

Metody Probabilistyczne i Statystyka

ZADANIE DOMOWE 1

Termin wysyłania (MS Teams): **06 listopada 2022 godz. 23:59**

Zadanie 1. [5 pkt]

Niech $f: [a, b] \rightarrow \mathbb{R}_+$ będzie funkcją ciągłą na przedziale $[a, b]$ przyjmującą wartości nieujemne, dla której chcemy wyznaczyć przybliżoną wartość całki $\int_a^b f(x) dx$. Rozważmy przedstawioną na wykładzie 2. prostą probabilistyczną metodę aproksymacji takich całek.

1. Generujemy niezależnie i jednostajnie losowo n punktów z prostokąta $[a, b] \times [0, M]$ dla ustalonego $M \geq \sup\{f(x) : x \in [a, b]\}$.¹
2. Zliczamy, ile spośród wylosowanych punktów leży „pod wykresem” funkcji f (punkt (x, y) leży „pod wykresem” f , jeśli $y \leq f(x)$) – oznaczmy tę liczbę przez C .
3. Jako aproksymację całki przyjmujemy wartość $\frac{C}{n}(b-a)M$ (gdzie $(b-a)M$ to pole powierzchni rozważanego prostokąta).

Uwaga. Tego typu algorytmy zrandomizowane, jak powyższa procedura estymacji wartości całek, znane są jako metody Monte Carlo. Do tego zagadnienia wrócimy w drugiej części kursu (wtedy też przekonamy się, dlaczego ta metoda w ogóle „działa”).

a) Przetestuj działanie przedstawionego algorytmu do obliczenia wartości poniższych całek.

- $\int_0^8 \sqrt[3]{x} dx$
- $\int_0^\pi \sin(x) dx$
- $\int_0^1 4x(1-x)^3 dx$

W tym celu zaimplementuj i przeprowadź eksperymenty polegające na wykonaniu dla każdego $n \in \{50, 100, \dots, 5000\}$ po $k = 50$ niezależnych powtórzeń algorytmu. Dla każdej z aproksymowanych całek przedstaw na wykresie² wyniki uzyskane w poszczególnych powtórzeniach (k punktów danych dla każdego n), ich średnią dla każdego n oraz prostą $y = I$, gdzie I jest dokładną wartością całki. Wyniki kolejnych powtórzeń, ich uśrednione wartości oraz dokładną wartość całki nanieś na wspólny wykres (przykładowy wykres przedstawia Rys. 1). Wyciągnij wnioski z przeprowadzonych eksperymentów.

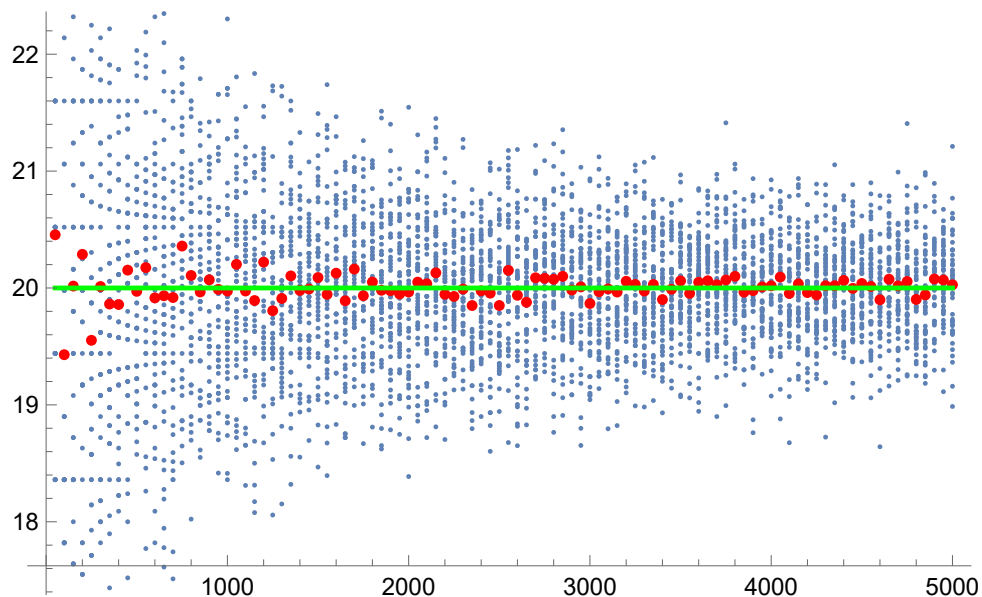
b) W podobny sposób wyznacz aproksymację liczby π .

HINT: Pole koła o środku $(1, 1)$ i promieniu 1 jest równe π .

Zadbaj o to, aby generator liczb pseudolosowych użyty w eksperymentach był „dobry” (tj. miał dobre własności statystyczne). Przykładowo, standardowa implementacja funkcji `rand()` w języku C nie jest dobrym generatorem. Możesz np. wykorzystać generator Mersenne Twister.

¹Jeśli dysponujemy jedynie generatorem pseudolosowym `rand()` zwracającym liczby z rozkładu jednostajnego na przedziale $[0, 1]$, to aby uzyskać losową liczbę z przedziału $[a, b]$, wystarczy zwrócić $a + (b-a) * \text{rand}()$. Poprawność tej metody uzasadnimy na jednym z późniejszych wykładów.

²Do wygenerowania wykresów możesz użyć dowolnego narzędzia, np. *numpy*, *Matlab*, *Excel*, *Mathematica*, ...



Rysunek 1: Wyniki eksperymentów dla całki $\int_1^3 x^3 dx$. Dla każdego $n \in \{50, 100, \dots, 5000\}$ wykonano po $k = 50$ niezależnych powtórzeń algorytmu. Niebieskie punkty przedstawiają wyniki poszczególnych powtórzeń, czerwone punkty odpowiadają wartości średniej dla każdego n , a zielona prosta $y = 20$ to prawdziwa wartość aproksymowanej całki.

Rozwiązanie zadania obejmujące

- implementację symulacji (kod źródłowy w wybranym języku programowania) oraz
- uzyskane wyniki i wyciągnięte na ich podstawie wnioski (pdf z wykresami wraz z krótkim opisem + wnioskami)

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