# Proiect: Simulare Monte Carlo pentru estimarea numărului mediu de bilete necesare pentru a câștiga la un joc de loterie

Ancuta Theodor-Constantin(251)

Moloceniuc Albert-Ionut(252)

Voica Stefan-Alexandru(251)

**CARE ESTE PROBABILITATEA MAI MARE???**





## 1.Alegeți o problemă relevantă (exemple: estimarea valorilor unor arii/volume dificile, determinarea unei probabilități în probleme de jocuri de noroc, simularea unui proces fizic, economic, sau social, etc.) Justificați alegerea problemei și descrieți formularea matematică a acesteia.

Am ales problema **estimării numărului mediu de bilete necesare pentru a obține o anumită potrivire de numere într-un joc de tip loterie 6/49**, utilizând simulări Monte Carlo. Spre deosebire de problema clasică a câștigării jackpot-ului (6/6), în acest proiect am modificat condițiile de joc pentru a analiza șansele de a obține **o potrivire de 1, 2 sau 3 numere din 6**.

1. **Aplicații practice și îmbunătățirea înțelegerii probabilităților**:

* Modificarea jocului (de la 6/6 la potriviri parțiale, cum ar fi 1/6, 2/6 sau 3/6) face simularea mai realistă și mai accesibilă, având în vedere că șansele de câștig parțial sunt semnificativ mai mari. Această ajustare reflectă mai bine structura reală a premiilor din jocurile de loterie, unde se acordă câștiguri și pentru potriviri parțiale.

2. **Rulare mai eficientă**:

* În problema clasică, numărul de bilete necesare pentru a câștiga jackpot-ul tinde să fie extrem de mare (E[X]≈13,983,816), ceea ce face ca simularea să fie greu de executat eficient. Prin reducerea cerințelor de potrivire (1/6, 2/6, 3/6), rezultatele sunt obținute mai rapid, iar numărul mediu de bilete necesare tinde să fie rezonabil.

## Explicați algoritmul de tip Monte Carlo folosit pentru generarea punctelor/simulărilor și modul în care acestea sunt utilizate pentru a aproxima soluția.

În viața reală, selecția numerelor dintr-un joc de tip loterie nu este complet aleatorie. Mulți jucători aleg numere bazate pe factori personali, precum date de naștere, numere norocoase sau alte preferințe. Astfel, distribuția numerelor alese nu este uniformă.

În acest proiect, ne-am inspirat din **statisticile publicate pe site-ul** [**lotteryextreme.com**](https://www.lotteryextreme.com/Romania/loto_649-statistics(3)), care oferă informații despre frecvențele de apariție ale numerelor câștigătoare din loteria 6/49 din România. Am utilizat aceste date pentru a introduce o **distribuție biased** în selecția numerelor de pe biletele generate, replicând astfel comportamentul real al jucătorilor.

Algoritmul Monte Carlo este o metodă stocastică utilizată pentru a rezolva probleme complexe prin generarea aleatorie de eșantioane și aplicarea de reguli probabilistice pentru a estima soluția. În contextul acestui proiect, algoritmul simulează extragerile de loterie prin generarea unui număr mare de bilete și evaluarea numărului de potriviri pentru fiecare bilet în raport cu biletul câștigător. Metoda se bazează pe ideea că, pe măsură ce numărul de simulări crește, rezultatele medii obținute converg către valoarea reală a probabilității, conform legii numerelor mari. Astfel, metoda Monte Carlo oferă o aproximare robustă a soluției, chiar și în cazul unor distribuții neuniforme sau a unor spații de probabilitate mari, cum este cazul loteriei 6/49.

## Deduceți teoretic cum metoda Monte Carlo poate aproxima valoarea dorită. Folosiți inegalități stochastice (cum ar fi inegalitatea Cebâșev-Hoeffding sau alte inegalități relevante) sau Teorema Limită Centrală pentru a estima numărul de simulări necesare pentru o anumită marjă de eroare și nivel de încredere.

În cadrul proiectului nostru, dorim să determinăm de câte simulări avem nevoie pentru a asigura o marjă de eroare specifică, cu un anumit nivel de încredere, atunci când aproximăm numărul mediu de bilete necesar pentru potriviri în loteria 6/49. Pentru acest scop, vom aplica **inegalitatea Chernoff-Hoeffding** pentru a determina numărul minim de simulări necesar.

Inegalitatea Chernoff-Hoeffding ne oferă o limită pentru probabilitatea ca diferența dintre media teoretică μ și media empirică X să depășească un prag z>0.

Această inegalitate ne spune **câte simulări sunt necesare pentru o anumită precizie și încredere**. În exemplul de mai sus, cu **117.260 simulări**, vom avea o estimare bună cu o eroare de cel mult 0.08 și o probabilitate de 95%.

[Paginile 1-2 din Demonstratii]

Următorul pas este să se determine dimensiunea sample-ului, adică numărul de bilete generate pentru fiecare simulare, astfel încât aceste condiții să fie respectate.

[Pagina 3 din Demonstratii]

## Implementați simularea Monte Carlo

**Descriere scurtă a funcțiilor:**

**1. pick\_number()**

Această funcție generează un număr aleatoriu între 1 și 49, conform unei distribuții biasate bazate pe frecvențele istorice ale numerelor câștigătoare. Se folosește o probabilitate cumulativă pentru selecție.

**2. simulari(nr\_simulari, nr\_bilete, numar\_minim\_potriviri)**

Această funcție rulează mai multe simulări Monte Carlo. Pentru fiecare simulare:

* Generează un bilet câștigător.
* Creează bilete jucate și verifică câte dintre acestea ating un număr minim de potriviri cu biletul câștigător. Returnează o listă cu rezultatele fiecărei simulări.

**3. generare\_histograma(nr\_simulari, nr\_bilete, numar\_minim\_potriviri)**

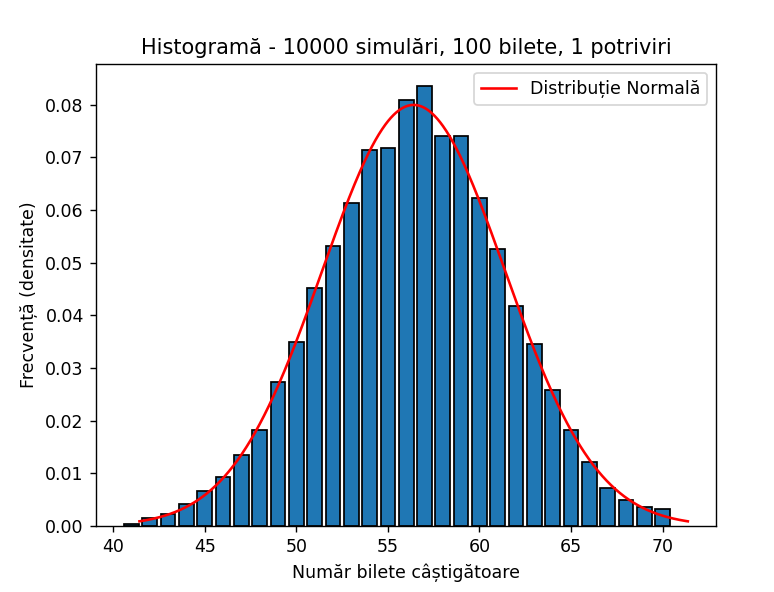
Această funcție creează o histogramă pentru a vizualiza distribuția numărului de bilete câștigătoare în toate simulările. Include suprapunerea unei distribuții normale pentru a analiza convergența rezultatelor.

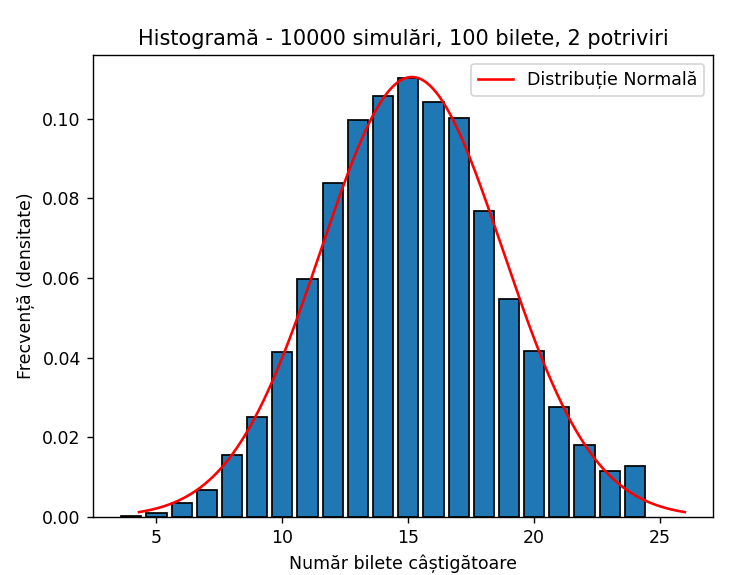
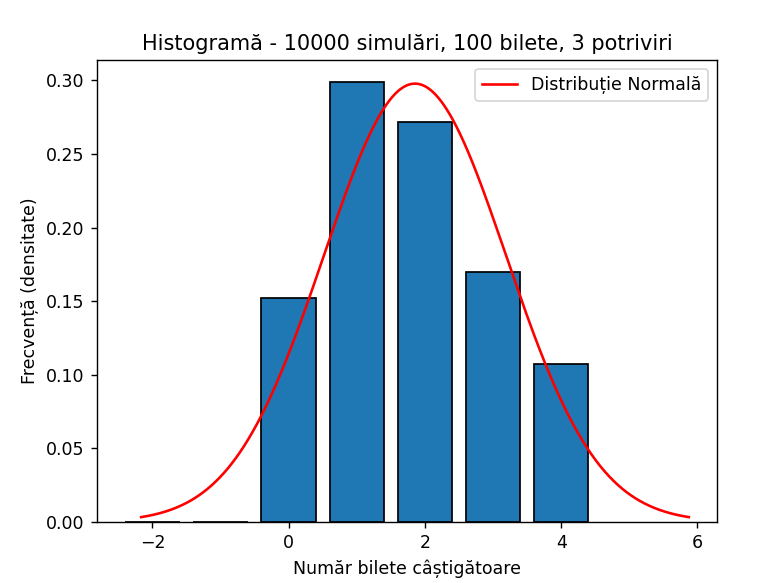
Codul utilizează trei scenarii diferite pentru simulări:

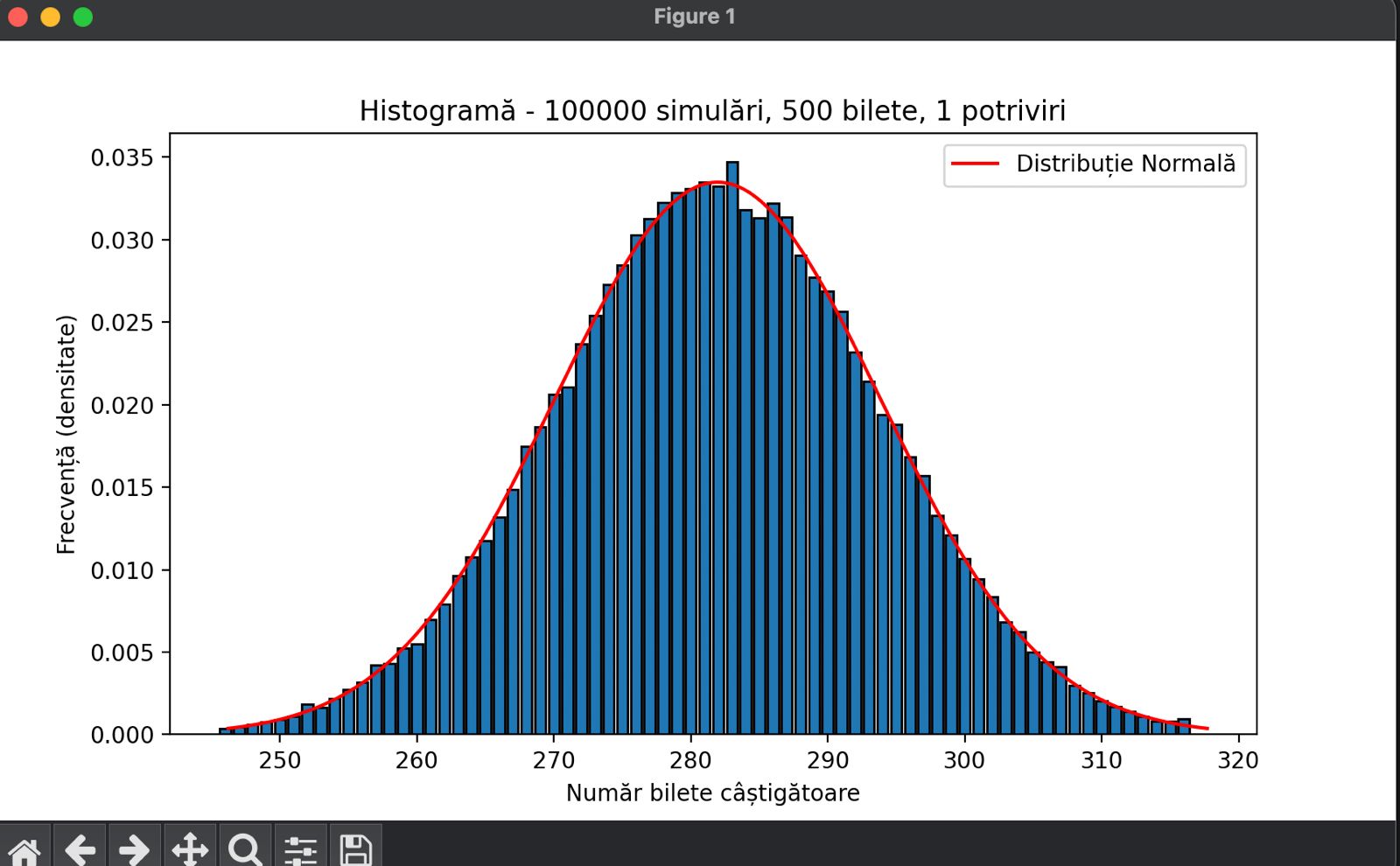
1. **Simulare 1**: 10.000 simulări, 100 bilete
2. **Simulare 2**: 100.000 simulări, 500 bilete
3. **Simulare 3**: 120.000 simulări, 1.000 bilete.

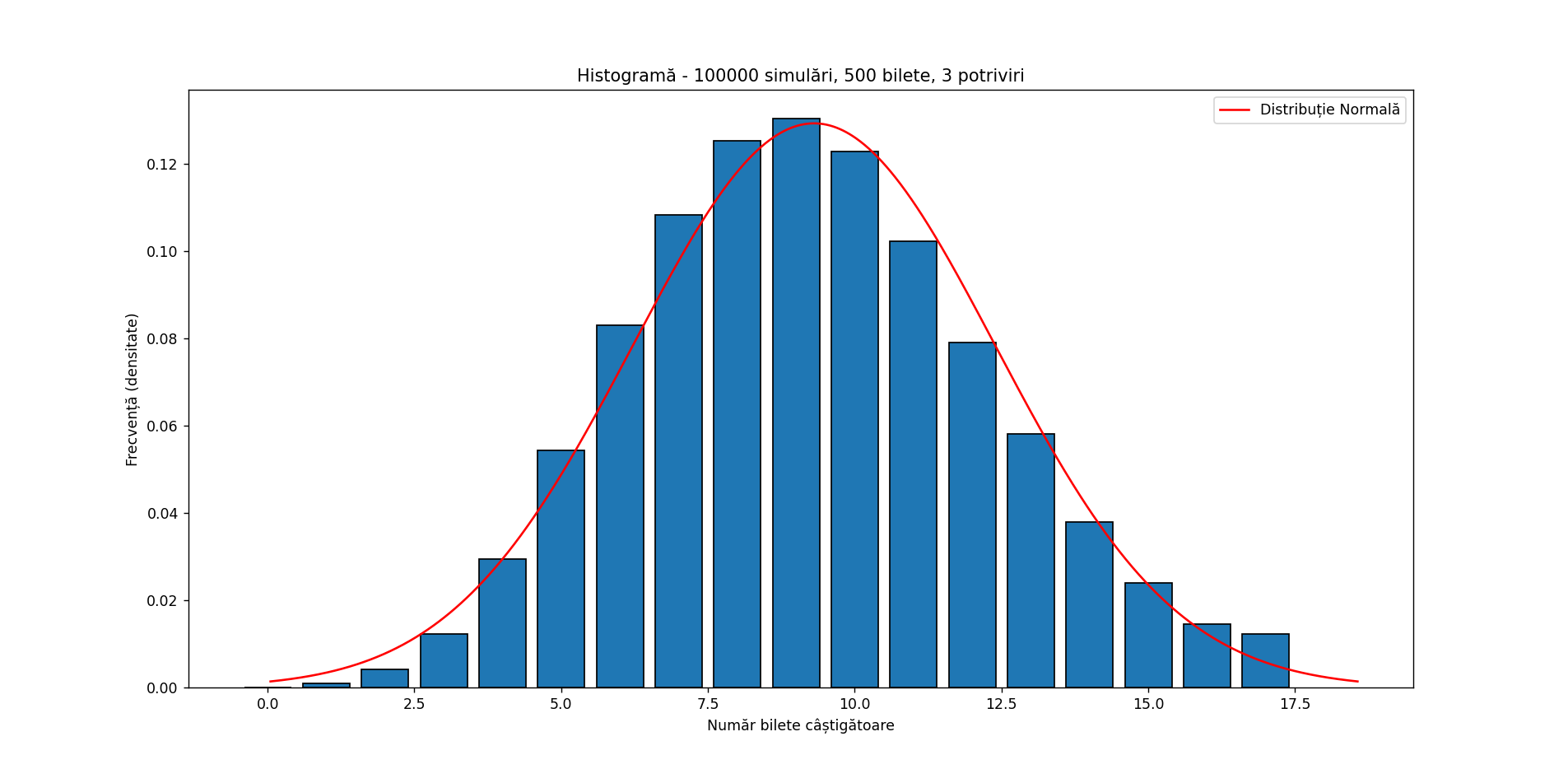
**Pentru simulări mici** (ex. simulare 1), distribuția poate avea o mai mare dispersie și valori extreme, ceea ce duce la un interval de eroare mai larg.

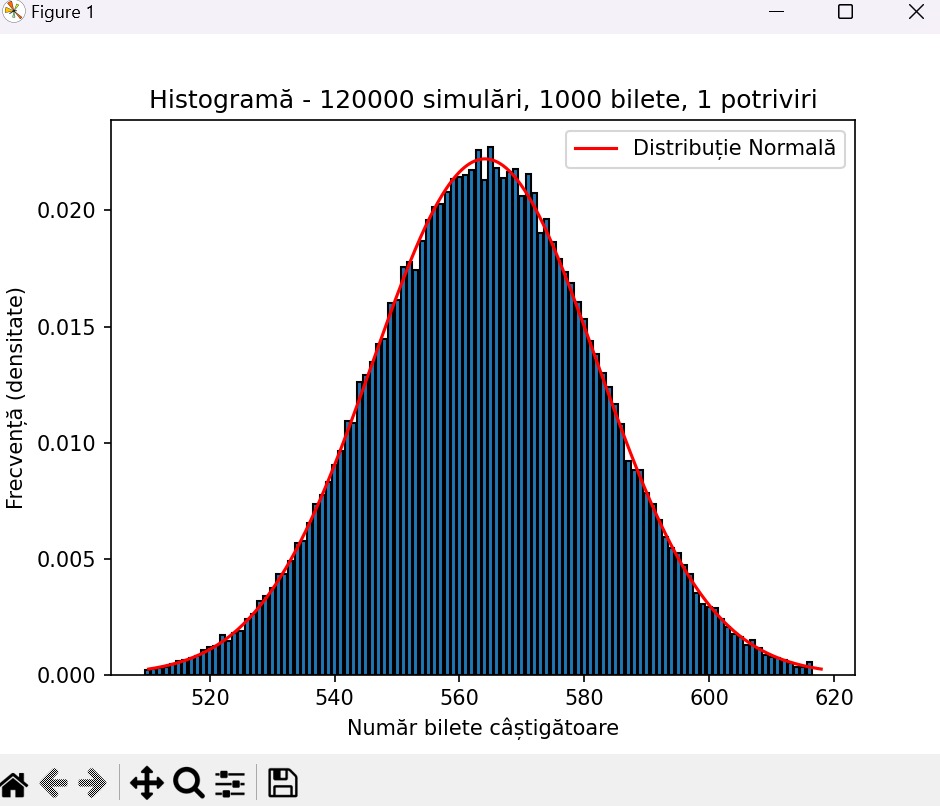
**Pentru simulări mari** (ex. simulare 3), datele converg mai bine către media teoretică, iar eroarea scade semnificativ, demonstrând stabilitatea rezultatelor. De asmenea, am rulat o simulare pt n = 1.000.000 și 1000 de bilete cu o potrivire.

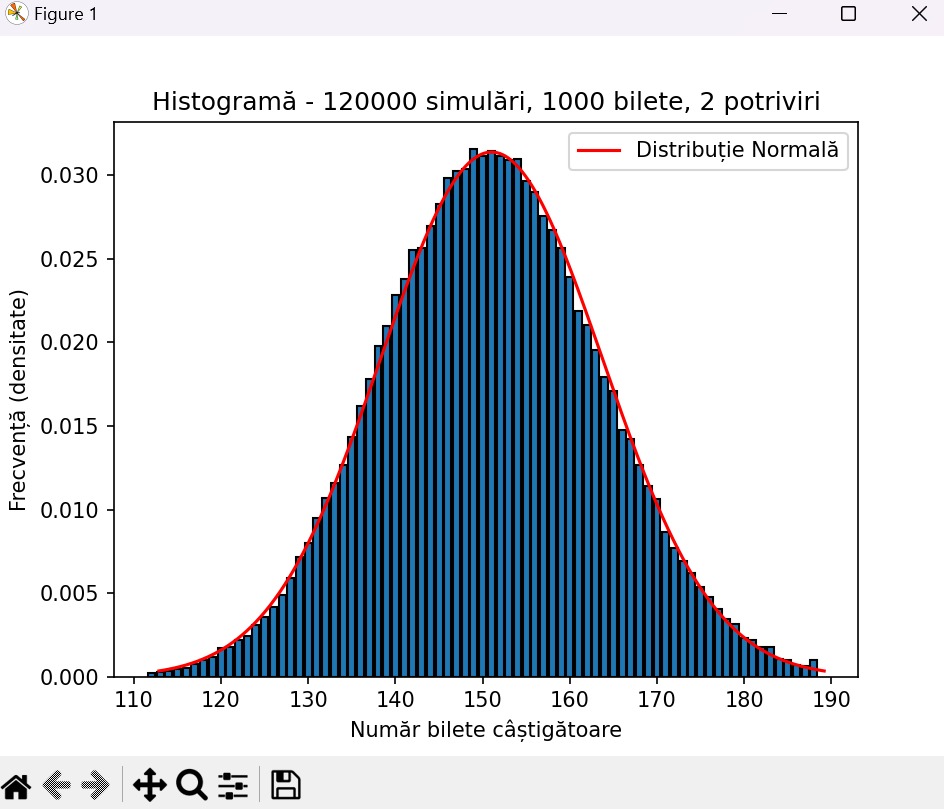


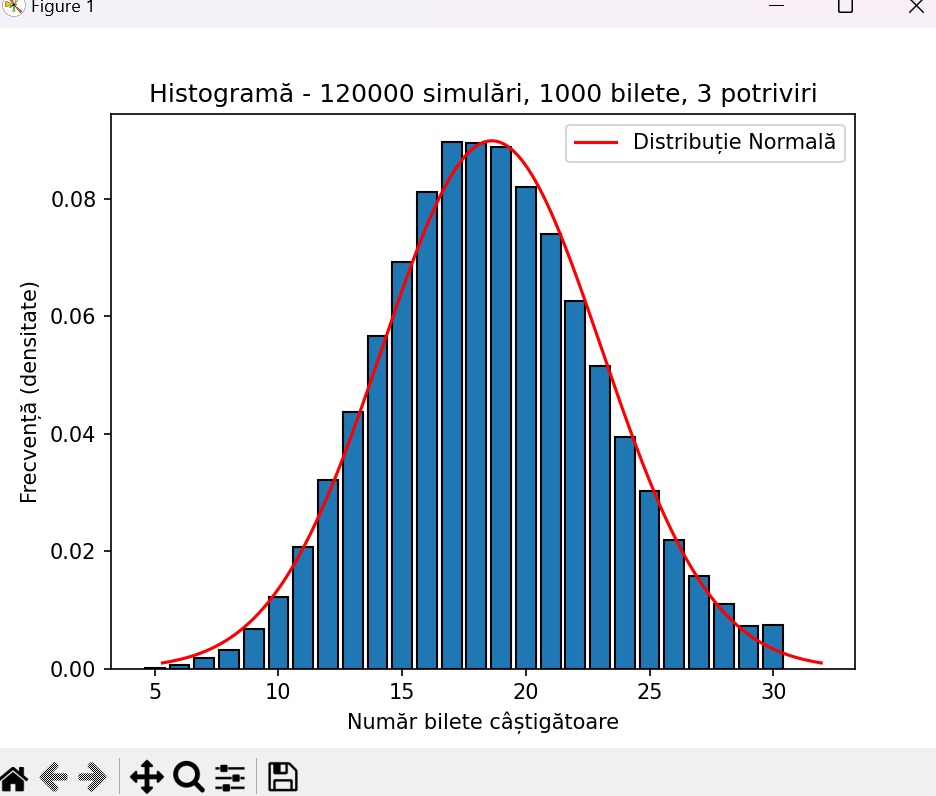


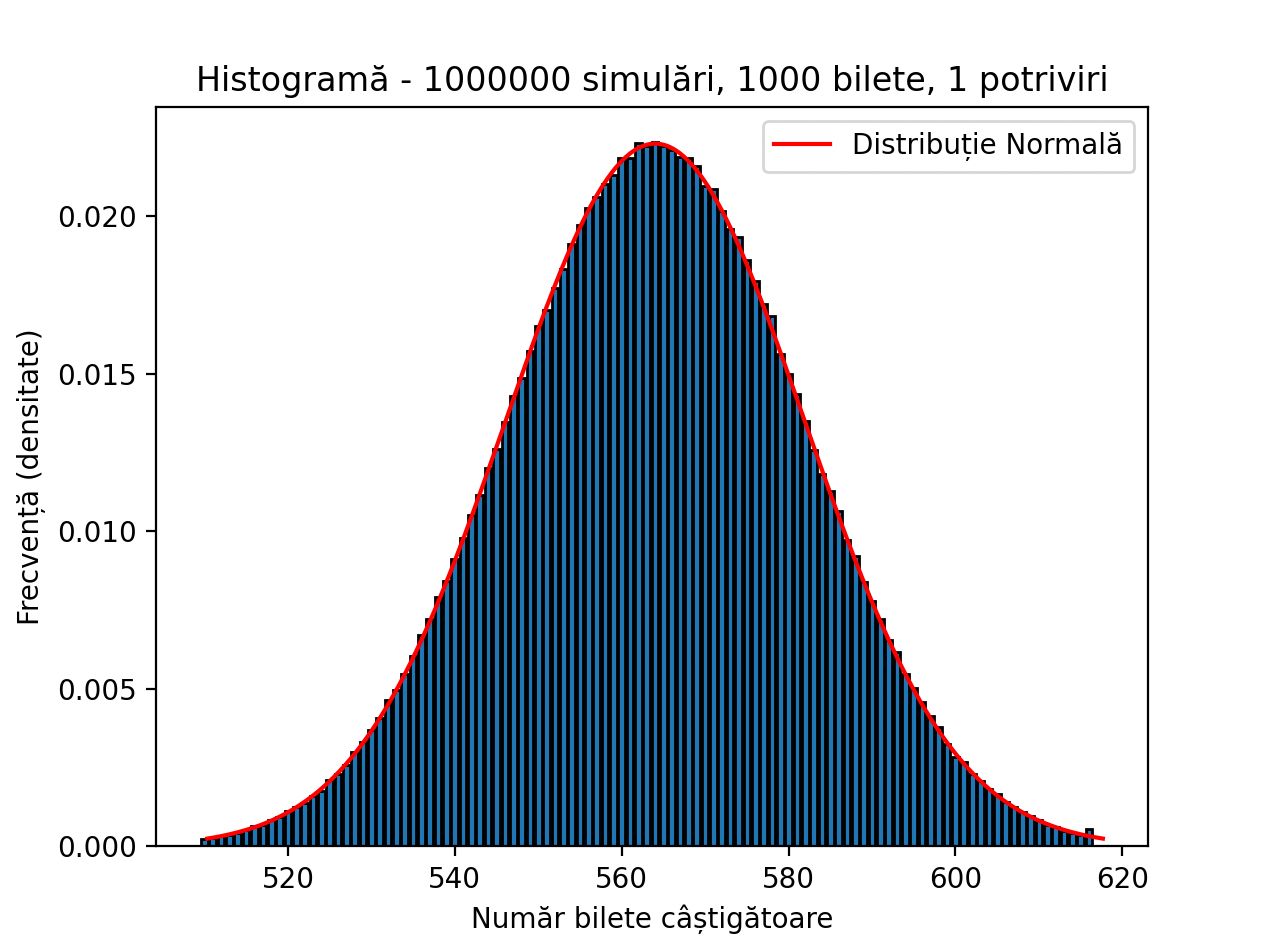












**Bibliografie**

**“Probability and Statistics for Computer Scientists”** de **Michael Baron**

<https://www.lotteryextreme.com/Romania/loto_649-statistics(3)>

<https://github.com/kirudang/Monte_Carlo_simulation_R>

<https://jmsallan.netlify.app/blog/2024-10-04-analyzing-a-lottery-with-monte-carlo-simulation/>

<https://www.investopedia.com/terms/m/montecarlosimulation.asp>

<https://aws.amazon.com/what-is/monte-carlo-simulation/>