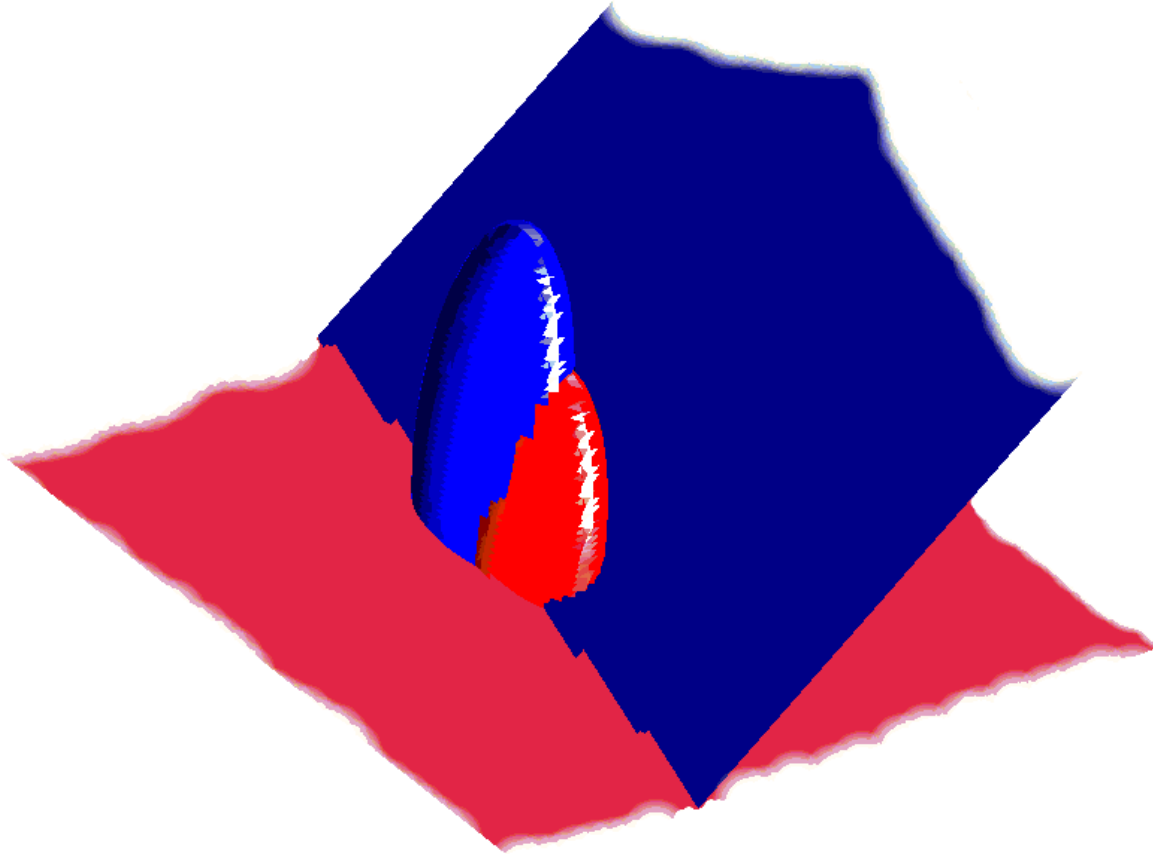


Classificare *senza sbagliare?*



G.E.M.

Un algoritmo per la costruzione di classificatori
da dati sperimentali con errore di generalizzazione garantito

*Seminario per il corso di IDENTIFICAZIONE DEI MODELLI E ANALISI DEI DATI
(30 Maggio 2011)*

Per contatti, approfondimenti, et cetera...

Algo Carè

Trasparenze: <http://www.ing.unibs.it/~algo.care/>

Mail: algo.care@ing.unibs.it

Telefono @DEA: [5935](tel:5935)

Prof. Marco C. Campi

Un algoritmo per la costruzione di classificatori
da dati sperimentali con errore di generalizzazione garantito

*Seminario per il corso di IDENTIFICAZIONE DEI MODELLI E ANALISI DEI DATI
(30 Maggio 2011)*

Per contatti, approfondimenti, et cetera...

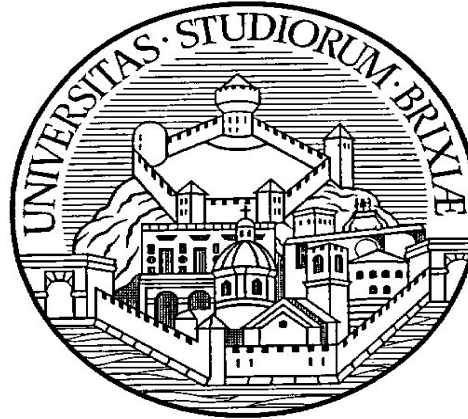
Algo Carè

Trasparenze: <http://www.ing.unibs.it/~algo.care/>

Mail: algo.care@ing.unibs.it

Telefono @DEA: [5935](tel:5935)

Prof. Marco C. Campi



Un nuovo algoritmo per la costruzione di classificatori da dati sperimentali con errore di generalizzazione garantito

Relatore:
Prof. **Marco C. Campi**

Tesi di laurea specialistica di:
Algo Carè
Matr. 60772

Marzo 2009

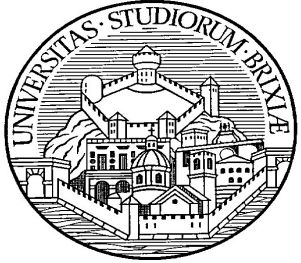
Classification with guaranteed probability of error

Marco C. Campi

Received: 1 July 2009 / Revised: 8 February 2010 / Accepted: 24 March 2010 /
Published online: 17 April 2010
© The Author(s) 2010

Abstract We introduce a general-purpose classifier that we call the *Guaranteed Error Machine*, or GEM, and two learning algorithms that are used for the training of GEM, a *real GEM algorithm* and an *ideal GEM algorithm*. The real GEM algorithm is for use in real applications, while the ideal GEM algorithm is introduced as a theoretical tool. Differently from most learning machines, GEM has a ternary-valued output, that is besides 0 and 1 it can return an *unknown* label, expressing doubt. Our central result is that, under general conditions, the statistics of the generalization error of the GEM machine obtained with the ideal GEM algorithm is *universal*, in the sense that *it remains the same, independently of the (unknown) mechanism that generates the data*. As a consequence, the user can select a desired level of generalization error and the learning machine is automatically adjusted so as to meet this desired level, and no knowledge of the data generation mechanism is required

Schema della presentazione



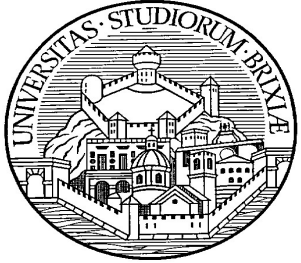
“Un **algoritmo** per la costruzione di **classificatori**
da dati sperimentali con **errore di generalizzazione *garantito***”

A. La classificazione (binaria)

B. L'algoritmo: GEM
(*Guaranteed Error Machine*)

C. Proprietà teoriche e risultati
sperimentali

Prima Parte



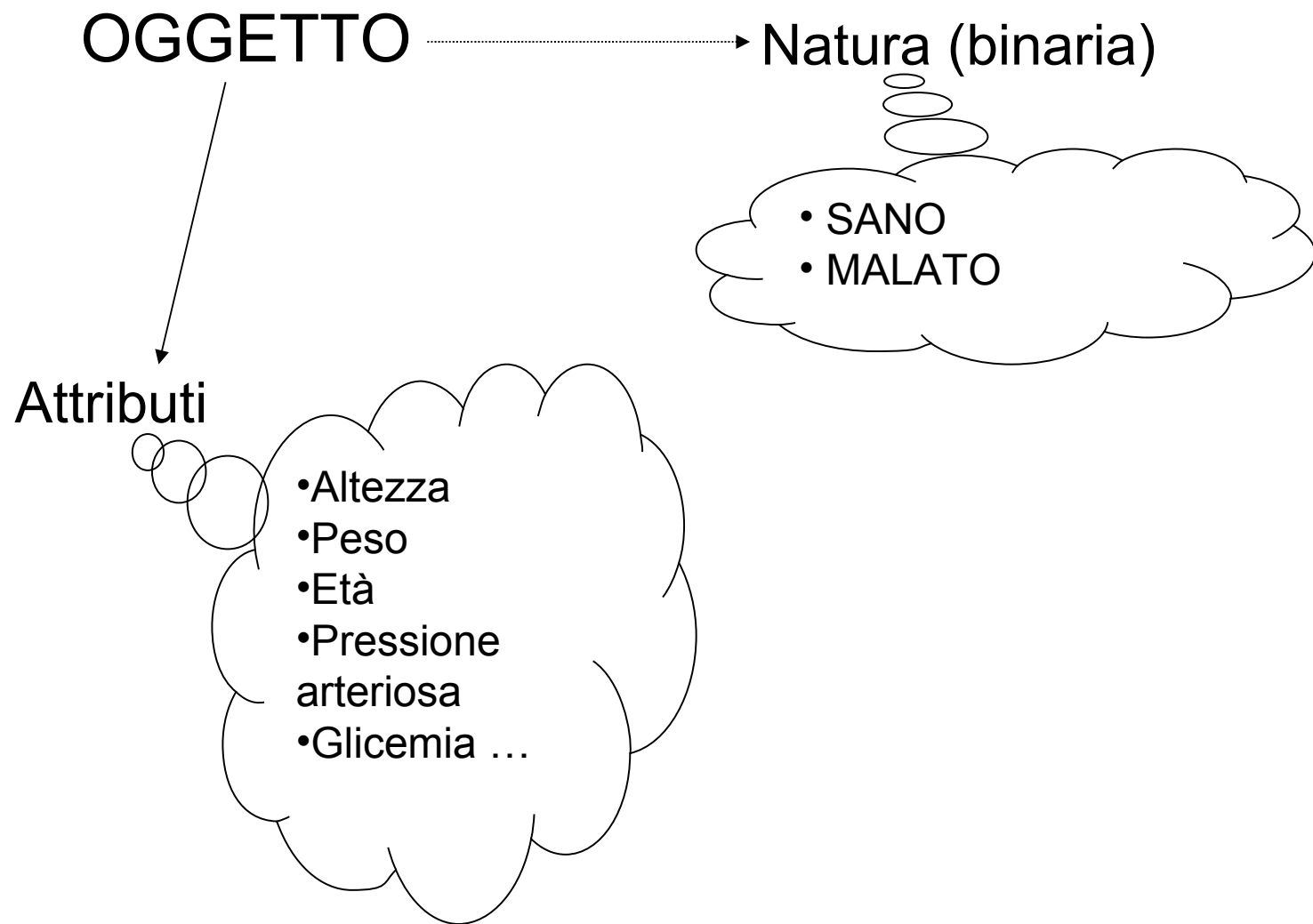
“Un **algoritmo** per la costruzione di **classificatori**
da dati sperimentali con **errore di generalizzazione *garantito***”

A. La classificazione (binaria)

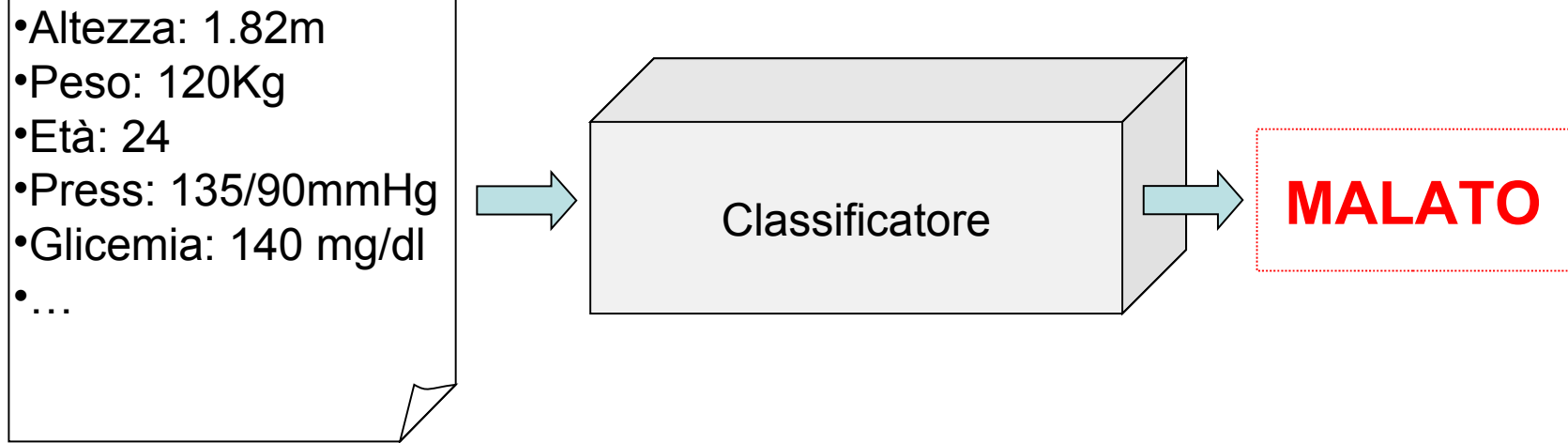
L'algoritmo: GEM
(*Guaranteed Error Machine*)

Proprietà teoriche e risultati
sperimentali

Concetti fondamentali



Un classificatore binario



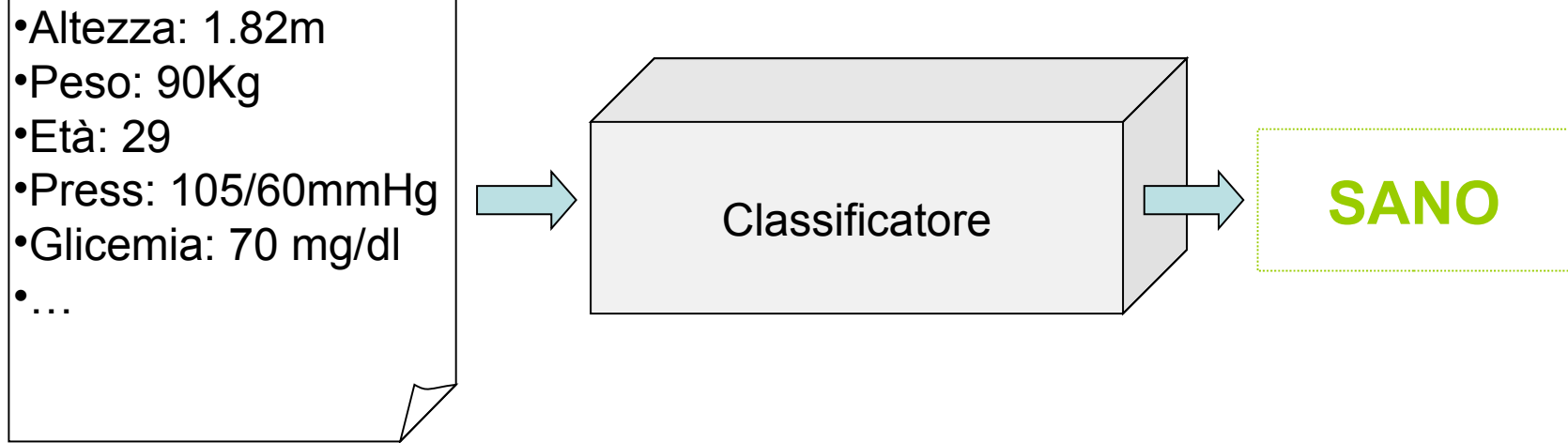
```
graph LR; Input["•Altezza: 1.82m<br/>•Peso: 120Kg<br/>•Età: 24<br/>•Press: 135/90mmHg<br/>•Glicemia: 140 mg/dl<br/>•..."] --> Classificatore[Classificatore]; Classificatore --> Output[MALATO];
```

•Altezza: 1.82m
•Peso: 120Kg
•Età: 24
•Press: 135/90mmHg
•Glicemia: 140 mg/dl
•...

Classificatore

MALATO

Un classificatore binario



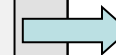
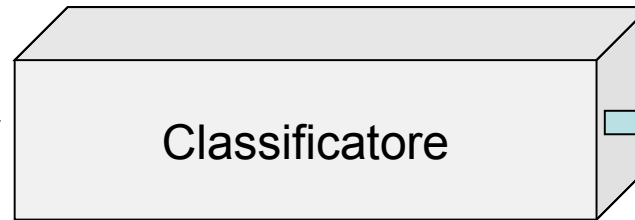
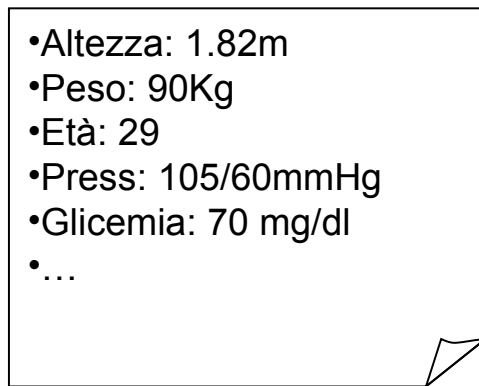
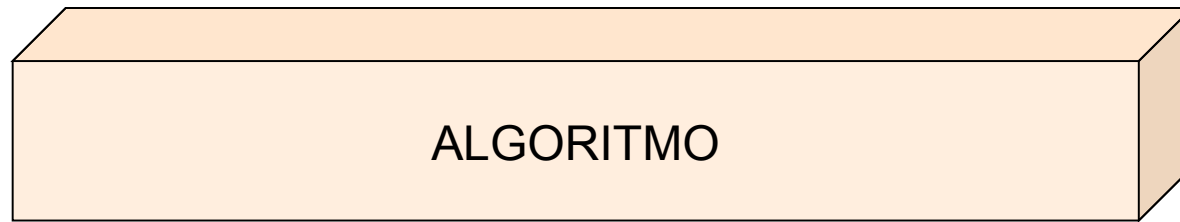
```
graph LR; Input["•Altezza: 1.82m<br/>•Peso: 90Kg<br/>•Età: 29<br/>•Press: 105/60mmHg<br/>•Glicemia: 70 mg/dl<br/>•..."] --> Classificatore[Classificatore]; Classificatore --> Output[SANO];
```

•Altezza: 1.82m
•Peso: 90Kg
•Età: 29
•Press: 105/60mmHg
•Glicemia: 70 mg/dl
•...

Classificatore

SANO

Costruire un classificatore



SANO

Valutare un classificatore

k attributi reali $\longleftrightarrow$ oggetto

reale natura degli oggetti $y(\cdot) : \mathbf{R}^k \rightarrow \{0,1\}$

classificatore binario $\hat{y}(\cdot) : \mathbf{R}^k \rightarrow \{0,1\}$

misclassificazione $\hat{y}(x) \neq y(x), x \in \mathbf{R}^k$

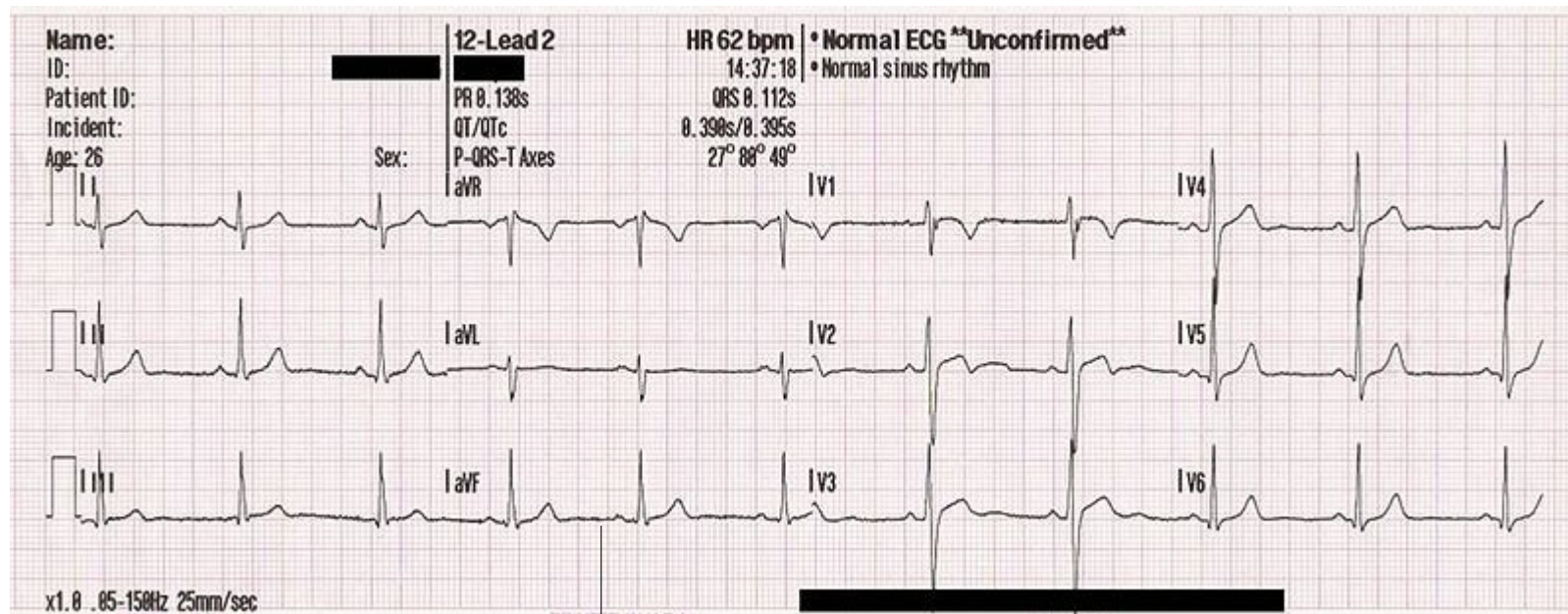
ERRORE DI GENERALIZZAZIONE

$$PE(\hat{y}) := \Pr(\hat{y}(x) \neq y(x))$$

Applicazioni

- Applicazioni medicali
- Applicazioni bancarie
- Riconoscimento di posta elettronica indesiderata
- Riconoscimento di pacchetti di rete
- ...

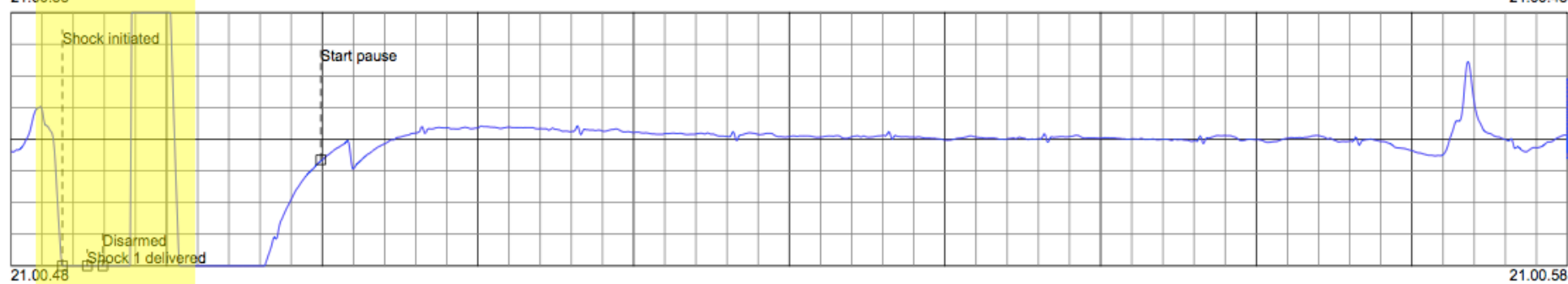
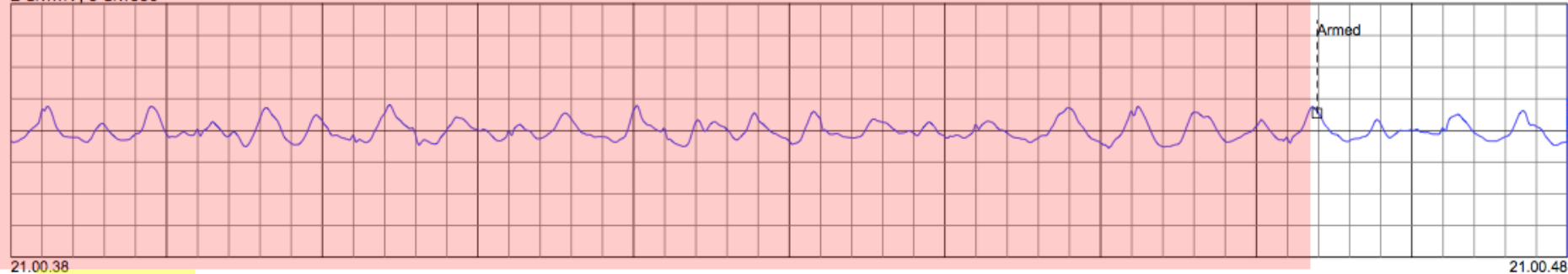
Un'applicazione “bresciana”



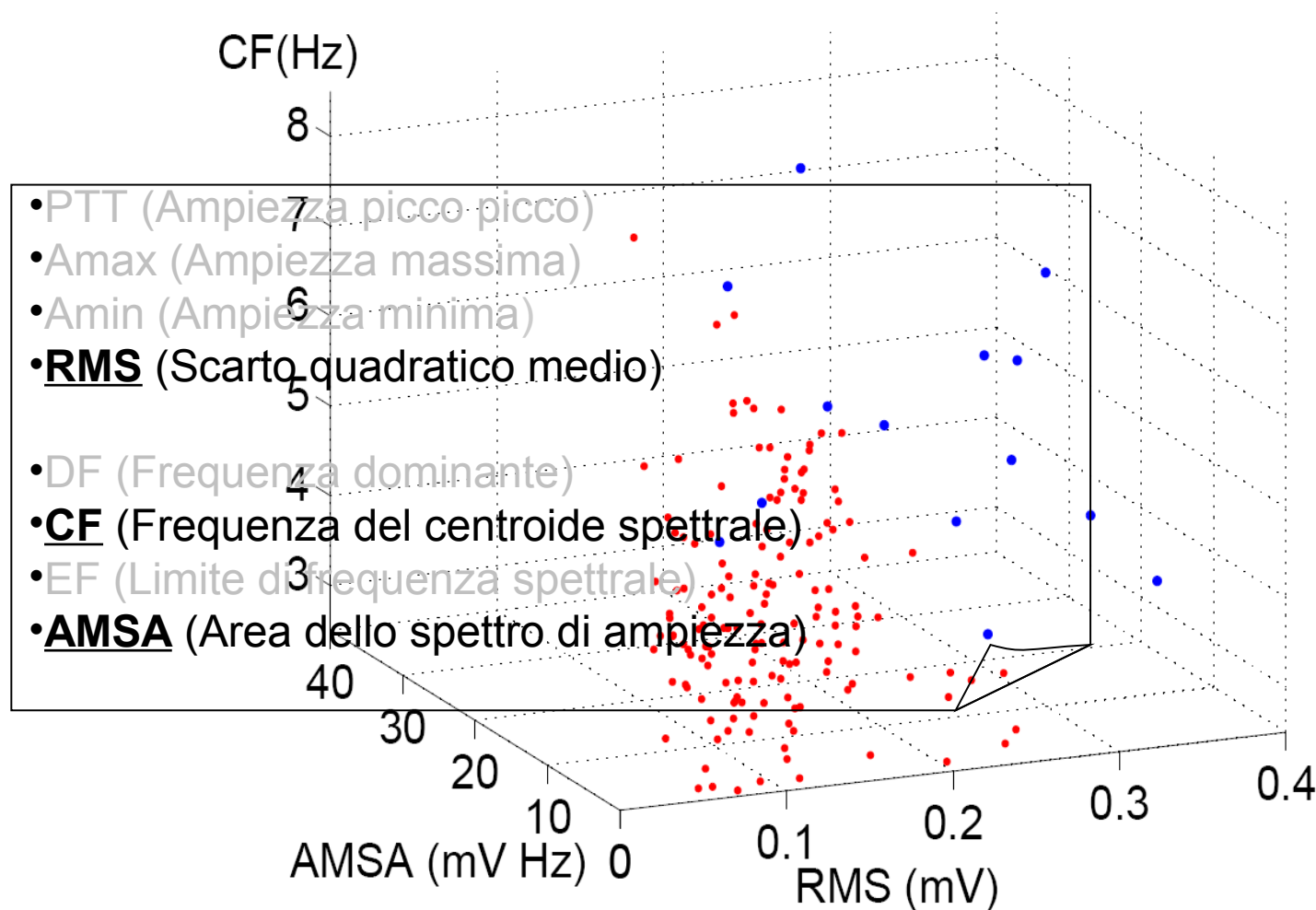
- PTT (Ampiezza picco picco)
- Amax (Ampiezza massima)
- Amin (Ampiezza minima)
- RMS (Scarto quadratico medio)
- DF (Frequenza dominante)
- CF (Frequenza del centroide spettrale)
- EF (Limite di frequenza spettrale)
- AMSA (Area dello spettro di ampiezza)

Un'applicazione “bresciana”

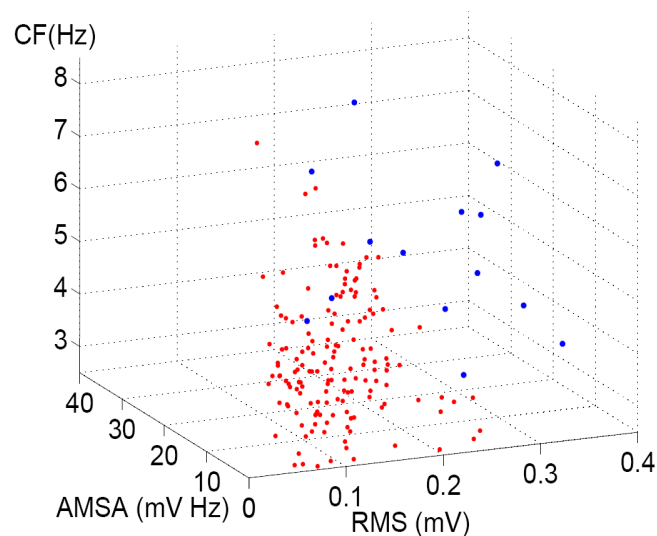
2 div/mV, 5 div/sec



Un'applicazione “bresciana”



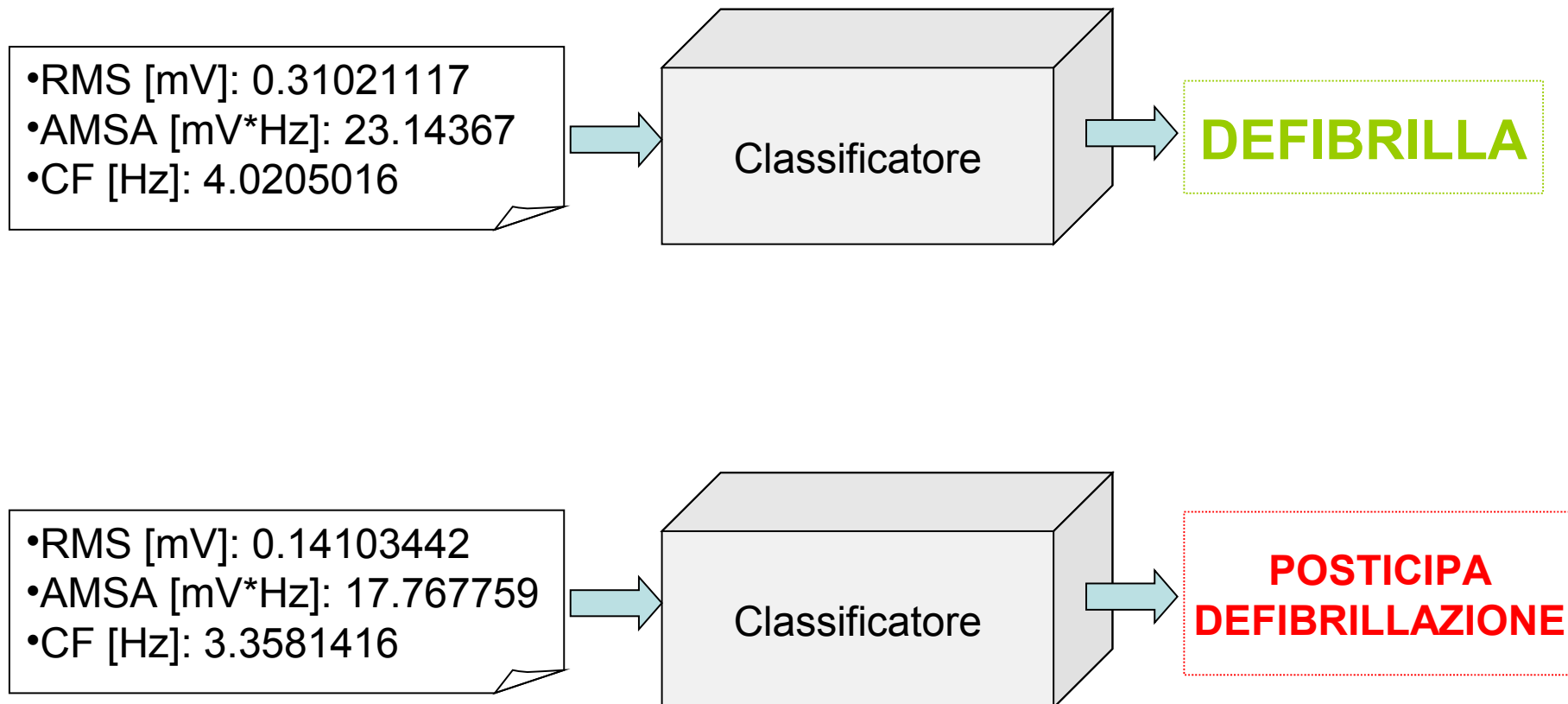
Un'applicazione “bresciana”



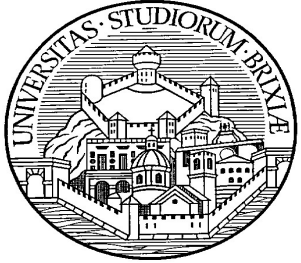
ALGORITMO

Classificatore

Un'applicazione “bresciana”



Schema della presentazione



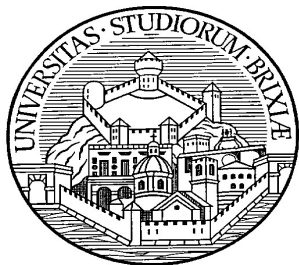
“Un **algoritmo** per la costruzione di **classificatori**
da dati sperimentali con **errore di generalizzazione *garantito***”

A. La classificazione (binaria)

B. L'algoritmo: GEM
(*Guaranteed Error Machine*)

C. Proprietà teoriche e risultati
sperimentali

Seconda Parte



“Un **algoritmo** per la costruzione di **classificatori** da dati sperimentali con **errore di generalizzazione *garantito***”

La classificazione (binaria)

B. L'algoritmo: GEM
(*Guaranteed Error Machine*)

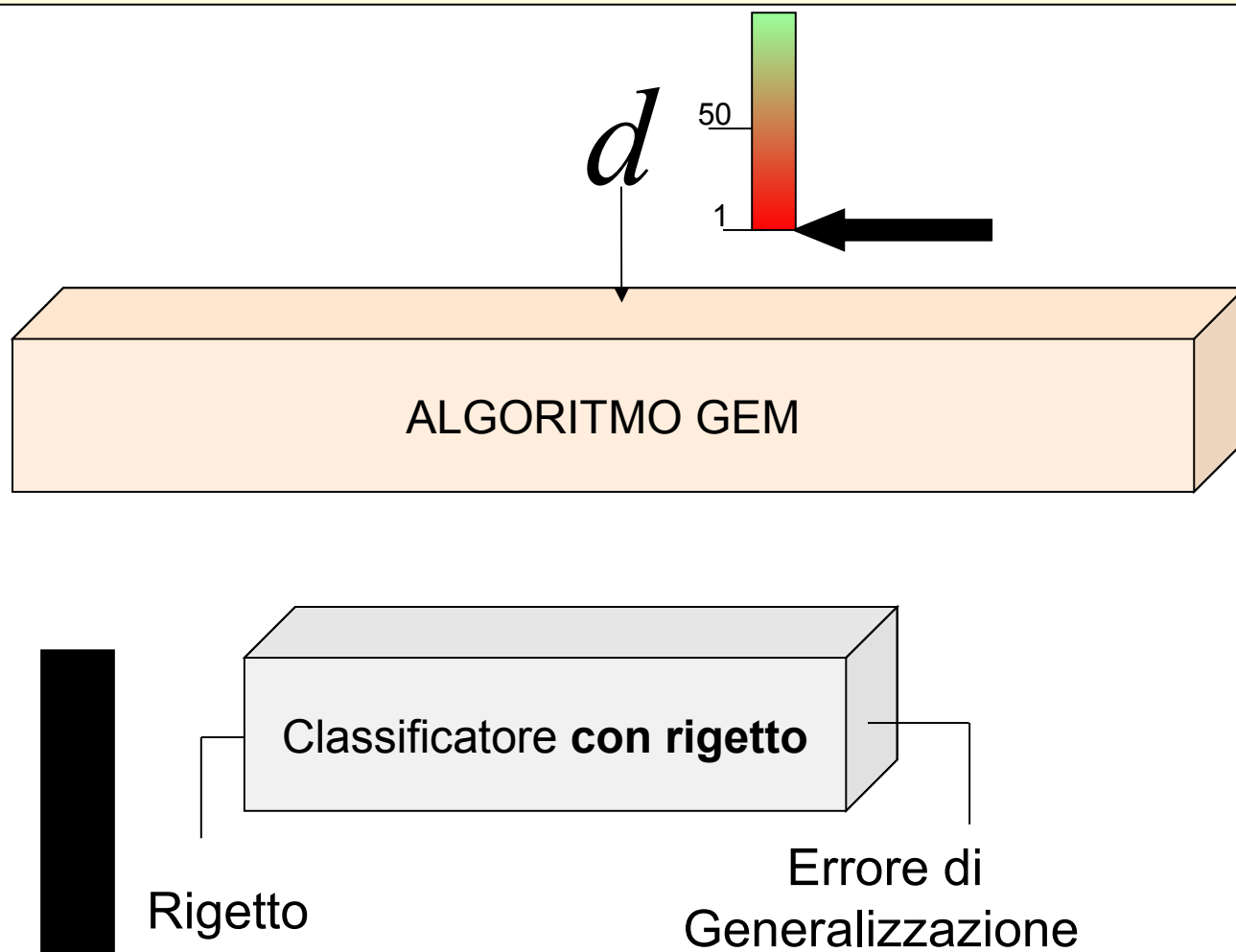
Proprietà teoriche e risultati
sperimentali

Classificatore con rigetto

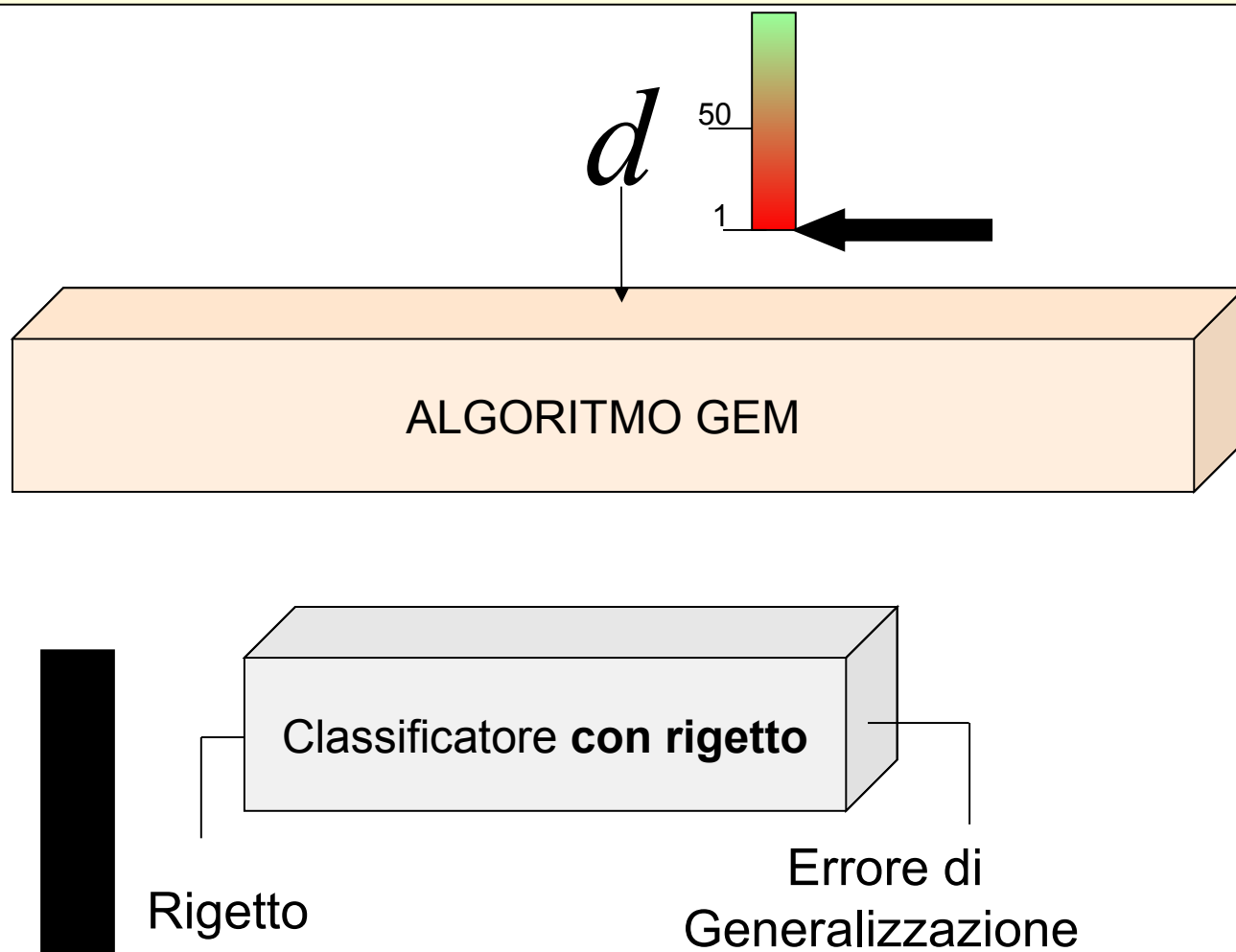
$$\hat{y}(\cdot) : \mathbf{R}^k \rightarrow \{0, 1, \textit{Rigetta}\}$$



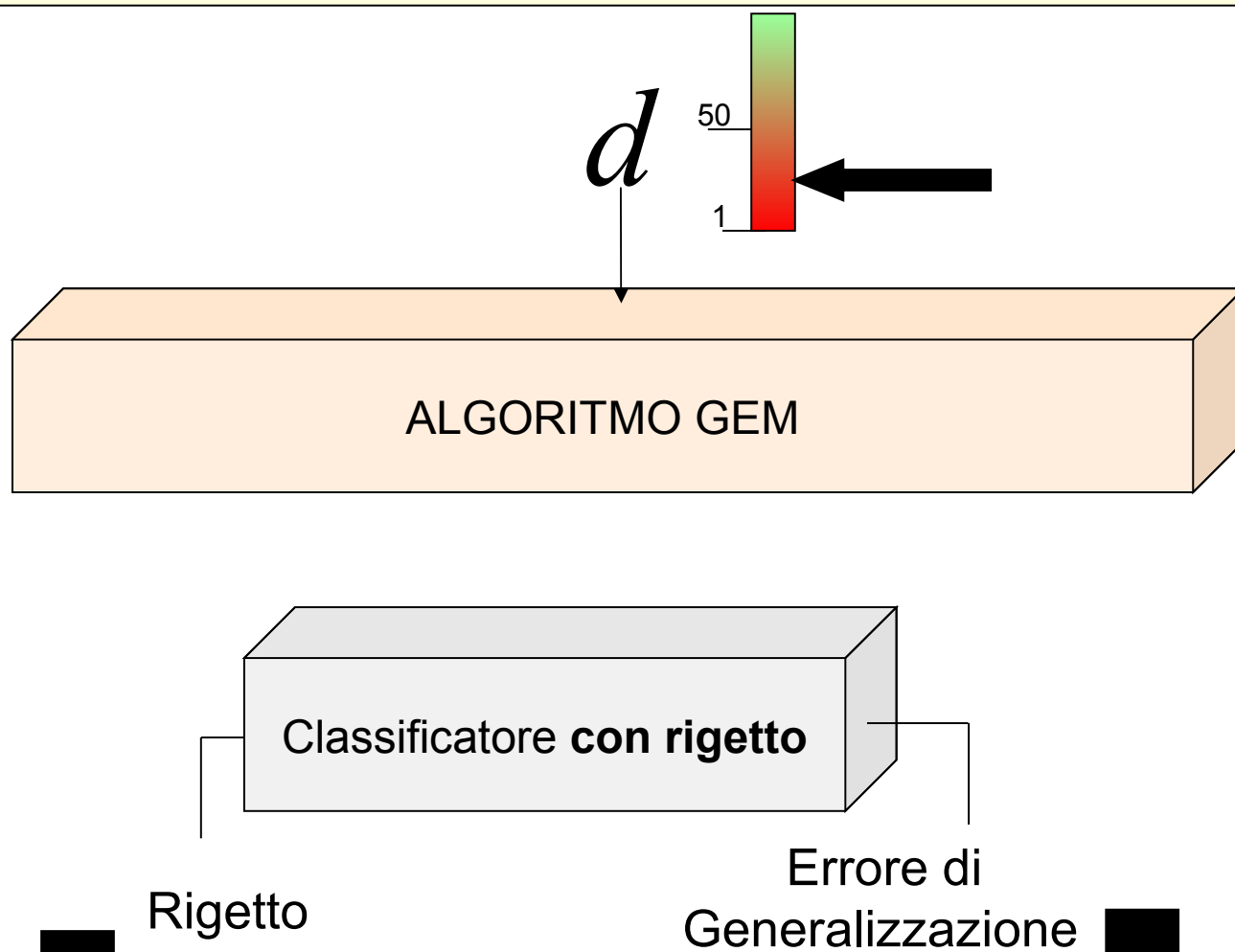
Il parametro “d”



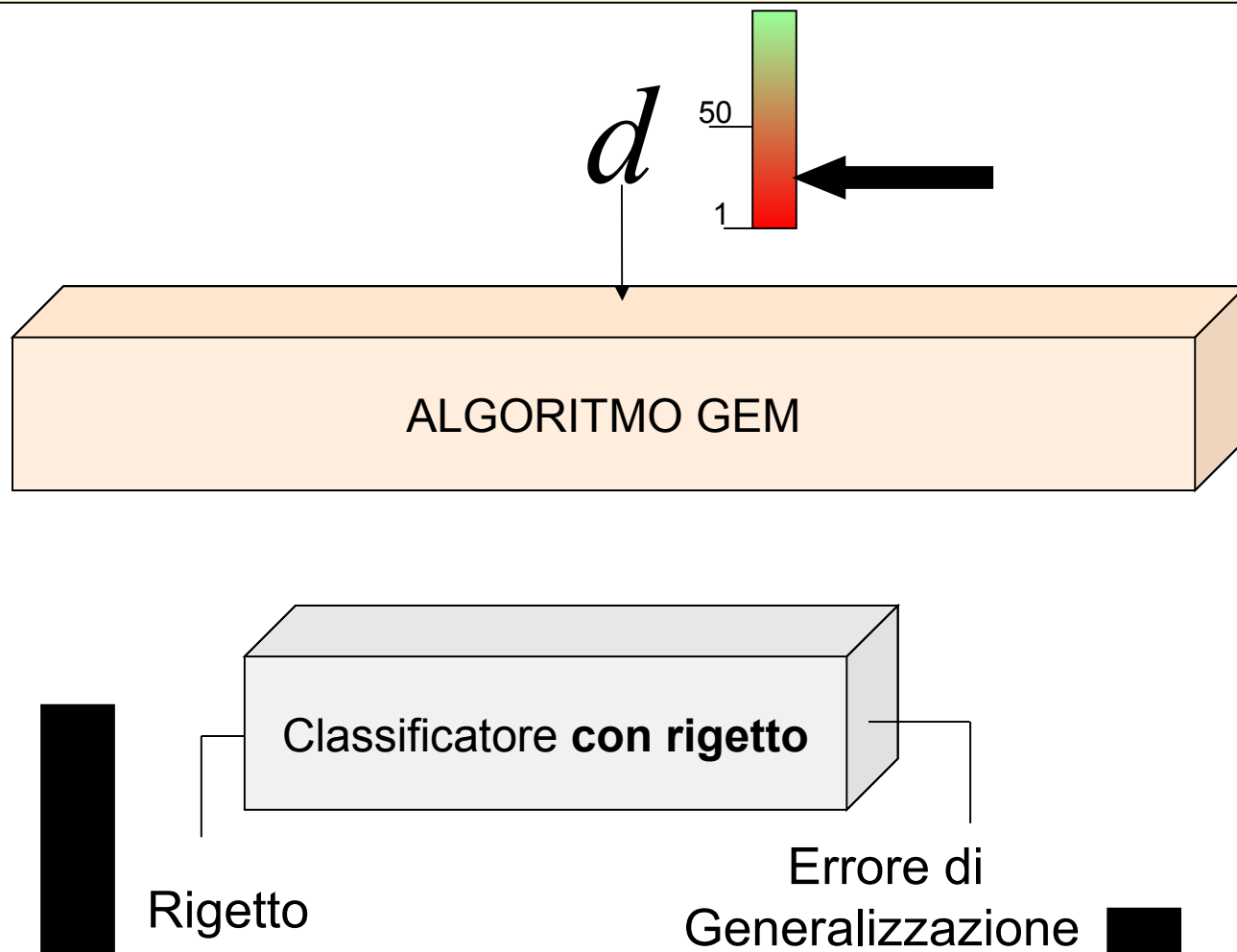
Il parametro “d”



Realtà “semplici”



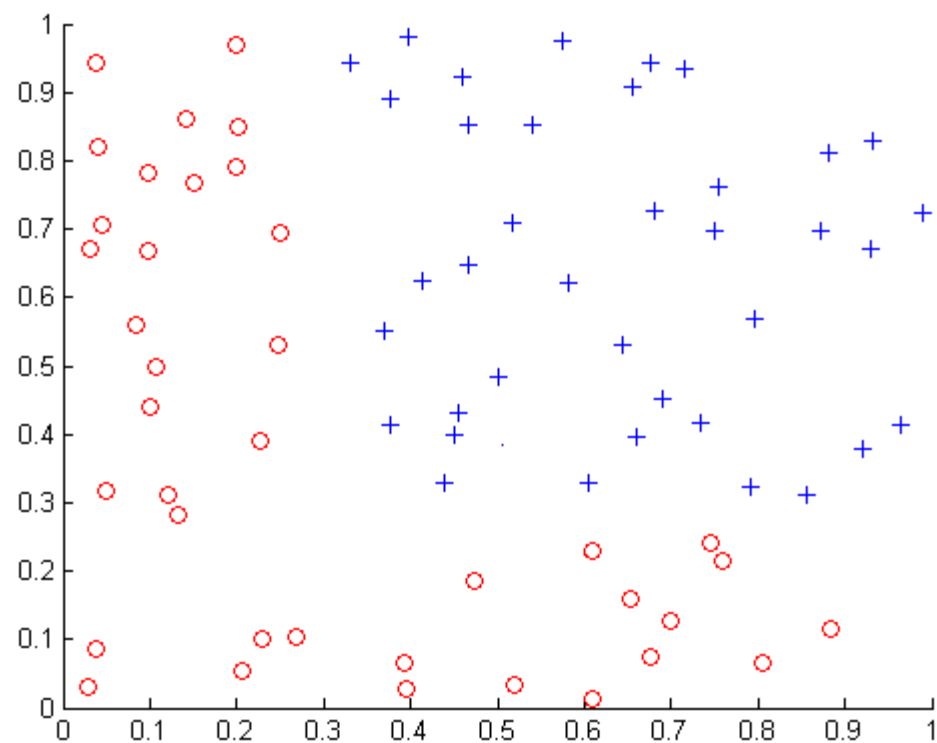
Realtà “complesse”



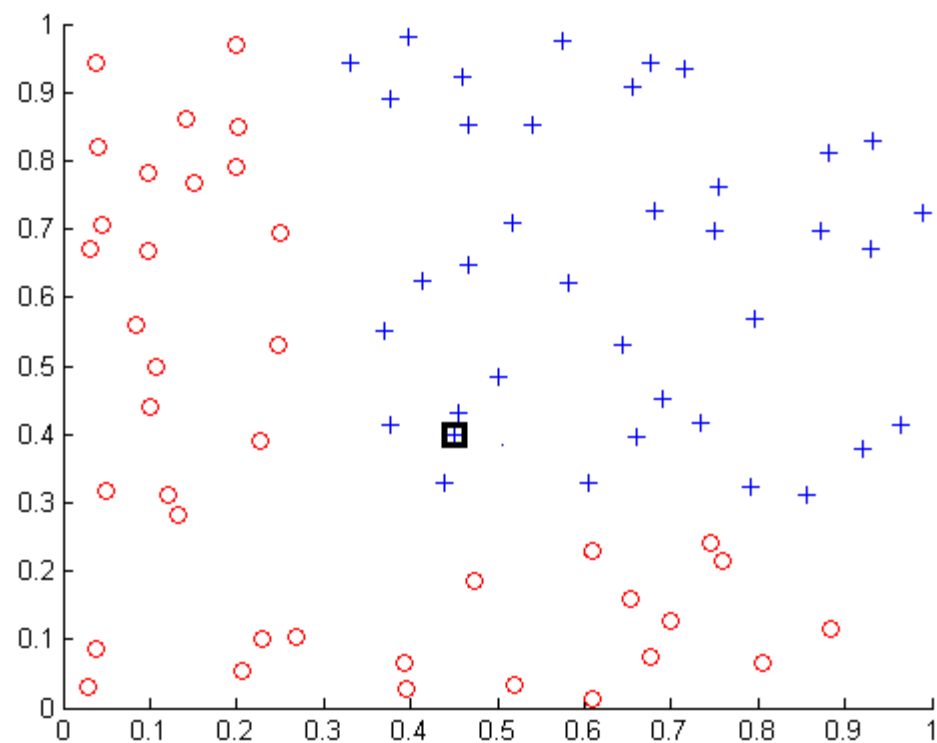
Logica di funzionamento (molto) in breve

- Costruzione incrementale di regioni
 - di natura alterna
 - di complessità decrescente (guidata da “d”)

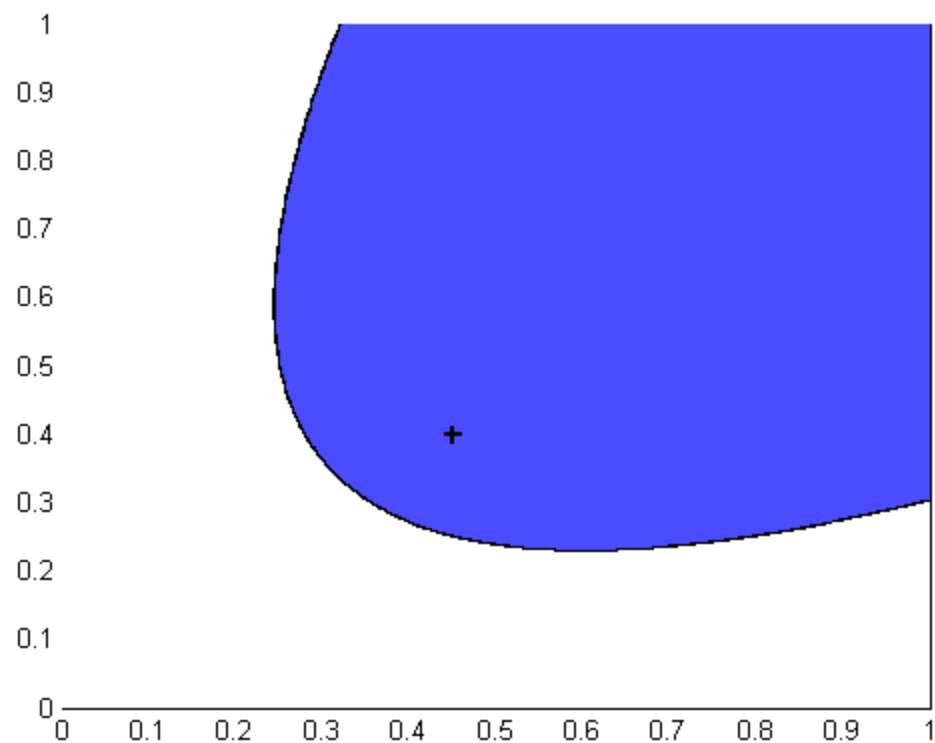
Un esempio (molto) semplice



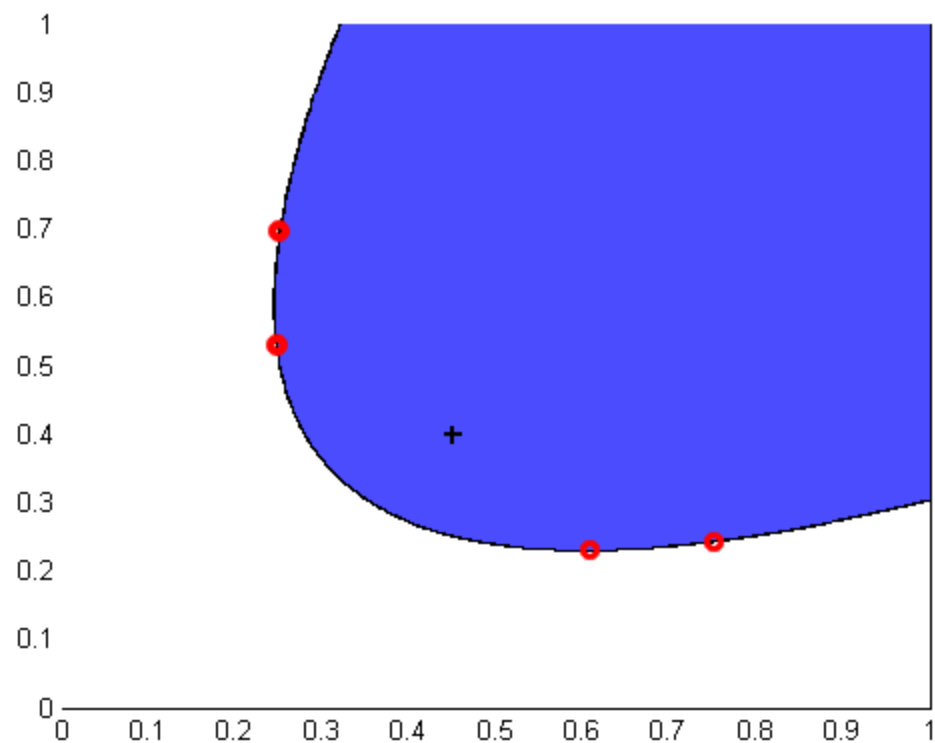
Un esempio (molto) semplice



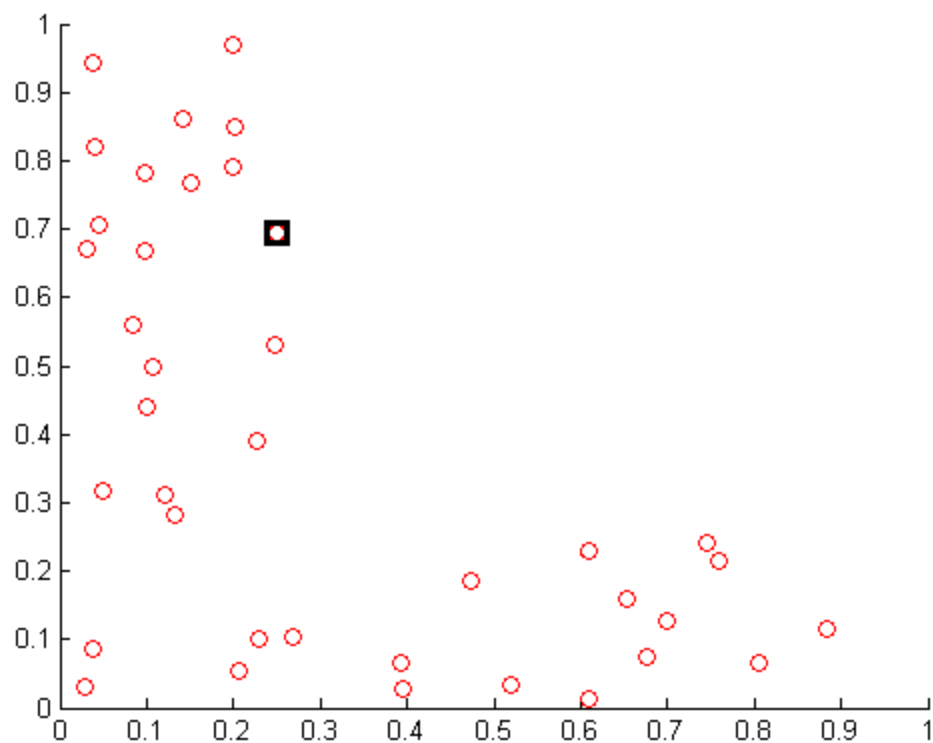
Un esempio (molto) semplice



Un esempio (molto) semplice

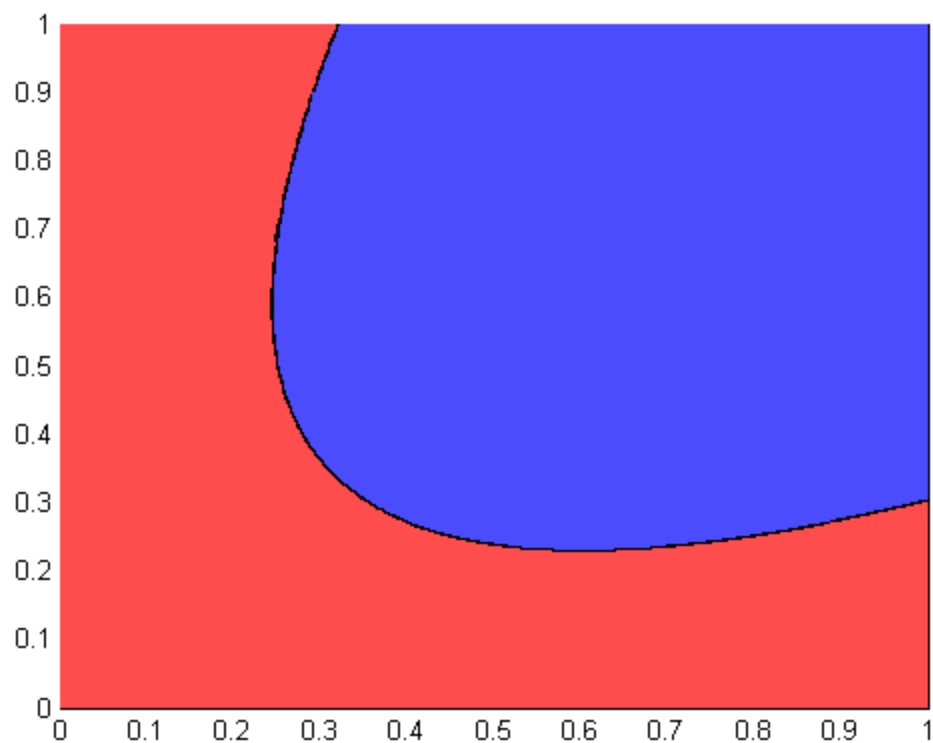


Un esempio (molto) semplice



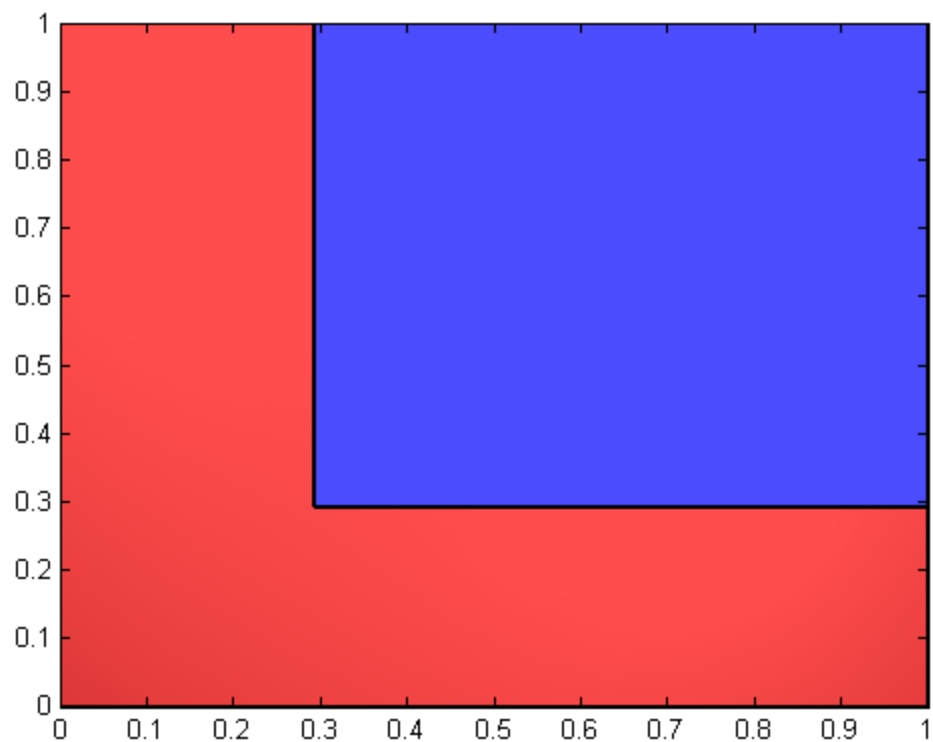
Un esempio (molto) semplice

$\hat{y}$

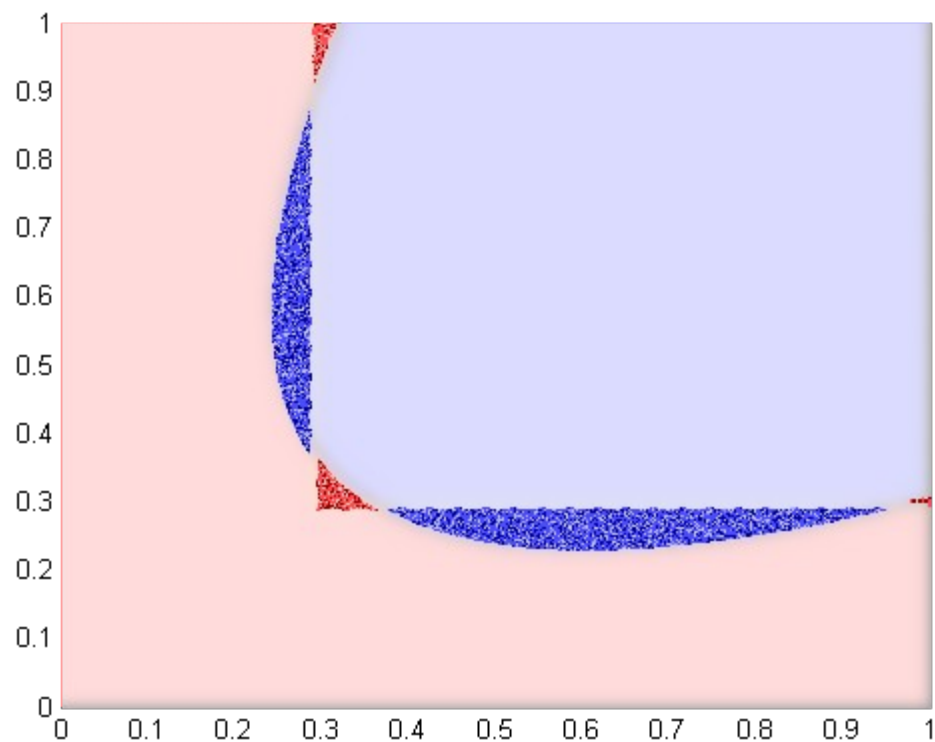


Un esempio (molto) semplice

y



Un esempio (molto) semplice

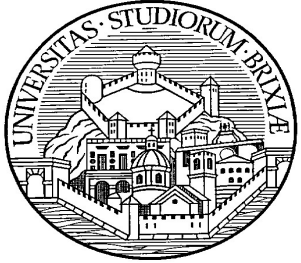
 $PE(\hat{y})$ 

PROMEMORIA

ERRORE DI GENERALIZZAZIONE

$$PE(\hat{y}) := \Pr(\hat{y}(x) \neq y(x))$$

Schema della presentazione



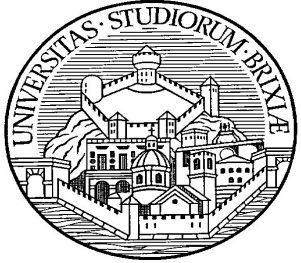
“Un **algoritmo** per la costruzione di **classificatori**
da dati sperimentali con **errore di generalizzazione garantito**”

A. La classificazione (binaria)

B. L'algoritmo: GEM
(*Guaranteed Error Machine*)

C. Proprietà teoriche e risultati
sperimentali

Terza Parte



“Un **algoritmo** per la costruzione di **classificatori**
da dati sperimentali con **errore di generalizzazione garantito**”

La classificazione (binaria)

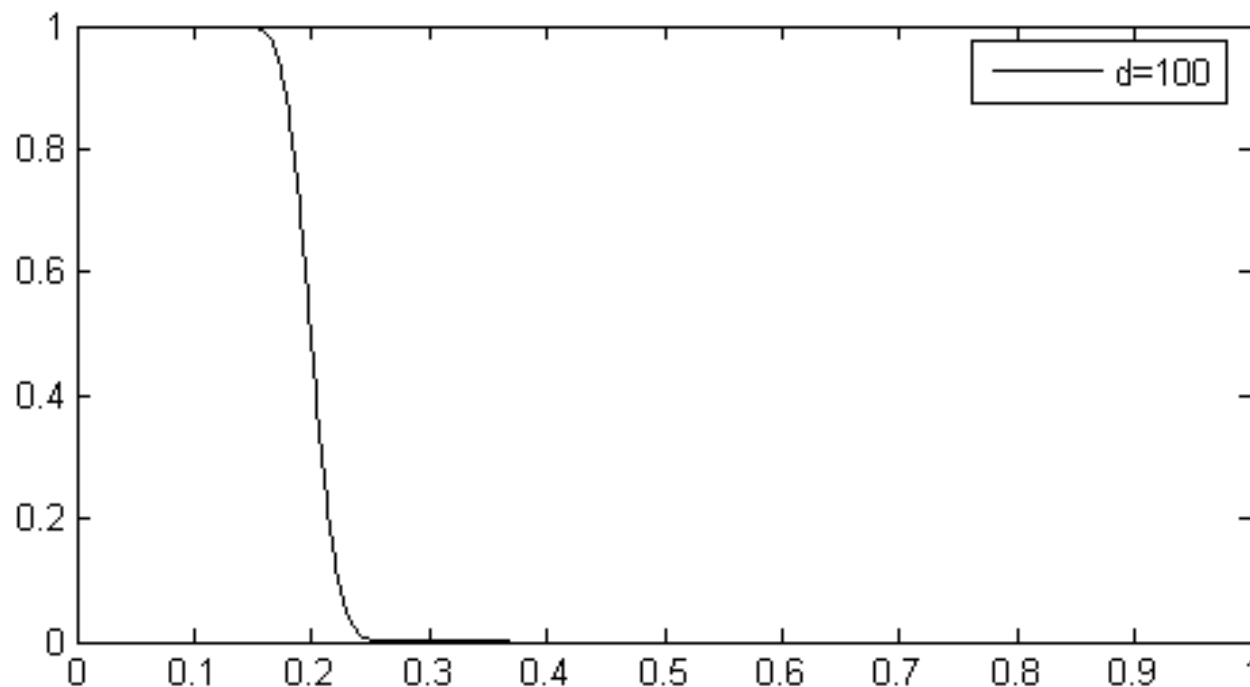
L'algoritmo: GEM
(*Guaranteed Error Machine*)

C. Proprietà teoriche e risultati
sperimentali

Il teorema fondamentale

La distribuzione dell'errore di generalizzazione di $\hat{y}$ è nota a priori ed è:

$$F_{PE}(z) = 1 - \sum_{i=0}^{d-1} \binom{N-1}{i} z^i (1-z)^{N-i-1}$$

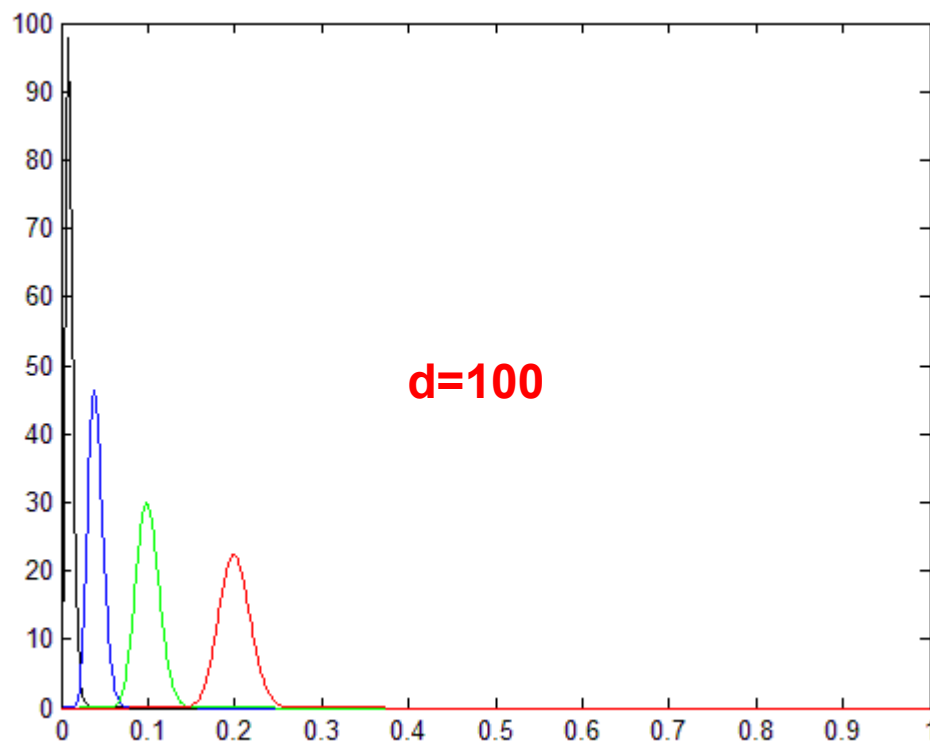


Il teorema fondamentale

La distribuzione dell'errore di generalizzazione di $\hat{y}$ è nota a priori ed è:

$$F_{PE}(z) = 1 - \sum_{i=0}^{d-1} \binom{N-1}{i} z^i (1-z)^{N-i-1}$$

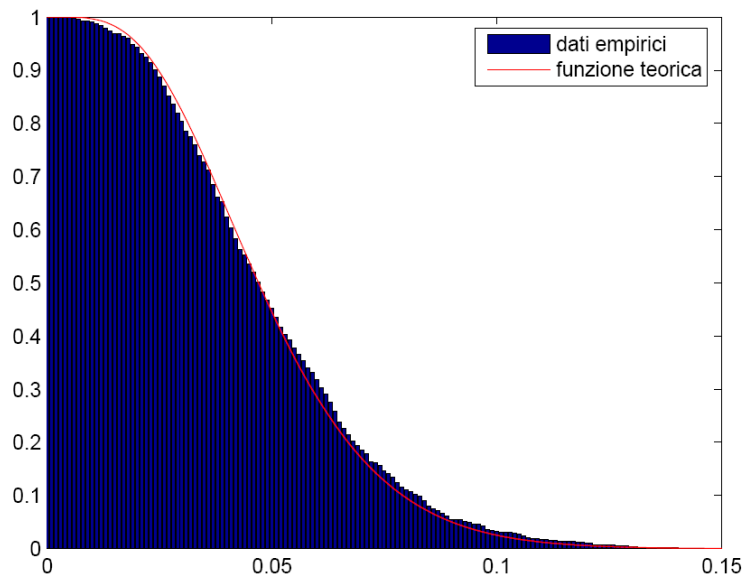
Densità (N=500)



Risultati sperimentali

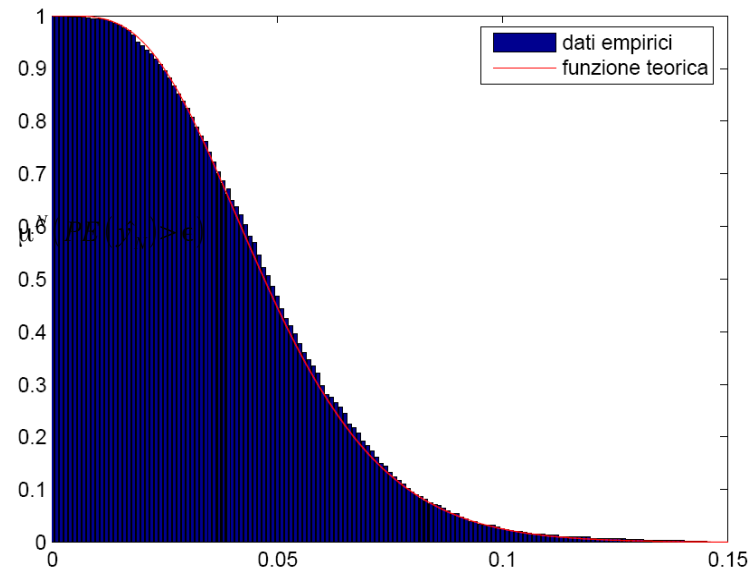
d=5

GEM



N=100, 1000 Multiestrazioni

Ideal GEM



N=100, 1000 Multiestrazioni

Esperimenti su dati reali (I)



- *Lunghezza* dei sepali
- *Larghezza* dei sepali
- *Lunghezza* dei petali
- *Larghezza* dei petali

Esperimenti su dati reali (II)

d	V.A.	Tot. Rigetti	Tot. Errori	Tot. Corretti
2	2.22%	58	2 (2.22%)	30
3	3.33%	55	5 (5.56%)	30
4	4.44%	48	6 (6.67%)	36
5	5.56%	38	5 (5.56%)	47
6	6.67%	32	6 (6.67%)	52
10	11.11%	0	8 (8.89%)	82

Esperimenti su dati reali (III)

Breast: 683 campioni; 10 attributi (interi)

Haberman: 294 campioni; 3 attributi (interi)

Pima: 768 campioni; 8 attributi (interi e reali)

Bupa: 345 campioni; 6 attributi (interi e reali)

Credit: 653 campioni; 15 attributi (categorici, reali, interi)

Glass: 163 campioni; 9 attributi (**reali**)

Iris Setosa VS Versicolor e Virginica: 149 campioni; 4 attributi (**reali**)

Iris Versicolor VS Virginica: 99 campioni; 4 attributi (**reali**)

Lettere AH: 1523 campioni; 16 attributi (interi)

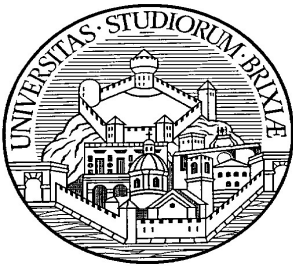
Lettere FT: 1571 campioni; 16 attributi (interi)

FINE

La classificazione (binaria)

L'algoritmo: GEM
(*Guaranteed Error Machine*)

Proprietà teoriche e risultati
sperimentali

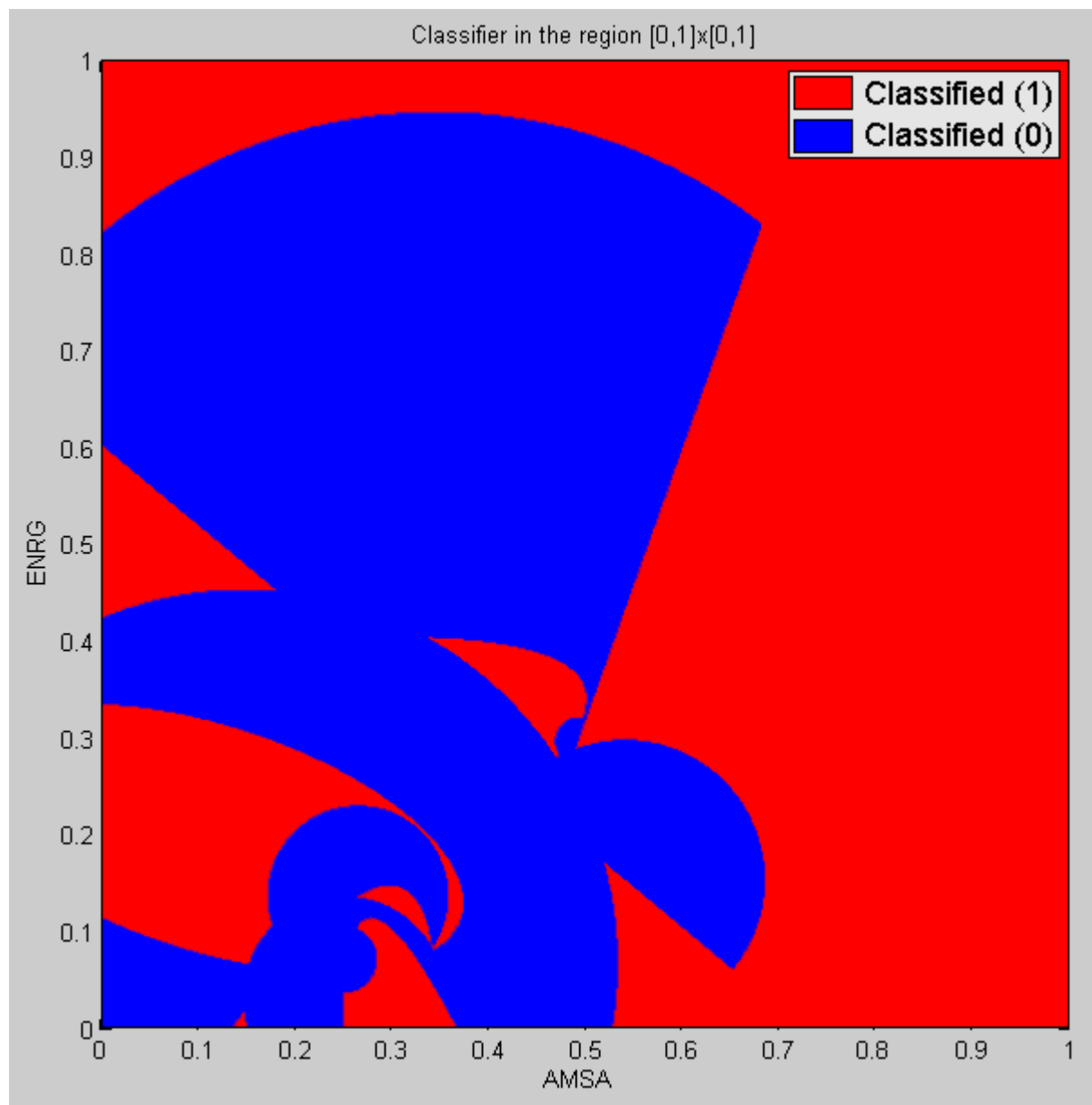


“Un algoritmo per la costruzione di classificatori
da dati sperimentali con errore di generalizzazione *garantito*”

Grazie per l'attenzione

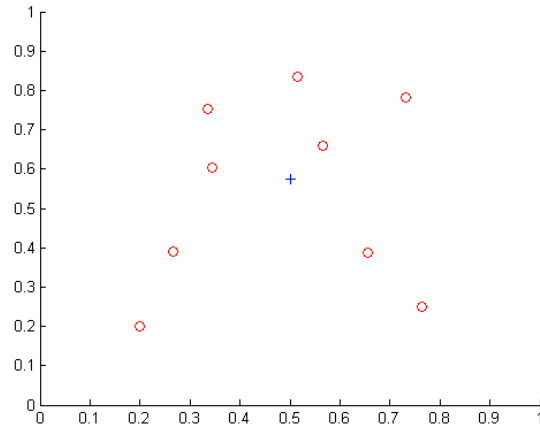
Bonus

GEM “asimmetrico”

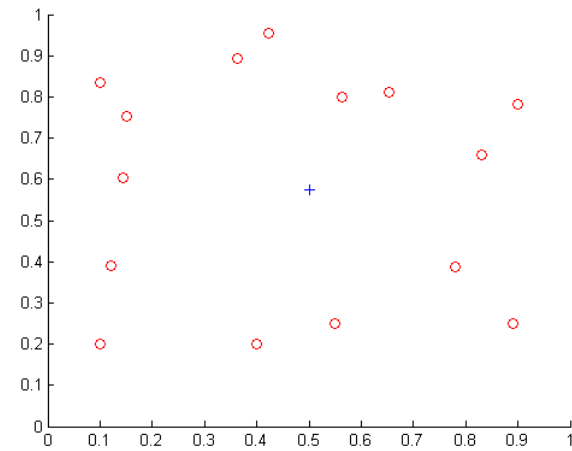


Tipi di regioni costruibili (I)

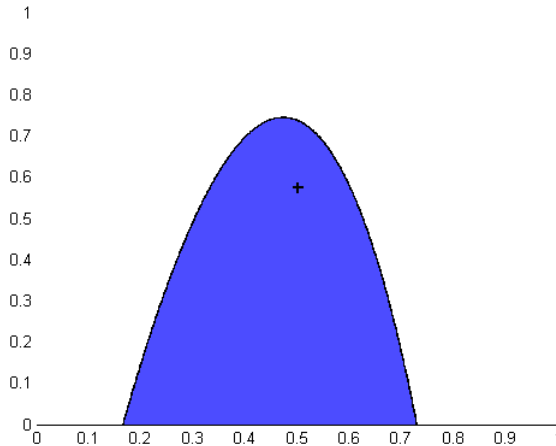
Parabole



Ellissi



I

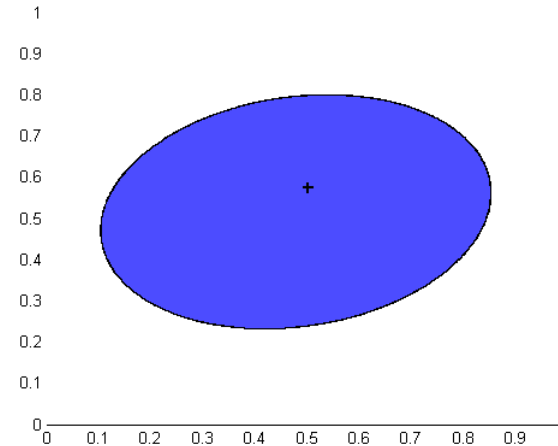


eci

$-$

A

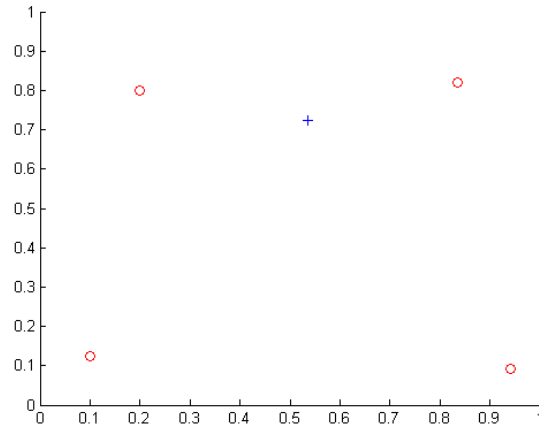
0



$1,$

Tipi di regioni costruibili (II)

Cerchi decentrati



$P_{n+1}(x_b, y(x_b)):$

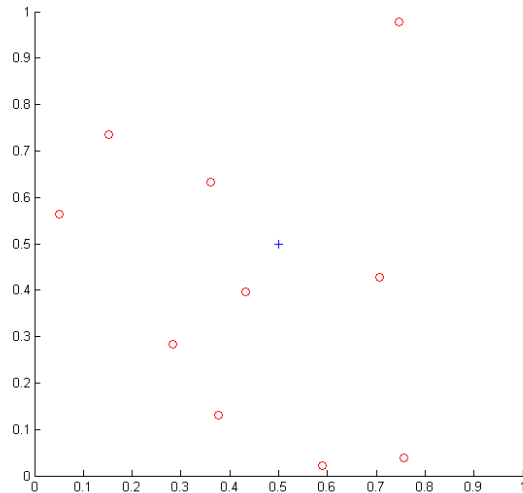
$\min_{k \geq 0, B \in \mathbb{R}^{1 \times n}} k$

con vincoli: $k \|x_j - x_b\|^2 + B(x_j - x_b) \geq 1,$
 $\forall x_j \in \mathbf{L}, y(x_j) \neq y(x_b)$



Tipi di regioni costruibili (III)

Cerchi



1

0.9

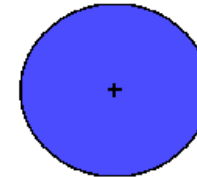
0.8

0.7

0.6

0.5

0.4



$P_{\text{CMaX}}(x_b, y(x_b)) :$

$\min_{k \geq 0}$

con vincoli:

k

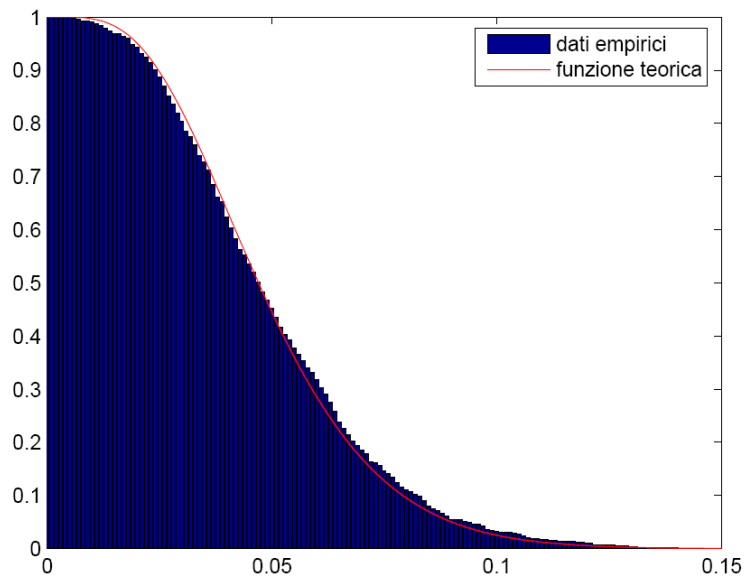
$k||x_j - x_b||^2 \geq 1,$

$\forall x_j \in L, y(x_j) \neq y(x_b)$

Risultati sperimentali

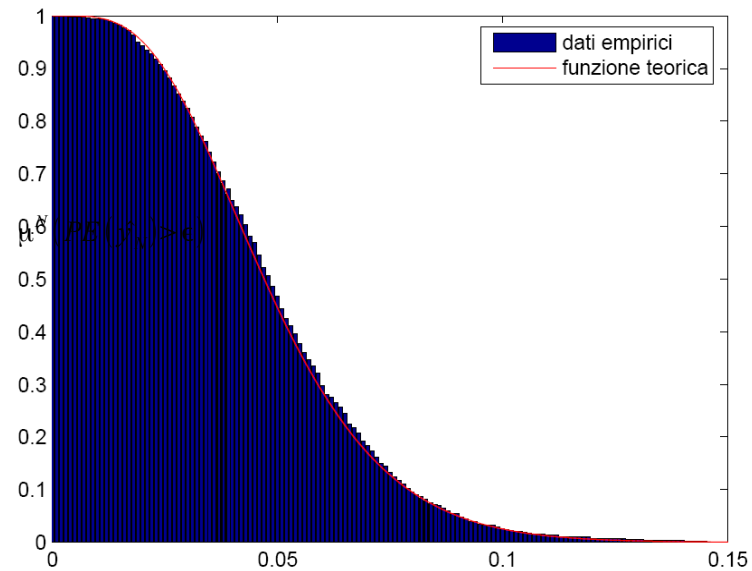
d=5

GEM



N=100, 1000 Multiestrazioni

Ideal GEM

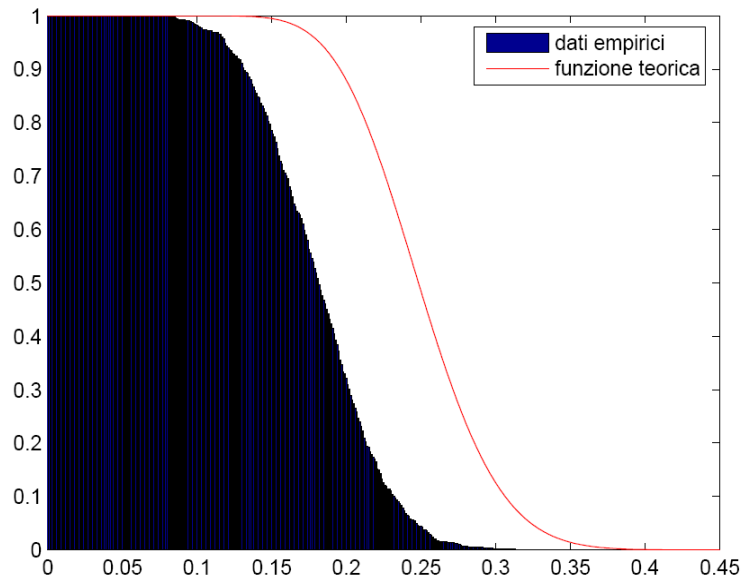


N=100, 1000 Multiestrazioni

Risultati sperimentali

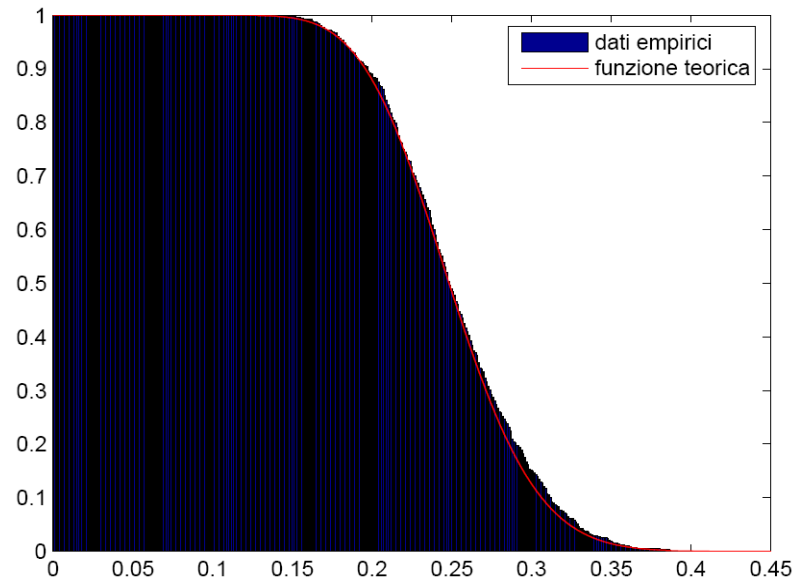
d=25

GEM



N=100, 1000Multiestrazioni

Ideal GEM



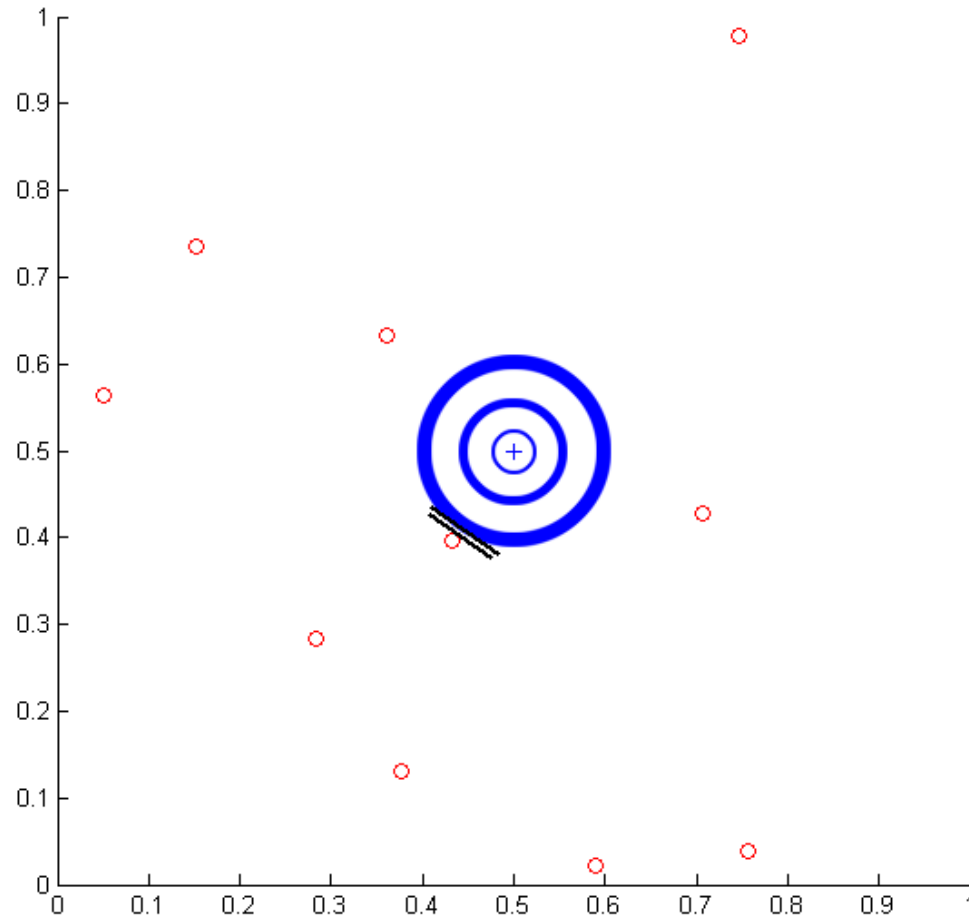
N=100, 1000Multiestrazioni

Risultati sperimentali (errori in alcuni test)

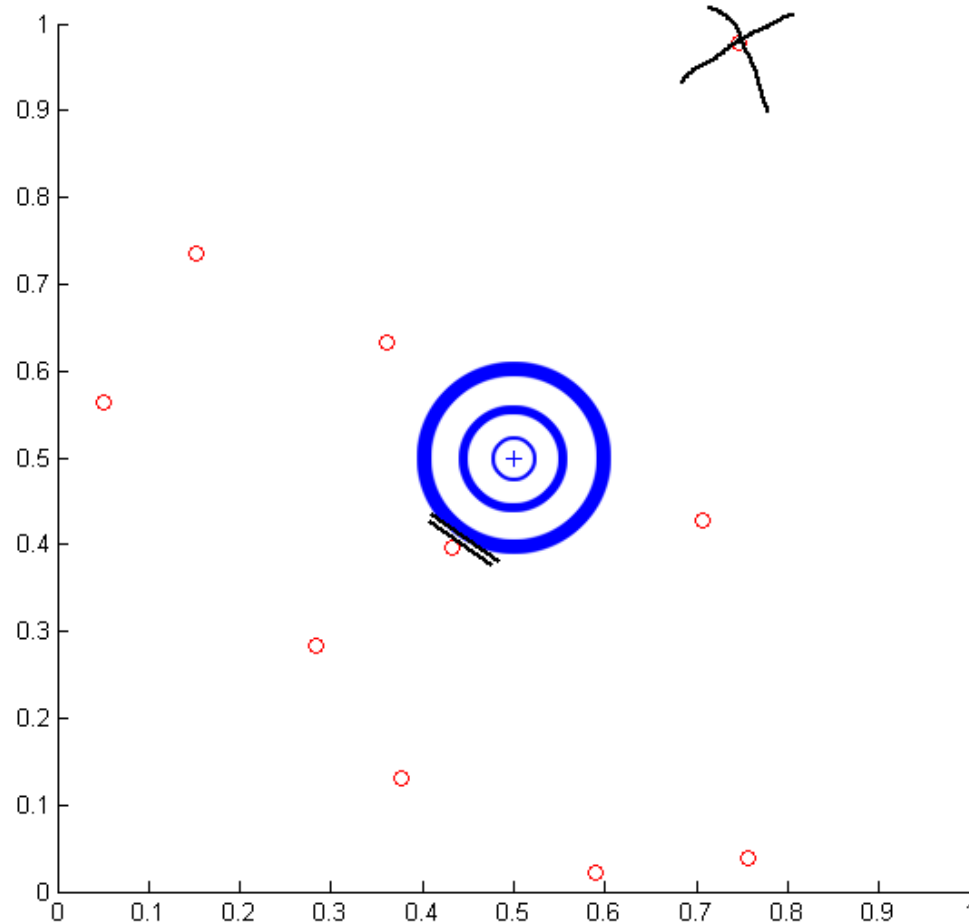
Dataset	GEM	NNC	SVM $C = \infty$	SVM $C < \infty$
Breast	28.12	29	27	19
Haberman	84.14	107	111	71
Pima	243.54	247	243	203
Bupa	130.9	124	121	107
Credit	151.7	214	205	190
Glass	40.75	36	42	34

- Competitivo in quanto classificatore senza rigetto
- Garanzie stringenti
- Robustezza a fronte di ipotesi teoriche non soddisfatte

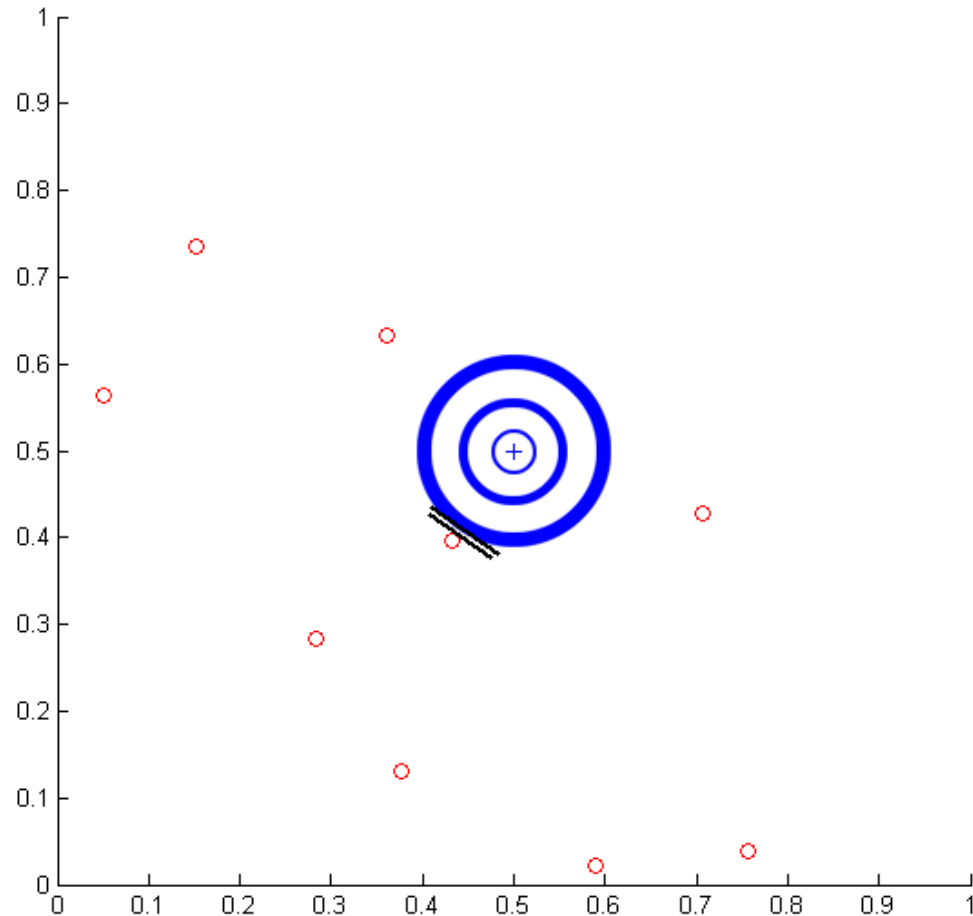
Relazione tra errore e punti sul bordo (spiegazione intuitiva)



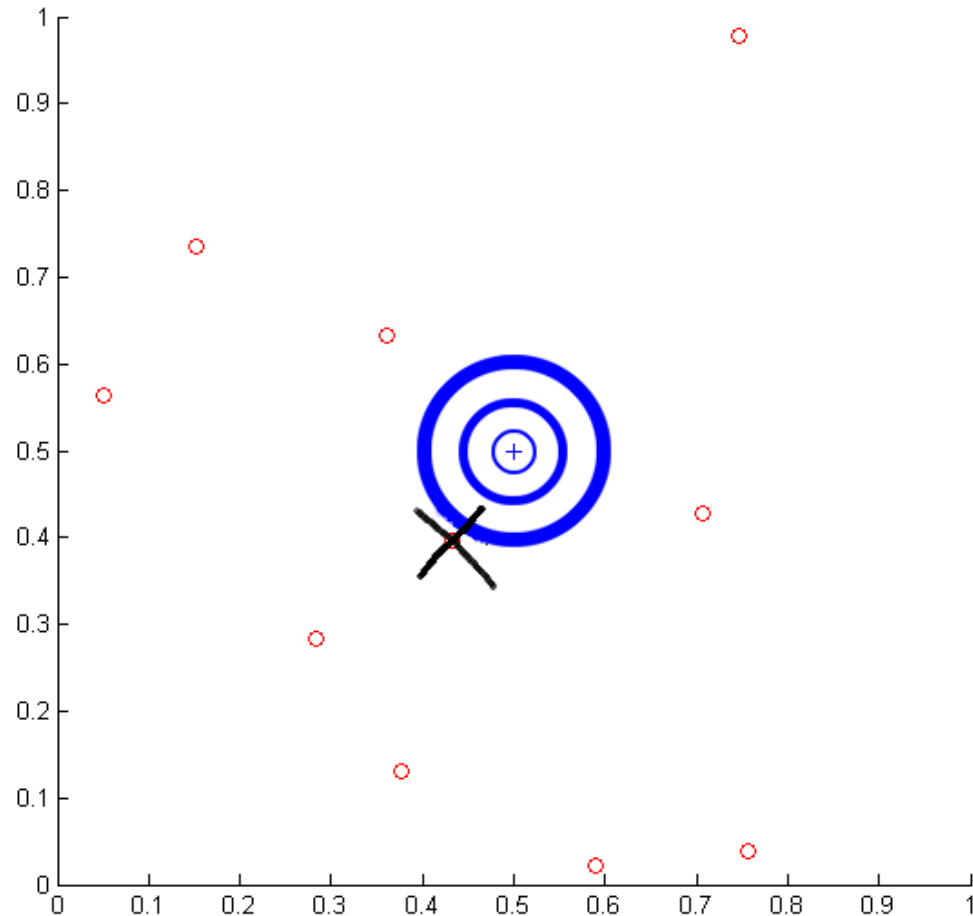
Relazione tra errore e punti sul bordo (spiegazione intuitiva)



Relazione tra errore e punti sul bordo (spiegazione intuitiva)

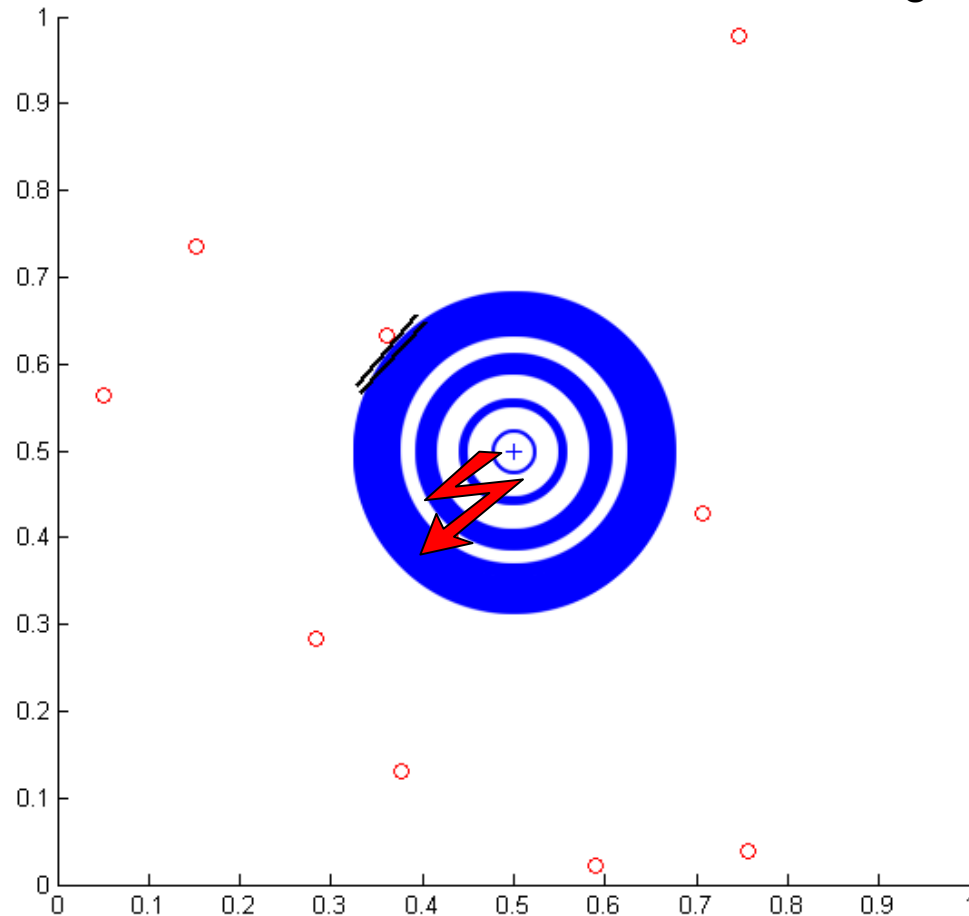


Relazione tra errore e punti sul bordo (spiegazione intuitiva)

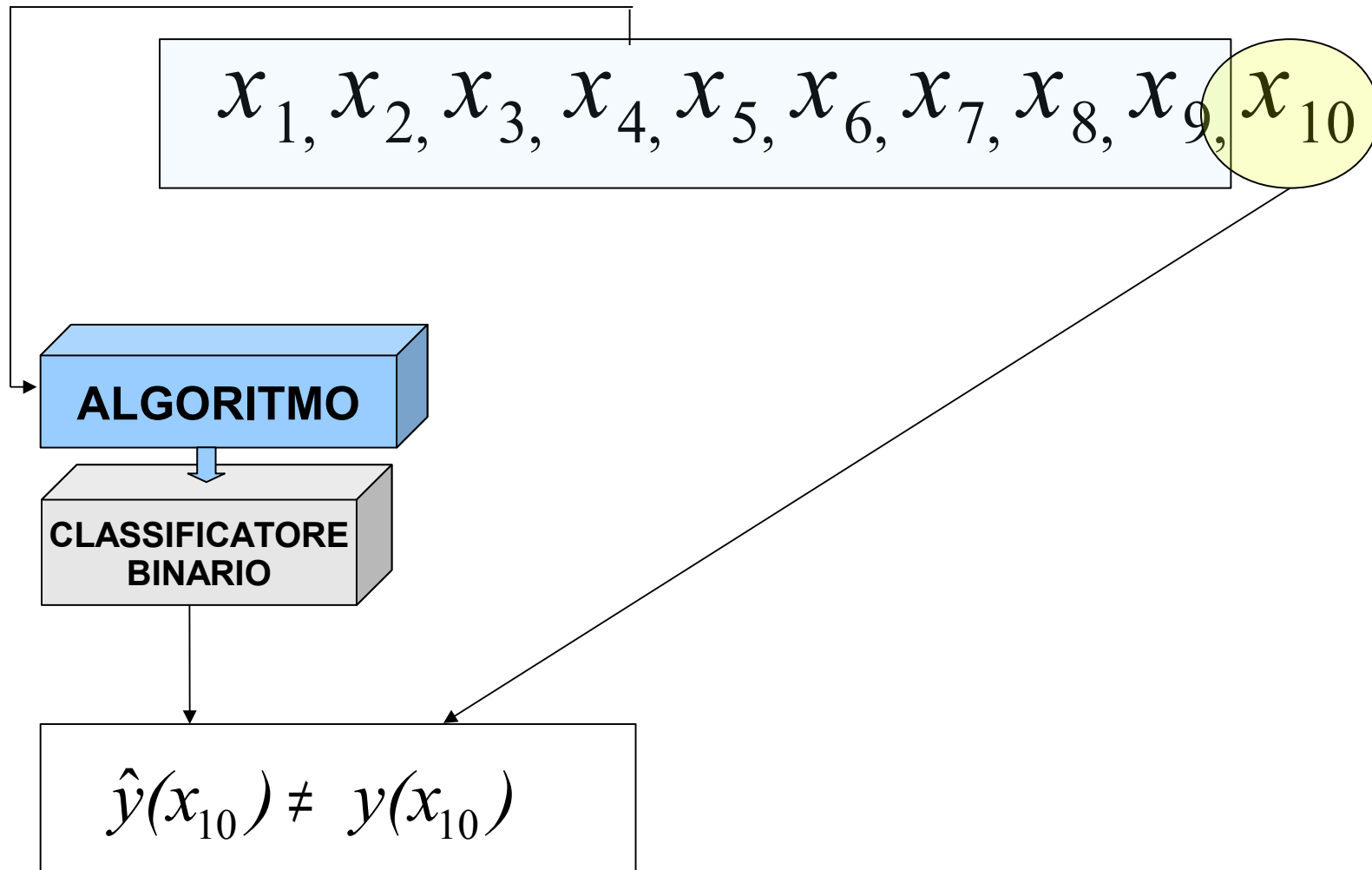


Relazione tra errore e punti sul bordo (spiegazione intuitiva)

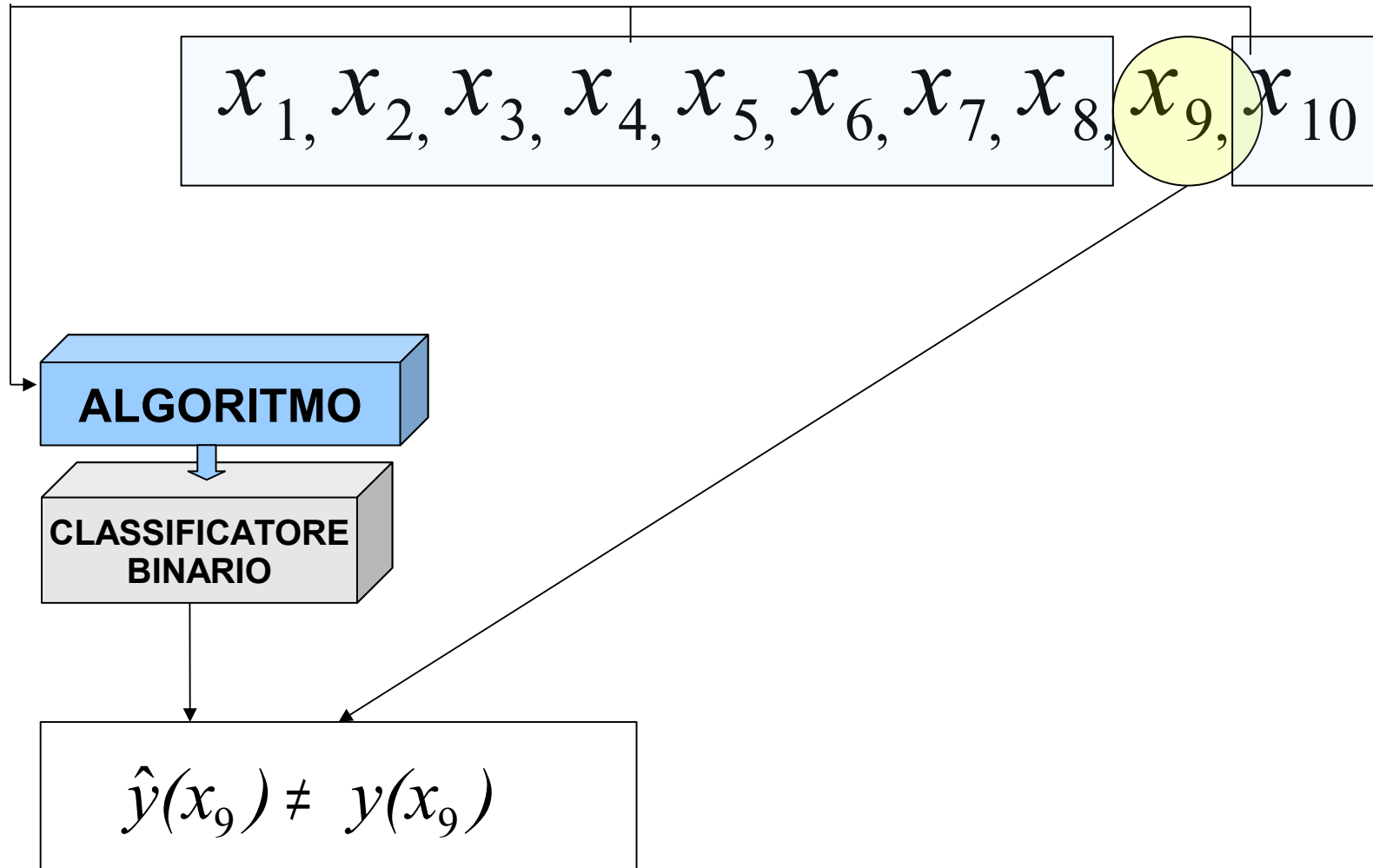
Un punto che si trovava sul bordo della regione-soluzione, se rimosso, rientra nella nuova regione costruita.



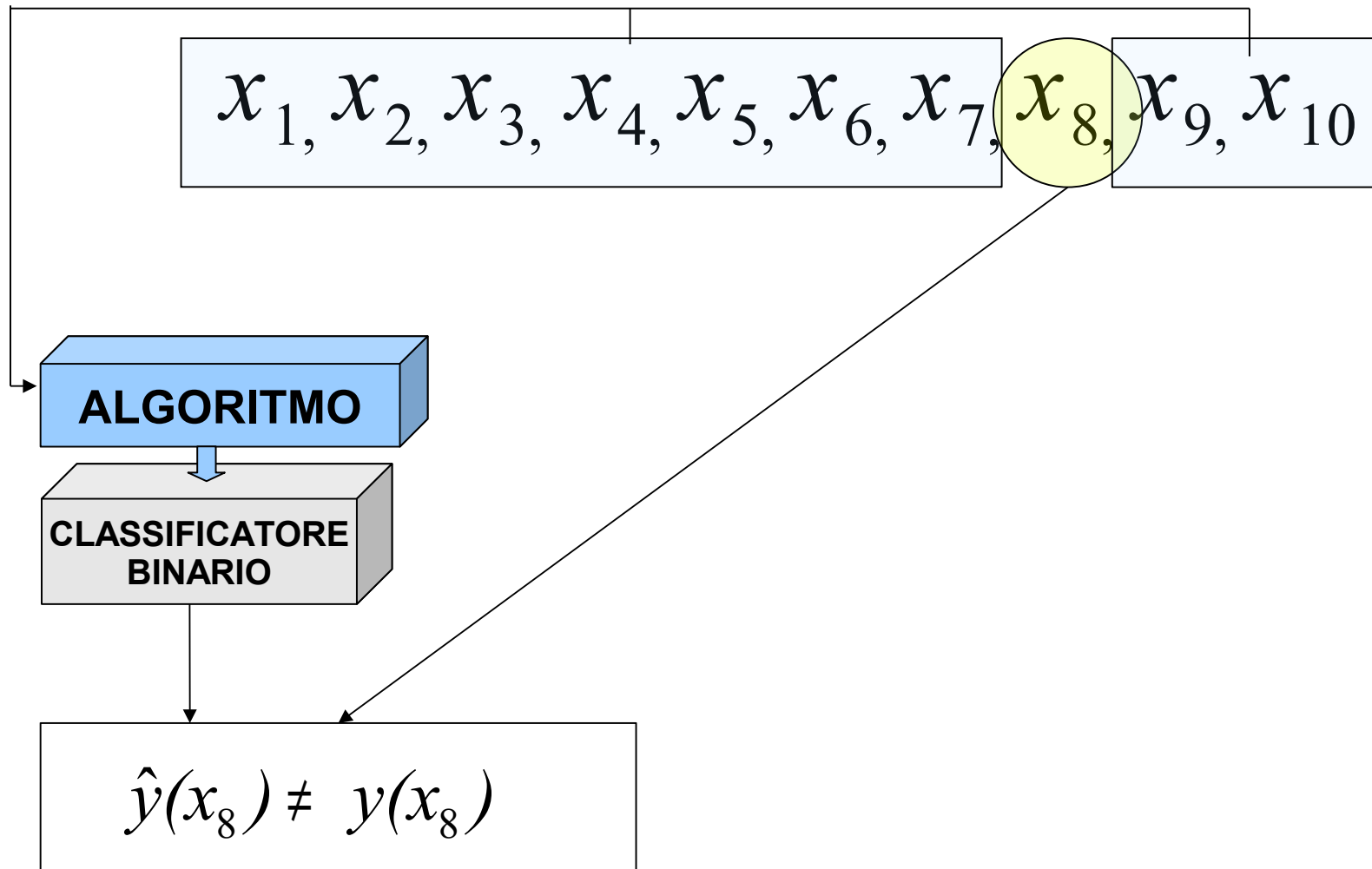
Il leave-one-out



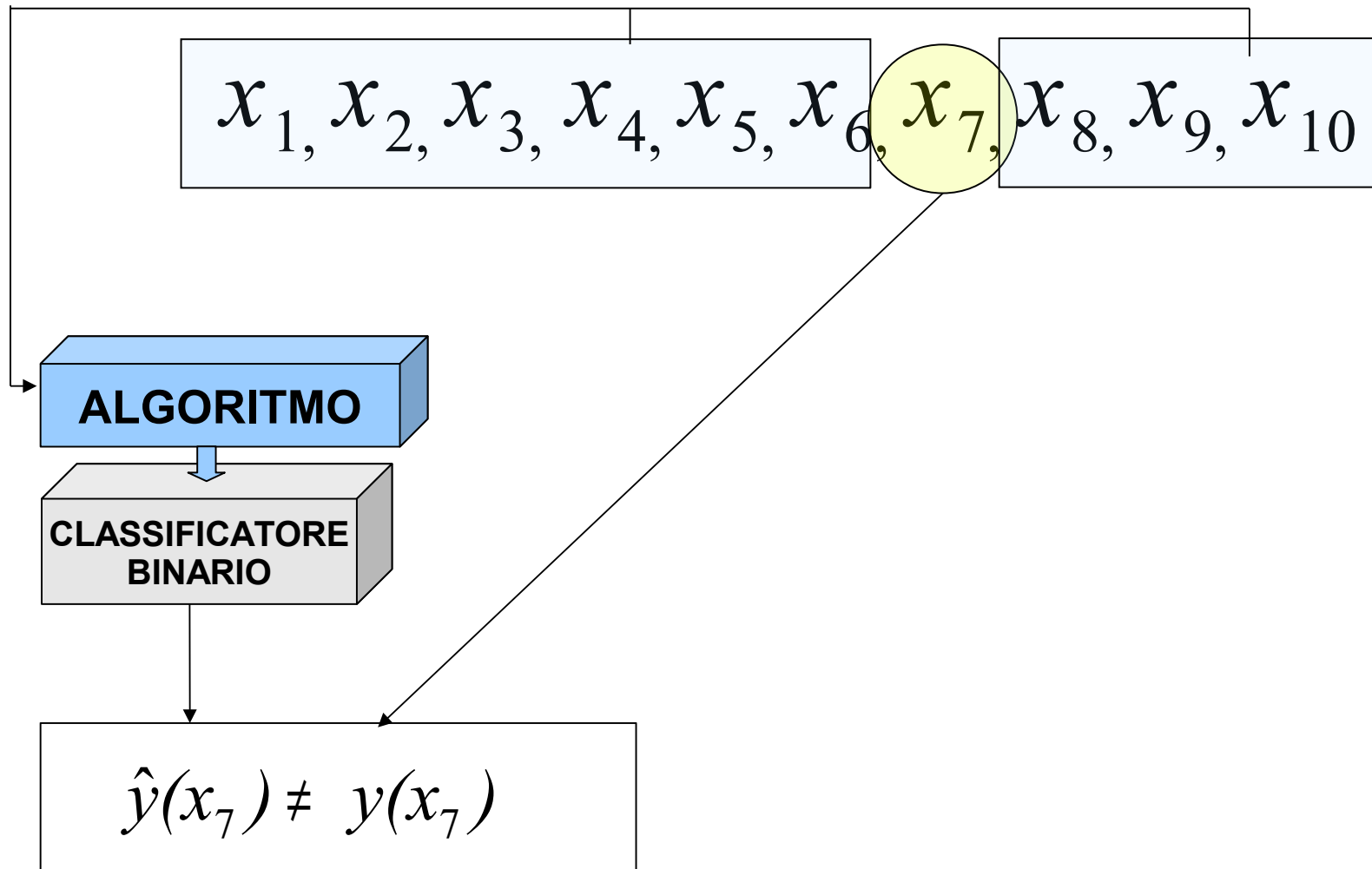
Il leave-one-out



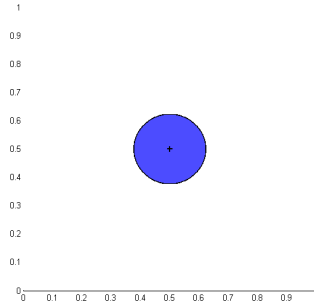
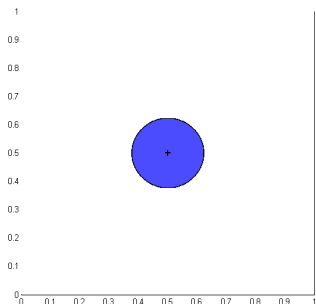
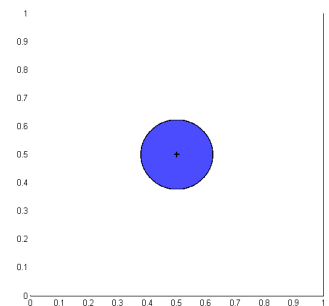
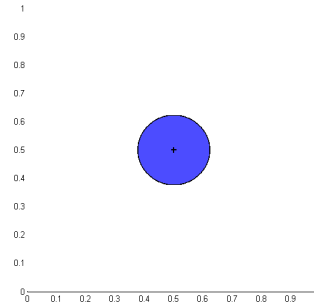
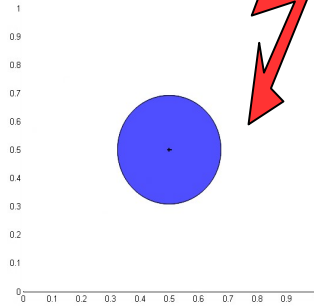
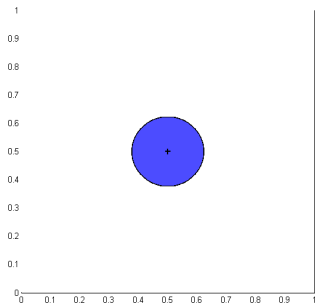
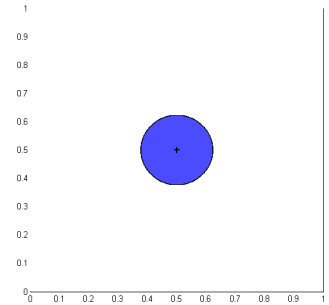
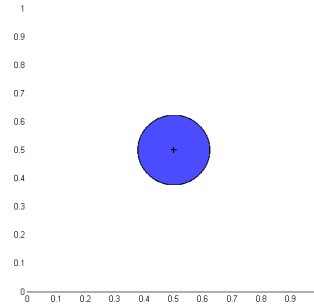
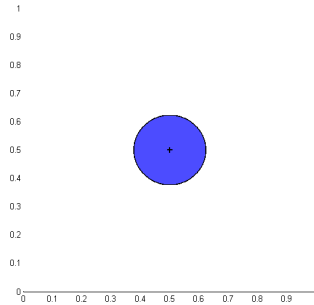
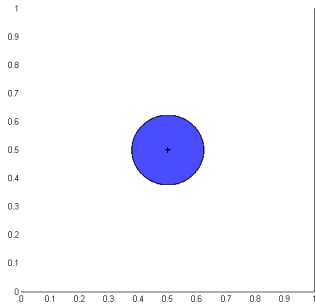
Il leave-one-out



Il leave-one-out



Il leave-one-out



$$\frac{1}{10}$$

$$\frac{\textit{punti sul bordo}}{N}$$

Relazione tra errore e punti sul bordo (spiegazione intuitiva)

- L'errore di leave-one-out è “molto legato” al valore atteso dell'errore di generalizzazione
- Dalla distribuzione dell'errore di generalizzazione si può ricavare il valore atteso dell'errore di generalizzazione: $\frac{d}{N}$
- L'algoritmo è stato costruito con proprietà che permettono di generalizzare quanto visto nel caso banale del cerchio (**ottimizzazione convessa**)