DESPOTA: un approccio basato sui test di permutazione per la ricerca della partizione su un dendrogramma

Dario Bruzzese
Dip. di Sanità Pubblica, Università di Napoli Federico II
dbruzzes@unina.it

Domenico Vistocco
Dip. di Economia e Giurisprudenza, Università di Cassino
vistocco@unina.it

1. Introduzione

La classificazione gerarchica è uno dei metodi di classificazione maggiormente utilizzati in molti contesti applicativi (Everitt et al., 2001). La possibilità di scegliere tra differenti partizioni alternative in funzione del livello di omogeneità interno delle corrispondenti classi è sicuramente uno dei punti di maggiore interesse dei metodi gerarchici che spesso li porta a preferire ai tradizionali metodi di partizionamento. La naturale rappresentazione grafica dell'insieme delle partizioni risultanti da un algoritmo di classificazione gerarchica è il dendrogramma. Nella scelta della partizione si procede usualmente utilizzando un taglio orizzontale del dendrogramma ma così facendo una serie di partizioni ospitate nell'albero non saranno mai esplorate. Tali partizioni potranno infatti essere individuate solo utilizzando un taglio su livelli differenti. La proposta mira appunto ad esplorare l'intero insieme di partizioni disponibili e sfrutta i test di permutazione per effettuare tale ricerca partendo dalla radice dell'albero e scendendo fino agli elementi terminali dello stesso. Un ulteriore vantaggio dell'algoritmo proposto è l'individuazione automatica del numero di classi da scegliere, caratteristica

questa che rende una tecnica implementabile agevolmente in sistemi di classificazione automatica.

2. L'algoritmo DESPOTA

L'algoritmo ripercorre la struttura ad albero di un dendrogramma seguendo però il percorso inverso a quello della sua generazione, dal momento che procede dal nodo radice sino al raggiungimento della regola di arresto.
Si indichi allora con n il numero di oggetti da classificare e con k il livello dell'albero in corrispondenza del quale due generiche classi L_k e R_k (Figura 1) sono state aggregate ($k=1,\ldots,n-1$ con k=1 che indica il livello del nodo radice); L_k fa riferimento alla classe di sinistra mentre R_k a quella di destra, in accordo a come tali classi sono disposte sul dendrogramma. Sia inoltre $h(L_k \cup R_k)$ l'altezza alla quale le due classi sono state aggregate e infine con $h(L_k)$ e $h(R_k)$ l'altezza in corrispondenza della quale le classi L_k ed R_k si sono costituite; nel caso di classi singleton l'altezza è posta pari a 0.

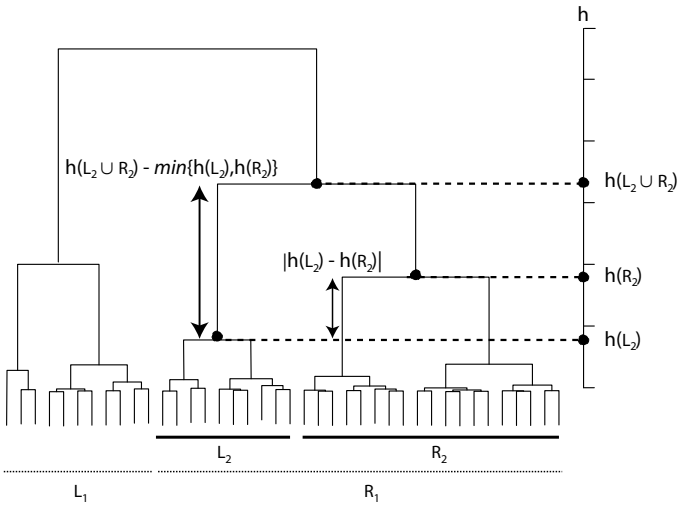

Figura 1: Notazione essenziale utilizzata a partire dal dendrogramma

Durante la fase di discesa, per ogni partizione che si incontra, da $k=1$ fino *a* $k=n-1$, si effettua un test di permutazione (Good, 1994) sulle due classi L_k ed R_k al fine di valutare se le stesse possano essere considerate provenienti dalla medesima popolazione o se invece queste debbano essere considerate

provenienti da popolazioni distinte. Per alcuni dettagli relativi al test utilizzato si rimanda al paragrafo seguente.
Nel caso in cui non si rifiuta l'ipotesi nulla si aggiunge la classe risultante dall'unione delle due classi L_k ed R_k all'insieme delle partizioni individuate Nelle successive fasi di discesa tutte le classi in L_k ed R_k non saranno più esplorate.
Se invece si rifiuta l'ipotesi nulla, durante la discesa sull'albero, l'algoritmo eseguirà un test anche sulle classi che si trovano innestate in L_k ed R_k. L'algoritmo si arresta quando non ci sono ulteriori classi da esaminare o nel caso limite in cui si raggiunge il livello k = n-1. L'algoritmo è presentato in maniera dettagliato nello schema seguente:

```
Input: Un dataset e il corrispondente dendrogramma
Output: Una partizione del dataset
1 inizializzazione:
2   livelliAggregazioneDaVisitare ← h(L1 ∪ R1)
3   classiIndividuate ← [ ]
4   k ← 1
5 repeat
6    if si accetta H0 (le due classi rappresentano
                          un unico cluster) then
7      • aggiungi Lk ∪ Rk all'insieme
         classiIndividuate
8    else
9      • aggiungi h(Lk) e h(Rk) a
         livelliAggregazioneDaVisitare
10     • ordina livelliAggregazioneDaVisitare in
         ordine crescente in funzione di h(.)
11   end
12   elimina il primo elemento da
              livelliAggregazioneDaVisitare
13 k ← k+1
14 until livelliAggregazioneDaVisitare è vuoto
```

3. La statistica test utilizzata

La statistica test utilizzata per stabilire se due classi debbano essere considerate provenienti o meno da una stessa popolazione può essere illustrata facendo riferimento nuovamente al dendrogramma presentato in Figura 1. In particolare, il valore assoluto della differenza tra L_k e R_k può essere interpretato come il costo minimo necessario per unire le due classi: un aumento del-

la misura di dissimilarità di una quantità pari a | $h(L_k)$ - $h(R_k)$ | porterebbe infatti all'unione delle due classi. Se si considera invece la differenza tra $h(L_k \cup R_k)$ e il valore più piccolo tra $h(L_k)$ e $h(R_k)$, si ottiene invece il costo effettivamente necessario per unire le due classi. Il rapporto tra i due suddetti costi

$$rc_k = \frac{\left|h(L_k) - h(R_k)\right|}{h(L_k \cup R_k) - \min\left[h(L_k), h(R_k)\right]}$$

rappresenta una misura normalizzata tra 0 ed 1 che caratterizza il processo di aggregazione. Tale rapporto è utilizzato come statistica test per valutare la similarità tra due classi nei successivi passi di discesa dell'albero. Un valore basso di rc_k segnala che il costo necessario per unire le due classi L_k e R_k è sensibilmente più grande del costo minimo ed è perciò da interpretarsi come segnale di presenza di due differenti classi. Nel caso opposto in cui il valore è vicino all'unità, il costo sostenuto per unire le due classi non è molto differente dal costo minimo, segnalando che le due classi possono essere interpretate come provenienti dalla stessa popolazione. L'ipotesi nulla testata dalla procedura di permutazione afferma che le due popolazioni $\mathcal{L}_k$ e $\mathcal{R}_k$ da cui sono estratti i due cluster osservati L_k e R_k debbano essere considerate come un unico gruppo C_k:

$$H_0: \mathcal{L}_k \equiv \mathcal{R}_k \equiv C_k.$$

Al fine di derivare la distribuzione di permutazione della statistica test sotto H_0, ad ogni passo della procedura gli elementi di L_k e R_k vengano permutati in modo da ottenere due classi L_k^m e R_k^m di pari cardinalità. Sotto l'ipotesi nulla la permutazione delle osservazioni di L_k e R_k è giustificata in quanto l'ipotesi di scambiabilità (Pesarin e Salmaso, 2010), richiesta nella teoria dei test di permutazione, è soddisfatta. Tenendo conto del fatto che il processo di classificazione è invariante a permutazioni delle osservazioni, si procede a generare un dendrogramma su ciascuna delle due classi L_k^m e R_k^m ottenute in seguito alla permutazione e si calcola la statistica sopra illustrata a partire dai due dendrogrammi ottenuti. Utilizzando l'approccio tipico dei test di permutazione, ripetendo il processo per M volte si ottiene la distribuzione della statistica di permutazione rc_k^m. Sotto H_0 il costo relativo calcolato sulle due classi permutate L_k^m e R_k^m non dovrebbe essere dissimile dal valore rc_k osservato sul dendrogramma iniziale. Al contrario, valori di rc_k^m più piccoli di rc_k (e quindi vicini a 0) segnalano la presenza di due gruppi distinti. Il p-value Monte Carlo può essere quindi agevolmente calcolato come proporzione dei valori in cui $rc_k^m \leq rc_k$.

Per ulteriori dettagli tecnici relativi alla statistica test si rimanda il lettore interessato al contributo (Bruzzese e Vistocco, 2012).

In Figura 2(a) è riportata la partizione ottenuta applicando l'algoritmo DESPOTA sul dataset E. Coli; il dataset contiene 336 proteine E. Coli e sette

caratteristiche calcolate sulle sequenze di amino acidi (Horton e Nakai, 1996). Il dendrogramma è stato ottenuto utilizzando la distanza euclidea e il criterio di agglomerazione di Ward. I quattro riquadri evidenziano la partizione ottenuta usando un livello di significatività $\alpha=0.01$ e un numero di ripetizioni $M=999$ per i test di permutazione. Sotto ciascun riquadro è riportato il p-value. L'ellisse corrisponde al ramo k=5 in corrispondenza del quale l'ipotesi nulla è stata rifiutata (p=0.005). Il riquadro tratteggiato, infine, fa riferimento al precedente livello k=4 in cui l'ipotesi nulla non è stata rifiutata (p=0.17). Le corrispondenti distribuzioni della statistica test, RC_4 *e* RC_5, sono riportate in Figura 2(b). Dall'analisi della figura risulta che sebbene i valori della statistica osservata nei due casi siano simili, $rc_4=0.45$ *e* $rc_5=0.6$, le differenze nella posizione e nella forma delle due distribuzioni portano a decisioni differenti nei due casi.

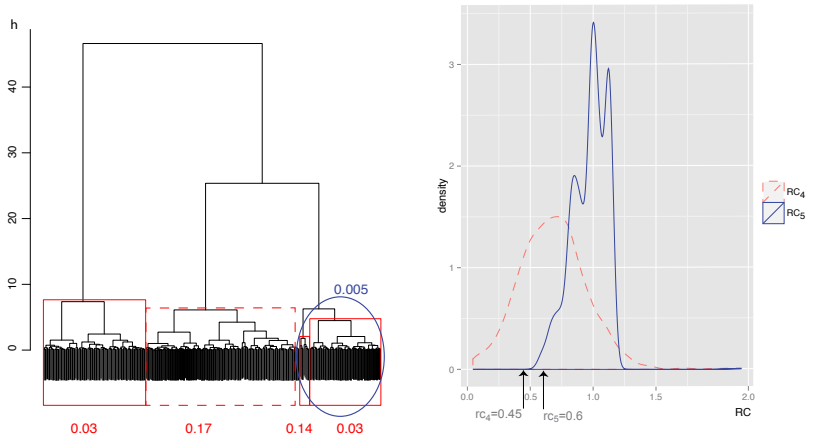

Figura 2 – La partizione ottenuta sul dataset E. Coli (a) e l'andamento della statistica test per i due livelli k=4 e k=5

Considerazioni conclusive

La procedura proposta genera in modo automatico il taglio di un dendrogramma prodotto da una classificazione gerarchica. La procedura è stata testata sia su dati reali che su dati sintetici. Per ragioni di spazio non è possibile presentare i risultati che sono riportati in dettaglio in (Bruzzese e Vistocco, 2012), unitamente ad un confronto con tecniche aventi finalità simili proposte in letteratura. In tutti i casi analizzati, le soluzioni ottenute risultano stabili rispetto al numero di cicli *Montecarlo* utilizzati. La scelta del livello di significatività consente inoltre di modulare il livello di granularità della partizione

finale. Ciò che la caratterizza, rispetto ad altri criteri automatici, è la possibilità che alle classi estratte possano associarsi differenti livelli dell'indice usato nella procedura di aggregazione. E' così possibile esplorare la bontà di soluzioni che non sarebbero altrimenti prese in considerazione ricorrendo a tagli orizzontali dell'albero.

Un aspetto meritevole di ulteriori approfondimenti riguarda il problema del multiple testing (Romano et al., 2008), che può essere generato dall'applicazione ripetuta di test nei vari passi dell'algoritmo. Una soluzione agevolmente praticabile potrebbe essere basata sull'uso di un livello di significatività flessibile, che si abbassa quando si scende sui livelli successivi dell'istogramma. In particolari le soluzioni proposte da Hochberg (1998) e Holm (1979) potrebbero essere facilmente implementabili nell'algoritmo DESPOTA. Entrambe sono soluzioni conservative e si differenziano per il tipo di approccio utilizzato: la prima usa infatti una strategia di tipo step-up mentre la seconda una strategia di tipo step-down.

Bibliografia

Everitt, B., Landau, M. e Leese, M. (2001) Cluster Analysis, 4th edition, Arnold, London, UK

Pesarin, F., Salmaso L. (2010) Permutation Tests for Complex Data. Theory, Applications and Software, John Wiley & Sons Ltd, Chichester, UK.

Good, P.I. (1994) Permutation Tests for Testing Hypothesis, Springer-Verlag, New York.

Bruzzese, D. e Vistocco, D. (2012) DESPOTA: DEndrogram Slicing through a PermutatiOn Test Approach, mimeo.

Horton, P., Nakai K. (1996). A probabilistic classification system for predicting the cellular localization sites of proteins. Proc. Int. Conf. Intell. Syst. Mol. Biol. (4), 109-115.

Romano, J.P., Shaikh, A.M. e Wolf, M. (2008). Formalized data snooping based on generalized error rates. Econometric Theory (24), 404-447.

Hochberg, Y. (1988). A sharper Bonferroni procedure for multiple tests of significance, Biometrika (75), 800-802.

Holm, S. (1979). A simple sequentially rejective multiple testing procedure. Scandinavian Journal of Statistics (6), 65-70.