Workflow examples with R2BEAT

Scenario 1

Only a sampling frame containing the units of the population of reference is available, no previous round of the sampling survey to be planned

In [1]:
```
# Install last version of R2BEAT and ReGenesees
#install.packages("devtools")
#devtools::install_github("DiegoZardetto/ReGenesees",dependencies = FALSE)
#devtools::install_github("barcaroli/R2BEAT",dependencies=FALSE)
library("R2BEAT")
```

```
Caricamento del pacchetto richiesto: devtools

Caricamento del pacchetto richiesto: usethis

Caricamento del pacchetto richiesto: sampling

Caricamento del pacchetto richiesto: glue
```

In [2]:
```
packageVersion("R2BEAT")
```

```
[1] '1.0.4'
```

In [3]:
```
## Sampling frame
load("pop.RData")
str(pop)
```

```
'data.frame':   2258507 obs. of  13 variables:
 $ region       : Factor w/ 3 levels "north","center",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ province     : Factor w/ 6 levels "north_1","north_2",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ municipality : num  1 1 1 1 1 1 1 1 1 1 ...
 $ id_hh        : Factor w/ 963018 levels "H1","H10","H100",..: 1 1 1 2 3 3 3 3 1114
1114 ...
 $ id_ind       : int  1 2 3 4 5 6 7 8 9 10 ...
 $ stratum      : Factor w/ 24 levels "1000","2000",..: 12 12 12 12 12 12 12 12 12 1
2 ...
 $ stratum_label: chr  "north_1_6" "north_1_6" "north_1_6" "north_1_6" ...
 $ sex          : int  1 2 1 2 1 1 2 2 1 1 ...
 $ cl_age       : Factor w/ 8 levels "(0,14]","(14,24]",..: 3 7 8 5 4 6 6 4 4 1 ...
 $ active       : num  1 1 0 1 1 1 1 1 1 0 ...
 $ income_hh    : num  30488 30488 30488 21756 29871 ...
 $ unemployed   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ inactive     : num  0 0 1 0 0 0 0 0 0 1 ...
```

Precision constraints

In [4]:
```
cv <- as.data.frame(list(DOM=c("DOM1","DOM2"),
                         CV1=c(0.02,0.03),
                         CV2=c(0.03,0.06),
                         CV3=c(0.03,0.06),
                         CV4=c(0.05,0.08)))
cv
```

A data.frame: 2 × 5

DOM	CV1	CV2	CV3	CV4
<chr>	<dbl>	<dbl>	<dbl>	<dbl>
DOM1	0.02	0.03	0.03	0.05
DOM2	0.03	0.06	0.06	0.08

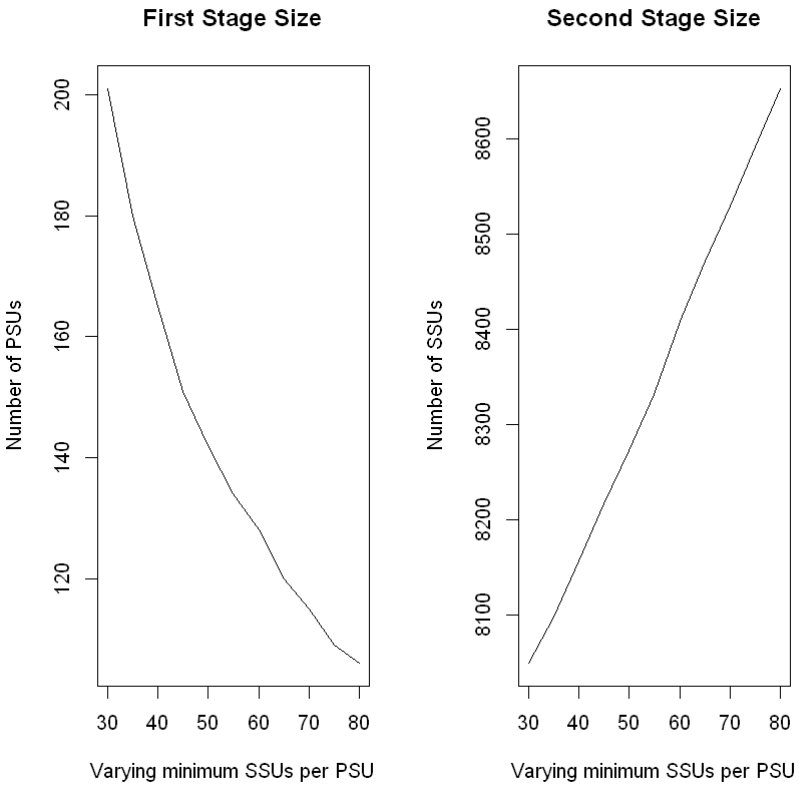
Sensitivity analysis

In [5]:

```
sens_min_SSU <- sensitivity_min_SSU (
              samp_frame=pop,
              errors=cv,
              id_PSU="municipality",
              id_SSU="id_ind",
              strata_var="stratum",
              target_vars=c("income_hh","active","inactive","unemployed"),
              deff_var="stratum",
              domain_var="region",
              delta=1,
              deff_sugg=1,
              min=30,
              max=80,
              plot=TRUE)
```

```
Calculating strata...
Computations are being done on population data

Number of strata:  24
... of which with only one unit:  0
Calculating rho in strata...
Stratum  1000
Stratum  2000
Stratum  3000
Stratum  4000
Stratum  5000
Stratum  6000
Stratum  7000
Stratum  8000
Stratum  9000
Stratum  10000
Stratum  11000
Stratum  12000
Stratum  13000
Stratum  14000
Stratum  15000
Stratum  16000
Stratum  17000
Stratum  18000
Stratum  19000
Stratum  20000
Stratum  21000
Stratum  22000
Stratum  23000
Stratum  24000
 1  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     81     114       195 8049
3          2     83     118       201 8049

 2  iterations PSU_SR PSU NSR PSU Total  SSU
```

```
1          0      0       0         0 7836
2          1     67     110       177 8099
3          2     68     112       180 8098

 3  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     49     110       159 8164
3          2     53     112       165 8157

 4  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     37     110       147 8222
3          2     43     108       151 8217

 5  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     31     104       135 8281
3          2     39     104       143 8271
4          3     38     104       142 8273

 6  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     29      98       127 8343
3          2     34     100       134 8332

 7  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     26      96       122 8406
3          2     32      96       128 8408

 8  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     26      86       112 8475
3          2     28      92       120 8472

 9  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     25      82       107 8545
3          2     29      86       115 8530

 10  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     20      80       100 8596
3          2     25      84       109 8592

 11  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7836
2          1     22      74        96 8662
3          2     20      86       106 8653
```



Preparation of inputs for allocation steps

In [6]:

```
## Preparation of inputs for allocation steps
samp_frame <- pop
samp_frame$one <- 1
id_PSU <- "municipality"
id_SSU <- "id_ind"
strata_var <- "stratum"
target_vars <- c("income_hh","active","inactive","unemployed")
deff_var <- "stratum"
domain_var <- "region"
delta =  1       # households = survey units
minimum <- 50    # minimum number of SSUs to be interviewed in each selected PSU
deff_sugg <- 1.5 # suggestion for the deff value

inp1 <- prepareInputToAllocation1(samp_frame,
                                  id_PSU,
                                  id_SSU,
                                  strata_var,
                                  target_vars,
                                  deff_var,
                                  domain_var,
                                  minimum,
                                  delta,
                                  deff_sugg)
```

```
Calculating strata...
Computations are being done on population data

Number of strata:  24
... of which with only one unit:  0
Calculating rho in strata...
Stratum  1000
Stratum  2000
Stratum  3000
Stratum  4000
```

```
Stratum  5000
Stratum  6000
Stratum  7000
Stratum  8000
Stratum  9000
Stratum  10000
Stratum  11000
Stratum  12000
Stratum  13000
Stratum  14000
Stratum  15000
Stratum  16000
Stratum  17000
Stratum  18000
Stratum  19000
Stratum  20000
Stratum  21000
Stratum  22000
Stratum  23000
Stratum  24000
```

In [7]:

```
head(inp1$strata)
```

A data.frame: 6 × 14

	N	M1	M2	M3	M4	S1	S2	S3	S4
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1000	197007	23959.87	0.6650322	0.2285807	0.10638708	22179.08	0.4719792	0.4199185	0.3083324
2000	261456	20966.65	0.6709886	0.2297519	0.09925953	19624.65	0.4698541	0.4206732	0.2990102
3000	115813	19814.73	0.6644591	0.2315975	0.10394343	14754.88	0.4721792	0.4218532	0.3051871
4000	17241	18732.72	0.6273418	0.2499275	0.12273070	13462.74	0.4835122	0.4329708	0.3281278
5000	101067	22070.31	0.6134445	0.2338845	0.15267100	17187.98	0.4869603	0.4232996	0.3596701
6000	47218	21069.07	0.6135796	0.2348469	0.15157355	17342.74	0.4869288	0.4239031	0.3586070

In [8]:

```
head(inp1$deff)
```

A data.frame: 6 × 6

	STRATUM	DEFF1	DEFF2	DEFF3	DEFF4	b_nar
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1000	1.5	1.5	1.5	1.5	4925.17500
12	2000	1.5	1.5	1.5	1.5	1005.60000
18	3000	1.5	1.5	1.5	1.5	222.71731
19	4000	1.5	1.5	1.5	1.5	47.89167
20	5000	1.5	1.5	1.5	1.5	2526.67500
21	6000	1.5	1.5	1.5	1.5	786.96667

In [9]:

```
head(inp1$effst)
```

A data.frame: 6 × 5

	STRATUM	EFFST1	EFFST2	EFFST3	EFFST4
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>
1	1000	1	1	1	1
2	2000	1	1	1	1
3	3000	1	1	1	1
4	4000	1	1	1	1
5	5000	1	1	1	1
6	6000	1	1	1	1

In [10]:
```
head(inp1$rho)
```

A data.frame: 6 × 9

	STRATUM	RHO_AR1	RHO_NAR1	RHO_AR2	RHO_NAR2	RHO_AR3	RHO_NAR3	RH
	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	
1	1000	1	0.0032494875	1	0.00001260175649	1	0.0000003631192	
2	2000	1	0.0028554017	1	0.00150936389450	1	0.0007420929883	
3	3000	1	0.0069678726	1	0.00162968276279	1	0.0006469515878	
4	4000	1	0.0114552934	1	0.00578473329221	1	0.0019797687826	
5	5000	1	0.0002677333	1	0.00000001682475	1	0.0000029484212	
6	6000	1	0.0057050500	1	0.00004270905958	1	0.0000397945795	

In [11]:
```
head(inp1$psu_file)
```

A data.frame: 6 × 3

	PSU_ID	STRATUM	PSU_MOS
	<dbl>	<fct>	<dbl>
1	1	12000	1546
2	2	12000	936
3	3	12000	367
4	4	10000	13032
5	5	12000	678
6	6	11000	3193

In [12]:
```
head(inp1$des_file)
```

A data.frame: 6 × 4

	STRATUM	STRAT_MOS	DELTA	MINIMUM
	<fct>	<dbl>	<dbl>	<dbl>
1	1000	197007	1	50

	STRATUM	STRAT_MOS	DELTA	MINIMUM
	<fct>	<dbl>	<dbl>	<dbl>
2	2000	261456	1	50
3	3000	115813	1	50
4	4000	17241	1	50
5	5000	101067	1	50
6	6000	47218	1	50

Allocation

In [13]:

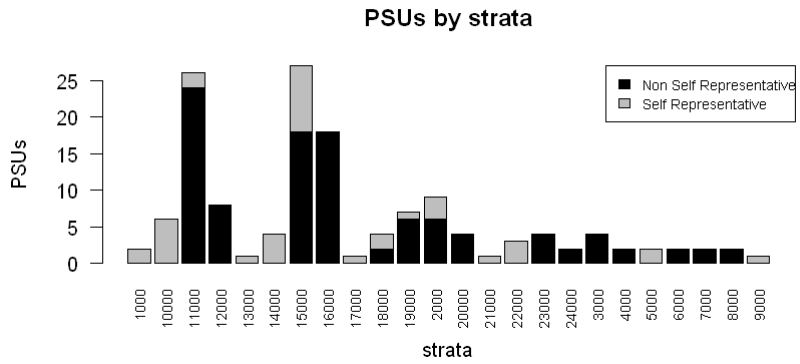
```
inp1$desfile$MINIMUM <- 50
alloc1 <- beat.2st(stratif = inp1$strata,
                   errors = cv,
                   des_file = inp1$des_file,
                   psu_file = inp1$psu_file,
                   rho = inp1$rho,
                   deft_start = NULL,
                   effst = inp1$effst,
                   minPSUstrat = 2,
                   minnumstrat = 50
                   )
```

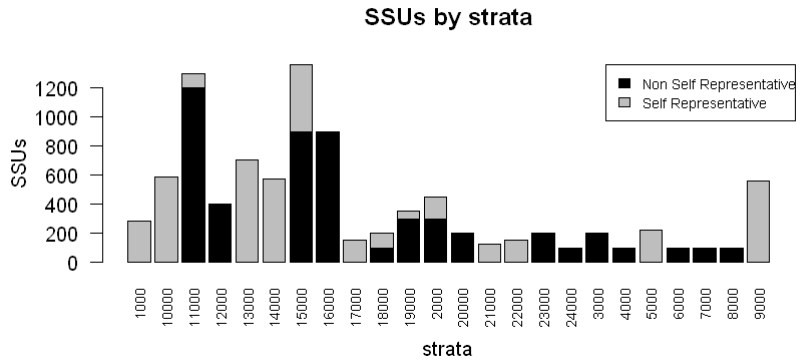
```
  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7887
2          1     31     104       135 8328
3          2     39     104       143 8317
4          3     38     104       142 8320
```

Selection of PSUs (I stage)

In [14]:

```
set.seed(1234)
sample_1st <- select_PSU(alloc1, type="ALLOC", pps=TRUE)
```





In [15]:

```
sample_1st$PSU_stats
```

A data.frame: 25 × 7

STRATUM	PSU	PSU_SR	PSU_NSR	SSU	SSU_SR	SSU_NSR
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1000	2	2	0	286	286	0
2000	9	3	6	452	152	300
3000	4	0	4	200	0	200
4000	2	0	2	100	0	100
5000	2	2	0	219	219	0
6000	2	0	2	100	0	100
7000	2	0	2	100	0	100
8000	2	0	2	100	0	100
9000	1	1	0	557	557	0
10000	6	6	0	587	587	0
11000	26	2	24	1300	100	1200
12000	8	0	8	400	0	400
13000	1	1	0	703	703	0
14000	4	4	0	577	577	0
15000	27	9	18	1361	461	900
16000	18	0	18	900	0	900
17000	1	1	0	154	154	0

STRATUM	PSU	PSU_SR	PSU_NSR	SSU	SSU_SR	SSU_NSR
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
18000	4	2	2	200	100	100
19000	7	1	6	350	50	300
20000	4	0	4	200	0	200
21000	1	1	0	125	125	0
22000	3	3	0	150	150	0
23000	4	0	4	200	0	200
24000	2	0	2	100	0	100
Total	142	38	104	9421	4221	5200

Selection of SSUs (II stage)

In [16]:

```
samp <- select_SSU(df=pop,
                   PSU_code="municipality",
                   SSU_code="id_ind",
                   PSU_sampled=sample_1st$sample_PSU,
                   verbose=TRUE)
```

```
PSU =  1  *** Selected SSU =  50
PSU =  4  *** Selected SSU =  72
PSU =  6  *** Selected SSU =  50
PSU =  8  *** Selected SSU =  557
PSU =  11  *** Selected SSU =  105
PSU =  13  *** Selected SSU =  79
PSU =  15  *** Selected SSU =  50
PSU =  16  *** Selected SSU =  50
PSU =  17  *** Selected SSU =  50
PSU =  18  *** Selected SSU =  50
PSU =  21  *** Selected SSU =  50
PSU =  24  *** Selected SSU =  50
PSU =  29  *** Selected SSU =  50
PSU =  33  *** Selected SSU =  50
PSU =  34  *** Selected SSU =  50
PSU =  36  *** Selected SSU =  50
PSU =  37  *** Selected SSU =  50
PSU =  40  *** Selected SSU =  87
PSU =  41  *** Selected SSU =  58
PSU =  51  *** Selected SSU =  186
PSU =  56  *** Selected SSU =  50
PSU =  59  *** Selected SSU =  50
PSU =  61  *** Selected SSU =  50
PSU =  67  *** Selected SSU =  50
PSU =  72  *** Selected SSU =  50
PSU =  80  *** Selected SSU =  50
PSU =  82  *** Selected SSU =  50
PSU =  85  *** Selected SSU =  50
PSU =  89  *** Selected SSU =  50
PSU =  90  *** Selected SSU =  50
PSU =  91  *** Selected SSU =  50
PSU =  92  *** Selected SSU =  50
PSU =  95  *** Selected SSU =  50
PSU =  98  *** Selected SSU =  50
PSU =  102  *** Selected SSU =  50
PSU =  104  *** Selected SSU =  50
```

```
PSU =  105  *** Selected SSU =  50
PSU =  109  *** Selected SSU =  50
PSU =  110  *** Selected SSU =  50
PSU =  112  *** Selected SSU =  50
PSU =  113  *** Selected SSU =  50
PSU =  117  *** Selected SSU =  52
PSU =  121  *** Selected SSU =  107
PSU =  122  *** Selected SSU =  50
PSU =  123  *** Selected SSU =  50
PSU =  126  *** Selected SSU =  50
PSU =  130  *** Selected SSU =  50
PSU =  138  *** Selected SSU =  50
PSU =  142  *** Selected SSU =  50
PSU =  155  *** Selected SSU =  50
PSU =  156  *** Selected SSU =  50
PSU =  161  *** Selected SSU =  50
PSU =  162  *** Selected SSU =  50
PSU =  170  *** Selected SSU =  50
PSU =  175  *** Selected SSU =  50
PSU =  180  *** Selected SSU =  50
PSU =  182  *** Selected SSU =  50
PSU =  186  *** Selected SSU =  52
PSU =  187  *** Selected SSU =  50
PSU =  189  *** Selected SSU =  50
PSU =  192  *** Selected SSU =  50
PSU =  197  *** Selected SSU =  50
PSU =  198  *** Selected SSU =  50
PSU =  201  *** Selected SSU =  57
PSU =  210  *** Selected SSU =  50
PSU =  211  *** Selected SSU =  50
PSU =  214  *** Selected SSU =  130
PSU =  221  *** Selected SSU =  103
PSU =  223  *** Selected SSU =  50
PSU =  224  *** Selected SSU =  50
PSU =  228  *** Selected SSU =  237
PSU =  233  *** Selected SSU =  50
PSU =  235  *** Selected SSU =  50
PSU =  238  *** Selected SSU =  50
PSU =  239  *** Selected SSU =  50
PSU =  241  *** Selected SSU =  50
PSU =  243  *** Selected SSU =  50
PSU =  246  *** Selected SSU =  50
PSU =  248  *** Selected SSU =  50
PSU =  251  *** Selected SSU =  50
PSU =  256  *** Selected SSU =  50
PSU =  259  *** Selected SSU =  703
PSU =  269  *** Selected SSU =  50
PSU =  270  *** Selected SSU =  50
PSU =  272  *** Selected SSU =  50
PSU =  275  *** Selected SSU =  50
PSU =  276  *** Selected SSU =  50
PSU =  283  *** Selected SSU =  50
PSU =  288  *** Selected SSU =  50
PSU =  290  *** Selected SSU =  50
PSU =  291  *** Selected SSU =  50
PSU =  292  *** Selected SSU =  50
PSU =  294  *** Selected SSU =  50
PSU =  302  *** Selected SSU =  50
PSU =  304  *** Selected SSU =  52
PSU =  306  *** Selected SSU =  50
PSU =  309  *** Selected SSU =  74
PSU =  311  *** Selected SSU =  50
PSU =  314  *** Selected SSU =  50
PSU =  315  *** Selected SSU =  50
PSU =  317  *** Selected SSU =  50
```

```
PSU =  324  *** Selected SSU =  50
PSU =  325  *** Selected SSU =  50
PSU =  330  *** Selected SSU =  212
PSU =  341  *** Selected SSU =  50
PSU =  342  *** Selected SSU =  50
PSU =  343  *** Selected SSU =  50
PSU =  347  *** Selected SSU =  50
PSU =  351  *** Selected SSU =  50
PSU =  363  *** Selected SSU =  50
PSU =  367  *** Selected SSU =  50
PSU =  370  *** Selected SSU =  50
PSU =  372  *** Selected SSU =  50
PSU =  373  *** Selected SSU =  50
PSU =  380  *** Selected SSU =  50
PSU =  382  *** Selected SSU =  169
PSU =  387  *** Selected SSU =  50
PSU =  390  *** Selected SSU =  50
PSU =  421  *** Selected SSU =  50
PSU =  425  *** Selected SSU =  50
PSU =  427  *** Selected SSU =  50
PSU =  439  *** Selected SSU =  50
PSU =  443  *** Selected SSU =  50
PSU =  445  *** Selected SSU =  154
PSU =  448  *** Selected SSU =  50
PSU =  452  *** Selected SSU =  50
PSU =  471  *** Selected SSU =  50
PSU =  474  *** Selected SSU =  50
PSU =  475  *** Selected SSU =  50
PSU =  476  *** Selected SSU =  50
PSU =  477  *** Selected SSU =  50
PSU =  480  *** Selected SSU =  50
PSU =  485  *** Selected SSU =  50
PSU =  492  *** Selected SSU =  50
PSU =  494  *** Selected SSU =  50
PSU =  496  *** Selected SSU =  125
PSU =  499  *** Selected SSU =  50
PSU =  502  *** Selected SSU =  50
PSU =  503  *** Selected SSU =  50
PSU =  504  *** Selected SSU =  50
PSU =  510  *** Selected SSU =  50
PSU =  512  *** Selected SSU =  50
--------------------------------
Total PSU =  142
Total SSU =  9421
--------------------------------
```

In [17]:

```
nrow(samp)
sum(alloc1$alloc$ALLOC[-nrow(alloc1$alloc)])
```

9421

8320

In [18]:

```
nrow(pop)
sum(samp$weight)
```

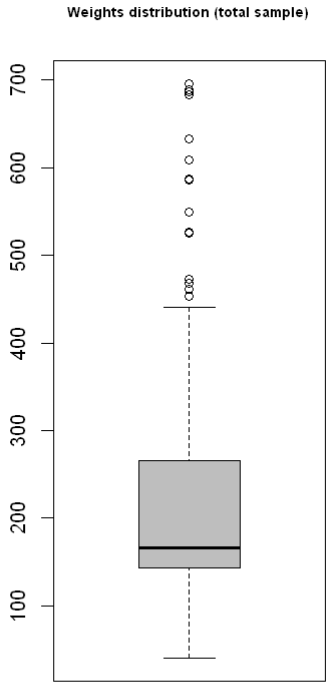
2258507

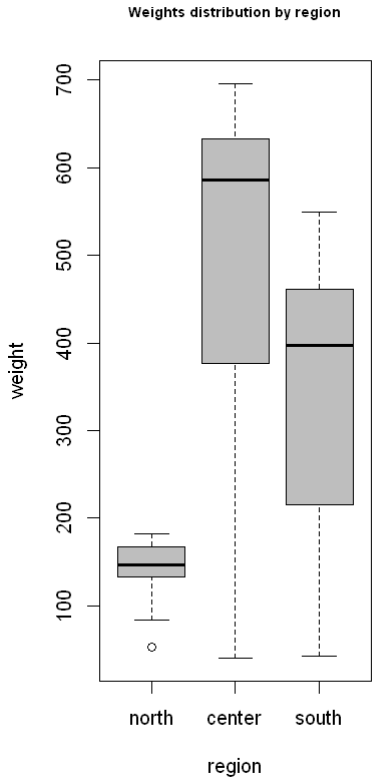
2258507

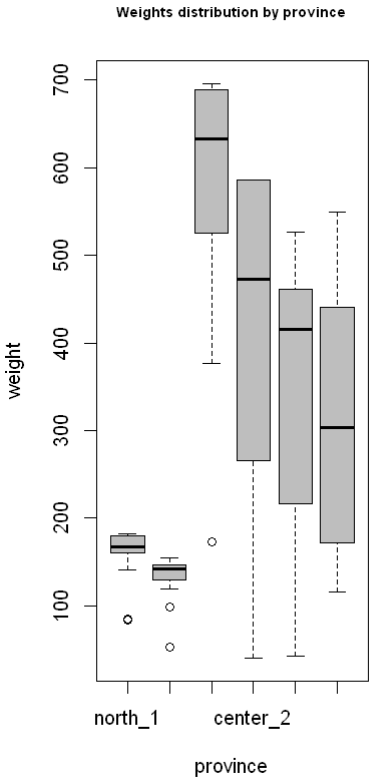
In [19]:

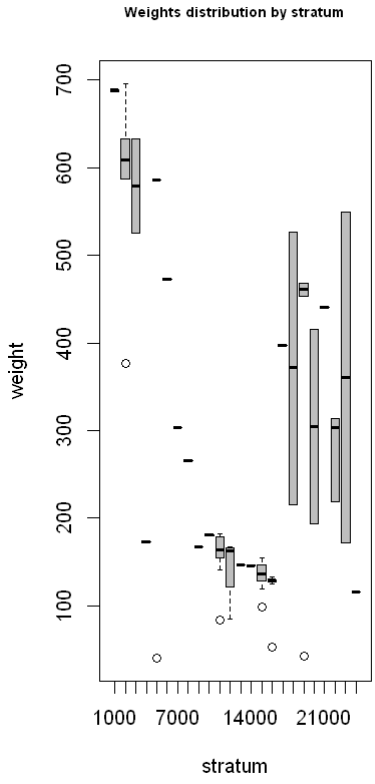
```
## Plot of weights distribution
par(mfrow=c(1, 2))
boxplot(samp$weight,col="grey")
title("Weights distribution (total sample)",cex.main=0.7)
```

```
boxplot(weight ~ region, data=samp,col="grey")
title("Weights distribution by region",cex.main=0.7)
par(mfrow=c(1, 2))
boxplot(weight ~ province, data=samp,col="grey")
title("Weights distribution by province",cex.main=0.7)
boxplot(weight ~ stratum, data=samp,col="grey")
title("Weights distribution by stratum",cex.main=0.7)
```









Precision constraints compliance control (by

simulation)

In [20]:
```
df=pop
df$one <- 1
PSU_code="municipality"
SSU_code="id_ind"
target_vars <- c("income_hh",
                 "active",
                 "inactive",
                 "unemployed")
```

In [21]:
```
# Domain level = national
domain_var <- "one"
set.seed(1234)
eval11 <- eval_2stage(df,
                      PSU_code,
                      SSU_code,
                      domain_var,
                      target_vars,
                      sample_1st$sample_PSU,
                      nsampl=100,
                      writeFiles=FALSE,
                      progress=TRUE)
eval11$coeff_var
```

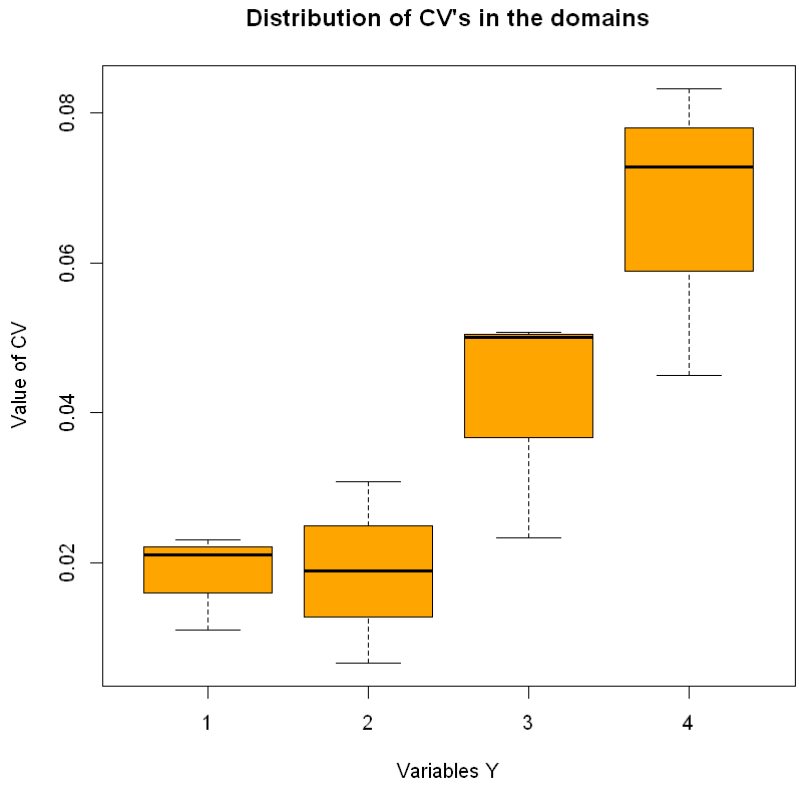
```
  |======================================================================| 100%
```

A data.frame: 1 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0091	0.0086	0.0238	0.0346	DOM1

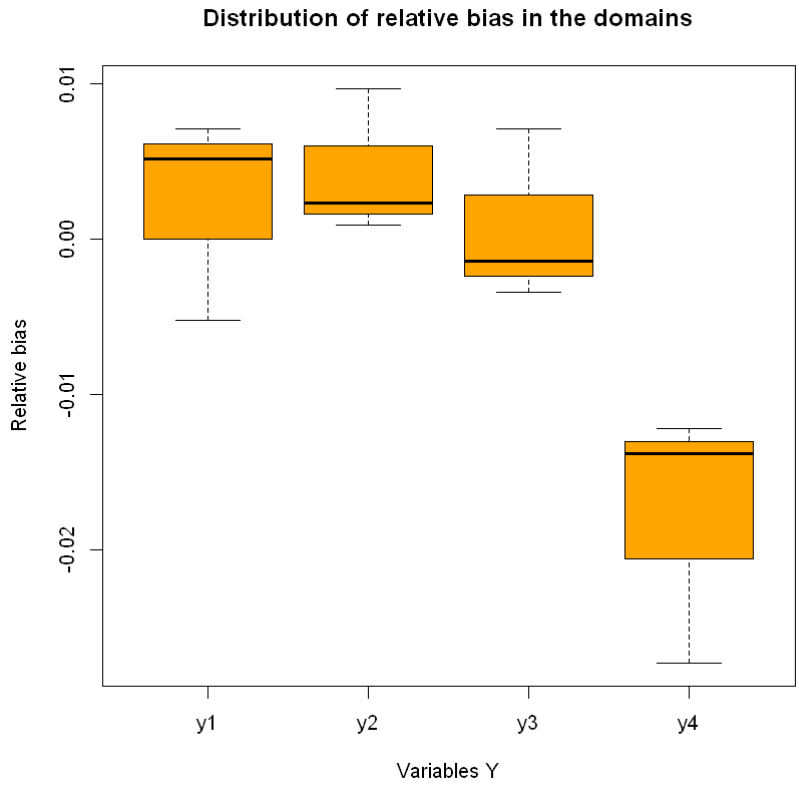
In [22]:
```
# Domain level = regional
domain_var <- "region"
set.seed(1234)
set.seed(1234)
eval12 <- eval_2stage(df,
                      PSU_code,
                      SSU_code,
                      domain_var,
                      target_vars,
                      sample_1st$sample_PSU,
                      nsampl=100,
                      writeFiles=FALSE,
                      progress=TRUE)
eval12$coeff_var
```

```
  |======================================================================| 100%
```



A data.frame: 3 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0109	0.0066	0.0234	0.0833	DOM1
0.0210	0.0189	0.0500	0.0728	DOM2
0.0231	0.0309	0.0508	0.0450	DOM3



In [23]:
```
alloc1$sensitivity
```

A data.frame: 4 × 6

	Type	Dom	V1	V2	V3	V4
	<chr>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
2	DOM1	1	1	0	1	1
6	DOM2	1	1	0	1	1181
10	DOM2	2	1	0	1	245
14	DOM2	3	184	1	45	1

In [24]:
```
save(samp,file="sample.RData")
```

Scenario 2

One previous round of the sampling survey is available.

Analysis of sampled data

In [25]:
```
library(ReGenesees)
```

```
------------------------------------------------------------

> The ReGenesees package has been successfully loaded. <

------------------------------------------------------------
```

```
Package: ReGenesees
Type: Package
Title: R Evolved Generalized Software for Sampling Estimates and Errors
        in Surveys
Description: Design-Based and Model-Assisted analysis of complex
        sampling surveys. Multistage, stratified, clustered, unequally
        weighted survey designs. Horvitz-Thompson and Calibration
        Estimators. Variance Estimation for nonlinear smooth estimators
        by Taylor-series linearization. Estimates, standard errors,
        confidence intervals and design effects for: Totals, Means,
        absolute and relative Frequency Distributions (marginal,
        conditional and joint), Ratios, Shares and Ratios of Shares,
        Multiple Regression Coefficients and Quantiles. Automated
        Linearization of Complex Analytic Estimators. Design Covariance
        and Correlation. Estimates, standard errors, confidence
        intervals and design effects for user-defined analytic
        estimators. Estimates and sampling errors for subpopulations.
        Consistent trimming of calibration weights. Calibration on
        complex population parameters, e.g. multiple regression
        coefficients. Generalized Variance Functions (GVF) method for
        predicting variance estimates.
```

```
Version: 2.1
Author: Diego Zardetto [aut, cre]
Maintainer: Diego Zardetto <zardetto@istat.it>
Authors@R: person("Diego", "Zardetto", role = c("aut", "cre"), email =
        "zardetto@istat.it")
License: EUPL
URL: https://diegozardetto.github.io/ReGenesees/,
        https://github.com/DiegoZardetto/ReGenesees/
BugReports: https://github.com/DiegoZardetto/ReGenesees/issues/
Imports: stats, MASS
Depends: R (>= 2.14.0)
ByteCompile: TRUE
RemoteType: github
RemoteHost: api.github.com
RemoteRepo: ReGenesees
RemoteUsername: DiegoZardetto
RemoteRef: HEAD
RemoteSha: c0bd789ed6ab88a4b3a02bd553f51d8f4ec857e2
GithubRepo: ReGenesees
GithubUsername: DiegoZardetto
GithubRef: HEAD
GithubSHA1: c0bd789ed6ab88a4b3a02bd553f51d8f4ec857e2
NeedsCompilation: no
Packaged: 2021-09-28 11:33:29 UTC; Giulio
Built: R 4.1.1; ; 2021-09-28 11:33:35 UTC; windows
```

In [26]:

```r
load("sample.RData")
str(samp)
```

```
'data.frame':   9421 obs. of  20 variables:
 $ municipality : Factor w/ 142 levels "1","4","6","8",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ id_ind       : int  11 19 34 67 141 166 170 191 208 308 ...
 $ region       : Factor w/ 3 levels "north","center",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ province     : Factor w/ 6 levels "north_1","north_2",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ id_hh        : Factor w/ 963018 levels "H1","H10","H100",..: 1114 4447 10002 2000
2 45558 52224 54446 58891 63335 95557 ...
 $ stratum      : Factor w/ 24 levels "1000","2000",..: 12 12 12 12 12 12 12 12 12 1
2 ...
 $ stratum_label: chr  "north_1_6" "north_1_6" "north_1_6" "north_1_6" ...
 $ sex          : int  2 2 2 1 1 1 1 2 1 2 ...
 $ cl_age       : Factor w/ 8 levels "(0,14]","(14,24]",..: 3 5 5 5 1 1 1 8 1 1 ...
 $ active       : num  1 1 0 1 0 0 0 0 0 0 ...
 $ income_hh    : num  17043 28143 4791 28043 27185 ...
 $ unemployed   : num  0 0 1 0 0 0 0 0 0 0 ...
 $ inactive     : num  0 0 0 0 1 1 1 1 1 1 ...
 $ Prob_1st     : num  0.187 0.187 0.187 0.187 0.187 ...
 $ Prob_2st     : num  0.0323 0.0323 0.0323 0.0323 0.0323 ...
 $ Prob_tot     : num  0.00604 0.00604 0.00604 0.00604 0.00604 ...
 $ weight       : num  166 166 166 166 166 ...
 $ SR           : num  0 0 0 0 0 0 0 0 0 0 ...
 $ nSR          : num  1 1 1 1 1 1 1 1 1 1 ...
 $ stratum_2    : chr  "12000-2" "12000-2" "12000-2" "12000-2" ...
```

In [27]:

```r
## Sample design description
samp$stratum_2 <- as.factor(samp$stratum_2)
sample.des <- e.svydesign(samp,
                          ids= ~ municipality + id_hh,
                          strata = ~ stratum_2,
                          weights = ~ weight,
                          self.rep.str = ~ SR,
                          check.data = TRUE)
```

```
# Empty levels found in factors: id_hh
# Empty levels have been dropped!

Warning message in e.svydesign(samp, ids = ~municipality + id_hh, strata = ~stratum_
2, :
"Sampling variance estimation for this design will take into account only leading co
ntributions, i.e. PSUs in not-SR strata and SSUs in SR strata (see ?e.svydesign and
?ReGenesees.options for details)"
```

In [29]:

```r
## Find and collapse lonely strata
ls <- find.lon.strata(sample.des)
if (!is.null(ls)) sample.des <- collapse.strata(sample.des)
```

```
# No lonely PSUs found!
```

In [30]:

```r
## Calibration with known totals
totals <- pop.template(sample.des,
             calmodel = ~ sex : cl_age,
             partition = ~ region)
totals <- fill.template(pop, totals, mem.frac = 10)
sample.cal <- e.calibrate(sample.des,
                          totals,
                          calmodel = ~ sex : cl_age,
                          partition = ~ region,
                          calfun = "logit",
                          bounds = c(0.3, 2.6),
                          aggregate.stage = 2,
                          force = FALSE)
```

```
# Coherence check between 'universe' and 'template': OK
```

Preparation of inputs for allocation steps

In [31]:

```r
samp_frame <- pop
RGdes <- sample.des
RGcal <- sample.cal
strata_var <- c("stratum")
target_vars <- c("income_hh",
                 "active",
                 "inactive",
                 "unemployed")
weight_var <- "weight"
deff_var <- "stratum"
id_PSU <- c("municipality")
id_SSU <- c("id_hh")
domain_var <- c("region")
delta <- 1
minimum <- 50

inp2 <- prepareInputToAllocation2(
        samp_frame,  # sampling frame
        RGdes,       # ReGenesees design object
        RGcal,       # ReGenesees calibrated object
        id_PSU,      # identification variable of PSUs
        id_SSU,      # identification variable of SSUs
        strata_var,  # strata variable
        target_vars, # target variables
        deff_var,    # deff variable
        domain_var,  # domain variable
        delta,       # Average number of SSUs for each selection unit
```

```
    minimum      # Minimum number of SSUs to be selected in each PSU
  )
```

In [32]:

```
head(inp2$strata)
```

A data.frame: 6 × 15

	stratum	STRATUM	N	M1	M2	M3	M4	S1	S2	
	<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	
1	1000	1000	196189	22364.93	0.6793373	0.2322720	0.08839079	16556.88	0.4667313	0.4
2	10000	10000	105863	29427.86	0.7921197	0.1912077	0.01667262	27040.22	0.4057907	0.3
3	11000	11000	205526	28506.72	0.7749079	0.2063259	0.01876620	43089.05	0.4176429	0.4
4	12000	12000	57552	24275.15	0.7521653	0.2222993	0.02553542	16210.24	0.4317553	0.4
5	13000	13000	103001	28517.71	0.7654855	0.2004329	0.03408161	22588.38	0.4236950	0.4
6	14000	14000	84100	24332.26	0.7365891	0.2336707	0.02974023	15393.85	0.4404834	0.4

In [33]:

```
head(inp2$deff)
```

A data.frame: 6 × 7

	stratum	STRATUM	DEFF1	DEFF2	DEFF3	DEFF4	b_nar
	<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1000	1000	0.992453	1.000576	1.001071	1.004864	143.00000
2	10000	10000	1.020615	1.014979	1.015570	1.002514	97.83333
3	11000	11000	1.661060	0.792896	0.637480	1.036756	50.00000
4	12000	12000	1.200498	2.667262	2.533419	1.983936	50.00000
5	13000	13000	1.014184	1.011510	1.013485	1.007290	703.00000
6	14000	14000	1.012919	0.996743	0.996826	1.010600	144.25000

In [34]:

```
head(inp2$effst)
```

A data.frame: 6 × 6

	stratum	STRATUM	EFFST1	EFFST2	EFFST3	EFFST4
	<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	1000	1000	1.061891	0.9511291	0.9071854	1.0137193
2	10000	10000	1.005724	0.9077114	0.8991158	0.9780552
3	11000	11000	1.005722	0.9309392	0.9240808	0.9998968
4	12000	12000	1.026967	0.9241132	0.9117161	0.9911560
5	13000	13000	1.006354	0.9244961	0.9085689	0.9977077
6	14000	14000	1.002360	0.9348739	0.9237139	1.0065308

In [35]:

```
head(inp2$rho)
```

A data.frame: 6 × 9

	STRATUM	RHO_AR1	RHO_NAR1	RHO_AR2	RHO_NAR2	RHO_AR3	RHO_NAR3	RI
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	
1	1000	1	-0.00005314789	1	0.000004056338	1	0.000007542254	
2	10000	1	0.00021289157	1	0.000154688468	1	0.000160791738	
3	11000	1	0.01349102041	1	-0.004226612245	1	-0.007398367347	
4	12000	1	0.00409179592	1	0.034025755102	1	0.031294265306	
5	13000	1	0.00002020513	1	0.000016396011	1	0.000019209402	
6	14000	1	0.00009018499	1	-0.000022736475	1	-0.000022157068	

In [36]:
```
head(inp2$psu_file)
```

A data.frame: 6 × 3

	PSU_ID	STRATUM	PSU_MOS
	<dbl>	<fct>	<dbl>
1	309	1000	50845
2	330	1000	146162
3	292	2000	24794
4	293	2000	19609
5	300	2000	13897
6	304	2000	36195

In [37]:
```
head(inp2$des_file)
```

A data.frame: 6 × 4

	STRATUM	STRAT_MOS	DELTA	MINIMUM
	<fct>	<dbl>	<dbl>	<dbl>
1	1000	197007	1	50
2	2000	261456	1	50
3	3000	115813	1	50
4	4000	17241	1	50
5	5000	101067	1	50
6	6000	47218	1	50

Allocation

In [38]:
```
set.seed(1234)
inp2$des_file$MINIMUM <- 50
alloc2 <- beat.2st(stratif = inp2$strata,
                   errors = cv,
                   des_file = inp2$des_file,
```

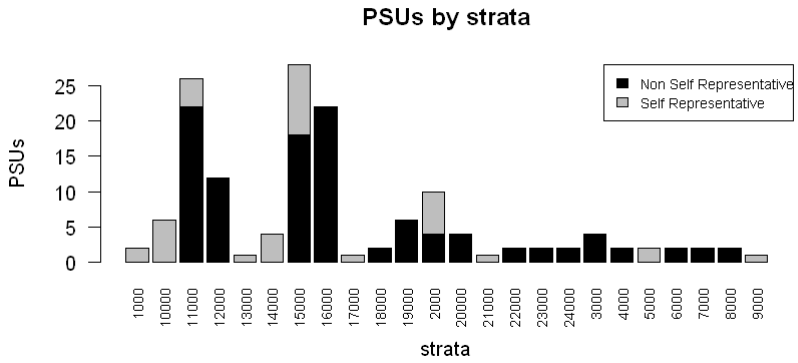
```
                    psu_file = inp2$psu_file,
                    rho = inp2$rho,
                    deft_start = NULL,
                    effst = inp2$effst,
                    minnumstrat = 2,
                    minPSUstrat = 2)
```

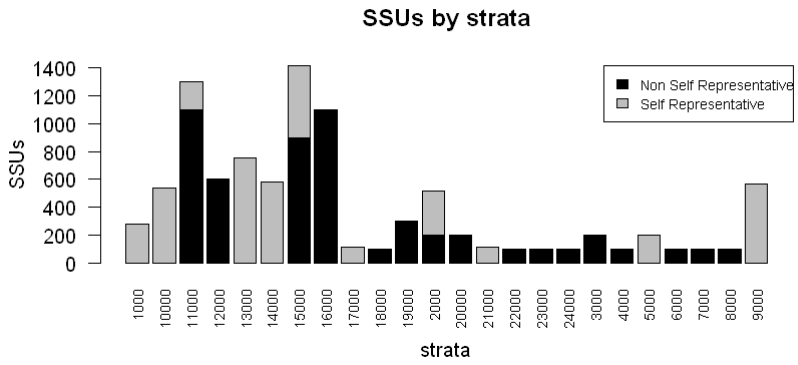
```
  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 9557
2          1     71      92       163 8464
3          2     38     108       146 8398
4          3     38     108       146 8396
```

Selection of PSUs (I stage)

In [39]:

```
set.seed(1234)
sample_1st <- select_PSU(alloc2, type="ALLOC", pps=TRUE)
```





In [40]:

```
sample_1st$PSU_stats
```

A data.frame: 25 × 7

STRATUM	PSU	PSU_SR	PSU_NSR	SSU	SSU_SR	SSU_NSR
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1000	2	2	0	279	279	0
2000	10	6	4	517	317	200
3000	4	0	4	200	0	200
4000	2	0	2	100	0	100
5000	2	2	0	202	202	0
6000	2	0	2	100	0	100

STRATUM	PSU	PSU_SR	PSU_NSR	SSU	SSU_SR	SSU_NSR
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
7000	2	0	2	100	0	100
8000	2	0	2	100	0	100
9000	1	1	0	564	564	0
10000	6	6	0	537	537	0
11000	26	4	22	1300	200	1100
12000	12	0	12	600	0	600
13000	1	1	0	756	756	0
14000	4	4	0	583	583	0
15000	28	10	18	1414	514	900
16000	22	0	22	1100	0	1100
17000	1	1	0	114	114	0
18000	2	0	2	100	0	100
19000	6	0	6	300	0	300
20000	4	0	4	200	0	200
21000	1	1	0	113	113	0
22000	2	0	2	100	0	100
23000	2	0	2	100	0	100
24000	2	0	2	100	0	100
Total	146	38	108	9579	4179	5400

Selection of SSUs (II stage)

In [41]:

```
set.seed(1234)
samp <- select_SSU(df=pop,
                   PSU_code="municipality",
                   SSU_code="id_ind",
                   PSU_sampled=sample_1st$sample_PSU,
                   verbose=TRUE)
```

```
PSU =  4  *** Selected SSU =  66
PSU =  8  *** Selected SSU =  564
PSU =  10  *** Selected SSU =  50
PSU =  11  *** Selected SSU =  96
PSU =  13  *** Selected SSU =  72
PSU =  15  *** Selected SSU =  50
PSU =  17  *** Selected SSU =  50
PSU =  19  *** Selected SSU =  50
PSU =  22  *** Selected SSU =  50
PSU =  27  *** Selected SSU =  50
PSU =  29  *** Selected SSU =  50
PSU =  34  *** Selected SSU =  50
PSU =  36  *** Selected SSU =  50
PSU =  40  *** Selected SSU =  79
PSU =  41  *** Selected SSU =  53
PSU =  44  *** Selected SSU =  50
```

```
PSU =  50  *** Selected SSU =  50
PSU =  51  *** Selected SSU =  171
PSU =  52  *** Selected SSU =  50
PSU =  53  *** Selected SSU =  50
PSU =  55  *** Selected SSU =  50
PSU =  56  *** Selected SSU =  50
PSU =  59  *** Selected SSU =  50
PSU =  61  *** Selected SSU =  50
PSU =  64  *** Selected SSU =  50
PSU =  67  *** Selected SSU =  50
PSU =  71  *** Selected SSU =  50
PSU =  72  *** Selected SSU =  50
PSU =  73  *** Selected SSU =  50
PSU =  77  *** Selected SSU =  50
PSU =  79  *** Selected SSU =  50
PSU =  85  *** Selected SSU =  50
PSU =  88  *** Selected SSU =  50
PSU =  92  *** Selected SSU =  50
PSU =  93  *** Selected SSU =  50
PSU =  94  *** Selected SSU =  50
PSU =  95  *** Selected SSU =  50
PSU =  96  *** Selected SSU =  50
PSU =  106  *** Selected SSU =  50
PSU =  107  *** Selected SSU =  50
PSU =  108  *** Selected SSU =  50
PSU =  109  *** Selected SSU =  50
PSU =  113  *** Selected SSU =  50
PSU =  114  *** Selected SSU =  50
PSU =  116  *** Selected SSU =  50
PSU =  117  *** Selected SSU =  53
PSU =  120  *** Selected SSU =  50
PSU =  121  *** Selected SSU =  109
PSU =  122  *** Selected SSU =  50
PSU =  123  *** Selected SSU =  50
PSU =  124  *** Selected SSU =  50
PSU =  130  *** Selected SSU =  50
PSU =  136  *** Selected SSU =  50
PSU =  138  *** Selected SSU =  50
PSU =  140  *** Selected SSU =  50
PSU =  142  *** Selected SSU =  50
PSU =  143  *** Selected SSU =  50
PSU =  156  *** Selected SSU =  50
PSU =  161  *** Selected SSU =  50
PSU =  162  *** Selected SSU =  50
PSU =  163  *** Selected SSU =  50
PSU =  168  *** Selected SSU =  50
PSU =  170  *** Selected SSU =  50
PSU =  175  *** Selected SSU =  50
PSU =  176  *** Selected SSU =  50
PSU =  180  *** Selected SSU =  50
PSU =  184  *** Selected SSU =  50
PSU =  186  *** Selected SSU =  53
PSU =  187  *** Selected SSU =  50
PSU =  188  *** Selected SSU =  50
PSU =  197  *** Selected SSU =  50
PSU =  198  *** Selected SSU =  50
PSU =  201  *** Selected SSU =  58
PSU =  205  *** Selected SSU =  50
PSU =  207  *** Selected SSU =  50
PSU =  209  *** Selected SSU =  50
PSU =  211  *** Selected SSU =  50
PSU =  213  *** Selected SSU =  50
PSU =  214  *** Selected SSU =  131
PSU =  221  *** Selected SSU =  104
PSU =  224  *** Selected SSU =  50
```

```
PSU =  228  *** Selected SSU =  239
PSU =  229  *** Selected SSU =  50
PSU =  233  *** Selected SSU =  50
PSU =  242  *** Selected SSU =  50
PSU =  246  *** Selected SSU =  50
PSU =  251  *** Selected SSU =  50
PSU =  253  *** Selected SSU =  50
PSU =  259  *** Selected SSU =  756
PSU =  269  *** Selected SSU =  50
PSU =  270  *** Selected SSU =  50
PSU =  271  *** Selected SSU =  50
PSU =  274  *** Selected SSU =  50
PSU =  276  *** Selected SSU =  50
PSU =  278  *** Selected SSU =  50
PSU =  281  *** Selected SSU =  50
PSU =  283  *** Selected SSU =  50
PSU =  288  *** Selected SSU =  50
PSU =  290  *** Selected SSU =  50
PSU =  291  *** Selected SSU =  50
PSU =  292  *** Selected SSU =  50
PSU =  293  *** Selected SSU =  50
PSU =  300  *** Selected SSU =  50
PSU =  302  *** Selected SSU =  50
PSU =  304  *** Selected SSU =  60
PSU =  306  *** Selected SSU =  50
PSU =  309  *** Selected SSU =  72
PSU =  315  *** Selected SSU =  50
PSU =  317  *** Selected SSU =  50
PSU =  321  *** Selected SSU =  50
PSU =  323  *** Selected SSU =  50
PSU =  330  *** Selected SSU =  207
PSU =  332  *** Selected SSU =  50
PSU =  335  *** Selected SSU =  50
PSU =  342  *** Selected SSU =  57
PSU =  345  *** Selected SSU =  50
PSU =  346  *** Selected SSU =  50
PSU =  347  *** Selected SSU =  50
PSU =  356  *** Selected SSU =  50
PSU =  372  *** Selected SSU =  50
PSU =  373  *** Selected SSU =  50
PSU =  378  *** Selected SSU =  50
PSU =  380  *** Selected SSU =  50
PSU =  381  *** Selected SSU =  50
PSU =  382  *** Selected SSU =  152
PSU =  383  *** Selected SSU =  50
PSU =  387  *** Selected SSU =  50
PSU =  411  *** Selected SSU =  50
PSU =  426  *** Selected SSU =  50
PSU =  428  *** Selected SSU =  50
PSU =  435  *** Selected SSU =  50
PSU =  445  *** Selected SSU =  114
PSU =  448  *** Selected SSU =  50
PSU =  452  *** Selected SSU =  50
PSU =  456  *** Selected SSU =  50
PSU =  459  *** Selected SSU =  50
PSU =  467  *** Selected SSU =  50
PSU =  473  *** Selected SSU =  50
PSU =  477  *** Selected SSU =  50
PSU =  484  *** Selected SSU =  50
PSU =  485  *** Selected SSU =  50
PSU =  496  *** Selected SSU =  113
PSU =  502  *** Selected SSU =  50
PSU =  504  *** Selected SSU =  50
PSU =  510  *** Selected SSU =  50
PSU =  512  *** Selected SSU =  50
```

```
--------------------------------
Total PSU =  146
Total SSU =  9579
--------------------------------
```

In [42]:

```
nrow(samp)
sum(alloc2$alloc$ALLOC[-nrow(alloc2$alloc)])
```

9579

8396

In [43]:

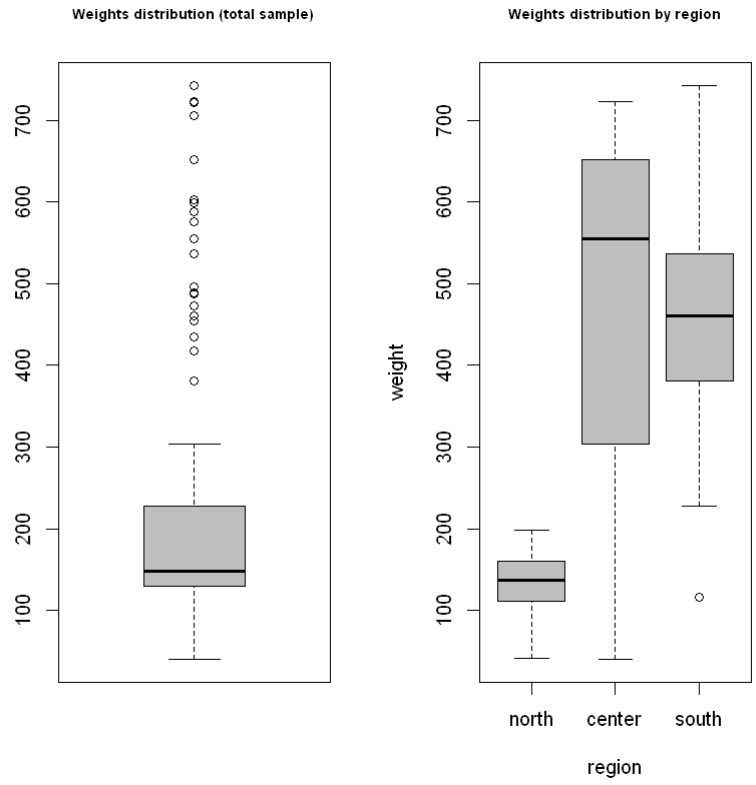
```
nrow(pop)
sum(samp$weight)
```

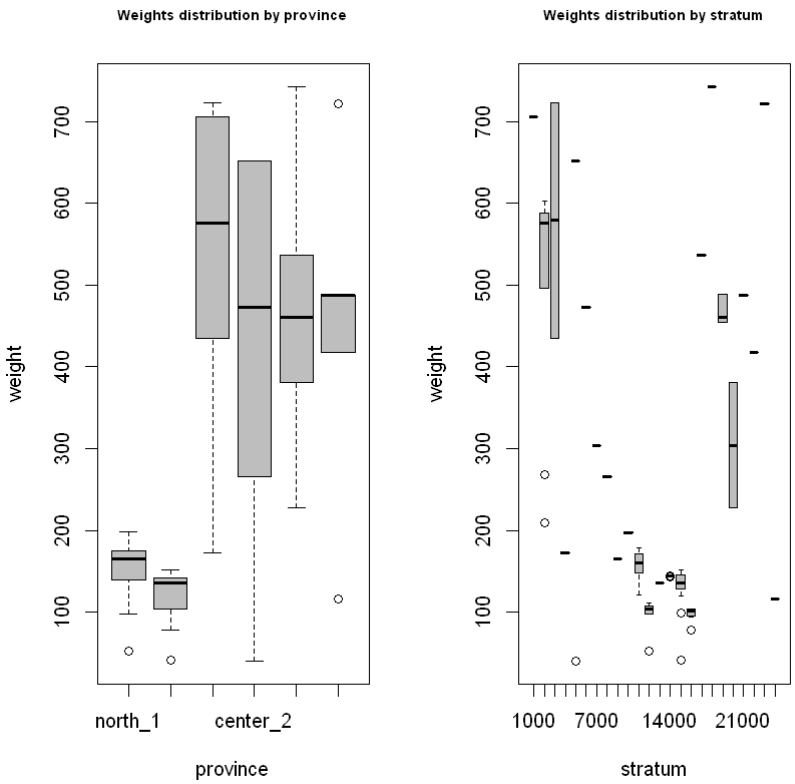
2258507

2258507

In [44]:

```
## Plot of weights distribution
par(mfrow=c(1, 2))
boxplot(samp$weight,col="grey")
title("Weights distribution (total sample)",cex.main=0.7)
boxplot(weight ~ region, data=samp,col="grey")
title("Weights distribution by region",cex.main=0.7)
par(mfrow=c(1, 2))
boxplot(weight ~ province, data=samp,col="grey")
title("Weights distribution by province",cex.main=0.7)
boxplot(weight ~ stratum, data=samp,col="grey")
title("Weights distribution by stratum",cex.main=0.7)
```





Precision constraints compliance control (by simulation)

In [45]:

```
df=pop
df$one <- 1
PSU_code="municipality"
SSU_code="id_ind"
target_vars <- c("income_hh",
                 "active",
                 "inactive",
                 "unemployed")
```

In [46]:

```
# Domain level = national
domain_var <- "one"
set.seed(1234)
eval21 <- eval_2stage(df,
                      PSU_code,
                      SSU_code,
                      domain_var,
                      target_vars,
                      PSU_sampled=sample_1st$sample_PSU,
                      nsampl=100,
                      writeFiles=FALSE,
                      progress=TRUE)
eval21$coeff_var
```

```
  |======================================================================| 100%
```

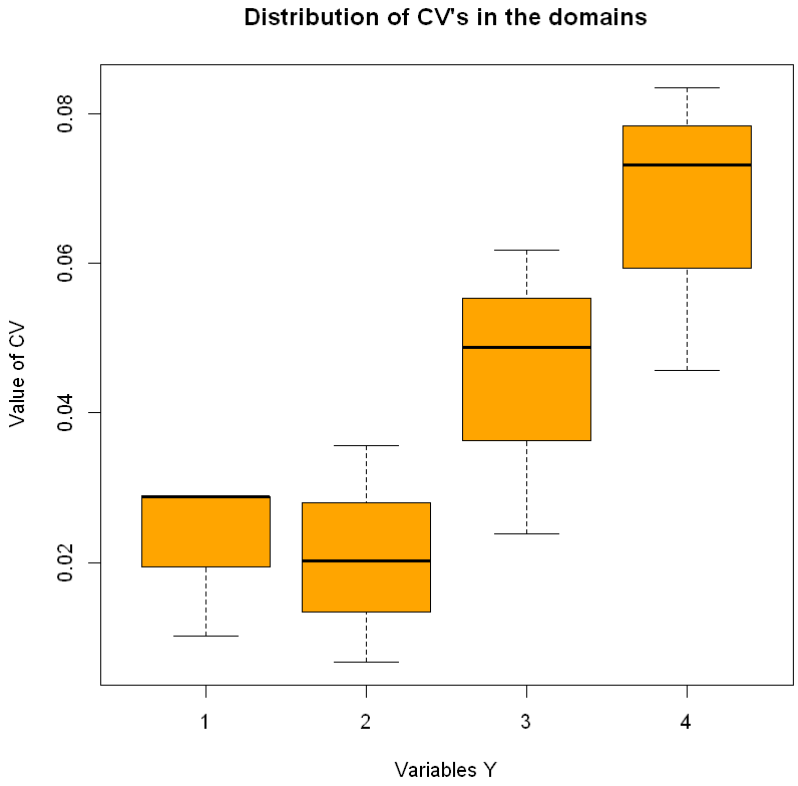
A data.frame: 1 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0126	0.0097	0.0252	0.0361	DOM1

In [47]:

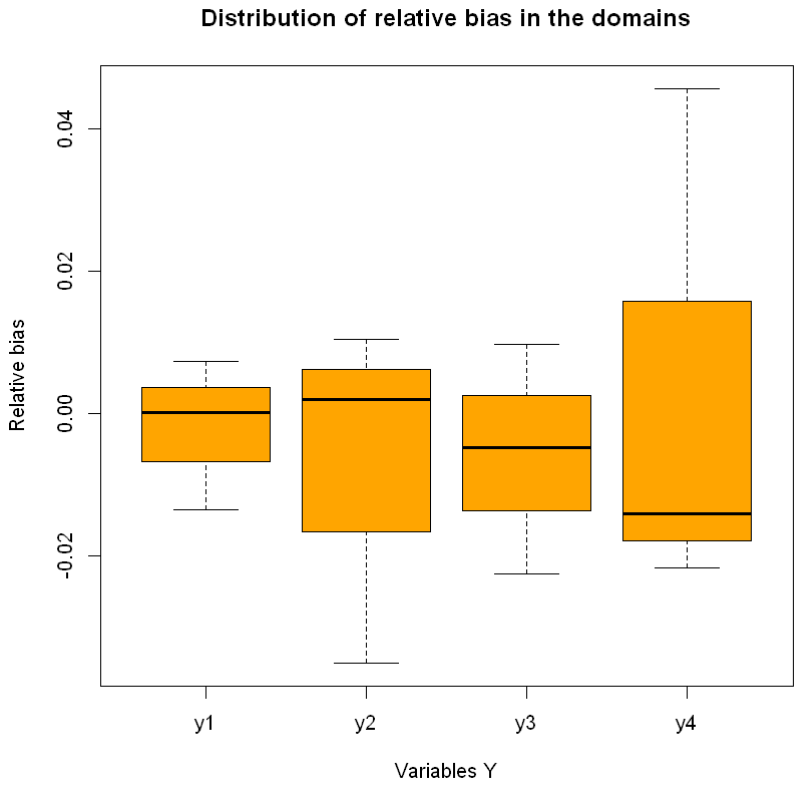
```
# Domain level = regional
domain_var <- "region"
set.seed(1234)
eval22 <- eval_2stage(df,
                      PSU_code,
                      SSU_code,
                      domain_var,
                      target_vars,
                      PSU_sampled=sample_1st$sample_PSU,
                      nsampl=100,
                      writeFiles=FALSE,
                      progress=TRUE)
eval22$coeff_var
```

```
  |======================================================================| 100%
```



A data.frame: 3 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0102	0.0067	0.0238	0.0835	DOM1
0.0287	0.0202	0.0488	0.0731	DOM2
0.0290	0.0357	0.0617	0.0457	DOM3



In [48]:
```
alloc2$sensitivity
```

A data.frame: 4 × 6

	Type	Dom	V1	V2	V3	V4
	<chr>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
2	DOM1	1	1	1	1	1
6	DOM2	1	1	0	1	1273
10	DOM2	2	1	1	14	211
14	DOM2	3	114	1	66	1

In [ ]: