Statistical disclosure control and synthetic data generation using R

Alexander Kowarik,
Head of Statistical Methods and Survey Methodology

Johannes Gussenbauer
Statistical Methods and Survey Methodology

Vienna, 21 November 2023

www.statistik.at

Content

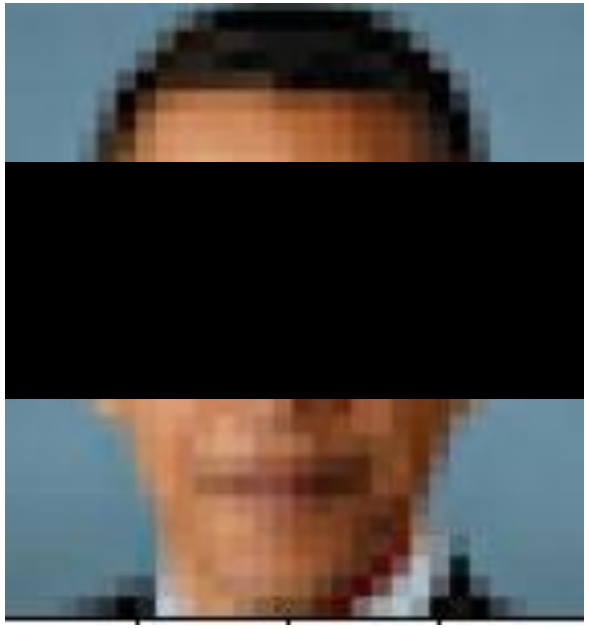
- SDC methods vs. synthetic data
- Micro data protection with sdcMicro
  - sdcApp(): Graphical User Interface
  - Target record swapping in R
- Anonymization of tabular data
- SDC for tabular data

Motivation

Who is this person?
Synthetic data

Sources: https://twitter.com/Chicken3gg/status/1274314622447820801, https://pbs.twimg.com/media/EbBrAKNX0A0NSO8?format=jpg&name=medium

Who is this person?
SDC applied



Sources: https://twitter.com/Chicken3gg/status/1274314622447820801 ; https://en.wikipedia.org/wiki/Social_policy_of_the_Barack_Obama_administration#/media/File:Official_portrait_of_Barack_Obama.jpg; https://pbs.twimg.com/media/EbBrAKNX0A0NSO8?format=jpg&name=medium

SDC Methods vs. Synthetic Data

Which approach is better?

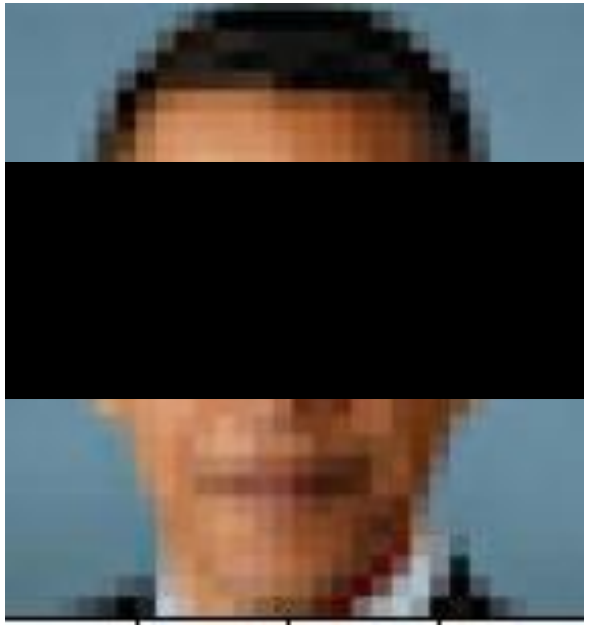
Original data / SDC applied / Synthetic data

Sources: https://twitter.com/Chicken3gg/status/1274314622447820801 ; https://en.wikipedia.org/wiki/Social_policy_of_the_Barack_Obama_administration#/media/File:Official_portrait_of_Barack_Obama.jpg

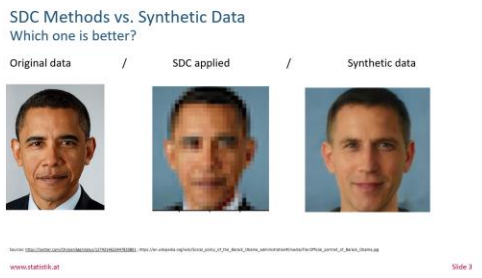
SDC Methods vs. Synthetic Data

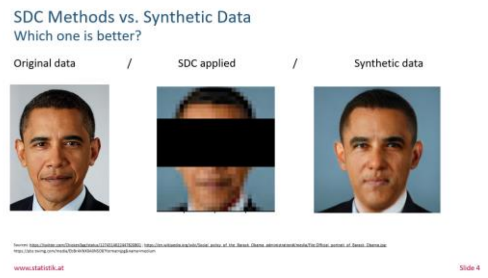
Which one is better?

Original data / SDC applied / Synthetic data



Sources: https://twitter.com/Chicken3gg/status/1274314622447820801 ; https://en.wikipedia.org/wiki/Social_policy_of_the_Barack_Obama_administration#/media/File:Official_portrait_of_Barack_Obama.jpg; https://pbs.twimg.com/media/EbBrAKNX0A0NSO8?format=jpg&name=medium


SDC Methods vs. Synthetic Data
Which one is better?
Original data / SDC applied / Synthetic data
www.statistik.at
Slide 3


SDC Methods vs. Synthetic Data
Which one is better?
Original data / SDC applied / Synthetic data
www.statistik.at
Slide 4

SDC methods vs. Synthetic data

Both: Reducing risk / Maximizing utility

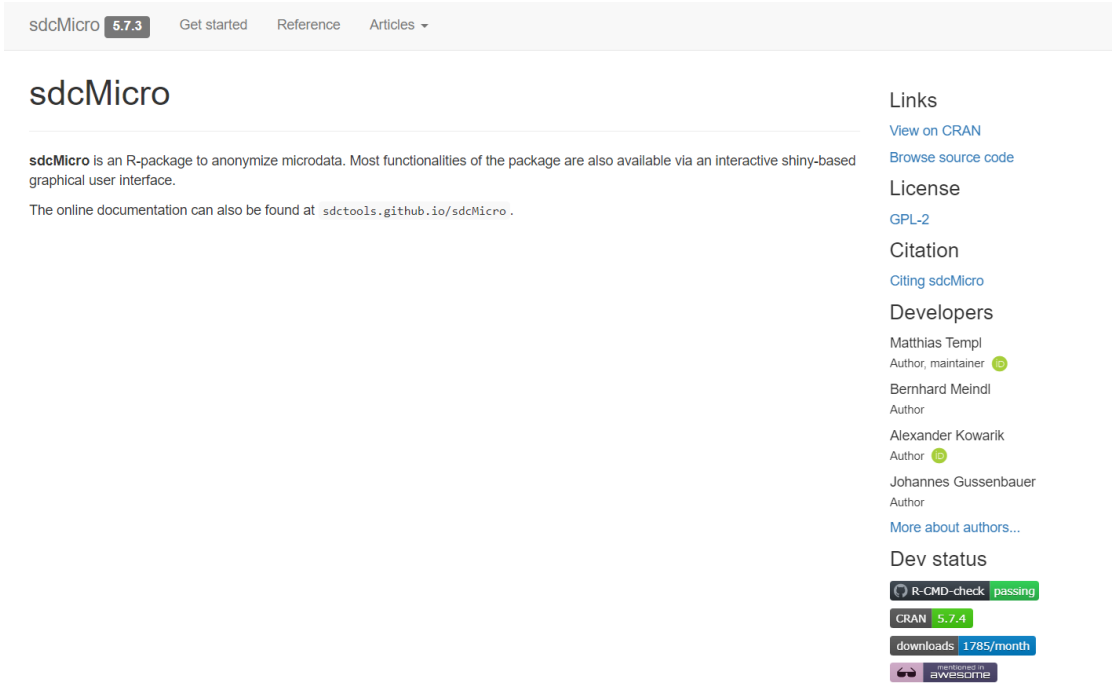
Both	/	SDC applied	/	Synthetic data
• More complicated for hierachical data, e.g. households • Re-identification risk != 0		- Loss of granularity - Individual/manual approach		- High granularity - Complex generation process - Re-identification risk harder to assess (attribute disclosure, membership inference)

Micro data protection

Micro data protection in R
sdcMicro

- Released versions are on CRAN: https://cran.r-project.org/web/packages/sdcMicro
- Development version and issue tracking on Github: https://github.com/sdcTools/sdcMicro
- Collaborative development with colleagues from Statistics Austria and other institutions
- Documentation:

  http://sdctools.github.io/sdcMicro/



sdcMicro
Categorical variables

- Risk estimation:
  - Concept of k-anonymity and
  - various methods to estimate the disclosure risk (individual risk, global risk, suda2)
- Deterministic methods
  - Non-perturbative protection methods
    - Top- and bottom coding
    - Recoding
    - Local suppression
    - Probabilistic methods
  - Perturbative protection methods
    - (Rank) swapping
    - Post-randomization (pram)

sdcMicro
Numerical variables

- Deterministic protection methods
  - Top- and bottom coding
  - Microaggregation
  - Rounding
- Perturbative protection methods based on randomness
  - Noise addition
  - (Rank)swapping
  - Shuffling

sdcMicroObj

- The S4-class sdcMicroObj Slots
  - original Data (@origData)
  - modified (key) variables (@manipKeyVars, @manipNumVars, …)
  - categorical key variables (@keyVars)
  - numerical key variables (@numVars)
  - computed disclosure risk (@risk)
  - previous object (@prev)

sdcMicroObj methods

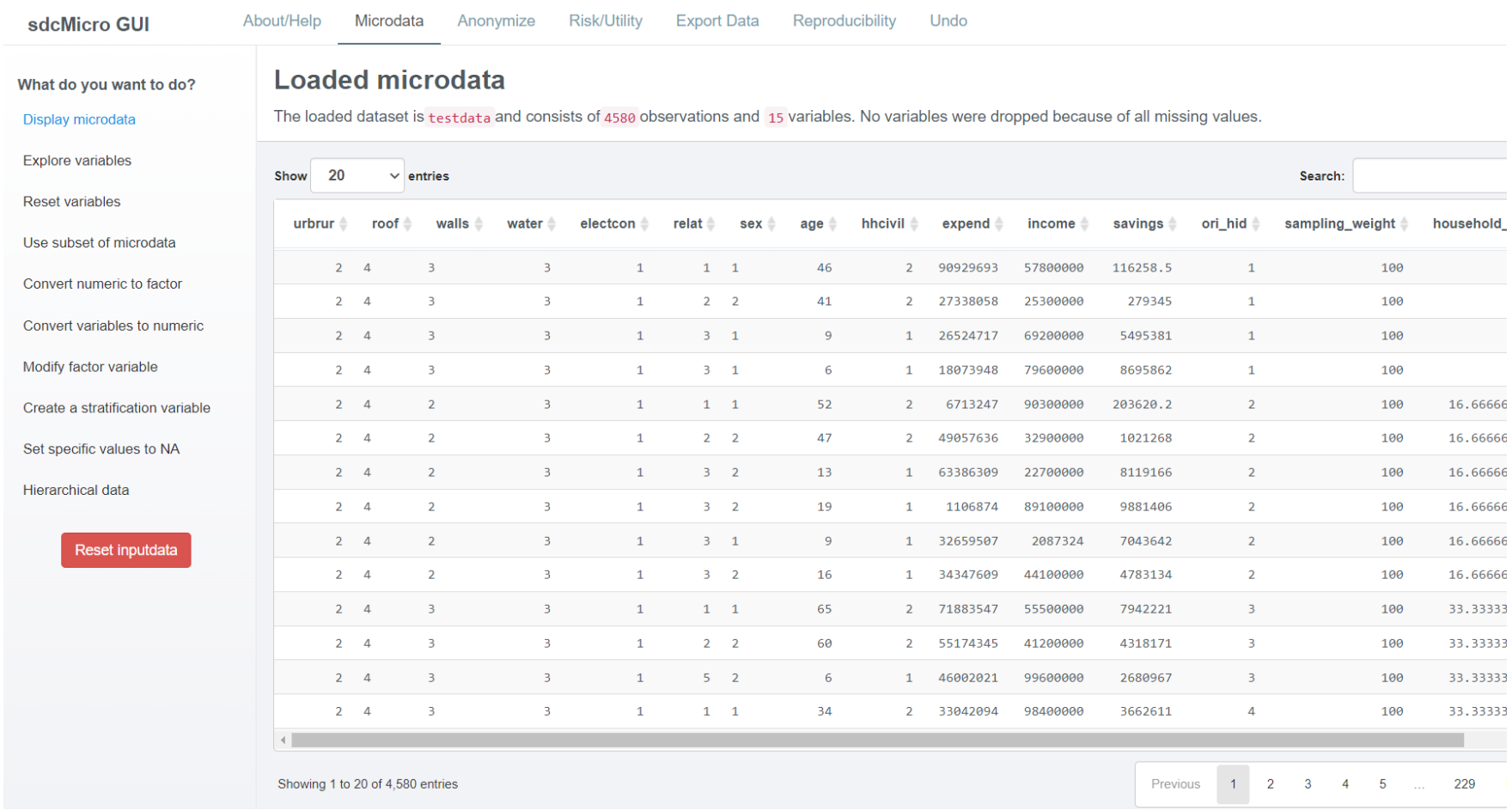
- The S4-class sdcMicroObj (some) Methods:
  - addNoise(): add noise to numerical key variables
  - Pram(): Post RAndomisation Method (PRAM)
  - rankSwap(): Rank Swapping
  - shuffle(): Shuffling and EGADP
  - report(): Create a report about the anonymization process
  - undolast(): undo last calculation
- . . .

sdcMicro
Tiny example

```
data(testdata)
sdc <- createSdcObj(testdata,
  keyVars=c('urbrur','roof','walls','water','electcon','relat‘),
  numVars=c('expend','income','savings'),
  w='sampling_weight')
### Display Risk
sdc@risk$numeric [1] 1
### use addNoise without Parameters
sdc <- addNoise(sdc, variables = c("expend","income"))
### risk changed
sdc@risk$numeric [1] 0.07729258
```

sdcApp
(trs not included)

- Online-Demo https://sdctools.shinyapps.io/sdcapp/



View/Analyze existing sdcProblem

Show summary

Explore variables

Add linked variables

Create new IDs

Anonymize categorical variables

Recoding

k-Anonymity

PRAM (simple)

PRAM (expert)

Supress values with high risks

Anonymize numerical variables

Top/bottom coding

Microaggregation

Adding noise

Rank swapping

Summary of dataset and variable selection

The loaded dataset consists of `4580` records and `15` variables.

Categorical key variable(s): `urbrur` `roof` `walls`

Numerical key variable(s): `expend`

Sampling weight: `sampling_weight`

Hierarchical identifier: `ori_hid`

Computation time

The current computation time was ~ `0.15 seconds` .

Information on categorical key variables

Reported is the number of levels, average frequency of each level and frequency of the smallest level for categorical key variables. In parentheses, the san
the original data. Note that NA (missing) is counted as a separate category.

Variable name	Number of levels	Average frequency	Frequency of smallest level
urbrur	3 (3)	2285.000 (2285.000)	643 (643)
roof	5 (5)	916.000 (916.000)	16 (16)
walls	3 (3)	1526.667 (1526.667)	50 (50)

Risk measures for categorical key variables

We expect `0.16` (`0.00%`) re-identifications in the population, as compared to `0.16` (`0.00%`) re-identifications in the original data.

`0` observations have a higher risk than the risk in the main part of the data, as compared to `0` observations in the original data.

SUDA2 risk measure

The SUDA algorithm is used to search for Minimum Sample Uniques (MSU) in the data among the sample uniques to determine which sample uniques are also special uniques i.e., have subsets that are also unique. See the help files for more information on SUDA scores.

Reset to choose a different sampling fraction parameter

Suda scores (sampling fraction is 0.1)

The table below shows the frequencies of the records with a suda score in the specified intervals.

Interval	Number of records
== 0	4580
(0.0, 0.1]	0
(0.1, 0.2]	0
(0.2, 0.3]	0
(0.3, 0.4]	0
(0.4, 0.5]	0
(0.5, 0.6]	0
(0.6, 0.7]	0
> 0.7	0

Attribute contributions

The table below shows the contribution of each categorical key variable to the SUDA scores. The contribution of a variable is

Variable selection

Variable name	Type	Additional suppressions by local suppression algorithm
urbrur	cat. key variable	0
roof	cat. key variable	0
walls	cat. key variable	0
expend	num. key variable	
sampling_weight	sampling weight	

Additional parameters

Parameter	Value

What do you want to do?

View the current script

Import a previously saved problem

Export/Save the current sdcProblem

View the current generated script

Browse and download the script used to generate your results. These can be used later as a reminder of what you did or entered into R from command-line to reproduce results.

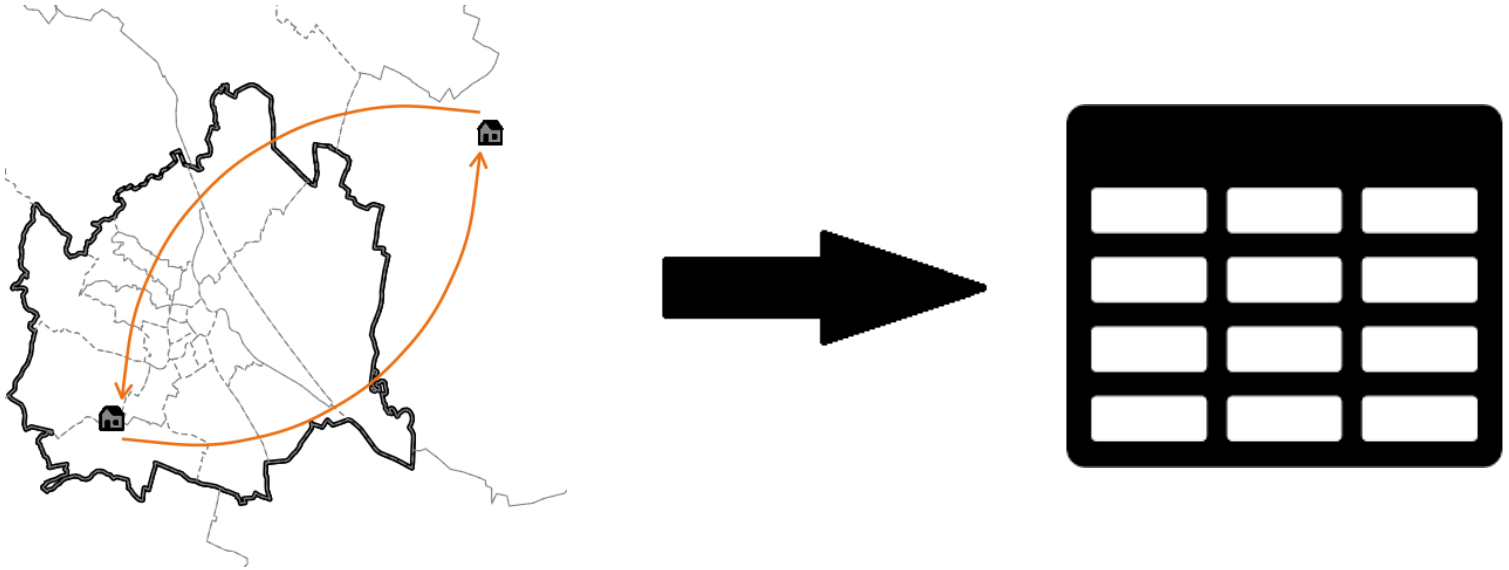
Save Script to File

```
require(sdcMicro)
obj$inputdata <- readMicrodata(path="testdata", type="rdf", convertCharToFac=FALSE, drop_all_missings=FALSE)
inputdataB <- inputdata

## Convert a numeric variable to factor (each distinct value becomes a factor level)
inputdata <- varToFactor(obj=inputdata, var=c("urbrur","water","electcon","relat"))
## Set up sdcMicro object
sdcObj <- createSdcObj(dat=inputdata,
        keyVars=c("urbrur","roof","walls"),
        numVars=c("expend"),
        weightVar=c("sampling_weight"),
        hhId=c("ori_hid"),
        strataVar=NULL,
        pramVars=NULL,
        excludeVars=NULL,
        seed=0,
        randomizeRecords=FALSE,
        alpha=c(1))

## Store name of uploaded file
opts <- get.sdcMicroObj(sdcObj, type="options")
opts$filename <- "testdata"
sdcObj <- set.sdcMicroObj(sdcObj, type="options", input=list(opts))


## calculating suda2 riskmeasure
sdcObj <- suda2(obj=sdcObj, DisFraction=0.1, missing=NA)
```

Target record swapping

Recommended methodology for the census

- Data-swapping technique applied on micro data
- TRS: Swap households across administrative/geographic regions
- Swapped microdata used for all outputs



Target record swapping

- High risk → small frequency counts of individuals on a set of key variables (usually)
  - **Geographic hierarchy x risk variables**
  - k-anonymity in our algorithm
  - (own risk can be provided)
- Household at high risk ↔ individual at high risk

- “Similarity” variables
- Variables on which the swapped households must agree on (hh size, …)
  - Preserves marginal distributions of those variables

Target record swapping
with sdcMicro

- Data preparation
  - Data needs to contain only integer/numeric columns
    - no decimal places
  - Convert column to integer using, e.g. `factor`

```
dat[,AGE.M:=as.integer(factor(AGE.M))]
```

  - Generate additional variables, e.g. for the similariy measure, e.g.
    - A truncate household size

```
dat[,Size:=pmin(5,Size)]
dat[!duplicated(HID),.N,by=.(Size)][order(Size)]
```

Target record swapping
parameters

- geographic hierarchy
  - *hierarchy = **c**("NUTS1","NUTS2")*
- Column name of household id
  - *hid = "HID“*
- Variables for internal risk calculation
  - *risk_variables = **c**("COC.M","POB.M")*
- Threshold for k-anonymity
  - *k_anonymity = 3*
- (Minimal) swap rate
  - swaprate = 0.05
- Similarity profile(s)
  - similar = list(c("Size“), c("Size“, “NationalityHead“))

Target record swapping
Call and output

```
swapped <- recordSwap(data = dat, … ,
return_swapped_id = TRUE,
seed = 123)
```

- return_swapped_id = TRUE get household ID of swapped household
- Seed to fix random seed

- Number of swapped households

```
dat_swapped[HID!=HID_swapped,uniqueN(HID)]
## [1] 520
```

Synthetic data generation

Johannes Gussenbauer

Overview

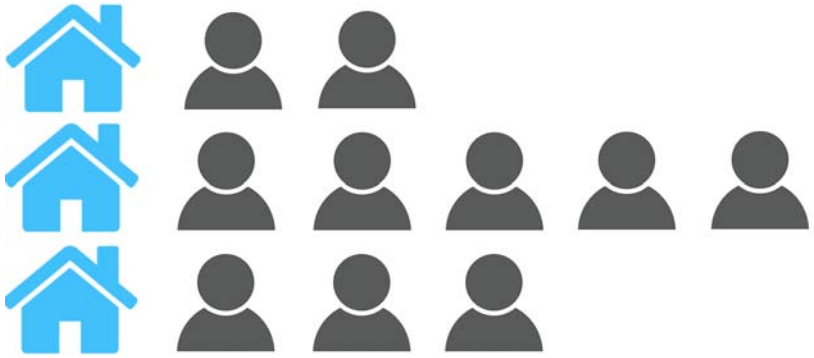
- Quick intro to R-package `simPop`

- Initialise and extend synthetic data

- Calibration of synth. population

simPop

- `simPop` R-Package to generate synthetic micro data ~ household data



- Why generate synthetic micro data?
  - Microdata more and more needed by scientific community / the public
  - Often not possible to disseminate micro data (GDPR, national data protection laws)
  - Available micro data highly censored/takes long time to get access

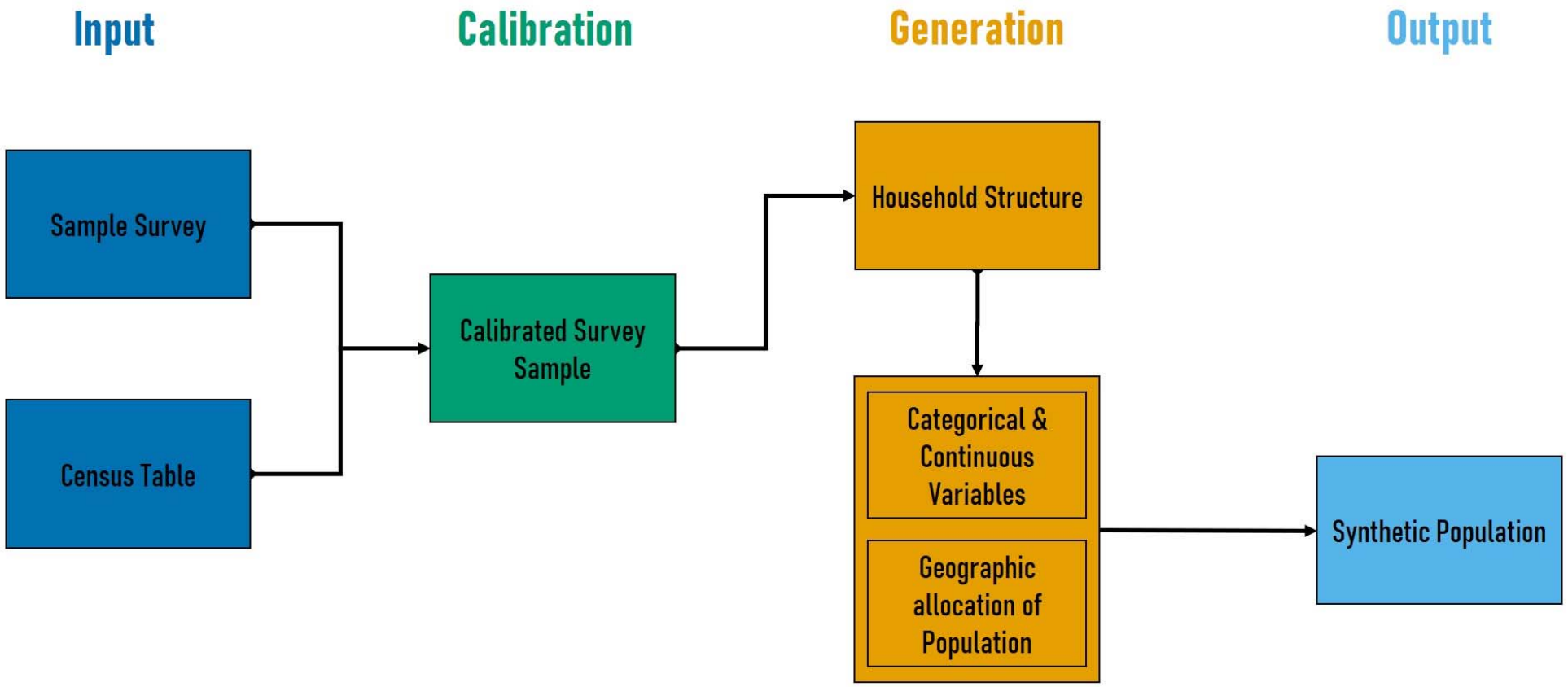
Q: Statistik Austria
www.statistik.at

simPop

- Synthetic data should
  - not reveal sensitive information
  - preserve correlation structure / Quasi-identical distribution

- Synthetic data can be used for
  - Prototype development
  - Micro-Simulation

simPop Workflow



Q: Statistik Austria
www.statistik.at

Initialise population

- simPop $\rightarrow$ specifically designed for synthesising populations (persons living in households)

- Specify inputs

```
data(eusilcS)
# specify input
inp <- specifyInput(data=eusilcS, hhid="db030", hhsize="hsize",
                    weight="rb050", strata="db040")
```

- Initialise synthetic population by defining household structure

```
simPopObj <- simStructure(data=inp, method="direct",
                          basicHHvars=c("age", "rb090"))
```

Simulate variables

- Sequentially add categorical or continuous variables

```
simPopObj <- simCategorical(simPopObj,
                            additional=c("pl030", "pb220a"),
                            method="multinom", nr_cpus=1)
```

- methods = c(“multinom”, “distribution”, “ctree”, “cforest”, “ranger”, “xgboost”)

```
simPopObj <- simContinuous(simPopObj, additional="netIncome",
                           method = "lm",
                           regModel = ~rb090+hsize+pl030+pb220a,
                           nr_cpus=1)
```

- methods = c(“multinom”, “lm”, “poisson”, “xgboost”)

Additional modelling functions

- Additional, more specific, modelling functions

1. Simulate categorical variables taking relationships between household members into account

```
ghanaP <- simRelation(simPopObj = ghanaP, relation = "relate",
                      head = "head",
                       additional = c("nation", "ethnic", "religion"),
                      nr_cpus = 1)
```

Additional modelling functions

2. Simulate components of continuous variables

```
simPopObj <- simComponents(simPopObj=simPopObj, total="netIncome",
  components=c("py010n","py050n","py090n","py100n",
               "py110n","py120n","py130n","py140n"),
  conditional = c("pl030"), replaceEmpty = "sequential", seed=1 )
```

3. Introduce smaller regions to already existing broader regions $\rightarrow$ `simInitSpatial()`
4. Fix age heaping $\rightarrow$ `correctHeaps()`, `correctSingleHeap()`

simPop Workflow

- After variables have been synthesised *calibrate* synthetic population to *fit* selected distributions

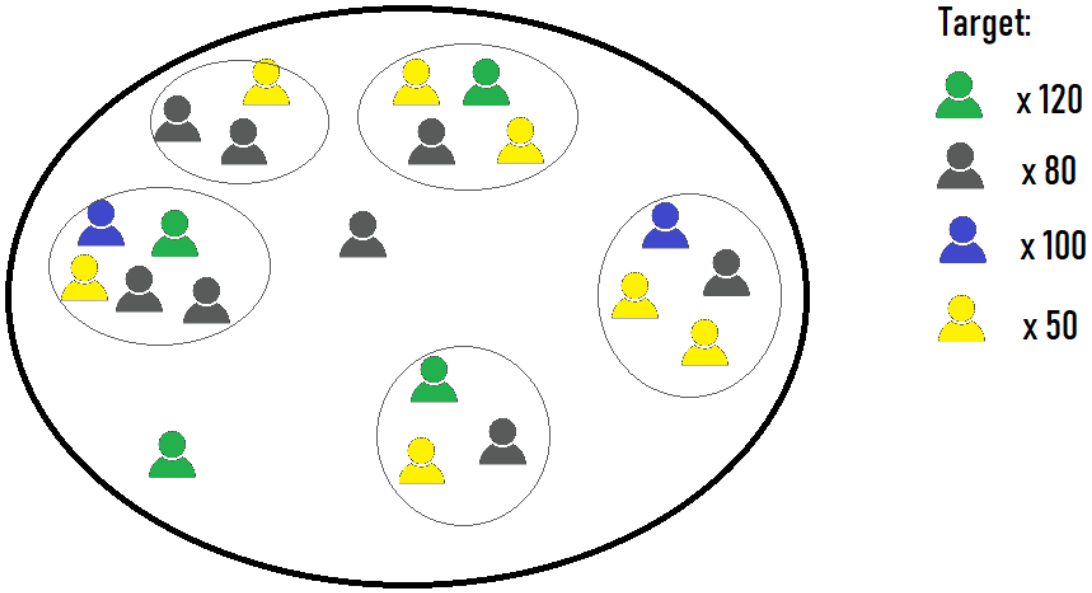
- Simulated annealing algorithm implemented in simPop

```
data("eusilcP")
# add margins
margins <- as.data.frame(
  xtabs(rep(1, nrow(eusilcP)) ~ eusilcP$region +
          eusilcP$gender + eusilcP$citizenship))
colnames(margins) <- c("db040", "rb090", "pb220a", "freq")
simPopObj <- addKnownMargins(simPopObj, margins)
```

```
# run calibration
simPop_adj2 <- calibPop(simPopObj, split="db040",
  temp=1, epsP.factor=0.1,
 epsMinN=10, nr_cpus = 1)
```

Calibrate synth. Population

- Calibrate synthetic population to “fit” selected distributions - how?



- Apply simulated annealing to try to find a local optimum

Q: Statistik Austria
www.statistik.at

Simulated annealing in `simPop`

1. Multiply synthetic population
2. Make innitial selection of households
3. Compare target distribution against distrubtion of selection using an objective function
4. Discard and redraw households with cerain probabilities
5. Check objective function again and accept/reject new solution
6. Repeat 4. and 5. until distributions “differ” by ϵ

Recent improvements

- Multiple distributions allowed on household and personal level

```
calibPop(
  inp, # <- simPopObj
  ...
  hhTables = NULL,
  persTables = NULL
)
```

- Objective / sampling probabilities / termination condition adjusted accordingly

Objective and sampling probability

- Objective, p target margins with $k_1, \ldots, k_p$ number of cells

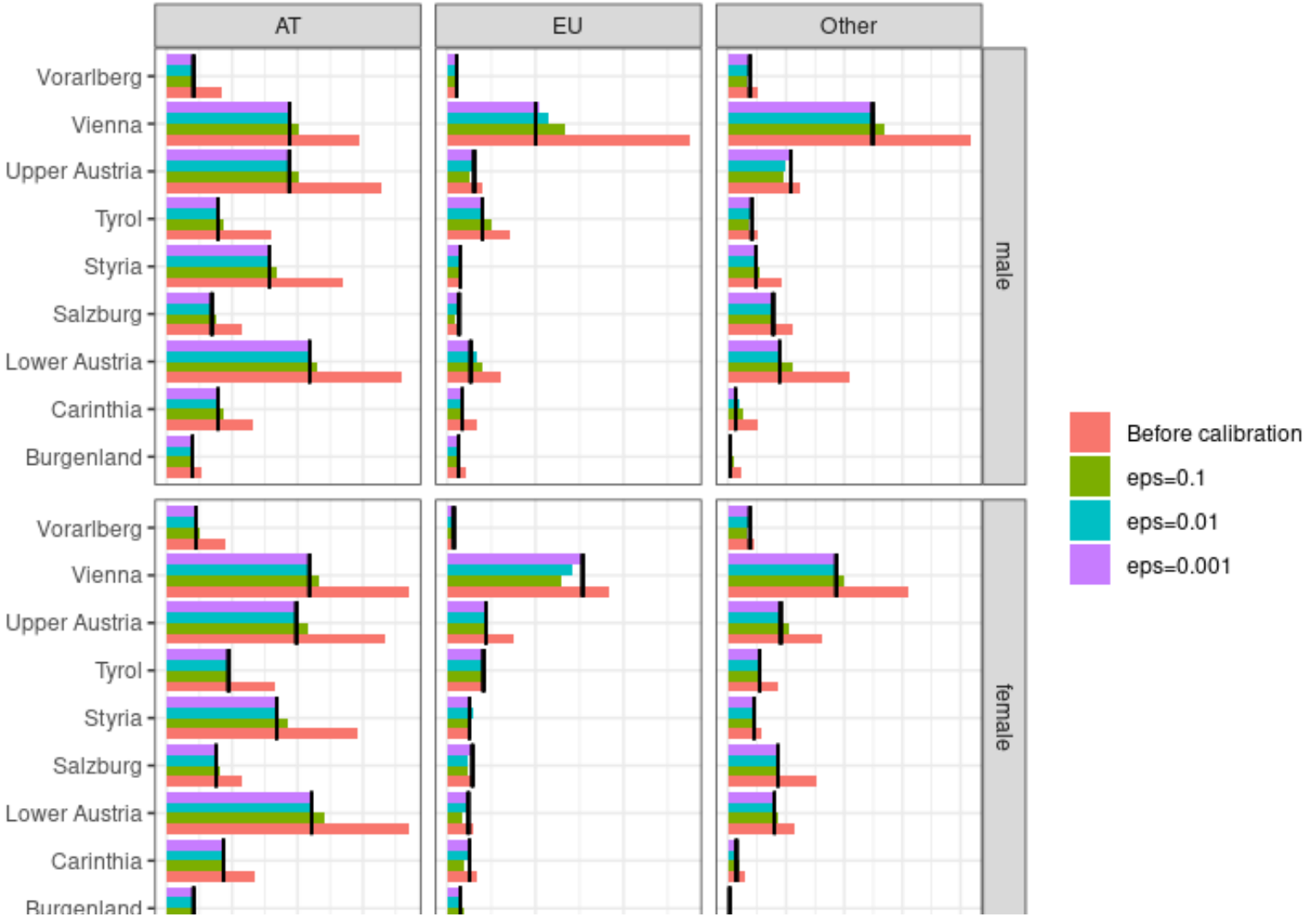
$$\sqrt{\frac{1}{p}\sum_{i=1}^{p}(\sum_{j=1}^{k_i}|n_{i,j}-\tilde{n}_{i,j}|)^2}$$

- Sampling prob. p_l for individual l

$$p_l = \begin{cases} \frac{f_l}{N_l} & \text{if } f_l > 0 \\ \exp(-\sum_{f_l \leq 0} \frac{f_l}{N_l}) \end{cases}; \quad f_l = \frac{1}{p}\sum_{i}^{p}|e_{i,j(i,l)}| \cdot sign(\sum_{i}^{p} e_{i,j(i,l)})$$

$$e_{\cdot\cdot\cdot} = n_{\cdot\cdot\cdot} - \tilde{n}_{\cdot\cdot\cdot}$$

Improvments of calibration



Q: Statistik Austria

www.statistik.at

References

- Münnich, Ralf and Josef Schürle (2003). “On the the simulation of complex universes in the case of applying the German Microcensus”
- Templ, Matthias, Alexander Kowarik, and Peter Filzmoser (2011). “Iterative stepwise regression imputation using standard & robustmethods”. In:Computational Statistics & Data Analysis55.10. DOI:10.1016/j.csda.2011.04.012, ISSN: 0167-9473, pp. 2793–2806.issn:0167-9473.
- Templ, Matthias, Bernhard Meindl, et al. (Aug. 2017). “Simulationof Synthetic Complex Data: The R Package simPop”. In:Journal ofStatistical Software79.doi:10.18637/jss.v079.i10.

Anonymization oftabular data

Prepared by Bernhard Meindl

Why?

Reasons for tabular data control

- **Disclosure risk** exists in aggregated (tabular) data
- **Goals:**
  - Protect statistical units that contribute to table
  - Take into account **trade-off** between risk and data-utility
- **Why?**
  - Publication requirements
  - Legal reasons
- **Statistical tables**:
  - frequency vs. magnitude tables
  - cells relate to each other
  - attackers can make use of those relations

What to protect against?

Group Attribute Disclosure

	female	male	Total
a	4	12	**16**
b	3	0	**3**
c	3	0	**3**
Total	**10**	**12**	**22**

- **"Definition"**: Attacker can learn an attribute about an individual
- **Example**:
  - only one `female` that is single $\rightarrow$ if additional tables exist (e.g `gender` x `maritial_status` x `income`), attackers can derive the income-group for the this unit
- *Further Issue*: $\rightarrow$ **Linked tables** (identical cells can appear in multiple tables)

Packages

Some relevant R-packages

We are going to introduce the following R packages:

- `sdcHierarchies`
- `sdcTable`
- `cellKey / pTable`
- `sdcSpatial`

sdcHierarchies

General Information

- **Goal / Idea**: Define (nested) hierarchies that are used to define statistical tables
- **Features**:
  - allows to programmatically construct (nested) hierarchies
  - Information about (invididual) nodes can be extracted ’ Hierarchies can be imported/exported from/to various formats
  - Shiny-App is provided to interactively create / modify / export hierarchies
- **Web**: `github.com/bernhard-da/sdcHierarchies`

sdcHierarchies

Create hierarchy

```
library(sdcHierarchies)
h <- hier_create(root = "Tot", nodes = LETTERS[1:3])
h <- hier_add(h, root = "A", nodes = paste0("a", 1:2))
h <- hier_add(h, root = "a1", nodes = "a11"); hier_display(h)
```

```
Tot
├─A
│ ├─a1
│ │ └─a11
│ └─a2
├─B
└─C
```

sdcHierarchies

Information about a node

```
1 hier_info(h, "A")
```

```
$name
[1] "A"

$is_rootnode
[1] FALSE

$level
[1] 2

$is_leaf
[1] FALSE

$siblings
[1] "B" "C"
```

sdcTable

General Information

- **Goal / Idea**: Protect statistical tables
- **Features**:
  - Setup of complex statistical tables
  - Identification primary sensitive table cells
  - Protect sensitive cells using different algorithms
  - Allow export of problem instances to τ-Argus
- **Web**: `github.com/sdcTools/sdcTable`

sdcTable

Setup Problem

```
1 library(sdcTable)
2 p <- sdc_testproblem(); print(str(p))
```

```
Formal class 'sdcProblem' [package "sdcTable"] with 8 slots
  ..@ dataObj         :Formal class 'dataObj' [package "sdcTable"] with 7
slots
  .. .. ..@ rawData      :Classes 'data.table' and 'data.frame':    100 obs.
of  5 variables:
  .. .. .. ..$ region            : chr [1:100] "A" "A" "A" "A" ...
  .. .. .. ..$ gender            : chr [1:100] "female" "female" "male"
"male" ...
  .. .. .. ..$ freq              : num [1:100] 1 1 1 1 1 1 1 1 1 1 ...
  .. .. .. ..$ tmpsamplingweights: num [1:100] 1 1 1 1 1 1 1 1 1 1 ...
  .. .. .. ..$ val               : num [1:100] 9 11 10 11 5 7 13 15 13 6 ...
  .. .. .. ..- attr(*, ".internal.selfref")=<externalptr>
  .. .. .. ..- attr(*, "sorted")= chr [1:2] "region" "gender"
  .. .. ..@ dimVarInd    : int [1:2] 1 2
  .. .. ..@ freqVarInd   : int 3
```

sdcTable

Extract Table

```
1 sdcProb2df(p, addDups = FALSE, dimCodes = "original", addNumVars = TRUE)
```

```
# A data frame: 15 × 6
   strID  freq sdcStatus   val region gender
   <chr> <dbl> <chr>      <dbl> <chr>  <chr>
 1 0000    100 s           1284 total  total
 2 0001     55 s            802 total  male
 3 0002     45 s            482 total  female
 4 0100     20 s            198 A      total
 5 0101     18 s            178 A      male
 6 0102      2 s             20 A      female
 7 0200     33 s            344 B      total
 8 0201     14 s            140 B      male
 9 0202     19 s            204 B      female
10 0300     22 s            224 C      total
11 0301     12 s            118 C      male
12 0302     10 s            106 C      female
```

sdcTable

Identify primary sensitive cells

```
1 p <- primarySuppression(p, type = "freq", maxN = 3)
```

- Table of different `"cell-stati"`
  - `"s"`: “safe” for publication
  - `"u"`: primary unsafe
  - `"x"`: secondary suppression

```
1 table(getInfo(p, "sdcStatus"))
```

```
 s  u
14  1
```

sdcTable

Protect table

```
1 p <- protectTable(p, method = "GAUSS")
2 getInfo(p, "finalData")
```

```
# A data frame: 15 × 5
   region gender  Freq   val sdcStatus
   <chr>  <chr>  <dbl> <dbl> <chr>
 1 total  total    100  1284 s
 2 total  male      55   802 s
 3 total  female    45   482 s
 4 A      total     20   198 s
 5 A      male      18   178 x
 6 A      female     2    20 u
 7 B      total     33   344 s
 8 B      male      14   140 s
 9 B      female    19   204 s
10 C      total     22   224 s
11 C      male      12   118 s
12 C      female    10   106 s
```

cellKey - ptable

General Information

- **Goal / Idea**: Persistent perturbation of statistical tables
- **Features**:
  - Allows generation of tables (similar to `sdcTable`)
  - Implements a perturbation algorithm that depends on record- and cell keys
  - Makes use of look-up tables (using `ptable` Package)
  - Different methods for frequency- and magnitude tables
  - Very useful for perturbation of large, linked tables
  - Drawback: Impact of perturbation visible (non-additive results)
- **Web**:
  - `github.com/sdcTools/ptable` | `github.com/sdcTools/cellKey`

cellKey + ptable

Setup

```
library(cellKey)
x <- ck_create_testdata()
tab <- ck_setup(
  x = x,
  rkey = 6, # digits
  dims = list(
    sex = hier_create(root = "Total", nodes = c("male", "female")),
    age = hier_create(root = "Total", nodes = paste0("age_group", 1:6))),
  w = "sampling_weight"
)
tab$print()
```

```
── Table Information ──────────────────────────────────────────────────────────
✓ 21 cells in 2 dimensions ('sex', 'age')
✓ weights: yes
── Tabulated / Perturbed countvars ────────────────────────────────────────────
☐ 'total'
```

cellKey + ptable

Define perturbation parameters

```
p_cnts <- ck_params_cnts(
  ptab = ptable::pt_ex_cnts()
)
print(head(p_cnts$params$ptable))
```

```
   i j           p  v         lb        ub type
1: 0 0 1.00000000  0 0.0000000 1.0000000  all
2: 1 0 0.50833333 -1 0.0000000 0.5083333  all
3: 1 2 0.47500000  1 0.5083333 0.9833333  all
4: 1 3 0.01666667  2 0.9833333 1.0000000  all
5: 2 0 0.16155827 -2 0.0000000 0.1615583  all
6: 2 2 0.55565037  0 0.1615583 0.7172086  all
```

- Assign parameters to specific variable

```
tab$params_cnts_set(v = "total", val = p_cnts)
```

cellKey + ptable

Perturb and evaluate

```
1 tab$perturb(v = "total")
2 tab$print()
```

```
── Table Information ──────────────────────────────────────────────────────────
✓ 21 cells in 2 dimensions ('sex', 'age')
✓ weights: yes
── Tabulated / Perturbed countvars ────────────────────────────────────────────
☒ 'total' (perturbed)
```

- Evaluate / extract results

```
1 tab$freqtab(v = "total")
```

```
# A data frame: 21 × 7
   sex   age        vname   uwc     wc  puwc     pwc
   <chr> <chr>      <chr> <dbl>  <dbl> <dbl>   <dbl>
 1 Total Total      total  4580 275677  4581 275737.
 2 Total age_group1 total  1969 118905  1969 118905
 3 Total age_group2 total  1143  68788  1144  68848.
 4 Total age_group3 total   864  52136   866  52257.
 5 Total age_group4 total   423  25028   425  25146.
```

sdcSpatial

General Information

- **Goal / Idea**: Protect spatial data
- **Features**:
  - Based on functionality from `raster` package
  - Allows to identify sensitive cells
  - Multiple algorithms (removal, smoothing, aggregation) to protect raster-cells are implemented
- **Web**:
  - `github.com/edwindj/sdcSpatial`
  - Helpful vignette: Intro to `sdcSpatial`

Please address queries to
alexander.kowarik@statistik.gv.at
johannes.gussenbauer@statistik.gv.at
bernhard.meindl@statistik.gv.at

STATISTIK AUSTRIA
Guglgasse 13, 1110 Wien

Independent statistics for evidence-based decision making

STATISTICS AUSTRIA
The Information Manager

What is the awesome list?

Curated list of software for official statistics

awesome

www.awesomeofficialstatistics.org

Search or jump to... / Pulls Issues Marketplace Explore

SNStatComp / awesome-official-statistics-software

Unwatch 30 | Unstar 161 | Fork 41

Social interactions

Code | Issues | Pull requests 1 | Actions | Projects | Wiki | Security | Insights

master | Go to file | Add file | Code

Awesome official statistics software awesome

An awesome list of open source statistical software packages useful for creating and accessing official statistics.

An item on this list is awesome because

1. it is free, open source, and available for download and
2. it is confirmed to be used in the production of official statistics by at least one institute or it provides access to official statistics publications.

Criteria

We prefer packages that are easy to install and use, have at least one stable version, and are actively maintained. Contributions are welcome.

An awesome list of statistical software for creating and accessing official statistics

official-statistics gsbpm

Contributors 15

+ 4 contributors

Working together

Contributions

Awesome contributions are welcome, here are ways to do it:

- The GitHub way: send us a pull request to add directly to this list.
- Add an item to the issue tracker issue tracker. (you need a GH account)
- Send an e-mail to `mark dot vanderloo at gmail dot com` or `olav dot tenbosch at gmail dot com` or tweet @markvdloo

License

This work is licensed under a Creative Commons Attribution 4.0 International License.

Open license