

Example with dataset 'Nations'

Giulio Barcaroli

Created 26 Aug, 2018; Last Update 06 settembre, 2018

Dataset "nations"

- Data on 207 countries related to demographic variables

```
data(nations)
```

```
head(nations)
```

```
#>      Country  TFR  contraception  infant.mortality  GDP  region
#> 1  Afghanistan 6.90           63           154  2848    Asia
#> 2    Albania 2.60           47           32   863  Europe
#> 3    Algeria 3.81           52           44  1531  Africa
#> 4 American-Samoa 1.35          71           11  2433 Oceania
#> 5    Andorra 1.61           71            7 19121  Europe
#> 6    Angola 6.69           19          124   355  Africa
#> Continent
#> 1      2
#> 2      1
#> 3      4
#> 4      5
#> 5      1
#> 6      4
```

2/16

Step 1: derive 'sampling frame' from dataset

```
library(SamplingStrata)
frame <- buildFrameDF(nations,
                      id="Country",
                      X="Country",
                      Y=c("TFR", "contraception",
                          "infant.mortality", "GDP"),
                      domainvalue = "Continent")
```

```
head(frame)
```

```
#>           id           X1  Y1 Y2  Y3  Y4 domainvalue
#> 1  Afghanistan  Afghanistan 6.90 63 154 2848          2
#> 2    Albania    Albania 2.60 47  32  863          1
#> 3    Algeria    Algeria 3.81 52  44 1531          4
#> 4 American-Samoa American-Samoa 1.35 71  11 2433          5
#> 5    Andorra    Andorra 1.61 71   7 19121          1
#> 6    Angola    Angola 6.69 19 124  355          4
```

Step 2: derive 'strata' from the frame

```
strata <- buildStrataDF(frame, progress = FALSE)
```

```
#>
#> Computations are being done on population data
#>
#> Number of strata: 207
#> ... of which with only one unit: 207
```

```
head(strata)
```

```
#>   STRATO N   M1 M2 M3   M4 S1 S2 S3 S4 COST CENS DOM1   X1
#> 1 Albania 1 2.60 47 32   863 0 0 0 0   1   0   1 Albania
#> 2 Andorra 1 1.61 71  7 19121 0 0 0 0   1   0   1 Andorra
#> 3 Armenia 1 1.70 22 25   354 0 0 0 0   1   0   1 Armenia
#> 4 Austria 1 1.42 71  6 29006 0 0 0 0   1   0   1 Austria
#> 5 Belarus 1 1.40 50 15   994 0 0 0 0   1   0   1 Belarus
#> 6 Belgium 1 1.62 79  7 26582 0 0 0 0   1   0   1 Belgium
```

Step 3: definition of precision constraints

```
cv <- as.data.frame(list(DOM=rep("DOM1",5),  
                        CV1=rep(0.1,5),  
                        CV2=rep(0.1,5),  
                        CV3=rep(0.1,5),  
                        CV4=rep(0.1,5),  
                        domainvalue=c(1:5)  
                        ))
```

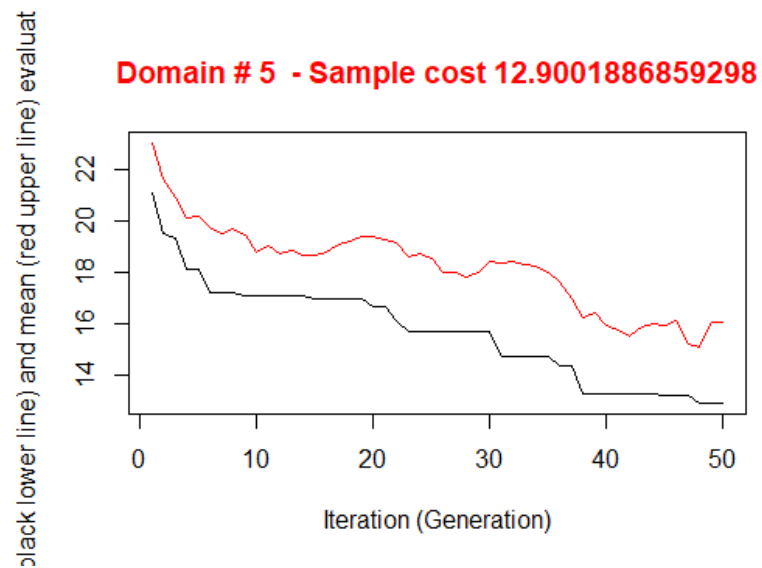
cv

```
#>   DOM CV1 CV2 CV3 CV4 domainvalue  
#> 1 DOM1 0.1 0.1 0.1 0.1          1  
#> 2 DOM1 0.1 0.1 0.1 0.1          2  
#> 3 DOM1 0.1 0.1 0.1 0.1          3  
#> 4 DOM1 0.1 0.1 0.1 0.1          4  
#> 5 DOM1 0.1 0.1 0.1 0.1          5
```

Step 4: Optimization

```
solution1 <-  
  optimizeStrata(  
    errors = cv ,  
    strata = strata,  
    iter = 50,  
    pops = 20,  
    suggestions = NULL,  
    showPlot = FALSE,  
    writeFiles = FALSE)
```

Solution



```
sum(ceiling(solution1$aggr_strata$SOLUZ))
```

```
#> [1] 89
```

```
nrow(solution1$aggr_strata)
```

```
#> [1] 30
```

7/16

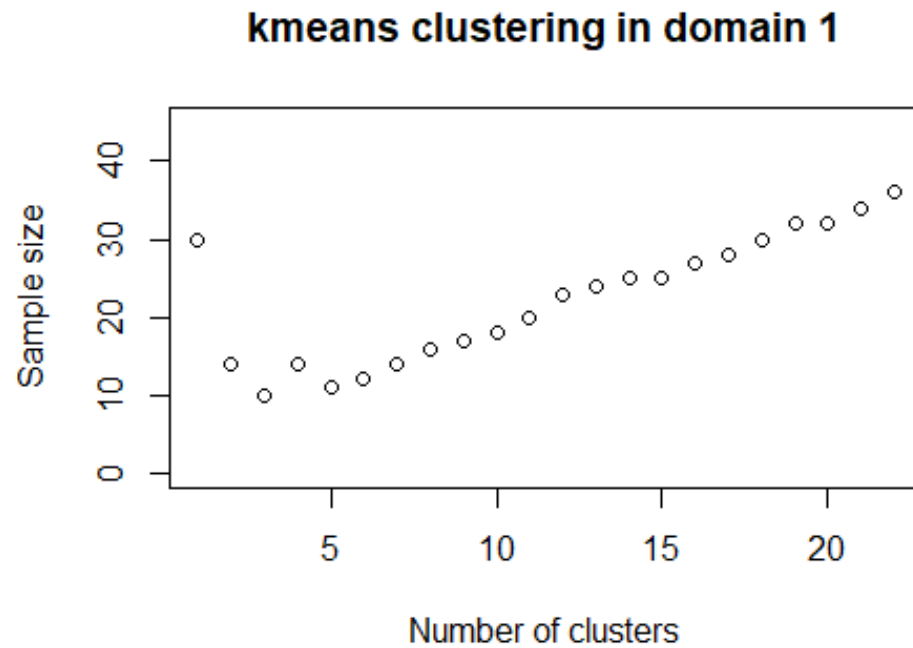
Initial suggestion with kmeans

```
kmean <- KmeansSolution(strata, cv, nstrata=NA, showPlot = F)
```

```
#>  
#> -----  
#> Kmeans solution  
#> -----  
#> *** Domain: 1 ***  
#> Number of strata: 3  
#> Sample size      : 10  
#> *** Domain: 2 ***  
#> Number of strata: 5  
#> Sample size      : 18  
#> *** Domain: 3 ***  
#> Number of strata: 5  
#> Sample size      : 17  
#> *** Domain: 4 ***  
#> Number of strata: 5  
#> Sample size      : 16  
#> *** Domain: 5 ***
```

8/16

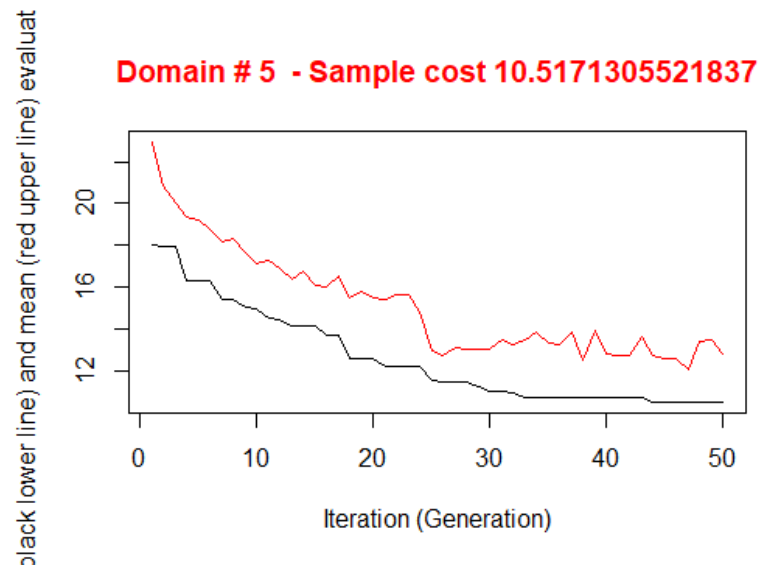
Best kmeans solution



Step 4: Optimization

```
solution2 <-  
  optimizeStrata(  
    errors = cv ,  
    strata = strata,  
    iter = 50,  
    pops = 20,  
    suggestions = kmean,  
    showPlot = FALSE,  
    writeFiles = FALSE)
```

New solution with initial suggestion



```
sum(ceiling(solution2$aggr_strata$SOLUZ))
```

```
#> [1] 64
```

```
nrow(solution2$aggr_strata)
```

```
#> [1] 23
```

11/16

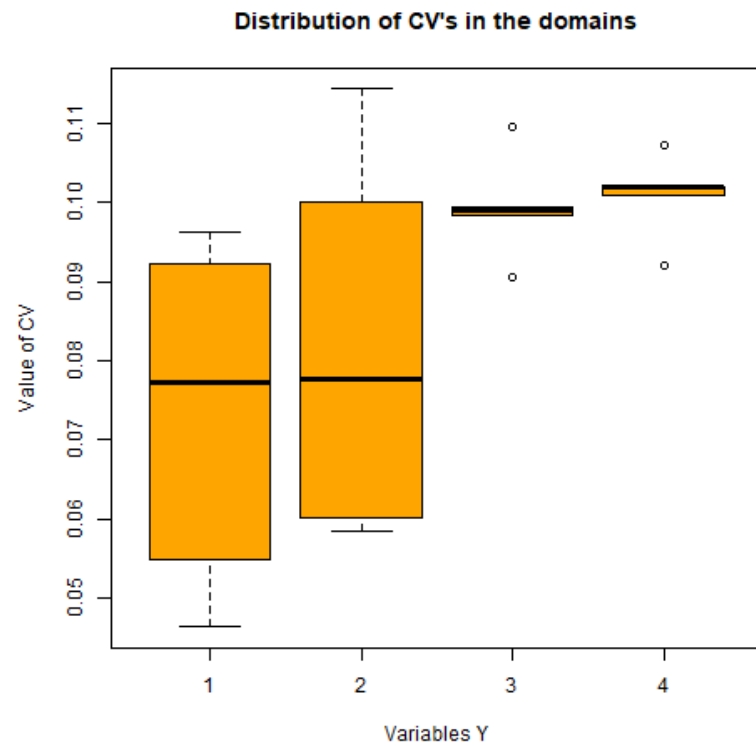
Step 5: expected CV

```
newstrata <- updateStrata(strata,solution2)
framenew <- updateFrame(frame,newstrata)
results1 <- evalSolution(framenew, solution2$aggr_strata,
                        200, writeFiles=TRUE)
```

```
results1$coeff_var
```

```
#>           CV1           CV2           CV3           CV4  dom
#> 1 0.05552746 0.05742911 0.09384126 0.09260974 DOM1
#> 2 0.09444563 0.10337950 0.10192588 0.10760970 DOM2
#> 3 0.07614528 0.06669145 0.10292034 0.11512929 DOM3
#> 4 0.04678365 0.11022906 0.09599446 0.10319698 DOM4
#> 5 0.09039865 0.08476337 0.09496739 0.09036526 DOM5
```

CV



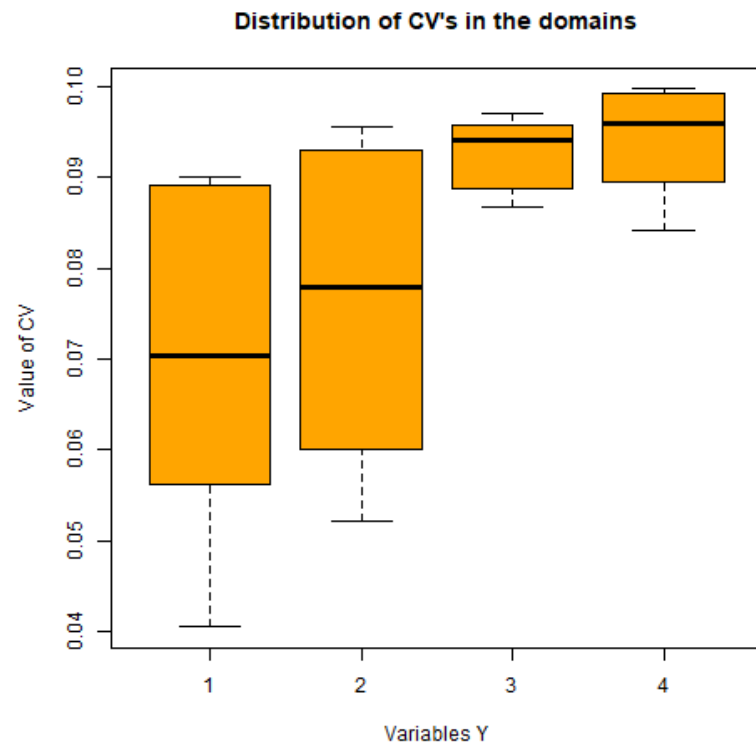
Improving expected CV

```
oustrata <- solution2$aggr_strata
outstrata$SOLUZ <- ceiling(solution2$aggr_strata$SOLUZ)
results2 <- evalSolution(framenew, outstrata, 200, writeFiles=TRUE)
```

```
results2$coeff_var
```

```
#>           CV1           CV2           CV3           CV4 dom
#> 1 0.05627154 0.06013496 0.09693417 0.09596132 DOM1
#> 2 0.08907188 0.09287718 0.08876856 0.08416040 DOM2
#> 3 0.07032160 0.05221197 0.09565112 0.09967724 DOM3
#> 4 0.04056805 0.09555357 0.08673278 0.09924053 DOM4
#> 5 0.09000523 0.07790927 0.09402760 0.08953323 DOM5
```

CV



Sample selection

Finally, we proceed in selecting the sample:

```
newstrata <- updateStrata(strata, solution2)
framenew <- updateFrame(frame,newstrata)
sample <- selectSample(frame=framenew,outstrata=solution2$aggr_strata)
```

```
#>
#> *** Sample has been drawn successfully ***
#> 57 units have been selected from 23 strata
#>
#> ==> There have been 5 take-all strata
#> from which have been selected 9 units
```

```
head(sample)
```

```
#>  DOMAINVALUE STRATO STRATUM      ID      X1      Y1 Y2 Y3      Y4 LABEL
#> 1             1         1 Georgia Georgia Georgia 1.90 17 23      343      1
#> 2             1         1  Bosnia  Bosnia  Bosnia 1.40 62 13      271      1
```

16/16