

# Package ‘SUSY’

July 21, 2025

**Version** 0.1.0

**Title** Surrogate Synchrony

**Suggests** gtools

**Description** Computes synchrony as windowed cross-correlation based on two-dimensional time series in a text file you can upload. 'SUSY' works as described in Tschacher & Meier (2020) <[doi:10.1080/10503307.2019.1612114](https://doi.org/10.1080/10503307.2019.1612114)>.

**License** GPL-2

**URL** <https://wtschacher.github.io/SUSY/>

**BugReports** <https://github.com/wtschacher/SUSY/issues>

**NeedsCompilation** no

**Author** Wolfgang Tschacher [aut, cre],  
David Tschacher [ctb],  
Jan Gorecki [ctb]

**Maintainer** Wolfgang Tschacher <[wolfgang.tschacher@upd.unibe.ch](mailto:wolfgang.tschacher@upd.unibe.ch)>

**Repository** CRAN

**Date/Publication** 2022-11-24 09:00:13 UTC

## Contents

|                              |   |
|------------------------------|---|
| as.data.frame.susy . . . . . | 2 |
| plot.susy . . . . .          | 3 |
| print.susy . . . . .         | 4 |
| susy . . . . .               | 5 |

|              |          |
|--------------|----------|
| <b>Index</b> | <b>7</b> |
|--------------|----------|

---

|                    |                                             |
|--------------------|---------------------------------------------|
| as.data.frame.susy | <i>susy to data.frame conversion method</i> |
|--------------------|---------------------------------------------|

---

## Description

Turns susy class object into a data.frame.

## Usage

```
## S3 method for class 'susy'  
as.data.frame(x, row.names=NULL, optional=FALSE, corr.no.abs=TRUE, ...)
```

## Arguments

|             |                                                                           |
|-------------|---------------------------------------------------------------------------|
| x           | A susy object.                                                            |
| row.names   | Ignored, only for consistency to generic as.data.frame method.            |
| optional    | Ignored, only for consistency to generic as.data.frame method.            |
| corr.no.abs | Logical, defaults to TRUE display correlation without the absolute value. |
| ...         | Ignored.                                                                  |

## Value

Returns data.frame.

## See Also

[susy](#)

## Examples

```
n = 1000  
data = data.frame(  
  var1 = runif(n, 300, 330),  
  var2 = runif(n, 300, 330)  
)  
res = susy(data, segment=30L, Hz=15L)  
as.data.frame(res)
```

---

|           |                         |
|-----------|-------------------------|
| plot.susy | <i>susy plot method</i> |
|-----------|-------------------------|

---

## Description

Generate plot(s) for a susy object.

## Usage

```
## S3 method for class 'susy'
plot(x, type=c(4, 5), ...)
```

## Arguments

|      |                                                                                                                                                                                                                                                                       |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| x    | A susy object.                                                                                                                                                                                                                                                        |
| type | Numeric, specifies the types of plot, defaults to c(4, 5). <ol style="list-style-type: none"> <li>1. <i>GMcrosscorrs</i></li> <li>2. <i>synchrony by segments</i></li> <li>3. <i>GM-Z</i></li> <li>4. <i>time series plot</i></li> <li>5. <i>Z not abs</i></li> </ol> |
| ...  | Ignored.                                                                                                                                                                                                                                                              |

## Details

Method can generate multiple types of plots by providing numeric vector to type argument. Note it will generate plots for each pair (cross computation) in x, so the final number of plots is `length(x) * length(type)`.

## Value

Returns NULL invisibly. Generate plot(s) as a side effect.

## See Also

[susy](#)

## Examples

```
n = 1000
data = data.frame(
  var1 = runif(n, 300, 330),
  var2 = runif(n, 300, 330),
  var3 = runif(n, 300, 330)
)
res = susy(data, segment=30L, Hz=15L, permutation=TRUE)
plot(res, type=c(3,5))
```

|            |                          |
|------------|--------------------------|
| print.susy | <i>susy print method</i> |
|------------|--------------------------|

---

## Description

Prints information about an susy object.

## Usage

```
## S3 method for class 'susy'  
print(x, corr.no.abs=TRUE, legacy=FALSE, ...)
```

## Arguments

|             |                                                                                                                           |
|-------------|---------------------------------------------------------------------------------------------------------------------------|
| x           | A susy object.                                                                                                            |
| corr.no.abs | Logical, defaults to TRUE display correlation without the absolute value.                                                 |
| legacy      | Logical, defaults to FALSE, when TRUE print will produce an output that matches the output of legacy SUSY implementation. |
| ...         | Extra arguments passed to print.data.frame method.                                                                        |

## Value

Returns x invisibly. Display output to console as a side effect.

## See Also

[susy](#)

## Examples

```
n = 1000  
data = data.frame(  
  var1 = runif(n, 300, 330),  
  var2 = runif(n, 300, 330)  
)  
res = susy(data, segment=30L, Hz=15L)  
res  
print(res, corr.no.abs=FALSE)  
print(res, digits=4)  
print(res, legacy=TRUE)
```

**Description**

Cross-correlations of two time series are computed up to a specific lag in seconds `maxlag`. Cross-correlation is done within segment of the time series. The size of segments `segment` can be chosen in seconds. Aggregation is then performed by transforming correlations to Fisher's Z, computing mean Z in each segment, then across all segments of the time series. Segment shuffling is used to create surrogate time series, on which the same computations are run. This provides effect sizes ES. SUSY provides these different synchrony measures for each twin time series: mean Z and ES of mean Z; mean absolute\_Z and ES of mean absolute\_Z.

**Usage**

```
susy(x, segment, Hz, maxlag=3L, permutation=FALSE,
     restrict.surrogates=FALSE, surrogates.total=500)
```

**Arguments**

|                                  |                                                                                                                                                                                                                                                                      |
|----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>x</code>                   | A data.frame of numeric columns.                                                                                                                                                                                                                                     |
| <code>segment</code>             | Integer, size in seconds. Must not be smaller than $2 * \text{maxlag}$ , must not be larger than half the the time series ( $\text{nrow}(x)/2$ ).                                                                                                                    |
| <code>Hz</code>                  | Integer, frames per second (sampling rate).                                                                                                                                                                                                                          |
| <code>maxlag</code>              | Integer, maximum lag for <code>ccf</code> in seconds. Default 3 seconds.                                                                                                                                                                                             |
| <code>permutation</code>         | Logical, default FALSE requires <code>x</code> to have even number of columns which are processed in pairs (1-2, 3-4, etc.). When permutation is TRUE then function computes all pairs combinations between columns provided in <code>x</code> ( $n*(n-1)/2$ pairs). |
| <code>restrict.surrogates</code> | Logical, default FALSE. Restrict the number of surrogates or not.                                                                                                                                                                                                    |
| <code>surrogates.total</code>    | Numeric, the number of generated surrogates, default 500. Ignored when <code>restrict.surrogates</code> is FALSE (default).                                                                                                                                          |

**Details**

Segments are non-overlapping, and the number of segments that fit into the time series may have a remainder (usually a few seconds at the end of the time series), which is not considered.

**Value**

Object of class `susy` is returned. Each cross correlation pair is an element in resulting object.

**See Also**

[plot.susy](#), [as.data.frame.susy](#), [print.susy](#)

**Examples**

```
n = 1000
data = data.frame(
  var1 = runif(n, 300, 330),
  var2 = runif(n, 300, 330),
  var3 = runif(n, 300, 330)
)

## use only first two columns
res = susy(data[, 1:2], segment=30L, Hz=15L)
length(res)
names(res)

## use all columns and permutation
res = susy(data, segment=30L, Hz=15L, permutation=TRUE)
length(res)
names(res)

## print susy
res
print(res, legacy=TRUE)

## plot susy
plot(res)
plot(res, type=1:2)
```

# Index

## \* **data**

as.data.frame.susy, [2](#)

plot.susy, [3](#)

print.susy, [4](#)

susy, [5](#)

as.data.frame.susy, [2](#), [5](#)

ccf, [5](#)

plot.susy, [3](#), [5](#)

print.susy, [4](#), [5](#)

susy, [2-4](#), [5](#)