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: plyr

Caricamento del pacchetto richiesto: devtools

Caricamento del pacchetto richiesto: usethis

Caricamento del pacchetto richiesto: sampling
```

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

```
[1] '2.1'
```

In [3]:
```
packageVersion("R2BEAT")
```

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

In [4]:
```
## 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 [5]:
```
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 [6]:

```
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,
              f=0.05,
              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

	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

	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

	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

	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

	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

	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

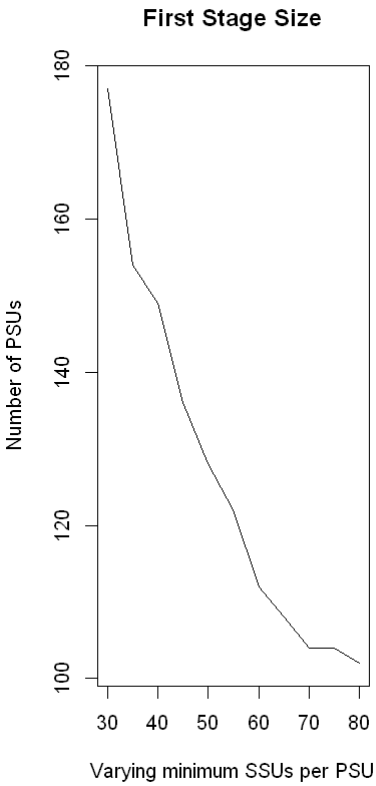
	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	19	90	109	8434
3	2	20	92	112	8434

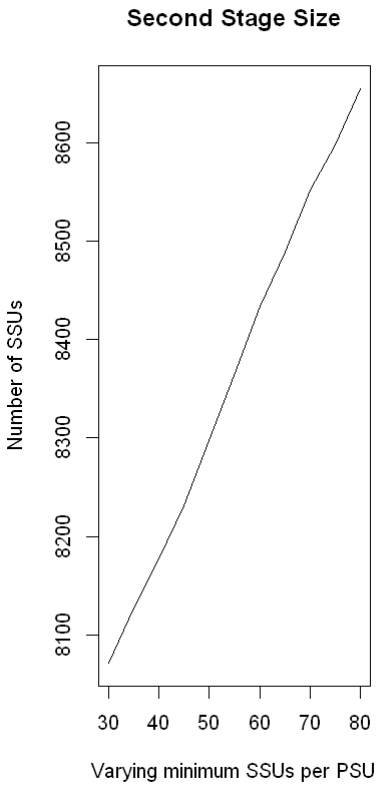
	iterations	PSU_SR	PSU NSR	PSU Total	SSU
1	0	0	0	0	7836
2	1	23	82	105	8482
3	2	18	90	108	8489

	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

	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

	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 [7]:

```
## 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
f = 0.05         # suggestion for the sampling fraction
deff_sugg <- 1.5 # suggestion for the deff value

inp <- prepareInputToAllocation1(samp_frame,
                                 id_PSU,
                                 id_SSU,
                                 strata_var,
                                 target_vars,
                                 deff_var,
                                 domain_var,
                                 minimum,
                                 delta,
                                 f,
                                 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 [8]:

```
head(inp$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 [9]:

```
head(inp$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 [10]:

```
head(inp$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 [11]:
```
head(inp$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 [12]:
```
head(inp$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 [13]:
```
head(inp$des_file)
```

A data.frame: 6 × 4

	STRATUM	STRAT_MOS	DELTA	MINIMUM
	<fct>	<dbl>	<dbl>	<dbl>

	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 [14]:

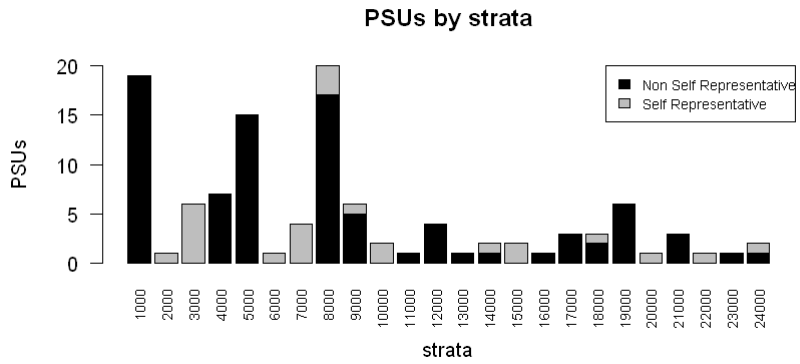
```
minPSUstrat <- 1
inp$desfile$MINIMUM <- 50
alloc <- beat.2st(stratif = inp$strata,
                  errors = cv,
                  des_file = inp$des_file,
                  psu_file = inp$psu_file,
                  rho = inp$rho,
                  deft_start = NULL,
                  effst = inp$effst,
                  epsilon1 = 5,
                  mmdiff_deft = 1,
                  maxi = 15,
                  epsilon = 10^(-11),
                  minPSUstrat,
                  minnumstrat = 2,
                  maxiter = 200,
                  maxiter1 = 25)
```

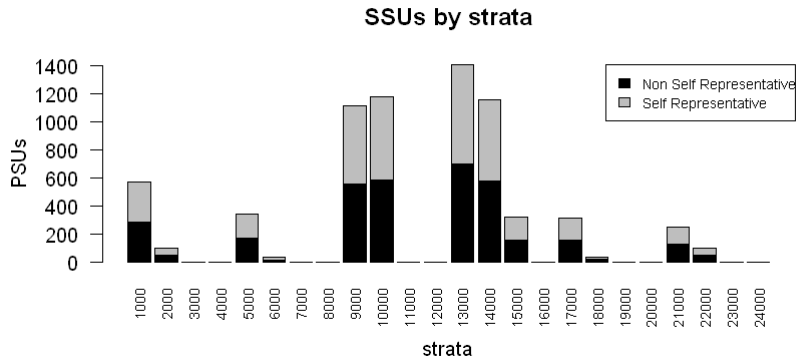
```
  iterations PSU_SR PSU NSR PSU Total   SSU
1          0      0       0         0  7836
2          1     26      82       108  8297
3          2     26      87       113 34715
4          3     44      86       130  8291
5          4     25      87       112  8296
```

Selection of PSUs (I stage)

In [15]:

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





In [16]:
```
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	19	0	19	1121	287	0
10000	2	2	0	287	588	0
11000	1	0	1	32	0	1121
12000	4	0	4	178	0	341
13000	1	0	1	51	704	0
14000	2	1	1	81	577	0
15000	2	2	0	172	161	1059
16000	1	0	1	53	0	808
17000	3	0	3	145	156	0
18000	3	1	2	133	19	114
19000	6	0	6	300	0	300
2000	1	1	0	558	52	324
20000	1	1	0	156	0	145
21000	3	0	3	127	126	0
22000	1	1	0	126	51	84
23000	1	0	1	21	0	127
24000	2	1	1	135	0	21

STRATUM	PSU	PSU_SR	PSU_NSR	SSU	SSU_SR	SSU_NSR
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
3000	6	6	0	588	0	178
4000	7	0	7	341	0	32
5000	15	0	15	808	172	0
6000	1	1	0	704	18	63
7000	4	4	0	577	0	53
8000	20	3	17	1220	0	51
9000	6	1	5	376	558	0
Total	112	25	87	8290	3469	4821

Selection of SSUs (II stage)

In [17]:

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

```
PSU =  1  *** Selected SSU =  72
PSU =  2  *** Selected SSU =  558
PSU =  3  *** Selected SSU =  51
PSU =  4  *** Selected SSU =  105
PSU =  5  *** Selected SSU =  54
PSU =  6  *** Selected SSU =  79
PSU =  7  *** Selected SSU =  52
PSU =  8  *** Selected SSU =  68
PSU =  9  *** Selected SSU =  51
PSU =  10  *** Selected SSU =  50
PSU =  11  *** Selected SSU =  57
PSU =  12  *** Selected SSU =  73
PSU =  13  *** Selected SSU =  52
PSU =  14  *** Selected SSU =  87
PSU =  15  *** Selected SSU =  58
PSU =  16  *** Selected SSU =  58
PSU =  17  *** Selected SSU =  54
PSU =  18  *** Selected SSU =  187
PSU =  19  *** Selected SSU =  56
PSU =  20  *** Selected SSU =  58
PSU =  21  *** Selected SSU =  53
PSU =  22  *** Selected SSU =  52
PSU =  23  *** Selected SSU =  15
PSU =  24  *** Selected SSU =  77
PSU =  25  *** Selected SSU =  52
PSU =  26  *** Selected SSU =  61
PSU =  27  *** Selected SSU =  54
PSU =  28  *** Selected SSU =  58
PSU =  29  *** Selected SSU =  56
PSU =  30  *** Selected SSU =  59
PSU =  31  *** Selected SSU =  61
PSU =  32  *** Selected SSU =  65
PSU =  33  *** Selected SSU =  65
PSU =  34  *** Selected SSU =  52
PSU =  35  *** Selected SSU =  54
PSU =  36  *** Selected SSU =  107
```

```
PSU =  37  *** Selected SSU =  82
PSU =  38  *** Selected SSU =  54
PSU =  39  *** Selected SSU =  53
PSU =  40  *** Selected SSU =  72
PSU =  41  *** Selected SSU =  56
PSU =  42  *** Selected SSU =  51
PSU =  43  *** Selected SSU =  54
PSU =  44  *** Selected SSU =  52
PSU =  45  *** Selected SSU =  48
PSU =  46  *** Selected SSU =  87
PSU =  47  *** Selected SSU =  62
PSU =  48  *** Selected SSU =  57
PSU =  49  *** Selected SSU =  57
PSU =  50  *** Selected SSU =  52
PSU =  51  *** Selected SSU =  130
PSU =  52  *** Selected SSU =  52
PSU =  53  *** Selected SSU =  103
PSU =  54  *** Selected SSU =  54
PSU =  55  *** Selected SSU =  237
PSU =  56  *** Selected SSU =  57
PSU =  57  *** Selected SSU =  53
PSU =  58  *** Selected SSU =  53
PSU =  59  *** Selected SSU =  66
PSU =  60  *** Selected SSU =  51
PSU =  61  *** Selected SSU =  58
PSU =  62  *** Selected SSU =  704
PSU =  63  *** Selected SSU =  55
PSU =  64  *** Selected SSU =  55
PSU =  65  *** Selected SSU =  51
PSU =  66  *** Selected SSU =  71
PSU =  67  *** Selected SSU =  51
PSU =  68  *** Selected SSU =  59
PSU =  69  *** Selected SSU =  60
PSU =  70  *** Selected SSU =  60
PSU =  71  *** Selected SSU =  63
PSU =  72  *** Selected SSU =  60
PSU =  73  *** Selected SSU =  56
PSU =  74  *** Selected SSU =  64
PSU =  75  *** Selected SSU =  15
PSU =  76  *** Selected SSU =  52
PSU =  77  *** Selected SSU =  74
PSU =  78  *** Selected SSU =  54
PSU =  79  *** Selected SSU =  52
PSU =  80  *** Selected SSU =  54
PSU =  81  *** Selected SSU =  32
PSU =  82  *** Selected SSU =  213
PSU =  83  *** Selected SSU =  91
PSU =  84  *** Selected SSU =  63
PSU =  85  *** Selected SSU =  57
PSU =  86  *** Selected SSU =  52
PSU =  87  *** Selected SSU =  18
PSU =  88  *** Selected SSU =  53
PSU =  89  *** Selected SSU =  51
PSU =  90  *** Selected SSU =  63
PSU =  91  *** Selected SSU =  3
PSU =  92  *** Selected SSU =  169
PSU =  93  *** Selected SSU =  51
PSU =  94  *** Selected SSU =  19
PSU =  95  *** Selected SSU =  29
PSU =  96  *** Selected SSU =  41
PSU =  97  *** Selected SSU =  55
PSU =  98  *** Selected SSU =  63
PSU =  99  *** Selected SSU =  51
PSU =  100  *** Selected SSU =  55
PSU =  101  *** Selected SSU =  53
```

```
PSU =  102  *** Selected SSU =  53
PSU =  103  *** Selected SSU =  54
PSU =  104  *** Selected SSU =  156
PSU =  105  *** Selected SSU =  54
PSU =  106  *** Selected SSU =  84
PSU =  107  *** Selected SSU =  21
PSU =  108  *** Selected SSU =  126
PSU =  109  *** Selected SSU =  53
PSU =  110  *** Selected SSU =  51
PSU =  111  *** Selected SSU =  23
PSU =  112  *** Selected SSU =  51
--------------------------------
Total PSU =  112
Total SSU =  8290
--------------------------------
```

In [18]:

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

8290

8296

In [19]:

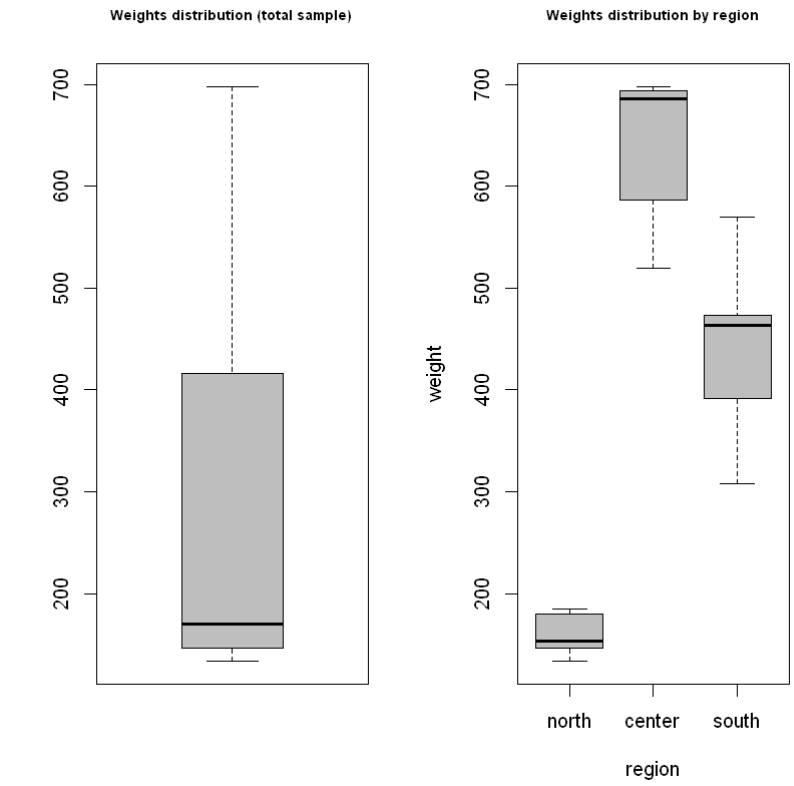
```
nrow(pop)
sum(samp$weight)
```

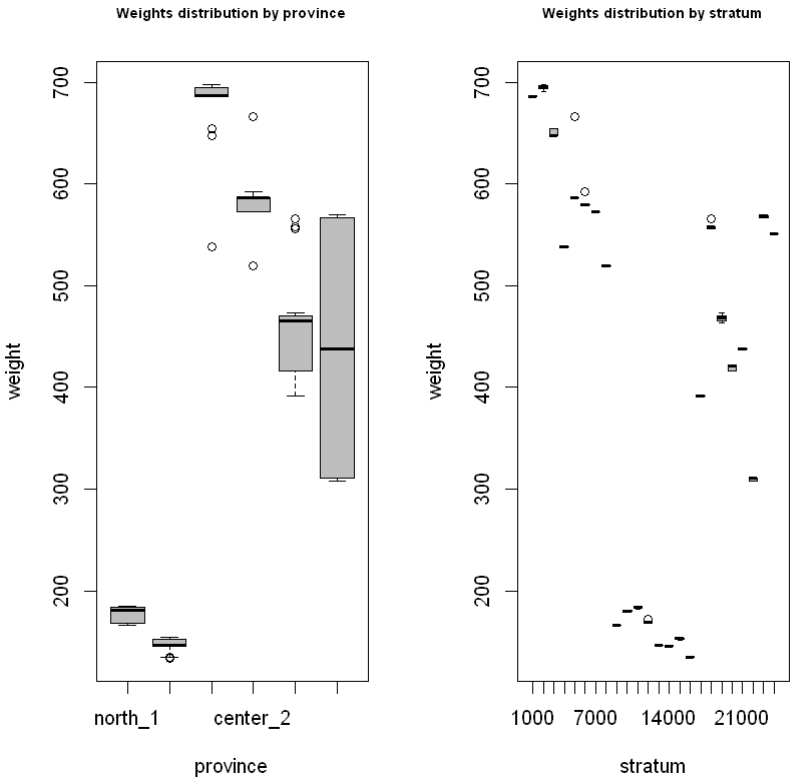
2258507

2258507

In [20]:

```
## 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 [21]:

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

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

In [22]:

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

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

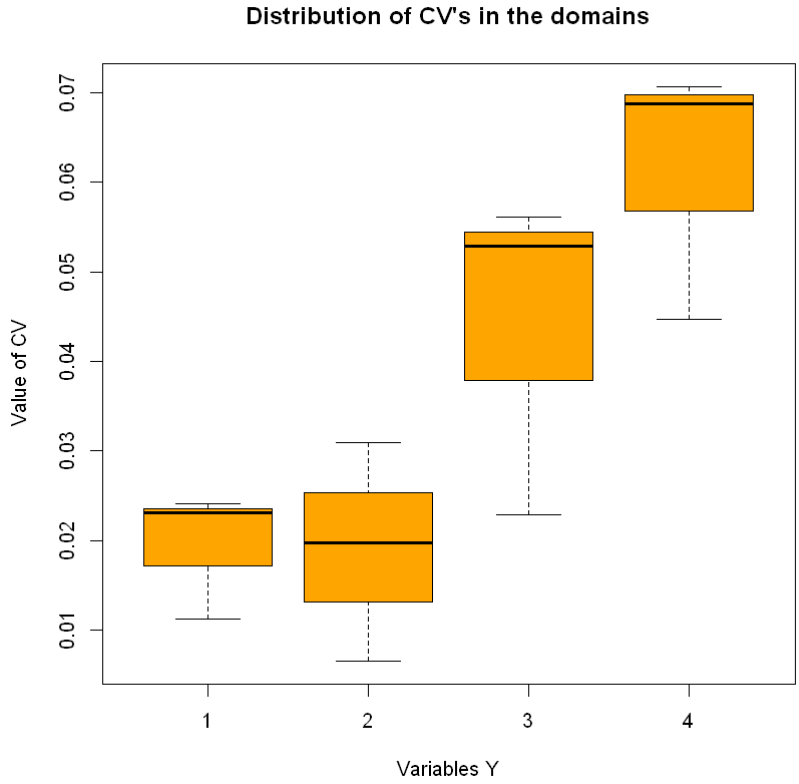
A data.frame: 1 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0104	0.0096	0.0254	0.0337	DOM1

In [23]:

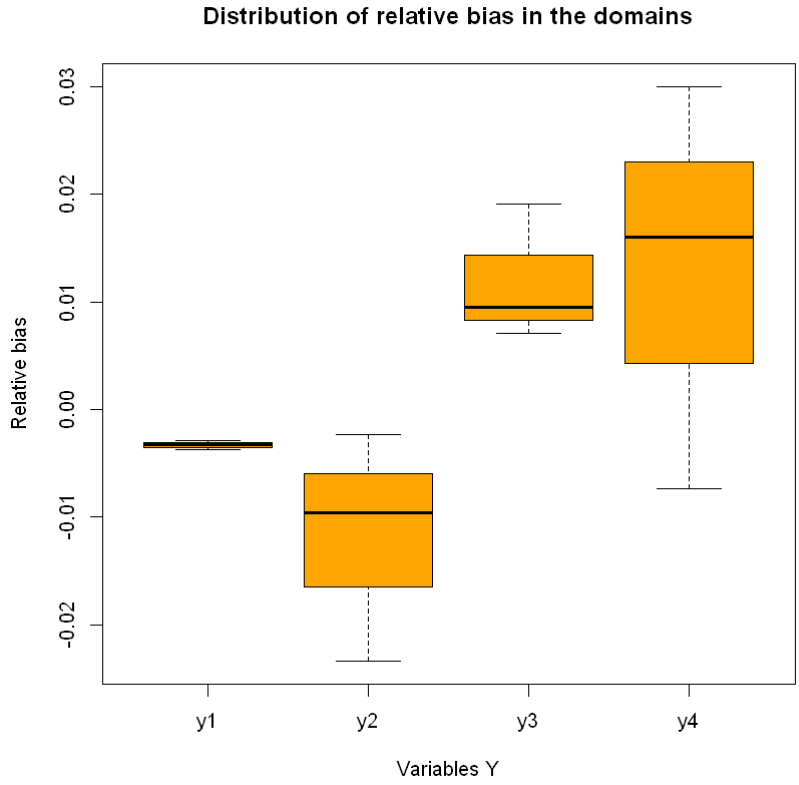
```
# Domain level = regional
domain_var <- "region"
set.seed(1234)
set.seed(1234)
eval <- eval_2stage(df,
                    PSU_code,
                    SSU_code,
                    domain_var,
                    target_vars,
                    sample_1st$sample_PSU,
                    nsampl=100,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

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



A data.frame: 3 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0113	0.0065	0.0229	0.0688	DOM1
0.0241	0.0197	0.0529	0.0707	DOM2
0.0231	0.0309	0.0561	0.0447	DOM3



In [24]:
```
alloc$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	192	1	37	1

In [25]:
```
save(samp,file="sample.RData")
```

Scenario 2

One previous round of the sampling survey is available.

Analysis of sampled data

In [26]:
```
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 [27]:

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

```
'data.frame':   8290 obs. of  20 variables:
 $ municipality : Factor w/ 112 levels "4","8","9","11",..: 31 31 31 31 31 31 31 31
31 31 ...
 $ id_ind       : int  14478 14681 14888 14914 15034 15047 15147 15169 15173 15224
...
 $ 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",..: 96659 96724 96795 96
802 96838 96844 96879 96888 96889 96910 ...
 $ stratum      : Factor w/ 24 levels "1000","2000",..: 11 11 11 11 11 11 11 11 11 1
1 ...
 $ stratum_label: chr  "north_1_5" "north_1_5" "north_1_5" "north_1_5" ...
 $ sex          : int  1 1 2 2 2 1 1 2 1 1 ...
 $ cl_age       : Factor w/ 8 levels "(0,14]","(14,24]",..: 1 5 1 4 4 2 4 2 4 4 ...
 $ active       : num  0 1 0 1 1 1 1 1 1 1 ...
 $ income_hh    : num  22289 28918 19792 25656 28864 ...
 $ unemployed   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ inactive     : num  1 0 1 0 0 0 0 0 0 0 ...
 $ Prob_1st     : num  0.484 0.484 0.484 0.484 0.484 ...
 $ Prob_2st     : num  0.0112 0.0112 0.0112 0.0112 0.0112 ...
 $ Prob_tot     : num  0.00543 0.00543 0.00543 0.00543 0.00543 ...
 $ weight       : num  184 184 184 184 184 ...
 $ 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  "110003" "110003" "110003" "110003" ...
```

In [28]:

```
## 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]:

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

```
# All lonely strata (87) successfully collapsed!

Warning message in collapse.strata(sample.des):
"No similarity score specified: achieved strata aggregation depends on the ordering
of sample data"
```

In [30]:

```
## 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]:

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

inp <- 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_vars, # strata variables
    target_vars, # target variables
    deff_vars,   # deff variables
    domain_vars, # domain variables
    delta,       # Average number of SSUs for each selection unit
    minimum      # Minimum number of SSUs to be selected in each PSU
  )
```

In [32]:

```
head(inp$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	196095	25283.04	0.6908830	0.2436486	0.06546845	23223.11	0.4621295	0.4
2	10000	10000	106119	28972.56	0.7762824	0.2033010	0.02041662	21596.37	0.4167350	0.4
3	11000	11000	205701	28591.64	0.7829250	0.1983132	0.01876181	35059.27	0.4122541	0.3
4	12000	12000	57396	25963.68	0.7575331	0.2277654	0.01470142	15511.53	0.4285752	0.4
5	13000	13000	103266	27476.83	0.7837272	0.1775604	0.03871240	22898.40	0.4117024	0.3
6	14000	14000	83998	24287.14	0.7602083	0.2065345	0.03325712	17173.44	0.4269562	0.4

In [33]:

```
head(inp$deff)
```

A data.frame: 6 × 7

	stratum	STRATUM	DEFF1	DEFF2	DEFF3	DEFF4	b_nar
	<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1000	1000	0.999876	0.999528	1.000412	1.004149	143.50000
2	10000	10000	1.027703	1.001044	0.988835	1.174327	98.00000
3	11000	11000	0.592872	0.457294	0.681566	1.838375	59.00000
4	12000	12000	3.071332	0.887471	0.856744	0.856610	48.71429
5	13000	13000	1.018096	1.022899	1.015849	1.001630	704.00000
6	14000	14000	1.014614	1.002947	1.002986	1.011830	144.25000

In [34]:

```
head(inp$effst)
```

A data.frame: 6 × 6

	stratum	STRATUM	EFFST1	EFFST2	EFFST3	EFFST4
	<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	1000	1000	0.9486306	0.7988180	0.7076215	1.0322402
2	10000	10000	0.9960368	0.8951515	0.8772252	0.9968016
3	11000	11000	1.0379964	0.9650269	0.8209022	1.0018701
4	12000	12000	0.9549886	0.8942016	0.8794657	1.0037357

	stratum	STRATUM	EFFST1	EFFST2	EFFST3	EFFST4
	<fct>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
5	13000	13000	0.9973725	0.9187114	0.8945896	1.0074768
6	14000	14000	0.9979007	0.9206678	0.9005948	1.0090391

In [35]:
```
head(inp$rho)
```

A data.frame: 6 × 9

	STRATUM	RHO_AR1	RHO_NAR1	RHO_AR2	RHO_NAR2	RHO_AR3	RHO_NAR3
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1000	1	-0.0000008701754	1	-0.000003312281	1	0.000002891228
2	10000	1	0.0002855979381	1	0.000010762887	1	-0.000115103093
3	11000	1	-0.0070194482759	1	-0.009357000000	1	-0.005490241379
4	12000	1	0.0434111497006	1	-0.002358392216	1	-0.003002371257
5	13000	1	0.0000257411095	1	0.000032573257	1	0.000022544808
6	14000	1	0.0001020174520	1	0.000020572426	1	0.000020844677

In [36]:
```
head(inp$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(inp$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

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

Allocation

In [38]:

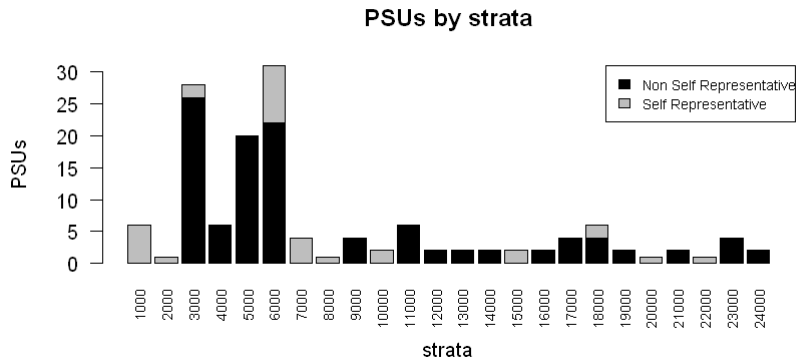
```
set.seed(1234)
minPSUstrat <- 2
inp$des_file$MINIMUM <- 50
alloc <- beat.2st(stratif = inp$strata,
                  errors = cv,
                  des_file = inp$des_file,
                  psu_file = inp$psu_file,
                  rho = inp$rho,
                  deft_start = NULL,
                  effst = inp$effst,
                  epsilon1 = 5,
                  mmdiff_deft = 1,
                  maxi = 15,
                  epsilon = 10^(-11),
                  minnumstrat = 2,
                  minPSUstrat,
                  maxiter = 200,
                  maxiter1 = 25)
```

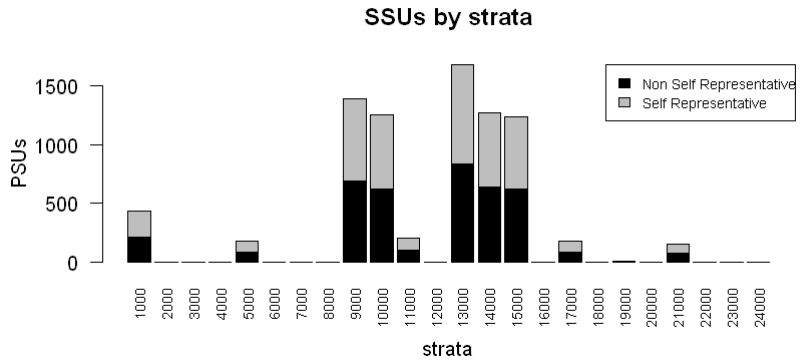
```
  iterations PSU_SR PSU NSR PSU Total  SSU
1          0      0       0         0 7402
2          1     44      72       116 9745
3          2     37     110       147 8717
4          3     30     106       136 9066
5          4     31     110       141 9026
```

Selection of PSUs (I stage)

In [39]:

```
set.seed(1234)
sample_1st <- select_PSU(alloc, 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	6	6	0	626	218	0
10000	2	2	0	218	626	0
11000	6	0	6	270	102	1396
12000	2	0	2	2	0	268
13000	2	0	2	2	839	0
14000	2	0	2	2	635	0
15000	2	2	0	88	619	1254
16000	2	0	2	28	0	1038
17000	4	0	4	124	88	0
18000	6	2	4	216	0	44
19000	2	0	2	44	6	210
2000	1	1	0	693	0	270
20000	1	1	0	88	0	124
21000	2	0	2	2	79	0
22000	1	1	0	79	0	56
23000	4	0	4	188	0	188
24000	2	0	2	56	0	2

STRATUM	PSU	PSU_SR	PSU_NSR	SSU	SSU_SR	SSU_NSR
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
3000	28	2	26	1498	0	134
4000	6	0	6	268	0	2
5000	20	0	20	1038	88	0
6000	31	9	22	1873	0	28
7000	4	4	0	635	0	2
8000	1	1	0	839	0	2
9000	4	0	4	134	693	0
Total	141	31	110	9011	3993	5018

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 =  1  *** Selected SSU =  77
PSU =  2  *** Selected SSU =  58
PSU =  3  *** Selected SSU =  693
PSU =  4  *** Selected SSU =  112
PSU =  5  *** Selected SSU =  84
PSU =  6  *** Selected SSU =  51
PSU =  7  *** Selected SSU =  52
PSU =  8  *** Selected SSU =  66
PSU =  9  *** Selected SSU =  52
PSU =  10  *** Selected SSU =  30
PSU =  11  *** Selected SSU =  56
PSU =  12  *** Selected SSU =  56
PSU =  13  *** Selected SSU =  52
PSU =  14  *** Selected SSU =  50
PSU =  15  *** Selected SSU =  92
PSU =  16  *** Selected SSU =  62
PSU =  17  *** Selected SSU =  50
PSU =  18  *** Selected SSU =  199
PSU =  19  *** Selected SSU =  51
PSU =  20  *** Selected SSU =  59
PSU =  21  *** Selected SSU =  51
PSU =  22  *** Selected SSU =  52
PSU =  23  *** Selected SSU =  58
PSU =  24  *** Selected SSU =  51
PSU =  25  *** Selected SSU =  52
PSU =  26  *** Selected SSU =  58
PSU =  27  *** Selected SSU =  52
PSU =  28  *** Selected SSU =  52
PSU =  29  *** Selected SSU =  52
PSU =  30  *** Selected SSU =  58
PSU =  31  *** Selected SSU =  56
PSU =  32  *** Selected SSU =  30
PSU =  33  *** Selected SSU =  52
PSU =  34  *** Selected SSU =  30
PSU =  35  *** Selected SSU =  58
```

```
PSU =  36  *** Selected SSU =  52
PSU =  37  *** Selected SSU =  56
PSU =  38  *** Selected SSU =  58
PSU =  39  *** Selected SSU =  30
PSU =  40  *** Selected SSU =  59
PSU =  41  *** Selected SSU =  66
PSU =  42  *** Selected SSU =  81
PSU =  43  *** Selected SSU =  52
PSU =  44  *** Selected SSU =  118
PSU =  45  *** Selected SSU =  54
PSU =  46  *** Selected SSU =  58
PSU =  47  *** Selected SSU =  63
PSU =  48  *** Selected SSU =  52
PSU =  49  *** Selected SSU =  52
PSU =  50  *** Selected SSU =  52
PSU =  51  *** Selected SSU =  54
PSU =  52  *** Selected SSU =  56
PSU =  53  *** Selected SSU =  54
PSU =  54  *** Selected SSU =  47
PSU =  55  *** Selected SSU =  56
PSU =  56  *** Selected SSU =  62
PSU =  57  *** Selected SSU =  52
PSU =  58  *** Selected SSU =  52
PSU =  59  *** Selected SSU =  64
PSU =  60  *** Selected SSU =  31
PSU =  61  *** Selected SSU =  80
PSU =  62  *** Selected SSU =  71
PSU =  63  *** Selected SSU =  64
PSU =  64  *** Selected SSU =  58
PSU =  65  *** Selected SSU =  51
PSU =  66  *** Selected SSU =  88
PSU =  67  *** Selected SSU =  52
PSU =  68  *** Selected SSU =  54
PSU =  69  *** Selected SSU =  52
PSU =  70  *** Selected SSU =  52
PSU =  71  *** Selected SSU =  143
PSU =  72  *** Selected SSU =  58
PSU =  73  *** Selected SSU =  51
PSU =  74  *** Selected SSU =  114
PSU =  75  *** Selected SSU =  52
PSU =  76  *** Selected SSU =  260
PSU =  77  *** Selected SSU =  54
PSU =  78  *** Selected SSU =  50
PSU =  79  *** Selected SSU =  52
PSU =  80  *** Selected SSU =  64
PSU =  81  *** Selected SSU =  47
PSU =  82  *** Selected SSU =  54
PSU =  83  *** Selected SSU =  839
PSU =  84  *** Selected SSU =  55
PSU =  85  *** Selected SSU =  50
PSU =  86  *** Selected SSU =  60
PSU =  87  *** Selected SSU =  55
PSU =  88  *** Selected SSU =  68
PSU =  89  *** Selected SSU =  52
PSU =  90  *** Selected SSU =  62
PSU =  91  *** Selected SSU =  31
PSU =  92  *** Selected SSU =  50
PSU =  93  *** Selected SSU =  72
PSU =  94  *** Selected SSU =  72
PSU =  95  *** Selected SSU =  68
PSU =  96  *** Selected SSU =  62
PSU =  97  *** Selected SSU =  58
PSU =  98  *** Selected SSU =  57
PSU =  99  *** Selected SSU =  26
PSU =  100  *** Selected SSU =  52
```

```
PSU =  101  *** Selected SSU =  57
PSU =  102  *** Selected SSU =  56
PSU =  103  *** Selected SSU =  17
PSU =  104  *** Selected SSU =  1
PSU =  105  *** Selected SSU =  52
PSU =  106  *** Selected SSU =  26
PSU =  107  *** Selected SSU =  17
PSU =  108  *** Selected SSU =  50
PSU =  109  *** Selected SSU =  50
PSU =  110  *** Selected SSU =  162
PSU =  111  *** Selected SSU =  1
PSU =  112  *** Selected SSU =  1
PSU =  113  *** Selected SSU =  1
PSU =  114  *** Selected SSU =  14
PSU =  115  *** Selected SSU =  14
PSU =  116  *** Selected SSU =  1
PSU =  117  *** Selected SSU =  2
PSU =  118  *** Selected SSU =  86
PSU =  119  *** Selected SSU =  1
PSU =  120  *** Selected SSU =  12
PSU =  121  *** Selected SSU =  53
PSU =  122  *** Selected SSU =  22
PSU =  123  *** Selected SSU =  22
PSU =  124  *** Selected SSU =  50
PSU =  125  *** Selected SSU =  53
PSU =  126  *** Selected SSU =  88
PSU =  127  *** Selected SSU =  52
PSU =  128  *** Selected SSU =  12
PSU =  129  *** Selected SSU =  3
PSU =  130  *** Selected SSU =  50
PSU =  131  *** Selected SSU =  3
PSU =  132  *** Selected SSU =  52
PSU =  133  *** Selected SSU =  28
PSU =  134  *** Selected SSU =  1
PSU =  135  *** Selected SSU =  40
PSU =  136  *** Selected SSU =  79
PSU =  137  *** Selected SSU =  54
PSU =  138  *** Selected SSU =  28
PSU =  139  *** Selected SSU =  40
PSU =  140  *** Selected SSU =  1
PSU =  141  *** Selected SSU =  54
--------------------------------
Total PSU =  141
Total SSU =  9011
--------------------------------
```

In [42]:

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

9011

9026

In [43]:

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

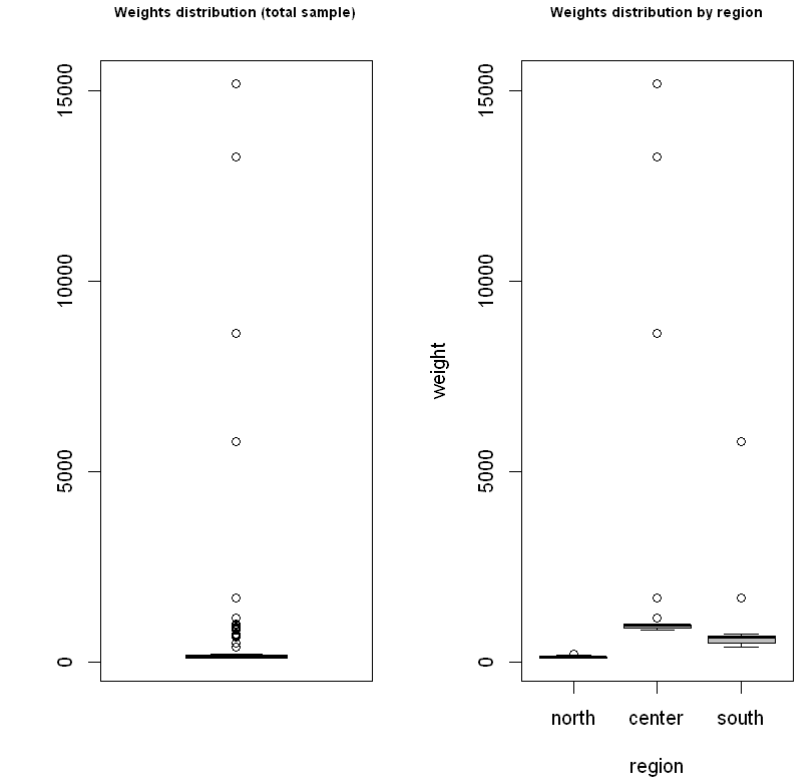
2258507

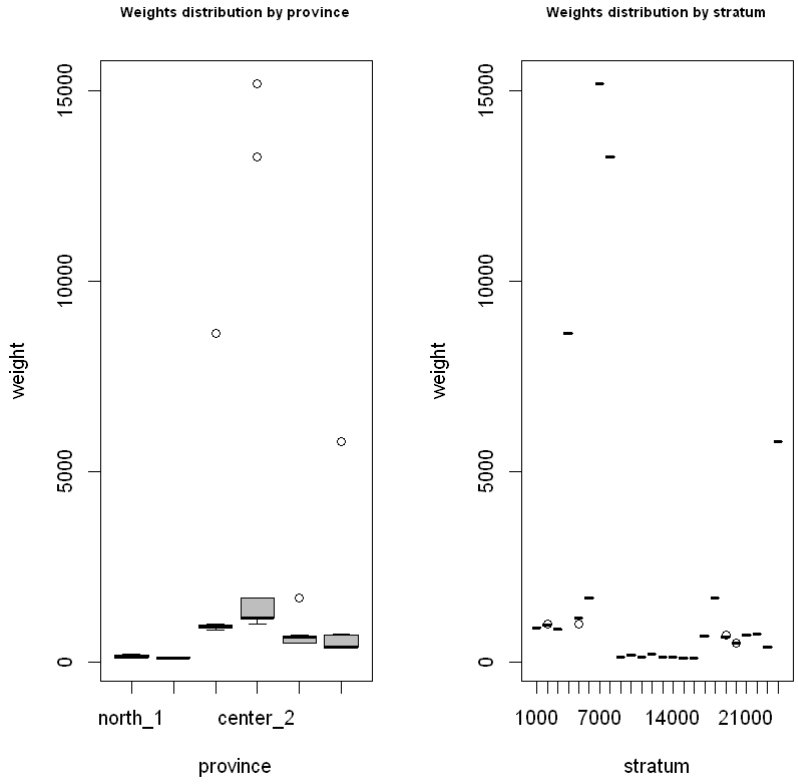
2258507

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)
eval <- eval_2stage(df,
                    PSU_code,
                    SSU_code,
                    domain_var,
                    target_vars,
                    PSU_sampled=sample_1st$sample_PSU,
                    nsampl=100,
                    writeFiles=FALSE,
                    progress=TRUE)
eval$coeff_var
```

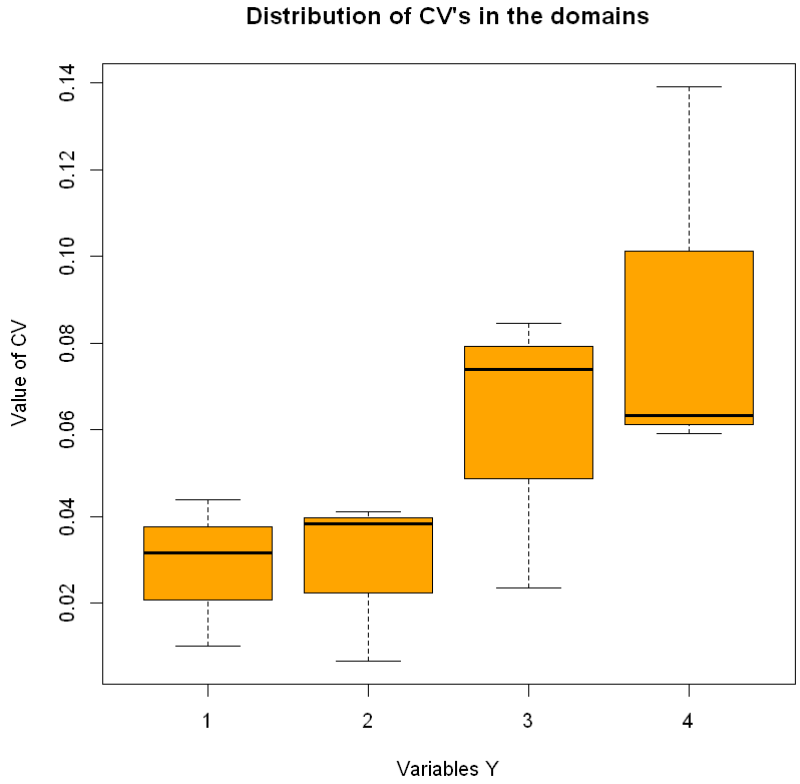
```
  |======================================================================| 100%
```

A data.frame: 1 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0162	0.0141	0.0391	0.059	DOM1

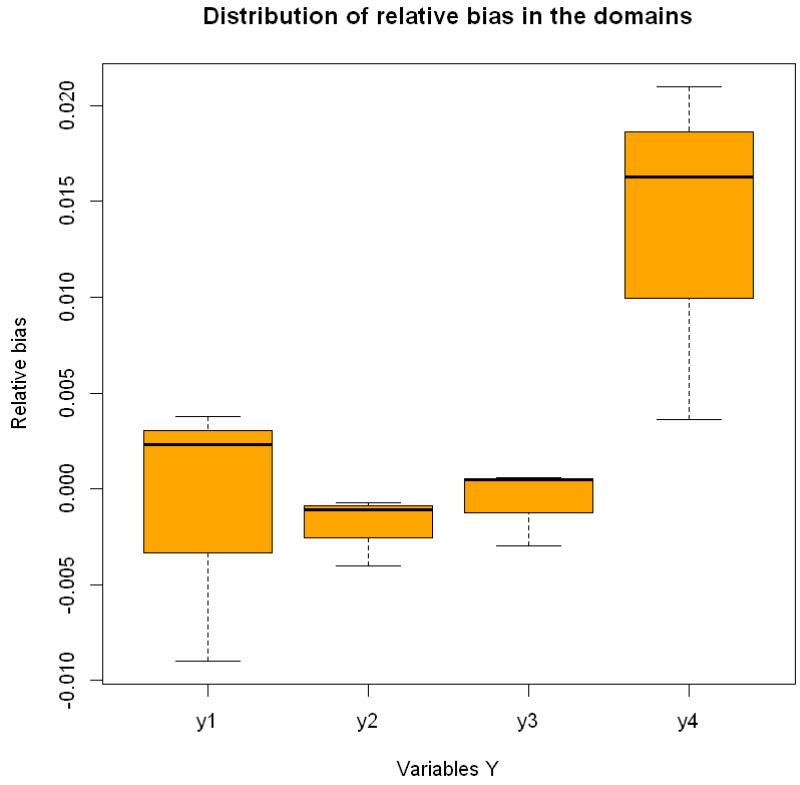
In [47]:
```
# Domain level = regional
domain_var <- "region"
set.seed(1234)
eval <- eval_2stage(df,
                    PSU_code,
                    SSU_code,
                    domain_var,
                    target_vars,
                    PSU_sampled=sample_1st$sample_PSU,
                    nsampl=100,
                    writeFiles=FALSE,
                    progress=TRUE)
eval$coeff_var
```

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



A data.frame: 3 × 5

CV1	CV2	CV3	CV4	dom
<dbl>	<dbl>	<dbl>	<dbl>	<chr>
0.0101	0.0066	0.0235	0.0633	DOM1
0.0438	0.0383	0.0845	0.1392	DOM2
0.0316	0.0412	0.0739	0.0593	DOM3



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

A data.frame: 4 × 6

	Type	Dom	V1	V2	V3	V4
	<chr>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
2	DOM1	1	1	0	1	78
6	DOM2	1	0	0	0	1495
10	DOM2	2	114	1	1	7
14	DOM2	3	1	1	110	1

In [49]:
```
save.image(file="R2BEAT_workflows.RData")
```