Improved and Novel Cluster Analysis for Bioinformatics, Computational Biology and All Other Data

Ruming Li 1, Xiu-Qing Li 2*, and Guixue Wang 3*

1, 2 Molecular Genetics Laboratory, Potato Research Centre, Agriculture and Agri-Food Canada
850 Lincoln Road, P.O. Box 20280, Fredericton, New Brunswick, E3B 4Z7, Canada
3 Key Laboratory of Biorheological Science and Technology (Chongqing University), Ministry of Education; Bioengineering College of Chongqing University, Chongqing, 400044, China

Abstract - Cluster analysis has been widely used in bioinformatics or biology to classify objects (items, DNA bands, markers, genes, individuals, species, taxa, etc.). Theoretically, there are numerous clustering methods available but only those that are well-established and proven are commonly used in practice. For their improved applicability, those methods that exploit the most data information and yield the better cluster properties should be focused and new algorithms are expected. The other issues and problems that are relevant to the performance of cluster analysis also need to be addressed. The well-prepared data can be significant to ameliorate the clustering results. They include the proper measure conversion from similarities to dissimilarities or distances and the necessary data standardizing transformation. Apart from z-score standardization, average- or mean-based scaling was introduced to achieve comparability among variables with the least value manipulation. A solution was given to improve updating the distance matrix with a centroid-related equation. A novel method called the percent similarity cluster analysis (PSCA) was devised to analyze the DNA band or marker data from electrophoresis and all other data.

Keywords: cluster analysis, clustering methods, clustering improvement, special clustering, data similarity and transformation, data scaling and standardization.

1 Introduction

Cluster analysis is a classification technique that is used to divide a set of objects (aka, entities, cases, points, observations, samples, or items) into separate subsets or clusters (aka, classes, groups or partitions). In the object space, each object has a multi-dimensional space that consists of a set of variables (aka, attributes, characters, or dimensions). The aim is to partition objects into mutually exclusive clusters such that the members in a cluster are sufficiently similar to each other and sufficiently dissimilar to non-members in other clusters [1,2]. Clustering can group data into a previously unknown or preset number of homogeneous clusters [3,4]. The resulting clusters can be visualized in a dendrogram (a tree diagram) that displays the fusions or divisions made at each successive step of cluster analysis.

There has been a rapid growth in the literature so far that addresses cluster analysis with somewhat different concepts and arguments since 1960. A broad range of sciences have been involved in adopting these clustering techniques, with varied assumptions, settings, and outcome interpretations. And a growing number of software programs for performing cluster analysis and the formation of cliques of cluster analysis users also make its application diverse. In response to some issues of cluster analysis, it is expected that there should be a practical approach that integrates and optimizes the existent clustering methods based upon their well-elucidated mathematic theory. Since cluster analysis is carried out as related to interdisciplinary realms, additional algorithms are needed to take care of data types that come from discipline-specific sources such as genomics in biology. Taking electrophoresis-aided banding technology for example, it has become the currently critical tool for us to ascertain significant information in biology as well as in other sciences. With this technology, a great volume of bands have spawned from PCR, DNA/RNA test, proteomics analysis and so on that provide a massive yet unique source of data [5,6]. Although DNA microarray-based gene expression data can be treated as quantitative and analyzed with a classic clustering method, electrophoresis-based band or marker data need to be otherwise tackled [7]. Unfortunately the traditional clustering algorithms are not adequate for all the questions in bioinformatics and computational biology at present. One of the questions, for instance, is the multiple "tied" objects that can be merged at once, rather than only two are joined at a time. There are certain inconsistencies among clustering algorithms and issues on some ambiguous notions that lead to different results. Other questions are that binary data processing by simple matching is unsuitable for band or marker data that can be multi-state, and that clustering a range of similar bands or markers by a similarity criterion is also unavailable in percent matching. This criterion in terms of percent similarity among bands or markers is due to the need of selectively classifying them. Particularly when the dimensionality of a data vector becomes very huge or the exactly percent matching is impractical to compute, clustering by a relaxed similarity criterion is necessary. When the input data type is quantitative and usually its magnitudes among the variables have unequal influence on the distance between objects, the conventional data standardization used is statistically debatable.
This study discussed all of these questions with major clustering methods and proposed some improved and new algorithms for classic cluster analysis from a practical perspective. In addition, we introduced the novel clustering method that is needed to deal with DNA band or marker data as well as other categorical data. The notation used throughout the paper will remain effective thereafter.

2 Data preparation and methods

The input data type is the first thing to be considered for a cluster analysis to be pragmatically applicable. There is a big data space in the real world that provides quite a few sources of analytical data; typically they fall into two major types: quantitative and categorical. In bioinformatics, for example, gene expression data can be analyzed in a quantitative mode, and some categorical (nominal or ordinal) variables such as binary data are also amenable to numerical analysis. A data set in coordinate form constitutes a data (or pattern) matrix such as DNA microarray-based genes (objects) and experimental conditions (variables) in gene expression profiles [8,9]. Clustering such a matrix may find gene expression patterns and functionally related genes, thereby suggesting the function of currently unknown genes. To perform such a cluster analysis, the similarity or proximity between gene expression profiles has to be properly measured. Usually this is measured in terms of Euclidean distance, as it provides the shortest length between two points in metric space. This distance is regarded as the most natural and also commonly used measure as compared to others.

Since almost all clustering methods operate on Euclidean distance, it is assumed in this study.

For binary data, the absence-presence usually scores 0-1. For ternary band or marker data, it may use “w” to label the “weak” band in a lane from electrophoretic tests and has the trio of states 0, 1, and w. For quaternary band or marker data, it may use one more state label “u” to indicate the “unidentified” band and so forth. The distance or dissimilarity between objects (testing samples) i and j for these nominal data is measured by mismatch score or percent disagreement that is defined as

\[ d(i, j) = 1 - \frac{n_{00} + n_{11}}{n} \] for binary (0-1) data

\[ d(i, j) = 1 - \frac{n_{00} + n_{11} + n_{ww}}{n} \] for ternary (0-1-w) data

\[ d(i, j) = 1 - \frac{n_{00} + n_{11} + n_{ww} + n_{uu}}{n} \] for quaternary (0-1-w-u) data

where 0 ≤ d(i, j) ≤ 1, n_{00}, n_{11}, n_{ww}, or n_{uu} is the number of respective 0-0, 1-1, w-w, and u-u matches, and n is the number of all attributes (dimensions). Notice that the fractional term in each of the expressions is the simple matching coefficient and is complementary to d(i, j); that is, its higher value corresponds to shorter distance or closer relationship. Practically, mismatch scores are better used for clusterings to safely retain the precision of results than the decimal d(i, j), which will be addressed in the next section.

The hierarchical (agglomerative) cluster analysis is a major clustering method applicable to the analytical data. In the hierarchical clustering with data matrix input, the first step is to calculate the distance matrix for all possible pairs of objects. Usually all the elements in this matrix are the squared Euclidean distances between objects but also can be the d(i, j) values computed from the above nominal data.

2.1 Similarity measure conversion

In addition to data matrix (raw data set), distance or dissimilarity and correlation-based similarity matrices also can be imported as special data types for cluster analysis. First of all, we need to clarify the notion of distance- and similarity-type data. That is, the former is the measure of a straight line in Euclidean space, whereas the latter is the measure of proximity or relationship between objects. The similarity-type data can be correlation measures (Pearson, Spearman rank, Kendall τ, etc.), or any other scores that measure the association or pattern resemblance between objects. In general, distances are the best bet to detect differences and correlations are often better to find similarities [10]. Those highly correlated values can be considered to be very similar to each other; as such, increasing similarities translates into decreasing distances or dissimilarities [3,11]. Prior to cluster analysis, all correlation-based similarities must be converted to dissimilarities (distance-type similarity or correlation-based distance). Just as distance is essentially a measure of dissimilarity, dissimilarity matrix here is a synonym for distance matrix in the sense that the minimum rather than the maximum is to be searched for in the matrix for clustering. There are a variety of ways to convert similarity measures, such as taking reciprocals or substracting from the upper bound that is 1 as constrained by a cophenetic matrix [12]. A correlation-based distance or dissimilarity d usually is defined in terms of the correlation (or similarity) coefficient r or s as

\[ d = 1 - r \quad \text{or} \quad d = 1 - s. \] (1)

It should be indicated that r or s be consistently positive. A negative r or s fails to give any information on distance measures due to its reverse direction of association even if all coefficients are negative. Thus, r or s takes on a value that ranges from 0 to 1 and cannot range from 0 to -1. What is the right upper bound that converts from similarity measures to distances while retaining as much of the original information as possible? The use of 1 as the upper bound may not robustly achieve the best correspondence between a similarity matrix and its cophenetic matrix. Table 1 illustrates some extreme similarity coefficients that are closer to 1 and the resulting correspondence is poor due to the loss of significant decimal digits for very small values. If a computing program operates on three or more decimal digits, the extreme r or s (.999) will be converted to d (.001), resulting in the loss of more significant decimal digits and so on. What is worse, the poor precision of those very small values in the cophenetic matrix makes inaccurate the drawing of a dendrogram. Statistically, the correlation coefficient r is the amount that explains the association of one variable with another and is meaningful only in this respect. However, the expression 1-r is not as interpretable and uniquely significant as the r is. From this perspective, there is no constraint that 1 is the only upper bound for conversion purposes. In essence, the criterion to achieve the best correspondence is the upper bound that results in the minimum loss of significant decimal digits for converted values.
From a practical point of view, if \( r_{\text{max}} \) or \( s_{\text{max}} \) is the maximum \( r \) or \( s \) in a correlation or similarity matrix and \( p \) is a percentage, then \( d \) in expression (1) is re-defined by

\[
d = (1 + p) r_{\text{max}} - r \quad \text{or} \quad d = (1 + p) s_{\text{max}} - s \tag{2}
\]

where \( p \) equals 10\%, which is proposed but can be other desired percentage. This expression not only ensures sufficiently significant decimal digits for converted values but it also eliminates data-dependent inconsistency on a proportional basis. Note that, as with expression (1), expression (2) does not satisfy the triangle inequality as well. Indeed, none of the correlation-based distances follows this constraint; this is the general property of the correlation coefficient. Unlike the Euclidean distance that is a true metric and does satisfy the triangle inequality, the correlation-based distance is sometimes thought of as semi-metric. Nevertheless, the similarity measure conversion is not such constrained because it merely turns the coefficients into other corresponding values in their mirror data space. It merely re-quantifies, in other way, the differentiation between distances using the same scale as the correlation coefficients in order to depict them in a dendrogram. What is more, the use of expression (2) leads to an improved and practically acceptable cluster analysis. As shown in Table 1, for instance, the extreme \( r \) or \( s \) (.99) will be converted to \( d .10 \) by (2) rather than to \( d .01 \) by (1). The significant digits of .10 begin from the first decimal place and hence retain two significant digits, whereas that of .01 begins from the second decimal place and retains only one significant digit. Thus, the former contains more of the original information about \( r \) or \( s \) than does the latter. With the upper bound, expression (2) also can handle the conversion of similarity matrices whose elements are not coefficients (\( s \geq 1 \)) such as match scores, proximity counts or similarity ranks.

2.2 Data standardizing transformation

The advantage of correlation measures is that they are generally not influenced by differences in scale between objects. On the other hand, distance measures are significantly affected by differences in scale across variables. From a data matrix, the distance between objects is determined by the sum over all differences of paired variables. If these variables are on different scales, their varying sizes would contribute differently to the distance. To balance the relative importance among the different variables and make multi-dimensional variation comparable, these variables should be transformed such that they are on a common scale. This scaling is necessary because information from each variable needs to be fairly, unbiasedly reflected in determining the distances with no artifacts. That is, without considering the principal components, all variables should contribute equally to the distance between objects. There can be a few transformation schemes: mean, median, maximum, range, and variance scaling that equalize overall variables to achieve unit-measure homogeneity.

Since cluster analysis generally is a nonparametric, descriptive method and has no distributional assumptions unless they are confirmed. In reality, it is difficult to find all the variables following one distribution, especially if the dimensionality is very huge. Therefore, any forced variance scaling (i.e., z-score standardization) based on the multivariate normality is not recommended. From this standpoint, the criterion to judge a sound transformation is how well it retains information of original data. It is something like saying how well it maintains as much “fidelity” of the raw data as it could. Of the five schemes, the mean, median, or maximum scaling imposes the least manipulation on raw data and therefore preserves the most information about it. Dividing a variable by its maximum appears to best scale it to 0-1 range but taking outlier into account would make this scaling less desirable, as the extremely large maximum could scale values to very small ones. Dividing a variable by its median is well-known for tackling outliers but that is better used in that case and is not always the case for all variables. In general, the arithmetic mean is the best data representative (with centrality) because its calculation exploits all values of data and hence reflects all information on a variable. Suppose \( x_{ij} \) is the observation and \( X_{ij} \) is the scaled one for the \( i \)th object and \( j \)th variable. Let \( \bar{x}_j \) be the arithmetic mean, \( \bar{x}_j \) be the median, and \( x_{\text{max},j} \) be the maximum for the \( j \)th variable. These three transformations are defined below by the mean, median, and maximum scaling, respectively:

\[
X_{ij} = \frac{x_{ij}}{\bar{x}_j}, \quad X_{ij} = \frac{x_{ij}}{\bar{x}_j}, \quad X_{ij} = \frac{x_{ij}}{x_{\text{max},j}}.
\]

The difference between the resultant unit mean and unit maximum would be considered to be different consistencies over variables. That is, although the mean scaling does not standardize data to 0-1 range as the maximum scaling does, it still scales them to the consistent differences across variables. This is because all variables use invariably their means to do the scaling. The transformed data are not required virtually to fall within the 0-1 range as long as they are equitable and comparable among variables on a unit-measure basis [13].

The soundness of the mean scaling can be shown by the property that the difference of any two mean-scaled values for variable \( j \) between objects \( i \) and \( k \) is equal to the difference of their non-scaled values divided by the common mean of variable \( j \). Namely,

\[
X_{ij} - X_{ik} = \frac{x_{ij}}{\bar{x}_j} - \frac{x_{ik}}{\bar{x}_j} = \frac{x_{ij} - x_{ik}}{\bar{x}_j}.
\]

This ensures that the contribution of each variable to the distance between objects is adjusted proportionally by its mean, regardless of the differences in scale. In Table 2, the equality (1:1) across variables is achieved at the high end for maximum-scaling and at the middle for mean-scaling. The former has the better inter-variable comparability only on one side near the
high end, while the latter has the better comparability on two sides near the middle. Mean- and maximum-scaling have a shiftable relationship, as their equality points can be shifted. Nevertheless, taking equality at the middle makes transformed values more widely and stably comparable than at one end. That is, with the representativeness (centrality) of the average, the mean scaling is robust to varying transformation, while maximum-scaling is prone to extreme transformation. By this nature, mean-scaled values retain the more characteristics of raw data and have the more equitable influence on the distance. For gene expression data that fail in the test of multivariate normality due to the more attributes in the tissue types, time series, and/or treatment conditions, the mean scaling is more sound than the forced z-score standardization.

Table 2. A demonstration of the terminal equality (one-side comparability across variables) obtained from maximum-scaling and the central equality (two-side comparability across variables) obtained from mean-scaling.

<table>
<thead>
<tr>
<th>O</th>
<th>Raw data matrix</th>
<th>By maximum scaling</th>
<th>By mean scaling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>Variable</td>
<td>Variable</td>
<td>Variable</td>
</tr>
<tr>
<td>1</td>
<td>0.0</td>
<td>5.0</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>1.0</td>
<td>6.0</td>
<td>0.25</td>
</tr>
<tr>
<td>3</td>
<td>2.0</td>
<td>7.0</td>
<td>0.50</td>
</tr>
<tr>
<td>4</td>
<td>3.0</td>
<td>8.0</td>
<td>0.75</td>
</tr>
<tr>
<td>5</td>
<td>4.0</td>
<td>9.0</td>
<td>1.00</td>
</tr>
</tbody>
</table>

2.3 Handling tied data and mergers

The first issue on ties in cluster analysis is the tied objects that could be encountered at level 1 of clustering. Handling this scenario is to merge all tied objects and then treat them as one object. That is, use any of the tied objects as a representative and include it in the participating objects, and the other tied objects are excluded from the remaining cluster analysis. The reason for this elimination is that the tied objects provide nothing more of distinctive data vectors and they are just simple redundancies and should count once in data processing; that is, one copy alone from duplicates suffices. Since the objects excluded fail to contain new meaningful information, they do not make a difference in the subsequent clusterings. The weights they may provide only do not improve but bias the cluster analysis in most cases unless weighted contribu-tions are under study.

The second issue is that the tied mergers may appear at each clustering level. The treatment in this scenario is to accept all tied mergers at each level [14] even if possibly merging all objects/subclusters into a single cluster at very few levels. The reason for this is that the principal aim of cluster analysis is obtaining all proper clusterings; logically, the earlier level they are partitioned at, the more reasonable clusters they are revealed as. If the analyst feels uncomfortable with this way, the first one of tied mergers ordered or sorted in the distance matrix would be taken.

3 Clustering results

3.1 Quantitative hierarchical cluster analysis

For a sound clustering methodology that could exploit as much information as being contained in the data, we focused on the average linkage (UPGMA), centroid method (UPGMC), and Ward’s minimum-variance method [11,15-17]. Their underlying computational algorithms and pragmatic iterative implementations were provided.

In the average linkage, let the capital D stand for a squared Euclidean distance, the lowercase d for a Euclidean distance (the square root of D) between objects/clusters, and the combinatorial pq for the merger of clusters p and q. Suppose i is any other cluster, n is the number of objects in a cluster, k is the kth object in the merger, and l is the lth object in cluster i. It has been verified that only the following two combinatorial forms of computation of D are equivalent to each other:

\[ D(pq, i) = \frac{1}{n_{pq}n_i} \sum_{k \neq i} \sum_{l \neq i} D(k,l) \]  
\[ D(pq, i) = \frac{n_{pq}}{n_{pq} + n_i} D(p,i) + \frac{n_i}{n_{pq} + n_i} D(q,i) \]  

where formula (1) uses all paired objects between pq and i, and the update equation (2) uses the previous-step D for separate p or q with i and hence gains implementation efficiency. They both can take care of either data or distance/similarity matrix input but equation (2) is actually adopted. This method has no issue or ambiguity on its mathematical derivation and practical application, although there is a weighted variant (WPGMA).

In the centroid method, let X be a cluster centroid (or mean vector). It has been verified that only the following two combinatorial forms of computation of D are equivalent to each other:

\[ D(pq, i) = \| \overline{X}_{pq} - \overline{X}_i \|^2 \]  
\[ D(pq, i) = \frac{n_{pq}}{n_{pq} + n_i} D(p,i) + \frac{n_i}{n_{pq} + n_i} D(q,i) \]  

where formula (3) directly employs the centroids to update the distance between the merger and any other cluster, and the update equation (4) is used for efficient implementation. The former can only be used with data matrix input, whereas the latter can deal with both data and distance/similarity matrix input and is adopted in practice.

There are ambiguous ways of calculating a cluster centroid, each resulting in a likely different quantities of distance. In Table 3, the d1pq is calculated by taking the mean over two previous centroids, whereas the d2pq takes the mean over all objects in a merger, for each step. The difference between the d1pq (= 6.0) and d2pq (= 6.3) illustrates this discordance. Only the way that employs all objects in a merger to get d2pq is considered to be rational for a true centroid (a real barycenter). Another validation of this as the right way is that it has been proved by the equivalence of two-form computations for d(pq,
Table 3. The stepwise cluster centroids (CC) and distances (d) calculated by the two previous-step centroids and by all the objects in a merger.

<table>
<thead>
<tr>
<th>#</th>
<th>Partitions of Clusters</th>
<th>CC1p</th>
<th>CC1q</th>
<th>d1pq</th>
<th>CC2p</th>
<th>CC2q</th>
<th>d2pq</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>(1) (2) (3) (4) (5)</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>4</td>
<td>(1, 2) (3) (4) (5)</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td>3</td>
<td>(1, 2) (3) (4, 5)</td>
<td>2.6</td>
<td>3.0</td>
<td>2.8</td>
<td>3.0</td>
<td>2.8</td>
<td>3.0</td>
</tr>
<tr>
<td>2</td>
<td>(1, 2) (3, 4, 5)</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
</tr>
<tr>
<td>1</td>
<td>(1, 2, 3, 4, 5)</td>
<td>10</td>
<td>15</td>
<td>6.0</td>
<td>15</td>
<td>6.3</td>
<td>6.3</td>
</tr>
</tbody>
</table>

In Ward’s method, suppose $i$ is the $i$th cluster for $l$ clusters, $j$ is the $j$th variable for $m$ variables, and $k$ is the $k$th object for $n$ objects in a cluster. The error sum of squares ($ESS$) within the $i$th cluster has been given by

$$ESS_i = \sum_{j=1}^{m} (X_{ij} - \bar{X}_i)^2$$

(5)

From (5), letting $ESS_{pq}$ be $ESS$ within the merger cluster, the minimum increase in $ESS$ ($\Delta ESST_{pq}$) has the following relationships with others:

$$\Delta ESST_{pq} = ESS_{pq} - ESS_p - ESS_q = \frac{n_p n_q}{n_p + n_q} |\bar{X}_p - \bar{X}_q|^2$$

(6)

upon which the metric criterion of distance between clusters is based, as the clustering proceeds.

Let $\Delta ESS_{pq,i}$ be the minimum increase in $ESS$ as clusters $pq$ and $i$ merge. It has been verified that the following two combinatorial forms of computing $\Delta ESS$ are equivalent to each other only if $D'_{pq}$ is a previous-step $\Delta ESS$:

$$\Delta ESS1_{pq,i} = \frac{n_p n_q}{n_p + n_q} |\bar{X}_{pq} - \bar{X}_i|^2 = \frac{n_p n_q}{n_p + n_q} D(pq,i)$$

(7)

$$\Delta ESS2_{pq,i} = \frac{n_p + n_q}{n_p + n_q} D'(p,q) = \frac{n_p + n_q}{n_p + n_q} D'(p,q)$$

(8)

Let $ESS_T$ be the total error sum of squares over all clusters, $ESS'_T$ be the previous-step $ESS_T$ for a cumulative operation, and $ESS(p, i)$ be $ESS_T$ as clusters $pq$ and $i$ merge. It has been verified that only the following three combinatorial forms of computation of $ESS_T$ derived from Ward’s method are equivalent to one another:

$$ESS(p, i) = ESS_T = \sum_{j=1}^{m} \sum_{k=1}^{n} (X_{ijk} - \bar{X}_{ij})^2$$

when $p$ and $q$ merge

(9)

$$ESS(p, i) = ESS'_T = ESS'_T + \Delta ESS1_{pq,i}$$

(10)

$$ESS(p, i) = ESS_T = ESS'_T + \Delta ESS2_{pq,i}$$

(11)

where formula (9) for $ESS$ originates from the primary Ward’s algorithm, the direct update equation (10) employs centroids that is related to $ESS_T$ from (7), and the update equation (11) is used for efficient implementation from (8). The variant (10) can only be used with data matrix input, whereas the variant (11) can deal with both data and distance/similarity matrix input and is adopted in practice.

To implement the Ward’s algorithm, the simple way is to convert all squared Euclidean distances ($D$) values in the distance matrix to the minimum increment $\Delta ESS_{pq}$ at the first step of iteration. First, this conversion is required to use the update equation (8) in which $D'(p, i)$, $D'(q, i)$, and $D'(p, q)$ must be the previous-step $\Delta ESS_{pi}$, $\Delta ESS_{qi}$, and $\Delta ESS_{pq}$, respectively. Second, all elements in the distance matrix must be of the type $\Delta ESS_T$ rather than the type $D$ in order to be comparable and be searchable for the minimum increase in $ESS$. For this reason, such elements must be either all $\Delta ESS_T$ values or all $D$ values and may not be of discordant type. To use the $\Delta ESS_T$ distance matrix, its element simply takes half of the $D$ value, as $n_p n_q/(n_p + n_q)$ becomes $1/2$ when any two objects merge initially ($n_{p} = 1, n_{q} = 1$). However, this way need to convert all $D$ values in the distance matrix and is thought to have the extra cost of computation.

To cope with this problem and keep using the $D$ distance matrix without having to use the $\Delta ESS_T$ distance matrix, the solution is to utilize the update equation (4) in the centroid method afore-mentioned. Since calculating a cluster centroid is this equation (13) provides a solution to using equation (4) instead of classic (8) and preserves all the desired properties that it remains capable of updating a distance matrix, that it retains direct use of squared Euclidean distances without having to convert them otherwise, and that it can handle both data and distance/similarity matrix input.

There is an inconsistency with the criterion of distance between clusters given by $ESS_T$, by $ESS_{pq}$ plus the previous-step value, or by $\Delta ESS$. Because $ESS_{pq}$ contains $ESS_p$ and $ESS_q$, it is not a net increase in $ESS$ due to the fusion of clusters $p$ and $q$. From the stepwise increment of $ESS$ (Table 4), one can see that the distance criterion should be given by $\Delta ESS_{pq}$--based $ESS_T$ rather than $ESS_{pq}$--based $ESS_T$. Another validation of $ESS_T$ as the distance criterion is that it has been proved by the
equivalence of three-form computations for $ESS_T$, while using $ESS_{pq}$ cannot yield such equivalence. It is unreasonable for $\Delta ESS$ per se to be the distance criterion in that the primary Ward’s algorithm takes care of $ESS_T$, not $\Delta ESS$ produced by the merger alone. $ESS_T$ measures overall distance, while $\Delta ESS$ measures inter-centroid distance as it is closely related to $D$ values as revealed by formula (7). The former gives non-centroid distance by accumulation (monotonic increase), while the latter functions more likely as a centroid method. To distinguish Ward’s method (with monotonicity) from a centroid-like method (without monotonicity), only $ESS_T$ is treated as the real Ward’s criterion of distance between clusters.

Table 4. The stepwise error sums of squares (ESSs) for the merger $ESS_{pq}$, the incremental $\Delta ESS_{pq}$, and the respective total $ESS_T$s.

<table>
<thead>
<tr>
<th>#</th>
<th>Partitions with Minimum $ESS_T$</th>
<th>$ESS_{pq}$</th>
<th>$\Delta ESS_{pq}$</th>
<th>$ESS_T$</th>
<th>$ESS_T$</th>
<th>$ESS_T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>(1) (2) (5) (7) (9) (10)</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>5</td>
<td>(1, 2) (5) (7) (9) (10)</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>4</td>
<td>(1, 2) (5) (7, 9, 10)</td>
<td>0.50</td>
<td>0.50</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>3</td>
<td>(1, 2) (5, 7, 9, 10)</td>
<td>2.00</td>
<td>2.00</td>
<td>3.00</td>
<td>4.00</td>
<td>4.00</td>
</tr>
<tr>
<td>2</td>
<td>(1, 2) (5, 7, 9, 10)</td>
<td>14.75</td>
<td>12.25</td>
<td>17.75</td>
<td>15.25</td>
<td>24.50</td>
</tr>
<tr>
<td>1</td>
<td>(1, 2, 5, 7, 9, 10)</td>
<td>67.33</td>
<td>52.08</td>
<td>67.33</td>
<td>67.33</td>
<td>104.17</td>
</tr>
</tbody>
</table>

Whatever forms these formula or update equations may take, they are the sums of squared deviations from mean vectors. Thus it makes sense that they should be restored to original metrics (i.e., square roots) after being subject to squaring. This not only provides a real measure in Euclidean space but also makes it consistent and comparable with those obtained by other clustering linkages. Moreover, the usage of Euclidean distance improves cluster representation as well in a dendrogram by scaling down the longer distances between clusters and scaling up the shorter ones (Fig. 2). This is because the distance could be inflated by squaring a greater-than-one node value in a dendrogram as shown in Figure 1. Particularly, the square root of decimal figures gives the larger values and hence removes the distortion caused by squaring the decimal distances. Therefore, the practical criterion of distance or similarity between clusterings should be given by Euclidean distance rather than by squared one. Namely,

$$d(pg, i) = \sqrt{ESS(pg, i)} = \sqrt{ESS_T}.$$  

### 3.2 Categorical hierarchical cluster analysis

Categorical variables such as the foregoing nominal attributes (dichotomous or multi-state) can be subjected to numerical analysis if they turn into a dissimilarity matrix by mismatch score or percent disagreement. The clustering methods above are applied to such data as well. Here we introduced a new hierarchical clustering method that allows for analysis of DNA band, marker, or other nominal data. With such clusterings, data need not turn into a dissimilarity matrix. This method is called the percent similarity cluster analysis (PSCA) and used primarily in such categorical settings.

This removes the distortion of the tree diagram caused by squaring operation on the distances as compared to Figure 1.

DNA band or marker data from electrophoresis consist of biological items (objects) each of which has a band vector whose elements (band values taking 0, 1, w, or u) are nominal attributes in a data matrix. They are analyzed through comparison of each pair of corresponding band values between items and those items showing a certain or higher percent band similarity are clustered. It is utilized to discover those banding patterns where identical, similar, and distinct bands are identified. Further, they are used for a comparative study of molecular identities such as the biological variety appraisal by DNA fingerprint clustering.

The underlying algorithm for PSCA is: From the first item, make all possible comparisons among items until get the first cluster in which members meet a mutual similarity criterion. Then from the second item (if clustered go to the next), make all possible comparisons among the remaining items until get the second cluster with the same mutual similarity criterion met. Iterate it until get the last-run cluster at the first level of clustering. Afterward, all the analogous comparisons for the next level are made against the relaxed similarity criterion and based on either clusters or singletons. The similarity criteria are from 100% down to 10%, relaxed by 10 (the default but can be

---

Fig. 1. The tree branch distance could be elongated by the squared greater-than-one node value (67.33) and shortened by the squared decimal node value (0.5). This causes the distorted graphic representation of clusters in the dendrogram.

Fig. 2. The square root of the decimal distance gives a larger node value (0.71) and the square root of the greater-than-one distance gives a smaller node value (8.21) from Fig. 1.
specified otherwise); so there are up to 10 (or other) levels of hierarchy. At level 1, each item is a cluster center for comparison. After each level, the number of cluster centers may be reduced due to the merger of clusters. For a merger, any element of its cluster center that does not match the corresponding band value of any member is marked. Clustering proceeds with comparison of the elements of cluster centers, which become less matchable after each fusion, until the last level (0% similarity) is reached. The tied items will not get weighted, as they fail to offer new information on the identity of an item.

Figure 3 illustrates a local portion of complex results of PSCA using the 10-band-per-item data from electrophoresis of 8 items with RT-PCR in our DNA fingerprinting.

![Figure 3](image)

Fig. 3. An illustration of part of the complex results from the percent similarity cluster analysis (PSCA) in the dendrogram. PK4, PK5 and PK6 are identical. PB3 through PA3 are 80% similar. PB5 is 60% similar to PA2. Entirely, all the items are 30% similar to one another.

### 4 Applicability and discussion

In the PSCA method, it is remarkable that the clustering result differs from the one by a classic clustering method that merges traditionally paired objects/clusters based on the minimum dissimilarity criterion. With PSCA, it merges multiple items or subclusters and only those that are consistently similar to one another are clustered based on the criterion that is alterable by level. In Figure 4, for instance, the bands for item 1 have a 90% similarity with the bands for items 2, 3, 4, and 5. However, the continuing pairwise comparisons from item 2 through item 5 turn out that item 3 is not consistently similar to all other items. So it will be excluded from this group of similar items and only items 1, 2, 4, and 5 are clustered. That is, only those items that have shown criterion-met mutual similarities are deemed to form a cluster. This clustering is applied to discovering banding patterns and interpreting their relations before we get the insight by other proofs. The labels “w” and “u” in the multi-state band or marker data are thus clustered for the convenience of later parallel pattern recognition, image processing, and data mining.

Cluster analysis is an unsupervised learning technique that could lead to different results via numerous approaches. Therefore, the choice of the right methodology is critical to the researcher. For hierarchical clustering, only the average, centroid, and Ward’s linkage are focused in that they safely result in no undesirable cluster properties generated by the use of little or one-sided data information. Taking for example the chaining effect produced by single linkage, it is often criticized because objects being similar at one end of a cluster may be markedly dissimilar at the opposite ends. Likewise complete linkage also has no control of the resulting cluster shape and doesn’t employ information from all member objects and tends to produce chaining clusters in the tree. Since clustering is a kind of collective or group behavior, not an individual behavior, a single object is not informative enough to reflect the entire cluster structure. Therefore, a between-cluster clusterability determined by a single object is not sound and reasonable, which makes its results difficult to interpret. These two methods usually perform well in cases when the objects actually form naturally distinct clumps in the data space. Instead of relying on extreme values as in single or complete linkage, one uses the average, centroid, and minimum variance to link clusters, which not only gains robustness but also warrants reliability. These linkages take the entire cluster structure into account and employ information from all member objects to determine a between-cluster clusterability; so its results are reasonable and interpretable. Generally such methods are less sensitive to noisy data, and outliers are not given any special favour in the cluster decision. Their resulting clusters have compact, spherical shapes where all members of a cluster tend to be tightly bound together. It is nevertheless advisable to acquire cluster information that is not based upon a particular algorithm and that should be objective-dependent. Indeed, there is no such thing as a single correct or desirable classification. To be safe in
cluster analysis, there is no better practice than one that removes outliers and has missing values retrieved.

The average linkage (UPGMA) is practicable and superior to the weighted variant (WPGMA) in that all objects receive equal weights in the computation, which conforms to the equal contribution of each object to the distance. The centroid linkage (UPGMC) is as preferable to the median linkage (WPGMC) as UPGMA is to WPGMA. The median linkage does not really imply an outlier-proof algorithm as it suggests literally, and is better used when unequal cluster sizes (the numbers of objects) are treated as biased.

Since centroid clustering is not monotonic it may produce reversals of the levels in the dendrogram. This reversal is regarded as a violation of the ultrametric property but it is true only with respect to graphic representation of clusters and is sensible to numeric operation on centroids. That is, the length of a branch of a tree corresponds to the inter-centroid distance, a shorter length or a reversal can be generated by the shorter distance from level to level. It is possible for the inter-centroid distance to be non-monotonic as a result of subtracting operation between such centroids even if they are monotonically increasing. Therefore, the result from centroid clustering is still interpretable, regardless of reversals.

The PSCA method can be used to gather information on mutual similarities or interrelations of DNA bands or marker data and the like, and to discover banding patterns in data structures and layouts from electrophoresis. The results are most applicable to identifying the most likely genotype of an unknown organism via DNA fingerprints. It can be employed as well in bioinformatics studies and to discriminate between items (e.g., samples and specimens) that produce bands or markers. Moreover, the patterns it reveals are useful in exploring a potential biological relatedness or affinity among the items (e.g., species) [19].

All improved algorithms, solutions and novel methods for cluster analysis covered in this paper have been implemented in our earlier software BioCluster for Windows and the latest ParCluster2.0 (to be separately published). They are available

---

**Fig. 5.** A demonstration of text-based special clusterings in addition to graph-based special clusterings, to the right of which is a listing of DNA band data that is used to check with the corresponding entries of biological items/varieties (taxa) on the left side. With this textual tree view, each item on the left side corresponds to its 10 bands on the right side for a convenience of studies and manipulation. The letter “U” in the band data listing stands for an “unidentified” band. For a 0-1 band data system, another letter that may appear is “W” to label the “weak” band in a lane from electrophoretic testing. A band data range can be extended to any other data such as amino acid sequences, etc.
upon request (rli@alumni.lsu.edu) or from some web sites. In ParCluster v.2.0, special clusterings in addition to the PSCA method also are available (Fig. 5).

5 References