

Macchine che leggono

Outline

- 1 Intelligenza Artificiale
- 2 Analisi di Immagini di Documenti
- 3 Alberi di decisione
- 4 Reti neurali artificiali

Intelligenza Artificiale

- Intelligenza Artificiale cerca non solo di **comprendere**, ma anche di **costruire** entità intelligenti
- Test di Turing (1950) formulato per fornire una **definizione operativa** di intelligenza:
 - ▶ Il computer **supera** il test se un intervistatore umano, dopo aver posto delle domande (scritte) non è in grado di dire se l'interlocutore è un umano o un computer
 - ▶ (un aereo vola anche se non somiglia ad un uccello, anzi è ben diverso...)

Test di Turing

- Per superare il test di Turing servono alcune capacità:
 - ▶ **Elaborazione di linguaggio naturale** (Natural Language Processing) → per interagire con il computer
 - ▶ **Rappresentazione della conoscenza** → per **memorizzare** ciò che conosce
 - ▶ **Ragionamento automatico** → per usare l'informazione per rispondere a domande (logica)
 - ▶ **Apprendimento automatico** → per adattarsi a nuove circostanze
- La simulazione fisica di una persona non è necessaria. Se lo fosse si avrebbe il **test di Turing totale**. Per questo servono:
 - ▶ **Visione artificiale** per percepire oggetti/persona
 - ▶ **Robotica** per manipolare oggetti

Apprendimento Automatico

- **Problemi di apprendimento ben definiti.** Si dice che un programma **impara** dall'esperienza E rispetto a qualche scopo S e misura di prestazioni P , se:
la sua prestazione per lo scopo S , misurata da P migliora con l'esperienza E
- Esempio per riconoscimento di caratteri manoscritti
 - ▶ S : riconoscere caratteri manoscritti
 - ▶ E : esempi etichettati
 - ▶ P : percentuale di caratteri riconosciuti
- Ipotesi di **apprendimento induttivo**
Ogni ipotesi che approssimi bene la funzione obiettivo su un insieme sufficientemente grande di esempi di apprendimento approssimerà bene la funzione obiettivo su un insieme di esempi non osservati

Analisi di Immagini di Documenti

Scopo: conversione di un'immagine di un documento in una appropriata forma simbolica

Esempi:

- libri, e riviste
- lettere commerciali
- fatture e documenti bancari

Anche “documenti” come:

- pagine web
- video

I sistemi *DIAR* richiedono l'integrazione di diverse competenze: elaborazioni di immagini, pattern recognition, elaborazione di linguaggio naturale, intelligenza artificiale.

azud	52	bache
<p>azud <i>m</i> o <i>azuda</i> <i>f</i> máquina para sacar agua de los ríos; presa en el río</p> <p>azuela <i>f</i> herramienta de carpintero para desbastar la madera</p> <p>azufalla <i>f</i> fruto del azufallo</p> <p>azufalo <i>m</i> árbol de fruto comestible y dulce (<i>Zizyphus vulgaris</i>)</p> <p>azufar <i>tr</i> impregnar de azufre, sahumar con él</p> <p>azufre <i>m</i> cuerpo simple de color amarillo (símbolo: S; núm. atómico 16; peso atómico 32,066)</p> <p>azufreña <i>f</i> mina de azufre</p>	<p>azul <i>adj</i> <i>m</i> color del cielo sin nubes</p> <p>azulado -<i>da</i> <i>adj</i> azul, que tira a azul</p> <p>azular <i>tr</i> poner azul; teñir de azul</p> <p>azulear <i>intr</i> mostrar el color azul, tirar a azul</p> <p>azulejo <i>m</i> ladrillo pequeño esmaltado azulete <i>m</i> viso azul que se da a la ropa blanca</p> <p>azulino -<i>na</i> <i>adj</i> que tira a azul</p> <p>azumbre <i>m</i> medida para líquidos: 2,13 litros</p> <p>azuzar <i>gtr</i> incitar (a un perro) para que embista; (fam.) irritar, excitar</p>	<p>B</p> <p>B, b / segunda letra del alfabeto</p> <p>B, abir, de Buato y Buato (en exámen)</p> <p>baba <i>f</i> saliva que fluye de la boca; saliva viscosa de ciertos animales; jugo viscoso de ciertas plantas; (Col. y Venez.) especie de calmán</p> <p>balador <i>m</i> lienzo que por limpieza se suspende al cuello de los niños</p> <p>babaza <i>f</i> baba de ciertos animales y plantas; molusco gasterópodo (<i>Limax</i>)</p> <p>babear <i>intr</i> expeler baba; (fam.) embelacarse contemplando a la persona amada</p> <p>Babel <i>m</i> y <i>f</i> ciudad antigua donde los descendientes de No quisieron edificar una torre para alcanzar el cielo; (fam.) lugar de desorden y confusión; (fam.) desorden, confusión babélico -<i>ca</i> <i>adj</i></p> <p>babera <i>f</i> o babero <i>m</i> balador</p> <p>Babia: estar en Babia (fam.) no prestar atención a aquello de que se trata</p> <p>babacea <i>adj</i> y <i>mf</i> (fam.) imbécil, bobo</p> <p>Babilonia <i>f</i> ciudad e imperio a orillas del Eufrates (2000-538 a. de J.C.); cualquier gran ciudad rica y desmoralizada babilónico -<i>ca</i> <i>adj</i> babilónico -<i>na</i> <i>adj</i> y <i>mf</i></p> <p>bable <i>m</i> dialecto asturiano</p> <p>babor <i>m</i> lado izquierdo de la embarcación, mirando a proa</p> <p>babosa <i>f</i> molusco gasterópodo (<i>Limax</i>)</p> <p>babosear <i>tr</i> llenar de babas; <i>intr</i> (fam.) obsequiar a una mujer con demostraciones de amor</p> <p>baboso -<i>na</i> <i>adj</i> que echa muchas babas; sucio, mal aseado; (fam.) demostado obsequioso con personas del otro sexo; <i>m</i> pez decapodéptico (<i>Stomatopoda</i>); <i>f</i> véase babosa</p> <p>babucha <i>f</i> zapato ligero y sin talón</p>

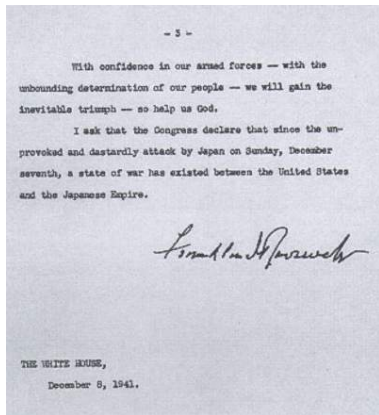
Dictionary

REPRINT and SERIES ORDER FORM				Ship this order to:	
QUANTITY	REPR	YEAR	AUTHOR OF SERIES	PRICE	PLEASE PRINT
			Businessmen and Government Series	\$3.50	NAME
					TITLE
					COMPANY
					STREET
					CITY
					STATE ZIP CODE
					<input type="checkbox"/> PAYMENT ENCLOSED <input type="checkbox"/> BILL COMPANY <input type="checkbox"/> BILL ME

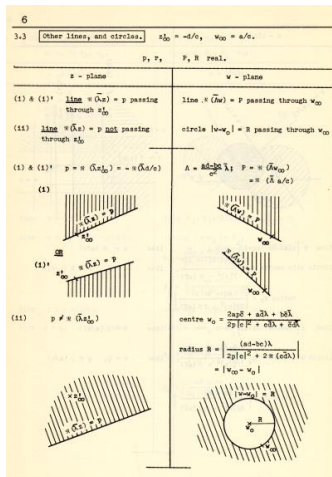
Form



Check



Correspondence



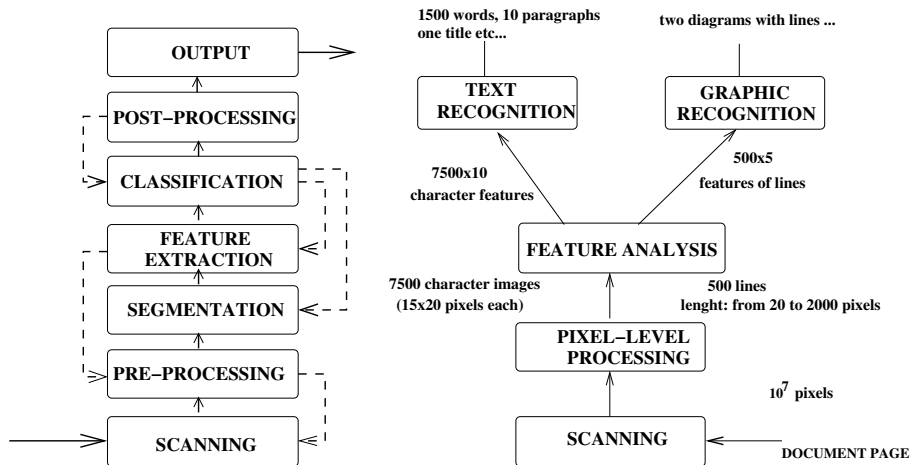
Mathematics

Applicazioni di DIAR

- Smistamento postale
- Lettura di assegni
- Estrazione di informazione da moduli
- Elaborazione di fatture
- Archiviazione automatica di documenti
- Riconoscimento di spartiti musicali
- Analisi di disegni (es. mappe catastali)

Passi di elaborazione in Pattern Recognition

- **Pre-processing:** migliora la qualità delle immagini
- **Segmentazione di oggetti:** identifica oggetti nel documento (es. caratteri, e simboli)
- **Estrazione caratteristiche** (feature): descrivono gli oggetti per classificarli (riconoscerli)
- **Riconoscimento oggetti:** una classe è assegnata ad ogni oggetto
- **Post-processing.** Aggiusta la classificazione con informazione contestuale (es. rispetto ad un dizionario)

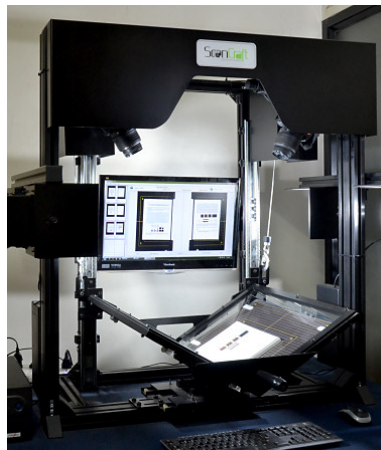


Acquisizione

Permette di ottenere una immagine digitale di un documento cartaceo

- **Fotocamere digitali/cellulari.** In genere bassa risoluzione e immagini soggette a distorsioni non lineari nei bordi
- **Scanner.** Acquisizione è basata sullo spostamento lineare di un array di sensori. Il documento deve essere posto sulla superficie dello scanner
- **Scanner di libri.** Dispositivi complessi (e costosi) usati per ottenere immagini da libri per mezzo di fotocamere digitali dedicate

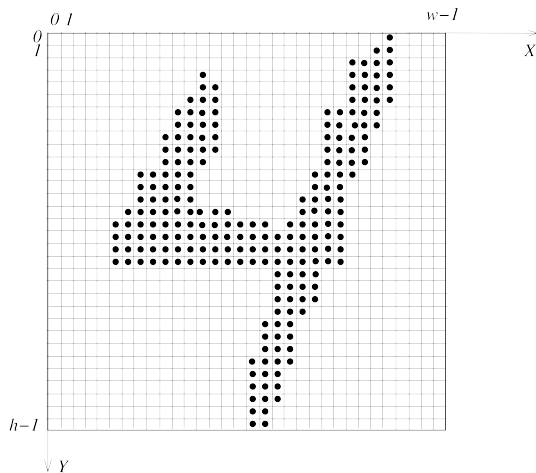
Book scanner



Book scanner



Pixel

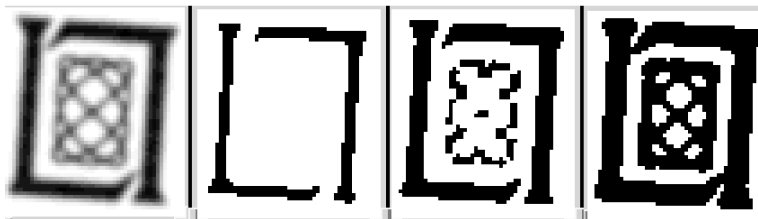


Pre-processing

- Produce una immagine “migliorata” più adatta per analisi successive
- Ad esempio:
 - ▶ Binarizzazione
 - ▶ Riduzione di rumore
 - ▶ Skew detection

Binarizzazione

- I pixel in una immagine a livelli di grigio (8 bit) hanno valori compresi tra 0 (nero) e 255 (bianco)
- Binarizzazione a soglia fissa:
 - ▶ Valori minori della soglia sono posti a 0 gli altri a 255



Segmentazione di immagini

- Eseguita per estrarre oggetti
- Due gruppi principali di approcci:
 - ▶ Metodi basati sul contorno
 - ▶ Metodi che si basano sull'identificazione di regioni omogenee
- Vediamo un metodo del secondo tipo: le componenti connesse

Componenti connesse

- Definite a partire da nozioni topologiche delle immagini digitali
 - **Adiacenza di pixel.** In una griglia rettangolare gli 8 pixel adiacenti ad un pixel **P** sono indicati con 8NN
I pixel più vicini (pixel 0,2,4,6) sono chiamati 4NN

5	6	7
4	P	0
3	2	1

- **Percorso:** una sequenza di pixel P_1, P_2, \dots, P_n tale che P_{k-1} ($k > 1$) e P_{k+1} ($k < n$) sono vicini di P_k

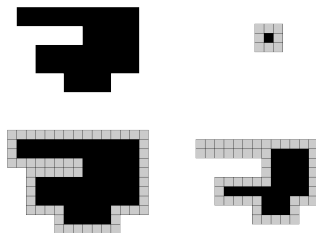
Componenti connesse

- Un insieme di pixel P è **connesso** se, per ogni coppia di pixel A e B in S , esiste un cammino da A a B e tutti i pixel nel cammino sono in S
- Una **componente connessa** è un insieme S di pixel neri connessi con un “path-8” (o un “path-4”)

SCATTI EFFETTUATI DAL 1 DICEMBRE AL 1 FEBBRAIO		SCATTI EFFETTUATI DAL 1 DICEMBRE AL 1 FEBBRAIO	
LETTURA AL 1 FEBBRAIO	529	LETTURA AL 1 FEBBRAIO	529
LETTURA AL 1 DICEMBRE	<u>348</u>	LETTURA AL 1 DICEMBRE	<u>348</u>
TOTALE SCATTI EFFETTUATI	181	TOTALE SCATTI EFFETTUATI	181

Operazioni morfologiche

- Operazioni morfologiche di base: **dilatazione** e **erosione**
- La **dilatazione**, è basata sulla sostituzione di ciascun pixel nero con un *elemento strutturante* nell'immagine trasformata
- L' **erosione** è il processo opposto: ogni occorrenza dell' *elemento strutturante* è rimpiazzata con un pixel nero



Localizzazione parole 1/2

- Usa operazioni morfologiche e componenti connesse
 - 1 Dilatazione con un elemento strutturante orizzontale (larghezza $<$ distanza tra parole)
 - 2 Erosione con un elemento strutturante uguale a quello della dilatazione
 - 3 Le componenti connesse ora corrispondono a parole

Localizzazione parole 2/2

<p>SCATTI EFFETTUATI DAL 1 DICEMBRE AL 1 FEBBRAIO</p> <p>LETTURA AL 1 FEBBRAIO 529</p> <p>LETTURA AL 1 DICEMBRE <u>348</u></p> <p>TOTALE SCATTI EFFETTUATI 181</p>	Immagine
<p>SCATTI EFFETTUATI DAL 1 DICEMBRE AL 1 FEBBRAIO</p> <p>LETTURA AL 1 FEBBRAIO 529</p> <p>LETTURA AL 1 DICEMBRE <u>348</u></p> <p>TOTALE SCATTI EFFETTUATI 181</p>	Componenti C
<p>SCATTI EFFETTUATI DAL 1 DICEMBRE AL 1 FEBBRAIO</p> <p>LETTURA AL 1 FEBBRAIO 529</p> <p>LETTURA AL 1 DICEMBRE <u>348</u></p> <p>TOTALE SCATTI EFFETTUATI 181</p>	Chiusura

Analisi del layout

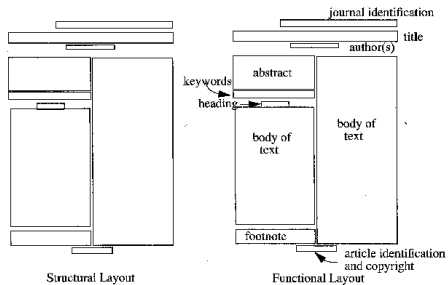
- L'analisi del layout segmenta il documento in regioni con contenuto omogeneo assegnandogli poi un opportuno ruolo (o funzione)
- La segmentazione permette di estrarre la struttura geometrica della pagina
- L'assegnazione di un ruolo logico ad ogni regione è indicato con **analisi del layout logico**

On the Problem of Local Minima in Backpropagation

David D. Lee and John J. Hopfield



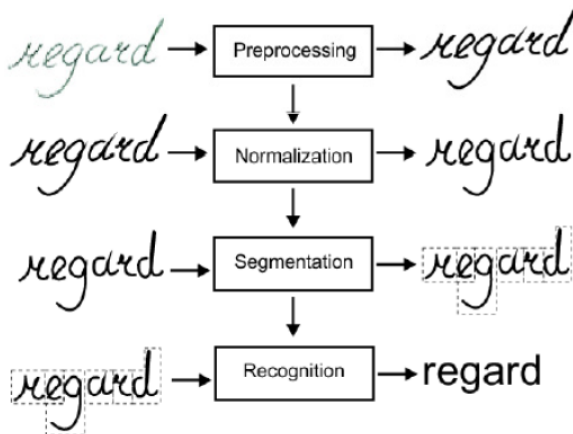
Original Document Page



On-line vs Off-line

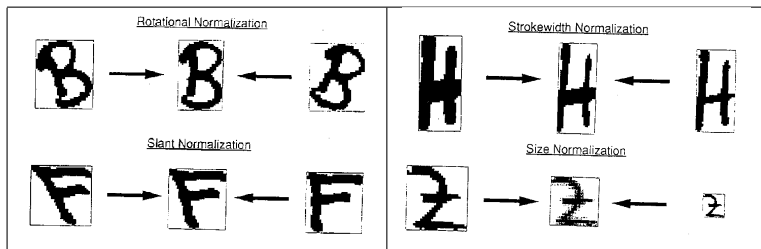
- Programmi di lettura suddivisi in *on-line* e *off-line*
 - ▶ On-line impiega tavolette grafiche o schermi tattili; i caratteri sono acquisiti (ed elaborati) mentre vengono scritti
 - ▶ Sistemi off-line elaborano immagini
- On-line sembra più difficile, ma ha alcuni vantaggi
 - ▶ L'informazione temporale facilita la segmentazione dei caratteri
 - ▶ Vantaggio più sottile: *adattamento dell'utente al sistema*

Riconoscimento parole: passi

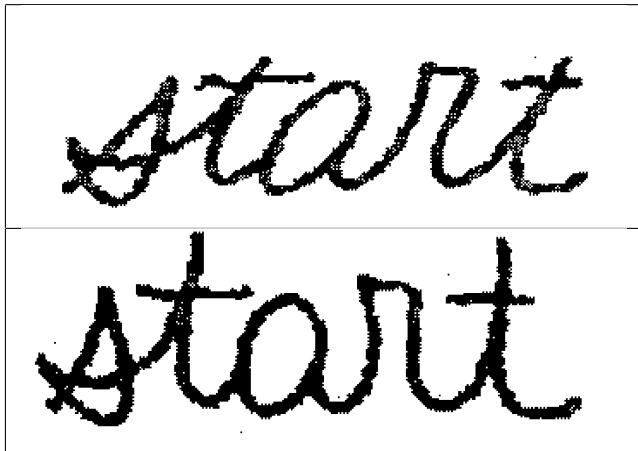


Preprocessing

- Normalizzazione di testo manoscritto prima della segmentazione e del riconoscimento dei caratteri



Preprocessing: correzione slant

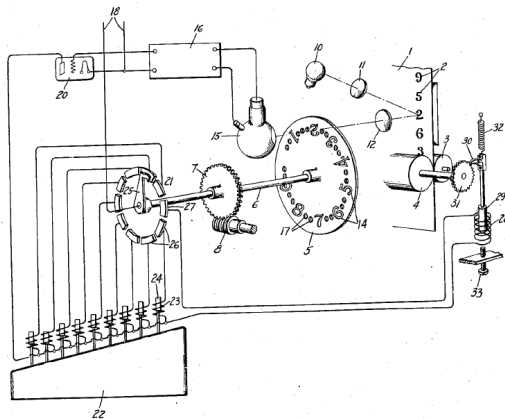


Prime macchine di lettura ...

June 27, 1933.

P. W. HANDEL
STATISTICAL MACHINE
Filed April 27, 1931

1,915,993



Template matching

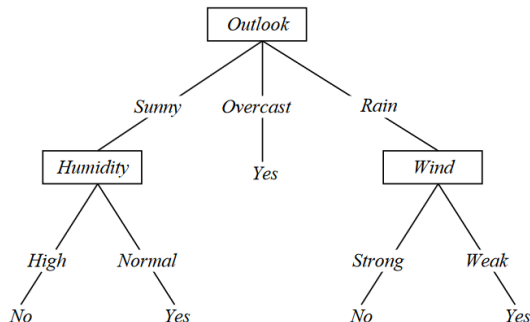
- Riconoscimento con un classificatore a minima distanza: identifica la maschera più simile al carattere incognito
- Problemi: richiede allineamento perfetto tra carattere e maschera; non tollera variazioni di scala e font diversi
- Negli anni 1960 sono stati introdotti alcuni font “facili” da riconoscere

Alberi

- **Alberi** in informatica:
Struttura dati che permette di rappresentare informazione gerarchica
- Esempio: albero genealogico (che in realtà è un grafo)
- Gli alberi hanno radice / nodi interni / foglie
- Vediamo alberi di decisione

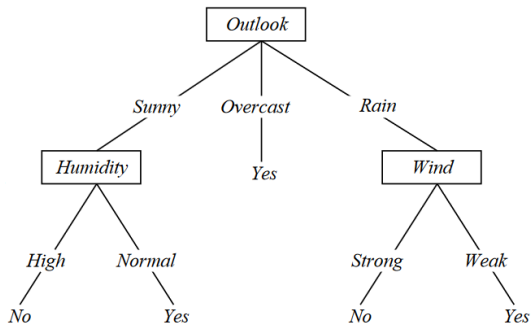
Alberi di decisione

Esempio: albero appreso per decidere se giocare a tennis



- In ogni nodo interno si testa un attributo dell'esempio
- Ogni arco indica un possibile valore per l'attributo
- Ogni foglia rappresenta una classe
- Si classifica un pattern navigando dalla radice a una foglia

- Dal punto di vista della **logica** un albero di decisione è una **disgiunzione** di **congiunzioni** di vincoli su valori di attributi degli esempi



(Previsione = Assolato \wedge Umidità = Normale)

\vee (Previsione = Nuvoloso)

\vee (Previsione = Piovoso \wedge Vento = Debole)

Esempi di apprendimento

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Algoritmo di apprendimento ID3

- Come scegliere il miglior attributo?
- **Information gain**: misura quanto bene un attributo separa gli esempi sulla base della loro classe
- Usa **entropia** (in teoria dell'informazione) che caratterizza l'(im)purità di una collezione arbitraria di esempi
- Dati gli esempi S positivi (+) e negativi (-) si ha:

$$\text{Entropia}(S) = -p_{(+)} \log_2 p_{(+)} - p_{(-)} \log_2 p_{(-)}$$

dove:

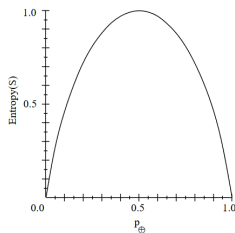
$p_{(+)}$ è la **proporzione di esempi positivi** in S e

$p_{(-)}$ è la **proporzione di esempi negativi** in S

(si definisce $0 \cdot \log_2 0 = 0$)

Entropia: esempi

- Entropia($9(+), 5(-)$) = $-\frac{9}{14} \log_2(\frac{9}{14}) - \frac{5}{14} \log_2(\frac{5}{14}) = 0,94$
- Se tutti gli esempi sono $(+)$ (o $(-)$) \Rightarrow Entropia = 0
- Se si hanno $1/2$ per ogni classe \Rightarrow Entropia = 1



$$Entropy(S) \equiv -p_+ \log_2 p_+ - p_- \log_2 p_-$$

- In **teoria dell'informazione**: numero minimo di bit richiesti per codificare un numero arbitrario di elementi in S

Scelta attributo: Information gain

- Entropia misura l'impurità di una collezione di esempi di apprendimento
- **Information gain**: riduzione attesa di entropia causata dal partizionamento degli esempi sulla base dell'attributo A

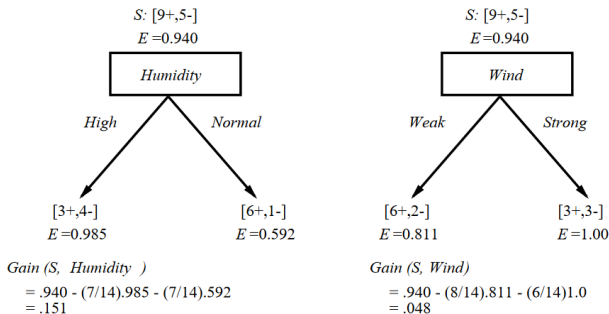
$$Gain(S, A) = Entropia(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropia(S_v)$$

$Values(A)$: insieme di valori di A

$$S_v = \{s \in S | A(s) = v\}$$

- Scegli attributo (poi il suo valore) che massimizza l'information gain

Scelta attributo



- S : dati iniziali con 9(+) e 5(−) e un'entropia di 0,94
- Dividendo con **Umidità** si hanno due collezioni:
 - ▶ 3(+), 4(−) con Umidità = High
 - ▶ 6(+), 1(−) con Umidità = Low
- Information gain maggiore con Umidità rispetto a Wind

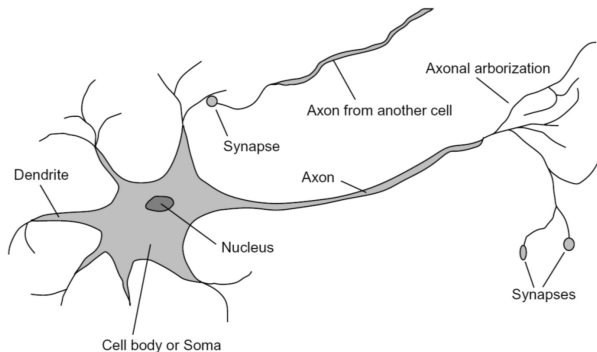
Machine learning

“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with experience E .” Mitchell (1997)

Esempio (classificazione: apprendimento supervisionato)

- T : riconoscere caratteri manoscritti
- E : immagini di cifre con etichetta assegnata
- P : percentuale di caratteri riconosciuti correttamente

Neuroni biologici



Nei neuroni biologici i segnali elettrici sono trasmessi ad altri neuroni tramite gli assoni connessi con le sinapsi agli altri neuroni.

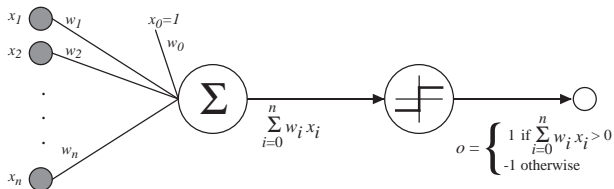
L'azione della sinapsi può essere eccitatoria o inibitoria

Numero di neuroni: 10^{11}

Numero di sinapsi 10^{15}

Modello MacCulloch & Pitt

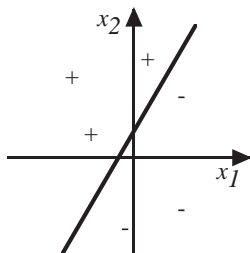
$$O = \text{sgn}\left(\sum_{j=1}^n w_j \cdot u_j - \theta\right) = \begin{cases} 1 & \text{se } w_0 + w_1x_1 + \dots + w_nx_n > 0 \\ -1 & \text{altrimenti.} \end{cases}$$



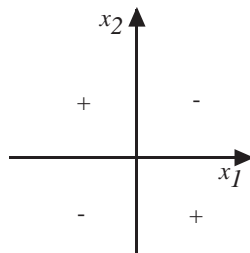
In notazione vettoriale:

$$o(\vec{x}) = \begin{cases} 1 & \text{se } \vec{w} \cdot \vec{x} > 0 \\ -1 & \text{altrimenti} \end{cases}$$

Superficie di decisione di un Perceptron



(a)



(b)

Alcune funzioni non possono essere rappresentate

- Ad esempio non separabili linearmente
- Quindi ci servono delle reti di neuroni...

Regola di apprendimento del perceptron

$$w_i \leftarrow w_i + \Delta w_i$$

con

$$\Delta w_i = \eta(t - o)x_i$$

dove:

- $t = c(\vec{x})$ è il valore obbiettivo (target)
- o è l'uscita del perceptron
- η è una costante piccola (esempio 0.1 o meno) chiamata *learning rate*

Si può provare che converge

- se i dati sono linearmente separabili
- e η è sufficientemente piccolo

Discesa del gradiente

- Consideriamo una semplice *unità lineare*:

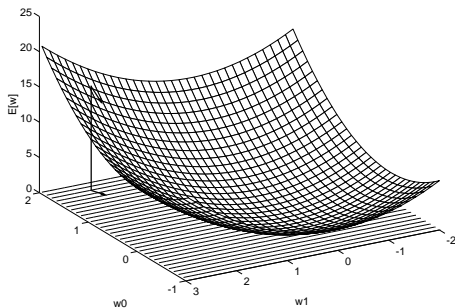
$$o = w_0 + w_1x_1 + \cdots + w_nx_n$$

- Apprendiamo w_i che minimizzino l'errore quadratico medio

$$E[\vec{w}] \equiv \frac{1}{2} \sum_{d \in D} (t_d - o_d)^2$$

- Dove D è l'insieme degli esempi di apprendimento
- Discesa del gradiente : per $f(x)$ la direzione di massima discesa nel punto x corrisponde all'opposto del suo gradiente $\nabla f(x)$
- La soluzione tende a un punto di minimo di f
- In questo caso $f(x) = E[\vec{w}]$

Discesa del gradiente



$$\nabla E[\vec{w}] \equiv \left[\frac{\partial E}{\partial w_0}, \frac{\partial E}{\partial w_1}, \dots, \frac{\partial E}{\partial w_n} \right]$$

Regola di apprendimento: $\Delta \vec{w} = -\eta \nabla E[\vec{w}]$ $\Delta w_i = -\eta \frac{\partial E}{\partial w_i}$

Discesa del gradiente

$$\begin{aligned}\frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2} \sum_d (t_d - o_d)^2 \\&= \frac{1}{2} \sum_d \frac{\partial}{\partial w_i} (t_d - o_d)^2 \\&= \frac{1}{2} \sum_d 2(t_d - o_d) \frac{\partial}{\partial w_i} (t_d - o_d) \\&= \sum_d (t_d - o_d) \frac{\partial}{\partial w_i} (t_d - \vec{w} \cdot \vec{x}_d) \\ \frac{\partial E}{\partial w_i} &= \sum_d (t_d - o_d) (-x_{i,d})\end{aligned}$$

Discesa del gradiente(*training_examples*, η)

Ogni esempio di apprendimento è una coppia del tipo $\langle \vec{x}, t \rangle$, dove \vec{x} è il vettore dei valori di ingresso e t è il valore target di uscita. η è il learning rate (e.g., .05).

- Inizializzare ogni w_i a qualche piccolo valore casuale
- Fino a quando si soddisfa la *condizione di terminazione* :
 - ▶ Inizializzare ogni Δw_i a zero.
 - ▶ Per ogni $\langle \vec{x}, t \rangle$ in *training_examples* , :
 - ★ Mostrare l'esempio \vec{x} in ingresso all'unità e calcolare l'uscita o
 - ★ Per ogni peso dell'unità lineare w_i , :

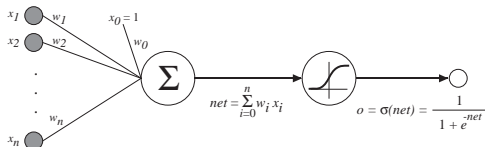
$$\Delta w_i = \Delta w_i + \eta(t - o)x_i$$

- ▶ Per ogni peso dell'unità lineare w_i , :

$$w_i = w_i + \Delta w_i$$

Unità sigmoidea

Funzione matematica che produce una curva sigmoide (con andamento ad "S")



$\sigma(x)$ è la funzione sigmoidea

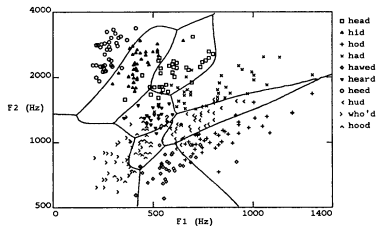
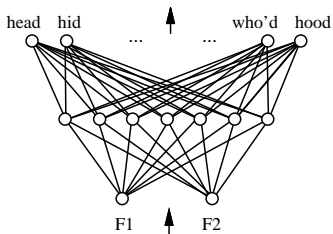
$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Proprietà interessante: $\frac{d\sigma(x)}{dx} = \sigma(x)(1 - \sigma(x))$

Si hanno regole di apprendimento basate sulla discesa del gradiente per:

- Una unità sigmoide
- Reti multistrato (*multilayer networks*) di unità sigmoidi → Backpropagation

Reti multistrato di unità sigmoidee



Errore del Gradiente per una unità

$$\begin{aligned}
 \frac{\partial E}{\partial w_i} &= \frac{\partial}{\partial w_i} \frac{1}{2} \sum_{d \in D} (t_d - o_d)^2 = \frac{1}{2} \sum_d \frac{\partial}{\partial w_i} (t_d - o_d)^2 \\
 &= \frac{1}{2} \sum_d 2(t_d - o_d) \frac{\partial}{\partial w_i} (t_d - o_d) \\
 &= \sum_d (t_d - o_d) \left(-\frac{\partial o_d}{\partial w_i} \right) = - \sum_d (t_d - o_d) \frac{\partial o_d}{\partial net_d} \frac{\partial net_d}{\partial w_i}
 \end{aligned}$$

Nota: $\frac{\partial o_d}{\partial net_d} = \frac{\partial \sigma(net_d)}{\partial net_d} = o_d(1 - o_d)$ e $\frac{\partial net_d}{\partial w_i} = \frac{\partial (\vec{w} \cdot \vec{x}_d)}{\partial w_i} = x_{i,d}$
 allora:

$$\frac{\partial E}{\partial w_i} = - \sum_{d \in D} o_d(1 - o_d)(t_d - o_d)x_{i,d}$$

Algoritmo Backpropagation

Inizializzare tutti i pesi a piccoli valori casuali.

Fino a "convergenza" :

- Per ogni esempio di apprendimento :

- 1 Presentare l'esempio in ingresso alla rete e calcolare le uscite della rete
- 2 Per ogni unità di uscita k :

$$\delta_k \leftarrow o_k(1 - o_k)(t_k - o_k)$$

- 3 Per ogni unità nascosta h :

$$\delta_h \leftarrow o_h(1 - o_h) \sum_{k \in \text{outputs}} w_{h,k} \delta_k$$

- 4 aggiornare ogni peso della rete $w_{i,j}$

$$w_{i,j} \leftarrow w_{i,j} + \Delta w_{i,j}$$

dove: $\Delta w_{i,j} = \eta \delta_j o_i$ ($x_i = o_i$ nell'input)

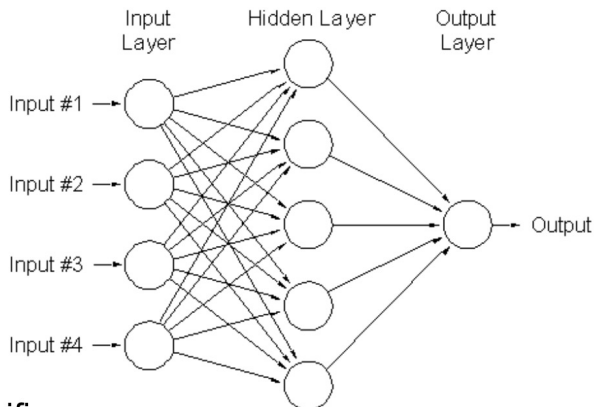
Ancora su Backpropagation

- Trova un minimo locale non necessariamente globale
 - ▶ In pratica spesso funziona bene (magari eseguendolo più volte)
- Spesso si include un *momento* α

$$\Delta w_{i,j}(n) = \eta \delta_j x_{i,j} + \alpha \Delta w_{i,j}(n-1)$$

- Minimizza l'errore su esempi di *training*
 - ▶ Generalizzerà bene su altri esempi?
- **Apprendimento** può richiedere migliaia di interazioni (epoche) → lento!
- **Usare una rete** appresa è molto veloce

MLP: Multi Layer Perceptron



Generalizzazione

- Una rete neurale **generalizza** quando la relazione ingresso-uscita calcolata dalla rete è corretta (o corretta in buona approssimazione) per coppie input-output (**test set**) non usate per addestrare la rete (**learning set**)
- Come può una rete generalizzare a partire da dati di apprendimento?
 - ▶ *Intuizione:* durante l'apprendimento la rete interpola punti nello spazio di ingresso corrispondenti ad esempi positivi
 - ▶ Punti tra esempi positivi vengono considerati positivi

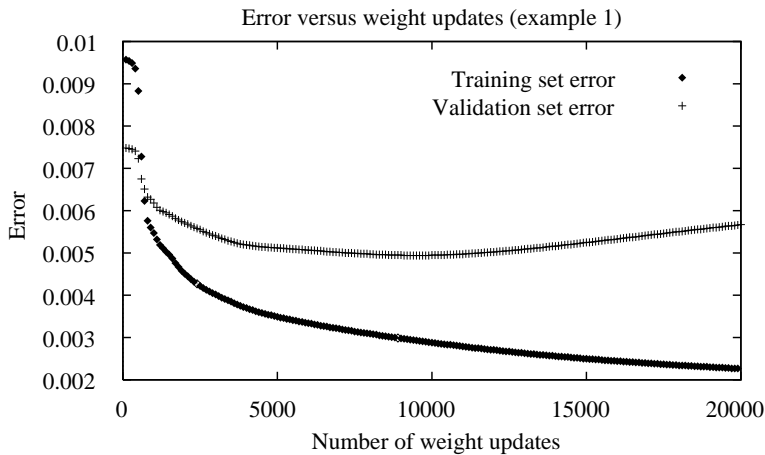
Arrestare il processo di apprendimento

- Due approcci:
 - ▶ Fissare un numero massimo di epoche (N) e salvare la rete dopo N epoche
 - ▶ Fissare un valore massimo per l'errore (target-output) durante l'apprendimento
- Semplici da implementare, ma non c'è relazione con le capacità di generalizzazione della rete appresa
- Apprendimento arrestato troppo presto \Rightarrow la rete non può riconoscere pattern di apprendimento
- Apprendimento arrestato troppo tardi \Rightarrow la rete “impara a memoria” gli esempi di apprendimento (*overfitting*) e non può generalizzare ad esempi sconosciuti

Cross-validation per arrestare l'apprendimento

- Dati divisi in *training set* e *test set*.
Il training set è diviso in due sotto-insiemi:
 - 1 Un insieme è usato per addestrare la rete
 - 2 L'altro sotto insieme (*validation set*) è usato per validare il modello ed arrestare l'apprendimento
- Ogni X epoche si sospende l'addestramento, si salvano i pesi e si valutano i risultati nell'insieme di validazione
- Si ha una iniziale diminuzione dell'errore di validazione (la generalizzazione della rete aumenta)
- Poi l'errore di validazione comincia ad aumentare **anche se l'errore di apprendimento diminuisce**
- Qui inizia l'overfitting

Overfitting



Valutazione prestazioni

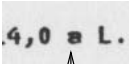
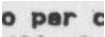
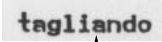


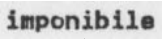


Vari modi:

- Misure globali: error rate
- Matrice di confusione
- Descrizioni qualitative (errori più comuni)

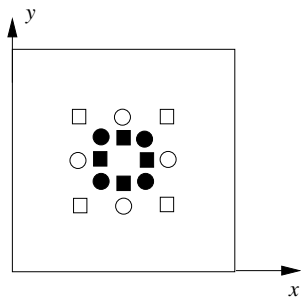
Matrice di confusione

	0	1	2	3	4	5	6	7	8	9	@	Canc
0	873	726	.
1	.	29	1	1399	.
2	.	.	763	190	1
3	.	.	.	771	.	1	204	.
4	632	8	.
5	550	66	.
6	1240	.	.	.	105	.
7	632	.	1	21	.
8	414	.	194	.
9	1507	18	.

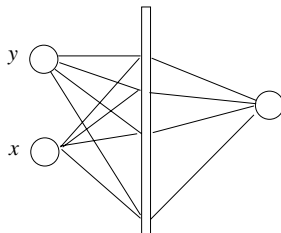
Errori comuni

 ↑ a → e	 ↑ e → a	 ↑ a → m	 ↑ s → a
 ↑ i → l	 ↑ i → l	 ↑ c → o	 ↑ c

Esempio

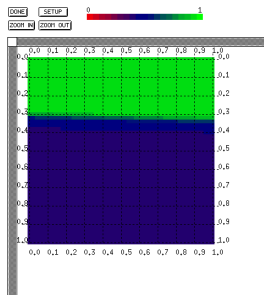


Learning set ●
 ES POSITIVI
 Validation set ■



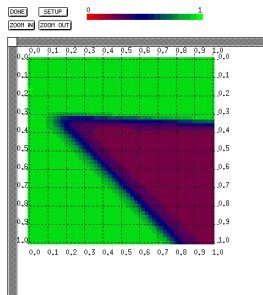
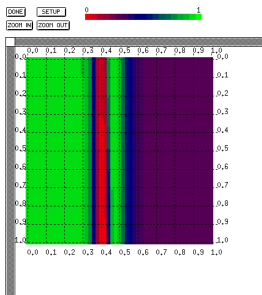
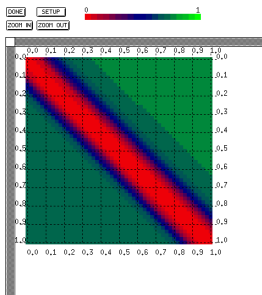
Learning set ○
 ES NEGATIVI
 Validation set □

Un neurone nello strato nascosto

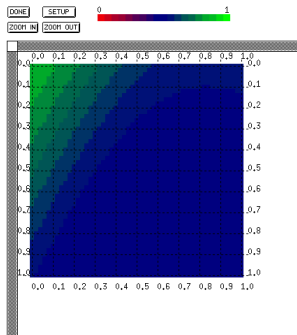
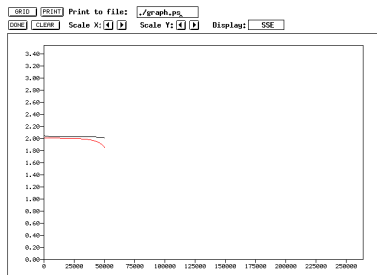


Due neuroni nello strato nascosto

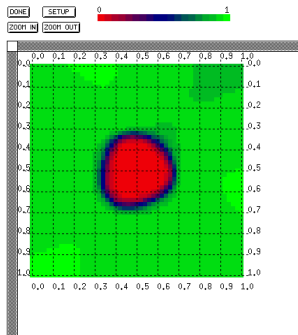
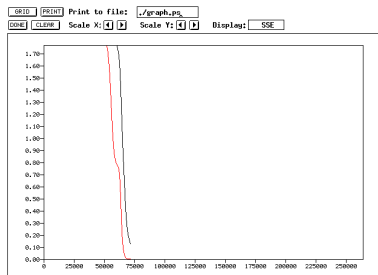
Alcune reti apprese...



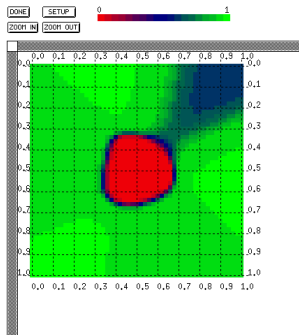
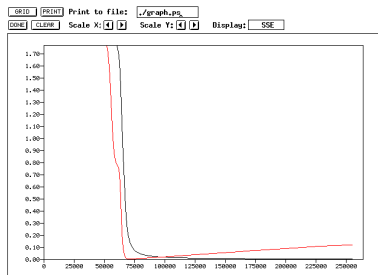
Tre neuroni nello strato nascosto



Tre neuroni nello strato nascosto



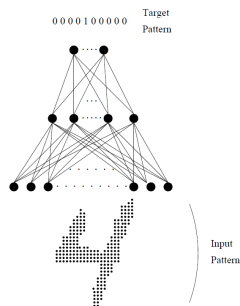
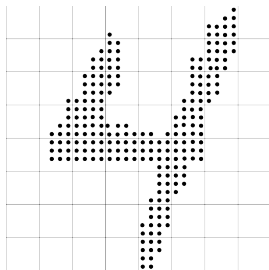
Tre neuroni nello strato nascosto



Un esempio

- Problema: classificare cifre manoscritte
- Descrizioni delle cifre estratte con un semplice programma
 - ① Presenza di un "foro" nell'immagine del carattere (feature booleana)
 - ② Rapporto tra larghezza ed altezza del carattere
- La classificazione può essere fatta con un semplice programma "hard-coded"
- o appresa...
 - ▶ con un albero di decisione
- Oppure, non si calcolano feature "a mano"
 - ▶ e si usa una rete neurale con l'immagine del carattere

- Prima si “riscale” il carattere in una dimensione di uscita fissa (es. una griglia 8×8)



Per saperne di più ...

- Scrivetemi a simone.marinai@unifi.it
- O iscrivetevi ad Ingegneria Informatica