SUPPLEMENTARY APPENDIX

Table of Contents

SUPPLEMENTARY METHODS

Cluster and Classification Analysis

A combination of techniques of cluster and classification analyses have been used to achieve the goal of our study.[1]

Cluster analysis is a data-reduction technique designed to uncover subgroups of observations within a dataset. It allows you to reduce a large number of observations to a much smaller number of clusters or types. A cluster is defined as a group of observations that are more similar to each other than they are to the observations in other groups. The two most popular clustering approaches are hierarchical agglomerative clustering and partitioning clustering. In agglomerative hierarchical clustering, each observation starts as its own cluster. Clusters are then combined, two at a time, until all clusters are merged into a single cluster.

In the partitioning approach, you specify K: the number of clusters sought. Observations are then randomly divided into K groups and reshuffled to form cohesive clusters. Within each of these broad approaches, there are many clustering algorithms to choose from. For hierarchical clustering, the most popular are single linkage, complete linkage, average linkage, centroid, and Ward's method. For partitioning, the two most popular are k-means (only for continuous variables) and partitioning around medoids (PAM) (for mixed variables).

An effective cluster analysis is a multistep process with numerous decision points. Each decision can affect the quality and usefulness of the results.

This section describes the 11 typical steps in a comprehensive cluster analysis:

1. *Choose appropriate attributes*. The first (and perhaps most important) step is to select variables that you feel may be important for identifying and understanding differences among groups of observations within the data. A sophisticated cluster analysis cannot compensate for a poor choice of variables.
2. *Scale the data*. If the variables in the analysis vary in range, the variables with the largest range will have the greatest impact on the results. This is often undesirable, and analysts scale the data before continuing. The most popular approach is to standardize each variable to a mean of 0 and a standard deviation of 1.
3. *Screen for outliers*. Many clustering techniques are sensitive to outliers, distorting the cluster solutions obtained. You can screen for (and remove) univariate outliers. An alternative is to use a clustering method that is robust to the presence of outliers (Partitioning around medoids is an example of the approach).
4. *Calculate distances*. Although clustering algorithms vary widely, they typically require a measure of the distance among the entities to be clustered.
5. *Select a clustering algorithm*. Next, you select a method of clustering the data. Hierarchical clustering is useful for smaller problems (say, 150 observations or less) and where a nested hierarchy of groupings is desired. The partitioning method can handle much larger problems. Once you have chosen the

hierarchical or partitioning approach, you must select a specific clustering algorithm. Each has advantages and disadvantages. The most popular methods are described in Lesmeister C[1] 2017. You may wish to try more than one algorithm to see how robust the results are to the choice of methods.

6. *Obtain one or more cluster solutions*. This step uses the method(s) selected in step 5.
7. *Determine the number of clusters present*. In order to obtain a final cluster solution, you must decide how many clusters are present in the data. This is a thorny problem, and many approaches have been proposed. It usually involves extracting various numbers of clusters (say, 2 to K) and comparing the quality of the solutions. There are functions (NbClust() of R package NbClust) that provides different indices to help you make this decision.
8. *Obtain a final clustering solution*. Once the number of clusters has been determined, a final clustering is performed to extract that number of subgroups.
9. *Visualize the results*. Visualization can help you determine the meaning and usefulness of the cluster solution. The results of a hierarchical clustering are usually presented as a dendrogram. Partitioning results are typically visualized using a bivariate cluster plot.
10. *Interpret the clusters*. Once a cluster solution has been obtained, you must interpret (and possibly name) the clusters. What do the observations in a cluster have in common? How do they differ from the observations in other clusters? This step is typically accomplished by obtaining summary statistics for each variable by cluster. For continuous data, the mean or median for each variable within each cluster is calculated. For mixed data (data that contain categorical variables), the summary statistics will also include modes or category distributions.
11. *Validate the results*. Validating the cluster solution involves asking the question, Are these groupings in some sense real, and not a manifestation of unique aspects of this dataset or statistical technique? If a different cluster method or different sample is employed, would the same clusters be obtained?

Classification analysis is used when there is the need to predict a categorical outcome from a set of predictor variables. The goal is to find an accurate method of classifying new cases into one of the two groups. So, given a predictor of variables x, and a categorical response variable y, it is the need to build a model for:

- Predicting the value of y for a new value of x;
- Understanding the relationship between x and y.

Classification Methods are Linear discriminant analysis, Logistic regression, Nonparametric methods as Nearest neighbor classifiers and classification trees, Machine learning methods as Bagging, Support vector machines (SVM) and Random Forest (RF).

In our analysis, PAM for clustering and RF both for clustering and for classification were used.

NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set

Most of the clustering algorithms depend on some assumptions in order to define the subgroups present in a data set. As a consequence, the resulting clustering scheme requires some sort of evaluation as regards its validity. The evaluation procedure has to tackle difficult problems such as the quality of clusters, the degree with which a clustering scheme fits a specific data set and the optimal number of clusters in a partitioning. In the literature, a wide variety of indices have been proposed to find the optimal number of clusters in a partitioning of a data set during the clustering process. However, for most of indices proposed in the literature, programs are unavailable to test these indices and compare them.

The R package NbClust[2] has been developed for that purpose. It provides 30 indices which determine the number of clusters in a data set and it offers also the best clustering scheme from different results to the user. In addition, it provides a function to perform k-means and hierarchical clustering with different distance measures and aggregation methods. Any combination of validation indices and clustering methods can be requested in a single function call. This enables the user to simultaneously evaluate several clustering schemes while varying the number of clusters, to help determining the most appropriate number of clusters for the data set of interest.

Distance measures available in NbClust package are: Euclidean distance, maximum distance, Manhattan distance, Canberra distance, binary distance and Minkowski distance. Several agglomeration methods are also provided by the NbClust package, namely: Ward[3], single[4-5], complete[6], average[5], McQuitty[7], median[8] and centroid.[5] All of these methods and distance measures are described in detail in Charrad M et al.[2]

One important benefit of NbClust is that the user can simultaneously select multiple indices and number of clusters in a single function call. Moreover, it offers the user the best clustering scheme from different results. The package is available from the Comprehensive R Archive Network (CRAN) at http://CRAN.R-project.org/package=NbClust.[2]

Partitioning Around Medoids

PAM method was used because it has several advantages: it's not sensitive to outliers as methods based on means, can handle large data set, and it can accommodate mixed data types when necessary. In PAM method, each cluster is identified by its most representative observation (called a medoid).

Furthermore, PAM method can be based on any distance measure; in our study a RF distance measure (dissimilarity matrix), was used. This distance measure will be discussed in the following Proximities Section.

The PAM algorithm is as follows:

1. Randomly select K observations (call each a medoid).
2. Calculate the distance/dissimilarity of every observation to each medoid.
3. Assign each observation to its closest medoid.
4. Calculate the sum of the distances of each observation from its medoid (total cost).
5. Select a point that is not a medoid and swap it with its medoid.
6. Reassign every point to its closest medoid.

7. Calculate the total cost.
8. If this total cost is smaller, keep the new point as a medoid.
9. Repeat steps 5-8 until the medoids don't change.

A medoid is an observation of a cluster that minimizes the dissimilarity between the other observations in that cluster.

Random Forest

A random forest is an ensemble learning approaches for supervised and unsupervised learning. Supervised RF allow for classification, while unsupervised RF allow for clustering. Multiple predictive models are developed, and the results are aggregated to improve classification rates.[1] The algorithm for a RF involves sampling cases and variables to create a large number of decision trees. Each case is classified by each decision tree.

Supervised Learning

Supervised RF is a classification analysis where the outcome needs to be specified. Assume that N is the number of cases in the training sample and M is the number of variables. Then the algorithm is as follows:

1. Grow a large number of decision trees by sampling N cases with replacement from the training set;
2. Sample m < M variables at each node. These variables are considered candidates for splitting in that node. The value m is the same for each node;
3. Grow each tree fully without pruning (the minimum node size is set to 1);
4. Terminal nodes are assigned to a class based on the mode of cases in that node;
5. Classify new cases by sending them down all the trees and taking a vote-majority rules.

An out-of-bag (OOB) error estimate is obtained by classifying the cases that aren't selected when building a tree, using that tree. This is an advantage when a validation sample is unavailable. Finally, the validation sample is classified using the RF and the predictive accuracy is calculated. Random forests tend to be very accurate compared with other classification methods. Additionally, they can handle large problems (many observations and variables), can handle large amounts of missing data, and can handle cases in which the number of variables is much greater than the number of observations. The provision of OOB error rates and measures of variable importance are also significant advantages. A significant disadvantage is that it's difficult to understand the classification rules (there are 500 trees) and communicate them to others.

Unsupervised Learning

In unsupervised learning the data consist of a set of x-vectors of the same dimension with no class labels or response variables. There is no figure of merit to optimize, leaving the field open to ambiguous conclusions. The usual goal is to cluster the data - to see if it falls into different piles, each of which can be assigned some meaning.

The approach in RF is to consider the original data as class 1 and to create a synthetic second class of the same size that will be labeled as class 2. The synthetic second class is created by sampling at random from the univariate distributions of the original data. Here is how a single member of class two is created - the first coordinate is sampled from the N values x(1; n). The second coordinate is sampled independently from the N values x(2; n), and so forth.

Thus, class two has the distribution of independent random variables, each one having the same univariate distribution as the corresponding variable in the original data. Class 2 thus destroys the dependency structure in the original data. But now, there are two classes and this artificial two-class problem can be run through random forests. This allows all of the random forests options to be applied to the original unlabeled data set. If the OOB misclassification rate in the two-class problem is, say, 40% or more, it implies that the x -variables look too much like independent variables to random forests. The dependencies do not have a large role and not much discrimination is taking place. If the misclassification rate is lower, then the dependencies are playing an important role.

Formulating it as a two-class problem has a number of payoffs. Missing values can be replaced effectively. Outliers can be found. Variable importance can be measured. Scaling can be performed (in this case, if the original data had labels, the unsupervised scaling often retains the structure of the original scaling). But the most important payoff is the possibility of clustering.

Proximities

Proximity measure is a pairwise measure between all the observations. If two observations end up in the same terminal node of a tree, their proximity score is equal to one, otherwise zero. At the termination of the RF run, the proximity scores for the observed data are normalized by dividing by the total number of trees. The resulting NxN matrix contains scores between zero and one, naturally with the diagonal values all being one. That's all there is to it.

Methods for Handling Missing Data

Because of missing data are a consistent number in our data set, imputation methods have been performed. In particular, in RF the missing values were replaced in the following way: if the m^{th} variable is not categorical, the method computes the median of all values of this variable in class j, then it uses this value to replace all missing values of the m^{th} variable in class j. If the m^{th} variable is categorical, the replacement is the most frequent non-missing value in class j.

References

1. Lesmeister, C. Mastering machine learning with R: Advanced prediction, algorithms, and learning methods with R 2017. second ed. Birmingham, UK: Packt Publishing.
2. Charrad M, Ghazzali N, Boiteau V, Niknafs A. NbClust: An R Package for determining the relevant number of clusters in a data set. J Stat Softw 2014;61:1-36.

3. Ward JH. Hierarchical grouping to optimize an objective function. J Am Stat Assoc 1963;58:236-44.
4. Florek K, Lukaszewicz J, Perkal J, Zubrzycki S. \Sur la Liaison et la Division des Points d'un Ensemble Fini. Colloquium Mathematicae 1951;2:282-85.
5. Sokal R, Michener C. A statistical method for evaluating systematic relationships. University of Kansas Science Bulletin 1958;38:1409-38.
6. Sorensen TA. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. Biologiske Skrifter 1948;5:1-34.
7. McQuitty LL. Similarity analysis by reciprocal pairs for discrete and continuous data. Educ Psychol Meas 1966;26:825-31.
8. Gower JC. A comparison of some methods of cluster analysis. Biometrics 1967;23:623-37.

Supplementary Table S1. Distribution of patients in the thalassemia International Health Repository.

Country	Number of patients	Original Classification	
		NTDT	TDT
Egypt	930	28	902
Iran	1952	710	1242
Italy	4349	1054	3295
Oman	224	43	181
Pakistan	293	142	151
Saudi Arabia	30	1	29
USA	132	49	83
Total	**7910**	**2027**	**5883**

TDT, transfusion-dependent thalassemia; NTDT, non-transfusion-dependent thalassemia.

Supplementary Table S2. Results of 'NbClust' Package using different types of clustering methods.

Method	Number of statistical indexes	Best number of clusters
Average	8	2
	9	3
	3	4
	1	5
	2	10
k-means	6	2
	7	3
	1	4
	3	5
	3	6
	3	7
	1	9
Ward.D	7	2
	7	3
	4	4
	2	5
	2	6