Clamix / Cars

Food

Specificity

Transforming car data into SO

The raw data were obtained from **Cars Catalog 1997** based on *Katalog Avtomobilov '97 / Posebna priloga Dela in Slovenskih novic April '97* (by Janko Blagojevič). Transformation into symbolic objects (SOs) by Vladimir Batagelj, 29. July 2010.

The scheme of analysis is described in `analysis.R` (see also `trace.txt`). The cars data set is already transformed in symbolic objects (`cars.so` and `cars.meta`). To recreate or change transformation see `makeCars.R`.

The relevant files are collected in cars97.zip .

```
cars.csv    - cars data set
cars.nam    - short and long names of cars
encCars.R   - encoding rules for the cars data set
makeCars.R  - creation of  cars.so  and  cars.meta
analysis.R  - example steps in analysis
trace.txt   - trace of analysis

cars.Rdata  - encoded cars data
cars.so     - cars data set transformed in symbolic objects
cars.meta   - metadata for cars symbolic objects - needed for
              post-analyses
cars.rez    - results of leaders method
dendro.pdf  - dendrogram from hierarchical clustering
```

See also 34th GfKl / 21-23. July 2010 (slides 13-16) and (Nataša's approach) Wednesday seminar 1187 / 27. October 2010 (slides 23-27).

Details

For transforming the data we have to prepare the encoding rules (file `encCars.R`) for all variables:

```
# V. Batagelj, 29.7.2010
# ------------------------------------------------------------------

encPrice <- list(
  "[950,2100]"   = function(x) x<=2100,
  "(2100,2800]"  = function(x) x<=2800,
  "(2800,3700]"  = function(x) x<=3700,
  "(3700,5000]"  = function(x) x<=5000,
  "(5000,6550]"  = function(x) x<=6550,
  "(6550,45000]" = function(x) x<=45000,
  "NA"           = function(x) TRUE )

levType <- c("LI","KL","EN","KA","KB","RO","TE","KU")

namesNumDoors <- c(2:5,"NA")
codeNumDoors <- function(x){
  if(is.na(x)) return(5)
  if((2<=x)&&(x<=5)) return(x-1)
  return(0)
}

namesNumPassen <- c(2:8,"NA")
codeNumPassen <- function(x){
  if(is.na(x)) return(8)
  if((2<=x)&&(x<=8)) return(x-1)
  return(0)
}

levMsite <- c("S","Z","SR")
```

```
levDrive <- c("S","Z","S4","E")

encLen <- list(
  "[2600,4010]"  = function(x) x<=4010,
  "(4010,4245]"  = function(x) x<=4245,
  "(4245,4470]"  = function(x) x<=4470,
  "(4470,4555]"  = function(x) x<=4555,
  "(4555,4761]"  = function(x) x<=4761,
  "(4761,6000]"  = function(x) x<=6000,
  "NA"           = function(x) TRUE )

encWid <- list(
  "[1350,1680]"  = function(x) x<=1680,
  "(1680,1700]"  = function(x) x<=1700,
  "(1700,1735]"  = function(x) x<=1735,
  "(1735,1780]"  = function(x) x<=1780,
  "(1780,1812]"  = function(x) x<=1812,
  "(1812,2000]"  = function(x) x<=2000,
  "NA"           = function(x) TRUE )

encHei <- list(
  "[1045,1388]"  = function(x) x<=1388,
  "(1388,1410]"  = function(x) x<=1410,
  "(1410,1420]"  = function(x) x<=1420,
  "(1420,1438]"  = function(x) x<=1438,
  "(1438,1490]"  = function(x) x<=1490,
  "(1490,2500]"  = function(x) x<=2500,
  "NA"           = function(x) TRUE )

encWba <- list(
  "[1450,2445]"  = function(x) x<=2445,
  "(2445,2510]"  = function(x) x<=2510,
  "(2510,2590]"  = function(x) x<=2590,
  "(2590,2680]"  = function(x) x<=2680,
  "(2680,2730]"  = function(x) x<=2730,
  "(2730,3500]"  = function(x) x<=3500,
  "NA"           = function(x) TRUE )

encLug <- list(
  "[75,279]"     = function(x) x<=279,
  "(279,360]"    = function(x) x<=360,
  "(360,429]"    = function(x) x<=429,
  "(429,480]"    = function(x) x<=480,
  "(480,560]"    = function(x) x<=560,
  "(560,5000]"   = function(x) x<=5000,
  "NA"           = function(x) TRUE )

levELu <- c("da","ne")

encFCa <- list(
  "[30,48]"      = function(x) x<=48,
  "(48,53]"      = function(x) x<=53,
  "(53,60]"      = function(x) x<=60,
  "(60,66]"      = function(x) x<=66,
  "(66,76]"      = function(x) x<=76,
  "(76,150]"     = function(x) x<=150,
  "NA"           = function(x) TRUE )

encWei <- list(
  "[225,940]"    = function(x) x<=940,
  "(940,1080]"   = function(x) x<=1080,
  "(1080,1220]"  = function(x) x<=1220,
  "(1220,1350]"  = function(x) x<=1350,
  "(1350,1540]"  = function(x) x<=1540,
  "(1540,3000]"  = function(x) x<=3000,
  "NA"           = function(x) TRUE )

encMlo <- list(
  "[70,455]"     = function(x) x<=455,
  "(455,490]"    = function(x) x<=490,
  "(490,515]"    = function(x) x<=515,
  "(515,550]"    = function(x) x<=550,
  "(550,650]"    = function(x) x<=650,
  "(650,3000]"   = function(x) x<=3000,
  "NA"           = function(x) TRUE )

encDsp <- list(
  "[796,1490]"   = function(x) x<=1490,
  "(1490,1690]"  = function(x) x<=1690,
  "(1690,1886]"  = function(x) x<=1886,
  "(1886,2000]"  = function(x) x<=2000,
  "(2000,2500]"  = function(x) x<=2500,
  "(2500,6000]"  = function(x) x<=6000,
  "NA"           = function(x) TRUE )
```

```
encMkw <- list(
  "[26,55]"      = function(x) x<=55,
  "(55,66]"      = function(x) x<=66,
  "(66,85]"      = function(x) x<=85,
  "(85,99]"      = function(x) x<=99,
  "(99,125]"     = function(x) x<=125,
  "(125,400]"    = function(x) x<=400,
  "NA"           = function(x) TRUE )

encMkm <- list(
  "[25,75]"      = function(x) x<=75,
  "(75,99]"      = function(x) x<=99,
  "(99,118]"     = function(x) x<=118,
  "(118,144]"    = function(x) x<=144,
  "(144,200]"    = function(x) x<=200,
  "(200,500]"    = function(x) x<=500,
  "NA"           = function(x) TRUE )

encRMp <- list(
  "[3400,4500]"  = function(x) x<=4500,
  "(4500,5250]"  = function(x) x<=5250,
  "(5250,5500]"  = function(x) x<=5500,
  "(5500,5700]"  = function(x) x<=5700,
  "(5700,6000]"  = function(x) x<=6000,
  "(6000,9000]"  = function(x) x<=9000,
  "NA"           = function(x) TRUE )

encMtq <- list(
  "[55,124]"     = function(x) x<=124,
  "(124,145]"    = function(x) x<=145,
  "(145,172]"    = function(x) x<=172,
  "(172,202]"    = function(x) x<=202,
  "(202,270]"    = function(x) x<=270,
  "(270,600]"    = function(x) x<=600,
  "NA"           = function(x) TRUE )

encRMt <- list(
  "[1600,2250]"  = function(x) x<=2250,
  "(2250,2950]"  = function(x) x<=2950,
  "(2950,3500]"  = function(x) x<=3500,
  "(3500,4000]"  = function(x) x<=4000,
  "(4000,4500]"  = function(x) x<=4500,
  "(4500,7000]"  = function(x) x<=7000,
  "NA"           = function(x) TRUE )

levTrans <- c("A3","A4","A5","R4","R5","R5*","R6","S5","T4","T5","RS")

levBreak <- c("K/B","K/K")

encFue <- list(
  "[3.6,5]"      = function(x) x<=5,
  "(5,5.6]"      = function(x) x<=5.6,
  "(5.6,6.1]"    = function(x) x<=6.1,
  "(6.1,6.9]"    = function(x) x<=6.9,
  "(6.9,8]"      = function(x) x<=8,
  "(8,20]"       = function(x) x<=20,
  "NA"           = function(x) TRUE )

encAcc <- list(
  "[4.4,9.1]"    = function(x) x<=9.1,
  "(9.1,10.6]"   = function(x) x<=10.6,
  "(10.6,11.6]"  = function(x) x<=11.6,
  "(11.6,12.9]"  = function(x) x<=12.9,
  "(12.9,15.3]"  = function(x) x<=15.3,
  "(15.3,32]"    = function(x) x<=32,
  "NA"           = function(x) TRUE )

encSpe <- list(
  "[130,163]"    = function(x) x<=163,
  "(163,174]"    = function(x) x<=174,
  "(174,187]"    = function(x) x<=187,
  "(187,200]"    = function(x) x<=200,
  "(200,215]"    = function(x) x<=215,
  "(215,400]"    = function(x) x<=400,
  "NA"           = function(x) TRUE )
```

The file `Clamix3.R` contains some functions supporting the encoding:

```
# creates an empty SO
emptySO <- function(nCats){
  nVar <- length(nCats)
  s <- vector("list", nVar)
  for(i in 1:nVar) s[[i]] <- double(nCats[i]+1)
```

```
  return(s)
}

# encode numerical vector using given encoding
encodeSO <- function(x,encoding,codeNA){
  if(is.na(x)) return(codeNA)
  for(i in 1:length(encoding)) if(encoding[[i]](x)) return(i)
}
```

The encoding is done by commands in the file `makeCars.R`:

```
setwd("C:/Users/Batagelj/work/clamix/clamix.R")
source("C:\\Users\\Batagelj\\work\\clamix\\clamix.R\\cars\\encCars.R")
source("C:\\Users\\Batagelj\\work\\clamix\\clamix.R\\clamix3.R")
cf <- read.table("./cars/cars.csv",header=TRUE,dec=".",row.names=1)
numSO <- nrow(cf)
nVar <- ncol(cf); nVarP <- nVar+1

carsSO <- vector("list",nVar)
names(carsSO) <- colnames(cf)
carsCats <- vector("list",nVar)
names(carsCats) <- colnames(cf)

carsCats$price <- names(encPrice)
carsSO$price <- sapply(cf$price,function(x) encodeSO(x,encPrice,7))

carsCats$type <- c(levType,NA)
carsSO$type <- as.integer(factor(cf$type,levels=levType))

carsCats$NumDoors <- namesNumDoors
carsSO$NumDoors <- sapply(cf$NumDoors,codeNumDoors)

carsCats$NumPassen <- namesNumPassen
carsSO$NumPassen <- sapply(cf$NumPassen,codeNumPassen)

carsCats$motorsite <- c(levMsite,NA)
carsSO$motorsite <- as.integer(factor(cf$motorsite,levels=levMsite))

carsCats$drive <- c(levDrive,NA)
carsSO$drive <- as.integer(factor(cf$drive,levels=levDrive))

carsCats$length <- names(encLen)
carsSO$length <- sapply(cf$length,function(x) encodeSO(x,encLen,7))

carsCats$width <- names(encWid)
carsSO$width <- sapply(cf$width,function(x) encodeSO(x,encWid,7))

carsCats$height <- names(encHei)
carsSO$height <- sapply(cf$height,function(x) encodeSO(x,encHei,7))

carsCats$wheelbase <- names(encWba)
carsSO$wheelbase <- sapply(cf$wheelbase,function(x) encodeSO(x,encWba,7))

carsCats$luggage <- names(encLug)
carsSO$luggage <- sapply(cf$luggage,function(x) encodeSO(x,encLug,7))

carsCats$enlarLugg <- c(levELu,NA)
carsSO$enlarLugg <- as.integer(factor(cf$enlarLugg,levels=levELu))

carsCats$fuelCapac <- names(encFCa)
carsSO$fuelCapac <- sapply(cf$fuelCapac,function(x) encodeSO(x,encFCa,7))

carsCats$weight <- names(encWei)
carsSO$weight <- sapply(cf$weight,function(x) encodeSO(x,encWei,7))

carsCats$maxLoad <- names(encMlo)
carsSO$maxLoad <- sapply(cf$maxLoad,function(x) encodeSO(x,encMlo,7))

carsCats$displace <- names(encDsp)
carsSO$displace <- sapply(cf$displace,function(x) encodeSO(x,encDsp,7))

carsCats$maxPowKW <- names(encMkw)
carsSO$maxPowKW <- sapply(cf$maxPowKW,function(x) encodeSO(x,encMkw,7))

carsCats$maxPowKM <- names(encMkm)
carsSO$maxPowKM <- sapply(cf$maxPowKM,function(x) encodeSO(x,encMkm,7))

carsCats$rpm_maxPow <- names(encRMp)
carsSO$rpm_maxPow <- sapply(cf$rpm_maxPow,function(x) encodeSO(x,encRMp,7))

carsCats$maxTorque <- names(encMtq)
carsSO$maxTorque <- sapply(cf$maxTorque,function(x) encodeSO(x,encMtq,7))
```

```
carsCats$rpm_maxTor <- names(encRMt)
carsSO$rpm_maxTor <- sapply(cf$rpm_maxTor,function(x) encodeSO(x,encRMt,7))

carsCats$transmiss <- c(levTrans,NA)
carsSO$transmiss <- as.integer(factor(cf$transmiss,levels=levTrans))

carsCats$breaks <- c(levBreak,NA)
carsSO$breaks <- as.integer(factor(cf$breaks,levels=levBreak))

carsCats$minFuelCon <- names(encFue)
carsSO$minFuelCon <- sapply(cf$minFuelCon,function(x) encodeSO(x,encFue,7))

carsCats$accelTime <- names(encAcc)
carsSO$accelTime <- sapply(cf$accelTime,function(x) encodeSO(x,encAcc,7))

carsCats$maxSpeed <- names(encSpe)
carsSO$maxSpeed <- sapply(cf$maxSpeed,function(x) encodeSO(x,encSpe,7))

save(carsSO,file="./cars/cars.Rdata")
nCats <- as.integer(sapply(carsCats,length))
so <- emptySO(nCats)
# !!! problems with ' in car names;  ' -> "'"  in the file  cars.nam
cn <- read.table("./cars/cars.nam",header=FALSE,sep=";")
long <- as.vector(cn[,2])
namedSO <- so
names(namedSO) <- c(names(carsCats),"num")
for(i in 1:nVar) names(namedSO[[i]]) <- carsCats[[i]]
save(nVar,nVarP,so,namedSO,numSO,long,carsCats,file="./cars/cars.meta")
SOs <- vector("list",numSO)
for(i in 1:numSO){
  st <- so
  for(j in 1:nVar) st[[j]][carsSO[[j]][[i]]] <- 1
  st$num <- 1
  names(SOs)[[i]] <- long[[i]]
  SOs[[i]] <- st
}
save(nVar,nVarP,so,numSO,SOs,file="./cars/cars.so")
```

The encoded data are saved to `cars.Rdata` and further transformed and saved to `cars.meta` and `cars.so`.

Clustering

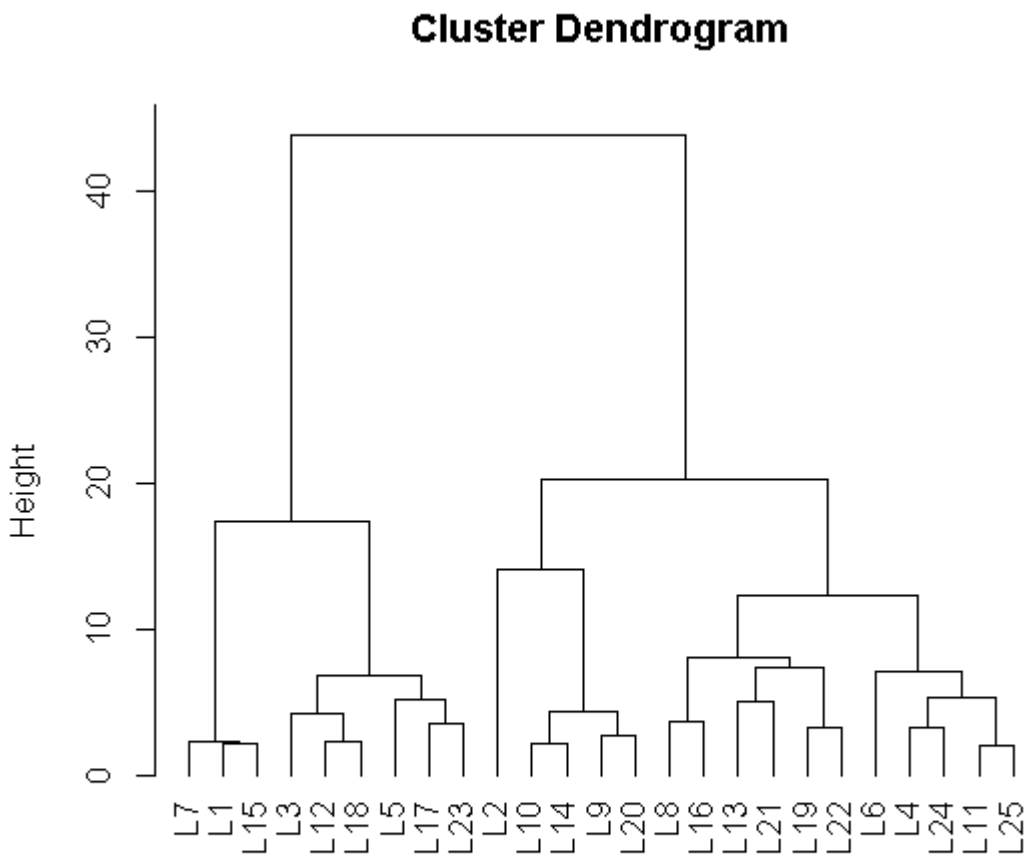
From the file specific.zip run the commands on `analysisCars2.R`. The cars data are stored on two files `cars.so` (symbolic objects) and `cars.meta` (metadata - names of units, variables and categories, …).

After loading the data we run the leaders method `leaderSO`. I started with 10 steps, 5 steps, 3 steps, 3 steps, … The local minimum was attained in 22th step.

Then we apply the hierarchical clustering `hclustSO` to leaders and plot the corresponding dendrogram. Finally we list the units in cluster 10.

```
> # gDist - Cars data / Analysis
> # VB: 30.7.2010
> setwd("C:/Users/Batagelj/work/clamix/clamix.R")
> source("C:\\Users\\Batagelj\\work\\clamix\\clamix.R\\clamix2.R")
> load("./cars2/cars.so")
> load("./cars2/cars.meta")
> alpha <-rep(1/nVar,nVar)
> rez <- leaderSO(SOs,25)

Step 1
clust
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23
 81 156  16 106  30  72 164  71  36   5  14  62  18  81  14  48  63   5  19  30  40  29 120
 24  25
 52  17
 [1] 0.3923260 0.4598792 0.3595308 0.3851367 0.3937892 0.3979975 0.3761325 0.4420161 0.4101287
[10] 0.3806951 0.3824818 0.3799591 0.3841496 0.4098006 0.3720880 0.4007372 0.3568077 0.3663876
[19] 0.3798095 0.4108323 0.3424131 0.3495385 0.3975587 0.3872848 0.3801581
 [1] -0.3923260 -0.4598792 -0.3595308 -0.3851367 -0.3937892 -0.3979975 -0.3761325 -0.4420161
 [9] -0.4101287 -0.3806951 -0.3824818 -0.3799591 -0.3841496 -0.4098006 -0.3720880 -0.4007372
[17] -0.3568077 -0.3663876 -0.3798095 -0.4108323 -0.3424131 -0.3495385 -0.3975587 -0.3872848
[25] -0.3801581
 [1] 24.894921 55.357940  4.995569 32.930782 10.302790 23.652320 49.411368 22.531687 13.268613
```

```
[10]  1.741471  4.518207 19.784571  5.659036 28.469946  4.440819 15.499487 19.835808  1.677481
[19]  6.198282 10.350611 12.508263  9.155846 37.397783 17.595627  5.444684
[1] 437.6239
Times repeat = 10

Step 2
....
[1] 284.6798
Times repeat = 2

Step 22
clust
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23
101 100  49  46  68 101  56  56  36  22  33  47  62  69  14  81  77  30  34  28  55  51  44
 24  25
 42  47
 [1] 0.2283072 0.3878423 0.2664915 0.3595681 0.3520013 0.3283766 0.3229371 0.4070963 0.3272792
[10] 0.3230292 0.3791764 0.3377964 0.3153966 0.3232070 0.1942700 0.3897975 0.3347691 0.2461111
[19] 0.3148124 0.3168662 0.3368341 0.3432556 0.3338962 0.2575004 0.3015635
 [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [1] 14.287129 21.138846  8.134223 11.652174 16.217760 23.987814  8.723214 15.433379  7.371795
[10]  5.354895  6.217949 10.392799 12.830025 17.256410  1.741758 19.580247 17.799700  4.924359
[19]  7.065611  5.858516 11.089510  9.917044  9.193182  7.915751 10.595745
[1] 284.6798

Step 23
clust
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23
101 100  49  46  68 101  56  56  36  22  33  47  62  69  14  81  77  30  34  28  55  51  44
 24  25
 42  47
 [1] 0.2283072 0.3878423 0.2664915 0.3595681 0.3520013 0.3283766 0.3229371 0.4070963 0.3272792
[10] 0.3230292 0.3791764 0.3377964 0.3153966 0.3232070 0.1942700 0.3897975 0.3347691 0.2461111
[19] 0.3148124 0.3168662 0.3368341 0.3432556 0.3338962 0.2575004 0.3015635
 [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [1] 14.287129 21.138846  8.134223 11.652174 16.217760 23.987814  8.723214 15.433379  7.371795
[10]  5.354895  6.217949 10.392799 12.830025 17.256410  1.741758 19.580247 17.799700  4.924359
[19]  7.065611  5.858516 11.089510  9.917044  9.193182  7.915751 10.595745
[1] 284.6798
Times repeat = 0
> save(rez,file="./cars2/cars25new.rez")
> hc <- hclustSO(rez$leaders)
> plot(hc,hang=-1)
> long[rez$clust==10]
 [1] "HYUNDAI GALLOPER 2.5 T/D-XL"         "JEEP WRANGLER 2.5 SPORT"
 [3] "JEEP WRANGLER 4.0 SPORT"             "JEEP WRANGLER 4.0 SPORT AUT"
 [5] "JEEP WRANGLER 4.0 SAHARA"            "JEEP WRANGLER 4.0 SAHARA AUT"
 [7] "KIA SPORTAGE 2.0i MRi"               "KIA SPORTAGE 2.0i MRDi"
 [9] "KIA SPORTAGE 2.0 TD INTERCOOLER"     "MITSUBISHI PAJERO 2.5 TD MT GL"
[11] "MITSUBISHI PAJERO 2.5 TD GLX"        "MITSUBISHI PAJERO 2.8 TD MT GLS"
[13] "MITSUBISHI PAJERO 3.0 V6-24 MT GLS" "MITSUBISHI PAJERO 3.0 V6-24 MT GLS"
[15] "MITSUBISHI PAJERO 3.5 V6-24 MT GLS" "NISSAN TERRANO II SR H/5"
[17] "NISSAN TERRANO II SE H/T"            "NISSAN TERRANO II TDi SR H/T"
[19] "NISSAN TERRANO II TDi SE H/T"        "SSANGYONG KORANDO FAMILY RS"
[21] "TOYOTA RAV 4"                        "TOYOTA RAV 4 GX"
```



notes/cars.txt · Last modified: 2012/11/02 02:25 by batagelj