczy

$$\mathbb{E} \mathbf{1}(A) = \tilde{\mathbb{E}} L_n \mathbf{1}(A).$$

Stąd aproksymując Y funkcjami prostym otrzymujemy (5.7). $\square$

Zauważmy, że L zdefiniowany przez (5.6) nazywa się ilorazem wiarygodności.

Przykład 5.3 Policzmy teraz iloraz wiarygodności dla przypadku, gdy f jest gęstością standardowego rozkładu normalnego $N(0,1)$, natomiast $\tilde{f}$ jest gęstością rozkładu normalnego $N(\mu,1)$. Wtedy

$$L_n = \exp\left(-\mu \sum_{j=1}^n X_j + \frac{\mu^2}{2}n\right). \quad (5.8)$$

Postępując odwrotnie zdefiniujmy $\tilde{\mathbb{P}}$ na $\mathcal{F}_n$ przez

$$d\tilde{\mathbb{P}} = L_n^{-1} d\mathbb{P}.$$

Zauważmy, że teraz musimy założyć, że $f(x) > 0$ p.w.. Wtedy przy nowej mierze probabilistycznej $\tilde{\mathbb{P}}$ zmienne $X_1, X_2, \dots$ są niezależne o jednakowym rozkładzie $N(\mu,1)$.

Zadania teoretyczne

4

5.1 Niech $F \sim \text{Exp}(\lambda)$ i $Z_1 = F^{-1}(U)$, $Z_2 = F^{-1}(1 - U)$. Pokazać, że

$$\mathbb{E} Z_i = 1/\lambda, \quad \mathbb{E} Z_i^2 = 2/\lambda^2,$$

oraz

$$\text{corr}(Z_1, Z_2) = -0.6449.$$

$$\text{Wsk. } \int_0^1 \log x \log(1 - x) dx = 0.3551.$$

5.2 Asmussen and Glynn [4]. Niech X, Y mają taką samą brzegową dystrybuantę F . Pokazać, korzystając z teorii przedstawionej poniżej, że dolne ograniczenie łącznej dystrybuanty $F(x, y)$ jest dane przez z twierdzenia Fréchet-Hoeffdinga

$$F(x, y) \geq F_{\text{FH}}(x, y)$$

gdzie $F_{\text{FH}}(x, y)$ jest dwuwymiarową dystrybuantą wektora $F^{\leftarrow}(U), F^{\leftarrow}(1 - U)$. Korzystając z wzoru

$$\mathbb{E} XY = \int_0^\infty \int_0^\infty [1 - F(x) - F(y) + F(x, y)] dx dy$$

pokazać, że

$$\rho = \text{corr}(X, Y) = \text{Cov}(X, Y) \geq \mathbb{E}[\log U \log(1 - U)] - 1 = -0.645.$$

5 7

5.3 Obliczyć wartość oczekiwaną i wariancję rozkładu Pareto $\text{Par}(\alpha)$.

5.4 Niech $Z_1 = F^{-1}(U)$, $Z_2 = F^{-1}(1-U)$. Oszacować korelację $\text{corr}(Z_1, Z_2)$ gdy F jest dystrybuantą:

- (a) rozkładu Poissona; $\lambda = 1, 2, \dots, 10$,
- (b) rozkładu geometrycznego $\text{Geo}(p)$.

⁵Trochę teorii:

By *copula* we mean any two-dimensional distribution function with margins uniformly distributed $U(0, 1)$. We will denote it by $C(t_1, t_2)$. Thus a function $C : [0, 1] \times [0, 1]$ is a copula, if

1. $\lim_{u_i \rightarrow 0} C(u_1, u_2) = 0$, for $i = 1, 2$,
2. $\lim_{u_1 \rightarrow 1} C(u_1, u_2) = u_2$ and $\lim_{u_2 \rightarrow 1} C(u_1, u_2) = u_1$
3. $C(v_1, v_2) - C(u_1, v_2) - C(v_1, u_2) + C(u_1, u_2) \geq 0$ for any $u_1 \leq v_1, u_2 \leq v_2$.

One can prove the following result. The general version for m -dimensional random vectors is called Sklar's theorem.⁶

Theorem For any distribution function $F_{xy}(t_1, t_2)$, there exists a copula $C(u_1, u_2)$ such that

$$F_{xy}(t_1, t_2) = C(F_x(t_1), F_y(t_2)), \quad -\infty < t_1, t_2 < \infty. \quad (5.9)$$

If the margins are continuous, then C is unique. Conversely, if C is a copula, and F_x, F_y are distribution functions, then the function F defined by (5.9) is a joint distribution with margins F_x, F_y .

Proposition Let F be a joint distribution function with continuous margins F_x, F_y . Then the unique copula of F is given by

$$C(u_1, u_2) = F(F_x^{\leftarrow}(u_1), F_y^{\leftarrow}(u_2)).$$

In two dimensions, i.e. the bivariate case, the Fréchet–Hoeffding Theorem states

$$\max(u + v - 1, 0) \leq C(u, v) \leq \min(u, v).$$

⁷Wsk.

$$\int_0^\infty \int_0^\infty xy \, dF(x, y) = \int_0^\infty \int_0^\infty \left(\int_0^\infty \int_0^\infty \mathbf{1}(t \leq x) \mathbf{1}(s \leq y) \, ds \, dt \right) dF(x, y).$$

5.5 a) Napisać algorytm na generowanie zmiennej losowej Y z gęstością

$$f(x) = \begin{cases} 2x & 0 \leq x \leq 1, \\ 0 & \text{w pozostałych przypadkach.} \end{cases}$$

Uzasadnić!

b. Przypuśćmy, że chcemy symulować $I = \mathbb{E}Y$, gdzie Y ma rozkład jak w części a. Ile należy zaplanować replikacji aby otrzymać I z dokładnością do jednego miejsca po przecinku, na poziomie istotności $\alpha = 0.05$, gdy

1. stosujemy zgrubny estymator Monte Carlo – CMC,
2. stosujemy metodę zmiennych antytetycznych.

Wsk: $\text{corr}(\text{rand}^{1/2}, (1 - \text{rand})^{1/2}) = -0.9245$. Przyjąć $b = 0.01$.

5.6 Napisać wzór na estymator warstwowy, jeśli warstwy definiowane są przez zmienną X , jak zostało to zaproponowane w części Modyfikacje sekcji 1.

5.7 (Metoda zmiennych antytetycznych.) Zastosować metodę zmiennych antytetycznych do oszacowania

$$I = \int_0^1 e^x dx$$

biorąc $Y_{2j-1} = e^{U_j}$, $Y_{2j} = e^{1-U_j}$. O ile procent zredukujemy wariancję w stosunku do estymatora CMC.

5.8 Zastosować metodę zmiennych antytetycznych do oszacowania

$$I = 4 \int_0^1 \sqrt{1-x^2} dx$$

biorąc $Y_{2j-1} = 4\sqrt{1-U_j^2}$, $Y_{2j} = 4\sqrt{1-(1-U_j)^2}$. O ile procent zredukujemy wariancję w stosunku do estymatora CMC?

5.9 [Asmussen & Glynn [4]] Przedyskutować metodę redukcji wariancji dla obliczenia

$$\int_0^\infty (x + 0.02x^2) \exp(0.1\sqrt{1+\cos x} - x) dx$$

metodą MC.

- 5.10 (Losowanie istotnościowe) Podać procedurę losowania istotnościowego dla estymacji

$$I = \int_0^1 e^{-x^2} dx$$

biorąc X o gęstości $\tilde{f}(x) = re^{-x}$ (wyliczyć r tak aby $\tilde{f}$ była gęstością na $(0, 1)$). Użyć komputera do wyliczenia odpowiednich całek. O ile procent zostanie zredukowana wariancja w stosunku do estymatora CMC.

- 5.11 Podać procedurę losowania istotnościowego dla estymacji

$$I = \int_0^1 x^{\alpha-1} e^{-x} dx$$

dla $\alpha = 3/4$, biorąc X jak w zadaniu 5.10, przyjmując $k(x) = x^{\alpha-1}e^{-x}$ oraz $f(x) = 1$. O ile procent zostanie zredukowana wariancja w stosunku do estymatora CMC.

- 5.12 Chcemy oszacować $\int_0^1 k(x) dx$ gdzie $k(x) = 4\sqrt{1-x^2}$. Niech $Y = k(U)$. Przeprowadzić analizę użycia metody zmiennych kontrolnych z $X = h(U)$, gdzie $h(x) = 4 - 4x$.

- 5.13 Pokazać, że w metodzie zmiennych kontrolnych otrzymamy to samo $1 - \rho^2$ dla $(Y, aX + b)$ niezależnie od $a \neq 0$ i b .

- 5.14 Pokazać, że estymator istotnościowy do obliczenia $I = \int_E k(x)f(x)$ zdefiniowany przez

$$k(x)f(x)/I$$

ma wariancję równą zero.

- 5.15

Madras [?] . Chcemy oszacować całkę $\int_0^4 x^{1/2}e^{-x} dx$ metodą MC, przy użyciu metody losowania istotnościowego z gęstością $\tilde{f}(x) = e^{-x}/(1 - e^{-4})$.
 (i) Zaprojektować algorytm przy użyciu n liczb losowych $U_1, \dots, U_n$.
 (ii) Czy ma sens zastosowanie metody zmiennych antytetycznych.
 Zastanowić się nad innymi metodami MC obliczenia całki.

5.16 [Madras [?]] (i) Chcemy symulować $I = \int_0^\infty k(x) dx$, gdzie przez

$$k(x) = \sin(x)(1+x)^{-2}e^{-x/5}$$

za pomocą estymatora istotnościowego przy losowaniu zmiennych o rozkładzie wykładniczym z parametrem λ . Pokazać, że dla pewnych wielkości λ wariancja estymatora istotnościowego jest nieskończona.

(ii) Pokazać, że jeśli $k(x) = \sin(x)(1+x)^{-2}$, to wariancja estymatora istotnościowego przy użyciu losowań o rozkładzie wykładniczym jest zawsze nieskończona. Zaprojektować dobrą metodą symulacji I .

(iii) Przypuśćmy, że całka I jest zbieżna warunkowo (tzn. nie jest zbieżna absolutnie). Pokazać, że wtedy każdy estymator istotnościowy ma nieskończoną średnią.

5.17 Niech $I = \mathbb{P}(L > x)$, gdzie

$$L = \sum_{j=1}^n D_n X_n,$$

$X_1, \dots, X_n$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $N(0,1)$, niezależnymi od $D_1, \dots, D_n$. Niech P ma rozkład $\text{Beta}(1,19)$ z gęstością $(1-p)^{18}/19$, $0 < p < 1$. Pod warunkiem $P = p$ zmienne $D_1, \dots, D_n$ są niezależnymi próbami Bernoulliego z prawdopodobieństwem sukcesu p , niezależnymi od $X_1, \dots, X_n$. Podać algorytm symulacji $I = \mathbb{P}(L > x)$ warunkową metodą MC, z warunkowaniem względem $\sum_{j=1}^n D_j$. Wziąć

$$x = 3\mathbb{E} L = 3n\mathbb{E} P\mathbb{E} X = 3 \cdot 100 \cdot 0.05 \cdot 3 = 45.$$

(Asmussen&Glynn, p. 147).

5.18 (Kwadraty łacińskie) W przypadku symulacji $I = \int_{[0,1]^d} g(x) dx$, gdy d jest duże, metoda warstw po każdej zmiennej będzie nieużyteczną ze względu na dużą liczbę warstw. Mianowicie, jeśli na każdym boku chcemy mieć n warstw, to razem w $[0,1]^d$ mamy n^d warstw. Wtedy rozszerzając przykład 1.6 z jednego wymiaru na d mielibyśmy

$$\mathbf{V} = \begin{pmatrix} V_{11} & \dots & V_{1n} \\ \vdots & \dots & \vdots \\ V_{d1} & \dots & V_{dn} \end{pmatrix},$$

gdzie $V_{ij} = \frac{i-1}{n} + \frac{U_{ij}}{n}$. Remedium jest wykorzystanie tzw. kwadratów łacińskich. Niech $U_i^{(j)}$ będą niezależnymi zmiennymi losowymi o rozkładzie $U(0,1)$ oraz $\pi_i = (\pi_i(1), \dots, \pi_i(n))$ będzie ciągiem niezależnych losowych permutacji $(1, \dots, n)$, $i = 1, \dots, d$. Dla $j = 1, \dots, n$

$$V^{(j)} = (V_1^{(j)}, \dots, V_d^{(j)}) \in [0, 1]^d,$$

gdzie

$$V_i^{(j)} = \frac{\pi_i(j) - 1 + U_i^{(j)}}{n}$$

Pokazać, że estymator

$$\hat{Y}_n^{LH} = \frac{1}{n} \sum_{j=1}^n g(V^{(j)})$$

jest estymatorem nieobciążonym I . Pokazać, że

$$\text{Var } \hat{Y}_n^{LH} = \frac{\sigma_Y^2}{n} + \frac{n-1}{n} \text{Cov}(g(V^{(1)}), g(V^{(2)}))$$

gdzie $\sigma_Y^2 = \text{Var } g(U_1, \dots, U_d)$.

Zadania laboratoryjne

- 5.19 Niech $k_1(x) = e^{\sqrt{x}}$ i $k_2(x) = (1 + \frac{\sqrt{x}}{50})^{50}$ oraz $X = U$. Można oszacować, że $|k_1(x) - k_2(x)| \leq 0.03$, a więc $d \leq 0.03$. Przeprowadzić eksperyment obliczenia I oraz b obiema metodami rozpatrywanymi w podrozdziale 2.2.
- 5.20 Niech F będzie dystrybuantą rozkładu standardowego normalnego. Niech $p_i = 1/10$. Korzystając z tablic rozkładu normalnego wyznaczyć a_j , takie, że

$$\begin{aligned} a_0 &= -\infty \\ a_1 &= F^{-1}(1/10) \\ a_2 &= F^{-1}(2/10) \\ &\vdots \\ a_9 &= F^{-1}(9/10) \\ a_{10} &= \infty. \end{aligned}$$

W pierwszym eksperymencie przeprowadzić 100 replikacji i narysować histogram z klasami $A_1, \dots, A_{10}$. Natomiast drugi eksperyment przeprowadzić przy losowaniu w warstwach losując w każdej 10 replikacji. Porównać otrzymane rysunki.

Rozdział VI

Symulacje stochastyczne w badaniach operacyjnych

¹ W tym rozdziale pokażemy wybrane przykłady zastosowań symulacji do konkretnych zadań w szeroko pojętej teorii badań operacyjnych, modelach telekomunikacyjnych, itd. Oczywiście przykładowe zadania podane będą w postaci uproszczonej. Nowym elementem pojawiającym się w tym rozdziale, jest to, że niektóre rozpatrywane procesy ewoluują losową w ciągłym czasie. Będziemy rozpatrywać pewne procesy stochastyczne, ale do symulacji będziemy odwoływać się do modelu a nie do konkretnej teorii procesów stochastycznych.

Aby dobrze zanalizować postawione zadanie należy

- zidentyfikować pytania które chcemy postawić,
- poznać system,
- zbudować model,
- zweryfikować poprawność założeń modelu,
- przeprowadzić eksperymenty z modelem,
- wybrać rodzaj prezentacji.

Ewolucja interesującej nas wielkości zmieniającej się wraz z czasem (np. ilość oczekujących zadań w chwili t , gdzie $0 \leq t \leq T_{\max}$, stan zapasów w

¹17.1.2018

chwili t , itd.) jest losowa. Zmienna, która nas interesuje może być skalarem lub wektorem. Zmienna ta może się zmieniać ciągle (patrz przykład realizacji ruchu Browna na rys. [????]) albo jedynie w chwilach dyskretnych. Wszystkie procesy możemy symulować w chwilach $0, \Delta, 2\Delta, \dots$ i wtedy mamy do czynienia z *symulacją synchroniczną*. Jednakże dla specyficznej klasy zadań, gdy ewolucja zmienia się dyskretnie możemy przeprowadzać *symulację asynchroniczną*, i to będzie głównym tematem rozważanym w tym rozdziale.

W wielu problemach mamy do czynienia ze zgłaszającymi się zadaniami/jednostkami itp. w sposób niezdeterminowany. Do opisu takiego strumienia zgłoszeń potrzebujemy modeli z teorii procesów stochastycznych. W następnym podrozdziale opiszemy tzw. proces Poissona (jednorodny i niejednorodny w czasie), który w wielu przypadkach jest adekwatym modelem.

1 Procesy zgłoszeń

1.1 Jednorodny proces Poissona

Zajmiemy się teraz opisem i następnie sposobami symulacji losowego rozmieszczenia punktów. Tutaj będziemy zajmowali się punktami na $\mathbb{R}_+$, które modelują *momenty zgłoszeń*. Proces zgłoszeń jest na przykład potrzebny do zdefiniowania procesu kolejkowego lub procesu ryzyka. Najprostszy sposób jest przez zadanie tak zwanego procesu liczącego $N(t)$. Jest to proces $N(t)$ ($t \geq 0$), który zlicza punkty do momentu t . Zauważmy, że ponieważ momenty zgłoszeń na $\mathbb{R}_+$ są losowe, więc $N(t)$ jest procesem losowym, który ma niemalejące, kawałkami stałe z jednostkowymi skokami realizacje i $N(0) = 0$. Ponadto przyjmujemy, że jako funkcja czasu t jest prawostronnie ciągły. Interpretacją zmiennej losowej $N(t)$ jest liczba punktów do chwili t lub w odcinku $(0, t]$. Zauważmy, że $N(t+h) - N(t)$ jest liczbą punktów w odcinku $(t, t+h]$. W dalszej części będziemy oznaczać przez $0 < A_1 < A_2 < \dots$ momenty zgłoszeń na osi dodatniej $(0, \infty)$. Przez

$$\tau_1 = A_1, \quad \tau_i = A_i - A_{i-1}, \quad (i = 2, \dots) \quad (1.1)$$

oznaczamy *ciąg odstępów między zgłoszeniami*. Wtedy

$$N(t) = \#\{i : A_i \leq t\}$$

nazywamy procesem liczącym.

Przez proces Poissona (lub *jednorodny proces Poissona*) o intensywności $\lambda > 0$ nazywamy proces liczący dla którego (τ_j) są niezależnymi zmiennymi losowymi o jednakowym rozkładzie wykładniczym $\text{Exp}(\lambda)$. Można udowodnić (patrz np Billingsley², że $(N(t))_{t \geq 0}$ jest procesem Poissona o intensywności λ wtedy i tylko wtedy gdy

1. liczby punktów w rozłącznych odcinkach są niezależne,
2. liczba punktów w odcinku $(t, t + h]$ ma rozkład Poissona z średnią λh .

Zauważmy, że

$$\mathbb{P}(N(t, t + h] = 1) = \lambda h e^{-\lambda h} = \lambda h + o(h), \quad (1.2)$$

i dlatego parametr λ nazywa się intensywnością procesu Poissona. Inna interpretacja to $\mathbb{E} N(t, t + h] = \lambda h$, czyli λ jest średnią liczbą punktów na jednostkę czasu. Zauważmy, że tutaj ta średnia, którą często zwie się intensywnością λ , jest niezależna od czasu t .

Przypomnijmy, że w podrozdziale III.2 zauważyliśmy natępujący fakt. Niech $\tau_1, \tau_2, \dots$ będą niezależnymi zmiennymi losowymi o jednakowym rozkładzie $\text{Exp}(1)$ oraz

$$N = \#\{i = 1, 2, \dots : \tau_1 + \dots + \tau_i \leq \lambda\}.$$

Wtedy zmienna losowa N ma rozkład Poissona z średnią λ . Okazuje się, że tak naprawdę kolejne sumy niezależnych zmiennych losowych o jednakowym rozkładzie wykładniczym $\text{Exp}(\lambda)$ definiują coś więcej, czyli proces Poissona. A więc i -ty punkt ma współrzędną $A_i = \tau_1 + \dots + \tau_i$, natomiast proces liczący można zapisać

$$N(t) = \#\{i = 1, 2, \dots : \tau_1 + \dots + \tau_i \leq t\}, \quad t \geq 0.$$

Podamy teraz procedurę matlabowską generowania procesu Poissona w odcinku $(0, t_{\max}]$.

²Prawdopodobieństwo i miara

Procedura poipro

```

% poipro simulate Poisson process
% of intensity lambda.
%
% Inputs: tmax - simulation interval
%         lambda - arrival intensity
%
% Outputs: n - number of arrivals in (0,tmax],
%          arrtime - time points of arrivals,
%                  if n=0 then arrtime='empty'

arrtime=-log(rand)/lambda;
if (arrtime>tmax)
    n=0; arrtime='empty'
else
    i=1;
    while (arrtime(i)<=tmax)
        arrtime = [arrtime, arrtime(i)-log(rand)/lambda];
        i=i+1;
    end
    n=length(arrtime);      % arrival times t1,...,tn
end

```

1.2 Niejednorodny proces Poissona

Jednorodny w czasie proces Poissona nie zawsze oddaje dobrze rzeczywisty proces zgłoszeń, ze względu na to że intensywność jest stała. Istnieją potrzeby modelowania procesu zgłoszeń w którym intensywność zależy od momentu t . Na przykład w cyklu całodobowym jest naturalnym rozróżnienie intensywności przybyć ze względu na porę dnia. Zdefiniujemy teraz *niejednorodny proces Poissona* z *funkcją intensywności* $\lambda(t)$. Jeśli intensywność jednorodnego procesu Poissona λ jest $\mathbb{P}(N(dt) = 1) = \lambda dt$ (to jest inny zapis własności (1.2)), to w przypadku procesu niejednorodnego chcemy $\mathbb{P}(N(dt) = 1) = \lambda(t) dt$. Formalnie definiujemy niejednorodny proces Poissona z funkcją intensywności $\lambda(t)$ jako proces liczący, taki że

1. liczby punktów w rozłącznych odcinkach są niezależne,

2. liczba punktów w odcinku $(t, t + h]$ ma rozkład Poissona z średnią $\int_t^{t+h} \lambda(s) ds$.

Okazuje się, że jeśli w interesującym nas odcinku $(0, t_{\max}]$, $\lambda(t) \leq \lambda_{\max}$ to proces niejednorodny można otrzymać z procesu Poissona o intensywności $\lambda_{\max}$ przez niezależne *przerzedzanie* punktów jednorodnego procesu Poissona; jeśli jest punkt w punkcie t to zostawiamy go z prawdopodobieństwem $p(t) = \lambda(t)/\lambda_{\max}$. Mamy więc schemat dwukrokowy. Najpierw generujemy proces Poissona o stałej intensywności $\lambda_{\max}$ a następnie przerzedzamy każdy punkt tego procesu, niezależnie jeden od drugiego, z prawdopodobieństwami zależnymi od położenia danego punktu.

Podamy teraz procedurę matlabowską generowania procesu Poissona w odcinku $(0, t_{\max}]$.

Procedura nhpoipro

```
% nhpoipro simulate nonhomogeneous Poisson process
% of intensity lambda(t).
%
% Inputs: tmax - simulation interval
%         lambda - arrival intensity function lambda(t)
%
% Outputs: n - number of arrivals in (0,tmax],
%          arrtime - time points of arrivals,
%                  if n=0 then arrtime='empty'

arrt=-log(rand)/lambdamax;
if (arrt>tmax)
    n=0; arrtime='empty';
else
    i=1;
    while (arrt(i)<=tmax)
        arrtime=arrt;
        arrt = [arrt, arrt(i)-log(rand)/lambdamax];
        i=i+1;
    end
m=length(arrtime);
b= lambda(arrtime)/lambdamax;
```

```

c= rand(1,m);
arrtime=nonzeros(arrtime.*(c<b))
n=length(arrtime);
end

```

1.3 Proces odnowy

Jeśli odstępy między zgłoszeniami $\tau_1, \tau_2, \dots$ tworzą ciąg niezależnych nieujemnych zmiennych losowych o jednakowym rozkładzie, to mówimy że ciąg $A_1, A_2, \dots$ zdefiniowany przez (1.1) jest *procesem odnowy*. W szczególności, jeśli o rozkładzie wykładniczym, to mamy proces Poissona. Niestety taki proces, jeśli nie jest procesem POisson, nie ma własności stacjonarności przyrostów, tj. gdy wektory

$$(N(s + t_2) - N(s + t_1), \dots, N(s + t_n) - N(s + t_{n-1}))$$

mają rozkład niezależny od $s > 0$ ($n = 2, 3, \dots, 0 \leq t_1 < t_2 < \dots$).

1.4 Co to jest proces Poissona na E i jak się go symuluje

Podamy teraz co się rozumie przez proces punktowy na $E \subset \mathbb{R}^d$. Niech E będzie przestrzenią stanów. Jak zwykle $(\Omega, \mathcal{F}, \mathbb{P})$ jest podstawową przestrzenią probabilistyczną. W przypadku procesów Poissona z poprzednich podrozdziałów $E = R_+$ lub $E = [0, t_{\max}]$. Mówimy, że $N(\cdot)$ jest procesem punktowym na E jeśli dla każdego B $N(B)$ jest zmienną losową przyjmującą wartości $\{0, \dots, \infty\}$ i dla każdego ω , $N_\omega(\cdot)$ jest miarą.

Mówimy, że proces punktowy N na E jest procesem Poisson z miarą intensywności Λ jeśli

P1 dla rozłącznych $B_1, \dots, B_n \in E$ zmienne losowe $N(B_1), \dots, N(B_n)$ są niezależne,

P2 $N(B) \sim Poi(\Lambda(B))$.

Matematyczna konstrukcja procesu Poissona daje narzędzia do symulacji procesu Poisson na dowolnej przestrzeni.

Konstrukcja procesu Poissona na E . Zakładamy $\Lambda(E) < \infty$.

Krok 1°. Wylosuj ν o rozkładzie Poissona $\text{Poi}(\Lambda(E))$.

Krok 2°. Jeśli $\nu = k$ wylosuj niezależnie k punktów $\xi_1, \dots, \xi_k$ z E z rozkładu $\Lambda(dx)/\Lambda(E)$.

Położmy $N(dx) = \sum_{j=1}^{\nu} \mathbf{1}(\xi \in dx)$. Proces punktowy N jest szukany procesem Poissona z miarą intensywności Λ .

Założmy, że $\Lambda(E) = \infty$, ale istnieją rozłączne $E_1, \dots, E_m$ takie, że $E = E_1 \cup \dots \cup E_m$ i $\Lambda(E_j) < \infty$ dla $j = 1, \dots, m$. Wtedy na każdej podprzestrzeni E_i konstruujemy proces Poissona z miarą intensywności $\Lambda_i(\cdot) = \Lambda(\cdot \cap E_i)$. Ważne jest aby przy konstrukcji przestrzegano niezależności pomiędzy przestrzeniami. Proces Poissona N na E jest sumą poszczególnych procesów Poissona na E_i .

2 Metoda dyskretnej symulacji po zdarzeniach

Wiele procesów stochastycznych modelujących interesujące nas ewolucje ma prostą strukturę. Przestrzeń stanów takiego procesu E jest przeliczalna a proces zmienia stan od czasu do czasu. Moment zmiany stanu nazywamy tutaj 'zdarzeniem'. Tak więc aby symulować system musimy przechodzić *od zdarzenia do zdarzenia*. Taka metoda nazywa się w języku angielskim *discrete event simulation*. Zadania które możemy symulować w ten sposób spotykamy w wielu teoriach, między innymi

- zarządzania produkcją,
- zapasów,
- systemów telekomunikacyjnych,
- niezawodności.

Przykład 2.1 Najprostszym przykładem jest symulacja procesu *on-off*, który przyjmuje wartości $E = \{0, 1\}$. Jest to proces $L(t)$ alternujący pomiędzy 1 (stan *on* – włączenia) i 0 (stan *off* – wyłączenia). Kolejne czasy włączeń i wyłączeń są niezależne, czasy włączeń są o tym samym rozkładzie F_{on} i czasy włączeń są o tym samym rozkładzie F_{off} . Zdarzeniami są kolejne momenty włączeń i wyłączeń. Zakładamy niezależność wszystkich czasów *on* i *off*. Jednym z wskaźników jest tak zwany współczynnik gotowości – stacjonarne prawdopodobieństwo tego, że system jest sprawny. Jeśli $\mathbb{E}T^{\text{on}} < \infty$ jest

średnim czasem bycia w stanie *on* a $\mathbb{E} T^{\text{off}} < \infty$ jest średnim czasem bycia w stanie *off* to wiadomo jest, że stacjonarny współczynnik gotowości wynosi $\mathbb{E} T^{\text{on}} / (\mathbb{E} T^{\text{on}} + \mathbb{E} T^{\text{off}})$. Symulacja $L(t)$ jest bardzo prosta. Na zmianę musimy generować T^{on} i T^{off} tak długo aż przekroczymy horyzont $t_{\max}$. Teraz możemy obliczyć estymator

$$\hat{\rho}(t_{\max}) = \frac{1}{t_{\max}} \int_0^{t_{\max}} L(s) ds.$$

Można pokazać, że jeśli dystrybuanty F_{on} , F_{off} są ciągłe, to estymator $\hat{\rho}(t)$ współczynnika ρ jest zgodny, ale nie jest on nieobciążony. Jeśli jednakże proces $L(t)$ zaczniemy losowo,

$$\mathbb{P}(L(t) = 1) = \mathbb{E} T^{\text{on}} / (\mathbb{E} T^{\text{on}} + \mathbb{E} T^{\text{off}})$$

oraz

$$\mathbb{P}(L(t) = 0) = \mathbb{E} T^{\text{off}} / (\mathbb{E} T^{\text{on}} + \mathbb{E} T^{\text{off}})$$

to estymator $\hat{\rho}(t_{\max})$ byłby nieobciążony, ale oczywiście taki przypadek wymaga znajomości $I = \rho$

Przykład 2.2 [Zagadnienie konserwatora] W urządzeniu jest N elementów, które mają bezawaryjny czas pracy o rozkładzie F ; jeśli któryś z elementów popsuje się, to czas naprawy przez konserwatora ma rozkład G . Załóżmy, że jest jeden konserwator. Jeśli więc jest on zajęty, to pozostałe pepsute muszą czekać. Przyjmujemy, że wszystkie zmienne są niezależne. Możemy się pytać, na przykład o liczbę niesprawnych elementów $L(t)$ w momencie $t \leq t_{\max}$. Jest to oczywiście zmienna losowa dla której możemy chcielibyśmy policzyć funkcję prawdopodobieństwa lub w szczególności średnią $\mathbb{E} L(t)$. Przy odpowiednich założeniach regularności ta wielkość dąży do pewnej stałej l , która jest oczekiwaną stacjonarną liczbą niesprawnych elementów. Możemy się również pytać o czas B do pierwszego momentu gdy wszystkie maszyny są zepsute. Ile wynosi $\mathbb{E} B$? Jeśli rozkłady F i G są dowolne, to nie znamy wzorów na postawione pytania. A więc pozostaje symulacja. Zdarzeniami są w momenty awarii i naprawy elementu. W przypadku awarii liczba niesprawnych zwiększa się o jeden natomiast naprawy zmniejsza się o jeden. Estymatorem l jest

$$\hat{l} = \frac{1}{t_{\max}} \int_0^{t_{\max}} L(s) ds.$$

Ogólny opis metody jest następujący. Przypuśćmy, że chcemy symulować $\mathbf{n}$ – stan systemu do momentu $t_{\max}$ – horyzont symulacji. Jedyne stochastyczne zmiany systemu są możliwe w pewnych momentach $A_1, A_2, \dots$. Pomiędzy tymi momentami system zachowuje się deterministycznie (przyjmujemy, że jest stały chociaż istnieją modele pozwalające opuścić to założenie – tzw. procesy kawałkami deterministyczne Markowa). Przypuśćmy, że mamy N typów zdarzeń $\mathcal{A}_i$ ($i = 1, \dots, N$). Pomysł polega na utworzeniu listy zdarzeniowej LZ w której zapisujemy momenty następnych zdarzeń po momencie obecnego; zdarzenie $\mathcal{A}_i$ będzie w momencie $t_{\mathcal{A}_i}$. Jeśli któreś zdarzenie nie może w następnym kroku nastąpić, to w liście zadaniowej nie będzie zarejestrowany jego moment pojawienia się. Tak może być na przykład z momentami zgłoszeń do systemu po zadanym horyzoncie $t_{\max}$.

Algorytm znajduje najbliższe zdarzenie, podstawia nową zmienną czasu, wykonuje odpowiednie aktywności, itd. Oczywiście aby symulacja była możliwa muszą być poczybione odpowiednie założenia niezależności.

Przykład 2.3 Narysujemy teraz realizację systemu kolejkowego M/M/1 z intensywnością zgłoszeń λ i intensywnością obsługi μ . Interesuje nas proces $L(t)$ liczby zadań w systemie w chwili t . Odsyłamy do dodatku ([????]) po informacje z teorii kolejek, w szczególności systemu M/M/1. Realizację takiego procesu można otrzymać korzystając z symulacji po zdarzeniach, jednakże my tutaj wykorzystamy ze specjalnej struktury procesów narodzin i śmierci. Proces $L(t)$ jest bowiem procesem narodzin i śmierci (B&D) z intensywnością narodzin λ i intensywnością śmierci μ . Korzystając więc z teorii procesów B&D (patrz dodatek XI.7.2) wiemy, że

- jeśli system jest niepusty to czasy do kolejnych skoków są wykładnicze z parametrem $\lambda + \mu$, skok górę jest z prawdopodobieństwem $\lambda/(\lambda + \mu)$, natomiast w dół z prawdopodobieństwem $\mu/(\lambda + \mu)$,
- jeśli system jest pusty to najbliższy skok (do jedynki) jest po czasie wykładniczym z parametrem λ .

Podamy teraz prosty algorytm na generowanie realizacji długości kolejki gdy $\lambda = 8/17$ i $\mu = 9/17$ aż do 100-nego skoku. Zauważmy, że tutaj $\lambda + \mu = 1$, i że $L(0) = 5$.³ Do wykreślenia realizacji będziemy używać matlabowskiego polecenia

³mm1v1.m

`stairs (x,y)`

gdzie $x = (x_1, \dots, x_n)$, $y = (y_1, \dots, y_n)$ które przy założeniu, że $x_1 < \dots < x_n$, wykreśla łamaną łączącą punkty

$$(x_1, y_1), (x_2, y_1), (x_2, y_2), \dots, (x_n, y_n).$$

```
clc;clear;
t=0;
s=5;
jumptimes=[t];
systemsizes=[s];
for i=1:1000
    if s>0
        t=t-log(rand);
        s=s+(2*floor((17/9)*rand)-1);
    else
        t=t-17*log(rand)/8;
        s=1;
    end
    jumptimes=[jumptimes,t];
    systemsizes=[systemsizes,s];
end
stairs(jumptimes,systemsizes)
```

Na rysunkach 2.1 i 2.2 są przykładowe symulacje jednej realizacji dla wybranych parametrów λ i μ .

Zmodyfikujemy teraz algorytm tak aby otrzymać realizację na odcinku $[0, t_{\max}]$. Realizując algorytm otrzymamy dwa ciągi $(0, t_1, \dots, t_{\max})$, $(s_0, s_1, \dots, s_{\max})$, które następnie służą do wykreślenia realizacji za pomocą polecenia “stairs”.

2.1 Modyfikacja programu na odcinek $[0, t_{\max}]$.

```
clc;clear;
t=0;
s=5;
tmax=1000;
jumptimes=[t];
systemsizes=[s];
```

```

until t<tmax;
    if s>0
        t=t-log(rand);
        s=0,s+(2*floor((17/9)*rand)-1);
    else
        t=t-7*log(rand)/8;
        s=1;
    end
    jumptimes=[jumptimes,t];
    systemsizes=[systemsizes,s];
end
jumptimes = [jumptimes, tmax];
systemsizes = [systemsizes,s];
stairs(jumptimes,systemsizes)

```

Teraz zastanowimy się jak obliczyć $\int_0^{t_{\max}} L(s) ds$ w celu obliczenia średniej długości kolejki

$$\text{srednia} = \frac{\int_0^{t_{\max}} L(s) ds}{t_{\max}}.$$

4

```

clc;clear;
t=0;
s=5; Lint=0;tdiff=0;
tmax=1000000;
while t<tmax
    told=t;
    if s>0
        t=t-log(rand);
        tnew=min(t,tmax);
        tdiff=tnew-told;
        Lint=Lint+tdiff*s;
        s=s+(2*floor((17/9)*rand)-1);
    else
        t=t-17*log(rand)/8;
        s=1;
    end
end

```

⁴mmlsredniav2

```

    end
end
srednia=Lint/tmax

```

Przeprowadzono eksperyment otrzymując następujące wyniki

tmax=1000;srednia =8.2627

tmax=100000;srednia = 8.1331

tmax=10000000;srednia = 7.9499

Z teorii (patrz dodatek [???) wiadomo, że z prawdopodobieństwem 1

$$\lim_{t \rightarrow \infty} \frac{\int_0^t L(s) ds}{t} = 8.$$

W rozdziale VII (patrz przykład [??ex.mm1.analiza??]) zastanowimy się jak odpowiednio dobrać $t_{\max}$.

Ogólny algorytm dyskretnej symulacji po zdarzeniach

INICJALIZACJA

```

podstaw zmienną czasu równą 0;
ustaw parametry początkowe 'stanu systemu' i liczników;
ustaw początkową listę zdarzeniową LZ;

```

PROGRAM

```

podczas gdy zmienna czasu < tmax;
wykonaj
    ustal rodzaj następnego zdarzenia;
    przesun zmienną czasu;
    uaktualnij 'stan systemu' & liczniki;
    generuj następne zdarzenia & uaktualnij listę zdarzeniową LZ
koniec

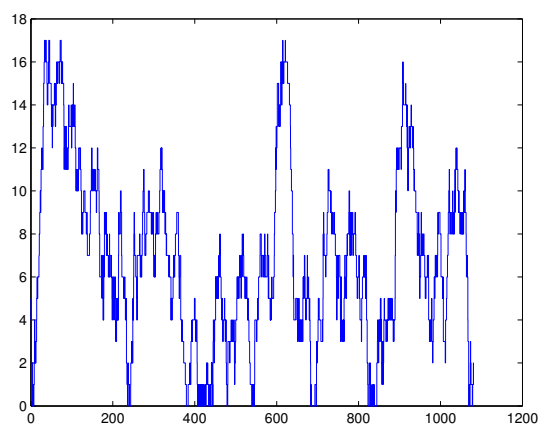
```

OUTPUT

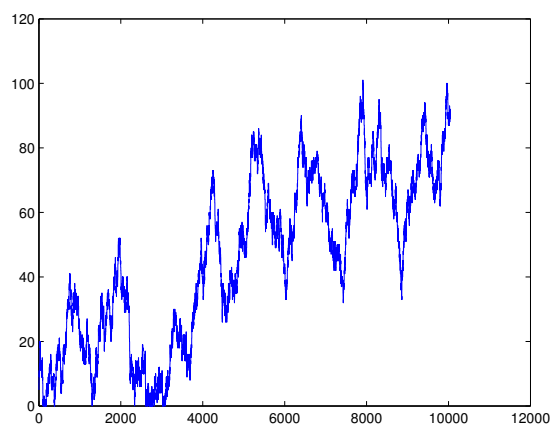
```

Obliczyć estymatory & przygotować dane wyjściowe;

```



Rysunek 2.1: Realizacja liczby zadań; $\lambda = \mu = .5$.



Rysunek 2.2: Realizacja liczby zadań; $\lambda = 8/17, \mu = 9/17$.

2.2 Symulacja kolejki z jednym serwerem

Metodę zilustrujemy na przykładzie algorytmu symulacji liczby zadań w kolejce GI/G/1. W dodatku XI.6 znajdują się fakty z teorii kolejek, a w szczególności opis systemu z jednym serwerem typu GI/G/1. Rozpatrujemy system w przedziale czasowym $[0, t_{\max}]$. Dalej będziemy pisać $t_{\max} = t_{\max}$. Interesują nas dwie ewolucje:

- proces liczby zadań $L(t)$ w systemie w chwili t ; $0 \leq t \leq t_{\max}$,
- ciąg czasów odpowiedzi $D(i)$.

Metoda opisana poniżej jest odpowiednia dla symulacji długości kolejki, a przy okazji zbadania czasu odpowiedzi. W przypadku symulacji jedynie kolejnych czasów odpowiedzi, dla systemu GI/G/1 z dyscypliną FIFO możemy wykorzystać prostą rekurencję podaną w dodatku w wzorze (XI.7.11).

Zajmiemy się więc symulacją procesu liczby zadań $L(t)$ w systemie w chwili t . Zakładamy, że dyscyplina obsługi jest zachowująca pracę, tj. w momencie skończenia obsługi zadania przez serwer, nie jest ważne które zadanie jest wybierane do obsługi (o ile w tym momencie czekają przynajmniej dwa zadania) ale tylko to, że serwer natychmiast podejmuje pracę.

Dla systemu GI/G/1 mamy dwa zdarzenia:

- *zgłoszenie* – $\mathcal{A}_1 = A$,
- *odejście* – $\mathcal{A}_2 = O$.

Poszczególne elementy algorytmu zależą od stanu *listy zdarzeniowej* – LZ, tj. od wielkości tA, tO – czasów następnego odpowiednio zgłoszenia i odejścia. Możliwe stany listy zdarzeniowej LZ to $\{tA\}, \{tO\}, \{tA, tO\}, \emptyset$.

Będziemy oznaczać przez

- t – zmienną czasu,
- n – zmienną systemową; liczba zadań w systemie,

Dane zbieramy (output variables) w:

- $t(i), n(i)$ – zmienne otrzymane (output variables) – moment kolejnego i -tego zdarzenia oraz stan systemu tuż przed tym momentem; zbierane w macierzy JS (jump times, system sizes). Początkowe wartości JS to $t = 0$ oraz stan początkowy s .

Niech F będzie dystrybuantą odstępu τ_i między zgłoszeniami oraz G dystrybuantą rozmiaru zadania. Na przykład w systemie M/M/1, F jest rozkładem wykładniczym z parametrem λ natomiast G jest rozkładem wykładniczym z parametrem μ . Zauważmy, że na podstawie znajomości $JS = \{(t(j), n(j))\}$ możemy wykreślić realizację $L(s)$, $0 \leq s \leq tmax$.

Podamy teraz algorytm na symulację stanów kolejki z jednym serwerem. Będziemy zakładać, że rozkłady F i G są ciągłe. Na początku przyjmujemy, że w chwili $t = 0$ system jest pusty ($n=0$). Jeśli $n = 0$, to odpowiada mu przypadek $LZ=\{tA\}$ jeśli $tA < t_{\max}$, lub $LZ = \text{empty}$ w przeciwnym razie. Na rysunku 2.3 ilustrujemy pierwsze kroki algorytmu dla przypadku gdy $tA < t_{\max}$ i pierwszy rozmiar zadania S_1 jest mniejszy od odstępu pomiędzy zgłoszeniem zadania pierwszego i drugiego.

Algorytm

POCZĄTEK (gdy w chwili 0 system jest pusty).

utwórz wektory/macierze:

JS -- macierz dwu-wierszowa; JS=[0;0]

Losujemy Tau o rozkładzie F.

Jeśli $\text{Tau} > t_{\max}$, to $LZ = \text{emptyset}$.

Jeśli $\text{Tau} \leq t_{\max}$,

to dołączamy do JS $t=n=0$,

oraz podstawiamy $tA = \text{Tau}$ i $LZ = \{tA\}$.

Następna część zależy już od stanu listy LZ.

PRZYPADEK 1. $LZ = \{tA\}$.

dołącz do JS: $t := tA$, $n := n+1$

generuj: zm. los. Tau o rozkładzie F do najbliższego zgłoszenia

generuj: zm. los. S o rozkładzie G (rozmiar następnego zadania)

resetuj: $t_0 = t + S$

resetuj: jeśli $t + \text{Tau} > t_{\max}$ to: nowy $LZ = \{t_0\}$

resetuj: jeśli $t + \text{Tau} \leq t_{\max}$, to $tA = t + \text{Tau}$; nowy $LZ = \{tA, t_0\}$

PRZYPADEK 2. $LZ = \{tA, t_0\}$, gdzie $tA \leq t_0$.

dołącz do JS: $t = tA$, $n = n+1$

generuj: Tau o rozkładzie F

resetuj: jeśli $t + \text{Tau} > t_{\max}$ to podstaw t_0 ; nowy $LZ = \{t_0\}$

resetuj: jeśli $t + \text{Tau} \leq t_{\max}$

to podstaw $tA = t + \text{Tau}$; nowy $LZ = \{tA, t_0\}$

PRZYPADEK 3. $LZ=\{t_A, t_0\}$, gdzie $t_A > t_0$.

dołącz do JS: $t=t_0$, $n=n-1$
 jeśli $n>0$: jeśli to generuj: Y o rozkładzie G i resetuj $t_0=t+Y$,
 nowa lista zdarzeniowa $LZ=\{t_A, t_0\}$
 jeśli $n=0$ to nowa $LZ=\{t_A\}$

PRZYPADEK 4: $LZ=\{t_0\}$

dołącz do JS: $t=t_0$, $n=\max(0, n-1)$, nowa lista zdarzeniowa
 $LZ=\{\text{emptyset}\}$.

PRZYPADEK 5: $LZ=\{\text{emptyset}\}$

dołącz do JS: $t=t_{\max}$, $n=n$.

KONIEC -- skończ symulację.

Teraz zrobimy kilka uwag o rozszerzeniu stosowalności algorytmu.

Dowolne warunki początkowe W przypadku, gdy w chwili 0 system nie jest pusty, musimy zadać/znać liczbę zadań w systemie w chwili 0, czas do zgłoszenia t_A oraz czas do zakończenia obsługi t_O . To znaczy musimy zadać listę zdarzeniową $\{t_A, t_O\}$ oraz zmienną 'n'. Zauważmy, że w najogólniejszym przypadku gdy system w chwili 0 jest już od pewnego czasu pracujący, to (n, t_A, t_O) jest wektorem losowym o zadanym rozkładzie. Oczywiście możemy przyjąć, że (n, t_A, t_O) jest deterministyczne. W ogólnym przypadku podanie tego rozkładu jest sprawą bardzo trudną. Ale na przykład w przypadku systemu M/M/1, t_A ma rozkład wykładniczy z parametrem λ (jeśli $n > 0$) oraz t_O ma rozkład wykładniczy z parametrem μ . Jest to konsekwencja braku pamięci rozkładu wykładniczego; patrz XI.5.6.

2.3 Czasy odpowiedzi, system z c serwerami

Wyobraźmy sobie, że do systemu wpływają zadania – k -te w chwili A_k rozmiaru S_k . Jest c serwerów. Serwer i -ty ($i = 1, \dots, c$) pracuje z szybkością $h(i)$. Bez straty ogólności możemy założyć, że $h(1) \geq h(2) \geq \dots \geq h(c)$.

A więc zadanie rozmiaru S_k na i -tym serwerze jest obsługiwane przez $S_k/h(i)$ jednostek czasu. Teraz będzie ważna dyscyplina obsługi – *zgodna z kolejnością zgłoszeń*. Zauważmy, że w przypadku większej liczby kanałów nie ma równoważności z dyscypliną FIFO. W przypadku gdy kilka serwerów jest pustych, zostaje wybrany najszybszy, czyli z najmniejszym numerem. Interesującą nas charakterystyką jest czas odpowiedzi (response time) – czas od momentu zgłoszenia zadania do opuszczenia przez niego systemu.

W tym wypadku mamy $c + 1$ zdarzeń: *zgłoszenie, odejście z pierwszego serwera, ..., odejście z c -tego serwera*. Poszczególne elementy algorytmu zależą od stanu *listy zdarzeniowej* – LZ, tj. od wielkości $t_A, t_O(1), \dots, t_O(c)$ – czasów następnego odpowiednio zgłoszenia i odejścia z odpowiedniego serwera. W liście zdarzeniowej musimy też przechowywać razem z $t_O(i)$ numer $K(i)$ zadania, które w tym momencie opuszcza system. W przypadku $c = 1$ wystarczyłoby rejestrowanie kolejnych momentów głośnień A_n i kolejnych momentów odejść z systemów, ponieważ w tym przypadku kolejność zgłoszeń odpowiada kolejności odejść. Przez krok będziemy rozumieli przejście od jednego zdarzenia do następnego.

Możliwe stany listy zdarzeniowej LZ są podzbiorami $\{t_A, t_O(1), \dots, t_O(c)\}$ (włącznie ze zbiorem pustym $\emptyset$).

Będziemy potrzebować

- $n_{\max}$ – liczba zadań symulowanych,

oraz

- n – zmienna systemowa; liczba zadań w chwili t ,
- NK – numer aktualnie czekającego na obsługę zadania,
- NA – liczba zgłoszeń do aktualnego kroku,
- NO – liczba odejść do aktualnego kroku.

Będziemy też zbierać następujące dane (output variables):

- $(A(i), O(i))$ – zmienne otrzymane (output variables) – moment zgłoszenia oraz moment wykonania i -tego zadania; zbierane w wektorach A , O .

Niech F będzie dystrybuantą odstępu τ między zgłoszeniami oraz G dystrybuenta rozmiaru zadania S . Zakładamy, że dystrybuanty F, G są ciągłe. Zauważmy:

- Na podstawie $A(i), O(i)$ możemy znaleźć ciąg czasów odpowiedzi $D(i) = O(i) - A(i)$.

3 ALOHA

W chwilach n ($n = 0, 1, 2, \dots$) 'użytkownicy' komunikują się przez wspólnie użytkowany kanał. Kiedy użytkownik ma wiadomość do nadania może ją przesyłać przez kanał. Wiadomość jest podzielona na pakiety jednakowej długości, które wysyłane są w 'szczelinach' (slots) czasowych (tej samej długości co pakiety). Jeśli w tej samej szczelinie chce transmitować więcej niż jeden użytkownik, wtedy następuje 'kolizja' o której dowiadują się użytkownicy i transmisja jest nieudana. Udana transmisja jest wtedy i tylko wtedy gdy dokładnie jeden pakiet jest transmitowany w szczelinie. Oczywiście, gdyby w następnej szczelinie założyć, że dotychczasowi użytkownicy będą znowu próbować nadawać, to nawet gdyby nie pojawiły się nowe wiadomości, na pewno nastąpiłaby kolizja. Aby rozwiązać ten problem i móc nadawać w takim kanale opracowano rozmaite protokoły, na przykład tak zwany *szczelinowa ALOHA* slotted ALOHA. W tych protokołach zakłada się, że użytkownicy mogą jedynie słyszeć kolizje, ale nie znają liczby pakietów oczekujących u poszczególnych użytkowników. Jeśli więcej niż jedna stacja zdecyduje się transmitować w tej samej szczelinie, wtedy mamy kolizję i żaden pakiet nie został przesłany, i wszyscy muszą czekać do momentu $n + 1$. Jeśli natomiast tylko jedna stacja zdecyduje się na transmisję, to trwa ona 1 jednostkę czasu i kończy się sukcesem.

Pomysł jest taki, żeby nadawanie pakietu po kolizji było opóźnione. Jak to zrobić zależy od protokołu.

Aby móc zbadać protokoły zbudujemy model matematyczny. Oś czasowa jest podzielona na szczeliny; i -ta jest $[i, i + 1)$, $i = 0, 1, \dots$. Aby opisać model potrzebujemy następujące składniki:

- X_n – liczba pakietów czekających na transmisję na początku szczeliny n -tej
- A_n – liczba nowych pakietów gotowych do nadawania na początku szczeliny n -tej.
- $\{h_i, 1 \leq i \leq K\}$ – strategia nadawania, gdzie $0 < h_i < 1$, $K \leq \infty$

Powyżej $0 < h_i < 1$ oznacza, że po i nieudanych transmisjach pakiet jest nadawany w następnej szczelinie z prawdopodobieństwem h_i . Zakłada się, że $\{A_n\}$ są niezależne o jednakowym rozkładzie z wspólną funkcją prawdopodobieństwa $\mathbb{P}(A_n = i) = a_i$, $i = 0, 1, \dots$ z $\mathbb{E} A_n = \lambda < \infty$. Możemy rozpatrywać następujące strategie retransmisji.

- $h_i = h$ – szczelinowa ALOHA; $h_i = h$
- $h_i = 2^{-i}$ – ETHERNET (exponential back-off).

Niech

- Z_n – liczba pakietów podchodzących do transmisji w i -tej szczelinie.

Zauważmy, że

$$X_{i+1} = X_i + A_i - \mathbf{1}(Z_i = 1), \quad i = 0, 1, \dots \quad (3.3)$$

Niech

$$N_n^s = \sum_{i=1}^n \mathbf{1}(Z_i = 1), \quad n = 1, \dots$$

będzie liczbą pakietów przetransmitowanych w pierwszych n szczelinach; $n = 1, 2, \dots$. Definiujemy *przepustowość* (*throughput*) jako

$$\lim_{n \rightarrow \infty} \frac{N_n^s}{n} = \gamma.$$

Zauważmy, że w protokole szczelinowa ALOHA, rozkład Z_n pod warunkiem $X_n = k$ jest taki sam jak $A + B(k)$, gdzie $\{B(k)\}$ jest zmienną losową niezależną od A z funkcją prawdopodobieństwa

$$\mathbb{P}(B_n(k) = i) = b_i(k) = \binom{k}{i} h^i (1-h)^{k-i}, \quad i = 0, 1, \dots, k$$

Warunkowy dryf definiuje się jako

$$\mathbb{E}[X_{n+1} - X_n | X_n = k] = \lambda - b_1(k)a_0 - b_0(k)a_1. \quad (3.4)$$

Gdy $k \rightarrow \infty$ to

$$\mathbb{E}[X_{n+1} - X_n | X_n = k] \rightarrow \lambda > 0$$

co sugeruje, że jak ciąg osiągnie wysoki pułap to ma dryf do nieskończoności. W teorii łańcuchów Markowa formalnie dowodzi się chwilowości ciągu (X_n) .

Do symulacji będziemy zakładać, że A_n mają rozkład $\text{Poi}(\lambda)$. Przykładowa symulacja (X_n) jest pokazana na rys. 3.4.

Natomiast na rys. 3.5 pokazane są lokalne przepustowości

$$B_n = \sum_{i=1}^n \mathbf{1}(Z_i = 1)/n; \quad n \leq N.$$

4 Prawdopodobieństwo awarii sieci

Rozważmy sieć połączeń, która jest spójnym grafem składającym się z węzłów, z których dwa s i t są wyróżnione. Węzły są połączone krawędziami ponumerowanymi $1, \dots, n$. Każda krawędź może być albo sprawna – 0 lub nie – 1. W przypadku niesprawnej krawędzi będziemy mówili o przerwaniu. możemy też mówić o usunięciu krawędzi, i wtedy nowy graf nie musi być dalej spójny.

Stany krawędzi opisuje wektor $e_1, \dots, e_n$, gdzie $e_i = 0$ lub 1. Sieć w całości może być sprawna – 0 jeśli jest połączenie od s do t lub nie (t.j. z przerwą) – 1, tzn graf powstały po usunięciu niesprawnych krawędzi nie jest spójny. Niech $\mathcal{S} = \{0, 1\}^n$ i $\mathbf{e} = (e_1, \dots, e_n)$. Formalnie możemy opisać stan systemu za pomocą funkcji $k : \mathcal{S} \rightarrow \{0, 1\}$. Mówimy, że sieć jest niesprawna i oznaczamy to przez 1 jeśli nie ma połączenia pomiędzy węzłami s i t , a 0 w przeciwnym razie. To czy sieć jest niesprawna (lub sprawna) decyduje podzbiór B niesprawnych krawędzi, tj $e_j = 1, j \in B, e_j = 0, j \notin B$. Podamy teraz przykłady. Oznaczmy $|\mathbf{e}| = \sum_{j=1}^n e_j$.

- *Układ szeregowy* jest niesprawny gdy jakakolwiek krawędź ma przerwanie. A więc $k(\mathbf{e}) = 1$ wtedy i tylko wtedy gdy $|\mathbf{e}| \geq 1$.
- *Układ równoległy* jest niesprawny jeśli wszystkie elementy są niesprawne. A więc $k(\mathbf{e}) = 1$ wtedy i tylko wtedy gdy $|\mathbf{e}| = n$.

Teraz przypuśćmy, że każda krawędź może mieć przerwanie z prawdopodobieństwem q niezależnie od stanu innych krawędzi. Opiszemy to za pomocą wektora losowego X przyjmującego wartości z $\mathcal{S}$. Zauważmy, że X jest dyskretną zmienną losową z funkcją prawdopodobieństwa

$$\mathbb{P}(X = \mathbf{e}) = p(\mathbf{e}) = q^{\sum_{j=1}^n e_j} (1 - q)^{n - \sum_{j=1}^n e_j},$$

dla $\mathbf{e} \in \mathcal{S}$. Naszym celem jest obliczenie prawdopodobieństwa, że sieć jest niesprawna. A więc

$$I = \mathbb{E} k(X) = \sum_{\mathbf{e} \in \mathcal{S}} k(\mathbf{e}) p(\mathbf{e})$$

jest szukanym prawdopodobieństwem, że sieć jest niesprawna. Z powyższego widzimy, że możemy używać estymatora dla I postaci

$$\hat{X}_n^{\text{CMC}} = \frac{1}{n} \sum_{i=1}^n k(X_i),$$

gdzie $X_1, \dots, X_n$ są niezależnymi replikacjami X . Ponieważ $k(X) = 0$ lub 1 ,

$$I = \mathbb{P}(k(X) = 1) = \mathbb{E} k(X) \quad \text{oraz} \quad \text{Var} k(X) = I(1 - I).$$

Proponujemy zrobić symulację dla konkretnego przykładu pochodzącego z książki Madrasa [?]. Sieć ma 11 węzłów połączonych 22 krawędziami i jest przedstawiona na rys. 4.6.

Ponieważ typowo prawdopodobieństwo I jest bardzo małe, więc aby wyliczyć szukane prawdopodobieństwo awarii musimy zrobić wiele replikacji. Dlatego jest bardzo ważne optymalne skonstruowanie dobrego estymatora dla I .

Zanalizujemy teraz estymator istotnościowy z $\tilde{p}(\mathbf{e}) = \beta^{|\mathbf{e}|}(1 - \beta)^{n-|\mathbf{e}|}$. Mamy

$$I = \sum_{\mathbf{e} \in \mathcal{S}} k(\mathbf{e}) p(\mathbf{e}) = \sum_{\mathbf{e} \in \mathcal{S}} \frac{k(\mathbf{e}) p(\mathbf{e})}{\tilde{p}(\mathbf{e})} \tilde{p}(\mathbf{e}).$$

Teraz zauważmy, że $k(\mathbf{e}) = 1$ wtedy i tylko wtedy gdy sieć jest niesprawna a w pozostałych przypadkach równa się zero. A więc

$$I = \sum_{\mathbf{e}: k(\mathbf{e})=1} \frac{p(\mathbf{e})}{\tilde{p}(\mathbf{e})} \tilde{p}(\mathbf{e}),$$

gdzie

$$\frac{p(\mathbf{e})}{\tilde{p}(\mathbf{e})} = \frac{q^{|\mathbf{e}|}(1 - q)^{n-|\mathbf{e}|}}{\beta^{|\mathbf{e}|}(1 - \beta)^{n-|\mathbf{e}|}} = \left(\frac{1 - q}{1 - \beta} \right)^n \left(\frac{q(1 - \beta)}{\beta(1 - q)} \right)^{|\mathbf{e}|},$$

gdzie $|\mathbf{e}| = \sum_{j=1}^n e_j$. Będziemy przyjmowali, że $\beta \geq q$ co pociąga

$$\frac{q(1 - \beta)}{\beta(1 - q)} \leq 1.$$

Niech teraz d będzie najmniejszą liczbą krawędzi z przerwaniem, powodująca niesprawność całej sieci. Na przykład dla sieci szeregowej d wynosi 1, a dla równoległej n . Dla sieci z Rys. 4.6 mamy $d = 3$. A więc, jeśli $k(\mathbf{e}) = 1$, to $|\mathbf{e}| \geq d$.

Przy założeniu $|\mathbf{e}| \geq d$

$$\begin{aligned} \left(\frac{1-q}{1-\beta}\right)^n \left(\frac{q(1-\beta)}{\beta(1-q)}\right)^{|\mathbf{e}|} &\leq \left(\frac{1-q}{1-\beta}\right)^n \left(\frac{q(1-\beta)}{\beta(1-q)}\right)^d, \quad |\mathbf{e}| \geq d \\ &= \frac{(1-q)^{n-d} q^d}{\beta^d (1-\beta)^{n-d}}. \end{aligned}$$

Teraz pytamy dla jakiego $\beta \geq q$ prawa strona jest najmniejsza. Odpowiedź na to pytanie jest równoważna szukaniu maksimum funkcji $\beta^d (1-\beta)^{n-d}$. Przyrównując pochodną do zera mamy

$$d\beta^{d-1}(1-\beta)^{n-d} - (n-d)\beta^d(1-\beta)^{n-d-1} = 0.$$

Stąd $\beta = d/n$. A więc dla $|\mathbf{e}| \geq d$ mamy

$$\left(\frac{1-q}{1-\beta}\right)^n \left(\frac{q(1-\beta)}{\beta(1-q)}\right)^{|\mathbf{e}|} \leq \left(\frac{1-q}{1-\frac{d}{n}}\right)^n \left(\frac{q(1-\frac{d}{n})}{\frac{d}{n}(1-q)}\right)^d = \delta.$$

Jeśli $d/n \geq q$, to oczywiście $\delta \leq 1$. Teraz

$$\begin{aligned} \tilde{\mathbb{E}} k^2(X) \frac{p^2(X)}{\tilde{p}^2(X)} &= \sum_{\{\mathbf{e}: k(\mathbf{e})=1\}} \frac{p^2(\mathbf{e})}{\tilde{p}^2(\mathbf{e})} \tilde{p}(\mathbf{e}) \\ &= \sum_{\{\mathbf{e}: \kappa(\mathbf{e})=1\}} \frac{p(\mathbf{e})}{\tilde{p}(\mathbf{e})} p(\mathbf{e}) \\ &= \sum_{\{\mathbf{e}: k(\mathbf{e})=1\}} \left(\frac{1-q}{1-\beta}\right)^n \left(\frac{q(1-\beta)}{\beta(1-q)}\right)^{|\mathbf{e}|} p(\mathbf{e}) \\ &\leq \delta \sum_{\{\mathbf{e}: |\mathbf{e}| \geq d\}} p(\mathbf{e}) = \delta I. \end{aligned}$$

Ponieważ I zazwyczaj jest małe, więc

$$\text{Var } \hat{Y}_n = \frac{I(1-I)}{n} \approx \frac{I}{n}.$$

Z drugiej strony

$$\text{Var } \hat{Y}_n^{\text{IS}} \sim \frac{\delta}{n} I.$$

A więc estymator IS ma $1/\delta$ razy mniejszą wariancję aniżeli by to robić za pomocą zgrubnego estymatora.

W szczególności przyjmijmy dla sieci z Rys. 4.6 prawdopodobieństwo przerwania $q = 0.01$. Dla tej sieci mamy $\beta = 3/22$ i stąd $\delta = 0.0053$. Czyli używając estymatora IS potrzebne będzie $\sqrt{1/\delta} \approx 14$ razy mniej replikacji.

Przykład 4.1 Obliczenie prawdopodobieństwa awarii sieci metodą warstw. Rozważmy schemat połączeń z rys. 4.6. Ponieważ typowo prawdopodobieństwo I jest bardzo małe, więc aby je obliczyć używając estymatora CMC musimy zrobić wiele replikacji. Dlatego jest bardzo ważne optymalne skonstruowanie dobrego estymatora dla I . Można na przykład zaproponować estymator warstwowy, gdzie j -tą warstwą $\mathcal{S}(j)$ jest podzbiór $\mathcal{S}(j)$ składających się z tych B takich, że $|B| = j$. Zauważmy, że

$$p_j = \mathbb{P}(X \in \mathcal{S}(j)) = \mathbb{P}(\text{dokładnie } j \text{ krawędzi ma przerwę}) = \binom{n}{j} q^j (1-q)^{n-j}.$$

Każdy $A \subset \mathcal{S}$ ma jednakowe prawdopodobieństwo $q^j (1-q)^{|\mathcal{S}|-j}$ oraz

$$\mathbb{P}(X = A | X \in \mathcal{S}(j)) = 1 / \binom{n}{j} \quad A \subset \mathcal{S}.$$

Aby więc otrzymać estymator warstwowy generujemy np_j replikacji na każdej warstwie $\mathcal{S}(j)$ z rozkładem jednostajnym.

5 Uwagi bibliograficzne

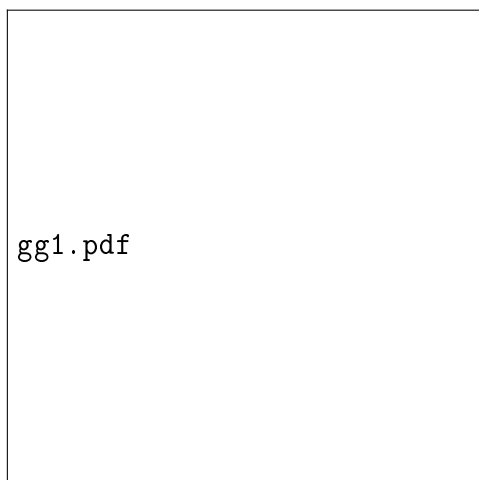
O procesach Poissona można przeczytać w wielu książkach z procesów stochastycznych. Jednakże warto polecić książki specjalistycznej, jak na przykład Kingmana [20, 21]. Metoda symulacji niejednorodnego procesu Poisson przez przedziedzanie pochodzi z pracy [32]. Metoda symulacji po zdarzeniach (discrete event simulation) ma bogatą literaturę. W szczególności można polecić podręcznika Fishmana [15], Bratleya i współautorów [7]. Jeśli procesy pomiędzy zdarzeniami nie są stałe ale zmieniają się w sposób deterministyczny, to takie procesy można modelować za pomocą procesów kawałkami

deterministycznie Markowa (piecewise deterministic Markov process); patrz monografia Davisa [10].

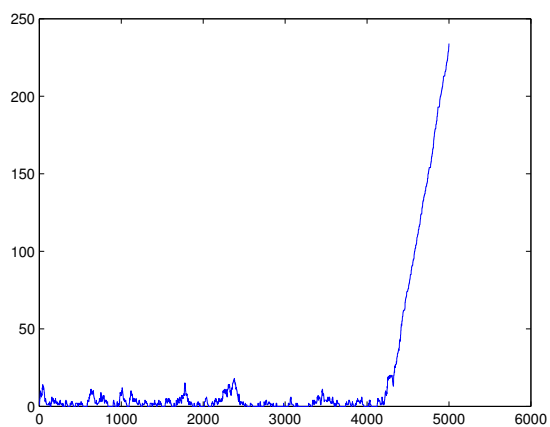
O zagadnieniu konserwatora i innych problemach teorii niezawodności można poczytać w książce Kopocińskiego [25]

O protokołach w transmisji rozproszonej można poczytać w książkach: Ross [1], str. 148, Bremaud [8], str. 108–110, 173–178, lub w pracach Kelly (1985) i Kelly & MacPhee [18, 19], Aldous (1987) [3].

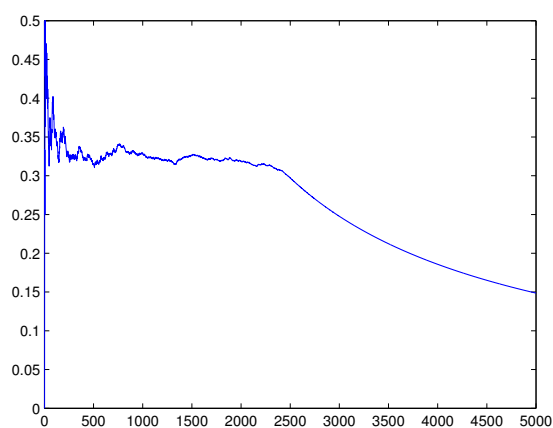
Analiza przepływu w sieci z rys. 4.6 jest wzięta z książki Madrasa [?].



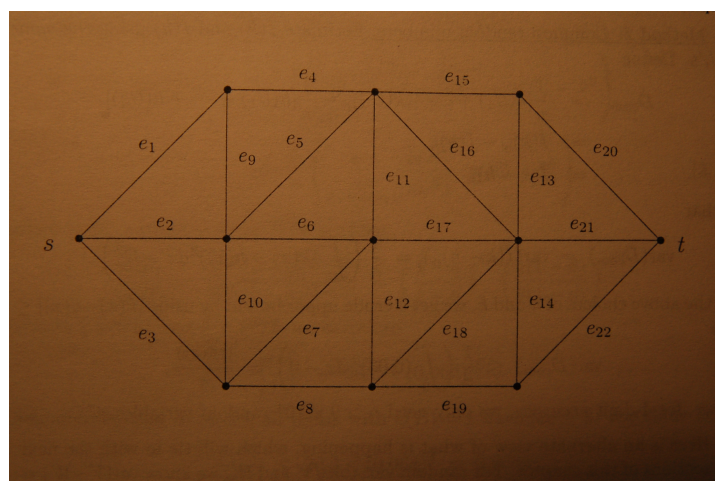
Rysunek 2.3: Ilustracja pierwszych kroków algorytmu dla $G/G/1$.



Rysunek 3.4: Liczba oczekujących pakietów X_n ($n \leq 5000$) w ALOHA;
 $\lambda = 0.31$, $h = 0.1$.



Rysunek 3.5: Szczelinowa ALOHA; Przepustowość B_n/n ($n \leq 5000$); $\lambda = 0.31$, $h = 0.1$.



Rysunek 4.6: Schemat połączeń w sieci.

Zadania teoretyczne

- 5.1 W szczelinowym modelu ALOHA z $\lambda = 0.31$ oraz $h = 0.1$ policzyć funkcję

$$\phi(k) = \mathbb{E}[X_{n+1} - X_n | X_n = k] = \lambda - b_1(k)a_0 - b_0(k)a_1, \quad k = 0, 1, \dots,$$

gdy A_i mają rozkład Poissona. Zrobić wykres. Czy można na podstawie tego wykresu skomentować rys. 3.4 i 3.5.

- 5.2 Dla sieci z rys. 4.6 oszacować dwustronnie I i następnie przeprowadzić analizę symulacji dla zgrubnego estymatora.

- 5.3 Dla przykładu 2.1 pokazać, że estymator $\hat{\rho}(t)$ jest zgodny.

- 5.4 Zastanowić się w jaki sposób optymalnie przeprowadzić symulację niejednorodnego procesu Poissona z

$$\lambda(t) = \begin{cases} 1 + \sin t & 0 \leq t \leq t_{\max}/ \\ 10 + \sin t & t_{\max} < t \leq t_{\max}. \end{cases}$$

Porównać z metodą z podrozdziału 1.2 oczekiwane liczby replikacji potrzebne do wygenerowania jednej realizacji procesu Poissona na $[0, t_{\max}]$

Zadania laboratoryjne

- 5.5 Wygenerować $A_1, A_2, \dots$, w odcinku $[0, 1000]$, gdzie $A_0 = 0$ oraz $A_1 < A_2 < \dots$ są kolejnymi punktami w niejednorodnym procesie Poissona z funkcją intensywności $\lambda(t) = a(2 - \sin(\frac{2\pi}{24}t))$. Przyjąć $a = 10$. Niech $A(t)$ będzie liczbą punktów w odcinku $[0, t]$. Zastanowić się jaki ma rozkład $A(t)$ i znaleźć jego średnią. Policzyć z symulacji $A(1000)/1000$ – średnią liczbą punktów na jednostkę czasu, zwaną asymptotyczną intensywnością $\bar{\lambda}$. Porównać z

$$\int_0^{24} \lambda(t) dt / 24.$$

- 5.6 Kontynuacja zad. refzad.proc.Poissona. Wygenerować $\tau_1, \tau_2, \dots, \tau_{1000}$, gdzie $\tau_i = A_i - A_{i-1}$, $A_0 = 0$ oraz $A_1 < A_2 < \dots$ są kolejnymi punktami w niejednorodnym procesie Poissona z funkcją intensywności $\lambda(t) = a(2 - \sin(\frac{2\pi}{24}t))$. Przyjąć $a = 10$. Obliczyć średni odstęp między punktami $\hat{\tau} = \sum_{j=1}^{1000} \tau_j / 1000$.

5.7 Zbadać jak szybko $P(L(t) = 1)$ w alternującym procesie *on – off* zbiega do współczynnika gotowości. Rozpatrzeć kilka przypadków:

- $F_{\text{on}}, F_{\text{off}}$ są wykładnicze,
- $F_{\text{on}}, F_{\text{off}}$ są Erlanga odpowiednio $\text{Erl}(3, \lambda)$ i $\text{Erl}(3, \mu)$,
- $F_{\text{on}}, F_{\text{off}}$ są Pareto.

Przyjąć $\mathbb{E} T^{\text{on}} = 1$ i $\mathbb{E} T^{\text{off}} = 2$.

5.8 Obliczyć średnią $\mathbb{E} L(i)$ ($i = 1, \dots, 10$) w systemie M/M/1 z

- a. $\lambda = 1/2$ i $\mu = 1$,
- b. $\lambda = 1$ i $\mu = 1$, jeśli $L(0) = 0$.

5.9 Przeprowadzić symulacje dobroci *czystego protokołu ALOHA*. Bardziej dokładnie, przypuśćmy, że pakiety przybywają zgodnie z procesem Poissona z intensywnością λ i wszystkie są długości 1. W przypadku kolizji, każdy z pakietów jest retransmitowany po czasie wykładniczym z parametrem μ . Zakładamy, że wszystkie zmienne są niezależne. W modelu musimy obliczyć następujące zmienne. Na podstawie symulacji zastanowić się nad przepustowością γ przy tym protokole.

5.10 Napisać algorytm na symulację protokołu ETHERNET.

5.11 Obliczyć średni czas do awarii systemu składającego się z N elementów połączonych równoległe z jednym konserwatorem, jeśli czas życia elementu ma rozkład wykładniczy $\text{Exp}(1/2)$, natomiast czas naprawy przez konserwatora $\text{Exp}(1)$. Policzyc dla $N = 1, \dots, 10$. Porównać z średnim czasem do awarii systemu składającego się z N elementów połączonych równoległe, ale bez konserwatora.

Projekt

Projekt 3 Zbadać dobroć protokołu szczelinowa ALOHA ze względu na parametr λ . Przez symulację znaleźć wartości krytyczne dla λ i określić typowe zachowanie się protokołu. Zbadać inną modyfikację protokołu szczelinowa ALOHA, w której dopuszcza się, że użytkownicy mają też dodatkową wiedzę o stanie X_n , i wtedy w n -tej szczelinie przyjmować $h = 1/X_n$. Czy ten protokół może być stabilny, tzn. mający przepustowość $\gamma > 0$; jeśli tak to kiedy.

Projekt 4A Produkt jest składany z komponentów na dwóch stanowiskach. Po zakończeniu prac na pierwszym stanowisku jest dalej opracowywany na drugim, z którego wychodzi gotowy produkt. Produkcja trwa każdego dnia przez 8 godzin. Na początku dnia w magazynie jest przygotowane n_1 komponentów, które sukcesywnie są dostarczane na stanowisko pierwsze. Przed stanowiskiem drugim może oczekiwać co najwyżej k_1 , w przeciwnym razie praca na stanowisku pierwszym jest zablokowana. Ile należy przygotować komponentów n_1 i jak duży bufor k_1 aby z prawdopodobieństwem nie mniejszym 0.9 była zapewniona produkcja przez cały dzień, tj. 8 godzin. Przez zaprzestanie produkcji rozumiemy, że zarówno magazyn jak i bufor przestają pracować. Jak duży należy przygotować magazyn $n_1 + k_1$. Przyjąć, że

- czas pracy na pierwszym stanowisku ma rozkład Erlanga $\text{Erl}(2,10)$,
- natomiast na drugim stanowisku jest mieszkanką rozkładów wykładniczych z gęstością $0.8 \times 9e^{-9x} + 0.2 \times 3e^{-3x}$,
- koszty magazynowania przed rozpoczęciem pracy czy w oczekiwaniu na prace na drugim stanowisku są liniowe od $n_1 + k_1$.

Projekt 4B Produkt jest składany z komponentów na dwóch stanowiskach. Po zakończeniu prac na pierwszym stanowisku jest dalej opracowywany na drugim, z którego wychodzi gotowy produkt. Produkcja trwa każdego dnia przez 8 godzin. Na początku dnia w magazynie jest przygotowane n komponentów, które sukcesywnie są dostarczane na stanowisko pierwsze (zakładamy, że nie ma przetojów). Przed stanowiskiem drugim może oczekiwać co najwyżej k produktów, w przeciwnym razie praca na stanowisku pierwszym jest zablokowana.

To znaczy, że jeśli w buforze przed stanowiskiem drugim jest k produktów, to wtedy maszyna pierwsza po skończeniu pracy nad produktem staje. Taki model nazywamy (n, k) .

1. Niech $k = 5$. Jak duże musi być n (podać minimalne) aby z prawdopodobieństwem nie mniejszym od 0.9 wyprodukować 35 produktów?
2. Ze względów technicznych przestój maszyny pierwszej jest kosztowny i musi być brany pod uwagę. Niech f_1 oznacza frakcję czasu w ciągu dnia gdy maszyna 1-sza nie pracuje. Zauważmy, że f_1 jest zmienną losową. Funkcja kosztu jest $C\mathbb{E} f_1 + k$. Oczywiście przy $k = n$ mamy $f_1 = 0$. Przyjmijmy $n = 120$. Niech $k_0(C)$ będzie k które minimizuje funkcję kosztu w zależności od C . Zbadać $k(C)$.
3. Ustalmy $n = 120$. Jaki duży musi być bufor k aby $\mathbb{P}(f_1 = 0) \geq 0.9$.

Przyjąć, że

- czas pracy na pierwszym stanowisku ma rozkład Erlanga $\text{Erl}(2, 10)$,
- natomiast na drugim stanowisku jest mieszkanką rozkładów wykładniczych z gęstością $0.8 \times 9e^{-9x} + 0.2 \times 3e^{-3x}$,

Projekt 5 Mamy dwa następujące warianty pracy 2-ch procesorów.

1. Dwa procesory pracują osobno. Zadania napływają do i -tego procesora zgodnie z procesem Poissona z intensywnością λ_i i mają rozkład rozmiaru $G^{(i)}$. Niech $D^{(i)}(t)$ będzie liczbą zadań obsłużonych przez i -ty procesor to chwili t . Zakładamy, że $\lambda_i < \int_0^\infty x dG^{(i)}(x)$, $i = 1, 2$. Wtedy przepustowość

$$\lim_{t \rightarrow \infty} D^{(i)}(t)/t = \lambda_i,$$

a więc dwa procesory, jeśli pracują osobno mają przepustowość $\lambda_1 + \lambda_2$.

2. Możemy tak zmienić konstrukcję, że w przypadku gdy jeden z procesorów nie ma pracy, to drugi pracuje z szybkością 2.

Zbadać o ile efektywniejszy jest zparowany procesor, w zależności od intensywności wejścia i rozkładów rozmiarów zadań. Przyjąć $\lambda_1 = 1$.

Projekt 6 W banku jest c stanowisk obsługi. Klienci zgłaszają się zgodnie z procesem Poissona z intensywnością λ i ich czasy obsługi mają rozkład G . Zbadać następujące protokoły kolejkowania.

1. Przed każdym okienkiem jest osobna kolejka. Klienci wybierają

okienko w sposób losowy.

2. Przed każdym okienkiem jest osobna kolejka. Klienci wybierają kolejne okienko, tj. $1, 2, \dots, c, 1, 2, \dots$

3. Jest jedna kolejka i gdy nadchodzi czas klient idzie do akurat uwolnionego okienka.

Rozdział VII

Analiza symulacji stabilnych procesów stochastycznych.

¹ Przypuśćmy, że chcemy obliczyć wartość I , gdy wiadomo, że z prawdopodobieństwem 1

$$I = \lim_{t \rightarrow \infty} \frac{\int_0^t X(s) ds}{t},$$

dla pewnego procesu stochastycznego $(X(s))_{s \geq 0}$. Wtedy naturalnym estymatorem jest

$$\hat{X}(t) = \frac{\int_0^t X(s) ds}{t}, \quad (0.1)$$

który jest (mocno) zgodnym estymatorem I . Jednakże zazwyczaj $\hat{X}(t)$ nie jest nieobciążonym estymatorem, tzn. $\mathbb{E} \hat{X}(t) \neq I$.

Modelem w pewien sposób idealnym jest sytuacja gdy proces stochastyczny jest stacjonarny. Wtedy estymator (0.1) jest nieobciążony. Zajmiemy się tym przypadkiem w następnym podrozdziale. Przypomnijmy, że przez proces stochastyczny $(X(t))_T$ z przestrzenią parametrów T nazywamy kolekcję zmiennych losowych. Za T możemy przyjąć np. $T = [0, \infty)$, $T = [a, b]$, itp. W tym rozdziale $T = [0, \infty)$.

¹17.1.2018

1 Procesy stacjonarne i analiza ich symulacji

Rozważmy proces stochastyczny $(X(t))_{t \geq 0}$ dla którego $\mathbb{E} X(t)^2 < \infty$, $(t \geq 0)$. Mówimy, że jest on *stacjonarny* jeśli spełnione są następujące warunki.

- (1) Wszystkie rozkłady $X(t)$ są takie same. W konsekwencji wszystkie $\mathbb{E} X(t)$ są równe jeśli tylko pierwszy moment $m = \mathbb{E} X(0)$ istnieje.
- (2) Dla każdego $h > 0$, łączne rozkłady $(X(t), X(t+h))$ nie zależą od t . W konsekwencji wszystkie kowariancje

$$R(h) = \text{Cov}(X(t), X(t+h)),$$

(jeśli tylko drugie momenty istnieją) nie zależą od t .

- (k) Dla każdego $0 < h_1 < \dots < h_{k-1}$, łączne rozkłady $(X(t), X(t+h_1), \dots, X(t+h_{k-1}))$ nie zależą od t .

Dla procesów stacjonarnych jest prawdziwe tzw. twierdzenie ergodyczne; dla procesu stacjonarnego $(X(t))_{t \geq 0}$ takiego, że $\mathbb{E} |X(t)| < \infty$, to z prawdopodobieństwem 1

$$\lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t X(s) ds$$

istnieje. Ponadto jeśli proces jest ergodyczny, granica jest stałą, a więc musi równać się $I = \mathbb{E} X(0)$.

Nie będziemy tutaj zajmowali się szczegółami teorii ergodycznej.² Łatwo jest podać przykład procesu stacjonarnego, który nie jest ergodyczny. Weźmy na przykład proces $X(t) = \xi$, gdzie ξ jest zmienną losową. Wtedy w trywialny sposób

$$\lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t X(s) ds = \xi.$$

Jesli ponadto są spełnione tzw. *warunki mieszania*, (to jest mocniejszy warunek niż ergodyczność) to wtedy dla pewnej stałej $\sigma^2 > 0$ (to nie jest $\text{Var} X(t)$) proces

$$\frac{\int_0^t X(s) ds - tI}{\sigma \sqrt{t}} \tag{1.2}$$

²referencje ?

dla $t \rightarrow \infty$, zbiega według rozkładu do $N(0, 1)$. Zastanówmy się, ile wynosi $\bar{\sigma}^2$. Zmienna

$$W = \frac{\int_0^t X(s) ds - tI}{\bar{\sigma}\sqrt{t}}$$

musi być asymptotycznie ustandaryzowana, to jest z średnią zero (trywialnie spełnione) oraz wariancją dążącą do 1. Mamy

$$\text{Var } W = \frac{\text{Var } \int_0^t X(s) ds}{\bar{\sigma}^2 t}. \quad (1.3)$$

Policzmy teraz mianownik.

Przepiszmy (1.2) w wygodnej dla nas formie

$$\frac{\int_0^t X(s) ds - tI}{\bar{\sigma}\sqrt{t}} = \frac{\sqrt{t}}{\bar{\sigma}}(\hat{X}(t) - I).$$

Chcemy aby $(\hat{X}(t) - I)$ było ustandaryzowane. Oczywiście $\mathbb{E}(\hat{X}(t) - I) = 0$, natomiast aby wariancja była równa 1 musimy unormować

$$\frac{\hat{X}(t) - I}{\sqrt{\text{Var } \hat{X}(t)}}.$$

Musimy zbadać zachowanie asymptotyczne mianownika. W tym celu udowodnimy następujący lemat.

Lemat 1.1 *Jeśli $\int_0^\infty |R(v)| dv < \infty$ i $\int_0^\infty R(s) ds \neq 0$, to dla $t \rightarrow \infty$*

$$\text{Var} \left(\int_0^t X(s) ds \right) \sim 2t \int_0^\infty R(v) dv.$$

Dowód. Mamy

$$\mathbb{E} \int_0^t X(s) ds = \int_0^t \mathbb{E} X(s) ds = mt.$$

Dalej bez straty ogólności możemy założyć $m = 0$. Wtedy

$$\begin{aligned}
 \text{Var} \left(\int_0^t X(s) ds \right) &= \mathbb{E} \left(\int_0^t X(s) ds \right)^2 \\
 &= \mathbb{E} \left[\int_0^t \int_0^t X(w)X(v) dw dv \right] \\
 &= \int_0^t \int_0^t \mathbb{E} (X(w)X(v)) dw dv \\
 &= \int_{(w,v) \in [0,t]^2: w \leq v} \mathbb{E} (X(w)X(v)) dw dv + \int_{(w,v) \in [0,t]^2: w > v} \mathbb{E} (X(w)X(v)) dw dv.
 \end{aligned}$$

Teraz dla $v \leq w$ mamy $\mathbb{E} X(w)X(v) = R(w-v)$. Mamy również dla $h > 0$ ponieważ

$$R(-h) = \mathbb{E} X(x)X(x+h) = \mathbb{E} X(x-h)X(x) = R(h).$$

i stąd

$$\begin{aligned}
 \int_{(w,v) \in [0,t]^2: w \leq v} \mathbb{E} X(w)X(v) dw dv &+ \int_{(w,v) \in [0,t]^2: w > v} \mathbb{E} X(w)X(v) dw dv \\
 &= 2 \int_0^t dv \int_0^v R(v-w) dv \\
 &\sim t 2 \int_0^\infty R(w) dw.
 \end{aligned}$$

□

Ponieważ

$$\text{Var } W = \frac{\text{Var} \int_0^t X(s) ds}{\bar{\sigma}^2 t} \rightarrow 1,$$

więc

$$\bar{\sigma}^2 = 2 \int_0^\infty R(s) ds.$$

Stąd mamy następujący fakt.

Fakt 1.2 *Jeśli $(X(t))_{t \geq 0}$ jest procesem stacjonarnym i ergodycznym, $\mathbb{E} X(t)^2 < \infty$ oraz $\int_0^\infty |R(v)| dv < \infty$ i $\int_0^\infty R(s) ds \neq 0$, to estymator $\hat{X}(t)$ jest nieobciążony (ma wartość oczekiwaną 1) oraz*

$$\lim_{t \rightarrow \infty} t \text{Var} (\hat{X}(t)) = \bar{\sigma}^2 = 2 \int_0^\infty R(s) ds. \quad (1.4)$$

Dowód. Mamy

$$\mathbb{E} \hat{X}(t) = \frac{1}{t} \int_0^t \mathbb{E} X(s) ds = \mathbb{E} X(0) = I.$$

Aby udowodnić (1.4) korzystamy z (1.3) oraz Lematu 1.1. $\square$

Zrobimy teraz kilka uwag.

- Przepiszmy (1.2) w wygodnej dla nas formie

$$\frac{\int_0^t X(s) ds - tI}{\bar{\sigma}\sqrt{t}} = \frac{\sqrt{t}}{\bar{\sigma}}(\hat{X}(t) - I).$$

Oczywiście przy odpowiednich założeniach ten proces zbiega według rozkładu do $N(0,1)$.

- Wielkość $\bar{\sigma}^2 = 2 \int_0^\infty R(s) ds$ zwie się TAVC (od *time average variance constant*) lub asymptotyczną wariancją.

Podsumujmy. Podobnie jak w rozdziale IV możemy napisać *fundamentalny związek*

$$b = z_{1-\epsilon/2} \frac{\bar{\sigma}}{\sqrt{t}}. \quad (1.5)$$

po między błędem b , poziomem ufności α oraz horyzontem symulacji t . Natomiast przedział ufności na poziomie α wynosi

$$\left(\hat{X}(t) - \frac{z_{1-\alpha/2}\bar{\sigma}}{\sqrt{t}}, \hat{X}(t) + \frac{z_{1-\alpha/2}\bar{\sigma}}{\sqrt{t}} \right).$$

co zapisujemy krótko

$$(\hat{X}(t) \pm \frac{z_{1-\alpha/2}\bar{\sigma}}{\sqrt{t}}).$$

Niestety w typowych sytuacjach nie znamy $\bar{\sigma}^2$ a więc musimy tą wielkość wyestymować z symulacji.³

³Czytaj dalej str. 540 Fishmana

2 Przypadek symulacji obciążonej

Będziemy teraz analizowali przypadek symulacji liczby I dla estymatora $\hat{X}$ spełniającego:

1. $I = \lim_{t \rightarrow \infty} \hat{X}(t)$ (z prawdopodobieństwem 1),
2. $\frac{\sqrt{t}}{\bar{\sigma}}(\hat{X}(t) - I)$ zbiega wg rozkładu do $\mathcal{N}(0, 1)$, gdzie

$$t \text{Var} \left(\int_0^t X(s) ds \right) \rightarrow \bar{\sigma}^2,$$

gdy $t \rightarrow \infty$ dla pewnej skończonej liczby $\bar{\sigma}^2 > 0$.

Ponieważ nie zakładamy stacjonarności, więc estymator $\hat{X}(t)$ jest zgodny ale nie jest nieobciążony. Powstaje więc problem analizy jak szybko $b(t) = \mathbb{E} \hat{X}(t) - I$ zbiega do zera. Zanalizujemy przypadek gdy $\beta = \int_0^\infty (\mathbb{E} X(s) - I) ds$ zbiega absolutnie, tzn. $\int_0^\infty |\mathbb{E} X(s) - I| ds < \infty$. Wielkość β można też przedstawić jako

$$\begin{aligned} \beta &= \lim_{t \rightarrow \infty} \int_0^t (\mathbb{E} X(s) - I) ds \\ &= \lim_{t \rightarrow \infty} (\mathbb{E} \int_0^t X(s) ds - tI) \\ &= \lim_{t \rightarrow \infty} t(\mathbb{E} \hat{X}(t) - I) \end{aligned}$$

nazywamy *asymptotycznym obciążeniem*. Wtedy

$$\begin{aligned} \mathbb{E} \hat{X}(t) - I &= \frac{\int_0^t (\mathbb{E} (X(s)) - I) ds}{t} \\ &= \frac{\int_0^\infty (\mathbb{E} (X(s)) - I) ds - \int_t^\infty (\mathbb{E} (X(s)) - I) ds}{t} \\ &= \frac{\beta}{t} + o(1/t). \end{aligned}$$

Ponieważ $\sqrt{\text{Var} \hat{X}(t)} \sim \bar{\sigma}/\sqrt{t}$ a więc zmienność naszego estymatora jest rzędu $O(t^{-1/2})$ czyli większa niż jego obciążenie.

Przykład 2.1 Aby uzmysłowić czytelnikowi trudności jakie można napotkać podczas symulacji rozpatrzmy przypadek symulacji średniej liczby zadań w systemie $M/G/\infty$ (po więcej informacji patrz dodatek XI.7.1). Będziemy rozpatrywać liczbę zadań $L(t)$ dla

1. $\lambda = 5$, i $S =_d \text{Exp}(1)$, $T_{\max} = 1000$,
2. $\lambda = 5$, i $S =_d 0.2X$, gdzie $X =_d \text{Par}(1.2)$, $T_{\max} = 1000$,
3. $\lambda = 5$, i $S =_d 0.2X$, gdzie $X =_d \text{Par}(1.2)$, $T_{\max} = 10000$,
4. $\lambda = 5$, i $S =_d 0.2X$, gdzie $X =_d \text{Par}(1.2)$, $T_{\max} = 100000$,
5. $\lambda = 5$, i $S =_d 1.2X$, gdzie $X =_d \text{Par}(2.2)$, $T_{\max} = 1000$,
6. $\lambda = 5$, i $S =_d 2.2X$, gdzie $X =_d \text{Par}(3.2)$, $T_{\max} = 1000$,

We wszystkich tych przykładach $\rho = \lambda \mathbb{E} S = 5$ a więc powinniśmy otrzymać zawsze średnią oczekiwaną liczbę zgłoszeń w systemie w warunkach stacjonarnych $\bar{l} = 5$ (patrz dodatek XI.7.1). Na rysunkach odpowiednio 2.1-2.6 pokazana jest wysymulowana realizacja

$$\hat{L}(t) = \frac{\int_0^t L(s) ds}{t}$$

w przedziale $1 \leq t \leq t_{\max}$.

Możemy zauważyć, że symulacja 1-sza szybko zbiega do granicznej wartości 5, chociaż horyzont $t_{\max}$ jest krótki. W tym przypadku $R(t) = 5 \exp(-t)$, a więc $\bar{\sigma}^2 = 2 \int_0^\infty R(s) ds = 10$, skąd $\bar{\sigma} = 3.1622$.

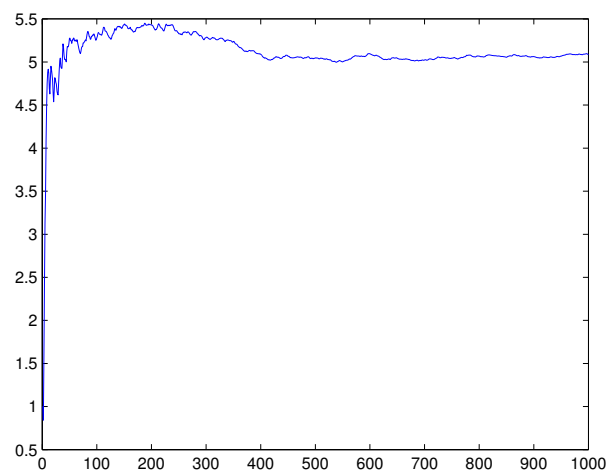
Natomiast z 2-gą jest coś dziwnego i dlatego warto ten przypadek bliżej przeanalizować. Policzmy najpierw $\bar{\sigma}^2$ ze wzoru (XI.7.8). Mamy

$$R(t) = 5 \int_t^\infty \left(1 + \frac{s}{0.2}\right)^{-1.2} ds = 5 \left(1 + \frac{t}{0.2}\right)^{-0.2}$$

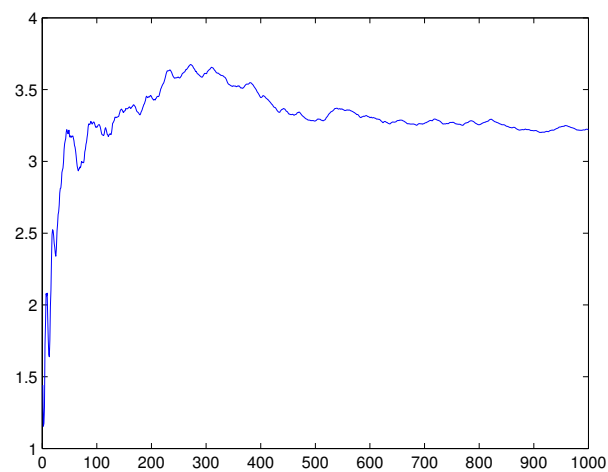
skąd mamy, że

$$\bar{\sigma}^2 = 2 \int_0^\infty R(t) dt = \infty .$$

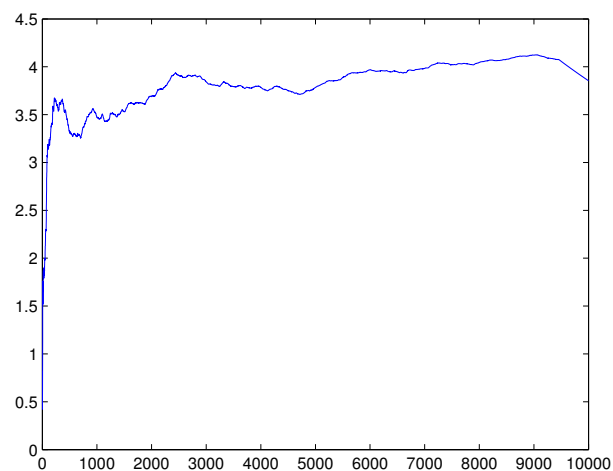
A więc nasze założenie o bezwzględnej całkowalności $R(t)$ nie jest spełnione. Nie można zatem stosować fundamentalnego związku, chociaż oczywiście ten estymator zbiega do 5. Zobaczymy to na dalszych przykładach na rys. 2.3, 2.4 (symulacja 3-cia i 4-ta).



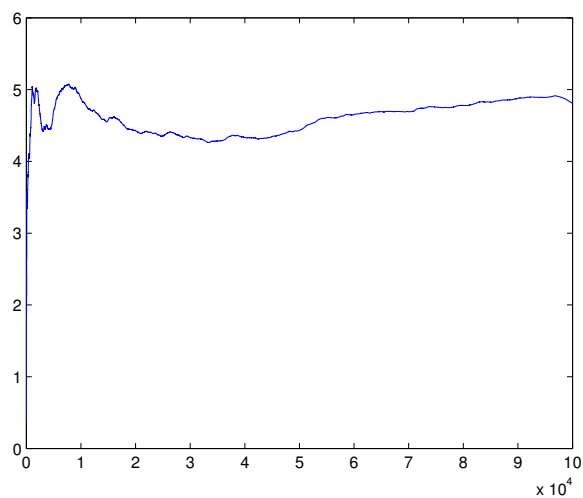
Rysunek 2.1: $\lambda = 5$, $\text{Exp}(1)$, $T_{\max} = 1000$; $\bar{l} = 5.0859$



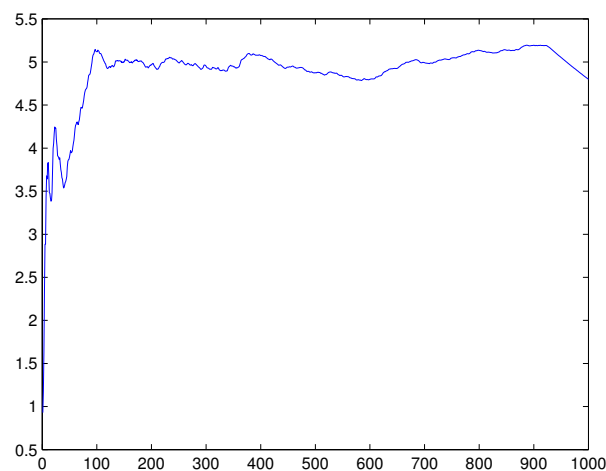
Rysunek 2.2: $\lambda = 5$, $\text{Par}(1.2) \times 0.2$, $T_{\max} = 1000$; $\bar{l} = 3.2208$



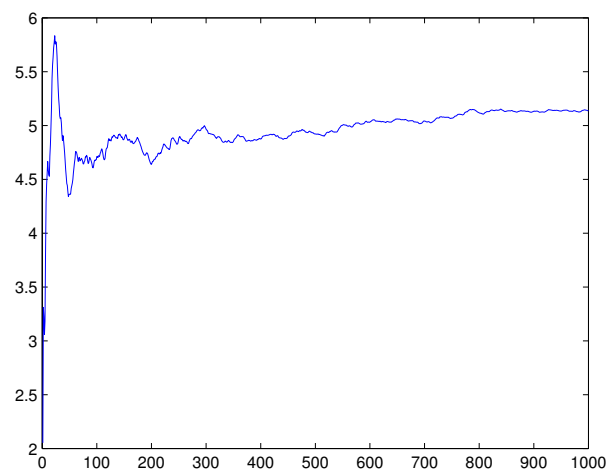
Rysunek 2.3: $\lambda = 5$, $\text{Par}(1.2) \times 0.2$, $T_{\max} = 10000$; $\bar{l} = 3.9090$



Rysunek 2.4: $\lambda = 5$, $\text{Par}(1.2) \times 0.2$, $T_{\max} = 100000$; $\bar{l} = 4.8395$



Rysunek 2.5: $\lambda = 5$, $\text{Par}(2.2) \times 1.2$, $T_{\max} = 1000$; $\bar{l} = 4.8775$



Rysunek 2.6: $\lambda = 5$, $\text{Par}(3.2) \times 2.2$, $T_{\max} = 1000$; $\bar{l} = 5.1314$

3. SYMULACJA GDY X JEST PROCESEM REGENERUJĄCYM SIĘ. 175

W przypadku sumulacji 5-tej mamy $\alpha = 2.2$ i wtedy $S = 1.2X$, gdzie $X =_{\text{d}} \text{Par}(2.2)$. Stąd

$$R(t) = 5 \int_t^\infty \left(1 + \frac{t}{1.2}\right)^{-2.2} ds = 5\left(1 + \frac{t}{1.2}\right)^{-1.2}.$$

Widzimy więc, że $\bar{\sigma}^2 = 2 \int_0^\infty R(t) dt = 60$, skąd mamy że aby $b = 0.5$ (jedna cyfra wiodąca) wystarczy $T_{\max} = 922$. Należy ostrzec, że nie wzięliśmy pod uwagę rozpędówki, przyjmując do obliczeń, że proces $L(t)$ jest od początku stacjonarny.

Jesli $\alpha = 3.2$ i wtedy $S = 2.2X$, gdzie $X =_{\text{d}} \text{Par}(3.2)$. Stąd

$$R(t) = 5 \int_t^\infty \left(1 + \frac{s}{2.2}\right)^{-3.2} ds = 5\left(1 + \frac{t}{2.2}\right)^{-2.2}.$$

Widzimy więc, że $\bar{\sigma}^2 = 2 \int_0^\infty R(t) dt = 18.3$, skąd mamy że aby $b = 0.5$ (jedna cyfra wiodąca) wystarczy $T_{\max} = 281$. Natomiast aby $b = 0.05$ to musimy mieć $T_{\max}$ większe od 28100.

3 Symulacja gdy X jest procesem regenerującym się.

Będziemy zakładać następującą strukturę procesu stochastycznego X . Przypuśćmy, że istnieją chwile losowe $0 \leq T_1 < T_2 < \dots$ takie, że pary zmiennych losowych $((Z_i, C_i))_{i \geq 1}$ są niezależne i począwszy od numeru $i \geq 2$ niezależne, gdzie

$$Z_0 = \int_0^{T_1} X(s) ds, Z_1 = \int_{T_1}^{T_2} X(s) ds, \dots$$

oraz $C_0 = T_1, C_1 = T_2 - T_1, \dots$. Takie procesy nazywamy procesami *regenerującymi się*. Wtedy mamy następujący rezultat (patrz Asmussen & Glynn [4]):

Fakt 3.1 *Z prawdopodobieństwem 1, dla $t \rightarrow \infty$*

$$\hat{X}(t) \rightarrow I = \frac{\mathbb{E} \int_{T_1}^{T_2} X(s) ds}{\mathbb{E} C_1}, \quad (3.6)$$

oraz

$$t^{1/2}(\hat{X}(t) - I) \rightarrow_{\text{d}} \mathcal{N}(0, \bar{\sigma}^2), \quad (3.7)$$

i $TAVC$ wynosi

$$\bar{\sigma}^2 = \frac{\mathbb{E} \zeta_1^2}{\mathbb{E} Z_1}$$

gdzie

$$\zeta_k = Z_k - IC_k .$$

Patrząc na postać (3.6) widzimy, że można też użyć innego estymatora, który jest krewniakiem estymatora $\hat{X}$. Mianowicie możemy wziąć

$$\hat{I}_k = \frac{Z_1 + \dots + Z_k}{C_1 + \dots + C_k} .$$

Jest oczywiste, że z prawdopodobieństwem 1

$$\hat{I}_k \rightarrow I$$

ponieważ

$$\frac{Z_1 + \dots + Z_k}{C_1 + \dots + C_k} = \frac{(Z_1 + \dots + Z_k)/k}{(C_1 + \dots + C_k)/k} \rightarrow \frac{\mathbb{E} Z_1}{\mathbb{E} C_1} .$$

Wtedy mamy następujące twierdzenie (patrz Asmussen & Glynn [4], Prop. 4.1).

Twierdzenie 3.2 Dla $k \rightarrow \infty$

$$k^{1/2}(\hat{I}_k - I) \rightarrow_d \mathcal{N}(0, \eta^2),$$

gdzie

$$\eta^2 = \frac{\mathbb{E} \zeta_1^2}{\mathbb{E} C_1}$$

i

$$\zeta_1 = Z_1 - IC_1 .$$

Stąd asymptotyczny $100(1 - \epsilon)\%$ przedział ufności jest dany przez

$$\hat{I}_k \pm z_{1-\epsilon/2} \hat{\eta}^2 / \sqrt{k}$$

gdzie

$$\hat{\eta}^2 = \frac{1}{k-1} \sum_{l=1}^k (C_l - \hat{I}_k Z_l)^2 / \left(\frac{1}{k} \sum_{l=1}^k C_l \right) .$$

Przykład 3.3 Proces $L(t)$ w M/M/1 jest regenerujący ponieważ

$$T_1 = \inf\{t > 0 : L(t-) > 0, L(t) = 0\}$$

oraz

$$T_{n+1} = \inf\{t > T_n : L(t-) > 0, L(t) = 0\}$$

definiuje momenty regeneracji procesu $L(t)$. Wynika to z własności braku pamięci rozkładu wykładniczego (patrz XI.5.6). Po osiągnięciu przez proces $L(t)$ zera, niezależnie jak długo czekaliśmy na nowe zgłoszenie, czas do nowego zgłoszenia ma rozkład wykładniczy $\text{Exp}(\lambda)$.

4 Przepisy na szacowanie rozpędówki

W wielu przypadkach proces po *pewnym czasie* jest już w przybliżeniu w warunkach stacjonarności. Następujący przykład ilustruje ten pomysł w sposób transparentny.

Przykład 4.1 Z teorii kolejek wiadomo, że jeśli w systemie M/M/1 wystartujemy w chwili $t = 0$ z rozkładu geometrycznego $\text{Geo}(\rho)$, to proces liczby zadań $L_1(t)$ jest stacjonarny. Dla uproszczenia wystartujemy proces niestacjonarny z $L_2(0) = 0$. Zakładamy przypadek $\rho < 1$. Wtedy proces $L_1(t)$ osiąga stan 0 w skończonym czasie. Ponieważ oba procesy mają tylko skoki ± 1 , więc mamy że procesy $L_1(t)$ i $L_2(t)$ muszą się spotkać, tzn.

$$T = \inf\{t > 0 : L_1(t) = L_2(t)\}$$

jest skończone z prawdopodobieństwem 1. Od tego momentu procesy ewoluują jednakowo. To pokazuje, że od momentu T proces $L_2(t)$ jest stacjonarny. Jednakże jest to moment losowy.

Bardzo ważnym z praktycznych względów jest ustalenie początkowego okresu s którego nie powinniśmy rejestrować. Ten okres nazywamy *długością rozpędówki* lub krótko *rozpędówką* (warm-up period). W tym celu definiujemy nowy estymator

$$\hat{X}(s, t) = \frac{\int_s^t X(v) dv}{t - s} . \quad (4.8)$$

Jedna realizacja – metoda średnich pęczkowych Dla tych czytelników, którzy znają teorię procesów stochastycznych, kluczową własnością zakładaną przez proces $X(t)$ jest spełnienie funkcjonalnej wersji centralnego twierdzenia granicznego

$$(\epsilon^{1/2}(X(t/\epsilon) - It/\epsilon))_{t \geq 0} \xrightarrow{\mathcal{D}} (\bar{\sigma}B(t))_{t \geq 0}, \quad (4.9)$$

gdzie $(B(t))_{t \geq 0}$ jest standardowym ruchem Browna.

Dzielimy odcinek symulacji $[0, t]$ na l pęczków jednakowej długości t/l (dla ułatwienia założmy, że l dzieli t) i zdefiniujemy *średnie pęczkowe*

$$\hat{X}_k(t) = \hat{X}((k-1)t/l, kt/l) = \frac{1}{t/l} \int_{(k-1)t/l}^{kt/l} X(s) ds,$$

gdzie $k = 1, 2, \dots$. Jeśli t ale również t/l są duże, to można przyjąć, że

(P1) $\hat{X}_k(t)$ ($k = 1, \dots, l$) ma w przybliżeniu rozkład normalny $\mathcal{N}(I, \bar{\sigma}^2 l/t)$;
 $(k = 1, \dots, l)$,

(P2) $\hat{X}_k(t)$ ($k = 1, \dots, l$) są niezależne.

Formalne uzasadnienie można przeprowadzić korzystając z założenia, że spełniona jest funkcjonalna wersja centralnego twierdzenia granicznego (4.9). Ponieważ

$$\hat{X}(t) = \frac{\hat{X}_1(t) + \dots + \hat{X}_l(t)}{l},$$

więc dla $t \rightarrow \infty$

$$l^{1/2} \frac{\hat{X}(t) - I}{\hat{s}_l(t)} \rightarrow_d T_{l-1}$$

gdzie T_{l-1} zmienną losową o rozkładzie t -Studenta z $l-1$ stopniami swobody i

$$\hat{s}_l^2(t) = \frac{1}{l-1} \sum_{i=1}^l (\hat{X}_i(t) - \hat{X}(t))^2.$$

A więc jeśli $t_{1-\epsilon/2}$ jest $1 - \epsilon/2$ kwantylem zmiennej losowej T_{l-1} , to asymptotyczny przedział ufności dla I jest

$$\hat{X}(t) \pm t_{1-\epsilon/2} \frac{\hat{s}_l(t)}{\sqrt{l}}.$$

Asmussen & Glynn [4] sugerują aby $5 \leq l \leq 30$.

4.1 Metoda niezależnych replikacji

Efekty symulacji podczas rozpędówki nie musimy rejestrować a dopiero po tym czasie zaczynamy rejestrować i następnie opracować wynik jak dla procesu stacjonarnego. Problemem jest, że nie wiemy co to znaczy po *pewnym czasie*. Fishman [14, 15] zaproponował następującą graficzną metodę szacowania rozpędówki. Zamiast symulowania jednej realizacji $X(t)$ ($0 \leq t \leq t_{\max}$) symulujemy teraz n realizacji $X^i(t)$, ($0 \leq t \leq t_{\max}$), gdzie $i = 1, \dots, n$. Każdą zaczynamy od początku z tego samego punktu, ale z rozłącznymi liczbami losowymi. Niech

$$\hat{X}^i(s, t) = \frac{\int_s^t X^i(v) dv}{t - s}$$

będzie opóźnioną średnią dla i -tej replikacji realizacji procesu ($i = 1, \dots, n$). Definiujemy

$$\hat{X}^\cdot(t) = \frac{1}{n} \sum_{i=1}^n X^i(t), \quad 0 \leq t \leq t_{\max},$$

oraz

$$\hat{X}^\cdot(t, t_{\max}) = \frac{1}{n} \sum_{i=1}^n \frac{\int_t^{t_{\max}} X^i(v) dv}{t_{\max} - t}, \quad 0 \leq t \leq t_{\max}.$$

Jeśli proces X jest stacjonarny to obie krzywe $\hat{X}^\cdot(t)$ i $\hat{X}^\cdot(t, t_{\max})$ powinny wyglądać podobnie. Pomijamy fluktuacje spowodowane niedużym n dla $\hat{X}^\cdot(t)$ oraz krótkością odcinka dla $\hat{X}^\cdot(t, t_{\max})$. Dlatego porównując na jednym wykresie

$$(\hat{X}^\cdot(s), \hat{X}^\cdot(s, t_{\max})), \quad 0 \leq s \leq t_{\max}$$

możemy oszacować jakie s należy przyjąć.

4.2 Efekt stanu początkowego i załadowania systemu

Na długość rozpędówki mają istotny wpływ przyjęty stan początkowy symulacji oraz załadowanie systemu.

Przykład 4.2 Tytułowy problem zilustrujemy za pracą Srikanta i Whitta [43] o symulacji prawdopodobieństwa straty

$$p_{\text{block}} = \frac{\rho^c / c!}{\sum_{j=0}^c \rho^j / j!}$$

(jest to tzw. wzór Erlanga) w systemie M/M/c ze stratami (patrz dodatek XI.7.3). Zadania wpływają z intensywnością λ natomiast ich średni rozmiar wynosi μ^{-1} . Jak zwykle oznaczamy $\rho = \lambda/\mu$. Dla tego systemu oczywiście znamy prawdopodobieństwo straty jak i wiele innych charakterystyk, co pozwoli nam zrozumieć problemy jakie mogą występować przy innych symulacjach. Zaczniemy od przeglądu kandydatów na estymatory. Niech $A(t)$ będzie liczbą wszystkich zadań (straconych i zaakceptowanych) które wpłynęły do chwili t a $B(t)$ jest liczba straconych zadań do chwili w odcinku $[0, t]$. Wtedy

$$B(t) = \sum_{j=1}^{A(t)} J_j$$

gdzie J_j jest zdefiniowane w (XI.7.13). A więc

$$B(t)/A(t) \rightarrow p_{\text{block}}$$

z prawdopodobieństwem 1. To uzasadnia propozycję estymatora p_{block}

$$\bar{B}_N(t) = \frac{B(t)}{A(t)}.$$

Estymator ten nazwiemy *estymatorem naturalnym*. Innym estymatorem jest tak zwany *prosty estymator*

$$\bar{B}_S(t) = \frac{B(t)}{\lambda t},$$

lub *niebezpośredni estymator* (indirect estimator)

$$\bar{B}_I(t) = 1 - \frac{\hat{L}(t)}{\rho}.$$

Podamy teraz krótkie uzasadnienia dla tych estymatorów. Czy są one zgodne? Pierwszy jest oczywiście zgodny. Podobnie wygląda sytuacja z estymatorem prostym B_S . Mianowicie

$$\begin{aligned} \lim_{t \rightarrow \infty} \frac{B(t)}{A(t)} &= \lim_{t \rightarrow \infty} \frac{B(t)}{t} \frac{t}{A(t)} \\ &= \lim_{t \rightarrow \infty} \frac{B(t)}{\lambda t}. \end{aligned}$$

ponieważ dla procesu Poissona, mamy $t/A(t) \rightarrow \lambda$ z prawdopodobieństwem 1. Natomiast dla estymatora niebezpośredniego aby pokazać zgodność trzeba wykorzystać tzw. wzór Little'a ⁴:

$$\hat{l} = \lambda(1 - p_{\text{block}})/\mu ,$$

gdzie $\hat{l}$ jest średnią. Jeśli natomiast by założyć, że system jest w warunkach stacjonarności, to można pokazać, że estymatory prosty i niebezpośredni są nieobciążone. W pracy Srikanta i Whitta [43] opisano następujący eksperyment z $\lambda = 380$, $\mu = 1$ i $c = 400$. Dla tego systemu wysymulowano, startując z pustego systemu, przy użyciu estymatora niebezpośredniego, z $t_{\max} = 5400$, mamy $p_{\text{block}} = 0.00017$. Tutaj TAVC wynosi około 0.00066. W tym zadaniu można policzyć numerycznie asymptotyczne obciążenie co przedstawione jest w tabl. 4.1 z $\alpha = \rho/c$. Zauważmy, że w przypadku $N(0) = 0$, TAVC jest

α	$N(0) = 0$	$N(0) = c$
0.7	1.00	-0.42
0.8	1.00	-0.24
0.9	0.99	-0.086
1.0	0.85	-0.0123
1.1	0.66	-0.0019
1.2	0.52	-0.0005

Tablica 4.1: Asymptotyczne obciążenie β dla $\hat{B}_I$

tego samego rzędu co obciążenia ($\beta/5400$ i 0.00066).

W następnym przykładzie zajmiemy się wpływem załadowania systemu.

Przykład 4.3 Pytanie o to jak długi należy dobrać horyzont symulacji $t_{\max}$ aby osiągnąć zadaną precyzję zależy często od współczynnika załadowania systemu który symulujemy. W systemie M/M/1 można policzyć TAVC $\bar{\sigma}^2$, która to wielkość poprzez fundamentalny związek rzutuje na $t_{\max}$. Mianowicie (patrz Whitt [47]) mamy

$$\bar{\sigma}^2 = \frac{2\rho(1 + \rho)}{(1 - \rho)^4} .$$

Oczywiście musimy założyć $\rho < 1$.

⁴Sprawdzić tożsamość.

Stąd widać, że $\bar{\sigma}^2$ gwałtownie rośnie, gdy $\rho \rightarrow 1$, czyli w *warunkach wielkiego załadowania*. Wzór powyższy jest szczególnym przypadkiem wzoru otrzymanego dla procesów narodzin i śmierci bez obliczania funkcji $R(t)$; patrz Whitt [47]. Stąd, korzystając z fundamentalnego związku, mamy że tak samo szybko rośnie horyzont symulacji aby utrzymać zadaną dokładność. Przeprowadzimy teraz analizę jak dobrać $t_{\max}$ do symulacji średniej liczby zadań w systemie. Takie symulacje przeprowadzaliśmy w podrozdziale VI.2.1. Przypomnijmy, że do eksperymentu przyjęto $\lambda = 8/17$ oraz $\mu = 9/17$. Wtedy średnia liczba zadań w systemie wynosi $\bar{l} = 8$ (patrz wzór XI.7.10). Ponieważ $\rho = 8/9$ więc

$$\bar{\sigma}^2 = 22032.$$

Jeśli ponadto przyjmiemy $b = 0.5$ i $\alpha = 0.05$ to wtedy $t_{\max} = 338550$. Natomiast gdy w jednym naszym z eksperymentów przyjęliśmy $t_{\max} = 10000000$ przy którym otrzymaliśmy wynik $\bar{l} = 7.9499$, to korzystając z fundamentalnego wzoru [??eq.stac.fund.zw??] otrzymamy, że połowa przedziału ufności wynosi 0.0920.

5 Uwagi bibliograficzne

Cały cykl prac na temat planowania symulacji w teorii kolejek napisał Whitt z współpracownikami; [45, 46, 47, 43]. Tym tematem zajmował się również Grassmann [49]

Zadania teoretyczne

- 5.1 Uzasadnić zgodność estymatorów z podrozdziału [??whit.srinikant??]. Pokazać, że estymatory B_I oraz B_S są nieobciążone jeśli proces stanu systemu $L(t)$ jest stacjonarny.

Zadania laboratoryjne

- 5.2 Przeprowadzić następujące eksperymenty dla M/M/1. Napisać program kreślący realizację $L(t)$ dla $0 \leq t \leq 10$. Eksperyment przeprowadzić dla
- (a) $\lambda = 1, \mu = 1.5$ oraz $L(0) = 0$.
 - (b) $\lambda = 1, \mu = 1.5$ i $L(0) = 2$.
 - (c) $\lambda = 1, \mu = 1.5$ i $L(0) = 4$.
 - (d) $L(0) \sim \text{Geo}(\rho)$, gdzie $\rho = \lambda/\mu = 2/3$.
- 5.3 Kontynuacja zad. 5.2. Obliczyć $\hat{L}(t_{\max})$, gdzie $\hat{L}(t) = (\int_0^t L(s) ds)/t$ gdy $L(0) = 0$.
- (a) Przeprowadzić eksperyment przy $\lambda = 1, \mu = 1.5$ oraz $L(0) = 0$ dla $t_{\max} = 5, 10, 50, 100, 1000$. Policzyc dla każdej symulacji połowy długości przedziały ufności na poziomie 95%.
 - (b) Przeprowadzić eksperyment przy $\lambda = 1, \mu = 1.1$ oraz $L(0) = 0$ dla $t_{\max} = 5, 10, 50, 100, 1000$.
- 5.4 Kontynuacja zad. 5.2. Przez replikacje rozumiemy teraz pojedynczą realizację $L(t)$ dla $0 \leq t \leq 10$. Dla replikacji $L_i(t)$ zachowujemy wektor $\mathbf{l}_i = (L_i(1), L_i(2), \dots, L_i(10))$ i niech 1000 będzie liczbą replikacji. Obliczyć

$$\hat{\mathbf{l}} = \left(\frac{1}{1000} \sum_{j=1}^{1000} L_j(0.5), \frac{1}{1000} \sum_{j=1}^{1000} L_j(1), \dots, \frac{1}{1000} \sum_{j=1}^{1000} L_j(10) \right).$$

Zrobić wykres średniej długości kolejki w przedziale $0 \leq t \leq 10$.

- (a) Przeprowadzić eksperyment dla $L(0) = 0$.
- (b) Przeprowadzić eksperyment dla $L(0) \sim \text{Geo}(\rho)$.

- 5.5 Napisać algorytm na generowanie dla systemu M/G/ ∞ estymatora

$$\frac{1}{t_{\max} - t_{\min}} \int_{t_{\min}}^{t_{\max}} L(t) dt$$

(patrz wzór (4.8)). Wsk. Zobacz algorytm w dodatku XI.7.1 z $t_{\min} = 0$.
Przeprowadzić eksperymenty omówione w przykładzie 2.1 z różnymi
 $t_{\min}$.

Projekt

- 5.6 Napisać program symulujący realizację procesu liczby zadań $L(t)$ w systemie M/G/ ∞ . Przeprowadzić eksperyment z
- (a) $\lambda = 5$ oraz $G = \text{Exp}(\mu)$ z $\mu = 1$.
 - (b) $\lambda = 5$ oraz rozmiary zadań mają rozkład Pareto z $\alpha = 1.2$ przemnożonym przez 0.2 (tak aby mieć średnią 1).
- Przyjąć $L(0) = 0$, $L(0) = 10$ oraz $L(0)$ ma rozkład Poissona $\text{Poi}(5)$. Dla wszystkich wypadków oszacować rozpędówkę. Na jednym wykresie umieścić $\int_0^t L(s) ds/t$ dla $1 \leq t \leq 100$ z symulacji (a) i (b). Przyjąć $L(0) = 0$.

Rozdział VIII

Łańcuchy Markowa i Metody Monte Carlo

1

1 Łańcuchy Markowa

Przez łańcuch Markowa rozumiemy ciąg zmiennych (elementów) losowych $X_0, X_1, X_2, \dots$ o wartościach w przestrzeni $\mathcal{E}$ i związany z sobą. W tym podrozdziale będziemy zakładać, że $\mathcal{E}$ jest skończona z elementami $\mathcal{E} = \{s_1, s_2, \dots, s_m\}$. Dla wygody będziemy dalej przyjmować $\mathcal{E} = \{1, \dots, m\}$. Przez *macierz przejścia* $\mathbf{P} = (p_{ij})_{i,j \in \mathcal{E}}$ rozumiemy macierz, której elementy spełniają:

$$p_{ij} \geq 0, \quad i, j \in \mathcal{E}$$

oraz

$$\sum_{j \in \mathcal{E}} p_{ij} = 1, \quad i \in \mathcal{E}.$$

Czasami, w oderwaniu od teorii łańcuchów Markowa mówimy też *macierz stochastyczna*. Niech $\boldsymbol{\mu} = (\mu_j)_{j \in \mathcal{E}}$ będzie rozkładem (funkcją prawdopodobieństwa) na $\mathcal{E}$.

Mówimy, że $X_0, X_1, \dots$ jest *łańcuchem Markowa z rozkładem początkowym* $\boldsymbol{\mu}$ i *macierzą przejścia* $\mathbf{P}$ jeśli dla $i_0, \dots, i_k \in \mathcal{E}$

$$\mathbb{P}(X_0 = i_0, X_1 = i_1, \dots, X_k = i_k) = \mu_{i_0} p_{i_0 i_1} \cdots p_{i_{k-1} i_k}.$$

¹17.1.2017

Ważnym pojęciem jest *rozkład stacjonarny*, tj. $\boldsymbol{\pi} = (\pi_j)_{j \in \mathcal{E}}$ (oczywiście jest to ciąg liczb nieujemnych sumujących się do 1) i spełniający

$$\pi_j = \sum_{i \in \mathcal{E}} \pi_i p_{ij}, \quad j \in \mathcal{E}.$$

Podamy teraz kilka potrzebnych dalej definicji. Elementy potęgi macierzy $\mathbf{P}^n$ oznaczamy $p_{ij}^{(n)}$. Dwa stany $i, j \in \mathcal{E}$ są *skomunikowane*, jeśli istnieją n_0, n_1 takie, że $p_{ij}^{(n_0)} > 0$ oraz $p_{ji}^{(n_1)} > 0$. Macierz $\mathbf{P}$ jest *nieredukowalna*, jeśli wszystkie stany są skomunikowane.

Mówimy, że macierz $\mathbf{P}$ jest *regularna* (*ergodyczna*), jeśli istnieje n_0 , takie że $\mathbf{P}^{n_0}$ ma wszystkie elementy ściśle dodatnie ($\mathbf{P}^{n_0} > \mathbf{0}$). Można pokazać, że $\mathbf{P}$ jest regularna wtedy i tylko wtedy gdy jest nieredukowalna i nieokresowa (odsyłamy po definicję okresowości do specjalistycznych książek z łańcuchów Markowa lub [?]).

Przykład 1.1 Rozważmy dwie macierze przejścia $\mathbf{P}_1$ i $\mathbf{P}_2$ dla łańcucha z stanami $\{1, 2, 3, 4\}$.

$$\mathbf{P}_1 = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 1/2 \end{pmatrix}$$

oraz

$$\mathbf{P}_2 = \begin{pmatrix} 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 1/2 \\ 1/2 & 1/2 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \end{pmatrix}.$$

Macierz $\mathbf{P}_1$ jest przykładem macierzy redukowalnej, natomiast $\mathbf{P}_2$ macierzy nieredukowalnej z okresem 2.

Fakt 1.2 *Jeśli $\mathbf{P}$ jest regularna, to istnieje tylko jeden rozkład stacjonarny $\boldsymbol{\pi}$, którego elementy są ściśle dodatnie, oraz*

$$\mathbf{P}^n \rightarrow \boldsymbol{\Pi},$$

gdzie $n \rightarrow \infty$, gdzie

$$\boldsymbol{\Pi} = \begin{pmatrix} \pi \\ \vdots \\ \pi \end{pmatrix}.$$

Jak symulować łańcuch Markowa X_n podaje następujący fakt.

Fakt 1.3 *Niech $\mathbf{P}$ i $\boldsymbol{\mu}$ będą dane. Istnieje funkcja $\varphi : \mathcal{E} \times [0, 1] \rightarrow \mathcal{E}$ taka, że jeśli Y_0 ma rozkład $\boldsymbol{\mu}$, to $Y_0, Y_1, \dots$ zdefiniowane rekurencyjnie*

$$X_{n+1} = \varphi(X_n, U_{n+1}), \quad n = 0, 1, \dots$$

gdzie $U_1, \dots$ jest ciągiem niezależnych zmiennych losowych o jednakowym rozkładzie jednostajnym na $[0, 1]$, jest łańcuchem Markowa z rozkładem początkowym $\boldsymbol{\mu}$ i macierzą przejścia $\mathbf{P}$.

Dowód Zdefiniujmy

$$\varphi(i, x) = \begin{cases} 1, & x \in [0, p_{i1}) \\ 2, & x \in [p_{i1}, p_{i1} + p_{i2}) \\ \vdots & \vdots \\ m, & x \in [p_{i1} + \dots + p_{i,m-1}, 1]. \end{cases}$$

□

Reprezentacja powyższa nie jest jednoznaczna.

Naszym tradycyjnym celem będzie obliczenie

$$I = \sum_{j \in \mathcal{E}} f(j) \pi_j,$$

gdy do dyspozycji mamy łańcuch Markowa (X_j) z rozkładem stacjonarnym $\boldsymbol{\pi}$. Analogicznie do rozważań rozdziału IV wprowadzamy estymator

$$\hat{Y}_n = \frac{1}{n} \sum_{j=0}^{n-1} f(X_j). \quad (1.1)$$

Jak się okaże poniżej, ten estymator jest mocno zgodny, ale przeważnie nie jest nieobciążony.

Fakt 1.4 *Jeśli macierz prawdopodobieństwa przejścia jest nieredukowalna, to*

$$\hat{Y}_n \rightarrow I = \sum_{j=1}^m \pi_j f(j), \quad p.n.p. \quad (1.2)$$

gdzie $n \rightarrow \infty$. Ponadto

$$\sqrt{n}(\hat{Y}_n - I) \rightarrow_d N(0, \bar{\sigma}^2),$$

gdzie

$$\bar{\sigma}^2 = \lim_{n \rightarrow \infty} n \text{Var } \hat{Y}_n. \quad (1.3)$$

Co więcej granice w (1.2) i (1.3) nie zależą od rozkładu początkowego μ .

Uwaga Zamiast formalnego dowodu pokażemy jego pomysł. Niech $i \in \mathcal{E}$ oraz podstawmy $Y_j = f(X_j)$. Ponieważ $\mathbf{P}$ jest nieredukowalna, więc łańcuch (X_n) nieskończenie często odwiedza stan i . To jest fakt z teorii łańcuchów Markowa. Ustalmy więc i takie, że $X_0 = i$. Niech $\gamma_j, j = 1, 2, \dots$ będą momentami odwiedzin (X_n) w i . Ponieważ (X_n) jest łańcuchem Markowa, więc cały ciąg $f(X_n)$ rozpada się na niezależne cykle $\{f(X_0), \dots, f(X_{\gamma_1-1})\}, \{f(X_{\gamma_1}), \dots, f(X_{\gamma_2-1})\}, \dots$. Zauważmy teraz, że

$$W_l = \sum_{j=\gamma_l}^{\gamma_{l+1}-1} f(X_j), \quad l = 1, 2,$$

są niezależnymi zmiennymi losowymi o jednakowym rozkładzie. Ponadto z teorii łańcuchów Markowa wiadomo, że $\mathbb{E}(\gamma_{l+1} - \gamma_l) < \infty$ oraz $\mathbb{E} W_l^2 < \infty$. Kluczem do dalszych rozważań jest przedstawienie

$$Y_1 + \dots + Y_n = W_1 + \dots + W_{\nu(n)} + R_n$$

gdzie $\nu(n) = \max\{j : \gamma_j \leq n\}$, oraz R_n nie zależy od W_j -ów. Teraz należy wykorzystać wariant mocnego prawa wielkich liczb oraz centralnego twierdzenia granicznego z losowym indeksem.

W rozdziale IV zdefiniowaliśmy pojęcie stacjonarności procesu stochastycznego $X(t)$. Jest to pojęcie ważne gdy parametr procesu jest $\mathbb{R}$ lub $\mathbb{R}_+$ ale też gdy $\mathbb{Z}$ lub $\mathbb{Z}_+$. Przypomnijmy więc to pojęcie dla ciągu zmiennych losowych $Z_n, n \in \mathbb{Z}_+$. Mówimy, że jest on *stacjonarny* jeśli spełnione są następujące warunki.

- (k) Dla każdego $k = 1, 2, \dots$ oraz $0 < h_1 < \dots < h_{k-1}$, łączne rozkłady $(Y_j, Y_{j+h_1}, \dots, Y_{j+h_{k-1}})$ nie zależą od j .

Konsekwencją definicji jest

- (1) Wszystkie rozkłady Y_j są takie same ($j \in \mathbb{Z}_+$). W konsekwencji wszystkie $\mathbb{E} Y_j$ są równe jeśli tylko pierwszy moment $m = \mathbb{E} Y_0$ istnieje.
- (2) Dla każdego $h = 1, \dots$, łączne rozkłady (Y_j, Y_{j+h}) nie zależą od j . W konsekwencji funkcja kowariancji $\rho_h = \text{Cov}(Z_j, Z_{j+h})$ zależy od h , jeśli tylko drugi moment istnieje.

Niech $\boldsymbol{\pi}$ będzie rozkładem stacjonarnym i $(X_n)_{n=0,1,\dots}$ będzie łańcuchem Markowa z rozkładem początkowym $\boldsymbol{\mu} = \boldsymbol{\pi}$ i macierzą przejścia $\boldsymbol{P}$. Wtedy $(Y_n)_{n=0,1,\dots}$, gdzie $Y_n = f(X_n)$ jest ciągiem stacjonarnym i funkcją kowariancji tego ciągu jest

$$\rho_k = \text{Cov}(Y_n, Y_{n+k}).$$

Jeśli przestrzeń stanów jest skończona i $\boldsymbol{P}$ jest regularna, to mamy $\sum_j |\rho_j| < \infty$. Wymaga to dowodu, na przykład z użyciem twierdzenia Perrona-Frobeniusa.

Zauważmy, że podobnie jak rozdziale IV stałą $\bar{\sigma}^2$ możemy nazywać TAVC. Pokażemy teraz inny wzór na TAVC.

Lemat 1.5 *Jeśli $\boldsymbol{P}$ jest regularna to*

$$\bar{\sigma}^2 = \rho_0 + 2 \sum_{k=1}^{\infty} \rho_k.$$

Dowód Ponieważ $\bar{\sigma}^2$ zdefiniowane w (1.3) nie zależy od warunku początkowego $\boldsymbol{\mu}$ więc rozważmy stacjonarny ciąg (Z_n) . Bez straty ogólności możemy założyć $\mathbb{E} Z_j = 0$. Wtedy

$$\begin{aligned} n \text{Var } \hat{Y}_n &= \frac{1}{n} \text{Var}(Y_0 + \dots + Y_{n-1}) \\ &= \frac{1}{n} (n \text{Var } Y_0 + 2 \sum_{0 \leq i < j \leq n} \text{Var}(Y_i, Y_j)) \\ &= \rho_0 + \frac{2}{n} \sum_{0 \leq i < j \leq n} \rho_{j-i} \\ &= \rho_0 + \frac{2}{n} \sum_{i=0}^{n-1} \sum_{h=1}^{n-i} \rho_h \\ &\sim \rho_0 + 2 \sum_{h=1}^{\infty} \rho_h. \end{aligned}$$

□

Jak estymować ρ_k i $\bar{\sigma}^2$? Możemy to zrobić przy użyciu estymatorów

$$\hat{\rho}_k = \frac{1}{n-k} \sum_{j=1}^{n-k} (Y_j - \hat{Y}_n)(Y_{j+k} - \hat{Y}_n)$$

oraz

$$\hat{\sigma}^2 \approx \sum_{k=-N}^N \hat{\rho}_k$$

Estymator $\hat{Y}_n$ jest oczywiście zgodny (dąży z prawdopodobieństwem 1 do $I = \sum_{j=1}^m \pi_j f(j)$), ale jeśli rozkład początkowy $\boldsymbol{\mu}$ zmiennej X_0 nie jest rozkładem stacjonarnym, to nie jest nieobciążonym. Jak szybko $\mathbb{E} \hat{Y}_n$ zbiega do I mówi następujący fakt.

Fakt 1.6 *Dla pewnej skończonej stałej A*

$$\mathbb{E} \hat{Y}_n = I + \frac{A}{n} + o\left(\frac{1}{n}\right).$$

Dowód Podstawmy $a_j = \mathbb{E} Y_j - \bar{y}$. Ponieważ $\mathcal{E}$ jest skończona, więc $\sum_j |a_j| < \infty$ (tego nie będziemy dowodzić). Teraz

$$\mathbb{E} \hat{Y}_n = \frac{1}{n} \sum_{j=0}^n (\mathbb{E} Y_j - I + I) = I + \frac{1}{n} \sum_{j=0}^{n-1} a_j = I + \frac{1}{n} \sum_{j=0}^{\infty} a_j - \frac{1}{n} \sum_{j=n}^{\infty} a_j$$

a ponieważ $\frac{1}{n} \sum_{j=n}^{\infty} a_j = o(1/n)$ więc dowód jest skończony. □

Uwaga Fakt 1.6 jest prawdziwy dla łańcuchów ze skończoną liczbą stanów.

Fakt 1.7 *Niech (Y_n) będzie łańcuchem Markowa z macierzą prawdopodobieństwa $\mathbf{P}$. Załóżmy, że istnieją ściśle dodatnie liczby $v_j, j \in \mathcal{E}$ takie, że*

$$v_i p_{ij} = v_j p_{ji}, \quad i, j \in \mathcal{E} \quad (1.4)$$

Wtedy

$$\pi_j = \frac{v_j}{\sum_{i \in \mathcal{E}} v_i}, \quad j \in \mathcal{E}$$

jest rozkładem stacjonarnym.

Dowód Bez straty ogólności możemy założyć, że v_j w (1.4) sumują się do jedności. Teraz

$$\sum_{i \in \mathcal{E}} v_i p_{ij} = \sum_{i \in \mathcal{E}} v_j p_{ji} = v_j \sum_{i \in \mathcal{E}} p_{ji} = v_j.$$

□

Warunek (1.4) będziemy nazywać DBC (od *detailed balance condition*). Dla stacjonarnych łańcuchów Markowa jest on równoważny odwracalności. Mówimy, że stacjonarny łańcuch Markowa $Y_0, Y_1, \dots$ jest *odwracalny*, jeśli dla $n = 0, 1, 2, \dots$

$$(Y_0, Y_1, \dots, Y_n) =_d (Y_n, Y_{n-1}, \dots, Y_0).$$

Zostawiamy czytelnikowi pokazanie następującego wniosku.

Wniosek 1.8 *Jeśli macierz przejścia $\mathbf{P}$ jest symetryczna, tj. $p_{ij} = p_{ji}$, to rozkład stacjonarny jest jednostajny na $\mathcal{E}$ czyli $\pi = 1/|\mathcal{E}|$, $j \in \mathcal{E}$.*

W dalszej części rozdziału będziemy mieli do czynienia z pewną klasą łańcuchów Markowa z przestrzenią stanów mającą strukturę grafu. Dlatego też zrobimy teraz bardzo krótkie wprowadzenie tego pojęcia.

Podstawowe pojęcia dotyczące grafów Graf G składa się z wierzchołków V oraz krawędzi E . Zbiór krawędzi można utożsamić z jakimś podzbiorem zbioru par (nieuporządkowanych) (i, j) , $i \neq j$, $i, j \in V$. Mówimy, że (i, j) są sąsiadami, jeśli jest krawędź łącząca i z j . Liczbę sąsiadów wierzchołka i oznaczamy przez $\deg(i)$. Mówimy, że graf jest *spójny*, jeśli jest możliwość przejścia po krawędziach z każdego wierzchołka do wszystkich innych, niekoniecznie w jednym kroku. Innym pojęciem jest okresowość która jest związana z pojęciem *bipartite*.² *Drogą* pomiędzy dwoma wierzchołkami grafu nazywamy krawędź pomiędzy nimi. Natomiast *drogą* jest trasa, polegająca na podróżowaniu od wierzchołka do wierzchołka po łączących je krawędziach.

Przykład 1.9 Proste błędzenie przypadkowe na grafie G . W prostym (symetrycznym) błędzeniu przypadkowym z stanu i przechodzimy do dowolnego sąsiada z tym samym prawdopodobieństwem. A więc

$$q_{ij} = \begin{cases} \frac{1}{\deg(i)}, & \text{jeśli } j \text{ jest sąsiadem } i, \\ 0, & \text{w przeciwnym razie.} \end{cases} \quad (1.5)$$

²A graph is bipartite if and only if it does not have cycles with an odd number of edges.

Łatwo sprawdzić, że jeśli $v_i = \deg(i)$ to równość (1.4) jest spełniona. A więc

$$\pi_i = \frac{\deg(i)}{\sum_{j \in V} \deg(j)}, \quad j \in V$$

jest rozkładem stacjonarnym. Polecamy się zastanowić kiedy macierz $\mathbf{Q}$ jest regularna a kiedy nieredukowalna. Podstawą jest pojęcie grafu spójnego,

Przykład 1.10 Błądzenie przypadkowe na hiperkostce. Rozważmy zbiór wierzchołków $\{0, 1\}^m$. Elementami są ciągi m -elementowe zer i jedynek. Sąsiadujące elementy są oddalone o 1 tj. jeśli $s_1 = (e_{11}, \dots, e_{1m})$ oraz $s_2 = (e_{21}, \dots, e_{2m})$, są one sąsiadami jeśli $\sum_{j=1}^m |e_{1j} - e_{2j}| = 1$. A więc z każdego wierzchołka wychodzi m krawędzi, czyli $\deg(i) = m$. Rozkład stacjonarny jest więc rozkładem jednostajnym na V .

Powyższe przykłady mają jedną cechę wspólną. W jednym kroku łańcuch się przemieszcza do najbliższego sąsiada. Zamiast więc pełnej struktury grafowej (V, E) wystarczy podać dla każdego stanu $i \in V$ listę jego sąsiadów $\mathcal{N}(i)$. Polecamy czytelnikowi zastanowić się nad równoważnością opisów przez zdefiniowanie wszystkich wierzchołków a zdefiniowanie zbioru sąsiadów dla wszystkich stanów.

Formalnie rodzina $\{\mathcal{N}(i) : i \in V\}$ podzbiorów V jest *systemem sąsiedztwa* jeśli $i \notin \mathcal{N}(i)$. Jeśli natomiast dla dowolnych $i \neq j \in V$ istnieje droga $i_1, \dots, i_l$, tzn. $i_1 \in \mathcal{N}(i), i_2 \in \mathcal{N}(i_1), \dots, j \in \mathcal{N}(i_l)$, to mówimy, że system sąsiedztwa jest *skomunikowany*.

2 MCMC

Omówimy teraz pewną metodę Monte Carlo łańcuchów Markowa (ang. Markov chain Monte Carlo; stąd MCMC). Naszym zadaniem jest generowanie obiektu losowego o wartościach w $\mathcal{E} = \{s_1, \dots, s_m\}$ o rozkładzie π , tj. o masie π_j na elemencie s_j (na razie to nie jest rozkład stacjonarny ponieważ jeszcze nie mamy macierzy przejścia). Bardziej dokładnie, chcemy obliczyć $\sum_{j \in \mathcal{E}} f(j)\pi_j$. Do tego celu skonstruujemy łańcuch Markowa (Y_n) z macierzą przejścia $\mathbf{P}$, dla którego π jest rozkładem stacjonarnym. Wtedy $\hat{Y}_n$ zdefiniowany w (1.1) będzie szukanym estymatorem $\sum_{j \in \mathcal{E}} f(j)\pi_j$. Przy pewnych warunkach (regularność $\mathbf{P}$) Y_n w przybliżeniu ma rozkład π . Rozkład π nazywa się rozkładem celowym (ang. target distribution).

2.1 Algorytmy Metropolisa

Niech π będzie naszym celowym rozkładem na $\mathcal{E}$. Wybieramy macierz przejścia $\mathbf{Q} = (q_{ij})_{i,j \in \mathcal{E}}$, która będzie grała rolę *macierzy generującej kandydatów*. Rozkład π nie jest rozkładem stacjonarnym dla $\mathbf{Q}$. Zdefiniujmy teraz macierz prawdopodobieństw przejścia $\mathbf{P}$ generującą łańcuch Markowa (Y_n) :

$$p_{ij} = \begin{cases} q_{ij} \min(1, \pi_j/\pi_i), & \text{jeśli } i \neq j \\ 1 - \sum_{k \neq i} q_{ik} \min(1, \pi_k/\pi_i), & \text{jeśli } i = j. \end{cases} \quad (2.6)$$

Zostawiamy czytelnikowi sprawdzenie, że $\mathbf{P} = (p_{ij})$ jest macierzą przejścia. Odpowiadający jej łańcuch Markowa jest (Y_n) . Następujące twierdzenie jest podstawą algorytmu Metropolisa.

Fakt 2.1 [*Metropolis*] Załóżmy, że $\mathbf{Q}$ jest regularną symetryczną macierzą prawdopodobieństw przejścia oraz π jest rozkładem na $\mathcal{E}$ z ściśle dodatnimi elementami. Wtedy $\mathbf{P}$ jest macierzą regularną z stacjonarnym rozkładem π .

Dowód Z wzoru (2.6) widzimy, że $q_{ij} > 0$ wtedy i tylko wtedy gdy $p_{ij} > 0$. Ponieważ $\mathbf{Q}$ jest regularna, więc $\mathbf{P}$ też musi być. Sprawdźmy teraz (1.4) z $v_j = \pi_j$. Mamy

$$\pi_i p_{ij} = \pi_i q_{ij} \min(1, \pi_j/\pi_i) = q_{ij} \min(\pi_i, \pi_j) = q_{ji} \min(\pi_j, \pi_i) = \pi_j p_{ji}$$

□

Na podstawie tego twierdzenia podamy teraz algorytm Metropolisa na generowanie łańcucha Markowa $Y_0, Y_1, \dots$ z rozkładem stacjonarnym π , na podstawie macierzy generującej kandydatów $\mathbf{Q}$.

1. $Y_k = i$
2. generuj X zgodnie z rozkładem $\mathbf{q}_i = (q_{i1}, \dots, q_{im})$.
3. podstaw $\alpha = \min(1, \pi_X/\pi_i)$
4. losuj rand; jeśli $\text{rand} \leq \alpha$ to podstaw $Y_{k+1} = X$,
5. w przeciwnym razie idź do 1.

Chcemy się pozbyć założenia symetryczności. Zastanówmy się najpierw nad ogólnym pomysłem. Jest to w pewnym sensie metoda podobna do metody eliminacji z rozdziału III. Możemy przepisać (2.6) w postaci

$$p_{ij} = \begin{cases} q_{ij}\alpha_{ij}, & \text{jeśli } i \neq j, \\ 1 - \sum_{k \neq i} q_{ik}\alpha_{ik}, & \text{jeśli } i = j. \end{cases}$$

gdzie $\alpha_{ij} = \min(1, \pi_j/\pi_i)$. Teraz możemy się spytać czy istnieją inne podstawienia dla α_{ij} takie, że

- (I) $\mathbf{P}$ jest macierzą przejścia regularną,
- (II) $\boldsymbol{\pi}$ jest rozkładem stacjonarnym dla $\mathbf{P}$,
- (III) macierz $\mathbf{P}$ jest macierzą łańcucha odwracalnego, tj. spełniającego warunek DBC (1.4).

Zauważmy, że jeśli $q_{ij} = \pi_j$, $\alpha_i = 1$ ($i, j \in \mathcal{E}$), to mamy niezależne replikacje (Y_j) o rozkładzie $\boldsymbol{\pi}$. Uogólnieniem metody Metropolis'a niewymagającej symetryczności $\mathbf{Q}$ jest podane w następującym fakcie

Fakt 2.2 [*Metropolis-Hastings*] Niech $\mathbf{Q}$ będzie regularną macierzą przejścia taką, że $q_{ij} > 0$ wtedy i tylko wtedy gdy $q_{ji} > 0$. Jeśli

$$\alpha_{ij} = \min\left(1, \frac{\pi_j q_{ji}}{\pi_i q_{ij}}\right), \quad i \neq j$$

to $\mathbf{P}$ spełnia postulaty (I)-(III) powyżej.

Dowód To, że $\mathbf{P}$ jest macierzą stochastyczną jest oczywiste. Teraz sprawdzimy, że dla $v_j = \pi_j$ jest spełniony warunek DBC (1.4). Mamy

$$\pi_i q_{ij} \min\left(1, \frac{\pi_j q_{ji}}{\pi_i q_{ij}}\right) = \min(\pi_i q_{ij}, \pi_j q_{ji}) = \pi_j q_{ji} \min\left(1, \frac{\pi_i q_{ij}}{\pi_j q_{ji}}\right).$$

□

Uwaga Ogólny schemat jest następujący. Dla macierzy przejścia generującą kandydatów $\mathbf{Q}$, szukamy tzw. *prawdopodobieństw akceptacji* α_{ij} . Problemem

jest aby $\mathbf{P}$ zdefiniowana przez $p_{ij} = \alpha_{ij}q_{ij}$ była macierzą przejścia. Dotychczas mieliśmy

$$\alpha_{ij} = \min(1, \frac{\pi_j}{\pi_i}) \quad (2.7)$$

lub

$$\alpha_{ij} = \min(1, \frac{\pi_j q_{ji}}{\pi_i q_{ij}}). \quad (2.8)$$

Niech

$$\alpha_{ij} = \frac{s_{ij}}{1 + t_{ij}}$$

dla pewnej macierzy symetrycznej (s_{ij}) i

$$t_{ij} = \frac{\pi_i q_{ij}}{\pi_j q_{ji}}$$

Musi być spełniony warunek

$$s_{ij} \leq 1 + \min(t_{ij}, t_{ji}).$$

W przypadku równości mamy (pokazać; zad. ³) (2.8). Jeśli q_{ij} są stałe, to mamy (2.7). Innym przypadkiem jest tzw. *algorytm Barkera*:

$$\alpha_{ij} = \frac{\pi_j q_{ij}}{\pi_j q_{ji} + \pi_i q_{ij}}.$$

W szczególnym przypadku gdy $q_{ij} = 1/|\mathcal{E}|$, to

$$\alpha_{ij} = \frac{\pi}{\pi_i + \pi_j}.$$

2.2 Przykład: Algorytm Metropolisa na grafie G .

Rozważmy graf G z przykładu 1.9. Niech $N = \max_{i \in V} \deg(i)$ oraz $\mathcal{N}(i)$ jest zbiorem wszystkich sąsiadów wierzchołka $i \in V$. Zdefiniujmy na przestrzeni stanów V łańcuch Markowa błądzący po “sąsiednich” wierzchołkach z macierzą przejścia $\mathbf{P}$:

$$q_{ij} = \begin{cases} \frac{1}{M}, & j \in \mathcal{N}(i) \\ 0, & i \neq j \text{ \& } j \notin \mathcal{N}(i) \\ 1 - \frac{\deg(i)}{M}, & i = j \end{cases}, \quad (2.9)$$

³Zadanie

gdzie $M \geq N$. Zakładamy, że łańcuch Markowa z taką macierzą przejścia jest nieprzywiedlny i nieokresowy. To założenie wynika ze struktury sąsiedztwa. Można też nietrudno pokazać, że jest odwracalny. Stąd też już łatwo widać, że rozkład stacjonarny tego łańcucha jest jednostajny $\mathcal{U}(V)$.

Chcemy teraz na przestrzeni stanów V skonstruować łańcuch Markowa (Y_n) z rozkładem stacjonarnym $\pi_j = b_j/B$, gdzie $b_j > 0$ oraz $B = \sum_{j \in V} b_j$. W zastosowaniach często przestrzeń stanów V jest bardzo duża i nie ma możliwości obliczyć B , natomiast bez trudności policzymy b_j/b_k . Jeśli jednak chcemy tylko generować zmienne o wartościach w V z rozkładem π , to możemy skorzystać z następującego lematu, na podstawie którego zbudujemy algorytm.

Lemat 2.3 *Rozważmy graf G z przykładu 1.9. Niech $N = \max_{i \in V} \deg(i)$ oraz $\mathcal{N}(i)$ jest zbiorem wszystkich sąsiadów wierzchołka $i \in V$. Zdefiniujemy na przestrzeni stanów V łańcuch Markowa błędzący po sąsiednich wierzchołkach z macierzą przejścia:*

$$p_{ij} = \begin{cases} \frac{1}{M} \min(1, \pi_j/\pi_i), & j \in \mathcal{N}(i) \\ 0, & i \neq j \text{ i } j \notin \mathcal{N}(i) \\ 1 - \sum_{k \neq i} p_{ik}, & i = j \end{cases},$$

gdzie $M \geq N$. Jeśli $\mathbf{Q}$ zdefiniowana przez (2.9) jest regularna, to $\mathbf{P}$ też z rozkładem stacjonarnym π .

Dowód Pokażemy, że DBC jest spełnione i stąd z Faktu 1.7 wynika teza. Dla $i \neq j$, jeśli $\pi_i \leq \pi_j$, to $p_{ij} = 1$ oraz $p_{ji} = \pi_i/\pi_j$. Stąd mamy DBC. Podobnie analizujemy przypadek $\pi_i \geq \pi_j$. $\square$

Zauważmy teraz, że $M = |V| \geq \max_{i \in V} \deg(i)$. Wykorzystamy to w następującym algorytmie.

Algorytm ITM

1. Łańcuch jest w stanie i
2. Losuj j z sąsiedztwa i z V z prawdopodobieństwem $1/M$
3. Akceptuj z prawdopodobieństwem $\min(1, \pi_j/\pi_i)$, gdzie j jest proponowanym nowym stanem
4. jeśli zaakceptowane, to nowy stan jest j , przeciwnym razie idź do 1.

2.3 Zastosowanie algorytmu Metropolisa do losowania dozwolonej konfiguracji w modelu z twardą skorupą

Rozważmy teraz spójny graf (V, E) . Chcemy pomalować niektóre wierzchołki grafu na czarno, pozostawiając resztę białymi. W wyniku otrzymamy tzw. konfigurację czyli element z $\{c, b\}^V$. Interesuje nas model z *twardą skorupą* (ang. hard core), który nie dopuszcza dwóch sąsiednich wierzchołków pomalowanych na czarno. Taką konfigurację nazywamy *dozwoloną*. Niech z_G będzie liczbą dozwolonych konfiguracji.

Na zbiorze wszystkich konfiguracji definiujemy rozkład

$$\pi_G(\xi) = \begin{cases} \frac{1}{z_G} & \text{jesli } \xi \text{ jest konfiguracją dozwoloną,} \\ 0 & \text{w przeciwnym razie.} \end{cases} \quad (2.10)$$

Element losowy o takim rozkładzie przyjmuje wartości w zbiorze Λ konfiguracji dozwolonych. Pytaniem jest jak losować taki element losowy. Problemem jest, że w przypadku skomplikowanych grafów z_G jest duże i trudne do obliczenia. W tym celu posłużymy się więc metodą MCMC. Skonstruujemy nieredukowalny i nieokresowy łańcuch Markowa (Y_n) , którego rozkład stacjonarny jest π_G . Łańcuch ten startuje z dowolnej konfiguracji dozwolonej. Ponieważ rozkład Y_n zbiega do π , więc będziemy przyjmować, że Y_n ma rozkład π . Jak zwykle pytaniem jest jakie duże należy wziąć n żeby można było rozsądnie przybliżać rozkład π przez rozkład Y_n .

Łańcuch Y_n ewoluuje na zbiorze konfiguracji dozwolonych Λ . Będziemy oznaczać przez $Y_n(v)$ kolor wierzchołka v . Niech $p_{\xi, \xi'}$ oznacza prawdopodobieństwo przejścia od konfiguracji dozwolonej ξ do ξ' . Algorytm zaczynamy od wyboru dozwolonej konfiguracji Y_0 . Przejście z Y_n do Y_{n+1} otrzymamy następująco.

1. Wylosuj “losowo” wierzchołek $v \in V$ i rzuć monetą.
2. Jeśli pojawi się “orzeł” oraz wylosowany został wierzchołek v , którego wszyscy sąsiedzi są “biali”, to pomaluj go na czarno; w przeciwnym razie w $n + 1$ -kroku wierzchołek v jest biały.
3. Kolory wszystkich pozostałych wierzchołków są niezmienione.

Niech $\xi \in \Lambda$ będzie konfiguracją. Dla wierzchołka v oznaczamy przez $\xi(v)$ kolor wierzchołka v oraz przez $m(\xi)$ oznaczamy liczbę czarnych wierzchołków. Strukturę sąsiedztwa na zbiorze konfiguracji dozwolonych Λ wprowadzamy

następująco. Mówimy, że $\xi' \in \mathcal{N}(\xi)$ jeśli $m(\xi) + 1 = m(\xi')$; dla pewnego v mamy $\xi(v) = 0, \xi'(v) = 1$ oraz $\xi(w) = \xi'(w) = 0$ dla $w \sim v$ lub jeśli $m(\xi') + 1 = m(\xi)$; dla pewnego v mamy $\xi(v) = 1, \xi'(v) = 0$. Funkcja prawdopodobieństwa przejścia $p_{\xi\xi'}$ jest następująca:

$$p_{\xi\xi'} = \begin{cases} \frac{1}{2|V|} & \text{jeśli } \xi' \in \mathcal{N}(\xi) \\ \frac{|V|-m(\xi)}{2|V|} & \text{jeśli } \xi' = \xi \\ 0 & \text{jeśli } |m(\xi) - m(\xi')| \geq 2 \end{cases}$$

Fakt 2.4 Łańcuch Markowa z macierzą prawdopodobieństwa $\mathbf{P}$ jest nieredukowalny i nieokresowy z rozkładem stacjonarnym π_G zadany w (2.10).

2.4 Rozkład Boltzmana na grafie $G = (V, E)$.

Rozważmy teraz graf $G = (V, E)$ z funkcją kosztu (ang. cost function) $H : V \rightarrow \mathbb{R}_+ \cup \infty$. Tutaj V jest zbiorem wierzchołków a E zbiorem krawędzi.

Będziemy rozważać drogi nie zawierające dwóch tych samych krawędzi. Innym opisem, który ma znaczenie w rozważanej teorii jest *sąsiedztwo*. Dla wierzchołka $i \in V$, przez $\mathcal{N}(i)$ oznaczamy zbiór tych $j \neq i$ dla których istnieje krawędź łącząca i z j . Mówimy wtedy, że j jest sąsiadem i . Istotnym faktem w tym podrozdziale jest, że błądzimy tylko po sąsiadach.

Jednym z ważnych problemów jest szukanie minimum funkcji H , ale o tym później. Przez D^* oznaczamy zbiór elementów z V na którym funkcja H osiąga globalne minimum. Interesuje nas rozkład Boltzmana na V

$$\pi_i(T) = \frac{e^{-H(i)/T}}{\sum_{j \in V} e^{-H(j)/T}}, \quad i \in V.$$

Z uogólnionego algorytmu Metropolis-Hastingsa (patrz fakt 2.2) aby generować rozkład $\pi_i(T)$ na V musimy położyć $p_{ij}(T) = q_{ij}\alpha_{ij}(T)$, gdzie Niech $\mathbf{Q}$ jest pewną macierzą generującą kandydata i

$$\alpha_{ij} = \alpha_{ij}(T) = \frac{1}{\deg(i)} \min \left(1, \frac{\deg(i)}{\deg(j)} e^{(H(i)-H(j))/T} \right), \quad i, j \in V$$

jest macierzą prawdopodobieństwa akceptacji.

Lemat 2.5 Rozkładem stacjonarnym dla macierzy przejścia $\mathbf{P}(T)$ zdefiniowanej przez

$$p_{ij}(T) = q_{ij}\alpha_{ij}(T) \tag{2.11}$$

jest

$$\pi_i(T) = \frac{e^{-H(i)/T}}{\sum_{j \in V} e^{-H(j)/T}}. \quad (2.12)$$

Można pokazać ten lemat przez własność DBC (pokazać; zad⁴)

Symulowane wyżarzanie

Symulowane wyżarzanie (od angielskiego *simulated annealing*) dalej w skrócie SA, jest metodą probabilistyczną znalezienia globalnego minimum funkcji H (czasami zwanej funkcją kosztu) posiadającej więcej niż jedno lokalne minimum. Przykładowe zastosowania są m.in. do zagadnienia komiwojażera, wprowadzonego w rozdziale I.

Jest to problem optyimizacyjny, który ogólnie można sformułować następująco. Zadaniem jest znalezienie globalnego minimum funkcji $H : V \rightarrow \mathbb{R}_+ \cup \infty$. Niech $D^* = \{j \in V : H(j) = \min_{i \in V} H(i)\}$. Pomysł na algorytm podpowiada następujący fakt.

Fakt 2.6

$$\lim_{T \rightarrow 0} \pi_i(T) = \begin{cases} \frac{1}{|D^*|} & i \in D^* \\ 0 & \text{w przeciwnym razie} \end{cases}$$

Dowód Niech $i \in D^*$. Mamy

$$\begin{aligned} \pi_i(T) &= \frac{e^{-H(i)/T}}{\sum_{j \in D^*} e^{-H(j)/T} + \sum_{j \in \mathcal{E}-D^*} e^{-H(j)/T}} \\ &= \frac{1}{\sum_{j \in D^*} 1 + \sum_{j \in \mathcal{E}-D^*} e^{-H(j)+H(i)/T}} \rightarrow \frac{1}{|D^*|}. \end{aligned}$$

□

Niech $Y_0, Y_1, \dots$ będzie łańcuchem Markowa z macierzą prawdopodobieństw przejścia

$$p_{ij}(T) = q_{ij} \exp\left(-\frac{1}{T}(H(j) - H(i))_+\right).$$

Jeśli są spełnione warunki twierdzenia Metropolis'a (fakt 2.1), to ten łańcuch ma rozkład stacjonarny $\pi_j(T)$ zadany przez (2.12).

Powyższy fakt daje podstawy heurystyczne dla następującego algorytmu korzystającego z mechanizmu *schładzania*. Jest to zadany ciąg temperatur

⁴Dac zadanie

$T_1 \geq T_2 \geq \dots \rightarrow 0$, i definiuje się tym razem niejednorodny w czasie łańcuch Markowa $Z_0, Z_2, \dots$ z macierzą prawdopodobieństwa n -tym kroku $\mathbf{P}(T_n)$.

Przejście od stanu i do stanu j podane jest poniżej.

Algorytm SA

1. Z_n jest w stanie i
2. Losuj j z sąsiedztwa i z V z prawdopodobieństwem $q_{\{ij\}}$
3. jeśli $H(j) \leq H(i)$, to $Z_{n+1} = j$
4. jeśli $H(j) > H(i)$, to $Z_{n+1} = j$ z prawdopodobieństwem $\exp(-(H(j) - H(i))/T_n)$, oraz $Z_{n+1} = i$ w przeciwnym razie.

Matematyczne podstawy do konstrukcji algorytmu podaje twierdzenie Hajka (1988)⁵. Można postawić następujące pytania.

- Czy Z_n “zbiega” do D^* ,
- jak szybko.

Zdefiniujmy co rozumiemy przez “zbiega”. Mówimy, że algorytm SA z schładzaniem T_n jest zbieżny do D^* , jeśli

$$\lim_{n \rightarrow \infty} \mathbb{P}(Z_n \in D^*) = 1.$$

Dokładniej okazuje się, że zbieżność (Z_n) jest tylko według prawdopodobieństwa do rozkładu jednostajnego D^* . Aby wysławić twierdzenie Hajka potrzebujemy pojęcia “stan i komunikuje się z D^* na wysokości h ”. Przez *drogę* (ang. path) z i do D^* rozumiemy ścieżkę z i do jakiegoś stanu z D^* . Natomiast, że jest na wysokości h oznacza, że najwyższe przewyższenie według H wzdłuż ścieżek jest h . Przez d^* rozumiemy najmniejszą wartość taką, że każdy stan $i \in V$ komunikuje się z D^* na wysokości h^* .

Twierdzenie 2.7 *Algorytm SA z schładzaniem (T_n) jest zbieżny wtedy i tylko wtedy gdy $T_n \rightarrow 0$ i*

$$\sum_n \exp(-d^*/T_n) = \infty.$$

⁵Hajek, B. 1988, Cooling schedules for simulated annealing, Math. Oper. Res. **13**, 311–329

A więc mamy zbieżność wtedy i tylko wtedy gdy

$$\sum_n \exp(-d/T_n) = \infty$$

dla pewnego $d \geq d^*$. Najpopularniejszym wyborem schładzania jest $T_n = d/\log n$. W pracy Geman & Geman pokazano wcześniej, że $d \geq \Delta = |V|(\max_{i \in V}(H(i) - \min H(i)))$.

Szczególną metodą schładzania metodą jest z kawałkami stałą temperaturą. Mianowicie k -ta zmiana jest w chwili n_k , gdzie $n_1 = e^d$ oraz $n_{k+1} = n_k + e^{dk}$ oraz w przedziale $n_k \leq n < n_{k+1}$ temperatura wynosi $1/k$. Zostawiamy czytelnikowi do pokazania, że warunek z twierdzenia Hajka jest spełniony.⁶

2.5 Zastosowanie algorytmu SA do problemu komiwojażera

W przykładzie [??ex.zagadnienie??] wprowadziliśmy tzw. zagadnienie komiwojażera. Zaczniemy od przykładu. Odległości między miastami są dane w tablicy 2.1 Należy znaleźć najkrótszą trasę z wychodzącą z Kutna, prze-

	Kutno	W-wa	Poznań	Kraków
Kutno	0	130	180	300
W-wa	130	0	320	350
Poznań	180	320	0	360
Kraków	300	350	360	0

Tablica 2.1: Odległości między miastami w km.

chodzącą jednokrotnie przez wszystkie pozostałe miasta i kończącą w Kutnie.

Ogólnie problem sformułujemy następująco. Dane jest n miast $1, \dots, n$ oraz miasto wyjściowe oznaczone numerem 0. Drogą jest $\xi = (0, \mathbf{e}, 0)$, gdzie $\mathbf{e}$ jest permutacją $1, \dots, n$. Zbiór wszystkich dróg ξ będzie zbiorem wierzchołków V . Teraz wprowadzamy strukturę sąsiedztwa. Niech

$$\xi = (0, e_1, \dots, e_i, \dots, e_j, \dots, e_n, 0) \in V$$

⁶ $n_k = e^d(e^{dk} - 1)/(e^d - 1) \approx \text{Const}e^{dk}$, skąd $k \approx \log n/d$.

. Niech $i < j$, i przepisujemy ξ zaznaczając blok (i, j)

$$\xi = (0, e_1, \dots, e_{i-1}, \underbrace{e_i, \dots, e_j}_{}, e_{j+1}, \dots, e_n, 0).$$

Wtedy ξ' jest sasiadem ξ (tj. $\xi' \in \mathcal{N}(\xi)$) jeśli

$$\xi' = (0, e_1, \dots, e_{i-1}, \underbrace{e_j, e_{j-1}, \dots, e_i}_{}, e_{j+1}, e_{j+2}, \dots, e_n, 0).$$

A więc każdy ξ ma $n(n-1)/2$ sąsiadów. Macierz przejścia $\mathbf{Q}$ jest zdefiniowana przez błędzenie przypadkowe po sąsiadach jak w przykładzie 1.9. A więc

$$p_{\xi\xi'}(T_n) = \begin{cases} \frac{2}{n(n-1)} \min(e^{(H(\xi)-H(\xi'))/T_n}, 1), & \xi' \in \mathcal{N}(\xi) \\ 0, & \text{w przeciwnym razie.} \end{cases}.$$

Rozważmy teraz $\mathbf{P}(n) = \mathbf{P}(T_n)$ dla pewnego schematu schładzania spełniającego warunki twierdzenia Hajka. Niech $Z_0, Z_1, \dots$ będzie stanem łańcucha po n krokach. Wiemy, że (Z_n) jest zbieżny do zbioru stanów D^* . Można pokazać, że dla schematu $T_n = d/\log n$ ($d \geq d^*$)

$$\max_{i \in V} \mathbb{P}(Z_n \neq D^* | Z_0 = i) \geq A/t^a,$$

gdzie stałe A, a są dodatnio zależne od funkcji H i systemu sąsiedztwa. A więc jeśli chcemy, żeby z prawdopodobieństwem mniejszym od ϵ mielibyśmy $Z_n \notin D^*$ musimy mieć $n \geq (A/\epsilon)^{1/a}$.

2.6 Trochę teorii niejednorodnych łańcuchów Markowa

Stany $Z_0, Z_1, \dots$ w metodzie SA tworzą *niejednorodny łańcuch Markowa* zadany przez kolekcję macierzy przejścia $\mathbf{P}(n) = \mathbf{P}(T_n)$ ($n = 1, 2, \dots$).

Ogólnie definiujemy niejednorodny łańcuch Markowa $(X_n)_{n=0,1,\dots}$ przyjmujący wartości w przestrzeni stanów $\mathcal{E}$ zadany przez kolekcję macierzy przejścia $\mathbf{P}(n) = \mathbf{P}(T_n)$ ($n = 1, 2, \dots$) i rozkład początkowy $\boldsymbol{\mu}$ jeśli

$$\mathbb{P}(X_0 = i_0, \dots, X_k = i_k) = \mu_{i_0} p_{i_0 i_1}(1) \cdots p_{i_{k-1} i_k}(k)$$

dla dowolnych $i_0, \dots, i_k \in \mathcal{E}$. A więc

$$p_{ij}(k) = \mathbb{P}(Z_{k+1} = j | X_k = i).$$

⁷Uwaga w pracy Bertsimasa i Tsitsiklisa.

Oznaczmy przez $\mathbf{P}(k, l) = \mathbf{P}(k)\mathbf{P}(k+1) \cdots \mathbf{P}(l)$ dla $k < l$. Oczywiście

$$\mathbb{P}(X_l = j | X_k = i) = (\mathbf{P}(k, l))_{ij}.$$

Będziemy potrzebowali do następnej definicji pojęcia odległości według wahanía pomiędzy dwoma rozkładami $\boldsymbol{\mu}, \boldsymbol{\nu}$ na $\mathcal{E}$:

$$d_{\text{tv}}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \frac{1}{2} \sum_{i \in \mathcal{E}} |\mu(i) - \nu(i)|.$$

⁸ Wprowadza się pojęcie mocnej ergodyczności, jeśli istnieje rozkład $\boldsymbol{\pi}$ na $\mathcal{E}$ taki, że dla każdego $k \geq 0$

$$\lim_{l \rightarrow \infty} \sup_{\boldsymbol{\mu}} d_{\text{tv}}(\boldsymbol{\mu} \mathbf{P}(k, l), \boldsymbol{\pi}) = 0.$$

Wracając do sekcji o SA okazuje się, że jeśli mamy schładzani $T_n = \Delta / \log n$ (w oznaczeniach sekcji 2.4), to $Z_n \rightarrow_{\text{tv}} \boldsymbol{\pi}$ (jest to zbieżność wg wahanía, ale ponieważ przestrzeń stanów jest skończona, jest to zwykła zbieżność wg rozkładu), gdzie $\boldsymbol{\pi}$ jest rozkładem jednostajnym na D^* .⁹

2.7 Rozkłady Gibbsa

Rozpatrzmy skończony graf (V, E) . Na V definiujemy konfiguracje jako funkcje $\lambda : V \rightarrow D$, gdzie D jest skończonym zbiorem. Wszystkie dozwolone konfiguracje oznaczmy przez Λ . Na przykład $D = \{c, b\}$ określa kolory wierzchołków odpowiednio biały i czarny. Teraz definiujemy funkcję H od konfiguracji λ (nazywana energią tej konfiguracji). Na przykład

$$H(\lambda) = \sum_{v \in V} U(\lambda(v)),$$

lub

$$H(\lambda) = \sum_{v' \in \mathcal{N}(v)} W(v, v') + \sum_{v \in V} U(\lambda(v)),$$

gdzie U, W są nieujemnymi funkcjami. Rozkład Gibbsa na zbiorze konfiguracji Λ jest zdefiniowany przez funkcję prawdopodobieństwa

$$p(\lambda) = Z(T) e^{-H(\lambda)/T},$$

⁸Zadanie: Pokazać, że $d_{\text{tv}}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{E \subset \mathcal{E}} |\boldsymbol{\mu}(E) - \boldsymbol{\nu}(E)|$.

⁹Bremaud[8]: str. 314

gdzie $Z(T)$ jest tak zwaną *funkcją partycji*

$$Z(T) = \sum_{\lambda \in \Lambda} p(\lambda).$$

W zastosowaniach problemem jest trudność obliczenia $Z(T)$. Ale nawet nieznając $Z(T)$ możemy generować (w przybliżeniu) element losowy na Λ o szukanym rozkładzie Gibbsa przy użyciu algorytmu Metropolisa w następującej formie.

3 Uwagi bibliograficzne

O metodzie MCMC można przeczytać w książce Asmussena i Glynn [4], Bremaud [8], Häggström [17]. Metoda simulated annealing jest omówiona w pracy Bersimas i Tsitsiklis [6].

4 Zadania

10

Zadania teoretyczne

4.1 Napisać algorytm na generowanie błędzenia przypadkowego na hiperkostce $\{0, 1\}^m$.

4.2 Niech $B = \sum_{i=1}^{\infty} i^{-2}$. Ta wielkość jest znana ($B = \pi^2/6$) ale to jest nam niepotrzebne. Zaprojektować przy użyciu metody Metropolis symulację rozkładu $\pi_i = 1/Bi^2$, $i = 1, 2, \dots$. Wsk. Sąsiedzi dla dowolnego $i > 1$ to $i - 1, i + 1$ oraz dla $i = 1$ sąsiadem jest tylko 2.

Rozdział IX

Symulacja procesów matematyki finansowej

W tym rozdziale pokażemy wybrane przykłady zastosowań symulacji w matematyce finansowej. Klasyczna matematyka finansowa opiera się na ruchu Browna (RB) z którego łatwo otrzymać geometryczny ruch Browna (GRB). Bardziej skomplikowane procesy są opisywane przez stochastyczne równania różniczkowe (SDE). Niesposób tu formalnie wyłożyć dość zaawansowaną współcześnie analizę stochastyczną, ale postaramy się w sposób przystępny opisać symulowane procesy (tutaj i w dodatku XI.2).

Zauważmy na początku, że chociaż realizacje badanych tutaj procesów są ciągłe, to czegoś takiego nie można wylosować na komputerze. Jedyne co możemy zrobić, to zamiast wylosowania realizacji $X(t), 0 \leq t \leq T$, losujemy realizację $(X(0), X(t_1), \dots, X(t_n))$ w momentach $0 = t_0, t_1, \dots, t_n \leq T$. Będziemy przyjmowali, że $t_1 < \dots < t_n$. Rozważane procesy są procesami dyfuzyjnymi, a więc z ciągłymi realizacjami procesy Markowa. Dla nas to oznacza, że rozkład $X(t_j)$ pod warunkiem $X(t_{j-1}), \dots, X_0$ zależy jedynie od $X(t_{j-1})$ ($j = 1, \dots, n$). A więc będziemy losowali rekurencyjnie, wiedząc, że

$$X(t_{j-1}) = y$$

losujemy $X(t_j)$. Do tego nam jest potrzebna znajomość funkcji przejścia $p_t(x, dy)$ i algorytm jak generować zmienną losową o rozkładzie $p_t(x, dy)$ dla dowolnych ustalonych t i x . Niestety znajomość funkcji $p_t(x, dy)$ jest wyjątkowa, nie mówiąc już o algorytmie generowania tego rozkładu. Dlatego będziemy też omawiać metody numeryczne dla stochastycznych równań różniczkowych, w szczególności tzw. metodę Eulera. Jednakże w odróżnieniu do

rozważań poprzednich w wyniku zastosowania metod numerycznych otrzymamy realizację aproksymacji procesu, a więc $\tilde{X}(t_1), \dots, \tilde{X}(t_n)$. Jeśli chcemy mieć proces stochastyczny z ciągłymi realizacjami, możemy połączyć punkty $X(0), \tilde{X}(t_1), \dots, \tilde{X}(t_n)$ w sposób ciągły, na przykład prostymi, i taka losowa łamana będzie przybliżała proces $X(t), 0 \leq t \leq T$. Zwróćmy uwagę, że mamy do czynienia z dwoma błędami; błąd spowodowany metodą Eulera oraz błąd dyskretyzacji. Zaczniemy od przykładu.

Przykład 0.1 Niech $\{S(t), 0 \leq t \leq T\}$ będzie ewolucją ceny akcji. Przez opcję azjatycką nazywamy możliwość zakupu akcji za cenę K jeśli cena w chwili T spełnia warunek $S(T) \geq K$, w przeciwnym razie rezygnację z zakupu. Jest to tzw. opcja europejska *call*. Inną opcją jest tzw. azjatycka, gdzie zamiast wartości $S(T)$ bierzemy po uwagę uśrednioną cenę $\int_0^T S(s) ds/T$. W takich razie korzyść kupującego jest odpowiednio

$$(S(T) - K)_+ \quad \text{lub} \quad \left(\frac{\int_0^T X(s) ds}{T} - K \right)_+.$$

Jeśli więc r jest bezryzykownym natężeniem stopy procentowej, to cena takich opcji byłaby

$$e^{-rT} \mathbb{E}(S(T) - K)_+ \quad \text{lub} \quad e^{-rT} \mathbb{E} \left(\frac{\int_0^T X(s) ds}{T} - K \right)_+.$$

Przypuśćmy, że znamy proces $S(t)$, który opisuje ewolucję ceny akcji. Matematycznie jest to zazwyczaj opisane przez stochastyczne równanie różniczkowe. Ponieważ nie mamy możliwości symulacji realizacji $\{S(t), 0 \leq t \leq T\}$ więc symulujemy $\tilde{S}(0), \tilde{S}(T/N), \dots, \tilde{S}(T)$. Dla opcji azjatyckiej będziemy obliczali jej cenę za pomocą

$$Y = e^{-rT} \left(\frac{\sum_{j=1}^N \tilde{S}(jT/N)}{T} - K \right)_+.$$

A więc estymator zdefiniowany przez Y nie będzie nieobciążony, mimo, że czasami rozkład $\tilde{S}(0), \tilde{S}(T/N), \dots, \tilde{S}(T)$ pokrywa się z rozkładem $S(0), S(T/N), \dots, S(T)$ (o takich przypadkach będziemy dalej mówili).

Jednakże nie zawsze jesteśmy w stanie symulować rozkład dokładny, ze względu na to, że nie znamy tego rozkładu ani procedury jego generowania. W przypadku gdy proces $S(t)$ jest zadany przez stochastyczne równanie

różniczkowe, to istnieje procedura numeryczna, generowania aproksymacji $\tilde{S}(0), \tilde{S}(T/N), \dots, \tilde{S}(T)$, który to wektor ma jedynie rozkład przybliżony do $S(0), S(T/N), \dots, S(T)$. Będzie to metoda Eulera, Milsteina, omawiane w dalszej części tego rozdziału. Możemy interpolować $\tilde{S}(0), \tilde{S}(T/N), \dots, \tilde{S}(T)$ do procesu $\{\tilde{S}(t), 0 \leq t \leq T\}$ w sposób ciągły i następnie badać jak szybko dąży do zera błąd postaci

$$\mathbb{E} \int_0^T |\tilde{S}(t) - S(t)|^p$$

lub

$$\mathbb{E} \sup_{0 \leq t \leq T} |\tilde{S}(t) - S(t)|.$$

Innym błędem zbadanym teoretycznie jest

$$e_s(N) = \mathbb{E} |S(1) - \tilde{S}(1)|.$$

1 Ruch Browna

Ze względu na to, że ruch Browna ma przyrosty niezależne (patrz dodatek XI.2.1) realizację $B(t)$ ($0 \leq t \leq T$) możemy symulować w punktach $t_0 = 0, t_1 = \Delta, 2\Delta, \dots, t_n = T$, gdzie $\Delta = T/n$. Jednakże dla algorytmu który podamy nie jest istotne, że $\Delta = t_i - t_{i-1}$. Możemy wysymulować w punktach $0 < t_1 < \dots < t_{n-1} < t_n = T$. Korzystając z tego, że

$$B(t_i) - B(t_{i-1}) =_d \sqrt{t_i - t_{i-1}} Z_i,$$

gdzie $Z_1, \dots, Z_n$ są iid standardowe normalne. Bardziej ogólny model to ruch Browna z dryfem $\text{BM}(\mu, \sigma)$, gdzie *dryf* $\mu \in \mathbb{R}$ i *współczynnikiem dyfuzji* $\sigma > 0$. Taki proces definiujemy przez

$$X(t) = \sigma B(t) + \mu t \tag{1.1}$$

gdzie $B(t)$ jest standardowym ruchem Browna. W języku stochastycznych równań różniczkowych proces X zdefiniowany w (1.1) możemy zapisać

$$dX(t) = \mu dt + \sigma dB(t). \tag{1.2}$$

Proces $X(t)$ ma dalej niezależne przyrosty (pokazać samemu!) a więc w punktach $0 < t_1 < \dots < t_{n-1} < t_n = T$ wartości tego procesu można otrzymać rekurencyjnie, zadając $X(0) = x$ oraz

$$X(t_{i+1}) = X(t_i) + \mu(t_i - t_{i-1}) + \sigma \sqrt{t_i - t_{i-1}} Z_{i+1}$$

dla $i = 1, \dots, n$, i $Z_1, \dots, Z_n$ są niezależnymi zmiennymi losowymi o jednokowym rozkładzie $N(0,1)$. Wynika to z tego, że $X(t_{i+1}) - X(t_i) = \mu(t_{i+1} - t_i) + \sigma(W(t_{i+1}) - W(t_i))$ ma rozkład normalny $N(\mu(t_{i+1} - t_i), \sigma^2(t_{i+1} - t_i))$.

Zostawiamy czytelnikowi zaproponowanie algorytmu symulacji procesu zadanego

$$dX(t) = a(t) dt + \sigma(t) dW(t)$$

z warunkiem początkowym $X(0) = x$, gdzie $\sigma(t) > 0$.

1.1 Konstrukcja ruchu Browna via most Browna

Przedstawimy teraz inną konstrukcję ruchu Browna, korzystając z mostu Browna. Rozpatrzmy $(B(s), B(t), B(u))$, gdzie $0 \leq u < s < t$.

Rozkład $B(s)$ pod warunkiem $B(u) = x$ i $B(t) = y$ jest normalny z średnią

$$\frac{(t-s)x + (s-u)y}{(t-u)}$$

oraz wariancją

$$\frac{(s-u)(t-s)}{(t-u)}.$$

W szczególności $B(t+h)$ pod warunkiem $B(t) = x$ i $B(t+2h) = y$ ma średnią $(x+y)/2$ i wariancję $h/2$.

Będziemy zakładać, że $T = 1$. Naszym celem jest wygenerowanie

$$b_0^k, \dots, b_{2^k-1}^k, b_{2^k}^k$$

mający łączny rozkład taki jak

$$(B(0), \dots, B((2^k - 1)/2^k), B(1)).$$

Algorytm:

1. Generuj b_0^0, b_1^0 , gdzie $b_0^0 = 0$ oraz $b_1^0 \sim \mathcal{N}(0, 1)$.
2. Mając wygenerowane $b_j^{k-1}, (j = 1, \dots, 2^{k-1})$, zauważamy, że $b_{2j}^k = b_j^{k-1}, (j = 1, \dots, 2^{k-1})$, natomiast dla $j = 2j + 1$, generujemy $b_i^k \sim \mathcal{N}(y, 2^{-k-1})$, gdzie $y = \frac{1}{2}(b_j^{k-1} + b_{j+1}^{k-1})$.

Reprezentację standardowego ruchu Browna. Niech $Z_0, Z_{i,j}; l = 1, 2, \dots, j = 1, \dots, 2^{l-1}$ iid $\sim N(0, 1)$. Definiujemy $b^k(t)$ jako

$$\Delta_0(t)Z_0 + \sum_{l=1}^k 2^{-(l+1)/2} \sum_{i=1}^{2^{l-1}} \Delta_{l,i}(t)Z_{l,i}, \quad t \in [0, 1].$$

Zauważmy, że

$$(b^k(0), b^k(1/2^k), \dots, b^k(1))$$

ma taki sam rozkład, jak

$$(B(0), \dots, B((2^k - 1)/2^k), B(1)).$$

Można pokazać, że dla $k \rightarrow \infty$ mamy zbieżność prawie wszędzie do procesu

$$\Delta_0(t)Z_0 + \sum_{l=1}^{\infty} 2^{-(l+1)/2} \sum_{i=1}^{2^{l-1}} \Delta_{l,i}(t)Z_{l,i}, \quad t \in [0, 1].$$

Ten proces ma ciągłe trajektorie i spełnia warunki ruchu Browna.

2 Geometryczny ruch Browna

Przypuśmy, że

$$X(t) = X(0) \exp(\sigma B(t) + \eta t)$$

gdzie $B(t)$ jest standardowym ruchem Browna. Stosując wzór Ito do $X(t) = f(B(t), t)$, gdzie $f(x, t) = \xi \exp(\sigma x + \eta t)$ mamy

$$\begin{aligned} dX(t) &= \\ &= \frac{\partial}{\partial t} f(B(t), t) dt + \frac{\partial}{\partial x} f(B(t), t) dB(t) + \frac{1}{2} \frac{\partial^2}{\partial x^2} f(B(t), t) \\ &= \eta \xi \exp(\sigma B(t) + \eta t) + \sigma \xi \exp(\sigma B(t) + \eta t) + \frac{1}{2} \sigma^2 \exp(\sigma B(t) + \eta t) \\ &= (\eta + \sigma^2/2) X(t) dt + \sigma X(t) dB(t). \end{aligned}$$

Podstawiając $\mu = \eta + \sigma^2/2$ widzimy, że stochastyczne równanie różniczkowe

$$dX(t) = \mu X(t) dt + \sigma X(t) dB(t)$$

z warunkiem początkowym $X(0) = \xi$ ma rozwiązanie

$$X(t) = \xi \exp((\mu - \sigma^2/2)t + \sigma B(t)).$$

Ten proces nazywamy *geometrycznym ruchem Browna* $X(t)$ i oznaczamy go przez $\text{GBM}(\mu, \sigma)$. Zauważmy, że dla $u < t$

$$\frac{X(t)}{X(u)} = \exp((\mu - \sigma^2/2)(t - u) + \sigma(B(t) - B(u)))$$

skąd widzimy, że wzory w rozłącznych odcinkach

$$\frac{X(t_1) - X(t_0)}{X(t_0)}, \frac{X(t_2) - X(t_1)}{X(t_1)}, \dots, \frac{X(t_n) - X(t_{n-1})}{X(t_{n-1})}$$

są niezależnymi zmiennymi losowymi. Ponieważ przyrosty $B(t)$ są niezależne i normalnie rozłożone, oraz $B(t) - B(u) =_d \sqrt{t - u}Z$, gdzie $Z \sim N(0, 1)$, więc wartości X w punktach $0 = t_0 < t_1 < \dots < t_n$ spełniają rekurencję

$$X(t_{i+1}) = X(t_i) \exp((\mu - \frac{1}{2}\sigma^2)(t_{i+1} - t_i) + \sigma\sqrt{t_{i+1} - t_i}Z_{i+1})$$

z $Z_1, \dots, Z_n$ niezależne o jednakowym rozkładzie $N(0, 1)$.

2.1 Model Vasicka

Rozważmy teraz proces zadany stochastycznym równaniem różniczkowym

$$dX(t) = \alpha(b - X(t))dt + \sigma dB(t)$$

gdzie $\alpha, b, \sigma > 0$. To równanie ma rozwiązanie

$$X(t) = X(0)e^{-\alpha t} + b(1 - e^{-\alpha t}) + \sigma \int_0^t e^{-\alpha(t-s)} dB(s).$$

o czym można się przekonać różniczkując (wg wzoru Ito) to rozwiązanie. Można też pokazać, że jeśli $0 < u < t$ to

$$X(t) = X(u)e^{-\alpha(t-u)} + b(1 - e^{-\alpha(t-u)}) + \sigma \int_u^t e^{-\alpha(t-s)} dB(s). \quad (2.3)$$

¹ A zatem rozkład $X(t)$ pod warunkiem, że $X(u) = \xi$, jest normalny $N(\mu(u, t), \sigma^2(u, t))$, gdzie

$$\mu(u, t) = e^{-\alpha u} + b(1 - e^{-\alpha u})$$

i

$$\sigma^2(u, t) = \frac{\sigma^2}{2}(1 - e^{-2\alpha(t-u)}).$$

Aby to uzasadnić trzeba wiedzieć, że $\int_0^T k(x) dB(x)$ ma rozkład normalny ze średnią 0 i wariancją $\int_0^T k^2(x) dx$. Liczymy teraz wariancję

$$\begin{aligned} \text{Var} \int_u^t e^{-\alpha(t-s)} dB(s) &= \\ \int_0^t e^{-2\alpha(t-s)} ds &= \\ = \frac{1}{2\alpha}(1 - e^{-2\alpha(t-u)}). \end{aligned}$$

Powyższe rozważania uzasadniają następujący algorytm dający dokładny rozkład procesu X w punktach $0 = t_0 < t_1 < \dots < t_n = T$:

$$X(t_{i+1}) = e^{-\alpha(t_{i+1}-t_i)} X(t_i) + \mu(t_i, t_{i+1}) + \sigma(t_i, t_{i+1}) Z_{i+1},$$

gdzie $Z_1, \dots, Z_n$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $N(0,1)$. Po podstawieniu funkcji $\mu(t_i, t_{i+1})$ i $\sigma^2(t_i, t_{i+1})$ i dokonaniu obliczeń otrzymujemy

$$\begin{aligned} X(t_{i+1}) &= \\ = e^{-\alpha(t_{i+1}-t_i)} X(t_i) + b(1 - e^{-\alpha(t_{i+1}-t_i)}) + \sigma \sqrt{\frac{1}{2\alpha}(1 - e^{-\alpha(t_{i+1}-t_i)})} Z_{i+1} \end{aligned} \quad (2.4)$$

1

$$\begin{aligned} X(t) &= X(u)e^{-\alpha(t-u)} + b(1 - e^{-\alpha(t-u)}) + \sigma \int_u^t e^{-\alpha s} dB(s) \\ &= (X(0)e^{-\alpha u} + b(1 - e^{-\alpha u}) + \sigma \int_0^u e^{-\alpha(u-s)} dB(s))e^{-\alpha(t-u)} \\ &\quad + b(1 - e^{-\alpha(t-u)}) + \sigma \int_u^t e^{-\alpha s} dB(s) \end{aligned}$$

Ponieważ naszym celem jest symulacja realizacji procesu X a więc gdy $\max_i(t_{i+1} - t_i)$ jest małe, więc możemy skorzystać z aproksymacji

$$e^{-\alpha(t_{i+1}-t_i)} \approx 1 - \alpha(t_{i+1} - t_i).$$

Po dokonaniu uproszczeń otrzymamy

$$X(t_{i+1}) = \alpha(t_{i+1} - t_i)X(t_i) + \alpha(t_{i+1} - t_i)(b - X(t_i)) + \sigma\sqrt{t_{i+1} - t_i}Z_{i+1}. \quad (2.5)$$

Jak zobaczymy dalej powyższy wzór będzie szczególnym przypadkiem schematu Eulera.

2.2 Schemat Eulera

Formalnie możemy zapisać ogólne równanie stochastyczne jako

$$dX(t) = a(X(t)) dt + b(X(t)) dB(t).$$

Pozostawiamy na boku ważne pytania o istnienie rozwiązań i jednoznaczność. Będziemy milcząco zakładać pozytywną odpowiedź na te pytania. Wtedy aproksymacje $\hat{X}(t_i)$ wartości $X(t_i)$ otrzymamy przez $\hat{X}(0) = X(0)$ i dla $0 = t_0 < t_1 < \dots < t_n = T$

$$\hat{X}(t_{i+1}) = \hat{X}(t_i) + a(\hat{X}(t_i))(t_{i+1} - t_i) + b(\hat{X}(t_i))\sqrt{t_{i+1} - t_i}Z_{i+1},$$

gdzie $Z_1, \dots, Z_n$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $N(0,1)$.

Przykład 2.1 [Proces Coxa Ingersolla Rossa (CIR)] Proces w modelu Vasicka posiada podstawową wadę, że przyjmuje wartości ujemne. Tej wady nie posiada proces pierwiastkowej dyfuzji opisaney stochastycznym równaniem różniczkowym

$$dX(t) = \alpha(b - X(t)) dt + \sigma\sqrt{X(t)} dB(t).$$

Dowodzi się, że jeśli $X(0) > 0$, to $X(t) \geq 0$ dla wszystkich $t > 0$ oraz jeśli ponadto $2\alpha b > \sigma^2$ to $X(t) > 0$ dla wszystkich $t > 0$. Zastosowanie schematu Eulera prowadzi to rekursji:

$$\hat{X}(t_{i+1}) = X(t_i) + \alpha(b - X(t_i))(t_{i+1} - t_i) + \sigma\sqrt{(X(t_i))_+}\sqrt{t_{i+1} - t_i}Z_{i+1},$$

gdzie $Z_1, \dots, Z_n$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $N(0,1)$, i $(a)_+ = \max(0, a)$.

3 Uwagi bibliograficzne

Podstawowym źródłem informacji o symulacji procesów stochastycznych teorii matematyki finansowej jest książka Glassermana [16]. Inną podstawową monografią omawiającą symulacje procesów stochastycznych jest [4].

Zadania teoretyczne

3.1 Dla standardowego ruchu Browna $\{W(t), 0 \leq t \leq 1\}$ definiujemy proces

$$\{\tilde{W}^h(t), 0 \leq t \leq 1\}$$

który jest zgodny z procesem W w punktach $W(h), W(2h), \dots, W(1)$ gdzie $h = 1/N$ i interpolowany liniowo do realizacji ciągłych. Pokazać, że

$$\mathbb{E} \int_0^1 |\tilde{W}^h(t) - W(t)| = c/N^{1/2},$$

gdzie $c = \sqrt{\pi/32}$.

3.2 Niech $X(t) = x + \int_0^t a(s) ds + \int_0^t b(s) dW(s)$, $0 \leq t \leq T$. Pokazać, że jeśli $0 < t_1 < t_2 < \dots < t_n$ można symulować rekurencyjnie, zadając $X(0) = x$ oraz

$$X(t_{i+1}) = X(t_i) + \int_{t_i}^{t_{i+1}} a(s) ds + \sqrt{\int_{t_i}^{t_{i+1}} b^2(s) ds} Z_{i+1}$$

dla $i = 1, \dots, n$, i $Z_1, \dots, Z_n$ są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $N(0,1)$.

Projekt

3.3 Według Asmussen&Glynn (2007). Zaprojektować obliczenie p metodą Monte Carlo, gdzie p jest intensywnością spłaty kredytu w następującym modelu.

Udzielono kredytu w wysokości k na okres T lat, który będzie spłacany ze stałą intensywnością p . Krótkoterminowa stopa kredytowa jest $r(t)$, $0 \leq t \leq T$. Jeśli $s(t)$ oznacza kwotę spłaconą do chwili t ; $s(0) = 0$ raz $q(t)$ oznacza ilość pieniędzy gdyby były trzymane na rachunku, to ignorując pozostałe koszty (administracja, podatki, zysk, itd) wielkość p wyznaczamy z równości $\mathbb{E} s(T) = \mathbb{E} q(T)$.

a) Uzasadnić, że

$$\begin{aligned} ds(t) &= (p + s(t)r(t))dt \\ dq(t) &= q(t)r(t) \end{aligned}$$

b) Jeśli przez $s_p(t)$ oznaczmy proces s obliczony przy intensywności spłaty p oraz $q_k(t)$ oznaczmy proces obliczony przy wartości początkowej $q(0) = k$, to pokazać, że

$$p = \frac{k\mathbb{E} q_1(T)}{\mathbb{E} s_1(T)}.$$

c) Przyjąć, że $r(t)$ jest procesem CIR. W pierwszym wariancie założyć, że jest on stacjonarny, ze średnią równą długoterminowej stopie procentowej c . Do obliczeń przyjąć, że długoterminowa stopa procentowa $c = 7\%$, $T = 5$, oraz dobrać α, σ^2 tak aby realizacje przypominały te z wykresu ...

d) Porównać z sytuacją, gdy $r(t) \equiv c$.

Rozdział X

Sumulacja zdarzeń rzadkich

1 Efektywne metody symulacji rzadkich zdarzeń

Niech będzie dana przestrzeń probabilistyczna $(\Omega, \mathcal{F}, \mathbb{P})$.

Rozważmy rodzinę zdarzeń $(A(x))$ poindeksowaną albo przez $x \in (0, \infty)$ lub $x \in \{1, 2, \dots\}$ i niech

$$I(x) = \mathbb{P}(A(x)).$$

Mówmy, że $(A(x))$ jest *rodziną zdarzeń rzadkich* jeśli

$$\lim_{x \rightarrow \infty} I(x) = 0.$$

Zauważmy, że dla bardzo małych $I(x)$ trzeba raczej kontrolować błąd względny niż bezwzględny. Przez $Y(x)$ oznaczamy zmienną losową taką, że $I(x) = \mathbb{E} Y(x)$. Wtedy

$$\hat{Y}_n = \frac{1}{n} \sum_{j=1}^n Y_j(x),$$

gdzie $Y_1(x), \dots, Y_n(x)$ są niezależnymi kopiami $Y(x)$, jest estymatorem nieobciążonym $I(x)$. A więc na poziomie istotności α (jak zwykle przyjmujemy $\alpha = 0.05$)

$$\begin{aligned} 1 - \alpha &\sim \mathbb{P}\left(\hat{Y}_n(x) - 1.96 \frac{\sigma(x)}{\sqrt{n}} \leq I(x) \leq \hat{Y}_n(x) + 1.96 \frac{\sigma(x)}{\sqrt{n}}\right) \\ &= \mathbb{P}\left(-\frac{1.96 \sigma(x)}{\sqrt{n} I(x)} \leq \frac{I(x) - \hat{Y}_n(x)}{I(x)} \leq \frac{1.96 \sigma(x)}{\sqrt{n} I(x)}\right). \end{aligned}$$

gdzie $\sigma^2(x) = \text{Var} Y(x)$. Stąd widzimy, że aby kontrolować błąd względny musimy rozważać wielkość $\sigma^2(x)/I^2(x)$, gdzie $x \rightarrow \infty$. Mianowicie jeśli na przykład chcemy kontrolować błąd względny na poziomie b_r , to wtedy z równości

$$b_r = \frac{1.96}{\sqrt{n}} \frac{\sigma(I)}{I}$$

skąd mamy

$$n = \frac{1.96^2}{b_r} \frac{\sigma^2(I)}{I^2}.$$

Potrzebę bardziej wysublimowanej teorii przy symulacji zdarzeń rzadkich pokazuje następujący przykład.

Przykład 1.1 Niech $A(x) \in \mathcal{F}$ i $I(x) = \mathbb{P}(A(x))$ jest bardzo małe. Powtarzamy niezależnie zdarzenia $A_1(x), \dots, A_n(x)$. Rozważmy estymator CMC dla $I(x)$ postaci

$$\hat{Y}_n = \frac{1}{n} \sum_{j=1}^n \mathbf{1}_{A_j(x)}.$$

Wtedy $\text{Var}(\mathbf{1}_{A_j(x)})/I^2(x) = I(x)(1-I(x))/I^2(x) \sim I(x)^{-1}$. Jeśli więc chcemy mieć błąd względny nie większy niż b_r , to musi być spełniona równość

$$\frac{1.96}{\sqrt{n}} I(x)^{-1} = b_r$$

czyli

$$n = \frac{1.96^2}{b_r^2} I(x)^{-1}.$$

A więc dla bardzo małych $I(x)$ duża liczba replikacji jest wymagana. Problemem jest jak szybko zbiega $I(x)$ do zera.

Stąd mamy następującą naturalną definicję dla klasy estymatorów zdefiniowanych przez $Y(x)$.

Definicja 1.2 Mówimy, że estymator $\hat{Y}_n(x)$ ma *ograniczony błąd względny* jeśli

$$\limsup_{x \rightarrow \infty} \frac{\text{Var}(Y(x))}{I^2(x)} < \infty.$$

Bardzo często estymatory z ograniczonym błędem względnym trudno znaleźć, lub trudno tą własność udowodnić. Dlatego wprowadza się trochę słabsze wymaganie, które jednakże często charakteryzuje zupełnie dobre estymatory.

2. DWA PRZYKŁADY EFEKTYWNYCH METOD SYMULACJI RZADKICH ZDARZEŃ

Definicja 1.3 Mówimy, że estymator $\hat{Y}$ logarytmicznie efektywny jeśli

$$\liminf_{x \rightarrow \infty} \frac{\log \text{Var}(Y(x))}{2 \log I(x)} \geq 1.$$

Następny fakt pokazuje równoważne wysłowienie tej własności, z której natychmiast będzie widać, że własność logarytmicznej efektywności jest słabsza od własności ograniczonego błędu.

Fakt 1.4 Własność logarytmicznej efektywności jest równoważna temu, że dla $0 < \epsilon < 2$ mamy

$$\limsup_{x \rightarrow \infty} \frac{\text{Var}(Y(x))}{I^{2-\epsilon}(x)} < \infty. \quad (1.1)$$

Do dowodu przyda się lemat.

Lemat 1.5 Niech $a(x) \rightarrow -\infty$. Mamy $\limsup a(x)b(x) < \infty$ wtedy i tylko wtedy gdy $\liminf b(x) \geq 0$.

Dowód Mamy (1.1) wtedy i tylko wtedy gdy

$$\limsup_{x \rightarrow \infty} \log \left(\frac{\text{Var}(Y(x))}{I^{2-\epsilon}(x)} \right) < \infty.$$

To znaczy

$$\limsup_{x \rightarrow \infty} \log I(x) \left(\frac{\log \text{Var}(Y(x))}{2 \log I(x)} - \left(1 - \frac{\epsilon}{2}\right) \right) < \infty.$$

Teraz należy skorzystać z lematu 1.5 a $a(x) = \log I(x)$ i $b(x) = \log \text{Var}(Y(x)) / (2 \log I(x)) - (1 - \frac{\epsilon}{2})$.

2 Dwa przykłady efektywnych metod symulacji rzadkich zdarzeń

Niech $\xi_1, \xi_2, \dots$ będą niezależnymi zmiennymi losowymi o jednakowym rozkładzie z gęstością $f(x)$ i oznaczmy $S_n = \xi_1 + \dots + \xi_n$. Celem naszym będzie obliczenie $I(x) = \mathbb{P}(S_n > (\mu + \epsilon)n)$, gdzie $\mu = \mathbb{E} \xi_1$ lub $I(x) = \mathbb{P}(S_n > x)$. Okazuje się, że efektywne algorytmy zależą od ciężkości ogona rozkładu ξ_1 . W przypadku lekko ogonowym przydatna będzie technika zamiany miary, którą trochę dokładniej omówimy poniżej.

2.1 Technika wykładniczej zamiany miary

Zacznijmy od pojedynczej kratowej zmiennej losowej ξ określonej na przestrzeni probabilistycznej $(\Omega, \mathcal{F}, \mathbb{P})$ z funkcją prawdopodobieństwa $\{p_k\}$. Niech $\hat{m}(\theta) = \sum_k e^{\theta k} p_k$ będzie funkcją tworzącą momenty. Będziemy zakładać, że $\hat{m}(\theta)$ jest skończona w pewnym przedziale otwartym (θ_1, θ_2) takim, że $\theta_0 < 0 < \theta_1$. Dla tych wartości $\theta \in (\theta_0, \theta_1)$ definiujemy nową funkcję prawdopodobieństwa przez

$$\tilde{p}_k^\theta = \frac{e^{\theta k} p_k}{\hat{m}(\theta)}.$$

Można zawsze tak dobrać $(\Omega, \mathcal{F}, \mathbb{P})$ i zmienną losową ξ tak, że $\mathbb{P}(\xi = k) = p_k$ i dla pewnej innej miary probabilistycznej $\tilde{\mathbb{P}}^\theta$

$$\tilde{\mathbb{P}}^\theta(\xi = k) = p_k^\theta.$$

Teraz rozpatrzmy ciąg niezależnych zmiennych losowych $\xi_1, \dots, \xi_n$ na $(\Omega, \mathcal{F}, \mathbb{P})$ o jednakowym rozkładzie takim jak ξ . Natomiast z nową miarą probabilistyczną $\tilde{\mathbb{P}}^\theta$ definiujemy następująco:

$$\begin{aligned} \tilde{\mathbb{P}}^\theta(\xi_1 = k_1, \dots, \xi_n = k_n) \\ &= \frac{e^{\theta(k_1 + \dots + k_n)}}{\hat{m}(\theta)} \mathbb{P}(\xi_1 = k_1, \dots, \xi_n = k_n) \\ &= \tilde{p}_{k_1}^\theta \dots \tilde{p}_{k_n}^\theta, \end{aligned}$$

czyli przy nowej mierze $\xi_1, \dots, \xi_n$ są dalej niezależne. Oznaczając przez $\kappa(\theta) = \log \hat{m}(\theta)$ i podstawiając $S_n = \xi_1 + \dots + \xi_n$ mamy podstawowy związek

$$\begin{aligned} \mathbb{P}(\xi_1 = k_1, \dots, \xi_n = k_n) &= \\ &= e^{-\theta S_n + n\kappa(\theta)} \tilde{\mathbb{P}}^\theta(\xi_1 = k_1, \dots, \xi_n = k_n). \end{aligned} \quad (2.2)$$

Niech teraz

$$A = \{(\xi_1, \dots, \xi_n) \in B\},$$

gdzie $B \subset \mathbb{Z}^n$ i $\tilde{\mathbb{E}}^\theta$ oznacza wartość oczekiwaną zdefiniowaną przez $\tilde{\mathbb{P}}^\theta$. Wtedy mamy następujący lemat, który jest prawdziwy niezależnie od typu rozkładu.

Lemat 2.1

$$\mathbb{P}(A) = \hat{m}^n(\theta) \tilde{\mathbb{E}}^\theta[e^{-\theta(\xi_1 + \dots + \xi_n)}; A] = \tilde{\mathbb{E}}^\theta[e^{n\kappa(\theta) - \theta S_n}; A]. \quad (2.3)$$

2. DWA PRZYKŁADY EFEKTYWNYCH METOD SYMULACJI RZADKICH ZDARZEŃ

Dowód Udowodnimy lemat gdy ξ_1 ma rozkład kratowy. Korzystając z (2.2) mamy

$$\begin{aligned}\mathbb{P}(A) &= \hat{m}(\theta) \sum_{(k_1, \dots, k_n) \in B} e^{-\theta(k_1 + \dots + k_n)} \tilde{\mathbb{P}}^\theta(\xi_1 = k_1, \dots, \xi_n = k_n) \\ &= \hat{m}^n(\theta) \tilde{\mathbb{E}}^\theta[e^{-\theta(\xi_1 + \dots + \xi_n)}; A].\end{aligned}\tag{2.4}$$

□

Lemat 2.2 Dla zmiennej losowej ξ_1 określonej na przestrzeni probabilistycznej $(\Omega, \mathcal{F}, \tilde{\mathbb{P}}^\theta)$ funkcja tworząca momenty

$$\tilde{\mathbb{E}}^\theta e^{s\xi_1} = \hat{m}(s + \theta).$$

Stąd

$$\tilde{\mathbb{E}}^\theta \xi_1 = \frac{d\hat{m}(\theta + s)}{ds} \Big|_{s=0} = \kappa'(\theta).$$

2.2 $\mathbb{P}(S_n > (\mu + \epsilon)n)$ - przypadek lekko-ogonowy

Będziemy obliczać

$$I(m) = \mathbb{P}(S_m > m(\mu + \epsilon)),$$

gdzie $\mathbb{E} \xi_1 = \mu$ i $\epsilon > 0$. Podstawiając $A = A(m) = \{S_m > m(\mu + \epsilon)\}$ w (2.4) mamy, że

$$Y(m) = e^{-\theta S_m + m\kappa(\theta)} \mathbf{1}_{A(m)}$$

przy mierze $\tilde{\mathbb{P}}^\theta$ definiuje nieobciążony estymator $\hat{Y}_n(m)$. Niech θ_0 spełnia

$$\mathbb{E}^{\theta_0} \xi_1 = \kappa'(\theta_0) = \mu + \epsilon.$$

Ponieważ κ jest funkcją wypukłą (własność funkcji tworzących momenty) więc rozwiązanie θ_0 musi być ściśle dodatnie. Nietrywialnym rezultatem jest następujące twierdzenie.

Twierdzenie 2.3 Estymator zdefiniowany przez

$$Y(m) = e^{-\theta S_m + m\kappa(\theta)} \mathbf{1}_{A(m)}$$

przy nowej mierze $\tilde{\mathbb{P}}^{\theta_0}$ jest logarytmicznie efektywny.

Zauważmy ponadto, że optymalny estymator jest typu istotnościowego i można pokazać, że w tej klasie jest on jedyny.

2.3 $\mathbb{P}(S_n > x)$ - przypadek ciężko-ogonowy

Jak zwykle przez $\bar{F}(x) = 1 - F(x)$ oznacza ogon dystrybucyjny ξ_1 . Będziemy rozpatrywać dwa przypadki: Pareto z $\bar{F}(x) = 1/(1+x)^\alpha$ lub Weibull $\bar{F}(x) = e^{-x^r}$ z parametrem $r < 1$. Teoria idzie bez zmian jeśli zamiast Pareto będziemy mieli rozkład Pareto-podobny; patrz definicja w [?]. Niech $A(x) = \{S_n > x\}$. Tym razem n jest ustalone, natomiast parametrem jest $x \rightarrow \infty$. Niech

$$I(x) = \mathbb{P}(S_n > x).$$

W dalszej części przez $\xi_{(1)} < \xi_{(2)} < \dots < \xi_{(n)}$ oznaczamy statystykę porządkową i $S_n = \xi_1 + \dots + \xi_n$, $S_{(n)} = \xi_{(1)} + \dots + \xi_{(n)}$ oraz $M_n = \xi_{(n)} = \max(\xi_1, \dots, \xi_n)$. Zdefiniujemy cztery estymatory $\hat{Y}_n^i$ ($i = 1, \dots, 4$) oparte na

$$\begin{aligned} Y^1(x) &= \mathbf{1}_{A(x)} \\ Y^2(x) &= \bar{F}(x - S_{n-1}) \\ Y^3(x) &= \frac{\bar{F}((x - S_{(n-1)}) \vee \xi_{(n-1)})}{\bar{F}(\xi_{(n-1)})} \\ Y^4(x) &= n\bar{F}(M_{n-1} \vee (x - S_{n-1})). \end{aligned}$$

Estymatory oparte na $Y_i(x)$ ($i = 1, \dots, 4$) są nieobciążonymi estymatorami ponieważ

$$\begin{aligned} \bar{F}(x - S_{n-1}) &= \mathbb{P}(S_n > x | S_{n-1}) \\ \frac{\bar{F}((x - S_{(n-1)}) \vee \xi_{(n-1)})}{\bar{F}(\xi_{(n-1)})} &= \mathbb{P}(S_n > x | \xi_{(1)}, \dots, \xi_{(n-1)}) \\ n\bar{F}(M_{n-1} \vee (x - S_{n-1})) &= n\mathbb{P}(S_n > x, M_n = \xi_n | \xi_1, \dots, \xi_n). \end{aligned}$$

Twierdzenie 2.4 *W przypadku gdy rozkład ξ jest Pareto, to estymator $\hat{Y}_n^3(x)$ jest logarytmicznie efektywnym a estymator $\hat{Y}_n^4(x)$ ma ograniczony błąd względny, natomiast gdy ξ_1 jest Weibullowski z $r < \log(3/2)/\log 2$ to estymator $\hat{Y}_n^4(x)$ estymator jest logarytmicznie efektywny.*

2.4 Uwagi bibliograficzne

Kolekcja prac: Rare Event Simulation using Monte Carlo Methods. Gerardo Rubino i Bruno Tuffin, Editors. Wiley, 2009.

2. DWA PRZYKŁADY EFEKTYWNYCH METOD SYMULACJI RZADKICH ZDARZEŃ

S.Juneja and P.Shahabuddin. Rare event simulation techniques: An introduction and recent advances. In S.G. Henderson and B.L. Nelson, eds, Simulation, Handbooks in Operations Research and Management Science, pp. 291–350. Elsevier, Amsterdam, 2006.

Podrozdziały 2.3 i 2.2 zostały opracowane na podstawie książki Asmussen i Glynn [4], do której odsyłamy po dowody twierdzeń i dalsze referencje.

Rozdział XI

Dodatek

1 Elementy rachunku prawdopodobieństwa

1.1 Rozkłady zmiennych losowych

W teorii prawdopodobieństwa wprowadza się zmienną losową X jako funkcję zdefiniowaną na przestrzeni zdarzeń elementarnych Ω . W wielu przypadkach, ani Ω ani X nie są znane. Ale co jest ważne, znany jest rozkład zmiennej losowej X , tj. przepis jak liczyć $F(B)$, że X przyjmie wartość z zbioru B , gdzie B jest podzbiorem (borelowskim) prostej $\mathbb{R}$.

Ze względu na matematyczną wygodę rozważa się rodzinę $\mathcal{F}$ podzbiorów Ω , zwaną rodziną zdarzeń, oraz prawdopodobieństwo $\mathbb{P}$ na $\mathcal{F}$ przypisujące zdarzeniu A z $\mathcal{F}$ liczbę $\mathbb{P}(A)$ z odcinka $[0, 1]$.

Rozkład można opisać przez podanie *dystribuanty*, tj. funkcji $F : \mathbb{R} \rightarrow [0, 1]$ takiej, że $F(x) = \mathbb{P}(X \leq x)$. Przez $\bar{F}(x) = 1 - F(x)$ będziemy oznaczać *ogon* of F .

W praktyce mamy dwa typy zmiennych losowych: dyskretne i absolutnie ciągłe. Mówimy, że X jest *dyskretną* jeśli istnieje przeliczalny zbiór liczb $E = \{x_0, x_1, \dots\}$ z $\mathbb{R}$ taki, że $\mathbb{P}(X \in E) = 1$. W tym przypadku definiujemy *funkcję prawdopodobieństwa* $p : E \rightarrow [0, 1]$ przez $p(x_k) = \mathbb{P}(X = x_k)$; para (E, p) podaje pełny przepis jak liczyć prawdopodobieństwa $\mathbb{P}(X \in B)$ a więc zadaje rozkład. Najważniejszą podklasą są zmienne losowe przyjmujące wartości w E który jest podzbiorem kraty $\{hk, k \in \mathbb{Z}_+\}$ dla pewnego $h > 0$. Jeśli nie będzie powiedziane inaczej to zakładamy, że $h = 1$, tzn $E \subset \mathbb{Z}_+$, i.e. $x_k = k$. Wtedy piszemy prosto $p(x_k) = p_k$ i mówimy, że rozkład jest *kratowy*.

Z drugiej strony mówimy, że X jest *absolutnie ciągłą* jeśli istnieje funkcja $f : \mathbb{R} \rightarrow \mathbb{R}_+$ taka, że $\int f(x) dx = 1$ i $\mathbb{P}(X \in B) = \int_B f(x) dx$ dla każdego $B \in \mathcal{B}(\mathbb{R})$. Mówimy wtedy, że $f(x)$ jest *gęstością* zmiennej X .

Rozkład dyskretnej zmiennej losowej nazywamy rozkładem dyskretnym. Jeśli X jest kratowy, to jego rozkład też nazywamy kratowy. Analogicznie rozkład zmiennej losowej absolutnie ciągłej nazywamy rozkładem absolutnie ciągłym. Czasami dystrybuanta nie jest ani dyskretna ani absolutnie ciągła. Przykładem jest *mieszanka* $F = \theta F_1 + (1 - \theta)F_2$, gdzie F_1 jest dyskretna z prawdopodobieństwem θ i F_2 jest absolutnie ciągła z prawdopodobieństwem $1 - \theta$.

Aby zaznaczyć, że dystrybuanta, gęstość, funkcja prawdopodobieństwa, jest dla zmiennej losowej X , piszemy czasem $F_X, p_X, f_X, \dots$.

Dla dwóch zmiennych losowych X i Y o tym samym rozkładzie piszemy $X \stackrel{\mathcal{D}}{=} Y$. To samo stosujemy do wektorów.

Mówimy, że wektor losowy $(X_1, \dots, X_n)$ jest *absolutnie ciągły* jeśli istnieje nieujemna i całkowalna funkcja $f : \mathbb{R}^n \rightarrow \mathbb{R}_+$ z

$$\int_{\mathbb{R}} \dots \int_{\mathbb{R}} f(x_1, \dots, x_n) dx_1 \dots dx_n = 1$$

taka, że $\mathbb{P}(X_1 \in B_1, \dots, X_n \in B) = \int_{B_1} \dots \int_{B_n} f(x_1, \dots, x_n) dx_n \dots dx_1$ dla wszystkich zbiorów borelowskich $B_1, \dots, B_n \in \mathcal{B}(\mathbb{R})$. Funkcja $f(x)$ nazywa się *gęstością* $\mathbf{X} = (X_1, \dots, X_n)^T$. Przytoczymy teraz pożyteczny rezultat dotyczący wzoru na gęstość przekształconego wektora losowego. Mianowicie niech będą dane funkcje $g_i : \mathbb{R} \rightarrow \mathbb{R}$, $i = 1, \dots, n$, które spełniają następujące warunki:

1. V jest otwartym podzbiorem $\mathbb{R}^n$, takim że $\mathbb{P}(\mathbf{X} \in V) = 1$,
2. przekształcenie $\mathbf{g} = (g_1, \dots, g_n)^T$ jest wzajemnie jednoznaczne na zbiorze A ,
3. przekształcenie $\mathbf{g}$ ma na V ciągłe pierwsze pochodne cząstkowe ze względu na wszystkie zmienne,
4. jacobian $J_{\mathbf{g}}(\mathbf{x}) = \frac{\partial(g_1, \dots, g_n)}{\partial(x_1, \dots, x_n)}$ jest różny od zera na A .

Oznaczmy przez $\mathbf{h} = (h_1, \dots, h_n)^T$ przekształcenie odwrotne do $\mathbf{g}$, tzn. $\mathbf{x} = \mathbf{h}(\mathbf{y}) = (h_1(\mathbf{y}), \dots, h_n(\mathbf{y}))$ dla każdego $\mathbf{y} \in \mathbf{g}(V)$. Rozważmy wektor $\mathbf{Y} = (\mathbf{X})$. Wektor ten ma gęstość $f_{\mathbf{Y}}$ zadaną wzorem:

$$f_{\mathbf{Y}}(\mathbf{y}) = f_{\mathbf{X}}(h_1(\mathbf{y}), \dots, h_n(\mathbf{y})) |J_h(\mathbf{y})| \mathbf{1}(\mathbf{y} \in \mathbf{g}(V)). \quad (1.1)$$

Jeśli zmienne losowe $(X_1, \dots, X_{n-1})$ ($X_1, \dots, X_n$) są absolutnie ciągłe z odpowiednio z gęstościami $f_{X_1, \dots, X_{n-1}}$ i $f_{X_1, \dots, X_n}$ i jeśli $f_{X_1, \dots, X_{n-1}}(x_1, \dots, x_{n-1}) > 0$, to

$$f_{X_n|X_1, \dots, X_{n-1}}(x_n | x_1, \dots, x_{n-1}) = \frac{f_{X_1, \dots, X_n}(x_1, \dots, x_n)}{f_{X_1, \dots, X_{n-1}}(x_1, \dots, x_{n-1})}$$

nazywa się i *warunkową gęstością* X_n pod warunkiem $X_1 = x_1, \dots, X_{n-1} = x_{n-1}$.

Dla ustalonego n i wszystkich $k = 1, 2, \dots, n$ i $\omega \in \Omega$, niech

$X_{(k)}(\omega)$ oznacza k -tą najmniejszą wartość $X_1(\omega), \dots, X_n(\omega)$. Składowe wektora losowego $(X_{(1)}, \dots, X_{(n)})$ nazywają się *statystyka porządkową* $(X_1, \dots, X_n)$.

1.2 Podstawowe charakterystyki

Niech X będzie zmienną losową. i $g : \mathbb{R} \rightarrow \mathbb{R}$ funkcją. Definiujemy $\mathbb{E}g(X)$ przez

$$\mathbb{E}g(X) = \begin{cases} \sum_k g(x_k)p(x_k) & \text{jeśli } X \text{ jest dyskretny} \\ \int_{-\infty}^{\infty} g(x)f(x)dx & \text{jeśli } X \text{ jest absolutnie ciągły} \end{cases}$$

pod warunkiem odpowiednio, że $\sum_k |g(x_k)|p(x_k) < \infty$ i $\int_{-\infty}^{\infty} |g(x)|f(x)dx < \infty$. Jeśli g jest nieujemna, to używamy symbolu $\mathbb{E}g(X)$ również gdy równa się nieskończoność. Jeśli rozkład jest mieszanką dwóch typów to definiujemy $\mathbb{E}g(X)$ by

$$\mathbb{E}g(X) = \theta \sum_k g(x_k)p(x_k) + (1 - \theta) \int_{-\infty}^{\infty} g(x)f(x)dx. \quad (1.2)$$

Czasami wygodne jest zapisanie $\mathbb{E}g(X)$ w terminach *całki Stieltjesa* t.j.

$$\mathbb{E}g(X) = \int_{-\infty}^{\infty} g(x)dF(x),$$

która jest dobrze określona przy pewnych warunkach na g i F . W szczególności dla $g(x) = x$ wartość $\mu = \mathbb{E}X$ jest nazywana *średnią*, lub *wartością oczekiwaną* lub *pierwszy momentem* X . Dla $g(x) = x^n$, $\mu^{(n)} = \mathbb{E}(X^n)$ jest nazywana n -tym *momentem*. *Momentem centralnym rzędu n* nazywamy $\alpha_n = \mathbb{E}(X - \mu)^n$. *Wariancją* X jest $\sigma^2 = \mathbb{E}(X - \mu)^2$ i $\sigma = \sqrt{\sigma^2}$

jest *odchyleniem standardowym* lub *dyspersją*. Czasami piszemy $\text{Var } X$ zamiast σ^2 . *Współczynnik zmienności* jest zdefiniowany przez $\text{cv}_X = \sigma/\mu$, i *współczynnik asymetrii* przez $\alpha_3\sigma^{-3} = \mathbb{E}(X - \mu)^3\sigma^{-3}$. Dla dwóch zmiennych losowych X, Y definiujemy *kowariancję* $\text{Cov}(X, Y)$ przez $\text{Cov}(X, Y) = \mathbb{E}((X - \mathbb{E} X)(Y - \mathbb{E} Y))$ jeśli tylko $\mathbb{E} X^2, \mathbb{E} Y^2 < \infty$. zauważmy, że równoważnie mamy $\text{Cov}(X, Y) = \mathbb{E} XY - \mathbb{E} X \mathbb{E} Y$. Jeśli $\text{Cov}(X, Y) > 0$, to mówimy, że X, Y są *dodatnio skorelowane*, jeśli $\text{Cov}(X, Y) < 0$ to *ujemnie skorelowane* i *nieskorelowane* jeśli $\text{Cov}(X, Y) = 0$. Przez *współczynnik korelacji* pomiędzy X, Y rozumiemy

$$\text{corr}(X, Y) = \frac{\text{Cov}(X, Y)}{\sqrt{\text{Var}(X)\text{Var}(Y)}}.$$

Medianą zmiennej losowej X jest dowolna liczba is $\zeta_{1/2}$ taka, że

$$\mathbb{P}(X \leq \zeta_{1/2}) \geq \frac{1}{2}, \quad \mathbb{P}(X \geq \zeta_{1/2}) \geq \frac{1}{2}.$$

Przez indykator $\mathbf{1}(A) : \Omega \rightarrow \mathbb{R}$ rozumiemy zmienną losową

$$\mathbf{1}(A, \omega) = \begin{cases} 1 & \text{jeżeli } \omega \in A, \\ 0 & \text{w przeciwnym razie.} \end{cases}$$

A więc jeśli $A \in \mathcal{F}$ to $\mathbf{1}(A)$ jest zmienną losową i $\mathbb{P}(A) = \mathbb{E} \mathbf{1}(A)$. Używa się następującej konwencji, że $\mathbb{E}[X; A] = \mathbb{E}(X \mathbf{1}(A))$ dla zmiennej losowej X i zdarzenia $A \in \mathcal{F}$.

Jeśli chcemy zwrócić uwagę, że średnia, n -ty moment, itd. są zdefiniowane dla zmiennej X lub dystrybunaty F to piszemy $\mu_X, \mu_X^{(n)}, \sigma_X^2, \dots$ or $\mu_F, \mu_F^{(n)}, \sigma_F^2, \dots$

1.3 Niezależność i warunkowanie

Mówimy, że zmienne losowe $X_1, \dots, X_n : \Omega \rightarrow \mathbb{R}$ są *niezależne* jeśli

$$\mathbb{P}(X_1 \in B_1, \dots, X_n \in B_n) = \prod_{k=1}^n \mathbb{P}(X_k \in B_k)$$

dla każdego $B_1, \dots, B_n \in \mathcal{B}(\mathbb{R})$ lub, równoważnie,

$$\mathbb{P}(X_1 \leq x_1, \dots, X_n \leq x_n) = \prod_{k=1}^n \mathbb{P}(X_k \leq x_k)$$

dla wszystkich $x_1, \dots, x_n \in \mathbb{R}$. Nieskończony ciąg $X_1, X_2, \dots$ jest ciągiem niezależnych zmiennych losowych, jeśli każdy skończony podciąg ma tę własność. Mówimy, że dwa ciągi $X_1, X_2, \dots$ i $Y_1, Y_2, \dots$ są niezależne jeśli

$$\mathbb{P}\left(\bigcap_{i=1}^n \{X_i \leq x_i\} \cap \bigcap_{i=1}^m \{Y_i \leq y_i\}\right) = \mathbb{P}\left(\bigcap_{i=1}^n \{X_i \leq x_i\}\right) \mathbb{P}\left(\bigcap_{i=1}^m \{Y_i \leq y_i\}\right)$$

dla wszystkich $n, m \in \mathbb{N}$ i $x_1, \dots, x_n, y_1, \dots, y_m \in \mathbb{R}$. Dla ciągu $X_1, X_2, \dots$ niezależnych i jednakowo rozłożonych zmiennych losowych wygodne jest wprowadzenie generycznej zmiennej losowej X , która ma taki sam rozkład. Mówimy, że zmienna losowa jest niezależna od zdarzenia, jeśli jest niezależna od indykatora $\mathbf{1}(A)$ tego zdarzenia.

Niech $A \in \mathcal{F}$. Warunkowym prawdopodobieństwem zdarzenia A' pod warunkiem A jest

$$\mathbb{P}(A' | A) = \begin{cases} \mathbb{P}(A' \cap A) / \mathbb{P}(A) & \text{jeżeli } \mathbb{P}(A) > 0, \\ 0 & \text{w przeciwnym razie.} \end{cases}$$

Często użyteczne jest wzór na *prawdopodobieństwo całkowite*. Niech $A, A_1, A_2, \dots \in \mathcal{F}$ będzie ciągiem zdarzeń taki, że $A_i \cap A_j = \emptyset$ dla $i \neq j$ i $\sum_{i=1}^{\infty} \mathbb{P}(A_i) = 1$. Wtedy,

$$\mathbb{P}(A) = \sum_{i=1}^{\infty} \mathbb{P}(A | A_i) \mathbb{P}(A_i). \quad (1.3)$$

Warunkową wartość oczekiwaną $\mathbb{E}(X | A)$ zmiennej losowej X pod warunkiem zdarzenia A jest warunkowa wartość oczekiwana względem rozkładu warunkowego z dystrybucją $F_{X|A}$, gdzie $F_{X|A}(x) = \mathbb{P}(\{X \leq x\} \cap A) / \mathbb{P}(A)$. Wtedy wzór na prawdopodobieństwo całkowite można sformułować ogólniej:

$$\mathbb{E} X = \sum_{i=1}^{\infty} \mathbb{E}(X | A_i) \mathbb{P}(A_i). \quad (1.4)$$

Ogólnie można zdefiniować dla dwóch zmiennych losowych X, Y warunkową wartość oczekiwaną $\mathbb{E}[X|Y]$. Niestety formalna definicja w pełnej ogólności wymaga pojęć teorii miary. Natomiast bez trudności można zrozumieć najbardziej przydatne przypadki; gdy zmienna losowa Y jest dyskretna i gdy X, Y mają rozkład ciągły z łączną gęstością $f(x, y)$.

Lemat 1.1 Niech X, Y, Z będą zmiennymi losowymi takimi, że odpowiednie wartości oczekiwane, wariancje i kowariancje istnieją skończone. Wtedy

- (i) $\mathbb{E}[X] = \mathbb{E}[\mathbb{E}[X|Y]]$,
- (ii) $\text{Var}[X] = \mathbb{E}[\text{Var}[X|Y]] + \text{Var}[\mathbb{E}[X|Y]]$,
- (iii) $\text{Cov}(X, Y) = \mathbb{E}[\text{Cov}(X, Y|Z)] + \text{Cov}(\mathbb{E}[X|Z], \mathbb{E}[Y|Z])$.

Twierdzenie 1.2 *Jesli $X_1, \dots, X_n$ są niezależnymi zmiennymi losowymi, i ϕ oraz ψ funkcjami niemalejącymi (lub obie nierosnącymi) n zmiennych, to*

$$\mathbb{E}[\phi(X_1, \dots, X_n)\psi(X_1, \dots, X_n)] \geq \mathbb{E}[\phi(X_1, \dots, X_n)]\mathbb{E}[\psi(X_1, \dots, X_n)]$$

przy założeniu że wartości oczekiwane są skończone.

Dowód Dowód jest indukcyjny względem liczby zmiennych $n = 1, 2, \dots$. Niech $n = 1$. Wtedy

$$(\psi(x) - \psi(y))(\phi(x) - \phi(y)) \geq 0,$$

skąd

$$\mathbb{E}(\psi(X) - \psi(Y))(\phi(X) - \phi(Y)) \geq 0.$$

Teraz rozpisując mamy

$$\mathbb{E}\psi(X)\phi(X) - \mathbb{E}\psi(X)\phi(Y) - \mathbb{E}\psi(Y)\phi(X) + \mathbb{E}\psi(Y)\phi(Y) \geq 0.$$

Przyjmując, że X i Y są niezależne o jednakowym rozkładzie jak X_1 mamy

$$\mathbb{E}[\phi(X_1)\psi(X_1)] \geq \mathbb{E}[\phi(X_1)]\mathbb{E}[\psi(X_1)].$$

Teraz zakładamy, że twierdzenie jest prawdziwe dla $n - 1$. Stąd

$$\begin{aligned} & \mathbb{E}[\phi(X_1, \dots, X_{n-1}, x_n)\psi(X_1, \dots, X_{n-1}, x_n)] \\ & \geq \mathbb{E}[\phi(X_1, \dots, X_{n-1}, x_n)]\mathbb{E}[\psi(X_1, \dots, X_{n-1}, x_n)], \end{aligned}$$

dla wszystkich x_n . Podstawiając $x_n = X_n$ mamy

$$\begin{aligned} & \mathbb{E}[\phi(X_1, \dots, X_{n-1}, X_n)\psi(X_1, \dots, X_{n-1}, X_n)] \\ & \geq \mathbb{E}[\phi(X_1, \dots, X_{n-1}, X_n)]\mathbb{E}[\psi(X_1, \dots, X_{n-1}, X_n)]. \end{aligned}$$

□

1.4 Transformaty

Niech $I = \{s \in \mathbb{R} : \mathbb{E} e^{sX} < \infty\}$. Zauważmy, że I musi być odcinkiem (właściwym) lub prostą półprostą lub nawet punktem $\{0\}$. *Funkcja tworząca momenty* $\hat{m} : I \rightarrow \mathbb{R}$ zmiennej losowej X jest zdefiniowana przez $\hat{m}(s) = \mathbb{E} e^{sX}$. Zauważmy różnice pomiędzy funkcją tworzącą momenty a *transformatą Laplace-Stieltjesa* $\hat{l}(s) = \mathbb{E} e^{-sX} = \int_{-\infty}^{\infty} e^{-sx} dF(x)$ zmiennej losowej X lub dystrybuanty F zmiennej X .

Dla zmiennych kratowych z funkcją prawdopodobieństwa $\{p_k, k \in \mathbb{N}\}$ wprowadza się *funkcję tworzącą* $\hat{g} : [-1, 1] \rightarrow \mathbb{R}$ zdefiniowaną przez $\hat{g}(s) = \sum_{k=0}^{\infty} p_k s^k$.

Niech teraz X będzie dowolną zmienną losową z dystrybuantą F . *Funkcję charakterystyczną* $\hat{\varphi} : \mathbb{R} \rightarrow \mathbb{C}$ zmiennej X jest zdefiniowana przez

$$\hat{\varphi}(s) = \mathbb{E} e^{isX}. \quad (1.5)$$

W szczególności jeśli F jest absolutnie ciągłą z gęstością f , to

$$\hat{\varphi}(s) = \int_{-\infty}^{\infty} e^{isx} f(x) dx = \int_{-\infty}^{\infty} \cos(sx) f(x) dx + i \int_{-\infty}^{\infty} \sin(sx) f(x) dx.$$

Jeśli X jest kratową zmienną losową z funkcją prawdopodobieństwa $\{p_k\}$, to

$$\hat{\varphi}(s) = \sum_{k=0}^{\infty} e^{isk} p_k. \quad (1.6)$$

Jeśli chcemy podkreślić, że transformata jest zmiennej X z dystrybuantą F , to piszemy $\hat{m}_X, \dots, \hat{\varphi}_X$ lub $\hat{m}_F, \dots, \hat{\varphi}_F$.

Zauważmy, też formalne związki $\hat{g}(e^s) = \hat{l}(-s) = \hat{m}(s)$ lub $\hat{\varphi}(s) = \hat{g}(e^{is})$. Jednakże delikatnym problemem jest zdecydowanie, gdzie dana transformata jest określona. Na przykład jeśli X jest nieujemna, to $\hat{m}(s)$ jest zawsze dobrze zdefiniowana przynajmniej dla $s \leq 0$ podczas gdy $\hat{l}(s)$ jest dobrze zdefiniowana przynajmniej dla $s \geq 0$. Są też przykłady pokazujące, że $\hat{m}(s)$ może być ∞ dla wszystkich $s > 0$, podczas gdy w innych przypadkach, że $\hat{m}(s)$ jest skończone $(-\infty, a)$ dla pewnych $a > 0$.

Ogólnie jest wzajemna jednoznaczność pomiędzy rozkładem a odpowiadającą jemu transformatą, jednakże w każdym wypadku sformułowanie twierdzenia wymaga odpowiedniego wysłowienia. Jeśli n -ty moment zmiennej losowej $|X|$ jest skończony, to n -ta pochodna $\hat{m}^{(n)}(0), \hat{l}^{(n)}(0)$ istnieje jeśli tylko $\hat{m}(s), \hat{l}(s)$ są dobrze zdefiniowane w pewnym otoczeniu $s = 0$. Wtedy mamy

$$\mathbb{E} X^n = \hat{m}^{(n)}(0) = (-1)^n \hat{l}^{(n)}(0) = (-i)^n \hat{\varphi}^{(n)}(0). \quad (1.7)$$

Jeśli $\hat{m}(s)$ i $\hat{l}(s)$ są odpowiednio dobrze zdefiniowane na $(-\infty, 0]$ i $[0, \infty)$, to trzeba pochodne w (1.7) zastąpić pochodnymi jednostronnymi, t.j.

$$\mathbb{E} X^n = \hat{m}^{(n)}(0-) = (-1)^n \hat{l}^{(n)}(0+), \quad (1.8)$$

Jeśli X przyjmuje wartość w podzbiorze $\mathbb{Z}_+$ i jeśli $\mathbb{E} X^n < \infty$, to

$$\mathbb{E} (X(X-1)\dots(X-n+1)) = \hat{g}^{(n)}(1-). \quad (1.9)$$

Inną ważną własnością $\hat{m}(s)$ (i również dla $\hat{l}(s)$ i $\hat{\varphi}(s)$) jest własność, że jeśli $X_1, \dots, X_n$ są niezależne to

$$\hat{m}_{X_1+\dots+X_n}(s) = \prod_{k=1}^n \hat{m}_{X_k}(s). \quad (1.10)$$

Podobnie, jeśli $X_1, \dots, X_n$ są niezależnymi zmiennymi kratowymi z $\mathbb{Z}_+$ to

$$\hat{g}_{X_1+\dots+X_n}(s) = \prod_{k=1}^n \hat{g}_{X_k}(s). \quad (1.11)$$

Ogólnie dla dowolnych funkcji $g_1, \dots, g_n : \mathbb{R} \rightarrow \mathbb{R}$ mamy

$$\mathbb{E} \left(\prod_{k=1}^n g_k(X_k) \right) = \prod_{k=1}^n \mathbb{E} g_k(X_k) \quad (1.12)$$

jeśli tylko $X_1, \dots, X_n$ są niezależne.

1.5 Zbieżności

Niech $X, X_1, X_2, \dots$ będą zmiennymi losowymi na $(\Omega, \mathcal{F}, \mathbb{P})$. Mówimy, że ciąg $\{X_n\}$ *zbiega z prawdopodobieństwem 1* do X jeśli

$$\mathbb{P}(\lim_{n \rightarrow \infty} X_n = X) = 1.$$

Piszemy wtedy $X_n \xrightarrow{\text{a.s.}} X$. Niech $X, X_1, X_2, \dots$ będą zmiennymi losowymi. Mówimy, że ciąg $\{X_n\}$ *zbiega według prawdopodobieństwa* do X jeśli dla każdego $\epsilon > 0$

$$\lim_{n \rightarrow \infty} \mathbb{P}(|X_n - X| > \epsilon) = 0.$$

Piszemy wtedy $X_n \xrightarrow{\text{Pr}} X$.

Niech $X, X_1, X_2, \dots$ będą zmiennymi losowymi. Mówimy, że ciąg $\{X_n\}$ *zbiega słabo* lub *zbiega według rozkładu* do X jeśli $F_{X_n}(x) \rightarrow F_X(x)$ dla wszystkich punktów ciągłości F_X . Piszemy wtedy $X_n \xrightarrow{\mathcal{D}} X$. Ta definicja jest równoważna następującej: $X_n \xrightarrow{\mathcal{D}} X$ wtedy i tylko wtedy gdy

$$\lim_{n \rightarrow \infty} \mathbb{E} g(X_n) = \mathbb{E} g(X) \quad (1.13)$$

dla każdej funkcji ciągłej i ograniczonej $g : \mathbb{R} \rightarrow \mathbb{R}$. Ten wariant definicji jest używany do zdefiniowania zbieżności według rozkładu na innych przestrzeniach niż $\mathbb{R}$. Warunkiem dostatecznym dla $X_n \xrightarrow{\mathcal{D}} X$ jest

$$\lim_{n \rightarrow \infty} \hat{m}_{X_n}(s) = \hat{m}_X(s) \quad (1.14)$$

dla odcinka otwartego nie tożsamego z $\{0\}$ gdzie $\hat{m}_X(s)$ istnieje. Ze zbieżności z prawdopodobieństwem 1 wynika zbieżność według prawdopodobieństwa, z której wynika zbieżność według rozkładu.

Następujące własności są często wykorzystywane. Jeśli $X_n \xrightarrow{\mathcal{D}} X$ i $Y_n \xrightarrow{\mathcal{D}} c \in \mathbb{R}$, to

$$X_n Y_n \xrightarrow{\mathcal{D}} cX, \quad X_n + Y_n \xrightarrow{\mathcal{D}} X + c. \quad (1.15)$$

2 Elementy procesów stochastycznych

Przez proces stochastyczny $(X(t))_{t \in T}$ z przestrzenią parametrów T nazywamy kolekcję zmiennych losowych. Za T możemy przyjąć np. $T = [0, \infty)$, $T = [a, b]$, itp. W tym rozdziale $T = [0, \infty)$. Zamiast kolekcji zmiennych losowych możemy rozpatrywać kolekcję wektorów losowych z $\mathbb{R}^d$. Przez przyrost procesu $(X(t))_{t \geq 0}$ na odcinku (a, b) rozumiemy $X(b) - X(a)$. Ważne klasy procesów definiuje się przez zadanie odpowiednich własności przyrostów. Na przykład możemy żądać aby na rozłącznych odcinkach przyrosty były niezależne. Zobaczmy to na specjalnym ważnym przykładzie ruchu Browna.

2.1 Ruch Browna

Przez standardowy ruch Browna (BM(0,1)) rozumiemy proces stochastyczny $(B(t))_{t \geq 0}$ spełniający warunki

1. $B(0) = 0$,

2. przyrosty $B(b_1) - B(a_1), \dots, B(b_k) - B(a_k)$ są niezależne jeśli tylko odcinki $(a_1, b_1], \dots, (a_k, b_k]$ są rozłączne ($k = 2, \dots$),
3. $B(b) \sim N(0, b)$
4. realizacje $B(t)$ są ciągłe.

Ważną własnością ruchu Browna jest *samopodobieństwo* z współczynnikiem $\alpha = 1/2$ tj.

$$B(t) =_d t^{1/2} B(1).$$

Definiuje się, dla pewnej klasy procesów X całkę stochastyczną Itô względem ruchu Browna

$$\int_0^T X(t) dB(t).$$

W szczególności jeśli funkcja podcałkowa k jest deterministyczna z $\int_0^T k^2(x) dx$ to $\int_0^T k(t) dB(t)$ ma rozkład normalny $N(0, \sigma^2)$, gdzie

$$\sigma^2 = \int_0^T k^2(t) dt.$$

Korzystając z pojęcia całki stochastycznej Itô definiujemy różniczkę stochastyczną $dk(B(t), t)$ następująco. Niech $k(x, t)$ będzie funkcję dwukrotnie ciągle różniczkowalną względem zmiennej x i jednokrotnie względem zmiennej t . Wtedy korzystając z tak zwanego wzoru Itô mamy

$$dk(B(t), t) = k_{0,1}(B(t), t) dt + k_{1,0}(B(t), t) dB(t) + \frac{1}{2} k_{2,0}(B(t), t) dt,$$

gdzie $k_{i,0}$ oznacza i -tą pochodną cząstkową k względem x a $k_{0,1}$ pierwszą pochodną cząstkową względem zmiennej t .

3 Rodziny rozkładów

Podamy teraz kilka klas rozkładów dyskretnych i absolutnie ciągłych. Wszystkie są scharakteryzowane przez skończony zbiór parametrów. Podstawowe charakterystyki podane są w tablicy rozkładów w podrozdziale 4.

W zbiorze rozkładów absolutnie ciągłych wyróżniamy dwie podklasy: rozkłady z lekkimi ogonami i ciężkimi ogonami. Rozróżnienie to jest bardzo ważne w teorii Monte Carlo.

Mówimy, że rozkład z dystrybuntą $F(x)$ ma wykładniczo ograniczony ogon jeśli dla pewnego $a, b > 0$ mamy $\overline{F}(x) \leq ae^{-bx}$ dla wszystkich $x > 0$. Wtedy mówimy też że rozkład ma *lekki ogon*. Zauważmy, że wtedy $\hat{m}_F(s) < \infty$ dla $0 < s < s_0$, gdzie $s_0 > 0$. W przeciwnym razie mówimy, że rozkład ma *ciężki ogon* a więc wtedy $\hat{m}_F(s) = \infty$ dla wszystkich $s > 0$.

Uwaga Dla dystrybunaty z ciężkim ogonem F mamy

$$\lim_{x \rightarrow \infty} e^{sx} \overline{F}(x) = \infty \quad (3.1)$$

dla wszystkich $s > 0$.

3.1 Rozkłady dyskretne

Wśród dyskretnych rozkładów mamy:

- *Rozkład zdegenerowany* δ_a skoncentrowany w $a \in \mathbb{R}$ z $p(x) = 1$ jeśli $x = a$ i $p(x) = 0$ w przeciwnym razie.
- *Rozkład Bernoulliego* $\text{Ber}(p)$ z $p_k = p^k(1-p)^{1-k}$ dla $k = 0, 1$; $0 < p < 1$: rozkład przyjmujący tylko wartości 0 i 1.
- *Rozkład dwumianowy* $B(n, p)$ z $p_k = \binom{n}{k} p^k (1-p)^{n-k}$ dla $k = 0, 1, \dots, n$; $n \in \mathbb{Z}_+$, $0 < p < 1$: rozkład sumy n niezależnych i jednakowo rozłożonych zmiennych losowych o rozkładzie $\text{Ber}(p)$. A więc, $B(n_1, p) * B(n_2, p) = B(n_1 + n_2, p)$.
- *Rozkład Poissona* $\text{Poi}(\lambda)$ z $p_k = e^{-\lambda} \lambda^k / k!$ dla $k = 0, 1, \dots$; $0 < \lambda < \infty$: jedna z podstawowych cegiełek na których zbudowany jest rachunek prawdopodobieństwa. Historycznie pojawił się jako graniczny dla rozkładów dwumianowych $B(n, p)$ gdy $np \rightarrow \lambda$ dla $n \rightarrow \infty$. Ważna jest własność zamkniętości ze względu na splot, tzn. suma niezależnych zmiennych Poissonowskich ma znowu rozkład Poissona. Formalnie piszemy, że $\text{Poi}(\lambda_1) * \text{Poi}(\lambda_2) = \text{Poi}(\lambda_1 + \lambda_2)$.

- *Rozkład geometryczny* $\text{Geo}(p)$ z $p_k = (1-p)p^k$ dla $k = 0, 1, \dots$; $0 < p < 1$: rozkład mający dyskretny brak pamięci. To jest $\mathbb{P}(X \geq i+j \mid X \geq j) = \mathbb{P}(X \geq i)$ dla wszystkich $i, j \in \mathbb{N}$ wtedy i tylko wtedy gdy X jest geometrycznie rozłożony.
- *Rozkład obcięty geometryczny* $\text{TGeo}(p)$ z $p_k = (1-p)p^{k-1}$ dla $k = 1, 2, \dots$; $0 < p < 1$: jeśli $X \sim \text{Geo}(p)$, to $X|X > 0 \sim \text{TGeo}(p)$, jest to rozkład do liczby prób Bernoulliego do pierwszego sukcesu, jeśli prawdopodobieństwo sukcesu wynosi $1-p$.
- *Rozkład ujemny dwumianowy* lub inaczej *rozkład Pascala* $\text{NB}(\alpha, p)$ z $p_k = \Gamma(\alpha+k)/(\Gamma(\alpha)\Gamma(k+1))(1-p)^\alpha p^k$ dla $k = 0, 1, \dots$; $\alpha > 0, 0 < p < 1$. Powyżej piszemy

$$\binom{x}{0} = 1, \quad \binom{x}{k} = \frac{x(x-1)\dots(x-k+1)}{k!},$$

dla $x \in \mathbb{R}$, $k = 1, 2, \dots$. Mamy

$$p_k = \binom{\alpha+k-1}{k} (1-p)^\alpha p^k = \binom{-\alpha}{k} (1-p)^\alpha (-p)^k.$$

Co więcej, dla $\alpha = 1, 2, \dots$, $\text{NB}(\alpha, p)$ jest rozkładem sumy α niezależnych i jednakowo rozłożonych $\text{Geo}(p)$ zmiennych losowych. To oznacza na przykład, że podklasa ujemnych rozkładów dwumianowych $\{\text{NB}(\alpha, p), \alpha = 1, 2, \dots\}$ jest zamknięta ze względu na spłot t.j. $\text{NB}(\alpha_1, p) * \text{NB}(\alpha_2, p) = \text{NB}(\alpha_1 + \alpha_2, p)$ jeśli $\alpha_1, \alpha_2 = 1, 2, \dots$ co więcej poprzedni wzór jest spełniony dla $\alpha_1, \alpha_2 > 0$ and $0 < p < 1$.

- *Rozkład logarytmiczny* $\text{Log}(p)$ z $p_k = p^k / (-k \log(1-p))$ for $k = 1, 2, \dots$; $0 < p < 1$: pojawia się jako granica obciętych rozkładów dwumianowych.
- *Rozkład (dyskretny) jednostajny* $\text{UD}(n)$ z $p_k = n^{-1}$ dla $k = 1, \dots, n$; $n = 1, 2, \dots$

3.2 Rozkłady dyskretne wielowymiarowe

- *Rozkład wielomianowy* $\text{M}(n, \mathbf{a})$, z $p_{\mathbf{k}} = \frac{n!}{k_1! \dots k_d!} a_1^{k_1} \dots a_d^{k_d}$ dla $\mathbf{k} = (k_1, \dots, k_d)$ takiego, że $k_i \in \mathbb{Z}_+$ i $k_1 + \dots + k_d = n$; zakładamy, że $\mathbf{a} = (a_1, \dots, a_d)$ jest funkcją prawdopodobieństwa; rozkład sumy n niezależnych wektorów losowych przyjmujący wartości $(0, \dots, 0, 1, 0, \dots, 0)$ (jedynka na i -tym miejscu) z prawdopodobieństwem a_i .

3.3 Rozkłady absolutnie ciągłe

Wśród rozkładów absolutnie ciągłych mamy:

- *Rozkład normalny* $N(\mu, \sigma^2)$ z $f(x) = (2\pi\sigma^2)^{-1/2}e^{-(x-\mu)^2/(2\sigma^2)}$ dla wszystkich $x \in \mathbb{R}$; $\mu \in \mathbb{R}, \sigma^2 > 0$: obok rozkładu Poissona druga cegiełka na której opiera się rachunek prawdopodobieństwa. Pojawia się jako granica sum niezależnych asymptotycznie zaniedbywalnych zmiennych losowych. Rozkłady normalne są zamknięte na branie splotów, tj. $N(\mu_1, \sigma_1^2) * N(\mu_2, \sigma_2^2) = N(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)$.
- *Rozkład wykładniczy* $\text{Exp}(\lambda)$ z $f(x) = \lambda e^{-\lambda x}$ dla $x > 0$; $\lambda > 0$: podstawowy rozkład w teorii procesów Markowa ze względu na własność braku pamięci, t.j. $\mathbb{P}(X > t + s \mid X > s) = \mathbb{P}(X > t)$ dla wszystkich $s, t \geq 0$ jeśli X jest rozkładem wykładniczym.
- *Rozkład Erlanga* $\text{Erl}(n, \lambda)$ z $f(x) = \lambda^n x^{n-1} e^{-\lambda x} / (n-1)!$; $n = 1, 2, \dots$: rozkład sumy n niezależnych i jednakowo rozłożonych $\text{Exp}(\lambda)$ zmiennych losowych. Tak więc, $\text{Erl}(n_1, \lambda) * \text{Erl}(n_2, \lambda) = \text{Erl}(n_1 + n_2, \lambda)$.
- *Rozkład chi kwadrat* $\chi^2(n)$ z $f(x) = (2^{n/2}\Gamma(n/2))^{-1}x^{(n/2)-1}e^{-x/2}$ jeżeli $x > 0$ i $f(x) = 0$ jeżeli $x \leq 0$; $n = 1, 2, \dots$: Rozkład sumy n niezależnych i jednakowo rozłożonych zmiennych losowych, które są kwadratami $N(0, 1)$ zmiennych losowych. To oznacza, że $\chi^2(n_1) * \chi^2(n_2) = \chi^2(n_1 + n_2)$.
- *Rozkład gamma* $\text{Gamma}(a, \lambda)$ z $f(x) = \lambda^a x^{a-1} e^{-\lambda x} / \Gamma(a)$ for $x \geq 0$; z parametrem kształtu $a > 0$ i parametrem skali $\lambda > 0$: jeśli $a = n \in \mathbb{N}$, to $\text{Gamma}(n, \lambda)$ jest rozkładem $\text{Erl}(n, \lambda)$. Jeśli $\lambda = 1/2$ i $a = n/2$, to $\text{Gamma}(n/2, 1/2)$ jest rozkładem $\chi^2(n)$.
- *Rozkład jednostajny* $U(a, b)$ z $f(x) = (b - a)^{-1}$ dla $a < x < b$; $-\infty < a < b < \infty$.
- *Rozkład beta* $\text{Beta}(a, b, \eta)$ z

$$f(x) = \frac{x^{a-1}(\eta - x)^{b-1}}{B(a, b)\eta^{a+b-1}}$$

da $0 < x < \eta$ gdzie $B(a, b)$ oznacza funkcję beta (patrz [??som.spe.fun??]); $a, b, \eta > 0$; jeśli $a = b = 1$, to $\text{Beta}(1, 1, \eta) = U(0, \eta)$.

- *Rozkład odwrotny gaussowski (inverse Gaussian distribution)* $IG(\mu, \lambda)$ z

$$f(x) = (\lambda/(2\pi x^3))^{1/2} \exp(-\lambda(x - \mu)^2/(2\mu^2 x))$$

dla wszystkich $x > 0$; $\mu \in \mathbb{R}, \lambda > 0$.

- *Rozkład statystyki ekstremalnej* $EV(\gamma)$ z $F(x) = \exp(-(1 + \gamma x)_+^{-1/\gamma})$ dla wszystkich $\gamma \in \mathbb{R}$ gdzie wielkość $-(1 + \gamma x)_+^{-1/\gamma}$ jest zdefiniowana jako e^{-x} kiedy $\gamma = 0$. W tym ostatnim przypadku rozkład jest znany jako *Rozkład Gumbela*. Dla $\gamma > 0$ otrzymujemy *rozkład Fréchéta*; rozkład ten jest skoncentrowany na półprostej $(-1/\gamma, +\infty)$. W końcu, dla $\gamma < 0$ otrzymujemy *rozkład ekstremalny Weibulla*, skoncentrowany na półprostej $(-\infty, -1/\gamma)$.

3.4 Rozkłady z ciężkimi ogonami

Przykładami absolutnie ciągłych rozkładów, które mają ciężki ogon są podane niżej.

- *Rozkład logarytmiczno normalny* $LN(a, b)$ z $f(x) = (xb\sqrt{2\pi})^{-1} \exp(-(\log x - a)^2/(2b^2))$ dla $x > 0$; $a \in \mathbb{R}, b > 0$; jeśli X jest o rozkładzie $N(a, b)$ to e^X ma rozkład $LN(a, b)$.
- *Rozkład Weibulla* $W(r, c)$ z $f(x) = rcx^{r-1} \exp(-cx^r)$ dla $x > 0$ gdzie parametr kształtu $r > 0$ i parametr skali $c > 0$; $\bar{F}(x) = \exp(-cx^r)$; jeśli $r \geq 1$, to $W(r, c)$ ma lekki ogon, w przeciwnym razie $W(r, c)$ ma ciężki.
- *Rozkład Pareto* $Par(\alpha)$ z $f(x) = \alpha(1/(1+x))^{\alpha+1}$ for $x > 0$; gdzie eksponent $\alpha > 0$; $\bar{F}(x) = (1/(1+x))^\alpha$.
- *Rozkład loggamma* $LogGamma(a, \lambda)$ z $f(x) = \lambda^a/\Gamma(a)(\log x)^{a-1}x^{-\lambda-1}$ dla $x > 1$; $\lambda, a > 0$; jeśli X jest $\Gamma(a, \lambda)$, to e^X ma rozkład $LogGamma(a, \lambda)$.
- *Rozkład Cauchy'ego* z $f(x) = 1/(\pi(1+x^2))$, gdzie $x \in \mathbb{R}$; rozkład który nie ma średniej z dwustronnie ciężkim ogonem.

3.5 Rozkłady wielowymiarowe

Przykładami absolutnie ciągłych wielowymiarowych rozkładów są podane niżej.

- *Rozkład wielowymiarowy normalny* $N(\mathbf{m}, \Sigma)$ ma wektor losowy $\mathbf{X} = (X_1, \dots, X_n)^T$ z wartością oczekiwaną $\mathbb{E} \mathbf{X} = \mathbf{m}$ oraz macierzą kowariancji Σ , jeśli dowolna kombinacja liniowa $\sum_{j=1}^n a_j X_j$ ma jednowymiarowy rozkład normalny. Niech $\Sigma = \mathbf{A}\mathbf{A}^T$. Jeśli $\mathbf{Z}$ ma rozkład $N((0, \dots, 0), \mathbf{I})$, gdzie $\mathbf{I}$ jest macierzą jednostkową, to $\mathbf{X} = \mathbf{A}\mathbf{Z}$ ma rozkład normalny $N(\mathbf{m}, \Sigma)$.

4 Tablice rozkładów

Następujące tablice podają najważniejsze rozkłady. Dla każdego rozkładu podane są: Średnia, wariancja, współczynnik zmienności zdefiniowany przez $\sqrt{\text{Var } X}/\mathbb{E} X$ (w.zm.). Podane są również: funkcja tworząca $\mathbb{E} s^N$ (f.t.) w przypadku dyskretnego rozkładu, oraz funkcja tworząca momenty $\mathbb{E} \exp(sX)$ (f.t.m.) w przypadku absolutnie ciągłych rozkładów, jeśli tylko istnieją jawne wzory. W przeciwnym razie w tabeli zaznaczamy *. Więcej o tych rozkładach można znaleźć w w podrozdziale XI.

rozkład	parametry	funkcja prawdopodobieństwa $\{p_k\}$
$B(n, p)$	$n = 1, 2, \dots$ $0 \leq p \leq 1, q = 1 - p$	$\binom{n}{k} p^k q^{n-k}$ $k = 0, 1, \dots, n$
$Ber(p)$	$0 \leq p \leq 1, q = 1 - p$	$p^k q^{1-k}; k = 0, 1$
$Poi(\lambda)$	$\lambda > 0$	$\frac{\lambda^k}{k!} e^{-\lambda}; k = 0, 1, \dots$
$NB(r, p)$	$r > 0, 0 < p < 1$ $q = 1 - p$	$\frac{\Gamma(r+k)}{\Gamma(r)k!} q^r p^k$ $k = 0, 1, \dots$
$Geo(p)$	$0 < p \leq 1, q = 1 - p$	$qp^k; k = 0, 1, \dots$
$Log(p)$	$0 < p < 1$ $r = \frac{-1}{\log(1-p)}$	$r \frac{p^k}{k}; k = 1, 2, \dots$
$UD(n)$	$n = 1, 2, \dots$	$\frac{1}{n}; k = 1, \dots, n$

Tablica 4.1: Tablica rozkładów dyskretnych

średnia	wariancja	w.zm.	f.t.
np	npq	$\left(\frac{q}{np}\right)^{\frac{1}{2}}$	$(ps + q)^n$
p	pq	$\left(\frac{q}{p}\right)^{\frac{1}{2}}$	$ps + q$
λ	λ	$\left(\frac{1}{\lambda}\right)^{\frac{1}{2}}$	$\exp[\lambda(s - 1)]$
$\frac{rp}{q}$	$\frac{rp}{q^2}$	$(rp)^{-\frac{1}{2}}$	$q^r(1 - ps)^{-r}$
$\frac{p}{q}$	$\frac{p}{q^2}$	$p^{-\frac{1}{2}}$	$q(1 - ps)^{-1}$
$\frac{rp}{1 - p}$	$\frac{rp(1 - rp)}{(1 - p)^2}$	$\left(\frac{1 - rp}{rp}\right)^{\frac{1}{2}}$	$\frac{\log(1 - ps)}{\log(1 - p)}$
$\frac{n + 1}{2}$	$\frac{n^2 - 1}{12}$	$\left(\frac{n - 1}{3(n + 1)}\right)^{\frac{1}{2}}$	$\frac{1}{n} \sum_{k=1}^n s^k$

Tablica 4.1: Tablica rozkładów dyskretnych (cd.)

rozkład	parametry	f. gęstości $f(x)$
$U(a, b)$	$-\infty < a < b < \infty$	$\frac{1}{b-a}; x \in (a, b)$
$\text{Gamma}(a, \lambda)$	$a, \lambda > 0$	$\frac{\lambda^a x^{a-1} e^{-\lambda x}}{\Gamma(a)}; x \geq 0$
$\text{Erl}(n, \lambda)$	$n = 1, 2, \dots, \lambda > 0$	$\frac{\lambda^n x^{n-1} e^{-\lambda x}}{(n-1)!}; x \geq 0$
$\text{Exp}(\lambda)$	$\lambda > 0$	$\lambda e^{-\lambda x}; x \geq 0$
$N(\mu, \sigma^2)$	$-\infty < \mu < \infty,$ $\sigma > 0$	$\frac{\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}{\sqrt{2\pi}\sigma}; x \in \mathbb{R}$
$\chi^2(n)$	$n = 1, 2, \dots$	$\left(2^{n/2}\Gamma(n/2)\right)^{-1} x^{\frac{n}{2}-1} e^{-\frac{x}{2}}; x \geq 0$
$\text{LN}(a, b)$	$-\infty < a < \infty, b > 0$ $\omega = \exp(b^2)$	$\frac{\exp\left(-\frac{(\log x - a)^2}{2b^2}\right)}{xb\sqrt{2\pi}}; x \geq 0$

Tablica 4.2: Tablica rozkładów absolutnie ciągłych

średnia	wariancja	w.zm.	f.t.m.
$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$	$\frac{b-a}{\sqrt{3}(b+a)}$	$\frac{e^{bs} - e^{as}}{s(b-a)}$
$\frac{a}{\lambda}$	$\frac{a}{\lambda^2}$	$a^{-\frac{1}{2}}$	$\left(\frac{\lambda}{\lambda-s}\right)^a; s < \lambda$
$\frac{n}{\lambda}$	$\frac{n}{\lambda^2}$	$n^{-\frac{1}{2}}$	$\left(\frac{\lambda}{\lambda-s}\right)^n; s < \lambda$
$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$	1	$\frac{\lambda}{\lambda-s}; s < \lambda$
μ	σ^2	$\frac{\sigma}{\mu}$	$e^{\mu s + \frac{1}{2}\sigma^2 s^2}$
n	$2n$	$\sqrt{\frac{2}{n}}$	$(1-2s)^{-n/2}; s < 1/2$
$\exp\left(a + \frac{b^2}{2}\right)$	$e^{2a}\omega(\omega-1)$	$(\omega-1)^{\frac{1}{2}}$	*

Tablica 4.2: Tablica rozkładów absolutnie ciągłych (c.d.)

rozkład	parametry	f. gęstości $f(x)$
Par(α)	$\alpha > 0$	$\alpha \left(\frac{1}{1+x} \right)^{\alpha+1}; x > 0$
W(r, c)	$r, c > 0$ $\omega_n = \Gamma \left(\frac{r+n}{r} \right)$	$rcx^{r-1}e^{-cx^r}; x \geq 0$
IG(μ, λ)	$\mu > 0, \lambda > 0$	$\left[\frac{\lambda}{2\pi x^3} \right]^{\frac{1}{2}} \exp \left(\frac{-\lambda(x-\mu)^2}{2\mu^2 x} \right); x > 0$
PME(α)	$\alpha > 1$	$\int_{\frac{\alpha-1}{\alpha}}^{\infty} \alpha \left(\frac{\alpha-1}{\alpha} \right)^{\alpha} y^{-(\alpha+2)} e^{-x/y} dy$ $x > 0$

Tablica 4.2: Tablica rozkładów absolutnie ciągłych (c.d.)

średnia	wariancja	w.z.	f.t.m.
$\frac{1}{\alpha-1}$	$\frac{\alpha}{(\alpha-1)^2(\alpha-2)}$	$[\frac{\alpha}{\alpha-2}]^{-\frac{1}{2}}$	*
$\alpha > 1$	$\alpha > 2$	$\alpha > 2$	
$\omega_1 c^{-1/r}$	$(\omega_1 - \omega_2^2) c^{-2/r}$	$\left[\frac{\omega_2}{\omega_1^2} - 1 \right]^{\frac{1}{2}}$	*
μ	$\frac{\mu^3}{\lambda}$	$\left(\frac{\mu}{\lambda} \right)^{\frac{1}{2}}$	$\frac{\exp\left(\frac{\lambda}{\mu}\right)}{\exp\left(\sqrt{\frac{\lambda^2}{\mu^2} - 2\lambda s}\right)}$
1	$1 + \frac{2}{\alpha(\alpha-2)}$	$\sqrt{1 + \frac{2}{\alpha(\alpha-2)}}$	$\frac{\int_0^{\frac{\alpha}{\alpha-1}} \frac{x^\alpha}{x-s} dx}{\frac{\alpha^{\alpha-1}}{(\alpha-1)^\alpha}}$
			$s < 0$

Tablica 4.2: Tablica rozkładów absolutnie ciągłych (cd.)

5 Elementy wnioskowania statystycznego

5.1 Teoria estymacji

W teorii estymacji rozpatruje się $(\Omega, \mathcal{F}, \mathbb{P}_\theta)$, gdzie $\theta \in \Theta$ oraz zmienne losowe $X_1, \dots, X_n$ zwane *próbą*. Jeśli ponadto dla każdego θ zmienne $X_1, \dots, X_n$ są niezależne o jednakowym rozkładzie (względem $\mathbb{P}_\theta$) to mówimy o *próbie prostej*. Niech $g : \Theta \rightarrow \mathbb{R}^d$. *Estymatorem* nazywamy dowolną statystykę $T_n = f(X_1, \dots, X_n)$, czyli zmienną losową, która jest funkcją od $X_1, \dots, X_n$. Jeśli

$$\mathbb{E}_\theta T_n = g(\theta),$$

to estymator T jest *estymatorem nieobciążonym* funkcji $g(\theta)$, jeśli

$$T_n \rightarrow g(\theta)$$

prawie wszędzie $\mathbb{P}_\theta$, to T_n jest *estymatorem mocno zgodnym* jeśli jest zbieżność według rozkładu (względem $\mathbb{P}_\theta$) to mówimy wtedy o *estymatorze zgodnym*.

Niech $X_1, X_2, \dots, X_n$ będzie próbą prostą. *Średnią próbkową* nazywamy zmienną

$$\hat{X}_n = \frac{1}{n} \sum_{j=1}^n X_j,$$

natomiast *wariancja próbkowa*

$$\hat{S}_n = \frac{1}{n-1} \sum_{j=1}^n (X_j - \hat{X}_n)^2.$$

Jeśli istnieje pierwszy moment skończony $m_1 = m_1(\theta) = \mathbb{E}_\theta X_1$ to średnia jest nieobciążonym estymatorem μ_1 ; jeśli wariancja σ^2 jest skończona to $\hat{S}_n$ jest nieobciążonym estymatorem σ^2 ; ponadto te estymatory są mocno zgodne.

Do testowania i wyznaczania przedziałów ufności przydatne jest następujące twierdzenie graniczne (patrz np. Asmussen [5]). Przypomnijmy, że α_n jest n -tym momentem centralnym zmiennej X_1 .

Fakt 5.1 *Jeśli $\mu_4 = \mathbb{E} X_1^4 < \infty$ to*

$$\sqrt{n} \begin{pmatrix} \hat{X}_n - m_1 \\ \hat{S}_n - \sigma^2 \end{pmatrix} \rightarrow N(\mathbf{0}, \Sigma),$$

gdzie

$$\Sigma = \begin{pmatrix} \sigma^2 & \alpha_3 \\ \alpha_3 & \alpha_4 - \sigma^4 \end{pmatrix}.$$

5.2 Przedziały ufności

Niech $X_1, \dots, X_n$ będzie próbą prostą na $(\Omega, \mathcal{F}, \mathbb{P}_\theta)$. *Przedziałem ufności na poziomie α* nazywamy odcinek $I = [T_L, T_R]$ spełniający warunki

- jego końce $T_L = T_L(X_1, \dots, X_n) \leq T_R = T_R(X_1, \dots, X_n)$ są funkcjami od próby i nie zależą od θ ,
- $\mathbb{P}_\theta(T_L \leq \theta \leq T_R) \geq 1 - \alpha$, dla każdego $\theta \in \Theta$.

Oczywiście przedział I ufności jest szczególnym przypadkiem zbioru ufności (confidence region). Ogólnie I może być zbiorem losowym zdefiniowanym przez próbę. Jest to pojęcie szczególnie przydatne w przypadku estymacji wielowymiarowej. W szczególności możemy rozpatrywać elipsoidy ufności.

Niech $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n$ będzie próbą prostą z n wektorów losowych o wartości w $\mathbb{R}^d$, z wektorem średniej $\boldsymbol{\mu}$ i macierzą kowariancji $\Sigma = (\sigma_{ij})_{i,j=1,\dots,d}$. *Średnią próbkową* nazywamy wektor

$$\hat{\mathbf{X}}_n = \frac{1}{n} \sum_{j=1}^n \mathbf{X}_j,$$

natomiast próbkową macierzą kowariancji jest

$$\hat{\Sigma}_n = \frac{1}{n-1} \sum_{m=1}^n (\mathbf{X}_m - \boldsymbol{\mu})^T (\mathbf{X}_m - \boldsymbol{\mu}).$$

Średnia próbkowa jest estymatorem niobciążonym wektora wartości oczekiwanej $\boldsymbol{\mu}$, natomiast próbkową macierzą kowariancji jest estymatorem nieobciążonym macierzy kowariancji Σ . Ponadto estymator $\hat{\Sigma}_n$ jest mocno zgodny oraz

$$(\hat{\mathbf{X}}_n - \boldsymbol{\mu})^T \hat{\Sigma}_n^{-1} (\hat{\mathbf{X}}_n - \boldsymbol{\mu}) \xrightarrow{\mathcal{D}} \chi^2(d), \quad (5.2)$$

gdzie $\chi^2(d)$ jest rozkładem chi kwadrat z d stopniami swobody.

5.3 Statystyka Pearsona

Przypuśćmy, że mamy wektor losowy $\mathbf{X}_n = (X_{n,1}, \dots, X_{n,d})$ o rozkładzie wielomianowym $M(n, \mathbf{a})$. Definiujemy statystykę Pearsona

$$C_n(\mathbf{a}) = \sum_{j=1}^d \frac{(X_{n,j} - na_j)^2}{na_j}.$$

Zakładamy, że wszystkie $a_j > 0$. Wtedy (patrz np [44])

$$C_n(\mathbf{a}) \xrightarrow{\mathcal{D}} \chi^2(d-1),$$

gdzie $\chi^2(d-1)$ jest rozkładem chi kwadrat z $d-1$ stopniami swobody.

[L'Ecuyer [27] p. 50, [44]].

5.4 Dystrybuanty empiryczne

Dla próby $X_1, \dots, X_n$ definiujemy *dystrybunatę empiryczną*

$$\hat{F}_n(t) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(X_i \leq t), \quad 0 \leq t \leq 1. \quad (5.3)$$

i statystyką testową

$$D_n = \sup_t |\hat{F}_n(t) - F(t)|.$$

Z twierdzenia Gliwienko–Cantelli mamy, że jeśli $X_1, \dots, X_n$ tworzą próbę prostą z dystrybuantą F to, z prawdopodobieństwem 1, ciąg D_n zmierza do zera. Natomiast, przy założeniu, że F jest funkcją ciągłą, to unormowane zmienne $\sqrt{n}D_n$ zbiegają według rozkładu do rozkładu λ -Kołmogorowa z dystrybuantą

$$K(y) = \sum_{k=-\infty}^{\infty} (-1)^k e^{-2k^2 y^2}, \quad y \geq 0.$$

[Renyi [36] p.493]

5.5 Wykresy kwantylowe

Coraz większą popularnością przy weryfikacji czy obserwacje pochodzą z danego rozkładu cieszą się metody graficzne. Na przykład do tego świetnie nadają się wykresy kwantylowe lub inaczej wykres Q-Q. Możemy też przy użyciu tych wykresów weryfikować czy obserwacje z dwóch prób mają ten sam rozkład.

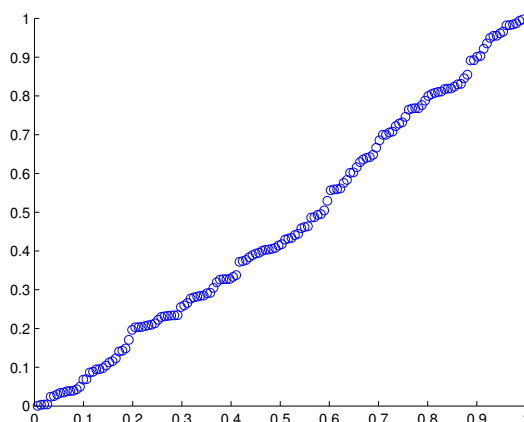
Aby otrzymać wykres kwantylowy należy wykresić w kwadracie kwadracie kartezjańskim $\mathbb{R} \times \mathbb{R}$ z jednej strony wartości funkcji kwantylowej dystrybucji empirycznej utworzonej z obserwacji przeciwko z drugiej strony teoretycznym wartościom funkcji kwantylowej. Przypomnijmy, że $F^{\leftarrow}(x)$ jest uogólnioną funkcją odwrotną. Jeśli F jest dystrybucją, to funkcja Q_F zdefiniowana przez $Q_F(y) = F^{\leftarrow}(y)$ nazywa się *funkcją kwantylową* of F . Jednocześnie definiujemy empiryczną wersję funkcji kwantylowej przez rozpatrzenie uogólnionej funkcji od dystrybucji empirycznej zdefiniowanej w (5.3); to jest $Q_n(y) = Q_{F_n}(y)$ tak że dla próby uporządkowanej $U_{(1)} \leq U_{(2)} \leq \dots \leq U_{(n)}$ mamy

$$\{Q_n(y) = U_{(k)}\} = \{(k-1)n^{-1} < y \leq kn^{-1}\}.$$

Wykres Q-Q jest więc zbiorem punktów $(Q_F(t), Q_n(t)) : 0 < t < 1$. Z definicji widzimy, że empiryczna funkcja kwantylowa $Q_n(y)$ będzie niemalejącą funkcją schodkową; jedynymi punktami wzrostu są wartości $\{k/n, 1 \leq k \leq n\}$, które tworzą kratę na osi x -ów. A więc wykres Q-Q tworzy też niemalejącą funkcję schodkową, ze skokami odpowiadającymi $t = k/n, 1 \leq k \leq n$. Wystarczy więc zrobić wykres w tych punktach tj. wykres $\{(Q_F(k/n), Q_n(k/n)) : k = 1, \dots, n\}$. Jednakże ze względu na drobny problem, że dla $k = n$ mamy $Q_G(k/n) = \infty$, zalecana jest następująca modyfikacja; wykres należy zrobić w punktach $\{x_k = k/(n+1), 1 \leq k \leq n\}$.

Ogólna filozofia wykresów kwantylowych polega na zaobserwowaniu linowości wykresu, co jest łatwe do zobaczenia. Na przykład jeśli chcemy zweryfikować, czy obserwacje pochodzą z rozkładu jednostajnego $U(0,1)$. Wtedy oczywiście $Q_F(x) = x$ i robimy wykres punktów $(x_k, U_{(k)})_{k=1, \dots, n}$. Na rys. 5.1 jest pokazany wykres Q-Q utworzony na podstawie 150 kolejnych rand.

Przykład 5.2 Dla standardowego rozkładu wykładniczego z dystrybucją $G(x)$ funkcja kwantylowa ma postać $Q_G(y) = -\log(1-y)$ jeśli $0 < y < 1$. Jeśli chcemy porównać wartości próbkowe z tymi od standardowego rozkładu wykładniczego wystarczy rozpatrzyć dwie funkcje kwantylowe. Wykreślamy więc dwie funkcje Q_G and Q_n .



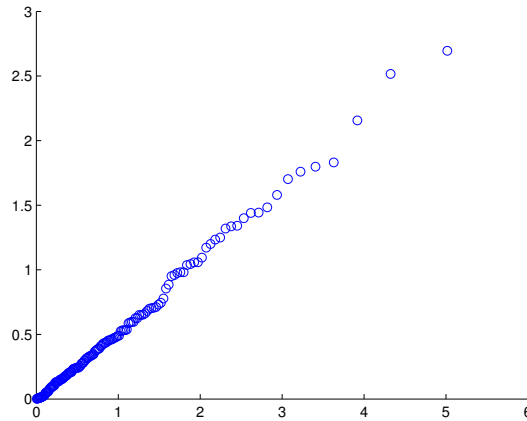
Rysunek 5.1: Jednostajny vs obserwacje jednostajne; 150 obserwacji.

Jeśli nasze obserwacje będą pochodzić z niekoniecznie standardowego rozkładu wykładniczego z parametrem λ tj. z $Q_F(y) = -\lambda^{-1} \log(1 - y)$, to spodziewamy się, że będą układały się wzdłuż pewnej prostej. Nachylenie tej prostej będzie λ^{-1} . Jeśli nasze obserwacje pochodzą z rozkładu o cięższych ogonach, to możemy spodziewać się wykresu rosnącego szybciej niż prosta, jeśli z lżejszego to wzrost będzie wolniejszy. Na rys. 5.1 jest pokazany wykres Q-Q utworzony na podstawie 150 kolejnych $-\log(\text{rand})/2$.

Ogólnie przez “rozkład teoretyczny” vs “obserwacje” wykres kwantylowy będziemy rozumieli wykres QQ, gdzie na osi x -ów mamy teoretyczną funkcję kwantylową a na osi y -ków mamy funkcję kwantylową zbudowaną na bazie dystrybucji empirycznej obserwacji. Możemy mieć też dwie próby. Wtedy będziemy mieli do czynienia “obserwacje I” vs “obserwacje II” wykres kwantylowy.

Jeśli obserwacje nie są dobrze zaprezentowane przez rozkład wykładniczy (patrz na przykład rys. 5.3 jak wyglądają na wykresie Q-Q obserwacje pochodzące z $\text{Pareto}(1.2)$ ustandaryzowane na średnią 1 przeciwko wykładniczej funkcji kwantylowej), to możemy weryfikować inne rozkłady. Wśród popularnych kandydatów znajdują się rozkład normalny, rozkład logarytmiczno normalny oraz rozkład Weibulla. Krótko omówimy te przypadki osobne.

- *Normalny wykres kwantylowy.* Przypomnijmy, że standardowy rozkład



Rysunek 5.2: wykładniczy vs obserwacje wykładnicze; 150 obserwacji.

normalny ma dystrybuantę

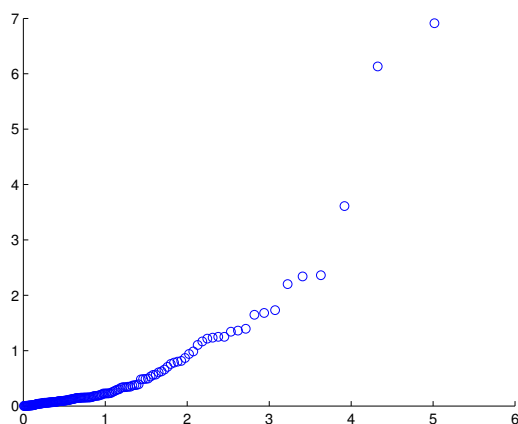
$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-y^2/2} dy$$

i niech $\Phi^{-1}(y)$ będzie odpowiadająca mu funkcją kwantylową. Wtedy standardowy normalny wykres kwantylowy będzie się składał z punktów

$$\{(\Phi^{-1}(k/(n+1)), U_{(k)}), 1 \leq k \leq n\}. \quad (5.4)$$

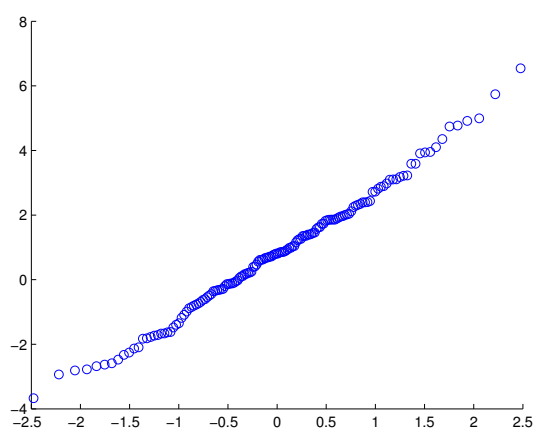
Zauważmy, że dla ogólnego rozkładu normalnego $N(\mu, \sigma^2)$ funkcja kwantylowa $Q(y) = \mu + \sigma\Phi^{-1}(y)$ (pokażać!). Tak więc jeśli normalny wykres kwantylowy jest w przybliżeniu linią, to jej nachylenie da oszacowanie parametru σ podczas gdy przecięcie w 0 daje oszacowanie μ . Na rys. 5.4 pokazany jest normalny wykres kwantylowy dla obserwacji pochodzących z rozkładu normalnego $N(1,2)$. Proponujemy czytelnikowi odczytać a wykresu, że $m = 1$ i $\sigma = 2$.

- *Logarytmiczno-normalny wykres kwantylowy.* Jest to aparat diagnostyczny do wykrywania logarytmiczno normalnego rozkładu. Wykorzystujemy następującą własność. Zmienna X jest logarytmiczno-normalna wtedy i tylko wtedy gdy $\log X$ jest normalnie rozłożoną zmienną losową. A więc wykres kwantylowy będzie zadany przez $\{(\Phi^{-1}(k/(n+1)), \log X_{(k)}), 1 \leq k \leq n\}$.



Rysunek 5.3: Wykładniczy vs obserwacje Pareto; 150 obserwacji.

- Paretowski wykres kwantylowy* jest często wykorzystywanym narzędziem. Przyponijmy, że jeśli U ma rozkład Pareto $\text{Par}(\alpha, c)$ to $\mathbb{P}(U \leq x) = 1 - ((1+x)/c)^{-\alpha}$ dla $x \geq c$, a więc $\log Q(y) = \log c - \alpha^{-1} \log(1-y)$. Paretowski wykres kwantylowy otrzymujemy rysując wykres punktów $\{(-\log(1 - k/(n+1)), \log U_{(k)}), 1 \leq k \leq n\}$. Jeśli dane pochodzą z rozkładu Pareto to wykres będzie miał kształt prostej przecinającej o na wysokości $\log c$ i nachyleniu α^{-1} . Zauważmy, że użycie Paretowskiego wykresu kwantylowego ma również sens gdy rozkład U jest Pareto podobny, tj. $\mathbb{P}(U > x) \sim x^{-\alpha} L(x)$, gdzie L jest funkcją wolno zmieniającą się.
- The Weibullowski wykres kwantylowy* Przypmnijmy, że rozkład ogona w tym wypadku wynosi $\bar{F}(x) = \exp(-cx^r)$. Funkcja kwantylowa jest wtedy $Q(y) = (-c^{-1} \log(1-y))^{1/r}$ dla $0 < y < 1$. Jeśli weźmiemy logarytm z tego wyrażenia, to znajdujemy, że $\log Q(y) = -r^{-1} \log c + r^{-1} \log(-\log(1-y))$ które automatycznie prowadzi do Weibullowskiego wykresu kwantylowego $\{(\log(-\log(1 - k/(n+1))), \log U_{(k)}), 1 \leq k \leq n\}$. Jeśli dane pochodzą z rozkładu Weibulla, to powinniśmy się spodziewać linii prostej o nachyleniu r^{-1} i przecinającej . $-r^{-1} \log c$.



Rysunek 5.4: Normalny vs obserwacje normalne; 150 obserwacji.

5.6 Własność braku pamięci rozkładu wykładniczego

Niektóre algorytmy istotnie wykorzystują tzw. *brak pamięci* rozkładu wykładniczego czasu obsługi i opuszczenie tego założenia będzie wymagało istotnych modyfikacji. Natomiast możemy założyć, że są znane momenty napływania zadań $A_1 < A_2 < \dots$. Tak jest na przykład gdy zgłoszenia są zgodne z niejednorodnym Procesem Poissona. Tak jest na przykład gdy odstępy między zgłoszeniami są niezależne o jednakowym rozkładzie (niewykładniczym). W każdym zdarzeniu (niech będzie ono w chwili t), gdy dotychczasowy algorytm mówił, że należy generować X i podstawić $t_A = t + X$, należy podstawić najwcześniejszy moment zgłoszenia zadania po chwili t .

Przy konstrukcji algorytmu dla systemów z jednym serwerem wykorzystywaliśmy tak zwaną własność *braku pamięci* rozkładu wykładniczego. Własność można sformułować następująco: dla:

$$\mathbb{P}(X > s + t | X > s) = \mathbb{P}(X > t), \quad s, t \geq 0.$$

Łatwo pokazać, że własność jest prawdziwa dla zmiennej losowej o rozkładzie wykładniczym $X \sim \text{Exp}(\lambda)$. Okazuje się również, że rozkład wykładniczy jest jedynym rozkładem o takiej własności.

Będzie też przydatny następujący lemat.

Lemat 5.3 *Jeśli $X_1, \dots, X_n$ są niezależnymi zmiennymi losowymi oraz $X_i \sim \text{Exp}(\lambda_i)$, to zmienne*

$$T = \min_{i=1, \dots, n} X_i, \quad I = \arg \min_{i=1, \dots, n} X_i$$

są niezależne oraz $T \sim \text{Exp}(\lambda_1 + \dots + \lambda_n)$ i $\mathbb{P}(J = j) = \lambda_j / (\lambda_1 + \dots + \lambda_n)$.

Dowód Mamy

$$\begin{aligned} \mathbb{P}(T > t, J = j) &= \mathbb{P}\left(\min_{i=1, \dots, n} X_i > t, \min_{i \neq j} X_i > X_j\right) \\ &= \int_0^\infty \mathbb{P}\left(\min_{i=1, \dots, n} X_i > t, \min_{i \neq j} X_i > x\right) \lambda_j e^{-\lambda_j x} dx \\ &= \int_t^\infty \mathbb{P}\left(\min_{i \neq j} X_i > x\right) \lambda_j e^{-\lambda_j x} dx \\ &= \int_t^\infty e^{-\sum_{i \neq j} \lambda_i x} \lambda_j e^{-\lambda_j x} dx \\ &= e^{-\sum_i \lambda_i t} \frac{\lambda_j}{\sum_i \lambda_i} \end{aligned}$$

6 Procesy kolejkowe

Zajmiemy się symulacją procesów które występują na przykład w zastosowaniach telekomunikacyjnych i przemysłowych. W tym celu będzie nam pomocny język i formalizm tzw. teorii kolejek. Teoria kolejek jest działem procesów stochastycznych. Zajmuje się analizą systemów, do których wchodzi/wpływają zadania/pakiety/itd mające być obsługiwane, i które tworzą kolejkę/wypełniają bufor, i które doznają zwłoki w obsłudze. Zajmiemy się teraz przeglądem podstawowych modeli i ich rozwiązań.

6.1 Standardowe pojęcia i systemy

Zadania zgłaszają się w momentach $A_1 < A_2 < \dots$ i są rozmiarów $S_1, S_2, \dots$. Odstępy między zgłoszeniami oznaczamy przez $\tau_i = A_i - A_{i-1}$. Zadania są obsługiwane według *regulaminów* jak na przykład:

- zgodnie z kolejnością zgłoszeń (FCFS - First Come First Served)
- obsługa w odwrotnej kolejności zgłoszeń (LCFS - Last Come First Served)
- w losowej kolejności (SIRO),
- z podziałem czasu (PS), tj jeśli w systemie jest n zadań to serwer poświęca każdemu zadaniu $1/n$ swojej mocy.
- najdłuższe zadanie najpierw – LPTF (longest processing time first),
- najkrótsze zadanie najpierw – SPTF (shortest processing time first),

Zakładamy tutaj, że zadania są natychmiast obsługiwane jeśli tylko serwer jest pusty, w przeciwnym razie oczekuje w buforze. W przypadku większej liczby serwerów będziemy przyjmować, że są one jednakowo wydolne, to znaczy nie odgrywa roli w przypadku gdy kilka jest wolnych, który będzie obsługiwał zadanie. Bufor może być nieskończony lub z ograniczoną liczbą miejsc. W tym ostatnim przypadku, jeśli bufor w momencie zgłoszenia do systemu jest pełny to zadanie jest stracone. Można badać:

- $L(t)$ – liczba zadań w systemie w chwili t ,
- $L_q(t)$ – liczba zadań oczekujących w buforze,

- W_n – czas czekania n -tego zadania,
- D_n – czas odpowiedzi dla i -tego zadania (czas od momentu zgłoszenia do momentu wyjścia zadania z systemu)
- $V(t)$ – *załadowanie* (workload) w chwili t , co jest sumą wszystkich zadań które zostały do wykonania (wliczając to co zostało do wykonania dla już obsługiwanych).

W przypadku systemów ze stratami, bada się prawdopodobieństwo, że nadchodzące zadanie jest stracone (blocking probability lub loss probability).

Przez $\bar{l}$, $\bar{w}$, $\bar{d}$, $\bar{v}$ będziemy oznaczać odpowiednio w warunkach stabilności: średnią liczbę zadań w systemie, średni czas czekania, średni czas odpowiedzi, średnie załadowanie.

7 Przegląd systemów kolejkowych

7.1 Systemy z nieskończoną liczbą serwerów

System w którym każde zgłoszenie ma przydzielony natychmiast serwer do obsługi. A więc nie ma czekania. Nieciekawym jest również czas odpowiedzi wynoszący tyle ile rozmiar zadania. Będziemy więc pytać się o liczbę $L(t)$ zadań w chwili t . Zadania zgłaszają się w momentach $0 \leq A_1 < A_2 < \dots$ i ich rozmiary wynoszą odpowiednio $S_1, S_2, \dots$. A więc i -te zadanie jest w systemie w odcinku $[A_i, A_i + S_i)$. Możemy formalnie zdefiniować

$$L(t) = \sum_{j=1}^{\infty} \mathbf{1}_{[A_i, A_i + S_i)}(t).$$

Jeśli interesujemy się procesem $L(t)$ jedynie w odcinku $(0, t_{\max}]$ to wtedy musimy znaleźć n takie że $A_n \leq t, A_{n+1} > t$; wtedy i -te zadanie jest w systemie w odcinku $[A_i, \min(t_{\max}, A_i + S_i))$. Algorytm na symulację $L(t)$ gdy dane są procedury generowania (A_i) oraz (S_i) jest następujący. Piszemy $A_i = A(i)$ oraz $S_i = S(i)$.

```
l=0;
A=-\log rand/lambda;
while A<=tmax
generate S;
```

```

l=l+min(tmax,A+S)-A;
generate A;
end
lbar=l/tmax

```

M/M/∞ Najłatwiejszym do pełnej analizy jest system gdy odstępy między zgłoszeniami (τ_i) są niezależnymi zmiennymi losowymi o jednakowym rozkładzie wykładniczym z parametrem λ . Wtedy zgłoszenia są zgodne z procesem Poissona z intensywnością λ . Czasy obsługi są niezależne o jednakowym rozkładzie μ . Wtedy mamy, niezależnie od wielkości $\rho = \lambda/\mu$

- $\lim_{t \rightarrow \infty} \mathbb{P}(L(t) = k) = \frac{\rho^k}{k!} e^{-\rho}$ oraz
- $\lim_{t \rightarrow \infty} \mathbb{E} L(t) = \rho$.

Bardziej interesujące z punkty widzenia symulacji jest istnienie granic z prawdopodobieństwem 1

- $$\lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t \mathbf{1}(L(t) = k) = \frac{\rho^k}{k!} e^{-\rho} \quad (7.5)$$

oraz

- $$\lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t L(t) = \rho \quad (7.6)$$

.

M/G/∞ Proces zgłoszeń zadań jest taki sam jak w M/M/∞, jednakże rozmiary kolejnych zadań $(S_j)_j$ są niezależne o jednakowym rozkładzie G . Musimy założyć, że $\mathbb{E} S_1 = \mu^{-1} < \infty$. W tym systemie jest nieograniczony zasób serwerów a więc żadne zadanie nie czeka i nie ma sensu mówić o czasie czekania który jest zerowy i czasie odpowiedzi, który równy jest rozmiarowi zadania. Wtedy mamy, niezależnie od wielkości $\rho = \lambda \mathbb{E} S < \infty$

- $\lim_{t \rightarrow \infty} \mathbb{P}(L(t) = k) = \frac{\rho^k}{k!} e^{-\rho}$ oraz

- $$\lim_{t \rightarrow \infty} \mathbb{E} L(t) = \rho \quad (7.7)$$

,

i również są prawdziwe wzory ([?? eq:erodicM/M/infty.1??]), (7.6).

Proces $L(t)$ będzie stacjonarny jeśli jako $L(0)$ weźmiemy zmienną losową o rozkładzie Poissona $\text{Poi}(\rho)$. Wtedy funkcja kowariancji ¹

$$R(t) = \mathbb{E}[(L(s) - \rho)(L(s+t) - \rho)] \quad (7.8)$$

$$= \lambda \int_t^\infty (1 - G(v)) dv. \quad (7.9)$$

Ponadto znamy wzór na obciążenie jeśli w chwili $t = 0$ nie ma żadnych zadań, tj. $L(0) = 0$:

$$\rho - \mathbb{E} L(t) = \lambda \int_t^\infty (1 - G(s)) ds.$$

W szczególności dla $M/D/\infty$, tj. gdy rozmiary zadań są deterministyczne mamy $R(t) = \rho(1 - \mu t)$ dla $t \leq \mu^{-1}$ oraz $R(t) = 0$ dla $t > \mu^{-1}$. ²

7.2 Systemy ze skończoną liczbą serwerów

M/M/1 Najłatwiejszym do pełnej analizy jest system gdy odstępy między zgłoszeniami (τ_i) są niezależnymi zmiennymi losowymi o jednakowym rozkładzie wykładniczym z parametrem λ . To założenie odpowiada, że zgłoszenia są zgodne z procesem Poissona z intensywnością λ . Czasy obsługi są niezależne o jednakowym rozkładzie μ . Wtedy, między innymi, wiadomo że w przypadku $\lambda < \mu$ (lub równoważnie $\rho < 1$, gdzie $\rho = \lambda/\mu$), niezależnie od wyżej wymienionej dyscypliny obsługi:

- $p_k = \lim_{t \rightarrow \infty} \mathbb{P}(L(t) = k) = (1 - \rho)\rho^k$ ($k = 0, 1, 2, \dots$) oraz

$$\bar{l} = \lim_{t \rightarrow \infty} \mathbb{E} L(t) = \frac{\lambda}{\mu - \lambda} \quad (7.10)$$

,

- $\bar{l}_q = \lim_{t \rightarrow \infty} \mathbb{E} L_q(t) = \frac{\lambda^2}{\mu(\mu - \lambda)}$.

Natomiast w przypadku dyscypliny FCFS mamy

- $\bar{w} = \lim_{n \rightarrow \infty} \mathbb{E} W_n = \frac{\lambda}{\mu(\mu - \lambda)},$

¹Cytowanie za Whittom [46]

²Napisac procedure generowania relalizacji M/G/infty.

- $\bar{d} = \lim_{n \rightarrow \infty} \mathbb{E} D_n = \frac{1}{\mu - \lambda}.$

Ponadto mamy z prawdopodobieństwem 1 zbieżność

- $\lim_{t \rightarrow \infty} \int_0^t L(s) ds / t = \frac{\lambda}{\mu - \lambda}$
- $\lim_{t \rightarrow \infty} \int_0^t L_q(s) ds / t = \frac{\lambda^2}{\mu(\mu - \lambda)}$

a w przypadku dyscypliny FCFS mamy

- $\lim_{k \rightarrow \infty} \sum_{j=1}^k W_j / k = \frac{1\lambda}{\mu(\mu - \lambda)}$
- $\lim_{k \rightarrow \infty} \sum_{j=1}^k d_j / k = \frac{1}{\mu - \lambda}$

M/M/c Proces zgłoszeń zadań oraz ich rozmiary są takie same jak w M/M/1. Natomiast w tym systemie jest c serwerów obsługujących z tą samą prędkością. Podobnie jak dla M/M/1 możemy pokazać, że, jeśli obsługa jest zgodna z kolejnością zgłoszeń, to, przy założeniu $\rho < c$

$$\bar{w} = \lim_{n \rightarrow \infty} \mathbb{E} W_n = \frac{p_W}{\mu(c - \rho)}$$

i mamy również z prawdopodobieństwem 1 zbieżność

$$\lim_{k \rightarrow \infty} \sum_{j=1}^k W_j / k = \bar{w}.$$

Mamy również dla dyscypliny FCFS ale i również pewnej klasy dyscyplin obsługi

$$p_k = \lim_{t \rightarrow \infty} \mathbb{P}(L(t) = k) = \begin{cases} \frac{\rho^k}{k!} p_0, & k = 0, \dots, c \\ \frac{\rho^k}{c! c^{k-c}} p_0, & k > c. \end{cases}$$

gdzie

$$p_0 = \left[\sum_{k=0}^c \frac{\rho^k}{k!} + \frac{\rho^{c+1}}{c!(c - \rho)} \right]^{-1}.$$

oraz z prawdopodobieństwem 1

$$\lim_{t \rightarrow \infty} \int_0^t L(s) ds / t = .$$

GI/G/1 Jest to system taki, że $(\tau_k)_k$ ($A_1 = \tau_1, A_2 = \tau_1 + \tau_2, \dots$) są niezależnymi zmiennymi losowymi o jednakowym rozkładzie F a rozmiary zadań $(S_k)_k$ tworzą ciąg niezależnych zmiennych losowych o jednakowym rozkładzie G . Przypuśćmy, że w chwili $t = 0$ system jest pusty, i pierwsze zadanie przychodzi w momencie A_1 . Wtedy oczywiście czas czekania tego zadania wynosi $W_1 = 0$. Dla systemów jednokanałowych prawdziwa jest rekurencja

$$W_{k+1} = (W_k + S_k - \tau_{k+1})_+. \quad (7.11)$$

Można rozpatrywać alternatywne założenia początkowe. Na przykład, przypuścić, że w chwili $A_0 = 0$ mamy zgłoszenie zadania rozmiaru S_0 , które będzie musiało czekać przez czas W_0 . W takim przypadku rozpatruje się różne założenia o rozkładzie W_0, S_0 i A_1 , ale zawsze trzeba pamiętać niezależności ażeby poniższe fakty były prawdziwe. Natomiast rekurencja ([**eqn:rekurencja GI/G/1**]) jest niezależna od wszelkich założeń probabilistycznych. Przy założeniach jak wyżej W_k tworzy łańcuch Markowa i jeśli $\mathbb{E} S_1 < \mathbb{E} \tau_1$, to istnieją granice $\lim_{k \rightarrow \infty} \mathbb{P}(W_k \leq x)$ oraz, jeśli $\mathbb{E} S_1^2 < \infty$

$$\bar{w} = \lim_{k \rightarrow \infty} \mathbb{E} W_k .$$

Ponadto z prawdopodobieństwem 1

$$\bar{w} = \lim_{k \rightarrow \infty} \sum_{j=1}^k W_j / k .$$

W przypadku, gdy τ_1 ma rozkład wykładniczy z $\mathbb{E} \tau_1 = 1/\lambda$ to

$$\bar{w} = \frac{\lambda \mathbb{E} S_1^2}{2(1 - \lambda/\mathbb{E} S_1)} . \quad (7.12)$$

Wzór (7.12) jest zwany wzorem Pollaczka-Chinczyna. Jeśli S ma rozkład wykładniczy z parametrem μ , to wtedy powyższy wzór pokrywa się z odpowiednim wzorem dla M/M/1.

W szczególności jeśli rozmiary zadań są niezależne o jednakowym rozkładzie wykładniczym, to oznaczamy taki system przez GI/M/1. Jeśli proces zgłoszeń zadań jest procesem Poissona, to wtedy piszemy M/G/1. Jeśli zarówno proces wejściowy jest Poissonowski jak i rozmiary zadań są wykładnicze, to piszemy M/M/1.

GI/G/c Jest to system taki, że odstępy między zgłoszeniami są $(\tau_k)_k$ ($A_1 = \tau_1, A_2 = \tau_1 + \tau_2, \dots$) a rozmiary zadań $(S_k)_k$. Jest c serwerów, i zadania oczekujące w buforze do obsługi zgodnie z kolejnością zgłoszeń. Zamiast rekurencji ([?eqn:rekurencja GI/G/1?]) mamy następującą. Oznaczmy $V_n = (V_{n1}, V_{n2}, \dots, V_{nc})$, gdzie $0 \leq V_{n1} \leq V_{n2} \leq \dots \leq V_{nc}$. Składowe tego wektora reprezentują dla n -go zadania (uporządkowane) załadowanie każdego z serwerów. A więc przychodząc idzie do najmniej załadowanego z załadowaniem V_{n1} i wobec tego jego załadowanie tuż po momencie zgłoszenia wyniesie $V_{n1} + S_n$. Po czasie τ_{n+1} (moment zgłoszenia $n + 1$ -szego) załadowania wynoszą $V_{n1} + S_n - \tau_{n+1}, V_{n2} - \tau_{n+1}, \dots, V_{nc} - \tau_{n+1}$. Oczywiście jeśli są ujemne to kładziemy 0, tj $(V_{n1} + S_n - \tau_{n+1})_+, (V_{n2} - \tau_{n+1})_+, \dots, (V_{nc} - \tau_{n+1})_+$. Teraz musimy wyniki uporządkować. Wprowadzamy więc operator $\mathcal{R}(\mathbf{x})$, który wektor $\mathbf{x}$ z $\mathbb{R}^c$ przedstawia w formie uporządkowanej tj. $x_{(1)} \leq \dots \leq x_{(c)}$. A więc dla systemu z c serwerami mamy rekurencję

$$\begin{aligned} & (V_{n+1,1}, V_{n+1,2}, \dots, V_{n+1,c}) \\ &= \mathcal{R}((V_{n1} + S_n - \tau_{n+1})_+, (V_{n2} - \tau_{n+1})_+, \dots, (V_{nc} - \tau_{n+1})_+). \end{aligned}$$

Przypuśćmy, że w chwili $t = 0$ system jest pusty, i pierwsze zadanie przychodzi w momencie A_1 . Wtedy oczywiście czas czekania tego zadania wynosi $V_1 = V_2 = \dots = V_c = 0$. Rekurencja powyższa jest prawdziwa bez założeń probabilistycznych. Jeśli teraz założymy, że ciąg (τ_n) i (S_n) są niezależne i odpowiednio o jednakowym rozkładzie F i G , oraz jeśli $\rho = \frac{ES}{ET} < c$, to system się stabilizuje i istnieje rozkład stacjonarny.

7.3 Systemy ze stratami

M/M/c ze stratami Proces zgłoszeń zadań oraz ich rozmiary są takie same jak w M/M/1. Natomiast w tym systemie jest c serwerów obsługujących z tą samą prędkością, ale nie ma bufora. Jeśli przychodzące zadanie zastają wszystkie serwery zajęte, to jest stracone. Można pokazać istnienie granicy

$$p_k = \lim_{t \rightarrow \infty} \mathbb{P}(L(t) = k) = \frac{\rho^k / k!}{\sum_{j=0}^c \rho^j / j!}, \quad k = 0, \dots, c$$

gdzie $\rho = \lambda / \mu$, i teraz ta granica istnieje zawsze niezależnie od wielkości ρ . W szczególności mamy też

$$\lim_{t \rightarrow \infty} \frac{\int_0^t \mathbf{1}(L(s) = c) ds}{t} = p_c.$$

Dla tego systemu szczególnie ciekawą charakterystyką jest tzw. prawdopodobieństwo straty, które możemy w intuicyjny sposób zdefiniować następująco. Niech

$$J_k = \begin{cases} 1 & k\text{-te zadanie jest stracone} \\ 0 & \text{w przeciwnym razie.} \end{cases} \quad (7.13)$$

Wtedy z prawdopodobieństwem 1 istnieje granica

$$p_{\text{block}} = \lim_{k \rightarrow \infty} \sum_{j=1}^k J_j / k. \quad (7.14)$$

Prawdopodobieństwo p_{block} jest stacjonarnym prawdopodobieństwem, że przychodzące zadanie jest stracone. Odróżnijmy p_{block} od p_c , które jest stacjonarnym prawdopodobieństwem, że system jest pełny, chociaż dla M/M/c ze stratami te prawdopodobieństwa się pokrywają. Tak nie musi być, jeśli zadania zgłaszają się do systemu nie zgodnie z procesem Poissona. Okazuje się, że tzw. wzór Erlanga

$$p_{\text{block}} = \frac{\rho^c / c!}{\sum_{j=0}^c \rho^j / j!}$$

jest prawdziwy dla systemów M/G/c ze stratami, gdzie rozmiary kolejnych zadań tworzą ciąg niezależnych zmiennych losowych z jednakową dystrybucją G , gdzie $\mu^{-1} = \int_0^\infty x dG(x)$ jest wartością oczekiwaną rozmiaru zadania.

8 Błądzenie przypadkowe i technika zamiany miary

8.1 Błądzenie przypadkowe

Zacznijmy od pojęcia *błądzenia przypadkowego*. Niech $\xi_1, \xi_2, \dots$ będzie ciągiem niezależnych zmiennych losowych o jednakowym rozkładzie. *Błądzeniem przypadkowym* nazywamy ciąg $\{S_n, n = 0, 1, \dots\}$ zdefiniowany następująco:

$$S_0 = 0, \quad S_n = \xi_1 + \dots + \xi_n, \quad n = 1, 2, \dots$$

Będziemy zakładać, że istnieje i jest skończony pierwszy moment $\mathbb{E}\xi_1$. Podamy teraz tzw. twierdzenie Dowód pierwszych dwóch przypadków jest

prostym zastosowaniem prawa wielkich liczb Kołmogorowa. Natomiast dowód trzeciego przypadku jest nietrywialny i można go na przykład znaleźć w książce [39].

Twierdzenie 8.1 *Z prawdopodobieństwem 1 mamy*

$$\lim_{n \rightarrow \infty} S_n = \infty, \quad \text{gdy } \mathbb{E} \xi_1 > 0,$$

$$\lim_{n \rightarrow \infty} S_n = -\infty, \quad \text{gdy } \mathbb{E} \xi_1 < 0,$$

oraz

$$\limsup_{n \rightarrow \infty} S_n = \infty, \quad \liminf_{n \rightarrow \infty} S_n = -\infty \quad \text{gdy } \mathbb{E} \xi_1 = 0.$$

Teraz niech $a < 0 < b$ i rozpatrzmy błędzenie $\{S_n\}$ do momentu opuszczenia przez niego przedziału (a, b) . Błędzenie może opuścić przedział (a, b) albo przeskakując przez b lub przeskakując przez a . Interesujące jest pytanie o prawdopodobieństwo opuszczenia na przykład przez b . Formalnie definiujemy to prawdopodobieństwo następująco. Niech $\tau = \min\{i \geq 1 : S_i \notin (a, b)\}$, będzie momentem opuszczenia przedziału. Szukane zdarzenie to

$$A = \{S_\tau \geq b, \tau < \infty\}.$$

Oczywiście jeśli a, b są skończone, to z twierdzenia 8.1 mamy, że $\mathbb{P}(\tau < \infty) = 1$ ale będzie nas interesował również przypadek, gdy $a = -\infty$. Wtedy oczywiście, również z twierdzenia 8.1, mamy

$$\mathbb{P}(\tau < \infty) = 1 \quad \text{jeśli } \mathbb{E} \xi_1 \geq 0,$$

$$\mathbb{P}(\tau < \infty) < 1 \quad \text{jeśli } \mathbb{E} \xi_1 < 0.$$

Dla zadania z $a = -\infty$, dalej zwanego zadaniem 1-szym będziemy liczyć $\mathbb{P}(\tau < \infty)$, natomiast dla zadania z a, b skończonymi, będziemy liczyć zadanie 2-go które ma na celu obliczyć $\mathbb{P}(S_\tau \geq b)$.

Zauważmy jeszcze, gdy $\mathbb{P}(\xi_1 = 1) = p$ i $\mathbb{P}(\xi_1 = -1) = q = 1 - p$, to błędzenie przypadkowe generowane przez ciąg $\xi_1, \dots$ nazywamy *prostym symetrycznym*.

8.2 Technika wykładniczej zamiany miary

Kontynuując rozważania z podrozdziału X.2.1 sformułujemy teraz ważną tożsamość, zwaną tożsamością Walda, prawdziwą dla zmiennych ν będących czasem zatrzymania dla $\{S_n\}$. Niech A będzie zdarzeniem takim, że

$A \cap \{\nu = k\} = \{(\xi_1, \dots, \xi_k) \in B^k\}$ dla $B^k \subset \mathbb{R}^k$, $k = 0, 1, \dots$. Oznaczamy to przez $A \in \mathcal{F}_\nu$. W dalszej części wykładu będziemy korzystali z $\nu = \tau$, który jest czasem wyjścia błędzenia przypadkowego z (a, b) , oraz $A = S_\nu \geq b$ lub $A = \Omega$. Przedstawimy teraz ważną tożsamość, która jest prawdziwa dla dowolnego rozkładu ξ , byleby $\theta \in (\theta_1, \theta_2)$.

Fakt 8.2 (Tożsamość Walda analizy sekwencyjnej) *Jeśli $A \subset \{\tau < \infty\}$, to*

$$\mathbb{P}(A) = \tilde{\mathbb{E}}^\theta[\hat{m}^\nu(\theta)e^{-\theta(\xi_1+\dots+\xi_\nu)}; A]. \quad (8.15)$$

Wniosek 8.3

$$\mathbb{P}(\tau < \infty) = \tilde{\mathbb{E}}^\theta[\hat{m}^\tau(\theta)e^{-\theta(\xi_1+\dots+\xi_\tau)}; \tau < \infty]. \quad (8.16)$$

Przypomnijmy, że $\mathbb{E} \xi_1 < 0$. Zbadajmy teraz funkcję $\hat{m}(\theta)$ w przedziale $\theta_1 < \theta < \theta_2$. Ponieważ

$$\frac{d\hat{m}(\theta)}{d\theta} \Big|_{\theta=0} = \mathbb{E} \xi_1 < 0,$$

więc funkcja $\hat{m}(\theta)$ w $\theta = 0$ równa się 1 i w tym punkcie maleje. Dalej będziemy przyjmować następujące założenie.

Założenie $\mathbb{E} \xi_1 < 0$ i istnieje $\gamma > 0$ dla której $\hat{m}(\gamma) = 1$, oraz

$$\frac{d\hat{m}(\theta)}{d\theta} \Big|_{\theta=\gamma} < \infty.$$

Przy tym założeniu funkcja $\hat{m}(\theta)$ ma przebieg następujący. W przedziale $[0, \gamma]$ funkcja ściśle maleje od 0 do θ_0 oraz ściśle rośnie od θ_0 do $\gamma + \epsilon$ dla pewnego $\epsilon > 0$. W $\theta = 0, \gamma$ przyjmuje wartość 1.

Lemat 8.4 *Dla zmiennej losowej ξ_1 określonej na przestrzeni probabilistycznej $(\Omega, \mathcal{F}, \tilde{\mathbb{P}}^\theta)$ funkcja tworząca momenty*

$$\tilde{\mathbb{E}}^\theta e^{s\xi_1} = \hat{m}(s + \theta).$$

Stąd

$$\tilde{\mathbb{E}}^\theta \xi_1 = \frac{d\hat{m}(\theta + s)}{ds} \Big|_{s=0} > 0, \quad \text{dla } \theta > \theta_0.$$

Dowód

Wniosek 8.5 Dla $\theta > \theta_0$ mamy $\tilde{\mathbb{P}}^\theta(\tau < \infty) = 1$. Stąd dla zadania Y

$$\mathbb{P}(\tau < \infty) = \mathbb{E}^\theta[\hat{m}^\tau(\theta)\tilde{e}^{-\theta(\xi_1+\dots+\xi_\tau)}].$$

W szczególności

$$\mathbb{P}(\tau < \infty) = \tilde{\mathbb{E}}^\gamma e^{-\gamma(\xi_1+\dots+\xi_\tau)}.$$

Natomiast dla zadania 2-go moment zatrzymania τ jest zawsze skończony i dlatego obliczenie $\mathbb{P}(\tau < \infty)$ jest nieciekawe bo wielkość ta równa się 1. Natomiast potrzebne są inne wielkości.

Wniosek 8.6

$$\mathbb{P}(S_\tau \geq b) = \tilde{\mathbb{E}}^\theta[\hat{m}^\nu(\theta)e^{-\theta(\xi_1+\dots+\xi_\tau)}; S_\tau \geq b].$$

W szczególności

$$\mathbb{P}(S_\tau \geq b) = \tilde{\mathbb{E}}^\gamma[e^{-\gamma(\xi_1+\dots+\xi_\tau)}; S_\tau \geq b].$$

Jak się okaże, wybór $\theta = \gamma$ jest w pewnym sensie optymalny.

Rozdział XII

Projekty studenckie

W tym rozdziale zebraliśmy kilka ciekawych projektów wykonanych przez studentów. Opis projektu 1 ze strony domowej dr P. Lorka:

`www.math.uni.wroc.pl/~lorek/teaching/pliki/2017\_symulacje\_projekt1\_ver2.pdf`

(z dnia 1.02.2018):

1 Opis projektu

Opis pojęć pojawiających się poniżej podany jest w następnych rozdziałach. Generator liczb pseudolosowych będzie oznaczany przez PRNG. Dużo rzeczy, które są tutaj opisane występuje (być może trochę inaczej sformułowane) w skrypcie prof. Rolskiego:

<http://www.math.uni.wroc.pl/~rolski/Zajecia/sym.pdf>

- Będziemy rozpatrywać wyniki działań następujących PRNG (większość do samodzielnego zaimplementowania, opisane w Rozdziale 4) Jeśli ziarno nie zostało wyspecyfikowanie - wybór należy do Ciebie.

- LCG(13, 1, 5)
- LCG(2^{10} , 3, 7)
- GLCG(2^{10} , {3, 7, 68})
- “state” (Matlabowski)
- “twister” (Matlabowski)
 - a) z ziarnem $u_0 = 0$
 - b) z ziarnem $u_0 = 1812433253$
- Excellowski:

$$u_i = (0.9821u_{i-1} + 0.211327) \bmod 1$$

- RC4(32), rozpatrz dwie wersje:
 - i) Zainicjalizuj jednym kluczem (np. jakąś funkcją obecnego czasu itp) i następnie badaj wynik PRGA, który powinien być tak długi jak potrzebujesz.
 - ii) Skonkatenuj krótsze (ustal długość) wyniki PRGA dla kluczy ¹ $K = 0, 1, 2, \dots$

Dodatkowo będziemy rozpatrywać binarne rozwinięcia liczb $\sqrt{2}$, e , π dostępne pod adresem:

π : <http://www.math.uni.wroc.pl/~rolski/Zajecia/data.pi>

e : <http://www.math.uni.wroc.pl/~rolski/Zajecia/data.e>

$\sqrt{2}$: <http://www.math.uni.wroc.pl/~rolski/Zajecia/data.sqrt2>

¹związane jest to z tzw. “key-related attacks”, zob. https://en.wikipedia.org/wiki/Related-key_attack

• ZADANIE.

- 1) Dla RC(32) i minimum 3 innych generatorów (w tym min. jednego, który nie jest opisany w tym dokumencie) wykonaj wybrane testy statystyczne. Rozważ używanie testu χ^2 (licząc statystykę $\hat{\chi}^2$), testu Kołmogorowa-Smirnova (licząc statystykę $\hat{D}_n$), testu serii, testu odstępów dni urodzin, testu kolizji. Generatory porównuj poprzez porównanie stosownych p -wartości.
- 2) Dla liczb $\pi, e, \sqrt{2}$ wykonaj *Frequency monobit test*
 - a) dla całych ciągów (wyliczając jedną p -wartość)
 - b) każdy ciąg podziel na m części (na każdej zapuszczając test, otrzymując łącznie m p -wartości). Zobacz opis w Rozdziale 3.2 (możesz przyjąć $\alpha = 0.05$).
 - c) Zamiast powyższego punktu b) możesz wykonać testy dotyczące “Uniform distribution of p -values” - zob. Rozdział 4.2.2 w “Nist Test Suite”, link poniżej.

W powyższym punkcie 1) wykorzystaj min 3 testy, w tym także min. 1 test, który nie jest opisany w tym dokumencie. Rozważ jeden z testów dostępnych w pakiecie DieHarder, opisane są na stronie:

<https://sites.google.com/site/astudyofentropy/background-information/the-tests/dieharder-test-descriptions>

albo występują w Nist Test Suite, a opisane są w dokumencie

<http://www.math.uni.wroc.pl/~lorek/teaching/files/nistspecialpublication800-22r1a.pdf>

Całość opisz w raporcie w formacie .pdf. Raport powinien zawierać wyjaśnienie wszystkich użytych rzeczy (tak, aby ktoś niechodzący na wykład mógł zrozumieć całość) oraz wnioski.

2 O dwóch statystykach

Opiszemy krótko dwa testy, χ^2 oraz test Kołmogorowa-Smirnova. Generator liczb pseudolosowych będzie w skrócie oznaczany przez PRNG.

2.1 Test χ^2

Założmy, że mamy n obserwacji (funkcja ciągu generowanego przez PRNG), każda jest z jednej z k możliwych kategorii. Niech Y_s oznacza liczbę obserwacji z kategorii s , oraz niech p_s będzie prawdopodobieństwem “wpadnięcia” do kategorii s . Dla dużych n oczekujemy, iż

$$Y_s \approx np_s.$$

Przy założeniu, że obserwacje są niezależne i każda rzeczywiście wpada do kategorii s z prawd. p_s statystyka

$$\hat{\chi}^2 = \sum_{i=1}^k \frac{(Y_i - np_i)^2}{np_i}$$

ma rozkład χ^2 z $k - 1$ stopniami swobody, co będziemy oznaczać przez $\chi^2(k - 1)$.

2.2 Test Kołmogorowa-Smirnova

Założmy, że zmienna losowa X ma rozkład ciągły o dystrybuancie $F_X(x)$. Założmy także, że mamy n obserwacji $X_1, \dots, X_n$ i chcemy testować czy są one niezależne i pochodzą z rozkładu $F_X(x)$. Zdefiniujmy *dystrybuantę empiryczną* $F_n(x)$ jako

$$\hat{F}_n(x) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(X_i \leq x)$$

Twierdzenie Gliwienki-Cantellego mówi, że jeśli $X_1, \dots, X_n$ jest próbką z rozkładu F_X , to

$$\sup_{x \in \mathbb{R}} |\hat{F}_n(x) - F(x)| \rightarrow 0, \quad n \rightarrow \infty,$$

z prawdopodobieństwem 1. Co więcej, statystyka

$$\hat{D}_n = \sqrt{n} \sup_{x \in \mathbb{R}} |\hat{F}_n(x) - F(x)|$$

dąży, gdy $n \rightarrow \infty$, do znanego rozkładu:

$$Pr(\hat{D}_n \leq t) \rightarrow H(t) = 1 - \sum_{i=1}^{\infty} (-1)^{i-1} e^{-2i^2 t},$$

H jest dystrybuantą rozkładu Kolmogorowa-Smirnova.

Dla rozkładu dyskretnego, sposób wyliczania D_n podany jest w [3].

3 Testowanie generatorów liczb pseudolosowych poprzez testy empiryczne

Istnieje wiele testów dla PRNG. Można je podzielić na dwie kategorie: **testy empiryczne** oraz **testy teoretyczne**. Testy teoretyczne wymagają wiedzy o działaniu samego generatora, ale sam ciąg nie musi de facto być generowany, dobre są zazwyczaj trudne do znalezienia. Z kolei testy empiryczne są wykonywane na ciągach otrzymanych z PRNG, żadna wiedza o tym jak zostały wygenerowane nie jest potrzebna.

Testy wymagają liczb o rozkładzie na odcinku $(0, 1)$ albo o rozkładzie dyskretnym na $\{0, 1, \dots, M\}$. W zależności od generatora, jego wynikiem może również być albo ciąg liczb z przedziału $(0, 1)$ albo liczby dyskretne. Niech $U_1, U_2, \dots$ oznaczają liczby z przedziału $(0, 1)$, wtedy możemy je przekształcić na liczby dyskretne ze zbioru $\{0, 1, \dots, M\}$ poprzez

$$Y_i = \lfloor MU_i \rfloor.$$

Podobnie - dla liczb dyskretnych Y_i z przedziału $\{0, 1, \dots, M\}$ możemy otrzymać liczby z $(0, 1)$ poprzez

$$U_i = \frac{Y_i}{M}.$$

3.1 p -wartość

W kontekście $\hat{\chi}^2$: możemy testować hipotezę:

H_0 obserwacje są niezależne, każda jest z kategorii s z prawd. p_s

H_1 obserwacje są nie pochodzą z takiego rozkładu

Jakie jest prawd., że jeśli obserwacje rzeczywiście są niezależne i pochodzą z omawianego rozkładu to zaobserwowalibyśmy coś $\geq \hat{Y}$? Jest to

$$p = Pr(\chi^2(k-1) > \hat{\chi}^2)$$

(co odczytujemy z tablic). Małe p -wartości oznaczają, iż prawdopodobieństwo zaobserwowania $\hat{\chi}^2$, lub czegoś większego, jest małe. Ustalając poziom ufności na α (typowo 0.05 lub 0.01) odrzucimy hipotezę H_0 jeśli $p \leq \alpha$.

W kontekście zmiennej losowej N o rozkładzie normalnym $N(0, 1)$ p -wartość liczona jest jako (gdzie s jest zaobserwowaną statystyką) $p = Pr(|N| > s)$

3.2 Testy empiryczne

Idea testowania: dla określonych zdarzeń, których prawdopodobieństwa znamy (zakładając jednostajność) liczymy statystyki $\hat{\chi}^2$ oraz $\hat{D}_n$ (lub inne) i wyliczamy ich p -wartości. Mamy dwa podejścia:

- Dla całego ciągu wyliczmy daną p -wartość. Generatory porównujemy porównując p -wartości.
- Dzielimy ciąg na m części. Na każdej wyliczamy statystykę i stosowną p -wartość. Załóżmy np., że $m = 1000$ oraz poziom istotności wynosi $\alpha = 0.05$. Załóżmy także, iż $passed = 990$ ciągów “zdało” test, tzn. p -wartość $\geq \alpha$. Oznaczmy proporcję tych ciągów przez $\rho = passed/m = 0.99$.

Zauważmy, że jeśli liczby są losowe, to każdy test powinien zdać z prawdopodobieństwem α . Niech Z_i będą iid o rozkładzie $Pr(Z_i = 1) = 1 - Pr(Z_i = 0) = 1 - \alpha$. Mamy $EZ = 1 - \alpha$, $Var_Z = \alpha(1 - \alpha)$, a z CTG

$$S_m = \frac{1}{\sqrt{m}} \sum_{i=1}^m \frac{Z_i - (1 - \alpha)}{\sqrt{\alpha(1 - \alpha)}}$$

ma (w przybliżeniu) rozkład normalny $N(0, 1)$. Zatem,

$$Pr \left\{ \rho \in \left[1 - \alpha \pm z_{1-\frac{\beta}{2}} \sqrt{\frac{\alpha(1-\alpha)}{m}} \right] \right\} \approx 1 - \beta$$

Można przyjąć $\beta = 0.003$, wtedy $z_{1-\frac{\beta}{2}} \approx 3$ (powyższe jest nazywane *zasadą 3 sigma*). Jeśli proporcja ρ jest poza tym przedziałem, rozumiemy to jako dowód tego, że dane nie pochodzą z rozkładu jednostajnego.

3.2.1 Frequency monobit test

Założmy, że $X_1, X_2, \dots$, są niezależnymi zmiennymi losowymi o jednakowym rozkładzie $Pr(X_i = -1) = Pr(X_i = 1) = 1/2$. Zauważmy, że $EX = 0, Var X = 1$. Zdefiniujemy

$$S_n = \frac{1}{\sqrt{n}} \sum_{i=1}^n X_i$$

Z CTG mamy, iż dla dużego n rozkład S_n jest w przybliżeniu $N(0, 1)$. Dlatego, p -wartością jest

$$p = Pr(|N| > |S_n|) = 2(1 - \phi(|S_n|)),$$

gdzie N jest zmienną losową o rozkładzie $N(0, 1)$, a ϕ jego dystrybuantą.

3.2.2 Frequency Test

Jeden z podstawowych testów. Mając dany ciąg $U_1, \dots, U_n$ możemy

- zastosować test K-S z dystrybuantą $F_X(x) = x, x \in (0, 1)$.
- zastosować test χ^2 w następujący sposób: ustalmy M i wyliczmy $Y_i = \lfloor MU_i \rfloor$.
Mamy M kategorii, $p_i = \frac{1}{M}$.

3.2.3 Test serii

Podobny do *Frequency test*, ale sprawdzane jest czy występowanie par elementów. Założmy, że $U_i \in \{0, 1, \dots, M-1\}$. Podzielmy ciąg na podciągi długości $2n$:

$$(U_1, U_2), (U_3, U_4), \dots, (U_{2n-1}, U_{2n}).$$

Policz wystąpienie każdej z par $(q, r), 0 \leq q, r < M$ i zastosuj test χ^2 z $M^2 - 1$ stopniami swobody.

Test może być łatwo rozszerzony na badanie dowolnych krotek (zob. skrypt [3]).

3.2.4 Test odstępów dni urodzin

Zapiszmy dni urodzin kolejnych osób $Y_1, \dots, Y_n \in \{1, \dots, k\}$ i ustawmy je w porządku niemalejącym $Y_{(1)} \leq \dots \leq Y_{(n)}$. Zdefiniujmy odstęp

$$S_1 = Y_{(2)} - Y_{(1)}, \dots, S_{n-1} = Y_{(n)} - Y_{(n-1)}.$$

Niech K będzie liczbą równych odstępów (tzn. ile razy mamy równość pomiędzy S -ami)

- Okazuje się, że jeśli n jest duże i $\lambda = n^3/(4k)$ małe, to przy założeniu o równomiernym rozkładzie Y_i (hipoteza $\mathcal{H}_0$) zmienna K ma w przybliżeniu rozkład Poissona z parametrem λ . Jednym z wyborów jest $n = 2^{10}, k = 2^{24}$, wtedy $\lambda = 16$. Zatem, jeśli $K = y$, to p -wartość wynosi

$$Pr(K \geq y | \mathcal{H}_0) \approx 1 - \sum_{j=0}^{y-1} \frac{\lambda^j}{j!} e^{-\lambda}$$

- Knuth [1] sugeruje $k = 512 = 2^9$ oraz $n = 2^{25}$. Prawdopodobieństwa liczby równych odstępów podane są w Fig. 1. Używając tych prawdopodobieństw można zastosować test χ^2 .

s	0	1	2	≥ 3
$Pr(K = s)$	0.368801	0.369035	0.183471	0.078692

Figure 1: Prawd. dla testu odstępów dni urodzin, $k = 2^{25}, k = 2^9$

3.2.5 Test kolizji

Zob. skrypt [3].

3.2.6 Test pokerowy

Zob. skrypt [3].

4 Generatory liczb pseudolosowych

Tutaj należałoby krótko wypisać podstawowe, tak by studenci mogli zaprogramować (głównie LCG) z różnymi parametrami, by były też “kiepskie”.
Może tutaj wersja RC4 ? (krótki opis)

4.1 LCG(M, a, c)

Generatory LCG (z ang. *linear congruential generator*) zmieniają swój stan zgodnie z rekurencją

$$x_n = (ax_{n-1} + c) \bmod M$$

Stan początkowy: x_0

4.2 GLCG($M, \{a_i\}_{i=1}^k$)

Generatory te zmieniają stan zgodnie z rekurencją

$$x_n = (a_1x_{n-1} + \dots + a_kx_{n-k}) \bmod M$$

Stan początkowy: $x_0, x_1, \dots, x_{k-1}$

4.3 RC4(n)

RC4 (w opisie będziemy pomijali (n)) jest tzw. szyfrem strumieniowym², który może być wykorzystany jako PRNG. Oznaczmy $[n] := \{0, 1, \dots, n-1\}$.

Jego tzw. *stanem wewnętrznym* jest (S, i, j) , gdzie S jest permutacją $[n]$, a $i, j \in [n]$ są indeksami. Jako wejście algorytm bierze klucz: L liczb, każda z $[n]$ (możemy myśleć, iż klucz jest dodatkowym parametrem naszego PRNG). Następnie:

²http://pl.wikipedia.org/wiki/Szyfr_strumieniowy

- KSA (z ang. *key scheduling algorithm*) inicjuje S permutacją identycznościową, następnie używając klucza K zmienia S w inną permutację (w zamyśle losową)
- PRGA (z ang. *Pseudo Random Generation Algorithm*) zwraca elementy tej permutacji jednocześnie ją updateując.

KSA oraz PRGA przedstawione są na Fig. 2. W oryginalnym RC4 mamy $n = 256$, natomiast liczby zwracane przez PRGA są "XORowane" z wiadomością otrzymując kryptogram. RC4 używany w trybie PRNG po prostu zwraca wyjście PRGA.

KSA(K)	PRGA
<pre> for $i := 0$ to $n - 1$ do $S[i] := i$ end for $j := 0$ for $i := 0$ to $n - 1$ do $j := j + S[i] + K[i \bmod L]$ swap($S[i], S[j]$) end for $i, j := 0$ </pre>	<pre> while $r \in \mathcal{N}_+$ do $i := i + 1$ $j := j + S[i]$ swap($S[i], S[j]$) $Y_r \leftarrow S[S[i] + S[j]]$ end while </pre>

Figure 2: RC4: KSA i PRGA. Wszystkie dodawania są wykonywane mod n

Uwaga. RC4 z $n = 256$ był do niedawna intensywnie używany. Jeśli w KSA zamienimy linijkę $j := j + S[i] + K[i \bmod L]$ na $j := \mathbf{random}(n)$, to można o tym algorytmie myśleć jako o tasowaniu kart. W kroku i -tym zamieniamy kartę na pozycji i z kartą na losowej pozycji. Jest to tzw. tasowanie Cyclic-To-Random, o którym wiadomo, iż potrzebne jest $O(n \lg n)$ kroków, by uzyskać jednostajną permutację. KSA wykonuje ich jednak tylko n , co jest jednym z jego poważnym mankamentów.

References

- [1] D. E. Knuth. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*,. Addison-Wesley, 1997.
- [2] P. L'Ecuyer, R. Simard, TestU01: A c library for empirical testing of random number generators, *ACM Trans. Math. Sofw.* 33(4), 2007.
- [3] T. Rolski, *Symulacje stochastyczne i teoria Monte Carlo*, skrypt UW, 2017. <http://www.math.uni.wroc.pl/~rolski/Zajecia/sym.pdf>

Testowanie generatorów liczb pseudolosowych.

“Wprowadzenie do symulacji i metod Monte Carlo”, projekt nr 1.

Maciej Dulęba

Streszczenie

W projekcie wybrano kilka popularnych generatorów liczb pseudolosowych i zbadano je z wykorzystaniem testów statystycznych. Zaimplementowano algorytmy generatorów, a analiza uzyskanych wyników pozwala określić, czy wygenerowane ciągi pseudolosowe spełniają własności, które powinny cechować dane losowe. Korzystając z wybranych miar jakości porównano generatory wskazując ich zalety i wady.

1 Wprowadzenie

Generowanie liczb losowych jest ważnym zadaniem, zarówno teoretycznym jak i praktycznym. Liczby losowe są powszechnie stosowane w symulacji zjawisk z wykorzystaniem metody Monte Carlo czy kryptografii. ‘Poprawne’ generowanie liczb pseudolosowych jest zatem istotne w zastosowaniach. Większość aktualnie stosowanych metod (poza np. ręcznym rzucaniem kostkami) tworzenia długich ciągów liczb losowych jest deterministyczna. Pojawia się zatem kwestia sprawdzenia, czy dany generator pseudolosowy spełnia własności, które są prawdziwe dla rzeczywiście losowych ciągów liczb. Na podstawie takich testów można stwierdzić, czy dany generator imituje ‘losowy’, czy też wykazuje niepożądane cechy. W testowaniu generatorów liczb pseudolosowych nie ma przyjętych obiektywnych miar czy wartości numerycznych określających ich jakość. Tak więc do użytkownika należy określenie, czy dany generator spełnia jego potrzeby.

2 Oznaczenia

W projekcie są stosowane następujące oznaczenia:

- $[M] = \{0, 1, \dots, M - 1\}$,
- X, Y - ciągi liczb losowych,
- X_i, Y_i - elementy X, Y numerowane od 1,
- x_i - wygenerowane liczby pseudolosowe $i = 1, 2, \dots, n$.

3 Testowane generatory liczb pseudolosowych

O generatorze liczb pseudolosowych można myśleć jako o pewnym algorytmie/procedurze, która przy ustalonych parametrach wejściowych zwraca (potencjalnie nieskończony) ciąg liczb pseudolosowych. W niniejszym projekcie do testów wykorzystano generatory liczb pseudolosowych opisane w kolejnych sekcjach. Postawy teoretyczne zaczerpnięto z opracowania [1] i skryptu [2].

3.1 GLCG

$\text{GLCG}(M, \{a_i\}_{i=1}^k)$ jest rodziną generatorów liniowych tworzących ciągi liczb $x_i \in [M]$. Wyrazy $x_1, \dots, x_k$ są w pewien sposób inicjowane, zaś kolejne wyrazy dane są wzorem:

$$x_n = (a_1 x_{n-1} + \dots + a_k x_{n-k}) \mod M \quad \text{dla} \quad n = k+1, k+2, \dots$$

Za [1], w testach przyjęto następujące parametry

- $M = 2^{10}$, $k = 3$,
- $\{a_1, a_2, a_3\} = \{3, 7, 68\}$,
- $\{x_1, x_2, x_3\} = \{11, 54, 73\}$.

z własnymi warunkami początkowymi.

3.2 RC4(n)

Algorytm RC4(n) służy do szyfrowania strumieniowego, ale może być także wykorzystywany jako generator liczb pseudolosowych. Dla zadanego n , algorytm zwraca liczby $x_i \in [n]$. RC4 w pamięci wewnętrznej przechowuje trójkę (S, i, j) , gdzie S jest permutacją $[n]$, zaś i, j są dwoma indeksami z $[n]$. W pierwszym kroku, korzystając z zewnętrznego klucza inicjalizującego K , algorytm generuje początkowe S startując z permutacji identycznościowej (krok ten nazywa się KSA - *key scheduling algorithm*). Następnie RC4 zwraca element z S jednocześnie zmieniając S (PRGA - *pseudo-random generation algorithm*). Krok ten powtarzany jest wielokrotnie i nie zależy już od klucza K . W kryptografii n jest naturalną potęgą 2. Generowane liczby są następnie poddane operacji xor z kolejnymi bitami strumienia wiadomości. Na użytek niniejszego projektu wystarczający jest ciąg liczb powstający w kroku PRGA. Poniżej przedstawiony jest pseudokod RC4(n) [1]

Algorithm 1 RC4(n)

<pre> KSA(K) for i = 0, 1, ..., n - 1 do S[i] = i end for j = 0 for i = 0, 1, ..., n - 1 do j = (j + S[i] + K[i]) mod n swap(S[i], S[j]) end for </pre>	<pre> PRGA() i, j = 0 r = 1 while true do i = (i + 1) mod n r = r + 1 j = (j + S[i]) mod n swap(S[i], S[j]) return x_r = S[(S[i] + S[j]) mod n] end while </pre>
---	--

RC4 występuje w wielu wariantach zależnie od doboru klucza. Czasami zakłada się, że K powinno być n bitową liczbą i wtedy w KSA $K[i] \in \{0, 1\}$ oznacza kolejne jej bity. Można także przyjąć, że K to n liczb ze zbioru $[n]$. Są także warianty, gdy klucz nie ma długości n (albo n bitów). Konkatenuje się wtedy wiele kopii K , aby uzyskać klucz odpowiedniej długości. W testach użyto dwóch wersji RC4 dla $n = 32$:

- RC4 (standardowy wariant) - K jest 32-bitową liczbą otrzymaną z aktualnego czasu komputera. W zapisie dziesiętnym: $K = \overline{\text{HHMMSSSSS}}$, gdzie HH - aktualna godzina, MM - minuta, SSSSS - sekunda wyrażona z dokładnością do $[ms]$. $K[i]$ to bity K , zaczynając od najmniej znaczącego.
- RC4cat (konkatenacja wielu ciągów) - stosując standardowy wariant RC4 dla kolejnych kluczy $K = 0, 1, \dots$ (a dokładniej ich 32-bitowych zapisów binarnych) tworzą krótsze ciągi liczb pseudolosowych o długości $M = 1000$, a następnie konkatenują (do uzyskania ciągu losowego żądanej długości).

3.3 Matlabowski twister

Twister jest jednym z trzech wbudowanych generatorów liczb pseudolosowych w MATLAB-ie. Jest generatorem popularnym także w innych pakietach matematycznych. Twister jest implementacją algorytmu Mersenne Twister, który korzysta z dużej liczby pierwszej Mersenne'a $2^{19937} - 1$. Używając tego generatora wygenerowano ciąg liczb $x_i \in [M]$ dla $M = 32$ (używając matlabowej funkcji *randi*). Obliczenia przeprowadzono dla dwóch wyborów ziarna generatora (parametru początkowego) $seed = 0$ oraz $seed = 1812433253$ (generatory oznaczono dalej, odpowiednio, Twist i Twist 2).

3.4 Rule 30

Rule 30 nosi nazwę generator używany w pakiecie obliczeń symbolicznych *Mathematica*. Opiera się na symulowaniu jednowymiarowego automatu komórkowego z regułą zmiany stanu zaproponowaną przez Stephena Wolframa i nazywaną regułą 30 [3]. W danym dwustronnie nieskończonym automacie komórkowym

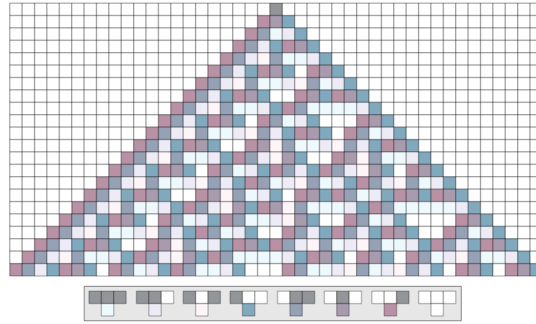
$$X_t[i], \quad i \in \mathbb{Z} \text{ (komórki)}, \quad t \in \mathbb{N} \text{ (czas)},$$

każda komórka reprezentuje jeden bit, a dyskretny parametr t oznacza czas. Wartość $X_{t+1}[i]$ zależy od wartości komórek $X_t[i-1], X_t[i], X_t[i+1]$ (nazywanych dalej wzorcem) i jest zadana przez

$$X_{t+1}[i] = \begin{cases} 0, & \text{dla wzorców } 111, 110, 101, 000, \\ 1, & \text{dla wzorców } 100, 011, 010, 001. \end{cases} \quad (1)$$

Wolfram sugeruje, aby używać wartości $X_t[0]$ dla $t \geq 1$ jako bitów losowych. Jako stan początkowy automatu wybrano

$$X_1[i] = \begin{cases} 1, & \text{dla } i = 0, \\ 0, & \text{w przeciwnym przypadku.} \end{cases} \quad (2)$$



Rysunek 1: Wyprowadzanie kolejnych stanów automatu. Ciemne kolory oznaczają 1.[3]

3.5 Nazewnictwo

W dalszej części są stosowane następujące skróty oznaczeń generatorów:

- GLCG – sekcja 3.1,
- RC4 – sekcja 3.2, inicjalizujący klucz K pewną funkcją aktualnego czasu komputera,
- RC4cat – sekcja 3.2, konkatenujący wiele ciągów dla różnych kluczy K ,
- Twister – sekcja 3.3, z ziarnem $seed = 0$,
- Twister 2 – sekcja 3.3, z ziarnem $seed = 1812433253$,
- Rule 30 – sekcja 3.4.

4 Testy statystyczne

Dla wymienionych generatorów liczb pseudolosowych przeprowadzono testy statystyczne, mające na celu sprawdzenie ich jakości. Zauważmy najpierw, że każdy z generatorów generuje liczby ze zbioru $[K]$ dla pewnego K . Niech $X = GEN(N, K)$ oznacza, że generator GEN wygenerował ciąg liczb pseudolosowych X długości N , oraz $X_i \in [K]$. Testujemy, czy X jest niezależnym ciągiem liczb jednostajnie wybieranych z $[K]$ (wtedy GEN jest losowym generatorem). Każdy z omówionych, w kolejnych sekcjach, testów zwraca jako wynik pewną wartość liczbową p określającą najczęściej prawdopodobieństwo,

że rzeczywiście zaobserwowano daną wartość (dla rozkładów dyskretnych i skończonych), albo wartość ogonów (dla rozkładów ciągłych) liczonej statystyki przy założeniu hipotezy zerowej (generator zwraca ciąg liczb z pewnego rozkładu). Małe wartości p mogą być przesłanką do odrzucenia hipotezy zerowej, czyli stwierdzenia, że generator nie jest losowy.

4.1 Test χ^2

4.1.1 Teoria

Rozważmy n obserwacji otrzymanych z generatora liczb pseudolosowych, należących do zbioru (pudełek) $[k]$. Testujemy, czy pochodzą one z pewnego rozkładu prawdopodobieństwa P . Niech zmienna losowa Y_s reprezentuje liczbę obserwacji o wartości s (wpadających do s -tego pudełka). Niech $p_s = P(s)$ będzie prawdopodobieństwem zaobserwowania elementu s dla rozkładu P . Jeśli obserwacje rzeczywiście są niezależne i pochodzą z rozkładu P to statystyka

$$\chi^2 = \sum_{i=0}^{k-1} \frac{(Y_i - np_i)^2}{np_i} \quad (3)$$

ma rozkład χ^2 z $k - 1$ stopniami swobody.

4.1.2 Implementacja

Niech $X = GEN(N, K)$. Zauważmy, że w teście χ^2 możemy przyjąć uniwersum $[K]$ z oczekiwanym $p_s = 1/K$. W GLCG mamy jednak $K \gg N$, tak więc większość pudełek będzie pusta lub jednoelementowa. W takim przypadku bierzemy $R|K$, i rozważamy ciąg Y , gdzie $Y_i = X_i \bmod R$. Jeśli X jest niezależny i ma rozkład jednostajny, to Y także ma taki rozkład (ale w $[R]$). Dla ciągu Y i uniwersum $[R]$ z $p_s = 1/R$ stosujemy test χ^2 .

4.1.3 Wyniki

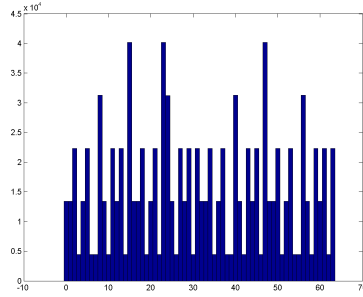
Poniżej zebrano wyniki testu χ^2 dla rozważanych generatorów. N oznacza długość wygenerowanego ciągu, K rozmiar uniwersum wartości zwracanych przez generator, a R liczbę pudełek.

Generator	N	K	R	p-wartość
Rule 30	10^4	2	2	0.522173
Twister	10^6	64	64	0.027996
Twister 2	10^6	64	64	0.686085
RC4	10^6	32	32	0.061448
RC4cat	10^6	32	32	0.571098
GLCG	10^6	2^{10}	64	0.000000

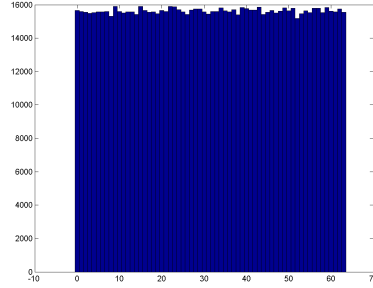
Implementacja Rule 30 generowała bity stosunkowo wolno, zatem ograniczono się, w tym przypadku, do ciągu długości 10^4 . Jak widać, dla większości generatorów otrzymane p-wartości są w miarę duże. Nie można zatem w tych przypadkach wykluczyć, że testowane generatory są dobre. Z wyników widać także, że GLCG według tego testu nie jest losowy, a dwa kolejne też mają p-wartości relatywnie małe. Na rys. 4.1.3 przedstawiono histogram pierwszych 10^6 wartości modulo 64 generowanych przez GLCG i, dla porównania, histogram dla Twistera z ziarnem 0.

4.2 Test Kołmogorowa-Smirnova

Sekcję opracowano na podstawie [1].



(a) Histogram wartości dla GLCG



(b) Histogram wartości dla Twistera

4.2.1 Teoria

Rozważmy ciągłą zmienną losową X o dystrybucji F_X . Mamy n niezależnych obserwacji z tego rozkładu. Definiujemy dystrybuantę empiryczną wzorem

$$\hat{F}_X(x) = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(X_i \leq x). \quad (4)$$

Wtedy statystyka

$$\hat{D}_n = \sqrt{n} \sup_{x \in \mathbb{R}} |\hat{F}_X(x) - F_X(x)| \quad (5)$$

dąży, dla $n \rightarrow \infty$, do rozkładu

$$P(\hat{D}_n \leq t) \rightarrow 1 - \sum_{i=1}^{\infty} (-1)^{i-1} e^{-2i^2 t} \quad (6)$$

zwanego rozkładem Kołmogorowa-Smirnova.

4.2.2 Implementacja

Niech $X = GEN(N, K)$. Będziemy chcieli korzystać z testu Kołmogorowa-Smirnova dla rozkładu $U[0, 1]$. Ustalmy $l \in \mathbb{N}$. Przekształcamy ciąg X , tak aby otrzymać ciąg Y^l elementów z $[K^l]$ (dla przypomnienia $X_i \in [K]$). Przyjmujemy $Y_j^l = \sum_{i=1}^l K^{l-i} X_{(j-1)l+i}$. Odpowiada to wzięciu po l elementów z X i potraktowaniu ich jako liczby zapisanej w systemie o podstawie K . Długość ciągu Y^l to $N' = \lfloor N/l \rfloor$. Jeśli X jest niezależny i ma rozkład jednostajny, to Y^l także ma taki rozkład. Dla $l \rightarrow \infty$ mamy $Y_j^l/K^l \xrightarrow{d} U[0, 1]$. Traktujemy zatem Y^l/K^l jako ciąg zmiennych o rozkładzie $U[0, 1]$. Liczymy dystrybuantę empiryczną $\hat{F}(x)$ dla tego ciągu. Nietrudno zauważyć, że jest ona funkcją schodkową o wartościach y_j^l/K^l . Możemy z tego skorzystać do liczenia $\hat{D}_n$. Mamy

$$\hat{D}_n = \sqrt{n} \max_{j \in \{1, \dots, N'\}} \left| \hat{F} \left(\frac{y_j^l}{K^l} \right) - \frac{y_j^l}{K^l} \right|. \quad (7)$$

Pozostaje zatem policzyć $P(\hat{D}_n > f)$ dla pewnego otrzymanego f . Ogon rozkładu $\hat{D}_n$ wyraża się nieskończonym, bardzo szybko zbieżnym, szeregiem, więc przybliżono go pierwszym tysiącem elementów.

4.2.3 Wyniki

Wyniki testu Kołmogorowa-Smirnova dla rozważanych generatorów przedstawiono w tabeli, gdzie N oznacza długość wygenerowanego ciągu, K rozmiar uniwersum wartości zwracanych przez generator, a l liczbę elementów X tworzących pojedynczy element Y . We wszystkich poniższych przykładach Y przyjmuje wartości z $[2^{30}]$.

Generator	N	K	l	p-wartość
Rule 30	10^4	2	30	0.252335
Twister	10^6	64	5	0.182904
Twister 2	10^6	64	5	0.284762
RC4	10^6	32	6	0.113426
RC4cat	10^6	32	6	0.064289
GLCG	10^6	2^{10}	3	0.000013

Wartości p dla testu Kołmogorowa-Smirnova są istotnie mniejsze niż dla testu χ^2 , choć większość jest powyżej dopuszczalnego poziomu 0.01.

4.3 Test serii Stevensa

Sekcję opracowano na podstawie dokumentacji zestawu testów NIST [4].

4.3.1 Teoria

Rozważmy binarną zmienną losową spełniającą $P(X_i = 0) = P(X_i = 1) = 1/2$ i ciąg n niezależnych zmiennych o tym rozkładzie. Definiujemy następujące zmienne:

$$R_i = \begin{cases} 1, & X_i \neq X_{i+1}, \\ 0, & \text{w przeciwnym przypadku.} \end{cases} \quad (8)$$

$$\pi_X = \frac{1}{n} \sum_{i=1}^n X_i. \quad (9)$$

oraz

$$V_n(X) = 1 + \sum_{i=1}^{n-1} R_i. \quad (10)$$

Wtedy V_n oznacza liczbę ciągów kolejnych identycznych elementów w X (bądź też liczbę zmian wartości w ciągu X), zaś π_X średnią wartość zmiennej X_i . Liczymy wartość p ze wzoru:

$$p = \begin{cases} \operatorname{erfc}\left(\frac{|V_n(X) - 2n\pi_X(1-\pi_X)|}{\sqrt{8n\pi_X(1-\pi_X)}}\right), & |\pi_X - 1/2| < \tau, \\ 0, & \text{w przeciwnym przypadku.} \end{cases} \quad (11)$$

Omawiany test występuje w zestawie NIST, z zalecaniem by $\tau = 2/\sqrt{n}$ oraz odrzucać hipotezę o niezależności i jednorodności X gdy $p < 0.01$.

4.3.2 Implementacja

Niech $X = GEN(N, K)$. Do testu serii jest wymagany ciąg binarny. Ponieważ K jest dla badanych przypadków potęgą dwójki, więc przekształcenie ciągu X w bitowy ciąg Y jest dość łatwe. Niech Z_i oznacza rozwinięcie binarne X_i (długości $\log(K)$). Określamy $Y = \{Rev(Z_1), Rev(Z_2), \dots, Rev(Z_N)\}$, gdzie $Rev(Z)$ to inwersja ciągu Z . Prz założeniu jednorodności i niezależności X , ciąg Y jest ciągiem niezależnych zmiennych binarnych o rozkładzie $P(Y_i = 0) = P(Y_i = 1) = 1/2$. Dla ciągu Y stosujemy test serii Stevensa.

4.3.3 Wyniki

W tabeli zebrano wyniki testu serii Stevensa dla rozważanych generatorów, gdzie N oznacza długość wygenerowanego ciągu, K rozmiar uniwersum wartości zwracanych przez generator.

Generator	N	K	p-wartość
Rule 30	10^4	2	0.767294
Twister	10^6	64	0.931471
Twister 2	10^6	64	0.674598
RC4	10^6	32	0.045006
RC4cat	10^6	32	0.620250
GLCG	10^6	2^{10}	0.000000

Dla większości generatorów test serii Stevensa daje stosunkowo duże wartości p . Zaskakująco niską wartość uzyskano dla generatora RC4, jedynie niewiele powyżej progu akceptacji (0.01). Generator GLCG również w tym teście nie może być uznany za losowy.

4.4 Test odstępów dni urodzin

Opracowano na podstawie skryptu [2].

4.4.1 Teoria

Rozważmy n niezależnych zmiennych X_i , jednostajnych w zbiorze $[k]$ (zbiór dni w roku oraz pewne n urodzin). Dla statystyki porządkowej $X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(n)}$ definiujemy odstęp między urodzinami

$$S_i = X_{(i+1)} - X_{(i)}. \quad (12)$$

Niech K będzie liczbą powtórzeń wśród S_i ($K = n - 1 - \text{Uniq}$, gdzie Uniq jest liczbą różnych wartości w zbiorze $\{S_1, S_2, \dots, S_{n-1}\}$). Wtedy jeśli n jest duże oraz $\lambda = n^3/(4k)$ jest małe, to K ma w przybliżeniu rozkład Poissona z parametrem λ

$$P(K = r) \approx \frac{\lambda^r}{r!} e^{-\lambda}. \quad (13)$$

Dla zaobserwowanego $K = y$ definiujemy wartość prawdopodobieństwa p wzorem

$$P(K \geq y) \approx 1 - \sum_{r=0}^{y-1} \frac{\lambda^r}{r!} e^{-\lambda}. \quad (14)$$

4.4.2 Implementacja

Niech $X = \text{GEN}(N, K)$. Rozważamy rok z $k = 2^{30}$ dniami oraz $n = 2^{11}$ urodzin, wtedy $\lambda = 2$. Niech $l \in \mathbb{N}$, takie, że $K^l = k$ (dla wszystkich rozważanych generatorów takie l można dobrać). Przekształcamy ciąg X , tak aby otrzymać ciąg Y^l elementów z $[k]$. Podobnie jak w teście Kołmogorowa-Smirnova przyjmujemy $Y_j^l = \sum_{i=1}^l K^{l-i} X_{(j-1)l+i}$. Odpowiada to wzięciu po l elementów z X i potraktowaniu ich jako liczby zapisanej w systemie o podstawie K . Długość ciągu Y^l to $N' = \lfloor N/l \rfloor$. Jeśli X jest niezależny i ma rozkład jednostajny, wtedy Y^l także ma taki rozkład. Aby skorzystać z testu odstępów dni urodzin dzielimy Y^l na fragmenty długości n (ostatni, niepełny fragment pomijamy). Dla każdego z tych fragmentów liczymy statystykę K . Otrzymujemy w ten sposób ciąg $\{K_1, \dots, K_w\}$, gdzie $w = \lfloor N'/n \rfloor$. Dla tego ciągu stosujemy test χ^2 , z pudełkami postaci $\{K=0\}, \{K=1\}, \{K=2\}, \{K=3\}, \{K \geq 4\}$. Teoretyczne prawdopodobieństwa przyjmowania tych wartości są przedstawione w tabeli (rozkład Poissona ze współczynnikiem 2).

s	0	1	2	3	≥ 4
$P(K = s)$	0.1353	0.2707	0.2707	0.1804	0.1429

Dla każdego K_i sprawdzono do której z powyższych klas należy i dla otrzymanych obserwacji zastosowano test χ^2 .

4.4.3 Wyniki

Poniższa tabela zawiera wyniki testu odstepu urodzin dla rozważanych generatorów. N oznacza długość wygenerowanego ciągu, K rozmiar uniwersum wartości zwracanych przez generator, l liczbę elementów X , które tworzą pojedynczy element Y^l . We wszystkich przykładach $k = 2^{30}$ (dni w roku), $n = 2^{11}$ (liczba urodzin), $\lambda = 2$ (parametr rozkładu Poissona). $w = \lfloor N'/n \rfloor$ oznacza liczbę podciągów Y^l , dla których wykonano test odstepu urodzin, otrzymując częściowe wyniki przed testem χ^2 .

Generator	N	K	l	w	p-wartość
Rule 30	10^4	2	30	0	—
Twister	10^7	64	5	976	0.883998
Twister 2	10^7	64	5	976	0.943152
RC4	10^7	32	6	813	0.899729
RC4cat	10^7	32	6	813	0.010135
GLCG	10^7	2^{10}	3	1627	0.000000

Dla generatora Rule 30 nie zamieszczono wyników, gdyż otrzymany ciąg urodzin był zbyt krótki, aby skorzystać z tego testu. Dla pozostałych generatorów otrzymano $\simeq 1000$ wyników cząstkowych, wystarczającą liczbę, by skorzystać z testu χ^2 . Generator RC4cat jest na granicy progu akceptacji, pozostałe generatory mają bardzo wysokie p wartości.

4.5 Wyniki zbiorcze

Poniższa tabela prezentuje zestawienie wyniki przeprowadzonych testów dla różnych generatorów (wartości prawdopodobieństw p dla odpowiednich testów):

Generator	test			
	χ^2	Kol-Smirnov	serii	odstępów
Rule 30	0.522173	0.252335	0.767294	—
Twister	0.027996	0.182904	0.931471	0.883998
Twister 2	0.686085	0.284762	0.674598	0.943152
RC4	0.061448	0.113426	0.045006	0.899729
RC4cat	0.571098	0.064289	0.620250	0.010135
GLCG	0.000000	0.000013	0.000000	0.000000

Można zauważyć, że wszystkie generatory, oprócz GLCG, przechodzą wszystkie testy. Co prawda zdarzają się sytuacje, gdzie uzyskana p wartość jest niewiele powyżej progu akceptacji 0.01. Patrząc ogólnie można stwierdzić, że najlepiej radzi sobie Rule 30 oraz druga wersja Twistera.

5 Testowanie losowości rozwinięć binarnych stałych matematycznych.

Ciekawym pomysłem na generator liczb losowych jest wykorzystanie rozwinięć (w rozważanym przypadku binarnych) pewnych niewymiernych (przestępnych) liczb. Przeprowadzono następujące testy dla rozwinięć binarnych liczb π , e , $\sqrt{2}$ długości odpowiednio 1004882, 1004882, 1004883.

5.1 Frequency monobit test

5.1.1 Teoria

Dla niezależnych zmiennych losowych $X_1, X_2, \dots$ o rozkładzie $P(X_i = -1) = P(X_i = 1) = 1/2$ rozważamy

$$S_n = \frac{1}{\sqrt{n}} \sum_{i=1}^n X_i.$$

Z centralnego twierdzenia granicznego wynika, że S_n ma rozkład dążący do $\mathcal{N}(0, 1)$, czyli dla próbek dających $S_n = x$ mamy p-wartość

$$p = P(|N| > |x|),$$

gdzie N oznacza zmienną $\mathcal{N}(0, 1)$. p jest więc prawdopodobieństwem, że zmienna o rozkładzie normalnym znajduje się na obustronnym ogonie powyżej $|x|$. Jeśli p ma małą wartość, dane najprawdopodobniej nie pochodzą z rozważanego rozkładu.

5.1.2 Implementacja

Rozważamy wygenerowany ciąg binarnych zmiennych losowych X_i . Hipotezą zerową jest, że użyty generator jest losowy, czyli X_i są niezależnymi zmiennymi losowymi o rozkładzie $P(X_i = 1) = P(X_i = 0) = 1/2$. Rozważamy zmienne $Y_i = 2X_i - 1$, dla których można zastosować *frequency monobit test*. Obliczamy wartość

$$s_n = \frac{1}{\sqrt{n}} \sum_{i=1}^n (2x_i - 1),$$

gdzie x_i są obserwacjami. Następnie liczymy

$$p = P(|N| > |s_n|),$$

gdzie N - zmienna o rozkładzie $\mathcal{N}(0, 1)$.

5.1.3 Wyniki

Wyniki testu *frequency monobit* dla rozważanych binarnych rozwinięć liczb przedstawiają się następująco:

Liczba	Cyfry	p-wartość
π	1004882	0.613721
e	1004882	0.926876
$\sqrt{2}$	1004883	0.817751

Co ciekawe, wszystkie otrzymane p-wartości są dosyć duże, nie ma zatem podstawy do odrzucenia hipotezy o braku losowości rozważanych rozwinięć binarnych.

5.2 Wielokrotny frequency monobit test

Część teoretyczną zaczerpnięto z [1].

5.2.1 Teoria

Dalej rozważamy ciąg niezależnych zmiennych losowych $X_1, X_2, \dots, X_n$ o rozkładzie $P(X_i = -1) = P(X_i = 1) = 1/2$. Tym razem jednak nie obliczamy p-wartości dla całego ciągu, ale dzielimy ciąg na m części, dla których liczymy p-wartości. Dla uproszczenia niech $n = m \cdot k$. Wtedy kolejne fragmenty mają następującą postać

$$X_{1+(i-1)k}, \dots, X_{ik} \quad \text{dla } i = 1, 2, \dots, m.$$

Dla każdego z tych podciągów obliczamy p-wartość, jak w sekcji 5.1, otrzymując m p-wartości $P_1, \dots, P_m$. Następnie ustalamy poziom istotności α .

Przy założeniu losowości ciągu X_i , zmienna P oznaczająca p-wartość ma rozkład $\mathcal{U}(0, 1)$. Zatem zmienna

$$Z_i = \begin{cases} 1, & P_i > \alpha, \\ 0, & \text{w przeciwnym przypadku,} \end{cases} \quad (15)$$

ma rozkład $P(Z_i = 1) = 1 - P(Z_i = 0) = 1 - \alpha$. Rozważamy

$$S_m = \frac{1}{\sqrt{m}} \sum_{i=1}^m \frac{Z_i - (1 - \alpha)}{\sqrt{\alpha(1 - \alpha)}}. \quad (16)$$

Z centralnego twierdzenia granicznego S_m ma w przybliżeniu rozkład $\mathcal{N}(0, 1)$. Ustalając prawdopodobieństwo błędu β z tablic, wyznaczamy (bądź wyliczamy w MATLABie) wartość $z_{1-\beta/2}$, że

$$P(|N| \leq z_{1-\beta/2}) = 1 - \beta.$$

Przekształcając (16) mamy

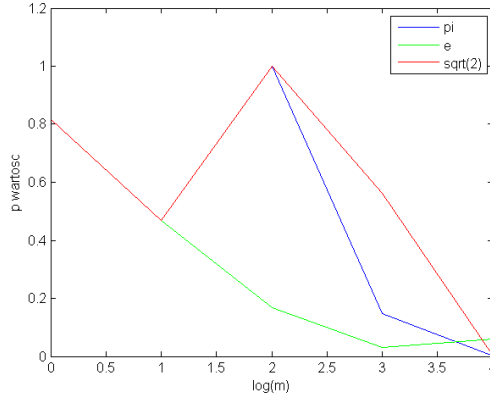
$$P\left(\frac{1}{m} \sum_{i=1}^m Z_i \in [(1 - \alpha) \pm z_{1-\beta/2} \sqrt{\alpha(1 - \alpha)/m}]\right) \approx 1 - \beta. \quad (17)$$

5.2.2 Implementacja

Rozważamy wygenerowany ciąg binarnych zmiennych losowych X_i . Dla podciągu $X_{1+(i-1)k}, \dots, X_{ik}$ obliczamy, zgodnie z zależnościami z sekcji 5.1.2, wartość p_i ($i = 1, 2, \dots, m$). Mając p_i bez trudu liczymy p-wartość zmiennej S_m (wzór (16)), wpierw wyliczając zaobserwowane Z_i (wzór (15)). Dodatkowo liczymy $\rho = \#\{i : p_i > \alpha\}/m$, czyli lewą stronę (17). Dla $\beta = 0.003$ zachodzi $z_{1-\beta/2} \approx 3$. Odrzucamy hipotezę, gdy ρ nie mieści się w przedziale wyznaczonym prawą stroną (17).

5.2.3 Wyniki

Na rys. 2 zebrano wykresy p-wartości dla $m \in 1, 10, 100, 1000, 10000$. Odrzucenie hipotezy następuje gdy $p < \beta$. Okazuje się, że obliczone wartości są poza przedziałem z (17) tylko dla rozwinięć $\pi, \sqrt{2}$ dla $m = 10000$. Może to być jednak spowodowane tym, że przy podziale ciągu na tak dużo fragmentów przybliżamy rozkład dwumianowy normalnym dla $\approx 10^6/m = 100$ obserwacji, co może być liczbą niewystarczającą do zastosowania centralnego twierdzenia granicznego. Ogólnie jednak otrzymane p-wartości są dosyć duże i nie można stwierdzić, że rozwinięcia rozważanych liczb nie są losowe.



Rysunek 2: Wykres p wartości dla testu *partial frequency monobit*.

6 Podsumowanie

Celem projektu było przetestowanie wybranych, popularnych generatorów liczb pseudolosowych. Do ich oceny wykorzystano testy χ^2 , Kołmogorowa-Smirnova, serii Stevensa oraz test odstępów dni urodzin. Okazuje się, że można samodzielnie zaimplementować najpopularniejsze testy i przeprowadzić eksperymenty na wystarczająco dużych zestawach danych, aby uzyskać istotne wyniki. Z wyników testów można zauważyć, że nie ma jednego idealnego generatora. Generatory dobrze radzące sobie na jednym zestawie testów mogą dawać gorsze wyniki na innych. Ważne zatem jest, aby testować generatory wieloma testami, sprawdzającymi różne cechy prawdziwej niezależności. Sądząc po wartościach p , rozwinięcia liczb

przestępnych mogą z powodzeniem być stosowane jako ciągi pseudolosowe. Ważnym aspektem w zastosowaniach generatorów jest także szybkość generowania kolejnych liczb losowych. Dla przykładu Rule 30 daje bardzo dobre wyniki we wszystkich zaimplementowanych testach, jednak generuje losowe bity bardzo wolno, wykorzystując przy tym dużo pamięci, zależnej od liczby wygenerowanych bitów.

Literatura

- [1] P. Lorek, Wprowadzenie do symulacji i metod Monte Carlo, projekt 1, dostęp 07.01.2018
http://www.math.uni.wroc.pl/~lorek//teaching/pliki/2017_symulacje_projekt1_ver2.pdf
- [2] T. Rolski, Symulacje stochastyczne i teoria Monte Carlo, dostęp 10.01.2018
<http://www.math.uni.wroc.pl/~rolski/Zajecia/sym.pdf>
- [3] Wikipedia, Rule 30, dostęp 10.01.2018
https://en.wikipedia.org/wiki/Rule_30
- [4] A. Rukhin et al., A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, dostęp 10.01.2018
<http://www.math.uni.wroc.pl/~lorek/teaching/files/nistspecialpublication800-22r1a.pdf>

Projekt 4 - symulacja produkcji na dwóch stanowiskach.

“Wprowadzenie do symulacji i metod Monte Carlo”, projekt nr 2.

Maciej Dulęba

Streszczenie

Celem projektu jest zasymulowanie produkcji pewnego produktu na dwóch stanowiskach i sprawdzenie jak wielkość magazynów przy tych stanowiskach wpływa na liczbę wyprodukowanych elementów, oraz prawdopodobieństwo zapewnienia ciągłości w produkcji przez cały dzień.

1 Zadanie [1]

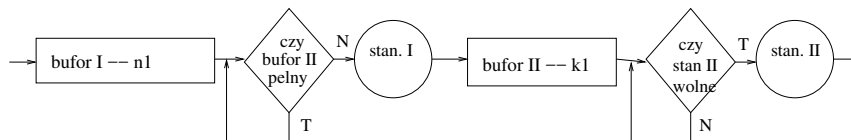
Mamy dane dwa stanowiska montażowe, na których składany jest dany produkt. Początkowo wszystkie n_1 komponentów (elementów) umieszczonych jest w magazynie. Komponenty kolejno trafiają na pierwsze stanowisko montażowe (oznaczane dalej jako I). Każdy element spędza na stanowisku I pewien czas T_1 , który jest zmienną losową. Następnie ten produkt trafia na drugie stanowisko montażowe (II), bądź też, jeśli stanowisko II jest zajęte, do kolejki przed stanowiskiem II, o rozmiarze k_1 . Zakładamy przy tym, że jeśli kolejka przed II stanowiskiem jest pełna, to na I stanowisku nie odbywa się montaż. Mogłoby się bowiem zdarzyć, że na stanowisku I zakończylibyśmy montaż wcześniej niż na II i ukończony produkt na stanowisku I nie mógłby trafić do pełnej już kolejki. Czas obsługi produktów na stanowisku II również jest pewną zmienną losową, T_2 . Dla danych rozkładów T_1, T_2 , należy tak dobrać parametry n_1, k_1 , aby z prawdopodobieństwem 90% praca trwała przez cały dzień, czyli 8h. Dodatkowo chcemy minimalizować $n_1 + k_1$. Dla niezerowej pojemności bufora II, działanie systemu można opisać automatem przedstawionym na rys. 1.¹

2 Model matematyczny

Wprowadźmy następujące oznaczenia:

- T_1^j - czas pracy nad j -tym elementem na stanowisku I.
- T_2^j - czas pracy nad j -tym elementem na stanowisku II.
- S_1^j - czas rozpoczęcia pracy nad j -tym elementem na stanowisku I.
- S_2^j - czas rozpoczęcia pracy nad j -tym elementem na stanowisku II.
- E_1^j - czas zakończenia pracy nad j -tym elementem na stanowisku I.
- E_2^j - czas zakończenia pracy nad j -tym elementem na stanowisku II.
- $N_{s1}(t)$ - liczba elementów, nad którymi zaczęto pracę na stanowisku I w chwili t .

¹Dla $k_1 = 0$ automat jest prostszy, zadanie może trafić na stanowisko I, gdy stanowisko II jest wolne.



Rysunek 1: Automat przepływu elementów przez system dwustanowiskowy dla niezerowej k_1 .

- $Ns_2(t)$ - liczba elementów, nad którymi zaczęto pracę na stanowisku II w chwili t .
- $Ne_1(t)$ - liczba elementów, nad którymi zakończono pracę na stanowisku I w chwili t .
- $Ne_2(t)$ - liczba elementów, nad którymi zakończono pracę na stanowisku II w chwili t .
- $Q(t)$ - liczba elementów w kolejce przed II stanowiskiem w chwili t .

Mamy następujące zależności (wynikające z definicji zmiennych):

$$T_i^j \sim T_i \quad \text{dla} \quad i = 1, 2, \quad (1)$$

$$E_i^j = S_i^j + T_i^j \quad \text{dla} \quad i = 1, 2, \quad (2)$$

$$Ns_i(t) = \max(k : S_i^k \leq t) \quad \text{dla} \quad i = 1, 2, \quad (3)$$

$$Ne_i(t) = \max(k : E_i^k \leq t) \quad \text{dla} \quad i = 1, 2, \quad (4)$$

oraz następującą równość

$$Q(t) = Ne_1(t) - Ns_2(t). \quad (5)$$

W bieżącej wielkości kolejki nie uwzględniamy elementów aktualnie montowanych na II stanowisku

$$S_2^j = \max(E_2^{j-1}, E_1^j). \quad (6)$$

Element j -ty zaczyna być przetwarzany na stanowisku II, gdy jest już ukończony na stanowisku I, oraz II stanowisko jest wolne.

$$S_1^j = \min(t : t \geq E_1^{j-1}, Ne_1(t) - Ne_2(t) \leq k_1) = \min(t : t \geq E_1^{j-1}, (j-1) - Ne_2(t) \leq k_1). \quad (7)$$

Element j -ty zaczyna być przetwarzany na stanowisku I, gdy poprzedni element został już na tym stanowisku ukończony. Dodatkowo musi się zmieścić w kolejce przed stanowiskiem II. Nie używamy tutaj zmiennej $Q(t)$, gdyż chcemy uwzględnić, czy jakiś element jest aktualnie montowany na II stanowisku. W zadaniu rozpatrywanym jako pierwsze, chcemy znaleźć n_1, k_1 , aby produkcja była zapewniona przez $8h$, czyli równoznacznie $E_2^{n_1} \geq 8$ (zakładając, że wszystkie dane odnoszące się do czasu będą podawane w godzinach) przy minimalnej sumie $n_1 + k_1$.

3 Algorytm

Działanie systemu zostanie zweryfikowane symulacyjnie (po implementacji odpowiedniego algorytmu) dla wybranych parametrów (n_1, k_1 i rozkładów zmiennych T_1, T_2). Zadania wchodzące na stanowisko I są numerowane, od 1, kolejnymi liczbami naturalnymi. W symulacji korzystamy z dyskretyzacji zdarzeń. Jakkolwiek opisany proces ma ciągłą naturę, to jednak stan produkcji zmienia się w skończenie wielu momentach – gdy na pewnym ze stanowisk rozpoczyna się lub kończy przetwarzanie elementu. W symulacji utrzymujemy zatem listę zadaniową zawierającą kolejne dyskretne momenty, w których stan systemu się zmienia. Niech t_i oznacza aktualnie rozważany moment czasowy, a s_1, e_1, s_2, e_2 , będą numerami zadań, które zaczęły/skończyły być przetwarzane na stanowiskach I, II (odpowiednio). Danymi początkowymi są $t_0 := 0, s_1 := 1, e_1 := 0, s_2 := 0, e_2 := 0$.

Określamy czas ukończenia elementu na stanowisku I/II, jeśli ono pracuje, jako

$$C_i := \begin{cases} E_i^{s_i} & \text{gdy } s_i < e_i, \\ \infty & \text{w p.p.} \end{cases} \quad i = 1, 2 \quad \text{odpowiednio, dla stanowiska I/II.} \quad (8)$$

W momencie t_i lista zadaniowa zawiera czasy $\{C_1, C_2\}$. Mamy

$$t_{i+1} := \min(C_1, C_2). \quad (9)$$

Można zauważyć, że jeśli $e_2 < n_1$, tzn. praca nie została jeszcze ukończona, to $t_{i+1} < \infty$. Musimy teraz uaktualnić s_1, e_1, s_2, e_2 według następujących reguł. Jeśli:

- $t_{i+1} = C_1$ - zwiększamy $e_1 := e_1 + 1$. Jeśli $s_2 = e_2$ (tzn. stanowisko II czekało na element ze stanowiska I), to $s_2 := s_2 + 1$, element ukończony na I przechodzi na II. Jeśli $s_1 < n_1$, oraz jeśli $e_2 - e_1 \leq k_1$ to nowy element może zostać zacząty na stanowisku I, zwiększamy $s_1 := s_1 + 1$.
- $t_{i+1} = C_2$ - zwiększamy $e_2 := e_2 + 1$. Jeśli $s_2 < e_1$ (tzn. są elementy w kolejce przed stanowiskiem II), to $s_2 := s_2 + 1$ i przetwarzamy nowy element na II. Jeśli $s_1 < n_1, s_1 = e_1$, (tzn. stanowisko I było zablokowane przez pełną kolejkę), to nowy element może trafić na stanowisko I, zwiększamy $s_1 := s_1 + 1$.

Podczas symulacji zachowujemy wektory zawierające istotne dane: t_i oraz s_1, e_1, s_2, e_2 dla każdego z przetwarzanych czasów. Symulacja jest zakończona, gdy $t_{i+1} = \infty$, tzn. $e_2 = n_1$, czyli wszystkie elementy zostały przetworzone na obu stanowiskach. Do zaimplementowania powyższej symulacji użyto języka R.

4 Symulacja

Rozważono następujące rozkłady czasu przetwarzania elementów (z przyjętymi wartościami parametrów):

- $T_1 \sim \text{Erl}(2, 10)$ - rozkład Erlanga,
- $T_2 \sim 0.8 \times \text{Exp}(9) + 0.2 \times \text{Exp}(3)$ - mieszanina rozkładów wykładniczych.

Dla powyższych rozkładów przeprowadzono symulację. Przy oznaczeniach z sekcji 2, szukamy n_1, k_1 , że $E_2^{n_1} \geq 8$ z prawdopodobieństwem 90%. Ustalmy n_1, k_1 . Niech zmienna losowa $Z = \mathbb{1}(E_2^{n_1} \geq 8)$. Liczone prawdopodobieństwo jest równe EZ . W pracy [1] podano ile replikacji zmiennej Z należy wygenerować, aby otrzymać estymator EZ zadaną dokładnością.

Założmy, że wygenerowano n replikacji $Z_1, Z_2, \dots, Z_n$, i otrzymano nieobciążony estymator

$$\hat{Z}_n = \frac{1}{n} \sum_{i=1}^n Z_i. \quad (10)$$

Dla uzyskania dokładność b na poziomie ufności α

$$P(|\hat{Z}_n - EZ| < b) = 1 - \alpha \quad (11)$$

musi zachodzić związek

$$n \geq z_{1-\alpha/2}^2 \frac{\text{Var}(Z)}{b^2}, \quad (12)$$

gdzie $z_{1-\alpha/2}$ jest $\alpha/2$ kwantylem rozkładu normalnego. Przyjmujemy poziom ufności $\alpha = 0.05$, wtedy $z_{1-\alpha/2} = 1.96$. Zauważmy, że do wyliczenia n potrzebna jest znajomość wariancji zmiennej Z . Jako że jest ona nieznana, używamy przybliżenia wariancją próbkową

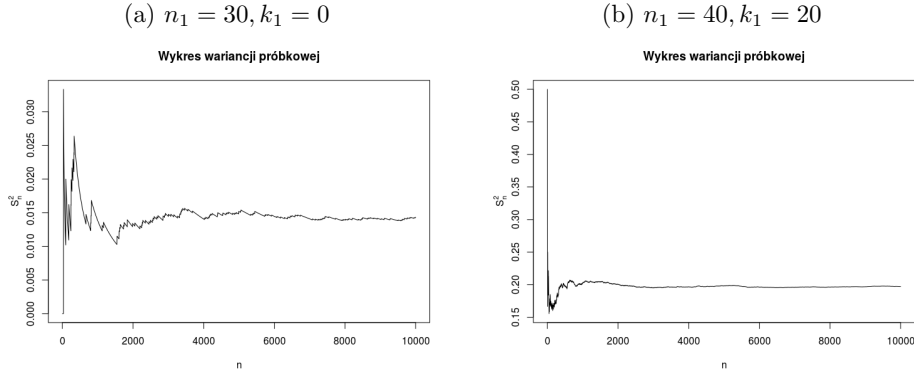
$$\text{Var}(Z) \approx \hat{S}_n^2 = \frac{1}{n-1} \sum_{i=1}^n (Z_i - \hat{Z}_n)^2. \quad (13)$$

Wykresy 2 przedstawiają $\hat{S}_n^2$ dla $n = 1, 2, \dots, 10000$, odpowiednio, dla par $n_1 = 30, k_1 = 0$ oraz $n_1 = 40, k_1 = 20$.

Na podstawie wykresów przyjęto $\text{Var}(Z) \approx 0.019$. Ustalono $b = 0.005$. Wtedy potrzebna liczba replikacji n pojedynczej symulacji spełnia $n > 2920$. We wszystkich dalszych rozważaniach przyjęto $n = 3000$ replikacji. Tabela 1 przedstawia wyniki symulacji.

5 Analiza wyników

Zauważmy, że optymalne pojemność bufora k_1 dla rozważanego problemu i ustalonych rozkładów T_1, T_2 (tzn. k_1 minimalizujące $n_1 + k_1$) wynosi $k_1 = 0$, co jest zgodne z intuicją: im krótsza kolejka przed II stanowiskiem, tym częściej praca na stanowisku I będzie wstrzymywana i mniej elementów będzie potrzebnych. Parametry n_1, k_1 o minimalnej sumie zapewniające działanie systemu przez cały dzień z prawdopodobieństwem 90% to $(n_1, k_1) = (27, 0)$. Udowodnimy następujący lemat:



Rysunek 2: Wykresy wariancji próbkowej.

k_1	n_1										
	23	24	25	26	27	28	29	30	31	32	33
0	0.53	0.65	0.75	0.85	0.90	0.95	0.97	0.99	0.99	1.00	1.00
1	0.04	0.06	0.10	0.15	0.22	0.28	0.38	0.48	0.56	0.66	0.75
2	0.01	0.02	0.03	0.04	0.07	0.10	0.16	0.24	0.29	0.38	0.47
3	0.01	0.01	0.02	0.03	0.04	0.06	0.10	0.14	0.20	0.27	0.34
4	0.01	0.01	0.01	0.01	0.03	0.04	0.07	0.09	0.14	0.21	0.26
5	0.00	0.01	0.01	0.02	0.02	0.04	0.05	0.08	0.12	0.17	0.22
6	0.00	0.01	0.01	0.01	0.02	0.03	0.05	0.07	0.11	0.16	0.21

Tabela 1: Prawdopodobieństwo działania systemu przez pełne 8h.

Lemat 1. Dla ustalonego n_1 , prawdopodobieństwo całodiennej (8h) pracy systemu dwustanowiskowego maleje wraz ze wzrostem k_1 (rozkłady T_1, T_2 dowolne, ustalone).

Dowód. Ustalmy n_1 oraz realizacje zmiennych T_i^j dla $i = 1, 2$. Rozważamy dwie wartości parametru k_1 : $k < k'$. Zmienne z sekcji 2 parametryzujemy teraz zmienną k_1 .

Indukcyjnie, względem j pokażemy, że $S_i^j(k') \leq S_i^j(k)$.

Dla $j = 0$ teza jest jasna.

Założmy, że teza zachodzi dla indeksów mniejszych niż j . Pokażemy ją dla j .

Z założenia indukcyjnego i (2) mamy

$$E_1^m(k') \leq E_1^m(k) \quad \text{dla} \quad m < j. \quad (14)$$

Zauważmy, że dla $t \leq S_1^j(k)$ z (4) oraz (14) wynika, że

$$Ne_2(t)(k') \geq Ne_2(t)(k). \quad (15)$$

Z (7) mamy

$$\begin{aligned} S_1^j(k') &= \min(t : t \geq E_1^{j-1}(k'), (j-1) - Ne_2(t)(k') \leq k') \leq \\ &\quad \min(t : t \geq E_1^{j-1}(k), (j-1) - Ne_2(t)(k') \leq k') \leq \\ &\quad \min(t : t \geq E_1^{j-1}(k), (j-1) - Ne_2(t)(k') \leq k) \leq \\ &\quad \min(t : t \geq E_1^{j-1}(k), (j-1) - Ne_2(t)(k) \leq k) = S_1^j(k), \end{aligned} \quad (16)$$

czyli także

$$E_1^j(k') \leq E_1^j(k). \quad (17)$$

Podstawiając (17) do (6) otrzymujemy

$$S_2^j(k') \leq S_2^j(k), \quad (18)$$

co kończy dowód indukcyjny.

Z $S_i^j(k') \leq S_i^j(k)$ łatwo wywnioskować tezę. □

W podobny sposób można udowodnić także:

Lemat 2. Dla ustalonego n_1 , oczekiwana liczba wykonanych zadań rośnie wraz ze wzrostem k_1 (rozkłady T_1, T_2 dowolne, ustalone).

6 Średnia liczba wyprodukowanych elementów

Zauważmy ciekawą własność. Dla przyjętych rozkładów T_1, T_2 mamy

$$ET_1 = 0.2, \quad ET_2 = 7/45 \approx 0.155,$$

czyli średnio stanowisko II pracuje szybciej niż stanowisko I. Gdyby obydwa stanowiska pracowały zawsze w swoim średnim czasie, to jeśli $k_1 > 0$ to stanowisko drugie nie będzie blokowało pracy na stanowisku I i będzie kończyło zadania o wiele wcześniej. W takiej sytuacji w czasie 8h łącznie powinno być wykonanych $8/0.2 - 1 = 39$ elementów (minus 1, gdyż ostatni element nie zostanie przetworzony na stanowisku II). Przeprowadzono eksperyment estymujący zmienną $e_2(8)$ przy nieograniczonym n_1 . W programie przyjęto wartość $n_1 = 120$ (ze względu na długość przeprowadzania symulacji dla wielu wartości n_1). Dla takiej wartości n_1 szansa, że na stanowisku I zabraknie elementów z nierówności Czebyszewa dla sumy rozkładów Erlanga jest mniejsza niż 0.01. Przy przeprowadzaniu symulacji rozumiemy analogicznie jak w przypadku symulacji zadania z całodziennym pracowaniem systemu z prawdopodobieństwem 90%.

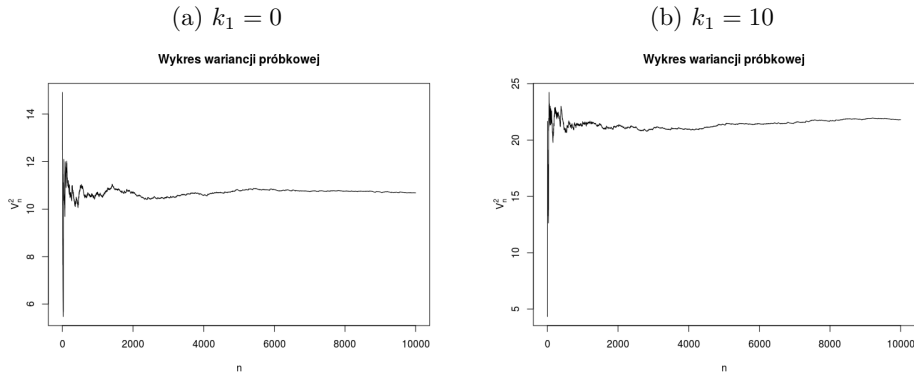
Ustalmy $n_1 = 120$ i wybraną wartość k_1 . Wprowadźmy zmienną losową $V = e_2(8)$. Szukamy EV . Załóżmy, że wygenerowano n replikacji $V_1, V_2, \dots, V_n$, i otrzymano nieobciążony estymator

$$\hat{V}_n = \frac{1}{n} \sum_{i=1}^n V_i. \quad (19)$$

Liczymy wariancję próbkową

$$\text{Var}(V) \approx \hat{V}_n^2 = \frac{1}{n-1} \sum_{i=1}^n (V_i - \hat{V}_n)^2 \quad (20)$$

Wykresy 3 przedstawiają $\hat{V}_n^2$ dla $n = 1, 2, \dots, 10000$, i, odpowiednio, par $n_1 = 120, k_1 = 0$ oraz $n_1 = 120, k_1 = 10$.



Rysunek 3: Wykresy wariancji próbkowej dla $n_1 = 120$.

Po analizie wykresów przyjmujemy $\text{Var}(V) \approx 20$. Ustalamy poziom ufności $\alpha = 0.05$ oraz $b = 0.2$. Aby osiągnąć dokładność daną (11) ze wzoru (12) mamy $n > 1921$. Przyjęto $n = 2000$ replikacji. Tabela 2 przedstawia wyniki symulacji.

rozkład	k_1								
	0	1	2	3	4	5	6	7	8
mieszanina wykładnicza	22.18	29.64	32.74	34.36	35.07	35.93	36.17	36.29	36.64
Pareto	22.18	31.91	34.94	35.77	36.62	36.95	37.07	37.18	37.32

Tabela 2: Oczekiwana liczba wykonanych elementów przez $8h$ dla ustalonego $n_1 = 120$ i zmiennego k_1 .

Widać zatem, Tab. 2, że dla $k_1 = 8$ otrzymujemy wyniki niewiele różniące się od 39. Rozważmy inny rozkład T_2 , o podobnej średniej, ale większych odchyleniach od niej (tzw. rozkład ciężkoogonowy). Niech

- $T_1 \sim \text{Erl}(2, 10)$ - rozkład Erlanga,
- $T_2 \sim \text{Par}(2, 7/90)$ - rozkład Pareto.

Wartość oczekiwana rozkładu Pareto jest identyczna z otrzymaną z mieszaniny rozkładów wykładniczych i wynosi $ET_2 = 7/45$. Licząc wariancję próbkową otrzymujemy $\text{Var}(V) \approx 25$. Dla poziomu ufności $\alpha = 0.05$ oraz $b = 0.2$ należy wykonać $n > 2401$ replikacji pojedynczej symulacji. Dalej przyjmujemy $n = 2500$. Tabela 2 przedstawia wyniki symulacji. Co ciekawe, wyniki nie różnią się znacząco dla obu przypadków rozkładu T_2 . Można by przypuszczać, że pojawiające się czasami w rozkładzie Pareto długie czasy wykonywania zadań na stanowisku II, i przez to potencjalnie większa szansa na zablokowanie stanowiska I powinna zmniejszyć oczekiwaną liczbę ukończonych elementów w tym samym czasie. Tak jednak nie jest, co więcej przy rozkładzie Pareto otrzymujemy średnio więcej wykonanych elementów.

7 Symulacja dla rozkładu ciężkoogonowego

Przeprowadzono także symulację dla wyjściowego problemu (liczba potrzebnych elementów zapewniająca całodzienną pracę z prawdopodobieństwem 90%) i zmienionego rozkładu T_2 , tym razem ciężkoogonowego. Rozważamy rozkłady

- $T_1 \sim \text{Erl}(2, 10)$ - rozkład Erlanga,
- $T_2 \sim \text{Par}(2, 7/90)$ - rozkład Pareto.

Analogicznie jak w początkowym przypadku wyliczono wariancję próbkową $\text{Var}(Z) \approx 0.23$. Ze względu na o wiele większą wariancję niż w pierwotnym zadaniu, tym razem przyjmujemy dopuszczalny błąd na poziomie $b = 0.01$. Ze wzoru (12) otrzymujemy potrzebną liczbę replikacji n pojedynczej symulacji większą niż 8835; wybrano $n = 9000$. Tabela 3 przedstawia wyniki symulacji. Wyniki są zbliżone do tych z Tab. 1 dla $k_1 = 0$ oraz 1, 5 – 2 razy mniejsze dla większych k_1 .

k_1	n_1										
	23	24	25	26	27	28	29	30	31	32	33
0	0.51	0.66	0.78	0.88	0.94	0.98	0.99	1.00	1.00	1.00	1.00
1	0.03	0.04	0.05	0.07	0.10	0.14	0.19	0.25	0.34	0.43	0.54
2	0.02	0.02	0.03	0.04	0.05	0.07	0.09	0.11	0.16	0.21	0.28
3	0.01	0.02	0.02	0.03	0.04	0.05	0.06	0.08	0.12	0.15	0.21
4	0.01	0.02	0.02	0.02	0.03	0.04	0.05	0.07	0.10	0.13	0.18
5	0.01	0.02	0.02	0.02	0.03	0.04	0.05	0.06	0.09	0.13	0.16
6	0.01	0.02	0.02	0.02	0.03	0.04	0.04	0.07	0.08	0.12	0.16

Tabela 3: Prawdopodobieństwo działania systemu przez pełne $8h$ dla T_2 Pareto.

8 Podsumowanie

W ramach projektu zasymulowano działanie systemu dwustanowiskowego z buforami wejściowymi. Czasy obróbki na poszczególnych stanowiskach były opisane wybranymi zmiennymi losowymi, a celem działania

systemu było zapewnienie (z prawdopodobieństwem 90%) całodienne działanie przy minimalizacji sumarycznej pojemności buforów. Uzyskano, że parametry n_1, k_1 o minimalnej sumie spełniające warunki zadania to $(n_1, k_1) = (27, 0)$. Teoretyczne policzenie prawdopodobieństwa całodiennej pracy dla ustalonych n_1, k_1 jest bardzo trudne i czasochłonne, zaś wykonana symulacja w przeciągu kilku minut potrafi podać wyniki z dokładnością do procenta. Zaobserwowano także, że zmiana rozkładu na stanowisku II (przy zachowaniu średniego czasu pracy) na rozkład Pareto nie wpływa znacząco na oczekiwaną liczbę wykonanych zadań przez $8h$. Kod symulacji napisano w języku R.

Literatura

- [1] T. Rolski, Symulacje stochastyczne i teoria Monte Carlo, dostęp 10.01.2018
<http://www.math.uni.wroc.pl/~rolski/Zajecia/sym.pdf>

WPROWADZENIE DO SYMULACJI I METOD MONTE CARLO

PROJEKT NR 1

Konrad Krystecki

Spis treści

Wstęp	2
1. Generatory	2
1.1. $LCG(M, a, c)$	2
1.2. $GLCG(M, a_{i=1}^k)$	2
1.3. Generator excellowski	3
1.4. RC4(N)	3
1.5. Exclusive OR Generator (XORG)	3
1.6. Knuth-TAOCP-2002	3
1.7. Mersenne-Twister [1]	3
1.8. Liczby niewymierne	3
2. Testy statystyczne	3
2.1. Test częstotliwości	4
2.2. Test serii	4
2.3. Test dyskretnej transformaty Fouriera [2]	4
2.4. Test pokerowy [3]	4
2.5. Test odstępów dni urodzin [3]	5
3. Rezultaty	5
3.1. Graficzne przedstawienia symulacji	5
3.1.1. $LCG(M, a, c)$	5
3.1.2. $GLCG(M, a_{i=1}^k)$	7
3.1.3. Generator excellowski	8
3.1.4. RC4(N)	9
3.1.5. Exclusive OR Generator (XORG)	11
3.1.6. Knuth-TAOCP-2002	12
3.1.7. Mersenne-Twister [1]	13
3.1.8. Liczby niewymierne	14
3.1.9. Wnioski	14
3.2. Wyniki liczbowe	14
3.2.1. Cały ciąg	14
3.2.2. Ciąg podzielony na fragmenty	15
4. Wnioski	15
4.1. LCG	15
4.2. GLCG i Knuth-TAOCP-2002	15
4.3. Excellowski	15
4.4. XORG	16
4.5. RC4	16
4.6. Mersenne-Twister	16
4.7. π	16
4.8. $e, \sqrt{2}$	16
4.9. Podsumowanie	16
Literatura	16

Wstęp

Celem projektu jest zbadanie zachowania różnych generatorów liczb psuedolosowych (ang. Pseudo Random Number Generator, PRNG). W poniższej pracy zostanie przedstawione pięć istotnie różnych generatorów. Rodzaje generatorów:

- a) Wygenerowane liczby naturalne z przedziału $[0, M]$
- b) Wygenerowane liczby rzeczywiste z przedziału $(0, 1)$
- c) Wygenerowane bity

Na potrzeby różnych testów przeprowadzane są następujące transformacje pomiędzy rodzajami wygenerowanych liczb

$$a) \rightarrow b) : Y \rightarrow \frac{Y}{M}$$

$$a) \rightarrow c) : Y \rightarrow \mathbb{1}(Y > \frac{M}{2})$$

$$b) \rightarrow a) : U \rightarrow \lfloor MU \rfloor$$

$$b) \rightarrow c) : U \rightarrow \mathbb{1}(U > \frac{1}{2})$$

Transformacje do bitów można by robić bardziej efektywnie patrząc na zapis binarny liczby, ale ten wybór wydawał mi się bardziej naturalny i z pewnością nie zmieniający losowości generatora. Poza zachowaniem PRNG poniższa praca bada cechy losowości rozwinięć dziesiętnych trzech różnych popularnych liczb rzeczywistych, co pozwoli sprawdzić czy mogą one zostać użyte jako generatory liczb losowych. Każdy generator użyłem do wygenerowania dwóch milionów liczb i na takiej próbce przeprowadzałem testy.

1. Generatory

W pracy będę rozważał zarówno proste (i naiwne) generatory jak i bardziej skomplikowane. Poniżej przedstawię krótkie informacje na temat każdego z nich, szersze opisy i dowody można odnaleźć w [1] lub [3].

1.1. $LCG(M, a, c)$

Pierwszym z generatorów jest bardzo prosty generator, gdzie kolejne liczby wyrażają się wzorem:

$$x_{n+1} = (x_n * a + c) \bmod M$$

Nawet przy odpowiednim doborze źródła oraz liczb M , a i c okres generatora będzie krótki, a otrzymywane liczby będą cykliczne, stąd spodziewanym jest, że generator nie będzie przechodził większości testów. Testowane są dwie krotki (M, a, c) :

- a) $(13, 1, 5)$
- b) $(2^{10}, 5, 7)$

1.2. $GLCG(M, a_{i=1}^k)$

Drugi z generatorów jest uogólnieniem pierwszego, kolejne liczby wyrażają się wzorem:

$$x_n = \sum_{i=1}^k x_{n-i} * a_i \bmod M$$

Dla odpowiednio dobranych liczb generator powinien być lepszy niż poprzedni, ale dla zadanych w projekcie parametrów niekoniecznie to zachodzi, testowane parametry to: $(2^{10}, (3, 7, 68))$.

1.3. Generator excellowski

Bardzo podobny generator do LCG, ale działający na przedziale (0,1). Kolejne liczby generowane są według następującego wzoru:

$$x_{n+1} = 0.9821x_n + 0.211327 \bmod 1$$

1.4. RC4(N)

Dwufazowy algorytm, który generuje losową permutację, a następnie tasuje ją i przy każdym nowym tasowaniu zwraca tasowaną wartość. Zazwyczaj używany w wersji N=256, w projekcie testowany jest w wersji N=32. Stosowane będą dwa rodzaje, wersja z kluczem jako funkcją czasu (oznaczana dalej przez (i)) oraz z generowaniem ciągów krótszej długości z różnymi, deterministycznymi kluczami (oznaczana dalej przez (ii)). Funkcja czasu to wzięcie aktualnej sekundy i odpowiednie przekonwertowanie z formatu R. W wersji (ii) brałem kolejne klucze zaczynając od 1, gdy doszedłem do 32, to jako kolejny element klucza brałem numer segmentu/32, następnie przesuwalem każdy element klucza o 1 aby uzyskać kolejny klucz. Dokładniejszy opis samego algorytmu RC4 można znaleźć w [3].

1.5. Exclusive OR Generator (XORG)

Źródłem w tym generatorze jest ciąg 127 bitów, a kolejne liczby wyrażają się wzorem:

$$x_n = x_{n-1} \oplus x_{n-127}$$

Pozornie prosty generator daje zadziwiająco dobre wyniki.

1.6. Knuth-TAOCP-2002

Wybrany przez mnie jeden z wbudowanych generatorów w R. Po przejrzaniu dokumentacji okazuje się, że jest to wersja GLCG z bardzo sensownie dobranymi parametrami:

$$x_n = x_{n-37} + x_{n-100} \bmod 2^{30}$$

ostatecznie generowane są liczby z przedziału (0,1), poprzez odpowiednie dzielenie.

1.7. Mersenne-Twister [1]

Jeden z najbardziej popularnych PRNG. Kolejne zmienne przedstawiane są jako ciągu ω bitów. Wzór generujący:

$$x_{i+n} = x_{i+m} \oplus (x_i^{upp} | x_{i+1}^{low})A$$

gdzie 'upp' oznacza $\omega - r$ górnych bitów, 'low' oznacza r dolnych bitów, — oznacza sklejanie, A to macierz rozmiaru $\omega * \omega$ składająca się z N_2 , gdzie $N_2 = 0, 1$.

1.8. Liczby niewymierne

Testom zostały poddane rozwinięcia dziesiętne trzech liczb niewymiernych: $\pi, e, \sqrt{2}$.

2. Testy statystyczne

W tym rozdziale dokonam prezentacji wykorzystywanych testów statystycznych, które mają zwerifikować losowość wybranych PRNG. Testy będą wykonywane w dwóch wersjach

- a) Test dla całego ciągu, sprawdzenie czy otrzymana p-wartość jest większa niż zadany poziom istotności α .

- b) Ciąg zostaje podzielony na fragmenty długości 1000. Dla każdego z fragmentów wykonywany jest osobno test, następnie dla danego poziomu istotności α liczone jest jaki odsetek p-wartości testów na fragmencie ciągu przekroczył α . Jeśli wartość odsetka wpadnie w zadany z góry przedział, test jest uznany za zdany.

2.1. Test częstotliwości

Najbardziej naturalny do wykonania test. Sprawdzamy, czy rozrzut danych po zadanym przedziale jest jednostajny. Dla każdego rodzaju danych korzystamy z innej statystyki testowej, w konsekwencji otrzymując testy badające sumę bitów - test chi-kwadrat (gdzie klatkami są po prostu liczby naturalne) oraz test Kołmogorowa-Smirnowa dla danych z odcinka (0,1). Główną wadą tego testu jest to, że nie sprawdza faktycznego rozłożenia danych, np. jeśli z 1000 liczb, pierwsze 500 byłoby jedynkami a kolejne 500 zerami to test uznałby, że zadany ciąg jest losowy, mimo że od razu widać, że taki ciąg z losowością ma niewiele wspólnego.

2.2. Test serii

Bardziej skomplikowany test, po części odpowiadający na problemy pierwszego z prezentowanych testów. Dzielimy ciąg na pary (nienachodzące na siebie). Zliczamy wystąpienia każdej z sąsiednich par i obliczamy test chi-kwadrat na otrzymanych liczbach. Ten test zauważa, jeśli nasz PRNG ma wyjątkowy krótki cykl, ponieważ niektóre pary będą występowały zdecydowanie częściej niż inne. Jeśli nasz PRNG ma wyjątkowy długi cykl to przy dwóch milionach obserwacji nie zdążymy go zauważyć.

2.3. Test dyskretniej transformaty Fouriera [2]

Jako wejście przyjmowany jest ciąg złożony z -1,1. Ciąg transformowany jest za pomocą dyskretniej transformaty Fouriera, tzn. według wzoru:

$$f_j = \sum_{k=1}^n x_k \exp(2\pi i(k-1)j/n)$$

gdzie n to długość ciągu, a x_k to jego kolejne elementy. Test sprawdza ile razy została przekroczona pewna wartość (inaczej mówiąc, sprawdza cykliczność ciągu). Jest to kolejny krok po teście serii, który powinien odrzucać coraz więcej generatorów (a z pewnością nie mniej).

2.4. Test pokerowy [3]

Podobnie jak w teście OPERM5, rozpatrujemy kolejne piątki liczb z ciągu. Podobnie jak w teście serii, wybrane piątki są rozłączne. Definiujemy wzorce:

- a) 5 różnych liczb = wszystkie różne
- b) 4 różne liczby = jedna para
- c) 3 różne liczby = dwie pary lub trójka
- d) 2 różne liczby = full albo kareta
- e) 1 różna liczba = wszystkie takie same

Liczmy prawdopodobieństwo teoretyczne, że dostaniemy r różnych liczb, obliczamy próbkowe częstotliwości i za pomocą testu chi-kwadrat porównujemy wyniki. Test ten nie wnosi zbyt dużo do testowania, ponieważ przy dużym zakresie generowania liczb prawdopodobieństwo, że wszystkie liczby będą różne dominuje pozostałe.

Test pokerowy wyraźnie nie sprawdził się, najpewniej lepsze byłoby dobranie maksymalnego numeru karty, ale nawet dla LCG(13,1,5), generator nie został odrzucony, mimo, że w każdej kolejnej piątce liczb otrzymywaliśmy pięć różnych numerów. Można by rozpatrywać kolejne ciągi bitów, tak, aby 'poprawić' losowość LCG, GLCG (gdzie zawsze będzie pięć różnych numerów), ale skoro nawet zmodyfikowany przechodzi test, to nie ma sensu tego poprawiać.

2.5. Test odstępów dni urodzin [3]

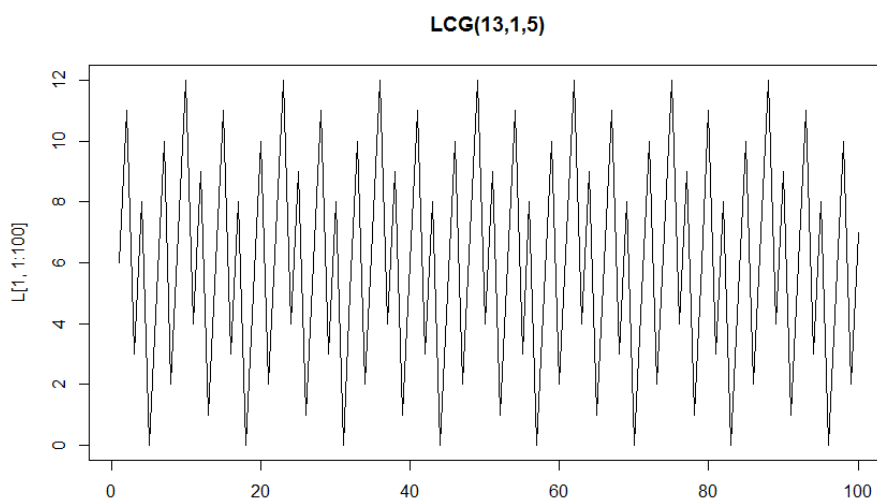
Zapisujemy n kolejnych dni urodzin osób (liczby z przedziału od 1 do k). Sortujemy otrzymane liczby, liczymy różnice kolejnych posortowanych dni urodzin. Liczbę równych odstępów oznaczamy przez K . Jeśli liczba $\frac{n^3}{4k}$ jest mała, to przy hipotezie o jednostajności rozkładu dni urodzin, K ma rozkład Poissona z parametrem $\lambda = \frac{n^3}{4k}$.

3. Rezultaty

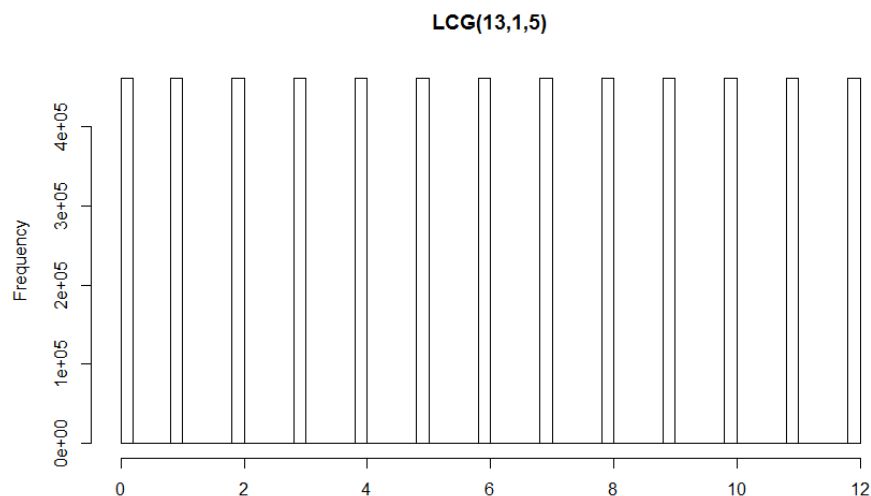
3.1. Graficzne przedstawienia symulacji

W tej części spojrzymy jak wyglądają dane i czy widać czym różnią się wygenerowane liczby. Pierwszy wykres będzie przedstawiał wykres pierwszych X wygenerowanych liczb (dla różnych generatorów różna długość ciągu wydawała mi się sensowna do obejrzenia), drugi będzie przedstawiał histogram z całej próbki. Chciałbym pokazać czy na podstawie tak prostych wykresów można odróżnić dobry generator od złego.

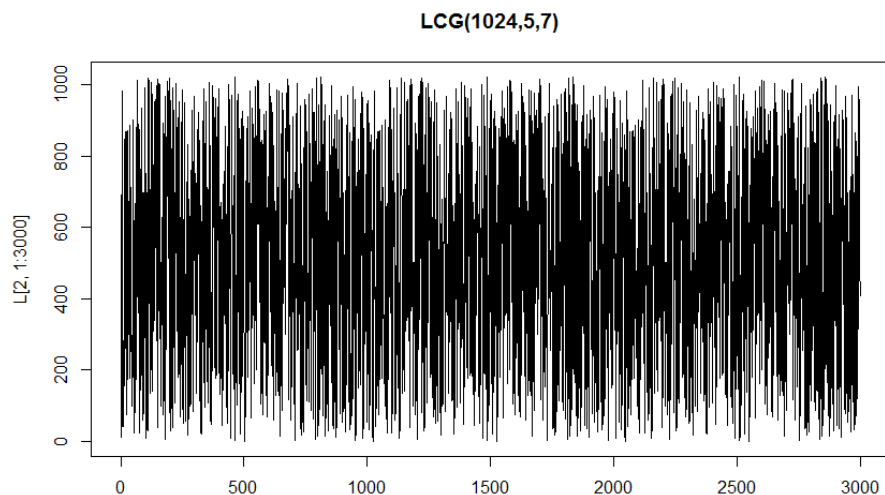
3.1.1. $LCG(M, a, c)$



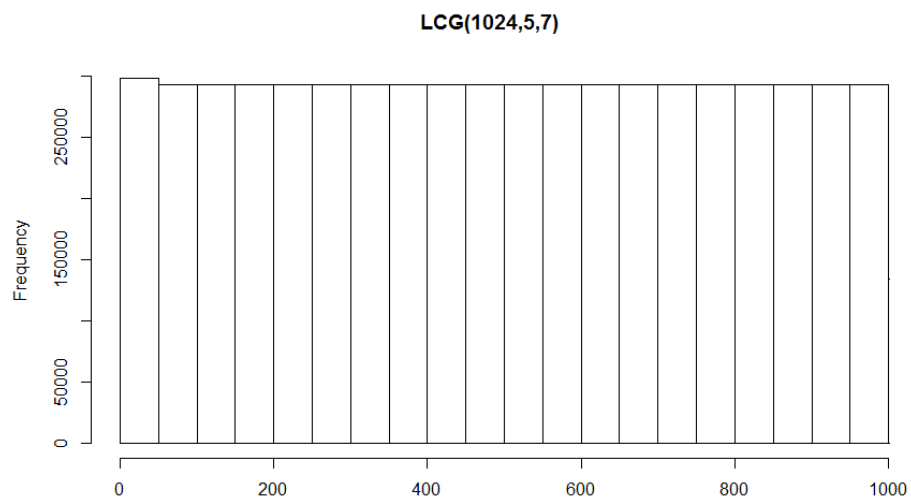
W związku z tym, że okres tego generatora jest wyjątkowo krótki, widać bardzo wyraźny wzór w jaki układają się liczby. Praktycznie na tej podstawie można by już przestać zajmować się tym generatorem dla wybranych parametrów.



Z drugiej strony, w związku z tym, że liczby są generowane tak idealnie cyklicznie, histogram jest perfekcyjny (z dokładnością do 1 wszystkich liczb będzie wygenerowanych tyle samo). Taki histogram też powinien wzbudzać podejrzenia, że coś jest zbyt idealne.

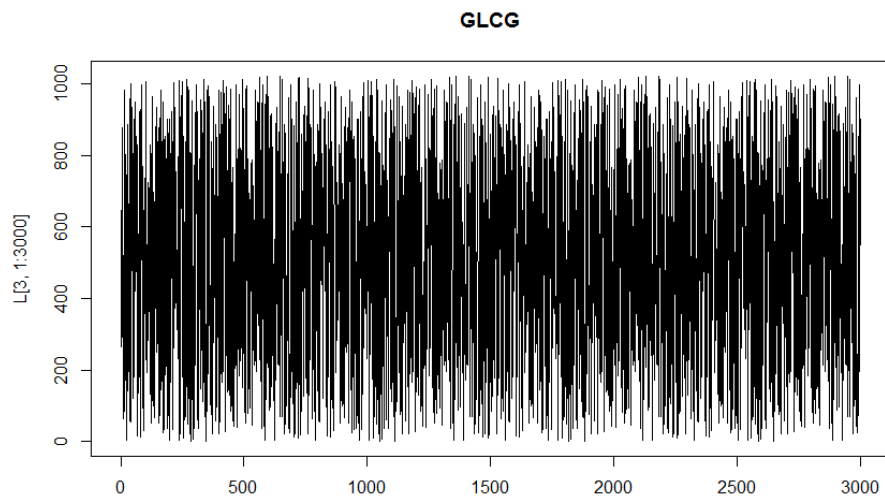


Dla trochę większych parametrów, zdecydowanie ciężiej jest znaleźć wzór w ścieżce wygenerowanej przez generator.

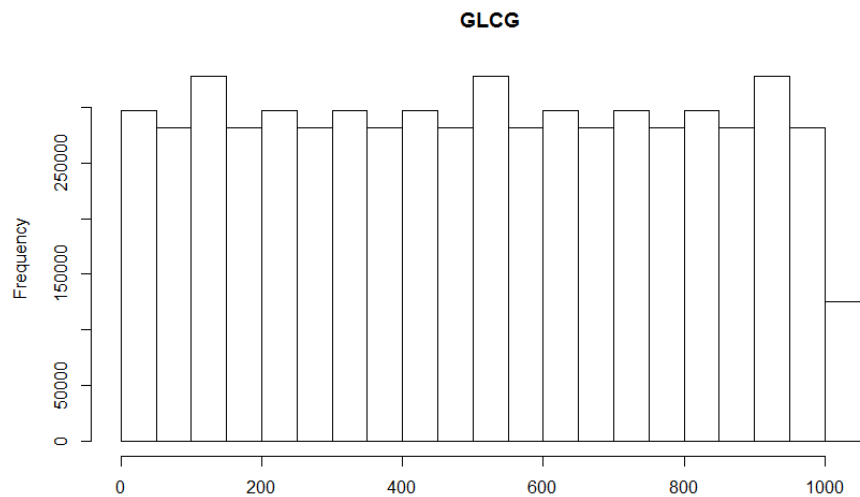


Pomijając błąd obliczeniowy R (przy zwiększeniu liczby podziałów histogramu wychodzi znowu prawie idealny prostokąt, słupki powinny różnić się o co najwyżej jeden, jednak wtedy histogram się zamazuje) widzimy, że histogram znowu jest 'idealny', co ponownie powinno być ostrzeżeniem.

3.1.2. $GLCG(M, a_{i=1}^k)$

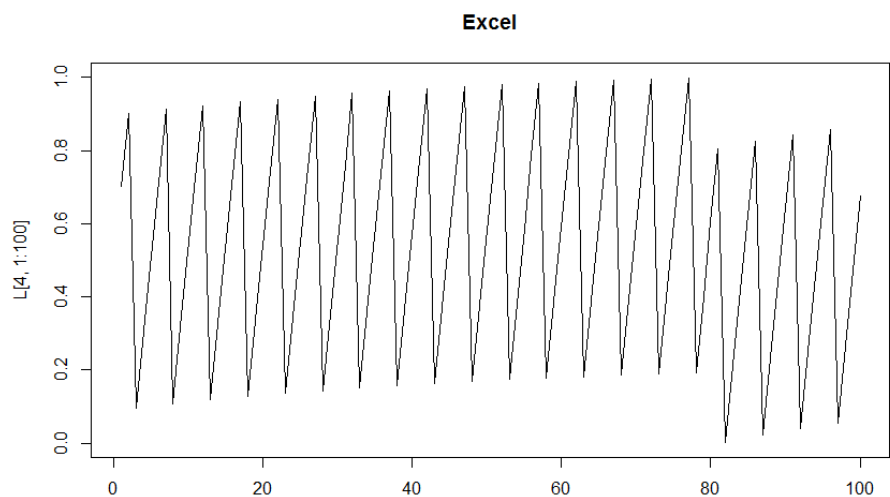


Znowu, duże parametry uniemożliwiają zobaczenie bardziej wyraźnych wzorów w przebiegu ścieżki.

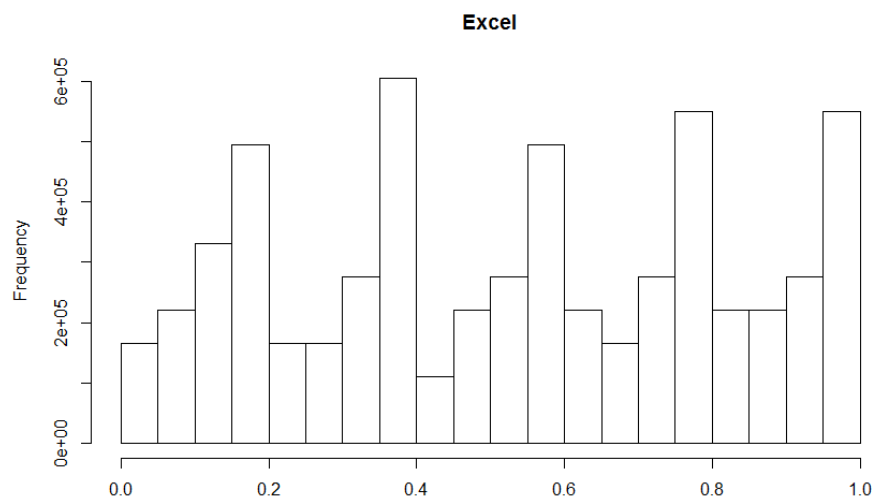


Tym razem histogram (z pominięciem ostatniego punktu, który jest kolejnym błędem funkcji hist w R) wygląda zdecydowanie gorzej i widać w nim regularności w podwyższaniu/obniżaniu słupków. Wynika to ze słabego doboru parametrów dla generatora, więc niektóre liczby występują zdecydowanie częściej.

3.1.3. Generator excellowski

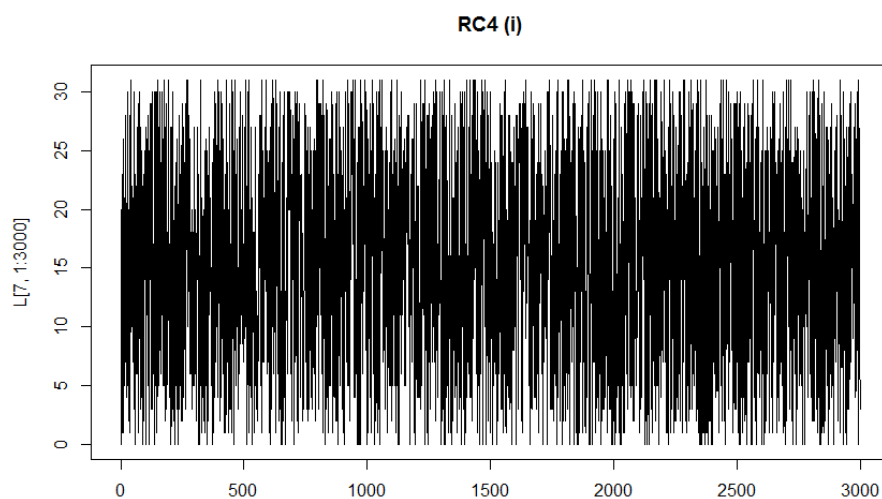


Widać bardzo wyraźny wzór w ścieżce, bez najmniejszych wątpliwości można odrzucić taki sposób generowania.

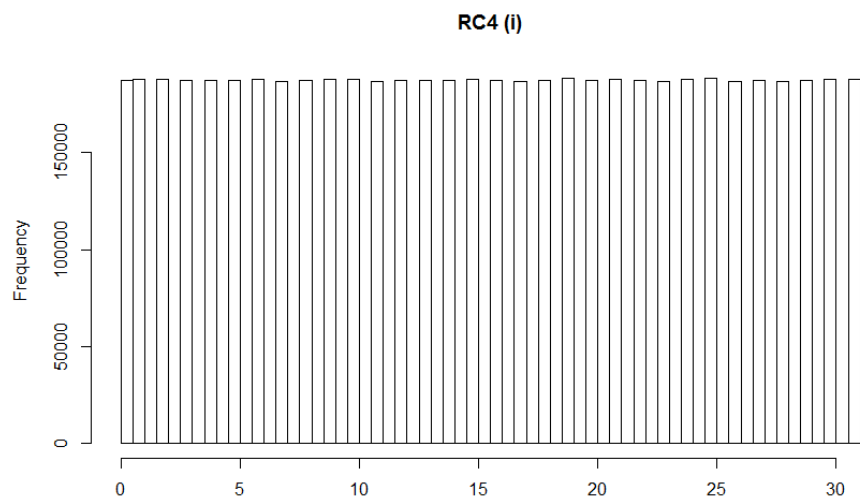


Potwierdza to tylko histogram, który bardzo wyraźnie odstaje od rozkładu jednostajnego.

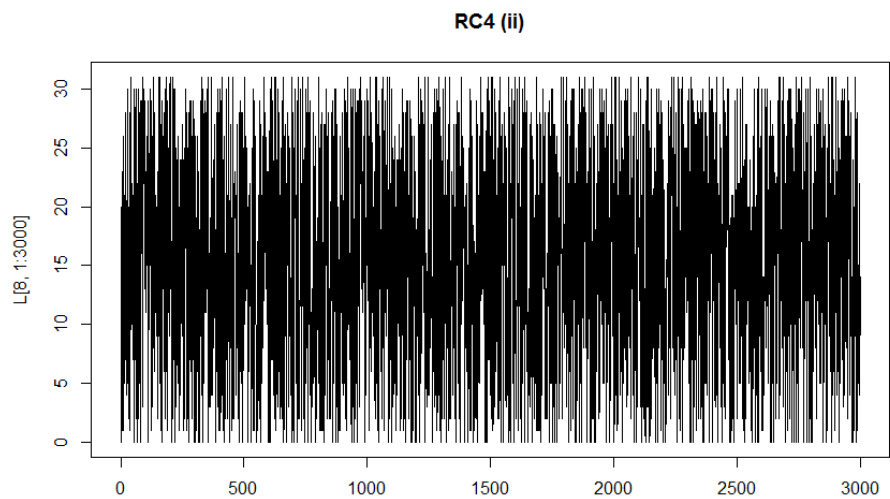
3.1.4. RC4(N)



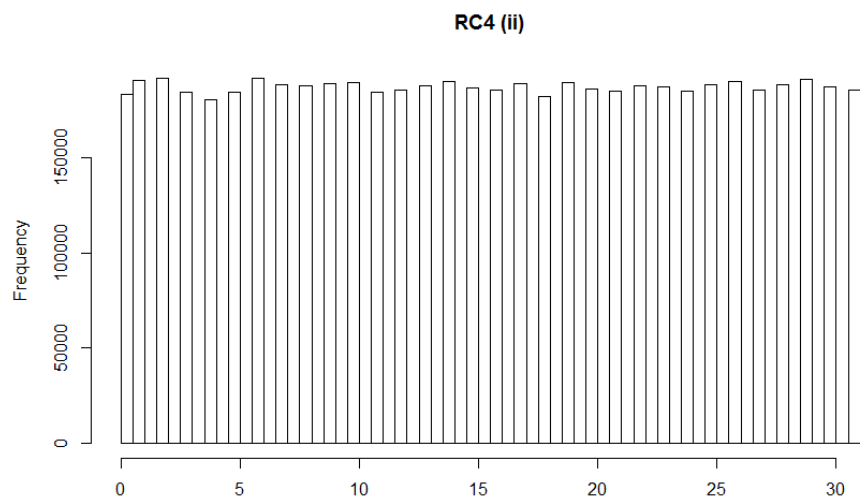
Znowu nie ma wyraźnego wzorca, nawet przy mniejszej liczbie potencjalnie wygenerowanych wartości.



Histogram wygląda lepiej niż w przypadku poprzednich generatorów, jest wystarczająco bliski, ale nie idealnie równy.

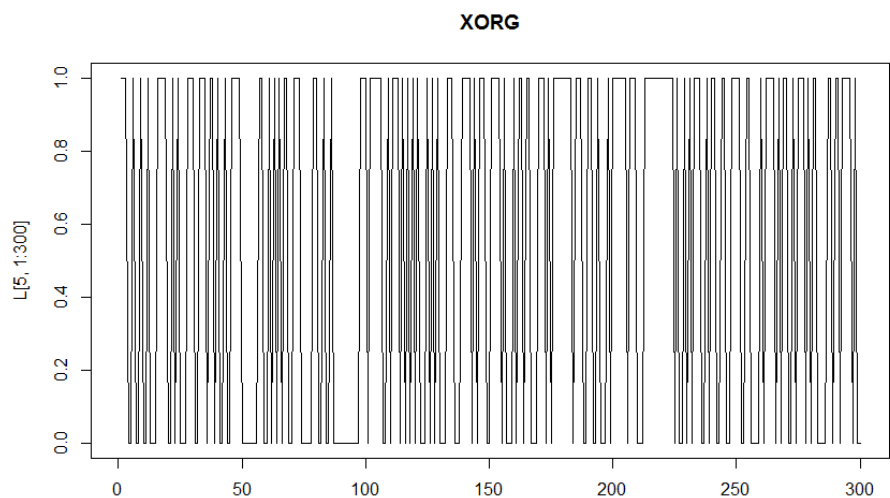


Ścieżka nie zawiera wyraźnego wzorca, podobnie jak w wersji (i).

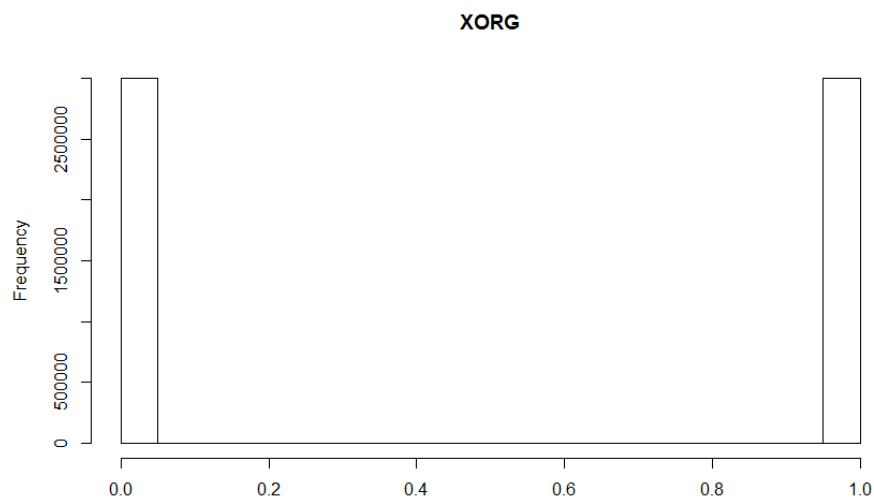


Histogram jest za to wyraźnie gorszy, odchylenia od oczekiwanej wartości są znacznie większe. Można by podejrzewać, że będzie problem z przechodzeniem testów, nawet frekwencyjnych. Najprawdopodobniej dobór kluczy był zły.

3.1.5. Exclusive OR Generator (XORG)

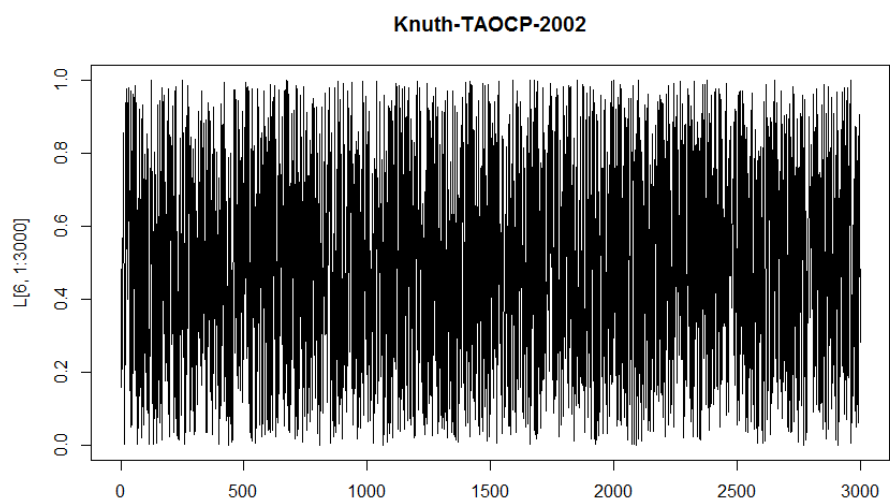


Ścieżka nie pokazuje żadnego wzorca.

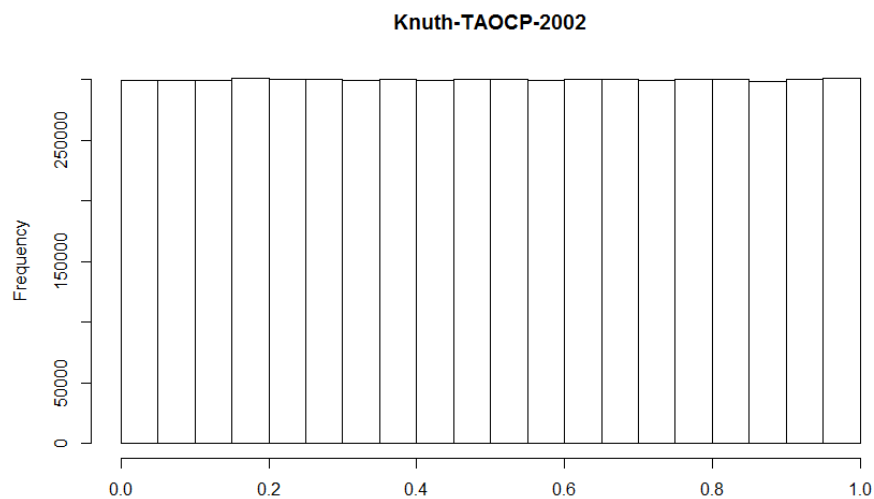


Ponieważ generator produkuje wyłącznie dwie wartości, histogram nie pokazuje nic interesującego.

3.1.6. Knuth-TAOCP-2002

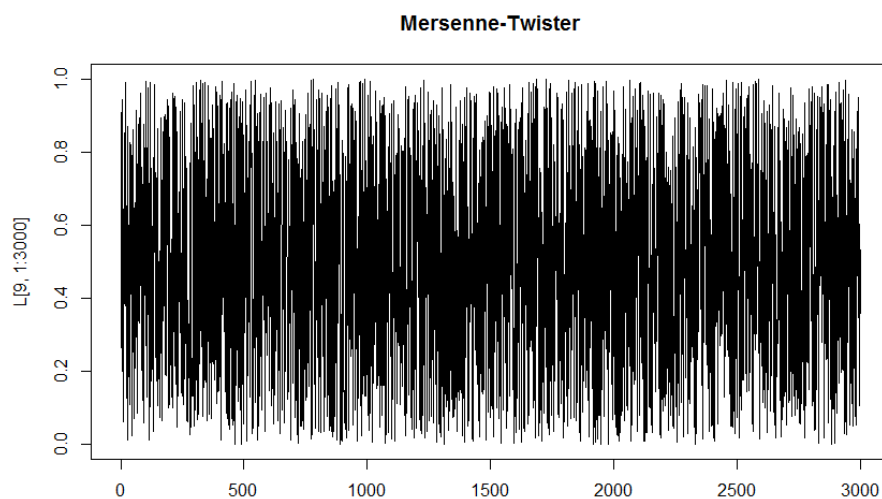


Podobnie jak dla GLCG, nie ma wyraźnego wzorca w ścieżce, szczególnie, że cykl jest teraz zdecydowanie dłuższy.

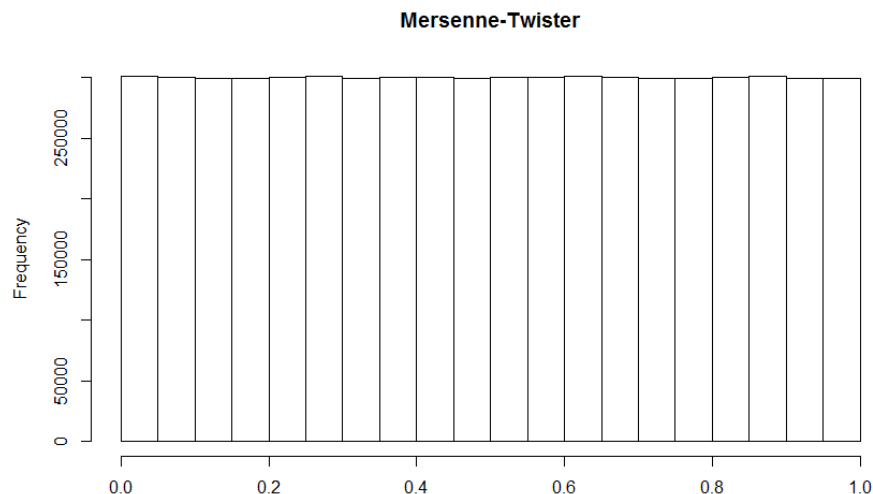


Widać, że dla lepszych parametrów GLCG też ma sens, histogram wygląda równo, ale nie idealnie, takiego histogramu można by spodziewać się po próbie z rozkładu jednostajnego.

3.1.7. Mersenne-Twister [1]



Podobna jakość generatora jak poprzedni pozwala oczekiwać, że znowu nie zobaczymy ścieżki.



Histogram podobny do poprzedniego generatora.

3.1.8. Liczby niewymierne

Wszystkie dane są bitami, więc prezentowane wykresy byłyby bardzo podobne do wykresów dla XORG, dlatego wykresy zostaną pominięte w tym przypadku.

3.1.9. Wnioski

Podobne wykresy pozwalają odróżniać wyłącznie bardzo złe generatory, wszystkie generatory, które będą choć trochę lepsze będą praktycznie nierozróżnialne.

3.2. Wyniki liczbowe

3.2.1. Cały ciąg

Poniższa tabela przedstawia otrzymane p-wartości dla próbek z każdego z generatorów. Tabela został podzielona na trzy części - najsłabsze generatory, lepsze generatory, rozwinięcia dziesiętne liczb niewymiernych. Liczby zostały zaokrąglone do trzech cyfr znaczących.

	Test częstotliwości	Test serii	Test Fouriera	Test pokerowy
<i>LCG</i> (13, 1, 5)	1	0	0	0.843
<i>LCG</i> (2 ¹⁰ , 5, 7)	1	0	0	0.999
<i>GLCG</i> (2 ¹⁰ , 3, 7, 68)	0	0	0	0.999
excellowski	0	0	0	0.998
XORG	0.649	0.638	0.075	N/A
RC4(32) (i)	0.494	0.46	0.133	1
RC4(32) (ii)	0	0	0	1
Knuth-TAOCP-2002	0.703	0.632	0.762	1
Mersenne-Twister	0.311	0.318	0.834	0.999
π	0.578	0.911	0.010	N/A
e	0.954	0.922	0.847	N/A
$\sqrt{2}$	0.812	0.930	0.582	N/A

3.2.2. Ciąg podzielony na fragmenty

Poniższa tabela przedstawia procent podciągów, które przeszły testy na poziomie istotności $\alpha = 0.05$. Tabela została podzielona na trzy części - najsłabsze generatory, lepsze generatory, rozwinięcia dziesiętne liczb niewymiernych. Liczby zostały zaokrąglone do trzech cyfr znaczących. W przypadku testu dni urodzin rozmiary podciągów są zależne od maksimum ciągu według wzoru: $k = \lfloor (8M)^{\frac{1}{3}} \rfloor$.

	Test częstotliwości	Test serii	Test Fouriera	Test pokerowy	Test urodzin
<i>LCG</i> (13, 1, 5)	1	0	0	1	1
<i>LCG</i> (2 ¹⁰ , 5, 7)	1	0	0	1	0.973
<i>GLCG</i> (2 ¹⁰ , 3, 7, 68)	0	0	0	1	0.901
excellowski	1	0	0	1	0.339
XORG	0.946	0.927	0.925	N/A	N/A
RC4(32) (i)	0.956	0.952	0.943	1	0.996
RC4(32) (ii)	0.91	0.971	0.939	1	0.997
Knuth-TAOCP-2002	0.946	0.953	0.948	1	0.991
Mersenne-Twister	0.96	0.934	0.952	1	0.991
π	0.951	0.952	0.926	N/A	N/A
e	0.955	0.962	0.934	N/A	N/A
$\sqrt{2}$	0.934	0.949	0.933	N/A	N/A

Aby test nie został odrzucony, wartość w tabeli musi zawierać się w przedziale

$$\rho \in [1 - \alpha \pm z_{1-\frac{\beta}{2}} \sqrt{\frac{\alpha(1-\alpha)}{m}}]$$

gdzie α to poziom istotności, β to pewna stała, a m to liczba podciągów. Przyjmując $\alpha = 0.05$, $\beta = 0.003$ oraz $m=1000$ otrzymujemy:

$$\rho \in [0.929, 0.970]$$

4. Wnioski

W poniższej sekcji przedstawię przemyślenia dotyczące wyników symulacji. Potencjalnie można wykonywać test pokerowy i urodzin dla generatorów bitowych łącząc kilka bitów w jedną liczbę, ale zmniejszyłoby to rozmiar próbki dla danych generatorów, a same testy nie dawały ciekawych wyników, więc nie zostało to sprawdzone.

4.1. LCG

Niezależnie od wersji, generator odpadał na każdym teście poza testami częstotliwości. Dla większego okresu wyniki wyszłyby lepsze. Najprostszy generator, bez zaskoczenia, że nie przechodzi testów.

4.2. GLCG i Knuth-TAOCP-2002

Jak wyraźnie widać po wynikach, generator mniejszy ze słabo dobranymi parametrami daje fatalne wyniki, najgorsze ze wszystkich generatorów. Jednak dobry dobór parametrów, jak w generatorze Knutha, pokazuje, że nawet GLCG może być dobrym generatorem. Generator Knutha obok Mersenne-Twistera był najlepszym generatorem z wybranych.

4.3. Excellowski

Po histogramie widać, że nie ma szans zostać uznany za losowy. O dziwo, niektóre z testów udało mu się przejść, ale tylko te, które były rozbite na mniejszych podciągach.

4.4. XORG

Ciekawy generator, prosty w implementacji. Nieco gorszy niż Knuth lub Mersenne-Twister, ale nadal przechodzi większość testów.

4.5. RC4

Jak widać, jeśli uzależnimy nasz klucz w jakiś sposób od czasu, to wyniki generowane przez RC4 są bardzo dobre, nieco gorsze niż Knuth i Mersenne-Twister, ale nadal przechodzą testy. Gdy zaś będziemy wprowadzać kolejne, krótkie, deterministyczne klucze, to wyniki stają się zdecydowanie gorsze i ta wersja nie przechodzi większości kluczowych testów.

4.6. Mersenne-Twister

Znany generator, ale dość skomplikowany w porównaniu do porównywalnie dobrze działającego generatora Knutha. Przechodzi wszystkie testy.

4.7. π

Rozwinięcie liczby π przechodzi najprostsze testy, ale ma problem z bardziej skomplikowanymi. Największy problem miało z testem Fouriera.

4.8. e , $\sqrt{2}$

Oba rozwinięcia dziesiętne dają bardzo dobre liczby losowe, przechodzą wszystkie zadane testy. Nie są tak dobre jak najlepsze generatory, ale są lepsze niż wszystko poza generatorem Knutha i Mersenne-Twisterem.

4.9. Podsumowanie

W wyniku testów i kalkulacji uważam, że wybrane generatory można podzielić na kilka klas:

- a) Najlepsze - Knuth-TAOCP-2002, Mersenne-Twister
- b) Bardzo dobre - e , $\sqrt{2}$
- c) Dobre - XORG, RC4 (i)
- d) Przyzwoite - π
- e) Słabe - pozostałe

Zdecydowanie za najlepsze pod względem ogólnym uważam Knutha oraz XORG ze względu na łatwość implementacji i zrozumienia oraz potencjalnie związaną z tym szybkość działania.

Literatura

- [1] <https://cran.r-project.org/web/packages/randtoolbox/vignettes/fullpres.pdf>
- [2] <http://www.math.uni.wroc.pl/~lorek/teaching/files/nistspecialpublication800-22r1a.pdf>
- [3] <http://www.math.uni.wroc.pl/~rolski/Zajecia/sym.pd>

WPROWADZENIE DO SYMULACJI I METOD MONTE CARLO PROJEKT NR 2

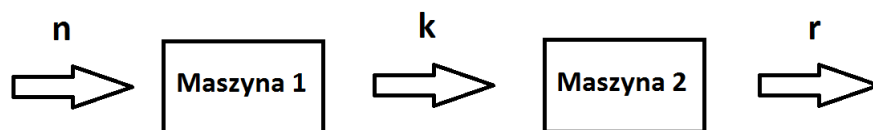
Konrad Krystecki

Spis treści

Wstęp	2
1. Funkcja kosztu niezależna od r	2
1.1. Wyniki	2
2. Funkcja kosztu zależna od r	3
2.1. Wyniki	4

Wstęp

W projekcie będę rozważał prosty model produkcji z dwoma maszynami. Celem projektu jest optymalizacja funkcji kosztów przy zadanych ograniczeniach oraz rozkładach czasów produkcji każdej z maszyn. Schemat pracy maszyn jest następujący:



n - liczba produktów na wejściu, k - rozmiar bufora drugiej maszyny, r - liczba wyprodukowanych produktów. W zadaniu przyjmujemy jako czas produkcji maszyny pierwszej przyjmujemy rozkład Erlanga(2,10), a czas produkcji maszyny drugiej jako mieszaną dwóch rozkładów wykładniczych: $0.8 \cdot \text{Exp}(3) + 0.2 \cdot \text{Exp}(9)$. Dodatkowe założenia, które przyjąłem:

- a) obie maszyny operują na swoich odpowiadających kolejkach (rozmiaru n dla pierwszej maszyny oraz co najwyżej k dla drugiej maszyny), tzn. nie ma żadnych produktów 'w środku' maszyn
- b) jeśli na kolejce do drugiej maszyny jest dokładnie k produktów, pierwsza maszyna nie pracuje

1. Funkcja kosztu niezależna od r

Zaproponowana została następująca funkcja kosztu: $C(n,k) = n+k$. Celem jest minimalizacja funkcji kosztu tak, aby

$$P(\text{w każdym momencie 8-godzinnego dnia pracy pracuje przynajmniej jedna maszyna}) > 0.9$$

Dla tak postawionego zadania zachodzi następujące fakty:

Fakt 1. *Obie maszyny nie pracują $\iff$ wszystkie produkty zostały wyprodukowane.*

Dowód Wynika bezpośrednio z założeń zadania. □

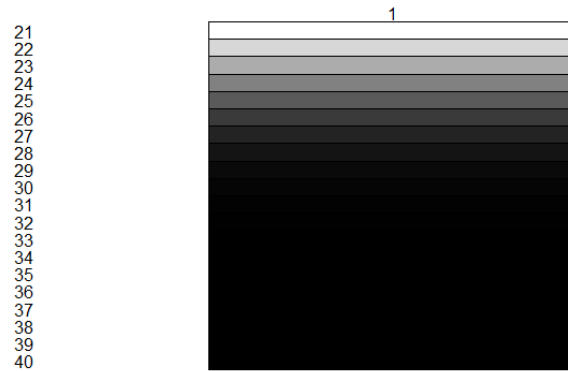
Fakt 2. *Jeżeli $C(n,k)$, $k > 2$, spełnia warunek 1, to $C(n,k-1)$ także spełnia warunek 1.*

Dowód Wynika bezpośrednio z faktu pierwszego oraz obserwacji, że zmniejszenie rozmiaru bufora k zmniejsza możliwości produkcyjne, tzn. wyprodukowanie tej samej liczby produktów przy mniejszym buforze k zajmie więcej czasu. □

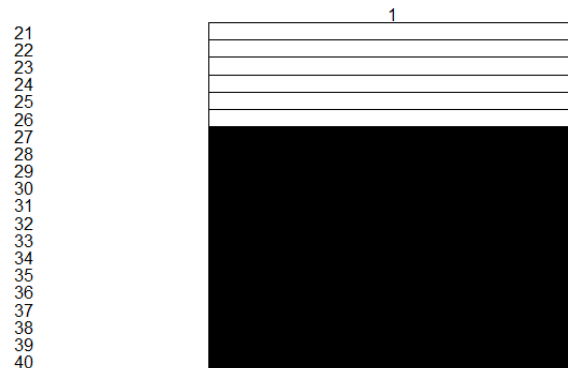
Powyższe dwa fakty pozwalają mi stwierdzić, że zadanie można sprowadzić do optymalizacji jednowymiarowej przy funkcji kosztu $C(n)=C(n,1)=n+1$.

1.1. Wyniki

Próba została przeprowadzona dla 100000 symulacji, szukałem najmniejszego n takiego, że warunek 1 jest spełniony. Im ciemniejszy wykres, tym większe prawdopodobieństwo.



Oczywiście wykres powinien być monotoniczny, co jest widoczne. Drugi wykres to indykator czy warunek 1 jest spełniony.



Widać, że optymalną parą dla postawionego problemu jest $(n,k)=(27,1)$.

2. Funkcja kosztu zależna od r

W dalszej części pracy chcę zaproponować jak zmodyfikować zadanie tak, aby optymalizacja do wykonania była naprawdę dwuwymiarowa, a nie trywializowała się do jednowymiarowego przypadku. Można zaproponować kilka rodzajów modyfikacji:

- Zmiana warunku 1 na taki, który wymusza czas pracy drugiej maszyny przez określony procent czasu w ciągu dnia
- Zmiana funkcji kosztu

Będę zajmował się drugą z przedstawionych możliwości, ponieważ wydaje mi się bardziej naturalna - za wykonaną pracę powinniśmy uzyskiwać jakąś korzyść, żeby promować więcej pracy. Najprostsza modyfikacja wyglądałaby następująco:

$$C(n, k, r) = n + k - (1 + \epsilon)E[r]$$

Ponieważ r jest losowe, musimy wprowadzić jakąś miarę na nim, najprostszą z pewnością jest wartość oczekiwana. ϵ ma na celu symulowanie marży jaką nakładamy jako producent przy sprzedawaniu naszych produktów. W dalszej pracy pomijamy całkowicie założenie 1, ponieważ przy podanej funkcji kosztu otrzymujemy już wystarczającą zachętę do działania, dodanie założenia odciełoby jedynie część rozwiązań z tablicy optymalizacyjnej. Zdecydowałem się także sprawdzić trochę zmodyfikowaną funkcję kosztu:

$$C(n, k, r) = n \log(n) + ck^2 - (1 + \epsilon)E[r]$$

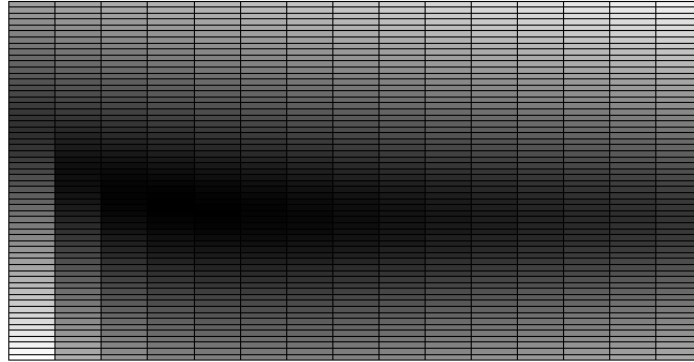
Zmiana k na k^2 wynikała z faktu, że bufor jest mały w stosunku do liczby produktów na wejściu, stąd duże jego wartości powinny być mocniej obciążone. Logarytm przy n oraz dodatkowa stała przy k mają już względy czysto wizualne, tak, aby wyniki miały lepszy rozkład na dwuwymiarowej planszy.

2.1. Wyniki

Pierwszy wykres zrobiony jest dla funkcji

$$C(n, k, r) = n + k - 2E[r]$$

, n zmienia się od 1 do 60, k od 1 do 15. W każdej komórce wykonanych zostało 5000 symulacji. Potencjalnie częścią zadania mogłoby być sprawdzenie jak zmiana ϵ wpływa na zmianę punktu optymalnego.

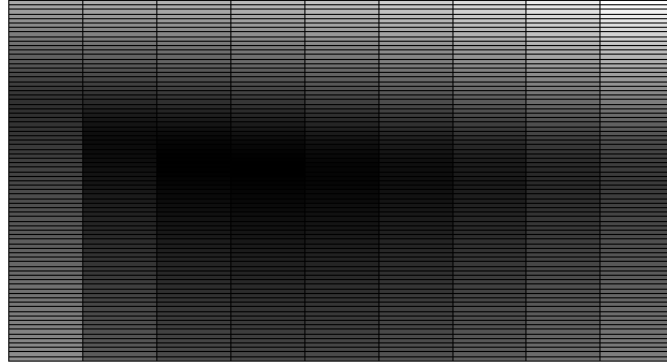


Optymalny dobór n i k to para $(n_0, k_0) = (35, 4)$, ale jeśli popatrzymy na n, k takie, że $C(n, k) < \frac{C(n_0, k_0)}{0.95}$, to dostaniemy niecałe 30 innych par, pośród których znajdziemy takie, które mają $k=3, 4, 5, 6$.

Dla porównania, poniżej prezentuję wykres zrobiony dla wartości n od 1 do 80 oraz k od 1 do 9 oraz funkcji kosztu:

$$C(n, k, r) = n \log(n) + 3k^2 - 20E[r]$$

W każdej komórce zostało wykonane 10000 symulacji i obliczona została funkcja kosztu. Im korzystniejsza jest dana komórka, tym bardziej jest czarna. Pole (1,1) znajduje się w lewym górnym rogu, na osi x znajdują się rozmiary bufora (k), na osi y rozmiary wejścia (n).



Optymalny dobór n i k to para $(n_0, k_0) = (38, 4)$, ale jeśli popatrzymy na n,k takie, że $C(n, k) < \frac{C(n_0, k_0)}{0.95}$, to dostaniemy ponad 30 innych par, pośród których znajdziemy takie, które mają k=3,4,5.

Bibliografia

- [1] .
- [2] Abramowitz M. & Stegun I. (1965) *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. National Bureau of Standards.
- [3] Aldous D. (1987) Ultimate instability of exponential back-off protocol for acknowledgment-based transmission control of random access communication channels. *IEEE Trans. on Inform. Theory* **IT-33**: 219–223.
- [4] Asmussen S. & Glynn P. (2007) *Stochastic Simulations; Algorithms and Analysis*. Springer, New York.
- [5] Asmussen S. (1999) *Stochastic Simulation with a view towards Stochastic Processes*. MaPhySto, University of Aarhus, www.maphysto.dk/cgi-bin/w3-mysql/publications/genericpublication.html?publ=89.
- [6] Bertsimas D. & Tsitsiklis J. (1993) Simulated annealing. *Statistical Science* **8**: 10–15.
- [7] Bratley P., Fox B., and Schrage L. (1987) *A Guide to Simulation*. Springer, New York.
- [8] Bremaud P. (1999) *Markov Chains, Gibbs Fields, Monte Carlo Simulations, and Queues*. Springer, New York.
- [9] Dagpunar J. (1988) *Principle of Random Variate Generation*. Clarendon Press, Oxford.
- [10] Davis M.H.A. (1993) *Markov Models and Optimization*. Chapman & Hall, London.

- [11] Diaconis P. Mathematical developments from the analysis of riffle shuffling .
- [12] Durrett R. (1996) *Probability: Theory and Examples*. Duxbury Press, Belmont.
- [13] Feller W. (1987) *Wstęp do rachunku prawdopodobieństwa*, vol. I. PWN, Warszawa.
- [14] Fishman G. (1966) *Monte Carlo*. Springer, New York.
- [15] Fishman G. (2001) *Discrete-Event Simulation*. Springer, New York.
- [16] Glasserman P. (2004) *Monte Carlo Methods in Financial Engineering*. Springer, New York.
- [17] Häggström O. (2000) *Finite Markov Chains and Algorithmic Applications*. Chalmers University of Technology, dmacs.rutgers.edu/dbwilson/exact.html/#haggstrom:course.
- [18] Kelly F. (1985) Stochastic models of computer communication systems. *J.R.Statist.Soc.* **47**: 379–395.
- [19] Kelly F.P. & MacPhee I. (198?) The number of packets transmitted by collision detect random access schemes. *Ann. Prob.* .
- [20] Kingman J. (1992) *Poisson processes*. Oxford University Press, Oxford.
- [21] Kingman J. (2002) *Procesy Poissona*. PWN, Warszawa.
- [22] Knuth D. (1997) *The Art of Computer Programming – Seminumerical Algorithms*, vol. II. Addison-Wesley, Reading, 3rd ed.
- [23] Knuth D. (2002) *Sztuka programowania – Algorytmy seminumeryczne*, vol. II. WNT, Warszawa.
- [24] Kolata G. (1990 Jan. 9) In shuffling cards, seven is winning number. *New York Times* .
- [25] Kopociński B. (1973) *Zarys teorii odnowy i niezawodności*. PWN, Warszawa.

- [26] Kotulski M. (2001) Generatory liczb losowych. *Matematyka Stosowana* **2**: 32–66.
- [27] L'Ecuyer P. (1998) Random number generation. In: J.E. Banks (ed.), *Handbook of Simulation*, chap. 4, 93137. Wiley.
- [28] L'Ecuyer P. (1999) Good parameters and implementations for combined multiple recursive random generation. *Oper. Res.* **47**: 159–164.
- [29] L'Ecuyer P. (2001) Software for uniform random number generation: distinguishing the good and the bad. In: B.P.J.S.D. Medeiros and M.R. eds (eds.), *Proceedings of the 2001 Winter Simulation Conference*, 95–105.
- [30] L'Ecuyer P. & Simard R. On the performance of birthday spacing tests with certain families of random number generators .
- [31] Levin D., Peres Y., and Wilmer E. *Markov Chains and Mixing Times*.
- [32] Lewis P.A.W. & Shedler G. (1979) Simulation of nonhomogeneous poisson process by thinning. *Naval Research Logistic Quartely* **26**: 403–413.
- [33] Madras N. (2002) *Lectures on Monte Carlo Methods*. AMS, Providence.
- [34] Marsaglia G. (1961) Generating exponential random variables. *Ann. Math. Stat.* .
- [35] Niederreiter H. (1992) *Random Number Generation and Quasi-Monte Carlo Methods*. Society for Industrial and Applied Mathematics, Philadelphia.
- [36] Rényi A. (1970) *Probability Theory*. Akadémia Kiadó, Budapest.
- [37] Resing J. (2008) Simulation. Manuscript, Eindhoven University of Technology.
- [38] Ripley B. (1987) *Stochastic Simulation*. Wiley, Chichester.
- [39] Rolski T., Schmidli H., Schmidt V., and Teugels J. (1999) *Stochastic Processes for Insurance and Finance*. Wiley, Chichester.
- [40] Ross S. (1991) *A Course in Simulation*. Macmillan, New York.

- [41] Rukhin A.e.a. (2010) *A Statistical Test Suite for Random and Pseudorandom Number Generator for Cryptographic Applications*. National Institute of Standards and Technology.
- [42] Savicky P. (2008) A strong nonrandom pattern in matlab default random number generator. Manuscript, Institute of Computer Science, Academy of Science of CR.
- [43] Srikant R. and Whitt W. (2008) Simulation run length planning for stochastic loss models. *Proceedings of the 1995 Winter Simulation Conference* **22**: 415–429.
- [44] van der Vaart A. (1998) *Asymptotic Statistic*. Cambridge University Press, Cambridge.
- [45] Whitt W. (1989) Planing queueing simulation. *Management Science* **35**: 1341–1366.
- [46] Whitt W. (1991) The efficiency of one long run versus independent replications in steady-state simulations. *Management Science* **37**: 645–666.
- [47] Whitt W. (1992) Asymptotic formulas for markov processes with applications to simulation. *Operations Research* **40**: 279–291.
- [48] Wieczorkowski R. & Zieliński R. (1997) *Komputerowe generatory liczb losowych*. WNT, Warszawa.
- [49] W.K. G. (2008) Warm-up periods in simulation can be detrimental. *PEIS* **22**: 415–429.
- [50] Zieliński R. (1970) *Metody Monte Carlo*. WNT, Warszawa.
- [51] Zieliński R. (1972) *Generatory liczb losowych*. WNT, Warszawa.
- [52] Zieliński R. i Zieliński W. (1990) *Tablice statystyczne*. PWN, Warszawa.

Skorowidz nazw

- n*-ty moment, 231
- średnia, 231
- średnia pęczkowa, 178
- średnia próbkowa, 251, 252
- błądzeniem przypadkowe
 - proste, symetryczne, 6
- generator liczb pseudolosowych, 16
- metoda akceptacji-eliminacji, 61
- Rozkład (dyskretny) jednostajny, 240
- rozkład ujemnie dwumianowy, 240
- zbieżność według prawdopodobieństwa,
 - 236
- zbieżność z prawdopodobieństwem 1,
 - 236
- algorytm, losowa permutacja17, ITM-Exp51, ITM-Par52, ITM-d53, DB55, ITR55, DP57, RM65, BM68, MB69
- algorytm , ITM51
- ALOHA
 - szczelinowa, 151
- alokacja proporcjonalna, 107
- asymptotyczna wariancja, 169
- asymptotyczne obciążenie, 170
- błądzenie przypadkowe, 267
- błądzenie przypadkowe
 - proste symetryczne, 268
- budżet, 92
- CCG, 22
- centralne twierdzenie graniczne, 89
- ciąg stacjonarny, 190
- CMC, 92
- CTG, 89
- czysty protokół ALOHA, 161
- długość rozpedówki, 177
- discrete event simulation, 139
- distribution
 - conditional, 233
 - Pareto, 242
- dni urodzin, 37
- dyskrepancja, 10
- dyspersja, 232
- dystrybuanta, 229
- dystrybuanta empiryczna, 30, 253
- efektywność, 92
- estymator, ł-trafił93, ążenie94, 251,
 - zgodny251, ążony251
- estymator antytetyczny, 114
- estymator chybił trafił, 94
- estymator istotnościowy, 121
- estymator mocno zgodny, 88
- estymator nieobciążony, 88
- estymator zgodny, 251
- FCFS, 260
- funkcja partycji, 206
- funkcja charakterystyczna, 235

- funkcja tworząca momenty, 235
- function
 - moment generating, 235
- fundamentalny związek, 89, 169
- funkcję tworzącą, 235
- funkcja
 - kwantyl, 254
- funkcja intensywności, 136
- funkcja prawdopodobieństwa, 229
- funkcje błędu (erf), 42
- gęstość, 230
- gęstością, 230
- Generator Fibonacciego, 22
- generator liczb losowych, 15
- GLCG, 21
- GLL, 15
- iloraz wiarygodność, 124
- kowariancja, 232
- LCFS, 260
- LCG, 21
- liczba losowa, 13
- logarytmicznie efektywny, 223
- losowy ciąg bitów, 16
- losowy ciąg liczb, 15
- macierz nieredukowalna, 188
- macierz przejścia, 187
- macierz stochastyczna, 187
- mediana, 232
- metoda środka kwadratu, 14
- metoda delty, 94
- metoda eliminacji, 61, ciągła 63
- metoda kongruencji liniowej, 19
- metoda warstw, 104
- metoda wspólnych liczb losowych, 115
- metoda zmiennych antytetycznych, 112
- metoda zmiennych kontrolnych, 117
- mieszanka, 230
- momenty zgłoszeń, 134
- niezależne zmienne losowe, 232
- niska dyskrepancja, 11
- obsługa z podziałem czasu, 260
- obsługa w losowej kolejności, 260
- obsługa w odwrotnej kolejności zgłoszeń, 260
- obsługa zgodna z kolejnością zgłoszeń, 260
- obszar akceptacji, 62, 63
- odchylenie standardowe, 232
- ogon, 229
- ogon dystrybuanty, 50
- ograniczony błąd względny, 222
- Pareto distribution, 242
- pierwszy moment, 231
- poziom istotności, 88
- poziom ufności, 88
- próba prosta, 251
- prawo arcusa sinusa, 7
- proces liczący, 134
- proces narodzin i śmierci, 141
- proces odnowy, 138
- proces Poissona, 135, niejednorodny 136
 - intensywność, 135
- proces Poissona , jednorodny 135
- proces stacjonarny, 166
- przedział ufności , 252
- PS, 260
- Q-Q wykres, 254
- QCG, 21
- quantile function, 254

- regulamin obsługi, 260
- replikacja, 7, 16
- rodzina zdarzeń rzadkich, 221
- rozkład
 - ciężki ogon, 239
 - lekki ogon, 239
- Rozkład Bernoulliego, 239
- rozkład Beta, 81
- rozkład beta, 241
- rozkład chi kwadrat, 241
- rozkład dwumianowy, 239
- Rozkład Erlanga, 241
- rozkład Frécheta, 242
- rozkład gamma, 241
- rozkład geometryczny, 240
- rozkład Gumbela, 80, 242
- rozkład jednostajny, 241
- rozkład jendostajny $U(A)$, 58
- rozkład kratowy, 229
- rozkład logarytmiczno normalny, 242
- rozkład logarytmiczny, 240
- rozkład loggamma, 242
- rozkład logistyczny, 79
- rozkład Makehama, 84
- rozkład normalny, 241
- rozkład obcięty geometryczny, 240
- rozkład odwrotny gaussowski, 242
- rozkład Pascala, 240
- rozkład połówkowo normalny, 64
- rozkład początkowy, 187
- rozkład podwójny wykładniczy, 81
- Rozkład Poissna, 239
- rozkład Raleigha, 67
- rozkład statystyki ekstremalnej, 242
- rozkład trójkątny, 79
- rozkład Weibulla, 242
- rozkład wielomianowy, 82
- rozkład wielowymiarowy normalny, 243
- rozkład wykładniczy, 241
- rozkład zdegenrowany, 239
- rozkład ekstremalny Weibulla, 242
- słaba zbieżność, 237
- SIRO, 260
- symetryczne błędzenie przypadkowe, 38
- symetrycznym błędzeniem przypadkowe
 - proste, 6
- symulacja asynchroniczna, 134
- symulacja synchroniczna, 134
- system sąsiedztwa, 194
- system sąsiedztwa
 - skomunkowany, 194
- TAVC, 169
- test kolizji, 35
- test odstępów dni urodzin, 37
- transformata Laplace-Stieltjesa, 235
- twierdzenie
 - Gliwienko-Cantelli, 253
- uogólniona funkcja odwrotna, 49
- uogólniona metoda kongruencji liniowej, 20
- wariancja, 231
- wariancja próbkowa, 251
- wartość oczekiwana, 231
- warunek wielkiego załadowania, 182
- warunkową gęstością, 231
- warunkowa wartość oczekiwana, 233
- warunkowe prawdopodobieństwo conditional, 233
- wektor losowy
 - absolutnie ciągły, 230
- współczynnik asymetrii, 232

współczynnik zmienności, 232

wzór Itô, 238

załadowanie, 261

zbiór ufności, 252

zbieżność według rozkładu, 237

zgrubny estymator Monte Carlo, 92

zmienna losowa

absolutnie ciągła, 230

dyskretna, 229