Costruzione di una Rete Neurale Artificiale per applicazioni Economico–Finanziarie

Prof. Crescenzio Gallo

c.gallo@unifg.it

Università degli Studi di Foggia

Dipartimento di Scienze Biomediche

Indice

Elenco delle tabelle

Elenco delle figure

Capitolo 1

Concetti preliminari

1.1 Introduzione

Negli ultimi anni, anche in campo economico finanziario, sta suscitando notevole interesse una nuova classe di modelli non lineari caratterizzati da un'architettura tesa a riprodurre il cervello umano noti come reti neuronali o più semplicemente reti neurali.

Le reti neurali possono rappresentare un vantaggio competitivo rispetto ai metodi tradizionali quali analisi statistica ed econometrica. Esse, infatti, si prestano ad un ampia gamma di applicazioni grazie alle loro capacità di approssimazione universale, apprendimento da osservazioni sperimentali, classificazione e generalizzazione.

1.2 Reti neurali naturali e artificiali

Le reti neurali artificiali nascono dalla volontà di simulare artificialmente l'organizzazione ed il funzionamento fisiologici delle strutture cerebrali umane.

È quindi necessario riferirsi alla rete neurale naturale. Essa è costituita da un grandissimo numero di cellule nervose (una decina di miliardi nell'uomo), dette neuroni, collegate tra loro in una complesse rete. Il comportamento intelligente è frutto delle numerose interazioni tra unità interconnesse. L'input di un neurone è costituito dai segnali di uscita dei neuroni ad esso collegati. Quando il contributo di questi ingressi supera una determinata soglia, il neurone, attraverso un'opportuna funzione di trasferimento, genera un segnale bio-elettrico che si propaga, attraverso i pesi sintattici, ad altri neuroni.

Caratteristiche significative di questa rete, che i modelli neurali artificiali intendono simulare, sono:

- il *parallelismo* dell'elaborazione, dovuto al fatto che i neuroni elaborano simultaneamente l'informazione, ed è il flusso informativo stesso che genera il coordinamento fra le varie aree;

- la *duplice funzione* del neurone che agisce allo stesso tempo da memoria e da elaboratore di segnali;

- il *carattere distribuito* della rappresentazione dei dati, ossia la conoscenza è distribuita in tutta la rete e non circoscritta o predeterminata;

- la *possibilità* della rete *di apprendere dall'esperienza*.

È proprio quest'ultima fondamentale capacità a consentirle di auto-organizzarsi, di adattarsi alle nuove informazioni in ingresso e di estrarre dagli esempi conosciuti i caratteri di specificità e di regolarità che stanno alla base della loro organizzazione.

Una rete neurale artificiale acquisisce questa attitudine in una opportuna fase di apprendimento.

1.3 Struttura di una rete neurale

Le reti neurali sono costituite da unità computazionali elementari (i neuroni) note come *Unità Elaborative*. I neuroni sono combinati secondo diverse architetture; per esempio, possono essere organizzati a strati (rete *multi-layer*), oppure possono avere una topologia in cui ogni neurone è collegato a tutti gli altri (rete completamente connessa). Nel resto di questa trattazione, si farà riferimento principalmente a reti a strati, composte da:

- l'*input* layer, costituito da n neuroni pari al numero di input della rete;

- l'*hidden* layer, composto da uno o più strati nascosti (o intermedi) costituito da m neuroni;

- l'*output* layer, costituito da p neuroni pari al numero di output desiderati.

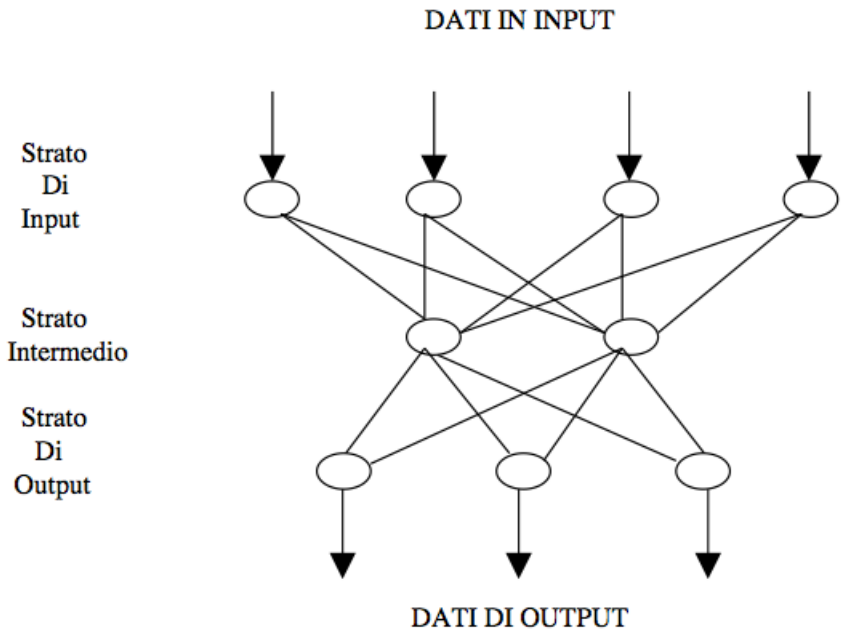

Figura 1.1: Struttura di rete neurale multi-layer di tipo feedforward

Le modalità di connessione permettono di distinguere tra due tipi di architetture. Nelle architetture *feedback*, la presenza di connessioni tra neuroni dello stesso strato o tra neuroni dello strato precedente realizza un collegamento di retroazione. Nelle architetture *feedforward*, le connessioni tra gli strati sono tali da non consentire retroazioni tra strati e quindi il segnale è trasmesso solo ai neuroni appartenenti allo strato successivo.

1.4 Il modello elementare del neurone artificiale

È l'unità fondamentale ed elemento di calcolo delle reti neurali, proposta da MCCulloch e Pitts nel 1943.

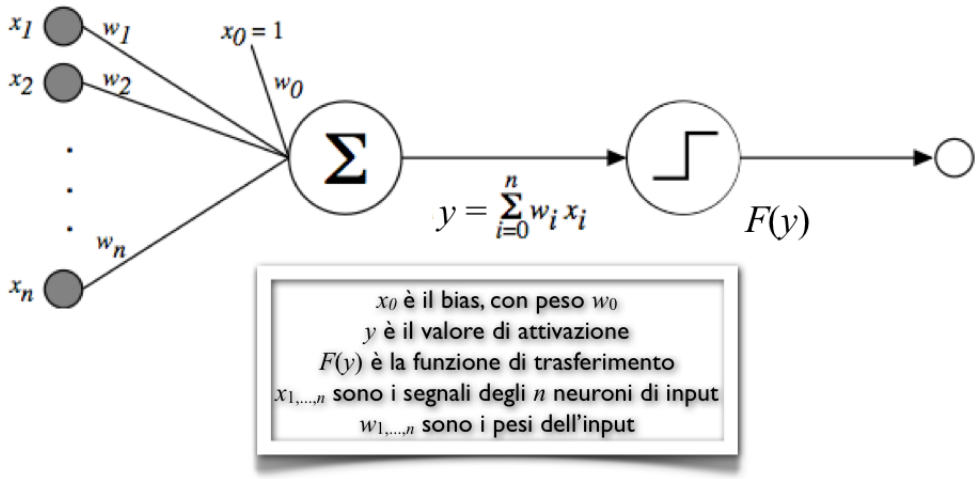

Figura 1.2: Un neurone artificiale elementare

Ogni neurone riceve in input n segnali dagli altri neuroni (il vettore $\mathbf{x}$), tramite connessioni di intensità $\mathbf{w}$ (pesi sinaptici). I segnali di input vengono consolidati in un potenziale post-sinaptico y, che è la somma pesata degli input.

La funzione somma calcola, così, il valore di attivazione che viene poi trasformato nell'output $F(y)$ da un'opportuna funzione di trasferimento o di attivazione.

I neuroni dello strato di input non hanno nessun input. Il loro stato di attivazione corrisponde ai dati in input alla rete, non eseguono nessun calcolo e la funzione di attivazione trasferisce il valore di input alla rete senza modificarlo.

La capacità operativa di una rete, ossia la sua conoscenza, è contenuta nelle sinapsi, cioè i pesi delle connessioni di input di ogni neurone. Questi ultimi assumono i valori corretti (cioè valori che permettono alla rete di fornire risposte entro un margine di errore tollerato) grazie all'addestramento.

1.5 Apprendimento di una rete neurale artificiale

La rete neurale non viene programmata in modo diretto ma addestrata esplicitamente, attraverso un algoritmo di apprendimento, per risolvere un dato compito, con un processo che porta

all'apprendimento tramite l'esperienza.

Si distinguono almeno tre tipi di apprendimento: supervisionato, non supervisionato e per rinforzo. Nel caso di apprendimento *non supervisionato*, la rete viene addestrata solamente in base ad un insieme di input, senza fornire il corrispondente insieme di output.

Per l'apprendimento *supervisionato* è necessario individuare, invece, un insieme di esempi consistenti in opportuni campioni degli input e dei corrispondenti output da presentare alla rete affinché questa impari a rappresentarli.

L'apprendimento *per rinforzo* è utilizzato nei casi in cui non è possibile specificare pattern di ingresso-uscita come per i sistemi ad apprendimento supervisionato. Viene fornito un rinforzo al sistema, il quale lo interpreta come un segnale positivo/negativo sul suo comportamento e aggiusta i parametri di conseguenza.

L'insieme delle configurazioni utilizzate per l'apprendimento della rete costituisce l'insieme di apprendimento, detto *learning* o *training* set.

Capitolo 2

Architetture e modelli di reti neurali

L'algoritmo di apprendimento è uno degli elementi più significativi tra i fattori (numero di strati, numero di neuroni, etc.) che concorrono a definire la configurazione specifica di una rete neurale e che, quindi, condizionano e determinano la capacità stessa della rete di fornire risposte corrette allo specifico problema.

Le diverse possibilità di configurazione sono innumerevoli, perciò la scelta della configurazione ottimale deve essere principalmente in funzione dell'obiettivo dell'applicazione.

2.1 Il perceptron

La rete più semplice è costituita da un solo neurone, con n input e un unico output. L'algoritmo di apprendimento di base del *perceptron* analizza la configurazione (*pattern*) d'ingresso e, pesando le variabili tramite le sinapsi, stabilisce a quale categoria d'uscita va associata la configurazione.

Tuttavia, questo tipo di architettura presenta il grosso limite di poter risolvere solo problemi linearmente separabili. Cioè, per ogni neurone di output, i valori di output che attivano il neurone devono essere chiaramente separabili da quelli che lo disattivano, tramite un iper-piano di separazione di dimensione $n - 1$.

2.2 Le reti Multi Layer Perceptron (MLP)

La rete neurale con uno strato di input, uno o più strati di neuroni intermedi e uno strato di output è denominata *Multi Layer Perceptron*. In una rete di tipo feedforward i segnali si propagano dall'input all'output solo attraverso i neuroni intermedi, non avendosi connessioni trasversali, né in retroazione.

Tali reti utilizzano, nella maggior parte dei casi, l'algoritmo di apprendimento *backpropagation*. Esso calcola gli opportuni pesi sintattici tra gli input e i neuroni degli strati intermedi e tra questi e gli output, partendo da pesi casuali e apportando ad essi piccole variazioni, graduali e progressive, determinate dalla stima dell'errore tra il risultato prodotto dalla rete e quello desiderato. Lo schema di apprendimento si basa, dunque, su una sequenza di presentazioni

di un numero finito di configurazioni con cui l'algoritmo di apprendimento converge verso la soluzione desiderata.

Le reti apprendono tramite una serie, talvolta anche prolungata, di tentativi che consentono di modellare in modo (quasi) ottimale i pesi che collegano gli input con l'output passando per i neuroni degli strati nascosti.

Esistono molti altri tipi di architetture più complesse, ma per gli scopi in queste sede proposti terminiamo qui il nostro approfondimento. Infatti, l'architettura supervisionata backpropagation è la più ampiamente utilizzata e diffusa per la capacità che questo insieme di modelli ha di generalizzare i risultati per un ampio numero di problemi finanziari.

Capitolo 3

Applicazioni delle reti neurali in ambito finanziario

Gli sviluppi applicativi delle reti neurali vedono coinvolte diverse componenti del mondo finanziario. Questo grande interesse nei confronti di tali tecnologie rende le applicazioni delle reti neurali talmente numerose da rendere difficile ogni tentativo di classificazione. Possiamo tuttavia provare a schematizzare gli ambiti applicativi in campo finanziario in tre grandi categorie:

- previsione di serie temporali;
- classificazione;
- approssimazione di funzioni.

3.1 Previsione di serie temporali

In quest'ambito rientrano la maggioranza delle applicazioni delle reti neurali nel settore finanziario. Lo scopo di gran parte di queste applicazioni è quello di realizzare profitti speculativi tramite trading (prevalentemente di breve o brevissimo periodo) di attività finanziarie quotate, principalmente azioni, tassi di cambio, future.

Si consideri, ad esempio, la previsione del tasso di cambio: può interessare solo il valore puntuale o piuttosto la tendenza di periodo per suggerire la posizione al cambista. Nel primo caso il risultato della rete neurale viene utilizzato per l'implementazione del trading system: l'indicazione *buy*, *hold* o *sell* viene mostrata mediante frecce che segnalano il successivo rialzo o ribasso della quotazione. Nel secondo caso, quello che interesserà all'analista sarà, non solo sapere che un mercato è in rialzo o in ribasso, ma principalmente avere un'idea dell'entità della fluttuazione per poterla confrontare con le dinamiche previste sugli altri mercati. Servirà, dunque, un sistema previsionale in grado di segnalare la futura variazione del prezzo o del tasso.

In ogni modo, in ciascuna delle possibili applicazioni, da un punto di vista operativo è necessario dividere la serie storica in due parti: l'una, composta dalle cosiddette osservazioni *in-sample*, funge da base per l'addestramento; l'altra, composta dalle osservazioni *out-of-sample*,

ha lo scopo di verificarne la validità. La rete può essere addestrata per fornire previsioni per più ampi orizzonti temporali, utilizzando le sue stesse previsioni a breve termine come input per le previsioni a lungo termine.

Per l'efficienza di questo tipo di applicazioni importante è la valutazione di particolari aspetti tecnici, quali:

- *scelta delle variabili di input*: deve avvenire considerando che la rete non è in grado di fornire alcuna funzione esplicativa e per questo potrebbe utilizzare variabili non significative; infatti, le relazioni tra variabili cambiano nel tempo e di conseguenza input significativi oggi potrebbero non esserlo più in futuro;

- *livello ottimo di apprendimento*: è necessario tener conto che un processo di training troppo corto non consente alla rete di cogliere le relazioni tra le variabili, mentre un training troppo lungo potrebbe rendere la rete incapace di generalizzare (*overtraining*);

- *scelta dell'orizzonte temporale* di riferimento per la previsione: è un fattore importante in quanto orizzonti temporali di previsione molto brevi accrescono il numero di previsioni corrette; di contro, indicazioni su orizzonti temporali di previsione lunghi risultano mediamente meno corrette, ma quelle corrette comportano un profitto medio più elevato.

3.2 Classificazione e discriminazione

Applicazioni tipiche di queste finalità riguardano il rischio di credito: ad esempio, suddivisione in classi di rating, decisioni di affidamento.

Nei casi di classificazione la rete ha il compito di assegnare gli input a un certo numero di categorie predefinite cui corrispondono altrettanti output.

Nei modelli destinati alla discriminazione essa deve anche creare le classi in cui suddividere i dati di input.

3.3 Approssimazione di funzioni

In questo caso, le reti vengono applicate in tutte le funzioni avanzate di *pricing* e di *risk managment* in cui manca una forma funzionale precisa per la valutazione degli strumenti. Si pensi, ad esempio, alle opzioni di tipo americano, alle opzioni esotiche e ai portafogli di opzioni.

Si tratta evidentemente di tematiche avanzate e complesse; proprio per questo numerosi sono gli attuali progetti di ricerca tra cui:

- modelli per l'investimento azionario;

- previsioni per applicazioni di *option pricing*;

- modelli per la valutazione e la negoziazione di *futures*;

- metodologie e apprendimento avanzati.

Capitolo 4

Costruzione di una rete neurale per la previsione finanziaria

Le reti neurali trovano un'ideale applicazione finanziaria in quanto sono in grado di riconoscere un'eventuale dinamica non casuale e non lineare dei prezzi e dei cambi. Solo raramente, infatti, i fenomeni finanziari si manifestano in forma lineare e mai mantengono nel tempo questa regolarità.

La costruzione di una rete neurale prevede le seguenti fasi:

- l'individuazione dell'*obiettivo* della previsione;
- la costruzione dell'*archivio* dei dati su cui attuare l'apprendimento della rete neurale;
- l'*apprendimento* e la scelta dell'*architettura* nell'aspetto dei suoi parametri significativi;
- l'individuazione degli *indicatori di errore* dell'output;
- la *previsione* della serie storica nei mercati finanziari.

4.1 Definizione dell'ambito d'indagine

Le aree di applicazione finanziaria delle reti neurali sono molto diversificate. Si va dalla gestione dei portafogli, alla valutazione dei titoli obbligazionari e azionari; dalle strategie di trading, alla previsione, che può riguardare prezzi azionari, titoli obbligazionari, tassi d'interesse o tassi di cambio.

Ulteriori decisioni relative alla definizione dell'ambito d'indagine sono:

- la *frequenza* dei dati su cui ottenere l'output: questa ovviamente dipende dall'obiettivo della previsione. Ai trader, ad esempio, interessano previsioni a brevissima scadenza e quindi la rete lavorerà su dati ad alta frequenza. Ai gestori di patrimoni o fondi sarà sufficiente un'indicazione meno frequente;
- l'*orizzonte temporale* della previsione: va considerato che l'accuratezza delle stime tende in genere a diminuire con l'aumentare del periodo richiesto dall'analista alla rete.

4.2 Costruzione dell'archivio dati

Consiste nella costruzione del set di informazioni che verranno utilizzate per riconoscere l'eventuale evoluzione delle variabili di output. Le fasi rilevanti sono:

- la raccolta;
- l'analisi e la trasformazione;
- la selezione delle variabili di input e di output.

4.2.1 Raccolta

In primo luogo è importante che le informazioni siano recuperate dai mercati con regolare frequenza di rilevazione.

In secondo luogo, da un punto di vista contenutistico, la rete dovrebbe essere teoricamente messa nelle medesime condizioni conoscitive dell'analista di mercato. Questo significa che i fattori che determinano le scelte sui mercati devono poter essere riconosciute dalla rete: risultano a tal fine rilevanti, quindi, sia i modelli teorici sia le informazioni che condizionano le aspettative economiche dei mercati.

La rete neurale per la previsione delle serie storiche finanziarie viene impostata utilizzando diverse tipologie di informazioni:

- *market* data: informazioni direttamente legate alle variabili di output;
- *intermarket* data: informazioni operativamente collegate con la variabile di output;
- *fundamental* data: informazioni dipendenti dalle componenti macroeconomiche fondamentali.

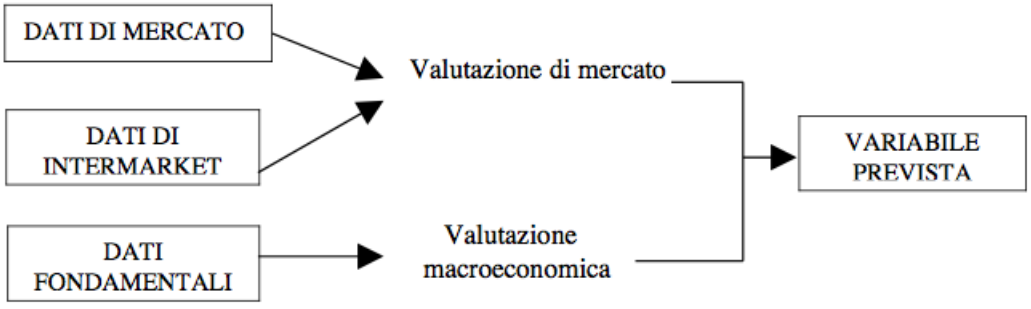

Figura 4.1: Variabili di input delle reti neurali per la previsione finanziaria

4.2.2 Analisi e trasformazione

Le fasi più delicate sono senza dubbio l'analisi e la trasformazione delle variabili di input e la preparazione dei dati, sia di input che di output.

Innanzitutto, per poter avvalersi del contributo informativo di determinati tipi di dati, caratterizzati da elevata rumorosità, è necessario l'utilizzo di trasformazioni che non alterino la dinamica del fenomeno (ad esempio, trasformazioni logaritmiche e regressioni statistiche). È il caso tipico di alcuni fenomeni finanziari caratterizzati da *mean reversion*, cioè dalla tendenza a non mantenere determinati trend crescenti o decrescenti per un prolungato periodo di tempo. Si pensi ai tassi di interesse e ai tassi di cambio, i quali tendono, nel medio termine a mantenersi sufficientemente stabili intorno a un valore medio di equilibrio. Livelli molto elevati dei tassi rispetto alla media di periodo riflettono una tendenza successiva a ritornare verso la media.

Ulteriori particolarità sono date dai valori eccezionali (*outlier*): se si desidera istruire la rete a riconoscere il pattern della serie storica nelle fasi "normali" è necessario ricorrere a opportune trasformazioni; se, al contrario, si vuole istruire la rete a riconoscere shock analoghi futuri non è necessaria alcuna trasformazione e la serie storica può essere lasciata inalterata.

Una soluzione spesso utilizzata passa per la *normalizzazione* della serie storica secondo varie tecniche.

Trasformazione per serie storiche con *mean reversion*. Una prima tecnica si realizza sottraendo al dato effettivo quello corrispondente della media mobile (Tab.4.1). In tal modo si elimina il trend e si riporta tutto su un piano che risulta maggiormente stabile e più prevedibile.

Tabella 4.1: Trasformazione per le serie storiche con *mean reversion*

Periodo	Dato originale	Media mobile (3 giorni)	Dato trasformato
1	25.3	—	—
2	22.6	23.20	-0.60
3	21.7	20.63	1.07
4	17.6	20.90	-3.30
5	23.4	20.77	2.63
6	21.3	24.40	-3.10
7	28.5	25.40	3.10
8	26.4	29.10	-2.70
9	32.4	32.10	0.30
10	37.5	—	—

Trasformazione con minimo e massimo prefissato. Un secondo metodo per la normalizzazione è quello di rendere la serie compresa fra un valore minimo e un valore massimo. Diverse possibilità si offrono, generalmente legate alla distanza dell'osservazione rispetto ai valori minimi e massimi. Se interessa far variare la serie normalizzata nell'intervallo

$[0, 1]$ la formula da adottare è la seguente:

$$v'_i = \frac{v_i - v_{\min}}{v_{\max} - v_{\min}} \tag{4.1}$$

dove $v_{\min}$ e $v_{\max}$ sono il valore minimo e quello massimo della serie.

La formula generale che permette di far variare la serie in un intervallo $[L_{\min}, L_{\max}]$ +è la seguente:

$$v'_i = L_{\min} + [L_{\max} - L_{\min}] \cdot \frac{v_i - v_{\min}}{v_{\max} - v_{\min}} \tag{4.2}$$

Si ipotizzi una serie storica di cui si conosca il valore massimo $v_{\max}$ e il valore minimo $v_{\min}$. Si consideri che è poi possibile fissare un limite oltre il quale il dato è considerato *outlier*: sia $L_{\max}$ il limite massimo e $L_{\min}$ il limite minimo. La Tabella 4.2 mostra come una serie storica possa essere normalizzata.

Tabella 4.2: Normalizzazione con minimo e massimo prefissato

v_i	$v_{\min}$	$v_{\max}$	$L_{\min}$	$L_{\max}$	v'_i
456	132	765	200	500	353.6
342	132	765	200	500	299.5
365	132	765	200	500	310.4
765	132	765	200	500	500.0
354	132	765	200	500	305.2
132	132	765	200	500	200.0
367	132	765	200	500	311.4
398	132	765	200	500	326.1
421	132	765	200	500	337.0

Normalizzazione con media e deviazione standard. Un altro metodo utilizzato per la normalizzazione della serie è quello che si basa sulla media e sulla deviazione standard della serie storica stessa. La formula della normalizzazione statistica è la seguente:

$$v'_i = \frac{v_i - \mu(v)}{\sigma(v)} \tag{4.3}$$

dove $\mu(v)$ è la media della serie storica analizzata e $\sigma(v)$ è la sua deviazione standard.

La serie storica dei dati v'_i trasformati in questo modo presenta la caratteristica di avere media nulla e deviazione standard unitaria (Tab. 4.3).

Queste trasformazioni sono ugualmente valide nel caso si voglia analizzare un fenomeno caratterizzato da elevata stagionalità (si pensi, in particolare, al prezzo delle *commodity* e dei relativi contratti derivati. In tal caso la media mobile deve avere la frequenza della stagionalità.

Tabella 4.3: Standardizzazione statistica

v_i	$\mu(v)$	$\sigma(v)$	v_i'	$\mu(v')$	$\sigma(v')$
456	400	164.48	0.34	0	1
342	400	164.48	-0.35	0	1
365	400	164.48	-0.21	0	1
765	400	164.48	2.22	0	1
354	400	164.48	-0.28	0	1
132	400	164.48	-1.63	0	1
367	400	164.48	-0.20	0	1
398	400	164.48	-0.01	0	1
421	400	164.48	0.13	0	1

Tabella 4.4: Assegnazione valori per il riconoscimento della stagionalità

Lunedì	(1,0,0,0,0)
Martedì	(1,1,0,0,0)
Mercoledì	(1,1,1,0,0)
Giovedì	(1,1,1,1,0)
Venerdì	(1,1,1,1,1)

In alternativa è possibile assegnare ai singoli giorni della settimana (Tab. 4.4), ai mesi o alle stagioni dei punteggi differenti, in modo da consentire alla rete di riconoscere l'effetto che si produce sulla variabile analizzata al trascorrere del tempo.

Quanto la normalizzazione condizioni l'efficacia dell'apprendimento della rete neurale è argomento abbastanza discusso: in generale si ritiene che il beneficio sia rilevante nella fase di classificazione e in termini di errore medio quadratico, mentre incerto è l'effetto quando la dimensione del dataset aumenta; inoltre, si registra un significativo rallentamento del processo di training.

Ulteriori particolarità delle serie storiche possono essere date dai valori eccezionali (*outlier*): il trattamento di questi dati può avvenire in due modi. Se si vuole istruire la rete per consentire un riconoscimento degli shock analoghi futuri è necessario lasciare la serie storica inalterata, sebbene il rischio sia quello di forzare l'intero modello all'adattamento su questi valori. Se, al contrario, si desidera stimare i parametri di una rete neurale maggiormente adatta a riconoscere il pattern della serie storica nelle fasi "normali", si può trasformare la dinamica intorno alla media del fenomeno applicando ai dati una distribuzione normale. A questo proposito si rammenta la forma generale della distribuzione normale, che consente di tagliare le osservazioni che caratterizzano le code (Tab. 4.5).

Tabella 4.5: Forma generale della distribuzione normale

$n\sigma$	Area della distribuzione
0.67	50%
1.00	68%
1.96	95%
2.58	99%

Riesaminando l'esempio della Tabella 4.3, i valori minimi e massimi vengono trasformati in relazione all'ampiezza della distribuzione normale prescelta (Tabella 4.6).

Tabella 4.6: Forma generale della distribuzione normale

$n\sigma$	**minimo**	**massimo**
Dati effettivi	365.00	765.00
0.67	289.80	510.20
1.00	235.52	564.48
1.96	77.62	722.38
2.58	-24.35	824.35

È ovvio che l'incidenza degli *outlier* deve essere minima, e il limite di variazione sarà pertanto condizionato dalla volatilità del fenomeno. Un sistema spesso applicato dagli analisti è

quello che accetta valori minimi e massimi collegati ad un predefinito coefficiente di deviazione standard. L'ipotesi di normalità è spesso solo un'approssimazione della reale distribuzione del fenomeno: in particolare, se la numerosità della serie è ridotta si suggerisce di valutare direttamente la numerosità e la significatività dei fenomeni che vengono scartati mediante la deviazione standard.

In generale, questa fase deve essere ispirata dalla finalità di ridurre il "rumore" della serie mediante l'utilizzo di trasformazioni che non alterino la dinamica del fenomeno. Fra quelle più diffuse citiamo:

- trasformazioni logaritmiche;
- analisi di Fourier;
- regressioni statistiche.

4.2.3 Selezione delle variabili di input e di output

Fondamentale è una continua calibrazione del database, con l'obiettivo di eliminare i dati che ex-post risultano non significativi in relazione al contenuto informativo della rete. In termini operativi, la procedura dovrebbe articolarsi nelle seguenti fasi:

1. definizione ampia del primo database;
2. primo apprendimento della rete;
3. valutazione dell'apporto informativo delle singole variabili;
4. analisi della matrice di correlazione fra le variabili di input;
5. eliminazione delle variabili meno significative;
6. successivo apprendimento della rete con il database ridotto.

Un processo così strutturato tende all'individuazione dell'archivio ottimale in funzione del problema analizzato, tramite uno schema iterativo.

4.3 Apprendimento

La costruzione della rete prevede necessariamente alcune fasi che consentono di fissare i parametri utili per l'apprendimento idoneo alla soluzione del problema.

La scelta dell'architettura più adatta per l'apprendimento e del meccanismo di connessione degli input tra loro e fra questi e l'output, passando per gli strati nascosti, è un elemento decisivo per il successo dell'operazione.

I parametri da determinare nella definizione dell'architettura sono:

Suddivisione temporale del database. Definita la forma dell'output e il contenuto del data set da cui estrarre le variabili di input, è necessario suddividere la serie storica in sottoperiodi, i quali determinano l'ambito di apprendimento (*training set*) e di valutazione. Quest'ultimo viene a sua volta distinto in *test set* e *generalisation set*.

In sostanza, la rete impara cercando di riconoscere la dinamica del *training set*, verifica come si adatta sul *test set* e poi si applica a un insieme di dati (*generalisation*) che non ha mai potuto osservare.

Non esistono regole universalmente valide per la suddivisione della serie storica da analizzare: le soluzioni più adottate sono $(60\% - 20\% - 20\%)$ e $(60\% - 30\% - 10\%)$ rispettivamente per il *training set*, il *test set* e il *generalisation set*.

Numero degli strati nascosti e dei neuroni da inserire in ciascuno strato. Numerosi sono i lavori empirici che utilizzano un solo strato nascosto in quanto è sufficiente ad approssimare funzioni non lineari con elevato grado di accuratezza. Tuttavia, questo approccio richiede un elevato numero di neuroni, andando a limitare il processo di apprendimento. Risulta, quindi, essere più efficace l'utilizzo di reti con due strati nascosti, soprattutto per previsioni su dati ad alta frequenza.

Questa scelta, oltre ad essere suggerita da un'apposita teoria, è supportata dall'esperienza, la quale mostra come, d'altro canto, un numero di strati nascosti superiore a due non produce miglioramenti nei risultati ottenuti dalla rete.

In riferimento alla quantità dei neuroni, va notato che un numero eccessivo di neuroni può generare *overlearning*, cioè i neuroni non sono in grado di generare una previsione affidabile perché riducono il contributo degli input; in questi casi, è come se la rete avesse "imparato a memoria" le risposte corrette, senza essere in grado di generalizzare. Al contrario, un numero troppo basso di neuroni riduce il potenziale di apprendimento della rete.

Occorre trovare una soluzione di *trade off* fra un numero troppo basso o troppo elevato di neuroni. Le formule proposte in letteratura sono assai varie e, in alcuni casi contraddittorie:

$$h = 2 \cdot n + 1 \tag{4.4}$$

$$h = 2 \cdot n \tag{4.5}$$

$$h = n \tag{4.6}$$

$$h = \frac{n + m}{2} + \sqrt{t} \tag{4.7}$$

dove:

- h è il numero di neuroni nascosti;
- n è il numero di neuroni di input;
- m è il numero di neuroni di output;
- t è il numero di osservazioni contenute nel training set.

I risultati empirici dimostrano come nessuna di queste regole risulti generalizzabile ad

ogni problema previsionale, sebbene un numero non irrilevante di risultati positivi sembra preferire la (4.6).

Meccanismi connettivi tra i differenti strati. Esistono diverse modalità di connessione:

1. *connessioni standard*: prevedono connessioni dirette, senza ritorni su se stessi, fra input e output che passano attraverso uno o più strati nascosti (Fig.4.2);

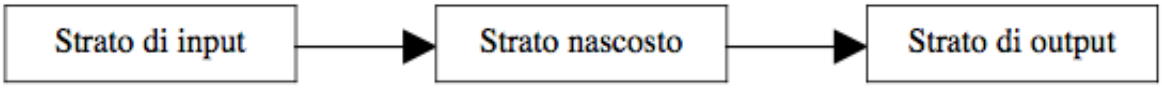

Figura 4.2: Architettura backpropagation a uno strato nascosto con connessioni standard

2. *connessioni a salti*: prevedono che la rete assegni dei pesi connettivi anche fra neuroni presenti in strati non adiacenti (Fig.4.3);

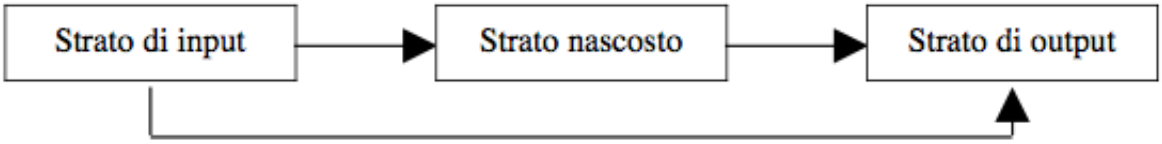

Figura 4.3: Architettura backpropagation a uno strato nascosto con connessioni a salto

3. *connessioni ripetute*: prevedono la possibilità che i neuroni assegnati agli strati nascosti possano ritornare sulle variabili di input con processi iterativi così da quantificare in modo preciso il peso connettivo (Fig.4.4). Risultano, perciò, particolarmente

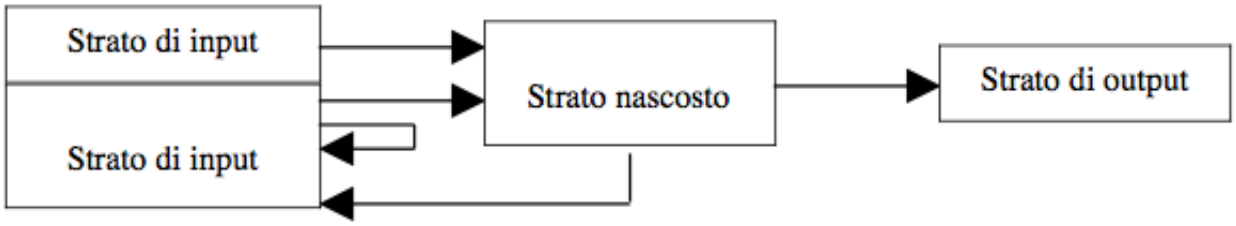

Figura 4.4: Architettura backpropagation a uno strato nascosto con connessioni ripetute

utili per l'analisi delle serie storiche finanziarie in quanto riconoscono le sequenze che si creano nell'ambito dei mercati.

Funzione di attivazione. Si distinguono otto tipi di diverse forme funzionali (vedi Tab. 4.7) che condizionano il legame dei neuroni (lineare, sinusoidale, gaussiana, etc.). È possibile definire una funzione di attivazione diversa per ogni strato.

Non esiste una regola teoricamente accettabile per definire la funzione di attivazione dei vari strati. Sebbene molti studi presentino funzioni differenti per i singoli strati, alcuni adottano la stessa funzione per gli strati input, nascosto/i e output.

Tabella 4.7: Funzioni di attivazione delle reti neurali

Funzione	**Formula**
Lineare	$f(x) = x$
Logistica (sigmoidea)	$f(x) = \frac{1}{1+\mathrm{e}^{-x}}$
Logistica simmetrica	$f(x) = \left(\frac{2}{1+\mathrm{e}^{-x}}\right)^{-1}$
Tangente iperbolica	$f(x) = \frac{\mathrm{e}^{x}-\mathrm{e}^{-x}}{\mathrm{e}^{x}+\mathrm{e}^{-x}}$
Tangente corretta	$f(x) = \tanh(c \cdot x)$
Sinusoidale	$f(x) = \sin(x)$
Gaussiana	$f(x) = \mathrm{e}^{-x^2}$
Gaussiana inversa	$f(x) = 1 - \mathrm{e}^{-x^2}$

La *funzione lineare* viene in genere utilizzata per lo strato che contiene l'output della rete neurale: la ragione è che questa funzione, pur essendo più rigida delle alternative, evita che il risultato tenda verso il minimo o il massimo. Meno efficace è, invece, l'utilizzo della funzione lineare negli strati nascosti, soprattutto se questi sono caratterizzati da un elevato numero di neuroni, che risulterebbero così connessi proprio su una base funzionale che si vuole superare con l'utilizzo della rete stessa. Il limite rilevante della funzione lineare è quello di non consentire un fitting adeguato per serie storiche caratterizzate da trend persistente.

La *funzione logistica* e quella *logistica simmetrica* presentano la caratteristica di variare, rispettivamente, negli intervalli $(0; 1)$ e $(-1; 1)$. La prima risulta essere particolarmente utile negli strati nascosti delle reti applicate alle serie storiche finanziarie. Alcuni problemi presentano caratteristiche dinamiche che risultano essere colte in misura più precisa dalla funzione simmetrica, soprattutto nello strato di input e in quelli nascosti. La maggior parte della letteratura empirica presenta l'utilizzo di questa funzione nello strato nascosto, sebbene senza che vi sia una robusta motivazione teorica.

La funzione *tangente iperbolica* consente di adattare la rete in modo affidabile negli strati nascosti (in particolare nelle reti a tre strati), soprattutto nel caso in cui l'analista abbia scelto una funzione logistica o lineare per l'output.

La *funzione sinusoidale* viene generalmente adottata nell'ambito della ricerca e si suggerisce di normalizzare input e output nel range $(-1; +1)$.

La *funzione gaussiana* si presta all'individuazione di particolari processi dinamici colti con architetture a due strati nascosti paralleli, con una funzione tangente nel secondo strato.

Regole di apprendimento. Definite le caratteristiche iniziali della rete neurale, è necessario definire i criteri di arresto dell'apprendimento. Se si vuole costruire una rete neurale con finalità previsionali è preferibile valutarne l'apprendimento sul test set; se l'obiettivo è quello di descrivere adeguatamente il fenomeno studiato è preferibile valutare l'apprendimento sul training set.

Va aggiunto che i parametri di apprendimento sono legati a indicatori di errore commessi dalla rete (errore medio, errore massimo, numero di epoche senza miglioramento dell'errore).

Aggiornamento dei pesi di connessione dei neuroni. Un ulteriore problema consiste nella scelta della regola di apprendimento: in particolare è necessario decidere con quale tasso di cambiamento la rete deve modificare la definizione dei pesi dei neuroni rispetto alla significatività dell'errore commesso.

Si consideri la Fig. 4.5. A sinistra si può osservare un progresso sia di apprendimento che di oscillazione con una frequenza che raggiunge il risultato atteso più velocemente; a destra invece la rete apprende modificando il proprio "sentiero" con un tasso inferiore e raggiunge in un tempo superiore l'obiettivo.

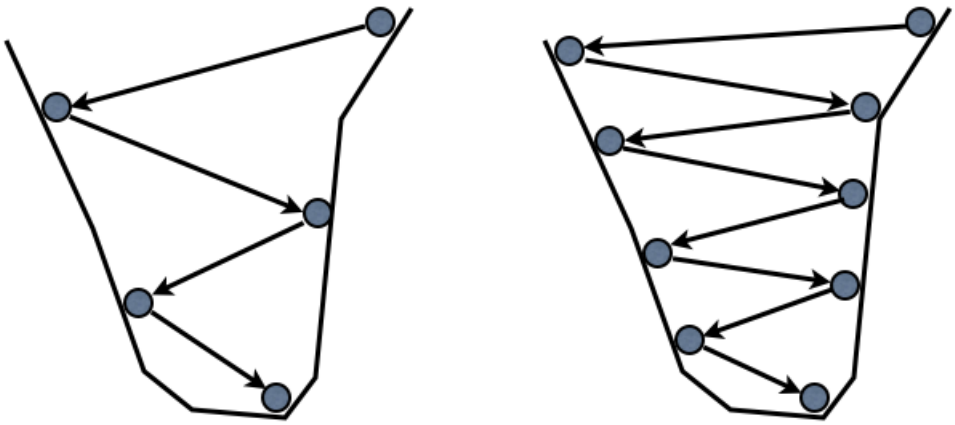

Figura 4.5: Sviluppo della rete con tassi di apprendimento differenti

Il tasso di apprendimento (o *learning rate*) η viene normalmente impostato inizialmente al valore:

$$\eta = \frac{\max(\mathbf{x}) - \min(\mathbf{x})}{N} \tag{4.8}$$

dove $\mathbf{x}$ rappresenta la serie di dati del training set. Mediante la compensazione dei valori di input $\mathbf{x}$ e del relativo numero N questo valore iniziale di η dovrebbe andare bene per molti problemi di classificazione, indipendentemente dal numero di campioni del training set e dell'intervallo dei suoi valori. Comunque, benché valori più alti di η potrebbero accelerare il processo di apprendimento, ciò potrebbe indurre oscillazioni che possono rallentare la convergenza.

In alternativa, è possibile fare uso del *momentum*, cioè della proporzione con cui si aggiunge al nuovo peso la variazione dell'ultimo peso raggiunto dalla rete neurale. In questo modo la rete può imparare anche a un tasso elevato ma non rischia di oscillare perché recupera un'elevata quota dell'ultimo peso raggiunto.

L'analista deve poi decidere il livello iniziale del peso da dare alla connessione fra neuroni

e valutare se le osservazioni sono caratterizzate da un elevato tasso di rumore. In questo caso sarà opportuno mantenere elevato il valore del *momentum*.

Il processo di apprendimento della rete passa naturalmente dall'individuazione progressiva del valore più adatto di questi parametri. Si impone pertanto una serie numerosa di tentativi che possono generare risultati sensibilmente differenti. Il suggerimento è quello di evitare radicali variazioni dei parametri.

4.4 Indicatori di errore

I parametri di apprendimento sono generalmente legati agli indicatori degli errori commessi dalla rete:

a) errore medio;

b) errore massimo;

c) numero di epoche senza miglioramento dell'errore.

Fissando un valore per questi parametri la rete si bloccherà una volta raggiunto il valore desiderato. In genere è più semplice fissare un elevato numero di epoche (che sono condizionate dalla numerosità delle osservazioni dei training set) che la rete analizza senza migliorare l'errore. Si può ritenere che un valore compreso tra 10.000 e 60.000 epoche sia sufficientemente sicuro per bloccare una rete la quale ormai con grande difficoltà può apprendere meglio di quanto abbia già fatto fino a quel momento. Naturalmente la scelta dipende anche dalla velocità con cui la rete raggiunge questi valori.

Un elemento per l'accettazione di una rete è la *convergenza*. Se anche gli errori risultano modesti ma l'oscillazione ha portato ad un'elevata divergenza, è opportuno verificare l'adeguatezza dei parametri (in particolare, tasso di apprendimento e momentum). La medesima considerazione deve esser fatta per quanto riguarda la distribuzione temporale degli errori commessi sul validation set.

Una volta verificata una corretta dinamica temporale dell'errore, almeno in termini grafici, è necessario misurarla quantitativamente. Tra i vari indicatori di errore sviluppati in ambito statistico spiccano:

a. l'indice di determinazione (R^2)

b. il Mean Absolute Error (MAE)

c. il Mean Absolute Percentage Error (MAPE)

d. il Mean Square Error (MSE)

e. il Root Mean Square Error (RMSE)

Si tratta di indicatori che misurano, in vario modo, il differenziale fra l'output originario e quello stimato dalla rete; solo se gli input, i neuroni, le funzioni di attivazione e tutti i parametri precedentemente descritti fossero perfettamente in grado di individuare il fenomeno originale gli scostamenti fra output reale e stimato sarebbero nulli, ottimizzando così i citati indicatori di errore.

Il limite di questi errori è quello di basarsi su un concetto di scostamento simmetrico rispetto al valore reale, mentre in finanza l'errore si misura solo in termini di perdita. Sarebbe pertanto opportuno stimare i pesi della rete sulla base dei profitti ottenuti; in termini operativi, si possono adottare delle strategie di filtro per rimuovere ex post il problema. Anche in relazione alla validazione della rete occorre adattarsi alle finalità dell'analista.

4.5 La previsione della serie storica

Una volta che la rete neurale è stata correttamente costruita, è necessario verificare la sua "bontà" previsionale. È infatti possibile che un modello riesca a descrivere ottimamente il training e il test set ma poi risulti del tutto inadeguato per quanto riguarda la sua generalizzazione, cioè — nel caso finanziario — la previsione.

L'analista dovrò pertanto testare sul generalisation set la rete neurale con le medesime tecniche già descritte. In primo luogo si dovranno misurare sulla serie storica mai osservata dalla rete gli indicatori di errore già descritti. Qualora questi dovessero risultare significativamente peggiori e, comunque, non accettabili dall'analista sulla base degli obiettivi originari, la rete dovrà essere ulteriormente testata.

La previsione finanziaria viene poi spesso condizionata dalla capacità del modello di individuare per tempo le inversioni cicliche del fenomeno. Questa proprietà può essere verificata anzitutto su base grafica (Fig. 4.6).

Inoltre, è possibile utilizzare l'output originale e quello stimato dalla rete per misurare quantitativamente il ritardo con cui la rete apprende l'inversione del ciclo. Numerosi sono gli strumenti utili per analizzare la capacità previsionale della rete neurale: a ogni inversione del ciclo la serie dei valori del fenomeno deve registrare una inversione del segno della variazione. Se, dunque, si passa da un trend crescente a uno decrescente, i segni delle variazioni passeranno da "+" a "−".

Per verificare con quale capacità la rete segnala questa inversione, un sistema efficace è quello che si basa sul calcolo dell'intervallo di confidenza dell'output della rete. La misurazione dell'intervallo di confidenza può essere fatta seguendo questo schema:

1. calcolo dello deviazione standard σ della serie di output prodotta dalla rete neurale;

2. definizione del limite di confidenza dell'analista: se questo è pari al 95%, in ipotesi di distribuzione normale, occorrerà calcolare 1.96 volte la deviazione standard;

3. misurazione dei valori estremi dell'intervallo: il valore massimo è dato dall'output cui si aggiunge $n\sigma$ in base all'intervallo scelto al punto precedente, mentre il valore minimo è dato dall'output cui si sottrae $n\sigma$;

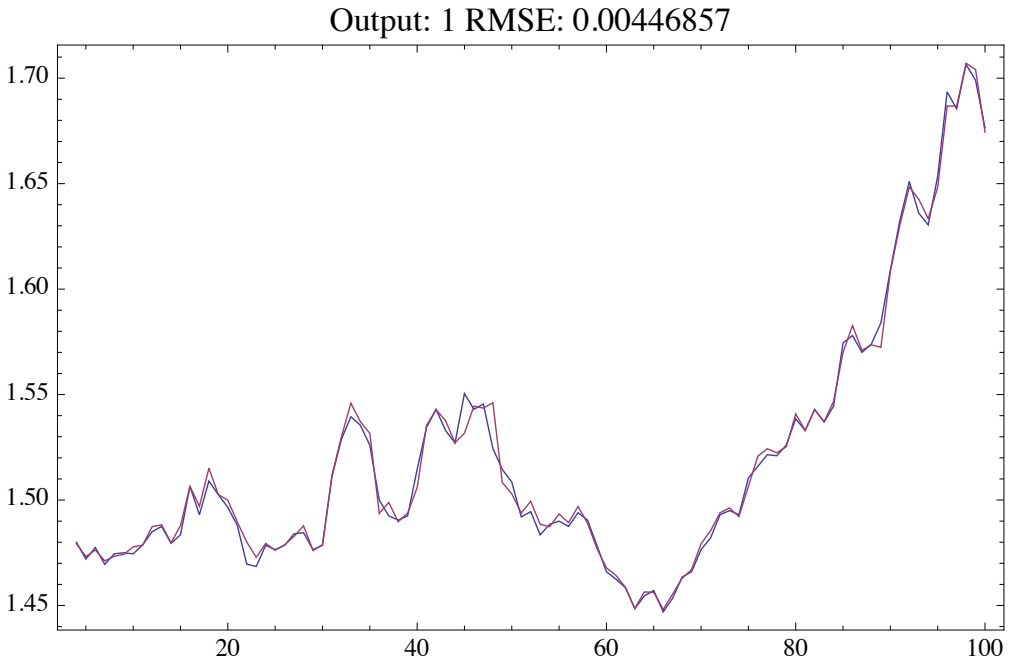

Figura 4.6: Output della rete neurale e confronto con i dati del generalisation set

4. si verifica in quali casi il segno del valore minimo e di quello massimo coincidono: fissato l'intervallo di confidenza, si può affermare che la probabilità di prevedere correttamente il trend della serie coincide con quella percentuale;

5. si controlla quante volte si commette realmente un errore di segnalazione, valutando in tal modo l'affidabilità dello strumento previsionale;

6. l'analista deciderà quale posizione di mercato prendere solo nei casi in cui valore minimo e massimo dell'intervallo hanno il medesimo segno: questo segno indicherà naturalmente la posizione coerente da prendere sul mercato;

7. nel caso in cui due o più osservazioni consecutive presentassero lo stesso segno, l'operatore avrebbe un'indicazione di mantenimento della posizione esistente.

Si consideri la Tabella 4.8, nella quale l'output della rete neurale è il valore del tasso di cambio lira/marco, per le varie giornate, con una deviazione standard pari a 4.53.

Si ipotizzi di voler realizzare un trading con il 70% di intervallo di confidenza: in questo caso vengono accettate solo le indicazioni caratterizzate da un minimo superiore o da un massimo inferiore al precedente. La prima indicazione di acquisto (Buy) del marco si ha quando il minimo previsto (1010.47) è superiore al dato precedente (1010.00); lo stesso dicasi per il secondo segnale di acquisto, con un minimo previsto di 1000.47 contro un precedente di 1000.00. Nel caso delle due indicazioni di vendita (Sell), la rete prevede rispettivamente un massimo di 1009.53 inferiore al dato precedente 1015.00 e un massimo di 1008.53 contro un precedente 1009.00.

Per l'utilizzo operativo di un trading system, l'analista può associare a ciascun segnale

Tabella 4.8: Indicazione di trading di una rete neurale

Output	σ**(output)**	**Max(output)**	**Min(output)**	**Trend**	**Trading**
1010.00	4.53	1014.53	1005.47	—	—
1015.00	4.53	1019.53	1010.47	↑	Buy
1005.00	4.53	1009.53	1000.47	↓	Sell
1002.00	4.53	1006.53	997.47	=	?
1000.00	4.53	1004.53	995.47	=	?
1005.00	4.53	1009.53	1000.47	↑	Buy
1007.00	4.53	1011.53	1002.47	=	?
1009.00	4.53	1013.53	1004.47	=	?
1004.00	4.53	1008.53	999.47	↓	Sell

di acquisto o vendita un determinato budget iniziale e confrontare sul generalisation set che risultato economico la rete avrebbe ottenuto se realmente si fossero seguite le indicazioni offerte.

Capitolo 5

Conclusioni

Le reti neurali artificiali, considerate nelle loro applicazioni economico-finanziarie, rappresentano il tentativo di individuare dinamiche di mercato che esistono ma che, per limiti di analisi, non si è ancora in grado di esplicitare in modelli strutturali. La differenza principale, infatti, fra il sistema di analisi matematica e il sistema finanziario è riconducibile alla forte componente dell'interazione sociale che è sempre più in grado di alterare l'eventuale equilibrio di mercato.

In ambito operativo, le reti neurali offrono un significativo e crescente contributo nella previsione in mercati caratterizzati da componenti di tipo caotico (ma comunque deterministico).

I risultati finanziari finora ottenuti sulla base di sistemi a reti neurali evidenziano un primato di tali modelli sulla previsione casuale dimostrando di poter fornire un reale vantaggio competitivo. Sotto un punto di vista tecnico, in questa sede sono emersi alcuni punti essenziali:

a) il modello che si impone come standard di mercato è quello della rete *feedforward* con apprendimento *backpropagation*: in primo luogo, infatti, l'algoritmo *backpropagation* è semplice, intuitivo e facilmente verificabile; in secondo luogo, esso è particolarmente adatto all'analisi di serie storiche;

b) i modelli utilizzati sono numerosi e disparati a dimostrazione del fatto che non esistono regole assolute per poter configurare in maniera ottimale la rete; esistono solo regole empiriche diverse da caso a caso;

c) conseguenza delle numerose diversità operative è la difficoltà nella confrontabilità tra gli studi di misurazione delle performance;

d) le reti neurali sono in grado di ricostruire la legge che descrive un dato fenomeno, riconoscendo in esso forme di regolarità e di struttura, e in seguito di fare previsioni. Tuttavia, risultano incapaci di fornire spiegazioni riguardo ai risultati raggiunti.

Va ricordato che la progettazione di tali reti si ottiene attraverso un procedimento che comporta molte prove e talvolta anche errori. Risulta, quindi, necessario proseguire nel campo della ricerca, affinché questa tecnologia sia supportata da una metodologia collaudata anche dal punto di vista rigorosamente scientifico e non solo dall'esperienza.

Indice analitico