



Corso di Laboratorio di Meccanica e Termodinamica

Modulo ROOT

Lezione IV

Silvia Arcelli

8 Maggio 2024

METODO MONTE CARLO

La paternità del nome si deve al matematico N.C. Metropolis (1944). Uno dei “Padri” del metodo Monte Carlo è stato Enrico Fermi, che lo utilizzò negli anni 30 e 40, sviluppandolo e formalizzandolo nell’ambito del Progetto Manhattan insieme ad altri scienziati, fra cui John von Neumann.

- Metodo statistico largamente usato in moltissimi ambiti, fra cui la fisica, per ottenere informazioni su un sistema quando l’uso di calcoli analitici è proibitivo per la complessità delle condizioni:

- Fisica Teorica**: calcolo di integrali, descrizione del comportamento di sistemi complessi

- Fisica Sperimentale**: simulare il comportamento di un sistema fisico (molte particelle con diverse proprietà intrinseche e cinematiche), simulazione degli effetti strumentali dei rivelatori,...

- Alla base del Metodo Monte Carlo sta la generazione di **Numeri Casuali**

GENERAZIONE DI NUMERI CASUALI

Numeri casuali: valore assunto da una variabile aleatoria, per definizione imprevedibile sulla base delle occorrenze passate.

- Alla base del **metodo Monte Carlo** ci sarebbero quindi sequenze di numeri casuali, “difficilmente” ottenibili in grandi quantità nel concreto (processi reali: lancio di una moneta, tempi di arrivo dei raggi cosmici,..)
- Per questioni di efficienza, i generatori di Monte Carlo si basano su sequenze di **numeri Pseudo-Casuali (PRNG)** che hanno alla base un **algoritmo matematico**, quindi per costruzione perfettamente deterministiche e predicibili:

$$X_{n+1} = f(X_n)$$

GENERAZIONE DI NUMERI CASUALI

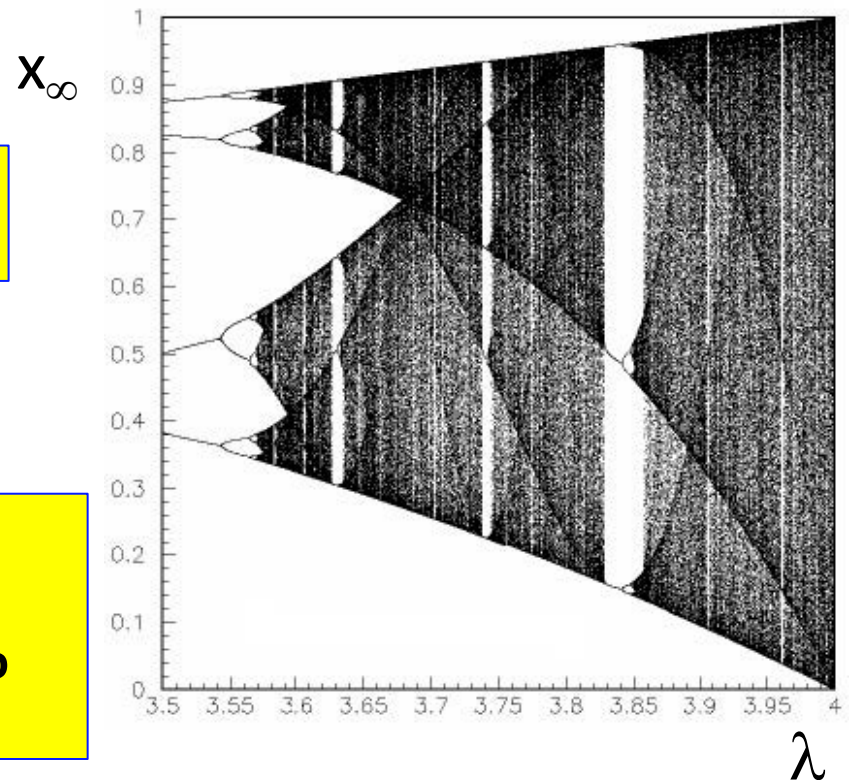
Gli algoritmi utilizzati hanno tuttavia delle proprietà tali per cui le sequenze da loro prodotte sono caratterizzate da comportamenti molto simili a quelle di numeri veramente casuali. Esempio concettuale ([mappa logistica](#)):

$$x_{n+1} = \lambda \cdot x_n (1 - x_n)$$

Comportamento asintotico
per **piccoli** valori del parametro ($\lambda \approx 3$):

$$n \rightarrow \infty \quad x_{\infty} = \frac{\lambda - 1}{\lambda}$$

Per **λ maggiori** si assiste ad una convergenza verso un insieme di valori oscillanti e infine a una transizione verso un comportamento “caotico”:



GENERAZIONE DI NUMERI PSEUDO CASUALI

Quindi esistono algoritmi che, pur essendo basati su una regola deterministica, in opportune condizioni esibiscono un comportamento “casuale”. I generatori di numeri **pseudo casuali (PRNG)** sul mercato generalmente producono:

- O sequenze di **interi uniformemente distribuiti** in un intervallo $[0, N_{MAX}]$
- O numeri **reali uniformemente distribuiti** in $[0,1]$.

Batterie di test per verificare **indipendenza** e **uniformità**. Gli algoritmi disponibili si possono qualificare:

- In termini di **prestazioni** (per esempio la loro velocità)
 - In termini del **periodo**, ovvero dopo quante iterazioni si ricominciano a ripercorrere nella sequenza stati già generati in precedenza. La sequenza è inizializzata in base a un **seme**. Semi diversi producono sequenze diverse e indipendenti.
-

GENERAZIONE DI DISTRIBUZIONI GENERICHE

•Gli algoritmi di generazione delle sequenze Pseudo Random producono quindi numeri secondo una **distribuzione uniforme** in un certo intervallo. Se si volessero (come accade) generare numeri distribuiti secondo altre pdf (pdf= probability density function), come si fa?

Numeri casuali di distribuzioni diverse da quella uniforme possono essere ottenuti a partire da numeri casuali uniformemente distribuiti applicando due tipi di metodi:

- 1. Metodo del rigetto (anche detto hit-or-miss)**
 - 2. Metodo della trasformazione inversa**
-

METODO DEL RIGETTO

Supponete di voler generare random una variabile x distribuita secondo una densità di probabilità $f(x)$ su un dominio finito $[a,b]$.

Metodo *“hit or miss”* o *“del rigetto”*:

(John von Neumann, 1951):

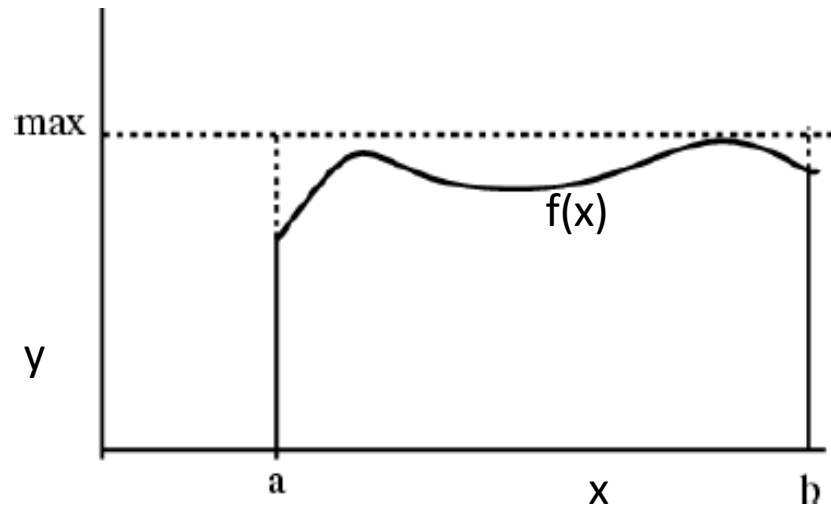
Dominio di x :

$$x = U(a, b),$$

Dominio di y :

$$y = U(0, \max)$$

$$\max = \max(f(x)) \text{ in } [a, b]$$

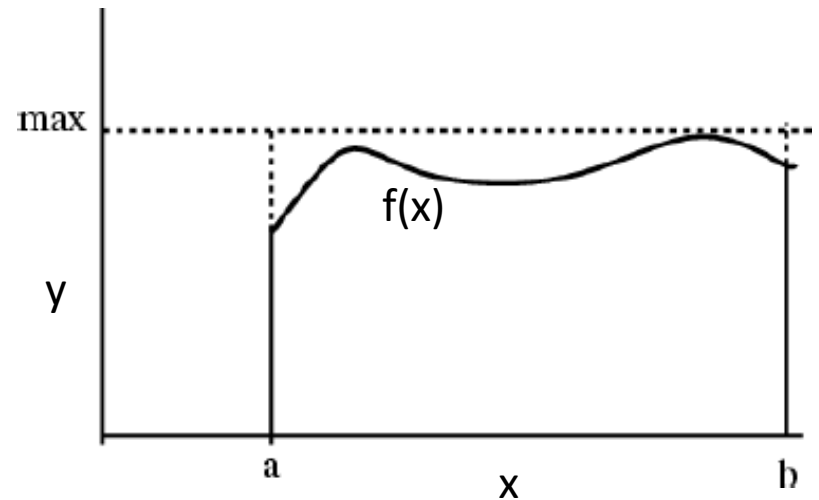


1) **Primo step:** generare due variabili casuali x, y distribuite uniformemente nei corrispondenti domini, utilizzando un PRNG uniforme.

METODO DEL RIGETTO

2) **Secondo Step**: definite una funzione $h(x,y)$ tale per cui:

$$\begin{aligned} h(x, y) &= 0 & \text{se } y > f(x) \\ h(x, y) &= 1 & \text{se } y < f(x) \end{aligned}$$



3) **Terzo Step**: Applicate il criterio di «hit or miss», ovvero:

- Se $h(x,y) = 0$ scartate l'estrazione
- se $h(x,y)=1$ accettate l'estrazione

$$x = U(a, b),$$

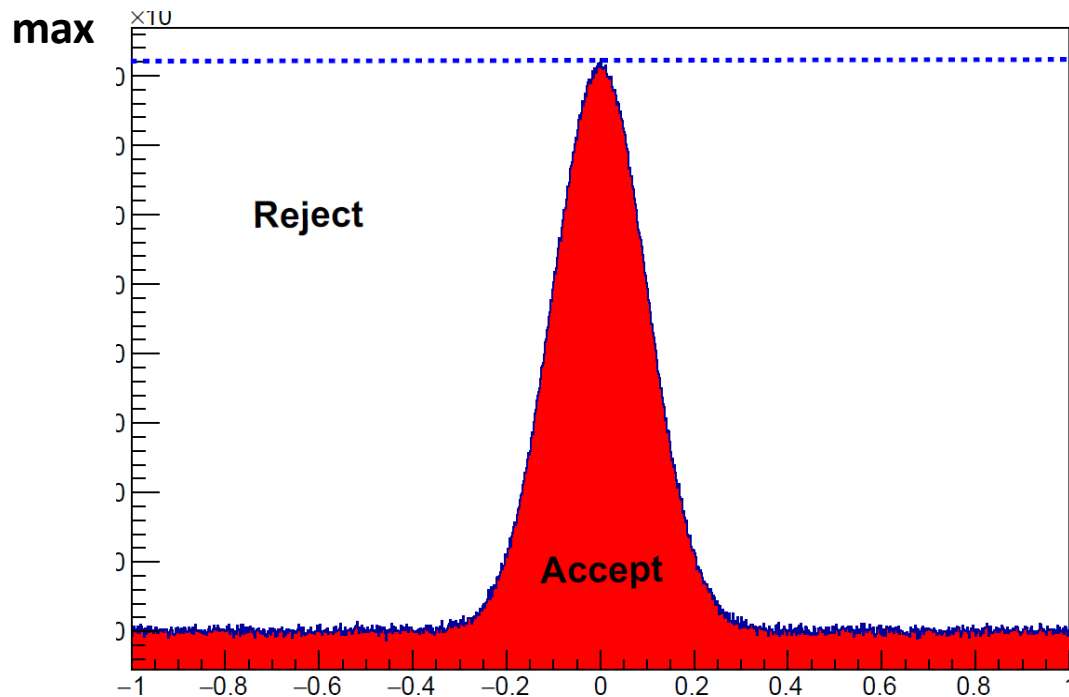
$$y = U(0, \max)$$

In questo modo la variabile x risulterà distribuita secondo $f(x)$. Sempre applicabile, anche se può essere molto inefficiente a seconda del tipo di funzione $f(x)$.

METODO DEL RIGETTO

Limitazioni:

- in linea di principio, è applicabile solo a pdf definite su un dominio finito (ma l'infinito su un computer non esiste...).
- Può essere un metodo poco efficiente, a seconda della forma della distribuzione



Per esempio, funzione che assume nel range valori molto diversi fra loro /presenza di un picco nella $f(x)$
→ alto rate di rigetto

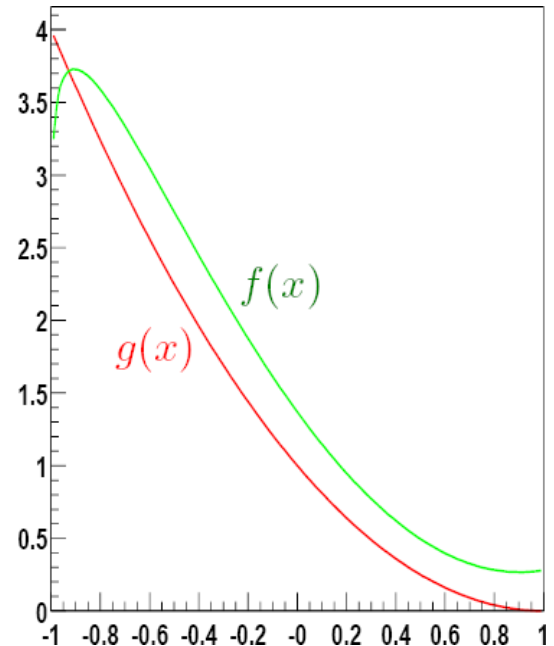
METODO DEL RIGETTO

Per **aumentare l'efficienza** del metodo:

- a) definire un'altra PDF $g(x)$ per la quale la generazione di punti casuali è semplice e veloce (per esempio con il metodo della trasformazione inversa, vedere slide successive), e tale che, con un opportuno fattore di scala S , sia verificata la condizione:

$$g'(x) = Sg(x) \geq f(x)$$

$g'(x)$ "copre" $f(x)$



METODO DEL RIGETTO

Fatto ciò,

- b)** Generate punti in x secondo la $g'(x)$, e un altro numero casuale $y \in U(0,1)$
- c)** Se $y \cdot g'(x) < f(x)$ l'estrazione viene accettata, altrimenti viene rigettata e si reitera il punto b).

In sostanza, “ridefinite” il max del caso precedente per diminuire il *rate* (frequenza) di rigetto

METODO DELLA TRASFORMAZIONE INVERSA

- Se la funzione $f(x)$ che definisce la pdf è **integrabile** e la sua **distribuzione cumulativa $F(x)$** è **invertibile** è molto più efficiente usare il metodo della **Trasformazione (o Ripartizione) inversa**:

$$\int_a^b f(x) dx = 1 \qquad P(x < X) = F(X) = \int_a^X f(x) dx$$

- La distribuzione cumulativa di probabilità è sempre una variabile aleatoria uniformemente distribuita in $[0, 1]$, indipendentemente dalla distribuzione di partenza. Il metodo sfrutta il fatto che la **variabile y** definita come:

$$y = F^{-1}(U) \qquad \text{dove:} \qquad U \in U(0,1)$$

sarà distribuita secondo la pdf di partenza, la **$f(x)$** .

METODO DELLA TRASFORMAZIONE INVERSA

- Dimostriamo che ciò è vero attraverso **la condizione sulla cumulativa**: ossia, che la **variabile y** prima definita ha una **probabilità cumulativa** rappresentata dalla **F(X)**:

$$P(y < X) = F(X) \quad \text{dove:} \quad \left\{ \begin{array}{l} y = F^{-1}(U) \\ U \in U(0,1) \end{array} \right.$$

- Da questo seguirà automaticamente che la sua p.d.f. è la **f(x)** (**corrispondenza 1-1 fra pdf e cumulativa**).

METODO DELLA TRASFORMAZIONE INVERSA

- Dimostrazione

$$P(y < X) = P(F^{-1}(U) < X) \quad (\text{per definizione di } y)$$

- Applicando F a entrambi i membri della disuguaglianza:

$$P(F^{-1}(U) < X) = P(F(F^{-1}(U)) < F(X)) = P(U < F(X))$$



La relazione d'ordine si preserva poiché la F è **monotona crescente**

METODO DELLA TRASFORMAZIONE INVERSA

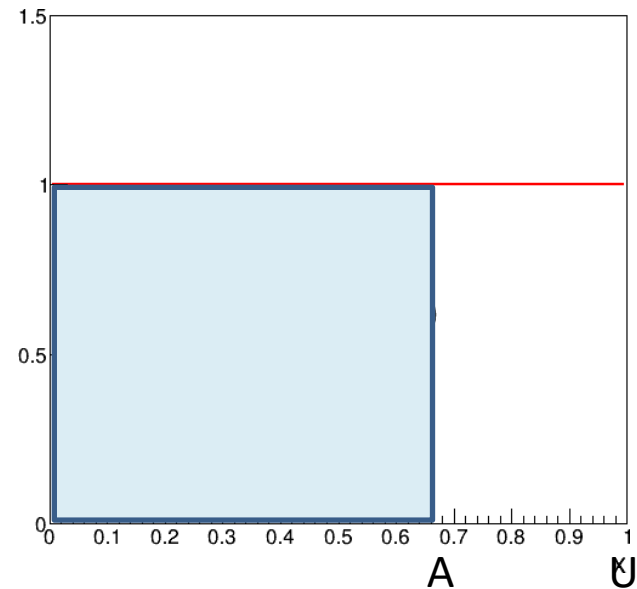
- U è, per ipotesi, un numero **uniformemente distribuito fra 0 e 1**. Qual è la probabilità per un numero siffatto di essere minore di un certo valore «A»?

$$U \in U(0,1)$$

$$P(U < A) = A$$

- E quindi possiamo concludere che:

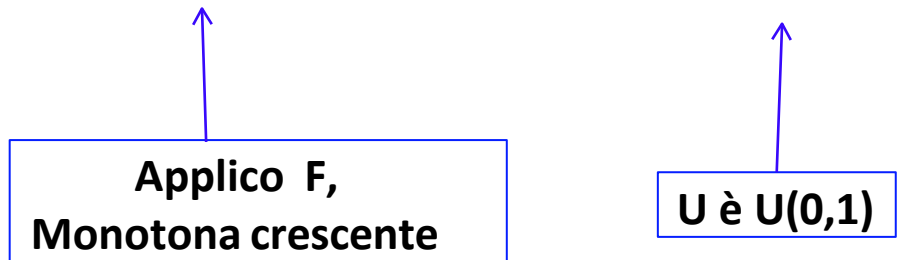
$$P(U < F(X)) = F(X)$$



METODO DELLA TRASFORMAZIONE INVERSA

- Riassumendo tutti i passaggi, si ha quindi:

$$P(y < X) = P(F^{-1}(U) < X) = P(F(F^{-1}(U)) < F(X)) = P(U < F(X)) = F(X)$$



- Usando questo risultato, genero la **variabile aleatoria y** attraverso l'inversa della **funzione cumulativa di probabilità F** calcolata in un numero random uniformemente distribuito, e automaticamente la variabile y risulterà distribuita secondo la pdf **$f(x)$** .

METODO DELLA TRASFORMAZIONE INVERSA

- Esempio: applicazione del metodo della trasformazione inversa alla **distribuzione esponenziale**:

$$f(x) = k e^{-kx} \qquad F(X) = \int_0^X f(x) dx = 1 - e^{-kX}$$

- Generate un numero U in $U(0,1)$ e invertite F per avere y :

$$y = F^{-1}(U) = -\frac{\ln(1 - U)}{k}$$

y sarà distribuita secondo una pdf di tipo esponenziale

METODO DELLA TRASFORMAZIONE INVERSA

- Nel caso della gaussiana unitaria di media 0 e sigma 1, $N(0,1)$:

$$f(x) = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{x^2}{2}}$$

- La pdf gaussiana 1-D **non è integrabile**, ma in due dimensioni lo è:

$$\iint_{x,y} \frac{1}{2\pi} \cdot e^{-\frac{x^2+y^2}{2}} dx dy = \iint_{r,\phi} \frac{1}{2\pi} \cdot e^{-\frac{r^2}{2}} r dr d\phi = 1$$

- E la cumulativa ha la forma:

$$F\left(\frac{R^2}{2}, \varphi\right) = \left(1 - e^{-\frac{R^2}{2}}\right) \cdot \frac{1}{2\pi} \varphi$$

Prodotto di **due cumulative indipendenti**,
una di una distribuzione esponenziale
e una di una distribuzione uniforme

METODO DELLA TRASFORMAZIONE INVERSA

- Per cui se genero due numeri random n_1 e n_2 in $U(0,1)$ e determino x e y come:

$$x = \sqrt{-2 \ln(1 - n_1)} \cos(2\pi n_2),$$

$$y = \sqrt{-2 \ln(1 - n_1)} \sin(2\pi n_2)$$

(Metodo di Box-Muller)

- avrò che x, y sono distribuite secondo una **PDF gaussiana 2-D** con $\mu = 0$ e $\sigma = 1$.
- Per ottenere una **gaussiana 1-D**, essendo x e y indipendenti, basta “ignorare” una delle variabili.
- Per poi generare secondo una **gaussiana generica** $N(\mu, \sigma)$, è sufficiente fare una **trasformazione lineare** del tipo:

$$x' = \sigma \cdot x + \mu$$

METODO DELLA TRASFORMAZIONE INVERSA

Limitazioni del metodo della trasformazione inversa:

- Solitamente richiede che la funzione di distribuzione cumulativa sia descrivibile analiticamente e che sia invertibile. Solo un ristretto numero di funzioni soddisfano queste richieste.
- Tuttavia è possibile integrare **numericamente** la funzione densità di probabilità e **tabulare** (per esempio in un istogramma) la funzione di distribuzione cumulativa. In questo modo è possibile invertire la funzione di distribuzione cumulativa numericamente (uno dei metodi utilizzati da ROOT).

PRNG in ROOT

La generazione di numeri reali in $U(0,1)$ viene effettuata in ROOT con **Rndm()**, uno dei metodi principali della classe TRandom:

- Classe **TRandom**:
 - LCG ($P \approx 10^9$)
 - Velocità: 3 ns/chiamata
- Classe **TRandom1**:
 - Algoritmo RANLUX ($P \approx 10^{171}$)
 - Velocità: 80 ns/chiamata
- Classe **TRandom2**:
 - Algoritmo Tausworthe ($P \approx 10^{26}$)
 - Velocità: 5 ns/chiamata
- Classe **TRandom3** (usato per istanziare il puntatore globale gRandom):
 - Algoritmo Mersenne Twister ($P \approx 10^{6000}$)
 - Velocità: 5 ns/chiamata

• L'inizializzazione della generazione random viene effettuata col metodo **SetSeed()**. Va fatto **una sola volta all'inizio**, prima del ciclo di generazione

• Importante per non ripetere sempre la stessa sequenza di numeri all'avvio del programma.

<https://root.cern.ch/doc/master/classTRandom.html>

PRNG in ROOT: distribuzioni generiche

- Le classi **TRandom** forniscono vari metodi per generare secondo distribuzioni generiche:

- Alcune funzioni sono generate esplicitamente, fra cui:

- **Uniform**(x1,x2)
- **Gaus**(media,sigma)
- **Poisson**(media)
- **Binomial**(ntot,prob)
- **Exp**(tau)
- **Integer**(imax)
- **Landau**(moda,sigma)



facciamo esempi
usando questa modalità,
verificando alcune proprietà statistiche

- In generale si può generare partendo **da qualunque funzione** :

Esempio:

```
TF1 *f1 = new TF1("f1","abs(sin(x)/x)*sqrt(x)",0,10);
```

```
double r = f1->GetRandom(); //r sarà una variabile distribuita secondo la pdf definita da f1
```

- Si può anche **riempire direttamente un istogramma**:

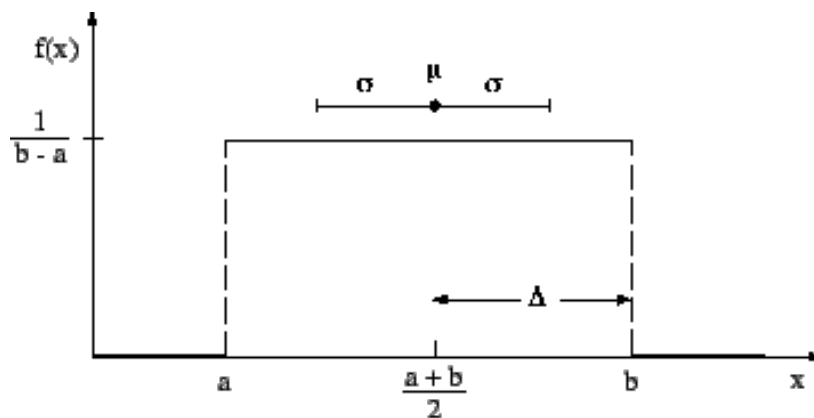
```
histo->FillRandom("f1",n); //histo conterrà n estrazioni di una variabile distribuita secondo la pdf definita dalla funzione f1
```

VERIFICHE CON ROOT

- In questa e la prossima lezione, generazione di variabili casuali con ROOT secondo:
 - Distribuzione Uniforme
 - Distribuzione Gaussiana
 - Distribuzione di Binomiale
 - Distribuzione di Poisson
 - Applicazione per distribuzione Lorentziana
- Esiti salvati in istogrammi. Analisi degli istogrammi per verificare alcune proprietà delle distribuzioni in esame

DISTRIBUZIONE UNIFORME

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{se } x \in [a, b] \\ 0 & \text{se } x \notin [a, b] \end{cases}$$



$$\mu = \int_a^b x \cdot f(x) \cdot dx = \int_a^b x \cdot \frac{1}{b-a} \cdot dx = \frac{(a+b)}{2}$$

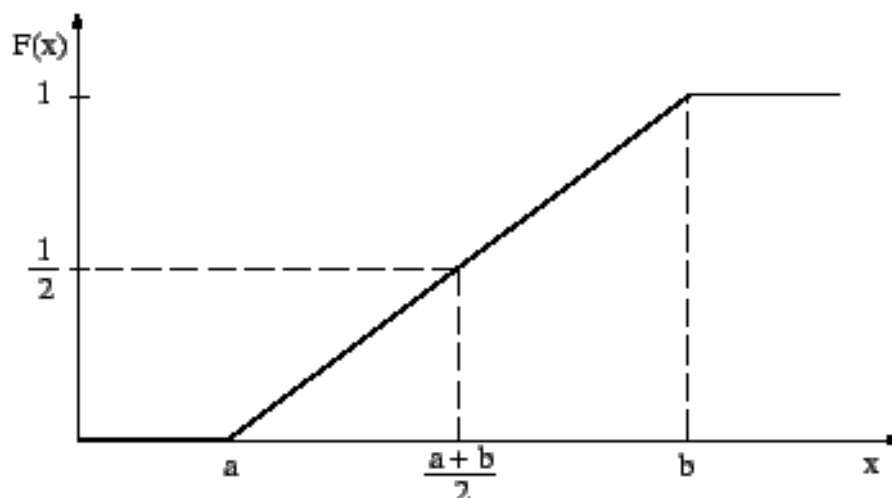
$$\sigma^2 = \int_a^b (x - \mu)^2 \cdot f(x) \cdot dx = \int_a^b \left(x - \frac{a+b}{2}\right)^2 \cdot \frac{1}{b-a} dx = \frac{(a-b)^2}{12}$$

Caratterizzata
da 2 parametri:
a e b

DISTRIBUZIONE UNIFORME

- La Funzione Cumulativa è Lineare:

$$F(x) = \frac{x - a}{b - a}$$



- La probabilità di osservare x in un certo intervallo è **proporzionale all'ampiezza dell'intervallo**

DISTRIBUZIONE UNIFORME

- Istogrammi di variabili generate secondo distribuzioni uniformi:

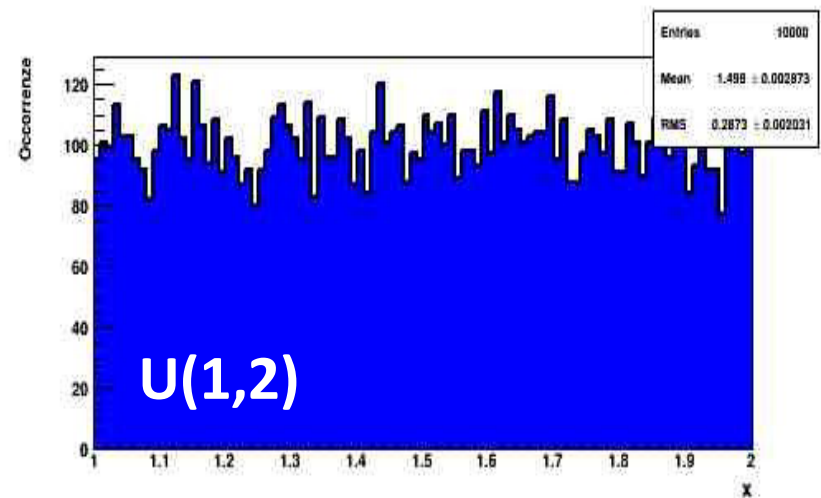
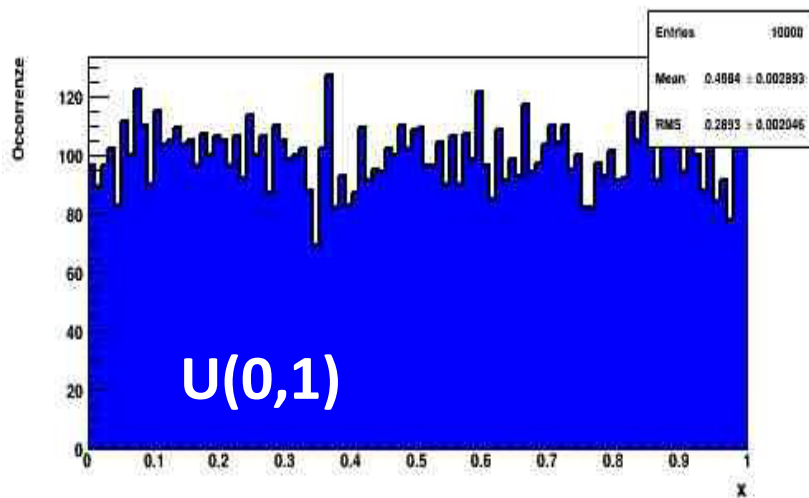
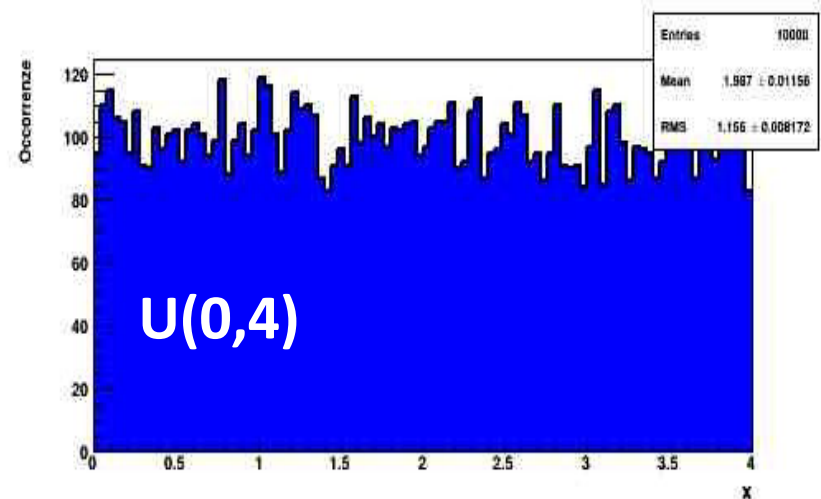
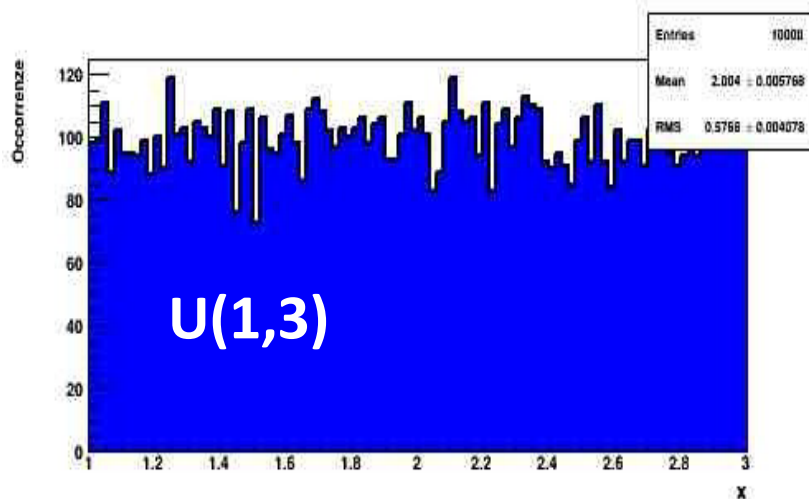
```
for(Int_t j=0;j<nngen;j++){ //ciclo di generazione
    Double_t x=gRandom->Uniform(xmin,xmax); //estrazione
    h->Fill(x); // riempimento istogramma
}
```

- verifica di media e RMS (con relativo errore) usando i metodi:
 - `h->GetMean(), h->GetMeanError(),`
 - `h->GetRMS(),h->GetRMSError():`
- Esempio per 10000 estrazioni da quattro distribuzioni uniformi:
 - 2 medie $(a+b)/2$ uguali, ampiezza dominio $(b-a)$ diverso
 - Per 2 medie diverse, stessa ampiezza dominio

Macro Uniform-Dist.C

DISTRIBUZIONE UNIFORME

Numero di estrazioni N=10000



DISTRIBUZIONE UNIFORME

- Tabella Media e RMS istogramma vs valori attesi di μ e σ

a	b	μ	σ	Media	RMS
1	3	2	0.577	2.004 +/- 0.006	0.576 +/- 0.004
0	4	2	1.153	1.987 +/- 0.011	1.156 +/- 0.008
0	1	0.5	0.289	0.498 +/- 0.003	0.289 +/- 0.002
1	3	1.5	0.289	1.498 +/- 0.003	0.287 +/- 0.002

Incertezze sulla media e RMS
consistenti con:



$$\Delta_x = \sigma / \sqrt{N}$$

$$\Delta_\sigma = \sigma / \sqrt{2N}$$

DISTRIBUZIONE GAUSSIANA

Se X_1, X_2, \dots, X_n sono n variabili casuali Normali indipendenti, la variabile casuale:

$$Y = \alpha_1 X_1 + \alpha_2 X_2 + \dots + \alpha_n X_n$$

è a sua volta una variabile casuale Normale con:

- Valor medio $\mu = \alpha_1 \mu_1 + \alpha_2 \mu_2 + \dots + \alpha_n \mu_n$
- Varianza $\sigma^2 = \alpha_1^2 \sigma_1^2 + \alpha_2^2 \sigma_2^2 + \dots + \alpha_n^2 \sigma_n^2$

Se si sommano N variabili gaussiane, la distribuzione è ancora Gaussiana, con valore medio la somma dei valori medi e come varianza la somma delle varianze

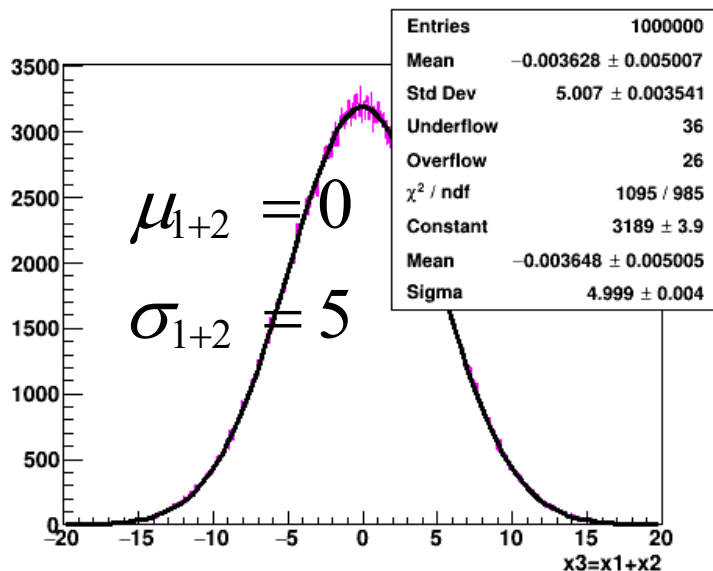
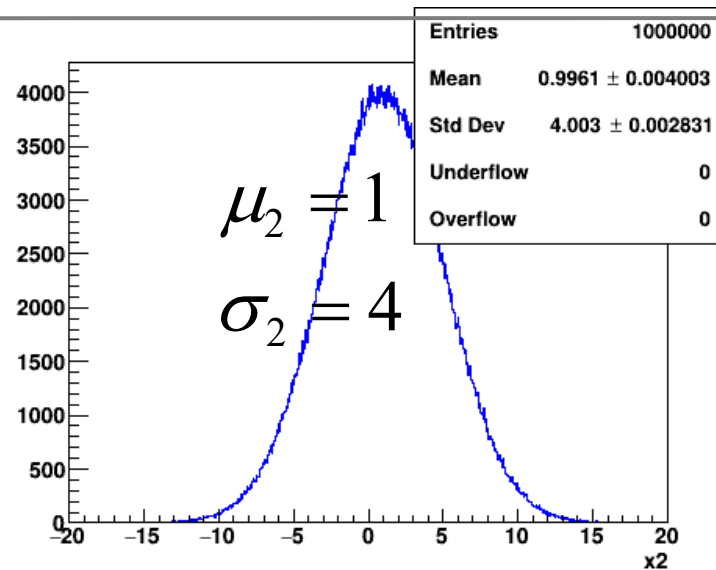
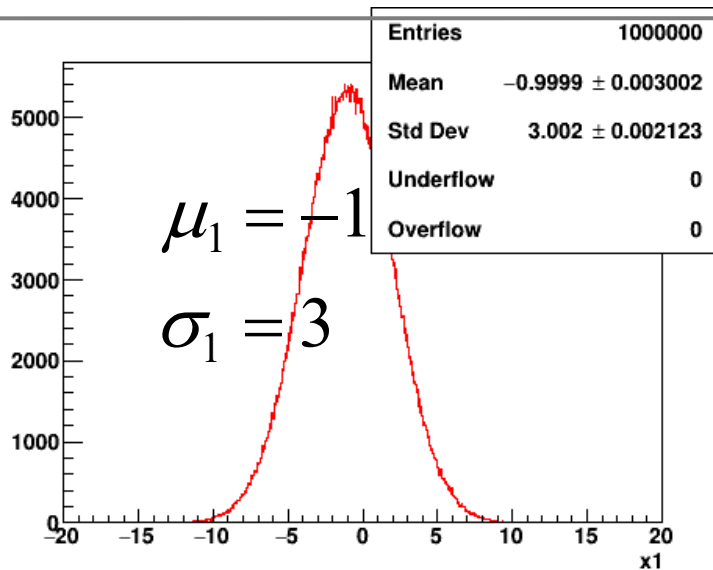
DISTRIBUZIONE GAUSSIANA

- 10^6 estrazioni di due variabili gaussiane $x1 \in G(-1,3)$ e $x2 \in G(1,4)$
- loro somma $x3=x1+x2$

```
Double_t x1,x2,x3;  
for(Int_t i=0;i<1000000;i++){ //ciclo di generazione  
    x1=gRandom->Gaus(-1,3); //estrazione x1  
    x2=gRandom->Gaus(1,4); //estrazione x2  
    x3=x1+x2; //variabile somma  
    h1->Fill(x1);  
    h2->Fill(x2);  
    h3->Fill(x3);  
}
```

Macro Gaussian-Sum.C

DISTRIBUZIONE GAUSSIANA



$$x_3 = x_1 + x_2$$

si ha :

$$\mu_3 = \mu_1 + \mu_2 = 0$$

$$\sigma_3 = \sqrt{\sigma_1^2 + \sigma_2^2} = 5$$