

55290
653910k

\\ 340 \\

La costruzione di modelli con algoritmi genetici

di

Tommaso Minerva

Ottobre 2000

Università degli Studi di Modena e Reggio Emilia
Dipartimento di Economia Politica
Via Berengario, 51
41100 Modena (Italia)
e-mail: minerva@unimo.it



Copia n. 653910

CLL.088.340

1 * costruzione di modelli con alg

Hanno collaborato a diverse fasi del progetto e alla realizzazione del software, e continuano a collaborare, Marcello Galli e Sandra Paterlini dell' Università di Modena e Stefano Favaro dell' Università di Venezia (Sede di Treviso). La prof.ssa Irene Poli e' stata continua ispiratrice e guida.

T. Minerva, S. Paterlini and I. Poli, 1999

"Algoritmi Ibridi per l' Analisi di Serie Storiche Finanziarie. Integrazione di Reti Neurali ed Algoritmi Genetici per la Previsione dell' Andamento del Contratto Future FIB30"

Scienza e Business 3-4, 1999.

T. Minerva and I. Poli, 2000

"Predicting Venice High Tides by using Radial Basis Networks and Genetic Algorithms"

accettato per la pubblicazione sul volume a cura di S. Borra, R. Rocci, M. Vichi e M. Schader della collana "Studies in Classification, Data Analysis and Knowledge Organization" edito dalla Springer-Verlag, Berlin, 2000.

T. Minerva, I. Poli and S. Paterlini, 2000

"A Genetic Algorithm for Neural Network Design (GANND) - A Financial Application"

in pubblicazione su Economics & Complexity, 2000.

T. Minerva, I. Poli, 2000

Time Series Modelling by Genetic Evolution

In preparazione

INDICE

INTRODUZIONE.....	3
1) TECNICHE DI SELEZIONE DI UN MODELLO STATISTICO.....	7
2) IL PROBLEMA DELL'OTTIMIZZAZIONE.....	21
3) TECNICHE COMPUTAZIONALI.....	36
4) SELEZIONE DI UN MODELLO LINEARE.....	43
4.1. IL MODELLO GENETICAMENTE EVOLUTO.....	51
4.2. IL MODELLO SFUMATO.....	61
4.3. CONCLUSIONI.....	70
5) SELEZIONE DEL MODELLO PER SERIE STORICHE.....	72
5.1. IL PROGRAMMA GATS.....	77
5.2. CONCLUSIONI.....	96
6) MODELLI NON LINEARI APPROSSIMATI: RETI NEURALI.....	97
6.1. IL NEURONE ARTIFICIALE E I SUOI COMPONENTI.....	103
6.2. PRINCIPALI TIPI DI RETE.....	106
6.3. IL PROGRAMMA GANND.....	111

7) OTTIMIZZAZIONE GENETICA.....	125
7.1. LA FUNZIONE DI ROSENBROCK.....	128
7.2. SIMULAZIONE DI RETI NEURALI.....	144
7.3. CONCLUSIONI.....	150
8) APPLICAZIONE DELLE TECNICHE DI MODELLIZZAZIONE GENETICA...151	
8.1. IL MODELLO PREVISIONALE DELLE MAREE DI VENEZIA.....	152
8.2. ANALISI SUL LIVELLO DI DISOCCUPAZIONE IN EMILIA ROMAGNA.....	171
CONCLUSIONI.....	181
BIBLIOGRAFIA	

INTRODUZIONE

Da alcuni anni si assiste a un utilizzo sempre più massiccio di metodologie software intelligenti in campo medico, finanziario, economico e statistico. Nell'accezione più ampia del termine, l'intelligenza artificiale si occupa della costruzione di algoritmi in grado di svolgere compiti assimilabili ai processi decisionali umani.

Normalmente un sistema automatizzato si compone di una macchina in grado di effettuare un repertorio prefissato di azioni, dove ciascuna azione corrisponde allo svolgimento di un compito ben definito. L'automazione industriale, per esempio, riguarda la progettazione, lo sviluppo e la produzione di beni materiali. L'obiettivo di un approccio automatizzato è specificatamente quello di aumentare l'efficienza, migliorare la qualità e ridurre il tempo richiesto per apportare cambiamenti [Shap90].

In termini più generali, la caratteristica più saliente di un simile sistema è quella di aver un comportamento prefissato tale per cui ciascuna delle sue risposte agli stimoli esterni è stata programmata in anticipo da un progettista di sistemi.

Ovviamente, un simile sistema presuppone l'esistenza di una risposta ideale a un insieme limitato di stimoli esterni.

Un sistema adattivo, d'altro canto, si fonda sul tentativo di limitare la quantità di conoscenza a priori necessaria alla definizione del comportamento del sistema. Al contrario, il sistema impara o adatta la sua risposta agli eventi misurati nel mondo

esterno secondo la propria esperienza, seguendo le linee guida di un obiettivo generale. Per esempio, si può addestrare una rete neurale a riconoscere le condizioni alle quali è conveniente acquistare o vendere un particolare pacchetto azionario. In questo caso, il progettista del sistema specifica l'intenzione di conseguire un profitto ma non fornisce le istruzioni esatte attraverso le quali il sistema potrà conseguire l'obiettivo in oggetto.

Esistono tre fattori che rendono un approccio di questo tipo particolarmente vantaggioso. In primo luogo, se il sistema può imparare dall'esperienza passata non è necessario che il progettista specifichi tutte le condizioni operative sotto le quali il sistema dovrà necessariamente funzionare (circostanza parecchio utile quando tali condizioni non sono note o comunque sono mal definite). In secondo luogo, emergono caratteristiche di flessibilità in relazione ai sistemi sviluppati per cui, in ambienti mutevoli, un sistema sarà in grado di adattarsi a nuove circostanze e regolare le sue risposte coerentemente con i cambiamenti, evitando la necessità di costose operazioni di riprogrammazione. In terzo luogo, si rendono possibili opportunità di innovazione. Un sistema in grado di apprendere dovrebbe sempre avere la capacità di pervenire a nuove scoperte. Un sistema lasciato libero di effettuare le correlazioni tra gli eventi che gli vengono sottoposti, dovrebbe essere in grado di individuare le relazioni finora sconosciute. In effetti è quest'ultima possibilità che determina il vantaggio più cospicuo offerto da un sistema adattivo rispetto agli approcci automatizzati più grandi.

In questa dispensa vengono illustrate alcune tecniche automatizzate per il riconoscimento di modelli statistici e, in questo ambito, per l'ottimizzazione delle funzioni utilizzate nella stima dei parametri.

In particolare si cerca di fornire una risposta alla problematica della selezione del "miglior" modello statistico da utilizzare in una indagine previsionale sia in ambito lineare che in ambito non lineare.

La selezione di un modello segue solitamente tre fasi:

1. Selezione delle variabili.
2. Individuazione della relazione funzionale tra le variabili e dei parametri a questa associati.

3. Stima del valore dei parametri.

La selezione delle variabili procede, solitamente, mediante tecniche step-by-step in cui di volta in volta viene introdotta o eliminata una variabile alla volta valutandone, mediante criteri opportuni, la capacità esplicativa. Per quanto riguarda la individuazione della relazione funzionale si distingue solitamente tra due gruppi: modelli lineari e modelli non lineari. Nel caso di modelli lineari la stima del valore dei parametri segue tecniche analitiche mentre nel caso di modelli non lineari la stima dei parametri viene di solito effettuata mediante tecniche numeriche di ottimizzazione la cui robustezza viene sottoposta a severa critica soprattutto nel caso di modelli non lineari molto complessi. Per quanto riguarda la individuazione della forma funzionale del modello (lineare, non lineare) di solito ci si lascia guidare da valutazioni sul fenomeno sotto esame oppure, in situazioni ad elevata complessità, si ricorre sempre più spesso a modelli non lineari approssimati (funzioni lineari a tratti, polinomi definiti a tratti, reti neurali) o nel caso di serie temporali a modelli lineari in cui la componente non lineare viene introdotta come un termine perturbativo (modelli ARMA, ARCH-GARCH, etc...).

In questa dispensa, utilizzando tecniche numeriche proprie dell'Intelligenza Artificiale (algoritmi evolutivi, adattivi, decisionali) ed in particolare utilizzando algoritmi ibridi (integrazione tra Algoritmi Genetici, Logica Fuzzy e Reti Neurali) vengono affrontate le seguenti problematiche:

1. Come scegliere le variabili da utilizzare nell'ambito di un modello lineare multivariato.
2. Come selezionare i regressori nell'ambito di un modello AR ed ARMA.
3. Come costruire un modello non lineare approssimato.
4. Come stimare i parametri di un modello non lineare con tecniche numeriche ottimizzate.
5. Quale criterio utilizzare per confrontare la bontà di due o più modelli.

Per le prime 4 sono state proposte soluzioni basate sugli Algoritmi Genetici mentre per la quinta si è proposto un criterio di valutazione basato sulla Logica Fuzzy.

Le tecniche sono valutate su insiemi di dati facilmente reperibili all'interno dei più accreditati pacchetti di sviluppo software o su dati simulati di cui viene fornito l'algoritmo di simulazione utilizzato. Un attento confronto con le tecniche standard viene condotto nei casi più significativi evidenziando un significativo aumento delle performances predittive con l'utilizzo delle tecnologie descritte in questa dispensa.

Infine si è proceduto a illustrare l'applicazione degli algoritmi a due casi reali: Previsioni del Livello di Marea della Laguna Veneziana [MiPo00] e Analisi della Disoccupazione in Emilia Romagna [LaMi00] evidenziando notevoli miglioramenti rispetto ad analoghe indagini condotte con tecniche classiche.

La dispensa comprende 8 capitoli organizzati nel seguente modo: nel primo capitolo vengono presentate le problematiche relative alle tecniche classiche di selezione di un modello statistico; nel secondo vengono presentate le problematiche e le tecniche di ottimizzazione di funzioni; nel terzo vengono descritte le caratteristiche delle principali tecniche computazionali evolute; nel quarto viene illustrato il programma di selezione di un modello lineare in ambiente multivariato, nel quinto capitolo viene riportato il programma di selezione di un modello per serie storiche; nel sesto il programma per l'analisi di modelli neurali; nel settimo capitolo il programma di ottimizzazione genetica di funzioni; ed infine nell'ultimo capitolo vengono presentate le analisi applicate ai due casi reali sopra citati.

1. TECNICHE DI SELEZIONE DI UN MODELLO STATISTICO

La fase più operativa della statistica è diretta alla costruzione di un modello per descrivere, comprendere, prevedere, simulare e controllare un fenomeno reale.

Per tali finalità diviene centrale la struttura logica e formale del modello di regressione mediante il quale si esplicita una relazione funzionale tra ciò che si intende spiegare (l'effetto, la risposta, il risultato) e quello che può esserne causa.

Un modello statistico è una rappresentazione semplificata (nessuna ipotesi, infatti, deve essere necessariamente complessa quando un'altra più semplice conduce alla stessa capacità esplicativa) analogica (in quanto una sorta di riproduzione della realtà che ne emula gli aspetti fondamentali al fine di ricavare induzioni e deduzioni utili) e necessaria (perché consente di esaminare la complessità del mondo mediante l'analisi di relazioni semplici e di maggiore intellegibilità) della realtà derivata da osservazioni sperimentali oltre che da deduzioni logiche.

L'aspetto dialettico nella costruzione di un modello statistico deriva dalle opposte esigenze di semplificare la struttura senza perdere in fedeltà, e tale conflitto è ineliminabile. Infatti, tutti i modelli sono intrinsecamente sbagliati: essi sono parzialmente e provvisoriamente utili, e sono destinati a essere sostituiti con l'avanzare del progresso scientifico e l'affinamento della conoscenza. Ciò che realmente conta non è la validità ontologica delle relazioni accertate ma l'efficacia

comparata in rapporto agli obiettivi. È l'obiettivo, infatti, che rende utile, efficace e temporaneamente valido il modello.

La costruzione di un modello si concretizza attraverso fasi successive: specificazione, stima e verifica del modello statistico. Non è un caso che la specificazione sia l'aspetto più delicato e importante dell'intera procedura; dalla sua correttezza, infatti, dipendono la validità e l'efficacia di tutte le fasi successive.

L'intero procedimento è rappresentato schematicamente in figura 1.1.

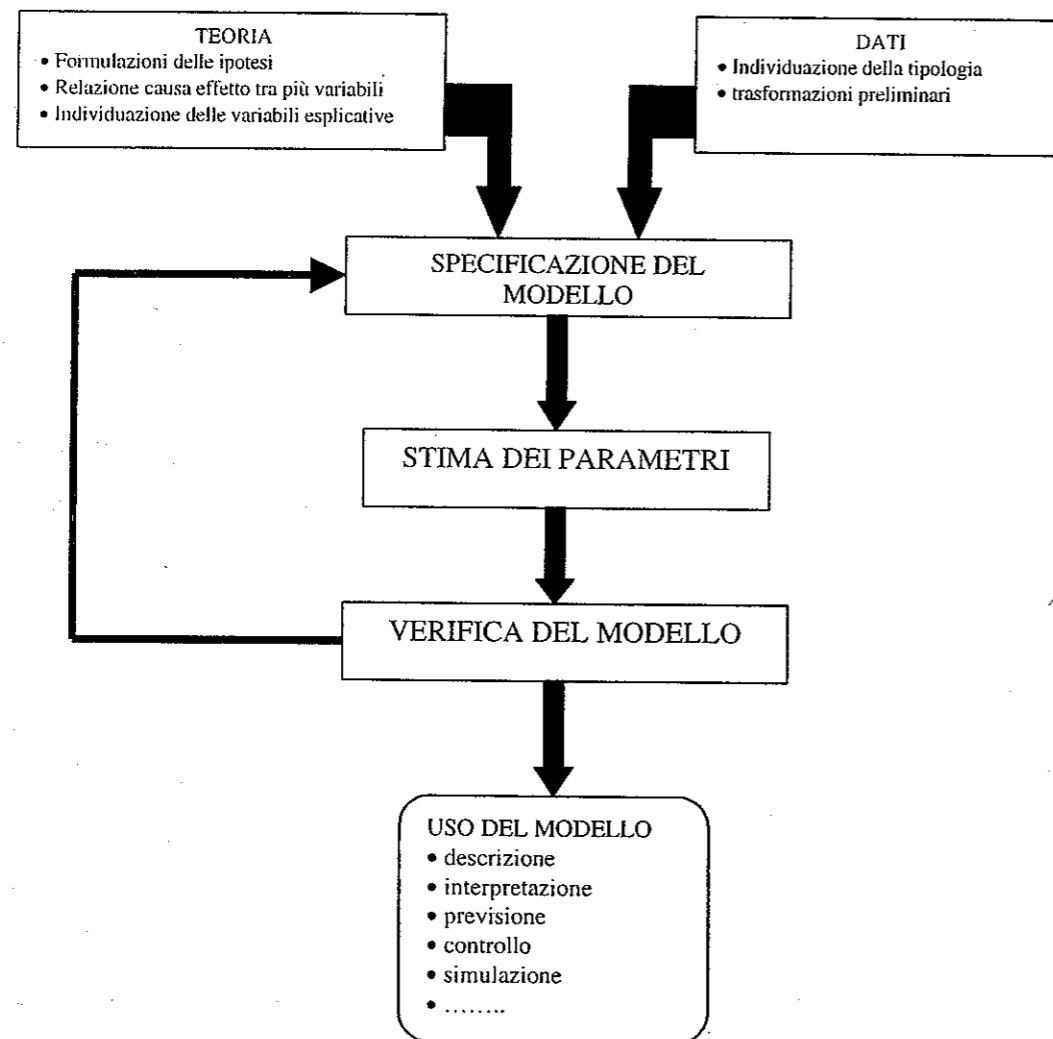


Fig. 1.1 : Fasi per la costruzione di un modello statistico.

La specificazione di un modello statistico inizia con l'esplicitare un legame tra i fenomeni di interesse nel modo seguente:

$$(1.1) \quad Y=f(X_1, X_2, \dots, X_p)$$

dove Y è la variabile da spiegare (variabile dipendente) mentre X_1, X_2, \dots, X_p (variabili esplicative) sono variabili prescelte per spiegare Y , grazie alla funzione f .

Tale relazione deriva dalla interazione tra conoscenze a priori e risultati sperimentali, e poiché non esistono settori scientifici ove è lecito ipotizzare un legame di natura strettamente deterministica, a tale relazione dovrà essere associata una componente stocastica. Pertanto (nell'ipotesi di componente stocastica additiva) la specificazione di un modello statistico assume la forma:

$$(1.2) \quad Y=f(X_1, X_2, \dots, X_p) + \varepsilon$$

dove ε è una variabile casuale ed è necessario fare opportune ipotesi sulla sua distribuzione.

Attente procedure di specificazione del modello sono essenziali per un corretto impiego delle informazioni empiriche disponibili. La ricerca di specificazione del modello si articola in diverse fasi: scelta della forma funzionale (lineare o non lineare), scelta delle variabili (lista delle variabili d'interesse e distinzione fra variabili endogene e esogene) e individuazione del corretto numero di variabili.

Tale ricerca solleva importanti problemi circa la scelta delle variabili esplicative per un modello (problema dell'individuazione del corretto modello statistico). Diviene fondamentale, infatti, stabilire univocamente quali variabili sono significative e quali ridondanti per spiegare il fenomeno Y , all'interno del quadro teorico accettato dagli studiosi di quel determinato settore.

Caratteristica comune alle varie tecniche utilizzate in questo tipo di ricerca è la definizione aprioristica dei tre seguenti elementi:

- 1) definizione del modo per muoversi nello spazio dei modelli;
- 2) specificazione di un criterio di valutazione del modello;

3) modalità del termine della ricerca;

Negli anni si sono quindi affermate diverse strategie, ritenute più o meno valide dai vari autori, per effettuare la scelta del modello "migliore":

- a) Tutte possibili ("All possible")
- b) Backward elimination
- c) Forward selection
- d) Stepwise regression
- e) Stagewise regression

Tutte possibili ("All Possible"): questa procedura è piuttosto lenta e complessa (rendendo necessario l'utilizzo di potenti calcolatori) in quanto consiste nella valutazione di ogni possibile regressione. Quindi se chiamiamo k il numero di variabili esplicative a disposizione e consideriamo il fatto che ogni variabile ha due alternative (appartenere o meno al gruppo di variabili esplicative) ci rendiamo conto del fatto che esistono $2^k - 1$ possibili soluzioni¹ (e quindi una crescita esponenziale del numero di regressioni all'aumentare del numero di possibili regressori).

Questa procedura si articola in quattro fasi:

- 1) Si effettua ogni possibile regressione misurando per ciascun modello un indicatore di "bontà" (come l'indice R^2 , per esempio);
- 2) Si raggruppano le diverse regressioni all'interno di differenti sottoinsiemi a seconda del numero di variabili esplicative utilizzate per quel modello;
- 3) Si ordina in modo decrescente, all'interno di ogni gruppo, dalla regressione "migliore" alla "peggiore";
- 4) I modelli che occupano il primo posto di ogni gruppo vengono confrontati tra loro allo scopo di scegliere il modello che meglio si addice a spiegare quel determinato fenomeno. Bisogna ricordare infatti che alcuni indicatori, come per esempio l' R^2 , aumentano all'aumentare del numero di regressori utilizzati rendendo quindi fuorviante un'analisi comparata dei modelli avente un diverso numero di regressori. Un'analisi dettagliata dei principali indicatori e delle loro caratteristiche sarà esposta in seguito.

Questo metodo è stato ampiamente caldeggiato da Sen e Srivasta [SeSr90] ma anche altrettanto criticato da Draper e Smith [DrSm66].

Backward elimination method: questo tipo di procedura è uno sviluppo del metodo "all possible regression" in quanto permette di effettuare non tutte le regressioni possibili ma solo la "migliore" per ogni sottoinsieme.

Anche questa procedura si articola in quattro fasi:

- 1) Si effettua la regressione con l'intero set di variabili disponibili;
- 2) Viene calcolato per ciascun regressore il p-value, cioè la probabilità che quel determinato regressore sia non significativo (sulla distribuzione t);
- 3) Se esistono regressori con un p-value che superi una determinata soglia si elimina quello con probabilità maggiore di essere non significativo (quello con un p-value maggiore).
Se invece sono tutti significativi, allora abbiamo trovato il modello che stavamo cercando e non si passa alla fase successiva.
- 4) Si effettua una nuova regressione con le variabili rimaste e si ritorna al punto 2.

L'uso di questa procedura è stato fortemente auspicato da Draper e Smith [DrSm66] in quanto rispetto alla precedente riduce il tempo di computazione mantenendo comunque la sua efficienza.

Hendry, Pagan e Sargan [HPS84], ricercatori della London School of Economics e forti sostenitori di questo tipo di procedura, sottolineano l'importanza della specificazione dinamica nelle fasi di costruzione dei modelli empirici basati su serie storiche², superando, per questa via, la precedente convinzione che la teoria

¹ Escludendo il modello "vuoto" in cui non compaiono variabili esplicative.

² L'analisi statistica delle serie storiche viene spesso definita metodologia Box-Jenkins [BoJe76] dal nome dei due studiosi che, con la sistemazione metodologica di un vasto insieme di analisi statistiche scaturita nel loro famoso libro del 1970, ne hanno favorito lo sviluppo e la diffusione. L'utilità di presentare alcune applicazioni di modellizzazione delle serie storiche (modellizzazione talvolta definita ARMA dalle iniziali di AR, *autoregressive* e MA, *moving average*) è legata a tre fattori. L'analisi statistica delle serie storiche può essere, anzitutto, vista come una metodologia capace di modellizzazione senza teoria economica: quando al ricercatore interessa ottenere una previsione di breve periodo di una particolare variabile economica non

economica sia da sola capace di condurre il ricercatore alla corretta specificazione del modello. In tal modo viene spostata parte dell'attenzione del ricercatore dalla teoria economica alle informazioni ottenibili dalla struttura e dalla natura dei dati a disposizione.

Graficamente la procedura è rappresentata in Fig. 1.2.

Forward selection method: il "backward elimination method" inizia dal modello più ampio, che utilizza tutte le variabili, per diminuirne il numero passo dopo passo finché non si raggiunge il modello "ottimale". Il "forward selection method" è il tentativo di raggiungere analoga conclusione partendo però dalla parte opposta. Consiste infatti nell'inserire variabili al modello a seconda del loro coefficiente parziale di correlazione. Pertanto :

- 1) Si seleziona la variabile maggiormente correlata con Y e si effettua la regressione.
- 2) Si calcola il coefficiente di correlazione parziale tra le altre variabili e la Y (equivalente a calcolare la correlazione tra i residui della regressione effettuata al punto precedente e i residui della regressione della variabile in questione sulle variabili esplicative preselezionate). A questo punto si seleziona la variabile con più alto coefficiente di correlazione e si effettua una nuova regressione di Y sulle variabili scelte.
- 3) Si ripete il punto precedente e per ogni regressione effettuata si calcola l' R^2 e il p-value di significatività dell'ultima variabile inserita. Quando la variabile inserita riduce la significatività dell'intero modello o risulta essere non significativa ci si ferma senza inserirla tra i regressori del modello.

facilmente modellizzabile dal punto di vista teorico o caratterizzata da penuria di informazioni sulle sue variabili esplicative, l'approccio B-J rappresenta sicuramente il più comodo (e spesso più affidabile) mezzo da intraprendere. Un secondo punto, strettamente collegato al precedente, è relativo all'utilizzazione delle previsioni "non condizionali" fornite dalla modellistica ARMA come *benchmark* per le previsioni dei modelli che incorporano teoria economica (modelli strutturali). Se la teoria economica impiegata dai modelli strutturali è vera, o più modestamente utile, questo più ampio insieme informativo deve consentire previsioni più accurate rispetto all'impiego della semplice modellistica ARMA. Se ciò non si verifica (i modelli ARMA prevedono meglio) evidentemente siamo in presenza di modelli econometrici strutturali caratterizzati da problemi di specificazione.

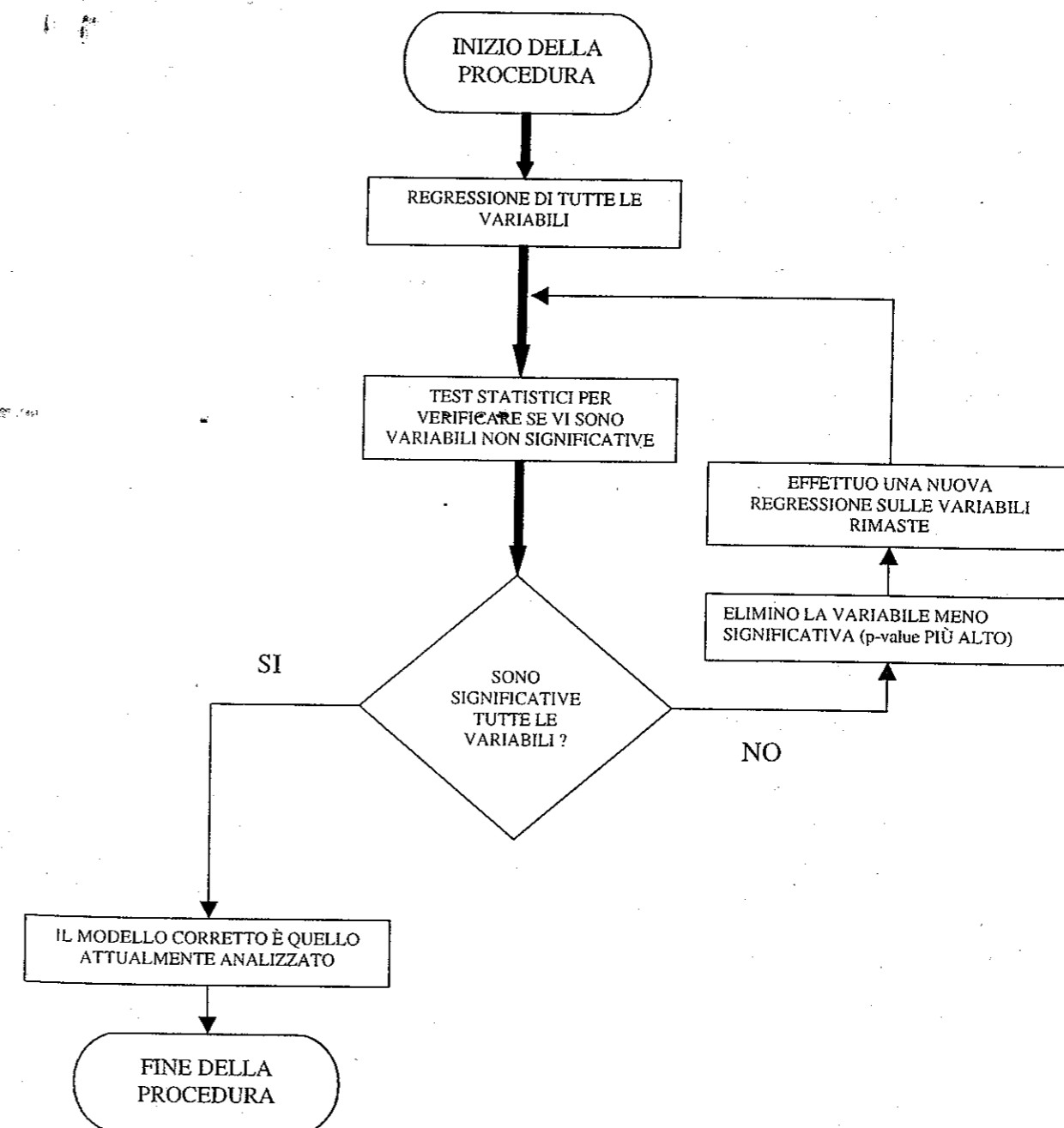


Fig. 1.2: Diagramma di flusso del Backward elimination method.

Questa tecnica è sicuramente più "economica" rispetto alle precedenti in quanto evita che si lavori con un numero di variabili superiore al necessario. Uno dei suoi svantaggi, però, è quello di non sforzarsi di esplorare gli effetti che l'introduzione di una nuova variabile può avere sul ruolo giocato da quella che è entrata allo stadio precedente. Questo difetto è risolto dall'uso della "stepwise regression procedure":

Stepwise regression procedure: nonostante il nome completamente differente, questa procedura è una versione perfezionata del "forward selection method" appena discusso. Il miglioramento consiste nel fatto di riesaminare per ogni inserimento di variabile quelle inserite nei passi precedenti. Con l'introduzione di una nuova variabile, anche la variabile più significativa tra quelle inserite fino a quel momento, infatti, potrebbe diventare superflua a causa della sua relazione con quest'ultima.

Per valutare tutto questo, ogni volta che viene introdotta una variabile si valuta se tra le variabili già inserite ce ne sono alcune che perdono la loro significatività (sempre osservando i p-value). Qualora ve ne siano, si elimineranno, se il modello complessivamente migliora.

Ripetendo quest'analisi passo dopo passo, ci si fermerà nel momento in cui tutte le variabili contenute nel modello sono significative e l'aggiunta di un'ulteriore variabile peggiora complessivamente il modello.

Questo metodo è giudicato il migliore da più autori (e ne sono state presentate anche alcune varianti [SeSr90]) anche se necessita di una supervisione continua (richiede giudizio critico con analisi dei residui a ogni passo) rendendo parecchio difficile fare affidamento su una sua meccanizzazione automatica.

The stagewise regression procedure: questa procedura non fornisce una vera soluzione dei minimi quadrati per le variabili incluse nel modello. L'idea base è la seguente: si prende la variabile maggiormente correlata con Y e si calcolano i residui della regressione. Successivamente si sceglie tra le rimanenti la variabile più

correlata coi residui stessi e si regredisce su di essi. Il processo continua fino a quando la risposta della regressione sui residui non è più significativa.

Come accennato il difetto di questo metodo è il criterio con cui si stimano i parametri che, non coincidendo con quello dei minimi quadrati, produce un complessivo peggioramento sulle eventuali previsioni richieste al modello. Presenta però il vantaggio di preservare la direzione attesa dei successivi residui a ogni passo (non possibili negli altri metodi a causa delle correlazioni tra le variabili già presenti nel modello e quella entrante o uscente).

A proposito dell'individuazione del corretto numero di variabili sorge il problema, attualmente assai dibattuto, circa la necessità di ridurre la struttura del modello (e quindi il numero di variabili esplicative).

Coloro che sono contrari a questo tipo di necessità si chiedono perché un regressore privo di importanza debba essere tolto quando, se non è rilevante, avrà comunque un coefficiente nullo. La sua eliminazione, inoltre, comporterebbe una notevole perdita di informazioni. Essi sostengono in alternativa di utilizzare metodi più raffinati per risolvere questi tipi di problemi.

Si sostiene che se uno stato del mondo fosse complesso non ci sarebbe ragione per semplificarlo se si potesse costruire un modello "migliore" più complesso. Inoltre, come affermato da S. Geisser [Geis95], a volte non si vuole interpretare una relazione, ma solamente predire un valore futuro.

Ovviamente, un simile quesito non può essere risolto solamente mediante la teoria pura, ma affrontando specificatamente ogni singolo caso sulla base dei dati a disposizione, della complessità del modello e del numero di variabili a disposizione.

Nell'analisi esercitativa che seguirà, talvolta si è stati "costretti" a ridurre il più possibile le dimensioni dei modelli per permettere all'elaboratore di effettuare i calcoli necessari e comunque per fornire previsioni entro un tempo accettabile (non avrebbe difatti senso permettere all'elaboratore di impiegare un tempo maggiore di t per effettuare una previsione al tempo t).

La problematica inerente alla selezione del modello non si riferisce soltanto alla scelta delle dimensioni di un modello, ma anche a come valutare differenti modelli e sulla base di quali indicatori preferirne uno piuttosto che un altro.

La ricerca non può essere omologata a test di ipotesi, così come avviene nella fase di verifica del modello statistico, in quanto deriva da ipotesi separate, cioè da famiglie di distribuzioni di cui una non è un caso particolare dell'altra.

Si preferisce piuttosto, dati più modelli completamente generali per gli stessi dati, minimizzare un opportuno indice, espresso in funzione dei parametri del modello, tenuto conto di due comportamenti opposti: al crescere del numero dei parametri di un determinato modello la varianza dei residui diminuisce (poiché migliora l'adattamento), ma aumentano i vincoli imposti dagli stessi parametri (e quindi peggiora la parsimonia).

Rientrano a pieno titolo in questa filosofia i seguenti indicatori:

- **L'indice di determinazione multipla corretto (*adjusted R²*)** introdotto nel 1961 da Theil [The61]:

$$(1.3) \quad \bar{R}^2 = 1 - \frac{\frac{ESS}{n-p-1}}{\frac{TSS}{n-1}};$$

Dove:

$$ESS = \sum_1^n (y_i - \hat{y}_i)^2 \text{ (devianza dei residui = Error Sum of Squares);}$$

$$TSS = \sum_1^n (y_i - \bar{y})^2 \text{ (devianza totale = Total Sum of Squares);}$$

n = numero di osservazioni;

p = numero di variabili;

- **L'indice di Mallows** introdotto nel 1973 [Mal73] e definito da:

$$(1.4) \quad C_p(p) = \frac{RSS_j}{S_p^2} - (n - 2p);$$

dove:

RSS_j = somma dei quadrati dei residui (Regression Sum of Squares) o devianza della regressione;

p = numero dei parametri del modello;

n = numero di osservazioni;

$$S_p^2 = \frac{\sum_1^n (\hat{E}_i)^2}{n - p - 1} = \text{varianza stimata sui residui in un modello con } p \text{ parametri.}$$

$$\hat{E}_i = Y - XB;$$

$$B = (X'X)^{-1} X'Y \text{ (stimatore dei minimi quadrati);}$$

- **L'indice AIC (*Asymptotic Information Criterion*)** proposto nel 1973 da Akaike [Akai73] e così definito:

$$(1.5) \quad AIC(p) = n \log(S_p^2) + 2(p);$$

Nonostante di questo indicatore esistano varianti certamente preferibili e sia stato spesso criticato poiché sovrapparametrizza il modello "ottimale", rimane sempre quello più utilizzato.

- **L'indicatore AICC (*Corrected AIC*)** presentato da Hurvich e Tsai nel 1989 [HuTs89] che rappresenta una correzione dell'indice AIC corretto:

$$(1.6) \quad AICC(p) = n \log(S_p^2) + 2 \frac{n}{n-p-2} (p);$$

• **L'indice BIC (Bayesian Information Criterion)** proposto anche questo da Akaike nel 1978 [Akai78] definito da:

$$(1.7) \quad BIC(p) = (n-p) \log\left(\frac{nS_p^2}{n-p}\right) + p \log\left(\frac{n(S_0^2 - S_p^2)}{p}\right);$$

che rappresenta una modifica dell'indice AIC per tener conto della riduzione di S_p^2 rispetto a S_0^2 (dove S_0^2 indica la varianza delle n osservazioni).

• **L'indicatore di Schwarz** [Schw78] proposto nel 1978 e, approssimativamente equivalente al criterio BIC [Prie81]:

$$(1.8) \quad SIC(p) = n \log(S_p^2) + p \log(n);$$

• **L'indice RIS** proposto nel 1978 da Rissanen [Rissa78] e così definito:

$$(1.9) \quad RIS(p) = n \log(S^2) + (p+1) \log(n+2) + 2 \log(p+1);$$

• **L'indicatore di HAN** proposto da Hannan e Quinn nel 1979 [HaQu79]:

$$(1.10) \quad HAN(p) = n \log(S^2) + 2p \log(n);$$

Tutti i precedenti criteri sono asintoticamente equivalenti³, con una tendenza alla sovrapparametrizzazione per l'AIC e una tendenza alla sottoparametrizzazione per l'HAN.

³ Anche se sono parecchi gli statistici che preferiscono utilizzare i criteri SIC e BIC.

Ritornando alle altre fasi per la costruzione del modello statistico (vedi ancora Fig. 1.1) bisogna osservare che la stima dei parametri di un modello può essere affrontata riconducendo tale problema allo schema più consolidato della teoria della stima.

Quando è possibile pianificare un esperimento, si determina un campione casuale di numerosità n dalla popolazione di riferimento e su ciascuna delle unità statistiche si rilevano sia il fenomeno da spiegare Y che le cause presumibili (X_1, X_2, \dots, X_p) .

In tal modo, su ciascuna unità del campione, si rilevano i seguenti dati:

$$(1.11) \quad (y_i; x_{i1}, x_{i2}, \dots, x_{ip}), \text{ per } i=1, 2, \dots, n$$

Pertanto, la specificazione del modello diventa la seguente:

$$(1.12) \quad Y_i = f(x_{i1}, x_{i2}, \dots, x_{ip}; \beta) + \varepsilon_i, \text{ per } i = 1, 2, \dots, n$$

In generale, la variabile Y per la i -esima unità statistica è la risultante di una parte "spiegabile" tramite le variabili X_1, X_2, \dots, X_p non stocastiche che costituiscono l'argomento della funzione $f(\bullet; \beta)$ e una parte "spiegabile di natura stocastica, dovuta alla componente casuale ε_i . La componente ε_i non è osservabile (se lo fosse la relazione funzionale sarebbe di tipo matematico) e nei dati osservati la variabile casuale ε_i si realizza nel numero e_i , che è deducibile dalla relazione seguente:

$$(1.13) \quad y_i = f(x_{i1}, x_{i2}, \dots, x_{ip}; \beta) + e_i, \text{ per } i = 1, 2, \dots, n$$

Quindi, $e_i = y_i - f(x_{i1}, x_{i2}, \dots, x_{ip}; \beta)$ sono le realizzazioni campionarie delle corrispondenti variabili casuali ε_i , ottenute sulla base del modello specificato. Le realizzazioni e_i sono note se si conoscono sia la forma funzionale del modello che i parametri β .

In questo modo ci siamo ricondotti alla tipica situazione inferenziale nella quale dal campione osservato si cerca di risalire alla determinazione numerica di uno o più

parametri della popolazione (in questo caso, i β) cui applicare un adeguato metodo di stima.

La verifica del modello statistico si concretizza in una serie articolata di decisioni inferenziali, spesso formalizzabili mediante il test delle ipotesi, orientate alla discussione critica del risultato ottenuto nella fase di stima. Se la verifica non conduce al rifiuto del modello stimato, allora tale modello può essere utilmente applicato; in caso contrario, occorre ripercorrere le tappe di specificazione-stima-verifica alla ricerca di un modello più soddisfacente.

2. IL PROBLEMA DELL'OTTIMIZZAZIONE

Come accennato in precedenza in questa dispensa si intendono affrontare oltre ai temi della selezione di un modello statistico e dell'approssimazione di un modello (a fini previsivi) anche le problematiche inerenti all'ottimizzazione di funzioni complesse da un punto di vista non-standard.

Dopo aver selezionato i parametri del modello statistico, infatti, la problematica si sposta all'ottimizzazione delle funzioni utilizzate per la stima dei parametri.

Spesso ci si trova a dover affrontare il problema di identificare il valore di un parametro che minimizza (o massimizza) una funzione. Eccetto il caso di poche (ma altrettanto significative) classi di funzioni o distribuzioni, la soluzione deve essere determinata per via numerica. Questo è il compito assolto dagli algoritmi di ottimizzazione e ricerca, capaci, appunto, di effettuare una ricerca tra le potenziali soluzioni del problema fino a determinare quella ottima. Tale argomento è diventato via via più interessante negli ultimi anni in relazione all'incrementato numero di fenomeni non lineari sempre più complessi (idrodinamica, applicazioni finanziarie, dinamiche sociali, etc...).

In termini generali il problema può essere posto nei seguenti termini:

sia

$$J(\theta_1, \dots, \theta_k)$$

una funzione il cui valore dipende solo dai parametri (variabili) $\theta_1, \dots, \theta_k$. Si costituisca un algoritmo capace di individuare i valori di $\theta_1, \dots, \theta_k$ che rendono minimo (o massimo¹) il valore di J .

Il problema può diventare estremamente arduo qualora si affronti la ricerca di un estremo in una funzione definita in uno spazio pluridimensionale (non si pensi solo a 3 o 4 variabili, ma anche a 30, 50, ...) o comunque quando si abbia a che fare con una funzione che non rispetti le condizioni di regolarità (continuità, derivabilità, ...).

Spesso si conosce o si può modellare una interazione locale ma risulta quasi impossibile fornire un modello globale di comportamento, oppure la relazione tra le variabili può essere talmente complessa che non si riesce a determinare una funzione capace di descriverla, né le tecniche di approssimazione riescono a fornire risultati apprezzabili.

Risulta quindi estremamente evidente la necessità di sviluppare algoritmi di ottimizzazione e ricerca che godano di proprietà di affidabilità anche rispetto a funzioni lontane dalle tipiche condizioni di regolarità e soprattutto risulta evidente come per utilizzare tali algoritmi sia necessario ricorrere a tecniche numeriche, cioè implementare un sistema automatico di calcolo.

Queste tecniche per essere "qualitativamente" ottimali devono sviluppare algoritmi robusti (e quindi in grado di fornire sempre il medesimo risultato), efficaci (capaci di convergere verso il minimo locale) nonché efficienti (cioè devono convergere velocemente verso la soluzione). Implementando questi algoritmi tramite elaboratori, è richiesta una ulteriore caratteristica: essi non devono consumare troppe risorse di memoria.

Esistono diversi metodi di ottimizzazione non vincolata, metodi che possono essere ampiamente raggruppati a seconda che si faccia uso o meno dell'informazione che deriva dal calcolo del gradiente.

¹ Si ricordi che lo studio del minimo o massimo non cambia i termini del problema dal momento che il massimo di una funzione può essere ricercato come il minimo dell'opposto o del reciproco della funzione stessa, e viceversa.

Principali algoritmi di ricerca:

Metodo di Newton-Raphson: rientra tra i metodi di calcolo indiretti che consistono nella ricerca degli estremi locali affrontando la ricerca degli zeri del gradiente: $\nabla J(\theta) = 0$. Se si riesce a determinare tutti gli zeri, allora si può essere certi che i punti estremanti appartengono a tale insieme. Si valuta quindi la funzione $J(\theta)$ in corrispondenza di tali punti e si sceglie quello che rende minimo (o massimo) $J(\theta)$. Il problema nasce dal fatto che non sempre tali algoritmi sono capaci di determinare tutti i punti in cui il gradiente si annulla (per esempio se le derivate non sono continue). Questo metodo permette di determinare l'ottimo attraverso un procedimento iterativo che utilizza l'approssimazione del polinomio di Taylor per approssimare il gradiente di $J(\theta)$.

Per comprendere al meglio il suo funzionamento vediamo come viene applicato nel caso di determinazione del minimo di una funzione (per semplificare ipotizziamo che sia una funzione a una sola variabile).

Sia $g(\theta) = J'(\theta)$ (imponendo quindi la derivabilità della funzione nel suo dominio). Ipotizzando che $g(\theta)$ sia derivabile (imponendo perciò l'esistenza delle derivate del secondo ordine di $J(\theta)$) e continua, si parte da un'approssimazione iniziale dello zero di $g(\theta)$: θ_0 .

Utilizzando lo sviluppo in serie del polinomio di Taylor (troncato al primo ordine) si può scrivere:

$$(2.1) \quad g(\theta_1) \approx g(\theta_0) + (\theta_1 - \theta_0)g'(\theta_0)$$

Da cui:

$$(2.2) \quad \theta_1 = \theta_0 - \frac{g(\theta_0)}{g'(\theta_0)}$$

Il valore di θ_1 rappresenta quindi un'approssimazione dello zero di $g(\theta_0)$. La tecnica viene a questo punto sottoposta a un processo iterativo, sostituendo θ_0 con θ_1 e ricavando un'approssimazione successiva iniziale θ_2 .

Continuando con l'approccio iterativo si ottiene una successione $\theta_0, \theta_1, \theta_2, \dots, \theta_k, \dots$ dove:

$$(2.3) \quad \theta_k = \theta_{k-1} - \frac{g(\theta_{k-1})}{g'(\theta_{k-1})}$$

Tale successione, sotto opportune condizioni di regolarità, converge alla soluzione cercata. Si nota facilmente come risultato cruciale in tale approccio la scelta dell'approssimazione iniziale θ_0 .

Ci si ferma quando $|\theta_k - \theta_{k-1}|$ è minore di un errore prefissato oppure quando il valore $g(\theta_k)$ è sufficientemente prossimo a zero entro un grado di approssimazione prefissato. Questo metodo per convergere con risultati sempre soddisfacenti verso l'ottimo richiede la condizione: $g''(\theta_k) > 0$ in tutti i punti del dominio (altrimenti potrebbe non portare alla determinazione del minimo). Inoltre la "bontà" del risultato dipende dalla convessità della funzione e dalla scelta del punto iniziale.

Generalizzando il problema, tale metodo può così essere rappresentato mediante il diagramma di flusso in Fig. 2.1.

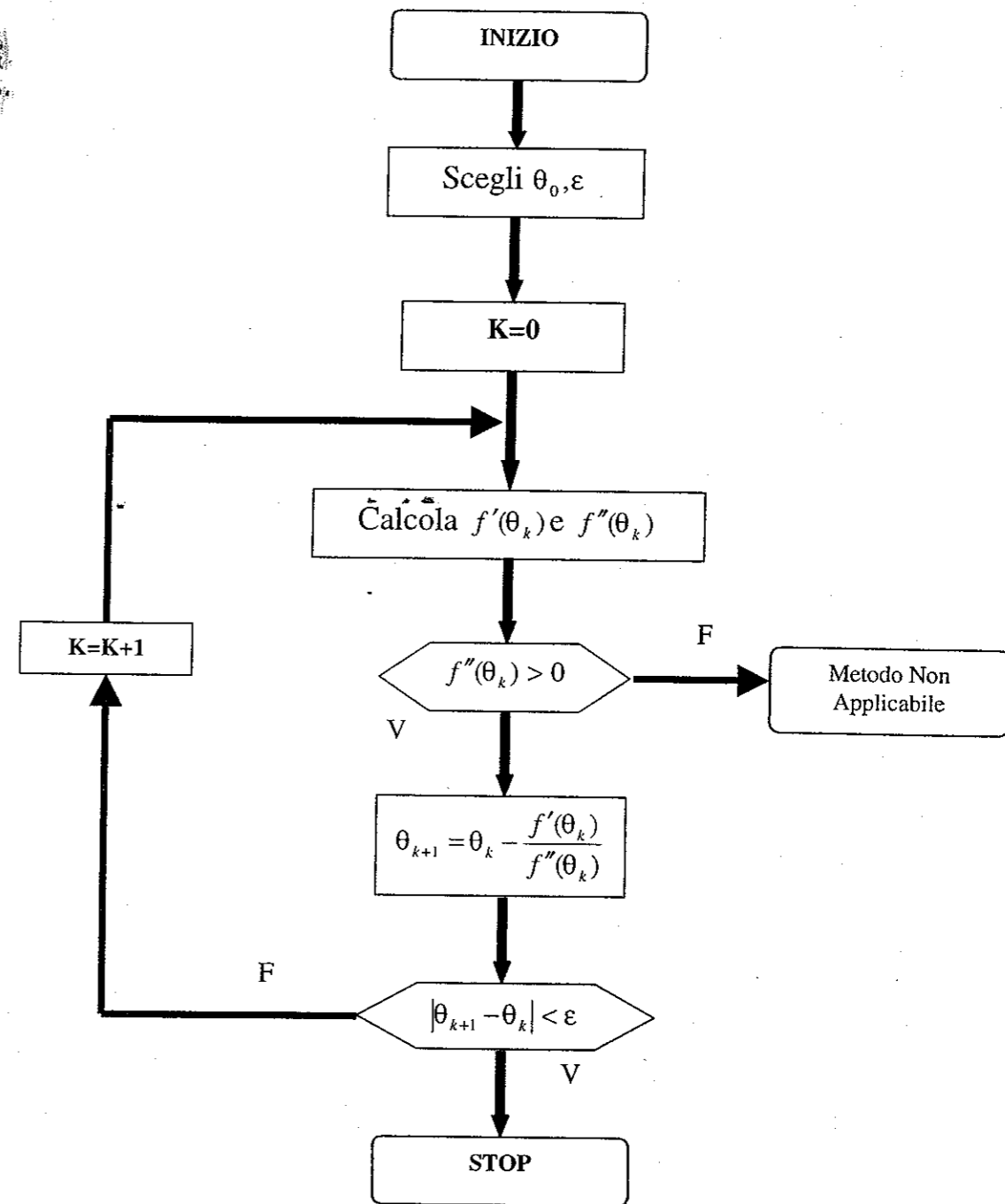


Fig. 2.1. Metodo di Newton-Raphson: diagramma di flusso.

Normalmente questo metodo utilizzato per risolvere equazioni del tipo $g(\theta) = 0$ è noto come "Metodo di Newton delle tangenti". Dalla Fig. 2.2 si può infatti osservare che disegnando la tangente alla curva $g(\theta)$ nel punto θ_{k-1} , questa determina come intersezione con l'asse delle ascisse il punto θ_k secondo l'algoritmo precedentemente descritto, e che quindi identificherà l'ascissa di un punto più prossimo alla soluzione ottima del problema. L'inclinazione di $g(\theta)$ nel punto θ_{k-1} è infatti :

$$(2.4) \quad g'(\theta_{k-1}) = \frac{g(\theta_{k-1})}{\theta_{k-1} - \theta_k}$$

e quindi:

$$(2.5) \quad \theta_k = \theta_{k-1} - \frac{g(\theta_{k-1})}{g'(\theta_{k-1})}$$

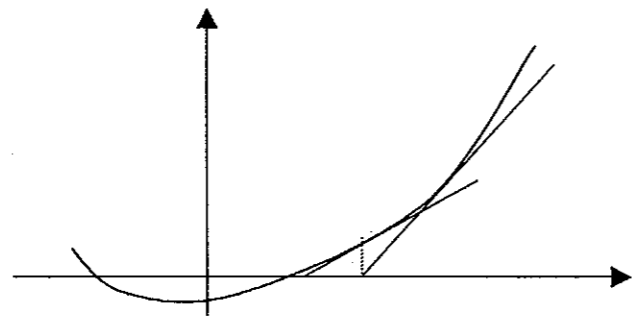


Fig. 2.2: Metodo di Newton o delle tangenti.

Nel caso in cui non fosse disponibile la derivata seconda si potrebbe tentare di risolvere il problema di ottimo approssimandola. Il cosiddetto "metodo della secante"

costituisce una possibile risposta a tale problema. In esso si prevede infatti la possibilità di approssimare la derivata seconda della funzione $J(\theta)$ nel seguente modo:

$$(2.6) \quad J''(\theta_k) = \frac{J'(\theta_k) - J'(\theta_{k-1})}{\theta_k - \theta_{k-1}}$$

e quindi:

$$(2.7) \quad \theta_{k+1} = \theta_k - \frac{\theta_k - \theta_{k-1}}{J'(\theta_k) - J'(\theta_{k-1})} J'(\theta_k)$$

Questo metodo richiede però la conoscenza di due punti iniziali. Se si pensa alla sua interpretazione geometrica si può facilmente intuire come a differenza del metodo di Newton, che usa la pendenza di g nel punto θ_k per determinare l'ascissa del nuovo punto, in questo caso si usi la secante passante per due punti appartenenti alla curva per determinare l'ascissa del punto più prossimo alla soluzione ottima.

Metodo di Newton-Raphson modificato: rientra, a differenza dei precedenti, tra gli approcci di calcolo diretti che cercano gli estremi locali partendo da un punto a caso nel campo di esistenza della funzione $J(\theta)$ e spostandosi in una direzione in cui il gradiente è massimo (o minimo)².

Tali approcci sono basati su iterazioni del tipo:

$$(2.8) \quad \theta_{k+1} = \theta_k - \gamma_k H_k^{-1} g_k$$

Dove:

² Rappresenta semplicemente la nozione di "scala la montagna nella direzione in cui è più pendente".

$$g_k = \nabla J(\theta_k)$$

$$H_k = (\nabla^2 J)_{\theta=\theta_k}$$

$$\gamma_k = \text{Step-Size}$$

in cui H_k rappresenta un'approssimazione della matrice Hessiana.

Si è osservato che, affinché si abbia convergenza verso la soluzione ottimale, non è sufficiente l'annullamento del gradiente della funzione, ma anche la matrice Hessiana associata alla funzione deve essere definita positiva. Questa è sicuramente una richiesta molto forte da soddisfare nel caso di problemi a più dimensioni. Inoltre il procedimento di inversione della matrice Hessiana non è affatto banale. Ecco perché si è sviluppato un metodo differente denominato metodo di Newton Raphson modificato che prevede l'introduzione di un parametro γ che consenta una ricerca di minimo unidimensionale lungo la direzione $d_k = -H_k^{-1}g_k$ per far sì che, anche nel caso in cui la matrice Hessiana non sia definita positiva, la direzione seguita dall'algoritmo sia comunque una direzione di discesa. Questo metodo presenta tuttavia l'inconveniente di richiedere sempre l'inversione della matrice Hessiana.

Metodo Quasi-Newton: con questo tipo di metodo la matrice Hessiana viene approssimata mediante l'uso di tecniche appropriate; tra queste, quella che si ritiene più generale è quella suggerita da Broyden, Fletcher, Goldfarb e Shanno (BFGS) nel 1970 [Broy70, Flet70, Gol70, Sha70]:

$$(2.9) \quad H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k}$$

dove:

$$s_k = \theta_{k+1} - \theta_k;$$

$$q_k = \nabla f(\theta_{k+1}) - \nabla f(\theta_k);$$

come H_0 si può scegliere qualunque matrice simmetrica positiva (per esempio la

matrice identità).

Un'altra tecnica molto importante è quella suggerita da Davidon, Fletcher e Powell [Dav59, FIPo63] (DFP) che si basa sulla precedente con l'unica differenza di sostituire q_k con s_k .

Metodi a direzioni coniugate: questo metodo si differenzia dai precedenti in quanto la "discesa" avviene lungo n direzioni fra loro coniugate rispetto alla matrice delle derivate seconde della funzione da ottimizzare³. Tra le principali tecniche che applicano questo metodo ricordiamo il metodo di Fletcher-Reeves, quello di Polak-Ribière, quello di Powell-Beale e quello di Moller.

Metodi di Gauss-Newton e Levenberg-Marquardt [Lev44, Marq63]: sfruttano entrambi procedure basate su algoritmi di ricerca dei minimi quadrati.

Nel primo, d_k viene ottenuto ad ogni iterazione come soluzione di:

$$(2.10) \quad \min_{\theta \in \mathbb{R}^n} \|A(\theta_k)d_k - J(\theta_k)\|_2^2$$

Mentre nel secondo, viene ottenuto risolvendo un sistema di equazioni lineari:

$$(2.11) \quad (A(\theta_k)^T A(\theta_k) + \lambda_k I)d_k = -A(\theta_k)J(\theta_k)$$

dove $A(\theta_k)$ rappresenta la matrice Jacobiana ottenuta dalla relazione $J^T J = H$.

La maggior parte dei metodi sopra descritti presenta poi un altro tipo di problematica: la ricerca, per ogni iterazione, della lunghezza dello Step-Size, cioè la lunghezza dello spostamento γ_k nella direzione $d_k = H_k^{-1}g_k$ (poiché negli altri

³ Data una matrice Q di dimensione $(n \times n)$, simmetrica e definita positiva, due vettori $d^{(1)}$ e $d^{(2)}$ si dicono coniugati tra loro rispetto a Q se vale la relazione: $d^{(1)T} Q d^{(2)} = 0$.

metodi è invece supposto pari a uno).

Questo parametro viene generalmente determinato attraverso le procedure cosiddette di "line search". Tra queste le principali sono:

Metodo di Fibonacci

Il metodo di Fibonacci vuole essere la risposta al problema della determinazione del criterio di scelta degli n punti, contenuti nell'intervallo $[a, b]$, in cui calcolare il valore che la funzione assume, criterio che deve consentire di ridurre al minimo l'intervallo finale in cui è contenuto l'ottimo. Esso identifica allora la strategia "ottimale" da seguire nel processo di selezione dei punti, e quindi la strategia che permette di avere la massima riduzione dell'intervallo di incertezza per un prefissato numero N di punti in cui determinare il valore della funzione.

Si intuisce allora che l'applicazione di tale metodo richiede a priori l'ipotesi di unimodalità della funzione (deve essere limitata e ammettere solo un punto di minimo!).

Si basa sull'applicazione della successione omonima $\{F_n\}_{n>0}$, generata in modo tale che sia verificata la seguente relazione: $F_n = F_{n-1} + F_{n-2}$ con $F_0 = F_1 = 1$. Consente a priori di stabilire il numero di misure necessarie per ottenere una precisione voluta. Definito $[a, b]$ l'intervallo in cui la funzione risulta definita, l'incertezza sulla posizione di minimo vale $I_1 = a - b$. Se si vogliono effettuare complessivamente n operazioni di misura, le prime due sono effettuate simmetricamente all'interno dell'intervallo $[a, b]$ a distanza $I_2 = (F_{n-1}/F_n)I_1$ dagli estremi di esso. I_2 rappresenta allora esattamente l'ampiezza del nuovo intervallo di incertezza che contiene al suo interno il punto nel quale la misura effettuata ha dato luogo al valore minore della funzione. Quindi, si procederà a determinare il terzo punto di misura nel nuovo intervallo: esso sarà scelto simmetrico al punto nel quale è stata effettuata l'ultima misura. L'intervallo di incertezza sarà allora pari a $I_3 = (F_{n-2}/F_n)I_1$. Al termine delle n misurazioni l'incertezza è ridotta a $I_n = (F_1/F_n)I_1$. In generale, volendo effettuare n misure, detta I_k l'ampiezza dell'intervallo di incertezza dopo aver effettuato k misure, sarà verificato che $I_k = (F_{n-k+1}/F_n)I_1$, dove I_1 è l'incertezza iniziale.

Metodo della sezione aurea

Di norma però la precisione desiderata in molti problemi non è espressa in termini del numero di valutazioni del valore della funzione da effettuare e inoltre il criterio di terminazione dell'algoritmo richiede generalmente che il valore della funzione si discosti nell'intervallo di incertezza finale di una quantità inferiore rispetto a una quantità piccola stabilita inizialmente. In questi casi, allora, una risposta più soddisfacente la dà il metodo della sezione aurea che deriva dall'osservazione:

$$(2.12) \quad \lim_{k \rightarrow \infty} \frac{F_{k-1}}{F_k} = \tau = 0.618$$

Tale metodo porta a ridurre l'ampiezza degli intervalli di ricerca, passo dopo passo, di una quantità pari a $1/\tau$. Ipotizzando che l'intervallo iniziale sia $[0, 1]$, i due punti saranno scelti in corrispondenza dei valori τ e $1 - \tau$. Non importa di quanto sia ridotto l'intervallo, ma piuttosto interessa il fatto che con tale metodo uno dei vecchi punti sarà in posizione corretta rispetto al nuovo intervallo. Il metodo della sezione aurea può allora essere visto come un caso limite del metodo di Fibonacci. Esso consente a ogni step di avere una riduzione dell'intervallo di incertezza di un fattore τ , e la lunghezza dell'intervallo di incertezza converge linearmente a zero.

Metodi di interpolazione polinomiale

I metodi di interpolazione polinomiale sono basati sull'idea di poter approssimare iterativamente la funzione grazie a polinomi in genere di primo, secondo o terzo grado, riuscendo ugualmente a determinare in tal modo il punto di ottimo e a fornire una forma più "semplice" della funzione che si deve analizzare.

Hanno il pregio di non richiedere informazioni sulle derivate della funzione da minimizzare. Richiedono infatti solo la conoscenza dei valori assunti dalla funzione in corrispondenza di determinati punti trovati all'inizio.

Questi metodi risultano essere in genere i più efficaci in termini di efficienza nel caso in cui la funzione da minimizzare sia continua.

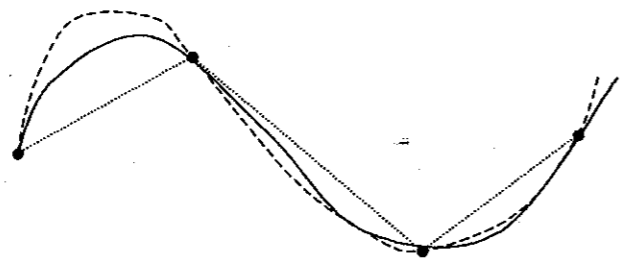


Fig. 2.3: Una funzione "smooth" (linea continua) è interpolata in modo migliore da un polinomio (linea a tratti) di ordine maggiore.



Fig. 2.4: Una funzione che presenta forti variazioni nei valori delle derivate di ordine superiore è approssimata in modo peggiore da polinomi di ordine alto.

Il numero di punti (meno 1) usato nello schema di interpolazione è chiamato ordine dell'interpolazione. Aumentare l'ordine dell'interpolazione non incrementa sempre la precisione dell'algoritmo (vedi Fig. 2.4): infatti nel caso in cui i punti fossero distanti dal punto di interesse x o la funzione presentasse cambiamenti repentini, il risultante polinomio di ordine maggiore tenderebbe a oscillare ampiamente fra i nuovi valori tabulati (fenomeno del cosiddetto "overfitting"). Questa oscillazione potrebbe non essere assolutamente collegata con il comportamento della "vera" funzione. Generalmente, però, aggiungere punti vicini ai punti desiderati migliora l'interpolazione, anche se comporta lo svantaggio della necessità di una tavola di valori di dimensioni più ampie, non sempre disponibile.

Salvo il caso in cui sia evidente che la funzione interpolante abbia una forma del tutto simile alla funzione vera, è necessario essere molto cauti con l'interpolazione di ordine maggiore: solitamente non si tollera un'interpolazione maggiore del quinto o sesto ordine. Nel caso in cui si abbia a disposizione un numero di punti maggiore dell'ordine di interpolazione desiderato, sarà allora necessario iniziare con una ricerca che porti all'individuazione dello spazio locale più giusto da utilizzare.

Sappiamo bene che attraverso due punti passa solo una retta, attraverso tre punti solo una parabola, e così via. Nei primi due casi sarà facile ricavare l'equazione della retta o della parabola passante per tali punti (data $y=mx+q$ e conoscendo $A=(x_1, y_1)$ e $B=(x_2, y_2)$ facilmente si riesce a risolvere il sistema delle equazioni determinate imponendo il passaggio della retta per tali punti, ricavando m e q), ma come si può fare se, avendo a disposizione n punti, si volesse ricavare il polinomio di ordine $n-1$ $y_1=f(x_1), y_2=f(x_2), \dots, y_n=f(x_n)$? Basta ricorrere alla classica formula esplicita di Lagrange:

$$(2.13) \quad P(x) = \frac{(x-x_2)(x-x_3)\dots(x-x_n)}{(x_1-x_2)(x_1-x_3)\dots(x_1-x_n)} y_1 + \dots + \frac{(x-x_1)(x-x_2)\dots(x-x_{n-1})}{(x_n-x_1)(x_n-x_2)\dots(x_n-x_{n-1})} y_n$$

Interpolazione lineare

È il metodo di interpolazione più rapido da implementare, imponendo di approssimare la funzione studiata con spezzate che congiungono punti adiacenti fra i punti inizialmente forniti.

Interpolazione quadratica o metodo della parabola

Dati tre punti $x_1 < x_2 < x_3$ tali che $f(x_1) > f(x_3) < f(x_2)$ si può sempre calcolare una parabola passante per essi, che sarà del tipo:

$$(2.14) \quad G(x) = ax^2 + bx + c$$

i cui coefficienti a, b, c saranno determinati dalla risoluzione del seguente sistema:

$$(2.15) \quad \begin{cases} f(x_1) = ax_1^2 + bx_1 + c \\ f(x_2) = ax_2^2 + bx_2 + c \\ f(x_3) = ax_3^2 + bx_3 + c \end{cases}$$

L'equazione della parabola approssimante sarà allora data da:

$$(2.16) \quad G(x) = f(x_1) \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} + f(x_2) \frac{(x-x_3)(x-x_1)}{(x_2-x_3)(x_2-x_1)} + \dots \\ \dots + f(x_3) \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}$$

Si procederà quindi alla determinazione di un nuovo punto x_4 risolvendo l'equazione $G'(x)=0$.

Il nuovo x_4 sarà allora ottenuto secondo la seguente formula:

$$(2.17) \quad x_4 = \frac{1}{2} \frac{f(x_1)(x_2^2 - x_3^2) + f(x_2)(x_3^2 - x_1^2) + f(x_3)(x_1^2 - x_2^2)}{f(x_1)(x_2 - x_3) + f(x_2)(x_3 - x_1) + f(x_3)(x_1 - x_2)}$$

Nel caso in cui x_4 fosse sufficientemente prossimo a x_3 , allora il nuovo punto potrebbe essere considerato un'approssimazione del minimo, oppure si dovrebbe procedere a cancellare rispettivamente x_1 o x_2 a seconda che si verifichi che $f(x_3) > f(x_4) < f(x_2)$ oppure che $f(x_1) > f(x_4) < f(x_3)$.

Non è possibile garantire che questo metodo converga a un minimo e talvolta si rende allora necessario introdurre alcune tecniche di perturbazione.

Interpolazione cubica

L'interpolazione cubica risulta essere molto utile nel caso in cui non sia particolarmente complesso il calcolo del gradiente della funzione e il calcolo di valori precisi della funzione in corrispondenza di determinati punti. Infatti, avendo a disposizione tali informazioni si potrà determinare un punto di minimo per la funzione di approssimazione cubica, imponendo che l'annullamento della sua derivata prima e la positività della sua derivata seconda.

Dati quattro punti iniziali si potrà allora determinare senza particolari difficoltà i valori dei coefficienti a, b, c e d del polinomio di terzo grado approssimante

$$(2.18) \quad G(x) = ax^3 + bx^2 + cx + d$$

e procedere quindi al calcolo dei valori di x che portano all'annullamento della derivata prima, ottenuti dalla risoluzione in questo caso di un'equazione di secondo grado del tipo

$$(2.19) \quad G'(x) = 3ax^2 + bx + c$$

Perché l'equazione abbia soluzione reale dovrà inoltre essere verificato che la scelta dei punti iniziali sia tale per cui la radice $6ax+2b$ sia positiva. La risoluzione dell'equazione condurrà allora alla determinazione di un punto di minimo e di uno di massimo.

3. TECNICHE COMPUTAZIONALI

In questi ultimi anni si stanno sempre di più affermando approcci computazionali ai problemi della ricerca, dell'approssimazione e della simulazione. In questo caso l'aspetto computazionale risulta prevalente rispetto a quello analitico che, talvolta, risulta del tutto assente. Nei prossimi capitoli verrà presentato e affrontato il tema dell'automazione di un processo di ricerca e di simulazione utilizzando alcune tecniche computazionali "soft" (Algoritmi Genetici, Reti Neurali, Logica Fuzzy, Algoritmi Ibridi¹, ...) confrontando i risultati ottenuti con le più tradizionali tecniche classiche sopra descritte.

Le discipline appartenenti al campo del "Soft Computing" [SC98] nascono con l'intento di fornire un insieme di tecnologie più flessibili, più idonee per affrontare i problemi del mondo reale, dove niente è assolutamente certo, ma spesso si hanno a disposizione molteplici possibili soluzioni. Attraverso le metodologie classiche, la formulazione dei problemi porta a definire sistemi perfettamente modellati o controllati, la cui configurazione risulta essere un procedimento molto complesso, data la mancanza di una informazione certa, precisa e completa. Da questa constatazione emerge allora chiaramente la necessità che ha portato all'affermazione

¹ Gli Algoritmi Ibridi sono strumenti "potenzialmente" superiori rispetto ai singoli metodi computazionali evoluti (Reti Neurali, Algoritmi Genetici, Logica Fuzzy, ...) in quanto nascono dalla volontà di combinare le differenti metodologie, nel tentativo di risolvere i difetti dei singoli strumenti con i pregi di altri.

di un diverso sistema di approcci, in un certo senso approssimati, capaci di trattare l'informazione imperfetta e incompleta ottenibile dal mondo reale.

L'enorme sviluppo dell'universo computazionale ha favorito l'affermarsi di queste discipline, che si sono spinte sempre oltre, nel tentativo di creare sistemi automatici di ragionamento, in grado di elaborare delle informazioni a supporto di processi decisionali, individuando quindi caso per caso la strategia da seguire. È allora in questo contesto che si può anche parlare di una sorta di "Intelligenza Artificiale"², evitando però di enfatizzare il termine *Intelligenza*. Non siamo infatti di fronte a una sorta di "miracolo" umano. Siamo semplicemente di fronte a un modo nuovo di "vedere" le cose, a una nuova possibile strada da seguire per giungere alla soluzione "ottima" di un dato problema, strada che sembra voler riproporre l'itinerario seguito dalla mente umana (anche se bisogna tenere ben presente che il cervello umano è una realtà ancora non del tutto conosciuta). I passi che si stanno muovendo portano allora verso tentativi di replicare la struttura cerebrale (*Reti Neurali*), introducendo quindi l'importante concetto di una elaborazione parallela e non più seriale, il contesto decisionale umano (*Logica Fuzzy*), nel tentativo di considerare le diverse sfumature del mondo, il contesto evolutivo umano (*Algoritmi Genetici*), dove si afferma il concetto di sopravvivenza dell'individuo migliore.

Si è scelto l'utilizzo di MATLAB come linguaggio di programmazione perchè, oltre alla sua affermazione in ambito scientifico come un linguaggio di programmazione efficace e versatile per la risoluzione di problemi più o meno complessi, possiede una ricca collezione di librerie specifiche di routines (toolboxes) che lo rendono adatto più di altri linguaggi scientifici a questo tipo di programmazione.

² Con il termine "Intelligenza Artificiale" si intende di norma una vera e propria disciplina atta a studiare i fondamenti teorici, i metodi principali, i criteri di progettazione e la costruzione di programmi che possano permettere al calcolatore di svolgere attività tipicamente umane, e di renderlo capace di sostituirsi allo stesso uomo nell'affrontare una molteplicità di problemi, nello stesso modo in cui egli li avrebbe affrontati, ottimizzando però il raggiungimento dei risultati.

Prima di esaminare i vari strumenti applicati sarà fornita di seguito una breve rassegna delle principali metodologie appartenenti al campo del "Soft Computing" e dell'Intelligenza Artificiale, concentrandoci su quelle tecniche che hanno ottenuto maggior successo, quali i sistemi esperti, il Simulated Annealing, le Reti Neurali, gli Algoritmi Genetici e la Logica Fuzzy.

Queste tecnologie sono state e vengono anche oggi utilizzate in ambito statistico in contrasto con le più convenzionali tecniche classiche.

I sistemi esperti costituiscono l'applicazione più conosciuta dell'Intelligenza Artificiale simbolica, che considera elaborazioni svolte in campo simbolico e non semantico. Lo scopo è quello di realizzare sistemi artificiali "intelligenti" basandosi sul cosiddetto paradigma simbolico che assimila il comportamento intelligente, tanto nell'uomo come nella macchina, a una elaborazione puramente formale di simboli, sintattica e non semantica.

Sono nati con lo scopo di risolvere problemi complessi di interesse pratico, quali diagnosi, progettazione e pianificazione, cercando di superare le possibili soluzioni ottenibili grazie agli algoritmi classici.

La ricerca di soluzioni riprende la dimostrazione automatica di teoremi, con deduzioni successive, partendo da assiomi. È un metodo che automaticamente porta all'affermazione di verità, grazie all'utilizzo dei principi deduttivo e induttivo, di cui ogni giorno l'uomo si avvale, in modo pressoché involontario. Sebbene siano risultati utili in diversi campi di applicazione e nonostante la loro capacità di evidenziare il percorso seguito per il raggiungimento della soluzione (e il motivo sottostante al raggiungimento della stessa) hanno mostrato numerosi punti deboli.

Un loro grande limite è costituito dalla difficoltà di costruire un insieme di leggi completo e consistente, che permetta di risolvere una molteplicità di problemi differenti. Risulta infatti molto difficile riuscire a rappresentare tutte le conoscenze empiriche, non formali, di buon senso, che ogni essere umano utilizza nel processo di risoluzione di uno specifico problema.

Inoltre, oggi non si accetta più tranquillamente la legge, che costituisce il fondamento dei sistemi esperti, *intelligenza = ragionamento formale*.

È infatti noto come il ragionamento formale sia un meccanismo tipicamente sequenziale, mentre il ragionamento umano si sviluppa in modo parallelo, grazie a un sistema nervoso costituito da neuroni interconnessi fra loro, che non agiscono in maniera isolata, ma partecipano congiuntamente nella ricerca della soluzione ottima.

I sistemi esperti non hanno inoltre la capacità di apprendere, non sono in grado di auto-modificare la loro base di conoscenza a seguito dell'elaborazione delle informazioni, e quindi della loro stessa esperienza.

Il *simulated annealing* [Aar89, Adl93], che sta trovando oggi sempre più consenso, viene classificato come tecnica probabilistica "hill-climbing". Esso trae origine dal processo di ricottura/raffreddamento dei metalli. In natura si può osservare come, dopo che la fonte di calore sia stata rimossa da un metallo liquefatto, la sua temperatura inizi a diminuire. In corrispondenza di ogni livello di temperatura si riduce l'energia delle particelle del metallo, e questo si irrigidisce. Tale processo continua fino a quando la temperatura dello stesso non coincida alla temperatura dell'ambiente circostante. In questo stadio l'energia è stata minimizzata e il metallo è completamente solido. L'intero processo è controllato e regolato in modo tale da evitare la debolezza tipica di un metallo sottoposto a raffreddamento troppo rapido. Esso porta a ridurre un indice della temperatura e calcola le corrispondenti stime dei parametri che sono stati ottimizzati. Questo è ottenuto variando i valori dei parametri ottimi corrispondenti al livello di temperatura precedente allo stadio considerato, di ammontari dipendenti dal valore della corrispondente temperatura: la misura della variazione diminuisce al diminuire della temperatura e ciò aiuta a migliorare la precisione dei parametri man mano che ci si avvicina all'ottimo globale. A ogni stadio sarà determinato il valore dell'energia ottenibile dai parametri individuati in quello stadio. Se il valore dell'energia nello stadio considerato risulterà essere inferiore al livello corrispondente allo stadio precedente, si procederà alla sostituzione dei parametri ottimi determinati precedentemente, con i parametri dello stadio attuale, altrimenti si procederà a scartare i parametri corrispondenti allo stadio considerato.

Il processo di variazione dei parametri, simulazione, calcolo dell'energia, selezione e riduzione della temperatura sarà ripetuto fintanto che l'energia non avrà raggiunto il livello minimo desiderato o fintanto che la temperatura non sarà diventata tanto bassa da non potere causare una variazione significativa dei parametri.

Quando l'algoritmo del *simulated annealing* sarà terminato, i parametri dovrebbero aver raggiunto i valori ottimi. Ciò non avverrà sempre. Infatti un grosso limite della procedura è la forte dipendenza del raggiungimento della soluzione ottima dalle condizioni iniziali. Esso consente di indagare l'area circostante alle condizioni iniziali, non fornendo quindi una completa esplorazione dello spazio di ricerca delle potenziali soluzioni. Ha fornito infatti buoni risultati nell'ambito della ricerca locale, il che porta a suggerire un suo potenziale utilizzo nella fase finale della risoluzione di problemi, affrontati inizialmente con altri metodi migliori nella ricerca globale, quali gli algoritmi genetici e le reti neurali.

Gli Algoritmi Genetici [Holl75, Gold89, Davi91, Mit96] costituiscono una classe di tecnologie di ricerca probabilistica che si ispirano all'evoluzione biologica. L'algoritmo fornisce uno strumento per l'interrogazione di insiemi molto estesi di dati e per analizzare relazioni funzionali complesse. Il processo di ricerca si basa su una versione simulata dell'evoluzione in senso darwiniano, nella quale una popolazione di soluzioni candidate vengono manipolate e condizionate, per mezzo di una strategia artificialmente generata di sopravvivenza dei soggetti più adatti dal punto di vista evolutivo.

Gran parte del lavoro effettuato dagli statistici viene dedicato alla costruzione di modelli, alla stima dei loro parametri, nonché alla convalidazione dei modelli stessi. La maggior parte del lavoro della statistica applicata viene intrapreso attraverso considerazioni e restrizioni di tipo matematico e computazionale. Così, per esempio, continuità e differenziabilità della forma funzionale di un modello sono requisiti artificiali imposti per una più facile e pronta disponibilità dei metodi di stima. Gli Algoritmi Genetici rimuovono tali ipotesi permettendo di trovare soluzioni ottimali in totale assenza di restrizioni imposte.

Qui studieremo in particolare il loro comportamento nei problemi di identificazione di modelli di regressione lineare e di modelli autoregressivi lineari nonché nell'ottimizzazione di funzioni.

Le Reti Neurali artificiali [Hag96], come si può facilmente intuire dallo stesso nome, si ispirano in modo esplicito al sistema nervoso umano. A differenza degli altri sistemi previsionali statistici, che si basano su regole e su modelli definiti a priori, le Reti Neurali modellano il sistema matematico che deve poi, dopo un opportuno allenamento, simulare le previsioni trovando in modo quasi naturale le connessioni e le correlazioni esistenti sull'insieme di dati in possesso da analizzare.

Le Reti Neurali, per la loro costruzione, simulano la struttura del cervello umano: vi sono cioè una serie di "neuroni" che rappresentano le diverse variabili che entrano in gioco nella previsione e che vengono date in analisi al sistema. Detti "neuroni" sono legati da "connessioni" che vengono simulate da funzioni matematiche non lineari. Ogni neurone modifica tramite questa funzione il valore della variabile, inviandone il risultato al neurone successivo. Ciascun neurone, a sua volta, riceve questi segnali da tutti i neuroni precedenti, li somma, e restituisce il risultato, a sua volta modificato, al neurone successivo o finale (previsione).

Il risultato finale è quindi la costruzione di un modello matematico che, a qualsiasi serie di dati forniti, risponda con una previsione ottenuta "pesando" opportunamente i vari input.

Gli aspetti interessanti sono:

- non occorre definire le regole che a priori definiscano e disciplinino la relazione e l'interazione tra i vari input e la previsione che si vuole realizzare;
- il sistema "scopre" gli input e le connessioni che meglio si correlano alle previsioni passate, usate per l'apprendimento.

Unica condizione è pertanto quella di presentare al sistema una serie di input che siano fortemente correlati alla variabile da predire, anche se non è necessario conoscere l'essenza e la forza che li lega.

Studieremo in particolare il loro comportamento nella stima dei parametri di modelli non lineari di previsione per serie storiche di tipo finanziario e in alcune analisi sperimentali relative alle previsioni del livello della marea a Venezia e della disoccupazione in Emilia Romagna.

Nei Sistemi Fuzzy [Zade84, DuPr80] viene abbandonata la tipica logica matematica bivalente, dove non esistono vie di mezzo tra il vero e il falso, il bianco e il nero, il giusto e lo sbagliato, a favore di una logica multivalente in cui sono ammesse le varie sfumature tra i due estremi.

Tale logica, che discende dalla Teoria degli Insiemi Sfumati, fornisce un meccanismo per consentire ai sistemi preposti ai processi decisionali di gestire informazioni caratterizzate da un basso grado di precisione. Se ne desume che tale logica non può essere un algoritmo né una classe di algoritmi come nel caso delle due precedenti tecnologie. Come tale, non costituisce in modo intrinseco uno strumento di automazione, ma piuttosto serve a facilitare la definizione degli ambiti di determinate forme di processo decisionale che potrebbero rendersi necessarie quando un sistema si trova a dover interagire con il mondo esterno. In particolare, la Logica Fuzzy costituisce uno strumento flessibile attraverso il quale un sistema può allo stesso tempo ricevere istruzioni e dare spiegazioni all'utente sulle azioni intraprese.

Un loro approccio verrà considerato soltanto allo scopo di definire, per un determinato modello statistico, quali e quanti parametri considerare sulla base di varie e contrastanti informazioni e valutazioni.

4. SELEZIONE DI UN MODELLO LINEARE

In questo capitolo viene presentato un esempio di modello lineare cercando di affrontare la problematica inerente alla sua specificazione (già esposta nel primo capitolo) sia attraverso le tecniche classiche più comuni, sia attraverso tecniche computazionali "intelligenti".

Il problema può essere posto nei seguenti termini:

dato l'insieme $X \equiv \{x_1, x_2, \dots, x_n\}$ di variabili indipendenti, di cui disponiamo k osservazioni, e la variabile dipendente Y , si tratta di determinare tra tutti i possibili sottoinsiemi propri e impropri di X (che come detto in precedenza sono, escludendo il solo modello vuoto, $2^n - 1$) quello che meglio degli altri specificati il modello in esame.

Se $Y = f(x, \Theta, \varepsilon)$ rappresenta il modello scelto, che in ipotesi di additività dell'errore ε può scriversi $Y = f(x, \Theta) + \varepsilon$, x rappresenta l'insieme delle variabili indipendenti scelte per "spiegare" la Y .

Supponendo ora che la relazione f che lega le variabili indipendenti a quella dipendente sia tipo lineare (univariato o multivariato) il problema successivo sarà quello di determinare i coefficienti dei parametri Θ attraverso le solite procedure dei minimi quadrati o di massima verosimiglianza.

Quindi, se in generale il problema prevede la risoluzione dei tre seguenti sottoproblemi:

- selezionare f
- selezionare x
- determinare Θ

nel caso specifico sarà determinante la selezione di x (date le ipotesi fatte su f e sul calcolo di Θ).

A questo proposito abbiamo osservato che le difficoltà da risolvere riguardano sia il numero di variabili da scegliere (o meglio quali variabili scegliere) sia i criteri coi quali effettuare questa scelta.

Per meglio comprendere le difficoltà che si incontrano sarà meglio procedere affrontando un caso particolare piuttosto rappresentativo basato su un campione di dati (contenuto nel data-set "choles") presente in MATLAB.

Il dataset "choles" (i dati sono contenuti nel file "Choles_all.mat") contiene come insieme di variabili indipendenti gli spettri di assorbimento di radiazione elettromagnetica misurata a 21 frequenze diverse su 264 campioni di sangue e come variabile dipendente il livello di colesterolo presente nei 264 campioni.

Ci si chiede: "Quali sono le frequenze tra le 21 disponibili che determinano il valore di Y ?"

Per poter rispondere a tale domanda è necessario spiegare le tecniche attraverso le quali si è cercato di risolvere il problema.

MATLAB dispone a questo riguardo di una sola funzione denominata "Stepwise" [STATISTIC Toolbox] che permette in modo manuale di selezionare o eliminare dall'insieme complessivo di variabili a disposizione quelle più o meno significative valutate sulla base del p-value calcolato sul test statistico t (quindi risulteranno non significative le variabili con p-value > 0.05 se si prende come riferimento un test statistico al 95%).

Purtroppo questo tipo di procedura, come detto, funziona manualmente e non permette assolutamente di generare in modo completamente automatico l'insieme di

variabili rilevanti (cioè, nel caso specifico, quelle frequenze che sono in grado di determinare il valore del livello di colesterolo nel sangue).

Pertanto tutte quante le tecniche di seguito riportate sono state generate attraverso codici computazionali che potessero funzionare in modo del tutto automatico indipendentemente dai dati utilizzati.

Inoltre, bisogna premettere che questo campione di dati presenta problemi di multicollinearità (cioè le variabili esplicative sono fra loro fortemente correlate) rendendo il confronto fra le diverse tecniche ancora più interessante.

Backward elimination method: questa metodologia, già spiegata nel primo capitolo, è stata implementata negli stessi termini del diagramma di flusso di Fig. 1.2. Tale procedura, quindi, per ogni iterazione scarta tra i regressori non significativi quello meno significativo, fermandosi quando tutte le variabili rimaste sono significative.

Con la stessa tecnica di procedimento (dal modello più ampio al modello più ristretto) si è poi cercato di generalizzare tale metodo al fine di poter utilizzare altri criteri. Si è così sviluppato un programma che prevedesse l'eliminazione di quelle variabili che, una volta "tolte", migliorano più delle altre il modello in base a un determinato indicatore.

Gli indicatori utilizzati sono quelli esposti nei capitoli precedenti: R^2 , \bar{R}^2 , AIC , $AICC$, BIC , SIC , RIS e HAN .

Criterio di stop sarà quindi il peggioramento del modello (sulla base di tali indici) all'ulteriore "mossa" successiva.

L'altra procedura implementata, facente parte delle cosiddette tecniche classiche, è quella di tipo "Forward Selection".

Ricordiamo che questa tecnica consiste nell'inserire una variabile per volta al modello a seconda del suo coefficiente parziale di correlazione (così come descritto nel primo capitolo) fino a che la successiva variabile inserita risulti essere non significativa.

Allo stesso modo di quanto fatto per il programma "Backward", anche qui si è cercato di generalizzare il metodo mediante un programma che consentisse l'utilizzo di questa procedura anche attraverso altri criteri.

Il programma in questione prevede di inserire in modo iterativo la variabile che più di ogni altra migliora la "bontà" del modello in base all'indicatore prescelto.

Il processo avrà termine quando, inserendo qualunque ulteriore variabile, il modello ha comunque un peggioramento complessivo.

È necessario ora spiegare come sono stati calcolati i vari indici sul campione di dati a nostra disposizione.

L'intero "data-set" (264 campioni di sangue) è stato suddiviso in tre parti:

- Training set (l'esatta metà dei campioni di sangue);
- Validation set (un quarto dell'intero set di dati);
- Test set (la rimanente parte).

Nonostante diverse siano state le modalità attraverso cui sono state suddivise le percentuali di questi campioni (provando a "rimiscolare" l'intero data-set ma sempre attribuendo il 50% dei dati al training set e il 25% rispettivamente al validation e al test set) non si sono avvertiti significativi cambiamenti nei risultati mostrati.

Per quanto riguarda il "training" set si può dire che esso costituisce quell'insieme di dati sul quale si è effettuata la regressione lineare con lo scopo di calcolare i coefficienti delle variabili selezionate.

Il "validation" invece costituisce quell'insieme di dati sul quale si sono calcolati gli indici utilizzati come criterio di valutazione della bontà del modello.

Sul "test" set invece si è calcolato l' R^2 e gli indici della regressione.

I risultati ottenuti sono indicati nelle due tabelle di seguito riportate.

METODO	CRITERIO	RSQ _{test}	MSEP	N.V.	N _t
Backward	<i>p-value</i>	0.788	5.85	12	-
Backward	<i>Theil</i>	0.808	5.36	14	5
Backward	<i>RSQ</i>	0.808	-	18	11
Backward	<i>AIC</i>	0.781	4.62	11	6
Backward	<i>AICC</i>	0.781	4.62	11	6
Backward	<i>BIC</i>	0.781	4.62	11	6
Backward	<i>SIC</i>	0.781	4.62	11	6
Backward	<i>RIS</i>	0.781	4.62	11	6
Backward	<i>HAN</i>	0.746	4.82	7	-

Tabella 4.1. Risultati ottenuti con il programma Backward sul campione di dati "choles".

METODO	CRITERIO	RSQ _{test}	MSEP	N.V.	N _t
Forward	<i>P_corr</i>	0.736	4.76	2	-
Forward	<i>Theil</i>	0.757	4.24	2	-
Forward	<i>RSQ</i>	0.763	4.22	3	1
Forward	<i>AIC</i>	0.757	4.24	2	-
Forward	<i>AICC</i>	0.757	4.24	2	-
Forward	<i>BIC</i>	0.757	4.24	2	-
Forward	<i>SIC</i>	0.757	4.24	2	-
Forward	<i>RIS</i>	0.757	4.24	2	-
Forward	<i>HAN</i>	0.757	4.24	2	-

Tabella 4.2. Risultati ottenuti con il programma Forward sul campione di dati "choles".

Nella prima colonna delle due tabelle, "metodo" indica il tipo di programma utilizzato per ottenere i risultati, nella seconda colonna "criterio" indica quale indice di riferimento si è utilizzato (*p-value* indica semplicemente che si è utilizzato il metodo "backward elimination" suesposto, *P_corr* invece indica l'utilizzo del "Forward selection" puro), nella terza colonna è stato inserito l' R^2 calcolato sul test set, nella quarta colonna viene riportato il valore della stima dello scarto quadratico medio previsivo, nella quinta colonna è evidenziato il numero di variabili scelte dal programma per quel tipo di criterio (dimensione del modello ottimo) e nell'ultima colonna viene indicato quante variabili tra quelle scelte risultano statisticamente non significative sulla base del test-t.

Per quanto concerne i criteri: *p-value* e *P_corr* sono già stati spiegati, *Theil* è l'indice di determinazione multipla corretto o \bar{R}^2 , *RSQ* rappresenta l' R^2 , *AIC* è l'indice di Akaike (Asymptotic Information Criterion), *AICC* è l'indicatore *AIC* corretto, *BIC* coincide con il "Bayesian Information Criterion", *SIC* sta a indicare l'indice di Schwarz, *RIS* quello di Rissanen e *HAN* quello proposto da Hannan e Quinn.

Nella tabella 4.3 sono invece rappresentati i risultati ottenuti con la procedura manuale di "Stepwise" del toolbox statistico di MATLAB e i dati relativi al modello saturo (completo di tutte le 21 frequenze).

METODO	CRITERIO	RSQ _{test}	MSEP	N.V.	N _t
Stepwise	MANUALE	0.736	4.76	2	-
Modello completo	-----	0.790	5.94	21	14

Tabella 4.3. Risultati ottenuti con la funzione "Stepwise" (eliminazione manuale delle variabili) e con il modello completo.

Dall'esame di queste tabelle si possono trarre alcune conclusioni: la presenza di variabili fortemente correlate rende sicuramente preferibile in generale i risultati ottenuti con il programma "forward" rispetto a quelli ottenuti con il programma "backward". Infatti, nonostante l' R^2 presenti valori leggermente più bassi, la dimensione del modello è decisamente inferiore (e questo in parte spiega un minore R^2) con la totalità delle variabili statisticamente significative (eccetto il caso in cui si è scelto come criterio di ottimizzazione l' R^2 , che presenta una sola variabile non significativa in base al test statistico t a 0.95). Inoltre anche lo scarto quadratico medio è decisamente minore a conferma della maggiore "bontà" dei modelli.

Premesso questo, analizziamo ora all'interno del medesimo programma i risultati dei diversi criteri di scelta:

- Nel programma "backward" la scelta di utilizzare il *p-value* come criterio, e quindi di giudicare la bontà del modello sulla base della sola significatività delle variabili in base al test-t, non risulta convincente in quanto lo scarto quadratico medio previsivo della regressione risulta assai più alto rispetto agli altri (MSE = 5.85) e il numero di variabili selezionate non è inferiore (N.V. = 12) sebbene queste siano ovviamente tutte significative.

Il criterio di scelta complessivamente migliore per l'utilizzo di questo programma sembra essere, nonostante la presenza di ben sei variabili non significative sulla base del test-t, il metodo che sfrutta come criterio di scelta l'indice *HAN* che seleziona un modello di dimensione inferiore e con uno scarto quadratico minore. Per quanto riguarda gli indicatori *AIC*, *AICC*, *BIC*, *SIC* e *RIS* i risultati sono i medesimi.

- Nel programma "forward", come già detto, i risultati migliorano.

Confrontando i singoli criteri bisogna osservare che nella maggior parte di essi i risultati sono tutti uguali con eccezione del "*P_corr*" che risulta essere il peggiore sia in termini di R^2 che di scarto quadratico medio previsivo. L' R^2 presenta risultati leggermente migliori sia in termini di R^2 che di scarto quadratico medio previsivo.

Concludendo possiamo affermare che, utilizzando le tecniche classiche modificate per introdurre la scelta attraverso indicatori differenti da quelli generalmente proposti nelle tecniche pure di "backward elimination" e "forward selection" ("*p-value*" nel "backward" e "*P_corr*" nel "forward"), si ottengono risultati migliori. Inoltre, i risultati evidenziati nella tabella 4.3 mostrano che le procedure automatizzate qui implementate migliorano i risultati sia rispetto al modello completo (prevedibile a causa della forte correlazione tra le variabili) sia rispetto alla procedura manuale "Stepwise" di MATLAB.

4.1. IL MODELLO GENETICAMENTE EVOLUTO

Tutte le tecniche classiche finora presentate soffrono di due difetti non trascurabili che le rendono fortemente criticabili: includono (o escludono) tutte, infatti, una variabile per volta "costruendo" un modello finale in funzione dell'ordine della sequenza attraverso la quale viene fatta la scelta.

Se si confrontano i risultati ottenuti con le due tecniche viste è possibile rendersi immediatamente conto di quanto questa "dipendenza dalla sequenza" sia importante. Multicollinearità a parte, infatti, a priori sarebbe stato lecito aspettarsi i medesimi risultati con lo stesso criterio per entrambe le procedure. Ma così non è stato.

L'altro difetto consiste nel fatto che, avendo automatizzato la procedura, viene meno la possibilità da parte dell'operatore di selezionare manualmente quelle variabili che, indipendentemente dal peggioramento o miglioramento globale del modello, la teoria sottostante giudichi imprescindibili.

La necessità di costruire un algoritmo efficiente e che permetta di selezionare un modello che non dipenda dalla sequenza delle variabili scelte, induce a provare a costruire un algoritmo genetico.

Gli algoritmi genetici [Holl75, Gold89, Davi91, Mit96] costituiscono una classe di tecnologie di ricerca probabilistica che si ispirano all'evoluzione biologica.

La Fig. 4.1 rappresenta il ciclo evolutivo dell'algoritmo genetico che costituisce il meccanismo centrale di ricerca. L'algoritmo imita l'evoluzione naturale attraverso il cambiamento (o evoluzione) ripetuto di una popolazione di soluzioni candidate nel tentativo di trovare la soluzione ottimale.

L'evoluzione artificiale è paragonabile a un allevamento. Ci sono mutamenti casuali nella popolazione ma è l'allevatore a decidere di volta in volta quali sono le caratteristiche interessanti che devono essere selezionate. Questo tipo di evoluzione si dice pertanto diretta per distinguerla da quella naturale (indiretta) in cui gli individui sono migliori se in generale sopravvivono e si adattano ai mutamenti che possono avvenire nell'ambiente.

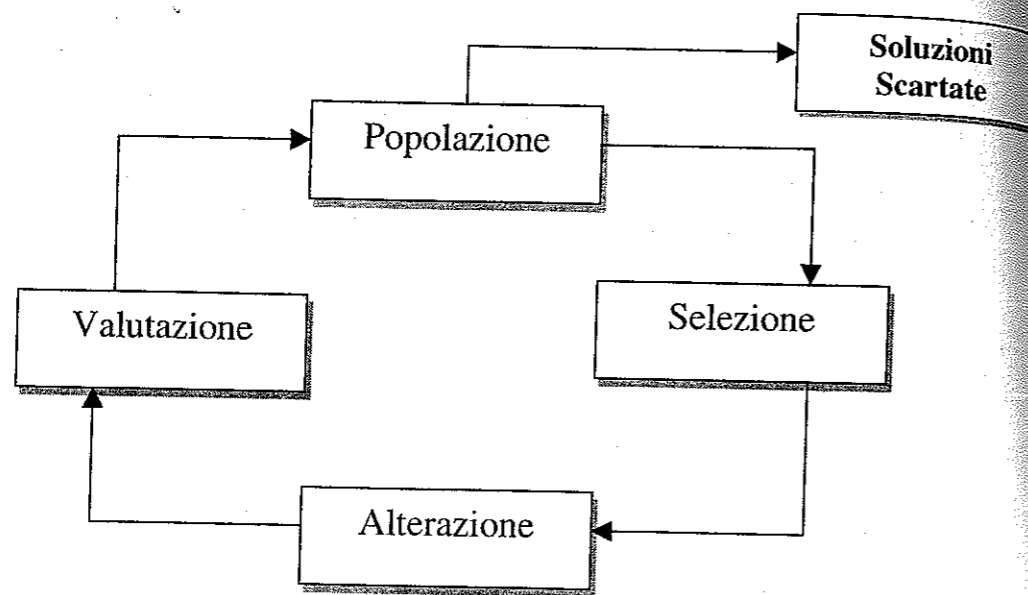


Fig. 4.1. Ciclo evolutivo di un Algoritmo Genetico.

Scopo del nostro algoritmo è quello di determinare tra le 21 frequenze dei nostri campioni quali riescano a determinare il livello di colesterolo nel sangue.

Si codificheranno pertanto le 21 frequenze in una sola stringa, che rappresenta un potenziale individuo (cromosoma) della popolazione corrispondente a una determinata generazione dell'algoritmo, la quale, sulla base del principio darwiniano di sopravvivenza dell'individuo migliore, e quindi di quello che ottiene la migliore prestazione in base all'indicatore prescelto, evolverà e fornirà le variabili che meglio valutino il livello di colesterolo nel sangue.

Di conseguenza nella nostra rappresentazione a ogni individuo sarà associato un possibile modello statistico.

La rappresentazione dell'individuo è riportata in Fig. 4.2.

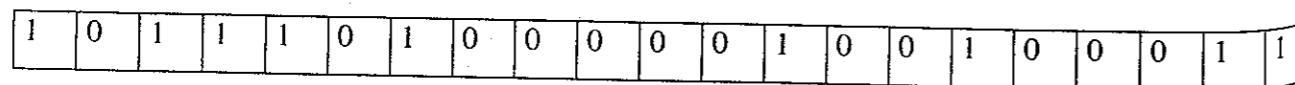


Fig. 4.2. Rappresentazione dell'individuo utilizzato nel nostro algoritmo.

Ogni stringa è suddivisa in 21 celle (geni), ognuna delle quali può trovarsi in uno dei due stati acceso-spento: 0 o 1 (allele).

Ogni gene corrisponde a una frequenza diversa, se il corrispondente allele avrà valore 1 allora significherà che quella frequenza sarà inclusa nel nostro modello, se invece avrà valore 0 significherà che sarà esclusa.

All'individuo rappresentato in Fig. 4.2 corrisponderà, dunque, il seguente modello:

$$(4.1) \quad Y = \beta_1 X_1 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \beta_7 X_7 + \beta_{13} X_{13} + \beta_{16} X_{16} + \beta_{20} X_{20} + \beta_{21} X_{21}$$

La popolazione iniziale viene generata in modo casuale. Il numero di individui e di generazioni massime potrà essere definito a piacimento (nel caso in esame sotto riportato si è scelto di generare una popolazione di 100 individui con un numero massimo di generazioni pari a 100). Quando la popolazione è stata codificata può iniziare il ciclo evolutivo dell'algoritmo genetico.

Ogni stringa della popolazione viene poi valutata secondo una "funzione di Fitness" che indica il suo grado di adattamento e che costituisce il criterio in base al quale determinare la probabilità di sopravvivenza dell'individuo nella generazione successiva. La ricerca dell'individuo migliore sarà fatta cercando il cromosoma (stringa) con fitness maggiore.

Nel nostro caso, per ogni individuo della popolazione si procederà a stimare i parametri effettuando una regressione lineare sul campione di dati inserito nel training-set. Il passo successivo corrisponde al calcolo degli indici sul validation-set.

Il valore di questi indicatori, normalizzati sull'intera popolazione in un intervallo compreso tra 0 e 2, costituisce il valore attribuito dalla nostra funzione di fitness.

Una volta determinati i valori di fitness dei singoli individui, si procederà a manipolare geneticamente questi cromosomi.

È questa la cosiddetta fase della selezione: si costruisce una popolazione intermedia che, una volta applicate le procedure di clonazione, crossover (ricombinazione) e mutazione, fornirà la popolazione della generazione successiva.

Gli individui cui corrisponde un valore di fitness migliore dovrebbero essere selezionati con una probabilità maggiore, nel pieno rispetto della teoria della selezione naturale.

Il procedimento di selezione risulta di fondamentale importanza poiché è lo strumento attraverso cui si cerca di convergere verso la soluzione migliore.

Con la procedura di clonazione una frazione della popolazione, selezionata casualmente in base alle probabilità attribuite dai rispettivi valori di fitness, viene passata alla generazione successiva senza subire alcun cambiamento.

Il crossover consiste in una versione artificiale della riproduzione sessuata. Se due individui possiedono valori elevati di fitness, allora l'algoritmo esplora l'eventualità che una combinazione dei loro geni possa far scaturire una progenie con caratteristiche di fitness ancora maggiore. Gli individui con elevati valori di fitness hanno un'alta probabilità di riprodursi, mentre gli individui caratterizzati da fitness scarsa dovrebbero avere una bassa probabilità di riprodursi.

Se il crossover rappresenta il mezzo attraverso il quale spostarsi all'interno dello spazio in base alle informazioni passate, la mutazione rappresenta l'innovazione. In concreto si tratta di modificazioni casuali nella struttura genetica degli individui (ovviamente con un basso tasso di probabilità). La mutazione permette perciò di esplorare nuove aree della superficie di risposta.

Una volta applicati gli operatori biologici e generata una nuova popolazione, il processo viene ripetuto finché non converge (tutte le unità della popolazione saranno uguali) oppure finché non venga violato qualche parametro fisso di controllo (come per esempio un numero prefissato di generazioni o non venga trovato quell'individuo in grado di soddisfare l'obiettivo inizialmente prefissato).

L'intera struttura dell'algoritmo genetico è quindi rappresentata in Fig. 4.3.

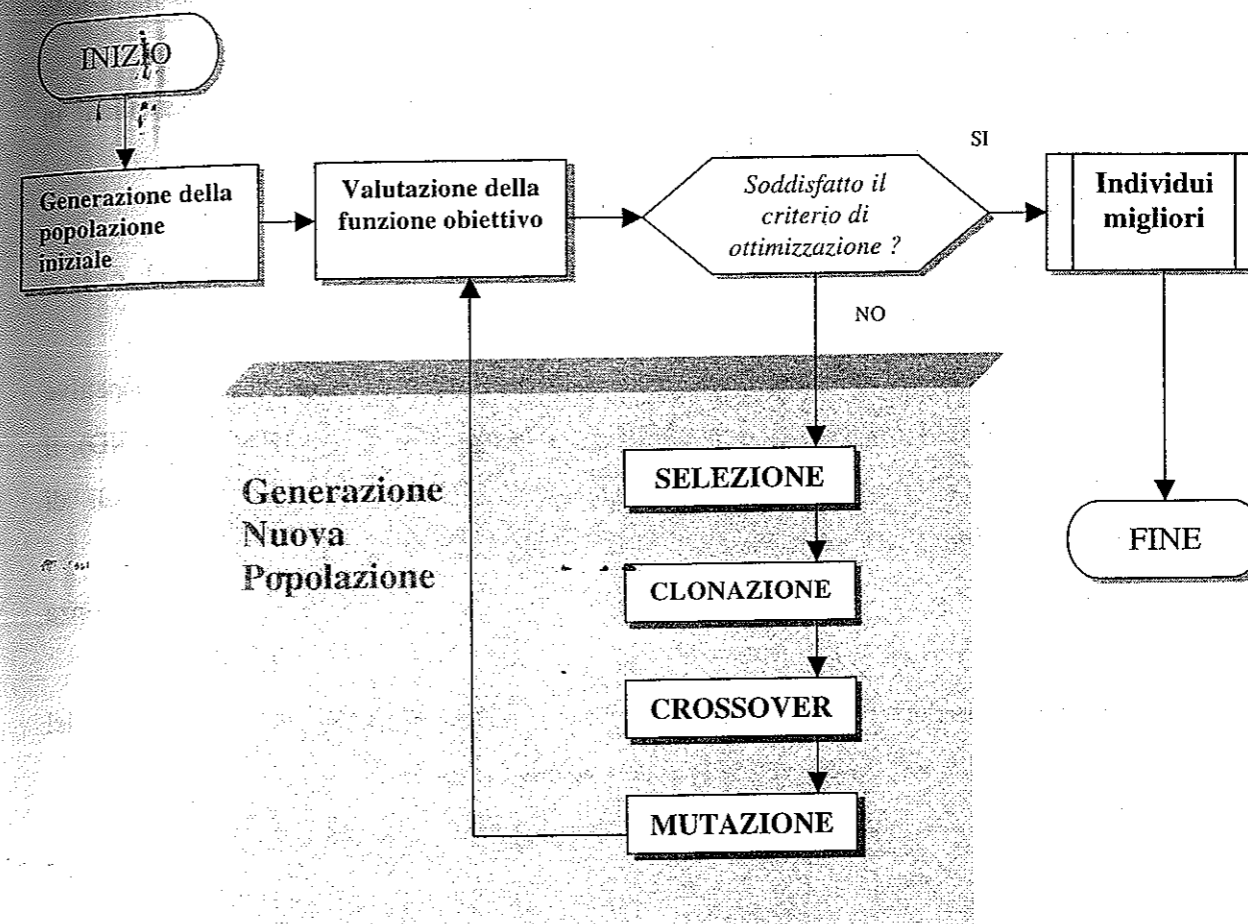


Fig. 4.3. Struttura logica di un algoritmo genetico.

Nel caso specifico in esame il criterio di ottimizzazione sarà soddisfatto soltanto quando si raggiungerà un numero di generazioni pari a 100 (che garantisce la convergenza così come verificato manualmente).

Gli altri parametri fissati nel programma sono le percentuali di mappatura dell'algoritmo. L'80% degli individui della popolazione (i migliori) vengono predisposti all'incrocio con altri individui. La probabilità di effettuare l'incrocio è ancora dell'80%. L'incrocio scelto è del tipo "single point crossover" che porta all'effettuazione di un solo taglio per ogni stringa nelle ricombinazioni. Non ci sono individui clonati tranne quelli per cui non avviene l'incrocio, che passano alla generazione successiva. Il restante 20% di individui della generazione successiva

viene generato e reinserito casualmente. La probabilità di una possibile mutazione nei cromosomi di ogni individuo è di circa il 3%.

Per meglio comprendere il funzionamento complessivo del programma in questione è meglio affidarsi al diagramma di flusso presentato in Fig. 4.4.

A questo riguardo bisogna sottolineare che:

- La normalizzazione dei dati, che avviene in un intervallo tra -1 e 1 , è necessaria per snellire le procedure e facilitare i calcoli;
- La struttura dell'algoritmo genetico viene definita attribuendo alle rispettive variabili il numero di individui che compongono ogni generazione (NIND), il numero massimo di generazioni (MAXGEN) e le altre percentuali di mappatura dell'algoritmo genetico;
- La generazione di ogni popolazione successiva segue la sequenza rappresentata in Fig. 4.3;
- La "funzione ciclo", che rappresenta il nucleo di individuazione e scelta del modello migliore, è rappresentata graficamente nella Fig. 4.5.

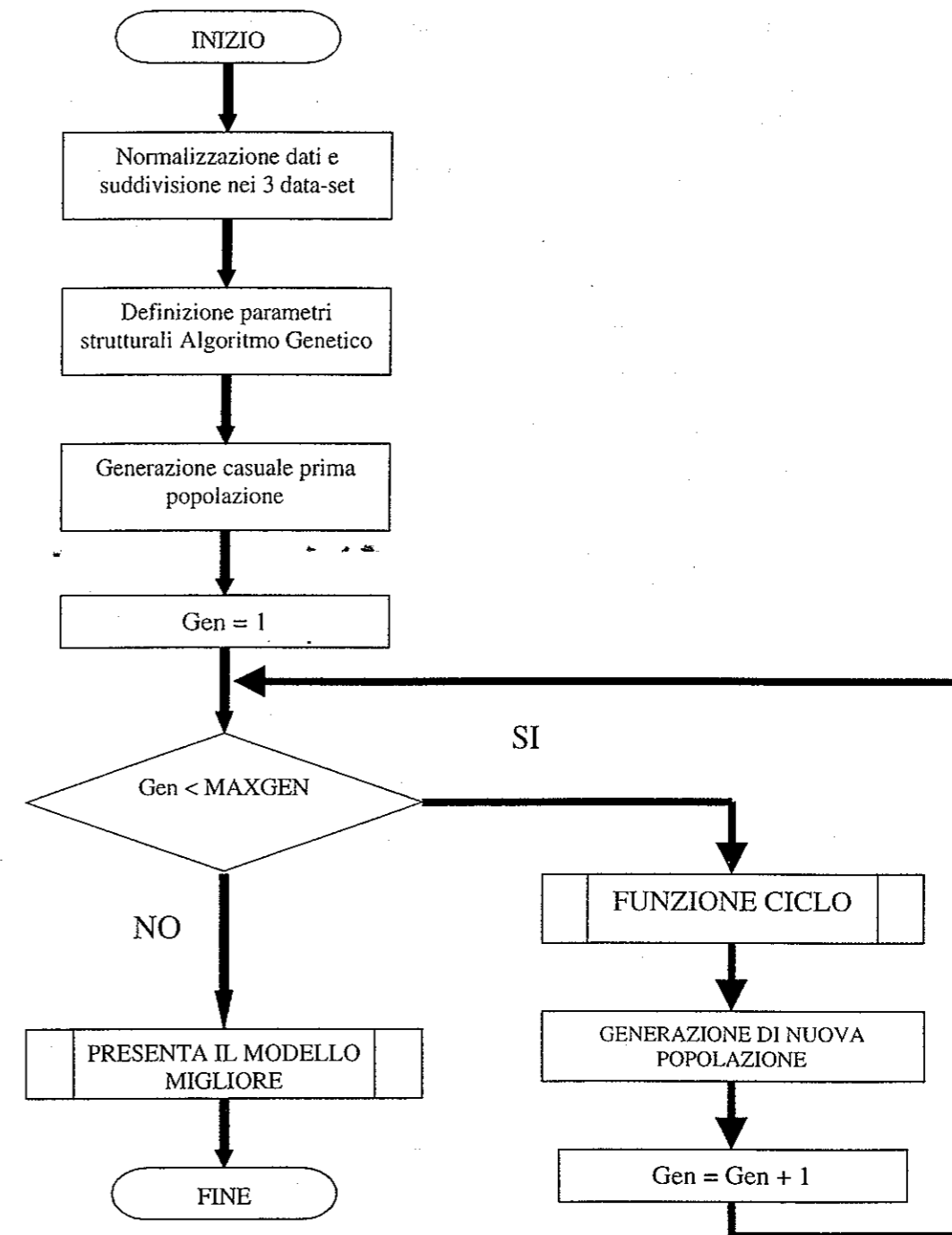


Fig. 4.4 Diagramma di flusso del programma che utilizza l'algoritmo genetico.

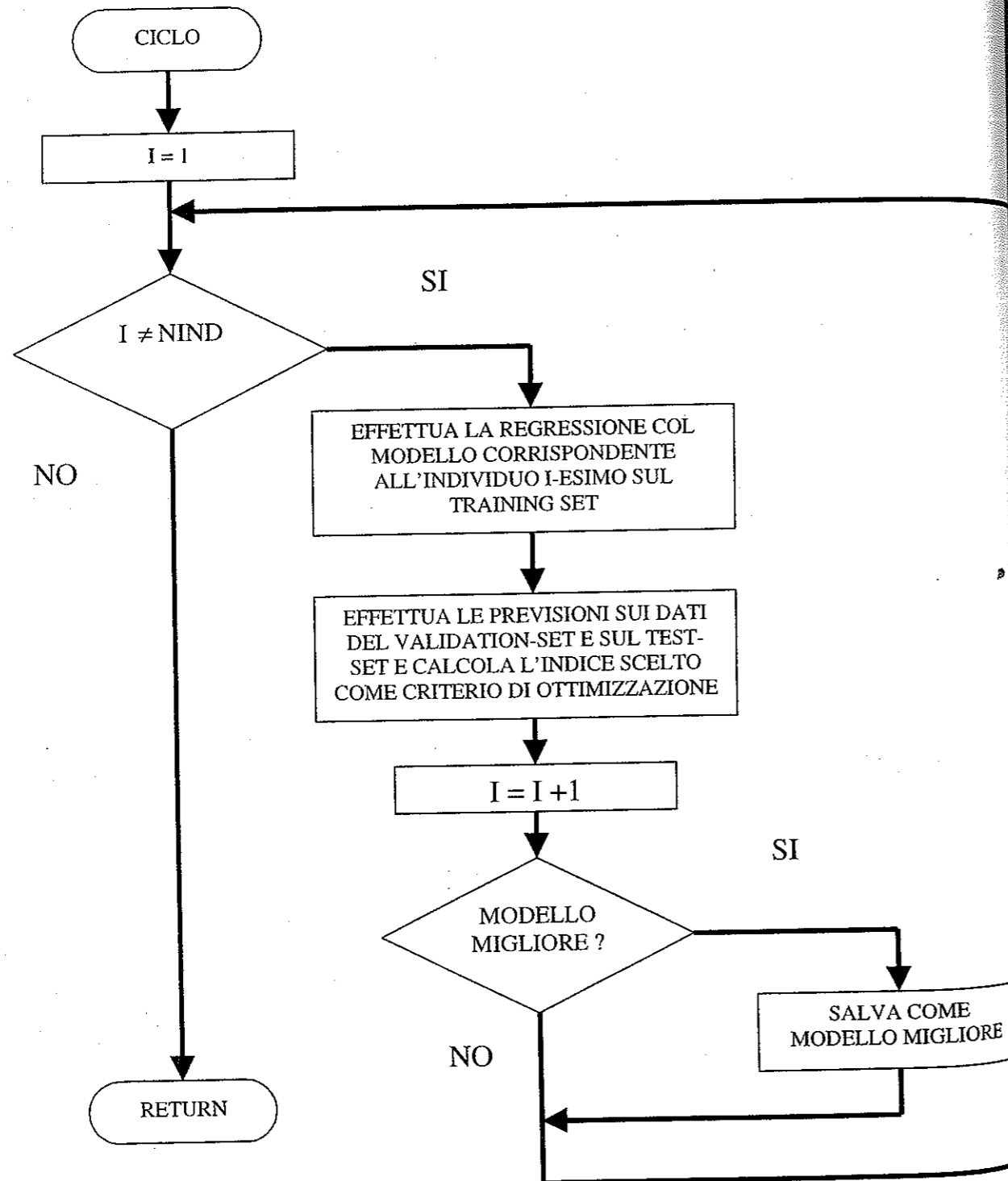


Fig. 4.5 Diagramma di flusso del sottoprogramma "Funzione Ciclo".

Nella tabella 4.4 sono riportati i risultati ottenuti con questo programma¹.

METODO	CRITERIO	RSQ _{test}	MSEP	N.V.	N _t
GALMS	RSQ	0.804	5.13	15	7
GALMS	AIC	0.808	3.87	5	-

Tabella 4.4. Risultati ottenuti con il programma GALMS (Genetic Algorithm for Linear Model Selection) sul campione di dati "choles".

Così come era prevedibile aspettarsi, otteniamo risultati decisamente migliori rispetto ai due programmi precedenti.

I problemi connessi alla forte correlazione tra le variabili esplicative, qui non influenzano in modo decisivo la dimensione del modello selezionato. Gli algoritmi genetici, lavorando in modo parallelo, ottimizzano una procedura che seleziona le variabili contemporaneamente e non in sequenza.

Il programma, inoltre, ottimizza il valore dell'indicatore scelto senza preoccuparsi del numero di variabili selezionate e della loro significatività. È questo il motivo per cui ci troviamo di fronte a modelli di dimensioni maggiori rispetto a quelli ottenuti con il programma "Forward" e di dimensioni inferiori rispetto a quelli del programma "Backward".

Comunque, l'aver implementato una procedura la cui ottimizzazione risulta essere indipendente dalla sequenza di scelta, permette di ottenere modelli più efficienti da un punto di vista predittivo.

Non possono certamente essere definiti sbalorditivi i risultati ottenuti con il criterio RSQ (soprattutto se confrontati con quelli ottenuti con il medesimo criterio

¹ Siccome questo tipo di procedura impiega parecchie risorse in termini di tempo, non si è sperimentata l'analisi per tutti gli indicatori utilizzati in precedenza, ma soltanto per i più rappresentativi. L'AIC si è rivelato un indicatore molto adatto (in quanto ha sempre ottenuto un MSEF più basso) a questo tipo di analisi.

col programma "Forward"), ma certamente ottimi sono i risultati ottenuti con il criterio *AIC*.

Si ottiene, infatti, un modello con 5 variabili esplicative tutte significative con uno scarto quadratico previsivo medio inferiore di oltre mezzo punto rispetto a tutti i modelli esaminati finora.

Tale impostazione non elimina né affronta i problemi connessi all'automazione della procedura.

Abbiamo, infatti, sempre a che fare con un programma che lavora completamente in modo automatico senza tenere in alcuna considerazione la teoria sottostante che l'utente potrebbe voler imporre o vincolare.

Ricordiamo infatti che in generale il problema della selezione del miglior modello statistico richiede pur sempre un compromesso tra l'automazione di una procedura che sia universalmente utilizzabile e il giudizio personale e soggettivo dell'utente interessato a tale analisi.

Un tentativo di risolvere questo tipo di problematica è stato fatto utilizzando un algoritmo che deriva le condizioni di ottimizzazione sulla base delle conoscenze soggettive e oggettive possedute a priori dall'utente.

Per far ciò si è impiegata la cosiddetta "Logica Fuzzy" e si è costruito un programma per la selezione di un modello che denomineremo "sfumato".

4.2. IL MODELLO "SFUMATO"

La Logica Sfumata (Fuzzy Logic) [Zade84, DuPr80] discende dalla "Teoria degli Insiemi Sfumati" [Zade65], intesa a sistematizzare la precedente idea di logica a valori infiniti, e fornisce un meccanismo per consentire ai sistemi preposti ai processi decisionali di gestire informazioni caratterizzate da un basso grado di precisione.

L'uomo è in grado di manipolare concetti imprecisi, come "abbastanza piccolo", "piuttosto grande", o "molto recente"; in concreto, la logica sfumata è un tentativo di consentire agli elaboratori elettronici di ragionare e contemporaneamente giustificare le proprie azioni in modo altrettanto flessibile. La logica tradizionale "Booleana" preclude in modo assoluto la appartenenza non totale a un insieme. La logica sfumata, invece, consentendo un'appartenenza parziale e specificando operazioni sugli insiemi basate sugli insiemi stessi così definiti, fornisce una struttura grazie alla quale operare una forma di ragionamento automatizzato molto più trasparente, e potenzialmente più raffinato.

La logica convenzionale utilizza insiemi con confini rigidi: questo significa che vi è una transizione istantanea da un insieme all'altro. Qualsiasi processo decisionale basato su questo tipo di confini deve pertanto gestire cambiamenti di stato bruschi e istantanei, che potrebbero divenire oltremodo artificiosi nella costruzione di modelli per il funzionamento del mondo reale.

La logica sfumata rimuove i confini rigidi attraverso l'uso di funzioni di appartenenza agli insiemi. Le funzioni di appartenenza indicano il grado di appartenenza a un insieme sfumato restituendo un valore che può variare tra 0 (per indicare un'appartenenza nulla) e 1 (per indicare appartenenza totale).

La Logica Sfumata, non può essere considerata, quindi, uno strumento di automazione né un algoritmo, ma piuttosto un mezzo attraverso cui poter facilitare la definizione degli ambiti di determinate forme di processo decisionale.

In tale contesto, si è pensato di costruire un indicatore (che si chiamerà *FIS*: "Fuzzy Inference System") ottenuto mediante l'utilizzo di questi criteri.

Esaminando i risultati ottenuti coi programmi "Backward", "Forward" e "GALMS", ci si può facilmente rendere conto della necessità di voler trovare un modello che presenti contemporaneamente le seguenti caratteristiche:

- 1) $R^2 \implies$ alto (tendente a 1).
- 2) Numero di variabili del modello (dimensione) \implies basso.
- 3) Variabili non significative (con $t\text{-test} < t_\alpha$) \implies basso (tendente a 0).

Solo lavorando con una logica di tipo Fuzzy si riesce a utilizzare come funzione di fitness un indicatore (FIS) che permetta di ottimizzare i tre parametri posti sopra (in antitesi tra loro) e di considerare tutte le altre valutazioni soggettive eventualmente poste dall'operatore.

Per comprendere meglio l'importanza di tutto questo si supponga di affrontare il problema della selezione delle variabili esplicative che "spieghino" il livello di altezza degli individui.

Quale giudizio di bontà si dovrebbe attribuire a un modello qualora, tra le variabili selezionate, non sia presente la variabile "altezza dei genitori"?

È ovvio che avendo la possibilità di giudicare il modello scelto mediante un indicatore che ritenga vincolante ai fini del giudizio di bontà l'inserimento di questa variabile, anche procedure automatizzate non produrrebbero gli inconvenienti generalmente a queste associabili.

Questo è il motivo per cui si ritiene doveroso implementare questo tipo di procedura.

La Logica Fuzzy si basa sulla descrizione del grado di appartenenza delle variabili a un insieme Fuzzy. Pertanto, se ritorniamo all'esempio simulato del campione di dati del file "choles", la logica sfumata si occuperà di assegnare a ogni variabile di input, lungo l'intero spazio dei valori assunti dalla variabile stessa (il cosiddetto "Universe of Discourse"), vari insiemi sfumati.

Quindi, se l'Universe of Discourse della variabile di input R^2 è l'insieme [0,1], si procederà a suddividerlo in alcuni insiemi sfumati i cui confini precisi di transizione vengono rimossi (vedi a riguardo la Fig. 4.6).

Agli insiemi sfumati così definiti vengono collegati quantificatori linguistici (nel caso della Fig. 4.6 : "BASSO", "MEDIO-BASSO", "MEDIO", "MEDIO-ALTO",

"ALTO") allo scopo di rendere più agevole l'interpretazione dei risultati, oltre che per facilitare le modalità di definizione degli insiemi.

Una volta definiti gli insiemi sfumati, qualsiasi valore discreto di R^2 misurato nel mondo reale può essere convertito in termini sfumati.

La Fig. 4.6 mostra come il valore 0.15 si posiziona sull'universo definito per la variabile " R^2 ". Utilizzando le funzioni di appartenenza, si ottengono appartenenza parziali per gli insiemi "BASSO" e "MEIO-BASSO" con valori rispettivamente di 0.4 e 0.6 ($R^2(0.15) = 0.4 / \text{BASSO}, 0.6 / \text{MEDIO-BASSO}$).

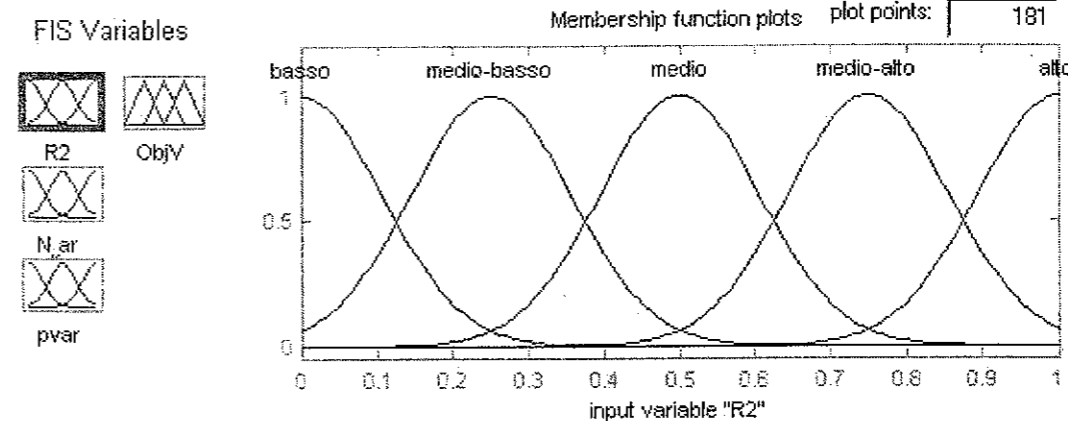


Fig. 4.6. Universe of Discourse di R^2 e relativi insiemi sfumati, coi rispettivi quantificatori linguistici.

Lo stesso procedimento deve essere applicato alle altre variabili di input (numero di variabili del modello e variabili non significative) e alla variabile di output (l'indicatore di bontà del modello che restituisce un valore compreso tra 0 e 1). I rispettivi spazi di valori assunti e gli insiemi sfumati relativi sono rappresentati nelle Figg. 4.7 e 4.8 e 4.9.

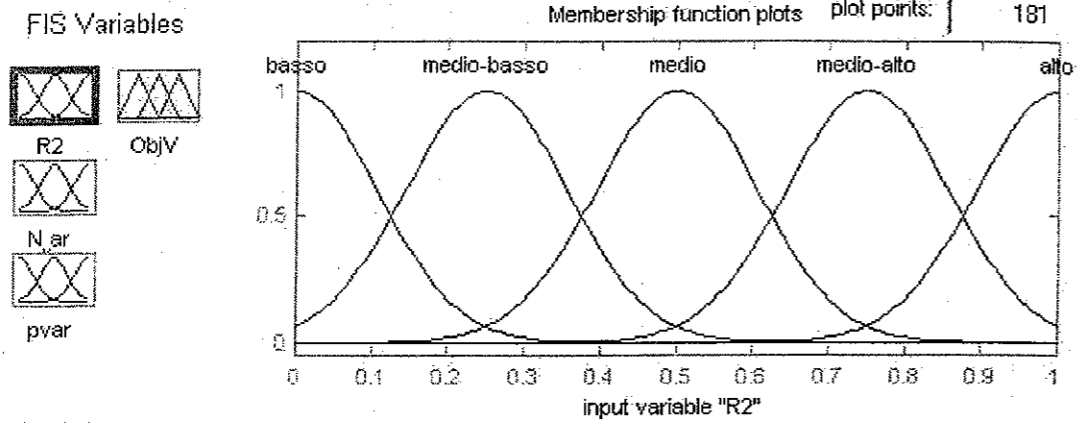


Fig. 4.7. Universe of Discourse di N_var (numero di variabili complessive del modello) e relativi insiemi sfumati, coi rispettivi quantificatori linguistici.

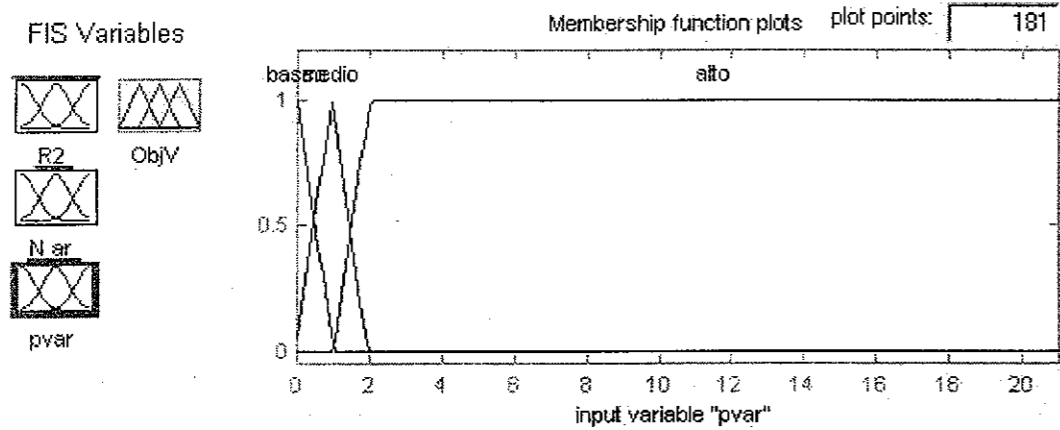


Fig. 4.8. Universe of Discourse di pvar (numero di variabili non significative in base al test-t) e relativi insiemi sfumati, coi rispettivi quantificatori linguistici.

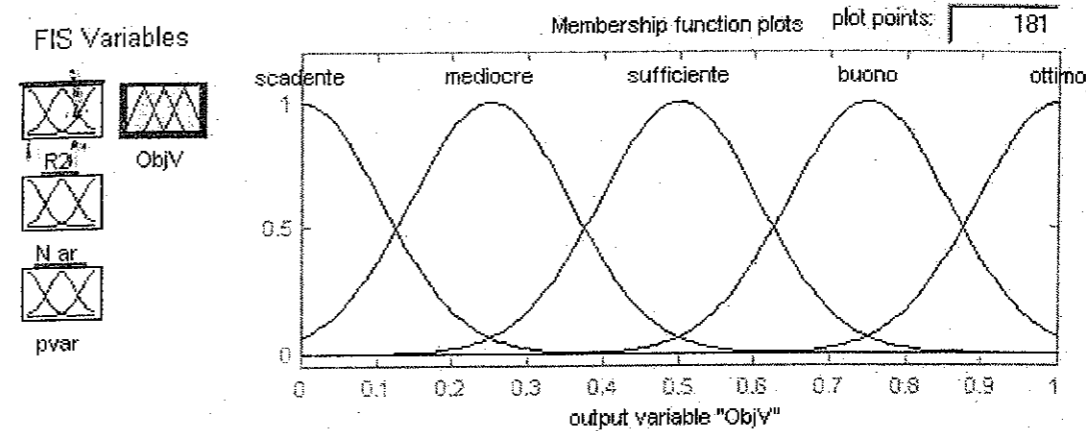


Fig. 4.9. Universe of Discourse di ObjV (variabile di output indicatore di bontà del modello) e relativi insiemi sfumati, coi rispettivi quantificatori linguistici.

Il processo di inferenza è poi definito tramite l'uso di una base di regole di produzione (relazioni logiche) nella forma "SE X E/O Y E/O ... ALLORA Z". Questa metodologia è direttamente compatibile con le basi di regole comunemente utilizzate per i sistemi esperti (con la sola differenza che nei sistemi esperti gli input devono ricadere sempre in un insieme definito e distinto che possa rendere vera la regola mentre in quelli sfumati possono appartenere a più di un insieme, rendendo vere determinate regole, ma "in una certa misura").

Per esempio: "Se R^2 è alto" E "numero di variabili è basso" E "numero di variabili non significative è nullo" ALLORA "il modello è ottimo".

Dopo aver definito l'insieme complessivo delle 75 regole ($5 \times 5 \times 3$: prodotto del numero di quantificatori linguistici utilizzati per le variabili di input) MATLAB restituisce una struttura (Fis.fis) che ricevendo in input il valore discreto di ciascuna delle tre variabili di input, restituisce il valore (come detto compreso tra 0 e 1) di confronto della bontà del modello.

Tale valore costituisce il criterio di scelta delle variabili esplicative del modello e verrà di seguito chiamato FIS (Fuzzy Inference System).

Il processo inferenziale tipico dei sistemi sfumati, quindi, consentendo di effettuare ragionamenti sulla base di tali appartenenze parziali e integrato da un modello di "desfumatura", tende a far ritornare il sistema a uno stato di valori discreti.

Applicando tale sistema ai tre programmi visti nei paragrafi precedenti si sono ottenuti i risultati mostrati nella tabella 4.5.

METODO	CRITERIO	RSQ _{test}	MSEP	N.V.	N _t
Backward	FIS	0.799	5.27	13	4
Forward	FIS	0.757	4.24	2	-
GALMS	FIS	0.772	4.12	2	-

Tabella 4.5. Risultati ottenuti con il criterio FIS sul campione di dati "choles".

Molto interessante è l'analisi dell'andamento dell'indice FIS al variare degli indicatori utilizzati per costruirlo (per poter rappresentare tale andamento su un grafico a tre dimensioni si è dovuto porre il numero di variabili complessive del modello costante, vedi Fig. 4.10).

Il ciclo iterativo dei programmi ha come criterio di stop l'eventuale peggioramento del modello e, se si parte dal modello nullo (programma "Forward"), la forte correlazione presente tra le variabili esplicative tende a farsi sentire immediatamente (essendo l'indice stesso costruito su tali informazioni) peggiorando il modello stesso. In questo modo il valore dell' R^2 non fa in tempo a crescere che l'algoritmo ha terminato il ciclo. Questo è il motivo per cui l' R^2_{test} (RSQ_{test}) è relativamente basso nel programma in questione rispetto al programma "Backward".

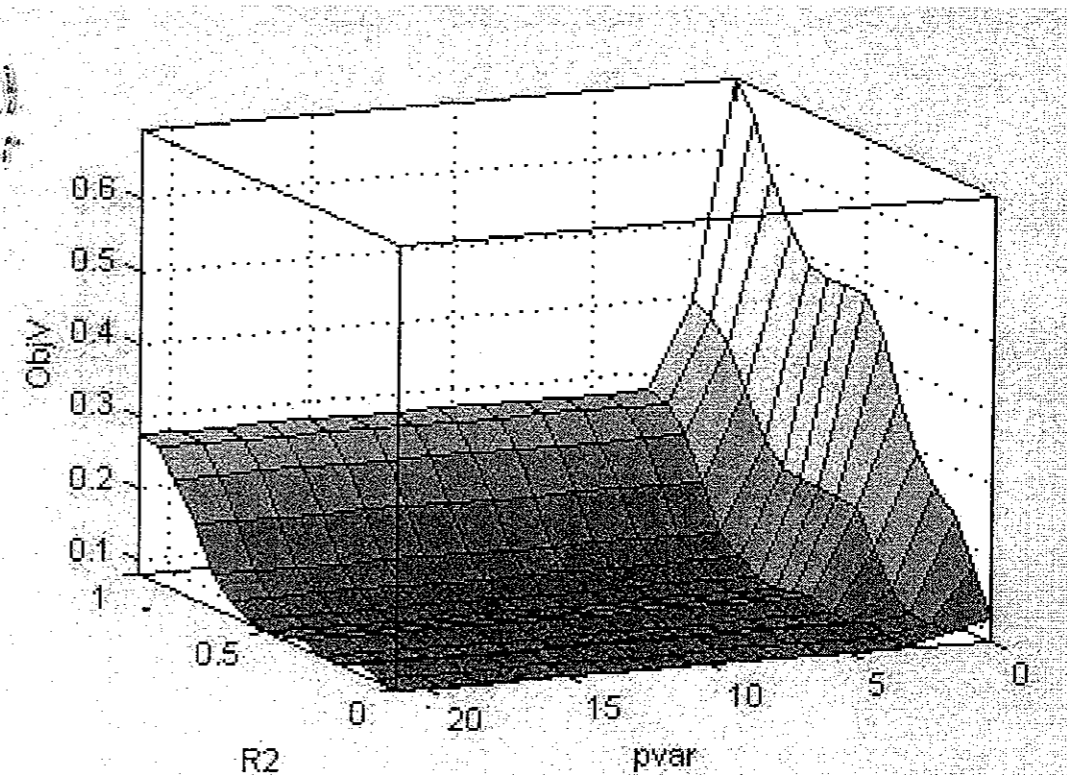


Fig. 4.10. Andamento dell'indice FIS al variare di R^2 e pvar (costante N_{var} pari al suo valore medio).

Viceversa, ma sempre per lo stesso motivo, utilizzando il programma Backward (partendo quindi dal modello completo con l'eliminazione di volta in volta di una variabile) l' R^2 già alto non scende così velocemente da ridurre immediatamente la bontà complessiva del modello (anzi, aumenta durante le prime iterazioni). In tal caso, infatti, l'algoritmo effettua ben 8 ($21 - 13 = 8$) cicli prima di fermarsi. In Fig. 4.11 è possibile notare l'andamento dell'indice al variare di R^2 e di N_{var} (in questo caso si è posto costante il numero di parametri non significativi) e in Fig 4.12 l'andamento dello stesso indice al variare di pvar e N_{var} (costante R^2).

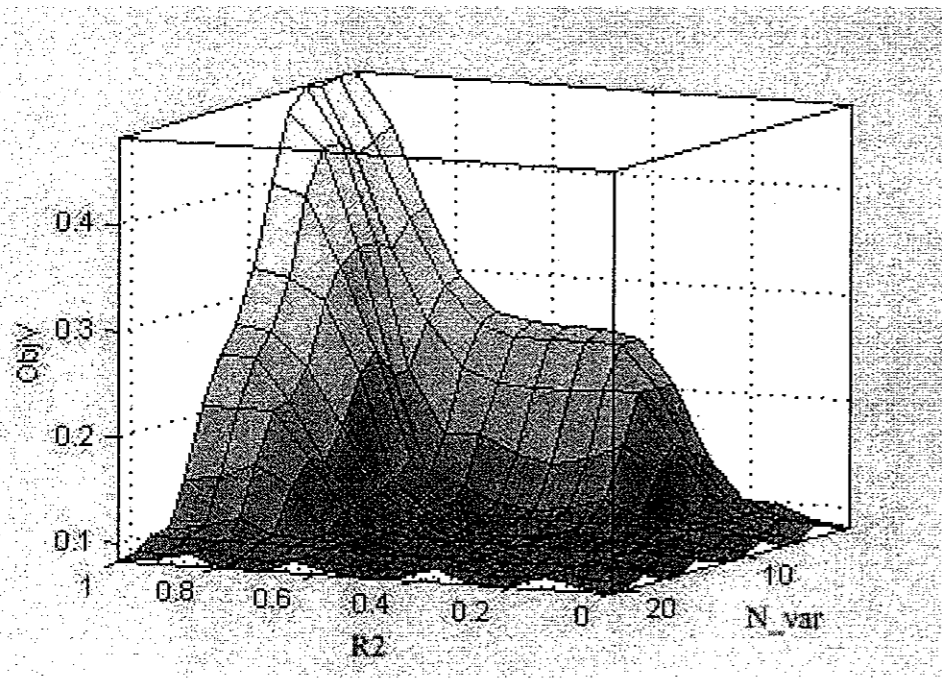


Fig. 4.11. Andamento dell'indice FIS al variare di R^2 e N_{var} (costante $pvar$ pari al suo valore medio).

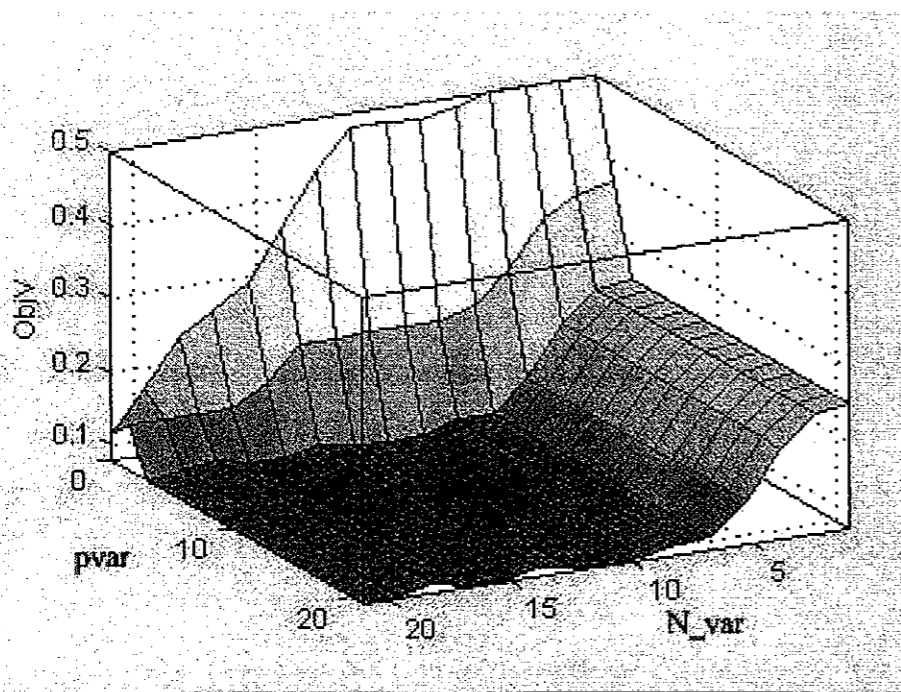


Fig. 4.12. Andamento dell'indice FIS al variare di $pvar$ e N_{var} (costante R^2 pari al suo valore medio).

Non sono certamente i risultati ottenuti con queste due metodologie quelli più interessanti dal punto di vista tecnico in quanto gli stessi risultati si erano già ottenuti con altri indicatori (almeno per quanto concerne il programma "forward", ma risultati molto simili si erano ottenuti anche nel programma "Backward").

Merita invece un approfondimento il risultato ottenuto con l'Algoritmo Ibrido².

Così come la maggior parte delle applicazioni pratiche che utilizzano sistemi ibridi, anche questo tipo di algoritmo è stato appositamente studiato per trarre vantaggio dai rispettivi punti di forza delle due metodologie.

In questo caso si è così tentato di risolvere il duplice problema che sta alla base delle applicazioni presentate nel primo paragrafo di questo capitolo: la necessità di avere un risultato che non dipendesse dalla sequenza di scelta delle variabili esplicative (a questo pone rimedio l'algoritmo genetico) e la necessità di evitare di costruire procedure completamente automatizzate che non tenessero in alcuna considerazione le valutazioni aprioristiche dell'utente (a ciò provvede la Logica Fuzzy).

I risultati così ottenuti sono molto buoni in quanto la dimensione del modello selezionato è bassa (due sole variabili selezionate entrambe significative), R^2_{test} è alto (0.772) e lo scarto quadratico medio previsivo è basso (4.12).

Da un punto di vista previsivo, forse è da preferire il risultato ottenuto con il programma "GALMS" applicato al criterio AIC, ma se l'obiettivo è quello di scegliere un modello che risulti semplice senza rinunciare alle buone capacità previsive, la scelta deve cadere su questo criterio (FIS).

D'altronde, come si è già discusso nel primo capitolo, non si può universalmente stabilire se una delle due procedure è migliore dell'altra. La scelta dovrà effettuarsi in base alle necessità e agli scopi della ricerca.

² Gli "Intelligent Hybrid System" o sistemi intelligenti ibridi [GoKh95] utilizzano più tecnologie intelligenti contemporaneamente. Nel caso della simulazione sopra esposta si è utilizzata la Logica Fuzzy contemporaneamente agli Algoritmi Genetici.

4.3. CONCLUSIONI

In questo capitolo sono state presentate alcune possibili soluzioni alle problematiche inerenti alla selezione del modello statistico lineare multivariato orientato alla previsione.

Mentre la stima dei parametri associati al modello statistico prescelto avviene utilizzando gli approcci classici, per quanto concerne la determinazione del miglior set di regressori (quanti e quali) da fornire come input al modello si sono confrontati gli approcci classici con algoritmi "intelligenti".

Si è visto che le tecniche classiche qui implementate ("backward elimination" e "forward selection") soffrono di imposizioni strutturali (dipendenza dalla sequenza) e vincoli di carattere fisico la cui automazione li rende criticabili sotto più punti di vista.

Gli Algoritmi Genetici rimuovono le imposizioni strutturali (e con esse si eliminano i problemi connessi alla dipendenza dalla sequenza) valutando ogni soluzione rispetto alle proprie capacità di descrivere la soluzione del problema in esame sottoposta a un criterio di selezione e propagata alla "iterazione" successiva per mezzo di operatori che emulano la selezione genetica. Il processo evolutivo converge verso la migliore soluzione entro un criterio di stop preimposto.

Tutti gli algoritmi proposti sono stati testati su un particolare set di dati ("choles") che ha mostrato come le tecniche computazionali non classiche risultino effettivamente più efficienti (non soltanto teoricamente).

Infine si è implementato un esempio semplice di sistema ibrido affiancando all'Algoritmo Genetico la teoria proposta dalla Logica Fuzzy al fine di rimuovere anche i problemi connessi all'automazione delle tecniche.

I risultati ottenuti sono stati interessanti nonostante si sia sperimentato un algoritmo piuttosto semplice di cui si rende necessario un approfondimento per un'analisi più accurata.

È necessario fare un'ultima considerazione per quanto riguarda il tempo impiegato nell'esecuzione dei programmi. Lavorando con un processore Pentium II a 400 Mhz, il tempo computazionale necessario per eseguire tutti i 2.097.151 ($2^{21}-1$) possibili modelli sarebbe di circa 116 ore e 30 minuti (circa cinque modelli per ogni secondo).

Il programma "Backward" impiega per eseguire l'intero programma circa un minuto e 30 secondi, il programma "Forward" impiega 12 secondi mentre il programma "GALMS" impiega circa 23 secondi per eseguire una generazione di 100 individui (quindi complessivamente impiega circa 38 minuti).

È evidente quindi che lavorando con gli algoritmi genetici il tempo complessivamente impiegato è notevolmente maggiore rispetto a quello impiegato con gli altri tipi di programmi.

Questo può disincentivare l'utilizzo di tali tecniche che, comunque, complessivamente rispondono meglio alle attese del ricercatore essendo migliori in termini di efficienza e robustezza. Inoltre, portando a convergenza l'algoritmo, il tempo impiegato risulta ridotto di circa 200 volte rispetto a quello impiegato per la verifica di tutti i modelli (tempo richiesto applicando la strategia "All possible" analizzata nel primo capitolo).

5. SELEZIONE DEL MODELLO PER SERIE STORICHE

L'analisi statistica delle serie storiche costituisce un momento fondamentale della riflessione scientifica sull'evoluzione dei fenomeni del mondo reale.

Una serie storica $\{x_t, t = 1, 2, \dots, n\}$ è una successione ordinata di numeri reali che misura un certo fenomeno X , esaminato con riguardo alla sua evoluzione rispetto al tempo t .

Lo studio delle osservazioni temporali è basato sul principio che la conoscenza di quanto è già avvenuto "determina" ciò che avverrà, secondo un principio generale di inerzia e di stabilità delle leggi che conosciamo e sul quale si basa la nostra vita quotidiana: dalle decisioni inconscie e regolari a quelle importanti e occasionali.

Al contrario di quanto esaminato nel capitolo precedente, in cui nei modelli considerati ogni dato è denso di informazioni sulla variabile casuale che lo ha generato ma non aggiunge alcuna informazione sulla osservazione successiva, l'analisi delle serie storiche fa riferimento alla teoria dei processi stocastici per evidenziare e utilizzare la connessione che caratterizza una successione di dati temporali.

Tale impostazione è strettamente connessa a un obiettivo previsivo in quanto, se il futuro è legato al presente e al passato secondo una legge nota, divengono immediati sia la previsione sia il controllo.

Fra gli obiettivi più rilevanti per l'analisi delle serie storiche, ricordiamo:

- La previsione;
- La simulazione;
- Il controllo;
- L'analisi strutturale;
- La ricerca di dati anomali;
- L'analisi econometrica;
- L'individuazione di componenti non osservabili;
- La classificazione e la discriminazione;
- Le rappresentazioni grafiche etc...

Per questo, la letteratura offre angolature e approcci differenti.

Un primo orientamento privilegia i metodi di analisi della struttura di dipendenza temporale senza un legame esplicito con i modelli statistici: in tale ambito possiamo comprendere le analisi sulle differenze successive, i test non parametrici, le analisi di correlazione, le analisi spettrali. In questa ottica, lo studio dei modelli statistici è vantaggioso per generare dati utili per l'individuazione di andamenti tipici delle funzioni prescelte.

Un secondo approccio introduce una prefissata classe di modelli statistici, convenientemente generale, di cui si analizzano le proprietà in termini di correlazione, spettro, etc... Il modello è, quindi, il punto centrale della ricerca anche se non costituisce il fine ma uno strumento per obiettivi statistici.

Da alcuni autori [Picc74] lo studio delle serie storiche viene introdotto soprattutto per migliorare la spiegazione dei dati (in particolare i residui) dopo aver utilizzato un modello esplicativo per le osservazioni. Tale approccio, ormai superato nelle presentazioni più recenti [Har93, Goli94], concepisce l'analisi delle serie storiche in termine di legami fra gli errori e mira a introdurre varianti che rendano accettabile il modello classico di regressione.

Infine si possono studiare le serie storiche in funzione di un obiettivo specifico, per esempio, come verrà proposto nei successivi paragrafi, per la previsione, esaminando metodi, modelli e proposte di analisi in funzione dello scopo stesso. Naturalmente sarà la natura dell'obiettivo a rendere questo approccio molto generale o molto limitante.

In questo capitolo verranno presentati alcuni esempi di simulazione di modelli basati su serie storiche e si cercherà di affrontare la problematica inerente alla loro determinazione mediante l'uso di algoritmi "intelligenti".

Tutti i modelli presentati vengono definiti esclusivamente come processi stocastici¹.

Se $X(t)$ è il valore del processo stocastico al tempo t ($t = 1, \dots, n$) un processo autoregressivo (AR) è definito dalla relazione che collega $X(t)$ ai valori precedenti:

$$(5.1) \quad X(t) = \Phi_1 X(t-1) + \Phi_2 X(t-2) + \dots + \Phi_p X(t-p) + \varepsilon_t$$

dove ε_t ($t = 1, \dots, n$) è una successione di variabili casuali incorrelate con media 0 e varianza costante σ_ε^2 .

La scelta di un processo autoregressivo potrebbe perciò essere giustificata dall'esigenza di "spiegare" $X(t)$ collegandolo ai valori precedenti con un sistema di pesi $\Phi_1, \Phi_2, \dots, \Phi_p$.

Tale procedura consente un rapido "aggiornamento" delle previsioni man mano che si aggiungono nuove osservazioni alla serie.

¹ Ogni variabile in un processo stocastico è una variabile casuale e le osservazioni evolvono nel tempo in base a determinate distribuzioni di probabilità. L'importanza dei processi stocastici in tali applicazioni è riconosciuta universalmente. Oggi, infatti, non è più consentito trattare con rigore una serie storica senza concepirla, nello stesso tempo, come "realizzazione" finita di un processo stocastico la cui struttura interna è e rimane fondamentale ignota al ricercatore.

Fissato p , il problema di stimare in una serie storica il vettore $\Phi = (\Phi_1, \Phi_2, \dots, \Phi_p)$ viene risolto con il metodo dei minimi quadrati o stima di massima verosimiglianza così come già discusso nel modello lineare.

Il problema fondamentale, pertanto, è quello di fissare l'ordine p del processo AR che si ritiene ben adeguato a rappresentare la realtà dei fenomeni.

Per semplificare la notazione utilizzata indicheremo l'ordine del processo tra parentesi (per esempio un modello autoregressivo del quinto ordine verrà indicato con $AR(5)$) e indicheremo il numero dei parametri diversi da zero con un apice (se il modello presenta solo tre dei cinque parametri diversi da zero verrà indicato pertanto con la notazione $AR^3(5)$).

Il problema quindi non sarà soltanto quello di determinare l'ordine del processo, ma anche quello di determinare il numero di elementi del vettore $\Phi = (\Phi_1, \Phi_2, \dots, \Phi_p)$ diversi da 0.

Pertanto il modello:

$$(5.2) \quad X(t) = \Phi_1 X(t-1) + \Phi_2 X(t-2) + \Phi_7 X(t-7) + \varepsilon_t$$

identifica un processo autoregressivo del settimo ordine e viene così indicato: $AR^3(7)$.

Nonostante queste differenze di notazione, in termini di problematica generale, l'analisi del modello AR non si discosterà affatto da quella riguardante il modello lineare vista nei precedenti capitoli.

Si dovranno risolvere, quindi, sia il problema del metodo utilizzato per la selezione dei parametri, sia quello inerente il criterio da applicare (con il metodo scelto).

In questa analisi, pertanto, allo scopo di non ripetere quella precedentemente svolta, non verrà analizzato un caso particolare come "choles", ma si simuleranno

delle serie storiche e si tenterà mediante un programma geneticamente evoluto di dare una risposta ai quesiti posti.

Il passo successivo è la simulazione di processi ARMA (autoregressivi - media mobile).

Una classe di processi stocastici può formarsi introducendo ai modelli AR un numero finito di ritardi nella media mobile.

Il modello può essere così rappresentato:

$$(5.3) \quad X(t) = \Phi_1 X(t-1) + \dots + \Phi_p X(t-p) + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q}$$

dove (p, q) rappresenta l'ordine del modello ARMA.

Anche per questa classe di modelli verrà utilizzata la notazione sopra presentata con le stesse osservazioni.

Il modello:

$$(5.4) \quad X(t) = \Phi_1 X(t-1) + \Phi_5 X(t-5) + \varepsilon_t + \theta_2 \varepsilon_{t-2}$$

viene indicato con la notazione ARMA^{2/1}(5,2).

5.1. IL PROGRAMMA GATS

Lavorando con MATLAB si pone il problema di esaminare come questi modelli vengono trattati in questo ambiente.

Il "System Identification Toolbox" è una routine di sistema di MATLAB che contiene le più comuni tecniche di stima dei parametri di modelli lineari basati su serie storiche permettendone anche la simulazione.

La notazione generalmente valida in questo campo è la seguente:

$$y = Gu + He$$

dove y è il segnale di output, u quello di input, e il disturbo (il sistema è graficamente ben rappresentato in Fig. 5.1). Tale modello dice che $y(t)$ è la somma di un contributo che viene da $u(t)$ e di un contributo che viene da $e(t)$. G rappresenta la dinamica del sistema, cioè la cosiddetta "transfer function" dall'input all'output, mentre H si riferisce alle proprietà dell'errore (e per questo viene denominato "noise model").

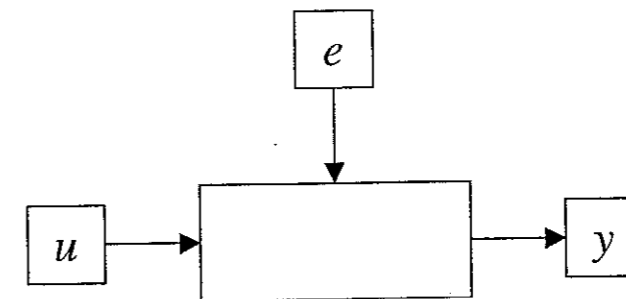


Fig. 5.1. Segnale di input u , segnale di output y , segnale di disturbo e .

Più dettagliatamente si può scrivere:

$$(5.5) \quad y(t) = G(q)u(t) + H(q)e(t)$$

dove :

$$G(q)u(t) = \sum_{k=1}^{\infty} g(k)u(t-k)$$

e :

$$G(q) = \sum_{k=1}^{\infty} g(k)q^{-k} ;$$

$$q^{-1}u(t) = u(t-1) ;$$

I numeri $\{g(k)\}$ sono denominati "impulse response" del sistema. Ovviamente $g(k)$ rappresenta l'output del sistema al tempo k se l'input è un singolo "impulso" al tempo zero.

Inoltre $e(t)$ è l'errore bianco (white noise) con varianza λ e in termini del suo (auto) spettro $\Phi_v(\omega)$ è così definito:

$$(5.6) \quad \Phi_v(\omega) = \lambda |H(e^{i\omega})|^2.$$

Piuttosto che specificare le funzioni G e H in funzione della variabile di frequenza ω , si possono descrivere in termini di q^{-1} , costruendo così la rappresentazione polinomiale utilizzata dalla routine in questione.

Tutti i modelli utilizzabili (AR, ARX, ARMA, ARMAX, ARARX, ARARMAX, ...) si possono rappresentare come casi speciali del modello generale :

$$(5.7) \quad A(q)y(t) = \frac{B(q)}{F(q)}u(t-nk) + \frac{C(q)}{D(q)}e(t)$$

se si considera un solo input e

$$(5.8) \quad A(q)y(t) = \sum_{i=1}^{nu} \frac{B_i(q)}{F_i(q)}u_i(t-nk_i) + \frac{C(q)}{D(q)}e(t)$$

se si considerano nu input².

$$\text{dove: } A(q) = 1 + a_1q^{-1} + \dots + a_{na}q^{-na} ;$$

$$B(q) = b_1 + b_2q^{-1} + \dots + b_{nb}q^{-nb+1} ;$$

$$C(q) = 1 + c_1q^{-1} + \dots + c_{nc}q^{-nc} ;$$

$$D(q) = 1 + d_1q^{-1} + \dots + d_{nd}q^{-nd} ;$$

$$F(q) = 1 + f_1q^{-1} + \dots + f_{nf}q^{-nf} .$$

E na, nb, nc, nd, nf corrispondono agli ordini dei rispettivi polinomi.

Pertanto se si desidera costruire un modello AR, la rappresentazione polinomiale è la seguente:

$$(5.9) \quad A(q)y(t) = e(t)$$

(ottenuta dalla precedente ponendo uguale a 0 le variabili nb, nc, nd, nf).

Mentre se si vuole costruire un modello ARMA la rappresentazione in termini polinomiali è:

² In questo caso si ha la cosiddetta struttura "Multi-Input-Single-Output".

$$(5.10) \quad A(q)y(t) = C(q)e(t)$$

(ottenuta dal modello generale uguagliando a 0 le variabili nb, nd, nf).

Esaminato il tipo di notazione utilizzato in MATLAB, viene ora presentata la funzione implementata in questa Toolbox per stimare i parametri dei modelli: la "PEM" (Parametric Estimating Models).

Questa funzione viene utilizzata per stimare i parametri del modello lineare generale con struttura "Multi-Input-Single-Output" sopra esaminato, con la procedura di ottimizzazione di Gauss-Newton (brevemente descritta nel capitolo secondo) basata, appunto, su algoritmi di ricerca dei minimi quadrati.

La sintassi utilizzata è la seguente:

$$th = PEM(z, nn, index, maxiter, tol, lim, maxsize, T, 'trace')$$

dove:

z : è una matrice contenente i dati di input-output $[y \ u]$;

nn : è un vettore contenente $[na \ nb \ nc \ nd \ nf \ nk]$; nn può essere sostituito con la struttura del modello considerato in formato "theta" e un valore iniziale di stima;

$index$: è un vettore riga contenente gli indici dei parametri da stimare (automaticamente considera tutti i parametri);

$maxiter$: questa variabile indica il numero massimo di iterazioni da effettuarsi durante la ricerca del minimo della norma della procedura di Gauss-Newton, utilizzata per la stima dei parametri (automaticamente è posta pari a 10);

tol : rappresenta il limite della norma di Gauss-Newton, raggiunto il quale la procedura si arresta (automaticamente è pari a 0.01);

lim : questa variabile indica il modo in cui il criterio dei minimi quadrati deve essere modificato per ottenerne uno più robusto (automaticamente viene posto pari a 1.16);

$maxsize$: costituisce il numero massimo di elementi della matrice formata dalla funzione (per le macchine di dimensioni ridotte è posta pari a 4096);

T : assegna l'intervallo di definizione della scala di frequenza (automaticamente è pari a 1);

'trace': se viene inserita questa variabile sono fornite all'utente informazioni ulteriori circa il processo di ricerca a ogni iterazione della procedura;

th : costituisce il dato di ritorno della funzione e rappresenta la struttura del modello con la stima dei parametri associati e la matrice di covarianza, il tutto in formato "theta".

Prima di procedere all'esame del programma per la determinazione dei parametri dei modelli basati su serie temporali, è il caso di spendere alcune parole sul processo di simulazione.

Sono state simulate alcune serie temporali (circa 200 complessivamente, ripartite tra processi AR e processi ARMA) generate da modelli il cui ordine viene stabilito casualmente in un intervallo $[1,10]$, per quanto riguarda quelli AR, e negli intervalli $([1,10],[1,10])$, per quanto riguarda i modelli ARMA.

Il termine di errore addizionato, generato casualmente da una distribuzione $N(0,1)$, viene moltiplicato per un coefficiente costante pari a 0.2.

Le serie così simulate sono ammesse poi alla fase successiva di analisi e stima solamente se il processo da esse generato risulta essere stazionario³.

Il programma utilizzato per la simulazione è stato inserito come "funzione" nel programma generale "GATS" che è descritto graficamente mediante diagramma di flusso in Figg. 5.2 e 5.3 [MiPo98a].

³ La non stazionarietà del processo aveva creato problemi in fase di specificazione del modello statistico, a volte sovrapparametrizzandolo e a volte sottoparametrizzandolo.

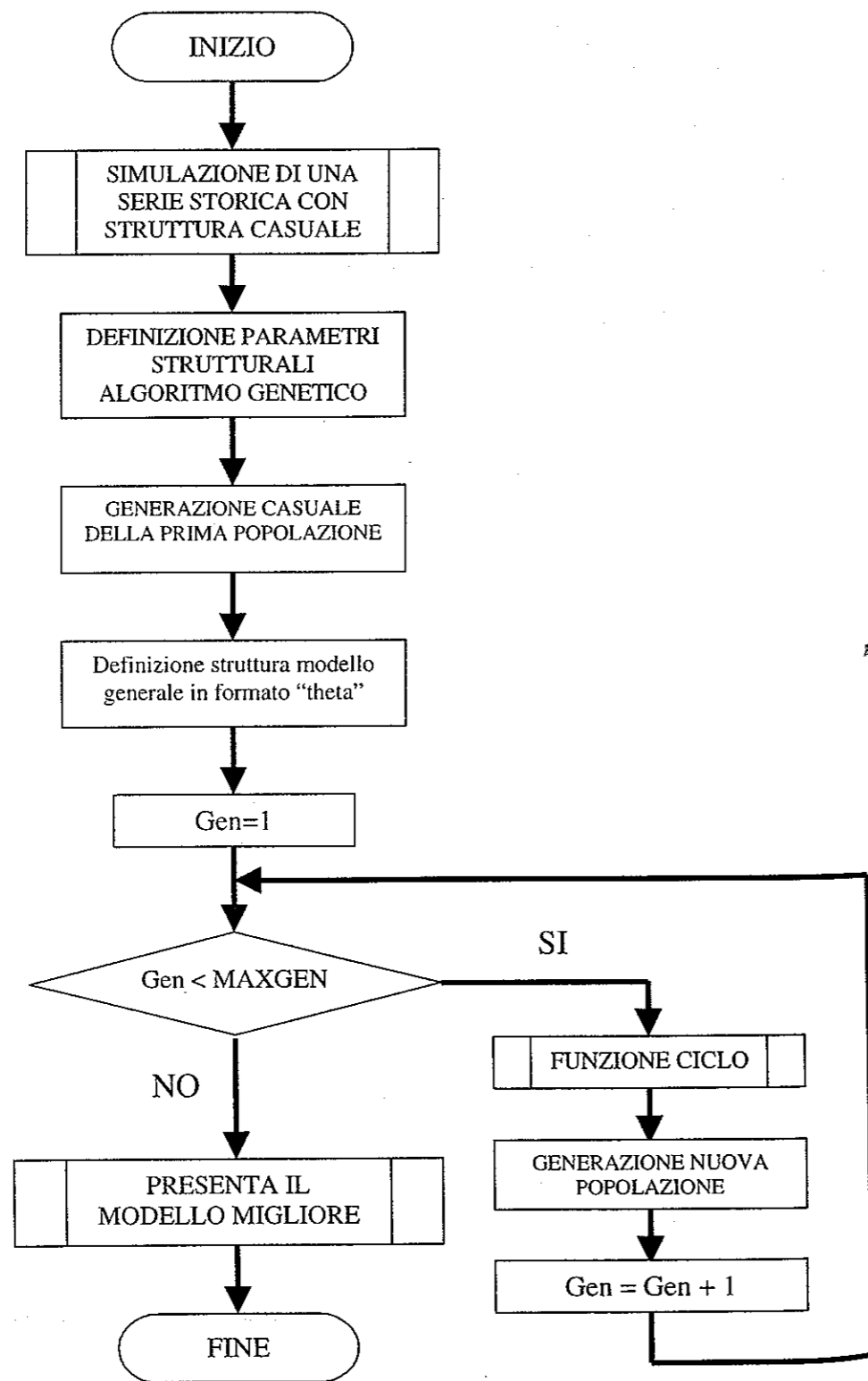


Fig. 5.2. Diagramma di flusso del programma di stima dei modelli basati su serie storiche.

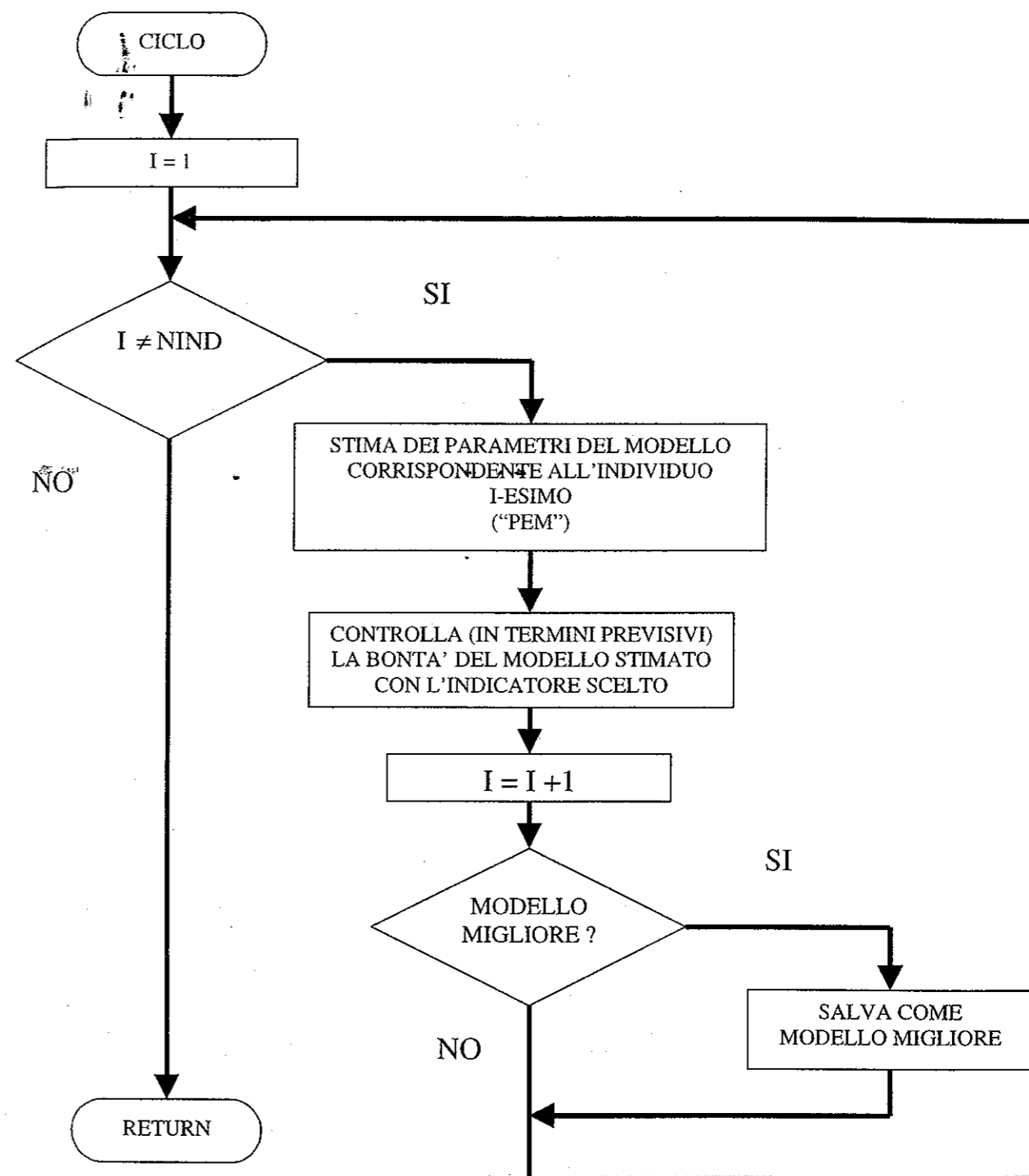


Fig. 5.3. Diagramma di flusso del sottoprogramma "Funzione ciclo".

Come si può notare, la struttura di questo programma ricalca quella vista per il modello lineare nel capitolo precedente con le variazioni opportune dettate dalle diverse esigenze.

La codifica degli individui associati alle popolazioni viene quindi modificata soltanto nella sua lunghezza: la stringa associata al modello AR è rappresentata in Fig. 5.4, mentre quella associata al modello ARMA è rappresentata in Fig. 5.5.

La lunghezza delle stringhe corrisponde all'ordine massimo associato ai due tipi di modelli. Nella stringa associata al modello ARMA le prime 10 celle individuano i ritardi nella variabile esplicativa, mentre le celle dall'undicesima alla ventesima individuano i ritardi nella media mobile.

0	1	0	0	1	1	0	0	0	0
---	---	---	---	---	---	---	---	---	---

Fig. 5.4. Rappresentazione dell'individuo utilizzato nell'algorithmo per indicare i modelli autoregressivi simulati.

AR										--	MA									
0	0	1	0	1	0	0	0	1	0	0	0	1	0	0	1	0	1	0	1	

Fig. 5.5. Rappresentazione dell'individuo utilizzato nell'algorithmo per indicare i modelli ARMA simulati.

Alla stringa rappresentata in Fig.5.4 corrisponde pertanto il modello AR³(6):

$$(5.11) \quad X(t) = \Phi_2 X(t-2) + \Phi_5 X(t-5) + \Phi_6 X(t-6) + \epsilon_t$$

Alla stringa rappresentata in Fig. 5.5 corrisponde, invece, il modello ARMA^{3,4}(9,10):

$$(5.12) \quad X(t) = \Phi_3 X(t-3) + \Phi_5 X(t-5) + \Phi_9 X(t-9) + \epsilon_t + \theta_3 \epsilon_{t-3} + \theta_6 \epsilon_{t-6} + \theta_8 \epsilon_{t-8} + \theta_{10} \epsilon_{t-10}$$

Come detto, prima si sono simulati processi autoregressivi di ordine massimo 10, poi processi autoregressivi-media mobile di ordine massimo (10,10) e infine si è tentato di "trovare" la struttura del modello sottostante alla serie storica mediante l'uso di un Algoritmo Genetico.

La scelta del modello più appropriato è stata fatta utilizzando l'indicatore AIC in termini previsivi.

Per far ciò si è suddiviso il campione di dati generato (l'intero "data-set") in due gruppi distinti:

- Il training set contenente l'80% dell'intero campione;
- Il test set contenente il restante 20%.

Al primo è stata applicata la "PEM" per stimare i parametri del modello corrispondente all'individuo i-esimo della popolazione, sul secondo si è calcolato l'AIC previsivo (una volta stimato il modello è stato infatti possibile prevedere i valori futuri).

Si è scelto di adottare l'AIC in quanto complessivamente si era mostrato un buon indicatore in tutte le situazioni in cui era stato utilizzato (per l'analisi critica dei vari indicatori implementabili come criteri di scelta del modello si vedano, infatti, i capitoli precedenti), e inoltre è un indice di raffronto molto utilizzato per la valutazione dei modelli statistici basati su serie temporali.

Ricordiamo che l'AIC è definito come:

$$(5.13) \quad AIC(p) = n \log(S_p^2) + 2(p);$$

dove:

p = numero dei parametri del modello;

$$S_p^2 = \frac{\sum_1^n (\hat{E}_i)^2}{n-p-1};$$

n = numero di osservazioni;

$$\hat{E}_i = Y - XB;$$

$$B = (X'X)^{-1} X'Y \text{ (stimatore dei minimi quadrati).}$$

Per quanto riguarda la struttura parametrica dell'Algoritmo Genetico e il criterio di ottimizzazione, nulla è mutato rispetto al modello lineare esaminato in precedenza: uguali sono le dimensioni della popolazione (NIND), uguali i cicli iterativi massimi da compiere (MAXGEN), uguali le altre percentuali di mappatura.

Nelle tabelle 5.1 e 5.2 sono rappresentati i risultati ottenuti divisi in base all'ordine del modello statistico corrispondente.

MODELLO	Modello correttamente identificato	Modello non identificato (modello proposto con S^2 minore)	Modello non identificato (modello proposto con AIC minore)	Modello non identificato (errore del programma)	TOTALE
AR(1)	100% (3)	-	-	-	100% (3)
AR(2)	100% (3)	-	-	-	100% (3)
AR(3)	50% (3)	50% (3)	-	-	100% (6)
AR(4)	100% (7)	-	-	-	100% (7)
AR(5)	90% (9)	10% (1)	-	-	100% (10)
AR(6)	86% (6)	-	14% (1)	-	100% (7)
AR(7)	100% (7)	-	-	-	100% (7)
AR(8)	64% (14)	18% (4)	18% (4)	-	100% (22)
AR(9)	92% (23)	4% (1)	4% (1)	-	100% (25)
AR(10)	88% (30)	3% (1)	6% (2)	3% (1)	100% (34)
TOTALE	85% (105)	8% (10)	6% (8)	1% (1)	100% (124)

Tabella 5.1. Risultati ottenuti sulle simulazioni di modelli AR.

Nella tabella 5.1 vengono evidenziati i risultati ottenuti con le simulazioni di modelli autoregressivi: nella prima colonna è indicato l'ordine del modello; nella seconda è riportata la percentuale di simulazioni generate con modelli di ordine indicato nella prima colonna che l'algoritmo ha correttamente previsto; nella terza colonna è invece indicata la percentuale di modelli che non è stata correttamente individuata, ma il programma ha selezionato un modello avente AIC minore a causa del minore S^2 (si tratta generalmente di modelli sovrapparametrizzati); nella quarta colonna è indicata la percentuale di quei modelli non correttamente selezionati, ma che presentano un valore di AIC più basso dovuto al minor numero di parametri presenti nel modello da stimare; nella quinta colonna sono riportate le percentuali degli errori commessi dal programma in quanto l'algoritmo non riesce a convergere verso la soluzione; nell'ultima colonna è indicato il totale.

Nelle colonne 2-6, in parentesi, sono riportati i valori assoluti delle simulazioni associate alle percentuali indicate.

Nella tabella 5.2 vengono presentati i risultati relativi alle simulazioni di modelli ARMA con le medesime indicazioni viste per la tabella 5.1.

Nelle tabelle 5.3 e 5.4, invece, vengono mostrati gli stessi risultati delle tabelle 5.1 e 5.2 suddivisi però in base alla dimensione del modello generatore della serie storica corrispondente.

Nella tabella 5.3 rientrano i modelli AR mentre nella tabella 5.4 quelli ARMA.

MODELLO	Modello correttamente identificato	Modello non identificato (modello proposto con S^2 minore)	Modello non identificato (modello proposto con AIC minore)	Modello non identificato (errore del programma)	TOTALE
ARMA(1,1)	100% (3)	-	-	-	100% (3)
ARMA(2,2)	25% (1)	50% (2)	25% (1)	-	100% (4)
ARMA(3,3)	20% (2)	40% (4)	40% (4)	-	100% (10)
ARMA(4,4)	-	100% (4)	-	-	100% (4)
ARMA(5,5)	-	100% (4)	-	-	100% (4)
ARMA(6,6)	-	100% (5)	-	-	100% (5)
ARMA(7,7)	-	50% (3)	50% (3)	-	100% (6)
ARMA(8,8)	-	100% (4)	-	-	100% (4)
ARMA(9,9)	-	100% (4)	-	-	100% (4)
ARMA(10,10)	-	100% (4)	-	-	100% (4)
ARMA(1,2)	100% (4)	-	-	-	100% (4)
ARMA(1,3)	80% (4)	20% (1)	-	-	100% (5)
ARMA(1,4)	50% (3)	33% (2)	17% (1)	-	100% (6)
ARMA(1,5)	50% (2)	50% (2)	-	-	100% (4)
ARMA(2,1)	100% (6)	-	-	-	100% (6)
ARMA(2,3)	60% (3)	20% (1)	20% (1)	-	100% (5)
ARMA(2,4)	-	100% (8)	-	-	100% (8)
ARMA(2,5)	25% (2)	75% (6)	-	-	100% (8)
TOTALE	32% (30)	57% (54)	11% (10)	-	100% (94)

Tabella 5.2. Risultati ottenuti sulle simulazioni di modelli ARMA.

MODELLO	Modello correttamente identificato	Modello non identificato (modello proposto con S^2 minore)	Modello non identificato (modello proposto con AIC minore)	Modello non identificato (errore del programma)	TOTALE
AR ¹	95% (19)	-	5% (1)	-	100% (20)
AR ²	75% (15)	10% (2)	15% (3)	-	100% (20)
AR ³	90% (18)	5% (1)	5% (1)	-	100% (20)
AR ⁴	90% (18)	5% (1)	5% (1)	-	100% (20)
AR ⁵	80% (16)	15% (3)	5% (1)	-	100% (20)
AR ⁶	79% (19)	13% (3)	4% (1)	4% (1)	100% (24)
TOTALE	85% (105)	8% (10)	6% (8)	1% (1)	100% (124)

Tabella 5.3. Risultati ottenuti sulle simulazioni di modelli AR.

MODELLO	Modello correttamente identificato	Modello non identificato (modello proposto con S^2 minore)	Modello non identificato (modello proposto con AIC minore)	Modello non identificato (errore del programma)	TOTALE
ARMA(1,1)	60% (9)	40% (6)	-	-	100% (15)
ARMA(1,2)	47% (7)	47% (7)	6% (1)	-	100% (15)
ARMA(2,1)	40% (6)	47% (7)	13% (2)	-	100% (15)
ARMA(2,2)	33% (5)	47% (7)	20% (3)	-	100% (15)
ARMA(3,1)	20% (3)	67% (10)	13% (2)	-	100% (15)
ARMA(3,2)	-	89% (17)	11% (2)	-	100% (19)
TOTALE	32% (30)	57% (54)	11% (10)	-	100% (94)

Tabella 5.4. Risultati ottenuti sulle simulazioni di modelli ARMA.

Com'è possibile osservare dalle tabelle presentate, i modelli selezionati non coincidono sempre con il modello utilizzato per la simulazione in quanto l'algoritmo converge verso una soluzione ottimale differente.

La causa di questo inconveniente è duplice:

- 1) I modelli sovrapparametrizzati a volte abbassano l' S_p^2 più di quanto si aumenti il termine $2p$, mentre modelli sottoparametrizzati a volte abbassano il termine $2p$ più di quanto possa crescere l' S_p^2 .
- 2) In secondo luogo, tale indicatore viene calcolato sulle previsioni effettuate e non sul campione di dati su cui vengono stimati i parametri. Queste previsioni non possono includere un termine di errore così come capita sui dati di confronto delle stesse: la conseguenza è che, se la componente di errore è molto "forte", l'algoritmo può stimare un modello differente che però in termini di AIC previsivo risulta migliore.

Quindi tutti i modelli stimati non correttamente, riportati in colonna tre e quattro non devono considerarsi, pertanto, errori del programma o difetti dell'algoritmo che non riesce a convergere, ma problemi connessi all'uso di indicatori non sempre efficienti o all'errore stocastico aggiunto alla serie.

Per meglio comprendere quanto possa essere importante questa affermazione, vengono di seguito presentati alcuni esempi di modelli simulati, stimati e previsti dal programma.

La Fig. 5.6 mostra con il colore rosso il test-set di una serie storica generata dal seguente modello:

$$(5.14) \quad X(t) = 0.3X(t-3) - 0.35X(t-6) + 0.4X(t-8) - 0.25X(t-9) - 0.3X(t-10) + \varepsilon_t$$

con il colore blu, invece, vengono evidenziate le previsioni ottenute con il "migliore" dei modelli stimati:

$$(5.15) \quad X(t) = 0.261X(t-3) - 0.376X(t-6) + 0.304X(t-8) - 0.281X(t-9) - 0.209X(t-10)$$

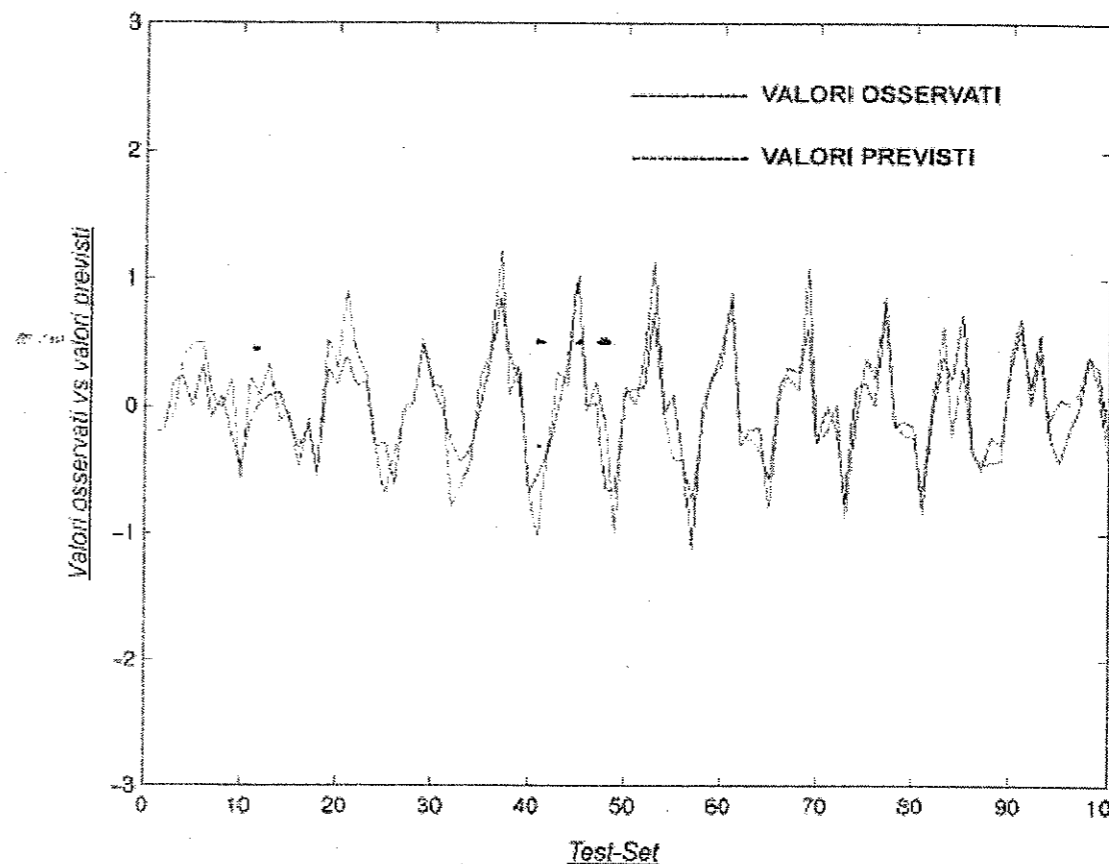


Fig. 5.6. Previsioni vs valori reali (osservati) di una serie storica generata da un modello autoregressivo del decimo ordine.

Come si può notare l'algoritmo converge correttamente verso il modello generatore della serie storica in quanto questo minimizza l' AIC previsivo. La differenza riscontrata nei coefficienti parametrici è causata dal termine di errore del modello generatore.

In questo caso l'algoritmo identifica correttamente il modello: $AR^5(10)$.

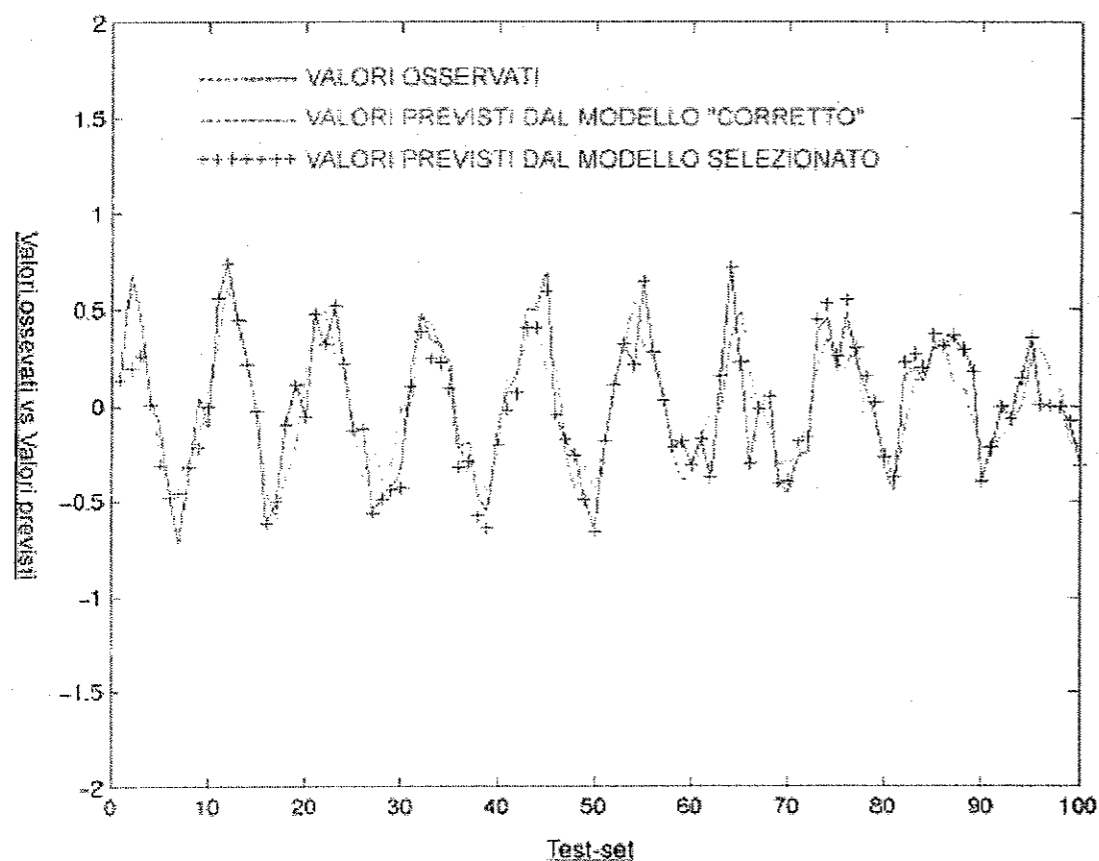


Fig. 5.7. Previsioni (stimate = blu, modello corretto = verde) vs valori osservati (rosso) di una serie storica generata da un modello autoregressivo del nono ordine.

In Fig.5.7 è invece rappresentata la serie storica (in rosso) generata con il modello $AR^5(9)$:

$$(5.16) \quad X(t) = -0.4X(t-1) + 0.3X(t-2) + 0.4X(t-5) + 0.3X(t-7) + 0.3X(t-9) + \varepsilon_t$$

In blu (" + ") sono evidenziate le previsioni stimate con il modello selezionato, $AR^4(9)$:

$$(5.17) \quad X(t) = -0.359X(t-1) + 0.343X(t-5) + 0.137X(t-7) + 0.348X(t-9)$$

In questo caso l'algoritmo non converge correttamente verso il modello generatore della serie storica poiché non è quello che minimizza l'AIC previsivo: l'algoritmo seleziona il modello "migliore" dal punto di vista previsivo.

Nella Fig. 5.7 è possibile vedere anche le previsioni stimate se il modello fosse stato correttamente individuato (in colore verde). Le previsioni risulterebbero notevolmente peggiorate se fosse stato selezionato questo modello.

La causa di questo, come detto, può essere imputata al termine di errore (o eventualmente all'indicatore utilizzato che, in questo caso, sottoparametrizzerebbe il modello).

Se per i modelli AR il problema non è molto rilevante, così come si evidenzia nella tabella 5.1 (in cui l'85% dei modelli simulati è correttamente identificato), per i modelli ARMA non si può dire lo stesso. Il problema non è affatto trascurabile: spesso (in quasi il 70% dei casi) l'algoritmo tende a convergere verso il minimo AIC che non corrisponde a quello del modello utilizzato per la simulazione (vedi tabella 5.2).

Le Figg. 5.8 e 5.9 mostrano le previsioni stimate sulle serie storiche generate da un modello ARMA.

In particolare la Fig. 5.8 mostra con il colore rosso il test-set di una serie storica generata con il seguente modello:

$$(5.18) \quad X(t) = 0.3X(t-3) - 0.4X(t-5) + 0.2X(t-7) + \varepsilon_t + 0.1\varepsilon_{t-2} - 0.15\varepsilon_{t-6}$$

mentre con il colore blu sono evidenziate le previsioni ottenute con il migliore dei modelli stimati:

$$(5.20) \quad X(t) = 0.27X(t-3) - 0.4X(t-5) + 0.22X(t-7) + \varepsilon_t + 0.1\varepsilon_{t-2} - 0.12\varepsilon_{t-6}$$

L'algoritmo converge correttamente verso il modello generatore della serie storica in quanto questo minimizza l'*AIC* previsivo.

In Fig. 5.9 è invece rappresentato il test set di una serie storica (in colore rosso) generata dal seguente modello:

$$(5.21) \quad X(t) = -0.1X(t-8) + \varepsilon_t + 0.35\varepsilon_{t-1}$$

In verde sono evidenziate le previsioni ottenute con il modello previsivo selezionato dall'algoritmo:

$$(5.22) \quad X(t) = -0.38X(t-1) + 0.05X(t-8) + \varepsilon_t + 0.06\varepsilon_{t-3}$$

In blu invece vengono mostrate le previsioni ottenibili se l'algoritmo avesse correttamente individuato il modello generatore della serie storica.

Anche in questo caso l'algoritmo seleziona un modello migliore per quanto riguarda l'*AIC* previsivo.

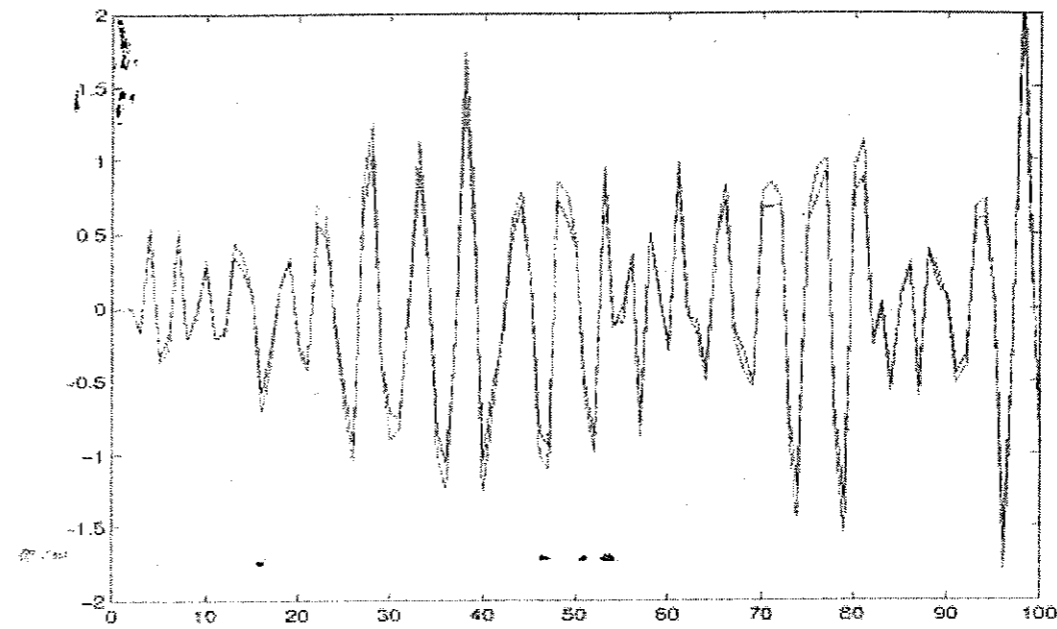


Fig. 5.8. Previsioni (blu) vs valori osservati (rosso) di una serie storica generata da un modello ARMA^{3,2}(7,6).

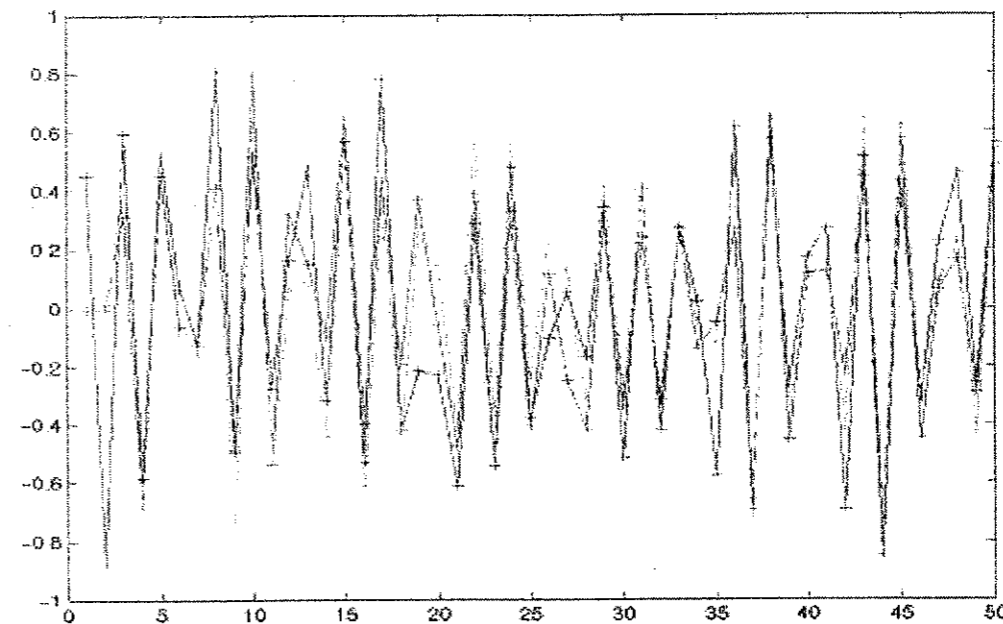


Fig. 5.9. Previsioni (stimate = verde, modello corretto = blu) vs valori osservati (rosso) di una serie storica generata da un modello ARMA.

Ulteriore approfondimento può effettuarsi esaminando le matrici di confusione dei risultati ottenuti utilizzando diversi criteri di selezione: AIC, BIC e SIC.

AR	1	2	3	4	5	6
1	95 %	15 %				
2	5 %	75 %	5 %			
3		5 %	90 %	5 %	5 %	
4		5 %	5 %	90 %		4 %
5				5 %	80 %	5 % err.
6					10 %	79 %
7					5 %	8 %
8						
9						4 %
10						

Tabella 5.9. Indicatore utilizzato : AIC (esperimenti complessivi circa 170).

AR	1	2	3	4	5	6
1	100 %	50 %	25 %			
2		50 %	20 %		10 %	
3			50 %	66 %	20 %	20 %
4			5 % err.	33 %	30 %	30 %
5					40 %	20 %
6						40 %
7						
8						
9						
10						

Tabella 5.10. Indicatore utilizzato : SIC (esperimenti complessivi 60).

AR	1	2	3	4	5	6
1						
2		100 %	20+10err. %			
3			70 %	33 %	20 %	10 %
4	20 %			66 %	50 %	30 %
5	20 %				10 %	30 %
6					20 %	10 %
7	20 %					20 %
8	40 %					
9						
10						

Tabella 5.11. Indicatore utilizzato : BIC (esperimenti complessivi circa 60).

Nelle tre tabelle, in colonna sono indicati il tipo di modello AR utilizzato per la simulazione dei dati (dove per tipo si intende il numero di parametri che caratterizzano il modello indipendentemente dalla posizione occupante all'interno dei 10 campi utilizzabili). In riga, invece, è rappresentato il tipo di modello identificato dal programma GATS.

"Err." indica la percentuale di modelli non stimata correttamente. Cioè le posizioni dei coefficienti nel modello non sono corrette indipendentemente dal numero degli stessi.

Utilizzando come indicatore il SIC si ha una netta tendenza alla sottoparametrizzazione (ad eccezione degli AR(1), ovviamente) che si accentua sempre più aumentando il numero di parametri utilizzati nel modello di simulazione.

I risultati ottenuti con l'indicatore BIC, invece, mostrano un sovrapparametrizzazione quando l'ordine del modello utilizzato è basso (come l'AR(1)) mentre mostrano una sottoparametrizzazione quando il modello utilizzato per la simulazione si complica.

Molto più stabili verso il modello effettivamente utilizzato sono i risultati ottenuti utilizzando l'AIC. In questo caso si ha una tendenza variabile in ugual misura a sovrapparametrizzare o sottoparametrizzare il modello. Soltanto aumentando il

numero dei parametri (AR(5) e AR(6)) si ha un maggiore tendenza alla sovrapparametrizzazione rispetto alla sottoparametrizzazione.

Complessivamente il numero di modelli correttamente individuato risulta notevolmente maggiore utilizzando l'AIC (oltre l'80 %) rispetto a quanto accade utilizzando gli altri due indicatori (circa il 50 %).

La percentuale di modelli stimati in cui il valore dell'indicatore (AIC, BIC, SIC) risulta inferiore a quello del modello utilizzato per la simulazione (il modello non è correttamente identificato ma la procedura porta alla scelta di un modello comunque migliore in termini dell'indicatore utilizzato rispetto a quello utilizzato per la simulazione) è superiore al 99%.

5.2. CONCLUSIONI

L'Algoritmo Genetico anche in questo caso ha risposto ottimamente alle aspettative. I modelli simulati sono stati correttamente specificati nella maggioranza dei casi e, quando questo non è avvenuto, è stato comunque scelto un modello che ha ottimizzato la procedura (AIC minore).

Se l'obiettivo finale era verificare il funzionamento degli Algoritmi Genetici e mettere a punto uno strumento che fosse in grado di prevedere correttamente anche in presenza di una componente di errore stocastica, i risultati sono sicuramente confortanti.

Soltanto l'1% dei modelli simulati è stato scelto in modo errato senza la convergenza dell'algoritmo verso l'ottimo (minimo valore assunto dall'AIC).

I risultati esposti nelle tabelle 5.1 e 5.2 mostrano che lo strumento utilizzato ha, complessivamente, buone capacità previsive.

Un'ultima considerazione per quanto riguarda gli strumenti utilizzati e i programmi a tale proposito implementati: in questo caso non si è reso necessario l'utilizzo di metodologie quali la "stepwise", la "backward elimination", la "forward selection".

L'utilizzo di tali metodologie è auspicabile, infatti, soltanto quando si cerca di ridurre il numero di parametri utilizzati, e non quando l'unico obiettivo prefissato sia quello di affinare un buon strumento previsivo. In questi casi infatti non importa la semplicità o la complessità del modello ma soltanto la sua capacità previsiva.

Con le stesse motivazioni si è rifiutata l'idea di implementare un algoritmo ibrido che affiancasse la Logica Fuzzy agli Algoritmi Genetici.

6. MODELLI NON LINEARI APPROSSIMATI: RETI NEURALI

Le Reti Neurali sono strumenti per l'elaborazione distribuita, originariamente progettati come metafore computazionali del cervello umano. Fanno parte di una classe più ampia di algoritmi capaci di apprendimento con applicazioni principali nei compiti di riconoscimento, di associazione e di manipolazione di strutture (pattern). Sono costituite da una moltitudine di unità elementari di elaborazione (processing elements, PE) interconnesse a formare una rete. Ciascuna unità compie una sequenza limitata di operazioni numeriche semplici e ne comunica i risultati alle unità adiacenti tramite una connessione cui è applicato un peso (detto peso sinattico). Nella maggior parte dei casi il neurone artificiale (nodo, o più semplicemente unità), è costituito da connessioni input-output, da un dispositivo per la sommatoria dell'attività di input e da una funzione di attivazione che viene applicata alla sommatoria dei valori di input al fine di determinare il livello di risposta proprio di ciascun nodo e, quindi, il suo livello di output (si veda la Fig. 6.1).

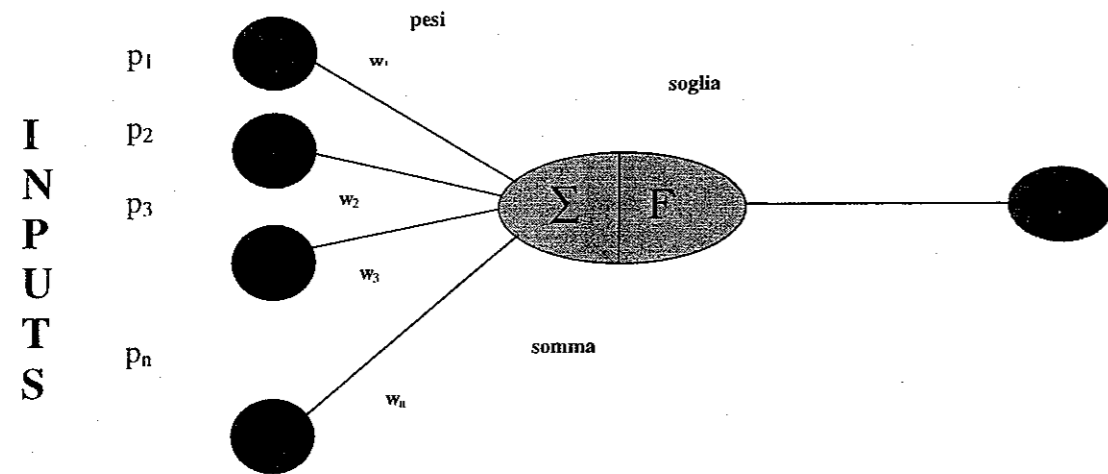


Fig. 6.1. Visualizzazione di un singolo neurone artificiale.

Variando l'intensità dei pesi sinattici delle connessioni si può ottenere una variazione della risposta funzionale della rete agli stimoli esterni, detti anche pattern di input. Le Reti Neurali non vengono programmate in modo diretto: si fa uso, invece, di un *algoritmo di apprendimento*. L'algoritmo di apprendimento è un meccanismo sistematico per la regolazione dell'intensità dei pesi della rete, con lo scopo di ottenere una determinata risposta funzionale, fissata come obiettivo per lo strato di unità di uscita, in corrispondenza di un dato insieme di input. Una rete neurale artificiale può allora essere vista semplicemente come una particolare implementazione di una mappa da R^n a R^m , dove n è il numero di input e m è il numero di output. Poiché la mappa che è implementata dalla rete neurale artificiale dipende dai pesi delle connessioni fra i singoli neuroni, essa può essere modificata variando opportunamente i valori di tali pesi. L'informazione sulla mappatura è immagazzinata nei valori dei pesi a livello di tutti i neuroni e quindi la rete neurale costituisce una rappresentazione distribuita della mappatura. Le reti neurali artificiali possono essere allora considerate una tecnica computazionale parallela, poiché l'output corrispondente a ogni dato input è ottenuto grazie all'unione dei rapporti input-output, che avvengono a livello di singolo neurone.

L'addestramento delle Reti Neurali ricorre principalmente a due paradigmi distinti: l'apprendimento supervisionato e quello non supervisionato. Nel primo caso, l'algoritmo di apprendimento esplora l'insieme dei possibili valori che i pesi possono assumere al fine di eliminare quelle configurazioni incompatibili con gli esempi relativi alla mappa di uscita desiderata. In questo senso si può ritenere che la "conoscenza" della rete addestrata risieda nell'insieme dei pesi. Per l'apprendimento supervisionato è pertanto necessario individuare un insieme di esempi di coppie input-output per l'addestramento, che la rete deve imparare a rappresentare.

Questa procedura è spesso usata in contesti previsionali o in applicazioni ai controlli di sistema [Bar83],[Whi92]. Lo schema del processo di apprendimento supervisionato è rappresentato in Fig. 6.2.

In sostanza, ciò che si richiede alla rete neurale addestrata è di comportarsi come un imitatore di funzioni, trasformando ciascun vettore di ingresso in corrispondenti vettori di uscita. Allenare la rete significherà allora aggiustare i pesi in modo tale che la mappa che è implementata da essa sia il più vicino possibile alla mappa desiderata. È proprio in questo senso che le reti neurali artificiali, utilizzando un processo di allenamento supervisionato, possono essere viste come **approssimatori di funzioni**.

Fissate l'architettura della *rete*, le funzioni di attivazione e una distribuzione dei pesi inizialmente casuale, la tecnica della retropropagazione consente di ottenere dalla rete il comportamento richiesto. Tale risultato viene conseguito attraverso una discesa differenziale (gradient descent) attraverso lo spazio vettoriale dei pesi in base a una funzione prefissata di errore o di costo. Quest'ultima funzione viene di solito definita come una singola misura di distanza vettoriale tra l'output della rete e il vettore di output desiderato. La rete quindi reitera il procedimento di apprendimento fino a riprodurre (entro limiti di accuratezza predeterminati) la risposta richiesta per tutti gli elementi dell'insieme di addestramento.

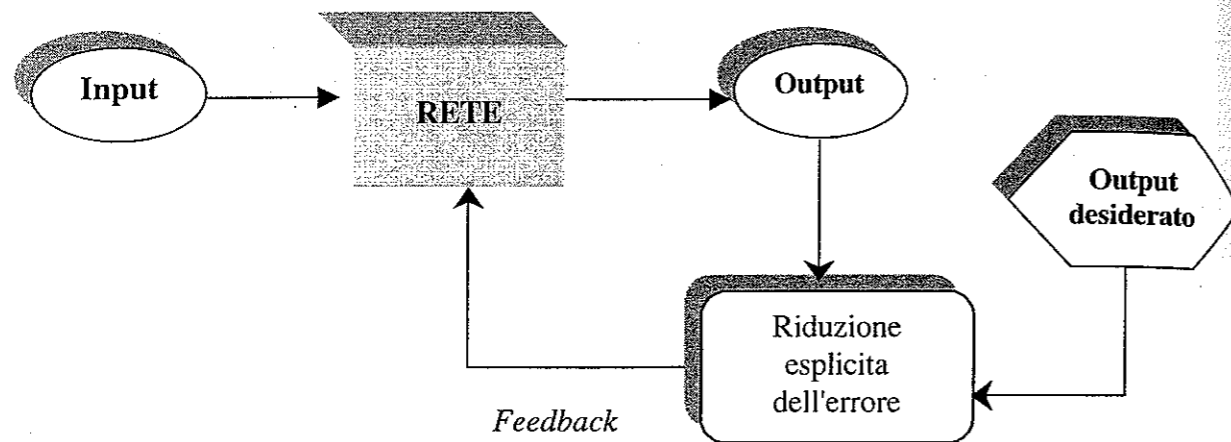


Fig. 6.2. Schema del processo di addestramento supervisionato.

La seconda forma di addestramento di Reti Neurali è l'apprendimento non supervisionato. Con questo tipo di tecniche la rete viene addestrata in base a un insieme di input senza fornire, così come avviene con l'apprendimento supervisionato, il corrispondente insieme di output. L'algoritmo si basa sull'assunto che all'interno dei dati esistano configurazioni significative (per esempio alcuni raggruppamenti) per le quali non esistono risposte esplicitamente prefissate: da questo si ricava la definizione di apprendimento non supervisionato. La versione più diffusa di questo genere di rete neurale è probabilmente la rete di Kohonen [Koho82]. Il suo obiettivo è quello di scoprire regolarità o configurazioni ricorrenti (pattern) all'interno dei dati, in modo da poter classificare in modo automatico un insieme di oggetti. In questo senso la rete neurale fornisce uno strumento simile a quello del raggruppamento statistico.

A causa di questa sua caratteristica la griglia di output addestrata viene spesso definita come una mappa topologica (topological map). Determinate regioni della griglia possono venire etichettate fornendo alla rete vettori tipici di input e annotando le unità di output che forniscono i livelli più significativi di risposta. La rete impara a raggruppare i vettori di input variando in modo adattivo i pesi delle connessioni,

seguendo un semplice insieme di regole. Il processo di apprendimento porta a un insieme di associazioni tra sottoinsiemi di vettori di input e specifici nodi di output.

Lo schema del processo di addestramento non supervisionato è rappresentato in Fig. 6.3.

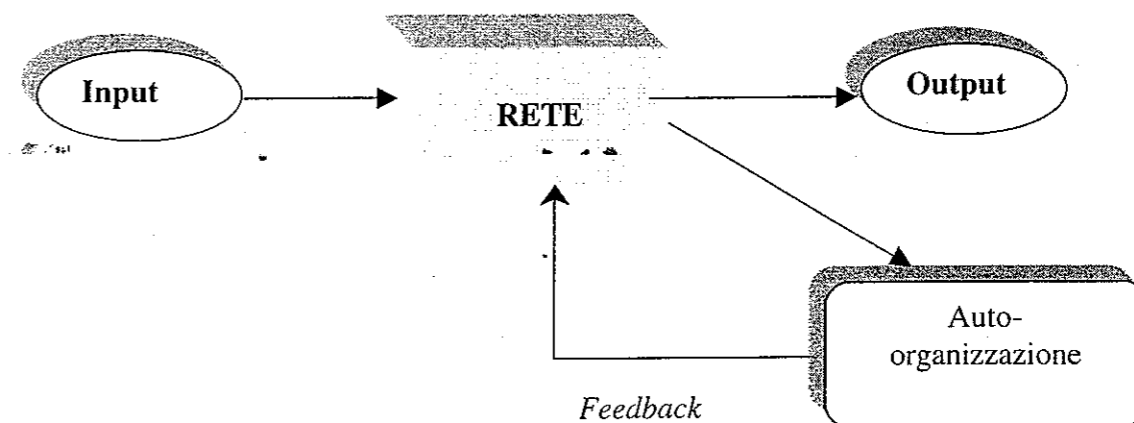


Fig. 6.3. Schema del processo di addestramento non supervisionato.

In termini più attuali, le Reti Neurali possono essere classificate come strumenti di modellizzazione non parametrica e non lineare con una riconosciuta capacità a comportarsi come approssimatori funzionali. Uno dei vantaggi principali di un algoritmo di apprendimento è quello di offrire la possibilità di generalizzare concetti. In termini immediati, una generalizzazione può essere vista come la capacità di un modello di operare su dati mai incontrati in precedenza, o su dati esclusi dall'insieme di apprendimento. In generale, ci si aspetta che la rete impari a imitare una certa funzione su di una collezione di esempi molto più piccola dell'intero insieme di dati (potenzialmente infinito).

L'assunto che sta alla base di questa aspettativa è che qualsiasi cosa che può essere appresa possieda una qualche forma di regolarità o di struttura sottostante: sarà questa a consentire di effettuare previsioni. Si tratta di un tema largamente dibattuto, e certamente non confinato al solo ambito degli studi sulle Reti Neurali. Sotto questo aspetto molte delle attuali ricerche sull'argomento si sovrappongono largamente alle forme tradizionali di tecnica di costruzione di modelli statistici.

Le Reti Neurali si rivelano particolarmente utili quando il sistema che si propongono di modellizzare è di tipo non lineare e su di esso è disponibile una ridotta conoscenza a priori, con particolare riguardo alla sua esplicitazione funzionale. Questo offre un sensibile vantaggio sulle tradizionali tecniche statistiche, le quali tendono sia alla linearità che a richiedere un esperto del dominio applicativo per la costruzione del modello. Un ulteriore vantaggio è dato dalla loro capacità di maneggiare in modo efficace pattern e correlazioni all'interno di dati "rumorosi", incompleti, o addirittura contraddittori.

Un limite fondamentale presentato dalle Reti Neurali è la loro incapacità intrinseca di fornire spiegazioni. In generale, il modello emerso in seguito al processo di apprendimento è di difficile interpretazione in riferimento alle correlazioni trovate. Emerge, sotto questo aspetto, un netto contrasto con altre tecnologie come, per esempio, quella dei sistemi esperti o a induzione di regole: caratteristica che esclude l'utilizzo delle Reti Neurali in quelle aree dove le spiegazioni delle azioni intraprese sono un elemento fondamentale. Un altro problema è quello della determinazione di parametri strutturali che compongono una singola Rete Neurale artificiale, come la scelta della tipologia più opportuna, dei tassi di apprendimento, delle funzioni di attivazione e di convalida del modello.

6.1. IL NEURONE ARTIFICIALE E I SUOI COMPONENTI

La Fig. 6.1 mostra gli elementi di un neurone artificiale. Gli inputs p_1, p_2, p_3 corrispondono ai segnali che giungono dall'esterno. Essi vengono opportunamente pesati, e i pesi (w_1, w_2, w_3) , indicano quindi il grado di importanza del segnale ricevuto. Il segnale giunge poi al corpo del neurone costituito dalla funzione di somma grazie alla quale gli input (opportunamente pesati) sono aggregati a formare un unico valore, a cui di norma si aggiunge un parametro di distorsione. Questo costituirà l'argomento della funzione di trasferimento che emetterà un segnale diverso a seconda del valore passato. La funzione di trasferimento ha il compito di controllare se il valore calcolato supera o meno una determinata soglia e di fornire l'output che costituisce il segnale per altri neuroni.

Quindi un singolo neurone potrà fornire il seguente output:

$$(6.1) \quad a = F\left(\sum_{i=1}^n w_i p_i + b\right)$$

dove a corrisponde all'output fornito dal neurone, w e b rispettivamente ai pesi e alla distorsione che si modificano nel tempo mentre la rete viene allenata secondo una legge di *learning* preventivamente stabilita, e F alla funzione di trasferimento prefissata.

Le componenti principali di un neurone artificiale sono:

- *Pesi*: è grazie a loro si riesce a emulare la diversa intensità della forza di impulso presente a livello delle sinapsi. Ogni neurone può ricevere più input simultaneamente, e ognuno di essi può avere una diversa importanza (molto importante se l'intensità associata è alta, poco importante se l'intensità associata è bassa). Misurano la forza di connessione fra gli input e il neurone. Sono parametri scalari che possono essere variati in modo tale che la rete mostri un comportamento desiderato o interessante per la risoluzione del problema. Possono essere aggiustati esogenamente o dalla rete stessa mentre

questa viene allenata.

- **Funzione di somma:** dopo l'attribuzione di un peso a ogni input si calcola la somma di tutti gli input opportunamente pesati. Geometricamente gli input e i pesi possono essere rappresentati da due vettori, (p_1, p_2, \dots, p_n) e (w_1, w_2, \dots, w_n) . Il segnale totale si ottiene moltiplicando ogni componente del vettore di input per il corrispondente elemento del vettore dei pesi $(w_1 * p_1, w_2 * p_2, \dots, w_n * p_n)$ e sommando i diversi risultati ottenuti dalla suddetta moltiplicazione $(w_1 * p_1 + w_2 * p_2 + \dots + w_n * p_n = \sum_{i=1}^n w_i p_i)$. Di norma viene anche aggiunto un parametro di distorsione, b , che va a moltiplicare un parametro fornito dall'esterno in ingresso alla rete, solitamente posto uguale a 1, per cui il segnale complessivo diventa: $\sum_{i=1}^n w_i p_i + b$. Questa somma costituirà l'argomento che sarà passato alla funzione di trasferimento. Questa non è l'unica opportunità possibile di passaggio degli input: i pesi e gli input possono infatti essere combinati in molti modi differenti prima di diventare argomento della funzione di trasferimento. Questa scelta sarà lasciata al ricercatore, che dovrà tenere conto del contesto di applicazione della rete neurale.
- **Funzione di Trasferimento:** l'importanza della funzione di trasferimento è fondamentale per la risposta fornita dalla rete; questa infatti è strettamente dipendente dalla funzione scelta. Ne esistono di diversi tipi (alcuni dei quali sono rappresentati in Fig. 6.4): lineare, "hard limit", sigmoide logaritmica, tangente sigmoide, etc... Argomento della funzione sarà chiaramente il valore somma degli inputs, di cui si è detto sopra. Questo valore sarà poi confrontato con un valore soglia: se è superiore si avrà la generazione di un segnale, se invece è inferiore sarà prodotto un segnale inibitorio o nessun segnale. Entrambi i tipi di risposta sono significativi.

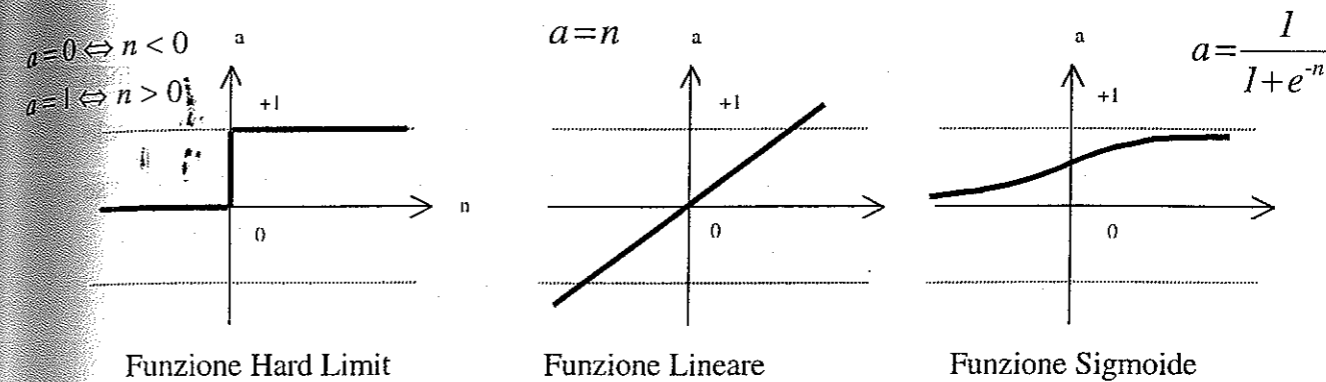


Fig. 6.4. Esempi di funzioni di trasferimento (in ascissa viene indicato il valore somma degli input, in ordinata il segnale restituito).

- **Scala e Limiti:** il risultato ottenuto dalla funzione di trasferimento può essere poi opportunamente modificato, per ottenere un valore più adatto al tipo di ricerca che si deve affrontare: si potrà quindi moltiplicare per un fattore di scala e/o aggiungere un determinato valore, mantenendosi però all'interno di limiti prefissati.
- **Funzione di output (Competizione):** solitamente l'output ottenuto da ogni neurone è proporzionale al valore determinato dalla funzione di trasferimento; ciò non preclude la facoltà di modifiche ulteriori: potrebbe, per esempio, essere modificato per evidenziare le sue capacità rispetto a quelle dei neuroni vicini. Dopo un opportuno confronto, è possibile determinare quali sono i neuroni maggiormente attivi che, perciò, "meritano" di partecipare in misura maggiore al processo di *learning* o di adattamento.
- **Funzione di Errore:** di norma le reti sono adatte a problemi in cui è tollerata una certa componente di errore. Questa può essere calcolata in modi diversi. Nell'apprendimento supervisionato il termine di errore è normalmente calcolato come differenza fra il valore ottenuto e quello desiderato (si veda la Fig. 6.2). Esso è di fondamentale importanza per l'addestramento della rete, in quanto le variazioni dei pesi sono calcolate in modo tale da minimizzare questa componente di errore.
- **Funzione di learning :** è il motore base per raggiungere la

risoluzione del problema. Essa rappresenta matematicamente il metodo seguito per modificare i pesi di connessione attribuiti agli input, in modo tale che la rete "impari" a risolvere il problema, "adattandosi" alle diverse informazioni ricevute.

6.2. PRINCIPALI TIPI DI RETE

Di norma in ogni rete neurale i singoli neuroni sono interconnessi a formare livelli diversi. In generale, come mostrato in Fig. 6.5, l'architettura di una rete risulta formata dalla combinazione di tre parti:

- uno strato di input, composto dai neuroni che ricevono gli input dall'esterno;
- uno strato di output, che fornisce al mondo esterno la risposta della rete al problema;
- uno o più strati nascosti, esternamente non visibili, essendo posti fra lo strato di input e quello di output.

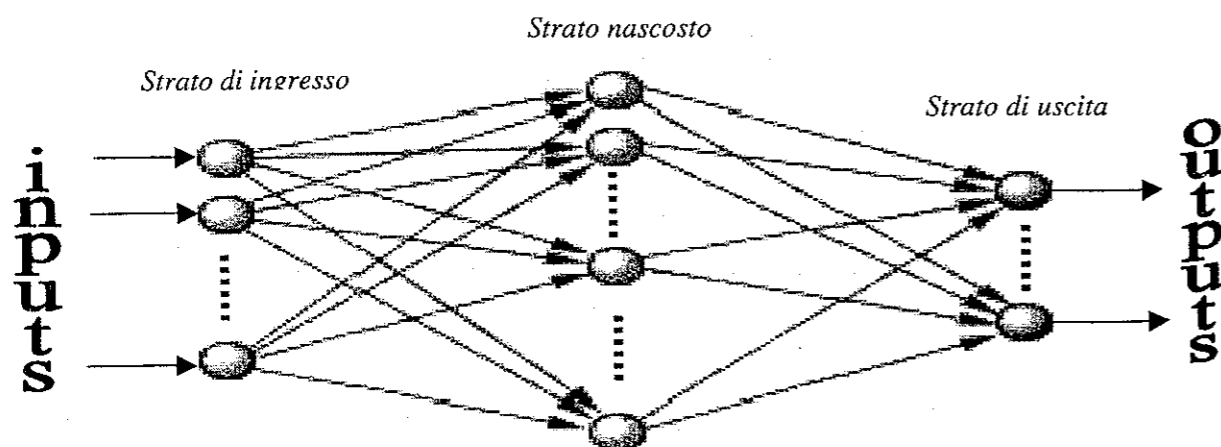


Fig. 6.5. Architettura di una rete neurale artificiale.

Questi strati possono essere fra loro collegati in modi differenti che identificano l'architettura della rete. A seconda del tipo di collegamento adottato esistono quattro tipi principali di reti neurali:

- *Layered feed-forward neural network*: il flusso di informazioni si muove in maniera unidirezionale, poiché ogni strato di neuroni riceve input solo dallo strato precedente e li trasmette solo allo strato successivo. Normalmente i dati per questo tipo di rete corrispondono a un solo insieme di input e producono un solo insieme di output. La rete risulta statica e senza memoria poiché la risposta della rete non dipende dalla risposta che essa aveva dato precedentemente. Può essere introdotta una modificazione a questo schema permettendo che l'errore sia retropropagato per correggere le connessioni dei pesi. Un esempio è dato dal modello di rete proposto in Fig. 6.5.
- *Recurrent neural network*: in questo caso il flusso di informazione non è unidirezionale, poiché alcuni input della rete corrispondono agli output da essa calcolati precedentemente, avendo la rete delle connessioni backward. Questo tipo di architettura permette alla rete di avere una memoria associativa. Infatti, ogni volta che un nuovo input è passato alla rete, la risposta non dipende solo da esso, ma anche dall'analisi svolta precedentemente (vedi Fig. 6.6).

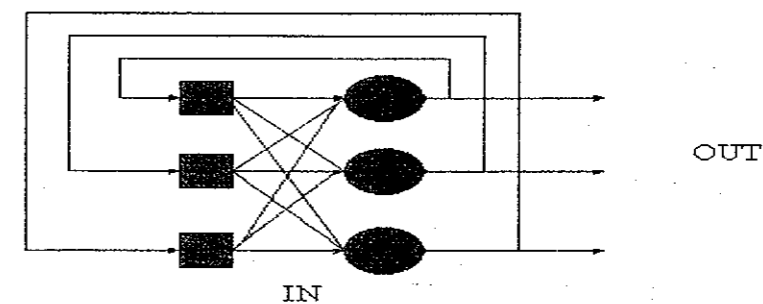


Fig. 6.6. Recurrent neural network.

- *Laterally connected neural networks*: essa è costituita da unità di input *feed-forward* e da uno strato di neuroni collegati lateralmente ognuno ai rispettivi vicini. Ogni neurone riceve input dall'esterno, ma anche da tutti i neuroni che lo circondano. Un esempio è dato dalla rete di Kohonen.
- *Hybrid network*: si denominano in tal modo quelle reti che raggruppano peculiarità tipiche delle reti sopra esaminate, nella stessa architettura. Un esempio può essere dato dalla macchina di *Boltzmann* [Hin84].

Tra i principali tipi di rete di seguito utilizzati nell'analisi sperimentale ricordiamo:

Reti di Elman [Elm90]: sono uno strumento molto utile per riconoscere e generare modelli spaziali e temporali. Sono costituite da due strati di neuroni. Il primo strato presenta una connessione del tipo *feedback* (generalmente con una funzione di trasferimento di tipo tangente sigmoide), mentre il secondo strato è unidirezionale e generalmente caratterizzato da una funzione di trasferimento lineare. Con questo tipo di rete può approssimare ogni funzione con un numero di discontinuità finito, in modo abbastanza preciso. La rete riceverà allora come inputs non solo quelli passati esternamente, ma, come si può vedere dalla Fig. 6.7, anche l'output calcolato a livello del primo strato e riferito all'epoca precedente. In tal modo si immagazzinano informazioni dell'epoca precedente e si usano per quella presente.

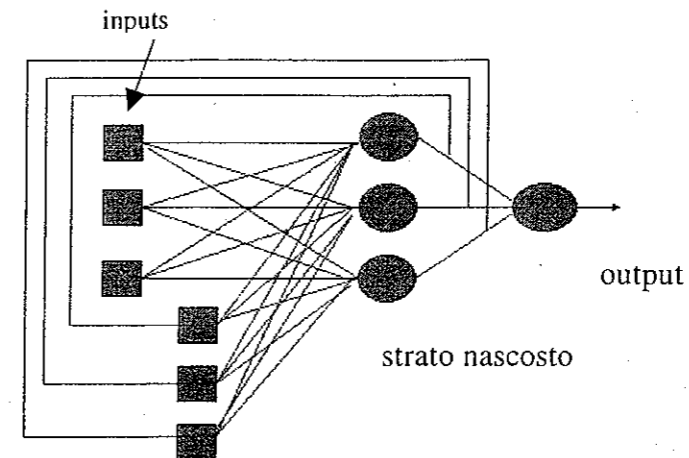


Fig. 6.7. Rete di Elman.

Radial Basis Function Network: questa denominazione identifica un tipo di rete *feed-forward* caratterizzata da uno strato di neuroni nascosti con una funzione di trasferimento di tipo radiale e uno strato di neuroni di output con una funzione di tipo lineare. I neuroni dello strato nascosto calcolano la misura data dal prodotto fra un parametro di distorsione e la distanza vettoriale fra il vettore dei pesi w e il vettore degli input x , come mostrato in Fig. 6.8, e poi eseguono una trasformazione non lineare usando una funzione di tipo radiale.

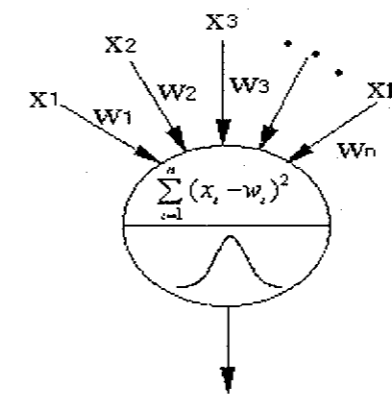


Fig. 6.8. Neurone artificiale usato nelle reti RBF.

Le funzioni radiali sono una classe speciale di funzioni caratterizzate da una risposta (output) che diminuisce all'aumentare della distanza tra gli input e i pesi. Fra le funzioni radiali più comunemente usate troviamo la gaussiana:

$$(6.2) \quad F(n, b) = e^{-(b \cdot n)^2}$$

dove n corrisponde alla distanza fra w e x , $n = \|x - w\|$, mentre b corrisponde a un parametro di distorsione.

Di norma si moltiplica la distanza per un parametro di distorsione che permette di correggere la sensibilità del neurone al variare della distanza. Ogni neurone dello strato nascosto è "fully-connected" con i neuroni dello strato di output, dove viene semplicemente calcolata la somma pesata dei valori forniti dallo strato nascosto e poi eseguita una semplice trasformazione lineare.

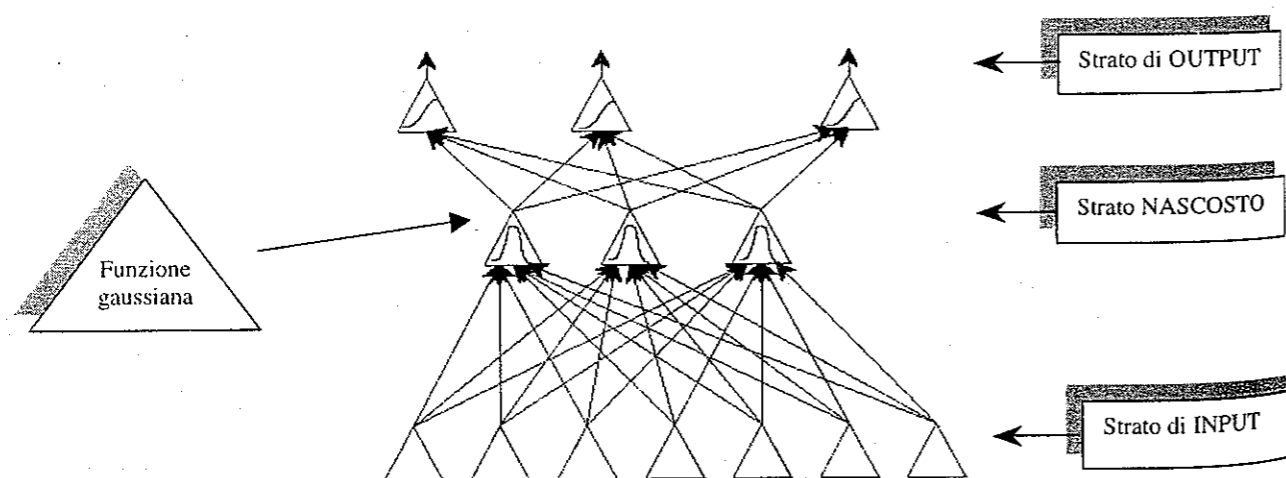


Fig. 6.9. Architettura di una rete RBF.

Questo tipo di rete è risultata molto efficace nei problemi di classificazione e di *mapping*. Inoltre Park & Sandberg [PaSa91] hanno mostrato che reti RBF con un solo strato nascosto hanno la capacità di approssimare ogni tipo di funzione.

6.3. IL PROGRAMMA GANND

La somiglianza dell'apprendimento tramite esempi, o apprendimento supervisionato, con l'analisi delle serie temporali ha comportato un crescente utilizzo delle Reti Neurali come strumenti per la previsione. In particolare si è fatto ricorso alle reti per la previsione degli andamenti dei corsi azionari, nell'analisi dei mercati, per la previsione dei tassi di cambio, oltre che per la valutazione dei rischi connessi alla concessione di prestiti o di assicurazioni e l'identificazione delle frodi.

Una delle ragioni che si possono dare per questo fenomeno è che le reti offrono un semplice strumento per la costruzione di modelli non parametrici di regressione. Questo costituisce chiaramente un vantaggio in quelle circostanze in cui si hanno scarse conoscenze *ex-ante* riguardo alle possibili relazioni tra le variabili determinanti: per esempio, nei casi in cui la serie obiettivo viene ipotizzata come casuale, come nella previsione dei prezzi sul mercato finanziario.

Già da tempo all'interno della Facoltà di Economia dell'Università di Modena e Reggio Emilia è iniziata una ricerca sull'uso delle Reti Neurali applicate alle previsioni degli andamenti di borsa [MiPo98], [Pat98], [MiPP99], [MiPP00].

Questa ricerca si è concentrata sui contratti futures¹ (in modo particolare il future sull'indice MIB30: il cosiddetto FIB30). La scelta è caduta su questo titolo in quanto consente buoni guadagni a fronte di variazioni limitate di prezzo.

Le tipologie di Reti Neurali scelte corrispondono alla rete di Elman, alla rete a base radiale e alla rete feed-forward a due strati con retropropagazione dell'errore. La toolbox di MATLAB "Artificial Neural Network" ha facilitato l'implementazione di queste tipologie di architetture neurali data la possibilità di utilizzare funzioni già precostituite.

Non esistono criteri universalmente validi per determinare i parametri strutturali di una rete neurale artificiale, sebbene sia necessario riuscire a calibrarli in maniera

¹ I futures sono contratti a termine tra due controparti per la compravendita di una determinata quantità di una "attività", a un prezzo e a una data futura stabiliti; il prezzo di questo contratto dipende dal valore dell'attività sottostante, che può essere di diverso tipo: possono infatti essere negoziati sia su valori mobiliari (financial futures), che su beni materiali (merci, metalli, ...) (commodities futures).

adeguata: è infatti molto importante riuscire a ottimizzare la struttura della rete in funzione della potenziale miglior performance che essa è in grado di fornire.

L'idea di utilizzare un algoritmo genetico presenta numerosi vantaggi. Disegnare una Rete Neurale artificiale richiede una profonda conoscenza del problema e della potenziale soluzione dello stesso, che di norma non si conosce. Quindi è spesso necessario testare differenti strutture di rete, nel tentativo di riuscire a determinare la struttura migliore. L'algoritmo genetico, invece, consente di automatizzare la procedura lasciando che sia il programma a fornire l'architettura migliore possibile. Partendo da una generazione casuale di possibili valori, porta a determinare le variabili che fanno sì che la rete risulti essere efficace per determinare la soluzione del problema, evitando il gravoso compito di un'analisi di tutte le possibili combinazioni dei differenti parametri strutturali.

Esaminiamo ora i passi attraverso cui si è giunti a rappresentare le variabili nell'Algoritmo Genetico.

Inizialmente la rappresentazione era la seguente:

- La prima casella del cromosoma era riservata all'ordine dei ritardi del modello AR sottostante la serie storica dell'indice del Fib30.
- Le caselle 2-9 determinavano quali ritardi considerare e costituivano il criterio base al quale modellare gli input da fornire alla rete.
- Le caselle 10-13 determinavano i parametri strutturali della rete.

A differenza della notazione utilizzata nei modelli mostrati nei precedenti capitoli in cui la rappresentazione avviene con notazione "binaria", quella utilizzata in questo caso era di tipo "intero".

Per meglio comprendere le motivazioni che hanno portato all'abbandono di questo tipo di rappresentazione è necessario presentare alcuni pratici esempi sul suo funzionamento.

Tralasciando le caselle di determinazione della struttura parametrica della rete, una stringa così composta:

3	3	5	6	8	1	3	2	8
---	---	---	---	---	---	---	---	---

renderebbe rilevanti soltanto i primi quattro geni. Il primo indica il numero dei parametri rilevanti (3) e i successivi tre indicano quali ritardi considerare. Il modello AR associato sarebbe un $AR^3(8)$:

$$(6.3) \quad y(t) = \alpha_3 y(t-3) + \alpha_5 y(t-5) + \alpha_6 y(t-6)$$

le successive 5 caselle risulterebbero irrilevanti nel programma.

Due sono i problemi che scaturiscono da questo tipo di rappresentazione:

1. Se nelle caselle rilevanti successive alla prima si presentano due geni con medesimo valore, l'individuo viene scartato per impossibilità a procedere.

Per esempio, il seguente cromosoma:

4	3	2	6	2	8	1	3	8
---	---	---	---	---	---	---	---	---

presenta 5 geni rilevanti. Osservando più attentamente la stringa è possibile notare che il ritardo (t-2) compare due volte.

Dalle analisi svolte e dalle prove effettuate è emerso che un gran numero di individui di ogni popolazione veniva scartato a causa di questo problema. Inoltre, all'aumentare del numero di ritardi massimi, la probabilità che si verifici tale inconveniente aumenta.

La causa di tutto questo era la quasi impossibilità di generare modelli AR di ordine k con k ritardi tutti significativi (modello saturo).

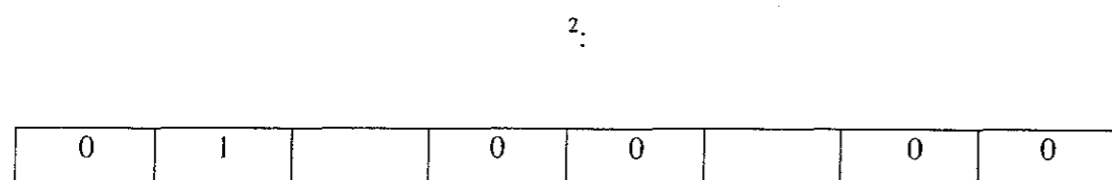
Altro inconveniente non affatto trascurabile era la perdita in termini temporali di risorse utilizzabili dal computer.

2. Non era possibile rappresentare individui il cui modello AR fosse di ordine massimo una potenza di due.

essità di dare una codifica numerica di tipo intero ai geni.

Questa tecnica di rappresentazione degli individui è stata quindi migliorata con lo

Si sono create stringhe di tipo binario formate da una sequenza di 0 e già utilizzato nei capitoli precedenti.



rappresenta il modello AR (6):

$$(6.4) \quad y(t) = \alpha_{t-2} y_{t-3} + \alpha_6 y_{t-6}.$$

Vengono così risolti entrambi i problemi sopra accennati: il primo in quanto più possibile avere due ritardi uguali, il secondo in quanto la lunghezza della stringa può essere fissata a piacimento senza alcuna restrizione numerica.

Dopo aver definito il numero di ritardi da considerare e la loro identificazione, si a modellare gli input da fornire alla rete. Alla rete viene fornita una matrice di dati le cui righe contengono i dati corrispondenti ai ritardi preventivamente colonne risulta determinato dalla lunghezza dell'intervallo di previsione. Nel caso in $t=35$, la matrice degli inputs (nell'ipotesi che

² Si noti che il mo definisce i caratteri strutturali della rete e comunque ha la medesima definizione in entrambe le rappresentazioni) ha ordine massimo 8, a differenza di qua una stringa costituita da 9 celle.

colonna contenente il dato al tempo 33 (=35-2), al tempo 32 (=35-3) e al tempo 29 (=35-6).

Nel caso in cui si vogliono prevedere i valori al tempo 35 e 36, come è schematizzato in Fig. 6.10, la matrice di inputs sarà formata da due colonne, di cui la prima esattamente corrispondente a quella definita nel caso precedente, mentre la seconda determinata dai dati all'epoca 34 (=36-2), all'epoca 33 (=36-3) e infine all'epoca 30 (=36-6). All'aumentare dell'intervallo di previsione, aumenterà il numero di colonne della matrice degli inputs.

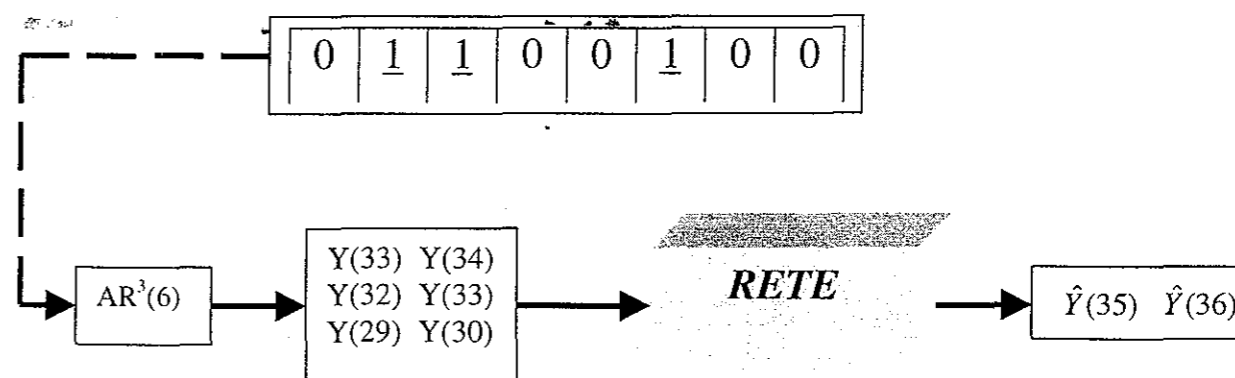


Fig. 6.10. Visualizzazione modellazione input da fornire alla rete.

Vediamo ora il contenuto delle ultime quattro celle (10-13) della stringa:

- La decima cella contiene il numero di neuroni che la rete ha nello strato nascosto.
- L'undicesima e la dodicesima cella contengono rispettivamente le funzioni di trasferimento dello strato di neuroni nascosto e dello strato di neuroni di output (da scegliere in un dominio di tre funzioni: lineare, iperbolico tangente sigmoid e sigmoide logaritmica).
- La tredicesima e ultima cella determina il valore del "learning rate". Rappresenta il tasso di apprendimento che misura la velocità di convergenza dei pesi iniziali delle connessioni verso quelli finali.

L'intera st

0	1	1	0	0	1	0	0	22	3	1	2
A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12

Identificazione dei ritardi	Numero di neuroni	Funzioni di trasferimento	Learning rate
-----------------------------	-------------------	---------------------------	---------------

Fig. 6.11. Visualizzazione della struttura di un singolo individuo dell'algoritmo genetico.

Il programma GANND è rappresentato nei diagrammi di flusso in Figg. 6.12, 6.13 e 6.14.

Si è applicato il programma sui dati tratti dalla serie storica a breve dei prezzi di chiusura del Fib30 dal 28/11/94 al 02/02/98. I dati sono stati ricavati dal programma di analisi tecnica MF analisi 5.00 [MFA97].

La serie storica dei prezzi, visualizzata in Fig. 6.15, è suddivisa in due intervalli corrispondenti al *Training-set* e al *Test-set*, utilizzati rispettivamente per allenare la rete e testare il livello di apprendimento raggiunto dalla stessa.

Si sono normalizzati i dati in un intervallo [0,1] con lo scopo di rendere più rapidi e precisi le procedure di calcolo dei pesi nella rete.

Come si può vedere in Fig. 6.15, il *Training set*, definito dall'insieme di dati a sinistra della linea rossa, è in questo caso composto da 500 dati (corrispondenti ai valori normalizzati dei prezzi di chiusura del Fib30 con scadenza più ravvicinata dal 12/01/1995 al 14/01/1997), mentre il *Test set*, definito dall'intervallo a destra della linea rossa, da 80 dati (corrispondenti ai valori normalizzati dei prezzi di chiusura del Fib30 con scadenza più ravvicinata dal 15/01/1997 al 09/05/1997).

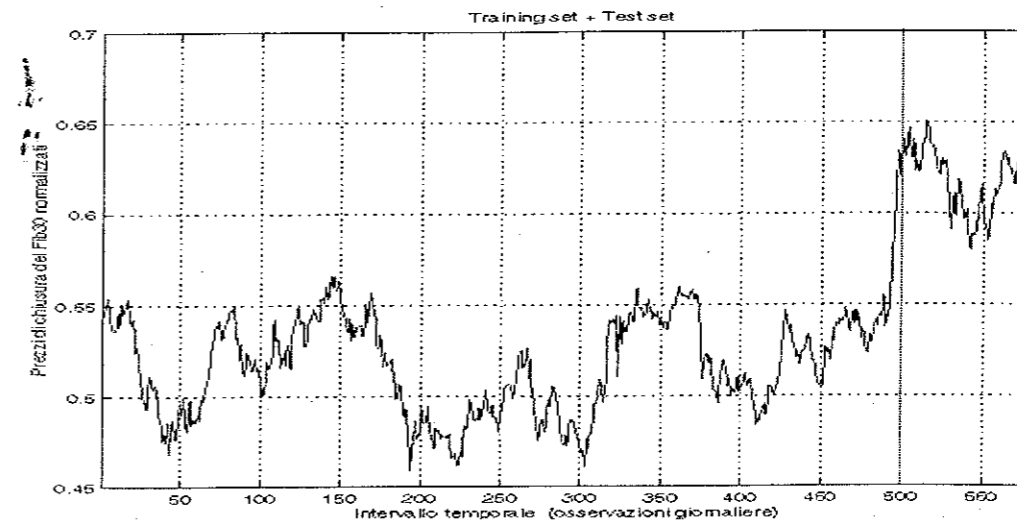


Fig. 6.11. Visualizzazione dati contenuti nel Training set e nel Test set.

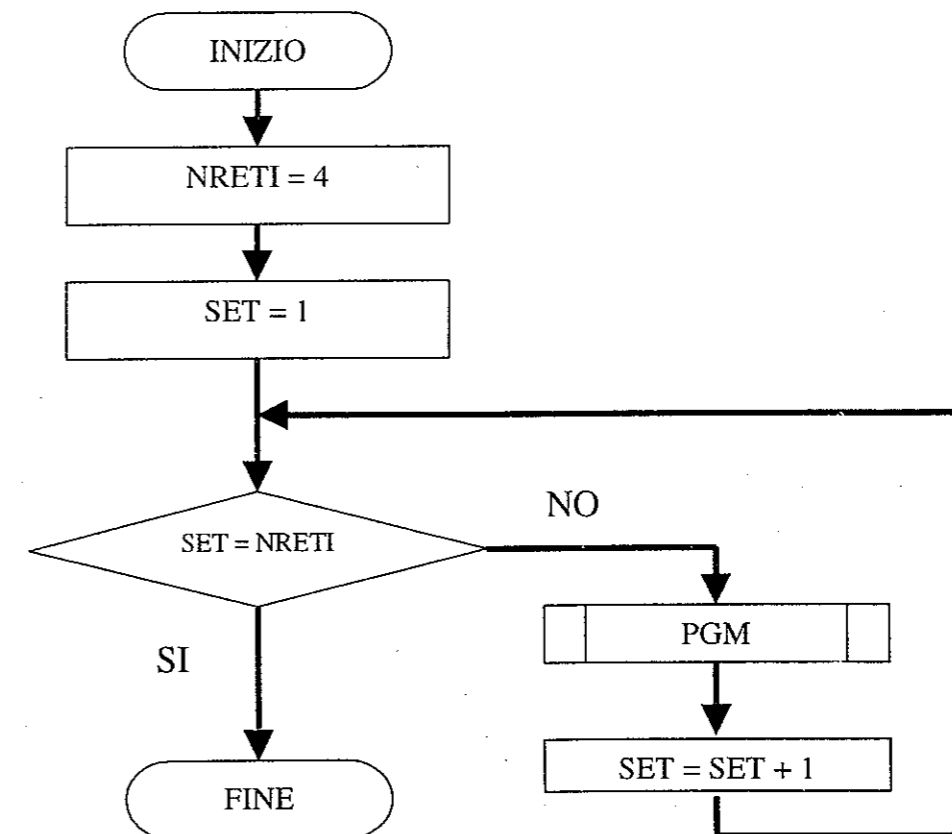


Fig. 6.12. diagramma di flusso del programma GANND (Genetic Algorithm for Neural Network Design).

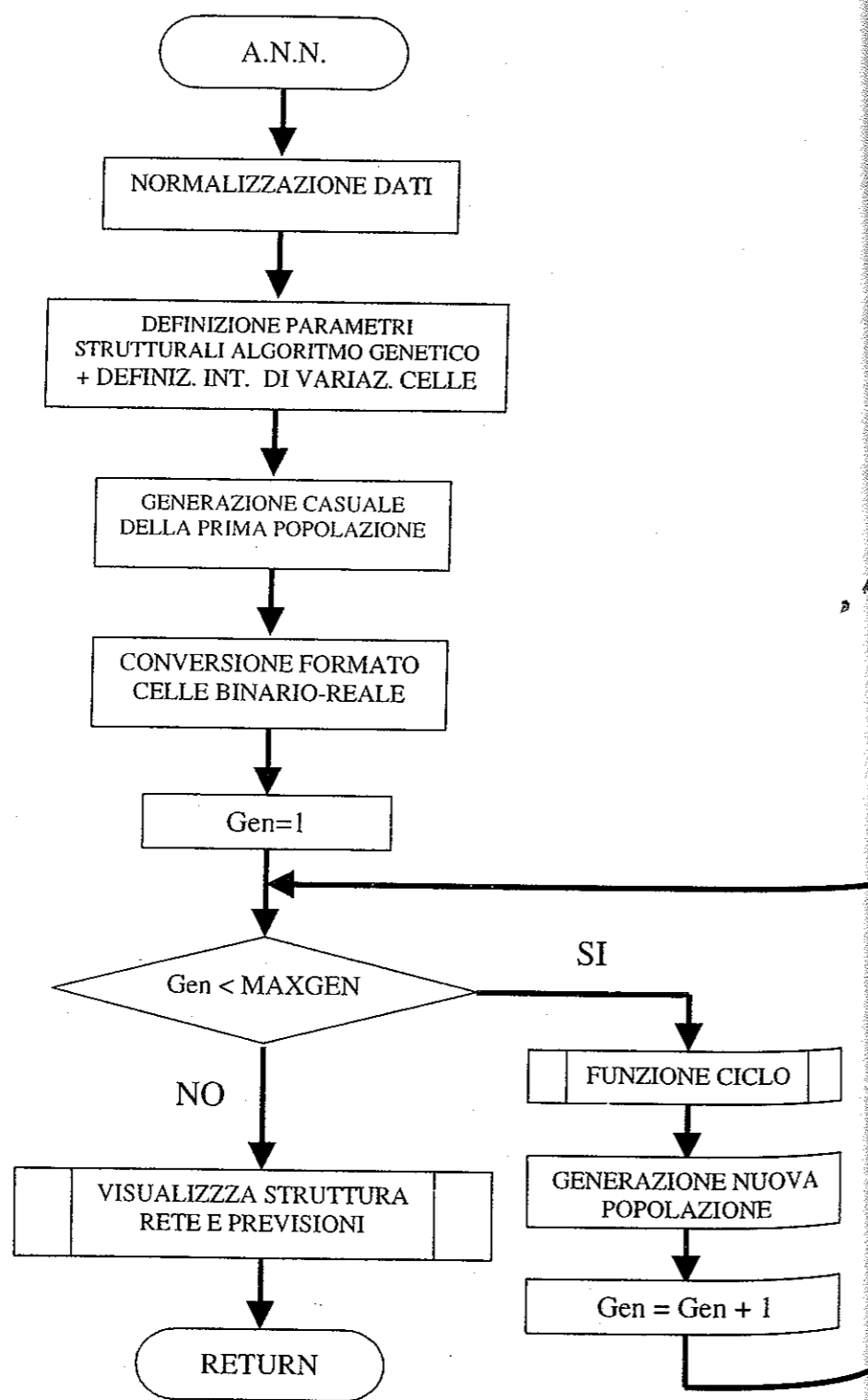


Fig. 6.13. Diagramma di flusso della procedura A.N.N. (Artificial Neural Network) del programma GANND.

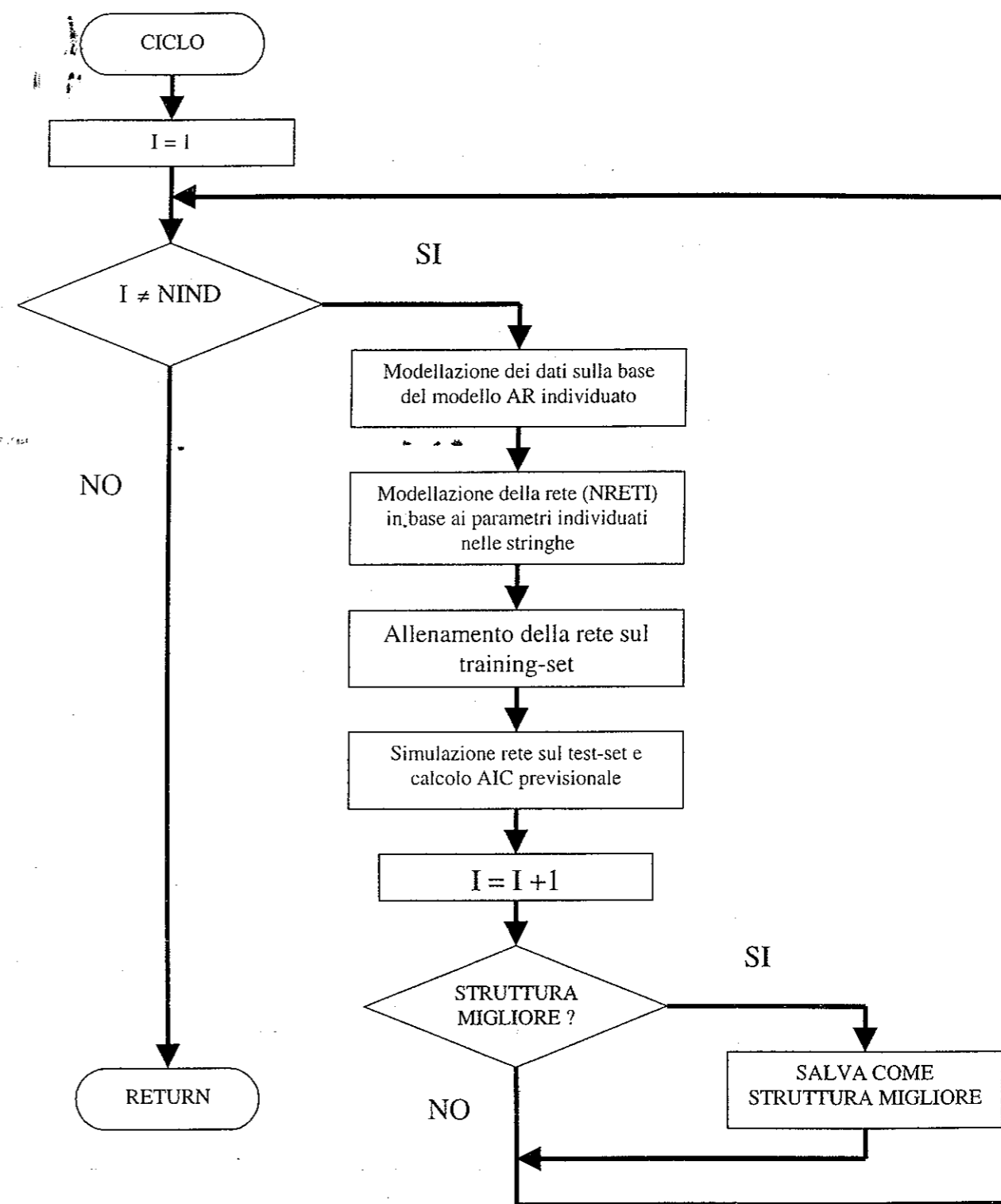


Fig. 6.14. Diagramma di flusso della procedura CICLO del programma GANND.

Come già detto, il programma implementa "a rotazione", una dopo l'altra, quattro tipologie di reti neurali: una feed forward a due strati con retropropagazione dell'errore, una rete di Elman e due reti a base radiale: una costruita nel modo classico visto in precedenza e una, la "Generalized Regression Network", in cui la *funzione di somma* è sostituita da una funzione *nprod* ottenuta moltiplicando i pesi dello strato nascosto per gli input dei neuroni dello stesso strato.

Come funzione di fitness per la scelta dell'individuo (struttura della rete e modello AR) migliore si è utilizzato l'indicatore *AIC* con le medesime motivazioni dei programmi precedenti.

I risultati ottenuti sono riportati graficamente nella Fig. 6.15.

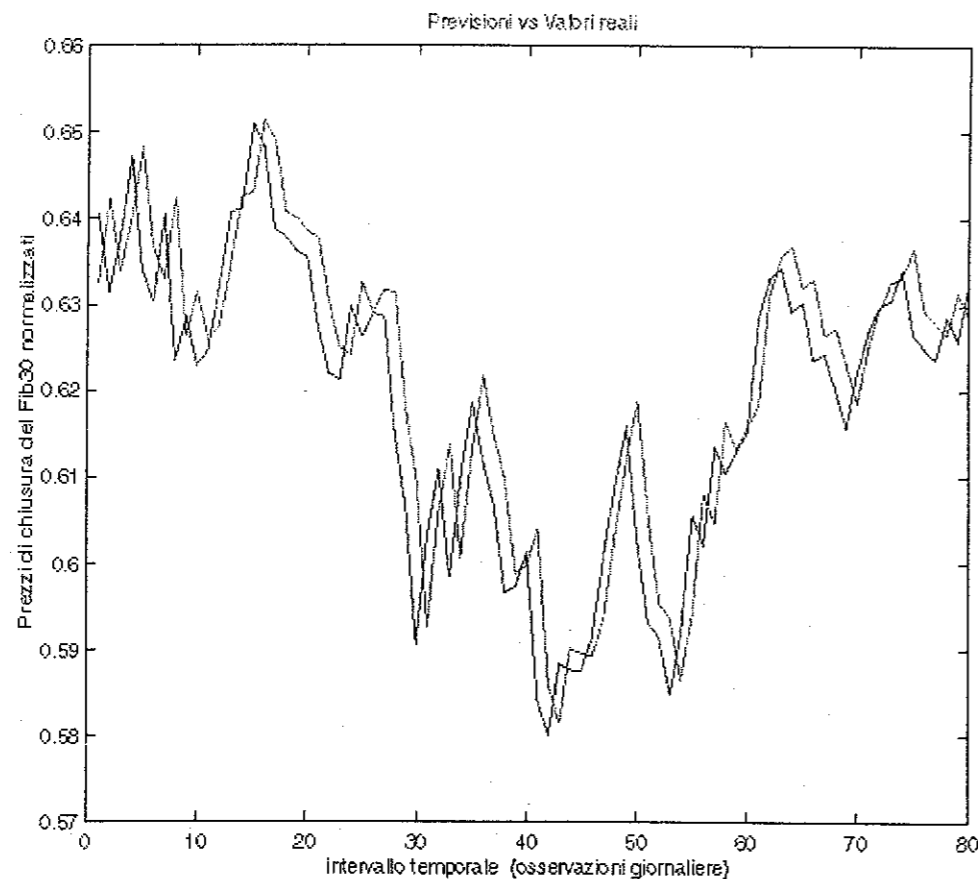


Fig. 6.15. Visualizzazione delle previsioni effettuate con il programma GANND (valori osservati vs valori previsti).

La scelta è caduta sulla rete di tipo a base radiale classico, il modello scelto è un $AR'(1)$.

I valori dell'*AIC* e dello scarto quadratico medio previsionale (rispettivamente -9.6381 e 6.3586⁻⁵) sono buoni ma le previsioni simulate evidenziano le difficoltà riscontrabili quando la serie storica ha una componente deterministica molto bassa: come si può vedere, l'output della rete ricalca la serie dei prezzi effettivi con un ritardo di un giorno. In sostanza, la rete addestrata sta suggerendo che la migliore ipotesi sul prezzo di domani è quello di oggi, vale a dire che la serie ha natura casuale.

La rete così addestrata, suggerendo che la migliore ipotesi sul prezzo di domani è quello di oggi, rispetta l'ipotesi dei Mercati Efficienti³ ma non effettua alcun tipo di previsione.

Appare allora chiaro come sia necessario apportare modifiche al programma onde cercare di ottenere risultati migliori.

Le strade compiute si dirigono in due direzioni diverse:

- Nota la primaria importanza dei valori forniti alla rete e della modellizzazione degli stessi, si è pensato di "passare" alla rete stessa input tratti da serie storiche di variabili differenti i cui andamenti fossero correlati al prezzo di chiusura del Fib30 e input determinati da indicatori di analisi tecnica quali le medie mobili ricavati da elaborazioni effettuate sul prezzo (tutte queste variabili sono riferite al giorno precedente quello di previsione).

La scelta delle variabili rilevanti è stata poi effettuata dall'algoritmo genetico con il medesimo meccanismo visto per la selezione dei ritardi del modello AR sottostante la serie storica dei prezzi.

L'elenco dettagliato degli input è il seguente: indice Comit, indice Mib generale, indice Mib30, indice Mibtel, prezzo minimo Fib30 a breve scadenza, prezzo massimo Fib30 a breve scadenza, prezzo di apertura Fib30 a breve scadenza, prezzo

³ L'ipotesi dei Mercati Efficienti suggerisce che l'andamento del prezzo dei titoli dovrebbe assumere la forma di un cammino casuale, e che la migliore previsione del prezzo che l'azione avrà domani è data dal valore odierno [Fama70].

di chiusura Fib30 a media scadenza, prezzo di chiusura Fib30 a lunga scadenza, media mobile lineare a 3 giorni, media mobile lineare a 5 giorni, media mobile lineare a 10 giorni, media mobile lineare a 15 giorni, media mobile lineare a 20 giorni, indice R.S.I.⁴ a 5 giorni e indice R.S.I. a 10 giorni.

Si è inoltre abbandonato il criterio puramente statistico dell'AIC previsionale per valutare il grado di fitness di ogni individuo, a favore di uno economico finanziario che selezioni l'individuo migliore in base al massimo profitto conseguito nell'operare le diverse operazioni di "trading" del titolo:

$$(6.5) \quad fitness = \frac{loss}{profit},$$

dove *loss* indica la perdita netta totale e *profit* indica il guadagno netto totale.

I risultati ottenuti con questa metodologia sono raffigurati nella Fig. 6.16.

Dalla figura si può notare come le previsioni non siano più ottenute dalla traslazione del valore del prezzo di chiusura relativo al periodo precedente ma le reti effettuano una vera previsione.

⁴ L' R.S.I. (Relative Strength Index) è uno degli indicatori più utilizzati nell'analisi tecnica per lo studio dell'andamento dei prezzi dei titoli.

$$R.S.I. = 100 - \frac{100}{\frac{\sum \Delta^+ i}{\sum \Delta^- i} + 1}, \text{ dove } \Delta^+ i = \text{media lineare dei rialzi e } \Delta^- i = \text{media lineare dei ribassi.}$$

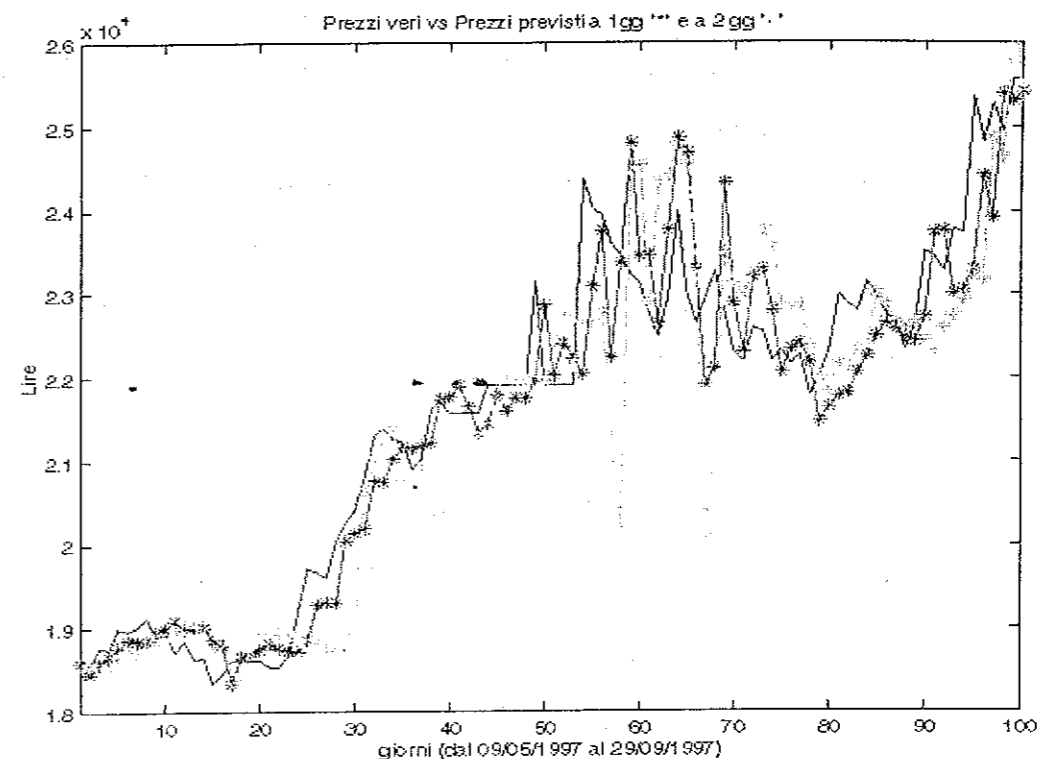


Fig. 6.16. Valori reali, previsioni a 1gg e previsioni a 2 gg relativi al periodo dal 09/05/1997 e 29/09/1997.

- La seconda via intrapresa per migliorare la procedura riguarda sia l'efficienza complessiva del programma, allo scopo di ridurre i tempi computazionali (e proprio in questa direzione bisogna inserire la ricodificazione dell'algoritmo genetico), sia la possibilità di prevedere Reti Neurali che procedano al calcolo dei pesi sinattici con lo scopo di minimizzare non lo scarto quadratico medio (funzione di errore) ma una funzione qualunque stabilita dall'utente.

In questo caso infatti è poco rilevante scegliere un individuo che massimizzi i guadagni se poi i pesi calcolati dalla rete non sono ottimizzati sul guadagno ma solo sullo scarto quadratico medio.

Ciò che è di fondamentale importanza, infatti, nelle previsioni dei prezzi di

titoli finanziari, è prevedere correttamente l'andamento del titolo (rialzo o ribasso) e non il suo valore effettivo.

Siccome MATLAB non permette di utilizzare le funzioni preimpostate nelle routines di sistema in questo modo, si è cercato di costruire (vedi il prossimo capitolo) alcuni tipi di rete come "feed forward a due strati" o "Elman", che ottimizzino una funzione scelta dall'utente.

Inoltre, come già accennato nel secondo capitolo, ottimizzare una funzione quando i parametri sono parecchi e le condizioni di regolarità non sono rispettate pone di fronte a problemi non facilmente risolvibili con le metodologie classiche.

Si è cercato pertanto di ottimizzare queste procedure utilizzando Algoritmi Genetici che, come detto, non richiedono l'imposizione delle condizioni di regolarità delle funzioni che descrivono il problema stesso.

Purtroppo questo tipo di reti non è stato utilizzato nel programma GANND, in quanto l'utilizzo degli Algoritmi Genetici nella procedura di ottimizzazione della funzione di errore inserita nella struttura della rete stessa, aumenta notevolmente il tempo di calcolo oltre limiti accettabili.

Soltanto l'utilizzo di elaboratori che permettano di sfruttare diverse CPU in modo parallelo consentirà in futuro di apprezzare questo tipo di strumento applicandolo alle previsioni finanziarie.

7. OTTIMIZZAZIONE GENETICA

La necessità di implementazione delle Reti Neurali che utilizzano una funzione di errore scelta dall'operatore inevitabilmente richiede la trattazione del problema dell'ottimizzazione di funzioni.

Come già detto, infatti, la funzione di errore, che costituisce uno dei componenti fondamentali dei neuroni che compongono una Rete Neurale artificiale, generalmente è rappresentata dallo scarto quadratico medio calcolato sul campione di allenamento dei dati messi a disposizione della rete.

Può capitare talvolta che l'ottimizzazione di questa funzione (minimo globale) non rappresenti pienamente l'obiettivo di un operatore interessato, invece, solo ai propri guadagni (vedi capitolo precedente) o a prevedere i picchi di andamento futuro di una certa serie storica.

In tutti questi casi, comunque, l'algoritmo che si vuole inserire come funzione di errore nella struttura neurale deve essere sempre ottimizzato.

Inoltre, il fatto che la funzione utilizzata debba essere massimizzata o minimizzata non cambia i termini del problema poiché il massimo di una funzione può essere ricavato come il minimo dell'opposto o del reciproco della funzione stessa, e viceversa.

Di seguito, pertanto, verranno utilizzati indifferentemente i termini di ottimizzazione o minimizzazione di una funzione per indicare lo stesso problema.

In economia, in ingegneria e nelle scienze in generale, i problemi di decisione sono modellati frequentemente ottimizzando il valore della funzione obiettivo sotto i vincoli dichiarati di possibilità.

Così, per esempio, un operatore di borsa può desiderare di ottimizzare la media geometrica di alcuni indici di borsa allo scopo di calcolare indicatori di analisi tecnica e poter misurare le "performance" di un determinato portafoglio di titoli, oppure un ingegnere durante la progettazione di un'ala di un aeroplano costituita da diversi tipi di materiali può aver bisogno di sapere con quali angoli connettere i vari materiali per ottimizzare la resistenza e il peso complessivo dell'ala, o ancora un chimico può avere la necessità di ottimizzare la decomposizione della Gaussiana di uno spettro complesso Mössbauer nelle singole Voigtiane che lo compongono.

Insomma, l'importanza attribuibile all'ottimizzazione globale, intesa come ricerca dell'ottimo generale di una data funzione soggetta o meno a determinati vincoli, nella vita di ogni giorno degli individui non può essere messa assolutamente in discussione.

In letteratura esistono numerosi approcci, sia numerici che analitici, che affrontano questo tipo di problematica. Anche se questi approcci vengono utilizzati con pieno successo quotidianamente nei quattro angoli del globo, tuttavia essi si stanno dimostrando insufficienti per affrontare fenomeni molto difficili la cui complessità deriva o dal tipo di interazione non lineare e non regolare tra le variabili in gioco o dalla moltitudine di questi o da entrambi i fattori.

Solitamente tali approcci prevedono condizioni di regolarità (continuità, derivabilità, etc...) delle funzioni che non sempre vengono rispettate dalle interazioni in gioco. Spesso si conosce o si può modellare una interazione locale ma riesce pressoché impossibile fornire un modello globale di comportamento, oppure la relazione tra le variabili può essere talmente complessa che non si riesce a determinare una funzione capace di descriverla, né le tecniche di approssimazione riescono a fornire risultati apprezzabili.

È evidente che gli approcci attualmente utilizzati riescano a fornire risposte solo parziali quando la complessità del problema risulta elevata. Nella maggior parte di questo tipo di approcci, una prima fase analitica di riduzione della complessità o

della determinazione delle relazioni assume un ruolo preponderante rispetto alla fase successiva di determinazione computazionale della soluzione. L'aspetto computazionale assume cioè solo un aspetto di 'servizio', un modo per velocizzare i processi ma non entra minimamente nella descrizione del fenomeno sotto esame.

In questi ultimi anni, invece, si stanno sempre più affermando approcci di calcolo in cui l'aspetto computazionale risulta prevalente rispetto a quello analitico (che spesso è assente). Un'ampia classe di questi approcci cerca di emulare i processi naturali ('soft') dell'evoluzione, della conoscenza, della valutazione e dell'interazione sociale per codificare processi computazionali di carattere biomorfo. Tra questi vanno ricordati gli approcci che direttamente si richiamano a processi biologici (Algoritmi Genetici, Reti Neurali, Automi Cellulari, Logica Fuzzy) o a processi fisici (Simulated Annealing) o a procedure che integrano i diversi approcci (Algoritmi Ibridi).

In questo capitolo verrà dapprima presentato un confronto tra due tecniche globali differenti di ottimizzazione: una metodologia classica di tipo deterministico (Quasi-Newton) e una di tipo probabilistico evoluta geneticamente; successivamente verrà presentata la simulazione di due tipologie di Reti Neurali in cui la funzione matematica associata viene ottimizzata mediante lo strumento più efficiente fra quelli scelti.

7.1. LA FUNZIONE DI ROSENBROCK

Come detto nel secondo capitolo, quando ci si trova a dover determinare il valore che ottimizza una determinata funzione, la soluzione, salvo il caso di poche classi di funzioni, deve essere determinata per via numerica.

In quella sede sono stati presentati la maggior parte dei principali algoritmi di ricerca e di ottimizzazione classica capaci di effettuare la ricerca tra le potenziali soluzioni del problema fino a determinarne quella ottimale.

Tutte queste procedure rientrano tra i cosiddetti algoritmi di calcolo deterministico che soffrono, chi in misura maggiore e chi in misura minore, della necessità di utilizzare le informazioni sul gradiente della funzione¹.

Esistono altre due classi di algoritmi di ricerca: gli algoritmi "enumerativi" e quelli randomizzati.

I primi testano tutti i possibili valori del parametro in questione (equivalente alla lettura di tutte le parole di un vocabolario per trovare quella cercata!). Tale tecnica è improponibile per spazi multiparametrici.

Si pensi, per esempio, a uno spazio parametrico a 10 parametri $\theta_1 \dots \theta_{10}$ in cui ogni parametro può assumere solo 10 valori (discreto...non continuo). Il numero di prove da effettuare sarebbe già notevole (pari a 10^{10} controlli diversi). La situazione diventa drammatica se bisogna lavorare su spazi parametrici continui in cui il numero di possibili valori per ogni parametro diventa infinito (o comunque molto elevato anche in caso di discretizzazione dello spazio parametrico).

Negli approcci randomizzati (o casualizzati²), invece, vengono definiti uno schema di ricerca e delle regole iterative (pur mantenendo la natura stocastica di tali regole).

¹ Queste informazioni infatti non sempre sono disponibili e, comunque, spesso per ottenerle è necessario parecchio tempo.

² Da non confondere con gli approcci "casuali" in cui non esiste nessuno schema di ricerca; la soluzione viene cercata a caso dopo aver campionato lo spazio parametrico con tecniche statistiche (assimilabile al tentativo di cercare al buio un oggetto mettendo le mani un po' dove capita).

A questa classe appartengono gli Algoritmi Genetici e il Simulated Annealing in cui una selezione casuale viene introdotta per guidare una ricerca altamente esplorativa e con un obiettivo ben preciso entro uno spazio parametrico opportunamente codificato.

Gli Algoritmi Genetici effettuano tale ricerca emulando l'evoluzione genetica di una popolazione (la soluzione cercata è l'individuo geneticamente migliore) mentre il Simulated Annealing emula il raffreddamento di un sistema fisico che tende verso lo stato a energia più bassa. È da sottolineare il fatto che "ricerca casualizzata" non vuol dire "senza direzione". La direzione di ricerca viene scelta attraverso un approccio probabilistico ma deve rispondere a criteri di selezione deterministici.

Vengono di seguito analizzati due metodi differenti. Il primo, metodo di Quasi-Newton, appartiene alla classe delle tecniche deterministiche, il secondo, la procedura genetica, appartiene a quella delle tecniche probabilistiche.

L'analisi è volta a determinare l'efficienza delle due procedure di fronte a una funzione che, per il suo andamento risulta un buon test di verifica per qualunque tipo di metodologia utilizzata: la cosiddetta funzione di "Rosenbrock" (detta anche funzione a banana).

Questa funzione è molto significativa in quanto presenta il suo minimo all'interno di una valle piana, lunga, stretta e parabolica. Trovare la valle è insignificante ai fini della convergenza verso l'ottimo; soltanto il minimo all'interno di essa permette, infatti, di valutare la prestazione delle diverse procedure.

Quella di Rosenbrock è una funzione bidimensionale (a due variabili indipendenti)³:

$$(7.1) \quad F_{Rosenbrock} = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2.$$

³ Quando si passa ad esaminare il caso concreto, per rendere più leggibili i grafici creati con MATLAB, verrà modificata la notazione delle variabili indipendenti $\theta_1, \dots, \theta_n$ con x_1, \dots, x_n .

Il suo punto di minimo è: $x_1 = 1$, $x_2 = 1$, dove $F_{Rosenbrock} = 0$.

La caratteristica di questa funzione che la rende molto indicata allo scopo è quella di presentare la matrice Hessiana singolare lungo la parabola di equazione:

$$(7.2) \quad 200x_1^2 - 200x_2 + 1 = 0$$

molto vicina al fondo della valle della funzione (cosa che ostacola i vari algoritmi di ottimizzazione).

Le Figg. 7.1 e 7.2 visualizzano da diverse angolazioni l'andamento della funzione.

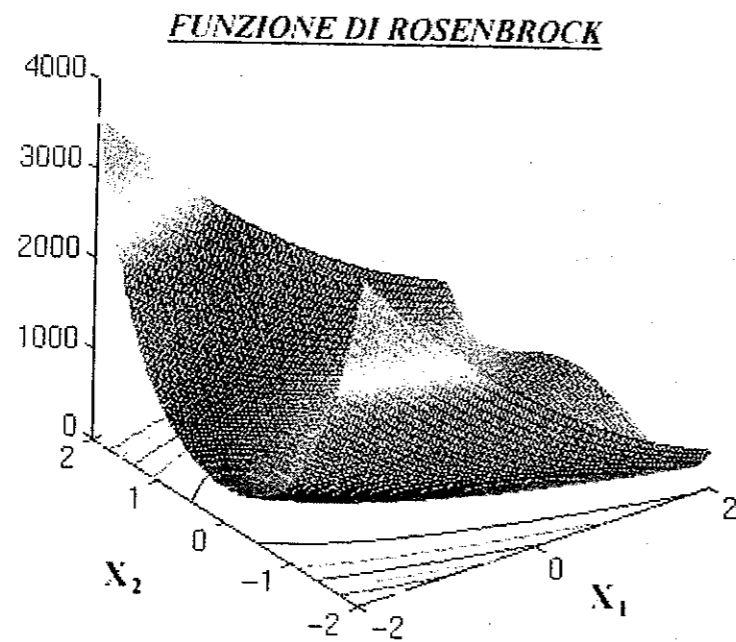


Fig. 7.1. Funzione di Rosenbrock.

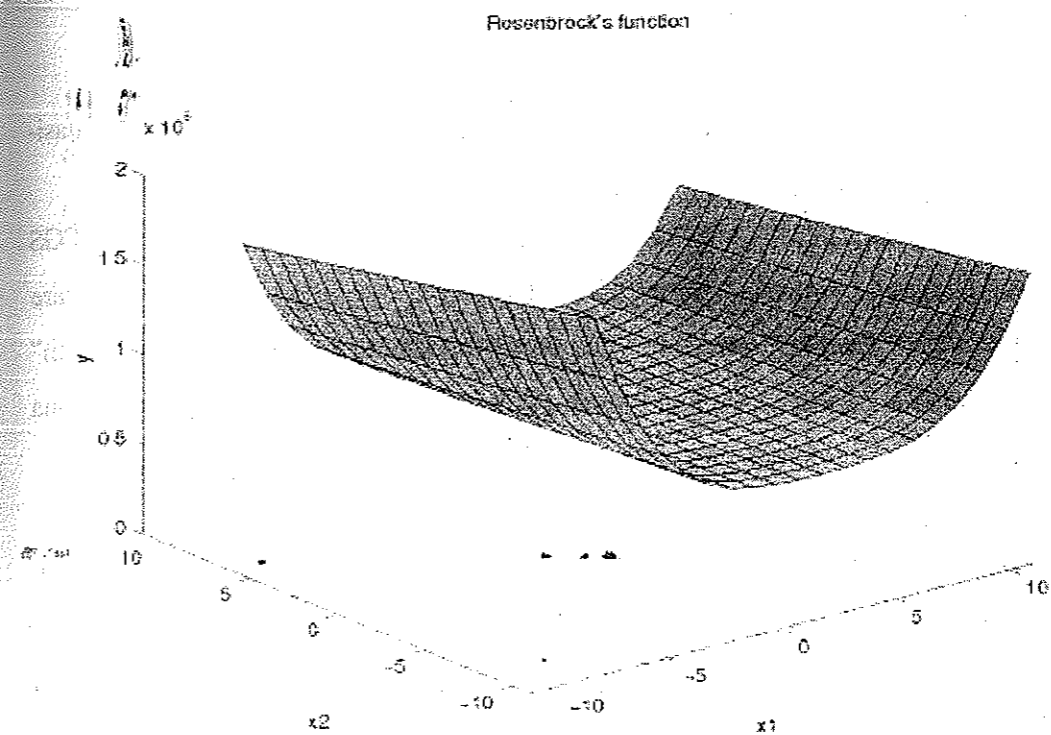


Fig. 7.2. Funzione di Rosenbrock (la "valle").

Esaminiamo ora il tipo di sintassi utilizzato in MATLAB nella routine di sistema "Optimization Toolbox" per il calcolo del minimo (massimo) di una funzione non vincolata: "*fminunc*" (minimum of an unconstrained multivariable function).

"*Fminunc*" è dunque una funzione preconstituita che permette di valutare il valore di θ nel punto in cui viene rilevato il valore minimo di $f(\theta)$.

La sintassi è la seguente:

$$\theta = \text{fminunc}(\text{fun}, \theta_0, \text{options})$$

dove:

θ = minimo globale trovato della funzione *fun*;

fun = funzione non-lineare e non soggetta a vincoli da minimizzare;

θ_0 = stima iniziale (o punto iniziale) da cui iniziare la ricerca;

options = parametri di definizione dell'algoritmo di ottimizzazione.

Questa procedura minimizza la funzione "fun" attraverso la tecnica di "Quasi-Newton".

Questa, per il calcolo della direzione di discesa d utilizza la tecnica BFGS (Broyden, Fletcher, Goldfarb, Shanno 1970, anche se esiste la possibilità di utilizzare la tecnica DFP di Davidon, Fletcher, Powell, 1963) e una procedura di "line-search" di tipo polinomiale misto quadratico-cubica (con la possibilità di implementare una procedura di tipo polinomiale cubica qualora sia disponibile il calcolo del gradiente).

La metodologia BFGS [Broy70, Flet70, Gol70, Sha70] calcola la matrice Hessiana (H) con la formula:

$$(7.3) \quad H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k s_k^T H_k}{s_k^T H_k s_k};$$

dove:

$$s_k = \theta_{k+1} - \theta_k$$

$$q_k = \nabla f(\theta_{k+1}) - \nabla f(\theta_k)$$

e in cui come H_0 viene utilizzata la matrice identità assicurando in questo modo la condizione richiesta per la matrice H , e cioè che sia definita positivamente (questo per assicurare che la direzione di ricerca sia sempre una direzione di discesa).

La matrice H a ogni iterazione successiva viene mantenuta positiva assicurandosi che $q_k^T s_k$ sia sempre positiva:

$$(7.4) \quad q_k^T s_k = \lambda_k (\nabla f(\theta_{k+1})^T d - \nabla f(x_k)^T d)$$

dove λ_k corrisponde allo Step-size (lunghezza del passo percorso lungo la discesa d_k alla k -esima iterazione del processo) e il secondo fattore corrisponde alla combinazione della direzione di discesa d con l'ultimo passo iterato nel processo e quello successivo.

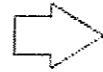
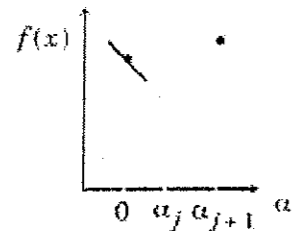
Per permettere questo, d_k viene calcolato in modo che λ_k e $-\nabla f(\theta_k)^T d$ siano sempre positivi e che la forma quadratica (eventualmente definita negativa) $\nabla f(\theta_{k+1})^T d$ risulti minore in valore assoluto.

L'altra tecnica di cui MATLAB permette l'utilizzo è quella suggerita da Davidon, Fletcher e Powell [Dav59, Flet63] che è basata sulla precedente con l'unica differenza della sostituzione di q_k con s_k nel calcolo della matrice Hessiana.

Come detto la procedura di "line-search" qui implementata è una procedura di tipo polinomiale mista quadratico-cubica: si cercano tre punti nel dominio della funzione che contengano il minimo senza la necessità di disporre di alcuna informazione sul gradiente della funzione (e per questo si utilizza la procedura di tipo polinomiale quadratica), dopodiché si interpolano tali punti con la procedura polinomiale di tipo cubico in quanto risulta più accurata della precedente.

La procedura di "line-search" è rappresentata graficamente in Fig. 7.3.

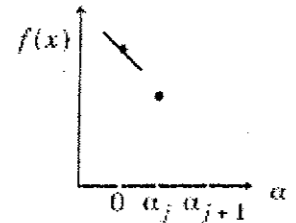
Case 1: $f(x_j) \geq f(x_k)$



Riduci Lunghezza Step

$$\alpha_{j+1} = \alpha_q$$

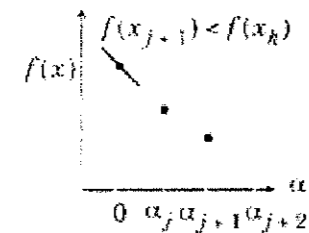
Case 2: $f(x_j) < f(x_k)$



Aumenta Lunghezza Step

$$\alpha_{j+1} = 1.2\alpha_q$$

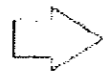
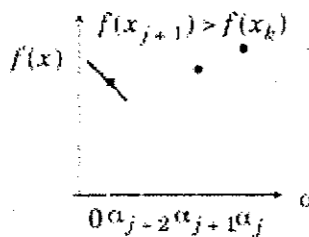
Case 3:



Aumenta Lunghezza Step

$$\alpha_{j+2} = \max\{1.2\alpha_q, 2\alpha_{j+1}\}$$

Case 4:



Riduci Lunghezza Step

$$\alpha_{j+2} = \alpha_c$$

Fig. 7.3. Visualizzazione della procedura di "line-search" implementata in MATLAB.

I termini α_c e α_q si riferiscono al minimo ottenuto rispettivamente dall'interpolazione (o estrapolazione) cubica e quadratica. Se la funzione fosse

fortemente non lineare α_c e α_q potrebbero risultare negativi, in questo caso la procedura li fisserebbe pari a $2\alpha_k$, in modo da mantenerli sempre positivi.

In Fig. 7.3 i casi 1 e 2 mostrano come la procedura utilizzi l'interpolazione quadratica per stimare il terzo punto che racchiude il minimo (avendo a disposizione anche il gradiente nel primo punto $f(0)$). I casi 3 e 4 mostrano come viene cambiata la lunghezza del passo (Step) quando la procedura implementata nei casi 1 e 2 fallisce. Quando finalmente il minimo è racchiuso tra i punti, l'interpolazione cubica viene implementata utilizzando i tre punti a disposizione e il gradiente calcolato sul primo punto $f(0)$. Se il punto così interpolato risulta maggiore degli altri tre utilizzati per l'operazione, viene sostituito con il punto che rende minore il valore della funzione.

Seguendo questa procedura a ogni iterazione la matrice Hessiana viene aggiornata analogamente a quanto avviene con l'interpolazione cubica pura (che richiederebbe informazioni sul gradiente qui non disponibili).

Qualora invece sia disponibile l'informazione sul gradiente, la procedura di "line-search" implementata da MATLAB nella function *fminunc*, è quella polinomiale cubica pura.

Dopo aver descritto come viene effettuata l'ottimizzazione della funzione con la procedura di tipo classico, passiamo ora a esaminare come è stata costruita la procedura genetica di tipo probabilistico "casualizzato".

MATLAB non possiede alcuna "function" per ottimizzare una funzione con questo tipo di metodologie: si è proceduto pertanto alla creazione del programma GAO (Genetic Algorithm for Optimization) il cui diagramma di flusso è rappresentato in Figg. 7.4 e 7.5.

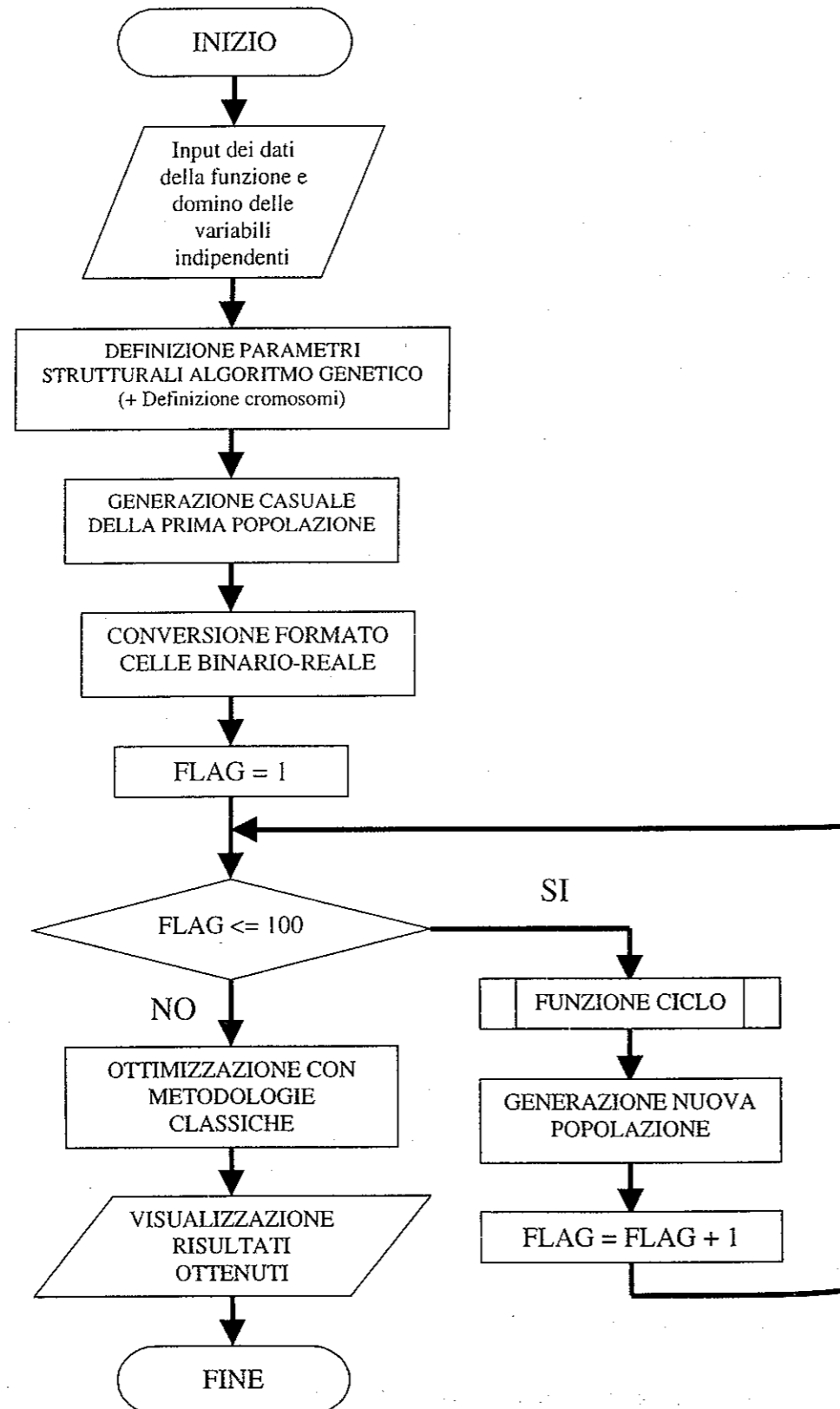


Fig. 7.4. Diagramma di flusso del programma GAO.

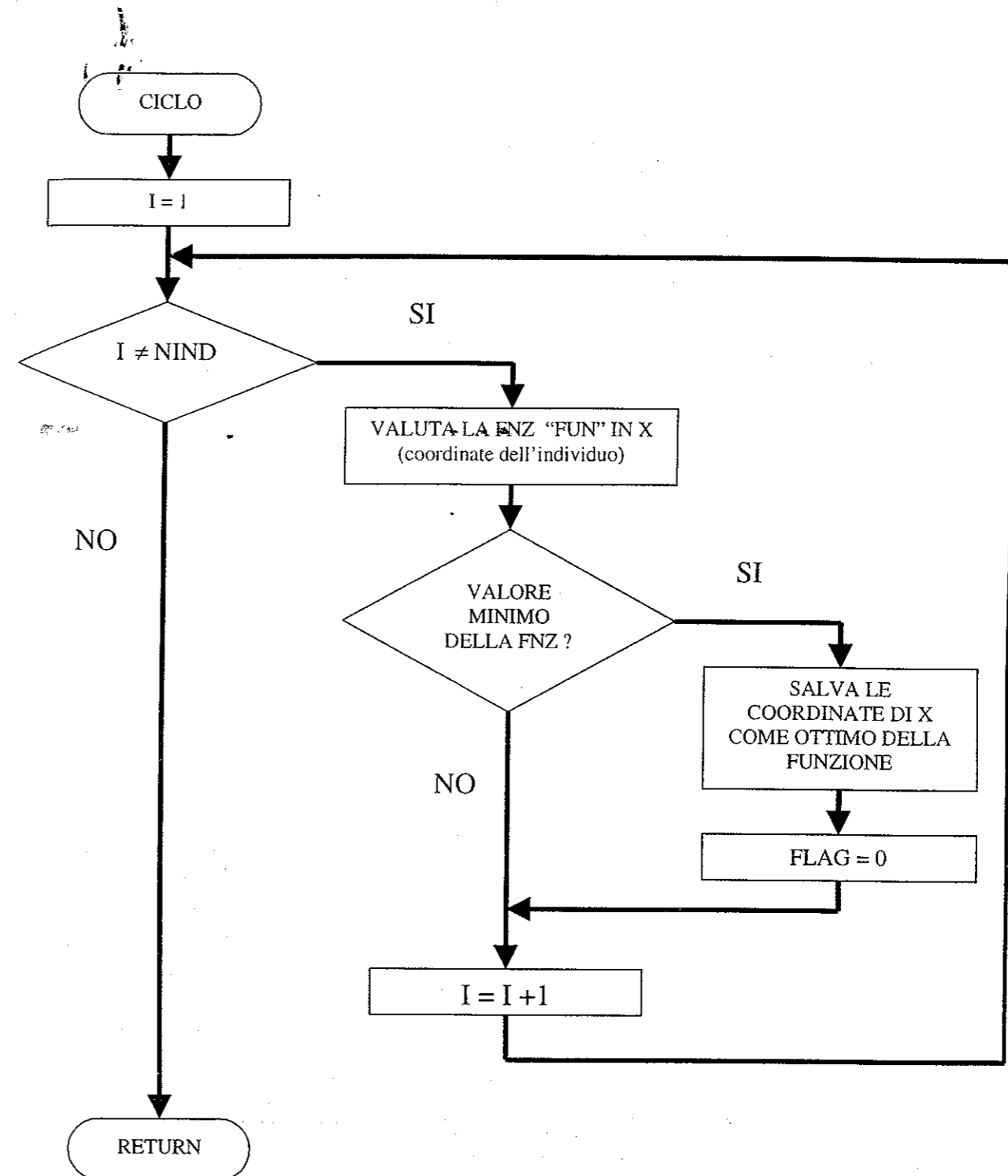


Fig. 7.5. Diagramma di flusso del sottoprogramma "funzione ciclo" del programma GAO.

La struttura del programma risulta sotto certi aspetti particolarmente innovativa rispetto alle altre applicazioni genetiche studiate nei precedenti capitoli.

La principale novità riguarda senz'altro la codifica degli individui della popolazione che abbandona una struttura di tipo rigido per adottarne una di tipo flessibile.

Al programma, infatti, è associata una procedura di caricamento dei dati che prevede l'inserimento del dominio (campo di ricerca su cui effettuare l'analisi) della funzione e della precisione (numero di cifre decimali con cui restituire i valori in output) associata alle variabili indipendenti utilizzate.

Tale procedura è riservata all'operatore che deve inserire in un file tutti i requisiti richiesti dalla funzione da ottimizzare.

Nel caso della funzione di Rosenbrock, per esempio, i parametri forniti sono stati i seguenti:

- x_1 : valore minimo possibile - 10, valore massimo possibile 10, numero di cifre decimali di precisione 6 (1e-06);
- x_2 : valore minimo possibile - 10, valore massimo possibile 10, numero di cifre decimali di precisione 6 (1e-06);

Si è lavorato quindi sul campo di valori evidenziato in Fig. 7.2.

Una volta caricati i dati, il programma calcola la lunghezza di ogni individuo della popolazione binaria con la seguente formula:

$$(7.5) \quad nbit = \sum_{i=1}^n \text{round} \left(0.5 + \log_2 \left(\frac{\max(x_i) - \min(x_i)}{\text{precisione}} \right) \right)$$

Questo permette di generare un individuo costituito da $nbit$ celle (che possono assumere valore 0 o 1) suddivisibili in n (numero di variabili indipendenti) parti, una per ogni variabile. Ognuna di queste parti, siccome vengono codificate con un sistema di tipo binario, permette di rappresentare il dominio di ciascuna variabile con la precisione fornita.

Nell'esempio proposto ogni individuo viene generato con una lunghezza complessiva di 50 celle⁴: le prime 25 celle rappresentano il campo di valori assumibili dalla variabile x_1 mentre le seconde 25 celle rappresentano il campo di valori assumibili dalla variabile x_2 .

Dopo aver generato in modo casuale la prima generazione, l'algoritmo converte la popolazione binaria in una popolazione reale in cui a ogni individuo saranno associati n valori.

Un'altra significativa differenza nella struttura del programma rispetto a quella dei precedenti consiste nella procedura utilizzata come criterio di stop: in questo caso il processo iterativo ha termine nel momento in cui per 100 generazioni successive non vi è stato alcun miglioramento nel valore di fitness⁵.

In base alle prove effettuate, infatti, si è potuto notare che 100 iterazioni (una per ogni generazione) senza alcun miglioramento nei valori di fitness assicurano la convergenza dell'algoritmo.

Al termine del programma è inoltre previsto il calcolo del minimo con la metodologia classica (eseguito con la funzione precostituita di MATLAB *fminunc*) utilizzando la formula di BFGS associata alla procedura mista di line-search quadratico-cubica.

Come punto di partenza vengono in questo caso utilizzati i valori associati al primo individuo della prima popolazione generata casualmente nell'Algoritmo Genetico. Ciò permette di non penalizzare nessuna procedura, utilizzando come punto di partenza il medesimo valore per entrambe e permettendo in questo modo di analizzare comparativamente i risultati.

⁴ Una lunghezza di 50 celle per rappresentare ogni individuo comporta la ricerca dell'ottimo tra ben 2^{50} possibili soluzioni.

⁵ Come funzione di fitness, ovviamente, viene utilizzata la funzione da minimizzare.

Il programma GAO è stato "lanciato" 1000 volte. I risultati ottenuti sono riportati in tabella 7.1.

ERRORE (minimo trovato - minimo reale)	PROCEDURA CLASSICA	ALGORITMO GENETICO
$\Delta f(x) < 10^{-5}$	775	28
$10^{-5} < \Delta f(x) < 10^{-4}$	8	47
$10^{-4} < \Delta f(x) < 10^{-3}$	24	126
$10^{-3} < \Delta f(x) < 10^{-2}$	29	381
$10^{-2} < \Delta f(x) < 10^{-1}$	48	404
$10^{-1} < \Delta f(x) < 10^0$	100	14
$10^0 < \Delta f(x) < 10^1$	10	-
$10^1 < \Delta f(x) < 10^2$	3	-
$\Delta f(x) > 10^2$	3	-
TOTALE	1000	1000

Tabella 7.1. Risultati del programma GAO (Genetic Algorithm for Optimization).

In questa tabella, nelle righe 2-10 vengono evidenziati il numero di prove in cui l'errore (valore di minimo trovato - valore di minimo reale) assume un valore compreso nell'intervallo descritto nella prima colonna.

Si ricorda che nel punto di minimo la funzione assume valore 0; ciò rende indifferente parlare di errore o minimo stimato della funzione.

La media complessiva dell'errore delle 1000 simulazioni con le due procedure è:

- con la procedura classica : 0.845 ;
- con l'Algoritmo Genetico : 0.015 ;

Le deviazioni standard dell'errore sono:

- con la procedura classica : 15.275 ;
- con l'Algoritmo Genetico : 0.022 ;

In Fig. 7.6 vengono mostrati gli istogrammi relativi ai degli errori ottenuti dal programma di ottimizzazione classica e genetica (in scala logaritmica).

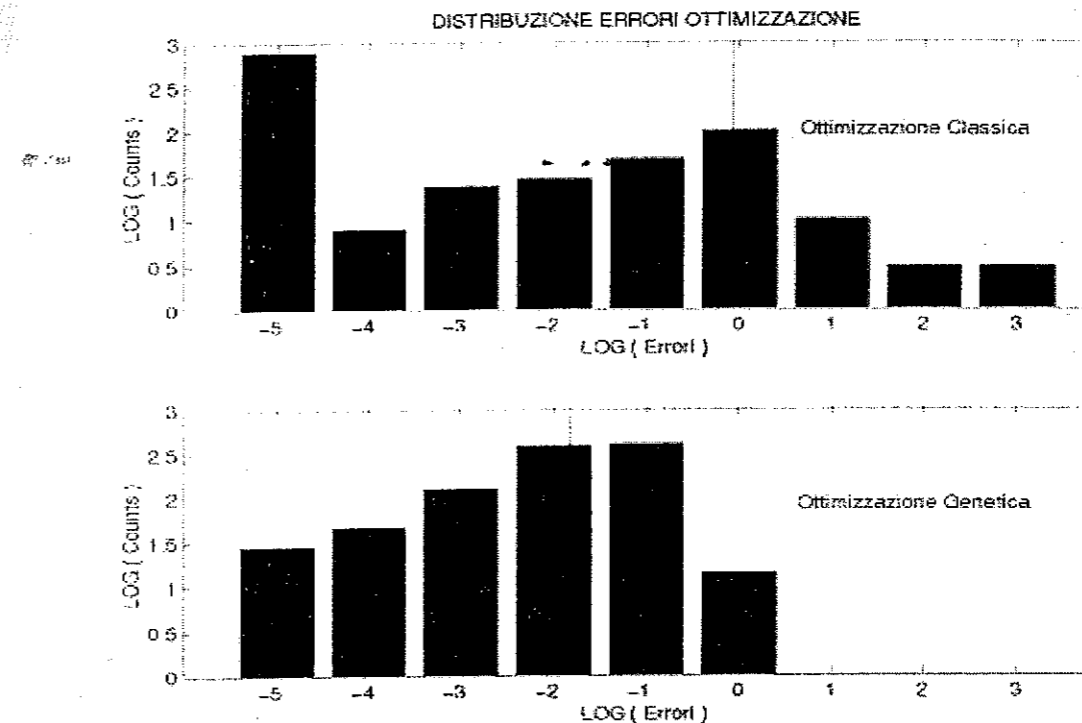


Fig.7.6. Istogrammi relativi ai risultati del programma di Ottimizzazione Classica vs Genetica. Le due linee azzurre verticali rappresentano la media dell'errore (con scala logaritmica).

Il valore medio assunto dalle variabili indipendenti, che nel punto di minimo è $x_1 = 1$ e $x_2 = 1$, è il seguente:

- con la procedura classica : $x_1 = 0.936$ e $x_2 = 0.928$;
- con l'Algoritmo Genetico : $x_1 = 0.979$ e $x_2 = 0.967$;

Le rispettive deviazioni standard sono:

- con la procedura classica : $std(x_1) = 0.219$ e $std(x_2) = 0.406$;
- con l'Algoritmo Genetico : $std(x_1) = 0.095$ e $std(x_2) = 0.186$;

I risultati mostrano come di fronte a una funzione complessa gli algoritmi classici siano poco efficienti. Anche dopo l'eliminazione dei casi in cui i valori della funzione ottenuti con gli algoritmi classici risultano maggiori di uno⁶, la media complessiva arriva a 0.041 con un errore standard pari a 0.139. Tali risultati restano ancora peggiori di quelli ottenuti con l'Algoritmo Genetico.

Questo dimostra che, nonostante l'esclusione degli "outliers" dai risultati ottenuti con tecniche classiche, quelli delle procedure genetiche risultano mediamente ancora più efficienti.

Un altro dato importante da notare è la distanza media tra il punto di minimo della funzione [1,1] e il valore trovato dai due algoritmi:

- con la procedura classica : $dist = 0.18$;
- con l'Algoritmo Genetico : $dist = 0.16$;

Le rispettive deviazioni standard sono:

- con la procedura classica : $std(dist) = 0.44$;
- con l'Algoritmo Genetico : $std(dist) = 0.14$;

Gli algoritmi classici, dunque, si dimostrano non molto robusti e poco efficienti in problemi di elevata complessità sebbene vengano quotidianamente utilizzati con successo per risolvere problemi a bassa complessità oppure di carattere specifico.

⁶ Il motivo per cui si è proceduto all'eliminazione di tali risultati era la possibilità che i casi in cui la funzione assumeva valori molto alti, potessero alterare la media oltre misura (outliers).

Se si potesse costruire un grafico dell'efficienza degli algoritmi di ottimizzazione in funzione della complessità del problema si troverebbe qualcosa del tipo rappresentato in Fig. 7.7, in cui si nota che l'efficienza degli algoritmi casualizzati (Algoritmi Genetici, GA, e Simulated Annealing, SA) si mantiene alta su un ampio spettro di problematiche; quelli puramente casuali mostrano una efficienza molto bassa; altissima efficienza è ottenibile da algoritmi fortemente specializzati ma solo per classi di problemi specificatamente definiti.

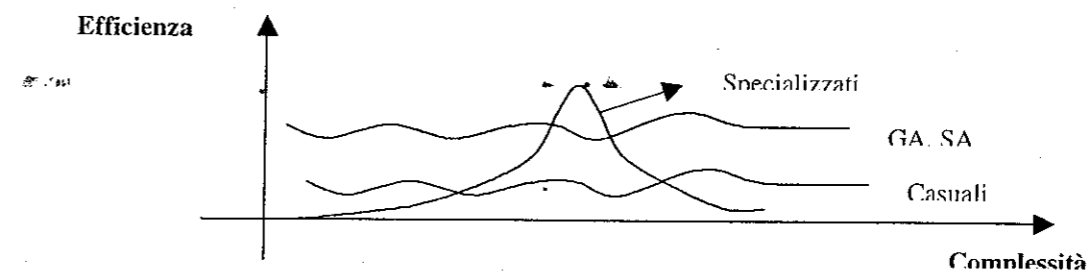


Fig. 7.7. Efficienza vs Complessità per alcune classi di algoritmi di ottimizzazione.

7.2. SIMULAZIONE DI RETI NEURALI

L'utilizzo di Algoritmi Genetici per ottimizzare una determinata funzione si è dimostrato più efficiente delle tecniche classiche quando la complessità del problema è alta.

Questa è la motivazione principale per cui nelle simulazioni che seguiranno si è deciso di implementare questa tecnica di minimizzazione della funzione di errore associata alla struttura neurale: quando si lavora con Reti Neurali, il numero di variabili indipendenti è molto alto comprendendo tutti i pesi sinattici delle connessioni tra i differenti neuroni. Si tratta, cioè, di un problema di alta complessità.

Per la simulazione di una Rete Neurale si è utilizzato il programma GAO esaminato nel precedente paragrafo. Ovviamente un po' di attenzione deve essere prestata nella definizione delle variabili e della funzione di fitness (che in questo caso corrisponderà alla funzione di errore associata alla rete).

La definizione delle variabili deve essere calibrata con molta attenzione quando i neuroni nella rete (e quindi i pesi associati alle connessioni tra di loro) sono parecchi. Bisogna infatti ricordare che in input il programma richiede il campo di ricerca di ogni variabile e sarebbe troppo dispersivo non calibrare in modo corretto questi valori.

Per questa realizzazione, sono state fatte alcune prove procedendo anche per tentativi sulla base delle informazioni ottenibili utilizzando le procedure di MATLAB precedentemente discusse (vedi capitolo sesto).

La codifica genetica degli individui coincide con quella esaminata nel capitolo precedente inserendo in ogni stringa la definizione delle caratteristiche dei pesi sinattici e dei fattori di distorsione.

Una volta stabilito il numero delle connessioni complessive, il dominio di ciascun peso sinattico e il livello di precisione associato (e quindi dopo aver proceduto alla codifica genetica degli individui), resta da risolvere il problema di come

rappresentare in termini matematici la struttura neurale per poter successivamente procedere all'ottimizzazione della funzione che ne risulta.

Rete feedforward: come detto nel precedente capitolo in una rete di questo tipo il flusso di informazioni si muove in maniera unidirezionale, poiché ogni strato di neuroni riceve gli input solo dallo strato precedente e li trasmette solo allo strato successivo. La rete risulta statica e senza memoria poiché la risposta della rete non dipende dalla risposta che essa aveva dato precedentemente.

La struttura della rete feedforward è rappresentata graficamente come in Fig. 7.8.

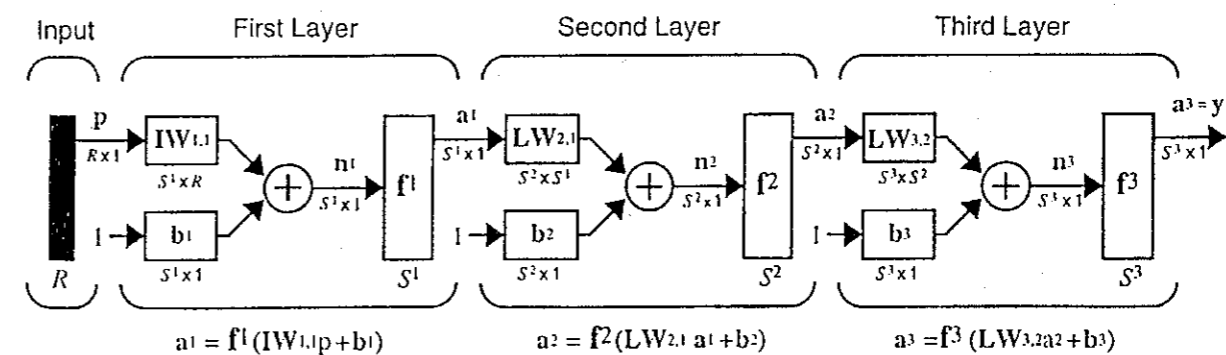


Fig. 7.8. Struttura di una rete feedforward a tre strati di neuroni.

Dove:

- R = numero degli elementi nel vettore di input;
- $p(i)$ = vettore di input alla rete;
- $S(i)$ = numero di neuroni presente nello strato i -esimo;
- f^i = funzione di trasferimento dell' i -esimo strato di neuroni;
- IW = matrice dei pesi degli input (Input Weight Matrix);
- LW = matrice dei pesi degli strati successivi al primo (Layer Weight Matrix);
- $b(i)$ = vettore delle distorsioni dello strato i -esimo;

- $n(i)$ = argomento della funzione di trasferimento dello strato i -esimo;
- $a(i)$ = vettore di output della funzione di trasferimento dello strato i -esimo.

Nel caso specifico simulato si è creata una rete a due strati di neuroni con funzioni di trasferimento 'tansig' (tangente sigmoide) e 'purelin' (lineare con coefficiente angolare pari a 1) rispettivamente per lo strato nascosto e quello di output.

Una rete di questo tipo può in generale essere utilizzata per approssimare qualunque funzione con un numero finito di discontinuità in modo piuttosto preciso se il numero di neuroni nello strato nascosto è sufficientemente alto (nel caso di seguito presentato è stato ipotizzato uguale a cinque).

Come training-set sono state utilizzate le matrici:

$$p = \begin{bmatrix} -1 & -1 & 2 & 2 \\ 0 & 5 & 0 & 5 \end{bmatrix} = \text{matrice input};$$

$$t = [-1 \quad -1 \quad 1 \quad 1] = \text{matrice target};$$

La struttura della rete così specificata è rappresentata in Fig. 7.9.

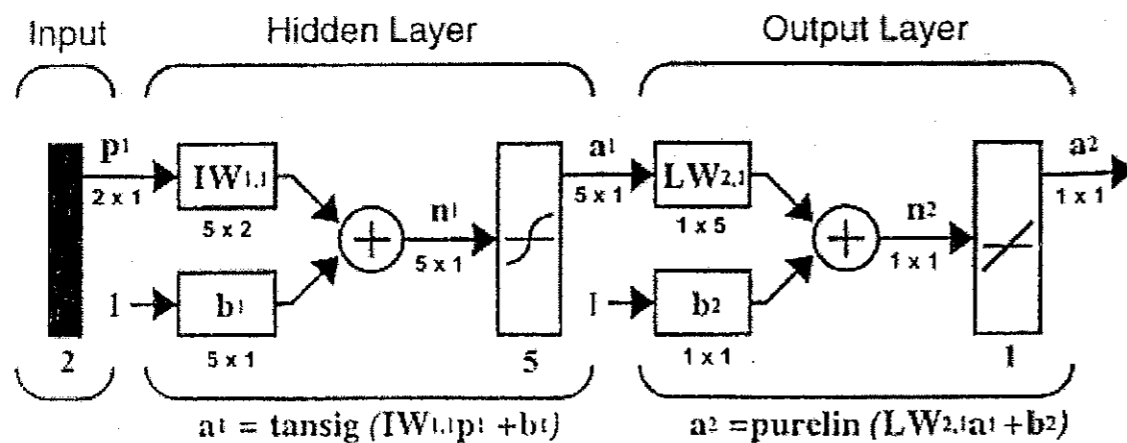


Fig. 7.9. Struttura della rete feedforward a due strati di neuroni.

Ogni individuo della popolazione reale sarà quindi formato da 21 campi associati alle variabili necessarie per definire la funzione associata alla rete utilizzata (5x2 pesi per il primo strato + 5 fattori di distorsione del primo strato + 1x5 pesi del secondo strato + 1 fattore di distorsione del secondo strato). La corrispondente popolazione binaria, prevedendo un intervallo di oscillazione di [-3 3] e un livello di precisione di 10^{-6} , sarà costituita da ben 483 celle.

La funzione associata alla rete da ottimizzare risulta la seguente:

$$(7.6) \quad a = LW_{2,1} \times \left[\frac{e^{(IW_{1,1} \times p + b_1)} - e^{-(IW_{1,1} \times p + b_1)}}{e^{(IW_{1,1} \times p + b_1)} + e^{-(IW_{1,1} \times p + b_1)}} \right] + b_2;$$

mentre la funzione di errore (funzione di fitness) utilizzata è:

$$(7.7) \quad f_{err} = \frac{\sum_{i=1}^{ninp} (a_i - t_i)^2}{ninp};$$

dove $ninp$ = numero di input passati alla rete per l'allenamento (nel caso sopra riportato $ninp = 4$).

La simulazione restituisce esattamente i valori di target [-1 -1 1 1] entro le 50 iterazioni previste.

Rete di Elman: è costituita da due strati di neuroni: il primo strato presenta una connessione del tipo *feedback* (generalmente con una funzione di trasferimento di tipo tangente sigmoide), mentre il secondo strato è unidirezionale e generalmente caratterizzato da una funzione di trasferimento lineare. La rete riceverà allora come input non solo quelli passati esternamente, ma anche l'output calcolato a livello del primo strato e riferito all'epoca precedente. In tal modo si immagazzinano informazioni dell'epoca precedente e queste vengono usate per quella presente.

La struttura, con le medesime notazioni utilizzate in Fig 7.8, è rappresentata in Fig. 7.10.

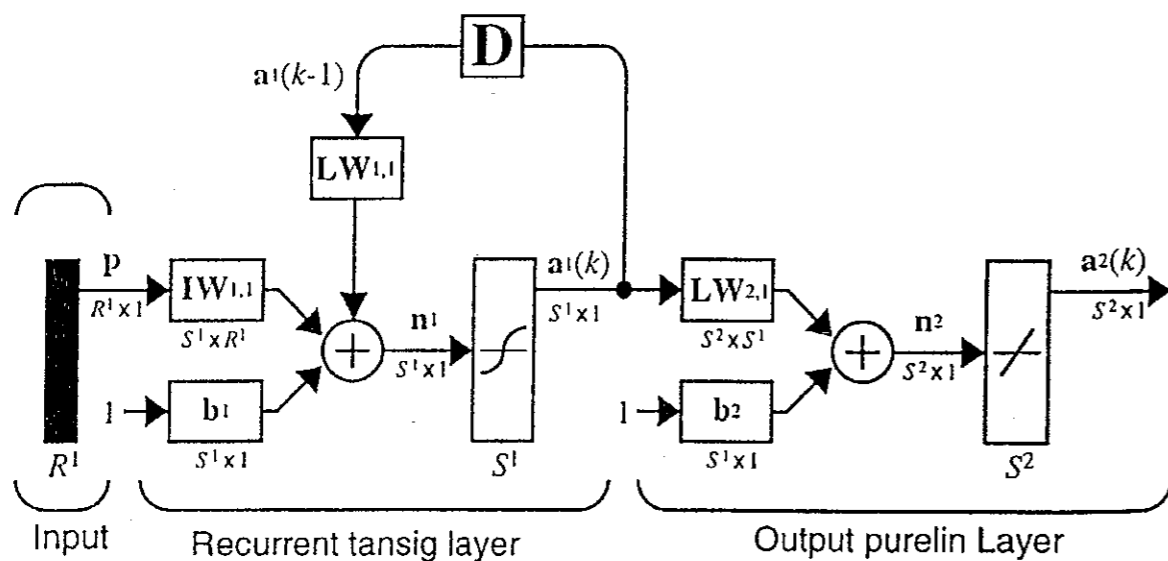


Fig. 7.10. Struttura di una rete di Elman.

Si può notare da questa figura che si tratta di una Rete Neurale ricorrente in cui il flusso di informazioni non è unidirezionale.

Questo comporta una definizione differente (e più complessa) della funzione da ottimizzare prevedendo un ciclo ricorsivo a ritroso⁷.

La funzione associata alla rete da ottimizzare risulta pertanto la seguente:

$$(7.8) \quad a = LW_{2,1} \times \left[\frac{e^{(IW_{1,1} \times p + LW_{1,1} \times a_1(rit) + b_1)} - e^{-(IW_{1,1} \times p + LW_{1,1} \times a_1(rit) + b_1)}}{e^{(IW_{1,1} \times p + LW_{1,1} \times a_1(rit) + b_1)} + e^{-(IW_{1,1} \times p + LW_{1,1} \times a_1(rit) + b_1)}} \right] + b_2;$$

dove $a_1(rit) = a_1(k-1)$ nella Fig. 7.10.

La funzione di errore che si utilizza è la stessa della rete feedforward.

Il numero di neuroni dello strato nascosto viene nella simulazione impostato pari a 10 e le matrici utilizzate per allenare la rete sono:

$$p = [1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1];$$

$$t = [0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1];$$

dove i valori del target sono posti pari a 0 quando due valori pari a 1 si presentano nel vettore p (all'epoca $i-1$ e all'epoca i).

Anche in questo caso la rete simula esattamente i valori di target entro le 50 iterazioni previste.

⁷ Anche la definizione delle stringhe rappresentative degli individui qui è molto più complessa e prevede la presenza di ben 1518 celle per rappresentare le 66 variabili necessarie (considerando il medesimo dominio e la stessa precisione utilizzati in precedenza con la rete "feed forward").

7.3. CONCLUSIONI

In questo capitolo sono state confrontate due tecniche di ottimizzazione: una classica di tipo deterministico e una "evolutive" di tipo probabilistico.

Il test di confronto, basato sulla funzione di Rosenbrock, ha mostrato una maggiore efficienza e robustezza per quanto concerne gli Algoritmi Genetici.

Ciò non significa che tale metodologia risulti sempre più efficiente a prescindere dalla funzione da ottimizzare, anzi, in Fig. 7.7 si può notare come per determinate classi di problemi la situazione sia rovesciata. Il fattore chiave però consiste nel fatto che gli Algoritmi Genetici rispondono ottimamente a qualunque tipo di funzione da ottimizzare. Proprio questa caratteristica li rende preferibili da parte di coloro che non conoscono esattamente l'andamento e le caratteristiche delle funzioni obiettivo o quando la loro complessità è notevolmente alta.

Queste argomentazioni sono risultate fondamentali per la scelta della metodologia da utilizzare nella simulazione di Reti Neurali (in cui, generalmente, la funzione associata è parecchio complessa).

I risultati mostrano che gli strumenti creati sono ottimi e le reti simulano correttamente, dopo qualche iterazione, i valori di target, rivelandosi utili mezzi di previsione per analisi più complesse (vedi capitolo sesto).

L'unico fattore negativo deve essere imputato alle risorse temporali "consumate" dall'algoritmo: i tempi richiesti per l'esecuzione di problemi un po' più complessi di quelli proposti crescono oltremisura rispetto a quelli necessari alle funzioni preimpostate di MATLAB (che utilizzano tecniche classiche di ottimizzazione). Ciò deve essere di incentivo all'approfondimento siccome viviamo in una società che, giorno dopo giorno, mette a nostra disposizione macchine sempre più potenti e veloci.

8. APPLICAZIONE DELLE TECNICHE DI MODELLIZZAZIONE GENETICA

In questo capitolo verranno presentati due esempi pratici di implementazione di alcuni degli strumenti esaminati nei capitoli precedenti riferiti all'analisi di specificazione di un modello statistico.

Verranno analizzate le problematiche da affrontare nella scelta del modello migliore per gli scopi che l'utente si propone in due casi tra loro molto differenti.

Il primo esempio si riferisce all'individuazione di un modello operativo previsionale del livello di marea misurato a Venezia durante il 1984 [MiPo00]; il secondo alla selezione di un modello che spieghi la durata in mesi del livello di disoccupazione calcolato in Emilia Romagna (sulla base dei dati raccolti dall'ISTAT e riferiti al periodo primo trimestre 1993 - primo trimestre 1995) [LaMi99].

Non vengono proposte soluzioni efficienti e ottimali per tutti i problemi che si devono affrontare, ma solo alcuni aspetti di questi e come utilizzare gli strumenti esaminati per risolverli. Si tratta pertanto di un'analisi volta a mostrare come le diverse tecniche strumentali perfezionate in questa trattazione possano interagire quando la situazione che si presenta ne permette l'utilizzo.

8.1. MODELLO PREVISIONALE DELLE MAREE DI VENEZIA

IL MODELLO LINEARE: Il singolare andamento delle maree nel mare Adriatico ha, da molti anni, attratto fisici e meteorologi impegnandoli in più direzioni per la realizzazione di schemi numerici in grado di interpretare il più fedelmente possibile il livello di marea.

Elemento di particolare attrazione si è dimostrato il fenomeno dell'acqua alta che interessa principalmente le lagune venete, soprattutto quella di Venezia con l'allagamento del centro storico e delle isole.

Questo evento provoca, da molti anni, danni rilevanti agli operatori economici veneziani, ai trasporti pubblici e privati, alla viabilità pedonale, all'attività portuale e da ultimo, ma non meno importante, causa condizioni di vita insalubri.

Gli studi per la previsione del livello di marea sono iniziati con la formulazione fisico-matematica del fenomeno mareale impostando le relazioni matematiche (equazioni dell'idrodinamica) dell'interazione aria-mare attraverso le condizioni iniziali e al contorno.

Si ottiene così la distribuzione nel tempo e nello spazio di alcune variabili quali per esempio la velocità di corrente e il livello marino.

Questo metodo, nonostante la buona affidabilità nella previsione, ha dimostrato ben presto grossi limiti, come la necessità di conoscere in tempo reale alcuni parametri meteomarini o la necessità di un potente mezzo di calcolo per ottenere velocemente i risultati.

Negli ultimi anni quindi si sono seguite strade alternative, apparentemente più superficiali, fondate non più sulla formulazione idrodinamica (metodo deterministico) ma su schemi per lo più empirici i cui risultati si sono rivelati interessanti grazie all'applicazione delle teorie statistiche (metodo empirico-statistico).

L'esistenza delle sesse (oscillazioni libere del mare) con un periodo di 22 ore per la frequenza fondamentale, suggerisce l'autoregressione, o l'uso dei livelli osservati nelle ore precedenti come predittori. In altre parole, questo significa che se

l'Adriatico fosse già in oscillazione, se non ci fossero nuovi disturbi e se non ci fosse smorzamento, il suo livello risulterebbe lo stesso di 22 ore prima.

Poi c'è il fattore meteorologico: sia la pressione atmosferica (effetto barometro) che il vento col suo trascinamento hanno effetti notevoli; poiché il vento dipende soprattutto dalla pressione, tanto vale provare solo l'uso di quest'ultima per il nostro scopo.

Anche la dipendenza non lineare tra la velocità del vento e il suo impatto sul mare può essere approssimata con la linearità, almeno come tentativo.

Si potrà giudicare la validità di queste approssimazioni solo dalla bontà dei risultati.

Siccome una parte piuttosto consistente del fenomeno mareale viene attribuita alle forze gravitazionali e poiché queste sono attualmente ben definite, l'analisi tecnica verrà basata soltanto sulla parte non astronomica del fenomeno.

Il modello è strutturato su un massimo di 182 predittori: si cerca tra tutti i possibili $2^{182}-1$ modelli quello "migliore" per ogni anticipo previsivo effettuato (1, 3, 6, 9, 12, 15 e 24 ore).

L'insieme dei predittori è così suddiviso: 132 livelli di marea osservata a Venezia, all'indietro nel tempo, da una certa ora T all'ora $T-131$ (cosiddetti parametri marini), 50 valori di pressione atmosferica osservati a Bari, Falconara, Genova, Pesaro, Pescara, Ravenna, Rimini, Teramo, Trieste, Venezia, ogni 3 ore dal tempo T al tempo $T-12$ (cosiddetti parametri metereologici).

Lo schema si presenta così:

$$(8.1) \quad h_{T+\tau} = \sum_{i=0}^I (a_i^{(\tau)} \times h_{T-i}) + \sum_{j=1}^J \sum_{k=0}^K b_{j,k}^{(\tau)} \times P_{(j,T-3k)}$$

dove:

- T = tempo in cui viene effettuata la previsione;
- τ = anticipo previsivo in ore che si vuole ottenere;

- $h_{T+\tau}$ = livello di marea previsto per il tempo T più il numero di ore di anticipo τ ;
- $a_i^{(\tau)}$ = coefficienti relativi alla marea a Venezia dello schema regressivo con $i=0, \dots, 131$ (massimo 132 coefficienti per ogni anticipo di previsione τ);
- h_{T-i} = livelli di marea osservati a Venezia al tempo $T-i$ (dal tempo T al tempo $T-131$ a seconda dei coefficienti selezionati);
- $b_{j,k}^{(\tau)}$ = coefficienti dello schema regressivo per le pressioni atmosferiche registrate nelle stazioni j ($j=1, 2, \dots, 10$; nell'ordine Bari, Falconara, Genova, Pesaro, Pescara, Ravenna, Rimini, Teramo, Trieste e Venezia) relativi alle 5 osservazioni priorarie k ($k=0, 1, 2, 3, 4$) di pressione atmosferica (sono state utilizzate massimo 5 osservazioni priorarie di pressione dal tempo T al tempo $T-12$ per ognuna delle 10 stazioni meteorologiche);
- $p_{j,T-3k}$ = pressioni atmosferiche registrate a Bari, Falconara, Genova, Pesaro, Pescara, Ravenna, Rimini, Teramo, Trieste e Venezia (indice j) nelle ore $T, T-3, \dots, T-12$.

I coefficienti (a_i e $b_{j,k}$) sono quindi, complessivamente, 182.

Per ogni anticipo τ la specificazione del modello "migliore" avviene mediante l'utilizzo di Algoritmi Genetici (utilizzando il programma di selezione del modello lineare "GALMS" con l'introduzione della componente autoregressa). Il programma seleziona tra i 182 predittori quelli che specificheranno il modello "migliore" in termini di AIC previsivo.

I coefficienti vengono calcolati sulla base dei dati di marea e pressione osservati nell'anno 1984 tra l'1/1/84 e il 19/10/84 (training set) secondo i metodi statistici classici (stima OLS).

In fase previsionale lo schema proposto è stato utilizzato per l'anno 1984 tra il 19/10/84 e il 31/12/84 (test set).

A livello sperimentale è stato protratto il valore di τ fino a 24 ore (assumendo, come detto sopra, i valori 1, 3, 6, 9, 12, 15, 24).

La codifica dell'algoritmo prevede pertanto l'utilizzo di una stringa costituita da 182 celle, ognuna delle quali può assumere valore 0 o 1 (0 indicherà l'esclusione di quel predittore dal modello, 1 indicherà invece l'inclusione).

Prima di esaminare i risultati ottenuti analizziamo anche gli altri modelli usati.

IL MODELLO NEURALE: Sia il cosiddetto "effetto barometro" causato dalla pressione atmosferica che il vento, con il suo trascinarsi, hanno effetti notevoli sul livello di marea. Poiché il vento dipende dal campo di pressione e la velocità del vento può essere rappresentata linearmente solo in modo approssimato, utilizzando questi dati viene spontaneo provare anche l'analisi di modelli non lineari.

Mediante l'utilizzo di Reti Neurali si cercherà, pertanto, una possibile componente non lineare tra i dati autoregressi di marea e di pressione con il valore previsto di marea. Le Reti Neurali con addestramento supervisionato sono infatti considerate buoni approssimatori di funzioni non lineari.

Gli strumenti tecnici per poter implementare questo tipo di analisi sono stati discussi nel capitolo quarto. Il programma GANND, infatti, è uno strumento per la selezione dei parametri strutturali della rete e dei predittori per la costruzione del modello "migliore" in termini previsivi. Purtroppo, come detto a suo tempo, tale programma non è del tutto efficiente a causa delle risorse temporali di cui necessita.

Inoltre lavorando con un così alto numero di parametri anche le risorse di memoria richieste diventano insufficienti.

Per risolvere questo inconveniente si è pensato di analizzare i risultati ottenibili "passando" alla rete (la cui struttura è stata selezionata manualmente dall'operatore) i parametri selezionati precedentemente col modello lineare¹ (il cui numero, variabile a seconda dell'anticipo previsionale, è compreso tra 79 e 98).

¹ Questo tipo di operazione non è molto corretto dal punto di vista teorico: si cerca, infatti, di analizzare i risultati ottenibili con una struttura non lineare del modello utilizzando parametri selezionati con un modello lineare.

Anche in questo caso la rete fatica a lavorare causando molto spesso problemi di "out of memory" (mancanza di risorse in termini di memoria necessarie al calcolo dell'algoritmo).

Si è cercato quindi di risolvere il problema utilizzando il programma di selezione dei parametri di un modello lineare mediante la tecnica "Backward" associata al "p-value". Questo ha permesso di ottenere modelli di dimensioni molto più ridotte (da un minimo di 23 a un massimo di 59 predittori).

Si è confrontato il valore di R^2 e dello scarto quadratico medio prima e dopo la riduzione dei parametri per verificare che il modello "ridotto" non fosse notevolmente peggiorato rispetto a quello "esteso". Una volta "accettata" la riduzione, i parametri del modello così determinato sono stati passati alla Rete Neurale.

Il procedimento seguito è rappresentato in Fig. 8.1.

L'intero set di dati relativi al 1984 è stato qui campionato diversamente rispetto al modello lineare al fine di poter applicare la metodologia di "Early Stopping".

Si è suddiviso il campione in tre parti:

- Training set (circa 5000 dati corrispondenti al periodo 6/1/84 - 1/8/84);
- Validation set (circa 1700 dati corrispondenti al periodo 2/8/84 - 19/10/84);
- Test set (la restante parte corrispondente al periodo 19/10/84 - 31/12/84).

Adottando la metodologia di "Early stopping" il Training set viene utilizzato esclusivamente durante l'allenamento per il calcolo del gradiente e l'aggiornamento della matrice dei pesi sinattici relativi alla rete implementata; il validation set viene utilizzato dall'algoritmo per il calcolo della funzione di errore associata alla struttura neurale implementata; il test set non viene utilizzato in fase di allenamento ma serve soltanto per il confronto di modelli diversi.

Questa tecnica permette di evitare problemi di "overfitting" molto frequenti quando si lavora con un numero alto di parametri. In pratica, il problema di "overfitting" si riscontra quando lo scarto quadratico medio nel campione utilizzato per l'addestramento della struttura neurale è molto basso ma, presentando alla rete nuovi campioni di dati, l'errore associato è spropositatamente alto. La rete ha

memorizzato i valori utilizzati per l'addestramento della stessa ma non ha imparato a generalizzare alcuna nuova situazione.

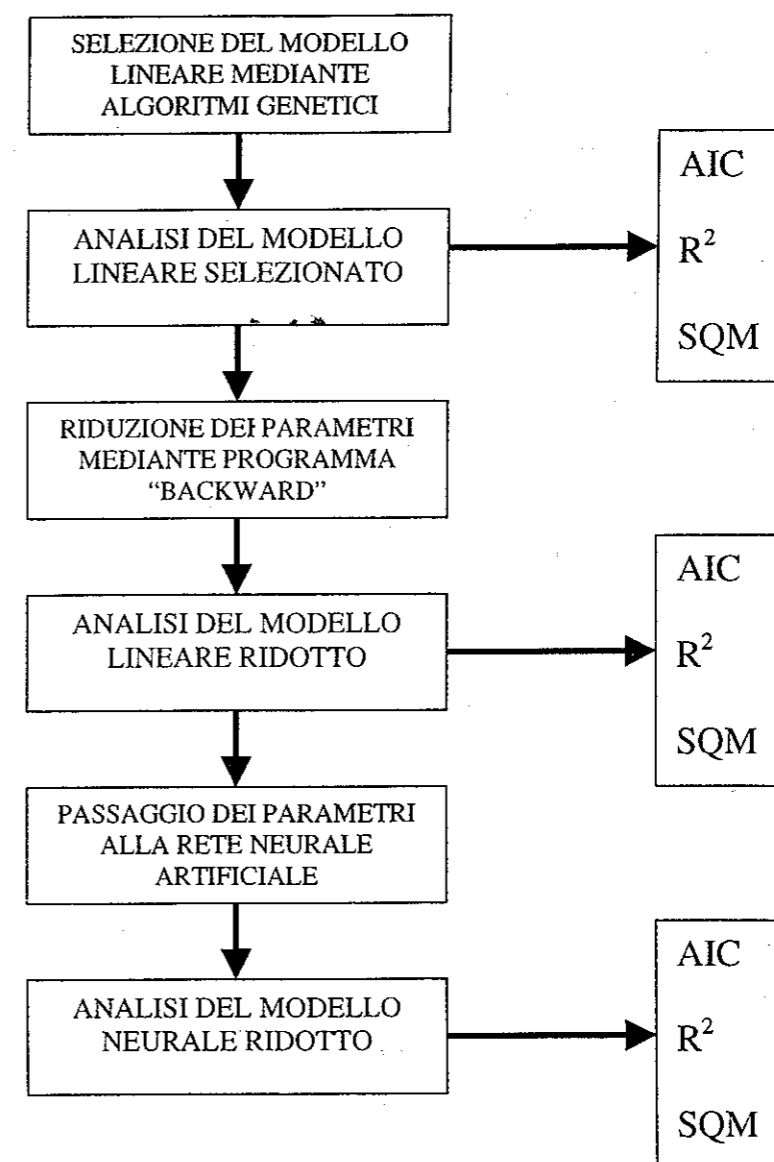


Fig. 8.1. Procedimento di analisi delle maree mediante l'uso di modelli lineari e neurali.

Si è provveduto quindi a confrontare, per tutti i casi di alta marea (in cui il livello di marea astronomica sommato alla componente meteorologica risulta superiore ai 100 centimetri di altezza) verificatisi dal 23 settembre 1984 al 31 dicembre 1984, l'indicazione del modello previsivo con l'effettivo azionamento o meno delle sirene per valori di marea superiori a 110 centimetri.

Vengono però prima presentati i risultati previsivi generali ottenuti con il modello lineare completo (Figg. 8.2 - 8.8) e col modello neurale ridotto (Figg. 8.9 - 8.15) nonché i valori dell' R^2 per i modelli esaminati con le relative dimensioni (tabella 8.1).

Anticipo Previsivo (in ore)	Dimensione	R^2	Dimensione	R^2	R^2
	modello lineare selezionato	modello lineare selezionato	modello lineare ridotto	Modello lineare ridotto	modello neurale ridotto
1 h.	98	0.988	59	0.988	0.985
3 h.	96	0.910	52	0.890	0.851
6 h.	91	0.872	44	0.872	0.884
9 h.	83	0.843	38	0.841	0.836
12 h.	92	0.812	37	0.810	0.825
15 h.	79	0.773	32	0.762	0.761
24 h.	79	0.709	23	0.709	0.724

Tabella 8.1. Dimensione e R^2 (calcolato sul Test set) dei modelli utilizzati nella previsione.

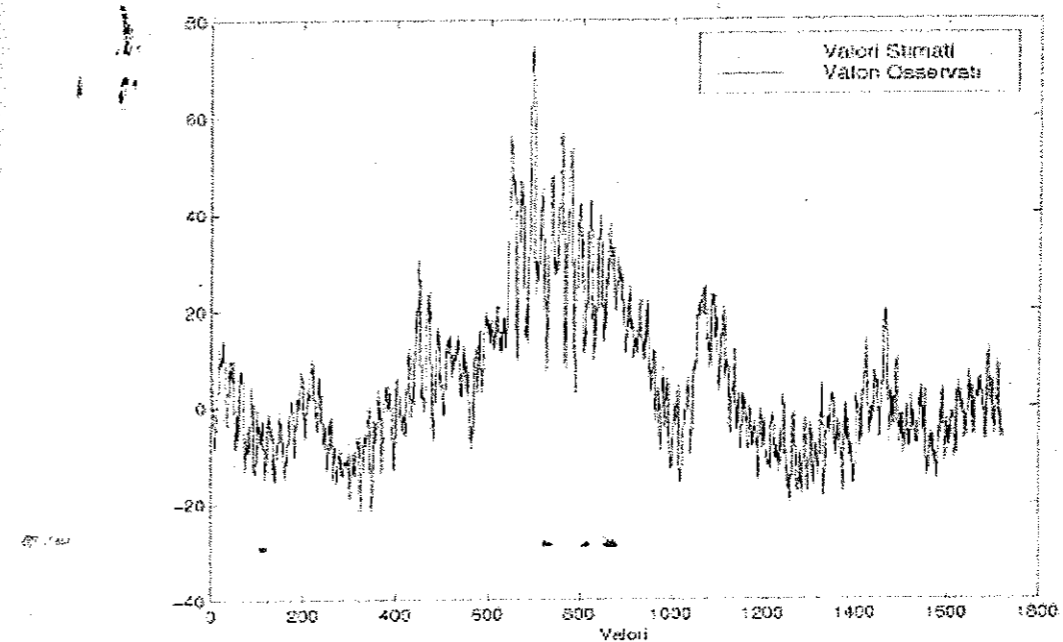


Fig. 8.2. Valori previsti vs valori osservati con un'anticipazione previsiva di un'ora ottenuta con modello lineare (previsioni effettuate dal 19/10/84 ore 04.00 al 31/12/84 ore 24.00).

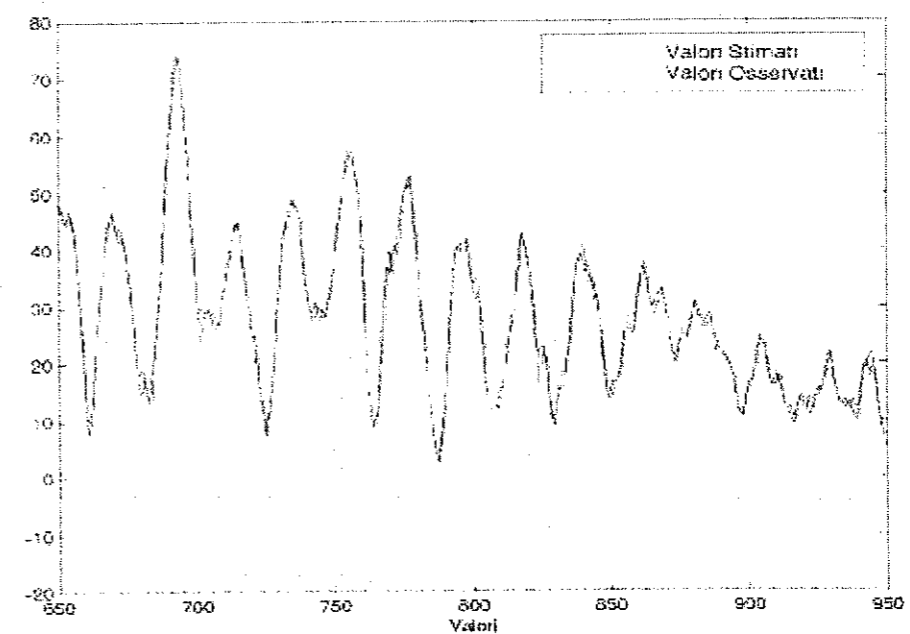


Fig. 8.3. Valori previsti vs valori osservati con un'anticipazione previsiva di un'ora (previsioni effettuate dal 15/11/84 ore 5.00 al 27/11/84 ore 17.00) ottenuta con modello lineare.

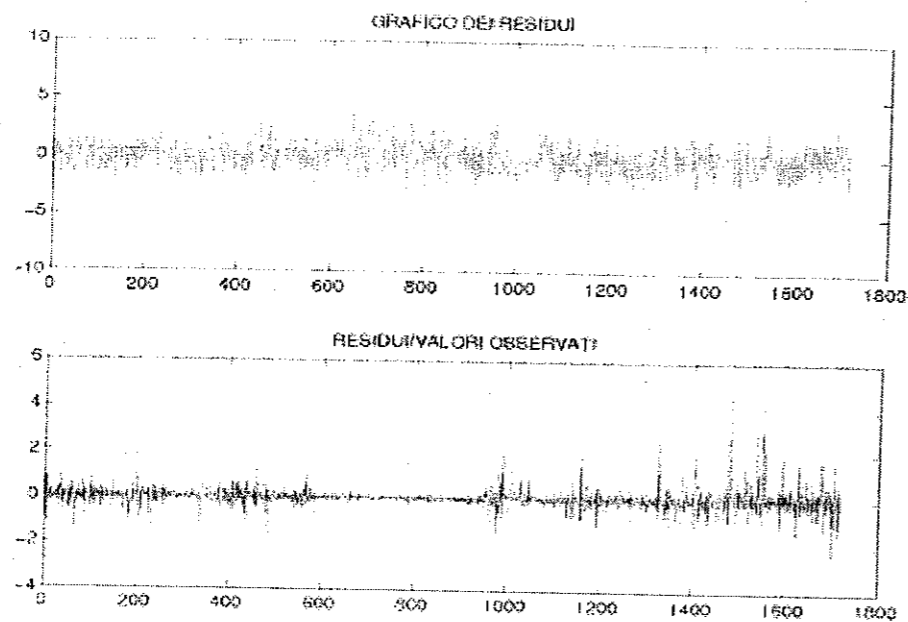


Fig. 8.4. Residui ottenuti sulla previsione a un'ora (previsioni effettuate dal 19/10/84 ore 4.00 al 31/12/84 ore 24.00) ottenuta con modello lineare.

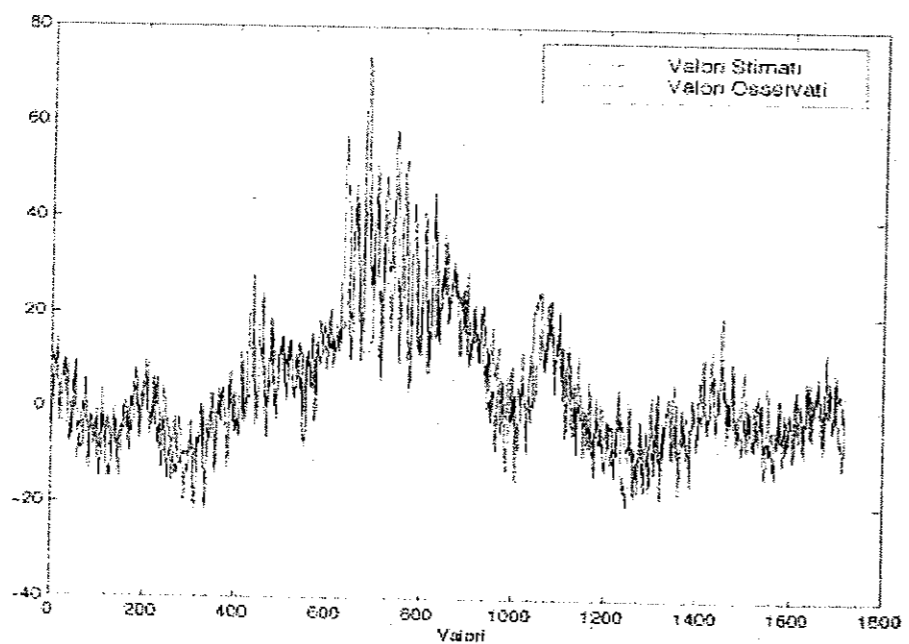


Fig. 8.5. Valori previsti vs valori osservati con un'anticipazione previsiva di 12 ore ottenuta con modello lineare.

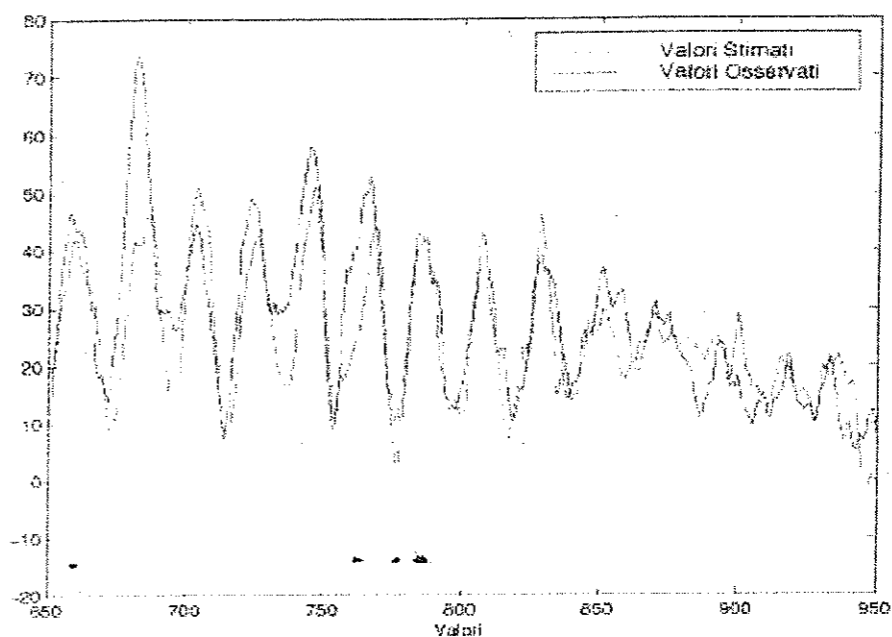


Fig. 8.6. Valori previsti vs valori osservati con un'anticipazione previsiva di 12 ore (previsioni effettuate dal 14/11/84 ore 18.00 al 27/11/84 ore 6.00) ottenuta con modello lineare.

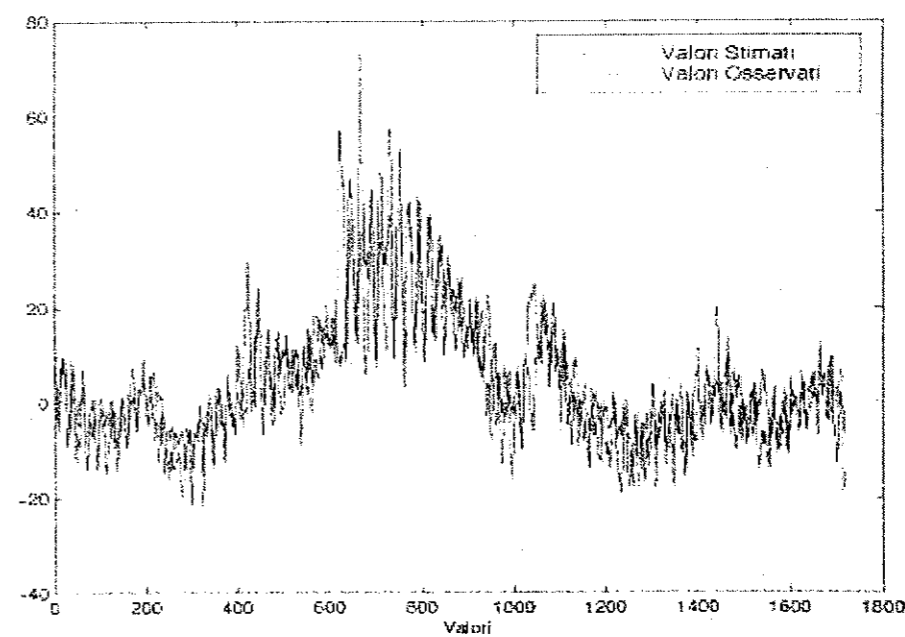


Fig. 8.7. Valori previsti vs valori osservati con un'anticipazione previsiva di 24 ore ottenuta con modello lineare.

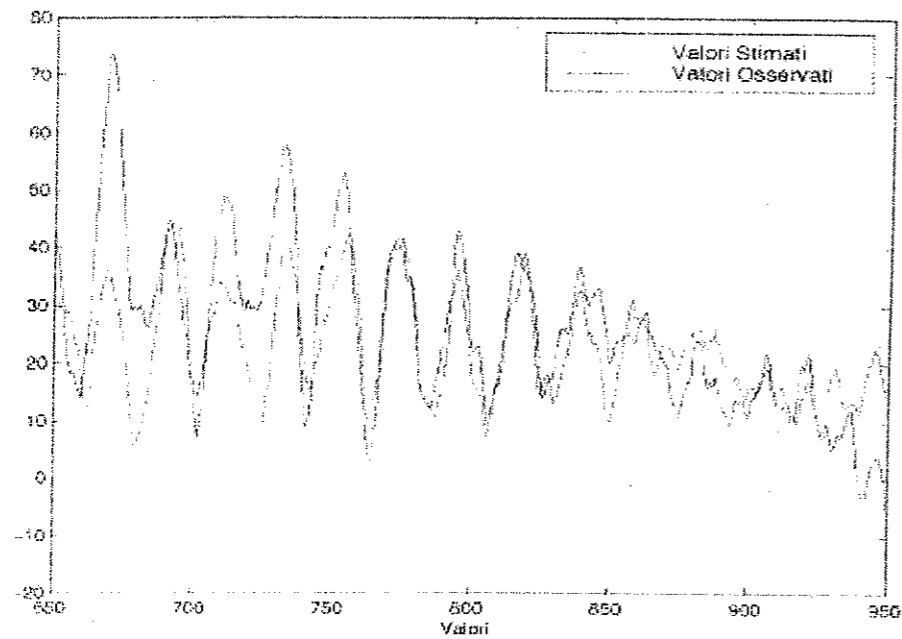


Fig. 8.8. Valori previsti vs valori osservati con un'anticipazione previsiva di 24 ore (previsioni effettuate dal 14/11/84 ore 6.00 al 26/11/84 ore 18.00) ottenuta con modello lineare.

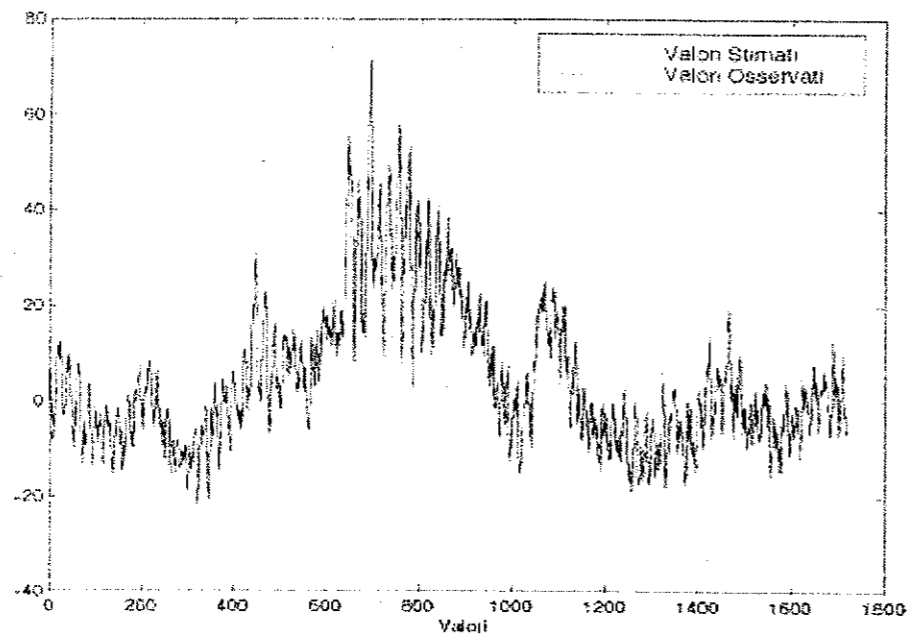


Fig. 8.9. Valori previsti vs valori osservati con un'anticipazione previsiva di un'ora ottenuta con modello neurale (sul test set).

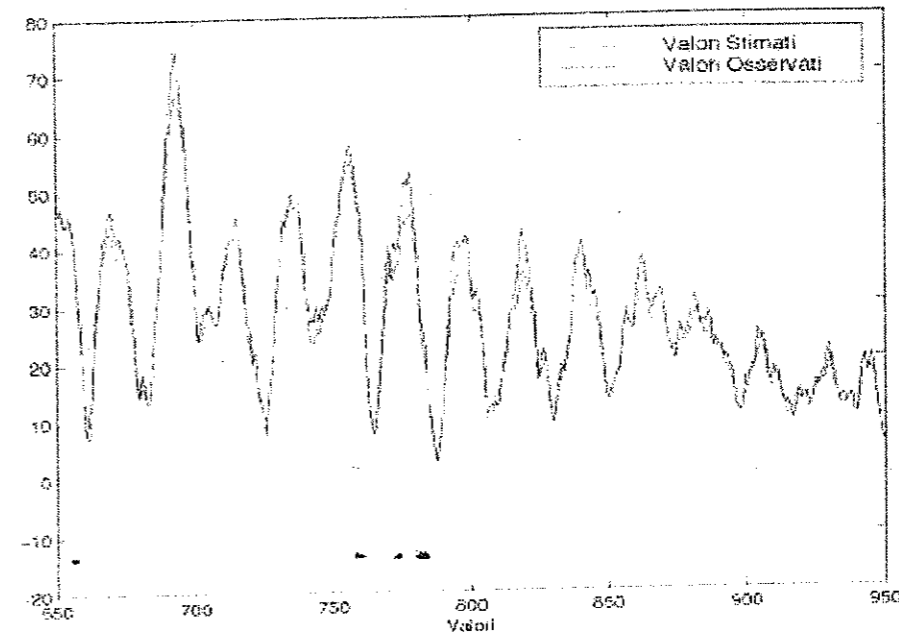


Fig. 8.10. Valori previsti vs valori osservati con un'anticipazione previsiva di un'ora ottenuta con modello neurale (previsioni effettuate dal 15/11/84 ore 5.00 al 27/11/84 ore 17.00).

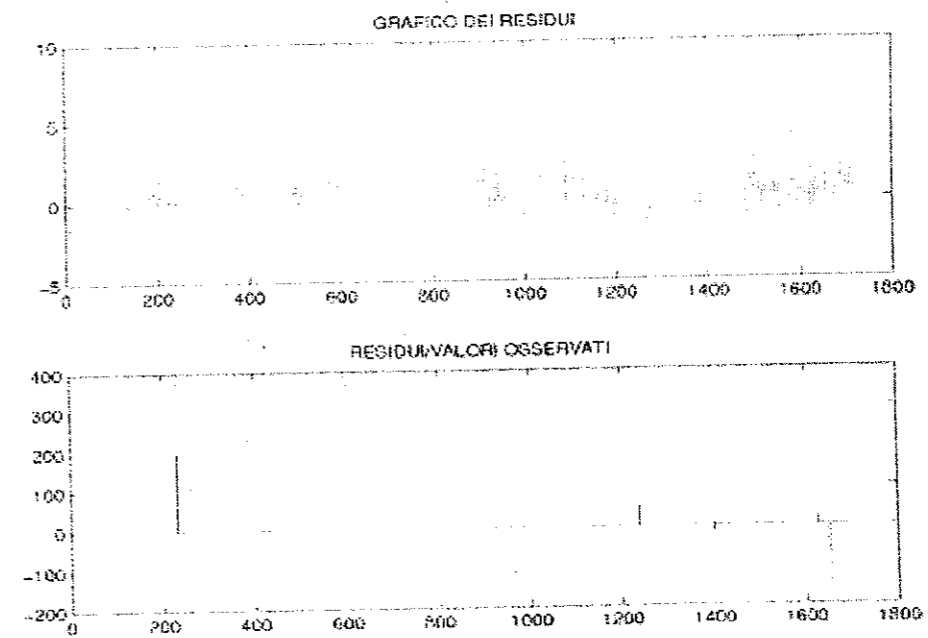


Fig. 8.11. Residui ottenuti sulla previsione a un'ora (dal 19/10/84 ore 4.00 al 31/12/84 ore 24.00) ottenuta con modello neurale.

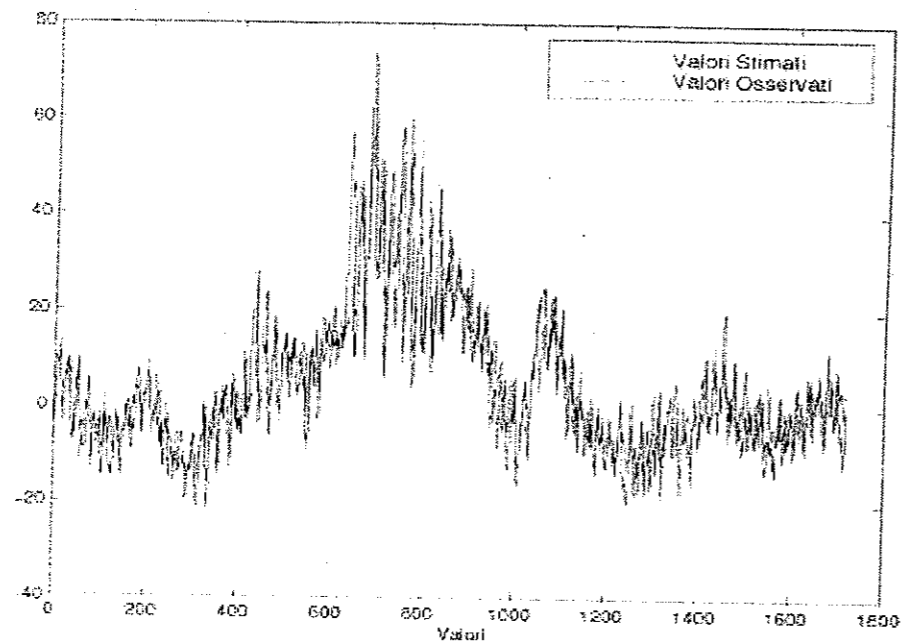


Fig. 8.12. Valori previsti vs valori osservati con un'anticipazione previsiva di 12 ore ottenuta con modello neurale.

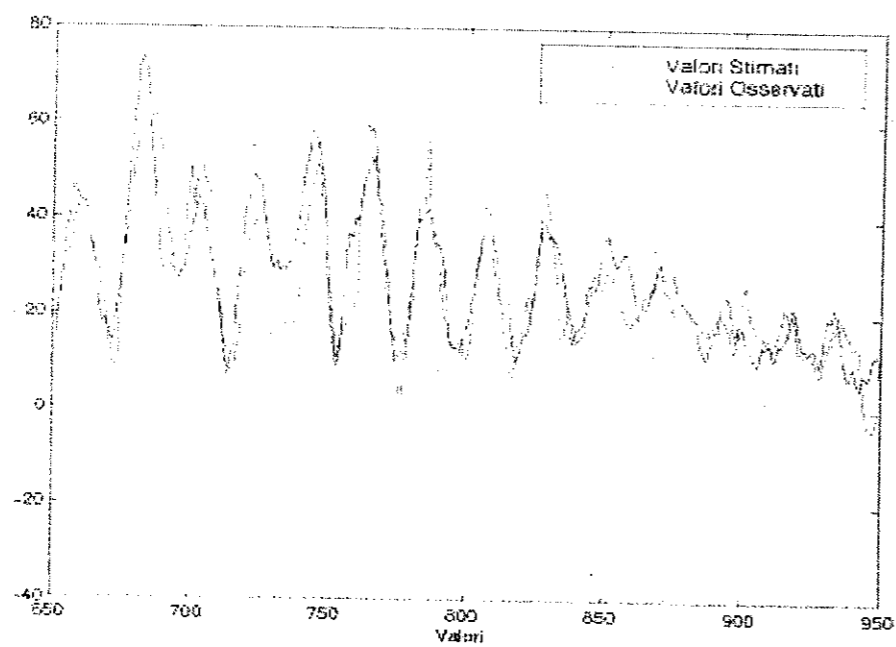


Fig. 8.13. Valori previsti vs valori osservati con un'anticipazione previsiva di 12 ore (previsioni effettuate dal 14/11/84 ore 18.00 al 27/11/84 ore 6.00) ottenuta con modello neurale.

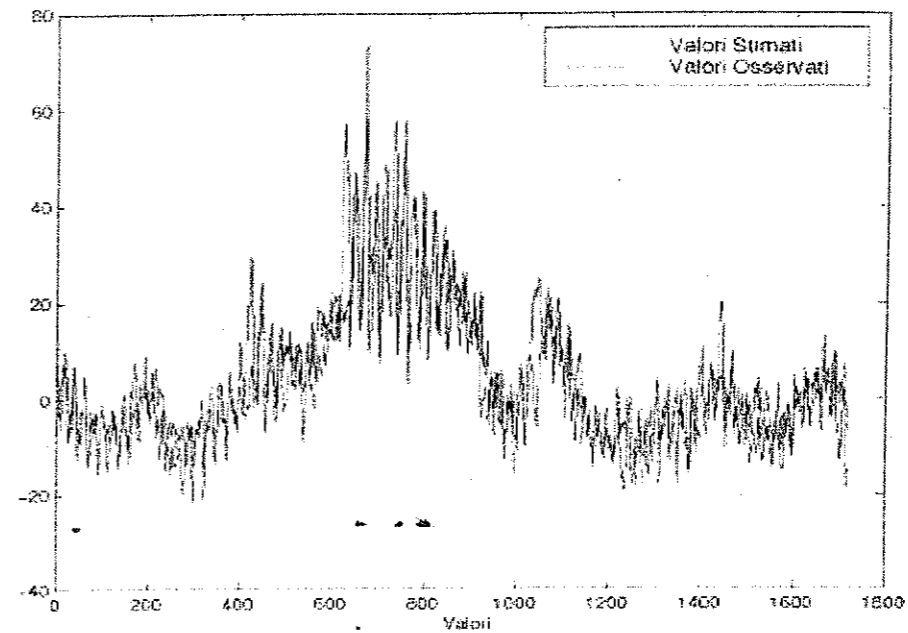


Fig. 8.14. Valori previsti vs valori osservati con un'anticipazione previsiva di 24 ore ottenuta con modello neurale.

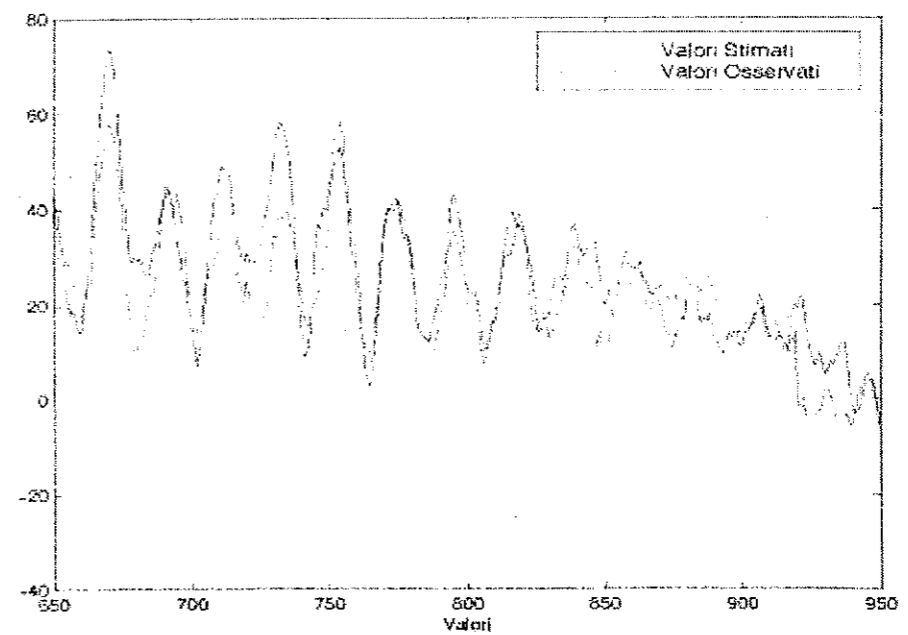


Fig. 8.15. Valori previsti vs valori osservati con un'anticipazione previsiva di 24 ore (previsioni effettuate dal 14/11/84 ore 6.00 al 26/11/84 ore 18.00) ottenuta con modello neurale.

Sono stati successivamente presi in esame, per tutti i casi di alta marea (almeno 100 cm.) verificatisi dal 23 settembre 1984 al 31 dicembre 1984, il livello previsto dai modelli sopra esposti e il relativo valore di raffronto osservato.

Nelle tabelle che seguono (8.2 - 8.8), nella prima colonna sono elencati i valori degli anticipi previsti rispetto al giorno e all'ora riportati sopra ad ognuna di esse; nella seconda e quarta colonna vengono mostrati i livelli di marea previsti rispettivamente col modello lineare esteso e col modello neurale ridotto per ogni anticipo previsivo; nella terza e quinta colonna vengono valutati, per ogni anticipazione oraria dei due modelli previsivi utilizzati, se le sirene di allarme, che a Venezia devono suonare ogni volta che si prevede un livello di marea superiore a 110 cm., sarebbero state azionate correttamente.

Giorno: 23/09/84 ore 20.00. Valore di raffronto osservato: 103 cm

<i>Anticipo Previsivo</i>	MODELLO LINEARE	CONFRONTO	MODELLO NEURALE	CONFRONTO
1 h.	102.5 cm	corretto	101.1 cm	corretto
3 h.	102.9 cm	corretto	102.5 cm	corretto
6 h.	97.6 cm	corretto	105.1 cm	corretto
9 h.	95.9 cm	corretto	98.2 cm	corretto
12 h.	86.1 cm	corretto	95.3 cm	corretto
15 h.	87.5 cm	corretto	90.0 cm	corretto
24 h.	75.6 cm	corretto	84.1 cm	corretto

Tabella 8.2. Previsione del livello di marea a diverse anticipazioni orarie utilizzando il modello lineare e quello neurale: i risultati.

Giorno: 23/09/84 ore 22.00. Valore di raffronto osservato: 105 cm

<i>Anticipo Previsivo</i>	MODELLO LINEARE	CONFRONTO	MODELLO NEURALE	CONFRONTO
1 h.	107.0 cm	corretto	105.9 cm	corretto
3 h.	104.4 cm	corretto	104.3 cm	corretto
6 h.	96.7 cm	corretto	97.9 cm	corretto
9 h.	96.2 cm	corretto	95.5 cm	corretto
12 h.	90.3 cm	corretto	93.2 cm	corretto
15 h.	86.7 cm	corretto	90.1 cm	corretto
24 h.	77.9 cm	corretto	86.3 cm	corretto

Tabella 8.3. Previsione del livello di marea a diverse anticipazioni orarie utilizzando il modello lineare e quello neurale: i risultati.

Giorno: 20/11/84 ore 9.00. Valore di raffronto osservato: 115 cm

<i>Anticipo Previsivo</i>	MODELLO LINEARE	CONFRONTO	MODELLO NEURALE	CONFRONTO
1 h.	112.2 cm	corretto	110.8 cm	corretto
3 h.	106.5 cm	<u>Non corretto</u>	112.0 cm	corretto
6 h.	101.9 cm	<u>Non corretto</u>	106.2 cm	<u>Non corretto</u>
9 h.	98.9 cm	<u>Non corretto</u>	99.7 cm	<u>Non corretto</u>
12 h.	97.9 cm	<u>Non corretto</u>	131.0 cm	corretto
15 h.	102.3 cm	<u>Non corretto</u>	109.0 cm	<u>Non corretto</u>
24 h.	105.4 cm	<u>Non corretto</u>	113.1 cm	corretto

Tabella 8.4. Previsione del livello di marea a diverse anticipazioni orarie utilizzando il modello lineare e quello neurale: i risultati.

Giorno: 21/11/84 ore 9.00. Valore di raffronto osservato: 119 cm

<i>Anticipo Previsivo</i>	MODELLO LINEARE	CONFRONTO	MODELLO NEURALE	CONFRONTO
1 h.	117.3 cm	corretto	114.8 cm	corretto
3 h.	119.9 cm	corretto	110.1 cm	corretto
6 h.	113.9 cm	corretto	115.1 cm	corretto
9 h.	119.1 cm	corretto	114.2 cm	corretto
12 h.	119.6 cm	corretto	130.0 cm	corretto
15 h.	122.5 cm	corretto	111.2 cm	corretto
24 h.	116.5 cm	corretto	117.4 cm	corretto

Tabella 8.5. Previsione del livello di marea a diverse anticipazioni orarie utilizzando il modello lineare e quello neurale: i risultati.

Giorno: 22/11/84 ore 9.00. Valore di raffronto osservato: 111 cm

<i>Anticipo Previsivo</i>	MODELLO LINEARE	CONFRONTO	MODELLO NEURALE	CONFRONTO
1 h.	111.1 cm	corretto	111.0 cm	corretto
3 h.	113.0 cm	corretto	110.4 cm	corretto
6 h.	103.9 cm	<u>Non corretto</u>	105.3 cm	<u>Non corretto</u>
9 h.	108.1 cm	<u>Non corretto</u>	106.5 cm	<u>Non corretto</u>
12 h.	106.6 cm	<u>Non corretto</u>	117.0 cm	corretto
15 h.	107.9 cm	<u>Non corretto</u>	104.4 cm	<u>Non corretto</u>
24 h.	104.8 cm	<u>Non corretto</u>	105.6 cm	<u>Non corretto</u>

Tabella 8.6. Previsione del livello di marea a diverse anticipazioni orarie utilizzando il modello lineare e quello neurale: i risultati.

Giorno: 23/11/84 ore 10.00. Valore di raffronto osservato: 103 cm

<i>Anticipo Previsivo</i>	MODELLO LINEARE	CONFRONTO	MODELLO NEURALE	CONFRONTO
1 h.	102.8 cm	corretto	102.4 cm	corretto
3 h.	101.9 cm	corretto	98.0 cm	corretto
6 h.	99.4 cm	corretto	95.4 cm	corretto
9 h.	101.8 cm	corretto	90.1 cm	corretto
12 h.	90.2 cm	corretto	108.0 cm	corretto
15 h.	94.9 cm	corretto	94.7 cm	corretto
24 h.	98.2 cm	corretto	94.8 cm	corretto

Tabella 8.7. Previsione del livello di marea a diverse anticipazioni orarie utilizzando il modello lineare e quello neurale: i risultati.

Giorno: 24/11/84 ore 10.00. Valore di raffronto osservato: 100 cm

<i>Anticipo Previsivo</i>	MODELLO LINEARE	CONFRONTO	MODELLO NEURALE	CONFRONTO
1 h.	100.0 cm	corretto	100.7 cm	corretto
3 h.	97.8 cm	corretto	91.0 cm	corretto
6 h.	92.9 cm	corretto	86.9 cm	corretto
9 h.	91.7 cm	corretto	83.6 cm	corretto
12 h.	92.0 cm	corretto	80.0 cm	corretto
15 h.	87.1 cm	corretto	84.9 cm	corretto
24 h.	89.0 cm	corretto	81.6 cm	corretto

Tabella 8.8. Previsione del livello di marea a diverse anticipazioni orarie utilizzando il modello lineare e quello neurale: i risultati.

I risultati mostrano una buona capacità previsiva del modello e spesso si presentano migliori dei sistemi di riferimento concretamente utilizzati a Venezia. Non sempre però i valori previsti si avvicinano ai valori osservati, soprattutto quando l'anticipo supera le 3 ore. Molto indicative sono, infatti, le tabelle e tutte le figure sugli andamenti previsivi che mostrano come i due modelli usati rilevano i picchi di minimo e di massimo senza ritardi temporali (così come avveniva, invece, con le previsioni di tipo finanziario). In particolare le tabelle 8.2 e 8.3 evidenziano questa caratteristica mostrando un livello di marea inferiore ai 110 cm con ben 24 ore di anticipo (a Venezia, invece, sono suonate le sirene di allarme poiché il modello previsivo utilizzato ha fallito).

Un'ultima considerazione deve essere fatta per quanto riguarda le modalità attraverso cui si sono ottenuti i risultati. I parametri strutturali della rete impiegata sono stati modificati di volta in volta mediante tentativi allo scopo di ottenere il miglior risultato previsivo possibile. Tale procedimento, lungo e complesso, permette, infatti, di verificare durante ogni singolo tentativo se il punto verso cui tende a convergere la rete è un minimo locale o globale. La tendenza delle Reti Neurali (che, ricordiamolo, sono assistite da procedure di ottimizzazione di tipo classico) a convergere verso minimi locali incentiva ad una maggiore e più efficiente utilizzazione degli strumenti implementati nel capitolo settimo. Richiesta, questa, difficilmente esaudibile nel contesto operativo attuale, ma certamente non impossibile in un prossimo futuro.

Un'ulteriore approfondimento è attualmente indirizzato verso una scelta di tipo genetico dei predittori da utilizzare e dei parametri strutturali della rete.

8.2. ANALISI SUL LIVELLO DI DISOCCUPAZIONE IN EMILIA ROMAGNA

In questo paragrafo vengono prese in esame le strategie implementate per la determinazione dei parametri di un modello statistico impiegato per spiegare correttamente il livello di disoccupazione mensile rilevato in Emilia Romagna.

I valori osservati e i parametri utilizzabili si riferiscono a un'indagine condotta da M. Lalla e F. Pattarin [LaPa99] su dati forniti dall'Istituto Centrale di Statistica (ISTAT) per la regione Emilia Romagna, dal primo trimestre 1993 al primo trimestre 1995. L'indagine è stata effettuata su un campione di 655 soggetti in cerca di occupazione, con un'età maggiore o uguale a 15 anni.

Il campione è stato suddiviso in tre sottogruppi: persone che hanno perso l'occupazione precedente per licenziamento, dimissioni o scadenza del contratto di lavoro a termine (*disoccupati*); persone che entrano per la prima volta nel mercato del lavoro e non hanno mai cercato in precedenza un'attività lavorativa (*nuovi ingressi*); persone che hanno smesso un'attività lavorativa e per un certo periodo di tempo non hanno più cercato un impiego uscendo così dalle forze di lavoro e ora hanno ricominciato a cercare un lavoro (*rientri*).

Per ogni componente della famiglia degli intervistati in età lavorativa sono stati rilevati la condizione professionale, notizie sull'attività stessa, durata della disoccupazione in corso e caratteristiche individuali che potrebbero influenzarla. Queste informazioni riguardano dati di carattere anagrafico, grado di istruzione dell'intervistato, componenti della famiglia, attività e periodi di disoccupazione precedenti, disponibilità, aspettative e abilità professionali, tipo di occupazione trovata dopo un periodo di disoccupazione e azioni compiute durante il periodo di disoccupazione.

Ponendo y uguale alla durata del periodo di disoccupazione espresso in mesi, il modello lineare può così rappresentarsi:

$$(8.2) \quad y = \sum_{i=1}^n a_i x_i$$

dove n corrisponde al numero dei parametri relativi a ogni domanda fatta agli intervistati, in questo caso il massimo è 53; a_i rappresenta il coefficiente stimato per ogni variabile esplicativa utilizzata (x_i).

Pertanto x_i costituisce il vettore riga dei regressori utilizzati relativi alle caratteristiche personali, famigliari e del mercato del lavoro attuale sopra accennate.

La scelta dei regressori è stata effettuata utilizzando i programmi "Backward", "Forward" e quello di selezione genetica ("GALMS") analizzati nel quarto capitolo.

L'utilizzo di un modello lineare per rappresentare la durata della disoccupazione richiede l'inserimento dell'intero set di dati nel training set. Il motivo per cui è stata presa questa decisione è da ricercare nell'uso di alcune variabili "dummy" tra quelle esplicative. L'uso di questo tipo di variabili, infatti, potrebbe causare l'inserimento dell'intero campione rappresentativo della variabile stessa nel test set ponendo non pochi problemi nel momento in cui si devono stimare i parametri del modello. Inoltre, l'alta variabilità della y (con valori compresi tra 0.1 e 100.5) non perdona un eventuale sbilanciamento del numero di dati rappresentativi per quelle "dummy" tra il training e il test set².

I risultati ottenuti sono mostrati in tabella 8.9. Nella prima colonna, "metodo" indica il tipo di programma utilizzato per ottenere i risultati, nella seconda colonna "criterio" indica quale indice di riferimento è stato utilizzato, nella terza colonna è evidenziato il numero di variabili scelte dal programma per quel tipo di criterio

² I problemi aumenterebbero maggiormente se si prevedesse il campionamento dei dati in tre gruppi: training, validation e test set.

(dimensione del modello ottimo), nella quarta colonna viene riportato il valore della stima dello scarto quadratico medio e nell'ultima colonna è stato inserito l' R^2 .

METODO	CRITERIO	DIMENSIONE	SQM	R ²
Backward	AIC	17	12.6916	0.1321
Backward	BIC	14	12.8349	0.1122
Backward	HAN	14	12.8349	0.1122
Backward	SIC	14	12.8349	0.1122
Backward	p-value	11	12.9091	0.1021
Forward	AIC	18	12.6890	0.1323
Forward	HAN	4	13.3109	0.0731
Forward	SIC	6	13.1292	0.0931
Forward	P_corr	8	13.9866	0.0118
GALMS	AIC	27	12.6532	0.1372
GALMS	BIC	20	13.6334	0.0771
GALMS	SIC	16	13.0012	0.0890

Tabella 8.9. Risultati ottenuti col modello lineare.

Il modello migliore sia in termini di scarto quadratico medio che di R^2 è quello ottenuto con il programma GALMS utilizzando come criterio selettivo l'AIC.

In Fig. 8.17 è riportato il relativo error-plot da cui si evidenzia una forte non linearità con evidenti effetti di eteroschedasticità.

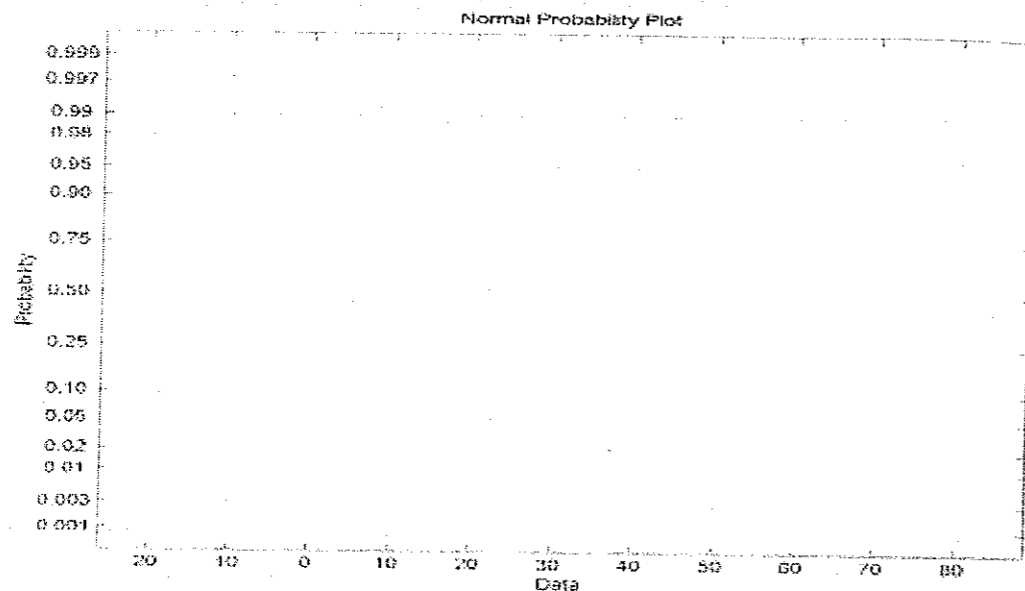


Fig 8.17. "Normal Probability Plot" dei residui ottenuti col programma GALMS e con criterio di selezione l'AIC.

In Fig. 8.18 viene mostrata la funzione di ripartizione campionaria e quella teorica dei residui.

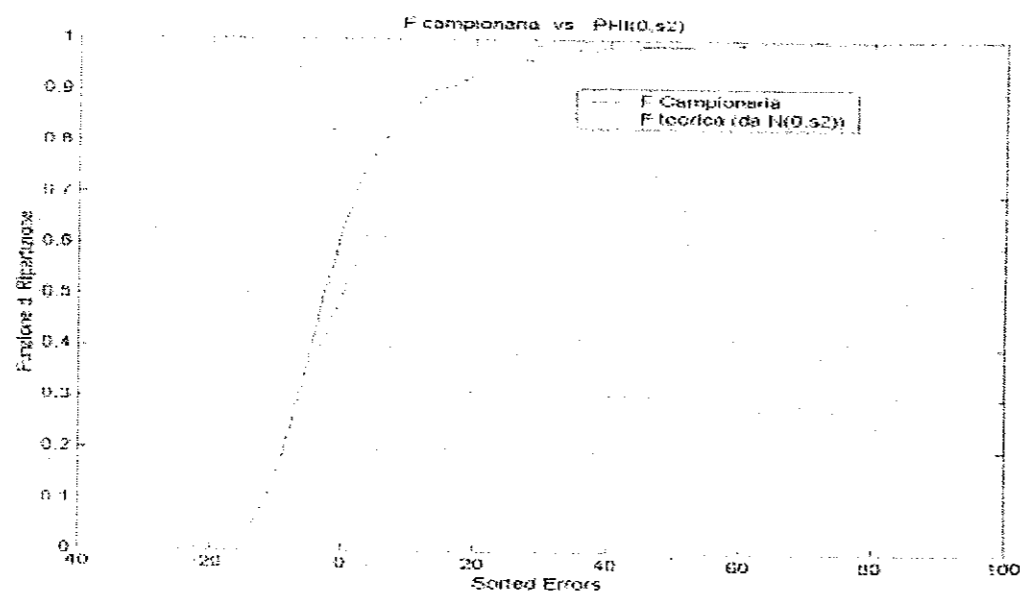


Fig 8.18. Funzione di ripartizione campionaria vs teorica dei residui ottenuti col programma GALMS e con criterio di selezione l'AIC.

Le Figg. 8.17 e 8.18 sottolineano la necessità dell'uso di modelli non lineari.

Si come i modelli statistici di durata sono espressi con funzioni di azzardo (hazard function)³ si è utilizzato il seguente modello:

$$(8.3) \quad h(t) = h_0(t) \exp(\sum_{i=1}^n a_i x_i) \Rightarrow \ln y = \sum_{i=1}^n a_i x_i$$

dove $h(t)$ esprime la probabilità che si fuoriesca dalla situazione di disoccupazione nell'istante che va da t a dt , (t) la probabilità di transizione iniziale o funzione di rischio base (cioè la probabilità di transizione da uno stato di disoccupazione ad uno

indipendentemente dai valori assunti dalle variabili esplicative), $\exp(\sum_{i=1}^n a_i x_i)$ esprime l'impatto dei regressori selezionati su $h(t)$.

Si è scelta la forma log-lineare del modello a rischi proporzionali (modello di Cox) in quanto è una delle più utilizzate.

Indicando con y la durata della disoccupazione si può scrivere:

$$(8.4) \quad y = e^{\sum_{i=1}^n a_i x_i} = \exp(\sum_{i=1}^n a_i x_i)$$

In questo modo si rende possibile il campionamento dei dati in tre gruppi (training set, validation set e test set) in quanto la variabilità della y viene ridotta sensibilmente (valori compresi tra -2.3 e +4.6) e con esso l'effetto delle "dummy" (rendendo possibile la stabilità del modello anche in caso di campionamento non bilanciato).

I risultati ottenuti sono riassunti in tabella 8.10. Anche in questo caso si è lavorato con i programmi "Backward", "Forward" e di selezione genetica come sopra.

³ Se X è un numero aleatorio che rappresenta la durata di un certo fenomeno (per esempio il tempo di vita dell'oggetto di studio) e k una data temporale, la funzione di azzardo è definita come $h(k) = P(X = k | X \geq k)$. Tale funzione esprime la probabilità di morte dell'oggetto nel giorno k , data la sopravvivenza fino all'inizio del giorno k .

METODO	CRITERIO	DIMENSIONE	SQM	R ²
Backward	AIC	20	0.8932	0.1484
Backward	BIC	25	0.8881	0.1584
Backward	HAN	14	0.9162	0.1016
Backward	SIC	16	0.9034	0.1271
Backward	p-value	7	0.9213	0.0941
Forward	HAN	14	0.9162	0.1016
Forward	SIC	16	0.9137	0.1073
Forward	P_corr	5	1.1279	0.0880
GALMS	SIC	10	1.1773	0.1640

Tabella 8.10. Risultati ottenuti col modello esponenziale.

Confrontando i dati delle tabelle si nota come il modello esponenziale ottenuto dalla selezione genetica utilizzando come criterio selettivo l'indicatore SIC sia il migliore in assoluto in termini di R^2 (sebbene presenti uno scarto quadratico medio più alto).

Prima di trarre conclusioni sull'analisi svolta esaminiamo le variabili scelte per questo modello e la coerenza dei segni dei coefficienti calcolati.

Il modello seleziona 10 parametri (tutti significativi), che sono:

- L'età e l'età al quadrato degli individui intervistati, i cui coefficienti hanno segno opposto (rispettivamente 1.220 e - 0.162), compensandosi così vicendevolmente;
- Il numero dei componenti della famiglia di età inferiore ai 5 anni, che presenta un coefficiente negativo (- 0.219) facilmente spiegabile col fatto che in famiglie con bambini piccoli la ricerca di un lavoro cresce sensibilmente la necessità di mantenere i figli;

- Il numero dei componenti della famiglia, con coefficiente positivo (0.118) giustificato dal fatto che la durata della disoccupazione aumenta per le persone che vivono in famiglie numerose poiché vi è un fenomeno di compensazione con gli altri componenti;
- La condizione professionale finale, che indica se l'individuo fa parte del sottogruppo dei *disoccupati* in base alla definizione data in precedenza (appartengono alla forza lavoro). Il coefficiente negativo di questa variabile (- 0.243) è facilmente spiegabile in quanto un'individuo che non abbia mai smesso di cercare un lavoro è destinato a ricevere un'offerta più velocemente;
- L'uscita dal mercato del lavoro; questo parametro, che indica se l'individuo è stato messo fuori dal mercato del lavoro, mostra un coefficiente negativo (- 0.222) evidenziando che colui che è uscito da questo mercato può trovare un impiego più velocemente;
- L'uscita dal settore agricolo o edile, che indica se l'ultima attività è stata svolta nei due settori menzionati e presenta un coefficiente negativo (- 0.545) a dimostrazione che chi esce da questi settori si inserisce meglio nel mercato del lavoro;
- La preferenza del lavoro nel comune di residenza, che presenta un coefficiente positivo (0.304) il quale indica che quanto maggiori sono le richieste di questo tipo tanto più difficile è trovare un lavoro;
- Il percepimento di un sussidio, che sorprendentemente mostra un coefficiente negativo (- 0.361) evidenziando una certa concordanza con alcuni studi in questo settore [Nic79],[Gro90],[NaSt93]. D'altronde il sussidio potrebbe incentivare la non ricerca del lavoro fino a quando la scadenza del diritto a percepirlo è lontana; quando invece si avvicina, la ricerca potrebbe diventare più frenetica e l'accettazione dei posti di lavoro eventualmente offerti meno vincolata (coerentemente con la teoria del *job-search*).
- L'entrata nel settore agricolo o edile, con un coefficiente positivo (0.284) che dimostra che colui che ha in precedenza trovato un lavoro in questi settori presenta un periodo di disoccupazione maggiore.

In Fig. 8.19 è riportata la funzione di ripartizione campionaria e teorica dei residui ottenuti utilizzando questo modello. La situazione rispetto al modello lineare è decisamente migliorata.

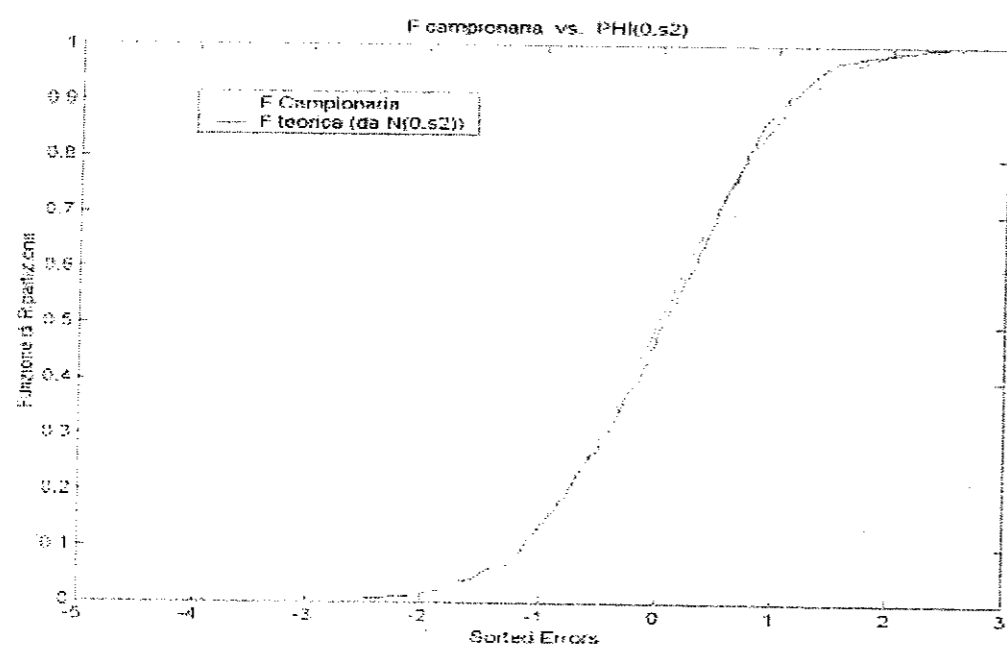


Fig 8.19. Funzione di ripartizione campionaria vs teorica dei residui ottenuti col modello esponenziale.

Dopo l'utilizzo di un modello non lineare come quello esponenziale si viene tentati dal desiderio di provarne anche uno neurale. Analogamente a quanto fatto con il lavoro del paragrafo precedente, anche qui i parametri "trasmessi" alla rete sono quelli trovati con il modello esponenziale sopra esposto.

I risultati presentano un R^2 uguale a 0.2835 con uno scarto quadratico medio di 0.9202 evidenziando un ulteriore miglioramento. In Fig. 8.20 è possibile osservare l'error-plot dei residui del modello e in Fig. 8.21 la funzione di ripartizione campionaria e quella teorica.

L'assenza di eteroschedasticità nei residui ottenuti utilizzando modelli non lineari mostra un netto miglioramento rispetto al modello di base (lineare).

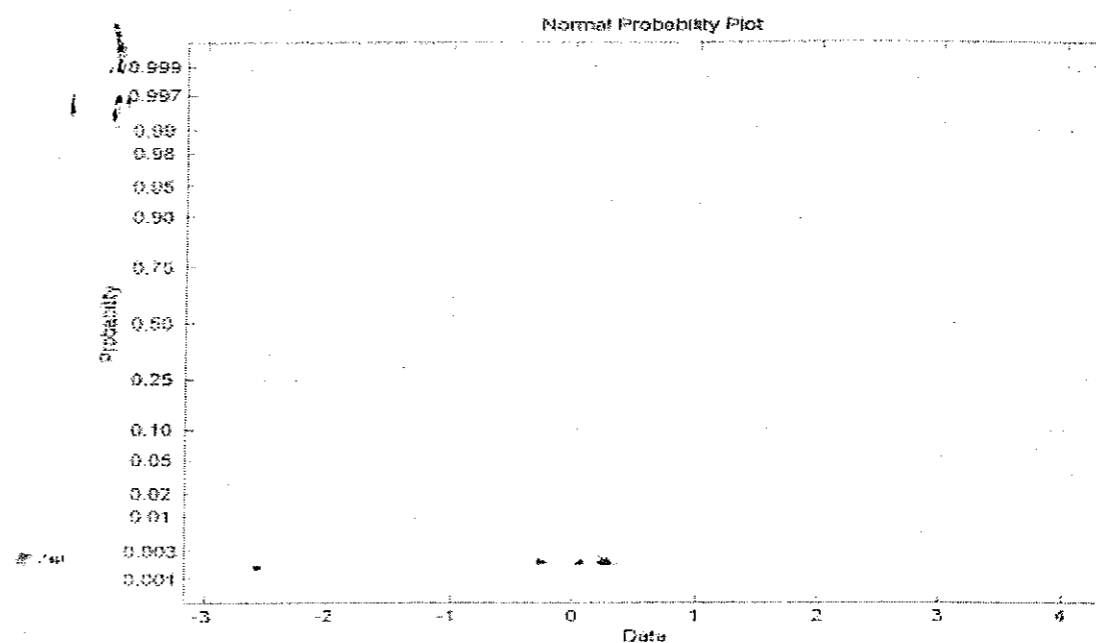


Fig 8.20. "Normal Probability Plot" dei residui ottenuti utilizzando il modello neurale.

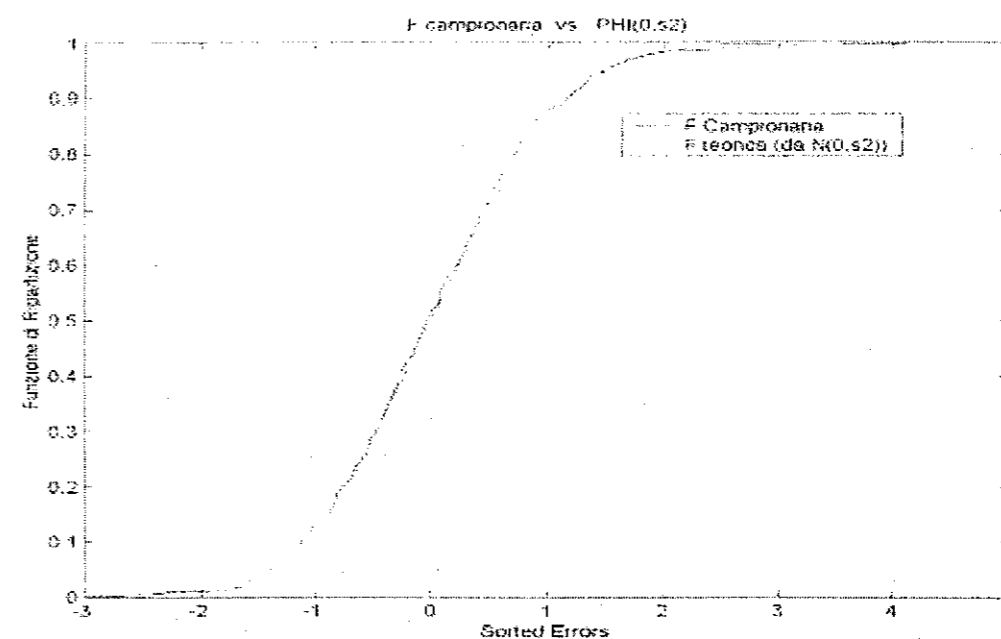


Fig 8.21. Funzione di ripartizione campionaria vs teorica dei residui ottenuti col modello neurale.

Molto utili per una valutazione dei risultati sono anche i valori dei test di adattamento di Kolmogorov e di Cramér e von Mises dei tre modelli sopra studiati riportati in tabella 8.11.

Modello	Criterio selettivo	Kolmogorov	Cramér e von Mises
Lineare	<i>AIC</i>	0.1441	4.55
Esponenziale	<i>SIC</i>	0.0439	0.3135
Neurale	<i>(miglior Esponenziale)</i>	0.0290	0.0901

Tabella 8.11. Risultati dei test di adattamento sui tre modelli esaminati (valori calcolati per $\alpha = 0.05$).

Questi mostrano un progressivo miglioramento di adattamento alla distribuzione ipotizzata passando dal modello lineare a quello neurale.

Per il modello esponenziale e per quello neurale è stato valutato anche l'indice di concordanza, c , considerando tutte le possibili coppie di osservazioni relative al test-set [Har84]. Per ogni data coppia la previsione si dice concordante se all'osservazione con una durata prevista maggiore corrisponde una durata osservata effettivamente più elevata. L'indice c corrisponde alla proporzione di concordanze su tutte le coppie possibili. Un indice $c=0.5$ rappresenta, quindi, un livello di concordanza associato ad un modello non predittivo, mentre un indice $c=1$ rappresenta un modello perfettamente predittivo (almeno dal punto di vista della concordanza). L'indice di concordanza per il modello esponenziale e per quello neurale è risultato pari a 0.58 per entrambi.

Anche in questo caso, come per l'analisi effettuata nel paragrafo precedente, sono valide le considerazioni e i suggerimenti per quanto riguarda gli ulteriori approfondimenti da effettuarsi sui modelli neurali.

CONCLUSIONI

In questa dispensa vengono prese in esame alcune delle principali tecnologie "intelligenti" per la selezione di un modello statistico mettendo a confronto in situazioni diverse vantaggi e svantaggi derivanti dal loro uso. Sono stati inoltre discussi i rapporti con gli approcci classici più tradizionali, implementando quando possibile, algoritmi automatizzati di modellizzazione statistica.

Sono stati presentati un insieme di algoritmi capaci di affrontare il problema della selezione del modello e dell'ottimizzazione non vincolata di una funzione a molte variabili senza imporre alcuna condizione di regolarità. In particolare sono state proposte procedure genetiche sia per la determinazione di modelli AR e ARMA nell'ambito dell'analisi previsiva delle serie temporali che della determinazione del modello in un ambito lineare multivariato orientato alla previsione.

Nel caso di modello statistico non lineare è stato implementato un algoritmo capace di affrontare il problema della costruzione del miglior modello neurale previsivo. In ambito multivariato è stato implementato un algoritmo di ottimizzazione non vincolata con l'obiettivo di fornire uno strumento di determinazione del minimo (o massimo) di una funzione multivariata che non rispetti le condizioni di regolarità, permettendo in questo modo la simulazione di Reti Neurali la cui funzione di errore sia determinabile dall'utente con gli stessi criteri degli altri parametri strutturali.

Sono state inoltre utilizzate alcune tipologie di Algoritmi Ibridi (Algoritmi Genetici + Logica Fuzzy, Algoritmi Genetici + Reti Neurali) allo scopo di facilitare l'operatore nell'analisi tecnica.

Tutte le tipologie di algoritmi utilizzati sono stati testati su un largo set di dati simulati prima di essere applicati alla soluzione di problemi di natura finanziaria, ambientale e sociale.

In particolare, per quanto concerne questi ultimi due problemi, gli strumenti ottenuti sono stati applicati a due casi sperimentali riguardanti la previsione del livello di marea a Venezia del 1984 e il livello di disoccupazione registrato in Emilia Romagna tra il 1993 e il 1995.

I risultati forniti sia dalle simulazioni che dalle analisi sperimentali sono sicuramente interessanti sebbene spesso non è stato possibile approfondire le analisi per brevità o per i limiti computazionali imposti dalle macchine. I metodi utilizzati hanno comunque permesso di accertare alcune tendenze di tipo operativo: la sostituzione di persone esperte con tecnologie informatiche deve essere considerato solo in caso estremo, mentre lo scopo principale per cui si ricorre a tali tecniche è quello di incrementare l'efficienza e di migliorare i livelli di coerenza ed affidabilità dei processi decisionali sfruttando nel migliore dei modi i dati e le informazioni a disposizione; le procedure automatizzate con metodologie evolute si sono rivelate più efficienti e robuste, nella generalità dei casi, rispetto a quelle classiche (anch'esse automatizzate).

Si ricorda infine che tutti gli strumenti implementati sono stati scritti in linguaggi MATLAB utilizzando, quando possibile, le funzioni inserite nei seguenti toolbox: Statistic toolbox, Genetic Algorithm toolbox, Neural Network toolbox, Optimization toolbox e System Identification toolbox.

Tutte le routine sono state testate sia in ambiente Unix su macchine "Sun Ultrasparc II" (che permette una migliore approssimazione di calcolo rispetto ai PC con gli attuali sistemi operativi) sia in ambiente WinTel.

BIBLIOGRAFIA

- [Aar89] Aarts E., Korst J., *Simulated Annealing and Boltzmann Machines*, J. Wiley & Sons, New York, 1989.
- [Adl93] Adler D., *Genetic Algorithm and Simulated Annealing: a Marriage Proposal*, IEEE International Conference on Neural Network, pp. 1104-1109, San Francisco, 1993.
- [Akai69] Akaike A., *Statistical predictor identification*, Annals of the Institute of Statistical Mathematics 22, pp. 203-217, 1969.
- [Akai73] Akaike A., *Information theory and an extension of the maximum likelihood principle*. In B.N. Petrov and F. Csai ed. "2nd International Symposium on Information Theory", pp. 267-281, Akademia Kiado, Budapest, 1973.
- [Akai78] Akaike A., *A Bayesian analysis of the minimum AIC procedure*, Annals of the institute of Statistical Mathematics 30, part A, pp. 9-14, 1978.
- [AnKo99] Anders U., Korn O., *Model selection in neural networks*, Neural Networks 12, pp. 309-323, 1999.
- [AV96] AAVV., *Using MATLAB*, The Maths Work, Natick, 1996.
- [Azo94] Azoff M.E., *Neural Network time series forecasting of financial makets*, J.Wiley & Sons, Chichester, 1994.
- [Bar83] Barto A., Sutton R., Anderson C., *Neuron-like adaptive elements can solve difficult learning control problems*, IEEE Transactions on Systems, Man and Cybernetics, vol.13, No.5, pp.834-846, 1983.
- [Bil95] Billings A., Zheng L., *Radial Basis Function Network configuration Using Genetic Algorithms*, Neural Networks, Vol.8, No.6, pp.877-890, Elsevier Science Ltd., 1995.
- [BrLa94] Bremer R.H., Langevin G.J., *The Genetic Algorithm for identifying the structure of a fixed model*, 1994.

- [Broy70] Broyden C.G., *The Convergence of a Class of Double-rank Minimization Algorithms*, J. Inst. Maths. Applics., Vol. 6, pp. 76-90, 1970.
- [BrVF98] Brown P.J., Vannucci M., Feam T., *Multivariate Bayesian variable selection and prediction*, J. of the Royal Statistical Society B (Statistical Methodology) 60, pp. 627-641, 1998.
- [Busc97] Buscema M., *Reti Neurali e finanza: esercizi, idee, metodi, applicazioni*, Armando, Roma, 1997.
- [CaPT86] Canestrelli P., Pastore F., Tommasin A., *Sviluppi di un modello operativo previsionale delle maree di Venezia e revisione di casi rilevanti*, Comune di Venezia, Assessorato ai Trasporti e Servizi Pubblici, 1986.
- [Chi95] A.Chipperfield, P.Fleming, H.Pohlheim, C.Fonseca, *Genetic Algorithm Toolbox User's Guide*, University of Sheffield, 1995.
- [CLL96] Chatterjee S., Laudato M., Lynch L.A., *Genetic algorithms and their statistical applications: an introduction*, Computational Statistics & Data Analysis 22, pp. 633-651, 1996.
- [Col99] Coleman T., Branch M.A., Grace A., *Optimization Toolbox for use with Matlab*, The Math Works Inc., Natick, 1999.
- [Cox94] Cox E., *Fuzzy system handbook: a practitioner's guide to building, using, and maintaining fuzzy system*, Academic Press, Boston, 1994.
- [Dav59] Davidon W.C., *Variable Metric Method for Minimization*, A.E.C. Research and Development Report, ANL-5990, 1959.
- [Davi91] L.Davis., *Handbook of Genetic Algorithm*, Van Nostrand Reinhold, New York, 1991.
- [DeBe97] H.Demuth, M.Beale, *Neural Networks Toolbox for use with Matlab*, The Maths Works Inc., Natick, 1997.
- [DrSm66] N.R.Draper, H. Smith, *Applied Regression Analysis, Wiley series in probability and mathematical statistics*, J. Wiley & Sons, New York, 1966.
- [DuPr80] Dubois D., Prada H., *Fuzzy Sets and System: Theory and Application*, Academic Press, San Diego, 1980.

- [Elm90] Elman J.L., *Finding structure in time*, Cognitive Science, vol.14, pp.179-211, 1990.
- [Enc98] *Enciclopedia Encarta98*, su CD-Rom, Microsoft, 1998.
- [Fama70] Fama E., *Efficient Capital Markets: A review of Theory and Empirical Work*, Journal of Finance, Vol. 25, pp. 383-417, 1970.
- [Flet70] Fletcher R., *A New Approach to Variable Metric Algorithms*, Computer Journal, Vol. 13, pp. 317-322, 1970.
- [Flet87] Fletcher R., *Practical methods of optimization, Second edition*, J. Wiley & Sons, Chichester, 1987.
- [FlPo63] Fletcher R., Powell M.J.D., *A Rapidly Convergent Descent Method for Minimization*, Computer J., Vol. 6, pp. 163-168, 1963.
- [GaMi99] M. Galli, T. Minerva "Algoritmi Genetici per l' Evoluzione di Modelli Lineari", Materiali di Discussione del Dipartimento di Economia Politica, 284, Università di Modena 1999.
- [Geis95] Geisser S., *Predictive Inference: An Introduction*, Chapman & Hall, New York, 1995.
- [GoKh95] Goonatilake S., Khebbal S., *Intelligent Hybrid Systems*, J. Wiley, 1985.
- [Gol70] Goldfarb D., *A Family of Variable Metric Updates Derived by Variational Means*, Mathematics of Computing, Vol. 24, pp. 23-26, 1970.
- [Gold89] Goldberg D.E., *Genetic Algorithms in search, optimization, and machine learning*, Addison-Wesley Publishing Corporation Inc., Reading (Mass.), 1989.
- [Goli94] Golinelli R., *Metodi Econometrici di base per l'analisi delle serie storiche: alcune applicazioni pratiche sul personal computer*, Cluep, Bologna, 1994.
- [Gre97] Greene W.H., *Econometric analysis*, McMillan, New York, 1997.
- [Gro90] Groot W., *The Effects of Benefit and Duration Dependence on Re-Employment Probabilities*, Economic Letters, 32, 4, pp. 371-376, 1990.
- [Gros82] S.Grossberg, *Studies of the Mind and Brain*, Dordrecht, Holland: Reidel Press, 1982.

- [Guo92] Guo Z., Uhrig R.E., *Using genetic Algorithms to Select Inputs for Neural Networks*, 1992.
- [Hag96] M.T.Hagan, H.B.Demuth, M.Beale, *Neural Network Design*, PWS-Kent, Boston, 1995.
- [Hair95] Hair Joseph F. Jr., *Multivariate data analysis: with readings*, Prentice-Hall, Englewood Cliffs, 1995.
- [HaQu79] Hannan E.J., Quinn B.G. *The determination of the order of an autoregression*, Journal of the Royal Statistical Society, B 41, pp. 190-195, 1979.
- [Har84] Harrel et al., *Regression modelling strategies for improved prognostic predictions*, Statistics in Medicine, 3, pp. 143-152, 1984.
- [Har93] Harvey A.C., *Time Series Model*, Harvester Wheatsheaf, New York, 1993.
- [Hin84] Hinton G., Sejnowski T., Ackley D., *Boltzmann Machines: Constraint Satisfaction Networks That Learn*, Carnegie Mellon University Technical Report, 1984.
- [Holl75] Holland J. H., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, 1975.
- [Holl95] J.H.Holland, *Hidden Order*, Reading, Addison Wesley Publishing Corporatin, Inc, 1995.
- [Hod97] Hodges E.M., *Applications of Genetic Algorithms in Time Series Analysis*, 1997.
- [HuTs89] Hurvich C.M., Tsai C.L., *Regression and time series model selection in small samples*, Biometrika 76, pp. 297-307, 1989.
- [HuTS] Hurvich C.M., Tsai C.L., Simonoff J.S., *Smoothing parameter selection in nonparametric regression using an improved Akaike information criterion*, J of the Royal Society B (Statistical methodology) 60, pp. 271-293, 1998.
- [HwDi97] Hwang J.T.G., Ding A.A., *Prediction Intervals for Artificial Neural Networks*, J. of the American Statistical Association 92, pp. 748-757, 1997.

- [Koho82] Kohonen T., *Self Organized Formation of Topologically Correct Feature Maps*, Biological Cybernetics 43, p. 59, 1982.
- [LaMi99] M. Lalla and T. Minerva, "Duration Models and Neural Networks to Analyse Unemployment Spells", Book of Short Papers SCO99 – Computational Statistics 1999 – Venezia 26-28 Settembre e sottoposto per la pubblicazione sulla rivista Metron.
- [LaPa99] Lalla M., Pattarin F., *Alcuni modelli per l'analisi delle durate complete e incomplete della disoccupazione: il caso Emilia-Romagna*, Working Paper n. 11, Dipartimento di Scienze Statistiche Università degli Studi di Padova, 1999.
- [Lev44] Leyenberg K., *A Method for the Solution of Certain Problems in Least Squares*, Quart. Appl. Math., Vol. 2, pp. 164-168, 1944.
- [LiTs98] Li H., Tsay R.S., *A Unified Approach to Identifying Multivariate Time Series Models*, J. of the American Statistical association 93, pp. 770-782, 1998.
- [Lju95] Ljung L., *System Identification Toolbox User's Guide*, The Math Works Inc., Natick, 1995.
- [Mal73] Mallows C.L., *Some comments on Cp*, Technometrics 15, pp. 661-675, 1973.
- [Marq63] Marquardt D., *An Algorithm for Least-Squares Estimation of Nonlinear Parameters*, SIAM J. Appl. Math., Vol. 11, pp 431-441, 1963.
- [McCl78] McClave J.T., *Estimating the Order of Autoregressive Models: The Max χ^2 Method*, J. of the American Statistical Association 73, pp. 122-128, 1978.
- [MFA97] *MF Analisi* Software di analisi tecnica, vers.5.01, Agosto 1997.
- [Mil89] Miller G., Todd P., Hedge S. *Designing neural network using genetic algorithms*, in J.D.Schaffer, Proceedings of the Third International Conference on Genetic Algorithm, Morgan Kauffmann, 1989.
- [MiPo98] T. Minerva and I. Poli, "A Genetic Alorithms selection method for Predictive Neural Nets and Linear Models", Materiali di Discussione del Dipartimento di Economia Politica, 228, Universita' di Modena, 1998.

- [MiPo98a] T. Minerva and I. Poli, "Building an ARMA model by using a Genetic Algorithm", *Materiali di discussione del dipartimento di Economia Politica*, 229, Università di Modena, (1998).
- [MiPo00] T. Minerva and I. Poli, "Predicting Venice High Tides by using Radial Basis Networks and Genetic Algorithms", accettato per la pubblicazione sul volume a cura di S. Borra, R. Rocci, M. Vichi e M. Schader della collana "Studies in Classification, Data Analysis and Knowledge Organization" edito dalla Springer-Verlag, Berlin, 2000.
- [MiPo00a] T. Minerva, I. Poli, "Time Series Modelling by Genetic Evolution", in preparazione, 2000.
- [MiPP99] T. Minerva, S. Paterlini and I. Poli, "Algoritmi Ibridi per l'Analisi di Serie Storiche Finanziarie. Integrazione di Reti Neurali ed Algoritmi Genetici per la Previsione dell'Andamento del Contratto Future FIB30", *Scienza e Business* 3-4, 1999.
- [MiPP00] T. Minerva, I. Poli and S. Paterlini - "A Genetic Algorithm for Neural Network Design (GANND) - A Financial Application" in pubblicazione su *Economics & Complexity*, 2000.
- [Mit96] M. Mitchell, *An Introduction to Genetic Algorithms*, Massachusetts, MIT Press, 1996.
- [NaSt93] Narendranathan W., Stewart M.B., *Modelling the Probability of Leaving Unemployment: competing Risks Models with Flexible Base-line Hazards*, *J. of the Royal Statistical Society C (Applied Statistics)*, 42, pp. 63-83, 1993.
- [Nic79] Nickel S., *The Effect of Unemployment and Related Benefits on the Duration of Unemployment*, *Economic Journal* 89, pp. 63-83, 1979.
- [Not95] Nottola C., Rossignoli C., *Intelligenza artificiale in banca, tendenze evolutive ed esperienze operative a confronto*, FrancoAngeli, Milano, 1995.
- [Orsi95] Orsi R., *Probabilità e inferenza statistica*, Il Mulino, Bologna, 1995.

- [Pat98] Paterlini S., *Un algoritmo genetico per il disegno di una rete neurale artificiale: applicazione alla previsione del prezzo di chiusura del Fib30*, tesi di laurea, Università degli Studi di Modena, 1998.
- [Picc74] Piccolo D., *Analisi delle serie temporali: i processi autoregressivi del secondo ordine*, Centro di specializzazione e ricerche economico-agrarie per il mezzogiorno, Portici, 1974.
- [Picc90] Piccolo D., *Introduzione all'analisi delle serie storiche*, La Nuova Scientifica, Roma, 1990.
- [Picc98] Piccolo D., *Statistica*, Il Mulino, Bologna, 1998.
- [PoJo94] Poli I., Jones R.D., *A Neural Net Model for Prediction*, *J. of the American Statistical Association* 89, pp. 117-121, 1994.
- [Pri81] Priestley M.B., *Spectral Analysis and Time Series*, Vols. 1 e 2, Academic Press, New York, 1981.
- [QuTs98] McQuarrie Allan D.R., Tsai Chih-Ling, *Regression and Time Series Model Selection*, World Scientific, Singapore, 1998.
- [Ref95] Refenes Apostolos P., *Neural Networks in the capital markets*, J. Wiley & Sons, Chichester, 1995.
- [Rissa87] Rissanen J., *Modeling by Shortest Data Description*, *Automatica* 14, pp. 465-471, 1987.
- [RoFB97] Ronchetti E., Field C., Blanchard W., *Robust Linear Model Selection by Cross-Validation*, *J. of the American Statistical Association* 92, pp. 1017-1023, 1997.
- [Rud96] Rudolph S., *On a genetic Algorithm for the Selection of Optimally Generalizing Neural Network Topologies*, *Proceeding of the 2nd International Conference of Adaptive Computin in Engineering Design and Control '96*, University of Plymouth, United Kingdom, pp. 76-86, 1996.
- [Rum86] Rumelhart D.E., McClelland J.L., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. I, Cambridge, MA: MIT Press, 1986.

- [SC98] Minerva T., Paterlini S., *Tecniche Computazionali per la Statistica, l'Economia e la Finanza*, Materiale Didattico a supporto del corso di Statistica Computazionale A.A. 1998-99, Università degli studi di Modena e Reggio Emilia.
- [Schw78] Schwarz G., *Estimating the dimension of a model*, Annals of Statistics 6, pp. 461-464, 1978.
- [SeSr90] Sen A., Srivastava M., *Regression Analysis. Theory, Methods and Applications*, Springer-Verlag, New York, 1990.
- [Sha70] Shanno D.F., *Conditioning of Quasi-Newton Methods for Function Minimization*, Mathematics of Computing, Vol. 24, pp. 647-656, 1970.
- [ShTs98] Shi P., Tsai Chih-Ling, *A note on the unification of the Akaike information criterion*, J. of the Royal Statistical Society B (Statistical Methodology) 60, pp. 551-558, 1998.
- [The67] Theil H, *Economics and Information Theory*, North-Holland Publ. Co., Amsterdam, 1967.
- [The61] Theil H, *Economic Foprecasts and Policy*, North-Holland Publ. Co., Amsterdam, 1961.
- [Tib99] Tibaldi M., *Tecniche di programmazione non lineare*, Pitagora Editrice, Bologna, 1999.
- [Wel99] Weller E.A., Feuer E.J., Frey C.M., Wesley M.N., *Parametric relative survival regression using generalized linear models with application to Hodgkin's lymphoma*, J. of the Royal Statistical Society C (Applied Statistics) 48, pp. 79-89, 1999.
- [Whi89] White H., *Some Asymptotic Results for Learning in Single Hidden-Layer Feedforward Network Models*, J. of the American Statistical Association 84, pp. 1003-1013, 1989.
- [Whi92] White D., Sofge D., *Handbook of Intelligent Control*, New York, Van Nostrand Reinhold, 1992.
- [Zade65] Zadeh L.A., *Fuzzy Sets*, Information and Control, Vol.8, pp. 338-353, 1965.

- [Zade84] Zadeh L.A., *Making Computers Think Like People*, IEEE Spectrum, Vol. 21:8, pp. 26-32, 1984.
- [Zade94] Zadeh L.A., *Fuzzy Logic and Soft Computing: Issues, Contentions and Perspectives*, In: IIZUKA'94:3rd International Conference on Fuzzy Logic, Neural Nets and Soft Computing, pp. 1-2, Iizuka, Japan 1994.

1. Maria Cristina Marcuzzo [1985] "Yvan Violet Robinson (1903-1983)", pp. 134
2. Sergio Lugaresi [1986] "Le imposte nelle teorie del sovrappiù", pp. 26
3. Massimo D'Angelillo e Leonardo Paggi [1986] "PCI e socialdemocrazie europee. Quale riformismo?", pp. 158
4. Gian Paolo Caselli e Gabriele Pastrello [1986] "Un suggerimento hobsoniano sul terziario ed occupazione: il caso degli Stati Uniti 1960/1983", pp. 52
5. Paolo Bosi e Paolo Silvestri [1986] "La distribuzione per aree disciplinari dei fondi destinati ai Dipartimenti, Istituti e Centri dell'Università di Modena: una proposta di riforma", pp. 25
6. Marco Lippi [1986] "Aggregations and Dynamic in One-Equation Econometric Models", pp. 64
7. Paolo Silvestri [1986] "Le tasse scolastiche e universitarie nella Legge Finanziaria 1986", pp. 41
8. Mario Forni [1986] "Storie familiari e storie di proprietà. Itinerari sociali nell'agricoltura italiana del dopoguerra", pp. 165
9. Sergio Paba [1986] "Gruppi strategici e concentrazione nell'industria europea degli elettrodomestici bianchi", pp. 56
10. Nerio Naldi [1986] "L'efficienza marginale del capitale nel breve periodo", pp. 54
11. Fernando Vianello [1986] "Labour Theory of Value", pp. 31
12. Piero Ganugi [1986] "Risparmio forzato e politica monetaria negli economisti italiani tra le due guerre", pp. 40
13. Maria Cristina Marcuzzo e Annalisa Rosselli [1986] "The Theory of the Gold Standard and Ricardo's Standard Comodity", pp. 30
14. Giovanni Solinas [1986] "Mercati del lavoro locali e carriere di lavoro giovanili", pp. 66
15. Giovanni Bonifati [1986] "Saggio dell'interesse e domanda effettiva. Osservazioni sul cap. 17 della General Theory", pp. 42
16. Marina Murat [1986] "Betwin old and new classical macroeconomics: notes on Lejonhufvud's notion of full information equilibrium", pp. 20
17. Sebastiano Brusco e Giovanni Solinas [1986] "Mobilità occupazionale e disoccupazione in Emilia Romagna", pp. 48
18. Mario Forni [1986] "Aggregazione ed esogeneità", pp. 13
19. Sergio Lugaresi [1987] "Redistribuzione del reddito, consumi e occupazione", pp. 17
20. Fiorenzo Sperotto [1987] "L'immagine neopopulista di mercato debole nel primo dibattito sovietico sulla pianificazione", pp. 34
21. M. Cecilia Guerra [1987] "Benefici tributari nel regime misto per i dividendi proposto dalla commissione Sarcinelli: una nota critica", pp. 9
22. Leonardo Paggi [1987] "Contemporary Europe and Modern America: Theories of Modernity in Comparative Perspective", pp. 38
23. Fernando Vianello [1987] "A Critique of Professor Goodwin's 'Critique of Sraffa'", pp. 12
24. Fernando Vianello [1987] "Effective Demand and the Rate of Profits. Some Thoughts on Marx, Kalecki and Sraffa", pp. 41
25. Anna Maria Sala [1987] "Banche e territorio. Approccio ad un tema geografico-economico", pp. 40
26. Enzo Mingione e Giovanni Mottura [1987] "Fattori di trasformazione e nuovi profili sociali nell'agricoltura italiana: qualche elemento di discussione", pp. 36
27. Giovanna Procacci [1988] "The State and Social Control in Italy During the First World War", pp. 18
28. Massimo Matteuzzi e Annamaria Simonazzi [1988] "Il debito pubblico", pp. 62
29. Maria Cristina Marcuzzo (a cura di) [1988] "Richard F. Kahn. A discipline of Keynes", pp. 118
30. Paolo Bosi [1988] "MICROMOD. Un modello dell'economia italiana per la didattica della politica fiscale", pp. 34
31. Paolo Bosi [1988] "Indicatori della politica fiscale. Una rassegna e un confronto con l'aiuto di MICROMOD", pp. 25
32. Giovanna Procacci [1988] "Protesta popolare e agitazioni operaie in Italia 1915-1918", pp. 45
33. Margherita Russo [1988] "Distretto Industriale e servizi. Uno studio dei trasporti nella produzione e nella vendita delle piastrelle", pp. 157
34. Margherita Russo [1988] "The effect of technical change on skill requirements: an empirical analysis", pp. 28
35. Carlo Grillenzoni [1988] "Identification, estimations of multivariate transfer functions", pp. 33
36. Nerio Naldi [1988] "'Keynes' concept of capital", pp. 40
37. Andrea Ginzburg [1988] "locomotiva Italia?", pp. 30
38. Giovanni Mottura [1988] "La 'persistenza' secolare. Appunti su agricoltura contadina ed agricoltura familiare nelle società industriali", pp. 40
39. Giovanni Mottura [1988] "L'anticamera dell'esodo. I contadini italiani della 'restaurazione contrattuale' fascista alla riforma fondiaria", pp. 40
40. Leonardo Paggi [1988] "Americanismo e riformismo. La socialdemocrazia europea nell'economia mondiale aperta", pp. 120
41. Annamaria Simonazzi [1988] "Fenomeni di isteresi nella spiegazione degli alti tassi di interesse reale", pp. 44
42. Antonietta Bassetti [1989] "Analisi dell'andamento e della casualità della borsa valori", pp. 12
43. Giovanna Procacci [1989] "State coercion and worker solidarity in Italy (1915-1918): the moral and political content of social unrest", pp. 41
44. Carlo Alberto Magni [1989] "Reputazione e credibilità di una minaccia in un gioco bargaining", pp. 56
45. Giovanni Mottura [1989] "Agricoltura familiare e sistema agroalimentare in Italia", pp. 84
46. Mario Forni [1989] "Trend Cycle and 'Fortuitous cancellation': a Note on a Paper by Nelson and Plosser", pp. 4
47. Paolo Bosi, Roberto Golinelli, Anna Stagni [1989] "Le origini del debito pubblico e il costo della stabilizzazione", pp. 26
48. Roberto Golinelli [1989] "Note sulla struttura e sull'impiego dei modelli macroeconomici", pp. 21
49. Marco Lippi [1989] "A Short Note on Cointegration and Aggregation", pp. 11
50. Gian Paolo Caselli e Gabriele Pastrello [1989] "The Linkage between Tertiary and Industrial Sector in the Italian Economy: 1951-1988. From an External Dependence to an International One", pp. 40
51. Gabriele Pastrello [1989] "Francois quesnay: dal Tableau Zig-zag al Tableau Formule: una ricostruzione", pp. 48
52. Paolo Silvestri [1989] "Il bilancio dello stato", pp. 34
53. Tim Mason [1990] "Tre seminari di storia sociale contemporanea", pp. 26
54. Michele Lalla [1990] "The Aggregate Escape Rate Analysed through the Queueing Model", pp. 23
55. Paolo Silvestri [1990] "Sull'autonomia finanziaria dell'università", pp. 11
56. Paola Bertolini, Enrico Giovannetti [1990] "Uno studio di 'filiera' nell'agroindustria. Il caso del Parmigiano Reggiano", pp. 164

57. Paolo Bosi, Roberto Golinelli, Anna Stagni [1990] "Effetti macroeconomici, settoriali e distributivi dell'armonizzazione dell'IVA", pp. 24
58. Michele Lalla [1990] "Modelling Employment Spells from Emilia Labour Force Data", pp. 18
59. Andrea Ginzburg [1990] "Politica Nazionale e commercio internazionale", pp. 22
60. Andrea Giommi [1990] "La probabilità individuale di risposta nel trattamento dei dati mancanti", pp. 13
61. Gian Paolo Caselli e Gabriele Pastrello [1990] "The service sector in planned economies. Past experiences and future perspectives", pp. 32
62. Giovanni Solinas [1990] "Competenze, grandi industrie e distretti industriali. Il caso Magneti Marelli", pp. 23
63. Andrea Ginzburg [1990] "Debito pubblico, teorie monetarie e tradizione civica nell'Inghilterra del Settecento", pp. 30
64. Mario Forni [1990] "Incertezza, informazione e mercati assicurativi: una rassegna", pp. 37
65. Mario Forni [1990] "Misspecification in Dynamic Models", pp. 19
66. Gian Paolo Caselli e Gabriele Pastrello [1990] "Service Sector Growth in CPE's: An Unsolved Dilemma", pp. 28
67. Paola Bertolini [1990] "La situazione agro-alimentare nei paesi ad economia avanzata", pp. 20
68. Paola Bertolini [1990] "Sistema agro-alimentare in Emilia Romagna ed occupazione", pp. 65
69. Enrico Giovannetti [1990] "Efficienza ed innovazione: il modello "fondi e flussi" applicato ad una filiera agro-industriale", pp. 38
70. Margherita Russo [1990] "Cambiamento tecnico e distretto industriale: una verifica empirica", pp. 115
71. Margherita Russo [1990] "Distretti industriali in teoria e in pratica: una raccolta di saggi", pp. 119
72. Paolo Silvestri [1990] "La Legge Finanziaria. Voce dell'enciclopedia Europea Garzanti", pp. 8
73. Rita Paltrinieri [1990] "La popolazione italiana: problemi di oggi e di domani", pp. 57
74. Enrico Giovannetti [1990] "Illusioni ottiche negli andamenti delle Grandezze distributive: la scala mobile e l'appiattimento delle retribuzioni in una ricerca", pp. 120
75. Enrico Giovannetti [1990] "Crisi e mercato del lavoro in un distretto industriale: il bacino delle ceramiche. Sez. I", pp. 150
76. Enrico Giovannetti [1990] "Crisi e mercato del lavoro in un distretto industriale: il bacino delle ceramiche. Sez. II", pp. 145
77. Antonietta Bassetti e Costanza Torricelli [1990] "Una riqualificazione dell'approccio bargaining alla selezione di portafoglio", pp. 4
78. Antonietta Bassetti e Costanza Torricelli [1990] "Il portafoglio ottimo come soluzione di un gioco bargaining", pp. 15
79. Mario Forni [1990] "Una nota sull'errore di aggregazione", pp. 6
80. Francesca Bergamini [1991] "Alcune considerazioni sulle soluzioni di un gioco bargaining", pp. 21
81. Michele Grillo e Michele Polo [1991] "Political Exchange and the allocation of surplus: a Model of Two-party competition", pp. 34
82. Gian Paolo Caselli e Gabriele Pastrello [1991] "The 1990 Polish Recession: a Case of Truncated Multiplier Process", pp. 26
83. Gian Paolo Caselli e Gabriele Pastrello [1991] "Polish firms: Private Vices Public Virtues", pp. 20
84. Sebastiano Brusco e Sergio Paba [1991] "Connessioni, competenze e capacità concorrenziali nell'industria della Sardegna", pp. 25
85. Claudio Grimaldi, Rony Hamoui, Nicola Rossi [1991] "Non Marketable assets and households' Portfolio Choice: a Case of Study of Italy", pp. 38
86. Giulio Righi, Massimo Baldini, Alessandra Brambilla [1991] "Le misure degli effetti redistributivi delle imposte indirette: confronto tra modelli alternativi", pp. 47
87. Roberto Fanfani, Luca Lanini [1991] "Innovazione e servizi nello sviluppo della meccanizzazione agricola in Italia", pp. 35
88. Antonella Caiumi e Roberto Golinelli [1992] "Stima e applicazioni di un sistema di domanda Almost Ideal per l'economia italiana", pp. 34
89. Maria Cristina Marcuzzo [1992] "La relazione salari-occupazione tra rigidità reali e rigidità nominali", pp. 30
90. Mario Biagioli [1992] "Employee financial participation in enterprise results in Italy", pp. 50
91. Mario Biagioli [1992] "Wage structure, relative prices and international competitiveness", pp. 50
92. Paolo Silvestri e Giovanni Solinas [1993] "Abbandoni, esiti e carriera scolastica. Uno studio sugli studenti iscritti alla Facoltà di Economia e Commercio dell'Università di Modena nell'anno accademico 1990/1991", pp. 30
93. Gian Paolo Caselli e Luca Martinelli [1993] "Italian GPN growth 1890-1992: a unit root or segmented trend representation?", pp. 30
94. Angela Politi [1993] "La rivoluzione frantesa. I partigiani emiliani tra liberazione e guerra fredda, 1945-1955", pp. 55
95. Alberto Rinaldi [1993] "Lo sviluppo dell'industria metalmeccanica in provincia di Modena: 1945-1990", pp. 70
96. Paolo Emilio Mistrulli [1993] "Debito pubblico, intermediari finanziari e tassi d'interesse: il caso italiano", pp. 30
97. Barbara Pistoiesi [1993] "Modelling disaggregate and aggregate labour demand equations. Cointegration analysis of a labour demand function for the Main Sectors of the Italian Economy: 1950-1990", pp. 45
98. Giovanni Bonifati [1993] "Progresso tecnico e accumulazione di conoscenza nella teoria neoclassica della crescita endogena. Una analisi critica del modello di Romer", pp. 50
99. Marcello D'Amato e Barbara Pistoiesi [1994] "The relationship(s) among Wages, Prices, Unemployment and Productivity in Italy", pp. 30
100. Mario Forni [1994] "Consumption Volatility and Income Persistence in the Permanent Income Model", pp. 30
101. Barbara Pistoiesi [1994] "Using a VECM to characterise the relative importance of permanent and transitory components", pp. 28
102. Gian Paolo Caselli and Gabriele Pastrello [1994] "Polish recovery form the slump to an old dilemma", pp. 20
103. Sergio Paba [1994] "Imprese visibili, accesso al mercato e organizzazione della produzione", pp. 20
104. Giovanni Bonifati [1994] "Progresso tecnico, investimenti e capacità produttiva", pp. 30
105. Giuseppe Marotta [1994] "Credit view and trade credit: evidence from Italy", pp. 20
106. Margherita Russo [1994] "Unit of investigation for local economic development policies", pp. 25
107. Luigi Brighi [1995] "Monotonicity and the demand theory of the weak axioms", pp. 20
108. Mario Forni e Lucrezia Reichlin [1995] "Modelling the impact of technological change across sectors and over time in manufacturing", pp. 25
109. Marcello D'Amato and Barbara Pistoiesi [1995] "Modelling wage growth dynamics in Italy: 1960-1990", pp. 38
110. Massimo Baldini [1995] "INDIMOD. Un modello di microsimulazione per lo studio delle imposte indirette", pp. 37
111. Paolo Bosi [1995] "Regionalismo fiscale e autonomia tributaria: l'emersione di un modello di consenso", pp. 38
112. Massimo Baldini [1995] "Aggregation Factors and Aggregation Bias in Consumer Demand", pp. 33
113. Costanza Torricelli [1995] "The information in the term structure of interest rates. Can stochastic models help in resolving the puzzle?" pp. 25
114. Margherita Russo [1995] "Industrial complex, pôle de développement, distretto industriale. Alcune questioni sulle unità di indagine nell'analisi dello sviluppo.", pp. 45
115. Angelika Moryson [1995] "50 Jahre Deutschland. 1945 - 1995" pp. 21
116. Paolo Bosi [1995] "Un punto di vista macroeconomico sulle caratteristiche di lungo periodo del nuovo sistema pensionistico italiano.", pp. 32
117. Gian Paolo Caselli e Salvatore Curatolo [1995] "Esistono relazioni stimabili fra dimensione ed efficienza delle istituzioni e crescita produttiva? Un esercizio nello spirito di D.C. North.", pp. 11
118. Mario Forni e Marco Lippi [1995] "Permanent income, heterogeneity and the error correction mechanism.", pp. 21
119. Barbara Pistoiesi [1995] "Co-movements and convergence in international output. A Dynamic Principal Components Analysis" pp. 14
120. Mario Forni e Lucrezia Reichlin [1995] "Dynamic common factors in large cross-section" pp. 17
121. Giuseppe Marotta [1995] "Il credito commerciale in Italia: una nota su alcuni aspetti strutturali e sulle implicazioni di politica monetaria" pp. 20
122. Giovanni Bonifati [1995] "Progresso tecnico, concorrenza e decisioni di investimento: una analisi delle determinanti di lungo periodo degli investimenti" pp. 25
123. Giovanni Bonifati [1995] "Cambiamento tecnico e crescita endogena: una valutazione critica delle ipotesi del modello di Romer" pp. 21
124. Barbara Pistoiesi e Marcello D'Amato [1995] "La riservatezza del banchiere centrale è un bene o un male? Effetti dell'informazione incompleta sul benessere in un modello di politica monetaria.", pp. 32
125. Barbara Pistoiesi [1995] "Radici unitarie e persistenza: l'analisi univariata delle fluttuazioni economiche.", pp. 33
126. Barbara Pistoiesi e Marcello D'Amato [1995] "Co-movements in European real outputs" pp. 20
127. Antonio Ribba [1996] "Ciclo economico, modello lineare-stocastico, forma dello spettro delle variabili macroeconomiche" pp. 31
128. Carlo Alberto Magni [1996] "Repeatable and una tantum real options a dynamic programming approach" pp. 23
129. Carlo Alberto Magni [1996] "Opzioni reali d'investimento e interazione competitiva: programmazione dinamica stocastica in optimal stopping" pp. 26
130. Carlo Alberto Magni [1996] "Vaghezza e logica fuzzy nella valutazione di un'opzione reale" pp. 20
131. Giuseppe Marotta [1996] "Does trade credit redistribution thwart monetary policy? Evidence from Italy" pp. 20
132. Mauro Dell'Amico e Marco Trubian [1996] "Almost-optimal solution of large weighted equicut problems" pp. 30
133. Carlo Alberto Magni [1996] "Un esempio di investimento industriale con interazione competitiva e avversione al rischio" pp. 20
134. Margherita Russo, Peter Børkey, Emilio Cubel, François Lévêque, Francisco Mas [1996] "Local sustainability and competitiveness: the case of the ceramic tile industry" pp. 66
135. Margherita Russo [1996] "Camionetto tecnico e relazioni tra imprese" pp. 190
136. David Avra Lane, Irene Poli, Michele Lalla, Alberto Roverato [1996] "Lezioni di probabilità e inferenza statistica" pp. 288
137. David Avra Lane, Irene Poli, Michele Lalla, Alberto Roverato [1996] "Lezioni di probabilità e inferenza statistica - Esercizi svolti -" pp. 302
138. Barbara Pistoiesi [1996] "Is an Aggregate Error Correction Model Representative of Disaggregate Behaviours? An example" pp. 24
139. Luisa Malaguti e Costanza Torricelli [1996] "Monetary policy and the term structure of interest rates", pp. 30
140. Mauro Dell'Amico, Martine Labbé, Francesco Maffioli [1996] "Exact solution of the SONET Ring Loading Problem", pp. 20
141. Mauro Dell'Amico, R.J.M. Vaessens [1996] "Flow and open shop scheduling on two machines with transportation times and machine-independent processing times in NP-hard", pp. 10
142. M. Dell'Amico, F. Maffioli, A. Sciomechen [1996] "A Lagrangean Heuristic for the Pirze Collecting Travelling Salesman Problem", pp. 14
143. Massimo Baldini [1996] "Inequality Decomposition by Income Source in Italy - 1987 - 1993", pp. 20
144. Graziella Bertocchi [1996] "Trade, Wages, and the Persistence of Underdevelopment" pp. 20
145. Graziella Bertocchi and Fabio Canova [1996] "Did Colonization matter for Growth? An Empirical Exploration into the Historical Causes of Africa's Underdevelopment" pp. 32
146. Paola Bertolini [1996] "La modernization de l'agriculture italienne et le cas de l'Emilie Romagne" pp. 20
147. Enrico Giovannetti [1996] "Organisation industrielle et développement local: le cas de l'agroindustrie in Emilie Romagne" pp. 18
148. Maria Elena Bontempi e Roberto Golinelli [1996] "Le determinanti del leverage delle imprese: una applicazione empirica ai settori industriali dell'economia italiana" pp. 31
149. Paola Bertolini [1996] "L'agriculture et la politique agricole italienne face aux recents scenarios", pp. 20
150. Enrico Giovannetti [1996] "Il grado di utilizzo della capacità produttiva come misura dei costi di transazione: una rilettura di "Nature of the Firm" di R. Coase", pp. 75
151. Enrico Giovannetti [1996] "Il I° ciclo del Diploma Universitario Economia e Amministrazione delle Imprese", pp. 25
152. Paola Bertolini, Enrico Giovannetti, Giulia Santacaterina [1996] "Il Settore del Verde Pubblico. Analisi della domanda e valutazione economica dei benefici", pp. 35
153. Giovanni Solinas [1996] "Sistemi produttivi del Centro-Nord e del Mezzogiorno. L'industria delle calzature", pp. 55
154. Tindara Addabbo [1996] "Married Women's Labour Supply in Italy in a Regional Perspective", pp. 85
155. Paolo Silvestri, Giuseppe Catalano, Cristina Bevilacqua [1996] "Le tasse universitarie e gli interventi per il diritto allo studio: la prima fase di applicazione di una nuova normativa" pp. 159
156. Sebastiano Brusco, Paolo Bertossi, Margherita Russo [1996] "L'industria dei rifiuti urbani in Italia", pp. 25
157. Paolo Silvestri, Giuseppe Catalano [1996] "Le risorse del sistema universitario italiano: finanziamento e governo" pp. 400
158. Carlo Alberto Magni [1996] "Un semplice modello di opzione di differimento e di vendita in ambito discreto", pp. 10
159. Tito Pietra, Paolo Siconolfi [1996] "Fully Revealing Equilibria in Sequential Economies with Asset Markets" pp. 17
160. Tito Pietra, Paolo Siconolfi [1996] "Extrinsic Uncertainty and the Informational Role of Prices" pp. 42
161. Paolo Bertella Farnetti [1996] "Il negro e il rosso. Un precedente non esplorato dell'integrazione afroamericana negli Stati Uniti" pp. 26
162. David Lane [1996] "Is what is good for each best for all? Learning from others in the information contagion model" pp. 18

163. Antonio Ribba [1996] "A note on the equivalence of long-run and short-run identifying restrictions in cointegrated systems" pp. 10
164. Antonio Ribba [1996] "Scomposizioni permanenti-transitorie in sistemi cointegrati con una applicazione a dati italiani" pp. 23
165. Mario Forni, Sergio Paba [1996] "Economic Growth, Social Cohesion and Crime" pp. 20
166. Mario Forni, Lucrezia Reichlin [1996] "Let's get real: a factor analytical approach to disaggregated business cycle dynamics" pp. 25
167. Marcello D'Amato e Barbara Pistoiesi [1996] "So many Italies: Statistical Evidence on Regional Cohesion" pp. 31
168. Elena Bonfiglioli, Paolo Bosi, Stefano Toso [1996] "L'equità del contributo straordinario per l'Europa" pp. 20
169. Graziella Bertocchi, Michael Spagat [1996] "Il ruolo dei licei e delle scuole tecnico-professionali tra progresso tecnologico, conflitto sociale e sviluppo economico" pp. 37
170. Gianna Boero, Costanza Torricelli [1997] "The Expectations Hypothesis of the Term Structure of Interest Rates: Evidence for Germany" pp. 15
171. Mario Forni, Lucrezia Reichlin [1997] "National Policies and Local Economies: Europe and the US" pp. 22
172. Carlo Alberto Magni [1997] "La trappola del Roe e la tridimensionalità del Van in un approccio sistemico" pp. 16
173. Mauro Dell'Amico [1997] "A Linear Time Algorithm for Scheduling Outflows with Communication Delays on Two or Three Processor" pp. 18
174. Paolo Bosi [1997] "Aumentare l'età pensionabile fa diminuire la spesa pensionistica? Ancora sulle caratteristiche di lungo periodo della riforma Dini" pp. 13
175. Paolo Bosi e Massimo Matteuzzi [1997] "Nuovi strumenti per l'assistenza sociale" pp. 31
176. Mauro Dell'Amico, Francesco Maffioli e Marco Trubian [1997] "New bounds for optimum traffic assignment in satellite communication" pp. 21
177. Carlo Alberto Magni [1997] "Paradossi, inverosimiglianze e contraddizioni del Van: operazioni certe" pp. 9
178. Barbara Pistoiesi e Marcello D'Amato [1997] "Persistence of relative unemployment rates across Italian regions" pp. 25
179. Margherita Russo, Franco Cavedoni e Riccardo Pianesani [1997] "Le spese ambientali dei Comuni in provincia di Modena, 1993-1995" pp. 23
180. Gabriele Pastrello [1997] "Time and Equilibrium, Two Elusive Guests in the Keynes-Hawtrey-Robertson Debate in the Thirties" pp. 25
181. Luisa Malaguti e Costanza Torricelli [1997] "The Interaction Between Monetary Policy and the Expectation Hypothesis of the Term Structure of Interest rates in a N-Period Rational Expectation Model" pp. 27
182. Mauro Dell'Amico [1997] "On the Continuous Relaxation of Packing Problems - Technical Note" pp. 8
183. Stefano Bordini [1997] "Prova di Idoneità di Informatica Dispensa Esercizi Excel 5" pp. 49
184. Francesca Bergamini e Stefano Bordini [1997] "Una verifica empirica di un nuovo metodo di selezione ottima di portafoglio" pp. 22
185. Gian Paolo Caselli e Maurizio Battini [1997] "Following the tracks of atkinson and micklewright the changing distribution of income and earnings in Poland from 1989 to 1995" pp. 21
186. Mauro Dell'Amico e Francesco Maffioli [1997] "Combining Linear and Non-Linear Objectives in Spanning Tree Problems" pp. 21
187. Gianni Ricci e Vanessa Debbia [1997] "Una soluzione evolutiva in un gioco differenziale di lotta di classe" pp. 14
188. Fabio Canova e Eva Ortega [1997] "Testing Calibrated General Equilibrium Model" pp. 34
189. Fabio Canova [1997] "Does Detrending Matter for the Determination of the Reference Cycle and the Selection of Turning Points?" pp. 35
190. Fabio Canova e Gianni De Nicolò [1997] "The Equity Premium and the Risk Free Rate: A Cross Country, Cross Maturity Examination" pp. 41
191. Fabio Canova e Angel I. Ubide [1997] "International Business Cycles, Financial Market and Household Production" pp. 32
192. Fabio Canova e Gianni De Nicolò [1997] "Stock Returns, Term Structure, Inflation and Real Activity: An International Perspective" pp. 33
193. Fabio Canova e Morten Ravn [1997] "The Macroeconomic Effects of German Unification: Real Adjustments and the Welfare State" pp. 34
194. Fabio Canova [1997] "Detrending and Business Cycle Facts" pp. 40
195. Fabio Canova e Morten O. Ravn [1997] "Crossing the Rio Grande: Migrations, Business Cycle and the Welfare State" pp. 37
196. Fabio Canova e Jane Marrinan [1997] "Sources and Propagation of International Output Cycles: Common Shocks or Transmission?" pp. 41
197. Fabio Canova e Albert Marcet [1997] "The Poor Stay Poor: Non-Convergence Across Countries and Regions" pp. 44
198. Carlo Alberto Magni [1997] "Un Criterio Strutturalista per la Valutazione di Investimenti" pp. 17
199. Stefano Bordini [1997] "Elaborazione Automatica dei Dati" pp. 60
200. Paolo Bertella Farnetti [1997] "The United States and the Origins of European Integration" pp. 19
201. Paolo Bosi [1997] "Sul Controllo Dinamico di un Sistema Pensionistico a Ripartizione di Tipo Contributivo" pp. 17
202. Paola Bertolini [1997] "European Union Agricultural Policy: Problems and Perspectives" pp. 18
203. Stefano Bordini [1997] "Supporti Informatici per la Ricerca delle soluzioni di Problemi Decisionali" pp. 30
204. Carlo Alberto Magni [1997] "Paradossi, Inverosimiglianze e Contraddizioni del Van: Operazioni Aleatorie" pp. 10
205. Carlo Alberto Magni [1997] "Tir, Roe e Van: Distorsioni linguistiche e Cognitive nella Valutazione degli Investimenti" pp. 17
206. Gisella Facchinetti, Roberto Ghiselli Ricci e Silvia Muzzioli [1997] "New Methods For Ranking Triangular Fuzzy Numbers: An Investment Choice" pp. 9
207. Mauro Dell'Amico e Silvano Martello [1997] "Reduction of the Three-Partition Problem" pp. 16
208. Carlo Alberto Magni [1997] "IRR, ROE and NPV: a Systemic Approach" pp. 20
209. Mauro Dell'Amico, Andrea Lodi e Francesco Maffioli [1997] "Solution of the cumulative assignment problem with a well-structured tabu search method" pp. 25
210. Carlo Alberto Magni [1997] "La definizione di investimento e criterio del Tir ovvero: la realtà inventata" pp. 16
211. Carlo Alberto Magni [1997] "Critica alla definizione classica di investimento: un approccio sistemico" pp. 17
212. Alberto Roverato [1997] "Asymptotic prior to posterior analysis for graphical gaussian models" pp. 8
213. Tindara Addabbo [1997] "Povertà nel 1995 analisi statica e dinamica sui redditi familiari" pp. 64
214. Gian Paolo Caselli e Franca Manghi [1997] "La transizione da piano a mercato e il modello di Ising" pp. 15
215. Tindara Addabbo [1998] "Lavoro non pagato e reddito esteso un'applicazione alle famiglie italiane in cui entrambi i coniugi sono lavoratori dipendenti" pp. 54
216. Tindara Addabbo [1998] "Probabilità di occupazione e aspettative individuali" pp. 36
217. Lara Magnani [1998] "Transazioni, contratti e organizzazioni: una chiave di lettura della teoria economica dell'organizzazione" pp. 39
218. Michele Lalla, Rosella Molinari e Maria Grazia Modena [1998] "La progressione delle carriere: i percorsi in cardiologia" pp. 46
219. Lara Magnani [1998] "L'organizzazione delle transazioni di subfornitura nel distretto industriale" pp. 40
220. Antonio Ribba [1998] "Recursive VAR orderings and identification of permanent and transitory shocks" pp. 12
221. Antonio Ribba [1998] "Granger-causality and exogeneity in cointegrated Var models" pp. 5
222. Luigi Brighi e Marcello D'Amato [1998] "Optimal Procurement in Multiproduct Monopoly" pp. 25
223. Paolo Bosi, Maria Cecilia Guerra e Paolo Silvestri [1998] "La spesa sociale nel comune Modena" Rapporto intermedio pp. 37
224. Mario Forni e Marco Lippi [1998] "On the Microfoundations of Dynamic Macroeconomics" pp. 22
225. Roberto Ghiselli Ricci [1998] "Nuove Proposte di Ordinamento di Numeri Fuzzy. Una Applicazione ad un Problema di Finanziamento" pp. 7
226. Tommaso Minerva [1998] "Internet Domande e Risposte" pp. 183
227. Tommaso Minerva [1998] "Elementi di Statistica Computazionale. Parte Prima. Il Sistema Operativo Unix ed il Linguaggio C" pp. 57
228. Tommaso Minerva e Irene Poli [1998] "A Genetic Algorithms Selection Method for Predictive Neural Nets and Linear Models" pp. 60
229. Tommaso Minerva e Irene Poli [1998] "Building an ARMA Model by using a Genetic Algorithm" pp. 60
230. Mauro Dell'Amico e Paolo Toth [1998] "Algorithms and Codes for Dense Assignment Problems: the State of the Art" pp. 35
231. Ennio Cavazzuti e Nicoletta Pacchiarotti [1998] "How to play an hotelling game in a square town" pp. 12
232. Alberto Roverato e Irene Poli [1998] "Un algoritmo genetico per la selezione di modelli grafici" pp. 11
233. Marcello D'Amato e Barbara Pistoiesi [1998] "Delegation of Monetary Policy to a Central Banker with Private Information" pp. 15
234. Graziella Bertocchi e Michael Spagat [1998] "The Evolution of Modern Educational Systems. Technical vs. General Education, Distributional Conflict, and Growth" pp. 31
235. André Dumas [1998] "Le système monétaire Européen" pp. 24
236. Gianna Boero, Gianluca Di Lorenzo e Costanza Torricelli [1998] "The influence of short rate predictability and monetary policy on tests of the expectations hypothesis: some comparative evidence" pp. 30
237. Carlo Alberto Magni [1998] "A systemic rule for investment decisions: generalizations of the traditional DCF criteria and new conceptions" pp. 30
238. Marcello D'Amato e Barbara Pistoiesi [1998] "Interest Rate Spreads Between Italy and Germany: 1995-1997" pp. 16
239. Paola Bertolini e Alberto Bertacchini [1998] "Il distretto di lavorazioni carni suine in provincia di Modena" pp. 29
240. Costanza Torricelli e Gianluca Di Lorenzo [1998] "Una nota sui fondamenti matematico-finanziari della teoria delle aspettative della struttura della scadenza" pp. 15
241. Christophe Croux, Mario Forni e Lucrezia Reichlin [1998] "A Measure of Comovement for Economic Indicators: Theory and Empirics" pp. 23
242. Carlo Alberto Magni [1998] "Note sparse sul dilemma del prigioniero (e non solo)" pp. 13
243. Gian Paolo Caselli [1998] "The future of mass consumption society in the former planned economies: a macro approach" pp. 21
244. Mario Forni, Marc Hallin, Marco Lippi e Lucrezia Reichlin [1998] "The generalized dynamic factor model: identification and estimation" pp. 35
245. Carlo Alberto Magni [1998] "Pictures, language and research: the case of finance and financial mathematics" pp. 35
246. Luigi Brighi [1998] "Demand and generalized monotonicity" pp. 21
247. Mario Forni e Lucrezia Reichlin [1998] "Risk and potential insurance in Europe" pp. 20
248. Tommaso Minerva, Sandra Paterlini e Irene Poli [1998] "A Genetic Algorithm for predictive Neural Network Design (GANND). A Financial Application" pp. 12
249. Gian Paolo Caselli Maurizio Battini [1998] "The Changing Distribution of Earnings in Poland from 1989 to 1996" pp. 9
250. Mario Forni, Sergio Paba [1998] "Industrial Districts, Social Environment and Local Growth" Evidence from Italy pp. 27
251. Lara Magnani [1998] "Un'analisi del distretto industriale fondata sulla moderna teoria economica dell'organizzazione" pp. 46
252. Mario Forni, Lucrezia Reichlin [1998] "Federal Policies and Local Economies: Europe and the US" pp. 24
253. Luigi Brighi [1998] "A Case of Optimal Regulation with Multidimensional Private Information" pp. 20
254. Barbara Pistoiesi, Stefania Luppi [1998] "Gli investimenti diretti esteri nell'America Latina e nel Sud Est Asiatico: 1982-1995" pp. 27
255. Paola Mengoli, Margherita Russo [1998] "Technical and Vocational Education and Training in Italy: Structure and Changes at National and Regional Level" pp. 25
256. Tindara Addabbo [1998] "On-the-Job Search a Microeconomic Analysis on Italian Data" pp. 29
257. Lorenzo Bertucelli [1999] "Il paternalismo industriale: una discussione storiografica" pp. 21
258. Mario Forni e Marco Lippi [1999] "The generalized dynamic factor model: representation theory" pp. 25
259. Andrea Ginzburg e Annamaria Simonazzi [1999] "Foreign debt cycles and the 'Gibson Paradox': an interpretative hypothesis" pp. 38
260. Paolo Bosi [1999] "La riforma della spesa per assistenza dalla Commissione Onofri ad oggi: una valutazione in corso d'opera" pp. 56
261. Marcello D'Amato e Barbara Pistoiesi [1999] "Go and soothe the row. Delegation of monetary policy under private information" pp. 23
262. Michele Lalla [1999] "Sampling, Maintenance, and Weighting Schemes for Longitudinal Surveys: a Case Study of the Textile and Clothing Industry" pp. 27
263. Pederzoli Chiara e Torricelli Costanza [1999] "Una rassegna sui metodi di stima del Value at Risk (Var)"
264. Paolo Bosi, Maria Cecilia Guerra e Paolo Silvestri [1999] "La spesa sociale di Modena. La valutazione della condizione economica" pp. 74
265. Graziella Bertocchi e Michael Spagat [1999] "The Politics Co-optation" pp. 14
266. Giovanni Bonifati [1999] "The Capacity to Generate Investment. An analysis of the long-term determinants of investment" pp. 22
267. Tindara Addabbo e Antonella Caiumi [1999] "Extended Income and Inequality by Gender in Italy" pp. 40
268. Antonella Caiumi e Federico Perali [1999] "Children and Intra-household Distribution of Resources: An Estimate of the Sharing Rule of Italian Households" pp. 24
269. Vincenzo Atella, Antonella Caiumi e Federico Perali [1999] "Una scala di equivalenza non vale l'altra" pp. 23

- 270 Tito Pietra e Paolo Siconolfi [1999] "Volume of Trade and Revelation of Information" pp. 33.
- 271 Antonella Picchio [1999] "La questione del lavoro non pagato nella produzione di servizi nel nucleo domestico (Household)" pp.58.
- 272 Margherita Russo [1999] "Complementary Innovations and Generative Relationships in a Small Business Production System: the Case of Kervit" pp.27.
- 273 André Dumas [1999] "L'Economie de la drouge" pp. 12.
- 274 André Dumas [1999] "L'Euro à l'heure actuelle" pp. 12.
- 275 Michele Lalla Gisella Facchinetti [1999] "La valutazione dell'attività didattica: un confronto tra scale di misura e insiemi sfocati" pp.32.
- 276 Mario Biagioli [1999] "Formazione e valorizzazione del capitale umano: un'indagine sui paesi dell'Unione Europea" pp.21.
- 277 Mario Biagioli [1999] "Disoccupazione, formazione del capitale umano e determinazione dei salari individuali: un'indagine su microdati nei paesi dell'Unione Europea" pp.15.
- 278 Gian Paolo Caselli Giulia Bruni [1999] Il settore petrolifero russo, il petrolio del Mar Caspio e gli interessi geopolitici nell'area" pp. 28.
- 279 Luca Gambetti [1999] "The Real Effect of Monetary Policy: a New Var Identification Procedure" pp.22.
- 280 Marcello D'Amato Barbara Pistori [1999] "Assessing Potential Targets for Labour Market Reforms in Italy" pp. 8.
- 281 Gian Paolo Caselli Giulia Bruni e Francesco Pattarin [1999] "Gaddy and Ickes Model of Russian Barter Economy: Some Criticisms and Considerations" pp.10.
- 282 Silvia Muzzioli Costanza Torricelli [1999] "A Model for Pricing an Option with a Fuzzy Payoff" pp. 13.
- 283 Antonella Caiumi Federico Perali [1999] "Povertà e Welfare in Italia in Relazione alla Scelta della Scala di Equivalenza" pp.25.
- 284 Marcello Galli Tommaso Minerva [1999] "Algoritmi Genetici per l'Evoluzione di Modelli Lineari *Metodologia ad Applicazioni*" pp.36.
- 285 Mario Forni Sergio Paba [1999] "Knowledge Spillovers and the Growth of Local Industries" pp. 20.
- 286 Gisella Facchinetti Giovanni Mastroleo [1999] "Un confronto tra uno score card ed un approccio fuzzy per la concessione del credito personale" pp.27.
- 287 Gisella Facchinetti Giovanni Mastroleo e Sergio Paba [1999] "A Statistical and Fuzzy Algorithm for the Identification of Industrial Districts" pp.6.
- 288 Tommaso Minerva [1999] "Didattica e Informatica. Una indagine Statistica relativa alla Provincia di Modena sul rapporto tra Insegnanti e Nuove Tecnologie" pp. 46.
- 289 Andrea Ginzburg [1999] "Sraffa e l'analisi sociale: alcune note metodologiche" pp. 37.
- 290 Consolato Pellegrino Carla Fiori [1999] "Piani Formalmente Euclidei" pp. 11.
- 291 Nicolina A. Malara, Maria Teresa Brandoli e Carla Fiori [1999] "Comportamenti di Studenti in Ingresso all'Università di Fronte allo Studio di Disequazioni" pp. 15.
- 292 Consolato Pellegrino Maria Teresa Brandoli [1999] "Il Principio D'Induzione Euristica-Mente Parlando" pp. 11.
- 293 Paolo Bertella Farnetti [1999] "Winston Churchill e l'unità europea" pp. 25.
- 294 Tindara Addabbo Massimo Baldini [1999] "Safety net and poverty dynamics in Italy in the early nineties" pp. 23.
- 295 Margherita Russo [2000] "Innovation Dynamics and Industrial Dynamics in a Local Production System. Changes in the Agents/Artifacts Space in Tile Decoration: from Silk Screen to Laser Engraved Silicon Cylinder" pp 45.
- 296 Gianluca Masci e Margherita Russo [2000] "L'attività brevettale nel distretto ceramico, 1971-1998" pp 41.
- 297 Paola Mengoli e Margherita Russo [2000] "Competenze, innovazione e sviluppo locale" pp 31.
- 298 Gian Paolo Caselli e Tommaso Minerva [2000] "The Transition Process in Russia and China and the Ising Model" pp 30.
- 299 Gisella Facchinetti, Giovanni Mastroleo e Sergio Paba [2000] "A Fuzzy Approach to the Empirical Identification of Industrial Districts" pp 7.
- 300 Tommaso Minerva, Irene Poli and Sebastiano Brusco [2000] "A Cellular Automaton as a Model to Study the Dynamics of an Industrial District" pp 6.
- 301 Gisella Facchinetti [2000] "Il problema della misurazione del rischio di credito: una rassegna critica di metodologie" pp 13.
- 302 Marco Mazzoli [2000] "Investments and Financial Structure with Imperfect Financial Markets: an Intertemporal Discrete-Time Framework" pp 13.
- 303 Giuseppe Marotta [2000] "Il credito commerciale in Italia: evidenza su dati d'impresa" pp 29.
- 304 Marco Mazzoli [2000] "Credit Channel and Industrial Firms' Market power" pp 15.
- 305 Gisella Facchinetti e Giovanni Mastroleo [2000] "The Mamdani and the γ -operator in a Fuzzy Logic Control System" pp 17.
- 306 Giovanni Solinas e Giovanni Mastroleo [2000] "Benchmarking, certificazione della qualità e piccole imprese. La sperimentazione di un modello europeo nelle piccole imprese in Emilia Romagna" pp 45.
- 307 Margherita Russo, Giorgio Allari, Silvano Bertini, Paolo Bonaretti, Elio De Leo, Giuseppe Fiorani and Gianni Rinaldini [2000] "The Challenges for the Next Debate: Notes for a Debate on the Development of the Emilia-Romagna Region" pp 27.
- 308 Giovanni Mastroleo [2000] "L'integrazione dell'indagine statistica con l'approccio fuzzy nel controllo di efficacia: il monitoraggio sugli obiettivi raggiunti nell'ambito di un P.O.M." pp 24.
- 309 Gisella Facchinetti, Stefano Bordon e Giovanni Mastroleo [2000] "Bank Creditworthiness Using Fuzzy Systems: A Comparison with a Classical Analysis Approach" pp 13.
- 310 Margherita Russo e Raffaele Giardino [2000] "Struttura e cambiamento nelle relazioni tra le imprese meccaniche. I. La popolazione di imprese meccaniche della provincia di Modena: procedure impiegate per integrare le informazioni amministrative del Registro Imprese e dell'Inps" pp 32.
- 311 Tommaso Minerva e Sandra Paterlini [2000] "Tecniche Computazionali per la Statistica, l'Economia e la Finanza. *Materiale Didattico a Supporto del Corso di Statistica Computazionale*" pp 52.
- 312 Costanza Torricelli e Silvia Muzzioli [2000] "Combining the Theory of Evidence with Fuzzy Sets for Binomial Option Pricing" pp.20.
- 313 Marco Mazzoli e Roberto Negrini [2000] "Strumenti finanziari negoziabili e incentivo-compatibili per le imprese cooperative. *Alcune considerazioni teoriche e di policy*" pp. 32.
- 314 Giacomo Galeotti e Tommaso Minerva [2000] "Algoritmi ibridi per l'ottimizzazione di un Portafoglio Azionario. *Simulazione stocastica filtrata mediante wavelet decomposition*" pp.33.
- 315 Alberto Roverato [2000] "Hyper Inverse Wishart Distribution for Non-Decomposable Graphs and its Application to Bayesian Inference for Gaussian Graphical Models" pp. 29.
- 316 Carlo Alberto Magni [2000] "Scomposizione di sovrapprofitti: Economic Value Added e valore aggiunto sistemico" pp. 25
- 317 Carlo Alberto Magni [2000] "Decomposition of a Certain Cash Flow Stream: Systemic Value Added and Net Final Value" pp. 30.
- 318 Carlo Alberto Magni [2000] "Systemic Value Added, Residual Income and Decomposition of a Cash Flow Stream" pp. 27.
- 319 Gisella Facchinetti e Giovanni Mastroleo [2000] "La valutazione del rischio di frode nel ramo assicurativo R.C. auto: una proposta in logica Fuzzy" pp. 16.
- 320 Gian Paolo Caselli e Gabriele Pastrello [2000] "Eltsin: Dimissioni o Licenziamento?" pp. 18.
- 321 Gisella Facchinetti, Carlo Alberto Magni e Giovanni Mastroleo [2000] "Real Options: a Fuzzy Approach for Strategic Investments" pp. 44.
- 322 Stefano Bordon [2000] "Applicazione Fuzzy per la determinazione del premio assicurativo" pp. 35.
- 323 Gabriele Pastrello [2000] "Una distrazione di Marx" pp. 17
- 324 Marco Mazzoli [2000] "Canale creditizio, struttura di mercato, modifiche istituzionali e meccanismo di trasmissione della politica monetaria" pp. 18.
- 325 Paola Bertolini e Luca Riazzi [2000] "L'applicabilità dello strumento futures al Mediterraneo riflessioni su un fallimento" pp 28.
- 326 Enrico Giovanetti [2000] "Istituzioni e costi transattivi: l'impatto della regolazione dell'offerta nelle filiere agroindustriali" pp 26.
- 327 Gian Paolo Caselli e Marta Rosso [2000] "La moneta elettronica: aspetti di regolamentazione finanziaria".
- 328 Barbara Pistori e Chiara Strozzi [2000] "Labor Productivity and Labor Cost Dynamics in Italy: the Role of Wage Bargaining" pp. 23.
- 329 Carlo Alberto Magni [2000] "Valore Aggiunto Sistemico: un'alternativa all'EVA quale indice di sovrapprofito periodale" pp.11.
- 330 Carlo Alberto Magni [2000] "On Decomposing Net Final Values: Systemic Value Added and Shadow Project" pp. 26.
- 331 Massimo Baldini [2000] "MAPP98: un Modello di Analisi delle Politiche Pubbliche" pp. 24.
- 332 Paolo Bosi, Massimo Baldini, Maria Cecilia Guerra e Paolo Silvestri [2000] "La scelta tra Ici e Addizionale all'Irpef nella Politica tributaria locale: aspetti distributivi" pp. 27.
- 333 Marina Murat e Sergio Paba [2000] "Flussi migratori e modelli di sviluppo industriale. *L'esperienza italiana dal dopoguerra agli anni novanta*" pp. 32.
- 334 Marco Mazzoli e Roberto Negrini [2000] "Incentive-Compatible Financial Instruments for Co-Operative Firms: a Few Policy Considerations" pp. 27.
- 335 Massimo Baldini e Paolo Bosi [2000] "Riforme trasparenti e proposte opache" pp. 10.
- 336 Paolo Bosi [2000] "La selettività nelle politiche sociali in Italia: riflessioni sull'esperienza dell'Isce" pp. n. 16.
- 337 Massimo Baldini, Paolo Bosi e Stefano Toso [2000] "Targeting Welfare in Italy: Old Problems and Perspectives of Reform" pp. n. 21.
- 338 Tindara Addabbo e Massimo Baldini [2000] "The Gender Impact of Workfare Policies in Italy and the Effect of Unpaid Work" pp. n. 15.
- 339 Gian Paolo Caselli e Grid Thoma [2000] "La storia economica albanese 1912-1939 e lo stabilirsi dell'egemonia italiana" pp. 46.