Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *http://www.jstatsoft.org/*

clamix: An R Package for Clustering Histogram Symbolic Data

Vladimir Batagelj
University of Ljubljana

Nataša Kejžar
University of Ljubljana

Simona Korenjak-Černe
University of Ljubljana

Abstract

Symbolic Data Analysis is based on special descriptions of data — symbolic objects. A special kind of symbolic object is representation with distributions (bar plots or histograms). This representation enables us to consider and include the variables of all types in the clustering process at the same time. We present the R package **clamix** that implements two clustering methods based on the symbolic object descriptions: the adapted leaders method and the adapted agglomerative hierarchical clustering Ward's method. Both methods are compatible — they can be viewed as the two approaches for solving the same clustering optimization problem. In the obtained clustering to each cluster is assigned its leader. The descriptions of the leaders offer simple interpretation of the cluster characteristics. The leaders method enables us to efficiently solve clustering problems with large number of units; while the agglomerative method is applied on the obtained leaders and enables us to decide upon the right number of clusters on the basis of the corresponding dendrogram.

Keywords: symbolic data, clustering, leaders method, hierarchical clustering, Ward's method, R.

1. Introduction

In the usual vector description of a data unit each component of the vector corresponds to a descriptor – *variable*. Variables can be measured in different scales: nominal, ordinal, numerical. When all variables are not of the same type we are dealing with so called *mixed* data.

One of the possible solutions of the *clustering problem with mixed units* is to choose a uniform description for such units. We selected the description of units with distributions (histogram interval-valued representation for numerical variables and histogram multi-valued representation for the rest (Billard and Diday (2006))). Such a description has many advantages:

- it enables us to deal with all types of variables,
- can be used to reduce large data set,
- and also preserves more detailed information about the data than the point values.

The description with distributions is a special kind of *symbolic object*, Billard and Diday (2006).

In this paper we present the implementation of the non-hierarchical (leaders) and hierarchical (Ward's) clustering procedures that were adapted in order to allow to consider units as symbolic objects (i.e. variables described as discrete distributions). Detailed proofs about the accurateness of the adapted procedures can be found in the paper Batagelj, Kejžar, and Korenjak-Černe (2010).

The package **clamix** is implemented using R system for statistical computing (R Development Core Team (2010)). It is available from R-Forge `https://r-forge.r-project.org/projects/clamix/`. Help with installing R package can be found by typing `help("install.packages")`. Upon successfully downloading the package, it can be loaded by typing `library("clamix")`. The user may then type `help(package = "clamix")` to see a list of available functions.

2. Symbolic objects

Let $\mathbf{U}$ be a set of units X described with one or more variables. For the description based on distributions the domain of each variable $V^j (j = 1, \cdots, m)$ is partitioned into k_j subsets $\{V_i^j,\ i = 1, \ldots k_j\}$.

For a cluster C (cluster can consist of one unit X) we denote with $f(i, C; V^j)$ the *frequency* and with

$$\pi(i, C; V^j) = \frac{f(i, C; V^j)}{\mathrm{card}(C; V^j)} \tag{1}$$

the *relative frequency* of the values of variable V^j in the i-th subset V_i^j.

For all variables $V^j (j = 1, \cdots, m)$ holds

$$\sum_{i=1}^{k_j} \pi(i, C; V^j) = 1.$$

The description $C(V^j)$ of the cluster C for variable V^j is a distribution, i.e. the vector of the frequencies on the subsets V_i^j $(i = 1, \cdots, k_j)$.

The cluster C is described with the vector $\boldsymbol{C}$ of the distributions:

$$\boldsymbol{C} = [C(V^1), \ldots, C(V^m)], \tag{2}$$

$$C(V^j) = [\pi(1, C; V^j), \ldots, \pi(k_j, C; V^j)]. \tag{3}$$

Such a description has the following important properties:

- it requires a *fixed space* per variable;

- it is *compatible with merging* of disjoint clusters – knowing the description of clusters C_1 and C_2, $C_1 \cap C_2 = \emptyset$, we can, without additional information, produce the description of their union

$$\pi(i, C_1 \cup C_2; V) = \frac{\text{card}(C_1; V)\, \pi(i, C_1; V) + \text{card}(C_2; V)\, \pi(i, C_2; V)}{\text{card}(C_1 \cup C_2; V)}; \tag{4}$$

- it produces an *uniform description* for all the types of descriptors.

2.1. Example

When investigating world population by countries one usually looks at population pyramids for countries. Each country represents a statistical unit and its population pyramid consists of the following information:

- intervals for age categories
- for each interval the number of men and women are known.

The symbolic object of a unit can then be considered as the unit with two (men, women) variables with histogram representation (year 2006):

$$country_{\text{SLO}} = \Big[\underset{\uparrow \text{ men}}{[204725, 298586, 304629, 170812]}, \quad \underset{\uparrow \text{ women}}{[193941, 288070, 301966, 247618]} \Big]$$

in the case when intervals for age categories are respectively $[0, 19], [20, 39], [40, 59], [60, +)$. Therefore 204,725 males and 193,941 females younger than 20 years lived in Slovenia in the year 2006. As for the last element in each histogram, 170,812 males and 247,618 females of age 60 or older lived in Slovenia in that year.

3. Clustering

3.1. Clustering as optimization problem

The leaders method is a local optimization method and it solves the following optimization problem: find a clustering $\mathbf{C}^*$ in the set of feasible clusterings Φ for which

$$P(\mathbf{C}^*) = \min_{\mathbf{C} \in \Phi} P(\mathbf{C}) \tag{5}$$

with the criterion function

$$P(\mathbf{C}) = \sum_{C \in \mathbf{C}} p(C) \qquad \text{where} \qquad p(C) = p(C, L_C) = \sum_{X \in C} d(X, L_C). \tag{6}$$

Here $d(.,.)$ denotes the dissimilarity measure between a cluster unit and its optimal leader L_C.

The set of feasible clusterings Φ is a *set of partitions into k clusters* of the finite set of units $\mathbf{U}$. The initial clustering can be obtained randomly with selected number of clusters k or can

be determined from the units by selection of the maximal allowed dissimilarity between the unit and the nearest leader.

The optimal leader L_C of the cluster C has to satisfy the following relation

$$L_C = \text{argmin}_{L \in \Psi} p(C, L), \tag{7}$$

where Ψ is the set of feasible representations of the clusters. In our approach, the leader L (a representative element) of the cluster C is also described as a vector of distributions

$$\begin{aligned} \boldsymbol{L} &= [L(V^1), \ldots, L(V^m)], & (8)\\ L(V^j) &= [\lambda(1, L; V^j), \ldots, \lambda(k_j, L; V^j)]. & (9) \end{aligned}$$

The dissimilarity measure between two units (or a unit and a leader) in the classical leaders method is defined as the sum of the dissimilarities between them on each variable V

$$d(X, L) = \sum_{j=1}^{k} d_0(X, L; V^j) \tag{10}$$

where

$$d_0(X, L; V) = \Big(\pi(X; V) - \lambda(L; V)\Big)^2 \tag{11}$$

This is a classical Euclidean distance measure from which we know of the following theorem:

Theorem 1 *(**?** old??) For the criterion function P with dissimilarity $d_0(.,.)$ the optimal leaders are uniquely determined with the* ***averages of of relative frequencies***

$$\lambda(i, L; V) = \frac{1}{\text{card}(C)} \sum_{X \in C} \pi(i, X; V). \tag{12}$$

$\lambda(L; V)$ is also a distribution.

The problem with this optimal distribution $\lambda(L; V)$ is that it is not also the distribution of the corresponding cluster. The leader represents a kind of *average shape of joint distributions.*

The new dissimilarity measure between two units (or a unit and a leader) is defined in order to solve for this problem. It is defined as the *weighted sum of the dissimilarities* between them on each variable V

$$d(X, L) = \sum_{j=1}^{k} \alpha_j d(X, L; V^j), \quad \alpha_j \geq 0, \quad \sum_j \alpha_j = 1 \tag{13}$$

where

$$d(X, L; V) = w_{(C;V)} \cdot \Big(\pi(X; V) - \lambda(L; V)\Big)^2. \tag{14}$$

This is a *generalized Euclidean distance measure.* The weight $w_{(C;V)} = \text{card}(C; V)$ accounts for the size of each variable. If $w_{(C;V)} = 1$ one gets the classical dissimilarity $d_0(.,.)$.

Theorem 2 *(Batagelj et al. (2010)) For the criterion function P with dissimilarity $d(.,.)$ the optimal leaders are uniquely determined with the* ***averages of frequencies***

$$\lambda(i, L; V) = \frac{1}{\mathrm{card}(i, C; V)} \sum_{X \in C} f(i, X; V). \tag{15}$$

$\lambda(L; V)$ is also a distribution.

3.2. Methods

Classical clustering methods face two problems: hierarchical methods are limited to small number of units; and nonhierarchical methods are mostly limited to units described with numbers and use for the cluster representation only one value (usually the center of the cluster). Focusing on the problem of clustering large data sets, we adapted the well known leaders method for the selected uniform representations of units and clusters. To reveal the internal structure of the reduced data set we adapted also the agglomerative hierarchical clustering method. We proved that for the selected dissimilarity, the adapted method is an adapted version of the Ward's hierarchical clustering method (Ward (1963)).

Leaders method and Ward's method are compatible — they are solving the same optimization problem.

Leaders method

One of the most popular clustering methods for large data sets is k-means method, which is a special version of the leaders method (Hartigan (1975), page 74). k-means method can be applied only on numerical variables. The leaders method as a variant of the dynamic clustering method (Diday (1979)) can be described with the following procedure:

determine an initial clustering
repeat
  determine leaders of the clusters in the current clustering
  assign each unit to the nearest new leader – producing a new clustering
until the leaders stabilize.

The initial clustering can be obtained randomly with selected number of clusters k or can be determined from the units by selection of the maximal allowed dissimilarity between the unit and the nearest leader.

The optimal leaders of the clusters are the corresponding weighted averages (centers of gravity) of the units from the cluster. The optimal clustering given the leaders $\mathbf{L}$ is determined by assigning each unit X to the closest leader $L_i \in \mathbf{L}$ in terms of weighted dissimilarity (see eq. 13).

Hierarchical clustering method

The standard agglomerative hierarchical clustering method is described with the following procedure:

each unit forms a cluster: $\mathbf{C}_n = \{\{X\}: X \in \mathbf{U}\}$;
they are at level 0: $h(\{X\}) = 0,\ X \in \mathbf{U}$;
for $k = n - 1$ **to** 1 **do**
  determine the closest pair of clusters
    $(p, q) = \text{argmin}_{i,j:i\neq j}\{D(C_i, C_j): C_i, C_j \in \mathbf{C}_{k+1}\}$;
  join the closest pair of clusters $C_{(pq)} = C_p \cup C_q$
    $\mathbf{C}_k = (\mathbf{C}_{k+1} \setminus \{C_p, C_q\}) \cup \{C_{(pq)}\}$;
    $h(C_{(pq)}) = D(C_p, C_q)$
    determine the dissimilarities $D(C_{(pq)}, C_s), C_s \in \mathbf{C}_k$
endfor

$\mathbf{C}_k$ is a partition of the finite set of units $\mathbf{U}$ into k clusters. The level $h(C)$ of the cluster $C_{(pq)} = C_p \cup C_q$ is determined by the dissimilarity between the joint clusters C_p and C_q by $h(C_{(pq)}) = D(C_p, C_q)$.

For the definition of dissimilarity between clusters we assume that

$$p(C_p \cup C_q) = p(C_p) + p(C_q) + D(C_p, C_q). \tag{16}$$

Theorem 3 *(Batagelj et al. (2010)) For the criterion function P and dissimilarity measure $d(.,.)$ the dissimilarity $D(C_p, C_q)$ can be calculated using the generalized Ward's relation:*

$$D(C_p, C_q) = \sum_j \alpha_j \frac{A_j \cdot B_j}{A_j + B_j}\Big(\lambda_p(i, L_p; V^j) - \lambda_q(i, L : q; V^j)\Big)^2$$

where $A_j = \text{card}(i, C_p; V^j)$ and $B_j = \text{card}(i, C_q; V^j)$; and

$$\lambda_p(i, L_p; V^j) = \frac{\sum_{X\in C_p} f(i, X; V^j)}{A_j} \quad \textit{and} \quad \lambda_q(i, L_q; V^j) = \frac{\sum_{X\in C_q} f(i, X; V^j)}{B_j}$$

This is a generalization of Ward's relation.

Note: this relation holds also for singletons $D(\{X\}, \{Y\}),\ X, Y \in \mathbf{U}$.

4. Using clamix

The package is used in order to cluster the selected data set. The data set can be large and each variable in the data set can be represented as a frequency/probability distribution over the possible values.

The package allows the user to:

- encode the data set to fit the representations with distributions
- transform data set into `symData` object
- cluster data with adapted leaders method
- cluster data with adapted hierarchical clustering method (chaining of leaders → hierarchical method is possible)

- analysis of clusters (dendrogram, statistical significance reports).

4.1. Encoding

Function `encodeSO` helps encoding numerical variables (single value variable) into frequency distribution bins. Encoding (the width and the number of bins) could be obtained by the function `makeEnc`.

** DODAJ + PRIMER TAKEGA DELA ** (ni narejeno za kodiranje spremenljivke - porazdelitve??!)

```
set.seed(42)
testset <- runif(100)
partCode <-"Rand"
ncat <- 10

makeEnc(testset,partCode,ncat,file="temp.R") # make "encRand" encoding
source("temp.R")
unlink("temp.R") # tidy up

testcat <- sapply(testset,function(x) encodeSO(x,encRand,NA)) # produce bins
toVector<- function(cat,ncat){x<-numeric(ncat);x[cat]<-1;return(x)}

# produce matrix
m <- sapply(testcat,function(x) toVector(x,ncat+1)) # one category more for "NA"
m <- as.data.frame(t(m))
names(m) <- names(get(paste("enc",partCode,sep=""))) # set names of variables
```

The value of a variable `m` is a `data.frame` of 100 units. This `data.frame` represents a variable random variable `testset` coded into 10 bins. Names of the bins are written as (usually) names of `data.frame` variables.

Function `makeEnc` produces the following code (a list of bins) into a temporary file "temp.R":

```
encRand <- list(
  "[0]" = function(x) x<=0,
  "(0,0.08998052]" = function(x) x<=0.08998062,
  "(0.08998052,0.2163854]" = function(x) x<=0.2163855,
  "(0.2163854,0.3590283]" = function(x) x<=0.3590284,
  "(0.3590283,0.4749971]" = function(x) x<=0.4749972,
  "(0.4749971,0.6117786]" = function(x) x<=0.6117787,
  "(0.6117786,0.6932048]" = function(x) x<=0.6932049,
  "(0.6932048,0.7846928]" = function(x) x<=0.7846929,
  "(0.7846928,0.914806]" = function(x) x<=0.9148061,
  "(0.914806,0.9888917]" = function(x) x<=0.9888918,
  "NA" = function(x) TRUE )
```

Encoding of ordinal or nominal variables can be made by the following code:

** ZA PRECISTITI ** prosojnica DATA AND METADATA - Karlsruhe?? **

```
cf <- read.table("./cars/cars.csv",header=TRUE,dec=".",row.names=1)
levType <- c("LI","KL","EN","KA","KB","RO","TE","KU")
typ <- factor(cf$type,levels=levType)
carsSO <- vector("list",26)
names(carsSO) <- colnames(cf)
carsSO$type <- as.integer(typ)
carsCats <- vector("list",26)
names(carsCats) <- colnames(cf)
carsCats$type <- c(levType,NA)
```

4.2. Transformation

A `list` with a number of `data.frame`s that consist of one variable described with distributions each can be transformed into a `symData` object. A `symData` object is an R class for description of histogram symbolic data. The class is represented as a list of:

`SOs`	a vector of symbolic objects (of class `symObject`); the transformed data set
`so`	an empty `symObject`
`namedSO`	an empty `symObject` with names for categories of each variable
`alpha`	a vector of weights α_j
`type`	a type of `symData`.

A `symObject` is an R class for description of a histogram symbolic object (one unit). It is represented as a list of variables. Each variable is represented as a vector. The last two components of the vector correspond to the number of `NA`s and the size (`num`) of the variable items.

The transformation of the data set is done via function `create.symData`.

```
data(popul06f)
data(popul06m)
datalist <- list("M"=popul06f,"F"=popul06m)
dataset <- create.symData(datalist,"fDist")
summary(dataset) # description of the class

summary symData :

Dimension (units x variables): 224 x 2
Type of distributions: fDist
Outlook of symObject:
symObject :
Number of variables: 2
M
  0-4   5-9 10-14 15-19 20-24 25-29 30-34 35-39 40-44 45-49 50-54 55-59 60-64 65-69
    0     0     0     0     0     0     0     0     0     0     0     0     0     0
70-74 75-79   80+    NA   num
```

```
    0     0     0     0     0
F
  0-4   5-9 10-14 15-19 20-24 25-29 30-34 35-39 40-44 45-49 50-54 55-59 60-64 65-69
    0     0     0     0     0     0     0     0     0     0     0     0     0     0
70-74 75-79   80+    NA   num
    0     0     0     0     0
```

Each `symObject` can be printed to the screen via `print` method:

```
print(dataset$SOs[[1]]) # print the first unit
```

```
symObject :
Number of variables: 2
M
 [1]  2682812  2205176  1875771  1581189  1328733  1112491   971031   792543   654282
[10]   536419   433404   341264   257107   178961   112708    60333    34298        0
[19] 15158522
F
 [1]  2800459  2316662  1977996  1670173  1407957  1193656  1026199   841592   696045
[10]   562946   444781   341583   251784   172667   106464    55525    31986        0
[19] 15898475
```

`symData` object can be checked via function `as.symData`. An empty `symObject` can be made via function `empty.symObject`.

4.3. Leaders method

Adapted leaders method for clustering `dataset` from the above transformation into 5 groups is run with

```
res2006leaders <- leaderSO(dataset,5)
```

The procedure is interactive — after the first step of the method the algorithm asks for the number of additional steps.

```
Times repeat =
```

If 0 or a negative number is specified the algorithm finishes. Otherwise it continues for the specified number of steps and asks again.

The result composes of `clust` — the partition of clusterings, `R` the minimal error (distance to leader) for each cluster, `p` the sum of errors for each cluster and `leaders_symData` a `symData` object of leaders for each cluster that is suitable for a direct input into `hclustSO` function (a function for the adapted hierarchical clustering). This part should be considered when the data set consists of more than 1000 units. It should be clustered with leaders method into a relatively large number of clusters first. Afterwards the resulting leaders should be clustered with hierarchical clustering method to decide upon the right number of clusters on the basis of the corresponding dendrogram.

4.4. Hierarchical clustering method

The adapted hierarchical clustering method for clustering `dataset` from the above transformation is run with

```
res2006hclust <- hclustSO(dataset)
```

The obtained results are compatible with the results of `hclust` from library `stats`. We can use the `plot` function from it to draw the dendrogram.

```
plot(res2006hclust,cex=0.2)
rect.hclust(res2006hclust,k=4)
```

The output dendrogram is found in the Figure 1.

4.5. Analysis

When the clusters are created they can be characterized according to the deviance from the total data set (`computeTotal`).

** DODATI, katero FUNKCIJO se bo implementiralo, prikazalo **

Acknowledgements

References

Batagelj V, Kejžar N, Korenjak-Černe S (2010). "A Way of Clustering Histogram Symbolic Data." Preprint.

Billard L, Diday E (2006). *Symbolic Data Analysis. Conceptual Statistics and Data Mining*. Wiley Series in Computational Statistics. Wiley.

Diday Eea (1979). *Optimisation en Classification Auomatique, Tomes 1., 2.* INRIA, Rocquencourt.

Hartigan J (1975). *Clustering Algorithms*. Wiley-Interscience: New York.

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, URL http://www.R-project.org.

Ward J (1963). "Hierarchical Groupinh to Optimize an Objective Function." *Journal of the American Statistical Association*, **58**, 236–244.

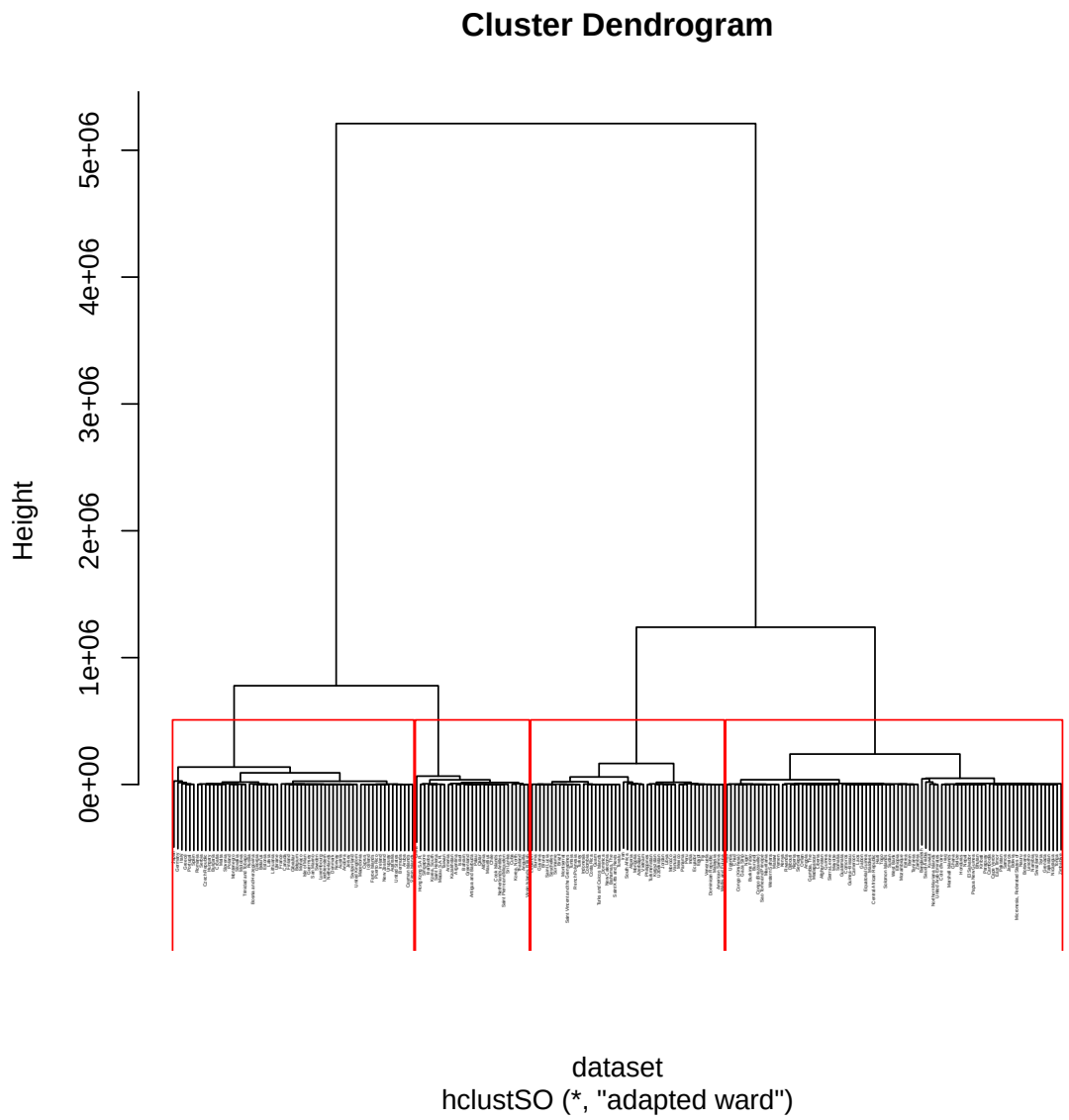

Figure 1: Dendrogram after the clustering with function `hclustSO`

Affiliation:

Nataša Kejžar
Faculty of Social Sciences
Kardeljeva ploščad 5
1000, Ljubljana, Slovenia
E-mail: natasa.kejzar@fdv.uni-lj.si
URL: http://www2.arnes.si/~nkejza/

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume VV, Issue II *Submitted:* yyyy-mm-dd
MMMMMM YYYY *Accepted:* yyyy-mm-dd