Università degli Studi di Roma "Roma Tre"
Anno Accademico 2002-2003

Facoltà di Scienze Matematiche Fisiche e Naturali
Corso di Laurea in Matematica

Sintesi della tesi di laurea

Alcuni modelli matematici per la genetica

Relatore
Prof. Brunello Tirozzi

Laureanda
Daniela Bianchi
Matricola: 24489\7

Lo studio svolto in questa tesi si fonda essenzialmente sul lavoro di Feng e Tirozzi ([23]), proseguendo le tematiche di ricerca in esso affrontate. In quel lavoro gli autori analizzano la rete neuronale artificiale creata da Kohonen (vedi [15]) e sviluppata ulteriormente dall'autore in lavori successivi.

Una rete di Kohonen è una rete neurale artificiale auto-organizzante costituita da un insieme di neuroni che si organizzano in qualche struttura topologica. Per rete neuronale artificiale si intende una struttura su un insieme di nodi, di unità, che simulano le cellule nervose ed il loro modo di trasmettersi i dati. Essa rappresenta, quindi, un sistema distribuito ad alto parallelismo e basato sul connessionismo. Le sue caratteristiche principali sono l'apprendimento, l'adattività, la flessibilità e l'auto-organizzazione dei dati.

Una rete di Kohonen è costituita da una serie di neuroni di *input*, che servono a calcolare la somma pesata di tutti gli *input* e da un singolo strato unidimensionale o multi-dimensionale di neuroni : tali neuroni calcolano l' *output* della rete. Ciascun neurone di *input* è connesso a tutti i neuroni del singolo strato ed ad ognuno di questi è assegnato un vettore peso di n-dimensioni (dove n è la dimensione degli *input*).

Feng e Tirozzi analizzano la convergenza dell'algoritmo di tale rete ricorrendo alla teoria delle supermartingale, sfruttando teoremi di convergenza e proprietà dei tempi di arresto. I risultati ottenuti dai due autori mostrano che la dinamica di apprendimento della rete converge, sotto opportune ipotesi, ad un minimo globale.

In questa tesi viene sfruttata la rete di Kohonen sia per la sua caratteristica di mantenere una struttura topologica fra lo spazio degli *input* e degli *output*, sia per essere non supervisionata. La prima proprietà permette di effettuare una scelta nella associazione degli *input* e degli *output*; la seconda rende la rete di Kohonen più adatta a classificare.

In particolare, la rete di Kohonen fornisce una classificazione dei geni allo scopo di evidenziare la funzione che essi hanno nei processi di espansione oncologica o di reazione delle cellule nervose.

La strumentazione matematica per l'analisi di questa rete si basa esclusivamente su elementi di teoria dei processi stocastici. Particolare importanza riveste il problema della convergenza ed unicità dell'algoritmo della rete di Kohonen, studiato da Feng e Tirozzi e rielaborato in questa tesi indebolendo le ipotesi assunte in ([23]).

Le proprietà di convergenza ed unicità permettono di conferire consistenza al metodo di *clusterizzazione*, ovvero classificazione, usato.

Lo studio della rete di Kohonen considerata viene applicato ai dati da *microarrays*, ovvero ai valori delle espressioni geniche ricavati da risultati sperimentali. In particolare, per *microarray* si intende un supporto solido sul quale sono immobilizzate, in posizioni ben definite, migliaia di sequenze di geni differenti.

Vista l'enorme quantità dei geni, è una grande sfida comprendere e interpretare i tanti dati che se ne ricavano, per questo si ricorre alle tecniche di *clustering*.

Tali tecniche si basano sull'organizzazione di oggetti in gruppi (*clusters*) in

modo tale che tali oggetti all' interno di un *cluster* siano simili in qualche senso. Tra le varie tecniche di classificazione, in questa tesi vengono illustrate solamente quelle che si utilizzano negli studi dei dati da *microarrays*, ossia i metodi quali *hierarchical clustering*, *K-means clustering* e *Self-organizing map*.

Quello che si ottiene, dopo la classificazione dei dati, è di poter supporre la categoria funzionale di alcuni geni, ovvero come possano interagire con l'evoluzione del ciclo vitale di una cellula. Questi risultati permettono di fornire una descrizione genica dei fenomeni presi in esame, consentendo da un lato di approntare strategie ottimali nello studio alla lotta al tumore, dall'altro di individuare le dinamiche sottostanti all'evoluzione delle reazioni delle cellule nervose.

L'analisi proposta viene legittimata dalla implementazione dell'algoritmo su dati empirici forniti dalla dottoressa Arianna Del Signore e dalla professoressa Paola Paggi del Dipartimento di Biologia Molecolare dell'Università La Sapienza di Roma e dal professore Raffaele A. Calogero dell' Unità di Genomica e Bioinformatica del Dipartimento di Scienze Cliniche e Biologiche dell'Università di Torino.

Il linguaggio di programmazione usato è il linguaggio C.

Inoltre, in una importante sezione di questo lavoro di tesi viene presentato un progetto di ricerca per lo studio dell'evoluzione delle proteine. Si tratta in effetti di una ulteriore estensione dei risultati raggiunti precedentemente, che propone la costruzione di un modello consistente sulla dinamica associata alle interazioni tra proteine e geni. Tale progetto di ricerca offre la possibilità di

adattare la teoria matematica legata alle equazioni differenziali e ai processi stocastici su problemi aperti nel campo della genetica.

Il corpo della tesi si sviluppa in sette capitoli nel modo seguente.

Nel Capitolo 1 si affronta la problematica del *clustering*. Questa tecnica tenta di scoprire come distribuire **N** oggetti in **M** *clusters* attraverso la minimizzazione di qualche criterio di ottimizzazione definito su ogni *cluster*.

I metodi di classificazione si dividono principalmente in : gerarchico e non gerarchico.

Esistono differenti tecniche di *clustering* tra le quali : *Hierarchical clustering*, *K-means clustering*, *Self-organizing map*

Il metodo Hierarchical clustering (raggruppamento gerarchico) consiste nell'unire piccoli *clusters* formandone dei più grandi, oppure con il dividerne grandi in più piccoli. I metodi gerarchici sono così ulteriormente divisi in: agglomerativi e divisivi.

La struttura degli algoritmi dei metodi gerarchici agglomerativi è la seguente:

1. Si dispone ogni elemento in un *cluster* contenente un solo elemento (se stesso). Si costruisce una matrice di prossimità, ossia si calcola la distanza tra tutte le coppie non ordinate di elementi (la distanza tra i gruppi è fornita dalla matrice **D**).

$$\mathbf{D}= \begin{pmatrix} 0 & d_{1,2} & .. & .. & d_{1,n} \\ 0 & 0 & .. & .. & d_{2,n} \\ 0 & 0 & .. & .. & .. \\ 0 & 0 & 0 & .. & d_{n-1,n} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

2. Si fondono i due gruppi che possiedono distanza minima ; la distanza

a cui avviene la fusione viene registrata.

3. Si calcola la distanza tra il nuovo gruppo, nato dalla fusione precedente, e i gruppi già esistenti. Si eliminano due righe e colonne dalla matrice **D** e vengono sostituite da una singola riga e colonna che contengono le nuove distanze.

4. Si ripetono i passi 2 e 3 fino a quando non si giunge ad una configurazione con un solo gruppo.

I metodi proposti differiscono per la modalità di calcolo della distanza tra gruppi al punto 3.
Nei metodi gerarchici divisivi la configurazione iniziale prevede che tutti gli elementi siano raggruppati in un unico *cluster*. Successivamente si divide il gruppo in due, poi ancora in due e così via fino ad arrivare ad una configurazione in cui ogni *cluster* contiene un solo elemento.
L'algoritmo è il seguente:

1. Si individua una coppia di elementi che presentano distanza massima

2. Si attribuiscono gli elementi rimanenti ai due gruppi corrispondenti agli elementi individuati al punto 1, in base alla distanza minima da questi ultimi.

3. Si iterano i passi precedenti finchè si avranno n gruppi.

Il metodo K-means fa parte della categoria dei metodi non gerarchici. Tali metodi cercano, direttamente, di decomporre l'insieme dei dati in un insieme di *clusters* disgiunti.

Il metodo *K-means clustering* è utile solamente quando si ha già una ipotesi sul numero di *clusters* che gli elementi dovrebbero formare, dato che va impostato a priori il numero dei desiderati gruppi.

In questo metodo viene definita una funzione di costo $c : \{\mathbf{X} : \mathrm{X} \subseteq \mathrm{S} \rightarrow R^{+}\}$ (dove S è l'insieme degli oggetti da classificare) la quale associa un costo ad ogni *cluster*.

L'obiettivo dell'algoritmo *k-means*, dopo aver calcolato il centroide di ogni *cluster* $\mathrm{S_i}$ (denominato $\overline{x}^i$), è minimizzare la seguente funzione di costo:

$$c(\mathrm{S_i}) = \sum_{r=1}^{|\mathrm{S_i}|} d(\overline{x}^i, x_s^i) \tag{1}$$

L'algoritmo *k-means* è il seguente:

1. Ripartisce casualmente gli elementi in k *clusters*
2. Calcola il centroide (la media o mediana) per ognuno dei *clusters*
3. Calcola la distanza tra ogni oggetto e il centroide di ogni *clusters*
4. Sposta ogni elemento nel *cluster* la cui media o mediana è più vicina a quell'elemento
5. Ricalcola i centroidi dei *clusters* interessati dalla riallocazione
6. Ripete le operazioni 3, 4, 5 finchè non sono necessari più spostamenti o ha raggiunto il massimo numero di iterazioni impostate.

Self-organizing map (SOM) è una ben conosciuta rete neurale artificiale (ANN).

L'architettura della SOM è molto semplice: è ad un singolo strato.
Gli elementi da classificare vengono presentati ai nodi di *input* e connessi con i nodi dell' *output*. I pesi tra i nodi di *input* ed *output* sono iterativamente modificati fino a quando un criterio di convergenza è soddisfatto.

Nel Capitolo 2 si presentano elementi della teoria dei processi stocastici, quali : le martingale, tempo di Markov, processso di Markov. Inoltre, allo scopo di illustrare il teorema di convergenza di una rete di Kohonen, si sono presentati e dimostrati alcuni lemmi e teoremi che di seguito si enunceranno.

Definizione 0.0.1. Sia L l'operatore su $V(n,x)$,$x \in \mathbb{R}^N$ tale che :

$$LV(n,x) = \int \mathrm{P}(n,x,n+1,dy)\ [V(n+1,y) - V(n,x)] \tag{2}$$

Lemma 0.0.1. *Sia $x(n)$ un processo di Markov e $\mathbb{E}V(n,x) < +\infty$.*
Allora:

$$\mathbb{E}\ [V(n+1,x(n+1))] - \mathbb{E}[\ V(n,x(n))] = \mathbb{E}LV(n,x(n)) \tag{3}$$

$$\mathbb{E}\ [V(n+1,x(n+1))] - \mathbb{E}\ [V(s,x(s))] = \sum_{u=s}^{n} \mathbb{E}LV(u,x(u)) \tag{4}$$

Teorema 0.0.2. *Si suppone che esista una funzione $V(n,x) \geq 0$ che soddisfi:*

$$LV(n,x) \leq -\eta(n)\varphi(n,x) \tag{5}$$

dove $n \geq 0$, $x \in \mathbb{R}^N$ e φ una funzione non negativa tale che

$$\inf_{n \geq Q, x \in U_{\varepsilon,R}} \varphi(n,x) > 0 \tag{6}$$

per tutti gli $R > \varepsilon > 0 \quad Q = Q(\varepsilon, R)$
e $U_{\varepsilon,R} = V_\varepsilon(B) \cap \{x(n) : |x(n)| < R\}$ con $V_\varepsilon = \mathbb{R}^{N \times M} \backslash U_\varepsilon(B)$,

$U_\varepsilon(B) = \{x : \rho(x, B) < \varepsilon\}$ *dove* $\rho(x, B) = inf_{y \in B}\ (x, y)$

e $B \subseteq \mathbb{R}^{N \times M}$.

Inoltre sia

$$\sum_{n=1}^{+\infty} \eta(n) = +\infty\ ,\ \eta(n) > 0 \tag{7}$$

e:

$$inf_{n \geq 0} V(n, x) \longrightarrow +\infty\ \ |x| \to +\infty \tag{8}$$

Allora si ha:

$$\mathrm{P}\{\sup_n |x(n)| = R < +\infty\} = 1 \tag{9}$$

$$\mathrm{P}\{\sum_{u=0}^{+\infty} \eta(u)\varphi(u, x(u)) < +\infty\} = 1 \tag{10}$$

$$\mathrm{P}\{\liminf_{n \to +\infty} \rho(x(n), B) = 0\} = 1 \tag{11}$$

Inoltre, in questo capitolo, si sono riportati e dimostrati un teorema di convergenza di una supermartingala e due lemmi propedeutici alla dimostrazione del teorema.

Definizione 0.0.3. Data la sequenza $X(t)$ con $t = 1, ..., T$ ed un intervallo chiuso $[a, b]$, $b > a$, sia $\bar{t} = \{t_1, t_2, ..t_{2H}\}$ la successione definita nel modo seguente:

- t_1 il primo tempo per cui $X(t) < a$
- t_2 il primo tempo dopo t_1 per il quale $X(t) > b$
- t_3 il primo tempo dopo t_2 per il quale $X(t) < a$

e così via. Inoltre sia t_{2H} il massimo degli istanti t in cui $X(t)$ esce da $[a, b]$, passando per b.

Lemma 0.0.2. *Sia* $H = H_{a,b}^{(T)}$ *il numero degli elementi della successione* $\bar{t} = \{t_1, t_2, ..t_{2H}\}$. *Allora*

$$(b-a)H_{a,b}^{(T)} \leq (a - X(T))^+ + \sum_{t=1}^{T-1} I(t)(X(t+1) - X(t)) \tag{12}$$

dove i termini della sequenza $I(t),\ t = 1, ..., T-1$ *valgono o 0 o 1 e* $I(t)$ *è univocamente determinata dai valori* $X(1), ..., X(t)$ *e* $B^+ = \max(0, B)$.

Lemma 0.0.3. *Sia* $X(t) = (X(t, \omega), \mathcal{F}_t)_{n\geq 0}$ *una supermartingala non negativa e* $H = H_{a,b}^{(T)}(\omega)$ *variabile aleatoria definita come nel lemma (0.0.2). Allora*

$$\mathbb{E}H_{a,b}^{(T)}(\omega) \leq \frac{\mathbb{E}(a - X(T))^+}{b-a} \tag{13}$$

Teorema 0.0.4. *Sia* $(X(t)\mathcal{F}_t),\ t = 1, 2..,$ *una supermartingala non negativa. Allora esiste il limite* $X(\omega) = \lim_{t\to+\infty} X(t)$ *con probabilità 1.*

Nel Capitolo 3 si illustra la rete di Kohonen e l'analisi formale della rete. In dettaglio nella rete di Kohonen è presente un singolo strato di unità di *output* $\mathcal{O}_i(n),\ i = 1, ...N$ al tempo n (con tempo si indica il passaggio dell' elemento di apprendimento), ognuna connessa ad un insieme di *inputs* $\xi_j(n),\ j = 1, ..., M$. Il vettore peso associato al neurone i-esimo è $\omega_i(n) = (\omega_{ij}(n), j = 1, ...M)$.

I segnali di *input* sono distribuiti con una legge di probabilità P non nota. Per ogni presentazione dell' *input* $\xi_j(n)$ alla rete, si sceglie una unità di *output*, chiamata vincitore. Il vincitore è l'unità di *output* il cui peso ha la distanza più piccola rispetto all' *input* al tempo n

$$||\omega_v(n) - \xi(n)||$$

dove $\xi(n) = (\xi_j(n), j = 1, ..., M)$ e $||.||$ rappresenta la norma Euclidea. Sia $\bar{\mathrm{I}}(.,.)$ la funzione caratteristica che definisce il neurone vincitore

$$\bar{\mathrm{I}}(\omega_v(n), \xi(n+1)) = \mathrm{I}_{\{||\omega_{\mathrm{v}}(\mathrm{n})-\xi(\mathrm{n}+1)||<||\omega_{\mathrm{j}}(\mathrm{n})-\xi(\mathrm{n}+1)||,\ \mathrm{j}\neq\mathrm{i}\}}$$

dove $\mathrm{I_A}$ è la funzione indicatrice, cioè $\mathrm{I_A}(x) = 1$ se $x \in \mathrm{A}$ e $\mathrm{I_A}(x) = 0$ se $x \notin \mathrm{A}$.

Ogni vettore peso è allora aggiornato con la seguente regola:

$$\omega_{ij}(i+1) = \omega_{ij}(n) + \eta(n)\Lambda(i,v)\bar{\mathrm{I}}(\omega_v(n), \xi(n+1)) \ . \ (\xi_j(n+1) - \omega_{ij}(n)) \quad (14)$$

per ogni $i = 1, ..., N$ e $j = 1, ...M$ o in forma vettoriale

$$\omega_i(n+1) = \omega_i(n) + \eta(n)\Lambda(i,v)\bar{\mathrm{I}}(\omega_v(n), \xi(n+1)) \ \cdot \ (\xi(n+1) - \omega_i(n)) \quad (15)$$

con $\eta(n)$ una funzione positiva tale che $\eta(0) < 1$,$\eta(n) \geq \eta(n+1)$ e $\Lambda(i,v)$ una funzione decrescente di $||i - v||$ compresa tra 0 ed 1.

Il fattore determinante nell' auto-organizzazione della rete è la scelta della funzione di vicinanza, $\Lambda(i,j)$, ovvero dell' ampiezza dell' area di attivazione attorno al vincitore, individuata dal raggio di interazione. Una scelta conveniente è quella di basarsi sulla norma euclidea fra unità i-esima ed il vincitore:

$$\Lambda(i,v) = \begin{cases} 1 \ se \ ||i-v|| \leq s \\ 0 \ altrimenti \end{cases} \quad (16)$$

Con $s = 1$, in una situazione mono-dimensionale la zona di attivazione comprende il vincitore e le due unità ai lati; per una disposizione bidimensionale, sono comprese le otto unità circostanti.

Un' altra possibile scelta è ridefinire la funzione di vicinanza in modo tale

che assuma valori continui (fra 0 e 1) in funzione della distanza $||i - v||$.

$$h(i, v, n) = \exp(\frac{-||i - v||^2}{\sigma(n)^2}) \tag{17}$$

dove $\sigma(t)$ è una funzione decrescente in n opportunamente scelta che controlla l' ampiezza della funzione. Spesso è usata:

$$\sigma(n) = \sigma_i(\frac{\sigma_f}{\sigma_i})^{\frac{n}{n_{max}}}$$

dove n_{max} rappresenta il numero massimo di iterazioni dell' algoritmo e σ_f, σ_i rispettivamente il valore finale ed iniziale del parametro.

Sono state pubblicate molte ricerche sulle reti di Kohonen ed i problemi matematici affrontati in questi studi teorici prevedono:

1. *La ricerca di una funzione di energia.* Il problema è trovare se esiste una funzione di energia dei vettori peso e dei vettori di input che viene minimizzata dalla legge di aggiornamento dei pesi.

2. *L' ordinamento delle unità della mappa.* Il problema è scoprire se i vettori peso, sottoposti alla legge di aggiornamento, si modificano in modo tale che l'ordinamento della mappa è assicurato.

3. *La convergenza dei vettori peso.* Il problema è scoprire le caratteristiche dei vettori peso dopo il processo di apprendimento.

Lo studio della rete di Kohonen e relative proprietà sarebbe di gran lunga semplificato se fosse possibile provare che la regola dell' aggiornamento dei pesi (14) la si può derivare da qualche funzione di energia che viene minimizzata, dato che tecniche tradizionali potrebbero essere usate per assicurare la

convergenza dell' algoritmo.

La possibilità dell' esistenza di funzioni di energia è stata studiata in molti articoli, per esempio in ([Kohonen 1991a],[16]), [Erwin et al. 1992b, [11]] e ([Tolat 1990 [24]]). In ([11]) viene mostrato che nel caso generale non è possibile trovare una tale funzione di energia e che la miglior soluzione è costruire un sistema che consiste in un insieme di funzioni di energia, una per ogni vettore peso (proposta in [24]). Tale insieme di funzioni di energia è descritto nel dettaglio in ([11]).

La proprietà dell' ordinamento è stata esaminata in molti articoli a partire dal 1981 (vedi [15], [11], [22]). Le dimostrazioni sono ristrette al caso unidimensionale dove l'ordine della mappa lo si può facilmente definire. Infatti per configurazione ordinata nel caso unidimensionale si intende la mappa dei dati di *input* tali che

$$|r-s| < |r-q| \Leftrightarrow |\omega_r - \omega_s| < |\omega_r - \omega_q|,\ \forall r,s,q \in \{1,2...,N\}$$

Esistono due configurazioni ordinate che sistemano i vettori peso o in un ordine ascendente o discendente.

La proprietà dell' ordinamento di una rete di Kohonen può esser così enunciata

- Dato un insieme di N numeri scelti in modo casuale che rappresentano i pesi in una mappa unidimensionale e un insieme di dati di *input* $x(n)$, il processo di una mappa di Kohonen definito dalla (14) aggiornerà i pesi in modo tale che si ordineranno ($\omega_i < \omega_{i+1},\ \forall i < N$ oppure

$\omega_i > \omega_{i+1}, \ \forall i < N$). Dopo che i pesi si sono ordinati, anche con ulteriori aggiornamenti essi non perderanno il loro ordine.

Un approccio usato per dimostrare questa proprietà è presentato in ([Kohonen 1982a] [15]) ed è quello di definire un parametro di ordine D e mostrare che ci sono molti più passi di apprendimento nell' algoritmo dove D decresce rispetto a quelli in cui aumenta.

Il parametro D è

$$\mathrm{D} = \sum_{i=2}^{N} \|\overline{\omega_i} - \overline{\omega_{i-1}}\| - \|\overline{\omega_N} - \overline{\omega_1}\| \geq 0 \tag{18}$$

l'uguaglianza la si ha se e solo se i neuroni sono ordinati; ovviamente D evolve durante l'apprendimento.

Un risultato più generale sulla proprietà dell' ordinamento è illustrato in ([Taylor] [22]). Nell'articolo viene presentata un' interpretazione geometrica che dá una condizione necessaria e sufficiente del decrescere del valore D e che può esser estesa facilmente al caso multi- dimensionale dello spazio di *input*.

Nel Capitolo 4 si analizza la convergenza della rete di Kohonen, utilizzando l'approccio presentato in ([Feng-Tirozzi 1997],[23]).

Ricordando le notazioni introdotte nel terzo capitolo

Definizione 0.0.5. Per un dato sottoinsieme compatto $\Omega \in \mathbb{R}^M$,la tassellazione di Voronoi $\Pi(y) = (\Pi(y)_i, i = 1, ..N))$ associata alla famiglia di vettori $y_1, ..., y_N$ è la partizione di Ω data da

$$\Pi_i(y) = \{x, \|y_i - x\| \leq \|y_j - x\|, j\} \ i = 1, ...N \tag{19}$$

Quindi una cella di Voronoi di una unità i contiene quei punti che sono più vicini al vettore peso ω_i, piuttosto che ad altri vettori peso.

Sia

$$g(y_1, y_2, ..., y_N; \omega_1, \omega_2, ...\omega_N) = \sum_{i=1}^{N}(y_i - \omega_i) \cdot (\int_{\Pi(y)_v} \Lambda(v, i)(x - y_i)f(x)dx). \tag{20}$$

dove $\omega = (\omega_1, \omega_2, ...\omega_N)$,$y = (y_1, y_2, ..., y_N) \in \mathbb{R}^M$.

f è la densità della distribuzione P con supporto compatto Ω di $\mathbb{R}^M$, $\Pi(y)$ è la tassellazione di Voronoi associata ad y .

L'idea presentata nell'articolo è quella di far notare che la funzione g può essere interpretata come il contributo principale della derivata di una funzione di Liapunov. Usando questa terminologia la dinamica (14) converge al minimo globale $y_1 = \omega_1, ..., y_N = \omega_N$ se le ipotesi del teorema che di seguito sono enunciate sono soddisfatte.

Definiamo:

$\Theta \equiv$\{l'insieme di tutte le tassellazioni di Voronoi associate ad $\{\omega_1(n), ..., \omega_N(n)\}$,per tutti gli n\}

Per $y \in \mathbb{R}^M$ si userà la convenzione che $y \in \Theta$ implica che esiste una tassellazione di Voronoi $\Pi(y)$ tale che $\{\Pi(y)_i, i = 1...N\} \in \Theta$.

Sia $\mathcal{F}_n$ la σ-algebra generata dalle variabili $\xi(k),\ k \leq n$.

Lemma 0.0.4. *Se*

$$\sum_{i=1}^{N}[\mathbb{E}(\|\omega_i(n+1) - \omega_i\|^2|\mathcal{F}_n) - \|\omega_i(n) - \omega_i\|^2] \leq 0 \tag{21}$$

si ha che

$$\sum_{i=1}^{N} \|\omega_i(n+1) - \omega_i\|^2$$

è una supermartingala.

Di seguito si presenteranno i risultati di Feng e Tirozzi in [23] che sono stati rielaborati in modo da indebolire, nel teorema principale, una condizione richiesta nelle ipotesi, ossia $\sum_n \eta(n)^2 < +\infty$. Questa analisi ha trovato conferma anche nelle apllicazioni, infatti si è notato che la scelta di funzioni con tale condizione non porta alla convergenza dell'algoritmo.

Teorema 0.0.6. *Ricordando la definizione dell' algoritmo (15), se esiste un unico punto* $\omega = (\omega_1, \omega_2, ...\omega_N) \in \mathbb{R}^M$ *tale che:*

$$g(y_1, y_2, ..., y_N; \omega_1, \omega_2, ...\omega_N) \leq 0 \ \in \Theta \tag{22}$$

dove l'uguaglianza la si ha se e solo se $y_i = \omega_i \ i = 1, ...N$ *e valgono:*

$$\sum_{n=1}^{+\infty} \eta(n) = +\infty \qquad \lim_{n \to +\infty} \eta(n) = 0 \tag{23}$$

allora quasi ovunque

$$\lim_{n \to +\infty} \omega_i(n) = \omega_i \qquad i = 1, ..., N$$

Lemma 0.0.5. *Sotto le ipotesi del teorema (0.0.6) esiste una costante* $\mathrm{B}(\varepsilon) > 0$ *tale che quasi ovunque*

$$\tau(\varepsilon) < \mathrm{C}(\varepsilon)$$

Nel Capitolo 5 si descrive la metodica dei DNA *microarrays*. Un microarray è definito come un supporto solido sul quale sono immobilizzate, in

posizioni ben definite, migliaia di sequenze di geni differenti.

I DNA *microarrays* sono costruiti attaccando alla superficie del supporto molecole di DNA a singolo filamento, che rappresentano migliaia di geni diversi, ognuno assegnato ad uno specifico sito sul minuscolo dispositivo. Ogni sito può contenere da alcune migliaia ad alcuni milioni di copie di un filamento di DNA. La metodica dei DNA *microarrays* consiste nell'ottenere dalle cellule sotto esame un cDNA, ossia la trascrizione dell'mRNA, estratto dalle cellule, nel più stabile DNA complementare. Il cDNA viene marcato con fluorescenza o radiattivo e applicato al chip dove si formerà un legame se il cDNA troverà la sua sequenza complementare di basi. La formazione di questo legame sta a significare che il gene rappresentato dal DNA sul chip era attivo, cioè espresso, nel campione analizzato. Successivamente scansionando il chip si ricavano le misurazioni delle intensità delle fluorescemze o del radioattivo

Questa tecnica è molto vantaggiosa perchè permette di misurare contemporaneamente migliaia di sequenze diverse, dando l'opportunità di stabilire quali geni siano attivi all'interno della cellula e quale sia il loro livello di espressione trascrizionale; inoltre data la dimensione ridotta dei chip il consumo dei reagenti è minore.

Data l'enorme quantità dei geni, è una grande sfida comprendere e interpretare i tanti dati che se ne ricavano, per questo si ricorre alle tecniche di *clustering*.

Nel Capitolo 6 si offre uno studio delle applicazioni della rete di Kohonen

ai dati da *microarrays*. L' obiettivo del lavoro svolto è quello di classificare le espressioni geniche in gruppi con *pattern* simili di espressione, dal momento che si suppone che geni co-espressi siano co-regolati ed intervengano così nella stessa funzione genica.

Il primo passo è stato implementare, in linguaggio C, l'algoritmo di una rete di Kohonen unidimensionale (dato che è in questo caso che si hanno più risultati matematici validi) e studiare la convergenza immettendo come dati di *input* le espressioni geniche.

Si è data una breve illustrazione dei dati usati che non sono altro che i risultati di due diversi tipi di esperimenti da *microarrays*, uno riguarda l'assotomia dei neuroni post-gangliari del sistema nervoso simpatico e l'altro lo studio dei geni coinvolti nel carcinoma lobulare (tumore alla mammella) dei ratti. Lo studio è stato rivolto all'analisi dell'evoluzione del tumore, dallo stadio di iperplasia tipica (presente alla decima settimana dalla contrazione del tumore) allo stadio di carcinoma lobulare (presente alla ventiduesima settimana).

In questo capitolo si è analizzato numericamente il problema della convergenza dell'algoritmo della rete di Kohonen, scoprendo così i motivi di una non possibile convergenza e la dipendenza dal parametro $\eta(n)$. Successivamente si sono classificati i dati, ossia suddivisi in base alla vicinanza dai rappresentanti. Una volta classificati i dati si è studiata l'evoluzione temporale dei geni dal momento che si sono avuti a disposizione esperimenti a tempi differenti, allo scopo di studiare da una parte l'evoluzione del tumore, dall'altra la rinascita delle terminazioni nervose.

Il lavoro svolto ha permesso di fare delle supposizioni su alcuni geni di cui non si conosce la funzione in modo da permettere di indirizzare i biologi sulla scelta di esperimenti per verificare che funzione effettivamente svolgano.

Nel Capitolo 7 si propone un progetto di ricerca che tratta un modello dell'attività dei geni, cioè le interazioni tra proteine-DNA e, indirettamente, tra DNA-DNA, allo scopo di ottenere un quadro globale coerente dell'attività cellulare. Si è analizzata una particolare classe di proteine, i fattori trascrizionali (TFs), dal momento che sono responsabili della regolazione dell'espressione genica. Questo processso è controllato dalla trascrizione, dalla traslazione e dalle attività cellulari; l'insieme di tali elementi viene definito regolazione genica oppure regolazione trascrizionale.
Si è posta l'attenzione proprio sulla regolazione genica, illustrando le reazioni chimiche che ne sono la struttura portante.
Una generica reazione chimica è del tipo:

$$n_a A + n_b B \xrightarrow{k} n_c C + n_d D$$

ed indica che n_a molecole del tipo A reagiscono con n_b molecole del tipo B formando n_c molecole del tipo C e n_d del tipo D. I termini a sinistra della freccia si dicono reagenti e quelli a destra prodotti. I termini n sono detti coefficienti stechiometrici e sono degli interi.
Di seguito si è mostrato come tradurre un'equazione chimica del tipo

$$A \xrightarrow{k} A' \tag{24}$$

in un modello di equazioni differenziali o in un modello stocastico. Nell'ap-

proccio con le equazioni differenziali la (24) indica che la concentrazione di A decresce in corrispondenza di un aumento della concentrazione di A'. Per un piccolo intervallo di tempo, dt, il tasso di cambiamento é dato da $k \times [A]$, dove $[A]$ è la concentrazione di A.

Nell'approccio stocastico la (24) indica che una singola molecola di A si è convertita in una molecola di A'; il numero totale delle molecole di A decresce passo per passo e contemporaneamente il numero totale delle molecole di A' aumenta. La probabilità che questo evento avvenga in un tempo dt è data da: $k \times \{\sharp A\}$, dove $\{\sharp A\}$ è il numero delle molecole di A presenti.

Una volta compreso come interpretare le equazioni chimiche si è potuto mostrare quali equazioni differenziali illustrano il modello chimico.

Date le equazioni

$$X + DNA \xrightarrow{k_1} X \cdot DNA \tag{25}$$

$$X \cdot DNA \xrightarrow{k_{-1}} X + DNA \tag{26}$$

dove X è una generica proteina e con $X \cdot DNA$ si indica che X si è legata al DNA.

Dalle equazioni (25), (26) si deduce che l'equazione per $[X \cdot DNA]$ è

$$\frac{[X \cdot DNA]}{dt} = k_1[X][DNA] - k_{-1}[X \cdot DNA] \tag{27}$$

Il primo termine a destra esprime la produzione del nuovo $[X \cdot DNA]$ dovuta all'equazione (25), mentre il secondo la degradazione dell'esistente $[X \cdot DNA]$ dovuta alla (26). Con la (27) e le concentrazioni iniziali di ogni tipo di molecola presente, si può trovare $[X \cdot DNA]$ come funzione del tempo.

Nel modello stocastico, dall' equazione (25) si deduce che lo stato del sistema,

in un piccolo lasso di tempo, cambierà dallo stato $(\{\sharp X\}, \{\sharp DNA\}, \{\sharp X \cdot DNA\})$ allo stato $(\{\sharp X\} - 1, \{\sharp DNA\} - 1, \{\sharp X \cdot DNA\} + 1)$ con probabilità $k_1\{\sharp X\}\{\sharp DNA\}dt$. Inoltre, dalla (26), che il sistema passerà dallo stato $(\{\sharp X\} + 1, \{\sharp DNA\} + 1, \{\sharp X \cdot DNA\} - 1)$ allo stato $(\{\sharp X\}, \{\sharp DNA\}, \{\sharp X \cdot DNA\})$ con probabilità $k_{-1}\{\sharp X \cdot DNA\}dt$. Il sistema può anche non cambiare stato e questo avviene con probabilità $1 - k_1\{\sharp X\}\{\sharp DNA\}dt - k_{-1}\{\sharp X \cdot DNA\}$.

Il processo generato dall'evoluzione di questo sistema é una catena di Markov a tempo continuo dato che lo stato del sistema al tempo t, opportunamente discretizzato, dipende solo dallo stato del sistema al tempo $t - 1$.

Per una delucidazione si è portato ad esempio il processo di regolazione genica del procariote Lambda, un virus che infetta *l'Escherichia coli.*

Bibliografia

[1] M.R. Anderberg , *Cluster Analysis for applications* Accademic Press, New York and London, (1973).

[2] Z-P. Lo e Bavarian , *On the rate of convergence in topology preserving neural networks* Biological Cybernetics, Vol.**65** 55–63, (1991).

[3] J. Bower, H. Bolouri, *Computational Modeling of Genetic and Biochemical Networks*, The MIT Press, Cambridge, (2001).

[4] C. Bouton, G. Pag'es, *Self-organization of the one-dimensional Kohonen algorithm with non-uniformly distributed stimuli*, Stochastic Processes and their Applications, Vol.**47**, 249–274, (1993).

[5] P.O. Brown, D. Botstein , *Exploring the new world of the genome with DNA microarrays*,Nature America, (1999).

[6] P. Cugini, M. Curione, C. Cammarota, F. Bernardini, D. Cipriani, R. De Rosa, P. Francia, T. De Laurentis, E. De Marco, A. Napoli, F. Fallucca, *Is a Reduced Entropy in Heart Rate Variability an Early Finding of Silent Cardiac Neurovegetative Dysautonomia in Type 2 Diabetes Mellitus?*, J. Clin. Basic Cardiol., No. 4, 289–295 (2001)

[7] M. Cottrell, J. Fort, G. Pages, *Two or three things that we know about the Kohonen algorithm*, preprint.

[8] D. Duggan, M. Bittner, Y. Chen, P. Meltzer, J.M. Trent, *Expression profiling using cDNA microarrays*, Review Nature America,(1999).

[9] M. Eisen, P.T. Spellman, P.O. Brown, D. Botstein , *Cluster analysis and display of genome-wide expression patterns*, Proc. Natl. Acad. Sci. USA, Vol.**95**, 14863–14868, (1998).

[10] Ed. Erwin, K. Obermayer, K. Schulten , *Self-Organizing maps: stationary states, metastability and convergence rate*, Biological Cybernetics, Vol.**67**, 35–45, (1992).

[11] Ed. Erwin, K. Obermayer, K. Schulten , *Self-Organizing maps: Ordering, convergence properties and energy function*, Biological Cybernetics, Vol.**67**, 47–55, (1992).

[12] G.P. Gladyshev, *On Thermodynamics, Entropy and Evolution of Biological Systems: What is Life from a Physical Chemist's Viewpoint*, Entropy, No.1, 9–20, (1999)

[13] M.L.T. Lee, F.C. Kuo, G.A. Whitmore, J.Sklar, *Importance of replication in microarray gene expression studies: Statistical methods and evidence from repetitive cDNA hybridizations*, PNAS, Vol.**97**, No.18, 9834–9839, (2000).

[14] T. KOHONEN, *Self-Organization and Associative Memory Process*, $3^{rd} Edition, Springer - Verlag, Berlin$, (1989).

[15] T. KOHONEN, *Analysis of a Simple Self-Organizing Process*, Biological Cybernetics, Vol.**44**, 135–140, (1982).

[16] T. KOHONEN, *Self-Organizing maps: optimization approaches*, Artificial Neural Networks, Vol.**1**, 891–990, (1991).

[17] B. MAYER, G. KÖHLER E S. RASMUSSEN, *Simulation and Dynamics of Entropy-driven, molecular self-assembly processes*, Physical Review E, Vol.**55**, No. 4, 4489–4500, (1997)

[18] M.B. NEVL'SON, R.Z. HAS'MINSKII , *Stochastic Approximation and Recursive Estimation* , Translation of Math. Monograph 47, Amer. Math. Soc., Providence, RI, (1976).

[19] E. RENSHAW , *Modelling Biological Populations in Space and Time*, Cambridge University Press,(1993)

[20] H. RITTER, K. SHULTEN , *On the stationary states of Kohonen's Self-Organizing Sensory Mapping*, Biological Cybernetics , Vol.**54**, 99–106, (1986).

[21] P. DE LEENHEER, H.L. SMITH , *Virus Dynamics: a global analysis*, SIAM J. Appl. Math., Vol.**63**, No. 4, 1313-1327.

[22] J.G. TAYLOR, M. BUDINICH, *On the ordering conditions for self-organising maps* , preprint.

[23] J.F. Feng, B. Tirozzi, *Convergence Theorem for the Kohonen Feature mapping Algorithms with VLRPs*, Computer Math. Applic., Vol.**33** No.3, 45–63, (1997).

[24] V.V. Tolat, *An analysis of Kohonen's self-organizing maps using a system of energy functions*,Biological Cybernetics , Vol.**64**,155–164, (1990).

[25] P. Tmayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, *Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation*, Proc. Natl. Acad. Sci. USA, Vol. **96**, 2907-2912, (1999).

[26] J. Vesanto, E. Alhoniemi, *Clustering of the Self-Organizing Map*, Accepted for publication in IEEE Transactions on Neural Networks

[27] D. Williams, *Probability with Martingales*, Cambridge University Press, (1991).