



CONSIGLIO NAZIONALE DELLE RICERCHE
ISTITUTO DI CALCOLO E RETI AD ALTE PRESTAZIONI

MIGET

MODELLI INTELLIGENTI PER LA GESTIONE DELLA FASCIA COSTIERA DELLA PROVINCIA DI TRAPANI

A. CIMINO, S. GAGLIO, S. VITABILE

RT-ICAR-PA-07-01

Giugno 2007



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

MIGET

MODELLI INTELLIGENTI PER LA GESTIONE DELLA FASCIA COSTIERA DELLA PROVINCIA DI TRAPANI

A. CIMINO¹, S. GAGLIO^{1,2}, S. VITABILE^{1,3}

Rapporto Tecnico N.: 1
RT-ICAR-PA-07-01

Data:
Giugno 2007

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

² Università degli Studi di Palermo - Dipartimento di Informatica

³ 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.

Sommario

Introduzione	2
1 Reti neurali artificiali ricorrenti.....	4
1.1 La serie storica.....	5
1.2 La preparazione dei dati.....	6
1.2.1 Controllo di qualità	6
1.2.2 Partizionamento	6
1.2.3 Normalizzazione	7
1.3 Il modello neurale sviluppato.....	7
1.3.1 Architettura di Elman.....	8
1.3.2 Risultati sperimentali.....	11
1.3.3 Analisi della sensibilità	17
2 Le reti bayesiane	20
2.1 Il modello bayesiano sviluppato	21
2.2 Descrizione dell'insieme dei dati	21
2.3 Risultati sperimentali.....	25
3 Integrazione dei modelli intelligenti nel framework ArcGIS 9.2: studio di fattibilità.....	28
3.1 Obiettivi generali	28
3.2 Il sistema ArcGIS	29
3.3 I modelli intelligenti sviluppati	30
3.4 Strategie di integrazione.....	32
3.4.1 Estensione delle funzionalità di ArcGIS	32
3.4.2 Sviluppo di un'applicazione indipendente.....	34
3.5 Conclusioni sullo studio di fattibilità	36
4 L'integrazione di un modello neurale e di un modello bayesiano in ArcGIS.....	37
Conclusioni	46
Indice delle figure	47
Bibliografia.....	48
Appendice A: Procedura di installazione.....	49
Installazione su disco fisso.....	49
Installazione in ArcGIS.....	50

Introduzione

Con il termine *Intelligenza Artificiale*[1], si indica l'abilità di un elaboratore di poter eseguire delle operazioni senza l'intervento dell'uomo, e che coinvolge capacità quali ragionamento, apprendimento, comunicazione e azione in ambienti complessi. Tramite queste capacità, l'elaboratore, è in grado di svolgere funzioni e ragionamenti, tipici di una mente umana.

Le metodologie sviluppate nel campo dell'Intelligenza Artificiale sono state utilizzate in diversi campi, quali la robotica, l'automazione industriale, l'elaborazione delle immagini ed anche nell'analisi dei dati. Quest'ultimo campo è di particolare importanza in quanto, a causa dell'elevata quantità di dati con cui dobbiamo convivere nei giorni nostri, occorre utilizzare dei modelli intelligenti per l'analisi dei dati che ci permettano di ottenere informazioni utili nel minor tempo possibile, cosa che ovviamente non otterremmo senza l'utilizzo congiunto degli elaboratori e delle tecniche messe a disposizione dall'Intelligenza Artificiale.

Il progetto *MIGET* ha lo scopo di sviluppare dei modelli intelligenti che possano analizzare i dati raccolti nell'ambito del progetto *DrepanCoast*, e di elaborare le previsioni sul comportamento di alcune grandezze.

Il processo di analisi, come mostrato in Figura 1, comincia con il rilevamento dei dati, che vengono classificati per tipologia, controllati ed eventualmente integrati tramite interpolazioni. Una volta che le serie storiche sono pronte, si individuano i modelli intelligenti più adeguati per la tipologia di analisi da effettuare, e si sviluppano diversi prototipi per poterne testare l'efficienza. Infine, si modificano i vari parametri per poterne ottimizzare il modello.

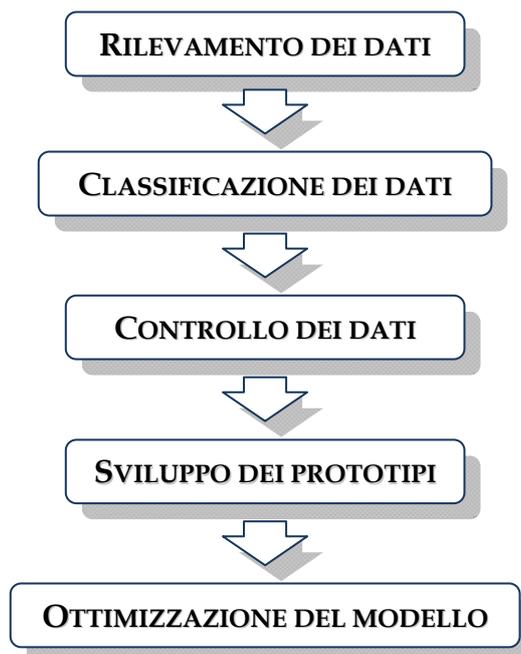


Figura 1: Il processo di analisi

Tramite la fase del rilevamento, è stata creata la banca dati del progetto *DrepanCoast*, che mette a disposizione dati che sono stati opportunamente classificati in base alle loro caratteristiche, vedi Figura 2, quali appunto dati meteorologici, dati ambientali, dati socio-economici, dati di pesca, dati sulle infrastrutture viarie ed altri. La serie storica di tali rilevamenti, fa riferimento al periodo che va dall'anno 2000 al 2006, anche se per alcune tipologie di dati la serie risulta essere ridotta, per mancanza di dati, o con valori stazionari, a causa di alcuni fenomeni che non variano nel corso degli anni.

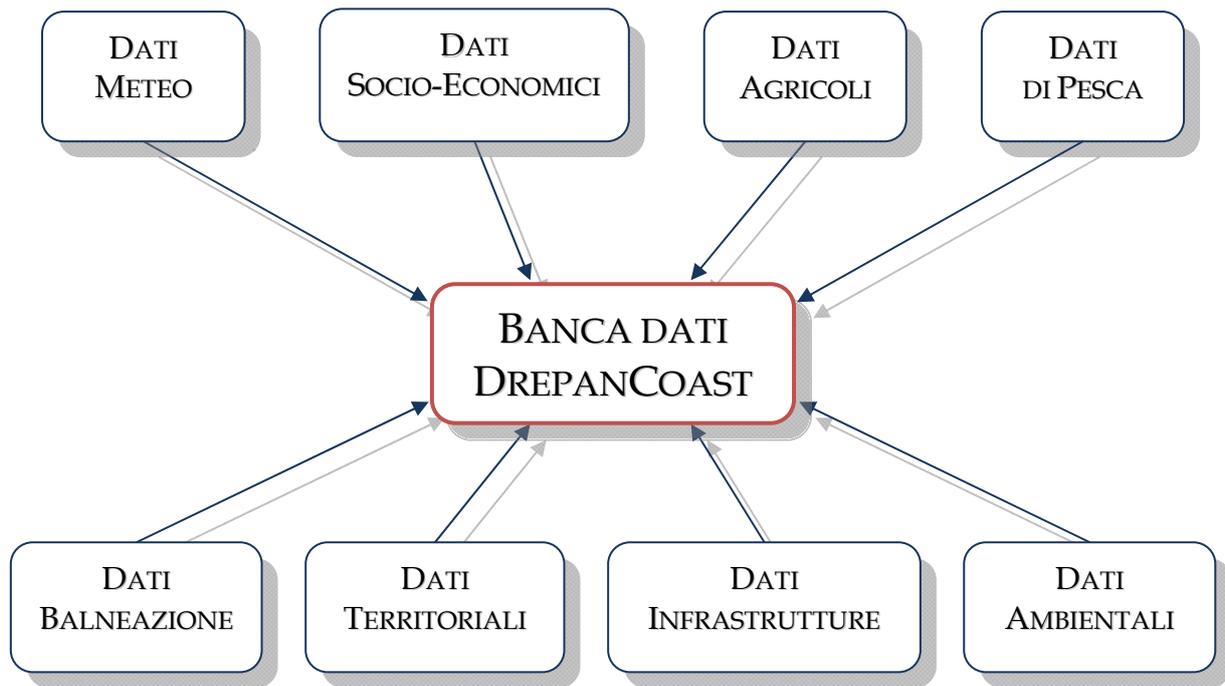


Figura 2: La banca dati del progetto DrepanCoast

Per l'analisi di questi dati sono stati sviluppati due modelli intelligenti, caratterizzati entrambi dalle architetture a grafo, con nodi e archi.

Il primo modello si basa sulle *Reti Neurali Artificiali (RNA)*, che sono delle reti di neuroni artificiali, che sviluppano alcune proprietà dei neuroni biologici, simulandone il funzionamento all'interno di un sistema informatico. Queste reti sono conosciute anche per il loro cosiddetto funzionamento a *scatola nera*, in quanto, fornendo gli ingressi e le uscite della rete durante la fase di addestramento, cerca di aggiornare dei parametri, detti pesi, per poterne individuare la relazione ingresso-uscita. Questa relazione matematica, rimarrà a far parte della rete, e non vi è alcun modo per poterne determinare la formula, anche se essa dovesse essere una relazione poco complessa.

Il secondo modello sviluppato si basa sulle *Reti Bayesiane (RB)*, note anche come reti di credenze, che sono dei grafi orientati e aciclici, in cui i nodi sono le variabili e gli archi rappresentano le relazioni di dipendenza probabilistica (che può essere incerta o imprecisa), tra i nodi della rete.

1 Reti neurali artificiali ricorrenti

Il modello della rete neurale artificiale, si basa sull'interconnessione di semplici unità elaborative chiamate neuroni. Il modello più semplice per la rappresentazione del neurone artificiale, è stato proposto da McCulloch e Pitts, ed è chiamato *Unità Logica a Soglia (ULS)*, mostrata in Figura 3.

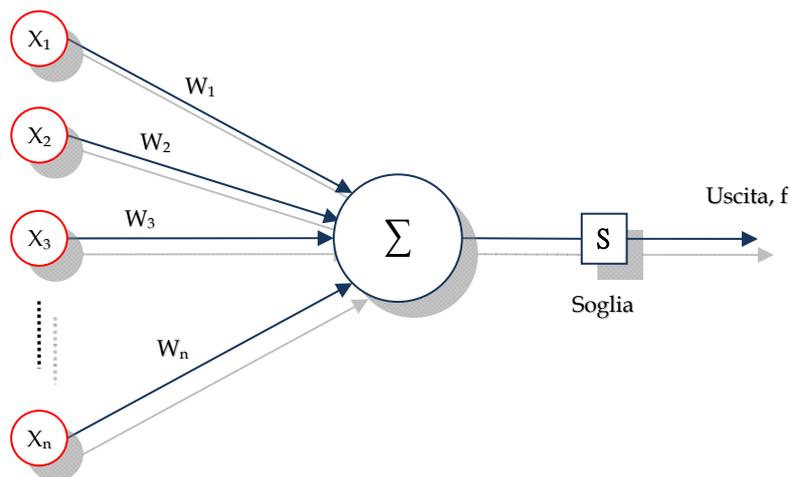


Figura 3: Un'unità logica a soglia

I neuroni, calcolano la somma pesata degli ingressi e confrontano questa somma con il valore della soglia e, a seconda se il valore di quest'ultima è superato o meno, viene prodotta l'uscita del neurone.

Il modello dell'unità logica a soglia è un caso semplice utilizzato principalmente per far comprendere il funzionamento di un neurone artificiale, ma esistono altri modelli in cui l'elaborazione eseguita dal neurone è più complessa e sofisticata.

Una rete neurale artificiale, è dunque composta dall'insieme di neuroni collegati tra di loro in modo da formare degli strati (o livelli), dove il primo rappresenta gli ingressi, l'ultimo rappresenta le uscite, e i livelli intermedi sono detti livelli nascosti.

A seconda di come vengono eseguite le connessioni tra i vari neuroni e i vari livelli, si hanno le diverse tipologie di rete neurale, che possono essere completamente connesse o meno, oppure essere connesse solo in avanti (*feedforward*) o presentare dei cicli (*feedback o ricorrenti*).

La caratteristica principale delle reti neurali, è quella di poter imparare, cioè fornendo alla rete un insieme di coppie ingresso-uscita, la rete riesce ad apprendere le regolarità presenti nella distribuzione temporale di questi ingressi in quella particolare *storia*.

Questa caratteristica, si è rilevata molto utile per il progetto MIGET, poiché le reti neurali si prestano bene per elaborare una previsione delle temperature.

Fornendo in ingresso un vettore di dati al tempo t , una volta addestrata, la rete elabora il vettore di uscite al tempo $t+1$.

Per il progetto *MIGET*, è stata scelta un'architettura di tipo ricorrente, che presenta cioè delle connessioni in avanti, ma anche delle connessioni indietro verso i livelli precedenti, creando così dei cicli all'interno della rete. Questo tipo di architettura è molto utile nel nostro caso, in quanto, come vedremo più avanti, otterremo come ingressi il doppio dei valori, considerando così gli ingressi al tempo $t-1$, ma anche gli ingressi al tempo $t-2$. Facendo così la rete riuscirà ad apprendere meglio le relazioni ingresso-uscita e a seguire meglio l'andamento delle grandezze oggetto d'analisi.

Le informazioni meteorologiche oggetto d'analisi, riguardano i seguenti parametri giornalieri rilevati in sei stazioni meteorologiche:

- Temperatura massima
- Temperatura minima
- Precipitazione
- Velocità del vento
- Direzione del vento

In Figura 4, è mostrato l'andamento normalizzato delle grandezze prese in esame.

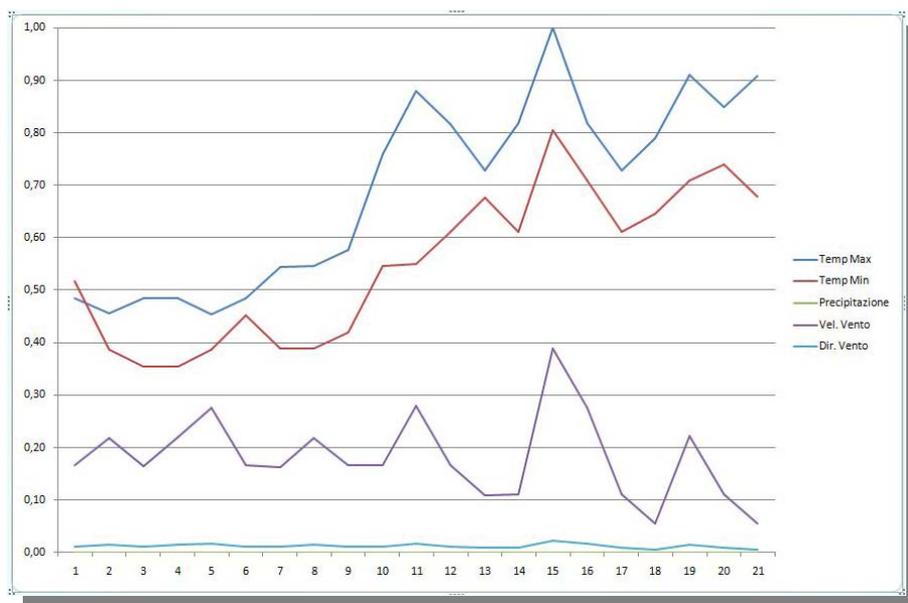


Figura 4: Andamento normalizzato delle grandezze analizzate.

1.1 La serie storica

I dati meteorologici sopra descritti si riferiscono a sei stazioni meteo presenti nella provincia di Trapani, e precisamente le stazioni sono situate nelle zone di Castellammare del golfo, Castelvetro, Erice, Marsala, Mazara del vallo e Trapani.

I dati sono stati rilevati nel periodo che va dal 01 gennaio 2002 al 31 luglio 2006, per un totale di 1673 giorni. Le rilevazioni sono state fatte con frequenza giornaliera, come è possibile vedere dalla Tabella 1.

Data	Temperatura Max °C	Temperatura Min °C	Precipitazione ml	Velocità del Vento m/s	Direzione del Vento ° gradi
02/03/2002	26,06	16,09	0	10,02	135
03/03/2002	24,05	11,03	0	9,03	180
04/03/2002	22,09	10,07	0	9,08	180
05/03/2002	30,04	17,08	0	17,03	180
06/03/2002	18,01	13,03	0	6,05	0
07/03/2002	18	14	0	5,05	315
08/03/2002	18,04	13,02	0,2	4,04	0
09/03/2002	21,04	10,01	0,2	5,03	135
10/03/2002	17	12,06	0,2	6,05	315

Tabella 1: I dati meteorologici rilevati

Questi dati sono stati divisi in tre insiemi, uno per l'addestramento della rete neurale di circa 1100 campioni, uno per la convalida dell'addestramento di 365 campioni, ed infine l'ultimo insieme di circa 210 campioni utilizzato per eseguire un test di verifica sui risultati generati dalla rete neurale.

1.2 La preparazione dei dati

Prima di procedere con l'addestramento della rete neurale, i dati devono essere controllati, separati e convertiti in una forma opportuna.

1.2.1 Controllo di qualità

Per tutte le tipologie di dati, occorre sempre eseguire un controllo sui campioni, in modo da poter rilevare degli errori nascosti che potrebbero compromettere l'efficienza del sistema stesso. Inizialmente occorre verificare che vi sia la continuità dei campioni, poiché alcune rilevazioni giornaliere potrebbero mancare, e quindi occorre ripristinare i dati mancanti tramite l'utilizzo di tecniche d'interpolazione. Una volta che la serie storica non presenta dati mancanti, si può procedere con il controllo sull'ammissibilità dei valori, cioè occorre verificare che i dati rilevati siano veritieri, controllando che appartengano a un dato intervallo di riferimento, secondo la tipologia di campione.

1.2.2 Partizionamento

Come già accennato in precedenza, i dati devono essere partizionati in tre gruppi, per eseguire l'addestramento, la convalida e il test della rete neurale.

L'insieme dei campioni per l'addestramento è stato ricavato dai dati dall'anno 2002 al 2004, ottenendo 1096 campioni. Per l'insieme di campioni da utilizzare per la convalida dell'addestramento, sono stati presi i dati dell'anno 2005, quindi 365 campioni, e infine per il test di verifica sulle prestazioni della rete, sono stati utilizzati i dati che vanno da gennaio 2006 a luglio 2006, circa 210 campioni.

1.2.3 Normalizzazione

Per migliorare le prestazioni della rete neurale, anche se non necessariamente richiesto, è necessario eseguire una normalizzazione dei dati che la rete deve elaborare, producendo così una più rapida convergenza ed evitando il rischio di minimi locali[5].

L'operazione è stata eseguita normalizzando le grandezze di ingresso e di uscita, nell'intervallo [0,1] utilizzando la formula

$$N = \frac{V - \min}{\max - \min} \quad (1)$$

dove V è il valore da normalizzare, \max e \min sono i valori massimo e minimo dell'intervallo a cui appartiene V , e infine N è il valore normalizzato nell'intervallo [0,1].

1.3 Il modello neurale sviluppato

Il modello neurale sviluppato per il progetto MIGET, si basa sulle reti neurali ricorrenti, cioè su quella tipologia di grafo che consente di avere sia connessioni in avanti da un livello al successivo, e sia connessioni verso livelli precedenti, creando così dei cicli all'interno della rete.

Un modello di rete neurale ricorrente, mostrato in Figura 5, fu sviluppato nel 1990 da Jeffrey L. Elman[6], che creò così una rete che impara a immagazzinare quelle proprietà dei dati percepiti precedentemente, che risultano appropriati al compito in questione. Essa è una variazione del perceptrone multistrato, formando così una rete a tre strati, in cui nel primo abbiamo il vettore con i dati di ingresso e delle unità di contesto, nel secondo strato abbiamo delle unità nascoste, e nell'ultimo strato vi sono le uscite dalla rete.

Le unità di contesto, sono degli ingressi aggiuntivi creati dalle connessioni ricorrenti con le unità nascoste, e ritardate di una unità di tempo. Sono proprio queste unità che permettono alla rete di basare le proprie azioni su proprietà apprese dai dati analizzati precedentemente. Tutte le unità della rete sono completamente connesse con le unità degli strati successivi.

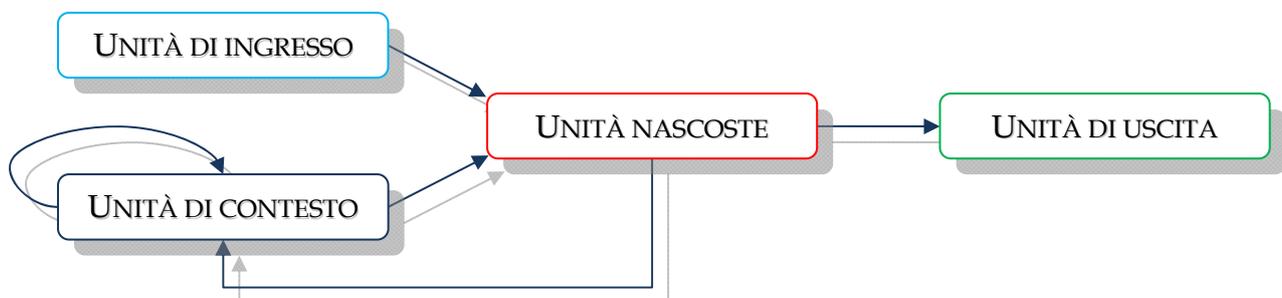


Figura 5: Lo schema dell'architettura neurale di Elman

Per lo sviluppo dell'architettura neurale, è stato utilizzato il software di simulazione *SNNS v4.2 (Stuttgart Neural Network Simulator)*[14], operante sia in ambiente Linux, sia in ambiente Windows. Tramite quest'applicazione è stato possibile mettere a punto una rete di Elman che fosse in grado di elaborare una previsione sulla temperatura massima al tempo $t+1$, una volta forniti in ingresso i dati meteo, descritti in precedenza, al tempo t .

1.3.1 Architettura di Elman

L'architettura di Elman sviluppata, è stata ottenuta creando, tramite il software SNNS v4.2, una rete di Elman che abbia le caratteristiche mostrate in Tabella 2.

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

Tabella 2: L'architettura neurale di Elman

Le funzioni di attivazione sono tutte di tipo lineare, tranne che per il caso delle unità nascoste in cui è stata utilizzata una funzione di tipo sigmoideale.

Gli ingressi della rete sono i cinque dati meteorologici al tempo t , quindi Temperatura massima, Temperatura minima, Precipitazione, Velocità del vento e Direzione del vento.

L'uscita della rete è la Temperatura massima al tempo $t+1$.

In Figura 6, è possibile vedere la rete neurale di Elman che è stata sviluppata.

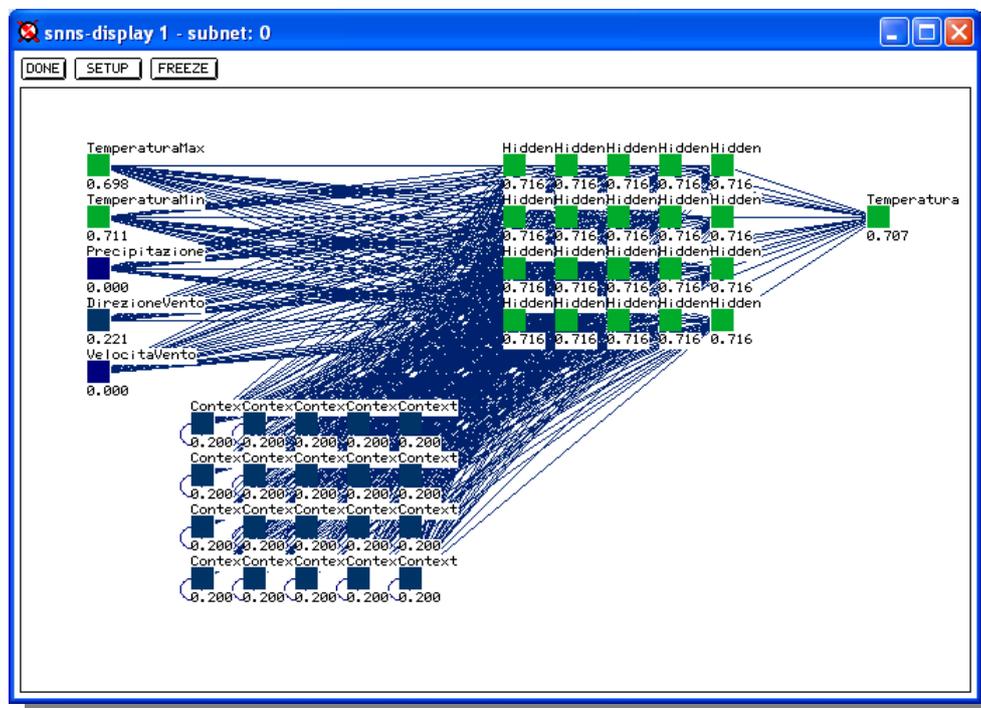


Figura 6: La rete neurale di Elman sviluppata

L'algoritmo utilizzato durante la fase di addestramento per minimizzare la funzione di errore, è l'algoritmo *JE_RPROP* (*Jordan-Elman Resilent BackPROPagation*). Questo algoritmo, di tipo adattivo, si è dimostrato robusto e veloce nella convergenza dell'errore[2][3], grazie alla sua caratteristica di eliminare l'influenza negativa della dimensione della derivata parziale sul passo utilizzato per l'aggiornamento dei pesi della rete.

La dimensione per l'aggiornamento dei pesi Δw_{ij} , è decisa dal coefficiente Δ_{ij} , chiamato valore di aggiornamento.

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t-1)}, & \text{se } \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ +\Delta_{ij}^{(t-1)}, & \text{se } \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ 0, & \text{altrimenti} \end{cases} \quad (2)$$

Con i valori di aggiornamento calcolati come segue:

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^+ * \Delta_{ij}^{(t-1)}, & \text{se } \frac{\partial E^{(t-1)}}{\partial w_{ij}} * \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ \eta^- * \Delta_{ij}^{(t-1)}, & \text{se } \frac{\partial E^{(t-1)}}{\partial w_{ij}} * \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ \Delta_{ij}^{(t-1)}, & \text{altrimenti} \end{cases} \quad (3)$$

dove con $\frac{\partial E^{(t)}}{\partial w_{ij}}$ si indica il gradiente.

Ogni volta che la derivata parziale del peso w_{ij} , cambia di segno, significa che l'ultimo aggiornamento eseguito è stato troppo grande e l'algoritmo è saltato sopra un minimo locale, il valore di aggiornamento Δ_{ij} è decrementato di un fattore η^- (che nel software SNNS è impostato ad un valore fisso pari a 0.5). Se la derivata mantiene il relativo segno, allora il valore di aggiornamento è incrementato di un fattore η^+ (che nel software SNNS è impostato ad un valore fisso pari a 1.2), per accelerare la convergenza.

Per l'aggiornamento dei pesi è stata utilizzata la funzione *JE_Order*. Questa funzione di aggiornamento propaga l'insieme dei campioni, dallo strato di ingresso verso lo strato nascosto, e poi verso lo strato di uscita. Dopo questa fase, è eseguito un aggiornamento sincrono di tutte le unità di contesto.

Per evitare che la rete si specializzi sui campioni utilizzati per l'addestramento, quest'ultimo è stato eseguito in 200 epoche. Superando questo valore, la rete non riesce a generalizzare ed elabora una corretta relazione tra ingresso e uscita, causando un degrado delle prestazioni.

I parametri utilizzati per la fase di addestramento sono i seguenti:

- $\Delta_0=0.01$ indica il valore di partenza per tutti i Δ_{ij}
- $\Delta_{\max}=40$ indica il limite superiore per tutti i Δ_{ij}
- $\alpha=4$ indica il decadimento dei pesi
- $\mu=0.5$ indica il parametro speciale

Quest'ultimo parametro, è utilizzato per propagare il valore appreso, o parte di esso, all'interno della rete.

Per la fase di inizializzazione è stata utilizzata la funzione *JE_Weights*. In questo tipo di rete vi sono due tipologie di neuroni, quelli normali e quelli di contesto. Questa funzione di inizializzazione, richiede l'inserimento di cinque parametri:

- $\alpha=1, \beta=-1$ indicano gli estremi dell'intervallo in cui i pesi sono casualmente scelti
- $\lambda=0$ indica il peso dei collegamenti auto ricorrenti
- $\gamma=1$ indica il peso dei collegamenti ricorrenti per le unità di contesto
- $\psi=0.2$ indica il valore iniziale di attivazione delle unità di cointesto

In Figura 7, è mostrato l'andamento dell'errore durante l'addestramento con i parametri descritti in precedenza, dove la curva in nero mostra l'andamento dell'errore per i dati di addestramento, mentre la curva in rosso mostra l'andamento dell'errore per i dati di convalida dell'addestramento.

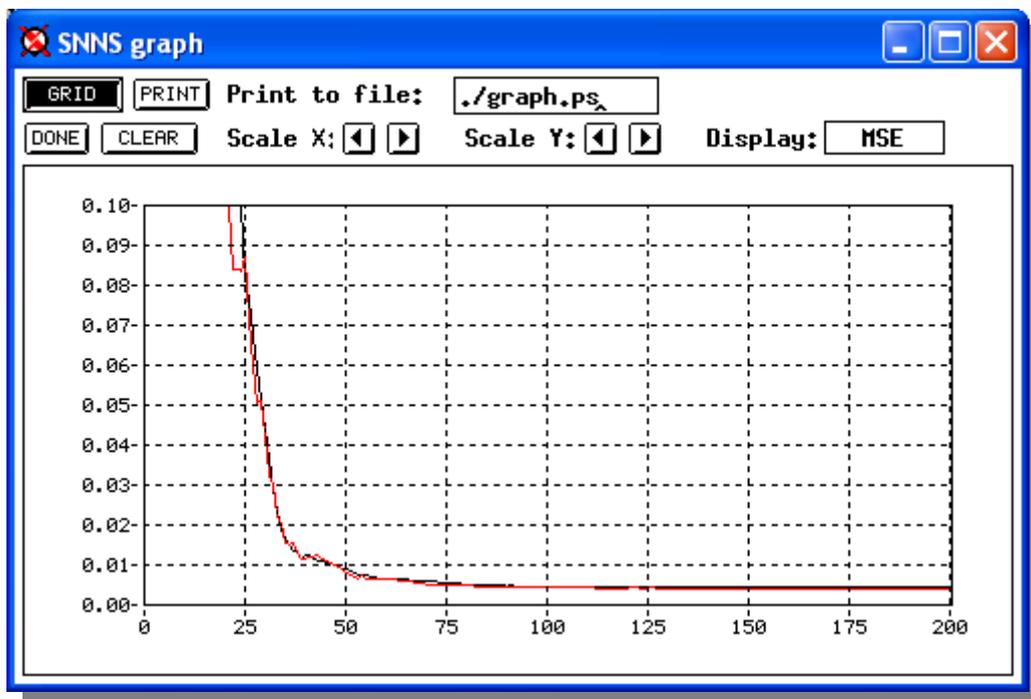


Figura 7: La rete neurale di Elman sviluppata

1.3.2 Risultati sperimentali

Di seguito saranno mostrati i risultati sperimentali ottenuti per ognuna delle sei stazioni meteorologiche precedentemente descritte.

1.3.2.1 Castellammare del Golfo

In Figura 8, è mostrato il grafico dell'andamento della temperatura massima prevista dalla rete neurale, confrontato con l'andamento reale.

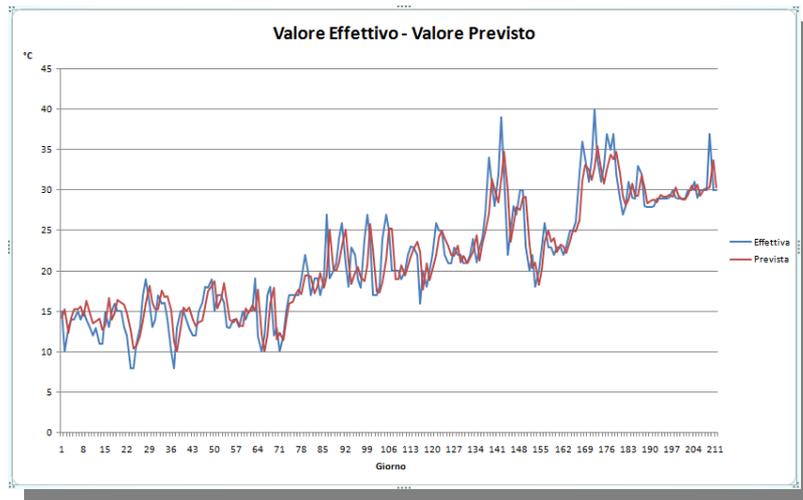


Figura 8: Il confronto tra il valore previsto e quello reale per la stazione di Castellammare

Dal grafico, è possibile notare come i valori previsti dalla rete, seguano l'andamento dei valori reali, tranne che in quei casi in cui vi è una brusca variazione di temperatura tra due giorni consecutivi. In Figura 9, è mostrata la distribuzione dell'errore assoluto.

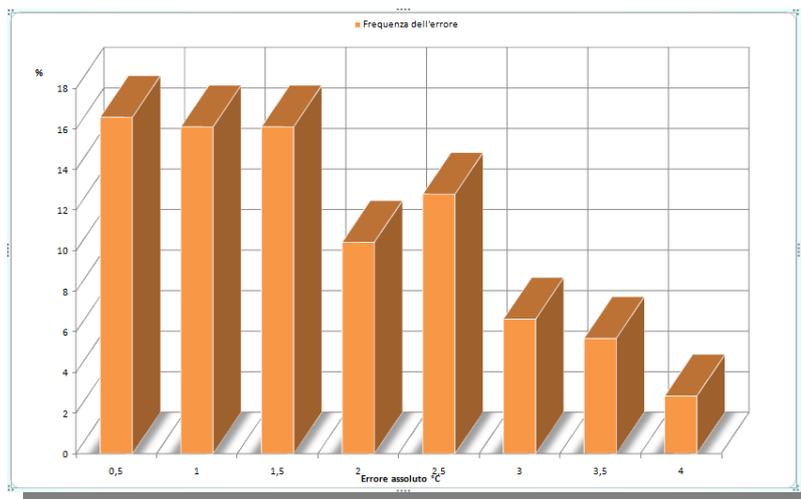


Figura 9: La frequenza dell'errore assoluto per la stazione di Castellammare

Dal grafico, si osserva che nel 60% dei casi, l'errore si mantiene sotto a 2 °C, e solo nel 2.8% dei casi arriva a 4 °C.

1.3.2.2 Castelvetrano

In Figura 10, è mostrato il grafico dell'andamento della temperatura massima prevista dalla rete neurale, confrontato con l'andamento reale.

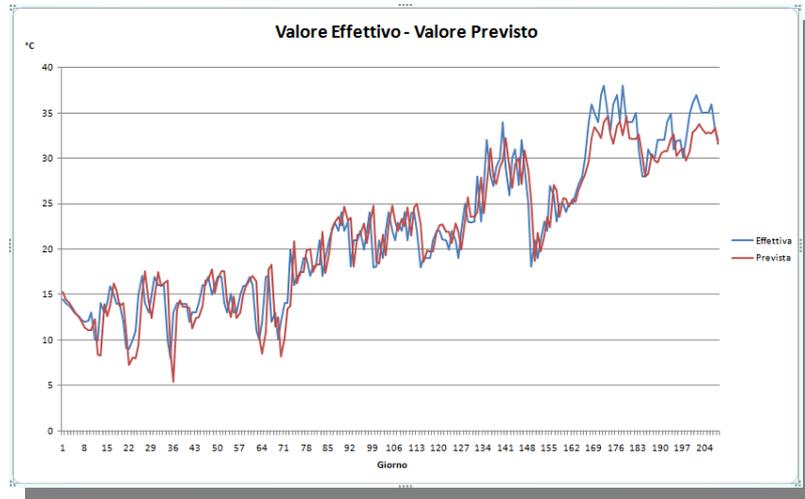


Figura 10: Il confronto tra il valore previsto e quello reale per la stazione di Castelvetrano

Analoghe considerazioni in riguardo alle brusche variazioni di temperatura, si possono fare anche in questo caso. In Figura 11, è mostrata la distribuzione dell'errore assoluto.

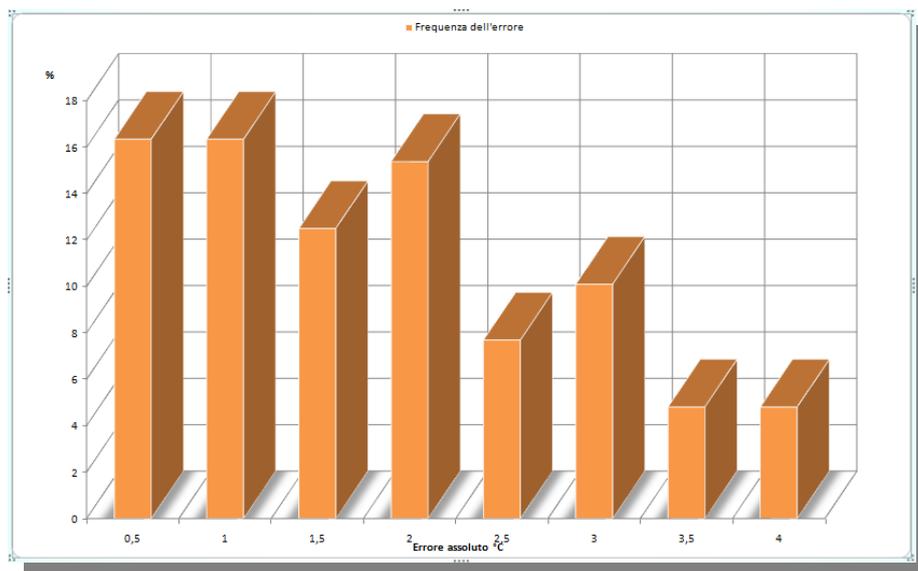


Figura 11: La frequenza dell'errore assoluto per la stazione di Castelvetrano

Dal grafico, si osserva che nel 61% dei casi, l'errore si mantiene sotto a 2 °C, e solo nel 4.8% dei casi arriva a 4 °C.

1.3.2.3 Erice

In Figura 12, è mostrato il grafico dell'andamento della temperatura massima prevista dalla rete neurale, confrontato con l'andamento reale.

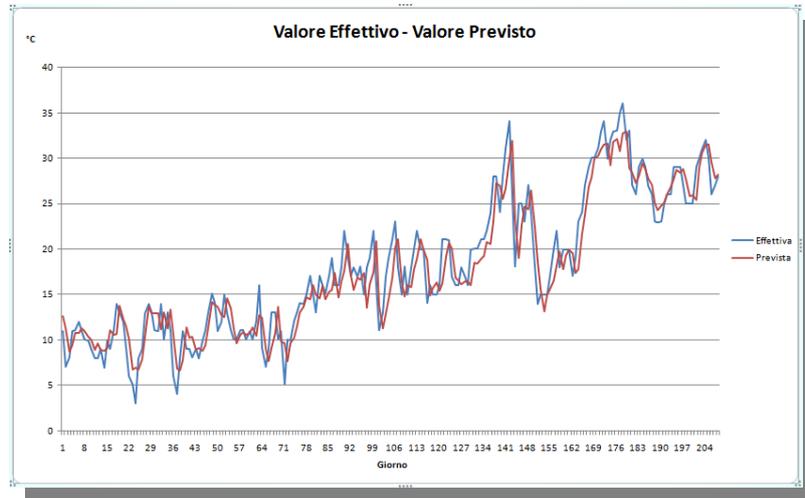


Figura 12: Il confronto tra il valore previsto e quello reale per la stazione di Erice

Analoghe considerazioni in riguardo alle brusche variazioni di temperatura, si possono fare anche in questo caso. In Figura 13, è mostrata la distribuzione dell'errore assoluto.

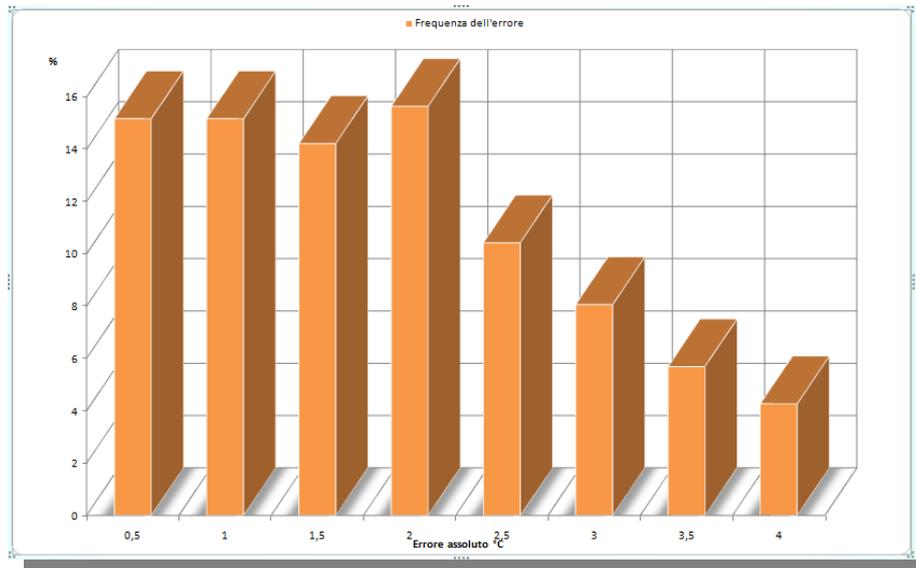


Figura 13: La frequenza dell'errore assoluto per la stazione di Erice

Dal grafico, si osserva che nel 60% dei casi, l'errore si mantiene sotto a 2 °C, e solo nel 4.3% dei casi arriva a 4 °C.

1.3.2.4 Marsala

In Figura 14, è mostrato il grafico dell'andamento della temperatura massima prevista dalla rete neurale, confrontato con l'andamento reale.

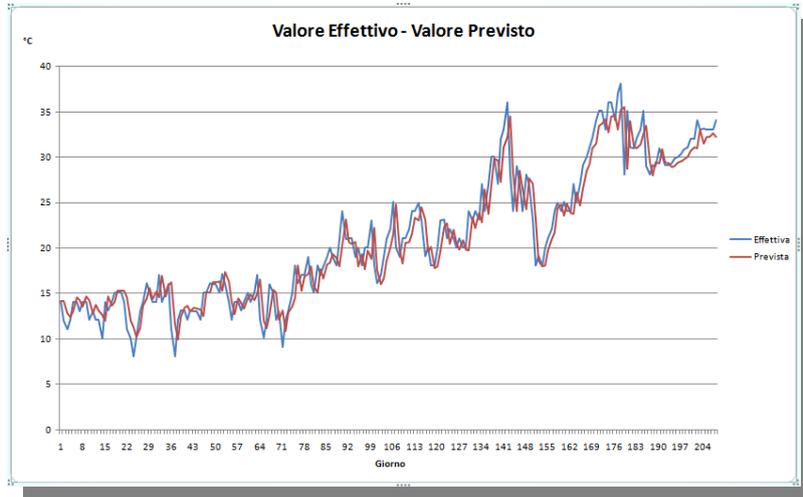


Figura 14: Il confronto tra il valore previsto e quello reale per la stazione di Marsala

Analoghe considerazioni in riguardo alle brusche variazioni di temperatura, si possono fare anche in questo caso. In Figura 15, è mostrata la distribuzione dell'errore assoluto.

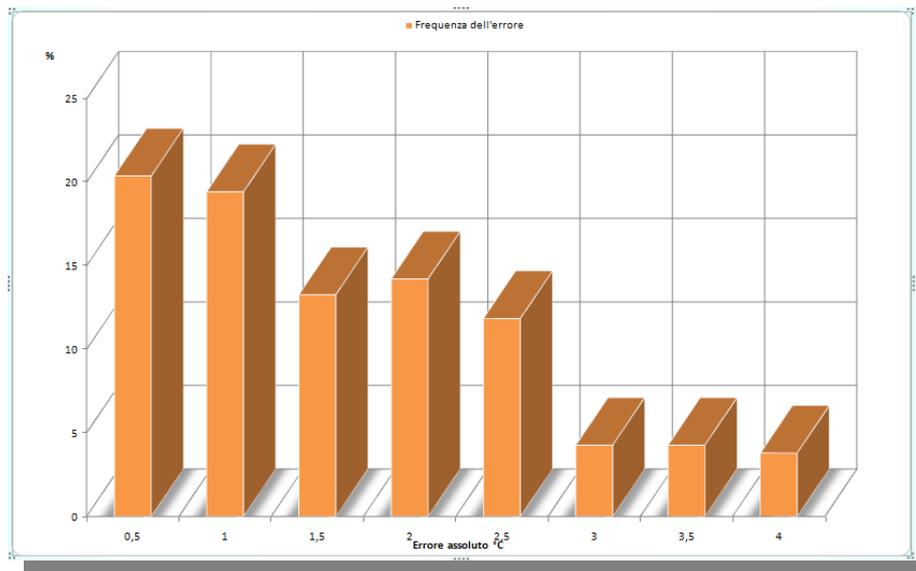


Figura 15: La frequenza dell'errore assoluto per la stazione di Marsala

Dal grafico, si osserva che nel 67% dei casi, l'errore si mantiene sotto a 2 °C, e solo nel 3,8% dei casi arriva a 4 °C.

1.3.2.5 Mazara del Vallo

In Figura 16, è mostrato il grafico dell'andamento della temperatura massima prevista dalla rete neurale, confrontato con l'andamento reale.

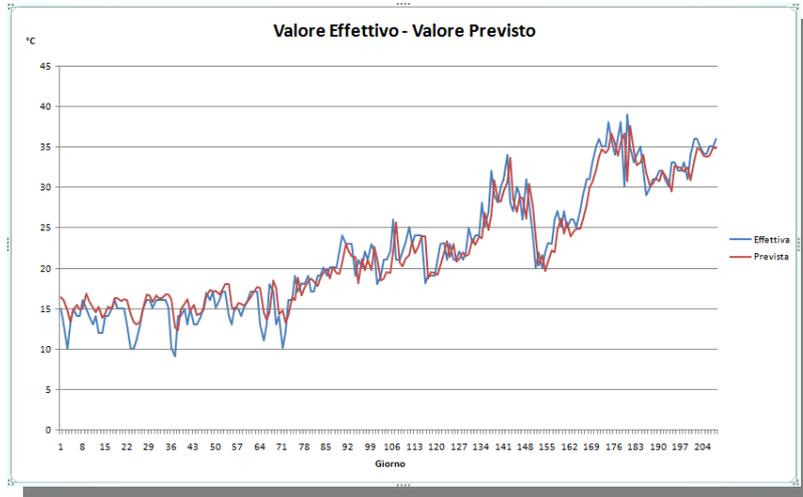


Figura 16: Il confronto tra il valore previsto e quello reale per la stazione di Mazara

Analoghe considerazioni in riguardo alle brusche variazioni di temperatura, si possono fare anche in questo caso. In Figura 17, è mostrata la distribuzione dell'errore assoluto.

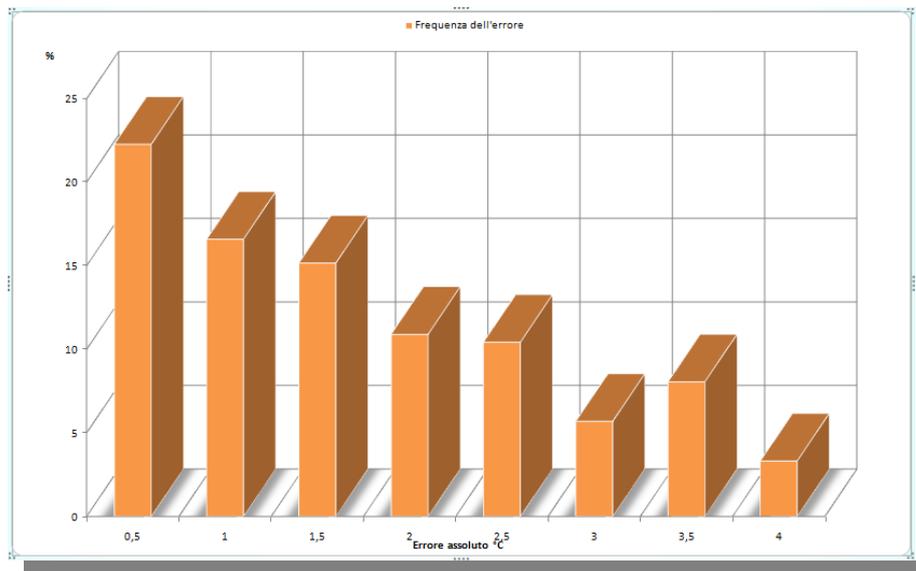


Figura 17: La frequenza dell'errore assoluto per la stazione di Mazara

Dal grafico, si osserva che nel 65% dei casi, l'errore si mantiene sotto a 2 °C, e solo nel 3.3% dei casi arriva a 4 °C.

1.3.2.6 Trapani

In Figura 18, è mostrato il grafico dell'andamento della temperatura massima prevista dalla rete neurale, confrontato con l'andamento reale.

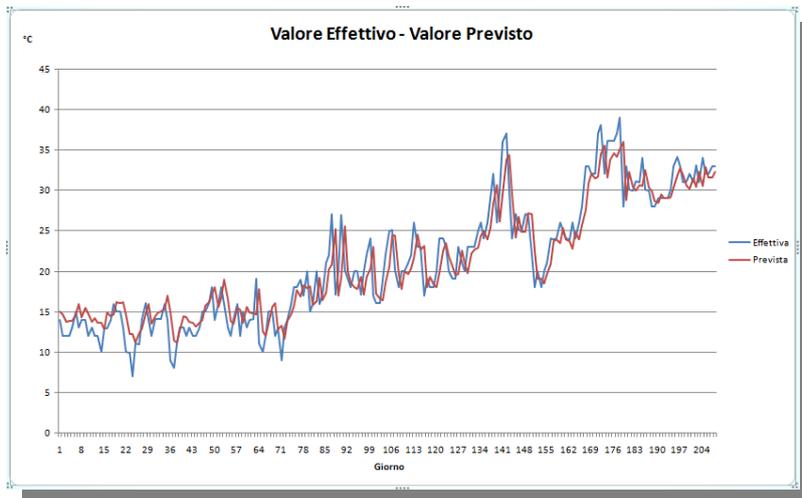


Figura 18: Il confronto tra il valore previsto e quello reale per la stazione di Trapani

Analoghe considerazioni in riguardo alle brusche variazioni di temperatura, si possono fare anche in questo caso. In Figura 19, è mostrata la distribuzione dell'errore assoluto.

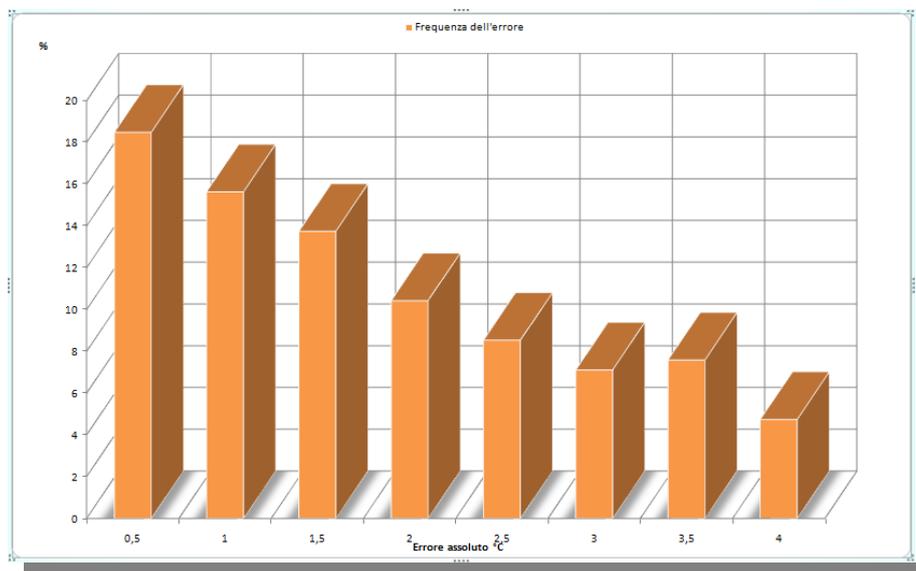


Figura 19: La frequenza dell'errore assoluto per la stazione di Trapani

Dal grafico, si osserva che nel 58% dei casi, l'errore si mantiene sotto a 2 °C, e solo nel 4.7% dei casi arriva a 4 °C.

1.3.3 Analisi della sensibilità

Come già spiegato in precedenza, le reti neurali artificiali hanno un funzionamento particolare, che talvolta è chiamato col nome di scatola nera. Proprio per questo tipo di funzionamento, è difficile capire le relazioni che legano gli ingressi con le uscite della rete, o quale tra i vari ingressi, influisce maggiormente sul valore dell'uscita.

L'analisi della sensibilità è eseguita per determinare appunto, quale tra i vari ingressi influisce maggiormente sull'uscita di una rete, fissando i valori di tutti gli ingressi, tranne che di quello di cui si vuole analizzare l'influenza.

Nelle seguenti figure, sono mostrati i grafici della sensibilità delle grandezze di ingresso per le sei stazioni meteorologiche. In questi grafici è possibile notare che l'ingresso che influisce maggiormente è proprio la Temperatura massima al tempo $t-1$.



Figura 20: I grafici della sensibilità per le stazioni di Castellammare e Castelvetro

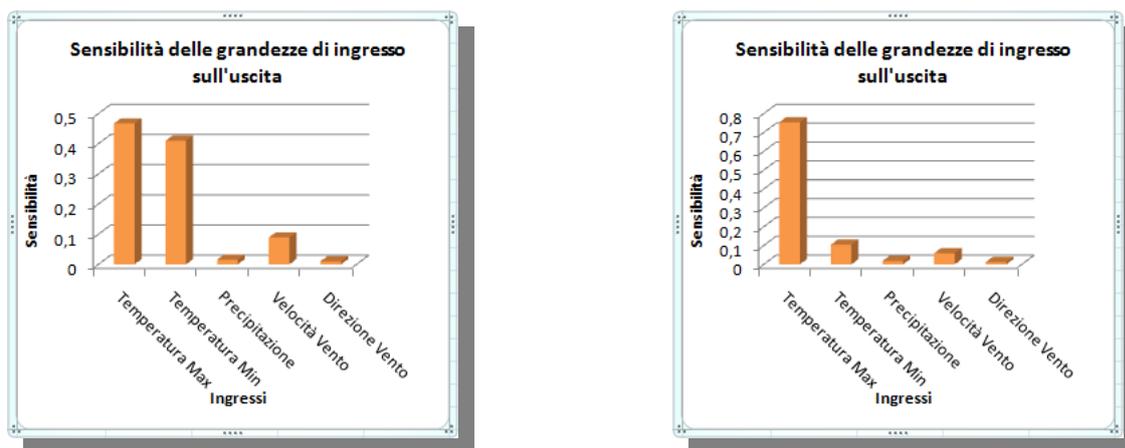


Figura 21: I grafici della sensibilità per le stazioni di Erice e Marsala

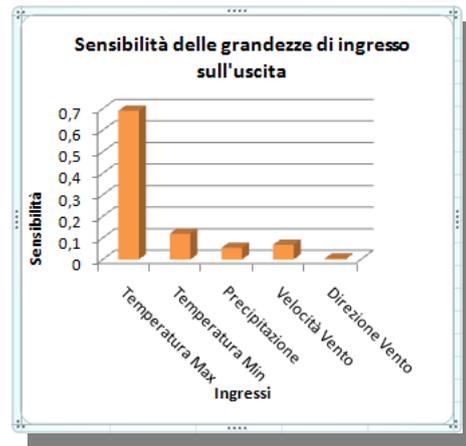
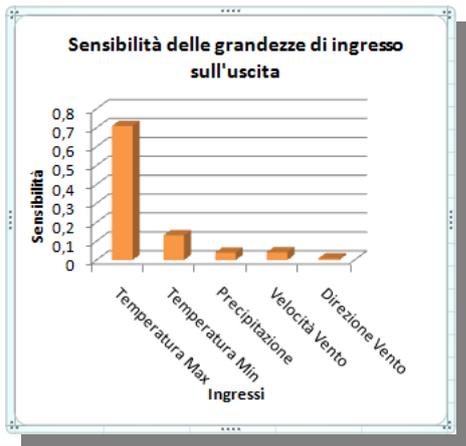


Figura 22: I grafici della sensibilità per le stazioni di Mazara del Vallo e Trapani

Da questi grafici è possibile notare che in tutte le sei stazioni meteorologiche, la grandezza che influisce maggiormente sull'uscita, è la temperatura massima a tempo $t-1$, ed un piccolo contributo sulla variazione è dato dagli altri parametri. Queste piccole variazioni non sono in egual misura per tutte le sei stazioni, ma variano a seconda della posizione geografica della stessa, in quanto alcune stazioni potrebbero trovarsi, ad esempio, più esposte al vento rispetto ad altre.

Un caso particolare è proprio quello della stazione meteorologica di Erice, che mostra come l'uscita è influenzata, quasi in egual misura, sia dalla temperatura massima, sia dalla temperatura minima. Questo potrebbe essere spiegato dal fatto che Erice si trova ad un'altitudine di circa 1000 m, e quindi, di notte, la temperatura minima scende a temperature più basse rispetto a quelle delle altre stazioni meteorologiche, provocando delle differenze di temperatura tra il giorno e la notte, a cui la rete neurale ricorrente dà molta importanza, durante la fase di addestramento, per l'elaborazione dei pesi e la successiva elaborazione della previsione al tempo t .

Nelle seguenti figure, è mostrato l'andamento dell'uscita in funzione di una grandezza, mantenendo fisse le altre.

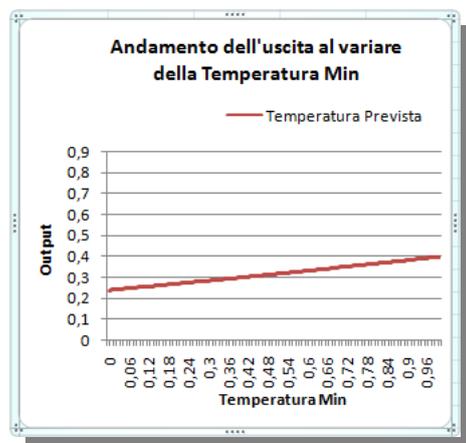
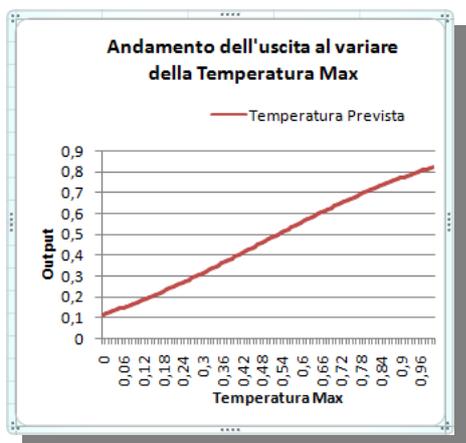


Figura 23: L'andamento dell'uscita al variare della temperatura massima e minima

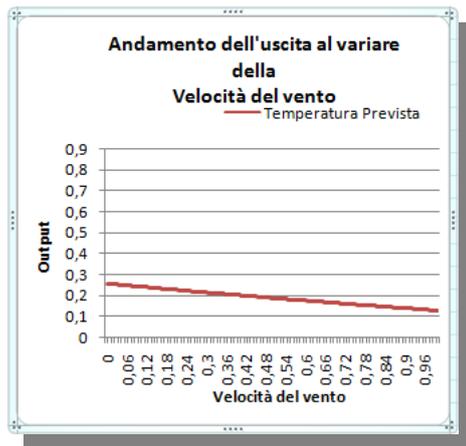
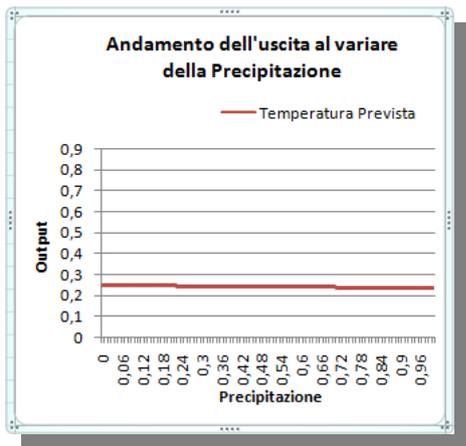


Figura 24: L'andamento dell'uscita al variare della precipitazione e della velocità e direzione del vento

2 Le reti bayesiane

Le reti bayesiane, note anche con il nome di reti probabilistiche o reti di credenze, sono dei modelli grafici di probabilità in cui i nodi sono un insieme di variabili aleatorie, e gli archi rappresentano le dipendenze causali fra le variabili.

Questo tipo di reti, sono utilizzate quando vi è la necessità di modellare la realtà in situazioni di incertezza, cioè quando sono in gioco le probabilità, e maggiormente, quando il verificarsi di un dato evento, può dare informazioni, in senso probabilistico, sul verificarsi di un altro evento che, in qualche modo, risulta essere correlato con il primo.

Le reti bayesiane si basano sul *Teorema di Bayes*, che fornisce un metodo per modificare il livello di conoscenza di una data ipotesi, alla luce di nuove informazioni[11].

Denotando con H_0 l'ipotesi nulla, e con E il dato empirico osservato, il teorema di Bayes può essere enunciato come:

$$P(H_0|A) = \frac{P(E|H_0)P(H_0)}{P(E)} \quad (4)$$

dove indichiamo con

- $P(H_0)$ la probabilità a priori
- $P(E | H_0)$ la funzione di verosimiglianza
- $P(E)$ la probabilità marginale
- $P(H_0 | E)$ la probabilità a posteriori di H_0 , dato E

Questa equazione è alla base di tutti i sistemi di Intelligenza Artificiale per l'inferenza probabilistica, poiché permette di esprimere l'indipendenza condizionale tra variabili, senza ricorrere ad un'estesa (e quindi difficilmente gestibile) tabella di probabilità congiunte, semplificando notevolmente il calcolo dei risultati delle interrogazioni.

Tramite questo teorema, è dunque possibile determinare la probabilità condizionata $P(A | B)$, che esprime la probabilità dell'ipotesi A , una volta osservata l'evidenza B .

In ambito del progetto *DrepanCoast*, sono stati raccolti i dati per diverse categorie di indicatori, quali ambientale, agricolo, peschiero, economico ed altri, ma anche se le rilevazioni riguardano un periodo di sei anni (da inizio 2000 a fine 2005), i dati raccolti, non potranno mai dare un'informazione completa in merito al calcolo di previsioni future. Se, infatti, tirando un dado, per dieci volte consecutive esce sempre la stessa faccia, non vuol dire che si ha la certezza che ogni qual volta si tirerà il dado, uscirà sempre la stessa faccia, anzi avrà sempre un grado di incertezza, dato dagli eventi inattesi. Questa è l'incertezza che le reti bayesiane, cercheranno di colmare tramite i meccanismi di inferenza probabilistica.

Il progetto *MIGET*, ha anche lo scopo di sviluppare dei modelli intelligenti basati sulle reti bayesiane, per l'analisi delle relazioni tra gli indicatori delle unità fisiografiche

paesaggistiche (UFP) della provincia di Trapani, cercando di prevedere il valore di alcuni indicatori, in relazione ai valori conosciuti di altri (evidenze).

2.1 Il modello bayesiano sviluppato

Per sviluppare il modello bayesiano, occorre inizialmente determinare la struttura della rete e, in seguito, procedere alla creazione delle *Tablette di probabilità condizionata (TPC)*.

La rete bayesiana, come indicato in precedenza, è un grafo orientato e aciclico, in cui i nodi sono le variabili e gli archi rappresentano le relazioni di dipendenza probabilistica tra i nodi. Per determinare la struttura della rete, occorre dunque identificare le variabili più importanti e, una volta determinati i valori che possono assumere, procedere con il collegamento dei vari nodi, definendo così le relazioni tra i nodi stessi.

Per lo sviluppo della rete, è stato utilizzato il software *Norsys Netica v3.19[15]*, che permette di creare la rete e addestrarla, in maniera molto semplice e veloce.

Poiché i dati raccolti per il progetto *DrepanCoast*, si riferivano a tredici unità paesaggistiche fisiografiche, per un periodo di sei anni, e che alcune tipologie di dati erano suddivise per stagione, è stato naturale assegnare una variabile per ogni tipo di indicatore da calcolare, una variabile per la stagione dell'anno a cui si riferiscono i dati, e una variabile per la zona di riferimento, per un totale di diciassette nodi.

2.2 Descrizione dell'insieme dei dati

Una volta identificate le variabili, e dunque i nodi della rete, occorre individuare gli stati, cioè i valori che ognuna di esse può assumere.

Gli indicatori calcolati nel progetto *DrepanCoast*, assumono valori continui compresi nell'intervallo tra 0 e 10, ma spesso, nei modelli bayesiani, risulta conveniente ridurre il numero di stati che un nodo può assumere. Tramite un'operazione di raggruppamento, il numero di stati è stato ridotto a tre, classificando il valore che l'indicatore può assumere in uno dei tre stati mostrato in Tabella 3.

Valore Indicatore	Stato
0 => 3,33	Low
3,33 => 6,66	Medium
6,66 => 10	High

Tabella 3: Gli stati dei nodi Indicatore

I nodi riguardanti gli indicatori sono quindici, e sono elencati nella Tabella 4, con i relativi stati che possono assumere.

Nodo	Stato ammissibile
Indice di densità di popolazione (Idp)	Low, medium, High
Indice di occupazione costa (Ioc)	Low, medium, High
Indice di sostenibilità edilizia (Ise)	Low, medium, High
Indice di incidenza turistica (Iit)	Low, medium, High
Indice antropico (Ian)	Low, medium, High
Indice delle infrastrutture viarie (Iiv)	Low, medium, High
Indice di balneabilità (Ib)	Low, medium, High
Indice di cattura pesca (Icp)	Low, medium, High
Indice della densità natanti (Idn)	Low, medium, High
Indice di pescosità (Ip)	Low, medium, High
Indice naturalistico (In)	Low, medium, High
Indice di utilizzazione agricola (Iua)	Low, medium, High
Indice di sostenibilità ricettiva (Isr)	Low, medium, High
Indice economico (Ie)	Low, medium, High
Indice di sostenibilità costiera (Isc)	Low, medium, High

Tabella 4: I nodi degli indicatori e i possibili stati ammissibili

Per quanto riguarda il periodo dell'anno, è stato creato un nodo, chiamato Stagione, che può assumere come valore uno delle quattro stagioni dell'anno, Primavera, Estate, Autunno e Inverno.

In Tabella 5, è mostrato il nodo della zona fisiografica, in cui è possibile vedere l'elenco delle unità fisiografiche paesaggistiche oggetto di studio.

Nodo	Stato ammissibile
Zone fisiografiche (UFP)	Alcamo Marina - Castellammare Zingaro San Vito Lo Capo San Vito - Pizzolungo Trapani Saline Isole Egadi e Pantelleria Marsala Lido Signorino Margi Mazara Capo Granitola Le Dune

Tabella 5: Il nodo delle zone fisiografiche (UFP)

Per determinare i collegamenti tra i vari nodi, sono stati fatti vari tentativi, cercando di sviluppare una rete secondo un certo livello di gerarchia.

Come si può vedere in Figura 25, si parte dal livello in cui sono fornite le sole informazioni riguardanti il periodo dell'anno e la zona di riferimento, poi si passa al livello in cui i dati sono stati elaborati e classificati secondo la tipologia, ed infine si arriva agli ultimi due livelli che riguardano gli indicatori calcolati. In particolare, l'ultimo livello si riferisce al solo *Indice di sostenibilità costiera*, che ingloba le informazioni degli altri indicatori, dando una valutazione globale per la zona indicata e per il dato periodo dell'anno.

Dopo aver determinato la struttura della rete, si passa alla fase di addestramento della stessa. Per questa fase, i dati sono stati preparati in forma tabellare, come mostrato in Tabella 6, per poterne facilitare l'utilizzo con il software *Netica*.

stagioni	UFP	Idp	Ioc	Ise	Iit	Ian	Ib	Icp	Idn	Ip	In	Isr	Iua	Ie	Iiv	Isc
Primavera	AlcamoMarinaCastellammare	10,0000	10,0000	8,9682	9,7555	9,6809	8,4277	0,9016	7,7694	4,3355	0,0000	4,7062	0,4308	2,5685	8,7579	6,1975
Primavera	Zingaro	3,0426	10,0000	9,9127	9,8121	8,1919	9,6364	2,1061	10,0000	6,0531	3,8385	3,2931	0,3262	1,8096	8,8087	6,6564
Primavera	SanVitoLoCapo	1,9567	9,5678	9,3697	9,8227	7,6792	7,3907	2,0140	0,0378	1,0259	0,0000	0,8120	0,0894	0,4507	8,9969	4,3487
Primavera	SanVitoPizzolungo	3,5617	10,0000	9,8369	9,7849	8,2959	9,7001	0,8439	10,0000	5,4220	0,8336	5,0000	0,5016	2,7508	8,8701	6,5349
Primavera	Trapani	3,9203	10,0000	5,4442	9,7878	7,2881	3,4362	4,8919	1,5699	3,2309	0,0000	2,2259	0,0000	1,1130	9,0357	4,2303
Primavera	Saline	5,1226	10,0000	9,5588	9,8225	8,6260	8,4356	0,0242	10,0000	5,0121	5,3386	9,1928	1,3339	5,2633	8,7337	6,8595
Primavera	IsoleEgadiEPantelleria	10,0000	10,0000	9,6520	9,3936	9,7614	9,4257	2,0724	10,0000	6,0362	3,4230	5,0000	1,3158	3,1579	8,8443	6,9745
Primavera	Marsala	4,7961	10,0000	8,6192	9,8121	8,3069	6,0475	0,6218	4,2435	2,4326	0,0000	8,8665	0,0000	4,4333	8,6022	5,4623
Primavera	LidoSignorino	4,8945	10,0000	9,2792	9,8121	8,4965	9,9986	1,3096	10,0000	5,6548	0,0000	5,2029	0,0000	2,6015	8,6843	6,6688
Primavera	Margi	10,0000	10,0000	9,8331	9,0308	9,7160	8,7261	0,9850	9,8952	5,4401	0,0000	8,6838	0,0000	4,3419	8,7830	6,9641
Primavera	Mazara	10,0000	10,0000	8,1511	9,0308	9,2955	5,0452	3,5876	2,3665	2,9771	0,0000	4,7734	0,0000	2,3867	8,8881	5,0646
Primavera	CapoGranitola	10,0000	10,0000	9,8828	8,6675	9,6376	8,8557	4,3079	10,0000	7,1539	0,0000	2,2654	0,0000	1,1327	8,7453	6,6002
Primavera	LeDune	10,0000	10,0000	9,4759	8,4559	9,4829	9,6309	0,0000	9,4976	4,7488	0,5704	1,6348	0,0000	0,8174	8,5793	6,0271

Tabella 6: I dati per l'addestramento

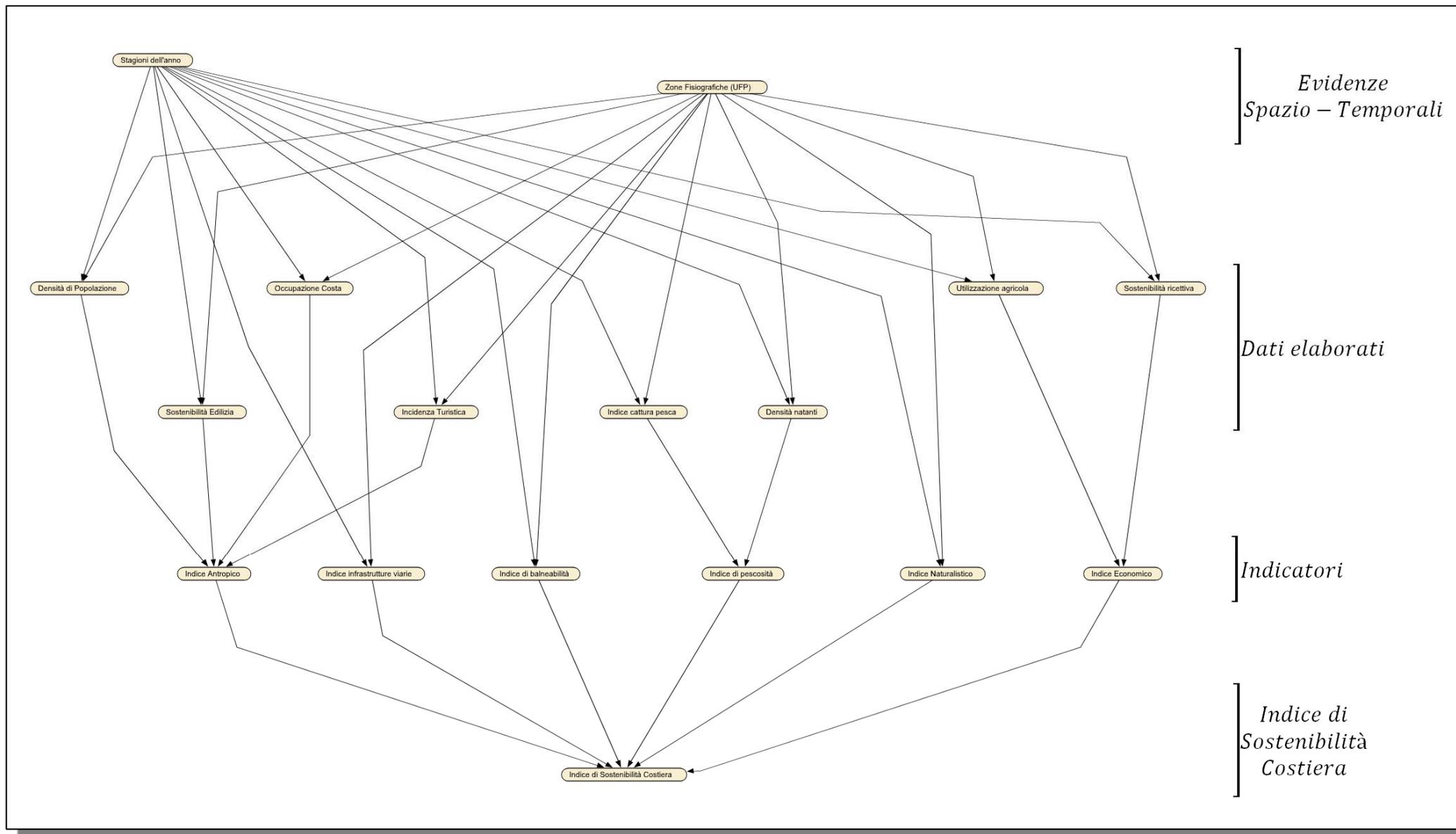


Figura 25: Il modello della rete bayesiana sviluppato

La serie storica, fa riferimento al periodo che va dall’inizio dell’anno 2000, fino alla fine dell’anno 2005, ed è stata divisa in due insiemi:

- Dal 2000 al 2004 per una fase di addestramento
- Il solo anno 2005 per una verifica dei risultati

Dopo che i risultati sono stati verificati, la rete è stata addestrata nuovamente con l’intero periodo storico, in modo da colmare le eventuali lacune probabilistiche che la rete potesse avere. Alla fine della fase di addestramento, le tabelle di probabilità condizionata sono state aggiornate, ma sarebbero necessari ulteriori casi di addestramento per poter avere più informazioni probabilistiche. Nei casi in cui le variabili assumano una distribuzione di probabilità uniforme (nessuna informazione), la rete non sarebbe in grado di eseguire una corretta inferenza, proponendo in uscita dei risultati che non sarebbero corretti.

In Tabella 7, è mostrata la tabella di probabilità condizionata per il nodo che si riferisce all’indice di sostenibilità costiera.

Indice Antropico	Indice di balneabil...	Indice di pescosità	Indice Naturalistico	Indice Economico	Indice infrastrutture via...	Low	Medium	High
Low	Low	Medium	Low	High	Low	39.202	22.619	38.179
Low	Low	Medium	Low	High	Medium	72.692	24.360	2.948
Low	Low	Medium	Low	High	High	23.910	23.408	52.683
Low	Low	Medium	Medium	Low	Low	6.496	25.341	68.163
Low	Low	Medium	Medium	Low	Medium	65.101	18.379	16.520
Low	Low	Medium	Medium	Low	High	35.209	22.431	42.360
Low	Low	Medium	Medium	Medium	Low	16.883	27.720	55.398
Low	Low	Medium	Medium	Medium	Medium	9.319	55.531	35.150
Low	Low	Medium	Medium	Medium	High	34.921	44.847	20.232

Tabella 7: La tabella di probabilità condizionata

2.3 Risultati sperimentali

Dopo che la rete è stata addestrata, è possibile utilizzarla applicandola ad un caso pratico. Ogni nodo di cui è noto lo stato, sarà indicato come un’evidenza che, tramite il software *Netica*, permetterà alla rete di eseguire le inferenze probabilistiche, e calcolare il valore per tutti i restanti nodi della rete.

In Figura 26, è mostrato un esempio in cui, impostando le evidenze per i nodi *Stagione*, *Zona Fisiografica*, *Densità di popolazione*, *Occupazione costa*, *Indice di infrastrutture viarie*, *Indice di balneabilità*, *Densità natanti*, *Indice di cattura pesca* e *Indice naturalistico*, tramite l’inferenza probabilistica vengono calcolati i valori dei restanti nodi. In particolare, si nota che con il 66.3% di probabilità, l’indice di sostenibilità costiera avrà un valore alto.

In Figura 27, è mostrato un altro esempio in cui fissando gli stati per i nodi *Stagione*, *Zona Fisiografica*, *Densità di popolazione*, *Sostenibilità edilizia*, *Indice di infrastrutture viarie*, *Indice di balneabilità*, *Indice di cattura pesca*, *Indice naturalistico* e *Sostenibilità ricettiva*, con il 67.4% di probabilità, l’indice di sostenibilità costiera avrà un valore medio.

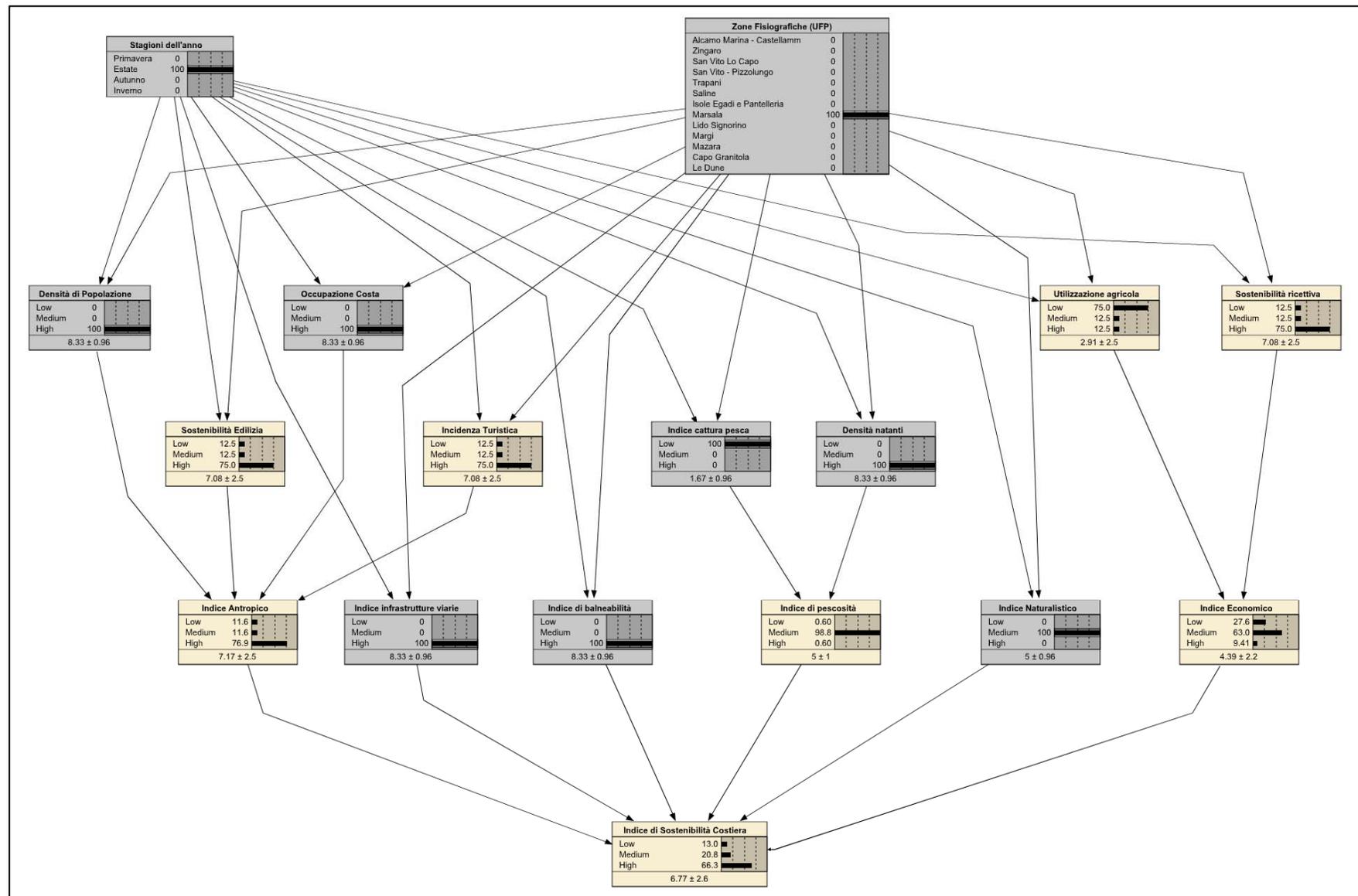


Figura 26: Un esempio di inferenza probabilistica

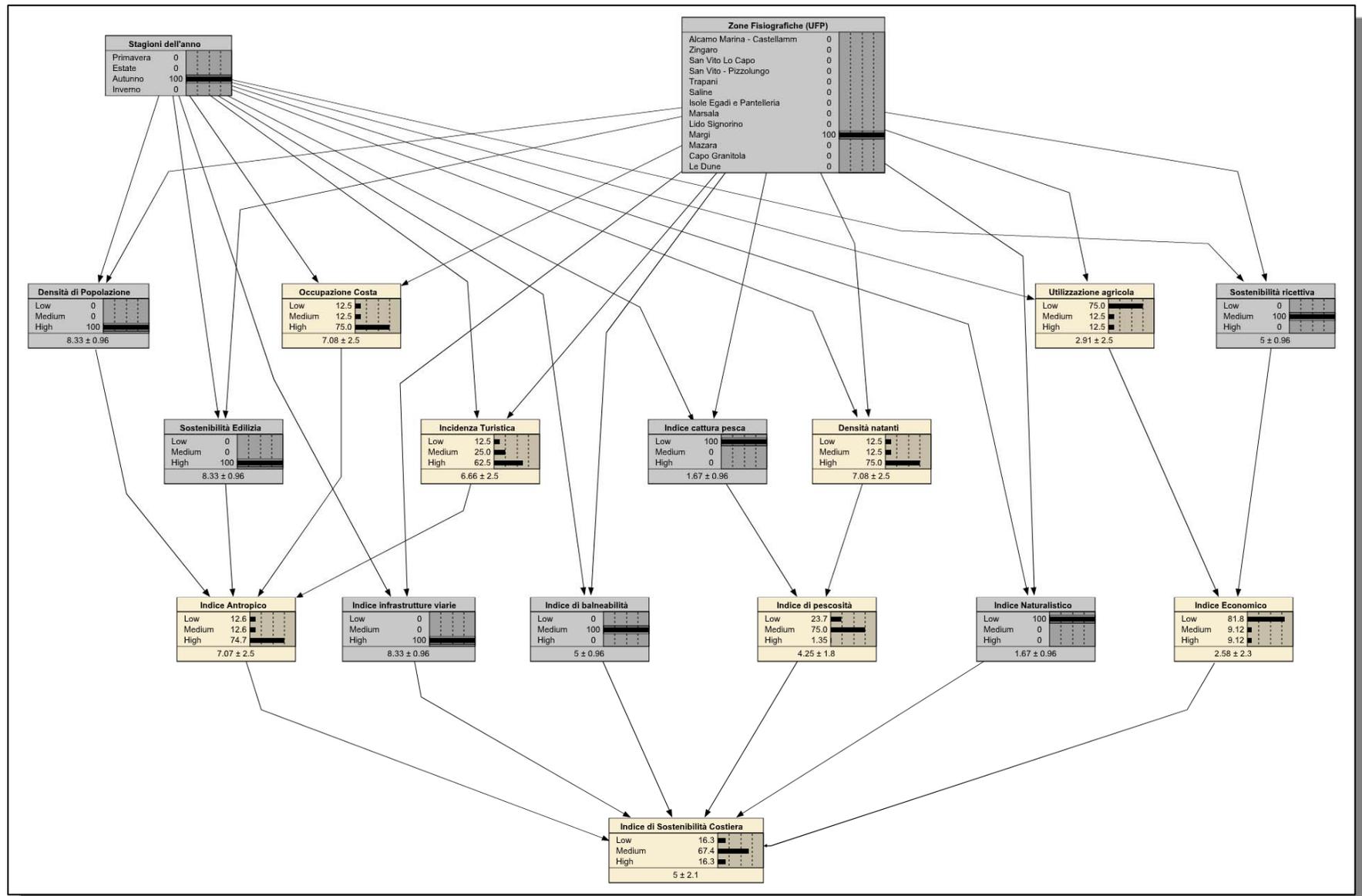


Figura 27: Un altro esempio di inferenza probabilistica

3 Integrazione dei modelli intelligenti nel framework ArcGIS 9.2: studio di fattibilità

3.1 Obiettivi generali

Il progetto si propone l'obiettivo di fornire un sistema informatico per la provincia di Trapani in relazione al progetto *DrepanCoast*. Il committente richiede l'integrazione dei risultati di previsione, ottenuti dai modelli intelligenti sviluppati nell'ambito del progetto *MIGET*, nel software per la gestione del sistema informativo geografico *ESRI ArcGIS 9 Desktop*. Per soddisfare le necessità di tutti gli utenti che utilizzeranno i supporti software del sistema, lo sviluppatore, in accordo con il committente, si propone di progettare un sistema sicuro, affidabile, facile da utilizzare e dotato di una chiara e completa documentazione.

L'integrazione proposta in questo progetto si fonda sull'idea di poter fornire all'utente finale, i dati di previsione elaborati dai modelli intelligenti sviluppati in ambito del progetto *MIGET*. I modelli intelligenti sviluppati per il progetto *MIGET* utilizzano tecniche di Intelligenza Artificiale, ed in particolare si basano sulle Reti Neurali Artificiali (RNA) e sulle Reti Bayesiane (RB). Entrambe le tipologie di reti sono dei grafi caratterizzate da nodi e connesse da archi.

Le Reti Neurali Artificiali sono state utilizzate per effettuare una previsione della temperatura del giorno successivo, analizzando i dati storici provenienti dalle rilevazioni effettuate su sei stazioni meteorologiche posizionate sul territorio della provincia di Trapani. La serie storica è composta dai seguenti dati meteorologici: la temperatura massima e minima della giornata, la piovosità, la velocità e la direzione del vento.

Per lo sviluppo dell'architettura della rete neurale artificiale, è stato utilizzato il simulatore *SNNS v4.2* (Stuttgart Neural Network Simulator), operante sia in ambiente Microsoft Windows, sia in ambiente Linux.

Le Reti di Bayesiane, note anche come reti di credenze o probabilistiche, sono state utilizzate per calcolare, in termini probabilistici, il valore di indicatori che rappresentano le parti dinamiche del sistema costiero della provincia di Trapani, quali ad esempio l'indicatore antropico (Ian), l'indice di pescosità (Ip), l'indice naturalistico (In) e altri indici, tra cui l'indice di sostenibilità costiera (Isc), che è l'indice globale. Questi indicatori sono stati calcolati sulla base dei dati storici forniti e fanno riferimento ad ognuna delle tredici Unità Fisiografiche Paesaggistiche (UFP). Queste UFP sono delle aree caratterizzate dal fatto che ciascuna di esse ha caratteristiche paesaggistiche, geologiche, morfologiche, climatiche omogenee. L'utilizzo di queste UFP, rende più facile lo studio delle caratteristiche territoriali e ambientali, e quindi di poterne prevedere l'andamento tramite l'analisi di opportuni indici e la rappresentazione su di un sistema GIS.

Per lo sviluppo della architettura della rete bayesiana, è stato utilizzato il software *Netica v3.19*, sviluppato dalla Norsys Software Incorp.

3.2 Il sistema ArcGIS

L'integrazione delle previsioni ottenute dai due modelli intelligenti, dovrebbe essere effettuata sul software *ESRI ArcGIS 9 Desktop*.

ArcGIS 9, la cui architettura è mostrata in Figura 28, è una famiglia di software integrati per la gestione del sistema informativo geografico. Esso si basa su una libreria di componenti software GIS condivisi, chiamata *ArcObject*. La ESRI fornisce un insieme di strumenti di sviluppo, denominato *ArcGIS Developer Kit*, che consente di creare applicazioni indipendenti, oppure di estendere le funzionalità dei software già esistenti come *ArcMap*.

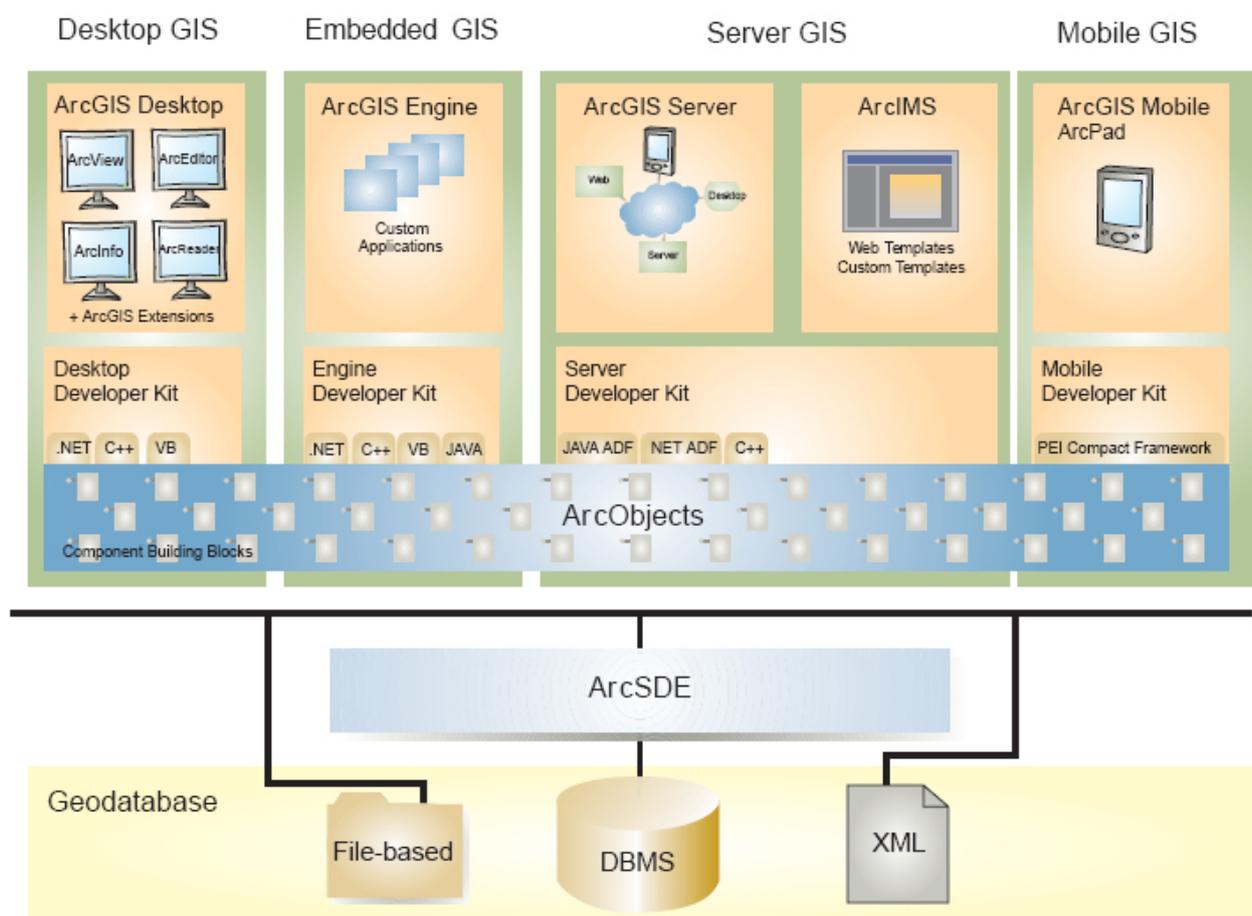


Figura 28: Il sistema ArcGIS 9

In entrambi i casi, occorre interfacciarsi, come mostrato in Figura 29, con questa libreria di componenti *ArcObject*.

Nel caso della creazione di un'applicazione indipendente, occorre utilizzare il prodotto *ESRI ArcGIS Engine* e i suoi strumenti di sviluppo, che permettono di utilizzare dei componenti già esistenti, per la creazione di soluzioni GIS mirate ad obiettivi specifici in base alle proprie esigenze. Gli ambienti di programmazione utilizzati per questa soluzione sono la piattaforma *.Net*, *Visual Basic*, *C++* e *Java*.

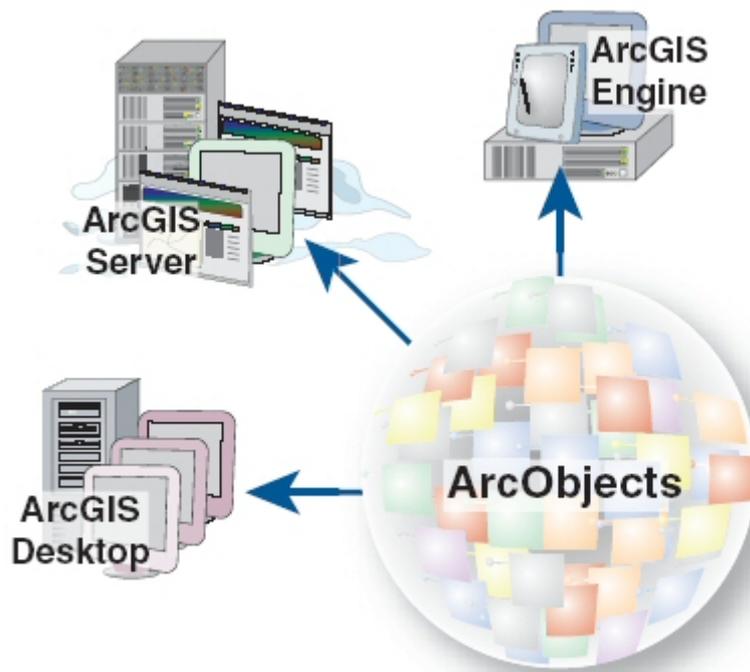


Figura 29: L'interazione tra sistemi ArcGIS e ArcObject

Per estendere le funzionalità del software *ESRI ArcGIS 9 Desktop*, occorre creare comandi personalizzati o uno strumento personalizzato. Un comando è un'operazione eseguita in seguito alla pressione su un bottone, lo strumento permette una maggiore interattività con l'utente finale, dando ad esempio la possibilità di modificare determinati parametri.

Per poter creare questi strumenti personalizzati, occorre utilizzare un linguaggio di programmazione capace di supportare la tecnologia Microsoft Component Object Model (COM) e distribuire lo strumento sottoforma di componente ActiveX (OCX) o Dynamic Link Library (DLL).

Alcuni linguaggi di programmazione che supportano le suddette caratteristiche sono *Visual Basic for Application (VBA)*, *Visual Basic .Net*, *C++*.

3.3 I modelli intelligenti sviluppati

Per lo sviluppo del modello della rete neurale ricorrente, è stato utilizzato il software SNNS v4.2 (Stuttgart Neural Network Simulator), mentre per lo sviluppo del modello della rete probabilistica bayesiana, è stato utilizzato il software Norsys Netica v3.19.

Entrambi i software permettono l'interfacciamento con un linguaggio di programmazione, in modo da poter utilizzare i risultati ottenuti, in un'applicazione esterna creata per eseguire compiti specifici.

Il software SNNS, operante in ambiente Microsoft Windows e Linux, è un'applicazione che può essere utilizzata tramite riga di comando o interfaccia grafica, che permette di creare la rete neurale ed addestrarla. Con questo software viene fornito anche uno strumento a riga di comando (snns2c), che offre la possibilità di esportare una rete neurale, creata con SNNS, in codice sorgente per il linguaggio C, come è mostrato in Figura 30.

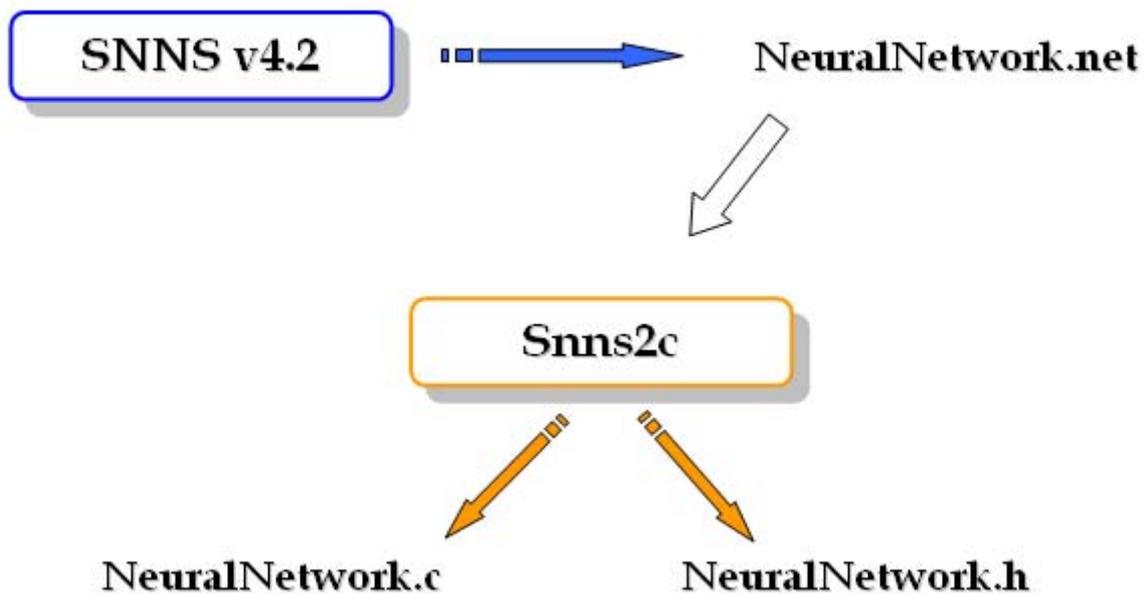


Figura 30: L'esportazione con lo strumento `ssns2c`

Questi sorgenti possono essere utilizzati per integrare la rete neurale addestrata, in un'applicazione indipendente.

Il software *Netica v3.19*, operante in ambiente Microsoft Windows e Mac OS, è un'applicazione con interfaccia grafica, che permette di creare ed addestrate una rete bayesiana, o rete di credenze. La rete può essere creata, addestrata ed effettuare un'inferenza probabilistica, sia tramite l'interfaccia grafica di *Netica v3.19*, sia tramite un programma scritto appositamente in uno dei linguaggi supportati. A questo scopo infatti, è possibile utilizzare le librerie che permettono di creare codice sorgente per i diversi linguaggi di programmazione, come *JAVA, C, C++, C#, Visual Basic*.

In Figura 31, è mostrato uno schema in cui una rete bayesiana è creata con *Netica v3.19*, ed in seguito, tramite l'interazione tra la libreria *Netica API* e l'ambiente di sviluppo Microsoft Visual C++, è possibile creare del codice sorgente che permetta di effettuare un'inferenza probabilistica sulla rete creata precedentemente.

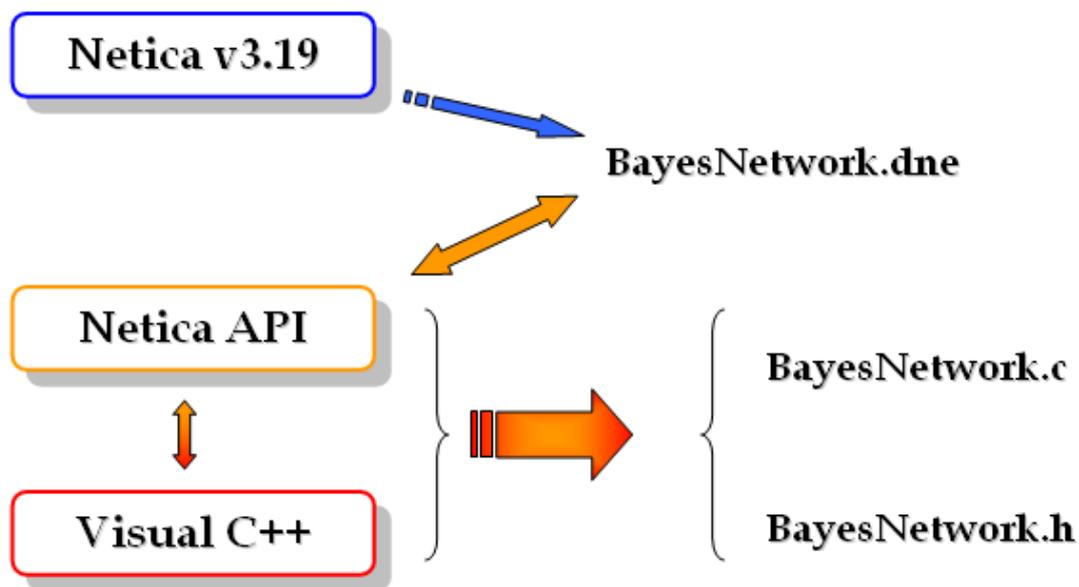


Figura 31: L'interazione delle librerie con l'ambiente Visual C++

3.4 Strategie di integrazione

Di seguito verrà mostrata in maniera più dettagliata, la soluzione proposta per permettere l'interazione tra i vari sistemi, e come effettivamente è possibile realizzarla. Verrà anche mostrata una valida alternativa a questa soluzione.

3.4.1 Estensione delle funzionalità di ArcGIS

L'implementazione di un'estensione, risulta essere la modalità più veloce per effettuare l'integrazione dei sistemi sopra descritti, in quanto uno dei sistemi è già pronto per poter ospitare dei componenti esterni. In questo caso dunque, l'unico sforzo implementativo riguarda la creazione del componente esterno che deve essere integrato.

Per come richiesto dal committente, l'applicazione in cui deve essere possibile utilizzare le previsioni calcolate dai modelli intelligenti, deve essere una delle applicazioni del sistema *ESRI ArcGIS Desktop*, ed in particolare l'applicazione *ArcMap*.

L'estensione da sviluppare sarà composta da una barra degli strumenti contenente due pulsanti, uno per la previsione effettuata dal software SNNS, e l'altro per la previsione effettuata dal software Netica. Questa barra degli strumenti verrà caricata nel software *ArcMap* e integrata nell'interfaccia come una qualsiasi altra barra degli strumenti già facente parte del software originale. Agendo su uno dei due pulsanti verrà mostrata una finestra di dialogo dove sarà permesso all'utente finale di impostare i vari parametri, diversi a seconda della previsione da effettuare, e visualizzare i risultati dell'elaborazione.

I componenti *ArcObject*, sono basati sulla tecnologia *Microsoft Component Object Model (COM)*, che è una piattaforma per componenti software che permette la comunicazione tra processi e la creazione dinamica di oggetti con qualsiasi linguaggio di programmazione

che supporta questa tecnologia. Questa infatti definisce un protocollo di comunicazione tra un componente software, o un modulo, ed un altro. Questo processo è mostrato in Figura 32.

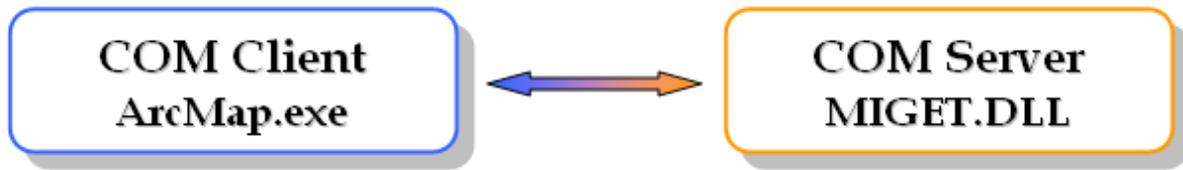


Figura 32: La comunicazione tra due oggetti COM

Per poter creare facilmente questi oggetti COM, viene utilizzata la libreria di modelli attivi, dall'inglese *Active Template Library (ATL)*, sviluppata da Microsoft. Queste librerie ATL sono delle classi modello scritte in linguaggio C++, progettate per essere piccole, veloci ed estensibili, e vengono talvolta considerate come una evoluzione della libreria di modelli standard (*Standard Template Library, STL*).

Tramite queste tecnologie, è dunque possibile implementare un oggetto COM Server, sotto forma di una libreria dinamica (*Dynamic Link Library, DLL*), che conterrà al suo interno il codice per la creazione della barra degli strumenti e dei due comandi relativi all'elaborazione delle previsioni, come mostrato in Figura 33.

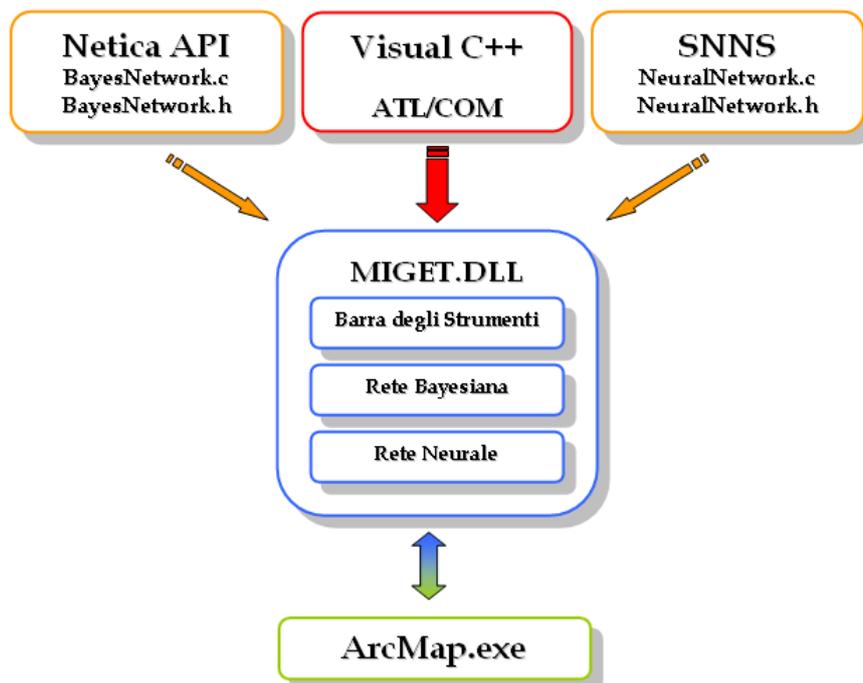


Figura 33: Il componente COM per il progetto MIGET

I comandi possono essere sviluppati integrando una interfaccia utente, aggiungendo al progetto Visual C++ una classe per la creazione di una finestra di dialogo ATL. In seguito sarà possibile aggiungere i vari controlli alla finestra, lavorando direttamente con le API di Windows nella stessa maniera in cui si crea un'applicazione che utilizza le *Microsoft Foundation Class (MFC)*.

Quando la libreria *MIGET.DLL* è pronta, dal software *ArcMap*, è possibile caricare la barra degli strumenti, tramite il comando *Customize...* presente nel menù *Tools* e, nella finestra che compare, premendo il pulsante *Add from file...* verrà data la possibilità di scegliere la cartella contenente la libreria. A caricamento effettuato, verrà mostrata la lista contenente tutte le barre degli strumenti disponibili e, impostando un segno di spunta accanto alla scritta *MIGET* verrà direttamente mostrata la barra degli strumenti nell'interfaccia del software *ArcMap*.

3.4.2 Sviluppo di un'applicazione indipendente

Lo sviluppo di un'applicazione indipendente, è possibile grazie all'utilizzo del prodotto *ESRI ArcGIS Engine*, questo infatti è una libreria di componenti GIS che permette di creare, tramite l'*ArcGIS Engine Developer Kit*, delle soluzioni GIS focalizzate su scopi specifici, come mostrato in Figura 34.

Dato che il sistema *ArcGIS* si basa sugli oggetti *ArcObject*, lo sviluppo di un'applicazione personalizzata si basa anch'essa sull'utilizzo e sull'implementazione di oggetti *ArcObject*. Inoltre, *ArcGIS Engine*, permette lo sviluppo di soluzioni per diversi sistemi operativi, come Windows UNIX e Linux, supportando una vasta gamma di ambienti di sviluppo, quali Visual C++, Java, COM, .Net e Visual Basic.

Per sviluppare l'applicazione utile per il nostro scopo, occorre dunque utilizzare il prodotto *ArcGIS Engine Developer Kit* congiuntamente ad uno degli ambienti di sviluppo sopra citati e, una volta registrati i componenti, utilizzare i vari controlli, strumenti, barre degli strumenti e librerie di oggetti, per includere le funzionalità GIS desiderate.

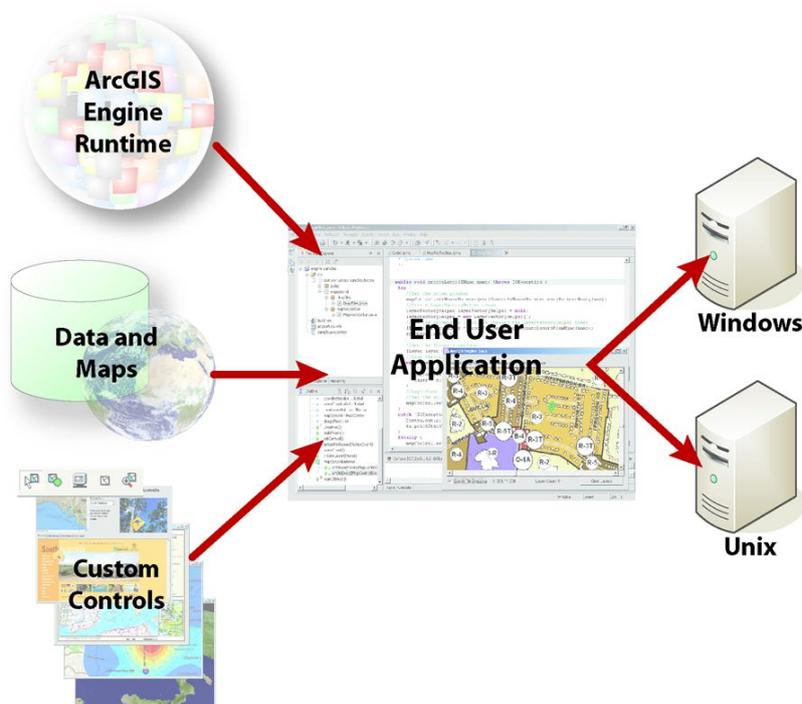


Figura 34: Il sistema ESRI ArcGIS Engine

L'applicazione, per soddisfare le funzionalità richieste dal committente, dovrà almeno prevedere l'utilizzo dei componenti per il caricamento, salvataggio, selezione e visualizzazione dei vari livelli (*Layer*), e implementare le due finestre di dialogo, analogamente al caso dell'estensione, per i comandi relativi al calcolo della previsione degli indici e della temperatura.

La procedura di interfacciamento dei sistemi *Netica*, *SNNS* e *ArcGIS*, mostrata in Figura 35, è la stessa di quella descritta per lo sviluppo dell'estensione, a parte qualche piccolo accorgimento per quanto riguarda il collegamento dei file generati dallo strumento *Snns2c*, che genera solo file in linguaggio C.

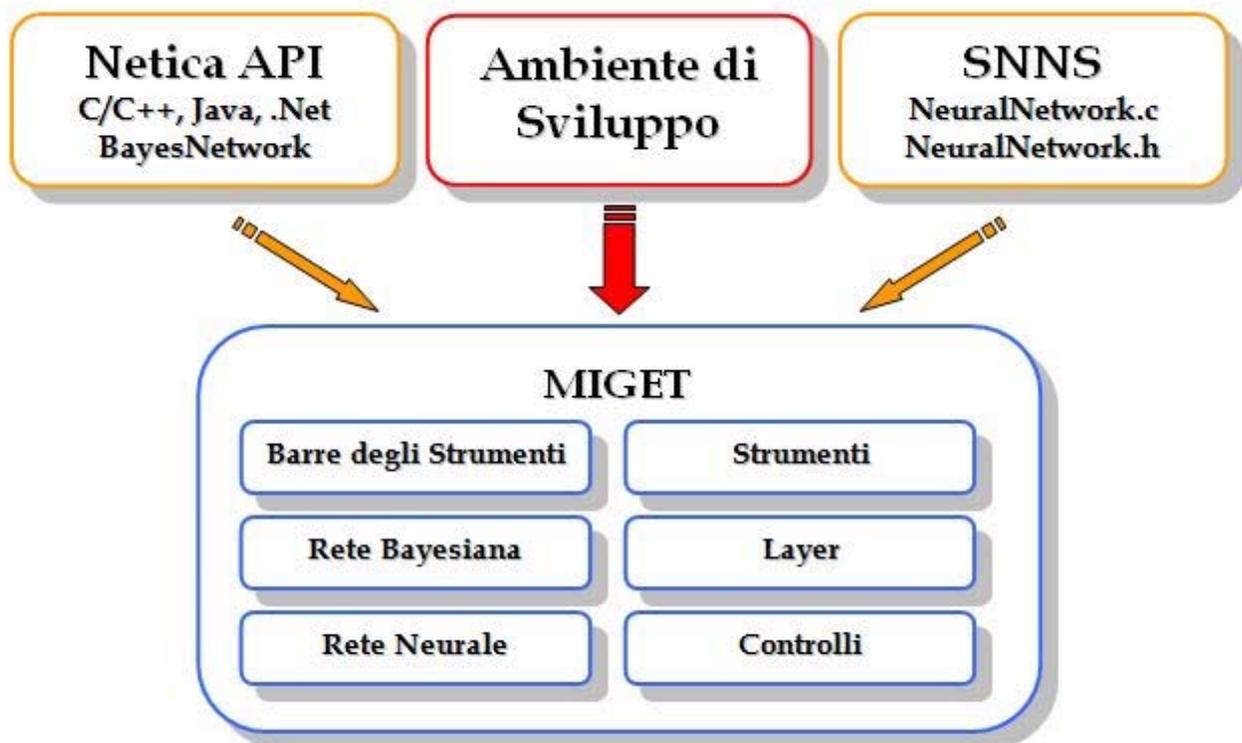


Figura 35: Il software personalizzato

L'applicazione personalizzata risultante, per qualsiasi sistema operativo sia stata realizzata (Windows, UNIX, Linux, Mac OS), necessita del prodotto *ArcGIS Engine Runtime*, che è rivolto all'utente finale, oppure del prodotto *ArcGIS Desktop*, che gli consentono di eseguire sul proprio computer le applicazioni sviluppate tramite *ArcGIS Engine Developer Kit*. Questa interazione è mostrata in Figura 36.

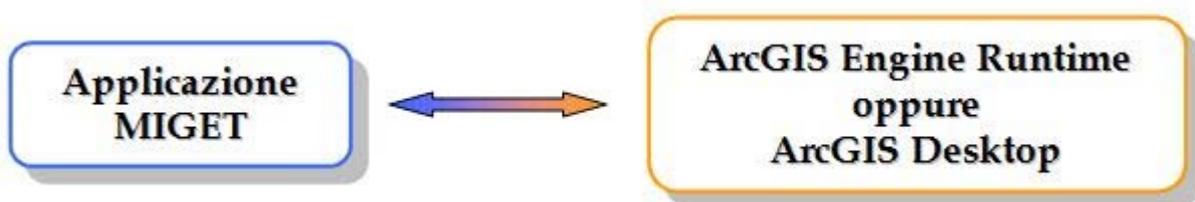


Figura 36: L'ArcGIS Engine Runtime o ArcGIS Desktop richiesti dal software

3.5 Conclusioni sullo studio di fattibilità

Entrambe le soluzioni proposte presentano ovviamente vantaggi e svantaggi, l'una rispetto all'altra. In particolare, l'estensione, presenta lo svantaggio di essere vincolata, dato che si tratta di una libreria nativa (DLL), al sistema operativo Windows, ed all'acquisto del prodotto *ESRI ArcGIS Desktop*, senza del quale non può essere utilizzata.

La seconda soluzione, quella dell'applicazione personalizzata, sotto questi aspetti risulterebbe la migliore scelta, in quanto, se venisse sviluppata con un linguaggio ad alta portabilità tipo JAVA, potrebbe essere eseguita su vari sistemi operativi, e l'utente finale avrebbe solo l'onere dell'acquisto della licenza del prodotto *ESRI ArcGIS Engine Runtime*, che gli permetterebbe l'esecuzione sul computer.

Di contro, il vantaggio che presenta la seconda soluzione, diventa un grosso svantaggio se consideriamo i tempi di progettazione, implementazione e testing dell'applicazione, che sono invece notevolmente ridotti nel caso dell'estensione.

4 L'integrazione di un modello neurale e di un modello bayesiano in ArcGIS

I modelli intelligenti sviluppati nell'ambito del progetto *MIGET*, sono di grande utilità pratica, ma come tutti gli strumenti software e gli algoritmi di elaborazione, per quanto semplici o complessi che siano, se non vi è il modo di farli utilizzare anche ad un utente meno esperto, servono a ben poco.

L'integrazione dei modelli discussi in precedenza, all'interno di un sistema informativo geografico (GIS, dall'inglese *Geographic Information System*), è stata pensata appositamente per permettere agli utenti finali utilizzatori di un GIS, di poter utilizzare con semplicità questi strumenti di previsione. Il sistema GIS in cui è stata eseguita l'integrazione, è il software *ESRI ArcGIS Desktop 9.2*[16], ed in particolare in *ArcMap*.

ESRI ArcGIS 9.2, la cui architettura è mostrata in Figura 37, è una famiglia di software integrati per la gestione del sistema informativo geografico. Esso si basa su una libreria di componenti software GIS condivisi, chiamata *ArcObject*. La *ESRI* fornisce un insieme di strumenti di sviluppo, denominato *ArcGIS Developer Kit*, che consente di creare applicazioni indipendenti, oppure di estendere le funzionalità dei software già esistenti come *ArcMap*.

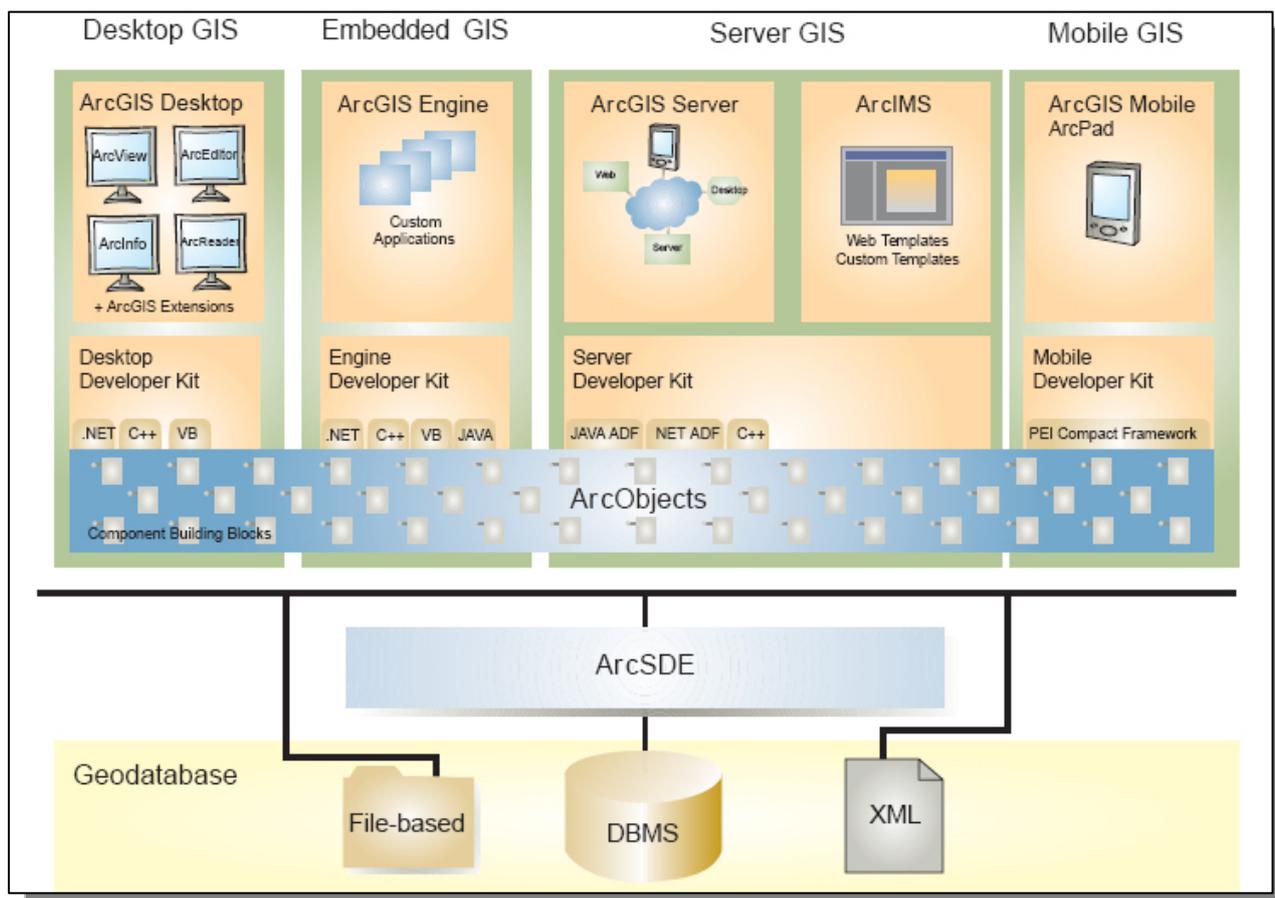


Figura 37: L'architettura del prodotto ESRI ArcGIS 9.2

L'integrazione è stata eseguita sviluppando un'estensione per il software *ArcMap*. L'estensione è uno strumento personalizzato o un comando, che è possibile caricare in *ArcMap*, per poter eseguire rispettivamente un insieme di operazioni che diano maggiore interattività all'utente, o un'operazione eseguita in seguito alla pressione su un bottone.

Questa estensione è stata sviluppata utilizzando la tecnologia *Microsoft Component Object Model (COM)*, utilizzando l'ambiente di sviluppo *Microsoft Visual C++*, e distribuita sotto forma di *Dynamic Link Library (DLL)*.

L'integrazione dei tre sistemi (*SNNS*, *Netica* e *ArcGIS*), è stata possibile grazie all'esistenza degli strumenti di sviluppo adeguati, che forniscono la possibilità di implementare delle procedure per accedere alle reti sviluppate e poterle utilizzare nella stessa maniera in cui si utilizzano con il software di base. In particolare il software *SNNS v4.2*, mette a disposizione uno strumento a riga di comando (*snns2c*), che offre la possibilità di esportare una rete neurale, creata con *SNNS*, in codice sorgente per il linguaggio C, come è mostrato in Figura 38.

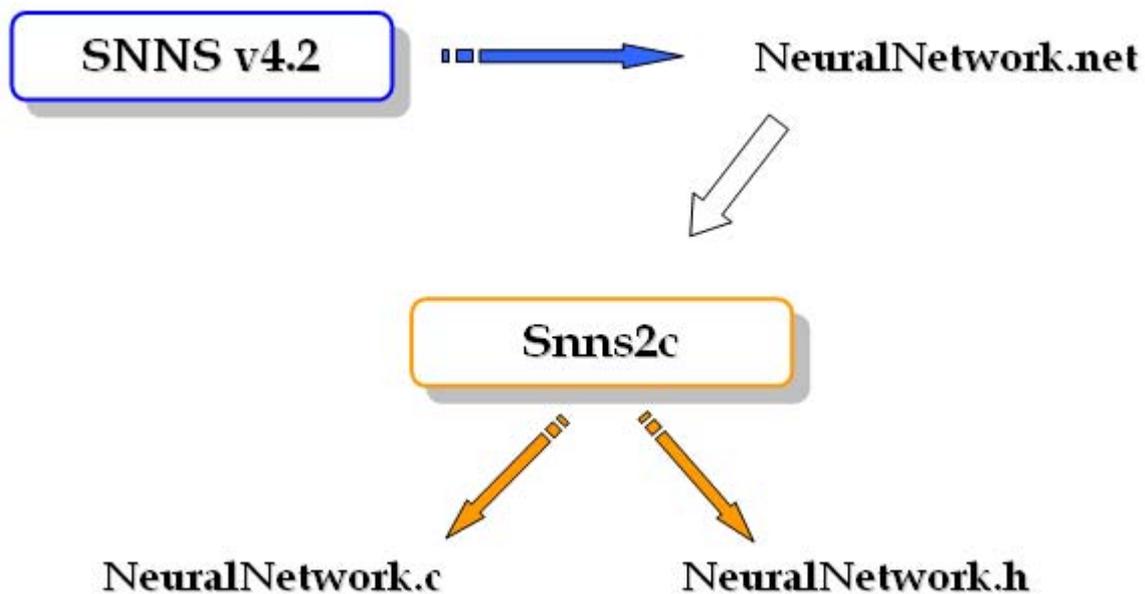


Figura 38: L'esportazione con lo strumento *snns2c*

Questi sorgenti, sono stati utilizzati per integrare le reti neurali delle sei stazioni meteorologiche, nella libreria *MIGET*.

In maniera simile, la rete bayesiana creata con *Netica v3.19*, è integrata nella libreria *MIGET*, ma questa volta non è utilizzato uno strumento esterno, ma le librerie per lo sviluppo di *Netica*.

In Figura 39, è mostrato uno schema in cui una rete bayesiana è creata con *Netica v3.19*, ed in seguito, tramite l'interazione tra la libreria *Netica API* e l'ambiente di sviluppo *Microsoft Visual C++*, è stato creato il codice sorgente che permetta di effettuare un'inferenza probabilistica sulla rete creata precedentemente.

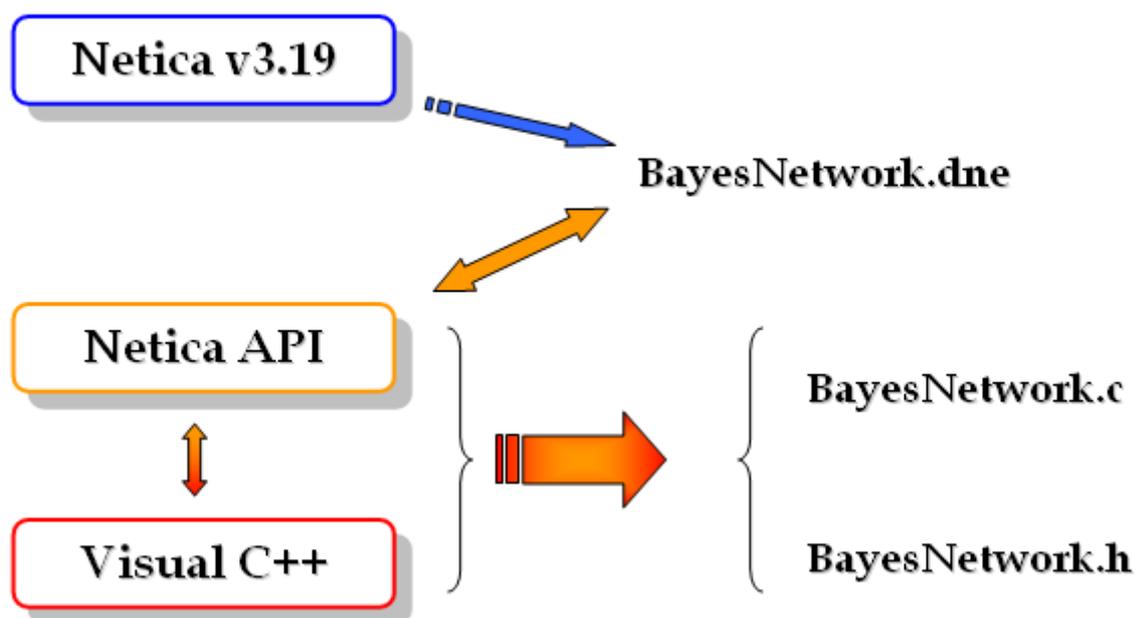


Figura 39: L'interazione delle librerie con l'ambiente Visual C++

I componenti *ArcObject*, sono basati sulla tecnologia *Microsoft Component Object Model (COM)*, che è una piattaforma per componenti software che permette la comunicazione tra processi e la creazione dinamica di oggetti con qualsiasi linguaggio di programmazione che supporta questa tecnologia. Questa infatti definisce un protocollo di comunicazione tra un componente software, o un modulo, ed un altro. Questo processo è mostrato in Figura 40.

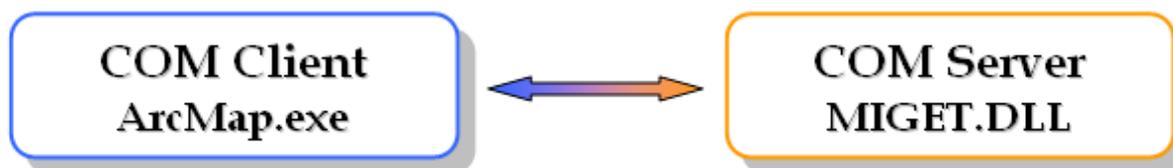


Figura 40: La comunicazione tra due oggetti COM

Per poter creare facilmente questi oggetti COM, viene utilizzata la libreria di modelli attivi, dall'inglese *Active Template Library (ATL)*, sviluppata da Microsoft. Queste librerie ATL sono delle classi modello scritte in linguaggio C++, progettate per essere piccole, veloci ed estensibili, e vengono talvolta considerate come una evoluzione della libreria di modelli standard (*Standard Template Library, STL*).

Tramite queste tecnologie, è stato possibile implementare un oggetto COM Server, sotto forma di una libreria dinamica (*Dynamic Link Library, DLL*), che contiene al suo interno il codice per la creazione della barra degli strumenti e dei due comandi relativi all'elaborazione delle previsioni, come mostrato in Figura 41.

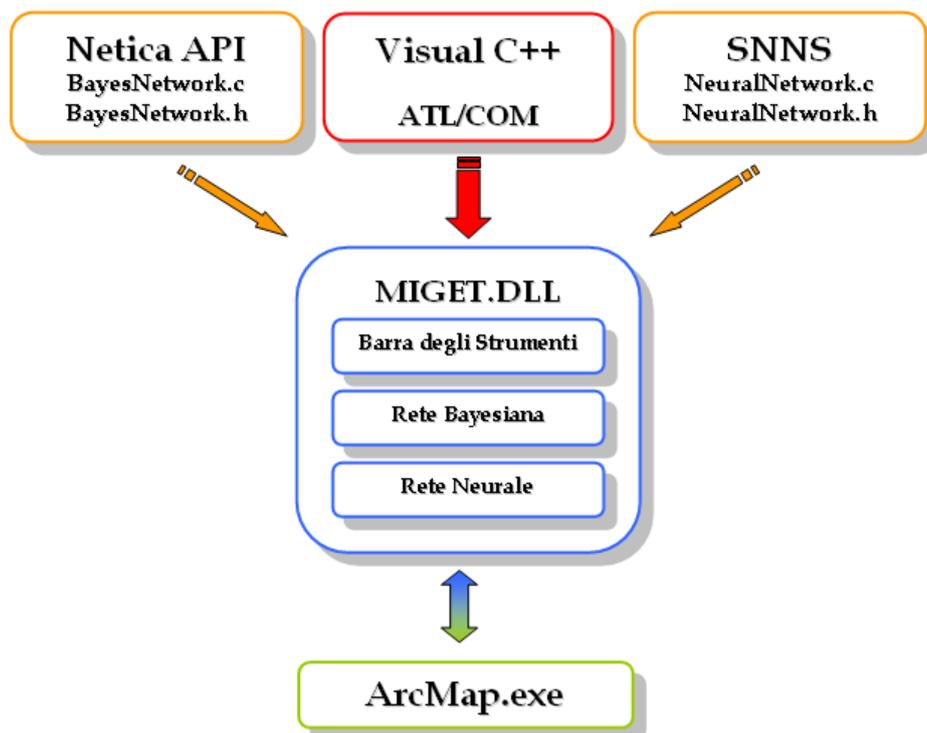


Figura 41: Il componente COM per il progetto MIGET

In Figura 42, è mostrata la barra degli strumenti, che contiene i due pulsanti, uno per accedere allo strumento di previsione per le temperature, e l'altro per accedere allo strumento di previsione per gli indicatori.



Figura 42: La barra degli strumenti

Il riscontro visivo è un fattore molto importante nell'analisi, in quanto fornisce un'idea sull'andamento di una grandezza, senza preoccuparsi del valore che essa assume. Quest'ultimo può comunque essere esaminato, per un'analisi più dettagliata. Visto che gli indicatori assumono valori compresi tra 0 e 10, è stato scelto di attribuire una scala di colori per indicare il valore assunto. Questa scala di valori è mostrata in Figura 43.



Figura 43: La scala dei colori per rappresentare i valori degli indicatori

Per le temperature, invece, è stato scelto di mostrare il valore elaborato, direttamente sulla mappa, tramite l'applicazione di etichette associate alla stazione meteorologica.

In Figura 44, è mostrata una situazione in cui nella mappa sono visibili i risultati dell'elaborazione di entrambi gli strumenti di previsione, mostrando le unità fisiografiche paesaggistiche (UFP) colorate in base al colore corrispondente al valore previsto, e per le temperature è mostrata l'etichetta con il valore elaborato dallo strumento che fa uso delle reti neurali artificiali. Inoltre è possibile notare che nella tabella dei contenuti, mostrata nella parte sinistra dell'immagine, sono visibili i valori calcolati per l'indicatore selezionato.

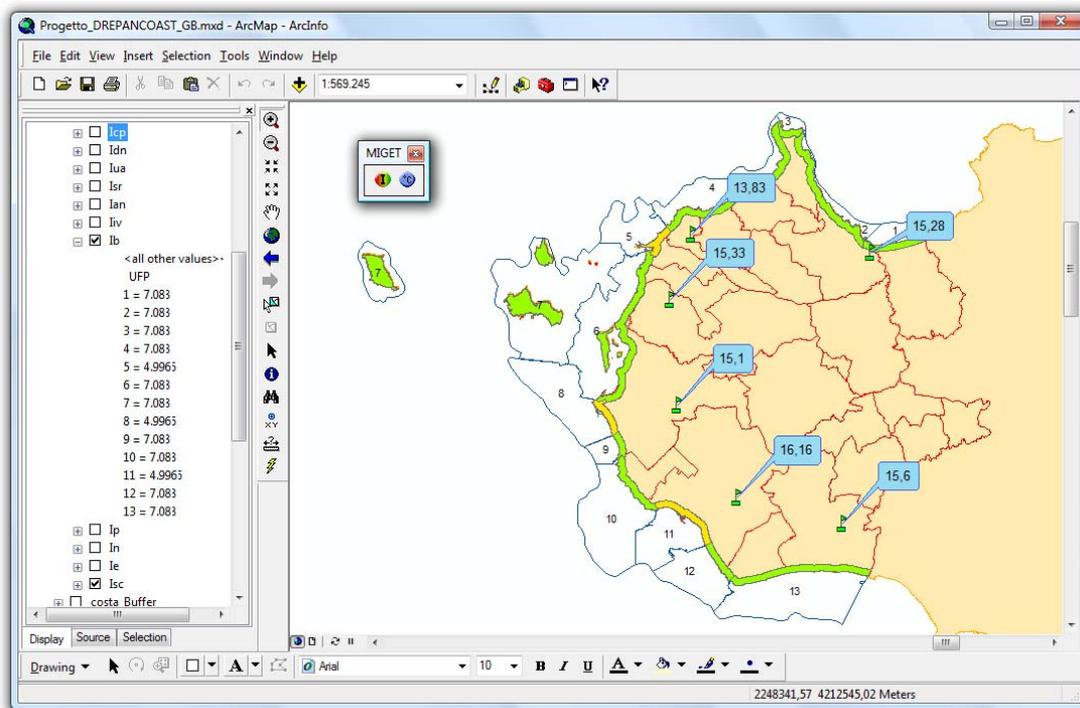


Figura 44: La barra degli strumenti

Le interfacce degli strumenti sono state progettate per essere semplici ed intuitive, e corredate anche di una semplice guida all'uso che spiega come utilizzare lo strumento. In Figura 45, è mostrata l'interfaccia dello strumento per le previsioni della temperatura.



Figura 45: L'interfaccia dello strumento per le previsioni della temperatura

In Figura 46, è mostrata l'interfaccia dello strumento per l'analisi degli indicatori.

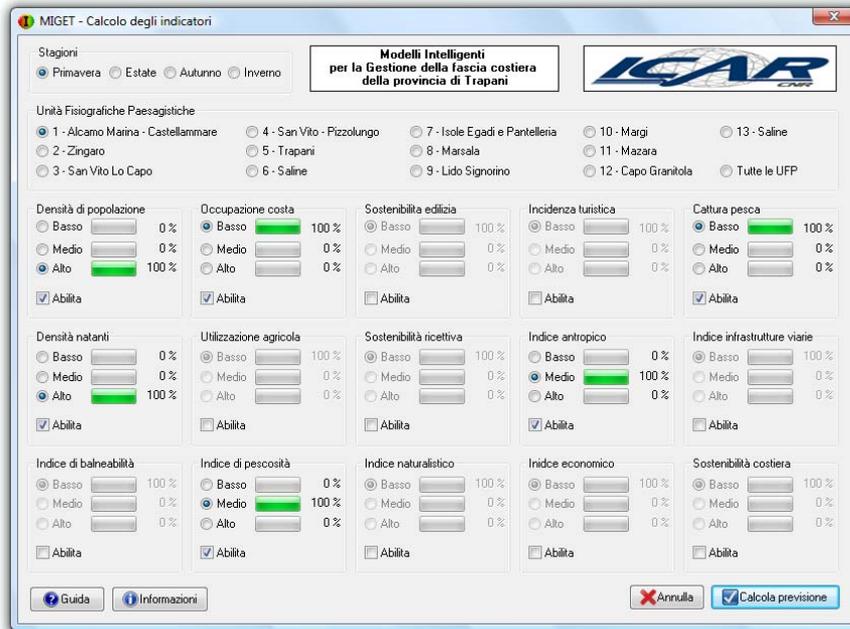


Figura 46: L'interfaccia dello strumento per l'analisi degli indicatori

Di seguito, sono mostrati gli esempi di funzionamento degli strumenti, in particolare nelle Figura 47 e Figura 48, lo strumento per le previsioni della temperatura è utilizzato per elaborare la temperatura massima al tempo t , per tutte le sei stazioni meteorologiche, dopo aver impostato i parametri meteorologici al tempo $t-1$.

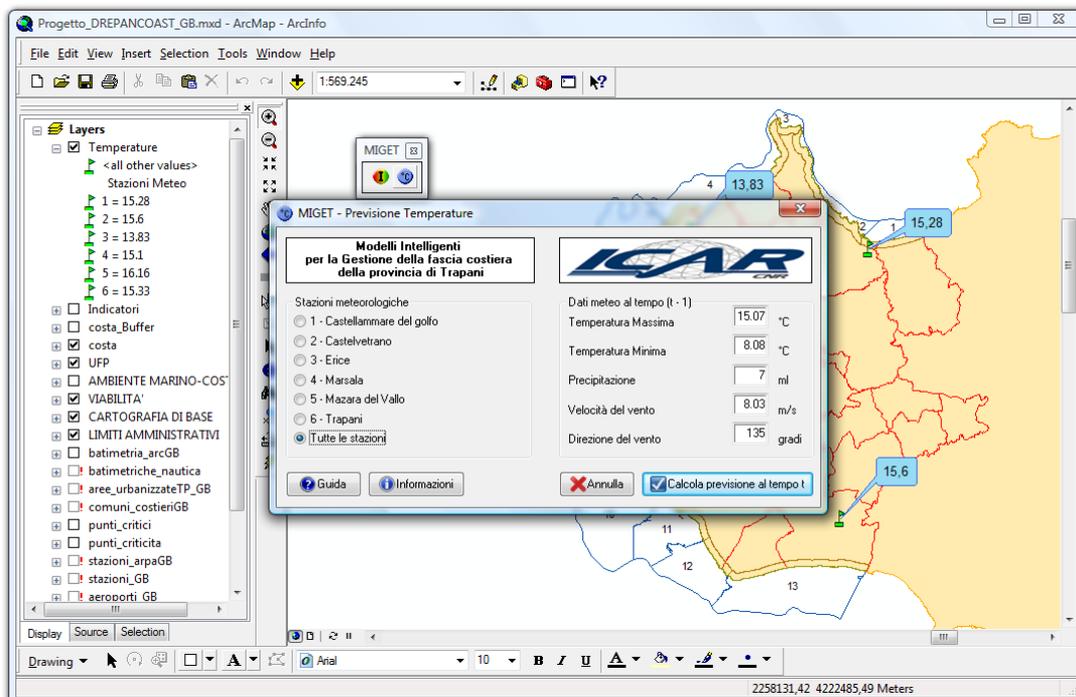


Figura 47: Un esempio di funzionamento dello strumento per le previsioni della temperatura

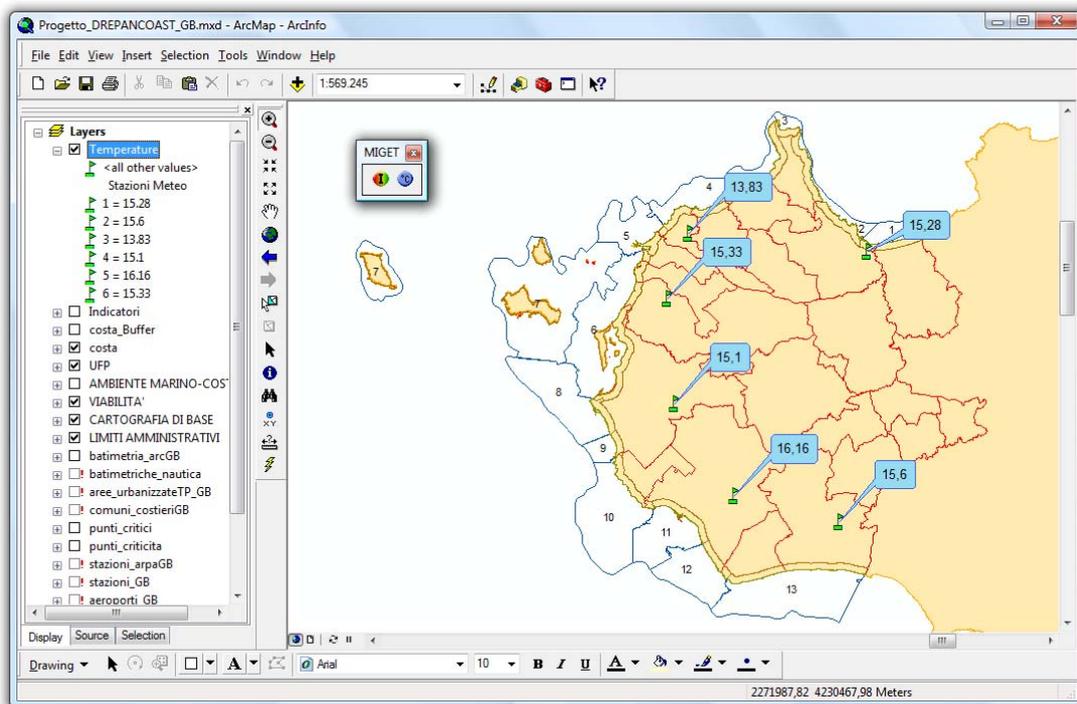


Figura 48: Il risultato dell'esempio visto in precedenza

Mentre nelle Figura 49 e Figura 50, è mostrata l'applicazione pratica dell'esempio descritto a pagina 26, riguardante l'analisi degli indicatori.

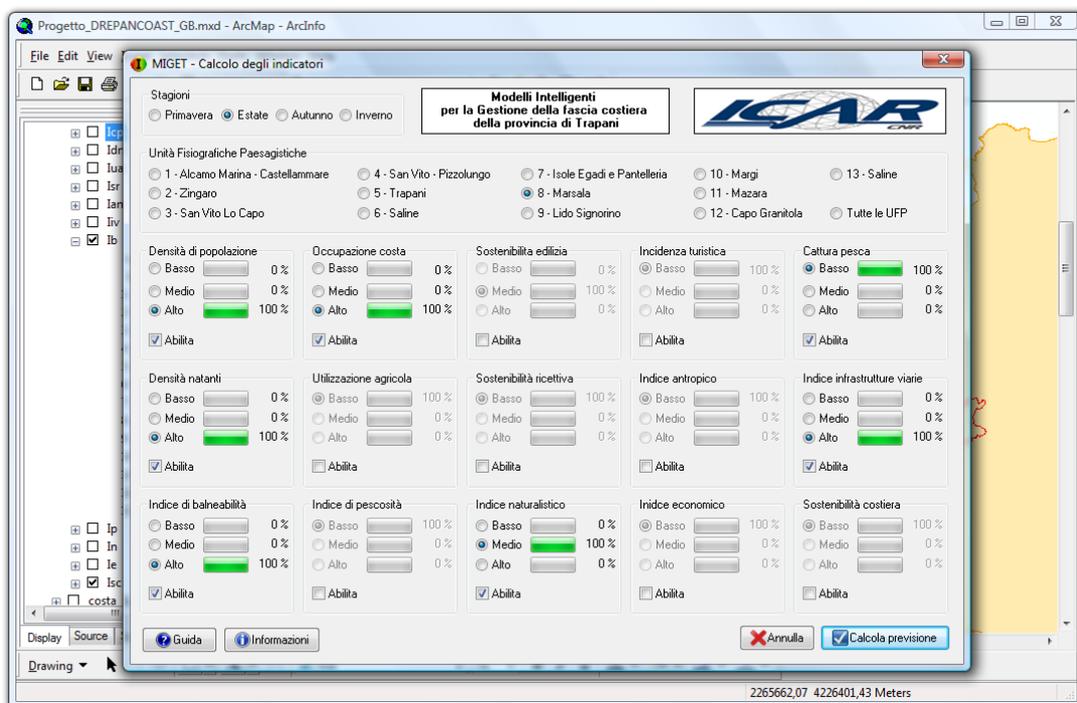


Figura 49: Un esempio di funzionamento dello strumento per l'analisi degli indicatori

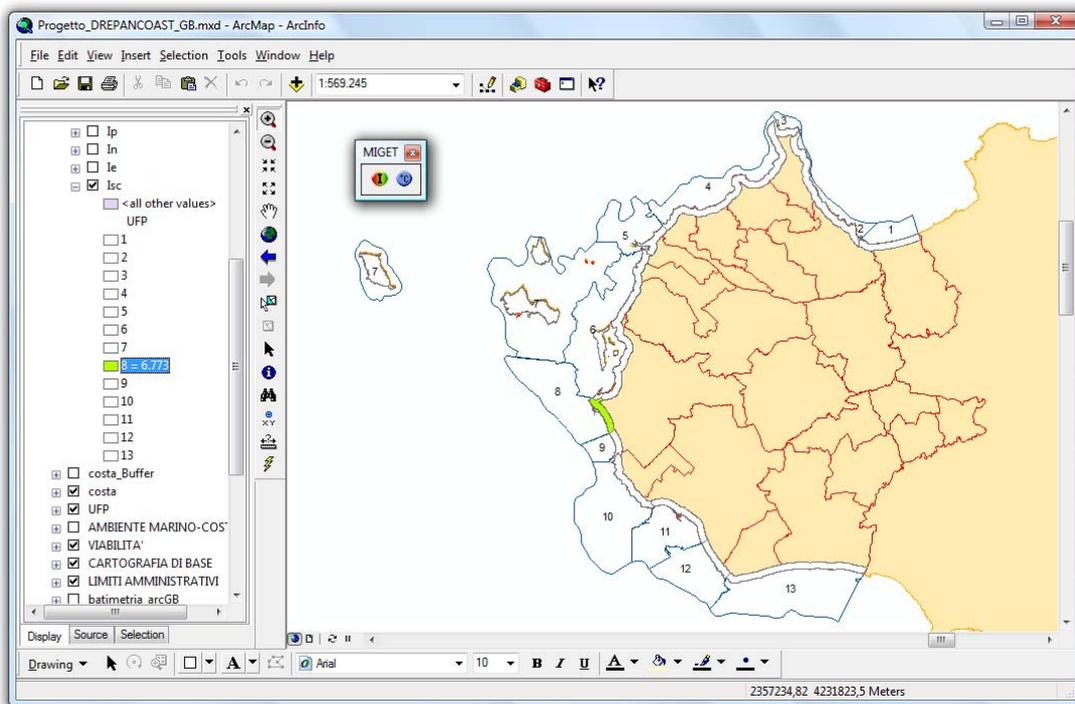


Figura 50: Il risultato dell'esempio visto in precedenza

Infine nelle Figura 51 e Figura 52, è mostrato l'altro esempio di applicazione pratica descritta a pagina 27, riguardante sempre l'analisi degli indicatori.

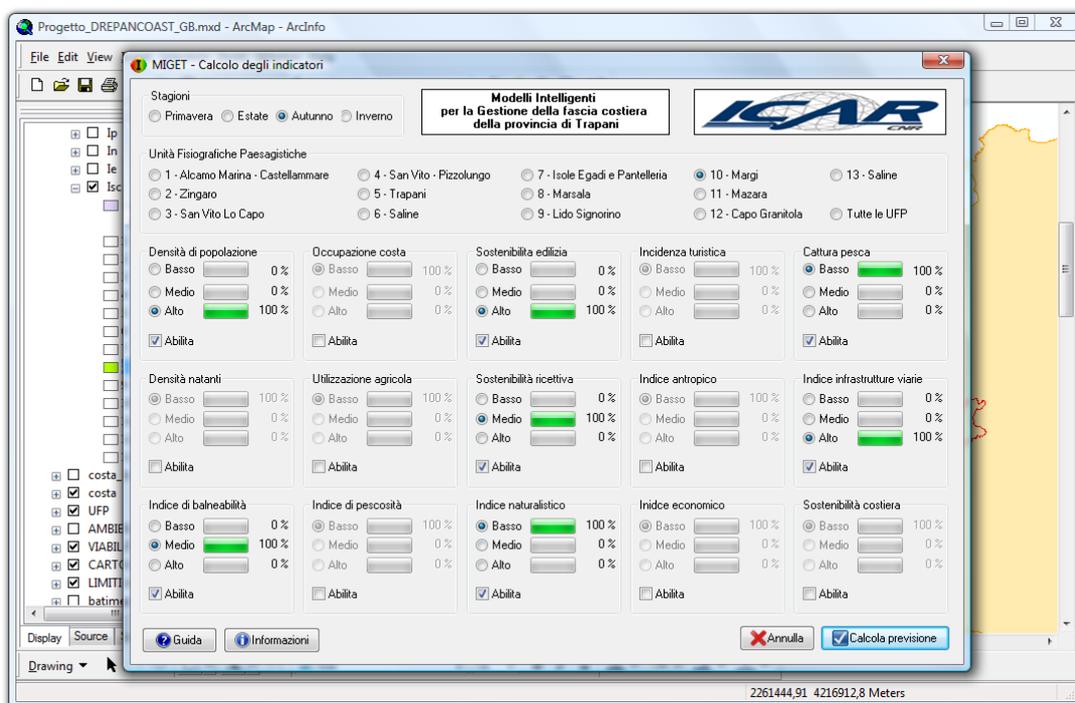


Figura 51: Un altro esempio di funzionamento dello strumento per l'analisi degli indicatori

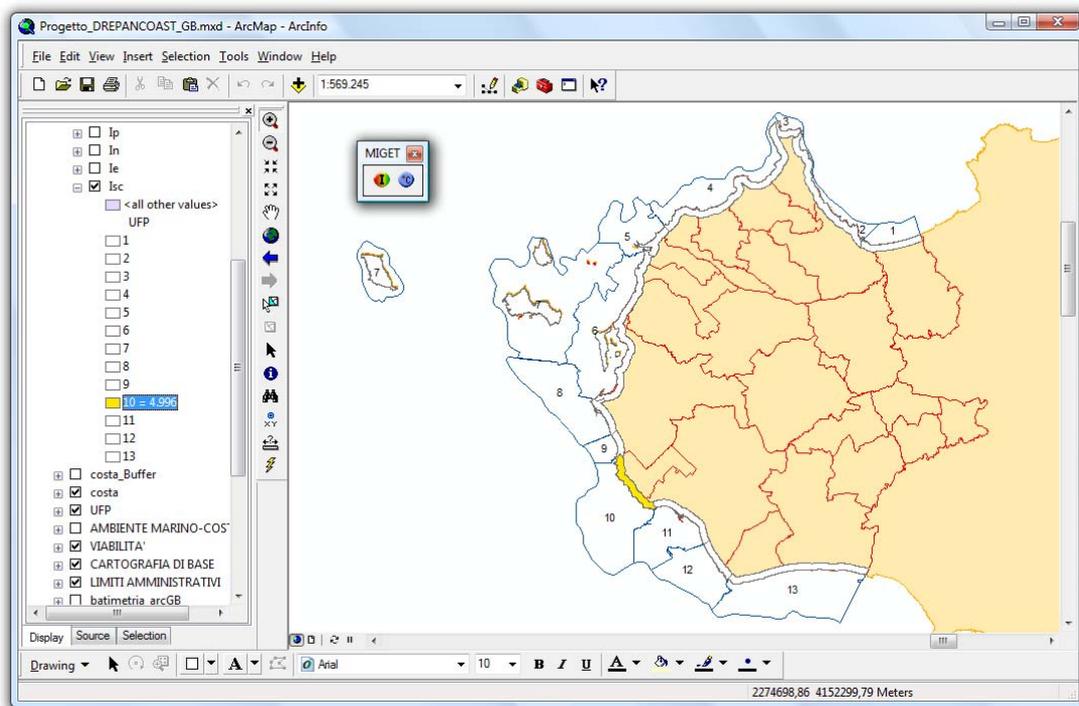


Figura 52: Il risultato dell'esempio visto in precedenza

Conclusioni

I modelli intelligenti sviluppati in ambito del progetto *MIGET*, permettono, tramite l'integrazione nel framework ArcGIS 9.2, di poter procedere ad una valutazione visiva sull'andamento di determinate grandezze oggetto di analisi, quali le varie tipologie di indicatori e le temperature.

Per quanto riguarda il modello per la rete neurale ricorrente, si è dimostrato adatto allo scopo di previsione per cui era stato progettato, mostrando delle difficoltà nella previsione solo nei casi in cui vi sia una brusca variazione di temperatura in due giorni consecutivi. Questo problema potrebbe essere risolto estendendo i dati di addestramento, fornendo alla rete un maggior numero di casi su cui basarsi per poter sopperire alle lacune mostrate. Per quanto riguarda invece il modello probabilistico per la rete bayesiana, per poter ottenere delle tabelle di probabilità condizionata più complete, occorrerebbe sia una maggiore informazione riguardante l'insieme di dati di addestramento, sia un maggiore raffinamento della struttura della rete stessa, modificando le connessioni tra i nodi ed eseguendo un'accurata analisi di sensibilità degli indicatori, analizzando l'influenza che ognuno di essi ha sull'indicatore di sostenibilità costiera.

Infine, per quanto riguarda l'integrazione dei modelli intelligenti in ArcGIS, si potrebbe migliorare l'interfaccia dello strumento per l'analisi degli indicatori, permettendo di inserire anche il valore numerico di cui se ne ha conoscenza, e non solo il livello di appartenenza (basso, medio, alto).

Un'altra miglioria che potrebbe essere aggiunta a questo strumento, è la possibilità di mostrare il grado di certezza/incertezza del valore previsto, ossia la probabilità con cui il valore previsto, appartiene ad un determinato livello (es. 67% alto).

In definitiva, l'utilizzo combinato di ArcGIS e l'applicazione dei modelli intelligenti delle reti neurali ricorrenti e delle reti probabilistiche bayesiane, permette di poter valutare l'indicatore di sostenibilità costiera e degli altri indicatori delle unità fisiografiche paesaggistiche, e anche l'andamento delle temperature massime giornaliere delle stazioni meteorologiche, fornendo un valido supporto per l'analisi e la gestione della fascia costiera della provincia di Trapani.

Indice delle figure

Figura 1: Il processo di analisi.....	2
Figura 2: La banca dati del progetto DrepanCoast	3
Figura 3: Un'unità logica a soglia	4
Figura 4: Andamento normalizzato delle grandezze analizzate.....	5
Figura 5: Lo schema dell'architettura neurale di Elman	7
Figura 6: La rete neurale di Elman sviluppata.....	8
Figura 7: La rete neurale di Elman sviluppata.....	10
Figura 8: Il confronto tra il valore previsto e quello reale per la stazione di Castellammare.....	11
Figura 9: La frequenza dell'errore assoluto per la stazione di Castellammare	11
Figura 10: Il confronto tra il valore previsto e quello reale per la stazione di Castelvetroano.....	12
Figura 11: La frequenza dell'errore assoluto per la stazione di Castelvetroano	12
Figura 12: Il confronto tra il valore previsto e quello reale per la stazione di Erice	13
Figura 13: La frequenza dell'errore assoluto per la stazione di Erice	13
Figura 14: Il confronto tra il valore previsto e quello reale per la stazione di Marsala	14
Figura 15: La frequenza dell'errore assoluto per la stazione di Marsala.....	14
Figura 16: Il confronto tra il valore previsto e quello reale per la stazione di Mazara	15
Figura 17: La frequenza dell'errore assoluto per la stazione di Mazara	15
Figura 18: Il confronto tra il valore previsto e quello reale per la stazione di Trapani	16
Figura 19: La frequenza dell'errore assoluto per la stazione di Trapani.....	16
Figura 20: I grafici della sensibilità per le stazioni di Castellammare e Castelvetroano	17
Figura 21: I grafici della sensibilità per le stazioni di Erice e Marsala	17
Figura 22: I grafici della sensibilità per le stazioni di Mazara del Vallo e Trapani.....	18
Figura 23: L'andamento dell'uscita al variare della temperatura massima e minima	18
Figura 24: L'andamento dell'uscita al variare della precipitazione e della velocità e direzione del vento	19
Figura 25: Il modello della rete bayesiana sviluppato	24
Figura 26: Un esempio di inferenza probabilistica.....	26
Figura 27: Un altro esempio di inferenza probabilistica.....	27
Figura 28: Il sistema ArcGIS 9	29
Figura 29: L'interazione tra sistemi ArcGIS e ArcObject	30
Figura 30: L'esportazione con lo strumento snns2c	31
Figura 31: L'interazione delle librerie con l'ambiente Visual C++	32
Figura 32: La comunicazione tra due oggetti COM	33
Figura 33: Il componente COM per il progetto MIGET.....	33
Figura 34: Il sistema ESRI ArcGIS Engine	34
Figura 35: Il software personalizzato	35
Figura 36: L'ArcGIS Engine Runtime o ArcGis Desktop richiesti dal software.....	35
Figura 37: L'architettura del prodotto ESRI ArcGIS 9.2	37
Figura 38: L'esportazione con lo strumento snns2c	38
Figura 39: L'interazione delle librerie con l'ambiente Visual C++.....	39
Figura 40: La comunicazione tra due oggetti COM	39
Figura 41: Il componente COM per il progetto MIGET.....	40
Figura 42: La barra degli strumenti.....	40
Figura 43: La scala dei colori per rappresentare i valori degli indicatori.....	40
Figura 44: La barra degli strumenti.....	41
Figura 45: L'interfaccia dello strumento per le previsioni della temperatura	41
Figura 46: L'interfaccia dello strumento per l'analisi degli indicatori.....	42
Figura 47: Un esempio di funzionamento dello strumento per le previsioni della temperatura	42
Figura 48: Il risultato dell'esempio visto in precedenza.....	43
Figura 49: Un esempio di funzionamento dello strumento per l'analisi degli indicatori.....	43
Figura 50: Il risultato dell'esempio visto in precedenza.....	44
Figura 51: Un altro esempio di funzionamento dello strumento per l'analisi degli indicatori.....	44
Figura 52: Il risultato dell'esempio visto in precedenza.....	45

Bibliografia

- [1] Nils J. Nilson, (1998), "Intelligenza Artificiale", titolo originale "Artificial Intelligence: A New Synthesis", APOGEO, edizione italiana a cura di S. Gaglio
- [2] 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.
- [3] 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.
- [4] Lapedes, R., Farber, R. (1987) "Non linear signal processing using neural networks: prediction and system modelling" Los Alamos National Laboratory, Los Alamos, New Mexico.
- [5] 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>
- [6] Elman, J. L. (1990), "Finding structure in time". In Cognitive Science, n.14 pp. 179-211.
- [7] 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.
- [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] 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>
- [10] Muriel G., Ioannis D., Sovan L. (2003) " Review and comparison of methods to study the contribution of variables in artificial neural network models"
- [11] Berthold M., Hand D. J. (1998). Intelligent data analysis. Springer Berlin 1998
- [12] 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
- [13] D.E. Rumelhart and J.L. McClelland. Parallel Distributed Processing, volume 1. MIT Press, 1986.
- [14] Stuttgart Neural Network Simulator, url: <http://www-ra.informatik.uni-tuebingen.de/SNNS/>.
- [15] Netica, Norsys Software Incor. url: <http://www.norsys.com/netica.html>
- [16] ArcGIS, ESRI, url: <http://www.esri.com/>
- [17] 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

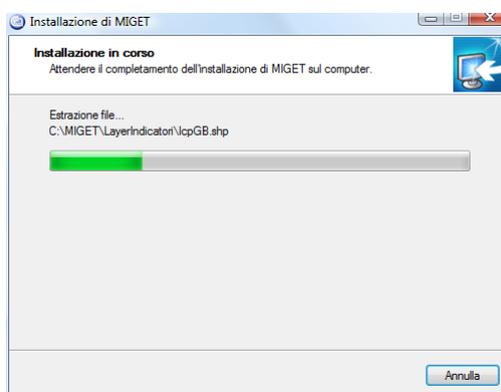
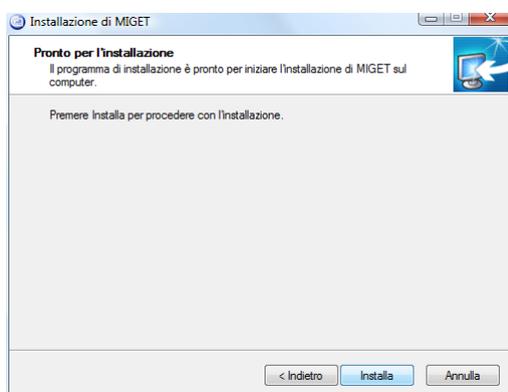
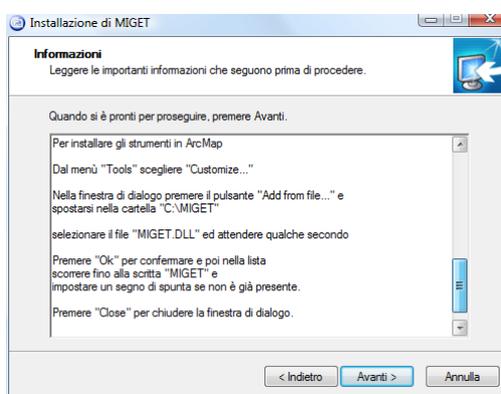
Appendice A: Procedura di installazione

L'installazione della libreria avviene in due fasi:

- Installazione sul disco fisso
- Installazione in ArcGIS

Installazione su disco fisso

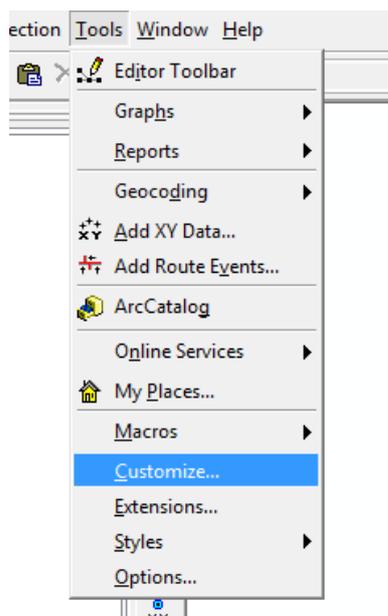
Per eseguire l'installazione della libreria sul disco fisso, utilizzare il pacchetto di installazione *InstallMIGET.exe*, e seguire i passi dell'installazione.



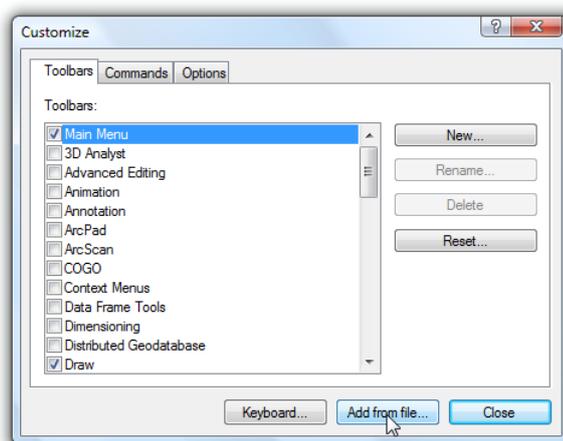
Installazione in ArcGIS

Per installare la barra degli strumenti *MIGET*, e gli strumenti per le previsioni, nel software *ArcMap*, seguire i passi descritti qui di seguito.

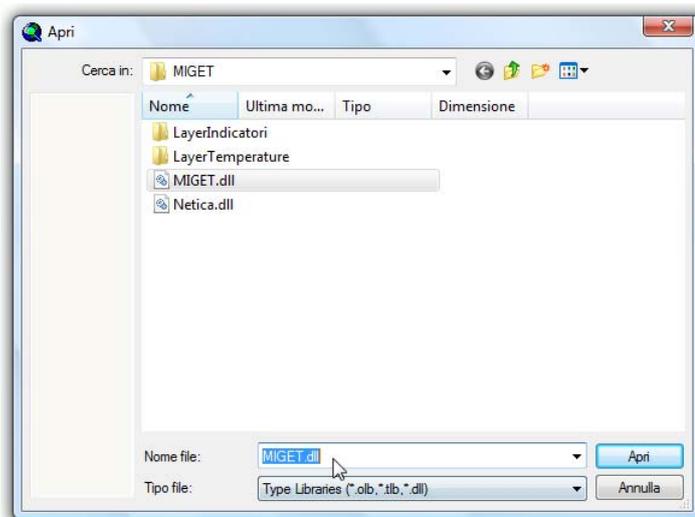
1. Avviare il software *ArcMap*
2. Dal menù *Tools* scegliere il comando *Customize ...*



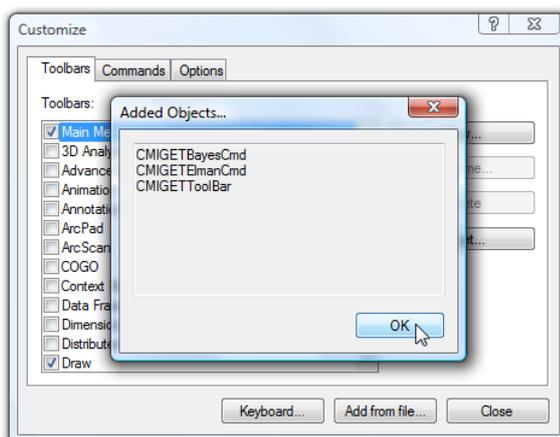
3. Nella finestra di dialogo che compare premere il pulsante *Add from file ...*



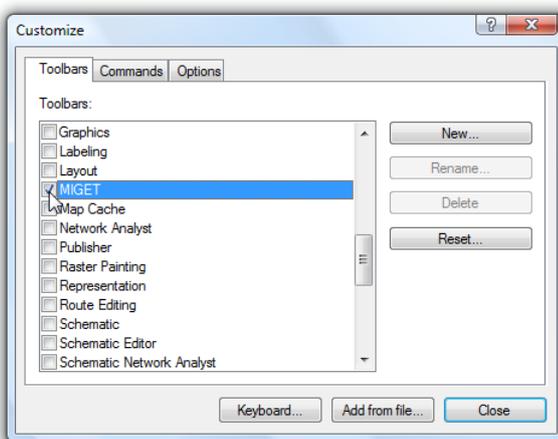
4. Spostarsi nella cartella *C:\MIGET* e selezionare il file *MIGET.dll*



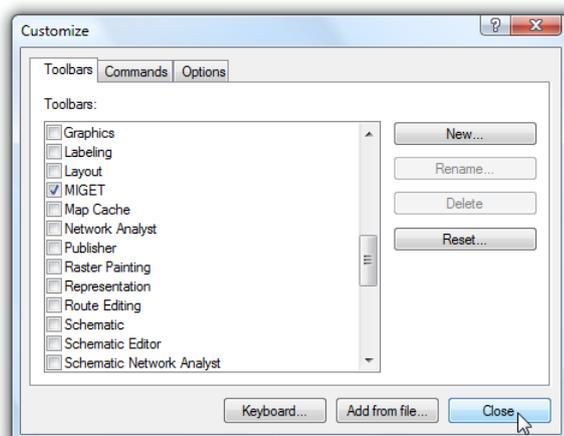
5. Attendere il caricamento dei moduli e confermare premendo il pulsante *Ok*



6. Scorrere nella lista fino a trovare la barra degli strumenti *MIGET* e impostare un segno di spunta accanto al nome



7. Chiudere la finestra di dialogo premendo il pulsante *Close*



8. La barra degli strumenti *MIGET* sarà ora visibile nel software *ArcMap*



La procedura di installazione è ora completa.