Nozioni di Apprendimento Automatico

Quando è Necessario l'Apprendimento (Automatico) ?

Quando il sistema deve...

- adattarsi all'ambiente in cui opera (anche personalizzazione automatica);
- migliorare le sue prestazioni rispetto ad un particolare compito;
- scoprire regolarità e nuova informazione (conoscenza) a partire da dati empirici;
- acquisire nuove capacità computazionali.

Perchè non usare un approccio algoritmico tradizionale ?

- impossibile formalizzare esattamente il problema (e quindi dare una soluzione algoritmica);
- presenza di rumore e/o incertezza ;
- complessità alta nel formulare una soluzione: non si può fare a mano;
- mancanza di conoscenza "compilata" rispetto al problema da risolvere;

Ruolo dei Dati

Tipicamente...

- si hanno a disposizione (molti ?) dati
  - ottenuti una volta per tutte;
  - acquisibili interagendo direttamente con l'ambiente;
- (forse) conoscenza del dominio applicativo, ma
  - incompleta;
  - imprecisa (rumore, ambiguità, incertezza, errori, ...);

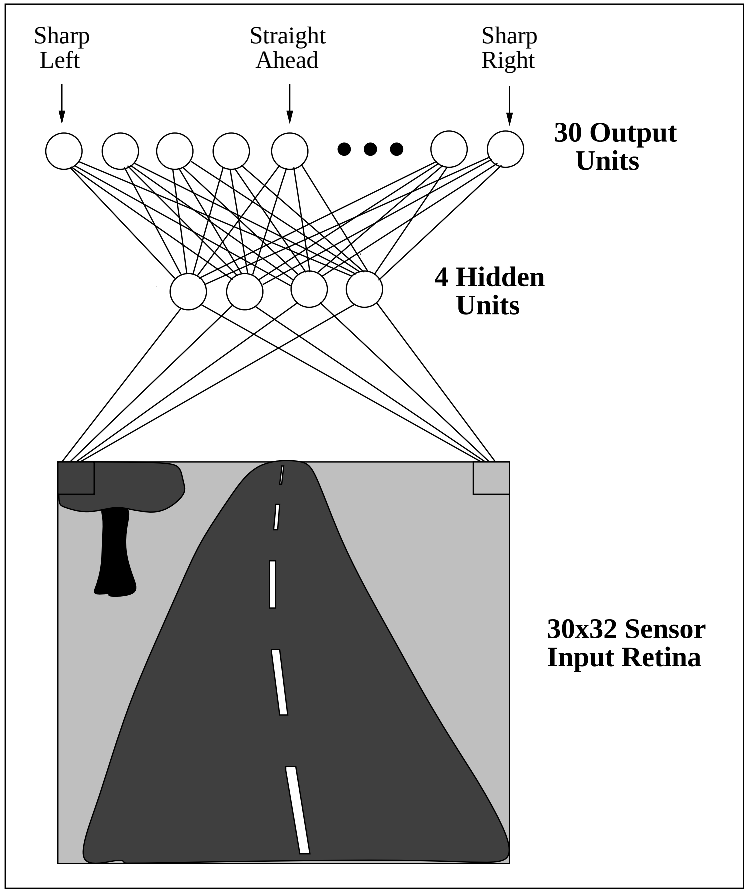
Desiderio: usare i dati per

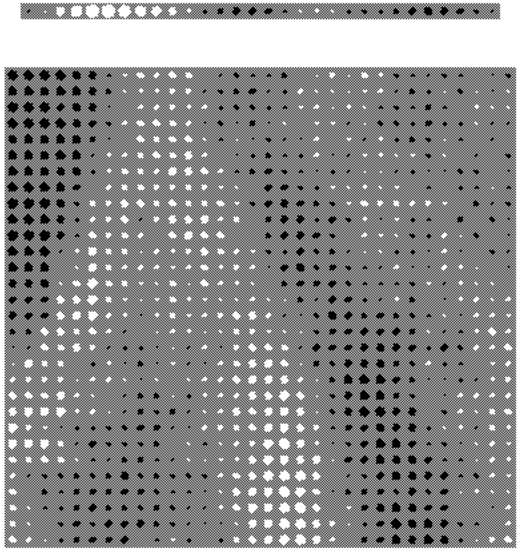
- ottenere nuova conoscenza;
- raffinare la conoscenza di cui si dispone;
- correggere la conoscenza di cui si dispone;

Es. - Riconoscimento di Cifre Manoscritte

- impossibile formalizzare esattamente il problema: disponibili solo esempi;
- possibile presenza di rumore e dati ambigui;

Es. - Guidare una Automobile





Es. - Estrarre Conoscenza Medica dai Dati

Patient103 time=1 →	*Patient103* time=2 ... →	*Patient103* time=n
Age: 23	Age: 23	Age: 23
FirstPregnancy: no	FirstPregnancy: no	FirstPregnancy: no
Anemia: no	Anemia: no	Anemia: no
Diabetes: no	Diabetes: YES	Diabetes: no
PreviousPrematureBirth: no	PreviousPrematureBirth: no	PreviousPrematureBirth: no
Ultrasound: ?	Ultrasound: abnormal	Ultrasound: ?
Elective C−Section: ?	Elective C−Section: no	Elective C−Section: no
Emergency C−Section: ?	Emergency C−Section: ?	Emergency C−Section: **Yes**
...	...	...

Linee di Ricerca all’interno dell’ Apprendimento Automatico

- induzione di regole/alberi di decisione,
- algoritmi connessionisti (reti neurali),
- “clustering” & “discovery”,
- apprendimento basato sulle istanze
- apprendimento Bayesiano,
- apprendimento basato sulla spiegazione,
- apprendimento con rinforzo,
- apprendimento induttivo guidato dalla conoscenza,
- ragionamento per analogia & basato sui casi,
- algoritmi genetici,
- programmazione logica induttiva, . . .

Principali Paradigmi di Apprendimento

Apprendimento Supervisionato:

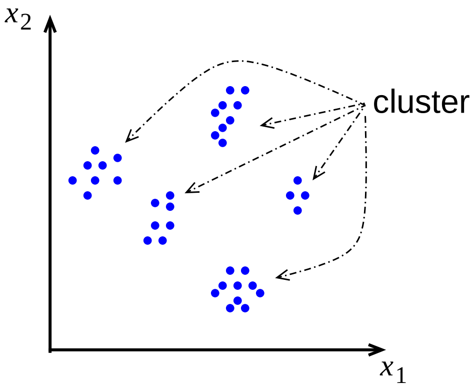
- dato in insieme di esempi pre-classificati, $Tr = \{(x^{(i)}, f(x^{(i)}))\}$, apprendere una descrizione generale che incapsula l'informazione contenuta negli esempi (regole valide su tutto il dominio di ingresso)
- tale descrizione deve poter essere usata in modo predittivo (dato un nuovo ingresso $\tilde{x}$ predire l'output associato $f(\tilde{x})$)
- si assume che un esperto (o maestro) ci fornisca la supervisone (cioè i valori della $f()$ per le istanze x dell'insieme di apprendimento)

Esempio di applicazione: classificazione di caratteri manoscritti

Principali Paradigmi di Apprendimento

Apprendimento Non-supervisionato:

- dato in insieme di esempi $Tr = \{x^{(i)}\}$, estrarre regolarità e/o pattern (valide(i) su tutto il dominio di ingresso)
- non esiste nessun esperto (o maestro) che ci fornisca un aiuto
  - Clustering



  - Scoperta di Regole (Discovery)

Esempio di applicazione: data mining su database strutturati

Principali Paradigmi di Apprendimento

Apprendimento con Rinforzo:

- Sono dati:
  - agente (intelligente ?), che può
    - trovarsi in uno stato s, ed
    - eseguire una azione a (all'interno delle azioni possibili nello stato corrente)
  - ed opera in un ambiente e, che applicando una azione a nello stato s restituisce
    - lo stato successivo, e
    - una ricompensa r, che può essere positiva (+), negativa (-), o neutra (0).
- Scopo dell'agente è quello di massimizzare una funzione delle ricompense (es. ricompensa scontata: $\sum_{t=0}^{\infty} \gamma^t r_{t+1}$ dove $0 \leq \gamma < 1$)

Esempio di applicazione: navigare sul Web alla ricerca di informazione focalizzata

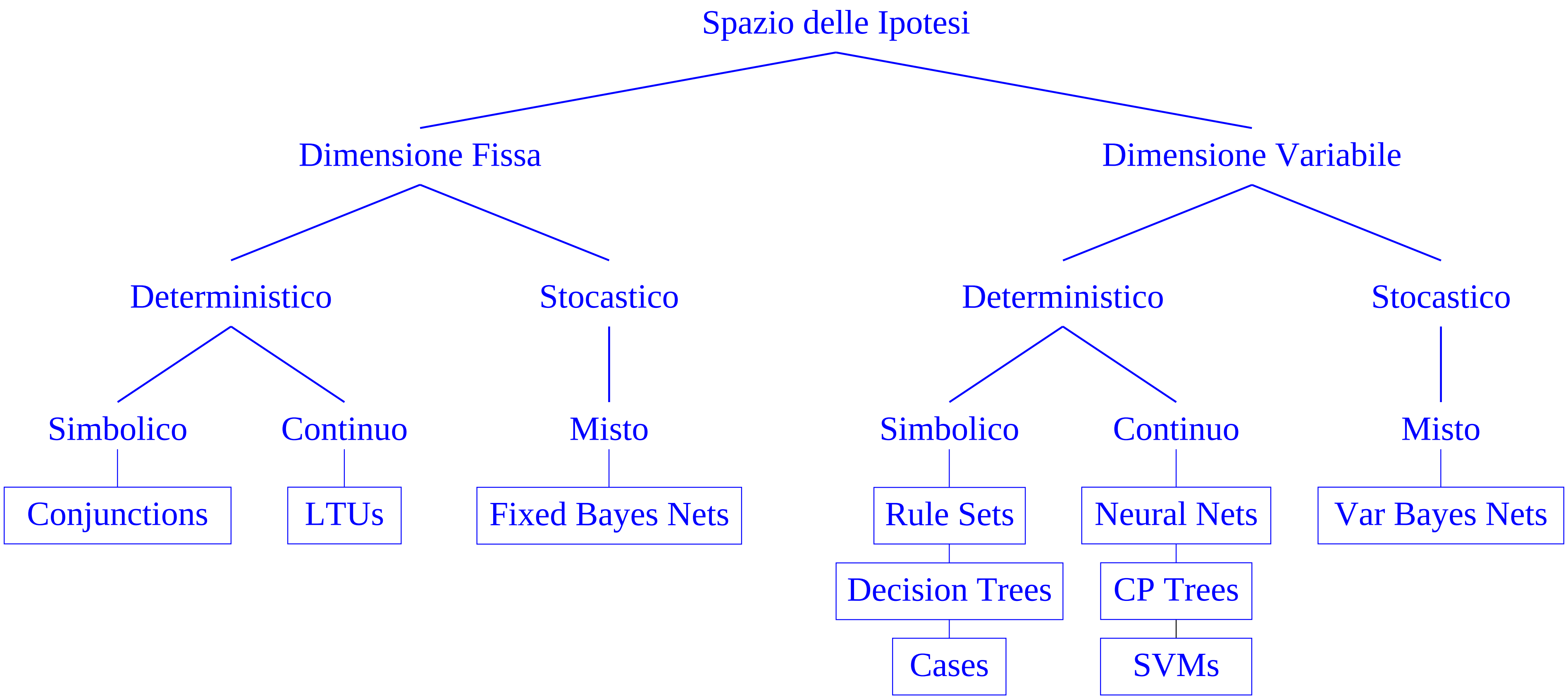
Ingredienti Fondamentali

- Dati di Allenamento
- Spazio delle Ipotesi, $\mathcal{H}$
  - costituisce l'insieme delle funzioni che possono essere realizzate dal sistema di apprendimento;
  - si assume che la funzione da apprendere f possa essere rappresentata da una ipotesi $h \in \mathcal{H}$... (selezione di h attraverso i dati di apprendimento)
  - o che almeno una ipotesi $h \in \mathcal{H}$ sia simile a f (approssimazione);
- Algoritmo di Ricerca nello Spazio delle Ipotesi, alg. di apprendimento

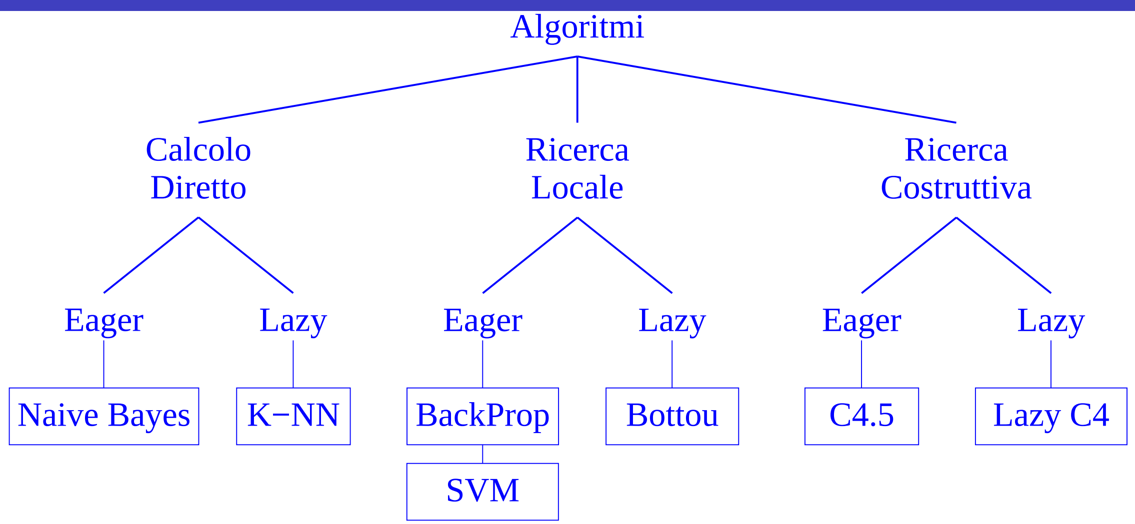
ATTENZIONE: $\mathcal{H}$ non può coincidere con l'insieme di tutte le funzioni possibili e la ricerca essere esaustiva $\rightarrow$ Apprendimento è inutile!!!

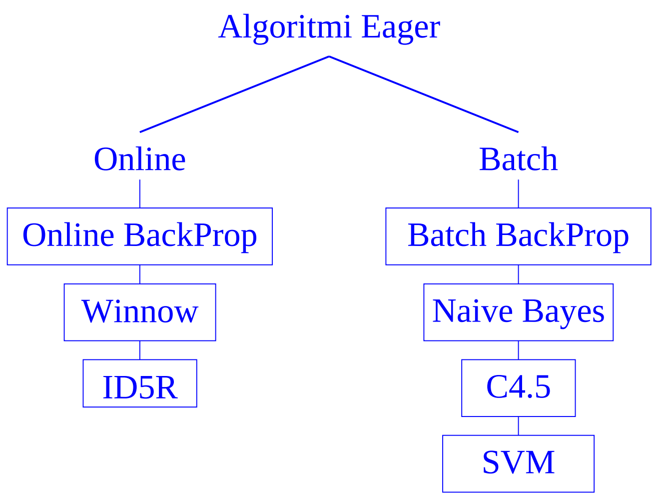
Si parla di **Bias Induttivo**: sulla rappresentazione ($\mathcal{H}$) e/o sulla ricerca (alg. di apprendimento)

Tassonomia (non completa) dello Spazio delle Ipotesi



Tassonomia (non completa) degli Algoritmi di Apprendimento

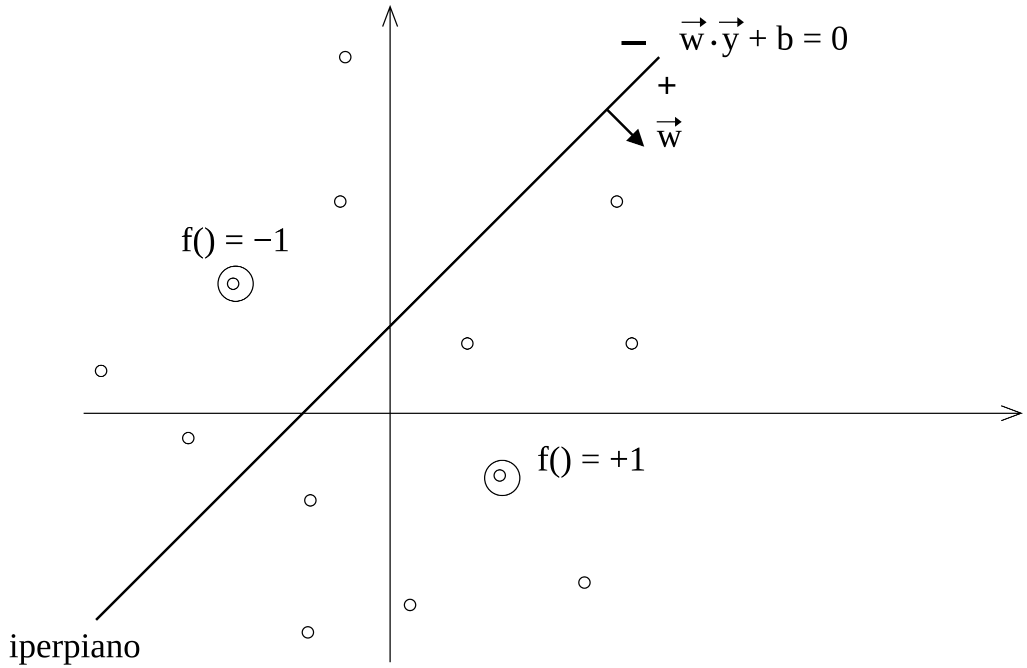




Spazio delle Ipotesi: Esempio 1

Iperpiani in $\mathbb{R}^2$

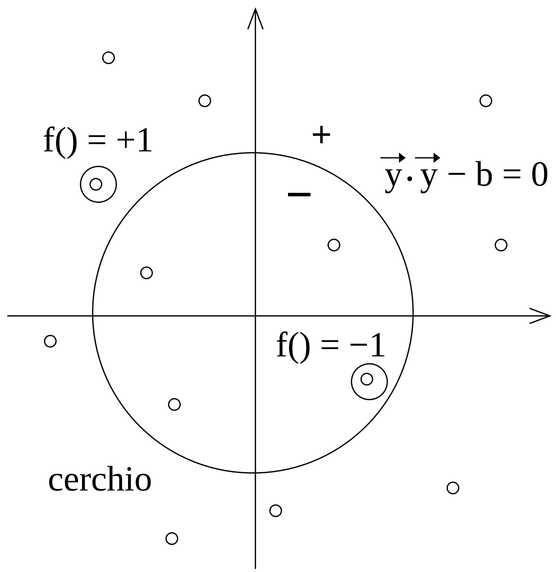
- Spazio delle Istanze $\rightarrow$ punti nel piano: $X = \{\vec{y} \in \mathbb{R}^2\}$
- Spazio delle Ipotesi $\rightarrow$ dicotomie indotte da iperpiani in $\mathbb{R}^2$:
  $\mathcal{H} = \{f_{(\vec{w},b)}(\vec{y}) | f_{(\vec{w},b)}(\vec{y}) = sign(\vec{w} \cdot \vec{y} + b), \vec{w} \in \mathbb{R}^2, b \in \mathbb{R}\}$



Spazio delle Ipotesi: Esempio 2

Dischi in $\mathbb{R}^2$

- Spazio delle Istanze $\longrightarrow$ punti nel piano: $X = \{\vec{y} \in \mathbb{R}^2\}$
- Spazio delle Ipotesi $\longrightarrow$ dicotomie indotte da dischi in $\mathbb{R}^2$ centrati nell'origine:
  $\mathcal{H} = \{f_b(\vec{y}) | f_b(\vec{y}) = sign(\vec{y} \cdot \vec{y} - b), b \in \mathbb{R}\}$



Spazio delle Ipotesi: Esempio 3

Congiunzione di m letterali positivi

- Spazio delle Istanze $\longrightarrow$ stringhe di m bit: $X = \{s | s \in \{0,1\}^m\}$
- Spazio delle Ipotesi $\longrightarrow$ tutte le sentenze logiche che riguardano i letterali positivi $l_1, \ldots, l_m$ (l_1 è vero se il primo bit vale 1, l_2 è vero se il secondo bit vale 1, etc.) e che contengono solo l'operatore $\wedge$ (**and**):

$$\mathcal{H} = \{f_{\{i_1,\ldots,i_j\}}(s) | f_{\{i_1,\ldots,i_j\}}(s) \equiv l_{i_1} \wedge l_{i_2} \wedge \cdots \wedge l_{i_j}, \{i_1, \ldots, i_j\} \subseteq \{1, \ldots, m\}\}$$

Es. $m = 3$, $X = \{0,1\}^3$
Esempi di istanze $\rightarrow$ $s1 = 101$, $s2 = 001$, $s3 = 100$, $s4 = 111$
Esempi di ipotesi $\rightarrow$ $h_1 \equiv l_2$, $h_2 \equiv l_1 \wedge l_2$, $h_3 \equiv true$, $h_4 \equiv l_1 \wedge l_3$, $h_5 \equiv l_1 \wedge l_2 \wedge l_3$
Notare che: h_1, h_2, e h_5 sono false per $s1$, $s2$ e $s3$ e vere per $s4$; h_3 è vera per ogni istanza; h_4 è vera per $s1$ e $s4$ ma falsa per $s2$ e $s3$

Spazio delle Ipotesi: Esempio 3

Congiunzione di m letterali positivi

- Domanda 1: quante e quali sono le ipotesi distinte nel caso $m = 3$?
- Domanda 2: quante sono le ipotesi distinte nel caso generale m ?

Spazio delle Ipotesi: Esempio 3

Congiunzione di m letterali positivi

- Domanda 1: quante e quali sono le ipotesi distinte nel caso $m = 3$?
  - Ris.(quali): $true, l_1, l_2, l_3, l_1 \wedge l_2, l_1 \wedge l_3, l_2 \wedge l_3, l_1 \wedge l_2 \wedge l_3$
  - Ris.(quante): 8
- Domanda 2: quante sono le ipotesi distinte nel caso generale m ?

Spazio delle Ipotesi: Esempio 3

Congiunzione di m letterali positivi

- Domanda 1: quante e quali sono le ipotesi distinte nel caso $m = 3$?
  - Ris.(quali): $true, l_1, l_2, l_3, l_1 \wedge l_2, l_1 \wedge l_3, l_2 \wedge l_3, l_1 \wedge l_2 \wedge l_3$
  - Ris.(quante): 8
- Domanda 2: quante sono le ipotesi distinte nel caso generale m ?
  - Ris.: 2^m, infatti per ogni possibile bit della stringa in ingresso il corrispondente letterale può apparire o meno nella formula logica, quindi:

$$\underbrace{2 \cdot 2 \cdot 2 \cdots 2}_{m \text{ volte}} = 2^m$$

Spazio delle Ipotesi: Esempio 4

Congiunzione di m letterali

- Spazio delle Istanze $\longrightarrow$ stringhe di m bit: $X = \{s | s \in \{0,1\}^m\}$
- Spazio delle Ipotesi $\longrightarrow$ tutte le sentenze logiche che riguardano i letterali $l_1, \ldots, l_m$ (anche in forma negata, $\neg l_i$) e che contengono solo l'operatore $\wedge$ (**and**):

$$\mathcal{H} = \{f_{\{i_1,\ldots,i_j\}}(s) | f_{\{i_1,\ldots,i_j\}}(s) \equiv L_{i_1} \wedge L_{i_2} \wedge \cdots \wedge L_{i_j}, \text{ dove } L_{i_k} = l_{i_k} \text{ oppure } \neg l_{i_k}, \{i_1,\ldots,i_j\} \subseteq \{1,\ldots,2m\}\}$$

Notare che se in una formula un letterale compare sia affermato che negato, allora la formula ha sempre valore di verità $false$ (formula non soddisfacibile)

Quindi, tutte le formule che contengono almeno un letterale sia affermato che negato sono equivalenti alla funzione che vale sempre $false$

Spazio delle Ipotesi: Esempio 4

Congiunzione di m letterali

Es. $m = 3$, $X = \{0, 1\}^3$

Esempi di istanze $\rightarrow$ $s1 = 101$, $s2 = 001$, $s3 = 100$, $s4 = 111$, $s5 = 000$

Esempi di ipotesi $\rightarrow$ $h_1 \equiv \neg l_2$, $h_2 \equiv \neg l_1 \wedge l_3$, $h_3 \equiv true$, $h_4 \equiv \neg l_1 \wedge \neg l_2 \wedge \neg l_3$

Notare che:

- h_1, è falsa per $s4$, e vera per $s1$, $s2$, $s3$ e $s5$;
- h_2 è falsa per $s1$, $s3$, $s4$ e $s5$ e vera per $s2$;
- h_3 è vera per ogni istanza;
- h_4 è falsa per $s1$, $s2$, $s3$, $s4$ e vera per $s5$;

Domanda: quante sono le ipotesi distinte nel caso generale m ?

Spazio delle Ipotesi: Esempio 4

Congiunzione di m letterali

Domanda: quante sono le ipotesi distinte nel caso generale m ?

Risposta: considerando che tutte le formule non soddisfacibili sono equivalenti alla funzione sempre falsa, allora non consideriamo formule dove compare un letterale sia affermato che negato.

Quindi, per ogni possibile bit della stringa in ingresso il corrispondente letterale può non apparire nella formula logica o, se appare, è affermato o negato:

$$\underbrace{3 \cdot 3 \cdot 3 \cdots 3}_{m \text{ volte}} = 3^m$$

E considerando la funzione sempre falsa si ha $3^m + 1$

Spazio delle Ipotesi: Esempio 5

Lookup Table

- Spazio delle Istanze $\rightarrow$ stringhe di m bit: $X = \{s | s \in \{0,1\}^m\}$
- Spazio delle Ipotesi $\rightarrow$ tutte le possibili tabelle di verità che mappano istanze di ingresso ai valori $true$ e $false$: $\mathcal{H} = \{f(s) | f : X \rightarrow \{true, false\}\}$

  Es.

l_1	l_2	$\cdots$	l_m	$f(s)$
0	0	$\cdots$	0	1
0	0	$\cdots$	1	0
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
0	1	$\cdots$	0	0
0	1	$\cdots$	1	1
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
1	0	$\cdots$	0	1
1	0	$\cdots$	1	1
1	1	$\cdots$	0	0
1	1	$\cdots$	1	1
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$

Spazio delle Ipotesi: Esempio 5

Congiunzione di m letterali

Domanda: quante sono le ipotesi distinte nel caso generale m ?

Spazio delle Ipotesi: Esempio 5

Congiunzione di m letterali

Domanda: quante sono le ipotesi distinte nel caso generale m ?

Risposta: tramite una tabella è possibile realizzare una qualunque funzione dallo spazio delle istanze nei volori $true$ e $false$.

Il numero di possibili istanze è:

$$\underbrace{2 \cdot 2 \cdot 2 \cdots 2}_{m \text{ volte}} = 2^m$$

e quindi il numero di possibili funzioni realizzabili è: 2^{2^m}

Osservazioni su Esempi 3, 4 e 5

Osservare che negli esempi 3, 4 e 5 lo spazio delle istanze è sempre lo stesso.

Gli spazi delle ipotesi invece (indichiamo con $\mathcal{H}_3$ quello relativo all'esempio 3, etc.) sono diversi e per ogni m fissato vale la seguente relazione: $\boxed{\mathcal{H}_3 \subset \mathcal{H}_4 \subset \mathcal{H}_5}$

Per esempio, dato $m = 3$

- la funzione booleana $f(s)$ che vale vero solo per le istanze 001 e 011 è contenuta in $\mathcal{H}_4$, infatti $f(s) \equiv \neg l_1 \wedge l_3 \in \mathcal{H}_4$, e in $\mathcal{H}_5$(è facile scrivere una tabella per cui la colonna relativa all'output della funzione è 1 solo in corrispondenza delle istanze 001 e 011), ma non in $\mathcal{H}_3$ perché non esiste la possibilità di descrivere $f(s)$ usando una congiunzione di letterali positivi.

- la funzione booleana $f(s)$ che vale vero solo per le istanze 001, 011 e 100 è contenuta in $\mathcal{H}_5$ (si può procedere come sopra), ma non in $\mathcal{H}_3$ e $\mathcal{H}_4$, perché non esiste la possibilità di descrivere $f(s)$ usando una congiunzione di letterali (positivi).

In particolare $\mathcal{H}_5$ coincide con l'insieme di tutte le funzioni booleane su X.

Alcune definizioni per l'apprendimento di concetti

Definizione: Un “concetto” su uno spazio delle istanze X è definito come una funzione booleana su X.

Definizione: Un esempio di un concetto c su uno spazio delle istanze X è definito come una coppia $(x, c(x))$, dove $x \in X$ e si ricorda che $c()$ è una funzione booleana.

Definizione: Sia h una funzione booleana definita su uno spazio delle istanze X. Si dice che h soddisfa $x \in X$ se $h(x) = 1$ ($true$).

Definizione: Sia h una funzione booleana definita su uno spazio delle istanze X ed $(x, c(x))$ un esempio di $c()$. Si dice che h è consistente con l'esempio se $h(x) = c(x)$. Inoltre si dice che h è consistente con un insieme di esempi Tr se h è consistente con ogni esempio in Tr.

Ordinamento parziale sulle ipotesi

Definizione: Siano h_i e h_j funzioni booleane definite su uno spazio delle istanze X. Si dice che h_i è più generale o equivalente di h_j (scritto $h_i \geq_g h_j$) se e solo se

$$(\forall x \in X)[(h_j(x) = 1) \rightarrow (h_i(x) = 1)]$$

Esempi

- $l_1 \geq_g (l_1 \wedge l_2)$
- $l_2 \geq_g (l_1 \wedge l_2)$
- $l_1 \not\geq_g l_2$ e $l_2 \not\geq_g l_1$ (non paragonabili)

Esercizio: apprendimento di congiunzioni di letterali

Algoritmo **Find-S**

/* trova l'ipotesi più specifica consistente con l'insieme di apprendimento */

- input: insieme di apprendimento Tr
- inizializza l'ipotesi corrente h alla ipotesi più specifica
  $h \equiv l_1 \wedge \neg l_1 \wedge l_2 \wedge \neg l_2 \wedge \cdots \wedge l_m \wedge \neg l_m$
- per ogni istanza positiva di apprendimento $(x, true) \in Tr$
  - rimuovi da h ogni letterale (affermato o negato) che non è soddisfatto da x
- restituisci h

Esempio di applicazione: $m = 5$

Esempio (positivo)	ipotesi corrente
	$h_0 \equiv l_1 \wedge \neg l_1 \wedge l_2 \wedge \neg l_2 \wedge l_3 \wedge \neg l_3 \wedge l_4 \wedge \neg l_4 \wedge l_5 \wedge \neg l_5$
1 1 0 1 0	$h_1 \equiv l_1 \wedge l_2 \wedge \neg l_3 \wedge l_4 \wedge \neg l_5$
1 0 0 1 0	$h_2 \equiv l_1 \wedge \neg l_3 \wedge l_4 \wedge \neg l_5$
1 0 1 1 0	$h_3 \equiv l_1 \wedge l_4 \wedge \neg l_5$
1 0 1 0 0	$h_4 \equiv l_1 \wedge \neg l_5$
0 0 1 0 0	$h_5 \equiv \neg l_5$

Notare che $\boxed{h_0 \leq_g h_1 \leq_g h_2 \leq_g h_3 \leq_g h_4 \leq_g h_5}$

Inoltre, ad ogni passo si sostituisce h_i con un' ipotesi h_{i+1} che costituisce una *generalizzazione minima* di h_i consistente con l'esempio corrente.

Quindi **Find-S** restituisce l'ipotesi pi ̀u specifica consistente con Tr

Osservazioni su **Find-S**

Find-S è in effetti uno schema di algoritmo che si può applicare a spazi di istanze e ipotesi diversi da quelli visti.

L'idea base dell'algoritmo è quella di effettuare una *generalizzazione minima* dell'ipotesi corrente quando questa non è più consistente con l'esempio corrente.

Notare che ogni volta che l'ipotesi corrente h viene *generalizzata* ad una nuova ipotesi h' ($h' \geq_g h$), tutti gli esempi positivi visti in precedenza continuano ad essere soddisfatti dalla nuova ipotesi h' (infatti, poiché $h' \geq_g h$, si ha che $\forall x \in X, \ (h(x) = 1) \rightarrow (h'(x) = 1)$)

Infi ne, se il concetto da apprendereè contenuto in $\mathcal{H}$, tutti gli eventuali esempi negativi sono soddisfatti automaticamente dalla ipotesi restituita da **Find-S** in quanto questa è l'ipotesi consistente più specifi ca, cioè quella che attribuisce il minor numero possibile di 1 alle istanze in X.

Esiste un motivo per preferire l'ipotesi consistente pi`u specifica ?

Principali Paradigmi di Apprendimento: Richiamo

Apprendimento Supervisionato:

- dato in insieme di esempi pre-classificati, $Tr = \{(x^{(i)}, f(x^{(i)}))\}$, apprendere una descrizione generale che incapsula l'informazione contenuta negli esempi (regole valide su tutto il dominio di ingresso)
- tale descrizione deve poter essere usata in modo predittivo (dato un nuovo ingresso $\tilde{x}$ predire l'output associato $f(\tilde{x})$)
- si assume che un esperto (o maestro) ci fornisca la supervisone (cioè i valori della $f()$ per le istanze x dell'insieme di apprendimento)

Find-S è un algoritmo di apprendimento supervisionato

Dati

Consideriamo il paradigma di Apprendimento Supervisonato

Dati a nostra disposizione (off-line)

$$\text{Dati} = \{(x^{(1)}, f(x^{(1)})), \ldots, (x^{(N)}, f(x^{(N)}))\}$$

Suddivisione tipica $(N = N_{tr} + N_{ts})$:

- $\boxed{\text{Training Set} = \{(x^{(1)}, f(x^{(1)})), \ldots, (x^{(N_{tr})}, f(x^{(N_{tr})}))\}}$
  usato direttamente dall'algoritmo di apprendimento;

- $\boxed{\text{Test Set} = \{(x^{(1)}, f(x^{(1)})), \ldots, (x^{(N_{ts})}, f(x^{(N_{ts})}))\}}$
  usato alla fine dell'apprendimento per stimare la bontà della soluzione.

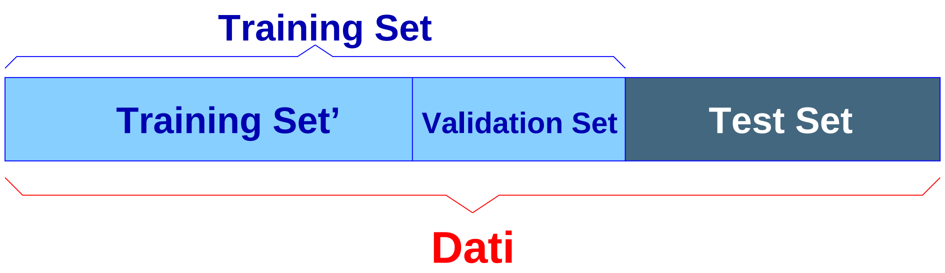
$$\underbrace{\boxed{\textbf{Training Set}}\boxed{\textbf{Test Set}}}_{\textbf{Dati}}$$

Dati (cont.)

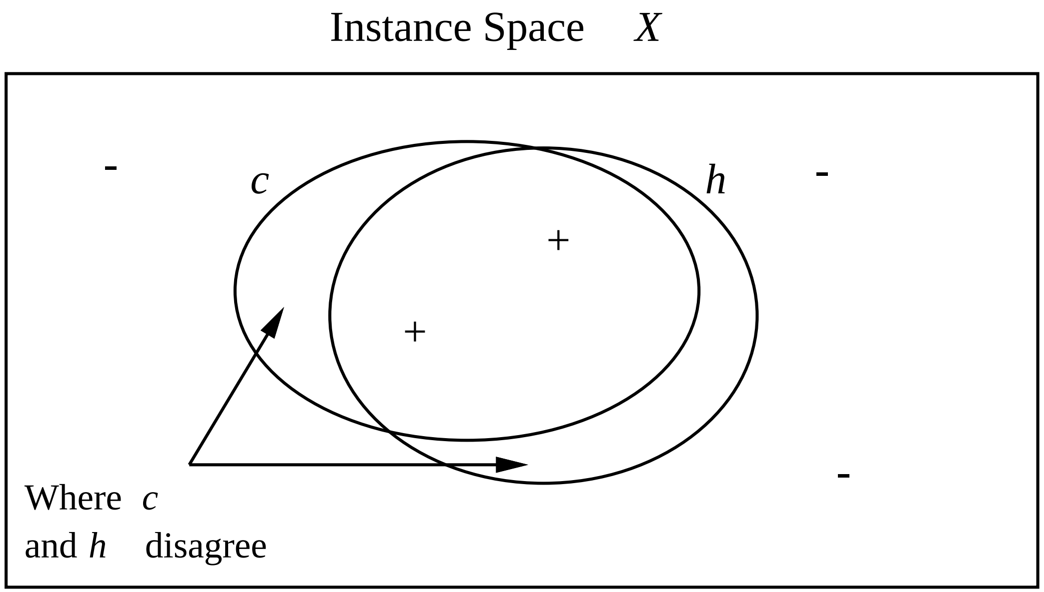
Se N abbastanza grande il Training Set è ulteriormente suddiviso in due sottoinsiemi ($N_{tr} = N_{\widehat{tr}} + N_{val}$) :

- $\boxed{\text{Training Set'} = \{(x^{(1)}, f(x^{(1)})), \ldots, (x^{(N_{\widehat{tr}})}, f(x^{(N_{\widehat{tr}})}))\}}$
  usato direttamente dall'algoritmo di apprendimento;
- $\boxed{\text{Validation Set} = \{(x^{(1)}, f(x^{(1)})), \ldots, (x^{(N_{val})}, f(x^{(N_{val})}))\}}$
  usato indirettamente dall'algoritmo di apprendimento.

Il Validation Set serve per scegliere l'ipotesi $h \in \mathcal{H}$ migliore fra quelle consistenti con il Training Set'



Errore Ideale

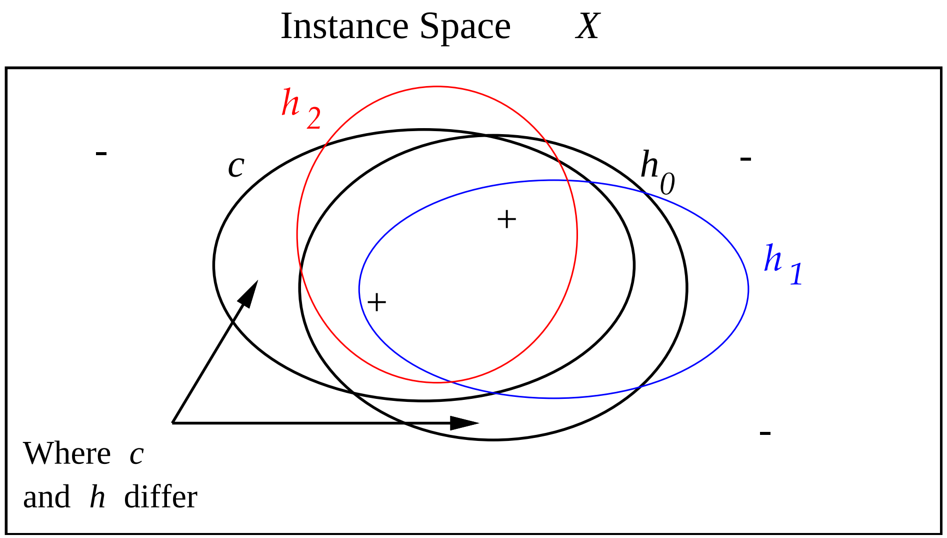


Supponiamo che la funzione f da apprendere sia una funzione booleana (concetto):

$$f : X \to \{0, 1\} \ (\{-, +\})$$

Def: L'**Errore Ideale** ($error_{\mathcal{D}}(h)$) di una ipotesi h rispetto al concetto f e la distribuzione di probabilità $\mathcal{D}$ (probabilità di osservare l'ingresso $x \in X$) è la probabilità che h classifi chi erroneamente un input selezionato a caso secondo $\mathcal{D}$: $error_{\mathcal{D}}(h) \equiv \underset{x \in \mathcal{D}}{Pr} [f(x) \neq h(x)]$

Errore di Apprendimento



Dato Tr = Training Set', più ipotesi possono essere consistenti: h_0, h_1, h_2 quale scegliere ?

Def: L'**Errore Empirico** ($error_{Tr}(h)$) di una ipotesi h rispetto a Tr è il numero di esempi che h classifi ca erroneamente: $error_{Tr}(h) \equiv \#\{(x, f(x)) \in Tr | f(x) \neq h(x)\}$

Def: Una ipotesi $h \in \mathcal{H}$ è sovraspecializzata (**overfit**) Tr se $\exists h' \in \mathcal{H}$ tale che $error_{Tr}(h) < error_{Tr}(h')$, ma $error_{\mathcal{D}}(h) > error_{\mathcal{D}}(h')$.

Il Validation Set serve per cercare di selezionare l'ipotesi migliore (evitare **overfi t**).