

## Apprendimento di funzioni linearmente separabili

Assumiamo di avere esempi di funzioni linearmente separabili.

### Algoritmo di Apprendimento per il Perceptron

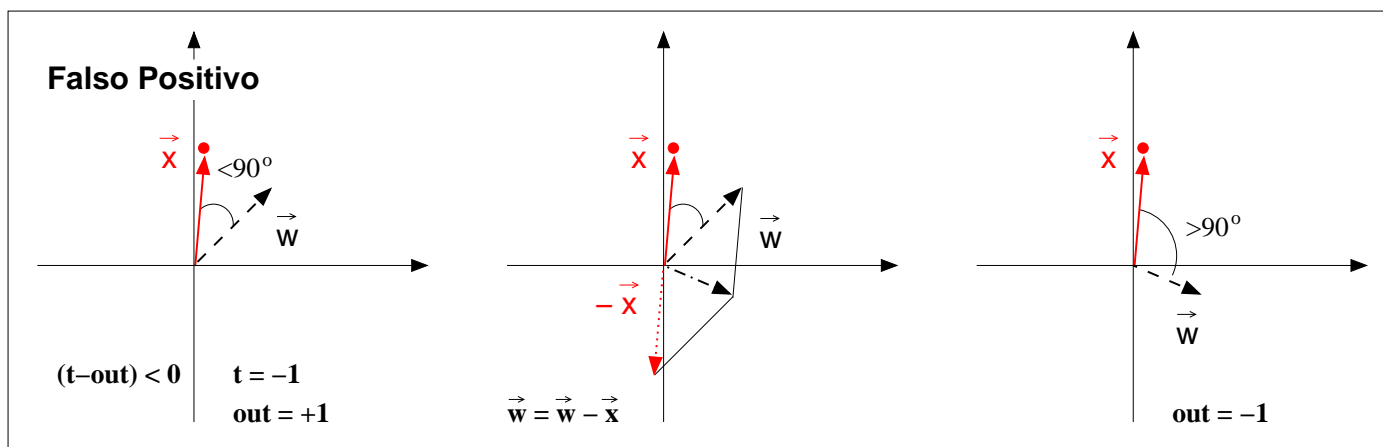
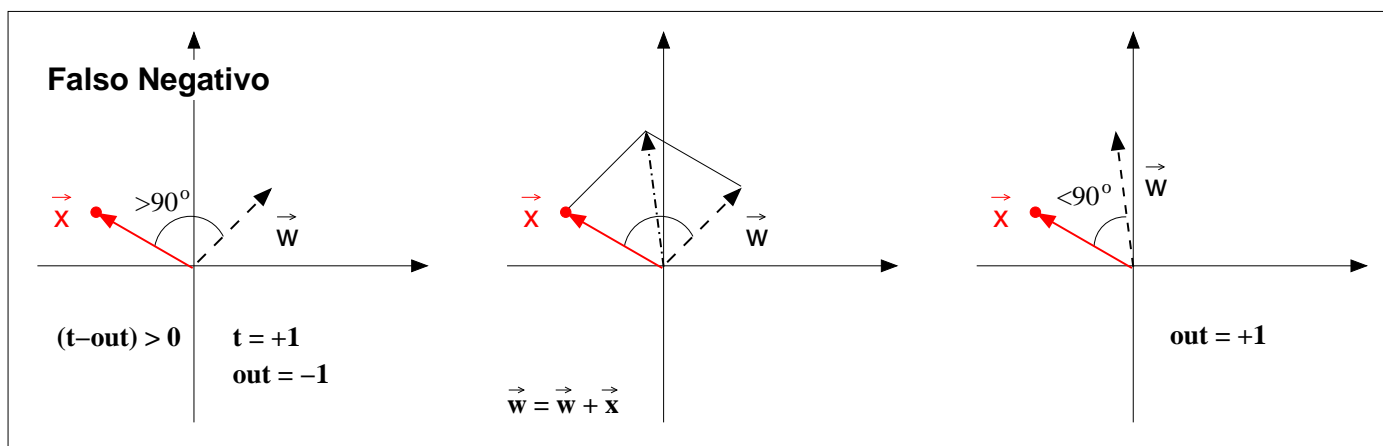
ingresso: insieme di apprendimento  $Tr = \{(\vec{x}, t)\}$ , dove  $t \in \{-1, +1\}$

1. inizializza il vettore dei pesi  $\vec{w}$  al vettore nullo (tutte le componenti a 0);
2. **ripeti**
  - (a) seleziona (a caso) uno degli esempi di apprendimento  $(\vec{x}, t)$
  - (b) **se**  $out = \text{sign}(\vec{w} \cdot \vec{x}) \neq t$  **allora**

$$\vec{w} \leftarrow \vec{w} + (t - out)\vec{x}$$

# Apprendimento per Perceptron

## Interpretazione geometrica



## Apprendimento per Perceptron

Non necessariamente un singolo passo di apprendimento 2(b) riuscirà a modificare il segno dell'output: potrebbero servire molti passi.

Per rendere più stabile l'apprendimento si aggiunge un coefficiente di apprendimento  $\eta$

$$\vec{w} \leftarrow \vec{w} + \eta(t - out)\vec{x}$$

con  $\eta > 0$  e preferibilmente  $\eta < 1$

Questo evita che il vettore dei pesi subisca variazioni troppo "violente" ogni volta che il passo 2(b) viene eseguito, e quindi cerca di evitare che esempi precedentemente ben classificati diventino classificati erroneamente a causa della forte variazione del vettore dei pesi.

Se l'insieme di apprendimento è LINEARMENTE SEPARABILE, si dimostra che l'algoritmo di apprendimento per il Perceptron termina con una soluzione in un numero finito di passi, altrimenti  $\vec{w}$  CICLA in un insieme di vettori peso non necessariamente ottimali (cioè che commettono il minimo numero possibile di errori)

# Apprendimento per Perceptron: esempio

$Tr:$	es.	$\vec{x}$	target
	1	(4,5)	1
	2	(6,1)	1
	3	(4,1)	-1
	4	(1,2)	-1

soglia  
→

es.	$\vec{x}'$	target
1'	(1,4,5)	1
2'	(1,6,1)	1
3'	(1,4,1)	-1
4'	(1,1,2)	-1

vettore dei pesi:  $\vec{w} = (w_0, w_1, w_2)$

supponiamo di partire con pesi "random":  $\vec{w} = (0, 1, -1)$  e  $\eta = \frac{1}{2}$

pesi	input	target	out	errore	nuovi pesi
(0,1,-1)	(1,4,5)	1			
	(1,6,1)	1			
	(1,4,1)	-1			
	(1,1,2)	-1			

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(0,1,-1)	(1,4,5)	1	-1	2	(1,5,4)
(1,5,4)	(1,6,1)	1	1	0	nessun cambiamento
(1,5,4)	(1,4,1)	-1	1	-2	(0,1,3)
(0,1,3)	(1,1,2)	-1	1	-2	(-1,0,1)

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-1,0,1)	(1,4,5)	1	1	0	nessun cambiamento
(-1,0,1)	(1,6,1)	1	? (-1)	2	(0,6,2)
(0,6,2)	(1,4,1)	-1	1	-2	(-1,2,1)
(-1,2,1)	(1,1,2)	-1	1	-2	(-2,1,-1)

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-2,1,-1)	(1,4,5)	1	-1	2	(-1,5,4)
(-1,5,4)	(1,6,1)	1	1	0	nessun cambiamento
(-1,5,4)	(1,4,1)	-1	1	-2	(-2,1,3)
(-2,1,3)	(1,1,2)	-1	1	-2	(-3,0,1)

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-3,0,1)	(1,4,5)	1	1	0	nessun cambiamento
(-3,0,1)	(1,6,1)	1	-1	2	(-2,6,2)
(-2,6,2)	(1,4,1)	-1	1	-2	(-3,2,1)
(-3,2,1)	(1,1,2)	-1	1	-2	(-4,1,-1)



## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-4,1,-1)	(1,4,5)	1	-1	2	(-3,4,5)
(-3,4,5)	(1,6,1)	1	1	0	nessun cambiamento
(-3,4,5)	(1,4,1)	-1	1	-2	(-4,1,3)
(-4,1,3)	(1,1,2)	-1	1	-2	(-5,0,1)

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-5,0,1)	(1,4,5)	1	? (-1)	2	(-4,4,6)
(-4,4,6)	(1,6,1)	1	1	0	nessun cambiamento
(-4,4,6)	(1,4,1)	-1	1	-2	(-5,0,5)
(-5,0,5)	(1,1,2)	-1	1	-2	(-6,-1,3)

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-6,-1,3)	(1,4,5)	1	1	0	nessun cambiamento
(-6,-1,3)	(1,6,1)	1	-1	2	(-5,5,4)
(-5,5,4)	(1,4,1)	-1	1	-2	(-6,1,3)
(-6,1,3)	(1,1,2)	-1	1	-2	(-7,0,1)

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-7,0,1)	(1,4,5)	1	-1	2	(-6,4,6)
(-6,4,6)	(1,6,1)	1	1	0	nessun cambiamento
(-6,4,6)	(1,4,1)	-1	1	-2	(-7,0,5)
(-7,0,5)	(1,1,2)	-1	1	-2	(-8,-1,3)

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-8,-1,3)	(1,4,5)	1	1	0	nessun cambiamento
(-8,-1,3)	(1,6,1)	1	-1	2	(-7,5,2)
(-7,5,2)	(1,4,1)	-1	1	-2	(-8,1,3)
(-8,1,3)	(1,1,2)	-1	-1	0	nessun cambiamento

## Apprendimento per Perceptron: esempio

pesi $\vec{w}$	input $\vec{x}'$	target $t$	out $out$	errore $(t - out)$	nuovi pesi $\vec{w} \leftarrow \vec{w} + \frac{1}{2}(t - out)\vec{x}'$
(-8,1,3)	(1,4,5)	1	1	0	nessun cambiamento
(-8,1,3)	(1,6,1)	1	1	0	nessun cambiamento
(-8,1,3)	(1,4,1)	-1	-1	0	nessun cambiamento
(-8,1,3)	(1,1,2)	-1	-1	0	nessun cambiamento

## Apprendimento di Reti di Perceptron

Abbiamo visto che un singolo Perceptron non riesce ad apprendere tutte le funzioni booleane (es. XOR)

Però una rete di Perceptron può implementare una qualunque funzione booleana (tramite AND, OR, NOT).

**Problema: come effettuare l'apprendimento di una rete di Perceptron ?**

Non si sa come assegnare “credito” o “colpe” alle unità nascoste:

### PROBLEMA DELL'ASSEGNAZIONE DEL CREDITO

Una possibile soluzione è quella di rendere il singolo neurone derivabile e sfruttare la tecnica di Discesa del Gradiente per apprendere i pesi “giusti”.

Vediamo, quindi, come la Discesa del Gradiente si applica ad un Perceptron “semplificato”.

## Discesa di Gradiente

Consideriamo un Perceptron SENZA la hard-threshold:

$$out(\vec{x}) = \sum_{i=0}^n w_i x_i = \vec{w} \cdot \vec{x}$$

e definiamo una misura dell'errore commesso da un particolare vettore dei pesi:

$$\text{Funzione Errore: } E[\vec{w}] = \frac{1}{2N_{Tr}} \sum_{(\vec{x}^{(i)}, t^{(i)}) \in Tr} \left( t^{(i)} - out(\vec{x}^{(i)}) \right)^2$$

dove  $N_{Tr}$  è la cardinalità dell'insieme di apprendimento  $Tr$ .

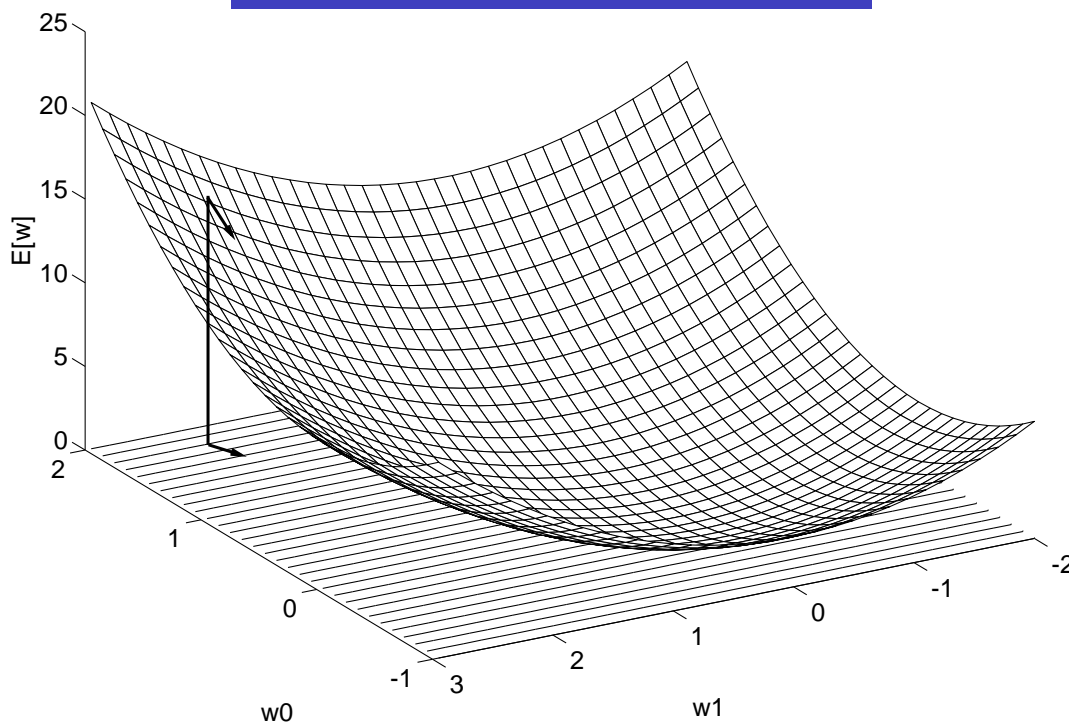
La funzione errore di sopra misura lo scarto quadratico medio (diviso 2) del valore target da quello predetto dal neurone ( $out$ ).

Ovviamente, se  $\forall (\vec{x}^{(i)}, t^{(i)}) \in Tr$  si ha  $out(\vec{x}^{(i)}) = t^{(i)} \rightarrow E[\vec{w}] = 0$

Quindi bisogna MINIMIZZARE  $E[\vec{w}]$  rispetto a  $\vec{w}$



## Discesa di Gradiente



Idea base: partire da un  $\vec{w}$  random e modificarlo nella direzione contraria al gradiente (che indica la direzione di crescita di  $E[\vec{w}]$ )

$$\underbrace{\nabla E[\vec{w}]}_{\text{gradiente}} \equiv \left[ \frac{\partial E}{\partial w_0}, \frac{\partial E}{\partial w_1}, \dots, \frac{\partial E}{\partial w_n} \right], \quad \Delta \vec{w} = -\eta \nabla E[\vec{w}], \quad \Delta w_i = -\eta \frac{\partial E}{\partial w_i}$$

## Calcolo del gradiente

$$\frac{\partial E}{\partial w_i} = \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2$$

## Calcolo del gradiente

$$\begin{aligned}\frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})^2\end{aligned}$$

## Calcolo del gradiente

$$\begin{aligned}\frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} 2(t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})\end{aligned}$$

## Calcolo del gradiente

$$\begin{aligned}\frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} 2(t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)}) \\ &= \frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - \vec{w} \cdot \vec{x}^{(d)})\end{aligned}$$

## Calcolo del gradiente

$$\begin{aligned}
 \frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2 \\
 &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})^2 \\
 &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} 2(t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)}) \\
 &= \frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - \vec{w} \cdot \vec{x}^{(d)}) \\
 \frac{\partial E}{\partial w_i} &= \frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) (-x_i^{(d)}) \\
 &= -\frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) (x_i^{(d)})
 \end{aligned}$$

## Discesa di Gradiente

### Gradient-Descent( $Tr, \eta$ )

ogni esempio di apprendimento è una coppia  $(\vec{x}, t)$ , dove  $\vec{x}$  è il vettore di valori in input, e  $t$  è il valore desiderato in output (target).  $\eta$  è il coefficiente di apprendimento (che ingloba  $\frac{1}{N_{Tr}}$ ).

- Assegna a  $w_i$  valori piccoli random
- Finché la condizione di terminazione non è verificata, fai
  - $\Delta w_i \leftarrow 0$
  - Per ogni  $(\vec{x}, t)$  in  $Tr$ , fai
    - \* Presenta  $\vec{x}$  al neurone e calcola l'output  $out$
    - \* Per ogni  $w_i$ , fai

$$\Delta w_i \leftarrow \Delta w_i + \eta(t - out)x_i$$

- Per ogni  $w_i$ , fai

$$w_i \leftarrow w_i + \Delta w_i$$

## Discesa di Gradiente con Sigmoide

Consideriamo un Perceptron con funzione sigmoidale:

$$out(\vec{x}) = \sigma\left(\sum_{i=0}^n w_i x_i\right) = \sigma(\vec{w} \cdot \vec{x})$$

dove si ricorda che  $\sigma(net) = \frac{1}{1+e^{-net}}$

Si noti che per  $\sigma()$  vale la seguente relazione

$$\sigma'(net) = \frac{d \sigma(net)}{d net} = \sigma(net)(1 - \sigma(net))$$

e ricordiamo che (derivata di funzioni composte)

$$\frac{d f(g(x))}{d x} = \frac{d f(g(x))}{d g(x)} \frac{d g(x)}{d x}$$



## Calcolo del gradiente con sigmoide

$$\begin{aligned}\frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} 2(t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})\end{aligned}$$

## Calcolo del gradiente con sigmoide

$$\begin{aligned}\frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})^2 \\ &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} 2(t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)}) \\ &= \frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - \sigma(\vec{w} \cdot \vec{x}^{(d)}))\end{aligned}$$

## Calcolo del gradiente con sigmoide

$$\begin{aligned}
 \frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2 \\
 &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})^2 \\
 &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} 2(t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)}) \\
 &= \frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - \sigma(\vec{w} \cdot \vec{x}^{(d)})) \\
 \frac{\partial E}{\partial w_i} &= \frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) \left( -\frac{\partial \sigma(\vec{w} \cdot \vec{x}^{(d)})}{\partial \vec{w} \cdot \vec{x}^{(d)}} \frac{\partial \vec{w} \cdot \vec{x}^{(d)}}{\partial w_i} \right)
 \end{aligned}$$

## Calcolo del gradiente con sigmoide

$$\begin{aligned}
 \frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)})^2 \\
 &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)})^2 \\
 &= \frac{1}{2N_{Tr}} \sum_{d \in Tr} 2(t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - out^{(d)}) \\
 &= \frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) \frac{\partial}{\partial w_i} (t^{(d)} - \sigma(\vec{w} \cdot \vec{x}^{(d)})) \\
 \frac{\partial E}{\partial w_i} &= \frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) \left( -\frac{\partial \sigma(\vec{w} \cdot \vec{x}^{(d)})}{\partial \vec{w} \cdot \vec{x}^{(d)}} \frac{\partial \vec{w} \cdot \vec{x}^{(d)}}{\partial w_i} \right) \\
 &= -\frac{1}{N_{Tr}} \sum_{d \in Tr} (t^{(d)} - out^{(d)}) \sigma(\vec{w} \cdot \vec{x}^{(d)}) (1 - \sigma(\vec{w} \cdot \vec{x}^{(d)})) x_i^{(d)}
 \end{aligned}$$