



Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Modelli Intelligenti per la gestione integrata dell'area costiera che si estende dallo Stagnone di Marsala a Marinella di Selinunte

F. Gennaro, S. Vitabile

Rapporto Tecnico N.:05
RT-ICAR-PA-05-05

Luglio 2005



Consiglio Nazionale delle Ricerche, Istituto di Calcolo e Reti ad Alte Prestazioni (ICAR)

– Sede di Cosenza, Via P. Bucci 41C, 87036 Rende, Italy, URL: www.icar.cnr.it

– Sezione di Napoli, Via P. Castellino 111, 80131 Napoli, URL: www.na.icar.cnr.it

– Sezione di Palermo, Viale delle Scienze, 90128 Palermo, URL: www.pa.icar.cnr.it



Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Modelli Intelligenti per la gestione integrata dell'area costiera che si estende dallo Stagnone di Marsala a Marinella di Selinunte

F. Gennaro¹, S. Vitabile^{1,2}

**Rapporto Tecnico N.: 05
RT-ICAR-PA-05-05**

**Data:
Luglio 2005**

¹ Istituto di Calcolo e Reti ad Alte Prestazioni, ICAR-CNR, Sezione di Palermo

² Università degli Studi di Palermo - Dipartimento di Biotecnologie Mediche e Medicina Legale - - Sez. di Radiologia

I rapporti tecnici dell'ICAR-CNR sono pubblicati dall'Istituto di Calcolo e Reti ad Alte Prestazioni del Consiglio Nazionale delle Ricerche. Tali rapporti, approntati sotto l'esclusiva responsabilità scientifica degli autori, descrivono attività di ricerca del personale e dei collaboratori dell'ICAR, in alcuni casi in un formato preliminare prima della pubblicazione definitiva in altra sede.

Indice

Introduzione	3
1 Reti neurali	6
1.1 Serie storica	7
1.2 Preprocessing dei dati	8
1.3 Modelli neurali sviluppati	9
1.3.1 Architettura di Elman	11
1.3.2 Architettura MLP	13
1.4 Risultati ottenuti	14
1.5 Analisi di sensibilità	16
2 Reti di Bayes	18
2.1 Il Modello bayesiano sviluppato	19
Riferimenti bibliografici	26

Introduzione

Con il termine Analisi Intelligente dei Dati (Intelligent Data Analysis - IDA) si identificano tutte le tecniche che sfruttano tecniche mutuata dall'Intelligenza Artificiale per effettuare l'analisi dei dati. L'importanza della Intelligent Data Analysis è confermata dal fatto che il mondo moderno è guidato dai dati. Noi siamo circondati dai dati, numerici e altri, che devono essere analizzati, processati o convertiti per fornire informazioni, risposte, al fine di incrementare la conoscenza di sistemi complessi e creare sistemi di supporto alle decisioni nei campi più svariati (finanziari, ambientali, industriali).

L'applicazione delle tecniche di Analisi Intelligente dei dati nel del progetto MOGEIFACO ha un duplice obiettivo. Da un lato si vogliono sviluppare modelli per la previsione di grandezze con strumenti, che partendo da serie storiche relative alla grandezza oggetto di indagine, siano in grado di prevederne il comportamento futuro. Un esempio tipico nella banca dati MOGEIFACO è costituito dalle serie climatiche. Dall'altro si vogliono sviluppare modelli mirati alla comprensione delle relazioni tra i dati disponibili con l'obiettivo di rispondere a quesiti del tipo: come l'incremento dell'indice di densità edilizia influenza l'indice di sostenibilità della zona fisiografica in questione?

È immediato comprendere che l'informazione aggiuntiva fornita dai modelli sviluppati può avere notevoli risvolti di carattere economico e sociale.

In funzione al contesto di riferimento e al tipo di informazione ricercata, l'applicazione e l'ottimizzazione delle tecniche di IDA richiede le seguenti fasi :

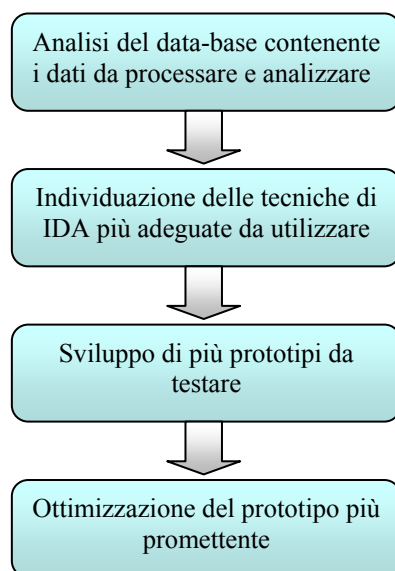


Figura 1- Fasi di sviluppo

La banca dati realizzata per il progetto MOGEIFACO (si veda la Figura 2), contiene una grande quantità di dati eterogenei che si riferiscono a molteplici aspetti. I dati a disposizione variano per:

- tipologia e natura (dati meteorologici, dati socio-economici, dati biologici, dati ambientali, dati di pescosità, ecc.);
- localizzazione e risoluzione territoriale (i dati biologici hanno un livello di dettaglio territoriale notevolmente maggiore rispetto ai dati socio-economici);
- risoluzione e intervalli temporali di rilevamento (i dati meteorologici sono rilevati con cadenza giornaliera mentre i dati socio-economici sono rilevati con cadenza stagionale o annuale, per i dati meteorologici sono disponibili serie decennali mentre per i dati socio-economici sono disponibili pochi rilevamenti).

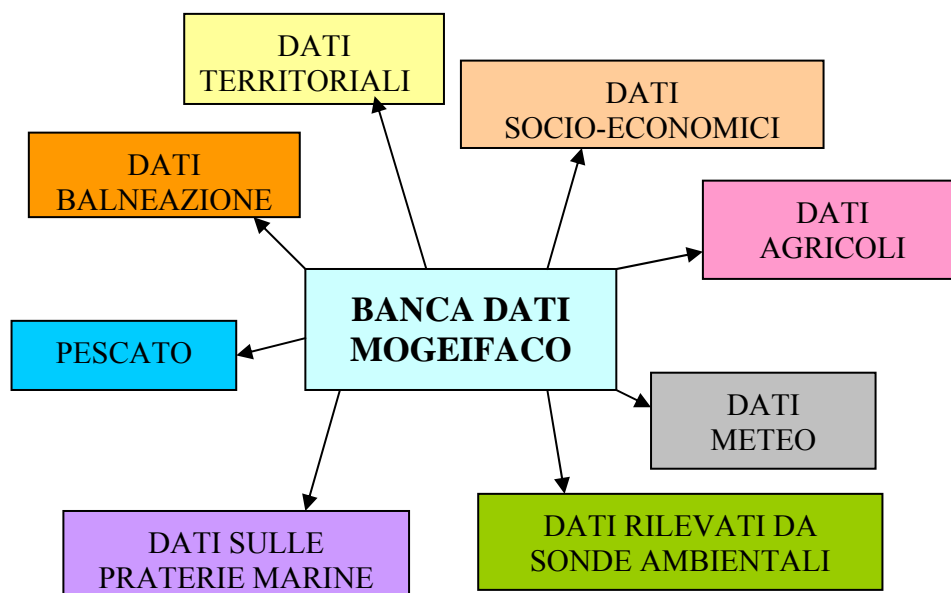


Figura 2- Banca dati MOGEIFACO in funzione alla tipologia e natura

I modelli intelligenti sviluppati per il progetto si basano sulle Reti Neurali Artificiali (RNA) e sulle Reti Bayesiane (RB). Sia le RNA che le RB sono delle architetture di tipo connessionistico caratterizzate da nodi e archi.

Nelle RNA non è necessario fornire alcuna informazione supplementare sul sistema da modellare: un insieme di addestramento adatto a rappresentare la dinamica del sistema è sufficiente per la costruzione delle relazioni ingresso-uscita del dominio applicativo. Il modo in cui conoscenza viene modellata dalla rete neurale non è esplicito per cui, in questo tipo di reti, risulta difficile trasferire sul modello la conoscenza di un esperto del sistema. Con riferimento alla piramide dell'informazione mostrata in Figura 3, esse interagiscono a basso livello di astrazione con i dati grezzi da elaborare.

Nel dominio applicativo di interesse, analizzando le serie temporali disponibili, si è scelto come caso studio la previsione della temperatura per il giorno successivo. In tale ambito, infatti, la serie temporale è composta da un numero significativo di rilevazioni.

In particolare sono stati realizzati dei modelli basati su le reti neurali di Elman (reti supportati da memoria a breve termine) che verranno descritte nel successivo paragrafo.

Le reti bayesiane, note anche come reti probabilistiche e reti di credenze, sono modelli adatti a rappresentarne la conoscenza in un dominio incerto. Ogni nodo rappresenta una variabile del sistema il cui valore è espresso in termini probabilistici. A sua volta (tramite archi) il comportamento di una variabile può essere condizionata dal comportamento delle altre. Nelle RB risulta semplice trasferire sul modello la conoscenza di un esperto del sistema, con la conseguenza di raggiungere buoni risultati anche con insieme di addestramento ridotto.

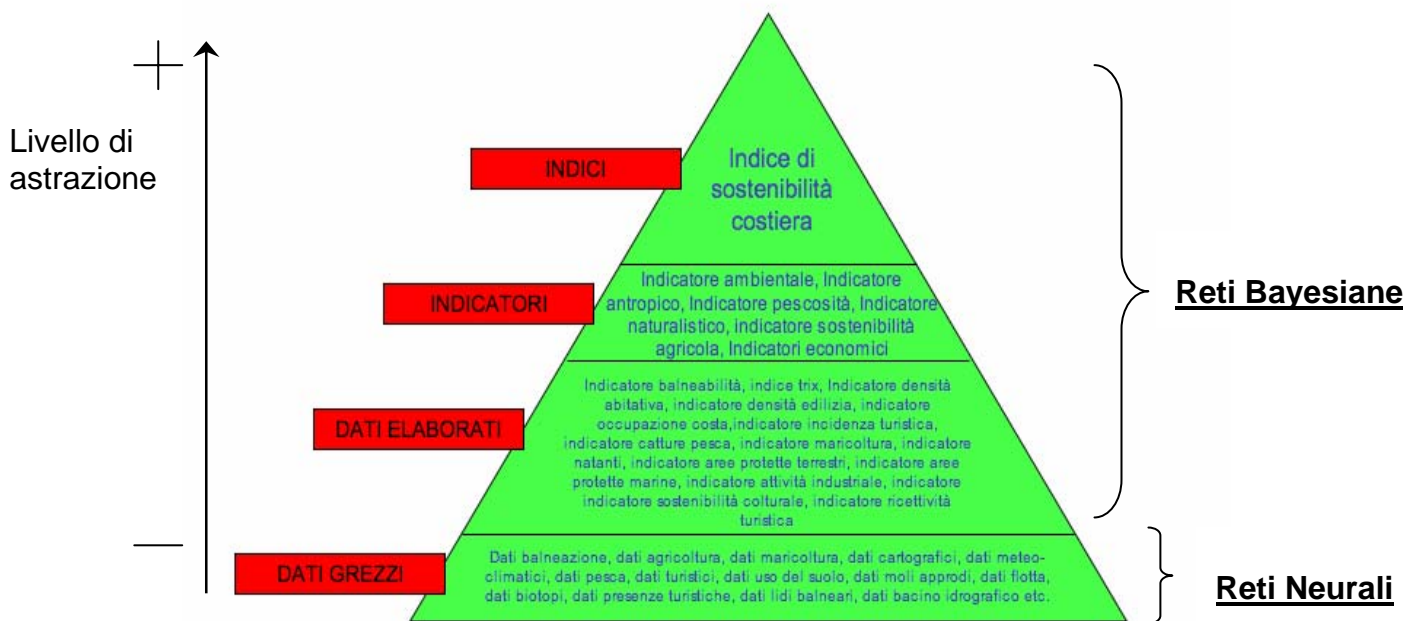


Figura 3 – Piramide dell'informazione

Di seguito saranno mostrati le caratteristiche e i risultati dei modelli intelligenti sviluppati nell'ambito del progetto: nella prima parte saranno mostrate e confrontate le architetture neurali sviluppate, sarà inoltre mostrato uno studio di sensibilità degli ingressi sulle prestazioni delle reti neurali; nella seconda parte sarà mostrata l'architettura bayesiana sviluppata e alcuni esempi di inferenza che è possibile fare con tale strumento.

1 Reti neurali

Le RNA sono un modello computazionale costituito da una rete di semplici unità di elaborazione non lineari (neuroni) dotate di una propria memoria che comunicano tra loro attraverso delle connessioni pesate su cui vengono scambiati dati. I neuroni lavorano in parallelo e quindi le RNA sono sistemi di elaborazione intrinsecamente paralleli con notevoli doti di resistenza all'errore (fault tolerant).

Il tipico approccio delle reti neurali è quello basato sul paradigma ingresso-uscita di tipo statico, dove il vettore di output $o(t+1)$ al tempo $t+1$ è diretta conseguenza del solo vettore di input $x(t)$ al tempo t , e quindi il vettore di output è prodotto da una propagazione in avanti dei segnali di ingresso che stimolano l'architettura neurale. Questo tipo di approccio è utile a risolvere problemi di tipo statici, come quelli di classificazione.

Nel caso di analisi di serie temporali per poter determinare la corretta uscita al tempo $t+1$, spesso risulta utile conoscere oltre che l'ingresso $x(t)$ anche l'evoluzione degli ingressi o degli stati che assume la rete. Per questo genere di problemi sono state sviluppate le reti neurali ricorrenti (RNR) che utilizzano un paradigma ingresso-uscita di tipo dinamico. Nelle RNR tramite delle connessioni ricorsive parte degli input o degli stati precedenti al tempo t concorrono alla determinazione del vettore di output al tempo $t+1$.

La scelta dell'architettura più opportuna per affrontare un determinato problema è sempre molto difficile, non esiste una metodologia universalmente accettata e spesso si procede per tentativi. Per quanto concerne il progetto MOGEIFACO, le RNA sono state utilizzate per prevedere la temperatura. Le informazioni meteorologiche contenute nelle serie storiche delle zone fisiografiche, riguardano i seguenti parametri giornalieri: temperatura massima, temperatura minima, umidità massima, umidità minima, direzione vento, velocità del vento.

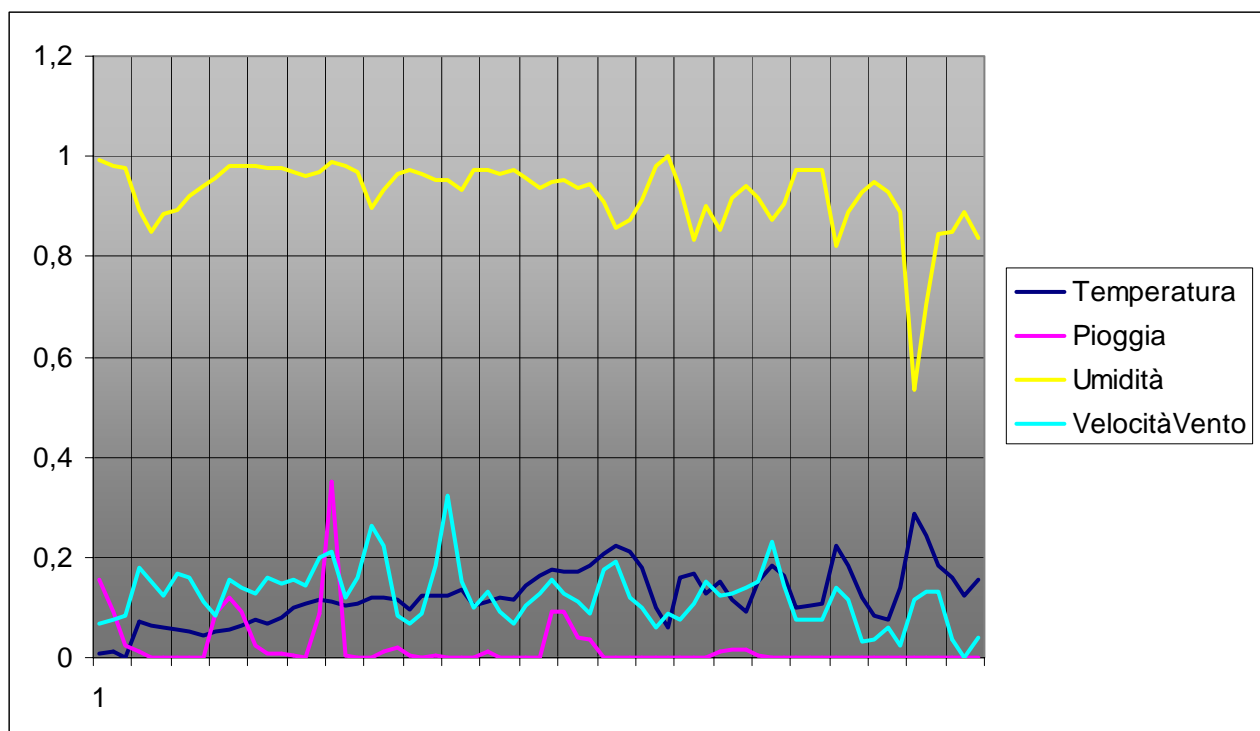


Figura 4 - Andamento delle grandezze normalizzate prese in esame

In Figura 4 viene mostrato un tipico andamento delle grandezze (normalizzate) prese in esame. Dal grafico risulta evidente che tali grandezze meteorologiche risultano essere reciprocamente legate. Scopo delle RNA è comprendere tali legami per poter predire il comportamento futuro di una o più parametri.

1.1 Serie storica

La serie temporale oggetto del caso studio si riferisce ai dati meteorologici della zona di San Nicola (Mazara del Vallo) nel periodo che va dal 01/01/2002 al 24/07/2003 per un totale di 1.070 giorni. Le informazioni meteorologiche contenute nella serie temporali riguardano i seguenti parametri con frequenza di rivelazione giornaliera (Tabella 1) : temperatura massima, temperatura minima, umidità massima, umidità minima, piovosità; e i seguenti parametri con rivelazione oraria: direzione vento, velocità del vento.

La scelta del periodo è stata condizionata dal desiderio di inserire anche i dati anemometrici, infatti tali dati presentano un periodo di osservazione molto ridotto rispetto gli altri dati ambientali quali temperatura, umidità, piovosità.

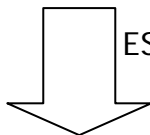
In Figura 5 viene mostrato il metodo utilizzato per ricavare i patterns che costituiscono l'insieme di addestramento e l'insieme di test su cui verrà effettuata la previsione. Si può notare che i dati con rivelazione giornaliera sono stati estesi, considerando le coppie [(temperatura max, umidità min), (temperatura min, umidità max)] riferite a due momenti della stessa giornata corrispondenti all'incirca alle ore 4 e alle ore 13. Con questa approssimazione plausibile è possibile ottenere una frequenza temporale di rilevazione doppia rispetto alla serie originale. I dati anemometrici sono rilevati con cadenza oraria e per renderli congrui con gli altri ingressi, vengono campionati rispettivamente alle ore 4 e alle ore 13.

Tabella 1 – Parametri meteorologici di ingresso alla rete neurale

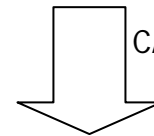
Zona di San Nicola, dal 01/01/2002 al 24/07/2003				
	Input della rete	Valore max	Valore min	Unità di misura
Rilevazione giornaliera	Temperatura	40	5	C°
	Umidità	95	5	%
	Pioggia	60	0	mm/giorno
Rilevazione oraria	Velocità del vento	15,1	0	m/sec
	Classe del vento	359	0	gradi polari

Giorno	Temp. Max	Temp. Min	Umidità Min	Umidità Max	Piovosità
03/01/2002	14	7	95	70	8
04/01/2002	10	4	95	60	0
05/01/2002	12	4	95	65	0,2
....

Dati anemometrici
rilevati con
cadenza oraria



ESTENSIONE



CAMPIONAMENTO

Giorno	Temperatura	Umidità	Piovosità	Direzione vento	Velocità Vento
03/01/2002 1°	7	70	0	100	2
03/01/2002 2°	17	40	0	95	0,4
04/01/2002 1°	5	70	0	11	4,7
04/01/2002 2°	18	50	0	25	5,6
05/01/2002 1°	8	70	3,6	329	1,7
05/01/2002 2°	17	55	3,6	3	3,9
....

Figura 5 – Costruzione dei campioni dell’insieme di addestramento e di test

Con questa costruzione del data-set di ingresso alla RNA abbiamo 2140 campioni riferiti a 1070 giorni da utilizzare per eseguire l’addestramento e il test della rete neurale.

1.2 Preprocessing dei dati

Una volta che i dati grezzi sono stati raccolti è necessario convertirli in un formato più opportuno per l’addestramento della rete ed, inoltre, risulta necessario fare un controllo della qualità dei dati.

Controllo della qualità dei dati

Il controllo della qualità dei dati può rivelare dati inaccettabili nascosti che, se non fossero eliminati, produrrebbero un degrado delle prestazioni. Il modo più semplice per realizzare un controllo della qualità è verificare se i dati appartengono ad un determinato intervallo (*range check*) tipico di riferimento. Se si riscontrano valori al di fuori di tale range, è lecito sospettare che si tratti di dati errati. Un altro tipico problema riscontrato nelle serie temporali sono i dati mancanti dovuti a probabili guasti temporanei dei sensori. In questo caso si cerca di ricavare i valori mancanti procedendo per interpolazione.

Partizionamento dei dati

Il partizionamento dei dati indica la divisione dell'insieme dei dati originali in due parti e precisamente: insieme di addestramento e insieme di test. Per definizione durante la fase di addestramento i pesi della rete vengono modificati allo scopo di minimizzare la funzione di errore legata alla differenza tra uscita reale e uscita desiderata, mentre l'insieme di test viene usato per esaminare le performance finali della rete addestrata. L'insieme di addestramento deve contenere una quantità sufficiente di esempi (coppie ingresso/uscita per reti supervisionate) allo scopo di conferire alla rete neurale una buona capacità di generalizzazione.

Il partizionamento dei dati per le simulazioni è stato fatto nel seguente modo: l'80% dei campioni totali (1712 esempi) è stato utilizzato per il train e il 20% per il test (428 esempi).

È quasi sempre di indubbio vantaggio eseguire delle operazioni di normalizzare sugli ingressi della rete che, sebbene non siano strettamente necessarie, producono una convergenza più rapida e diminuiscono il rischio di minimi locali [4].

La normalizzazione è stata eseguita normalizzando tutte le grandezze di ingresso e uscita nel range [0,1] in base alla formula:

$$Z' = \frac{Z - \max}{(\max - \min)}$$

dove Z è il valore da normalizzare, max e min si riferiscono ai valori massimi e minimi dell'insieme di riferimento e Z' il valore normalizzato.

1.3 Modelli neurali sviluppati

A partire del 1989, Elman [5] si occupò della realizzazione di architetture neurali in grado di prevedere l'andamento futuro in base alla conoscenza storica di una o più grandezze. Egli individuò due possibili soluzioni: una rappresentazione esplicita ed una implicita del tempo. Nel primo caso si interpretano in maniera parallela eventi intrinsecamente seriali come quelli che avvengono in istanti successivi. In questo modo si dà una rappresentazione spaziale del tempo: il primo evento temporale viene associato con il primo elemento del vettore di ingresso, il secondo evento temporale con il secondo elemento del vettore di ingresso e così via fino ad un numero finito di eventi passati. L'intero vettore in ingresso viene elaborato in parallelo dalla rete (Figura 6): si può affermare che le reti MLP (*Multi Layers Perceptron*) appartengono a questa tipologia di reti.

Questa soluzione, presenta alcuni svantaggi. Innanzitutto deve essere presente un buffer in grado di conservare un numero finito di valori di ingresso in istanti precedenti fissando la finestra temporale da dare in ingresso alla rete, ed inoltre questa soluzione non ha un equivalente nel modello biologico del cervello.

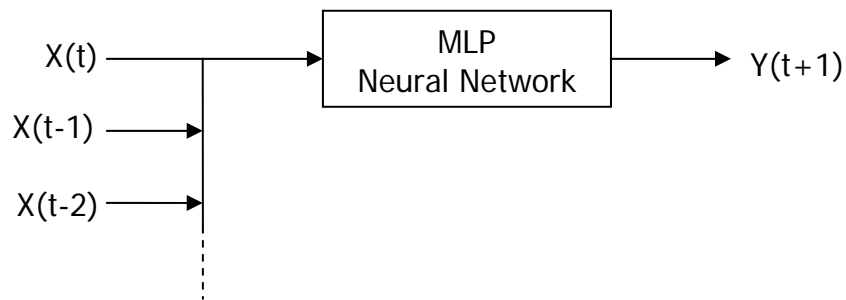


Figura 6 - MLP con memoria a breve termine statica

Un approccio al problema, migliore sotto diversi punti di vista, è quello di rappresentare il tempo in maniera implicita, cioè attraverso il paradigma ingresso-stato-uscita e non come “dimensione” ulteriore dell’input. In ogni istante tutte le informazioni passate vengono *riassunte* nel valore dello stato. Elman individuò nella ricorsione (Figura 7) lo strumento per dare alle RNA le capacità dinamiche necessarie a rappresentare adeguatamente il tempo.

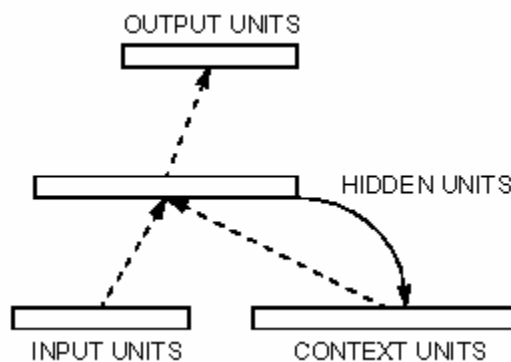


Figura 7 - Architettura neurale proposta da Elman

La rete ricorrente di Elman, nella sua versione base, può pensarsi come variazione di una normale rete feed-forward aggiungendo un feedback *hidden-hidden* attraverso un nuovo strato, detto *context units*. Le connessioni *input-hidden*, *context-hidden* e *hidden-output* sono dotate di peso adattivo, mentre le connessioni *hidden-context*, hanno peso fisso, in quanto hanno la sola funzione di copia di una frazione del valore di attivazione. Solitamente tali valori e vengono posti a uno.

Sia l’architettura di Elman sia l’architettura MLP sono state sviluppate nel tentativo di trovare un modello che possa rispondere in maniera soddisfacente al problema della previsione delle serie climatiche. Per lo sviluppo e la simulazione di tali architetture è stato utilizzato il simulatore SNNS (Stuttgart Neural Network Simulator)¹, operante in ambiente Linux [13].

¹ Sviluppato presso l’Institute for Parallel and Distributed High Performance Systems (IPVR) dell’università di Stoccarda, con lo scopo di realizzare un ambiente di simulazione efficiente e flessibile per la ricerca sulle RNA e le loro applicazioni.

In riferimento alla serie temporale (Figura 5), si vuole, quindi mettere a punto una architettura neurale in grado di prevedere la temperatura nello step successivo a quello noto. Dunque se l'ultimo dato presente nella serie è la temperatura massima, si cerca di predire il valore della temperatura minima, altrimenti se l'ultimo dato presente nella serie è la temperatura minima, si cerca di predire il valore della temperatura massima.

1.3.1 Architettura di Elman

Seguendo il modello di Elman è stata sviluppata la seguente architettura, definita dalla Tabella 2. Le funzioni di attivazione sono lineari per gli strati di *input*, *context* e *output*, in quanto tali strati funzionano solitamente come buffer; mentre lo strato di *hidden* è stata scelta la funzione sigmoideale.

Tabella 2 – Architettura neurale di Elman

Strato	N° di Neuroni	Funzione di attivazione
Input	5	Act_IdentityPlusBias
Hidden	20	Act_Logistic
Context	20	Act_Identity
Output	1	Act_IdentityPlusBias

I cinque ingressi della rete sono (Tabella 1) : Temperatura, Umidità, Pioggia, Velocità del vento, Classe del vento. L'unica uscita è data dal valore della temperatura al tempo $t+1$.

Inserendo gli ingressi al tempo t , senza fornire nessuna informazione sui valori precedenti, la rete tenta di predire il comportamento della temperatura al tempo $t+1$.

In Figura 8 è mostrata l'architettura di Elman sviluppata.

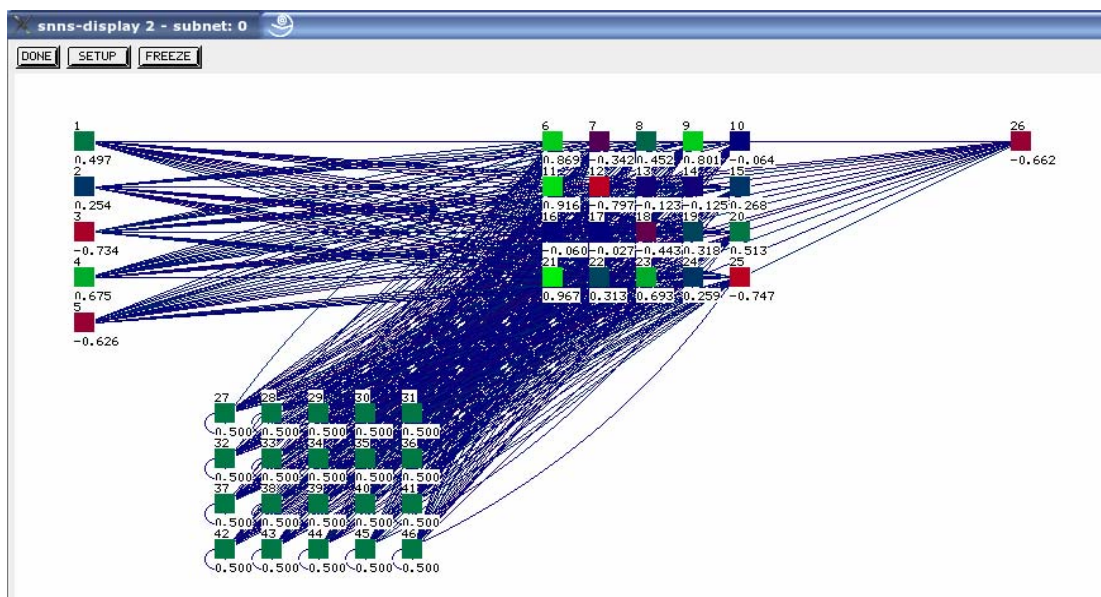


Figura 8 - Architettura di Elman sviluppata

Per la minimizzazione della funzione di errore è stato utilizzato l’algoritmo RPROP (Resilient Backpropagation). Tale algoritmo di tipo adattativo mostra particolari doti di robustezza e velocità di convergenza dell’errore[1][2]. Il principio base della RPROP è quello di eliminare l’influenza negativa che ha la dimensione della derivata parziale sul *weight step* (passo utilizzato per l’aggiornamento dei pesi). La dimensione per l’aggiornamento dei pesi Δw_{ji} (1) è esclusivamente decisa dal coefficiente $\Delta_{ij}(t)$ (2) chiamato “update-value”. Ogni volta che la derivata parziale del peso corrispondente w_{ji} cambia segno indica che l’ultimo aggiornamento era troppo grande, e che l’algoritmo è saltato sopra un minimo locale. In questo modo l’RPROP usa uno step adattativo per l’aggiornamento dei pesi.

$$\Delta w_{ji}(t) = -\Delta_{ji}(t) * \text{sign}(g_{ji}(t)) \quad (1)$$

dove Δ_{ji} è calcolato come:

$$\Delta_{ji} = \begin{cases} 1.2 * \Delta_{ji}(t-1) & \text{if } g_{ji}(t) \cdot g_{ji}(t-1) > 0 \\ 0.5 * \Delta_{ji}(t-1) & \text{if } g_{ji}(t) \cdot g_{ji}(t-1) < 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

dove $g(t)$ è il gradiente.

L’addestramento è stato eseguito in 200 epoche, con valori maggiori vi è il rischio di cadere in situazioni di overfitting (la rete si specializza eccessivamente sui patterns di addestramento) .

L’algoritmo RPROP è stato parametrizzato con Δ_0 pari 0.2 e Δ_{\max} pari a 50.

In Figura 9 è mostrato l’andamento dell’addestramento con i parametri selezionati.

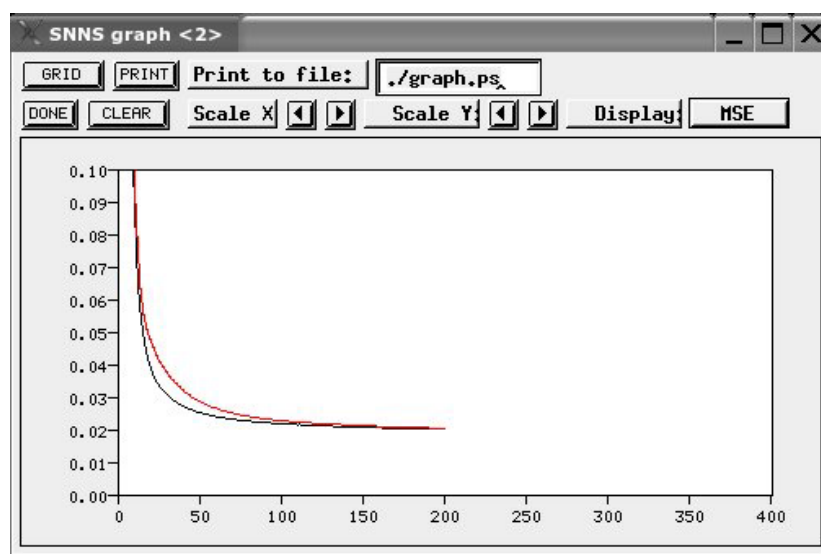


Figura 9 - Addestramento della rete di Elman sviluppata

1.3.2 Architettura MLP

L'architettura neurale MLP implementata è definita dalla Tabella 3. Le funzioni di attivazione sono lineari per gli strati di *input*, *output*, in quanto tali strati funzionano solitamente come buffer; mentre di tipo sigmoideale per lo strato di hidden.

Tabella 3– Architettura neurale MLP

Strato	N° di Neuroni	Funzione di attivazione
Input	15	Act_IdentityPlusBias
Hidden	40	Act_Logistic
Output	1	Act_IdentityPlusBias

La rete presenta 15 ingressi. Essi si riferiscono ai valori di $X(t)$, $X(t-1)$ e $X(t-2)$ dove $X(t)$ è un vettore costituito da $X(t)=[\text{Temperatura}(t), \text{Umidità}(t), \text{Pioggia}(t), \text{Velocità del vento}(t), \text{Classe del vento}(t)]$. L'unica uscita è data dal valore della temperatura al tempo $t+1$.

Inserendo gli ingressi al tempo t , $t-1$ e $t-2$ la rete tenta di predire il comportamento della temperatura al tempo $t+1$.

In Figura 10 è mostrata l'architettura Mlp sviluppata.

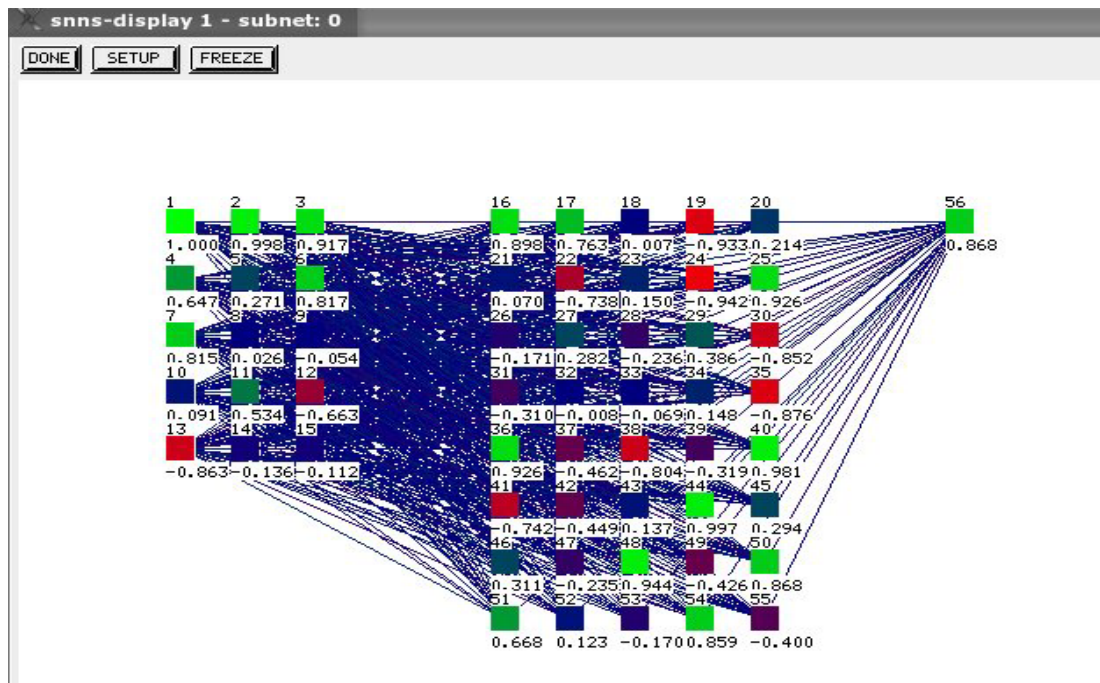


Figura 10 - Architettura Mlp sviluppata

L'addestramento della rete è stato eseguito utilizzando il ben noto algoritmo Backpropagation. Tale algoritmo introdotto da E. Rumelhart e J.L. McClelland [12] risulta ampiamente usato per

l'addestramento di reti neurali di tipo feed-forward. Tale algoritmo è detto *online* perché l'aggiornamento dei pesi della rete avviene per ogni pattern (campione) utilizzato per il *train*.

Il valore dell'aggiornamento dei pesi Δw_{ji}

(3) è espresso in funzione di η, δ_j e o_j dove:

- η è un fattore di apprendimento costante;
- δ_j è la funzione errore (4), data dalla differenza tra il valore reale di uscita e il valore desiderato;
- o_j uscita della unità j ;

$$\Delta w_{ij} = \eta \delta_j o_i \quad (3)$$

$$\delta_j = \begin{cases} f'_j(\text{net}_j)(t_j - o_j) & \text{Se l'unità } j \text{ è di } output \\ f'_j(\text{net}_j) \sum_k \delta_k w_{jk} & \text{Se l'unità } j \text{ è di } hidden \end{cases} \quad (4)$$

L'addestramento è stato eseguito in 400 epoche, con valori maggiori vi è il rischio di cadere in situazioni di overfitting. L'algoritmo Backpropagation è stato parametrizzato con η pari 0.2, e Δ_{\max} pari a 0.1.

1.4 Risultati ottenuti

Il confronto tra i modelli studiati è stato effettuato mediante degli indici. Gli indici utilizzati sono il MAE (Errore Assoluto Medio) che fornisce una indicazione quantitativa di quanto le previsioni si avvicinino ai valori reali, l'MSE (Errore Quadratico Medio) che fornisce un'idea di quanto il modello rispecchi i dati forniti, e il $Q_p = H^*(I+O)$ che corrisponde al costo computazionale nella fase di funzionamento [15].

Tabella 4 – Valori di MAE e MSE

Strato	$Q_p = H^*(I+O)$	MAE	MSE
Architettura di Elman	420	1,29	0,025
Architettura MLP	640	1,40	0,04

Osservando il valore del MAE nella Tabella 4, si nota che l'architettura neurale di Elman fornisce una previsione più precisa pur avendo un costo computazionale minore.

Nella Figura 11 viene mostrato il grafico dell'andamento del valore effettivo e del valore previsto della temperatura nell'architettura di Elman.

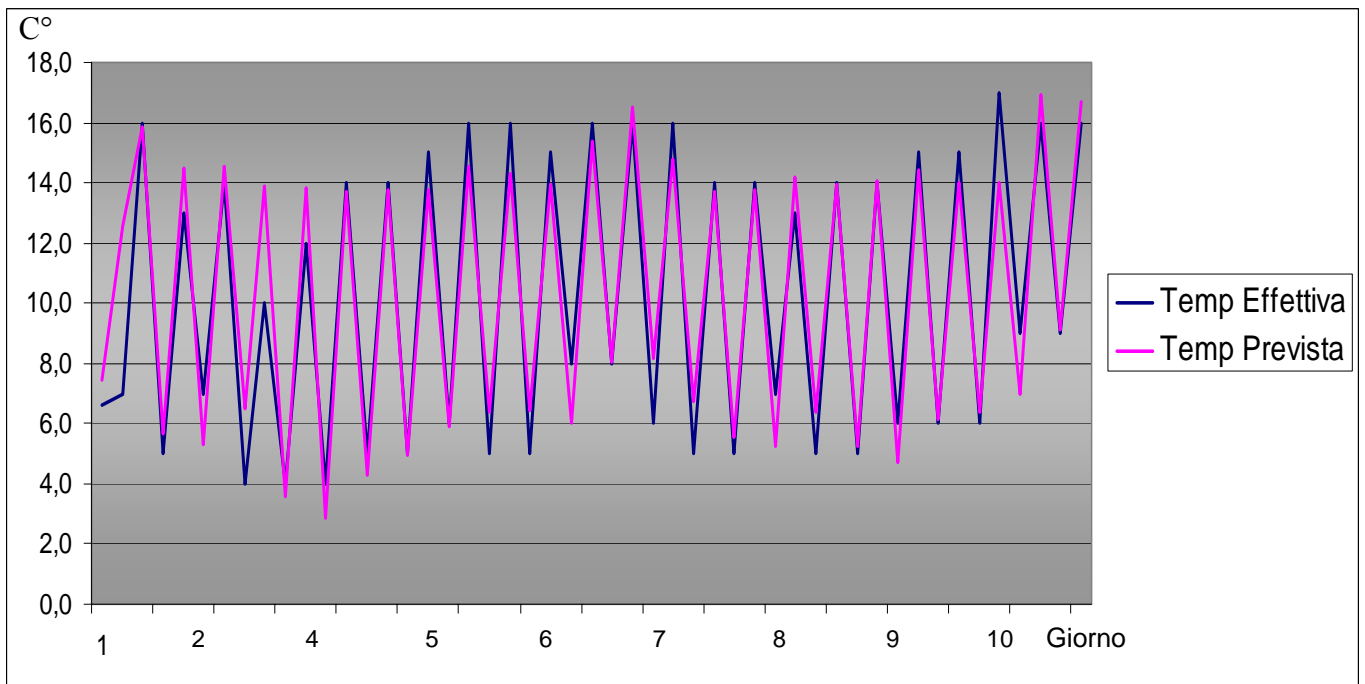


Figura 11- Andamento del valore effettivo e del valore previsto della temperatura nell'architettura di Elman

In Figura 12 (sempre riferito all'architettura di Elman) viene mostrato come si distribuisce l'errore assoluto, si osserva che nel 67 % dei casi l'errore si mantiene al di sotto di 1,5C° e solo nel 7% dei casi l'errore supera di 3 C°

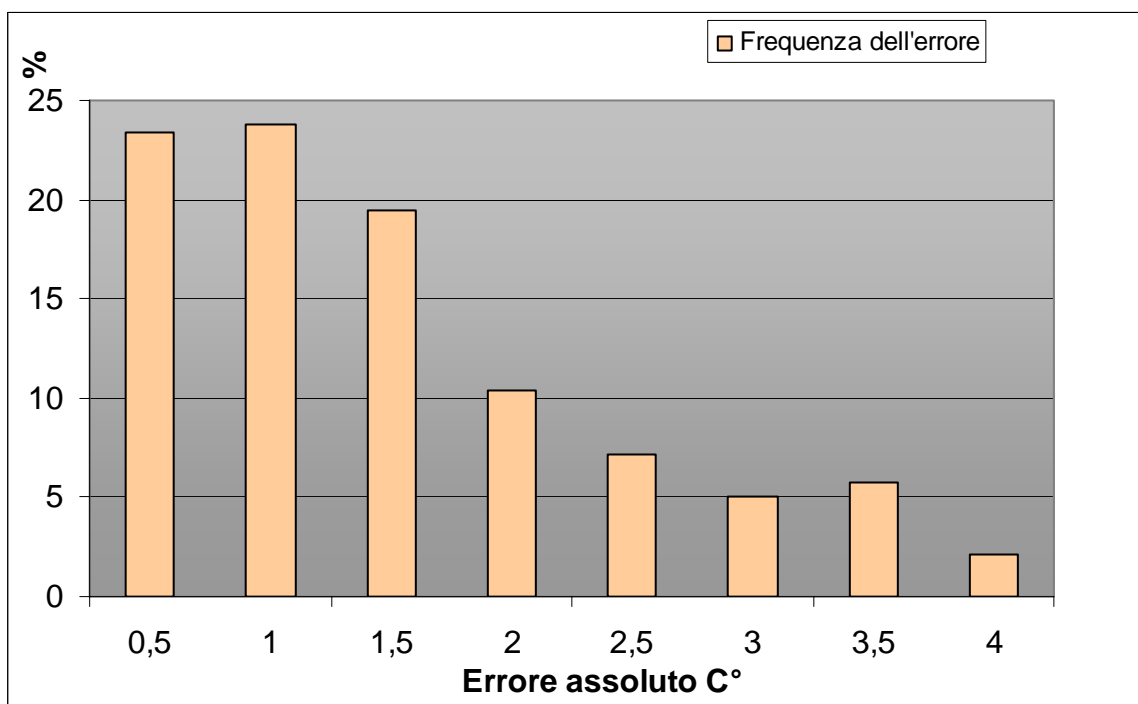


Figura 12 – Frequenza dell'errore assoluto

1.5 Analisi di sensibilità

Le architetture neurali spesso vengono chiamate *black-box* perché risulta difficile capirne la dinamica interna, e dunque cogliere le relazioni che lega le variabili identificate come input del problema e quelle identificate come output.

Un problema interessante è capire come la modifica del valore di un ingresso si ripercuota sulla uscita. Infatti risulta logico pensare che ogni variabile di ingresso influenzi in modo differente l'uscita.

Per affrontare tale problematica sono state proposte varie soluzioni e altrettanti algoritmi, ognuno dei quali fornisce risultati leggermente differenti dagli altri [9]. Il metodo *profile*[6] è stato, applicato sull'architettura di Elman sviluppata precedentemente per prevedere la temperatura al tempo $t+1$. L'idea base è quella di studiare l'influenza di una variabile di input, fissando i valori delle altre variabili di ingresso.

In Figura 13 sono mostrati i risultati ottenuti. Come prevedibile, la temperatura al tempo t è l'ingresso che presenta un peso maggiore nella costruzione della uscita.

In Figura 14 sono mostrati i grafici ottenuti variando una variabile per volta e fissando tutte le altre variabili meno una.

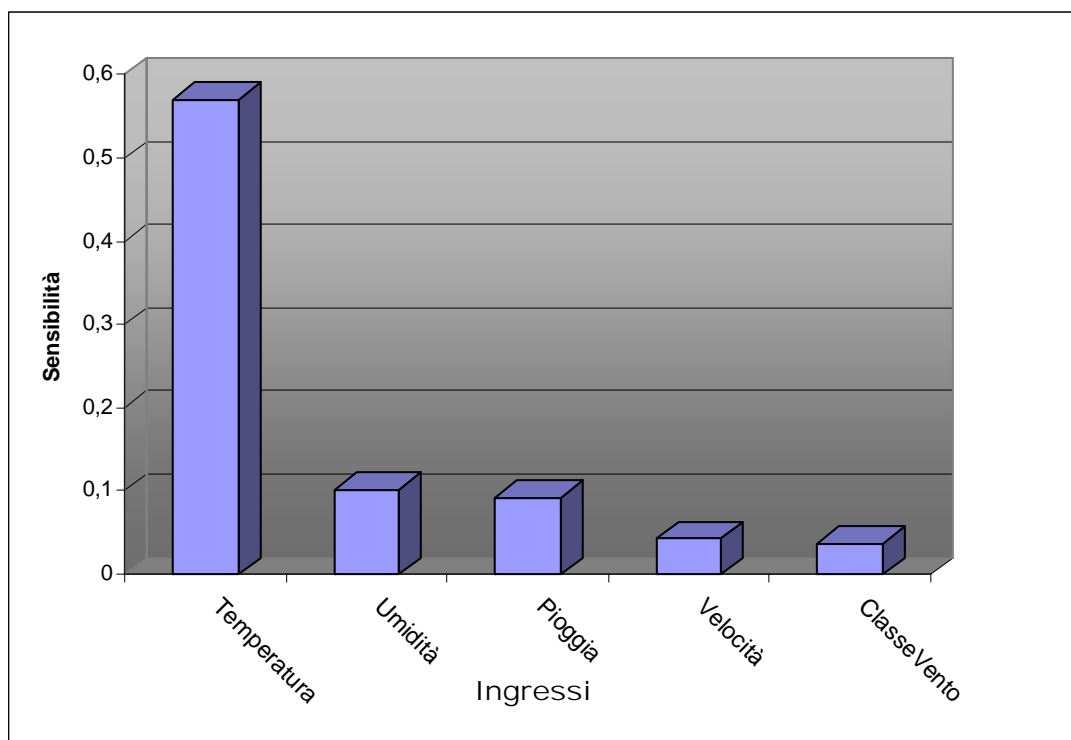
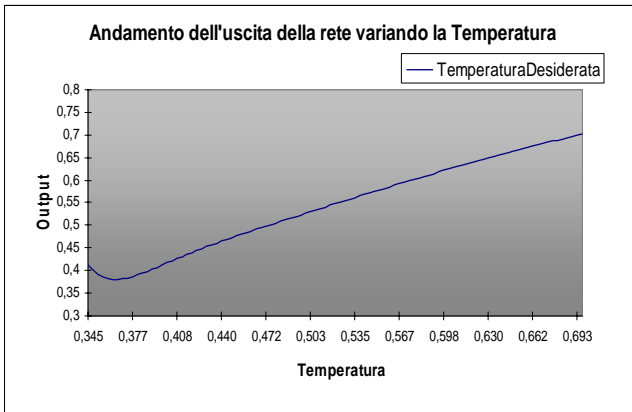
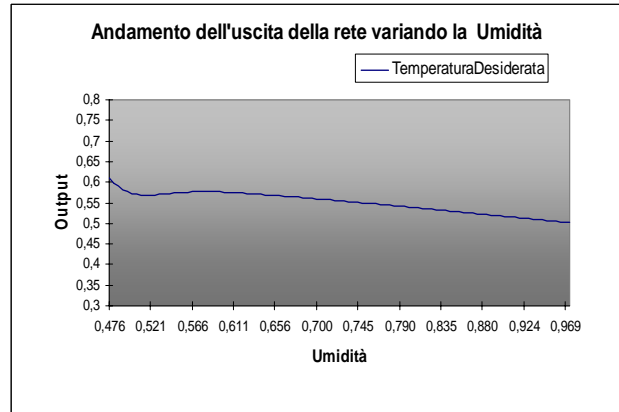


Figura 13 – Sensibilità delle grandezze di ingresso sull'uscita

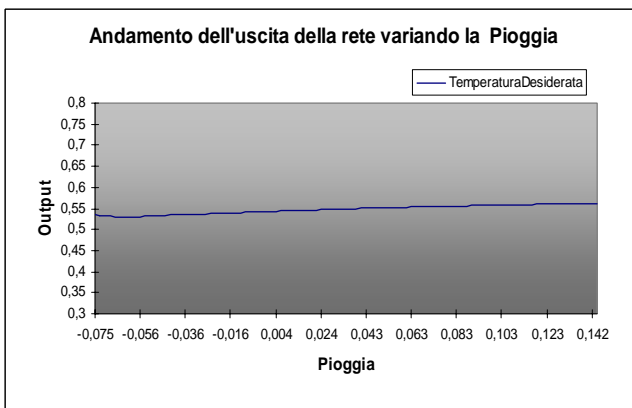
a)



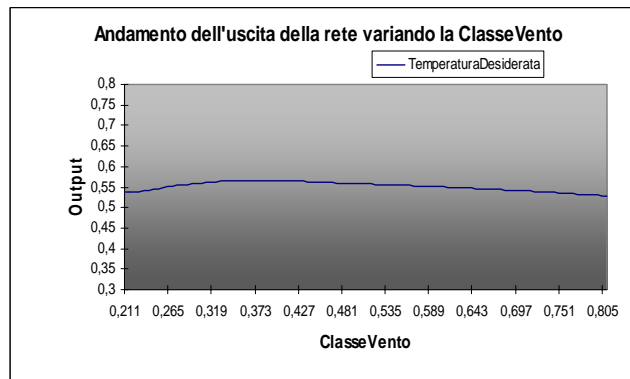
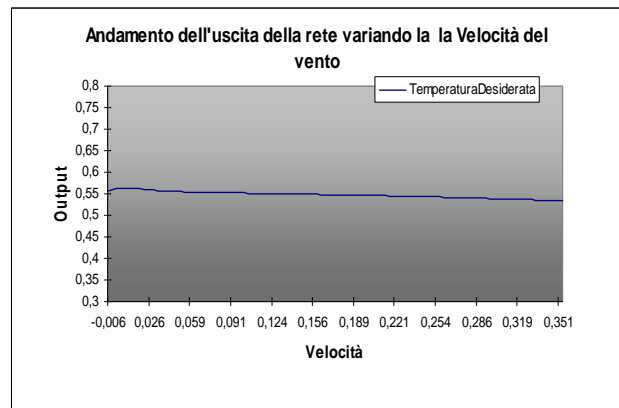
b)



c)



d)



e)

Figura 14 – a) Uscita della rete di Elman variando la temperatura, b) Variando l'umidità, c) Variando la pioggia, d) Variando la velocità del vento, e) Variando la classe del vento

2 Reti di Bayes

Le Reti Bayesiane (RB), note anche come reti probabilistiche e reti di credenze, sono modelli grafici rappresentanti la conoscenza in un dominio incerto. Basandosi sulla regola di Bayes, le RB esprimono relazioni di dipendenza condizionale (archi) tra le variabili in gioco (nodi). Il vantaggio principale del ragionamento probabilistico rispetto a quello logico sta nella possibilità di giungere a descrizioni razionali anche quando non vi è abbastanza informazione di tipo deterministico sul funzionamento del sistema. Come nel caso delle RNA, anche le RB apprendono dai dati tuttavia con due differenze fondamentali. La prima è che la conoscenza di esperti può facilmente essere codificata nelle RB, incrementando l'efficienza e l'accuratezza dell'apprendimento, la seconda è che l'informazione codificata nei nodi e negli archi delle RB addestrate risulta più comprensibile. Inoltre in molti sistemi complessi, esprimere le relazioni in forma probabilistica fornisce un modo naturale di codificare la conoscenza.

Dati due eventi A e B, se questi sono in qualche modo correlati, è ragionevole pensare che il fatto che uno dei due è già avvenuto possa migliorare la conoscenza della probabilità dell'altro. A partire da queste considerazioni, si può quindi impostare una rete di conoscenze sfruttando delle informazioni già note per meglio prevedere altre variabili.

Per modellare tale comportamento è necessaria una teoria statistica dell'evidenza: tale teoria è fornita dalla statistica bayesiana (5), il cui concetto fondamentale è la probabilità condizionale $P(A | B)$, che esprime la probabilità dell'ipotesi A una volta osservata l'evidenza B.

Il teorema di Bayes afferma che:

$$P(A | B) = \frac{P(B | A) \cdot P(A)}{\sum_{n=1}^k P(B | A) \cdot P(A_n)} \quad (5)$$

dove k è il numero dei possibili stati di A.

Il risultato dell'analisi dei dati basata sul metodo di Bayes è la probabilità a posteriori che rappresenta una correzione, un miglioramento della probabilità a priori alla luce dell'evidenza fornita dai dati[10].

Una delle caratteristiche più attraenti del sistema è che man mano che i dati diventano disponibili, la rete bayesiana consente di combinarli inglobandoli per aggiornare le credenze e aumentare la conoscenza.

Le RB, dati i valori esatti (dette evidenze) di alcune variabili di prova, tramite il meccanismo dell'inferenza, consentono di calcolare la distribuzione di probabilità a posteriori per un insieme di variabili di interrogazione. Inoltre ogni nodo può agire sia come variabile di interrogazione che come variabile di prova.

Nel progetto MOGEIFACO le RB sono state utilizzate per sviluppare un modello che mira alla comprensione delle relazioni tra gli indici dell'area costiera di interesse.

Sono tre i motivi per cui si è deciso di utilizzare le RB:

- 1) Risulta naturale esprimere i legami fra i vari indici e le relative zone costiere d'interesse in termini probabilistici.
- 2) Dal momento che è un insieme di addestramento è limitato, risulta molto utile potere trasferire la conoscenza di un esperto del campo sul modello.
- 3) Man mano che si rendono disponibili nuove evidenze, queste possono essere facilmente inglobate nel modello.

2.1 Il Modello bayesiano sviluppato

La realizzazione di una RB si articola in due fasi: la specifica della struttura del dominio e la definizione per ogni nodo della tabelle di probabilità condizionata (TPC).

In questo studio, per ricavare la struttura della rete si è utilizzato il metodo di deduzione con un esperto (expert elicitation), mentre le TPC sono state ottenute automaticamente a partire dai dati utilizzando il software Netica nella versione 2.17, sviluppata dalla Norsys Software Incorp[13].

L'esperto per ricavare una struttura che soddisfi il modello, deve seguire i seguenti passi :

- Identificare le variabili più importanti, ed assegnare ad ogni variabile un nodo;
- Identificare gli stati che possono assumere le variabili;
- Identificare le possibili relazioni che possono intercorrere tra le variabili;
- Valutarne la bontà della struttura, tramite l'adozione di una metrica;

Identificazione delle variabili più importanti

L'obiettivo era quello di sviluppare un modello che potesse rappresentare la dinamica degli indicatori delle dieci zone fisiografiche nelle quattro stagioni. È risultato naturale assegnare ad ogni indicatore una variabile (e dunque un nodo). Con riferimento alla Figura 3, per evitare di rendere troppo complesso e difficile l'addestramento si è deciso di escludere i "dati grezzi" dal modello.

Oltre agli indicatori sono stati aggiunti altri due nodi (Zone Fisiografiche, Stagione), per mettere in relazione gli indicatori con le rispettive zone fisiografiche e il periodo dell'anno a cui si riferiscono.

Identificazione gli stati del modello

Gli indicatori sviluppati assumono un valore numerico compreso fra 0 e 10. Nei modelli bayesiani spesso risulta conveniente, per l'addestramento, tramite operazioni di raggruppamento (clustering e applicazione delle tecniche fuzzy) ridurre il numero degli stati che possono assumere le variabili del modello. Tramite operazione di clustering si è proceduto a ridurre il numero degli stati degli indicatori a soli tre valori, come evidenziato in Tabella 5.

Tabella 5 – Riduzione del numero degli stati

Valore Indicatore	Stato nodo
da 0 a 3,33	Low
da 3,33 a 6,66	Medium
da 6,66 a 10	Height

In Tabella 6 sono mostrati i nodi che fanno parte del modello sviluppato e i relativi stati che possono assumere.

Tabella 6 - Variabili e stati del modello. In grassetto sono mostrate le variabili ottenute tramite aggregazione di altre variabili.

Variabile o nodo	Stati del nodo
Zone_Fisiografiche	Stagnone, Marsala, Rina, Biscione, Capo_Feto, Tonnarella, Quarara, Capo_Granitola, Tre_Fontane, Selinunte
Stagione	Primavera, Estate, Autunno, Inverno
Balneazione	Low, Medium, Height
Trix	Low, Medium, Height
Ind_Ambientale	Low, Medium, Height
Densita_abitativa	Low, Medium, Height
Sostenibilita_edilizia	Low, Medium, Height
Occ_spiagge	Low, Medium, Height
Incidenza_turistica	Low, Medium, Height
Ind_Antropico	Low, Medium, Height
Pescosita	Low, Medium, Height
Numero_natanti	Low, Medium, Height
Ind_Pescosita	Low, Medium, Height
Ind_Naturalistico	Low, Medium, Height
Sfruttamento_agricolo	Low, Medium, Height
Ricettivita_turistica	Low, Medium, Height
Ind_Economico	Low, Medium, Height
Ind_Sostenibilita	Low, Medium, Height

Identificazione delle relazioni del modello

Tramite relazioni (rappresentate da archi) è possibile definire le dipendenze condizionali tra le variabili (nodi) del modello. Uno degli scopi del progetto MOGEIFACO è attribuire un valore chiamato “Indice di Sostenibilità” ad ogni zona fisiografica. Tale indicatore è espresso in funzione al valore assunto dagli altri indicatori (si trova infatti al livello di astrazione più in alto della Figura 3). In base a questa osservazione è stato naturale rappresentare il modello bayesiano in maniera gerarchica, rispettando le relazioni fra i vari indicatori.

In Figura 15 è mostrata l'architettura bayesiana sviluppata, essa è possibile dividerla in 4 livelli:

- SPAZIO-TEMPORALE: composto dai nodi Zone_Fisiografiche e Stagione permette di individuare in maniera univoca la zona fisiografica di riferimento e il relativo periodo.
- DATI ELABORATI: il valore assunto da questi nodi è ottenuto tramite processi di aggregazione dei dati grezzi (Figura 3).
- INDICATORI: il valore assunto da questi nodi è ottenuto tramite processi di aggregazione dei dati elaborati (Figura 3) sfruttando la conoscenza di un esperto del dominio.
- INDICE DI SOSTENIBILITÀ: il valore assunto da questo nodo è ottenuto tramite processi di aggregazione degli indicatori.

Valutazione tramite una metrica della bontà del modello sviluppato

Come nelle RNA anche per i modelli RB spesso si procede per tentativi. Dopo aver ottenuto un primo modello, si apportano su tale modello alcuni cambiamenti, modificando ad esempio il numero dei nodi, le relazioni (archi) tra i nodi; con il fine di ottenere modelli diversi che meglio approssimano la rappresentazione del dominio applicativo. Su tali nuovi modelli è stata applicata una metrica (nel nostro caso la *Metric learning* [11]) per valutarne la bontà.

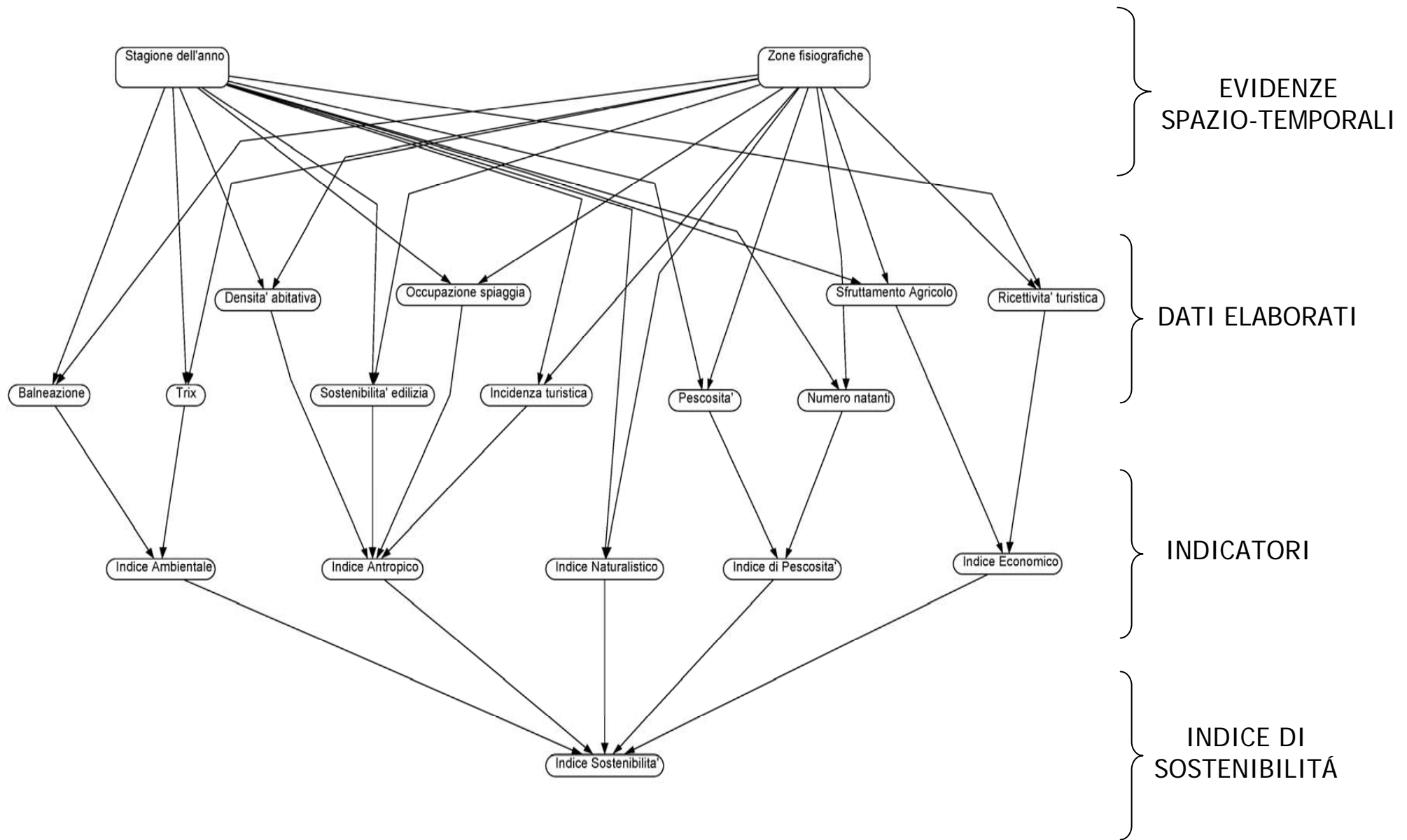


Figura 15 – Modello Bayesiano sviluppato

Dopo aver determinato la struttura della rete si procede con la definizione per ogni nodo della Tabelle di Probabilità Condizionata (TPC). L'apprendimento della rete bayesiana è stato eseguito a partire dai casi noti, dove un caso è dato dall'insieme di tutte le evidenze dei nodi.

Il numero di casi a disposizione lascia diverse combinazioni delle variabili con una distribuzione di probabilità uniforme (nessuna informazione), se queste combinazioni dovessero verificarsi durante le query, la rete non sarebbe in grado di fornire una corretta inferenza. Per ovviare a ciò sarebbero necessari ulteriori casi di addestramento per determinare TPC più complete. Per evitare il rischio di avere risultati eccessivamente anomali si è proceduto manualmente alla compilazione dei dati mancanti procedendo per interpolazione dai casi noti. In Figura 16 è mostrata la tabella di probabilità condizionata (TCP) del nodo "Indice Sostenibilità".

Indice Antropico	Indice Naturalistico	Indice di Pescosità	Indice Ambientale	Indice Economico	Low	Medium	Height
Low	Low	Low	Height	Medium	10.000	90.000	0.000
Low	Low	Low	Height	Height	0.000	100.00	0.000
Low	Low	Medium	Low	Low	100.00	0.000	0.000
Low	Low	Medium	Low	Medium	60.000	40.000	0.000
Low	Low	Medium	Low	Height	20.000	80.000	0.000
Low	Low	Medium	Medium	Low	70.000	30.000	0.000
Low	Low	Medium	Medium	Medium	20.000	80.000	0.000
Low	Low	Medium	Medium	Height	0.000	100.00	0.000
Low	Low	Medium	Height	Low	30.000	70.000	0.000
Low	Low	Medium	Height	Medium	0.000	100.00	0.000
Low	Low	Medium	Height	Height	0.000	100.00	0.000
Low	Low	Height	Low	Low	90.000	10.000	0.000
Low	Low	Height	Low	Medium	30.000	70.000	0.000

Figura 16 - Tabella di Probabilità Condizionata del nodo "Indice Sostenibilità"

Dopo che una rete di credenze è stata costruita, può essere applicata ad un caso particolare. Ogni variabile (nodo) di cui il valore (stato) è noto diventerà un'evidenza, lo strumento software permette di fare inferenze probabilistiche per trovare le credenze per tutte le altre variabili.

Di seguito sono mostrati due esempi di inferenza probabilistica nel modello sviluppato. In Figura 17 si può notare che fissando gli stati dei nodi *Stagione dell'anno*, *Zona fisiografica*, *Densità abitativa*, *Occupazione spiaggia*, *Pescosità*, *Numero natanti*, *Indice ambientale*, *Indice naturalistico* e *l'Indice economico*, tramite il meccanismo dell'inferenza probabilistica si possono calcolare i valori in termini probabilistici dei restanti nodi. In particolare si nota che molto probabilmente (56,3%) in funzione delle scelte fatte, l'indice di sostenibilità sarà alto (Height). In Figura 18 è mostrato un altro esempio in cui fissando il valore dei nodi: *Stagione dell'anno*, *Zona fisiografica*, *Balneazione*, *Densità abitativa*, *Occupazione spiaggia*, *Indice naturalistico*, *Pescosità*, *Numero natanti* e *l'Indice economico*, si ottiene come indice di sostenibilità della zona fisiografica selezionata un valore medio (63,6%).

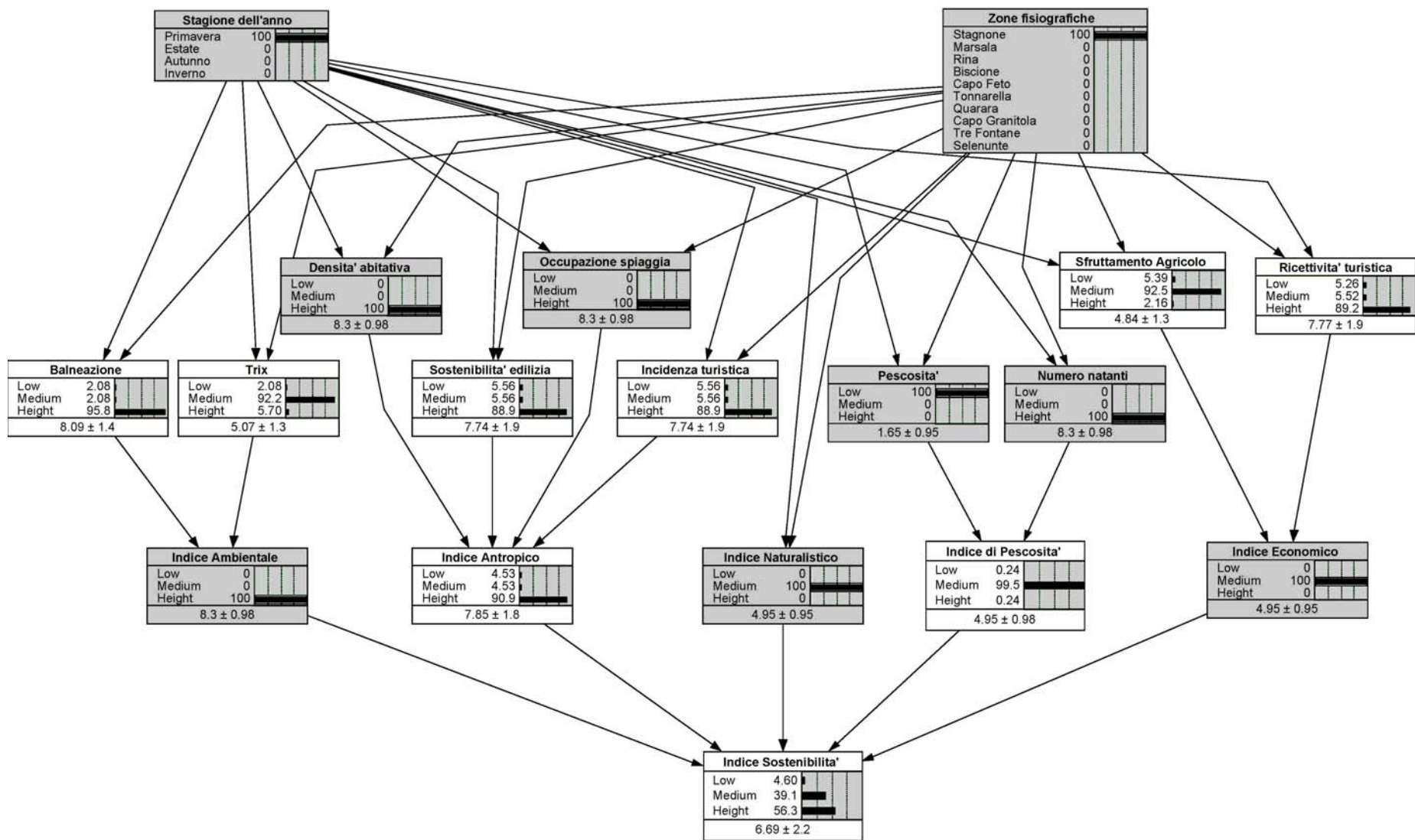


Figura 17 – Esempio di inferenza probabilistica nel modello sviluppato : nell’area fisiografica denominata “Stagnone” in “Primavera” si ha un indice di sostenibilita' elevato (probabilita' 56,3%)

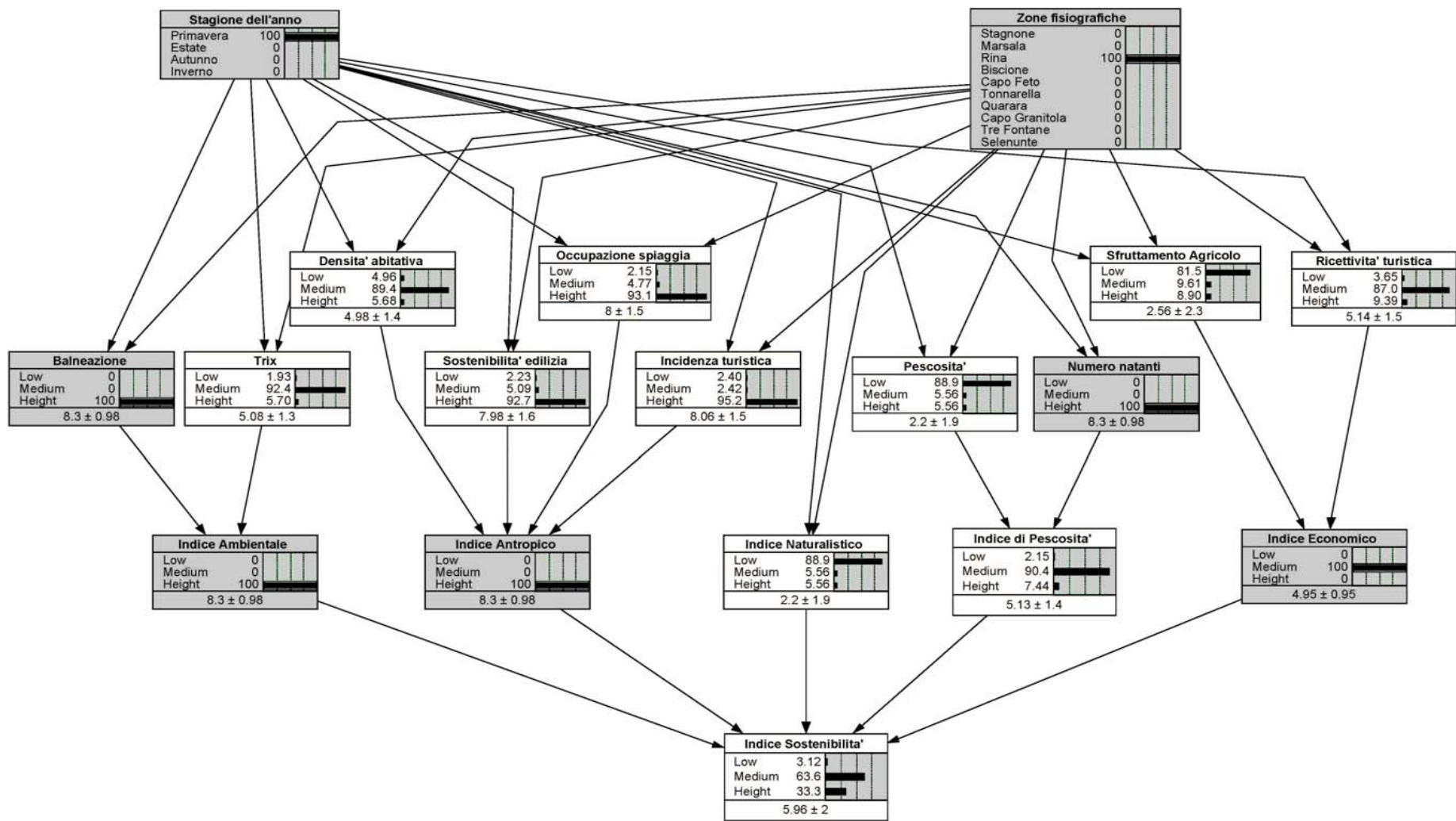


Figura 18 - Esempio di inferenza probabilistica nel modello sviluppato: nell'area fisiografica denominata "Rina" in "Primavera" si ha un indice di sostenibilita' medio (probabilita' 63,6%)

Riferimenti bibliografici

- [1] M Riedmiller and H Braun. A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In Proceedings of the IEEE International Conference on Neural Networks 1993 (ICNN 93), 1993.
- [2] Hannan, J.M. & Bishop, J.M., (1997), A comparison of fast training algorithms over two real problems, , Proc. 5th Int. Conf. ANN, pp: 1-6, Cambridge, UK.
- [3] Lapedes, R., Farber, R. (1987) "Non linear signal processing using neural networks: prediction and system modelling" Los Alamos National Laboratory, Los Alamos, New Mexico.
- [4] W.S. Sarles. (a cura di) (2002) Neural networks FAQs.(dal newsgroup Usenet comp.ai.neural-net). Da: <ftp://ftp.sas.com/pub/neural/FAQ.html.zip>
- [5] Elman, J. L. (1990), "Finding structure in time". In Cognitive Science, n.14 pp. 179-211.
- [6] Lek, S., Belaud, A., Dimopoulos, I., Lauga, J., Moreau, J.,1995. Improved estimation, using neural networks, of the food consumption of fish populations. Marine Freshwater Research 46, 1229-1236.
- [7] D. Heckerman, (1996) "A Tutorial on learning with Bayesian Networks" Tech. Report Microsoft Research da: <ftp://ftp.research.microsoft.com/pub/tr/TR-95-06.PS>
- [8] D. Heckerman, (1996) "A Tutorial on learning with Bayesian Networks" Tech. Report Microsoft Research da: <ftp://ftp.research.microsoft.com/pub/tr/TR-95-06.PS>
- [9] Muriel G., Ioannis D., Sovan L. (2003) " Review and comparison of methods to study the contribution of variables in artificial neural network models"
- [10] Berthold M., Hand D. J. (1998). Intelligent data analysis. Springer Berlin 1998
- [11] Friedman, N., e Goldszmidt, M., "Learning Bayesian Networks with Local Structure" in Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence, pp. 252-262, San Francisco: Morgan Kaufmann, 1996
- [12] D.E. Rumelhart and J.L. McClelland. Parallel Distributed Processing, volume 1. MIT Press, 1986.
- [13] Stuttgart Neural Network Simulator, url: <http://www-ra.informatik.uni-tuebingen.de/SNNS/>.
- [14] Netica, Norsys Software Incor. url: <http://www.norsys.com/netica.html>
- [15] G.Pilato, F. Sorbello, G. Vassallo - "An Innovative Way to Measure the Quality of a Neural Network without the Use of the Test Set" - IJACI International Journal of Advanced Computational Intelligence - Vol.5, No.1, pp.31-36, 2001