

Reti neurali artificiali e analisi dei dati per la ricerca sociale: un nuovo paradigma?

Giovanni Di Franco

Publicato in *Sociologia e Ricerca Sociale*, n. 56, 1998, pp. 35-75.

1. Premessa

Recentemente l'SPSS per Windows, il pacchetto di analisi dei dati più diffuso nelle scienze sociali, ha reso disponibile un nuovo modulo, Neural Connection, attraverso il quale gli utenti possono gestire le reti neurali artificiali (d'ora in poi Rna) per le loro analisi dei dati. Nella *brochure* di presentazione di Neural Connection si leggono frasi come queste:

"Finalmente le reti neurali sul vostro pc, in un prodotto semplice e potente. Neural Connection è lo strumento di lavoro ideale per chiunque abbia la necessità di costruire modelli interpretativi dei propri dati. Problemi di classificazione, modelli predittivi dei dati, previsioni. Neural Connection uno strumento flessibile e adatto a tutti. Non occorre essere analisti esperti o programmatori per usare Neural Connection. Modelli migliori più velocemente. E' noto che le reti neurali hanno capacità di riconoscimento dei modelli presenti nei dati superiori alle tecniche di analisi statistica. Le reti neurali, infatti, imparano a riconoscere i modelli dall'esperienza, e si adattano anche a situazioni non lineari o mutevoli nel tempo. Questa flessibilità rende superflua una buona parte del lavoro di indagine e di verifica che caratterizza le tecniche di analisi tradizionali, facendo risparmiare molto tempo all'analista. Sfruttare la potenza e la flessibilità di Neural Connection per: classificazione, modelli predittivi, segmentazione dei dati e previsioni. Modelli accurati, facili da costruire. Neural Connection mette a vostra disposizione un'interfaccia a oggetti semplice e intuitiva. Ogni strumento, rappresentato da un'icona, dispone di una propria finestra di dialogo per modificare le impostazioni e i parametri di funzionamento, definiti automaticamente dal sistema." (SPSS per Windows, 1995, p. 12, corsivi aggiunti).

Un linguaggio trionfalistico che in una *brochure* di presentazione si può tollerare; ma anche nella rivista dell'SPSS *Keywords*, che si rivolge ad un pubblico di utenti esperti, il tono non cambia molto:

"Neural Connection vi consente l'analisi dei dati con le reti neurali complementare alle tradizionali analisi statistiche. Gli strumenti di Neural Connection si basano su un approccio che produce la costruzione di modelli ed esplorazioni dei dati più produttive. [...] Vai al di là dei metodi statistici tradizionali con le reti neurali. Le reti neurali analizzano i dati in un modo diverso rispetto ai tradizionali metodi statistici. Quando sono applicate ai dati, le reti neurali apprendono i modelli che esistono nei dati; a differenza dei metodi statistici tradizionali, non assumono nei dati linearità e omogeneità di varianza. Quindi, le reti neurali possono trovare interazioni e modelli non lineari più facilmente che non i metodi statistici tradizionali. Ci sono molti vantaggi a lavorare con le reti neurali. Due vantaggi chiave sono: 1) le reti neurali sono migliori nella manipolazione dei dati irregolari rispetto ai metodi statistici tradizionali, così che i vostri modelli sono più accurati; 2) le reti neurali non impongono un'equazione globale ai dati: in questo modo producono modelli più ricchi." (*Keywords*, 1995, n.58, pp. 2-3, corsivi aggiunti).

Infine, sul manuale dell'utente di Neural Connection si legge:

"Sostanzialmente, le reti neurali sono semplicemente un nuovo modo di analizzare i dati. Ciò che le rende utili è la loro capacità di apprendere complessi modelli e trends nei dati, una capacità che è unica per le reti neurali. [...] Neural Connection è stato disegnato in modo che l'utente possa gestire immediatamente questa tecnica, senza il bisogno di diventare un esperto di calcolo neurale" (SPSS inc., 1997a, pp. 2-3).

Queste citazioni meritano, a nostro avviso, due commenti che è opportuno sottoporre a chi intendesse avvalersi delle Rna nelle proprie ricerche:

1. le Rna non sono solo uno strumento di analisi dei dati finalizzato alla classificazione e alla previsione;
2. Il programma Neural Connection rende disponibile solo molto parzialmente le potenzialità attualmente esistenti nei settori di ricerca che fanno uso del calcolo neurale.

Per quanto riguarda il primo punto, bisogna dire che la principale funzione delle Rna è quella di simulare un dato fenomeno per permetterne lo studio. Storicamente le Rna sono state proposte per emulare alcune funzioni del cervello umano e del sistema nervoso. Dedicheremo il prossimo paragrafo ad una presentazione della storia del connessionismo che è l'approccio di ricerca nel quale sono state proposte le Rna. Il terzo paragrafo sarà dedicato ad una presentazione delle caratteristiche tecniche delle Rna, riferendoci in particolare ad un tipo denominato rete *feedforward* a più strati.

A proposito del secondo punto, bisogna dire che per poter gestire effettivamente le potenzialità offerte dal calcolo neurale sarebbe preferibile rivolgersi a programmi specialistici dedicati esclusivamente alle Rna. Ancora una volta si deve sottolineare il pericolo insito nella eccessiva semplificazione e facilità di uso che i pacchetti modulari di elaborazione dei dati propongono agli utenti esagerandone le prestazioni (detto per inciso, anche il programma di analisi dei dati SPADN ha reso disponibile una procedura di *clustering* basata su un tipo di Rna). Con questo non si vogliono negare i vantaggi dell'attuale disponibilità di procedure di analisi dei dati tanto diverse tra loro che questi pacchetti integrati offrono; ma la situazione comincia a somigliare a quella degli scaffali di un ipermercato nel quale sono allineati innumerevoli prodotti tra i quali diventa difficile scegliere quello o quelli che realmente servono. Lo stesso linguaggio usato nei documenti di queste case produttrici sembra totalmente assimilato a quello adottato dalle imprese che vendono prodotti o servizi: si sottolineano semplicità, velocità e qualità del risultato, il tutto garantito dal marchio che propone i prodotti.

Questo lavoro si ripromette fondamentalmente due obiettivi: presentare le Rna in maniera soddisfacente; applicare delle Rna a dati di natura sociologica confrontandole con tecniche statistiche tradizionali (come l'analisi discriminante lineare e la regressione multipla lineare), per valutarne gli eventuali vantaggi per le ricerche sociali.

2. Reti Neurali Artificiali: breve storia e concetti introduttivi

Non si può parlare di Rna senza fare riferimento al connessionismo, un approccio che tenta di simulare l'intelligenza biologica su un calcolatore prendendo a modello il cervello in quanto organo fisico. Questo è un primo e fondamentale elemento di distinzione tra il connessionismo da un lato e il cognitivismo e l'intelligenza artificiale dall'altro. In questi ultimi approcci è stata accolta, esplicitamente o implicitamente, la metafora della mente come calcolatore elettronico. Nel connessionismo, al contrario, il cervello è la metafora con cui si studia la mente. Da anni tra gli opposti fautori dei due diversi approcci è in atto un'aspra contesa per la quale si rinvia a Parisi (1989), Cammarata (1990) e Buscema (1994).

A giudizio dello psicologo Domenico Parisi (1989), esperto di Rna, questo insieme di idee, di teorie e di tecniche computazionali, a partire dalla seconda metà degli anni '80, rappresenta una vera e propria rivoluzione scientifica nello studio della mente e del cervello. Quindi a partire dagli ultimi due decenni, il connessionismo "rappresenta per molti aspetti un vero e proprio capovolgimento dei punti di vista e dei modi di procedere che da decenni si erano ormai consolidati con l'intelligenza artificiale e con la scienza cognitiva" (Parisi, 1989, p.9).

I primi tentativi di impostare su dei calcolatori dei sistemi intelligenti che emulassero le attività cerebrali erano stati elaborati in neurocibernetica negli anni trenta. Negli anni '40 McCulloch e Pitts costruirono i primi sistemi intelligenti basati sulla simulazione dell'attività cerebrale. Negli anni '70 le ricerche sulle Rna conobbero una fase di stasi poiché i sistemi realizzati mostravano una scarsa efficienza. Dalla seconda metà degli anni '80, grazie alla disponibilità di sistemi paralleli e di nuovi algoritmi di apprendimento (in particolare la retropropagazione dell'errore), l'interesse verso le reti è cresciuto rapidamente. Nel contempo, i progressi limitati dell'intelligenza artificiale simbolica nella costruzione di sistemi intelligenti di tipo generale mediante le tecniche di manipolazione simbolica proprie di tale approccio (sistemi esperti, linguaggi logici, reti semantiche) e l'interesse tecnologico per architetture di calcolatori più vicine a quello che appare il modo di funzionare del sistema nervoso (parallelo più che sequenziale), hanno alimentato la contesa fra i sostenitori del cognitivismo (o approccio simbolico) e del connessionismo. Secondo Cammarata (1990) l'approccio simbolico è più adatto a processi di intelligenza conscia, come il ragionamento di esperti umani o la dimostrazione di teoremi; ma non sembra riflettere la natura di molti processi di intelligenza inconscia come quelli legati al riconoscimento di immagini o di suoni. Un crescente numero di ricercatori si sta convincendo che l'approccio ideale alla soluzione dei problemi legati alla percezione non è quello simbolico ma quello connessionista. E' comunque riconosciuto che il paradigma connessionista, e in particolare le Rna, non solo possono affrontare nuove classi di problemi, che eludono l'informatica tradizionale e l'intelligenza artificiale, ma nella soluzione di problemi già affrontati conferiscono vantaggi in termini di semplicità e di efficienza. Allo stato attuale, le applicazioni delle Rna sono moltissime: dalle sofisticate tecnologie militari per la guida dei missili ai dispositivi di sicurezza negli aeroporti, ai meccanismi di crittografia, a quelli di riconoscimento di forme e immagini, al controllo della qualità nei processi di automazione industriale, alla cancellazione adattiva del rumore nelle telecomunicazioni, fino ai dispositivi di messa a fuoco automatica delle macchine fotografiche e/o delle telecamere, etc.

Il numero dei problemi affrontabili con le Rna è praticamente illimitato. Secondo Buscema (1994), è utile affrontare un problema con le Rna quando è un problema complesso per il quale altre tecniche di analisi non hanno fornito risultati soddisfacenti e per il quale è ritenuta utile anche una soluzione approssimata anziché assolutamente esatta.

Di seguito elenchiamo sinteticamente i passi più significativi nell'evoluzione dell'approccio connessionista (Cammarata, 1990):

- nel 1943 McCulloch e Pitts propongono la realizzazione di Rna dando un modello formale delle unità (neuroni artificiali) e delle connessioni (sinapsi artificiali). I loro interessi erano rivolti all'individuazione delle funzioni computabili dalle Rna. Il modello fu utile per studiare l'effetto delle modificazioni delle sinapsi sulle prestazioni del sistema nervoso;
- nel 1949 Hebb, assumendo l'apprendimento biologico come fenomeno sinaptico, propone una regola, nota come "legge di Hebb", per la modifica dei pesi sinaptici. La regola si basa su un concetto elementare che modifica i pesi sinaptici aumentando il peso della connessione tra due unità se queste sono per più volte di seguito contemporaneamente attive o contemporaneamente inattive, mentre diminuisce il peso della connessione tra due unità in stati di attivazione alterni, secondo una costante che rappresenta il tasso di apprendimento. Applicando questo tipo di apprendimento si mettono a punto dei dispositivi in grado di svolgere non solo svariate funzioni ma anche di apprendere il compito loro richiesto;
- nel 1957 Roseblatt mette a punto il perceptrone, un dispositivo in grado di riconoscere e classificare correttamente immagini anche mai viste prima, manifestando quindi capacità di capire piuttosto che di imparare a memoria, attraverso un meccanismo di associazioni fra stimoli simili. Il perceptrone mostra inoltre la robustezza e flessibilità tipica dei sistemi

biologici: a differenza di quanto accade nei computer, un guasto, un errore nell'input non compromette l'intera elaborazione ma provoca un lieve peggioramento delle prestazioni;

- nel 1969 Minsky e Papert, approfondendo le possibilità del perceptrone, ne evidenziano i limiti e l'incapacità di risolvere molte classi di problemi. Nel loro libro *Perceptron* (1969) esprimono molte perplessità sulle possibilità di migliorare le Rna. Ciò comporta una fase di sfiducia che si protrasse per tutti gli anni '70. Il tentativo di riprodurre le facoltà logiche dell'uomo divenne un compito affidato all'intelligenza artificiale;
- nel 1982 Hopfield propone l'uso di Rna per la realizzazione di memorie autoassociative, dimostrando anche l'applicabilità nella soluzione di problemi di ottimizzazione. Tali memorie erano in grado di mettere a disposizione le informazioni che contenevano anche a seguito di una richiesta formulata in modo non corretto; potevano quindi correggere eventuali errori nei dati elaborati dal calcolatore sulla base delle informazioni memorizzate. Tale compito era stato già studiato dall'intelligenza artificiale, ma la sua realizzazione con la programmazione dei calcolatori convenzionali era apparsa difficile;
- nel 1986 Rumelhart, Hinton e Williams propongono un potente algoritmo di apprendimento, detto di retropropagazione dell'errore, per le Rna con un'architettura che generalizza il perceptrone, consentendo il superamento di molti suoi limiti. Tale tecnica consiste nel confrontare iterativamente la prestazione fornita dalla rete e quella desiderata in risposta ad una determinata combinazione di stimoli di ingresso, e nel modificare di conseguenza le interconnessioni, fino a ottenere la prestazione voluta. Con la retropropagazione dell'errore si addestra la rete ad eseguire un determinato compito in risposta a certi stimoli. Non si definiscono a priori le connessioni della rete, ma queste vengono a modificarsi progressivamente, in modo automatico, fino a che la rete non è grado di compiere il compito richiesto. La relativa semplicità e praticità del nuovo algoritmo riportò in auge il connessionismo anche a seguito delle prime applicazioni commerciali del calcolo neurale;
- molto rilevanti sono stati anche i contributi di Kohonen, che ha ideato un meccanismo di apprendimento per Rna che non richiede l'esistenza di un obiettivo e che è attivato nelle Rna dette non supervisionate o autopoietiche. Si è giunti così alla costruzione di Rna in grado di acquisire conoscenza in modo autonomo senza dover imparare a riprodurre un qualche tipo di risultato.

In prima approssimazione, possiamo dire che il connessionismo è un approccio che si oppone a quello cognitivista cercando di superare la classica distinzione cartesiana tra mente e cervello. A seguito di questa distinzione le discipline che si occupavano delle capacità mentali e dell'intelligenza, come la psicologia, si erano separate dalle neuroscienze che studiavano il cervello e il sistema nervoso centrale come organi del corpo umano. "Il connessionismo rappresenta la promessa di un superamento di tutti [...] (gli) ostacoli che hanno fino ad oggi impedito alla scienza di andare molto avanti nella conoscenza in questo campo. Il connessionismo offre alla psicologia dei metodi di indagine empirica e dei concetti e dei modelli teorici coerenti e unitari capaci di reggere il confronto, quanto a rigore, con i metodi e i modelli teorici delle ben più consolidate scienze della natura. Nello stesso tempo offre alle neuroscienze la possibilità di integrare il loro apparato di strumenti, oggi ristretto esclusivamente ai metodi e ai concetti delle scienze naturali, con strumenti nuovi, che forse permetteranno a queste scienze di capire meglio che cosa fa il cervello ai livelli più complessivi di integrazione e come quest'organo fisico possa produrre prestazioni e funzioni complicate e misteriose come quelli di percepire, pensare, usare il linguaggio, pianificare, apprendere, interagire socialmente, e così via. [...] In altre parole, il connessionismo è l'ipotesi che la mente e il cervello possano essere studiati con un identico insieme di concetti e di metodi di indagine empirica. Questa è la base del vero e proprio superamento del dualismo cartesiano" (Parisi, 1989, p.11-12).

Il connessionismo è legato a tutta una serie altri strumenti concettuali come le Rna, i sistemi dinamici non lineari, i sistemi dinamici complessi, i sistemi di elaborazione parallela distribuita, le memorie associative, etc. Strumenti che a loro volta si servono della simulazione sui calcolatori. I vantaggi offerti da questo approccio sono molto interessanti anche per le scienze sociali. Innanzitutto se un fenomeno deve essere simulato su un calcolatore, è necessario rendere esplicita e formalizzare tutta la conoscenza di cui si dispone su di esso. Inoltre, una volta che si è in grado di simulare un dato fenomeno, diventa possibile manipolarlo in modi che non sarebbero consentiti con altre tecniche di indagine, per ragioni etiche e per altri vincoli dettati dalle limitate risorse di cui si dispone in una qualsiasi indagine scientifica.

D'altra parte, l'uso del calcolatore ha permesso di studiare con successo fenomeni caratterizzati da alto dinamismo, alto parallelismo e forte complessità, governati da regole di cambiamento che possono essere descritte da equazioni non lineari, praticamente risolvibili solo ricorrendo a un calcolatore elettronico. "Con il calcolatore sta emergendo una 'matematica sperimentale' che ha prodotto i modelli dei sistemi dinamici non lineari, i sistemi caratterizzati da 'caos deterministico', la stessa geometria dei frattali. Questi modelli vengono applicati a fenomeni complessi come la dinamica dei fluidi e quella delle popolazioni, certi fenomeni economici e, appunto le reti neurali" (Parisi, 1989, p.13-14, virgolette nel testo).

Ricapitolando, il connessionismo, costruendo sistemi neurali artificiali basati esclusivamente su regole matematiche, tenta di costruire sistemi intelligenti. Con il termine 'Rna' si intende un insieme di regole di calcolo che simulano un comportamento tipico della struttura cerebrale degli esseri viventi. Questa è la fondamentale differenza del connessionismo rispetto all'approccio simbolico, il cui modello di intelligenza è quindi basato sulla manipolazione di simboli attraverso l'uso di regole che costituiscono il programma da eseguire. Diversamente dai modelli computazionali usati nei sistemi esperti, nelle reti non esiste un programma che specifica le operazioni da eseguire, ma la computazione è definita attraverso le caratteristiche delle unità di elaborazione e delle loro interconnessioni. Una rete apprende e generalizza attraverso l'esperienza che acquisisce piuttosto che attraverso un programma che determina il suo comportamento.

Nell'intelligenza artificiale, invece, la strada seguita nel costruire macchine intelligenti (i cosiddetti sistemi esperti) consiste nel tentativo di trasferire le conoscenze dagli esseri umani alla macchina attraverso un sistema di regole codificate in qualche tipo di linguaggio simbolico. Ad esempio, volendo costruire un sistema esperto capace di effettuare delle diagnosi mediche in relazione ad un insieme di sintomi rilevati su campioni di pazienti, il compito dei ricercatori di intelligenza artificiale consiste nell'interrogare un certo numero di specialisti medici e di codificare esattamente le regole da essi seguite nel loro lavoro in un programma che viene poi immesso su un calcolatore. E' chiara l'enorme difficoltà, e in certi casi l'impossibilità, di un tentativo di trasferire le conoscenze e le esperienze di un certo numero di specialisti umani in un programma di regole che consentano ad una macchina di fornire un'esatta diagnosi in relazione ad un insieme di sintomi. Nei vari tentativi effettuati finora in questo campo i risultati sono stati sempre insoddisfacenti.

L'alternativa che il connessionismo propone consiste nella costruzione di sistemi neurali artificiali capaci di apprendere, e successivamente di generalizzare, sulla base dell'esperienza che viene loro somministrata. Ci sono molti tipi di Rna, distinti per architettura, regole di apprendimento, funzioni di trasferimento del segnale, etc. In questa sede non c'è spazio per presentarli tutti. Ci soffermeremo in particolare su quelle reti, dette "supervisionate", che nella fase di addestramento hanno un obiettivo da raggiungere, al contrario di quelle dette "non supervisionate" (o autopoietiche) che non hanno, nella fase di addestramento, un obiettivo predeterminato da raggiungere.

Le Rna sono costituite da molte unità di elaborazione (dette unità o nodi o neuroni artificiali), di solito ordinate in strati, dal funzionamento molto semplice tra loro interconnesse. Nella rete così costituita si fa passare un segnale (in forma di esempi, anche detti *patterns*) che attiva o inibisce le unità. Esse, con opportune regole matematiche, trasferiscono il segnale ad altre unità fino a produrre un *output* quantitativo. In altre parole, una unità riceve attivazione (o inibizione) dalle unità da cui arrivano connessioni e, a sua volta, manda attivazione (o inibizione) alle unità verso le quali ha delle connessioni.

Si è già detto che in questa sede ci occuperemo prevalentemente di un tipo di reti, dette *feedforward*, che hanno le unità disposte su almeno tre strati e connessioni di tipo unidirezionale tra ogni unità di uno strato e tutte le altre unità dello strato successivo. Anche se in letteratura il termine “connessione” è usato indifferentemente per legami monodirezionali e bidirezionali, in questa sede si preferisce usare il termine “legame” tra unità in quanto il termine “connessione” è intrinsecamente bidirezionale.

L’attivazione o inibizione che arriva ad una certa unità attraverso le altre unità cui è connessa dipende dai pesi che caratterizzano i legami. Se un peso su un legame è alto questo fa passare molta attivazione; un peso basso fa passare poca attivazione. Un peso positivo trasmette attivazione; un peso negativo inibizione.

Ciò che caratterizza le Rna è il loro funzionamento in parallelo: ogni nodo della rete costituisce una unità di elaborazione autonoma che effettua dei calcoli matematici in parallelo con tutte le altre. Nei sistemi seriali, invece, le operazioni vengono effettuate una dopo l’altra, in sequenza. Un esempio di sistema seriale è costituito dagli odierni calcolatori. In questi, il sistema è diviso in due parti fondamentali: un’unità centrale di elaborazione (la CPU) e una memoria passiva di dati. Il sistema funziona in quanto la CPU esegue una sequenza di istruzioni (il programma) sui dati conservati nella memoria. I calcolatori odierni non sono fatti per apprendere conoscenza; l’intelligenza risiede nel programma che è stato scritto (in un apposito linguaggio) e inserito nella macchina da un essere umano. Questo programma non si modifica o migliora per il solo fatto di venire applicato. Emulando il funzionamento in parallelo del cervello, è possibile costruire calcolatori di nuovo tipo. Lo sviluppo di calcolatori a funzionamento parallelo consentirebbe tutta una serie di applicazioni che con gli attuali calcolatori non sono possibili o richiedono un tempo eccessivo. Questo perché nei calcolatori a funzionamento parallelo in ogni frazione di secondo moltissime operazioni avvengono contemporaneamente, e ciò innalza il livello delle prestazioni. Non si è vincolati alla velocità con cui avviene la singola operazione come negli attuali calcolatori. Inoltre, l’uso di calcolatori a funzionamento parallelo comporterebbe una trasformazione dei sistemi di produzione del *software* in quanto la prestazione finale dipende da un grande numero di semplici processi locali che non richiedono un sofisticato programma che controlli tutto dall’alto. Ciò, tra l’altro, consentirebbe una maggiore adattabilità e resistenza ai malfunzionamenti del sistema.

Una Rna riesce ad apprendere un compito, risolvere un problema, quando il propagarsi in parallelo dell’attivazione della rete raggiunge uno stato di equilibrio (e questo matematicamente significa aver trovato il minimo di una funzione), ossia quando l’attivazione arriva sulle unità di uscita (*output*) della rete. Come detto, l’aspetto fondamentale delle Rna è la loro capacità di apprendere, ma su questo punto bisogna intendersi. Infatti, quello che le Rna apprendono e quello che consente loro di eseguire dei compiti o risolvere dei problemi sono dei pesi che stanno sui legami e che regolano quanta attivazione o inibizione si propaga nella rete e come. In altri termini, nelle Rna l’apprendimento, cioè l’acquisizione di conoscenza e di capacità, consiste in un processo di modifica dei pesi sui legami. Le Rna sono quindi intrinsecamente quantitative, apprendono pesi numerici, li trasformano matematicamente e forniscono un risultato matematico.

Nella fase di addestramento di una rete, lo stato iniziale (ossia i pesi iniziali sui legami) sono definiti casualmente, di solito in un intervallo molto piccolo (ad esempio tra $-0,1$ e $+0,1$). Vengono presentati alla rete alcuni esempi ognuno associato, nelle reti supervisionate, ad un *target*. La rete deve imparare, per ogni esempio, a produrre un *output* il più simile possibile al *target*. La differenza tra l'*output* della rete e il suo *target* costituisce l'errore: la rete modifica, attraverso un meccanismo che viene detto retropropagazione dell'errore, i pesi sui legami fino a rendere minima la distanza tra *output* e *target*.

Le unità (o i nodi) emulano le cellule nervose del cervello (neuroni); i legami (o connessioni quando sono bidirezionali) tra le unità emulano i collegamenti sinaptici che esistono tra l'assone di un neurone e i dendriti di un altro neurone. In effetti, fino ad ora le ricerche con le Rna hanno permesso di riprodurre solo alcune, ma importanti, proprietà del cervello umano, che comunque non sono riproducibili in altro modo. "E' evidente che ogni modello coglie alcuni aspetti del pezzo di realtà che vuole modellare e ne lascia fuori altri. Il problema è riprodurre gli aspetti importanti, non riprodurli tutti. I risultati già conseguiti dal connessionismo sembrano dimostrare che il termine rete neurale non è del tutto inappropriato" (Parisi, 1989, p. 20).

Come detto, le Rna, nelle loro proprietà matematiche, fanno parte di una classe più ampia di modelli formulati per lo studio di sistemi complessi, a dinamica non lineare, di tipo caotico, etc. Questi modelli sono stati introdotti nei settori di ricerca più innovativi in diversi ambiti disciplinari. Il loro carattere generale e astratto li rende applicabili a fenomeni molto diversi tra loro, e quindi di interesse potenziale molto ampio, e anche a fenomeni sociali, dal comportamento nei mercati finanziari alle dinamiche demografiche, etc.

Il problema che ci poniamo è se le Rna possono essere applicate utilmente nella ricerca sociale, oltre che come complesso di algoritmi non lineari di elaborazione dei dati, anche come strumento per simulare fenomeni sociali (vedi sul punto i contributi di Negrotti, Donnanno e Sacchi, 1995, Capecchi, 1996, Conte, 1997 e Ricolfi, 1997b). E' difficile assimilare i fenomeni sociali a quelli neurofisiologici; per questo motivo, le analogie dei nodi di una Rna con i neuroni, delle connessioni con le sinapsi, etc., che è possibile nello studio della mente/cervello, non è possibile in questi altri casi. Si tratta tuttavia di valutare se l'astrattezza delle strutture e dei processi postulati nelle Rna intese come modelli di sistemi dinamici complessi non lineari possono consentire una loro applicazione anche allo studio di fenomeni sociali. In questo caso è necessario stabilire l'interpretazione da dare a concetti come unità, connessione, attivazione/inibizione, peso sulle connessioni, regola di apprendimento, stato di equilibrio e così via.

D'altra parte l'uso delle Rna consentirebbe la possibilità di superare in parte alcuni limiti delle analisi condotte con tecniche tradizionali. Ad esempio, l'uso delle Rna non richiede alcuna ipotesi sulla natura delle distribuzioni delle variabili del sistema e delle correlazioni fra loro. Per questa ragione è possibile il trattamento di variabili cardinali e/o categoriali ordinate e non ordinate (opportunamente codificate; cfr. SPSS inc., 1997b). Con tale approccio l'analisi vera e propria del sistema viene lasciata alla rete, che da sola si crea i propri criteri per riprodurre il comportamento e di conseguenza si mette in grado di formulare previsioni sul sistema stesso. A parere di Fabbri e Orsini (1993), questo è al tempo stesso un pregio e un difetto delle Rna: è un pregio perché così il ricercatore non è condizionato da ipotesi aprioristiche nella scelta delle unità della rete; il difetto consiste nel fatto che la rete non può far altro che riprodurre in maniera fenomenologica il comportamento del sistema analizzato, senza contribuire alla conoscenza delle relazioni interne fra le singole parti di cui il sistema si compone. Detto in termini diversi, il contributo interpretativo del modello è nullo. Questo problema, però, è in parte superabile, come vedremo più avanti, in quanto sono stati

messi a punto alcuni dispositivi che consentono di interrogare la rete su quello che è riuscita a riprodurre.

Se l'approccio simulativo delle Rna ai fenomeni sociali si rivelasse possibile ed utile, ciò consentirebbe dei progressi significativi nelle discipline sociali anche perché contribuirebbe alla costituzione di una base omogenea di concetti, modelli e tecniche di simulazione. Se i fenomeni sociali possono essere pensati come sistemi dinamici complessi (in quanto formati da un grande numero di elementi che interagiscono in base a regole di carattere puramente quantitativo e non simbolico, e che mutano nel tempo originando una complessa dinamica collettiva: così ad esempio la borsa, un sistema economico, un'azienda) allora bisogna accettare la possibilità di simularli su un calcolatore con risultati più pregnanti di quelli ottenibili con gli strumenti matematici tradizionali. A nostro avviso, quindi, l'approccio connessionistico applicato alle scienze sociali è promettente e merita, quanto meno, di essere messo alla prova.

3. *Qualche dettaglio tecnico sulle Rna*

Illustreremo alcuni concetti chiave delle Rna, e in particolare delle reti *feedforward* con almeno uno strato nascosto caratterizzate da un metodo di apprendimento detto a retropropagazione dell'errore (d'ora in poi EBP). Questo tipo di reti è stato proposto da un gruppo di ricercatori dell'università di San Diego in California (Rumelhart, McClelland, 1986).

Anziché ricorrere a formalizzazioni matematiche ci varremo di rappresentazioni grafiche.

Un aspetto caratterizzante delle Rna è la capacità di apprendere; l'apprendimento consiste nella ricerca dell'insieme di pesi sui legami appropriato per ogni specifico compito. La rete parte da un stato in cui i pesi sono assegnati a caso; quindi l'*output* risultante sarebbe, al tempo *t-zero*, altrettanto casuale. Attraverso il suo addestramento ha luogo una modifica progressiva dei pesi sui legami della rete fino ad ottenere l'insieme di pesi che produce l'*output* desiderato. Peraltro, anche dopo un lungo addestramento, le Rna non producono dei risultati molto precisi. Questo che sembra un handicap in compiti in cui è richiesta un'alta precisione, diventa interessante in compiti di classificazione e di riconoscimento. Infatti in un compito di classificazione, oggetti simili possono venire inseriti nella medesima classe; di conseguenza anche *patterns* (nel nostro linguaggio casi) affetti da rumore, distorsioni o incompleti possono venire classificati in maniera corretta. Ciò dimostra che le reti possiedono un'alta tolleranza al rumore; questa caratteristica è importante se si pensa che nell'analisi dei dati spesso capita di imbattersi in dati di qualità scarsa. La particolarità più interessante dei modelli neurali è però costituita dalla loro capacità di generalizzare: se in ingresso viene presentato un *pattern* diverso da quelli utilizzati per l'apprendimento, la rete riesce, entro certi limiti, a classificarlo in maniera corretta (sempre che esista una classe per quel *pattern*). Da tempo infatti, le aziende che producono sondaggi usano le Rna per individuare la volontà di voto degli elettori che non si sono espressi. Nella ricerca sociale, sfruttando la capacità di generalizzazione delle Rna, si potrebbero trattare i casi che presentano risposte mancanti senza essere così costretti a non utilizzarli o a sostituire le risposte mancanti con valori tipici (media, moda e mediana) della relativa distribuzione.

Un'altra caratteristica importante è costituita dal fatto che per risolvere analoghi problemi le reti non necessitano di nuovi algoritmi applicativi: i modelli neurali possono infatti adeguarsi flessibilmente a situazioni complesse e mutevoli nel tempo, direttamente nel caso di Rna non supervisionate, con un nuovo addestramento per quelle supervisionate.

Schematicamente una Rna è costituita da: a) un elevato numero di semplici unità di elaborazione (neuroni artificiali); b) un elevato numero di legami tra le unità (sinapsi artificiali); c) uno schema di controllo parallelo e distribuito; d) un algoritmo di apprendimento.

Una Rna *feedforward* è formata da un certo numero di unità interrelate mediante legami che sono, in questo tipo di rete, unidirezionali. Attraverso i legami si trasmette attivazione o inibizione da una unità (o nodo) all'altra. Ogni unità ha un certo numero di legami in arrivo con altre unità e un altro certo numero di legami in partenza verso altre unità (vedi fig. 1).

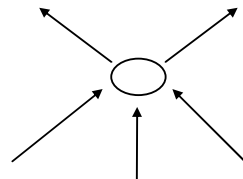


Figura 1: Unità con tre legami in arrivo e due in partenza

Esiste uno strato di unità, dette unità di *input*, che non hanno legami in arrivo ma solo legami in partenza. Lo stato di attivazione di queste unità di *input* è determinato dall'esterno della rete. Ed esiste un secondo strato di unità, dette unità di *output*, che hanno solo legami in arrivo e nessun legame in partenza. In pratica, lo stato di attivazione delle unità *output* viene letto dall'esterno e ci dice come la rete ha reagito all'*input* arrivato dall'esterno.

Lo stato di attivazione di un'unità è pari ad una combinazione matematica di tutte le attivazioni e le inibizioni che giungono a quella unità attraverso i suoi legami in arrivo. Da ciascun legame in arrivo giunge all'unità una certa quantità di attivazione o di inibizione, che viene ponderata con un valore matematico (detto peso sul legame) che caratterizza ogni legame. Il peso può avere segno positivo o negativo, e questo determina il fatto che venga trasmessa attivazione (segno positivo) o inibizione (segno negativo; cfr. fig.2)

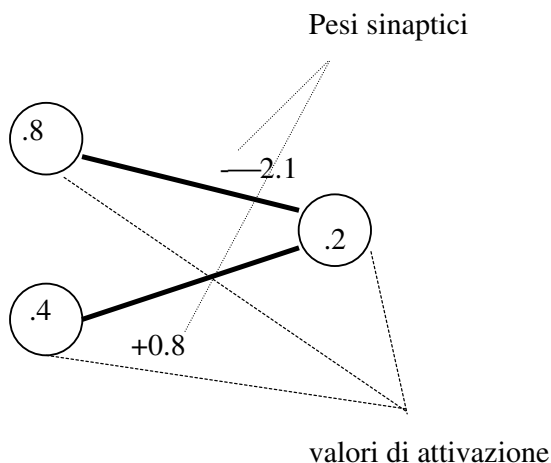


Figura 2: Pesi sui legami

La quantità di attivazione o di inibizione che arriva attraverso un certo legame è determinata da due fattori. Un fattore è il peso sul legame (un legame con peso +0,8 fa arrivare più attivazione di un legame con peso +0,2). Il secondo fattore è lo *stato di*

attivazione dell'unità da cui parte il legame, che può essere più o meno alto. I due fattori vengono moltiplicati fra loro, e il risultato è la quantità di attivazione o di inibizione che arriva ad una certa unità attraverso un certo legame. Nelle Rna *feedforward*, lo stato di attivazione di una certa unità varia da un minimo (0) ad un massimo (1). Ad un momento dato a una certa unità arrivano un certo numero di attivazioni e di inibizioni. La prima cosa che l'unità deve fare è sintetizzare tutte queste attivazioni e inibizioni in un valore unico, che viene detto *input* netto per quella unità. L'*input* netto è normalmente la somma algebrica di tutte le attivazioni e di tutte le inibizioni che arrivano all'unità.

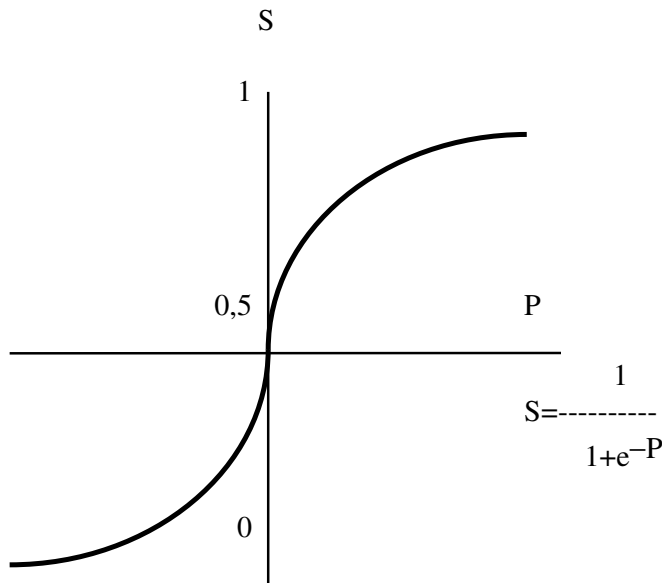


Figura 3: La funzione di trasferimento sigmoide

L'*input* netto viene poi trasformato attraverso un algoritmo matematico nello stato di attivazione dell'unità. L'algoritmo che trasforma l'*input* netto in stato di attivazione è nel nostro caso la cosiddetta 'funzione logistica', o sigmoide, con valori continui e saturazione. Possono anche essere usati altri algoritmi (detti funzioni di trasferimento) come la funzione identità o lineare senza saturazione, la funzione lineare con saturazione, la funzione a gradino con valori binari o bipolari, e altre ancora.

Nella funzione sigmoide lo stato di attivazione varia tra un minimo ed un massimo (0 e 1). Quando l'*input* netto è 0, lo stato di attivazione è 0,5. L'algoritmo è sensibile a piccole variazioni dell'*input* netto, che producono forti scostamenti verso il basso o verso l'alto dello stato di attivazione nella parte centrale del campo di variazione (vedi figura 3).

Determinato lo stato di attivazione attraverso l'algoritmo che segue la funzione sigmoide, questo determina come l'unità data influenza le altre unità con cui la prima è collegata da legami da essa in partenza.

L'architettura delle Rna può essere di diversi tipi. In figura 4 se ne presentano alcuni esempi.

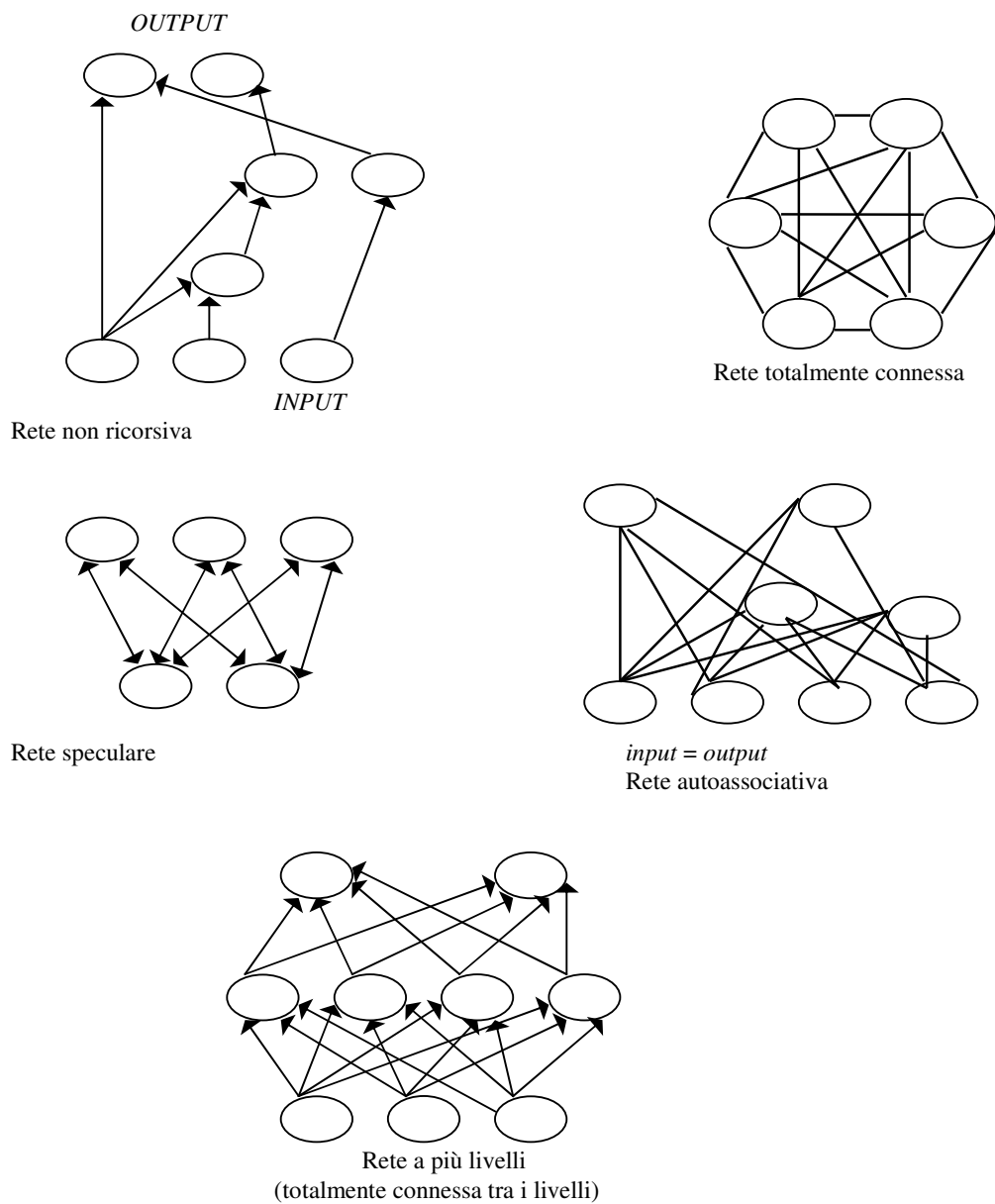


Figura 4: Alcune architetture di Rna

Nelle Rna *feedforward* le unità sono raggruppate in *strati* e le unità di uno stesso strato non sono legate tra loro; possono essere legate solo con unità di altri strati.

Quando la rete ha solo uno strato di ingresso (*input*) e uno strato di uscita (*output*), si chiama perceptrone. Ciascuna unità di ingresso è legata con ognuna delle unità di uscita con un legame la cui intensità è data dal peso; non esistono collegamenti orizzontali tra unità di *output*; la propagazione dei segnali è unidirezionale dall'*input* verso l'*output* (reti *feedforward*, vedi fig.5).

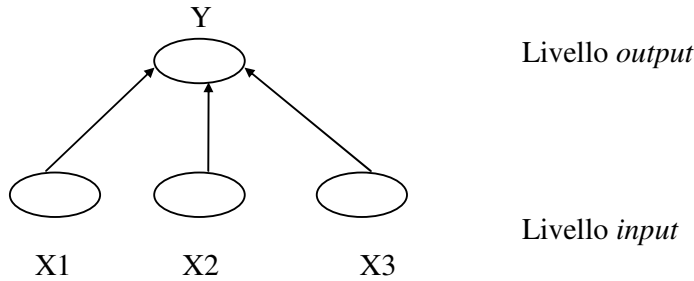


Figura 5: Il perceptrone (rete *feedforward* a due strati *input* → *output*)

Ogni unità di un perceptrone ha un insieme di *input*, ognuno con un peso che rappresenta la forza del legame sinaptico del neurone.

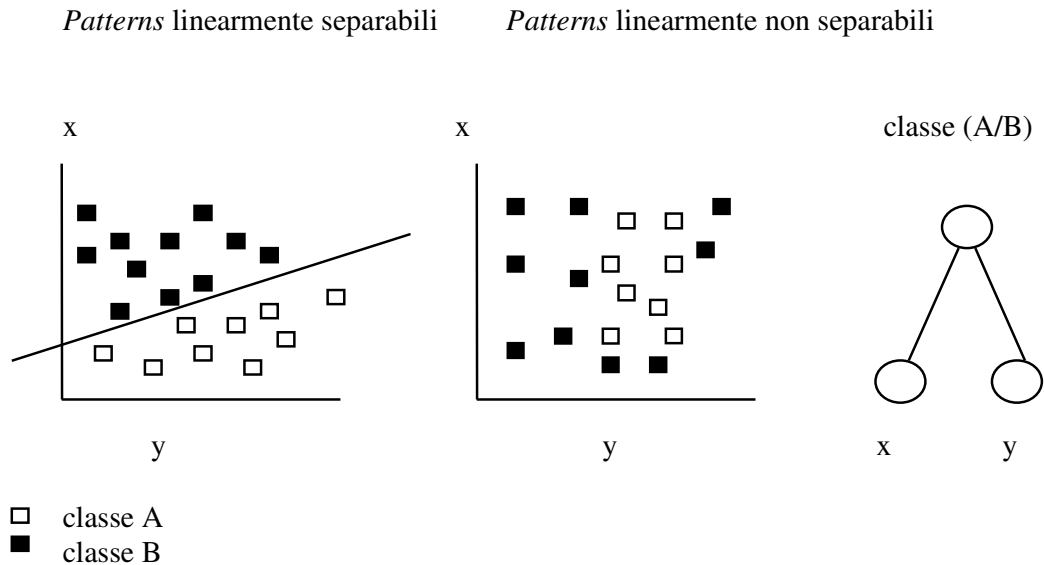


Figura 6: La figura mostra il caso di un perceptrone con due unità di ingresso e una di uscita. Le due unità di ingresso codificano le coordinate dei *patterns* rispetto al piano bidimensionale, l'unità di uscita codifica il tipo di *pattern* in due classi A e B. Il compito del perceptrone è quello di classificare i *patterns* di ingresso in due classi distinte. Due insiemi di *patterns* vengono presentati. Il primo insieme è distribuito in modo tale che i *patterns* delle due classi, distribuiti in uno spazio bidimensionale, possano essere separati da una linea retta (il perceptrone è in grado quindi di apprendere a discriminare le due classi). Nel secondo insieme invece i *patterns* delle due classi non possono essere separati da una linea (di conseguenza il perceptrone non può essere in grado di discriminarli correttamente).

L'*input* globale del neurone è un vettore n-dimensionale (x_1, x_i, x_n) e con pesi associati (p_1, p_i, p_n). Per ottenere l'*output* del perceptrone ogni elemento del vettore di *input* viene moltiplicato per il suo peso, e tutti i valori così ottenuti vengono sommati. L'unità dà come *output* 1 se la somma è maggiore di un certo valore di soglia, altrimenti dà 0. Il grosso limite del perceptrone è costituito dalla sua incapacità di eseguire compiti di classificazione per problemi non linearmente separabili (vedi fig.6).

In sostanza un perceptrone non fa altro che imparare una serie di associazioni dirette tra coppie di *patterns* di attivazione. La rete associa al *pattern* di *input* il *pattern* di *output* aggiustando progressivamente i pesi sui legami che collegano direttamente le unità di *input* alle unità di *output* in modo tale che la rete immagazzini non una singola associazione tra un *pattern* di ingresso e un *pattern* di uscita, ma tante associazioni quanti sono gli esempi (*patterns*) che deve apprendere. Nel far questo il perceptrone non si costruisce nessuna *rappresentazione interna* dei diversi esempi che ha appreso e questo impedisce che il perceptrone abbia la possibilità di evidenziare le somiglianze e le differenze tra i diversi esempi. Proprio perché il perceptrone non può costruirsi una rappresentazione interna autonoma, non è in grado di fare inferenze su proprietà nuove dei *pattern*, cioè su proprietà di cui non ha fatto diretta esperienza (che non le sono state direttamente insegnate).

Quando la separazione lineare è impossibile il perceptrone non riesce a risolvere problemi anche apparentemente semplici. Questa limitazione può essere superata aggiungendo nella rete un ulteriore strato di unità poste tra quelle di *input* e quelle di *output*. Questo strato intermedio viene chiamato nascosto proprio perché è interno alla rete e non ha collegamenti con l'esterno del sistema, a differenza dello strato di *input* che riceve informazioni dall'esterno e dello strato di *output* che trasmette informazione all'esterno (fig. 7). Nel programma Neural Connection questo tipo di reti viene chiamato MLP (*Multi Layer Perceptron*; cfr. SPSS inc., 1997b).

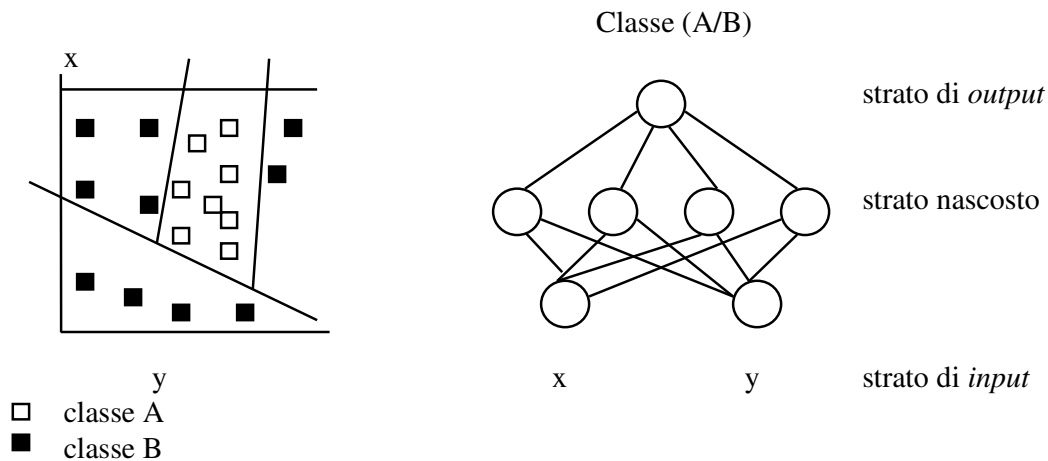


Figura 7: I *patterns* di ingresso delle due classi, distribuiti in un piano bidimensionale, non sono separabili linearmente. Tuttavia, possono essere separati tracciando un certo numero di rette (in questo caso quattro). La figura mostra una possibile soluzione al problema e la corrispondente architettura neurale multistrato, che contiene un numero sufficiente di neuroni interni per la risoluzione del compito.

Secondo il teorema di Kolmogorov (Cammarata, 1990) una rete multistrato dotata di un sufficiente numero di unità nascoste è in grado di apprendere una qualsiasi funzione. Si può quindi ricorrere a reti in cui allo strato di *input* e a quello di *output* vengono aggiunti uno o più strati intermedi (fig. 8).

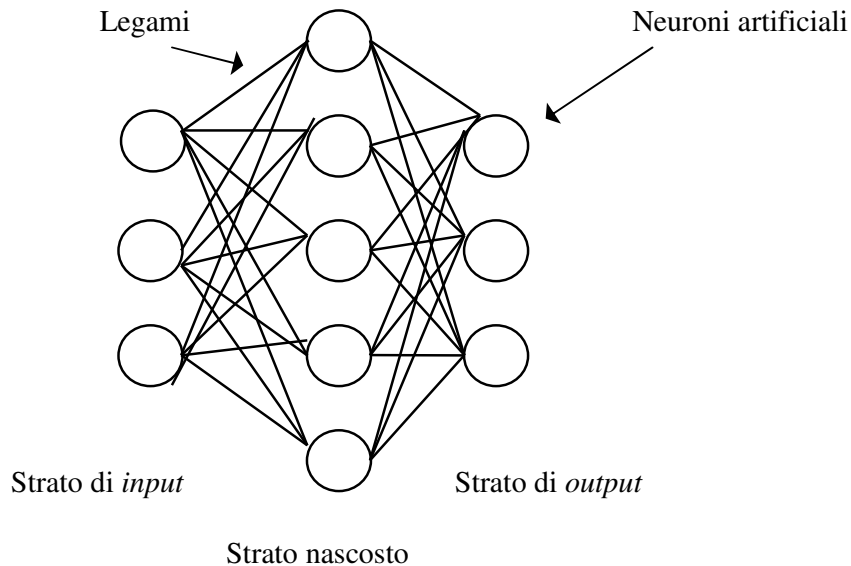


Figura 8: Schema generico di una Rna *feedforward* con uno strato nascosto

Una Rna multistrato è quindi in grado di riconoscere se un dato *pattern* appartiene o meno ad una data classe A separandola dalla classe non A e di apprendere a riconoscere le classi partendo da una struttura di legami modificabile progressivamente nella fase di addestramento. In questa si minimizza l'errore cambiando i pesi dei legami con vari criteri che garantiscono, anche nel caso di non separabilità lineare, la convergenza del processo iterativo verso la soluzione corretta. Anche in questo caso il modo con cui una rete risponde a un *pattern* di attivazione esterna dipende interamente dai pesi che stanno sui legami tra le unità. Quello che si vuole da una rete è che produca un certo *pattern* di attivazione sulle unità di *output*. Ma il peso di uno specifico legame entra nel determinare il valore di attivazione di una determinata unità *output* combinandosi con i pesi degli altri legami e con lo stato di attivazione delle altre unità nascoste. Gli stati di attivazione delle unità nascoste, a loro volta, dipendono da un gran numero di legami provenienti dal basso, e così via. Quindi diventa pressoché impossibile stabilire quale deve essere il peso di ciascun legame, tanto più che se una unità di *output* deve essere attivata o meno in una particolare occasione dipende dal *pattern* complessivo di attivazione che ci si aspetta da tutte le unità di *output* in quella occasione.

La caratteristica comune a molte Rna è che queste reti inizialmente hanno dei pesi scelti a caso sui loro legami; esistono criteri con cui le reti modificano automaticamente questi pesi fino ad assegnare loro quei valori che consentono di rispondere nel modo desiderato a una certa stimolazione esterna. All'inizio la rete darà risposte casuali agli stimoli esterni. Tuttavia, esposta ad esperienze ripetute, la rete modificherà progressivamente i suoi pesi in modo che essi alla fine produrranno la prestazione desiderata.

Sono stati definiti diversi metodi di apprendimento per le Rna. Quello che noi considereremo è un apprendimento supervisionato dall'esterno, cioè un apprendimento in cui esiste un preciso obiettivo (*target*) esterno associato ad ogni *pattern* di *input* che di volta in volta impone alla rete quale è la prestazione desiderata. La rete modifica di conseguenza i pesi sui legami finché, dopo un certo numero di volte, riesce a riprodurre approssimativamente l'*output* corretto per ogni *input*.

In altri metodi di apprendimento (detti non supervisionati) la rete scopre regolarità, mappe o tipologie negli esempi senza riferirsi a un *target*. Questo tipo di reti si chiamano reti di Kohonen, dal nome dello studioso finlandese che le ha proposte (cfr. SPSS inc., 1997b).

Come detto, in questa sede presentiamo il criterio di apprendimento supervisionato EBP (sigla di *error back-propagation*). L'algoritmo EBP consente alla rete di confrontare, per ciascuna unità di *output*, il valore ottenuto con il valore desiderato e di usare la differenza per modificare i pesi sui legami nella giusta direzione, in modo che dopo un certo numero di cicli di apprendimento i pesi sui legami determinino sulle unità di *output* i valori di attivazione desiderati. L'algoritmo richiede un'architettura di rete multistrato: uno strato di *input*, uno di *output* ed uno o più strati nascosti (intermedi). Ogni unità di uno strato è legata alle unità degli strati precedenti; non esistono legami orizzontali tra unità dello stesso strato ed il segnale si propaga unidirezionalmente dall'*input* verso l'*output* attraverso la gerarchia degli strati intermedi (reti *feedforward*). Il procedimento prevede due fasi: nella prima gli esempi di *input* vengono trasmessi verso l'*output*, il valore di attivazione degli *output* viene valutato e confrontato con i valori attesi; nella seconda fase l'errore calcolato viene retropropagato dall'*output* verso gli strati intermedi e da questi allo strato di *input* (fig.9).

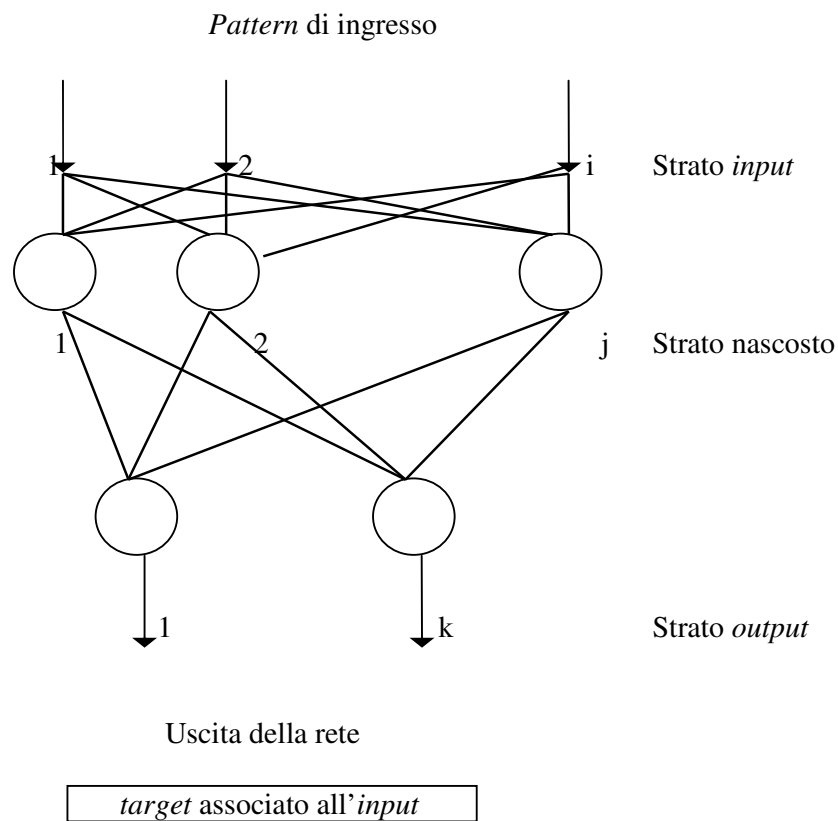


Figura 9: Rete neurale di tipo *back-propagation* con uno strato intermedio

Gli errori delle unità di *output* permettono di stimare gli errori delle unità intermedie e aggiornare i pesi dei loro legami. Quindi si stimano gli errori delle unità di *input* e si aggiornano i pesi dei loro legami. L'algoritmo viene eseguito per tutti i valori dell'insieme di addestramento (*training set*) fino ad ottenere i valori corretti per i pesi di tutti i legami. Dopo una prima presentazione degli esempi, per ognuno dei quali i pesi vengono aggiornati, si può procedere ad altri cicli di presentazione sino a quando l'errore quadratico medio su tutto il

training set non scende al di sotto di una certa soglia, o quando a nuove iterazioni non corrisponde una diminuzione dell'errore e quindi la rete ha raggiunto uno stato stabile, cioè un minimo della funzione di errore. Per un certo numero di volte (che può variare da poche centinaia a diverse migliaia: ogni ciclo compiuto sull'intera serie si chiama *epoca*) elabora questa serie di esempi dell'insieme di addestramento.

Una descrizione matematica del funzionamento dell'algoritmo EBP è in Cammarata (1990). L'algoritmo EBP può essere applicato a una rete con qualunque numero di strati; il numero di unità di ogni strato può variare da strato a strato. Il numero delle unità di *input* e delle unità di *output* è determinato dal problema che si vuole risolvere, mentre non esistono dei criteri per determinare il numero ottimale degli strati intermedi e quello delle unità che ne fanno parte. In genere non si introducono più di uno o due strati intermedi, e il numero delle loro unità è inferiore a quelle dell'*input*. Il vantaggio dell'algoritmo EBP rispetto alle precedenti tecniche di modifica dei pesi è la sua capacità di computare un errore non solo per lo strato delle unità di *output* — per le quali la cosa è molto facile dato che esse ricevono il *target* dall'esterno — ma anche per lo strato di unità nascoste. In questo modo nell'apprendimento vengono modificati non solo i pesi sui legami tra unità nascoste e unità di *output* ma anche quelli dei legami tra unità di *input* e unità nascoste. In una Rna del tipo descritto l'associazione tra *input* e *output* è mediata dalle unità nascoste e dai legami, ciascuno con il suo peso, che uniscono le unità di *input* con quelle nascoste e queste con le unità di *output*. Più esattamente possiamo dire che una Rna si fa una rappresentazione interna dell'*input*, e che la sua risposta all'*input* dipende da questa rappresentazione interna. Questo aspetto delle reti richiama il concetto di variabile latente proprio molte tecniche statistiche tradizionali. La rappresentazione interna di un *input* non è, ovviamente, di tipo simbolico; essa non è altro che l'insieme di valori di attivazione che risultano sulle unità nascoste quando la rete riceve un *input*. Sono questi valori di attivazione che determinano quali saranno i valori di attivazione delle unità di *output*, sulla base dei pesi sui legami tra unità nascoste e unità di *output*. Ed è sulla base di questa rappresentazione interna prodotta durante l'apprendimento che la rete riconosce somiglianze e differenze ed è in grado di inferire e generalizzare.

L'algoritmo EBP presenta diversi punti deboli, tra i quali la lentezza del processo di apprendimento, che richiede spesso un gran numero di "epoche" prima di ridurre in modo accettabile il livello dell'errore globale. Può essere comunque accelerato aggiungendo alla formula di aggiustamento dei pesi un termine che tiene conto dell'aggiornamento nell'epoca precedente: questo ulteriore termine è controllato da un parametro compreso tra zero e uno detto *momento*. Un altro inconveniente sta nella possibilità che l'algoritmo non converga. Infatti, maggiore è il valore del tasso di apprendimento, maggiore sarà la rapidità di apprendimento della rete. Questo però comporta la possibilità di oscillazioni della funzione di errore attorno ad un valore minimo. D'altra parte un valore troppo piccolo del tasso di apprendimento può portare a tempi di addestramento troppo lunghi quindi il valore del parametro viene stabilito spesso per tentativi. L'EBP garantisce (di fatto, anche se ancora non ce n'è una dimostrazione matematica) che la convergenza verso il minimo globale avvenga per una grande varietà di compiti, in particolare evitando che la rete cada in un *minimo locale*, cioè in un assetto di pesi da cui non riesce a spostarsi ma che non corrisponde all'errore minimo globale che sta cercando di raggiungere.

L'apprendimento termina quando l'errore complessivo è sufficientemente basso e comunque non accenna a ridursi ulteriormente aumentando il numero delle epoche. A questo punto la rete ha appreso, cioè è in grado di fornire l'*output* (approssimativamente) corretto per ciascun *input*. Quello che è più importante è che la rete dimostra di possedere una capacità di andare al di là di quello che gli è stato esplicitamente insegnato. Questa capacità di generalizzazione, di inferenza, di andare al di là degli esempi appresi, si manifesta in vari

modi. Se la rete ha appreso a dare una certa risposta ad un esempio, darà questa risposta anche ad una versione deteriorata, parziale, oscurata dal rumore, di tale esempio. Se un *pattern* è stato classificato come appartenente ad una certa classe, anche *patterns* simili mai visti prima verranno classificati come appartenenti a quella classe. Se la risposta che è stata insegnata alla rete per un certo *pattern* contiene talune parti non specificate, la rete sarà in grado di indovinare (inferire) quali sono le corrette parti mancanti.

Ci sono altri fattori, oltre la EBP, che possono entrare in gioco nell'apprendimento. Ad esempio, si può variare il *tasso di apprendimento* (*learning rate*), cioè quanto deve essere grande il cambiamento del peso di un legame dato un certo errore. In genere si preferisce fare cambiamenti piccoli per avere un apprendimento graduale e senza sbalzi, i quali possono aumentare invece che diminuire l'errore complessivo. Un altro fattore che si può variare è il *momento*, cioè se e quanto il cambiamento che introduco ora deve essere influenzato dai cambiamenti introdotti sullo stesso peso le volte precedenti. C'è poi il *bias*, cioè un valore di attivazione che ciascuna unità tende a prendere indipendentemente dalle attivazioni e dalle inibizioni che le arrivano dalle altre unità. Il *bias* è diverso da unità ad unità e viene appreso nel corso dell'apprendimento in quanto è costituito da attivazione o inibizione che giunge all'unità, attraverso un legame con peso apprendibile, da una ipotetica unità speciale che ha sempre attivazione 1. Questi meccanismi aggiuntivi sono indicativi della flessibilità delle reti multistrato e dell'apprendimento EBP.

Però, tutta questa variabilità di fattori, che deriva in primo luogo dal fatto che ogni rete riceve all'inizio una sua specifica assegnazione casuale di pesi, comporta che tutto il decorso dell'apprendimento e il suo risultato finale varieranno da rete a rete. Quindi, se si ripete lo stesso esperimento su reti diverse, cioè aventi assegnazione iniziale di pesi differente e/o modifiche nei parametri sopra esposti, non si potranno avere risultati identici ma soltanto simili. Altre differenze possono derivare dalla durata dell'addestramento (in numero di epoche) e dal modo con cui la rete vede i diversi *patterns*. Anche variando la durata dell'addestramento e l'ordine di presentazione degli esempi, si ottengono risultati leggermente diversi, anche se l'apprendimento avviene lo stesso.

Per quanto riguarda la durata della fase di apprendimento, bisogna tenere presente un ulteriore problema: se una Rna subisce un lungo apprendimento c'è il rischio farla apprendere "troppo" (*overtraining*) pregiudicando così la sua capacità di generalizzazione. Infatti se una rete apprende troppo bene gli esempi usati nel corso dell'addestramento, riuscirà meno bene a classificare altri esempi (diversi da quelli usati nell'addestramento) nella fase di *test* (con ciò si intende la fase in cui la rete ha già appreso i pesi nella fase di addestramento e adesso risponde autonomamente a nuovi esempi che le vengono sottoposti senza più la necessità che ad ognuno sia associato un target). E' quindi più importante che la Rna riesca ad apprendere bene dei prototipi sottostanti gli esempi, piuttosto che essere in grado di rispondere correttamente ad ogni *input* nella fase di addestramento. La conclusione da trarre è che il concetto di prototipo è centrale per le Rna come base per la classificazione. Questo sposta l'accento dalle classi definite in termini di proprietà come si fa normalmente, alle classi definite in termini di prototipo. La capacità di estrapolare, di rispondere sensatamente al nuovo, è una delle più importanti proprietà delle Rna, e uno dei loro principali vantaggi rispetto ai sistemi di analisi tradizionali. Ogni rete risponde in modo sensato a *patterns* nuovi, cioè che non facevano parte dei *patterns* con cui è stata addestrata. Tuttavia, la risposta, in questo caso, è in genere meno buona di quella data per i *patterns* su cui è stata addestrata: la rete è più incerta; se deve classificare il *pattern* nuovo nella classe A, dà un valore di attivazione di 0,8 o di 0,7, invece che di 0,9. Invece, quello che succede con i prototipi è che, se si presenta alla rete il *pattern* prototipo, che pure non ha mai visto prima, la risposta della rete può essere ancora migliore di quella data ai *patterns* su cui si è esercitata tante volte.

In sintesi, l'algoritmo di apprendimento può essere interpretato come la discesa lungo una funzione qualsiasi da un suo punto generico, le cui coordinate sono i pesi iniziali assegnati a caso e l'errore iniziale, al suo unico punto di minimo. Il tasso di apprendimento può essere interpretato come il passo di discesa. A questo punto rimane da considerare il problema dei *minimi locali* (un minimo locale è dato da una certa configurazione dei pesi sinaptici che non corrisponde al miglior punto in assoluto dello spazio dei pesi, in termini di errore globale, ma al miglior punto della zona locale dello spazio intorno al punto stesso, vedi fig.10). Quello che la rete cerca è un valore minimo dell'errore globale, cioè l'assetto di pesi sui legami che dia il minimo di errore per tutti i *patterns* di *input*. Invece, un minimo locale è un assetto di pesi che mantiene l'errore ancora piuttosto alto e tuttavia la rete non riesce a sfuggire da tale assetto, cosa che comporterebbe prima un aumento dell'errore e poi una sua discesa verso un errore più basso. Nelle funzioni non lineari non c'è un unico punto di minimo assoluto, ma si possono trovare diversi minimi locali che rappresenterebbero delle soluzioni subottimali per la rete. Si può dimostrare matematicamente che non si hanno minimi relativi quando il minimo assoluto corrisponde al valore ideale $E = 0$. Non esiste invece una dimostrazione analitica di convergenza verso il minimo globale per Rna con unità di *output* che usano funzioni di attivazione non lineari (come ad esempio la funzioni sigmoide).

Funzione di errore

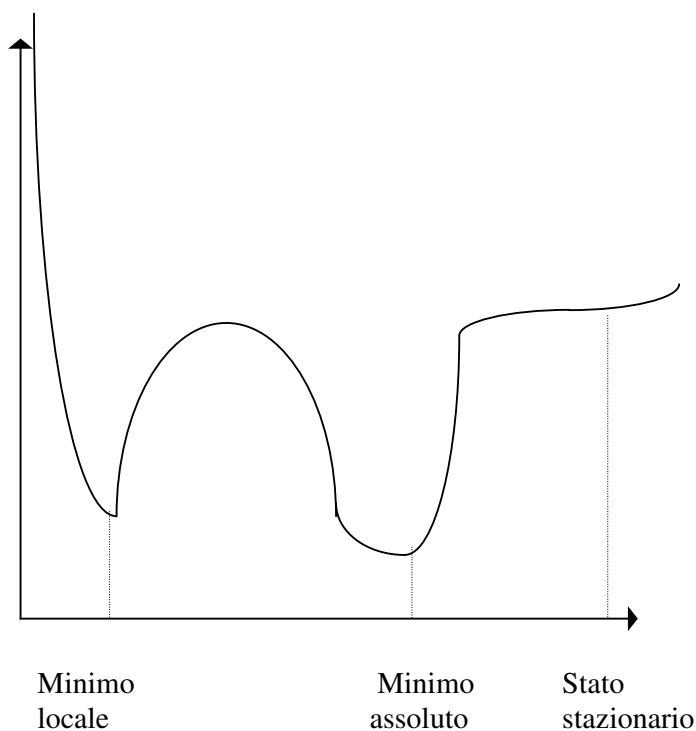


Figura 10: Andamento tipico della funzione di errore, sono evidenziati i punti in cui può convergere l'algoritmo

In pratica però l'intrappolamento in minimi locali si verifica molto raramente e in questi casi basta riniziare la fase di addestramento con un diverso insieme di *pattern* di esempi o diverse condizioni iniziali.

A conclusione di questo rapido *excursus* sulle Rna *feedforward* possiamo ricapitolarne gli aspetti più importanti evidenziandone i punti di forza e quelli critici.

Le Rna sono capaci di apprendere come dimostra il fenomeno della generalizzazione, cioè consentono di risolvere problemi tramite l'associazione della soluzione cercata ai dati. In realtà i metodi di apprendimento delle reti sono applicazioni di metodi statistici noti (approssimazione stocastica) a una nuova classe di modelli di regressione non lineare, in tal senso si può interpretare come una regressione non lineare applicata ad una funzione Rna la determinazione dei pesi della rete. Il vantaggio è quello di disporre di una funzione estremamente flessibile, evitando le componenti soggettive dell'errore di specificazione, in quanto sono i parametri a determinare implicitamente quale sia la funzione latente che una Rna approssima. Se la forma analitica della funzione sottostante al problema oggetto di studio è nota, o è assimilabile ad una forma funzionale nota, il problema della stima dei parametri si riconduce al caso dei minimi quadrati non lineari e il ricorso alle Rna non è giustificato; lo diviene quando non si è in grado di formulare congetture attendibili su tale forma. In questo caso l'uso delle Rna è più agevole e produttivo nei confronti di altre procedure complesse e restrittive nei prerequisiti. L'uso delle Rna è quindi efficace come metodo per l'identificazione di relazioni non lineari nascoste. Correlata alla capacità di apprendere è quella di prevedere. Le Rna offrono delle buone prestazioni sia nella previsione univariata, quando cioè si vuole prevedere il comportamento di una variabile di un sistema che si evolve nel tempo sulla base dell'andamento passato della variabile stessa, sia nell'analisi multivariata, quando invece si cerchi di prevedere l'andamento di una variabile del sistema in evoluzione osservando il comportamento passato di più variabili di esso. Diversi studi hanno posto in evidenza come le Rna consentono di effettuare delle buone approssimazioni ed estrapolazioni. Dato che un problema di previsione può essere riportato ad un problema di approssimazione ed estrapolazione è possibile utilizzare le reti per approssimare le regolarità presenti nelle variazioni nel tempo della variabile che si vuole predire. Le Rna si adattano flessibilmente a situazioni complesse che cambiano nel tempo, in modo diretto se l'apprendimento è non supervisionato, tramite riaddestramento se l'apprendimento è supervisionato. Inoltre sono adatte al trattamento di dati rumorosi, incompleti o affetti da errori di rilevazione. In virtù di questa capacità di adattamento ai dati, le Rna sono molto robuste, ossia hanno un'alta tolleranza ai guasti e disfunzioni. Altra importante caratteristica è la velocità computazionale che deriva dal loro parallelismo e l'associazione *input / output* è molto rapida in quanto i calcoli da eseguire sono somme pesate e decisioni di soglia; costituiscono quindi una valida alternativa ai metodi tradizionali per l'esecuzione di calcoli complessi.

I punti critici delle Rna sono, innanzi tutto, l'apprendimento lungo e scarsamente incrementale; oltre a richiedere un gran numero di epoche prima di conseguire un errore sufficientemente piccolo, l'apprendimento deve essere ripetuto quando la situazione rappresentata dagli esempi subisce delle modifiche sostanziali a meno che l'apprendimento non sia continuo o non supervisionato. In caso di riaddestramento inoltre è difficile conservare i pesi appresi che rimangono validi e occorre spesso ripartire da zero.

Come è ovvio anche per le Rna, come in qualsiasi caso, è necessaria una ricca e rappresentativa (rispetto al problema oggetto di studio) banca dati affinché le fasi di apprendimento e di generalizzazione siano effettivamente controllabili.

Altri problemi possono derivare dalla scarsa precisione dei risultati forniti dalle Rna e dalla loro incerta affidabilità: le prestazioni passate di una rete non garantiscono in modo assoluto quelle future. Esiste il pericolo che la generalizzazione non sia completa e che quindi gran parte degli *inputs* non richiami *outputs* corretti. Inoltre, per applicare una Rna ad un dato problema non esistono criteri rigorosi per progettare la rete più adatta, ma bisogna procedere per tentativi ed errori avendo, come si è detto, numerosi gradi di libertà nella scelta di ogni parametro. Inoltre, ogni Rna ha una sua individualità. Se si ripete lo stesso esperimento su

un'altra rete non si ottengono gli stessi identici risultati, anche se comunque nella maggior parte dei casi questi convergono verso uno stesso livello. Quest'ultima rappresenta un'altra interessante proprietà delle Rna; esse sono in grado di fornire risultati simili, in termini di prestazioni, con una varietà di assetti interna dei pesi. Evidentemente quello che è importante non è il valore di un certo peso, ma l'insieme complessivo dei pesi su tutti i legami.

Infine, la critica più frequentemente sollevata contro l'utilità delle Rna è che, anche quando riescono bene nel compito assegnato, non consentono di spiegare il loro funzionamento su un piano sostantivo (nel caso di ricerche sociologiche potremmo dire sul piano delle relazioni tra le variabili). Da un modello scientifico di qualcosa, ci aspettiamo non solo che riesca a predire o a riprodurre quel qualcosa, ma ci aspettiamo anche che sia trasparente, ovvero che ci faccia capire in che modo quel qualcosa funziona, quali sono i meccanismi, i processi, i principi che stanno dietro di esso. Le Rna, secondo questa critica, rischiano di ottenere il primo obiettivo, ma non nel secondo. Una Rna che è riuscita ad imparare un certo compito ed è anche capace di estendere le sue prestazioni a situazioni nuove, mostrando così di aver incorporato i meccanismi e i principi sottostanti a quel compito, può tuttavia essere poco trasparente riguardo a questi meccanismi e a questi principi, non facendoli emergere con chiarezza e non riuscendo così nell'obiettivo di *spiegare* e di *far capire* i fenomeni in questione. La loro natura strettamente quantitativa, l'intrico dei legami, i pesi su questi legami, gli effetti di un fenomeno locale di attivazione sul resto della rete, sono tutti fattori che rendono opaco il comportamento delle reti come strumenti di spiegazione e di comprensione.

A seguito di queste critiche sono stati prodotti diversi dispositivi per analizzare come si organizzano e si strutturano le Rna e la ricerca in questo campo è uno dei settori di maggiore attività nel connessionismo. Uno di questi tentativi è l'applicazione alle reti della *cluster analysis* di tipo gerarchico. Come si ricorderà, nelle reti con unità nascoste l'attivazione delle unità di *input* da parte di un *pattern* esterno determina gli stati di attivazione nelle unità nascoste. Questi costituiscono la rappresentazione interna che la rete si fa di quel particolare *pattern*. Sottoponendo tutti i vettori di valori numerici che indicano gli stati di attivazione delle unità nascoste per tutti i *patterns* ad una *cluster analysis* è possibile valutare il grado di somiglianza di questi vettori confrontandoli l'uno con l'altro. Il risultato è una serie di raggruppamenti di vettori simili tra di loro, raggruppamenti che si organizzano in una struttura gerarchica. In questo modo si può cercare di interpretare quale sia la gerarchia delle rappresentazioni interne della rete. E' quindi l'organizzazione interna rivelata dalla *cluster analysis* che può spiegare la capacità di fare inferenze della rete. Le Rna sono strutture intrinsecamente casuali e probabilistiche dove l'informazione emerge da un base di rumore di fondo e all'inizio non c'è una distinzione netta tra informazione e rumore. L'informazione emerge gradualmente dal rumore con l'apprendimento. I *patterns* hanno gradi continui di somiglianza tra di loro, la divisione tra le classi è più forte delle divisioni all'interno delle classi, ma non è qualitativamente differente. Un *pattern* può essere a mezza strada tra due classi.

La *cluster analysis* delle rappresentazioni interne di un rete è solo una delle tecniche di cui è necessario disporre per capire come si struttura una rete nel corso dell'apprendimento e come questa strutturazione spiega le prestazioni della rete al termine dell'apprendimento. Un'altra possibilità è la tecnica del *lesionamento* della rete. Una rete può essere lesionata o danneggiata in vari modi, e l'effetto della lesione può essere studiato. La tecnica del lesionamento può essere usata come tecnica sperimentale di manipolazione di una rete per controllare ipotesi sul ruolo di vari elementi e aspetti di una rete nel produrre una certa prestazione. Ad esempio, si può introdurre del rumore su tutti i pesi della rete, o su un sottoinsieme di tali pesi, aumentando o diminuendo a caso i valori quantitativi di tali pesi. Si

possono tagliare un certo numero di legami cioè portare a zero i loro pesi. Oppure ancora, si possono eliminare un certo numero di unità, portando a zero i pesi su tutti i legami in uscita da queste unità. Una volta formulate delle ipotesi sul ruolo di certe unità o di certi legami nel determinare la prestazione della rete, si possono lesionare in uno di questi modi le unità o i legami interessate e controllare le conseguenze di queste manipolazioni sulla prestazione della rete.

Basati su questa tecnica, sul programma Neural Connection sono presenti un simulatore e un modulo *What if* che consentono di dialogare con la rete ponendole delle domande e ricevendo dalla rete delle risposte di cui ci occuperemo nella prossima sezione (Spss inc., 1997b).

4. Esempi di applicazione delle Rna e confronto con tecniche statistiche tradizionali

Il primo esempio che presentiamo riguarda un problema di classificazione non separabile linearmente. Si tratta di riconoscere la classificazione dei collegi elettorali in otto classi prodotta nell'ambito delle ricerche condotte dall'Osservatorio di Sociologia elettorale dell'università di Roma (cfr. Di Franco, 1995, 1996 e 1997). Per questo proposito, assumiamo come nota la classificazione dei collegi elettorali della Camera dei Deputati in otto gruppi (Profondo Nord 104 collegi, Medio Nord 81, Nord Urbano 56, Roma 24, classe Media 33, Profondo Sud 79, Medio Sud 57 e Sud Urbano 41) e ci chiediamo in che misura questa classificazione è riproducibile usando l'analisi discriminante lineare e le Rna *feedforward* con strati nascosti.

Per valutare le differenze tra i gruppi saranno usate, nei due procedimenti, venti variabili (per ulteriori specificazioni cfr. Di Franco, 1995 e 1997) rilevate per connotare le caratteristiche dei collegi:

1. Tasso natalità;
2. Tasso unità locali;
3. Tasso di disoccupazione;
4. Tasso di analfabetismo;
5. Tasso di scolarizzazione superiore;
6. Tasso di sofferenza matrimoniale;
7. Tasso di ritirati dal lavoro;
8. Tasso di terziario avanzato;
9. Tasso di occupati nel primario;
10. Tasso di occupati nel secondario;
11. Tasso di occupati nel terziario;
12. Indice di sviluppo economico alto;
13. Indice di sviluppo economico medio;
14. Tasso di unità locali nell'industria;
15. Tasso di unità locali nel commercio;
16. Tasso di unità locali nei servizi;
17. Rapporto addetti industria su unità locali nell'industria;
18. Rapporto addetti nel commercio su unità locali nel commercio;
19. Rapporto addetti nei servizi su unità locali nei servizi;
20. Concentrazione demografica nel collegio.

Esaminiamo per primi i risultati prodotti dall'analisi discriminante lineare. In questo caso il numero delle funzioni canoniche discriminanti estraibili è uguale a 7 (il numero di classi meno 1), e delle venti variabili trattate, solo due — la percentuale di occupati nel terziario e la percentuale di unità locali nell'industria sul totale delle unità locali — non passano il *test* di tolleranza per cui non vengono usate nel procedimento di estrazione delle funzioni discriminanti.

L'obiettivo dell'analisi discriminante lineare consiste in una trasformazione della matrice delle correlazioni tra le variabili impiegate che massimizzi il rapporto tra varianza tra i gruppi e varianza nei gruppi sotto la condizione che in ogni gruppo le variabili siano distribuite normalmente e che le matrici di covarianza per tutti i gruppi siano uguali. La prima funzione discriminante estratta è una combinazione lineare delle p variabili rilevate su n casi classificati in k gruppi che massimizza il rapporto tra la varianza tra i gruppi e quella interna ai gruppi delle p variabili (Di Franco, 1997). Le successive funzioni discriminanti si calcolano in maniera analoga aggiungendo il vincolo dell'ortogonalità con la prima. Una volta individuate le funzioni discriminanti, si possono calcolare i coefficienti di correlazione tra le variabili originali e le funzioni discriminanti e i punteggi dei casi sulle funzioni discriminanti. A questo punto si può sfruttare l'analisi discriminante in funzione previsionale, riclassificando gli stessi casi, di cui si conosce l'appartenenza ai k gruppi, secondo le funzioni discriminanti, o attribuendo dei nuovi casi, dei quali si conoscono i valori sulle variabili, ad una delle k classi. La procedura di classificazione avviene in questo modo. Per ogni gruppo noto si calcola un centroide considerando le medie dei casi di ogni gruppo rispetto ai loro punteggi sulle funzioni discriminanti. Nel nostro caso quindi si definiscono otto centroidi (uno per ogni gruppo). Si calcola poi la distanza di ogni caso, usando i suoi punteggi sulle funzioni discriminanti, nei confronti di ogni centroide e si assegna (con una data probabilità) al gruppo verso cui presenta la distanza minore. Il confronto tra le assegnazioni prodotte con i punteggi discriminanti e l'appartenenza effettiva di ogni caso ad un gruppo, che ovviamente è nota, consente il calcolo della percentuale di classificazioni corrette.

Nella tabella 1 si mostrano le sette Funzioni Canoniche Discriminanti (F_{cn}) che risultano dall'analisi. La prima ha un autovalore di 19,14 e riproduce la varianza tra i gruppi in rapporto alla varianza nei gruppi per un 48,4%; il valore della correlazione canonica è pari a .97, mentre λ è molto basso (0,004). Quindi, con il test del χ^2 , possiamo rifiutare l'ipotesi che le medie nei gruppi siano uguali con una bassissima probabilità di sbagliare. Elevando al quadrato il coefficiente di correlazione canonica, sappiamo che la proporzione di varianza della prima funzione discriminante riprodotta dall'appartenenza alle otto classi di collegi è pari al 95%.

La seconda funzione ha un autovalore pari a 14,8 e riproduce il 37,4% della varianza; la correlazione canonica è .97 e il valore di λ è di 0,07. Anche la seconda funzione è significativa, mentre la proporzione della sua varianza riprodotta dalle differenze tra i gruppi è del 94%.

La terza funzione ha un autovalore di 4,31, che equivale al 10,9% di varianza riprodotta. Così, sommando le prime tre funzioni discriminanti, otteniamo il 96,7% della varianza riprodotta. Il valore della correlazione canonica della terza funzione è di .90, mentre λ è di 0,35. La proporzione di varianza della terza funzione riprodotta dall'appartenenza dei collegi ai gruppi è dell'81%.

Tab. 1: Le funzioni canoniche discriminanti estratte dall'analisi

Fnc	Autovalore	% di Varianza	% Cum. Varianza	Correlazione Canonica	Dopo Fnc	Wilks' Lambda	Chi ²	Gradi di Libertà	Significati vità
1	19,14	48,4	48,4	.97	0	.00	3899,68	126	.000
2	14,80	37,4	85,8	.97	1	.00	2519,05	102	.000
3	4,31	10,9	96,7	.90	2	.07	1249,74	80	.000
4	0,71	1,8	98,5	.64	3	.35	481,62	60	.000
5	0,50	1,3	99,7	.58	4	.60	235,84	42	.000
6	0,10	0,3	100,0	.31	5	.90	49,12	26	.004
7	0,01	0,0	100,0	.09	6	.99	3,86	12	.985

Tra le sette funzioni estratte dall'analisi, le prime tre, cui corrispondono i coefficienti di correlazione canonica che massimizzano le differenze tra i gruppi, sono quelle che hanno la gran parte delle loro varianze riprodotte dalle differenze riscontrate tra i gruppi. Infatti, i coefficienti lambda delle prime tre funzioni, che indicano la proporzione di varianza non riprodotta dalla differenza tra i gruppi, sono bassissimi (soprattutto i primi due). Le altre quattro funzioni rappresentano, invece, quote di variabilità residue e solo parzialmente attribuibili alla differenza tra i gruppi. Tali risultati sono significativamente corroborati dal test del Chi^2 , calcolato sul coefficiente lambda considerato come una variabile.

Le funzioni discriminanti sono semanticamente interpretabili valutando i coefficienti standardizzati della funzione discriminante (che sono comparabili ai coefficienti di regressione standardizzati nell'equazione di regressione multipla). Essi sono forniti da quelle variabili che massimizzano il rapporto tra la devianza tra i gruppi e la devianza entro i gruppi; cioè quelle che discriminano maggiormente i gruppi. Per ragioni di spazio e per la finalità dei nostri scopi omettiamo di presentare questi coefficienti e passiamo a controllare i risultati ottenuti con l'analisi.

Per prima cosa è utile controllare la figura 11 nella quale si presenta la mappa territoriale costruita intersecando le prime due funzioni discriminanti. Nel grafico sono riprodotti i confini tra i territori predetti nei quali probabilmente ricadranno gli otto gruppi indicati dall'etichetta numerica di ognuno (1 = Profondo Nord, 2 = Medio Nord, 3 = Nord Urbano, 4 = Roma, 5 = Media, 6 = Sud Urbano, 7 = Medio Sud, 8 = Profondo Sud). I centroidi dei gruppi sono indicati con un asterisco e sono all'interno dei confini indicati dal numero che rappresenta il gruppo. Considerando nel suo insieme la figura è evidente che gli otto gruppi non sono separabili linearmente.

La tabella 2 riassume i risultati della classificazione dei collegi ottenuta con l'analisi discriminante lineare. Dei 475 collegi 423 (89%) sono stati classificati correttamente. In dettaglio, 97 dei 104 collegi della classe del Profondo Nord si ritrovano nella stessa classe, mentre 4 (3,8%) sono stati collocati nel Medio Nord e 3 (2,9%) nella classe Media e Dispersa. Per la classe del Medio Nord (81 collegi) ne troviamo 75 ben classificati, mentre 2 sono stati classificati nella classe Profondo Nord, 1 nel Nord Urbano e 3 nella classe Media e Dispersa. Dei 56 collegi della classe del Nord Urbano ne troviamo 50 nella stessa classe, 4 nel Medio Nord e 2 nella classe Media. Tutti i 24 collegi della classe Roma sono ben classificati dalla procedura. Nella classe Media e Dispersa (33 collegi) si registrano 24 collegi ben classificati, 2 nel Profondo Nord, 6 nel Medio Nord e 1 nel Medio Sud. Dei 41 collegi del Sud Urbano, 38 sono ben classificati e 3 vengono assegnati alla classe Media e Dispersa. La classe del Medio Sud è quella che fa registrare il più alto numero di cattive classificazioni. Infatti, dei 57 collegi solo 41 sono nella stessa classe, mentre 10 sono nella classe Media e Dispersa, 2 nel Sud Urbano e 4 nel Profondo Sud. Infine il Profondo Sud (79 collegi) ha 74 collegi ben classificati

e 5 spostati nella classe del Medio Sud. In sintesi, l'analisi discriminante riproduce molto bene sei classi (con percentuali di assegnazioni corrette intorno al 90% dei casi), mentre ha delle difficoltà a classificare correttamente i collegi della classe Media e Dispersa (72,7% di assegnazioni corrette) e del Medio Sud (71,9%).

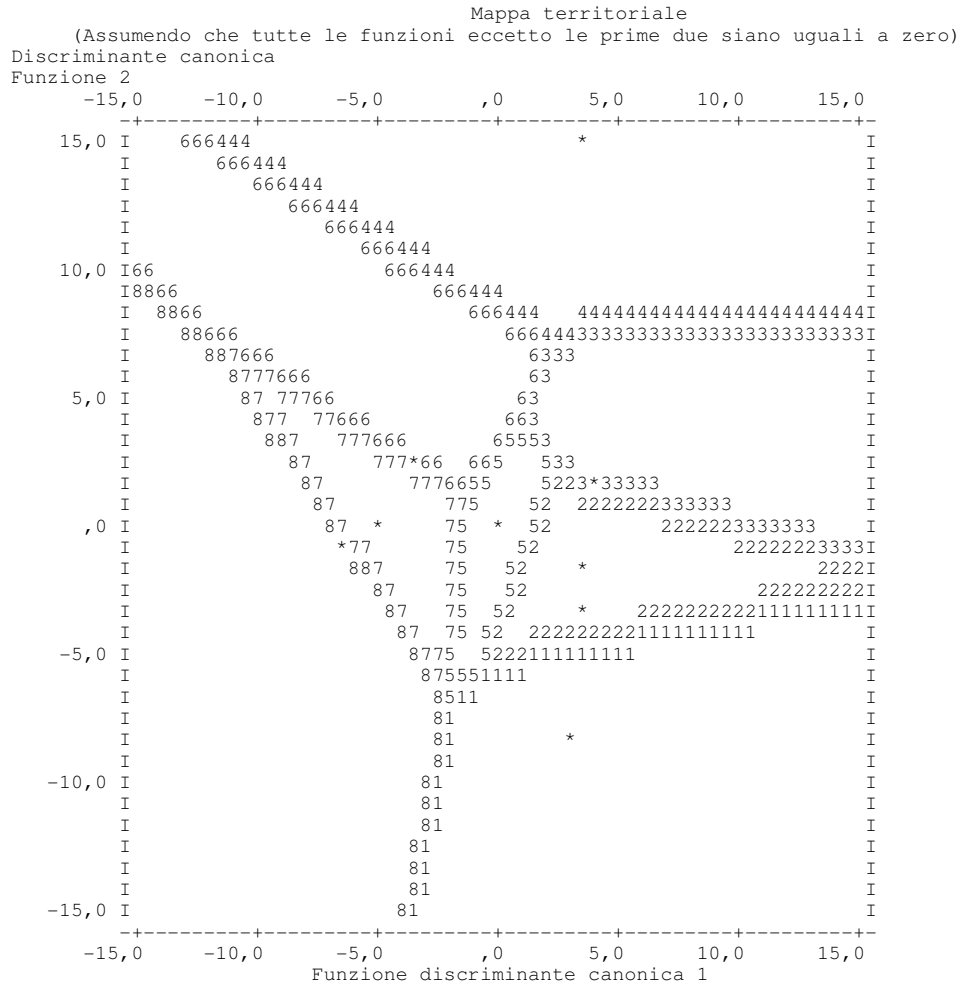


Figura 11: La mappa territoriale delle otto classi di collegi ottenuta sulle prime due funzioni discriminanti

Tab. 2: Risultati della classificazione ottenuta con l'analisi discriminante

Gruppo Noto	Numero di Casi	Appartenenza di Gruppo Predetta dall'Analisi discriminante lineare							
		1	2	3	4	5	6	7	8
Profondo Nord	104	97 93,3%	4 3,8%	0 0,0%	0 0,0%	3 2,9%	0 0,0%	0 0,0%	0 0,0%
Medio Nord	81	2 2,5%	75 92,5%	1 1,2%	0 0,0%	3 3,7%	0 0,0%	0 0,0%	0 0,0%
Nord Urbano	56	0 0,0%	4 7,1%	50 89,3%	0 0,0%	2 3,6%	0 0,0%	0 0,0%	0 0,0%
Roma	24	0 0,0%	0 0,0%	0 0,0%	24 100,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%
Media e Dispersa	33	2 6,1%	6 18,2%	0 0,0%	0 0,0%	24 72,7%	0 0,0%	1 3%	0 0,0%
Sud Urbano	41	0 0,0%	0 0,0%	0 0,0%	0 0,0%	3 7,3%	38 92,7%	0 0,0%	0 0,0%
Medio Sud	57	0 0,0%	0 0,0%	0 0,0%	0 0,0%	10 17,5%	2 3,5%	41 71,9%	4 7%
Profondo Sud	79	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	5 6,3%	74 93,7%

Passiamo ora ad illustrare i risultati ottenuti con le Rna. Innanzi tutto bisogna dire che, come sappiamo, il primo problema consiste nella definizione del modello neurale. Infatti non conosciamo a priori la migliore architettura della rete in funzione del nostro obiettivo di classificazione, sappiamo, però, che per un problema di classificazione il modello di rete *feedforward* con almeno uno strato di unità nascoste e con un apprendimento supervisionato assolve bene il compito. Il problema riguarda perciò il numero di unità nascoste da inserire tra le venti unità di *input* (una per ogni variabile disponibile) e l'unica unità di *output* (che riproduce la classificazione delle otto Italie). Si è proceduto al controllo di diverse architetture di rete partendo da una rete con 20 unità di *input*, sette unità nascoste e una di *output*, incrementando le unità nascoste di una unità negli esperimenti successivi, variando anche il tempo di apprendimento per controllare il problema dell'*overtraining*. Nella tabella 3 si riportano i risultati di questi esperimenti. Come si può constatare leggendo la colonna dell'errore medio, ognuna delle dieci Rna provate presenta una ottima capacità di riproduzione della classificazione. Non avendo variato nessun altro parametro nella procedura di addestramento, i diversi risultati delle Rna dipendono dall'iniziale assegnazione casuale dei pesi sui legami e dalla durata dell'apprendimento (cfr. colonna epoche di apprendimento in tab.3). In generale si può dire che la numerosità delle unità nascoste dipende dalla complessità del problema, per cui più il compito è difficile e più unità nascoste sono necessarie per risolverlo (cercando di non superare, comunque, il numero di unità di *input*). Inoltre, maggiore è la complessità dell'architettura della rete e maggiore sarà il tempo necessario per la fase di addestramento. La decima Rna, dopo diecimila epoche di addestramento ha raggiunto un errore medio molto piccolo (cinque per mille). In questo caso, ovviamente la rete riesce a classificare correttamente tutti i 475 collegi (tab. 4).

Tab. 3: I risultati di dieci Rna con diverse architetture e diverse durate di apprendimento

Architettura della Rna (<i>input-nascoste-output</i>)	Epoche di apprendimento	Errore medio
1) 20-7-1	1.300	0,063
2) 20-7-1	2.700	0,090
3) 20-8-1	1.300	0,070
4) 20-8-1	2.700	0,062
5) 20-9-1	1.300	0,059
6) 20-9-1	2.700	0,056
7) 20-10-1	1.300	0,062
8) 20-10-1	2.700	0,051
9) 20-10-1	9.500	0,044
10) 20-10-1	10.000	0,005

Tab. 4: Risultati della classificazione ottenuta con la Rna *feedforward* con un strato nascosto di dieci unità dopo 10.000 epoche di apprendimento

Gruppo Noto	Numero di Casi	Appartenenza di Gruppo Predetta dalla Rna <i>feedforward</i> con uno strato nascosto							
		1	2	3	4	5	6	7	8
Profondo Nord	104	104 100%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%
Medio Nord	81	0 0,0%	81 100%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%
Nord Urbano	56	0 0,0%	0 0,0%	56 100	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%
Roma	24	0 0,0%	0 0,0%	0 0,0%	24 100,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%
Media e Dispersa	33	0 0,0%	0 0,0%	0 0,0%	0 0,0%	33 100%	0 0,0%	0 0,0%	0 0,0%
Sud Urbano	41	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	41 100%	0 0,0%	0 0,0%
Medio Sud	57	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	57 0,0%	0 0,0%
Profondo Sud	79	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	0 0,0%	79 100%

Passiamo ad una seconda applicazione usando ancora la banca dati sui collegi elettorali. Ci poniamo ora un problema di previsione confrontando le Rna con un modello di regressione lineare multipla riprendendo una nostra applicazione utile alla riproduzione dell'area del non voto (costituita dalla somma delle percentuali sugli elettori delle astensioni più i voti non validi, sigla ANV94) alle elezioni del 1994 su base ecologica (Di Franco, 1997). Si vuole quindi una stima dell'area del non voto nei collegi elettorali della Camera alle elezioni politiche del 1994 esprimendola come funzione lineare di quattro variabili esplicative:

1. un indice di modernizzazione/arretratezza economica (tasso di disoccupazione sigla TDIS);
2. un indice di modernizzazione/arretratezza socio-demografica (percentuale di analfabeti e alfabeti privi di titolo di studio ANAL);
3. un indice del livello di urbanizzazione-terziarizzazione (percentuale di occupati nel terziario sul totale degli occupati OCC3);
4. la somma dei voti ottenuti alle elezioni del 1994 dal PDS e da RC (PDS_RC94).

Per maggiori dettagli sulla scelta delle variabili inserite nel modello si rinvia a Di Franco (1997).

Nella tabella 5 si presenta la matrice delle correlazioni tra le cinque variabili inserite nel modello di regressione multipla.

Tab. 5: La matrice delle correlazioni tra le cinque variabili del modello di regressione multipla lineare

TDIS	1.0				
ANAL	.69	1.0			
OCC3	.33	.01	1.0		
PDS_RC	-.02	.15	.07	1.0	
ANV94	.77	.69	.24	-.11	1.0
	TDIS	ANAL	OCC3	PDS_RC	ANV94

Considerando l'entità dei coefficienti di correlazione tra le variabili inserite nel modello, possiamo escludere problemi di multicollinearità. Si può quindi procedere al calcolo dei coefficienti della regressione multipla della variabile ANV94 sulle quattro variabili indipendenti, dalla quale otteniamo i seguenti risultati con il criterio diretto per l'ingresso delle variabili nell'equazione:

R (coefficiente di correlazione multipla) = .903;
 R^2 (coefficiente di determinazione) = .816;
Adjusted R² (coefficiente di determinazione corretto) = .814;
Standard Error (errore standard della stima) = 4,042.

Questi risultati esprimono un buon adattamento del modello ai dati effettivi: l'81% della varianza di ANV94 è riprodotta dal modello. Nella tabella 6 si mostrano i risultati dei coefficienti di regressione multipla delle variabili indipendenti.

Tab. 6: I risultati del modello di regressione multipla lineare

Variabile	B	SE B	Beta
PDS_RC94	-0,19	0,017	-.227
TDIS	0,44	0,063	.223
OCC3	0,13	0,016	.177
ANAL	0,98	0,042	.699
Costante	-1,63	1,048	

Nella tabella per ogni variabile indipendente sono indicati i seguenti coefficienti:

B = coefficiente di regressione multipla non standardizzato;
 SE B = errore *standard* del coefficiente di regressione multipla non standardizzato;
 Beta = coefficiente di regressione multipla standardizzato;

Nell'ultima riga sono indicati il valore dell'intercetta e il suo errore *standard*. Ovviamente nella colonna dei coefficienti di regressione multipla standardizzati la costante non compare. In questo caso, lavorando con l'intera popolazione dei collegi elettorali della Camera, non abbiamo problemi di inferenza, per cui possiamo assumere tutti i coefficienti come significativi (diversi da zero).

L'equazione di regressione multipla, usando i coefficienti di regressione standardizzati, è:

$$ANV94 = .70(ANAL) - .23(PDS_RC94) + .22(TDIS) + .12(OCC3)$$

I coefficienti di regressione multipla standardizzati ci permettono di confrontare l'influenza delle singole variabili indipendenti sulla dipendente. D'altra parte, questi coefficienti hanno un valore relativo, perché valgono solo per le variabili considerate nel modello; immettendo o togliendo delle altre variabili, infatti, essi assumono dei valori diversi. In questo senso l'assunto della corretta specificazione del modello è determinante per la validità teorica del modello di regressione adottato.

Possiamo concludere che la variabile più influente nella riproduzione delle percentuali dell'area del non voto del 1994 nei collegi, tra le quattro considerate, è la percentuale di analfabeti presenti nei collegi; tuttavia anche le altre tre variabili, seppure in misura minore, influenzano la variabile dipendente.

Per controllare il rispetto degli assunti della regressione multipla, analizziamo la distribuzione dei residui della regressione standardizzati (fig. 12).

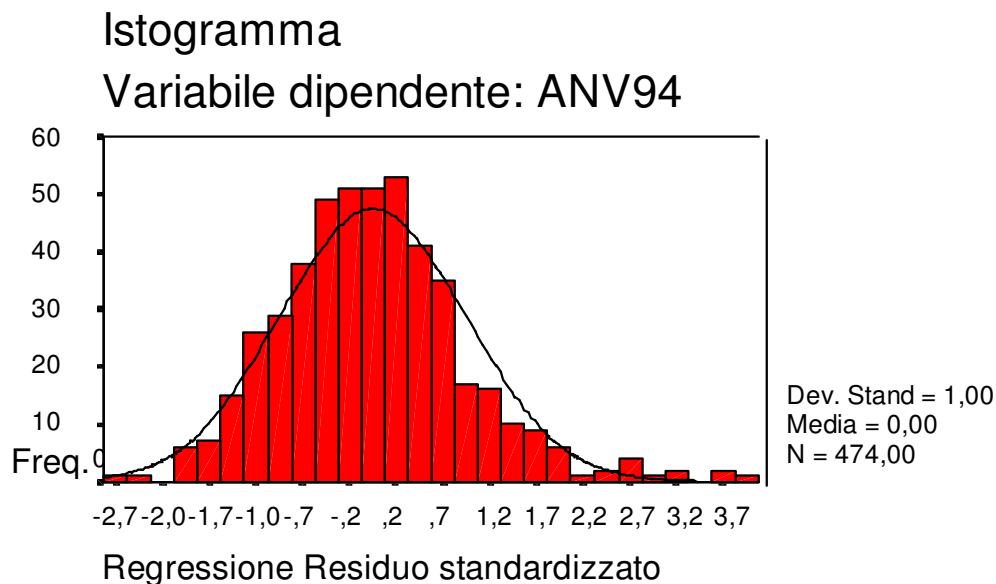


Figura 12: Istogramma dei residui standardizzati della regressione multipla

L'istogramma mostra una buona approssimazione della curva normale, anche se evidenzia uno sbilanciamento verso i valori più alti della media e un profilo leggermente più leptocurtico (appuntito). Ciò è causato dalla presenza di cinque valori anomali tutti molto al di sopra della media dell'area del non voto (che è del 19,8%). In questi cinque collegi (anche a causa di anomalie nell'offerta partitica) si sono registrate percentuali dell'area del non voto nel 1994 di più del doppio della media (rispettivamente del 46,8%, 46,8%, 49,4%, 50,3% e 53,5%). La presenza di questi cinque collegi anomali pregiudica l'efficienza del modello, per cui si potrebbero escludere e ricalcolare i coefficienti della regressione multipla. Per i nostri fini, al contrario, è preferibile lasciare questi casi per valutare il comportamento delle Rna nella stima dei loro stati sulla variabile dipendente.

Il problema della stima dell'area del non voto può essere affrontato con una Rna *feedforward* con uno strato di unità nascoste. Anche in questo caso si pone il problema dell'architettura della rete, tenendo presente che in questo caso il problema è più complesso rispetto a quello precedente. Infatti, l'*output* della rete deve fornire per ogni collegio un valore quantitativo (e non assegnare ogni caso ad un numero limitato di classi). Per economia di spazio, non presentiamo tutti gli esperimenti fatti, che sono stati molti, ci limitiamo a dire che

la rete migliore (costituita da quattro nodi di *input*, otto nascosti e uno di *output*), dopo un apprendimento durato duemila epoche, ha raggiunto un errore medio pari a 0,346. Ovviamente questa rete non rappresenta la migliore soluzione in assoluto, si potrebbe andare avanti con gli esperimenti per cercarne altre migliori, ma non è questo quello che ci interessa.

Nella tabella 7 si presenta un confronto tra i valori caratteristici dell'area del non voto effettiva del 1994, e dalle stime prodotte dalla regressione multipla e della Rna.

Tab. 7: Confronto tra i valori caratteristici dell'area del non voto 1994 e le stime prodotte dalla regressione multipla e dalla Rna.

	Media	Scarto Tipo	Varianza	Minimo	Massimo
ANV94 valori effettivi	19,78	9,38	88,001	7,16	53,49
Stima ANV94 con la regressione multipla	19,78	8,47	71,803	6,42	40,74
Stima ANV94 con la Rna	19,78	8,80	77,513	7,79	44,92

Si può notare come sia il modello di regressione multipla sia la rete riproducano perfettamente la media effettiva dell'area del non voto, mentre meno bene gli altri valori caratteristici. In questo caso la rete si avvicina sempre di più al valore effettivo degli altri valori caratteristici (scarto tipo, varianza, minimo e massimo) dimostrando di avere un rendimento migliore rispetto al modello di regressione multipla. Nella tabella 8 si presentano le correlazioni tra i valori effettivi dell'area del non voto e quelli stimati con i due procedimenti.

Tab. 8: Matrice di correlazione tra i valori effettivi dell'area del non voto e le stime prodotte con i modelli di regressione multipla e quello neurale

ANV94 effettiva	1,000		
ANV94 regressione	0,903	1,000	
ANV94 rete	0,939	0,963	1,000
	ANV94 effettiva	ANV94 regressione	ANV94 rete

Anche in questo caso la rete dimostra una migliore capacità di riproduzione della varianza dell'area del non voto effettiva. Infatti calcolando il coefficiente di determinazione tra le stime prodotte dalla rete e i valori effettivi otteniamo un risultato pari all'88% a fronte dell'81% ottenuto con la regressione multipla. Questo migliore risultato della rete si spiega in quanto riesce ad avvicinare di più i risultati dei cinque collegi anomali che abbiamo prima evidenziato e che, evidentemente, non sono linearmente dipendenti dalle variabili indipendenti del modello di regressione.

Nel paragrafo precedente avevamo lasciato in sospeso la presentazione di due dispositivi presenti nel programma Neural Connection, il simulatore e il modulo *What if*, (SPSS inc., 1997b), perché ci eravamo riservati la possibilità di metterli alla prova. Come si è detto, anche quando una rete riesce a fornire risultati eccellenti, questi risultati non sono immediatamente intelligibili. Per cui prima si deve lavorare per mettere a punto una rete che funzioni bene, e abbiamo visto che non è una cosa immediata, e poi si deve studiare il funzionamento della rete per scoprire perché funziona bene. In questo campo sono stati proposti vari metodi, abbiamo già fatto riferimento alla *cluster analysis* e al lesionamento della rete, ora vediamo come funzionano i già citati dispositivi del simulatore e del *What if* su Neural Connection. Questi dispositivi, consentono, in qualche modo, di interrogare la rete e chiederle in relazione di cosa vari il risultato. Per prima cosa si selezionano obbligatoriamente due variabili per volta, tra tutte quelle presenti nello strato di *input*, attraverso il simulatore, mentre tutte le altre si tengono su un valore costante. Poi si accede al modulo *What if* dove le due variabili si possono collocare su due valori di riferimento qualsiasi, ad esempio le loro

rispettive medie, e quindi si fa variare una variabile in un certo modo per vedere come questa variazione influenza l'uscita della rete. Questo procedimento esemplifica la funzionalità del metodo simulativo, in quanto è possibile andare al di là dei dati effettivi facendo ipotesi che sono oltre il campo di variazione dei dati raccolti. Questo è possibile perché la rete ha appreso una funzione che varia in un intervallo di valori infiniti per cui, se la funzione appresa è effettivamente sottostante al fenomeno studiato, si può mettere a frutto convenientemente la sua capacità di generalizzazione.

Per esemplificarne il funzionamento, abbiamo interrogato la rete scegliendo il tasso di analfabetismo ed il tasso di disoccupazione che sono stati posti sulle loro medie (rispettivamente 15,2% e 8,7%) e poi abbiamo posto quattro domande (naturalmente avremmo potuto effettuarne numerose altre, anche scegliendo le altre variabili di *input* disponibili) ottenendo le seguenti risposte:

- a) quando il tasso di disoccupazione aumenta del 58,9% (passando da 8,7% a 13,8%) la stima dell'area del non voto aumenta del 38% (dal 15,6% al 21,5%);
- b) quando il tasso di disoccupazione aumenta dell'88% (da 8,7% a 16,4%) la stima dell'area del non voto aumenta del 69,6% (dal 15,6% al 26,4%);
- c) quando il tasso di disoccupazione aumenta del 110% (da 8,7% a 18,3%) la stima dell'area del non voto aumenta del 56,6% (dal 15,6% al 24,4%, si noti come in questo caso abbiamo una relazione non lineare non monotona);
- d) quando il tasso di disoccupazione diminuisce del 73,7% (da 8,7% a 2,3% che è un valore non presente in nessun collegio) la stima dell'area del non voto diminuisce del 30,1% (dal 15,6% al 10,9%).

In conclusione, riteniamo che questi due dispositivi siano effettivamente molto utili, anche se dovrebbero essere potenziati perché in una rete con molte unità di *input* il numero dei confronti tra coppie di variabili cresce tanto rapidamente che diventa praticamente improponibile effettuare tutti i possibili controlli. D'altra parte sarebbe auspicabile che il ricercatore sapesse preventivamente quali variabili e quali ipotesi di relazioni tra le variabili porre sotto controllo.

Riferimenti bibliografici

- M. Buscema, G. Didoné, M. Pandin, 1994, *Reti neurali autoriflessive. Teoria, metodi, applicazioni, confronti*, Roma, Armando.
- M. Buscema, 1994, *Squashing Theory. Modello a reti neurali per la previsione dei sistemi complessi*, Roma, Armando.
- M. Buscema, F. Matera, T. Nocentini, P.L. Sacco, 1997, *Reti neurali e finanza*, Roma, Armando.
- S. Cammarata, 1990, *Reti neurali. Una introduzione all'altra intelligenza artificiale*, Etaslibri.
- V. Capecchi, 1996, "Tre Castelli, una Casa e la Città inquieta", in Cipolla De Lillo (a c. di), 1996.
- C. Cipolla, A. De Lillo, (a c. di), 1996, *Il Sociologo e le sirene. La sfida dei metodi qualitativi*, Milano, Angeli.
- R. Cipriani, S. Bolasco, (a c. di), 1995, *Ricerca qualitativa e computer. Teorie, metodi e applicazioni*, Milano, Angeli.
- P. M. Churchland, 1995, *Il motore della ragione la sede dell'anima. Viaggio attraverso il cervello umano*, Milano, Il Saggiatore.
- Q. Conte, 1997, "Il metodo simulativo", in Ricolfi (a c. di), 1997.
- G. Di Franco, 1995, "Una metodologia per l'analisi ecologica dei risultati elettorali: le elezioni politiche del marzo 1994". In *Sociologia e Ricerca Sociale*, n.47-48, pp. 151-178.
- G. Di Franco, 1996, "Le otto Italie della Camera e del Senato. Caratteristiche socio-economiche dei collegi elettorali della Camera e del Senato", in *Sociologia e Ricerca Sociale*, n.50, pp.22-49.
- G. Di Franco, 1997, *Tecniche e modelli di analisi multivariata dei dati*, Roma, Seam.
- G. Fabbri, R. Orsini, 1993, *Reti neurali per le scienze economiche*, Milano, Franco Muzzio.
- Keywords, 1995, n.58, *Neural Connection from SPSS. Use neural network technology to help build and perform analysis more efficiently than traditional methods*, Chicago, SPSS inc.
- M. Negrotti, D. Donnanno, G. Sacchi, 1995, "Tecnologia della previsione: expert system e neural nets", in Cipriani e Bolasco (a c. di), 1995, pp.259-269.
- D. Parisi, 1989, *Intervista sulle reti neurali. Cervello e macchine intelligenti*, Bologna, Il Mulino.
- E. Pessa, 1993, *Reti neurali e processi cognitivi*, Roma, Di Rienzo.
- P.T. Quinlan, 1996, *Connessionismo e psicologia. Una prospettiva psicologica per la ricerca sulle reti neurali*, Bologna, Il Mulino.
- L. Ricolfi (a c. di), 1997a, *La ricerca qualitativa*, Roma, NIS.
- L. Ricolfi, 1997b, "La ricerca empirica nelle scienze sociali: una tassonomia", in Ricolfi (a c. di), 1997a
- D.E. Rumelhart, J.L. McClelland (a c. di), 1986, *Parallel distributed processing, vol.1 Foundations, vol.2 Psychological and biological models*, Cambridge, Mit Press; tr. it., *PDP. Microstruttura dei processi cognitivi*, Bologna, Il Mulino, 1991.
- SPSS per Windows, 1995, *Create il sistema perfetto per soddisfare le vostre necessità!*, Bologna, SPSS Italia.
- SPSS Inc., 1997a, *Neural Connection 2.0 User's Guide*, Chicago, SPSS inc.
- SPSS Inc., 1997b, *Neural Connection 2.0 Applications Guide*, Chicago, SPSS inc.
- H. White, 1988, *Economic prediction using neural network: che case of IBM dialy stock returns*, San Diego, University of California.
- H. White, 1989, "An additional hidden test for neglected nonlinearity in multilayer feedforward networks", in *Proceeding of the International Joint Conference on neural networks*, Washington.

H. White, 1989, *Learning in artificial neural networks: a statistican prospective*, San Diego, University of California.