Clustering of mixed symbolic data based on cluster leaders

(Sub)sets

Vladimir Batagelj

UP Koper and IMFM Ljubljana

SDA 2023 – IX Workshop on Symbolic Data Analysis

CNAM, Paris, November 2-4, 2023

Outline

Vladimir Batagelj: vladimir.batagelj@fmf.uni-lj.si
Current version of slides (November 2, 2023 at 03 : 26): slides PDF
https://github.com/bavla/SDA

Clustering problem

We extend the approach to the clustering of modal symbolic data proposed in [9] to symbolic data in which a symbolic object $X = [x_1, x_2, \ldots, x_k]$ is described by a list of values of symbolic variables x_i that can be of different types (interval, histogram, set, classical scalar, etc.).

We use the criterion function of the following form

$$P(\mathbf{C}) = \sum_{C \in \mathbf{C}} p(C)$$

The *total error* $P(\mathbf{C})$ of the partition $\mathbf{C}$ is the sum of *cluster errors* $p(C)$ of its clusters $C \in \mathbf{C}$.

Cluster error

There are many ways to measure the cluster error $p(C)$. We shall assume a model in which the error of a cluster is the sum of deviations of its units from the cluster's representative T. For a given representative T and a cluster C we define the cluster error with respect to T:

$$p(C, T) = \sum_{X \in C} d(X, T),$$

where d is a selected dissimilarity measure. The best representative T_C is called a *leader*

$$T_C = \operatorname*{argmin}_T p(C, T)$$

Then, we define

$$p(C) = p(C, T_C) = \min_T \sum_{X \in C} d(X, T)$$

Cluster error

We assume that the leader T has the same description structure as the SOs, $T = [t_1, t_2, \ldots, t_k]$.

We introduce a dissimilarity measure between SOs and T with

$$d(X, T) = \sum_i \alpha_i d_i(x_i, t_i), \qquad \alpha_i \geq 0$$

where d_i is a dissimilarity compatible with the type of i^{th} variable.

Given a cluster C, the corresponding leader $T_C \in \mathbf{T}$ is the solution of

$$T_C = \operatorname*{argmin}_T \sum_{X \in C} d(X, T) = \operatorname*{argmin}_T \sum_{X \in C} \sum_i \alpha_i d_i(X, T)$$

$$= \operatorname*{argmin}_T \sum_i \alpha_i \sum_{X \in C} d_i(x_i, t_i) = [\operatorname*{argmin}_{t_i} \sum_{X \in C} d_i(x_i, t_i)]_{i=1}^k$$

By denoting $T_C = [t_1^*, t_2^*, \ldots, t_k^*]$ we obtain the following requirement:

$$t_i^* = \operatorname*{argmin}_{t_i} \sum_{X \in C} d_i(x_i, t_i).$$

Leaders

Because of the additivity of the model, we can observe each variable separately and simplify the notation by omitting the index i.

$$t^* = \underset{t}{\operatorname{argmin}} \sum_{X \in C} d(x, t)$$

In the following, we discuss the solutions to this optimization problem for different types of symbolic variables. For clustering symbolic data with types with known solutions, we can adapt the hierarchical clustering algorithm [2] and the leaders' algorithm [8].

Modal variables

A *modal* variable x is described by a list of frequencies (counts) $\mathbf{f} = (f_1, f_2, \ldots, f_k)$. In the following, we will use an equivalent representation by the pair $(n, \mathbf{p})$ where $n = \sum f_i$, $\mathbf{p} = (p_1, p_2, \ldots, p_k)$, and $p_i = f_i/n$.

The basic dissimilarities and the leader t, the leader z of the merged clusters and dissimilarity between merged clusters. Indices i and j are omitted.

$$w_t = \sum_{X \in C_t} w_x \qquad P_t = \sum_{X \in C_t} w_x p_x \qquad Q_t = \sum_{X \in C_t} w_x p_x^2$$

$$H_t = \sum_{X \in C_t} \frac{w_x}{p_x} \qquad G_t = \sum_{X \in C_t} \frac{w_x}{p_x^2}$$

Leaders

	$\delta(x,t)$	t	z	$D(C_u, C_v)$
δ_1	$(p_x - t)^2$	$\frac{P_t}{w_t}$	$\frac{w_u u + w_v v}{w_u + w_v}$	$\frac{w_u \cdot w_v}{w_u + w_v}(u - v)^2$
δ_2	$\left(\frac{p_x - t}{t}\right)^2$	$\frac{Q_t}{P_t}$	$\frac{uP_u + vP_v}{P_u + P_v}$	$\frac{P_u}{u}\left(\frac{u-z}{z}\right)^2 + \frac{P_v}{v}\left(\frac{v-z}{z}\right)^2$
δ_3	$\frac{(p_x - t)^2}{t}$	$\sqrt{\frac{Q_t}{w_t}}$	$\sqrt{\frac{u^2 w_u + v^2 w_v}{w_u + w_v}}$	$w_u \frac{(u-z)^2}{z} + w_v \frac{(v-z)^2}{z}$
δ_4	$\left(\frac{p_x - t}{p_x}\right)^2$	$\frac{H_t}{G_t}$	$\frac{H_u + H_v}{\frac{H_u}{u} + \frac{H_v}{v}}$	$G_u(u - z)^2 + G_v(v - z)^2$
δ_5	$\frac{(p_x - t)^2}{p_x}$	$\frac{w_t}{H_t}$	$\frac{w_u + w_v}{H_u + H_v}$	$w_u \frac{(u-z)^2}{u} + w_v \frac{(v-z)^2}{v}$
δ_6	$\frac{(p_x - t)^2}{p_x t}$	$\sqrt{\frac{P_t}{H_t}}$	$\sqrt{\frac{P_u + P_v}{\frac{P_u}{u^2} + \frac{P_v}{v^2}}}$	$\frac{P_u}{u}\frac{(u-z)^2}{uz} + \frac{P_v}{v}\frac{(v-z)^2}{vz}$

Interval variables

An *interval* variable x is described by the interval $[\underline{x}, \overline{x}]$ determined by its smallest value $\underline{x}$ and its largest value $\overline{x}$.

$$d(x,y) = \delta(\overline{x}, \overline{y}) + \delta(\underline{x}, \underline{y})$$

For $\delta = \delta_1$, $\delta_1(x,t) = (x-t)^2$ we get

$$t^* = (\overline{t}^*, \underline{t}^*) = \text{argmin}_t \sum_{x \in C} w_x d(x,t) =$$

$$= \text{argmin}_t \sum_{x \in C} w_x (\overline{x} - \overline{t})^2 + \text{argmin}_t \sum_{x \in C} w_x (\underline{x} - \underline{t})^2$$

and finally

$$\overline{t}^* = \frac{\sum_{x \in C} w_x \, \overline{x}}{\sum_{x \in C} w_x} \qquad \text{and} \qquad \underline{t}^* = \frac{\sum_{x \in C} w_x \underline{x}}{\sum_{x \in C} w_x}$$

Generalized Ward's relation for δ_1

To obtain compatibility with the adapted leaders' method, we define the dissimilarity between clusters C_u and C_v, $C_u \cap C_v = \emptyset$, as [2]

$$D(C_u, C_v) = p(C_u \cup C_v) - p(C_u) - p(C_v).$$

For a selected basic dissimilarity $\delta_1(x, t) = (x - t)^2$ we get

$$D(C_u, C_v) = \sum_i \alpha_i \frac{w_{ui} \cdot w_{vi}}{w_{ui} + w_{vi}}((\overline{u}_i - \overline{v}_i)^2 + (\underline{u}_i - \underline{v}_i)^2)$$

a *generalized Ward's relation* (with weights and with more variables). Note that this relations holds also for singletons $C_u = \{X\}$ or $C_v = \{Y\}$, $X, Y \in \mathcal{U}$.

Sets / multi modal

Given a basic set S a set variable Y over S can get for its value any subset of S. It can be represented as its characteristic (binary) vector.

We introduce two notions: a binary representative t and a multi-set representative R. We define a dissimilarity

$$d(R,t) = \sum_{i:t_i=0} R_i + \sum_{i:t_i=1} (M - R_i) \quad \text{where} \quad M = \max_i R_i$$

$$d(R,t) = (1-t) \cdot R + t \cdot (\mathbf{M} - R)$$

Describing units in clusters using multi-set representatives we have for the cluster error for leader t

$$p(C,t) = \sum_{X \in C} d(X,t)$$

Let $R_i = \sum_{X \in C} X_i$. The minimum value $p(C) = p(C,t^*)$ is attained for

$$t^* = as.integer([M \leq 2R_i])$$

Dissimilarity

t^* is a leader of the cluster C. Since cluster errors are easy to compute, we define the dissimilarity between two disjoint clusters C_p and C_q needed in the hierarchical clustering procedure as

$$D(C_p, C_q) = p(C_p \cup C_q) - p(C_p) - p(C_q)$$

Remark: Set symbolic variables are essentially hypergraphs that play a fundamental role in HOI (higher-order interactions) in the network science [5].

Symbolic data set

Oils

Symbolic data set SD is based symbolic data frame SDF with additional fields format, info and head

```
> SD$SDF
              Gravity Freezing   Iodine  Saponif       MajorAcids                         MAbin
Linseed 0.930, 0.935 -27, -18 170, 204 118, 196    1, 2, 3, 4, 5 1, 1, 1, 1, 1, 0, 0, 0, 0
Perilla 0.930, 0.937   -5, -4 192, 208 188, 197    1, 2, 3, 4, 6 1, 1, 1, 1, 0, 1, 0, 0, 0
Cotton  0.916, 0.918   -6, -1  99, 113 189, 198    1, 3, 4, 5, 6 1, 0, 1, 1, 1, 1, 0, 0, 0
Sesame  0.920, 0.926   -6, -4 104, 116 187, 193    1, 3, 4, 6, 7 1, 0, 1, 1, 0, 1, 1, 0, 0
Camelia 0.916, 0.917 -21, -15   80, 82 189, 193             1, 3 1, 0, 1, 0, 0, 0, 0, 0, 0
Olive   0.914, 0.919     0, 6   79, 90 187, 196       1, 3, 4, 6 1, 0, 1, 1, 0, 1, 0, 0, 0
Beef      0.86, 0.87   30, 38   40, 48 190, 199    3, 4, 5, 8, 6 0, 0, 1, 1, 1, 1, 0, 1, 0
Hog     0.858, 0.864   22, 32   53, 77 190, 202 1, 3, 4, 5, 6, 9 1, 0, 1, 1, 1, 1, 0, 0, 1
```

wiki

Symbolic data set structure

```
> str(Oils)
List of 4
 $ format: chr "SDAJSON"
 $ info  :List of 7
  ..$ dataset: chr "Oils"
  ..$ title  : chr "Oils and fats"
  ..$ by     : chr "Ichino M., Yaguchi H."
  ..$ ref    : chr "Generalized Minkowski metrics for mixed feature-type data analysis. IEEE Tra
  ..$ href   : chr [1:2] "https://ieeexplore.ieee.org/document/286391" "https://github.com/bavla
  ..$ creator: chr "V. Batagelj"
  ..$ date   : chr "Mon Oct 30 01:14:37 2023"
 $ head  :List of 3
  ..$ nUnits: num 8
  ..$ nVars : num 6
  ..$ vars  :List of 6
  .. ..$ V1:List of 2
  .. .. ..$ ID  : chr "Gravity"
  .. .. ..$ type: chr "interval"
  .. ..$ V2:List of 2
  .. .. ..$ ID  : chr "Freezing"
  .. .. ..$ type: chr "interval"
  .. ..$ V3:List of 2
  .. .. ..$ ID  : chr "Iodine"
  .. .. ..$ type: chr "interval"
  .. ..$ V4:List of 2
  .. .. ..$ ID  : chr "Saponif"
  .. .. ..$ type: chr "interval"
  .. ..$ V5:List of 4
  .. .. ..$ ID  : chr "MajorAcids"
  .. .. ..$ type: chr "set"
  .. .. ..$ cats: chr [1:9] "L" "Ln" "O" "P" ...
  .. .. ..$ long: chr [1:9] "linoleic" "linolenic" "oleic" "palmitic" ...
  .. ..$ V6:List of 4
  .. .. ..$ ID  : chr "MAbin"
  .. .. ..$ type: chr "members"
  .. .. ..$ cats: chr [1:9] "L" "Ln" "O" "P" ...
  .. .. ..$ long: chr [1:9] "linoleic" "linolenic" "oleic" "palmitic" ...
 $ SDF   :'data.frame': 8 obs. of  6 variables:
```

Dissimilarities


```
dMembers <- function(Y,p,q){
  P <- Y[[p]]$R; Q <- Y[[q]]$R; pp <- Y[[p]]$p; pq <- Y[[q]]$p
  R <- P+Q; M <- max(R); t <- as.integer(2*R >= M)
  ppq <- sum((1-t)*R + t*(M-R))
  return(ppq - pp - pq)
}

dIntSq <- function(Y,p,q){
  P <- Y[[p]]$L; Q <- Y[[q]]$L
  wp <- Y[[p]]$s; wq <- Y[[q]]$s
  return(wp*wq*sum((P-Q)**2)/(wp+wq))
}

# computes dissimilarity between SOs
# global: alpha, dSel, nSel
distSO <- function(U,p,q){
  D <- numeric(nSel)
  for(i in 1:nSel) {
    X <- dSel[[i]]; d <- X$d; Y <- U[[i]]
    D[i] <- d(Y,p,q)
  }
  dis <- as.numeric(D %*% alpha)
  if (is.na(dis)) dis <- Inf
  return(dis)
}
```

Update

```
updateL <- function(U,dSel,j,ip,iq){
  dt <- dSel[[j]]$dType; Y <- U[[j]]
  pp <- Y[[ip]]$p; pq <- Y[[iq]]$p
  if(dt == "membersR"){
    P <- Y[[ip]]$R; Q <- Y[[iq]]$R
    R <- P+Q; M <- max(R); t <- as.integer(2*R >= M)
    s <- Y[[ip]]$s + Y[[iq]]$s
    ppq <- sum((1-t)*R + t*(M-R))
    return(list(L=t,R=R,s=Y[[ip]]$s+Y[[iq]]$s,p=ppq))
  } else if(dt == "intervalSq"){
    P <- Y[[ip]]$L; Q <- Y[[iq]]$L
    Pr <- Y[[ip]]$R; Qr <- Y[[iq]]$R
    wp <- Y[[ip]]$s; wq <- Y[[iq]]$s
    t <- (wp*P+wq*Q)/(wp+wq); R <- c(min(Pr,Qr),max(Pr,Qr))
    return(list(L=t,R=R,s=wp+wq))
  } else cat(j,ip,iq, "Error\n")
}
```

Hierarchical clustering


```
hclustSO <- function(SD,dSel){
  orDendro <- function(i){if(i<0) return(-i)
    return(c(orDendro(m[i,1]),orDendro(m[i,2])))}

  nUnits <- SD$head$nUnits; nmUnits <- nUnits-1; nSel <- length(dSel)
  npUnits <- nUnits+1; n2mUnits <- nUnits+nmUnits
  w <- rep(1,nUnits)
  alpha <<- vars <- rep(NA,nSel)
  for(i in 1:nSel) {
    X <- dSel[[i]]; vars[i] <- X$var; alpha[i] <<- X$alpha }
  H <- SD$SDF[,vars]; U <- H
  for(i in 1:nSel) for(j in 1:nUnits)
    U[[i]][[j]] <- list(L=H[[i]][[j]],R=H[[i]][[j]],s=1,p=0)
  D <- matrix(nrow=nUnits,ncol=nUnits)
  for(p in 1:nmUnits) for(q in (p+1):nUnits) {
    D[q,p] <- D[p,q] <- distSO(U,p,q)
  }
  diag(D) <- Inf
  active <- 1:nUnits; m <- matrix(nrow=nmUnits,ncol=2)
  node <- rep(0,nUnits); h <- numeric(nmUnits)
  for(j in npUnits:n2mUnits) { U[nrow(U)+1,] <- vector("list",nSel)
    for(i in 1:nSel) U[[i]][[j]] <- list(L=NA,R=NA,s=1,p=0)}
  rownames(U)[npUnits:n2mUnits] <- paste("L",1:nmUnits,sep="")
```

. . . Hierarchical clustering

```
  for(k in 1:nmUnits){
    ind <- active[sapply(active,function(i) which.min(D[i,active]))]
    dd <- sapply(active,function(i) min(D[i,active]))
    pq <- which.min(dd)
    p<-active[pq]; q <- ind[pq]; h[k] <- D[p,q]
    if(node[p]==0){m[k,1] <- -p; ip <- p
    } else {m[k,1] <- node[p]; ip <- node[p]}
    if(node[q]==0){m[k,2] <- -q; iq <- q
    } else {m[k,2] <- node[q]; iq <- node[q]}
    ik <- nUnits + k
    for(j in 1:nSel) U[[j]][[ik]] <- updateL(U,dSel,j,ip,iq)
    active <- setdiff(active,p)
    for(s in setdiff(active,q)){
      is <- ifelse(node[s]==0,s,node[s])
      D[s,q] <- D[q,s] <- distSO(U,ik,is)
    }
    node[[q]] <- ik
  }
  for(i in 1:nmUnits) for(j in 1:2)
    if(m[i,j]>nUnits) m[i,j] <- m[i,j]-nUnits
  hc <- list(merge=m,height=h,order=orDendro(nmUnits),
    labels=rownames(SD$SDF),method=NULL,call=NULL,dist.method=NULL,
    leaders=U[npUnits:n2mUnits,])
  class(hc) <- "hclust"
  return(hc)
}
```

Running

```
> wdir <- "C:/Users/vlado/docs/papers/2023/SDA/Paris/test"
> setwd(wdir)
> library(jsonlite)
> b <- "https://raw.githubusercontent.com/bavla/"
> source(paste(b,"SDA/main/code/symclus.R",sep=""))
> SD <- fromJSON(paste(b,"symData/master/SDAJSON/Oils.json",sep=""))
> # source("symclus.R")
> # SD <- fromJSON("Oils.json")
> # str(SD)
> date()
[1] "Mon Oct 30 02:50:20 2023"
> dSel <- list( list(var=6,dType="membersR",d=dMembers,alpha=1),
+               list(var=1,dType="intervalSq",d=dIntSq,alpha=1000))
> nSel <- length(dSel); alpha <- rep(NA,nSel)
> for(i in 1:nSel) alpha[i] <- dSel[[i]]$alpha
> hc <- hclustTest(SD,dSel)
> # hc <- hclustSO(SD,dSel)
> plot(hc,hang=-1)
> hc$leaders
```

Oils

```
> hc <- hclustTest(SD,dSel)
vars: 6 1    alpha: 1 1000
          [,1]   [,2]   [,3]   [,4]   [,5]   [,6]   [,7]   [,8]
Linseed    Inf 2.0020 2.2425 4.0905 3.2600 3.2560 8.5625 8.1125
Perilla 2.0020    Inf 2.2785 2.1105 3.2980 1.2900 8.6945 8.2565
Cotton  2.2425 2.2785    Inf 2.0400 3.0005 1.0025 4.7200 4.1400
Sesame  4.0905 2.1105 2.0400    Inf 3.0485 1.0425 7.3680 6.8440
Camelia 3.2600 3.2980 3.0005 3.0485    Inf 2.0040 7.6725 7.0865
Olive   3.2560 1.2900 1.0025 1.0425 2.0040    Inf 5.6585 5.0805
Beef    8.5625 8.6945 4.7200 7.3680 7.6725 5.6585    Inf 3.0200
Hog     8.1125 8.2565 4.1400 6.8440 7.0865 5.0805 3.0200    Inf
>>> 1 3 6 3 6 1.0025 1 1
3 Lp1: 1 0 1 1 1 1 0 0 0     6 Lq1: 1 0 1 1 0 1 0 0 0
3 Rp1: 1 0 1 1 1 1 0 0 0     6 Rq1: 1 0 1 1 0 1 0 0 0
3 Lp2: 0.916 0.918     6 Lq2: 0.914 0.919
3 Rp2: 0.916 0.918     6 Rq2: 0.914 0.919
9 Lk1: 1 0 1 1 1 1 0 0 0     9 Lk2: 0.915 0.9185
9 Rk1: 2 0 2 2 1 2 0 0 0     9 Rk2: 0.914 0.919
active: 1 2 4 5 6 7 8
h: 1.0025 0 0 0 0 0 0
>>> 2 1 2 1 2 2.002 1 1
1 Lp1: 1 1 1 1 1 0 0 0 0     2 Lq1: 1 1 1 1 0 1 0 0 0
1 Rp1: 1 1 1 1 1 0 0 0 0     2 Rq1: 1 1 1 1 0 1 0 0 0
1 Lp2: 0.93 0.935     2 Lq2: 0.93 0.937
1 Rp2: 0.93 0.935     2 Rq2: 0.93 0.937
10 Lk1: 1 1 1 1 1 1 0 0 0     10 Lk2: 0.93 0.936
10 Rk1: 2 2 2 2 1 1 0 0 0     10 Rk2: 0.93 0.937
active: 2 4 5 6 7 8
h: 1.0025 2.002 0 0 0 0 0
```

Oils

```
>>> 3 4 6 4 9 2.054167 1 2
4 Lp1: 1 0 1 1 0 1 1 0 0     9 Lq1: 1 0 1 1 1 1 0 0 0
4 Rp1: 1 0 1 1 0 1 1 0 0     9 Rq1: 2 0 2 2 1 2 0 0 0
4 Lp2: 0.92 0.926      9 Lq2: 0.915 0.9185
4 Rp2: 0.92 0.926      9 Rq2: 0.914 0.919
11 Lk1: 1 0 1 1 0 1 0 0 0     11 Lk2: 0.9166667 0.921
11 Rk1: 3 0 3 3 1 3 1 0 0     11 Rk2: 0.914 0.926
active: 2 5 6 7 8
h: 1.0025 2.002 2.054167 0 0 0 0
>>> 4 7 8 7 8 3.02 1 1
7 Lp1: 0 0 1 1 1 1 0 1 0     8 Lq1: 1 0 1 1 1 1 0 0 1
7 Rp1: 0 0 1 1 1 1 0 1 0     8 Rq1: 1 0 1 1 1 1 0 0 1
7 Lp2: 0.86 0.87      8 Lq2: 0.858 0.864
7 Rp2: 0.86 0.87      8 Rq2: 0.858 0.864
12 Lk1: 1 0 1 1 1 1 0 1 1     12 Lk2: 0.859 0.867
12 Rk1: 1 0 2 2 2 2 0 1 1     12 Rk2: 0.858 0.87
active: 2 5 6 8
h: 1.0025 2.002 2.054167 3.02 0 0 0
>>> 5 5 6 5 11 4.012333 1 3
5 Lp1: 1 0 1 0 0 0 0 0 0     11 Lq1: 1 0 1 1 0 1 0 0 0
5 Rp1: 1 0 1 0 0 0 0 0 0     11 Rq1: 3 0 3 3 1 3 1 0 0
5 Lp2: 0.916 0.917      11 Lq2: 0.9166667 0.921
5 Rp2: 0.916 0.917      11 Rq2: 0.914 0.926
13 Lk1: 1 0 1 1 0 1 0 0 0     13 Lk2: 0.9165 0.92
13 Rk1: 4 0 4 3 1 3 1 0 0     13 Rk2: 0.914 0.926
active: 2 6 8
h: 1.0025 2.002 2.054167 3.02 4.012333 0 0
```

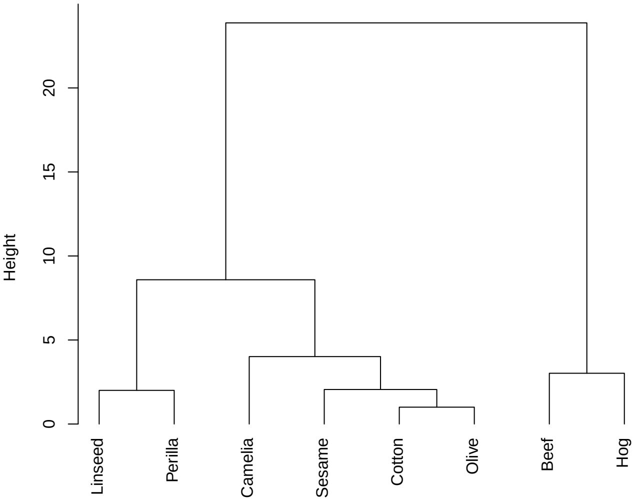
Oils

```
>>> 6 2 6 10 13 8.584333 2 4
10 Lp1: 1 1 1 1 1 1 0 0 0     13 Lq1: 1 0 1 1 0 1 0 0 0
10 Rp1: 2 2 2 2 1 1 0 0 0     13 Rq1: 4 0 4 3 1 3 1 0 0
10 Lp2: 0.93 0.936     13 Lq2: 0.9165 0.92
10 Rp2: 0.93 0.937     13 Rq2: 0.914 0.926
14 Lk1: 1 0 1 1 0 1 0 0 0     14 Lk2: 0.921 0.9253333
14 Rk1: 6 2 6 5 2 4 1 0 0     14 Rk2: 0.914 0.937
active: 6 8
h: 1.0025 2.002 2.054167 3.02 4.012333 8.584333 0
>>> 7 6 8 14 12 23.87017 6 2
14 Lp1: 1 0 1 1 0 1 0 0 0     12 Lq1: 1 0 1 1 1 1 0 1 1
14 Rp1: 6 2 6 5 2 4 1 0 0     12 Rq1: 1 0 2 2 2 2 0 1 1
14 Lp2: 0.921 0.9253333     12 Lq2: 0.859 0.867
14 Rp2: 0.914 0.937     12 Rq2: 0.858 0.87
15 Lk1: 1 0 1 1 1 1 0 0 0     15 Lk2: 0.9055 0.91075
15 Rk1: 7 2 8 7 4 6 1 1 1     15 Rk2: 0.858 0.937
active: 8
h: 1.0025 2.002 2.054167 3.02 4.012333 8.584333 23.87017
```

Oils / leaders

```
> hc$leaders
                                                    MAbin
L1 1, 0, 1, 1, 1, 1, 0, 0, 0, 2, 0, 2, 2, 1, 2, 0, 0, 0, 2
L2 1, 1, 1, 1, 1, 1, 0, 0, 0, 2, 2, 2, 2, 1, 1, 0, 0, 0, 2
L3 1, 0, 1, 1, 0, 1, 0, 0, 0, 3, 0, 3, 3, 1, 3, 1, 0, 0, 3
L4 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 2, 2, 2, 2, 0, 1, 1, 2
L5 1, 0, 1, 1, 0, 1, 0, 0, 0, 4, 0, 4, 3, 1, 3, 1, 0, 0, 4
L6 1, 0, 1, 1, 0, 1, 0, 0, 0, 6, 2, 6, 5, 2, 4, 1, 0, 0, 6
L7 1, 0, 1, 1, 1, 1, 0, 0, 0, 7, 2, 8, 7, 4, 6, 1, 1, 1, 8
                                                Gravity
L1                0.9150, 0.9185, 0.9140, 0.9190, 2.0000
L2                     0.930, 0.936, 0.930, 0.937, 2.000
L3 0.9166667, 0.9210000, 0.9140000, 0.9260000, 3.0000000
L4                     0.859, 0.867, 0.858, 0.870, 2.000
L5                0.9165, 0.9200, 0.9140, 0.9260, 4.0000
L6 0.9210000, 0.9253333, 0.9140000, 0.9370000, 6.0000000
L7           0.90550, 0.91075, 0.85800, 0.93700, 8.00000
```

Oils / Dendrograme



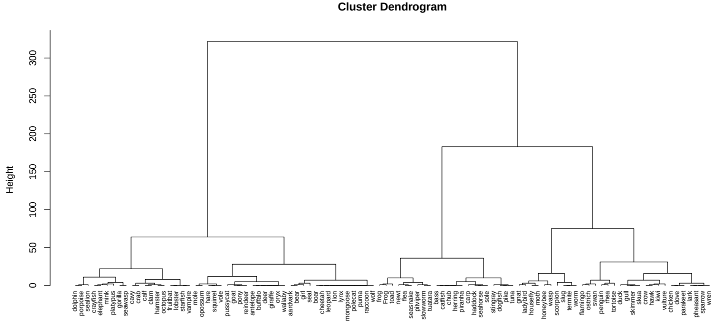
wiki

Zoo / Running

wiki

```
> SD <- fromJSON(paste(b,"symData/master/SDAJSON/ZooSDA.json",sep=""))
> # str(SD)
> # artificial second variable because of problems with a single variabl
> nUnits <- SD$head$nUnits
> u <- vector("list",nUnits)
> for(i in 1:nUnits) u[[i]] <- c(1,2)
> SD$SDF$skip <- u
> date()
[1] "Thu Nov  2 01:07:15 2023"
> dSel <- list( list(var=4,dType="membersR",d=dMembers,alpha=1),
+               list(var=5,dType="intervalSq",d=dIntSq,alpha=0))
> nSel <- length(dSel); alpha <- rep(NA,nSel)
> for(i in 1:nSel) alpha[i] <- dSel[[i]]$alpha
> hc <- hclustSO(SD,dSel)
> plot(hc,hang=-1,cex=0.7)
```

Zoo / dendrogram



wiki

Zoo / top level clusters

```
> m <- hc$merge
> for(i in 90:100)
+    cat("L",i," = L",m[i,1]," + L",m[i,2],sep="","\n")
L90 = L70 + L81
L91 = L84 + L75
L92 = L82 + L78
L93 = L89 + L86
L94 = L90 + L83
L95 = L85 + L91
L96 = L88 + L87
L97 = L93 + L94
L98 = L92 + L95
L99 = L96 + L98
L100 = L97 + L99
```

Zoo / Top level leaders

```
> lab <- SD$head$vars$V4$cats
> for(i in 80:100)
+   cat(i,":",lab[as.logical(hc$leaders$battrs[[i]]$L)],"\n")
80 : eggs aquatic predator toothed backbone fins tail
81 : hair milk toothed backbone breathes tail catsize
82 : hair eggs airborne breathes
83 : hair milk predator toothed backbone breathes tail catsize
84 : feathers eggs airborne aquatic predator backbone breathes tail
85 : feathers eggs backbone breathes tail catsize
86 : hair eggs milk aquatic predator toothed backbone breathes tail
     domestic
87 : eggs aquatic predator toothed backbone fins tail
88 : eggs aquatic predator toothed backbone breathes tail
89 : hair milk aquatic predator toothed backbone breathes tail catsize
90 : hair milk toothed backbone breathes tail catsize
91 : feathers eggs airborne backbone breathes tail
92 : eggs airborne breathes
93 : hair eggs milk aquatic predator toothed backbone breathes tail
     catsize
94 : hair milk predator toothed backbone breathes tail catsize
95 : feathers eggs airborne backbone breathes tail
96 : eggs aquatic predator toothed backbone fins tail
97 : hair milk predator toothed backbone breathes tail catsize
98 : feathers eggs airborne backbone breathes tail
99 : eggs predator backbone breathes tail
100 : hair eggs predator toothed backbone breathes tail catsize
```

Conclusions

- integrate modal symbolic variables
- implement adapted leaders method
- resolve the single variable problem in R
- monotonicity of set dendrograms
- weights
- fairness – balancing variables
- additional interesting data sets (The World Factbook)

Acknowledgments

The computational work reported in this presentation was performed using R library `symclus`. The code and data are available at Github/Bavla bavla.

This work is supported in part by the Slovenian Research Agency (research program P1-0294, research program CogniCom (0013103) at the University of Primorska, and research projects J5-2557, J1-2481, and J5-4596), and prepared within the framework of the COST action CA21163 (HiTEc).

References I

- Anderberg MR (1973). Cluster analysis for applications. Academic Press, New York

- Batagelj, V. (1988). Generalized ward and related clustering problems. In Bock, H.H. (ed) *Classification and related methods of data analysis*, pp. 67–74, North-Holland, Amsterdam.

- Batagelj, V., Kejžar, N., Korenjak-Černe, S. (2015). Clustering of Modal Valued Symbolic Data. arXiv:1507.06683

- Batagelj, V., Praprotnik, S.(2016). An algebraic approach to temporal network analysis based on temporal quantities. Social Network Analysis and Mining, 6(1), 1-22

- Federico Battiston, Giulia Cencetti, Iacopo Iacopini, Vito Latora, Maxime Lucas, Alice Patania, Jean-Gabriel Young, Giovanni Petri: Networks beyond pairwise interactions: Structure and dynamics. Physics Reports, Volume 874, 2020, Pages 1-92.

References II

Bock HH, Diday E (eds) (2000). Analysis of symbolic data. Exploratory methods for extracting statistical information from complex data. Springer, Heidelberg

Billard L, Diday E (2006). Symbolic data analysis. Conceptual statistics and data mining. Wiley, Chichester

Hartigan, J.A. (1975) . Clustering algorithms. Wiley-Interscience, New York.

Kejžar, N., Korenjak-Černe, S., Batagelj, V. (2021). Clustering of modal-valued symbolic data. *Advances in Data Analysis and Classification 15*, 513–541.

Kejžar N, Korenjak-Černe S, Batagelj V (2011) Clustering of distributions: A case of patent citations. *Journal of Classification* 28(2):156-183

Diday E (1979) Optimisation en classification automatique. Tome 1.,2. INRIA, Rocquencourt (in French)

References III

Everitt BS, Landau S, Leese M (2001) Cluster analysis. Fourth Edition. Arnold, London

Korenjak-Černe, S., Kejžar, N., Batagelj, V.(2020) Clustering and generalized ANOVA for symbolic data constructed from open data. In: Diday, Edwin (ed.), et al. Advances in data science : symbolic, complex, and network data. Volume 4, Big data, artificial intelligence and data analysis. SET coordinated by Jacques Janssen. Newark: John Wiley & Sons., str. 209-228.

Korenjak-Černe S, Batagelj V (2002) Symbolic data analysis approach to clustering large datasets. In: *Jajuga K, Sokołowski A, Bock HH (eds) 8th Conference of the International Federation of Classification Societies, July 16-19, 2002, Cracow, Poland, Classification, clustering and data analysis*. Springer, Berlin, pp 319-327

Korenjak-Černe S, Batagelj V, Japelj Pavešić B (2011) Clustering large data sets described with discrete distributions and its application on TIMSS data set. *Statistical Analysis and Data Mining* 4(2):199-215

References IV

- Korenjak-Černe S, Kejžar N, Batagelj V (2015) A weighted clustering of population pyramids for the world's countries, 1996, 2001, 2006. *Population Studies* 69(1):105-120
- Ward JH. (1963) Hierarchical grouping to optimize an objective function. J Am Stat Assoc 58:236-244