Università di Verona

A.A. 2020-21

Machine Learning & Artificial Intelligence

Funzioni Discriminanti Lineari

Vittorio Murino

Classificatori con funzioni discriminanti lineari

- Dato un problema a 2 classi, l'obiettivo è quello di creare una funzione lineare $g(\mathbf{x})$ che separi gli esempi appartenenti alle due classi

  $g(\mathbf{x}) = \mathbf{w}^t\mathbf{x} + w_0$

  dove $\mathbf{x}=[x_1,...,x_n]$, $\mathbf{w}=[w_1,...,w_n]$ (pesi) e w_0 *bias* (soglia).

- *Un campione* $\mathbf{x}_i$ *è classificato come appartenente a* ω_1 *se* $\mathbf{w}^t\mathbf{x}_i + w_0 > 0$, *a* ω_2 *se* $\mathbf{w}^t\mathbf{x}_i + w_0 < 0$,

- Più genericamente, consideriamo come $\mathbf{w'} = [w_0,w_1,...,w_n]$ e $\mathbf{x'} = [1,x_1,...,x_n]$, quindi la regola di decisione diventa:

  Un campione $\mathbf{x}_i$ *è classificato come appartenente a* ω_1 *se* $\mathbf{w'}^t\mathbf{x}_i' > 0$, *a* ω_2 *se* $\mathbf{w'}^t\mathbf{x}_i' < 0$

- Supponiamo di avere un insieme di m campioni $x_1,...,x_m$, alcuni etichettati ω_1 ed altri etichettati ω_2
- Vogliamo utilizzare tali campioni per determinare i pesi $\mathbf{w}$ e w_0 della funzione discriminante lineare
- Si supponga inoltre che esista una soluzione per la quale la probabilità di errore è molto piccola.
- Quindi, un ragionevole approccio è la ricerca di un vettore di pesi tale che la probabilità di commettere errore sui campioni sia minima.
- Se esiste un vettore di pesi tale da rendere nulla la probabilità di errore, allora i campioni si dicono *linearmente separabili*.

- L’obiettivo è quindi quello di calcolare tali pesi per cui

  $\mathbf{w}^t\mathbf{x}_i > 0$ per ogni $\mathbf{x}_i$ appartenente a ω_1

  $\mathbf{w}^t\mathbf{x}_i < 0$ per ogni $\mathbf{x}_i$ appartenente a ω_2
- In quest'ultimo caso si può anche dire che $\mathbf{x}_i$ è classificato correttamente se $\mathbf{w}^t(-\mathbf{x}_i) > 0$.
- Questo suggerisce una normalizzazione che semplifica il trattamento nel caso di due diverse classi, ossia il fatto che si possa solo trovare il vettore dei pesi tale che $\mathbf{w}^t\mathbf{x}_i > 0$ per tutti i campioni a prescindere dalle classi.
- Questo vettore è chiamato vettore separatore o vettore soluzione.

- Il vettore dei pesi $\mathbf{w}$ si può considerare come un punto nello spazio dei pesi.
- Ogni campione $\mathbf{x}_i$ pone un vincolo sulla possibile collocazione del vettore soluzione.
- L'equazione $\mathbf{w}^t\mathbf{x}_i = 0$ definisce un iperpiano passante per l'origine dello spazio dei pesi avente $\mathbf{x}_i$ come vettore normale.
- Il vettore soluzione, se esiste, deve essere nella parte positiva di ogni iperpiano e deve giacere nell'intersezione degli n semispazi.
- Quindi, ogni vettore in questa regione è il vettore soluzione e la regione corrispondente è la regione soluzione.

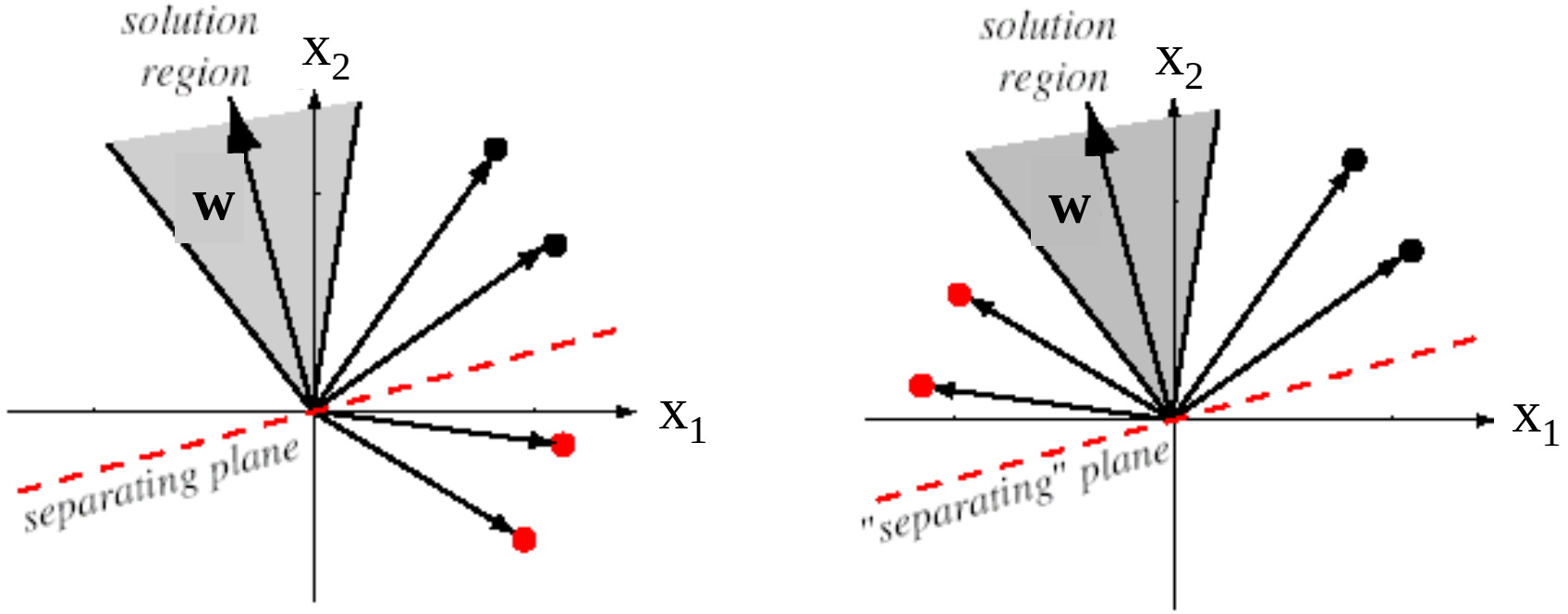


FIGURE 5.8. Four training samples (black for ω_1, red for ω_2) and the solution region in feature space. The figure on the left shows the raw data; the solution vectors leads to a plane that separates the patterns from the two categories. In the figure on the right, the red points have been "normalized"—that is, changed in sign. Now the solution vector leads to a plane that places all "normalized" points on the same side. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

- Risulta chiaro, quindi, che se il vettore soluzione esiste non è unico.
- Ci sono diversi modi per imporre requisiti addizionali per vincolare il vettore soluzione.
- Uno potrebbe essere quello di cercare il vettore dei pesi di lunghezza unitaria che massimizzi la minima distanza dei campioni dal piano separatore.
- Un'altra potrebbe essere quella di cercare il vettore dei pesi a lunghezza minima che soddisfi

$$\mathbf{w}^t \mathbf{x}_i \geq b, \quad \forall i$$

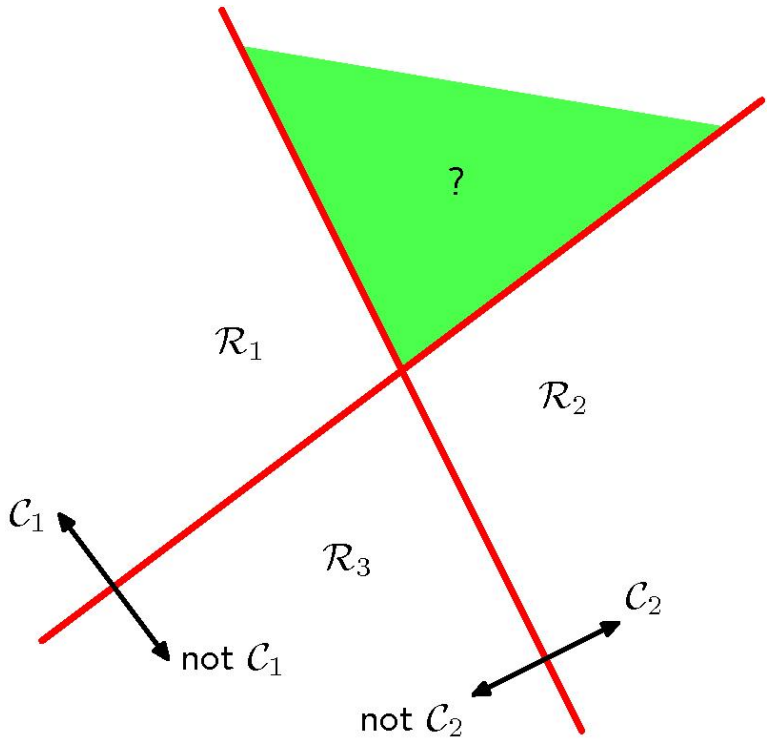
dove *b* è una costante positiva chiamata *margine*.

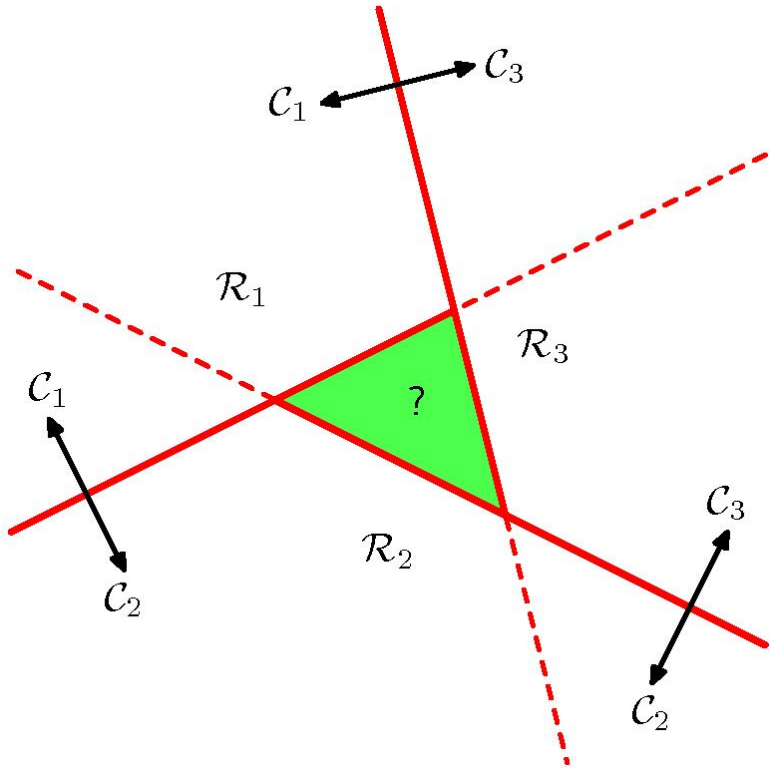
- Queste tecniche cercano di trovare un vettore soluzione vicino al centro della regione soluzione:
  - in questo modo nuovi campioni potrebbero essere classificati correttamente.

- Nel caso di problemi a C classi, possono essere *costruiti* C*-1* classificatori $g_i(\mathbf{x})$, uno per ogni classe C_i contro *non-*C_i, chiamati classificatori *one-vs-rest*

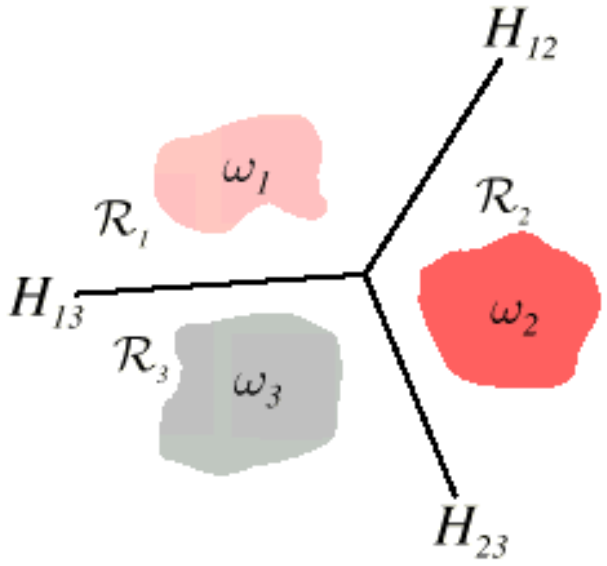
- Oppure si possono costruire $C(C-1)/2$ classificatori binari (*one-vs-one*), e quindi classificare un punto a maggioranza

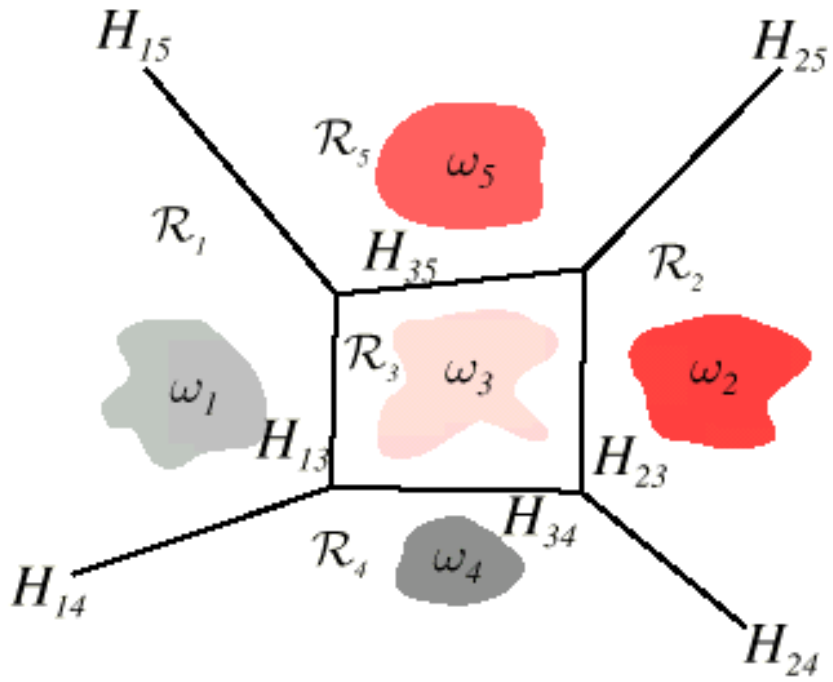
- Entrambi portano a zone di ambiguità.





- Si risolve costruendo un singolo classificatore a C classi che comprende C funzioni lineari e quindi un vettore ignoto $\mathbf{x}_i$ viene assegnato alla classe C_i se $g_i(\mathbf{x}) > g_j(\mathbf{x})$ per ogni i diverso da j.





- In generale lo spazio delle *feature* può essere partizionato in sottoregioni coincidenti con le diverse classi di riconoscimento.
- Ma le superfici di decisione che delimitano le sottoregioni appartenenti a classi differenti possono essere descritte da funzioni discriminanti che possono essere lineari, lineari a tratti o non lineari.

Classificatore lineare

$$g(\mathbf{x}) = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

dove w_i è l'elemento del vettore dei pesi e w_0 è il peso di soglia.

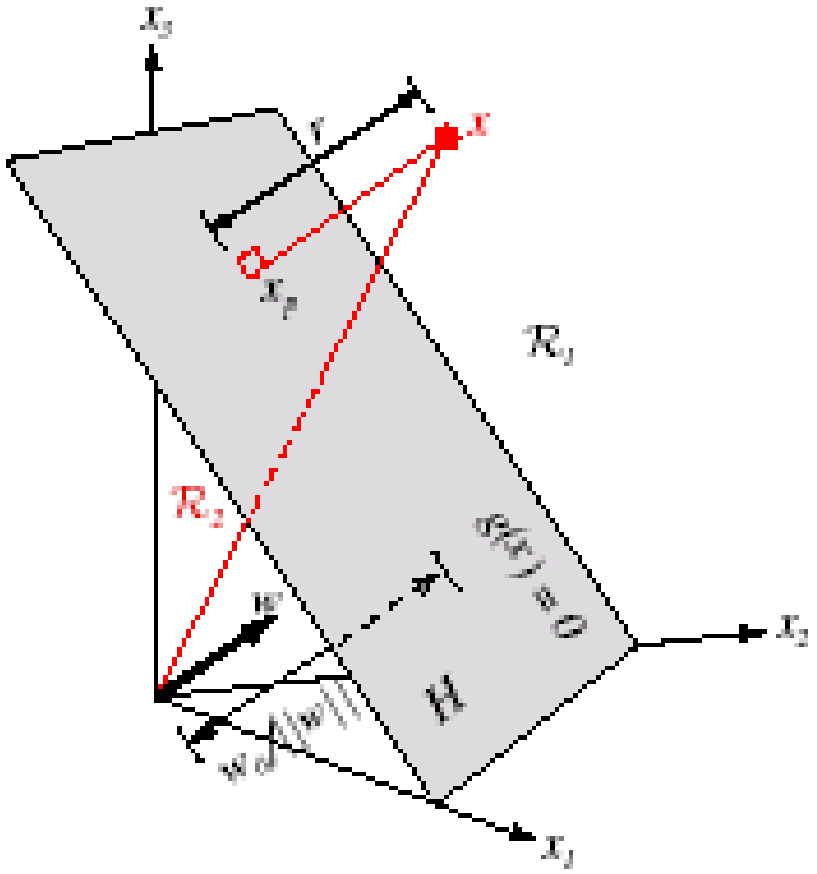
Classificatore quadratico

$$g(\mathbf{x}) = w_0 + \sum_{i=1}^{n} w_i \cdot x_i + \sum_{i=1}^{n} \sum_{j=1}^{n} v_{ij} x_i x_j$$

L'equazione $g(\mathbf{x}) = 0$ determina una superficie di decisione che separa i punti appartenenti a classi diverse.

- Nel caso lineare devo cercare di determinare un vettore $\mathbf{w}$, che definisce una superficie di decisione rappresentata come un iperpiano.
- Nel secondo caso, in modo simile, devo determinare, oltre a $\mathbf{w}$, anche la matrice [$\mathbf{V}$] (superficie di decisione rappresentata come una iperquadrica).
- Si dimostra che $\mathbf{w}$ è ortogonale a qualsiasi vettore giacente sull'iperpiano.
- La funzione discriminante $g(\mathbf{x})$ dà una misura algebrica della distanza di $\mathbf{x}$ dall'iperpiano.
- Quindi, una funzione discriminante lineare divide lo spazio delle *feature* mediante una superficie di decisione rappresentata come un iperpiano.

- Considerando il problema dal punto di vista geometrico, l'orientazione di tale superficie è determinata dal vettore normale $\mathbf{w}$, mentre la locazione della superficie è determinata dal peso di soglia w_0.



- Ci occuperemo in gran parte del calcolo di funzioni discriminanti lineari perchè:
  a) le funzioni lineari sono le più semplici;
  b) ogni funzione discriminante è riconducibile ad una funzione lineare a tratti, a meno di un certo errore.
  c) è possibile effettuare trasformazioni non-lineari sulle feature per avere poi piani di classificazione descrivibili con funzioni discriminanti lineari.

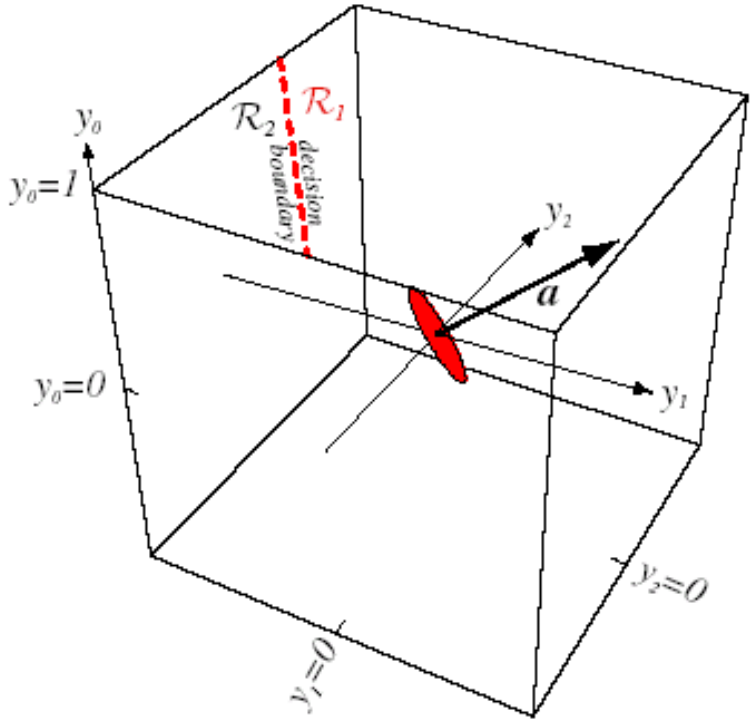
Vettore aggiunto (o aumentato)

- Poichè nel caso dei classificatori lineari si vuole una espressione del tipo $\mathbf{w'}^{t}\mathbf{x'} > 0$, occorre che $\mathbf{x}$ sia un vettore con una ulteriore componente aggiunta pari a 1 che ci consenta di determinare w_0 insieme alle altre componenti w_i.

- Il problema è trasformato in forma omogenea:
  - l'aggiunta di una componente uguale su tutti i vettori $\mathbf{x}$ non modifica la distanza tra 2 punti lasciando inalterata la classificazione;
  - la nuova superficie (iperpiano) di decisione passa attraverso l'origine anche se la superficie originaria (ovvero, considerando solo i campioni originali senza la componente aggiunta) può essere ovunque nello spazio delle feature;

- la distanza dei nuovi campioni dalla superficie di decisione è minore o uguale alla distanza dei campioni originali dalla superficie di decisione originale.

- In definitiva, il problema di trovare un vettore dei pesi $\mathbf{w}$ e una soglia w_0 è stato trasformato nel problema di trovare un unico vettore dei pesi $\mathbf{w}'$.



- Quindi, il problema di determinare le funzioni discriminanti dato un certo insieme di campioni di training, coincide con quello di determinare i pesi $\mathbf{w}$ per ciascuna classe *i*.
- Se pertanto avrò due classi ω_1 ed ω_2, mi aspetterò che sia possibile trovare un vettore $\mathbf{w}$' tale che

$$\mathbf{w}'^{t}\,\mathbf{x}_m^1 > 0$$

$$\mathbf{w}'^{t}\,\mathbf{x}_m^2 < 0$$

Quindi, di poter associare $\mathbf{x}_m^1$ alla classe ω_1 e $\mathbf{x}_m^2$ alla classe ω_2.

- In genere, per evitare di trovare un vettore $\mathbf{w}$ dalle componenti troppo piccole, si preferisce porre una condizione più stringente sul vettore dei pesi associato alla funzione discriminante di una classe, del tipo:

$$\mathbf{w}'^{t}\,\mathbf{x}_m^k > b$$

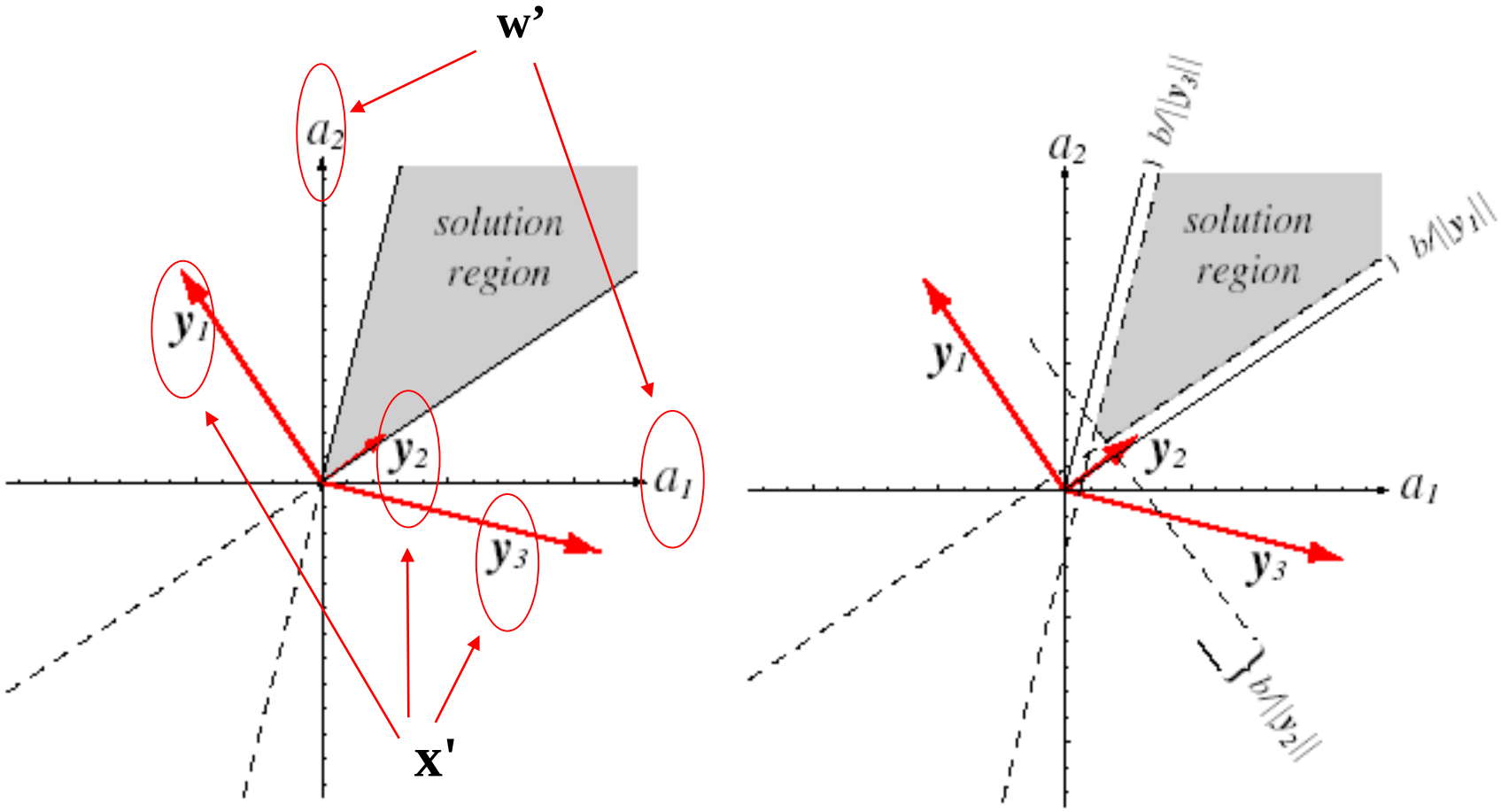
con $b > 0$, b margine (in generale potrebbe esserci un diverso b per ogni campione).

- Con l’introduzione del margine, la regione soluzione giace all’interno della regione soluzione originale ed è isolata rispetto ai vecchi contorni della regione soluzione dal margine.

- Intuitivamente, il margine dovrebbe migliorare la regione di decisione affinchè nuovi campioni vengano classificati correttamente.

- Solitamente, si pone $b = 1$ senza perdita di generalità.

- Una volta stabilita l'analogia tra i due problemi vediamo i metodi con cui si può trovare il vettore dei pesi.



We are in the case of adjunct vector

Funzioni discriminanti lineari

- Si consideri quindi una funzione h del tipo:

$$g_i(\mathbf{x}) = h_i(\mathbf{w}, \mathbf{x}') = w_0 + \sum_{k=1}^{n} w_k x_k$$

Tecnica del Gradiente Discendente

- La tecnica del Gradiente Discendente è uno degli approcci più semplici per il calcolo di una funzione discriminante.
- È un metodo iterativo di assestamento progressivo dei pesi che si basa sulla seguente proprietà:

  il vettore gradiente nello spazio W punta nella direzione di massimo scarto di una funzione da massimizzare/minimizzare.

- La procedura consiste nell'aggiornare il valore del vettore dei pesi al passo $k+1$ con un contributo proporzionale al modulo del gradiente stesso al passo precedente e può essere formulata come:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \rho_k \nabla J(\mathbf{w})\Big|_{\mathbf{w}=\mathbf{w}(k)} \qquad (1)$$

  dove $J(\mathbf{w})$ è una funzione di valutazione che deve essere minimizzata.

- $J(\mathbf{w})$ viene scelta in modo tale da raggiungere il minimo all'avvicinarsi di $\mathbf{w}$ alla soluzione ottima, ovvero convessa.

- Il minimo di $J(\mathbf{w})$ si ottiene spostando $\mathbf{w}$ in direzione opposta al gradiente.
- ∇ è il simbolo dell'operatore gradiente, dato da:

$$\nabla J = \begin{bmatrix} \dfrac{\partial J}{\partial w_1} \\ \vdots \\ \vdots \\ \vdots \\ \dfrac{\partial J}{\partial w_n} \end{bmatrix}$$

- ρ_k è uno scalare opportuno che varia con l'iterazione, k, fissando l'"ampiezza" nella correzione.

- La procedura che viene seguita al passo k-esimo può essere riassunta come:
  1) Scegliere arbitrariamente un vettore dei pesi $\mathbf{w}(1)$ e calcolare il vettore gradiente $J(\mathbf{w}(1))$;
  2) Ottenere così un vettore $\mathbf{w}(2)$ ad una certa distanza (fissata anche per mezzo di ρ_k) nella direzione di massima discesa (*steepest descent*).

- Possiamo andare oltre per determinare la scelta ottima di ρ_k .

- A tal fine, dobbiamo fare qualche ipotesi in più sulla forma di $J(\mathbf{w})$.

- In particolare, supponiamo che J sia approssimabile in serie di Taylor al secondo ordine, cioè:

$$J\left[\mathbf{w}(k+1)\right] \cong J\left[\mathbf{w}(k)\right] + \nabla J\left[\mathbf{w}(k)\right]\left(\mathbf{w}(k+1) - \mathbf{w}(k)\right) + \frac{1}{2}\left(\mathbf{w}(k+1) - \mathbf{w}(k)\right)^t \mathbf{D}\left(\mathbf{w}(k+1) - \mathbf{w}(k)\right) \quad (2)$$

- $\mathbf{D}$ è la matrice Hessiana di J (di dimensione $(n+1)\times(n+1)$) cioè:

$$\mathbf{D} \Rightarrow D_{ij} = \left.\frac{\partial^2 J}{\partial w_i \partial w_j}\right|_{\mathbf{w}=\mathbf{w}(k)}$$

- Sostituendo l'equazione (1) nella (2) si ha:

$$J\left[\mathbf{w}(k+1)\right] \cong J\left[\mathbf{w}(k)\right] - \rho_k \left\|\nabla J\left[\mathbf{w}(k)\right]\right\|^2 + \frac{1}{2}\rho_k^2 \nabla J^t \mathbf{D} \nabla J \quad (3)$$

- Poniamo a questo punto di voler trovare quel valore di ρ_k per cui si ha un minimo della funzione J al passo k+1.
- Questo significa imporre il seguente vincolo:

$$\frac{\partial J\left[w(k+1)\right]}{\partial \rho_k} = 0 \qquad (4)$$

- La condizione (4), applicata alla (3) si traduce nel vincolo:

$$-\left|\nabla J\left[\mathbf{w}(k)\right]\right|^2 + \rho_k \nabla J^t \mathbf{D} \nabla J = 0$$

che dà luogo ad un valore ottimo di ρ_k dato da: $$\rho_k = \left.\frac{|\nabla J|^2}{\nabla J^t \mathbf{D} \nabla J}\right|_{\mathbf{w}=\mathbf{w}(k)}$$

- Un altro modo possibile per ottenere la soluzione del nostro problema (rendere minimo $J(\mathbf{w})$) è quello di cercare un minimo rispetto a $\mathbf{w}(k+1)$, cioè porre:

$$\frac{\partial J}{\partial \mathbf{w}(k+1)} = 0$$

- Applicando questo vincolo alla (2) si ottiene:

$$\nabla J\left[\mathbf{w}(k)\right] \cdot \mathbf{1} + \frac{1}{2}\mathbf{1}^t \mathbf{D}\left(\mathbf{w}(k+1) - \mathbf{w}(k)\right) + \frac{1}{2}\left(\mathbf{w}(k+1) - \mathbf{w}(k)\right)^t \mathbf{D} \cdot \mathbf{1} = 0$$

- Ricordando che $D^t = D$ per la simmetria dell'Hessiano, e la proprietà delle matrici che $A^t D = D^t A$, si ottiene la seguente relazione:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mathbf{D}^{-1} \nabla J\Big|_{\mathbf{w}=\mathbf{w}(k)}$$

- In pratica, in questo caso si ha $\rho_k = \mathrm{D}^{-1}$ cioè l'algoritmo di *Newton-Raphson*.
- Possono esistere alcuni problemi inerenti a queste scelte di ρ_k, in quanto, per esempio, D^{-1} può non esistere.
- Inoltre, l'inversione di matrici è computazionalmente onerosa, e va ripetuta ad ogni passo dell'algoritmo.
- Inoltre, l'assunzione di superfici di secondo grado può essere non sufficientemente precisa.
- Pertanto, spesso si preferisce scegliere ρ_k una volta per tutte e mantenerlo quindi costante.
- Una scelta differente è rappresentata da $\rho_k = 1/\mathrm{k}$. In questo modo:
  a) si aumenta velocità di convergenza;
  b) si evitano oscillazioni.

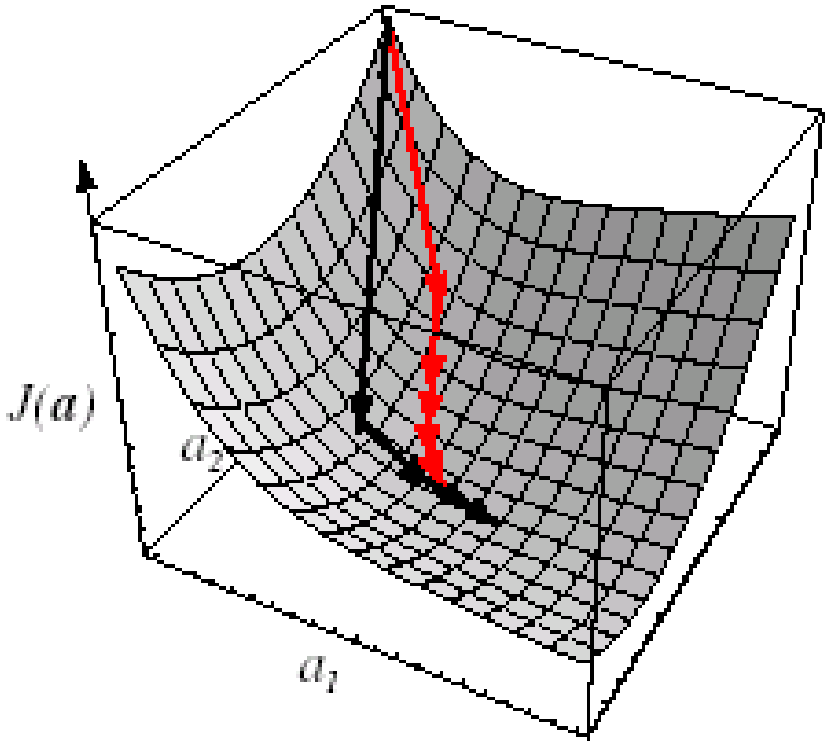


FIGURE 5.10. The sequence of weight vectors given by a simple gradient descent method (red) and by Newton's (second order) algorithm (black). Newton's method typically leads to greater improvement per step, even when using optimal learning rates for both methods. However the added computational burden of inverting the Hessian matrix used in Newton's method is not always justified, and simple gradient descent may suffice. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Metodo del Percettrone (*Perceptron*)

- Consideriamo il problema di costruire una funzione per risolvere la diseguaglianza $\mathbf{w}^t\mathbf{x}_i > 0$.
- La scelta più ovvia è quella di definire un funzionale $J(\mathbf{w};\mathbf{x}_1,\ldots,\mathbf{x}_n)$ rappresentato dal numero di campioni mal classificati da $\mathbf{w}$.
- Siccome tale funzionale è costante a tratti, il metodo della discesa del gradiente non è molto adatto a tale problema.
- Una scelta migliore per J può essere perciò:

$$J(\mathbf{w}) = -\sum_{i \in X} \mathbf{w}^t \mathbf{x}_i \tag{5}$$

dove X è l'insieme di punti classificati non correttamente da $\mathbf{w}$.

- Geometricamente, $J(\mathbf{w})$ è proporzionale alla somma delle distanze dei campioni mal classificati dal confine di decisione.
- Converge solo se esiste una separazione lineare tra le 2 classi: l'ipotesi è molto forte.
- Siccome la i-esima componente del gradiente di $J(\mathbf{w})$ è pari a $\partial J/\partial \mathrm{w_i}$, si può osservare dall'eq. (5) che:

$$\nabla J = -\sum_{i \in X} \mathbf{x}_i$$

- e quindi l'algoritmo di discesa del gradiente è

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \rho_k \cdot \sum_{i \in X} \mathbf{x}_i, \quad \text{con} \quad \rho_k = \frac{1}{k}$$

- Si noti che $\sum_{i \in X} \mathbf{x}_i$ non dipende da $\mathbf{w}$
  ed è un fattore correttivo molto brusco e quindi prima di arrivare alla convergenza ci possono essere delle forti oscillazioni.

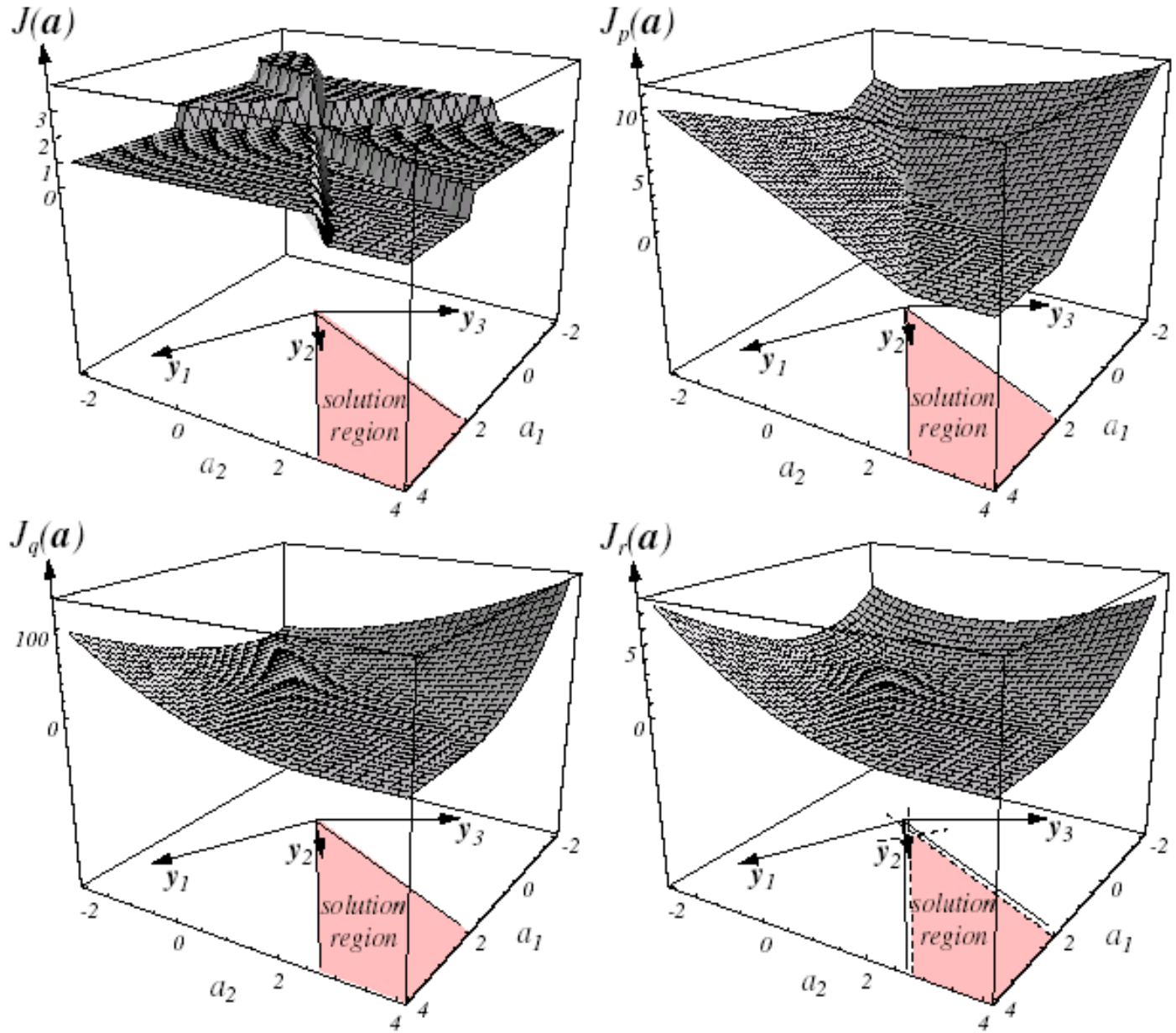
- Questo metodo è stato usato per simulare sia in modo *hardware* che *software* la prima rete neuronale ed si è in seguito evoluto a più complesse versioni come, ad esempio il *perceptron* multilivello (vedi lezione sulle reti neurali)

Metodo del rilassamento

- La funzione $J(\mathbf{w})$ assume la forma

$$J_q(\mathbf{w}) = \sum_{i \in X} \left(\mathbf{w}^t \mathbf{x}_i\right)^2$$

- Anche questa funzione considera i campioni mal classificati ed è minimizzata quando $\mathbf{w}$ è il vettore soluzione.
- A differenza di prima, il gradiente di J_q è continuo, mentre quello della funzione relativa al perceptron non lo è.
- Quindi tale funzione è adatta per trovare superfici dolci (*smooth*), ma questo può portare dei problemi.
- Infatti, in questo modo i punti molto lontani dall'origine pesano molto.
- Un altro problema è che, sperimentalmente, si nota che spesso la soluzione diventa quella banale $\mathbf{w} = 0$.


J(a)
3
2
1
0
y1
y2
y3
solution region
-2
0
2
4
a2
a1
Jp(a)
10
5
0
Jq(a)
100
0
Jr(a)
5
0

- Un'altro tipo di $J(\mathbf{w})$ può quindi essere:

$$J(\mathbf{w}) = \frac{1}{2} \sum_{i \in X} \left\{ \frac{\left(\mathbf{w}^t \mathbf{x}_i - b\right)^2}{\left\|\mathbf{x}_i\right\|^2} \right\}$$

  dove $b \geq 0$ margine (p.es., $b = 1$) e $\mathbf{w}^t \mathbf{x}_i \leq b$.

- Tutti i 3 metodi sopra citati hanno la necessità della separabilità lineare.
- In quest'ultimo caso allora,

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \rho_k \sum_{i \in X} \left\{ \frac{\left(\mathbf{w}^t \mathbf{x}_i - b\right)\mathbf{x}_i}{\left\|\mathbf{x}_i\right\|^2} \right\}$$

  con $\mathbf{x}_i$ che verificano $\mathbf{w}^t \mathbf{x}_i \leq b$ e $\mathbf{w}$ viene calcolato con la solita regola.

- $J(\mathbf{w}) > 0$ sempre e $J(\mathbf{w}) = 0$ sse $\mathbf{w}^t \mathbf{x}_i \geq b \quad \forall \mathbf{x}_i$

Metodo del MSE (*Minimum Square Error*)

- Le funzioni finora discusse consideravano i campioni mal classificati.
- Si assuma ora una funzione che consideri tutti i campioni.
- L'obiettivo è quello di cercare di verificare $\mathbf{w}^t \mathbf{x}_i = b_i$, dove b_i è una costante positiva arbitraria.
- Il problema è stato spostato dalla ricerca della soluzione per un insieme di diseguaglianze lineari al problema della ricerca della soluzione per un insieme di equazioni lineari.

- Si supponga di avere un certo *training-set*, ovvero:
  - 1º campione: $x_{1,1}w_1 + x_{1,2}w_2 + \ldots + x_{1,n}w_n + x_{1,n+1}w_{n+1} = b_1, \quad b_1 \geq 0$
  - 2º campione: $x_{2,1}w_1 + x_{2,2}w_2 + \ldots + x_{2,n}w_n + x_{2,n+1}w_{n+1} = b_2, \quad b_2 \geq 0$
  - …….
  - N-esimo campione: $x_{N,1}w_1 + x_{N,2}w_2 + \ldots + x_{N,n}w_n + x_{N,n+1}w_{n+1} = b_N, \quad b_N \geq 0$

dove w_{n+1} equivale al w_0 visto precedentemente e il vettore aggiunto sarà (cambiamento di segno per ω_2):

$x_{i,n+1} = 1$ per ω_1 e $x_{i,n+1} = -1$ per ω_2

- Si ottengono quindi N equazioni (equivalenti ai campioni di training, N_1 per ω_1 e N_2 per ω_2), con $N = N_1+N_2$ ipotesi di partenza.

- Quindi, in termini matriciali avrò:

$$\mathbf{X} \cdot \mathbf{w} = \mathbf{b}$$

dove

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} & \pm 1 \\ x_{2,1} & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \vdots & \vdots \\ x_{N,1} & \cdots & \cdots & x_{N,n} & \pm 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_n \\ w_{n+1} \end{bmatrix}$$

- In generale $N >> n+1$ e quindi ci sono più equazioni che incognite. In questo caso, tipicamente, non è possibile ricavare una soluzione esatta ($\mathbf{w} = \mathbf{X}^{-1}\,\mathbf{b}$ non ha significato).
- Chiamo ora **e** il vettore errore così definito

  $\mathbf{e} = \mathbf{X}\,\mathbf{w} - \mathbf{b}$
- Pongo inoltre (ottenendo l'errore quadratico)

  $J_M(\mathbf{w}) = (\mathbf{X}\,\mathbf{w} - \mathbf{b})^t\,(\mathbf{X}\,\mathbf{w} - \mathbf{b})$
- Potrei applicare ora uno dei metodi prima visti per calcolare w_{k+1}.
- Quello basato sul calcolo dell'Hessiana è troppo oneroso, quindi continuo ad applicare il primo (eq. (1)) anche se converge più lentamente.

- Voglio in pratica che J_M abbia un minimo rispetto a $\mathbf{w}$, quindi calcolo il gradiente:

$$\nabla_w J_M(\mathbf{w}) = 2\,\mathbf{X}^t\,(\mathbf{X}\,\mathbf{w} - \mathbf{b}) = 2\,(\mathbf{X}\,\mathbf{w} - \mathbf{b})^t\,\mathbf{X} = 0$$

$$\Rightarrow \mathbf{X}^t\,\mathbf{X}\,\mathbf{w} = \mathbf{X}^t\,\mathbf{b}$$

- Se moltiplico per $[\mathbf{X}^t\,\mathbf{X}]^{-1}$, ottengo

$$\mathbf{w} = [\mathbf{X}^t\,\mathbf{X}]^{-1}\,\mathbf{X}^t\,\mathbf{b} = \mathbf{X}^{\#}\,\mathbf{b}$$

  dove $\mathbf{X}^{\#} = [\mathbf{X}^t\,\mathbf{X}]^{-1}\,\mathbf{X}^t$ è la pseudoinversa di $\mathbf{X}$.

- Allora $\mathbf{w} = \mathbf{X}^{\#}\,\mathbf{b}$ è la soluzione ottima secondo il metodo MSE.
- $\mathbf{X}^{\#}$ è però difficile da ottenere (devo invertire una matrice n+1×n+1).
- Per contro, non ho nessun processo iterativo, ossia ottengo la soluzione $\mathbf{w}$ in modo analitico.

- La soluzione MSE dipende dal vettore margine $\mathbf{b}$ dato che differenti scelte di $\mathbf{b}$ conducono a differenti proprietà della soluzione.
- Se $\mathbf{b}$ è fissato arbitrariamente, non esiste dimostrazione che la soluzione MSE dia un vettore separatore nel caso di separabilità lineare.
- È tuttavia ragionevole pensare che, minimizzando la funzione quadratica, si possa ottenere una utile funzione discriminante sia nel caso con separabilità lineare che non lineare.
- Notare che con una scelta oculata di $\mathbf{b}$, la funzione discriminante MSE è relativa al discriminante lineare di Fisher

Metodo del *Least* MSE (LMSE) o Widrow-Hoff

- Come specificato in precedenza

  $J_M(\mathbf{w}) = (\mathbf{X}\,\mathbf{w} - \mathbf{b})^t\,(\mathbf{X}\,\mathbf{w} - \mathbf{b}) = \|\,\mathbf{X}\,\mathbf{w} - \mathbf{b}\,\|^2$

  può essere minimizzata attraverso una procedura di discesa del gradiente.

- Tale approccio ha due vantaggi rispetto al metodo della pseudoinversa:
  1) evita i problemi che sorgono quando $\mathbf{X}^t\mathbf{X}$ è singolare,
  2) evita di trattare matrici di grosse dimensioni.

- Inoltre, il calcolo richiesto è in pratica uno schema iterativo con retroazione (*feedback*) che deve far fronte automaticamente con problemi computazionali dovuti all'arrotondamento e troncamento.

- Poichè $\nabla J_M(\mathbf{w}) = 2\,\mathbf{X}^t\,(\mathbf{X}\,\mathbf{w} - \mathbf{b})$, l'algoritmo di discesa del gradiente è

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \rho_k\,\mathbf{X}^t\,(\mathbf{X}\mathbf{w}_k - \mathbf{b}) \quad \text{con } \mathbf{w}_1 \text{ arbitrario,}$$

- È facile dimostrare che se $\rho_k = \rho_1/k$ (dove ρ_1 è una costante positiva arbitraria), allora l'eq. sopra genera una sequenza di vettori di peso che converge ad un vettore limite dato da

$$\mathbf{X}^t\,(\mathbf{X}\,\mathbf{w} - \mathbf{b}) = 0$$

- In tal modo, l'algoritmo di discesa del gradiente conduce alla soluzione senza considerare se $\mathbf{X}^t\mathbf{X}$ sia singolare o meno.

- Utilizzando quindi questo approccio e considerando un campione per volta, ottengo

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \rho_k (b_k - \mathbf{w}_k^t \mathbf{x}_k) \mathbf{x}_k$$

che è chiamato metodo *LMSE* (o *Widrow-Hoff*).

- b_k è il valore per il campione k-esimo del margine $\mathbf{b}$ ($b_k \geq 0$), ovvero dipende dal campione $\mathbf{x}_k$.

- Questo algoritmo di discesa appare simile a quello di rilassamento: la differenza primaria tra i due è la regola di correzione di errore, tale che $\mathbf{w}_k^t \mathbf{x}_k$ è sempre minore di b_k, mentre la regola di Widrow-Hoff "corregge" $\mathbf{w}_k$ ogniqualvolta $\mathbf{w}_k^t \mathbf{x}_k \neq b_k$.

- In molti casi, è impossibile soddisfare tutte le equazioni $\mathbf{w}_k^t \mathbf{x}_k = b_k$ e quindi la correzione non cessa mai.
- In tal modo, ρ_k deve decrescere con k per ottenere la convergenza del metodo.
- Questa è una delle tecniche più usate per i classificatori lineari.
- Lo svantaggio è che devo iterare almeno 3-4 volte su tutti i campioni cioè $K = 3\text{-}4N$.
- Ad ogni iterazione prendo infatti un campione diverso, poichè k è l'indice dell'iterazione ma anche dei campioni (i.e., $\mathbf{x}_k$).

Esempio (Widrow-Hoff): LMSE deterministico

- Siano date le due classi seguenti:

  ω_1 : (0,0,0,1) (1,0,0,1) (1,0,1,1) (1,1,0,1)

  ω_2 : (0,0,1,1) (0,1,0,1) (0,1,1,1) (1,1,1,1)

  e si utilizzi la formula ricorsiva:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \rho_k \left(\mathbf{z}_k - \mathbf{w}_k^t \mathbf{x}_k \right) \mathbf{x}_k$$

- Sono già con il vettore aggiunto, ma non ho cambiato segno ad ω_2 (però ho messo $\mathbf{z}_k$ e non $\mathbf{b}_k$).

- All'inizio quindi si avrà:

  $\mathbf{w}_1 = \mathbf{0}$, $\rho_k = 1/k$, $\mathbf{z}_k = \pm \mathbf{1}$

- Quindi:

$$\mathbf{w}_2 = \mathbf{w}_1 + \rho_1\left(1 - \mathbf{w}_1^t\mathbf{x}_1\right)\mathbf{x}_1 = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

$$\mathbf{w}_3 = \mathbf{w}_2 + \rho_2\left(1 - \mathbf{w}_2^t\mathbf{x}_2\right)\mathbf{x}_2 = \begin{pmatrix}0\\0\\0\\1\end{pmatrix} + \frac{1}{2}\mathbf{x}_2(1-1) = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

$$\vdots$$

- Il fatto che $\mathbf{w}_3 = \mathbf{w}_2$ è puramente casuale, e si deve comunque continuare i calcoli, "ciclando" più volte su tutti i campioni, ovvero si continua a modificare $\mathbf{w}$ finchè $(\mathbf{z}_k - \mathbf{w}_k{}^t\,\mathbf{x}_k) = 0$ per ogni campione, smorzando ad ogni iterazione mediante ρ_k.

Altri riferimenti bibliografici

- [Gonzalez, 1974]

  J.T.Tou, R.C.Gonzalez, *Pattern Recognition Principles*, Addison-Wesley Publ. Comp., Massachusetts, 1974.

- [Fukunaga, 1990]

  K.Fukunaga, *Introduction to Statistical Pattern Recognition*, Second Edition, Academic Press, Inc., Boston, 1990.

- [Duda, 1973]

  R.O.Duda, P.E.Hart, *Pattern Classification and Scene Analysis*, John Wiley and Sons, New York, 1973.