

# Le reti neurali

Alberto Borghese

Università degli Studi di Milano  
Laboratory of Applied Intelligent Systems (AIS-Lab)  
Dipartimento di Scienze dell'Informazione  
[borghese@dsi.unimi.it](mailto:borghese@dsi.unimi.it)



A.A. 2005-2006

1/43

<http://homes.dsi.unimi.it/~borghese>



## Sommario



**Dal neurone artificiale alle reti neurali**

L'apprendimento in reti di perceptroni

Esempio con unità lineari

Modelli neurali ed agenti

A.A. 2005-2006

2/43

<http://homes.dsi.unimi.it/~borghese>



## Brains cause minds (J. Searle)



## Le reti neurali

Se il neurone biologico consente l'intelligenza, perché non dovrebbe consentire l'intelligenza artificiale un neurone sintetico?

“.. a neural network is a system composed of *many simple processing elements* operating in *parallel* whose function is determined by *network structure, connection strengths*, and the *processing performed at computing elements* or nodes. ... Neural network architectures are inspired by the architecture of biological nervous systems, which use many simple processing elements operating in parallel to obtain high computation rates”. (DARPA, 1988)....



## A cosa servono?



Le reti neurali offrono i seguenti specifici vantaggi nell'elaborazione dell'informazione:

- Apprendimento basato su esempi (non è richiesta l'elaborazione di un modello aderente alla realtà)
- Autoorganizzazione dell'informazione nella rete
- Robustezza ai guasti (codifica ridondante dell'informazione)
- Funzionamento in tempo reale (realizzazione HW)
- Basso consumo (0.5nW ÷ 4nW per neurone, 20W per il SN).



A.A. 2005-2006

5/43

<http://homes.dsi.unimi.it/~borgnese>



## Cosa sono le reti neurali artificiali?



- Le reti neurali sono algoritmi non lineari per l'**approssimazione** di soluzioni di problemi dei quali non esiste un modello preciso (o se esiste è troppo oneroso computazionalmente), mediante l'utilizzo di esempi (dati e uscite) oppure per classificazioni. Connessioni con il dominio della statistica. *Modelli semi-parametrici.*
- Sono un capitolo importante negli argomenti di intelligenza artificiale.
- Da un altro punto di vista possono essere utilizzate per lo studio delle reti neurali naturali, ovvero dei processi cognitivi.

A.A. 2005-2006

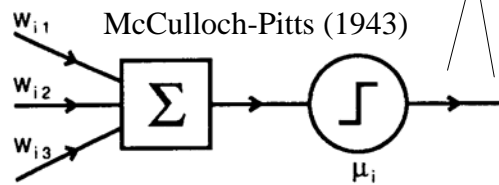
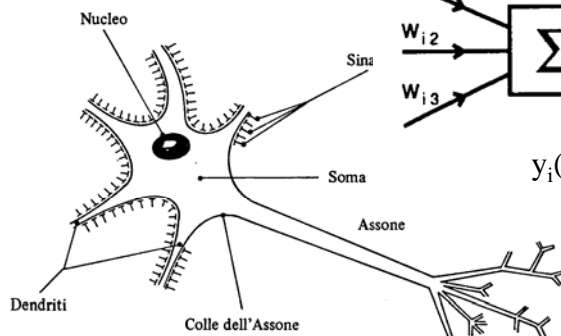
6/43

<http://homes.dsi.unimi.it/~borgnese>



## Il neurone artificiale

- *Potenziale di azione (tutto o nulla).*
- *Integrazione nel soma.*
- *Soglia di attivazione.*



$$y_i(t+1) = \Theta(w_{ij}u_j(t) - \mu_i)$$

$$\Theta(x) = \begin{cases} 1 & \text{se } x \geq 0 \\ 0 & \text{altrimenti} \end{cases}$$

$$-1 < x < +1$$

Neurone come elemento di calcolo universale: in grado di calcolare qualsiasi funzione logica (cioè implementabile in un computer).

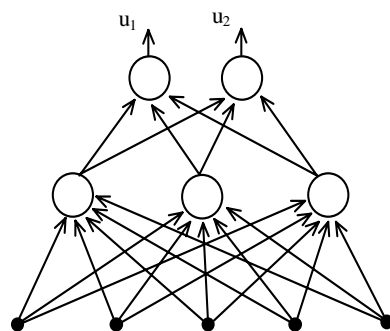
A.A. 2005-2006

7/43

<http://homes.dsi.unimi.it/~borgnese>



## Una rete neurale



Livelli di unità di attivazione

Collegamento in cascata

Capacità di approssimazione universale

Perceptrone: layered networks

A.A. 2005-2006

8/43

<http://homes.dsi.unimi.it/~borgnese>



## Costituenti delle reti neurali



*Un neurone artificiale è costituito da:*

- Un insieme di input (provienienti da altri neuroni)
- Un peso che rappresenta l'efficacia ed il segno della sinapsi.
- Una funzione di attivazione che trasforma gli input nell'output del neurone.

*Una rete neurale è costituita da:*

- Un insieme di neuroni artificiali.
- La connettività tra neuroni.

$$y_i(t+1) = \Theta(w_{ij}u_j(t) - \mu_i)$$



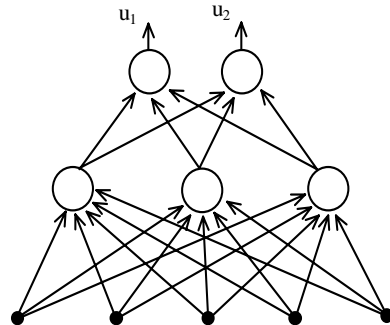
## Critica al modello di McCulloch-Pitts



- I neuroni reali non possono essere ridotti ad un dispositivo a soglia. Lo spike ha la sua forma continua che ha una durata di qualche millisecondo.
- Il tempo di propagazione lungo i dendriti non viene considerato.
- La variazione delle forma d'onda del potenziale di membrana lungo il dendrita non viene considerata.
- Gli input non sono sincroni.
- Le interazioni tra input non sono lineari.
- I pesi sono supposti costanti.



## Le famiglie di modelli



Spiking neurons. Sono neuroni la cui uscita è il singolo spike. Modellazione realistica.

Connessionismo classico. Uscita compresa tra min – Max. Frequenza di scarica.



## Sommario



Dal neurone artificiale alle reti neurali

**L'apprendimento in reti di perceptroni**

Esempio con unità lineari

Modelli neurali ed agenti



## I vari tipi di apprendimento



**Supervisionato** (learning with a teacher). Viene specificato per ogni pattern di input, il pattern desiderato in output.

**Non-supervisionato** (learning without a teacher). I neuroni verranno associati a pattern di ingresso contigui. Clustering. Mappe neurali.

**Apprendimento con rinforzo** (reinforcement learning, learning with a distal teacher). L'ambiente fornisce un'informazione del tipo success or fail.

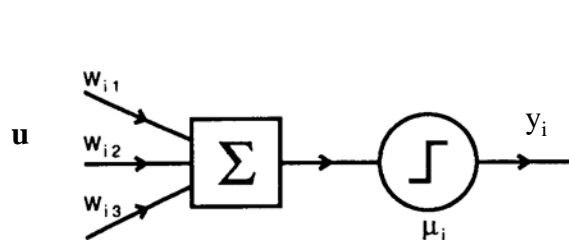
A.A. 2005-2006

13/43

<http://homes.dsi.unimi.it/~borgnese>

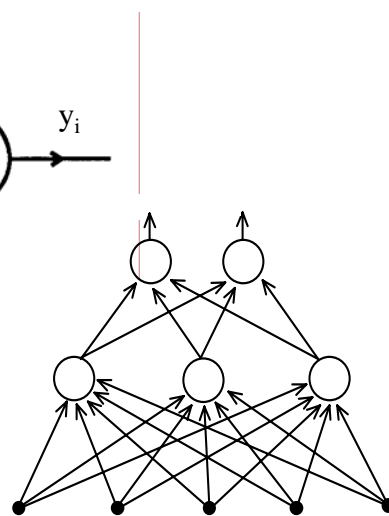


## Le reti con unità non-lineari



$$y_i = g(w_{ij}u_j - \mu_i)$$

$g(\cdot)$  continua e differenziabile.



A.A. 2005-2006

14/43

<http://homes.dsi.unimi.it/~borgnese>

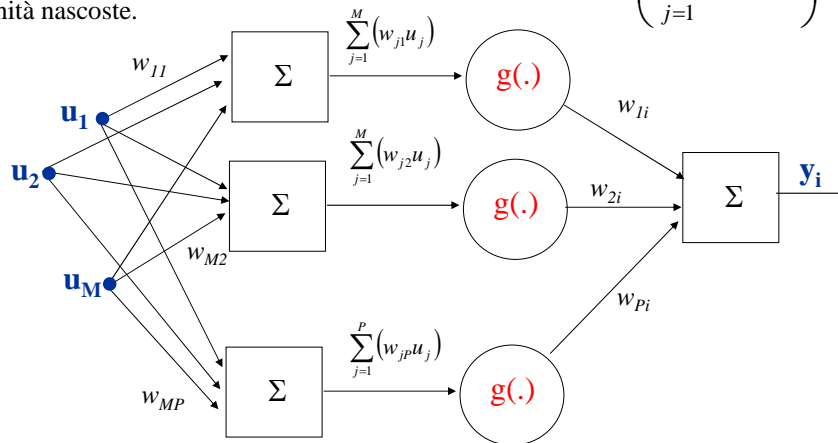


## MLP : Multi-layer Perceptron



- M intressi
- N uscite
- P unità nascoste.

$$h_i = g\left(\sum_{j=1}^M (w_{ij}u_j)\right)$$



A.A. 2005-2006

15/43

<http://homes.dsi.unimi.it/~borgnese>



## Complessità della funzione realizzabile

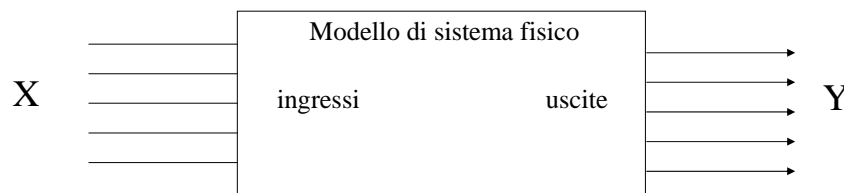


Quanti più neuroni artificiali vengono connessi tanto più la funzione complessiva approssimabile diviene più complessa

$$Y = |y_1, y_2, y_3, \dots, y_n|^T$$

$$y_i = g(X)$$

$$X = |x_1, x_2, x_3, \dots, x_m|^T$$



Reti neurali = approssimatori universali.

A.A. 2005-2006

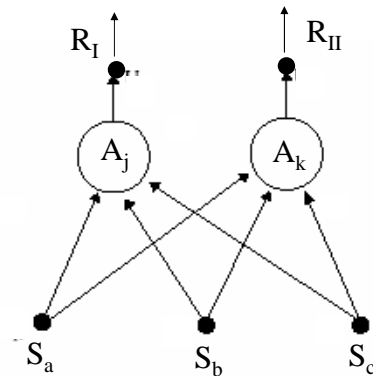
16/43

<http://homes.dsi.unimi.it/~borgnese>





## Apprendimento



Apprendimento è la modifica dei parametri  $\{w_{ij}\}$  e  $\{\mu_j\}$  in modo tale che la rete neurale approssimi la trasformazione tra i pattern di input e di output.

$$y_i = g(w_{ij}u_j - \mu_i)$$

A.A. 2005-2006

17/43

<http://homes.dsi.unimi.it/~borgnese>



## Apprendimento supervisionato

$$\min_{\{w\}} J(.) \quad J = \|Y^D - g(W^{nuovo}U)\| \leq \|Y^D - g(W^{vecchio}U)\|$$

$Y^D$  è l'uscita desiderata nota.

- Si tratta di un problema di minimizzazione di una cifra di merito ( $J$ ) sullo spazio di parametri  $W$ .

### **Soluzione iterativa:**

Obiettivo: se esiste una soluzione, trovare  $\Delta W$  in modo iterativo tale che l'insieme dei pesi  $W^{nuovo}$  ottenuto come:

$$W^{nuovo} = W^{vecchio} + \Delta W$$

dia luogo a un errore sulle uscite di norma minore che con  $W^{vecchio}$

A.A. 2005-2006

18/43

<http://homes.dsi.unimi.it/~borgnese>



## Minimizzazione di funzioni di più variabili



$\min(J\{\mathbf{w}\} | \dots)$  funzione costo od errore

$$\text{Gradiente: } \mathbf{g}(\mathbf{w}) = \frac{\partial J(\{w\} | \dots)}{\partial w_1} \frac{w_1}{|w_1|} + \frac{\partial J(\{w\} | \dots)}{\partial w_2} \frac{w_2}{|w_2|} +$$

$$\frac{\partial J(\{w\} | \dots)}{\partial w_3} \frac{w_3}{|w_3|} + \frac{\partial J(\{w\} | \dots)}{\partial w_4} \frac{w_4}{|w_4|} + \dots$$

Modifico il valore dei pesi di una quantità proporzionale alla pendenza della funzione costo rispetto a quel parametro.

Estensione della tecnica del gradiente a più variabili.

$$\Delta \mathbf{w} = -\eta \mathbf{g} \Leftrightarrow \Delta w_{ij} = -\eta g_{ij} \Leftrightarrow \Delta w_{ij} = -\eta \frac{\partial J(\{w\} | \dots)}{\partial w_{ij}}$$

Serve un' **approssimazione iniziale** per i pesi  $W_{ini} = \{w_j\}_{ini}$ .

A.A. 2005-2006

19/43

<http://homes.dsi.unimi.it/~borgnese>



## Apprendimento supervisionato tramite gradiente



Coppie input/output note.

Definizione di una funzione costo che misuri l'errore sull'uscita.

Modifica dei valori dei pesi in modo tale che la funzione costo sia minimizzata.

Reti multi-strato hanno elevata capacità computazionale, ma anche elevata complessità.

A.A. 2005-2006

20/43

<http://homes.dsi.unimi.it/~borgnese>

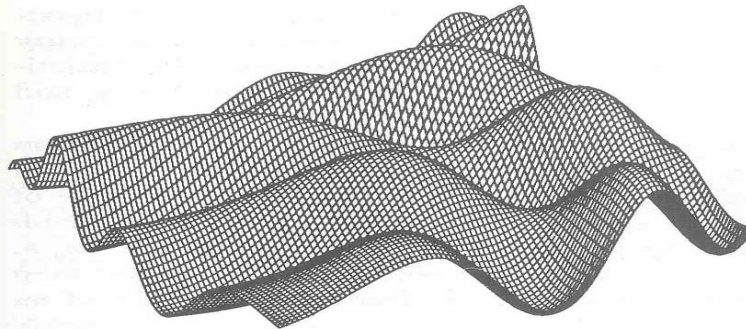


## Problemi nell'apprendimento supervisionato tramite gradiente



•Nota:  $W_{ini}$  è generalmente casuale e può condizionare la convergenza degli algoritmi iterativi.

•I problemi di convergenza sono legati all'esistenza di minimi locali del funzionale  $J(w | \dots)$



A.A. 2005-2006

21/43

<http://homes.dsi.unimi.it/~borgnese>



## La pratica dell'apprendimento supervisionato



Fino a quando l'apprendimento non è stato completato:

1. Presentazione di un pattern di input / output.
2. Calcolo dell'output della rete con il pattern corrente.
3. Calcolo dell'incremento dei pesi.

Aggiornamento dei pesi.

Aggiornamento dei pesi:

- Per trial (ogni pattern)
- Per epoca (ogni insieme di pattern).

A.A. 2005-2006

22/43

<http://homes.dsi.unimi.it/~borgnese>



## Perceptrone con unità di attivazione continue



Possiamo derivare una regola di apprendimento di spirito Hebbiano per una qualsiasi funzione di attivazione continua

$$y = g\left(\sum_{j=1} w_{ij} u_j - \mu_i\right) = g\left(\sum_{j=0} (w_{ij} u_j)\right)$$

Si tratta di un problema di minimizzazione di una cifra di merito,  $J$ , sullo spazio di parametri  $W$ :

$$J = \left\| \underbrace{y^D - g(W^{nuovo}U)}_{\text{Errore}} \right\| \leq \left\| y^D - g(W^{vecchio}U) \right\|$$

Errore

Devo trovare  $\{w\}$  :  $E(w)$  è minimo.

$$J = E(\mathbf{w}) = \frac{1}{2} \sum_p \left[ \sum_j (y_{jp}^D - y_{jp})^2 \right] = \frac{1}{2} \sum_p \left[ \sum_j \left( y_{jp}^D - g\left(\sum_i w_{ij} u_{ip}\right) \right)^2 \right]$$

A.A. 2005-2006

23/43

<http://homes.dsi.unimi.it/~borgnese>



## Sommario



Dal neurone artificiale alle reti neurali

L'apprendimento in reti di perceptroni

**Esempio con unità lineari**

Modelli neurali ed agenti

A.A. 2005-2006

24/43

<http://homes.dsi.unimi.it/~borgnese>



## Unità di attivazione lineari



$$y_j = g\left(\sum_{i=1}^M w_{ij}u_i - \mu_j\right) = g\left(\sum_{i=0}^M (w_{ij}u_i)\right)$$

*Caso lineare (g = 1):*

$$y_j = \sum_i w_{ij}u_i - \mu_j = \sum_i (w_{ij}u_i) \quad \implies \quad \mathbf{Y} = \mathbf{W} \mathbf{U}$$

Soluzione di un sistema lineare nei pesi!!

Condizione di risolubilità:  $\mathbf{W}$  di rango massimo  $\rightarrow$   
 $\{w\}$  sono linearmente indipendenti.



## Unità lineari, soluzione iterativa



$$J = E(\mathbf{w}) = \frac{1}{2} \sum_p \left[ \sum_j (y_{jp}^D - y_{jp})^2 \right] = \frac{1}{2} \sum_p \sum_j \left( y_{jp}^D - \left( \sum_i w_{ij}u_{ip} \right) \right)^2$$

$$\Delta w_{ij} = -\eta \frac{\partial}{\partial w_{ij}} \frac{1}{2} \sum_j \left( y_j^D - \left( \sum_i w_{ij}u_i \right) \right)^2$$

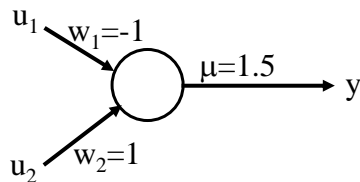
$$\Delta w_{ij} = +\eta \sum_j \left( y_j^D - \left( \sum_j w_{ij}u_i \right) \right) u_i = +\eta (y_j^D - y_j) u_i$$

Hebbian learning

$\delta$  rule (Hoff, 1960)



## Esempio di delta rule - I



$$U = \{-1, 1\} \quad y^D = -1 \\ \eta = 0.2$$

$u_1$	$u_2$	$y^D$
-1	-1	-1
-1	1	-1
1	-1	-1
1	1	1

$$y = \sum_{i=1} w_i u_i - \mu = \sum_{i=0} (w_i u_i) = (-1)(-1) + (1)(1) - 1.5 = 0.5 \gg -1 \\ u_0 = -1 \quad w_0 = \mu$$

A.A. 2005-2006

27/43

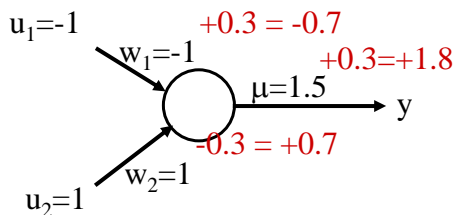
<http://homes.dsi.unimi.it/~borgnese>



## Esempio di delta rule - II



$$U = \{-1, 1\} \quad y^D = -1 \\ \eta = 0.2$$



$$y = \sum_{i=1} (w_i u_i - \mu) = \\ \sum_{i=0} (w_i u_i) = -0.4 > -1$$

$$\Delta w_{ij} = +\eta (y_j^D - y_j) u_i$$

$$\Delta \mu = \Delta w_0 = \eta (y_i^D - y_i) u_0 = \eta (-1 - 0.5)(1) = +0.30$$

$$\Delta w_1 = \eta (y_i^D - y_i) u_1 = \eta (-1 - 0.5)(-1) = +0.30$$

$$\Delta w_2 = \eta (y_i^D - y_i) u_2 = \eta (-1 - 0.5)(1) = -0.30$$

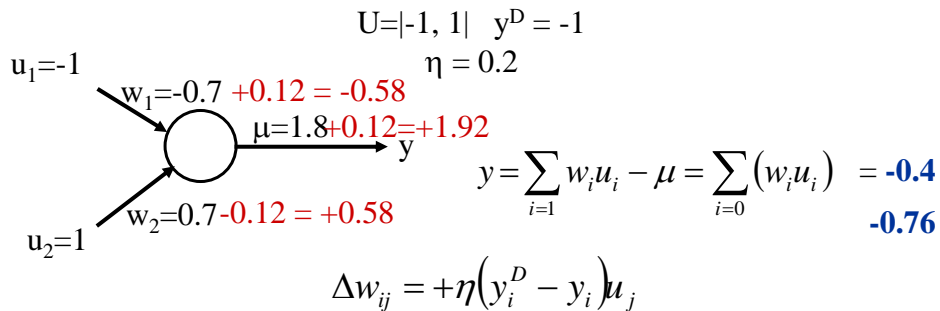
A.A. 2005-2006

28/43

<http://homes.dsi.unimi.it/~borgnese>



## Esempio di delta rule - III



$$\Delta \mu = \Delta w_0 = \eta (y_i^D - y_i) u_0 = \eta (-1 - (-0.4)) (1) = +0.12$$

$$\Delta w_1 = \eta (y_i^D - y_i) u_1 = \eta (-1 - (-0.4)) (-1) = +0.12$$

$$\Delta w_2 = \eta (y_i^D - y_i) u_2 = \eta (-1 - (-0.4)) (1) = -0.12$$

A.A. 2005-2006

29/43

<http://homes.dsi.unimi.it/~borgnese>



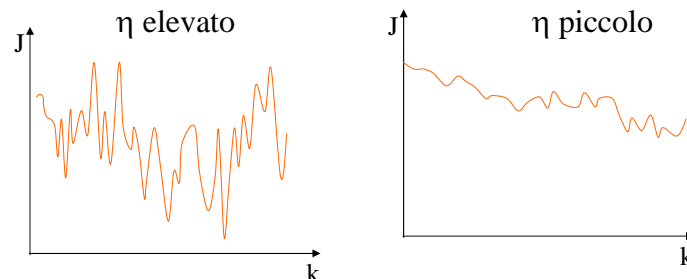
## Ruolo di $\eta$ - learning rate



$$\Delta w_{ij} = +\eta (y_j^D - y_j) u_i$$

Calmiera il  $\Delta w_{ij}$  per evitare che :

- Un peso sia specifico di un'unità ingresso-uscita.
- Oscillazioni durante l'apprendimento senza convergenza.

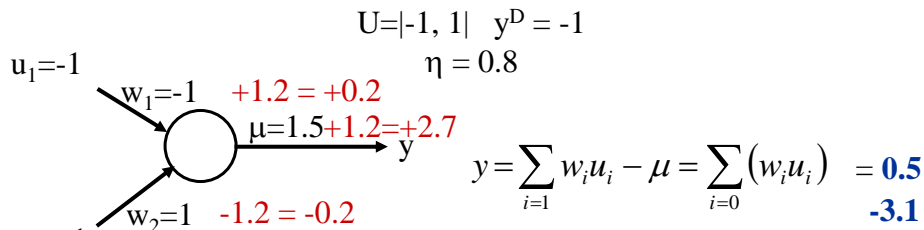


A.A. 2005-2006

□  $\eta$  può variare durante l'addestramento. [p://homes.dsi.unimi.it/~borgnese](http://homes.dsi.unimi.it/~borgnese)



## Esempio di delta rule - Cattiva scelta di $\eta$



$$\Delta w_{ij} = +\eta(y_j^D - y_j)u_i$$

$$\Delta \mu = \Delta w_0 = \eta(y_i^D - y_i)u_0 = \eta(-1 - 0.5)(1) = +1.2$$

$$\Delta w_1 = \eta(y_i^D - y_i)u_1 = \eta(-1 - 0.5)(-1) = +1.2$$

$$\Delta w_2 = \eta(y_i^D - y_i)u_2 = \eta(-1 - 0.5)(1) = -1.2$$

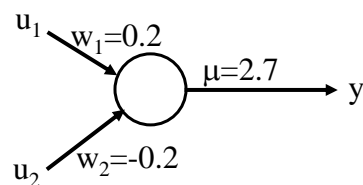
A.A. 2005-2006

31/43

<http://homes.dsi.unimi.it/~borgnese>



## Esempio di specializzazione sui pattern a, b, c



$u_1$	$u_2$	$y^D$	
-1	-1	-1	a
-1	1	-1	b
1	-1	-1	c
1	1	1	d

a  $y = \sum_{i=1} w_i u_i - \mu = \sum_{i=0} (w_i u_i) = (0.2)(-1) + (-0.2)(1) - 2.7 = -3.1$

b  $y = \sum_{i=1} w_i u_i - \mu = \sum_{i=0} (w_i u_i) = (0.2)(-1) + (-0.2)(1) - 2.7 = -2.9$

c  $y = \sum_{i=1} w_i u_i - \mu = \sum_{i=0} (w_i u_i) = (0.2)(1) + (-0.2)(-1) - 2.7 = -2.3$

d  $y = \sum_{i=1} w_i u_i - \mu = \sum_{i=0} (w_i u_i) = (0.2)(1) + (-0.2)(1) - 2.7 = -2.7$

Errato su d

A.A. 2005-2006

32/43

<http://homes.dsi.unimi.it/~borgnese>





## Unità non-lineari, soluzione iterativa



$$J = E(\mathbf{w}) = \frac{1}{2} \sum_p \left[ \sum_j (y_{jp}^D - y_{jp})^2 \right] = \frac{1}{2} \sum_p \left[ \sum_j \left( y_{jp}^D - g\left(\sum_i w_{ij} u_{ip}\right) \right)^2 \right]$$

$$\Delta w_{ijp} = -\eta \frac{\partial}{\partial w_{ij}} \frac{1}{2} \sum_j \left( y_{jp}^D - g\left(\sum_i w_{ij} u_{ip}\right) \right)^2 =$$

$$\eta \sum_j \left( y_{jp}^D - g\left(\sum_i w_{ij} u_{ip}\right) \right) g'\left(\sum_i w_{ij} u_{ip}\right) u_i = +\eta \left( y_{jp}^D - y_{jp} \right) g'\left(\sum_i w_{ij} u_{ip}\right) u_{ip}$$

$\delta$  rule

A.A. 2005-2006

33/43

<http://homes.dsi.unimi.it/~borgnese>



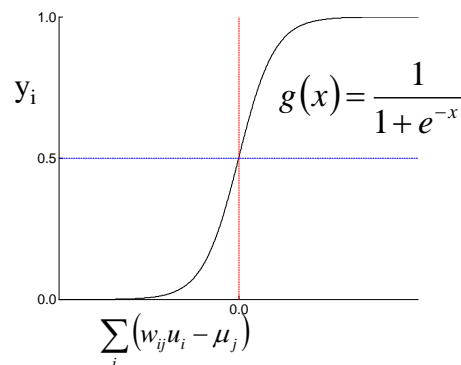
## Perceptrone con unità di attivazione logistiche



$$g'(x) = g(x) \cdot (1 - g(x))$$

$$y_j = g\left(\sum_i w_{ij} u_i - \mu_j\right)$$

$$g'(x) = \frac{e^{-x}}{(1 + e^{-x})^2} = \frac{1}{1 + e^{-x}} \left(1 - \frac{1}{1 + e^{-x}}\right)$$



A.A. 2005-2006

34/43

<http://homes.dsi.unimi.it/~borgnese>



## Update dei pesi per funzione logistica



$$J = E(\mathbf{w}) = \frac{1}{2} \sum_p \left[ \sum_j (y_{jp}^D - y_{jp})^2 = \frac{1}{2} \sum_j \left( y_{jp}^D - g\left(\sum_i w_{ij} u_{ip}\right) \right)^2 \right]$$

$$\Delta w_{ijp} = +\eta \sum_j (y_{jp}^D - g(\cdot)) g'(\cdot) u_i = +\eta \underbrace{(y_{jp}^D - y_j)}_{\delta \text{ rule}} y_j (1 - y_j) u_{ip}$$

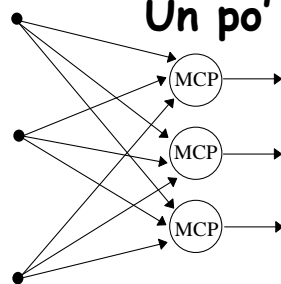
$\delta$  rule

NB  $y_i \in [0, 1]$ . Per  $y_i = 0$  o  $y_i = 1$  non c'è apprendimento anche se l'uscita è sbagliata. Quando si verifica questa situazione?

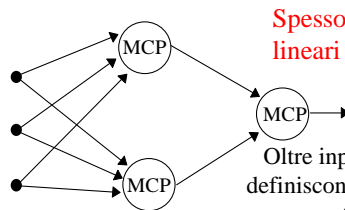
Si cerca di mantenere le unità lontane dalla saturazione.



## Un po' di tassonomia



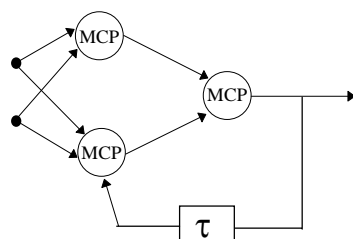
Perceptrone semplice



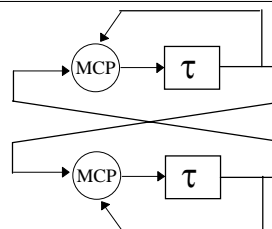
Perceptrone multistrato

Spesso unità lineari

Oltre input/output si definiscono anche unità nascoste (**hidden units**)



Ricorrente



Ricorrente completamente connessa: autoassociativa (ingresso=stato)



## Riassunto - topologia



I neuroni connessioneisti sono basati su:

- Ricevere una somma pesata degli ingressi.
- Trasformarla secondo una funzione non-lineare (scalino o logistica)
- Inviare il risultato di questa funzione all'uscita o ad altre unità.

Le reti neurali sono topologie ottenute connettendo tra loro i neuroni in modo opportuno e riescono a calcolare funzioni molto complesse.



## Riassunto - Apprendimento



Algoritmi iterativi per adattare il valore dei parametri (pesi).

Definizione di una funzione costo che misura la differenza tra valore fornito e quello desiderato.

Algoritmo (gradiente) che consente di aggiornare i pesi in modo da minimizzare la funzione costo.

Training per pattern (specializzazione) o per epoche.



## Problemi



### *Quando si termina l'algoritmo di apprendimento?*

Bootstrap – Vengono estratti pattern con ripetizioni.

Cross-Validation - Errore sull'insieme di training =  
Errore sull'insieme di test.

Utilizzare lo “structural risk” invece dell’”empirical risk”.

*Si vuole evitare che la rete si specializzi troppo sui pattern di training e non sia in grado di interpolare.*



## Sommario

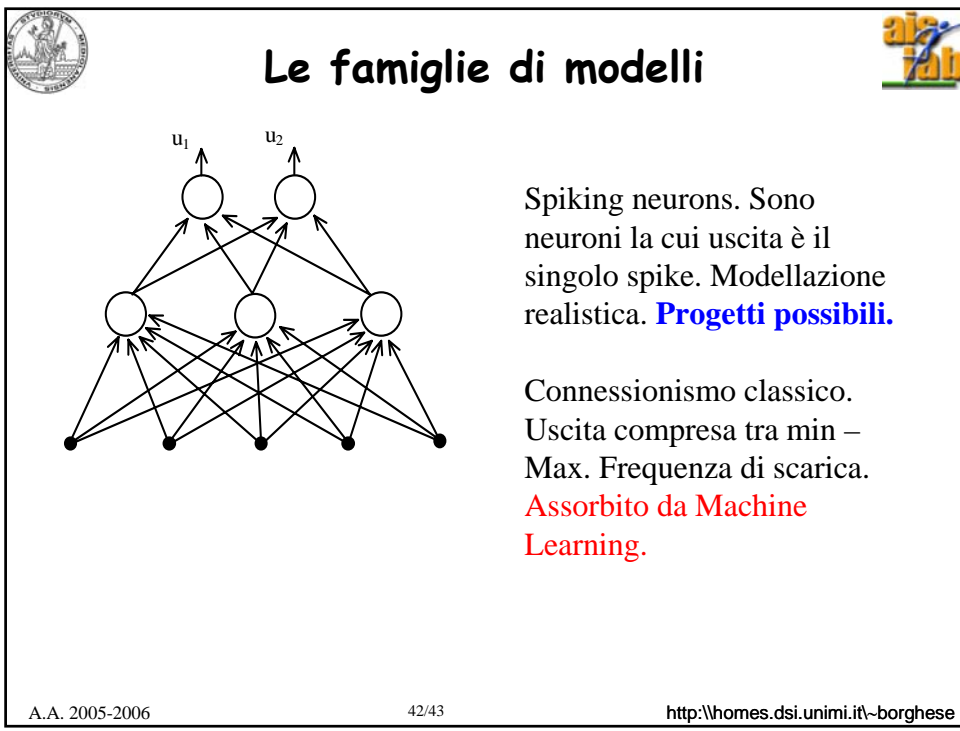
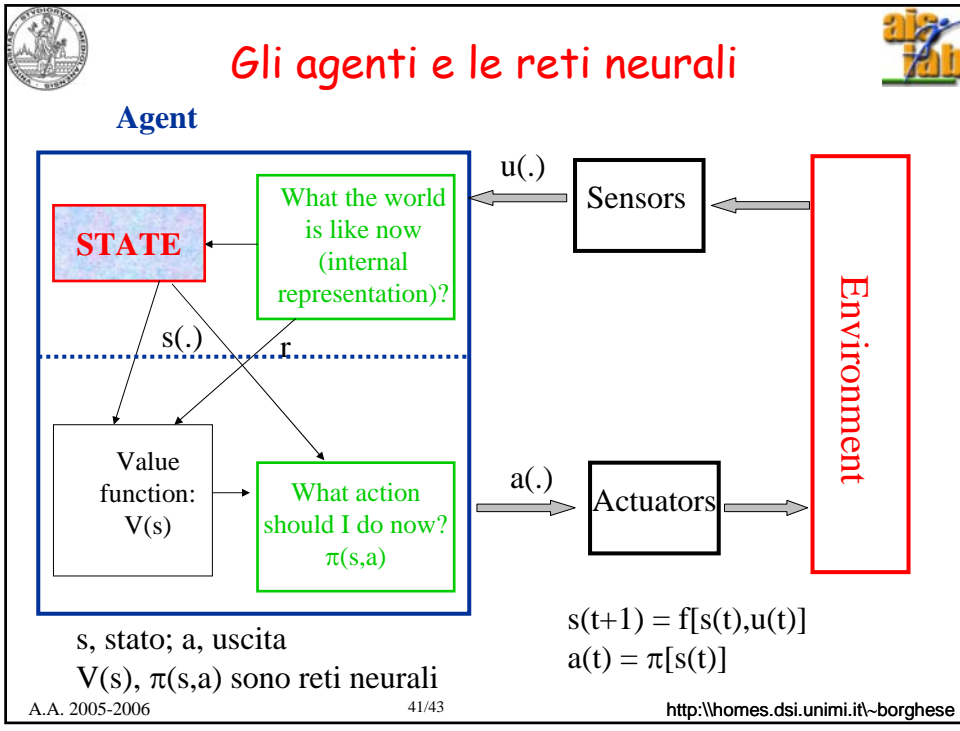


Dal neurone artificiale alle reti neurali

L'apprendimento in reti di perceptroni

Esempio con unità lineari

**Modelli neurali ed agenti**





## Sommario



Dal neurone artificiale alle reti neurali

L'apprendimento in reti di perceptroni

Esempio con unità lineari

Modelli neurali ed agenti