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     50     122       172 8072
3          2     51     126       177 8071

 2  iterations PSU_SR PSU NSR PSU Total  SSU
```

1	0	0	0	0	7836
2	1	31	120	151	8130
3	2	34	120	154	8127

3	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	27	112	139	8185
3	2	35	114	149	8177

4	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	25	104	129	8236
3	2	32	104	136	8231

5	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	25	100	125	8296
3	2	28	100	128	8297

6	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	19	96	115	8368
3	2	20	102	122	8364

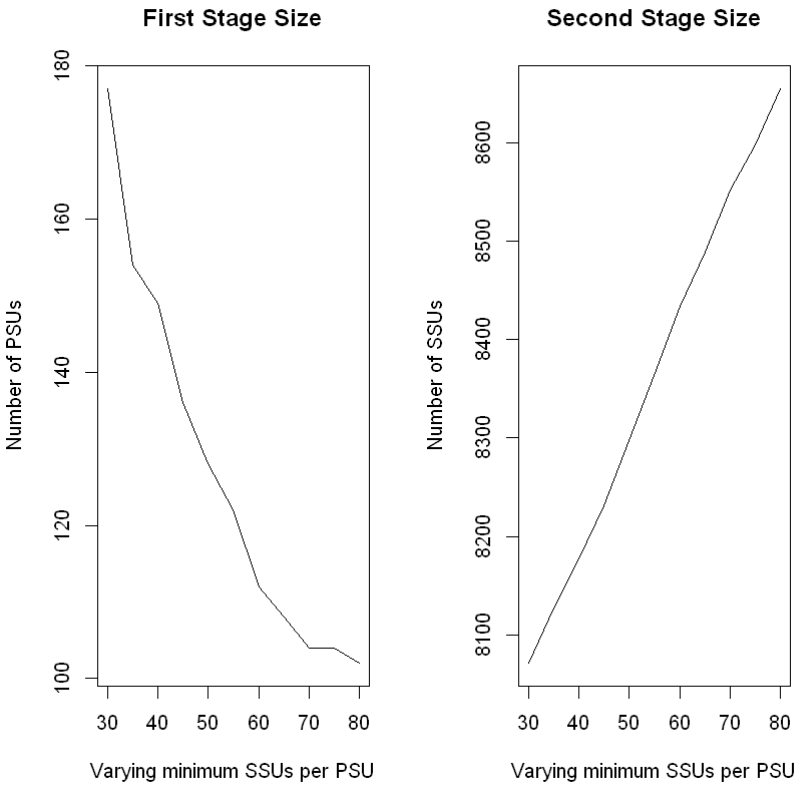
7	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	19	90	109	8434
3	2	20	92	112	8434

8	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	23	82	105	8482
3	2	18	90	108	8489

9	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	20	78	98	8547
3	2	18	86	104	8552

10	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	20	76	96	8601
3	2	22	82	104	8598

11	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	16	72	88	8685
3	2	20	82	102	8655



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]:

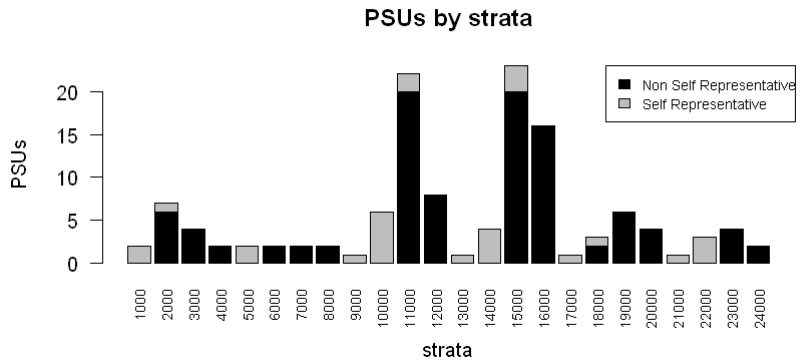
```
inp1$desfile$MINIMUM <- 50
alloc1 <- beat.2st(stratif = inp1$strata,
                   errors = cv,
                   des_file = inp1$des_file,
                   psu_file = inp1$psu_file,
                   rho = inp1$rho,
                   deft_start = NULL,
                   effst = inp1$effst,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

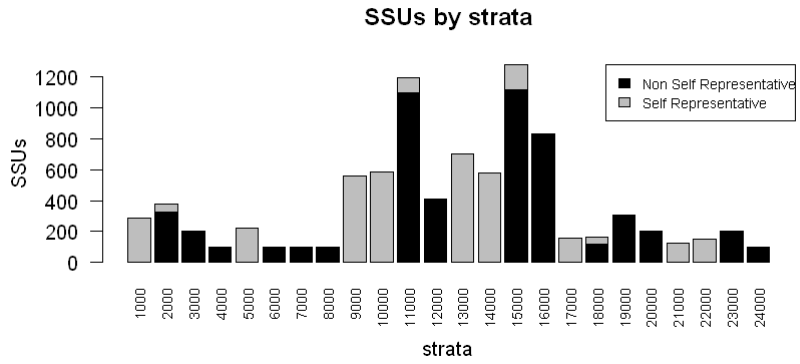
```
  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7887
2          1     25     100       125 8342
3          2     28     100       128 8344
```

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	287	287	0
2000	7	1	6	380	52	328
3000	4	0	4	204	0	204
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	558	558	0
10000	6	6	0	588	588	0
11000	22	2	20	1198	100	1098
12000	8	0	8	410	0	410
13000	1	1	0	704	704	0
14000	4	4	0	577	577	0
15000	23	3	20	1281	161	1120
16000	16	0	16	830	0	830
17000	1	1	0	157	157	0

STRATUM	PSU	PSU_SR	PSU_NSR	SSU	SSU_SR	SSU_NSR
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
18000	3	1	2	166	50	116
19000	6	0	6	308	0	308
20000	4	0	4	200	0	200
21000	1	1	0	127	127	0
22000	3	3	0	151	151	0
23000	4	0	4	204	0	204
24000	2	0	2	100	0	100
Total	128	28	100	9049	3731	5318

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 =  4  *** Selected SSU =  72
PSU =  6  *** Selected SSU =  52
PSU =  8  *** Selected SSU =  558
PSU =  10  *** Selected SSU =  52
PSU =  11  *** Selected SSU =  105
PSU =  13  *** Selected SSU =  79
PSU =  15  *** Selected SSU =  56
PSU =  17  *** Selected SSU =  53
PSU =  21  *** Selected SSU =  56
PSU =  24  *** Selected SSU =  50
PSU =  25  *** Selected SSU =  50
PSU =  27  *** Selected SSU =  50
PSU =  30  *** Selected SSU =  56
PSU =  33  *** Selected SSU =  50
PSU =  35  *** Selected SSU =  52
PSU =  38  *** Selected SSU =  60
PSU =  40  *** Selected SSU =  87
PSU =  41  *** Selected SSU =  58
PSU =  42  *** Selected SSU =  50
PSU =  47  *** Selected SSU =  52
PSU =  51  *** Selected SSU =  187
PSU =  56  *** Selected SSU =  53
PSU =  61  *** Selected SSU =  60
PSU =  67  *** Selected SSU =  52
PSU =  72  *** Selected SSU =  56
PSU =  73  *** Selected SSU =  51
PSU =  77  *** Selected SSU =  58
PSU =  82  *** Selected SSU =  52
PSU =  90  *** Selected SSU =  52
PSU =  96  *** Selected SSU =  58
PSU =  101  *** Selected SSU =  51
PSU =  106  *** Selected SSU =  62
PSU =  107  *** Selected SSU =  50
PSU =  109  *** Selected SSU =  50
PSU =  111  *** Selected SSU =  52
PSU =  113  *** Selected SSU =  62
```

```
PSU =  116  *** Selected SSU =  50
PSU =  117  *** Selected SSU =  52
PSU =  119  *** Selected SSU =  57
PSU =  121  *** Selected SSU =  107
PSU =  124  *** Selected SSU =  50
PSU =  125  *** Selected SSU =  55
PSU =  130  *** Selected SSU =  64
PSU =  139  *** Selected SSU =  52
PSU =  144  *** Selected SSU =  50
PSU =  153  *** Selected SSU =  50
PSU =  161  *** Selected SSU =  50
PSU =  162  *** Selected SSU =  56
PSU =  163  *** Selected SSU =  50
PSU =  168  *** Selected SSU =  54
PSU =  170  *** Selected SSU =  58
PSU =  175  *** Selected SSU =  57
PSU =  183  *** Selected SSU =  54
PSU =  184  *** Selected SSU =  50
PSU =  185  *** Selected SSU =  54
PSU =  186  *** Selected SSU =  52
PSU =  189  *** Selected SSU =  51
PSU =  191  *** Selected SSU =  54
PSU =  192  *** Selected SSU =  50
PSU =  194  *** Selected SSU =  50
PSU =  197  *** Selected SSU =  62
PSU =  198  *** Selected SSU =  64
PSU =  201  *** Selected SSU =  57
PSU =  202  *** Selected SSU =  55
PSU =  203  *** Selected SSU =  54
PSU =  210  *** Selected SSU =  51
PSU =  214  *** Selected SSU =  130
PSU =  220  *** Selected SSU =  54
PSU =  221  *** Selected SSU =  103
PSU =  228  *** Selected SSU =  237
PSU =  259  *** Selected SSU =  704
PSU =  265  *** Selected SSU =  55
PSU =  269  *** Selected SSU =  56
PSU =  271  *** Selected SSU =  52
PSU =  274  *** Selected SSU =  50
PSU =  281  *** Selected SSU =  50
PSU =  283  *** Selected SSU =  53
PSU =  286  *** Selected SSU =  53
PSU =  288  *** Selected SSU =  62
PSU =  289  *** Selected SSU =  58
PSU =  290  *** Selected SSU =  55
PSU =  292  *** Selected SSU =  64
PSU =  293  *** Selected SSU =  50
PSU =  302  *** Selected SSU =  50
PSU =  303  *** Selected SSU =  52
PSU =  304  *** Selected SSU =  52
PSU =  309  *** Selected SSU =  74
PSU =  314  *** Selected SSU =  50
PSU =  317  *** Selected SSU =  50
PSU =  321  *** Selected SSU =  52
PSU =  323  *** Selected SSU =  50
PSU =  330  *** Selected SSU =  213
PSU =  331  *** Selected SSU =  50
PSU =  332  *** Selected SSU =  50
PSU =  342  *** Selected SSU =  64
PSU =  343  *** Selected SSU =  50
PSU =  360  *** 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 =  390  *** Selected SSU =  50
PSU =  395  *** Selected SSU =  50
PSU =  402  *** Selected SSU =  50
PSU =  416  *** Selected SSU =  50
PSU =  424  *** Selected SSU =  58
PSU =  425  *** Selected SSU =  58
PSU =  435  *** Selected SSU =  54
PSU =  438  *** Selected SSU =  50
PSU =  445  *** Selected SSU =  157
PSU =  447  *** Selected SSU =  50
PSU =  471  *** Selected SSU =  54
PSU =  472  *** Selected SSU =  50
PSU =  476  *** Selected SSU =  50
PSU =  479  *** Selected SSU =  50
PSU =  485  *** Selected SSU =  50
PSU =  488  *** Selected SSU =  50
PSU =  490  *** Selected SSU =  52
PSU =  496  *** Selected SSU =  127
PSU =  502  *** Selected SSU =  51
PSU =  503  *** Selected SSU =  50
PSU =  505  *** Selected SSU =  50
PSU =  506  *** Selected SSU =  50
PSU =  508  *** Selected SSU =  50
PSU =  511  *** Selected SSU =  52
--------------------------------
Total PSU =  128
Total SSU =  9049
--------------------------------
```

In [17]:

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

9049

8344

In [18]:

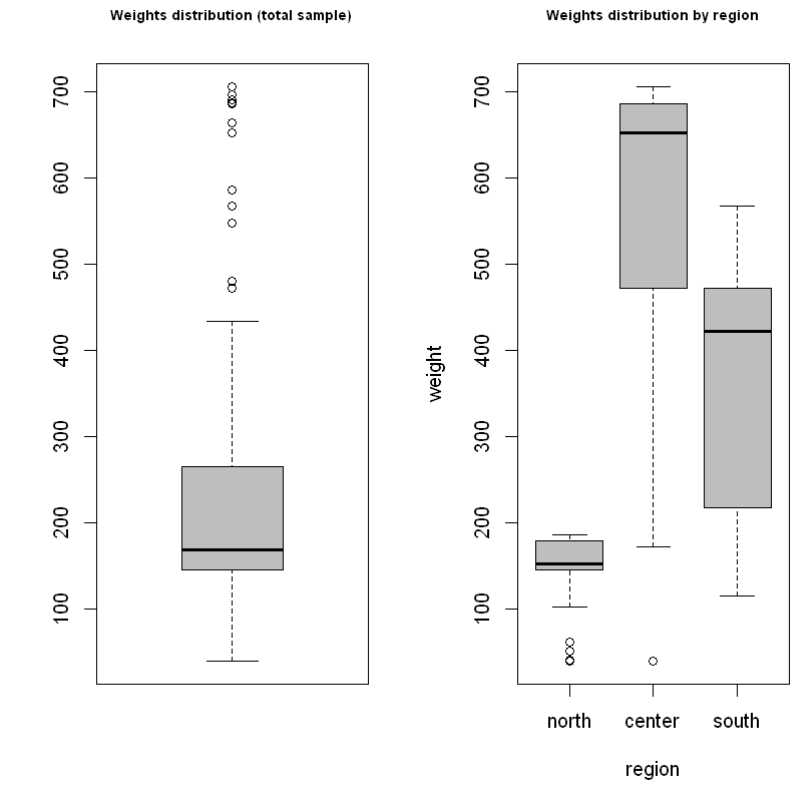
```
nrow(pop)
sum(samp$weight)
```

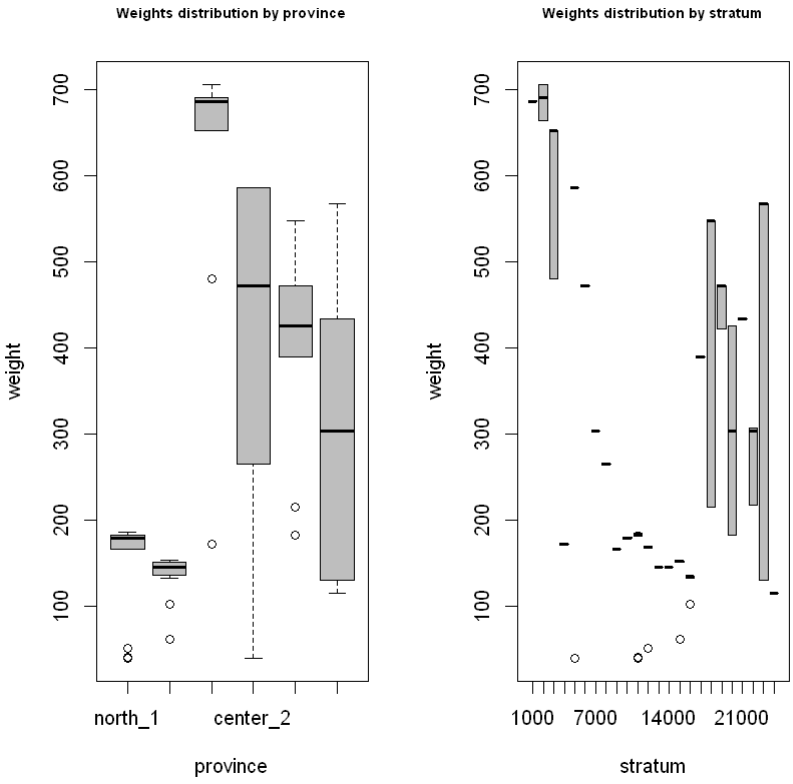
2258507

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 [22]:

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

```
                  "inactive",
                  "unemployed")
```

In [23]:

```
# 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%
```

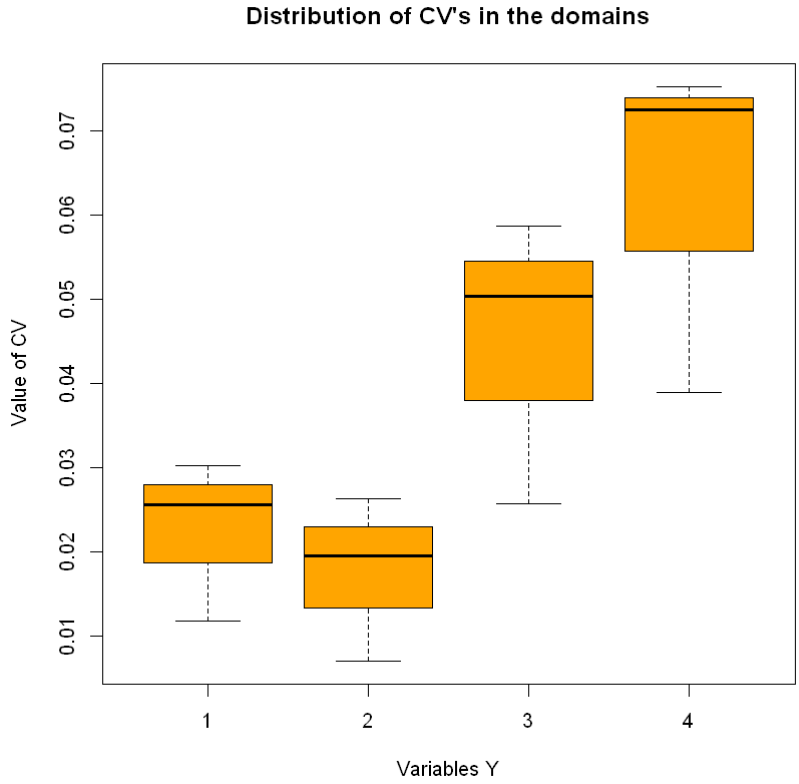
A data.frame: 1 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0125	0.009	0.0266	0.0319	DOM1

In [24]:

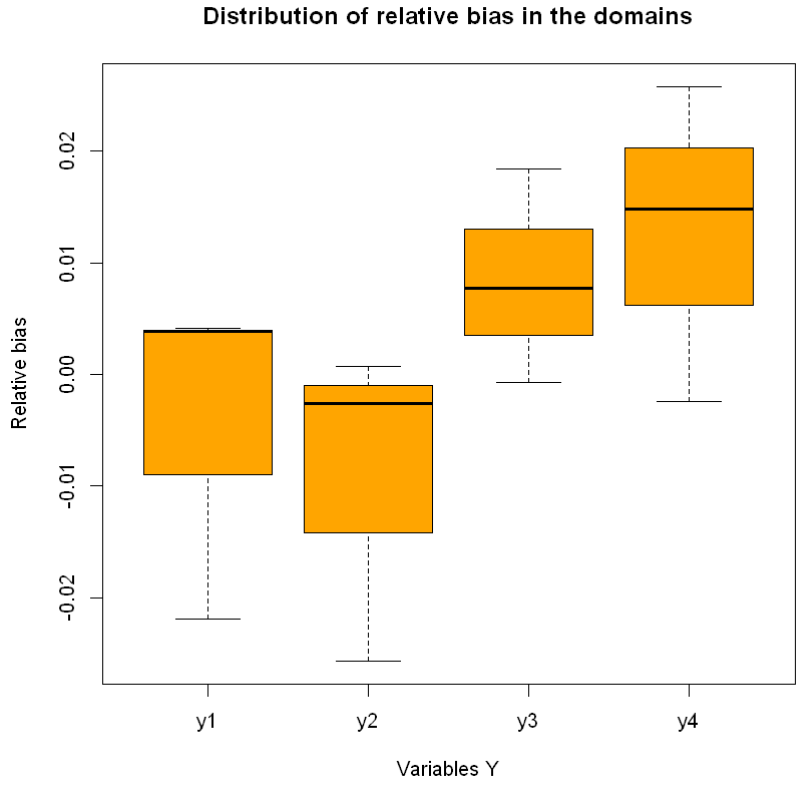
```
# Domain level = regional
domain_var <- "region"
set.seed(1234)
set.seed(1234)
eval12 <- eval_2stage(df,
                      PSU_code,
                      SSU_code,
                      domain_var,
                      target_vars,
                      sample_1st$sample_PSU,
             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

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



A data.frame: 3 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0118	0.0070	0.0257	0.0752	DOM1
0.0303	0.0196	0.0587	0.0725	DOM2
0.0256	0.0264	0.0503	0.0390	DOM3



In [26]:
```
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	1184
10	DOM2	2	1	0	1	246
14	DOM2	3	203	1	27	1

In [27]:
```
save(samp,file="sample.RData")
```

Scenario 2

One previous round of the sampling survey is available.

Analysis of sampled data

In [20]:
```
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 [21]:

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

```
'data.frame':   9049 obs. of  20 variables:
 $ municipality : Factor w/ 128 levels "4","6","8","10",..: 4 4 4 4 4 4 4 4 4 4 ...
 $ id_ind       : int  1580 1592 1624 1633 1705 1724 1737 1791 1807 1834 ...
 $ 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",..: 440225 440269 440391
440424 440702 440757 440802 440990 441057 441180 ...
 $ 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 1 1 2 2 2 1 1 1 ...
 $ cl_age       : Factor w/ 8 levels "(0,14]","(14,24]",..: 6 2 6 4 3 2 4 5 6 1 ...
 $ active       : num  1 1 1 1 1 1 1 1 1 0 ...
 $ income_hh    : num  25236 33867 12907 24261 139679 ...
 $ unemployed   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ inactive     : num  0 0 0 0 0 0 0 0 0 1 ...
 $ Prob_1st     : num  0.215 0.215 0.215 0.215 0.215 ...
 $ Prob_2st     : num  0.0277 0.0277 0.0277 0.0277 0.0277 ...
 $ Prob_tot     : num  0.00595 0.00595 0.00595 0.00595 0.00595 ...
 $ weight       : num  168 168 168 168 168 ...
 $ 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  "120001" "120001" "120001" "120001" ...
```

In [22]:

```
## 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 [28]:

```
## Find and collapse lonely strata
ls <- find.lon.strata(sample.des)
sample.des <- collapse.strata(sample.des)
```

```
# No lonely PSUs found!

Error in find.lPSU(analyze.strata(design)): No point in strata collapsing: no lonely
PSUs found!
Traceback:

1. collapse.strata(sample.des)
2. find.lPSU(analyze.strata(design))
3. stop("No point in strata collapsing: no lonely PSUs found!")
```

In [29]:

```
## 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 [30]:

```
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 [31]:
```
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	199187	25855.55	0.6477995	0.2201875	0.13201306	21753.09	0.4776561	0.4
2	10000	10000	106038	29925.84	0.7587198	0.2159735	0.02530666	27411.66	0.4278599	0.4
3	11000	11000	205965	27984.95	0.7926826	0.1900771	0.01724029	22859.58	0.4053849	0.3
4	12000	12000	57672	25815.56	0.7862010	0.1855170	0.02828208	28698.40	0.4099866	0.3
5	13000	13000	102787	28770.46	0.7650873	0.2037165	0.03119621	23459.62	0.4239443	0.4
6	14000	14000	83996	24609.56	0.7503445	0.2212803	0.02837520	17776.82	0.4328136	0.4

In [32]:
```
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	1.010392	1.017108	0.997143	0.995809	143.50000
2	10000	10000	1.096035	1.001972	1.010905	1.000440	98.00000
3	11000	11000	1.375480	1.295916	1.077708	1.779027	54.45455
4	12000	12000	3.072391	1.395598	0.687271	2.198334	51.25000
5	13000	13000	1.014123	1.008403	1.010085	1.007358	704.00000
6	14000	14000	1.015870	0.993466	0.994993	1.010549	144.25000

In [33]:
```
head(inp2$effst)
```

A data.frame: 6 × 6

	stratum	STRATUM	EFFST1	EFFST2	EFFST3	EFFST4
	<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	1000	1000	0.9889815	0.8658658	0.7153548	0.9739408
2	10000	10000	1.0094833	0.9043117	0.8893724	0.9929141
3	11000	11000	1.0256252	0.8155081	0.7844797	1.0176709

	stratum	STRATUM	EFFST1	EFFST2	EFFST3	EFFST4
	<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
4	12000	12000	0.9886671	0.9182343	0.9122660	0.9535976
5	13000	13000	1.0007770	0.9269399	0.9133172	0.9984593
6	14000	14000	0.9868458	0.9415934	0.9278457	1.0213785

In [34]:
```
head(inp2$rho)
```

A data.frame: 6 × 9

	STRATUM	RHO_AR1	RHO_NAR1	RHO_AR2	RHO_NAR2	RHO_AR3	RHO_NAR3	RHO
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<
1	1000	1	0.00007292632	1	0.00012005614	1	-0.00002004912	
2	10000	1	0.00099005155	1	0.00002032990	1	0.00011242268	
3	11000	1	0.00702428571	1	0.00553584354	1	0.00145372109	
4	12000	1	0.04124161194	1	0.00787259701	1	-0.00622346269	
5	13000	1	0.00002008962	1	0.00001195306	1	0.00001434566	
6	14000	1	0.00011078534	1	-0.00004561257	1	-0.00003495288	

In [35]:
```
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 [36]:
```
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

	STRATUM	STRAT_MOS	DELTA	MINIMUM
	<fct>	<dbl>	<dbl>	<dbl>
5	5000	101067	1	50
6	6000	47218	1	50

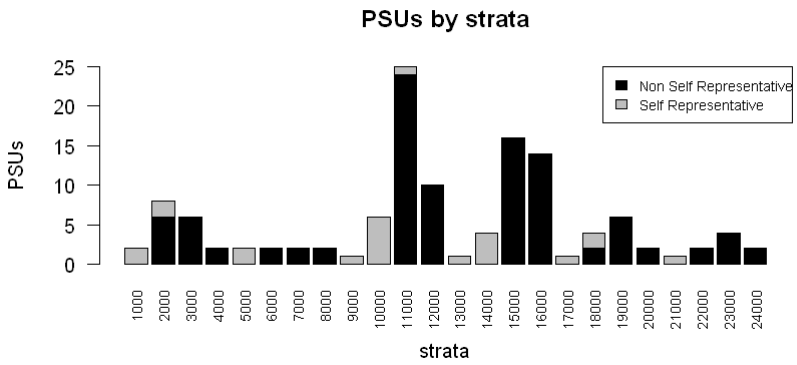
Allocation

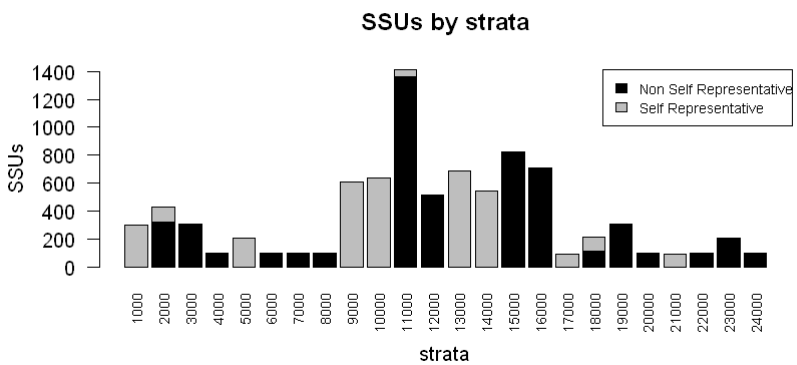
In [37]:
```
set.seed(1234)
inp2$des_file$MINIMUM <- 50
alloc2 <- beat.2st(stratif = inp2$strata,
                   errors = cv,
                   des_file = inp2$des_file,
                   psu_file = inp2$psu_file,
                   rho = inp2$rho,
                   deft_start = NULL,
                   effst = inp2$effst,
                   minnumstrat = 2,
                   minPSUstrat = 2)
```

```
  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 8259
2          1     50      78       128 8078
3          2     24     102       126 8054
4          3     23     102       125 8068
```

Selection of PSUs (I stage)

In [38]:
```
set.seed(1234)
sample_1st <- select_PSU(alloc2, type="ALLOC", pps=TRUE)
```





In [39]:

```
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	299	299	0
2000	8	2	6	431	111	320
3000	6	0	6	312	0	312
4000	2	0	2	100	0	100
5000	2	2	0	211	211	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	613	613	0
10000	6	6	0	636	636	0
11000	25	1	24	1412	50	1362
12000	10	0	10	514	0	514
13000	1	1	0	685	685	0
14000	4	4	0	547	547	0
15000	16	0	16	822	0	822
16000	14	0	14	708	0	708
17000	1	1	0	97	97	0
18000	4	2	2	218	100	118
19000	6	0	6	306	0	306
20000	2	0	2	100	0	100
21000	1	1	0	92	92	0
22000	2	0	2	100	0	100
23000	4	0	4	208	0	208
24000	2	0	2	100	0	100
Total	125	23	102	8811	3441	5370

Selection of SSUs (II stage)

In [40]:

```
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 =  78
PSU =  6  *** Selected SSU =  54
```

```
PSU =  7  *** Selected SSU =  51
PSU =  8  *** Selected SSU =  613
PSU =  11  *** Selected SSU =  113
PSU =  13  *** Selected SSU =  86
PSU =  15  *** Selected SSU =  68
PSU =  16  *** Selected SSU =  58
PSU =  22  *** Selected SSU =  62
PSU =  24  *** Selected SSU =  50
PSU =  29  *** Selected SSU =  62
PSU =  30  *** Selected SSU =  58
PSU =  31  *** Selected SSU =  51
PSU =  38  *** Selected SSU =  56
PSU =  39  *** Selected SSU =  52
PSU =  40  *** Selected SSU =  94
PSU =  41  *** Selected SSU =  63
PSU =  42  *** Selected SSU =  60
PSU =  49  *** Selected SSU =  50
PSU =  51  *** Selected SSU =  202
PSU =  52  *** Selected SSU =  53
PSU =  55  *** Selected SSU =  56
PSU =  56  *** Selected SSU =  52
PSU =  61  *** Selected SSU =  56
PSU =  63  *** Selected SSU =  51
PSU =  72  *** Selected SSU =  68
PSU =  74  *** Selected SSU =  53
PSU =  75  *** Selected SSU =  50
PSU =  76  *** Selected SSU =  54
PSU =  77  *** Selected SSU =  57
PSU =  78  *** Selected SSU =  57
PSU =  80  *** Selected SSU =  60
PSU =  84  *** Selected SSU =  56
PSU =  87  *** Selected SSU =  50
PSU =  91  *** Selected SSU =  52
PSU =  94  *** Selected SSU =  50
PSU =  96  *** Selected SSU =  51
PSU =  97  *** Selected SSU =  50
PSU =  106  *** Selected SSU =  58
PSU =  107  *** Selected SSU =  52
PSU =  110  *** Selected SSU =  50
PSU =  113  *** Selected SSU =  58
PSU =  117  *** Selected SSU =  50
PSU =  119  *** Selected SSU =  52
PSU =  121  *** Selected SSU =  102
PSU =  137  *** Selected SSU =  51
PSU =  138  *** Selected SSU =  50
PSU =  140  *** Selected SSU =  50
PSU =  161  *** Selected SSU =  50
PSU =  172  *** Selected SSU =  51
PSU =  180  *** Selected SSU =  52
PSU =  182  *** Selected SSU =  50
PSU =  183  *** Selected SSU =  51
PSU =  186  *** Selected SSU =  50
PSU =  187  *** Selected SSU =  52
PSU =  192  *** Selected SSU =  50
PSU =  196  *** Selected SSU =  50
PSU =  197  *** Selected SSU =  51
PSU =  198  *** Selected SSU =  52
PSU =  203  *** Selected SSU =  50
PSU =  210  *** Selected SSU =  51
PSU =  214  *** Selected SSU =  123
PSU =  218  *** Selected SSU =  54
PSU =  221  *** Selected SSU =  98
PSU =  222  *** Selected SSU =  50
PSU =  228  *** Selected SSU =  224
PSU =  256  *** Selected SSU =  52
```

```
PSU =  259  *** Selected SSU =  685
PSU =  268  *** Selected SSU =  52
PSU =  270  *** Selected SSU =  51
PSU =  271  *** Selected SSU =  50
PSU =  277  *** Selected SSU =  50
PSU =  278  *** Selected SSU =  50
PSU =  280  *** Selected SSU =  52
PSU =  281  *** Selected SSU =  50
PSU =  287  *** Selected SSU =  52
PSU =  289  *** Selected SSU =  54
PSU =  293  *** Selected SSU =  58
PSU =  300  *** Selected SSU =  50
PSU =  302  *** Selected SSU =  50
PSU =  304  *** Selected SSU =  57
PSU =  308  *** Selected SSU =  52
PSU =  309  *** Selected SSU =  77
PSU =  312  *** Selected SSU =  54
PSU =  313  *** Selected SSU =  52
PSU =  315  *** Selected SSU =  58
PSU =  318  *** Selected SSU =  52
PSU =  321  *** Selected SSU =  54
PSU =  330  *** Selected SSU =  222
PSU =  331  *** Selected SSU =  50
PSU =  332  *** Selected SSU =  50
PSU =  335  *** Selected SSU =  50
PSU =  338  *** Selected SSU =  50
PSU =  342  *** Selected SSU =  54
PSU =  347  *** Selected SSU =  52
PSU =  352  *** Selected SSU =  50
PSU =  354  *** Selected SSU =  50
PSU =  359  *** Selected SSU =  50
PSU =  360  *** Selected SSU =  50
PSU =  372  *** Selected SSU =  50
PSU =  379  *** Selected SSU =  50
PSU =  380  *** Selected SSU =  50
PSU =  382  *** Selected SSU =  161
PSU =  385  *** Selected SSU =  50
PSU =  389  *** Selected SSU =  52
PSU =  390  *** Selected SSU =  50
PSU =  413  *** Selected SSU =  50
PSU =  418  *** Selected SSU =  51
PSU =  424  *** Selected SSU =  59
PSU =  426  *** Selected SSU =  52
PSU =  445  *** Selected SSU =  97
PSU =  448  *** Selected SSU =  59
PSU =  449  *** Selected SSU =  50
PSU =  467  *** Selected SSU =  51
PSU =  477  *** Selected SSU =  50
PSU =  478  *** Selected SSU =  50
PSU =  485  *** Selected SSU =  50
PSU =  490  *** Selected SSU =  52
PSU =  491  *** Selected SSU =  50
PSU =  495  *** Selected SSU =  52
PSU =  496  *** Selected SSU =  92
PSU =  500  *** Selected SSU =  52
PSU =  502  *** Selected SSU =  50
PSU =  511  *** Selected SSU =  52
PSU =  512  *** Selected SSU =  50
--------------------------------
Total PSU =  125
Total SSU =  8811
--------------------------------
```

In [41]:

```
nrow(samp)
```

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

8811

8068

In [42]:

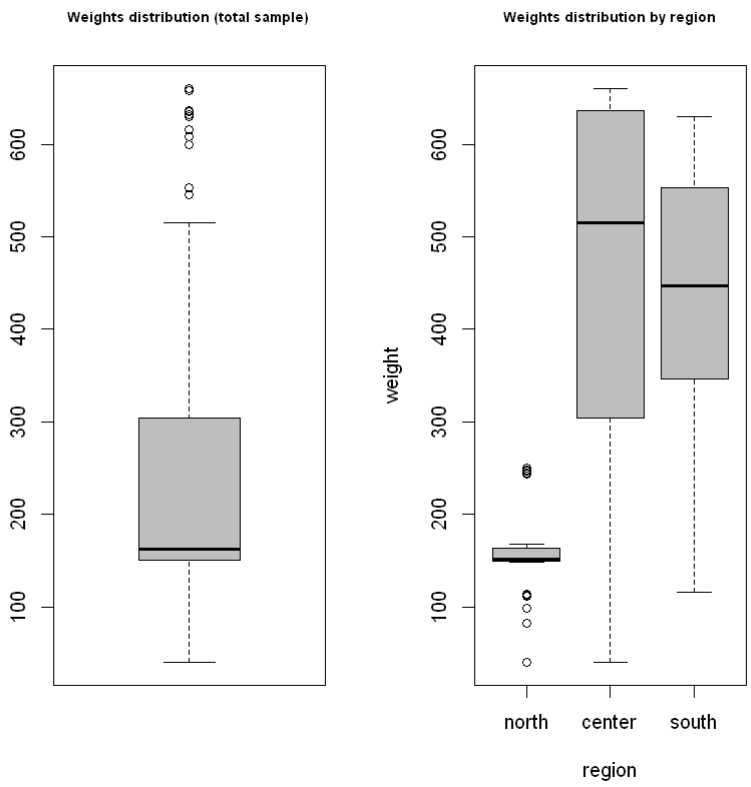
```
nrow(pop)
sum(samp$weight)
```

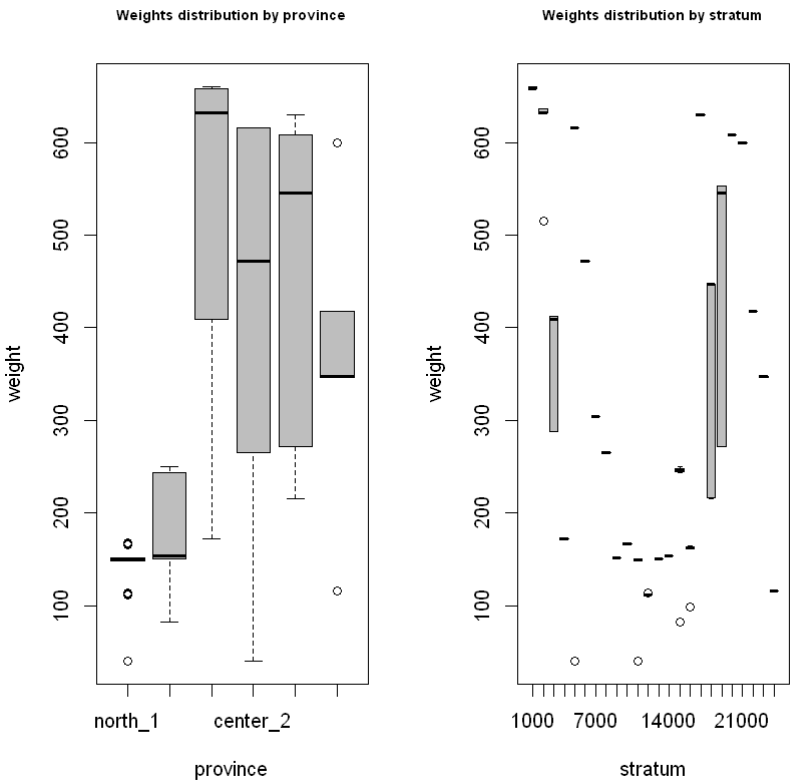
2258507

2258507

In [43]:

```
## 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 [55]:

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

In [56]:

```
# 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%
```

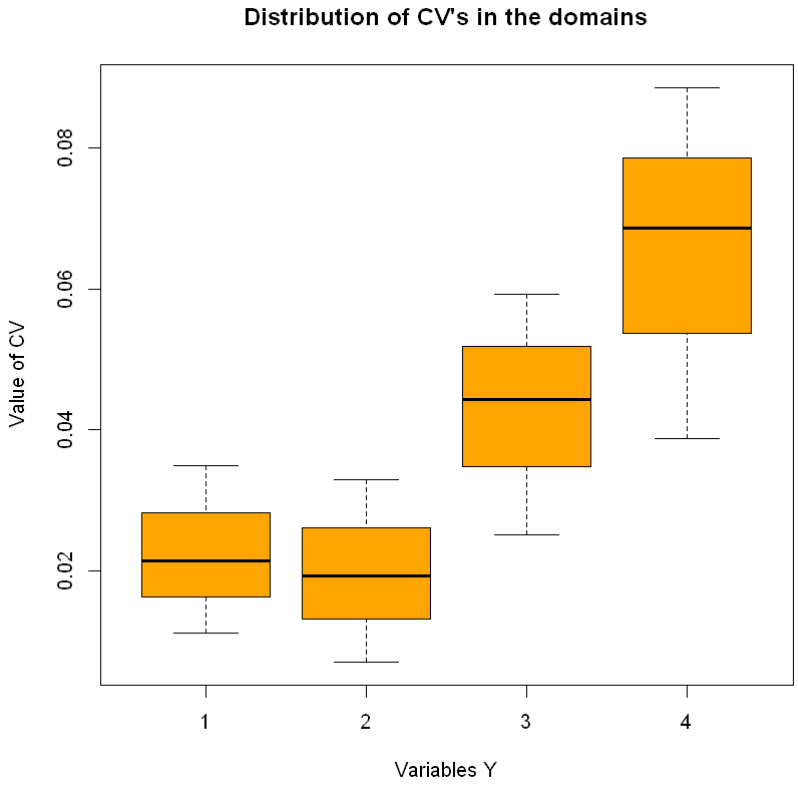
A data.frame: 1 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0106	0.0085	0.0236	0.031	DOM1

In [57]:

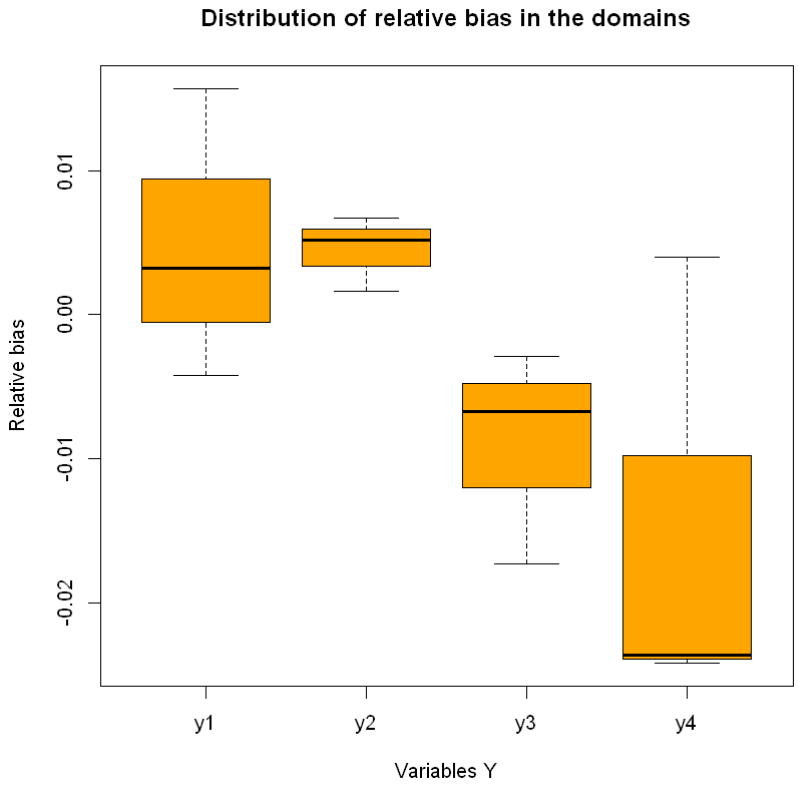
```
# Domain level = regional
domain_var <- "region"
set.seed(1234)
eval22 <- eval_2stage(df,
                      PSU_code,
                      SSU_code,
                      domain_var,
                      target_vars,
                      PSU_sampled=sample_1st$sample_PSU,
                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

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



A data.frame: 3 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0112	0.0070	0.0251	0.0886	DOM1
0.0214	0.0192	0.0444	0.0686	DOM2
0.0350	0.0329	0.0593	0.0388	DOM3



In [59]:

```
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	1158
10	DOM2	2	1	1	1	257
14	DOM2	3	1	1	197	1

In [49]:

```
save.image(file="R2BEAT_workflows.RData")
```