

Generator Liczb (pseudo)Losowych

Filip Szymański

1. Opis teoretyczny problemu

Głównym celem projektu jest stworzenie generatora całkowitych liczb pseudolosowych, oraz na podstawie jego pochodne funkcje generujące liczby losowe z rozkładów kolejno: jednostajnego, Bernoulliego, dwumianowego, Poissona, wykładniczego i normalnego. Następnie dane funkcje należy przetestować testem sprawdzającym losowość, oraz testem sprawdzającym poprawność rozkładu.

Hipotezy badawcze: czy funkcje generują poprawne rozkłady i są losowe.

2. Opis implementacji rozwiązania

Klasa Generator:

Zawiera wszystkie funkcje do generowania liczb losowych ze wszystkich zadanych rozkładów oraz konstruktor, którego parametrem jest seed no generatora G na podstawie którego działają wszystkie inne generatory.

G(left, right):

Jest to główny generator do liczb całkowitych, bazowany na generatorze Mersenne Twister (działa na podstawie liniowych przekształceń macierzy).

Parametry wejściowe określają zakres na jakim generowane są liczby (maksymalny zakres to $[0, 2^{33} - 1]$).

J(a, b):

Generator liczb z rozkładu jednostajnego, prosty w działaniu.

Argumenty określają zakres generowanych liczb (najlepsza jakość liczb dla przedziału $[0, 1]$).

B(p):

Generator liczb z rozkładu Bernoulliego, prosty w działaniu.

Argument określa prawdopodobieństwo zdarzenia.

D(n, p):

Generator liczb z rozkładu dwumianowego, symuluje n testów Bernoulliego z prawdopodobieństwem p.

Argumenty określają parametry rozkładu.

P(lambda):

Generator liczb z rozkładu Poissona, działa na zasadzie symulacji odległości między wydarzeniami z rozkładu Poissona rozkładem jednostajnym.

Argument określa parametr rozkładu.

W(lambda):

Generator liczb z rozkładu wielomianowego, prosty w działaniu.

Argument określa parametr rozkładu.

N(mu, sigma):

Generator liczb z rozkładu normalnego, funkcja działa na podstawie transformacji Boxa-Mullera.

Argumenty określają parametry rozkładu.

runs_test(gen, n, out):

Funkcja przeprowadzająca testy serii dla 3 podstawowych funkcjach generujących liczby (**G()**, **J()**, **B()**),.

Argumenty:

- Generator którego używamy
- Ilość danych na których przeprowadzamy testy
- Tablica na wyniki testów

Jest to implementacja standardowej funkcji do sprawdzania losowości danych. Dzielimy dana na dwa zbiory, następnie szukamy ilości serii liczb należących do tej samej grupy oraz porównujemy ją z ilością danych w poszczególnych grupach. Wyjściowa wartość powinna mieć rozkład normalny. Jeśli prawdopodobieństwo otrzymania danej liczby z rozkładu normalnego jest bardzo małe uznajemy, że dane nie są losowe w przeciwnym wypadku uznajemy że są.

Chi_Square_X(gen, n, params):

Funkcje wykonujące test chi kwadrat dla danego rozkładu funkcji X().

Argumenty:

- Generator którego używamy
- Ilość danych na których przeprowadzamy testy
- Ewentualnie parametry do rozkładu

Jest to implementacja testu chi kwadrat sprawdzającej zgodność z zakładanym rozkładem. Test polega na sprawdzenie odchylenia ilości danych w pewnych przedziałach od ilości spodziewanych. Odchylenie to powinno być wartością o rozkładzie chi kwadrat z $k - 1$ stopniami swobody, gdzie k to ilość przedziałów.

Funkcja zwraca wynik testu.

W obu testach wartość znaczącą uznałem jako 5%

3. Eksperymenty

Testy serii przeprowadziłem na różnych ilościach danych od 10000 do nawet 100000000, za każdym razem nie było żadnych problemów z ich przejściem.

Testy chi kwadrat przeprowadzałem na wszystkich funkcjach na różnych ilościach danych (od 1000 do 10000000) z różnymi parametrami i też nie było problemów z ich przejściem.

Przeprowadzając 1000 kolejnych testów z takimi samymi parametrami akceptacja była na poziomie 93% co jest zadowalającym wynikiem.

4. Interpretacja Wyników

Wszelkie testy wykazują, że generatory generują dane (pseudo)losowe z poprawnymi rozkładami więc hipoteza z punktu 1. wydaje się być potwierdzona. Niezależnie od wyników testów oraz ich ilości nigdy nie mamy pewności czy nasze liczby są losowe, oraz jest to wprost niemożliwe ponieważ przeczy to naturze komputerów, lecz można stwierdzić, że generatory są wystarczająco dobre do generowania liczb pseudolosowych.

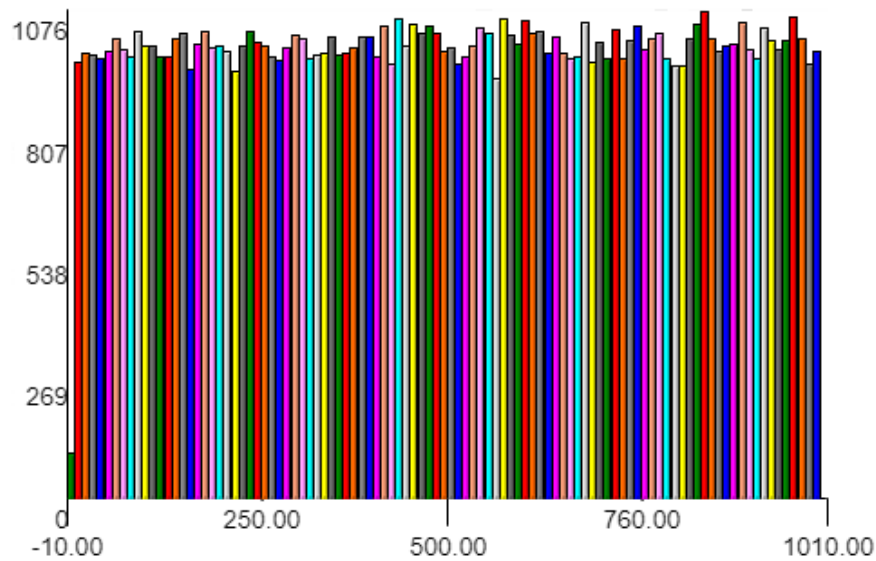
5. Opis Plików projektu

Plik main.cpp zawiera cały kod funkcji generatorów i testów z nimi związanych.

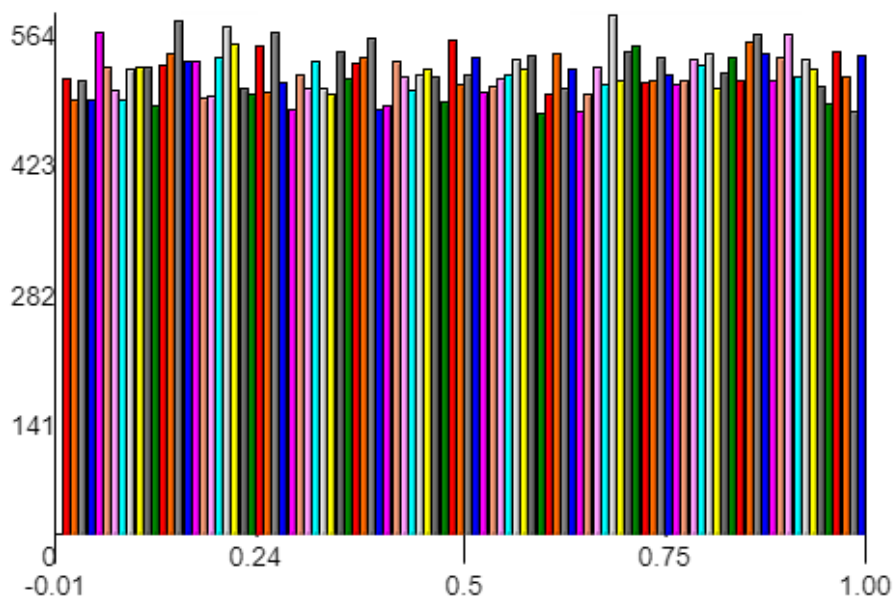
Plik testy.txt zawiera wyniki paru testów przeprowadzonych na generatorach.

6. Histogramy dla danych wszystkich rozkładów

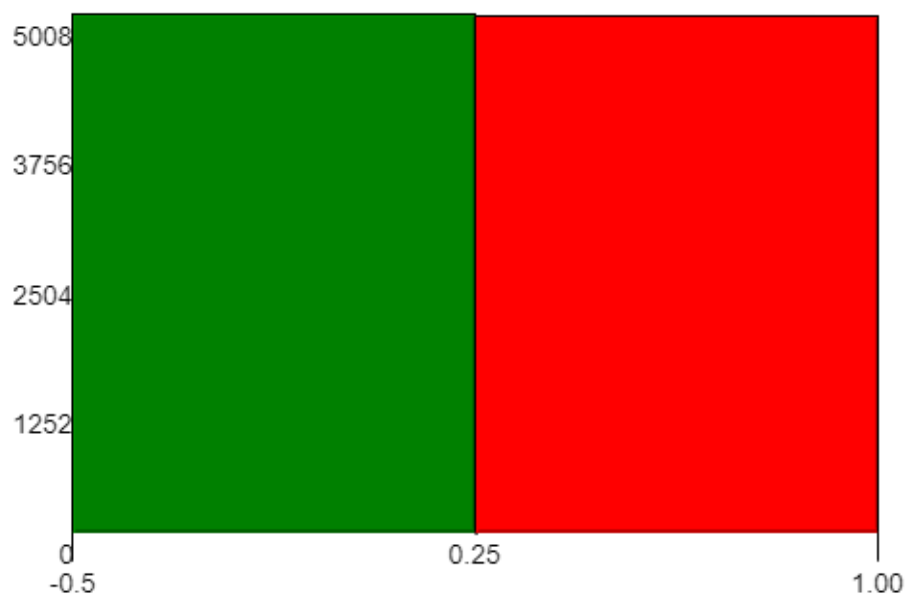
Liczby całkowite (0, 1000)



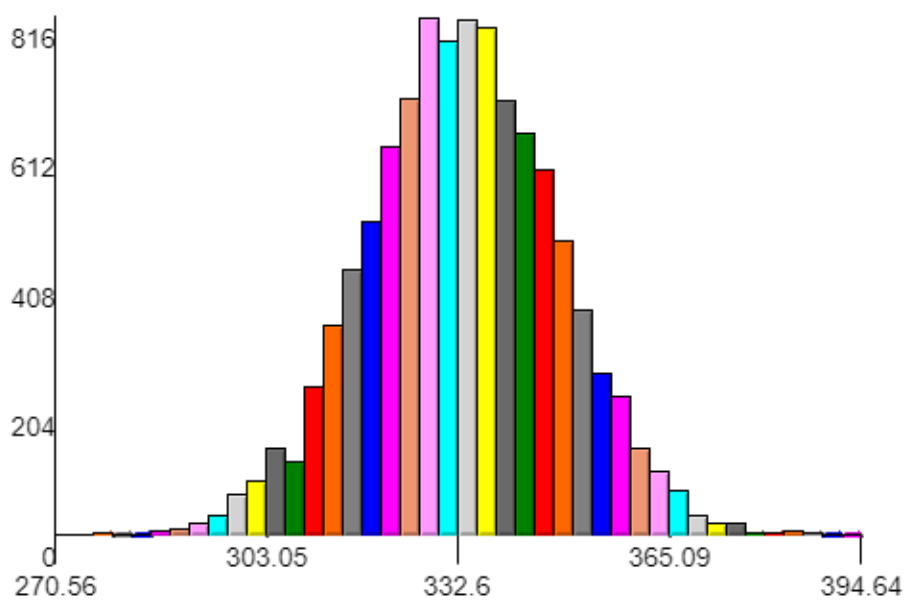
Rozkład jednostajny (0, 1)



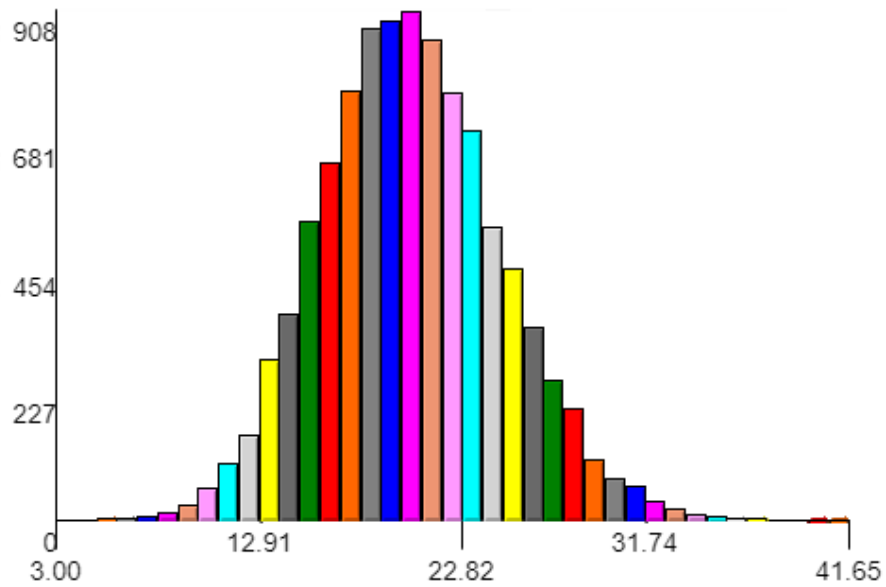
Rozkład Bernoulliego



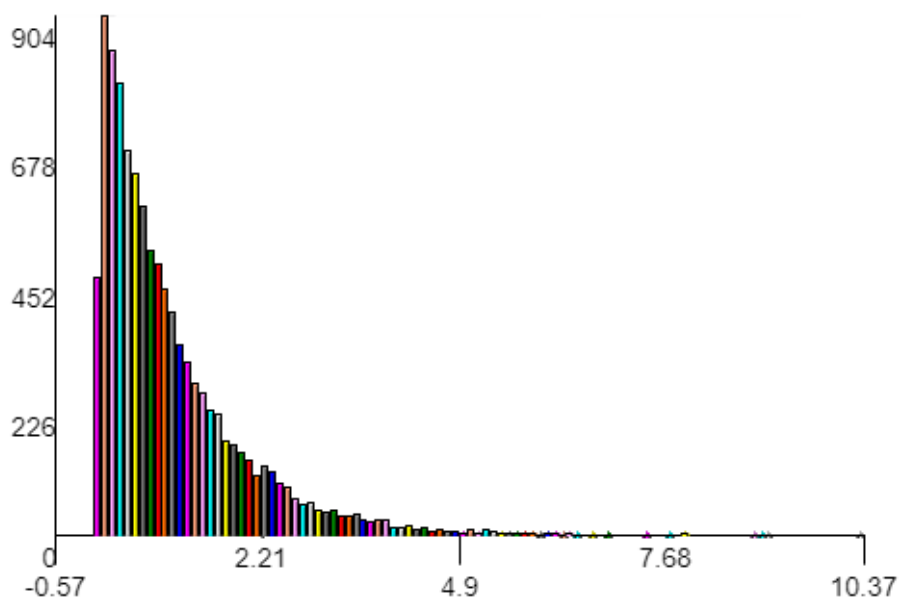
Rozkład dwumianowy (1000, 1/3)



Rozkład Poissona ($\lambda = 20$)



Rozkład wykładniczy ($\lambda = 1$)



Rozkład normalny (0, 1)

