

Metodi Computazionali della Fisica

**Tecniche Monte Carlo
per l'integrazione numerica.**

Cos'è un metodo Monte Carlo ?

1953, Nicolaus Metropolis :

“Il metodo Monte Carlo si riferisce a qualsiasi metodo computazionale che usa i numeri aleatori come parte essenziale dell’algoritmo “

In genere il metodo Monte Carlo si prefigge di calcolare integrali multidimensionali con metodi stocastici.

Primo problema: generare dei numeri aleatori secondo la distribuzione voluta.

Come generare numeri aleatori ?

- Usare un qualche sistema caotico (Palline nel cesto – Lotto,)
- Usare un processo che è per sua natura aleatorio
 - Decadimento radioattivo
 - Rumore termico
 - Arrivo dei raggi cosmici dallo spazio
 - Arrivo dei treni

Dal punto di vista algoritmico non esiste un modo per generare numeri aleatori: si parla di **numeri pseudo-aleatori**.

Generatori di numeri Pseudo-Aleatori

- Il generatore di numeri casuali che può essere ottenuto da un computer segue, in genere, una **distribuzione uniforme nell'intervallo [0,1]**
- La maggior parte dei generatori di numeri casuali hanno due cose in comune:
 - L'utilizzo di numeri primi grandi
 - L'uso della funzione modulo
- L'algoritmo genera un numero intero tra 0 e M, che poi si mappa in un numero tra zero e uno usando:

$$X_n = I_n / M$$

Metodo congruente lineare (Lehmer, 1948)

Molti generatori di numeri casuali sono del tipo

$$I_{n+1} = (aI_n + c) \bmod(m)$$

- $a, c \geq 0, m > I_0, a, c$
- **Vantaggio :**
 - Molto veloce
- **Problema :**
 - Una scelta poco oculata delle costanti può portare ad un ciclo piccolo
 - La relazione si ripeterà dopo un periodo non più grande di m (in genere intorno a $m/4$)
 - Compilatore C **RAND_MAX : $m = 32767$**

Altri algoritmi

Si può migliorare il comportamento del generatore di numeri casuali utilizzando per esempio il seguente algoritmo

$$I_n = (a \times I_{n-1} + b \times I_{n-2}) \bmod(m)$$

Richiede due semi iniziali e può avere un periodo più grande di m

***Esempio: Il generatore
RANMAR***

Disponibile nella libreria del CERN

–Richiede 103 semi iniziali

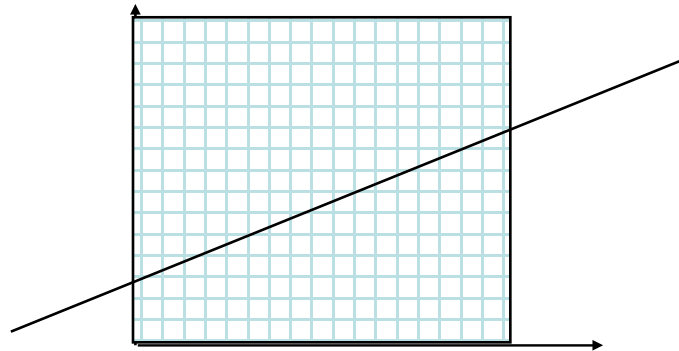
–Periodo : intorno a 10^4

Proprietà dei numeri pseudo-aleatori

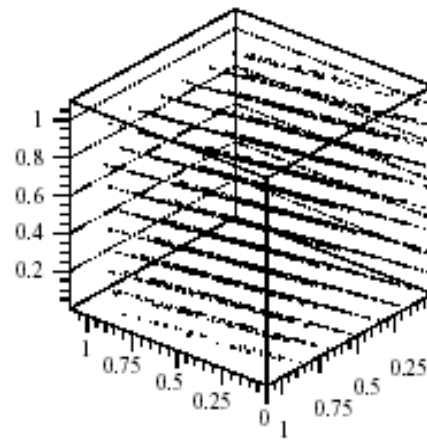
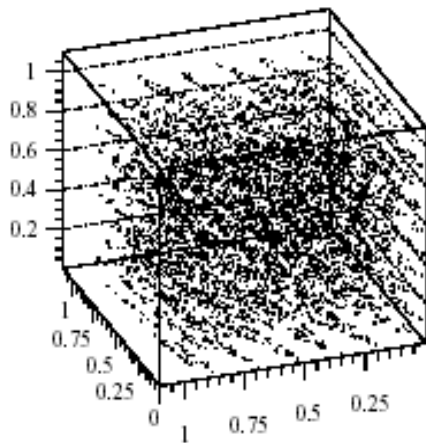
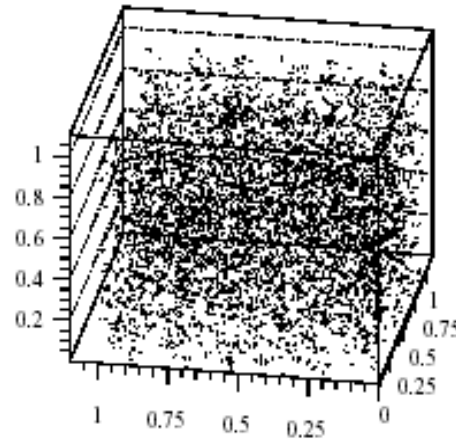
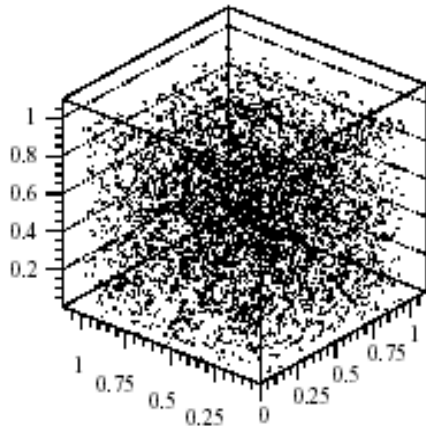
1. Questi algoritmi generano *successioni periodiche* (quindi non aleatorie). Pensiamo solo a cosa accade se ad un certo punto il numero generato coincide con il seme iniziale. Si ha quindi a disposizione un **numero finito di punti**. Inoltre la generazione di un numero dipende dal valore di quelli precedenti (*correlazione*).

Proprietà dei numeri pseudo-aleatori

2. La restrizione ad un numero finito di numeri comporta dei problemi negli spazi ad alte dimensioni. Molti punti generati risultano infatti coplanari. Per dimensione 10 e per numeri aleatori a 32-bit, si avranno coperti solo 126 iper-piani nello spazio a 10 dimensioni.



Distribuzione 3D da RANDU (IBM)



**I problemi si
notano quando si
guarda lungo
l'angolo giusto.**

Funzioni disponibili nei compilatori

- drand48()** – fornisce un numero pseudo-aleatorio nell'intervallo $(0,1)$, utilizzando la doppia precisione (double precision).
- Routine abbastanza buona.
 - Può non essere trasportabile in sistemi differenti.

Inizializzare con il seme

- La maggior parte degli algoritmi hanno una sorta di stato iniziale che può essere inizializzato.
- Ciò può essere fatto utilizzando routines di inizializzazione come *srand*, *srandom* o *srand48*.
 - Perché fare ciò ?

Inizializzare con il seme

Due ragioni:

1. Lo stato di default genera sempre la stessa successione. Non certo aleatorio.....

Soluzione: Chiamare il metodo di inizializzazione usando come seed i bit meno significativi dell'orologio di sistema.

2. Il processo deve sempre essere deterministico per essere ripetibile (stesso seed di partenza).

Trasformazioni di numeri aleatori

- Molte librerie di computer forniscono numeri pseudo-aleatori che assumo valori in un intervallo ben definito.
- Occorre quindi trasformare questi numeri in modo che assumano valori nel range desideratoto .
- Due esempi comuni:
 - Interi aleatori da zero ad un certo numero massimo M .
 - Numeri aleatori in singola o doppia precisione che assumano valori tra zero ed uno.

Campionamento di punti in aree non rettangolari

- In 2D, si possono volere *punti* distribuiti in maniera aleatoria e uniforme in una certa regione.
 - Quadrato – determinare le componenti x e y in maniera indipendente.
 - Rettangolo - ???
 - Cerchio - ???
 - *Modo sbagliato* – determinare r e θ in maniera indipendente.

Metodo giusto ?

Tecniche Monte Carlo

- **Problema:** Qual'è la probabilità che la somma di 10 lanci di un dado dia esattamente 32 ?
- **Modo esatto.** Contare tutti i modi possibili per fare 32 con 10 dadi (eventi indipendenti).
- **Modo Approssimato (pigro).** Simulare il lancio di 10 dadi (diciamo 500 volte), contare le volte che la somma fornisce 32, e dividere per 500.
- **Il metodo pigro può avvicinarsi al risultato esatto in modo molto veloce.**

Tecniche Monte Carlo

Possibili applicazioni

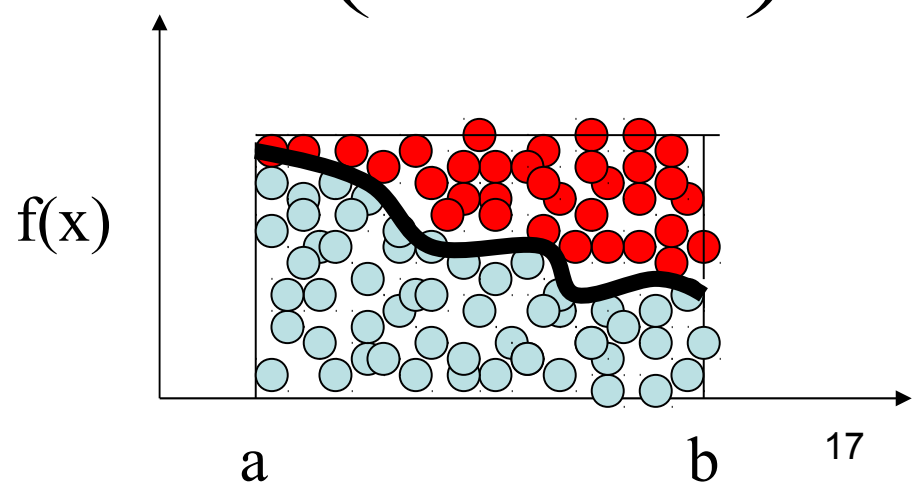
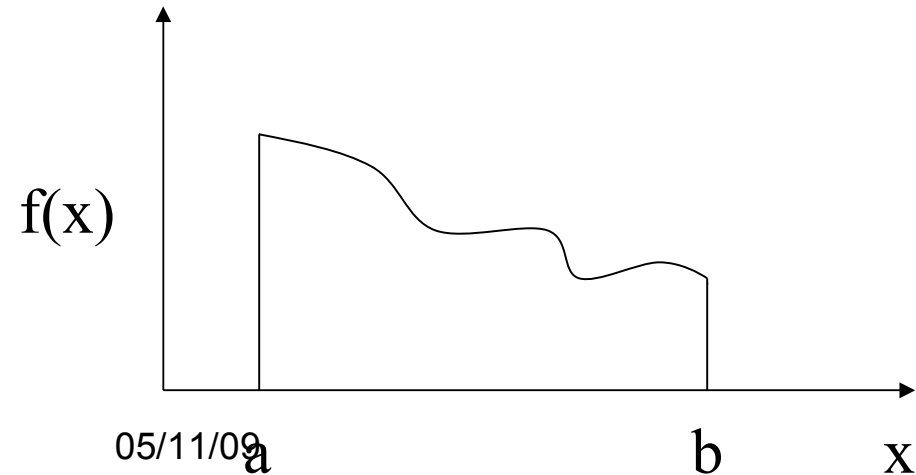
- Integrazione di funzioni complesse
- Simulazione di sistemi complessi
- Computer graphics - Rendering.
- Fenomeni fisici – trasporto radioattivo
- Simulazione del Bingo
- Comunicazioni – tassi di errore nella trasmissione

Integrazione:

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

- Metodo 1: Integrazione analitica
- Metodo 2: Metodi di Quadratura (trapezio, Simpson)
- **Metodo 3:** Metodo Monte Carlo – campionare in maniera aleatoria l'area compresa tra $a < x < b$ e $0 < y < \max(f(x))$

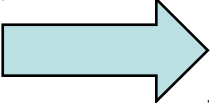
$$\int_a^b f(x) dx \approx \max(f(x))(b-a) \left(\frac{\# \text{ light blue circles}}{\# \text{ light blue circles} + \# \text{ red circles}} \right)$$



Metodo hit and miss: $\int_a^b f(x)dx$

In maniera intuitiva:

$$\int_a^b f(x)dx \approx \max(f(x))(b-a) \left(\frac{\# \text{ } \text{blue circle}}{\# \text{ } \text{blue circle} + \# \text{ } \text{red circle}} \right)$$

 $\text{Area}_{\text{Box}} \bullet \Pr\{\bar{y} \leq f(\bar{x})\}$

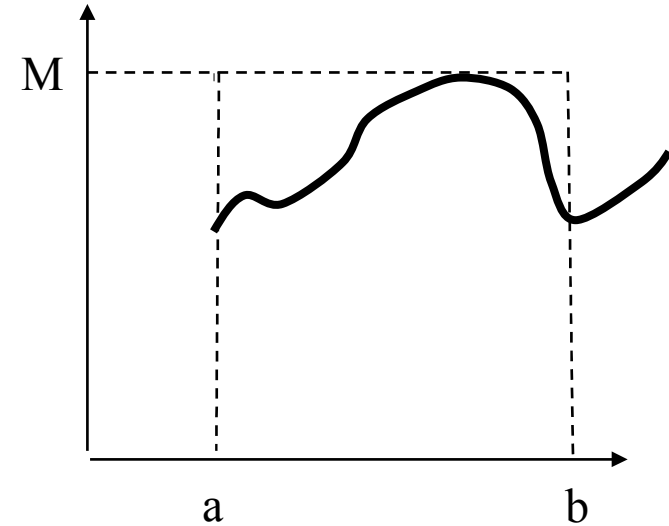
Notare che

$$\Pr\{\bar{y} \leq f(\bar{x})\} = \Pr\{(\bar{x}, \bar{y}) \in \text{Area delimitata da } f(x)\}$$

Algoritmo hit and miss

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

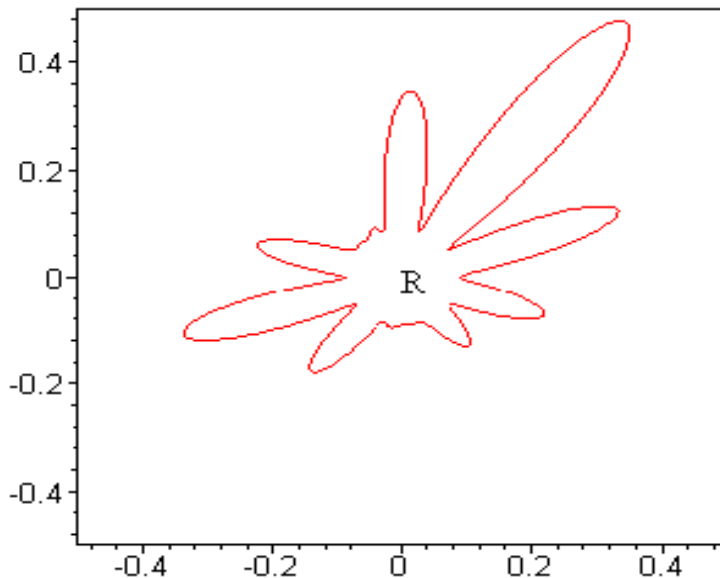
1. Porre $i=1$;
2. Scegliere due numeri aleatori, A_i e A_{i+1} tra 0 e 1;
3. Porre $x=(a+(b-a) A_i)$ e $y=M A_{i+1}$;
4. Se $f(x) > y$ porre $p_i=1$ (hit);
altrimenti $p_i=0$ (miss);
5. Quando $i=N$ (numero massimo d'iterazioni) stop;
altrimenti porre $i=i+1$ e tornare al punto 2.



$$\rightarrow I \approx \left(\frac{1}{N} \sum_{i=1}^N p_i \right) (b-a) M$$

Aree di regioni a forma complessa

- Il dominio su cui si integra la funzione f può essere di forma molto complessa.
- Come si possono campionare dei punti pesati da una $f(x)$ in una griglia che copre la regione R ?



Esempio :

Supponiamo di voler calcolare la distanza quadratica media dall' origine sul dominio R raffigurato a sinistra

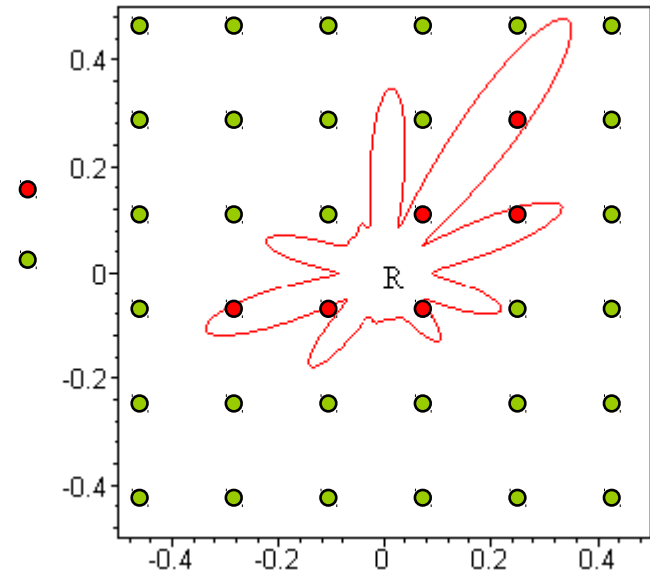
$$\langle r^2 \rangle = \frac{\iint_R (x^2 + y^2) dx dy}{\iint_R dx dy}$$

$$f(x, y) = (x^2 + y^2)$$

Integrazione su domini più semplici

$$\langle r^2 \rangle = \frac{\int_{-0.5}^{+0.5} dx \int_{-0.5}^{+0.5} dy (x^2 + y^2) s(x, y)}{\int_{-0.5}^{+0.5} dx \int_{-0.5}^{+0.5} dy s(x, y)}$$

$s = \begin{cases} 1 & \text{inside R} \\ 0 & \text{outside R} \end{cases}$
 $s(x, y)$: funzione caratteristica di R.



E' chiaro che la griglia deve essere sufficientemente fine !

Altro modo: utilizzare il metodo hit and miss con

$$f(x, y) = s(x, y)(x^2 + y^2)$$

Porre $x = (a + (b - a) A_i)$, $y = (a + (b - a) A_{i+1})$ e $z = M A_{i+2}$

Se $f(x, y) > z$ porre $p_i = 1$ (hit); altrimenti $p_i = 0$ (miss);

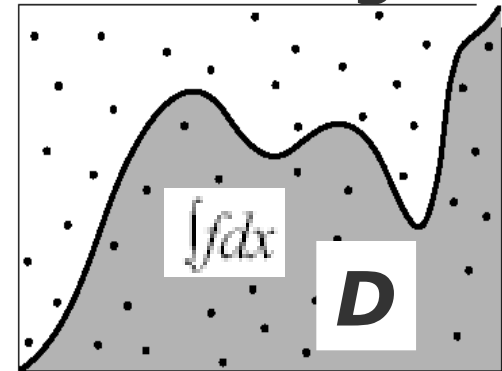
$$I \approx \left(\frac{1}{N} \sum_{i=1}^N p_i \right) (b - a)^2 M$$

Integrazione Monte-Carlo

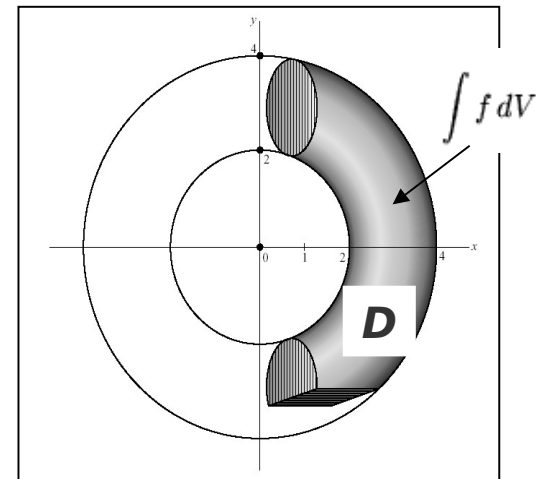
- Integrare una funzione su un dominio complicato
 - D : dominio complicato.
 - D' : dominio semplice che include D .
- Scegliere dei punti aleatori su D' :
- Contare N : punti su D
 N' : punti su D'

$$\frac{\text{Volume}_D}{\text{Volume}_{D'}} = \frac{N}{N'}$$

D' : rettangolare



D' : cerchio



Stima di π usando Monte Carlo

- La probabilità che un punto aleatorio si trovi dentro il cerchio unitario vale:

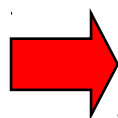
$$P(x^2 + y^2 < 1) = \frac{A_{circle}}{A_{square}} = \frac{\pi}{4}$$

- Se prendiamo un numero aleatorio N volte ed M di queste il punto si trova all'interno del cerchio unitario:

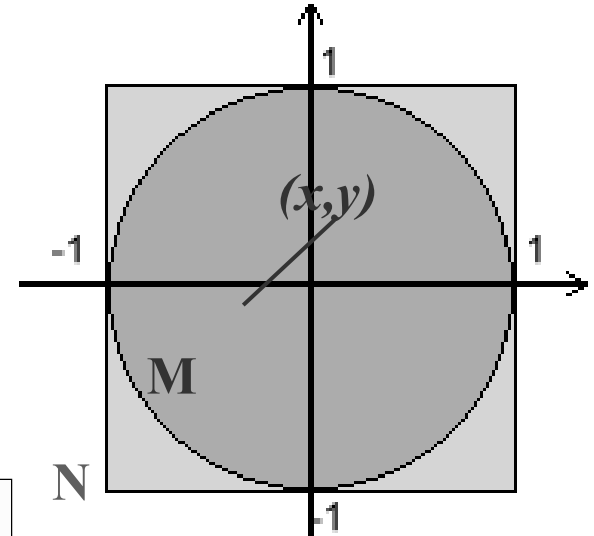
$$P^0(x^2 + y^2 < 1) = \frac{M}{N}$$

- Se N diventa molto grande,

$$P^0 \xrightarrow{N \rightarrow \infty} P$$



$$\frac{4M}{N} \xrightarrow{N \rightarrow \infty} \pi$$



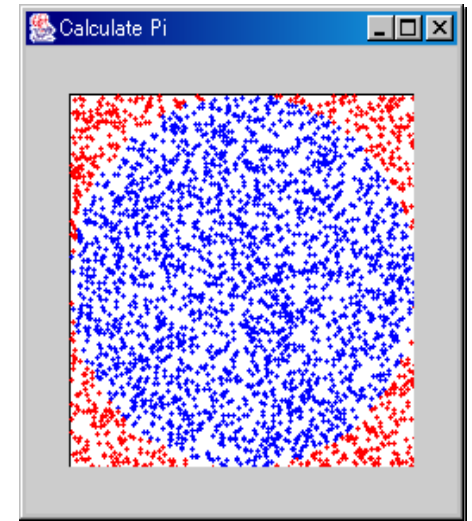
Pseudocodice per la stima di π

```
double x, y, pi;  
const long m_nMaxSamples = 100000000;  
long count=0;  
for (long k=0; k<m_nMaxSamples; k++) {  
    x=2.0*drand48() – 1.0; // Trasforma nel range [-1,1]  
    y=2.0*drand48() – 1.0;  
    if (x*x+y*y<=1.0) count++;  
}  
pi=4.0 * (double) count / (double)m_nMaxSamples;
```


Stima di π con il Monte Carlo

- Risultati:

- | | | |
|-------|------------|--------------|
| – N = | 10,000 | Pi= 3.1388 |
| – N = | 100,000 | Pi= 3.1452 |
| – N = | 1,000,000 | Pi= 3.14164 |
| – N = | 10,000,000 | Pi= 3.141603 |
| – | ... | |



Integrale come valor medio

$$I = \int_a^b f(x)dx = \frac{\int_a^b f(x)dx}{\int_a^b dx} \times \int_a^b dx = \langle f \rangle_{\mu_L} \times \mu_L(a, b)$$

dove $\langle f \rangle_{\mu_L}$ Valor medio di $f(x)$ rispetto alla misura uniforme

$\mu_L(a, b) = (b - a)$ Misura di Lebesgue dell'insieme su cui si integra

Legge forte dei grandi numeri

Data una successione di variabili aleatorie indipendenti (V.A.I.) x_i , anche i valori $f(x_i)$ formano una successione di V.A.I. e vale

$$\Pr \left\{ \lim_{N \rightarrow \infty} \frac{1}{N} \sum_i f(x_i) = \langle f \rangle \equiv \frac{I}{b-a} \right\} = 1$$

La stima del valor medio converge (in probabilità) all'integrale I per un numero sufficientemente grande di campionamenti.

Errore Monte Carlo

I metodi Monte Carlo utilizzati sino ad ora si basano su una somma di punti aleatori, $r=(x,y,\dots,z)$ generati in maniera indipendente l'uno dall'altro. Si parla di una successione di *variabili aleatorie indipendenti*

Dalla teoria delle probabilità si può mostrare che per una successione di *variabili aleatorie indipendenti* (V.A.I.) l'errore nella stima dell'integrale decresce con il numero di campioni N come

$$\varepsilon \propto \frac{1}{\sqrt{N}}$$

indipendentemente dalla dimensione d !!!

Teorema del Limite Centrale

Più precisamente, per una successione f_i di V.A.I. e per N sufficientemente grande, il valor medio stimato $\langle f \rangle = (1/N) \sum f_i$

segue una distribuzione Gaussiana con valore di aspettazione, $E(f)$ e varianza $\sigma^2 = \text{var}(f)/N$ dove

$$\text{var}(f) = E(f^2) - E(f)^2.$$

σ e' detta deviazione standard

Efficienza del Metodo Monte Carlo

Si e' visto che $\sigma \sim \frac{1}{\sqrt{N}}$

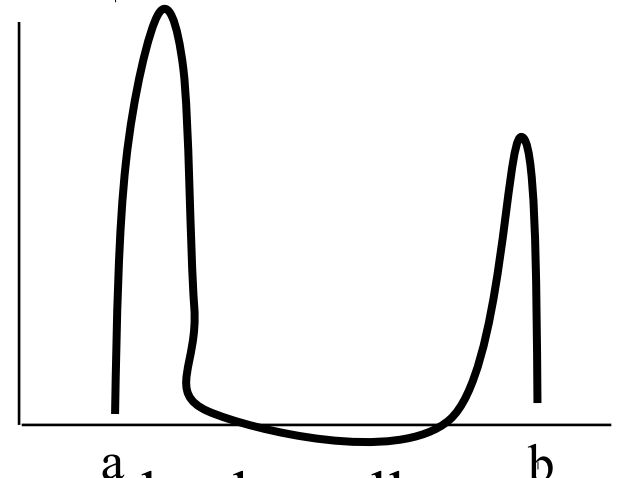
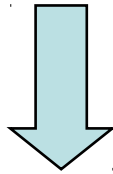
Ricordiamo i metodi deterministici:

- Regola del trapezio $\text{err} \sim N^{-2/d}$
- Regola di Simpson $\text{err} \sim N^{-4/d}$
- Regola di Gauss ad m punti $\text{err} \sim N^{-(2m-1)/d}$

Il metodo Monte Carlo diventa migliore per dimensioni d sufficientemente grandi !

Importance sampling

Grandi variazioni nel valore di $f(x)$ comportano una forte incertezza nella stima finale dell'integrale.



Idea: effettuare un cambio di variabile in modo che nelle nuove variabili la funzione integranda sia *'meno variabile'*

In pratica:

$$I = \int_a^b f(x) dx = \int_a^b \frac{f(x)}{g(x)} g(x) dx; \quad g(x) \geq 0, \quad \int_a^b g(x) dx = 1$$

Importance sampling

$$I = \int_a^b f(x)dx = \int_a^b \frac{f(x)}{g(x)}g(x)dx; \quad g(x) \geq 0, \quad \int_a^b g(x)dx = 1$$

N.B. Ora la funzione integranda è $f(x)/g(x)$ mentre i punti aleatori dovranno essere scelti non più rispetto alla misura uniforme (Lebesgue) dx ma rispetto alla misura di probabilità $g(x)dx = d\mu(x)$ (misura di Stieltes)

$$I = \int_a^b f(x)dx = \int_a^b \frac{f(x)}{g(x)}g(x)dx =$$

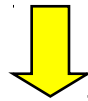
Notare:

$$\frac{\int_a^b \frac{f(x)}{g(x)}g(x)dx}{\int_a^b g(x)dx} \times \int_a^b g(x)dx = \left\langle \frac{f}{g} \right\rangle_g \mu_g(a,b)$$

Importance sampling

Algoritmo:

1. Porre $i=1$
2. Scegliere una v.a. X con densità di probabilità $g(X)$
3. Porre $h_i = f(X) / g(X)$
4. Se $i=N$ dove N è il numero d'iterazioni richieste \rightarrow STOP
altrimenti porre $i=i+1$ e torna al punto 2



$$\frac{1}{N} \left(\sum_{i=1}^N h_i \right) \xrightarrow{N \rightarrow \infty} I$$

Note:

1. $g(X)$ deve essere scelta in modo che $f(X) / g(X)$ sia il più possibile **costante** (piccola varianza)
2. E' una tecnica generale che si applica ad ogni dimensione
3. Richiede la generazione di numeri aleatori con densità di probabilità $g(X)$ (problema non banale)

Importance sampling

Quale sarebbe la $g(x)$ ottimale ?

E' quella che rende il rapporto $f(x)/g(x)$ il più possibile indipendente da x . La scelta più ovvia sarebbe porre $g(x)=f(x)$

$$I = \int_a^b f(x)dx = \int_a^b \frac{f(x)}{g(x)} g(x)dx = \left\langle \frac{f}{g} \right\rangle_g \mu_g(a, b)$$

Però..

$$\stackrel{g=f}{=} \mu_f(a, b) = \int_a^b f(x)dx = I$$

e si ritorna al punto di partenza

La scelta ottimale $g(x)=f(x)$ implicherebbe la conoscenza dell'integrale che vogliamo calcolare.....

Problema da risolvere in Meccanica Statistica

Sia \mathbf{X} un punto (stato) dello spazio delle fasi S e sia O un'osservabile del sistema: $O \rightarrow O(\mathbf{X})$

Quantità da stimare: $\langle O \rangle = \frac{1}{Z} \int_S O(\mathbf{X}) \rho(\mathbf{X}) d\mathbf{X}$


dove $Z = \int_S \rho(\mathbf{X}) d\mathbf{X}$ e $\rho(\mathbf{X}) = \exp(-H(\mathbf{X}) / k_b T)$

Esempi:

- Gas o fluidi $\rightarrow M$ particelle in volume V , $\mathbf{X} = (\vec{r}_1, \vec{r}_2, \dots, \vec{r}_M)$
- Ferromagneti \rightarrow Reticolo ad M siti, $\mathbf{X} = (\sigma_1, \sigma_2, \dots, \sigma_M)$, σ : spins

Stima Monte Carlo di $\langle O \rangle$ (metodo inefficiente)

Generare configurazioni $\mathbf{X}^{(i)} = (\vec{r}^{(i)}_1, \vec{r}^{(i)}_2, \dots, \vec{r}^{(i)}_M)$ dove ogni posizione $\vec{r}^{(i)}_j$ è ottenuta scegliendo le tre coordinate $(x_j^{(i)}, y_j^{(i)}, z_j^{(i)})$ in maniera aleatoria (cioè generando 3 numeri aleatori)

 $\langle O \rangle = \frac{\sum_{i=1}^T O(\mathbf{X}^{(i)}) e^{-\beta H(\mathbf{X}^{(i)})}}{\sum_{i=1}^T e^{-\beta H(\mathbf{X}^{(i)})}}$

Metodo corretto ma estremamente **inefficiente**

Per un valore di $\beta \Rightarrow \langle E \rangle = E_m(\beta)$ ed il contributo rilevante alla somma viene dalle configurazioni aventi energia E con $E_m - \Delta E \leq E \leq E_m + \Delta E$

Del resto $\Delta E \sim \sigma(E)$ e $\sigma(E) \xrightarrow{N \rightarrow \infty} 0$

Occorre utilizzare il metodo di **importance sampling** !!!!

Problema: Stimare

$$\langle O \rangle = \frac{1}{Z} \sum_{\{\mathbf{x}\}} O(\mathbf{x}) e^{-\beta H(\mathbf{x})}$$

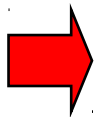
Per una π arbitraria

si può scrivere:

$$\langle O \rangle = \frac{\sum_{\{\mathbf{x}\}} O(\mathbf{x}) \frac{e^{-\beta H(\mathbf{x})}}{\pi(\mathbf{x})} \pi(\mathbf{x})}{\sum_{\{\mathbf{x}\}} \frac{e^{-\beta H(\mathbf{x})}}{\pi(\mathbf{x})} \pi(\mathbf{x})} = \frac{\left\langle O \frac{e^{-\beta H}}{\pi} \right\rangle_{\pi}}{\left\langle \frac{e^{-\beta H}}{\pi} \right\rangle_{\pi}}$$

Idea Generare T ‘punti’ $\mathbf{x}^{(i)}$ con probabilità

$$\pi(\mathbf{x}^{(i)}) \propto Z^{-1} \exp(-\beta H(\mathbf{x}^{(i)}))$$



$$\langle O \rangle = \frac{\left\langle O \frac{e^{-\beta H}}{\pi} \right\rangle_{\pi}}{\left\langle \frac{e^{-\beta H}}{\pi} \right\rangle_{\pi}} \cong \frac{1}{T} \sum_{i=1}^T O(\mathbf{x}^{(i)})$$

Monte Carlo dinamici

Problema: Trovare un modo per generare configurazioni $\mathbf{X}^{(i)}$ **indipendenti** secondo la probabilità assegnata π .

I metodi precedenti (**Monte Carlo statici**) non sono efficienti.

Vi sono metodi semplici che generano configurazioni $\mathbf{X}^{(i)}$ **correlate** secondo una π assegnata.

Si parla di **Monte Carlo dinamici** che sono basati sulla teoria delle **Catene di Markov**.

Monte Carlo dinamici e catene di Markov

S: spazio delle configurazioni $\mathbf{X}^{(i)}$,
 π misura di probabilità da cui campionare

Idea: *Produrre una traiettoria (stocastica) nello spazio S che produca punti $\mathbf{X}^{(i)}$ correlati ma che per tempi lunghi sono distribuiti secondo la misura di probabilità π .*

In questo modo se con $\{ \mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \mathbf{X}^{(3)}, \dots, \mathbf{X}^{(T)} \}$ indichiamo la traiettoria generata si avrà :

$$\langle O \rangle \equiv \frac{1}{T} \sum_{i=1}^T O(\mathbf{X}^{(i)})$$

Il processo stocastico da usare è noto come **Catena di Markov** introdotto in Fisica da Metropolis è oggi uno dei più potenti ed utilizzati algoritmi Monte Carlo.

Come generare configurazioni secondo una distribuzione π desiderata ?

Sia $\Pr\{X_{t+1} | X_t\}$ la probabilità di essere in X_{t+1} al tempo $t+1$ dato che la traiettoria si trovava in X_t al tempo t . Si parla di **probabilità di transizione**.

Def. **Catena di Markov**

Una catena di Markov sullo spazio S è una successione di V.A. $X_0, X_1, X_2, \dots, X_t$, a valori in S , $i_0, i_1, i_2, \dots, i_t$, tale che

$$\Pr\{X_{t+1} = j | X_t = i_t, X_{t-1} = i_{t-1}, \dots, X_0 = i_0\} = \Pr\{X_{t+1} = j | X_t = i_t\}$$

Processo che dipende solo dalla probabilità condizionata a due punti. Importa solo il punto precedente.

Matrice di transizione

La probabilità di transizione tra due stati si scrive anche

$$\Pr\{X_{t+1} = j | X_t = i\} \equiv P_{i,j}(t) \quad \text{In generale } P \text{ dipende da } t$$

Def. Una catena di Markov si dice **tempo omogenea** se

$$P_{ij}(t) = P_{ij} \quad \forall i, j$$

Catena di Markov

La probabilità di una certa traiettoria $\{i_0, i_1, i_2, \dots, i_t\}$ sarà quindi

$$\begin{aligned} \Pr\{X_0 = i_0, X_1 = i_1, X_2 = i_2, \dots, X_t = i_t\} = \\ = \Pr\{X_0 = i_0\} \Pr\{X_1 = i_1 | X_0 = i_0\} \Pr\{X_2 = i_2 | X_1 = i_1\} \dots \Pr\{X_t = i_t | X_{t-1} = i_{t-1}\} \end{aligned}$$

dove $\Pr(X_0 = i_0)$: probabilità dello stato iniziale

Noti $\Pr(X_0 = i_0)$ e $\Pr\{X_t = i_t | X_{t-1} = i_{t-1}\}$

La dinamica stocastica di Markov è completamente definita.

Come costruire le matrici di Markov assegnata una distribuzione π ?

Occorre costruire una matrice P_{ij} che abbia le seguenti proprietà:

- (a) **Irriducibilità** : Per ogni coppia i, j in S esiste un t tale che $P_{ij}^{(t)} > 0$
- (b) **Stazionarietà di π** : Per ogni i, j in S

$$\sum_{j \in S} \pi(j) P_{ji} = \pi(i) \quad \text{il vettore } \pi \text{ è un punto fisso della matrice } P$$

- (b') **Bilancio dettagliato** : Per ogni i, j in S

$$\pi(j) P_{ji} = \pi(i) P_{ij}$$

Si vede subito che (b') è una condizione più forte di (b)

Algoritmi tipo Metropolis

Si prefigge di soddisfare la condizione (b') in maniera generale.

Sia P_{ij}^o la matrice di transizione delle mosse proposte. Essa genera mosse proposte $i \rightarrow j$ che verranno poi accettate o respinte rispettivamente con probabilità a_{ij} e $(1-a_{ij})$.

N.B. Mossa rifiutata significa considerare la transizione nulla $i \rightarrow i$


La matrice di transizione della catena di Markov P_{ij} è così definita:

$$\begin{aligned} i \neq j \quad & P_{ij} = P_{ij}^o a_{ij} \\ i = j \quad & P_{ij} = P_{ii} = 1 - \sum_{j \neq i} P_{ij} = 1 - \sum_{j \neq i} P_{ij}^o a_{ij} \\ & = P_{ii}^o + \sum_{j \neq i} P_{ij}^o (1 - a_{ij}) \end{aligned}$$

Algoritmi tipo Metropolis (cont.)

La condizione b': $\pi(j)P_{ji} = \pi(i)P_{ij}$ diventa

$$P_{ij}^o a_{ij} \pi(i) = P_{ji}^o a_{ji} \pi(j) \quad \text{per } i \neq j.$$


$$\frac{a_{ij}}{a_{ji}} = \frac{\pi(j)}{\pi(i)} \frac{P_{ji}^o}{P_{ij}^o} \quad \text{per } i \neq j.$$

Basta porre: $a_{ij} = F\left(\frac{\pi(j)}{\pi(i)} \frac{P_{ji}^o}{P_{ij}^o}\right)$ dove $F:[0,\infty] \rightarrow [0,1]; \frac{F(z)}{F(1/z)} = z$

$$a_{ij} = F\left(\frac{\pi(j)}{\pi(i)} \frac{P_{ji}^o}{P_{ij}^o}\right) \quad \text{dove } F:[0,\infty] \rightarrow [0,1]; \quad \frac{F(z)}{F(1/z)} = z$$

Scelte molto utilizzate:

Metropolis $F(z) = \min(1, z)$

Heat bath: $F(z) = \frac{z}{1+z}$

In molti casi $P_{ij}^o = P_{ji}^o$ **condizione di reversibilità**



$$a_{ij} = F\left(\frac{\pi(j)}{\pi(i)}\right)$$

$$a_{ij} = F\left(\frac{\pi(j)}{\pi(i)}\right)$$

In problemi di meccanica statistica:

$$\pi(i) = \frac{1}{Z} e^{-H(i)/k_B T}$$

 $a_{ij} = \min(1, e^{-\Delta H/k_B T})$ dove $\Delta H = H(j) - H(i)$

N.B. Se $\Delta H < 0 \rightarrow$

$$a_{ij} = \min(1, e^{-\Delta H/k_B T}) = 1 \quad (\text{mossa sempre accettata})$$

Algoritmo Metropolis per problemi di Meccanica statistica

Sia $X^{(n)}$ il punto campionato allo step n e sia $E(X^{(n)})$ la sua energia. Si propone il passaggio dalla configurazione $i = X^{(n)}$ alla configurazione X_j con probabilità P_{ij}^0 . Sia $E(X_j)$ l'energia della nuova configurazione.

Si accetta la nuova configurazione con probabilità

$$a_{ij} = \min(1, e^{-\Delta E / k_B T}) \quad \Delta E = E(X^{(n+1)}) - E(X_j)$$

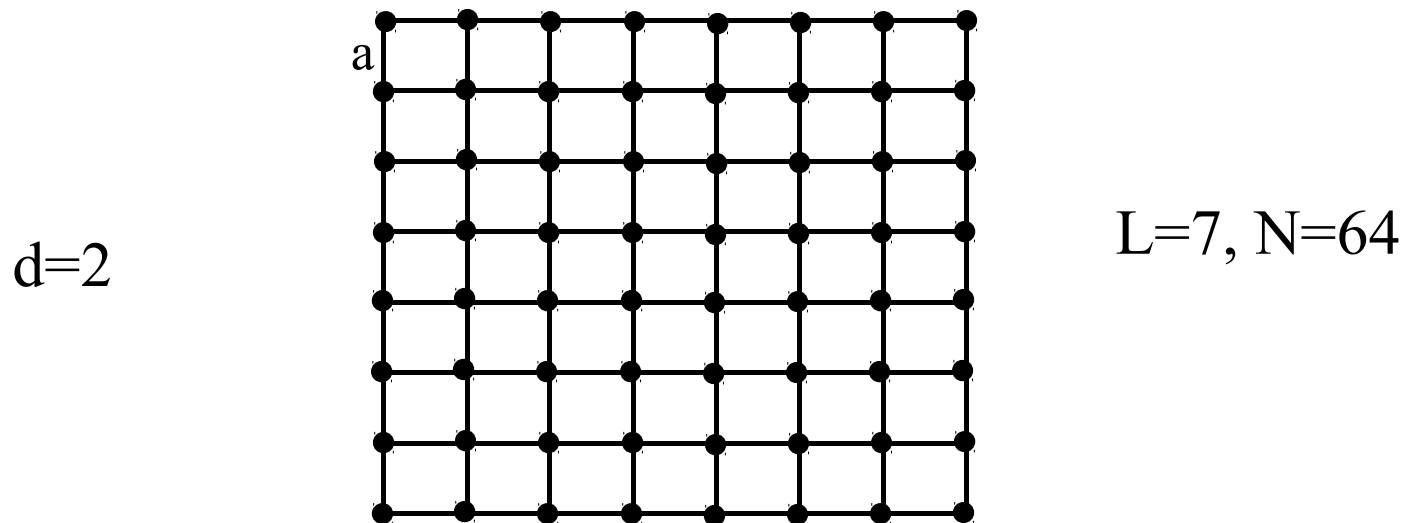
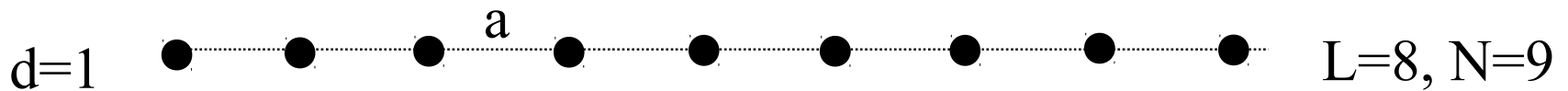


Se $\Delta E < 0$ si accetta la nuova configurazione: $X^{(n+1)} = X_j$

Se $\Delta E > 0$ si sceglie un numero aleatorio r tra 0 e 1 e se $r < e^{-\Delta E / k_B T}$ si accetta altrimenti si rifiuta: $X^{(n+1)} = X^{(n)}$

Modello di Ising

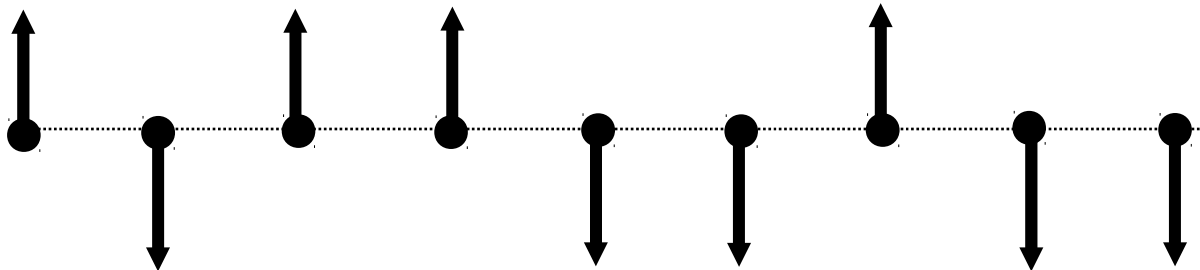
Consideriamo lo spazio discreto Z^d (reticolo ipercubico)
(in $d=1$ catena, in $d=2$ reticolo quadratico, in $d=3$ reticolo cubico).
Sia L la dimensione lineare ed $a=1$ il passo del reticolo.
Se d è la dimensione dell' ipercubo \rightarrow il numero di siti $N=(L+1)^d$



Modello di Ising

Su ogni sito i -esimo del reticolo definiamo una variabile di spin σ_i che può assumere i valori $+1$ e -1 . Si dice che lo spin può assumere lo stato **UP** (freccia in alto) o **DOWN** (freccia in basso).

Esempio di una configurazione con $N = 9$ in $d=1$



$$\{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6, \sigma_7, \sigma_8, \sigma_9\} = \{+1, -1, +1, +1, -1, -1, +1, -1, -1\}$$

Modello di Ising

Per ogni configurazione $\{\sigma\}$, in presenza di un campo magnetico esterno omogeneo h , si assegna un'energia data dall'Hamiltoniano:

$$H(\{\sigma\}) = -\frac{J}{2} \sum_{\langle i,j \rangle} \sigma_i \sigma_j + h \sum_i \sigma_i$$

Dove con $\langle i, j \rangle$ indichiamo la somma sui primi vicini (j -esimi) dell' i -esimo spin. Nel caso di $d=1$ i primi vicini dello spin i -esimo sono lo spin $(i-1)$ -esimo e lo spin $(i+1)$ -esimo.

J : costante di accoppiamento. $J > 0$ per interazione ferromagnetica che favorisce l'allineamento degli spins.

Modello di Ising

Problema agli estremi.

$$? \leftarrow \sigma_1 \quad \sigma_N \rightarrow ?$$

Condizioni periodiche:

$$\sigma_N \leftarrow \sigma_1 \quad \sigma_N \rightarrow \sigma_1$$

d=1 : catena di spins chiusa

d=2 : superficie toroidale

Modello di Ising

Se il sistema di Ising è in equilibrio con un bagno termico a temperatura T allora vale la teoria di Boltzmann-Gibbs secondo la quale la probabilità di una configurazione (stato del sistema) $\{\sigma\}$ è data da:

$$P_N(\{\sigma\} | T, h) = \frac{1}{Z_N(T, h)} e^{-H(\{\sigma\})/k_B T}$$

dove:

$$Z_N(T, h) = \sum_{\{\sigma\}} e^{-H(\{\sigma\})/k_B T}$$

Modello di Ising

L'energia libera (di Gibbs) per spin che descrive il sistema all'equilibrio termodinamico è data da:

$$g(T, h) = \lim_{N \rightarrow \infty} g_N(T, h) = -k_b T \lim_{N \rightarrow \infty} \frac{1}{N} \log Z_N(T, h)$$

Transizione di fase:

In generale un sistema ha una **transizione di fase** a $T=T_c$, se $g(T)$ non è analitica in T_c . Spesso la non analicità di $g(T)$ si traduce in singolarità delle derivate (a partire da un certo ordine m) di $g(T)$. Si dice che la **transizione di fase è di ordine m** .

Transizione para-ferromagnetica

Nel caso di sistemi che presentano una **transizione di fase paramagnetica-ferromagnetica**, le derivate importanti di g da considerare sono

$$m(T, h) \propto \frac{\partial g(T, h)}{\partial h} \quad \text{magnetizzazione per spin}$$

$$\chi_T(h) \propto \frac{\partial^2 g(T, h)}{\partial h^2} \quad \text{suscettività magnetica}$$

$$U(T, h) \propto \frac{\partial g(T, h)}{\partial T} \quad \text{Energia interna}$$

$$C_h(T) \propto \frac{\partial U(T, h)}{\partial T} \propto \frac{\partial^2 g(T, h)}{\partial T^2} \quad \text{Calore specifico (a campo magnetico costante)}$$

Modello di Ising

- Modello più semplice che descrive la transizione dalla fase paramagnetica alla fase ferromagnetica.
- Suggerito da Lenz come soggetto di tesi di dottorato a Ising.
- Ising lo risolve esattamente (soluzione esplicita dell'energia libera di Gibbs) in $d=1$ dove però trova che la transizione si verifica a $T=0$.

Monte Carlo per Modello di Ising

Goal:

Costruire la matrice $P_{ij}(T, h)$ che generi una dinamica stocastica nello spazio delle configurazioni $\{\sigma\}$ tale che, nello stato stazionario, campioni secondo la distribuzione di probabilità assegnata

$$\pi_N(\{\sigma\} | T, h) = \frac{1}{Z_N(T, h)} e^{-H(\{\sigma\})/k_B T}$$

$$H(\{\sigma\}) = -\frac{J}{2} \sum_{\langle i, j \rangle} \sigma_i \sigma_j + h \sum_i \sigma_i$$

Ricavare dalle configurazioni campionate le osservabili di interesse quali la magnetizzazione, la suscettività ed il calore specifico.

Quantità da stimare nel modello di Ising

$$m(\{\sigma\}) = \frac{1}{N} \sum_{i=1}^N \sigma_i$$

Magnetizzazione per spin di una configurazione.

$$\langle m \rangle_{T,h} = \frac{1}{Z_N(T,h)} \sum_{\{\sigma\}} m(\{\sigma\}) e^{-H(\{\sigma\})/k_B T} \quad \Rightarrow \quad \bar{m} = \frac{1}{\#campioni} \sum_{i=1}^{\#campioni} m(\{\sigma\}^{(i)})$$

$$\langle \chi \rangle_{T,h} \propto \left\langle \left(m - \langle m \rangle \right)^2 \right\rangle_{T,h}$$

$$U_N(T,h) = \langle H \rangle_{T,h} = \frac{1}{Z_N(T,h)} \sum_{\{\sigma\}} H(\{\sigma\}) e^{-H(\{\sigma\})/k_B T} \quad \Rightarrow \quad \bar{U} = \frac{1}{\#campioni} \sum_{i=1}^{\#campioni} H(\{\sigma\}^{(i)})$$

$$\langle C \rangle_{T,h} \propto \left\langle \left(H - \langle H \rangle \right)^2 \right\rangle_{T,h}$$

Modello di Ising

- Il caso $d=2$ viene risolto esattamente da Onsager.
- In $d=2$ la transizione si verifica per un valore $T=T_c > 0$.
- In particolare per $h=0$, $T \rightarrow T_c$ si ha divergenza logaritmica del calore specifico e divergenza della suscettività magnetica.
- In $d=3$ non esiste soluzione analitica e ci si deve basare o su approssimazioni varie (per esempio campo medio) o su simulazioni Monte Carlo.

Algoritmo Metropolis per modello di Ising

Supponiamo che $\{\sigma\}$ sia la configurazione attuale.

1. Scegliamo uno spin a caso, σ_i , tra gli N che formano il modello e proponiamo, come nuova configurazione, σ' , che differisce da $\{\sigma\}$ nel solo valore dello spin σ_i .
2. Calcoliamo la differenza di energia $\Delta E = E(\sigma') - E(\sigma)$.
3. Se $\Delta E \leq 0$ accettiamo la nuova configurazione e il passo temporale è finito.
4. Altrimenti estraiamo un numero casuale r da una distribuzione uniforme in $[0,1]$

se $r \leq \exp(-\Delta E / T)$ accettiamo la configurazione nuova σ'

se $r > \exp(-\Delta E / T)$ teniamo la vecchia configurazione

Programma Monte Carlo per simulare Ising 2D

ising.cpp

```
// Simulazione del Modello di Ising in due dimensioni  
// con il metodo Monte Carlo (algoritmo Metropolis)
```

```
#include <cmath >
```

```
#include <iostream >
```

```
#define Lx 50
```

```
#define Ly 50
```

```
double J = +1; // accoppiamento ferromagnetico
```

```
int N=Lx*Ly;    // numero di spins
```

```
int s[Lx][Ly]; // Matrice degli spins
```

```
double T; // Temperatura
```

```
double h; // Campo magnetico
```

```
long int seed;
```

```
int main ()
```

```
{..... srand48(seed); ...}
```

Calcolo efficiente del fattore di Boltzmann $w = \exp(-\beta \Delta H)$

$$H(\{\sigma\}) = -\frac{J}{2} \sum_{\langle i,j \rangle} \sigma_i \sigma_j + h \sum_i \sigma_i$$

Per una simulazione a T e h fissati ci sono solo 10 valori distinti di w che si possono calcolare una sola volta. Infatti la somma dei 4 spins vicini può assumere i seguenti valori:

$$\sum_{\text{neigh } j} \sigma_j = 4 \quad \text{se tutti up}$$


$$\sum_{\text{neigh } j} \sigma_j = +2 \quad \text{se tre up e uno down}$$

$$\sum_{\text{neigh } j} \sigma_j = 0 \quad \text{se due up e due down}$$

$$\sum_{\text{neigh } j} \sigma_j = -2 \quad \text{se uno up e tre down}$$

$$\sum_{\text{neigh } j} \sigma_j = -4 \quad \text{se tutti 4 down}$$

Poichè $\sigma_i = \pm 1$


$$\sigma_i \sum_{\text{neigh } j} \sigma_j = +4, +2, 0, -2, -4.$$

Calcolo efficiente del fattore di Boltzmann $w = \exp(-\beta \Delta E)$

Se il campo magnetico h è diverso da zero allora occorre anche tener conto del termine $h\sigma_i$ che può assumere i due valori $\pm h$.

Ci sono quindi 10 valori di w :

```
double w[17][3]; // Fattori di Boltzmann
void computeBoltzmannFactors ( )
{
  int i;
  for (i = -8; i <= 8; i += 4) {
    w[i + 8][0] = exp( - (i * J + 2 * h) / T);
    w[i + 8][2] = exp( - (i * J - 2 * h) / T);}
}
```

Il primo indice si calcola come

$$8 + 2\sigma_i \sum_{\text{neigh } j} \sigma_j = 0, 4, 6, 12, 16$$

E il secondo come

$$1 + \sigma_i = 0, 2$$

Inizializzazione del programma

```
int steps = 0;

void initialize ( ) {
  int i,j;
  for (i = 0; i < Lx; i++)
    for (j = 0; j < Ly; j++)
      s[i][j] = dran48() < 0.5 ? +1 : -1;
  computeBoltzmannFactors();
  steps = 0;
}
```


Uno step Metropolis

```
int MetropolisStep ( ) {  
    int i,j, iPrev,iNext,jPrev,jNext; int sumNeighbors, delta_ss;  
    double ratio;  
    // Scegliere un sito in maniera aleatoria  
    i = (int)(Lx*dran48()); j = (int)(Ly*dran48());  
    // Ricerca siti vicini (condizioni periodiche)  
    iPrev = i == 0 ? Lx-1 : i-1;    iNext = i == Lx-1 ? 0 : i+1;  
    jPrev = j == 0 ? Ly-1 : j-1;    jNext = j == Ly-1 ? 0 : j+1;  
    sumNeighbors = s[iPrev][j] + s[iNext][j] + s[i][jPrev] +    s[i][jNext];  
    delta_ss = 2*s[i][j]*sumNeighbors;  
    // rapporto dei fattori di Boltzmann  
    ratio = w[delta_ss+8][1+s[i][j]];  
    if (dran48() < ratio)  
    { s[i][j] = -s[i][j]; // Nuova configurazione:Cambio segno allo spin i-esimo  
      return 1;}  
    else return 0;}  
}
```

Uno step Monte Carlo

```
double acceptanceRatio;  
void oneMonteCarloStepPerSpin ( )  
{  
  int accepts,i;  
  accepts = 0;  
  for (i = 0; i < N; i++)  
    if (MetropolisStep())  
      ++accepts;  
  acceptanceRatio = accepts/double(N);  
  ++steps;  
}
```

Calcolo osservabili

```
double magnetizationPerSpin ( )  
{int sSum = 0; int i,j,sSum,ssSum;  
  for (i = 0; i < Lx; i++) for (j = 0; j < Ly; j++) { sSum += s[i]  
[j]; }  
  return sSum / double(N); }
```

```
double energyPerSpin ( )  
{  int sSum = 0; int ssSum = 0; int i,j,iNext,jNext;  
  for (i = 0; i < Lx; i++)  
    for (j = 0; j < Ly; j++)  
      { sSum += s[i][j];  
        iNext = i == Lx-1 ? 0 : i+1;  
        jNext = j == Ly-1 ? 0 : j+1;  
        ssSum += s[i][j]*(s[iNext][j] + s[i][jNext]);  
      }  
  return -(J*ssSum + h*sSum)/N;  
}
```

Esercizio 2: simulazione Monte Carlo del modello di Ising

Step 1 : Scrivere il programma main che chiami in maniera opportuna le routines introdotte in precedenza

In particolare dati come input il campo magnetico h e la temperatura il programma scriva su output le serie temporali (in Monte Carlo steps) dell'energia, della magnetizzazione e dei loro quadrati nella forma seguente:

MC step	E	E^2	m	m^2
1
.....				
N

Andamento delle serie temporali.

Una delle prime cose da osservare in una simulazione Monte Carlo è l'andamento temporale delle osservabili di interesse, per le quali cioè si stimano i valori medi. In particolare si deve essere sicuri che la simulazione abbia raggiunto lo stato di equilibrio dopo un certo tempo t_{eq} . E' solo dopo questo tempo che si possono utilizzare i valori delle serie per la stima dei valori medi.

$$\langle O \rangle \propto \frac{1}{N - t_{eq}} \sum_{i=t_{eq}}^N O_i$$

Convergenza verso l'equilibrio

Un metodo per verificare il raggiungimento dell'equilibrio, cioè la **termalizzazione**, consiste nel simulare due o più sistemi che partono da condizioni iniziali molto diverse e considerarli come termalizzati solo quando tutte le osservabili che vogliamo misurare sono uguali (all'interno delle fluttuazioni termiche).

Step 2 :

Per un sistema con $L=30$ e $T=2.5$ ($T_c = 2.27$) mostrare le serie temporali per m ed E nel caso di tre condizioni iniziali diverse:

1. Valori spin scelti a caso ($m=0$)
2. Tutti gli spin valgono 1 ($m=+1$)
3. Tutti gli spins valgono -1 ($m=-1$)

Hint: *Per fare ciò occorre cambiare la routine `initialize()`*

N.B. $T = 2.5$ è sufficientemente più alta della temperatura critica per avere una convergenza all'equilibrio veloce.

Step 3: Usando sempre $L=30$, stimare il tempo di termalizzazione per diverse temperature. Per temperature maggiori di T_c occorre stimare t_{eq} abbastanza bene e potete fare un grafico di t_{eq} in funzione di T per vedere come questo cresce per $T \rightarrow T_c$

Hint: Si può stimare t_{eq} come il tempo a cui $m^{(t)}$ (o $E^{(t)}$ o $-E^{(t)}$) raggiungono lo zero per la prima volta. Essendo la dinamica stocastica la stima di t_{eq} varierà di run in run. Occorre sempre fare più runs, verificare che la stima non vari troppo con il numero random iniziale e poi prendere il valore più alto.

Step 4: Ripetere la stima esattamente al punto critico $T_c = 2.27$. Per $T=T_c$ come cresce t_{eq} aumentando L da 30 a 50 ?

Transizione di fase

Step 5 :

Calcolare il valor medio di E e m , il calore specifico C_v e la suscettività magnetica χ in funzione di T per $L=30$ e 50 .
e per $2.0 \leq T \leq 2.4$ a passi di $\Delta T=0.05$.

Graficare $\langle E \rangle$, $\langle m \rangle$, C_v , χ in funzione di T .

E' possibile vedere delle indicazioni di transizione di fase ?