
Corso di Basi di Dati Distribuite

Algoritmi di Apprendimento avanzato per l'Information Retrieval

Alessandro Moschitti

Dipartimento di Informatica Sistemi e produzione
Università di Roma “Tor Vergata”
Email: moschitti@info.uniroma2.it

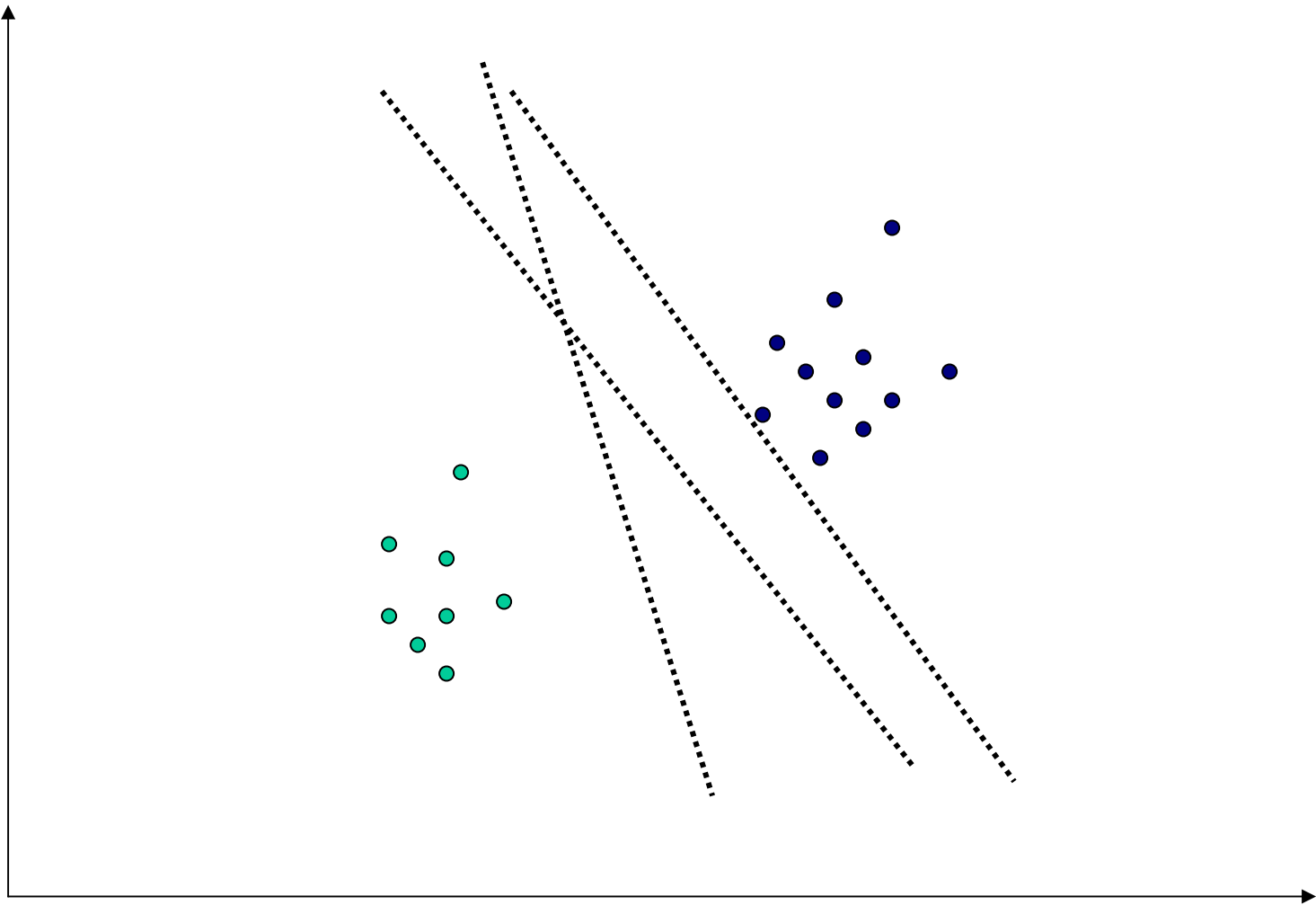


Sommario

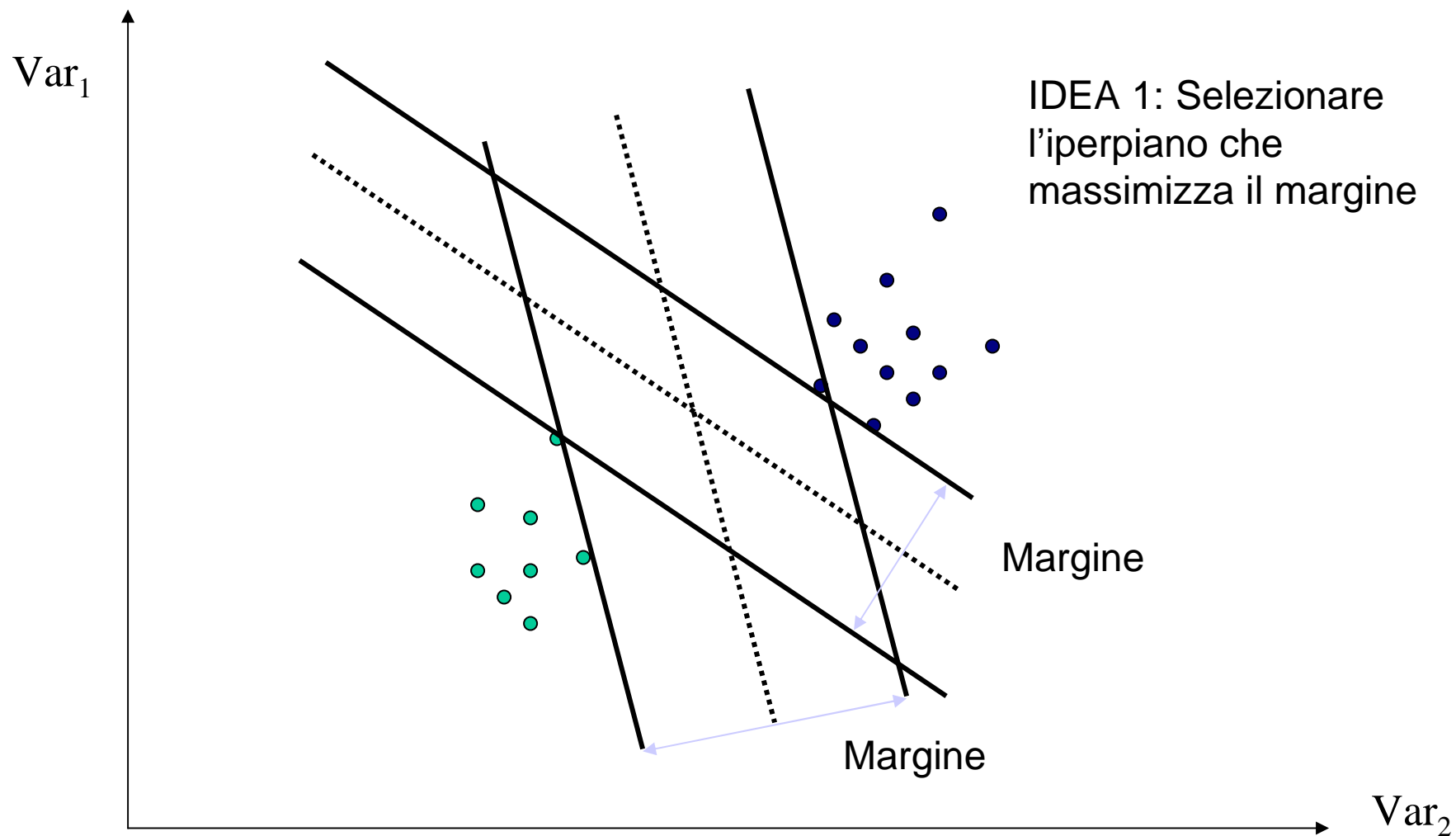
- Support Vector Machines
 - Ottimizzazione con margine *hard*
 - Ottimizzazione con margine *soft*



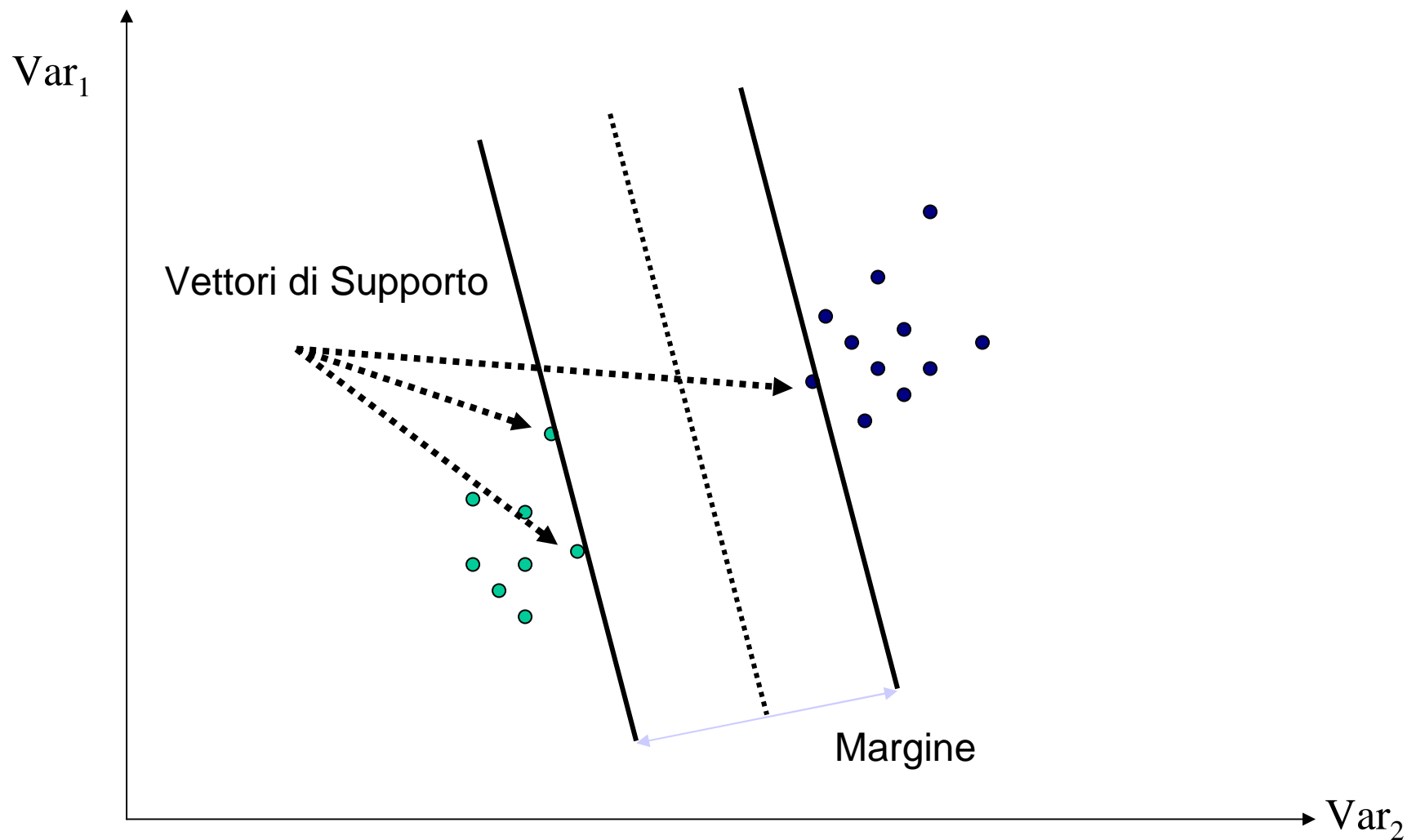
Quale iperpiano scegliere



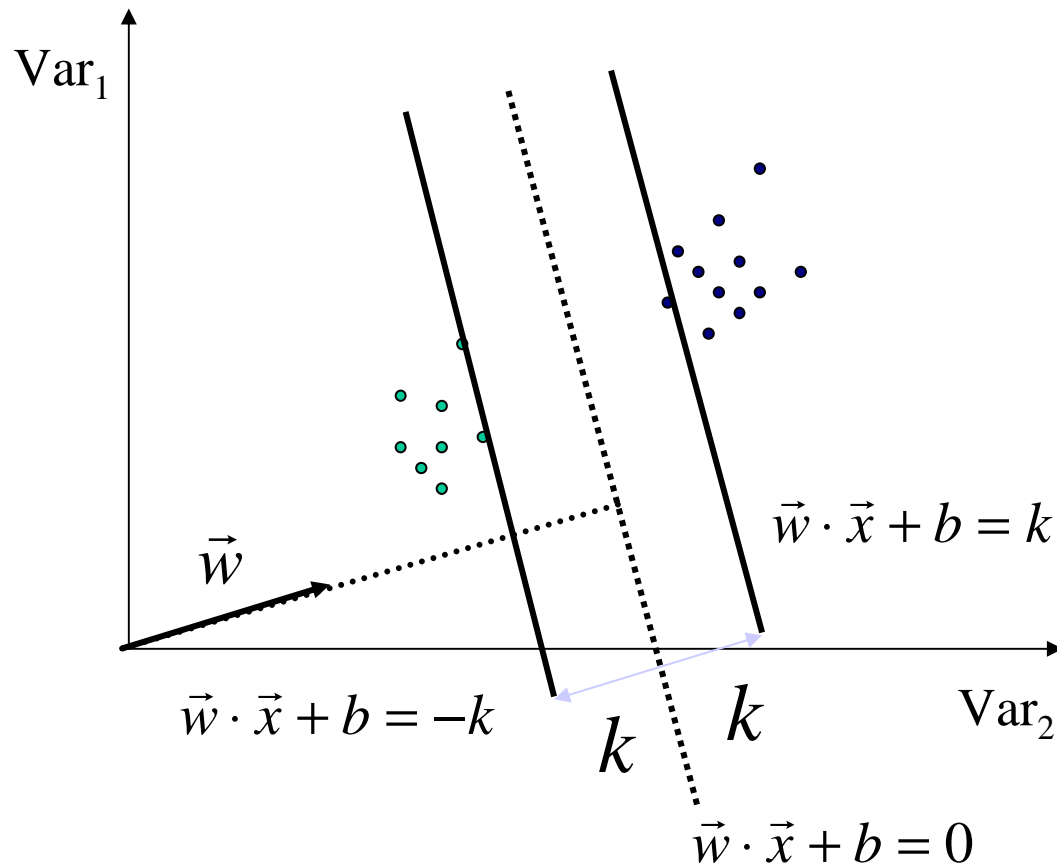
Classificatore con Massimo Margine



Vettori di supporto



Come trovare il margine massimo?



La misura del margine è:

$$\frac{2|k|}{\|\vec{w}\|}$$

Il problema da risolvere è

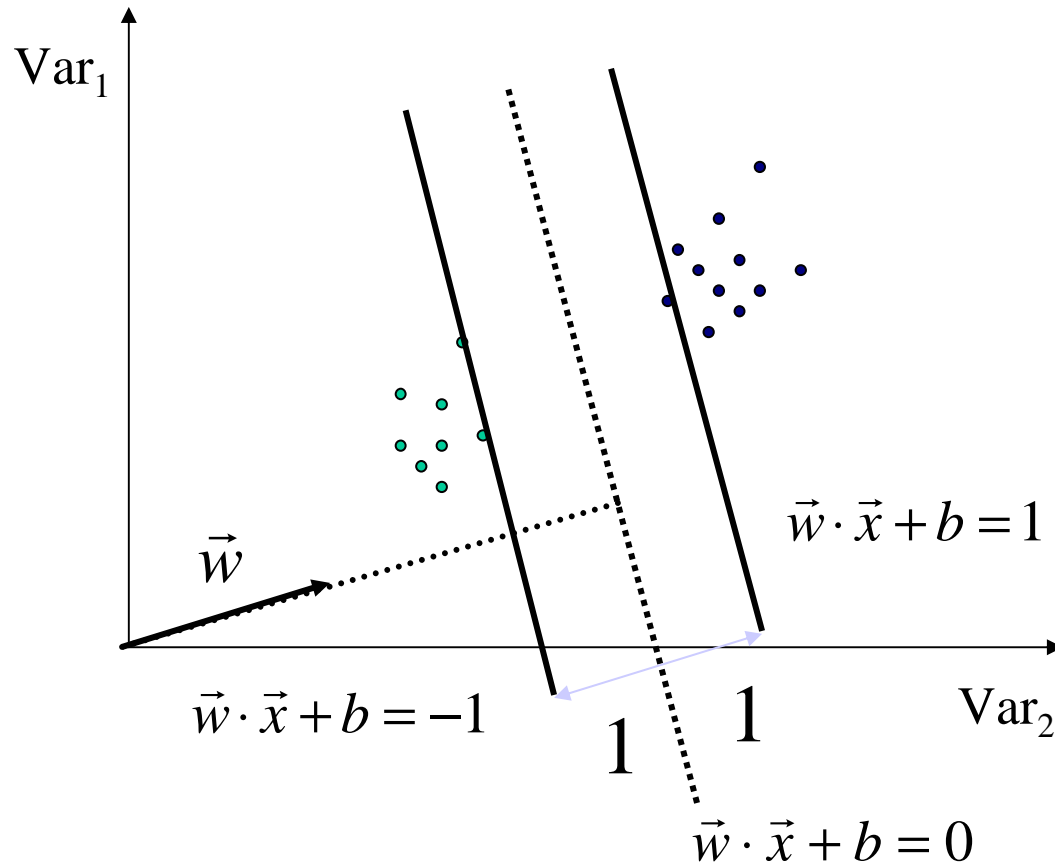
$$\text{MAX} \frac{2|k|}{\|\vec{w}\|}$$

$\vec{w} \cdot \vec{x} + b \geq +k$, se \vec{x} è positivo

$\vec{w} \cdot \vec{x} + b \leq -k$, se \vec{x} è negativo



Scalando l'iperpiano...



Ci sarà una scala tale
che $k=1$.

Il problema diventa:

$$\max \frac{2}{\|\vec{w}\|}$$

$\vec{w} \cdot \vec{x} + b \geq +1$, se \vec{x} è positivo

$\vec{w} \cdot \vec{x} + b \leq -1$, se \vec{x} è negativo



Problema nella forma finale

$$\begin{array}{l} \max \frac{2}{\|\vec{w}\|} \\ \vec{w} \cdot \vec{x}_i + b \geq +1, \quad y_i = 1 \\ \vec{w} \cdot \vec{x}_i + b \leq -1, \quad y_i = -1 \end{array} \quad \Rightarrow \quad \begin{array}{l} \max \frac{2}{\|\vec{w}\|} \\ y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1 \end{array} \quad \Rightarrow$$

$$\begin{array}{l} \min \frac{\|\vec{w}\|}{2} \\ y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1 \end{array} \quad \Rightarrow \quad \begin{array}{l} \min \frac{\|\vec{w}\|^2}{2} \\ y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1 \end{array}$$



Definizione di errore sul training

- Dati di Training

$$f : R^N \rightarrow \{\pm 1\} \quad (\vec{x}_1, y_1), \dots, (\vec{x}_l, y_l) \in R^N \times \{\pm 1\}$$

- Rischio (errore) empirico

$$R_{emp}[f] = \frac{1}{l} \sum_{i=1}^l \frac{1}{2} |f(\vec{x}_i) - y_i|$$

- Rischio (errore)

$$R[f] = \int \frac{1}{2} |f(\vec{x}) - y| dP(\vec{x}, y)$$



Caratterizzazione dell'errore (parte 1)

- Dalla teoria del PAC-learning si ha (*Vapnik*):

$$R(\alpha) \leq R_{emp}(\alpha) + \varphi\left(\frac{d}{\ell}, \frac{\log(\delta)}{\ell}\right)$$

$$\varphi\left(\frac{d}{\ell}, \frac{\log(\delta)}{\ell}\right) = \sqrt{\frac{d(\log \frac{2\ell}{d} + 1) - \log(\frac{\delta}{4})}{\ell}}$$

dove d è la VC-dimension, ℓ è il numero di esempi di training, δ è il bound alla probabilità di avere tale errore e α è un parametro del classificatore.



Molte altre versioni di tale teorema

Theorem 2.11 (*Vapnik and Chervonenkis, [Vapnik, 1995]*)

Let H be a hypothesis space having VC dimension d . For any probability distribution D on $X \times \{-1, 1\}$, with probability $1 - \delta$ over m random examples S , any hypothesis $h \in H$ that is consistent with S has error no more than

$$\text{error}(h) \leq \epsilon(m, H, \delta) = \frac{2}{m} \left(d \times \ln \frac{2e \times m}{d} + \ln \frac{2}{\delta} \right),$$

provided that $d \leq m$ and $m \geq 2/\epsilon$.



Caratterizzazione dell'errore (parte 2)

Teorema 2 (Vapnik): Si consideri la classe delle ipotesi degli iperpiani $\text{sgn}(\vec{w} \cdot \vec{x} + b)$ dove $y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1$, $\|\vec{w}\| \leq A$ ed i vettori \vec{x} sono tutti contenuti in una sfera di raggio R allora la VC-dimension d della classe di ipotesi è minore di $\text{Min}(A^2 R^2, n) + 1$



Problema da ottimizzare

- L'iperpiano ottimo:
 - Minimizzare $\tau(\vec{w}) = \frac{1}{2} \|\vec{w}\|^2$
 - Soggetto a $y_i ((\vec{w} \cdot \vec{x}_i) + b) \geq 1, i = 1, \dots, l$
- Il duale è più semplice da trattare



Definizione del Lagrangiano

Def. 2.24 Let $f(\vec{w})$, $h_i(\vec{w})$ and $g_i(\vec{w})$ be the objective function, the equality constraints and the inequality constraints (i.e. \geq) of an optimization problem, and let $L(\vec{w}, \vec{\alpha}, \vec{\beta})$ be its Lagrangian, defined as follows:

$$L(\vec{w}, \vec{\alpha}, \vec{\beta}) = f(\vec{w}) + \sum_{i=1}^m \alpha_i g_i(\vec{w}) + \sum_{i=1}^l \beta_i h_i(\vec{w})$$



Problema di ottimizzazione Duale

The Lagrangian dual problem of the above primal problem is

$$\text{maximize } \theta(\vec{\alpha}, \vec{\beta})$$

$$\text{subject to } \vec{\alpha} \geq \vec{0}$$

$$\text{where } \theta(\vec{\alpha}, \vec{\beta}) = \inf_{w \in W} L(\vec{w}, \vec{\alpha}, \vec{\beta})$$



Trasformazione nel duale

- Dato il Lagrangiano del nostro problema

$$L(\vec{w}, b, \vec{\alpha}) = \frac{1}{2} \vec{w} \cdot \vec{w} - \sum_{i=1}^m \alpha_i [y_i (\vec{w} \cdot \vec{x}_i + b) - 1]$$

- Per risolvere il duale dobbiamo calcolare

$$\theta(\vec{\alpha}, \vec{\beta}) = \inf_{w \in W} L(\vec{w}, \vec{\alpha}, \vec{\beta})$$

- Imponiamo le derivate = 0, rispetto a \vec{w}

$$\frac{\partial L(\vec{w}, b, \vec{\alpha})}{\partial \vec{w}} = \vec{w} - \sum_{i=1}^m y_i \alpha_i \vec{x}_i = \vec{0} \quad \Rightarrow \quad \vec{w} = \sum_{i=1}^m y_i \alpha_i \vec{x}_i$$



Trasformazione duale (cont.)

- e rispetto a b

$$\frac{\partial L(\vec{w}, b, \vec{\alpha})}{\partial b} = \sum_{i=1}^m y_i \alpha_i = 0$$

- Sostituiamo nella funzione obiettivo

$$\begin{aligned} L(\vec{w}, b, \vec{\alpha}) &= \frac{1}{2} \vec{w} \cdot \vec{w} - \sum_{i=1}^m \alpha_i [y_i (\vec{w} \cdot \vec{x}_i + b) - 1] = \\ &= \frac{1}{2} \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j \vec{x}_i \cdot \vec{x}_j - \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j \vec{x}_i \cdot \vec{x}_j + \sum_{i=1}^m \alpha_i \\ &= \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j \vec{x}_i \cdot \vec{x}_j \end{aligned}$$



Problema (duale) finale

$$\text{maximize} \quad \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j \vec{x}_i \cdot \vec{x}_j$$

$$\text{subject to} \quad \alpha_i \geq 0, \quad i = 1, \dots, m$$

$$\sum_{i=1}^m y_i \alpha_i = 0$$



Teorema di Khun-Tucker

- Condizioni necessarie e sufficienti per avere un soluzione ottima

$$\frac{\partial L(\vec{w}^*, \vec{\alpha}^*, \vec{\beta}^*)}{\partial \vec{w}} = \vec{0}$$

$$\frac{\partial L(\vec{w}^*, \vec{\alpha}^*, \vec{\beta}^*)}{\partial \vec{\beta}} = \vec{0}$$

$$\alpha_i^* g_i(\vec{w}^*) = 0, \quad i = 1, \dots, m$$

$$g_i(\vec{w}^*) \leq 0, \quad i = 1, \dots, m$$

$$\alpha_i^* \geq 0, \quad i = 1, \dots, m$$



Conseguenze dei vincoli

- Vincoli di Lagrange: $\sum_{i=1}^l a_i y_i = 0 \quad \vec{w} = \sum_{i=1}^l \alpha_i y_i \vec{x}_i$

- Vincolo di Karush-Kuhn-Tucker

$$\alpha_i \cdot [y_i (\vec{x}_i \cdot \vec{w} + b) - 1] = 0, \quad i = 1, \dots, l$$

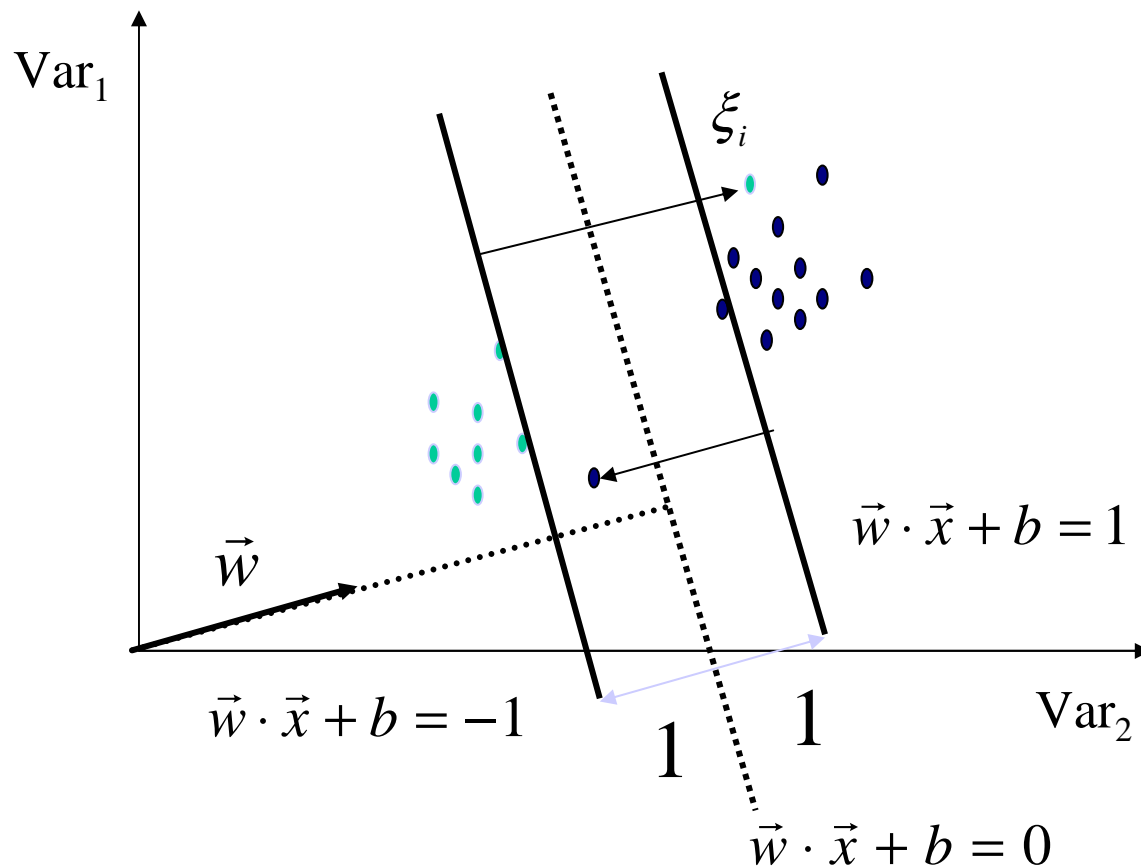
- I vettori di supporto sono quelli che hanno α_i non nullo

- Per ricavare b applichiamo la seguente formula

$$b^* = -\frac{\vec{w}^* \cdot \vec{x}^+ + \vec{w}^* \cdot \vec{x}^-}{2}$$



Dati di training non separabili linearmente

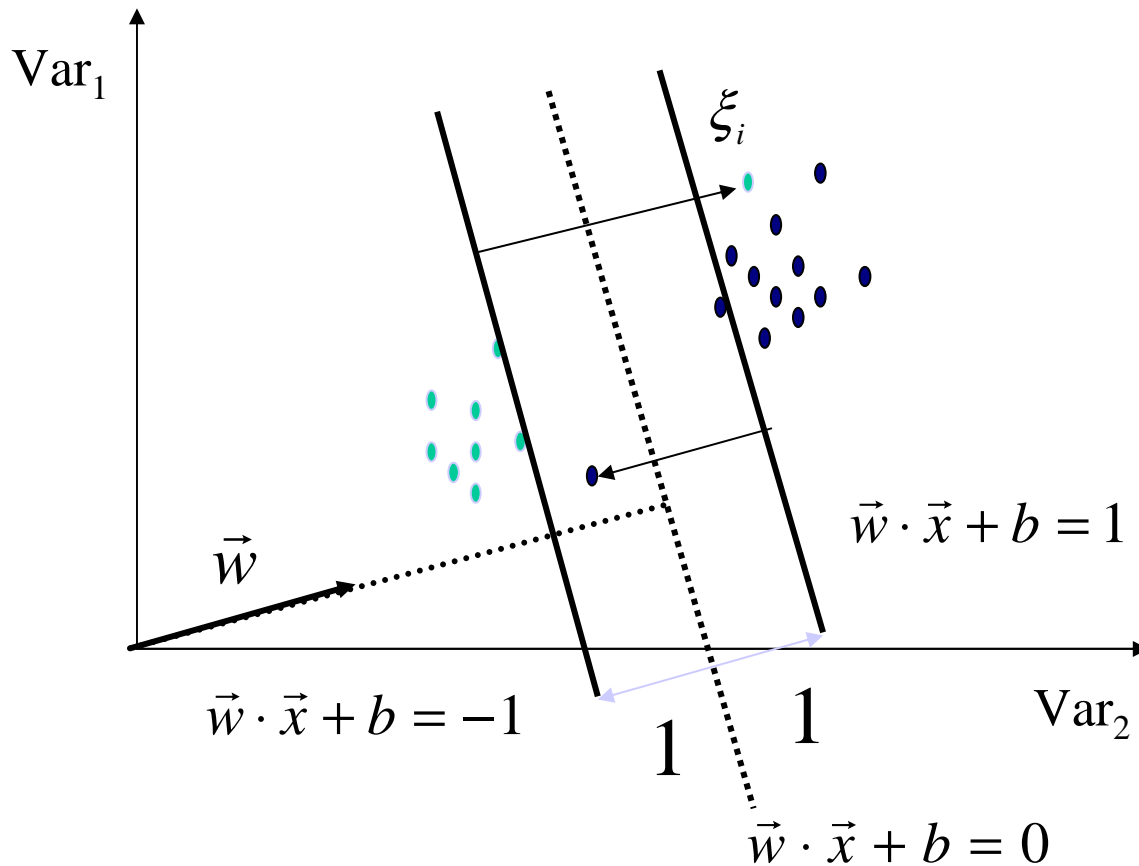


Si introducono le
variabili di slack ξ_i

Si permettono degli
errori penalizzando la
funzione di
ottimizzazione



Soft Margin SVMs



I nuovi vincoli sono :

$$y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1 - \xi_i \quad \forall \vec{x}_i$$
$$\xi_i \geq 0$$

La funzione obiettivo penalizza gli esempi incorrettamente classificati

$$\min \frac{1}{2} \|\vec{w}\|^2 + C \sum_i \xi_i$$

C è il trade-off tra margine ed errore



Conversione nel duale

$$\begin{cases} \min & \|\vec{w}\| + C \sum_{i=1}^m \xi_i^2 \\ & y_i(\vec{w} \cdot \vec{x}_i + b) \geq 1 - \xi_i, \quad \forall i = 1, \dots, m \\ & \xi_i \geq 0, \quad i = 1, \dots, m \end{cases}$$

$$L(\vec{w}, b, \vec{\xi}, \vec{\alpha}) = \frac{1}{2} \vec{w} \cdot \vec{w} + \frac{C}{2} \sum_{i=1}^m \xi_i^2 - \sum_{i=1}^m \alpha_i [y_i(\vec{w} \cdot \vec{x}_i + b) - 1]$$

- Deriviamo rispetto a \vec{w} , $\vec{\xi}$ e b



Derivate parziali

$$\frac{\partial L(\vec{w}, b, \vec{\xi}, \vec{\alpha})}{\partial \vec{w}} = \vec{w} - \sum_{i=1}^m y_i \alpha_i \vec{x}_i = \vec{0} \quad \Rightarrow \quad \vec{w} = \sum_{i=1}^m y_i \alpha_i \vec{x}_i$$

$$\frac{\partial L(\vec{w}, b, \vec{\xi}, \vec{\alpha})}{\partial \vec{\xi}} = C \vec{\xi} - \vec{\alpha} = \vec{0}$$

$$\frac{\partial L(\vec{w}, b, \vec{\xi}, \vec{\alpha})}{\partial b} = \sum_{i=1}^m y_i \alpha_i = 0$$



Sostituzione nella funzione obiettivo

$$\begin{aligned} &= \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j \vec{x}_i \cdot \vec{x}_j + \frac{1}{2C} \vec{a} \cdot \vec{a} - \frac{1}{C} \vec{a} \cdot \vec{a} = \\ &= \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j \vec{x}_i \cdot \vec{x}_j - \frac{1}{2C} \vec{a} \cdot \vec{a} = \\ &= \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j \left(\vec{x}_i \cdot \vec{x}_j + \frac{1}{C} \delta_{ij} \right), \end{aligned}$$

- δ_{ij} di Kronecker



Problema duale di ottimizzazione finale

$$\sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j \left(\vec{x}_i \cdot \vec{x}_j + \frac{1}{C} \delta_{ij} \right)$$

$$\alpha_i \geq 0, \quad \forall i = 1, \dots, m$$

$$\sum_{i=1}^m y_i \alpha_i = 0$$



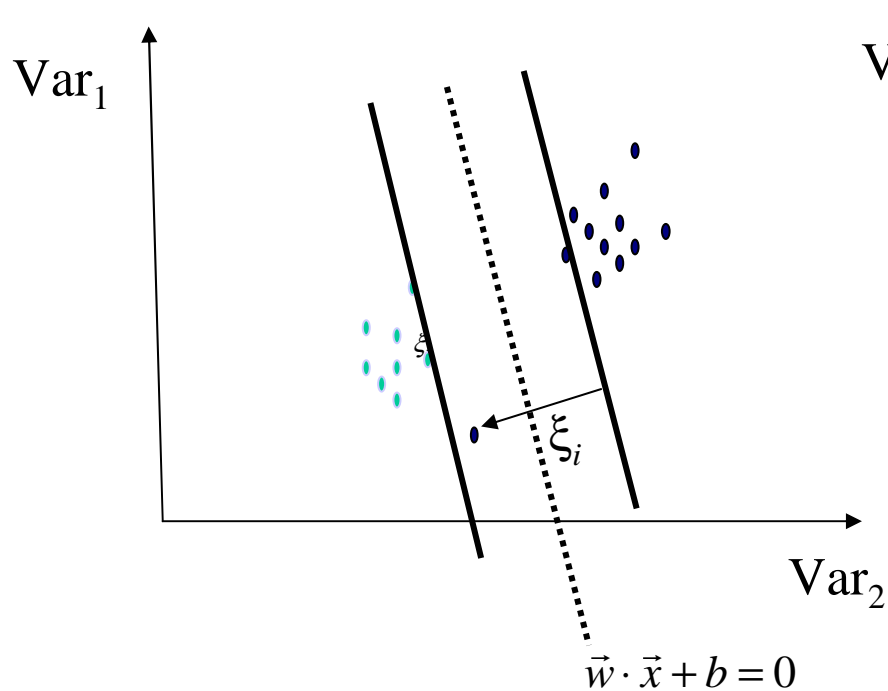
Soft Margin Support Vector Machines

$$\min \frac{1}{2} \|\vec{w}\|^2 + C \sum_i \xi_i \quad \begin{array}{l} y_i (\vec{w} \cdot \vec{x}_i + b) \geq 1 - \xi_i \quad \forall \vec{x}_i \\ \xi_i \geq 0 \end{array}$$

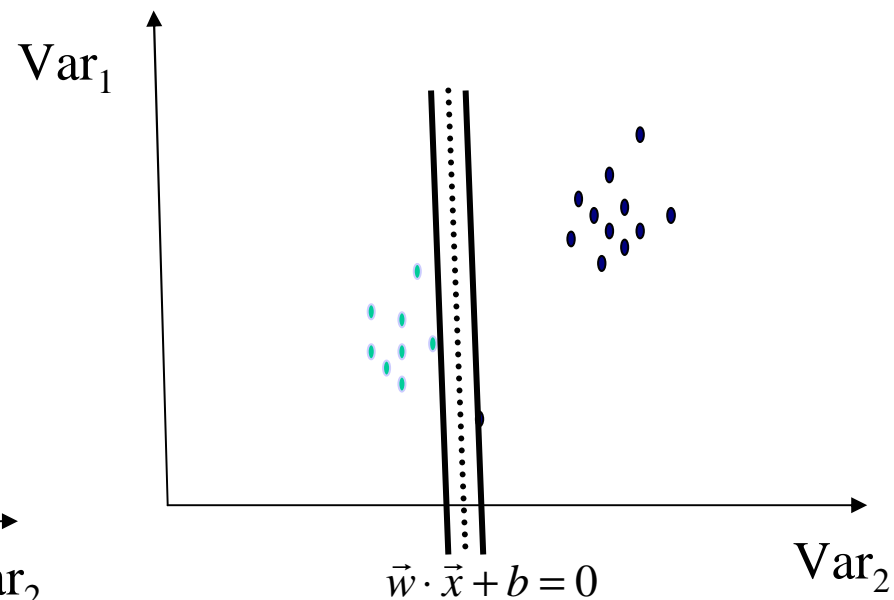
- L'algoritmo prova a mantenere a zero ξ_i e massimizzare il margine
- NB: L'algoritmo non minimizza il numero di errori (problema NP-completo); minimizza la somma delle distanze dall'iperpiano
- Se $C \rightarrow \infty$, la soluzione tende a quella del *hard-margin*
- *Attenzione !!!*: se $C = 0$ si ottiene $\|\vec{w}\| = 0$. Infatti posso sempre trovare $y_i b \geq 1 - \xi_i \quad \forall \vec{x}_i$
- Se aumento C tendo a diminuire il numero di errori. All'infinito il numero errori devono essere 0. Questo è esattamente la formulazione *hard-margin*.



Robustezza dei *Soft* vs *Hard* Margin SVMs



Soft Margin SVM



Hard Margin SVM



Soft vs Hard Margin SVMs

- *Il Soft-Margin* ha sempre una soluzione
- Il Soft-Margin è più robusto agli esempi *strani*
- *L' Hard-Margin* non richiede parametri



SVM-light: un implementazione delle SVMs

- Implementa il soft margine
- Contiene le procedure per la soluzione dei problemi di ottimizzazione
- Classificatore binario
- Esempi e descrizioni al sito:
<http://www.joachims.org/>
(<http://svmlight.joachims.org/>)



Riferimenti

- *A tutorial on Support Vector Machines for Pattern Recognition*
 - **Articolo scaricabile dalla rete**
- *The Vapnik-Chervonenkis Dimension and the Learning Capability of Neural Nets*
 - **Presentazione scaricabile dalla rete**
- Computational Learning Theory
(Sally A Goldman Washington University St. Louis Missouri)
 - **Scaricabile dalla rete**
- *AN INTRODUCTION TO SUPPORT VECTOR MACHINES*
(and other kernel-based learning methods)
N. Cristianini and J. Shawe-Taylor Cambridge University Press
 - **Da comprare**
- *The Nature of Statistical Learning Theory*
Vladimir Naumovich Vapnik - Springer Verlag (December, 1999)
 - **Da comprare**

