

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

SCUOLA DI SCIENZE
Corso di Laurea in Matematica

**CATENE DI MARKOV REVERSIBILI
E APPLICAZIONI
AL METODO MONTECARLO
BASATO SULLE CATENE DI MARKOV**

Tesi di Laurea in Calcolo delle Probabilità e Statistica

Relatore:
Chiar.mo Prof.
Massimo Campanino

Presentata da:
Alessandra Iannuzzi

I Sessione
Anno Accademico 2014-2015

Introduzione

Gli argomenti trattati in questa tesi sono le catene di Markov reversibili e alcune applicazioni al metodo Montecarlo basato sulle catene di Markov. Nel primo capitolo vengono descritte alcune delle proprietà fondamentali delle catene di Markov e in particolare delle catene di Markov reversibili, questo primo capitolo termina con alcuni esempi come il *modello dell'urna di P. e T. Ehrenfest*. Nel secondo capitolo viene descritto il metodo Montecarlo basato sulle catene di Markov, il quale attraverso la simulazione di catene di Markov cerca di stimare la distribuzione di una variabile casuale o di un vettore di variabili casuali con una certa distribuzione di probabilità.

Vengono analizzati:

- l'*algoritmo di Hastings-Metropolis*, il quale genera una catena che ha come probabilità limite la funzione di probabilità desiderata;
- il *campionamento di Gibbs*, che è un caso particolare dell'algoritmo di Hastings-Metropolis;
- un procedimento usato per risolvere problemi di ottimizzazione.

L'ultimo capitolo è dedicato ad un esempio in cui utilizzando Matlab sono evidenziati alcuni aspetti studiati nel corso della tesi.

Nel corso dell'elaborato la sigla CM sta per catene di Markov.

Indice

Introduzione	i
1 Catene di Markov	1
1.1 Catene reversibili	3
2 Metodo Montecarlo basato sulle catene di Markov	11
2.1 Algoritmo di Hastings-Metropolis	11
2.2 Campionamento di Gibbs	14
2.3 Un algoritmo di ottimizzazione globale	18
3 Simulazione con Matlab	23
A Funzioni Matlab	27
Bibliografia	29

Elenco delle figure

3.1	Traiettorie della catena con T fissata	24
3.2	Traiettorie della catena con T decrescente	25

Elenco delle tabelle

3.1	Distanze e cammini a T fissata	25
3.2	Distanze e cammini con T decrescente	26

Capitolo 1

Catene di Markov

Sia $\{X_n, n = 0, 1, 2, 3, \dots\}$ un processo stocastico su un insieme finito o numerabile di possibili valori, sia questo S . L'insieme S è chiamato *spazio degli stati*, supponiamo sia del tipo $\{0, 1, 2, 3, \dots\}$. Scrivendo $X_t = i$ indichiamo che il processo è allo stato i al tempo t . $p_{i,j}$ indica la *probabilità di transizione*, secondo la *proprietà di Markov* la catena nello stato i passa allo stato j indipendentemente dagli stati passati $i - 1, i - 2, \dots$, cioè $P\{X_{n+1} = j | X_n = i, \dots, X_0 = i_0\} = P\{X_{n+1} = j | X_n = i\} = p_{i,j}$. Tale processo è detto *catena di Markov* con probabilità di transizione $p_{i,j}$ $i, j = 0, 1, \dots$

Definizione 1.1. Data una catena di Markov con $|S| = N + 1, N + 1$ finito la matrice

$$\mathbf{P} = \begin{pmatrix} p_{00} & p_{01} & \dots & p_{0N} \\ p_{10} & p_{11} & \dots & p_{1N} \\ \vdots & \vdots & & \vdots \\ p_{N0} & p_{N1} & \dots & p_{NN} \end{pmatrix}$$

è detta *matrice di transizione*.

Gli elementi della matrice hanno le proprietà

$$p_{i,j} \geq 0 \quad \sum_{j=0}^N p_{i,j} = 1 \quad i = 0, \dots, N$$

Definizione 1.2. Una CM è detta *omogenea* se $P\{X_k = s | X_{k-1} = s'\} = P\{X_1 = s | X_0 = s'\}$, cioè la probabilità di transizione dipende solo dello stato del sistema al tempo precedente non dal tempo stesso.

Sia $p_{i,j}^{(n)}$ la probabilità di transizione in n passi,

$$p_{i,j}^{(n)} = P\{X_{k+n} = j | X_k = i\} \quad n \geq 0, i, j \geq 0$$

Questa non dipende da k , ma solo da n ed è l'elemento di posto i, j della matrice \mathbf{P}^n , cioè $[\mathbf{P}^n]_{ij}$. Per convenzione si pone

$$p_{s,s'}^{(0)} = \begin{cases} 1 & \text{se } s = s' \\ 0 & \text{altrimenti} \end{cases}$$

Definizione 1.3. Sia $A_s^+ = \{n | p_{s,s}^{(n)} > 0\}$, se $A_s^+ \neq \emptyset$ il *periodo* di uno stato s è il MCD tra gli elementi di A_s^+ .

Se uno stato ha periodo 1 è detto *aperiodico*.

Consideriamo una CM omogenea con un numero di stati finito, lo stato i *comunica* con lo stato j se per qualche $n \geq 0$ $p_{i,j}^{(n)} > 0$. Si può indicare con $i \rightarrow j$ e significa che esiste un percorso che con probabilità positiva che va da i a j . Se $i \rightarrow j$ e $j \rightarrow i$ allora i due stati si dicono *equivalenti*, $i \leftrightarrow j$. L'essere equivalenti è una relazione di equivalenza. Proviamo che valgono le tre proprietà, riflessiva, simmetrica e transitiva.

- (i) Ogni stato i è equivalente a se stesso.
- (ii) Se i è equivalente a j , allora j è equivalente a i .
- (iii) Prima di tutto supponiamo ci siano tre stati i, j, k tale che i comunica con j e j comunica con k . Se $i \rightarrow j$ significa che $\exists n_1$ tale che $p_{i,j}^{(n_1)} > 0$, analogamente per $j \rightarrow k$ $\exists n_2$ tale che $p_{j,k}^{(n_2)} > 0$. Abbiamo $p_{i,k}^{(n_1+n_2)} = [\mathbf{P}^{n_1+n_2}]_{ik} = \sum_s p_{i,s}^{(n_1)} p_{s,k}^{(n_2)} \geq p_{i,j}^{(n_1)} p_{j,k}^{(n_2)} > 0$. Quindi i comunica con k . Analogamente si prova che k comunica con i , concludiamo che i e k sono equivalenti.

Una CM è detta *irriducibile* se c'è solo una classe di equivalenza, cioè tutti gli stati sono equivalenti tra loro.

Teorema 1.0.1. (Teorema ergodico) *Sia $(X_t)_{t \in \mathbb{N}}$ una CM omogenea, con S spazio degli stati finito, $|S| = N$, irriducibile e aperiodica. Allora*

- *esiste $\Pi = \{\pi_1, \dots, \pi_N\}$ distribuzione sullo spazio degli stati, tale che*

$$0 \leq \pi_s \leq 1 \quad \forall s \in S \quad e \quad \sum_{s \in S} \pi_s = 1$$
- *$\lim_{n \rightarrow \infty} p_{s',s}^{(n)} = \pi_s, \forall s' \in S$ con velocità esponenziale.*

Osservazione 1. Una CM che soddisfa le ipotesi del teorema ergodico è detta *ergodica* e converge ad una distribuzione invariante sugli stati e indipendentemente dallo stato iniziale. Sia $\Pi = (\pi_s)_{s \in S}$ la distribuzione invariante detta anche *stazionaria*. Se $P\{X_0 = s\} = \pi_s \quad \forall s$, cioè Π è presa come distribuzione iniziale, allora $\forall s \in S, \forall n \geq 0 \quad P\{X_n = s\} = \pi_s$. Da questo segue

$$\begin{aligned} \pi_s &= P\{X_1 = s\} \\ &= \sum_{s' \in S} P\{X_0 = s'\} p_{s',s} \\ &= \sum_{s' \in S} \pi_{s'} p_{s',s} \end{aligned}$$

Osservazione 2. Sotto le ipotesi del teorema ergodico si può dimostrare che le quantità $\pi_s, s \in S$ sono le uniche soluzioni del sistema di equazioni lineari

$$\begin{cases} \pi_s = \sum_{s' \in S} \pi_{s'} p_{s',s} & s \in S \\ \sum_{s \in S} \pi_s = 1 \end{cases}$$

π_s è asintoticamente la frazione di tempo in cui la CM è nello stato s , $\pi_{s'} p_{s',s}$ è la percentuale di tempo in cui la CM entra nello stato s da s' . La somma su tutti gli stati s della frazione di tempo in cui la catena è in s è 1.

1.1 Catene reversibili

Prendiamo una catena di Markov $(X_t)_{t \in \mathbb{Z}}$ ergodica con distribuzione stazionaria $(\pi_s)_{s \in S}$ e consideriamo il processo inverso, consideriamo la medesima catena andando indietro nel tempo. Osserviamo che ora $t \in \mathbb{Z}$, cioè estendiamo il tempo da \mathbb{N} a \mathbb{Z} . Partendo dal tempo n si ha una successione di stati

$X_n, X_{n-1}, X_{n-2}, \dots$, prima di tutto verifichiamo che il processo inverso è una CM, cioè

$$P\{X_m = j | X_{m+1} = i, X_{m+2} = i_{m+2}, \dots\} = P\{X_m = j | X_{m+1} = i\}$$

Infatti

Sia $m + 1$ il tempo presente. Essendo X_0, X_1, X_2, \dots una CM, la distribuzione condizionata futura X_{m+2}, X_{m+3}, \dots dato lo stato presente X_{m+1} è indipendente dallo stato passato X_m . Inoltre la relazione di indipendenza è simmetrica, quindi dato X_{m+1} , X_m è indipendente da X_{m+2}, X_{m+3}, \dots

Sia $q_{i,j}$ la probabilità di transizione della catena inversa

$$\begin{aligned} q_{i,j} &= P\{X_m = j | X_{m+1} = i\} \\ &= \frac{P\{X_m = j, X_{m+1} = i\}}{P\{X_{m+1} = i\}} \\ &= \frac{P\{X_m = j\}P\{X_{m+1} = i | X_m = j\}}{P\{X_{m+1} = i\}} \\ &= \frac{\pi_j p_{j,i}}{\pi_i} \end{aligned}$$

Se $q_{i,j} = p_{i,j} \forall i, j$ la catena è detta catena di Markov *reversibile*. Da quanto visto sopra la condizione di reversibilità può essere espressa anche con

$$\pi_i p_{i,j} = \pi_j p_{j,i} \quad \forall i, j \quad (1.1)$$

L'equazione (1.1) indica che il tasso con cui il processo va da i a j è uguale al tasso con cui il processo va da j a i .

Osservazione 3. L'equazione (1.1) è condizione necessaria per la reversibilità.

Infatti

Una transizione $i \rightarrow j$ indietro nel tempo è equivalente ad una transizione in avanti $j \rightarrow i$. Se $X_m = i$ e $X_{m-1} = j$ allora guardando all'indietro osserviamo una transizione $i \rightarrow j$, guardando

in avanti osserviamo una transizione $j \rightarrow i$. Il tasso con cui la catena in avanti va da $j \rightarrow i$ è sempre uguale al tasso con cui la catena inversa va da $i \rightarrow j$.

Osservazione 4. Se troviamo x_i numeri non negativi, la cui somma è 1 che soddisfano (1.1) allora la CM è reversibile e le x_i sono le probabilità limite.

Infatti

$$x_i p_{i,j} = x_j p_{j,i} \quad \forall i, j, \quad \sum_i x_i = 1 \quad (1.2)$$

sommando su i si ha

$$\sum_i x_i p_{i,j} = x_j \sum_i p_{j,i} = x_j \sum_i x_i = 1$$

siccome le probabilità limite π_i sono l'unica soluzione della precedente espressione si ha che $x_i = \pi_i \quad \forall i$.

Teorema 1.1.1. *Una CM ergodica con $p_{i,j} = 0$ ogni volta che $p_{j,i} = 0$ è reversibile se e solo se iniziando dallo stato i qualsiasi percorso di ritorno verso i ha la stessa probabilità del percorso inverso, cioè*

$$p_{i,i_1} p_{i_1,i_2} \cdots p_{i_k,i} = p_{i,i_k} p_{i_k,i_{k-1}} \cdots p_{i_1,i} \quad \forall \text{ stato } i, i_1, \dots, i_k. \quad (1.3)$$

Dimostrazione.

\Leftarrow

Fissati gli stati i e j riscriviamo (1.3) $p_{i,i_1} p_{i_1,i_2} \cdots p_{i_k,j} p_{j,i} = p_{i,j} p_{j,i_k} \cdots p_{i_1,i}$ sommiamo su tutti gli stati i_1, \dots, i_k si ha $p_{i,j}^{(k+1)} p_{j,i} = p_{i,j} p_{j,i}^{(k+1)}$ mandando $k \rightarrow \infty$ si ha $\pi_j p_{j,i} = p_{i,j} \pi_i$, cioè la CM è reversibile.

\Rightarrow

Consideriamo tre strati i, j e k . Vogliamo provare che

$$p_{i,k} p_{k,j} p_{j,i} = p_{i,j} p_{j,k} p_{k,i} \quad \forall i, j, k \quad (1.4)$$

Questo indica che il percorso $i \rightarrow j \rightarrow k \rightarrow i$ ha la stessa probabilità del percorso inverso $i \rightarrow k \rightarrow j \rightarrow i$.

Per ipotesi la CM considerata è reversibile, quindi il percorso $i \rightarrow j \rightarrow k \rightarrow i$ ha lo stesso tasso del percorso inverso $i \rightarrow k \rightarrow j \rightarrow i$, abbiamo $\pi_i p_{i,k} p_{k,j} p_{j,i} = \pi_i p_{i,j} p_{j,k} p_{k,i}$. Considerando $\pi_i (> 0)$, otteniamo (1.4). \square

Vediamo alcuni esempi di CM reversibili.

Esempio 1.1. (*passeggiata aleatoria*) Consideriamo una passeggiata aleatoria con stati $0, 1, \dots, M$ e probabilità di transizione

$$p_{i,i+1} = \alpha_i = 1 - p_{i,i-1} \quad i = 1, \dots, M - 1$$

$$p_{0,1} = \alpha_0 = 1 - p_{0,0}$$

$$p_{M,M} = \alpha_M = 1 - p_{M,M-1}$$

Questa CM, che può spostarsi solo da uno stato ad uno dei due più vicini, è reversibile. Il numero di transizioni da $i \rightarrow i+1$ ad ogni tempo deve differire al massimo di 1 dal numero di transizioni da $i+1 \rightarrow i$. Tra due spostamenti qualsiasi da $i \rightarrow i+1$ deve essercene uno da $i+1 \rightarrow i$, e viceversa, essendoci solo un modo per raggiungere i da uno stato più alto passando per $i+1$. Quindi il tasso con cui il processo va da $i \rightarrow i+1$ è uguale al tasso da $i+1 \rightarrow i$, cioè la CM considerata è reversibile.

Ora possiamo trovare le probabilità limite usando (1.1).

Per $i = 0, j = 1$ (1.1) diventa $\pi_0 p_{0,1} = \pi_1 p_{1,0}$, sapendo che $\alpha_1 = 1 - p_{1,0}$ e $p_{0,1} = \alpha_0$ si ha $\pi_0 \alpha_0 = \pi_1 (1 - \alpha_1)$. In generale si ha

$$\pi_0 \alpha_0 = \pi_1 (1 - \alpha_1)$$

$$\pi_1 \alpha_1 = \pi_2 (1 - \alpha_2)$$

\vdots

$$\pi_i \alpha_i = \pi_{i+1} (1 - \alpha_{i+1}) \quad i = 0, 1, \dots, M - 1$$

Risolviamo rispetto π_0 ,

$$\pi_1 = \frac{\alpha_0}{1 - \alpha_1} \pi_0$$

$$\pi_2 = \frac{\alpha_1}{1 - \alpha_2} \pi_1 = \frac{\alpha_1 \alpha_0}{(1 - \alpha_2)(1 - \alpha_1)} \pi_0$$

in generale

$$\pi_i = \frac{\alpha_{i-1} \cdots \alpha_0}{(1 - \alpha_i) \cdots (1 - \alpha_1)} \pi_0 \quad i = 1, \dots, M$$

Siccome $\sum_0^M \pi_i = 1$

$$1 = \pi_0 \left[1 + \sum_{j=1}^M \frac{\alpha_{j-1} \cdots \alpha_0}{(1 - \alpha_j) \cdots (1 - \alpha_1)} \right]$$

Concludendo

$$\pi_0 = \left[1 + \sum_{j=1}^M \frac{\alpha_{j-1} \cdots \alpha_0}{(1 - \alpha_j) \cdots (1 - \alpha_1)} \right]^{-1}, \quad \pi_i = \frac{\alpha_{i-1} \cdots \alpha_0}{(1 - \alpha_i) \cdots (1 - \alpha_1)} \pi_0 \quad i = 1, \dots, M \quad (1.5)$$

Esempio 1.2. (*modello dell'urna di P. e T. Ehrenfest*) Questo modello è un caso particolare dell'esempio precedente, usato per descrivere il movimento delle molecole. Ci sono due urne in cui sono distribuite in modo arbitrario M molecole. Ad ogni intervallo di tempo fissato viene estratta una molecola dalla prima urna e messa nell'altra. Il numero di molecole nella prima urna è una CM dello stesso tipo dell'esempio sopra con

$$\alpha_i = 1 - \frac{i}{M} \quad i = 0, 1, \dots, M$$

Usando $(a + b)^n = \sum_{k=0}^n \binom{n}{k} a^{n-k} b^k$ con $a = b = \frac{1}{2}$, si ha

$$1 = \left(\frac{1}{2} + \frac{1}{2} \right)^M = \sum_{j=0}^M \binom{M}{j} \left(\frac{1}{2} \right)^M$$

e (1.5) troviamo le probabilità limite

$$\pi_1 = M\pi_0, \quad \pi_2 = \binom{M}{2}\pi_0 \quad \dots \quad \pi_i = \binom{M}{i}\pi_0 \quad i = 1, \dots, M$$

$$\begin{aligned} \pi_0 &= \left[1 + \sum_{j=1}^M \frac{(M - j + 1) \cdots (M - 1)M}{j(1 - j) \cdots 1} \right]^{-1} \\ &= \left[\sum_{j=0}^M \binom{M}{j} \right]^{-1} \\ &= \left(\frac{1}{2} \right)^M \end{aligned}$$

$$\pi_i = \binom{M}{i} \left(\frac{1}{2}\right)^M \quad i = 0, 1, \dots, M$$

Questa è una distribuzione binomiale di parametri $(M, \frac{1}{2})$. Dopo un certo tempo le posizioni di ogni molecola sono indipendenti tra loro e ogni molecola ha la stessa probabilità di trovarsi in una delle due urne. Consideriamo una molecola, la sua posizione è indipendente da quella delle altre $M - 1$ molecole ed può essere scelta tra le altre con probabilità $\frac{1}{M}$. Inoltre per simmetria è ugualmente probabile che sia in una delle due urne.

Esempio 1.3. (*grafo connesso*) Un grafo è formato da un insieme i cui elementi sono detti *nodi* e da un insieme i cui elementi sono coppie di nodi, detti *archi*. Gli archi non hanno direzione. Se c'è un cammino tra ognuno delle $\binom{n}{2}$ distinte coppie di nodi il grafo si dice *connesso*.

Consideriamo un grafo connesso e un numero w_{ij} associato all'arco (i, j) per ogni arco, detto *peso* dell'arco. Se ad un certo tempo la particella si trova al nodi i al tempo successivo si troverà al nodo j con probabilità

$$p_{i,j} = \frac{w_{ij}}{\sum_j w_{ij}}, \quad \text{con } w_{ij} = 0 \text{ se } (i, j) \text{ non é un arco}$$

Partendo dall'equazione (1.1) verifichiamo che il processo è reversibile.

$$\pi_i \frac{w_{ij}}{\sum_j w_{ij}} = \pi_j \frac{w_{ji}}{\sum_i w_{ji}}$$

ovviamente $w_{ij} = w_{ji}$

$$\frac{\pi_i}{\sum_j w_{ij}} = \frac{\pi_j}{\sum_i w_{ji}}$$

che è equivalente a

$$\frac{\pi_i}{\sum_j w_{ij}} = c, \quad \pi_i = c \sum_j w_{ij}$$

oppure poichè $\sum_i \pi_i = 1$

$$\pi_i = \frac{\sum_j w_{ij}}{\sum_i \sum_j w_{ij}}$$

Questo prova che la catena è tempo reversibile.

Esempio 1.4. Supponiamo di avere un insieme di n elementi, numerati da 1 e n , disposti in una lista ordinata. Ad ogni unità di tempo viene prelevato un elemento dalla lista, l'elemento i è scelto con probabilità P_i . Successivamente l'elemento è posizionato nella lista scalando di un posto all'indietro rispetto al suo posto precedente (regola *one-closer*).

Per ogni vettore di probabilità $\mathbf{P} = (P_1, \dots, P_n)$ la lista può essere modellata come una CM con $n!$ stati. Dimostriamo che è una CM reversibile. Questo segue dal teorema 1.1.1, vediamo prima il caso per $n = 3$ e consideriamo il cammino dallo stato $(1, 2, 3)$ a se stesso (l'elemento spostato è in grassetto):

$$(1, \mathbf{2}, 3) \rightarrow (2, 1, \mathbf{3}) \rightarrow (2, \mathbf{3}, 1) \rightarrow (3, 2, \mathbf{1}) \rightarrow (3, \mathbf{1}, 2) \rightarrow (1, 3, \mathbf{2}) \rightarrow (1, 2, 3)$$

Il prodotto delle probabilità di transizione

$$\text{in avanti} \quad P_2 P_3 P_3 P_1 P_1 P_2 = P_1^2 P_2^2 P_3^2$$

$$\text{all'indietro} \quad P_3 P_3 P_2 P_2 P_1 P_1 = P_1^2 P_2^2 P_3^2$$

In generale sia f_i il numero di spostamenti dell'elemento i in avanti, se il cammino parte da un certo stato fissato per ritornare a questo l'elemento i sarà spostato all'indietro f_i volte. Il numero di spostamenti all'indietro dell'elemento i sono esattamente il numero di spostamenti in avanti nel percorso inverso, quindi il prodotto delle probabilità di transizione per il percorso diretto e per il suo inverso è

$$\prod_i P_i^{f_i + r_i}$$

con r_i numero di volte in cui l'elemento i è in prima posizione e il percorso, diretto o inverso, non cambia stato.

Per qualsiasi permutazione i_1, i_2, \dots, i_n di $1, 2, \dots, n$ sia $\pi(i_1, i_2, \dots, i_n)$ la probabilità limite sotto la regola one-closer, abbiamo

$$P_{i_{j+1}} \pi(i_1, \dots, i_j, i_{j+1}, \dots, i_n) = P_{i_j} \pi(i_1, \dots, i_{j+1}, i_j, \dots, i_n) \quad \forall \text{ permutazione.}$$

Capitolo 2

Metodo Montecarlo basato sulle catene di Markov

Supponiamo di voler simulare una variabile casuale X con funzione di probabilità $P\{X = j\} = p_j \quad j = 1, \dots, N$. L'idea del metodo Montecarlo basato sulle catene di Markov è:

- * costruire una CM aperiodica, irriducibile e con distribuzione limite $p_j, \quad j = 1, \dots, N$;
- * simulare la catena per n passi;
- * restituire X_n .

Inoltre questo metodo si usa per stimare il valore atteso $E[h(X)] = \sum_{j=1}^N h(j)p_j$, per qualche funzione h . Si genera una sequenza di stati X_1, \dots, X_n , come descritto sopra, e si usa l'estimatore $\frac{1}{n} \sum_{i=1}^n h(X_i)$.

2.1 Algoritmo di Hastings-Metropolis

Siano $b(1), \dots, b(m)$ numeri positivi e $B = \sum_{j=1}^m b(j)$, supponiamo m grande e B difficile da calcolare. Vogliamo simulare una variabile o una sequenza di variabili casuali con distribuzione di probabilità $\pi(j) = \frac{b(j)}{B}$. Per fare questo usiamo l'*algoritmo di Hastings-Metropolis*, il quale consiste nel co-

struire una CM reversibile le cui probabilità limite sono $\pi(j) \quad j = 1, \dots, m$.

Consideriamo:

- \mathbf{Q} matrice di transizione di una CM irriducibile sugli interi $1, \dots, m$, $q(i, j)$ elemento di posto i, j
- $\{X_n, n \geq 0\}$ una CM, quando $X_n = i$ viene generata una variabile casuale X tale che $P\{X = j\} = q(i, j)$, $j = 1, \dots, m$.

$$\text{Se } X = j, \quad \text{allora } X_{n+1} = \begin{cases} j & \text{con probabilità } \alpha(i, j) \\ i & \text{con probabilità } 1 - \alpha(i, j) \end{cases}$$

Sotto queste ipotesi la sequenza degli stati è una CM con probabilità di transizione

$$p_{i,j} = q(i, j)\alpha(i, j) \quad \text{se } j \neq i$$

$$p_{i,i} = q(i, i) + \sum_{k \neq i} q(i, k)(1 - \alpha(i, k))$$

Proviamo che si tratta di una CM reversibile con probabilità stazionarie $\pi(j) = \frac{b(j)}{B}$. Dobbiamo verificare che l'equazione di reversibilità sia soddisfatta

$$\pi(i)p_{i,j} = \pi(j)p_{j,i} \quad \text{per } i \neq j$$

che è equivalente a

$$\pi(i)q(i, j)\alpha(i, j) = \pi(j)q(j, i)\alpha(j, i) \quad (2.1)$$

Consideriamo

$$\pi(j) = \frac{b(j)}{B} \quad \text{e} \quad \alpha(i, j) = \min\left(\frac{\pi(j)q(j, i)}{\pi(i)q(i, j)}, 1\right) \quad (2.2)$$

allora (2.1) è verificata. Infatti:

$$\star \text{ se } \alpha(i, j) = \frac{\pi(j)q(j, i)}{\pi(i)q(i, j)} = \frac{b(j)q(j, i)}{b(i)q(i, j)} \text{ allora } \alpha(j, i) = 1$$

$$\star \text{ se } \alpha(i, j) = 1 \text{ allora } \alpha(j, i) = \frac{\pi(i)q(i, j)}{\pi(j)q(j, i)} = \frac{b(i)q(i, j)}{b(j)q(j, i)}$$

In entrambi casi segue l'equazione (2.1). Osserviamo che non è necessario conoscere B per definire la CM.

Riassumiamo i passaggi per generare una CM reversibile con probabilità limite $\pi(i) = \frac{b(i)}{B}$ $i = 1, \dots, m$ usando l'algoritmo di Hastings-Metropolis.

1. Si sceglie \mathbf{Q} matrice di transizione di una CM irriducibile con probabilità di transizione $q(i, j)$ $i, j = 1, \dots, m$. e un intero k tra 1 e m .
2. Sia $n = 0$ e $X_0 = k$.
3. Si genera una variabile casuale X tale che $P\{X = j\} = q(X_n, j)$ e un numero casuale U .

4. Se

$$U < \frac{b(X)q(X, X_n)}{b(X_n)q(X_n, X)} \quad \text{allora} \quad NS = X$$

altrimenti $NS = X_n$.

5. $n = n + 1, X_n = NS$.

6. Tornare al punto 3.

Esempio 2.1. Sia \mathcal{L} l'insieme dei sottografi di un dato grafo che hanno la proprietà che $\forall(i, j)$ coppia di vertici c'è un unico percorso da i a j , questi sottografi si chiamano *alberi*. Vogliamo generare un elemento casuale con distribuzione uniforme appartenente ad \mathcal{L} . Per iniziare diamo la seguente definizione, due alberi sono detti *vicini* se tutti gli archi di uno degli alberi tranne uno sono anche archi dell'altro albero. Sia $N(s)$ l'insieme dei vicini di s , $|N(s)|$ il numero di elementi di tale insieme. Definiamo la probabilità di transizione come $q(s, t) = \frac{1}{|N(s)|}$ se $t \in N(s)$, cioè ogni stato che è un vicino di s ha la stessa probabilità di essere il successivo. Le probabilità limite desiderate sono $\pi(s) = C$, da questo segue $\pi(s) = \pi(t)$ e $\alpha(s, t) = \min\left(\frac{|N(s)|}{|N(t)|}, 1\right)$. Questo indica che se s è lo stato presente allora è scelto casualmente uno dei suoi vicini, chiamiamolo t .

- Se il grado del vertice t è minore di quello di s , cioè t ha meno vicini di s , il prossimo stato è t .
- Altrimenti è generato un numero casuale U , con distribuzione uniforme sull'intervallo $(0, 1)$, se $U < \frac{|N(s)|}{|N(t)|}$ il prossimo stato è t ; altrimenti è s .

2.2 Campionamento di Gibbs

La versione più usata dell'algoritmo di Hastings-Metropolis è il *campionamento di Gibbs*. Sia $\mathbf{X} = (X_1, \dots, X_n)$ un vettore casuale discreto con distribuzione di probabilità $p(\mathbf{x})$ nota a meno di una costante moltiplicativa, cioè $p(\mathbf{x}) = Cg(\mathbf{x})$ con $g(\mathbf{x})$ noto, C no. Vogliamo generare un vettore casuale con la stessa distribuzione di \mathbf{X} . Si assume che $\forall i$ e $\forall x_j, j \neq i$ possiamo generare una variabile casuale X avente funzione di probabilità

$$P\{X = x\} = P\{X_i = x | X_j = x_j, j \neq i\} \quad (2.3)$$

Consideriamo una CM con stati $\mathbf{x} = (x_1, \dots, x_n)$. Quando lo stato presente è \mathbf{x} , viene scelta una coordinata tra $1, \dots, n$ con uguale probabilità. Supponiamo sia scelta la coordinata i , allora si genera una variabile casuale con funzione di probabilità (2.3). Se $X = x$, un possibile candidato per lo stato successivo è $\mathbf{y} = (x_1, \dots, x_{i-1}, x, x_{i+1}, \dots, x_n)$. Si usa l'algoritmo di Hastings-Metropolis con

$$q(\mathbf{x}, \mathbf{y}) = \frac{1}{n} P\{X_i = x | X_j = x_j, j \neq i\} = \frac{p(\mathbf{y})}{nP\{X_j = x_j, j \neq i\}}$$

Vogliamo che la funzione di probabilità limite sia p , usando (2.2) si ha che \mathbf{y} è accettato come stato successivo con probabilità

$$\begin{aligned} \alpha(\mathbf{x}, \mathbf{y}) &= \min\left(\frac{p(\mathbf{y})q(\mathbf{y}, \mathbf{x})}{p(\mathbf{x})q(\mathbf{x}, \mathbf{y})}, 1\right) \\ &= \min\left(\frac{p(\mathbf{y})p(\mathbf{x})}{p(\mathbf{x})p(\mathbf{y})}, 1\right) \\ &= 1 \end{aligned}$$

Quindi il nuovo candidato per lo stato successivo è sempre accettato. Inoltre si noti che per calcolare α non è necessario conoscere C .

Nei seguenti esempi abbiamo applicato il campionamento di Gibbs. A differenza di quanto detto fin'ora vengono generate delle CM con spazio degli stati continuo, sia S . Diamo alcuni cenni riguardanti questo caso.

- La probabilità di transizione è sostituita dalla *densità di probabilità di transizione*. Per calcolare la probabilità di transizione da un punto ad un insieme si fa l'integrale della densità sull'insieme.
- Nel caso discreto $\sum_{j=0}^N p_{i,j} = 1 \quad i = 0, \dots, N$, ora è l'integrale della densità su S ad essere uguale ad 1.
- La densità di probabilità di transizione in due passi è $p^{(2)}(x, y) = \int_S p(x, z)p(z, y)dz$, in generale in n passi è

$$p^{(n)}(x, y) = \int_S p^{(n-1)}(x, z)p(z, y)dz.$$

Gli esempi trattati non sono complicati e si procede in modo simile al caso in cui lo spazio degli stati è discreto.

Esempio 2.2. Supponiamo di voler generare n punti uniformemente distribuiti nel cerchio di raggio 1 e centro l'origine, condizionati dall'evento che tra ogni coppia di punti la distanza è almeno d . Poniamo

$$\beta = P\{\text{due punti hanno tra loro una distanza almeno di } d\}$$

Usiamo il campionamento di Gibbs per trovare gli n punti.

- Consideriamo n punti nel cerchio con la proprietà che ogni due punti siano ad una distanza maggiore di d l'uno dall'altro.
- Generiamo un numero casuale U , uniformemente distribuito tra 0 e 1, e poniamo $I = \text{Int}(nU) + 1$

- Generiamo un punto casuale nel cerchio. Se questo si trova ad una distanza maggiore di d dagli $n - 1$ punti senza considerare x_I , allora x_I viene sostituito da tale punto.
- Ripartiamo dal secondo punto.

Dopo un gran numero di iterazioni otterremo gli n punti con approssimativamente la distribuzione desiderata.

Esempio 2.3. Siano $X_i, i = 1, \dots, n$ delle variabili casuali indipendenti, ognuna con distribuzione esponenziale di parametro $\lambda_i, i = 1, \dots, n$. Sia $S = \sum_{i=1}^n X_i$, supponiamo di voler generare un vettore casuale $\mathbf{X} = (X_1, \dots, X_n)$ condizionato all'evento che $S > c$, per qualche costante c grande. Lo scopo è generare il valore di un vettore casuale con funzione di densità

$$f(x_1, \dots, x_n) = \frac{1}{P\{S > c\}} \prod_{i=1}^n \lambda_i e^{-\lambda_i x_i} \quad \text{se} \quad \sum_{i=1}^n x_i > c$$

Per iniziare consideriamo un vettore $\mathbf{x} = (x_1, \dots, x_n)$ con $x_i > 0, i = 1, \dots, n$ e $\sum_{i=1}^n x_i > c$. Allora generiamo un numero casuale U , poniamo $I = \text{Int}(nU) + 1$ e supponiamo $I = i$. Vogliamo generare una variabile casuale esponenziale X con parametro λ_i , condizionata all'evento che $X + \sum_{j \neq i} x_j > c$, cioè $X > c - \sum_{j \neq i} x_j$. Sapendo che l'esponenziale condizionata ad essere maggiore di un certa costante positiva è distribuita come la costante più l'esponenziale, generiamo una variabile casuale esponenziale Y di parametro λ_i e poniamo

$$X = Y + \left(c - \sum_{j \neq i} x_j \right)^+, \quad \text{dove } b^+ = \begin{cases} b & \text{se } b > 0 \\ 0 & \text{altrimenti} \end{cases}$$

Il valore di x_i è posto uguale ad X ed inizia una nuova iterazione.

Riassumiamo i passaggi.

- Consideriamo un vettore \mathbf{x} con $x_i > 0, i = 1, \dots, n$ e $\sum_{i=1}^n x_i > c$, un numero casuale U , uniformemente distribuito tra 0 e 1, e poniamo $N = 0$.
- Poniamo $I = \text{Int}(nU) + 1, S = \sum_{j \neq i} x_j$.

- Generiamo $x_i = Y + (c - S)^+$, con $Y = -\frac{\log(U)}{\lambda_i}$
- Aggiorniamo il valore di N con $N + 1$ e ripartiamo dal secondo punto.

Ora supponiamo di voler stimare $\alpha = P\{h(\mathbf{X}) > a\}$, con $\mathbf{X} = (X_1, \dots, X_n)$, h è una funzione arbitraria di \mathbf{X} , α è molto piccolo. Per i valori $-\infty = a_0 < \dots, < a_k = a$,

$$\alpha = \prod_{i=1}^k P\{h(\mathbf{X}) > a_i | h(\mathbf{X}) > a_{i-1}\}$$

Possiamo ottenere un estimatore di α prendendo il prodotto degli estimatori delle quantità $P\{h(\mathbf{X}) > a_i | h(\mathbf{X}) > a_{i-1}\} \forall i = 1, \dots, k$. Per stimare $P\{h(\mathbf{X}) > a_i | h(\mathbf{X}) > a_{i-1}\}$ usiamo il campionamento di Gibbs.

1. Porre $J = N = 0$.
2. Scegliere un vettore \mathbf{x} tale che $h(\mathbf{x}) > a_{i-1}$.
3. Generare un numero casuale U e porre $I = \text{Int}(nU) + 1$.
4. Poniamo $I = k$, generare X avente la distribuzione condizionata di X_k dato $X_j = x_j, j \neq k$.
5. Se $h(x_1, \dots, x_{k-1}, X, x_{k+1}, \dots, x_n) \leq a_{i-1}$ tornare al punto 4.
6. $N = N + 1, x_k = X$.
7. Se $h(x_1, \dots, x_n) > a_i$, allora $J = J + 1$
8. Tornare al punto 3.

Il rapporto tra il valore finale di J e N è la stima di $P\{h(\mathbf{X}) > a_i | h(\mathbf{X}) > a_{i-1}\}$.

Esempio 2.4. Vogliamo stimare il numero di permutazioni $\mathbf{x} = (x_1, \dots, x_n)$ per cui $t(\mathbf{x}) > a$, dove $t(\mathbf{x}) = \sum_{j=1}^n jx_j$ ed a è tale che il numero di permutazioni è molto piccolo rispetto ad $n!$. Sia $\mathbf{X} = (X_1, \dots, X_n)$ con uguale probabilità una delle $n!$ permutazioni e sia $\alpha = P\{T(\mathbf{X}) > a\}$, allora α è

piccolo e la quantità di interesse è $\alpha n!$. Posto $0 < a_0 < a_1 < \dots < a_k = a$ abbiamo

$$\alpha = \prod_{i=1}^k P\{T(\mathbf{X}) > a_i | T(\mathbf{X}) > a_{i-1}\}$$

Per stimare $P\{T(\mathbf{X}) > a_i | T(\mathbf{X}) > a_{i-1}\}$ possiamo usare il procedimento appena descritto o l'algoritmo di Hastings-Metropolis.

Nel secondo caso generiamo una CM con distribuzione limite

$$\pi(\mathbf{x}) = \frac{1}{N_{i-1}} \quad \text{se } T(\mathbf{x}) > a_{i-1}$$

dove N_{i-1} è il numero di permutazioni \mathbf{x} tali che $T(\mathbf{x}) > a_{i-1}$. La percentuale degli stati generati \mathbf{x} per cui $T(\mathbf{x}) > a_i$ è l'estimatore di $P\{T(\mathbf{X}) > a_i | T(\mathbf{X}) > a_{i-1}\}$.

Osservazione 5. Consideriamo l'esempio **2.1**. Sia \mathcal{L} l'insieme delle permutazioni (x_1, \dots, x_n) di $(1, \dots, n)$ tale che $\sum jx_j > a$, a è una costante fissata. Vogliamo generare un elemento casuale appartenente a \mathcal{L} uniformemente distribuito. Due permutazioni si dicono *vicine* se una è ottenuta dall'altra scambiando due delle sue coordinate. Procedendo come nell'esempio **2.1** si ottengono i medesimi risultati, in particolare $\pi(\mathbf{s}) = \frac{1}{|\mathcal{L}|}$.

2.3 Un algoritmo di ottimizzazione globale

Consideriamo l'algoritmo di Hastings-Metropolis analizzato all'inizio del capitolo. Siano π una distribuzione di probabilità su $1, \dots, m$ tale che $\pi(i) > 0$ $i = 1, \dots, m$ e \mathbf{Q} matrice di transizione su $1, \dots, m$ irriducibile e simmetrica. Per ottenere una variabile aleatoria con distribuzione approssimativamente uguale a π usiamo l'algoritmo di Hastings-Metropolis, simulando una CM formata dalla sequenza degli stati.

- Scegliamo \mathbf{Q} come definito sopra.
- $\{X_n, n \geq 0\}$ una CM, quando $X_n = i$ viene generata una variabile casuale X tale che $P\{X = j\} = q(i, j)$, $j = 1, \dots, m$.

Sia $X = j$

- se $\pi(j) \geq \pi(i)$ si pone $X_{n+1} = j$
- se $\pi(j) < \pi(i)$ si pone $X_{n+1} = \begin{cases} j & \text{con probabilità } \frac{\pi(j)}{\pi(i)} \\ i & \text{con probabilità } 1 - \frac{\pi(j)}{\pi(i)} \end{cases}$

La probabilità di transizione di tale CM è

$$p_{i,j} = \begin{cases} q(i,j) & \text{se } \pi(j) \geq \pi(i) \\ q(i,j) \frac{\pi(j)}{\pi(i)} & \text{se } \pi(j) < \pi(i) \\ 1 - \sum_{i \neq j} p_{i,j} & \text{se } j = i \end{cases} \quad (2.4)$$

Supponiamo che la distribuzione di probabilità sia del tipo

$$\pi(i)^\epsilon = \frac{e^{-\frac{H(i)}{\epsilon}}}{Z_\epsilon}$$

dove H è una funzione sull'insieme degli stati, ϵ è un parametro positivo e Z_ϵ è la costante di normalizzazione, $Z_\epsilon = \sum_i e^{-\frac{H(i)}{\epsilon}}$. La probabilità di transizione (2.4) diventa

$$p_{i,j}^\epsilon = \begin{cases} q(i,j) & \text{se } H(j) \leq H(i) \\ q(i,j) e^{-(H(j)-H(i))/\epsilon} & \text{se } H(j) > H(i) \\ 1 - \sum_{i \neq j} p_{i,j}^\epsilon & \text{se } j = i \end{cases}$$

Sia i lo stato presente e j il candidato per lo stato successivo.

- Se $H(j) \leq H(i)$ si pone $X_{n+1} = j$
- Se $H(j) > H(i)$ si pone $X_{n+1} = \begin{cases} j & \text{con probabilità } e^{-(H(j)-H(i))/\epsilon} \\ i & \text{con probabilità } 1 - e^{-(H(j)-H(i))/\epsilon} \end{cases}$

Osserviamo che non è necessario conoscere il valore di Z_ϵ .

Proposizione 2.3.1. *Se i_1, \dots, i_k sono gli stati in cui H raggiunge il minimo assoluto, allora per $\epsilon \rightarrow 0$ la distribuzione π^ϵ converge alla distribuzione uniforme su i_1, \dots, i_k .*

Dimostrazione. Iniziamo con l'osservare che sommando alla funzione H una costante α il numeratore e il denominatore di π^ϵ risultano moltiplicati entrambi per e^α , quindi la definizione di π^ϵ non cambia. Supponiamo che il

minimo di H sia 0. Si ha

$$\lim_{\epsilon \rightarrow 0} e^{-\frac{H(i)}{\epsilon}} = \begin{cases} 1 & \text{se } i \text{ è uno dei punti di minimo} \\ 0 & \text{altrimenti} \end{cases}$$

e

$$\lim_{\epsilon \rightarrow 0} Z_\epsilon = k$$

Quindi

$$\lim_{\epsilon \rightarrow 0} \pi(i)^\epsilon = \begin{cases} \frac{1}{k} & \text{se } i \text{ è uno dei punti di minimo} \\ 0 & \text{altrimenti} \end{cases}$$

In particolare se il punto di minimo assoluto è unico

$$\lim_{\epsilon \rightarrow 0} \pi(i)^\epsilon = \begin{cases} 1 & \text{se } i \text{ è il punto di minimo} \\ 0 & \text{altrimenti} \end{cases}$$

□

Questa proposizione mostra che se ϵ è piccolo la distribuzione π^ϵ si concentra sugli stati dove H assume un valore minore. Abbiamo trovato un metodo per determinare il minimo assoluto di una funzione H sull'insieme $\{1, \dots, m\}$. Supponiamo che l'insieme su cui è definita H abbia una cardinalità molto grande, per trovare il punto di minimo assoluto non si può ricorrere ad un calcolo diretto. In questo caso il problema di ottimizzazione globale si risolve utilizzando l'algoritmo descritto sopra.

Una famosa variazione di questo algoritmo è conosciuta con il nome di *simulated annealing*. A differenza del procedimento appena descritto ad ogni iterazioni si considera un valore di ϵ sempre più piccolo. Si sceglie una successione $(\epsilon(n))_n$ che sia decrescente e che tenda a 0 abbastanza lentamente. Il risultato di questo algoritmo è una CM che converge ad una distribuzione concentrata sui punti di minimo assoluto della funzione H .

Esempio 2.5. (*Il problema del commesso viaggiatore*) Il commesso viaggiatore parte dalla città 0 e poi visita tutte le città $1, 2, \dots, r$ tornando infine alla città 0. Sia i_1, i_2, \dots, i_r una possibile permutazione di $1, \dots, r$ tale che il commesso parte da 0, poi visita tutte le città nell'ordine i_1, i_2, \dots, i_r e torni

in 0. Sia S l'insieme delle permutazioni delle r città. Vogliamo determinare un tragitto $0 \rightarrow i_1 \rightarrow i_2 \rightarrow \dots \rightarrow i_r \rightarrow 0$ tale che la distanza percorsa dal commesso viaggiatore sia minima. Osserviamo che non importa quale sia la città di partenza perchè dopo aver visitato tutte le città dovrà tornare nella prima città, dove è iniziato il viaggio.

Consideriamo funzione

$$H(\omega) = d_{0,i_1} + d_{i_1,i_2} + \dots + d_{i_{r-1},i_r} + d_{i_r,0}$$

dove $\omega \in S$, $d_{i,j}$ indica la distanza tra due città. Questa funzione indica la lunghezza del percorso ω . Per trovare il percorso più breve cerchiamo il minimo della funzione H su S .

Consideriamo ω e $\hat{\omega}$ due permutazioni. Le due permutazioni si dicono *vicine* se una può essere ottenuta dall'altra scambiando due città nell'ordine di visita, cioè se $\omega = (i_1, \dots, i_r)$, allora $\hat{\omega} = (i_1, \dots, i_{h-1}, i_k, i_{h+1}, \dots, i_{k-1}, i_h, i_{k+1}, \dots, i_r)$. Sia $N(\omega)$ l'insieme dei vicini di ω e $|N(\omega)|$ il numero di elementi di tale insieme. Consideriamo la matrice \mathbf{Q} tale che

$$q(\omega, \omega') = \begin{cases} \frac{1}{|N(\omega)|} & \text{se } \omega' \text{ è un vicino di } \omega \\ 0 & \text{altrimenti} \end{cases}$$

Osserviamo che \mathbf{Q} è simmetrica per come è stata costruita. Inoltre è irriducibile, perchè partendo da una permutazione possiamo ricavarne un'altra effettuando un certo numero di volte lo scambio di due indici. Dopo aver costruito la matrice di transizione \mathbf{Q} procediamo con la ricerca del minimo assoluto di H .

1. Scegliere casualmente una permutazione ω e due indici h, k con $h \neq k$.
2. Considerare la permutazione $\hat{\omega}$, ottenuta da ω scambiando l'elemento di posto k con quello di posto h .
3. Calcolare il valore di H in $\hat{\omega}$.
 - Se $H(\hat{\omega}) < H(\omega)$, si sostituisce $\hat{\omega}$ a ω e si torna al punto 1.

- Se $H(\hat{\omega}) > H(\omega)$, con probabilità $e^{-(H(\hat{\omega})-H(\omega))/\epsilon}$ si sostituisce $\hat{\omega}$ a ω e si torna al punto 1; con probabilità $1 - e^{-(H(\hat{\omega})-H(\omega))/\epsilon}$ si rifiuta $\hat{\omega}$ e si torna al punto 1.

Capitolo 3

Simulazione con Matlab

Ora analizziamo il *problema del commesso viaggiatore* e simuliamo la ricerca del percorso più breve.

Iniziamo scrivendo una funzione per calcolare la distanza percorsa dal commesso viaggiatore a seconda del cammino intrapreso, $\mathbf{d}=\text{distanza}(\mathbf{n},\mathbf{x},\mathbf{c})$.

Consideriamo come unità di misura delle distanze il chilometro.

- I dati di input sono \mathbf{n} il numero delle città da visitare, $\mathbf{x}=(x_1,\dots,x_n)$ una permutazione di $(1,\dots,n)$ che è il primo percorso considerato e \mathbf{c} una matrice, l'elemento $c(i,j)$ indica la distanza tra la città i e la città j .
- Porre $\mathbf{d}=c(x_n,x_1)$, questo indica la distanza tra l'ultima città e quella da cui è iniziato il viaggio.
- Per ogni $k=1,\dots,n-1$ si somma al valore di \mathbf{d} l'elemento della matrice $c(k,k+1)$
- Il dato di output è \mathbf{d} , la distanza totale percorsa.

Procediamo descrivendo la funzione per cercare il minimo, $[\mathbf{x}\ \mathbf{sx}]=\text{tps}(\text{iter},\mathbf{n},\mathbf{c})$.

- I dati di input sono iter , il numero di iterazioni, \mathbf{n} il numero delle città da visitare e la matrice \mathbf{c} .

- Porre T uguale ad un certo valore, corrisponde a ϵ dell'esempio 2.5 e x uguale ad una permutazione di $(1, \dots, n)$. Indicare con s_x la lunghezza del cammino x .
- Per ogni $k = 1, \dots, \text{iter}$
 - (a) Generare due interi casuali tra 1 e n , siano i, j e un numero casuale tra 0 e 1, sia u .
 - (b) Trovare la permutazione y , scambiando tra loro l' i -esimo e il j -esimo elemento di x . Si calcola la lunghezza di questo nuovo percorso, sia s_y .
 - (c) Se $s_y < s_x$ o $u < e^{-(s_y - s_x)/T}$ si pone $x = y$ e $s_x = s_y$.
- I dati di output sono x , il percorso intrapreso e s_x , la lunghezza di tale percorso.

Supponiamo che il commesso visiti 12 città. Prima di tutto costruiamo una matrice 12x12 tale che l'elemento di posto ij , con $i \neq j$ indichi la distanza tra la città i e la città j . Simuliamo la catena per 60 passi e per 600 passi. Si ottengono i seguenti grafici.

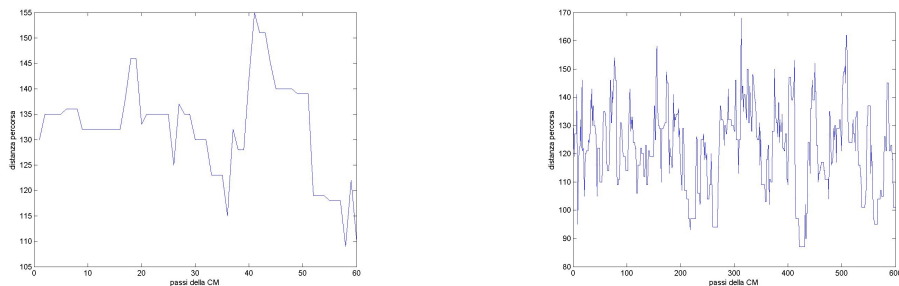


Figura 3.1: Traiettorie della catena con T fissata

Nel primo caso il commesso viaggiatore visitando le città nel seguente ordine 9 10 12 2 5 4 8 7 6 3 11 1 9 percorre 110 km. Nel secondo caso visitando le città nel seguente ordine 10 5 4 9 11 3 1 8 6 7 2 12 10 percorre 101 km.

Nelle seguenti tabelle la prima riga indica il cammino più breve tra quelli trovati durante l'implementazione della funzione, la seconda indica il risultato ottenuto.

	distanza percorsa	città visitate
minimo	109 km	12 10 9 1 5 4 8 7 6 3 11 2 12
risultato	110 km	9 10 12 2 5 4 8 7 6 3 11 1 9

	distanza percorsa	città visitate
minimo	87 km	3 7 2 1 8 9 4 6 7 2 12 10 3
risultato	101 km	10 5 4 9 11 3 1 8 6 7 2 12 10

Tabella 3.1: Distanze e cammini a T fissata

Osserviamo che in entrambi casi il percorso trovato non coincide con quello più breve tra quelli considerati.

Ora ad ogni iterazione aggiorniamo il valore di T, ponendo $T=0.99T$. Si ottengono i seguenti grafici.

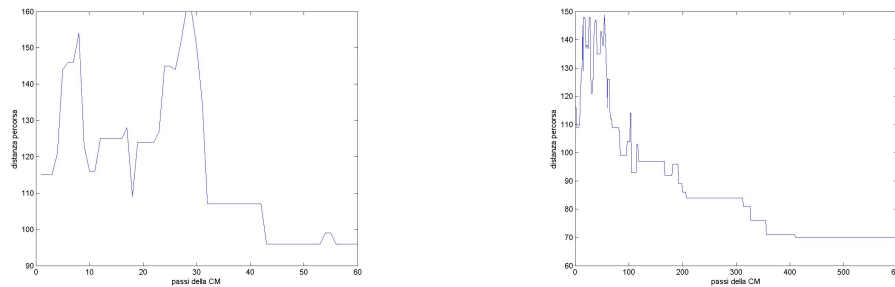


Figura 3.2: Traiettorie della catena con T decrescente

Nel primo caso il commesso viaggiatore visitando le città nel seguente ordine 4 3 9 10 12 11 2 7 8 1 5 6 4 percorre 96 km. Nel secondo caso visitando le città nel seguente ordine 8 1 2 11 3 7 6 4 5 12 10 9 8 percorre 70 km.

	distanza percorsa	città visitate
minimo	96 km	4 3 9 10 12 11 2 7 8 1 5 6 4
risultato	96 km	4 3 9 10 12 11 2 7 8 1 5 6 4

	distanza percorsa	città visitate
minimo	70 km	8 1 2 11 3 7 6 4 5 12 10 9 8
risultato	70 km	8 1 2 11 3 7 6 4 5 12 10 9 8

Tabella 3.2: Distanze e cammini con T decrescente

Osserviamo che prendendo ad ogni iterazione un valore minore di T il risultato dell'algoritmo coincide con il minimo cammino tra quelli considerati. Inoltre aumentando il numero di iterazioni si nota come la catena si concentra sugli stati su cui la distanza percorsa dal commesso viaggiatore è minore.

Appendice A

Funzioni Matlab

```
function d=distanza(n,x,c)
d=c(x(n),x(1));
for i=1:n-1
    d=d+c(x(i),x(i+1));
end
return
end
```

```
function [x sx]=tps(iter,n,c)
T=10;
x=randperm(n); %permutazione di(1,...,n)
sx=distanza(n,x,c);
lung_h_percorsi=zeros(1,iter);
for k=1:iter
    y=x;
    i=randi(n); %indici da scambiare
    j=randi(n);
    y([i j])=x([j i]);
    sy=distanza(n,y,c);
    u=rand();
    if sx>=sy
```

```
        x=y;
        sx=sy;
    elseif u<exp(-(sy-sx)/T)
        x=y;
        sx=sy;
    end
    lungh_percorsi(k)=sx;
    % T=.99*T; da aggiungere per far diminuire T ad ogni iterazione
end
figure
plot(lungh_percorsi)
xlabel('passi della CM')
ylabel('distanza percorsa')
return
end
```

Bibliografia

- [1] Sheldon M. Ross, *Introduction to Probability Models*, Elsevier Academic Press, 2010.
- [2] Sheldon M. Ross, *Simulation*, Elsevier Academic Press, 2002.
- [3] F. Biagini, M. Campanino, *Elementi di probabilità e statistica*, Springer, Milano, 2006.
- [4] P. Baldi, *Calcolo delle probabilità e statistica*, McGraw-Hill, 1998.
- [5] <http://www.math.wsu.edu/faculty/genz/416/lect/l10-4.pdf>