

## Reti Neurali Artificiali per lo studio del mercato

Università degli studi di Brescia - Dipartimento di metodi quantitativi

Marco Sandri (sandri.marco@gmail.com)

### Funzione di risposta di una RNA - Parte 1

#### Obiettivi:

- (1) realizzare in modo molto semplice una rete neurale artificiale (RNA) **senza fare uso del toolbox Neural Networks** ed impostando direttamente la formula della funzione di risposta della RNA in forma matriciale:

$$y = \text{logsig}(X * \Omega_1) * \Omega_2$$

dove **logsig** è la funzione sigmoide  $1/(1+e^{-x})$ ,  $\Omega_1$  è la matrice che raccoglie i pesi delle connessioni fra strato di input e strato nascosto ed  $\Omega_2$  è la matrice che raccoglie i pesi delle connessioni fra strato nascosto e strato di output;

- (2) mostrare come cambia la funzione di risposta della RNA al variare di:
- numero di neuroni nello strato nascosto; questo parametro (che determina la complessità delle rete) nello script viene chiamato **Nhid** (riga 5);
  - intensità dei pesi delle connessioni; questo parametro nello script viene chiamato **amp\_pesi** (riga 17);

Variando i parametri **Nhid** e **amp\_pesi** si osservano alcuni fatti molto importanti:

- all'aumentare del numero di neuroni nello strato nascosto la risposta della rete si fa sempre più irregolare, frastagliata, complessa, in grado quindi di rappresentare comportamenti sempre meno "lisci" (smooth);
- indipendentemente dalla complessità della rete, l'intensità delle connessioni può determinare comportamenti più o meno lisci della RNA: intensità dei pesi molto basse (e quindi connessioni molto deboli o addirittura disattivate) portano ad avere reti con funzioni di risposta molto lisce, molto regolari. Man mano che i pesi aumentano e quindi man mano che le connessioni si fanno più intense, la risposta della rete diventa sempre più irregolare, sempre liscia.

- (3) Le considerazioni del punto (2) fanno luce su una importante idea volta a controllare il fenomeno dell'overfitting, idea che tratteremo più avanti parlando della cosiddetta **regularization**: premesso che una RNA con una funzione di risposta molto liscia rischia di entrare in overfitting molto meno di una RNA con una funzione di risposta molto irregolare, un metodo per controllare l'overfitting sarà quello di **controllare non solo il numero di neuroni nello strato nascosto ma anche l'intensità dei pesi delle connessioni**.

In fase di stima di una RNA vedremo che per evitare l'overfitting, si andrà ad introdurre una penalità per le RNA con pesi di elevata intensità, andando così a preferire quelle con connessioni debolmente attivate e con una più bassa attitudine a sovra-adattarsi ai dati.

**Attenzione!** In questo esempio **i pesi della rete non vengono ottimizzati** ma impostati in modo del tutto casuale. Non stiamo cioè assolutamente stimando i pesi della rete sulla base di un criterio di ottimo (minimi quadrati). Lo scopo è qui soltanto di vedere come cambia la funzione di risposta della rete al variare del numero di neuroni nello strato nascosto e dell'intensità delle connessioni. La stima dei parametri della RNA feedforward (cioè l'apprendimento della rete) verrà esaminato più avanti.

# Calcolo della funzione di risposta della rete

```
close all; clear all;  
Ninp=2; Nhid=10; Nout=1; N=50;
```

Imposta i parametri della rete: no. input, no. neuroni strato nascosto, no. output e no. osservazioni per ciascun input

```
x1 = linspace(-2,2,N);  
x2 = x1;  
[X1,X2] = meshgrid(x1,x2);  
X = [ones(N^2,1) X1(:) X2(:)];
```

Genera dati campionari

```
amp_pesi = 3;  
Omega1=amp_pesi*randn(Ninp+1,Nhid);  
Omega2=amp_pesi*randn(Nhid,Nout);
```

Imposta i pesi della rete in modo casuale

```
y = logsig(X*Omega1)*Omega2;  
Y = reshape(y,N,N);  
surf(X1,X2,Y);
```

Calcola l'output y della rete e traccia il grafico di  $y = f(x_1, x_2)$  cioè della funzione di risposta  $\text{output} = f(\text{input})$

# Calcolo della funzione di risposta della rete

