



Corso Luigi Einaudi, 55 - Torino

Appunti universitari

Tesi di laurea

Cartoleria e cancelleria

Stampa file e fotocopie

Print on demand

Rilegature

NUMERO: 1784A -

ANNO: 2015

A P P U N T I

STUDENTE: Iannizzi Giada

MATERIA: Classificazione e interpretazione dei dati biomedici -
Prof. Balestra

Il presente lavoro nasce dall'impegno dell'autore ed è distribuito in accordo con il Centro Appunti.

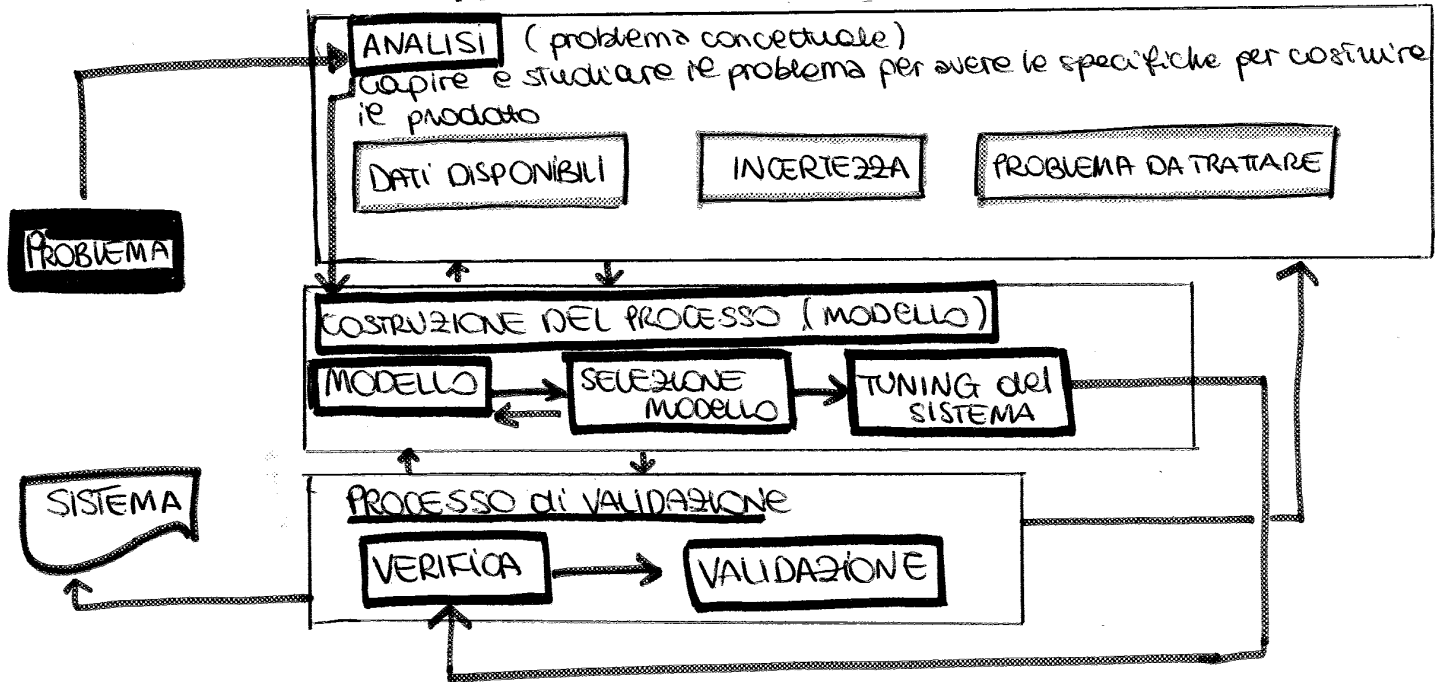
Tutti i diritti sono riservati. È vietata qualsiasi riproduzione, copia totale o parziale, dei contenuti inseriti nel presente volume, ivi inclusa la memorizzazione, rielaborazione, diffusione o distribuzione dei contenuti stessi mediante qualunque supporto magnetico o cartaceo, piattaforma tecnologica o rete telematica, senza previa autorizzazione scritta dell'autore.

**ATTENZIONE: QUESTI APPUNTI SONO FATTI DA STUDENTIE NON SONO STATI VISIONATI DAL DOCENTE.
IL NOME DEL PROFESSORE, SERVE SOLO PER IDENTIFICARE IL CORSO.**

CLASSIFICAZIONE E INTERPRETAZIONE DEI DATI BIOMEDICI

(1)

Come costruire un'applicazione?



1° PROBLEMA:

Come possiamo descrivere le caratteristiche di un set di elementi?

Soluzione: STATISTICA DESCRITTIVA

Esistono 2 statistiche

DESCRITTIVA
 descrizione del campione
 • CONOSCENZA DATO
 • TEST STATISTICI
 • ERRORE ASSOCIATO AL RISULTATO E GRAFICI

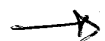
INFERENZIALE
 Si utilizza il campione per una generalizzazione riguardo alla popolazione dalla quale il campione è stato prelevato.

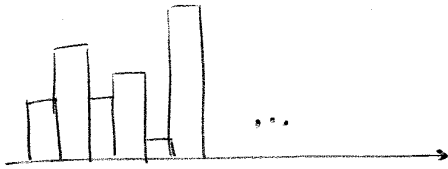
POPOLAZIONE → include tutti gli oggetti di interesse ← PARAMETRI (μ, σ)

CAMPIONE → solo una parte della popolazione ← STATISTICA (x, s)

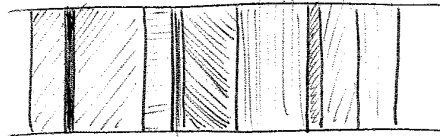
Esistono vari tipi di campionamento:

- Random
- Systematico
- Cluster
- Suddivisione
- CONVENIENCE: metodo peggiore, ma + utilizzato anche x questioni pratiche (soprattutto x patologie rare). Inserisco nel campione chi c'è.





Il percentile fa sì che le barre siano tutte alte uguali ma con intervalli \neq : (3)



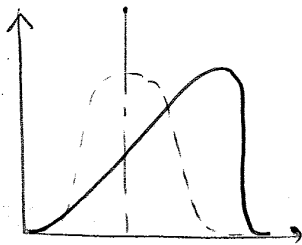
◦ **FREQUENZA DI DISTRIBUZIONE**

È un display ordinato di ogni valore in un data set insieme alle frequenze. Rappresenta le volte che si ripete il valore.

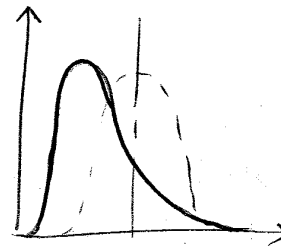
◦ **SKEWNESS**

Misura della simmetria della GAUSSIANA, o meglio della mancanza di simmetria.

$$\text{Skewness} = \frac{1}{(n-1)\sigma^3} \sum_{i=1}^n (x_i - \bar{x})^3$$



sk < 0

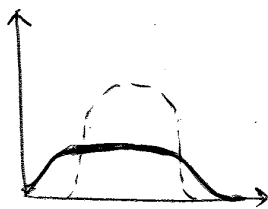


sk > 0

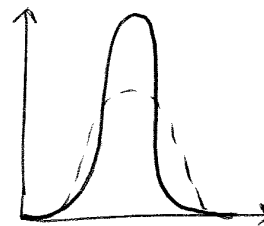
◦ **KURTOSIS**

Misura che indica se il picco della gaussiana è +o- alto delle normale.

$$ku = \frac{1}{(n-1)\sigma^4} \sum_{i=1}^n (x_i - \bar{x})^4$$



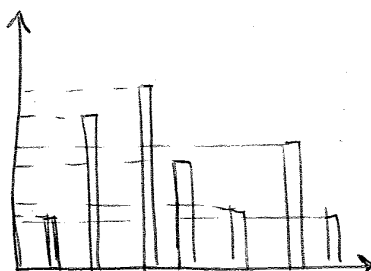
ku < 0



ku > 0

DIAGRAMMI:

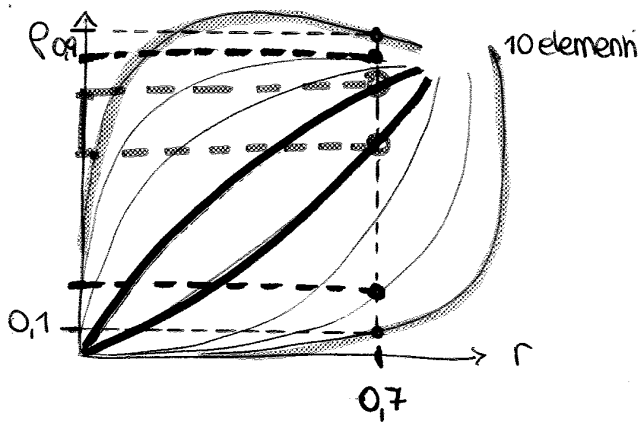
- **BAR DIAGRAM**: i dati devono essere divisi in classi:



Calcola le frequenze

! ρ = coeff. di correlazione della pop
 • r = " " " del campione

GRAFICO:



■ \approx 400 elementi

▨ \approx minimi elementi

$r=0,7 \rightarrow \rho$ è compreso tra
 su 10 elem. 0,1 e 0,9

⇓
 TROPPO INCERTEZZA

$r=0,7$
 su 20 elem. $\rightarrow \rho$ è compreso in
 un intervallo + stretto

$r=0,7$
 su 400 elem $\rightarrow 0,6 \div 0,8$

Intervallo ragionevolmente vicino al mio
 valore reale.

Per cui è una forte dipendenza con la numerosità del campione !!

↑ numerosità campione

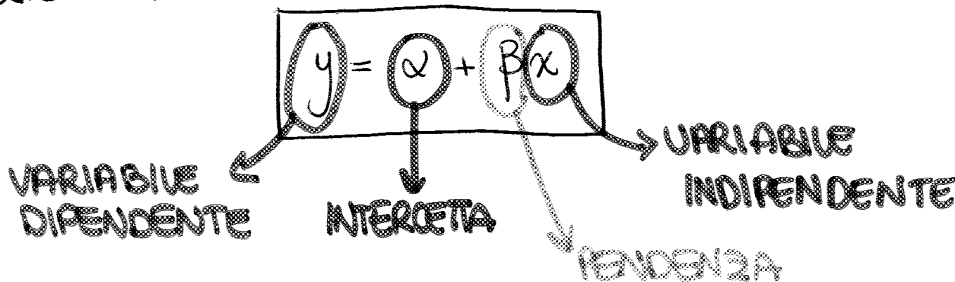
↑ attendibilità del coeff

3° PROBLEMA

Come possiamo ottenere un piccolo set di parametri che rappresentano il set di dati?

SOLUZIONE: REGRESSIONE LINEARE.

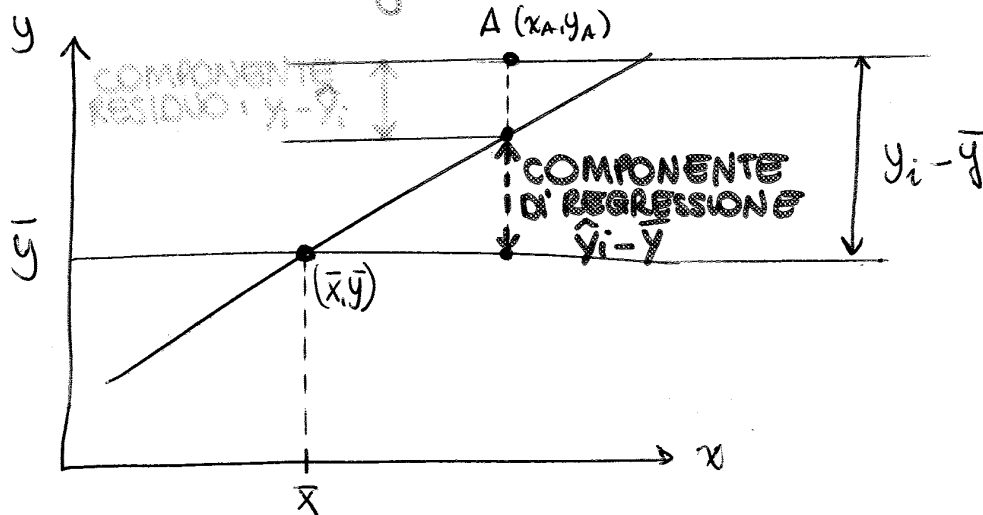
REGRESSIONE LINEARE SEMPLICE: approccio x predire una risposta quantitativa Y sulla base di una singola variabile X. Si assume che ci sia una relazione approssimativamente lineare tra X e Y che può essere rappresentata dalla RETTA DI REGRESSIONE:



α, β per la pop
 a, b = i campioni

$y = f(x)$
 retta, f = funz qualsiasi, ma la RL è un sottogruppo dei metodi di fitting.

! Se σ^2 è troppo alta, la regressione lineare non è un buon modello per i dati.



Il fitting è buono quando:

\sum componenti di regressione \gg \sum componenti residui

In ogni caso se \sum compon. residui è ALTA il fitting NON sarà buono. Il problema ora è capire l'affidabilità dei valori dei coefficienti

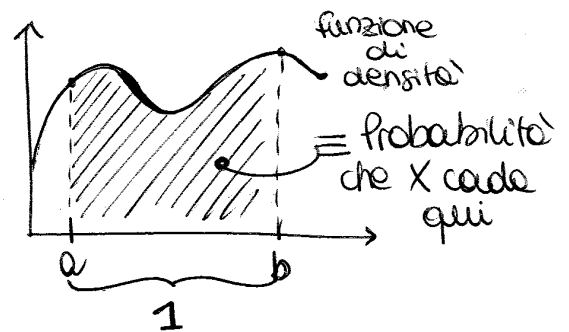
Ricorda: nelle applicazioni reali abbiamo accesso ad un set di dati di osservazioni dalle quali possiamo calcolare il minimo quadrati; comunque la retta di regressione lineare non è osservabile.

Memo:

DISTRIBUZIONE di PROBABILITÀ

Lista delle probabilità associate ad ognuno dei possibili valori della variabile.
 ≡ funzione di prob.

FUNZIONE DI DENSITÀ della PROBABILITÀ



TEST D'IPOTESI

Dobbiamo verificare l'attendibilità di un dato: un'hp statistica è un'assunzione che riguarda il parametro della popolazione. Può essere V o F. Il test d'hp è utilizzato a accettare o rifiutare l'hp statistica

HP STATISTICA

IPOTESI NULLA

H_0 ipotesi sul campione che deve essere disprovata (non c'è relazione tra X e Y), $H_0: \beta_1 = 0$

IPOTESI ALTERNATIVA

$H_1 (H_a)$ hp valida (almeno una è disprovata H_0) $H_1: \beta_1 \neq 0$ (c'è una certa relaz. tra X e Y)

ERRORE STANDARD

Ci dice quanto siamo distanti da un singolo estimatore, ad es la media. ⁽⁹⁾
 È definito come (della media)

$$\text{VAR}(\bar{x}) = \frac{\sigma^2}{n}$$

$n = \#$ elementi che costituiscono il campione
 $\uparrow n \quad \downarrow \text{VAR}(\bar{x})$

Ci dice la differenza tra \bar{x} e μ .

Può essere calcolato in funzione di a e b:

$$\text{VAR}(a) = \sigma^2 \left[\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \right], \quad \text{VAR}(b) = \frac{\sigma^2}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

Generalmente σ^2 non è noto, ma è ricavabile dai dati.

Questo estimatore è noto come RESIDUAL STANDARD ERROR:

$$\text{RSE} = \frac{\text{RSS}}{n-2}$$

RSS = valore dei residui

Posiamo inoltre individuare un intervallo di confidenza \Rightarrow individuando un range di valori (in %) entro una determinata probabilità % che il valore vero sarà in quel range.

ES: 95% \rightarrow se questo è l'intervallo, il mio vero valore sarà in quel range con il 95% di probabilità

ES: Regressione Lineare \rightarrow 95% confidence interval per β :

$$b \pm 2 \cdot \text{VAR}(b)$$

C'è il 95% di probabilità che β sarà nell'intervallo:

$$b - 2 \cdot \text{VAR}(b) \quad , \quad b + 2 \cdot \text{VAR}(b)$$

Idem per α .

REGRESSIONE LINEARE MULTIPLA

Normalmente noi abbiamo più di un predictor. Come fare? Estendere RLS a più variabili per comprenderle tutte. Possiamo fare ciò assegnando ad ogni variabile una pendenza \neq 0 modello.

Supponiamo di avere p variabili indipendenti. Devo relazionarle alla singola variabile dipendente che ho:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p$$

\Rightarrow sono gli effetti medi su y di una unità crescente in x_j mantenendo tutte le variabili indipendenti fisse.

Sono incogniti \rightarrow Metodo dei minimi quadrati \rightarrow

1. Prototipo = elemento a cui è associata una serie di caratteristiche che assumono un preciso valore caratteristico di quel determinato elemento. Può non essere reale e ad una classe possono essere associati più prototipi. (11)

Elemento di sono dei set di FEATURES = CARATTERISTICHE ⇒ DATI!

$$x = [x_1, x_2, \dots, x_p]$$

3. Classificare un elemento significa assegnare l'elemento alla classe a cui prototipo è più vicino all'elemento stesso.

ES:
$$\begin{array}{l} A - p_1 \quad d_1 \\ B - p_2 \quad d_2 \\ C - p_3 \quad d_3 \end{array} \rightarrow u \quad \min(d_1, d_2, d_3) = d_2 \text{ allora posso associare la classe corrispondente: } d_2 : u \in B$$

Normalmente ho solo 1 prototipo per classe. Quindi il problema è calcolare "d₂". Ecco che entra in gioco l'altra caratteristica.

2. Possibili misure di similarità per dati numerici:

- DISTANZA EUCLIDEA (NORMA L2):

$$d(x, y) = \|x - y\| = \sqrt{(x - y)^T (x - y)} = \sqrt{\sum_{i=1}^k (x_i - y_i)^2}$$

dove k = numero di dati.

- NORMA L1:

$$d(x, y) = \|x - y\| = |(x - y)| = \sum_{i=1}^k |x_i - y_i|$$

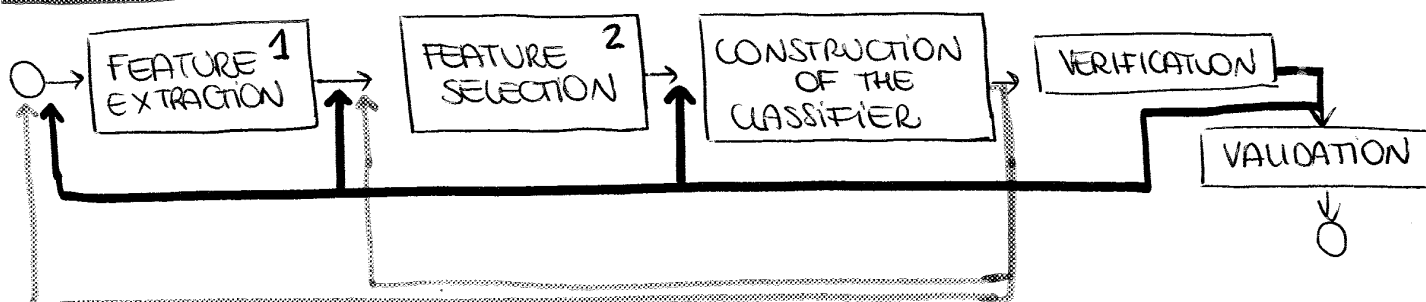
mette in evidenza maggiormente le differenze tra le caratteristiche rispetto ad L2.

- NORMA L_∞:

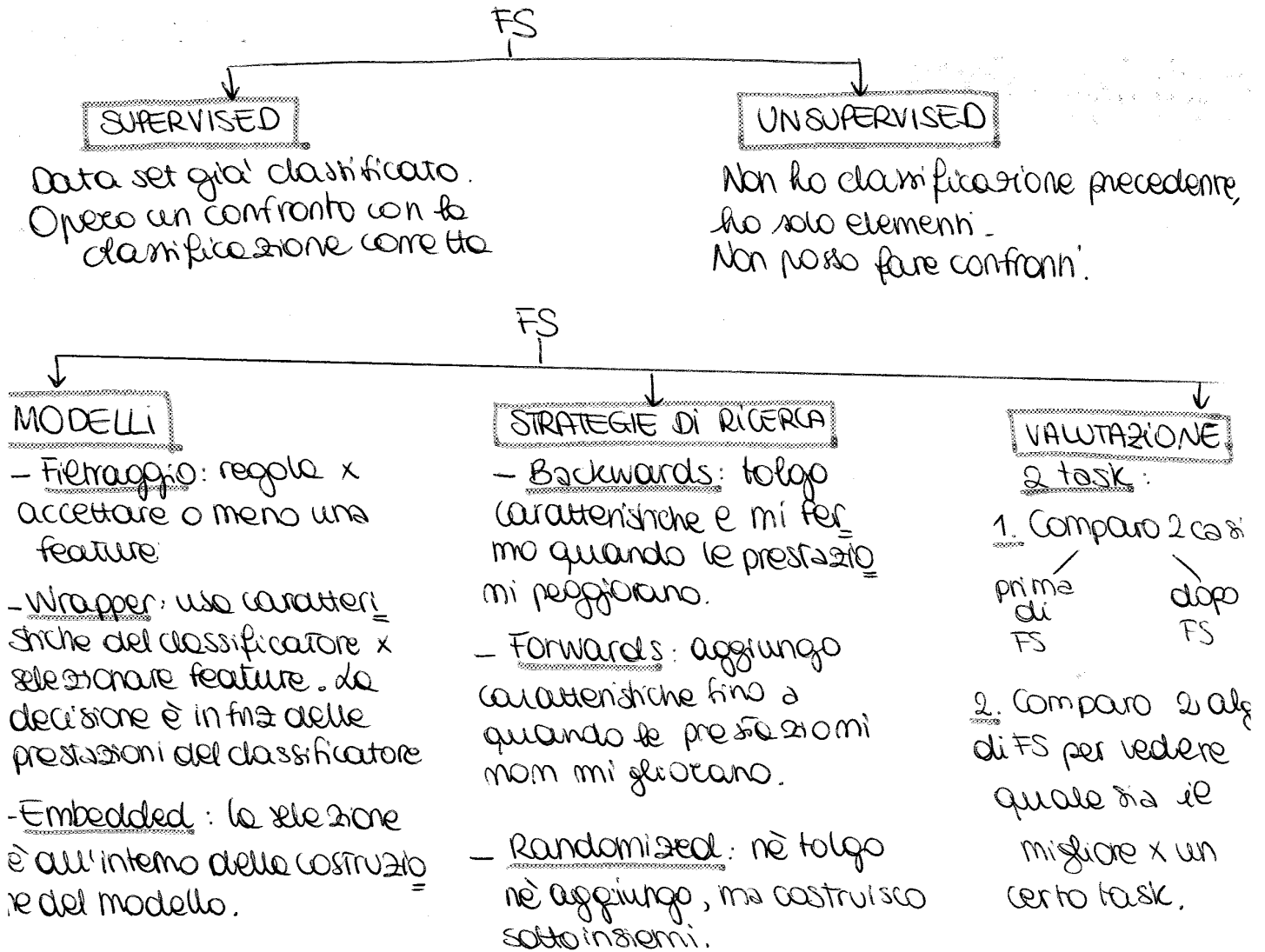
$$d(x, y) = \max_k |x_i - y_i|$$

Posso avere dati che sono INSIEMI o STRINGHE.

SVILUPPO DI UN CLASSIFICATORE



→ seleziono un sottoinsieme di caratteristiche in fnz delle prestazioni. (13)
 Non ho un # variabili o priori.



Feature Selection Bias

Utilizzando il medesimo training set data per classificare e feature selection posso risultare in questo bias. Quindi meglio separarli:

prestanzi peggiori

COMPUTATIONAL INTELLIGENCE (CI)

Comprende metodi che derivano dallo studio di meccanismi adattivi che permettono o facilitano il comportamento intelligente.

- È la parte dell'intelligenza artificiale (AI).

Quindi i metodi CI sono ispirati dalla natura e dalla biologia:

1. Evolutionary computing (EC)
2. Neural network
3. Swarm intelligence
4. Fuzzy system
5. Artificial immune systems

} in un sistema possiamo usare più di un metodo



EVOLUTIONARY COMPUTING

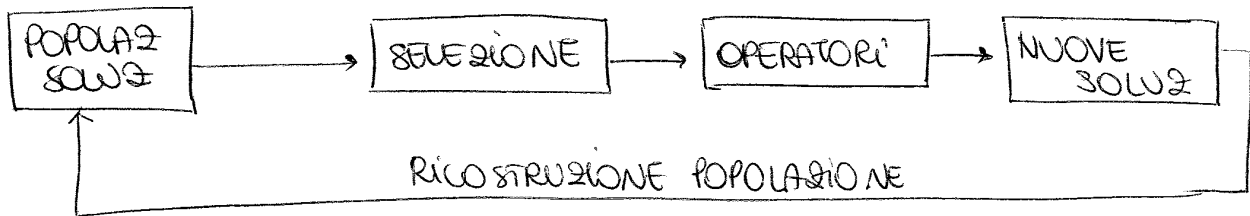
Si rifà alla ricerca locale. L'idea di base è semplice:

una popolazione di soluzioni candidate si evolve tramite successive iterazioni di variazione e selezione random.

Tutti i metodi si basano sulla probabilità = partendo dallo stesso un sistema di soluz posso trovare anche una soluz finale \neq .

ALGORITMO GENETICO

Il concetto su cui si basa è quello della RIPRODUZIONE; si parte infatti da una pop iniziale di soluz, della quale viene selezionato un sottoinsieme, al quale vengono applicati degli operatori ottenendo nuove soluz, che verranno integrate alle soluz iniziali ottenendo una nuova popolazione di soluz x l'iterazione successiva.



→ CROMOSOMI o GENOTIPO = caratteristiche di un individuo / variabile

→ GENI = ogni variabile da ottimizzare

→ ALLELE = assegnazione di un valore da un dominio alle variabile corrispondente.

▼ Trovare la giusta rappresentazione delle soluz candidate (es cromosom)

Rappresentazione binaria: $x \in \{0, 1\}^p \iff x = \{x_1, x_2, \dots, x_p\}$
 ↑
 Regola di codifica/decodifica

Popolazione Iniziale

Il primo passo è generare la popolazione iniziale.



Assegnare un valore random dal dominio ad ogni gene di ogni cromosoma. L'obiettivo della selezione random è di assicurarsi che la pop iniz sia una rappresentazione uniforme dell'intero spazio di ricerca. La dimensione della pop iniz è funzione della lunghezza della stringa (k).

Fitness

Funzione usata a quantificare quanto sia buona la soluzione rappresentata da un cromosoma. Tipicamente è una funzione matematica, ma non solo. →

Selezione

(17)

L'obiettivo principale è di enfatizzare le soluzioni migliori. A ciò si arriva tramite due passi:

- selezione di nuova popolazione: generata alla fine di ogni generazione con popolazione della generazione successiva.
- riproduzione: i figli sono creati attraverso l'applicazione di crossover e/o mutazione. Individui "superiori" dovrebbero avere > opportunità di riprodursi x assicurare che i figli contengano materiale genetico degli individui migliori.

Gli operatori selezione sono caratterizzati da **SELECTIVE PRESSURE** = tempo nel quale si arriva ad avere una popolazione uniforme \equiv n° preponderante di individui uguali.

Un operatore con un alto livello di selective pressure diminuisce la diversità nella popolazione + rapidamente che gli operatori con una bassa selective pressure che possono portare ad una convergenza prematura a soluzioni subottimali. $SP \uparrow$ limita le possibilità di esplorazione della popolazione.

Abbiamo altri operatori

- **RANDOM SELECTION**: ogni individuo (debole o forte) ha la stessa prob. di essere selez.
- **PROPORTIONAL**: "selez. elem. + adatto" \rightarrow roulette wheel; valore di fitness normalizzato \rightarrow stochastic universal
- **TOURNAMENT**: "seleziona in modo random un gruppo di n_{st} individui"
- **RANK-BASED**: " "
- **ELITISM**: " "
- **HALL OF FAME**: "viene creata una hall of fame dalla quale vengono selezionati genitori su cui applicare gli operatori, eppure vengono selezionati gli individui migliori in modo all'ultima generazione"

$\frac{1}{n_{st}} \rightarrow$ dim. rep.

n_{st} non troppo grande \rightarrow previene il dominio dei + forti
 n_{st} piccolo \rightarrow individui deboli possono essere scelti

⚠ Devo porre delle **STOPPING CONDITION**: numero max di iterazioni meno n_{im} o con i miglioramenti della fitness.

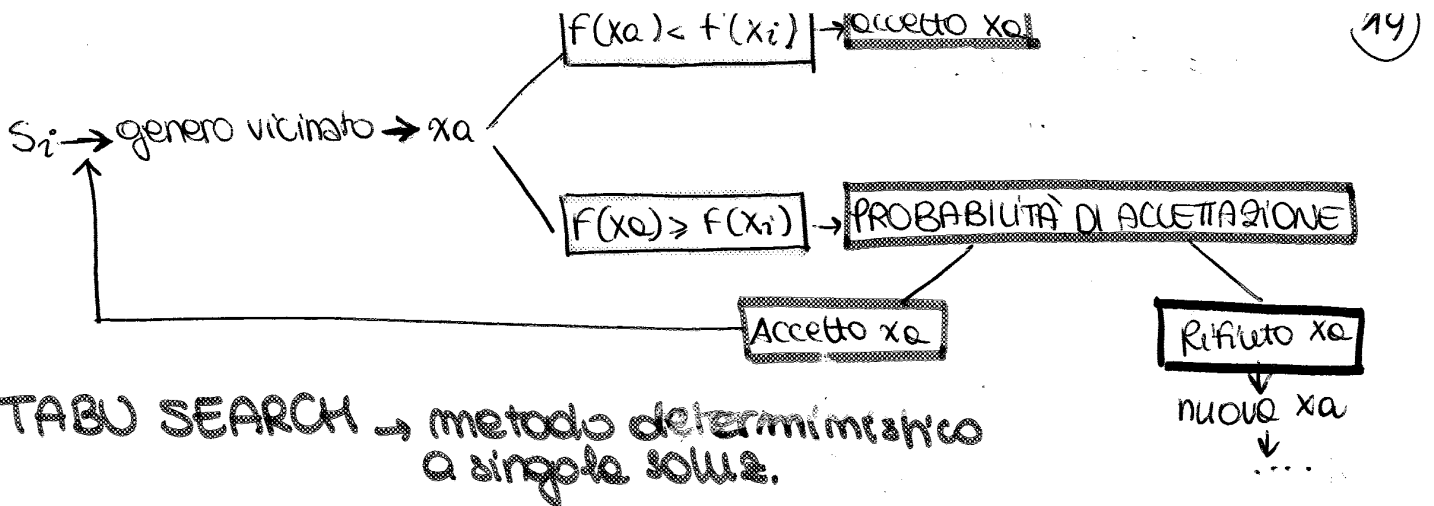
METODI EURISTICI

Abbiamo già parlato degli GA che fanno parte dei metodi euristici.

Per poterli utilizzare abbiamo bisogno di almeno 100 soluzioni, il tempo computazionale deve essere ragionevole, e l'algoritmo deve spaziare tutto lo spazio delle possibili soluzioni.

Caratteristiche:

- Sono ispirati alla natura
- Possono avere o no la memoria
- Deterministici vs stocastici:
 - Deterministici: utilizzando le stesse soluzioni iniziali otterremo le stesse soluzioni finali
 - Stocastici: vengono applicate regole random, stesso punto iniziale \rightarrow diverse finali



TABU SEARCH → metodo deterministico a singole soluz.

Idea principale: la MEMORIA forza la ricerca per esplorare nuove aree dello spazio.

Caratteristiche:

- soluz. iniziale che è utilizzata come soluz. corrente
- soluz. migliore nel vicinato è selezionata come soluz corrente anche se non migliore la soluz corrente.
- TS scarta il vicinato che ha già esplorato durante tutti i cicli.

↳ TABU LIST (memoria a breve termine)

de qui: TABU = PROIBIZIONE → MEMORIA

Posso superare il TABU → ASPIRATION CRITERIA (miglioramento della soluz)

SWARM INTELLIGENCE

Algoritmi ispirati al comportamento collettivo di insetti per recuperare cibo

- Caratteristiche:
- particelle semplici
 - " che cooperano anche in modo indiretto
 - si basano sul concetto di mobilità delle particelle nello spazio

ACO : ANT COLONY ORGANIZATION (Per funzionamento vedi di seguito)

Step:

- Random activity pattern per la ricerca del cibo
- Non appena il cibo è stato localizzato, activity patterns diventano organizzazioni con molte + formiche che seguono il medesimo percorso per il cibo
- Seguono tutte lo stesso percorso [+ BREVE] ≡ migliore.

Quindi l'algoritmo prevede di costruire la soluz. attuale in funzione delle soluzioni precedenti che hanno fornito il risultato migliore in termini della funz. obiettivo.

ACO inizia con l'inizializzazione del feromone ed è composto da:

- solution construction
 - pheromone update
-

FUZZY LOGIC

Reflects how people think. It attempts to model our sense of words, our decision making and our common sense.

Fuzzy logic is not logic that is fuzzy, but logic that is used to describe fuzziness.

Permette di superare l'assurdità (fuzziness) di distinzioni precise quanto limitanti in classi.

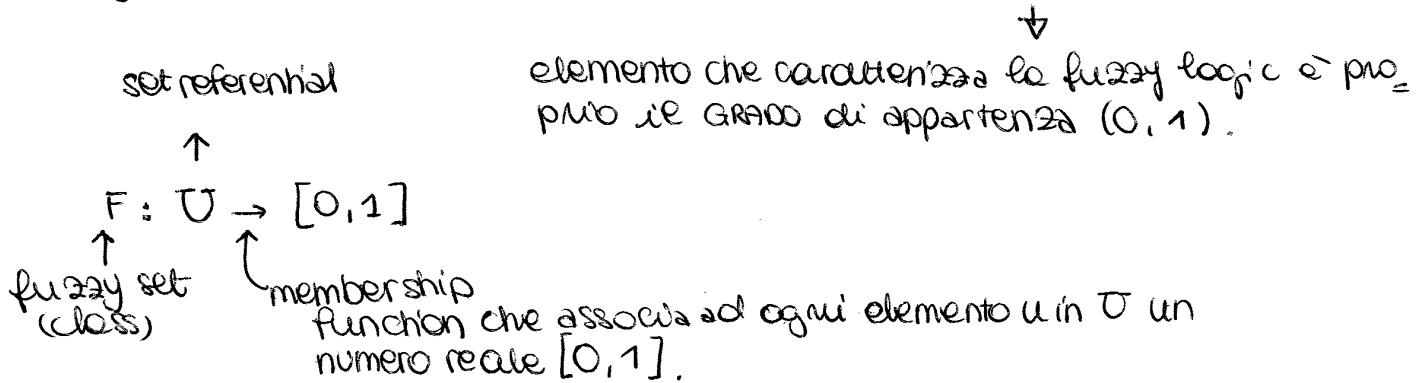
Costruisce una modellizzazione formale dell'incertezza la quale può influenzare i dati che si vogliono analizzare oppure può derivare da misure o dalla rappresentazione di un concetto indefinito, cioè che non ha confini o limitazioni chiare, utilizzando una metodologia rigorosa e matematica.

Fuzzy set

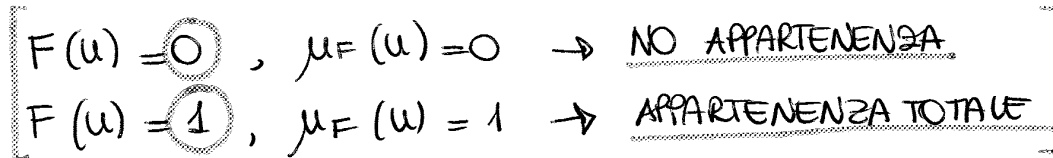
LOGICA CRISP: le variabili hanno solo un valore preciso (V o F, 1 o 0, bianco o nero).

LOGICA FUZZY

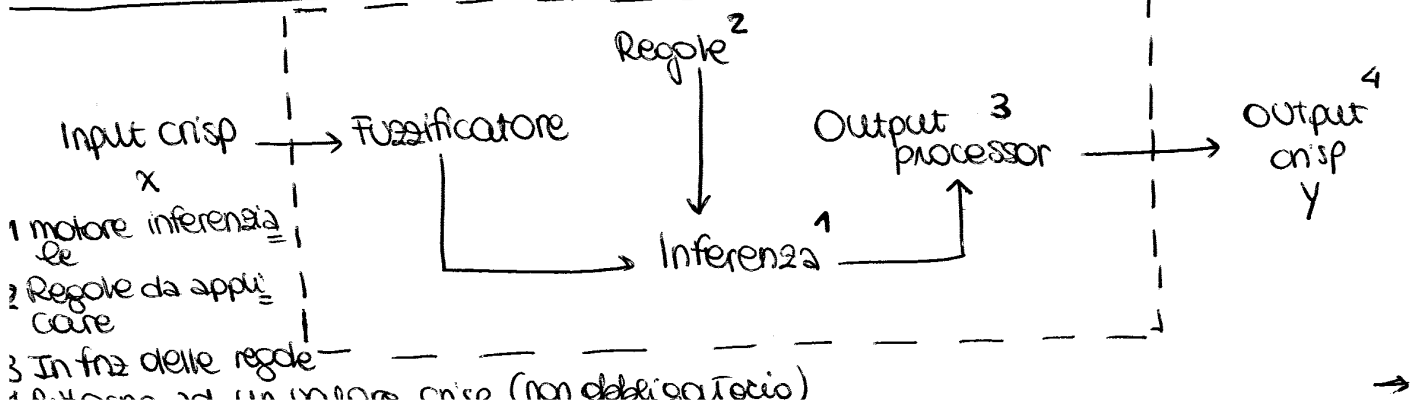
A fuzzy set is a class with a continuum of membership grades "



Il valore della MF, $\mu_F(u)$, rappresenta il grado di appartenenza di u in F .



CLASSIFICATORE FUZZY:



RANDOM SET VIEW : ottengo un range di valori x di cui scindiamo le varie età. (23)

ES:	GIO	AD	AN	valore min	Analizziamo giovani:	valore max
P1:	18-27	33-51	≥60	18	-----	35
P2:	≤30	30-60	≥60	18	-----	27
P3:	18-35	40-60	≥60	18	-----	30
P4:	18-30	35-55	≥65	18	-----	35
				18	-----	31

Valore minimo di partenza:

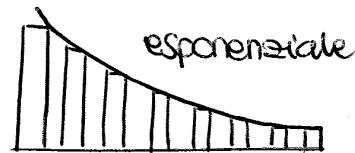
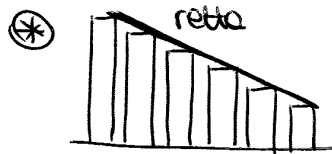
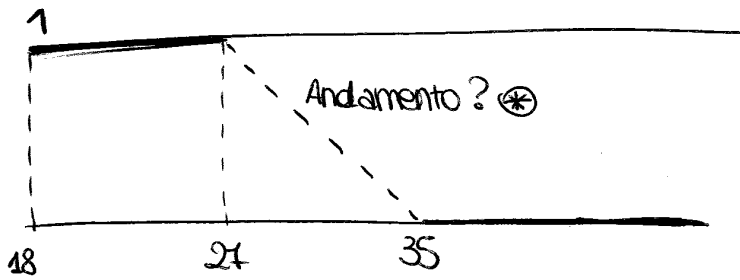
18 anni

• Valore minimo massimo :

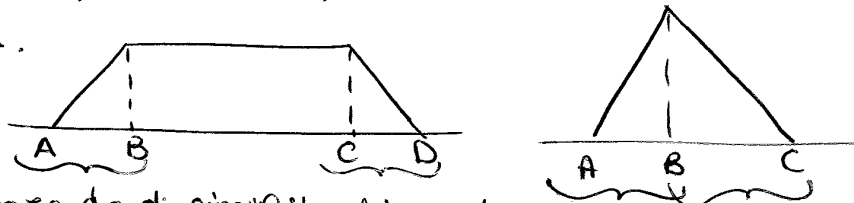
27 anni

• Valore massimo :

35 anni



✓ SIMILARITY VIEW : prende punti di riferimento. Per definire i punti importanti di μ .



La MF misura il grado di similitudine di un elemento rispetto al set in questione. È un esempio perfetto del set che appartiene a quel set con un grado max. Gli altri invece sono misurati attraverso la distanza relativa dall'esempio perfetto.

FUZZINESS & PROBABILITY

$$\text{Probabilità} = \frac{\text{numero di eventi favorevoli verificati}}{\text{numero di possibilità}}$$

logica fuzzy → descrivo il grado di appartenenza di un elemento ad una classe

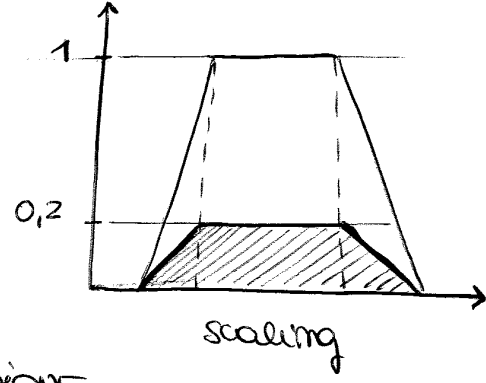
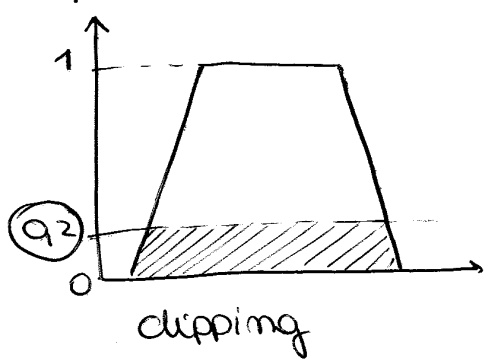
entrambe si riferiscono ad un grado di incertezza (o certezza) degli eventi. Il grado di certezza dato dalle probabilità statistiche si è significato prima che l'evento occorra. Mentre un set fuzzy è importante dopo che l'evento è già avvenuto.



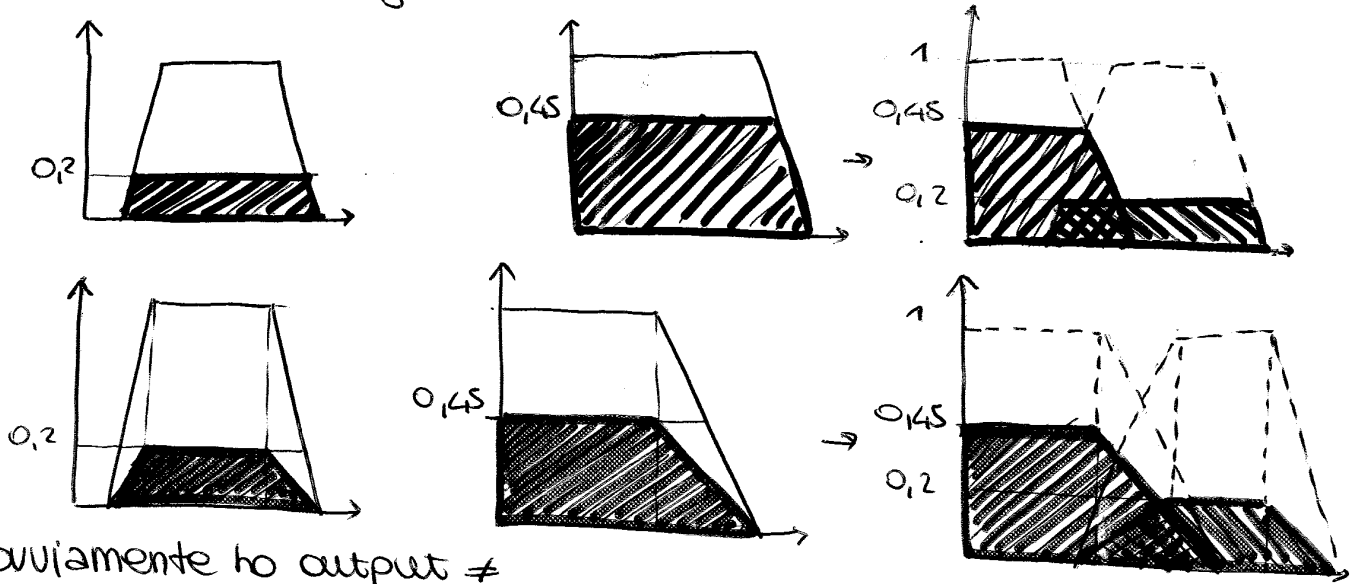
INFERENZA

(25)

Il metodo più comune di correlazione delle regole con sequenze con le valenze dell'antecedente è di tagliare MF conseguenza al livello del valore vero dell'antecedente (CLIPPING).
 SCALING permette di mantenere la forma originale della MF.



Normalmente ho + regole → AGGREGAZIONE:



ovviamente ho output ≠

DEFUZZIFICAZIONE

Spesso come risultato vogliamo ottenere un valore crisp. Il metodo più utilizzato è CENTROID TECHNIQUE = punto in cui una linea verticale divide l'aggregato in 2 masse uguali (COG):

$$COG = \frac{\int_a^b \mu_A(x) \cdot x \, dx}{\int_a^b \mu_A(x) \, dx} \rightarrow COG = \frac{\sum_{x=a}^b \mu_A(x) \cdot x}{\sum_{x=a}^b \mu_A(x)}$$

TECNICHE DI INFERENZA

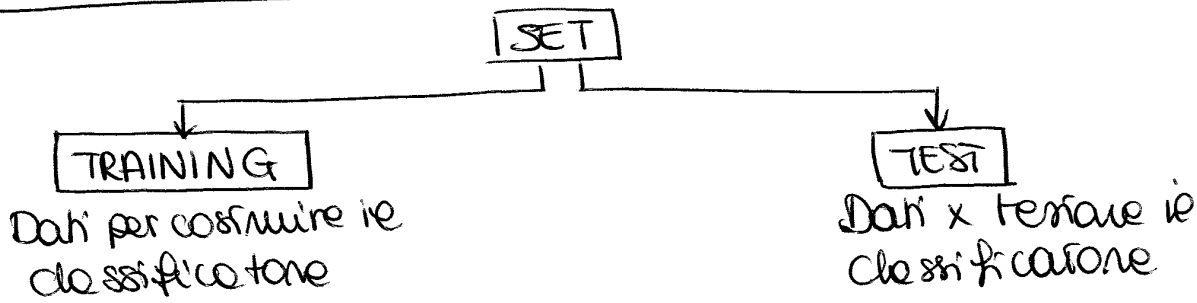
✓ MANDANI : 4 step

1. fuzzification degli input
2. Valutazione delle regole
3. Aggregazione delle regole e degli output
4. Defuzzificazione

✓ SUGENO : ≈ Mandani, ma invece di un fuzzy set uso una fnz matematica degli input.
 conseguente di una regola!!

TRAINING SET E TEST SET

(27)



Dalla popolazione iniziale estraggo elementi (in modo casuale o no) e ottengo training e test set.

Posso avere ≠ situazioni:

- Training set = test set: situazione migliore possibile. Le prestazioni non possono essere migliorate
- Suddivido test set e training set: 2 parti separate senza elementi in comune. Non devono essere numericamente piccole e devono rappresentare nel modo + corretto la classe con numero ≥ di elementi / classe.

Ordine di grandezza — Training set: 30% - 50%

- LEAVE-ONE-OUT: quando ho pochi elementi. Dal set tot tolgo 1 elemento e lo classifichiamo

CLUSTERING

Dato un set di elementi noi vogliamo suddividerlo in gruppi = cluster dal momento che non conosciamo le classifiche di partenza.

Δ con fuzzy:

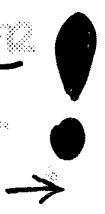
- FUZZY: classi = regole, costruisco prototipo partendo dagli elementi
- CLUSTERING: non conosco né QUANTE classi né i PROTOTIPO.

Algoritmo:

1. Hp: certo n° di cluster
2. Suddivide con una regola gli elementi del data set e li classifichiamo nei vari cluster
3. Costruisce un prototipo
4. Spostamento dei vari elementi nei cluster (NON modifica i prototipi).

ES: Supponiamo un ≠ n° di cluster con lo stesso metodo:

5 } Quale è il n° di cluster migliore?
 7 } PRINCIPIO: MINIMA VARIABILITÀ NEL CLUSTER
 10 } MASSIMA DISTANZA TRA CLUSTER.



Come trovo il nuovo prototipo: valor medio di ogni variabile
 (Σ tutti gli elementi per posizione e divido per 6)

(29)

$$\left\{ \frac{23}{6}, \frac{75}{6}, \frac{86.2}{6} \right\}$$

$$\left\{ 3.83, 12.5, 14.4 \right\} \text{ Nuovo prototipo}$$

Riformo le classi e ad es E1 è finito in un'altra classe:

Questo algoritmo minimizza l'errore quadratico medio:

ISODATA

K-means ha un problema legato alla necessità di definire a priori un numero di cluster, questo normalmente non porta alla definizione del # ottimale di cluster.

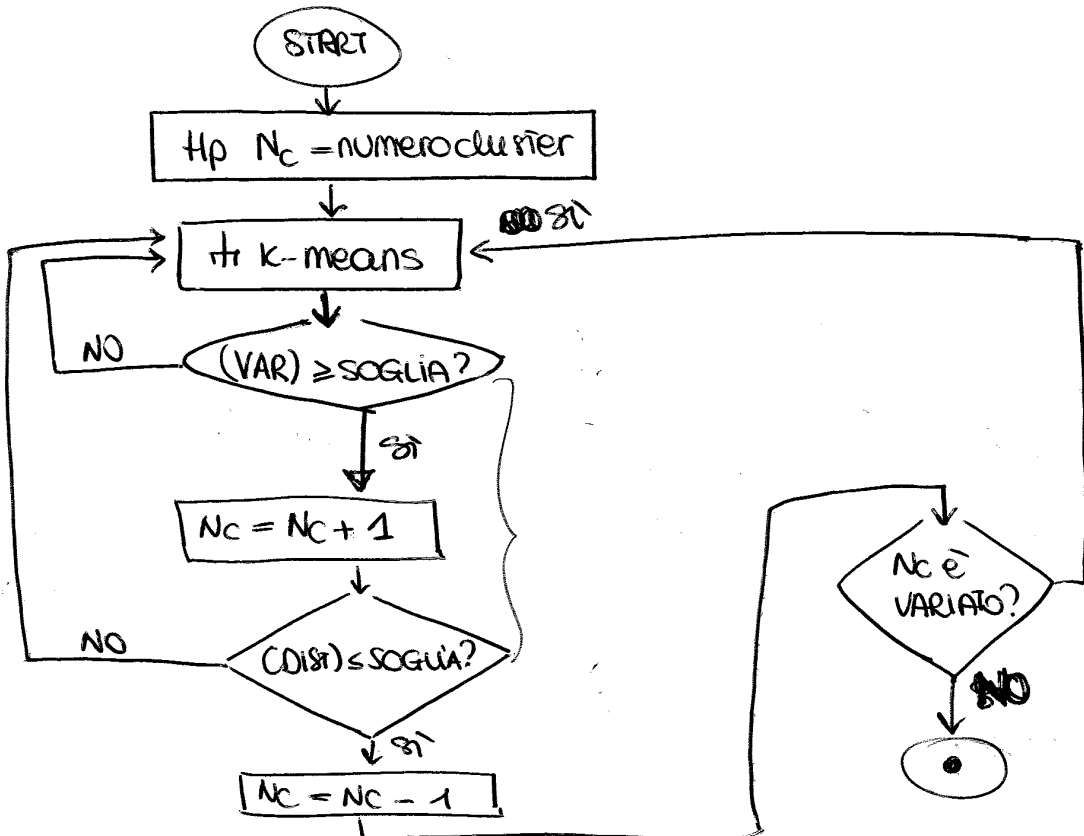
Per avviare al problema:

- riclassificare i campioni partendo da un numero diverso di cluster
- post-elaborare il risultato andando a unire cluster che risultino simili
- usare l'algoritmo **ISODATA**

Estensione del k-means con euristiche per selezionare automaticamente il numero di cluster.

Parametri:

NMINLEX, ND, σ_s^2 , DMERGE, NMERGE



RETI NEURALI

(31)

I DATI sono un p.to di partenza: per prendere una decisione bisogna basarsi sull' INFORMAZIONE.

per ottenerla è necessario fornire CONOSCENZA.

La maggior parte dei sistemi presenta una FASE di APPRENDIMENTO = il sistema apprende conoscenza, e una FASE di UTILIZZO = il sistema applica la conoscenza fornendo l'informazione.

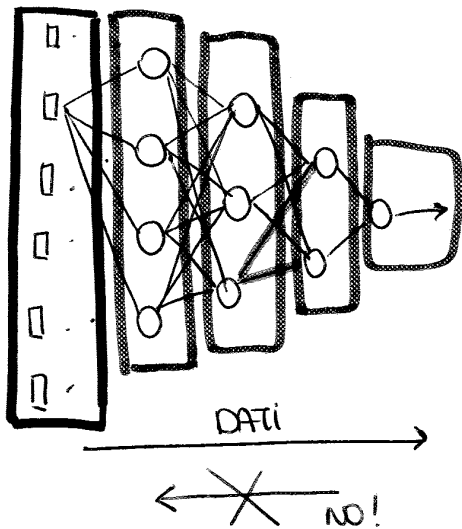
ALGORITMO di APPRENDIMENTO

- Il TRAINING SET deve essere:
- sufficientemente numeroso
 - sufficientemente omogeneo
 - sufficientemente completo

- 1. Fase di training = si sottopone il sistema ad esempi, TRAINING SET
 2. Fase di validazione: si verifica che il riconoscimento effettuato sia corretto o meno.

Altra caratteristica importante delle reti neurali è la STRUTTURA / ARCHITETTURA:

Grado:



LAYER DI INPUT

LAYER NASCOSTI

LAYER DI OUTPUT

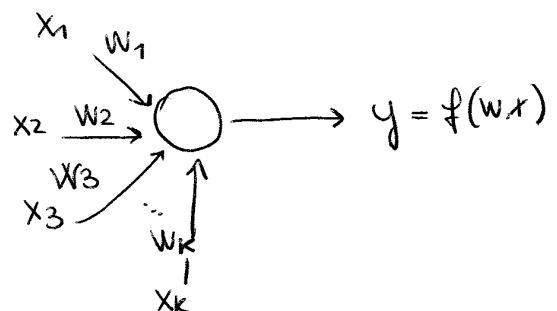
ARCHI A CUI È ASSOCIATO UN PESO (w_i)

Struttura a feed-forward

MECCANISMO di ELABORAZIONE

Struttura definita PERCEPTION:

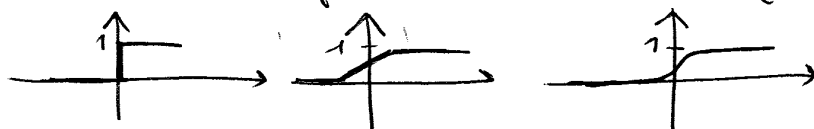
l'uscita $y = f\left(\sum_{i=1}^K w_i \cdot x_i\right)$



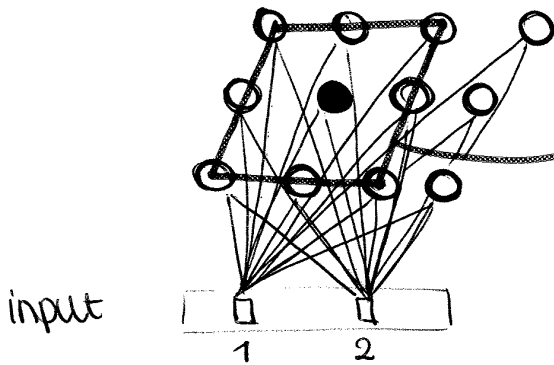
I pesi che hanno un valore compreso tra $[0; 1]$ fanno sì che un certo dato di input influenzi in maniera + o - (signi differenti) l'uscita del singolo neurone.

Altra caratteristica delle reti neurali è il NEURONE:

Ogni neurone può presentare una \neq funz. di attivazione (LOGSIG, TAGSIG, PURELIN)



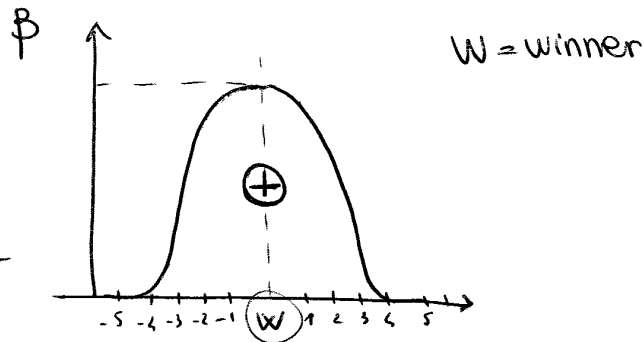
Alternative al competitive learning sono le **MAPPE DI KOHONEN**. Funziona come il C.L. ma le modifiche dei pesi riguardano anche i neuroni vicini al vincitore per rendere anche + simili quelli vicini e + diversi quelli distanti.
 Unico layer nascosto bidimensionale (n° neuroni dipende dal n° di cluster che viene stimato, normalmente $>$ del # di cluster).



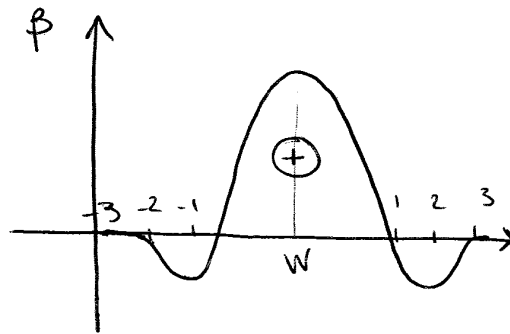
VINCITORE
INTORNO: posso avere varie forme e la dimensione dipende dalle dimensioni della mappa.

La modifica dei pesi del neurone vincitore e dell'intorno è gestita tramite una FUNZIONE e un COEFFICIENTE β . Il vincitore è quello che subisce le modifiche $>$ e gli altri $<$ modificano in fn z della loro distanza dal vincitore. Funzioni utilizzate:

◦ **GAUSSIANA**:
 massimo in corrispondenza del vincitore. Decresce in modo simmetrico a dx e sx fino ad assumere un valore nullo x i neuroni + distanti.



◦ **A CARPELLO MESSICANO**:
 Massimo in corrispondenza del vincitore, decresce in modo simmetrico fino a valore negativo x accentuare le distanze dal vincitore.



Allo fine dell'addestramento \rightarrow Balle che contengono neuroni fricatamente vicini.



RETI NEURALI

(31)

Tendono spunto dai meccanismi di ragionamento del nostro cervello. Fanno parte degli "intelligenza". È lo strumento per eccellenza legato all'apprendimento.

Fasi:

- TRAINING : training set
- VALIDAZIONE : test set
- UTILIZZO : elementi che \notin training set

È un processore massivo parallelo che ha la capacità di immagazzinare conoscenze e renderle disponibili all'uso.

2 parti:

- conoscenza è appresa da network attraverso un processo di apprendimento
- connessione interneurale ha la sua forza nei pesi sinaptici che sono usati per immagazzinare conoscenza.

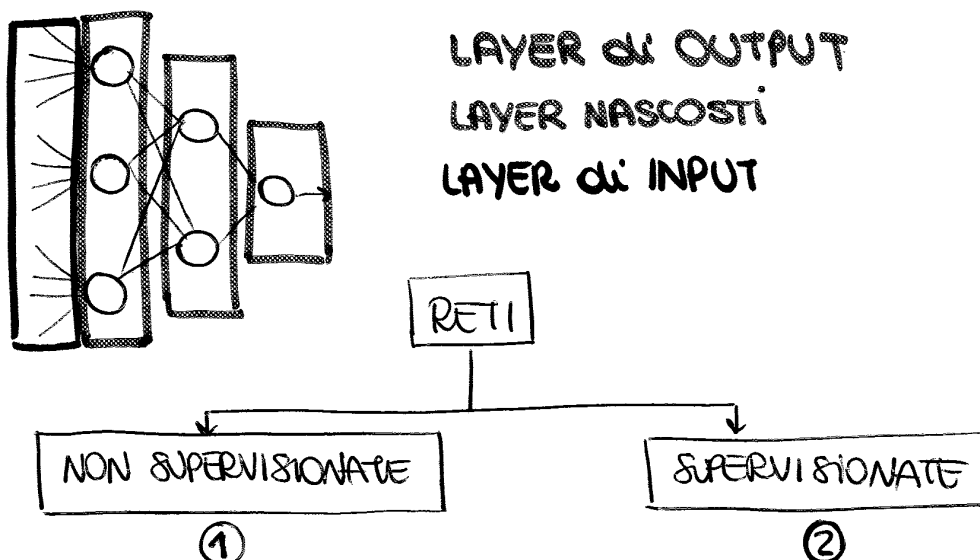
Processo di apprendimento è chiamato ALGORITMO di APPRENDIMENTO → funzione è modificare i pesi sinaptici della rete.

Rete neurale processa le info non tramite un algoritmo sequenziale. Decomposizione parallela di info complesse in elementi base.

Elementi su cui si basa:

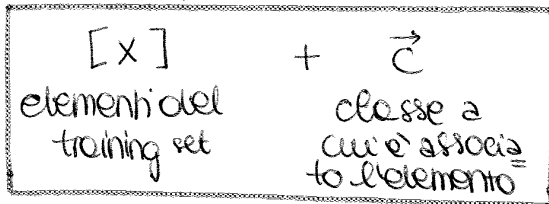
- ARCHITETTURA: definisce la struttura della rete = numero di neuroni artificiali nella rete e la loro interconnettività
- NEURONI: elementi che processano e sono caratterizzati dalla fnz di attivazione
- ALGORITMO DI APPRENDIMENTO: parte + impo.

La rete neurale è un GRATO formato dai nodi (neuroni artificiali dove avvengono i conti).



Quando si è parlato di algoritmi di apprendimento, abbiamo detto (33) essere la parte + importante. Questi possono essere:

o SUPERVISIONATI: devono avere un TRAINING SET composto da:

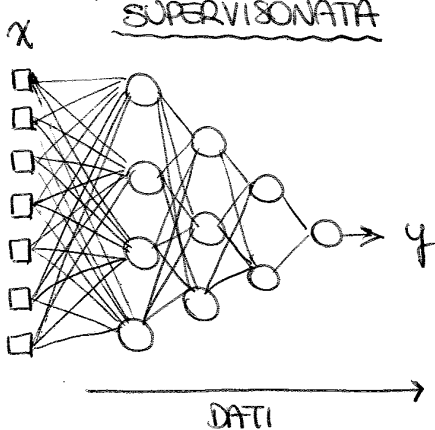


Partendo da un insieme di pesi iniziale, vengono classificati gli elementi in $[x]$, si confrontano i risultati della rete con \vec{c} e poi i pesi vengono modificati. Se è stato raggiunto il numero max di iterazioni, mi fermo, altrimenti riclassifico, ...

Algoritmo + utilizzato è BACK PROPAGATION (vedi dispense)

o NON SUPERVISIONATI: il TRAINING SET è composto dalla sola matrice $[x]$ e si basa sul CLUSTERING. Anche qui abbiamo modifica dei pesi, la classificazione degli elementi in $[x]$, ma il confronto avviene all'interno dell'insieme dei pesi, con gli input. Molte tipologie di algoritmo.

Esempi di rete SUPERVISIONATA e NON SUPERVISIONATA:

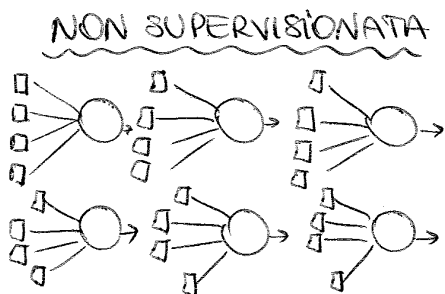


Problema: come organizzare la struttura. Ogni neurone è connesso a quelli del layer successivo, ma all'interno non ho nessun collegamento.

L'input è legato alle caratteristiche

INPUT: x
OUTPUT: y

Output unico generale

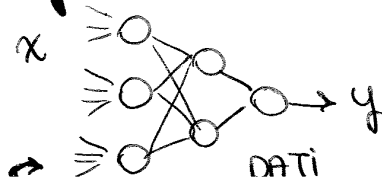


Ogni neurone non è connesso con quello successivo, me' con altri, ma solo con gli input.

Un solo layer

Un output locale \forall neurone.

1 Back propagation

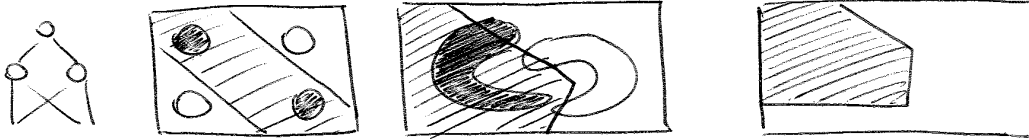


L'errore si propaga all'indietro

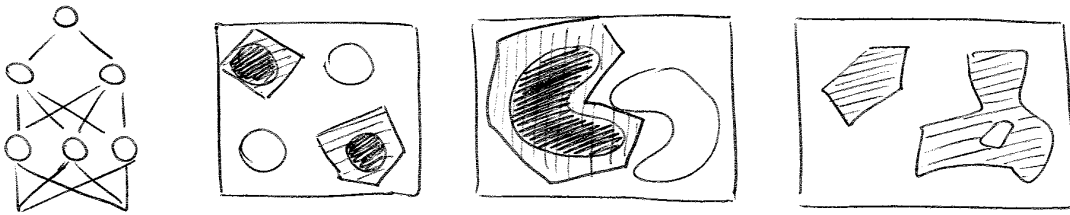
$$x \Rightarrow \text{O} \rightarrow y \quad y = f(w, x) = f(\sum wx)$$

ERRORE = $f(c, f)$

- 1, 2: normalmente regioni = 2 subree. separazione in due parti (35) ti, ma non ho + una retta come elemento separatore.

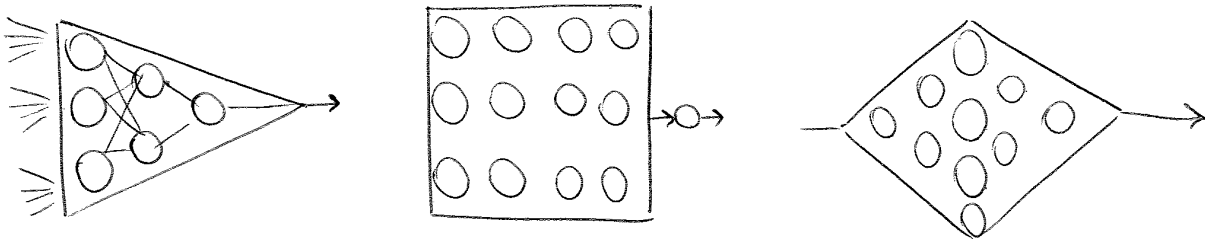


- # layer = # caratteristiche che contengono l'input:
 ↑ layer, le classi fanno aree di qualunque conformazione



Definito il num di layer, devo definire il numero di neuroni e la funzione f . f dovrebbe essere definita neurone x neurone, per cui potenzialmente cambia \forall layer.

Forme reti:



Decidere la forma a priori è complesso. Devo perciò fare varie prove.

Competitive learning

Alg. di apprendimento che vuole vincere sugli altri. I neuroni competono tra loro per diventare attivi.

Il neurone di output che vince è chiamato "winner-takes-all".

Durante la fase di apprendimento solo i pesi del neuroni vincitori vengono aggiornati.

○ $W_1 \{w_{11}, w_{12}, \dots, w_{1p}\}$

○ $W_2 \{w_{21}, w_{22}, \dots, w_{2p}\}$

○ $W_3 \{w_{31}, w_{32}, \dots, w_{3p}\}$

$x_1 = \{x_{11}, \dots, x_{1p}\}$

calcolo:

$$d_1 = \sum_{j=1}^p |w_{1j} - x_{1j}|$$

$$d_2 = \sum_{j=1}^p |w_{2j} - x_{1j}|$$

$$d_3 = \sum_{j=1}^p |w_{3j} - x_{1j}|$$



FACCIAMO IL PUNTO DELLA SINTESI...

Parliamo di **STATISTICA** quando vogliamo:

1. Descrivere le caratteristiche di un SET di elementi → STATISTICA DESCRITTIVA
2. Correlare il set di dati campione con la popolazione da cui il campione è stato prelevato → STATISTICA INFERENZIALE

① Parliamo di: media, varianza, mediana, moda, coeff di variazione, Range, % di, SKEWNESS e KURTOSIS.

Le rappresentazioni sono

- Bar diagram
- Box plot
- Scatter plot

② Parliamo di COVARIANZA e coefficiente di correlazione (ρ, r).

3. Ottenere un piccolo set di parametri che rappresenti però l'intero set di dati → REGRESSIONE LINEARE

③ Parliamo di REGRESSIONE LINEARE SEMPLICE:

$$y = \alpha + \beta x$$

e di test di ipotesi

Parliamo anche di REGRESSIONE LINEARE MULTIPLA quando abbiamo a che fare con molte variabili

• Parliamo di **OTTIMIZZAZIONE** quando vogliamo selezionare dal nostro set di dati solo le soluzioni, variabili che sono utili per un certo scopo, dato dalla nostra funzione obiettivo (f). Quindi le caratteristiche dell'OTTIMIZZAZIONE sono:

- S : insieme delle soluzioni
- f : funzione obiettivo (spesso sconosciuta)

Quello che otteniamo è una serie di soluzioni dalle quali deve essere selezionata quella migliore →

OTTIMO GLOBALE / OTTIMO LOCALE in modo iterativo →

Come ricerca la soluzione ottima?

RICERCA LOCALE :

1. Inizializzazione : scelta di una soluz (soluz. corrente) e calcolo della funzione obiettivo
- 2. Generazione del vicinato : ricerca nell'intorno della soluz corrente una nuova soluz
3. Test di accettazione : verifica che la soluz trovata nel vicinato possa essere accettata come migliore
4. Test di terminazione : se l'esito è positivo la procedura è terminata.

Quali metodi utilizzo per l'ottimizzazione? CI:

1. ALGORITMI GENETICI
2. SIMULATED ANNEALING
3. TABU SEARCH
4. METODO DEL GRADIENTE
5. ACO } SWARM INTELLIGENCE

Metodi euristici

- ① GA :
- stima delle soluzioni : $2^k - 1$ ($k = \text{dim stringa}$)
 - • Generazione della popolazione iniziale (random), N
 - FITNESS • SELEZIONE (vari metodi): per trovare un sottoinsieme M che contiene alcune soluz assunte come GENITORI
 - A queste M soluzioni applico gli operatori genetici (CROSSOVER, MUTAZIONE) per ottenere nuove M soluzioni, FIGLI.
 - Integro alle N soluz iniziali per mantenere il numero di partenza

- ② SIMULATED ANNEALING : ricerca locale nella quale posso selezionare anche soluz peggiori tramite la PROBABILITA' di ACCETTAZIONE.

3. TABU SEARCH : variante del simulated annealing, ma con MEMORIA → non posso esplorare un vicinato già visitato.
4. METODO del ∇ : base della ricerca locale con gradienti fissi.

5. ACO : si basa sulle formiche e sulla ricerca del cibo. Elemento caratteristico → PHERORMONE : il rilascio di questo da parte di una formica che ha trovato la via breve permette alle altre formiche di seguire questo pattern.

Parliamo di **CLASSIFICAZIONE** quando dato un elemento e una classe (set di classi) dobbiamo associare quell'elemento alla classe. Come fare?

- PROTOTIPI della classe : FEATURE EXTRACTION
- REGOLA DI CLASSIFICAZIONE
- MISURA DI SIMILITÀ

Una volta selezionati i prototipi → FEATURE SELECTION : eliminazione di feature ridondanti, irrilevanti e mantenute le rilevanti.

Metodi di classificazione:

1. FUZZY LOGIC
2. CLUSTERING
3. RETI NEURALI

① FUZZY LOGIC : misura dell'incertezza di un DATO. Si basa sul concetto di GRADO DI APPARTENENZA → MEMBERSHIP FUNCTION

② CLUSTER : obiettivo è la suddivisione del set di dati iniziale in classi (= cluster). Concetti base:

- MINIMA VARIANZA INTRA-CLUSTER
- MASSIMA DISTANZA TRA CLUSTER

Metodi per clusterizzazione:

- K-MEANS
- ISODATA
- DENDROGRAMS

③ RETI NEURALI: Fanno parte dei metodi euristici.
~~Algoritmo~~ ^{Metodo} legato all'apprendimento: si aveva la rete con funzioni di TRAINING per ottenere una classificazione il + possibile ottimo rispetto ad una classificazione di potenza (supervisionate) oppure rispetto a nulla (non supervisionate). Elementi base:

- ARCHITETTURA
- NEURONI
- ALGORITMO DI APPRENDIMENTO