METODI NON SUPERVISIONATI NELL'ANALISI ESPLORATIVA DI DATI DA DNA MICROARRAY

Stefano Rovetta
Dipartimento di Informatica e Scienze dell'Informazione, Università di Genova
Istituto Nazionale di Fisica della Materia, Unità di Genova

Francesco Masulli
Dipartimento di Informatica, Università di Pisa
Istituto Nazionale di Fisica della Materia, Unità di Genova

L'analisi genetica molecolare si svolge tradizionalmente attraverso uno studio molto dettagliato e focalizzato di un gene o di un numero ridotto di gruppi di geni. La tecnica recente dei microarray a DNA consente un nuovo metodo di indagine, basato sull'analisi contemporanea di migliaia di geni. È così diventato possibile eseguire una analisi di tipo esplorativo, estensiva, che fornisca indicazioni per focalizzare le successive indagini nelle direzioni più promettenti, oppure per lo studio delle complesse reti di interazione tra i geni. Questi attuali scenari sono tuttavia resi possibili solo dalla stretta coordinazione tra tecniche biomolecolari e algoritmi di elaborazione.

1 Introduzione ai microarray a DNA

1.1 *La tecnologia dei microarray*

Il Progetto Genoma diede luogo a un cambio di prospettiva nell'identificazione di geni. Diventava possibile concepire *screening* e analisi dei geni su larga scala, e addirittura operare analisi su interi genomi in un unico passo [32]. Lo sviluppo di questa tecnologia è stato reso possibile da varie innovazioni, tra cui due particolarmente importanti. La prima, messa a punto presso

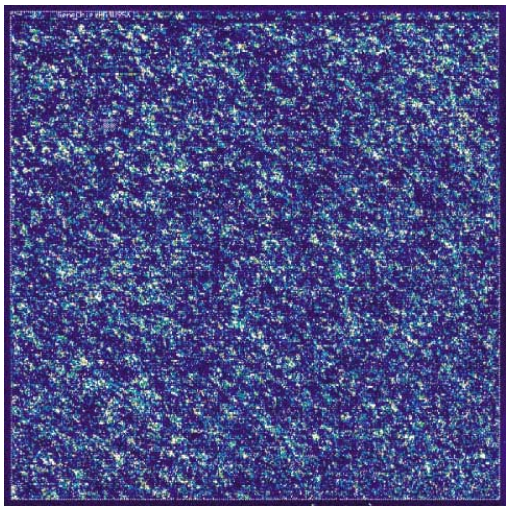


Fig. 1 – Un singolo microarray evidenzia l'espressione di migliaia di geni simultaneamente. Immagine fornita da Affymetrix.

l'Università di Stanford [30, 31], consente l'uso di supporti non porosi, come il vetro, che hanno grossi vantaggi nella miniaturizzazione e nel rilevamento tramite fluorescenza rispetto ai supporti a membrana porosa (nylon) adottati inizialmente. Con tale tecnica, che fa uso di uno *spotter* robotizzato, è stato possibile posizionare circa 10 000 cloni di DNA complementare, cDNA, su un singolo vetrino per microscopia, ottenendo quindi schiere ad altissima densità che possono essere rilevate con sonde (*probes*) etichettate per fluorescenza a due colori.

La seconda tecnica, che è stata sviluppata a partire dai processi produttivi in uso nella tecnologia dei semiconduttori [14, 21], è la sintesi diretta di oligonucleotidi su substrato di vetro attraverso la fotolitografia ad alta risoluzione. Questo secondo procedimento è utilizzato dalla società Affymetrix, che produce chip contenenti fino a 500 000 oligonucleotidi su un'area di 1,28 × 1,28 centimetri [1]. Si tratta di una tecnica molto versatile, per via della altissima densità possibile ma anche grazie al fatto che la sintesi diretta su chip permette di produrre qualunque configurazione a partire da basi di dati di sequenze geniche. Anche se la prima tecnica fu usata per chip basati su DNA complementare, e la seconda faceva uso di oligonucleotidi, tale distinzione non è precisa: oggi entrambe le categorie di array possono essere prodotte

attraverso spotting robotizzato. Per quanto riguarda la lettura delle informazioni, i microarray vengono predisposti in modo da rilevare differenze di ibridazione in due distinte configurazioni sperimentali, per esempio tessuto sano/tessuto tumorale. A seconda della tecnologia, si ottiene una immagine in cui la proporzione tra due colori (rosso e verde) fornisce la lettura richiesta, oppure si ricavano due letture distinte tra cui occorre effettuare il confronto.

Per concludere la panoramica sulle tecnologie disponibili, possiamo notare che, a seconda dell'applicazione, può essere comunque vantaggioso ricorrere ad array a minore densità su supporto a membrana porosa, che presentano maggiore semplicità costruttiva e minor costo. Ricordiamo poi che altre tecnologie si stanno rapidamente sviluppando sulla base di quella dei microarray a DNA, quali per esempio i microarray proteici e di tessuti.

1.2 *Applicazioni*

Quali sono gli usi possibili per questa potente metodologia di indagine? Anche se nuove applicazioni vengono proposte quasi quotidianamente, possiamo identificare due vaste classi applicative prevalenti.

- Screening dell'espressione genica. La maggioranza degli studi si concentra attualmente sull'uso dei microarray per l'analisi dei livelli di espressione del RNA, sia attraverso microarray a cDNA che con microarray a oligonucleotidi gene-specifici. Nel caso di organismi dei quali il genoma sia noto completamente, sono possibili rilevamenti su tutto il genoma: un esempio è dato dal lievito Saccharomyces cerevisiae. La tecnologia per effettuare lo stesso tipo di analisi per il genoma umano sarà disponibile nel prossimo futuro.

- Screening delle variazioni nel DNA. Questa applicazione è di portata molto generale e richiede microarray a oligonucleotidi. Consiste nel rilevare variazioni individuali nelle sequenze su larga scala, e trova impiego per esempio nella ricerca di mutazioni in geni di cui si conosce il coinvolgimento in determinate patologie, come il carcinoma mammario, o nella tipizzazione ad alta risoluzione del sistema HLA per individuare potenziali donatori perfettamente immunocompatibili in caso di trapianti.

Fig. 2 – Il prcedimento di analisi con microarray.

1.3 *I passi del procedimento di analisi*

Un esperimento di analisi con microarray a DNA coinvolge non solo la tecnica bioanalitica molecolare, ma anche successivi passi di estrazione ed interpretazione dell'informazione rilevata [27]. Possiamo succintamente descrivere il procedimento come illustrato in figura 2. Per assicurare uniformità e ripetibilità, sono stati proposti protocolli come quello denominato MIAME (Minimum Information About Microarray Experiments) [8, 20].

Dopo un'accurata *pianificazione* e lo *svolgimento dell'esperimento*, che è buona norma preveda sonde replicate e di controllo per tenere sotto controllo eventuali errori di rilevamento, si procede all'*acquisizione dell'immagine* attraverso scansione ottica a risoluzione appropriata, e alla memorizzazione delle immagini rilevate unitamente ai dati descrittivi dell'esperimento. Quindi occorre procedere all'*analisi dell'immagine* per identificare i valori di ibridazione a partire dalle caratteristiche visive, utilizzando i procedimenti usuali nell'analisi di immagini, ossia segmentazione, validazione e stima del grado di affidabilità dei valori ottenuti. Il risultato di questo passo può essere per esempio un file in un formato leggibile da un comune programma di foglio elettronico (spreadsheet). A questo punto i dati, ormai condensati nella forma di valori numerici, possono essere organizzati in tabelle e sottoposti a

elaborazione preliminare e normalizzazione, allo scopo di ridurre il più possibile le variabilità statistiche e portare i valori in forma adeguata a facilitare le successive elaborazioni (rappresentandoli per esempio su scala logaritmica).

La vera e propria analisi dell'esperimento inizia con l'*identificazione dei geni espressi differenzialmente*, attraverso valutazioni statistiche più o meno sofisticate. È quindi possibile procedere a una o più tra le successive fasi: l'*analisi esplorativa*, oggetto di questo capitolo; la *predizione delle classi* (o *classificazione*), per valutare quanto i dati rilevati siano correlati con le condizioni sperimentali oggetto dell'indagine (per esempio tessuto sano/tessuto tumorale); infine l'*analisi di pathway*, per comprendere le interazioni tra diversi geni e i complessi meccanismi che attivano e regolano le funzioni, fisiologiche o patologiche, all'interno della cellula.

2 Clustering nell'analisi esplorativa dei dati

2.1 *Organizzazione dei dati sperimentali*

Per fissare le idee, supponiamo che l'obiettivo scientifico sia valutare il livello di espressione genica in due condizioni sperimentali. Queste condizioni sperimentali potrebbero essere per esempio due categorie di tessuto (sano e tumorale), oppure due trattamenti (farmaco e controllo), oppure due prognosi differenti, e così via.

Supponiamo inoltre di avere effettuato N esperimenti. Questi esperimenti potrebbero essere prelievi di tessuto di due differenti nature, corrispondenti alle due condizioni sperimentali di cui sopra. Per esempio, supponiamo di avere un gruppo di pazienti affetti da tumore e di prelevare da ciascuno di essi un campione di tessuto sano e uno di tessuto tumorale.

Ciascuno dei nostri esperimenti consiste nell'osservazione del livello di espressione di G geni, e il risultato che abbiamo ottenuto è una serie di misure x sulla differenza tra i livelli di espressione (o espressione differenziale) nelle due condizioni sperimentali.

Un modo semplice e pratico per organizzare questi nostri dati è quello tabulare. Creiamo una tabella o *matrice* a G colonne, ognuna corrispondente a un gene. Avremo quindi N righe, ognuna corrispondente a un esperimento (per esempio a un distinto microarray). L'organizzazione dei dati è quindi la seguente:

$$X = \begin{bmatrix} (\text{esperimento } 1) & x_{11} & x_{12} & \cdots & x_{1G} \\ (\text{esperimento } 1) & x_{21} & x_{22} & & x_{2G} \\ \vdots & & & \ddots & \vdots \\ (\text{esperimento } N) & x_{N1} & x_{N2} & \cdots & x_{NG} \end{bmatrix}$$

È possibile organizzare i dati anche secondo la disposizione trasposta, a N colonne e G righe (naturalmente avremo $x_{ij} = y_{ji}$ in quanto $X = Y^{\mathrm{T}}$):

.

$$Y = \begin{bmatrix} (\text{gene } 1) & y_{11} & y_{12} & \cdots & y_{1N} \\ (\text{gene } 2) & y_{21} & y_{22} & & y_{2N} \\ & & & \ddots & \vdots \\ (\text{gene G}) & y_{G1} & y_{G2} & \cdots & y_{GN} \end{bmatrix}$$

La nostra tabella dati verrà quindi registrata in un file. Esistono diversi formati di file possibili, dalle tipologie più generali a formati specializzati. I file di "testo semplice", "plain text", sono creati utilizzando solo caratteri (alfanumerici e simbolici) appartenenti a un ridotto alfabeto, per esempio la tabella standard ASCII. Si può dire che qualunque programma può leggere un file di testo semplice, purché i dati contenuti siano disposti in qualche modo noto o ragionevolmente comprensibile. Un altro formato molto diffuso è il file di foglio elettronico: esso è sufficientemente flessibile da poter contenere tabelle di varia natura e dimensione (entro determinati limiti), e tuttavia è ancora un formato generico, utilizzabile per scambi tra programmi diversi. Esistono poi le basi di dati, adeguate per tabelle di dimensioni molto elevate, più adatte alla memorizzazione e alla consultazione che non ad un'analisi computazionalmente intensiva, e formati cosiddetti "proprietari", cioè caratteristici di uno specifico software e interscambiabili soltanto fra programmi appositamente progettati.

2.2 *Analisi esplorativa e clustering*

Si parla di analisi esplorativa quando lo scopo dell'indagine risiede nel tentativo di estrarre dai dati un contenuto informativo per grandi linee, sulla base del quale verranno successivamente impostate le successive fasi dell'analisi. Normalmente (ma non necessariamente), la fase esplorativa dell'analisi è del tipo non supervisionato, ovvero non fa uso delle eventuali

informazioni addizionali sul fenomeno in esame, quali possono essere diagnosi già effettuate per altre vie, categorie già note a cui i tessuti appartengono, e così via.

Il *clustering* o *cluster analysis* è una forma molto popolare di analisi esplorativa. Consiste nell'identificare, all'interno dell'insieme dei dati sperimentali, possibili suddivisioni che possono essere interpretate come "categorie naturali". Le categorie possono essere eventualmente formate di sotto-categorie. Un esempio classico di clustering è il lavoro svolto da Linneo quando, sulla base di caratteristiche di vario tipo, stabilì le categorie in cui si suddividono e raggruppano naturalmente gli esseri viventi, ossia la loro classificazione tassonomica.

Il clustering è l'analisi che qui ci interessa descrivere; è estremamente diffuso nell'analisi dei dati da microarray, ma anche in molti altri contesti; non è tuttavia l'unica procedura possibile. Possiamo ricordare per esempio vari tipi di *selezione delle variabili*, in cui lo scopo consiste nel selezionare le variabili sperimentali significative eliminando quelle con il minore contenuto informativo; l'*analisi delle componenti principali*, per individuare gruppi di variabili correlate il cui contributo collettivo sia più significativo di quello delle variabili individuali; l'*analisi della distribuzione*, per valutare ipotesi su quale sia la distribuzione di probabilità dei dati o almeno su alcune sue caratteristiche.

In generale, possiamo dire che lo scopo dell'analisi esplorativa consiste nel generare ipotesi di lavoro sulle quali basare le successive fasi della ricerca, per esempio per progettare ulteriori esperimenti di laboratorio a scopo di conferma.

Tornando al clustering, esso viene impiegato nel campo dei microarray con l'obiettivo di identificare raggruppamenti nell'insieme dei dati. Questi raggruppamenti saranno poi interpretati come casi tipici, e potremo quindi cercare di approfondire con le successive fasi dello studio se tali casi tipici hanno effettivamente un significato biologico.

Per chiarire l'aspetto dell'interpretazione dei dati sperimentali, ritorniamo per un momento alla rappresentazione tabulare dei dati che abbiamo introdotto in precedenza. Possiamo leggere le due possibili tabelle X e Y come corrispondenti a due tipi differenti di analisi.

La tabella X ha le righe corrispondenti ciascuna a un singolo esperimento. Ogni riga è composta dei valori di espressione (nel nostro esempio di riferimento, l'espressione differenziale tra due condizioni) di ciascun gene. Essa rappresenta quindi il profilo di espressione di tutti i geni in esame corrispondente a un determinato esperimento.

In questo caso, righe simili rappresentano esperimenti in cui si è verificata un'analoga distribuzione dei livelli di espressione. Per esempio, se nel mio insieme di dati sperimentali sono rappresentati casi di tumori macroscopica-

mente omogenei, ma legati a meccanismi patogenetici differenti a livello molecolare, ottengo gruppi diversi che possono corrispondere ciascuno a un differente meccanismo molecolare di sviluppo del tumore.

La tabella *Y* ha le righe corrispondenti ciascuna a un singolo gene. Ogni riga è composta dei valori di espressione (differenziale) di un dato gene in ciascun esperimento. Essa rappresenta quindi il comportamento dell'espressione di quel gene al variare degli esperimenti.

In questo secondo caso, quindi, righe simili corrispondono a geni che sono tutti sovraespressi o sottoespressi negli stessi esperimenti, mentre righe differenti rappresentano geni che, nello stesso esperimento, sono espressi in modo diverso. Il clustering rivelerà quindi gruppi di geni che sono espressi in modo simile in tutti gli esperimenti, e che perciò possono essere parte del medesimo meccanismo di regolazione.

Notiamo a questo punto che, nell'esposizione, abbiamo fin qui sempre affermato che i risultati *possono* avere una data interpretazione. Ribadiamo quindi che stiamo descrivendo una tecnica di analisi esplorativa, il cui fine è di produrre delle ipotesi che il lavoro successivo dovrà confermare. Questa osservazione deve anche spingere il ricercatore a valutare e bilanciare le varie fasi del lavoro (e una appropriata conoscenza delle tecniche di clustering è essenziale per evitare di investire troppo tempo nella fase esplorativa).

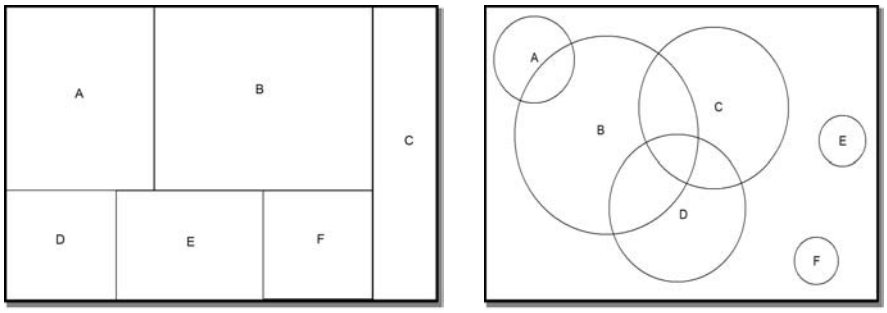
2.3 *Caratteristiche delle tecniche di clustering*

Metodi per il clustering sono stati sviluppati in ambiti diversi e partendo da ipotesi differenti. Il concetto stesso di cosa sia un cluster è soggetto a interpretazioni personali, dipendenti in modo oggettivo dall'impostazione del problema, ma anche dalle assunzioni fatte dal ricercatore, quindi con una componente soggettiva. Esiste quindi una varietà di tecniche e di algoritmi, che possono essere categorizzati in vari modi per rendere più sistematica la discussione.

Una prima categorizzazione possibile è quella fra tecniche *partitive* e tecniche *gerarchiche*. Il clustering partitivo parte dall'ipotesi che i dati possono essere organizzati in una *partizione*, ossia che l'insieme dei dati può essere suddiviso in più sottoinsiemi, in modo tale che:

1) l'unione di tutti i sottoinsiemi dà l'insieme totale dei dati;
2) l'intersezione di ogni coppia di sottoinsiemi è vuota.

Il concetto matematico di partizione è illustrato in figura 3. In (a), i sottoinsiemi A, B, C, D, E, F formano una partizione del rettangolo esterno. In (b), i sottoinsiemi A, B, C, D, E, F non formano una partizione perché non



(a) **(b)**

Fig. 3 – I sottoinsiemi A, B, C, D, E possono costituire (a) o non costituire (b) una partizione.

comprendono tutto il rettangolo esterno (condizione 1), e perché A, B, C, D sono in parte sovrapposti (condizione 2).

Il clustering gerarchico attribuisce ai dati una strutturazione per certi versi simile, ma basata sul concetto di "livelli di dettaglio". Una prima divisione grossolana viene ulteriormente sottoposta a divisioni successive, ripetutamente, ottenendo così una *gerarchia* di divisioni. Una simile gerarchia è illustrata in figura 4.

Un'altra categorizzazione possibile riguarda le proprietà dell'algoritmo, in particolare la complessità computazionale (la legge che lega le caratteristiche del problema al tempo necessario a risolverlo) e l'occupazione di memoria. In questo senso si possono identificare due grandi categorie: nella prima i cluster vengono rappresentati attraverso punti rappresentativi o prototipi, che sono calcolati iterativamente, modificandoli un passo alla volta fino al raggiungimento di una soluzione accettabile. In questo tipo di algoritmo viene memorizzata la tabella dati, che viene scorsa più volte, mentre ad ogni iterazione si ricalcolano tutte le distanze tra ogni prototipo e i dati.

Questo tipo di algoritmo viene detto *stored data* e la sua complessità è dominata dal tempo necessario a ricalcolare le distanze, che per k prototipi e n osservazioni di v variabili, è grosso modo proporzionale a knv per ogni iterazione, mentre la memoria necessaria è proporzionale a $(k + n)v$.

Nella seconda categoria, i cluster vengono semplicemente rappresentati come insiemi di dati, ed è quindi sufficiente calcolare una volta tutte le distanze e poi memorizzare la matrice delle distanze. Questo tipo di algoritmo, detto *stored matrix*, ha un tempo di esecuzione che dipende dalla tecnica usata, ma in genere è dominato dal tempo di calcolo della matrice, che è grosso modo

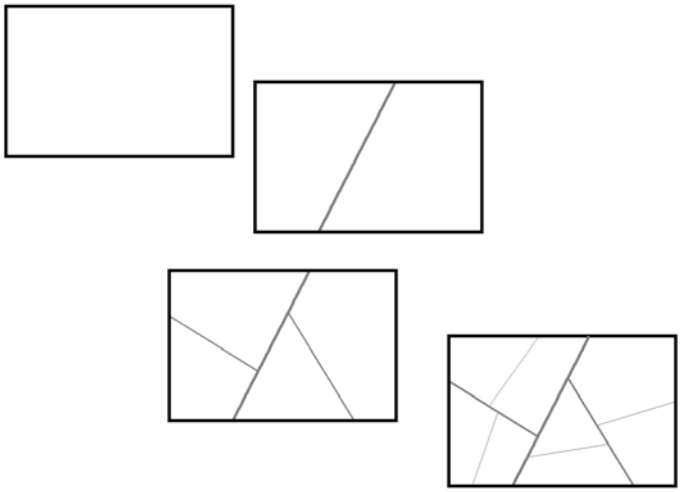


Fig. 4 – Una sequenza che illustra una gerarchia di sottoinsiemi.

proporzionale a $(nv)^2$; per quanto riguarda la memoria necessaria, questa è proporzionale a n^2.

A seconda dei problemi, quindi, potrà convenire utilizzare algoritmi dell'una o dell'altra categoria. Possiamo dire che un tipico problema di analisi di espressione genica può avere $N = 10 \div 100$ e $G = 1000 \div 10000$. Quindi, considerando il tempo di calcolo, potremo preferire algoritmi del primo tipo per l'analisi di Y e del secondo tipo per l'analisi di X. Occorre però fare una considerazione. Le risorse di calcolo tipicamente disponibili ormai anche su calcolatori desktop tendono a rendere irrilevanti queste valutazioni in molti casi. La scelta dell'algoritmo potrà quindi spesso essere basata su altre considerazioni.

2.4 *Distanze e similarità*

Abbiamo finora citato il concetto di "distanza fra osservazioni" senza ulteriori dettagli. Una distanza è una funzione che, date due righe di una matrice dati, dà come risultato un valore che è una valutazione di dissimilarità fra le due righe. In generale, una distanza deve possedere alcune proprietà: non

assumere valori negativi, risultare nulla se applicata a righe uguali, rispettare la disuguaglianza triangolare. In alcuni casi, è più naturale ragionare in termini di somiglianza o similarità. Una similarità è una funzione che ha andamento opposto alla distanza, nel senso che è massima quando la distanza risulta minima, e si può quindi calcolare con formule del tipo 1 – *dist* oppure 1/*dist*.

Supponendo di disporre di dati metrici (ogni osservazione $\mathbf{x}_i$ o $\mathbf{y}_j$ è un vettore di valori reali), esistono varie scelte possibili che passiamo in rapida rassegna nel seguito.

La più naturale definizione di distanza fra due vettori **v** e **w** (di dimensione *m*) è la distanza euclidea,

$$dist = \sqrt{\sum_{i=1}^{m} (v_i - w_i)^2} \ .$$

Questa definizione si può generalizzare secondo la definizione di Minkowski

$$dist = \sqrt[h]{\sum_{i=1}^{m} |v_i - w_i|^h}$$

quando *h* vale 2 si riottiene la definizione euclidea, mentre per $h = 1$ si ha un'altra comune definizione di distanza:

$$dist = \sum_{i=1}^{m} |v_i - w_i| \ \ ,$$

la cosiddetta *distanza di Manhattan* o *cityblock distance*.

Il concetto statistico di correlazione suggerisce un modo per valutare la similarità tra due dati. Consideriamo le componenti dei due generici vettori **v** e **w** come una sequenza di coppie di variabili. Possiamo vedere i vettori come un campione statistico di *m* coppie di osservazioni $\{ (v_1,w_1), (v_2,w_2), ..., (v_m,w_m) \}$. La sequenza v_i e la sequenza w_i sono tanto più simili quanto più, statisticamente, componenti omologhe tendono ad avere valori vicini. In statistica, questa valutazione è realizzato attraverso il concetto di correlazione. Il *coefficiente di correlazione di Pearson* è un numero reale che misura quanto le variazioni di un campione sono associate a simili variazioni di un altro campione:

$$r = \frac{\sum_{i=1}^{m} (v_i - E\{\mathbf{v}\})(w_i - E\{\mathbf{w}\})}{\sum_{i=1}^{m} (v_i - E\{\mathbf{v}\})^2 \sum_{i=1}^{m} (w_i - E\{\mathbf{w}\})^2}$$

dove E{**x**} è la media delle componenti del generico vettore **x**.

Tale numero può assumere valori nell'intervallo [–1,+1], e precisamente:

- $r = 1$ se per ogni i è $v_i = w_i$
- $r = 0$ se non esiste alcuna legge che lega i valori di v_i e w_i;
- $r = -1$ se per ogni i è $v_i = -w_i$

La correlazione misura una relazione probabilistica lineare, cioè la probabilità che i due campioni siano legati da una legge di proporzionalità.

Se vogliamo valutare un tipo più generale di legge, possiamo ricorrere alla correlazioni fra i ranghi. Tale concetto è espresso dalla *correlazione fra ranghi di Spearman*, che è un coefficiente di correlazione calcolato non sui valori dei campioni, ma sui loro *ranghi*, le posizioni occupate in una lista ordinata. In tal modo si possono valutare relazioni arbitrarie, purché di tipo monotòno.

Il concetto di correlazione è molto utile nell'analisi di dati da microarray, in quanto consente di confrontare valori tendenziali anziché assoluti (la normalizzazione dei dati da microarray non è in grado di contrastare tutte le possibili fonti di variabilità fra esperimenti diversi).

Si può infine calcolare la differenza angolare o *distanza coseno* fra due vettori. Essa è definita in funzione del coseno dell'angolo fra due vettori, e calcolata come segue:

$$\cos\theta_{\mathbf{v},\mathbf{w}} = \frac{\sum_{i=1}^{m} v_i w_i}{\sqrt{\sum_{i=1}^{m} v_i^2 \sum_{i=1}^{m} w_i^2}}$$

Il coseno dell'angolo è una espressione molto ragionevole della similarità fra due vettori quando questi hanno modulo (lunghezza) simile.

Confrontando questa espressione con le precedenti, vediamo che essa coincide con la correlazione tra i due vettori se questi hanno media delle

componenti nulla; inoltre per vettori di modulo unitario la distanza coseno e la distanza euclidea sono legati da una relazione quadratica

$$\left(dist_{\text{euclidea}}\right)^2 = 1 - \cos\theta_{\text{v,w}}.$$

Vettori di modulo unitario si ottengono quando, per eliminare variazioni di misura da un esperimento all'altro, ogni esperimento viene normalizzato individualmente. Nel caso dei microarray, questa operazione mantiene l'informazione sul livello relativo di espressione di ciascun gene, eliminando il livello assoluto che, essendo soggetto a forti variazioni, non è confrontabile fra un esperimento e l'altro.

3 Tecniche di clustering partitivo

3.1 *c-Means*

Una delle tecniche di clustering più popolari è quella delle *c* medie o *c-Means* [2]. Dato un numero prestabilito *c* di cluster a cui attribuire i dati, si può rappresentare ciascun cluster attraverso un punto di riferimento o prototipo **y**. Tale punto di riferimento può essere scelto in molti modi. Tuttavia, una scelta spesso ragionevole è calcolare ogni prototipo come il baricentro di un cluster, ossia la posizione media dei punti nel cluster, detta il *centroide*.

Con tale rappresentazione, si ottiene un modello di cluster che è definito in questo modo: il punto più rappresentativo del cluster *j*-esimo è il suo baricentro $\mathbf{y}_j$ (che è ottenuto da un calcolo, e in generale non fa parte dei dati realmente misurati); i dati appartenenti al cluster *j*-esimo sono tutti quei dati $\mathbf{x}_k$ che sono più prossimi al prototipo $\mathbf{y}_j$ che a qualunque altro prototipo $\mathbf{y}_l$ ($l \neq j$), dove ovviamente la prossimità è valutata secondo la definizione di distanza prescelta.

L'algoritmo *c*-Means si può schematizzare come segue:

0. Inizializzare i *c* prototipi $\mathbf{y}_1 \ldots \mathbf{y}_c$

1. Attribuire ogni punto $\mathbf{x}_k$ al cluster che ha prototipo più vicino:
   $u_{jk} = 1$ per j = indice del prototipo più vicino
   $u_{jk} = 0$ per ogni altro j

2. Ricalcolare la posizione dei prototipi come media dei punti nel cluster:

$$\mathbf{y}_j = \frac{\sum_i u_{jk} \mathbf{x}_k}{u_{jk}}$$

3. Iterare da 1 fino al soddisfacimento di un criterio di arresto (convergenza)

Alcune osservazioni riguardo a questo algoritmo. L'inizializzazione può essere effettuata in molti modi; due tecniche comuni sono prendere *c* punti a caso nell'insieme dati, che assicura che i centroidi si trovino all'interno della distribuzione dei dati, oppure prendere *c* punti completamente casuali, che consente un numero molto maggiore di possibili inizializzazioni differenti (questo risulta utile nel caso che l'algoritmo sia applicato multiple volte alla ricerca di un ottimo globale).

La variabile u_{jk} è una *variabile indicatrice*, che vale 1 quando il punto $\mathbf{x}_k$ appartiene al cluster $\mathbf{y}_j$, 0 altrimenti. Essa si definisce *appartenenza* del punto *k* al cluster *j*. Una variabile indicatrice si può interpretare come un valore logico di verità: l'evento "$\mathbf{x}_k$ appartiene al cluster *j*-esimo" è vero se e solo se u_{jk} vale 1.

Le appartenenze devono soddisfare determinati vincoli. Un vincolo importante è quello della normalità della somma (sum-normality): per il punto *k*, deve essere

$$\sum_j u_{jk} = 1,$$

ossia la somma delle appartenenze ai cluster deve essere 1 per ogni punto (ogni punto appartiene a esattamente 1 cluster).

Per via della tecnica di ottimizzazione adottata, non è detto che l'algoritmo arrivi a una soluzione valida: ci sono molti minimi locali, cioè soluzioni ammissibili; e non c'è un modo diretto di sapere se, una volta raggiunta una soluzione, essa sia la migliore possibile o no.

Dal punto di vista della rappresentazione dei cluster, potrebbe essere desiderabile mantenere un'informazione su quanto un punto appartiene a un cluster. I punti più lontani hanno "meno diritto" di quelli più vicini di essere assegnati a un cluster. Se esistesse un modo per mantenere questa informazione, sarebbe possibile valutare la qualità dei cluster durante l'applicazione dell'algoritmo, e avere quindi qualche indicazione più precisa sulla bontà della soluzione verso la quale esso tende. In altre parole, l'algoritmo potrebbe essere meno sensibile ai minimi locali. Inoltre, questa informazione potrebbe essere di aiuto nell'uso del clustering ottenuto, per poter distinguere tra punti più e meno tipici.

L'algoritmo che segue risolve proprio questo problema.

3.2 *Clustering fuzzy*

Si parla di *logica fuzzy* [33] quando, oltre ai valori di verità estremi *vero* e *falso*, si ammette anche un continuum di valori intermedi tra il vero e il falso. Nel caso fuzzy, possiamo assumere che la grandezza u_{ij} non sia più una variabile intera ($u_{ij} \in \{0,1\}$) ma una quantità reale. I valori che essa assume si possono quindi far corrispondere al continuum di valori di verità della logica fuzzy.

A partire da questa impostazione, è possibile sviluppare differenti algoritmi di clustering, distinti ma accomunati dall'uso di variabili reali come indicatori *fuzzy* di appartenenza.

Il principale membro della famiglia del clustering fuzzy si chiama Fuzzy *c*-Means [5, 11]. L'algoritmo Fuzzy *c*-Means è analogo a *c*-Means tranne per il modo di calcolare i valori di appartenenza:

$$u_{jk} = \sum_l \left(\frac{dist_{jk}}{dist_{jl}} \right)^{\frac{2}{\mu-1}}$$

dove μ è un parametro che decide quanto fuzzy deve essere il clustering che cerchiamo. L'algoritmo si può quindi scrivere come segue:

0. Inizializzare i c prototipi $\mathbf{y}_1 \ldots \mathbf{y}_c$

1. Attribuire ogni punto $\mathbf{x}_i$ ai cluster in funzione delle distanze dai rispettivi centroidi:

$$u_{jk} = \sum_l \left(\frac{dist_{jk}}{dist_{jl}} \right)^{\frac{2}{\mu-1}}$$

2. Ricalcolare la posizione dei prototipi come media dei punti nel cluster:

$$\mathbf{y}_j = \frac{\sum_i u_{jk} \mathbf{x}_k}{u_{jk}}$$

3. Iterare da 1 fino al soddisfacimento di un criterio di arresto (convergenza)

Si può vedere come in questo caso siano possibili valori di u_{jk} compresi in tutto l'intervallo reale [0, 1], non più solo nell'insieme discreto {0, 1}. Il vincolo di normalità della somma resta valido. Con questa impostazione, la variabile di appartenenza va interpretata in un modo differente: ora non è più possibile distinguere tra appartenenza e non-appartenenza, in quanto u_{jk} qui valuta il *grado* di appartenenza ai cluster. Di conseguenza si può dire che ogni punto appartiene ad ogni cluster, ma con gradi di appartenenza differenti (dipendenti dalle relative distanze punto-centroide). L'appartenenza ai cluster più prossimi sarà dunque elevata, ma si potrà avere un grado residuale di appartenenza anche nei confronti di cluster molto lontani.

Il permanere della condizione di normalità implica in questo caso che le appartenenze vanno lette come frazioni, o percentuali, di appartenenza, per cui per esempio $u_{jk} = 0.9$ indica che il punto k appartiene per il 90% al cluster j, e per il restante 10% agli altri cluster. Notiamo che in questo caso le appartenenze possono anche essere interpretate come probabilità, anch'esse soggette per definizione al vincolo di normalità. Potremo quindi definire questa condizione anche "vincolo probabilistico".

L'ottimizzazione avviene con variabili tutte reali. La convergenza è più lenta, ma i minimi locali sono meno probabili.

Si ha poi un vantaggio nell'interpretazione: poiché i punti sono attribuiti a tutti i cluster con gradi diversi di appartenenza, è possibile individuare casi limite ed eventualmente prendere decisioni di conseguenza. Questo è il caso dei punti atipici, che non rappresentano alcun cluster (gli *outlier*). Sempre sul piano interpretativo, tuttavia, esiste anche uno svantaggio: per via della sum-normality, punti lontani da tutti i cluster possono venire attribuiti "fortemente" al cluster più vicino, anche se (in valore assoluto) esso è comunque distante.

Una illustrazione di questo possibile problema è in figura 5: qui sono rappresentati due cluster in uno spazio dati bidimensionale; i centri, posti alle coordinate (0.3, 0.3) e (0.7, 0.7), sono evidenziati con quadrati neri; i dati con puntini. Quattro punti esemplificativi sono indicati con numeri. Il punto in posizione 1 coincide con il centro del cluster 2 (anche se ciò è difficile da illustrare con precisione), ed è chiaramente appartenente al cluster 2; quindi $u_{11} = 0$, $u_{21} = 1$. Il punto in posizione 2 è a metà strada fra i due cluster; esso appartiene al 50% a ciascuno dei due cluster ($u_{12} = 0.5$, $u_{22} = 0.5$), e questo risponde all'intuizione. Il punto 3 è anch'esso equidistante dai due cluster, e anche per esso vale $u_{13} = 0.5$, $u_{23} = 0.5$. Tuttavia, è evidente dalla figura che i due punti 2 e 3 non sono "equivalenti": ci piacerebbe che il valore di appartenenza fosse differenziato, a mostrare che il punto 2 è più "tipico" dei due cluster di quanto non lo sia il punto 3. Quanto al punto 4, anch'esso evidenzia un'anomalia: poiché esso ha appartenenza molto bassa al cluster 1, la

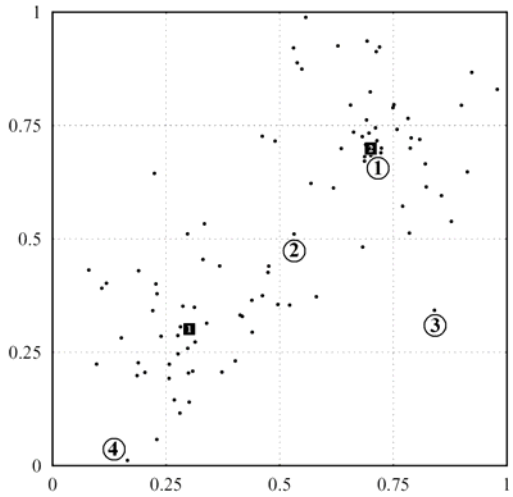


Fig. 5 – Esempi di posizioni dei dati rispetto ai cluster.

sua appartenenza al cluster 1 risulta elevata ($u_{14} = 1$, $u_{24} = 0$). Esso quindi risulta "tipico" del cluster 1 tanto quanto il punto 1 lo è del cluster 2: risultato singolare, visto che il punto 4 è particolarmente lontano dal prototipo 1, mentre il punto 1 è addirittura coincidente con il prototipo 2.

Ci piacerebbe dunque poter ottenere non un grado di appartenenza, espresso in termini relativi (percentuale), ma un grado di *tipicità*, espresso in assoluto, che dia informazione su quanto ogni punto sia rappresentativo di ciascun cluster.

3.3 *L'approccio possibilistico*

L'algoritmo detto *Possibilistic c-Means* [18, 19] punta ad alleviare il problema che abbiamo ora evidenziato. Esso è basato su un principio molto semplice: eliminare il vincolo di normalità.

Naturalmente, prendere un algoritmo (come per esempio Fuzzy *c*-Means) ed eliminare semplicemente il vincolo dà luogo a un algoritmo che genera soluzioni banali: infatti la soluzione con $u_{jk} = 0$ per ogni j e per ogni k è sempre ammissibile. L'algoritmo di clustering possibilistico si realizza imponendo alcuni vincoli elementari (che garantiscano almeno un appartenenza non nulla per ogni punto e almeno un punto per ogni cluster) e soprattutto sostituendo il

vincolo di normalità con un termine di penalizzazione, che allontana dalle soluzioni banali.

Si ottiene una classe di algoritmi che si differenziano per il termine di penalizzazione scelto. Un esempio, che riportiamo nel seguito, è la seconda versione presentata dagli autori.

0. Inizializzare i c prototipi $\mathbf{y}_1 \ldots \mathbf{y}_c$

1. Attribuire ogni punto $\mathbf{x}_i$ ai cluster in funzione delle distanze dai rispettivi centroidi:

$$u_{jk} = e^{dist_{jk} / \beta}$$

2. Ricalcolare la posizione dei prototipi come media dei punti nel cluster:

$$\mathbf{y}_j = \frac{\sum_i u_{jk} \mathbf{x}_k}{u_{jk}}$$

3. Iterare da 1 fino al soddisfacimento di un criterio di arresto (convergenza)

Il calcolo dell'appartenenza, al punto 1, è interessante in quanto formalmente si può collegare ad altre tecniche di clustering, sempre basate sull'applicazione di termini di penalità o regolarizzazione [24], quali quella presentata in [29]. In esso entra un parametro β che regola l'ampiezza dei cluster, o in altri termini la risoluzione dell'algoritmo.

In assenza di vincolo di normalità o "probabilistico", si parla di caso "possibilistico", in quanto i valori delle appartenenze possono essere interpretati non più come valori di probabilità, ma come gradi di possibilità.

Anche questo algoritmo, pur soddisfacendo l'obiettivo di lavorare su gradi di tipicità anziché di appartenenza, presenta inconvenienti. Il primo è stato evidenziato in [3], e consiste nel fatto seguente: pur avendo penalizzato (e quindi ragionevolmente eliminato) le soluzioni banali, sono sempre possibili soluzioni a cluster coincidenti. Questo perché nei modelli in cui è presente il vincolo di normalità si ha una interazione fra i centri cluster, che tende ad evitare che essi si sovrappongono; nel caso possibilistico il vincolo è rimosso, l'interazione non è presente, e non si ha alcun meccanismo che tenda ad evitare le soluzioni con centroidi coincidenti.

Un altro aspetto, affrontato in alcuni lavori come [26], riguarda di nuovo la rappresentazione: piuttosto che operare esclusivamente sulle tipicità, sarebbe più utile poter avere informazioni sia di tipo assoluto (tipicità) che di tipo relativo (appartenenza).

Descriviamo ora una tecnica che mira a rispondere a queste esigenze.

3.4 *Clustering parzialmente possibilistico*

La tecnica del clustering *parzialmente possibilistico* [22, 23] si basa su una transizione continua (soft) tra il modello fuzzy convenzionale o "probabilistico" e quello possibilistico.

Anziché rimuovere il vincolo di normalità della somma, lo sostituiamo con un vincolo simile, ma espresso in forma intervallare:

$$\sum_j u_{jk}^{[\xi]} = 1$$

dove $[\xi]$ è un parametro di nostra scelta che non è un valore reale ma un *intervallo*. La matematica degli intervalli è uno strumento nato per il controllo degli errori nel calcolo numerico; essa ha però trovato molte applicazioni nell'ambito del soft computing [16, 9, 28].

Su questo parametro richiediamo solo che gli estremi siano non negativi. A seconda dell'intervallo scelto, otteniamo i casi seguenti:

- Quando l'ampiezza di $[\xi]$ è nulla (ossia esso è un intervallo degenere, un singolo numero reale), il vincolo coincide con quello di normalità della somma.

- Quando l'ampiezza di $[\xi]$ è infinita, il vincolo è sempre soddisfatto e tutto va come se non vi fosse alcun vincolo.

- Quando l'ampiezza di $[\xi]$ è finita, ossia gli estremi sono due valori reali, il vincolo obbliga le appartenenze ad assumere valori in un determinato insieme che è "più esteso" che nel caso probabilistico, ma "meno esteso" che nel caso possibilistico.

Il terzo caso costituisce una situazione intermedia tra i due casi estremi, e rappresenta il contributo principale di questa tecnica.

Una interpretazione geometrica può aiutare a comprendere meglio i casi possibili. Seguendo le convenzioni del calcolo degli intervalli, l'equazione che

esprime il vincolo si può interpretare come una coppia di disequazioni. Le variabili $u_{1k},\ldots,u_{ck}$ sono allora le coordinate di un punto in uno spazio vettoriale c-dimensionale. Le disequazioni sono soddisfatte da punti che appartengono a zone di tale spazio di forma ben definita: nel caso degenere, in cui la coppia di disequazioni si riduce a una equazione, si tratta di un segmento di iperpiano (nel caso bidimensionale $c = 2$, un segmento di retta); nel caso in cui il vincolo è sempre soddisfatto, poiché le appartenenze sono non negative e non superiori a 1, si tratta dell'ipercubo a lato unitario (nel caso bidimensionale, un quadrato di lato 1); nei casi intermedi, si tratta di una zona che possiamo visualizzare nel caso bidimensionale, e che in figura 5 corrisponde alle zone a forma di occhio.

Possiamo quindi osservare che il vincolo proposto realizza, a seconda del parametro impostato, sia il vincolo di normalità del caso convenzionale che l'assenza di vincolo del caso possibilistico; esso aggiunge inoltre una zona di transizione continua tra i due modelli.

Questa tecnica si può implementare in algoritmi differenti, e consente di ottenere una varietà di proprietà intermedie tra i due modelli estremi, e inoltre di realizzare strumenti non disponibili nei casi estremi. Una tecnica pratica per realizzare il parametro ad intervalli consiste nello stabilire un parametro reale $\alpha \in [0,1]$ e porre

$$[\xi] = [\, \alpha, 1/\alpha \,].$$

In tal modo è necessario impostare solo un valore reale.

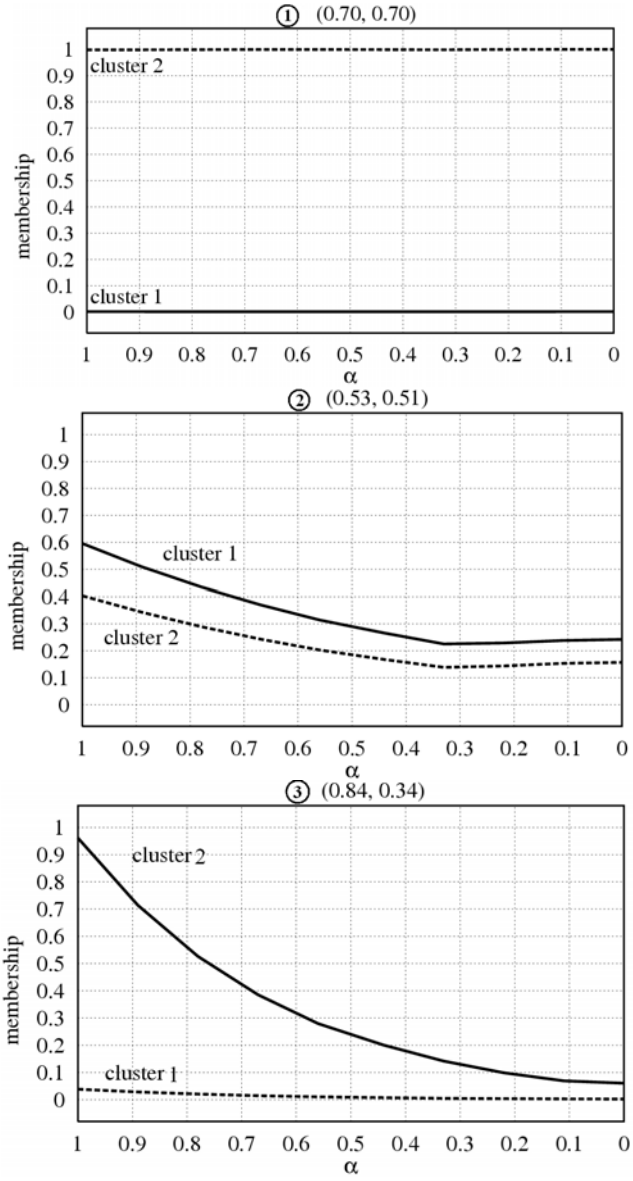


Fig. 7 – Appartenenza ai due cluster per tre punti nel problema di esempio, al variare del parametro α.

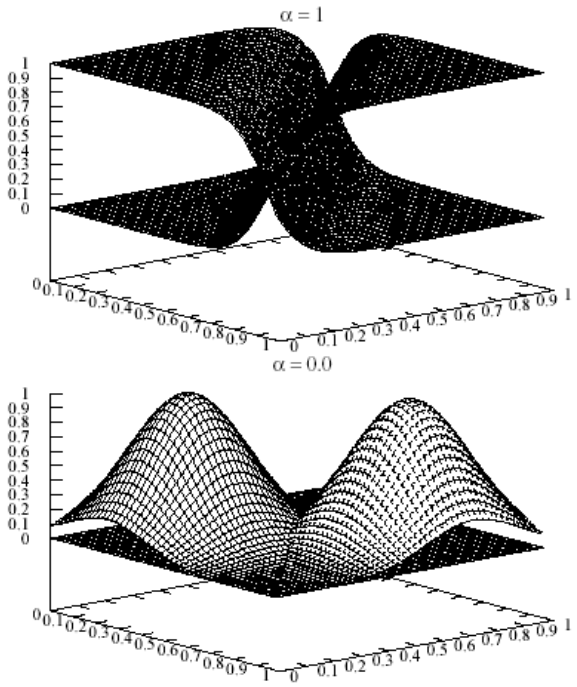


Fig. 8 – Visualizzazione tridimensionale dei valori di appartenenza a due cluster, nei casi convenzionale (α = 1.0, sopra) e possibilistico (α = 0.0, sotto)

Il problema dei cluster coincidenti, tipico del modello possibilistico, può essere alleviato realizzando un algoritmo aggiornato attraverso passi parziali (per esempio un algoritmo "on-line", o di ottimizzazione stocastica). In tal modo è possibile iniziare con un modello fuzzy convenzionale (con [ξ] intervallo degenere) e gradualmente spostarsi verso il modello possibilistico. Si ottiene una inizializzazione del metodo possibilistico che, pur alleviando la tendenza a soluzioni coincidenti, non la elimina del tutto; resta sempre la possibilità di sfruttare tale proprietà nel modo evidenziato in [19], per esempio per ottenere una regolazione automatica del numero di cluster (i cluster coincidenti vengono fatti collassare in uno).

Possiamo vedere graficamente, sempre con riferimento all'esempio di figura 3, come variano i valori di appartenenza ai cluster al variare del parametro α.

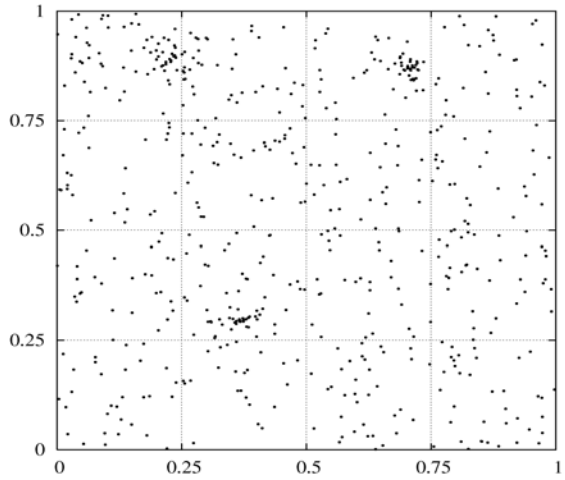


Fig. 9 – Di questi dati, il 90% sono outlier.

Eseguito il clustering, valutiamo il valore dell'appartenenza ai cluster 1 e 2 per tre punti interessanti. (Ricordiamo che per $\alpha = 1$ siamo nel caso convenzionale o "probabilistico", e il caso possibilistico si ha per $\alpha = 0$; per altri valori siamo in casi intermedi.)

Nel caso del punto 1, l'appartenenza al cluster non varia al variare del modello. Come ci si aspetta, un punto coincidente con il centro cluster ha sempre appartenenza completa a tale cluster e appartenenza nulla all'altro.

Per il punto 2, nel caso probabilistico le due appartenenze sono vicine al 50%. Nella transizione verso il modello possibilistico ($\alpha \to 0$), il *rapporto* tra le due appartenenze tende a restare pressoché costante, ma il *valore* decresce visibilmente per entrambe.

Il punto 3 è lontano da entrambi i cluster, ma si trova più vicino al cluster 2 che al cluster 1. Nel caso probabilistico, si ha quindi il paradosso che la sua appartenenza al cluster 2 è molto elevata: più elevata, per esempio, di quella del punto 2, che pure è più prossimo al centroide, ma in una posizione in cui "si sente" anche l'influenza del cluster 1. La transizione verso il modello possibilistico riporta l'appartenenza a livelli più ragionevoli: per $\alpha \to 0$ è chiaro che si tratta di un punto atipico (outlier) per entrambi i cluster, con un livello di appartenenza basso sia verso il cluster 1 che verso il cluster 2. Siamo quindi in grado, con questa tecnica, di identificare gli outlier.

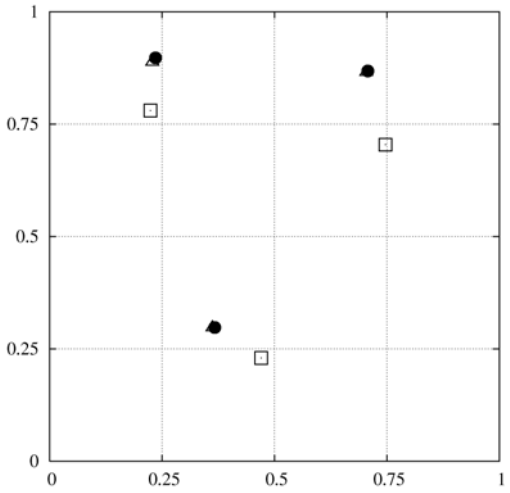


Fig. 10 – Centri cluster ottenuti sul problema di Fig. 9.
● centri veri; □ centri Fuzzy c-Means; Δ centri Parzialmente Possibilistico

Il profilo delle appartenenze, per tutti i punti del piano nell'esempio, è illustrato in figura 8 in due grafici tridimensionali (sul piano orizzontale le coordinate dei punti dati, sull'asse *z* i valori delle appartenenze ai due cluster). Questo diagramma è presentato sia per il caso probabilistico (grafico in alto) che per il caso possibilistico (grafico in basso).

Possiamo così apprezzare l'andamento generale delle funzioni di appartenenza nei due casi. Si può vedere come, nel caso possibilistico, le appartenenze tendano ad abbassarsi all'allontanarsi dai due centri cluster, che, ricordiamo, sono alle coordinate (0.3, 0.3) e (0.7, 0.7). Ciò non succede nel caso probabilistico. Come abbiamo detto, questo può essere un aspetto utile oppure no, e la possibilità di transizione continua da un modello all'altro consente di effettuare un'analisi più espressiva.

Un'interessante applicazione della tecnica parzialmente possibilistica si ottiene quando il parametro intervallare è asimmetrico, per esempio della forma $[\alpha, 1]$. In tal caso, se un punto ha appartenenza elevata a due o più cluster, questi entrano in competizione e si realizza il modello probabilistico; al contrario, quando le appartenenze sono basse, il modello risulta possibilistico perché l'interazione viene a mancare.

L'effetto di questo accorgimento è quello di rendere l'algoritmo di clustering molto più insensibile agli outlier. Le figure seguenti illustrano un insieme di dati in cui il 10% dei punti è raggruppato in tre cluster, mentre il restante 90% è sparso in modo casuale. Ovviamente la distribuzione dei cluster è rappresentata solo dal primo 10%, mentre gli altri punti si possono considerare outlier. Nell'operazione di media che identifica i centri cluster, il loro contributo è solo una distorsione, ossia la posizione trovata non corrisponde al centro del cluster, ma presenta un errore. (Si può pensare agli outlier come a una massa sparsa, la cui attrazione gravitazionale sposta il centroide rispetto alla posizione corretta definita dai punti del cluster.)

Nel confronto fra fuzzy *c*-Means e clustering parzialmente possibilistico, la tecnica asimmetrica è l'unica in grado di portare i centri cluster in posizione pressoché esatta, come evidenziato in figura 10.

Esistono altri studi che tendono a mettere in relazione gli approcci convenzionale e possibilistico [13, 26]. La tecnica che abbiamo presentato qui fornisce però una transizione continua dall'uno all'altro modello, ed è per questo motivo che esso presenta le interessanti proprietà illustrate fin qui.

4 Tecniche di clustering gerarchico

4.1 *Clustering agglomerativo gerarchico*

Con il nome di clustering agglomerativo gerarchico viene indicata una famiglia di algoritmi basata sull'analisi della matrice delle distanze (o similarità), che contiene le distanze tra tutte le possibili coppie di punti. Una volta misurata la distanza fra ciascuna coppia di dati, possiamo abbandonare la tabella dei dati e lavorare solo sulla matrice. Un effetto di questo modo di procedere è che non è necessario che i dati siano metrici (possono essere qualuunque tipo di oggetto) e che la matrice contenga vere e proprie distanze. Anche valutazioni soggettive, che non rispondono alla definizione di distanza, possono essere utilizzate.

Questo tipo di algoritmi è molto usato in biologia molecolare, in quanto adatto al confronto e all'analisi di espressione di gruppi anche estesi di geni [12, 15].

L’algoritmo di clustering gerarchico agglomerativo non è un algoritmo iterativo, ma *ricorsivo*. Esso non mira ad ottimizzare un criterio di clustering per approssimazioni successive, ma a creare una struttura gerarchica di cluster applicando la stessa regola ad ogni livello della gerarchia. Tale gerarchia viene

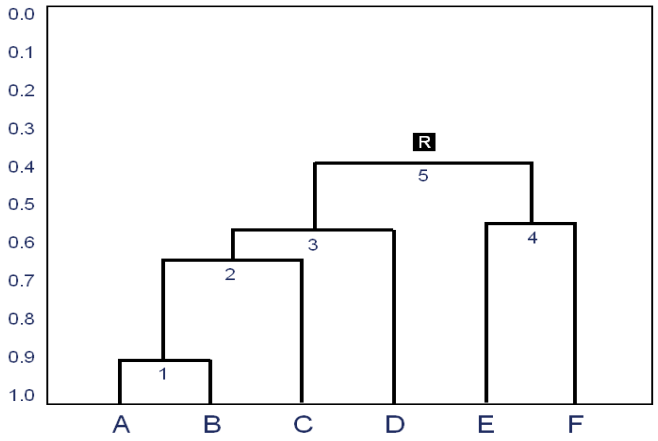


Fig. 11 – Un dendrogramma. A-F sono i dati. 1-5 sono i cluster (numerati in ordine di aggregazione). R è la radice.

poi visualizzata tipicamente attraverso un diagramma ad albero detto "dendrogramma" (figura 11), in cui la distanza o similarità tra due oggetti corrisponde alla posizione su un asse verticale (in figura calibrato da 0.0 a 1.0).

Possiamo schematizzare l'algoritmo come segue:

Al primo passo:

Unire i due oggetti più vicini (che sono dei dati) e formare un cluster di due, che è un nuovo "oggetto". Eliminare i due oggetti uniti e sostituirli con il cluster che li contiene.

Ad ogni passo successivo:

Unire i due oggetti più vicini (che sono due dati, o due cluster, o un dato e un cluster) e formare un nuovo cluster di livello gerarchico superiore. Eliminare i due oggetti uniti e sostituirli con il cluster che li contiene.

Risultato finale:

Una gerarchia di dicotomie dei dati, rappresentata p.es. con il familiare dendrogramma o altre rappresentazioni (p.es. altri tipi di diagramma ad albero).

Abbiamo detto che si tratta di una famiglia di algoritmi. Oltre alla scelta del tipo di distanza o similarità fra i dati, scelta che è comune agli altri metodi di clustering, in questo caso dobbiamo anche stabilire il modo in cui viene valutata la distanza fra *due cluster* (o fra un dato e un cluster). Poiché i cluster sono insiemi, esistono varie definizioni di distanza fra cluster.

Nella tecnica del *single linkage*, la distanza $dist_{ij}$ fra due cluster i e j è la minima distanza fra un dato del cluster i e un dato del cluster j. Per questo motivo si definisce anche tecnica del *nearest neighbour*. Tende a dar luogo a cluster poco compatti, in un effetto detto del *concatenamento*, che corrispondono a rami dell'albero molto allungati e poco ramificati.

Per contrastare questa tendenza, si può ricorrere alla tecnica del *complete linkage*. In questo caso la distanza $dist_{ij}$ fra due cluster i e j è la massima distanza fra un dato del cluster i e un dato del cluster j. Si definisce anche tecnica del *farthest neighbour*. Tende a dar luogo a cluster molto compatti e di dimensioni simili fra loro.

Average linkage è la tecnica per cui la distanza $dist_{ij}$ fra due cluster i e j *è* la distanza fra le medie del cluster i e del cluster j. In realtà, in questo caso esistono vari modi per definire la media. Il più usato è definito *unweighted pair-group method average*, UPGMA: la distanza fra due cluster è la media delle distanze fra un dato in i e un dato in j. Questa tecnica è semplice e pratica, ma funziona in modo ragionevole solo se la dissimilarità adottata rispetta la disuguaglianza

$$dist(\mathrm{AC}) \leq dist(\mathrm{AB}) + dist(\mathrm{BC})$$

Nel caso particolare in cui la dissimilarità è una distanza vera e propria, questa disuguaglianza è soddisfatta. Varianti prevedono l'uso dei centroidi (punti medi) o dei punti mediani. Si possono usare quindi solo per dati metrici.

Nel caso in cui i cluster siano sbilanciati in dimensione, si può modificare la tecnica precedente aggiungendo come pesi le cardinalità dei cluster (numero di oggetti contenuti). Si ottiene il metodo detto w*eighted pair-group method average* o WPGMA.

La tecnica detta *metodo di Ward* usa come "distanza" fra cluster la somma delle deviazioni quadratiche rispetto alla media. Due cluster sono considerati "i più prossimi" se la loro unione è quella che presenta l'aumento minimo nella somma delle deviazioni quadratiche rispetto ai due cluster di partenza.

4.2 *Osservazioni sulle tecniche agglomerative in alta dimensionalità*

Le tecniche di clustering agglomerativo gerarchico presentano come vantaggi principali semplicità e chiarezza. In certi ambiti la struttura ad albero che ne deriva è particolarmente adatta a rappresentare proprio la struttura che si ricerca nei dati, come nel caso di alberi filogenetici. Occorre tenere presente che ogni livello dell'albero comprende solo una diramazione in due, e questo può non essere un modello completamente fedele.

Si hanno tuttavia anche alcuni svantaggi. Il primo che possiamo citare è l'instabilità dell'albero o dendrogramma ottenuto (piccole variazioni nei dati possono indurre alberi differenti). Il rimedio per questo problema è usualmente ottenuto attraverso tecniche di *ricampionamento statistico*.

Un altro aspetto che può essere svantaggioso è il fatto che, alla fine dei conti, non si tratta di un vero algoritmo di clustering. Esso infatti non dà indicazioni sulla struttura "naturale" dei dati, perché è basato solo su dicotomie.

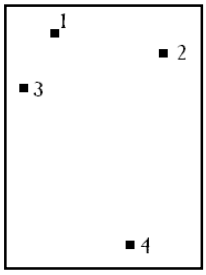
Un ulteriore svantaggio consiste nel fatto che, per avere un risultato utile, dobbiamo clusterizzare tutti i livelli gerarchici. Nel gergo del data mining (disciplina che si occupa del trattamento di grosse basi di dati): non è un algoritmo *anytime*, un algoritmo che dà subito una risposta grezza e procede a raffinarla, che quindi posso arrestare anytime, in qualunque momento.

Proponiamo qui una soluzione a questi problemi, propri delle tecniche agglomerative gerarchiche, e ad altri problemi più generali. Le difficoltà evidenziate derivano soprattutto dal fatto che l'algoritmo è agglomerativo, ossia procede in modo "bottom-up". Occorrerebbe invece utilizzare un algoritmo divisivo ("top-down"). Tipicamente, nel caso divisivo è possibile a ogni passo prendere decisioni di validità più ampia, mentre nel caso agglomerativo le decisioni sono prese solo su base locale.

Un altro problema è rappresentato dal calcolo delle distanze in spazi ad altissima dimensionalità. Già con d su valori moderati o piccoli (d = 10-15) si dimostra che, dato un insieme di punti e un punto "di query", la distanza rispetto al punto più prossimo e quella rispetto al più lontano tendono a coincidere [4].

Per ridurre l'effetto della convergenza delle distanze si utilizzare un accorgimento comunemente adottato in statistica, e cioè si può passare su scala ordinale. Non si usano più i valori delle distanze, ma solo i relativi ranghi, ossia le posizioni di ciascun dato in una lista ordinata per valore di distanza.

Un'ulteriore ostacolo all'uso di normali tecniche di clustering è costituito dalla cosiddetta "maledizione della dimensionalità" [10], quel fenomeno per



Punto	*1*	*2*	*3*	*4*
I vicino	3	1	1	3
II vicino	2	3	2	2
III vicino	4	4	4	1

Fig. 12 – Illustrazione delle proprietà del principio dei punti in prospettiva. A destra un insieme di dati e a sinistra i rispettivi ranghi delle distanze

cui le considerazioni geometriche valide in uno spazio a n dimensioni possono non essere più valide passando a $n + 1$ dimensioni.

Normalmente, un cluster può essere definito:

- da una elevata densità di punti (come nel caso dei metodi tipo *c*-Means);
- dai punti che si trovano vicini fra loro (come nel caso delle tecniche agglomerative)

In particolare, quando non ho almeno *altrettanti punti dati quante sono le dimensioni dello spazio*, la distribuzione dei punti è necessariamente confinata a un sottospazio dello spazio dei dati. In queste condizioni è difficile considerare valido il concetto di "densità" dei punti. D'altra parte abbiamo visto che il concetto stesso di "molto vicino" tende a perdere significato.

Occorre quindi operare in base a un principio differente

4.3 *Il principio dei "punti in prospettiva"*

Il principio che proponiamo qui è il seguente: definire i cluster come insiemi di punti che condividono lo stesso *punto più lontano*. Iterativamente, si può poi passare a considerare il secondo più lontano, il terzo, e così via.

Si opera analizzando per ogni punto la lista dei ranghi degli altri punti, ordinati per distanza rispetto ad esso. Il clustering avviene confrontando tali liste anziché i valori di distanza. Consideriamo un insieme di punti come appar-

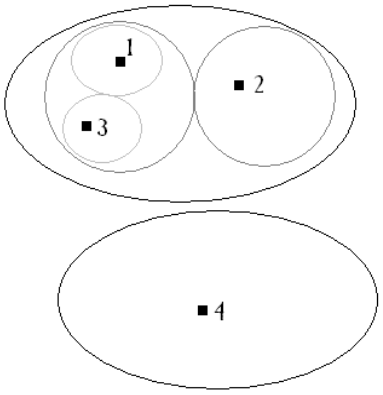


Fig. 13 – Esempio di clustering (si veda Fig. 12)

tenente allo stesso cluster se condividono lo stesso punto più lontano. Al livello *k*, valuteremo separatamente i punti presenti nei cluster al livello precedente, *k* – 1, e considereremo un sotto-cluster come formato da tutti i punti che hanno in comune lo stesso punto di rango *k*, ossia il *k*-esimo punto in ordine di lontananza.

Un algoritmo che usa un principio simile (ma ragionando sui punti comuni fra i più vicini) è la tecnica di Jarvis e Patrick [17].

Possiamo vedere che, a differenza delle tecniche di clustering agglomerativo, qui si usano punti di riferimento per definire i cluster; ma a differenza delle tecniche tipo *c*-Means, i punti di riferimento sono tratti dall'insieme dei dati, e non calcolati.

Di quali proprietà gode questo singolare principio di operazione?

Una importante proprietà è la seguente: dato, per ciascun punto, l'elenco dei restanti punti ordinati per distanza, punti vicini tendono ad avere il *nearest neighbor* differente e il *farthest neighbor* uguale.

L'illustrazione di questa proprietà è presentata in figura 12, dove sono mostrati sulla sinistra 4 punti dati, e sulla destra, per ciascuno di essi, la lista degli altri punti in ordine di lontananza.

Altri aspetti riguardano il fatto che viene utilizzata la matrice delle distanze, e quindi, una volta calcolate tutte le coppie di distanze, il procedimento è indipendente dalla dimensionalità dei dati.

I punti che appartengono a un cluster sono accomunati dall'avere un punto o insieme di punti in comune in coda alla lista ordinata dei vicini. Un insieme di

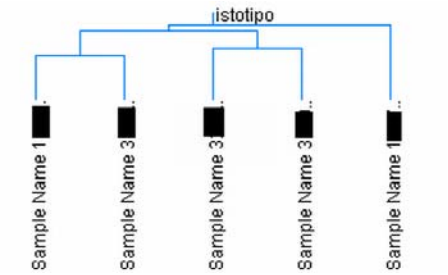


Fig. 14 – Dendrogramma per una verifica preliminare

punti, aventi il punto *j*-esimo come *k*-esimo elemento della lista dei vicini, viene ulteriormente suddiviso sulla base del (*k* + 1)-esimo elemento. Quindi il clustering procede in maniera gerarchica *divisiva***.**

Il procedimento di divisione si può arrestare (al raggiungimento di una data profondità o per assenza di ulteriori differenziazioni). Quindi la complessità computazionale è limitata, in quanto può essere controllata; normalmente, negli algoritmi divisivi, risulta inutile approfondire le suddivisioni oltre un certo limite perché esse risultano non statisticamente significative.

Non è dunque necessario effettuare la suddivisione di tutti i cluster fino al livello di singole osservazioni; mentre, per creare un albero agglomerativo, è necessario partire dalle singole osservazioni e arrivare fino alla radice.

Un aspetto importante è che i cluster non sono vincolati ad essere suddivisi in due sottocluster: se dai dati emerge una suddivisione differente, l'algoritmo è in grado di rilevarla. Si tratta quindi di un vero e proprio algoritmo di clustering, in quanto la struttura che esso suggerisce per i dati non è vincolata dall'algoritmo, ma scaturisce dai dati stessi. Tutto ciò mantenendo il concetto di gerarchia, che facilita lo studio del responso ottenuto.

Vediamo ora i risultati di alcune verifiche sperimentali. Per valutarne l'attendibilità, è stato effettuato preliminarmente un confronto con la tecnica agglomerativa gerarchica standard, applicata a dati da DNA microarray per lo studio dell'espressione di geni collegati a tumore polmonare, in 5 casi, rilevando l'espressione differenziale tra tessuto sano e tessuto tumorale.

I campioni sono molto pochi, quindi il risultato che ci si aspetta è che il dendrogramma ottenuto sia uguale a quello fornito dalla tecnica standard del clustering agglomerativo gerarchico. Questa prova in effetti fornisce esattamente il risultato desiderato, che è il dendrogramma di figura 14.

Abbiamo successivamente effettuato prove di clustering di dataset supervisionati. Questo per verificare se in problemi reali la suddivisione adottata dà luogo a cluster significativi. Il procedimento consiste nell'effettuare il clustering, e poi attribuire a ciascun cluster la classe che risulta maggioritaria fra i dati che esso contiene. I dati utilizzati sono i seguenti:

- Pima indians diabetes [7]
- Wisconsin diagnostic breast cancer [7]
- Leukemia (training set only) [15]
- Leukemia (training set + test set together) [15]
- Lyme disease [6, 25]

I primi due sono problemi di diagnosi clinica disponibili pubblicamente nell'*UCI Machine Learning Repository* [7]. I dati sulla leucemia sono tratti da [15] e riguardano la distinzione a livello molecolare tra i due sottotipi di leucemia acuta, mieloide e linfoblastica. Il problema "Lyme disease" è un problema di diagnosi clinica già studiato in passato da uno degli autori [6, 25].

I risultati sono illustrati in Tabella 1. Essi sono stati ottenuti con i seguenti parametri: distanza euclidea; pre-processing, quello indicato di volta in volta nella tabella. Come già detto, cluster ottenuti sono stati etichettati con il target maggioritario ("calibrati") e sono state valutate le proporzioni di classificazioni corrette.

Problema	*n*	Preprocessing	Errore %
PIMA INDIANS DIABETES – UCI	768	normalizzato rispetto a media/devst (parametri forniti dalle note UCI)	12.40%
WISCONSIN DIAGNOSTIC BREAST CANCER – UCI	569	normalizzato rispetto a media/2xdevst	5.60%
LEUKEMIA - SOLO TRAINING SET	38	nessuno	0.00%
LEUKEMIA - TOTALE	72	nessuno	6.90%
LYME DISEASE	684	normalizzato rispetto media/2*devst (parametri calcolati)	6.00%

Tab. 1 – Risultati delle prove su problemi supervisionati

Dai risultati qui presentati si può osservare come il metodo sia in grado di ottenere risultati paragonabili ai metodi supervisionati. Questo si evince dal confronto delle cifre (percentuale di errore) con quelle presentate nelle fonti originali dei problemi [6, 7, 15, 25], che sono di entità simile e in alcuni casi superiore. Questa osservazione suggerisce che il metodo sia in grado di sfruttare in modo molto ragionevole la struttura dei dati, in quanto esso è applicato senza utilizzare informazioni di tipo supervisionato nella fase di clustering, ma solo in quella di calibrazione, nella quale ormai la gerarchia è fissata.

5 Conclusioni

Abbiamo passato in rassegna una scelta di tecniche di analisi non supervisionata, analizzandone le applicazioni su dati da microarray a DNA. In particolare, ci siamo concentrati su quella che è probabilmente la categoria più importante (il clustering). Riteniamo alla fine della trattazione che sia opportuno sottolineare nuovamente come una buona conoscenza di tali tecniche sia necessaria (anche se non sufficiente) per un loro uso appropriato e soprattutto per una razionale interpretazione dei risultati sperimentali ottenuti.

Bibliografia

[1] Affymetrix, sito World Wide Web, http://www.affymetrix.com .

[2] Ball G. H. e Hall G. H. (1967), "ISODATA, an iterative method of multivariate analysis and pattern classification", Behavioral Science, vol. 12, pp. 153–155.

[3] Barni M., Cappellini V. e Mecocci A. (1996), "Comments on 'A Possibilistic Approach to Clustering'", IEEE Trans. on Fuzzy Systems, vol.4, no.3, pp.393-396

[4] Beyer K., Goldstein J., Ramakrishnan R. e Shaft U. (1999), "When is 'nearest neighbor' meaningful?", in 7th International Conference on Database Theory Proceedings (ICDT'99),. 217-35. Springer-Verlag.

[5] Bezdek, J. C. (1981) Pattern Recognition with Fuzzy Objective Function Algoritms, Plenum Press, New York

[6] Bianchi G., Buffrini L., Monteforte P., Rovetta G., Rovetta S. e Zunino R. (1994), "Neural approaches to the diagnosis and characterization of lyme disease", in Proceedings of the 7th IEEE Symposium on Computer-Based Medical Systems, IEEE Press, USA, pp. 194-199.

[7] Blake C. L. e Merz C. J. (1998), "UCI repository of machine learning databases", URL:http://www.ics.uci.edu/mlearn/MLRepository.html.

[8] Brazma, A. et al. (2001) "Minimum information about a microarray experiment (MIAME)-toward standards for microarray data", Nature Genetics 29, 365–371.

[9] Drago G. P. e Ridella S. (1999), "Possibility and necessity pattern classification using an interval arithmetic perceptron", Neural Computing and Applications, vol. 8, no. 1, pp. 40–52.

[10] Duda R. O. e Hart P. E. (1973), Pattern Classification and Scene Analysis, John Wiley and Sons, New York (USA).

[11] Dunn J. C. (1974), "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters", Journal of Cybernetics, vol. 3, pp. 32–57.

[12] Eisen M. B., Spellman P. T. , Brown P. O. e Botstein D. (1998), "Cluster analysis and display of genome-wide expression patterns", Proceedings of the National Academy of Sciences, Vol. 95, Issue 25, 14863-14868.

[13] Flores-Sintas A., Cadenas J. M. e Martin F. (1998), "Local geometrical properties application to fuzzy clustering", Fuzzy Sets and Systems, vol. 100, pp. 245–256.

[14] Fodor S. P., Read J. L., Pirrung M. C., Stryer L., Lu A. T., Solas D. (1991) "Light-directed, spatially addressable parallel chemical synthesis". Science 251, 767-773.

[15] Golub T. R. et al. (1999), "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring", Science, Vol. 286, 531–537.

[16] Ishibuchi H. e Nii M. (2000), "Neural networks for soft decision making", Fuzzy Sets and Systems, vol. 115, no. 1, pp. 121–140.

[17] Jarvis R. A. e Patrick E. A. (1973), "Clustering Using a Similarity Measure Based on Shared Near Neighbors", IEEE Transactions on Computers, C22, 1025-1034.

[18] Krishnapuram R. e Keller J. M. (1993), “A possibilistic approach to clustering”, IEEE Transactions on Fuzzy Systems, vol. 1, no. 2, pp. 98-110.
[19] Krishnapuram R. e Keller J. M. (1996), “The possibilistic C-Means algorithm: insights and recommendations”, IEEE Transactions on Fuzzy Systems, vol. 4, no. 3, pp. 385–393.
[20] Leung Y. F. e Cavalieri D. (2003), “Fundamentals of cDNA microarray data analysis”, TRENDS in Genetics vol. 19, No. 11, pp. 649-659
[21] Lockhart D. J. et al. (1996),“Expression monitoring by hybridization to high-density oligonucleotide arrays”, Nature Biotechnol. 14, 1675–1680.
[22] Masulli F. e Rovetta S. (2003), “An Algorithm to Model Paradigm Shifting in Fuzzy Clustering”, in B. Apolloni, M. Marinaro, R. Tagliaferri, curatori, “Neural Nets – 14th Italian Workshop on Neural Nets”, Lecture Notes in Computer Science, Vol. 2859, pp.70-76
[23] Masulli F. e Rovetta S. (2003), “The Graded Possibilistic Clustering Model”, in Proceedings of the 2003 International Joint Conference on Neural Networks, Portland, USA, July 2003, pp. 791-795.
[24] Miyamoto S. e Mukaidono M. (1997), “Fuzzy C-Means as a regularization and maximum entropy approach”, in Proceedings of the Seventh IFSA World Congress, Prague,, pp. 86–91.
[25] Moneta C., Parodi G., Rovetta S. e Zunino R. (1992), “Automated diagnosis and disease characterization using neural network analysis”, in Proceedings of the 1992 IEEE International Conference on Systems, Man and Cybernetics, IEEE Press, USA, pp. 123-128.
[26] Pal N. R., Pal K. and Bezdek J. C. (1997), “A Mixed c-Means Clustering Model”, in FUZZIEEE97, pp. 11-21.
[27] Quackenbush J. (2001), Computational analysis of microarray data”, Nature Review Genetics, Vol. 2, N. 6, pp. 418-427.
[28] Ridella S., Rovetta S. e Zunino R. (2000), “IAVQ: Interval-Arithmetic Vector Quantization for Image Compression”, IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 47, N. 12, pp. 1378-1390.
[29] Rose K., Gurewitz E. e Fox G. (1990), “A deterministic annealing approach to clustering”, Pattern Recognition Letters, vol. 11, pp. 589–594.
[30] Schena M., Shalon D., Davis R. W. e Brown, P. O. (1995) “Quantitative monitoring of gene expression patterns with complementary DNA microarray”, Science 270, 467-470.
[31] Schena, M. et al. (1996), “Parallel human genome analysis: microarray-based expression monitoring of 1000 genes”, Proc. Natl Acad. Sci. USA 93, 10614–10619.
[32] Wen, X. et al. (1998), “Large-scale temporal gene expression mapping of central nervous system development”, Proc.Natl Acad. Sci. USA 95, 334–339.
[33] Zadeh, L. A. (1965) , “Fuzzy sets”, Information and Control, Vol. 8, pp. 338-353.