Example with dataset ‘Nations’

Giulio Barcaroli

Created 26 Aug, 2018; Last Update 11 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
```

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
```

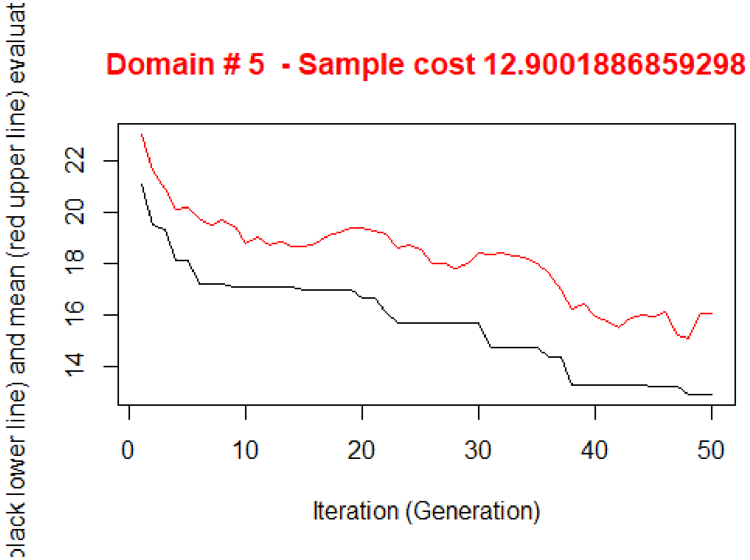
```
#>    DOM CV1 CV2 CV3 CV4 domainvalue
#> 1 DOM1 0.1 0.1 0.1 0.1           1
#> 2 DOM1 0.1 0.1 0.1 0.1           2
#> 3 DOM1 0.1 0.1 0.1 0.1           3
#> 4 DOM1 0.1 0.1 0.1 0.1           4
#> 5 DOM1 0.1 0.1 0.1 0.1           5
```

Step 4: Optimization

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

```
#>
#>  *** Domain :  1   1
#>  Number of strata :  45
#> ---------------------------------------------
#> Optimal stratification with Genetic Algorithm
#> ---------------------------------------------
#>  *** Parameters ***
#> ---------------------------
#> Domain:  1
```

Solution



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

```
#> [1] 84
```

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

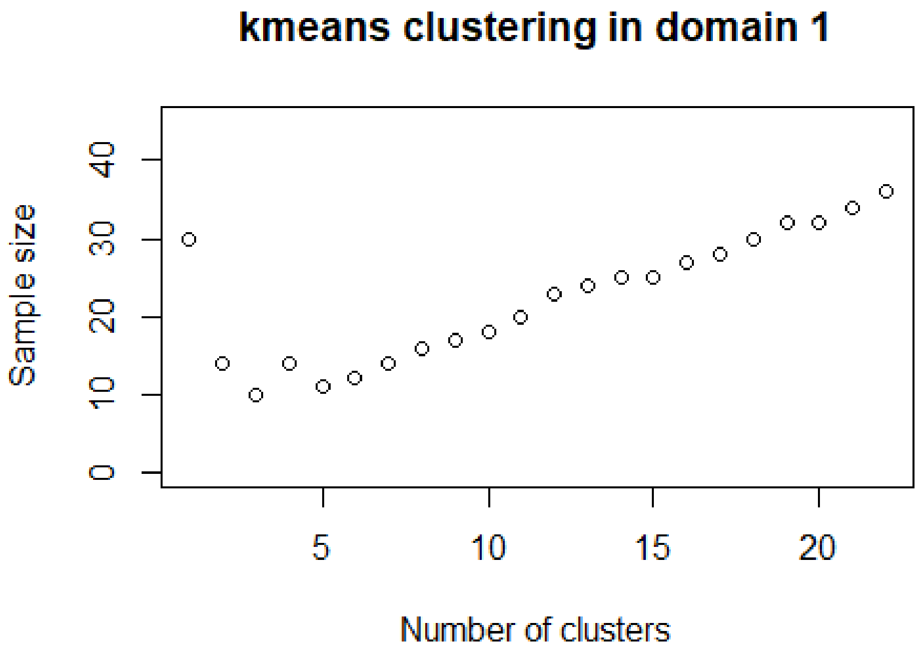
```
#> [1] 32
```

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  ***
```

Best kmeans solution

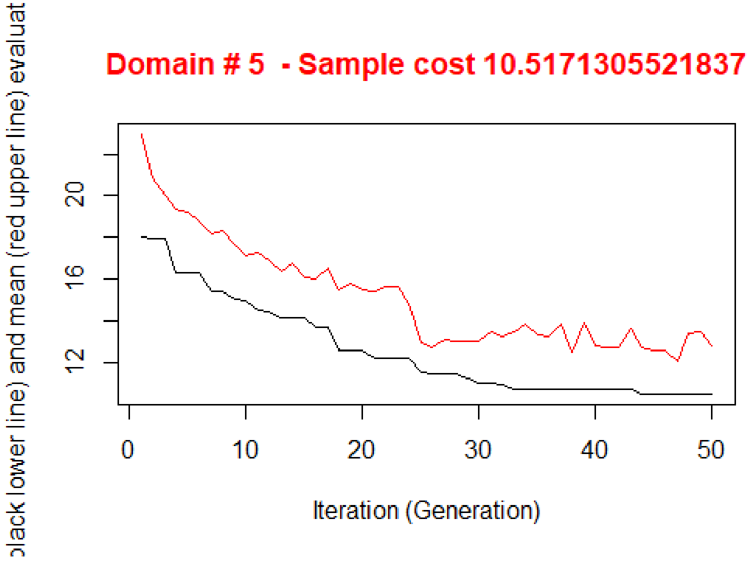


Step 4: Optimization

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

```
#>
#>  *** Domain :  1   1
#>  Number of strata :  45
#> ---------------------------------------------
#> Optimal stratification with Genetic Algorithm
#> ---------------------------------------------
#>  *** Parameters ***
#> ---------------------------
#> Domain:  1
```

New solution with initial suggestion



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

```
#> [1] 62
```

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

```
#> [1] 21
```

11/15

Analysis of aggregated strata

```
newstrata <- updateStrata(strata,solution2,writeFiles = T)
aggr <- read.delim("strata_aggregation.txt")
aggr$X1 <- as.factor(aggr$X1)
levels(aggr$X1) <- levels(frame$X1)
head(aggr)
```

```
#>   DOM1 AGGR_STRATUM       X1
#> 1    1            1  Albania
#> 2    1            1  Armenia
#> 3    1            1  Belarus
#> 4    1            1   Bosnia
#> 5    1            1 Bulgaria
#> 6    1            1  Estonia
```

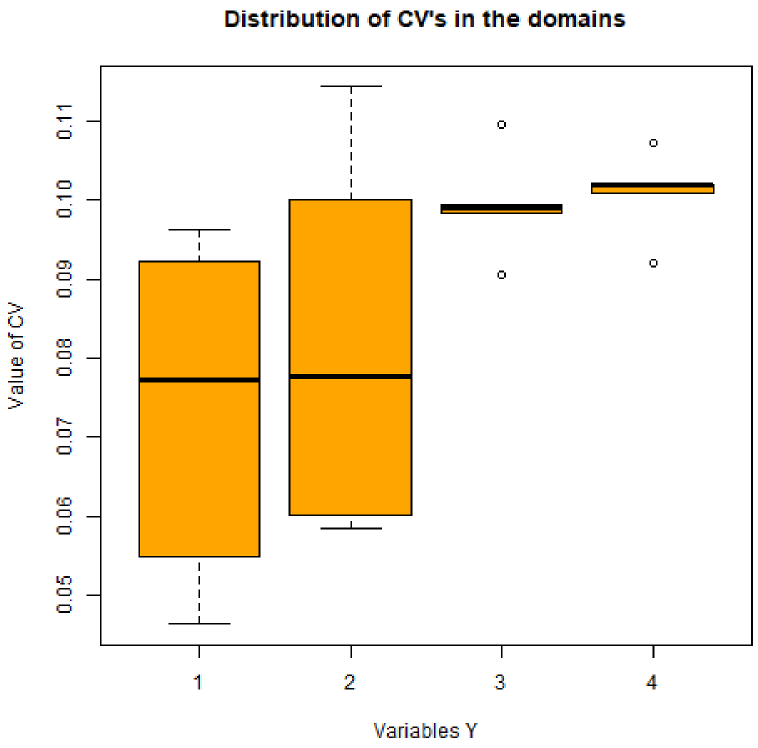
Step 5: expected CV

```
framenew <- updateFrame(frame,newstrata)
results1 <- evalSolution(framenew, solution2$aggr_strata,
                          100, writeFiles=TRUE, progress=FALSE)
```

```
results1$coeff_var
```

```
#>           CV1        CV2        CV3        CV4  dom
#> 1 0.05986399 0.07655969 0.11472746 0.11026417 DOM1
#> 2 0.10415993 0.10106564 0.11298257 0.11176035 DOM2
#> 3 0.06438737 0.05509381 0.10599716 0.11494179 DOM3
#> 4 0.05171360 0.10614111 0.09595126 0.10705901 DOM4
#> 5 0.09647061 0.08817363 0.09757728 0.09014078 DOM5
```

CV


Distribution of CV's in the domains
Value of CV
0.05
0.06
0.07
0.08
0.09
0.10
0.11
1
2
3
4
Variables Y

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 ***
#>  55  units have been selected from  21  strata
#> 
#> ==> There have been  2  take-all strata 
#> from which have been selected  3 units
```

```
head(sample)
```

```
#>   DOMAINVALUE STRATO    STRATUM        ID        X1   Y1 Y2 Y3    Y4
#> 1           1      1    Ukraine    Ukraine    Ukraine 1.38 23 18   694
#> 2           1      1    Moldova    Moldova    Moldova 1.80 22 26   383
```