

Package ‘surveyplanning’

May 9, 2026

Type Package

Title Survey Planning Tools

Version 4.0

Date 2020-05-20

Depends R (>= 3.0.0)

Imports data.table (>= 1.11.4), laeken, stats

Maintainer Juris Breidaks <r~~csb~~@csb.gov.lv>

Description Tools for sample survey planning, including sample size calculation, estimation of expected precision for the estimates of totals, and calculation of optimal sample size allocation.

License GPL (>= 2)

Encoding UTF-8

Repository CRAN

URL <https://csblatvia.github.io/surveyplanning/>

BugReports <https://github.com/CSBLatvia/surveyplanning/issues/>

NeedsCompilation yes

LazyData true

RoxygenNote 7.1.0

Author Juris Breidaks [aut, cre],
Martins Liberts [aut],
Janis Jukams [aut]

Date/Publication 2020-05-20 11:20:02 UTC

Contents

surveyplanning-package	2
dom_optimal_allocation	3
expsize	6
expvar	7
min_count	10

min_prop	11
MoE_P	12
MoE_Y	13
optsize	14
prop_dom_optimal_allocation	16
round2	18
s2	19

Index	20
--------------	-----------

surveyplanning-package

Survey Planning Tools

Description

Tools for sample survey planning, including sample size calculation, estimation of expected precision for the estimates of totals, and calculation of optimal sample size allocation.

Details

Package: surveyplanning
 Version: 2.9
 Date: 2017-10-26
 Depends: R (>= 3.0.0), data.table (>= 1.10.4), stats, laeken
 License: GPL (>= 2)
 URL: <https://github.com/CSBLatvia/surveyplanning/>
 BugReports: <https://github.com/CSBLatvia/surveyplanning/issues/>

Index:

dom_optimal_allocation	Optimal sample size allocation
expsize	Sample size calculation
expvar	Expected precision for the estimates of totals
min_count	Minimal count of respondents for the given relative margin of error
min_prop	Minimal proportion for the given relative margin of error
MoE_Y	Margin of error for count
MoE_P	Margin of error for proportion
optsize	Optimal sample size allocation
s2	Population variance estimation
surveyplanning-package	Survey Planning Tools

Author(s)

Juris Breidaks [aut, cre], Martins Liberts [aut], Janis Jukams [aut]

Maintainer: Juris Breidaks <rscsb@csb.gov.lv>

 dom_optimal_allocation

Optimal sample size allocation

Description

The function computes optimal sample size allocation over strata and domain for population.

Usage

```
dom_optimal_allocation(
  id,
  Dom,
  H,
  Y,
  Rh = NULL,
  deffh = NULL,
  indicator,
  sup_w,
  sup_cv,
  min_size = 3,
  correction_before = FALSE,
  dataset = NULL
)
```

Arguments

id	Variable for unit ID codes. One dimensional object convertible to one-column data.table or variable name as character, column number.
Dom	Optional variables used to define population domains. If supplied, values are calculated for each domain. An object convertible to data.table or variable names as character vector, column numbers.
H	The unit stratum variable. One dimensional object convertible to one-column data.table or variable name as character, column number.
Y	Variable of interest. Object convertible to data.table or variable names as character, column numbers.
Rh	The expected response rate in each stratum (optional). If not defined, it is assumed to be 1 in each stratum (full-response). Object convertible to one-column data.table, variable name as character, or column number.
deffh	The expected design effect for the estimate of variable (optional). If not defined, it is assumed to be 1 for each variable in each stratum. If is defined, then variables is defined the same arrangement as Yh. Object convertible to data.table, variable name as character vector, or column numbers.
indicator	Variable for detection fully surveyed units. Object convertible to data.table or variable names as character, column numbers.

sup_w	Variable for weight limit in domain of stratum. Object convertible to <code>data.table</code> or variable names as character, column numbers.
sup_cv	Variable for maximum coefficient of variation (CV) in percentage for domain. Object convertible to <code>data.table</code> or variable names as character, column numbers.
min_size	A numeric value for sample size.
correction_before	by default FALSE; correction of sample size is made before ending, if true, correction of sample size is made at the end.
dataset	Optional survey data object convertible to <code>data.table</code> with one row for each stratum.

Value

A list with eight data objects:

data	An object as <code>data.table</code> , with variables: <code>id</code> - variable with unit ID codes, <code>Dom</code> - optional variables used to define population domains, <code>H</code> - the unit stratum variable, <code>Y</code> - variable of interest, <code>Rh</code> - the expected response rate in each stratum, <code>deffh</code> - the expected design effect, <code>indicator</code> - variable for full surveys, <code>sup_w</code> - variable for weight limit in domain of stratum, <code>sup_cv</code> - Variable for maximum coefficient of variation, <code>poph</code> - population size, <code>nh</code> - sample size .
nh_larger_than_Nh	An object as <code>data.table</code> , with variables: <code>H</code> - the stratum variable, <code>nh</code> - sample size, <code>poph</code> - population size.
dom_strata_size	An object as <code>data.table</code> , with variables: <code>H</code> - the unit stratum variable, <code>Dom</code> - optional variables used to define population domains, <code>sup_w</code> - variable for weight limit in domain of stratum, <code>poph</code> - population size, <code>nh</code> - sample size, <code>sample100</code> - sample size for fully surveyed units, <code>design_weights</code> - design weights.
dom_size	An object as <code>data.table</code> , with variables: <code>Dom</code> - optional variables used to define population domains, <code>poph</code> - population size, <code>nh</code> - sample size, <code>sample100</code> - sample size for fully surveyed units, <code>design_weights</code> - design weights.

size An object as `data.table`, with variables:
 poph - population size,
 nh - sample size,
 sample100 - sample size for fully surveyed units.

dom_strata_expected_precision
 An object as `data.table`, with variables:
 H - stratum,
 variable - the name of variable of interest,
 estim - total value,
 deffh - the expected design effect,
 s2h - population variance S^2 ,
 nh - sample size,
 Rh - the expected response rate,
 deffh - the expected design effect,
 poph - population size,
 nrh - expected number of respondents,
 var - expected variance,
 se - expected standard error,
 cv - expected coefficient of variance.

dom_expected_precision
 An object as `data.table`, with variables:
 Dom - domain,
 variable - the name of variable of interest,
 poph - the population size,
 nh - sample size,
 nrh - expected number of respondents,
 estim - total value,
 var - the expected variance,
 se - the expected standard error,
 cv - the expected coefficient of variance.

total_expected_precision
 An object as `data.table`, with variables:
 variable - the name of variable of interest,
 poph - the population size,
 nh - sample size,
 nrh - expected number of respondents,
 estim - total value,
 var - the expected variance,
 se - the expected standard error,
 cv - the expected coefficient of variance.

See Also

[expsize](#), [optsize](#), [prop_dom_optimal_allocation](#)

Examples

```
library("laeken")
library("data.table")
```

```

data("ses")
data <- data.table(ses)
data[, H := paste(location, NACE1, size, sep = "_")]
data[, id := .I]
data[, full := 0]
data[, sup_cv := 10]
data[, sup_w := 20]
#vars <- dom_optimal_allocation(id = "id", dom = "sex",
#                               H = "H", Y = "earnings",
#                               indicator = "full",
#                               sup_w = "sup_w",
#                               sup_cv = "sup_cv",
#                               min_size = 3,
#                               correction_before = FALSE,
#                               dataset = data)
#vars
dataset=data)
#vars

```

expsize

Sample size calculation

Description

The function computes minimum sample size for each stratum to achieve defined precision (CV) for the estimates of totals in each stratum. The calculation takes into account expected totals, population variance, expected response rate and design effect in each stratum.

Usage

```
expsize(Yh, H, s2h, poph, Rh = NULL, deffh = NULL, CVh, dataset = NULL)
```

Arguments

Yh	The expected totals for variables of interest in each stratum. Object convertible to <code>data.table</code> , variable names as character vector, or column numbers.
H	The stratum variable. One dimensional object convertible to one-column <code>data.table</code> , variable name as character, or column number.
s2h	The expected population variance S^2 for variables of interest in each stratum. Object convertible to <code>data.table</code> , variable name as character vector, or column numbers.
poph	Population size in each stratum. One dimensional object convertible to one-column <code>data.table</code> , variable name as character, or column number.
Rh	The expected response rate in each stratum (optional). If not defined, it is assumed to be 1 in each stratum (full-response). Object convertible to one-column <code>data.table</code> , variable name as character, or column number.
deffh	The expected design effect for the estimates of totals (optional). If not defined, it is assumed to be 1 for each variable in each stratum. Object convertible to <code>data.table</code> , variable name as character vector, or column numbers.

CVh	Coefficient of variation (in percentage) to be achieved for each stratum. One dimensional object convertible to one-column data.table, variable name as character, or column number.
dataset	Optional survey data object convertible to data.table with one row for each stratum.

Value

A data.table is returned by the function, with variables:
 H - stratum,
 variable - the name of variable of interest,
 estim - total value,
 deffh - the expected design effect,
 s2h - population variance S^2 ,
 CVh - the expected coefficient of variation,
 Rh - the expected response rate,
 poph - population size,
 nh - minimal sample size to achieve defined precision (CV).

See Also

[expvar](#), [optsize](#), [MoE_P](#)

Examples

```
library("data.table")
data <- data.table(H = 1:3, Yh = 10 * 1:3,
  Yh1 = 10 * 4:6, s2h = 10 * runif(3),
  s2h2 = 10 * runif(3), CVh = rep(4.9,3),
  poph = 8 * 1:3, Rh = rep(1, 3),
  deffh = rep(2, 3), deffh2 = rep(3, 3))

size <- expsize(Yh = c("Yh", "Yh1"), H = "H",
  s2h = c("s2h", "s2h2"), poph = "poph",
  Rh = "Rh", deffh = c("deffh", "deffh2"),
  CVh = "CVh", dataset = data)

size
```

expvar

Expected precision for the estimates of totals

Description

The function computes expected precision as variance, standard error, and coefficient of variation for the estimates.

Usage

```
expvar(
  Yh,
  Zh = NULL,
  H,
  s2h,
  nh,
  poph,
  Rh = NULL,
  deffh = NULL,
  Dom = NULL,
  dataset = NULL
)
```

Arguments

Yh	The expected totals for variables of interest in each stratum. Object convertible to <code>data.table</code> , variable names as character vector, or column numbers.
Zh	Optional variables of denominator for the expected ratio estimation in each stratum. Object convertible to <code>data.table</code> , variable names as character vector, or column numbers.
H	The stratum variable. One dimensional object convertible to one-column <code>data.table</code> , variable name as character, or column number.
s2h	The expected population variance S^2 for variables of interest in each stratum. Variables is defined the same arrangement as Yh. Object convertible to <code>data.table</code> , variable name as character vector, or column numbers.
nh	Sample size in each stratum. One dimensional object convertible to one-column <code>data.table</code> , variable name as character, or column number.
poph	Population size in each stratum. One dimensional object convertible to one-column <code>data.table</code> , variable name as character, or column number.
Rh	The expected response rate in each stratum (optional). If not defined, it is assumed to be 1 in each stratum (full-response). Object convertible to one-column <code>data.table</code> , variable name as character, or column number.
deffh	The expected design effect for the estimates of totals (optional). If not defined, it is assumed to be 1 for each variable in each stratum. If is defined, then variables is defined the same arrangement as Yh. Object convertible to <code>data.table</code> , variable name as character vector, or column numbers.
Dom	Optional variables used to define population domains. Only domains as unions of strata can be defined. If supplied, estimated precision is calculated for each domain. An object convertible to <code>data.table</code> , variable names as character vector, or column numbers.
dataset	Optional survey data object convertible to <code>data.table</code> with one row for each stratum.

Value

A list with three data objects:

resultH	An object as data.table, with variables: H - stratum, variableY - the name of variable of interest, variableZ - the name of optional variable of denominator for the expected ratio estimation, estim - total value, deffh - the expected design effect, s2h - population variance S^2 , nh - sample size, Rh - the expected response rate, poph - population size, nrh - expected number of respondents, var - expected variance, se - expected standard error, cv - expected coefficient of variance.
resultDom	An object as data.table, with variables: Dom - domain, variableY - the name of variable of interest, variableZ - the name of optional variable of denominator for the expected ratio estimation, poph - the population size, nh - sample size, nrh - expected number of respondents, estim - total value, var - the expected variance, se - the expected standart error, cv - the expected coefficient of variance.
result	An object as data.table, with variables: variableY - the name of variable of interest, variableZ - the name of optional variable of denominator for the expected ratio estimation, poph - the population size, nh - sample size, nrh - expected number of respondents, estim - total value, var - the expected variance, se - the expected standart error, cv - the expected coefficient of variance.

See Also

[expvar](#), [optsize](#)

Examples

```
library("data.table")
```

```

data <- data.table(H = 1:3, Yh = 10 * 1:3,
                  Yh1 = 10 * 4:6, s2h = 10 * runif(3),
                  s2h2 = 10 * runif(3), nh = rep(4 * 1:3),
                  poph = 8 * 1:3, Rh = rep(1, 3),
                  deffh = rep(2, 3), deffh2 = rep(3, 3))

vars <- expvar(Yh = c("Yh", "Yh1"), H = "H",
              s2h = c("s2h", "s2h2"),
              nh = "nh", poph = "poph",
              Rh = "Rh", deffh = c("deffh", "deffh2"),
              dataset = data)

vars

```

min_count

Minimal count of respondents for the given relative margin of error

Description

The function computes minimal proportion for the given relative margin of error. The calculation takes into sample size, population size, margin of error, expected response rate and design effect.

Usage

```
min_count(n, pop, RMoE, confidence = 0.95, R = 1, deff_sam = 1, deff_est = 1)
```

Arguments

n	The expected sample size.
pop	Population size.
RMoE	The expected relative margin of error.
confidence	Optional positive value for confidence interval. This variable by default is 0.95.
R	The expected response rate (optional). If not defined, it is assumed to be 1 (full-response).
deff_sam	The expected design effect of sample design for the estimates (optional). If not defined, it is assumed to be 1.
deff_est	The estimated design effect of estimator for the estimates (optional). If not defined, it is assumed to be 1.

Value

The estimate of minimal count of respondents for the given relative margin of error.

See Also

[expvar](#), [optsize](#), [MoE_P](#)

Examples

```

min_count(n = 15e3, pop = 2e6, RMoE = 0.1)

## Not run:
library("data.table")
min_count(n = c(10e3, 15e3, 20e3), pop = 2e6, 0.1)

n <- seq(10e3, 30e3, length.out = 11)
# n <- sort(c(n, 22691))
n

RMoE <- seq(.02, .2, length.out = 10)
RMoE

dt <- data.table(n = rep(n, each = length(RMoE)), RMoE = RMoE)
dt[, Y := min_count(n = n, pop = 2.1e6, RMoE = RMoE, R = 1) / 1e3]
dt

## End(Not run)

```

min_prop

Minimal proportion for the given relative margin of error

Description

The function computes minimal proportion for the given relative margin of error. The calculation takes into sample size, population size, margin of error, expected response rate and design effect.

Usage

```
min_prop(n, pop, RMoE, confidence = 0.95, R = 1, deff_sam = 1, deff_est = 1)
```

Arguments

n	The expected sample size.
pop	Population size.
RMoE	The expected relative margin of error.
confidence	Optional positive value for confidence interval. This variable by default is 0.95.
R	The expected response rate (optional). If not defined, it is assumed to be 1 (full-response).
deff_sam	The expected design effect of sample design for the estimates (optional). If not defined, it is assumed to be 1.
deff_est	The estimated design effect of estimator for the estimates (optional). If not defined, it is assumed to be 1.

Value

The estimate of minimal proportion for the given relative margin of error.

See Also

[expvar](#), [optsize](#), [MoE_P](#)

Examples

```
min_prop(n = 100, pop = 1000, RMoE = 0.1)
```

MoE_P

Margin of error for proportion

Description

The function computes margin of error for proportion. The calculation takes into proportion, expected response rate and design effect.

Usage

```
MoE_P(P = 0.5, n, pop, confidence = 0.95, R = 1, deff_sam = 1, deff_est = 1)
```

Arguments

P	The expected proportion for variable of interest.
n	The expected sample size.
pop	Population size.
R	The expected response rate (optional). If not defined, it is assumed to be 1 (full-response).
deff_sam	The expected design effect of sample design for the estimates (optional). If not defined, it is assumed to be 1.
deff_est	The estimated design effect of estimator for the estimates (optional). If not defined, it is assumed to be 1.
confidence	Optional positive value for confidence interval. This variable by default is 0.95.

Value

The estimate of margin of error for proportion.

See Also

[expvar](#), [optsize](#), [MoE_Y](#)

Examples

```

library("data.table")
n <- 100
pop <- 1000

MoE_P(P = 0.5, n = n, pop = pop)

DT <- data.table(P = seq(0, 1, 0.01))
DT[, Y := round(pop * P)]
DT[, AMoE := MoE_P(P, n = 100, pop = 1000)]
DT[Y > 0, RMoE := AMoE / Y]
DT

```

MoE_Y	<i>Margin of error for count</i>
-------	----------------------------------

Description

The function computes margin of error for count. The calculation takes into proportion, expected response rate and design effect.

Usage

```
MoE_Y(P = 0.5, n, pop, confidence = 0.95, R = 1, deff_sam = 1, deff_est = 1)
```

Arguments

P	The expected proportion for variable of interest.
n	The expected sample size.
pop	Population size.
confidence	Optional positive value for confidence interval. This variable by default is 0.95.
R	The expected response rate (optional). If not defined, it is assumed to be 1 (full-response).
deff_sam	The expected design effect of sample design for the estimates (optional). If not defined, it is assumed to be 1.
deff_est	The estimated design effect of estimator for the estimates (optional). If not defined, it is assumed to be 1.

Value

The estimate of margin of error for count.

See Also

[expvar](#), [optsize](#), [MoE_P](#)

Examples

```

library("data.table")
n <- 100
pop <- 1000

MoE_Y(P = 0.5, n = n, pop = pop)

DT <- data.table(P = seq(0, 1, 0.01))
DT[, Y := round(pop * P)]
DT[, AMoE := MoE_Y(P, n = 100, pop = 1000)]
DT[Y > 0, RMoE := AMoE / Y]
DT

```

optsize

Optimal sample size allocation

Description

The function computes optimal sample size allocation over strata.

Usage

```

optsize(
  H,
  n,
  poph,
  s2h = NULL,
  Rh = NULL,
  deffh = NULL,
  fullsampleh = NULL,
  dataset = NULL
)

```

Arguments

H	The stratum variable. One dimensional object convertible to one-column <code>data.table</code> , variable name as character, or column number.
n	Total sample size. One dimensional object with length one.
poph	Population size in each stratum. One dimensional object convertible to one-column <code>data.table</code> , variable name as character, or column number.
s2h	The expected population variance S^2 for variables of interest in each stratum (optional). If not defined, it is assumed to be 1 in each stratum. Object convertible to <code>data.table</code> , variable name as character vector, or column numbers.
Rh	The expected response rate in each stratum (optional). If not defined, it is assumed to be 1 in each stratum (full-response). Object convertible to one-column <code>data.table</code> , variable name as character, or column number.

deffh	The expected design effect for the estimate of variable (optional). If not defined, it is assumed to be 1 for each variable in each stratum. If is defined, then variables is defined the same arrangement as Yh. Object convertible to <code>data.table</code> , variable name as character vector, or column numbers.
fullsampleh	Variable for detection fully surveyed stratum (optinal). If not defined, it is assumed to be 1 in each stratum (full-response). Object convertible to one-column <code>data.table</code> , variable name as character, or column number.
dataset	Optional survey data object convertible to <code>data.table</code> with one row for each stratum.

Value

An object as `data.table`, with variables:

H - stratum,
 variable - the name of variable for population variance S^2 ,
 s2h - population variance S^2 ,
 Rh - the expected response rate,
 deffh - the expected design effect,
 poph - population size,
 deffh - design effect,
 fullsampleh - full sample indicator,
 nh - sample size.

Details

If s2h and Rh is not defined, the sample allocation will be calculated as proportional allocation (proportional to the population size). If Rh is not defined, the sample allocation will be calculated as Neyman allocation.

See Also

[expsize](#), [dom_optimal_allocation](#)

Examples

```
library("data.table")
data <- data.table(H = 1 : 3,
                  s2h=10 * runif(3),
                  s2h2 = 10 * runif(3),
                  poph = 8 * 1 : 3,
                  Rh = rep(1, 3),
                  dd = c(1, 1, 1))

vars <- optsize(H = "H",
               s2h = c("s2h", "s2h2"),
               n = 10, poph = "poph",
               Rh = "Rh",
               fullsampleh = NULL,
               dataset = data)

vars
```

 prop_dom_optimal_allocation

Optimal sample size allocation for proportion

Description

The function computes optimal sample size allocation over strata and domain for proportion.

Usage

```
prop_dom_optimal_allocation(
  H,
  Dom,
  pop = NULL,
  R = NULL,
  deff = NULL,
  se_max = 0.5,
  prop = 0.5,
  min_size = 3,
  step = 1,
  unit_level = TRUE,
  dataset = NULL
)
```

Arguments

H The stratum variable. One dimensional object convertible to one-column data . table or variable name as character, column number.

Dom Variables

used to define population domains. An object convertible to data . table or variable names as character vector, column numbers.

pop The

population size in each stratum.

R The

expected response rate in each stratum (optional). If not defined, it is assumed to be 1 in each stratum (full-response). Object convertible to one-column data . table, variable name as character, or column number.

deff The

expected design effect for the estimate of variable (optional). If not defined, it is assumed to be 1 for each variable in each stratum. If is defined, then variables is defined the same arrangement as Yh. Object convertible to data.table, variable name as character vector, or column numbers.

se_max Variable

for maximum standarterror (se) in domain.

prop The

excepted ratio proportion.

min_size A

numeric value for minimal sample size.

step A

value for pace.

unit_level A

logical value, if dataset is prepared for unit level then value TRUE, othercase FALSE.

dataset Optional

aggregated survey data object convertible to data.table with one row for each stratum.

Value

A list with two data objects:

datah	An object as data.table, with variables: H - the unit stratum variable, Dom - variables used to define population domains, poph - the population size in each stratum, Rh - the expected response rate in each stratum, deffh - the expected design effect, s2h - variance in domain of stratum, sup_cv - Variable for maximum coeficient of variation, poph - population size, nh - sample size .
aggr_Dom	An object as data.table, with variables: Dom - optional variables used to define population domains, pop_Dom - population size, sample_size_Dom - optional variables used to define population domains, sample_size - optional variables used to define population domains, pop - sample size

See Also

[expsize](#), [optsize](#), [dom_optimal_allocation](#)

Examples

```
library("data.table")
library("laeken")
data("eusilc")
eusilc <- data.table(eusilc)
dataset <- eusilc[, .(poph = sum(db090)), by = c("db040")]
dataset[, dom := "1"]
res <- prop_dom_optimal_allocation(H = "db040", Dom = "dom",
                                  pop = "poph", R = NULL,
                                  deff = NULL, se_max = 0.5,
                                  prop = 0.5, min_size = 3,
                                  step = 1, unit_level = FALSE,
                                  dataset = dataset)
```

round2

Rounding numbers

Description

The function rounds the values in its first argument to the specified number of decimal places (default 0).

Usage

```
round2(x, n)
```

Arguments

x a numeric vector.
n integer indicating the number of decimal places.

Value

Rounded value

See Also

[expsize](#), [dom_optimal_allocation](#)

Examples

```
dar <- 100 * runif(3)
dar
round2(dar, 1)
```

s2

Population variance

Description

The function to estimate population variance S^2 .

Usage

```
s2(y, w = NULL)
```

Arguments

y	Study variable.
w	Survey weight (optional). If not defined, it is assumed to be 1 for each element.

Value

Population variance S^2 or the estimate of population variance s^2 .

Details

If w is not defined, the result is equal to the result of the function var.

Examples

```
s2(1:10)
s2(1:10, rep(1:2, each = 5))
all.equal(s2(1:10), var(1:10))
```

Index

- * **package**
 - surveyplanning-package, 2
- * **surveysampling**
 - dom_optimal_allocation, 3
 - expsize, 6
 - expvar, 7
 - min_count, 10
 - min_prop, 11
 - MoE_P, 12
 - MoE_Y, 13
 - optsize, 14
 - prop_dom_optimal_allocation, 16
 - round2, 18

- dom_optimal_allocation, 3, 15, 17, 18

- expsize, 5, 6, 15, 17, 18
- expvar, 7, 7, 9, 10, 12, 13

- min_count, 10
- min_prop, 11
- MoE_P, 7, 10, 12, 12, 13
- MoE_Y, 12, 13

- optsize, 5, 7, 9, 10, 12, 13, 14, 17

- prop_dom_optimal_allocation, 5, 16

- round2, 18

- s2, 19
- surveyplanning
 - (surveyplanning-package), 2
- surveyplanning-package, 2