Laboratorio di Calcolo per Fisici, Sesta esercitazione

Canale D-K, Docente: Lilia Boeri

Lo scopo della sesta esercitazione di laboratorio è di fare pratica con le istruzioni di input/output da file e la scrittura modulare di programmi tramite funzioni.

La distribuzione binomiale, o distribuzione di Bernoulli, è una distribuzione di probabilità discreta che descrive la probabilità che in n esperimenti si verifichi k volte un evento con probabilità individuale p. La distribuzione è data da:

$$P(k) = \frac{n!}{\nu!(n-k)!} p^k (1-p)^{n-k},$$

(in questa formula $n!$ indica il *fattoriale* di n). Trattandosi di una distribuzione discreta, si ha: $\sum_{k=0}^{N} P(k) = 1$ (*condizione di normalizzazione*).
Esempi di processi descritti dalla distribuzione binomiale sono il lancio di una moneta ($p_{testa} = p_{croce} = \frac{1}{2}$), il lancio di un dado a 6 facce, l'estrazione di palline o bossoli da un'urna, l'estrazione (ripetuta) di una carta da un mazzo di carte, etc.
Consideriamo per esempio il lancio di un dado a 6 facce; se definiamo come evento favorevole (*successo*) l'apparire di una determinata faccia, per esempio il numero 3, la probabilità p che si verifichi il singolo evento sarà $p = \frac{1}{6}$; la distribuzione binomiale descriverà la probabilità che in n lanci successivi di un singolo dado a 6 facce appaia esattamente k volte la faccia con il numero 3.

► Prima parte:

1. Creare un programma `bernoulli.c` che calcoli e scriva su un file `bernoulli.dat` la distribuzione di Bernoulli per valori arbitrari di n e p. Il programma dovrà contenere almeno una chiamata a una funzione esterna, che implementi la funzione fattoriale.

2. Il file di output deve contenere, su due colonne:
   0 $P(0)$
   k $P(k)$
   n $P(n)$

3. Dopo aver verificato che il programma funzioni correttamente e che la $P(k)$ soddisfi la condizione di normalizzazione, fissare $p = \frac{1}{6}$ e graficare con `python` la funzione $P(k)$ per i seguenti valori di n: $n = 2$; $n = 4$; $n = 10$; $n = 20$. Che cosa si nota all'aumentare di n? Perchè? Salvate i dati usati per le figure su diversi file `bernoulli_n.dat` e i grafici su file immagine `bernoulli_n.gif`.

► Seconda parte:

Creare un secondo programma `lanci.c` che simuli l'evento descritto dalla distribuzione di Bernoulli. In particolare, il programma deve simulare una serie N_{lanci} di lanci di n dadi a 6 facce, e per ciascuno di questi lanci registrare il numero di volte k in cui compare la faccia numero 3 del dado. L'istogramma delle occorrenze k, opportunamente rinormalizzato, tenderà alla distribuzione di Bernoulli se il numero dei lanci N_{lanci} è sufficientemente grande. Il programma `lanci.c` deve:

1. Chiedere all'utente di inserire il numero n di dadi lanciati per ciascun lancio e il numero di lanci N_{lanci};

2. Chiamare una funzione esterna che simuli il lancio di un dado a 6 facce utilizzando un generatore di numeri casuali;

3. Costruire un istogramma che contenga il numero di successi k per ciascun lancio.

4. Riscalare opportunamente l'istogramma ottenuto al punto precedente in modo che la *condizione di normalizzazione* della distribuzione sia soddisfatta.

5. Salvare su un file l'istogramma riscalato. Per il punto successivo, può essere utile aprire il file in modalità aggiungi (*append*), in modo da non sovrascrivere i dati a ogni esecuzione del file.

► **Terza parte**

Confrontare (graficamente) i dati generati dai due programmi per gli stessi valori di n e k. Qual è il numero minimo di lanci da usare nel secondo programma per ottenere un campionamento ragionevole della distribuzione di Bernoulli? Se si esegue il secondo programma più volte per uno stesso numero di lanci N_{lanci}, come cambia la distribuzione? Che cosa si può fare per diminuire le *fluttuazioni* nel risultato? Scrivete le vostre osservazioni e risposte su un file `risposte.c`.