

Machine Learning: Algoritmi e Modelli

Reti Neurali Artificiali

Il Neurone Artificiale

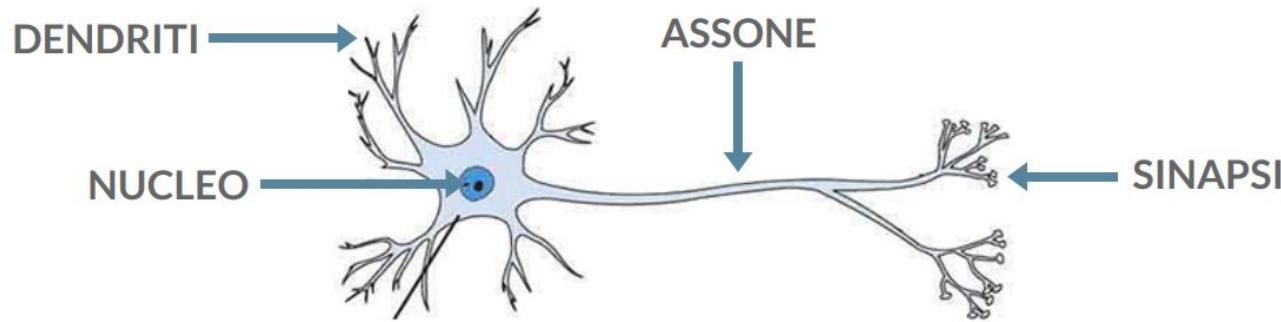
presentato da
Giuseppe Gullo

Cosa è una Rete Neurale Artificiale?

Una Rete Neurale Artificiale è un modello di machine learning
che ~~replica~~ prende ispirazione dal funzionamento dei **neuroni** all'interno del cervello

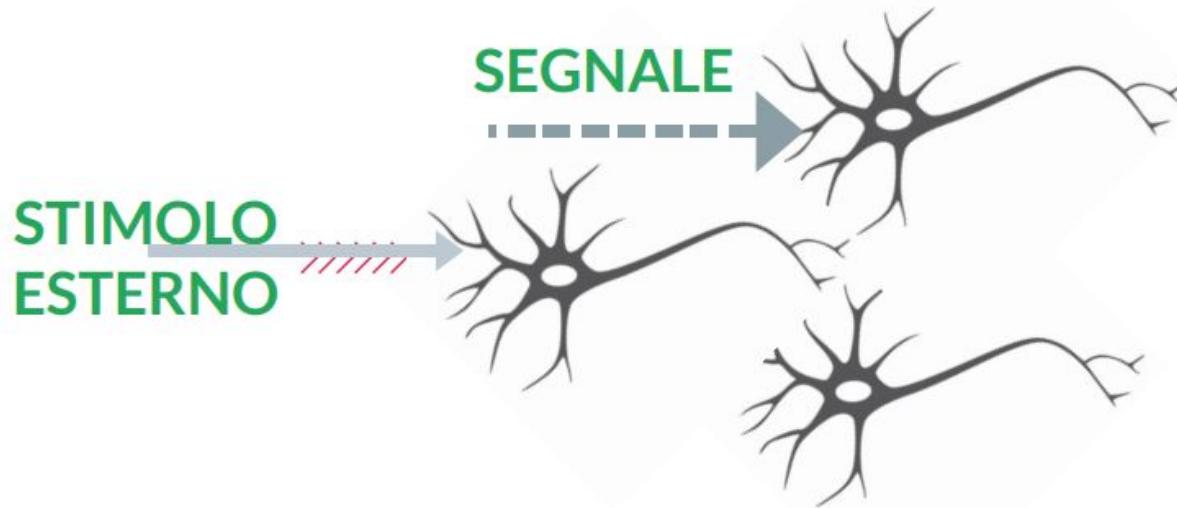
Cosa è un neurone?

Il neurone è il transistor del cervello

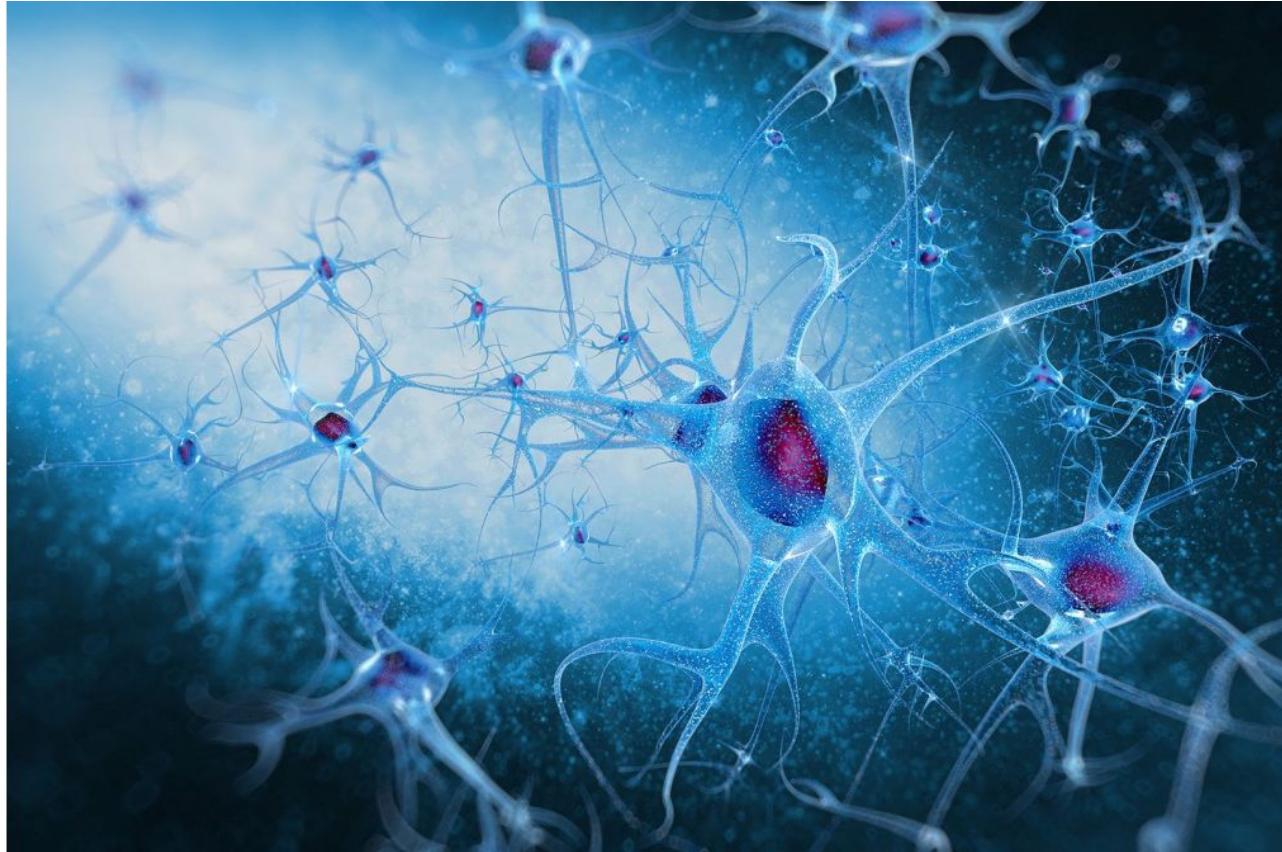


Come viaggia il segnale?

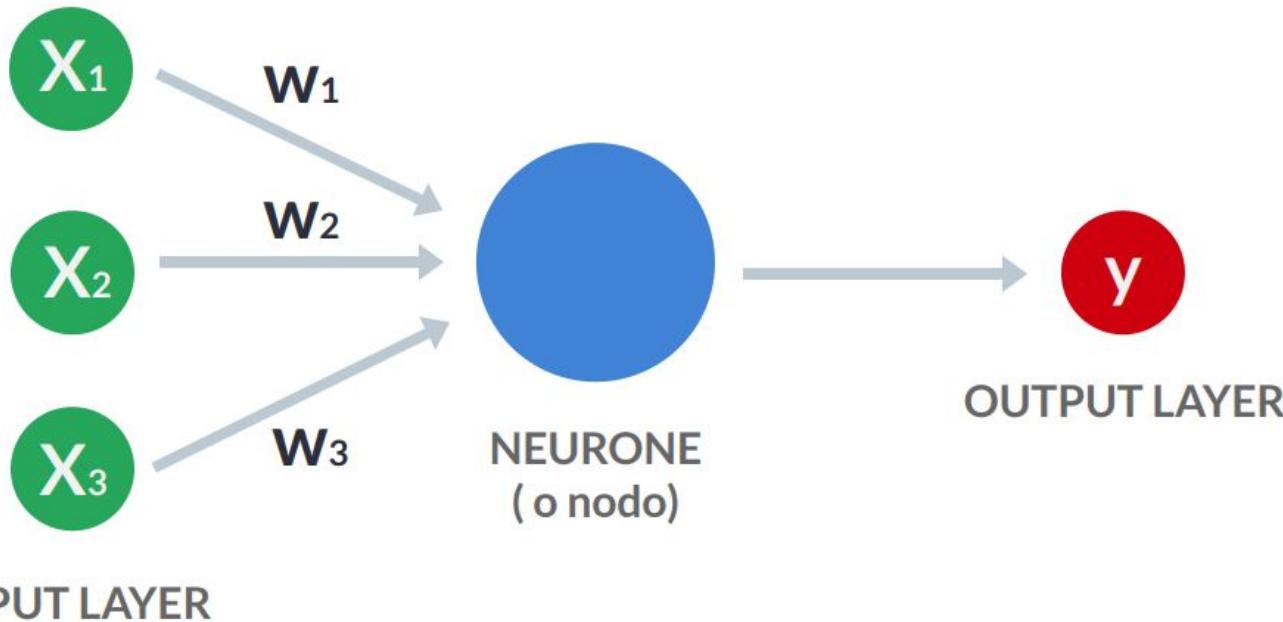
Il segnale viaggia lungo l'assone fino ai dendriti e si propaga a cascata da neurone a neurone tramite le sinapsi



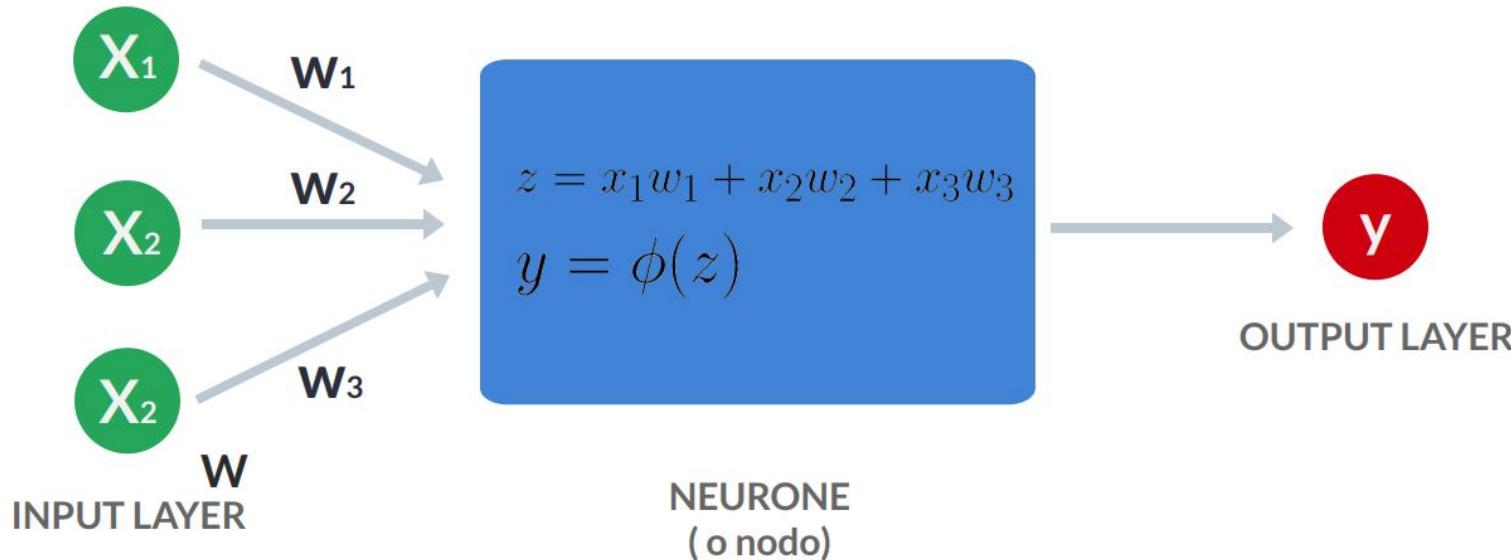
Neuroni che si attivano insieme si legano insieme in complessi reticolari, **le reti neurali**



Un Neurone Artificiale



Il Perceptron (Percettrone)



Machine Learning: Algoritmi e Modelli

Reti Neurali Artificiali

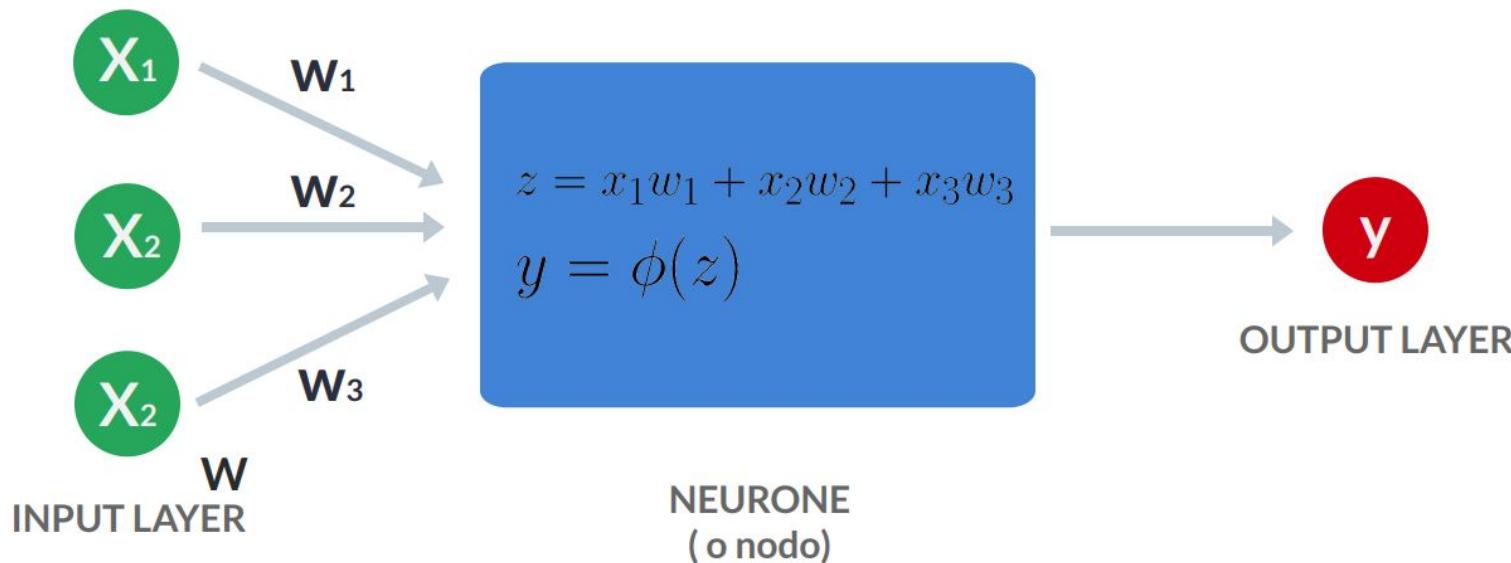
Le funzioni di attivazione

presentato da
Giuseppe Gullo

La funzione di attivazione nei neuroni artificiali

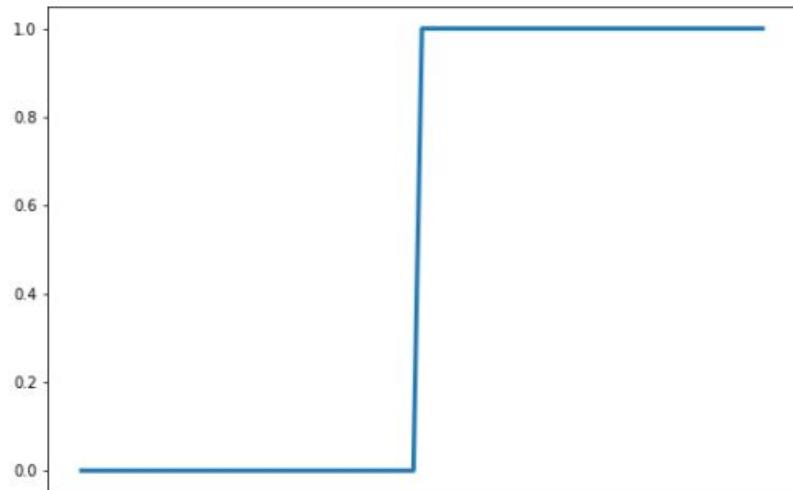
La funzione di attivazione nei neuroni artificiali definisce come la somma pesata degli input viene trasformata in output

La funzione di attivazione nel percettrone



Funzioni di attivazione

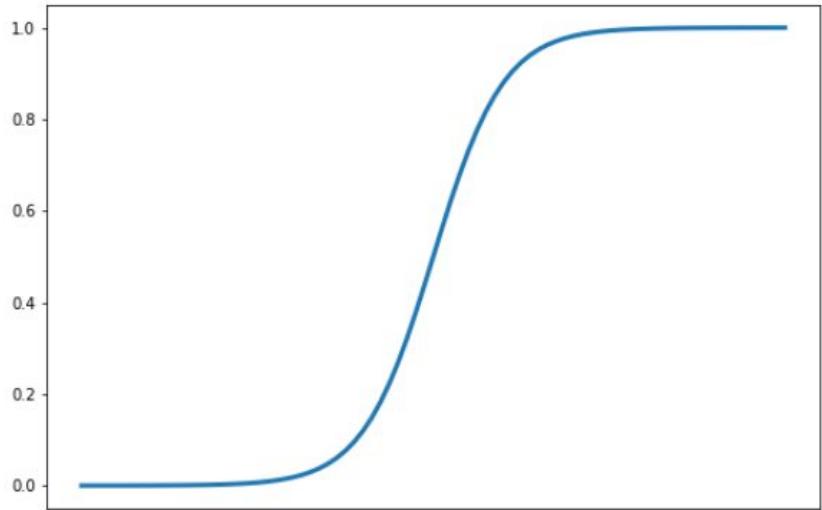
Step Function



$$\phi(z) = \begin{cases} 1 & \text{se } z \geq 0 \\ 0 & \text{se } z < 0 \end{cases}$$

Funzioni di attivazione

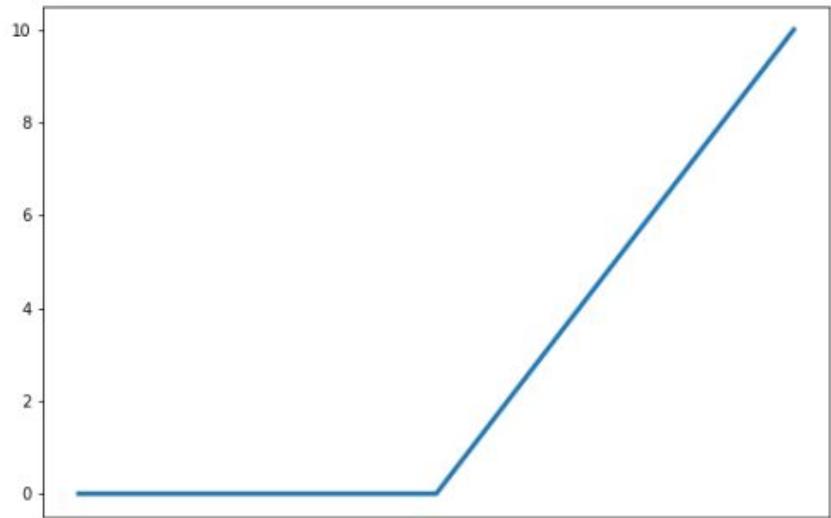
Sigmoide



$$\phi(z) = \frac{1}{1 + e^{-z}}$$

Funzioni di attivazione

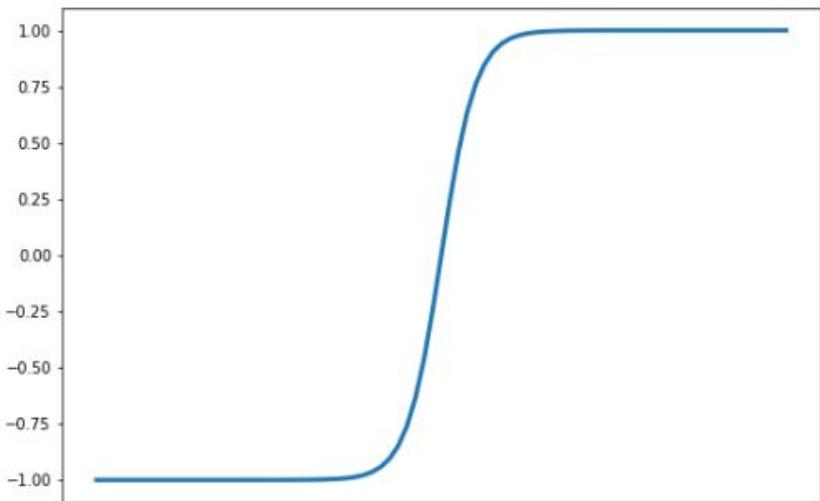
Rectified Linear Unit (ReLU)



$$\phi(z) = \max(0, z)$$

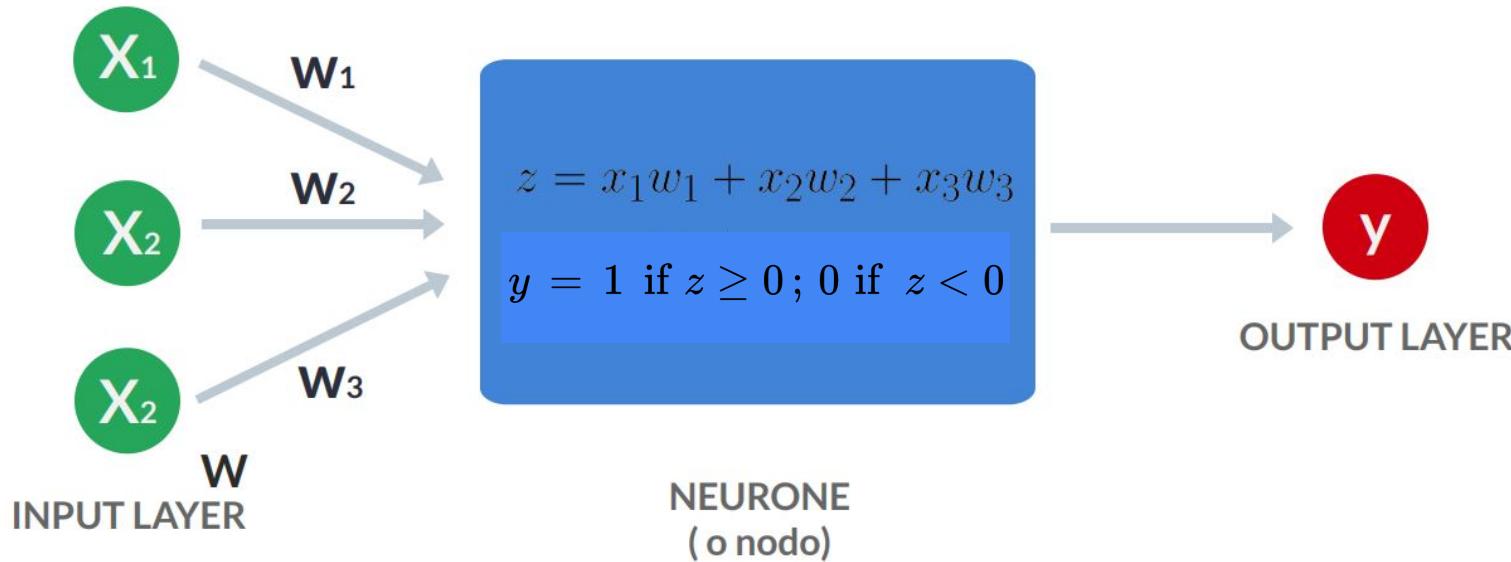
Funzioni di attivazione

Tangente Iperbolica (Tanh)

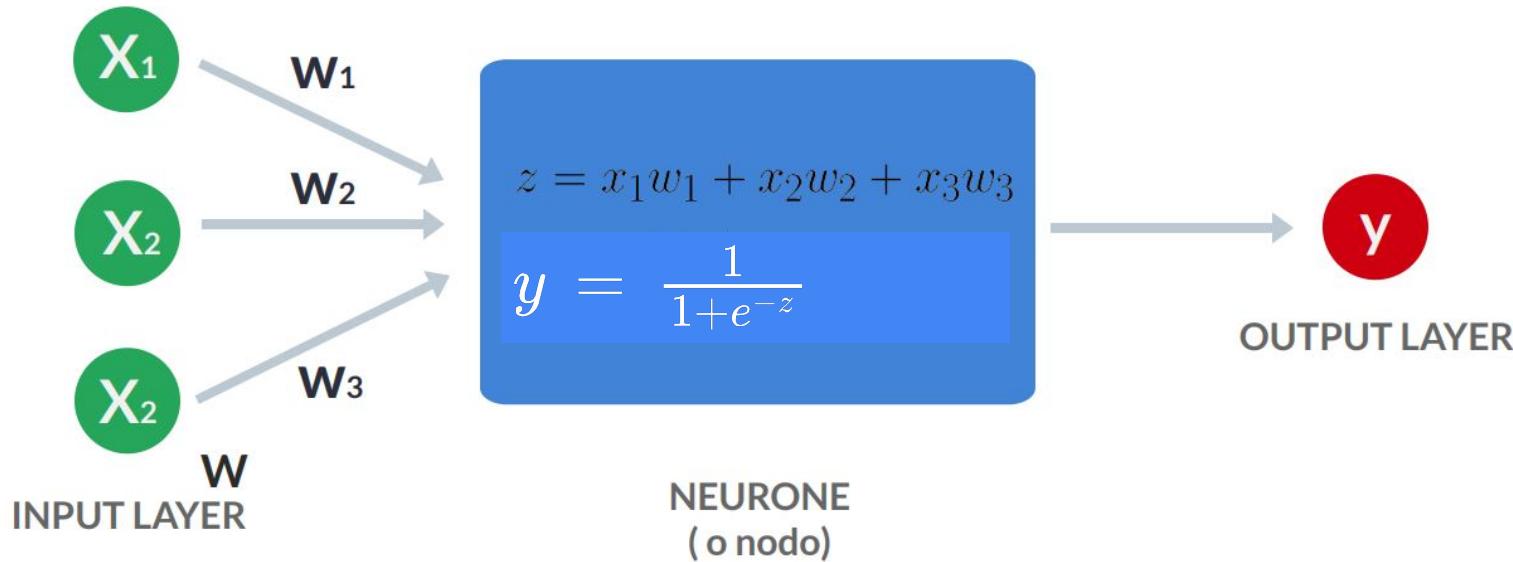


$$\phi(z) = \frac{1 - e^{-2z}}{1 + e^{-2z}}$$

Tipicamente il percettrone utilizza la **Step Function**



Un perceptron che usa la sigmoide come funzione di attivazione
ha lo stesso funzionamento di una **regressione logistica**



Machine Learning: Algoritmi e Modelli

Reti Neurali Artificiali

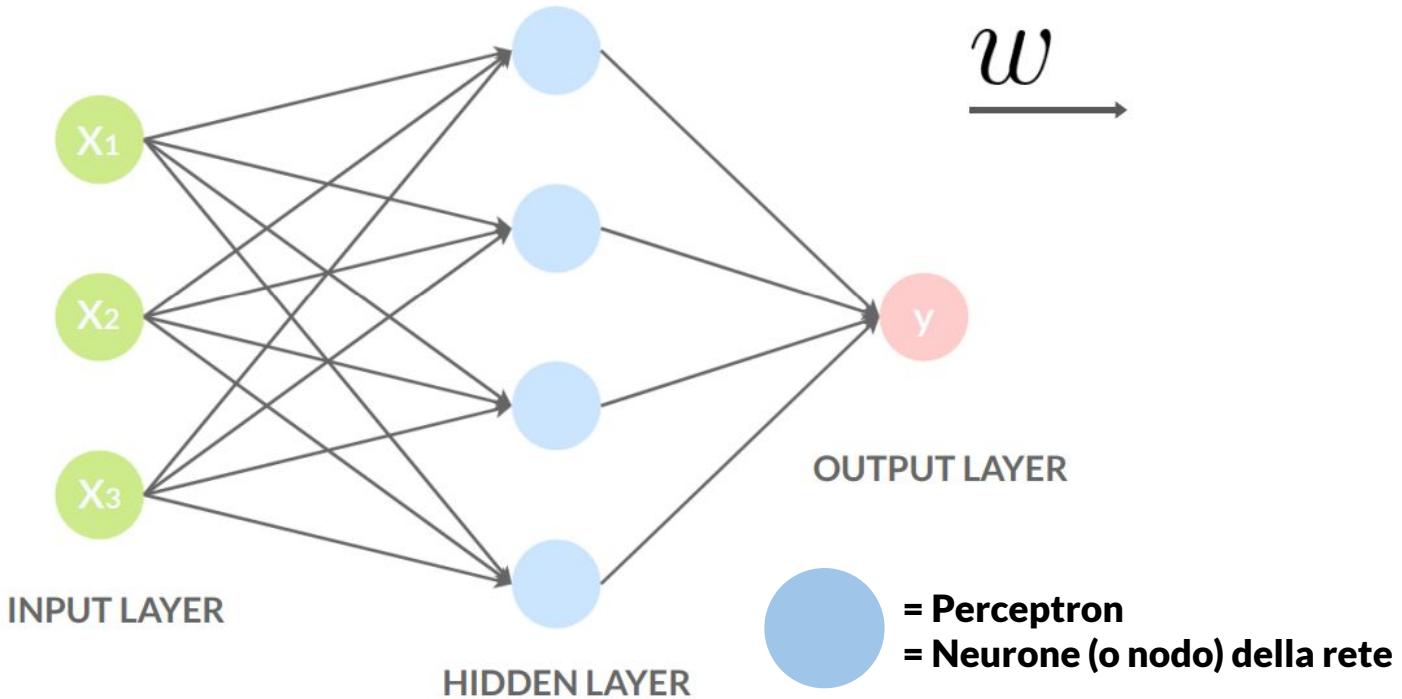
Il Multilayer Perceptron

presentato da
Giuseppe Gullo

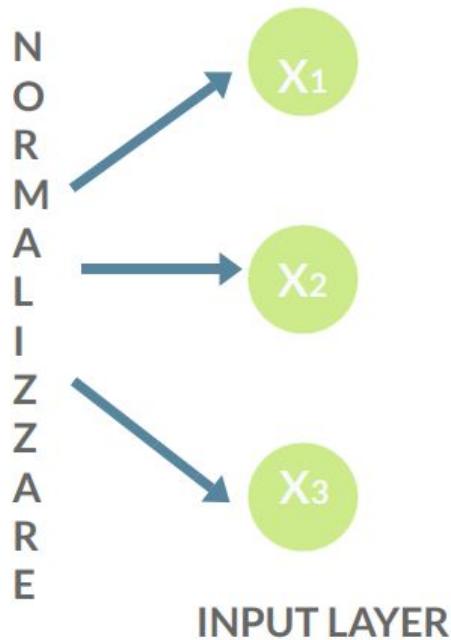
Il Multilayer Perceptron (Percettrone Multistrato)

Chiamato anche **Vanilla Neural Network**,
il Multilayer Perceptron è l'esempio più comune di rete neurale artificiale

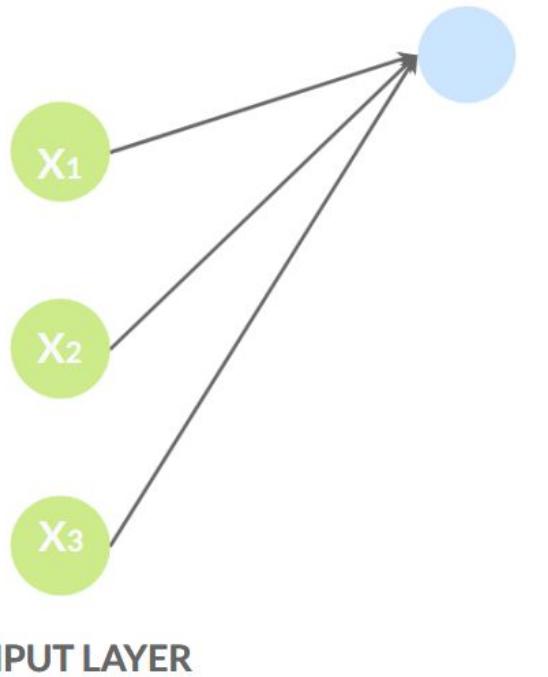
Il Multilayer Perceptron (Percettrone Multistrato)



II Multilayer Perceptron (Percettrone Multistrato)

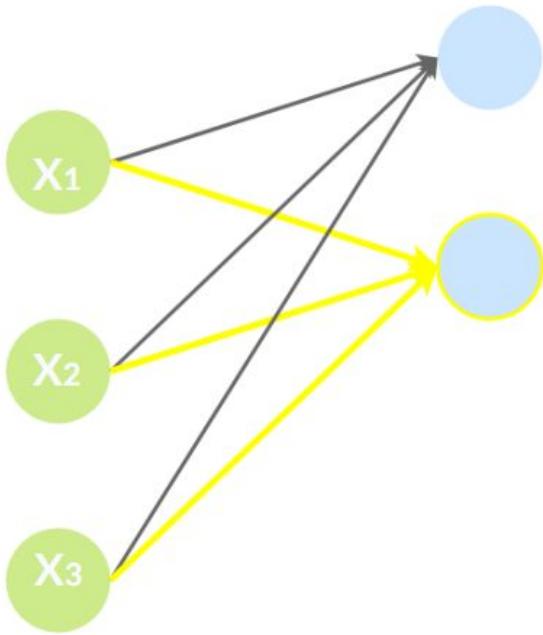


II Multilayer Perceptron (Percettrone Multistrato)



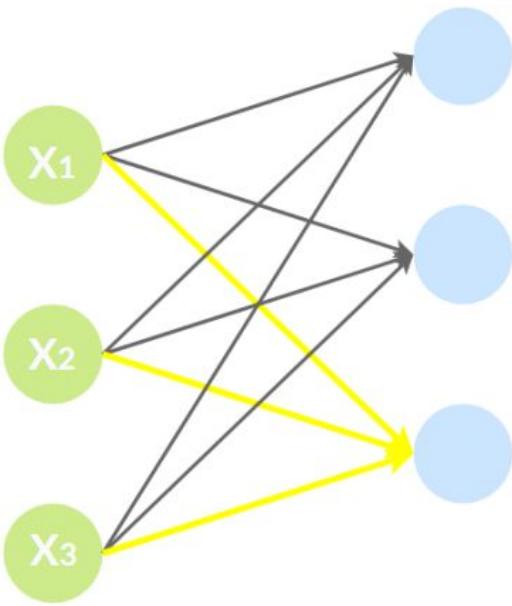
INPUT LAYER

II Multilayer Perceptron (Percettrone Multistrato)



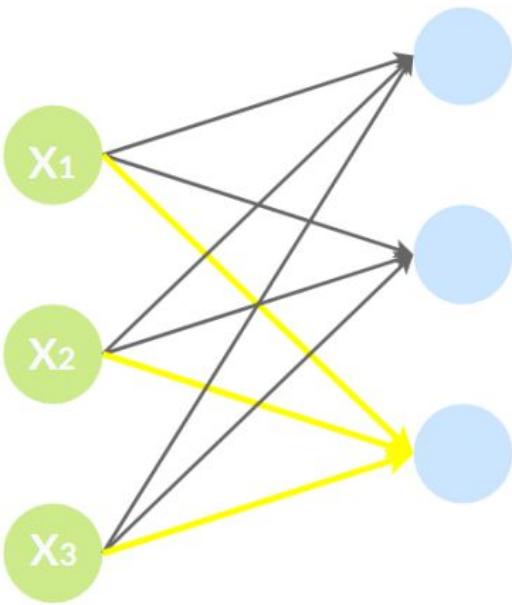
INPUT LAYER

II Multilayer Perceptron (Percettrone Multistrato)



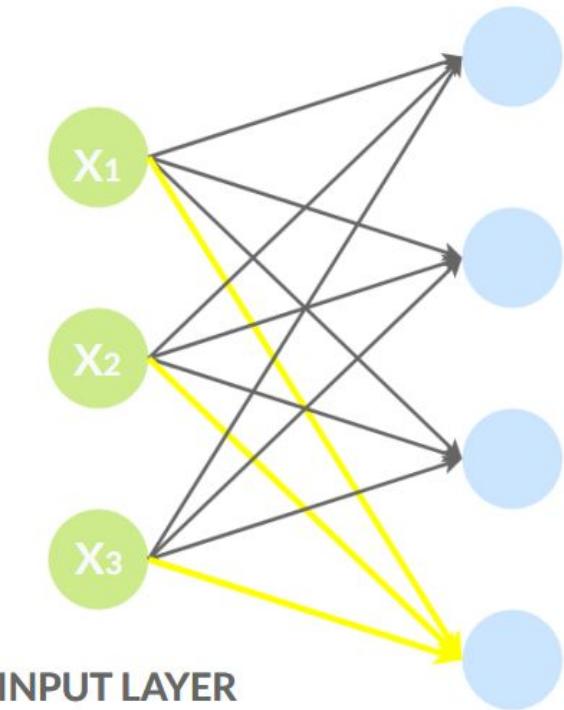
INPUT LAYER

II Multilayer Perceptron (Percettrone Multistrato)

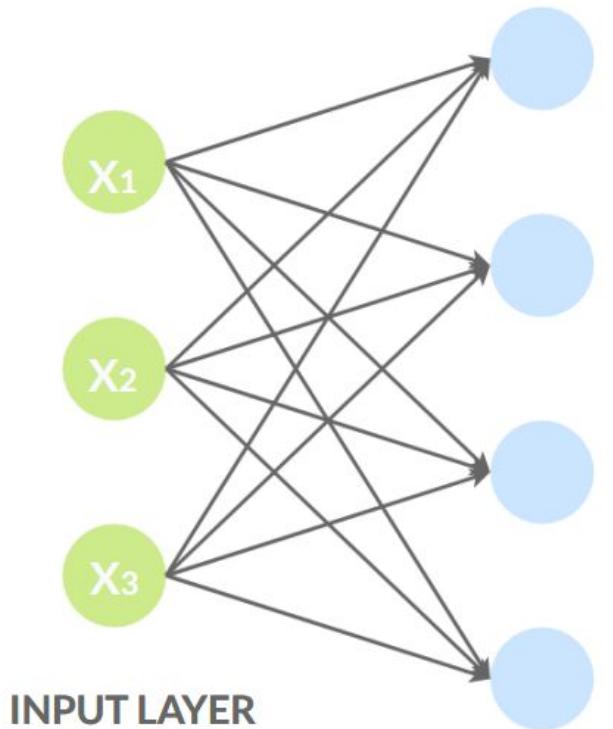


INPUT LAYER

II Multilayer Perceptron (Percettrone Multistrato)



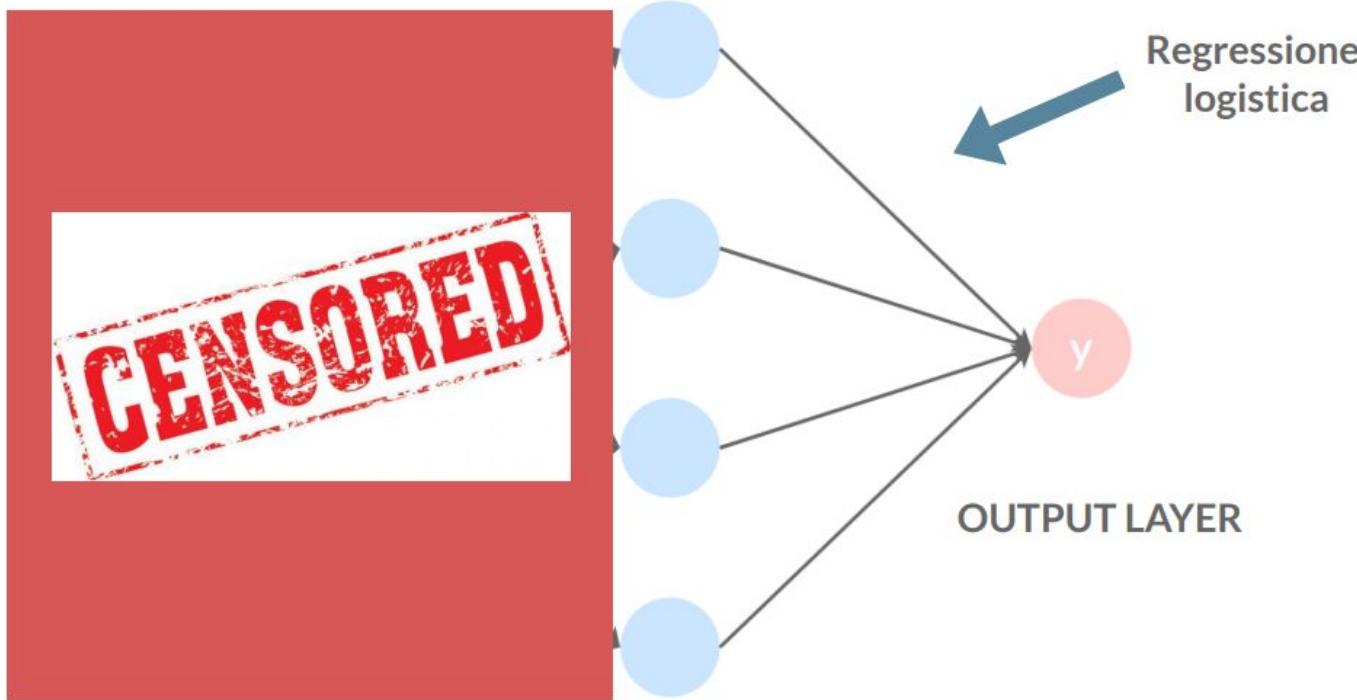
II Multilayer Perceptron (Percettrone Multistrato)



II Multilayer Perceptron (Percettrone Multistrato)

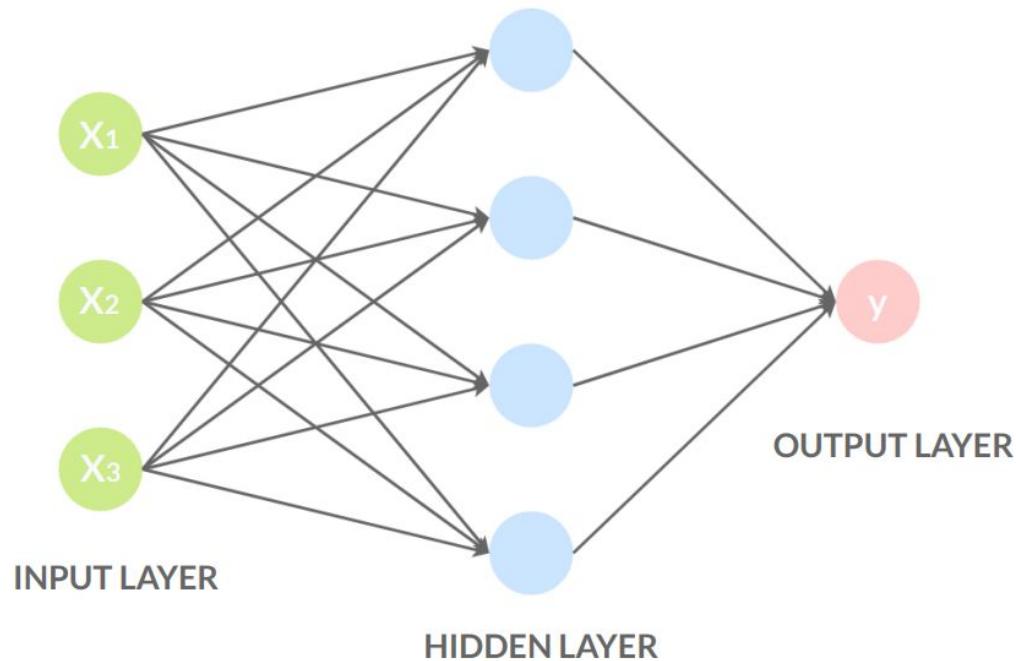


II Multilayer Perceptron (Percettrone Multistrato)



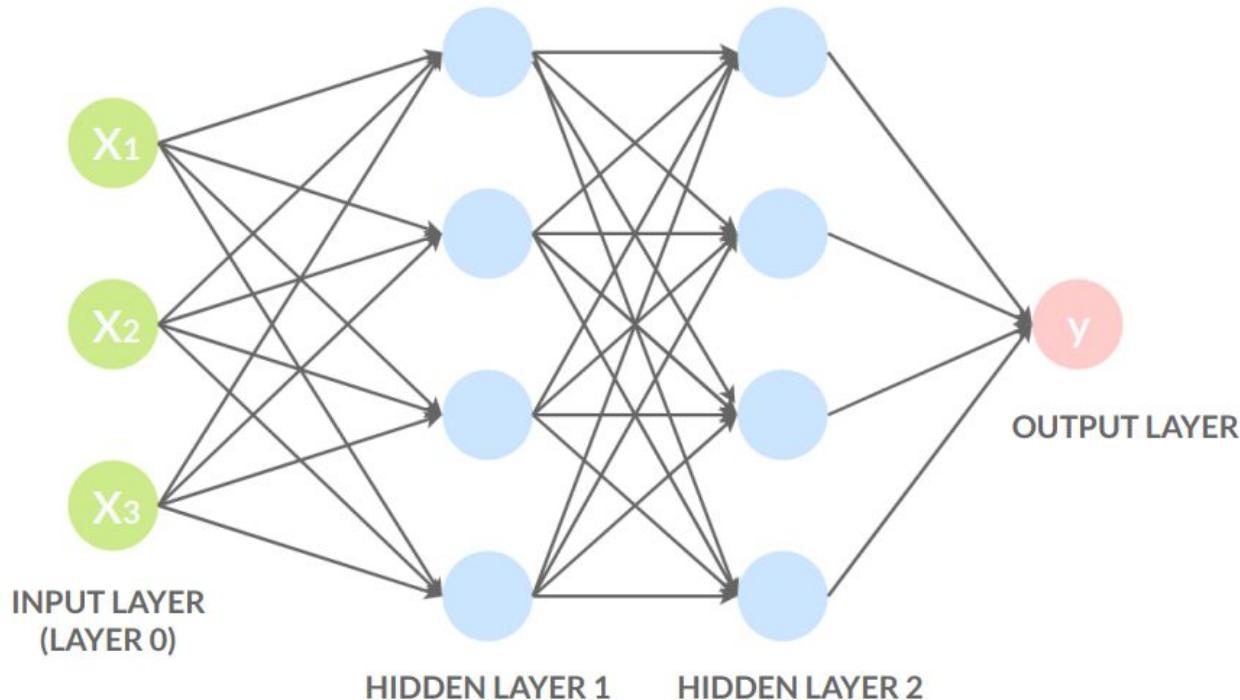
Il Multilayer Perceptron (Percettrone Multistrato)

Automatizza la parte di **Feature Engineering**



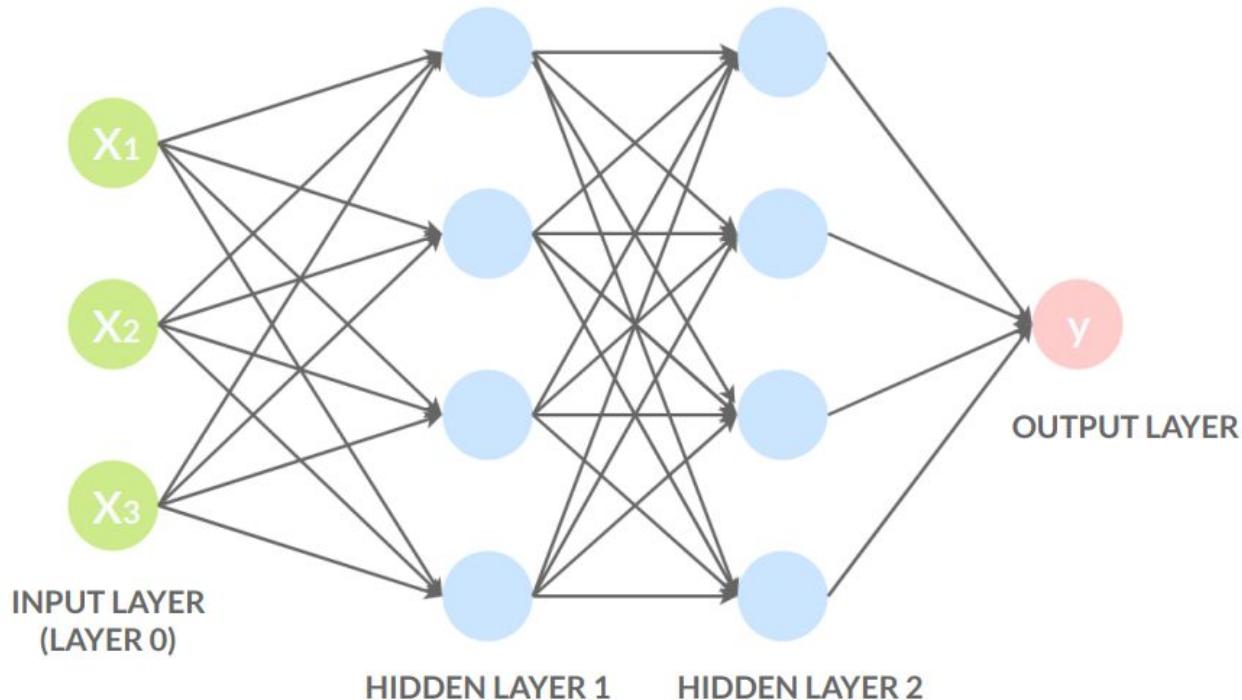
Il Multilayer Perceptron (Percettrone Multistrato)

Numero di strati nascosti, numero di nodi e funzioni di attivazione degli strati nascosti sono iperparametri

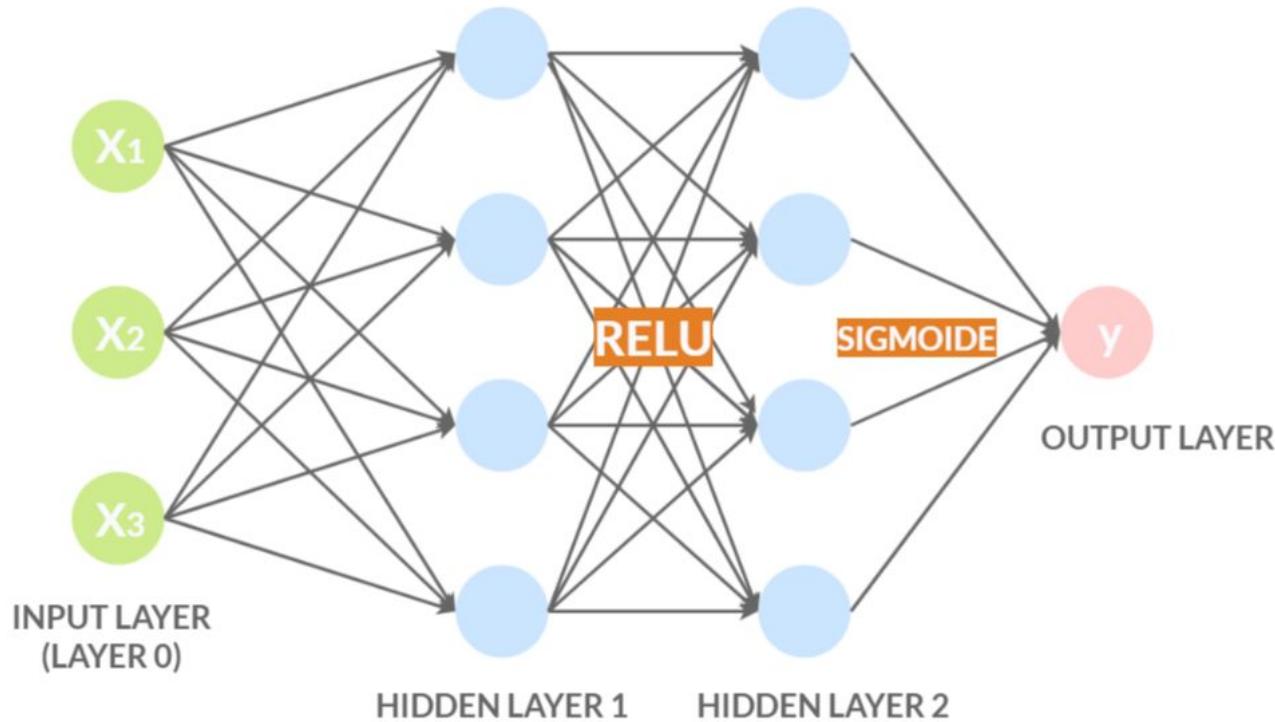


Il Multilayer Perceptron (Percettrone Multistrato)

Aggiungendo più strati nascosti entriamo nei meandri del Deep Learning

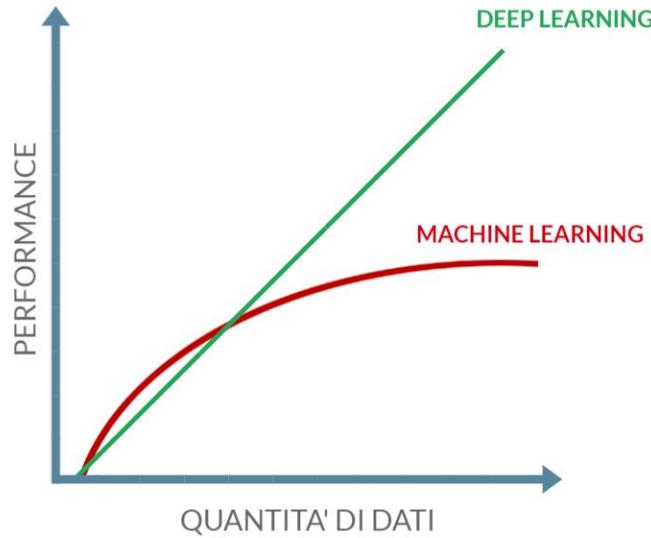


Quale funzione di attivazione utilizzare?



Vantaggi del MLP

- Può apprendere relazioni, anche molto complesse, nei dati
- Può migliorare costantemente all'aumentare dei dati



Svantaggi del MLP

- Richiede molti dati
- Richiede molte risorse di calcolo
- Ci sono molti iperparametri
- E' difficile, se non impossibile, da interpretare

Usa un MLP se:

- I modelli più semplici non portano a risultati soddisfacenti
- Hai molti dati a disposizione
- Hai intenzione di raccogliere nuovi dati nel tempo per migliorare il modello

Non usare un MLP se:

- Basta una regressione logistica
- Hai pochi dati
- Hai poche risorse di calcolo
- Hai bisogno di spiegare perché il modello fornisce un determinato risultato

Machine Learning: Algoritmi e Modelli

Reti Neurali Artificiali

Addestramento di una Deep Neural Network

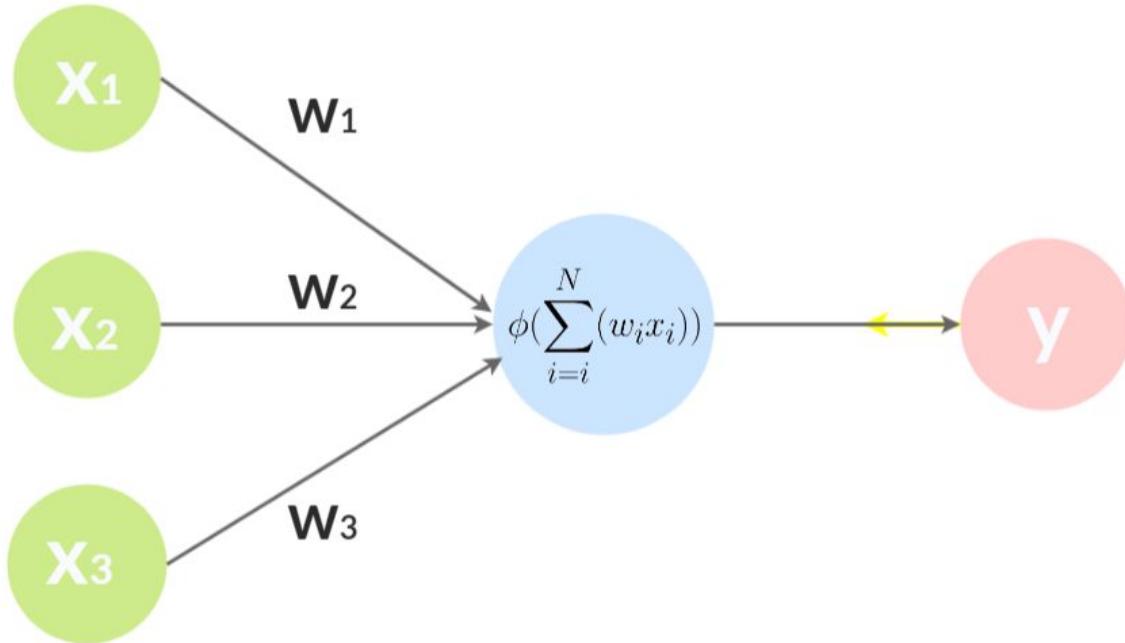
presentato da
Giuseppe Gullo

Addestramento di una Deep Neural Network (Rete Neurale Profonda)

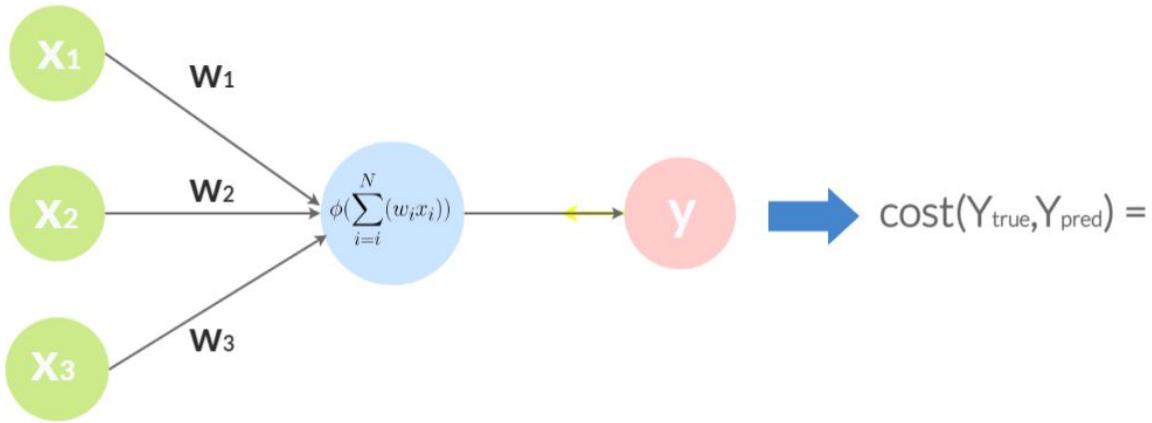
Avviene sempre tramite algoritmi di ottimizzazione

- Gradient Descent
- RProp e RMProp
- Adadelta
- AdaGrad
- AdaMax
- ADAM

Addestramento di un Perceptron

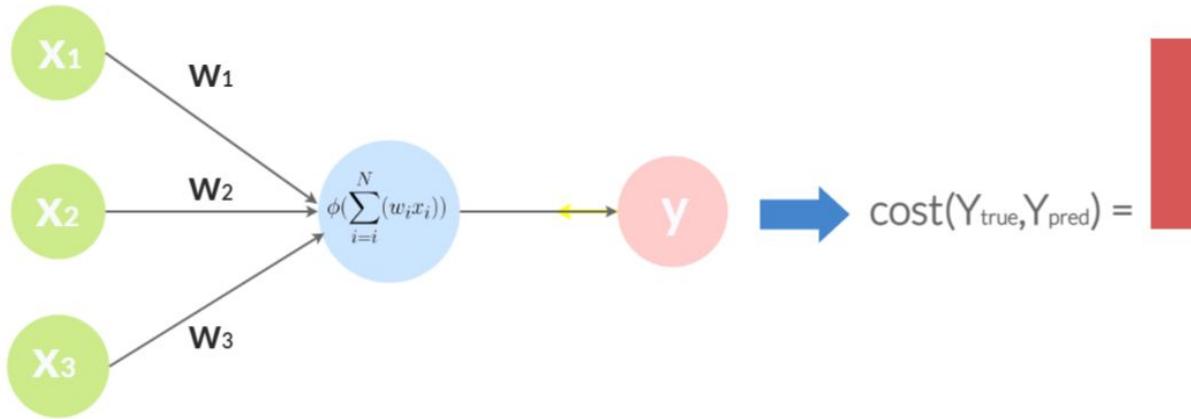


Addestramento di un Perceptron



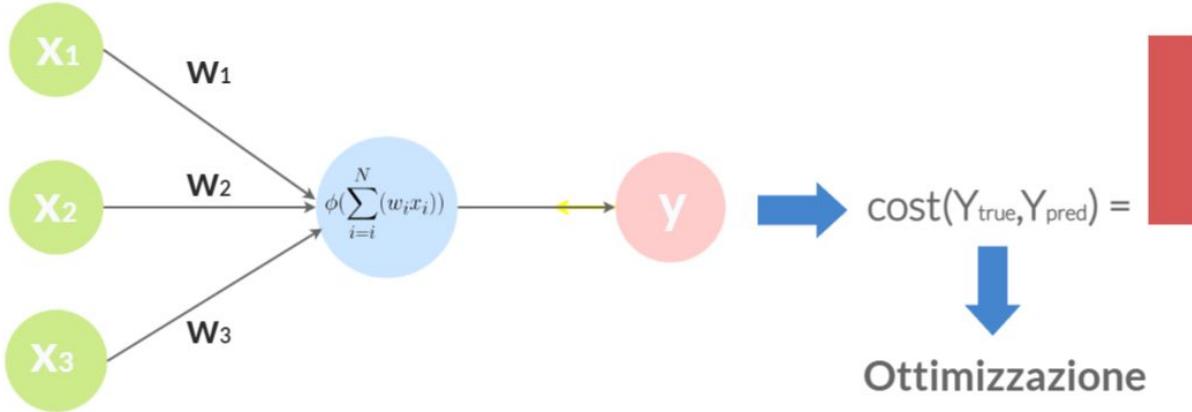
EPOCA: 1

Addestramento di un Perceptron



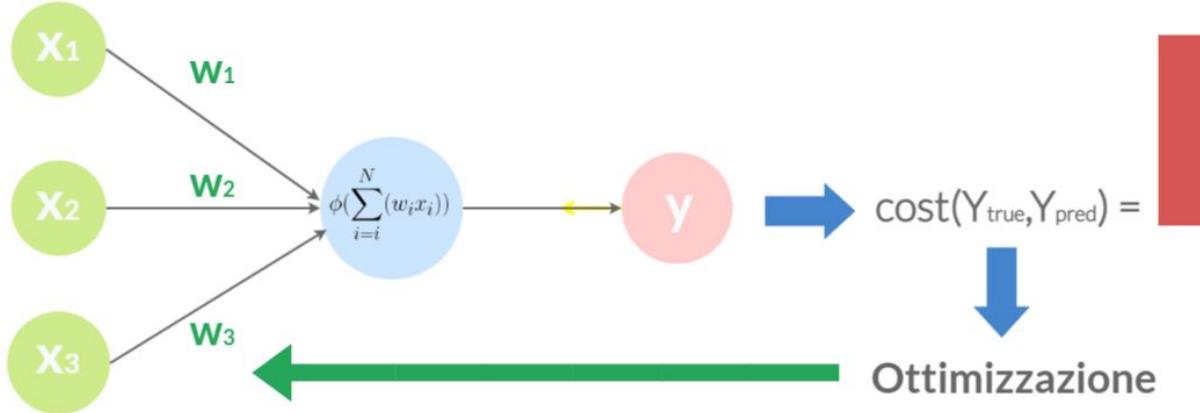
EPOCA: 1

Addestramento di un Perceptron



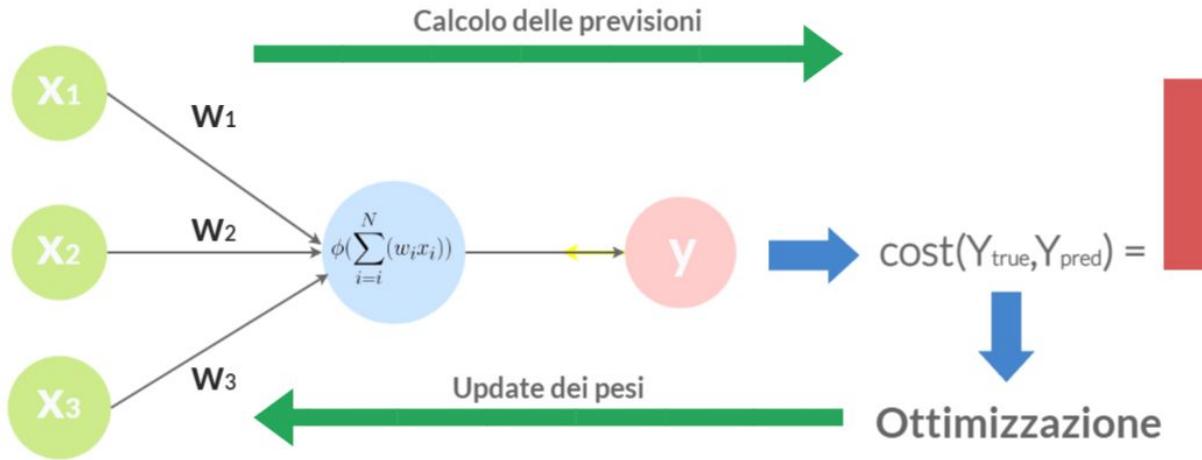
EPOCA: 1

Addestramento di un Perceptron



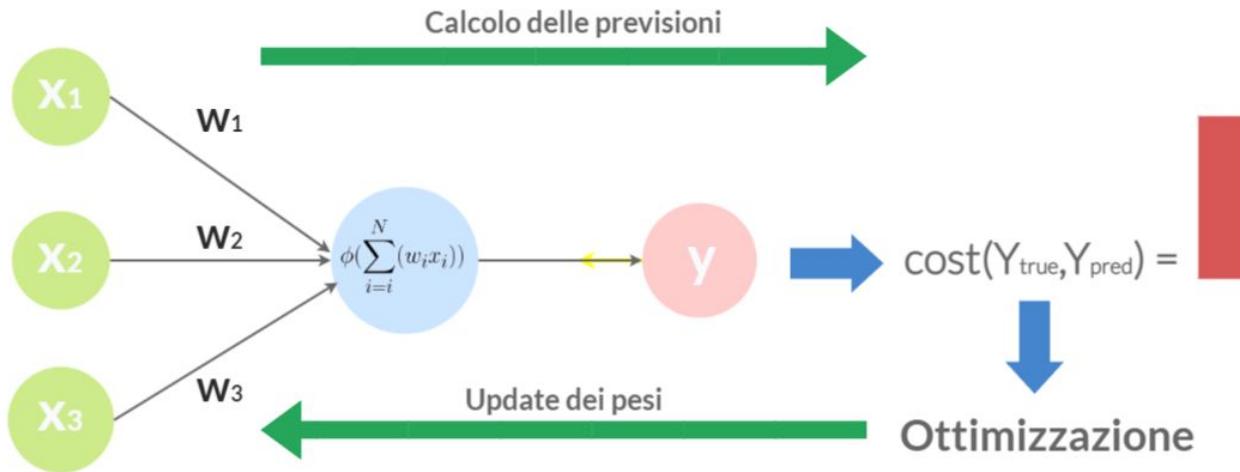
EPOCA: 1

Addestramento di un Perceptron



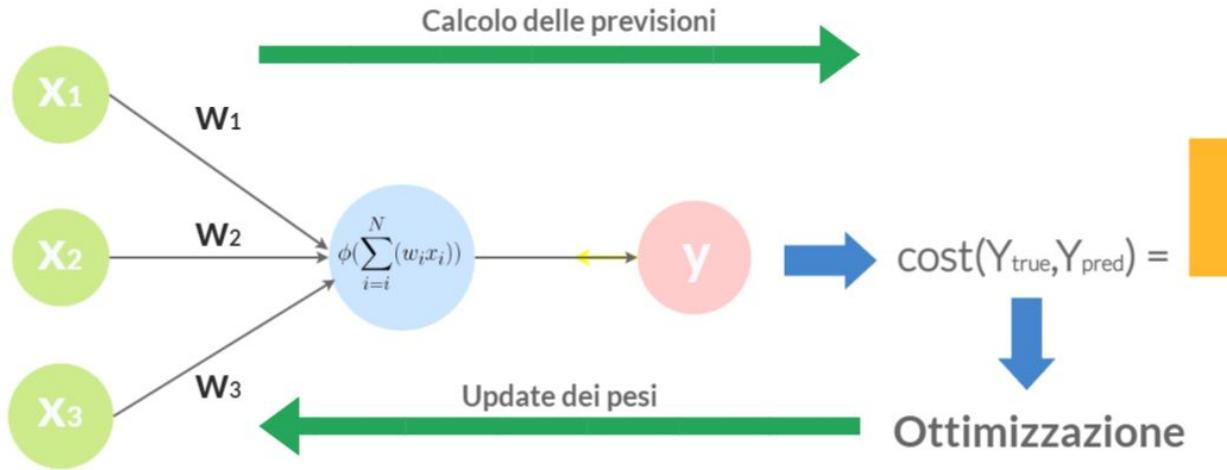
EPOCA: 2

Addestramento di un Perceptron



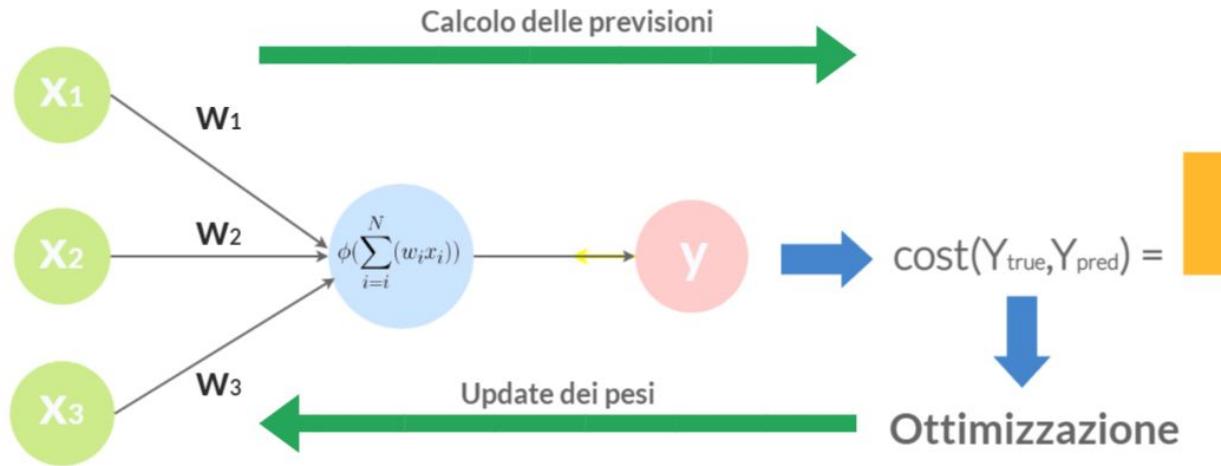
EPOCA: 2

Addestramento di un Perceptron



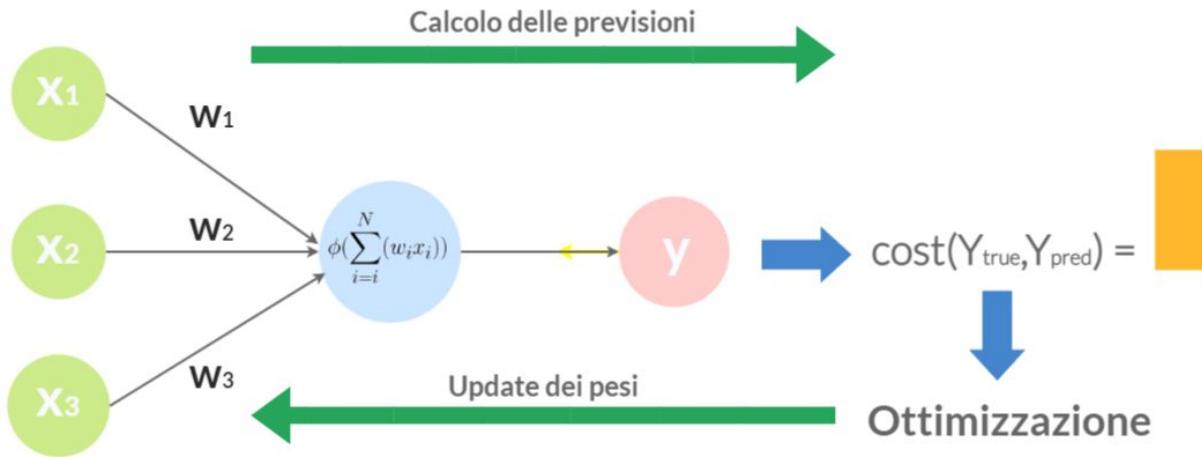
EPOCA: 10

Addestramento di un Perceptron



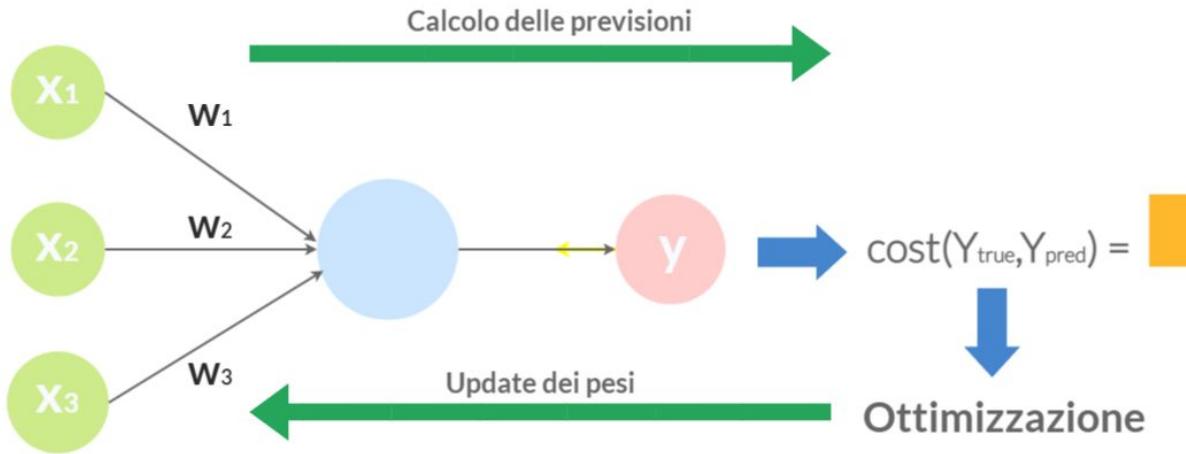
EPOCA: 20

Addestramento di un Perceptron



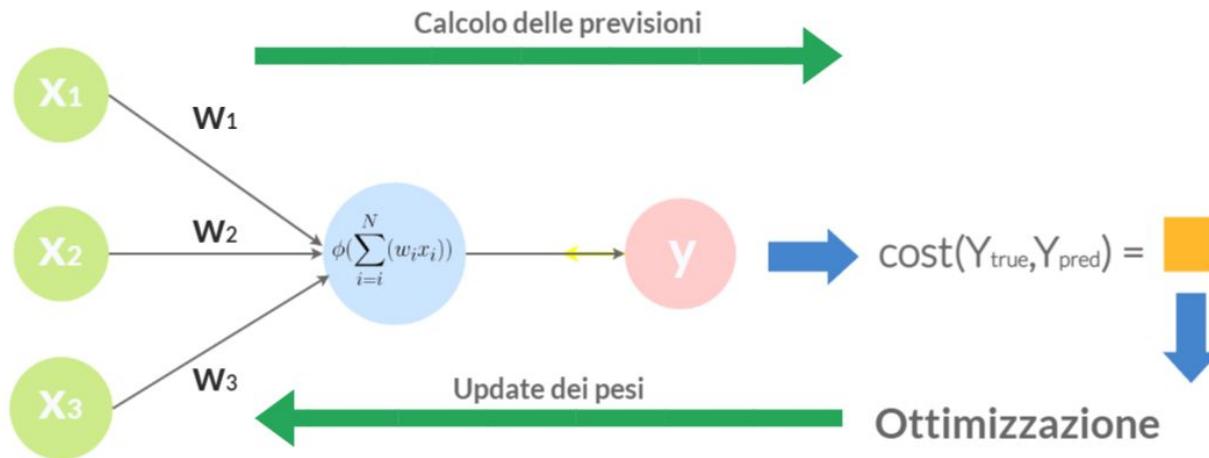
EPOCA: 20

Addestramento di un Perceptron



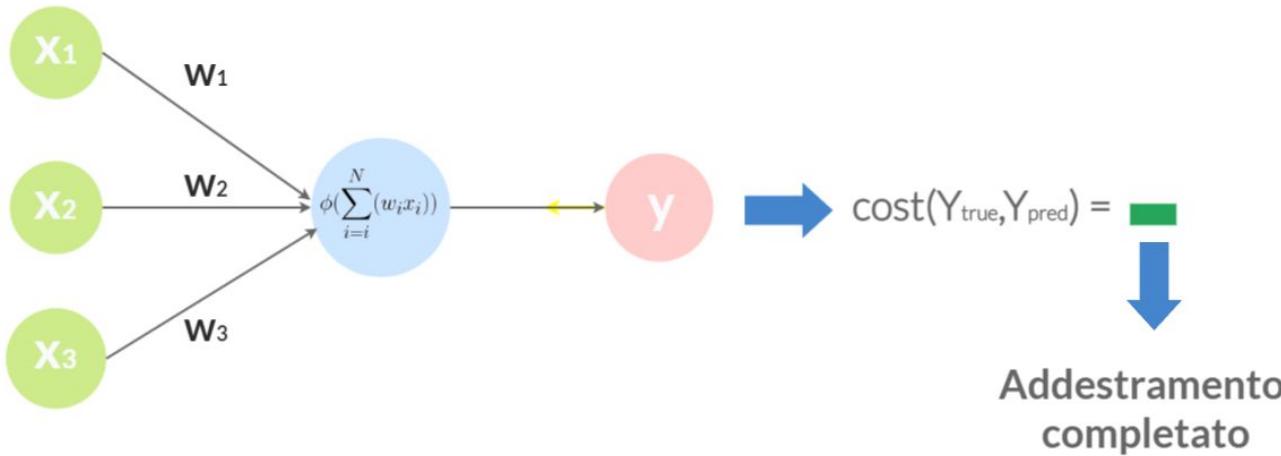
EPOCA: 50

Addestramento di un Perceptron



EPOCA: 80

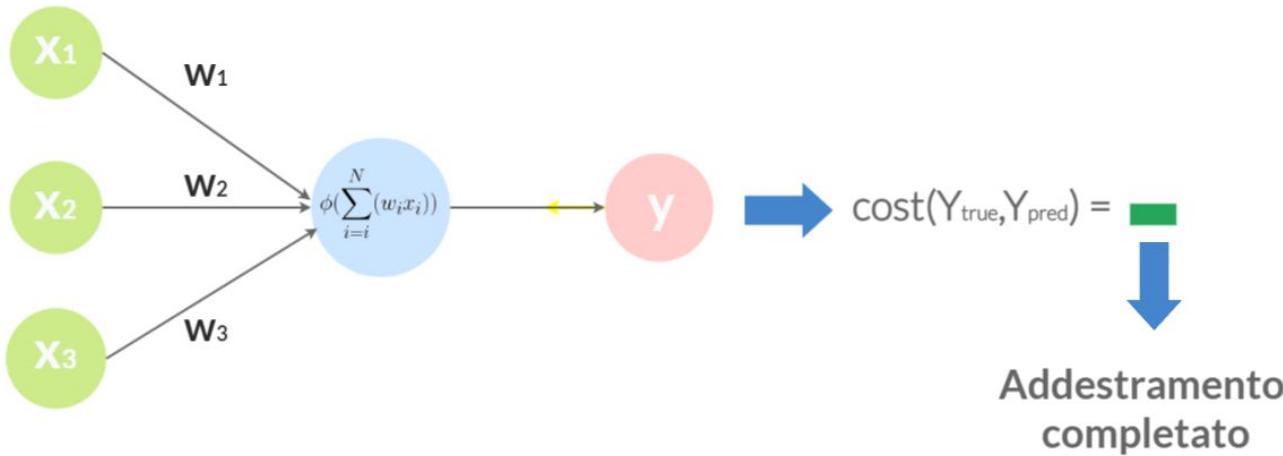
Addestramento di un Perceptron



EPOCA: 100

Addestramento di una Rete Neurale Profonda

Addestramento di un Perceptron

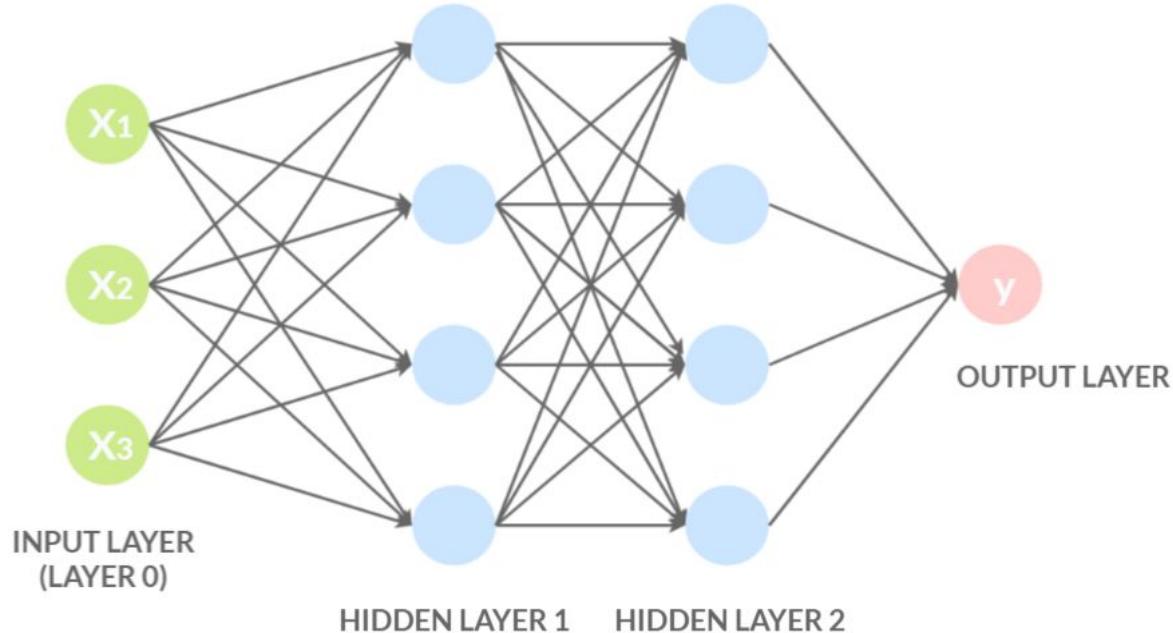


EPOCA: 100

Addestramento di una Deep Neural Network

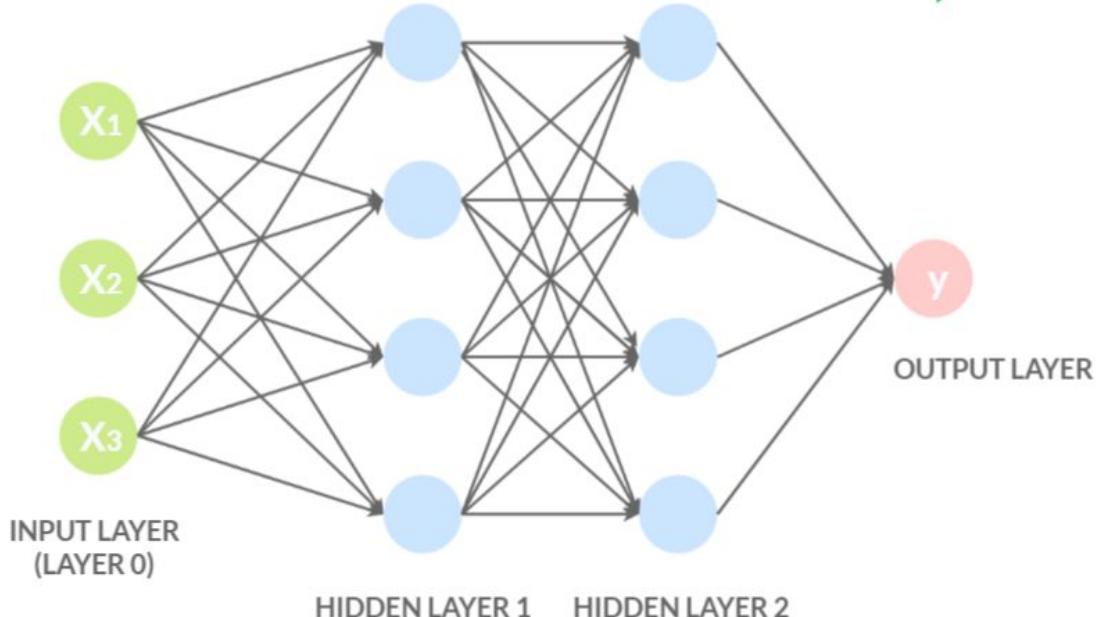
PROBLEMA

Una rete neurale può avere un enorme numero di pesi,
per eseguire l'ottimizzazione dobbiamo sapere quanto ogni peso di ogni layer ha influenzato l'errore.



Addestramento di una Deep Neural Network

FORWARD PROPAGATION: Calcolare le predizioni



BACKWARD PROPAGATION: Aggiornare i pesi

SOLUZIONE: BACKPROPAGATION

Propagando all'indietro l'errore la backpropagation ci permette di sapere in che quantità ogni nodo di ogni layer ha influito sull'errore e possiamo utilizzare questi valori per aggiornare i pesi eseguendo l'ottimizzazione.

Machine Learning: Algoritmi e Modelli

Reti Neurali Artificiali

Universal Approximation Theorem

presentato da
Giuseppe Gullo

Universal Approximation Theorem

Afferma che un MLP con un singolo strato nascosto e un numero sufficientemente grande di neuroni può approssimare qualsiasi funzione continua con qualsiasi grado di precisione.

Universal Approximation Theorem

Solo se

1. La funzione da approssimare è continua
2. La funzione di attivazione è quella corretta

Universal Approximation Theorem

La teoria ci dice che si applica solo ad un MLP con un unico strato nascosto.

La pratica ha dimostrato il contrario.