

# Package ‘GaussSuppression’

September 24, 2021

**Type** Package

**Title** Tabular Data Suppression using Gaussian Elimination

**Version** 0.1.0

**Date** 2021-09-23

**Maintainer** Øyvind Langsrud <oyl@ssb.no>

**Depends** Matrix

**Imports** SSBtools, RegSDC, stats, methods

## Description

A statistical disclosure control tool to protect tables by suppression using the Gaussian elimination secondary suppression algorithm. Primary suppression functions for the minimum frequency rule, the dominance rule and a directly-disclosive rule are included. General primary suppression functions can be supplied as input. Suppressed frequencies can be replaced by synthetic decimal numbers as described in Langsrud (2019) <[doi:10.1007/s11222-018-9848-9](https://doi.org/10.1007/s11222-018-9848-9)>.

**License** Apache License 2.0

**Encoding** UTF-8

**RoxygenNote** 7.1.2

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Øyvind Langsrud [aut, cre],  
Daniel Lupp [aut]

**Repository** CRAN

**Date/Publication** 2021-09-24 07:50:02 UTC

## R topics documented:

DominanceRule . . . . .	2
GaussSuppressDec . . . . .	3
GaussSuppressionFromData . . . . .	4
MaxContribution . . . . .	7

Ncontributors . . . . .	8
NcontributorsHolding . . . . .	9
SuppressDirectDisclosure . . . . .	10
SuppressionFromDecimals . . . . .	11

<b>Index</b>	<b>13</b>
--------------	-----------

---

DominanceRule	<i>Dominance (n,k) rule for magnitude tables</i>
---------------	--

---

### Description

Supports application of multiple values for n and k. The function works on magnitude tables containing negative cell values by calculating contribution based on absolute values.

### Usage

```
DominanceRule(data, x, crossTable, numVar, n, k, ...)
```

### Arguments

data	the dataset
x	ModelMatrix generated by parent function
crossTable	crossTable generated by parent function
numVar	vector containing numeric values in the data set
n	parameter n in dominance rule.
k	parameter k in dominance rule.
...	unused parameters

### Details

Currently the implementation only supports a single numeric variable.

### Value

logical vector that is TRUE in positions corresponding to cells breaching the dominance rules.

### Author(s)

Daniel Lupp

---

GaussSuppressDec	<i>Cell suppression with synthetic decimal numbers</i>
------------------	--

---

### Description

[GaussSuppressionFromData](#) is run and decimal numbers are added to output by a modified (for sparse matrix efficiency) version of [SuppressDec](#).

### Usage

```
GaussSuppressDec(
  data,
  ...,
  output = NULL,
  digits = 9,
  nRep = NULL,
  rmse = pi/3,
  sparseLimit = 500,
  rndSeed = 123,
  runIpf = FALSE,
  eps = 0.01,
  iter = 100,
  mismatchWarning = TRUE,
  whenDuplicatedInner = NULL,
  whenMixedDuplicatedInner = warning
)
```

### Arguments

data	Input daata as a data frame
...	Further parameters to <a href="#">GaussSuppressionFromData</a>
output	NULL (default), "publish", "inner", "publish_inner", or "publish_inner_x" (x also).
digits	Parameter to <a href="#">RoundWhole</a> . Values close to whole numbers will be rounded.
nRep	NULL or an integer. When >1, several decimal numbers will be generated.
rmse	Desired root mean square error of decimal numbers. Variability around the expected, according to the linear model, inner frequencies. The expected frequencies are calculated from the non-suppressed publishable frequencies.
sparseLimit	Limit for the number of rows of a reduced x-matrix within the algorithm. When exceeded, a new sparse algorithm is used.
rndSeed	If non-NULL, a random generator seed to be used locally within the function without affecting the random value stream in R.
runIpf	When TRUE, additional frequencies are generated by iterative proportional fitting using <a href="#">Mipf</a> .

eps	Parameter to <a href="#">Mipf</a> .
iter	Parameter to <a href="#">Mipf</a> .
mismatchWarning	Whether to produce the warning "Mismatch between whole numbers and suppression", when relevant. When nRep>1, all replicates must satisfy the whole number requirement for non-suppressed cells. When mismatchWarning is integer (>0), this will be used as parameter digits to <a href="#">RoundWhole</a> when doing mismatch checking (can be quite low when nRep>1).
whenDuplicatedInner	Function to be called when default output and when cells marked as inner correspond to several input cells (aggregated) since they correspond to published cells.
whenMixedDuplicatedInner	Function to be called in the case above when some inner cells correspond to published cells (aggregated) and some not (not aggregated).

**Value**

A data frame where inner cells and cells to be published are combined or output according to parameter output.

**Author(s)**

Øyvind Langrød

**Examples**

```
z1 <- SSBtoolsData("z1")
GaussSuppressDec(z1, 1:2, 3)
GaussSuppressDec(z1, freqVar = "ant", formula = ~ region + hovedint, maxN = 10)
```

---

GaussSuppressionFromData

*Cell suppression from input data containing inner cells*

---

**Description**

Aggregates are generated followed by primary suppression followed by secondary suppression by Gaussian elimination by [GaussSuppression](#)

**Usage**

```
GaussSuppressionFromData(
  data,
  dimVar = NULL,
  freqVar = NULL,
  numVar = NULL,
```

```

weightVar = NULL,
charVar = NULL,
hierarchies = NULL,
formula = NULL,
maxN = 3,
protectZeros = TRUE,
secondaryZeros = FALSE,
candidates = CandidatesDefault,
primary = PrimaryDefault,
forced = NULL,
hidden = NULL,
singleton = SingletonDefault,
singletonMethod = ifelse(secondaryZeros, "anySumNOTprimary", "anySum"),
printInc = TRUE,
output = "publish",
x = NULL,
crossTable = NULL,
preAggregate = is.null(freqVar),
extraAggregate = preAggregate & !is.null(charVar),
...
)

```

### Arguments

data	Input data as a data frame
dimVar	The main dimensional variables and additional aggregating variables. This parameter can be useful when hierarchies and formula are unspecified.
freqVar	A single variable holding counts (name or number).
numVar	Other numerical variables to be aggregated
weightVar	weightVar Weights (costs) to be used to order candidates for secondary suppression
charVar	Other variables possibly to be used within the supplied functions
hierarchies	List of hierarchies, which can be converted by <a href="#">AutoHierarchies</a> . Thus, the variables can also be coded by "rowFactor" or "", which correspond to using the categories in the data.
formula	A model formula
maxN	Suppression parameter. Default: Cells having counts <= maxN are set as primary suppressed.
protectZeros	Suppression parameter. Default when TRUE: Empty cells (count=0) are set as primary suppressed.
secondaryZeros	Suppression parameter.
candidates	GaussSuppression input or a function generating it (see details) Default: <a href="#">CandidatesDefault</a>
primary	GaussSuppression input or a function generating it (see details) Default: <a href="#">PrimaryDefault</a>
forced	GaussSuppression input or a function generating it (see details)
hidden	GaussSuppression input or a function generating it (see details)

singleton	GaussSuppression input or a function generating it (see details) Default: <a href="#">SingletonDefault</a>
singletonMethod	<a href="#">GaussSuppression</a> input
printInc	<a href="#">GaussSuppression</a> input
output	One of "publish" (default), "inner", "publish_inner", "publish_inner_x", "publish_x", "inner_x", and "input2functions" (input to supplied functions). Here "inner" means input data (possibly pre-aggregated) and "x" means dummy matrix (as input parameter x).
x	x ( <a href="#">modelMatrix</a> ) and <a href="#">crossTable</a> can be supplied as input instead of generating it from <a href="#">ModelMatrix</a>
crossTable	See above.
preAggregate	When TRUE, the data will be aggregated within the function to an appropriate level. This is defined by the dimensional variables according to <a href="#">dimVar</a> , hierarchies or formula and in addition <a href="#">charVar</a> .
extraAggregate	When TRUE, the data will be aggregated by the dimensional variables according to <a href="#">dimVar</a> , hierarchies or formula. The aggregated data and the corresponding x-matrix will only be used as input to the singleton function and <a href="#">GaussSuppression</a> . This extra aggregation is useful when parameter <a href="#">charVar</a> is used. Supply "publish_inner", "publish_inner_x", "publish_x" or "inner_x" as output to obtain extra aggregated results. Supply "inner" or "input2functions" to obtain other results.
...	Further arguments to be passed to the supplied functions.

### Details

The supplied functions for generating [GaussSuppression](#) input takes the following arguments: [crossTable](#), x, [freq](#), [num](#), [weight](#), [maxN](#), [protectZeros](#), [secondaryZeros](#), [data](#), [freqVar](#), [numVar](#), [weightVar](#), [charVar](#), [dimVar](#) and ... where the two first are [ModelMatrix](#) outputs ([modelMatrix](#) renamed to x). The vector, [freq](#), is aggregated counts ( $t(x) \% \% data[[freqVar]]$ ). Similarly, [num](#), is a data frame of aggregated numerical variables. It is possible to supply several primary functions joined by c, e.g. (c(FunPrim1,FunPrim2)). All NAs returned from any of the functions force the corresponding cells not to be primary suppressed.

### Value

Aggregated data with suppression information

### Author(s)

Øyvind Langsrud and Daniel Lupp

### Examples

```
z1 <- SSBtoolsData("z1")
GaussSuppressionFromData(z1, 1:2, 3)
```

```

z2 <- SSBtoolsData("z2")
GaussSuppressionFromData(z2, 1:4, 5, protectZeros = FALSE)

# Data as in GaussSuppression examples
df <- data.frame(values = c(1, 1, 1, 5, 5, 9, 9, 9, 9, 9, 0, 0, 0, 7, 7),
                 var1 = rep(1:3, each = 5), var2 = c("A", "B", "C", "D", "E"))

GaussSuppressionFromData(df, c("var1", "var2"), "values")
GaussSuppressionFromData(df, c("var1", "var2"), "values", formula = ~var1 + var2, maxN = 10)
GaussSuppressionFromData(df, c("var1", "var2"), "values", formula = ~var1 + var2, maxN = 10,
                          primary = function(freq, crossTable, maxN, ...)
                                which(freq <= maxN & crossTable[[2]] != "A" & crossTable[, 2] != "C"))

# Combining several primary functions
# Note that NA & c(TRUE, FALSE) equals c(NA, FALSE)
GaussSuppressionFromData(df, c("var1", "var2"), "values", formula = ~var1 + var2, maxN = 10,
                          primary = c(function(freq, maxN, ...) freq >= 45,
                                       function(freq, maxN, ...) freq <= maxN,
                                       function(crossTable, ...) NA & crossTable[[2]] == "C",
                                       function(crossTable, ...) NA & crossTable[[1]] == "Total"
                                       & crossTable[[2]] == "Total"))

# Similar to GaussSuppression examples
GaussSuppressionFromData(df, c("var1", "var2"), "values", formula = ~var1 * var2,
                          candidates = NULL, singleton = NULL, protectZeros = FALSE, secondaryZeros = TRUE)
GaussSuppressionFromData(df, c("var1", "var2"), "values", formula = ~var1 * var2,
                          singleton = NULL, protectZeros = FALSE, secondaryZeros = FALSE)
GaussSuppressionFromData(df, c("var1", "var2"), "values", formula = ~var1 * var2,
                          protectZeros = FALSE, secondaryZeros = FALSE)

# Examples with zeros as singletons
z <- data.frame(row = rep(1:3, each = 3), col = 1:3, freq = c(0, 2, 5, 0, 0, 6:9))
GaussSuppressionFromData(z, 1:2, 3, singleton = NULL)
GaussSuppressionFromData(z, 1:2, 3, singletonMethod = "none") # as above
GaussSuppressionFromData(z, 1:2, 3)
GaussSuppressionFromData(z, 1:2, 3, protectZeros = FALSE, secondaryZeros = TRUE, singleton = NULL)
GaussSuppressionFromData(z, 1:2, 3, protectZeros = FALSE, secondaryZeros = TRUE)

```

---

MaxContribution

*Find major contributors to aggregates*


---

## Description

Assuming aggregates are calculated via a dummy matrix by  $z = t(x) \%*\% y$ , the  $n$  largest contributors are found (value or index) for each aggregate.

## Usage

```
MaxContribution(x, y, n = 1, decreasing = TRUE, index = FALSE)
```

**Arguments**

x	A (sparse) dummy matrix
y	Vector of input values (contributors)
n	Number of contributors to be found
decreasing	Ordering parameter. Smallest contributors found when FALSE.
index	Indices to y returned when TRUE

**Value**

Matrix with largest contributors in first column, second largest in second column and so on.

**Author(s)**

Øyvind Langsrud

**See Also**

[ModelMatrix](#)

**Examples**

```
library(SSBtools)

z <- SSBtoolsData("sprt_emp_withEU")
z$age[z$age == "Y15-29"] <- "young"
z$age[z$age == "Y30-64"] <- "old"

a <- ModelMatrix(z, formula = ~age + geo, crossTable = TRUE)

cbind(as.data.frame(a$crossTable), MaxContribution(a$modelMatrix, z$ths_per, 1))
cbind(a$crossTable, MaxContribution(a$modelMatrix, z$ths_per, 10))
cbind(a$crossTable, MaxContribution(a$modelMatrix, z$ths_per, 10, index = TRUE))

b <- ModelMatrix(z[, -4], crossTable = TRUE, inputInOut = c(TRUE, FALSE, TRUE))

cbind(b$crossTable, MaxContribution(b$modelMatrix, z$ths_per, 10))
```

---

Ncontributors

*Find the number of unique groups contributing to aggregates*

---

**Description**

Assuming aggregates are calculated via a dummy matrix by  $z = t(x) \%*\% y$ , the the number of unique contributing groups, according to a grouping variable, are found for each aggregate. The missing group category is not counted.

**Usage**

```
Ncontributors(x, groups)
```

**Arguments**

x	A (sparse) dummy matrix
groups	Vector of group categories

**Value**

Vector of numbers of unique groups

**Author(s)**

Øyvind Langsrud

**See Also**

[ModelMatrix](#)

**Examples**

```
library(SSBtools)

z <- SSBtoolsData("sprt_emp_withEU")
z$age[z$age == "Y15-29"] <- "young"
z$age[z$age == "Y30-64"] <- "old"
z$groups <- c("A", "A", "B", "A", "B", "C")

a <- ModelMatrix(z, formula = ~age*eu + geo + year, crossTable = TRUE)

cbind(as.data.frame(a$crossTable), nGroups = Ncontributors(a$modelMatrix, z$groups))
cbind(as.data.frame(a$crossTable), nYears = Ncontributors(a$modelMatrix, z$year))
cbind(as.data.frame(a$crossTable), nUnique_ths_per = Ncontributors(a$modelMatrix, z$ths_per))
```

---

NcontributorsHolding [Ncontributors](#) with *holding-indicator*

---

**Description**

The aggregates (columns of x) are grouped by a holding indicator. Within each holding group, the number of unique groups (output) is set to be equal.

**Usage**

```
NcontributorsHolding(x, groups, holdingInd = NULL)
```

**Arguments**

x	A (sparse) dummy matrix
groups	Vector of group categories
holdingInd	Vector of holding group categories

**Details**

A representative within the holding group is used to calculate output by [Ncontributors](#). The one with maximal column sum of x is chosen as the representative. Normally this will be an aggregate representing the holding group total. When holdingInd is NULL (default), the function is equivalent to [Ncontributors](#).

**Value**

Vector of numbers of unique groups

**Author(s)**

Øyvind Langsrud

---

SuppressDirectDisclosure

*Suppression of directly-disclosive cells*

---

**Description**

Function for suppressing directly-disclosive cells in frequency tables. The method detects and primary suppresses directly-disclosive cells with the [FindDisclosiveCells](#) function, and applies a secondary suppression using Gauss suppression (see [GaussSuppressionFromData](#)).

**Usage**

```
SuppressDirectDisclosure(  
  data,  
  dimVar,  
  freqVar,  
  coalition = 1,  
  secondaryZeros = coalition,  
  candidates = DirectDisclosureCandidates,  
  ...  
)
```

**Arguments**

data	the input data
dimVar	main dimensional variables for the output table
freqVar	variable containing frequency counts
coalition	numeric variable, parameter for primary suppression. Default value is 1.
secondaryZeros	logical or numeric value for secondary suppression. If logical, it is converted to resp numeric value (0 or 1). If numeric, it describes the largest number that is prioritized over zeroes in secondary suppression. Default value is equal to coalition.
candidates	function parameter for gauss suppression.
...	optional parameters that can be passed to the primary suppression method. See <a href="#">FindDisclosiveCells</a> for details.

**Details**

Currently, the method has no support for hierarchical data.

**Value**

data.frame containing the result of the suppression

**Author(s)**

Daniel Lupp

**Examples**

```

tex <- data.frame(v1 = rep(c('a', 'b', 'c'), times = 4),
                 v2 = c('i','i', 'i','h','h','h','i','i','i','h','h','h'),
                 v3 = c('y', 'y', 'y', 'y', 'y', 'y','z','z', 'z', 'z', 'z', 'z'),
                 freq = c(0,0,5,0,2,3,1,0,3,1,1,2))
SuppressDirectDisclosure(tex, c("v1", "v2", "v3"), "freq")
SuppressDirectDisclosure(tex, c("v1", "v2", "v3"), "freq", coalition = 2, unknown.threshold = 10)

```

---

SuppressionFromDecimals

*Cell suppression from synthetic decimal numbers*

---

**Description**

Decimal numbers, as calculated by [GaussSuppressDec](#), are used to decide suppression (whole numbers or not). Technically, the calculations are done via [GaussSuppressionFromData](#), but without running [GaussSuppression](#). All suppressed cells are primary suppressed.

**Usage**

```
SuppressionFromDecimals(
  data,
  decVar,
  freqVar = NULL,
  numVar = NULL,
  preAggregate = FALSE,
  digits = 9,
  ...
)
```

**Arguments**

data	Input data as a data frame
decVar	One ore several (nRep>1) decimal number variables.
freqVar	A single variable holding counts (not needed)
numVar	Other numerical variables to be aggregated
preAggregate	Parameter to <a href="#">GaussSuppressionFromData</a>
digits	Parameter to <a href="#">RoundWhole</a> . Values close to whole numbers will be rounded.
...	Other parameters to <a href="#">GaussSuppressionFromData</a>

**Details**

Several decimal number variables reduce the probability of obtaining whole numbers by chance.

**Value**

Aggregated data with suppression information

**Author(s)**

Øyvind Langsrud

**Examples**

```
z2 <- SSBtoolsData("z2")

# Find suppression and decimal numbers with "fylke" in model
a <- GaussSuppressDec(z2, dimVar = c("region", "fylke", "hovedint"),
  freqVar = "ant", protectZeros = FALSE, maxN = 2,
  output = "inner")

# Add decimal numbers to data
z2$freqDec <- a$freqDec

# Find suppression with "kostragr" in model
b <- SuppressionFromDecimals(z2, dimVar = c("region", "kostragr", "hovedint"),
  freqVar = "ant", decVar = "freqDec")
```

# Index

AutoHierarchies, [5](#)

CandidatesDefault, [5](#)

DominanceRule, [2](#)

FindDisclosiveCells, [10](#), [11](#)

GaussSuppressDec, [3](#), [11](#)

GaussSuppression, [4](#), [6](#), [11](#)

GaussSuppressionFromData, [3](#), [4](#), [10–12](#)

MaxContribution, [7](#)

Mipf, [3](#), [4](#)

ModelMatrix, [6](#), [8](#), [9](#)

Ncontributors, [8](#), [9](#), [10](#)

NcontributorsHolding, [9](#)

PrimaryDefault, [5](#)

RoundWhole, [3](#), [4](#), [12](#)

SingletonDefault, [6](#)

SuppressDec, [3](#)

SuppressDirectDisclosure, [10](#)

SuppressionFromDecimals, [11](#)