



**Corso Luigi Einaudi, 55 - Torino**

**Appunti universitari**

**Tesi di laurea**

**Cartoleria e cancelleria**

**Stampa file e fotocopie**

**Print on demand**

**Rilegature**

NUMERO: 1553A -

ANNO: 2015

# **A P P U N T I**

STUDENTE: Bettale

MATERIA: Classificazione e Interpretazione + domand d'esame.  
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.**

# APPLICATION DEVELOPMENT SVILUPPO BENE APPICAZIONI

COME POSSIAMO COSTRUIRE UN'APPICAZIONE?

DAL PROBLEMA → SI ARRIVA AL SISTEMA  
3 PASSI  
PROBLEMA

## 1° ANALISI (CONCETTUALIZZAZIONE DEL PROBLEMA)

CAPIRE E STUDIARE BENE IL PROBLEMA X AVERE  
LE SPECIFICHE VUOLTE E X SAPERE OVVI SONO:

- I DATI A DISPOSIZIONE
- IL RUOLO E LA GESTIONE DELL'INTELLIGENZA
- LE PROBLEMATICHE E CHE TIPO DI APPROCCIO  
BISOGNA UTILIZZARE

## 2° PROCESSO DI COSTRUZIONE DI UN MODELLO

GRAZIE A MATEMATICA, LOGICA, INFORMATICA...  
E SCELTA DEL METODO CON CUI UTILIZZARE QUESTO  
MODELLO

MODELLO E METODO E IL TUTTO DEVE SUCCESSIVAMENTE  
ESSERE ADEGUATO AL SISTEMA SPECIFICO

(TUNING)

UNA FINE SI OTTIENE IL PROTOTIPO DEL SISTEMA

⚠ nel caso ci si accorge che manca qualcosa  
SI PUÒ TORNARE INDIETRO ALLA FASE DI ANALISI

## 3° PROCESSO DI VERIFICA E VALIDAZIONE

XC IL PRODOTTO DEVE — SODDISFARE LE SPECIFICHE (VER.)  
— ESSERE FUNZIONALE X CHI (VAL.)  
LO UTILIZZA

SISTEMA PRONTO

⚠ ogni volta che si è risolto un problema,  
ce ne si crea un altro (cerchio ∞)

NOI PARTIAMO DAL PUNTO 2, DAL CONSTRUCTION PROCESS



## ② Statistica inferenziale

inferenza = deduzione generalizzativa

Si fanno delle inferenze su come un valore statistico rappresenta tutta la popolazione

serie x valutare la rappresentatività del campione, altrimenti non si può estendere subito un risultato di un trial a tutta la popolazione  
 → obiezione tipica del trial: criteri di selezione x arruolarsi!

sono tecniche che permettono di usare i campioni x fare delle **generalizzazioni** sulla popolazione o su quale sono presi i campioni

⚠ lettere x indicare - popolazione  $\mu, \sigma$  - campioni  $x, s, a$

[ la popolazione include tutti gli oggetti di interesse, mentre il campione è solo una porzione della popolazione.

Parametri → associati alle popolazioni, lettere greche

Statistiche → associate ai campioni, lettere romane

quando si usano i metodi inferenziali, è importante che il campione rappresenti accuratamente la popolazione

## 5 modi x estrarre un campione (5 tipi di estrazione)

① **E. casuale**: casualità totale ogni elemento nella popolazione ha la stessa probabilità di essere estratto  
 (RANDOM)  
 è il modo preferito di estrazione, ma è modo difficile da fare

② **E. sistematica**: campionamento sistematico ogni  $N$  elementi  
 è + facile del random  
 ex: segnavi

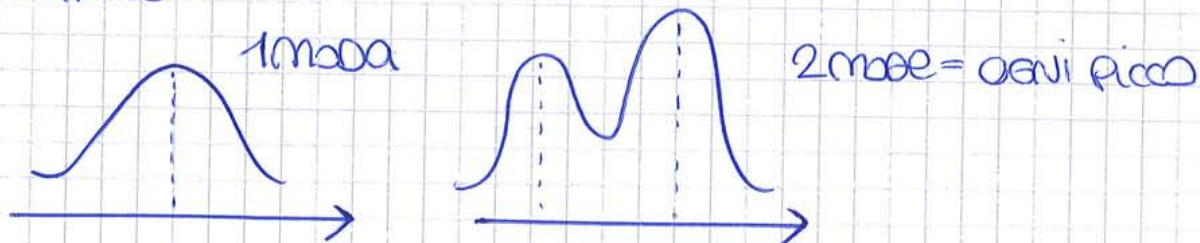


## ● MODA

valore che si ripete più frequentemente tra tutte le osservazioni in un campione

può non essere unica

valore che + frequentemente assume la variabile



## ● VARIANZA

Dice quanto è dispersa la popolazione:

$\uparrow$  : campione disomogeneo

$\downarrow$  : campione omogeneo  $\rightarrow \bar{x}$  significativo

$$s^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

Dipende dall'unità di misura della variabile!

## ● Deviazione Standard

$$s = \sqrt{s^2}$$

→ risolvere il problema della dipendenza e confrontare come variano 2 valori e la loro riproducibilità

## ● Coefficiente di variazione

$$100\% \cdot \left( \frac{s}{\bar{x}} \right) \quad \text{valore adimensionale!}$$

questo valore rimane sempre lo stesso indipendentemente da quali unità di misura si usano

## ● RANGE

Differenza tra l'osservazione massima e la minima

coppia formata dal valore minimo e dal valore massimo che trova nel campione da l'idea di quanto è ampio il campo di valori della mia variabile

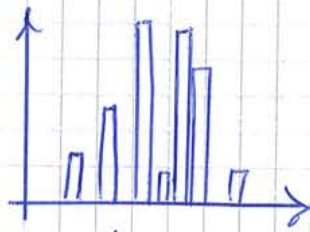


Ci sono inoltre 3 possibili rappresentazioni

Grafiche:

① Diagramma a barre

Gli elementi sono suddivisi in gruppi e poi ne si calcolano le frequenze

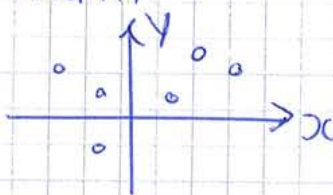


in Excel

- Data analysis
- Istogramma
- INPUT (colonna di valori)
- BIANCO (intervallo 20:40-160)
- OUTPUT (foglio dove)
- LEGENDA

② Diagramma casuale (SCATTER)

Ci sono 2 variabili, 1 su ascissa e 1 su ordinata e vengono inseriti i vari elementi come punti



③ Diagramma di BOX

Il campione viene diviso in 4 percentili

È un box che contiene il 50% dei dati

Dice se la popolazione è equamente distribuita grazie alla posizione della mediana



**2° PROBLEMA** che relazione c'è tra i parametri della popolazione e i corrispondenti parametri del campione?

ovvero, dato un set di elementi presi da una pop, come usarlo x studiare QST POP?

- con la statistica descrittiva sappiamo come è fatto il campione
- poi si passa alla statistica inferenziale x fare delle **stime** e sapere come sono legati tutti i parametri visti ↓

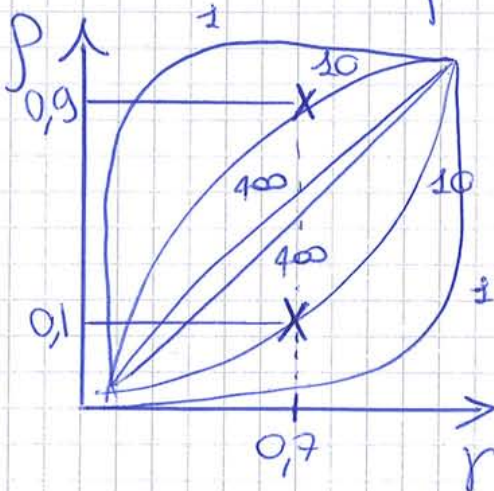
il valore vero della popolazione si raggiunge se aumentano molto i campioni

(questo è un problema usuale quando si cerca di analizzare i dati acquisiti durante un TRIALS)



Solitamente, x una stima veritiera bisogna avere  
 $p > 0,7$  v  $p < -0,7$   
 se  $-0,7 < p < 0,7$ , dubbi!! i punti sono molto asperci!  
 Grafici pp. 13

x analizzare  $p$  si usa il **DIAGRAMMA AD OCCHIO**



ci sono 2 curve in base al numero di elementi analizzati

$r$  è la stima della correlazione che si calcola sul campione su TOT elementi

Trovato  $r$  (ex  $r = 0,7$ ) lo interseco con le curve nel numero di elementi, poi parto i 2 valori a sx e vedo in che range sta  $p$

$r = 0,7 \rightarrow p \in [0,1; 0,9] \times 10$  elementi!  
 non va bene!

$r = 0,7 \rightarrow p \in [0,6-0,8] \times 100$  elementi

intervallo ragionevole e significatività maggiore!

La numerosità del campione è quindi importante:

con maggiori elementi, la stima migliora!

"deduzioni" o "generalizzazioni"

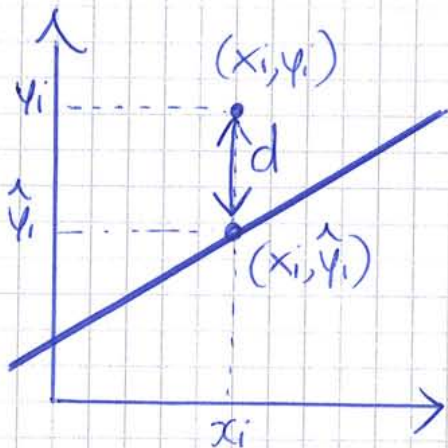
Le inferenze vanno sempre documentate e

ragionate, soprattutto quando si hanno pochi

campioni! (cio non vale x la statistica descrittiva)



L'obiettivo è stimare  $\alpha$  e  $\beta$  in modo che il modello lineare della retta si adatti ai dati disponibili: cioè trovare un'intercetta  $\alpha$  ed una pendenza  $\beta$  in modo che la linea risultante sia la più vicina possibile ai dati.



Per ogni punto ho 2 coppie di valori

$(x_i, y_i)$  quella reale

$(x_i, \hat{y}_i)$  quella che sta sulla retta della regressione

$$\hat{y}_i = f(x_i)$$

misuriamo la distanza

$$d = y_i - \hat{y}_i$$

$d$  rappresenta l'1-esimo **Residuo**, cioè la differenza tra il valore reale osservato e il valore previsto dal modello lineare supposto.

Si vuole una retta che minimizza la somma <sup>dei quadrati</sup> di tutte

queste distanze  $S = \sum_{i=1}^n d_i^2$

ottenendo le 2 formule

$$b = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} ; a = \bar{y} - b\bar{x}$$

$(\bar{x}, \bar{y})$  è un punto che sta sempre sulla retta

Bisogna però valutare successivamente anche la **Bontà** della regressione effettuata, tramite 2 valori:

- il componente dei residui :  $Res = y_i - \hat{y}_i$

- il componente della regressione :  $Reg = \hat{y}_i - \bar{y}$

una buona regressione si ha sempre se il valore assoluto del componente della regressione è maggiore di quello dei residui (sommatoria)

$$\|\hat{y}_i - \bar{y}\| \text{ (ampio)} > \sum_{i=1}^n \|y_i - \hat{y}_i\| \text{ (piccolo)}$$

$$\|Reg\| > \sum \|Res\|$$



$H_1$ : IPOTESI ALTERNATIVA  
 viene affermata se si disprova  $H_0$   
 Dice che le osservazioni del campione sono influenzate da qualche causa non casuale  
 (cioè effettivamente i 2 valori sono diversi)

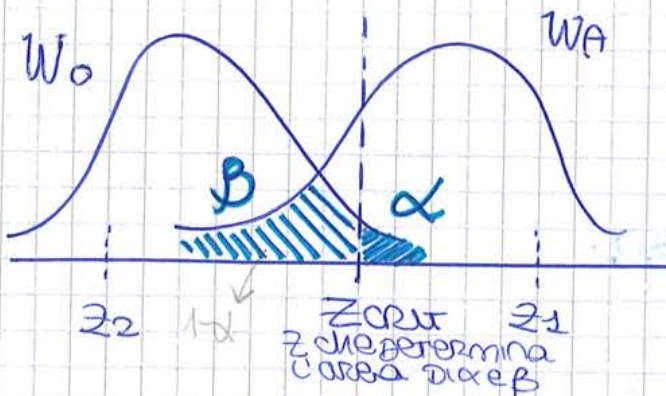
CASACUNA IPOTESI HA ASSOCIATA UNA FUNZIONE DI DENSITÀ  $W_0$  e  $W_1$ , solitamente normale (📈)

SI HANNO POI I DECISION ERRORS, cioè 2 TIPI DI ERRORI che possono risultare da un test di ipotesi <sup>imperfezione delle probabilità</sup>

● **errore di tipo I**: - è un errore di significatività  
 - avviene quando si disprova  $H_0$  se in realtà essa è vera  
 - la probabilità di commettere quest errore è chiamata livello di significatività oppure  $\alpha$

● **errore di tipo II**: - avviene quando si considera  $H_0$  vera quando invece è falsa (x cui non si riesce a disprovarla)  
 - la probabilità di non commettere questo errore è chiamata potenza del test  
 - la probabilità di commettere quest errore è  $\beta$

$\alpha$  e  $\beta$  sono quindi  $\neq$  da prima, se in questo caso sono i 2 errori e sono collegati tra loro dallo spostamento di  $Z_{crit}$ , valore x il quale si decide per un'ipotesi o l'altra.



Le funzioni di densità  $W_0$  e  $W_1$  rappresentano le ipotesi  $H_0$  e  $H_1$

calcolo  $Z$ :

se  $Z > Z_{crit} (Z_1)$   
 siamo in  $W_1 \rightarrow H_0$

se  $Z \leq Z_{crit} (Z_2)$   
 siamo in  $W_0 \rightarrow H_1$

NB: nel test di ipotesi solitamente si fissa o decide  $\alpha$ , o su si impone al max un certo valore  $\alpha$  non si riesce a calcolare  $\beta$

$\alpha < 0.05$



## Assessing the Accuracy of the Coefficient Estimates

How far off will that single estimate of  $\mu$  be? In general, the standard error gives us an idea of how far of a single estimate, for example of the mean, is.

The standard error of the mean estimate is

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

Roughly speaking, the standard error tells us the average amount that this estimate  $\bar{x}$  differs from the actual value of  $\mu$ .

In a similar way we can estimate the standard errors associated with  $a$  and  $b$ :

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

In general,  $\sigma^2$  is not known, but can be estimated from the data. This estimate is known as the *residual standard error*, and is given by the formula

$$RSE = RSS/(n - 2).$$



Multivariate normal regression is the regression of a d-dimensional response on a design matrix of predictor variables, with normally distributed errors. The errors can be heteroscedastic and correlated.

```
>> [beta, sigma, E]=mvregress(x,y);
```

```
beta =
```

```
1.0e+03 *
```

```
-0.0008
```

```
0.3911
```

```
-1.4477
```

```
2.4037
```

```
sigma = 1.0707e+03
```

```
sum(abs(E)) = 5.3456e+03
```

In statistics, a collection of random variables is *heteroscedastic* if there are sub-populations that have different variabilities from others.

## Relationship between $b$ and $r$

The relationship between the sample regression coefficient ( $b$ ) and the sample correlation coefficient ( $r$ ) is

$$b = r \frac{S_y}{S_x}$$

↙  
COEFFICIENTE  
DI  
CORRELAZIONE  
( $r$ )  
STANDARD ( $r$ )



## Multiple linear regression

Simple linear regression is a useful approach for predicting a response on the basis of a single predictor variable.

However, in practice we often have more than one predictor.

How can we extend our analysis in order to accommodate more predictors?

One option is to run separate simple linear regressions, each of which uses a different variable.

The approach of fitting a separate simple linear regression model for each predictor is not entirely satisfactory.

Instead of fitting a separate simple linear regression model for each predictor, a better approach is to extend the simple linear regression model so that it can directly accommodate multiple predictors. We can do this by giving each predictor a separate slope coefficient in a single model.

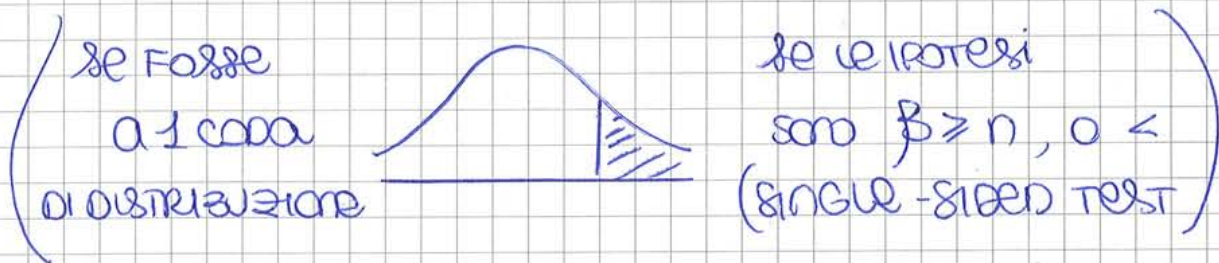
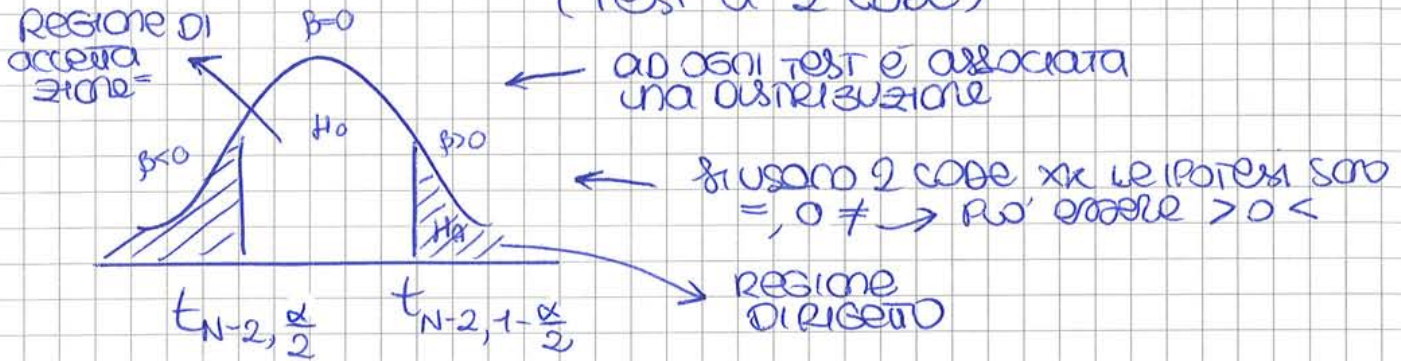




## ESEMPIO DI TEST:

TEST x REGRESSIONE LINEARE SEMPLICE:

$$\begin{cases} H_0 \text{ se } \beta = 0 \\ H_A \text{ se } \beta \neq 0 \end{cases} \quad \begin{array}{l} \text{PER TESTARE QUESTE IPOTESI} \\ \text{\& USARE UN TWO-SIDED TEST} \\ \text{(TEST A 2 CODE)} \end{array}$$



PASSI DA SEGUIRE:

① si calcola il valore della statistica, cioè il valore della variabile nella distribuzione a partire dai dati

$$t = b'_\beta \left( \frac{S^2_{\epsilon x}}{L_{xx}} \right)^{1/2}$$

② si decide la regola di decisione, si confronta il valore calcolato con altri valori di riferimento di tabella.

se  $\alpha$  è il livello di significatività (che si può imporre) si usa  $\alpha/2$  perché abbiamo 2 code

- rigetto  $H_0$  se,  $t > t_{N-2, 1-\frac{\alpha}{2}}$  ,  $t < t_{N-2, \frac{\alpha}{2}}$
- accetto  $H_0$  non riesco a dimostrarla se,  $t_{N-2, \frac{\alpha}{2}} < t < t_{N-2, 1-\frac{\alpha}{2}}$

↳ non si è riusciti nel proprio intento la regressione non ha valore

③ il valore di p è dato da  $p = 2 \cdot \Delta$  se  $t < 0$   
 $p = 2 \cdot \Delta$  se  $t > 0$   
 regola: se  $p < \alpha$ , si può rifiutare  $H_0$



# 4° PROBLEMA

Dato un grande numero di variabili, si è sicuri che sono tutte utili?

Si vuole estrapolare il sottoinsieme di **variabili utili** x affrontare il problema:

## SCELTA DELLE FEATURE (CARATTERISTICHE)

Ci sono molte ≠ applicazioni (REGRESSIONE MULTIPLA, CLASSIFICAZIONE ...) e pure molti ≠ metodi x risolvere il problema.

uno dei metodi è la OTTIMIZZAZIONE

Bisogna definire la **classificazione**:

- Dato un certo numero di classi e dato 1 elemento, si vuole assegnare questo elemento ad 1 classe!

Nella maggior parte dei metodi di classificazione, l'elemento è assegnato univocamente ad una classe, ma ad esempio, nella logica fuzzy, un elemento può essere assegnato a più classi → si definisce il GRADO DI APPARTENENZA

Ogni metodo di classificazione è caratterizzato da:

- 1 insieme di prototipi associati alle variabili
- 1 misura di similarità
- 1 regola di classificazione

## PROTOTIPI = insieme di caratteristiche utilizzate x rappresentare una classe

PROTOTIPO = - è un elemento rappresentativo di  $T$  (pattern) gli elementi della classe

- non è detto sia reale
- ad una classe possono essere associati più prototipi

Ogni elemento è caratterizzato da un insieme di caratteristiche, cioè  $p$  variabili che assumono un determinato valore  $x$  questo elemento

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

anche il prototipo ha un valore associato ad ogni variabile → queste variabili sono le caratteristiche



# MISURE di SIMILARITÀ

x un essere umano è facile vedere la somiglianza tra 2 elementi, ma non è così x una macchina.

Lo strumento che si usa x comparare automaticamente 20+ elementi è la misura di similarità.

∃ diverse misure di sim., le più note dal punto di vista matematico sono:

● Distanza euclidea, norma  $L_2$

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

con  $k$  = numero di caratteristiche  
è un'indicazione media

● norma  $L_1$

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

Differenza in valore assoluto

individua la complessiva distanza di tutte le variabili

● norma  $L_\infty$

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

massimo delle differenze

individua la classe meno opportuna

(mette in evidenza situazioni particolari)

Lo stesso se non ho numeri da confrontare, ma stringhe ed insiemi

EX: Insiemi L e M

$L \cap M \Rightarrow$  massima intersezione = massima similarità

$\frac{L \cap M}{L \cup M} \Rightarrow$  spesso si normalizza rispetto all'unione:  
serie se a sono distribuite di elementi negli insiemi che confronto

EX: x confrontare le stringhe si lavora su pezzi di stringhe in comune



# 3 esempi di classificatori

## a) Valutazione della complessità aziendale (C.A.)

IL RESPONSABILE CAPO INFERMIERISTICO DELLA CLINICA CELLINI HA SVILUPPATO IL METODO MAP X GLI INFERMIERI, CHE DEVONO AVERE

- INFORMAZIONI DELLA CARTELLA CLINICA
- INFORMAZIONI SULLA +0 - AUTOSUFFICIENZA
- " " SUI CARE - GIVERS (PERSONE CHE SVOLGONO PARTE DEL LAVORO DEGLI INFERMIERI, COME IL MANGIARE)

TUTTE QUESTE VARIABILI SERVONO X VALUTARE LA COMPLESSITÀ AZIENDALE, CHE PUÒ ESSERE:

ALTA - medio/Bassa - BASSA - (mai nulla)!

MAP { metodo aziendale professionalizzante } serve per:

- MONITORARE COME SI MODIFICA NEL TEMPO LA C.A. DEL PAZIENTE (SOPRATTUTTO X QUEL IN UN'EMERGENZA, 2/4 MESI)
  - FAR VEDERE COME SI DISTRIBUISCE LA C.A. DI UN REPARTO, IN UN ARCO DI TEMPO, X CAPIRE SE IL PERSONALE È SUFFICIENTE
- una soluzione è avere persone in + che ruotano su reparti diversi, dove ce ne bisogno = molto complicato DAL PUNTO DI VISTA ORGANIZZATIVO

IL TEAM HA RUPRESO IL MAP X MIGLIORARNE LE PRESTAZIONI, E SI CONTINVERÀ ANCORA NEL PROGETTO.

PUNTI PRINCIPALI:

- 1) CONOSCERE I PROCESSI INFERMIERISTICI (ANALISI E MATEMATIZZAZIONE)
- 2) STUDIARE IL METODO (ANALISI DEL MAP)
- 3) PROPOSTA DI EVOLUZIONE DEL METODO, BASATA SU LOGICA FUZZY
- 4) TEST DELLA NUOVA IMPLEMENTAZIONE SU ≠ PAZIENTI DI ≠ REPARTI
- 5) ANALISI DEL TEST:
  - IL CLASSIFICATORE NON DA' NON CLASSIFICANTI (PRIMO OBIETTIVO IMPORTANTE)
  - RIPORTA 89% CORRETTE
  - 11% NON CORRETTE

→ DA MIGLIORARE ANCORA E DIMINUIRE IL + POSSIBILE

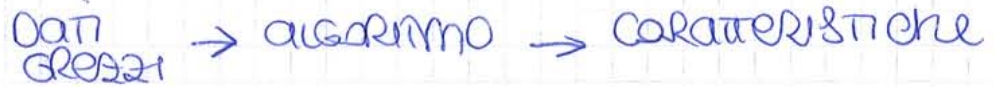
NON È IL CLASSIFICATORE PERFETTO CHE NON SBAGLIA MAI.



# FEATURE EXTRACTION PRIMA FASE DELLO SVILUPPO DEL CLASSIFICATORE

2 metodi  $\leftarrow$  si prendono i dati così come sono

si applicano ai dati degli algoritmi che costruiscono delle caratteristiche a partire dai dati grezzi



analizziamo l'estrazione delle caratteristiche in 3 gli esempi:

## (a) map

ci sono 4 dimensioni / variabili / caratteristiche

- collaborabilità del paziente
- stabilità clinica
- autogestibilità
- ambiente

Ogni dimensione è caratterizzata da un certo numero di parametri = modalità  $\rightarrow$  ogni peso con il proprio peso

Ogni modalità è descritta da una serie di variabili - possibili range di valori che la modalità può assumere  $\rightarrow$  3 diversi stati sui corrispondono valori numerici  
ogni modalità ha un rinvesso massimo ed ogni variabile ha un peso relativo sulla modalità

In base alla somma dei rinvessi si classifica il livello di complessità dell'oggetto:

$$(B \cdot MB \cdot MA \cdot A)$$

x cui ogni dimensione è ottenuta come somma pesata di diversi parametri

## (b) US image segmentation

L'analisi delle immagini è basata sui pixel ma lavorare sui singoli pixel non è ragionevole  
 $\rightarrow$  porta troppo rumore ed a dati poco affidabili

vengono perciò analizzati 4 pixel (a parte un) e si lavora sull'intensità di un'area media  
 $\rightarrow$  filtraggio  $\times$  diminuire il rumore

Grazie all'intensità diversa a seconda della zona, si individuano le 4 parti





⚠ Le caratteristiche correlate tra loro, non sono x forza ridondanti

Caratteristica informativa = è automaticamente correlata con il concetto/i decisionali ma è anche automaticamente scorrelata dalle altre caratteristiche

Gli **obiettivi** principali della feature selection sono:

- Diminuire la complessità del modello
- Migliorare le performance del sistema

Tra i **Benefici** della selezione delle caratteristiche ci sono:

- Visualizzazione dei dati facilitata
- Ridurre le misure e i risultati di archiviazione
- Ridurre i tempi di addestramento e di utilizzazione
- Migliorare le performance di predizione

La **Rilevanza** delle feature può essere considerata

FORTE
DEBOLLE

Una caratteristica fortemente rilevante implica che non può essere rimossa dal dataset, senza che ciò porti ad una perdita dell'accuratezza di predizione

Una feature debolmente rilevante può, ogni tanto, contribuire all'accuratezza, ma ciò dipende da quali feature sono considerate

IMPO: non è detto che se la caratteristica non

fornisce informazione individualmente, <sup>individualmente</sup> singolarmente, non possa

essere importante se presa insieme ad altre caratteristiche

→ non sempre le relazioni <sup>tra le</sup> caratteristiche sono lineari

quindi se tolgo 1 caratteristica di 2 correlate non è sempre la cosa più giusta!

Si sceglie il metodo di feature selection in funzione di quale filosofia si vuole adottare

ci sono 2 modi di affrontare il problema:

- Creare una classifica delle caratteristiche e prendere le prime, che presentano quindi correlazione massima → lavoro direttamente sulle variabili (K selezionate) / o su avere una soglia di importanza



→ **Randomised**: non si tagano o non si aggiungono feature si fanno dei sottoinsiemi (non troppo casuali) e si prova a vedere e valutare il risultato ottenuto



IL TRAINING SET È L'INSIEME DI TUTTI GLI ELEMENTI (con ognuno la sua classe corrispondente, se è una feature selection supervisioned)

IL PROCEDIMENTO È: • COSTRUIRE UN INSIEME DI FEATURE, POI VALUTARE IL RISULTATO CONFRONTOANDOLO CON LA CLASSIFICAZIONE ESATTA GIÀ PRESENTE. SI ANALIZZANO GLI STESSI RISULTATI CON + O - CARATTERISTICHE INSERITE. CI SI ASPETTEREBBE CHE CON + FEATURE SI OTTEGGANO I RISULTATI MIGLIORI, MA È COSÌ A CAUSA DELLE FEATURE RIDONDANTI E DI QUELLE CHE PROVOCANO RUMORE; SI VERIFICA QUINDI SEMPRE SE TOGLIENDO FEATURE LA SITUAZIONE MIGLIORA ANCORA O NO.

SI EFFETTUANO POI SEMPRE MISURE SULLA QUANTITÀ DELLE FEATURE E UNA **VALUTAZIONE FINALE**, CHE DI SOLITO È FORMATA DA 2 FASI:

- COMPARARE 1 2 CASE: PRIMA E DOPO LA FEATURE SELECTION (F.S.)
- COMPARARE 2 ALGORITMI DI F.S. × VERERE SE UNO È MIGLIORE DI UN ALTRO

BISOGNA ANCHE ANALIZZARE IL BIAS CHE SI PUÒ ANDARE A CREARE:

SPERO, USARE LO STESSO TRAINING SET SIA × LA FEATURE SELECTION, CHE PER L'ADDESTRAMENTO DEL CLASSIFICATORE, PORTA A PRESTAZIONI INFERIORI, PER CUI, SE SI PUÒ, SE NE PREFERISCE USARE 2 DIVERSI (SE NON È MOLTO PICCOLO E STANO IMPOSSIBILITATI).



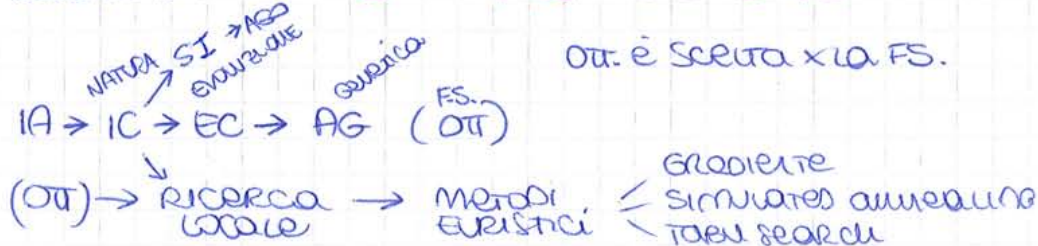
I metodi della CI sono ispirati alla **NATURA** e alla **BIOLOGIA**, cioè la CI mima i fenomeni naturali e biologici per sviluppare sistemi capaci di costruire soluzioni x problemi difficili e multifaccettati.

Sono tutti metodi molto ≠ tra loro ma allo stesso tempo collegati in que-  
 due volte si costruiscono sistemi dove si utilizzano più metodi.

- evolutionary computing
- reti neurali
- logica fuzzy
- swarm intelligence
- artificial immune system

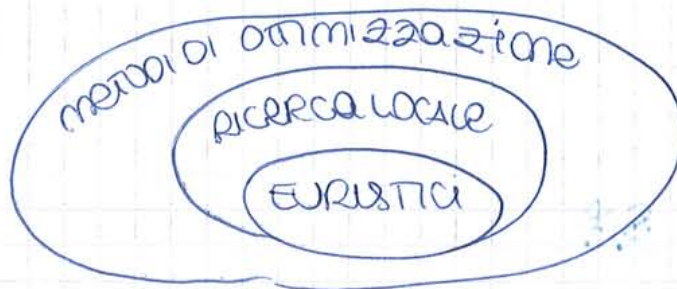
I metodi della CI (la maggior parte) non sono deterministici, ma contengono parti in cui si fa riferimento alla **PROBABILITÀ** (metodi probabilistici) che portano ad una maggiore difficoltà di implementazione ed al fatto che non si ottiene sempre lo stesso risultato, dato lo stesso input! Questi metodi vanno sotto il nome di **SOFT COMPUTING**. Tecniche da ≠ paradigmi possono anche essere combinate x creare sistemi ibridi.

Per la feature selection si usa il metodo degli **ALGORITMI GENETICI** che fanno parte della **EVOLUTIONARY COMPUTATION!** essa ha come obiettivo di mimare i processi dell'evoluzione naturale che ha come idea di fondo che sopravvive il più adatto all'ambiente, mentre il debole muore. E molti altri metodi che condividono una serie di caratteristiche e sono tutti utilizzati x risolvere problemi di **OTTIMIZZAZIONE**





⚠ perché esistono dei sistemi che evolvono nel tempo  
ex: voli interrotti x una nevicata  
se ne trova una soluzione immediata, non x forza la migliore



L'evolutionary computing (EC) si riferisce a sistemi computer-based problem solving che usano modelli computazionali dei processi di **evoluzione**, come la selezione naturale (dove sopravvive il migliore) e la riproduzione.

I metodi che si riferiscono alla ricerca locale usano est. idea:

- Partono da soluzioni (popolazione iniziale)
- Generano nuove soluzioni
- Ricostruiscono la popolazione

= una popolazione di soluzioni candidate, e  
evoluta nel corso di iterazioni successive  
di variazione e selezione casuale

Questi metodi sono caratterizzati dal fatto che fanno tutti uso del concetto di **PROBABILITÀ!**

L'evoluzione infatti non è deterministica!

Si può quindi trovare una soluzione finale diversa, anche rispetto allo stesso inizio.

Quindi, un **EA** algoritmo di evoluzione (genetico) è

una ricerca stocastica di un'ottima soluzione

(poiché la funzione ha + ottimi, si va in senso **probabilistico**)  
e si possono riconoscere ottimi  $\neq$

L'algoritmo migliore è quello che permette una variabilità non troppo eccessiva, cioè dato uno stesso inizio, porta quasi sempre alla stessa fine! non sempre a 1 diversa.



---

### Algorithm 8.1 Generic Evolutionary Algorithm

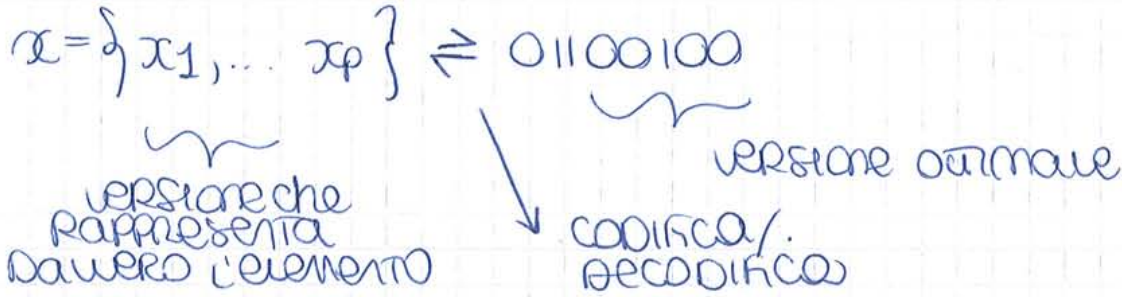
---

Let  $t = 0$  be the generation counter;  
Create and initialize an  $n_x$ -dimensional population,  $\mathcal{C}(0)$ , to consist of  $n_s$  individuals;  
**while** *stopping condition(s) not true* **do**  
    Evaluate the fitness,  $f(\mathbf{x}_i(t))$ , of each individual,  $\mathbf{x}_i(t)$ ;  
    Perform reproduction to create offspring;  
    Select the new population,  $\mathcal{C}(t + 1)$ ;  
    Advance to the new generation, i.e.  $t = t + 1$ ;  
**end**

---



Deve  $\exists$  una regola che permette la codifica e decodifica della soluzione (=stringa binaria)



2 categorie di problemi:

a) Rappresento la caratteristica con presenza (1) o assenza (0)

Si hanno  $p$  caratteristiche, a seconda che ci sono o no ho una combinazione binaria  $\neq$

⚠️ una famiglia!  
 ci sono problemi dove <sup>non</sup> tutte le soluzioni sono possibili si mettono dei vincoli in base a cosa serve  $x_i$  si confronta se la soluzione vale con i propri vincoli

b) Determinare 1 numero di bit  $\times$  ogni variabile (caso) e codificarne il valore attraverso il numero che c'è nella rappresentazione binaria/decimale

associa ad ogni struttura un numero da 1 a 100  
 vedrai ogni bit servano ad esempio  $\times$  rappresentare 99

15 sezioni. ognuna delle quali rappresenta una struttura con un numero da 1 a 100

il vincolo è che non si può avere 2 volte la stessa struttura



non  $\exists$  una correlazione lineare tra  $N_S$  e  $N_{iterazioni}$

$\left\{ \begin{array}{l} N_S \uparrow, N_{iter} \downarrow \\ N_S \downarrow, N_{iter} \uparrow \end{array} \right.$

ex:  $\begin{array}{cc} 100 & ; & 200 \\ \uparrow N_S & & \downarrow N_{iter} \\ 200 & ; & 150 \end{array}$

non sono dati dal metodo ma bisogna scegliere ogni volta

## ● Funzione di Fitness (FF)

Rappresenta l'obiettivo, ciò che si vuole ottimizzare

è una funzione usata x quantificare quanto buona è una soluzione, rappresentata da un cromosoma

Soltanto è una funzione matematica, ma può anche essere una f. logica, un modello di simulazione

Soltanto, è una misura assoluta di fitness, ma se non si trova ost. funzione se ne usa una relativa.

$\exists$  diversi tipi di problemi di ottimizzazione ed ognuno ha un'inferenza nella formulazione della FF:

### ● Ott. senza vincoli (unconstrained)

la + sempre  
massimizza o minimizza  
FF è la funzione obiettivo

ex: migliorare un servizio

### ● Ott. con vincoli (constrained)

massimizza o minimizza senza superare un tot  
FF. contiene 2 obiettivi: il principale e una funzione di penalità

ex: migliorare un servizio senza superare un monte che

### ● Ott. con più obiettivi (MOP)

+ complessa  
nella FF bisogna rappresentare tutti gli obiettivi e x farlo ci sono  $\neq$  modi  
• somma pesata  
• classificazione

ex: massimiz. qualità e minimiz. costi

### ● Soluzioni che si modificano nel tempo Problemi dinamici e rumorosi

meno info,  
+ complessi  
la FF deve tenere in considerazione la variabile del tempo



Il CO può essere

- ad un punto fisso (punto di taglio) costante x tutta l'iterazione
- a punto di taglio generato casualmente

maschera 

0	0	1	1	1
---	---	---	---	---

 → scambio di bit vicini

maschera 

0	0	1	1	0	0
---	---	---	---	---	---

essendo un operatore variabile, è difficile usare la ripetibilità delle soluzioni

una volta generate le soluzioni, si prendono tutte se l'ott. non è vincolata, o si visitano ognuna x vedere se rispetta i vincoli, se l'ott. è vincolata

## ● OPERATORI DI SELEZIONE

- PRINCIPALE OPERATORE
- deve enfatizzare le soluzioni migliori
- la selezione si applica in 2 momenti

### ● CREAZIONE di una nuova popolazione a partire dai figli

sceglia solo dai figli o anche coi genitori  
 deve assicurare che i buoni individui sopravvivano alle prossime generazioni  
 bisogna evitare che si incastra in un massimo locale

### ● RIPRODUZIONE, scelta dei possibili genitori

si applicano su operatori di riproduzione.  
 sendo prendere sia i genitori migliori (x co) sia su individui con mutazioni (best)

la loro caratteristica principale è la **pressione selettiva**

- = tempo (misurabile in numero di iterazioni) nel quale si arriva ad avere una popolazione formata da un numero preponderante di individui (uniforme)
- = velocità alla quale la soluzione (migliore) occupa il la popolazione

se alta, potremmo bloccarci in un ottimo locale, senza uscirne più, a tutte le soluzioni gli si ottengono vicine, magari in un ottimo lontano da quello globale  
 meglio bassa se si vogliono buone ripetibilità e buone soluzioni



se alta, limita quindi le abilità di esplorazione della popolazione



## ● Condizioni di STOP

Gli operatori di evoluzione sono applicati iterativamente fino a che le condizioni di STOP non sono soddisfatte

- limitare il numero di iterazioni (generazioni)
- $\neq$  IL numero di valutazioni della FF

non deve essere un limite troppo basso, si bisogna poter esplorare il lo spazio di ricerca

## ● PARAMETRI di CONTROLLO e TUNING

PM = tasso di mutazione (↓)

PC = tasso di crossover (↑)

possono migliorare la performance

Sovramente sono mantenuti costanti ma si possono anche usare parametri che cambiano dinamicamente



condizioni di stop su numero di iterazioni

## Stopping Conditions

The evolutionary operators are iteratively applied in an EA until a stopping condition is satisfied.

The simplest stopping condition is to limit the number of generations that the EA is allowed to execute, or alternatively, a limit is placed on the number of fitness function evaluations. This limit should not be too small, otherwise the EA will not have sufficient time to explore the search space.

In addition to a limit on execution time, a convergence criterion is usually used to detect if the population has converged.



HA una serie di  
Grafici che permettono di capire  
Cosa succede

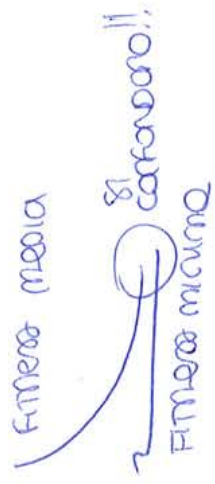
USO come metodo di selezione  
ovvero di prendere i migliori  
- solver - sempre auspicabili

**GAFF OPTIMIZATION**

TOOL  
che misura  
l'asportazione

Rischio delle  
funzioni che  
NOTES PROBABILI

= sommare  
particolare  
di alcuni ricercatori



Modificai  
settaggi  
↓  
problem setup &  
results

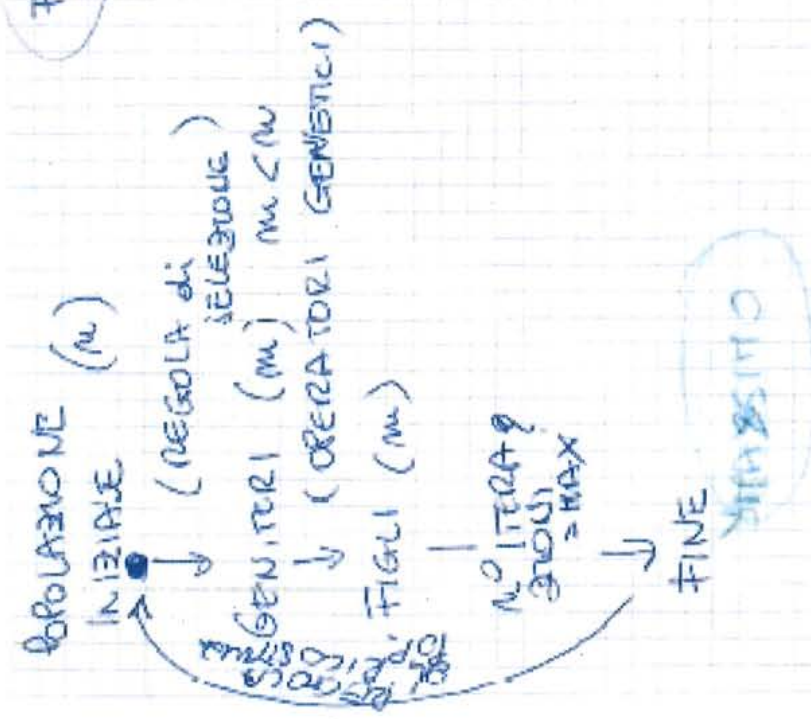
Implementazione particolare

- 0 element migliori
- 50% CO
- 80% mutazione

elit cut specify 0  
 CO fraction 0,5  
 mutation rate 0,8



**FITNESS**



- COME IMPLEMENTARE GA
- 1] FITNESS  
 CODIFICA SOLUZIONE  
 → SOTTARE 4° SOLUT.
  - 2] N° DI INDIVIDUI POPOLAZIONE → N° ITERAZIONI  
 FISSO O VARIABILE  
 TESTATO SPERIMENTALMENTE
  - 3] GENITORI → NUMEROSITÀ  
 → REGOLA SELEZIONE
  - 4] DEFINIRE IMPLEMENTAZIONE OPERATORI GENETICI  
 → VERIFICA SPERIMENTALE
  - 5] RICOSTRUZIONE POPOLAZIONE → REGOLA
- ⇒ STABILITÀ DELLA SOLUZIONE







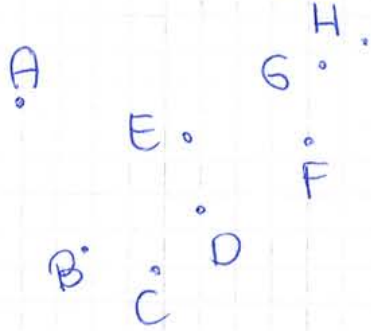


o costruite

**vicinato** = insieme delle soluzioni raggiungibili dalla soluzione corrente attraverso un'operazione elementare  
 = soluzioni che distano un certo intervallo dalla soluzione

esempio: il vicinato del GA sono tutte le soluzioni ottenute applicando gli operatori genetici (figu)

esempio: percorso migliore x visitare i pazienti a domicilio, andando da tutti nel minor tempo possibile



1<sup>a</sup> soluzione = la funzione obiettivo e la somma degli spostamenti in un tempo  $t_1$   
 ABCDEFGH

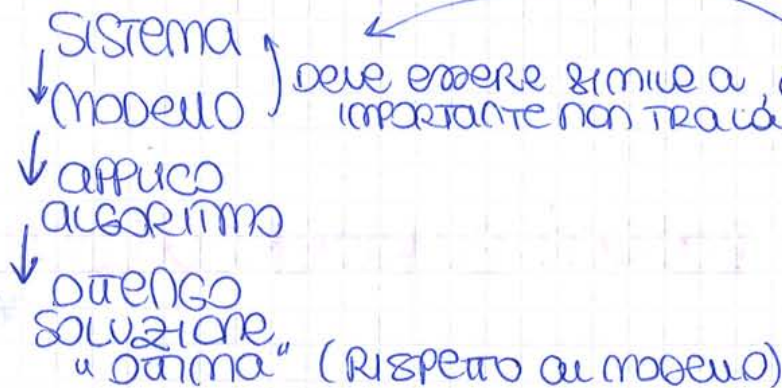
il vicinato si ottiene spostando una lettera (A) in tutte le posizioni o cambiando sempre con operazioni elementari

- BACDEFHG
- CBADEFHG
- DBC AEFHG
- EBCDAFGH ...

dopo A scambio B ecc!

un vantaggio è che si possono costruire funzioni obiettivo complesse quanto si vogliono, x cui si rappresentano meglio i reali obiettivi che si vogliono raggiungere

ogni tanto infatti ci sono relazioni matematiche che obbligano a semplificare e trascurare alcuni parametri



deve essere simile a, ma non sarà mai =, importante non trascurare gli elementi importanti

una fine si prende la soluzione e la reinserisco nel sistema



# METODO del SIMULATED ANNEALING

= Ricottura simulata / Raffreddamento del metallo

Le idee alla base del annealing (ricottura) sono state riutilizzate x ricercare le soluzioni ammissibili di un problema di ottimizzazione, con l'obiettivo di convergere ad una soluzione ottimale

annealing = quando un metallo si raffredda, gli atomi si posizionano in una zona dello spazio

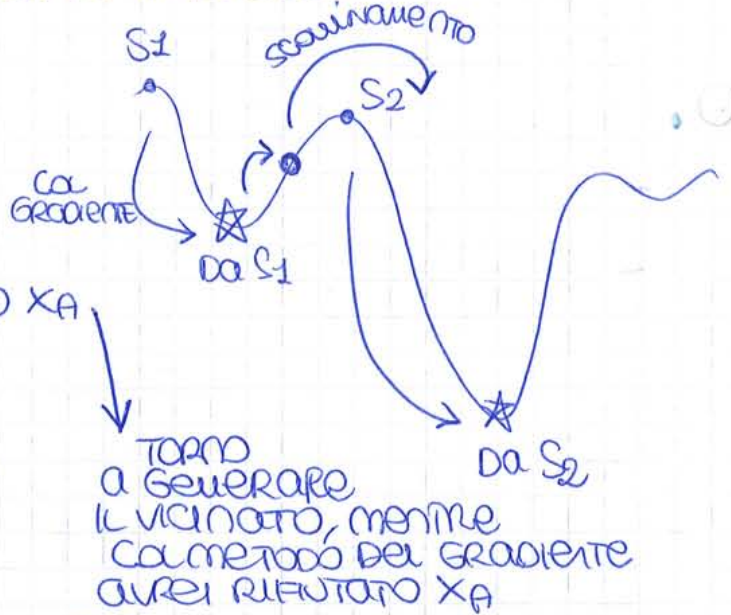
- è un metodo stocastico = probabilistico, basato su singola soluzione
- supera il problema della soluzione iniziale del gradiente !!
- è una variante della ricerca locale, in cui un sottoinsieme delle soluzioni ammissibili è esplorata spostando ripetutamente dalla soluzione corrente ad una vicina
- idea delle **UPHILL MOVES** = mosse che scollinano

Queste mosse fanno sì che non ci si blocchi in un ottimo locale, ma può partire sia verso un ottimo peggiore che uno migliore

Queste mosse devono essere usate con moderazione ed in modo controllato

sono quindi permessi movimenti di risalita, ma la loro frequenza è governata da una funzione di probabilità che cambia mentre l'algoritmo procede

- ↓ Soluzione iniziale RANDOM
- ↓ Generazione del vicinato
- ↓ si seleziona nel vicinato una soluzione random  $x_A$
- ↓ se  $f(x_A) < f(x_I)$ , accetto  $x_A$
- ↓ se  $f(x_A) \geq f(x_I)$ ,  
usa una probabilità che dice se accettare la soluzione o rifiutarla
- ↓ se la rifiuto, seleziono una nuova  $x_A$
- ↓ se la accetto, genero il vicinato rispetto a questa nuova soluzione



La probabilità utilizzata diminuisce con l'aumentare delle interazioni, altrimenti e come se mi movessi a caso senza un fine



Si deve vietare lo scambio almeno x  
l'iterazione successiva e oltre fino a  
che una soluzione con lo scambio ha un  
valore di funzione obiettivo non ancora  
ottenuto per tutte le soluzioni candidate finora  
→ cioè è migliore e dà un vantaggio  
sufficiente da compensare il divieto

- Metodo che non si usa mai x medical o health
- è un metodo mai usato per problemi di
  - scelta del cammino più breve
  - segnalazione / localizzazione
  - soluzione con non molti elementi (x è lento) } x essere vantaggioso

## CARATTERISTICHE PRINCIPALI DEL TS

TS inizia con una soluzione iniziale che è usata come soluzione corrente

La soluzione migliore nel vicinato è selezionata come la nuova soluzione corrente solo se non sta migliorando la soluzione corrente

x evitare cicli di generazione  $\infty$ , TS rigetta i vicinati che sono già stati precedentemente selezionati.

TS mantiene memoria delle soluzioni o dei movimenti recentemente applicati, che è chiamata lista tabù. La lista tabù rappresenta la memoria a breve termine e solitamente contiene un numero costante di movimenti.

Un movimento della lista tabù non è accettato, ma x certe condizioni chiamati criteri di aspirazione, questo movimento può essere accettato. Le soluzioni ammissibili nel vicinato sono quelle che non appartengono alla lista tabù o appartengono ai criteri di aspirazione.

alcuni meccanismi avanzati sono comunemente introdotti nell'algoritmo tabù search x dialogare con:

→ intensificare la ricerca in regioni locali

- intensificazione (memoria a medio termine):

la MLT immagazzina le soluzioni migliori trovate durante la ricerca. L'intensificazione ha il ruolo di sfruttare l'informazione delle migliori soluzioni trovate x guidare la ricerca in regioni vicine + grande esplorazione dello spazio

- Diversificazione (memoria a lungo termine):

la MLT immagazzina le informazioni delle soluzioni visitate durante la ricerca



# ACO

## OTTIMIZZAZIONE DELLA COLONIA DELLE FORMICHE

- METODO PROBABILISTICO
- SI PARTE DA UNA POPOLAZIONE

- ISPIRATO ALL'OSSERVAZIONE DEL COMPORTAMENTO DELLE FORMICHE
- Hanno ≠ comportamenti collettivi:
  - SUMENTAZIONE
  - DIVISIONE DEL LAVORO
  - COSTRUZIONE DEI NIDI
  - ORGANIZZAZIONE DEL UNITERO
  - CURA DELLA NIDIATA
- È una forma di comunicazione indiretta tra le formiche chiamata STIGMERGIA
- Riescono a comunicare a distanza grazie al **FERORMONE**



- DAGLI STUDI SULLA COMUNICAZIONE FEROMONALE SONO STATI SVILUPPATI I PRIMI MODELLI ALGORITMICI DEL COMPORTAMENTO DI FORAGGIAMENTO = ABILITÀ DELLE FORMICHE DI TROVARE IL PERCORSO + BREVE TRA IL NIDO E LA FONTE DI CIBO.
- Gli algoritmi sono basati su 3 step:
  - all'inizio c'è un'attività caotica e casuale nella ricerca del cibo
  - quando la fonte di cibo è stata localizzata, le attività diventano più organizzate con sempre + formiche che seguono lo stesso percorso verso la fonte di cibo
  - presto tutte le formiche seguiranno lo stesso, + breve percorso
- Si inizializzano più formiche (ex: 20)
  - ↓
  - Quando tutte sono terminate, si ottegono 20 soluzioni
  - ↓
  - si calcola la fitness e si incrementa la variabile ferormone
- il ferormone è x0 una sostanza volatile, cioè ha un'azione decrescente nel tempo (processo di evaporazione) mentre viene aumentato se le formiche trovano cibo (processo di rinforzo)



## esempio di possibili implementazioni: [DIPOSO]

ci sono sempre 3 fasi:

- Inizializzazione  $\leftarrow$  Fermare Formiche
- costruzione della soluzione  $\leftarrow$  evaporazione
- aggiornamento del ferormone  $\leftarrow$  RINFORZO

Risolvono problemi generali come trovare il percorso più corto tra 2 nodi su un grafico

USATI X: Problemi di ottimizzazione percorso minimo  
 Navigatore del telefonino  
 Commesso viaggiatore

NO Feature selection

## Gestione dei vincoli

Molti problemi di ottimizzazione continui o discreti sono vincolati.

una funzione obiettivo con dei vincoli non è facile da gestire

I vincoli possono essere:

- Lineari
- non lineari
- di disuguaglianza
- di uguaglianza

←  
 Soltanto se non si mettono se non strettamente necessari. Altrimenti il problema non è abbastanza flessibile e non si trova una soluzione

esempi: Piano investimenti  
 max numero apparecchiature,  $\Delta$  Budget

30 richieste:  $x_1, \dots, x_{30}$   
 Ogna con un costo:  $c_1, \dots, c_{30}$

Vettore: 1 richiesta soddisfatta  
 0 non comprato

si vuole  $\max \sum x_i, \Delta \sum c_i x_i \leq \text{Budget}$

ci sono 2  $\neq$  approcci:

- agiscono sulla rappresentazione delle soluzioni o delle funzioni obiettivo
- non convergono a // // ex ottimizzazioni multi-obiettivo.



- **Quality-based pheromone update:** This strategy updates the pheromone value associated with the best found solution among all ants (or the best  $k$  solutions where the number  $k$  is smaller than the number of ants) [218]. The added values depend on the quality of the selected solutions. For instance, for each component belonging to the best solution  $\pi^*$ , a positive value  $\Delta$  is added:

$$\tau_{i\pi^*(i)} = \tau_{i\pi^*(i)} + \Delta, \quad \forall i \in [1, n]$$

- **Rank-based pheromone update:** The best  $k$  ant solutions are allowed to update the pheromone with an amount depending on the ranking of the solutions [95].
- **Worst pheromone update:** The ant generating the worst solution will decrease the pheromone trails related to the components of the solution [154].
- **Elitist pheromone update:** The best solution found so far will update the pheromone to enforce an intensification of the search [728]. A minimum (resp. maximum) value for the pheromone is defined such as each choice has a minimum (resp. maximum) probability to be selected.



## Constraint handling

*Many continuous and discrete optimization problems are constrained.*

Dealing with constraints is a not trivial problem.

Constraints may be of any kind:

Linear or nonlinear

Equality or inequality

Constraint handling strategies may be related to two different approaches:

- a) Strategies that act on the representation of solutions or objective function
- b) Strategies not directly related to the representation of solutions or objective function, like multiobjective optimization and coevolutionary models.



∃ infatti ≠ modi x codificare una soluzione,

EX: STRINGA BINARIA

PER IL PRIMO APPROCCIO CI SONO 4 STRATEGIE:

- MATH
- S. DI RIGETTO: si controlla se la soluzione non rispetta i vincoli e nel caso la si elimina e se ne genera un'altra
  - S. DI PENALIZZAZIONE: si aggiunge una funzione obiettivo che viene incrementata o diminuita di un valore detto penalità. La funzione obiettivo penalizzerà le soluzioni irrealizzabili, da cui però si possono trovare soluzioni migliori
  - S. DI RIPARAZIONE: si segue una richiesta x rientrare nel budget si trasforma una soluzione non fattibile in una fattibile
  - S. DI PRESERVAZIONE: rappresentazione e operatori specifici assicurano la generazione di soluzioni fattibili



Spesso infatti è impossibile distinguere i membri che appartengono ad una classe da quelli che non appartengono

↳ ex: quando una collina diventa una montagna? non si riesce a dirlo ma si sa di cosa sta parlando

La logica convenzionale o booleana utilizza distinzioni nette: **CONFINI NETTI** tra le classi.

auto > 1,8m      se Tom è auto 1,79m...  
Basso < 1,8m

La logica fuzzy invece non stabilisce un confine netto ma un **graduale passaggio**

La LF permette di rappresentare 2 tipi di incertezze

↳ misure (tecniche)  
conceiti (vaghezza di non definizione)  
↓  
anche se nel senso comune ci si capisce

### DEFINIZIONI:

- **Vaghezza**: ogni concetto è detto vago se non è definito da confini netti
- **Imprecisione**: sempre una caratteristica del linguaggio, più in particolare riguarda i concetti misurabili (proprietà metriche)
- **incertezza**: riguarda non il linguaggio, ma il rivendicare se una proposizione vale o no

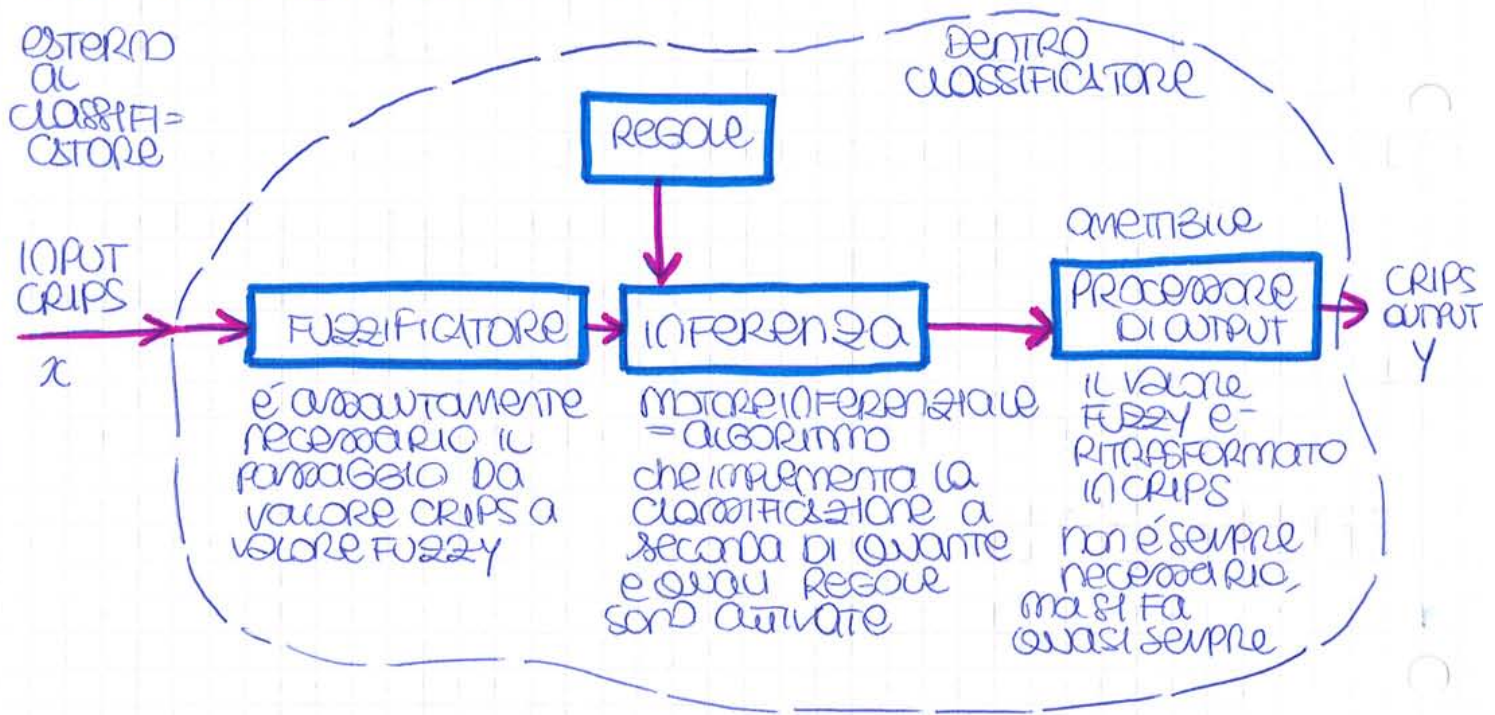
non useremo la fuzzy logic x la classificazione, ma in realtà copre aspetti molto diversi che si combinano tra loro senza dei confini ben definiti, ma sfocati e che si sovrappongono

La FL nel suo senso ampio ha infatti **4** faccette principali:

- **FL/L**: **Strutturatura logica**: legata al ragionamento è sostanzialmente la logica fuzzy in senso stretto, un sistema logico che, a differenza di quelli convenzionali, si rivolge ad una formalizzazione di modi di ragionamento che sono indicativi e non esatti



# CLASSIFICATORE FL



ci sono 6 FASI x COSTRUIRE IL CLASSIFICATORE

- ① modellizzazione delle caratteristiche (concetti) per mezzo delle funzioni di appartenenza
- ② costruzione dell'insieme di funzioni x passare da input crisp a fuzzy
- ③ elicitation e costruzione delle regole fuzzy
- ④ definizione della modalità di aggregazione
- ⑤ definizione del metodo da utilizzare x defuzzificare il risultato
- ⑥ validazione e tuning del sistema
- ⑦ validazione

① La **MEMBERSHIP FUNCTION** è la funzione di appartenenza

**MF** esempio: età  $x = \{x_1, x_2, \dots, x_n\}$   
 $\downarrow$   
 età

l'età può essere rappresentata in classi che ne rappresentano il concetto



Come si costruiscono le membership function?

Ci sono 3 approcci:

① **LIKELIHOOD view**: Vista della verosimiglianza

Si intervista un gruppo di persone sufficientemente elevato

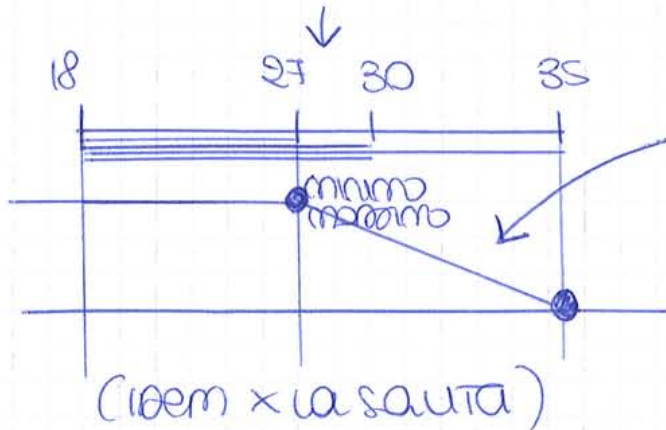
ex: 35 anni e' G.O.D.? Si segnano le risposte  
 JAH e' alto? SI-NO

② **RANDOM SET view**: Vista di insiemi casuali

Si intervistano chiedendo i range, che non x forza debbono essere continui

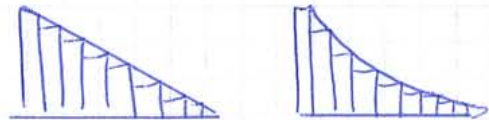
ex:

1	18-27	25-31	60
2	$\leq 30$	30-60	$\geq 60$
3	18-35	40-60	$\geq 60$
4	18-30	35-55	$\geq 60$



Si individuano i punti poi si collegano con una retta, un exp...

xx il passaggio da 1 a 0 puo' essere to-lineare



Il numero di persone intervistate dipende da ≠ aspetti (ad esempio possibilità economiche etc...)

- poche xsn → si riesce a marcare solo 1 o 0
- tante xsn → si può anche decisa e salta

oppure invece che da persone si parte da esempi (valori min e max che la variabile assume) → inizio MF

③ **Similarity view**: si usano dei prototipi di riferimento

si chiede ad un esperto che valori assume la MF nei vari punti

Bisogna decidere di quanti punti si ha bisogno



Costruire le MF vuol dire individuare questi punti



La PROBABILITÀ è definita dal numero di eventi favorevoli verificati sul numero totale di eventi avvenuti (a posteriore)

$$p(\text{moneta}) = \frac{1}{2}, \quad p(\text{dado}) = \frac{1}{6}$$

$p(\text{persona} > 45 \text{ anni e diabetica}) = ?$  serve un campione rappresentativo

La PROBABILITÀ si utilizza se l'evento non si è verificato o l'informazione non è disponibile x sapere la PROBABILITÀ di accadimento

↳ La probabilità non serve + se l'evento si è verificato, mentre il grado di appartenenza ad una funzione di app è sempre lo stesso, e ancora rilevante anche dopo l'evento

si può usare x una PROBABILITÀ associata ad un evento fuzzy, ricordando che la PROBABILITÀ  $\neq$  dal grado di appartenenza

INOLTRE, la PROBABILITÀ assume indipendenza dagli eventi, la LF no.

La PROBABILITÀ assume un modello di mondo chiuso di cui sa tutto, e si basa su misure di frequenza di eventi che si verificano  
 La FL invece si basa su misure descrittive del dominio (in termini di MF) invece di misure soggettive di frequenza.

2

Le REGOLE FUZZY sono delle condizioni che si possono verificare o no rispetto le antecedenti

RF

REGOLE CLASSICHE

Bisogna mettere delle soglie  
 $> 100 = \text{veloce}, \dots$   
 ma velocità anche  $\times 90!$

REGOLE FUZZY

superano il concetto di soglia che non è stabilita o condivisa da tutti  
 alta = non si da un numero preciso



$x$  è RAPPRESENTATO DA 3 MF :  $A_1, A_2, A_3$

$y$  è RAPPRESENTATO DA 2 MF :  $B_1, B_2$

### REGOLA ①

IF  $x$  IS  $A_3$  ( $\mu_{A_3}(x) \neq 0$ )  
 OR  $y$  IS  $B_1$  ( $\mu_{B_1}(y) \neq 0$ )  
 THEN...

**OR**  $\Rightarrow$  almeno 1 MF deve avere valore  $\neq 0$

Caso (39; 0,7)  $\rightarrow$   $\mu_{A_3}(x) = 0,9$   $\rightarrow$  REGOLA ATTIVA  
 $\mu_{B_1}(y) = 0,2$

Caso (2; 0,5)  $\rightarrow$   $\mu_{A_3}(x) = 0$   $\rightarrow$  REGOLA ATTIVA  
 $\mu_{B_1}(y) = 1$

### REGOLA ②

IF  $x$  IS  $A_2$  ( $\mu_{A_2}(x) \neq 0$ )  
 AND  $y$  IS  $B_2$  ( $\mu_{B_2}(y) \neq 0$ )  
 THEN...

**AND**  $\Rightarrow$  tutte le MF devono avere valore  $\neq 0$

### REGOLA ③

IF  $x$  IS NOT  $A_1$  ( $\mu_{A_1}(x) = 0$ )

**NOT**  $\Rightarrow$  deve avere MF = 0

Le REGOLE sono esaminate 1 ad 1.

PER OGNI REGOLA, LA LOGICA FUZZY:

- ① PRIMA esamina / valuta gli antecedenti
- ② POI RIVERSA I CONTENUTI sulle conseguenze  
 cioè applica i risultati alle conseguenze

nei sistemi classici : A vera  $\rightarrow$  C vera

nei sistemi FUZZY : l'antecedente è una dichiarazione fuzzy, tutte le regole sono prese / hanno valore  $\times$  una certa misura / parzialmente

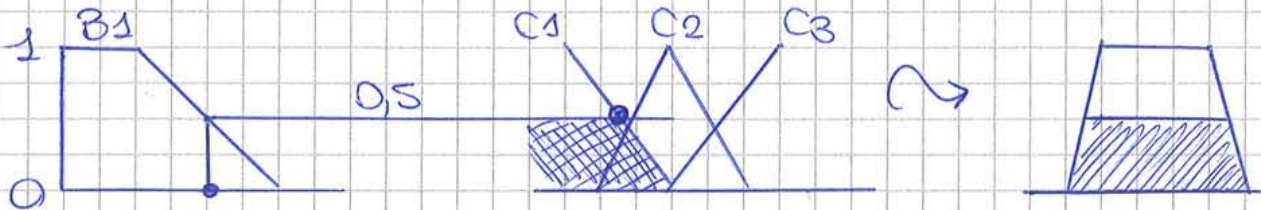
A vera  $\times$  certi gradi di appartenenza  $\rightarrow$

C vera  $\times$  lo stesso grado



● **clipping** è il metodo + comune di collegare la regola conseguente con il vero valore della regola antecedente → si taglia la MF conseguente al livello della "verità" antecedente anche chiamata correlazione minima

l'insieme fuzzy perde dell'informazione, ma si preferisce xk usa una matematica - completa e + veloce e un risultato + facile da defuzzificare

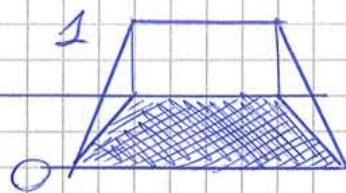


si taglia la MF di output!

● **scaling** o prodotto di correlazione

preleva in modo migliore la forma originale dell'insieme fuzzy

la MF originale della regola conseguente è adattata moltiplicando tutti i suoi gradi di appartenenza x il valore vero della regola antecedente così si perde meno informazione



comporta un diverso modo di aggregazione

in generale un sistema fuzzy incorpora + di una regola i risultati della "sparo" di ciascuna regola sono aggregati in un singolo insieme fuzzy di output e poi quest risultato è defuzzificato in un singolo amero la maggior parte delle volte.



### ● inferenza mamdani

+ comunemente usata

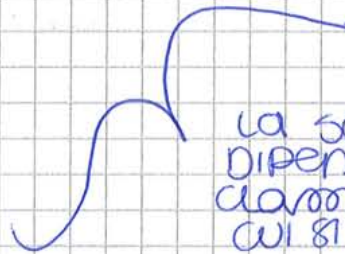
è un processo formato da 4 step

- Fuzzificazione delle variabili di input
- valutazione delle regole
- aggregazione degli output delle regole
- defuzzificazione finale

Processo x trovare il centroide non computazionalmente efficiente

### ● inferenza zugenò

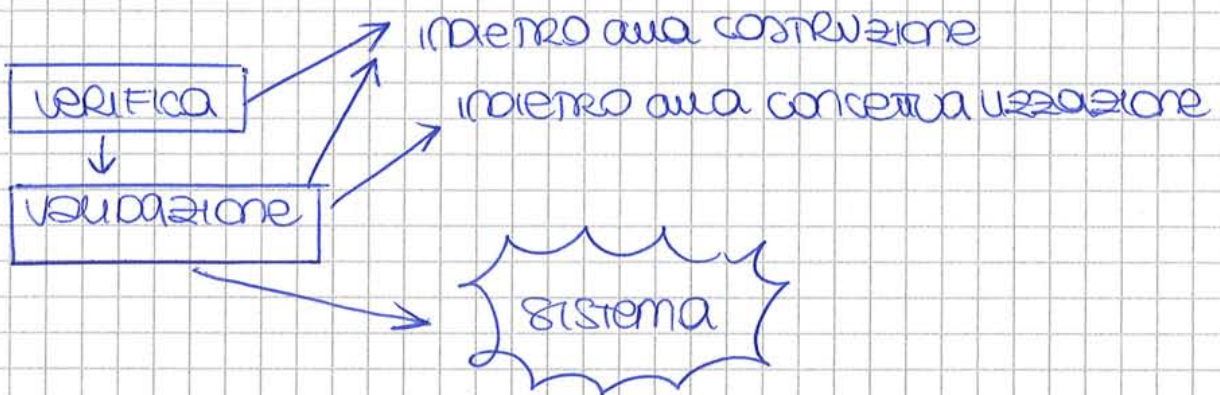
Come conseguenza non usa un fuzzy set ma una funzione matematica sulle variabili di input  $f(x,y)$



la scelta dipende dal classificatore di cui si ha bisogno

## 6

### Processo di validazione



verifica e validazione sono 2 cose ≠

● **VERIFICA** : soddisfa le specifiche?

Cioè se si ha fatto la cosa giusta!

● **VALIDAZIONE** : tutti gli elementi devono essere classificati (anche se non si) avrà mai il 100%

con campione rappresentativo!



Se fuori dalla diagonale è tutto = 0, il classificatore ha classificato correttamente il 100% del set, ma non succede mai!

esempio:

A = 120 : 101A, 12B, 7C  
 B = 96 : 0A, 90B, 6C  
 C = 102 : 40A, 10B, 52C

	A	B	C
A	101	0	40
B	12	90	10
C	7	6	52

Tutto ciò che è fuori dalla diagonale è un errore e deve essere minimizzato!

non si arriva ad un risultato perfetto, ma si controllano se le prestazioni sono adeguate/accettabili x il proprio problema/scopo!

ex: C = 40A troppo alto  
 B = 6C accettabile

Gli elementi della matrice possono essere valori assoluti o relativi.

Se non si riescono ad avere risultati accettabili bisogna rivedere molto indietro nel proprio progetto!

Si differenziano il training set ed il test set

**TRS** = insieme degli elementi usati x costruire il classificatore (dati/esempi)

**TES** = insieme dei dati x testare il classificatore

come sono costruiti?

si parte da una popolazione da cui si estraggono gli elementi in  $n$  modi (vedi sube "estrazione"  $\leftarrow$ ) x costruire il TRS, poi si sceglie il TES.



esempio:

$$x_i = 1, \dots, n$$

① TOGLIO  $x_1$ , TRS =  $(2, \dots, n)$  → CLASSIFICO  $x_1$

② TOGLIO  $x_2$ , TRS =  $(1, 3, \dots, n)$  → //  $x_2$

e si va avanti a riempire la matrice fino a  $x_n$ , poi si sommano su elementi

in ogni casella della matrice →

la matrice è ≠ da quella costruita nei

2 casi precedenti,  $x_c$  qui include  $n$  classificatori ≠!

	A	B
A	$x_1$	
B	$x_2$	

L'ipotesi fondamentale è che il singolo elemento non modifichi così tanto il classificatore, quindi

$$\text{class}(n) = \text{class}(n-1)$$

|| ci si aspettano una prestazioni inferiori rispetto a  
TES = TRS

**SCREENING TEST** = chiedere se si è affetti o no da una malattia

se il classificatore è costruito a supportare uno screening test, sono definiti:

● SENSIBILITÀ

= PROBABILITÀ DI AVERE UN TEST POSITIVO SE LA PERSONA HA LA PATOLOGIA (VV) → max

● SPECIFICITÀ

= PROBABILITÀ DI AVERE UN TEST NEGATIVO SE LA PERSONA NON È MALATA (FF) → max

● FALSO NEGATIVO

= persona ammalata ma con test negativo = male!  
↳ da ricorre assolutamente! (F → V!)

● FALSO POSITIVO → viene richiesto con successivi controlli

= persona non ammalata, ma con test positivo (V → F!)



# CLUSTERING

Differenza fondamentale:

- in una classificazione le classi sono consecutive
- in un clustering NO

Metodo molto utile

Precede la classificazione:

- Data una classificazione supervisionata (ho a disposizione degli esempi classificati)

TRAINING SET } composto da elementi  $\{x_i\}$ ,  $C_i \rightarrow$  classe corretta

grazie alla presenza di  $C_i$ , si può costruire la confusion matrix

variabili usate x la classificazione

L'obiettivo è minimizzare l'errore tra la classificazione corretta e il risultato del classificatore:

$\min(\text{errore}) \rightarrow 0$  (=0 è un evento molto raro)

Tutta la costruzione del classificatore serve a minimizzare l'errore ed è possibile proprio se si conoscono già le classi corrette

avremo anche TEST SET }  $y: \{x_j\}, C_j$

- Se la classificazione non si conosce, non si conosce  $C_i$  e non si può minimizzare nessun errore se non si hanno termini di confronto!

$i: \{x_i\}, C_i$

Si usa il clustering x suddividere tutti gli elementi in un numero di sottogruppi (cluster)

Il clustering si usa quindi per

← estrarre nuova conoscenza  
 costruire regole di un classificatore fuzzy @  
 feature selection

Il difficile è definire quanti cluster si vogliono

nel caso @ i prototipi della classificazione sono le regole del classificatore fuzzy (il prototipo si ricostruisce con gli elementi che appartengono alla classe con certezza)

ma se non si conoscono i prototipi?

se non si conosce quante classi sono?



un metodo di clustering è caratterizzato da:

- una **misura di similarità**
- una **regola di classificazione**

e volendo anche

- come da un' iterazione alla successiva si costruiscono i prototipi delle classi

i passi che si seguono in un metodo clustering sono:

- Inizializzazione random  
(prototipi costruiti con elementi a caso)
- Iterazione
- Aggiornamento del prototipo con un metodo definito
- e così via...

Impossibilità di differenza tra:

- classificazione: le classi sono note ← sia il loro numero
- clustering: le classi sono sconosciute ← sia il loro prototipi

L'approccio comune è procedere con un metodo iterativo riassunto in 2 fasi:

- Trovare una divisione iniziale ragionevole (NLC) dove solitamente si ipotizza un numero di cluster e si trovano i prototipi in modo random
- Muovere gli elementi campione da un cluster ad un altro in modo da ridurre la funzione obiettivo

Questi metodi producono soluzioni sub-ottimali, non si è sicuri di trovare la soluzione migliore possibile (è un metodo euristico), ma sono trattabili computazionalmente.

I principali metodi iterativi possono essere divisi in 2 gruppi:

- ← Flat
- Gerarchici