Metody Probabilistyczne i Statystyka

ZADANIE DOMOWE 4

Termin wysyłania (MS Teams): **28 stycznia 2024 godz. 23:59**

Zadanie 1. [1 pkt] *Nierówności ogonowe dla rozkładu dwumianowego* $\mathrm{Bin}(n, \frac{1}{2})$

Celem tego zadania jest porównanie ograniczeń otrzymanych przy użyciu nierówności Markowa i Czebyszewa na „ogony" rozkładu dwumianowego $\mathrm{Bin}(n, \frac{1}{2})$ dla $n \in \{100, 1\,000, 10\,000\}$ z dokładnymi wartościami szacowanych prawdopodobieństw.

Niech $X \sim \mathrm{Bin}(n, \frac{1}{2})$. Zastosuj nierówność Markowa oraz Czebyszewa do oszacowania

(a) $P(X \geqslant 1\frac{1}{5}\,\mathbf{E}\,(X))$,

(b) $P(|X - \mathbf{E}\,(X)| \geqslant \frac{1}{10}\,\mathbf{E}\,(X))$.

Do obliczenia dokładnych wartości prawdopodobieństw tych zdarzeń wykorzystaj wybrany pakiet matematyczny / bibliotekę do obliczeń numerycznych. W rozwiązaniu skorzystaj z faktu, że rozkład $\mathrm{Bin}(n, \frac{1}{2})$ jest rozkładem symetrycznym.

Przedstaw (np. w formie tabeli) i zwięźle omów uzyskane wyniki. Która z nierówności daje dokładniejsze oszacowania?

Zadanie 2. [2 pkt] *Błądzenie losowe na liczbach całkowitych*

Zdefiniujmy $S_N = \sum_{n=1}^{N} X_n$, gdzie zmienne losowe X_n, $1 \leqslant n \leqslant N$, są niezależne i każda przyjmuje wartości 1 oraz -1 z prawdopodobieństwem $\frac{1}{2}$. Dla $N = 0$ przyjmijmy $S_0 = 0$ (patrz np. wykład 11).

(a) Wyznacz numerycznie dystrybuanty zmiennych losowych S_N dla $N \in \{5, 10, 15, 20, 25, 30\}$. Dla każdego N wygeneruj odpowiednie wykresy / histogramy.

W tym celu możesz albo wyliczyć dokładne wartości odpowiednich prawdopodobieństw, albo przybliżyć dystrybuanty „eksperymentalnie", tj. wyznaczyć dystrybuantę empiryczną S_N albo podzielić zakres wartości $[-N, N]$ zmiennej losowej S_N na odpowiednią liczbę równych przedziałów i dla każdego z nich zliczać, ile spośród k wygenerowanych wartości (dla odpowiednio dobranej liczby powtórzeń k) jest nie większych niż górna granica danego przedziału.

(b) Porównaj wyznaczone dystrybuanty z dystrybuantą rozkładu normalnego, który miałby aproksymować rozkład S_N.

(c) Powtórz punkty (a) i (b) dla $N = 100$.

Przedstaw i zwięźle omów uzyskane wyniki oraz płynące z nich wnioski.

W rozwiązaniu zadania możesz skorzystać z wybranego narzędzia / pakietu obliczeń matematycznych (np. Matlab, Wolfram *Mathematica*, ...). W Matlabie do obliczania dystrybuanty rozkładu normalnego możesz użyć funkcji `normcdf`, a do wyznaczania dystrybuanty empirycznej dla danego zbioru wartości – funkcji `ecdf`. W Mathematice do wyznaczania dystrybuanty służy funkcja `CDF`, a do wyznaczenia dystrybuanty empirycznej można wykorzystać rozkłady empiryczne `EmpiricalDistribution` (patrz przykłady w dokumentacji). Możesz też np. wykorzystać histogramy z odpowiednio znormalizowaną (`cdf`) pionową osią, podobnie jak to jest opisane w zadaniu 3(c).

Zadanie 3. [2 pkt] *Błądzenie losowe na $\mathbb{Z}$ – rozkład „czasu spędzonego nad osią OX"*

Niech $X_1, X_2, \ldots$ będzie ciągiem niezależnych zmiennych losowych o rozkładzie jak w zadaniu 2. i niech $S_N = \sum_{n=1}^{N} X_n$ dla $N \in \mathbb{N}$. Ciąg zmiennych losowych (proces losowy) $(S_N)_{N\in\mathbb{N}}$ nazywamy prostym błądzeniem losowym na liczbach całkowitych.

Interpretacja tego procesu jest następująca: startujemy w punkcie 0 i w każdym momencie czasu $N \geqslant 1$ niezależnie z jednakowym prawdopodobieństwem idziemy jeden krok w górę ($X_N = 1$) albo jeden krok w dół ($X_N = -1$); przykładowa trajektoria takiego błądzenia losowego dla pierwszych 13 kroków przedstawiona jest na rys. 1.

Celem tego zadania jest eksperymentalne zbadanie rozkładu frakcji czasu, którą rozważany proces „spędza nad osią OX". Formalnie, niech

$$D_n = \mathbb{1}\left(S_n > 0 \vee S_{n-1} > 0\right), \quad n = 1, 2, \ldots,$$

tj. $D_n = 1$, gdy błądzenie losowe w chwili n albo znajdowało się nad osią OX, albo „zeszło w dół do 0"; w przeciwnym przypadku $D_n = 0$. Niech ponadto $L_N = \sum_{n=1}^{N} D_n$, tzn. L_N zlicza, w ilu momentach czasu od 1 do N błądzenie losowe znajdowało się nad osią OX. Oznaczmy $P_N = \frac{L_N}{N}$ („frakcja czasu" – dzielimy L_N przez łączną liczbę kroków N; $P_N \in [0, 1]$).

Dla każdego $N \in \{100, 1\,000, 10\,000\}$ wykonaj następujące czynności.

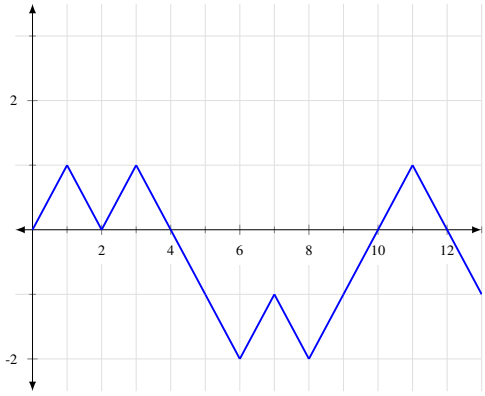
(a) Wygeneruj niezależnie $k = 5\,000$ realizacji procesu błądzenia losowego (S_N) (zasymuluj pierwsze N kroków) – oznaczmy te realizacje przez $(S_N)^{(1)}, \ldots, (S_N)^{(k)}$.

(b) Dla każdej realizacji błądzenia losowego $(S_N)^{(i)}$, $1 \leqslant i \leqslant k$, wyznacz wartość $P_N^{(i)}$ – frakcja czasu, którą błądzenie losowe „spędziło nad osią OX".

(c) Dla tak wyznaczonych wartości $(P_N^{(1)}, \ldots, P_N^{(k)})$ wygeneruj histogram z 20 „kubełkami" (ang. *bins*) równej szerokości (podziel przedział $[0, 1]$ na 20 rozłącznych podprzedziałów równej szerokości). Postaraj się znormalizować pionową oś histogramu tak, aby zamiast liczby wartości w poszczególnych kubełkach wyznaczona była estymacja funkcji gęstości prawdopodobieństwa. W tym celu np. w Matlabie w funkcji `histogram` możesz użyć opcji `'Normalization'` z wartością `'pdf'` (patrz dokumentacja; przykład: `histogram(X,'Normalization','pdf')`), a w Mathematice w funkcji `Histogram` możesz jako wartość argumentu `hspec` podać `"PDF"`.

(d) Porównaj uzyskany histogram z wykresem gęstości rozkładu arcusa sinusa z zadania 5. z listy 7 na ćwiczenia.

Przedstaw i zwięźle omów uzyskane wyniki oraz płynące z nich wnioski.



Rysunek 1: Przykładowa realizacja procesu prostego błądzenia losowego na liczbach całkowitych – pierwsze 13 kroków.

★ **Zadanie 4. [dodatkowe 3 pkt]** *Testowanie generatorów liczb pseudolosowych (PRNG)*

(a) Zapoznaj się pokrótce z ideą testów statystycznych (testowanie hipotez) – patrz np. rozdział 9.4 z podręcznika M. Baron, *Probability and Statistics for Computer Scientists*, 2nd Edition, Chapman & Hall/CRC Press, 2013.

(b) Zajrzyj do specyfikacji testów NIST dla generatorów liczb pseudolosowych – przejrzyj pokrótce wprowadzenie oraz przeczytaj opis kilku przykładowych testów. Postaraj się znaleźć intuicyjne wyjaśnienie dla poszczególnych kroków algorytmu, w szczególności zasad określających wynik testu.

(c) Zapoznaj się z narzędziem do przeprowadzenia testów NIST ze strony Zsolta Molnara. Poddaj testom NIST:

i) output „słabego" generatora liczb losowych (np. generatora *linear congruential generator, LCG*) z wybranego języka programowania (np. C, Java, Python, . . .),

ii) output „przyzwoitego" generatora liczb losowych (np. generatora *Mersenne Twister*) z wybranego języka programowania (np. C, Java, Python, . . .),

iii) pseudolosowe ciągi *„Radioactive decay RNG"*, *„ADC noise RNG"*, *„JavaScript pseudo RNG"* wygenerowane przez narzędzie na stronie Zsolta Molnara.

Zwięźle omów wyniki przeprowadzonych testów oraz płynące z nich wnioski.

(d) Załóżmy, że Kubuś Puchatek dysponuje źródłem doskonałej losowości, tzn. hipotetycznym generatorem dowolnie długich ciągów niezależnych bitów, z których każdy jest 0 lub 1 z prawdopodobieństwem $1/2$. Czy jeśli Kubuś Puchatek podda tak wygenerowany ciąg idealnie losowych bitów testom NIST, to będzie miał gwarancję, że wszystkie testy przejdą (*Passed*)? Odpowiedź uzasadnij.

Rozwiązanie zadania obejmujące

- implementację (kody źródłowe w wybranym języku programowania) oraz
- plik pdf ze sprawozdaniem

należy przesłać na platformę MS Teams. Nie należy dołączać żadnych zbędnych plików.