Journal of Classification 32:285-304 (2015)
DOI: 10.1007/s00357-015-9179-x

DESPOTA: DEndrogram Slicing through a PemutatiOn Test Approach

Dario Bruzzese

Department of Public Health, University of Naples “Federico II”, Italy

Domenico Vistocco

Department of Economics and Law, University of Cassino, Italy

Abstract: Hierarchical clustering represents one of the most widespread analytical approaches to tackle classification problems mainly due to the visual powerfulness of the associated graphical representation, the dendrogram. That said, the requirement of appropriately choosing the number of clusters still represents the main difficulty for the final user. We introduce DESPOTA (**DE**ndrogram **S**licing through a **P**ermutati**O**n **T**est **A**pproach), a novel approach exploiting permutation tests in order to automatically detect a partition among those embedded in a dendrogram. Unlike the traditional approach, DESPOTA includes in the search space also partitions not corresponding to horizontal cuts of the dendrogram. Applications on both real and syntethic datasets will show the effectiveness of our proposal.

Keywords: Hierarchical clustering; Cluster detection; Permutation tests.

The authors wish to thank Professor Ibai Gurrutxaga and his colleagues for kindly providing the data and the R code used in their paper: this allowed us to make a worthwhile comparison of the two methods. The authors are also grateful to Professor Jaromir Antoch for helpful comments on a previous draft of the paper and the three anonymous referees for their valuable suggestions which helped us to improve the final version of this paper.

All computation and graphics were done in the R language (R Development Core Team 2010) using the basic packages and the additional cluster (Maechler et al. 2011), ggplot2 (Wickham 2009) and NbClust (Charrad et al. 2013) packages.

Authors’ Addresses: D. Bruzzese, Dept. of Public Health, University of Naples “Federico II”, Via S. Pansini, 5, I-80131, Naples, Italy, email: dario.bruzzese@unina.it; D. Vistocco, Dept. of Economics and Law, University of Cassino, Via S. Angelo, Località Folcara I-03043 Cassino (Frosinone), ITALY, email: vistocco@unicas.it.

Published online: 8 July 2015

1. Introduction

There are several fields that require objects to be classified starting from a set of attributes: life and medical sciences, economics, social sciences, engineering, and computer sciences to mention only a few (Everitt, Landau, and Leese 2001). Unsupervised classification, also named clustering, consists in determining a set of groups such that objects belonging to the same group are similar and in the meanwhile objects from different groups are dissimilar. Clustering algorithms can be roughly divided into partitioning and hierarchical algorithms: the former provide the user with a unique partition of the available objects starting from a specified number of groups, whereas the latter provide a set of partitions among which the user is called to select the appropriate one.

An effective representation of the output of a hierarchical algorithm is the dendrogram, which permits the whole spectrum of available partitions to be visualized on a tree-like structure. However, the selection of a unique partition is usually required for hierarchical clustering to be operative and, in common practice, this task is accomplished by cutting the tree through a horizontal line such that the resulting connected branches of the tree define the elements of the partition. This approach has two well known drawbacks: the discretion of the cut level and the inappropriateness in detecting not well-separated uniform clusters. The general indication of cutting the dendrogram next to a large jump between two subsequent tree branches is quite rational and limits the user discretion but it remains a rule of thumb. Moreover, clusters could differ in terms of their own internal consistency in a way that the same threshold value would not be suitable for all of them.

Several criteria have been proposed in the literature to select the 'best' partition by optimizing some cluster validity indexes (see e.g. Milligan 1981; Milligan and Cooper 1985; Kim and Ramakrishna 2005; Wu, Yang, and Hsieh 2009; Lago-Fernández and Corbacho 2010). The seminal work of Milligan (1981) describes above thirty internal criterion measures coming from a wide variety of fields. Their behavior as stopping rules in a hierarchical clustering context has been studied in a later paper (Milligan and Cooper 1985) by taking either the optimum value across hierarchy levels or the maximum difference between successive levels of the hierarchy. In this work, the best three performing rules emerging from Milligan and Cooper (1985), the Calinski and Harabasz (CH) index (Calinski and Harabasz 1974), the Duda and Hart (Duda) index (Duda and Hart 1973) and the Hubert and Levin (C-Index) index (Hubert and Levin 1976), will be used to place our proposal in the mainstream literature.

More recently, Gurrutxaga et al. (2010) combined a new cluster validity index with a searching strategy (SEP/COP) for exploring the extended hierarchy housed in a dendrogram, i.e. a strategy able to select partitions not necessarily corresponding to horizontal cuts of the dendrogram. In the wake of this approach, our proposal still exploits the extended hierarchy as the reference solution set but it takes advantage of a permutation test (Pesarin and Salmaso 2010) to extract the 'best' solution. We feel that the proposed approach is truer to the spirit of hierarchical clustering as the same guiding principle is used both to grow the tree and to choose the final solution, as will be shown. A competing method (GAP), proposed by Tibshirani, Walther, and Hastie (2001), permits to estimate the number of clusters starting from the output of any clustering algorithm. It is based, like DESPOTA, on a Monte Carlo approach to derive the reference distribution of a test statistic but it requires as input the different partitions among which the optimal one has to be selected. Our method, instead, locally explores the branches of the dendrogram picking up those clusters that will join the final partition.

Meanwhile our proposal differs from those suggested in Shimodaira (2004), Ryota and Shimodaira (2011), Liu, Hayes, Nobel, and Marron (2008) and Park, Manjourides, Bonetti, and Pagano (2009), whose aim is to assess the significance of all the clusters in a dendrogram without offering a validated partition to the final user. In consideration of the above, the proposed method is well-suited to be implemented in an automatic pattern recognition decision system.

The rest of the paper is organized as follows. Section 2 provides the main notation along with the motivation behind our work. Section 3 details the general framework of DESPOTA, i.e. the algorithm and the permutation test structure. A working example is shown to see DESPOTA in action. Section 4 shows some results both on real and synthetic datasets through a comparison with the SEP/COP algorithm, with whom we share the idea of the extended hierarchy as the reference solution set, with the GAP method, with whom we have in common the Monte Carlo framework, and with the three previously stated stopping rules. Some concluding remarks and avenues for future research follow in the final section.

2. Preliminary Remarks

Before describing our basic idea, let us recall that given a set $\mathcal{X} = \{x_1, x_2, ..., x_N\}$ of N $p-$dimensional objects, a partition P of $\mathcal{X}$ is defined as a collection of clusters $\{C_1, C_2, \ldots, C_g\}, g \in \{1, 2, \ldots N\}$, such that the following hold: $C_i \subseteq \mathcal{X}$, $\bigcup_i C_i = \mathcal{X}$ and $C_i \cap C_j = \emptyset$, $\forall i \neq j \in \{1, 2, \ldots, g\}$.

Due to the traditional approach that exploits horizontal lines for cutting the dendrogram, hierarchical clustering provides the user with a hierarchy of nested partitions, $\mathcal{H} = \{P_1 \supset P_2 \supset \ldots \supset P_N\}$, where the inclusion relation between two partitions means that each cluster belonging to a partition P_s is a subset of a cluster of a partition P_r iff $s > r$. This feature limits the search space in which to look for the optimal partition by bounding the cardinality of $\mathcal{H}$ to N because, for each desired number of clusters g, there is only one element of $\mathcal{H}$ providing g groups. But, actually, only P_1, P_2 and P_N are unique, given the dendrogram. Partitions P_1 and P_N are the two (unique) trivial solutions corresponding, respectively, to a single cluster containing all the data points and to N singleton clusters, whereas P_2 is the unique solution giving two clusters complying with the hierarchical agglomeration process.

However, the dendrogram houses an extended hierarchy of partitions $\mathcal{H}^* = \left\{\left\{P^1\right\}, \left\{P^2\right\}, \ldots, \left\{P^N\right\}\right\}$ because the set of partitions in g clusters still complying with the hierarchical agglomeration process, i.e. $\{P^g\}$, has cardinality $\#(P^g) = (g-1) \cdot \#\left(P^{g-1}\right)$, $g \notin \{1, 2, N\}$, where $\#(\mathcal{S})$ denotes the cardinality of a generic set $\mathcal{S}$. This extended hierarchy, $\mathcal{H}^*$, is a superset of $\mathcal{H}$ and is characterized by the following property that relaxes the inclusion ordering relation between any pair of element of $\mathcal{H}$: for each $P^* \in \{P^g\}$ there exists a $P^+ \in \left\{P^{g-1}\right\}$ such that $P^+ \subset P^*$.

As an example, Figure 1 shows the dendrogram obtained on a synthetic dataset containing four different clusters of the same cardinality generated from multivariate normal distributions with different mean vectors and different variance-covariance matrices. The Ward criterion with the Euclidean distance was used to grow the tree.

Panels (a), (b) and (c) of Figure 1 highlight the partitions in two, three and four groups, respectively, belonging to the hierarchies $\mathcal{H}$ and $\mathcal{H}^*$. The above property of $\mathcal{H}^*$ can be verified, for instance, by observing that each element of $\left\{P^4\right\}$ is included (at least) in a partition belonging to $\left\{P^3\right\}$. The solid line in panel (c) shows the 4-cluster solution obtained by cutting the dendrogram with a traditional horizontal criterion; this partition is the only one belonging to $\mathcal{H}$ that can produce a 4-cluster solution and isolates a very small cluster on the left-hand side of the dendrogram while leaving grouped the two clusters on the right that, on the contrary, contain units belonging to different populations. The 'true' partition, still housed in the dendrogram, can be revealed only by cutting the tree with the dashed line characterized by two local thresholds with different heights. It is evident that this partition belongs to $\mathcal{H}^*$.

The widening of the search space from $\mathcal{H}$ to $\mathcal{H}^*$ makes it worth implementing a procedure able to automatically explore the extended set of partitions in $\mathcal{H}^*$ in order to find the optimal solution: the proposed algorithm, DESPOTA, exploits the theoretical framework of permutation tests to reach this goal. A considerable 'side effect' of our approach is the automatic

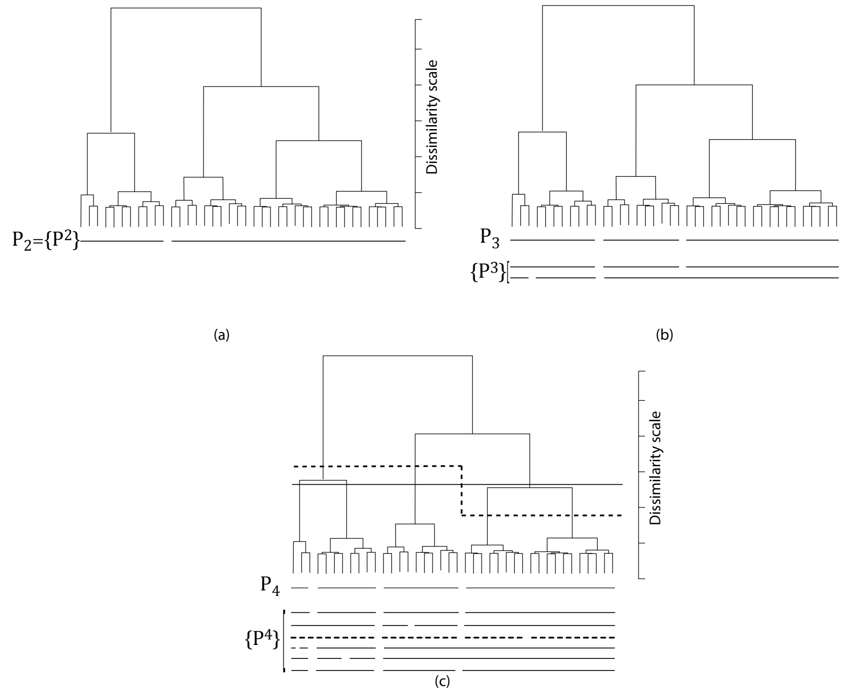

Figure 1. Panels (a), (b) and (c) show the partitions $\mathcal{H}$ and $\mathcal{H}^*$ obtained on a synthetic dataset for $g = \{2, 3, 4\}$ groups , respectively. The elements of $\mathcal{H}$ and $\mathcal{H}^*$ are shown below each dendrogram. Panel (c) reveals the only possible partition in 4 groups belonging to $\mathcal{H}$ corresponding to a traditional horizontal cut. However the 'true' solution, shown through a dashed line, is contained in $\mathcal{H}^*$ and can be revealed only by using a non-horizontal cut approach.

identification of the number of clusters, thus avoiding the 'user dilemma' regarding the appropriate choice of g.

3. The Proposal

3.1 Outline of the Procedure

The algorithm retraces the tree downward, starting from the root of the dendrogram, where all objects are classified in a unique cluster, and moving down a *partial threshold* until a link joining two clusters is encountered. A permutation test is then performed in order to verify whether the two clusters should be considered a single group (the null hypothesis) or not (the alternative one). If the *Null* cannot be rejected, the corresponding branch will become an element of the final partition and none of its sub-branches will be processed any longer[1]. Otherwise each of them will be further visited in the course of the procedure. Actually, in both cases, the partial threshold will continue its path down the tree until there are no more branches that stand the test (i.e. the *Null* can no longer be rejected).

The proposed procedure is detailed in Algorithm 1, where we use $k = 1, \dots, N-1$ to denote the level at which two generic clusters are merged during the hierarchical agglomerative process. In particular, $k = 1$ refers to the uppermost link of the dendrogram, i.e. the one that produces the partition P^1, while $N-1$ refers to the lowest link of the dendrogram where the two most similar singletons are merged and the partition P^{N-1} is obtained.

We use L_k and R_k to refer to the left and right cluster (according to the dendrogram visualization) joined at level k. Let $h(.)$ be the metric used in the hierarchical classification process which depends on the distance function between two objects (i.e. Euclidean, Manhattan, etc.) and on the aggregation method for computing the distance between two clusters (i.e. Single linkage, Complete linkage, Ward method, etc.)[2].

Finally, $h(L_k)$ and $h(R_k)$ are the values of the dissimilarity measure at which L_k and R_k are obtained, while $h(L_k \cup R_k)$ denotes the value of the dissimilarity measure required to merge the two cluster at level k. For the sake of illustration, Figure 2 depicts the above defined quantities for $k = 2$ with respect to the dendrogram obtained on the synthetic dataset introduced in the previous section.

The values of $h(.)$ of all the clusters that have to be processed are stored in the vector $aggregationLevelsToVisit$ which is initialized with $h(L_1 \cup R_1)$. This vector is dynamically updated during the procedure by adding and/or removing elements according to the permutation test results. The counterpart of the vector $aggregationLevelsToVisit$ is the vector $detectedClusters$ populated by all the clusters for which the $Null$ is accepted. At the end of the algorithm, the $detectedClusters$ vector will contain the detected partition.

[1] Actually, it could happen that sub-branches of a cluster for which the null was not rejected may result as distinct clusters at a lower level of the dendrogram. However, besides any computational considerations, an exhaustive search on all the nodes of the dendrogram could provide a plethora of alternative solutions contrasting with the original aim of automatically detecting a partition.

[2] When a reversal occurs in a dendrogram, a problem that can arise in case of Centroid and Median linkage, the procedure solves it by *reflecting* the corresponding height.

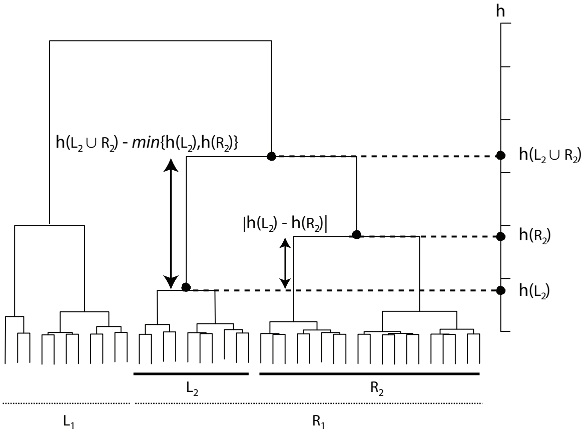

Figure 2. Exemplification of the main notation adopted.

3.2 The Test Statistic and the Permutation Reference Distribution

In this section, we describe the statistic allowing us to establish the similarity between two clusters that form the basis of the proposed permutation procedure and the corresponding reference distribution.

According to the notation introduced in Figure 2, the absolute value of the difference between $h\left(L_k\right)$ and $h\left(R_k\right)$ ($k = 1, ..., N-1$) can be considered as the *minimum cost* necessary to merge the two classes; minimum because the dissimilarity measure used in the agglomeration process must rise by at least $|h(L_k) - h(R_k)|$ in order to merge the two clusters. The difference between $h\left(L_k \cup R_k\right)$ and $\min\left\{h(L_k), h(R_k)\right\}$ can instead be considered as the *cost* actually incurred for merging L_k and R_k.

The ratio between these two costs:

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

is thus a normalized measure, ranging from 0 to 1, that characterizes the aggregation process resulting in the new class $L_k \cup R_k$ and is indeed used as the test statistic in the permutation test. A value of rc_k close to 0 reflects a situation where the cost required to merge the two clusters is very much larger than minimum cost necessary to merge L_k and R_k. Therefore it is reasonable to decide in favour of the presence of two different clusters. On the other hand, a value of rc_k close to 1 denotes two clusters requiring for their merging a cost which does not differ greatly from the minimum necessary cost. In this case the observed rc_k suggests that the two groups should be considered as a unique cluster.

```
Input: A dataset and its related dendrogram
Output: A partition of the dataset
initialization:
aggregationLevelsToVisit ← $h(L_1 \cup R_1)$
detectedClusters ← [ ]
k ← 1
repeat
    if $H_0$ is accepted (the two clusters represent a unique class) then
        . add $L_k \cup R_k$ to detectedClusters
    else
        . add $h(L_k)$ and $h(R_k)$ to aggregationLevelsToVisit
        . sort aggregationLevelsToVisit in descending order according to $h(.)$
    end
    remove the first element from aggregationLevelsToVisit
    k ← k+1
until aggregationLevelsToVisit is empty
```

Algorithm 1: The $DESPOTA$ algorithm

The null hypothesis of the permutation procedure states that the two population clusters $\mathcal{L}_k$ and $\mathcal{R}_k$, from which the two (observed) clusters L_k and R_k are extracted, actually represent a unique group $\mathcal{C}_k$, i.e.;

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

In order to derive the permutation distribution of the test statistic under H_0, let L_k^m and R_k^m denote the two new classes obtained by permuting the elements between L_k and R_k, preserving the original cardinality of both the clusters. Under the Null, the permutation of the statistical units between L_k and R_k is theoretically justified because the exchangeability assumption is satisfied (Good 1994).

As a matter of fact, the hierarchical clustering process is invariant with respect to the permutation of the original observations and thus growing a single dendrogram on the permuted set would simply re-establish the same structure. For this reason, after L_k^m and R_k^m are obtained, a new dendrogram is generated for each of them according to the dissimilarity metric $h(.)$ used to grow the original tree. The heights of the root nodes of the corresponding dendrograms, $h\left(L_k^m\right)$ and $h\left(R_k^m\right)$, allow us to define the ratio:

$$rc_k^m = \frac{|h(L_k^m) - h(R_k^m)|}{h\left(L_k^m \cup R_k^m\right) - \min\left\{h(L_k^m), h(R_k^m)\right\}}.$$

Repeating the permutation step M times, the values of rc_k^m $(m = 1, \ldots, M)$ will provide the permutation distribution of the test statistic RC_k.

It is worth noting that the value of $h\left(L_k^m \cup R_k^m\right)$ appearing in the denominator of rc_k^m has to be computed by mimicking the chosen agglomeration method as the two clusters L_k^m and R_k^m are distinctly managed in each permutation step. As a consequence, the numerator of rc_k^m is not longer necessarily a part of the denominator, thus permitting values greater than 1. From the graphical point of view (see Figure 2) this means that the minimum cost incurred to merge the two permuted clusters can be higher than the actual one.

Under H_0 we expect that the relative cost computed on the statistical units of L_k^m and R_k^m should be close to that observed on the original dendrogram rc_k. Viceversa, values of rc_k^m $(m = 1, \ldots, M)$ on the left of rc_k, and thus near to 0, indicate the presence of two homogeneous groups that could be formed. For this reason the Monte Carlo p-value is computed as the proportion of times the observed value rc_k is greater than or equal to the permuted values rc_k^m :

$$p = \frac{\#\left(rc_k^m \leq rc_k\right)}{M}.$$

3.3 A Working Example

In this section we present a working example of $DESPOTA$ using the *E. Coli* dataset which contains 336 proteins from E. Coli classified into eight classes. Seven attributes were computed from the amino acid sequences (Horton and Nakai 1996). The left-hand panel of Figure 3 shows the corresponding dendrogram obtained by using the Euclidean distance and the Ward agglomeration criterion. The four boxes highlight the partition detected by $DESPOTA$ (in this example the significance level α was set equal to 0.01 and M equal to 999). For each cluster the associated p-values are reported. The ellipse corresponds to a branch ($k = 5$) of the tree for which the null hypothesis was rejected ($p = 0.005$), while the dashed box refers to the previous level where the $Null$ was not rejected ($k = 4, p = 0.17$). The corresponding reference permutation distributions of the two test statistics, RC_4 and RC_5, are shown in the right-hand panel of Figure 3. Although the observed value of the statistic were similar ($rc_4 = 0.45, rc_5 = 0.6$), differences in shape and location between the two distributions led to rejection of the $Null$ for $k = 5$ and to its acceptance for $k = 4$.

4. Main Results

In the following subsections some major results of the $DESPOTA$ algorithm will be shown with respect to both synthetic and real datasets.

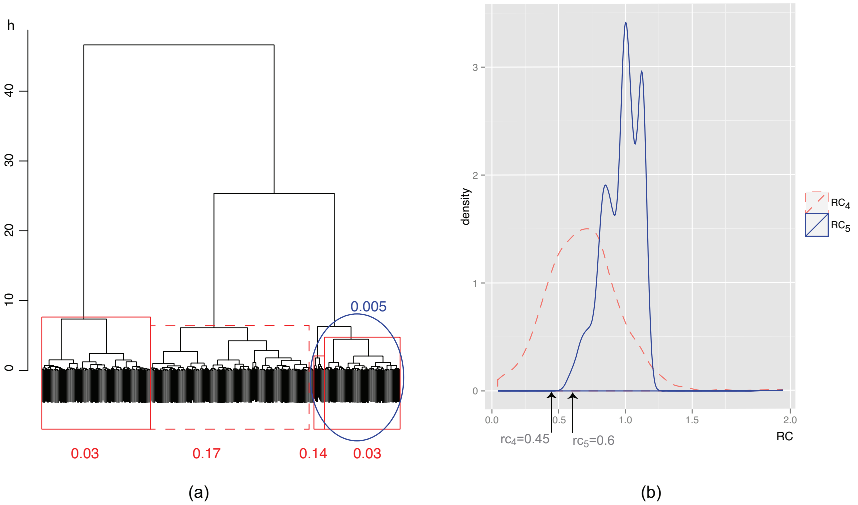

Figure 3. Panel (a) shows the partition obtained by applying $DESPOTA$ on the *E. Coli* dataset. The boxes refer to the four detected clusters while the ellipse refers to a branch of the dendrogram for which $H_0 : \mathcal{L}_k \equiv \mathcal{R}_k \equiv \mathcal{C}_k$ was rejected. Numbers near the boxes and the ellipse show the p-values associated to each cluster. The solid line in panel (b) shows the reference permutation distribution of the corresponding test statistic. The dashed line in panel (b) refers to the permutation distribution associated to a branch of the dendrogram for which the $Null$ was not rejected ($k = 5$); this branch is highlighted in panel (a) using a dashed box. Although the observed value of the statistic were similar, DESPOTA reached different decisions due to the different shape and location of the permutation distributions.

The performance of the proposed method will be compared with that of the GAP and the SEP/COP algorithms and with the results provided by the main cluster validity indexes (CH, Duda and C-Index) in order to highlight values and defects.

The synthetic data subsection will mainly evaluate the effect of the dataset features, in terms of number of clusters and number of variables, along with the effects of tuning parameters on the detected solution. The real data subsection will instead focus on the ability of $DESPOTA$ to grasp the real cluster structure embedded in the data.

4.1 Synthetic Data

The DESPOTA algorithm was tested and compared with the aforementioned competitors on several synthetic datasets. The first simulation concerns the scenario of unclustered data: 100 different data sets each of which consists of 100 observations uniformly distributed over the 10-dimensional unit hypercube. Both GAP and DESPOTA always succeed in pointing out the lack of structure while SEP/COP often fails to detect such pattern. With respect to the three validity indexes, actually they cannot reveal unclustered data as they require at least a two cluster solution to be computed; however DUDA index was the only criterion suggesting the simplest solution (two clusters) while the other two indexes provided inappropriate results (data not shown).

Clustered data were instead generated according to the random cluster generation method proposed in Qiu and Joe (2006; 2009). Generated data differ in terms of number of clusters ($c = 2, 3, 4, 5, 6, 7$) and number of variables ($p = 5, 10, 15$). For each combination of c and p, $s = 100$ different datasets were simulated.

The artificial data were generated using a value of 0.01 for the separation index (Qiu and Joe 2006) between any cluster and its nearest neighbor cluster; indeed, this value reflects a close cluster structure. The method used to generate the covariance matrix for clusters exploits the spectral decomposition of a symmetric matrix by first randomly generating p eigenvalues ($\lambda_1 >, \dots, > \lambda_p$) and then using the columns of a randomly generated orthogonal matrix $\boldsymbol{Q} = (\boldsymbol{q}_1, \dots, \boldsymbol{q}_p)$ as the corresponding eigenvectors. Finally, the covariance matrix $\boldsymbol{\Sigma}$ is obtained as $\boldsymbol{Q}\ diag(\lambda_1, \dots, \lambda_p)\ \boldsymbol{Q}^\top$. The cluster sizes were randomly generated from a pre-specified range $[50 - 200]$ in order to produce reasonable variability of cluster sizes. The R-code (R Development Core Team 2010) to reproduce the synthetic data is provided in Appendix 1.

In case of compact clusters, Ward's method appears best and is somewhat robust against the dimension, number and separation of clusters (Kuiper and Fisher 1975). As our simulation settings reflect such cluster characteristics, in this subsection we focused exclusively on Ward's aggregation criterion.

Figure 4 compares SEP/COP, GAP, DESPOTA, CH, DUDA and C-Index on the simulated datasets for $p = 10$ with respect to the number of detected clusters. For the DESPOTA algorithm two different significance levels ($\alpha = 0.01, 0.05$) were used. Each point in the plot denotes the number of detected clusters for a single dataset. Different panels refer to different values of c. Points were jittered to reduce overplotting. It is simple to grasp the overestimation of the cluster structure produced by the SEP/COP algorithm which increases with the number of clusters in the dataset (this pattern is not directly visible from the figure as, for sake of clarity, those points denoting a number of cluster greater than 10 have been collapsed). Having so many detected clusters clearly represents a serious drawback for the actual usability of the procedure. The GAP method performs well in the different settings, especially when c is small ($c \leq 5$), but showing a wors-

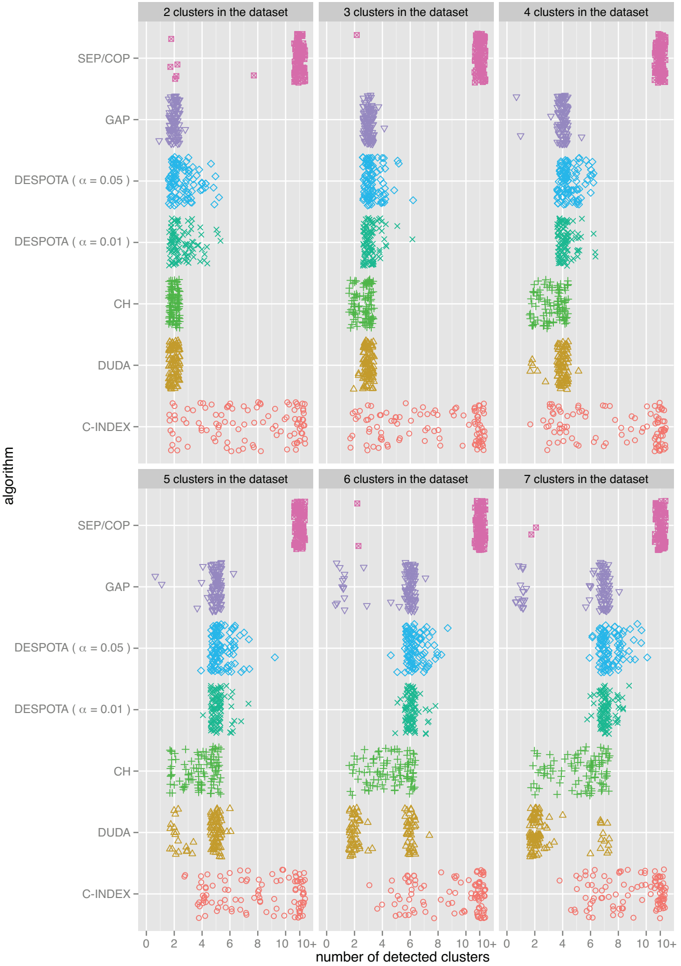

Figure 4. Comparison among SEP/COP, GAP, DESPOTA (using two different levels of α), CH, DUDA and C-Index on synthetic data. Each panel refers to a different simulation setting providing 100 different datasets with $p = 10$ variables for each value of $c \in \{2, 3, 4, 5, 6, 7\}$. Each point denotes the number of detected clusters for a single dataset; a small amount of jitter was used to avoid overplotting. To improve readability the solutions with a number of detected clusters greater than 10 have been collapsed.

ening with the increase of the number of groups. In particular, in case of an high number of clusters, it often erroneously points out a lack of structure. The non-monotone behavior of the GAP index could explain such feature. Actually, as the authors suggest, ‘it is important to examine the entire GAP curve rather than simply to find the position of its maximum’ (Tibshirani, Walthier, and Hastie 2001). However in an automatic pattern recognition framework such solution is hardly practicable. With respect to the three cluster validity indexes, DUDA and CH show a good performance in case of a small number of clusters but drift to underestimate the solution when the number of cluster becomes higher, with DUDA showing less variability. The unreliable behavior of C index is, instead, evident from the wide range of the obtained solutions.

The third and fourth row of Figure 4, namely $DESPOTA\,(\alpha = 0.05)$ and $DESPOTA\,(\alpha = 0.01)$, offer a better insight into the performance of the proposed algorithm allowing to evaluate its sensitivity with respect to the number of clusters and to the significance level. It can be noticed that a significance level of 0.01 always yields the best results also in terms of the variability in the number of detected clusters (only in the case of $c = 2$, do the results actually appear unstable).

Similar results were observed for the other two settings ($p = 5$ and $p = 15$) and thus, for the sake of space, they will not be shown.

It is worthy of notice that due to the data generating process, the simulated data do not stress one of the main feature of both DESPOTA and SEP/COP algorithm, i.e. their searching path along the extended hierarchy. The following subsection will permit to point out such peculiarity.

4.2 Real Data

In the following DESPOTA and its competitors are compared using several real datasets, some of which have been discussed in Gurrutxaga et. al (2010) including also some additional datasets commonly used in different classification works (Johnson and Wichern 1982; Banfield and Raftery 1993). In Appendix 2 information on the data sources is provided.

In the general framework of hierarchical classification (Wishart 1969), five agglomeration methods were considered: single linkage (sl), complete linkage (cl), average linkage (al), centroid (cen) and Ward's method (w). These agglomeration criteria are those implemented in the main statistical software.

The comparison consisted in the following steps:

- For each dataset and for each agglomeration method the corresponding dendrogram was created.
- For each dendrogram, all the partitions housed in the hierarchy were compared with the a-priori partition using the Hubert and Arabie Adjusted Rand Index (ARI) (Hubert and Arabie 1985), a corrected-for-chance version of the Rand index; the partition associated to the maximum observed value of ARI was chosen as benchmark[3].

[3] ARI can take negative values, and its upper bound is 1. The closer ARI values are to 1, the better the agreement between cluster assignments.

- The performance of the three algorithms (SEP/COP, GAP and DESPOTA) and of the three cluster validity indexes (CH, DUDA and C-Index) was then evaluated both in terms of ARI and in terms of the number of detected clusters.

From Table 1 it clearly stands out the poorest performance of CH and C-index that, in almost all datasets and agglomeration methods, climb down the whole hierarchy suggesting the trivial solution of $n - 1$ singletons.

For many datasets the largest difference among the other methods relates to the single linkage. While SEP/COP and DUDA almost always detect two clusters, GAP in many cases points out a lack of structure (see Section 4.1) and DESPOTA is instead more variable. It is worth noting that the two-cluster solution in the case of single linkage criterion corresponds to an unrealistic partition where a singleton is opposed to a cluster including all the other units. The large number of clusters detected by DESPOTA is most likely tied to the chaining effect of the Single Linkage (Everitt, Landau, and Leese 2001), as only for such criterion this drawback is observed. If we instead focus on the widely used Ward's Sum of Squares criterion, DESPOTA provides solutions nearer to the benchmark.

In some cases DESPOTA overperforms the benchmark: this happens when the detected partition corresponds to a non-horizontal cut of the dendrogram. As an example, Figure 5 depicts the dendrogram for the *Crude Oil* dataset obtained using the Centroid criterion for growing the tree. The different letters at the bottom of the tree refer to the three different classes in the natural partition. Despota points out a non-horizontal three clusters solution that is the closest to the real structure whereas the horizontal three clusters solution would bunch the two smaller classes while taking out a singleton from the largest class. SEP/COP suggests a partition in only two clusters while the benchmark solution leads to a too thin granularity resulting in seven groups.

As a general remark, it may be observed that some datasets reveal a weak cluster structure with respect to the ARI values. In particular, such index is less than 0.20 for all the classification criteria in the *Cereals*, *Vehicle* and *Yeast* datasets. According to the results in Steinley (2004) and Warrens (2008), this corresponds to a very poor agreement between the detected and the natural partition. Similar results are obtained also when using a non hierarchical approach, thus denoting an authentic lack of separability in the data. Notwithstanding they have been kept in the analysis for the sake of comparison with previous classification works.

Table 1. Comparison of SEP/COP, GAP, DESPOTA ($\alpha = 0.01$), CH, DUDA and C-Index on real data in terms of the Arabie Adjusted Rand Index (ARI). The dimensions of each dataset are shown in square brackets while *k* refers to the number of natural clusters. Each rows corresponds to a different agglomeration criterion: single linkage (sl), complete linkage (cl), average linkage (al), centroid (cen) and Ward's method (w). The *bmk* column concerns the benchmark partition, i.e. the best partition detectable using a traditional cut approach. Each cell is composed of the following information: value of the AR index (number of clusters associated to the specific partition). For the SEP/COP and the DESPOTA algorithm, asterisks denote non horizontal partitions.

		Adjusted Rand Index						
		bmk	sep/cop	gap	despota	CH	Duda	C-Index
	sl	0.11 (3)	-0.01 (2*)	0 (1)	0.08 (4*)	0.00 (42)	0.10 (4)	0.01 (39)
Cereals	cl	0.06 (2)	0.07 (5)	0 (1)	0.04 (3)	0.00 (42)	0.04 (3)	0.01 (39)
[43×8]	al	0.11 (3)	-0.01 (2)	0 (1)	-0.01 (2)	0.00 (42)	-0.01 (9)	0.01 (39)
$k = 3$	cen	0.11 (3)	-0.01 (2)	0 (1)	-0.01 (2)	0.00 (42)	0.11 (3)	0.01 (39)
	wl	0.06 (2)	0.07 (5*)	0 (1)	0.06 (2)	0.00 (42)	0.00 (7)	0.01 (4)
	sl	0.37 (7)	-0.02 (2)	0 (1)	-0.02 (2)	0.00 (55)	-0.02 (2)	0.00 (55)
Crude Oils	cl	0.44 (2)	0.55 (11*)	0 (1)	0.44 (2)	0.00 (55)	0.44 (2)	0.00 (55)
[56×5]	al	0.51 (3)	0.55 (11*)	0 (1)	0.12 (6*)	0.00 (55)	0.51 (3)	0.00 (55)
$k = 3$	cen	0.49 (7)	0.44 (2)	0 (1)	0.51 (3*)	0.00 (55)	0.44 (2)	0.00 (55)
	wl	0.23 (7)	-0.02 (2)	0 (1)	0.22 (4)	0.00 (55)	0.22 (4)	0.00 (55)
	sl	0.41 (12)	0.01 (2)	0 (1)	0.40 (13*)	0.00 (144)	0.01 (2)	0.00 (144)
Diabetes	cl	0.58 (7)	0.32 (3)	0.35 (2)	0.35 (2)	0.00 (144)	0.32 (3)	0.00 (144)
[145×3]	al	0.54 (11)	0.30 (7*)	0.35 (2)	0.27 (10*)	0.00 (144)	0.32 (3)	0.00 (144)
$k = 3$	cen	0.51 (10)	0.30 (7*)	0.35 (2)	0.31 (4*)	0.00 (144)	0.35 (2)	0.00 (144)
	wl	0.64 (3)	0.30 (7*)	0.35 (2)	0.64 (3)	0.00 (144)	0.61 (4)	0.00 (144)
	sl	0.05 (18)	0.01 (2)	0 (1)	0.11 (54*)	0.00 (335)	0.00 (2)	0.00 (335)
E.Coli	cl	0.69 (3)	0.69 (3)	0.69 (3)	0.40 (2)	0.00 (335)	0.69 (3)	0.00 (335)
[336×7]	al	0.74 (8)	0.76 (14*)	0 (1)	0.03 (2)	0.00 (335)	0.40 (3)	0.00 (335)
$k = 8$	cen	0.43 (18)	0.01 (2)	0 (1)	0.01 (2)	0.00 (335)	0.00 (2)	0.00 (335)
	wl	0.70 (5)	0.69 (3)	0.69 (3)	0.72 (4*)	0.00 (335)	0.70 (5)	0.00 (335)
	sl	0.08 (18)	0.01 (2)	0 (1)	0.23 (47)	0.00 (350)	0.00 (2)	0.00 (350)
Ionosphere	cl	0.20 (11)	0.21 (36*)	0.18 (4)	0.18 (3*)	0.00 (350)	0.18 (5)	0.00 (350)
[351×34]	al	0.11 (16)	0.01 (2)	0 (1)	0.01 (3)	0.00 (350)	0.00 (2)	0.00 (350)
$k = 2$	cen	0.08 (18)	0.01 (2)	0 (1)	0.06 (13)	0.00 (350)	0.00 (2)	0.00 (350)
	wl	0.32 (4)	0.31 (9*)	0.12 (14)	0.32 (4)	0.00 (350)	0.19 (6)	0.00 (350)
	sl	0.57 (2)	0.57 (2)	0.57 (2)	0.56 (3)	0.00 (147)	0.57 (2)	0.00 (147)
Iris	cl	0.64 (3)	0.64 (3)	0.59 (4)	0.42 (2)	0.00 (147)	0.59 (4)	0.00 (147)
[150×4]	al	0.76 (3)	0.57 (2)	0.76 (3)	0.64 (8*)	0.00 (147)	0.76 (3)	0.00 (147)
$k = 3$	cen	0.71 (8)	0.57 (2)	0.57 (2)	0.57 (2)	0.00 (147)	0.57 (2)	0.00 (147)
	wl	0.76 (3)	0.57 (2)	0.68 (4)	0.57 (2)	0.00 (147)	0.45 (6)	0.00 (147)
	sl	0.01 (7)	0.00 (2)	0.00 (2)	0.09 (41*)	0.00 (845)	0.00 (2)	0.00 (3)
Vehicle	cl	0.13 (5)	0.04 (2)	0.10 (3)	0.05 (3*)	0.00 (845)	0.11 (8)	0.10 (3)
[846×18]	al	0.14 (10)	0.06 (2)	0.06 (2)	0.16 (10*)	0.00 (845)	0.07 (3)	0.06 (2)
$k = 4$	cen	0.12 (6)	0.08 (2)	0.08 (2)	0.07 (7*)	0.00 (845)	0.08 (2)	0.08 (2)
	wl	0.16 (7)	0.03 (3*)	0.14 (6)	0.15 (4*)	0.00 (845)	0.10 (3)	0.14 (4)
	sl	0.01 (37)	0.01 (5*)	0.01 (6)	0.01 (42*)	0.00 (1453)	0.00 (2)	0.00 (1453)
Yeast	cl	0.13 (12)	0.00 (2)	0 (1)	0.00 (2)	0.00 (1453)	0.00 (2)	0.00 (1453)
[1484×8]	al	0.15 (26)	0.00 (2)	0 (1)	0.01 (3)	0.00 (1453)	0.00 (2)	0.00 (1453)
$k = 10$	cen	0.02 (38)	0.00 (2)	0.00 (3)	0.00 (2)	0.00 (1453)	0.00 (2)	0.00 (1453)
	wl	0.15 (7)	0.12 (42*)	0.15 (7)	0.14 (9*)	0.00 (1453)	0.15 (7)	0.00 (1453)

5. Conclusions

The output of hierarchical clustering methods is typically displayed as a dendrogram describing a family of nested partitions. However, the exploitable partitions are usually restricted to those relying on horizontal cuts of the tree, missing the possibility to explore the whole set of partitions housed in the dendrogram. In this paper, we proposed an algorithm, DESPOTA, exploiting the methodological framework of permutation tests, that permits a partition to be automatically found where clusters do not nec-

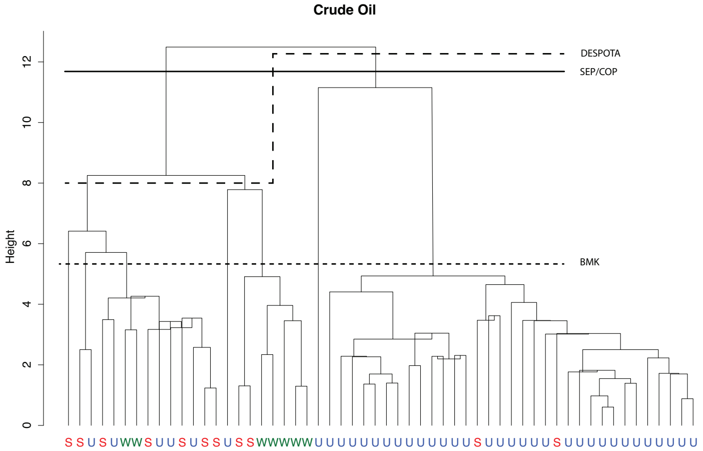

Figure 5. Comparison of the DESPOTA, SEP/COP and benchmark solutions for the *Crude Oil* dataset (Ward distance with Centroid criterion). The letters at the bottom of the tree refer to the three different classes in the natural partition.

essarily obey the above principle. Our solution adapts to every choice of the distance metric and agglomeration criterion used to grow the tree. The results obtained both on synthetic and real datasets show that DESPOTA performs well in situations characterized by different data and cluster structures.

An issue worth investigating concerns the multiple testing problem (Romano, Shaikh, and Wolf 2008). To this aim, a practicable solution could consist in the use of an adaptive significance level, i.e. in automatically lowering the value of α when moving down the dendrogram. Among the approaches available in literature, the step-up (Hochberg 1988) or the step-down (Holm 1979) strategy to adjust p-value could be easily implemented in DESPOTA and should permit to control multiplicity of the proposed approach even if in a conservative way. We are confident that this strategy would also solve the weak instability of the results which emerged from our analysis of the synthetic datasets.

Appendix 1. R Code for Synthetic Data Generation

```r
library(clusterGeneration)

genDataset <- function(numberOfClusters, numberOfVariables){
       outputList <- genRandomClust(numClust=numberOfClusters,
                                   sepVal=0.01,
                                   numNonNoisy=numberOfVariables,
                                   numNoisy=0,
                                   numOutlier=0,
                                   numReplicate=1,
                                   fileName="cl1")
        data <- data.frame(outputList$datList$cl1)
        clusters <- outputList$memList$cl1
        list(data=data, clusters=clusters)
}

variables2use <- c(5,10,15)
numberOfClusters2use <- 2:7
listData100seeds<-lapply(variables2use,
                         function(y)
                           lapply(numberOfClusters2use,
                            function(x)
                                 lapply(1:100,function(i,x,y){
                                  set.seed(i)
                                  genDataset(x, y)
                                 },
                                 x, y)))

#listData100seeds[[i]][[j]][[k]]$data points to:
# - the k-th generated dataset (k = 1 : 100),
# - for the j-th required number of clusters (j = 2 : 7)
# - with the i-th number of variable (i in [5,10,15])

#listData100seeds[[i]][[j]][[k]]$clusters points to the
# corresponding cluster membership
```

Appendix 2. Real Datasets Documentation

[**Cereals Dataset**]

Johnson, R.A. and Wichern, D.W. (1982). Applied Multivariate Statistical Analysis. Prentice Hall, pp 666–667.
Website: http://www.public.iastate.edu/~maitra/stat501/datasets/

[**Crude Oils Dataset**]
Johnson, R.A. and Wichern, D.W. (1982). Applied Multivariate Statistical Analysis. Prentice Hall, pp 662–663.
Website: http://www.public.iastate.edu/~maitra/stat501/datasets/

[**Diabetes dataset**]
R Package: mclust
G.M. Reaven and R.G. Miller, Diabetologica 16:17-24 (1979).

[**E.coli Dataset**]
R Package: MMST
A. Izenman (2008), Modern Multivariate Statistical Techniques, Springer.

[**Ionosphere dataset**]
R Package: MASS
Sigillito, V.G., Wing, S.P., Hutton, L.V., Baker, K.B. (1989). Classification of radar returns from the ionosphere using neural networks. Johns Hopkins APL Technical Digest, 10, 262-266.

[**Iris Dataset**]
R Package: Datasets
Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) The New S Language. Wadsworth Brooks/Cole.

[**Vehicle dataset**]
R Package: MASS
Turing Institute Research Memorandum TIRM-87-018 ”Vehicle Recognition Using Rule Based Methods” by Siebert,JP (March 1987).

[**Yeast Dataset**]
R Package: MMST
A. Izenman (2008), Modern Multivariate Statistical Techniques, Springer.

References

BANFIELD, J.D., and RAFTERY, A.E. (1993), “Model Based Gaussian and Non Gaussian Clustering”, *Biometrics, 49*, 803–821.

CALINSKI, R.B., and HARABASZ, J. (1974), “A Dendrite Method for Cluster Analysis”, *Communications in Statistics, 3*, 1–27.

CHARRAD M., GHAZZALI N., BOITEAU V., HUBERT M., and NIKNAFS A. (2013), *An Examination of Indices for Determining the Number of Clusters: NbClust Package*, R Package Version 1.3.

DUDA, R.O., and HART, P.E. (1973), *Pattern Classification and Scene Analysis*, New York: Wiley.

EVERITT, B., LANDAU, M., and LEESE, M. (2001), *Cluster Analysis* (4^{th} ed.), London: Arnold.

GOOD, P.I. (1994), *Permutations Tests for Testing Hypotheses*, New York: Springer-Verlag.

GURRUTXAGA, I., ALBISUA, I., ARBELAITZ, O., MARTÌN, J.I., MUGUERZA, J., PÈREZ, J.M., and PERONA, I. (2010), "SEP/COP: An Efficient Method to Find the Best Partition in Hierarchical Clustering Based on a New Cluster Validity Index", *Pattern Recognition, 43(10),* 3364–3373.

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.

HORTON P., and NAKAI K. (1996), "A Probabilistic Classification System for Predicting the Cellular Localization Sites of Proteins", *Proceedings of the International Conference on Intelligent Systems for Molecular Biology, 4,* 109–115.

HUBERT, L.J., and LEVIN, J.R. (1976), "A General Statistical Framework for Assessing Categorical Clustering in Free Recall", *Psychological Bulletin, 83,* 1072–1080.

HUBERT, L., and ARABIE, P. (1985), "Comparing Partitions", *Journal of Classification, 2,* 193–218.

JOHNSON, R.A., and WICHERN, D.W. (1982), *Applied Multivariate Statistical Analysis*, Upper Saddle River, NJ: Prentice Hall.

KIM, M., and RAMAKRISHNA, R.S. (2005), "New Indices for Cluster Validity Assessment", *Pattern Recognition Letters, 26(15)*, 2353–2363.

KUIPER, K.K., and FISHER, L. (1975), "A Monte Carlo Comparison of Six Clustering Procedures", *Biometrics, 31,* 777–783.

LAGO-FERNÁNDEZ, L.F., and CORBACHO, F. (2010), "Normality-Based Validation for Crisp Clustering", *Pattern Recognition, 43,* 782–795.

LIU, Y., HAYES, D.N., NOBEL, A., and MARRON, J.S. (2008), "Statistical Significance of Clustering for High-Dimension, Low-Sample Size Data", *Journal of the American Statistical Association, 103(483)*, 1281–1293.

MAECHLER M., ROUSSEEUW P., STRUYF A., HUBERT M., and HORNIK K. (2011), *Cluster: Cluster Analysis Basics and Extensions,* R Package Version 1.14.1.

MILLIGAN, G.W. (1981), "A Monte Carlo Study of Thirty Internal Criterion Measures for Cluster Analysis", *Psychometrika, 46(2),* 187–199.

MILLIGAN, G.W., and COOPER, M.C. (1985), "An Examination of Procedures for Determining the Number of Clusters in a Dataset", *Psychometrika, 52(2),* 159–179.

PARK, P.J., MANJOURIDES, J., BONETTI, M., and PAGANO, M. (2009), "A Permutation Test for Determining Significance of Clusters with Applications to Spatial and Gene Expression Data", *Computational Statistics and Data Analysis, 53(12),* 4290–4300.

PESARIN, F., and SALMASO, L. (2010), *Permutation Tests for Complex Data. Theory, Applications and Software*, Chichester: John Wiley and Sons.

QIU, W.L.. and JOE, H. (2006), "Separation Index and Partial Membership for Clustering", *Computational Statistics and Data Analysis, 50,* 585–603.

QIU, W.L.. and JOE, H. (2006), "Generation of Random Clusters with Specified Degree of Separation", *Journal of Classification, 23(2),* 315–334.

QIU, W.L., and JOE, H. (2009). *ClusterGeneration: Random Cluster Generation (with Specified Degree of Separation)*, R package version 1.2.7.

R DEVELOPMENT CORE TEAM (2010), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0, http://www.R-project.org.

ROMANO, J.P., SHAIKH, A.M., and WOLF, M. (2008), “Formalized Data Snooping Based on Generalized Error Rates”, *Econometric Theory, 24,* 404–447.

RYOTA, S., and SHIMODAIRA, H. (2011), *pvclust: Hierarchical Clustering with P-Values via Multiscale Bootstrap Resampling*, R package version 1.2-2, http://CRAN.R-project.org/package=pvclust.

SHIMODAIRA, H. (2004), “Approximately Unbiased Tests of Regions Using Multistep-Multiscale Bootstrap Resampling”, *Annals of Statistics, 32,* 2616–2641.

STEINLEY, D. (2004), “Properties of the Hubert-Arabie Adjusted Rand Index”, *Psychological Methods, 9(3),* 386–396.

TIBSHIRANI, R., WALTHER, G., and HASTIE, T. (2001), “Estimating the Number of Clusters in a Data Set via the Gap Statistic, *Journal of Royal Statistical Society B, 83(2),* 411–423

WARRENS, M.J.(2008), “On the Equivalence of Cohens Kappa and the Hubert-Arabie Adjusted Rand Index”, *Journal of Classification, 25,* 177–183.

WICKHAM, H. (2009), *ggplot2: Elegant Graphics for Data Analysis*, New York: Springer.

WISHART D. (1969),“An Algorithm for Hierarchical Classification”, *Biometrics, 25,* 165–170.

WU, K.-L., YANG, M.-S., and HSIEH, J.-N. (2009), “Robust Cluster Validity Indexes”, *Pattern Recognition, 42(11),* 2541–2550.