Machine learning
Reti neurali

A cura di Giulio Pons e Marco Orlandi
Share Makerspace Cusano

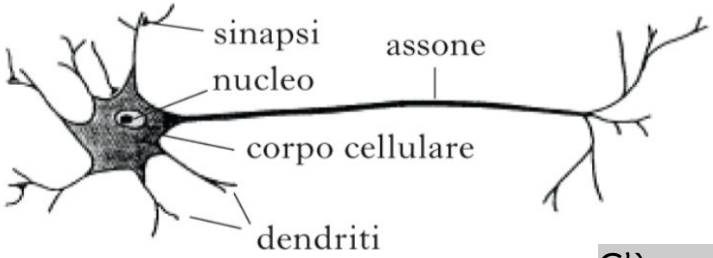
1 2

SHARE
MAKERSPACE CUSANO

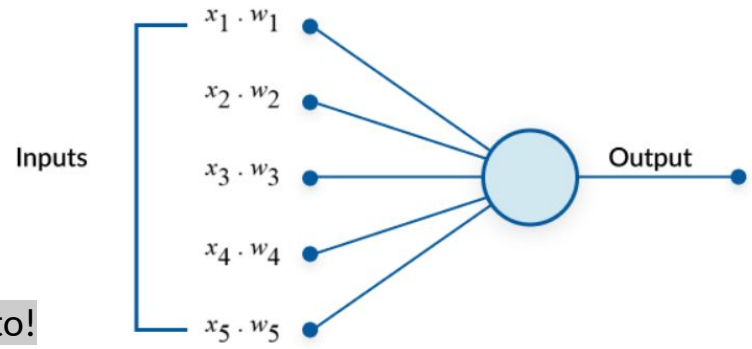
Cenni storici

L'intelligenza artificiale è una ampia disciplina dell'informatica che consente la creazione di sistemi che ad un osservatore esterno sembrano dotati di intelligenza.

- Con i primi calcolatori nasce già l'idea di creare macchine intelligenti (teoria del test di Turing, 1950)
- Le **reti neurali**, ispirate ai nostri neuroni, sono studiate e teorizzate già negli anni '60.
- Il limite della capacità di calcolo e la mole di dati necessari, blocca lo sviluppo di questa tecnologia.
- Nel 2007 con le GPU (processori grafici per i videogiochi) il limite tecnologico cade



C'è apprendimento!



Machine learning

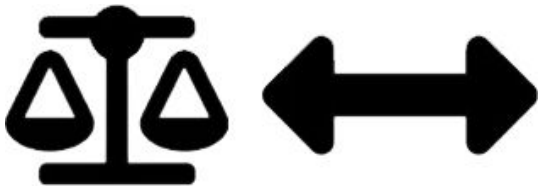
Le reti neurali permettono di affrontare un problema a partire dai dati e di ottenere così un modello (la rete neurale) che può essere interrogato con nuovi dati per ottenere delle previsioni.

La macchina quindi apprende.

Questi sviluppi hanno portato negli a brillanti risultati:

- riconoscimento ocr
- filtro antispam
- riconoscimento vocale
- riconoscimento immagini / volti
- guida autonoma
- ...

Esempio classificatore:
Gatto delle sabbie e gatto della giungla



Vogliamo riconoscere le due specie di gatto a partire da due dati: peso e lunghezza.

Dati 1

											
⚖	9	2	3,2	9,1	1,6	8,4	8	3,1	6,3	3,4	
↔	70	50	49,4	74,6	48,3	74,6	72,8	45,8	91,4	53,6	

I dati costituiscono la conoscenza, sono fissi, non modificabili.

Dati 2 - nuovo dato da classificare

											
⚖	9	2	3,2	9,1	1,6	8,4	8	3,1	6,3	3,4	2,9
⟷	70	50	49,4	74,6	48,3	74,6	72,8	45,8	91,4	53,6	55,2

SHARE
MAKERSPACE CUSANO

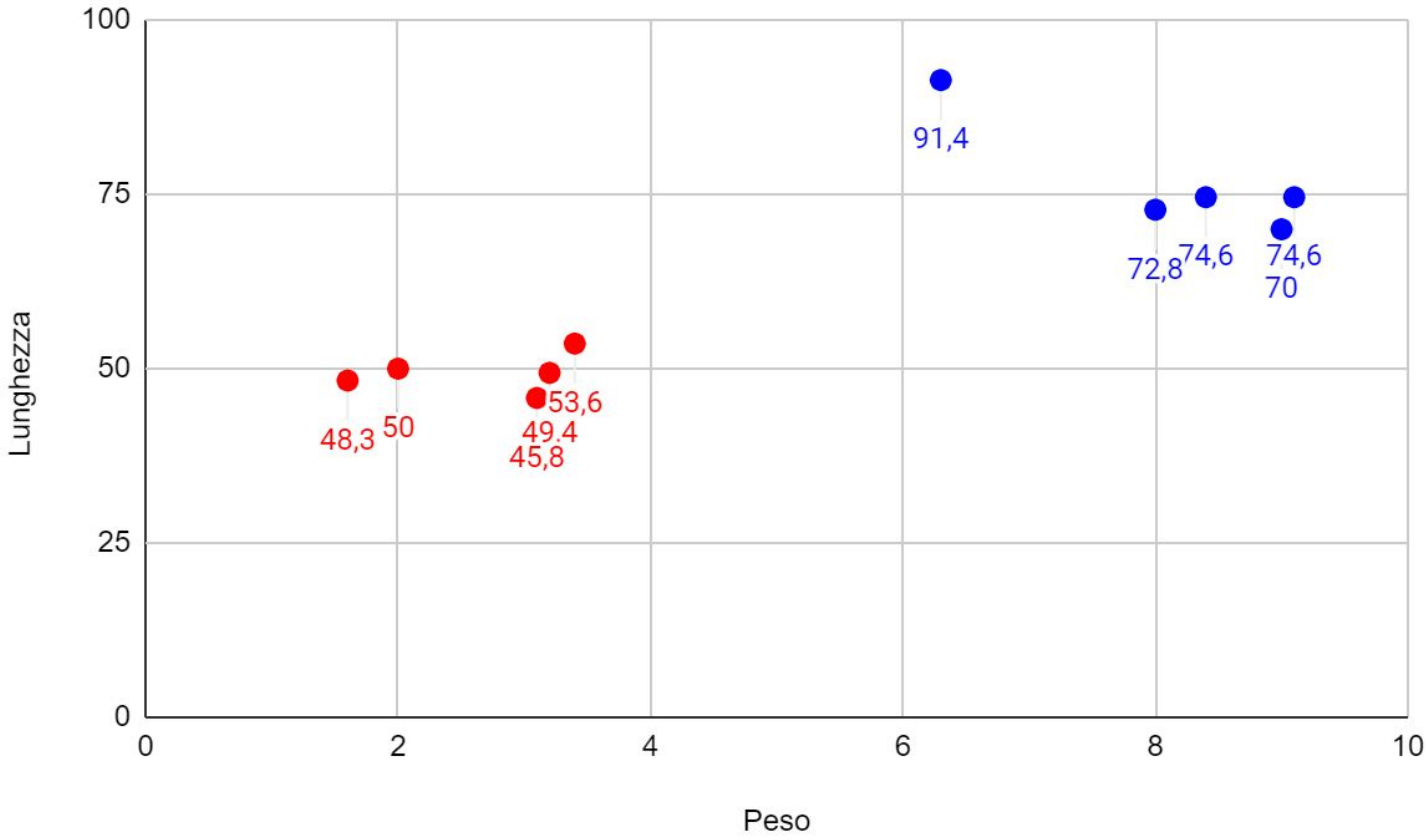
Dati 3 - osservazioni preliminari

											
⚖	9	2	3,2	9,1	1,6	8,4	8	3,1	6,3	3,4	2,9
⟷	70	50	49,4	74,6	48,3	74,6	72,8	45,8	91,4	53,6	55,2

SHARE
MAKERSPACE CUSANO

Dati 4 - visualizzazione

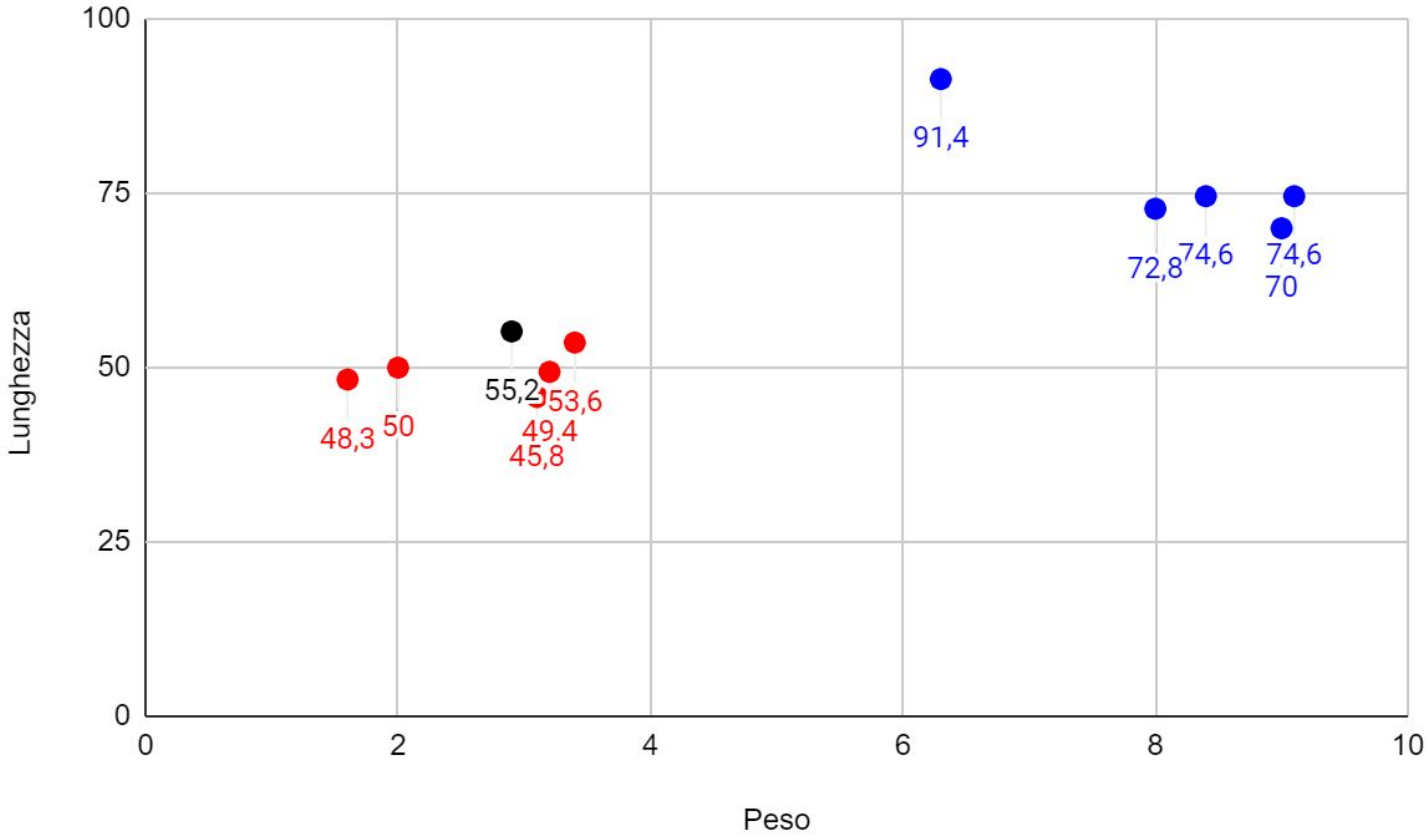
											
⚖	9	2	3,2	9,1	1,6	8,4	8	3,1	6,3	3,4	2,9
↔	70	50	49,4	74,6	48,3	74,6	72,8	45,8	91,4	53,6	55,2



SHARE MAKERSPACE CUSANO

Dati 4 - visualizzazione

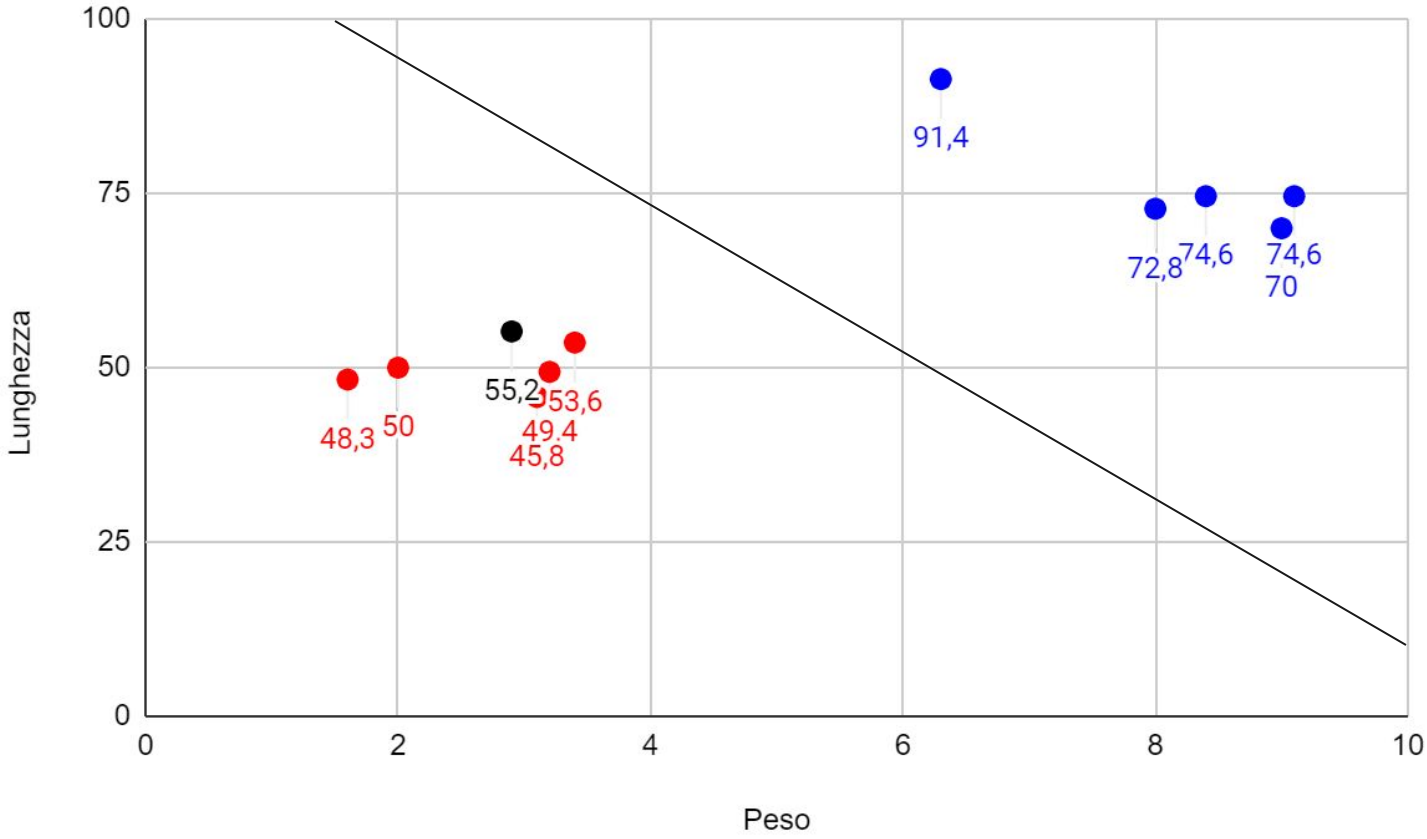
⚖	9	2	3,2	9,1	1,6	8,4	8	3,1	6,3	3,4	2,9
↔	70	50	49,4	74,6	48,3	74,6	72,8	45,8	91,4	53,6	55,2



SHARE MAKERSPACE CUSANO

Dati 4 - visualizzazione

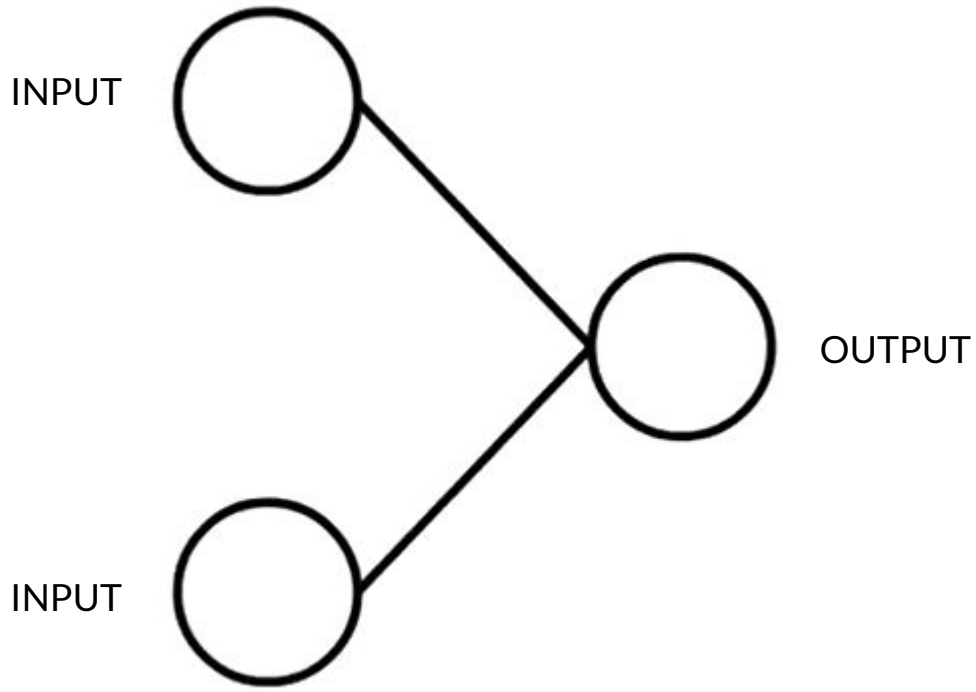
											
⚖	9	2	3,2	9,1	1,6	8,4	8	3,1	6,3	3,4	2,9
↔	70	50	49,4	74,6	48,3	74,6	72,8	45,8	91,4	53,6	55,2



SHARE MAKERSPACE CUSANO

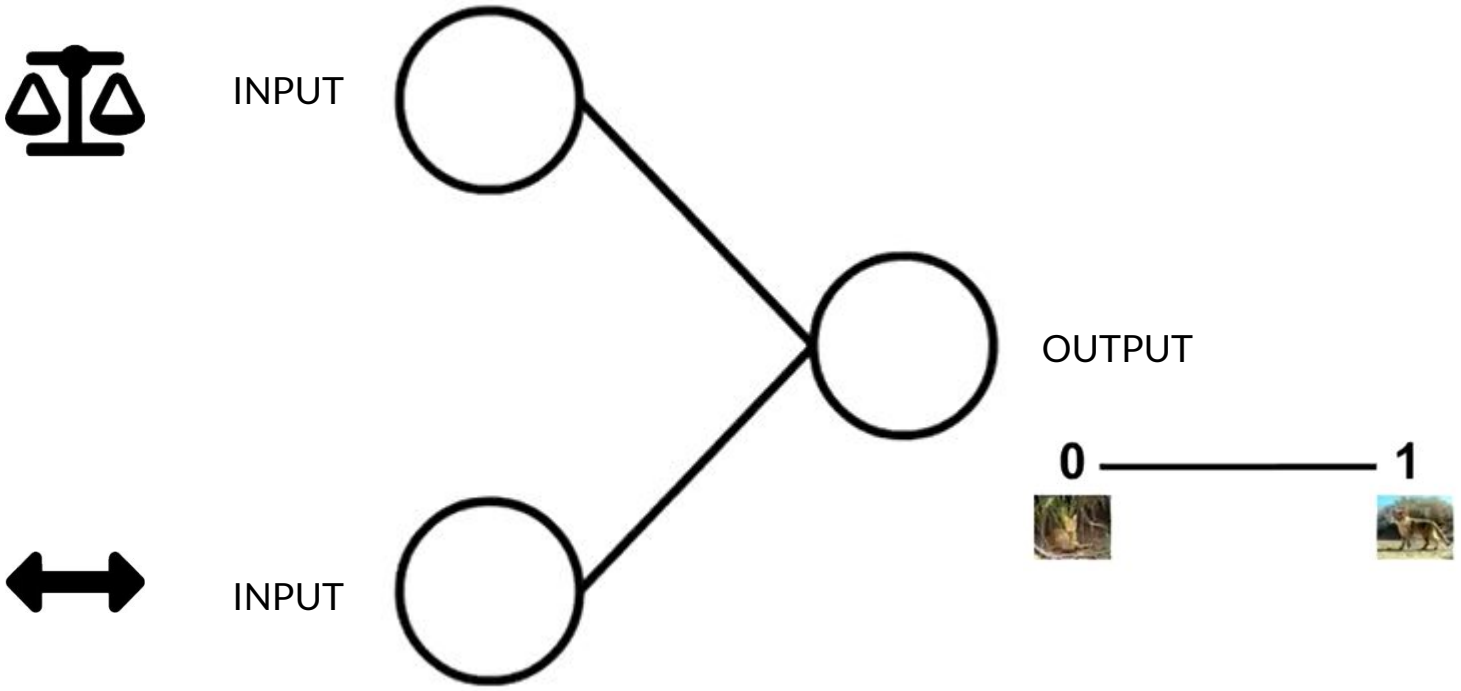
Uso le reti neurali (RN)

- Se ho migliaia di dati l'approccio "guardo sul grafico" non è applicabile
- Posso però insegnare ad una rete neurale a classificare le due specie di gatti, addestrandola prima con i dati.



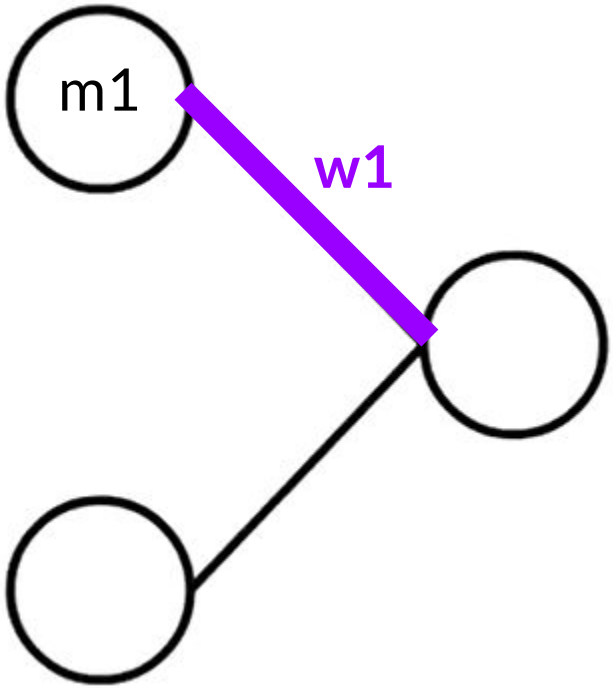
Uso le reti neurali

- la rete neurale è di fatto è un programma di un computer che calcola una **funzione** utilizzando gli ingressi e restituendo un risultato
- ho due ingressi per peso e lunghezza e un'uscita con cui voglio classificare il gatto (della giungla 0 o delle sabbie 1)



Teoria: Funzione RN

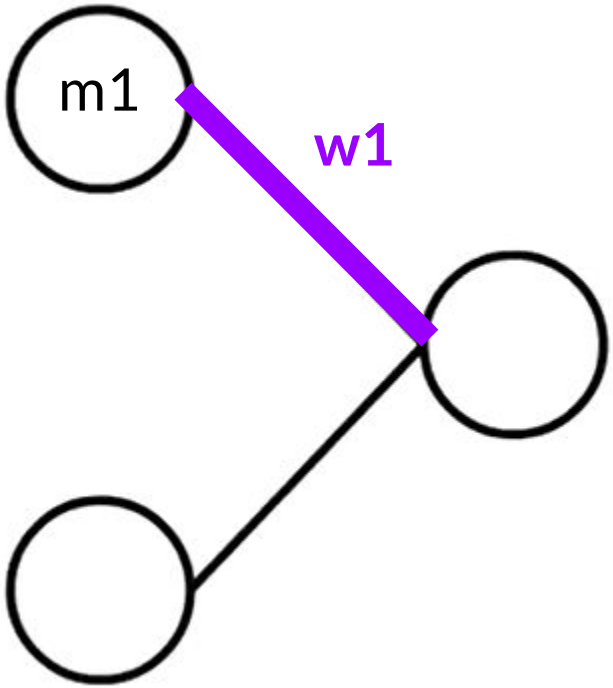
m = measure
w = weight



SHARE
MAKERSPACE CUSANO

Funzione RN

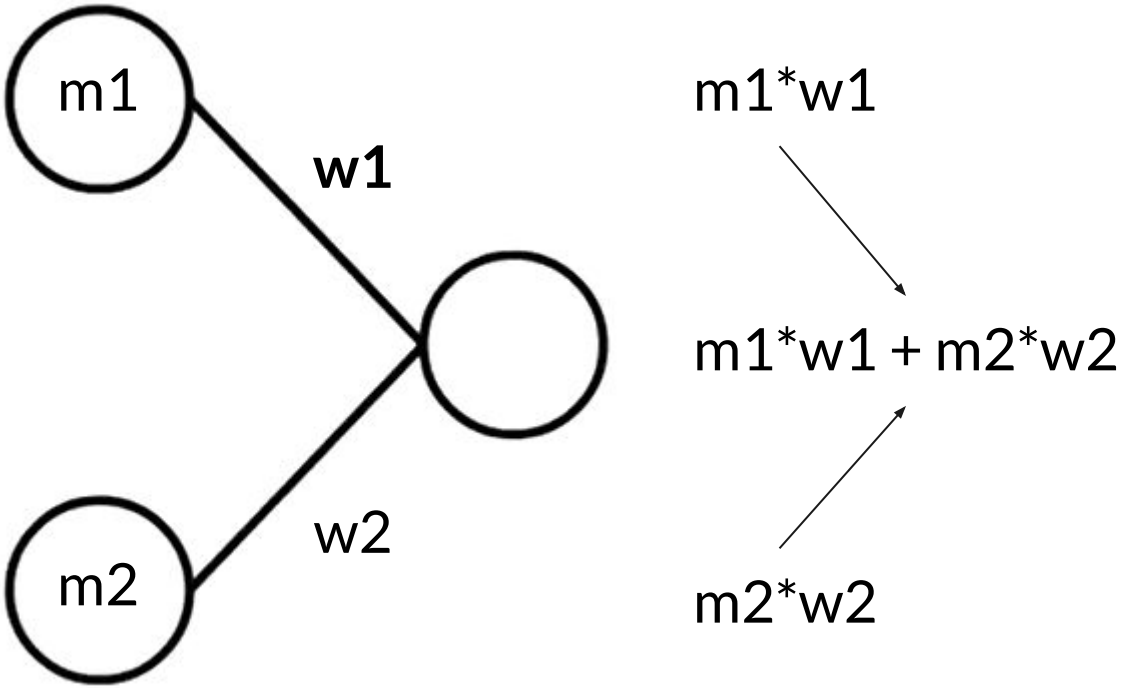
la prima parte della funzione è la **combinazione lineare** degli ingressi



m1*w1

Funzione RN

la prima parte della funzione è la **combinazione lineare** degli ingressi

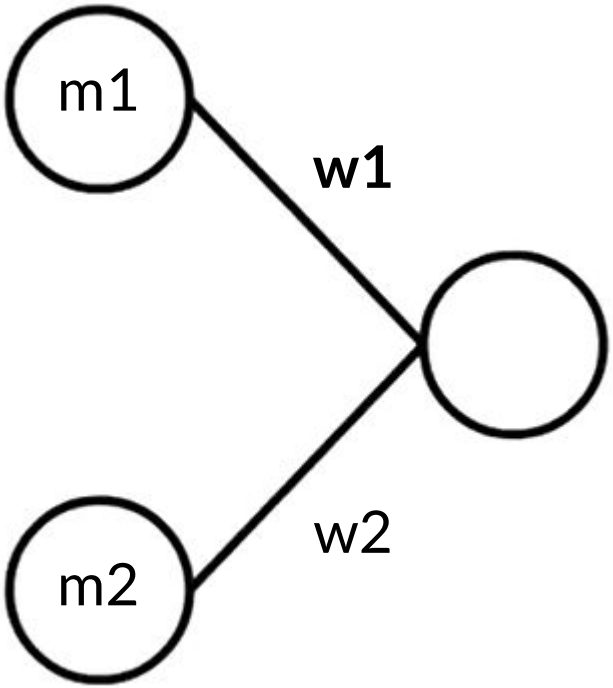


SHARE
MAKERSPACE CUSANO

Funzione RN

la prima parte della funzione è la **combinazione lineare** degli ingressi

aggiungo un altro parametro, il **bias**



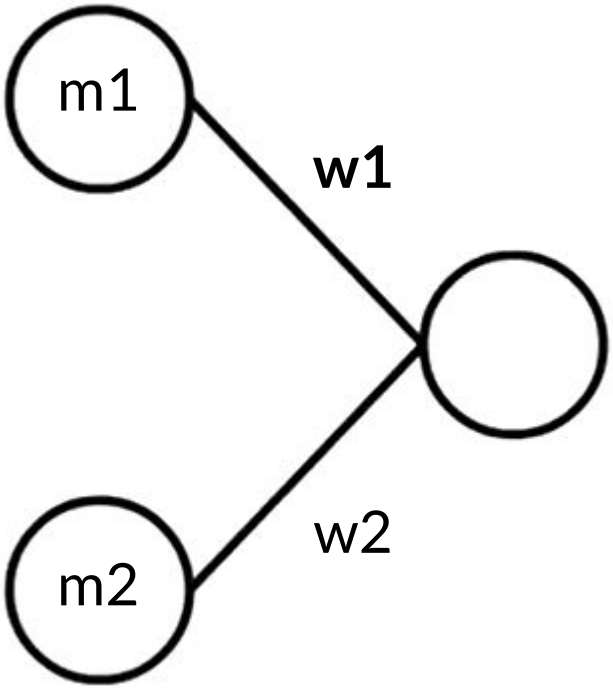
$$m1*w1 + m2*w2 + bias$$

Funzione RN

la prima parte della funzione è la **combinazione lineare** degli ingressi

aggiungo un altro parametro, il **bias**

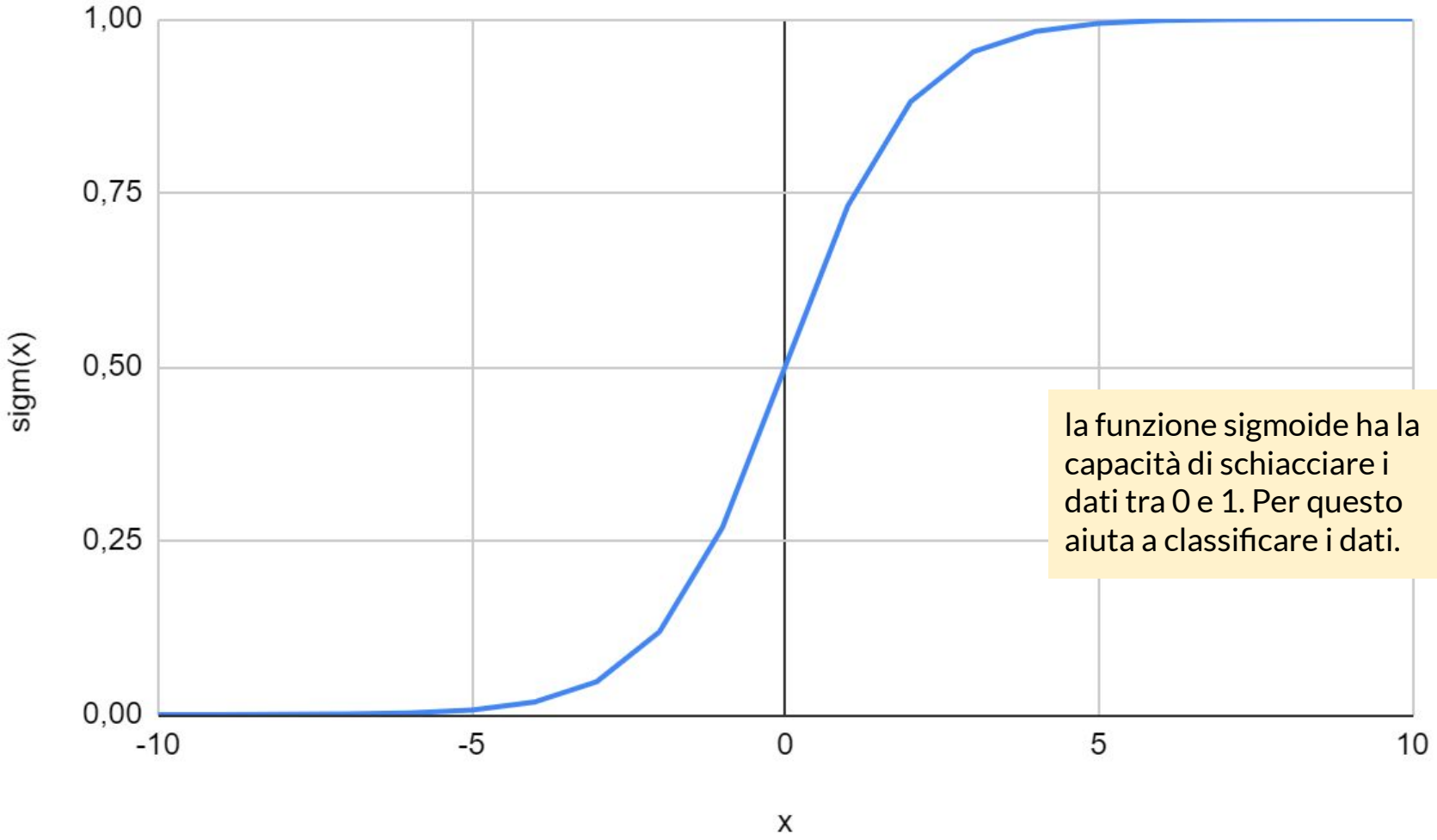
e applico la **sigmoide**



$$\text{Sigm}(\; m1*w1 + m2*w2 \; + \; b \;)$$

Funzione sigmoide

$$\text{Sigmoide}(s) = \frac{1}{1+e^{-s}}$$

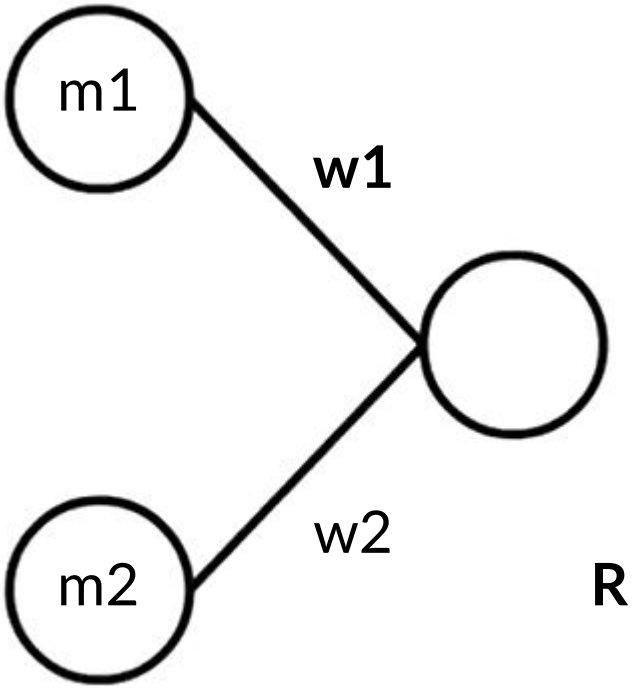


la funzione sigmoide ha la capacità di schiacciare i dati tra 0 e 1. Per questo aiuta a classificare i dati.

SHARE MAKERSPACE CUSANO

Torniamo all'esempio...

RN (m1,m2)



⚖	9	2	3,2	9,1	1,6
⟷	70	50	49,4	74,6	48,3

w1=? w2=? b=?

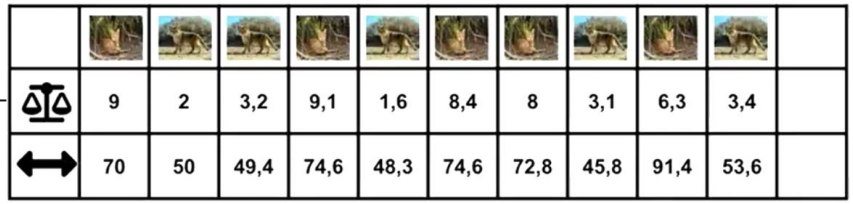
RN (9 , 70) = Sigm(9*w1 + 70*w2 + b)

inizialmente non si conoscono le incognite e si mettono valori a piacere

SHARE MAKERSPACE CUSANO

Introduciamo la funzione di costo

- Poiché inizialmente non si conoscono pesi e bias, li si mette a piacere e si stabilisce una **funzione di costo**
- Con i dati si fanno delle prove e si modificano iterativamente i parametri (w1, w2 e b) per ridurre al minimo la funzione di costo



DATI

FUNZIONE DI COSTO

PREVISIONE

è un'altra funzione il cui risultato voglio rendere minimo modificando i parametri

RN (9 , 70) = Sigm(9*w1 + 70*w2 + b)

SHARE MAKERSPACE CUSANO

Funzione di costo

p = previsione = RN (m1,m2) = Sigm(m1*w1 +m2*w2 + b)

o = obiettivo = 1 | 0 caso per caso dai dati

costo = $(p - o)^2$ **Errore quadratico**

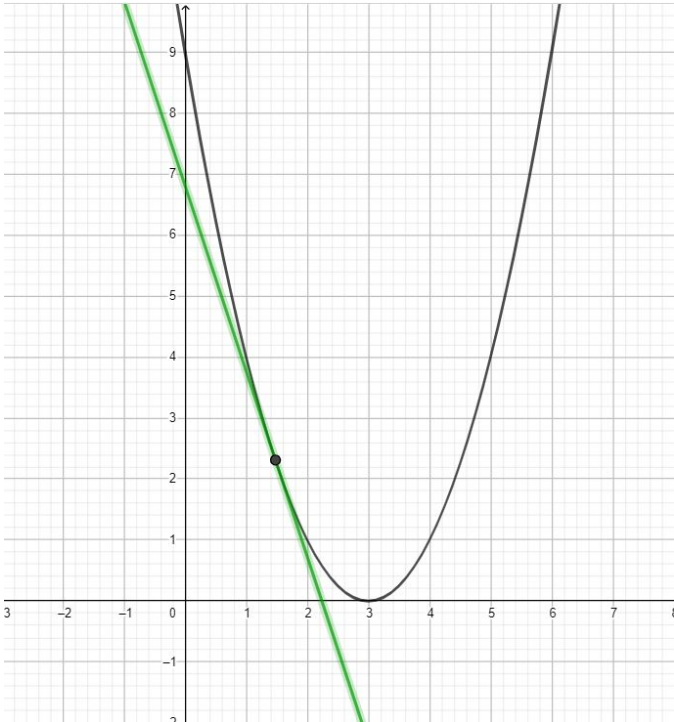
Foglio excel calcolatore

Ridurre la funzione di costo

Per ridurre la funzione di costo il programma procede per approssimazioni successive, calcolando la derivata della funzione di costo.

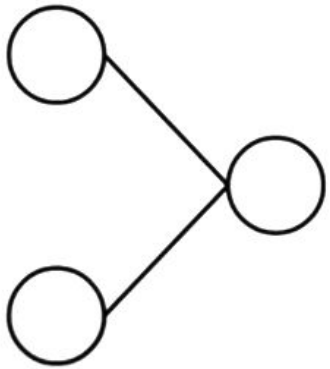
Considerando una parabola (**(p-o)^2 è una parabola**), la derivata indica la pendenza della retta tangente alla parabola. Quando si riduce la pendenza si arriva al minimo della parabola.

I pesi ad ogni iterazione vengono quindi modificati di una frazione della derivata, in questo modo si arriva gradualmente, iterazione dopo iterazione, al minimo (dove c'è il vertice della parabola).



SHARE
MAKERSPACE CUSANO

Ricapitolando



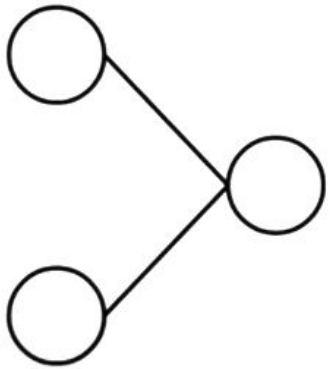
Forward propagation

$$p = RN(9, 70) = Sigm(2*w1 + 50*w2 + b)$$

Funzione di costo

$$costo(w1,w2,b) = (p - o)^2 = (1 - Sigm(2*w1 + 50*w2 + b)$$

Ricapitolando



Back Propagation

modifico parametri

$$w1 = w1_{precedente} - frazione * derivata_{w1}$$

$$w2 = w2_{precedente} - frazione * derivata_{w2}$$

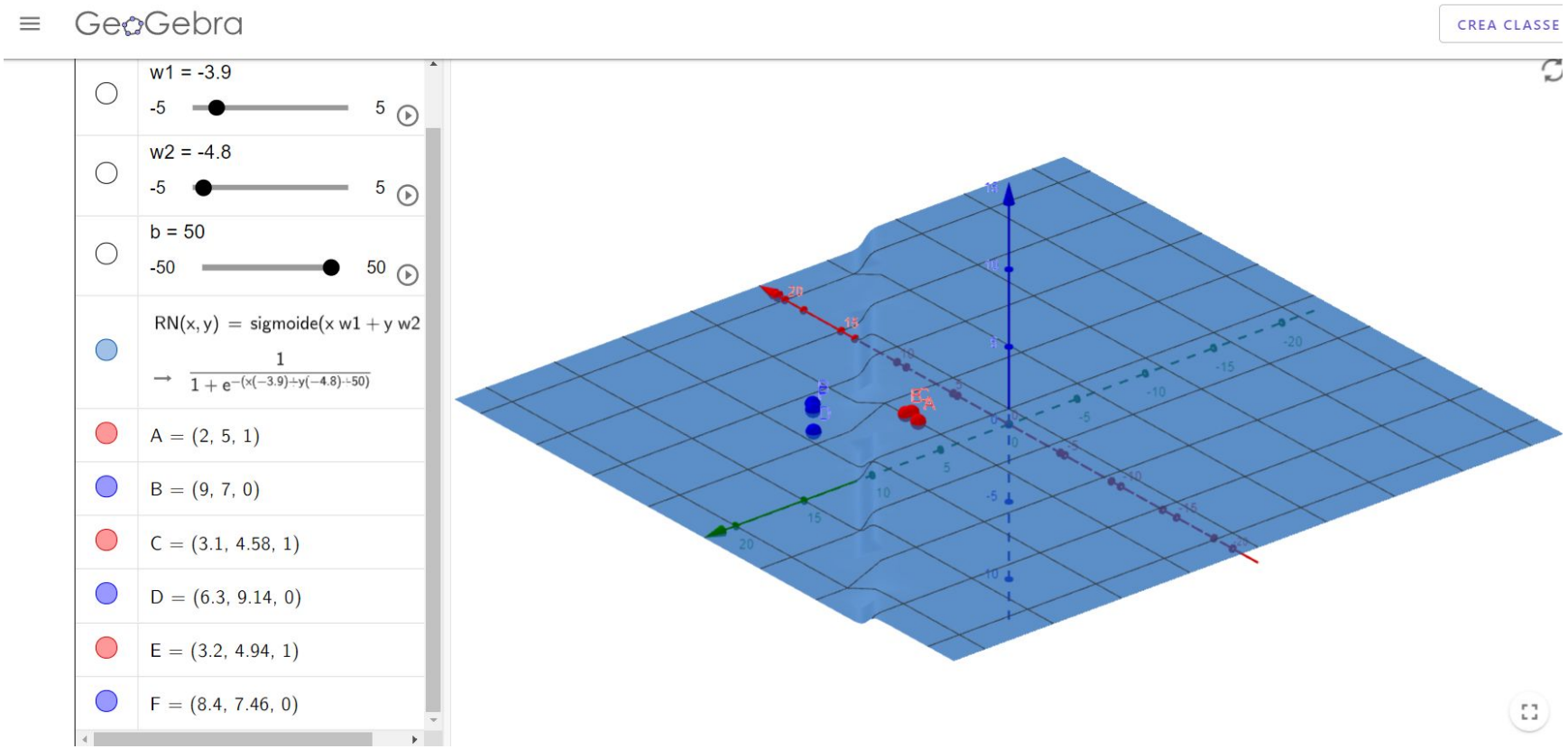
$$b = b_{precedente} - frazione * derivata_{b}$$

ripeto

Visualizzazione 3D

Grafico

Foglio excel calcolatore

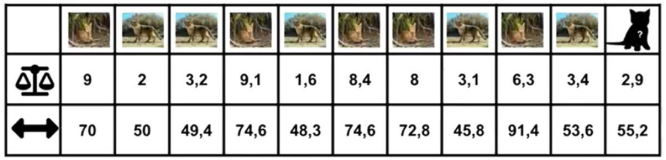


Con Tensorflow.js

SHARE MAKERSPACE CUSANO

Tipologie di machine learning

- Apprendimento supervisionato (classificatore, regressore)



⚖	9	2	3,2	9,1	1,6	8,4	8	3,1	6,3	3,4	2,9
⟷	70	50	49,4	74,6	48,3	74,6	72,8	45,8	91,4	53,6	55,2

- Apprendimento non supervisionato (individua strutture nei dati)

- Apprendimento per rinforzo (agenti autonomi con obiettivi)

Fine

References:

- Test di Turing
- Applicazioni dell'IA
- Introduzione alle reti neurali di Riccardo Talarico 1 e 2
- Corso Elements of AI
- AI plays Snake
- Matematico stupito