



Introduzione al Machine Learning

Laboratorio di Bioinformatica "InfoLife"
Università di Foggia - Consorzio C.IN.I.



Dott. Crescenzo Gallo
crescenzo.gallo@unifg.it

Cos'è il Machine Learning?

- Fa parte dell'Intelligenza Artificiale
- Riguarda particolari attività, senza l'obiettivo di costruire automi intelligenti
- È relativa alla progettazione di sistemi che migliorano (o almeno cambiano) man mano che acquisiscono conoscenza o esperienza



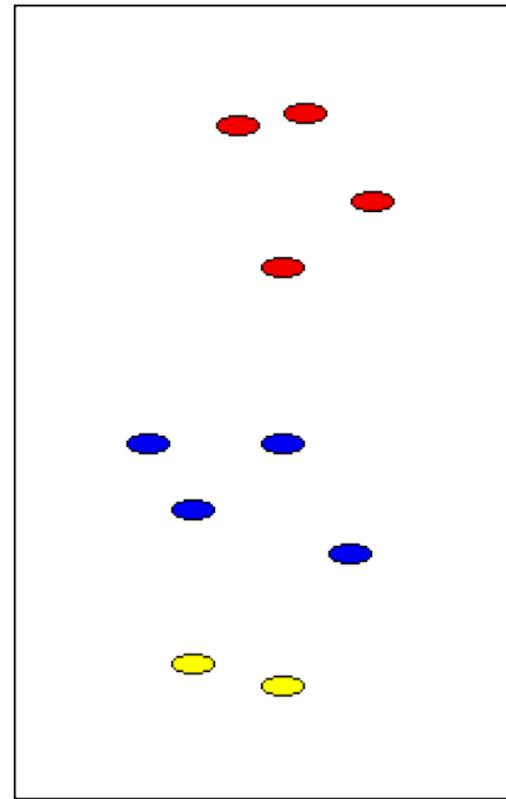
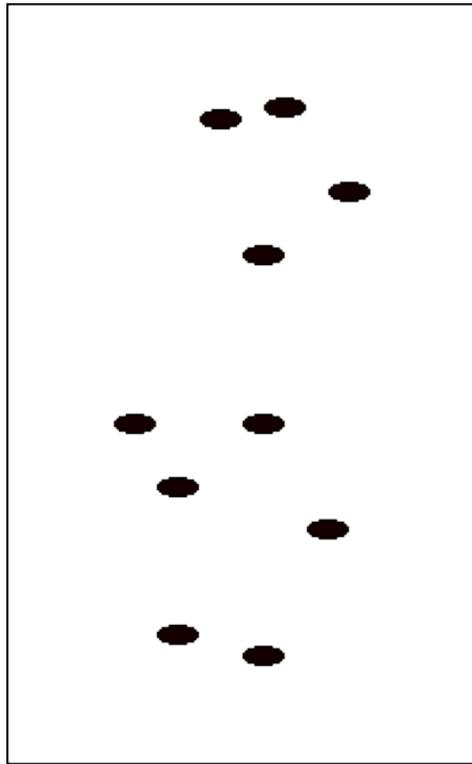
Attività tipiche del ML

- Clustering
- Classificazione
- Categorizzazione
- Filtro/Selezione
- Riconoscimento (*pattern recognition*)
- Simulazione di giochi
- Comportamento autonomo



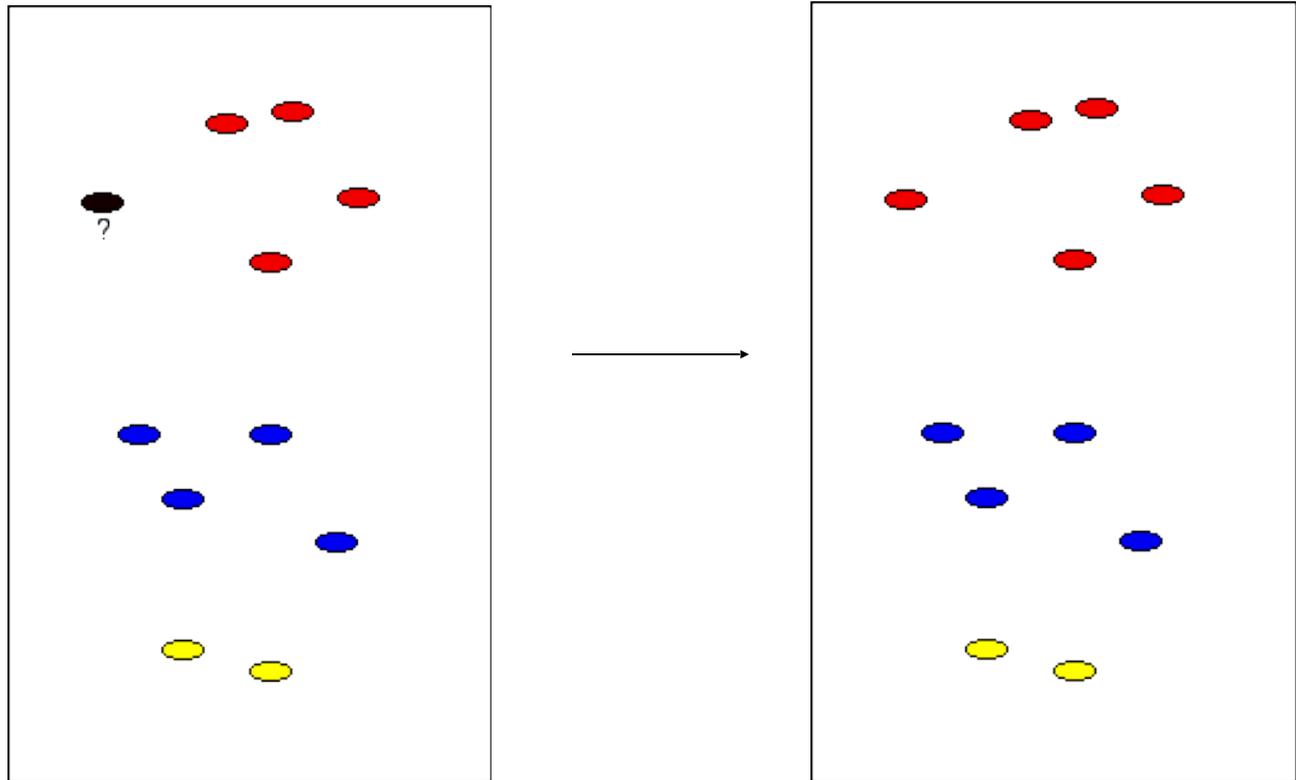
Attività tipiche del ML

- Clustering



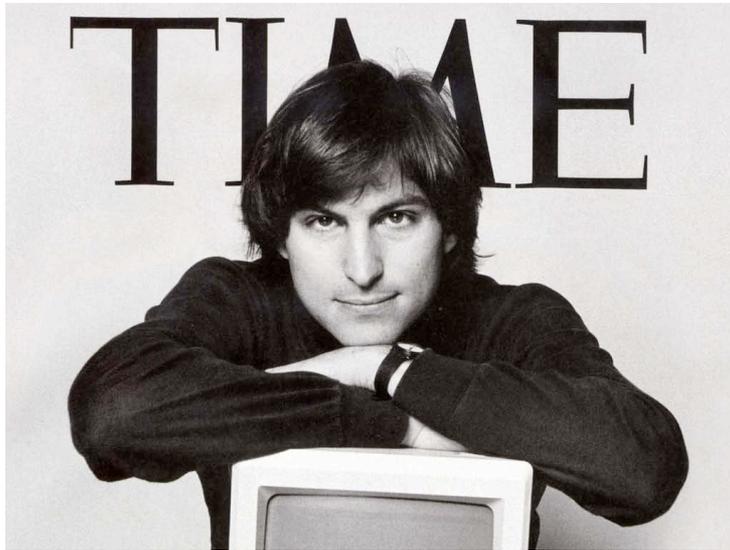
Attività tipiche del ML

- Classificazione/Categorizzazione



Attività tipiche del ML

- Riconoscimento (di immagini)



Vincent Van Gogh

Joe Di Maggio

Mohammed Ali

→ Steve Jobs

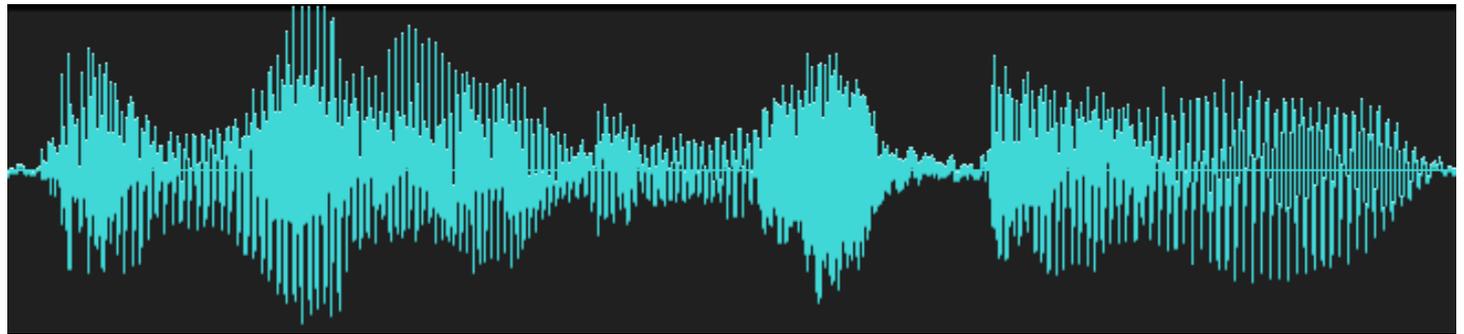
Bill Gates

Winston Churchill

Alfred Hitchcock

Attività tipiche del ML

- Riconoscimento (di suoni)



Mr Tambourine man

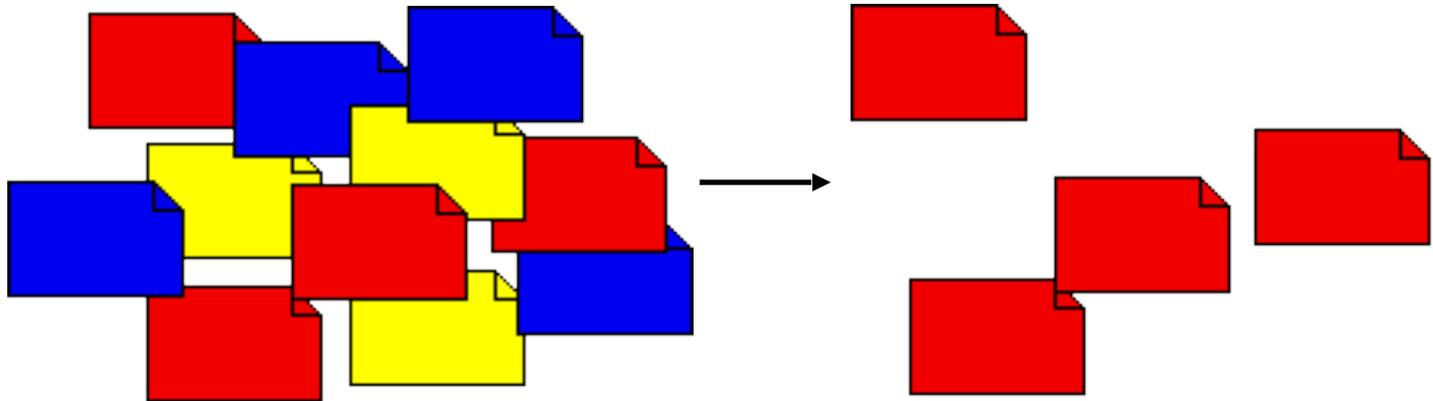
Teach your children

La Canzone del padre

Il suonatore Jones

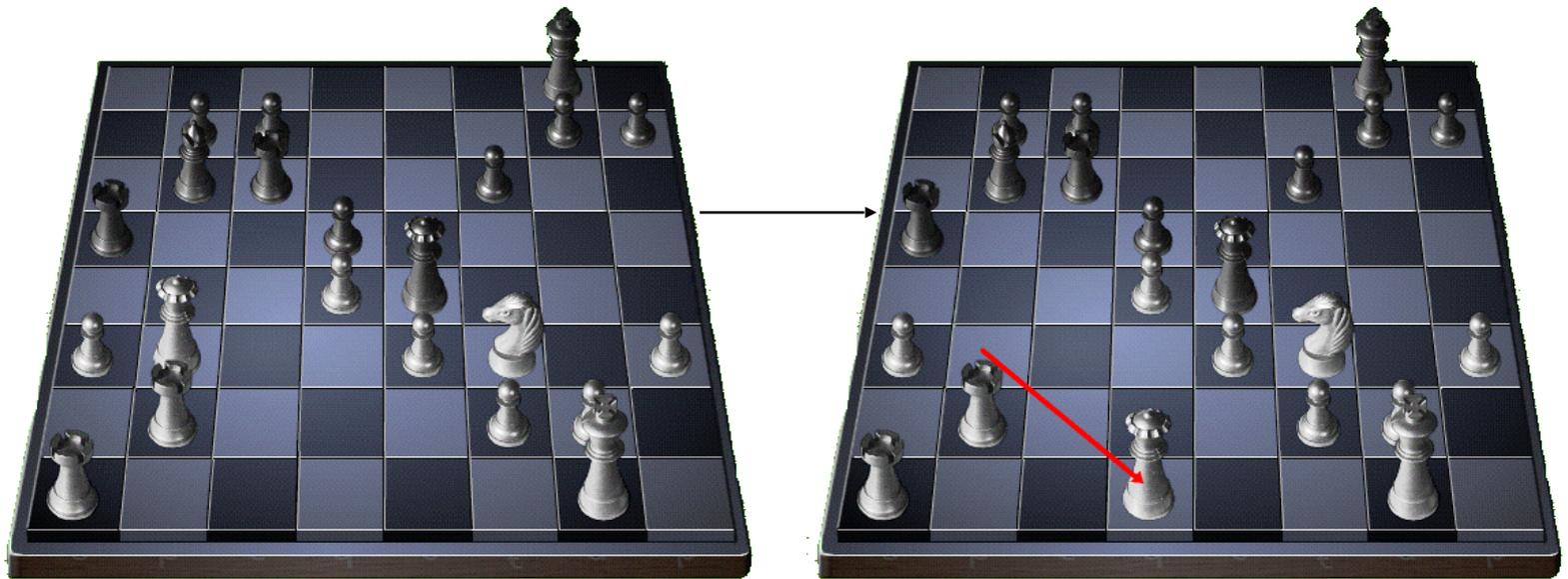
Attività tipiche del ML

- Filtro/Selezione



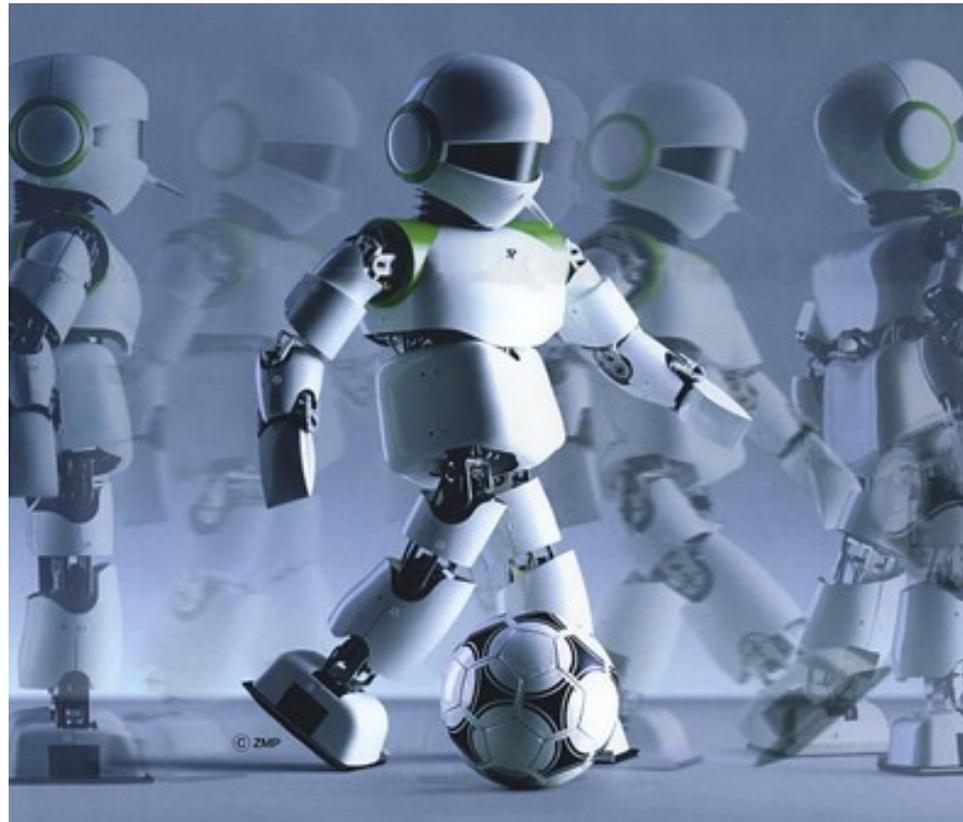
Attività tipiche del ML

- Simulazione di giochi



Attività tipiche del ML

- (Apprendimento e) Comportamento autonomo



Parole chiave del ML

- Data Mining (analisi di dati)
- Knowledge Management (KM – gestione della conoscenza)
- Information Retrieval (IR – ricerca di informazioni)
- Expert Systems (sistemi esperti)
- Text Mining (individuazione e controllo di argomenti in testi e social networks)



Chi si occupa di ML?

- Ricerca e industria
- In stretta sinergia tra di loro
- Non molti moduli ML/KM riutilizzabili, al di fuori di pochi sistemi commerciali
- Il KM è visto come una componente chiave della strategia delle grosse imprese
- Il ML è un settore di ricerca estremamente attivo con basso "costo di ingresso"



Quando è utile il ML?

- Quando vi sono grandi quantità di dati
- Quando non vi sono abbastanza persone per risolvere un problema, o quelle disponibili non possono operare con maggiore rapidità
- Quando ci si può permettere di sbagliare in qualche caso (efficacia vs precisione)
- Quando occorre trovare modelli (*pattern*)
- Quando non si ha più nulla da perdere...



Teoria e Terminologia ML

- Il ML è relativo ad una funzione obiettivo (*target*) collegata ad un set di esempi (casi, istanze)
- La funzione obiettivo è spesso chiamata ipotesi
- Esempio: con una Rete Neurale, una *trained network* (rete addestrata) è un'ipotesi
- L'insieme di tutte le possibili funzioni obiettivo è chiamato lo spazio delle ipotesi
- Il processo di addestramento può essere visto come una ricerca nello spazio delle ipotesi



Teoria e Terminologia ML

- Le tecniche di ML
 - **escludono** alcune ipotesi
 - **preferiscono** alcune ipotesi rispetto ad altre
- Le regole di esclusione e preferenza di una tecnica ML costituiscono la sua capacità (*bias*) di induzione
- Se una tecnica non è "biased" (cioè dotata di *buone* regole), non può imparare e quindi **generalizzare**
- Esempio: i bambini che imparano a fare le moltiplicazioni (comprendere invece che imparare a memoria: quanti danni vengono fatti alle elementari...)



Teoria e Terminologia ML

- Idealmente, una tecnica ML
 - non deve escludere l'ipotesi "corretta", cioè lo spazio delle ipotesi deve includere l'ipotesi obiettivo;
 - preferisce l'ipotesi obiettivo sulle altre.
- Misurare il grado di soddisfacimento di tali criteri è importante e talvolta complicato



Valutazione di Ipotesi

- Spesso la conoscenza della "bontà" di un'ipotesi:
 1. è necessaria per sapere come si comporterà nel mondo reale;
 2. può essere usata per migliorare la tecnica di apprendimento o perfezionare i suoi parametri;
 3. può essere usata da un algoritmo di apprendimento per migliorare automaticamente l'ipotesi.
- Di solito la valutazione viene effettuata su dati di test
 - ▶ i dati di test devono essere separati dai dati usati nella fase di apprendimento;
 - ▶ i dati di test usati sono di solito noti come dati di validazione o hold-out;
 - ▶ i dati di addestramento (training), validazione e test dovrebbero sempre essere diversi e non essere mescolati tra di loro.



Valutazione di Ipotesi

- Alcune misure statistiche utilizzate sono: error rate (tasso di errore), accuracy (accuratezza), precision (precisione, cioè esattezza), recall (sensibilità, cioè completezza), F1
- Vengono calcolate per mezzo di tabelle di contingenza

| | | <i>Reale</i> | |
|------------------------|----|--------------|----------|
| | | Sì | No |
| <i>Cal- colato</i> | Sì | <i>a</i> | <i>b</i> |
| | No | <i>c</i> | <i>d</i> |

a = True Positive (TP)

b = False Positive (FP)

c = False Negative (FN)

d = True Negative (TN)



Valutazione di Ipotesi

| | | <i>Reale</i> | |
|------------------|----|--------------|----------|
| | | Sì | No |
| <i>Calcolato</i> | Sì | <i>a</i> | <i>b</i> |
| | No | <i>c</i> | <i>d</i> |

- Error rate = $(b+c)/(a+b+c+d)$
- Accuracy = $(a+d)/(a+b+c+d)$
- Precision = $p = a/(a+b)$
- Recall = $r = a/(a+c)$
- $F_1 = 2 \cdot p \cdot r / (p+r)$

- Precision = frazione di "veri positivi" tra i casi trovati positivi
- Recall = frazione di "veri positivi" trovati rispetto a tutti i veri positivi
- F_1 combina precision e recall per una valutazione più completa



Valutazione di Ipotesi

| | | <i>Reale</i> | |
|------------------|----|--------------|------|
| | | Sì | No |
| <i>Calcolato</i> | Sì | 2663 | 467 |
| | No | 1081 | 2675 |

- Esempio (di classificazione)
- Si osservi che la *precision* è più alta del *recall* – questo indica un classificatore "cauto"

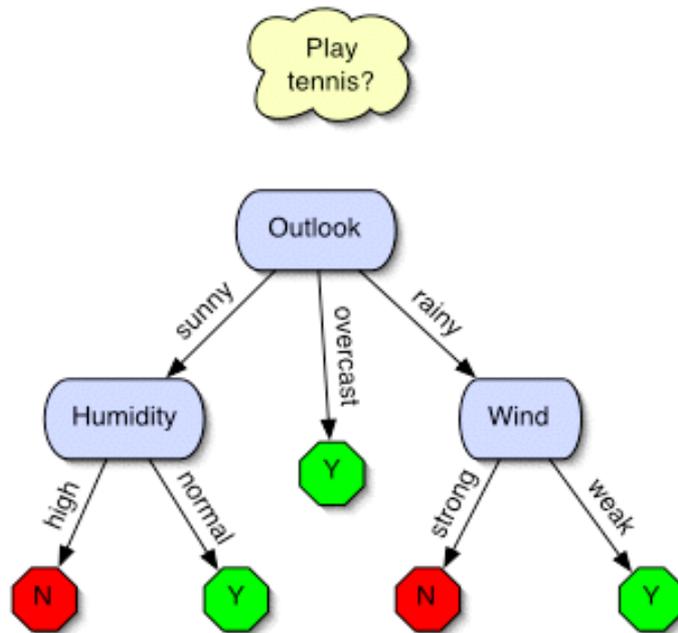
Precision = 0.851, Recall = 0.711, $F_1 = 0.775$

Questi valori dipendono dal tipo di classificazione (e dal numero di classi), e non vanno confrontati con altre classificazioni.

Spesso è utile confrontare le classi separatamente, quindi fare la media (*macro-averaging*).



Alberi di Decisione

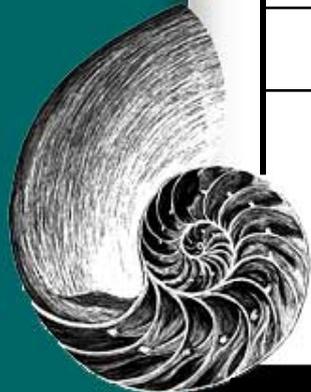


- Concettualmente semplici
- Rapida valutazione
- Strutture di decisione esaminabili
- Possono *apprendere* da dati di training
- Possono essere difficili da costruire
- Possibile "overfit" dei dati di training
- Di solito sono preferibili alberi di decisione più semplici, cioè più *piccoli*

Alberi di Decisione

- Esempio di dati di training (il noto dataset "Play tennis?"):

| Outlook | Temp | Humid | Wind | Play? |
|----------|------|--------|--------|-------|
| Sunny | Hot | High | Weak | No |
| Sunny | Hot | High | Strong | No |
| Overcast | Hot | High | Weak | Yes |
| Rain | Mild | High | Weak | Yes |
| Rain | Cool | Normal | Weak | Yes |
| Rain | Cool | Normal | Strong | No |
| Overcast | Cool | Normal | Strong | Yes |
| | | ... | | |



Alberi di Decisione

- Come costruiamo l'albero dai dati di addestramento?
- Vogliamo ottenere alberi il più piccoli possibile.
- Quale attributo (Outlook, Wind, etc.) è il miglior classificatore?
- Abbiamo bisogno di una misura di quanto un attributo contribuisca al risultato finale della classificazione.
- Si utilizza a tale scopo l'*Information Gain* (IG), che si basa sull'*entropia* dei dati di addestramento.
- L'attributo con il più alto IG è il classificatore "più utile" poichè fornisce la massima riduzione dell'entropia.



Alberi di Decisione

$$Entropy = \sum_i p_i \log_2 \left(\frac{1}{p_i} \right)$$

- Deriva dalla teoria dell'informazione (Information Theory) di Claude Shannon.
- Misura l'incertezza di una decisione tra opzioni alternative.
- Si rappresenta con il valore atteso (in senso probabilistico) del numero di bit necessari per specificare il valore di un attributo.
- i rappresenta il valore di un attributo I , p_i rappresenta la probabilità (frequenza) di quel valore.



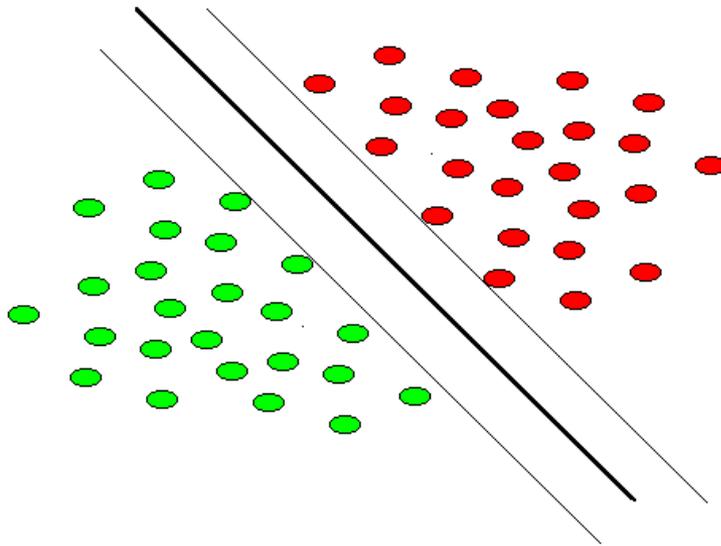
Alberi di Decisione

$$Gain(S, I) = Entropy(S) - \sum_{i \in I} \frac{|S_i|}{|S|} Entropy(S_i)$$

- Gli S_i sono i sottoinsiemi di S in cui l'attributo I ha valore i
- IG è l'entropia originale meno l'entropia dato il valore i
- Ad ogni nodo di decisione (*splitting node*) viene trovato $\operatorname{argmax}_I (Gain(S, I))$
- Per massimizzare IG basta minimizzare il secondo termine sulla destra, essendo $Entropy(S)$ costante
- Questo è noto come "algoritmo ID3" (J. R. Quinlan, 1986)



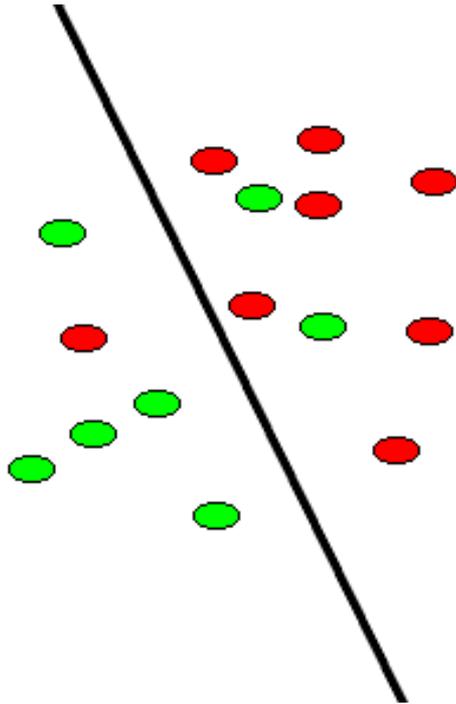
Support Vector Machine (SVM)



- Un'altra tecnica ML
- Misura le feature (attributi) quantitativamente inducendo uno spazio vettoriale
- Trova la *superficie di decisione* ottima



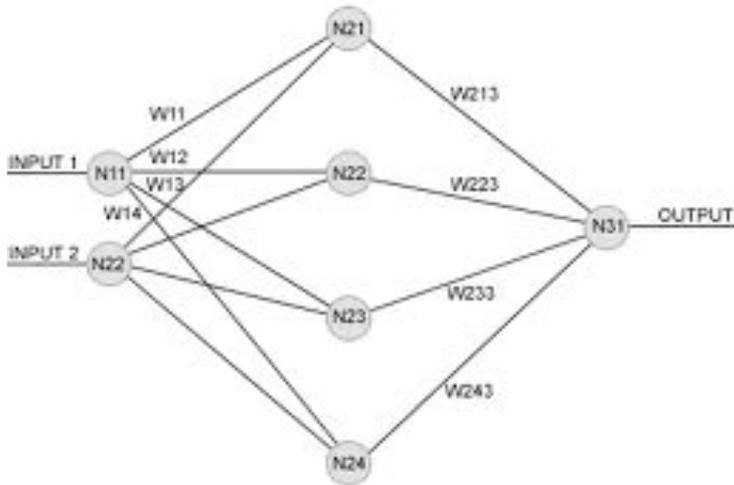
Support Vector Machine (SVM)



- I dati possono non essere separabili
- Gli stessi algoritmi di solito funzionano per trovare la "migliore" superficie di separazione
- Si possono usare differenti forme di superfici
- Di solito scala bene con il numero di feature, non molto bene con il numero di casi



Artificial Neural Network (ANN)



- Evoluzione delle SVM
- Rappresentazione artificiale del cervello umano che cerca di simulare il processo di apprendimento
- Utili per problemi supervisionati (classificazione) e non (clustering)
- "Robuste" e "resistenti" al rumore nei dati
- Deep Neural Network per problemi complessi (Google TensorFlow, ...)

Artificial Neural Network (ANN)

A 3D Kohonen Feature Map



| | |
|------------------------|--------------------------|
| input values: 10 | learning cycle: 0 |
| input layer: 3 neurons | learning rate: 0.6 |
| map size: 9x9 neurons | activation area: 3.0 |
| weights: 243 | elapsed time: 0.0010 sec |

Classificazione di Testi

```
Hello This is Ma
Elem. School in
I am 5 digits l
I am divisible t
My digits add up
When my first ar
get a multiple c
My lowest digit
If you add up m
No number is use
On the left of 1
right decrease t
increase by 3.
...

Thanks
It's that third variable that is confus
General tips for solving problems with r
100 = X.
100 = 10.
I have g
Former
$10, pig:
animal.
oni from Floris
Problem is this
Formula:
(1+x)^n = 1 + nx + (n(n-1))/(1*2)
(n(n-1)...(n-r+1))/(r(r
if n belongs to R. How can I proceed
2) In the expansion of (1+px^2)^7,
Find the value of p. The given an
3) Given that (1+ax+bx^2)^10 = 1 - 3
of the constants a, b. The given
to proceed by taking 10(ax + bx^2
but not b.
10 or
ve t:
```

- La classificazione di testi (Text Categorization), e la classificazione in generale, è una tecnica ML estremamente potente
- Si applica bene a molte aree:
 - Document management
 - Information Retrieval
 - Gene/protein identification
 - Spam filtering
- Concetto abbastanza semplice
- Comporta molte sfide tecniche



Naïve Bayes Categorization

- Semplice e rapida tecnica di Machine Learning
- Siano $c_{1\dots m}$ le categorie (classi), e $w_{1\dots n}$ le parole di un dato documento

$$\text{Best category} = \underset{c_i}{\operatorname{argmax}} p(c_i | w_{1\dots n})$$

Il termine precedente non è computazionalmente fattibile – i dati sono troppo sparsi!



Naïve Bayes Categorization

- Applichiamo il Teorema di Bayes:

$$p(a|b) = \frac{p(a)p(b|a)}{p(b)}$$

$$\text{Best category} = \operatorname{argmax}_{c_i} \frac{p(c_i)p(w_{1\dots n}|c_i)}{p(w_{1\dots n})}$$

$$= \operatorname{argmax}_{c_i} p(c_i)p(w_{1\dots n}|c_i)$$

$$\approx \operatorname{argmax}_{c_i} p(c_i)p(w_1|c_i)p(w_2|c_i)\cdots p(w_n|c_i)$$



Naïve Bayes Categorization

$$\approx \operatorname{argmax}_{c_i} p(c_i) p(w_1|c_i) p(w_2|c_i) \cdots p(w_n|c_i)$$

- Le quantità $p(c_i)$ e $p(w_j|c_i)$ possono essere calcolate dal training set
- $p(c_i)$ è la frazione del training set che appartiene alla categoria c_i
- $p(w_j|c_i)$ è la frazione di parole in c_i che sono w_j
- Occorre trattare le parole non esaminate, non vogliamo che nessun $p(w_j|c_i)$ sia zero
- Tipicalmente si vuole che le parole non esaminate siano state viste 0.5 volte (o strategie simili)



Grazie per l'attenzione!

