

Laboratorio di Calcolo I

Applicazioni : Metodo Monte Carlo

Monte Carlo

- Il metodo di Monte Carlo è un metodo per la risoluzione numerica di problemi matematici che utilizza **numeri casuali**.
- Si applica sia a problemi che coinvolgano processi statistici che nella risoluzione di diversi problemi matematici quali la determinazione di un'area (integrale).

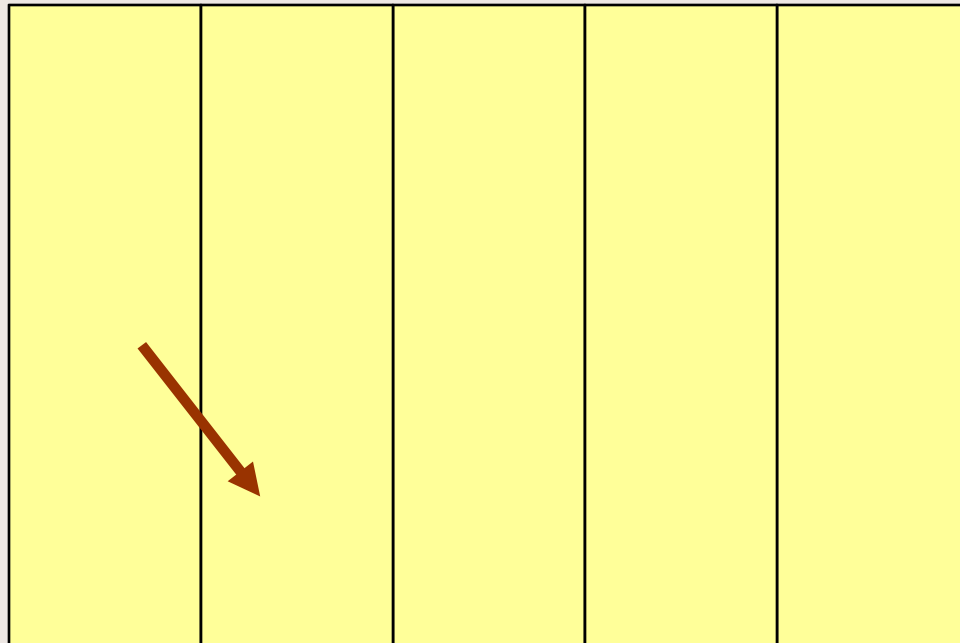
Ago di Buffon

- Il primo Monte Carlo data della fine del XVIII secolo e serviva per la determinazione statistica del numero π :
- Si lascia cadere un ago di lunghezza d su una superficie sulla quale si siano tracciate righe parallele a distanza d l'una dall'altra...

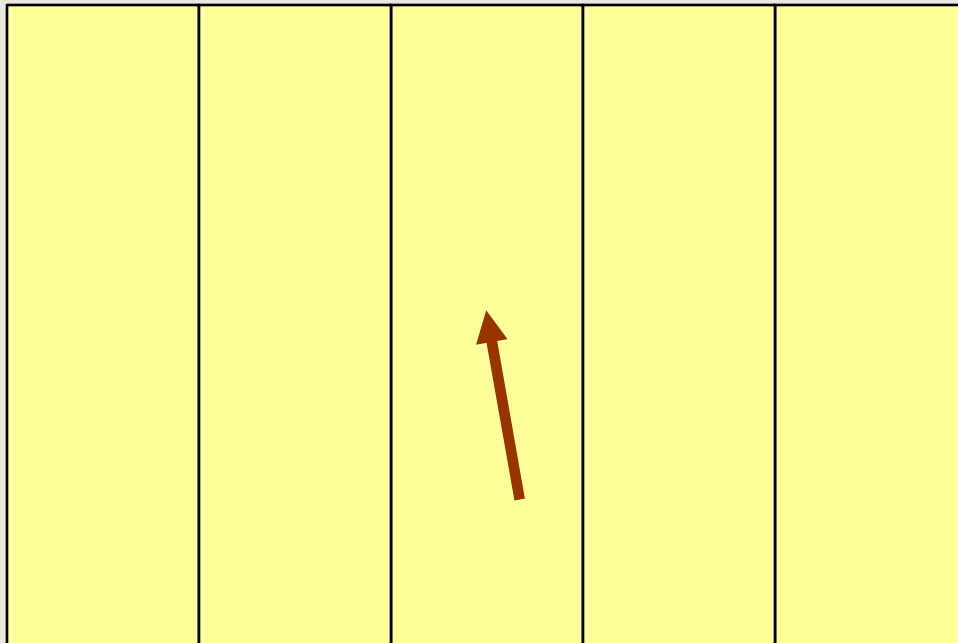
Ago di Buffon

--	--	--	--	--

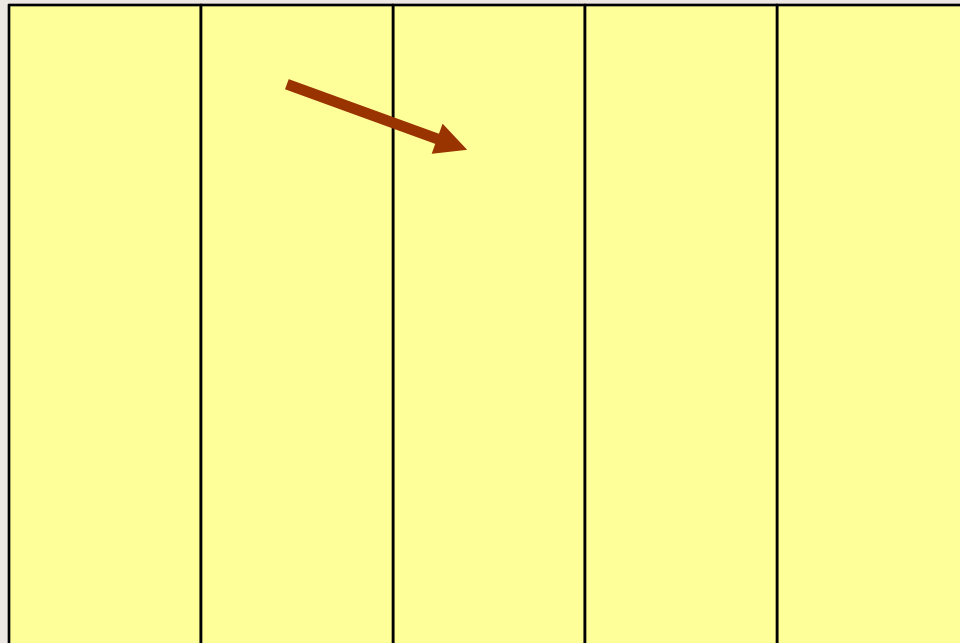
Ago di Buffon



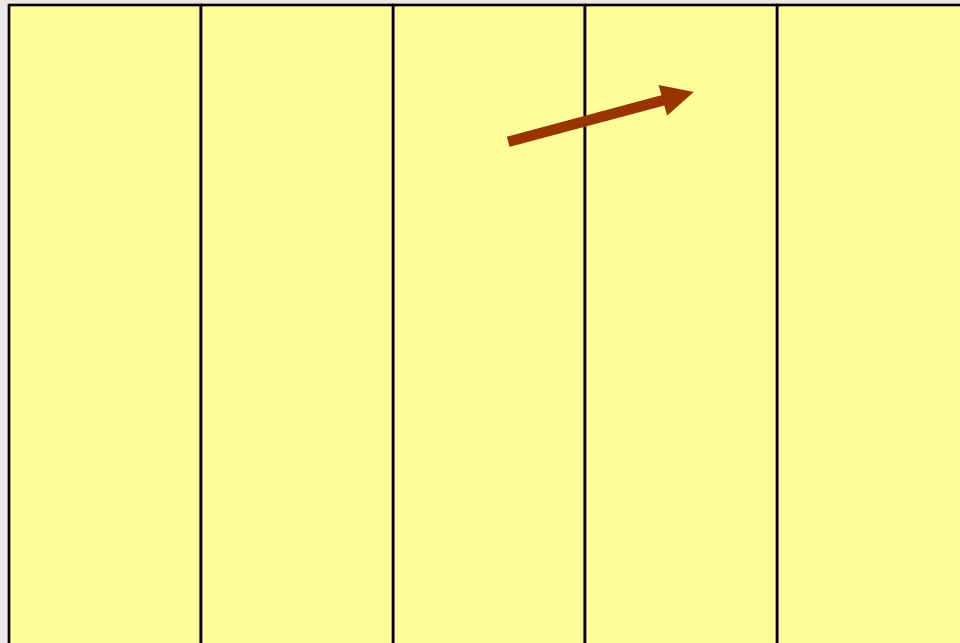
Ago di Buffon



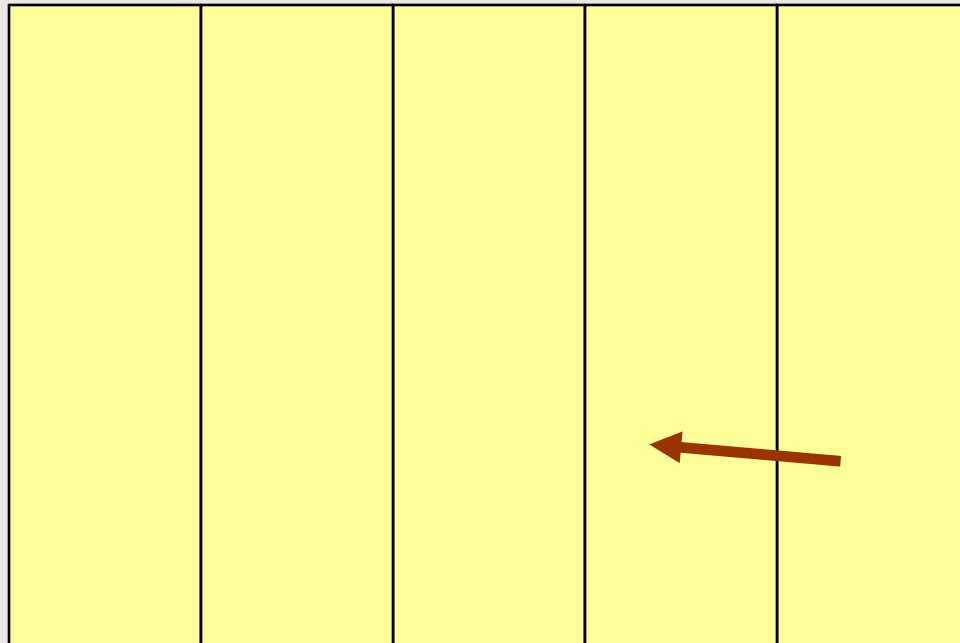
Ago di Buffon



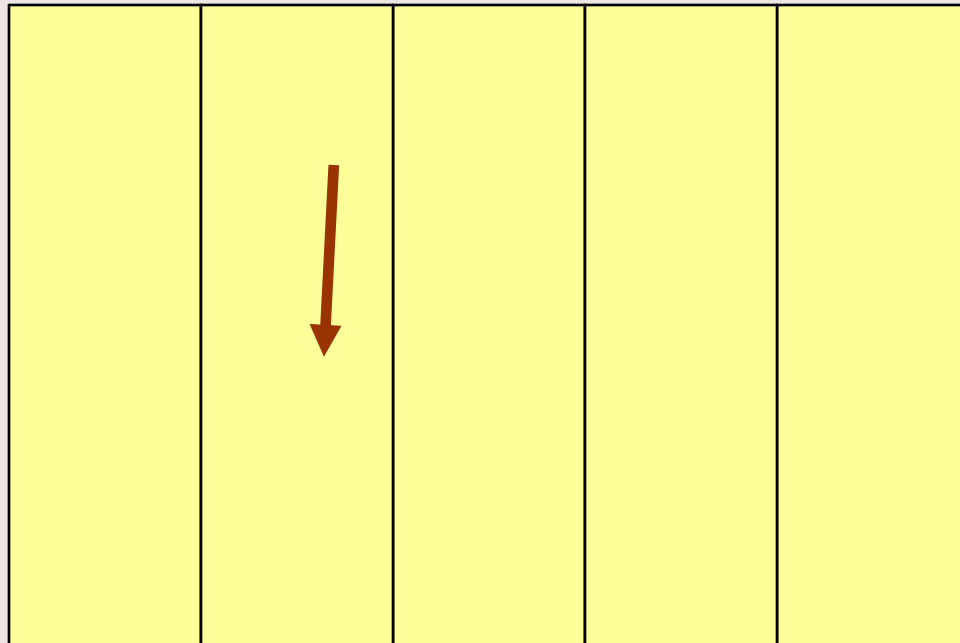
Ago di Buffon



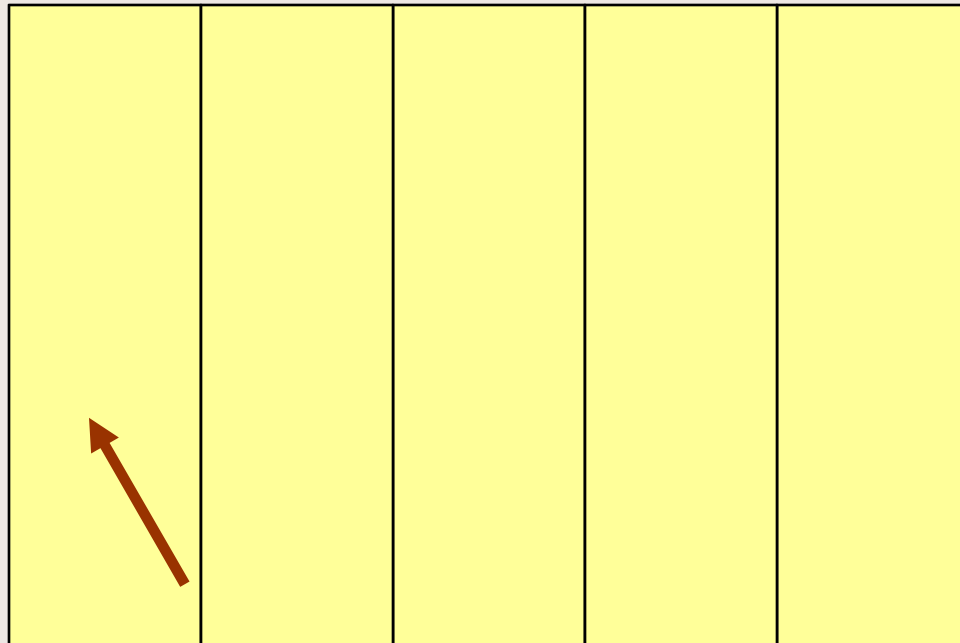
Ago di Buffon



Ago di Buffon



Ago di Buffon



Ago di Buffon

- La probabilità che l'ago incroci una linea è data da $P = 2/\pi$
- Se l'ago viene lanciato N volte, indicando con N_x il numero di volte che l'ago incrocia una linea

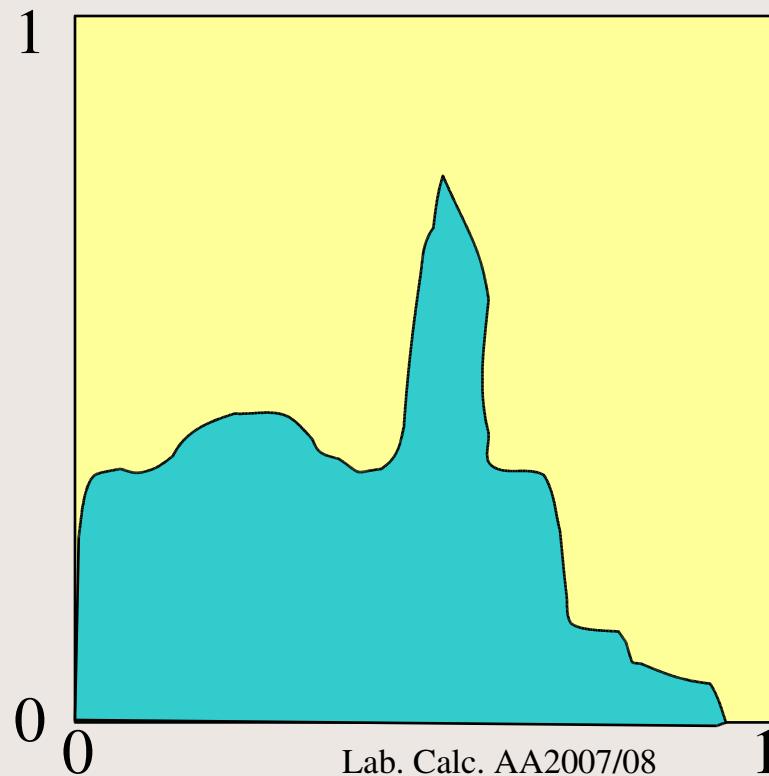
N_x/N tende a P all'aumentare di N

e quindi

$2 N/N_x$ tende a π

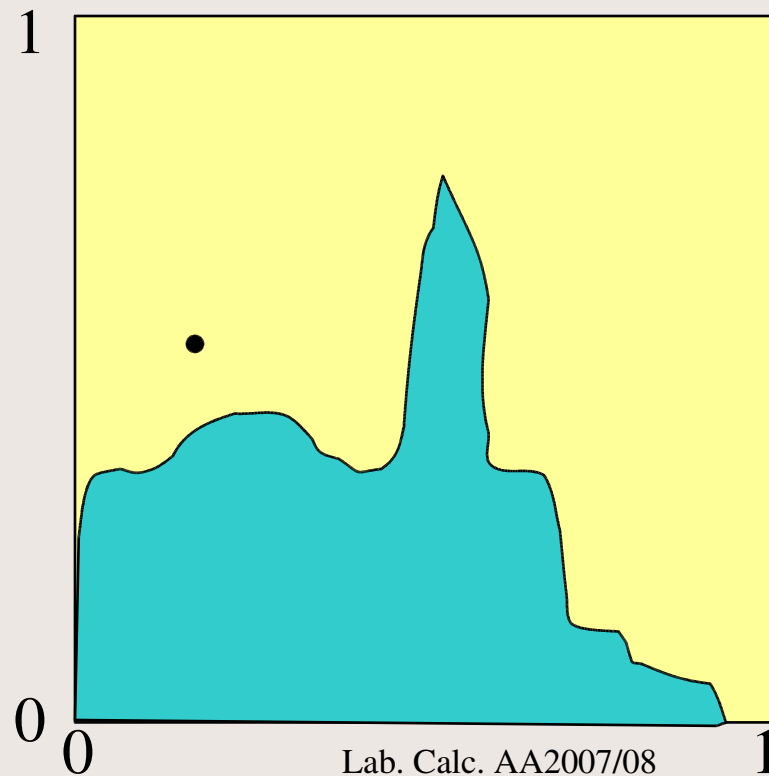
Misura di un'area

- Sia una generica superficie A completamente contenuta in un quadrato di lato unitario



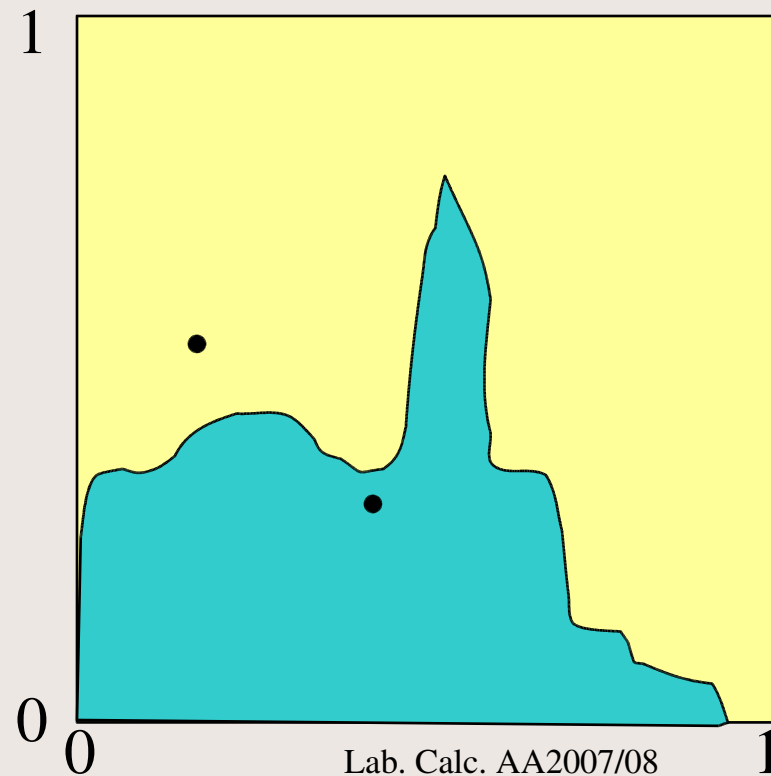
Misura di un'area

- estraiamo una coppia di numeri casuali (x,y) tra loro indipendenti e compresi tra 0 e 1 e li rappresentiamo come un punto nel piano xy

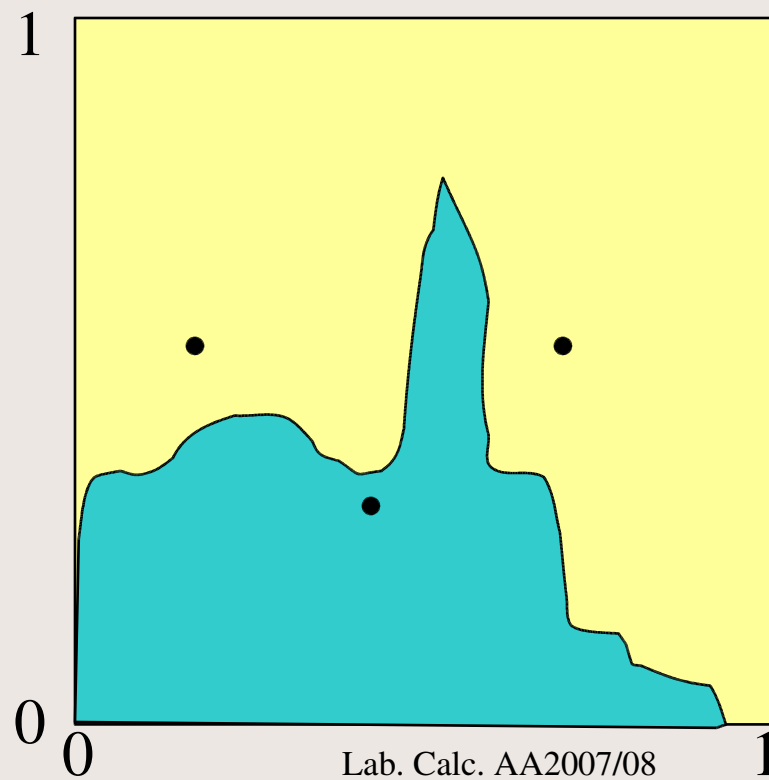


Misura di un'area

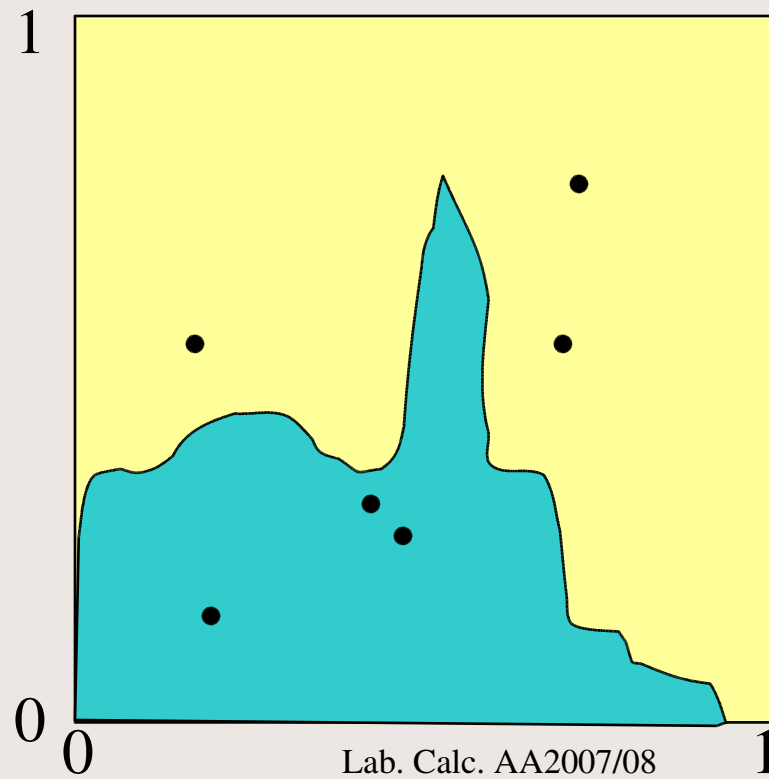
- Procediamo estraendo altre coppie



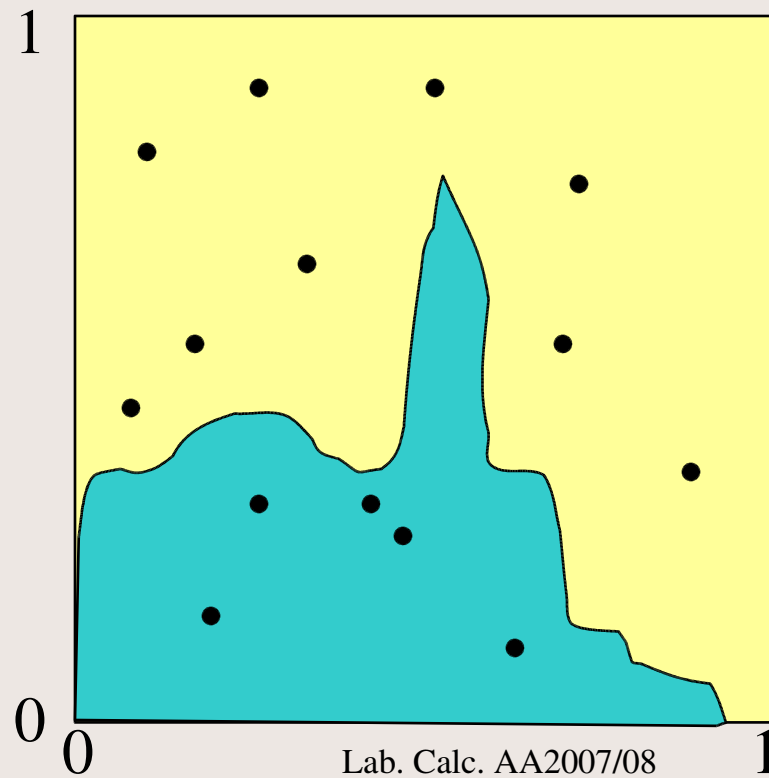
Misura di un'area



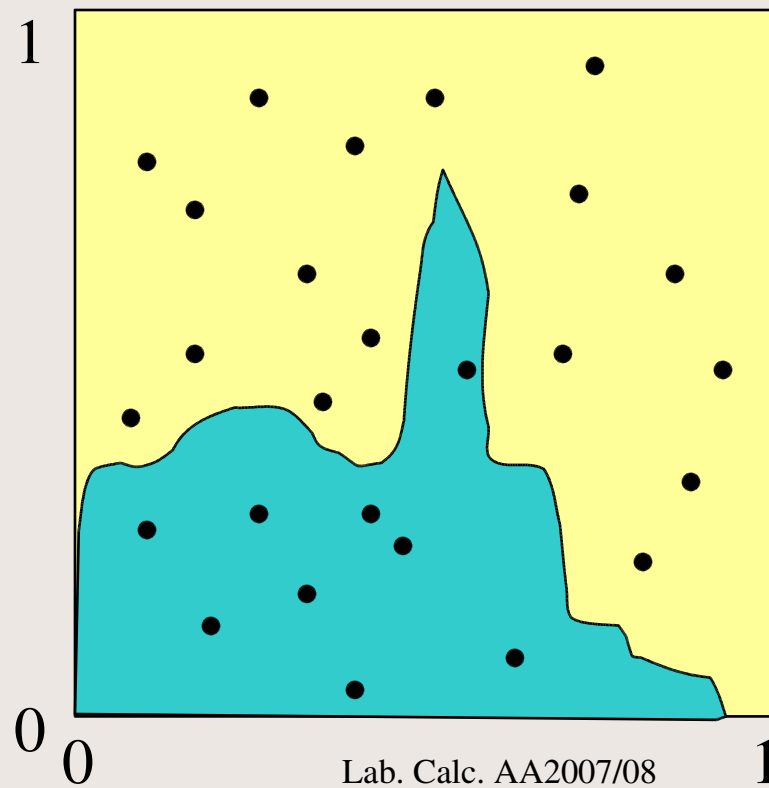
Misura di un'area



Misura di un'area

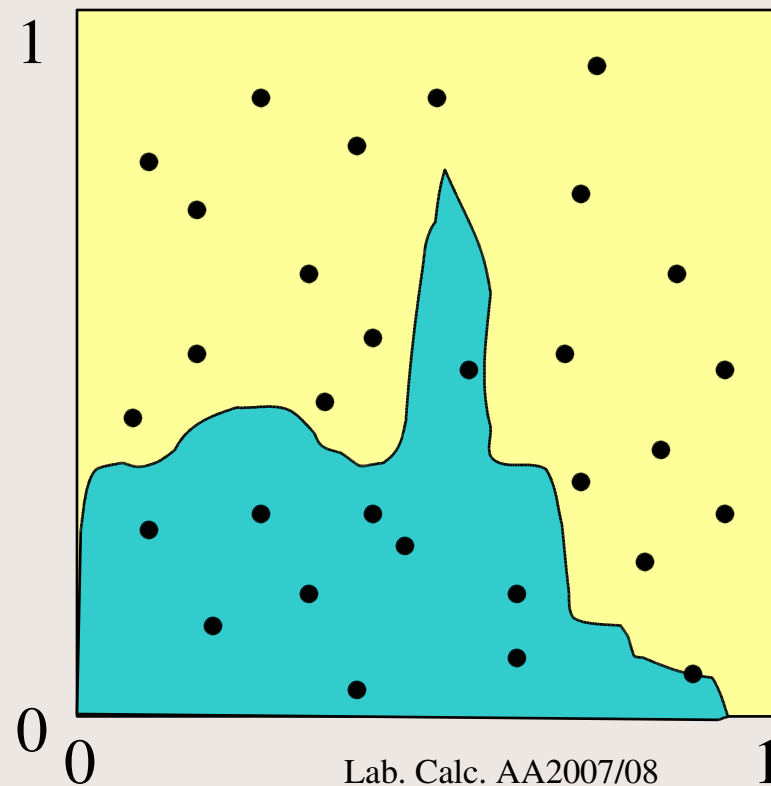


Misura di un'area



Misura di un'area

- La densità di punti geometrici estratti sarà uniforme...



Misura di un'area

- Quindi **la probabilità** che il punto (x,y) estratto capiti all'interno dell'area che vogliamo misurare è esattamente uguale all'area stessa.
- Se **estraiamo** N coppie di numeri e N' è il numero di punti che cadono all'interno della figura il rapporto N'/N tende al crescere di N all'area della figura.

Sequenze di numeri casuali

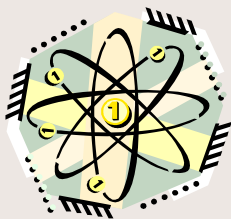
- Sequenze di numeri casuali possono essere generate



- Utilizzando un sistema caotico, la cui evoluzione è deterministica ma che dipende criticamente dalle condizioni iniziali: il lancio di un dado, l'estrazione di un numero al lotto o con la roulette etc...



- Utilizzando un processo che sia intrinsecamente casuale, come il decadimento radioattivo di un nucleo, il rumore termico, il tempo di arrivo dei raggi cosmici sulla terra...



Sequenze di numeri casuali

- Ma anche
 - Ricorrendo a numeri random raccolti in tavole contenenti vari milioni di numeri, spesso insufficienti per molte applicazioni
 - Ricorrendo a dei generatori di numeri che sono **algoritmi** in grado di produrre sequenze di numeri riproducibili che simulano una successione di numeri casuali. Si parla in questo caso di numeri **pseudo-random**.

Numeri pseudo-random

- **Vantaggi**

- Si ottengono mediante algoritmi semplici
- Occupano poco spazio nella memoria del calcolatore
- La serie è riproducibile e la sua qualità è controllabile

- **Svantaggi**

- Possono esserci delle correlazioni
- La serie di numeri è comunque limitata

Il generatore

- Nelle librerie C associate al compilatore gcc da noi usato troviamo le funzioni `rand()` e `random()`.
- Per sapere come si usano basta dare il comando `man random` o `man rand`
- Per usarle basta includere il file `<stdlib.h>`
- I numeri pseudo-random prodotti sono interi (a 32 bit) distribuiti uniformemente tra 0 e `RAND_MAX` ($2^{31} - 1 = 2147483647$).

generazione di numeri pseudo-random tra 0 e 1 (listato 5.6)

```
#include <stdlib.h>
#include <stdio.h>
#define MAX 1000

main() {
    double x[MAX];
    int i;
    for (i = 0; i < MAX; i++) {
        x[i] = (double)rand()/RAND_MAX;
        /* dividere per RAND_MAX+1. se non si vuole includere 1 */
        printf("%lf\n",x[i]);
    }
}
```

inizializzazione

- l'algoritmo di generazione dei numeri casuali deve essere inizializzato
- `rand()` ha un'inizializzazione di default che fa sì che venga generata sempre la stessa sequenza di numeri partendo da 1
- `srand(int seed)` permette di inizializzarla ad un seme (seed) diverso da 1
- utilizzando sempre lo stesso seme si riottiene sempre la stessa sequenza. Per variarlo in modo arbitrario si può utilizzare il tempo in secondi misurato dall'orologio del computer (`time(0)` o tramite `gettimeofday`).

Distribuzioni non uniformi

- Il calcolatore genera una sequenza di numeri **distribuita uniformemente** su un certo intervallo.
- Per molte applicazioni abbiamo bisogno di numeri distribuiti secondo una **generica funzione di distribuzione** $f(x)$.
- Vediamo alcuni metodi utilizzabili a questo scopo.

Metodo diretto per variabili continue

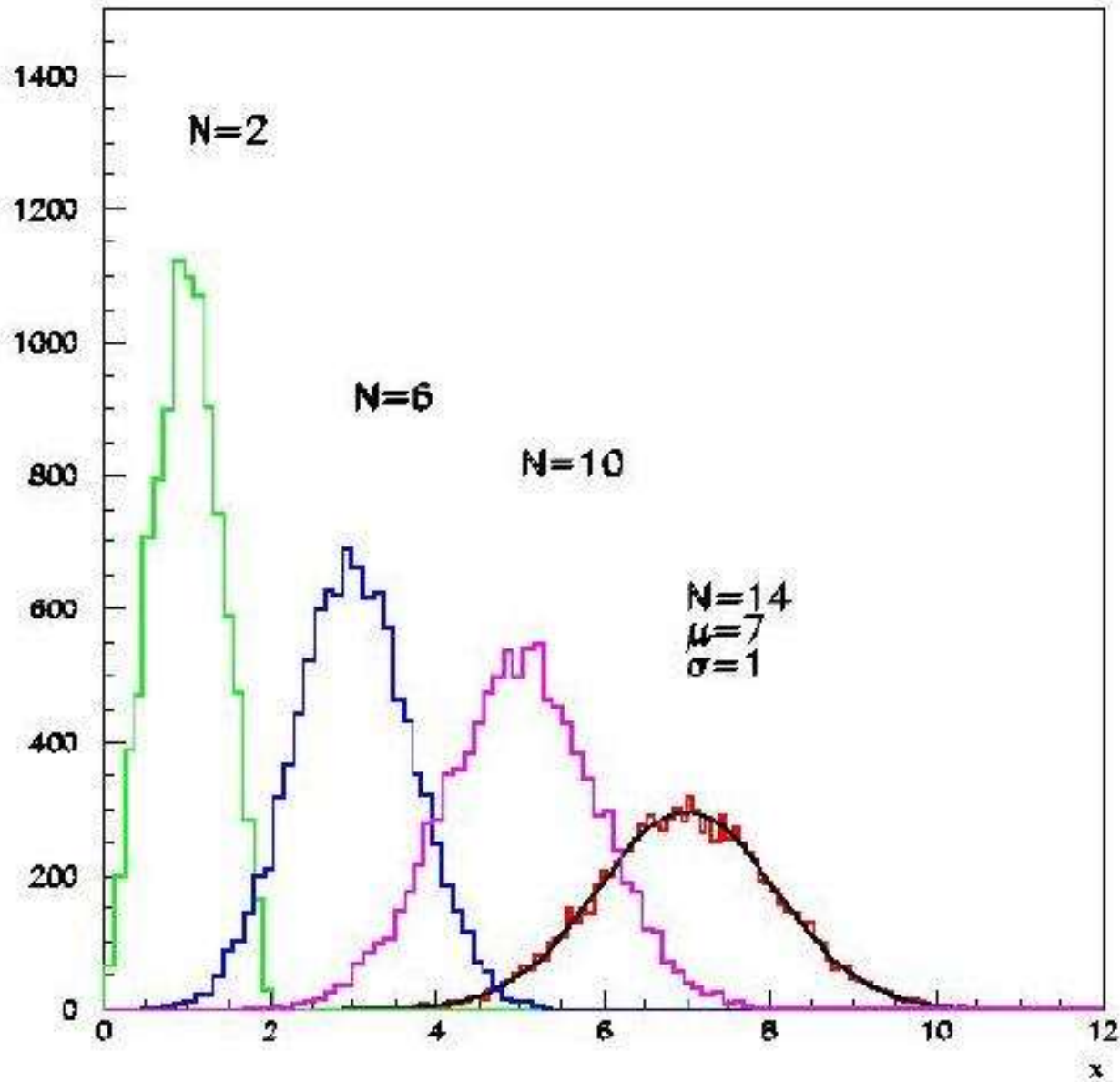
- Per estrarre una variabile aleatoria secondo la **funzione di distribuzione $f(x)$** (normalizzata a 1) osserviamo che **la funzione di probabilità integrata, $F(x)$** (integrale da $-\infty$ a x della $f(x)$), è a sua volta una variabile aleatoria **distribuita uniformemente** tra 0 e 1
- Ne consegue che se si riesce ad esprimere la variabile aleatoria x in funzione della variabile aleatoria F , estraendo quest'ultima tra 0 e 1, si ottiene una variabile x distribuita secondo $f(x)$.

Metodo della reiezione

- Per simulare la funzione di distribuzione $f(x)$ sull'intervallo $[a,b]$, noto il valore massimo assunto dalla funzione nell'intervallo, f_{max} , è sufficiente
 - **Estrarre** una variabile x distribuita uniformemente tra a e b
(se r varia tra 0 e 1 x si ottiene come $x=r*(b-a)+a$)
 - **Estrarre** una variabile y distribuita uniformemente tra 0 e f_{max}
 - Se $y > f(x)$ si ripete l'estrazione della coppia (x,y)
 - Se $y < f(x)$ si accetta il valore di x che è una variabile aleatoria distribuita secondo la $f(x)$

Generazione di numeri gaussiani

- Numeri pseudo-random con funzione di distribuzione gaussiana possono essere anche generati sfruttando il **teorema del limite centrale** che afferma che:
 - *La somma di un numero N di variabili aleatorie con valori medi e varianze finite ha una distribuzione che per N tendente all'infinito tende ad una gaussiana di valor medio "somma dei valori medi" e varianza "somma delle varianze" delle distribuzioni di partenza*
 - Partendo da distribuzioni uniformi già per $N=12$ si ottengono delle gaussiane praticamente perfette



Uso dei numeri pseudo-random

- alcune applicazioni dei numeri pseudo-random
 - Calcolo di integrali
 - Simulazione di eventi discreti
 - Simulazione della risposta di uno strumento
- I numeri pseudo-random sono molto utilizzati anche per simulare fenomeni fisici di natura statistica.

Eventi discreti

- Esempi:
 - Lancio di una moneta (distribuzione di probabilità binomiale con $p=q=0.5$)
 - Lancio di un dado (distribuzione di probabilità binomiale $p=1/6$ $q=5/6$ se si studia il numero di lanci che diano 6)
 - Estrazioni del lotto (distribuzione di probabilità binomiale $p=1/90$ se si studia un numero in particolare)
 - Sondaggio di opinione con due sole risposte, SI e NO (distribuzione binomiale)
 - Sondaggio di opinione con più risposte ammesse (distribuzione multinomiale)

Simulazione di eventi discreti

- Per simulare l'esito di eventi discreti che abbiano probabilità a priori $p_1, p_2, p_3 \dots p_n$ (con $\sum p_i = 1!$) si estrae un numero r distribuito uniformemente tra 0 e 1
 - Se $r < p_1$ si considera l'evento di tipo 1
 - Se $p_1 < r < (p_1 + p_2)$ si considera l'evento di tipo 2
 - Se $(p_1 + p_2) < r < (p_1 + p_2 + p_3)$ si considera l'evento di tipo 3
 - Etc...
- Si otterranno frequenze di eventi dei vari tipi che, se calcolate con N estrazioni con N tendente a infinito, tenderanno alle probabilità a priori.