

# IN550 Machine Learning

## Metodi di ottimizzazione matematica

Vincenzo Bonifaci

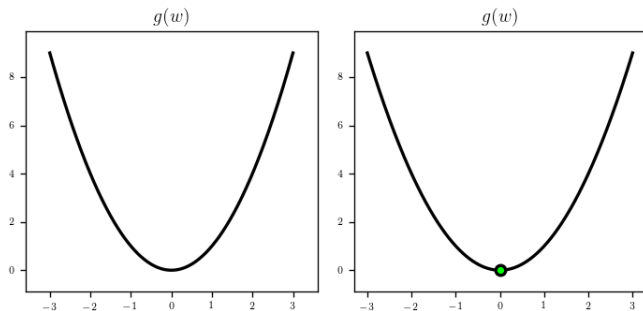
## Problema di minimizzazione globale

**Input:** Una funzione  $g : \mathbb{R}^N \rightarrow \mathbb{R}$

**Output:**  $w^* \in \mathbb{R}^N$  tale che  $g(w^*) \leq g(w)$  per ogni  $w \in \mathbb{R}^N$

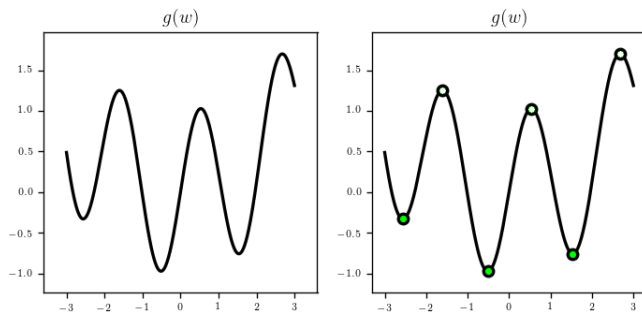
$w^*$  è un *minimo globale* della funzione  $g$

# Minimi globali



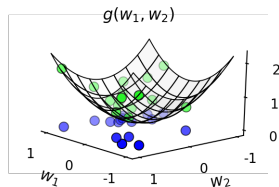
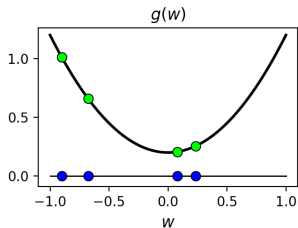
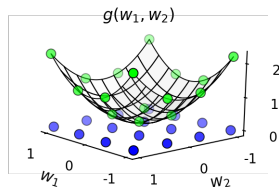
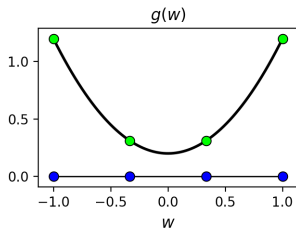
$w^* \in \mathbb{R}^N$  tale che  $g(w^*) \leq g(w)$  per ogni  $w$  in  $\mathbb{R}^N$

# Minimi locali

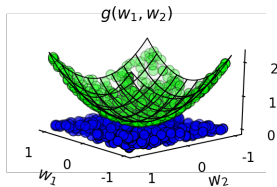
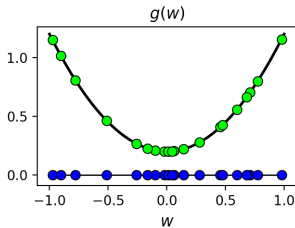
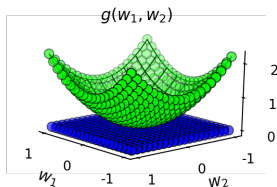
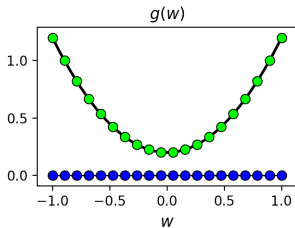


$w^* \in \mathbb{R}^N$  tale che  $g(w^*) \leq g(w)$  per ogni  $w$  in un intorno di  $w^*$

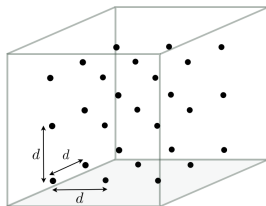
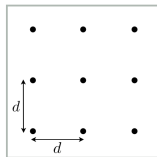
# Due semplici metodi di approssimazione



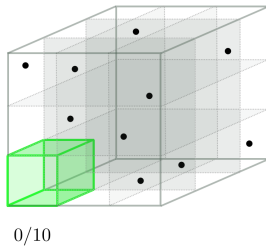
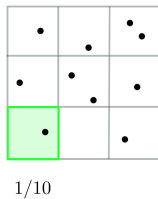
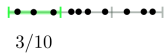
# Due semplici metodi di approssimazione



# La maledizione della multidimensionalità

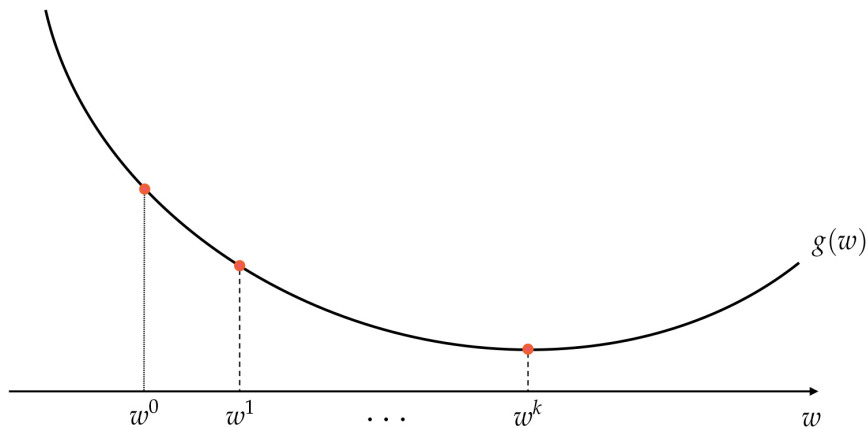


# La maledizione della multidimensionalità

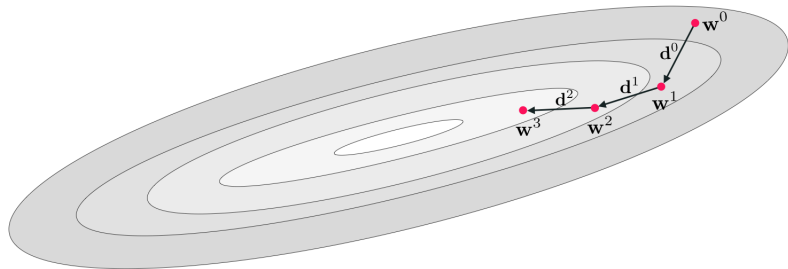




# Metodi di ricerca locale



# Metodi di ricerca locale

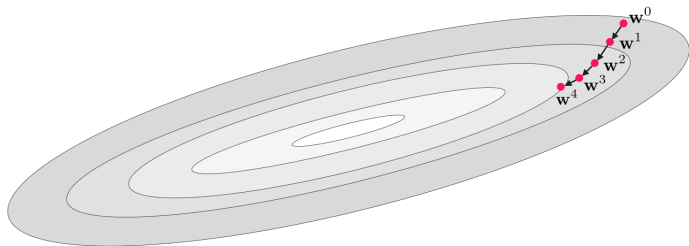
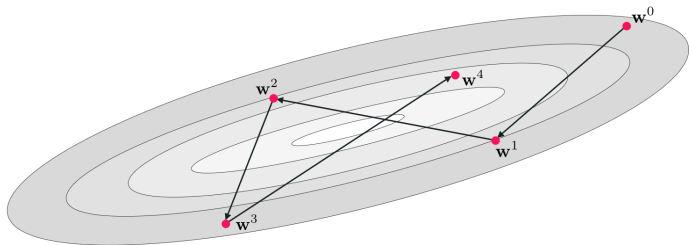


$$w^{(t+1)} = w^{(t)} + d^{(t)}$$

Una condizione desiderabile (**non sempre** soddisfatta) è che

$$g(w^{(0)}) > g(w^{(1)}) > g(w^{(2)}) > \dots > g(w^{(t)}) > \dots$$

# Lunghezza del passo



## Direzione di discesa e passo

Per controllare la lunghezza del passo possiamo porre, più in generale,

$$w^{(t+1)} = w^{(t)} + \eta d^{(t)}$$

- $d^{(t)}$  è la *direzione di discesa* all'iterazione  $t$
- $\eta > 0$  è il *passo* (o *tasso di apprendimento*)

Poiché

$$\|w^{(t+1)} - w^{(t)}\| = \|\eta d^{(t)}\| = \eta \|d^{(t)}\|$$

la lunghezza dello spostamento è direttamente proporzionale a  $\eta$

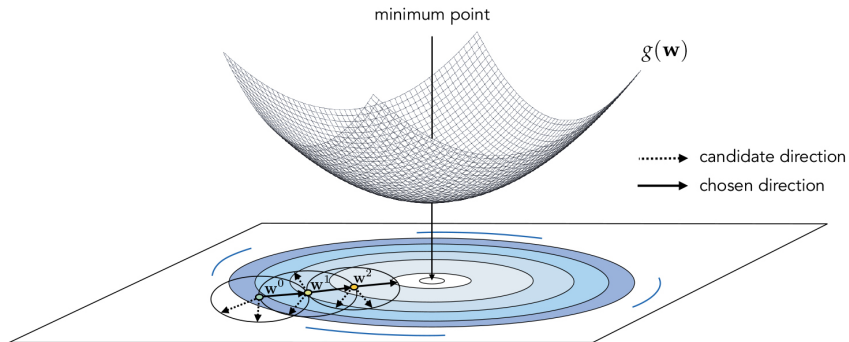
## Randomized Local Search

**Input:** Funzione  $g$ , punto iniziale  $w^{(1)}$

**Inizializzazione:**  $t = 1$

- 1 Genera  $K$  direzioni casuali  $\{d^{(k)}\}_{k=1}^K$
- 2 Valuta  $g(w^{(t)} + \eta d^{(k)})$  per  $k = 1, \dots, K$
- 3 Poni  $w^{(t+1)} = w^{(t)} + \eta d^{(k)}$ , dove  $k$  è l'indice corrispondente al valore più basso osservato
- 4 Se  $g(w^{(t+1)}) < g(w^{(t)})$ , incrementa  $t$  e ricomincia

# Esempio



## Metodi di ordine 0, 1, 2, ...

Randomized Local Search è un esempio di metodo “di ordine 0”:

Un *metodo di ordine 0* utilizza solo i valori della funzione  $g$

Un *metodo di ordine 1* utilizza anche i valori delle derivate prime di  $g$

Un *metodo di ordine 2* utilizza anche i valori delle derivate seconde di  $g$

# Approssimazioni di Taylor: Caso univariato ( $N = 1$ )

## Approssimazione di ordine 1

$$a_1(w) = g(v) + g'(v)(w - v)$$

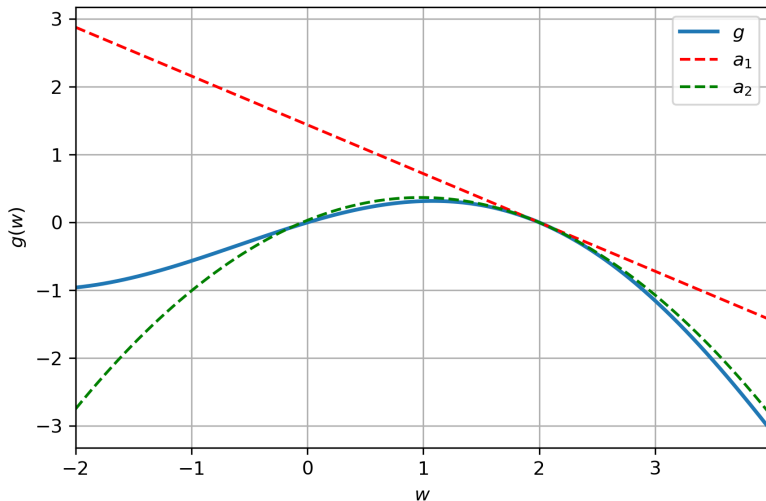
## Approssimazione di ordine 2

$$a_2(w) = g(v) + g'(v)(w - v) + \frac{1}{2}g''(v)(w - v)^2$$



# Esempio

$$g(w) = (w + 8)(w + 4)w(w - 2)(w - 8)/1000, \quad v = 2$$



## Gradiente di $g$ nel punto $v$

$$\nabla_w g(v) \stackrel{\text{def}}{=} \begin{bmatrix} \frac{\partial g}{\partial w_1}(v) \\ \frac{\partial g}{\partial w_2}(v) \\ \dots \\ \frac{\partial g}{\partial w_N}(v) \end{bmatrix}$$

## Hessiana di $g$ nel punto $v$

$$\nabla_w^2 g(v) \stackrel{\text{def}}{=} \begin{bmatrix} \frac{\partial^2 g}{\partial w_1^2}(v) & \frac{\partial^2 g}{\partial w_1 \partial w_2}(v) & \dots & \frac{\partial^2 g}{\partial w_1 \partial w_N}(v) \\ \frac{\partial^2 g}{\partial w_2 \partial w_1}(v) & \frac{\partial^2 g}{\partial w_2^2}(v) & \dots & \frac{\partial^2 g}{\partial w_2 \partial w_N}(v) \\ \dots & \dots & \dots & \dots \\ \frac{\partial^2 g}{\partial w_N \partial w_1}(v) & \frac{\partial^2 g}{\partial w_N \partial w_2}(v) & \dots & \frac{\partial^2 g}{\partial w_N^2}(v) \end{bmatrix}$$

# Approssimazioni di Taylor: Caso generale ( $N \geq 1$ )

## Approssimazione di ordine 1

$$a_1(w) = g(v) + \nabla g(v)^\top (w - v)$$

## Approssimazione di ordine 2

$$a_2(w) = g(v) + \nabla g(v)^\top (w - v) + \frac{1}{2}(w - v)^\top \nabla^2 g(v)(w - v)$$

$\nabla g(v)$  è il *gradiente* di  $g$  (vettore delle derivate prime) in  $v$

$\nabla^2 g(v)$  è l'*Hessiana* di  $g$  (matrice delle derivate seconde) in  $v$

# Condizione di ottimalità al prim'ordine

Se  $g$  è differenziabile, la condizione

$$\nabla g(v) = 0$$

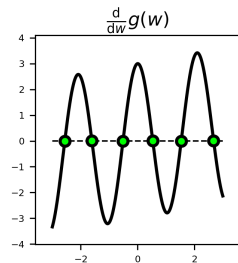
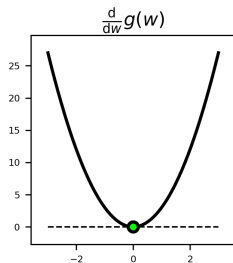
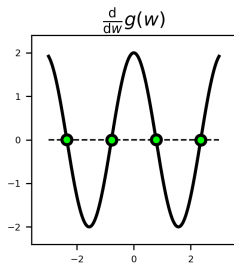
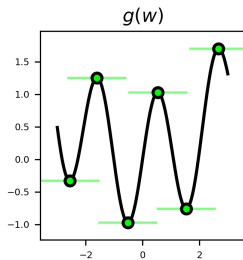
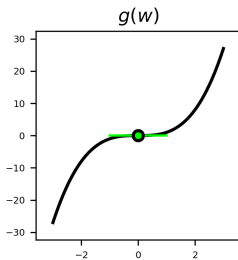
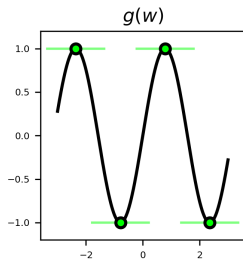
è una condizione **necessaria** affinché  $v$  sia un minimo globale di  $g$  in  $\mathbb{R}^N$

Non è (in generale) sufficiente: identifica solo i *punti critici* di  $g$ :

- minimi/massimi locali
- punti di sella

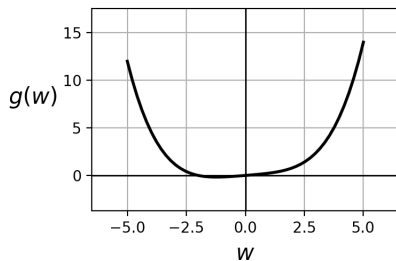
(anche detti *punti stazionari* della funzione)

# Punti critici: Esempi



# Un esempio

$$g(w) = \frac{1}{50}(w^4 + w^2 + 10w)$$



## Un altro esempio

$$g(w) = a + b^\top w + w^\top C w$$

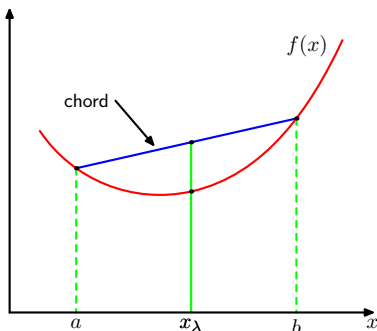
con  $a \in \mathbb{R}$ ,  $b \in \mathbb{R}^N$ ,  $C \in \mathbb{R}^{N \times N}$  ( $C$  simmetrica)

$$\nabla g(w) = 0 \Leftrightarrow Cw = -\frac{1}{2}b$$

## Funzione convessa (definizione di ordine 0)

Una funzione  $f : \mathbb{R}^N \rightarrow \mathbb{R}$  è *convessa* se per ogni  $\lambda \in [0, 1]$ ,  $a, b \in \mathbb{R}^N$ ,

$$f(\lambda a + (1 - \lambda)b) \leq \lambda f(a) + (1 - \lambda)f(b)$$





## Funzioni convesse: definizione alternativa (ordine 1)

Se  $f$  è differenziabile,  $f$  è convessa se e solo se

$$f(b) \geq f(a) + \nabla f(a)^\top (b - a) \quad \text{per ogni } a, b \in \mathbb{R}^N$$

## Funzioni convesse: definizione alternativa (ordine 2)

Se  $f$  è due volte differenziabile,  $f$  è convessa se e solo se l'Hessiana

$$\nabla^2 f(x)$$

ha tutti gli **autovalori** non-negativi, per ogni  $x \in \mathbb{R}^N$

(Una matrice con tutti gli autovalori  $\geq 0$  è detta *semidefinita positiva*)

Esempi con  $N = 1$ :

- $g(w) = w^3$   
non è convessa
- $g(w) = e^w$   
è convessa
- $g(w) = \sin(w)$   
non è convessa
- $g(w) = w^2$   
è convessa
- $g(w) = |w|$   
è convessa

# Funzioni convesse vs. non convesse: Esempi

Esempi con  $N > 1$ :

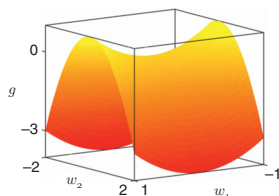
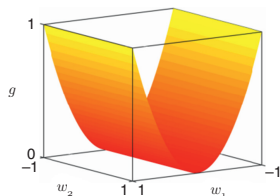
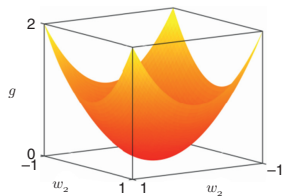
$$g(w) = \frac{1}{2} w^\top Q w + r^\top w + b$$

con  $Q$  simmetrica

$$\nabla^2 g(w) = Q$$

- con  $Q = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}$  è convessa
- con  $Q = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$  è convessa
- con  $Q = \begin{bmatrix} 2 & 0 \\ 0 & -1 \end{bmatrix}$  non è convessa

# Funzioni convesse vs. non convesse: Esempi



# Condizione di ottimalità per funzioni convesse

Se  $g$  è convessa e differenziabile, la condizione

$$\nabla g(w) = 0$$

è **necessaria e sufficiente** affinché  $w$  sia un minimo globale.

Infatti, per ogni  $w'$ ,

$$g(w') \geq g(w) + \nabla g(w)^\top (w' - w) = g(w)$$

⇒ Nelle funzioni convesse, i minimi locali sono anche globali ⇐

- Ogni funzione lineare è convessa
- Se  $f$  è convessa e  $c \geq 0$ ,  $c \cdot f(x)$  è convessa
- Se  $f$  e  $g$  sono convesse,  $f(x) + g(x)$  è convessa
- Se  $f$  e  $g$  sono convesse,  $\max(f(x), g(x))$  è convessa
- Se  $a$  è lineare e  $f$  è convessa,  $f(a(x))$  è convessa
- Se  $C$  è simmetrica con autovalori  $\geq 0$ ,  $x^T Cx$  è convessa
- Se  $C = v \cdot v^T$ ,  $x^T Cx$  è convessa
- Se  $\nabla^2 f$  è simmetrica con autovalori  $\geq 0$ ,  $f(x)$  è convessa

Una *norma* è una funzione  $\|\cdot\| : \mathbb{R}^N \rightarrow \mathbb{R}$  con le seguenti proprietà:

- 1  $\|x\| = 0$  se e solo se  $x = 0$
- 2  $\|x\| \geq 0$  per ogni  $x \in \mathbb{R}^N$
- 3  $\|cx\| = c \|x\|$  per ogni  $x \in \mathbb{R}^N$  e  $c > 0$
- 4  $\|x + y\| \leq \|x\| + \|y\|$

Esempi:

- norma  $L_2$  (euclidea):  $\|x\|_2 = (\sum_j |x_j|^2)^{1/2}$
- norma  $L_1$ :  $\|x\|_1 = \sum_j |x_j|$
- norma  $L_p$ :  $\|x\|_p = (\sum_j |x_j|^p)^{1/p}$  ( $p \geq 1$ )
- norma  $L_\infty$  (uniforme):  $\|x\|_\infty = \max_j |x_j|$

Ogni norma è una funzione convessa (perché?)



- La funzione *quadrato dell'errore*

$$\ell(y) = (y - y^*)^2$$

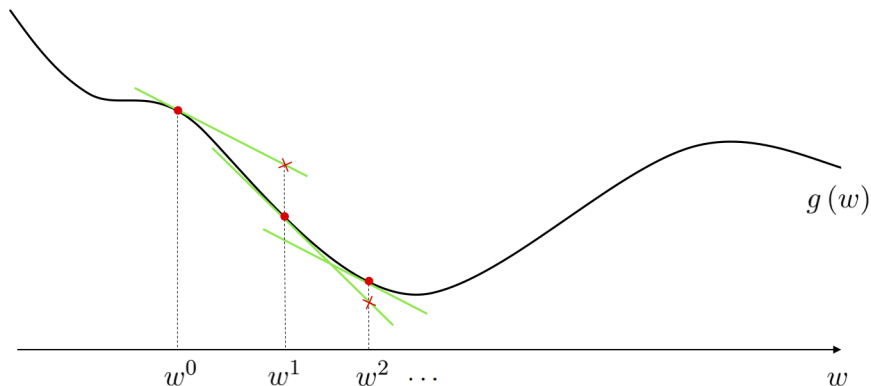
è convessa in  $y$

- La funzione *costo 0-1*

$$\ell(y) = \begin{cases} 0 & \text{se } y = y^* \\ 1 & \text{se } y \neq y^* \end{cases}$$

**non** è convessa in  $y$  (non è neanche continua)

# Discesa del gradiente [Gradient Descent]



Cauchy (1847)

# Algoritmo di discesa del gradiente (GD)

## Algoritmo Gradient Descent (versione generica)

**Input:** Funzione  $g$ , punto iniziale  $w^{(1)}$

**1** Per  $t = 1, \dots, T$ :

$$w^{(t+1)} = w^{(t)} - \eta \cdot \nabla g(w^{(t)})$$

**2** Restituisci il  $w^{(t)}$  col minimo valore di  $g(w^{(t)})$ :

$$w^{\text{GD}} = \underset{t=1, \dots, T}{\operatorname{argmin}} g(w^{(t)})$$

L'algoritmo ha due *iperparametri*:  $\eta$  (passo) e  $T$  (numero di passi)

## Teorema (Convergenza di GD – Versione 1)

Sia  $g : \mathbb{R}^N \rightarrow \mathbb{R}$  convessa. Se esistono costanti  $D, G > 0$  tali che:

- $\|w^{(1)} - w^*\| \leq D,$
- $\|\nabla g(w^{(t)})\| \leq G$  per  $t = 1, \dots, T,$

allora per  $\eta = D/(G\sqrt{T})$  il metodo soddisfa

$$g(w^{\text{GD}}) - g(w^*) \leq \frac{DG}{\sqrt{T}}$$

$\Rightarrow$  GD converge al valore minimo di  $g$  per  $T$  crescente

L'errore decresce asintoticamente almeno come  $1/\sqrt{T}$

# Convergenza di GD

**Def.** Una funzione convessa  $g : \mathbb{R}^N \rightarrow \mathbb{R}$  è detta  $\beta$ -smooth se  $\forall x, y$

$$g(y) - g(x) - \nabla g(x)^\top (y - x) \leq \frac{\beta}{2} \|x - y\|^2$$

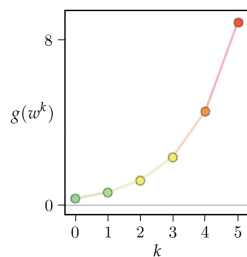
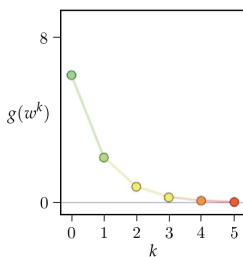
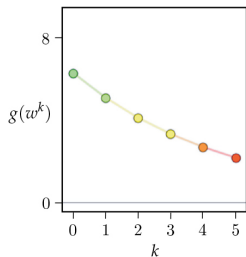
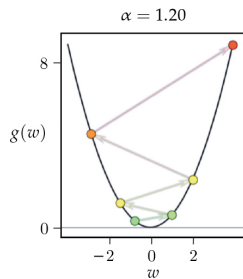
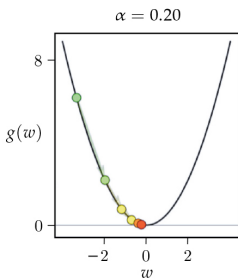
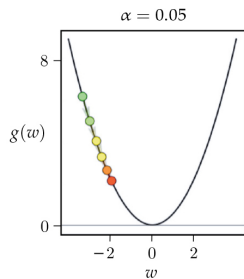
## Teorema (Convergenza di GD – Versione 2)

Sia  $g : \mathbb{R}^N \rightarrow \mathbb{R}$  convessa e  $\beta$ -smooth. Allora GD con  $\eta = 1/\beta$  soddisfa

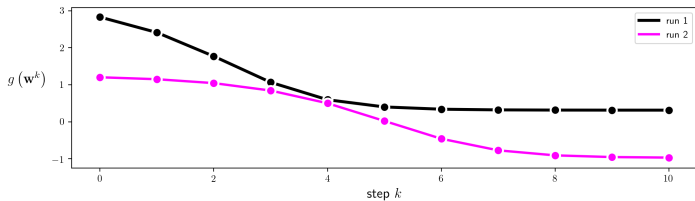
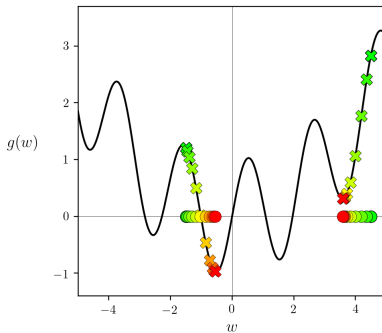
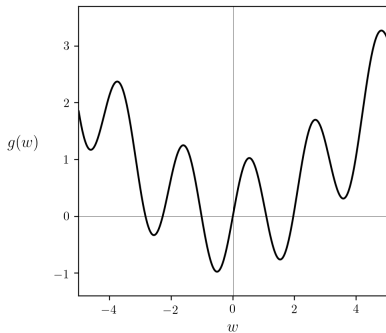
$$g(w^{\text{GD}}) - g(w^*) \leq \frac{2\beta \left\| w^{(1)} - w^* \right\|^2}{T - 1}$$

In questo scenario l'errore decresce asintoticamente almeno come  $1/T$

# Esempio con diversi valori del passo $\eta$



# Esempio non convesso



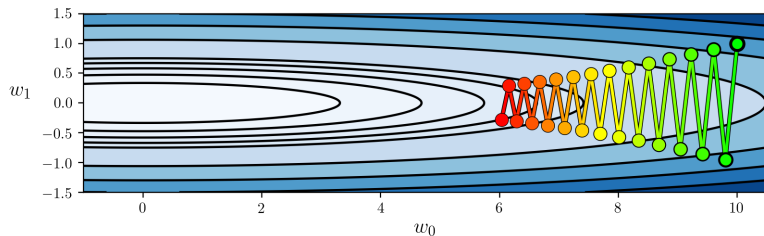
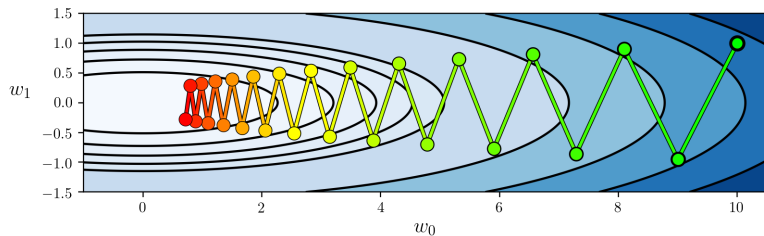
## Due problematiche di GD

- La **direzione** del gradiente negativo può oscillare, portando l'algoritmo a muoversi a "zig-zag" e convergere lentamente
- La **magnitudine** del gradiente negativo si contrae vicino ai punti critici, rallentando la discesa

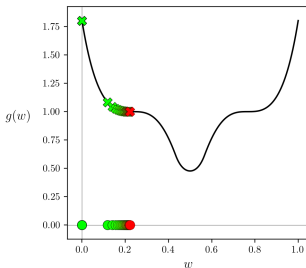
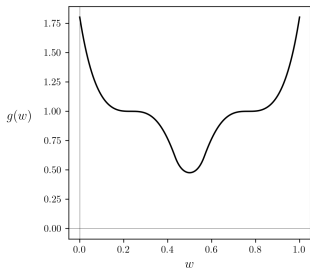
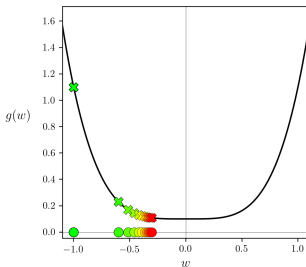
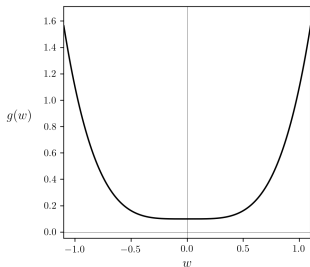
Per questo motivo le librerie di ML adottano varianti di GD più sofisticate (Momentum, Adam, RMSprop...)



# Movimento a zig-zag: Esempi

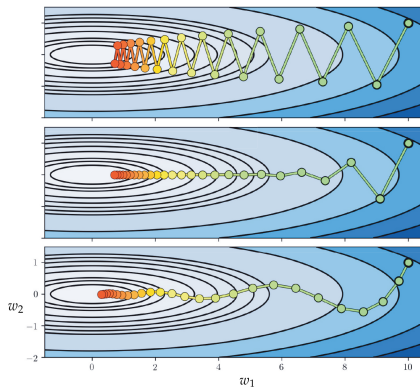


# Rallentamento vicino ai punti critici: Esempi



# Varianti sofisticate di GD

- Momentum
- Gradiente normalizzato
- Adagrad/RMSprop
- Adam



Discussi nell'Appendice A del libro di testo

# Metodi del secondo ordine

GD è un esempio di *metodo del primo ordine* in quanto usa solo:

- i valori della funzione,  $g(x)$  (scalari)
- i valori del gradiente,  $\nabla g(x)$  (vettori  $N \times 1$ )

I *metodi del secondo ordine* utilizzano anche

- i valori dell'Hessiana,  $\nabla^2 g(x)$  (matrici  $N \times N$ )

Il *metodo di Newton* ne è l'esempio più noto

L'uso di metodi del secondo ordine nel ML è **fortemente limitato** dal fatto che richiedono la manipolazione esplicita di matrici  $N \times N$  (potenzialmente enormi) ad ogni passo dell'algoritmo

## Metodo di Newton-Raphson

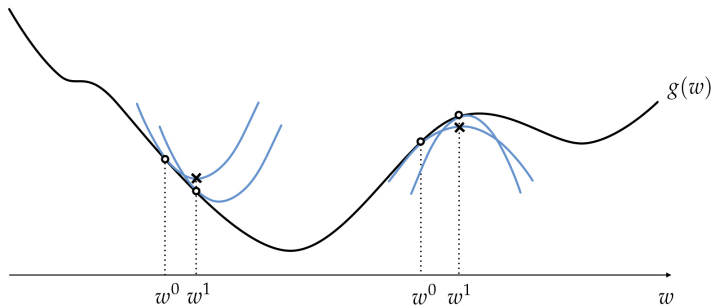
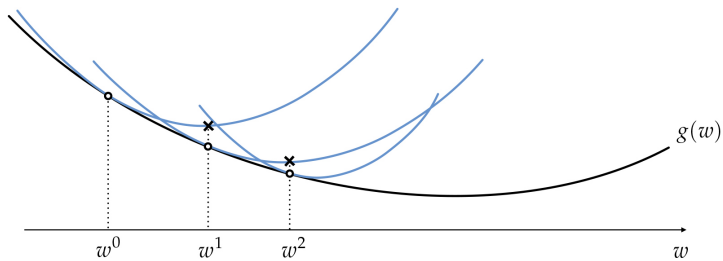
**Input:** Funzione  $g$ , punto iniziale  $w^{(1)}$

1 Per  $t = 1, \dots, T$ :

$$w^{(t+1)} = w^{(t)} - [\nabla^2 g(w^{(t)})]^{-1} \nabla g(w^{(t)})$$

2 Restituisci  $w^{(T)}$

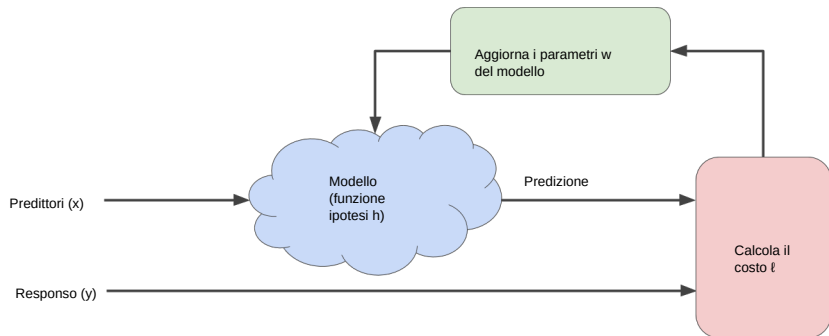
# Esempio (metodo di Newton)



## Vantaggi e svantaggi dei metodi del secondo ordine

- Richiedono tipicamente meno iterazioni per convergere
- La singola iterazione è più computazionalmente costosa (richiede di invertire una Hessiana  $N \times N$ )
- Applicabili solo a funzioni differenziabili e convesse (altrimenti l'Hessiana non sempre è definita e/o invertibile)

# Minimizzazione iterativa del rischio empirico





# Discesa del gradiente (GD): calcolo di un passo

Il metodo GD calcola  $\nabla g(w)$  ad ogni passo

Nel ML,  $g$  è il rischio empirico, funzione di **tutti** gli esempi di training:

$$g(w) = \frac{1}{m} \sum_{i=1}^m \overbrace{\ell(h(x^{(i)}), y^{(i)})}^{\ell_i(w)} = \frac{1}{m} \sum_{i=1}^m \ell_i(w)$$
$$\nabla g(w) = \frac{1}{m} \sum_{i=1}^m \nabla \ell_i(w)$$

dove:

- $h$  è la funzione ipotesi codificata dal vettore  $w$
- $\ell_i(w)$  è il costo sull'esempio  $i$ -esimo in funzione di  $w$

## Gradient Descent (Batch GD)

- 1 Inizializza  $w^{(1)}$
- 2 Per  $t = 1, \dots, T$ :
  - Poni  $w^{(t+1)} = w^{(t)} - \eta \cdot \frac{1}{m} \sum_i \nabla \ell_i(w^{(t)})$
- 3 Restituisci il  $w^{(t)}$  col minimo valore di  $g(w^{(t)})$ ,  $t = 1, \dots, T$

Ogni passo del metodo GD richiede di considerare **tutti** gli  $m$  esempi (si parla di modalità *batch*)

# Discesa stocastica del gradiente (SGD)

Per dei passi più rapidi, si usa una variante stocastica di GD

## Stochastic Gradient Descent (SGD)

- 1 Inizializza  $w^{(1)}$
- 2 Per  $t = 1, \dots, T$ :
  - Estrai un esempio  $i$  a caso
  - Poni  $w^{(t+1)} = w^{(t)} - \eta \cdot \nabla \ell_i(w^{(t)})$
- 3 Restituisci il  $w^{(t)}$  col minimo valore di  $g(w^{(t)})$ ,  $t = 1, \dots, T$

Ogni passo del metodo SGD richiede di considerare **un solo** esempio

Grande risparmio computazionale rispetto al metodo GD batch

Ma la localizzazione dell'ottimo diviene meno precisa

# Esempio 1: SGD per la regressione lineare

## Esempio

Derivare la regola di aggiornamento di SGD nel seguente scenario:

- Ipotesi lineari:  $h(x) = w^\top x$
- Costo quadratico:  $\ell(w) = (w^\top x - y)^2$

Calcolando il gradiente di  $\ell$ , otteniamo la regola di aggiornamento

## Regola Least-Mean-Squares (LMS)

$$w \leftarrow w - 2\eta (w^\top x - y) x$$

## Esempio 2: SGD per la regressione LAD

### Esempio

Derivare la regola di aggiornamento di SGD nel seguente scenario:

- Ipotesi lineari:  $h_w(x) = w^\top x$
- Costo LAD:  $\ell(w) = |w^\top x - y|$

Calcolando il (sub)gradiente di  $\ell$ , otteniamo la regola di aggiornamento

$$w \leftarrow w - \eta \left( \text{sgn}(w^\top x - y) \right) x$$

$$\text{dove } (\text{sgn}(z))_j = \begin{cases} +1 & \text{se } z_j > 0 \\ -1 & \text{se } z_j < 0 \\ 0 & \text{se } z_j = 0 \end{cases}$$

# Mini-batch SGD

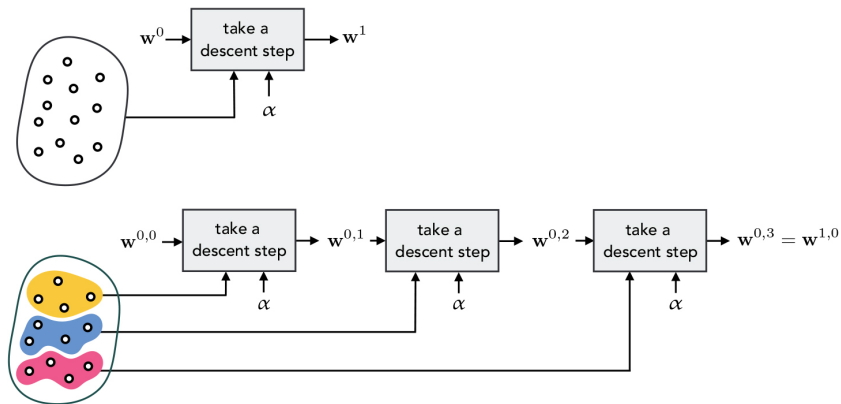
Mini-batch SGD è un compromesso tra GD e SGD

## Mini-Batch SGD

- 1 Inizializza  $w^{(1)}$
- 2 Per  $t = 1, \dots, T$ :
  - Estrai un lotto (*batch*)  $B$  di esempi **a caso**
  - Poni  $w^{(t+1)} = w^{(t)} - \eta \cdot \frac{1}{|B|} \sum_{i \in B} \nabla \ell_i(w^{(t)})$
- 3 Restituisci il  $w^{(t)}$  col minimo valore di  $g(w^{(t)})$ ,  $t = 1, \dots, T$

Ogni passo di mini-batch SGD richiede di considerare  $|B|$  esempi  
Se  $|B| = 1$ , equivale a SGD

# Batch GD vs. Mini-Batch SGD



# Batch GD vs. SGD vs. Mini-Batch SGD

