METODI STATISTICI E COMPUTAZIONALI

Stefania Spagnolo

Dipartimento di Matematica e Fisica, Univ. del Salento

APPLICAZIONI DELLA TECNICA MONTE CARLO

Testi consigliati:

W. Kinzel & G. Reents "Physics by Computer" P. 157-163

W.R. Gibbs "Computation in Modern Physics" P. 25-46

Dispense L. Angelini Università di Bari P. 29-44
Dispense D. Martello Università del Salento Cap. 1

G. Cowan, "Monte Carlo techniques" in The review of particle physics 2021

Applicazioni della tecnica Monte Carlo

Metodi MC per l'integrazione di funzioni

- rigetto
- Media di [b-a]f(x)

Metodo MC per implementare un generatore di numeri pseudocasuali non uniforme

- rigetto
- inversione

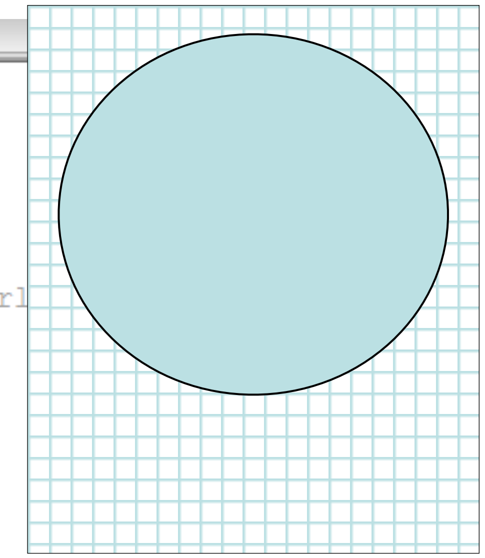
LEZIONE 2

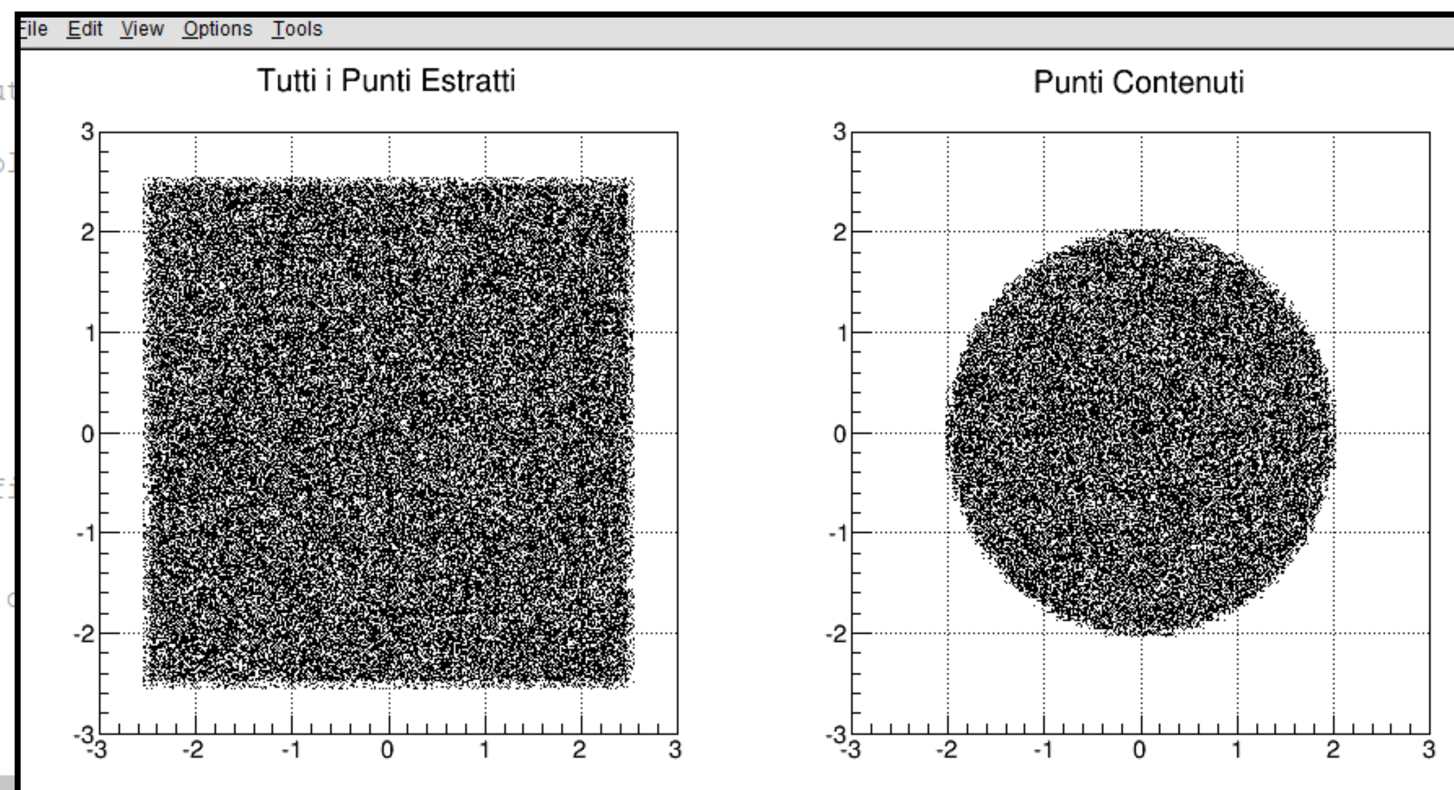
UN ESEMPIO DI MONTE CARLO: L'AREA DEL CERCHIO

Macro: Cerchio.C

re | Generatore1.C | Generatore2.C | Generatore3.C | Generatore4.C | Dado.C | Cerchio.C

```
#include "TFile.h"
// Creo una script di nome Cerchio
void Cerchio(){
// creo un "oggetto" generatore di numeri random
   TRandom R(344545);
   int N=100000;
// creo un oggetto file lo collego al file di nome esempio1.root e gli dico di crearlo (ricrearl
   TFile f("cerchio.root","RECREATE");
// Raggio del cerchio
   double raggio=2.;
// creo un oggetto histogramma
   TH2D * h1 = new TH2D ("h1","Tutti i Punti Estratti",80,-3.,3.,80,-3.,3.);
   TH2D * h2 = new TH2D ("h2","Punti Contenuti",80,-3.,3.,80,-3.,3.);
// inizializzo sequenza di conteggio
   int In=0;
   for (int i=0; i<N ; i++) {
      double x,y;
// Uso l'oggetto generatore per generare punti in maniera uniforme
// su un quadrato di lato 5 centrato in 0,0
      x=R.Uniform()*5.-2.5;
      y=R.Uniform()*5.-2.5;
// Riempio l'oggetto histogramma con tut
      h1->Fill(x,y);
// Riempio l'oggetto histogramma con sol
      if (sqrt(x*x+y*y)<raggio) {
         h2->Fill(x,y);
         In++;
      }
   }
// Calcolo area
   double area=5.*5.*In/N;
   printf("Area cerchio %lf\n",area);
// scrivo gli oggetti histogramma nel fi
   h1->Write();
   h2->Write();
// chiudo il file attraverso il metodo c
   f.Close();
}
```



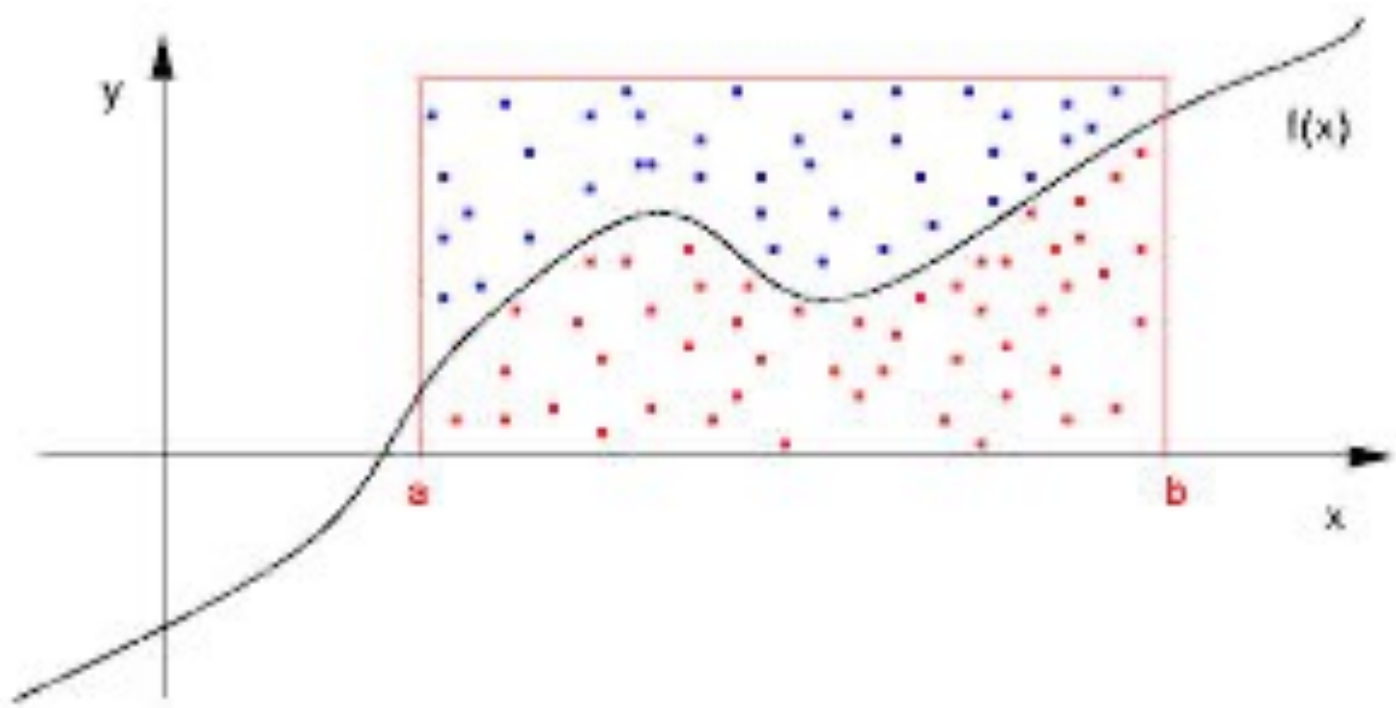


MISURA DI INTEGRALI CON IL METODO DELLA REIEZIONE O RIGETTO

- La tecnica Monte Carlo che abbiamo usato per determinare la superficie di un cerchio può essere usata per calcolare un integrale

Calcoliamo l'integrale $\int_1^{10} (2x^2 + 1)dx = 675$

$$= \left[(2/3)x^3 + x\right]_1^{10} = 2000/3 + 10 - 2/3 - 1$$



Misura di integrali con il metodo della reiezione o rigetto

- **Primo passo**: identificare un rettangolo (di area A) che inscrive la funzione nell'intervallo di integrazione.

- **Secondo passo**: estrarre N punti casuali all'interno del rettangolo e contare il numero N_a di punti (accettati) che cadono sotto la curva che rappresenta la funzione

- **Terzo passo**: la frazione di punti accettati sul totale moltiplicata per A da il valore dell' integrale

hitormiss.C | FiguraIntegrale.C | Cerchio.C

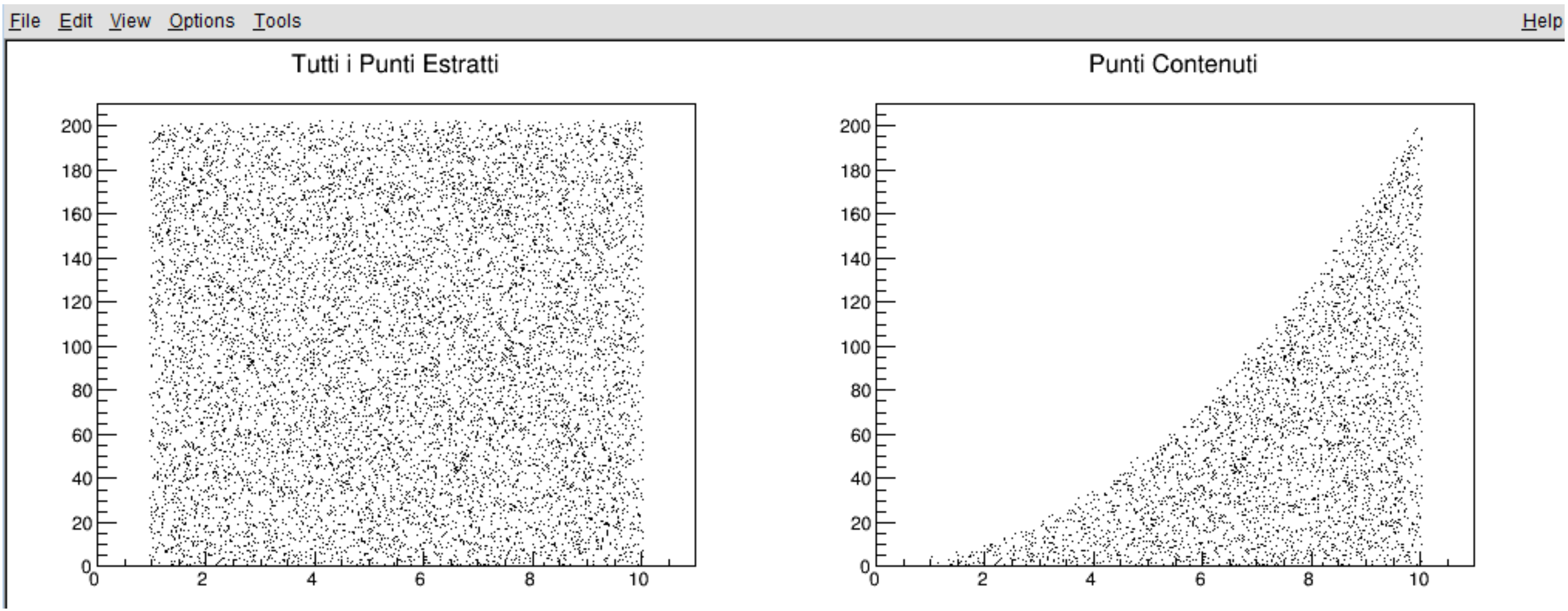
```
//
// calcoliamo l'integrale di questa funzione
double f(double x) {
    return 2.*x*x+1.;
}
void Reiezione(){
// creo un "oggetto" generatore di numeri random
   TRandom R(12345);
   int N=10000;
// creo un oggetto file lo collego al file di nome Reiezione.root
// e gli dico di crearlo (ricrearlo) quindi è un output
   TFile f("Reiezione.root","RECREATE");
   TH2D * h1 = new TH2D ("h1","Tutti i Punti Estratti",80,0.,11.,80,0.,210);
   TH2D * h2 = new TH2D ("h2","Punti Contenuti",80,0.,11.,80,0.,210);
// inizializzo sequenza di conteggio
   int In=0;
   for (int i=0; i<N ; i++) {
      double x,y;
// Uso l'oggetto generatore di numeri random per generare numeri random in maniera uniforme
      x=R.Uniform()*9+1;
      y=R.Uniform()*201;

// Riempio l'oggetto histogramma con tutti numeri random generati
      h1->Fill(x,y);
// Riempio l'oggetto histogramma con solo i punti contenuti
      if (y<f(x)) {
           h2->Fill(x,y);
           In++;
      }
   }
// Calcolo area
   double area=9.*201.*In/N;
   printf("Integrale = %lf\n",area);
// scrivo gli oggetti histogramma nel file
   h1->Write();
   h2->Write();
// chiudo il file attraverso il metodo dell'oggetto di tipo file.
   f.Close();
}
```

MISURA DI INTEGRALI CON IL METODO DELLA REIEZIONE O RIGETTO

- Quale rettangolo ?

- Mappa dei punti estratti casualmente

- Mappa dei punti accettati



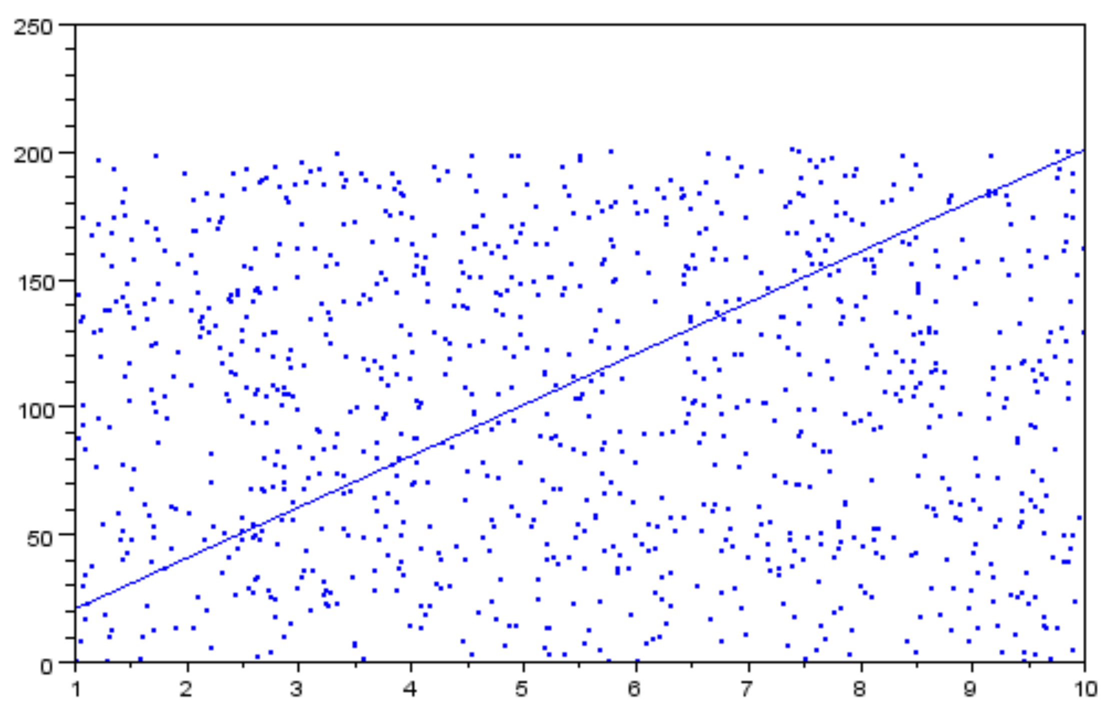
MISURA DI INTEGRALI CON IL METODO DELLA REIEZIONE O RIGETTO

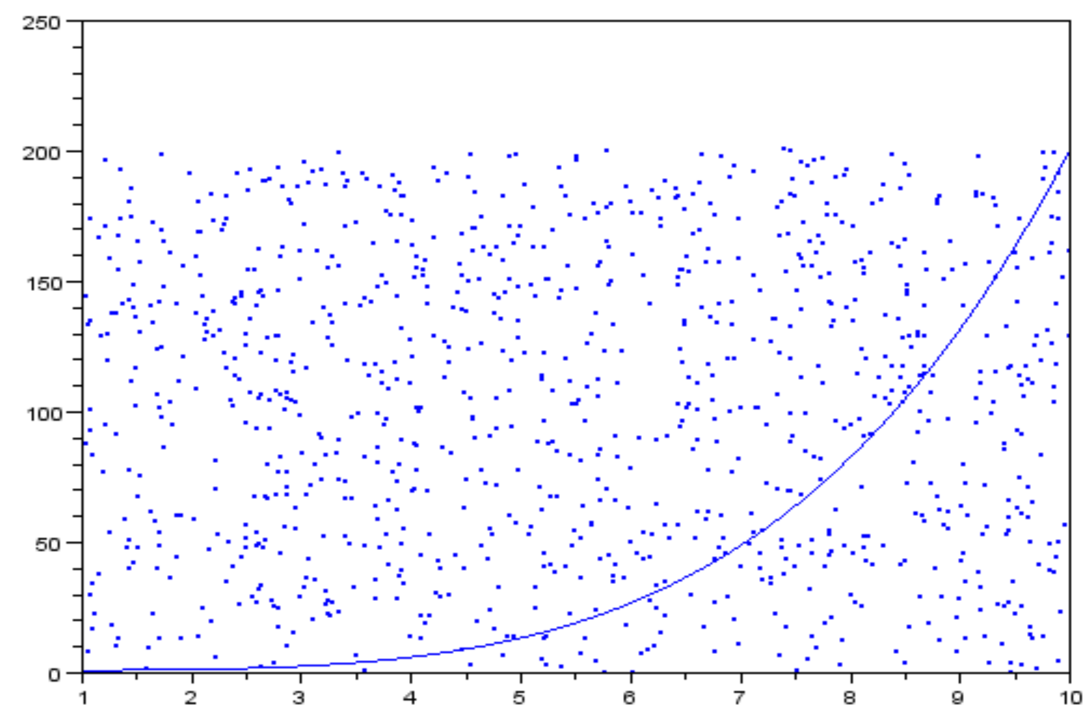
- Con il metodo del rigetto è possibile calcolare integrali definiti anche di funzioni $f(x)$ non definite positive, basta, infatti, considerare la funzione: $g(x) = f(x) - f_{min} > 0$ sempre nel dominio di integrazione se f_{min} e' il minimo di $f(x)$
  - $$\int_a^b f(x)dx = \int_a^b g(x)dx + f_{min}(b-a)$$

- Non e' possibile calcolare con questo metodo integrali di funzioni con singolarità anche se matematicamente (analiticamente) integrabili
  - Solo per funzioni limitate nel dominio di definizione e' possibile determinare il minimo e il massimo, necessari alla costruzione del rettangolo

MISURA DI INTEGRALI CON IL METODO DELLA REIEZIONE O RIGETTO

- Efficienza del metodo:
  - NOTA:
    - Numerosita' del campione che contribuisce alla stima statistica dell'integrale e' $N_{accettati}$
    - efficienza $\rightarrow \epsilon = N_{accettati}/N_{totali}$
    - numero di estrazioni totali $N_{estrazioni-totali} = 2N$
- E' evidente che il metodo è tanto più efficiente quanto più l'area del rettangolo che include la funzione è prossima alla superficie che si deve determinare.





MISURA DI INTEGRALI CON IL METODO DELLA REIEZIONE O RIGETTO

Macro: Integrale_Reiezione.C

- Efficienza del metodo:
  - NOTA:
    - Numerosita' del campione che contribuisce alla stima statistica dell'integrale e' $N_{accettati}$
    - efficienza $\rightarrow \epsilon = N_{accettati}/N_{totali}$
    - numero di estrazioni totali $N_{estrazioni-totali} = 2N$
- E' evidente che il metodo è tanto più efficiente quanto più l'area del rettangolo che include la funzione è prossima alla superficie che si deve determinare.

- Stima dell'integrale
  - $I_{MC,rejection} = \text{Area} \times \dfrac{N_{acc}}{N} = \text{Area} \times \epsilon$
  - Qual e' l'errore statistico ?
    - Vedremo che l'errore statistico sulla nostra stima di epsilon e' $\sigma(\epsilon) = \sqrt{\epsilon(1-\epsilon)/N}$ (distribuzione di prob. binomiale)
    - Quindi l'errore su $I_{MC,rejection} = \text{Area} \times \sqrt{\epsilon(1-\epsilon)/N}$

METODI MONTE CARLO

- Il metodo Monte Carlo è in sostanza un metodo di integrazione numerica
- Le basi matematiche dell'integrazione Monte Carlo sono le basi della statistica
  - ***Definizione di una variabile casuale***
  - ***Distribuzioni delle variabili casuali. Media, varianza, covarianza…***
  - ***La legge dei grandi numeri***
  - ***Il teorema del limite centrale***

torniamo su questo concetto

Integrazione con il metodo Monte Carlo

LA LEGGE DEI GRANDI NUMERI

- La legge dei grandi numeri mi garantisce di poter calcolare un integrale con tecniche Monte Carlo.
- In termini generici, ***per la legge dei grandi numeri*** si può dire che:
  - la media di una sequenza di variabili aleatorie indipendenti e caratterizzate dalla stessa distribuzione di probabilità (n misure della stessa grandezza, n lanci della stessa moneta ecc.) è una approssimazione, che migliora al crescere di n, della media della distribuzione

Variabile aleatoria	Distribuzione di probabilità	Media (o valore di aspettazione)	
x	$g(x)$	$\mu = \int_a^b g(x) x dx$	
$y = f(x)$	$g(x)$	$\boxed{\mu = \int_a^b f(x) g(x) dx}$	**law of the unconscious statistician**, or **LOTUS**

DIGRESSIONE - LOTUS

Da rivedere nella lezione 7-8, funzioni di variabili aleatorie

law of the unconscious statistician, or **LOTUS**

Definition 5.2 (Law of the unconscious statistician (LOTUS)) The **"law of the unconscious statistician" (LOTUS)** says that the expected value of a transformed random variable can be found without finding the distribution of the transformed random variable, simply by applying the probability weights of the original random variable to the transformed values.

$$\text{Discrete } X \text{ with pmf } p_X: \qquad \mathrm{E}[g(X)] = \sum_x g(x) p_X(x)$$

$$\text{Continuous } X \text{ with pdf } f_X: \qquad \mathrm{E}[g(X)] = \int_{-\infty}^{\infty} g(x) f_X(x) dx$$

The left-hand side of LOTUS, $\mathrm{E}[g(X)]$, represents finding the expected value the "long way": define $Y = g(X)$, find the distribution of Y (e.g., using the cdf method in Section 4.6), then use the definition of expected value to compute $\mathrm{E}(Y)$. LOTUS says we don't have to first find the distribution of $Y = g(X)$ to find $\mathrm{E}[g(X)]$; rather, we just simply apply the transformation g to each possible value x of X and then apply the corresponding weight for x to $g(x)$.

https://bookdown.org/kevin_davisross/probsim-book/law-of-the-unconscious-statistician-lotus.htm

DIGRESSIONE - LOTUS

Da rivedere nella lezione 7-8, funzioni di variabili aleatorie

law of the unconscious statistician, or **LOTUS**

Continuous X with pdf f_X: $\quad \mathrm{E}[g(X)] = \int_{-\infty}^{\infty} g(x) f_X(x) dx$

$$E[Y = g(x)] = \text{def} = \int y f_Y(y) dy$$

$f_Y(y)$ -> pdf for y

https://bookdown.org/kevin_davisross/probsim-book/law-of-the-unconscious-statistician-lotus.htm

INTEGRAZIONE DI FUNZIONE CON IL METODO MC

Supponiamo di voler valutare il seguente integrale

$$I = \int_a^b f(x)\, dx \tag{2.2}$$

dove $f(x)$ è una funzione reale di variabile reale, ovviamente integrabile in $[a, b]$, con $b > a$. Consideriamo la seguente funzione

$$g(x) = \begin{cases} \frac{1}{b-a} & \text{per } a \leq x \leq b, \\ 0 & \text{altrove.} \end{cases}$$

Densita' di probabilita' uniforme tra a e b della variabile casuale x

L'integrale può essere riscritto nella forma

$$I = (b-a) \underbrace{\int_a^b f(x)\, g(x)\, dx}_{\boxed{\langle F \rangle}}$$

Valore di aspettazione μ=<F> di f(x) per x variabile casuale distribuita con probabilita' uniforme tra a e b

$g(x)$ rappresenta una distribuzione di probabilità uniforme in $[a, b]$; possiamo quindi considerare l'integrale come valore di aspettazione di $f(x)$, con x variabile aleatoria distribuita secondo $g(x)$.

INTEGRAZIONE DI FUNZIONE CON IL METODO MC

- Per la legge dei grandi numeri, estratti N valori di x, $\{x_i\}$, la media campionaria (aritmetica) di $f(x_i)$ $\bar{F}$ tende a <F> per N->infinito

Media sul campione di N valori di x

N->infinito

$$\bar{F} = \frac{1}{N}\sum_{i=1}^{N} f(x_i) \rightarrow \; < F >$$

$$= \int_a^b f(x)g(x)dx$$

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

Integrale di f(x)

$$= \frac{1}{b-a} I$$

- **L'integrale *I* sull'intervallo [a,b] di una funzione f(x) può essere stimato come $(b-a)\bar{F}$ con approssimazione tanto maggiore quanto più grande è N**
  - *Qual è l'errore che commetto ?*

IL TEOREMA DEL LIMITE CENTRALE

- ***La somma di variabili aleatorie indipendenti e identicamente distribuite*** con media μ e varianza σ², ***indipendentemente dalla distribuzione di probabilità*** di partenza, al tendere della dimensione del campione a infinito tende a distribuirsi come una ***variabile casuale gaussiana***

$$\bar{F} = \frac{1}{N}\sum_{i=1}^{N} f(x_i) \rightarrow < F >$$

$\bar{F}$ si distribuisce secondo una pdf gaussiana con media μ = <F> e varianza var($\bar{F}$) = σ² = var(f) / N

La deviazione standard σ di $\bar{F}$ sarà

$$\sqrt{var(f)}/\sqrt{N} = \sigma(f)/\sqrt{N}$$

INTEGRAZIONE DI FUNZIONE CON IL METODO MC

Lo stimatore MC dell'integrale I, cioè $I_{MC} = (b-a)\bar{F}$, è affetto da un errore pari a $(b-a)\ \sigma(f(x))$ / $\sqrt{N}$

che possiamo stimare dal campione come (b-a) RMS({f(x_i)}) / sqrt(N)

Stima campionaria della deviazione standard di f

- **L'integrale *I* sull'intervallo [a,b] di una funzione f(x) può essere stimato come $(b-a)\bar{F}$ con approssimazione tanto maggiore quanto più grande è N**
  - *Qual è l'errore che commetto ?*

INTEGRAZIONE DI FUNZIONE CON IL METODO MC

- ESEMPIO

Calcoliamo l'integrale $\int_1^{10} (2x^2 + 1)dx = 675$

- Per le considerazioni appena illustrate (da legge dei grandi numeri e teorema del limite centrale) ho

$$\frac{1}{N}\sum_{i=1}^{N}(2x_i^2 + 1)_{N \to \infty} \to \frac{1}{b-a}\int_a^b (2x^2 + 1)dx$$

x_i valori della variabile casuale x distribuita in maniera uniforme tra a e b

- Quindi se estraggo N numeri casuali {xi} calcolo la f(xi) e ne faccio la media ottengo asintoticamente il valore dell'integrale diviso l'ampiezza dell'intervallo.
- Nel nostro esempio: 675/9=75

INTEGRAZIONE DI FUNZIONE CON IL METODO MC

Macro: Integrale_MonteCarlo.C

```
//
//  Calcolo integrale con Metodo Monte Carlo
//
// calcoliamo l'integrale di questa funzione
double f(double x) {
     return 2.*x*x+1.;
}
void Integrale(){
// creo un "oggetto" generatore di numeri random
   TRandom R(345345);
   int N=100000;
// Inizializzo somma e somma2
   double somma=0,somma2=0;
   for (int i=0; i<N ; i++) {
      double x,y;
// Uso l'oggetto generatore di numeri random per generare numeri random in maniera uniforme
      x=R.Uniform()*9+1;
      y=f(x);
      somma+=y;
      somma2+=y*y;
   }
// Calcolo medie
   double media=somma/N;
   double media2=somma2/N;
// Calcolo Area
   double area=media*(10.-1.);
// Calcolo errore
   double sigma=sqrt(media2-media*media);
   double errore=sigma*(10.-1.)/sqrt(N);
   printf("Integrale = %lf  +/- %lf\n",area,errore);
}
```

Una formula alternativa per la varianza è

$$\sigma_X^2 = \mathbb{E}[X^2] - \mathbb{E}[X]^2$$

Questa formula è più pratica per calcolare la varianza.

GENERATORE DI NUMERI PSEUDO-CASUALI NON UNIFORMEMENTE DISTRIBUITI

ANCORA SUL

METODO DI REIEZIONE

GENERATORI PER PDF NON UNIFORMI / RIGETTO

- Nel simulare un processo spesso è necessario estrarre numeri casuali in modo tale da seguire una specifica distribuzione.

- Serve, cioè, che i numeri estratti seguano una distribuzione di probabilità assegnata.

- Nei corsi precedenti avete incontrato alcune distribuzioni probabilità molto usate (Binomiale, Poisson, Gauss), ma in generale possono esserci processi aleatori che seguono funzioni di probabilità di tipo diverso.
  - Molti processi fisici sono processi stocastici

- Per simulare questi processi, quindi, occorre estrarre numeri casuali in modo tale che essi seguano un'arbitraria distribuzione di probabilità.

- Una qualunque funzione definita in un intervallo (a,b), continua e derivabile N volte, definita positiva e tale che il suo integrale in (a,b) sia pari ad 1 può rappresentare una distribuzione di probabilità di una qualche variabile aleatoria (o processo fisico)

$$g(x) \geq 0 \qquad \forall x \in [a,b]\,; \qquad \int_a^b g(x)dx = 1$$

GENERATORI PER PDF NON UNIFORMI / RIGETTO

- L'idea di base è molto semplice: ***dato un insieme di numeri casuali, ne vogliamo eliminare una parte in maniera che i rimanenti siano distribuiti secondo una certa densità di probabilità***.

- Supponiamo di voler estrarre numeri che seguano la distribuzione di probabilità g(x) definita in (a,b) e supponiamo che in (a,b) la funziona abbia massimo g_{max} o che esista un g_{max} tale che

$$g(x) < g_{max} \qquad \forall x \in [a,b]$$

  - 1) estraiamo in maniera causale e secondo una distribuzione uniforme x_i nell'intervallo [a, b]
  - 2) definiamo un criterio di accettazione per x_i tale che gli x_i accettati risultino distribuito come la pdf che si vuole emulare:
    - Come ?
      - La probabilita' di accettazione di x_i deve essere proporzionale a $g(x_i)$

GENERATORI PER PDF NON UNIFORMI / RIGETTO

- La ***sequenza ottenuta { x_i' }*** risulta distribuita secondo una ***distribuzione di probabilità*** pari al prodotto delle probabilità di estrazione e di accettazione

$$\frac{1}{b-a}dx$$

Probabilità di estrazione di x_i

Estratto yi tra 0 e g_max secondo una distribuzione uniforme, probabilità che $y_i < g(x_i)$ $\quad g(x_i)/g_{max} \longrightarrow$ **Probabilità di accettazione di x_i**

$$\boxed{p(x)dx = \frac{1}{b-a}\ g(x_i)/g_{max}\ dx}$$

p(x) e g(x) differiscono solo per la normalizzazione

GENERATORI PER PDF NON UNIFORMI / RIGETTO

Osservazione

- La ***sequenza ottenuta* { x_i' }** risulta distribuita secondo una ***distribuzione di probabilità*** pari al prodotto delle probabilità di estrazione e di accettazione

$$\frac{1}{b-a}dx$$ Probabilità di estrazione di x_i

probabilità che $y_i < g(x_i)$ $$g(x_i)/g_{max}$$ Probabilità di accettazione di x_i

$$p(x)dx = \frac{1}{b-a}\, g(x_i)/g_{max}\, dx$$

NOTA: ritroviamo il metodo di integrazione con la tecnica del rigetto

$$I = \int_a^b g(x)dx = (b-a)g_{max}\int_a^b p(x)dx = A_{rettangolo}\frac{N_{accettati}}{N}$$

GENERATORI PER PDF NON UNIFORMI / RIGETTO

In pratica

- Si estraggono punti uniformemente distribuiti su un rettangolo che racchiude la funzione densità di probabilità g(x) che si vuole riprodurre
- Si accettano come valori random validi della variabile x, le ascisse dei punti che cadono sotto la curva della funzione densità di probabilità.

```
//
//  Estraggo numeri secondo questa distribuzione di probabilità
double f(double x) {
     return (2.*x*x+1.)/675.;
}
void Generatore(){
// creo un "oggetto" generatore di numeri random
   TRandom R(345345);
   int N=1000000;
// creo un oggetto file lo collego al file di nome Reiezione.root
// e gli dico di crearlo (ricrearlo) quindi è un output
   TFile f("Reiezione.root","RECREATE");
   TH1D * h1 = new TH1D ("h1","Istrogramma di frequenze",80,0.,11.);
// inizializzo sequenza di conteggio
   int In=0;
   for (int i=0; i<N ; i++) {
      double x,y;
// Uso l'oggetto generatore di numeri random per generare numeri random in maniera uniforme
      x=R.Uniform()*9+1;
      y=R.Uniform()*201;

// Riempio l'oggetto histogramma con solo i punti contenuti
      if (y<f(x)) {
           h1->Fill(x);
           In++;
      }
   }
   printf("Ho estratto n=%g valori utili\n",In);
// scrivo gli oggetti histogramma nel file
   h1->Write();
// chiudo il file attraverso il metodo dell'oggetto di tipo file.
   f.Close();
}
```

GENERATORE DI NUMERI PSEUDO-CASUALI NON UNIFORMEMENTE DISTRIBUITI
-
METODO DELL'INVERSIONE

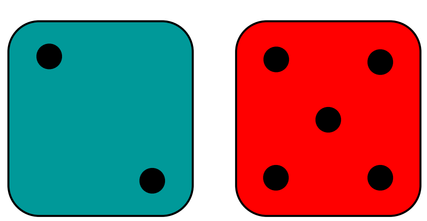
GENERATORI PER PDF NON UNIFORMI / INVERSIONE

- Il metodo di reiezione è semplice ma scarsamente efficiente.
- Se si vogliono ottenere N valori estratti casualmente occorre generare molto più di 2N numeri casuali. Esistono altri algoritmi che permettono, per certi tipi di distribuzioni, di ottenere gli N valori ricorrendo a un numero molto inferiore di estrazioni.

- Cominciamo col considerare una variabile aleatoria discreta, cioè in grado di assumere solo un numero N finiti di valori.

Quale è la distribuzione di probabilità della somma dei numeri sulle facce di due dadi?

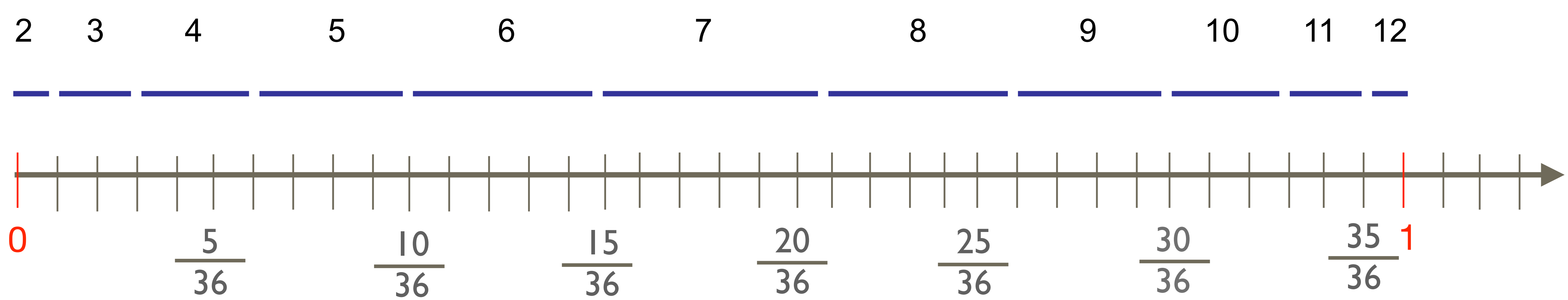
x_i	2	3	4	5	6	7	8	9	10	11	12
$P(x_i)$	1/36	2/36	3/36	4/36	5/36	6/36	5/36	4/36	3/36	2/36	1/36



GENERATORI PER PDF NON UNIFORMI / INVERSIONE

- Se costruisco una successione di segmenti ognuno di lunghezza $p(x_i)$ otterrò un segmento la cui lunghezza totale sarà 1
  - Infatti $\sum_{i=1}^{12} p(x_i) = 1$

x_i	2	3	4	5	6	7	8	9	10	11	12
$p(x_i)$	1/36	2/36	3/36	4/36	5/36	6/36	5/36	4/36	3/36	2/36	1/36

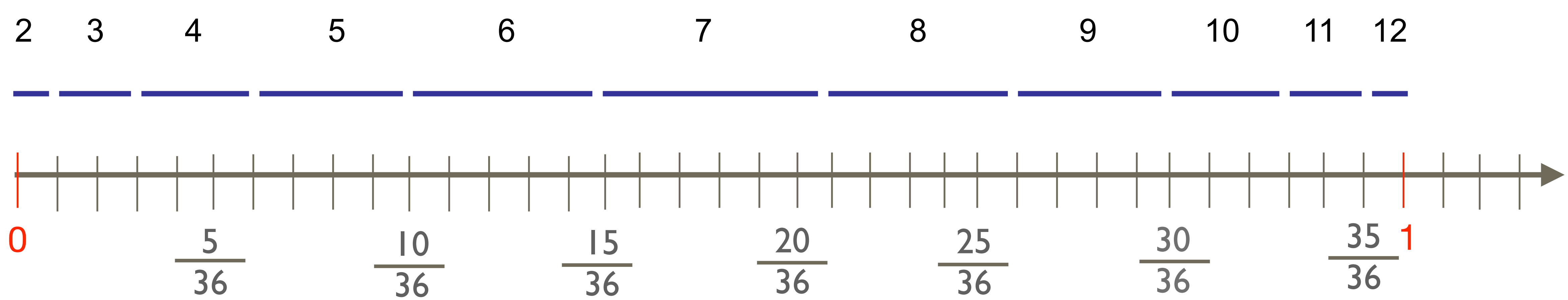


Lunghezza totale =1

Generatori per PDF non uniformi / Inversione

- Se estraggo un numero casuale η_i uniformemente tra 0 e 1 questo cadrà in un sotto-segmento con probabilità proporzionale alla lunghezza del sotto-segmento.
- Quindi, ***gli esiti x_i***, estratti sulla base dell'intervallo in cui cade η_i (uniformemente distribuito in [0,1]) ***seguono la distribuzione di probabilità discreta $p(x_i)$***.

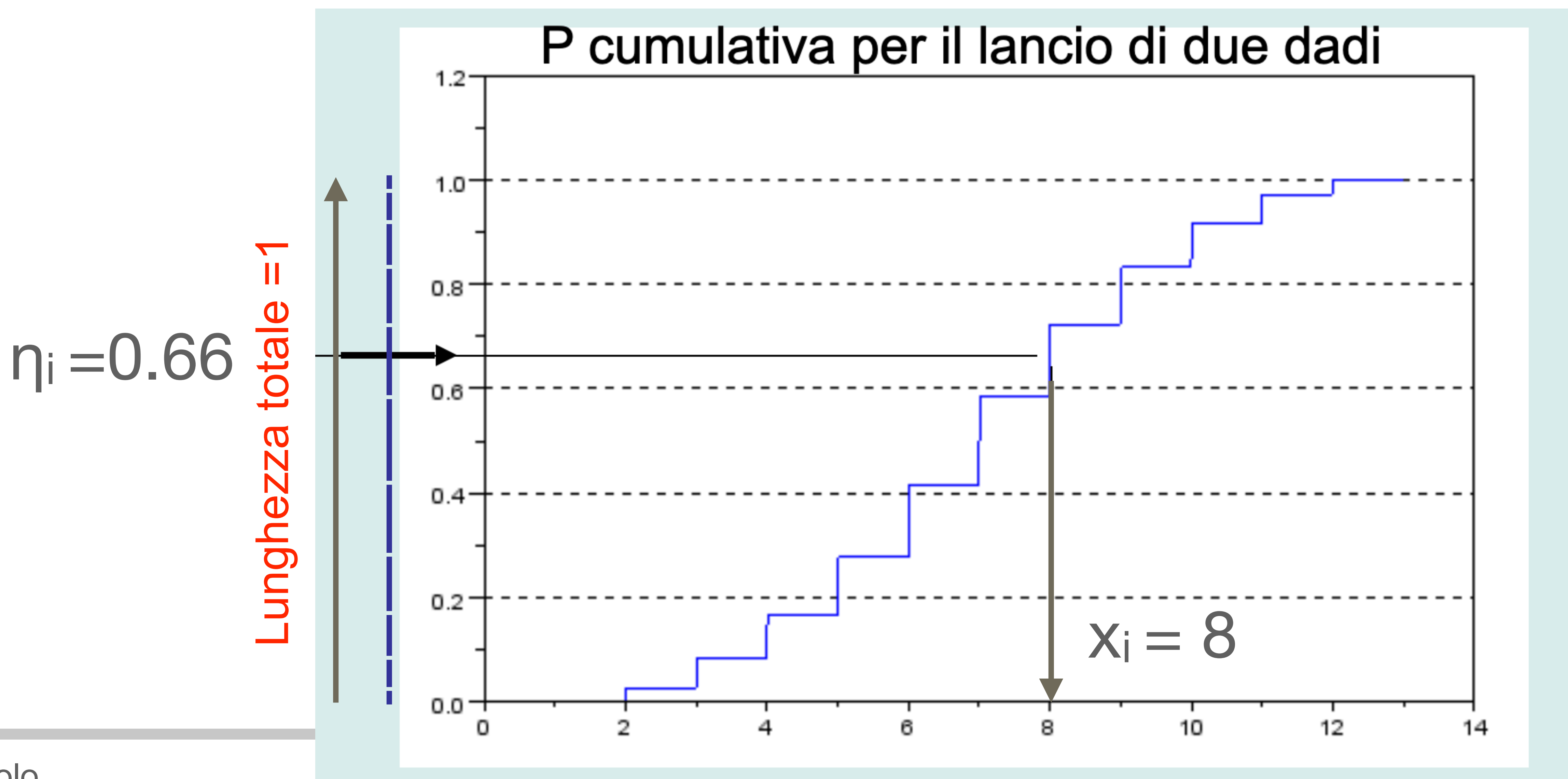
x_i	2	3	4	5	6	7	8	9	10	11	12
$p(x_i)$	1/36	2/36	3/36	4/36	5/36	6/36	5/36	4/36	3/36	2/36	1/36



Lunghezza totale =1

GENERATORI PER PDF NON UNIFORMI / INVERSIONE

- Consideriamo ora la ***probabilità cumulativa***, cosi' definita:
  - la probabilità cumulativa del valore x_k è la probabilità di ottenere un qualsiasi valore compreso tra x_1 a x_k
  - questa è data da: $\longrightarrow$ $P_k = \sum_{i=1}^{k} p(x_i)$
  - ovviamente per k=N avremmo $P(x_N) = 1$



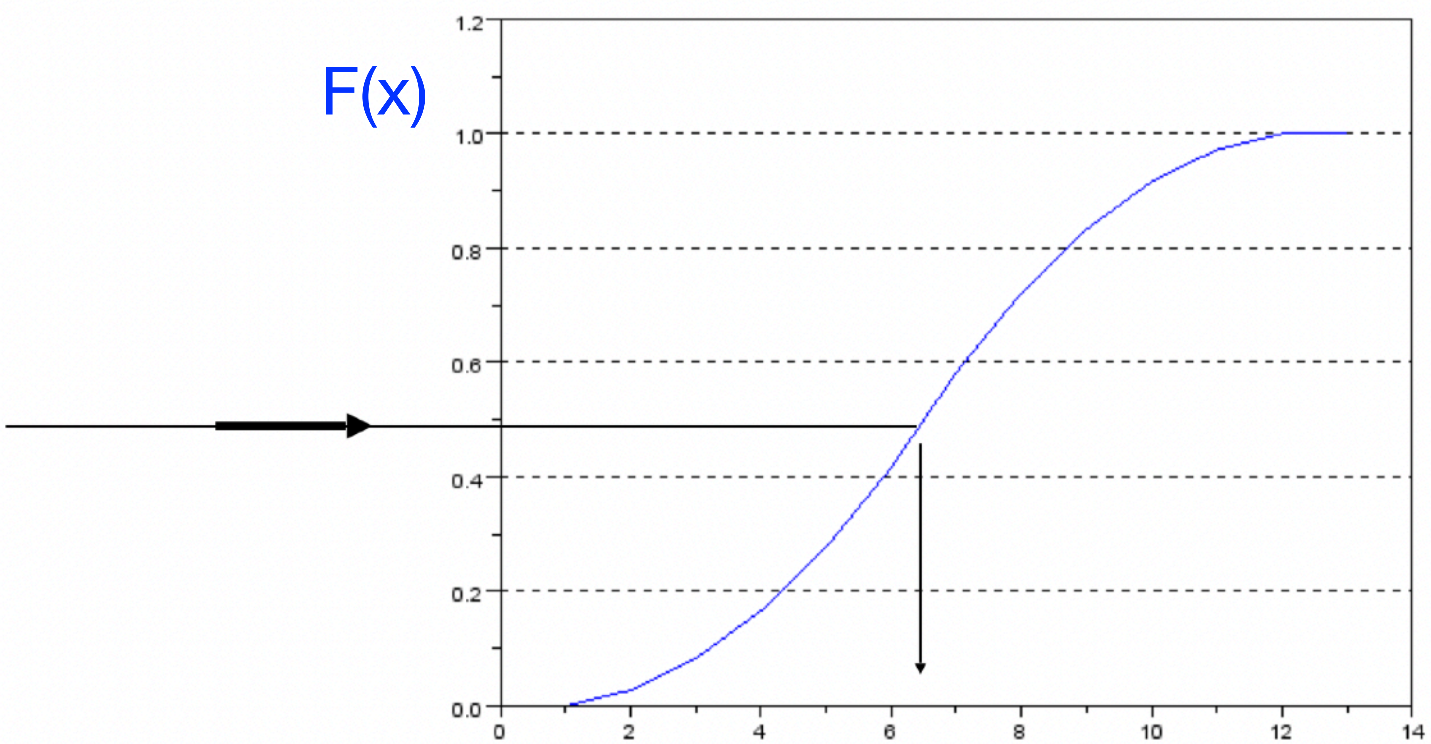
GENERATORI PER PDF NON UNIFORMI / INVERSIONE

Nel caso di variabili aleatorie continue posso utilizzare lo stesso metodo. Supponiamo di avere una distribuzione di probabilità g(x) di una variabile aleatoria continua. La sua distribuzione cumulativa sarà:

voglio generare secondo la pdf ***g(x)***

Cumulativa di ***g(x)*** $$F(x) = \int_{-\infty}^{x} g(t)dt$$

Trattandosi di una distribuzione cumulativa essa sarà una funzione monotona crescente che assume valori in [0,1].



Posso procedere allo stesso modo.

GENERATORI PER PDF NON UNIFORMI / INVERSIONE

- Come fare passo per passo:
  - ***Fase preliminare analitica:***
    - Determinare una primitiva (=cumulativa F) della distribuzione di probabilità che si vuole campionare

$$y = F(x) = \int_{-\infty}^{x} g(t)dt$$

    - Invertire la primitiva:

$$h(y) = F^{-1}(y)$$

  - ***Fase numerica di estrazione dei valori***
    - Estrarre un numero casuale y in maniera uniforme in (0,1)
    - Calcolare x = h(y) a partire da y utilizzando l'inversa della primitiva

GENERATORI PER PDF NON UNIFORMI / INVERSIONE

ESEMPIO

Consideriamo la distribuzione di probabilità:

$$g(x) = 2x \qquad [0,1]$$

Definita nell'intervallo [0,1].
Questa possiede tutti i requisiti di una distribuzione di probabilità.

- Definita positiva
- Integrale uguale a 1
- Continua e derivabile.

Calcolarne una sua primitiva è semplice cosi come è semplice invertire quest'ultima.

Primitiva $y=F(x)= x^2$

$$x = F^{-1}(y) = \sqrt{y}$$

Inversa della primitiva

GENERATORI PER PDF NON UNIFORMI / INVERSIONE

- Si determina analiticamente l’inversa della funzione cumulativa.
- Si estraggono numeri casuali in maniera uniforme tra 0 e 1
- Si utilizza la funzione inversa per ottenere il valore casuale voluto.

```
//
//  estraggo valori secondo una distribuzione data
// usando il metodo dell'inversione
//
// distribuzione di probabilità o di frequenze
//
double f(double x) {
     return 2.*x;
}
double inv(double y) {
     return sqrt(y);
}
void Inversione(){
// creo un "oggetto" generatore di numeri random
   TRandom R(12345);
   int N=100000;
// creo un oggetto file lo collego al file di nome inversione.root
// e gli dico di crearlo (ricrearlo) quindi è un output
   TFile f("inversione.root","RECREATE");
// creo un oggetto histogramma
   TH1D * h = new TH1D ("h","Istogramma delle Frequenze",80,0,1.);
   for (int i=0; i<N ; i++) {
      double x,y;
//estraggo numeri in maniera uniforme tra 0 e 1
      y=R.Uniform();
// uso l'inversa della primitiva
      x=inv(y);

// Riempio l'oggetto histogramma con i numeri random generati
      h->Fill(x);
   }
// scrivo l'oggetto histogramma nel file
   h->Write();
// chiudo il file attraverso il metodo dell'oggetto di tipo file.
   f.Close();
}
```

Macro: GenRandom_MetodoInv.C

QUALI MACRO ?

- hitormiss.C ==>>> integrazione di funzione di variabile reale con il metodo MC della reiezione
  - Nota: non si dimostra va validità della stima dell’errore -> TO DO
- Integrale_MonteCarlo.C ==>>> integrazione di funzione di variabile reale con il metodo MC e verifica della bontà della stima dell’errore
- hitormiss2.C ==>> generatore di numeri pseudo-casuali con il metodo della reiezione.
- GenRandom_MetodoInv.C —-> funzione genInv() illustra un esempio di un generatore di numeri pseudocasuali implementato con il metodo dell’inversione (con funzione cumulativa invertibile analiticamente)
- eserciziL3.C ——>>> generatore di numeri pseudocasuali da distribuzione sperimentale ??

FINE