

# Package ‘rscc’

September 13, 2021

**Type** Package

**Title** R Source Code Similarity Evaluation by Similarity Coefficients

**Version** 0.1.0

**Date** 2021-09-11

**Description**

Evaluates R source codes by variable and/or functions names. Similar source codes should deliver similarity coefficients near one. Since neither the frequency nor the order of the used names is considered, a manual inspection of the R source code is required to check for similarity. Possible use cases include detection of code clones for improving software quality and of plagiarism amongst students' assignments.

**License** GPL-3

**URL** <https://github.com/sigbertklinke/rscc> (development version)

**Imports** crayon, formatR, highlight, igraph

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**Suggests** rmarkdown, knitr

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Sigbert Klinke [aut, cre]

**Maintainer** Sigbert Klinke <sigbert@hu-berlin.de>

**Repository** CRAN

**Date/Publication** 2021-09-13 12:20:02 UTC

## R topics documented:

as_igraph . . . . .	2
browse . . . . .	2
matrix2dataframe . . . . .	3
similarity_coeff . . . . .	4
sims . . . . .	5
sourcecode . . . . .	6

**Index**[7](#)


---

as_igraph	<i>as.igraph</i>
-----------	------------------

---

**Description**

Converts a data frame of similarity coefficients into a graph.

**Usage**

```
as_igraph(x, ...)
```

**Arguments**

x	a similarity object
...	further parameters used by <a href="#">igraph::graph_from_adjacency_matrix</a>

**Value**

an igraph object

**Examples**

```
files <- list.files(path=system.file("examples", package="rsc"), pattern="*.R$", full.names = TRUE)
prgs <- sourcecode(files, basename=TRUE)
coeff <- similarities(prgs)          # variables only
# a similarity coefficients equal to zero does not create an edge!
g <- as_igraph(coeff, diag=FALSE)
# thicker edges have higher similarity coefficients
plot(g, edge.width=1+3*igraph::E(g)$weight)
```

---

browse	<i>browse</i>
--------	---------------

---

**Description**

Creates a temporary HTML file with source codes and opens it into a browser using `browseURL`.

**Usage**

```
browse(prgs, simdf, n = sum(simdf[, 3] > 0), width.cutoff = 60, css = NULL)
```

**Arguments**

prgs	sourcecode object
simdf	similarity object
n	integer: number of comparisons to show (default: $\text{sum}(\text{simf}[,3]>0)$ )
width.cutoff	integer: an integer in [20, 500]: if a line's character length is at or over this number, the function will try to break it into a new line (default: 60)
css	character: file name of CSS style for highlighting the R code

**Value**

invisibly the name of the temporary HTML file

**Examples**

```
# example files are taken from https://CRAN.R-project.org/package=SimilaR
files <- list.files(system.file("examples", package="rsc"), "*.R$", full.names=TRUE)
prgs <- sourcecode(files)
simil <- similarities(prgs)
if (interactive()) browse(prgs, simil)
```

---

matrix2dataframe	<i>matrix2dataframe</i>
------------------	-------------------------

---

**Description**

Converts a numeric matrix to a data frame with decreasing or increasing values: First column row index, second column col index and third column the value. If the matrix is symmetrical, only the upper triangle is taken into account.

**Usage**

```
matrix2dataframe(m, decreasing = TRUE, tol = 1e-09)
```

**Arguments**

m	numeric: a matrix of values
decreasing	logical: should the sort order be increasing or decreasing (default: TRUE)
tol	numeric: tolerance used to detect if matrix is symmetrical (default: 1e-9)

**Value**

a data frame with an attribute matrix with m

**Examples**

```
# non-symmetric
x <- matrix(runif(9), ncol=3)
matrix2dataframe(x)
# symmetric
x <- x+t(x)
matrix2dataframe(x)
```

---

similarity_coeff	<i>similarity_coeff</i>
------------------	-------------------------

---

**Description**

Computes a similarity coefficient based on the unique elements set1 and set2 in relation to setfull. If setfull is NULL then setfull is set to unique(c(set1, set2)). For more details, see the vignette vignette("rsc").

**Usage**

```
similarity_coeff(
  set1,
  set2,
  setfull = NULL,
  coeff = c("jaccard", "braun", "dice", "hamann", "kappa", "kulczynski", "ochiai",
            "phi", "russelrao", "matching", "simpson", "sneath", "tanimoto", "yule")
)
```

**Arguments**

set1	vector: elements to compare
set2	vector: elements to compare
setfull	vector: elements to compare (default: NULL)
coeff	character: coefficient to compute (default: "jaccard"), abbreviations can be used

**Value**

a numeric similarity coefficient

**Examples**

```
s1 <- 1:3
s2 <- 1:5
similarity_coeff(s1, s2)
s1 <- letters[1:3]
s2 <- LETTERS[1:5]
similarity_coeff(s1, s2)
```

sims

*similarities***Description**

sims and similarities both calculate for each pair of source code objects the similarity coefficients and return a data frame with the coefficients in descending order. A larger coefficient means a greater similarity. If

- type=="vars" then the results of all.vars(.),
- type=="funs" then the results of setdiff(all.names(.),all.vars(.), and
- type=="names" then the results of all.names(.)

are used.

**Usage**

```
sims(...)
```

```
similarities(
  prgs,
  type = c("vars", "funs", "names"),
  all = FALSE,
  ignore.case = TRUE,
  minlen = 2,
  decreasing = TRUE,
  tol = 1e-09,
  same.file = TRUE,
  coeff = c("jaccard", "braun", "dice", "hamann", "kappa", "kulczynski", "ochiai",
            "phi", "russelrao", "matching", "simpson", "sneath", "tanimoto", "yule")
)
```

**Arguments**

...	all parameters in sims are given to similarities
prgs	sourcecode object
type	character: either "vars", "funs", "names" (default: "vars")
all	logical: should the similarity coefficients computed based on all sourcecode objects or just the two considered (default: FALSE)
ignore.case	logical: If TRUE, case is ignored for computing (default: TRUE)
minlen	integer: minimal name length to be considered (default: 2)
decreasing	logical: should the sort order be increasing or decreasing (default: TRUE)
tol	numeric: tolerance used to detect if matrix is symmetric (default: 1e-9)
same.file	logical: should be similarities kept if in same file (default) or set to zero (default: TRUE)
coeff	character: coefficient to compute (default: "jaccard"), abbreviations can be used

**Value**

a data frame with the results

**Examples**

```
# example files are taken from https://CRAN.R-project.org/package=SimilaR
files <- list.files(system.file("examples", package="rsc"), "*.R$", full.names=TRUE)
prgs <- sourcecode(files)
names(prgs) <- basename(names(prgs))
similarities(prgs)          # variables only
similarities(prgs, type="f") # functions only
similarities(prgs, type="n") # all names
```

---

sourcecode

*sourcecode*

---

**Description**

reads and parses files with R source code

**Usage**

```
sourcecode(x, ...)
```

```
## Default S3 method:
```

```
sourcecode(x, basename = FALSE, silent = FALSE, minlines = -1, ...)
```

**Arguments**

x	character: filenames
...	unused
basename	logical: should only the basename used for sourcecode objects (default: FALSE)
silent	logical: should the report of messages be suppressed (default: FALSE)
minlines	integer: only expressions with minlines lines are considered (default: -1), if minlines<0 then whole files will be considered

**Value**

a sourcecode object

**Examples**

```
# example files are taken from https://CRAN.R-project.org/package=SimilaR
files <- list.files(system.file("examples", package="rsc"), "*.R$", full.names=TRUE)
prgs <- sourcecode(files)
```

# Index

`as_igraph`, [2](#)

`browse`, [2](#)

`igraph::graph_from_adjacency_matrix`, [2](#)

`matrix2dataframe`, [3](#)

`similarities(sims)`, [5](#)

`similarity_coeff`, [4](#)

`sims`, [5](#)

`sourcecode`, [6](#)